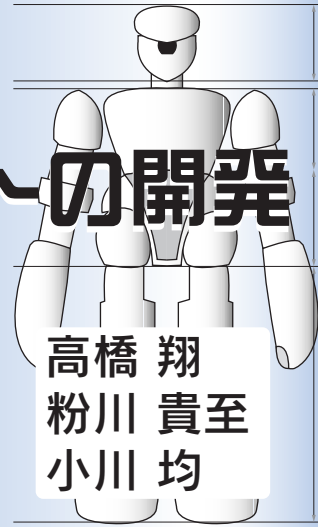


# ネット経由でモーション変更可能な二足歩行ロボットの開発



高橋 翔  
粉川 貴至  
小川 均

## 第3回 Javaでロボット・アプリケーションを作る

今回は、二足歩行ロボット「SPC-101」や「SPC-101C」の制御アプリケーション・ソフトウェアをJava言語で開発する際に利用できるJavaパッケージ「ritumei.airlab.spc101」について解説する。このパッケージとサンプル・プログラムは、Webサイト([http://www.airlab.ics.ritsumeai.ac.jp/robot\\_platform/](http://www.airlab.ics.ritsumeai.ac.jp/robot_platform/))で公開されており、誰でも利用できる。(編集部)

### 1. SPC-101 用 Java パッケージ開発の背景

筆者らは、ロボットを情報システムや情報家電と連携させ、人とシステムのインタラクションをより円滑にするための知的コミュニケーションを行う研究に取り組んでいます。一昨年末、縁あってスピーシーズのネットワーク・ロボット「SPC-101」に出会いました。

SPC-101は、ロボット・アプリケーション開発用API「OpenRoads」を利用することで、ロボットの内部システムを意識することなくロボット・アプリケーション・ソフトウェアを開発できます。筆者らは、OpenRoadsを利用してJavaアプリケーションを開発するためのJavaパッケージ「ritumei.airlab.spc101」を作成しました。ここでは、このパッケージについて、パッケージ内部の仕組みやソース・コード、サンプル・プログラムなどを紹介します。

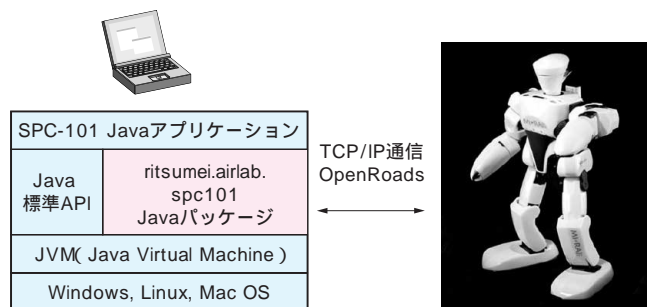


図1 ロボット・アプリケーション開発の構成

SPC-101は、ロボット・アプリケーション開発用のAPI「OpenRoads」を備えている。筆者らはOpenRoadsを利用したJavaパッケージを開発した。これを使うことにより、パソコン上で、SPC-101を制御するJavaアプリケーション・プログラムをより簡単に開発できる。SPC-101はNetBSD配下のTCP/IPサーバとして動作しており、ポート番号5001番でソケットを開放している。パソコンからの制御はこのソケットを使用し、1対1で通信する。

### ネットワークを介して制御する利点

筆者らは、ロボットを以下のような位置付けで利用しています。

- 1)人とのインターフェースとして利用する
- 2)ほかのシステムと連携させる
- 3)複雑な(高度な)処理はパソコンで行う

一般的に言って、ロボットの制御ソフトウェアを開発するためには、ロボットの専門知識や組み込みアプリケーション・ソフトウェア開発の知識が必要です。しかし、ネットワーク越しに制御できるAPIを備えたロボットであれば、ロボット内のシステムを意識することなく容易にロボット・アプリケーション・ソフトウェアを開発できます。

筆者らは、インターネット上にあるさまざまな資源や、パソコン上で動く高度なアプリケーション・ソフトウェアを、ロボットが提供するサービスとして利用しようとしています。その際に、ロボット用として専用アプリケーション・ソフトウェアを開発するよりも、パソコン上で開発したアプリケーション・ソフトウェアのインターフェース部分をロボットと結合できれば、汎用性と開発容易性がより高まると考えています。

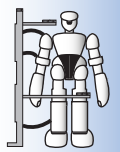
### Javaで開発する利点

開発言語としてJavaを選んだ主な理由は次の通りです。

- 1)開発環境が簡単に無料で整えられる<sup>注1</sup>
- 2)ネットワークを中心にライブラリが充実している
- 3)プログラミング初級者でも開発できる

Javaの開発環境はハードウェアやOSに依存しません。パソコン(WindowsでもLinuxでも)上でJVM(Java Virtual Machine)を動作させ、Javaの標準APIや筆者ら

注1: Javaの開発環境JDK(Java Development Kit)は、Sun Microsystems社のWebサイトから無料でダウンロードできる。



が開発した Java パッケージを利用することにより、パソコン主体のロボット・アプリケーション・ソフトウェア開発を容易に行えます(図1)。

## 2. SPC-101 のネットワーク・インターフェース



ここで、パソコンと SPC-101 の間のネットワーク・インターフェースについて解説します。なお、パケットの詳細構成については、スピーシーズの Web サイト(<http://www.speechys.com/openroads.html>)で公開されている OpenRoads の仕様書<sup>1)</sup>を参照してください。

### 接続

SPC-101 は、NetBSD 上で割り当てられた IP アドレスと TCP ポート番号 5001 番でソケットを開放しています。パソコンから制御する際はこのソケットを使用して、1対1で通信します。

### パケットの種類と構成

パケットの種類として、データを送信するパケットや実行を指示するパケット、サーボの個別制御を指示するパケット、サーボやロボットの情報を要求するパケットなどがあります。これらのパケットを組み合わせると、SPC-101 に内蔵されている MiniSD カード内に保存されているモーションを実行したり、パソコンからモーション・ファイルを送信して実行したりします。

例として、パソコンからモーション・ファイルを送信してモーションを実行する際のパケットの流れを図2に示します。パソコン側からファイル・データを送る際、一度にすべてのデータを送ることはできないので、最大 256 バイトずつに分割して送信します。SPC-101 側ではモーションの一時ファイルを作成していきま。また、ファイル・

データを含むパケットをパソコンから受信すると、ACK パケットを返します。これを繰り返し、すべてのデータを送信し終わったら、SPC-101 側で完成した一時ファイルと呼び出す実行パケットをパソコン側から送信します。そうすると、SPC-101 側では一時ファイルが呼び出され、モーションが実行されます。

モーション・ファイルの中には、各サーボの角度情報や LED 表示、再生するサウンド・ファイル名などの情報が時間軸に沿って記述されています。

## 3. 開発した Java パッケージの概要



筆者らが開発した Java パッケージ `ritsumei.airlab.spc101` は、モーションやポーズの実行、個々のサーボの制御、LED 表示、音声再生などといった SPC-101 の標準的な制御と、サーボのステータス情報を取り扱う機能を備えています(表1)。この Java パッケージを利用すれば、パケットの詳細を意識することなく、行いたい操作をプログラムに記述していくだけで済みます。

`ritsumei.airlab.spc101` パッケージは、`SPC101` クラス、`SPC101Control` クラス、`ServoStatusManage` クラスという三つのクラスを含んでいます。標準的なロボット制御を扱うクラスが `SPC101Control` クラスになります。`SPC101` クラスは、サーボ名や可動範囲など、SPC-101 固有のサーボ情報を扱います。`ServoStatus`

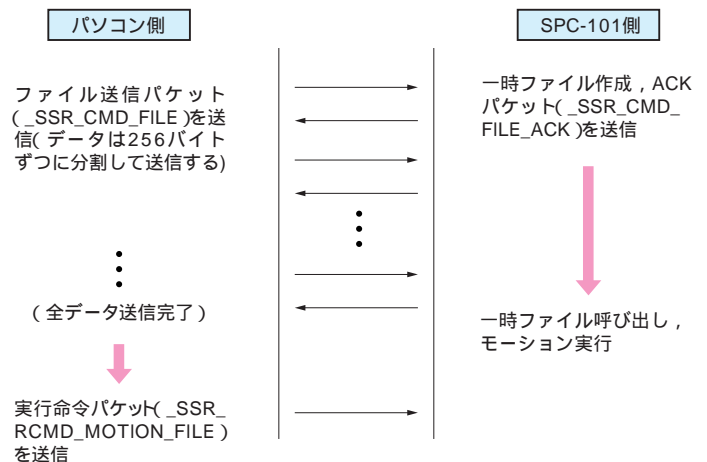


図2 ファイルデータ送信時のパケットの流れ

パソコン側からファイル・データを 256 バイトずつに分割して送信する。SPC-101 側はパケットを受信すると ACK パケットを返す。すべてのファイル・データを転送し終わったら、パソコン側から実行パケットを送信し、SPC-101 側でモーションが実行される。

### コラム モーション・ファイルの種類

モーション・ファイルには、モーション・データ・ファイル(\*.mtn)と転送用モーション・データ・ファイル(\*.mtr)の2種類があり、`actMotion`メソッドで使用するモーション・ファイルは後者の転送用モーション・データ・ファイル(\*.mtr)です。このファイルは、モーション・エディタでは[モーション展開]ボタンを押すことで作成できます。