

# RT システムビルダー 操作マニュアル

---

Ver.0.7.1

2015/08/02



株式会社 グローバルアシスト

---

# 目次

<b>1</b>	<b>概要</b> .....	<b>3</b>
1.1	ツール概要.....	3
1.2	対象.....	3
1.3	制限.....	3
<b>2</b>	<b>インストールと設定</b> .....	<b>4</b>
2.1	RT システムビルダーのインストール.....	4
2.2	RT システムビルダーの初期設定.....	6
2.3	RT システムビルダー画面の起動.....	6
<b>3</b>	<b>モデル作成サポート機能</b> .....	<b>7</b>
3.1	OPENRTM-AIST 事前定義型インポート機能.....	7
3.2	IDL ファイル定義型インポート機能.....	9
3.3	RTC プロファイルインポート機能.....	11
<b>4</b>	<b>RT コンポーネント設計用モデル作成機能</b> .....	<b>12</b>
4.1	RT コンポーネントの指定方法.....	12
4.2	インポートした RTC ブロックの利用方法.....	14
4.2.1	データポートのみ持つ RT コンポーネントの場合.....	14
4.2.2	サービスポートを持つ RT コンポーネントの場合.....	14
4.3	RT コンポーネント本体の情報設定.....	15
4.4	コンフィギュレーションセット情報の設定.....	17
4.5	データポート情報の設定.....	18
4.6	サービスポート情報の設定.....	19
4.7	サービスインターフェース情報の設定.....	20
<b>5</b>	<b>RTC プロファイルの生成</b> .....	<b>21</b>
<b>6</b>	<b>デフォルト値設定</b> .....	<b>22</b>
6.1	基本設定.....	22
6.2	アクティビティドキュメント設定.....	23
6.3	ポート設定.....	23
<b>7</b>	<b>改版履歴</b> .....	<b>24</b>

# 1 概要

## 1.1 ツール概要

RTシステムビルダーは、SysML モデルから RT コンポーネントのプロファイル(RtcProfile)を自動生成するためのツールです。本ツールはモデリングツール Enterprise Architect のアドインとして実装されています。本ツールを用いることで、上位工程で行ったモデリング結果から、RT コンポーネントの開発まで繋げることが可能となります。また、既存の RT コンポーネントのプロファイル情報や IDL で定義したデータ型情報をインポートして、モデル構築時に利用する事も可能となります。

本ツールの動作に必要な環境は以下の通りです。

No.	環境	備考
1	モデリングツール Enterprise Architect ( <a href="http://www.sparxsystems.jp/">http://www.sparxsystems.jp/</a> )	注意:バージョン 11.x 以降で動作確認を行っております。
2	MDG Technology for SysML ( <a href="http://www.sparxsystems.jp/products/MDG/mdgsysml.htm">http://www.sparxsystems.jp/products/MDG/mdgsysml.htm</a> )	ご使用になられる Enterprise Architect に合ったバージョンを利用してください
3	.Net Framework Ver.4 ( <a href="http://www.microsoft.com/ja-jp/default.aspx">http://www.microsoft.com/ja-jp/default.aspx</a> )	

また RT コンポーネントの開発を行うためには、以下の開発環境をインストールしてください。

No.	環境	備考
1	OpenRTM-aist-1.X.X ( <a href="http://www.openrtm.org/">http://www.openrtm.org/</a> )	RT コンポーネントが動作するために必要な RT ミドルウェアです。お使いの環境に合ったバージョンをインストールしてください。
2	RTCBuilder ( <a href="http://www.openrtm.org/openrtm/ja/node/30">http://www.openrtm.org/openrtm/ja/node/30</a> )	RT コンポーネントのソースを自動生成するツールです。

## 1.2 対象

本ドキュメントでは、作成した SysML モデルから、RT コンポーネント向けプロファイルを作成する方法を説明しています。SysML 自体の説明や、Enterprise Architect の操作方法に関する説明、RT ミドルウェア (OpenRTM-aist)の説明、RTCBuilder の説明などにつきましては、各ツールの操作マニュアルなどを参照してください。(なお、以降の説明中の各図は Windows7 のスクリーンショットです。)

## 1.3 制限

本ツールは、OpenRTM-aist を対象に開発されたものです。その他の RTC プラットフォームに対する動作は想定しておりません。また、本ツールは Windows 向けに開発されたツールです。その他の OS 上での動作確認は行っていません。

## 2 インストールと設定

RT システムビルダーは、UML モデリングツール EnterpriseArchitect のアドインとして実装されています。このため、事前に EnterpriseArchitect 本体および SysML アドイン(MDG Technology for SysML)をインストールする必要があります。(EnterpriseArchitect のインストール方法については、ツールに付属のマニュアルを参照してください。)

### 2.1 RT システムビルダーのインストール

RT システムビルダーのインストールは、インストール用バッチファイル「RTSystemBuilderSetup.bat」を実行して行います。こちらのバッチファイルを実行後、EnterpriseArchitect を起動し、上部メニュー中の「アドイン・拡張」の部分に「RTSystemBuilder」が追加されていれば、インストールは完了です。

※インストール用バッチファイルは、レジストリの情報を変更するため、管理者権限のあるユーザが「管理者として実行」で実行してください。

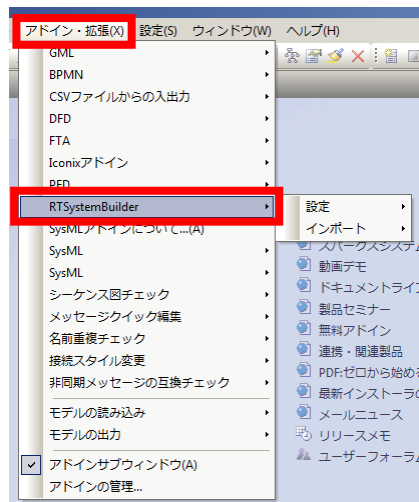


図 2.1.1 RT システムビルダーのインストール

もしも、上記メニューが表示されていない場合は、上部メニュー中の「アドイン・拡張」中の「アドインの管理」を選択し、【アドインマネージャ】を起動してください。【アドインマネージャ】内の「有効なアドイン」リストの中に「RTSystemBuilder」が含まれている場合は、インストール自体は成功しています。右側の「起動時に読み込み」欄にチェックを入れて、Enterprise Architect 本体を再起動してください。

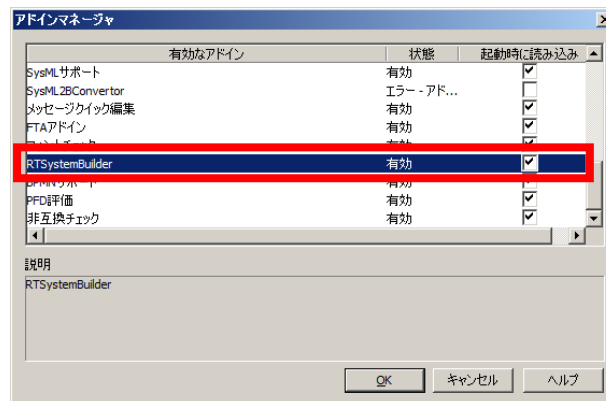


図 2.1.2 アドインマネージャの設定

インストールバッチを実行しても【アドインマネージャ】内に「RTSystemBuilder」の項目が追加されていない場合には、ツールのインストールに失敗しています。以下の手順で手動インストールを行っててください。

01.RT システムビルダの DLL ファイル(RTSystemBuilder.dll)を、Enterprise Architect 本体のインストールディレクトリにコピーしてください。

→デフォルト設定のまま Enterprise Architect をインストールしている場合、インストール先は以下のディレクトリとなります。

C:\Program Files (x86)\SparxSystems Japan\EA\

02.コマンドプロンプトを起動し、EnterpriseArchitect 本体のインストールディレクトリに移動してください。その後、アセンブリ登録ツール(RegAsm.exe)を実行してください。

→アセンブリ登録ツール(RegAsm.exe)は、通常、以下のディレクトリに存在します。

C:\Windows\Microsoft.NET\Framework\v4.0.30319\RegAsm.exe

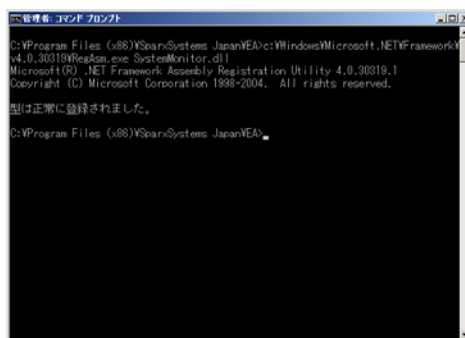


図 2.1.3 アセンブリ登録ツール

03.「スタート」メニューの「プログラムとファイルの検索」もしくは「プログラムを指定して実行」にて、「regedit」と入力し、レジストリエディタを起動します。

04.左側のウィンドウで、「HKEY\_CURRENT\_USER\Software\SparxSystems\EAAddins」を選択します

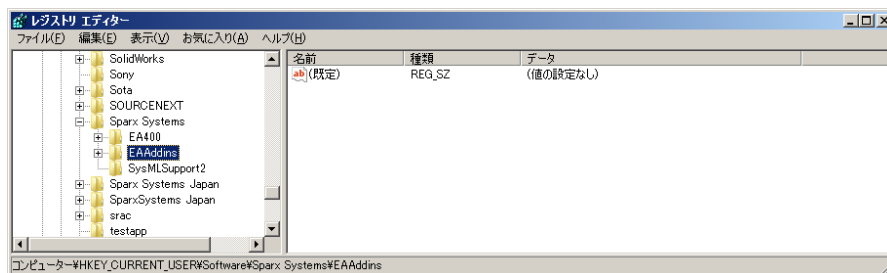


図 2.1.4 レジストリエディタ

05.上記「EAAddins」を右クリックして、新規にキーを作成し、アドインの名称を入力します。

→アドインの名称は任意です。DLL のファイル名や名前空間と一致している必要はありません。ここで付けた名称が EnterpriseArchitect のアドインマネージャに表示されます。

06.作成したキーに、「RTSystemBuilder.Connector」と設定します。

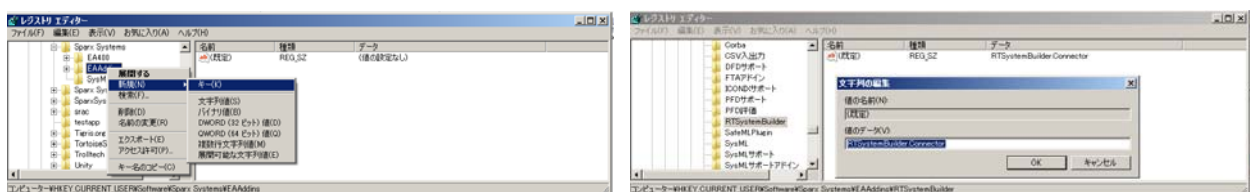


図 2.1.5 レジストリの設定

以上の操作で、RT システムビルダーの情報が Enterprise Architect に登録されます。Enterprise Architect を再起動し、上部メニュー中の「アドイン・拡張」の部分に「RTSystemBuilder」が追加されていることを確認してください。

※上記の手動インストールを行っても、上部メニュー中の「アドイン・拡張」の部分に「RTSystemBuilder」が追加されない場合は、開発元までご連絡ください。

## 2.2 RT システムビルダーの初期設定

RT システムビルダーでは、各種ファイルを生成する際に、一時的な作業ディレクトリを使用します。このため、最初に作業ディレクトリの指定を行ってください。

RTシステムビルダー用作業ディレクトリの設定は、上部メニュー中の「アドイン・拡張」メニューから「RTSystemBuilder」→「設定」→「作業ディレクトリ設定」を選択してください。ディレクトリ選択画面が表示されますので、対象ディレクトリを選択してください。

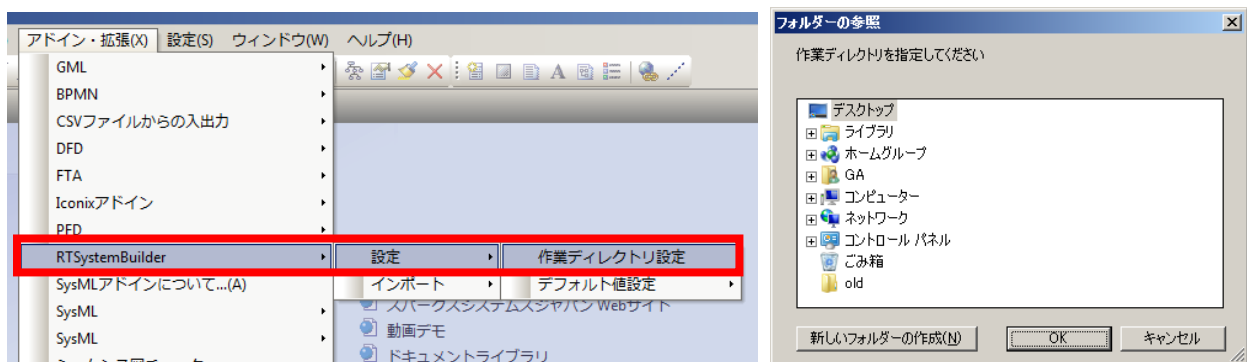


図 2.2.1 作業ディレクトリの設定

## 2.3 RT システムビルダー画面の起動

RT システムビルダーでは、各種設定を行う画面をアドイン・サブウィンドウとして用意しています。アドイン・サブウィンドウが表示されていない場合、上部メニュー中の「アドイン・拡張」メニュー中の「アドイン・サブウィンドウ」のチェックを ON にしてください。「アドイン」というタブ名の画面がアドイン・サブウィンドウとして表示されます。

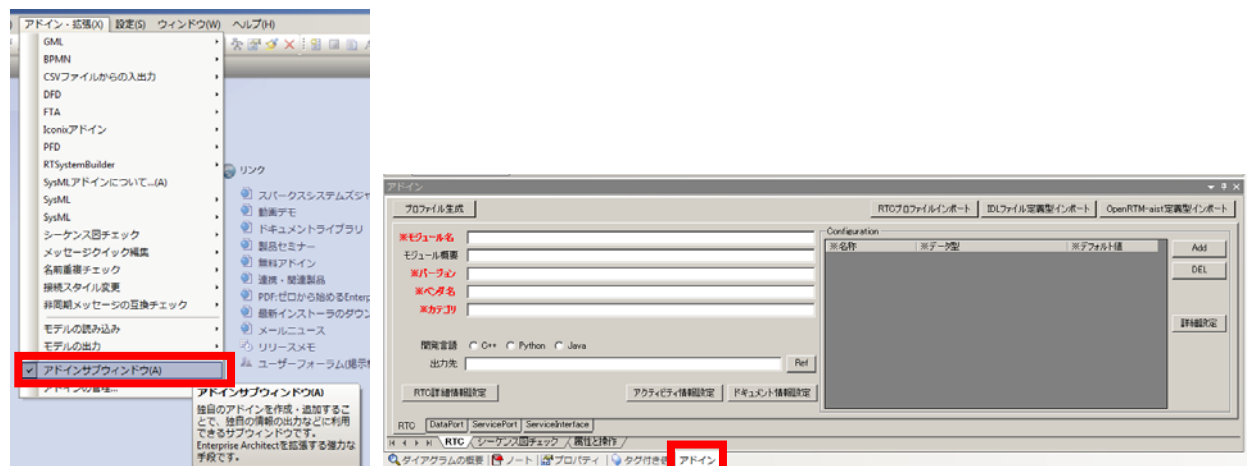


図 2.3.1 アドイン・サブウィンドウの表示

### 3 モデル作成サポート機能

本機能は、RT システムビルダーでRTコンポーネントの設計を行う際に、既存コンポーネントの情報や事前定義されたデータ型を再利用して、モデル作成をサポートする機能です。

#### 3.1 OpenRTM-aist 事前定義型インポート機能

本機能は、OpenRTM-aistで事前定義されている各種型情報を、RTシステム設計時に利用するための機能です。各種標準仕様で定義されている型情報を SysML のブロック要素としてインポートします。

上部メニュー中の「アドイン・拡張」メニューから「RTSystemBuilder」→「インポート」→「OpenRTM-aist 定義型インポート」を選択するか、RT システムビルダー用画面内の「OpenRTM-aist 型インポート」ボタンをクリックしてください。【インポート対象選択】画面が表示されますので、モデル内にインポートを行いたいデータ型を選択し、「OK」をクリックしてください。

なお、本機能でインポート可能なデータ型は以下のとおりです。

- OpenRTM-aist 標準/拡張型  
OpenRTM-aist がデフォルトで用意しているデータ型。BasicDataType.idl, ExtendedDataTypes.idl, InterfaceDataTypes.idl で定義されている内容が対象。「RTC」パッケージ配下にインポート。
- カメラ機能共通インターフェース  
「カメラ機能共通インタフェース仕様書(第 2.1 版)」で定義されているデータ型。「Img」パッケージ配下にインポート。
- 自律移動機能共通インターフェース  
「自律移動機能共通インタフェース仕様書(第 1.0 版)」で定義されているデータ型。「IIS」パッケージ配下にインポート。
- ロボットアーム制御機能共通インターフェース  
「ロボットアーム制御機能共通インタフェース仕様書(SI 単位系準拠 第 1.0 版)」で定義されているデータ型。「JARA\_ARM」パッケージ配下にインポート。

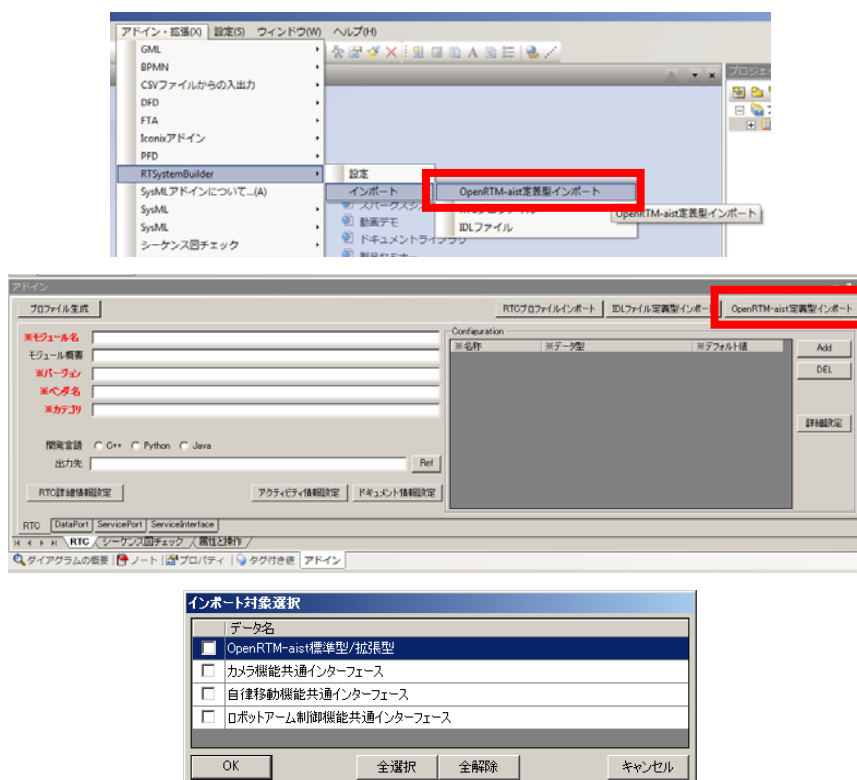


図 3.1.1 OpenRTM-aist 定義型インポート

インポートを実行すると、作成中モデルのルートパッケージ配下にインポート対象パッケージが追加されます。そして追加した対象パッケージ配下に各種データ型用のブロックが追加されます。

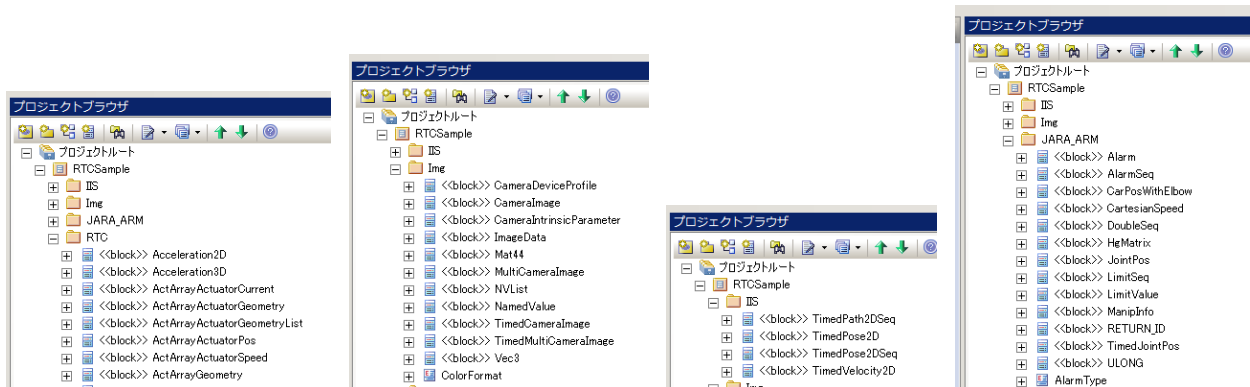


図 3.1.2 OpenRTM-aist 定義型インポート後の状態

なお、ルートパッケージ配下にインポート対象パッケージと同一名のパッケージが存在する場合には、以下の警告画面が表示され、処理を継続するかを確認します。ここで「はい」を選択した場合、既存のパッケージとは別パッケージとしてインポート対象パッケージが追加され、データ型の追加が行われます。

※既存のパッケージ内にインポートされるのではなく、同一名称のパッケージを新たに追加してインポートを行います。

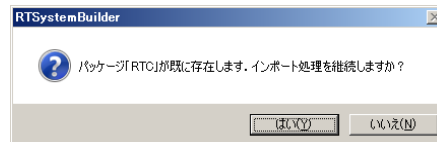


図 3.1.3 OpenRTM-aist 定義型インポート処理の確認



### 3.2 IDL ファイル定義型インポート機能

本機能は、ユーザが独自に定義したデータ型情報を、RT システム設計時に利用するための機能です。IDL ファイルの中身をパースし、定義されている型情報を SysML のブロック要素としてインポートします。

上部メニュー中の「アドイン・拡張」メニューから「RTSystemBuilder」→「インポート」→「IDL ファイル」を選択するか、RTシステムビルダー用画面内の「IDL ファイル定義型インポート」ボタンをクリックしてください。【ファイル選択】画面が表示されますので、モデル内にインポートを行いたい IDL ファイルを選択し、「OK」をクリックしてください。

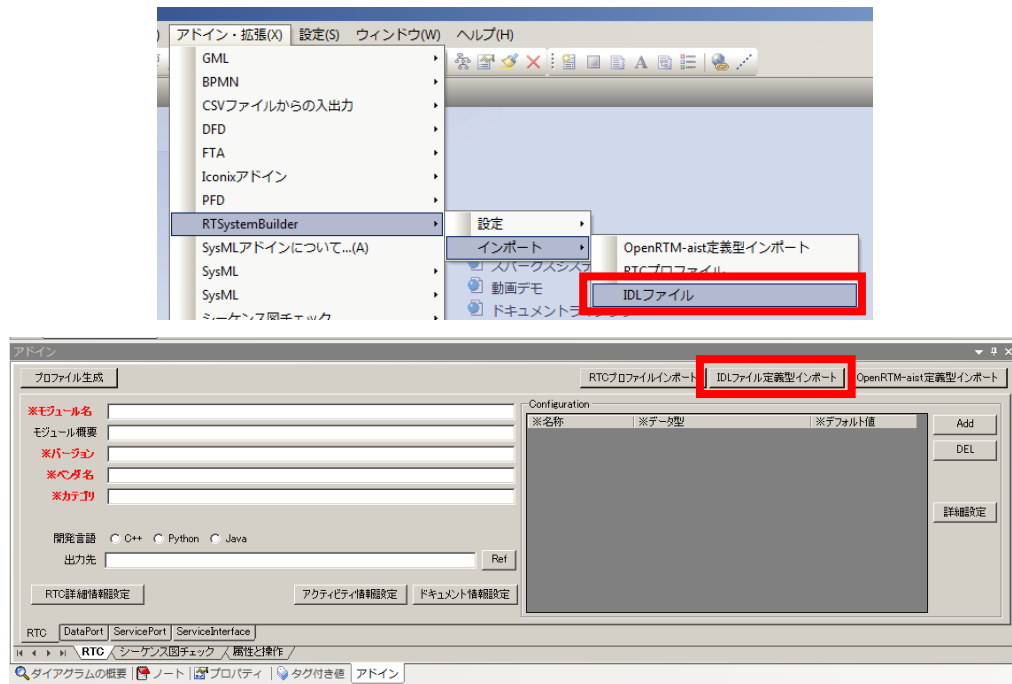


図 3.2.1 IDL ファイル定義型インポート

対象 IDL ファイルを選択し、インポートを実行すると、作成中モデルのルートパッケージ配下に IDL 内で定義されている各種データ型用のブロックが追加されます。なお、追加対象のデータ型は以下となります。

- ・構造体: struct で定義された型。内部メンバ情報とともにインポートされる。
- ・列挙型: enum で定義された型。内部メンバとともにインポートされる。
- ・型定義: typedef で定義された型。宣言元の型と宣言先の型がインポートされる。

また IDL ファイル内でモジュール(module)が定義されている場合、各モジュールに対応したパッケージ構造も追加され、定義に従った形でブロックが追加されます。

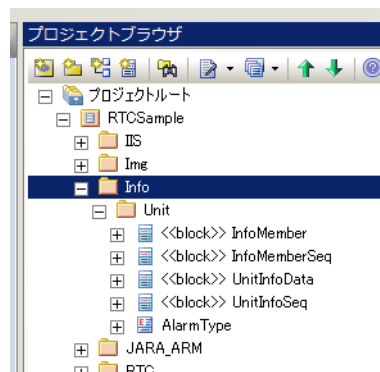


図 3.2.2 IDL ファイル定義型インポート後の状態

なお、インポート先のモデル階層にインポート対象の型と同一名称、同一定義のブロックが既に存在する場合には、何も処理は行いません。

しかし、インポート先のモデル階層にインポート対象の型と**同一名称、異なる定義**のブロックが既に存在する場合、以下のような【IDL インポート】画面が表示され、対象の型が一覧表示されます。インポート元のIDL ファイル内の定義を優先する型に対してチェックボックスを ON にしてください。チェックボックスが ON になった型情報に対応するブロックは、モデル内の定義が IDL ファイル内の定義に更新されます。(チェックボックスが OFF の型については、何も処理を行いません。)

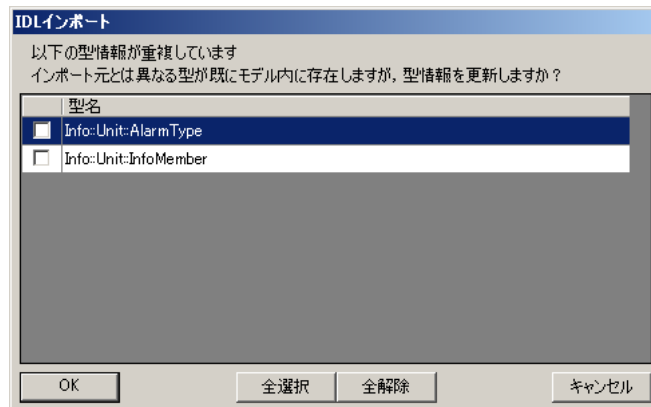


図 3.2.3 インポート対象選択画面

### 3.3 RTC プロファイルインポート機能

本機能は、既存の RT コンポーネントの情報を、RT システム設計時に利用するための機能です。RTC プロファイル(XML ファイル)の内容をパースし、定義されている RT コンポーネントを SysML のブロック要素としてインポートします。

上部メニュー中の「アドイン・拡張」メニューから「RTSystemBuilder」→「インポート」→「RTC プロファイル」を選択するか、RT システムビルダー用画面内の「RTC プロファイルインポート」ボタンをクリックしてください。【ファイル選択】画面が表示されますので、モデル内にインポートを行いたい RTC プロファイルを選択し、「OK」をクリックしてください。

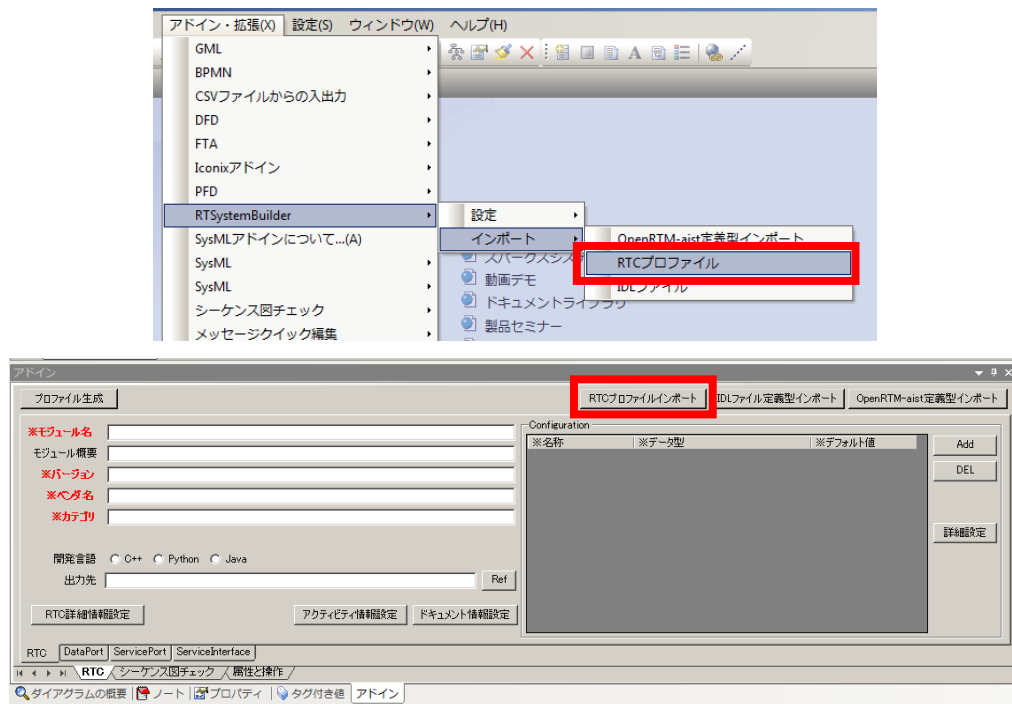


図 3.3.1 RTC プロファイルインポート

対象プロファイルを選択し、インポートを実行すると、作成中モデルのルートパッケージ配下に「Imported」というパッケージが追加されます(既に存在する場合には追加されません)。そして、対象 RTC プロファイル内で定義されている「カテゴリ名」と同一名称のパッケージが追加され、その配下に RTC に対応するブロックが追加されます。

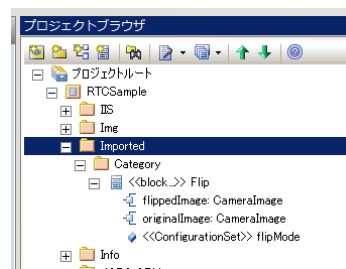


図 3.3.2 RTC プロファイルインポート後の状態

なお、インポートされたブロックには、選択した RTC プロファイルで定義されていた情報が自動的に格納されます。また、ポート情報、コンフィギュレーションセット情報が含まれる場合には、それらに対応した要素(ポート、属性)も自動的に追加されます。更に、ポートに設定された型、コンフィギュレーションセットに設定された型が既存のモデルに存在しない場合には、型定義に対応するブロックも自動的に追加されます。

**※一部、RTC プロファイルの中でインポートできない情報も存在します。詳しくは第4章を参照してください。**

## 4 RT コンポーネント設計用モデル作成機能

本機能は、SysML のブロック定義図および内部ブロック図の情報を用いて、RT コンポーネントのプロファイルを生成するための機能です。

なお、作成したモデルの各種情報をデフォルト表示するため、事前に以下の設定を行っておくと便利です。上部メニューの「ツール」→「オプション」を選択し、表示された【オプション】画面で「要素」を選択。「ポートとパートの型を常に表示」を選択。

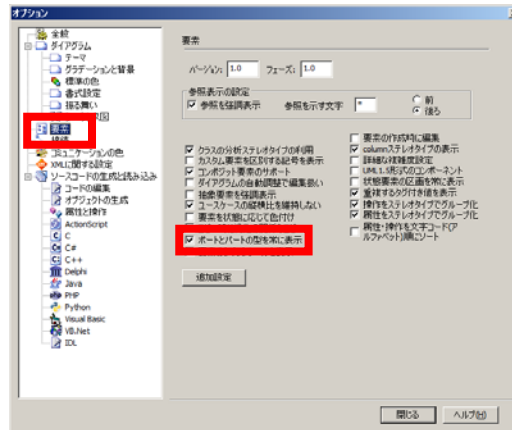


図 4.1.1 ツールのオプション設定

### 4.1 RT コンポーネントの指定方法

RT コンポーネントは、システムを構成する要素の一部となりますので、SysML のブロックを用いて定義します。

作成したモデル中のどの要素を RT コンポーネントとするかを指定するためには、ブロック定義図内で該当ブロックに、以下の手順でステレオタイプ「RTC」を追加してください。

01. 該当ブロックをダブルクリックもしくは右クリックで表示されるコンテキストメニューから「プロパティ」を選択
  02. 表示された【プロパティ】画面で、「ステレオタイプ」右側の「…」ボタンをクリック
  03. 表示された【ステレオタイプ】画面で、「新規」をクリック
  04. 表示された【新規ステレオタイプ】画面で、「RTC」と入力し、「OK」ボタンをクリック
  05. 【ステレオタイプ】画面に追加された「RTC」のチェックボックスを ON にし、「OK」をクリック
- ※上記手順の 03,04 につきましては、初めてステレオタイプを追加する場合の手順となります。2回目以降は、ステレオタイプ「RTC」は追加済みですので、こちらの手順は不要です。

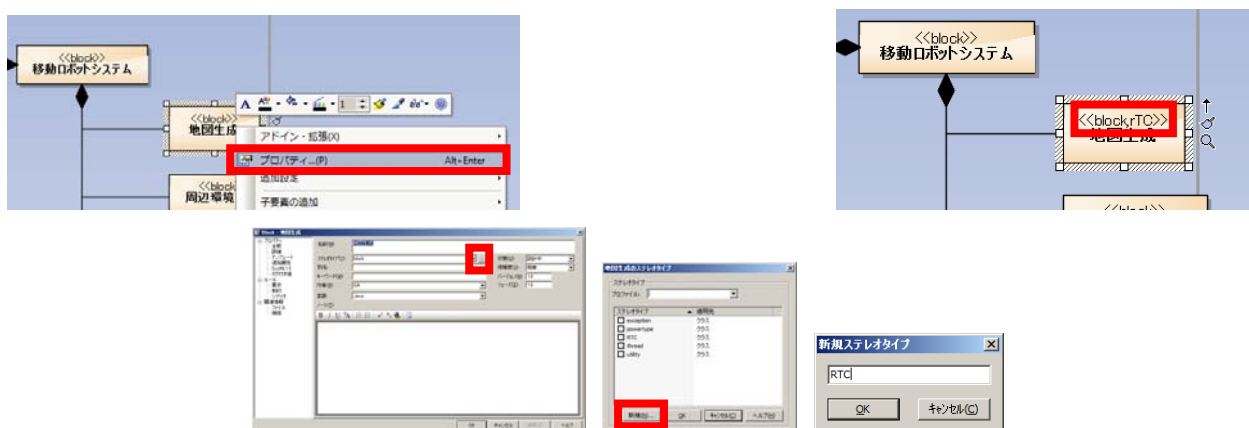


図 4.1.1 ステレオタイプ「RTC」の追加

ブロック定義図でステレオタイプ「RTC」を追加した要素を親に持つ「プロパティ」を内部ブロック図内で選択すると、RT システムビルダー用画面内に各種情報が表示されます。また、付属の「ポート」を選択した場合、ポートの種類に応じて、データポートの情報、サービスポートの情報がそれぞれ表示されます。

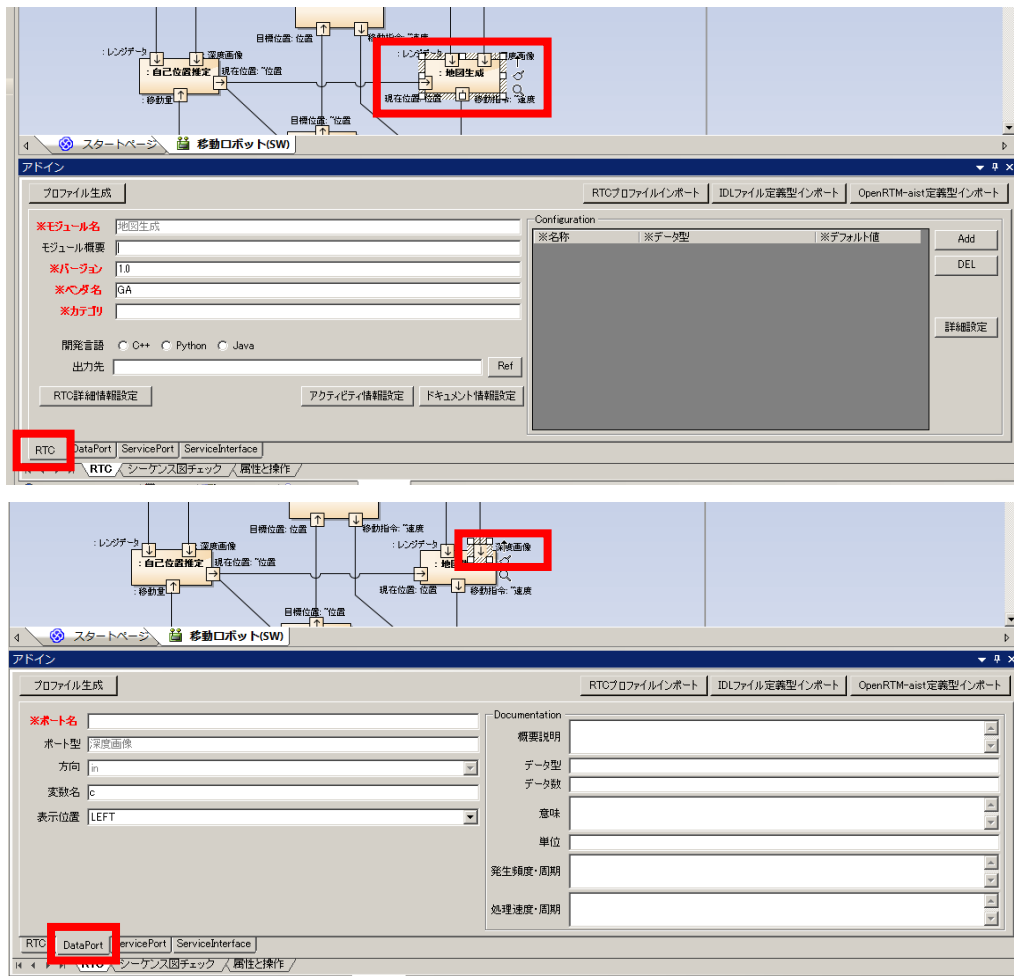


図 4.1.2 RTC 情報、ポート情報の表示

SysML モデル中と実際の RT コンポーネントでは各種名称を変更したい場合も多いため、RT コンポーネント本体、データポート、サービスポート、コンフィギュレーションセットの名称につきましては、「別名」が優先されます(もしも「別名」が設定されていない場合は、「名称」の内容を使用します)。モデルと実際の名称を変更したい場合には、「別名」を利用してください。なお、SysML のモデルエディタ画面では基本的に「名称」が表示されます(設定によって変更可能)。RT システムビルダー用画面内では、「別名」が設定されている場合には「別名」が、「別名」が設定されていない場合には、「名称」が表示されます(RTC プロファイルに設定される名称と同じ名称が表示されます)。

## 4.2 インポートした RTC ブロックの利用方法

RTC プロファイルインポート機能を用いて、既存の RTC をインポートした場合には、既にステレオタイプが設定されていますので、生成されたブロックをドラッグアンドドロップ(DnD)するだけで RTC として使用可能です。

### 4.2.1 データポートのみ持つ RT コンポーネントの場合

単純に対象ブロックを DnD するだけで利用可能です。DnD 時に表示される設定画面では、「動作の選択」は「そのまま配置」、「付属要素」は「全て」を選択してください。

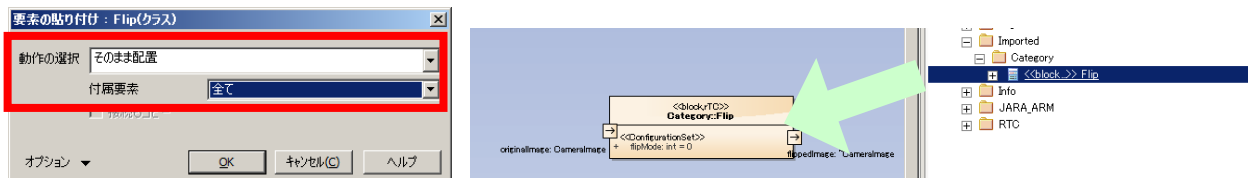


図 4.2.1 データポートのみの RTC の利用

※インポート RTC ブロックを利用した場合、内部ブロック図内で対象ポートのプロパティ画面を表示しても、名称や型、方向などの詳細情報を変更することはできません(「OK」ボタンが選択できないようになります)。もしもこれらの情報を修正したい場合には、ブロック定義図もしくは【プロジェクトブラウザ】内で該当ポートのプロパティを変更してください。ただし、この方法を用いて変更した内容は、対象ポートを利用して全ての要素の影響しますのでご注意ください。

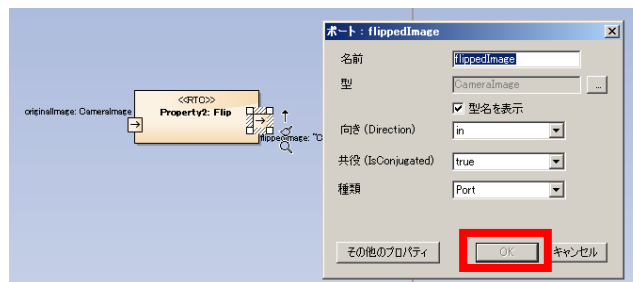


図 4.2.2 インポート RTC ブロックの属性変更

### 4.2.2 サービスポートを持つ RT コンポーネントの場合

上記の「データポートのみを持つ RT コンポーネントの場合」と同様な手順で、モデル内に追加します。しかし、この状態ではサービスインターフェースが表示されていないので、サービスポートを右クリックして表示されるコンテキストメニュー中から「付属要素」を選択してください。そして表示された【付属要素】画面右上の「継承要素も表示」チェックボックスを ON にしてください。一覧画面にインターフェースが表示されますので、「名前」列の右側のチェックボックスを ON にして「閉じる」ボタンを選択すると、モデル内にサービスインターフェースが表示されます。

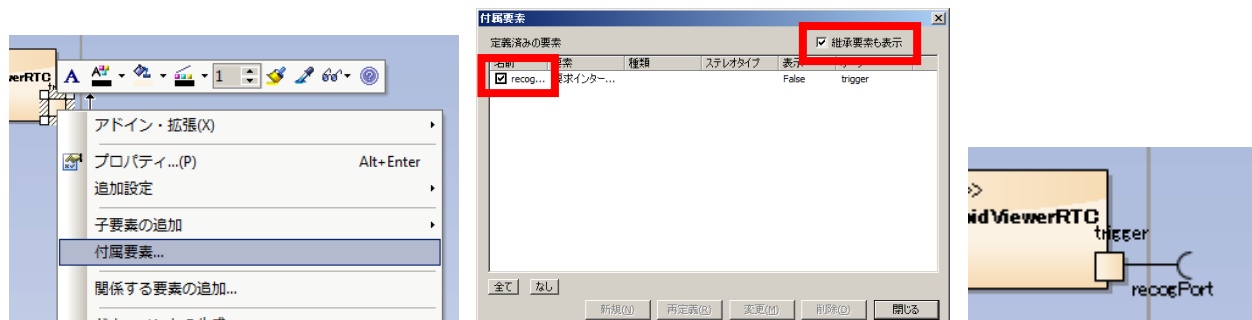


図 4.2.3 サービスインターフェースの表示

### 4.3 RT コンポーネント本体の情報設定

ブロック定義図でステレオタイプ「RTC」を追加した要素を親に持つ「プロパティ」を内部ブロック図内で選択すると、RT システムビルダー用画面内の「RTC」タブが自動的に選択され、RT コンポーネントの各種情報が表示されます。

なお、本ツールで指定できる内容は、RTCBUILDER と同等の内容となります。このため、各情報の詳細につきましては、RTCBUILDER のマニュアルをご参照ください。

※RTCBUILDER の「**コンフィギュレーション・プロファイル**」中の「**RT-Component Configuration Parameter**」および「**ドキュメント情報**」中の「**バージョンアップログ**」、「**言語・環境**」情報中の「**環境**」情報につきましては、本ツールでは未対応です。もしもこれらの情報を設定したい場合には、RTCBUILDER を用いて設定を行ってください。

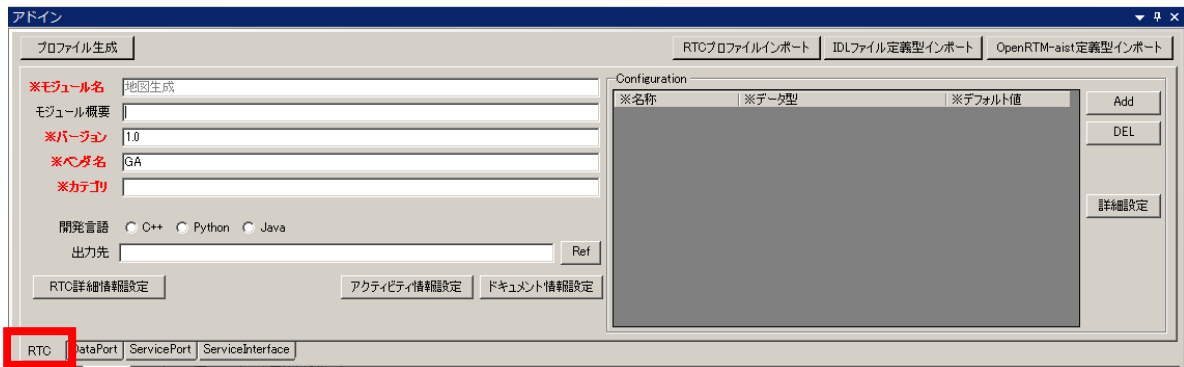


図 4.3.1 RT コンポーネントの情報

画面左側は、RT コンポーネントの基本情報となります。※印がついた項目は必須入力項目です。ただし、「モジュール名」につきましては、割り当てられたブロックの名称が設定され、本画面で変更する事はできません。「モジュール名」を変更したい場合には、該当ブロックの名称を変更してください。また、その他の項目については、必要に応じて設定を行ってください。

更に、本ツールでは複数 RT コンポーネントの情報を設定、プロファイル生成することができるため、RTC プロファイルの出力先を個別に指定する事が可能です。

RT コンポーネントの詳細情報を設定する場合には、画面左下の各ボタンを選択してください。

「RTC 詳細情報設定」ボタンを選択すると、以下のような【RTC 詳細情報】画面が表示されます。「コンポーネント種類」は必須項目となります。それ以外の項目はオプションとなりますので、必要に応じて設定を行ってください。

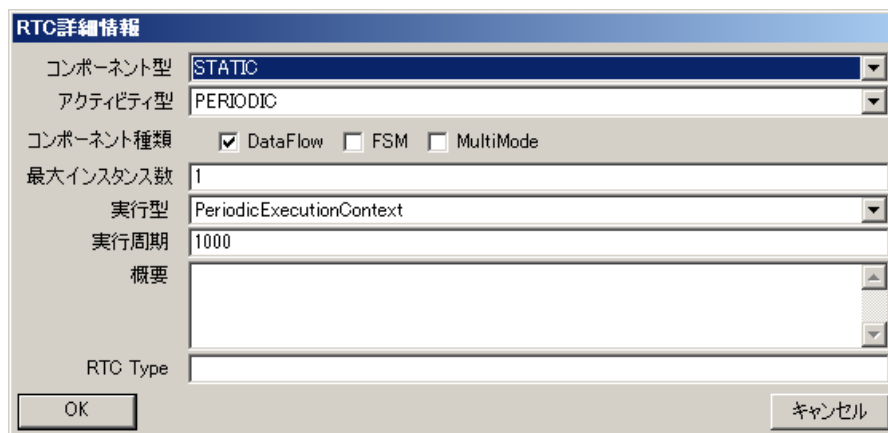


図 4.3.2 RT コンポーネント詳細情報の設定

「ドキュメント情報設定」ボタンを選択すると、以下の【ドキュメント設定】画面が表示されます。こちらの項目も全てオプションとなっておりますので、必要に応じて設定を行ってください。

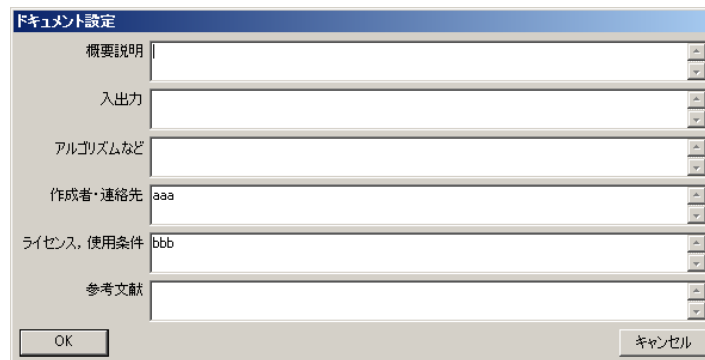


図 4.3.3 ドキュメント情報の設定

「アクティビティ情報設定」ボタンを選択すると、以下の【アクティビティ設定】画面が表示されます。設定を行いたいアクティビティに対応するボタンを選択すると【アクティビティ詳細設定】画面が表示されますので、必要な情報を設定してください。

なお、【アクティビティ詳細設定】画面で「ON」に設定されたアクティビティは、【アクティビティ設定】画面でボタンが青色となります。

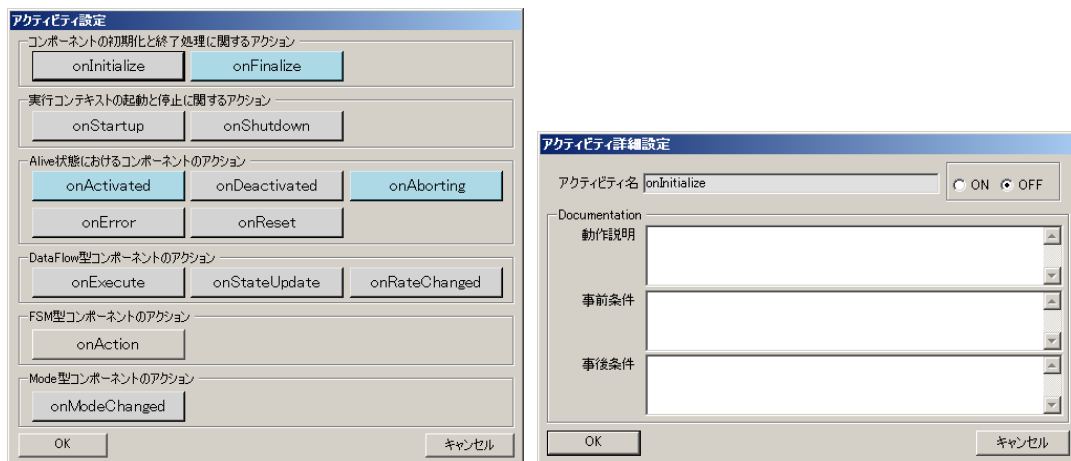


図 4.3.4 アクティビティ情報の設定



## 4.4 コンフィギュレーションセット情報の設定

RT コンポーネントが持つコンフィギュレーションセットは、ブロックの属性として定義します。「RTC」タブ内の左側を利用して、定義することが可能です。

新規コンフィギュレーションセットを追加したい場合には、対象要素を内部ブロック図内で選択した後、「Add」ボタンを選択してください。逆に、既存のコンフィギュレーションセットを削除したい場合には、対象要素を内部ブロック図内で選択した後、グリッド内で該当項目を選択し、「Delete」ボタンを選択してください。

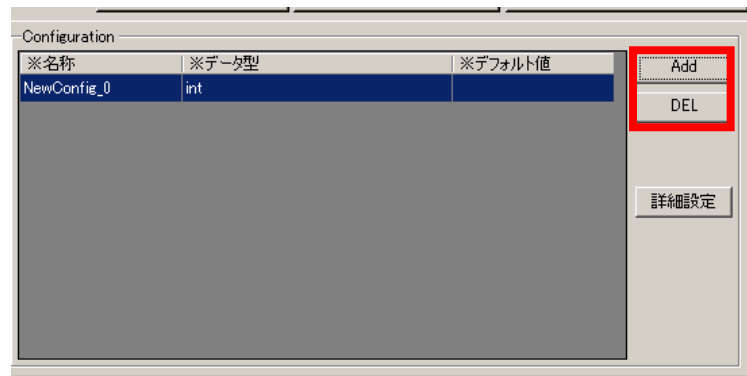


図 4.4.1 コンフィギュレーションセットの追加/削除

新規追加されたコンフィギュレーションセットは、各種設定がデフォルト設定のままとなっています。各コンフィギュレーションセットの「名称」「デフォルト値」につきましては、グリッド内で直接編集が可能です。それ以外の項目を変更したい場合には、該当項目を選択し、「詳細設定」ボタンを選択してください。以下の【ConfigurationSet 詳細設定】画面が表示され、各項目の設定を行うことができます。

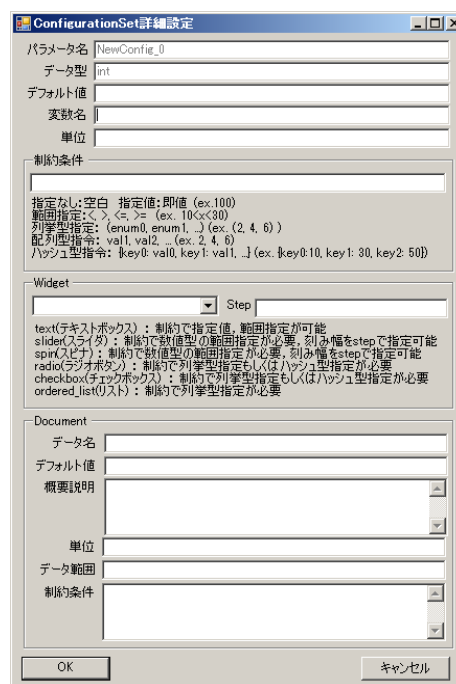


図 4.4.2 コンフィギュレーションセットの編集

なお、コンフィギュレーションセットの情報は、対象プロパティに割り当てられたブロックの属性として定義されています。このため、上記の手順で変更を行った場合には、この親ブロックの属性が変更されますのでご注意ください。

また、すでにブロックに定義済みの属性をコンフィギュレーションセットとして使用した場合には、属性のステレオタイプに「ConfigurationSet」を追加してください。

## 4.5 データポート情報の設定

RT コンポーネントのデータポートは、フロープロパティを持つポートとして定義します。内部ブロック図内で該当要素を選択すると、RT システムビルダー用画面内の「DataPort」タブが自動的に選択されデータポートの各種情報が表示されます。

※ステレオタイプ「RTC」を追加した要素を親に持つ「プロパティ」内に定義されたポートが対象となります。

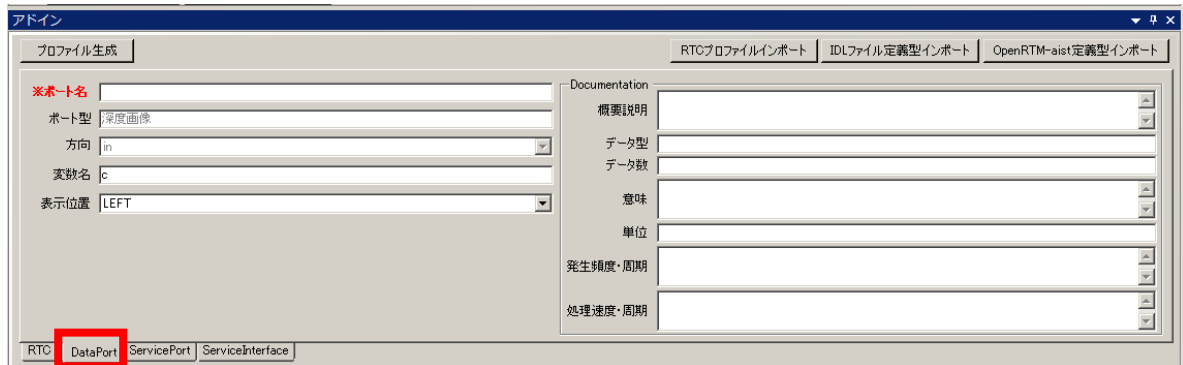


図 4.5.1 データポート情報の設定

「ポート名」「ポート型」「方向」につきましては、対象ポートの情報が表示され、本画面で変更する事はできません。これらの情報を変更したい場合には、該当ポートの情報を変更してください。

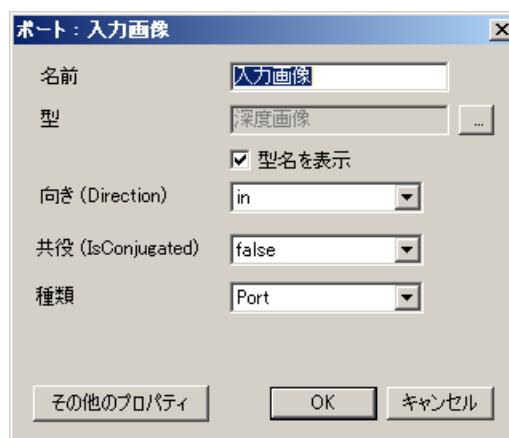


図 4.5.2 ポート情報の編集

また、その他の各種情報につきましては、どれもオプション扱いとなっておりますので、必要に応じて各種項目を設定してください。

## 4.6 サービスポート情報の設定

RT コンポーネントのサービスポートは、フロープロパティを持たない通常のポートとして定義します。内部ブロック図内で該当要素を選択すると、RT システムビルダー用画面内の「ServicePort」タブが自動的に選択されサービスポートの各種情報が表示されます。

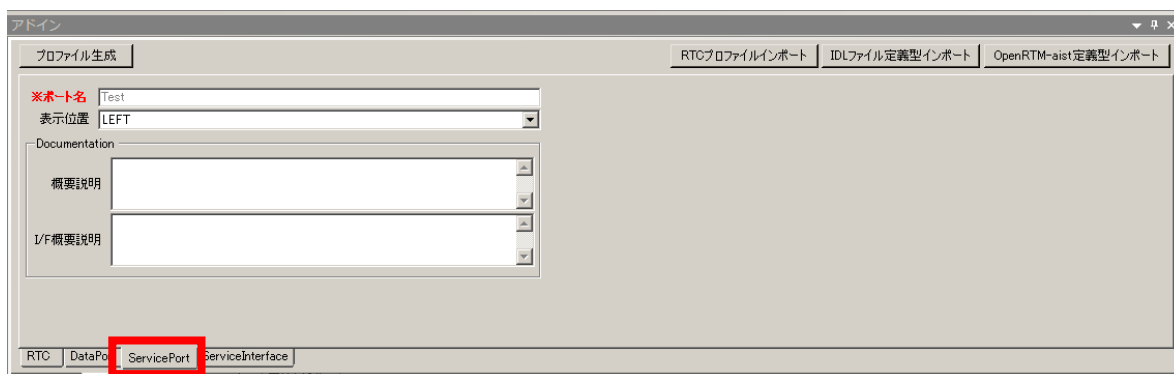


図 4.6.1 サービスポート情報の設定

「ポート名」につきましては、対象ポートの情報が表示され、本画面で変更する事はできません。「ポート名」を変更したい場合には、該当ポートの情報を変更してください。

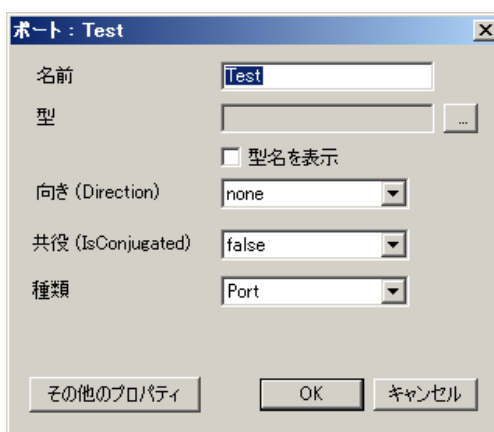


図 4.6.2 ポート情報の編集

また、その他の各種情報につきましては、どれもオプション扱いとなっておりますので、必要に応じて各種項目を設定してください。

## 4.7 サービスインターフェース情報の設定

RT コンポーネントのサービスインターフェースは、フロープロパティを持たない通常のポートに追加されたインターフェースとして定義します。内部ブロック図内で該当要素を選択すると、RT システムビルダー用画面内の「ServiceInterface」タブが自動的に選択されサービスインターフェースの各種情報が表示されます。

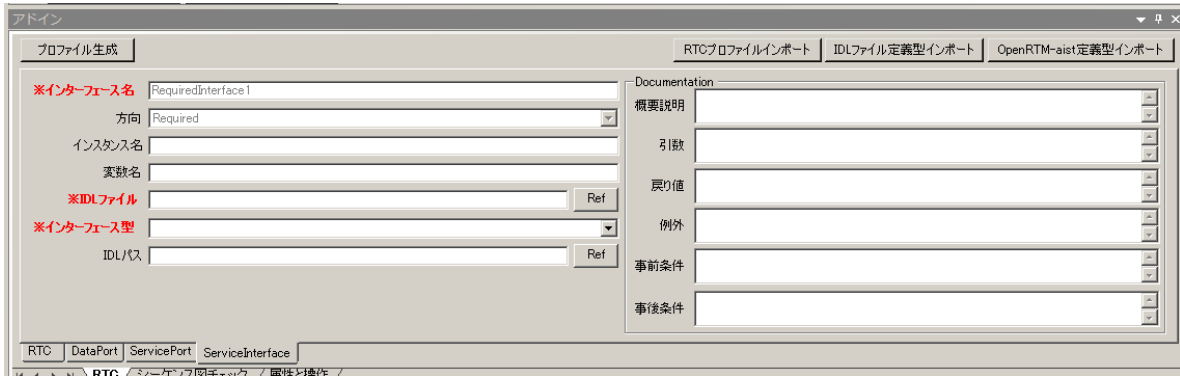


図 4.7.1 サービスインターフェース情報の設定

なお、「インターフェース型」を選択する際には、事前に「IDL ファイル」を設定してください。対象 IDL 内で定義されているインターフェース情報がドロップダウンリスト内に表示されますので、使用したいインターフェースを選択してください。

また、モデル中でサービスインターフェースを追加したい場合には、対象ポートを右クリックして表示されるコンテキストメニュー内から「子要素の追加」を選択し、「要求インターフェース」もしくは「提供インターフェース」を選択してください。

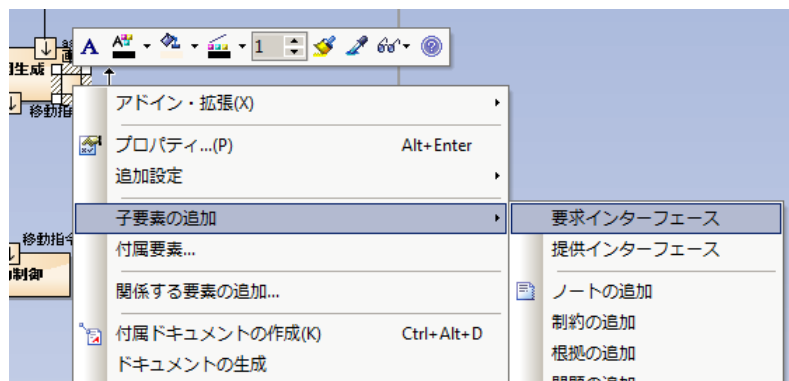


図 4.7.2 サービスインターフェースの追加

「ServiceInterface」タブの「インターフェース名」「方向」につきましては、対象インターフェースの情報が表示され、本画面で変更する事はできません。これらの情報を変更したい場合には、該当インターフェースの情報を編集してください。



図 4.7.3 サービスインターフェース情報の編集

## 5 RTC プロファイルの生成

本機能は、モデルに設定した情報から RTC プロファイルを生成するための機能です。モデル内に複数の RT コンポーネントが定義されている場合、全ての RT コンポーネントのプロファイルを生成します。

モデル内の全ての RT コンポーネントへの情報設定が完了後、RT システムビルダー用画面左上の「プロファイル生成」ボタンを選択してください。モデル内に含まれる RT コンポーネントが一覧表示されます。

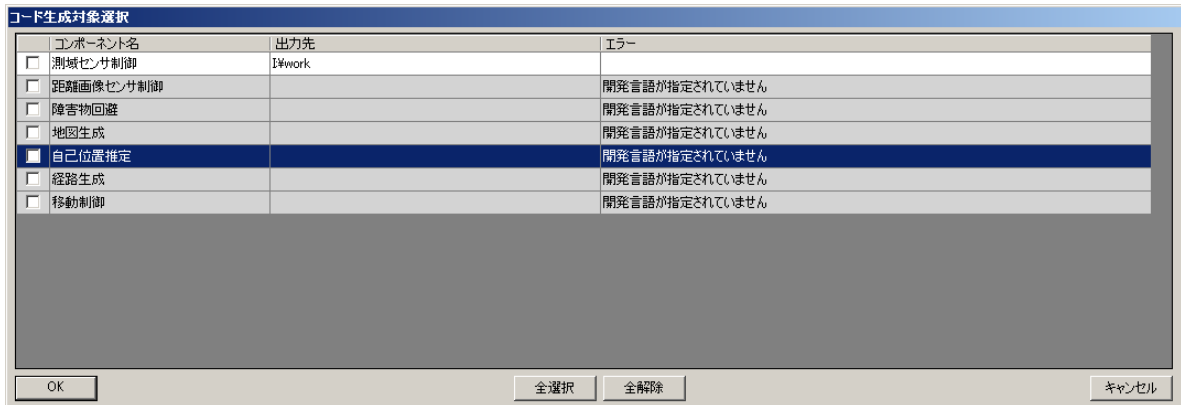


図 5.1.1 RTC プロファイルの生成

RTC プロファイルを生成する項目のチェックボックスを ON にして「OK」を選択すると、対象 RT コンポーネントのプロファイルが指定したディレクトリ内に出力されます。

なお、RT コンポーネントに必要な必須項目が指定されていなかったり、設定した情報に不備が存在する場合には、一覧表中でグレー表示され、チェックボックスは選択できないようになります。エラー内容が一番右の列に表示されますので、表示内容に従って設定情報の修正を行ってください。

生成した RTC プロファイルから、RT コンポーネントのコードを生成する方法、作成したコンポーネントを実行する方法などにつきましては、RTCBUILDER のマニュアルを参照してください。

また、SysML のポートの場合、「方向」として「inoout(入出力)」を指定することができますが、OpenRTM-aist ではこのような方向のデータポートを作成することができません。このため、モデル中のポートとして「inout」を指定した場合には、自動的に2つのポートが生成されます。この際、ポートの「ポート名」および「変数名」につきましては、元の名称に「In」「Out」の接尾語が追加されます(OpenRTM-aist では単一コンポーネント内に同一名称のポート、もしくは同一変数名のポートを生成することができないためです)。この場合、生成した RTC プロファイルを再度読み込んでも、元のモデルとは定義が異なってインポートされますのでご注意ください。

## 6 デフォルト値設定

RTシステムビルダーでは、RTコンポーネントの各種情報について、デフォルト値を予め設定することが可能です。設定したデフォルト値は、新規 RT コンポーネントの設計を行う際に、自動的に設定されます。

各種設定は、上部メニュー中の「アドイン・拡張」メニューから「RTSystemBuilder」→「設定」→「デフォルト値設定」を選択してください。

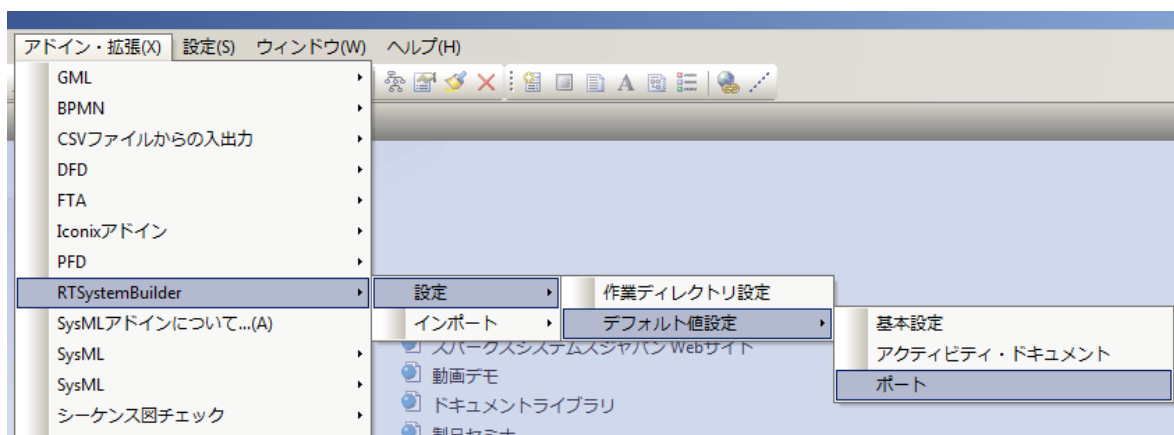


図 6.1.1 デフォルト値設定

### 6.1 基本設定

【基本設定デフォルト値設定】画面では、RTコンポーネント本体に関する情報およびコンフィギュレーションセットの情報に関するデフォルト値を設定する事が出来ます。ここで設定した値は、「RTC」タブ内、【RTC詳細情報】画面および【ConfigurationSet 詳細設定】画面で使用されます。

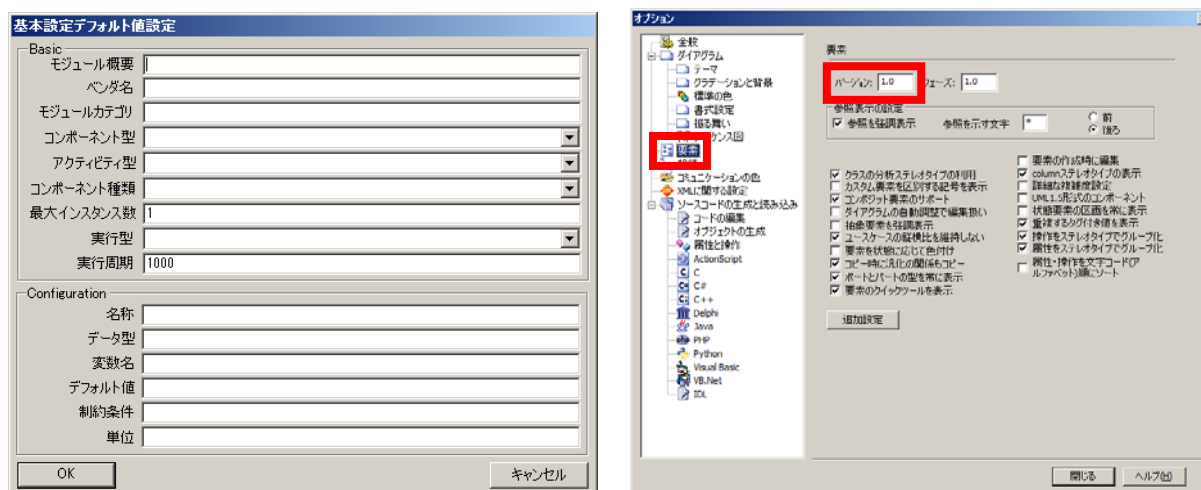


図 6.1.2 基本設定デフォルト値設定

なお、「モジュール名」につきましては、ブロックの名称を利用しているため、デフォルト値を設定する事はできません。また、「バージョン番号」のデフォルト値を変更したい場合には、上部メニューの「ツール」→「オプション」を選択し、表示された【オプション】画面で「要素」を選択し、「バージョン」欄の値を修正してください。

## 6.2 アクティビティ・ドキュメント設定

【アクティビティ・ドキュメントデフォルト値設定】画面では、RT コンポーネント本体のアクティビティ情報、ドキュメント情報のデフォルト値を設定することが可能です。ここで設定した値は、【アクティビティ設定】画面、【アクティビティ詳細設定画面】および【ドキュメント設定】画面で使用されます。

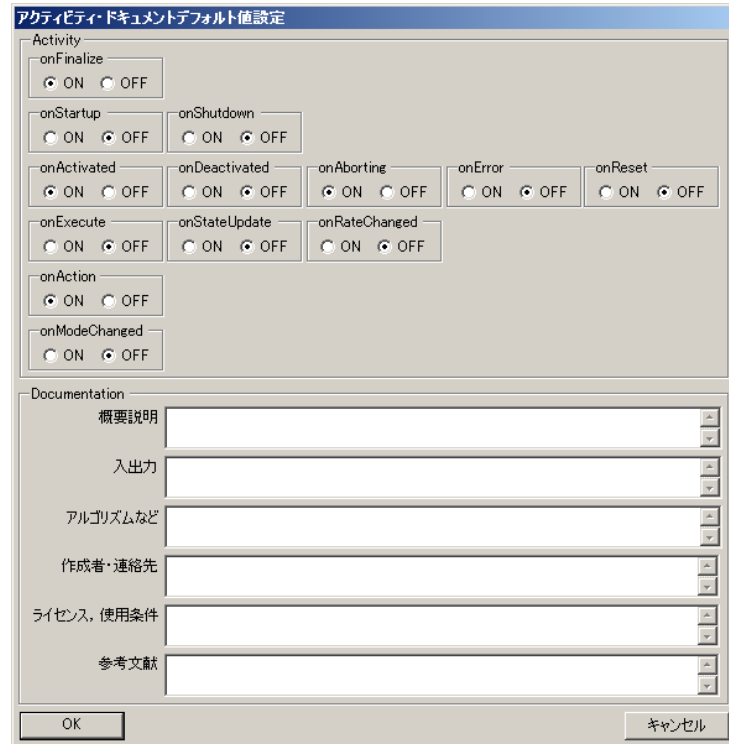


図 6.2.1 アクティビティ・ドキュメントデフォルト値設定

## 6.3 ポート設定

【ポートデフォルト値設定】画面では、データポート、サービスポートおよびサービスインターフェースのデフォルト値を設定することが可能です。ここで設定した値は、「DataPort」タブおよび「ServiceInterface」タブで使用されます。

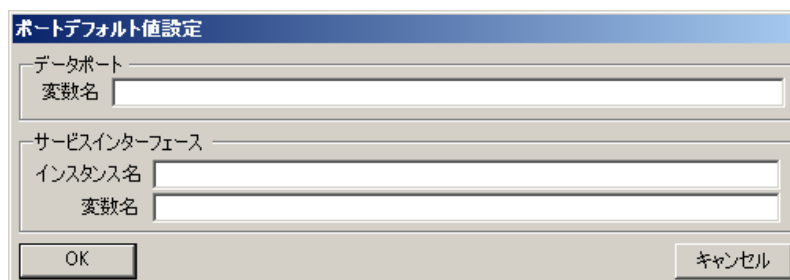


図 6.3.1 ポートデフォルト値設定

## 7 改版履歴

日付	バージョン	内容
2015/03/09	0.7.0	β 版向け新規作成
2015/08/02	0.7.1	プロフィール生成機能修正