

REST を利用した OpenTP1 連携サンプル 説明資料

2022. **3**
March

このガイドには、次の表に示す変更履歴が含まれています。

作成年月	変更内容	変更区分(*1)
2022年3月	初版	—

なお、単なる誤字・脱字などは、お断りなく訂正しました。

注 *1

変更区分 C：案内を変更（変更又は削除）します。既存のユーザは、使い方を変更する必要があります。

変更区分 A：既存のユーザには影響ありません。新しい案内を適用する場合だけ、使い方を変更する必要があります。

変更区分 S：案内の変更はありません。説明の追加・変更があります。

はじめに

このサンプルは、あらかじめ用意したビジネスプロセスの利用を通じて、システムの構築をイメージしやすくすることを目的としています。

このドキュメントでは、サンプルの説明とサンプルの動作方法について説明します。

■対象とする製品

このサンプルは次の製品で利用することを想定しています。

- ・ P-2943-7TB4 uCosminexus Service Architect 11-10 以降
- また、通信先に使用する TP1/Server Base は次の製品を想定しています。
- ・ P-2964-2214 uCosminexus TP1/Server Base(64) 07-06

■対象 OS

このサンプルは、対象製品が動作する次のバージョンの OS で実施することを想定しています。

- ・ Windows 10 Pro (64 ビット版)
- ・ Windows 10 Enterprise (64 ビット版)
- ・ Windows Server 2016 Standard
- ・ Windows Server 2016 Datacenter
- ・ Windows Server 2019 Standard
- ・ Windows Server 2019 Datacenter

■商標類

- ・ Microsoft、Windows、および Windows Server は、マイクロソフト 企業グループの商標です。
- ・ その他記載の会社名、製品名は、それぞれの会社の商標もしくは登録商標です。

■このガイドをご使用いただく際の注意点

このガイドに記載されている仕様は、製品の改良により予告なく変更されることがあります。

■発行元

株式会社日立製作所 サービス&プラットフォームビジネスユニット
サービスプラットフォーム事業本部

All Rights Reserved. Copyright (C) 2022, Hitachi, Ltd.

目次

1 ファイル構成	1
1.1 ファイル構成	2
2 提供サンプル	3
2.1 サンプルで作成できるもの	4
2.1.1 処理概要	4
2.1.2 HTTP 受付	5
2.1.3 TP1 アダプタ	5
2.2 利用するコンポーネントの説明	6
2.2.1 処理の流れ	6
2.2.2 HTTP 受付	7
2.2.3 TP1 アダプタ	7
2.2.4 ビジネスプロセス	8
2.3 サンプルの実行手順	11
2.3.1 HCSC コンポーネントの作成	12
2.3.2 定義ファイルの配置	12
2.3.3 HCSC サーバランタイム定義の変更	13
2.3.4 リポジトリの配備・開始	13
2.3.5 TP1 アダプタからアクセスするサービスの準備	13
2.3.6 リクエストの実行	13
2.3.7 HCSC コンポーネントの停止・削除	14
2.3.8 HCSC サーバの停止	14
2.3.9 ビジネスプロセスのデバッグ機能を利用した確認	15
3 OpenTP1 サーバの設定	16
3.1 OpenTP1 のサービスの準備	17
4 トラブルシュート	22
4.1 トラブルシュート	23
付録 A. 変更した定義ファイルのマニュアル記載箇所	24

1 ファイル構成

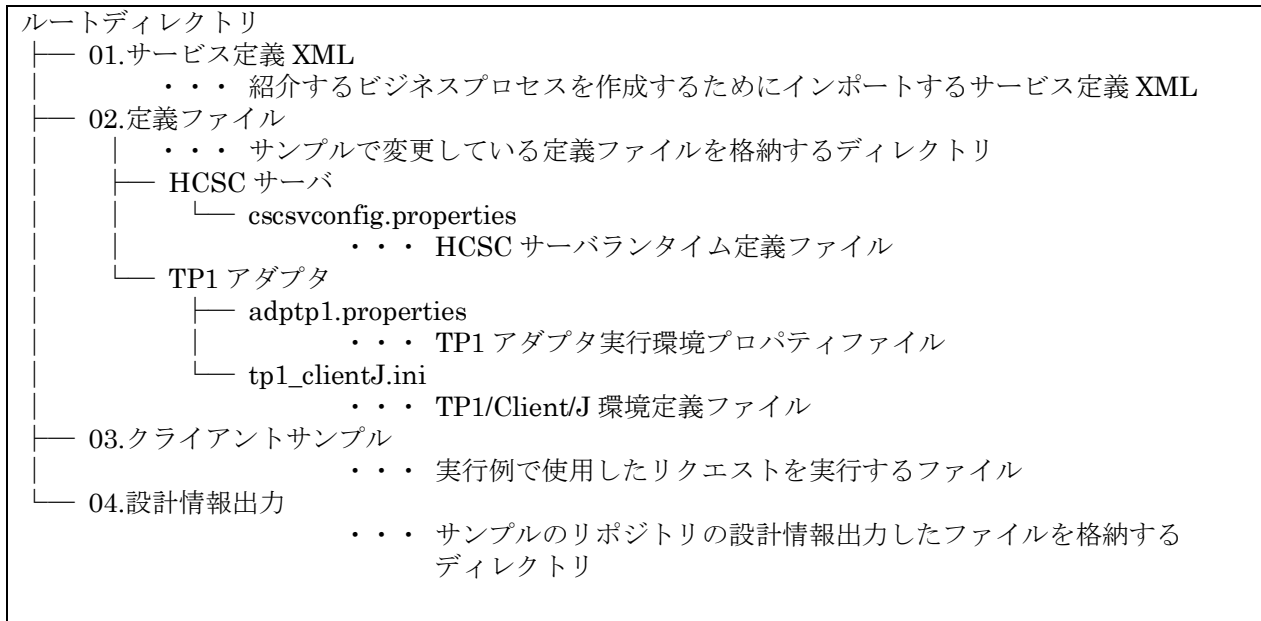
この章では、サンプルで提供するファイルの構成について説明します。

本章の構成

1.1 ファイル構成

1.1 ファイル構成

サンプルで提供するファイル構成を次に示します。



2 提供サンプル

この章では、提供するサンプルについて説明します。

本章の構成

- 2.1 サンプルで作成できるもの
- 2.2 利用するコンポーネントの説明
- 2.3 サンプルの実行手順

2.1 サンプルで作成できるもの

このサンプルでは、TP1/Server Base を利用しているレガシーシステムに対して、REST 通信を行うクライアントからアクセスする一連の HCSC コンポーネントを作成します。

2.1.1 処理概要

サンプルで作成できるシステムの概要を説明します。

このサンプルは、バイナリ電文のやりとりだけができる OpenTP1 がインストールされたサーバと、JSON での通信を行う REST クライアントの間の通信を仲介します。

通常、REST クライアントからは、バイナリ電文を送信することはできません。また、TP1/Server Base はバイナリ電文を使用したリクエスト送受信がメインであるため、REST クライアントとは通信できません。

そこで、HTTP 受付を利用して REST クライアントとの通信を行います。

また、ビジネスプロセスを使用して HTTP 受付からのリクエストをバイナリ形式の電文に変換し、TP1 アダプタを利用して TP1/Server Base とバイナリ電文の通信を行います。

サンプルでは、次の図のように HCSC コンポーネント(HTTP 受付、ビジネスプロセス、TP1 アダプタ)を呼び出して処理していきます。

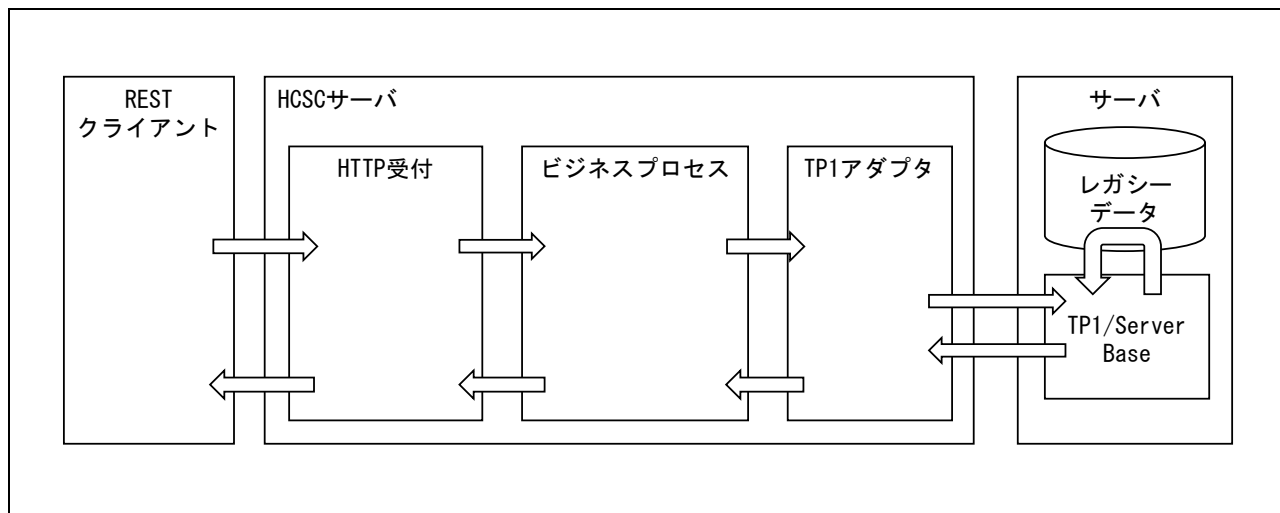


図 2-1 処理概要

2.1.2 HTTP 受付

HTTP 受付を利用することで、HTTP クライアントから HTTP 通信を使用してビジネスプロセスを呼び出すことができます。

HTTP 受付の機能については、次のマニュアルを参照してください。

「Cosminexus V11 BPM/ESB 基盤 サービスプラットフォーム 解説」
「2.13 HTTP 通信を使用したクライアントとの接続」

2.1.3 TP1 アダプタ

TP1 アダプタは、RPC 通信を使用して OpenTP1 のシステムを呼び出すことができます。

TP1 アダプタの機能については、次のマニュアルを参照してください。

「Cosminexus V11 BPM/ESB 基盤 サービスプラットフォーム 解説」
「2.7.3 HCSC サーバから TP1 アダプタを使った TP1 サービス部品の呼び出し」

2.2 利用するコンポーネントの説明

提供するサンプルで使用するコンポーネントの詳細を説明します。

2.2.1 処理の流れ

サンプルの処理の流れを、次の図に示します。図中の数字は図の下に示す処理の説明と対応しています。

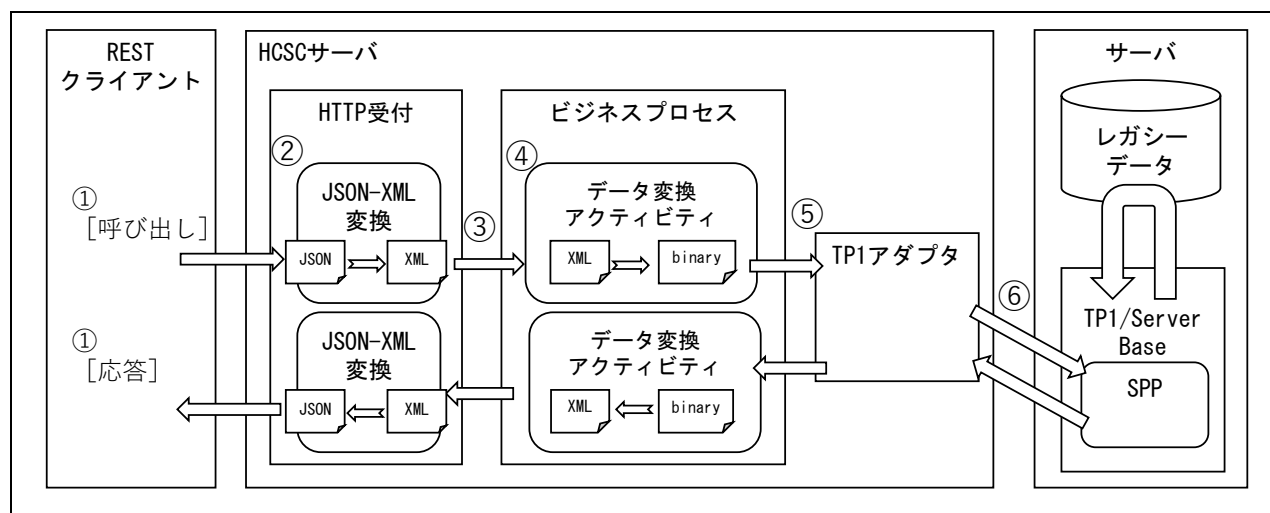


図 2-2 サンプルの処理概要

処理の流れは次のとおりです。

① クライアントから「HTTP 受付」の呼び出し・応答

[呼び出し]

クライアントからは、ボディ電文に JSON 文を設定し HTTP 受付を呼び出します。

クライアントは、HTTP メソッドの POST を指定します。

[応答]

Open TP1 からの応答データを JSON 形式でクライアントに応答します。

② JSON-XML 変換機能

HTTP 受付で利用できる JSON-XML 変換機能は、HTTP 受付で JSON 文の送受信をするために使用します。

ビジネスプロセスでデータ変換アクティビティを利用する場合、JSON 文はそのまま使用できないため、JSON-XML 変換機能を使用する必要があります。

③ 「HTTP 受付」から「ビジネスプロセス」の呼び出し

HTTP 受付は呼び出しサービス選択機能を利用して作成しています。

呼び出しサービス選択機能は、HTTP 受付サービス選択定義ファイルを指定することで 1 つの受付から複数のビジネスプロセスの呼び分けができる機能です。

④ データ変換アクティビティ

ビジネスプロセスに定義したデータ変換アクティビティでは、要求電文用のデータ変換アクティビティと、応答電文用のデータ変換アクティビティを定義します。

- 要求電文用のデータ変換アクティビティ

XML 形式の要求電文を TP1 アダプタで送信するためのバイナリ電文に変換します。

- 応答電文用のデータ変換アクティビティ

TP1 アダプタから受け取ったバイナリ電文を XML 電文に変換します。

フォルトが発生した場合は、フォルト内容を応答電文に設定します。

⑤ 「ビジネスプロセス」から「TP1 アダプタ」の呼び出し

ビジネスプロセスのサービス呼出アクティビティを定義し、TP1 アダプタを呼び出します。

TP1 アダプタの応答は、ビジネスプロセスの分岐アクティビティで正常・フォルトの判断を行い、正常終了時の電文、フォルト終了時の電文作成を行います。

⑥ 「TP1 アダプタ」から「サーバ (OpenTP1)」の呼び出し

TP1 アダプタから、OpenTP1 に定義したサービス提供プログラム (SPP) を呼び出します。

サーバから例外が返ってきた場合は、サービスアダプタの「システム例外をフォルトに変換する」機能を使用して、フォルトを送信します。

2.2.2 HTTP 受付

HTTP クライアントから指定された HTTP メソッドでビジネスプロセスを呼び分けられるように、HTTP 受付は呼び出しサービス選択機能を使用します。

また、HTTP クライアントから送信される JSON 電文をビジネスプロセスで利用できる XML に変換するため、JSON-XML 変換機能を使用します。

利用している JSON-XML 変換機能の機能の詳細については、次のマニュアルを参照してください。

「Cosminexus V11 BPM/ESB 基盤 サービスプラットフォーム 解説」

「2.13.10 HTTP 受付の JSON-XML 変換機能」

2.2.3 TP1 アダプタ

TP1 アダプタから、作成した TP1 サーバに接続します。

TP1 サーバから例外が返ってきた場合は、ビジネスプロセスにフォルトを応答します。

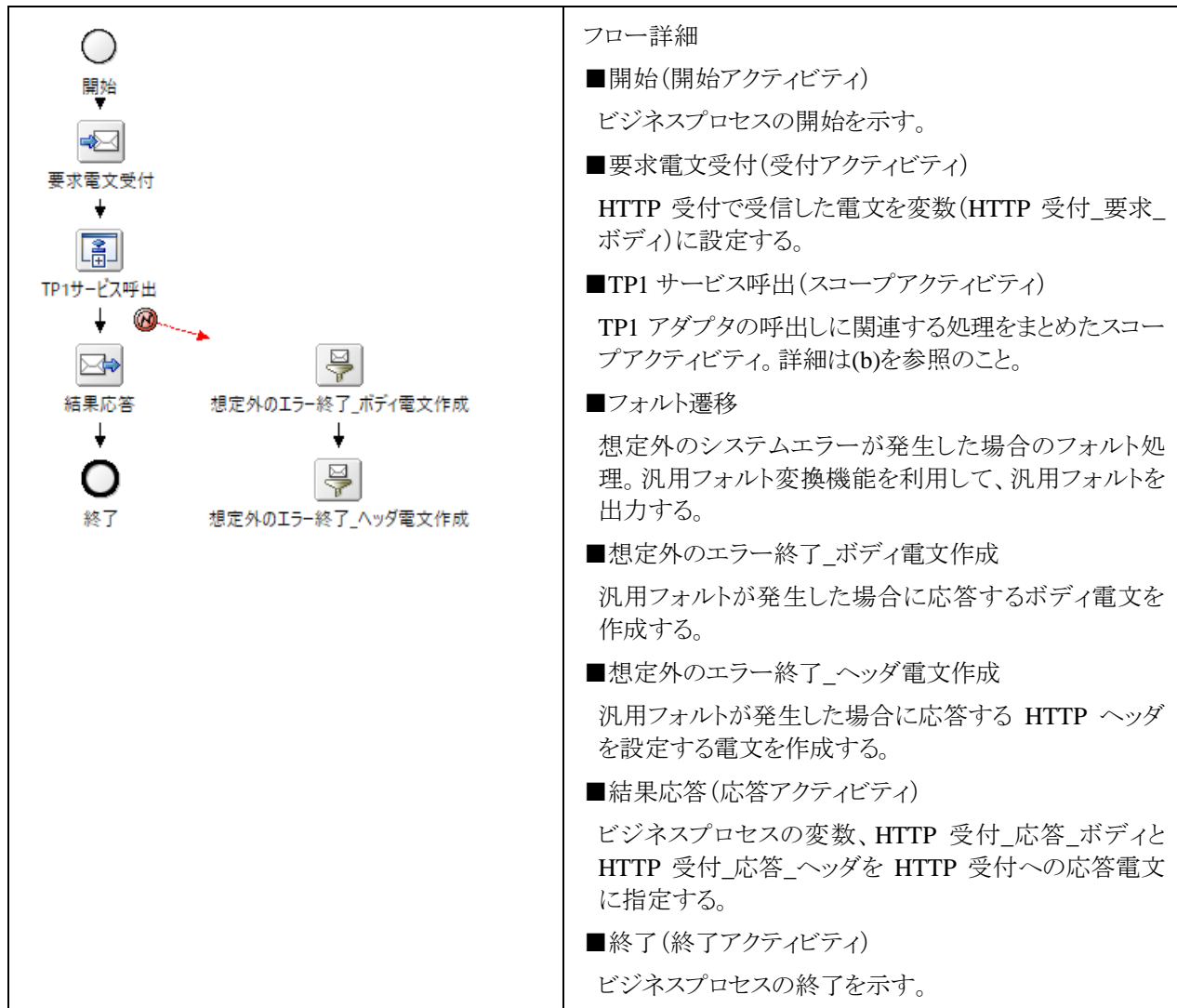
サービスアダプタでシステム例外が発生した場合は、「システム例外をフォルトに変換する」機能を使用して、ビジネスプロセスにフォルトを応答します。

2.2.4 ビジネスプロセス

ビジネスプロセスの定義内容を説明します。

(a) メインフロー

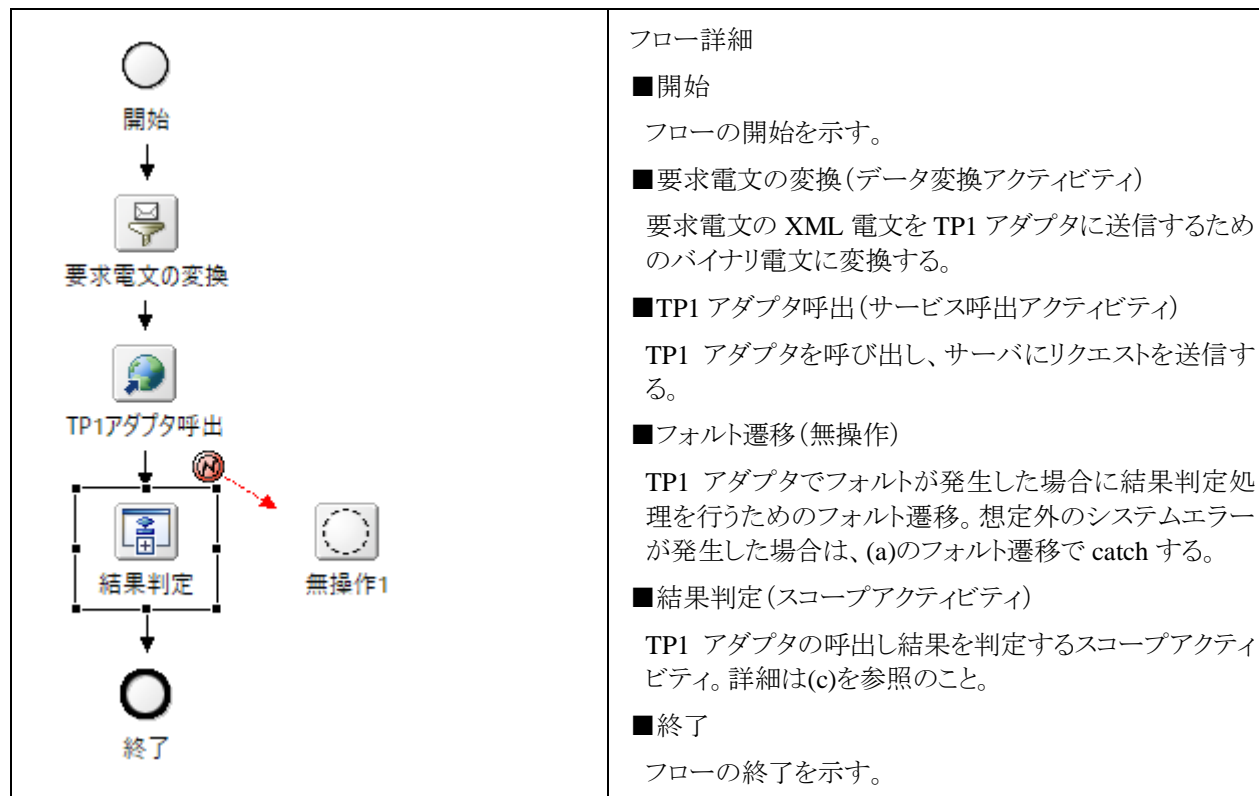
ビジネスプロセスが呼ばれた際に最初に呼ばれる部分です。処理単位でスコープアクティビティを使用しています。



(b) 「TP1 サービス呼出」スコープアクティビティの内容

TP1 アダプタを呼び出すスコープアクティビティの内容を次に示します。

「TP1 サービス呼出」スコープアクティビティでは、要求電文の変換を行い、TP1 アダプタの要求電文を作成します。

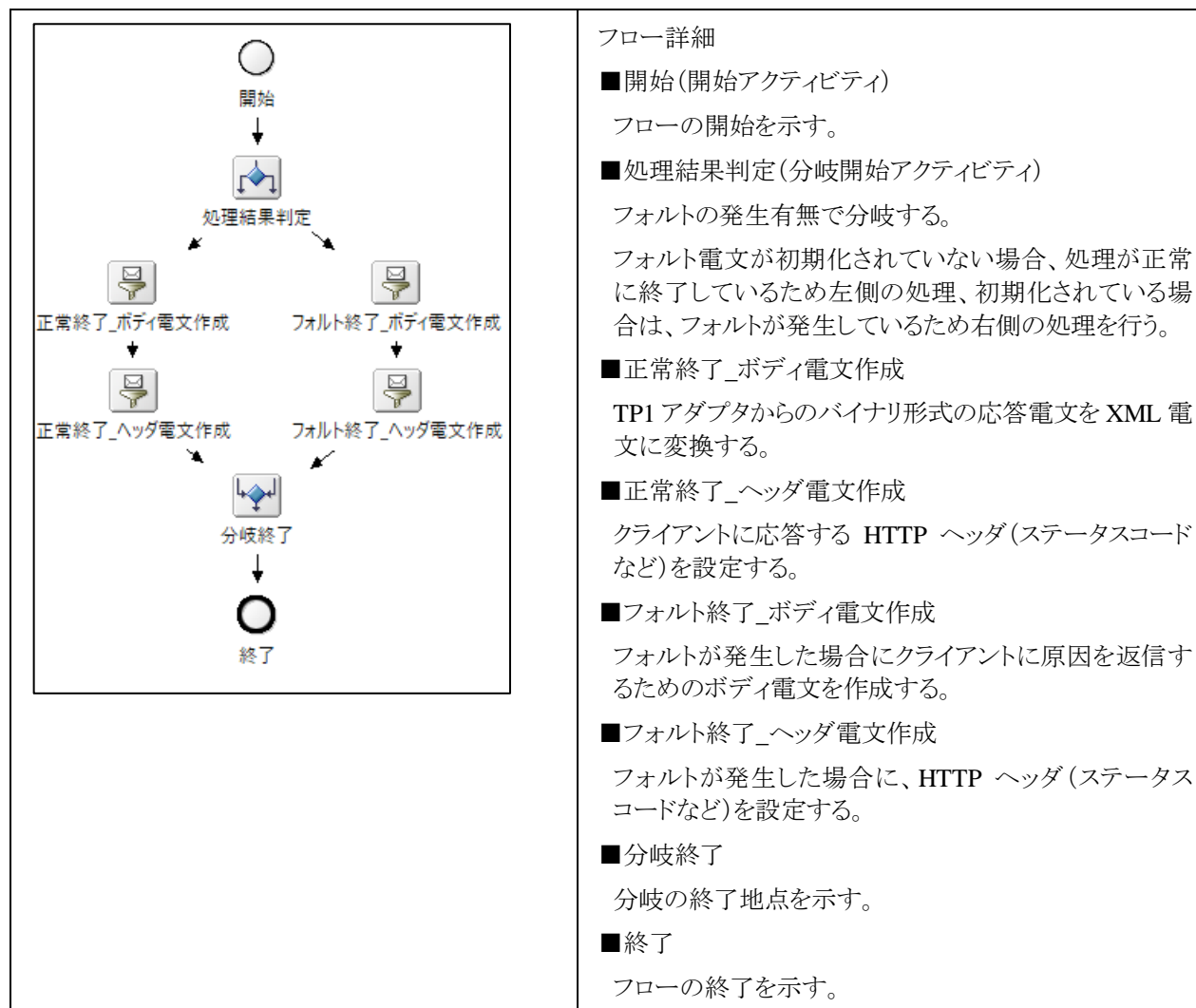


(c) 「結果判定」スコープアクティビティの内容

結果を判定するスコープアクティビティの内容を次に示します。

処理の判定は、フォルト電文が初期化されているかで判断します。

HTTP の応答コードはヘッダ電文で設定するため、ボディ電文用とヘッダ電文用のデータ変換アクティビティを用意します。



2.3 サンプルの実行手順

このサンプルを実行するために必要な手順を示します。

実行に必要な開発環境やテスト環境の作成手順は、次のマニュアルを参照してください。

「Cosminexus V11 BPM/ESB 基盤 サービスプラットフォーム ファーストステップガイド」
「3. サンプルプログラムの開発・テスト環境を構築する」

ただし、HCSC 簡易セットアップの画面の指定が上記のマニュアルとは異なります。

HCSC 簡易セットアップを利用して HCSC サーバを作成しますが、サンプルでは次の設定で構築した HCSC サーバを前提としています。

- DB:なし
- RM:なし
- SOAP モード:SOAP1.1/1.2 併用モード
- J2EE サーバ:推奨モード

マニュアルで説明している HCSC 簡易セットアップの指定と異なる個所を、次に説明します。

表 2-1HCSC 簡易セットアップ画面の設定の違い

#	マニュアルの記載	このサンプルで利用する設定
1	[DB あり/RM なしモデル] ラジオボタンを選択します。	[DB/RM なしモデル] のラジオボタンを選択します。
2	[V9 互換モード] ラジオボタンを選択します。	[推奨モード] ラジオボタンを選択します。

HCSC 簡易セットアップの画面では、次の図に示す赤枠の個所を指定してセットアップします。

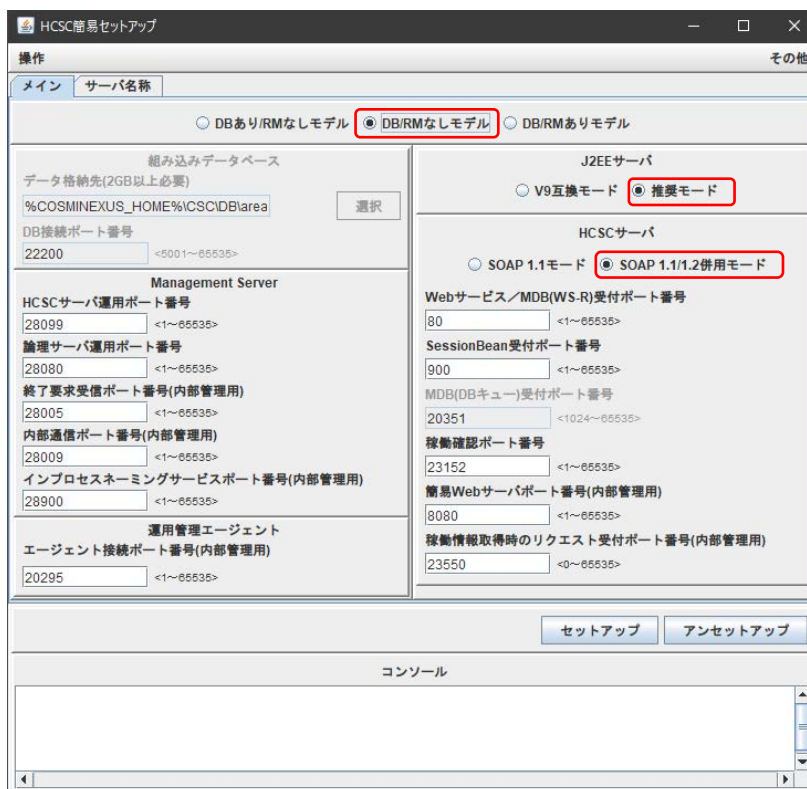


図 2-3HCSC 簡易セットアップ画面の指定箇所

2.3.1 HCSC コンポーネントの作成

開発環境のサービス定義のインポート機能を利用して、サンプルの HCSC コンポーネントを作成します。

サービス定義のインポートを選択すると、「ディレクトリの選択」ダイアログが表示されますので、「1.1 ファイル構成」で示した「01.サービス定義 XML」のディレクトリを指定してください。

サービス定義のインポート機能の詳細については、次のマニュアルを参照してください。

「Cosminexus V11 BPM/ESB 基盤 サービスプラットフォーム 開発ガイド 基本開発編」

「6.4 サービス定義のインポート」

2.3.2 定義ファイルの配置

TP1 アダプタの動作に必要な定義ファイルを、「1.1 ファイル構成」で示した「02.定義ファイル」の「TP1 アダプタ」のディレクトリにある 2 つのファイルをコピーします。

コピー先のディレクトリを次に示します。

<Cosminexus インストールディレクトリ>%CSC%\custom-adapter\TP1\config

注 Cosminexus のインストールディレクトリを変更している場合は、プロパティファイル「adtp1.properties」の

「tp1client.config」に記載しているファイルパスを変更してください。

2.3.3 HCSC サーバランタイム定義の変更

アクティビティで発生したシステム例外をフォルトに変換するため、「1.1 ファイル構成」で示した「02.定義ファイル」の「HCSC サーバ」ディレクトリのファイルを利用して定義を変更します。HCSC サーバランタイム定義を変更するには cscsvconfig コマンドを利用します。

次に示すコマンドを実行して、HCSC サーバランタイム定義を変更します。

```
cscsvconfig -csc HCSC -operation set -propfile cscsvconfig.properties
```

HCSC サーバランタイム定義の変更を反映するには、HCSC サーバの再起動が必要です。

HCSC サーバの起動・停止コマンドを利用して HCSC サーバランタイム定義を変更後に HCSC サーバを再起動してください。

- HCSC サーバの停止コマンド

```
cscsvstop -csc HCSC
```

- HCSC サーバの起動コマンド

```
cscsvstart -csc HCSC
```

2.3.4 リポジトリの配備・開始

開発環境の機能を利用して、リポジトリの配備と起動を一括で実行します。

詳細な手順については、次のマニュアルを参照してください。

```
「Cosminexus V11 BPM/ESB 基盤 サービスプラットフォーム 開発ガイド 基本開発編」  
「8.5 HCSC コンポーネントを HCSC サーバに配備して開始する処理の一括実行」
```

2.3.5 TP1 アダプタからアクセスするサービスの準備

TP1 アダプタからアクセスする OpenTP1 サーバと、サーバで起動するサービスの準備をします。

詳細な手順については、「3. OpenTP1 サーバの設定」を参照してください。

2.3.6 リクエストの実行

「1.1 ファイル構成」で示した「03.クライアントサンプル」に格納している index.html をブラウザで表示します。

「リクエスト実行」ボタンをクリックして、リクエストを送信します。

初期状態では、正常終了するリクエスト電文が設定されています。

変更しないで「リクエスト実行」ボタンをクリックするとリクエストが正常終了する動作を確認できます。

また、「再読み込み」ボタンをクリックすると初期状態に戻すことができます。

【実行結果例】

クライアントのサンプルでリクエストを実行した応答結果を示します。

- 初期設定のリクエスト電文
リクエストを起動した状態で、変更しないで「リクエスト実行」ボタンをクリックすると、次に示した内容で HTTP 受付にアクセスします。

HTTP メソッド: POST

ボディ電文: {"name": "Saitoh", "sale": 1234}

- 正常終了時の応答

上記に示したリクエストを実行して正常終了した場合は、次に示す応答が返ってきます。

HTTP ヘッダ

ステータスコード: 200

Content-type: application/json;charset=UTF-8

ボディ電文: {"nomal": {"name": "Saitoh", "sex": "M", "age": "24"}}

- リクエストがフォルト終了する場合の応答

接続先サービスが停止している場合や、OpenTP1 のサンプルに存在しないデータを取得しようとした場合、正常時とは異なる応答を返します。

例) 接続先のサービスが停止している場合の応答を次に示します。

HTTP ヘッダ

ステータスコード: 503

Content-type: application/json;charset=UTF-8

ボディ電文: {"fault": {"content": "RpcErrorFault", "factor": "internal"}}

2.3.7 HCSC コンポーネントの停止・削除

利用した HCSC コンポーネントの停止と削除には、開発環境の機能を利用します。

詳細な手順については、次のマニュアルを参照してください。

「Cosminexus V11 BPM/ESB 基盤 サービスプラットフォーム 開発ガイド 基本開発編」

「8.6 HCSC コンポーネントを停止して HCSC サーバから削除する処理の一括実行」

2.3.8 HCSC サーバの停止

利用したテスト環境の停止する場合は、起動中のテスト環境を停止したあと、Eclipse を終了します。

詳細な手順については、次のマニュアルを参照してください。ただし、このサンプルでは HiRDB を利用していない

ため、HiRDB の停止手順は不要です。

「Cosminexus V11 BPM/ESB 基盤 サービスプラットフォーム ファーストステップガイド」

「7.2 テスト環境の停止」

2.3.9 ビジネスプロセスのデバッグ機能を利用した確認

ビジネスプロセスのデバッグでは、ビジネスプロセスの処理をステップ実行できます。

ビジネスプロセスの処理をステップ実行をすることで、アクティビティ間をどのように進んでいるかを確認したり、アクティビティで使用している変数の中身を確認・更新したりできます。

サンプルの実行には直接関係しませんが、ビジネスプロセスの詳細な処理を確認したい場合に利用できます。

ビジネスプロセスのデバッグ機能の利用方法や、デバッグで利用できる機能については、次のマニュアルを参照してください。

「Cosminexus V11 BPM/ESB 基盤 サービスプラットフォーム 開発ガイド 基本開発編」

「10. ビジネスプロセスのデバッグ」

3 OpenTP1 サーバの設定

この章では、OpenTP1 サーバの設定について説明します。

本章の構成

3.1 OpenTP1 のサービスの準備

3.1 OpenTP1 のサービスの準備

TP1 アダプタからアクセスする OpenTP1 の構築手順と設定を説明します。

1. 環境変数の確認

OpenTP1 インストール後、「DCCONFPATH」の環境変数に次のパスが設定されていることを確認してください。設定されていない場合は、設定してください。

変数	値
DCCONFPATH	<OpenTP1 のインストールディレクトリ> ¥conf

2. OpenTP1 環境設定

Windows のスタートメニューから、[OpenTP1]の[OpenTP1 環境設定]を実行し、OpenTP1 環境設定を実行します。「OpenTP1 環境設定」ダイアログが表示されたら、「定義種別」が「TP1_Server Base System Definition」であること、「モデル」が「Simple Model」であることを確認し、「次へ」をクリックします。

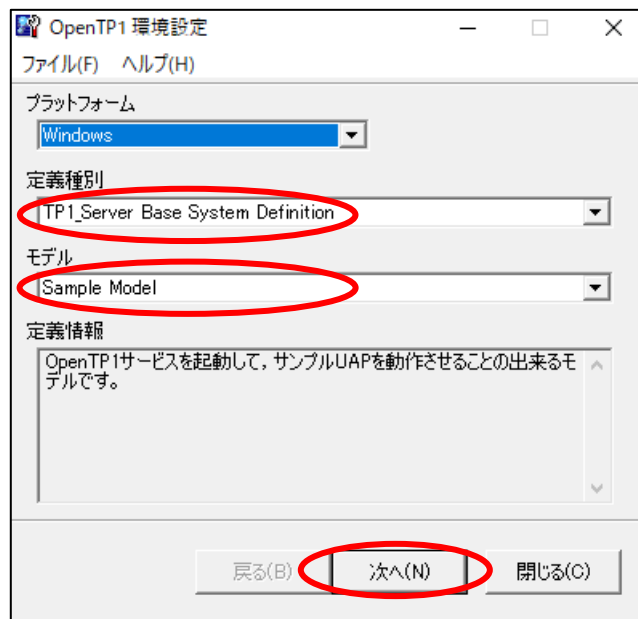


図 3-1 OpenTP1 環境設定(その 1)

「パラメータ情報」は変更しないで、「次へ」をクリックします。

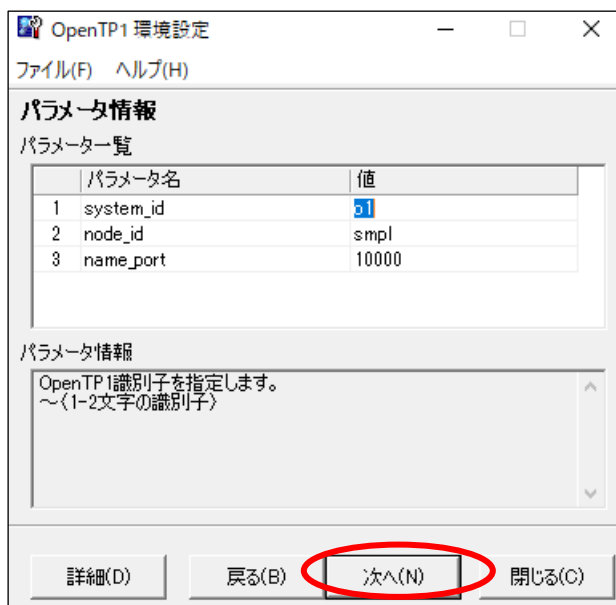


図 3-2OpenTP1 環境設定(その 2)

「格納先情報」も変更しないで、「実行」をクリックします。

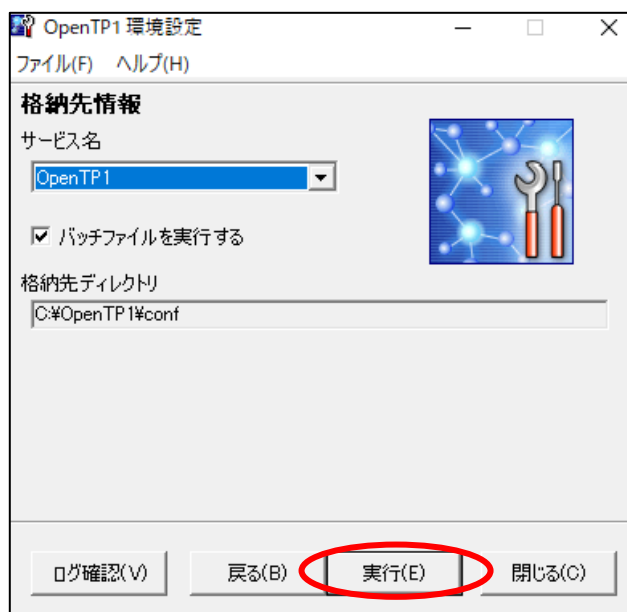


図 3-3OpenTP1 環境設定(その 3)

実行後、図 3-3 の「格納先ディレクトリ」に表示されたディレクトリにファイルが生成されていることを確認してください。

3. サンプル SPP の定義作成

Windows のスタートメニューから、[OpenTP1]の[OpenTP1 環境設定]を実行し、OpenTP1 環境設定を実行します。「OpenTP1 環境設定」ダイアログが表示されたら、「定義種別」が「User Service Definition」であること、「モデル」が「SPP Sample Model」であることを確認し、「次へ」をクリックします。

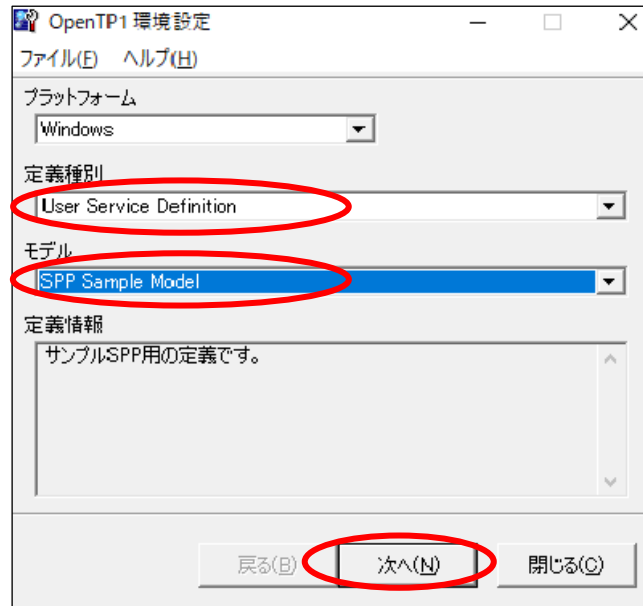


図 3-4 OpenTP1 環境設定(その 4)

「パラメータ情報」を確認し、「次へ」をクリックします。

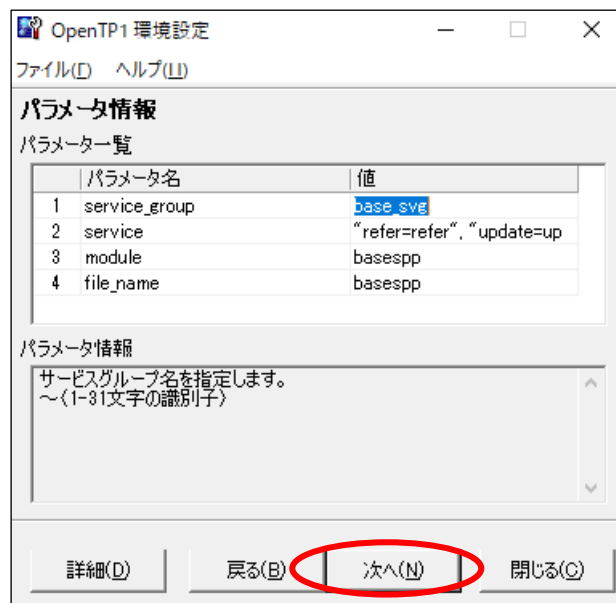


図 3-5 OpenTP1 環境設定(その 5)

「次へ」をクリックすると、図 3-3 と同様のダイアログが表示されるため、「実行」をクリックします。

4. サンプル SPP の作成

OpenTP1 の製品で提供されているサンプルは、<OpenTP1 インストールディレクトリ>%examples%base%aplib%c に格納されています。

サンプルを実行するためには、実行形式のファイルを作成しサービスを起動する必要があります。

次の手順で実行形式のファイルを作成します。

- i. <OpenTP1 インストールディレクトリ>%examples%base%aplib%c をカレントディレクトリに設定し、nmake コマンドを実行してください。

```
<OpenTP1 インストールディレクトリ>%examples%base%aplib%c>nmake
```

- ii. コマンド実行ディレクトリに生成された「basespp.exe」を、<OpenTP1 インストールディレクト>%aplib% に格納します。

5. OpenTP1 の起動

Windows のスタートメニューから、[OpenTP1]の[OpenTP1 起動/停止]を実行します。

表示されたダイアログから「起動」ボタンをクリックして、「停止」ボタンが活性状態になると OpenTP1 サーバが正常に起動できた状態となります。

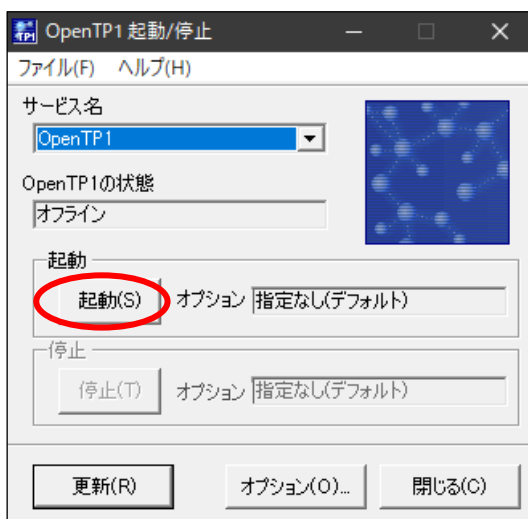


図 3-6 OpenTP1 起動/停止ダイアログ

6. OpenTP1 のサービスの起動

Windows のスタートメニューから、[OpenTP1]の[OpenTP1 コンソール]を開いてください。また、「OpenTP1 コンソール」とは別にコマンドプロンプトを開いて、次のコマンドを実行してください。

```
任意ディレクトリ>dcsvstart -u basespp
```

「OpenTP1 コンソール」に次の情報が出力されれば起動が正常に終了しました。

```
30900 o1 2020/05/22 12:28:03 adm KFCA01811-I サーバ basespp 開始中です。
30900 o1 2020/05/22 12:28:03 adm KFCA01813-I サーバ basespp がオンライン状態になりました。
```

【注意事項】

「basespp が見つかりません」という旨のメッセージが出力された場合は、環境変数「DOCONFATH」が設定されてない可能性があるため、再度確認してください。「DOCONFATH」が正しく設定されている場合は「DCDIR」や「Lib」等の環境変数が正しく設定されているか確認してください。

4 トラブルシュート

本章の構成

4.1 トラブルシュート

4.1 トラブルシュート

サンプル作成・実行時にエラーや障害が発生した場合、ログファイルに出力されるエラーメッセージを基に障害を特定して原因を解消します。

障害に対して取得できる情報の種類、および障害の対処方法については、次のマニュアルを参照してください。

「Cosminexus V11 BPM/ESB 基盤 サービスプラットフォーム システム構築・運用ガイド」

「7. 障害対策」

付録A. 変更した定義ファイルのマニュアル記載箇所

サンプルを使用するに当たり、デフォルト値から設定値を変更した定義ファイルおよびその参照マニュアルを、次の表に示します。

表 A-1 参照マニュアル一覧

#	定義ファイル名	参照マニュアル
1	HTTP 受付サービス選択定義ファイル	BPM/ESB 基盤 サービスプラットフォーム リファレンス
2	HTTP 受付定義ファイル	BPM/ESB 基盤 サービスプラットフォーム リファレンス
3	TP1 アダプタ通信構成定義ファイル	BPM/ESB 基盤 サービスプラットフォーム リファレンス
4	TP1/Client/J 環境定義ファイル	OpenTP1 クライアント使用の手引 TP1/Client/J 編
5	TP1 アダプタ環境定義ファイル	BPM/ESB 基盤 サービスプラットフォーム リファレンス
6	TP1 アダプタ実行環境プロパティファイル	BPM/ESB 基盤 サービスプラットフォーム リファレンス
7	HCSC サーバランタイム定義ファイル	BPM/ESB 基盤 サービスプラットフォーム リファレンス

付録B. サンプルで使用するOpenTP1のサービス

このサンプルで使用する OpenTP1 のサービスについては、次のマニュアルを参照してください。

「OpenTP1 使用の手引 Windows(R)編」 「サンプルプログラム」

「OpenTP1 プログラム作成の手引」 「サンプルプログラムの仕様」

—以上—