



ROBOT SYSTEM DESIGN
LABORATORY

wasanbonのためのシステム構築ツール RTSBuilder

名城大学

福田 真斗

加藤 美沙

大原 賢一

開発背景

近年、ロボットの開発が加速しており、
ロボット用ミドルウェアの利用が促進されている。



OpenRTM

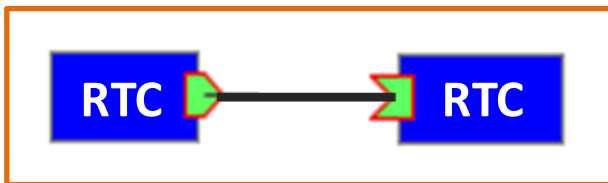


ROS

OpenRTM

独立性や再利用性の高い
モジュールを容易に作成可能

RTS



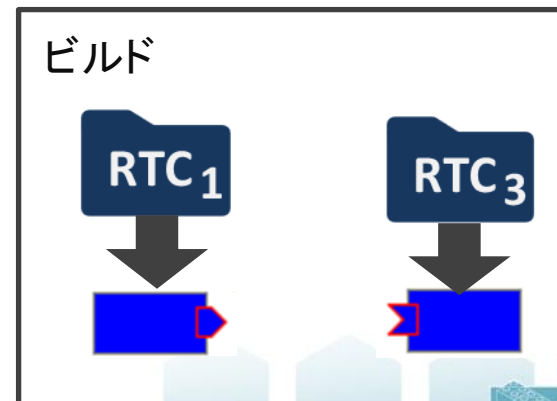
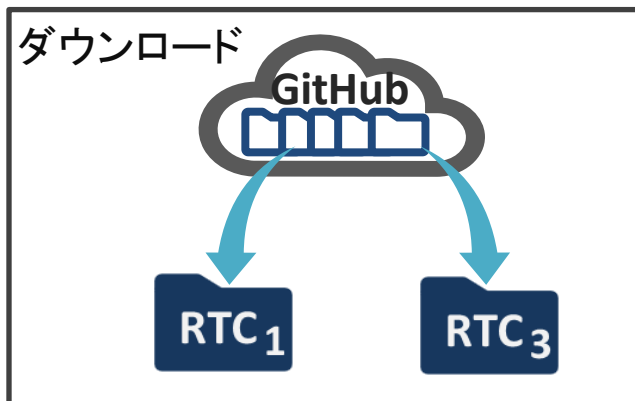
RTC (RTコンポーネント)

- ロボットの機能や制御を提供する
コンポーネントの基本単位

RTS (RTシステム)

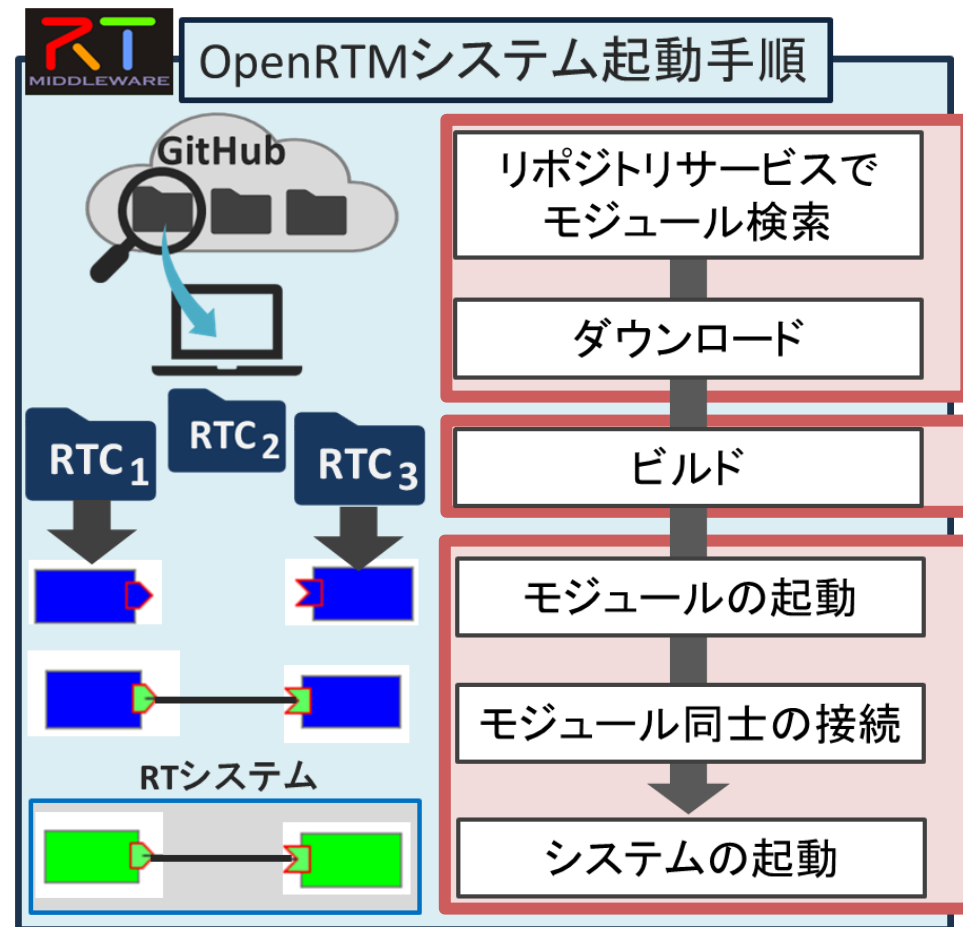
- RTCを組み合わせたシステム

● システムを動かす際、各RTCに対して実行管理を行う ▶ 行程が多くなる



Wasanbonの概要

- ▶ OpenRTMをシステム単位で効率良く利用できるフレームワーク



Wasanbonのリポジトリ管理方法

- リポジトリのURLを使用して, システムの管理ができる.

- RTC, RTSのリポジトリ管理

GitHub上のRTS内のRTCは, リポジトリを指定するだけでよい

- 簡単なモジュールの収集

GitHub上で探す必要なく, 登録リスト(binder)からダウンロード

▶ wasanbon binder にシステムを登録する必要がある

wasanbon binder

システムの基本的な情報を管理

```
1 MarkerPoseEstimation:
2   description: Pose estimation system with ArUco marker
3   type: git
4   url: https://github.com/rsdlab/MarkerPoseEstimation.git
5   platform: win, ubuntu
6 MobileRobotControl:
7   description: Mobile robot control system with Joystick
8   type: git
9   url: https://github.com/rsdlab/MobileRobotControl.git
10  platform: win, ubuntu
```

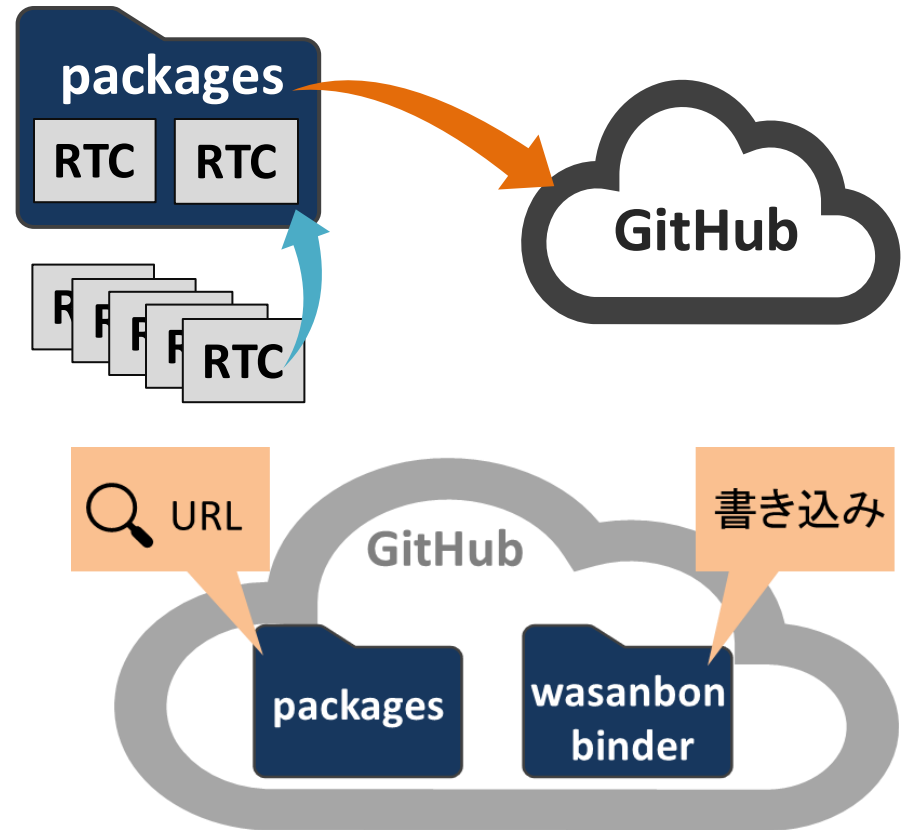
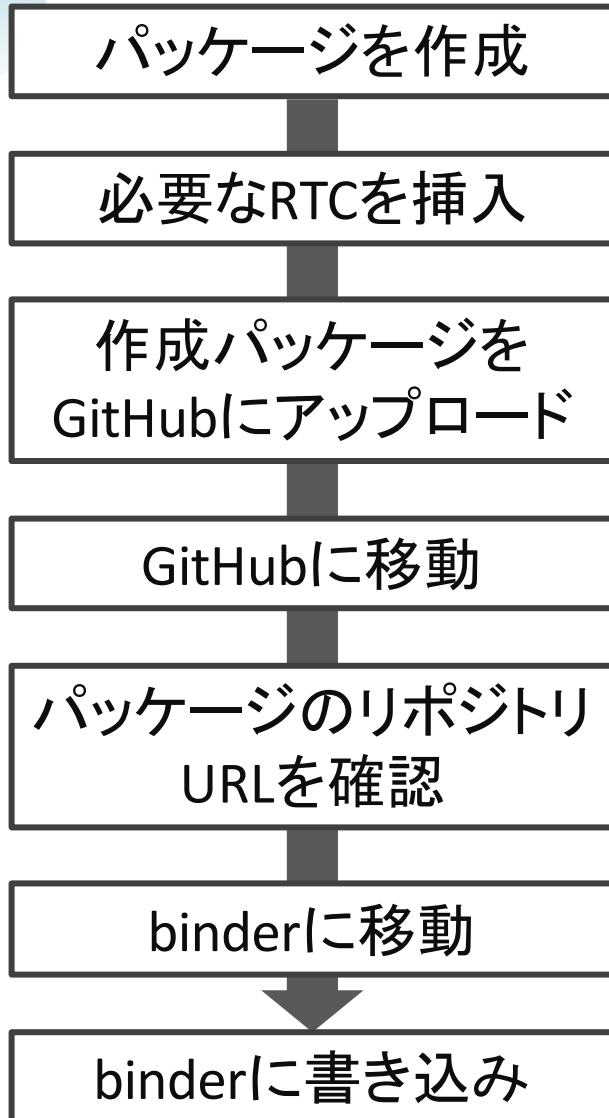
システム名

概要

リポジトリ

使用可能プラットフォーム

システムをwasanbon binderに登録



➡ 行程が多いため、スムーズにシステムをwasanbon binderに登録できるツールが必要

ツールに求められる仕様検討

パッケージの作成手順の簡易化

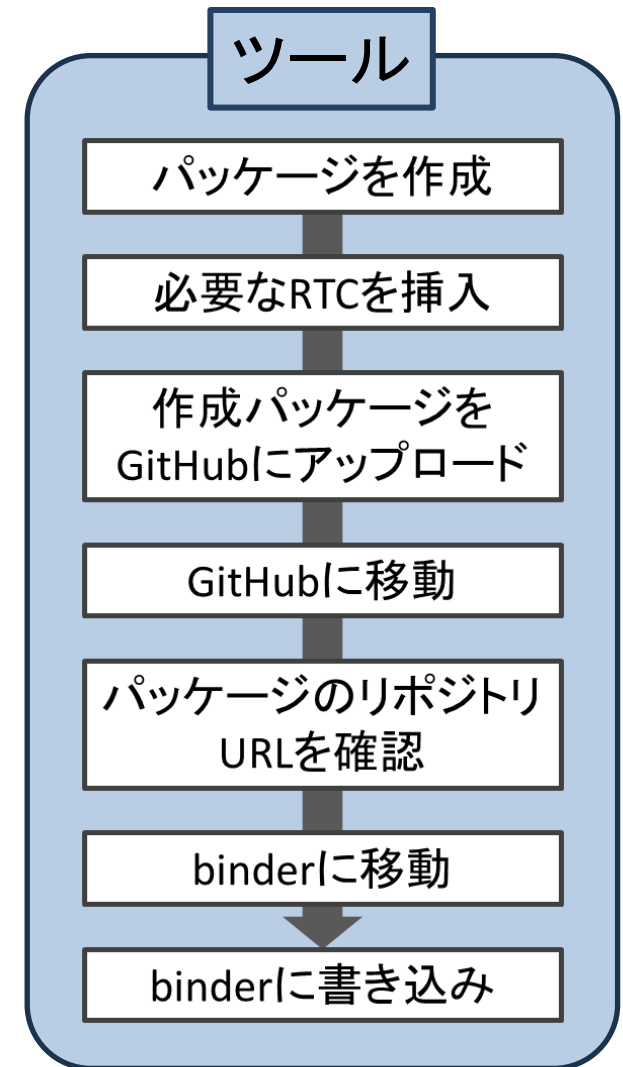
- ・ パッケージを自動的に用意する
- ・ RTCを探しやすくする
- ・ システムの中身を見やすくする

GitHubやbinderに作用する

- ・ GitHubと相互に動作する
- ・ wasanbon binderと相互に動作する
- ・ パッケージのリポジトリ位置の選択
- ・ binderのリポジトリ位置の選択
- ・ アップロード情報の設定

既存手順は様々な場所での作業が多い

- ・ 各作業間の移動を減らす

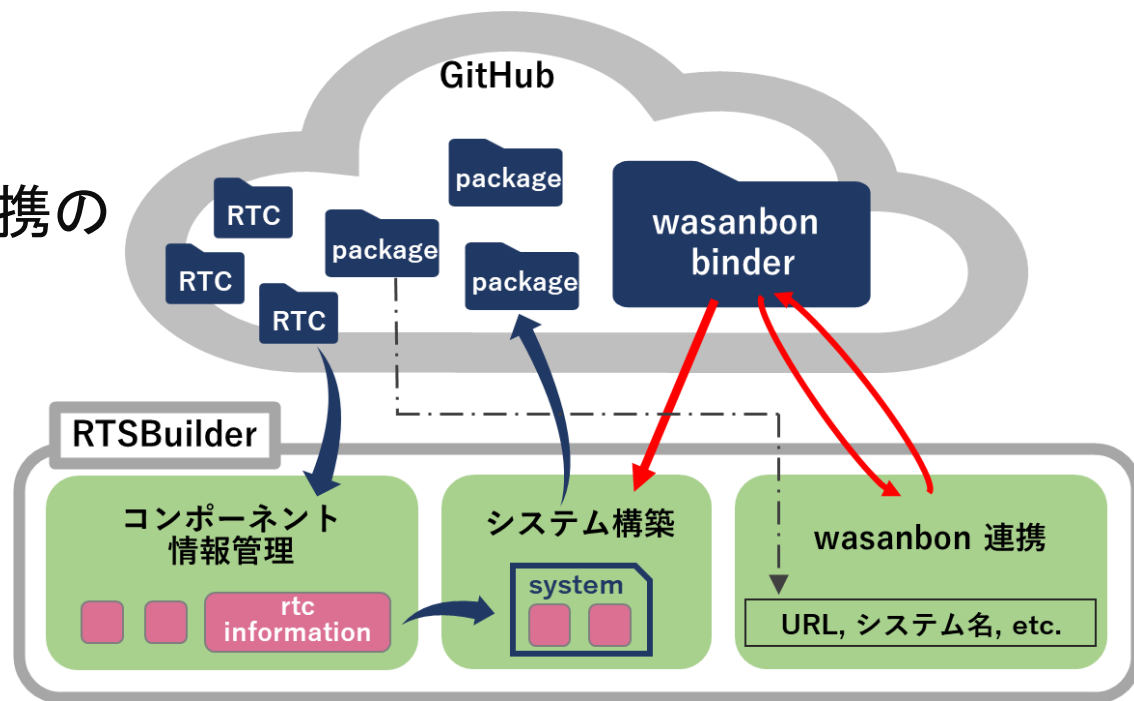
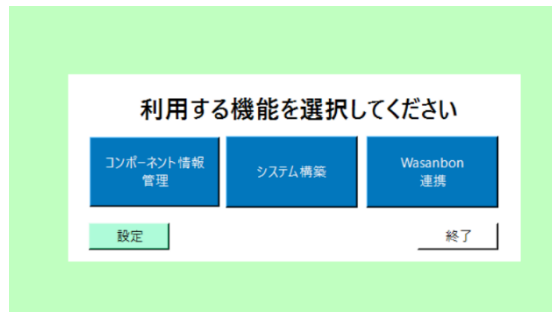


RTSystemBuilder

wasanbon利用を前提にしたシステム構築支援ツール

構成

コンポーネント情報管理,
システム構築, wasanbon連携の
3つの要素で構成



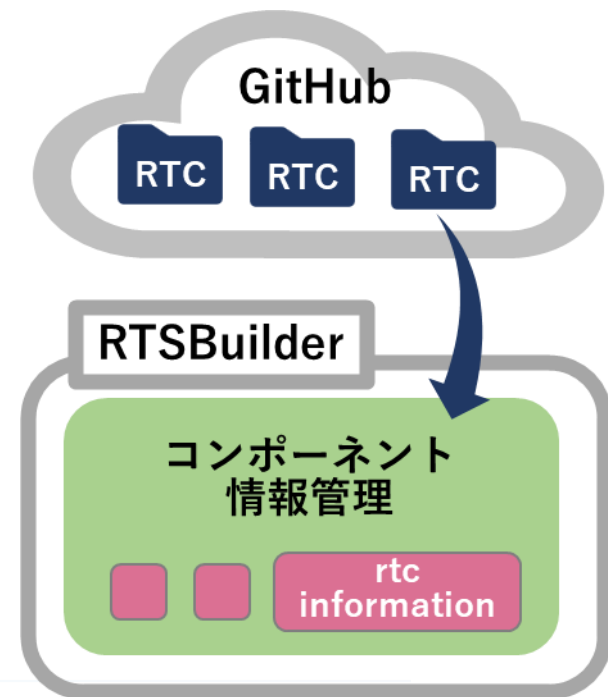
機能

- ・コンポーネント情報確認
- ・システムパッケージ作成
- ・システムパッケージをGitHubに登録
- ・wasanbon binderに登録

パッケージ作成から
binder登録をスムーズに

コンポーネント情報管理

- ・ システム構築に必要なRTCをリポジトリ上から本ツールに追加
(リポジトリURLを指定する)
- ・ RTCの情報を確認
コンポーネント名, リポジトリ, データポート, etc



コンポーネント情報管理

コンポーネント一覧

検索 全件取得

追加 更新 削除

登録名	コンポーネント名	ベンダ名	カテゴリ	実装言語	リポジトリ	コメント
exam_module02	exam_module02	VenderName	Category	Python	https://github.com/rsdlab/exam_module02	
exam_module03	exam_module03	VenderName	Category	Python	https://github.com/rsdlab/exam_module03	
exam_module01	exam_module01	VenderName	Category	Python	https://github.com/rsdlab/exam_module01	

【データポート】

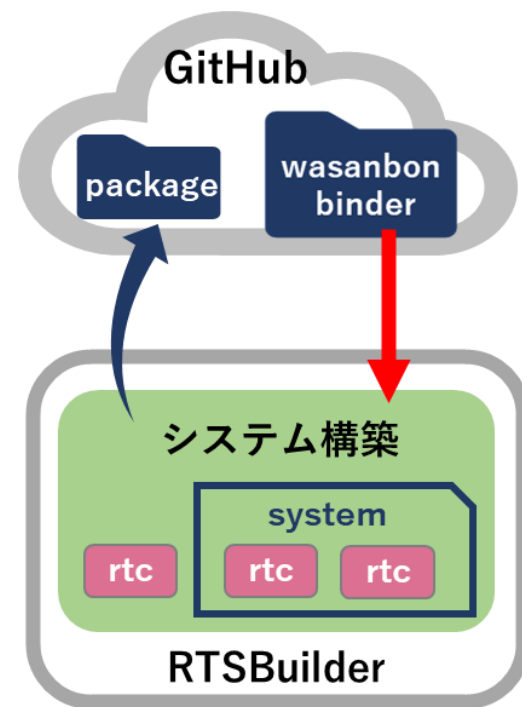
[InPort]
- data_in (RTC::TimedShort)
[OutPort]
- data_out (RTC::TimedShort)

【サービスポート】

閉じる

システム構築

- 複数RTCを選択して新規システムを作成
- 新規システムをパッケージとしてGitHubにアップロード
- wasanbon binderに登録されている既存システムの編集
(RTCの削除, 追加, etc...)



システム構築

システム構成

ベースリポジトリ URI: システム名: 新規作成 情報取得 情報設定 情報クリア 概要更新 削除

コンポーネント名	概要	URL
exam_module01		https://github.com/rsdlab/exam_module01.git
exam_module02		https://github.com/rsdlab/exam_module02.git
exam_module03		https://github.com/rsdlab/exam_module03.git

コンポーネント一覧

検索 全件取得 対象システムに登録

登録名	コンポーネント名	ベンダ名	カテゴリ	実装言語	リポジトリ	コメント
exam_module02	exam_module02	VenderName	Category	Python	https://github.com/rsdlab/exam_module02	
exam_module03	exam_module03	VenderName	Category	Python	https://github.com/rsdlab/exam_module03	
exam_module01	exam_module01	VenderName	Category	Python	https://github.com/rsdlab/exam_module01	

【データポート】

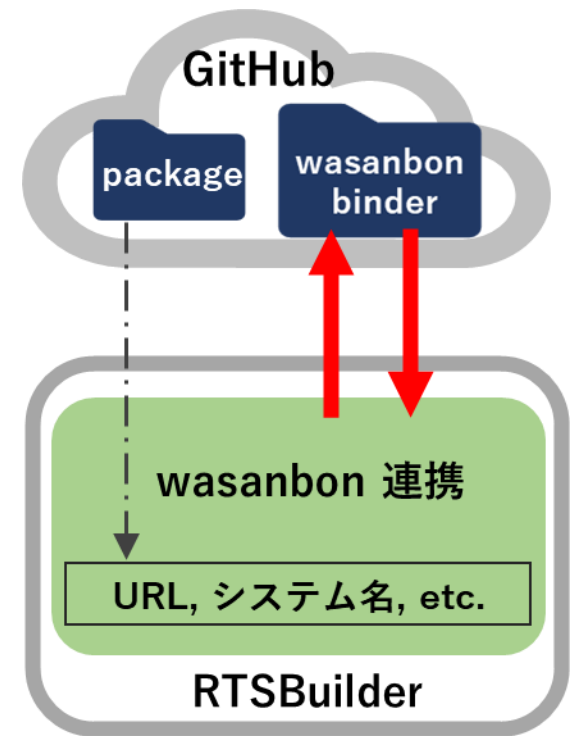
[InPort]
- data_in (RTC:TimedShort)
[OutPort]
- data_out (RTC:TimedShort)

【サービスポート】

終了

wasanbon連携

- リポジトリ上のシステムを wasanbon binder に登録
- wasanbon binder の編集
(システムの削除, 概要の変更, etc...)



Wasanbon連携

登録済みシステム

WasanbonリポジトリURI : 情報取得 情報設定 追加 更新 削除

システム名	概要	タイプ	URL	プラットフォーム
MarkerPoseEstimation	Pose estimation system with ArUco marker	git	https://github.com/rsdlab/MarkerPoseEstimation.git	win, ubuntu
MobileRobotControl	Mobile robot control system with Joystick	git	https://github.com/rsdlab/MobileRobotControl.git	win, ubuntu

閉じる

新規システム登録手順 (RTSystemBuilder)

システムの作成

- ① 新規システムのベースリポジトリURIと名前を設定する。
「新規作成」ボタンを押す。
- ② 「コンポーネント一覧」で, RTCを表示する。
追加したいRTCを選択し, 「対象システムに登録」ボタンを押す。

システム構築

システム構成

ベースリポジトリURI: システム名:

コンポーネント名	概要	URL
exam_module01		https://github.com/rsdlab/exam_module01.git
exam_module02		https://github.com/rsdlab/exam_module02.git
exam_module03		https://github.com/rsdlab/exam_module03.git

コンポーネント一覧

登録名	コンポーネント名	ベンダ名	カテゴリ	実装言語	リポジトリ	コメント
exam_module02	exam_module02	VenderName	Category	Python	https://github.com/rsdlab/exam_module02	
exam_module03	exam_module03	VenderName	Category	Python	https://github.com/rsdlab/exam_module03	
exam_module01	exam_module01	VenderName	Category	Python	https://github.com/rsdlab/exam_module01	

【データポート】

[InPort]
- data_in (RTC::TimedShort)
[OutPort]
- data_out (RTC::TimedShort)

【サービスポート】

終了

新規システム登録手順 (RTSystemBuilder)

GitHubにアップロード

- ③ 「情報設定」ボタンを押す.
- ④ 「システム情報設定」画面で, システム説明や
コミットメッセージを入力する. (GitHubに反映される)

システム構築

システム構築

システム構成

ベースリポジトリ URI : システム名 :

コンポーネント名	概要	URL
exam_module01		https://github.com/rsdlab/exam_module01.git
exam_module02		https://github.com/rsdlab/exam_module02.git
exam_module03		https://github.com/rsdlab/exam_module03.git

コンポーネント一覧

登録名	コンポーネント名	ベンダ名
exam_module02	exam_module02	VenderNam
exam_module03	exam_module03	VenderNam
exam_module01	exam_module01	VenderNam

【データポート】

[InPort]
- data_in (RTC::TimedShort)
[OutPort]
- data_out (RTC::TimedShort)

Component Database

システム情報設定

システム説明 :

ネームサーバー :

コミットメッセージ :

対象システムに登録

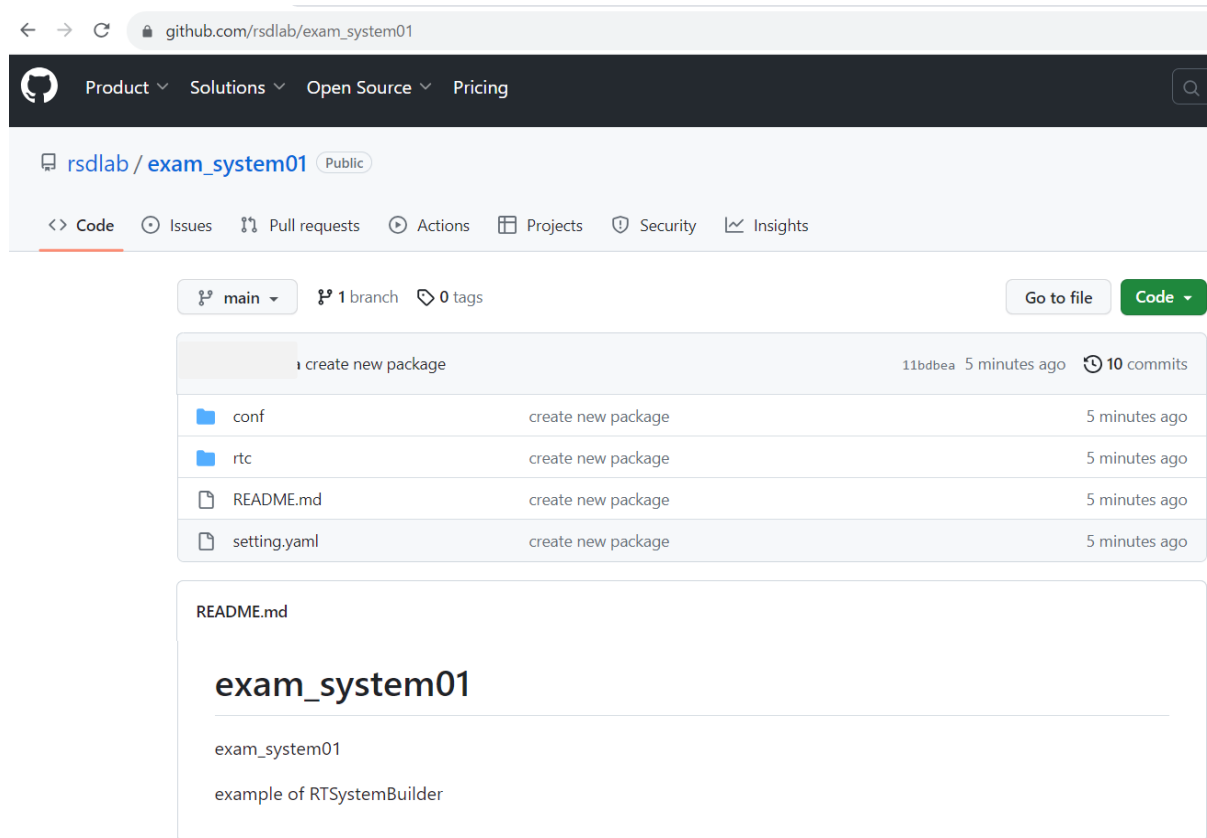
	コメント
b/exam_module02	
b/exam_module03	
b/exam_module01	

終了

新規システム登録手順 (RTSystemBuilder)

⑤ 「OK」ボタンを押す。

新規システムのパッケージがGitHubに登録される。



新規システム登録手順 (RTSystemBuilder)

wasanbon binder登録

- ⑥ 「システム追加」画面が表示される。
（「システム名」と「URL」は記入されている）
- ⑦ 概要とプラットフォームを記入し、「OK」ボタンを押す。

システム追加

WasanbonリポジトリURI :

システム名 :

概要 :

URL :

プラットフォーム : ☒ Windows ☐ macOS ☒ Ubuntu

新規システム登録手順 (RTSystemBuilder)

- ⑧ wasanbon連携画面が表示され、システム情報が追加される。
「情報設定」ボタンを押す。
- ⑨ コミットメッセージ画面が表示される。
コミットメッセージを入力し、「OK」ボタンを押す。

Wasanbon連携

登録済みシステム

WasanbonリポジトリURI : 情報取得 情報設定 追加 更新 削除

システム名	概要	タイプ	URL	プラットフォーム
MarkerPoseEstimation	Pose estimation system with ArUco marker	git	https://github.com/rsdlab/MarkerPoseEstimation.git	win, ubuntu
MobileRobotControl	Mobile robot control system with Joystick	git	https://github.com/rsdlab/MobileRobotControl.git	win, ubuntu
exam_system01	example of RTSystemBuilder	git	https://github.com/rsdlab/exam_system01.git	win, ubuntu

Component Database

コミットメッセージ

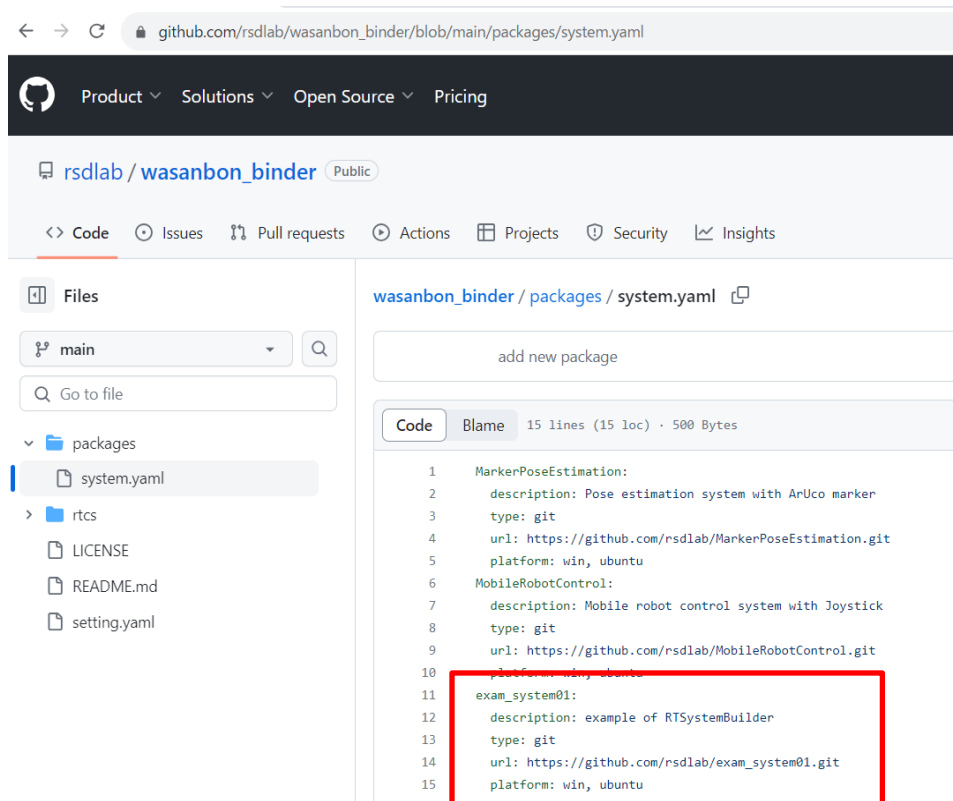
OK キャンセル

閉じる



新規システム登録手順 (RTSystemBuilder)

wasanbon binderに登録される.



The screenshot shows the GitHub repository `rsdlab/wasanbon_binder` at the `main` branch. The file `packages/system.yaml` is selected, showing its content. The file defines three packages: `MarkerPoseEstimation`, `MobileRobotControl`, and `exam_system01`. The `exam_system01` package is highlighted with a red box.

```
1 MarkerPoseEstimation:
2   description: Pose estimation system with ArUco marker
3   type: git
4   url: https://github.com/rsdlab/MarkerPoseEstimation.git
5   platform: win, ubuntu
6 MobileRobotControl:
7   description: Mobile robot control system with Joystick
8   type: git
9   url: https://github.com/rsdlab/MobileRobotControl.git
10  platform: win, ubuntu
11 exam_system01:
12   description: example of RTSystemBuilder
13   type: git
14   url: https://github.com/rsdlab/exam_system01.git
15   platform: win, ubuntu
```

 **RTSystemBuilder**を使うことで, 同ツール上でスムーズに
パッケージ作成からwasanbon binder登録が可能に



まとめ

RTSystemBuilder

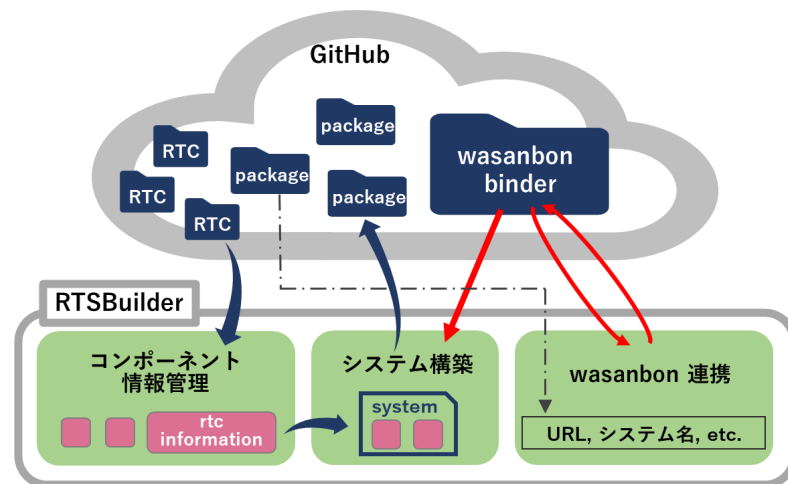
～wasanbonを前提としたシステム構築支援ツール～

パッケージの作成

GitHubにアップロード

wasanbon binderに登録

スムーズに



GitHub URL : <https://github.com/RTSystemDesigner/RTSystemBuilder>

課題 : RTC同士の接続は手動で行わなければならない。
今後, この機能について拡張を検討している。

謝辞: 本研究は, JST(ムーンショット型研究開発事業)
 Grant番号(JPMJMS2031)の支援を受けたものです。