

# Enterprise Developer チュートリアル

## リモート メインフレーム COBOL 開発 : CICS システム間通信 Eclipse 編

### 1. 目的

本チュートリアルは Linux 上に COBOL プロジェクトと CICS 用 Enterprise Server インスタンスを複数作成し、SysC を介してインスタンス間通信を行う手順の習得を目的としています。また、ローカル Windows マシンから、リモート Linux マシンへ、Eclipse のリモート メインフレーム COBOL プロジェクトを使用して実施します。

### 2. 前提

- 本チュートリアルで利用したリモートマシン OS : Red Hat Enterprise Linux Server release 8.7
- 本チュートリアルで利用したローカルマシン OS : Windows 11 Pro
- リモートマシンに Enterprise Developer 9.0 for Linux and Unix がインストールされていること
- ローカルマシンに Enterprise Developer 9.0 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
  
- メインフレーム COBOL 開発:CICS チュートリアルを終了していること
- リモート メインフレーム COBOL 開発:JCL チュートリアルを終了していること

### 3. 実施するシナリオ

Linux マシン上に CICS 機能を利用する COBOL プロジェクトを 2 つ作成し、それぞれ実行させる Enterprise Server インスタンスを作成します。1つ目を CICS1 インスタンス、2 つ目を CICS2 インスタンスと称し、CICS1 インスタンス から CICS2 インスタンスへ通信を介したルーティングを行う下記 4 つのシナリオを実施します。

- A) CICS1 インスタンスへログインして、全てのトランザクションを CICS2 インスタンスで実行させる。
- B) CICS1 インスタンスへログインして、特定トランザクションだけを CICS2 インスタンスで実行させる。
- C) CICS1 インスタンスへログインして、トランザクションから出力するデータを CICS2 インスタンスに存在するファイルへ出力する。
- D) CICS1 インスタンスへログインして、トランザクションのプログラムから CICS2 インスタンスに存在するプログラムへリンクさせる。

#### 4. チュートリアル手順の概要

1. リモートマシンの準備
2. チュートリアルの準備
3. Eclipse の起動
4. リモート メインフレーム COBOL プロジェクトの作成
5. プロジェクトプロパティの設定
6. ビルドの実行
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. シナリオ A に対するインスタンスの準備
10. シナリオ A に対するルーティングの実施
11. シナリオ B に対するインスタンスの準備
12. シナリオ B に対するルーティングの実施
13. シナリオ C に対するインスタンスの準備
14. シナリオ C に対するルーティングの実施
15. シナリオ D に対するコードと実施方法の紹介
16. Enterprise Server インスタンスの停止
17. リモートマシンの切断

## 4.1 リモートマシンの準備

リモートマシンの準備を行うために、リモートマシンヘルトユーザーでログインします。

- 1) 環境変数 LANG に SJIS ロケールを設定します。

コマンド例) export LANG=ja\_JP.sjis

```
#export LANG=ja_JP.sjis
```

- 2) COBOL を実行する環境を設定します。製品フォルダ配下の bin フォルダ内に存在する cobsetenv を実行すると、環境変数の COBDIR が COBOL 環境として設定された旨のメッセージが表示されます。

コマンド例) ./opt/mf/ED90PU1/bin/cobsetenv

```
#!/opt/mf/ED90PU1/bin/cobsetenv
COBDIR set to /opt/mf/ED90PU1
```

- 3) COBOL 作業モードを設定します。

COBOL の作業モード (32-bit または 64-bit) を指定します。cobmode コマンドまたは環境変数 COBMODE を使用して設定します。

64-bit 設定コマンド例) export COBMODE=64

```
#export COBMODE=64
```

- 4) Micro Focus™ Directory Server (MFDS) を起動します。

ESCWA からリモートマシンのホスト名:86(デフォルトポート番号)を指定しても接続できない場合は、mfds コマンドを実行して MFDS を起動します。32-bit 稼働環境用には mfds32 コマンド、64-bit 稼働環境用には mfds64 コマンドを実行することも可能です。

コマンド例) mfds &

上記 "&" を付加すると、設定済の COBOL 環境変数を基に別プロセスで mfds が起動されます。

```
#mfds &
[1] 15850
```

- 5) ローカルマシンからのアクセス方法を RSE に指定した場合は接続ポートの解放を行います。本チュートリアルでは RSE を使用しますので解放します。SSH 接続の場合はポートの解放は必要ありません。

COBOL 環境の配下に存在する startrdodaemon を実行します。

コマンド例) \$COBDIR/remotedev/startrdodaemon 5000

上記 5000 をポート番号へ指定しない場合には、デフォルトの 4075 がポート番号として指定されます。

```

$COBDIR/remotedev/startrdodaemon 5000
Starting RSE daemon...

Reading "/opt/mf/ED90PU1/remotedev/rdo.cfg" ...

Server port range loaded from "/opt/mf/ED90PU1/remotedev/rdo.cfg": 10000-10003

#Daemon running on: kthost, port: 5000
```



### 注意

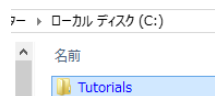
ディレクトリ配下の書き込み権限がない場合はビルド時にエラーとなります。

ls -l コマンドなどで権限を確認後、chmod コマンドなどで適切な権限設定を前もって実行してください。

## 4.2 ローカルマシンの準備

例題プログラムに関連するリソースを用意します。

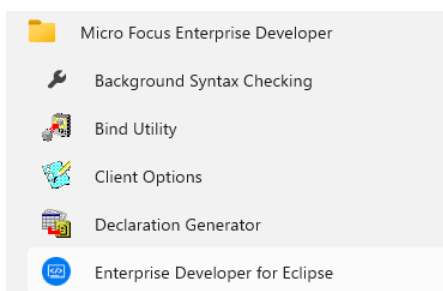
- 1) 使用する例題プログラムは、CICS チュートリアルキットに添付されている Tutorials.zip に圧縮されています。これを C:¥ 直下に解凍します。



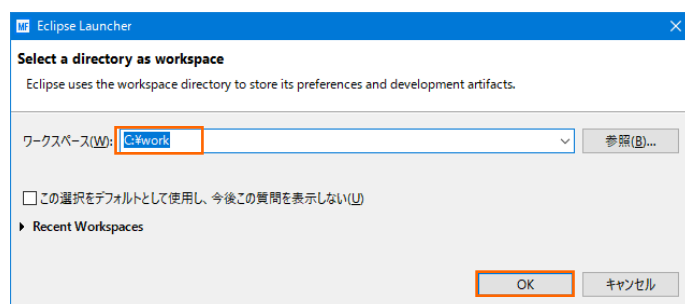
- 2) Eclipse のワークスペースで使用する work フォルダを C:¥ 直下に作成します。

## 4.3 Eclipse の起動

- 1) ローカルマシンで 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。



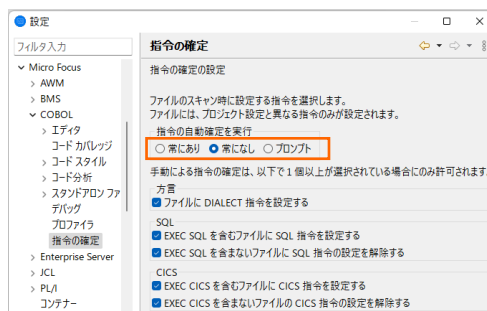
- 3) [ようこそ] タブが表示される場合は [Open COBOL Perspective] をクリックして、COBOL パースペクティブを開きます。



- 4) パースパクティブ表示後、[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。

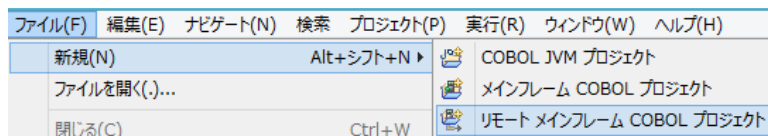


- 5) 既存ファイルのインポート時、自動的にコンパイル指令が指定される機能が用意されていますが、本チュートリアルではこれを解除します。[ウィンドウ] プルダウンメニューの [設定] > [Micro Focus] > [COBOL] > [指令の確定] > [指令の自動確定を実行] で [常になし] を選択し、[適用して閉じる] ボタンをクリックします。



#### 4.4 リモート メインフレーム COBOL プロジェクトの作成

- 1) 新しい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] > [リモート メインフレーム COBOL プロジェクト] を選択します。



- 2) プロジェクト作成ウィンドウには以下のように入力します。

項目名	説明
プロジェクト名	任意です。ここでは CICS1 を指定します。
ファイル システムを選択	リモートマシンと接続するファイル システムを指定します。ここでは [リモート ファイル システム(RSE)] を選択します。

##### リモートメインフレーム COBOL プロジェクト

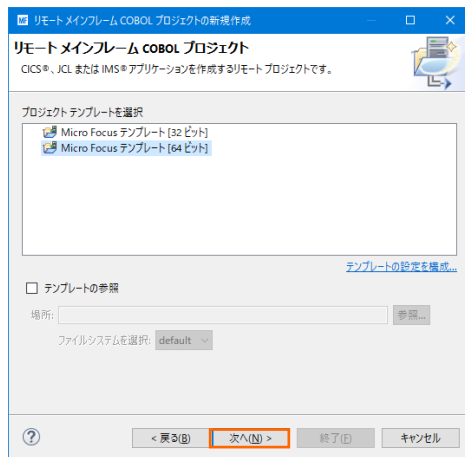
CICS®、JCL または IMS®アプリケーションを作成するリモート

プロジェクト名:

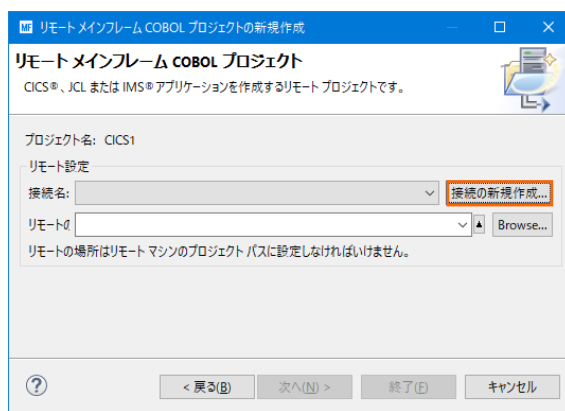
ファイル システム

ファイル システムを選択:

- 3) テンプレート指定ウィンドウでは [Micro Focus テンプレート 64 ビット] を選択して [次へ] ボタンをクリックします。



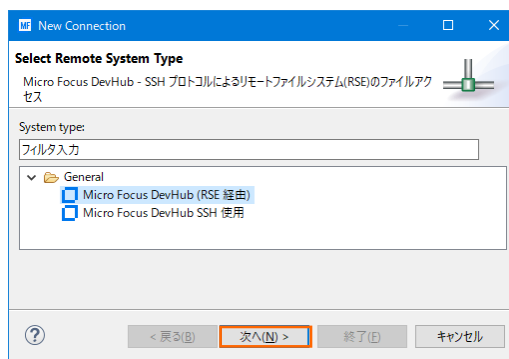
4) 新しい接続を作成するため、[接続の新規作成] ボタンをクリックします。



5) 接続タイプでは2種類から選択可能です。

5-1) [RSE 経由] の場合

① [RSE 経由] を選択して [次へ] ボタンをクリックします。本チュートリアルでは RSE を使用しますので、こちらの手順で作成します。



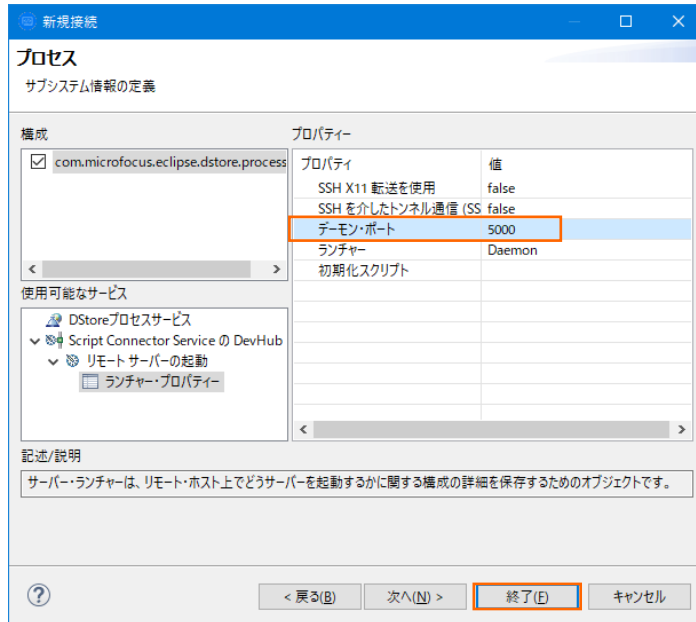
② [ホスト名] ヘルモートマシン名または IP アドレスを指定して [次へ] ボタンをクリックします。  
[接続名] は任意に変更可能です。

ホスト名 :	<input type="text" value="XXXXXXXXXX"/>
接続名 :	<input type="text" value="RHEL8"/>
記述/説明 :	<input type="text"/>

- ③ 下記画面の [使用可能なサービス] > [Script Connector Service の DevHub] > [リモート サーバーの起動] > [ランチャー・プロパティ] > [デーモン・ポート] 項目値を、前項で指定したリモートマシンのポート 5000 へ変更後、[終了] ボタンをクリックします。デフォルトポート番号(4075)を解放した場合は前画面で [終了] ボタンをクリックして構いません。

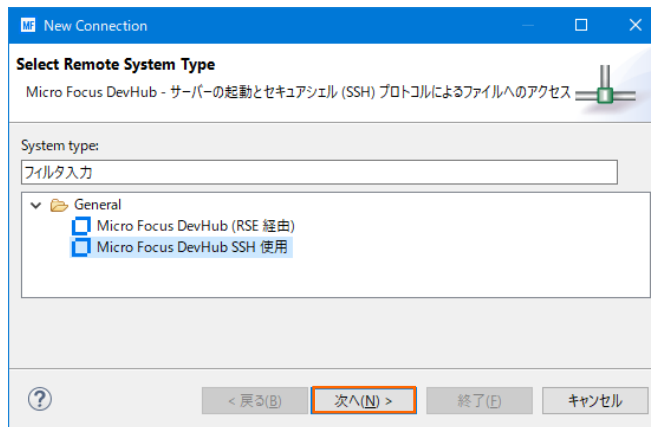
デフォルト値)4075

変更値)5000



## 5-2) [SSH 使用] の場合

- ① [SSH 使用] を選択して [次へ] ボタンをクリックします。



- ② [ホスト名] へリモートマシンまたは IP アドレスを指定して [次へ] ボタンをクリックします。  
[接続名] は任意に変更可能です。

Host name: [REDACTED]  
 Connection name: RHEL8SSH  
 Description/Note: [REDACTED]

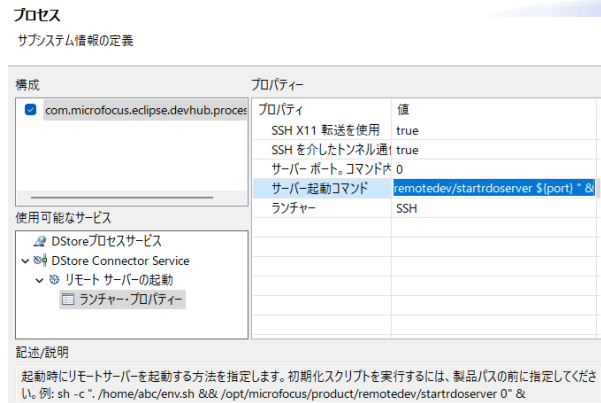
- ③ 下記画面の [使用可能なサービス] > [DStore Connector Service] > [リモート サーバーの起動] > [ランチャー・プロパティ] > [サーバー起動コマンド] 項目値を製品のインストールパスへ修正後、[終了] ボタンをクリックします。

変更前の値)

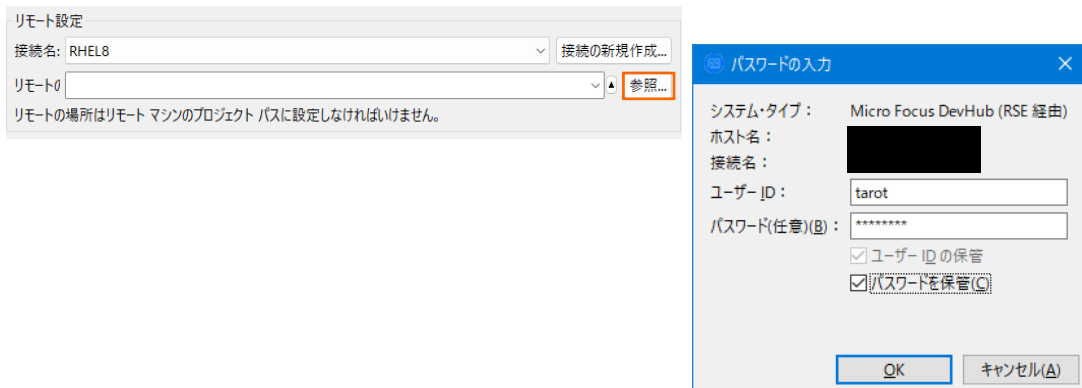
```
sh -c "/opt/microfocus/EnterpriseDeveloper/remotedev/startrdoserver ${port} " &
```

変更後の値例)

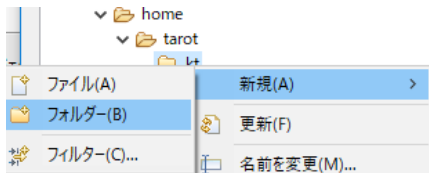
```
sh -c "/opt/mf/ED90PU1/remotedev/startrdoserver ${port} " &
```



- 6) リモートマシンにプロジェクトを作成するロケーションを指定するため、[参照] ボタンをクリックします。リモートマシンへのログオンウィンドウが表示された場合には、権限を持つユーザー ID とパスワードを指定してアクセスしてください。



- 7) リモートマシンのブラウザウィンドウが表示されますので、配置したいパスへプロジェクト用のディレクトリを作成します。作成可能なロケーションを右クリックして [新規] > [フォルダー] を選択します。

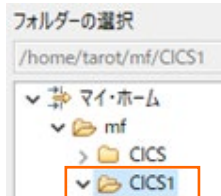




- 8) 新しいディレクトリ名は任意ですが、ここではプロジェクト名と同様の CICS1 を指定して [終了] ボタンをクリックします。

接続名(A) :	RHEL8
親フォルダー(C) :	/home/tarot/mf
新規フォルダー名(D) :	CICS1

- 9) プロジェクトディレクトリへ CICS1 を選択して、[OK] ボタンをクリックします。



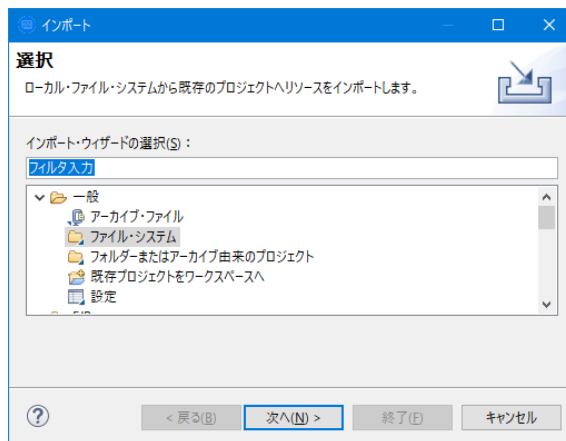
- 10) 指定項目を確認後、[終了] ボタンをクリックします。

プロジェクト名:	CICS1	
リモート設定		
接続名:	RHEL8	接続の新規作成...
リモート0	/home/tarot/mf/CICS1	参照...
リモートの場所はリモート マシンのプロジェクト パスに設定しなければいけません。		

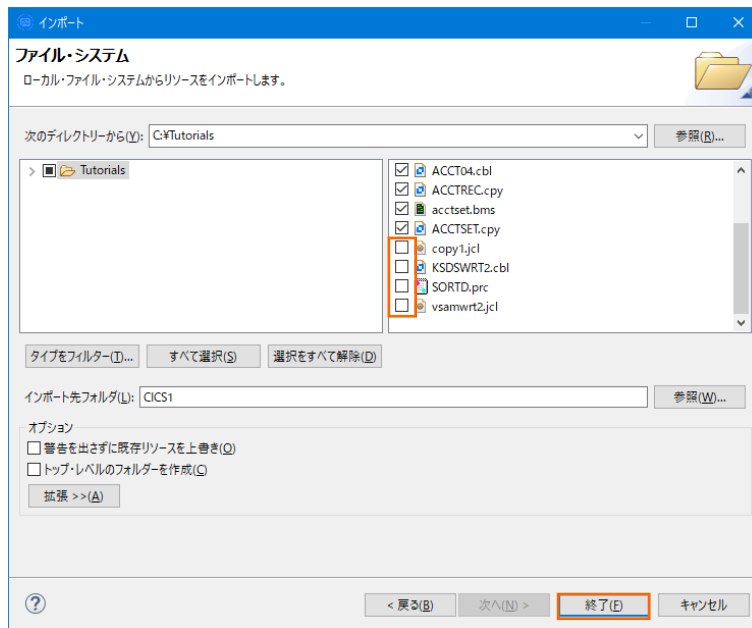
- 11) 同様の操作で、CICS2 プロジェクトも作成してください。COBOL エクスプローラーに 2 つのリモートプロジェクトが表示されます。

> 📁 CICS1 [RHEL8:/home/tarot/mf/CICS1]  
> 📁 CICS2 [RHEL8:/home/tarot/mf/CICS2]

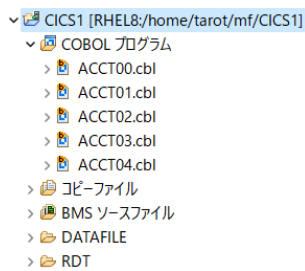
- 12) 用意した例題プログラム類をインポートします。CICS1 プロジェクトを右クリックして [インポート] > [インポート] を選択し、インポートウィンドウにて [一般] > [ファイル・システム] を選択後 [次へ] ボタンをクリックします。



- 13) 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、左側の [Tutorials] へチェックし、不要な下から 4 ファイルのチェックをはずします。ファイル名先頭に [ACCT] が付く上部 8 ファイルが残りますので [終了] ボタンをクリックします。この実行により、リモートマシンのプロジェクトディレクトリへ例題プログラムが配置されます。



- 14) COBOL エクスプローラー内に表示されている CICS1 プロジェクトにインポートしたファイルやフォルダが表示されていることを確認します。



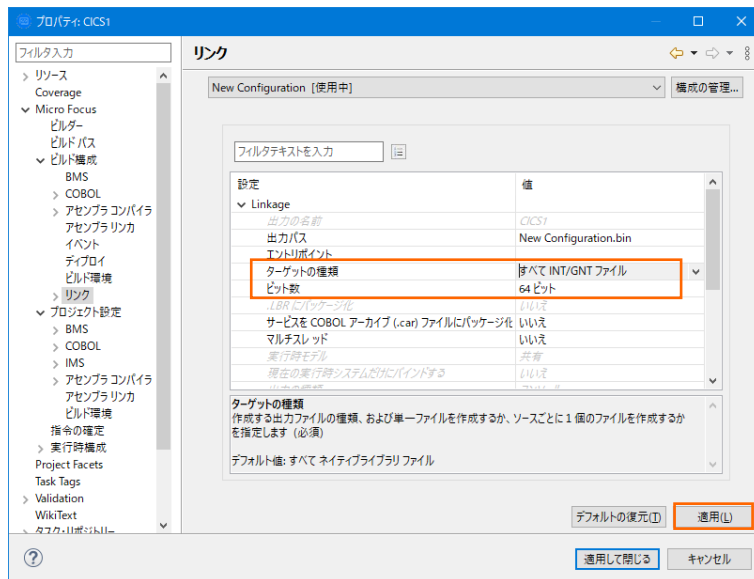
- 15) 同じ手順で CICS2 プロジェクトへ例題プログラム類をインポートします。

#### 4.5 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 命令を含むプログラム、コピー文が含まれています。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。

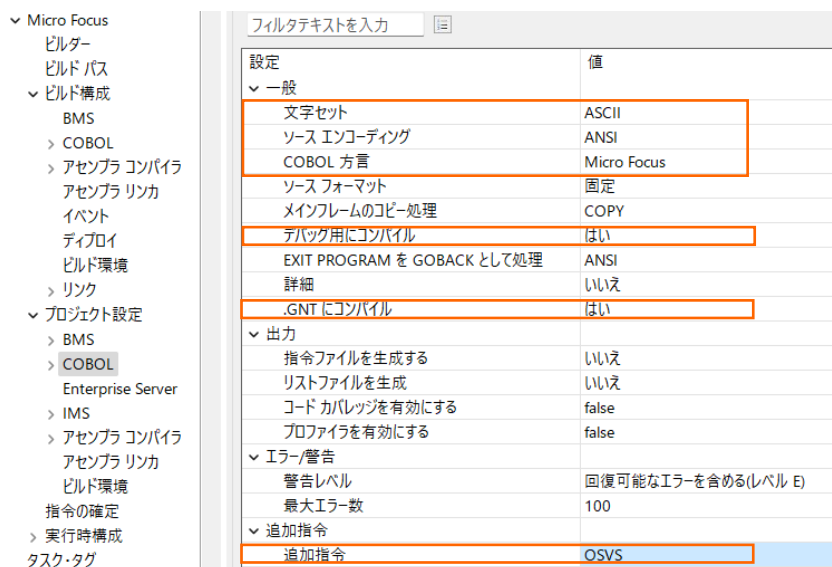
- 1) COBOL エクスプローラー内の CICS1 , CICS2 各プロジェクトを右クリックして [プロパティ] を選択します。
- 2) 左側ツリービューの [Micro Focus] > [ビルド構成] > [リンク] を選択して、下記項目を指定します。指定後は [適用] ボタンをクリックしてください。

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。
ビット数	稼働ビット数を指定します。ここでは [64 ビット] を指定します。

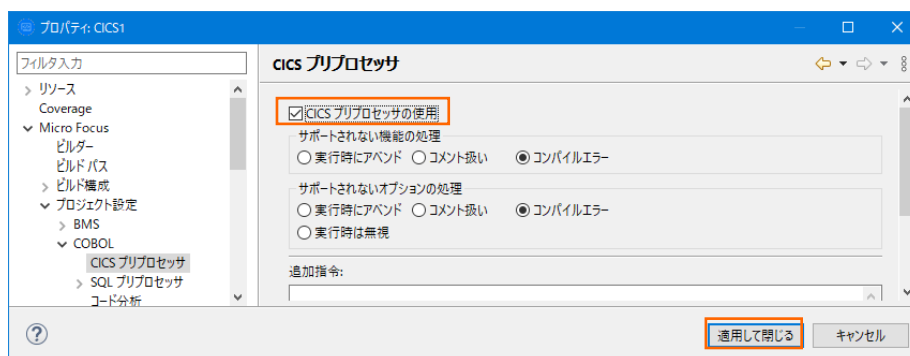


- 3) 左側ツリービューの [Micro Focus] > [プロジェクト設定] > [COBOL] を選択して、下記項目を指定します。指定後は [適用] ボタンをクリックしてください。

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。
ソースエンコーディング	サンプルソースは SJIS を使用しているため、ANSI を指定します。
言語の方言	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 OSVS を入力します。

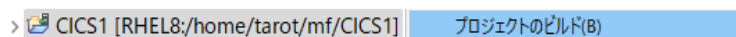


- 4) 左側ツリービューの [Micro Focus] > [プロジェクト設定] > [COBOL] > [CICS プリプロセッサ] を選択して、[CICS プリプロセッサの使用] チェックボックスをオンにして [適用して閉じる] ボタンをクリックするとプロパティウィンドウが閉じます。

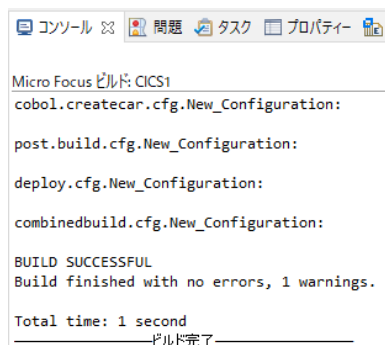


## 4.6 ビルドの実行

- 1) COBOL エクスプローラー内の CICS1 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。CICS2 プロジェクトも同様にビルドします。



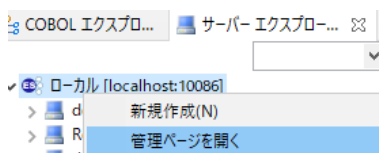
- 2) コンソールタブで成功を確認します。BMS ファイルもリモートマシン上でビルドされ、MOD ファイルが出力されます。



- 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 フォルダ配下に実行ファイル(.gnt ファイル)が作成されていることを確認してください。

## 4.7 Enterprise Server インスタンスの設定

- 1) CICS 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。Eclipse の [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 [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 ESCWA が表示されます。画面上部の [ネイティブ] をクリックします。



- 3) Eclipse で作成した REHL8 が表示されていない場合は、ターゲットマシンの MFDS に接続するために右側ペインの [追加] をクリックします。既に表示されている場合は追加する必要はありません。



- 4) [名前] には任意の接続名称を、[ホスト] にはターゲットマシンの IP アドレスを、[ポート] には MFDS のデフォルトポートである 86 を指定して [保存] をクリックします。

Directory Server

名前 \*

RHEL8

ホスト \*

[REDACTED]

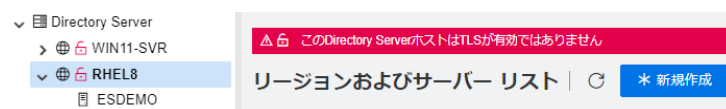
ポート \*

86

説明

[REDACTED]

- 5) 左側ペインに作成した [RHEL8] が表示されますので、これをクリックし、右側ペインの画面中央にある [新規作成] ボタンをクリックして、ターゲットマシンに Enterprise Server インスタンスを作成します。



- 6) 表示された画面の [名前] に CICS1 を入力します。64 ビットの実行可能ファイルを生成したため [64 ビット作業モード] にチェックし、[TN3270 リスナーの作成] がチェックされていることを確認後、ポート番号に 9004 を入力します。

最後に [保存] ボタンをクリックします。

リージョンの新規作成

名前 \*

CICS1

説明

[REDACTED]

▲ 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

保存 戻る

**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。

- 7) 作成した CICS1 インスタンスが一覧に表示されます。CICS1 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。

名前	タイプ	ステータス	64ビット	MSS有効	セキュリティ	PAC	
CICS1	Region	Stopped	✓	✓	デフォルト		[編集]

- 8) [リージョンの機能] の [MSS 有効] がチェックされていることを確認します。

リージョンの機能

MSS有効  JES有効  IMS有効  PL/I有効

MQ有効

- 9) 表示画面の下にある [動的デバッグを許可] にチェックします。この指定により、Eclipse からの動的デバッグが可能になります。

ローカルコンソールを表示  動的デバッグを許可  システム起動時に開始する

64ビット作業モード  以前のログを削除

- 10) [追加設定] の [構成情報] 欄へ、プロジェクトディレクトリパスと日本語半角カナを有効にするため下記内容を設定し、最後に [適用] ボタンをクリックします。

入力値)

[ES-Environment]

MFCODESET=9122

追加設定

構成情報

[ES-Environment]  
MFCODESET=9122



重要

入力値は全て半角英数字で指定してください。

- 11) 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。



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。

**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

12) 同様の手順で CICS2 インスタンスを作成します。CICS1 インスタンスと異なる TN3270 ポート番号を指定してください。

13) 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。

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

14) 各リスナーの [tcp] にはリモートマシンの IP アドレスを入力して [適用] ボタンをクリックします。

**リスナー プロパティ** |

名前 \*

**▲ 此のエンドポイントはネットワーク経由でアクセス可能になり、TLSが無**

プロトコル  ホスト名またはIP アドレス \*

15) 2つのインスタンス作成が終了したら、画面左側ペインの [RHEL8] をクリックして一覧画面に戻ります。

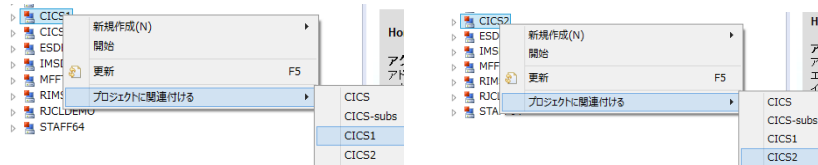
- ▼  Directory Server
  - >  WIN11-SVR
  - ▼  RHEL8

## 4.8 Enterprise Server インスタンスの開始と確認

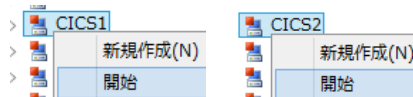
1) Eclipse に戻り、サーバーエクスプローラー内のリモート環境に CICS1 と CICS2 インスタンスが表示されていることを確認します。表示されていない場合はリモート環境を右クリックし、[更新] を選択してリフレッシュしてください。

- 2) サーバークリエータ内のリモート環境に存在する CICS1 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] > [CICS1] を選択します。これにより CICS1 プロジェクトから実行されるアプリケーションは CICS1 インスタンスで処理されることになります。

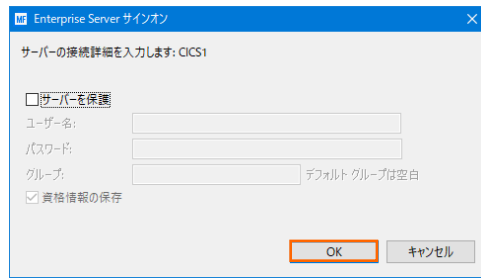
CICS2 インスタンスも同様に CICS2 プロジェクトへ関連付けます。



- 3) CICS1 インスタンスと CICS2 インスタンスをそれぞれ右クリックして [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。



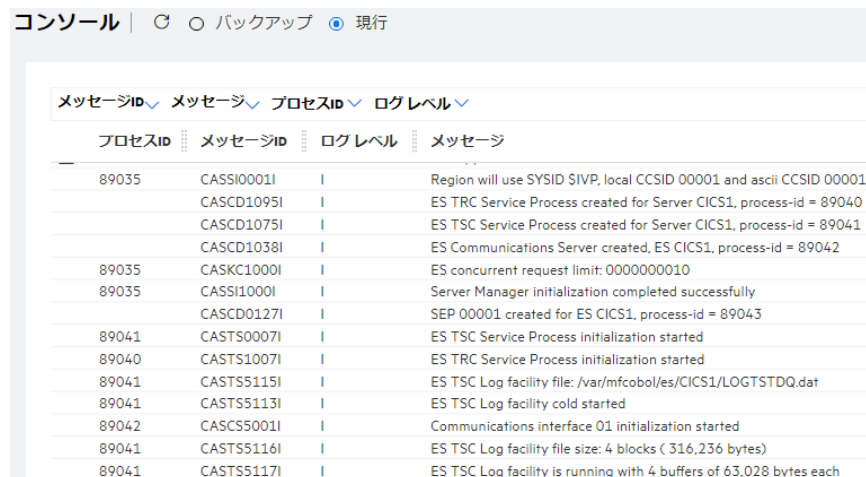
- 5) ESCWA へ移動して開始状態であることを確認後、[編集] アイコンをクリックします。

名前	タイプ	ステータス	64ビット	MSS有効	セキュリティ	PAC
CICS1	Region	Started	✓	✓	デフォルト	
CICS2	Region	Started	✓	✓	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。



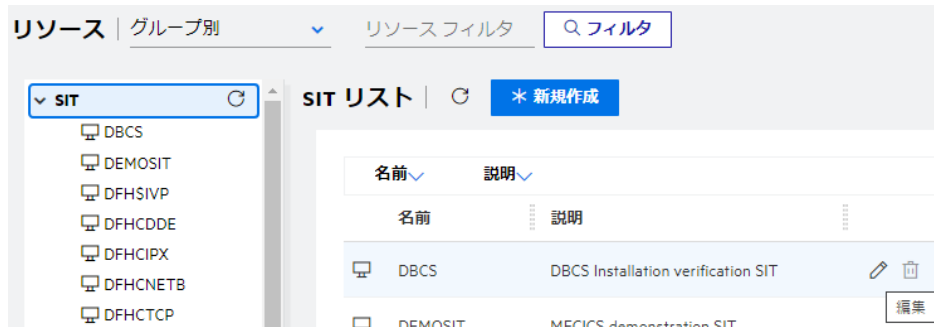


## 注意

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) CICS2 インスタンスの SysID をユニークな値にします。CICS2 インスタンスを [編集] アイコンで開き、画面上部の [モニター] プルダウンメニューから [CICS] > [リソース] > [グループ別] を選択し、左側ペインの [SIT] をクリックします。

SIT 一覧が表示されますので、CICS2 インスタンスに指定した [DBCS] の [編集] アイコンをクリックします。



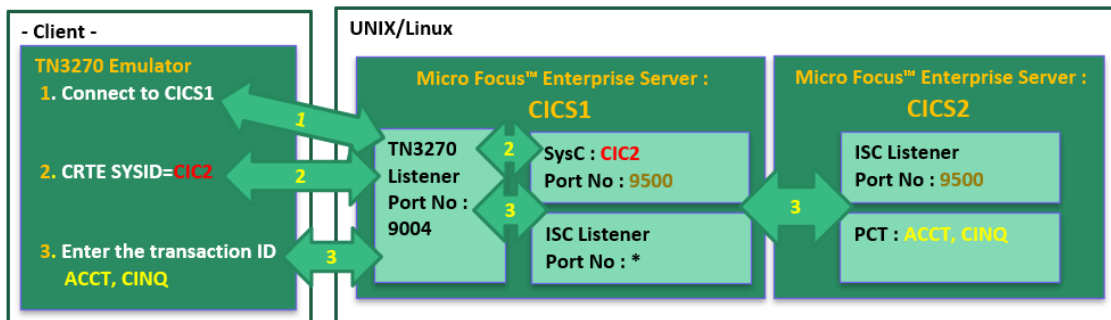
- 8) SIT 設定項目の [SysID] を \$IVP から CIC2 に変更して [適用] ボタンをクリックします。

SysID   
CIC2

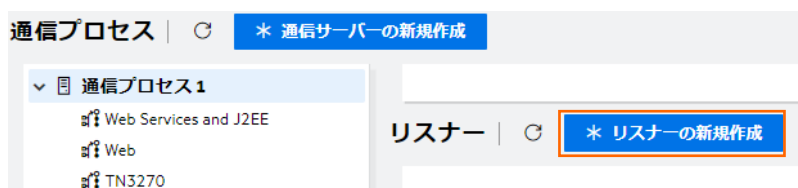
## 4.9 シナリオ A に対するインスタンスの準備

シナリオ)CICS1 インスタンスへログインして、全てのトランザクションを CICS2 インスタンスで実行させる。

【概要図】



- 1) CICS1 インスタンスへ CICS 間通信に必要な ISC リスナー定義を作成します。ESCWA から CICS1 インスタンスの編集画面を開き、[一般] プルダウンメニューから [リスナー] を選択し、[リスナーの新規作成] ボタンをクリックします。



- 2) 下記内容を入力後、[保存] ボタンをクリックすると、自動的にリスナーが開始されます。

項目名	説明
名前	任意です。ここでは ISC1 を指定しますが、他インスタンスと重ならないように指定してください。
IP アドレス	IP アドレスもしくは * を指定します。
ポート	使用していないポート番号もしくは * を指定します。
会話タイプ	[MSS システム間通信] を選択します。 自動的に [レガシーMicro Focus アプリケーション形式] ヘチェックが入ります。

**注意**

該当 SIT のスタートアップリストに DFHISC が必要です。本チュートリアルでは設定済です。

- 1 DFHBMS
- 2 DFHCONS
- 3 DFHEDF
- 4 DFHHARDC
- 5 DFHISC
- 6 DFHOPER
- 7 DFHSIGN
- 8 DFHSPI
- 9 DFHTYPE
- 10 DFHVTAM
- 11 DFHSACCT
- 12 DFHTERM
- 13 DFHSIVP
- 14 DBCS

- 3) CICS1 インスタンスに CICS 間通信に必要な SysC 定義を作成します。[CICS] プルダウンメニューから [CICS] > [リソース] > [タイプ別] を選択し、左側ペインで [SysC] をクリック後、[新規作成] ボタンをクリックします。

- 4) 下記項目を入力し [保存] ボタンをクリックします。

項目名	説明
名前	4文字で指定します。任意ですが通信相手先が認識可能な名前を指定します。ここでは CIC2 を指定します。この名前がリモートランザクションの SYSID となります(下記画面参照)。
グループ	CICS1 インスタンスで指定している SIT を指定します。ここでは DBCS を選択します。
説明	説明を任意で入力します。
MF ノード	通信相手インスタンスが存在するアドレスを指定します。この場合は CICS2 インスタンスが存在する Linux/UNIX マシンの IP アドレスです。(例:11.22.33.44)
MF ポート	CICS 間通信に使用するポート番号を指定します。他で使用していない 4 桁の数値を指定してください。ここでは 9500 を指定します。
ネット名	通信相手インスタンス名を指定します。ここでは CICS2 です。
セッション最大	1 以上の数値を設定します。ここでは 4 を指定します。

cci システムの接続 リソースの作成 | **保存**

名前  グループ

説明

---

接続パラメーター

プロトコル  ネット名  セッション最大

MF ノード  MF ポート

- 5) CICS2 インスタンスへ ISC リスナー定義を作成します。CICS1 インスタンスと同様の手順で、CICS1 インスタンスの SysC 定義で指定した [MF ポート] 番号を指定します。リスナー追加画面で下記項目を入力後、[保存] ボタンをクリックします。

項目名	説明
名前	ここでは ISC2 を指定します。
IP アドレス、ポート	CICS1 インスタンスが存在するアドレスと CICS1 インスタンスの SysC 定義で指定した [MF ポート] 番号を指定します。同じアドレス内であれば “*:9500” が指定可能です。(例:11.22.33.44:9500)
会話タイプ	[MSS システム間通信] を選択します。 自動的に [レガシーMicro Focus アプリケーション形式] ヘチェックが入ります。

リスナープロパティ | **保存**

名前

レガシー Micro Focus アプリケーション形式

**▲ このエンドポイントはネットワーク経由でアクセス可能になり、TLSが無効になります。**

プロトコル  ホスト名または IP アドレス  ポート

- 6) CICS1 と CICS2 インスタンスを再起動します。

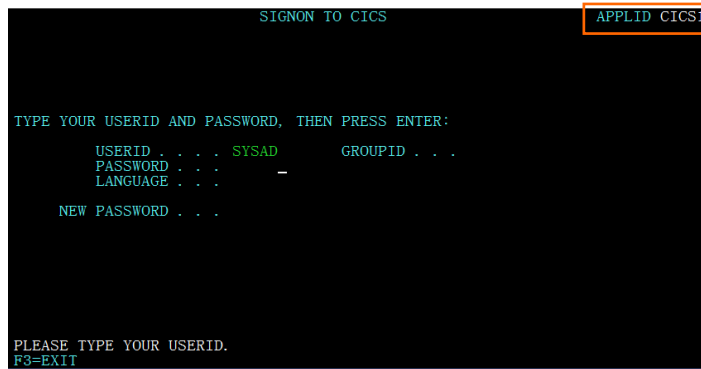
CICS1 インスタンスのコンソールログで SysC 接続を確認します。下記内容が出力されていれば認識されています。

CASCS1122I	I	Attempting to establish ISC connection CIC2 to system CICS2
CASCS1109I	I	Connection CICS2 (sysid CIC2) is activated contention winner

#### 4.10 シナリオ A に対するルーティングの実施

シナリオ)CICS1 インスタンスへログインして、全てのトランザクションを CICS2 インスタンスで実行させる。

- 1) TN3270 エミュレータを CICS1 インスタンスの TN3270 リスナーポート(9004)へ接続します。

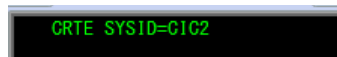


- 2) [USERID] と [PASSWORD] に SYSAD を入力してログインします。

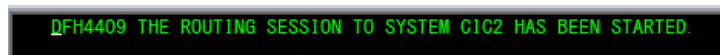
- 3) 画面クリア後に下記コマンドを実行します。

補足)TN3270 エミュレータで、使用しているキーボード設定のクリア設定をご確認ください。

CRTE SYSID=CIC2



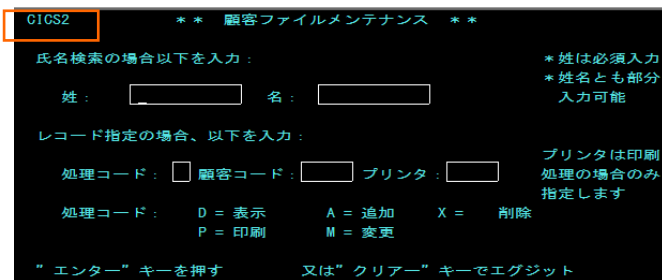
- 4) ルーティングが開始されました。



- 5) 画面クリア後に下記トランザクションを実行します。確認するために画面左上にインスタンス名を表示しています。同じように出力するためには BMS ファイルに項目の追加が必要になります。

トランザクション)ACCT

CICS2 インスタンスで指定トランザクションが起動されています。



- 6) 確認のため、クリアを2回実施後に下記コマンドを実行します。

CINQ

CICS2 インスタンスへルーティングされていることが確認できます。



PCT を実行しますので ACCT を入力してください。

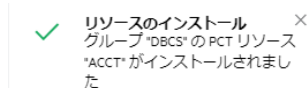
プログラム管理テーブル リソースの作成 | 保存 | 戻る ?

名前	グループ	
ACCT	DBCS	
説明		
CICS primer transaction		
TranClass		
DFHTCL00		
プログラム名	リモート SysID	リモート PCT
ACCT00	CIC2	ACCT

- 3) 追加を反映させるために [インストール] ボタンをクリックします。

プログラム管理テーブル - ACCT (DBCS) | 適用 | **インストール**

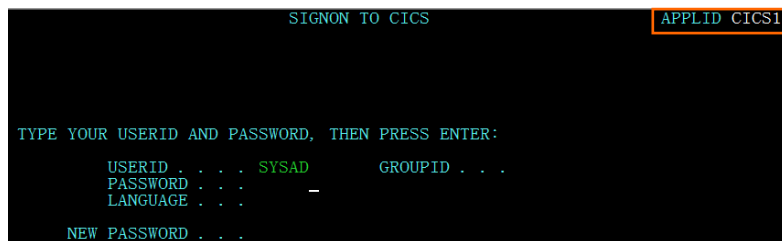
- 4) 画面右上にインストール済のメッセージが表示されます。



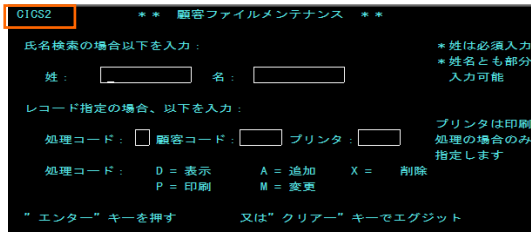
#### 4.12 シナリオ B に対するルーティングの実施

シナリオ)CICS1 インスタンスへログインして、特定トランザクションだけを CICS2 インスタンスで実行させる。

- 1) リモートとして作成した PCT 定義の ACCT だけを CICS2 インスタンスへルーティングします。  
TN3270 エミュレータを CICS1 インスタンスへ接続します。

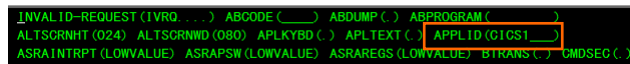


- 2) [USERID] と [PASSWORD] に SYSAD を入力してログインします。
- 3) 画面クリア後に下記トランザクションを実行します。この例では画面左上にインスタンス名を表示しています。  
トランザクション)ACCT  
CICS2 インスタンスで指定トランザクションが起動されています。



- 4) クリアを2回実施後に下記コマンドを実行すると、CICS1 インスタンスで稼働していることが確認できますので、PCT で指定したトランザクションのみが CICS2 インスタンスで実行されたこととなります。

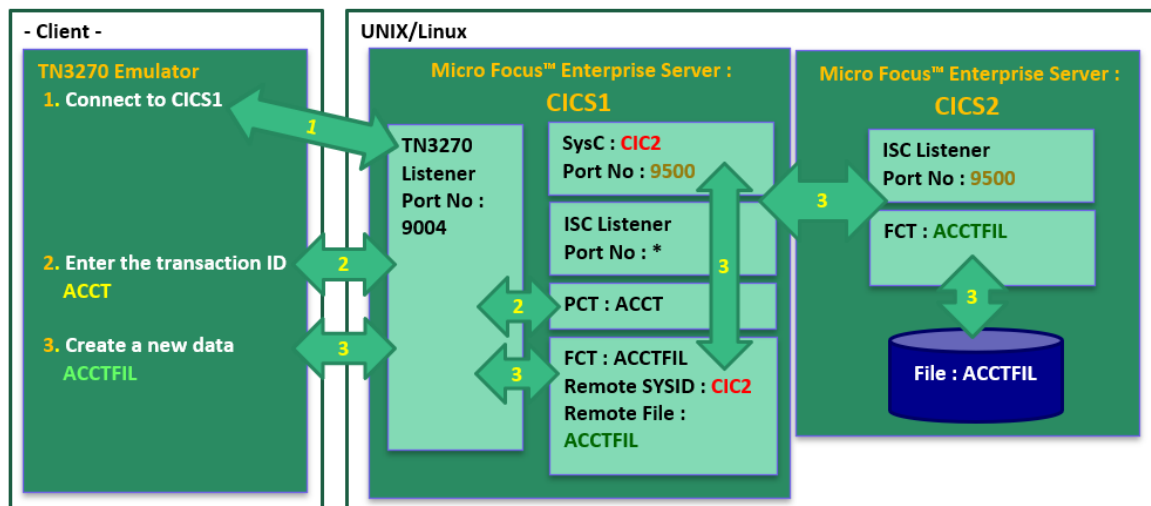
CINQ



#### 4.13 シナリオ C に対するインスタンスの準備

シナリオ)CICS1 インスタンスへログインして、トランザクションから出力するデータを CICS2 インスタンスに存在するファイルへ出力する。

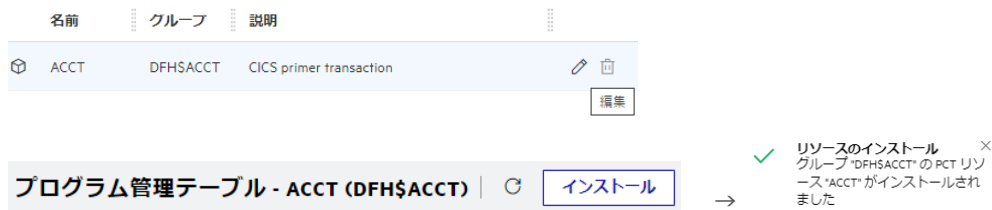
【 概要図 】



- 1) CICS1 インスタンスへ前項でリモート PCT として作成した ACCT を削除するため、[DBCS] グループの [ACCT] ヘカーソルを合わせ、[削除] アイコンをクリックします。

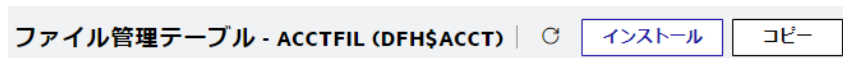


- 2) 既存の ACCT を有効にするため、[DFH\$ACCT] グループにある ACCT の [編集] アイコンをクリックし、[インストール] ボタンをクリックします。



- 3) CICS1 インスタンスへ FCT 定義を作成します。CICS1 インスタンスを [編集] アイコンで開き、画面上部の [モニター] プルダウンメニューから [CICS] > [リソース] > [タイプ別] を選択し、左側ペインの [FCT] をクリックします。

FCT 一覧が表示されますので、[DFH\$ACCT] グループに属する [ACCTFIL] の [編集] アイコンをクリックし、[コピー] ボタンをクリックします。



- 4) 下記を入力して [保存] ボタンをクリックします。

項目名	説明
名前	ACCTFIL を入力します。
グループ	SIT で指定している DBCS を選択します。



- 5) 一旦保存した [DBCS] グループの [ACCTFIL] に下記を入力して、[適用] ボタンをクリックします。

項目名	説明
リモート SYSID	SysC 定義の名前である CIC2 を入力します。
リモートファイル	CICS2 インスタンスで実行する FCT 定義名を入力します。ここでは既存の同名 FCT を実行しますので ACCTFIL を入力してください。
LRECL 最小 LRECL 最大 Key 位置 Key 長	0 を入力します。空白など、他の値を入力しないでください。





- 6) [適用] ボタンクリック後、下記の項目に数値が再表示されますので、これを確認してください。

LRECL 最小 63      LRECL 最大 383

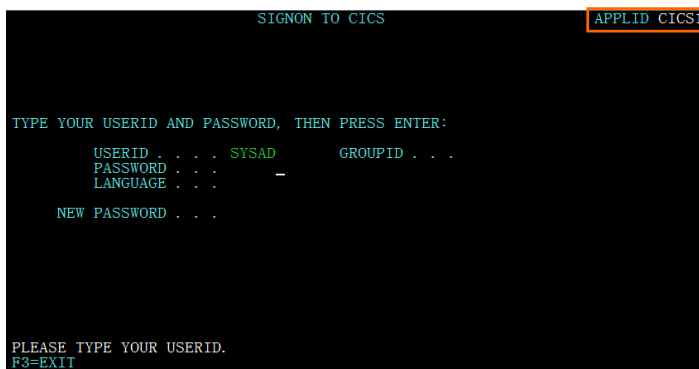
キー位置 1      キー長 5

- 7) 追加した FCT を反映させるため、CICS1、CICS2 インスタンスを再起動します。

#### 4.14 シナリオ C に対するルーティングの実施

シナリオ)CICS1 インスタンスへログインして、トランザクションから出力するデータを CICS2 インスタンスに存在するファイルへ出力する。

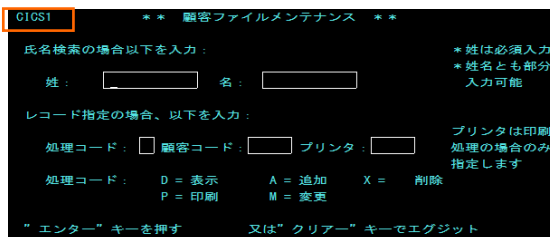
- 1) TN3270 エミュレータを CICS1 インスタンスへ接続します。



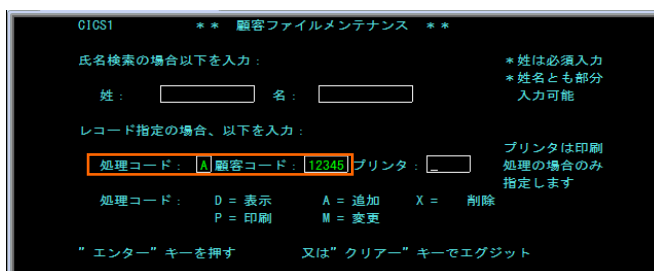
- 2) [USERID] と [PASSWORD] に SYSAD を入力してログインします。  
 3) 画面クリア後に下記トランザクションを実行します。この例では画面左上にはインスタンス名を表示しています。

ACCT

CICS1 インスタンスで指定トランザクションが起動されています。

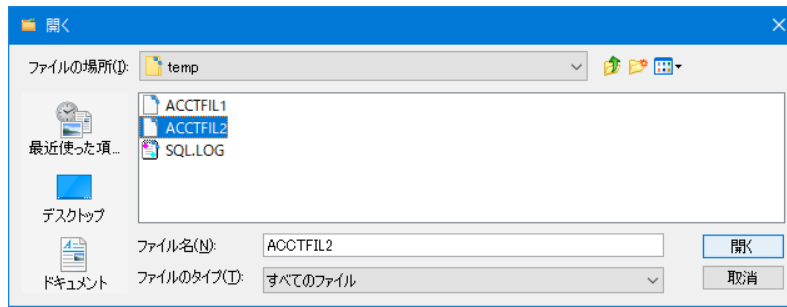


- 4) データを追加するためタブキーで項目間を移動させ、処理コードに A、顧客コードに 12345 を入力して実行キーを押下してください。

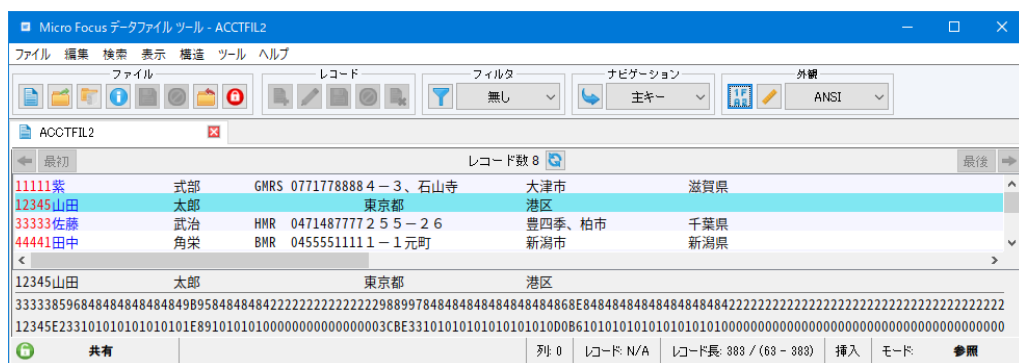




- 10) [ファイル] プルダウンメニュー > [開く] > [データファイル] を選択してウィンドウを開きます。[ファイル名] の [参照] ボタンをクリックし、[ファイルのタイプ] に [すべてのファイル] を指定して CICS2 インスタンスに存在した ACCTFIL2 ファイルを共有モードで開きます。

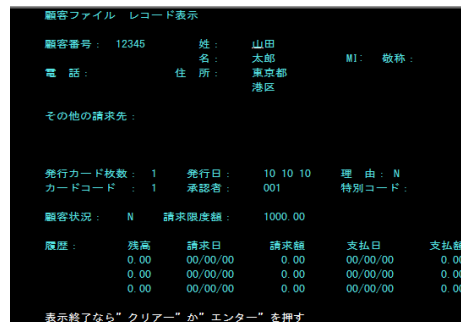


- 11) 前項で作成したデータが CICS2 の FCT を通じて CICS2 配下のファイルに追加されています。

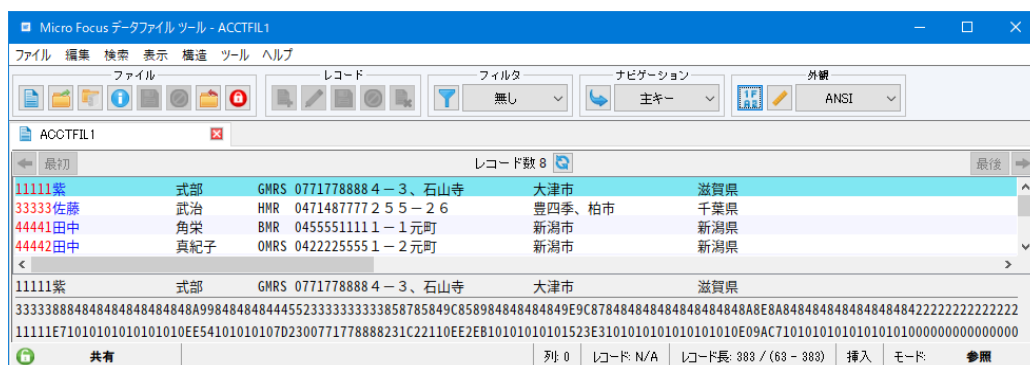


補足)CICS2 インスタンスへログインしてデータを確認することもできます。

処理コード:D、顧客番号:12345



- 12) CICS1 インスタンスに存在するファイル内容も同様に確認します。追加データは存在していません。

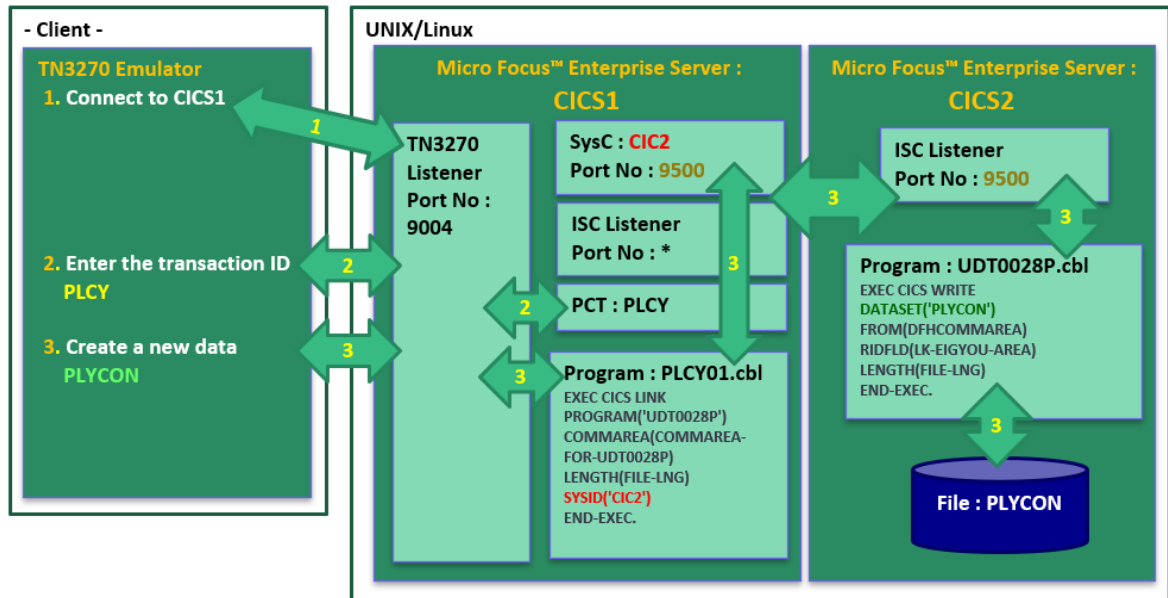


#### 4.15 シナリオ D に対するコードと実施方法の紹介

シナリオ)CICS1へログインして、トランザクションのプログラムから CICS2 に存在するプログラムへリンクさせる。

CICS1 インスタンスに存在するプログラム PLCY01.cbl から SysC に定義してある CICS2 を利用して CICS2 インスタンスに存在するプログラム UDT0028P.cbl を呼び出し、CICS2 に存在するファイルヘータを出力します。

【概要図】



- 1) TN3270 エミュレータを CICS1 インスタンス(9004)へ接続し、PLCY トランザクションを実行します。



- 2) CICS1 インスタンスで起動された画面の各項目へ入力後、実行キーを押下します。



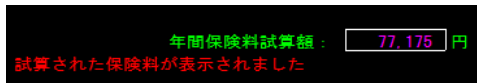
- 3) プログラムコードから、CICS2 インスタンスに存在するプログラム UDT0028P.cbl が呼び出されています。

```

029800* 顧客情報ファイル書き出し機能の呼び出し
029900*
030000 EXEC CICS LINK PROGRAM('UDT0028P')
030100 COMMAREA(COMMAREA-FOR-UDT0028P)
030200 LENGTH(FILE-LNG) SYSD('CIC2')
030300 END-EXEC.

```

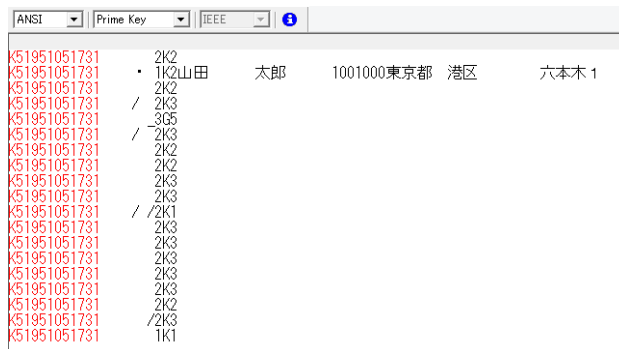
- 4) CICS2 インスタンスに存在するプログラム UDT0028P.cbl から値が返却され、同時にファイルに入力内容が出力されます。



- 5) 入力データが出力された CICS2 インスタンスに存在する PLYCON ファイル内容をデータツールから確認します。最後に追加したデータが存在しています。

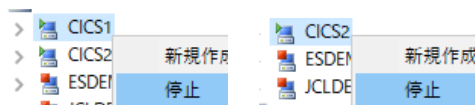


- 6) CICS1 インスタンスに存在する PLYCON ファイル内容をデータツールから確認します。追加データは存在していないため、CICS2 に存在するプログラムへのリンクが成功していることが確認できます。



## 4.16 Enterprise Server インスタンスの停止

- 1) CICS1 と CICS2 インスタンスを停止します。



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

名前	タイプ	ステータス	64ビット	MSS有効
CICS1	Region	Stopped	✓	✓
CICS2	Region	Stopped	✓	✓

#### 4.17 リモートマシンの切断

必要であれば、リモートマシンで使用したポートの遮断をルートユーザーで行います。

ポート遮断コマンド) \$COBDIR/remotedev/stoprpdodaemon 5000

### 5. 免責事項

本チュートリアル of 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 of 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。