

組み込み分野での活用が期待される 軽量 Ruby への取り組み

新世代のスクリプト言語として広く普及し始めているRubyの長所を幅広い分野に活かすため、当社は、よりコンパクトな「軽量Ruby」の研究開発に参画し、とりわけ、組み込み機器で活用可能なRubyチップや評価ボードへの取り組みを進めています。当面は産業機器での実績を積みながら、将来的にはエコロジー、エコノミーを実現する家電機器への普及を図っていく考えです。

オブジェクト指向軽量言語を さらにコンパクトに

Rubyは、従来の本格的なオブジェクト言語と比べて手軽で書きやすい次世代のスクリプト言語として、ここ数年世界的に注目を集めている言語の1つです。また、日本人(まつもとゆきひろ氏)により開発されたということもあり、日本語による文献も多く日本人技術者にとって親しみやすいのも特徴の1つです。

Rubyは、従来のスクリプト言語が用いられてきた領域でのオブジェクト指向プログラミングを実現するものですが、同じくコンパイル処理を必要としないPerlやPythonなどと比較しても、少ない行数でさまざまな機能を実現でき、可読性が高く、プログラマがストレスなく、より容易にプログラミングを行うことができるのが最大の特徴です(表-1)。

表-1 cat コマンド相当の機能をスクリプトで実現した場合

Perl で記述した場合	Python で記述した場合	Ruby で記述した場合
<pre>use File::Basename; \$name = basename(\$0); sub catn { my \$file = \$_[0]; foreach \$line (<\$file>) { ++\$lineNo; print "%6s %s", \$lineNo, \$line; } } \$lineNo = 0; if (\$#ARGV < 0) { &catn(STDIN); } else { foreach \$filePath (@ARGV) { open(IN, "<\$filePath"); or die "\$name: \$filePath: \$!"; &catn(IN); close(IN); } }</pre>	<pre>import sys, os def catn(file): global lineNo for line in file: lineNo += 1 print("%6s" % lineNo, line, end=" ") lineNo = 0 if (len(sys.argv) <= 1): catn(sys.stdin) else: for filePath in sys.argv(1:): try: file = open(filePath) catn(file) except IOError: print("%6s (%s)" % (lineNo, filePath), sys.exc_info()[1], file=sys.stdout) exit(1) else: file.close()</pre>	<pre>def catn(file) if file.each_line { line \$lineNo += 1 print "%6s %s", \$lineNo, line } end \$lineNo = 0 if ARGV.length < 1 then catn STDIN else begin ARGV.each { filePath file = File.open filePath catn file file.close } rescue STDERR.puts "#File:basename \$0: #[\$!]" end end</pre>
↓		
<p>(さらに言語が持つ標準のエラー処理機能を利用すれば...)</p> <pre>lineNo = 0 ARGV.each_line { line lineNo += 1 print "%6s %s", lineNo, line }</pre>		

クラス定義、ガベージコレクション、マルチスレッド、例外処理、イテレータ・クロージャ、演算子オーバーロードなどの機能を持つほか、グルー言語としての使い方が可能で、Cプログラムやライブラリを呼び出す拡張モジュールを組み込むことができます。テキスト処理関係の能力に優れ、また移植性も高く、当初の個人の趣味レベルから急速に普及が進み、大手企業や商用サイトでも相次いで採用される動きが見られます。

オブジェクト指向軽量(Lightweight)言語と呼ばれるジャンルの1つであるRubyは、これまでWebシステムに適した言語体系として普及してきましたが、ここにきて、機能を絞り込んで、より軽量化した専用言語とすることで応用用途の拡大を図る「軽量Ruby」の取り組みが進められています。

Rubyがコンパクトになることで、サーバ分散環境での利用、リアルタイムOS上で動作するタスクとしての利用、ゲームなどのアプリケーション組み込み言語(スクリプト)への利用など、さまざまな用途が期待されますが、中でも、画面やハードディスクドライブを備えていない、あるいはCPUリソースの少ない機器にも応用可能な組み込み分野に特化したものを「組み込みRuby」と呼んでいます。

組み込み機器の生産性向上を実現する 「組み込みRuby」

現在、組み込みソフトウェア分野では、開発規模の大型化やソフトウェアエンジニアの不足により効率的な開発が求められています。Rubyは、組み込みソフトウェア開発に特化した言語として開発されることで、この分野では主流となるC言語に比べ飛躍的に生産性が向上すると期待されています。

組み込みRubyの取り組みは、既存の組み込み環境に使われているリアルタイムOS上にRubyを搭載すること、もう1つは、直接Rubyを走らせるRubyチップ、の2つのアプローチで進められています。リアルタイムOS版のRubyはVxWorksを使ってロボット

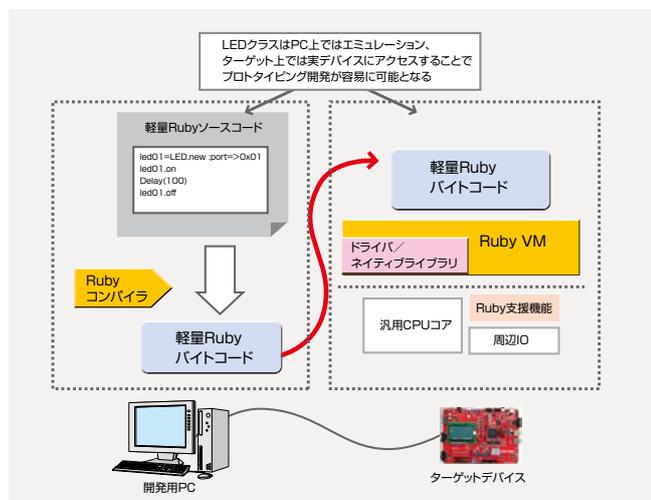


図-1 組み込みRubyの開発環境・動作環境イメージ

を制御する言語に活用しようとしています。

当社が手掛けているのはRubyチップの試作開発で、Rubyで開発したソフトウェアを、家電機器をはじめとする組み込み製品で動作させるための基盤となるものです。



図-2 組み込みRubyチップ評価ボード

従来のRubyはスクリプト言語であるため、インタープリターによってソースコードそのままの状態でも走りますが、組み込みRubyの場合はそれとは異なる動作環境になります。まず開発用のPC上でソースコードを記述し、次にRubyコンパイラによって中間言語(Javaでいうバイトコード)に変換します。そして、生成された中間言語を、ターゲットの機器で解釈しながら実行するRuby VM(バーチャルマシン)の上で動作させます(図-1)。これにより、従来のRubyと比較してメモリ消費量が抑えられ、さらにRuby VMを支援するハードウェアロジックを搭載した専用チップにより高速に動作させることができます。このRuby VMを支援するための専用ロジックを搭載したチップをRubyチップと呼んでいます。

今回、軽量Rubyを動作させる、家電などへの搭載を想定した「組み込みRubyチップ評価ボード」の開発を行いました(図-2)。Rubyチップそのものは、現段階では最適な支援機能を実験するために書き換え可能なロジックLSIであるFPGA(Field-Programmable Gate Array)で実装しました。現在、Ruby支援機能のロジック実装を進めています。

産業用途からスマートホームまでの 応用に期待

エコロジー、エコノミーの時代を迎え、家電機器においても電

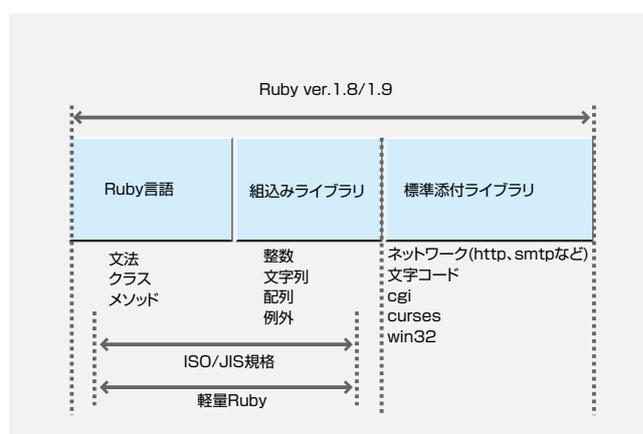


図-3 軽量Rubyの仕様範囲

力制御や管理などのインテリジェンスが要求され、家電製品単体ではなく機器同士の連携によってシステム全体で最適化しようとするのが考えられます。こうした機器間のネットワークにはテキスト処理が多く用いられるため、新しいプロトコル対応などRubyの記述のしやすさと柔軟さがより活かされます。

しかし、この分野に実績のないRubyがすぐに採用されるのは難しく、あくまで最終ゴールを家電機器におきながら、現在は農場の土壌の温湿度、成分などのセンサーデータを管理端末へ送信するセンサーゲートウェイ、工場のラインなどで製造機器からの状態情報を集中管理ホストへ送信する中継器、といったマシン・ツー・マシンのネットワークが要求される産業用途で実績を積んでいく考えです。

現在、Rubyは国際標準化(ISO規格)の動きがあり、2010年9月に最終ドラフトが完成しました。軽量Rubyはこの国際標準仕様に準拠しています(図-3)。

組み込みRubyについては、福岡県の「軽量Rubyを用いた組み込みプラットフォームの研究・開発」への取り組みが2010年度の経済産業省地域イノベーション創出研究開発事業に採択され、(株)福岡CSKを幹事会社として、開発者のまつもと氏や九州工大の先生などが研究員として参加、当社はアドバイザー兼チップ・ボード製造担当としてRubyチップの開発に取り組んでいます。

現在はRubyのチップや評価ボードの受託を進める当社ですが、今後はロジック部分のIP販売、さらにはこの先、用途が確立してくれば小さなPCのようなRubyボックスの開発にも取り組んでいきたいと考えています。

(新規事業開発室 高野秀隆)