

RTミドルウェアサマーキャンプ2011

RTミドルウェア概要

日時: 2011年8月29日(月) 14:20- 15:30

場所: 産業技術総合研究所

つくば第2事業所 本部・情報棟1階 交流会議室

(独)産業技術総合研究所 知能システム研究部門

栗原眞二、神徳徹雄、安藤慶昭



<http://www.openrtm.org/>

概要

- RTミドルウェア概要
- OpenRTM-aistについて
- 幾つかのテクニック

本日の狙い:

RTミドルウェア技術のコンセプトを理解し、皆でロボット技術の共有と再利用を考えていただくきっかけとする。

RTとは?

- RT = Robot Technology cf. IT
 - #Real-time
 - 単体のロボットだけでなく、さまざまなロボット技術に基づく機能要素をも含む (センサ、アクチュエータ, 制御スキーム、アルゴリズム、etc....)

産総研版RTミドルウェア

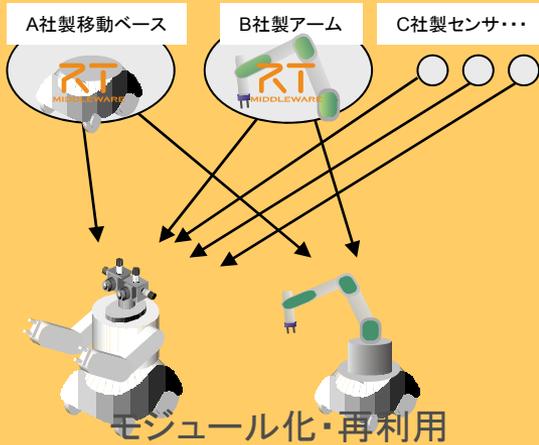
OpenRTM-aist

- RT-Middleware (RTM)
 - RT要素のインテグレーションのためのミドルウェア
- RT-Component (RTC)
 - RT-Middlewareにおけるソフトウェアの基本単位

RTミドルウェアの目的

モジュール化による問題解決

コストの問題



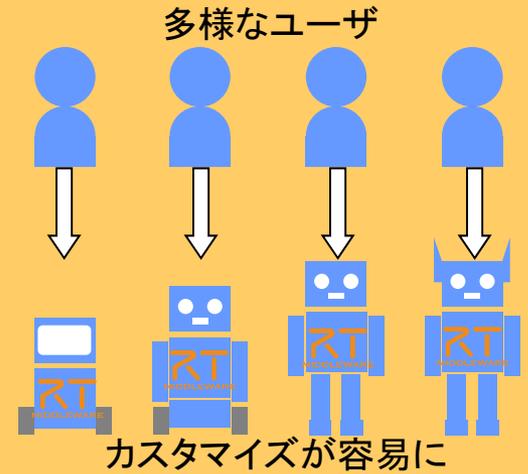
ロボットの低コスト化

技術の問題



最新技術を利用可能

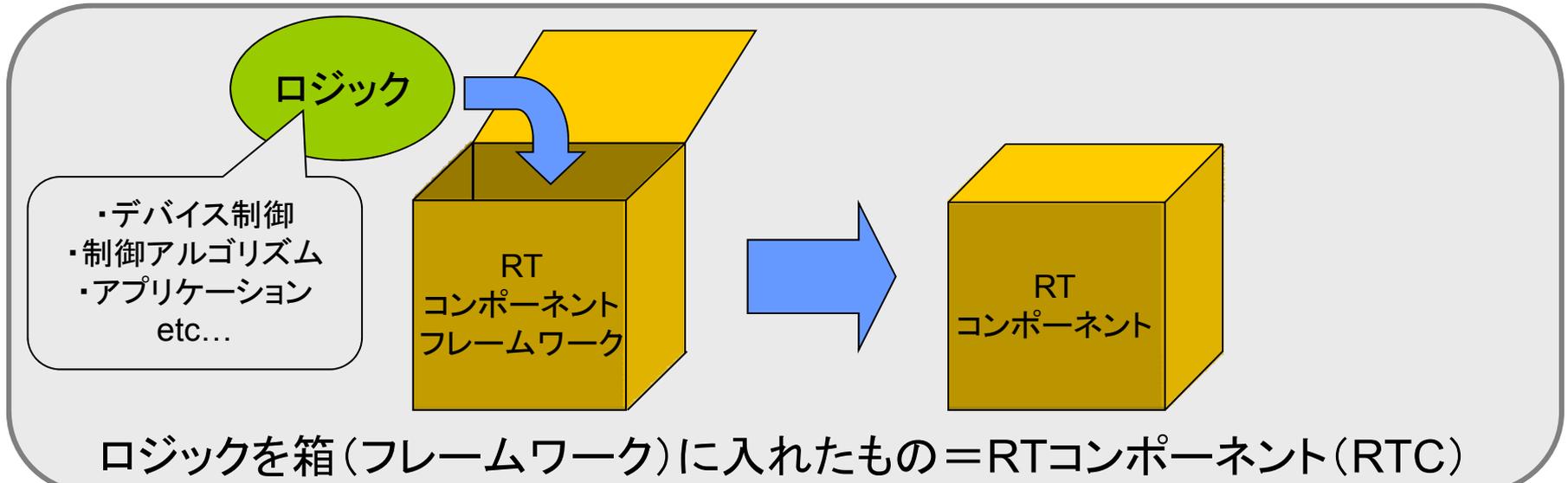
ニーズの問題



多様なニーズに対応

ロボットシステムインテグレーションによるイノベーション

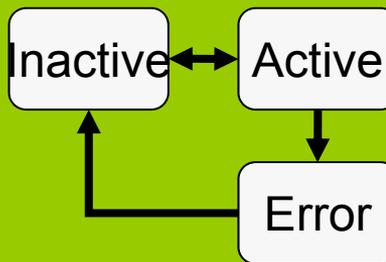
RTミドルウェアとRTコンポーネント



RTコンポーネントの主な機能

アクティビティ・実行コンテキスト

共通の状態遷移



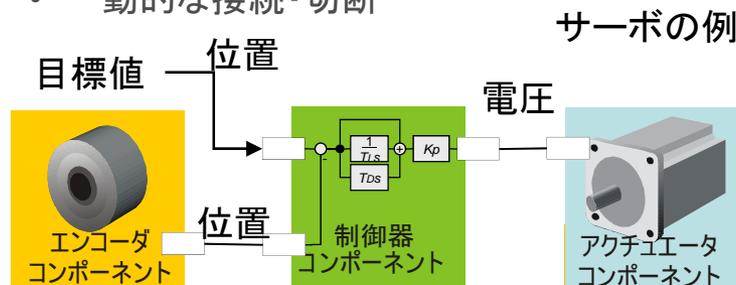
複合実行



ライフサイクルの管理・コアロジックの実行

データポート

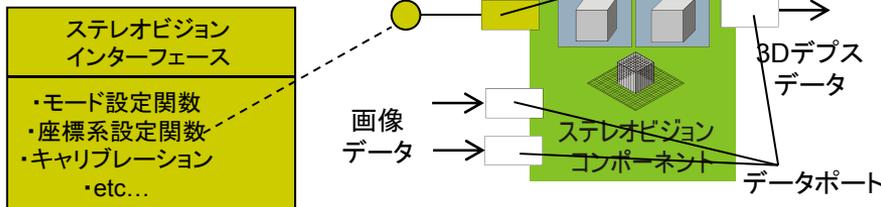
- データ指向ポート
- 連続的なデータの送受信
- 動的な接続・切断



データ指向通信機能

サービスポート

- 定義可能なインターフェースを持つ
- 内部の詳細な機能にアクセス
 - パラメータ取得・設定
 - モード切替
 - etc...



サービス指向相互作用機能

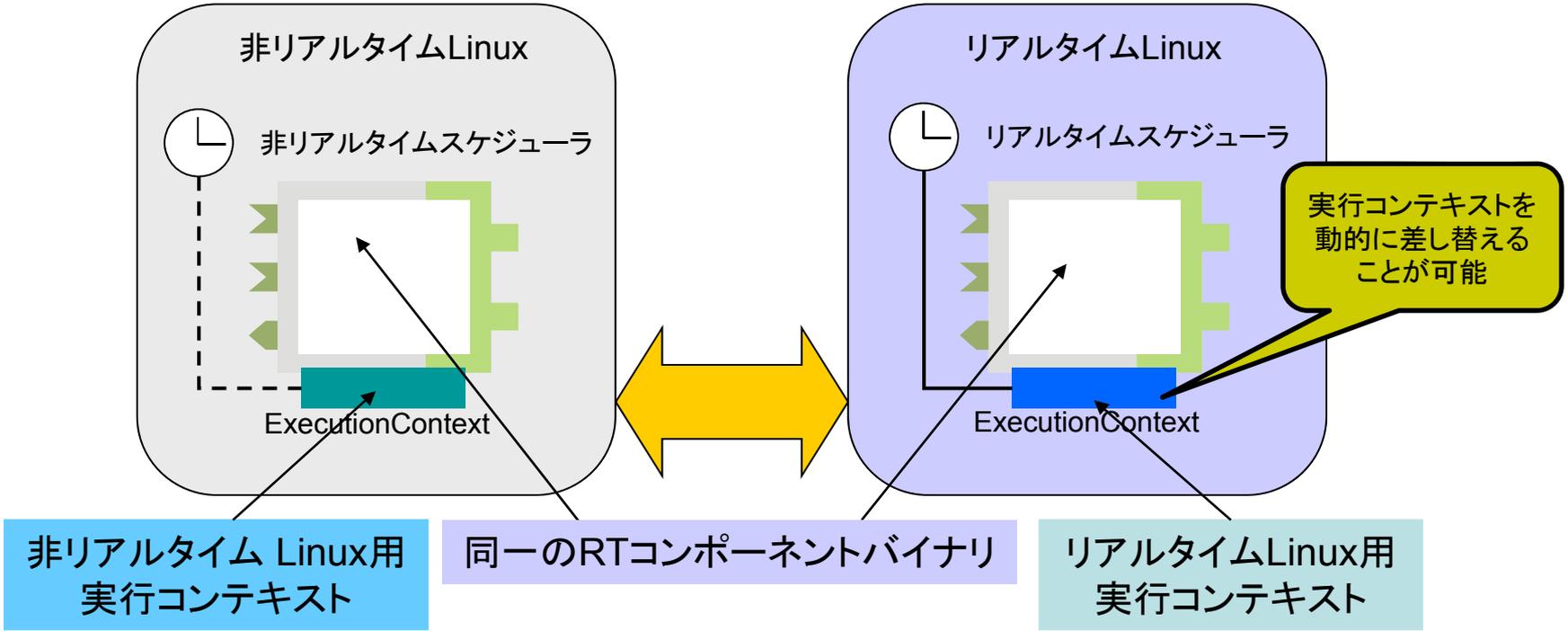
コンフィギュレーション

- パラメータを保持する仕組み
- いくつかのセットを保持可能
- 実行時に動的に変更可能

複数のセットを動作時に切り替えて使用可能

セット名	名前				
	値				
セット名	名前				
	値				

リアルタイム実行コンテキスト

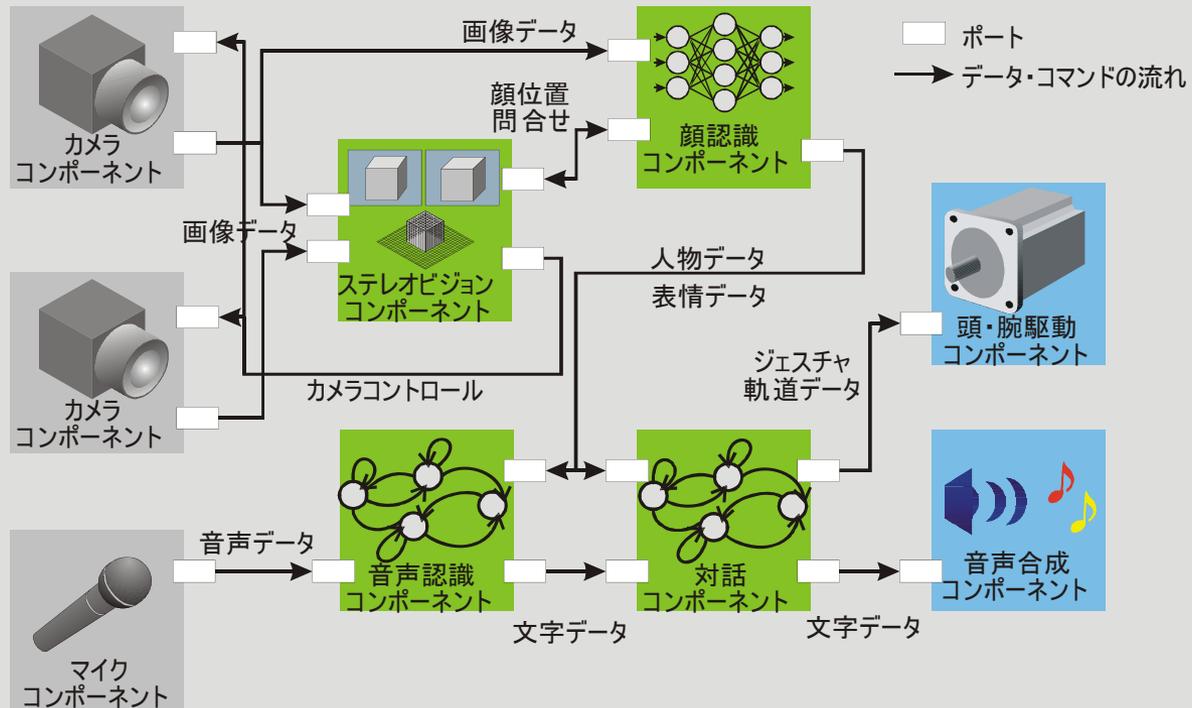


非リアルタイムLinux環境で作られたRTコンポーネントを再コンパイルせずにリアルタイムLinux上でリアルタイム実行可能

RTCの分割と連携



ロボット体内のコンポーネントによる構成例



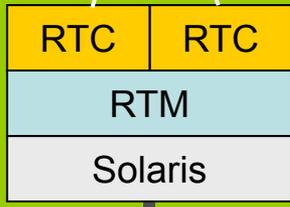
(モジュール)情報の隠蔽と公開のルールが重要

RTミドルウェアによる分散システム

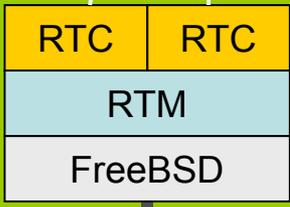
RTMにより、ネットワーク上に分散するRTCをOS・言語の壁を越えて接続することができる。

ネットワーク

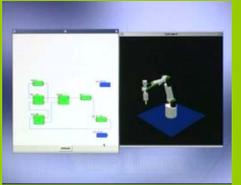
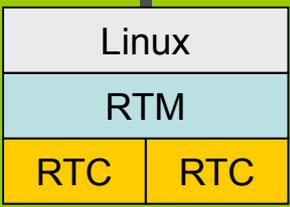
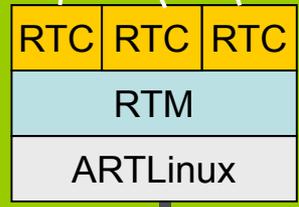
ロボットA



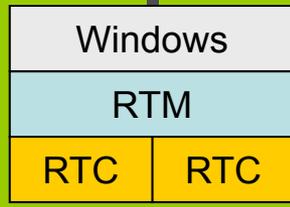
ロボットB



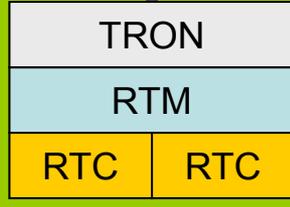
ロボットC



アプリケーション



操作デバイス



センサ

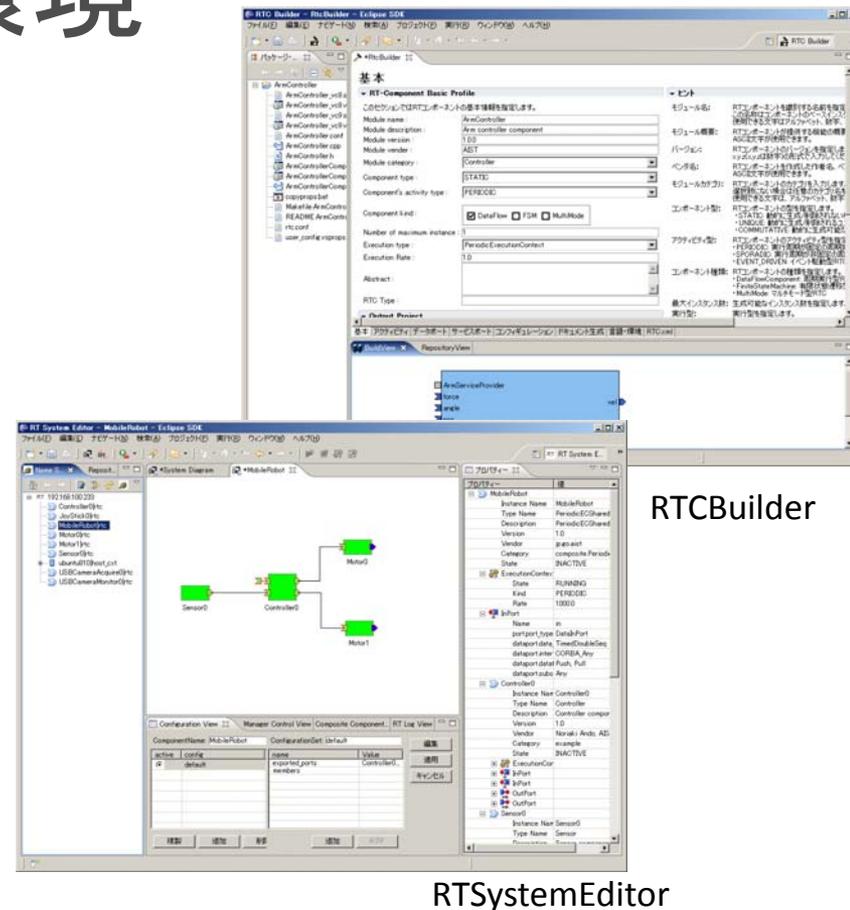
RTC同士の接続は、プログラム実行中に動的に行うことができる。

OpenRTM-aist

- コンポーネントフレームワーク + ミドルウェアライブラリ
- コンポーネントインターフェース:
 - OMG Robotic Technology Component Specification ver1.0 準拠
- OS
 - 公式: FreeBSD, Linux (Fedora, Debian, Ubuntu, Vine, Scientific), Windows
 - 非公式: Mac OS X, uITRON, T-Kernel, VxWorks
- 言語:
 - C++ (1.1.0), Python (1.0.0), Java (1.0.0)
 - .NET (implemented by SEC)
- CPU アーキテクチャ (動作実績):
 - i386, ARM9, PPC, SH4
 - PIC, dsPIC, H8 (RTC-Lite)
- ツール (Eclipse プラグイン)
 - テンプレートソースジェネレータ: rtc-template、RTCBuilder
 - システムインテグレーションツール: RTSystemEditor
 - その他
 - Pattern weaver for RT-Middleware (株式会社テクノロジックアートより発売中)

開発環境

- RTCBuilder (GUI版)
- rtc-template (コマンドライン版)
 - RTコンポーネントのテンプレートコードジェネレータ
 - GUI画面で必要事項を入力
 - C++, Python, Java, C#等のコードを自動生成
- RTSystemEditor (GUI版)
- RTShell (コマンドライン版)
 - ネットワーク上のすべてのコンポーネントの操作が可能
 - コンポーネントのON/OFF、パラメータの変更、状態監視
 - コンポーネント間の接続



RTCBuilder

RTSystemEditor

RTC・RTM統合開発環境の整備
 RTC設計・実装・デバッグ、RTMによるインテグレーション・デバッグまでを一貫して行うことができる統合開発環境をEclipse上に構築

OpenRTMの利点

- 共通コンポーネントフレームワークを提供
 - OMG標準
 - コールバックベースの枠組み、共通状態マシン、複合化に対応
 - 大部分のコード生成を自動化
- 多言語対応
 - C++, Java, Python, .NET (by SEC)
- 多様なOSへのネイティブ対応
 - FreeBSD, Linux, Mac OS X, Windows
 - 試験的:TOPPERS, T-Kernel, VxWorks
- ツールの提供
 - Eclipseベースのツール群 (RTCB, RTSE)
 - コマンドラインツール群 (rtchell)
- デュアルライセンス (EPLと個別ライセンス)
 - RTCにはライセンスが及ばない(RTCのバイナリ供給が可能に)
 - 商用化、事業化、組込み用途には個別ライセンスで対応

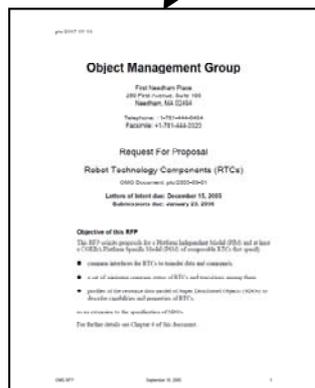
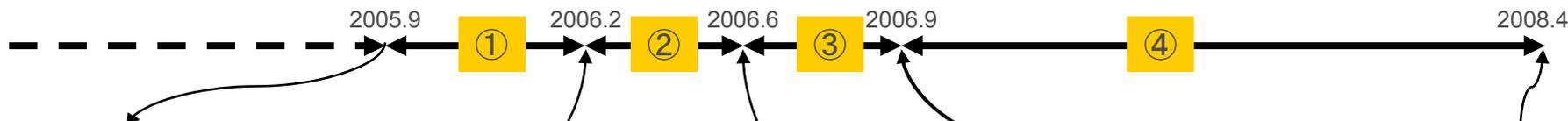
OMGにおける標準化

① OpenRTM-aistに基づく
初期提案の作成

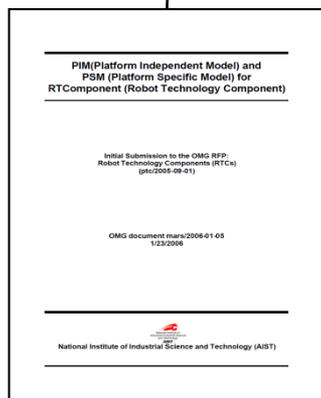
② 共同提案書作成
のための詳細を議論

③ 標準作業部会
への提出・採択

④ 最終文書化委員会
における整合性の議論



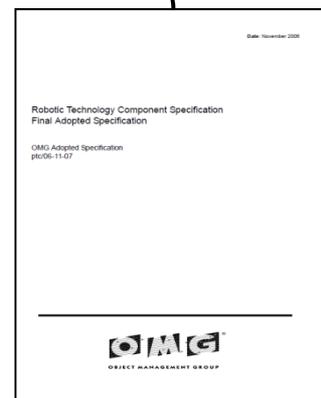
RFP発行
(RFP=標準仕様
提案の呼びかけ)



Initial Submission



Joint Submission



Adopted Specification

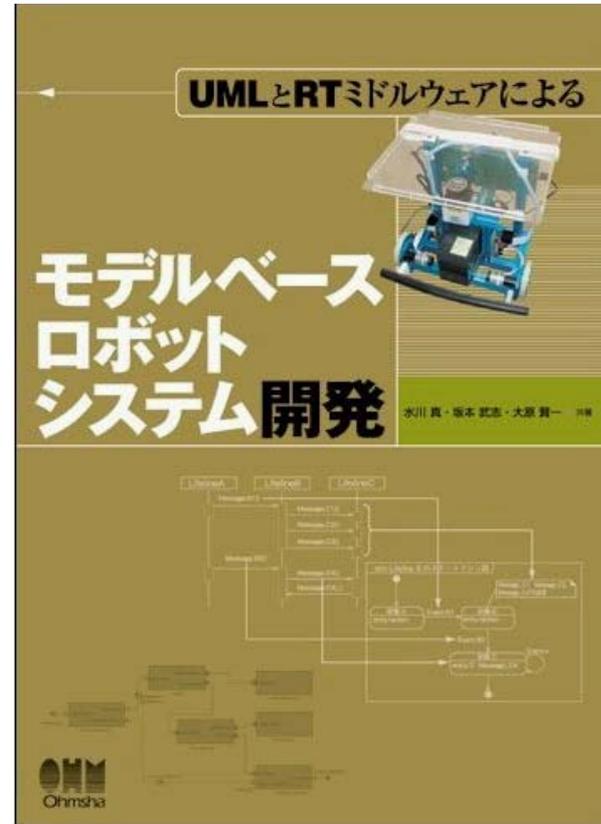
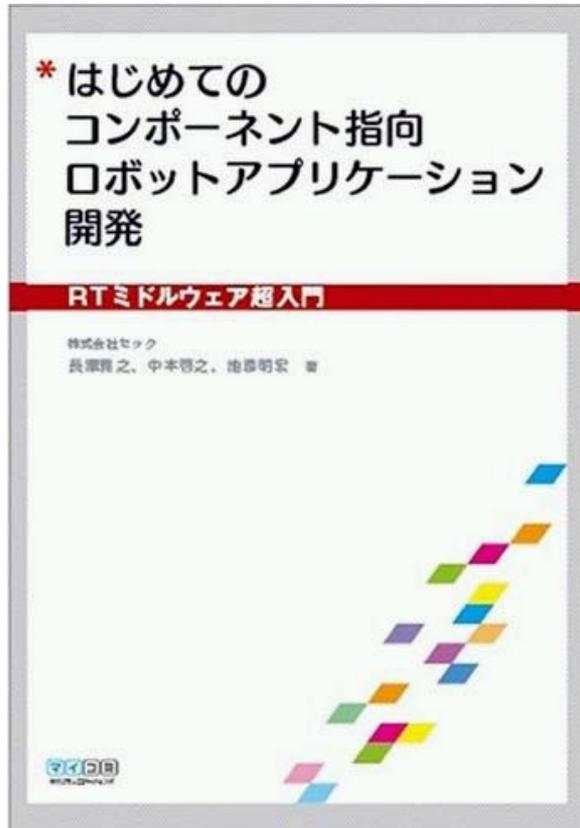
標準仕様の
公式リリース

多様な実装

- OpenRTM-aist: 産総研実装
 - C++, Python, Java の3種類
- OpenRTM.NET: 株式会社SEC版実装
 - .NET版: VB, C#
- 韓国ETRI
 - OPRoS コンポーネント: 一部準拠
- PALRO: 富士ソフト
 - 小型ヒューマノイド制御フレームワークがC++ソースレベルでOpenRTM互換
- GostaiRTC: 仏GOSTAI & Thales
 - OMG RTC Local PSM に準拠



書籍 (1)



- はじめてのコンポーネント指向ロボットアプリケーション開発 ~RTミドルウェア超入門~
- 長瀬 雅之、中本 啓之、池添 明宏 著

- UMLとRTミドルウェアによるモデルベースロボットシステム開発
- 水川 真, 大原 賢一, 坂本 武志 著

書籍(2)

2010年(平成22年)6月15日発行(通巻203号) 年10回発行(1,3,4,5,6,7,9,10,11,12月)

ISSN 0289-182

URL : <http://www.rsj.or.jp/>

Journal of the Robotics Society of Japan

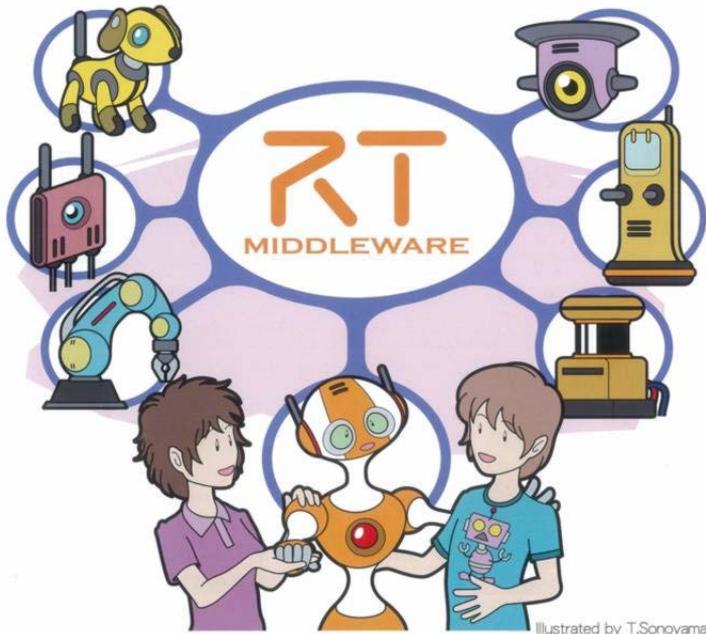
日本ロボット学会誌



June 2010
Vol.28 No.5

[特集]

使えるRTミドルウェア



Illustrated by T.Sonoyama

RSJ 社団法人 日本ロボット学会

「使えるRTミドルウェア」特集号
日本ロボット学会論文誌 vol.28, no.5

再利用性の高いロボットシステムを
RTミドルウェアを用いてどう構築する
かという“方法論”に関する実践
的な特集号

http://www.openrtm.org/OpenRTM-aist/html/JRSJ2010_28_5.html

新Webページ

ダウンロードサイトから、ユーザコミュニティページへ

- 新メーリングリスト
- フォーラムの設置
 - メーリングリストとの相互乗り入れ
- プロジェクト公開ページ
 - ユーザが自分のRTCやツールを自由に公開可能
- Wikiページ
 - ユーザが自由に編集できるページ
 - ちょっとしたノウハウ等の共有

まとめ

- ロボット用ミドルウェア: OpenRTM
 - ロボットに適した共通フレームワークの提供
 - OMG国際標準
 - 多様な実装、多様な言語、OSに対応

初心者のための テクニック

テクニック(分からないこと)



- ホームページから情報を探す
- **検索エンジンで情報を探す**
(OpenRTMの問題でないことも多い)
- 近くのOpenRTMユーザに聞いてみる
- ユーザメーリングリストで、**自分の状況をきちんと説明して**質問する。皆から貰った情報を結果とともに**整理して報告**する。

[ホーム](#) » [コミュニティ](#) » [メーリングリスト](#)

テクニック(日本語)



日本語のユーザ名(ログインアカウント)

現在のところ、日本語のユーザ名を使うと通信に利用している分散ミドルウェア(OmniORB)が正常に動作しない。

プログラムの再利用

外国で開発されたプログラムやUNIX用のプログラムの再利用をするときには、「日本語の利用」や「スペースを含むフォルダ名」が想定されていない。

ユーザ名、フォルダ名に日本語を使わない

テクニック(資源を考慮)



計算資源(計算機の負荷)

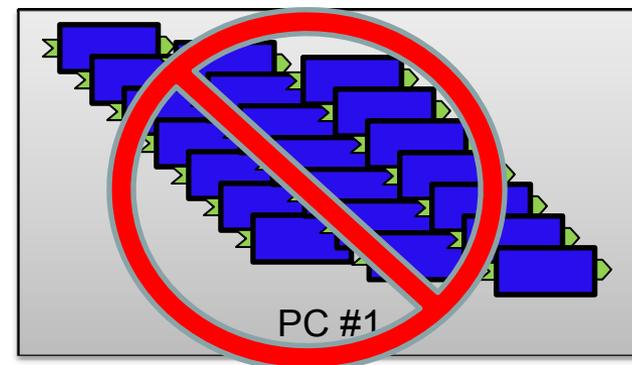
不必要に実行周期が早くないか？ 計算機
の能力を超えた数のRTCを載せていないか？

rtc.conf

```
exec_cxt.periodic.rate:10
実行コンテキストの周波数[Hz]を指定。
::有効範囲:(0, 1000000].
::デフォルト:1000.
```

- センサ出力が80msec毎なのに1msecでループを無駄に回していないか？
- OSのプロセス切り替え周期設定が10msecなのに10KHzで回そうとしていないか？

数多くの画像処理RTC



Pentium3-300MHz

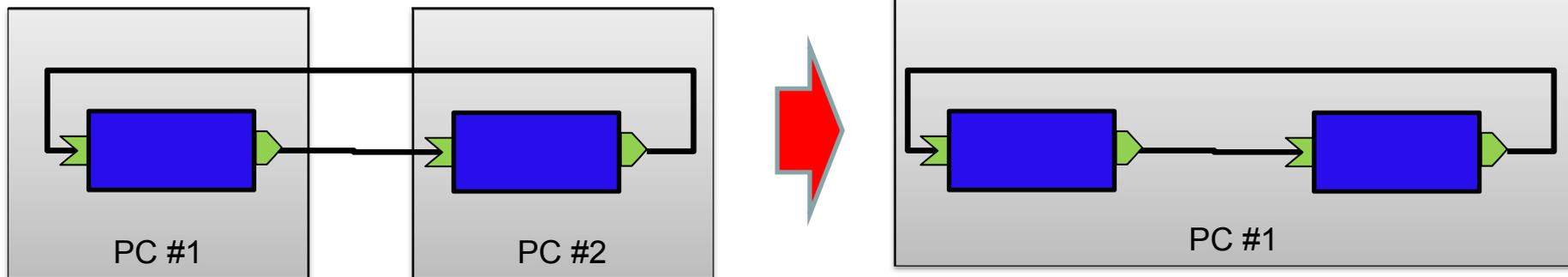
RTミドルウェアは魔法ではない

テクニック(資源を考慮)



実時間(リアルタイム)処理

現状ではネットワークを介したループは無理。
時間にクリティカルな処理は必要なRTCを
同じ計算機に乗せる。



RTミドルウェアは魔法ではない

テクニック(初期化処理)



- onInitialize:
コンポーネントを使うのに必須となる資源の確保(チェック)など、後から再度初期化することの必要のない処理
- onActivated:
コンポーネントを立ち上げてから、パラメタを変更して初期化する必要がある処理

ひとことで初期化と言っても...

テクニック（接続タイプ）



Flash（同期送信に注意）

同期送信ではデータが受信側まで届いて送信が完了するまで書き込み側が待たされる。送信側で時間にクリティカルな処理は駄目。



New

（非同期送信）
を選択する

RTミドルウェアは魔法ではない

接続タイプ

処理に合わせたデータフロー選択

- **Flush**（同期送信）

書き込みが終了するまで呼び出したコンポーネントの実行が待たされる。ネットワークを介した接続に使うと性能低下。

- **New**（非同期送信）

書き込みは異なるスレッドで実施。push ポリシー（送信バッファ制御）をサポート

- **Periodic**（周期送信）

事前に設定された周期の別スレッドとして送信を実行。pushポリシー（送信バッファ制御）をサポート。

テクニック(通信路指定)



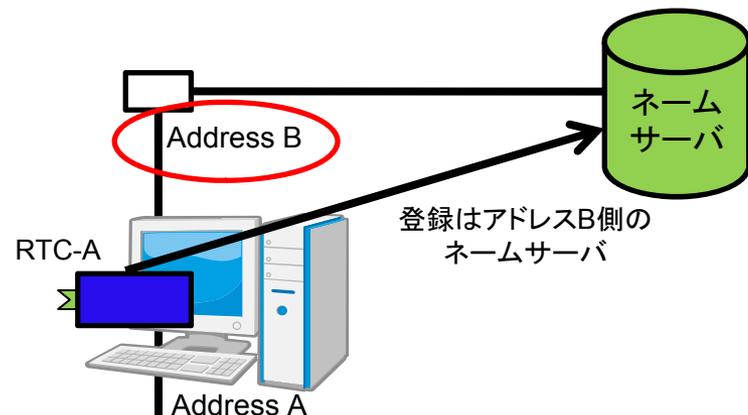
通信経路が複数ある場合

複数のネットワークインタフェースを持つ場合(Ethernetの他に無線やVirtual Machineの通信経路がアクティブな場合)、ネームサーバに間違っただ情報が登録される場合がある。

rtc.conf

```
corba.endpoints: 192.168.0.12:  
(最後のコロンを忘れずに!)
```

- 正しい経路のネットワークインタフェース情報を設定する



通信路を確認

これからのロボット開発

ソフトウェアのモジュール化が現実になることで...

- 既存のモジュールを組み合わせて設計
- 既存のシステム設計をテンプレートとして活用
- システムのカスタマイズが容易
- 開発したシステムのメンテナンス性も高まる
- ロボットを作ることよりも、ロボット技術を利用したサービス開発に研究開発資源を集中
- 研究成果は論文だけでなく、モジュール化して提供する。(技術移転も容易であり、技術の比較検討も容易)

これからのロボット開発

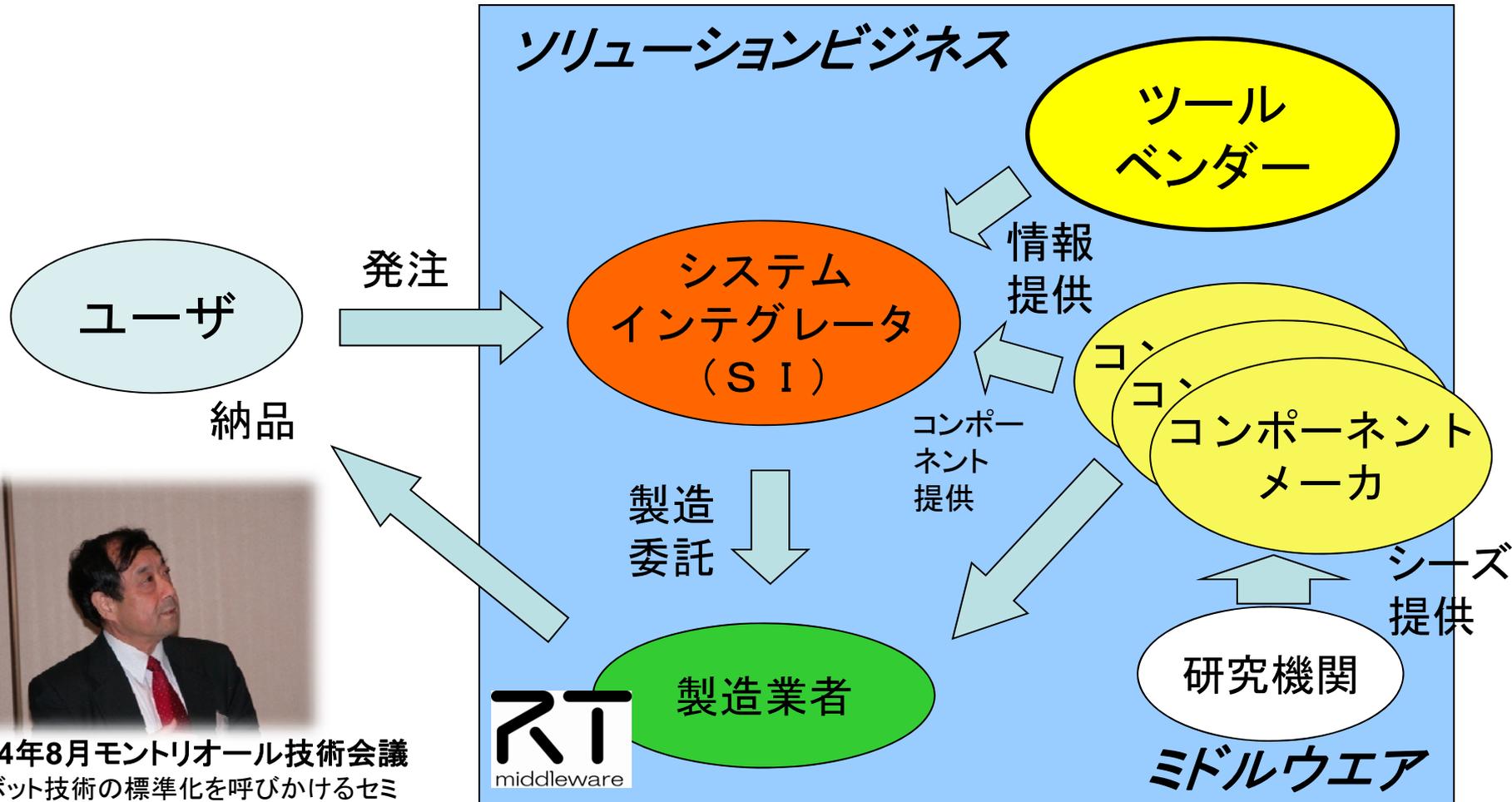
ソフトウェアのモジュール化が現実になることで...

ロボットを作ることよりも、**ロボット技術を利用した異分野融合によるサービス開発**に研究開発資源を集中

- 生活支援・介護のシステム化
- 農林水産分野のシステム化
- 実験系研究のシステム化
- 医療分野のシステム化
- 交通・物流分野のシステム化
- セキュリティ・防衛分野のシステム化 など

社会の中の諸課題をRT技術を導入して
システム化して効率を高めることで解決を目指す

期待される未来のロボット産業のモデル



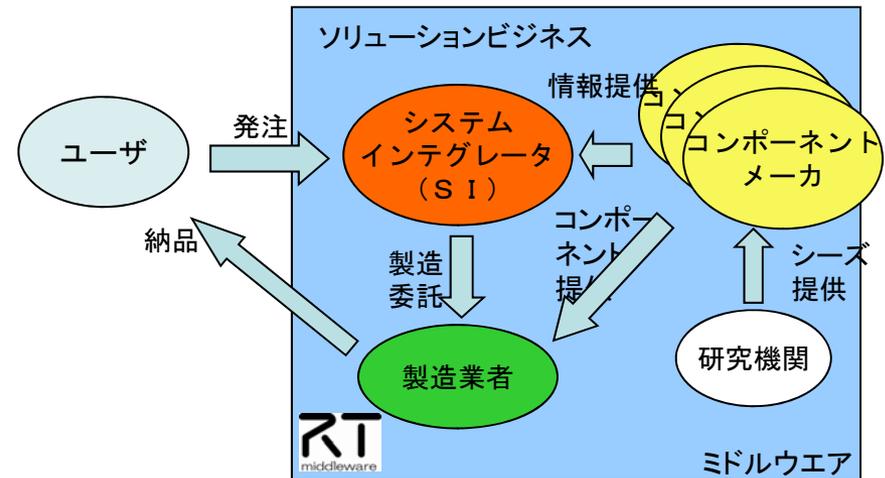
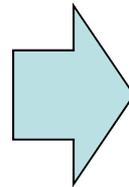
2004年8月モントリオール技術会議
 (ロボット技術の標準化を呼びかけるセミナーを企画し、基調講演する谷江和雄
 President of IEEE Robotics and Society)

ミドルウェア: SI、コンポーネントメーカー、製造業者間の
 交流を促進する情報基盤

まとめ

ロボット技術の共有と蓄積を目的として、ソフトウェアのモジュール化の推進

RT
middleware



- 多品種少量生産の製品の開発コストを下げ、市場を創成。(技術の再利用、相互接続性)
- 論文で終わっていた、ロボット技術を集積して、高度化を促進する。(技術の共有)

コンポーネント開発例

- ホームページのプロジェクトページ (産総研)
(近日公開予定) <http://www.openrtm.org/>
- [RT ミドルウェアコンテスト](http://www.openrtm.org/rt/rtmcontest.html) (産総研)
<http://www.openrtm.org/rt/rtmcontest.html>
- [NEDO 知能化モジュール集2009](http://www.nedo.go.jp/kankobutsu/pamphlets/02kikai/chinou.pdf) (NEDO)
<http://www.nedo.go.jp/kankobutsu/pamphlets/02kikai/chinou.pdf>
- [RT ミドルウェア技術カタログ2010](http://www.openrtm.org/rt/RTMcatalog2010_v2.pdf) (産総研)
http://www.openrtm.org/rt/RTMcatalog2010_v2.pdf
- [RT ミドルウェア技術カタログ2009](http://www.openrtm.org/rt/RTMcatalog2009.pdf) (産総研)
<http://www.openrtm.org/rt/RTMcatalog2009.pdf>

RTミドルウェアによる開発解説

OpenRTM YouTube チャンネル

<http://www.youtube.com/user/OpenRTM>
(rtshell入門など)

東京大学のRTM-ROS総合運用プロジェクト

<http://code.google.com/p/rtm-ros-robotics/>

– ChoreonoidとOpenHRIについて(原功)

<http://rtm-ros-robotics.googlecode.com/svn/wiki/hara110706.pdf>

– ヒューマノイドロボットHRP-4におけるOpenRTM-aist
応用事例(金広文男)

<http://rtm-ros-robotics.googlecode.com/svn/wiki/20110713Kanehiro.pdf>