

(社) 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

SICE System Integration Division
Annual Conference

SICE
SI 2004

講演概要集

SICE[®]

2004.12.17-19
つくば国際会議場

RT コンポーネントによるロボットシステム開発

-RT ミドルウェアの基本機能に関する研究開発 (その 10)-

○安藤慶昭 (産総研), 末廣尚士 (産総研), 北垣高成 (産総研)
 神徳徹雄 (産総研), 尹祐根 (産総研)

Robot System Development using RT Components - R & D of RT Middleware Fundamental Functions (Part 10) -

*Noriaki ANDO (AIST), Takashi SUEHIRO (AIST), Kosei KITAGAKI (AIST),
 Tetsuo KOTOKU (AIST) and Woo-keun YOON (AIST)

Abstract— We have developed a framework of RT-component which promotes application of Robot Technology (RT) in various field. In this paper, we introduce a component development flow and method. A system development methodology using RTComponent, RTComponent framework with some method will be shown. Conclusion and future work will be described.

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

1. はじめに

ロボット要素技術 (RT[1]) のソフトウェアモジュール化を実現し、再利用性を高めるためのソフトウェアプラットフォーム「RT ミドルウェア」の基本機能に関する研究開発を行っている。

著者らは、これまでコンポーネントの中核部分である RT コンポーネントのアーキテクチャを提案してきた []。本稿では、RT コンポーネントの実際の作成方法、作成された複数の RT コンポーネントを組み合わせ、実際のシステムを構築する際に必要な機能について議論を行う。

2. RT コンポーネント開発の流れ

RT ミドルウェアは、コンポーネントを開発したいユーザ (コンポーネントデベロッパ) が持つ既存のソフトウェア資産、あるいは新たに作成したソフトウェアを容易にコンポーネント化するためのフレームワークを持つ [2, 3]。

図 1 に RT コンポーネント作成の流れを示す。

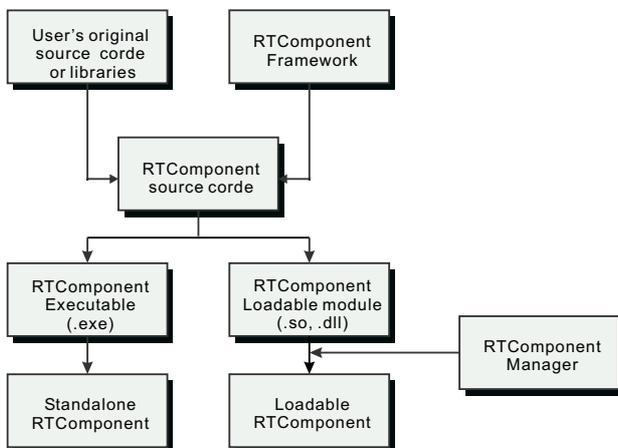


Fig.1 RT コンポーネントの開発フロー

コンポーネントデベロッパは、既存のソフトウェア

資産の関数・クラス等をコンポーネントフレームワークに埋め込むことで、ソフトウェア部品として再利用可能な RT コンポーネントを容易に作成し、分散環境で利用することができる。

このように作成された RT コンポーネントは、単体で実行可能な「スタンドアロンコンポーネント」および、動的にロードされ実行される「ロードブルモジュールコンポーネント」の 2 つの形式として生成することができる。「スタンドアロンコンポーネント」および「ロードブルモジュールコンポーネント」は以下のような特徴を持つ。

スタンドアロンコンポーネント 単体で実行可能なバイナリ形式のコンポーネント。1 プロセスにつき原則として 1 種類のコンポーネント (複数のインスタンスを持つことができる。) に対応する。他のコンポーネントとの通信は CORBA 経由で行うため、速度的には不利であるが、他のコンポーネントがエラーで停止しても、影響を被らないため安定性では有利となる。

ロードブルモジュールコンポーネント マネージャにより動的にロードされ、実体化されるコンポーネント。1 つのプロセスに対して複数の種類の複数のインスタンスを持つことができる。コンポーネント間の通信はプロセス内部の単なる関数呼び出しとなる (これは ORB 実装に依存するが、ほとんどの実装では単なる関数呼び出しに変換される。) ため、速度的に有利となる。しかしながら、プロセス内のコンポーネントがセグメント違反等の致命的エラーを起こすと、すべてのコンポーネントが停止するため、安定でないコンポーネントと同時に動作させると全体が不安定になるデメリットがある。

一旦インターフェース等の仕様が確定し、デバッグされ安定に動作するコンポーネントが完成すれば、そのコンポーネントを利用するユーザは、そのコンポー

ネットのソースコードに手つけることなしに、他のコンポーネントと組み合わせ、ロボットシステムを構築することができる。したがって、デベロッパはソースコード、ロードブルモジュール形式、実行ファイル形式の任意の形式で配布することができる。

3. コンポーネントのアセンブル

1つのコンポーネントそれ自体では、ほとんどの場合、何らかの役に立つ仕事をするシステムを構成することはできない。コンポーネントの利用者は、複数のコンポーネントを部品として組み合わせることによりシステムを構築する。コンポーネントを連携させる方法には、図2に示すように、

- アセンブリ GUI
- XMLにより記述されたアセンブル情報
- 他の RT コンポーネント
- スクリプト言語
- 一般的なプログラム

等がある。

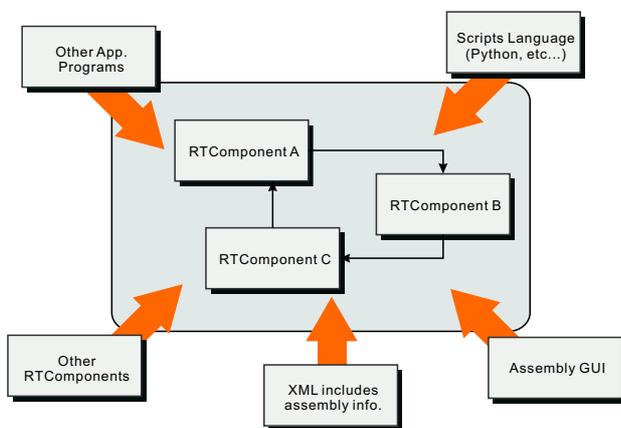


Fig.2 コンポーネントのアセンブル

アセンブリ GUIは、制御ブロック線図のように、コンポーネントの InPort/OutPort の接続や、コンポーネントの ON/OFF を GUI 画面上から操作することができるインターフェースであり、ユーザは視覚的にコンポーネントを組み合わせることができる。この接続情報は、XML形式で保存され、このXMLファイルを用いることで以前の接続情報を再現することができる。

また、コンポーネントの接続や ON/OFF は、他のコンポーネントからも操作することができるため、動的な接続の変更などをプログラムから行うことができる。

RT コンポーネントは現在、スクリプト言語として Python をサポートしている。Python のようなスクリプト言語を用いることにより、コンパイルすることなくラビッドプロトタイピングが行える。また、高速な実行速度を必要としない上位のアプリケーションレベルのシステムを構築する際には、スクリプト言語のような高級言語がより適している。

また、RT ミドルウェアは分散オブジェクトミドルウェア：CORBA を利用しており、通常の CORBA の使用方法を用いれば、通常のアプリケーションプログラムからコンポーネントを制御することができる。

4. 複合コンポーネント

コンポーネントの連携の密度をより高くしたい、あるいは複数のコンポーネントをあたかも一つのコンポーネントのように取り扱いたい場合がある。このような場合には、複合コンポーネントにより、複数のコンポーネントを一つのコンポーネントとすることができる。

複合コンポーネントには、大きく分けて「非同期複合コンポーネント」「同期複合コンポーネント」がある。非同期複合コンポーネントは、

- 個々の InPort/OutPort を複合コンポーネントの InPort/OutPort として扱う。
- 個々のアクティビティはそれぞれスレッドを持ち並列に動作する。
- 個々のアクティビティはそれぞれ状態を持ち、すべては必ずしも一致しない。

同期複合コンポーネントは、

- 個々の InPort/OutPort を複合コンポーネントの InPort/OutPort として扱う。
- 個々のアクティビティは一つのスレッドを持ち直列に動作する。
- 複合コンポーネントとしての状態を持ち、すべてのコンポーネントの状態が一致する。

といった特徴がある。

コンポーネントの粒度が小さく、連携の密度が高い場合には複数のコンポーネントを多数同期させリアルタイム実行する必要がある場合には、同期複合コンポーネントの利用が適している場合が多い。たとえば、マニピュレータと力センサをそれぞれ個別のコンポーネントとし、これら二つのコンポーネントを用いてマニピュレータの力制御を行おうとする場合には、個々のコンポーネントを並列にリアルタイム化するだけでは安定な力制御を行うことは出来ない。RT ミドルウェアは、まったく個別に作成されたコンポーネント同士を組み合わせ、一つの RT システムを容易に構築するフレームワークを提供している。

5. おわりに

本稿では、複数の RT コンポーネントを用いて、システムを構築するための手法、および複合コンポーネントによるシステムの構成手法について検討を行った。今後は、本稿で示した RT コンポーネントおよびコンポーネントマネージャを実際のロボットに対して適用し、実際の実時間制御等での利用可能性を検討する。また、コンポーネント利用の利便性を向上させる種々のサービスや、各種ツールを開発する予定である。同時に、外部に対してこれらをリリースし実際に様々なロボットに対して使ってもらうことでフィードバックを得てゆきたいと考えている。

参考文献

- [1] 「21世紀におけるロボット社会創造のための技術戦略調査報告書」, (社)日本機械工業連合会, (社)日本ロボット工業会, 2001.
- [2] 北垣, 末廣, 神徳, 平井, 谷江: “RT ミドルウェア技術基盤の研究開発について -ロボット機能発現のために必要な要素技術開発-”, ロボティックシンポジウム予稿集, pp.487-492, 2003.
- [3] 安藤, 末廣, 北垣, 神徳, 尹, 「RT 要素のモジュール化および RT コンポーネントの実装」, ロボティックシンポジウム予稿集, pp.288-293, 2004.