

7コマ目 OpenRTM-aistの環境構築



OpenRTM-aistの概要



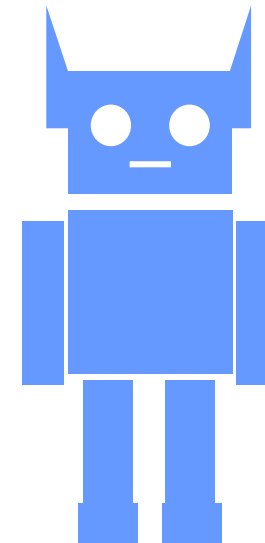
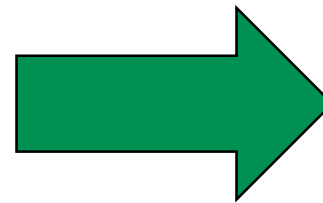
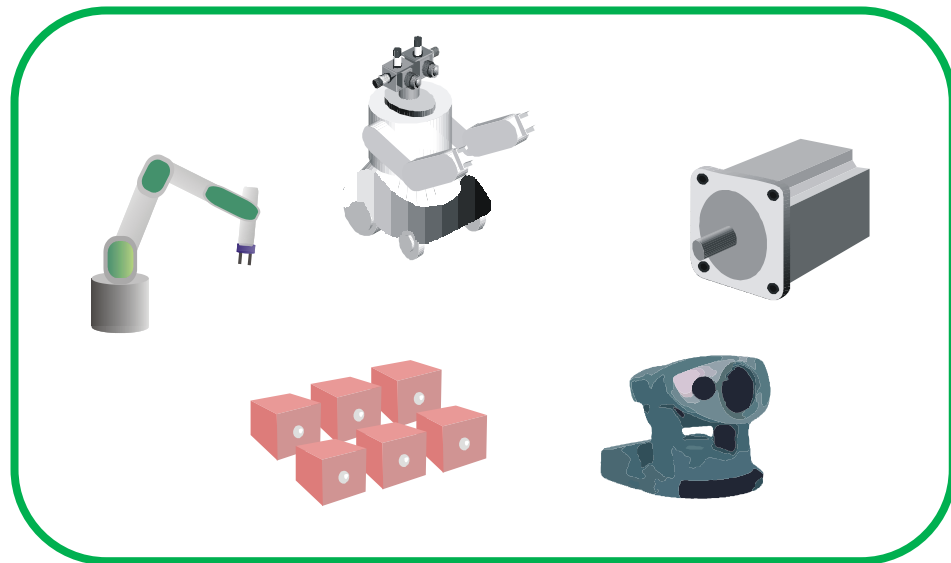
講義資料提供

- 国立研究開発法人産業技術総合研究所（産総研，AIST）
 - ロボットイノベーション研究センター
<https://unit.aist.go.jp/rirc/>



RTとは？

- RT = Robot Technology (\neq Real Time)
 - IT = Information Technology
- ロボット技術：単体のロボットだけではない
 - センサ，アクチュエータ，制御，アルゴリズムなどの集合体



RTMとは？

- RT-Middleware (RTM)
 - RT要素のインテグレーション(統合)のためのミドルウェア
- RT-Component (RTC)
 - RT-Middlewareにおけるソフトウェアの基本単位
- 本講習では、**産総研版RTミドルウェアOpenRTM-aistを使用したロボット制御プログラミング**を行う

従来のシステム



ジョイスティック



ロボットアーム



**互換性がある
インターフェース同士は接続可能**

従来のシステム



ジョイスティック



ロボットアーム



互換性がない
インターフェースは接続できない

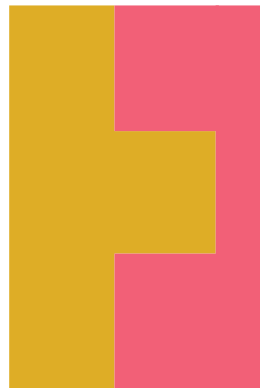
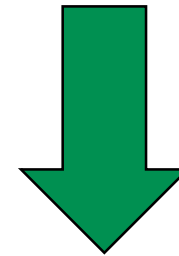
RTミドルウェアによるシステム



ジョイスティック



別々に作られたソフトウェア
モジュール同士をつなぐための
共通インターフェースを提供

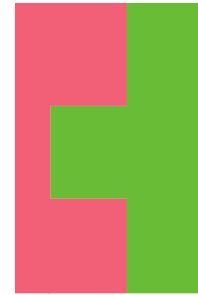


ソフトウェアの再利用性の向上により
RTシステム構築がより容易になる

RTミドルウェアによるシステム



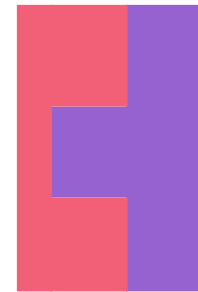
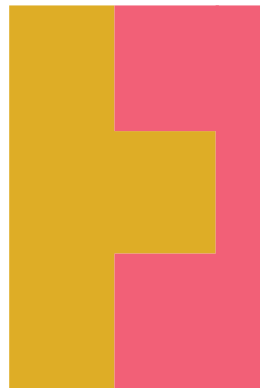
ジョイスティック



接続可能



ロボットアーム



接続可能



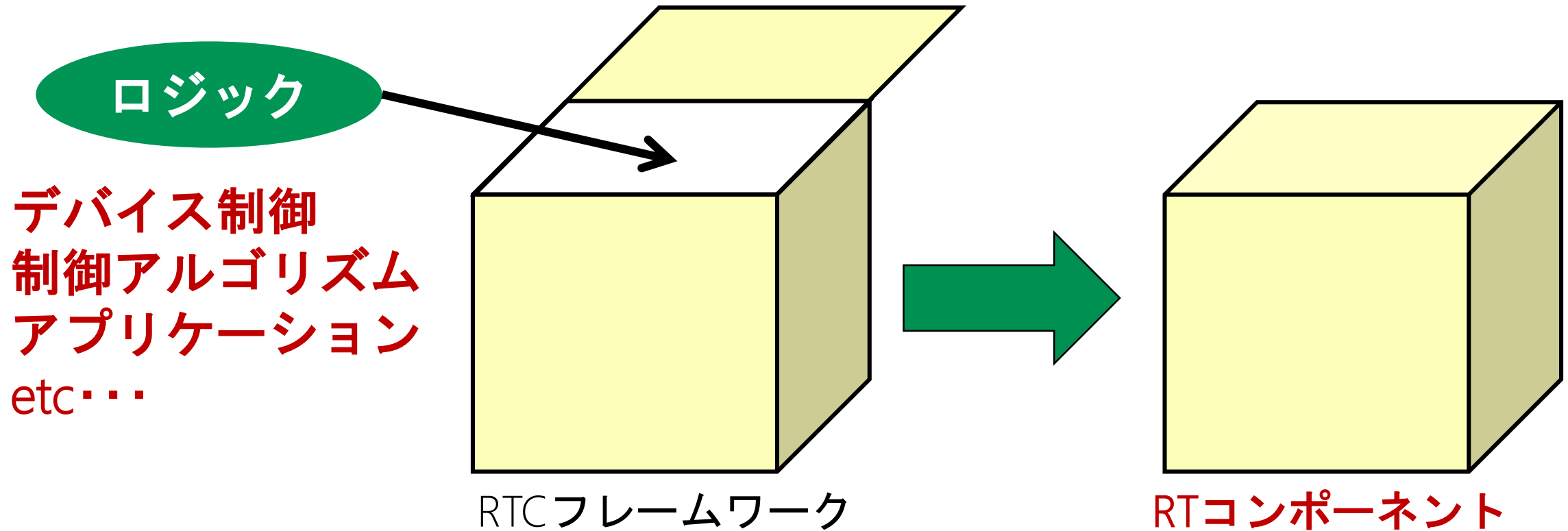
ロボットアーム

RTMの目的

- コスト問題
 - コンポーネントの再利用・組み合わせで，ロボットの低コスト化，開発期間の短縮，品質の向上
- 技術問題
 - システム開発者が最新技術を利用可能
- ニーズ問題
 - 多様なユーザがカスタマイズが容易
- **ロボットシステムインテグレーションによるイノベーション**

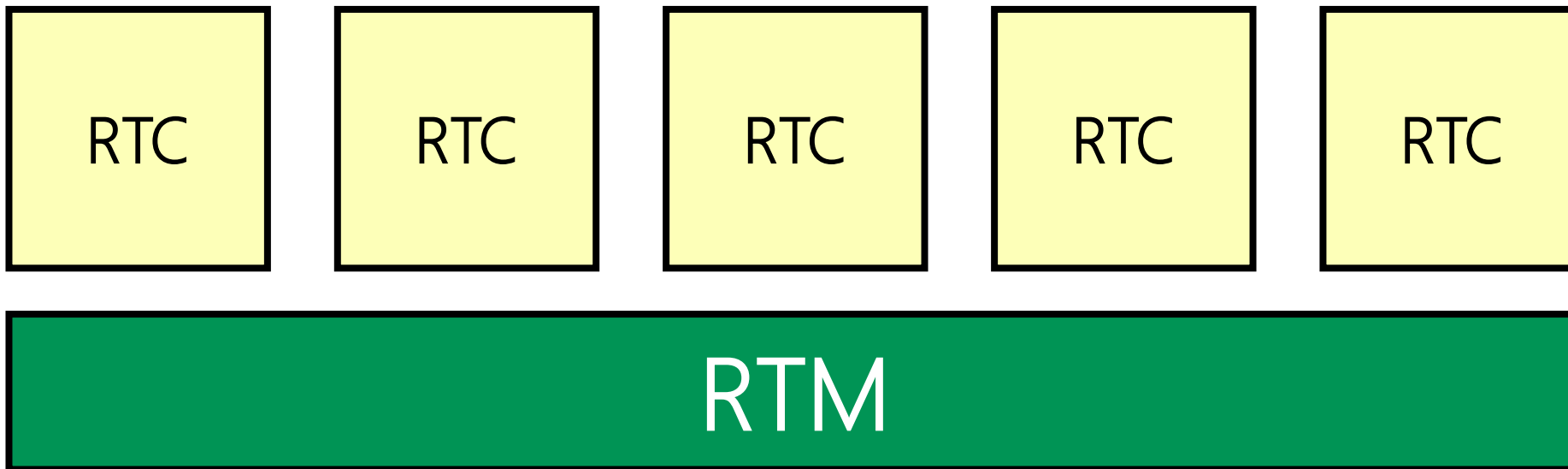
RTコンポーネント (RTC)

- RTC：ロジックをフレームワーク（箱）に入れたもの
 - フレームは自動的に生成されるため，ロジックだけを書けば良い

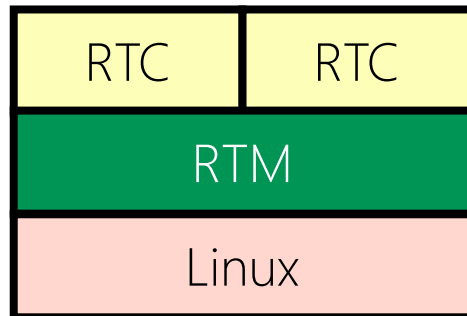


RTミドルウェア (RTM)

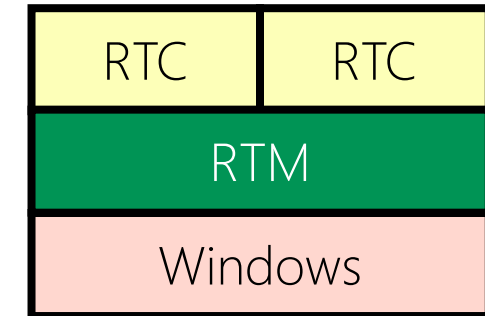
- RTM：RTCを動かす実行環境（OSのようなもの）
 - つまり，ミドルウェアのことを表す
 - RTCはネットワーク上に分散可能



RTミドルウェアによる分散システム

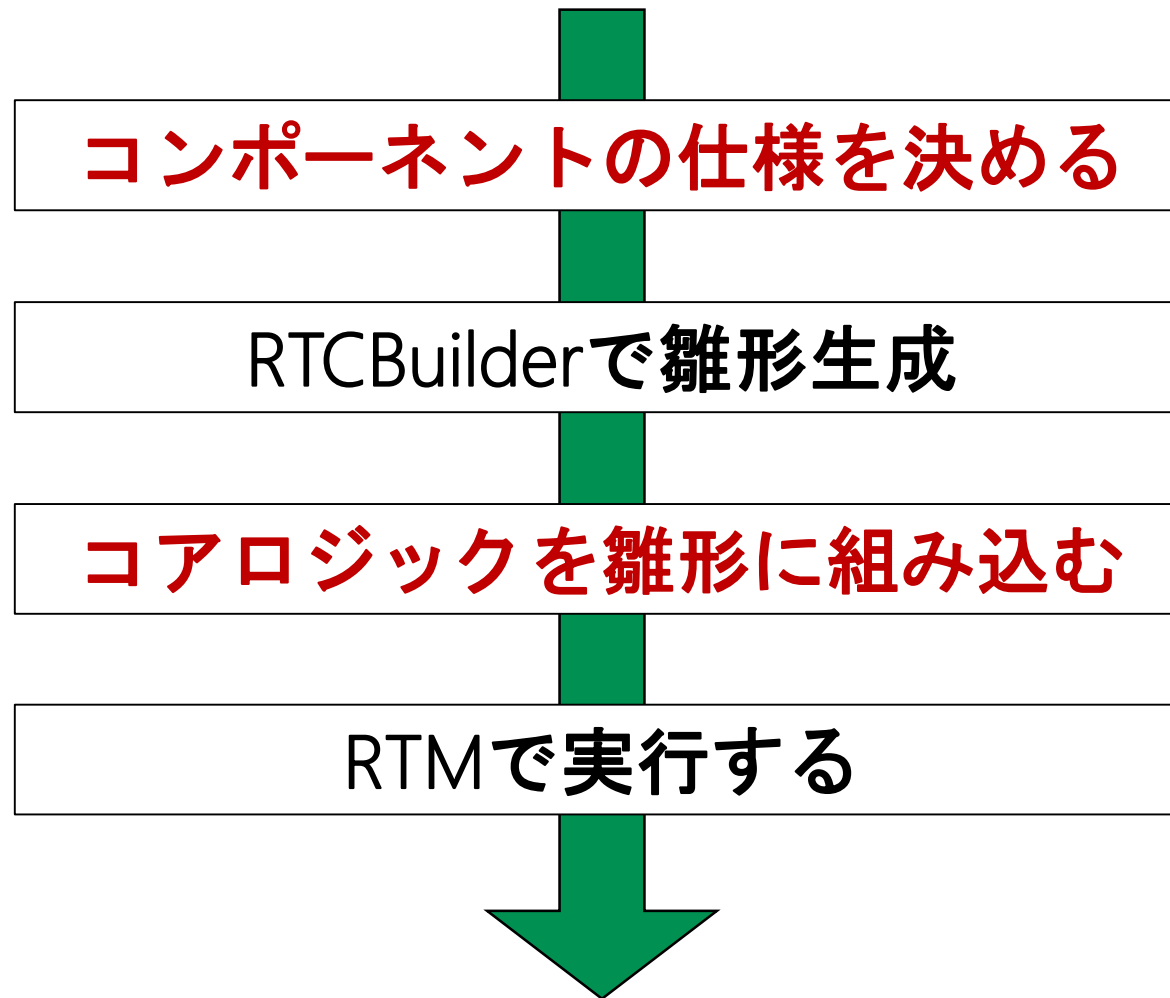


RTC同士の接続は
プログラム実行中に
動的に行える

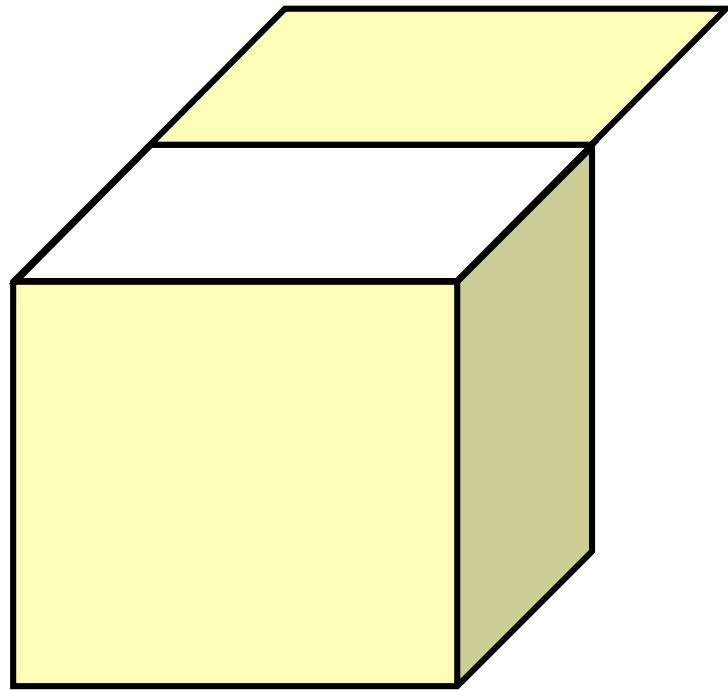


RTMを使うことで、ネットワーク上に分散するRTCを
OS・プログラム言語の壁を越えて接続できる

OpenRTM-aistを使った開発の流れ



フレームワークとコアロジック

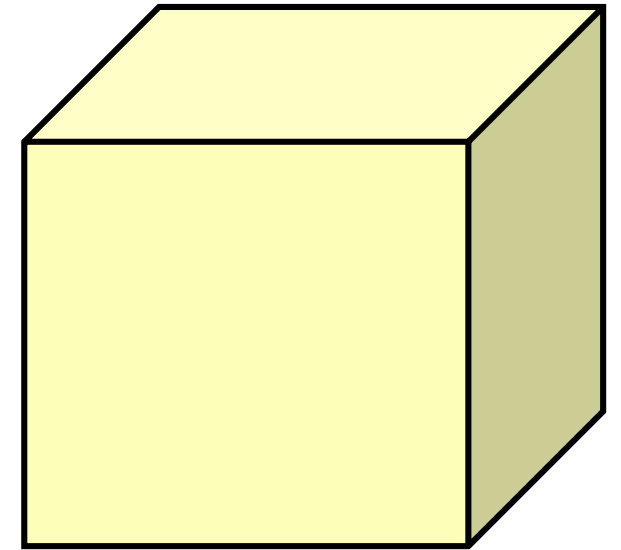


RTCフレームワーク

+



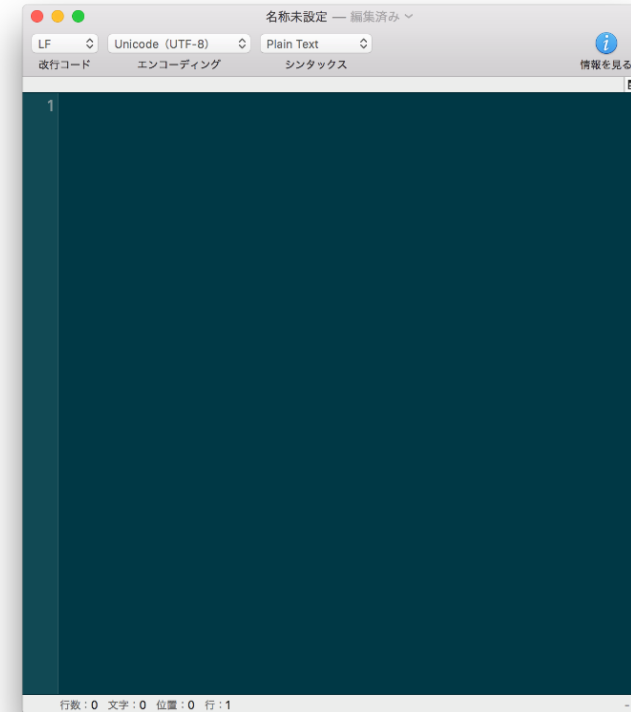
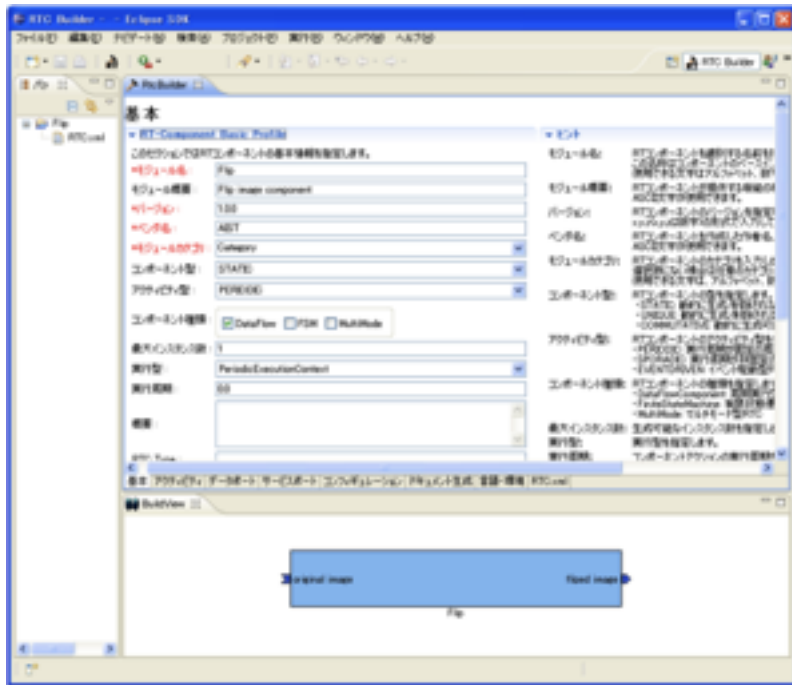
=



RTコンポーネント

RTCフレームワーク + コアロジック = RTコンポーネント

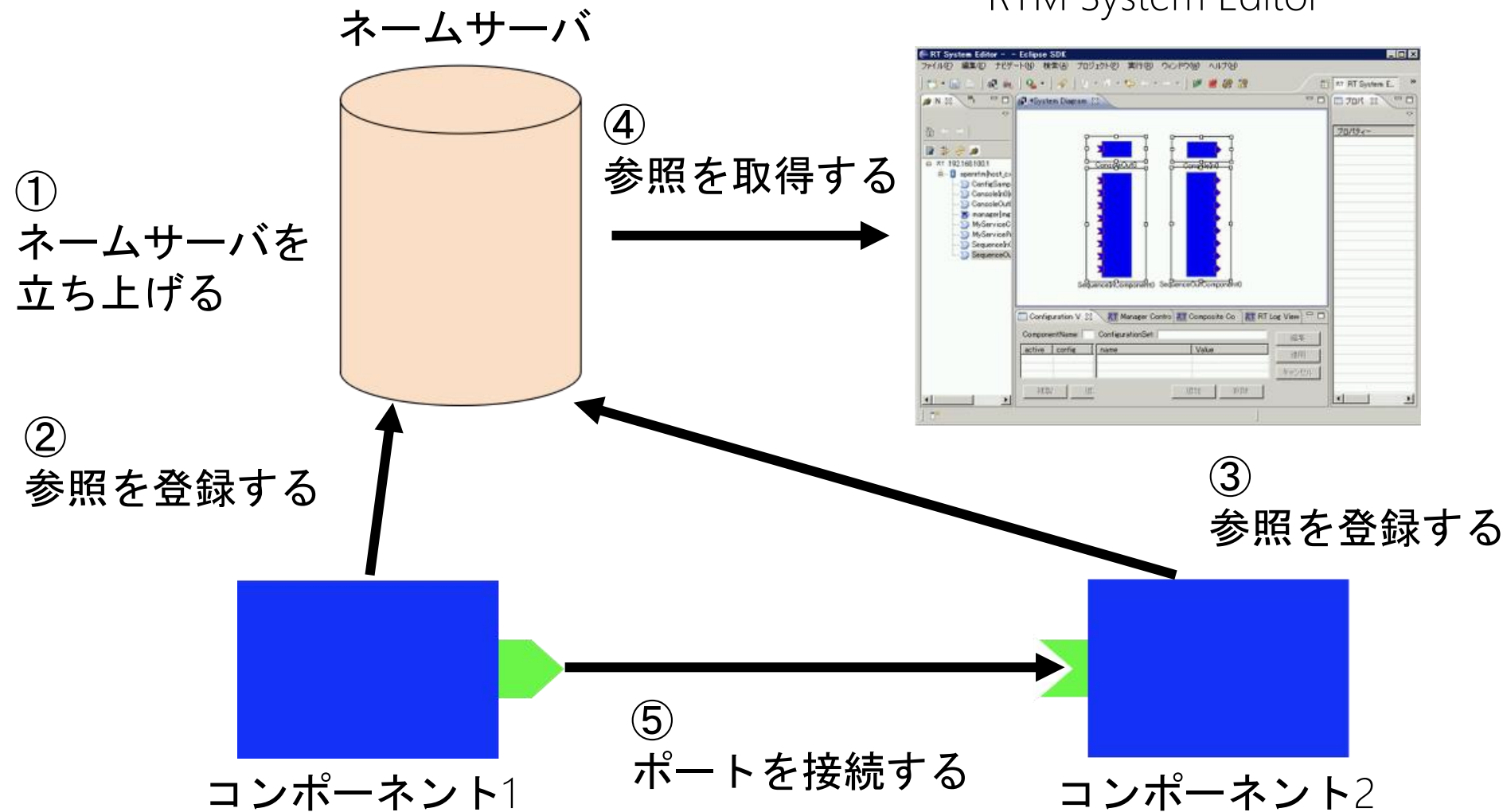
RTC作成例 | 本講習会の場合



RTCBuilderで、
コンポーネントの仕様の入力し、
テンプレートコードを生成する

テキストエディタで、
テンプレートコードに
ロジックを組み込む

動作シーケンス



OpenRTM-aistの開発環境構築

受講者PC側



インストールソフトウェアリスト

- Python2.7.10
- OpenRTM-aist 1.1.2-RELEASE
- Bonjour
- Pygame
- サクラエディタ

Python2.7.10

- インストーラを以下のURLからダウンロード

<https://www.python.org/ftp/python/2.7.10/python-2.7.10.msi>

- ダウンロードしたインストーラをクリックしてインストール

Python2.7.10

- [Add python.exe to Path] を [Will be installed on local hard drive] に設定
 - パスが自動的に設定される



OpenRTM-aist 1.1.2-RELEASE

- インストーラを以下のURLからダウンロード

http://tmp.openrtm.org/pub/Windows/OpenRTM-aist/1.1/OpenRTM-aist-1.1.2-RELEASE_x86.msi

- ダウンロードしたインストーラをクリックしてインストール

Bonjourインストール

- iTunesをインストールすると，同時にBonjourがインストールされる

<https://www.apple.com/jp/itunes/download/>

Pygameインストール

- インストーラを以下のURLからダウンロード
※自分のPCにあった32bitのバージョンを選択する

<http://pygame.org/download.shtml>

- ダウンロードしたインストーラをクリックしてインストール

サクラエディタのインストール

- インストーラを以下のURLからダウンロード
 - 最新版をダウンロードする

<https://sakura-editor.github.io/download.html>

- ダウンロードしたインストーラをクリックしてインストール

OpenRTM-aistの開発環境構築 Raspberry Pi側



コマンドリストのダウンロード

- インストールに必要なコマンドを以下URLのzip内ファイルに記載
<https://rtc-fukushima.jp/wp/wp-content/uploads/2018/11/07Install.zip>
- 一部コマンドが長くて一行に収まらないものがあるので、
その場合、テキストからコピーして使用してください

Raspberry Piにインストール

- リポジトリのアップデート
 - 端末に以下を入力

```
$ sudo apt update
```

- OpenRTM-aistのインストール
 - sources.listを修正
 - 端末に以下を1行で入力 (>> の前後に半角スペースを入れる)

```
$ sudo su -c 'echo "deb http://tmp.openrtm.org/pub/Linux/raspbian/ stretch main" >> /etc/apt/sources.list'
```

公開キーの登録

- 端末に以下を入力

```
$ sudo apt install dirmngr
```

```
$ gpg --keyserver pgpkeys.mit.edu --recv-key 4BCE106E087AFAC0
```

```
$ gpg -a --export 4BCE106E087AFAC0 | sudo apt-key add -
```

必要ソフトウェアをインストール

- OpenRTM-aistのインストール
 - 端末に以下を入力

```
$ sudo apt-get update
$ sudo apt-get -y --force-yes install gcc g++ make uuid-dev
$ sudo apt-get -y --force-yes install libomniorb4-dev omniidl
$ sudo apt-get -y --force-yes install omniorb-nameserver
$ sudo apt-get -y --force-yes install openrtm-aist openrtm-aist-dev
$ sudo apt-get -y --force-yes install openrtm-aist-example
$ sudo apt-get -y --force-yes install openrtm-aist-python
$ sudo apt-get -y --force-yes install openrtm-aist-python-example
$ sudo apt-get -y --force-yes install python-omniorb
```