Enterprise Developer チュートリアル

メインフレーム PL/I 開発: CICS Visual Studio 2022 編

1. 目的

本チュートリアルでは、PL/I 言語で書かれた CICS 命令を含むソースをオープン環境へ移行後、Visual Studio 2022 を使用してプロジェクトの作成、コンパイル、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 Microsoft Visual Studio 2022 がインストールされていること
- Windows 開発環境に Enterprise Developer 9.0 for Visual Studio 2022 がインストール済であること。
- TN3270 エミュレータがインストール済で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. PL/I ソリューションのインポート
- 4. プロジェクトプロパティの確認
- 5. ビルドの実行
- 6. 文字エンコーディングの設定
- 7. Enterprise Server インスタンスの設定
- 8. Enterprise Server インスタンス開始と確認
- 9. CICS の実行
- 10. PL/I ソースの動的デバッグ
- 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。

- Visual Studio のソリューションを保存するフォルダを C:¥ 直下に作成します。
 例)C:¥VS¥PLI
- 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 CICSDEMO フォルダを作成した C:¥VS¥PLI ヘコピーします。

例)C: ${}$ Users ${}$ Public ${}$ Documents ${}$ Micro Focus ${}$ Enterprise Developer ${}$ Samples ${}$ PLI-VS ${}$ CICSDEMO



また、実行時に使用する system フォルダを C:¥VS¥PLI¥CICSDEMO 直下に新規作成してください。

ディスク (C:)	> VS >	PLI > CICSDEM	0 >
名前		~	
📑 sys	stem		

3.2 Visual Studio の起動

1) Visual Studio 2022 を起動します。

Visual Studio 2022

2) 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] プルダウンメニューの [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 ツール] > [指令] > [PL/I] > [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。_____



3.3 PL/I ソリューションのインポート

 1) 用意した例題ソリューションを表示します。[ファイル] プルダウンメニューから [開く] > [プロジェクト/ソリ ューション] を選択し、[プロジェクトを開く] ウィンドウにて前項でコピーした C:¥VS¥PLI¥CICSDEMO¥CICS に存在する CICS.sln を選択後 [開く] ボタンをクリックします。

📢 プロジェクト/ソリューションを開く			×
← → · · ↑	イスク(C:) → VS → PLI → CICSE	emo → cics ~ Č	CICSの検索 ・
整理 ▼ 新しいフォルダー			III 🕶 🔟 🔇
💻 PC	^ 名前 [^]	更新日時	種類 サイズ
🧊 3D オブジェクト	🖻 CICS.pliproj	2021/05/12 20	:19 PL/Iプロジェクト
🚮 My Mainframe 2.0	GICS.sin	2017/03/08 21	:40 Microsoft Visual S
🖊 ダウンロード			
🔜 デスクトップ			
🔮 ドキュメント			
📰 ピクチャ			
🔚 ビデオ			
🎝 ミュージック			
🏪 ローカル ディスク (C:)			
🔐 CD ドライブ (E:)			
🚎 Takahashi (¥¥VBoxSvr) (Z:)	v <		>
ファイル名(N):	CICS.sIn	~	すべてのプロジェクト ファイル (*.sln;* 〜
		□ プロジェクトを読み込まない (L)	開く(<u>Q</u>) キャンセル

2) 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。



3) [ソリューション エクスプローラー] にインポートしたソリューションと 3 つのプロジェクトが表示されます。



3.4 プロジェクトプロパティの確認

この例題は Subroutines プロジェクトで作成される LIB オブジェクトを CICS プロジェクトがリンクして DLL を生成する内容になっています。まずはプロジェクトの設定値を確認します。

1) サブとなるプロジェクトの設定を確認します。[ソリューション エクスプローラー] 内 Subroutines プロジェクトの [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 [アプリケーション] を選択すると、LIB 生成を指定していることが確認できます。

アプリケーション	構成(C): なし、 ~	
SQL		
Debug	フラットフォーム(M): なし ~	
従属パス		
BMS	アプリケーション	
IMS	ファイル名 Subroutines	s.lib
PL/I	出力の種類 Static Libra	ry (.lib)
	出力の名前 Subroutines	3

3) 左側ツリービューの [PL/I] を選択すると下記項目が確認できます。出力パスを変更して保存してください。

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。 x86 が選択されており 32-bit 稼働が指定されています。
出力パス	生成された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が、こ こでは Enterprise Server インスタンスの設定値と合わせるため に .¥bin¥Debug¥ 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。

アプリケーション	構成(C)・ アクティブな (Dabug) > プラットフォーム(M)・ アクティブな (v86)	~
SQL	(1977) (1977) (Debug) · · · · · · · · · · · · · · · · · · ·	
Debug		
従属パス	✓ 全般	
BMS	CICS プリプロセッサを有効にする	いいえ
	DLI プリプロセッサを有効にする	No
IMS	プラットフォームターゲット	x86
PL/I	マクロプリプロセッサを有効にする	いいえ
PL/I リンク	出力パス	.¥bin¥Debug¥
	 PL/I Compile Settings 	
アセンフラ	EXEC プリプロセッサ オプション (-optexec)	plitest
	エンディアン (-bigendian)	
	カバレッジ データを生成する (-testcover)	いいえ
	システム	
	データ収集の有効化 (-dc)	はい
	デバッグ用にコンパイル (-debug)	はい
	リストファイルを出力 (-1)	いいえ
	最適化レベル (-opt)	-noopt
	追加オプション	-margins 2,120

- 4) 次にメインとなるプロジェクトの設定を確認します。[ソリューション エクスプローラー] 内 CICS プロジェク トの [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。
- 5) 左側ツリービューの [アプリケーション] を選択すると、DLL 生成を指定していることが確認できます。

アプリケーション	構成(<u>C</u>): なし	~
SQL	The state of the state	
Debug	ノフットノオーム(<u>M</u>): なし	
従属パス	₩ 2↓ □	
BMS	アプリケーション	
IMS	ファイル名	
PI /I	出力の種類	Dynamic Library (.dll)
	出力の名前	CICS

 左側ツリービューの [従属パス] を選択すると、同じソリューションに含まれている CICS-Includes プロ ジェクトのパスを PL/I インクルードパスへ指定していることがわかります。このように指定することで共有イ ンクルードファイルへのアクセスが可能になります。

アプリケーション	構成(<u>C</u>): なし ~
SQL	7
Debug)))))))))(<u>)</u>): (<u>)</u>): (<u>)</u>): (<u>)</u>):
従属パス	種類: PL/I インクルードパス ~
BMS	
IMS	¥CICS-Includes

メインフレーム PL/I 開発:CICS Visual Studio 2022 編

7) 左側ツリービューの [PL/I] を選択すると下記項目が確認できます。出力パスを変更して保存してください。

項目名	説明
CICS プリプロセッサを有効にする	プログラムでは EXEC CICS 構文を使用しているため"はい"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。x86 が選択されており 32-bit 稼働が指定 されています。
出カパス	生成された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が、 ここでは Enterprise Server インスタンスの設定値と合わせるため に .¥bin¥Debug¥ を指定します。
システム	CICS として機能させるため CICS を選択します。他に -ims と -mvs が選択可能です。
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。

アプリケーション	構成(C): アクティブな (Debug) 🗸 プラットフォーム(M):	アクティブな (x86)
SQL		7 7 7 17 10 (100)
Debug		
従属パス	CICS プリプロセッサを有効にする	はい
BMS	DLI プリプロセッサを有効にする	No
55	プラットフォームターゲット	x86
IMS	マクロプリプロセッサを有効にする	はい
PL/I	出力パス	.¥bin¥Debug¥
PL/Lリンク	 PL/I Compile Settings 	
	EXEC プリプロセッサ オプション (-optexec)	
アセンノフ	エンディアン (-bigendian)	
	カバレッジ データを生成する (-testcover)	いいえ
	システム	CICS (-cics)
	データ収集の有効化 (-dc)	はい
	デバッグ用にコンパイル (-debug)	はい
	リストファイルを出力 (-1)	いいえ
	最適化レベル (-opt)	-noopt
	追加オプション	

8) 左側ツリービュー [PL/I リンク] を選択すると、サブプロジェクトで生成した LIB ファイルをリンクさせてい ることが確認できます。前項で指定した LIB の出力パスを [リンクする LIB] に指定して保存します。

アプリケーション SOL	構成(<u>C</u>): アクティブな (Debug) > プラットフォーム(<u>M</u>): アクティブな (x86) >
Debug	
従属パス	Y PL/I Link Settings
BMS	<u>کرج</u> ک
	リンクする LIB "¥CICS-Subs¥bin¥Debug¥subroutines.lib"
IMS	追加オプション
PL/I	
PL/I リンク	

5) 左側メニュー [Debug] を選択し、[プロファイル] には CICS を、[起動] には CICS を、[トランザクション] には TTEA を指定し、設定を保存します。

アプリケーション	構成(C): なし ~ ブラットフォーム(M): なし	
SQL		
Debug	プロファイル: CICS	
従属パス	起動: CICS	
BMS		
IMS	ユーザー:	
PL/I	端末:	
PL/I リンク	トランザクション: TTEA	- Debug - x86
アセンブラ	リンクレベル プログラム:	ルを上書き保存 (Ctrl+S)

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 CICS 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に指定した DLL ファイル等が作成されていることを確認します。

スク(C:) > VS > PLI > CICSDE	MO > CICS > bin > Debug
名前	^
CCMCS.MOD	
STAFF02.adt	
STAFF02.dcf	
STAFF02.DLL	
🔄 STAFF02.pdb	

3.6 文字エンコーディングの設定

Enterprise Server インスタンスを運用、管理する Enterprise Server Common Web Administration (以降 ESCWA)では、スプールやデータ内容などに含まれる日本語を正しく表示させるために、事前に文字セット を所定のフォルダへ展開します。製品マニュアルの「リファレンス > コードセットの変換 > CCSID 変換テーブ ルのインストール > CCSID 変換テーブルをインストールするには」を参照しながら進めてください。

1) CCSID 変換テーブルをインストールします。

製品マニュアルにリンクされている下記の IBM CCSID 変換テーブルを、Web ブラウザから任意のフォル ダヘダウンロードします。アドレスは変更される可能性がありますので、製品マニュアルにてご確認ください。 <u>http://www.microfocus.com/docs/links.asp?vc=cdctables</u>

- 製品インストールフォルダ配下の etc フォルダに CCSID フォルダがない場合はこれを作成します。
 例)C:¥Program Files (x86)¥Micro Focus¥Enterprise Developer¥etc¥CCSID
- 3) ダウンロードファイルに含まれている Package2.zip を展開します。
- 4) 展開した Package2 フォルダに含まれる IBM-932.zip を展開します。
- 5) 展開した IBM-932 フォルダを切り取り、作成した CCSID フォルダ配下へ貼り付け、14 ファイルが含ま れていることを確認します。

-	🔜 👳 IBM-932					- 🗆	×
ファイル	ホーム 共有 表示						~ ?
$\leftarrow \rightarrow$	 	→ Micro Focus → Enterpr	ise Developer >	etc > CCSID > IBN	1-932 ~ ඊ	IBM-932の検索	م
^	名前	種類 ^	サイズ	更新日時			^
	03A434B0.MU-R-A2	MU-R-A2 ファイル	28 KB	1999/10/14 10:04			
	03A434B0.MU-R-A3	MU-R-A3 ファイル	28 KB	2002/01/15 11:58			
	03A434B0.MU-R-D	MU-R-D ファイル	28 KB	1999/10/14 10:04			
	03A434B0.PACKAGE	PACKAGE ファイル	5 KB	2002/01/16 10:37			
	34B003A4.RPMAP12A	RPMAP12A ファイル	336 KB	1999/10/14 10:03			
	34B003A4.RPMAP120	RPMAP120 ファイル	336 KB	1999/10/14 10:03			
	03A434B0.TPMAP11A	TPMAP11A ファイル	329 KB	1999/10/14 10:02			
	03A434B0.TPMAP12A	TPMAP12A ファイル	329 KB	2002/01/15 11:58			
	03A434B0.TPMAP110	TPMAP110 ファイル	329 KB	1999/10/14 10:02			
4	34B003A4.UM-E-A21	UM-E-A21 ファイル	54 KB	1999/10/14 10:04			
2	34B003A4.UM-E-D12	UM-E-D12 ファイル	54 KB	1999/10/14 10:04			
5	03A434B0.UPMAP12A	UPMAP12A ファイル	446 KB	1999/10/14 10:01			
	03A434B0.UPMAP13A	UPMAP13A ファイル	490 KB	2002/01/15 13:42			
► 14 個の項	03A434B0.UPMAP120	UPMAP120 ファイル	447 KB	1999/10/14 10:01			▼

詳細については、製品マニュアルの「ディプロイ > 構成および管理 > Enterprise Server の構成および 管理 > Enterprise Server Common Web Administration > [Native] > [Directory Servers] > リージョンとサーバー > リージョン > エンタープライズ サーバー リージョンの文字エンコーディングの サポート」をご参照ください。

6) Windows サービスとして起動している Micro Focus Enterprise Server Common Web Administration を再起動し、インストールした CCSID をロードさせます。

Q #-ピス				-		×
ファイル(E) 操作(<u>A</u>)	表示(V) ヘルプ(H)					_
(+ + 💼 🗑 🖉	à 📴 🖬 🖬 🕨 🔳 🕪 👘					
🔍 サービス (ローカル)	・サービス (ローカル)					
	Micro Focus Enterprise Server	4前 ^	説明	状態	スタート	F: ^
	Common Web Administration	California CES daemon	Micro Focus CES daemon	実行中	自動	
	サービスの使い	Server Micro Focus Directory Server	Micro Focus Directory Server	実行中	自動	
	サービスの再起動	Children Micro Focus EA Integration Service Host	Micro Focus EA Integration Service Host	実行中	自動	
		Micro Focus Enterprise Analyzer Web Service	Micro Focus Enterprise Analyzer Web Service	実行中	自動	
	5m 00	🙀 Micro Focus Enterprise Server Common Web Administration	Micro Focus Enterprise Server Common Web	実行中	自動	
	説明: Misso Focus Enternico Sonior	Alicro Focus Event Monitor Service Shutdown Coordinator	Micro Focus Event Monitor Service Shutdow	実行中	自動	
	Common Web Administration	Alicro Focus SEE Administration Server	Micro Focus SEE Administration Server		手動	
		Alicro Focus SEE Listener	Micro Focus SEE Listener		手動	
		Micro Focur SEE Monitor	Micro Focur SEE Monitor	*汗曲	0.25	×
	1418 (BUT)	<				>
	(当我人傳筆)					

3.7 Enterprise Server インスタンスの設定

 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 [Micro Focus Server] を右クリックして [管理] を選択します。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。

	(ローカル)	
サーバー エクスプローラー	Micro Focus Directory Server	名前
ひ × 93 雪 🗞 🕒	サ <u>ービスの停止</u> サービスの再起動	MessagingService_2f50d MFADLDS MFADLDS
👂 🚆 Micro Focus Analysis Server		A MFES
👂 🚆 Micro Focus ES.NET Server	説明: Micro Focus Directory Server	Micro Focus AutoPass Daemon
🔺 🧮 Micro Focus Server		Micro Focus Directory Server
管理(A)		Micro Focus EA Integration Service Host Micro Focus Enterprise Server Common Web Administration

 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 ESCWA が表示されます。 ESCWA では Micro Focus Directory Server サービスのポートへ接続して登録された Enterprise Server インスタンスを管理するため、まず接続を作成します。既に存在している場合には、インスタンス作成 に進んでください。 左側ペインで [Directory Server] をクリックし、右側ペインの [追加] ボタンをクリックします。



[ホスト] は localhost またはマシンの IP アドレスを、 [ポート] は Micro Focus Directory Server の

追加ウィンドウでは [名前] は任意ですが、ここでは Default を、

デフォルトポートである 86 を入力して [保存] ボタンをクリックします。

Default		
ホスト *		
localhost		
ポート*		
86		
説明		
* 入力必須の項目で	<u>- </u>	

 3) 例題のフォルダには Enterprise Server インスタンスのサンプルが含まれており、これをインポートします。 C:¥VS¥PLI¥CICSDEMO¥CICS_SERVERS.xml がインポート対象のファイルです。 作成した [Default] 接続をクリックし、右側ペインの [インポート] ボタンをクリックします。

リージョンおよびサーバー リスト 〇 * 新規作成 自すべて削除 ④ エクスポート 創 コピー 〇 更新

4) [インポートタイプの選択] では [XML] を選択して [次へ] をクリックします。

		_
土	インボート タイプの選択	<
インポート	○ JSON クライアントブラウザが実行されているホスト上のファイルを選択して、json ファイルをインボートしま す。	
■ ^{手順1:インポート タイ} イ プ	 XML クライアントブラウザが実行されているホスト上のファイルを選択して、xml ファイルをインポートしま 	
┣ 手順2.アップロード	す。 〇 レガシー Directory Serverが実行されているホスト上のディレクトリの場所を選択して、レガシー リボジトリ (dat ファ イルのディレクトリ)をインボートします。	

5) [ファイルのアップロード] をクリックし、XML ファイルを指定します。

企	ファイルのアップロード	×
インポート		
≣ ^{手順1:} インポートタイ プ	クリックまたはドラッグ&ドロップしてファイルをアップロード	
🖬 手順 2: アップロード		

6) [ファイルを受け付けました] メッセージを確認後、[次へ] ボタンをクリックします。

ファイルのアップロード	×
クリックまたはドラッグ&ドロップしてファイルをアップロード	
ファイルを受け付けました。続行してください。	

7) 手順3 では [次へ] をクリックします。

土	インボート オプション
インポート	🗌 すべてのセキュリティ構成のインポートと置換* 💡
	□ 既存のリージョンの上書き* 💡
■ 手順 1: インポートタイ ■ プ	
🖪 手順 2: アップロード 🗸	
☆ 手順 3: オプション	
土 手順4: インポート	

8) 手順4 では [インポート] をクリックします。

<u>↑</u> インポート	インボートの実行 <u> </u> インボート
■ 手順 1: インポート タイ プ	選択されたインポート オプション
🚹 手順 2: アップロード 🗸	• すべてのセキュリティ構成のインポートと置換: X
✿ 手順 3:オプション ✓	 ・
▲ 手順4: インポート	

9) インポートの完了が通知されますので、[完了] ボタンをクリックします。

インボートの実行 インポートに成功しました。終了する場合は、「完了」を、他の xml ファイルをインポートする場合、「戻 る」を押します。

10)32 ビットアプリケーション稼働用の STAFF と、64 ビットアプリケーション稼働用の STAFF64 インスタ ンスが追加されます。Visual Studio では 32 ビットアプリケーション稼働用にコンパイルしましたので、ビ ット数が一致する STAFF インスタンスを使用します。

	名前	タイブ	ステータス	64ビット	MSS有効	セキュリティ
=	STAFF	Region	Stopped		\checkmark	デフォルト
Ξ.	STAFF64	Region	Stopped	\checkmark	\checkmark	デフォルト

Х

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。

11) 設定を変更するため、名前をダブルクリック、または [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 PAC		_
Ξ.	STAFF	Region	Stopped	\checkmark	デフォルト	0	<mark>م</mark> 🖗 🗘
							編集

12)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、 [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。指定後は [適用] ボタンをクリックします。

ー般的なプロ/	ペティ С 適用 💼 削除			
開始オプション				
2前+0	8.7 <i>= 1, = 1, 1</i> , 2, 1, 1, 0	☑ ローカル コン ♀ ソールを表示	図 動的デバッグを ♀ 許可	□ システム起動時 ♀ に開始する
STAFF	C:\VS\PLI\CICSDEMO\system	□ 64ビット作業モ 9 -ド	 以前のログを削 Q 除 	

13) [リージョンの機能] では、[MSS 有効]、[JES 有効]、[PL/I 有効] のチェックがオンになっていることを確認 します。

リージョンの機能			
✓ MSS有効 ♀ □ MQ有効 ♀	☑ JES有効 Q	□ IMS有効 ♀	☑ PL/I有効 ♡

14) [追加設定] の [構成情報] 欄に環境変数を設定します。文字エンコーディングを指定する MFACCCGI_CHARSET 環境変数に IBM-932 を認識させるための値である Shift_JIS と、プロジェ クトのパスを指定する環境変数を設定し、最後に [適用] ボタンをクリックします。

変更前)

[ES-Environment]	
CICSDEMO=C:¥Users¥Public¥Documents¥Micro Focus¥Enterprise Developer¥Samples¥"PLI-VS or	
PLI-Eclipse"¥CICSDEMO	
ES_SSTM_CICS="\$CICSDEMO¥sstmcics.jcl"	
#MF_CHARSET=E	
	_

変更後)

[ES-Environment]
CICSDEMO= C:¥VS¥PLI¥CICSDEMO
ES SSTM CICS="\$CICSDEMO¥sstmcics.icl"
#MF CHARSET=E
MFACCCGI CHARSET=Shift JIS

追加設定

構成情報 💡

[ES-Environment] CICSDEMO-C:\VS\PLI\CICSDEMO ES_SSTM_CICS="\$CICSDEMO\sstmcics.jcl" #MF_CHARSET=E MFACCCGI_CHARSET=Shift_JIS

ES_SSTM_CICS 環境変数:
CICS 環境で JCL を使用する際に SSTM CICS 環境の初期化に使用される JCL の所在地を指定しま
す。

15) 画面上部の [CICS] プ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確認します。構成情報に指定した環境変数を使用して値が入力されています。

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。例題では STAFF 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。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。.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。例題で用意されているファ 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。

 ・ 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

16) 画面上部の [JES] プ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確認します。構成 情報に指定した環境変数を使用して値が入力されています。

項目名	説明
JES プログラム パス	COBOL アプリケーションの実行可能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	カタログファイルを出力するパスと、その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	ジョブ実行時に生成されるスプールデータやカタログされるデータセットの デフォルトパスを指定します。
システムプロシージャライブラリ	プロシージャライブラリの名前を指定します。例題では指定しません。

JES プログラムパス 🖇	システム カタログ 🔉
\$CICSDEMO\CICS\bin\debug	SCICSDEMO\staff_base\catalog.da
データセットの省略時ロケーション 🛿	システム プロシージャ ライブラリ
<pre>\$CICSDEMO\staff_base\</pre>	/

17) [イニシエータ] には 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ることを確認します。

ニシエータ С *新規作成	JESイニシエータ
名前── クラス── 説明〜	名前*Q INIT1
名前 クラス 説明	クラス 👔
INIT1 abcdefghijkl	abcdefghijklmnopqrstuvwxyz0123456789
	説明 ♀
	* 入力必須の項目です

18)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] になっていることがあります。 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。例題インスタンスが既に [Stopped] の場合はこれを確認します。

ステータスの設定 💡		実際のアド
Stopped	~	tcp:0.0.0.0:
	ステータスの設定 🛿 Stopped	ステータスの設定 🛿 Stopped 💙

19) [TN3270] リスナーを選択すると右側ペインにポート番号が表示されます。TN3270 エミュレータからの 接続では、このポートを指定します。

通信プロセス C	* 通信サーバ		
✓ 目 通信プロセス 1 就 Web Services and J 就 Web 就 TN3270	12EE		
	プロトコル・ 🎖 tcp 🖌	ホスト名またはiP アドレス・Q	ポート*♀ 5150

20) 画面上部の [一般] プルダウンメニューから [PL/I] を選択し、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	オン、オフ指定します。ここではオンを指定します。
Codewatch ソース パス	デバッグで使用するソースファイルのパスを指定します。
Codewatch STB パス	デバッグで使用するデバッグファイルのパスを指定します。例)XXX.stb
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。

PL/	n I C 適用
	☑ PL/i有効 ♀
	□ PLITEST アタッチのプロンプト ♡
	Codewatch ソースパス Q
	<pre>\$CICSDEMO\CICS;\$CICSDEMO\CICS-Subs</pre>
	Codewatch STB パス Q
	<pre>\$CICSDEMO\CICS\bin\Debug;\$CICSDEMO\CICS-Subs\bin\Debug</pre>
	PL/I 構成ディレクトリ Q
	SCICSDEMO

21) 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。

✓ I Directory Server
✓ II ← Default

3.8 Enterprise Server インスタンスの開始と確認

- 1) Visual Studio の [サーバー エクスプローラー] 内に STAFF 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 STAFF 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] > [CICS] を選択します。これにより CICS プロジェクトから実行される処理は STAFF インスタンスで処理 されることになります。

🖻 🗣 STAFF	プロジェクトと関連付ける	•	CICS
	コンソールログを表示(L)		Subroutines

3) [サーバー エクスプローラー] 内の STAFF 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: STAF		
□ サーバーの認証情報を要求		
ユーザー名:		
パスワード:		
グループ:		場合は空白
☑ 認証の保存		
	ОК	キャンセル

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。

	名前	タイプ	ステータス	64ビット MSS有効	セキュリティ	
Ξ.	STAFF	Region	Started	\checkmark	デフォルト	🧷 🔅 📋
						編集

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、正常に開始されたこ とを確認します。

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。

/セージID	メッセージ、	✓ プロセスID ✓ ログ	グレベルン	
91	プロセスル	メッセージル	ログレベル	メッセージ
2022		CASCD1038I	1	ES Communications Server created, ES STAFF, process-id = 9476
2022	2344	CASKC6027I	1	Enterprise Server time keeper daemon initialised
2022	6040	CASKC1000I	1	ES concurrent request limit: 000000010
2022	6040	CASSI1000I	1	Server Manager initialization completed successfully
2022	7108	CASTS1007I	T. Contraction	ES TRC Service Process initialization started
2022	8164	CASTS0007I	1	ES TSC Service Process initialization started
2022	8164	CASTS5115I	1	ES TSC Log facility file: C:\VS\PLI\CICSDEMO\system\LOGTSTDQ.dat
2022	8164	CASTS5113I	1	ES TSC Log facility cold started
2022	8164	CASTS5116I	1	ES TSC Log facility file size: 4 blocks (316,236 bytes)
2022	8164	CASTS5117I	1	ES TSC Log facility is running with 4 buffers of 63,020 bytes each
2022	9476	CASCS5001I	1	Communications interface 01 initialization started
2022	7108	CASTS1002I	1	ES TRC Service Process 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。

3.9 CICS の実行

ご使用の TN3270 エミュレータを前項で確認した TN3270 リスナーポート(localhost:5150)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。

PCOMM の例)

642 7502 6	~7 B																	
マーの道沢																		
Es:	F1 F2	13	F4	Pa	Te	17	-F8	[7]	E [1]	0 71	F18	PS:	S.k	l'se				
* 1	2 3	3 4	5	U.S.	7	8	9	3	•	^	¥ BS	ins.	Ea	Ellb.	3Lk	1		-
Tab	C ¥	E	3	T []	7	U	I 0	F	6	1		Del	Exd	Pin	7	3	8	
賞数	1 5	1	7	G	н	J	K	1	-	: 1	i `				4	ā	ß	
Shift	Z	1 0		E	K	K				1	Shift				1	2	3	
OL:1	£lt.	無	\$			意换	1.2	5	•l:		Ctrl	<		>				1 × 1
R • 79:	ションの演進						1,180	行 の	現行アク	2:50	2.E							
2017/201710	0:							DS		->		基本		[da	n.]			
						•				>		Shift		(bar	sepace			-i
装装(1):										->		Ctr1		lce:	el.			-11
16度年代										-2		115		lof	t undol			- 1
VIEC-	R J	15	E.2	e e						->		a' uite		001	s.			-
1174476	1.114 - 1		2795-	17-31		177				-		Contract C		line	el			
1 2442	G**,*.2	113240	0702-	• • • • •	erro	- D 1 F				-,		CIFICALI	• W	indotes	がライセンス	118.32		-
											Street a			2000 10	pptours (0.2-			

1) 接続すると、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。

	Signon to CICS	APPLID STAFF
Type your userid and pa	ssword, then press ENTER:	
USERID Password Language	SYSAD Groupid	
New Password		
SIT の設定)		
cicsの構成 C		
システム初期化テーブル		
(SIT) Q		
STAFF		
		開始Tran ID 💡
SIT ST ST ST	TAFF C ﷺ ⊐ピー ûn⊯	CESN

2) ログイン後、画面をクリアして PCT 名である TTEA を入力して実行キーを押すと、検索画面が表示されます。 [Name] に WILSON と入力して実行キーを押下します。

Staff Dir	ectory - Searc	h Criteria
Type a name, ci Type name as la	ty, or ZIP. st name, first	name.
Name: <u>WILSON</u>		
City:		
State: <u>IL</u>		
ZIP:		
PCT の設定)		
> JCT	プログラム管理テー	-ブルエントリ - TTEA C
✓ PCT		
CESN		
TTEA	TranClass 💡	ノロクラム名 💡
🗘 x'fefffff	DFHTCL00	STAFF02

 WILSON の情報が表示されます。表示されたデータは C:¥VS¥PLI¥CICSDEMO¥staff_base¥DBA.dat ファイルから読み込まれたものです。

	Staff Directory - Staff Information Summary
Name: Pref:	WILSON, MICHAEL MIKE
Designations:	CLU, FIIC
Mailing address: City: State:	1115 BROADWAY AVE. SUITE A2 LINCOLN IL ZIP: 24061-3482
Office address: City: State:	1115 BROADWAY AVE. SUITE A2 LINCOLN IL ZIP: 24061-3482
Phone:	000-000-0000 000-000-0000

DBA.dat の内容)

■ Micro Focus データフォ	アイルツール - DBA.dat				- П	×
						\sim
ノア1ル 福楽 使宗 参	を示 構造 ツール ヘルノ			114	U 20	
				±≠- ~ 15	ANSI ~	
📄 DBA.dat	×					
← 最初	レコード数 100 🔄	最後 🔿	DBA_LAYOUT-DEFA	ULT 🕕 構造に無効な項[目が含まれています	
130000WILSON	MICHAEL	MIKE ^	フィールド	PICTURE	値	^
130001LEE JR	MICHAEL	MIKE	Ø 01 DBA LAYOUT			
130002RUSSELL	MICHAEL	MIKE	0 02 ST STAFF CODE	Char(6)	130000	
130003ATKINSON	MICHAEL	MIKE	© 02 NAME		100000	
130004HUTTON	MICHAEL	MIKE	© 03 LAST	Char(30)	WILSON	
130005YAN	MICHAEL	MIKE	© 03 FIRST	Char (15)	MTCHAFL	
130006SMITH	MICHAEL	MIKE	02 PREF NAME	Char(10)	MIKE	
130007STROUP	MICHAEL	MIKE	02 BUS ADDR			
130008RIGBY	MICHAEL	MIKE	🛇 03 LINE 1	Char(30)	1115 BROADWAY AVE.	
130009SQUIRE	MICHAEL	MIKE	Ø 03 LINE2	Char(30)	SUITE A2	
130010PITT	MICHAEL	MIKE	© 03 CITY	Char(30)	LINCOLN	
130011CLIFFORD	MICHAEL	MIKE	03 STATE	Char(2)	IL	_
1300120'SHIELDS	MICHAEL	MIKE	🛛 03 ZIP	Char(9)	240613482	_
130013CARROLL	MICHAEL	MIKE 🗸	02 MAIL_ADDR			
<		>			1	~
130000WILSON	MICHAEL	MIKE				
3333335445442222222	22222222222222222222244444442222	222224444222				
13000079C3FE000000	000000000000000000000000000000000000000	00000D9B500C				
共有			- 列:0 レコード:	N/A レコード長: 400 / 400	挿入in 表示 Ning の 参照	センフ

3.10 PL/I ソースの動的デバッグ

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] へのチェックと 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) [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。



2) 画面左下が [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。



 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。

		EXEC CICS RECEIVE MAPSET('OCMOS') MAP('S0934') RESP(RC);
⇒		IF RC ^= DFHRESP(NORMAL) THEN
		DO; SIGNAL ERROR; END; ELSE;
		REMOVE LOW VALUES FROM INPUT FIELDS ***/
		IF NAMEDI = (45) '00'x THEN NAMEDI ⊘ NAMEDI Q表示 マ WILSON -□ ELSE;
		IF CITYDI = (30) '00'x THEN CITYDI = (30) ''; ELSE;
100 %		図 問題は見つかりませんでした
ローカル	ŀ	
□-力) 検索	l↓ (Ctrl+E) ▶ → ↓ 検索の詳細度: -
□-力) 検索 名前	ll (Ctrl+E ່ງ) ・ ク ◆ ↓ 検索の詳細度: 値

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。

4) 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。



5) 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。



6) 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。



- 7) TN3270 エミュレータを切断します。
- 8) 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。

💭 準備完了

3.11 終了処理

1) Visual Studio の [サーバー エクスプローラー] 内で STAFF インスタンスを右クリックして [停止] を 選択し、開始中のインスタンスを停止します。



2) STAFF インスタンスの停止状態を確認後に、Visual Studio 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。

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。