Micro Focus メインフレームソリューション

スターターズキット

7. Enterprise Developer データベース連携

7.1 チュートリアルの準備

本チュートリアルを実行する前に「5. Enterprise Developer JCL チュートリアル」を終了している必要 があります。ここで使用する例題プログラムは、キットに添付されている DBtutorial.zip に圧縮されて います。これを C:¥ の直下に解凍しておきます。

ここでは JCL バッチからの使用方法として紹介していますが、CICS や IMS プログラムで Oracle, DB2, SQL Server などとのデータベース連携を XA リソース方式で行う場合は同様の設定となります。

なお、対象のデータベースクライアントは同じ環境にインストール済みで、プリコンパイラも含め接続が 確認された環境であることが前提です。

本書作成時に使用したデータベース環境は下記の通りです。

Oracle

Oracle クライアント 11g 11.2.0.1.0 32bit DB: リモート環境(Oracle 10g 10.2.0.1.0 64bit) (Oracle Pro*COBOL プリコンパイラ使用)

DB2

DB2 クライアント 9.7.0 32bit DB: ローカル環境(DB2 9.7.0 32bit) (DB2 PREP プリコンパイラ使用)

SQL Server

ODBCドライバー: SQL Server Native Client 10.0 DB: ローカル環境(SQL Server 2008 R2 32bit) (ODBC 使用)

7.2 埋め込み SQL 付き COBOL ソース

オリジナルの埋め込み SQL 付き COBOL ソースは予め Micro Focus 形式の COBOL ソースにプリコ ンパイルしてから使用することも可能ですが、製品機能であるプリプロセッサ機能からプリコンパイラを 呼び出し内部的にプリコンパイルすることによりオリジナルのソースのイメージのままでのデバッグが 可能であり、管理もし易くなります。ここでは後者の方法を紹介します。

1) [プロジェクト] > [自動ビルド] をオフにしておきます。



COBOL エクスプローラ内で MSSDEMO を右クリックして [インポート] > [インポート] を選択します。

COB	X ISSD	<u>&</u> . ナビ) <u></u> サ−) □ □ (□ <u>な</u> ▽ EMO			
		New	•	1	
		コピー 貼り付け 削除(D) 移動(V) 名前変更(M) ビルド アクション	Ctrl+C Ctrl+V Delete F2		
アウト 表示する:	2 2 8	指令の確定 インポート(I) エクスポート(O) リフレッシュ(F) プロジェクトを閉じる(S)	F5		リモート COBOL プロジェクト ローカル COBOL プロジェクトをリモート プロジェクトへ変換 Net Express プロジェクトの変換 インポート(I)

3) 以下のダイアログで [一般] > [ファイルシステム] を選択し、[次へ] をクリックします。

選択	
ローカル・ファイル・システムから既存のプロジェクトへリソースをインポートします。	
インポート・ソースの選択(S):	
フィルター入力	
 → 一般 ● アーカイブ・ファイル ● ファイル・システム ● 既存プロジェクトをワークスペースへ ■ 設定 > CVS > Micro Focus >> Micro Focus インターフェイスマッパー >> Remote Systems >> インストール >> チーム >> ブニパノン、開致 	E
(?) < 戻る(B) 次へ(N) > 売了(F) キャンセル

4) 以下のダイアログの上の [参照] ボタンをクリックします。

ファイル・システム		
ソースは空にできません。		
欠のディレクトリーから(Y):		▼ 参照(R)
	ペア選択(c) 選択をすべて#256(0)	
タインのフィルター(1)… 洗フォルダー(L): MSSDEMO	(C)展示((3) 対面)(な.3、くら話はな(4))	参照(W)
+		
イノション	2)	
イノショノ □警告を出さずに既存リソースを上書き(O	5)	
コンション 警告を出さずに既存リソースを上書き(O Create top-level folder(C) 拡張(A) >>	<i>,</i>	
オンション 警告を出さずに既存リソースを上書き(O Create top-level folder(C) 拡張(A) >>		

5) 以下のダイアログで C:¥DBturotial を選択し、[OK] をクリックします。

🛯 🚢 ローカル ディスク (C:)	
D 🍶 app	
Þ 退 DB2	
퉬 DBtutorial	=
🛛 退 PerfLogs	_
🛛 퉬 Program Files	
🛛 📔 ProgramData	
>] Tutorials	
N Windows	•

6) C:¥DBturotial 直下に解凍されているファイルをすべて選択し、[完了] をクリックします。

e dbdemoi.jd @ TBLCRTE.cbl @ TBLDROR.cbl @ TBLFETC.cbl @ TBLISRT.cbl	▼ 参照(R)
dbdemoi.jd TBLCRTE.cbl TBLDROR.cbl TBLFETC.cbl TBLISRT.cbl	▼参照(R)
dbdemol.jd TBLCRTE.cbl TBLDROP.cbl TBLFETC.cbl TBLFETC.cbl TBLISRT.cbl	
	参照(W)

7) 今回追加インポートしたソースに対して個別の設定を行います。Shift キーまたは Ctrl キーを押しな がら対象ファイルを複数選択したら、右クリックしメニューから [プロパティー] を選択します。



8) 先の JCL チュートリアルでのプロパティー設定以外に、ここでは使用するデータベース製品に合わせ、埋め込み SQL 対応のプリプロセッサの設定を追加します。

Oracle の例: COBSQL プリプロセッサを使用しますので、以下の指令を追加します。

P(COBSQL) ENDP

プロパティー: TBLCRTE.cbl	, TBLDRORcbi, TBLFETC.cbi	:
フィルター入力	COBOL 🗘 * 🖒 *	-
COBOL	▼アイルの固有な設定を可能にする(F)	
	文字集合: alia方言: ASCII	
	ソースフォーマット: 固定 ・	
	メインフレームのコピー処理: COPY -	
	 □ 指令ファイルの生成 □ リストファイルを生成 □ デバッグ用にコンパイル(D) □ 出力の表示 □ J.GNT にコンパイル 	
	皆告レベル: 回復可能なエラーを含める(レベル E) 最大エラー数: 100	
	追加指令: DIALECT(ENTCOBOL) p(cobsql) endp	
	- デフォルトの頃元(D) (第用(A)	
?	ок <i>±</i> т>tли	

DB2 の例:

DB2 ECM プリプロセッサを使用しますので、以下の指令を追加します。 DB2(DB=SAMPLE,VALIDATE=RUN)

👿 プロパティー: TBLCRTE.cbl,	I, TBLDROP.cbl, TBLFETC.cbl	□ X
フィルター入力	COBOL 🔅 👻	¢ • ▼
COBOL	アイルの固有な設定を可能にする(F)	
	文字集合: ASCII 言語方言: Enterprise COBOL for z/OS ソース フォーマット: 固定 メインフレームのコピー処理: COPY 一 指令ファイルの生成	
	□ ウストック用にコンパイル(D) □ 出力の表示 ☑ .GNT にコンパイル 	
	警告レベル: 回復可能なエラーを含める(レベル E) 最大エラー数: 100 追加指令: 100	8200
	DIALEC I (ENICOBUL) do2(do=sample,validate=run)	*
	デフォルトの復元(D) 道用(A)	
?	OK キャンセル	

SQL Server の例:

OPENESQL プリプロセッサを使用しますので、以下の指令を追加します。 SQL(DBMAN=ODBC,BEHAVIOR=JCL,TARGETDB=MSSQLSERVER)

w プロパティー: TBLCRTE.cbl,	TBLDROP.cbl, TBLFETC.cbl		_ _ ×
フィルター入力	COBOL		$\Diamond \bullet \bullet \Diamond \bullet \bullet \bullet$
▷ COBOL	マープアイルの固有な設定を可能にする(F)		^
	文字集合: 重語方言: ソース フォーマット:	SCII Interprise COBOL for z/OS	
	メインフレームのコピー処理:		
	 描令ファイルの生成 リストファイルを生成 デバッグ用にコンパイル(D) 出力の表示 GNT にコンパイル 		E
	警告レベル:	回復可能なエラーを含める(レベル E) ・	
	最大エラー数: 1	00	
	違加指令: DIALECT(ENTCOBOL) SQL(DBMAN=ODBC,BEHAVIOR	R=JCL, TARGETDB=MSSQLSERVER)	*
?		<u>ОК</u> ≠ 1	PYTH

9) CICS プリプロセッサは不要ですので以下のようにチェックを外します。

WF プロパティー: TBLCRTE.cbl	, TBLDRORcbl, TBLFETC.cbl	_ D ×
フィルター入力	CICS 設定	⇔ • ⇔ • •
COBOL CICS 設定 SQL 設定	☑ ファイルの固有な設定を可能にする(F)	
	■ Exec cics サポートされない機能の処理	
	○美行時にアベント ○コメント扱い ◎コンパイルエラー	
	サポートされないオブションの処理 の実行時にアベンド のコンパイルエテー	
	◎ 実行時は無視	
		^
		Ŧ
	「デフォルトの頃元(D)	適用(A)
?	ОК	キャンセル

10) 既に 8) で埋め込み SQL 対応のプリプロセッサを使用するために追加指令を行っていますので、
 ここでは[EXEC SQL] のチェックをオフにして、[OK] をクリックします。(必要に応じて使い分け可能です)

ブロバティー: TBLCRTE.cbl,	TBLDROR.cbl, TBLFETC.cbl	And an other statements of the statement		_ D X
フィルター入力	SQL 設定			↔ + ⇒ + +
COBOL CICS 設定 SQL 設定	- · ファイルの固有な設定を可能にする(F))		
	EXEC SQL: プリプロセッサの種類: OpenESQL 指令: フィルタテキストを入力			*
	設定	値		
	ACCESS			
	ACCESS			E
	ANEIOZENTRY			
	AUTOCOMMIT			
	REHAVIOR			
	BEINVIOR			
	CHECK			
	CHECK			
	CHECKBOPCORSOR			
	CONCAT			
	CONNECTIONROOL			
	CTRACE			
	CURSORCASE			
	CORSORCASE			₹
				^
	SOL 指令: <括弧内の継承された設定	>		
	(これらの指令は有効でけたりません	"EVEC SOI" オブションを使って有効にしてください)		
	(こ1・ワジョウは行がてはのりよせん。	ERECSQL AJ ZEZ CEMMILUE (LEUN)		- -
			デフォルトの復元(D)	適用(A)
?			ОК	キャンセル

7.3 プログラムのコンパイル

1) [自動的にビルド] をオンにします。



2) 自動ビルドが実行され、以下のように再コンパイルが完了します。なお、警告が出力される場合が ありますが、この例題では無視して構いません。

🖳 コンソール 🛙 🖹 問題 🙆 タスク	
COBOL Build	
imscobbuild:	
combinedbuild:	
BUILD SUCCESSFUL	
Build finished with no errors.	
Total time: 1 second	

7.4 XA スイッチモジュールの生成

この環境で実行するプログラム内では、XA スイッチモジュール経由でデータベースと接続することになります。

ここでは使用するデータベース製品に合わせた XA スイッチモジュールを作成します。作成手順は以下の通りです。

 スタートメニューで、[Micro Focus Enterprise Developer] > [ツール] > [Enterprise Developer コ マンドプロンプト(32-bit)] を管理者ユーザで起動します。<Enterprise Developer インストールディ レクトリ>¥src¥enterpriseserver¥xa ディレクトリに移動します。



 2) 使用するデータベース製品に合わせた XA スイッチモジュールを build コマンドで作成します。JCL 機能で IKJEFT01 ユーティリティを使用する場合は、xxx1pc.dll サブモジュール生成も必要です。

Oracle の例: build < ora10 または ora11 > を実行します。ここでは下記コマンドを実行します。 build ora11 C:¥Program Files¥Micro Focus¥Enterprise Developer¥src¥enterpriseserver¥xa>build orall Building 32-bit switch module... Micro Focus COBOL Version 2.1.01046 Copyright (C) Micro Focus 1984-2013. All rights reserved. * チェック終了:エラー(はありません-コード生成を開始します * Generating ESORAXA * Data: 3256 Code: 6568 Literals: 3072 Micro Focus COBOL - CBLLINK utility Version 2.1.0.88 Copyright (C) Micro Focus 1984-2013. All rights reserved. Microsoft (R) Incremental Linker Version 10.00.40219.01 Copyright (C) Microsoft Corporation. All rights reserved. ESORAXA.obj cbllds00001058.obj Creating library ESORA11XA.lib and object ESORA11XA.exp Microsoft (R) Manifest Tool version 5.2.3790.2076 Copyright (c) Microsoft Corporation 2005. All rights reserved.

次に下記コマンドを実行します。

build ora1pc

C:¥Program Files¥Micro Focus¥Enterprise Developer¥src¥enterpriseserver¥xa>build oralpc Building 32-bit switch module... Micro Focus COBOL Version 2.1.01046 Copyright (C) Micro Focus 1984-2013. All rights reserved. * Cobsql Integrated Preprocessor * CSQL-I-018: Oracle ブリコンバイラトランスレータを起動します。 * CSQL-I-020: Oracle ブリコンバイラの出力を処理中。 * CSQL-I-001: COBSQL:チェッカへの引き渡しを完了しました。 * チェック終了:エラーはありません-コード生成を開始します * Generating ESORAOPC * Data: 4450 Code: 11272 Literate: 4456 Literals: * Data: Code: 11272 3360 Micro Focus COBOL - CBLLINK utility Version 2.1.0.88 Copyright (C) Micro Focus 1984-2013. All rights reserved. Microsoft (R) Incremental Linker Version 10.00.40219.01 Copyright (C) Microsoft Corporation. All rights reserved. ESORAOPC.obj cbllds00000FA8.obj Creating library ESORAOPC.lib and object ESORAOPC.exp Microsoft (R) Manifest Tool version 5.2.3790.2076 Copyright (c) Microsoft Corporation 2005. All rights reserved. If you have upgraded your installation from an earlier version of Net Express or Studio, then you may need to additionally rebuild the main switch module, that is, execute : build ora If you intend to execute JES-initiated transactions under Enterprise Server you must copy ESORAOPC.DLL to a directory on your PATH, such as your COBOL bin directory. Attempting to copy ESORAOPC.DLL to your COBOL bin directory... C:¥Program Files¥Micro Focus¥Enterprise Developer¥src¥enterpriseserver¥xa≻

DB2 の例:

ここでは下記コマンドを実行します。

build db2

C:¥Program Files¥Micro Focus¥Enterprise Developer¥src¥enterpriseserver¥xa>build db2 Building 32-bit switch module Micro Focus COBOL - CBLLINK utility Version 2.1.0.88 Copyright (C) Micro Focus 1984-2013. All rights reserved.	ш
Micro Focus COBOL Version 2.1.01046 Copyright (C) Micro Focus 1984-2013. All rights reserved. * チェック終了:エラーはありません-コード生成を開始します * Generating ESDB2XA * Data: 3064 Code: 5760 Literals: 2892 Microsoft (R) Incremental Linker Version 10.00.40219.01 Copyright (C) Microsoft Corporation. All rights reserved.	
ESDB2XA.obj cbllds000008C8.obj Creating library ESDB2XA.lib and object ESDB2XA.exp Microsoft (R) Manifest Tool version 5.2.3790.2076 Copyright (c) Microsoft Corporation 2005. All rights reserved.	
C:¥Program Files¥Micro Focus¥Enterprise Developer¥src¥enterpriseserver¥xa>	

次に build db21pc < DB のエイリアス > を実行します。ここでは下記コマンドを実行します。

build db21pc sample



SQL Server の例:

ここでは下記コマンドを実行します。

build mssql

C:¥Program Files¥Micro Focus¥Enterprise Developer¥src¥enterpriseserver¥xa>build mssal Building 32-bit switch module... Micro Focus COBOL - CBLLINK utility Version 2.1.0.90 Copyright (C) Micro Focus 1984-2013. All rights reserved. Micro Focus COBOL Version 2.1.01046 Copyright (C) Micro Focus 1984-2013. All rights reserved. * チェック終了:エラーはありません- コード生成を開始します * Generating ESMSSOL Data: Code: 17080 Literals: 7172 Microsoft (R) Incremental Linker Version 10.00.40219.01 Copyright (C) Microsoft Corporation. All rights reserved. ESMSSQL.obj cbllds00000C5C.obj Creating library ESMSSQL.lib and object ESMSSQL.exp Microsoft (R) Manifest Tool version 5.2.3790.2076 Copyright (c) Microsoft Corporation 2005. All rights reserved. If you intend to execute JES-initiated transactions under Enterprise Server you must copy ESMSSQL.DLL to a directory on your PATH, such as your COBOL bin directory. Attempting to copy ESMSSQL.DLL to your COBOL bin directory... C:¥Program Files¥Micro Focus¥Enterprise Developer¥src¥enterpriseserver¥xa>

7.5 XA リソースの設定

本チュートリアルでは例題 JCL を Enterprise Server にサブミットして実行します。ここでは CICS および JCL チュートリアルで使用した MSSDEMO リージョンを使用しますので、XA リソースを使用可能と するための準備を行います。

- 1) MSSDEMO が開始状態の場合にはいったん停止します。
- 2) Enterprise Server Admin にて MSSDEMO の [編集] ボタンをクリックします。

(編集	MFES (MSS)	MSSDEMO	停止 開始	1 top:10.18.11.116*:61228* (WIN-8AK22918G88.microfocus.com) 3 リスナー 詳細	-/10	Default	Server: [CP 1: [CASST(

3) [サーバー] > [プロパティ] > [XA リソース] を開き、[追加] ボタンをクリックします。

<mark>ーバー…</mark> リスナー (3) サービス	(4) 】ハンドラ (3) 】 パッケージ (0)
<mark>プロパティ</mark> 構成 診断)	過去の統計
一般 】 XAリソース (0) MSS	(♥) スクリプト アクセス権 セキュリティ

4) 使用するデータベース製品に合わせた XA リソースを設定します。

ID: IKJEFT ユーティリティに渡す DSN TSO コマンドの SYSTEM パラメタで渡す DB2 サブシステム名(このチュートリアルでは XADB とします)

名前: XA リソース名として任意の名前を指定 モジュール: XA スイッチメインモジュールのパス(xxx1pc.dll は指定無し) OPEN 文字列: DB オープン用の指定(設定済みの DB 環境設定に依存)

Oracle の例:

例えば以下のように設定し、[追加] ボタンをクリックします。

เื− เ≀ิ ว ())リスナー (3) - 構成 〕 診 Aリンース (0)	サービス (4))。 湖…) 過去の MSS… (�) 〕	ハンドラ (3) 統計 スクリプト	パッケージ (0 アクセス権)]
ID:					
XADB					
ORACLE	ХА				
モジュー	_/**				
ocus\Ent OPEN文 ORACLE	rrprise Developer 콘列: _XA+SesTm=10	\src\enterprises 0+SqlNet=tok-p	server\xa\ES par+Acc=P/s	ORA11XA.DLL	
CLOSES 説明:	· ··· ··91]:				
有効: 🔽					

DB2 の例:

例えば以下のように設定し、[追加] ボタンをクリックします。

⁹ ロパラ	イ 構成 診断 過去の統計
[→] 报2	スムリンース (U) MISS (♥) スクリフト アクセス権 セキュリティ
名前:	
DB2	KA.
モジュ	ール:
Foc	s\Enterprise Developer\src\enterpriseserver\xa\ESDB2XA.DLL
OPE	文字列:
	ample CT=100
DB=	
DB=	E文字列:
DB= CLOS 説明:	E文字列:
DB= CLOS 説明: 有効:	■文字列:

SQL Server の例:

例えば以下のように設定し、[追加] ボタンをクリックします。

< >	Server MSSDEMO [停止]
ナーバー	リスナー (2) サ <i>ー</i> ビス (4) ハンドラ (3) パッケージ (0)
プロパ	<mark>ティ</mark> 構成 】診断 過去の統計
一般	XAUソース (0) MSS (✔) スクリプト アクセス権 セキュリティ
ID:	
XAD	B
名前	:
SQL	SVR XA
モジ	 ⊐ ─ J⊎:
Foc	us/Enterprise Developer/src/enterpriseserver/xa/ESMSSQL DLL
OPF	N文字列:
DSM	I=JCLTEST1
CLO	SE文字列:
020	
108E	
67.71	
有効	: 🔽
追加	ロ キャンセル

5) MSSDEMO を開始します。



 XA スイッチモジュールが Enterprise Server に正しくロードできると、Enterprise Server Administration のコンソールログには以下のメッセージが出力されます。(DB2 の場合、 NAME(DB2 for WINDOWS) となり、SQL Server の場合、NAME(Micro Focus SQL Server XA struct) となります。)

20123	<u> </u>	
ナーバ・	ー リスナー (3) サービス (4) ハンドラ (3) パッ	ケージ (0)
ロバテ	イ 構成 診断 過去の統計	
トレース	」 ダンプ <mark>ESコンソール</mark> CSコンソール	
画面	更新 Show entries from 1 to 10 ③ Show last 10 lines	of 45 total entries
Entry	Event	Show Entire Log
Entry 38	Event 130927 18090515 1976 MSSDEMO JES000042I SSTM not enabled: CICS	Show Entire Log
Entry 38 37	Event 130927 18090515 1978 MSSDEMO JES0000421 SSTM not enabled: CICS 130927 18090536 1978 MSSDEMO CASXO00201 XADB XA interface loade	Show Entire Log 18:09:04 d. Name(Oracle_XA), Registration Mode(Static) 18:09:
Entry 38 37 38	Event 130927 18090515 1976 MSSDEMO JE5000042I SSTM not enabled: CICS 130927 18090536 1976 MSSDEMO CASX00020I XADB XA interface loade 130927 18090566 1976 MSSDEMC CASX00021I XADB XA interface local	18:09:04 d. Name(Oracle_XA). Registration Mode(Static) 18:09:05
Entry 36 37 38 39	Event 130927 18090515 1976 MSSDEMO JE5000042I SSTM not enabled: CICS 130927 18090536 1978 MSSDEMO CASX0020I XADB XA interface loade 130927 18090566 1976 MSSDEMO CASX00021I XADB XA interface local 130927 18090801 1976 MSSDEMO CASX0015I XADB XA interface initial	Show Entire Log 18:09:04 d. Name(Oracle_XA), Registration Mode(Static) 18:09:05 transaction support enabled 18:09:05 ized successfully 18:09:05
Entry 38 37 38 39 40	Event 130927 18090515 1976 MSSDEMO JE5000042I SSTM not enabled: CICS 130927 18090536 1976 MSSDEMO CASX00020I XADB XA interface loade 130927 18090556 1976 MSSDEMO CASX00021I XADB XA interface local 130927 18090601 1976 MSSDEMO CASSI5001I PLTPI - No PLT Specified	Show Entire Log 18:09:04 d. Name(Oracle_XA), Registration Mode(Static) 18:09: transaction support enabled 18:09:05 ized successfully 18:09:05 i 18:09:08
Entry 38 37 38 39 40 41	Event 130927 18090515 1976 MSSDEMO JES000042I SSTM not enabled: CICS 130927 18090536 1976 MSSDEMO CASX00020I XADB XA interface loade 130927 18090566 1976 MSSDEMO CASX00021I XADB XA interface loade 130927 18090801 1976 MSSDEMO CASSI6001I PLTPI - No PLT Specified 130927 18090821 1976 MSSDEMO CASSI6001I PLTPI - No PLT Specified 130927 18090821 1976 MSSDEMO CASSI6040I Active SEP memory stra	Show Entire Log 18:09:04 kd. Name(Oracle_XA), Registration Mode(Static) 18:09:05 transaction support enabled 18:09:05 ized successfully 18:09:05 i 18:09:06 tegy set to x'00000001', retain count 100 18:09:08
Entry 38 37 38 39 40 41 42	Event 130927 18090515 1976 MSSDEMO JES000042I SSTM not enabled: CICS 130927 18090538 1976 MSSDEMO CASX00020I XADB XA interface loade 130927 18090568 1976 MSSDEMO CASX00021I XADB XA interface load 130927 18090801 1976 MSSDEMO CASSI601I PLTPI - No PLT Specified 130927 18090821 1976 MSSDEMO CASSI600I Active SEP memory stra 130927 18090821 MSDEMO CASSI600I Active SEP memory	Show Entire Log 18:09:04 kd. Name(Oracle_XA). Registration Mode(Static) 18:09: transaction support enabled 18:09:05 ized successfully 18:09:05 i 18:09:06 tegy set to x'00000001', retain count 100 18:09:08 ed successfully 18:09:06
Entry 38 37 38 39 40 41 42 43	Event 130927 18090515 1976 MSSDEMO JES000042I SSTM not enabled: CICS 130927 18090536 1976 MSSDEMO CASX00020I XADB XA interface loade 130927 18090568 1976 MSSDEMO CASX00021I XADB XA interface loade 130927 18090621 1976 MSSDEMO CASSI6001I PLTPI - No PLT Specified 130927 18090821 1976 MSSDEMO CASSI6001 Active SEP memory stra 130927 18090821 1976 MSSDEMO CASSI6001 Active SEP memory stra 130927 18090821 1976 MSSDEMO CASSI6001 SEP initialization complete 130927 18090862 1280 MSSDEMO CASSI0005I Batch initiator started for	Show Entire Log 18:09:04 kd. Name(Oracle_XA). Registration Mode(Static) 18:09:05 transaction support enabled 18:09:05 ized successfully 18:09:05 # 18:09:06 tegy set to x'00000001', retain count 100 18:09:08 ef successfully 18:09:06 r job classes "ABC" 18:09:06
Entry 38 37 38 39 40 41 42 43 44	Event 130927 18090515 1976 MSSDEMO JES000042I SSTM not enabled: CICS 130927 18090538 1976 MSSDEMO CASX00020I XADB XA interface loade 130927 18090601 1976 MSSDEMO CASX00021I XADB XA interface load 130927 18090601 1976 MSSDEMO CASSI6001I PLTPI - No PLT Specified 130927 18090621 1976 MSSDEMO CASSI6001 SEP initialization complete 130927 18090862 1280 MSSDEMO CASSI6001 SEP initialization complete 130927 18090867 4404 MSSDEMO CASSI6021I PLTPI Phase 2 - No PLT	Show Entire Log 18:09:04 kd. Name(Oracle_XA), Registration Mode(Static) 18:09:05 transaction support enabled 18:09:05 ized successfully 18:09:05 # 18:09:06 tegy set to x'00000001', retain count 100 18:09:08 ed successfully 18:09:08 tj ob classes "ABC" 18:09:08 Specified 18:09:08

7.6 データベースアクセスを含む COBOL バッチプログラムの実行

1) COBOL エクスプローラ内で dbdemo1.jcl をダブルクリックしエディタで開きます。



このジョブは4つのステップから構成され下記の様に連携されています。

• STEP01:

DB にテーブルを新規作成

- STEP02:
- JCL に記述された SYSIN データをテーブルに追加
- STEP03:
- テーブルからデータを全件読み出し SYSOUT へ出力
- STEP04:
- テーブルを削除
- COBOL エクスプローラ内で dbdemo1.jcl を右クリックして [Enterprise Server へのサブミット] を選択します。

mo1.jcl	RUN PROGRAM(TBL
New	×
開く(0) アプリケーションから開く	•
⊐Ľ-	Ctrl+C
貼り付け	Ctrl+V
削除(D)	Delete 1
移動(V)	
名前変更(M)	F2
Enterprise Server へのサブミ	v۲
	mol.jcl New 開く(0) アプリケーションから開く コピー 貼り付け 静除(D) 移動(V) 名前変更(M)

3) 以下のように DBDEMO1 ジョブが実行された旨のメッセージがコンソールに表示されます。



以降は、JCL チュートリアルを参考にして、スプールビューの Out Hold キューおよび Output キューを 開き、実行された DBDEMO1 ジョブの結果を確認したり、バッチプログラムのデバッグも同様に実行し てみてください。また、各 DB 付属の管理ツールを使用すると、各ステップにおける DB テーブルの状況 が確認できます。

以上、