



第1章 自己位置推定&地図作成, 動作計画ツール, 画像処理ライブラリ, コミュニティ…なんでもある

世界中の研究者やエンジニアと情報共有! ロボット開発ソフトウェア ROS

山本 要介 Yosuke Yamamoto

ロボットのソフトウェアの世界では、ROS(Robot Operating System)がデファクト・スタンダードとなりつつあります。こういったものなのかを簡単に紹介します。

世界中でロボット開発に使われているROS

■ オープンソースで開発されているソフトウェア

● **世界中のROS開発者コミュニティの助けがある**
ROSはRobot Operating System(ロボット・オペレーティング・システム)の頭文字で日本語では一般的に「ロス」と発音されます。ロボットを扱うためのオープンソース・ソフトウェアです。

ROSとは何か?について、図1に示すように「狭義のROS(Small ROS)」と「広義のROS(Large ROS)」を分けて説明します。

「狭義のROS」は「通信ライブラリ」と言えます。ROSの源流となるソフトウェアは、2007年までさかのぼります。2012年からは非営利の財団OpenSource Robotics Foundation(OSRF)が設立されて、開発やサポートが行われています。

「広義のROS」は、狭義のROSである通信ライブラリに加えて、**ROSで使えるツールやライブラリ群、それらを開発・サポートするROSのコミュニティを含めたエコ・システム全体**です。

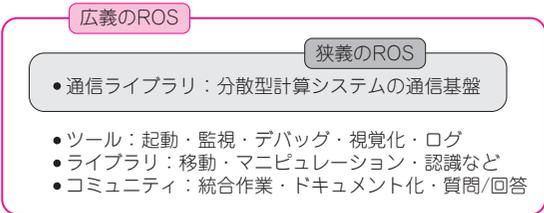


図1 ROSという言葉が示す範囲
ROSが他のロボット開発用ミドルウェアと大きく異なるのは、多彩なツールやライブラリがあることに加えて、コミュニティでの活動が活発なところ

ROSのツール群は、ソフトウェアの起動、監視、デバッグ、視覚化、ログ取得の機能などがあり、大変便利です。これらのツール上で機能するライブラリ群は、認識、移動、マニピュレーション(腕ロボットなどの操作)などが提供されています。これらのツールとライブラリの数々は、世界に広がるオープンソース・コミュニティにて、活発に情報交換や開発が行われています。

● 開発に良い循環が生まれている

ROSが開発される発端は、ロボットの研究では、センサなどから情報を取得するドライバ、情報の視覚化ツールなどの開発に多くの時間が費やされてしまい、研究すべき内容に割ける時間が少ない、という問題意識でした。ROSはこの問題を改善し、**ロボットの研究そのものに時間をかけられるようにすることで、ロボット技術を発展させることを目的**としています。

ただ、ロボット用分散通信ライブラリはROSの他にもあり、開発や研究の手間を減らすという観点、つまり通信ライブラリとしての「狭義のROS」は、特に目新しくありません。

ROSが広まっている背景は、ROSがオープンソース・ソフトウェアであるとともに、「広義のROS」であるツール、ライブラリ、コミュニティを大切にすることで、多くのユーザを共同開発者として取り込み、リリース→テスト→改良→…の良い循環を生み出している点でしょう。

ROSのアーキテクチャ

● 分散アーキテクチャを実現するノード、コア、通信の組み合わせ

ROS上で動くセンサ・ドライバ、認識、移動、マニピュレーションなどの**分散型プロセスは「ノード」と呼ばれます**。ノードは、C++、Python、Java、LISP、Rubyなどさまざまなプログラミング言語で作成可能です。

ROSのノードの動きをまとめるのが「ROSコア」です。ROSコアは「ROSマスタ」、「パラメータ・サ