

# SDO (Super Distributed Object) に基づいた RT-Component オブジェクトモデル

安藤慶昭 (産総研), 神徳徹雄 (産総研), ルメア オリビエ (産総研),  
北垣高成 (産総研), 末廣尚士 (産総研)

## RT-Component Object Model based on SDO (Super Distributed Object)

\*Noriaki ANDO (AIST), Tetsuo KOTOKU (AIST), Olivier LEMAIRE (AIST),  
Kosei KITAGAKI (AIST) and Takashi SUEHIRO (AIST)

**Abstract**— We have developed a framework of RT-Component which promotes application of Robot Technology (RT) in various field. In this paper, standardization of RT-Component specification and PIM (Platform independent model) of RT-Component are discussed for platform permanence. Adopting standardized SDO model and inheriting the verified model, RT component will become more practical. Therefore we chose SDO (Super Distributed Object) specification published from OMG (Object Management Group) as a base component model. Finally RT-Component PIM based on SDO will be proposed.

**Key Words:** RT(Robot Technology), software component, middleware, robot system, standardization

### 1. はじめに

著者らは、システムの分析、設計、実装から成る、体系的なシステムインテグレーション (図 1) を支援する実装プラットフォーム「RT ミドルウェア」および、ソフトウェアのモジュール化のフレームワークとして「RT コンポーネント」の研究開発を行ってきた [1, 2, 3]。これにより、これまで人に属していたシステムインテグレーションの知識を、特定のコンポーネントモデルを有する実装プラットフォーム上に再構成することで、既存のシステムの構造が明確となり、システムの設計や分析に関する構造化されたパターンの抽出が可能になるものと期待できる。

本稿では、これまで提案してきた RT コンポーネントのモデルを、すでにソフトウェア標準化団体 OMG (Object Management Group) において標準化されている分散コンポーネントのための仕様 SDO (Super Distributed Object) に基づいて定義し、プラットフォーム非依存モデル (PIM: Platform Independent Mode) 化を行うことを提案する。

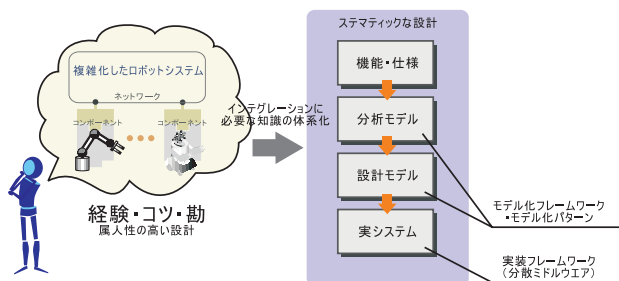


Fig.1 A Robot Systems Modeling Flow: The RT system should be modeled and designed through systematic design flow independent from the persons' experience and knowhow.

### 2. OMG における標準化と MDA

RT コンポーネントモデルをオープン化および標準化するため、著者らは OMG における活動を行ってきた [4]。本節では、ソフトウェア標準化団体 OMG および、OMG が提唱するモデル駆動アーキテクチャ MDA について概説する。

#### 2.1 OMG

OMG(Object Management Group) は 1989 年に設立された世界最大のソフトウェア標準化団体である。OMG が主導して策定した代表的標準仕様としては、分散オブジェクトミドルウェアである CORBA (Common Object Request Broker Architecture)、ソフトウェア分析/設計のためのモデリング言語である UML (Unified Modeling Language) などがある。

その他、各分野におけるソフトウェア標準の策定を行っており、特定のソフトウェア企業に依存しない中立の非営利団体として、オープンなプロセスによって各種標準を策定している。

#### 2.2 MDA と PIM, PSM

MDA (Model Driven Architecture) は OMG が提唱する、モデリング主導のシステムの開発、ライフサイクルの管理を実現するための参照アーキテクチャである。中心となるモデルは、プラットフォームに非依存な PIM (Platform Independent Model) とプラットフォーム依存モデル PSM (Platform Specific Model) の 2 階層から構成される。

PIM はプラットフォームに依存しないシステムのモデルであり、UML により特定の言語や OS、ミドルウェアなどに依存せず、かつ曖昧さを排除して構築されたモデルを指す。PSM は PIM から生成される各プラットフォームに特化したモデルであり、開発者はこのモデルを元に実装を行う。CORBA や.NET 等に特化したモデルが PIM からツールにより自動的に生成されることが想定されている。

MDA の技術は実用段階に入りつつあるものの、これをサポートするツール等はまだまだ少なく、一般的ではな

いという点は否めない。しかしながら、モデルを実装依存部分と非依存部分に分け定義する考え方は、RTシステム開発においても重要な指針となるものと考え採用した。

### 2.3 モデル化の重要性

著者らはこれまで、RT コンポーネントモデルを提案するとともに、実装として OpenRTM-aist を提供してきた [5]。実装を公開し多くのユーザに実際に使用してもらうことで、コンポーネントモデルとしての妥当性を検証することができる [6, 7, 8, 9, 10]。

OpenRTM-aist は、コンポーネントモデルを CORBA プラットフォーム上に実装したものであり、RT コンポーネントモデルから導き出される実装のひとつではない。コンポーネントモデルの仕様はオープンであり、その仕様に基づいた実装を誰でも自由に作成することができる。上述の PSM レベルで仕様が共通であれば、異なる実装同士でも相互運用することができる。すなわち、PSM とはプラットフォームレベルで相互運用性を保証するモデルである。

また、PIM と実装とは独立しているため、実装者は Web Service や Java RMI といった異なるプラットフォームに対しても PIM から生成される PSM に基づき実装することができる。実装プラットフォームが異なれば、相互運用性は実現できないが、共通のモデルに基づいた実装同士であれば、異なるプラットフォーム間のブリッジを容易に実現でき、ブリッジ経由での相互運用を行うことも可能である。

このような例は、軽量プロトコルにより PIC 等マイコンデバイスを RT ミドルウェアネットワークに参加させる目的で開発された RTComponent-Lite によっても実証されている [11]。また、著者らは CAN (Controller Area Network) 上に RT コンポーネントモデルに基づいたコンポーネントを実装し、さらに CORBA ベースの RT コンポーネントとの相互運用も可能であることを確認した。

## 3. SDO (Super Distributed Object)

SDO は分散化された様々なハードウェアデバイスやソフトウェアコンポーネントの論理的な表現を定義した規格であり、OMG (Object Management Group) により標準化されている。SDO が指す Super Distribution とは、非常に多くのオブジェクトそれ自身が自律的にタスクを処理する多数のオブジェクトをネットワーク上に配置することであり、それらオブジェクトを統合することによりユビキタスネットワークにおけるアプリケーションを実現することを想定している。

### 3.1 SDO の特徴

ユビキタスネットワークやユビキタスコンピューティングをターゲットとしている点において、SDO は Universal Plug and Play, HAVi, OSGi, ECHONET, Jini といった規約に類似したものである。しかしながら、SDO は MDA の考え方に基づき、PIM (Platform Independent Mode) で規定されているため、プラットフォームや実装技術に依存しないモデルおよび標準規約を提供している点が他の規約とは異なる [12]。

RT コンポーネントは、ネットワークに分散配置された個々のタスクを自律的に処理するコンポーネントとして提案を行ってきた。この考え方は、SDO の自律的なオブジェクトの分散配置という考え方と共通のものである。さらに、RT コンポーネントも、プラットフォーム

Table 1 SDO resource data model.

Device Profile	SDO がどういったデバイスあるいはソフトウェア機能を提供するかを表す情報が格納される。
Status	SDO 自身が持つ状態を表す。状態は複数種類も持つことができる。
Service Profile	SDO 自身が外部に対して提供するサービスのプロファイル。サービスは複数持つことができる。
Configuration Profile	SDO の内部設定情報。
Organization	SDO は複数の SDO が集まり、グループを作ることができる。これを Organization と呼ぶ。
Organization Profile	Organization に関するプロファイル情報。

や実装技術に依存しないモデル化を指向しており、SDO の考え方と共通する部分が多くある。

著者らは、SDO を検討した結果、RT コンポーネントを SDO を継承した形でモデル化、および標準化することにより、SDO に基づいたシステムと相互運用を図りつつ、ロボットシステムに特化した機能を付与できると考えた。また、既存の検証済みのモデルを利用することにより、多くの部分のモデルの妥当性が保証されているため、より実用的なモデルを提案できると考えた。

### 3.2 SDO リソースデータモデル

SDO は上述したように、ネットワークに分散する多様なデバイスやソフトウェアの論理的な表現を定義することができる。これら、SDO 化されたデバイスやソフトウェアのプロファイルは、リソースデータモデルとして定義されている。リソースデータモデルの UML クラス図を図 2 に示す。

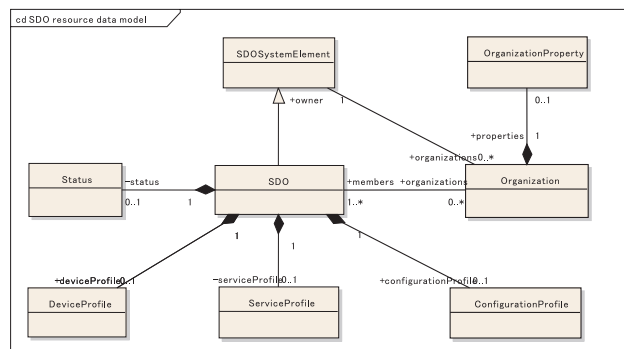


Fig.2 SDO resource data model.

各リソースデータの詳細について、OMG から公表されている仕様 [12] に PIM として詳細に定義されている。

SDO のリソースデータモデルは大まかに分けて表 1 に示すプロファイル情報が含まれる。以上のプロファイル情報により SDO が規定され、これらのデータにアクセスするためのインターフェースも PIM に基づき定められている。

## 4. RT-Component の標準化

本節では RT コンポーネントの OMG における標準化と、RT コンポーネントのプラットフォーム非依存モ

デル (PIM) を示す。

#### 4.1 SDO と RT コンポーネントの関係

RT コンポーネントは OMG における標準化にあたって、すでに OMG で標準化されている 上述の SDO のモデルを一部継承する形でモデル化することとした (図 3)。本稿で提案する RT コンポーネント PIM は、SDO のロボットドメインに特化した仕様として位置づけられる。

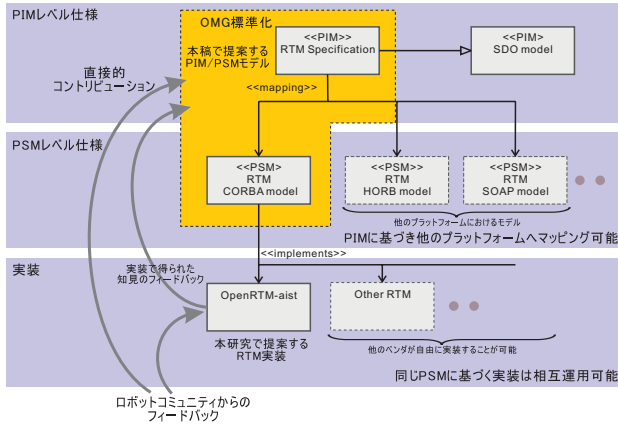


Fig.3 The relation among RTComponent's PIM, PSM and implementation (OpenRTM-aist).

#### 4.2 RT コンポーネント PIM

RT コンポーネントの PIM は大きく分けて、

リソースデータモデル RT コンポーネントが持つ機能のプロファイル情報を記述するためのデータモデル。

インターフェース リソースデータにアクセスするための各種インターフェースのモデル。

から構成される。

図 4 に RT コンポーネントのリソースデータモデルを、図 5 に RT コンポーネントオブジェクトが持つインターフェース (クラス図) を示す。それぞれ、属性と操作の詳細は紙面の都合上割愛する。

##### 4.2.1 リソースデータモデル

図 4 に示すように、RT コンポーネント自身のプロファイル情報は SDO のプロファイル情報を継承しており、SDO をロボット用分散オブジェクトとして特化したものとして定義される。RTComponentProfile は RT コンポーネントアーキテクチャに基づいた、Activity の状態 (これは RTComponentProfile 自身に含まれるため図に表れない)、InPort, OutPort, RTC Service, RTCS Consumer 等それぞれのプロファイル情報を保持している。

RTCInPortProfile および RTCOutPortProfile はそれぞれ、そのポートが扱うデータ型等の情報を保持するデータである。また、InPort と OutPort は相互の接続に関して、接続プロファイル (Subscription Profile) を持つ。Subscription Profile はお互いのオブジェクトの参照と接続タイプ等の接続に関する情報を保持し、InPort と OutPort の双方向関連を形成するデータとなる。

RTCServiceProfile は、サービスが提供するサービスのタイプ等の情報を保持し、一方 RTCCustomerProfile

はコンシューマーが要求するサービスのタイプ等の情報を保持する。これらは、SDO に利用可能な ServiceProfile データがあるためこれを継承する。また、サービスとコンシューマーも InPort/OutPort と同様に接続 (依存) 関係を持つため、その接続に関するデータを保持する RTCCustomerProfile を持つ。RTCCustomerProfile は、サービスおよびコンシューマーへの参照を保持し、双方向関連を形成する。

##### 4.2.2 インターフェース

上述のリソースデータへのアクセスや、その他の操作はインターフェース図 (図 5) に示す構成で定義される。RTComponent クラスは RT コンポーネントの本体となるオブジェクトであり、全てのリソースデータやオブジェクトへの参照はここを起点に取得される。RTCActivityAdmin および RTCActivity は Activity への操作を提供するサービスインターフェースである。

RTCInPortAny および RTCOutPortAny は Any 型への InPort および OutPort へのマッピングの一例を示す。実際には、データ型毎に RTCPortBase を継承する形でオブジェクトが構築される。

RTCService は、RT コンポーネントが提供するサービスインターフェースの基底であり、実際にはコンポーネントの開発者が RTCService クラスを継承して独自のサービスインターフェースを記述する。RTCCustomer は、コンポーネントが依存するサービスのプロファイル情報を取得したり、サービスへの参照を与えたりする操作を提供するインターフェースである。

また、本稿では詳細は述べないが、Activity の状態遷移や、各種オブジェクトの相互作用時のシーケンス図等により RT コンポーネントの PIM が記述される [13]。

#### 5. おわりに

本稿では、コンポーネント指向 RT システムのインテグレーションのためのプラットフォームとして、RT コンポーネントの PIM を提案した。プラットフォームは永続的に利用できることが望ましく、技術の進歩に追従できるように、実装技術に独立な方法でモデル化し、そのモデルを維持していくことが望まれる。既存の OMG 標準である SDO 仕様を、ロボットシステムのためのコンポーネントモデルとして拡張する形で RT コンポーネントモデルを再定義した。

RT コンポーネントモデルのうち、SDO モデルが提供する基本機能のモデルの妥当性は保証されており、RT コンポーネントに特有な機能についてのみ妥当性の検証を行えば良い。

今後は、本稿で提案した PIM を詳細にわたり定義したうえで、OMG における標準化活動を行うとともに、我々が提供する実装 OpenRTM-aist をこの PIM に完全に準拠したものにし、様々なシステムに適用することで、モデルの妥当性を検証する予定である。

#### 参考文献

- [1] 末廣 尚士, 北垣 高成, 神徳 徹雄, 尹 祐根, 安藤 慶昭, "RT コンポーネントの実装例.RT ミドルウェアの基本機能に関する研究開発 (その 1)", 第 21 回 日本ロボット学会学術講演会予稿集, p.1F27, 2003.09
- [2] 末廣 尚士, 北垣 高成, 神徳 徹雄, 尹 祐根, 安藤 慶昭, "RT コンポーネントの実装例.RT ミドルウェアの基本機能に関する研究開発 (その 2)", 第 21 回 日本ロボット学会学術講演会予稿集, p.1F28, 2003.09

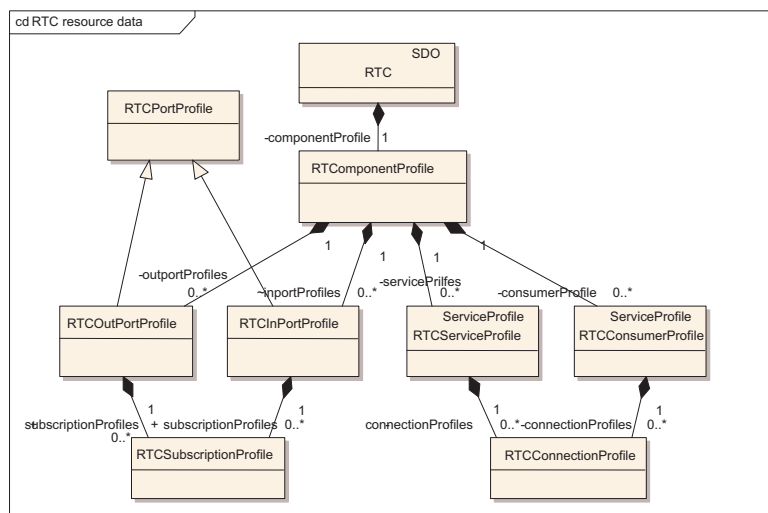


Fig.4 The RTComponent's PIM (Resource data model).

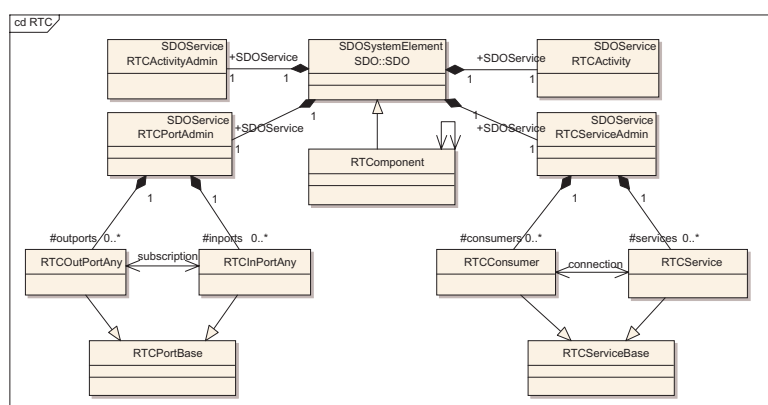


Fig.5 The RTComponent's PIM (Class diagram).

- [3] 安藤 慶昭, 末廣 尚士, 北垣 高成, 神徳 徹雄, 尹 祐根, "RT 要素のモジュール化および RT コンポーネントの実装", 第 9 回 ロボティクスシンポジア, pp.288-293, 2004.03
- [4] 神徳 徹雄, 末廣 尚士, 安藤 慶昭, 尹 祐根, 北垣 高成, 平井 成興, 谷江 和雄, 水川 真, "RT ミドルウェア標準化活動への誘い", 日本機械学会 ロボティクス・メカトロニクス講演会 2005, p.2P1-N-066, 2005.06, 神戸
- [5] OpenRTM-aist and OpenRTM-aist tutorial web page, <http://www.is.aist.go.jp/rt/OpenRTM-aist-Tutorial>
- [6] 稲村 渡, 藤井 正和, 坂野 肇, 田中 康仁, "JAUS および RT ミドルウェアを規範とした分散アーキテクチャの提案", 第 6 回 計測自動制御学会システムインテグレーション部門講演会, p.3C2-1, 2005
- [7] 池添 明宏, 松久 孝志, 中本 啓之, 長瀬 雅之, "組込みシステム向け RT ミドルウェアの開発", 第 6 回 計測自動制御学会システムインテグレーション部門講演会, p.3C2-5, 2005
- [8] 池添 明宏, 中本 啓之, 羽田 芳朗, 高瀬 國克, "RT ミドルウェアによるレーザレンジファインダコンポーネントの構築", 第 6 回 計測自動制御学会システムインテグレーション部門講演会, p.3C3-2, 2005
- [9] 山下 智輝, 伊東 一郎, 熊沢 四郎, 早坂 友輔, 江龍晃, "RT ミドルウェアによる農作業支援ロボットシステムの開発 -OpenRTM-aist を適用した収穫物運搬ロボットの試作-", 第 6 回 計測自動制御学会システムインテグレーション部門講演会, p.3C3-3, 2005
- [10] 土屋 裕, 水川 真, 末廣 尚士, "RT ミドルウェアを用いた PAS 分散処理系の考察", 第 6 回 計測自動制御学会システムインテグレーション部門講演会, p.3C3-4, 2005
- [11] 安藤 慶昭, 鈴木 喬, 大原 賢一, 大場 光太郎, 谷江 和雄, "組込機器のための軽量 RT コンポーネント:RTComponent-Lite", 計測自動制御学会システムインテグレーション部門講演会 2005 (SI2005), p.3C2-2, 2005.12
- [12] OMG public document, "Platform Independent Model (PIM) & Platform Specific Model (PSM) for Super Distributed Objects (SDO)", formal/04-11-01: PIM and PSM for Super Distributed Objects, v1.0, 2004.11
- [13] OMG public document, PIM (Platform Independent Model) and PSM (Platform Specific Model) for RTComponent (Robot Technology Component), Initial Submission to the OMG RFP: Robot Technology Components (RTCs)(ptc/2005-09-01), mars/2006-01-05, 01/23/2006