

平成16年度
流通サプライチェーン全体最適化促進事業

＜実証実験＞

環境設定書
別紙「ビジネスモジュール導入手順書」

平成17年3月
日本電気株式会社

改版履歴

日付	版数	改版内容
2005 年 3 月 31 日	初版	新規

実証実験統括責任者
日本電気株式会社 ： 曾根田 雄

検 印

目 次

1. ビジネスモジュールインストール手順書概略	1 - 1
1.1 インストール手順概略	1 - 1
1.2 インストール後の作業	1 - 1
1.3 推奨されるハードウェアスペック	1 - 1
1.4 前提条件	1 - 2
2. プラットフォームインストール	2 - 1
2.1 BMに必要なOS, ミドルウェア	2 - 1
2.2 OSとミドルウェアのインストール	2 - 1
3. BMインストール前準備	3 - 1
3.1 作業概略と作業の一覧	3 - 1
3.2 BizEngine 環境用前準備	3 - 1
3.3 WLI 環境用前準備	3 - 2
4. BMインストール	4 - 1
4.1 作業概略と作業の一覧	4 - 1
4.2 BMインストール作業詳細	4 - 1
5. 動作確認	5 - 1
5.1 単一サーバでのセルフテスト	5 - 1
5.2 複数サーバ間の疎通テスト	5 - 3
6. BMクライアント型通信アダプタのインストール（オプション）	6 - 1
6.1 概要	6 - 1
6.2 BMクライアント型通信アダプタのプラットフォーム要件	6 - 1
6.3 BMクライアント型通信アダプタのインストール	6 - 2
6.4 BM本体の設定	6 - 4
6.5 クライアント型通信アダプタの開始と終了	6 - 4
7. BMのカスタマイズ方法	7 - 1
7.1 業務アプリケーションアダプタのデプロイ	7 - 1
7.2 通信設定のコンフィグレーション	7 - 1
8. Appendix	8 - 1
8.1 BMとイベントスケジューラの起動・終了方法	8 - 1
8.2 WLIでのコンフィグレーション作成手順	8 - 2
8.3 WLIでのセルフテスト実施の為の作成方法	8 - 6
8.4 WLIでのCPAの作成方法	8 - 1 0

9. ライセンス	9-1
----------------	-----

1. ビジネスモジュールインストール手順書概略

1.1 インストール手順概略

ビジネスモジュール（以下、**BM** と記す）をインストールするための手順を記述する。**BM** のインストールは大まかに下記の手順に沿って行なわれる。

表 1.1 インストール手順概略

#	手順名	作業内容	説明章番号
1	プラットフォームインストール	BM 稼動の前提となるプラットフォームアプリケーション (DB、通信サーバ等) のインストール	2
2	BM インストール前準備	通信コンポーネントの設定等	3
3	BM インストール	BM 本体のインストール	4

詳細な作業内容はインストール先の **OS** と通信サーバの組み合わせにより異なる。以下、適宜この組み合わせの単位ごとに具体的な作業の詳細手順を記述する。

1.2 インストール後の作業

当インストール手順書ではインストール後に行うと想定される作業も記述する。インストール後の作業の概略は下記に示すとおりである。

表 1.2 インストール後の作業概略

#	手順名	作業内容	説明章番号
1	動作確認	BM が正しくインストールできたかどうかの確認手順	5
2	BM のカスタマイズ方法	2 つの拠点間でデータ通信を行うための設定手順	7

1.3 推奨されるハードウェアスペック

BM を稼動させるのに推奨されるハードウェアスペックは以下のとおりである。

表 1.3 **BM** 稼動のための推奨ハードウェアスペック

#		Windows	Solaris
1	CPU	Pentium 1GHz 以上	UltraSPARC-II 500MHz 以上
2	メモリ	1G バイト以上	1G バイト以上
3	HDD 容量	※1	※1
4	通信環境	Ethernet	Ethernet

※1 トランザクション量により異なる。**BM** インストールには **20M** バイト必要。

1.4 前提条件

本書では、以下の条件でのインストールについての記述である。

- アプリケーションサーバは **7001(HTTP)**と **7002(HTTPS)**のポートを使用する。
- アプリケーションサーバのユーザ名/パスワードは **system/weblogic** を使用する。
- **DB** サーバは **1521** のポートを使用する。
- **DB** 名として **BPCDB** を使用する。
- **DB** のユーザ名/パスワードは、**WEBLOGIC/WEBLOGIC** 及び **BPC/BPC** を使用する。
- **DB** サーバとビジネスモジュールは同一のサーバで実行する。

2. プラットフォームインストール

2.1 BMに必要なOS, ミドルウェア

BMに必要なOSとミドルウェアは以下のものとなる。当インストール手順書では以下の3つの環境構成を想定し、それぞれ構成A、構成B、構成Cとして具体的に説明する。

(構成A) Windows + WebLogic Server + BizEngine 環境

(構成B) Solaris + WebLogic Integration 環境

(構成C) Windows + WebLogic Integration 環境

※ WebLogic Integration は以降、WLI と記す。

2.2 OSとミドルウェアのインストール

BMのインストール前に、下記の表2. 1のOSとミドルウェアをインストールしておく必要がある。これらのOS、ミドルウェアのインストール方法については、本手順書では記述しない。各OS、ミドルウェアのマニュアルを参照のこと。

表 2.1 BMのインストールに必要なOS, ミドルウェア

#	OS ミドルウェア	構成 A (Windows+BizEngine)	構成 B (Solaris+WLI)	構成 C (Windows+WLI)
1	OS	Windows 2000 または Windows Server 2003	Solaris 9	Windows 2000 または Windows Server 2003
2	JDK(※1)	Java2 SDK SE 1.4.1_05	Java2 SDK SE 1.4.2_04	Java2 SDK SE 1.4.2_04
3	アプリケーションサーバ	WebLogic Server 8.1 SP2	-(※2)	-(※2)
4	通信サーバ	BizEngine 3.0.2	WebLogic Integration 8.1 SP3 + パッチ (※3)	WebLogic Integration 8.1 SP3 + パッチ (※3)
5	データベース	Oracle 9.2.0.1.0 (Standard Edition or Enterprise Edition)	Oracle 9.2.0.1.0 (Standard Edition or Enterprise Edition)	Oracle 9.2.0.1.0 (Standard Edition or Enterprise Edition)
6	OS 作業ユーザ 名(※4)	Administrator 権限のある 任意のユーザ	【bpc】ユーザ	Administrator 権限のある 任意のユーザ
7	OS 作業ユーザ グループ名	任意のグループ	任意のグループ	任意のグループ

※1 Java2 SDK については、WebLogic Server または、WLI に付属のものを使用するので、Java2 SDK のインストールは不要。ただし、Java2 SDK に必要な OS パッチの投入は必要。

※2 通信サーバとして WLI を利用する場合、WebLogic Server 8.1SP3 がこれに付属しているためアプリケーションサーバを別途用意する必要はない。

※3 適用するパッチについては後述する。

※4 【OS 作業ユーザ】とは、OS インストール後にミドルウェアをインストールする際作業するユーザのこと。指定のユーザがインストール先サーバに用意されていない場合はユーザを新規作成してミドルウェアのインストールを行う必要がある。

3. BM インストール前準備

3.1 作業概略と作業の一覧

BM インストール前準備とは、前章でインストールした OS やミドルウェアと BM を連携させるための作業を指している。下記にそれぞれの環境で必要/不要である前準備の一覧を示す（表中【○】は必要、【×】は不要を示す）。

表 3.1 環境構成別のインストール作業一覧

#	前準備名	構成 A (Windows+BizEngine)	構成 B (Solaris+WLI)	構成 C (Windows+WLI)	節番号
1	BizEngine 環境用前準備	○	×	×	3.2
2	WLI 環境用前準備	×	○	○	3.3

3.2 BizEngine 環境用前準備

以下に BizEngine と BM を連携させるための前準備を記述する。

(1) 環境変数の設定(for BizEngine)

環境変数はビジネスモジュールおよび、ミドルウェアのインストールディレクトリを識別するため、以下のように設定する。

本手順書ではデフォルトのディレクトリにインストールされると想定している。環境変数 ORACLE_HOME は Oracle のインストールディレクトリではなく、ora92 のディレクトリを指定する。

表 3.2 インストールディレクトリを表す環境変数(for BizEngine)

#	環境変数名	設定値	本ドキュメントで想定するインストールディレクトリ ※1
1	JAVA_HOME	Java2 SDK のホームディレクトリ	C:\¥bea¥jdk141_05
2	BEA_HOME	WebLogic Server のホームディレクトリ	C:\¥bea
3	BIZ_HOME	BizEngine のホームディレクトリ	C:\¥BizEngine
4	BPC_HOME	BM の ホームディレクトリ(提供している zip を解凍して作成された【bpc】フォルダの場所)	C:\¥bpc
5	ORACLE_HOME	ORACLE のホームディレクトリ	C:\¥oracle¥ora92

※1 Windows のシステム環境変数に設定する。

※2 パス名には空白（スペース）漢字を使用しないこと(英数字を使用する)。

3.3 WLI 環境用前準備

以下に **WLI** と **BM** を連携させるための前準備を記述する。

(1) 環境変数の設定(for WLI)

環境変数はビジネスモジュールおよび、ミドルウェアのインストールディレクトリを識別するため、以下のように設定する。

プロダクトによっては、インストールディレクトリ内のディレクトリ配置をインストール時に指定可能なことがあるが、ディレクトリ配置はデフォルト設定を使用しなければならない。

環境変数 **ORACLE_HOME** は **Oracle** のインストールディレクトリではなく、**Solaris** の場合は **orahome**, **Windows** の場合は **ora92** のディレクトリを指定する。

表 3.3 インストールディレクトリを表す環境変数(for WLI)

#	環境変数名	設定値	本ドキュメントで想定するインストールディレクトリ	
			構成 B(Solaris+WLI) ※1	構成 C(Windows+WLI)※2
1	JAVA_HOME	Java2 SDK のホームディレクトリ	/export/home/bpc/bea/jdk142_04	C:¥bea¥jdk141_05
2	BEA_HOME	WebLogic Integration のホームディレクトリ	/export/home/bpc/bea	C:¥bea
3	BPC_HOME	BM の ホームディレクトリ (提供している zip を解凍して作成された【bpc】フォルダの場所)	/export/home/bpc/apl	C:¥bpc
4	ORACLE_HOME	ORACLE のホームディレクトリ	/export/home/oracle/orahome	C:¥oracle¥ora92

※1 構成 B(Solaris 環境)では、bpc ユーザの.cshrc(ログインシェルに **cs**h を使用する場合)、または、.profile(ログインシェルに **sh** を使用する場合)に環境変数定義を加える。

※2 構成 C(Windows 環境)では、**Windows** のシステム環境変数に環境変数を追加する。

※3 パス名には空白 (スペース) 漢字を使用しないこと(英数字を使用する)。

(2) パッチの適用

BM を **WLI** で動作させるために以下のパッチを適用させる。パッチは構成 **B(Solaris 環境)**、構成 **C(Windows 環境)**を問わず同じ方法で適用する。

以下のパッチファイルを適用する。

表 3.4 パッチファイル一覧

#	パッチファイル名	適用方法
1	CR201789_813.jar	\$BEA_HOME/weblogic81/server/lib に置く

2	CR179600_81sp3.jar	\$BEA_HOME/weblogic81/server/lib に置く
3	wliconsole.war	\$BEA_HOME/weblogic81/integration/lib/console/wliconsole.war をこのファイルと入れ替える（バックアップは取っておくこと）
4	wlxbean_CR189655_sp3.jar	\$BEA_HOME/weblogic81/server/lib/wlxbean.jar をこのファイルと入れ替える（バックアップは取っておくこと）

(3) WLI データベースのコンフィグレーション

WLI が利用するデータベーススキーマ定義を以下の手順に沿って行う。

① Oracle ユーザの作成

SQL*Plus 等を使用して、提供媒体に含まれる、下表の SQL スクリプトを SYSTEM ユーザで実行する。下表に従い環境に応じたスクリプトを実行することより、WLI で使用する Oracle ユーザが作成される。作成されるユーザの情報を表 3. 6 に示す。

表 3.5 WLI の Oracle ユーザの作成 SQL スクリプト

対象環境	SQL スクリプト ファイル名
【構成 B】	./wli/unix/createWliUser.sql
【構成 C】	./wli/windows/createWliUser.sql

表 3.6 WLI 用の Oracle ユーザの設定情報

#	項目名	設定値
1	ユーザ名	WEBLOGIC
2	パスワード	WEBLOGIC
3	デフォルト表領域	USERS
4	テンポラリ表領域	TEMP

② テーブルの作成

SQL*Plus 等を使用して、WLI に含まれる、下表の SQL スクリプトを WEBLOGIC ユーザで実行する。

表 3.7 WLI のスキーマ修正 SQL スクリプト

対象環境	SQL スクリプト ファイル名
【構成 B】	\$BEA_HOME/weblogic81/integration/dbscripts/oracle/wli_archive.sql
【構成 C】	\$BEA_HOME/weblogic81/integration/dbscripts/oracle/wli_runtime.sql

③ スキーマの修正

SQL*Plus 等を使用して、提供媒体に含まれる、下表の SQL スクリプトを WEBLOGIC ユーザで実行する。下表に従い環境に応じたスクリプトを実行することより、スキーマの修正が行われる。

表 3.8 WLI のスキーマ修正 SQL スクリプト

対象環境	SQL スクリプト ファイル名
【構成 B】	./wli/unix/upgradeWli.sql
【構成 C】	./wli/windows/upgradeWli.sql

④ Oracle 初期化パラメータの設定値の変更

Oracle の初期化パラメータの値を以下のように変更する。

表 3.9 Oracle 初期化パラメータの設定値の変更

#	設定項目	初期値	設定値
1	open_cursors	300	3000

(4) アプリケーションサーバのコンフィグレーションの作成

WLI では、BM が動作するドメインを作成する必要がある。ここでは、BM 用の **Integration** ドメインを作成する手順を示す。

① BM のコンフィグレーションテンプレートファイルのコピー

提供媒体に含まれる、`./wli/BM-tmpl.jar` ファイルを
`$BEA_HOME/weblogic81/common/templates/domains` へコピーする。

② BM の **Integration** ドメインの作成

以下のコマンドを入力しコンフィグレーションを作成する。

表 3.10 コンフィグレーション作成のためのスクリプト

構成	入力コマンド
構成 B(Solaris 環境) ※.1	<code>\$BEA_HOME/weblogic81/common/bin/config.sh -mode=console</code>
構成 C(Windows 環境)	<code>\$BEA_HOME/weblogic81/common/bin/config.cmd -mode=console</code>

※ ドメイン テンプレートは「WebLogic Integration Domain For BM 8.1.3.0」を使用する。
(詳しくは、Appendix 8.2 を参照すること)

4. BM インストール

4.1 作業概略と作業の一覧

BM 本体のインストール作業は OS や通信サーバの種類に関係なく以下の手順に沿って行なわれる。

表 4.1 BM インストールの作業一覧

#	実施項目	説明章番号
1	BM コンポーネントのインストール	4.2. (1)
2	BM 用データベースの作成	4.2. (2)
3	BM の環境設定	4.2. (3)

4.2 BM インストール作業詳細

(1) BM コンポーネントのインストール

ビジネスモジュールの提供媒体には表 4.2 のファイルが含まれる。提供ファイルのアーカイブを解凍し、表 3.2、表 3.3 で示した環境変数を適切に設定した後、下表に従い環境に応じたインストールバッチを実行することにより、必要なファイルがコピーされる。

表 4.2 提供媒体に含まれるファイル

ファイル名	内容	対象環境
BM.zip	ビジネスモジュールで提供されるファイルのアーカイブ	(全て)
instBiz.bat	Windows + WebLogic + BizEngine の環境用インストールバッチ	【構成 A】
instWLI.sh	Solaris9 + WebLogic Integration の環境用インストールバッチ	【構成 B】
instWLI.bat	Windows + WebLogic + BizEngine の環境用インストールバッチ	【構成 C】

※構成 B の場合で、提供メディア上の instWLI.sh に実行パーミッションが無い場合は、インストールを行うマシンのハードディスクへ全ファイルをコピーし、instWLI.sh に chmod コマンドで実行パーミッションを設定する必要がある。

※提供媒体には、他にクライアント型通信を行う場合の構成で使用される instbizx.bat が含まれるが、本ドキュメントの対象ではない為、表へは記載していない。

インストールバッチを実行後、環境変数 BPC_HOME で指定されたディレクトリには、以下のサブディレクトリが作成される。

表 4.3 ビジネスモジュールインストール時に生成されるディレクトリ

ディレクトリ	格納されるファイル
Bin	ビジネスモジュールの起動終了の起動、各種管理用ツール
Config	コンフィグファイル
Setup	ビジネスモジュールのインストールと環境設定に使用されるファイル
Lib	ライブラリファイル
Log	ログファイル(インストール直後はファイルは存在しない)
Profile	テスト用業務アダプタのデータファイル

(2) BM 用データベーススキーマの作成

BM 利用するデータベーススキーマ定義を以下の手順に沿って行う。

① BM 用の Oracle ユーザの作成

SQL*Plus 等を使用して、下記の SQL スクリプトを **SYSTEM** ユーザで実行する。下表に従い環境に応じたスクリプトを実行することより、**BM** で使用する **Oracle** ユーザが作成される。作成されるユーザの情報を表に示す。

表 4.4 BM の Oracle ユーザの作成 SQL スクリプト

対象環境	SQL スクリプト ファイル名
【構成 A】【構成 C】	\$BPC_HOME/setup/db/windows/createUser.sql
【構成 B】	\$BPC_HOME/setup/db/unix/createUser.sql

表 4.5 BM 用の Oracle ユーザの設定情報

#	項目名	設定値
1	ユーザ名	BPC
2	パスワード	BPC
3	デフォルト表領域	USERS
4	テンポラリ表領域	TEMP

② BM 用のテーブルの作成

SQL*Plus 等を使用して、下記の SQL スクリプトを **BPC** ユーザで実行する。下表に従い環境に応じたスクリプトを実行することより、**BM** で使用するテーブルが作成される。

表 4.6 BM のテーブル作成 SQL スクリプト

対象環境	SQL スクリプト ファイル名
【構成 A】【構成 C】	\$BPC_HOME/setup/db/windows/createTable.sql
【構成 B】	\$BPC_HOME/setup/db/unix/createTable.sql

(3) BM 環境設定

ここでは、データベースのコネクションプールの作成、アプリケーションのデプロイを行う。

① Oracle JDBC ドライバの配置

Oracle に付属する JDBC ドライバ(classes12.jar)を\$BPC_HOME/lib へ配置する。

② BM を起動する

ディレクトリを\$BPC_HOME/bin へ移動し、以下のコマンドを実行する。

```
> startbpc
```

起動が完了すると、以下のメッセージが出力される。このメッセージが出力されることを確認後に「セットアップスクリプトの実行」を行う。

<2004/12/03 17 時 25 分 23 秒 JST> <Notice> <WebLogicServer> <BEA-000355> <スレッド "ListenThread.Default" は IP アドレス *.* のポート 7001 をリスンしています。>

③ セットアップスクリプトの実行

WebLogic が起動後、\$BPC_HOME/setup/bin ディレクトリにあるセットアップスクリプトを実行する。以下で示すように構成毎のスクリプトがあり、構成にあったスクリプトを\$BPC_HOME/bin ディレクトリへ移動後、実行する。実行時に引数の指定は必要ない。

表 4.7 セットアップスクリプト

ファイル名	内容	対象環境
setupBiz.bat	Windows + WebLogic + BizEngine の環境用バッチ	【構成 A】
setupWLI.sh	Solaris9 + WebLogic Integration の環境用バッチ	【構成 B】
setupWLI.bat	Windows + WebLogic + BizEngine の環境用バッチ	【構成 C】

次のような警告メッセージが出力されることがあるが、ビジネスモジュールのホットデプロイができないだけで、ビジネスモジュールの動作自体には実害は無い。

<2004/12/03 17 時 25 分 01 秒 JST> <Warning> <EJB> <BEA-010001> <EJB 'EventSubTableOperation' のデプロイ中に、クラス jp.go.meti.dscm.ejb.EventSubTableOperation がシステム クラスパスからロードされました。そのため、このクラスはサーバの実行中は再ロードできません。今後これを回避するには、クラスをサーバのクラスパスに入れないようにしてください。>

5. 動作確認

単一サーバでテストするセルフテスト、2 つのサーバを使用しテストする疎通テストについて説明する。

5.1 単一サーバでのセルフテスト

BM のインストールが正しく行なわれているかどうか、単一サーバ内でのデータ送受信を行い確認することができる。この単一サーバでの送受信テストを「セルフテスト」と呼ぶ。ここでは動作確認の方法としてセルフテストの実施方法を記述する。

(1) セルフテストの実行

セルフテストは以下の手順にて行なわれる。

① プロパティファイルの設定値変更

`$BPC_HOME/config/bpc.properties` の `bpc.constraint.self_party_name` プロパティの値に **self** を設定する。

```
bpc.constraint.self_party_name=self
```

② Trading Partner 情報の設定

【構成 B(WLI+Solaris)】、【構成 C(WLI+Windows)】の場合のみ行う設定手順は、「8.2 WLI でのコンフィグレーション作成手順」を参照のこと。

③ BM とイベントスケジューラを起動する

BM を起動後にイベントスケジューラを起動する。起動の方法は、「8.1 BM とイベントスケジューラの起動・終了方法」を参照のこと。

④ テスト用発注データの送信

下記 URL へアクセスし、フォームへの表 5.2 セルフテスト時のフォームへの入力値で示す値を入力し[送信]ボタンを押下する。

<http://localhost:7001/bpc/>

この操作により、`$BPC_HOME/profile/load/order/order-sample.dat` の送信が行われる。

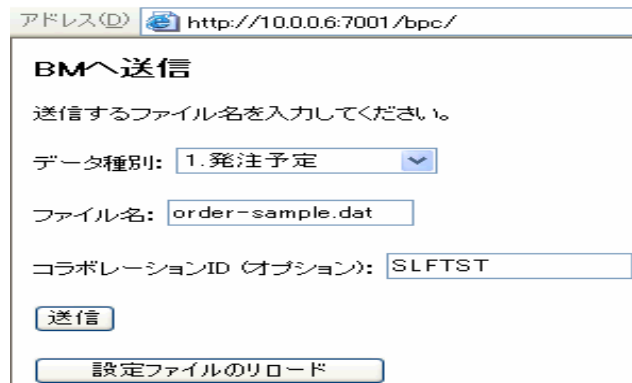


図 5.1 WLI の Oracle ユーザの作成 SQL スクリプト

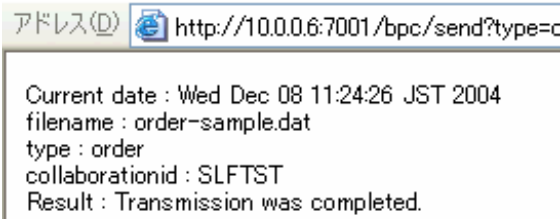
表 5.2 セルフテスト時のフォームへの入力値

入力項目	入力値
データ種別	1.発注予定
ファイル名	order-sample.dat
コラボレーション ID (オプション)	SLFTST

(2) セルフテストの確認

送信されたデータは受信され、\$BCP_HOME/profile/save/order ディレクトリに、受信データとして保存される。セルフテストの確認は、以下 4 つの項目について行う。

表 5.3 セルフテストの確認項目

<input type="checkbox"/>	<p>Web ブラウザで Result が Transmission was completed. と出力されている。</p> 
<input type="checkbox"/>	\$BPC_HOME/log/BPC.log にエラーが発生していない。
<input type="checkbox"/>	<p>\$BPC_HOME/log/BPC.log で結果コードが 0000 で終了し、[NoticeOrderEvent]の終了メッセージで出力が終了している。</p> <pre> 2004/12/03 15:26:38 [AP-5020] []業務 AP アダプタの受信処理を完了しました。 【CTX=00979b92c0a8028300baa6baed596c72,CLB=SLFTST,結果コード=0000,原因=N/A】 at jp.go.meti.dscm.adapter.sample.ReceiveOrder 2004/12/03 15:26:38 [BM-3017] []プロセス(NoticeOrderEvent)を終了します。 【CTX=00979b92c0a8028300baa6baed596c72,CLB=SLFTST】 at jp.go.meti.dscm.constraint.ordering.NoticeOrder </pre>
<input type="checkbox"/>	<p>\$BCP_HOME/profile/save/order ディレクトリに作成されたファイルと \$BCP_HOME/profile/load/order/order-sampled.dat を【構成 A】【構成 C】の場合は、FC コマンド【構成 B】の場合は、diff コマンドの実行し、相違点は検出されない。</p>

5.2 複数サーバ間の疎通テスト

ここでは、ビジネスモジュールを導入した複数のサーバ間の疎通テストを行う場合の手順について記述する。

(1) 前提

それぞれの **BM** について、インストールおよびセルフテストが完了していること。ここでの疎通テストでは **SSL** およびメッセージへのデジタル署名は使用しない。

(2) 疎通テストのシナリオ

ここでは、小売業者と卸売業者間での以下の通信を想定する。

- 1) 小売業者から発注を送信し、卸売業者が受信する。
- 2) 卸売業者から ASN を送信し、小売業者が受信する。

以降、企業を表す **ID**(以降、**partyId** と記す)として小売側は「**retailA**」、卸側は、「**wholesaleB**」とした場合を例として、テスト方法を説明する。

表 5.4 本例で使用する partyId

ロール	partyId
小売側	retailA
卸側	wholesaleB

設定のテンプレートとしては表 5, 5 のファイルを使用する。各ファイルで **partyId** は、小売側を **partyA**、卸側を **partyB** とした設定のサンプルとなっている。それぞれのファイル名およびファイル内容の「**partyA**」「**partyB**」を、それぞれ上記表 5.4 の **PartyId** に置き換えることにより設定を行う。以下、各ファイル別に、**ID** 置き換え作業の詳細を記述する。

表 5.5 テンプレートファイル

ファイル名	内容
partyA-partyB.cpa	BizEngine の BM の設定で使用する。ebXML の CPPA 2.0 の CPA 定義のテンプレートファイル。 1
partyA-partyB-collaboration.xml	コラボレーション記述 XML のテンプレートファイル。企業(拠点)間の取引を定義したファイル。
partyA-partyB-event.xml	イベント記述 XML のテンプレートファイル。企業(拠点)間の取引で行われる処理の詳細を定義したファイルで、起動時刻の設定等を行うことができる。
partyA-partyB.xml	通信プロファイルのテンプレート。2 拠点間の通信パラメタを定義したファイル。

手順 1) 自社の partyId の設定

自社の partyId を \$BPC_HOME/config/ config/bpc.properties に設定する。
変更するプロパティは、bpc.constraint.self_party_name

【retailA】	【wholesale】
retailA とする。	wholesaleB とする。

手順 2) コラボレーション記述 XML ファイルの作成

- 1) \$BPC_HOME/config/desc/partyA-partyB-collaboration.xml のコピーを retailA-wholesaleB-collaboration.xml というファイル名で作成する。
- 2) ファイル retailA-wholesaleB-collaboration.xml をエディタで開き” partyA” を” retailA” に” partyB” を” wholesaleB” に文字列置換する（それぞれ 10 箇所の置換対象がある）。

※\$BPC_HOME/config/desc ディレクトリにあらかじめ入っている他の xml ファイルは、残しておいてかまわない。

手順 3) イベント記述 XML ファイルの作成

- 1) \$BPC_HOME/config/desc/partyA-partyB-event.xml の コ ピ ー を retailA-wholesaleB-event.xml というファイル名で作成する。
- 2) ファイル retailA-wholesaleB-event.xml をエディタで開き” partyA” を” retailA” に” partyB” を” wholesaleB” に文字列置換する（それぞれ 2 箇所の置換対象がある）。

※\$BPC_HOME/config/desc ディレクトリにあらかじめ入っている他の xml ファイルは、残しておいてかまわない。

手順 4) 通信プロファイル定義の作成

- 1) \$BPC_HOME/config/desc/partyA-partyB.xml のコピーを以下の命名規約に従ったファイル名で作成する。

ファイル命名規約

- partyId の辞書順の大小で“-”（ハイフン）で 2 つの partyId でつなげる。
- {partyId の小さいほう}-{partyId の大きいほう}.xml という名前とする。

それぞれ partyId が、retailA, wholesaleB であるので、コミュニケーションプロファイル名は、retailA-wholesaleB.xml となる（partyId が Awholesale,Bretail であれば、Awholesale- Bretail.xml となる）。

- 2) 作成したファイルをエディタで開き” partyA” を” retailA” に” partyB” を” wholesaleB” に文字列置換する（それぞれ 8 箇所の置換対象がある）。

※\$BPC_HOME/config/tp ディレクトリにあらかじめ入っている他の xml ファイルは、残しておいてかまわない。

手順 5) CPA の設定

この設定は **BizEngine** と **WLI** で手順が異なる。以下に手順を示す。

■BizEngine の場合

- 1) \$BIZ_HOME/conf/cppcpa/partyA-partyB.cpa のコピーを retailA-wholesaleB.cpa というファイル名で作成する。
- 2) ファイル retailA-wholesaleB.cpa をエディタで開き” partyA” を” retailA” に” partyB” を” wholesaleB” に文字列置換する（それぞれ 5 箇所の置換対象がある）。

※\$BIZ_HOME/conf/cppcpa ディレクトリにあらかじめ入っている他の cpa ファイルは、残しておいてかまわない。

■WLI の場合

WLI コンソールから設定する

- 1) retailA の Trading Partner 情報を作成。
- 2) retailA の Binding 情報を作成。
- 3) wholesaleB の Trading Partner 情報を作成。
- 4) wholesaleB の Binding 情報を作成。
- 5) Service Profile を作成。

※詳細の設定方法は、[Appendix 8.4]を参照のこと。

手順 6) データの送受信

【retailA】側と【wholesaleB】側でのデータの送信方法とデータ受信の確認について記述する。

【retailA】側

発注データを送信し ASN データの受信を確認する。

① 発注データの送信

下記 URL へアクセスし、ファームへの表 5. 2 セルフテスト時のフォームへの入力値で示す値を入力し[送信]ボタンを押下する。

<http://localhost:7001/bpc/>

この操作により、\$BPC_HOME/profile/load/order/order-sample.dat の送信が行われる。

表 5.6 retailA 側の発注データ送信時のフォームへの入力値

入力項目	入力値
データ種別	1.発注予定
ファイル名	order-sample.dat
コラボレーション ID (オプション)	retailA_wholesaleB_ORDER

retailA 側では図 5. 7 のように“プロセス(SendOrderEvent)を終了します。”というメッセージが出力され、wholesaleB 側では、受信時に図 5. 9 のようなメッセージが出力される。

```
16:50:52.972 D [TS-1018] [ ]J-XML メッセージを送信完了しました。
【CTX=00b1a8b0c0a8030301e3f663138b8b50】 at
jp.go.meti.dscm.communication.adapter.bpc.BpcSenderAdapterImpl
16:50:53.159 I [BM-3017] [ ]プロセス(SendOrderEvent)を終了します。
【CTX=00b1a8b0c0a8030301e3f663138b8b50,CLB=retailA_wholesaleB_ORDER】 at
jp.go.meti.dscm.constraint.ordering.SendOrder
```

図 5.7 発注データ送信時のメッセージ出力

② ASN データの受信確認

受信した ASN のデータは、\$BCP_HOME/profile/save/shippingnotice ディレクトリに保存される。retailA 側では図 5. 8 のように“プロセス(NoticeShippingNoticeEvent)を終了します。”というメッセージが出力される。

```
16:54:25.030 I [AP-5020] [ ]業務 AP アダプタの受信処理を完了しました。
【CTX=00b1abd5c0a8034401e0e3ca63916461,CLB=retailA_wholesaleB_ORDER,結果コード=0000,原因
=N/A】 at jp.go.meti.dscm.adapter.sample.ReceiveShippingNotice
16:54:25.046 I [BM-3017] [ ]プロセス(NoticeShippingNoticeEvent)を終了します。
【CTX=00b1abd5c0a8034401e0e3ca63916461,CLB=retailA_wholesaleB_ORDER】 at
jp.go.meti.dscm.constraint.postshipping.NoticeShippingNotice
```

図 5.8 ASN データ受信時のメッセージ出力

【wholesaleB】側

ASN データの受信を確認し、発注データを送信する。

① 発注データの受信確認

受信した発注のデータは、\$BCP_HOME/profile/save/order ディレクトリに保存される。wholesaleB 側では図 5.9 のように”プロセス(NoticeOrderEvent)を終了します。”というメッセージが出力される。

```
18:26:49.296 I [AP-5020] [ ]業務 AP アダプタの受信処理を完了しました。
【CTX=00b20078c0a80303017100972fb7563c,CLB=retailA_wholesaleB_ORDER,結果コード=0000,原因=N/A】 at
jp.go.meti.dscm.adapter.sample.ReceiveOrder
18:26:49.316 I [BM-3017] [ ]プロセス(NoticeOrderEvent)を終了します。
【CTX=00b20078c0a80303017100972fb7563c,CLB=retailA_wholesaleB_ORDER】 at
jp.go.meti.dscm.constraint.ordering.NoticeOrder
```

図 5.9 発注データ受信時のメッセージ出力

② ASN データの送信

下記 URL へアクセスし、フォームへ表 5.10 に示す値を入力し[送信]ボタンを押下する。

<http://localhost:7001/bpc/>

この操作により、下記のデータの送信処理が行われる。

\$BPC_HOME/profile/load/shippingnotice/asn-sample.dat

表 5.11 セルフテスト時のフォームへの入力値

入力項目	入力値
データ種別	2.出荷確定(ASN)
ファイル名	asn-sample.dat
コラボレーション ID (オプション)	retailA_wholesaleB_ORDER

wholesaleB 側では図 5.11 のように“プロセス(SendShippingNoticeEvent)を終了します。”というメッセージが出力され、retailA 側では、受信時に図 5.8 のようなメッセージが出力される。

```
18:29:38.645 D [TS-1018] [ ]J-XML メッセージを送信完了しました。【CTX=00b2031ac0a8034401a48a8433f91d44】
at jp.go.meti.dscm.communication.adapter.bpc.BpcSenderAdapterImpl
18:29:38.756 I [BM-3017] [ ]プロセス(SendShippingNoticeEvent)を終了します。
【CTX=00b2031ac0a8034401a48a8433f91d44,CLB=retailA_wholesaleB_ORDER】 at
jp.go.meti.dscm.constraint.postshipping.SendShippingNotice
```

表 5.12 ASN データ送信時のメッセージ出力

6. BM クライアント型通信アダプタのインストール（オプション）

6.1 概要

ビジネスモジュールは、クライアント型通信をサポートするため、「クライアント型通信サーバ」と連携する機能を備える。ここでは、クライアント型通信サーバと連携する際に必要な作業について説明する。

クライアント型通信を利用しない場合は、本章の作業は必要ない。

クライアント型通信サーバと連携するためには、クライアント型通信サーバが稼動しているサーバに「クライアント型通信アダプタ」をインストールする必要がある。

6.2 BM クライアント型通信アダプタのプラットフォーム要件

クライアント型通信アダプタのインストール先に対するプラットフォーム要件は以下となる。

表 6.1 BM 稼動のための推奨ハードウェアスペック

#		
1	CPU	Pentium 1GHz 以上
2	メモリ	128M バイト以上 (OS 利用分除く)
3	HDD 容量	20M バイト+通信ログ格納分※1

※1 トランザクション量により異なる。BM インストールには 20M バイト必要。

表 6.2 BM クライアント型通信アダプタの動作する OS, ミドルウェア

#	OS ミドルウェア	
1	OS	Windows 2000 または Windows Server 2003
2	JDK(※1)	Java2 SDK SE 1.4.2_04
3	Oracle JDBC ドライバ (classes12.zip)	本クライアント型通信アダプタが接続するビジネスモジュール本体の利用する Oracle Database Server に付属しているもの。
4	WebLogic クライアントライブラリ (wlclient.jar)	本クライアント型通信アダプタが接続するビジネスモジュール本体の利用する Weblogic Server に付属しているもの。
5	クライアント型通信サーバ	※2

※1 Java2 SDK に必要な OS パッチが適用されていること。

※2 SFI 社 BizX サーバ、およびその稼動に必要なソフトウェア (SQL Server 等)

詳細は BizX サーバの稼動条件を参照ください。

6.3 BM クライアント型通信アダプタのインストール

(1) BM コンポーネントのインストール

- 1) 表 3.2 で示した環境変数を適切に設定する

表 6.3 インストールディレクトリを表す環境変数

#	環境変数名	設定値	本ドキュメントで想定するインストールディレクトリ ※1
1	JAVA_HOME	Java2 SDK のホームディレクトリ	C:\¥j2sdk1.4.2_04
4	BPC_HOME	BM の ホームディレクトリ(提供している zip を解凍して作成された [bpc] フォルダの場所)	C:\¥bpc

※1 Windows のシステム環境変数に設定する。

※2 パス名には空白（スペース）漢字を使用しないこと(英数字を使用する)。

- 2) ビジネスモジュールの提供ファイルのアーカイブを任意の作業用ディレクトリに展開する
- 3) 展開されたファイルに含まれる instbizx.bat を実行する。

インストールバッチを実行後、環境変数 **BPC_HOME** で指定されたディレクトリには、以下のサブディレクトリが作成される。

表 6.4 ビジネスモジュールインストール時に生成されるディレクトリ

ディレクトリ	格納されるファイル
bin	ビジネスモジュールの起動終了の起動、各種管理用ツール
config	コンフィグファイル
setup	ビジネスモジュールのインストールと環境設定に使用されるファイル
lib	ライブラリファイル
log	ログファイル(インストール直後はファイルは存在しない)

(2) BM クライアント型通信アダプタの環境設定

① Oracle JDBC ドライバの配置

Oracle に 付 属 す る JDBC ド ラ イ バ (\$ORACLE_HOME/jdbc/lib/classes12.jar)を、クライアント型通信アダプタをインストールしたサーバの\$BPC_HOME/lib へコピーする。

② WebLogic クライアントライブラリの配置

WebLogic Server 8.1SP3 に付属するクライアントライブラリ (\$BEA_HOME/server/lib/wlclientjar) を、クライアント型通信アダプタをインストールしたサーバの \$BPC_HOME/lib へコピーする。

③ クライアント型通信アダプタの環境設定

クライアント型通信アダプタをインストールしたマシンで設定を行う。

設定は、送信用のものと受信用の二つからなる。以下、表 6.5 および表 6.6 に修正必要な箇所を示す。

表 6.5 bpc.bizx.sender.properties のパラメータ

プロパティ名称	意味	備考
Bpc.bizx.out_dir	クライアント型通信サーバにデータを渡す際に使用されるディレクトリ。	クライアント型通信の設定で [クライアント>基本属性>データ受信>入力ディレクトリパス] に指定したディレクトリを設定する。
Bpc.bizx.outtemp_dir	BMが受信したデータが保存されるディレクトリ。	任意のディレクトリを指定する。
bpc.db.url	データベースへの接続URL。	BM_HOSTの部分をBM本体が動作しているのサーバのホスト名に変更する。

※パスの設定時は、ファイルのセパレータ(¥)を重ねて記述する必要がある。たとえば、bpc.bizx.out_dir に設定するディレクトリ名が、C:¥bizx¥out の場合、ファイルのセパレータ(¥)を重ねて C:¥¥bizx¥¥out と設定する。

表 6.6 bpc.bizx.receiver.properties のパラメータ

プロパティ名称	意味	備考
bpc.bizx.in_dir	クライアント型通信サーバが受信したデータが保存されるディレクトリ。BMにデータを渡す際に使用される。	クライアント型通信サーバ Clientの設定で[クライアント>基本属性>データ送信>出力ディレクトリパス]に指定したディレクトリを設定する。
bpc.bizx.in_history_dir	クライアント型通信サーバが受信したデータのバックアップが保存される。	任意のディレクトリを指定する。
bpc.db.url	データベースへの接続URL	BM_HOSTの部分をBMサーバが利用するデータベースサーバの稼働するのホスト名に変更する。
bpc.bizx.jndi.provider_url	JNDIのプロバイダURL	BM_HOSTの部分をBMサーバのホスト名に変更する。

※パスの設定時は、ファイルのセパレータ(¥)を重ねて記述する必要がある。たとえば、bpc.bizx.in_dir に設定するディレクトリ名が、C:¥bizx¥in の場合、ファイルのセパレータ(¥)を重ねて C:¥¥bizx¥¥in と設定する。

6.4 BM 本体の設定

ここでは、インストールしたクライアント型通信アダプタとの接続のために、BM のサーバ側で行う設定について説明する。

クライアント型通信を利用するコラボレーションについては、**communication-profile** 要素に **transfer-type** 属性を次のように付加する。

```
<communication-profile id="partyA" transfer-type="bizx"/>
```

6.5 クライアント型通信アダプタの開始と終了

① Sender/Receiver の起動

ディレクトリを\$BPC_HOME/bin に移動

次のコマンドを実行する

bizxSender (センダ)

bizxReceiver (レシーバ)

② Sender/Receiver の終了

センダ/レシーバが起動しているコマンドプロンプトの画面で **CTRL+C** 押下で停止する

7. BM のカスタマイズ方法

BM を利用して指定のデータフォーマットを利用して他社と業務データの交換を行うためには BM のカスタマイズが必要である。以下にカスタマイズが必要な項目について述べる。

7.1 業務アプリケーションアダプタのデプロイ

業務アプリケーションアダプタ（以下、業務 AP アダプタと記す）とは、BM 上で動作するコンポーネントで、既存業務システムと業務データの交換を行う役割を担う。特定フォーマットのデータを BM でデータ通信を行うには業務 AP アダプタを作成しこれを BM へ認識させる必要がある。これを以下の手順で行う。

(1) 業務 AP アダプタの作成

「業務 AP アダプタ開発書」に従い業務 AP アダプタを作成する。

(2) jar ファイルの作成

業務 AP アダプタの作成によりできた class ファイルをアーカイブし 1 つの jar ファイルを作成する。jar ファイルの名前は、bpc-user-adapter.jar という名前で作成する。

(3) bpc-user-adapter.jar のデプロイ

\$BPC_HOME/lib に配置する。

※bpc-user-adapter.jar に対するクラスパスの設定は、WebLogic の起動スクリプトで設定済みであるので、クラスパスの設定を行う必要はない。

7.2 通信設定のコンフィグレーション

ビジネスモジュールで各拠点間とのメッセージ交換を行うのに必要なコンフィグレーションを行う必要がある。これを以下に示す。詳細は、「環境設定書」を参照のこと。

表 7.1 コンフィグレーション作業

#	作業項目	説明
1	通信条件の登録	【構成 B】【構成 C】で必要。WLI の Integration Administration Console を使用し、拠点の情報を設定する。
		【構成 A】で必要。ebXML の CPPA 2.0 に基づく 2 拠点間の情報を定義する CPA ファイルを作成し \$BIZ_HOME/conf/cppcpa に配置する。

		BM で使用する通信プロファイルを作成し\$BPC_HOME/config/tp に配置する。
2	コラボレーション定義ファイルの作成	BM で使用するコラボレーション記述 XML、イベント記述 XML を作成し\$BPC_HOME/config/desc に配置する。 コラボレーション記述 XML ファイルは、企業（拠点）間の取引を定義したファイルで、作成したファイルで、イベント記述 XML ファイルは、企業（拠点）間の取引で行われる処理の詳細を定義したファイルで、起動時刻の設定等を行うことができる。

8. Appendix

8.1 BM とイベントスケジューラの起動・終了方法

ここでは、BM の起動・終了の方法と、イベントスケジューラの起動・終了の方法を記述する。

① BM の起動

ディレクトリを\$BPC_HOME/bin に移動

次のコマンドを実行する

```
startbpc
```

② BM の終了

ディレクトリを\$BPC_HOME/bin に移動

次のコマンドを実行する

```
stopbpc
```

③ イベントスケジューラの起動

ディレクトリを\$BPC_HOME/bin に移動

次のコマンドを実行する

```
./bpc-console scheduler_start
```

④ イベントスケジューラの終了

ディレクトリを\$BPC_HOME/bin に移動

次のコマンドを実行する

```
./bpc-console scheduler_stop
```

8.2 WLI でのコンフィグレーション作成手順

実際の画面では、説明が表示されるが、以下の操作手順中の画面には紙面上、説明を省略しているところもある。

(1) 【コンフィグレーションの作成または拡張】 画面

コンフィグレーションの作成または拡張:

コンフィグレーションを作成するか拡張するかを選択します。
この選択に基づいて、 Configuration Wizard は新規のコンフィグレーションを生成、または既存のコンフィグレーションを拡張します。

→1|新しい WebLogic コンフィグレーションの作成

2|既存の WebLogic コンフィグレーションの拡張

選択するインデックス番号を入力してください または [Exit][Next]>

入力を行わず、リターンキーを押下する。

(2) 【ドメイン テンプレートを選択】 画面

ドメイン テンプレートを選択:

リストからテンプレートを選択するか、別のテンプレート ディレクトリを選択してください。

* [C:\bea813\weblogic81\common\templates\domains]

1|WebLogic Integration Domain FOR BM 8.1.3.0
| Ministry of Economy, Trade and Industry

2|Basic WebLogic Portal Domain 8.1.3.0

3|Basic WebLogic Integration Domain 8.1.3.0

4|Basic WebLogic Workshop Domain 8.1.3.0

選択するインデックス番号を入力してください または [Down][Exit][Previous][Next]> 1

「WebLogic Integration Domain FOR BM 8.1.3.0」を選択する。画面に表示されていない場合は、「WebLogic Integration Domain FOR BM 8.1.3.0」が表示されるまで、**Down** を入力し次画面を表示し、選択する。この例では、**1** 画面目に表示されているので、**1** を入力しリターンキーを押下する。

(3) [コンフィグレーション オプションの選択] 画面

```
コンフィグレーション オプションの選択：
-----

    *ウィザードをエクスプレス モードで実行しますか?

->1|はい

    2|いいえ

選択するインデックス番号を入力してください または [Exit][Previous][Next]>
```

入力を行わず、リターンキーを押下する。

(4) [管理ユーザ名とパスワードのコンフィグレーション] 画面

```
管理ユーザ名とパスワードのコンフィグレーション：
-----

管理ロールを自動的に割り当てられたユーザを作成します。 このユーザは、開発モード サーバを起動
するためのデフォルトの管理者です。

|          名前          |          値          |
|-----|-----|
1|      *User name:      |          system      |
2|      *User password:  |          **********  |
3| *Confirm user password: |          **********  |
4|      Description:     | This user is the default administrator. |

オプションの選択：
  1 - 変更 "User name"
  2 - 変更 "User password"
  3 - 変更 "Confirm user password"
  4 - 変更 "Description"

選択するオプション番号を入力してください または [Exit][Previous][Next]>
```

入力を行わず、リターンキーを押下する。

(5) [ドメイン モードのコンフィグレーション] 画面

ドメイン モードのコンフィグレーション:

このドメインの開発モードまたはプロダクション モードを有効化します。

->1|開発モード

2|プロダクション モード

選択するインデックス番号を入力してください または [Exit][Previous][Next]> 2

2を入力しリターンキーを押下する。

(6) [Java SDK の選択] 画面

Java SDK の選択:

->1|JRockit SDK 1.4.2_04-8130 @ C:\bea813\jrockit81sp3_142_04

2|Sun SDK 1.4.2_04 @ C:\bea813\jdk142_04

3|他の Java SDK

選択するインデックス番号を入力してください または [Exit][Previous][Next]> 2

Sun の SDK を選択する。この例では 2 を入力しリターンキーを押下する。

※Solaris では、下記の内容は表示されない。

「JRockit SDK 1.4.2_04-8130 @ C:\bea813\jrockit81sp3_142_04」

(7) [このコンフィグレーションの対象ドメイン ディレクトリを選択] 画面

このコンフィグレーションの対象ドメイン ディレクトリを選択:

”対象の場所” = [C:\bea813\user_projects\domains]

新規入力 対象の場所 または [Exit][Previous][Next]>

入力を行わず、リターンキーを押下する。

(8) 【このコンフィグレーションのアプリケーション ディレクトリを選択】 画面

このコンフィグレーションのアプリケーション ディレクトリを選択します。:

"対象の場所" = [C:\bea813\user_projects\applications]

新規入力 対象の場所 または [Exit][Previous][Next]> Next

入力を行わず、リターンキーを押下する。

(9) 【ドメイン情報の編集】 画面

ドメイン情報の編集:

名前	値
1 | *Name: | integration |

以下の値を入力します "Name" または [Exit][Previous][Next]>

入力を行わず、リターンキーを押下する。

(10) 【終了】 画面

ドメインを作成しています...

0% 25% 50% 75% 100%
[-----|-----|-----|-----]
[*****]

**** ドメインの作成が完了しました。 ****

8.3 WLI でのセルフテスト実施の為の作成方法

Web ブラウザを使用して、以下の URL にアクセスすると WLI の管理画面が表示される。

<http://localhost:7001/wliconsole>

※別マシンから設定を行う場合は **localhost** の部分は、適切なアドレスを設定すること。

操作方法や項目の詳細は以下の URL を参考にすること。ここでは設定する値について言及する。

<http://edocs.beasys.co.jp/e-docs/wli/docs81/manage/tpm.html>

(1) Trading Partner Profile の作成

self_from と **self_to** という Business ID の Profile を作成する。

Home 画面から Trading Partner Management を選択、Profile Management を選択する。

画面左上の Partner Profiles ペインの Create New をクリックする。「Add Trading Partner Profile」画面が表示されるので、以下の内容を入力し、[Add profile]ボタンを押下する。

*	Name	self_from	self_to
	Description		
*	Business ID	self_form	self_to
*	Business ID Type	text	Text
	Default Trading Partner		
*	Type	LOCAL	LOCAL
*	Status	ENABLED	ENABLED
	Email		
	Address		
	Phone		
	Fax		
	WLS User Name		

※*印のついた項目を入力する。*印のついていない項目は入力不要。

(2) Binding の作成

(1)の作成後の「**View and Edit Trading Partner Profile**」画面から[Add binding]ボタンを押下する。

バインディングのタイプの選択(Choose type of binding)画面が表示されるので、Ebxml 2.0 をチェックし[Create Binding] ボタンを押下する。「Add Binding (Step 2 of 2)」画面が表示されるので、以下の内容を入力し、[Add binding]ボタンを押下する。

		【self_from】	【self_to】
	Name		
*	Default Binding	チェックする	チェックする
*	Transport Protocol	HTTP	HTTP
*	Transport Protocol Version	1.1	1.1
	EndPoint		
	Timeout		
*	Delivery Semantics	ONCEANDONLYONCE	ONCEANDONLYONCE
	Retry Count		
	Retry Interval		
	Persist Duration		
	Signature Certificate		
	Signature Required		
	Signature Receipt Required		

※*印のついた項目を入力する。*印のついていない項目は入力不要。

(3) Service の作成

Home 画面から **Trading Partner Management** を選択、**Service Management** を選択する。「**View and Edit Services**」画面が表示される。

*	Name	/ReceiveProject/processes/MessageReceiver.jp	processes. ebxmlCtrlTest
*	Type	Process	Service Control
*	Business Protocol	EBXML	EBXML
	Description		

※*印のついた項目を入力する。*印のついていない項目は入力不要。

(4) Service Profile の作成

(3)で作成した 2 つのサービス、それぞれに対し、**Service Profile** を作成する。

① ReceiveProject/processes/MessageReceiver.jp

サービス名 **ReceiveProject/processes/MessageReceiver.jp** をクリック、「**Service Details**」画面が表示される。次に[**Add Service Profile**]ボタンを押下する。「**Add Service Profile**」画面で以下の内容を入力し、[**SuBMit**]ボタンを押下する。

Service Profile Id	self_from-self_to-cpa	
Status	ENABLED	
Message Tracking Level	NONE	
Name	LOCAL	REMOTE
	self_from	self_to

「**Configure Authentication**」画面が表示されるので、[**No**]ボタンを押下する。

② processes.EbxmlCtrl への Service Profile の作成

「**View and Edit Services**」画面へ戻り、サービス名 **processes.EbxmlCtrl** をクリック、「**Service Details**」画面が表示される。次に[**Add Service Profile**]ボタンを押下する。「**Add Service Profile**」画面で以下の内容を入力し、[**SuBMit**]ボタンを押下する。

Service Profile Id	self_from-self_to-cpa	
Status	ENABLED	
Message Tracking Level	NONE	
Name	LOCAL	REMOTE
	self_from	self_to

「**Configure Authentication**」画面が表示されるので、[No]ボタンを押下する。

○送受信ログの出力先の設定

Home 画面から **Trading Partner Management** を選択、**Configuration** を選択する。「**General Configuration**」画面が表示される。以下の内容を設定する。

Message Tracking Level	NONE
Mode	Test
Directory	ログを出力するディレクトリ
Trace Raw Messages	Yes

8.4 WLI での CPA の作成方法

Web ブラウザを使用して、以下の URL にアクセスすると WLI の管理画面が表示される。

<http://localhost:7001/wliconsole>

※別マシンから設定を行う場合は **localhost** の部分は、適切なアドレスを設定すること。操作方法や項目の詳細は以下の URL を参考にすること。ここでは設定する値について言及する。

<http://edocs.beasys.co.jp/e-docs/wli/docs81/manage/tpm.html>

(1) Trading Partner Profile の作成

Home 画面から Trading Partner Management を選択、Profile Management を選択する。画面左上の Patner Profiles ペインの Create New をクリックする。「Add Trading Partner Profile」画面が表示されるので、以下の内容を入力し、[Add profile]ボタンを押下する。

※1		【retailA】	【wholesaleB】
*	Name	retailA	wholesaleB
	Description		
*	Business ID	retailA	wholesaleB
*	Business ID Type	Text	text
	Default Trading Partner		
*	Type	※2	※2
*	Status	ENABLED	ENABLED
	Email		
	Address		
	Phone		
	Fax		
	WLS User Name		

※1 *印のついた項目を入力する。*印のついていない項目は入力不要。

※2 【retailA】側での設定と 【wholesaleB】側での設定は設定内容が異なる（以下のように設定する）。

Type の 【retailA】 側での設定			
		【retailA】	【wholesaleB】
*	Type	LOCAL	REMOTE
Type の 【wholesaleB】 側での設定			
		【retailA】	【wholesaleB】
*	Type	REMOTE	LOCAL

(2) Binding の作成

(1)の作成後の「**View and Edit Trading Partner Profile**」画面から[Add binding]ボタンを押下する。

バインディングのタイプの選択(Choose type of binding)画面が表示されるので、**Ebxml 2.0** をチェックし[Create Binding] ボタンを押下する。「**Add Binding (Step 2 of 2)**」画面が表示されるので、以下の内容を入力し、[Add binding]ボタンを押下する。

※1		【retailA】	【wholesaleB】
	Name	デフォルト値を使用	デフォルト値を使用
*	Default Binding	チェックする	チェックする
*	Transport Protocol	HTTP	HTTP
*	Transport Protocol Version	1.1	1.1
*	EndPoint	http://retailA:7001/axis/services/EbXMLMessageHandler	http://wholesaleB:7001/axis/services/EbXMLMessageHandler
	Timeout	デフォルト値を使用	デフォルト値を使用
*	Delivery Semantics	ONCEANDONLYONCE	ONCEANDONLYONCE
*	Retry Count	3	3
*	Retry Interval	1 hour	1 hour
*	Persist Duration	2 hour	2 hour
	Signature Certificate	NONE	NONE
	Signature Required	未チェック	未チェック
	Signature Receipt Required	未チェック	未チェック

※1 *印のついた項目を入力する。 *印のついていない項目は入力不要。

(3) Service Profile の作成

Home 画面から **Trading Partner Management** を選択、**Service Management** を選択する。「**View and Edit Services**」画面が表示される。

以下の 2 つのサービスが登録されており、それぞれに対し、**Service Profile** を作成する。

ReceiveProject/processes/MessageReceiver.jpdl
processes.EbxmlCtrl

① ReceiveProject/processes/MessageReceiver.jpdl への Service Profile の作成

サービス名 **ReceiveProject/processes/MessageReceiver.jpdl** をクリック、「**Service Details**」画面が表示される。次に[Add Service Profile]ボタンを押

下する。「Add Service Profile」画面で以下の内容を入力し、[SuBMit]ボタンを押下する。

【retailA】側での設定と【wholesaleB】側での設定は設定内容が異なる（以下のように設定する）。

【retailA】での設定

Service Profile Id	retailA- wholesaleB-cpa	
Status	ENABLED	
Message Tracking Level	ALL	
Name	LOCAL	REMOTE
	retailA	wholesaleB

【wholesaleB】での設定

Service Profile Id	retailA- wholesaleB-cpa	
Status	ENABLED	
Message Tracking Level	ALL	
Name	LOCAL	REMOTE
	wholesaleB	retailA

「Configure Authentication」画面が表示されるので、[No]ボタンを押下する。

② processes.EbxmlCtrl への Service Profile の作成

「View and Edit Services」画面へ戻り、サービス名 **processes.EbxmlCtrl** をクリック、「Service Details」画面が表示される。次に[Add Service Profile] ボタンを押下する。「Add Service Profile」画面で以下の内容を入力し、[SuBMit]ボタンを押下する。

【retailA】側での設定と【wholesaleB】側での設定は設定内容が異なる（以下のように設定する）。

【retailA】での設定

Service Profile Id	retailA- wholesaleB-cpa	
Status	ENABLED	
Message Tracking Level	ALL	
Name	LOCAL	REMOTE
	retailA	wholesaleB

【wholesaleB】での設定

Service Profile Id	retailA- wholesaleB-cpa	
Status	ENABLED	
Message Tracking Level	ALL	
Name	LOCAL	REMOTE
	wholesaleB	retailA

「Configure Authentication」画面が表示されるので、[No]ボタンを押下する。

9. ライセンス

本製品に関する利用許諾条件に関しては、本体に添付のソフトウェア使用許諾契約書を参照してください。

本製品が使用しているソフトウェアのライセンスに関して

"This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>)".

"This product includes software developed by The ExoLab (<http://www.exolab.org/>)".

"This product includes software developed by James House (<http://http://www.quartzscheduler.org/>)".

本製品で利用しているその他のソフトウェアのライセンスに関してはインストール媒体の **license¥Readme.txt** を参照してください。