

第31回 日本ロボット学会 学術講演会

2013年9月4日(水)～6日(金)

THE 31st
ANNUAL CONFERENCE
OF THE ROBOTICS SOCIETY OF JAPAN

講演概要集



主催：一般社団法人 日本ロボット学会

ロボットシステム開発における SysML の活用

○岩永 寿来((株)チェンジビジョン) 安藤 慶昭(産総研) ビグズ ジェフ(産総研)

1. はじめに

現在、高度なサービスロボットの開発をより効率的に行うため、産業技術総合研究所が開発・配布する OpenRTM-aist など、国際標準化団体 OMG において標準化された、Robotic Technology Component (RTC)標準[1]に基づく実装である、ロボットテクノロジーミドルウェア(RTM)の研究開発と提供が盛んに行われている。

ロボットテクノロジーシステム(RTシステム)は、複数のシステムが組み合わさり、さらに高度なサービスを提供する、複合システム(System-of-Systems)で構成されることも多く、システムの大規模複雑化が進んでいる。

特にこのようなシステム開発では、システムの理解や開発者間での円滑なコミュニケーションのため、モデル駆動開発の重要性が高まっている。それに伴い、システムの様々な側面から記述が行える、システムエンジニアリングのための標準モデリング言語である、Systems Modeling Language (SysML) [2] が RT システム開発でも用いられるようになってきた。

また、よりよい RT システム開発を効率的に行うには、開発者間のコミュニケーションのためだけでなく、SysML モデルで設計したシステム構成や RT コンポーネントの実装に連携可能な枠組みが求められる。

本稿では、SysML を用いて RT システムを設計し、同一の SysML モデルから、異なる OMG RTC に準拠した RTM 実装上のコンポーネントが相互運用して動作するロボットシステムを開発を通して、SysML を用いた RT システム開発を効率的に行うために必要なプロセスや求められるプラットフォームの実証と考察について述べる。

2. 実証システムの概要とツール連携の全体像

実証システムとして、Kinect を通したオペレータのジェスチャに割り振った 8 の字や前後走行など決まった動きを、OMG RTC 標準に準拠した異なる実装の 2 つのミドルウェア、OpenRTM-aist とプライベートな RTM 実装上で、それぞれ相互運用して動作する RT システムを取り上げた。

実証システムは、主に次の 3 つのコンポーネントを含む、図 1 のシステム構成で作成した。

- Kinect Input コンポーネント：Kinect を利用し骨格の位置情報を取得、送信するコンポーネント
- Controller コンポーネント：Kinect コンポーネントから受け取った骨格の位置情報を組み合わせ、

規定のジェスチャを行った時に、状態遷移を行い、ロボットに対して 8 の字や前後運動を行うことを指示するコンポーネント。

- Robot コンポーネント：Controller から指示された動きを行うロボットコンポーネント。

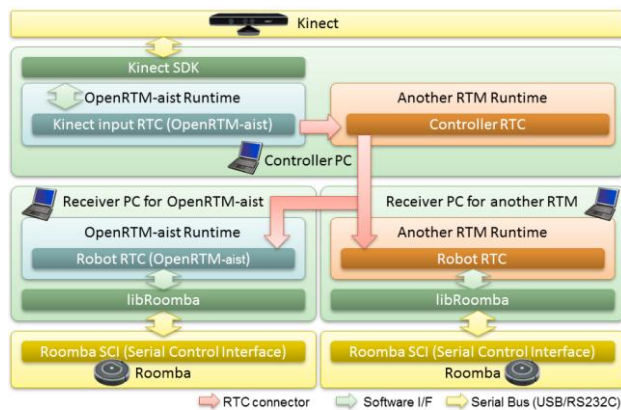


図 1 実証システムの構成

図 2 は実際に、RT システムのデモンストレーションを行なっている図である。Controller PC に配置された Kinect Input コンポーネントがオペレータの骨格を把握し、Controller コンポーネントは決められたジェスチャに割り振られた 8 の字などの走行を、WIFI を経由して、Receiver PC に配置された Robot コンポーネントに指示している。

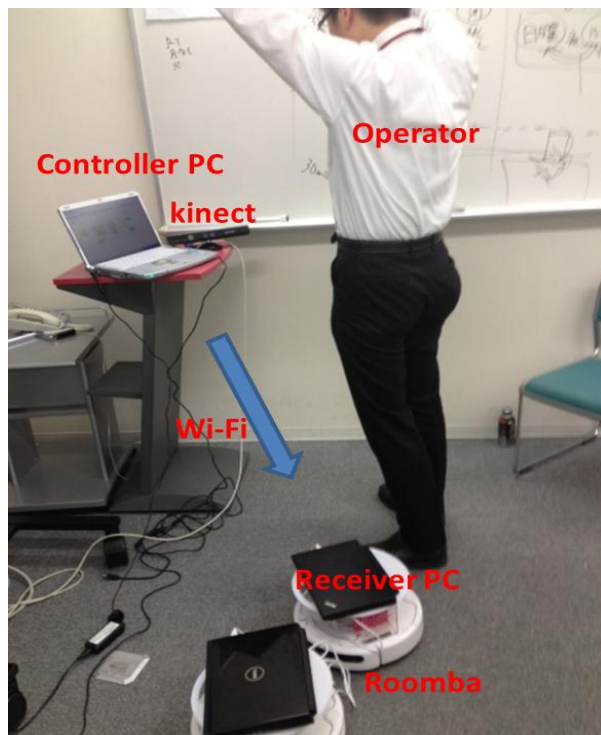


図 2 RT システムのデモンストレーション

3. SysML を用いた RT システム開発プロセス

RT システムの開発は、図 3 に示すように、求められる要求と分析、設計を SysML で行い、得られたモデルから RTM が提供する IDE に連携し、RT コンポーネントの実装へ連携するプロセスで実施した。

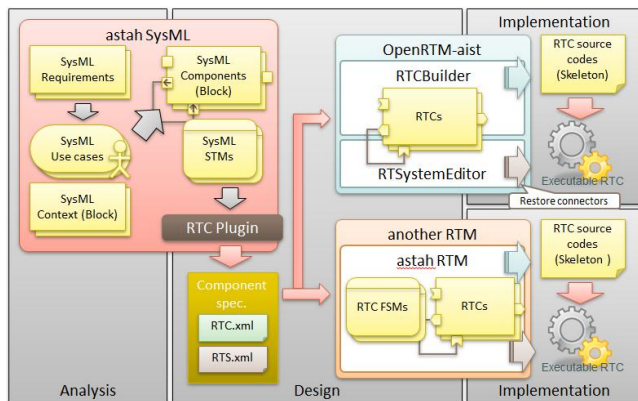


図 3 開発プロセス

SysML は、システム全体を捉えるため、システム開発初期の、ハードウェア/ソフトウェアに分割される前の段階での使用が効果的であり、メカ、エレキ、ソフトの間のコミュニケーションの断絶を緩和できると考えられている。また図 4 に示す主要なダイアグラムの 4 本柱で構成され、できるだけ UML を流用するコンセプトで、足りないセマンティクスと表記法を新たに定義、拡張している。

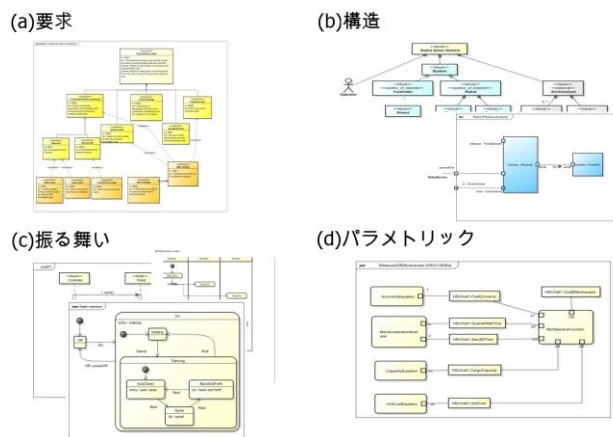


図 4 SysML の主要な図

今回の実証では、要求図やユースケース図を利用し RT システムに求められる要求を分析し、必要なシステム構成や振る舞いを、ブロック定義図やステートマシン (状態遷移) 図を利用して分析、設計した。

4. 要求の分析

要求を示す図 (図 4-a) では、図 5 に示すように、システムに求められる要求の構成、要求間の関係を

記述する要求図 (req: Requirement Diagram) を利用した。<<satisfy>> 関係を「要求」と「コンポーネント」の間に引くことで、コンポーネントの実装意図を論理的に表現できる。

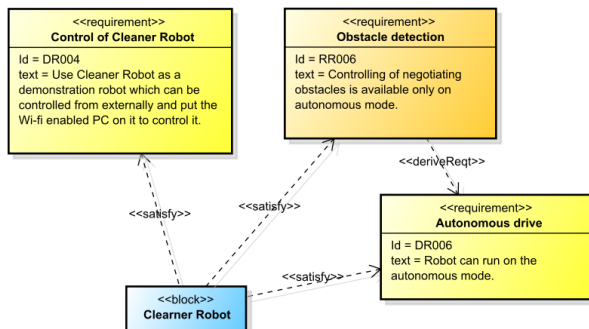


図 5 要求図による要求とコンポーネントの関連

この設計要素が導出された要求は何であるか。また設計要素を変更した際に、どのような影響があるかといった、要求とコンポーネントの間などでのトレーサビリティ、インパクト分析の実現も必要である。このように要求と設計要素とを紐付けることで、そのような分析も可能となる。

5. 構造の設計

構造を示す図 (図 4-b) では、システムを構成する要素の役割と仕様を示すブロック定義図 (bdd: Block Definition Diagram) と、その要素間の接続関係を表し、要素の役割や利用方法を示す内部ブロック図 (ibd: Internal Block Diagram) を利用した。

内部ブロック図にはポートと呼ばれる外部要素との接続点が定義されており (図 6)、OpenRTM-aist が提供するデータポートやサービスポートにマッピングすることで、システム構成の記述に利用することができる。

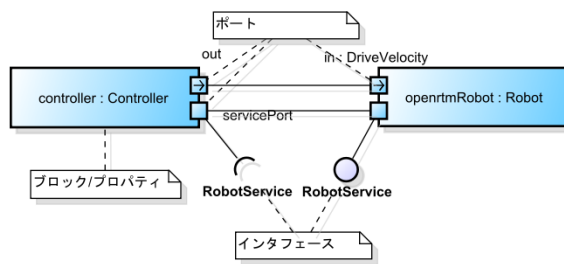


図 6 内部ブロック図での RTC 要素の表現

また、ポート間の接続はコネクタやアイテムフローで表現できるため、OMG RTC 標準で定義されているコネクタにマッピング可能である。ただし、データをどのように送信するか、受信するかという設定であるデータフロータイプや、データを送信するタイミングの設定であるサブスクリプションタイプなど、OpenRTM-aist で定義されるコネクタのプロパティは、SysML モデル上では単純には表現できない。この点は後述するように、RT システム向けに

SysML を拡張することで実現可能と考える。

SysML にはブロック間のサービス呼び出しを可能とするインタフェースである、要求・提供インタフェースも定義できるため、OpenRTM-aist のサービスサポート上で提供・要求されるインタフェースにマッピング可能である。

このように、SysML によって、システムを構成するハードやソフトウェア、それらの関係が明確にでき、またソフトウェアコンポーネントの一部については、OMG RTC 標準や各種 RTM に適合した、RT システムの設計が行えるため、関係者間の円滑なコミュニケーションの実現に効果的であると考えられる。

ただし、RT システムでは、リアルタイム性(時間概念)をモデルに表現する必要もあるため、SysML のシーケンス図や、SysML のみならず、組み込みシステム向けのドメイン特化モデリング言語 MARTE (Modeling and Analysis of Real-Time and Embedded Systems) [3]などの利用も検討が今後必要だろう。

6. 振る舞いの設計

振る舞いを表す図(図 4-b)のうち、ステートマシン図をロボットの振る舞いの設計に利用した。具体的には、Kinect で検出したジェスチャに応じて、対応するロボットの動作モードをステートマシン図により表現した。

OMG RTC 標準では、コンポーネントのタイプとしてイベント駆動型の FSM(Finite State Machine)コンポーネントを定義している。今回 OpenRTM-aist と相互運用を行う、プライベートな RTM 実装にて、ステートマシンを実現する FSM コンポーネントの実装が用意されており、その機能を利用した。

またステートマシン図から RT コンポーネントの実装を生成し、加えて実行時にその状態遷移の状況をモニタリングするツールを作成した。ステートマシンは実装が複雑でソースからはステートマシンの構造がつかみにくい、モデルとして設計されたステートマシンから直接、実装を生成することで、確実に設計に沿った形で実現することができた。

このように、FSM コンポーネントの実装は、プライベートな RTM 実装にて、OMG RTC 標準をベースに独自の仕組みで実現されているが、OMG RTC 標準では同等の FSM コンポーネントが定義されている。さらに、現在これを拡張する Finite State Machine for RTC 標準(FSM4RTC) [4]の策定が進んでおり、それらの標準を組み合わせると、ツールによる FSM コンポーネントの実装が非常に容易になる見込みである。

7. SysML モデルから実装への連携

SysML は、システムの仕様、分析、設計、Verification、Validation の記述を支援し、システムの様々な側面から記述が行える、システムエンジニアリングするための表記法である。このうち構造の分析、設計が行

える、ブロック定義図はコンポーネントを用いて構成されるシステム全体の記述へ、内部ブロック図は RT コンポーネントの定義へ容易にマッピングできることは上述した。

そこで、図 7 に示すようにブロック定義図から RT システム構成を示す、RT システム仕様記述方式 RT-System (RTS) Profile[5]を生成し、内部ブロック図からは RT コンポーネントの構造を示す、RT コンポーネント仕様記述方式 RTComponent Profile[6]を生成するツールを作成した。

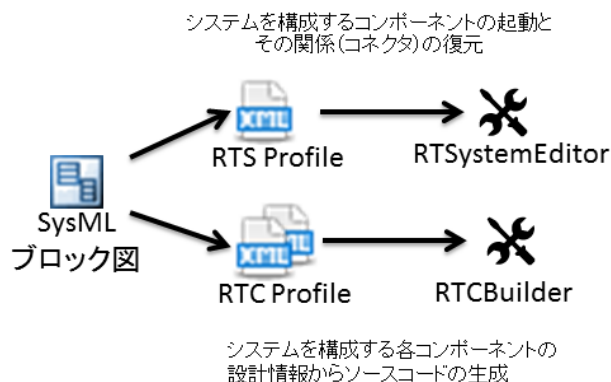


図 7 SysML から RTS、RTC Profile への連携

この 2 つの仕様記述方式を用い、各 RTM が提供する IDE、例えば RTSPProfile を用いて RTC をリアルタイムにグラフィカル操作する機能を持つ RT SystemEditor 上でシステムを構成するコンポーネントの起動やコネクタの復元を行い、RTC Profile から、RTC のソースコードのひな型を生成できる RTCBuilder に連携し、OpenRTM-aist に適した細かい設定を行い、実際の RT コンポーネントのソースコードを生成する方法を検証した。

検証の結果、モデルの情報から、RTS Profile や RTC Profile によって、各 RTM が提供するツール間で共通の形式でデータ交換が行えるが、SysML から例えば、RT コンポーネントの種別や、ポートやコネクタのプロパティなど、これら RT システム向けの Profile を効率的に設計、生成できるよう、今後 SysML を RTC 向けに拡張し、標準化する必要があると考える。

ただし、共通の SysML モデルから、複数の実装が異なる RTM 上で利用可能とするために、SysML では、RT システムとして OMG RTC など標準仕様に沿った共通的なモデルを定義し、各種 RTM 向けの細かい設定は、そのモデルのデータを連携した各種 RTM 側のツールにて行うことが望ましい。

現在、OpenRTM-aist が提供するポートやコネクタの一部プロパティは、OMG RTC 標準として策定されていないが、現在 OMG では、それらの仕様に関わる、FSM4RTC 標準や Unified Component Model for DRTE Systems 標準(UCM) [7]が検討されている。SysML の RTC 向け拡張を検討する際は、これらの動向を反映する必要があるだろう。

8. おわりに

本稿では、RT システムにおける SysML を用いたモデル駆動開発について述べた。SysML によって、システムへの要求、構造、機能、振る舞い、パフォーマンスなど、システムの様々な視点からの記述を文章中心ではなくモデルとして記述でき、開発者間の円滑なコミュニケーションが行えた。

また SysML のブロック定義図や内部ブロック図によるコンポーネントのモデルと、RT コンポーネントとの類似性から、モデルによる設計から実装への連携、モデル駆動開発が実現できた。

ただし、実証では SysML での RT システム、RT コンポーネントの表現力のミスマッチといった問題点が確認された。この問題を解決するためには、SysML の RTC 向け拡張の標準化が必要であることが分かった。

また RT システム開発における、モデル駆動開発を促進するため、今後 SysML から RTS Profile や RTC Profile を生成するツールや、独自の RTM 実装で実現した FSM コンポーネントの標準化と設計ツールの提供も必要と考える。

参 考 文 献

- [1] Object Management Group, “Robotic Technology Component Specification Version 1.1”, formal/12-09-1
- [2] Object Management Group, “OMG Systems Modeling Language Version 1.3”, formal/12-06-01
- [3] Object Management Group, “UML Profile for MARTE: Modeling and Analysis of Real-Time Embedded Systems Version 1.1”, formal/11-06-02
- [4] Object Management Group, “Finite State Machine Component for RTC RFP”, robotics/13-06-11
- [5] OpenRTM-aist, “RT システム仕様記述方式 (RTSProfile) Version 0.2”,
<http://www.openrtm.org/OpenRTM-aist/documents/RTSProfile.pdf>
- [6] OpenRTM-aist, “RT コンポーネント仕様記述方式 (RTCProfile) Version 0.2”,
<http://www.openrtm.org/OpenRTM-aist/documents/RTCProfile.pdf>
- [7] Object Management Group, “Unified Component Model for DRTE Systems DRAFT RFP”, mars/13-05-03