

ロボット活用型市場化適用技術開発プロジェクト —ロボットのプラットフォーム化を目指す研究開発（第2報）—

Technology Development Project for Robot Commercialization Applications
- Research and development aiming at platformization of robots (2nd report) -

○正 安藤 慶昭（産総研） 正 琴坂 信哉（埼玉大学）
正 岡田 慧（東大） 増田 昌庸（NEDO）

Noriaki ANDO, AIST, n-ando@aist.go.jp
Shinya KOTOSAKA, Saitama University
Kei OKADA, University of Tokyo
Masanobu MASUDA, NEDO

The R&D project “Technology Development Project for Robot Commercialization Applications” is conducting R & D to introduce robots into unused areas. Although robot is expected to be utilized in the such as manufacturing field including “Sanpin-Sangyou (food, medicine, cosmetics)” due to issues such as technology and price etc., it is difficult to introduce there. The project is aiming to promote the introduction by introducing robot hardware and software platform into tasks and fields that have not yet been introduced. In this paper, we will introduce current activities and results of this R&D project so far.

Key Words: Robot software platform, software quality, OSS, system integration, commercialization, application

1. はじめに

NEDO による委託事業「ロボット活用型市場化適用技術開発プロジェクト」（以下市場化プロジェクト）において、ロボットを未活用領域に導入するための研究開発が、2017 年度から 9 グループ・17 組織（ソフト開発事業者 1 グループ、ハード開発事業者 9 グループ）により行われている[1].

本研究開発事業は、三品（食品、医薬品、化粧品）産業を含むモノづくり分野、サービス分野・生活支援分野等「ロボット未活用領域」、すなわちロボットの活用が期待されつつも、技術・価格等の課題により導入が進んでいない作業・分野に対し、ロボットのハードウェア・ソフトウェアをプラットフォーム化することによる導入促進を目指している。特に、ロボットハードウェアおよびソフトウェアの共通化・プラットフォーム化を促進することで、ロボット導入コストの多くを占める SI（システムインテグレーション）費用の削減と効率化を促し、ロボット導入の障壁を下げることを目標にしている。

本稿では、[2]に引き続き本研究開発事業の背景、未活用領域へのロボット導入戦略とこれまでの取り組みと課題について議論する。

2. 未活用領域へのロボット適用

図 1 に示すように、従来から産業用ロボットは、自動車・電器等製造業に代表される大量生産ラインにおいて、大規模に導入されることで効果を発揮してきた。その際のロボットのシステムインテグレーションの主体は社内の生産技術部門（社内 SIer）であり、生産ラインの設計、開発、導入および保守に至るまで自前で実施することによりきめ細やかな対応が可能であった。

一方で、自動化が求められつつもまだ導入が本格化していない分野においては、その対象は自社 SIer を持たない中小・中堅企業や、大企業における小規模な導入にとどまる場合が多い。システムインテグレーション作業についても、個別に異なる作業対象のような多様で小規模な需要が多く、SIer にとっては利益が上がりにくく、ユーザにとっては費用対効果が低く見られがちであるといった問題がある。

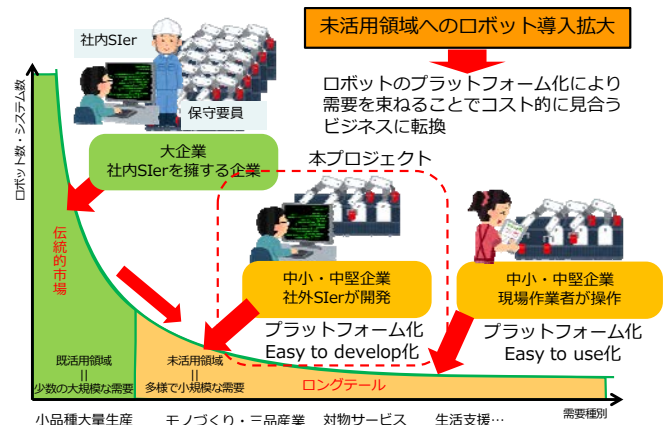


Fig. 1 Introducing robots from conventional areas to unused areas.

「ロボット新戦略」[3]においては、未活用領域へのロボット導入には、種々の技術開発とともにシステムインテグレーションを効率化するためのロボットのプラットフォーム化（easy to develop 化）を促進することで SI 費用の削減と効率化が必要であり、最終的には、現場の作業員など誰もが道具のように使いこなせる「Easy to use」なロボットの確立が必要であることが指摘されている。

3. ソフトウェアプラットフォーム

ロボットのソフトウェアは基本的に、図 2 に示すように、OS・ミドルウェアなどの基盤領域と、ロボット機能を実現する分野・機能別ソフトウェア、システムを統合するための SI 領域に分けられる。このうち、未活用領域が対象にするロボットの場合、個別の要件に都度対応するため SI 費用が大きな比重を占めていることがコスト増加の原因と指摘されている。

そこで、本事業では、ソフトウェア開発コスト削減のため、オープンソースソフトウェア（OSS）を積極的に活用することを目指している。OSS は無償で優れたソフトウェアを利用できる利点がある一方で、ドキュメントの不足や品質、ライセンス、特許など商用利用する上での課題も多い。本事業では、OSS を実用レベルで活用するためのこうした課題解決にも取

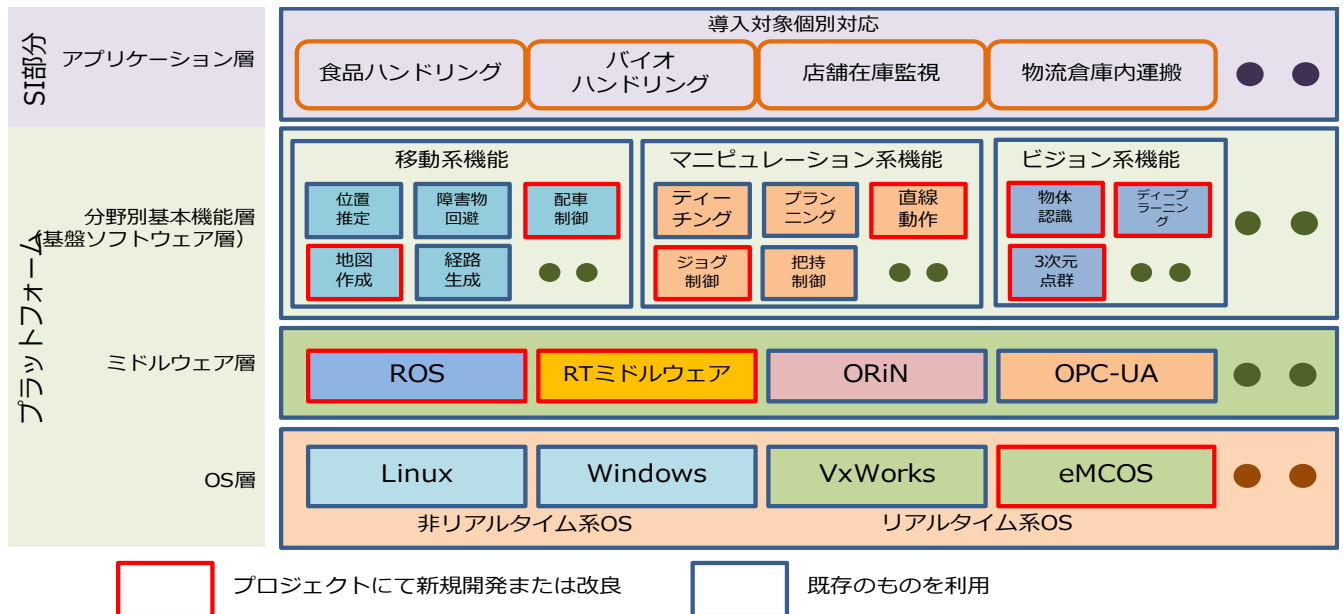


Fig. 2 The layer structure of the robot software platform.

り組んでいる。

また、ソフトウェアのプラットフォーム化を行う上で、ソフトウェアの機能にかかわる部分以外、すなわち非機能要件に関しても、諸問題の解決が必要である。これに関しては、プロジェクト内での取り組み以外に、後述する RRI（ロボット革命イニシアティブ協議会）内に設置した調査検討委員会において課題解決に取り組んでいる。

する過程で、実用上当然あるべき機能が実装されていないケースや、改善すべき機能があることが明らかとなった。例えば、アームのティーチング操作に必要なジョグコントロール機能がない、アームの関節角軌道をグラフ表示する機能がない、といった機能の不足に関しては新たにプラグインなどを実装[5]して対処した。このほか、利用上の情報不足、商用利用上問題となるライセンスが含まれるパッケージの存在等についても、ドキュメント作成、ライセンス調査等を行い、課題を解決した。

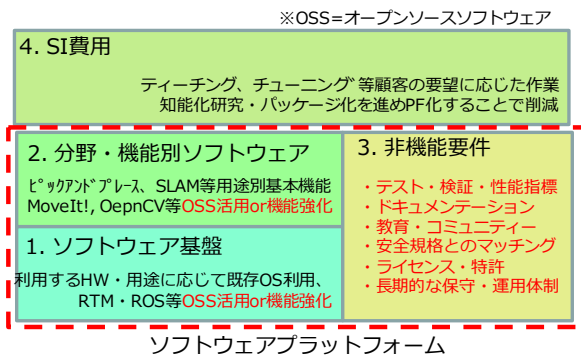


Fig. 3 The robot software platform and system integration cost.

3.1 OSS(オープンソースソフトウェアの活用)

本事業では、ロボットミドルウェアとしては ROS (Robot Operating System) や RT ミドルウェア (OpenRTM-aist) を利用し、その上で動作するナビゲーション、マニピュレーションやビジョン等の基本パッケージを、多くのプロジェクト事業者が利用している。既存のモジュールでは対応できない機能については拡張や新規に開発するなどして、多くのシステムにおいて、共通的に利用されるモジュール群を抽出、整備することを目指す。このようにして整備された共通部分（ソフトウェアプラットフォーム）上に、未活用領域の個別アプリケーション部分を構築することで、コスト削減を実現する。

3.2 ハード開発事業者に基づく ROS パッケージの改善

ハード開発事業者が実際のアプリケーションに ROS を適用

3.3 ROS (Robot Operating System) 等の品質管理

OSS を活用する上での課題の一つとしては、品質管理が難しいことが挙げられる。こうした課題を解決する方法として、ROS 関連ソフトウェアの品質解析と改善に取り組んでいる。

本事業で利用されている ROS 関連ソフトウェアについて調査を行い、多くのシステムで利用されているソフトウェアを抽出し、それらを品質の解析を実施している(表 1)。

ソースコードの記述を解析し、記述上明らかな不具合、潜在的な不具合と考えられる記述等を検出する静的解析、ソースコードにおける関数・クラスの数や行数、関数間・クラス間の結合度合いなど、構造に関する指標により評価する構造解析等を実施した。詳細はここでは示さないが、商用ソフトウェアの数倍～倍程度の不具合出現率を示すことが明らかになり、何らかの対策が必要であることが示された。今後、修正を含め対策について検討を行う予定である。

Tab. 1 Widely used ROS related packages in the project.

パッケージ名	利用者数
ROS core	10
Rviz	5
MoveIt!	5
Navigation Stack関係 gmapping, navigation, move_base, amcl, map_server 等	5
GAZEBO	3
ros_control	3
opencv	3
Rqt	3

3.4 継続的インテグレーション手法

日々コミュニティで開発される OSS は、商用製品のような一定の品質を保証することは難しい。ソースコードの変更毎にテストを実施しデグレートの発生を抑制する手法として継続的インテグレーション (CI: Continuous Integration) がある。ロボットシステムにおいては物理的に動作するシステムを対象とするため完全な CI の実施が困難であることが課題であったが、CI に実際のロボットを含めることでテストをより確実なものとする手法：物理統合テストフレームワーク (PIT-FW: Physical Integration Testing Framework) [4] の研究開発を行っている。これにより、ソフトウェアのみのテストでは発覚しない不具合を事前に発見することが可能となる。

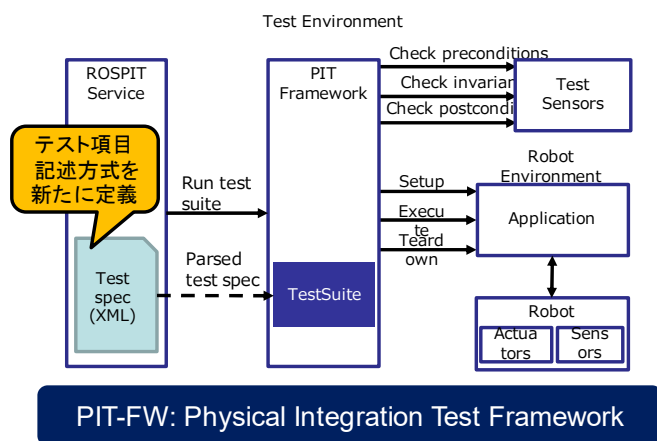


Fig. 4 Proposed physical integration test (PIT) framework.

3.5 その他非機能要件に関する取り組み

非機能要件に該当する安全設計、特許・ライセンス、評価指標、アーキテクチャやインターフェースの共通化、といった諸問題については、業界全体でのコンセンサスが重要となる。これらの課題を議論する場として、RRI 内に調査検討委員会を設置し、継続的に議論を行っている。以下に示す5つの委員会を設置し、これらの課題をプロジェクト参加事業者のみならず関心のある参加企業も含めて広く議論している (図 5)。

- ロボット安全設計開発調査検討委員会
- ソフトウェア特許・ライセンス調査検討委員会
- ロボットシステム開発プロセス・品質管理調査検討委員会
- ソフトウェアアーキテクチャ調査検討委員会
- 次世代ソフトウェア実装手法調査検討委員会

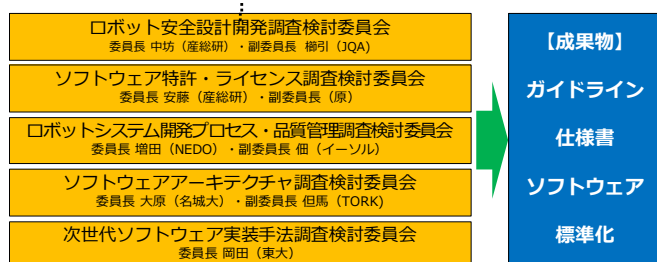


Fig. 5 Five Research Investigation Committee in RRI WG3.

それぞれの委員会のアウトプットとしては、ガイドライン (例：図 6) や仕様書、実装されたソフトウェアや標準化も視野に入れている。これらの活動により、技術開発のみでは解決が難しい OSS を用いたロボットシステム開発にまつわる諸問題に対して、一定の解決策を提示することを目指している。

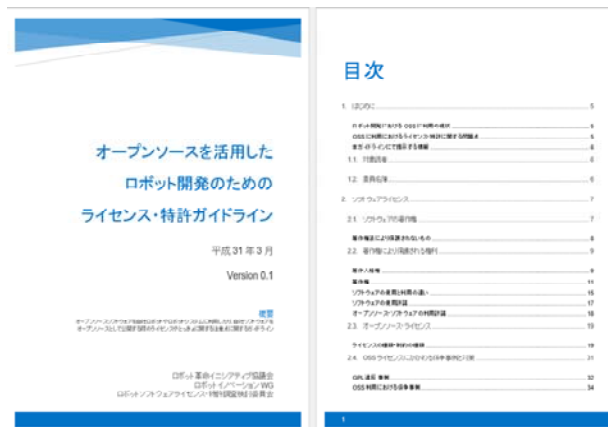


Fig. 6 Draft of the license and patent guideline for OSS based robot development.

4 おわりに

本稿では、2017 年から開始された NEDO による委託事業「ロボット活用型市場化適用技術開発プロジェクト」について、[1]に引き続き、ソフトウェアプラットフォームの開発と非機能要件の諸問題の解決について、これまでの活動及び成果について紹介した。重要なポイントとしては、機能的な開発のみならず、ソフトウェアプラットフォームが有すべき共通機能にかかわる機能・非機能要件について解決策を明らかにすることであり、これにより、未活用領域におけるロボットのシステムインテグレーションを容易にすることであり。最終的には、本プロジェクトで開発された各種ソフトウェアと、それを搭載したロボットハードウェアが、未活用領域において実用化され、同様の分野へと横展開されることを目指す。

参考文献

- [1] 「ロボット活用型市場化適用技術開発プロジェクト」紹介ハンドブック (2017 年度版), 国立研究開発法人新エネルギー・産業技術総合開発機構, http://www.nedo.go.jp/library/pamphlets/RB_hbook201709.html
- [2] 安藤, 琴坂, 岡田, “ロボット活用型市場化適用技術開発プロジェクト—ロボットのプラットフォームを目指す研究開発—”, ROBOMECH2018, pp.2A1-F07, 2018
- [3] 「ロボット新戦略」Japan's Robot Strategy —ビジョン・戦略・アクションプラン—, 日本経済再生本部, 2015.2
- [4] Floris Erich et al. “Design and Development of a Physical Integration Testing Framework for Robotic Manipulators”. In: 2019 IEEE/SICE International Symposium on System Integration. 2019, pp. 602–607.
- [5] Tokyo Opensource Robotics Kyokai github page, “jog_control” package, https://github.com/tork-a/jog_control