

市場化 P J 最終成果報告会

2020年1月29日

ロボット活用型市場化適用技術開発プロジェクト
ソフトウェアコンソーシアム

東京大学

岡田 慧

ロボットのソフトウェアプラットフォーム

1. ハードウェアが購入できること,
2. ソフトウェアが簡単に導入できること,
3. 他の人も使っていること.

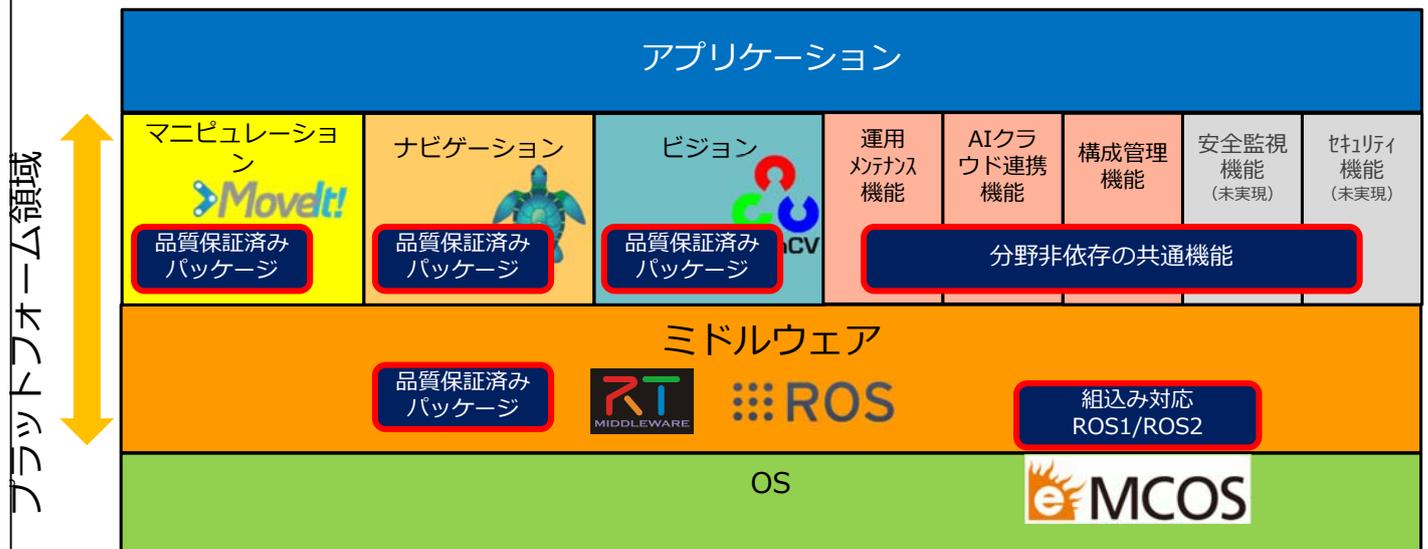
これがプラットフォームロボットのスタート地点

あとは、これをいかに多くの人に使ってもらって、
ロボット未活用領域を開拓.

金の切れ目が縁の切れ目にならないように、プロ
ジェクトが終わっても使い続けられるように

本プロジェクトにおける成果物

機能要件に関する成果物



非機能要件に関する成果物

- ・テスト・検証に関するエビデンス
- ・教育・コミュニティー
- ・モジュールの性能指標
- ・安全規格とのマッチングに関するガイドライン
- ・ドキュメンテーション
- ・ライセンス・特許に関するガイドライン
- ・共通インターフェース仕様
- ・長期的な保守・運用体制

成果概要 1. 教示データ再利用ツール“teachingPlugin”

- ・ [I-(1)-①] (機能要件) マニピュレーション機能に関するコンポーネント・ツールの開発

① 日本電産ロボ (i611)対応作業

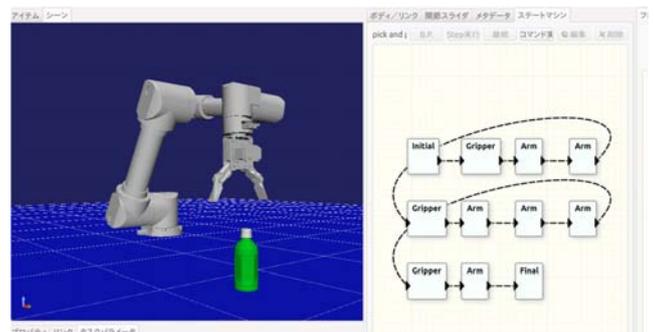
- ロボットとグリッパのモデルを読み込み
- シミュレータ環境で動作確認
 - ・ NextageOPENやUR3と同じpick&place動作パターンを実行可能であることを確認
- FollowJointTrajectoryAction / GripperAction 対応
 - ・ シミュレーション動作確認今後実機動作確認予定

② ドキュメント化と配布

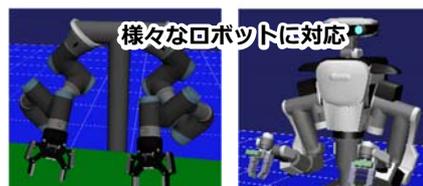
- web文書を作成
 - ・ <https://hanai.bitbucket.io/teachingplugin/>
- 開発状況に合わせて随時更新中

③ 多様なロボットへの対応

- Controllerの構造化
- タスク記述コマンド群と下位のコントローラ間のマッピングが容易に実現可能
- 「複数のJointPath + 二指並行グリッパ」のモデルまでを想定

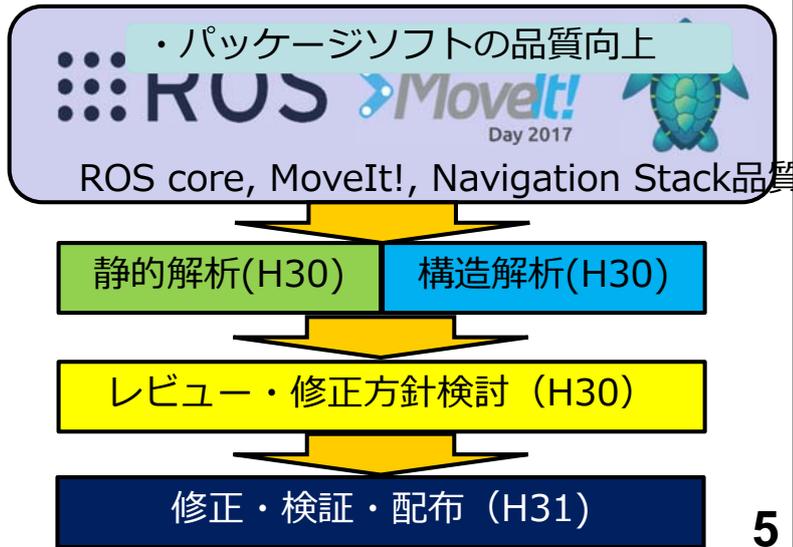
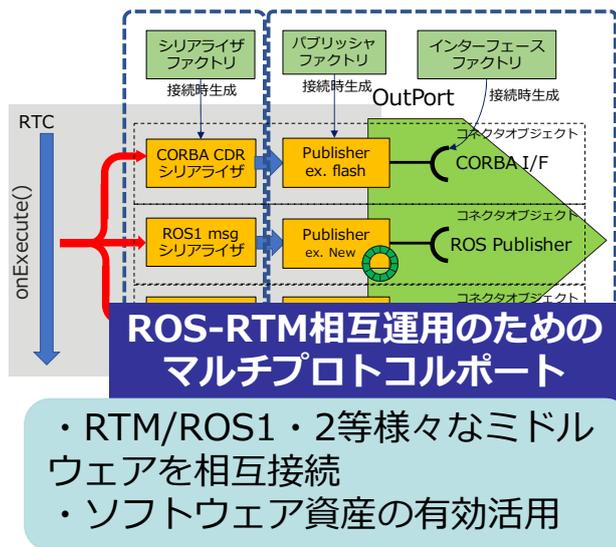


- ・ SI作業コストの多くを占める「教示」を簡略化
- ・ 過去の教示データを同様の作業へ再利用可能に



成果概要 2. ミドルウェア相互運用フレームワーク

- [I-(2)-②] (機能要件) 運用性向上のための技術開発 (デプロイメント)
 - ① OpenRTM-aist-1.2に実装、異言語間連携検証終了、リリース予定 (H30年度中)
- [I-(2)-③] (機能要件) 相互運用性向上のための技術開発 (ROS-RTM相互運用)
 - ① アーキ設計・プロトと実装中、OpenRTM-aist-2.0にてリリース予定 (H31年度予定)
- [I-(3)] (機能要件) SI効率化・安全性・運用性向上ツール開発
 - ① RTSystemEditor改修終了、FSM機能実装中。(H30年度中)
- [I I] (非機能要件) 安全・アーキテクチャ・ライセンス・品質管理
 - ① RRI WG3 研究会内に各種調査検討委員会の立ち上げ、H30→議論ガイドライン化
 - ② ソフトウェア品質管理、ROSコード静的解析・構造解析 (新規に実施中)



成果概要 3. 組み込み向けミドルウェア (イーソル)

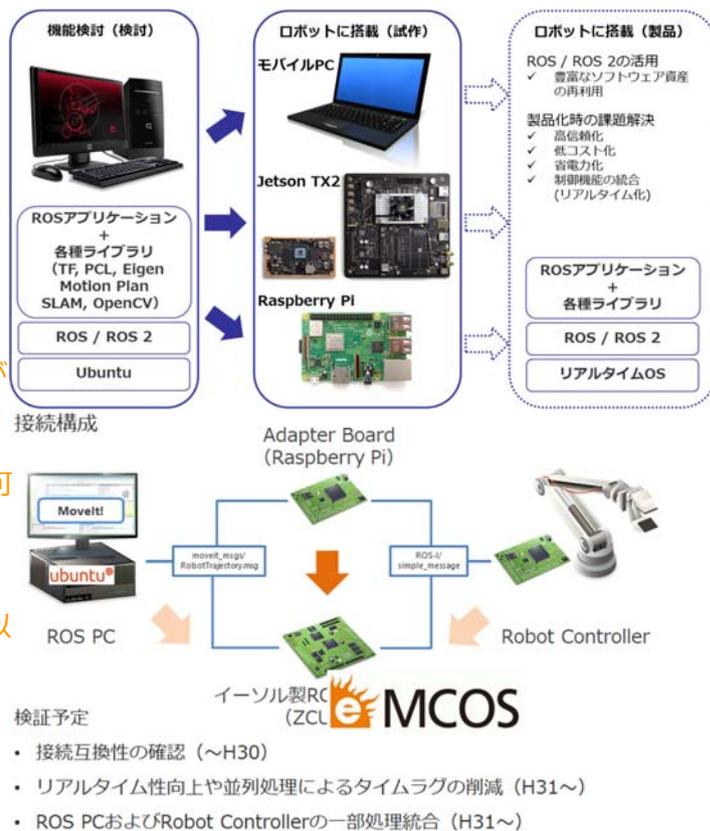
ロボットメーカーにとっての使いやすさ

ROS / ROS 2の活用

- ✓ 試作で開発したソフトウェアを組み込み製品に持ち込めること
 - C++のノードはパッチなしで動作することを目指して開発中
 - 主要なROS / ROS 2 関連の各種ライブラリも移植中

製品化時の課題解決

- ✓ 自社で検証が必要な範囲を削減できること
 - 商用リアルタイムOSの利用でOS部分の検証が削減可能
- ✓ 機能毎に異なる信頼性要求に対応できること
 - 動作コアを指定してメモリ分離・時間分離が可能 (予定)
- ✓ 必要最小限のハードウェアリソースに最適化できること
 - リアルタイムOSの利用でUbuntu対応ハード以外の選択が可能
- ✓ リアルタイム性を要する制御機能を統合できること
 - リアルタイムOSの利用でリアルタイムスレッドの統合が可能



成果概要 4. 安全開発ガイドの開発 (JQA)

サービスロボットの普及に関する課題 (RRI WG3 調査検討委員会での議論を通じて)

社会問題：労働力不足→ロボットの利活用

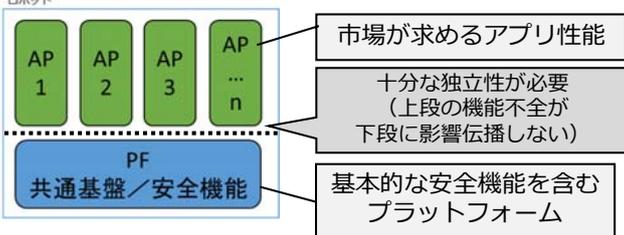


新しいサービス形態となるため、
使い手と作り手、双方の認識あわせが重要

使い手からのフィードバックを前提とした
イタレーションでの開発プロセスを提案

安全機能を中心とした開発プロセスの提案
(AIST、eSOL、JQA)

安全機能とアプリ機能の分離



7

安全開発ガイドのコンセプト

サービスロボットの課題に即した、安全機能を中心とした開発プロセスの提案とロボットの安全機能の開発に必要な周辺情報を織り込んだ解説

安全開発ガイドの目次案

- はじめに
 - 自律型生活支援ロボットについて
 - ロボットの開発の複雑さ
 - 安全開発の原則
 - 自律型生活支援ロボットの安全対策
 - ロボット制御ソフトウェアの特徴
 - ソフトウェアの安全対策
 - 安全対応技法
 - 安全対応プロセス
 - 安全対応観点
 - COTSの活用
 - ロボット開発チームのスキル
- 付録：参考情報

成果概要 5. コミュニティ形成・講習会等

イベント開催、講演

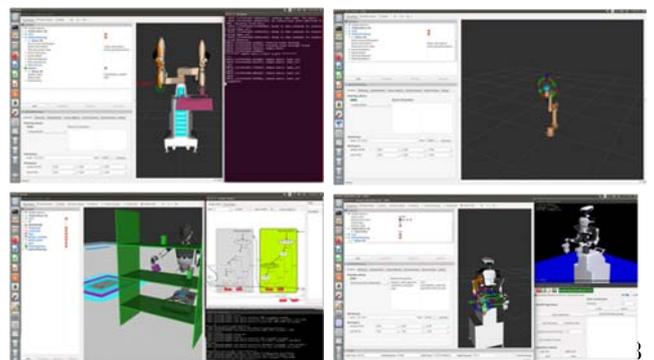
- World MoveIt! Day 2018 柏の葉
(<https://rosjp.connpass.com/event/101407/>)
 - MoveIt!のローカルイベントを主催
 - THK SeedNoid提供
- ROS-Industrial Workshop 2018 Singapore
(<https://rosindustrial.org/events/2018/6/27/ric-ap-workshop>)
 - ROSの産業応用を目指した国際コンソーシアムのAsia Pacific Workshopで講演
- ROSConJP 2018 秋葉原 (<http://roscon.jp/>)
 - 一般講演で活動と作成パッケージを紹介
 - プログラム委員
- B-Boost フランス、ボルドー (<https://b-boost.fr/>)
 - オープンソース・ソフトウェアの国際会議で招待講演



World MoveIt! Day 柏の葉 (約30人が参加)

ソフトPJ成果USB

- ROSをインストール済みのUbuntu 16.04 OSが軌道できるUSB
- 以下の各ロボットのROSプラットフォームソフトウェアのデモンストレーションが実行可能
 - 川崎重工業 Duaro
 - THK Seed-noid
 - 富士ソフト FSRobo
 - カワダロボティクス Nextage



集中研究拠点@柏の葉

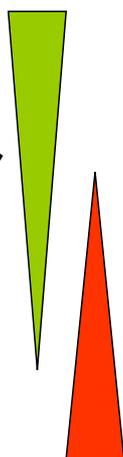
- 産総研特定集中研究専門員制度（特専）
- 研究場所は柏東大FC
 - 対象ロボットのROS化・RTM化を担当
- スキルレベルは求めないが、ソフトウェア担当者、あるいは今後ソフトウェアの面倒を見る人が原則。初心者でもOK。



つくばエクスプレス 秋葉原駅より約30分
 つくばエクスプレス つくば駅より約20分
 地下鉄千代田線 根津駅より 約40分
 （北千住乗り換え）

OSS活用の狙い

- アプリケーション
- 知能モジュール
- ライブラリ
- シミュレータ
- 通信ライブラリ
- デバイスドライバ
- 開発ツール



事業化
 未活用領域の
 ターゲット領域

ツール
 ROSの得意と
 する領域



WillowGarage社のスライドより。赤が研究に必要なツール作成等の雑作業。緑が研究そのもの。現状は上。多くの時間をツール作業に費やす。ROSは研究サポートを行うツール。（Steve Cousins speaking at Robo Development : <http://www.willowgarage.com/blog/2008/11/17/steve-cousins-speaking-robot-development-tuesday> より）

→ OSSをベースに各社のロボットシステムを構築

- ねらい1：世界中の研究成果を柏ロボットに取り込み統合できるように
- ねらい2：OSSの効率的な開発・保守環境により更なる発展を可能に

OSSの活用フレームワーク

- オープンソフトの活用部分
 - 技術的なコア*
(自律移動, 軌道計画)
 - センサ, アクチュエータドライバ
- 自社開発部分
 - サービス的なコア
(ユーザインターフェース, アプリケーション, サポート連携, 安全関連. . .)
 - 基幹システムとの連携

オープン

クローズ

ユーザ/市場に応じたUI/アプリ
事業コア

共通ソフトウェア基盤・ツール

ROS Visualization

- GUIツールキット
- Rviz,デバッグ, ブラウザ連携

MoveIt!

- 軌道計画
- 運動学・逆運動学
- 自己干渉検出

Navigation

- 自律移動
- 経路計画・軌道追従
- 自己位置同定
- SLAM地図生成

ロボットハードウェア

ros_control

- 軌道補間
- 実時間制御

robot_driver

- 台車制御
- オドメトリ計算

ロボットハードウェアデザイン

OSSの活用フレームワーク

- オープンソフトの活用部分
 - 技術的なコア*
(自律移動, 軌道計画)
 - センサ, アクチュエータドライバ
- 自社開発部分
 - サービス的なコア
(ユーザインターフェース, アプリケーション, サポート連携, 安全関連. . .)
 - 基幹システムとの連携

オープン

クローズ

ユーザ/市場に応じたUI/アプリ
事業コア

共通ソフトウェア基盤・ツール

ROS Visualization

- GUIツール
- Rviz,デバ

MoveIt!

- 軌道計画
- 運動学・逆運動学
- 自己干渉検出

Navigation

- 自律移動
- 経路計画・軌道追従
- 自己位置同定
- SLAM地図生成

ロボットハードウェア

ros_control

- 軌道補間
- 実時間制御

robot_driver

- 台車制御
- オドメトリ計算

ロボットハードウェアデザイン

Industrial Open source
Controller(IPC-8)

ROS Motion Controller(RMC)

DENSO b-CAP Comm.



RC-8 (Corei7)



Robot Pendant
Safety Line

Robot Controller (RC-8)

Robot Motion Server (RMS)
Software Safety Function :
- Joint velocity, Joint Position
- EEF Velocity, EEF Pose



RC-8 (Atom)

Ethernet Comm. (bCAP)

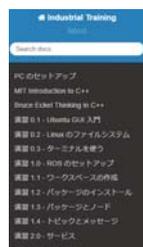
Robot Hardware

ハードコンソ要求に基づくROSパッケージの作成・改良・改善

要求元	目的/用途	問題点・課題・要求事項	修正内容(要求)	ステータス	変更対象モジュール名/変更内容概要	公開
自発	画像処理のテンプレートマッチングのサンプルの追加	OpenCVのテンプレートマッチングのサンプルに使えそうなものがない。	汎用性の高いテンプレートマッチングの機能を用いたノードを追加	完了	opencv_apps	公開済み
自発	地図生成ソフトウェアのリリース	Google製の地図生成ソフトウェアのリリースを待たず。	既存パッケージのKineticへのリリースを行った。	完了	cartographer_ros	公開済み
富士ソフト	ループバック動作のros_controllerノードを含むパッケージ。	jog_controlパッケージの動作テストなどに必要。	URDFのみでのロボット動作確認のためループバックコントローラパッケージを新規作成。	完了	fake_joint	公開済み
自発	MoveIt! チュートリアル	標準パッケージであるMoveIt!の使い方がイマイチよくわからない。	MoveIt!の日本語チュートリアルを作成し、公開した。	完了	tork_moveit_tutorial	公開済み
	ROS Wiki の日本語訳	ROSのパッケージを実際にリリースできる人が少ない。	ROSの継続的インテグレーションであるBloomについてドキュメントを日本語化。	完了	ROS Wiki / bloom	公開済み
自発	ROS-Industrial トレーニング教材の翻訳	ROS-Iの内容が知りたい、ROS-Iパッケージの使い方がよくわからない。	ROS-Industrialのトレーニングコースの内容を日本語化し公開。	完了	ROS Industrial トレーニング教材	公開済み

公開ドキュメント

- **ROS-Industrial トレーニング教材**
(<https://industrial-training-jp.readthedocs.io/ja/latest/>)
 - ROS-Industrialのトレーニングコースの内容を日本語化
- **ROS Wiki** (<http://wiki.ros.org/ja/bloom>)
 - ROSのCIシステムのドキュメントを日本語化



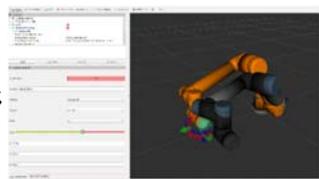
ROS-I トレーニング教材日本語訳

柏の葉集中研究拠点での連携による成果

ハードコンソ要求に基づくROSパッケージの作成・改良・改善

要求元	目的/用途	問題点・課題・要求事項	修正内容(要求)	ステータス	変更対象モジュール名/変更内容概要	公開
富士ソフト	アームのティーチング操作の改良 (Sier)	産ロボのペンダント機能に相当するジョグ動作機能が必要。(MoveIt!には存在せず)	ロボットアームのジョグコントロールUIと制御を新たに作成。	対応中	jog_control	公開済み
産総研、YOODS、他	産業用ロボットアームのROS化	三菱電機製産ロボに、ROSから動作させるドライバパッケージが存在しなかった。	MELFAシリーズコントローラのドライバパッケージを公開する。	完了	melfa_robot	公開済み
KHI	ロボット関節角軌道のグラフ化	MoveIt!の生成軌道はrviz上でアニメーションとして表示されるが、各関節の速度・加速度のグラフ表示がない。	ロボットの関節角軌道の時間プロファイル(グラフ)を表示するrqtプラグインを新たに作成。	完了	rqt_joint_trajectory_plot	公開済み
自発	MotionJPEGフォーマットに関する機能追加とメンテナンス	libuvc_cameraがKinetic未対応、かつMotionJPEG未対応。	MotionJPEGフォーマットへの対応とKineticへのリリース。	完了	libuvc_camera	公開済み

jog_controlパッケージ



- **jog_control** (https://github.com/tork-a/jog_control)
 - ロボットアームのジョグ動作のパッケージ
- **fake_joint** (https://github.com/tork-a/fake_joint)
 - ループバック動作のros_controllerノード
- **rqt_joint_trajectory_plot**
(https://github.com/tork-a/rqt_joint_trajectory_plot)
 - 時系列軌道のグラフによるプロット



melfa_robotパッケージ

- **melfa_robot** (https://github.com/tork-a/melfa_robot)
 - 三菱電機製の産業用ロボットアーム(MELFAシリーズ)をROSから制御するためのパッケージ

ROSのコード品質の見える化

ROS解析結果

指摘重要度ラベル	指摘件数 [件]	指摘密度 [件 /KLOC]
重大	2,943	2.2
エラー	1,767	1.3
警告	2,845	2.1
レビュー	82,420	60.2
総指摘件数	89,975	65.8

全体

指摘重要度ラベル	指摘件数 [件]	指摘密度 [件 /KLOC]
重大	1,312	5.1
エラー	384	1.5
警告	394	1.5
レビュー	14,698	57.4
総指摘件数	16,788	65.6

本質機能のみ

Navigation Stack解析結果

指摘重要度ラベル	指摘件数 [件]	指摘密度 [件 /KLOC]
重大	67	2.0
エラー	231	6.9
警告	106	3.2
レビュー	1,979	59.5
総指摘数	2,383	71.7

MoveIt!解析結果

指摘重要度ラベル	指摘件数 [件]	指摘密度 [件 /KLOC]
重大	115	0.8
エラー	606	4.4
警告	173	1.2
レビュー	11,284	81.1
総指摘数	12,178	87.5

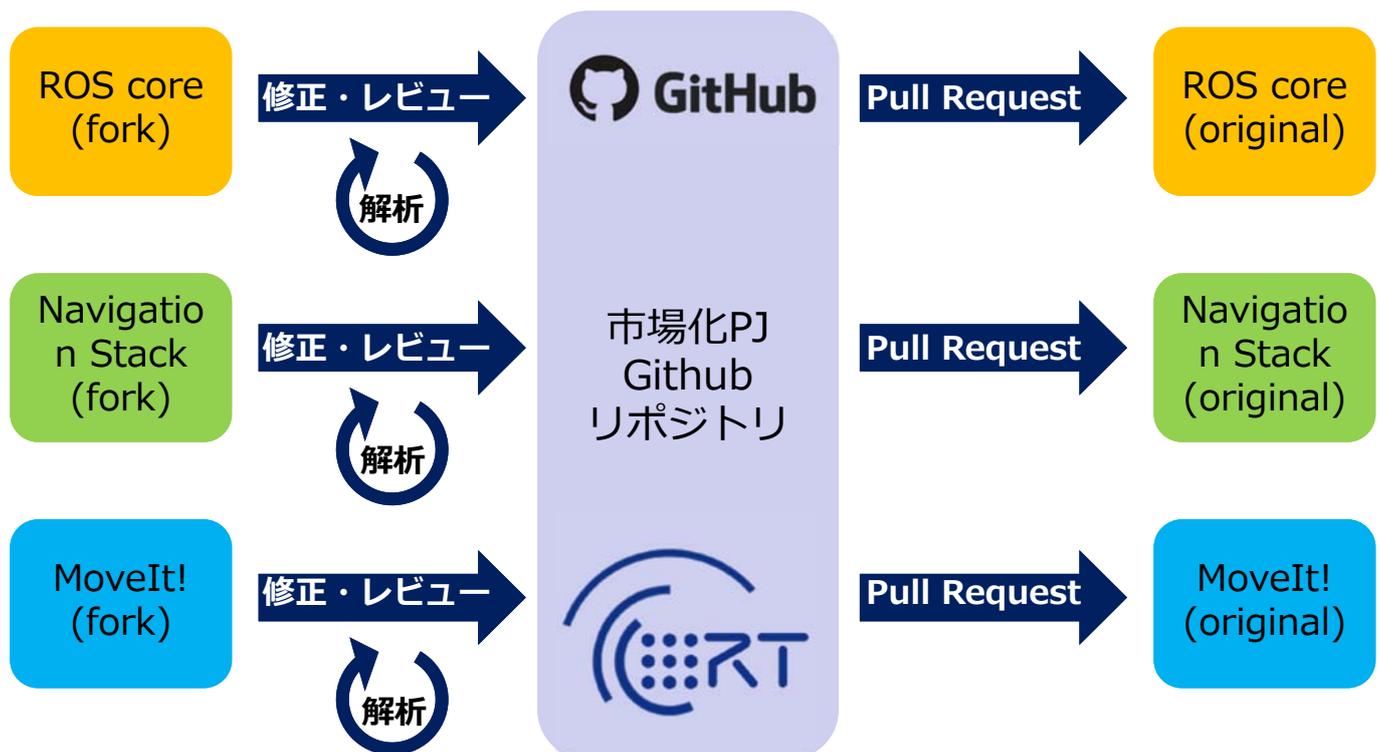
一般的な製品における指摘密度

指摘密度[件/KLOC]			
重大	エラー	計	
15	0.5	0.4	0.9

一定レベル以上の品質を保っているものの、製品化レベルには、さらなる品質向上が求められる。

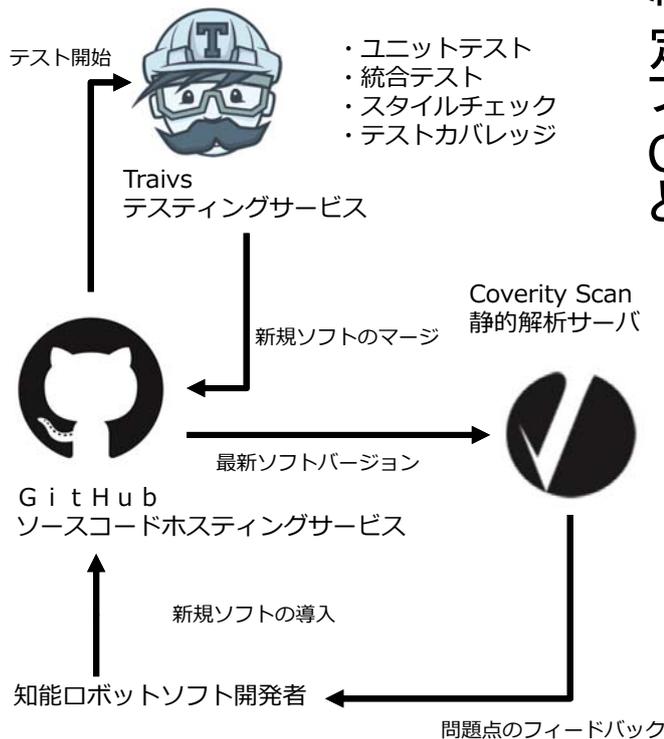
↓
静的解析結果を反映・修正

ROSコードの高品質化の取り組み



静的解析結果に基づき、コードレビュー、修正案検討、修正を実施中
最終的には、オリジナルリポジトリに修正を提案 (PullRequest)

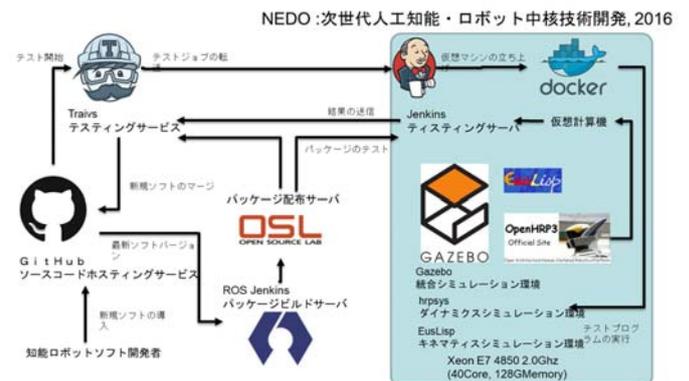
継続的検証技術によるソフトウェア品質の常時検証



継続的検証システム

定常的に変化, 更新しているオープンソースソフトウェアへの対応

Continuous Integration システムとロボットシステムの連携



NEDO 2015 超広域認識行動計画学習ロボット知能ソフトウェア要素群の透過的継続的システムインテグレーション管理機構技術の研究開発

NEDO 2011 次世代ロボット知能化技術開発プロジェクト・RTM-ROS 相互運用開発環境

まとめ

1. ハードウェアが購入できる
2. ソフトウェアが簡単に導入できる
3. 他の人も使っていること.

多くの人に使ってもらうことで, ロボット未活用領域を開拓する基盤プラットフォームを確立

プロジェクトが終わっても使い続けられるような仕掛けづくり