

# Jeeves フレームワーク：ロボットサービスのための分散処理フレームワーク

五十嵐登<sup>†</sup> 中川幸子<sup>†</sup> 大山直人<sup>†</sup> 清水将吾<sup>†</sup> 成田雅彦<sup>†</sup> 加藤由花<sup>†</sup>

産業技術大学院大学 産業技術研究科<sup>†</sup>

## 1. はじめに

近年，複数のロボットや各種センサ，デバイス等を連携したサービスを提供したいという要求が高まっており，様々なサービス基盤が提案されている[1]．このような背景の下，ロボットサービスのインターネット化を目指して，RSNP (Robot Service Network Protocol) が提案されている[2]．RSNP はロボットサービスをインターネット経由で利用・共有するためのロボットサービスプロトコルであり，RSi (Robot Service Initiative) [3]により仕様化されている．RSNP のシステム構成を図1に示す．



図1 RSNP のシステム構成

RSNP は Web サービスを基盤としていることから，インターネット分野の開発者を取り込み，ロボットサービスの普及に寄与してきた．しかし，RSNP はロボットとロボットサービス間の通信プロトコルを定義したものであり，ユーザの要求に対し複数のロボット間で処理を分散させるようなサービスの場合，スクラッチからアプリケーションを開発する必要があった．本稿では，サービス要求を行うユーザと，実際に処理を行うロボットとの連携を疎結合とすることにより，複数ロボット間の処理分散を容易に実現するフレームワークを提案する．

## 2. フレームワークの構成

提案するフレームワークは，ユーザからの要求に応じ，複数のロボット間で処理分散が可能なアーキテクチャとして Publisher/Subscriber モデルを採用することとした．ロボットとサービス間の通信には RSNP を使用し，分散の内部処理にはメッセージキューイング方式を利用する．フレームワークの名前は，英国の小説に登場する有能な執事の名前にちなみ，Jeeves フレームワークと名付けた．

Jeeves フレームワークの設計にあたっては以下の3つの設計指針を策定した．

- ユーザからのサービス要求は，ロボットの状態，実装を意識せずに行えること．

これにより，サービス要求とロボット処理の実装を分離する．

- ユーザからのサービス要求は，複数ロボットで同時に処理が可能であること．

これにより，サービス要求に対し，複数台のロボットまたは複数のロボット機能を同時に利用可能とする．例えば，複数台の Web カメラでの監視や，Web カメラとロボットアームの操作を両立連携させる処理等を実現する．

- ユーザからのサービス要求は，負荷分散が可能であること．

これにより，同一の機能を持つ複数台のロボットで処理の負荷分散を可能とする．

これらの設計指針を満たすために，4つの機能から成るアーキテクチャを設計した．

### 1) ユーザ要求受付機能

ユーザのサービス要求を受け付ける機能

### 2) ロボット処理取得機能

ロボットがロボット処理を取得する機能 (ロボット端末側の機能を含む)

### 3) 処理割当機能

サービス要求とロボット処理のマッチング処理を行い，ユーザとロボットを関連付ける機能

### 4) サービス提供機能

ロボットサービスの実体

### 3. フレームワークの実装

提案したフレームワークの有効性を検証するため、Amazon EC2 上に Jeeves フレームワークを実装した。システムの構成を図 2 に示す。

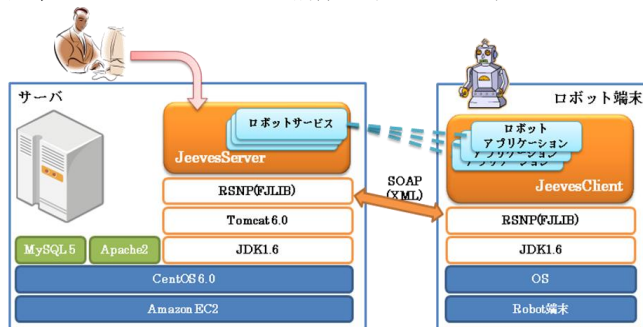


図 2 システムの構成

Jeeves フレームワークは、RSNP の一実装である FJLIB (富士通研で開発された Java ライブラリ) を利用して構築され、分散処理を実現するフレームワークである JeevesServer および JeevesClient の上に、ロボットサービス (サーバ側アプリ) およびロボットアプリケーション (ロボット側アプリ) を実装する形態となっている。サービス要求とロボット処理のマッチングはフレームワークの機能として提供されるため、各サービスの開発者はその実現形態を意識せずにソフトウェアの開発が可能である。

Jeeves フレームワークを利用したロボットサービスの処理フローを図 3 に示す。

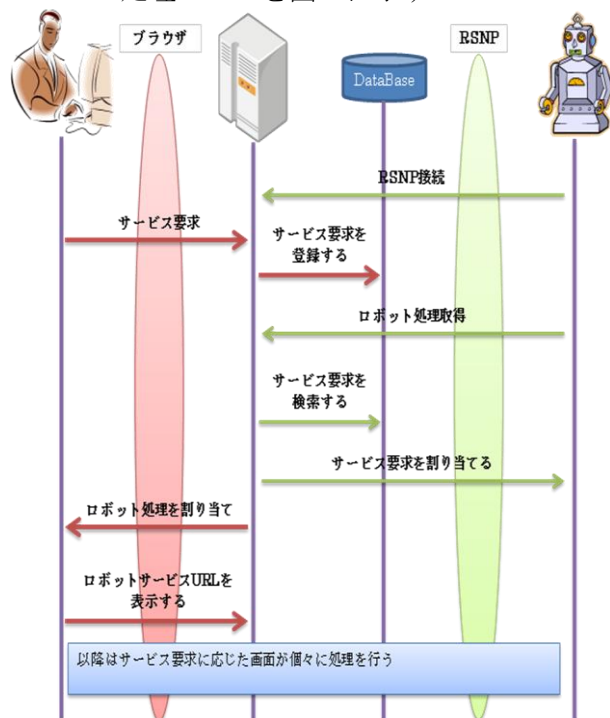


図 3 処理のフロー

サービスの利用者は、ユーザ要求受付機能を利用し、Jeeves フレームワークにサービスを要求する。この状態で、ロボット端末は、ロボット処理取得機能を利用してロボット処理を取得する。このとき、処理割当機能により、条件に合ったサービス要求とロボット処理の割当が行われる。その後、実際のサービスが開始される。

### 4. 考察

Jeeves フレームワーク上に、プロトタイプシステムとして Web カメラによる見守りサービスを構築し、フレームワークの検証を行った。ここでは、インターネット技術者はロボットの状態を意識せずにサービス構築を行い、結果として複数種類のロボット (富士通製サービスロボット enon およびレゴロボ) に共通なロボットサービスの提供を実現した。一方、ロボット技術者はロボット特有の内部処理の開発に注力することができた。

3) の処理割当機能について、現状では先着順 (FIFO) にサービス要求をロボット処理に割り当てているが、何らかの優先制御が必要なサービスが多く存在することがわかった。優先制御の一方式については、文献[4]に示す。

### 5. まとめ

本稿では、ロボットサービスのための分散処理フレームワークを提案し、実装を行った結果を報告した。今後、ロボット対ロボットの連携や、サービス要求を分割する仕組みの検討などを行なっていく予定である。

### 参考文献

- [1]. KuffnerJames. Cloud-Enabled Humanoid Robots : Humanoids2010 Workshop “What's Next” , 2010 年.
- [2]. 成田, 村川, 植木, 中本, 日浦, 平野, 蔵田, 加藤. 普及期のロボットサービス基盤を目指す RSNP (Robot Service Network Protocol) 2.0 の開発 : 日本ロボット学会誌 Vol.27 No.8 pp.857~867, 2009.
- [3]. RSi - Robot Service initiative. 2011. <http://robotsservices.org/>.
- [4]. 角田 龍太, 齊藤 由香利, 阪口 和明, 中山 央士, 清水 将吾, 成田 雅彦, 加藤 由花, "Jeeves フレームワークにおけるタスク割当方式の提案", 情報処理学会 第 74 回全国大会, 2012(発表予定)