



## Java vs C#

### —競争原理はプログラミング言語にも働くか?—

前田 敦司 / 筑波大学電子・情報工学系

委員会が作ったプログラミング言語には魅力がないなどと言われる。1人またはごく少数の人間が、その思想や嗜好を色濃く反映させた言語は当然に個性的で、多くの場合エレガントで、しばしば使いやすい。では企業の開発チームが設計した言語はどのようなのだろうか？

ここでは、JavaとC#という2つの言語について考えてみたい。これらは、それぞれSunとMicrosoftという企業がその経営戦略上の重要な武器として位置づけている言語である。最初に設計したのは少数の個人であったのだろうが、現在の形になるまでに、企業戦略を強く反映させたものになっていると思われる。

これらは「魅力的な言語」だろうか？ プログラミング言語のどのような点が人々を魅了するのだろうか？ 広く使われ、永く生き残る言語の条件とはどのようなものだろうか？

#### 言語の人気

たいていの実用品と同様、プログラミング言語にも生態系における地位に似たものがあり、それをめぐった争いが常に続いている。生物界の生存競争や、他の工業製品の競争と違っているのは、その「慣性」の大きさである。プログラム言語がこの先生き残るかどうかは、その言語が過去にどれだけ使われたかによるところが大きい。

形ある工業製品が、やがては損耗して使えなくなってしまうのに対して、プログラミング言語を使って書いたコードは永久に残る。それを捨て去って新たな言語に切り換えるのはコストがかかるから、永く使われた言語というものは容易に消えない。プログラミング言語の生態系では、先住者が常に有利な立場にある。

教育の現場では、レガシーコードの問題がそれほど深刻ではないため、プログラミング言語の流行りすたりはソフトウェア開発現場よりも速い(はずなのだが...)。研究の分野でも、論文でアルゴリズムを説明するような場合には慣性の問題など皆無に等しいので、その時々研究者が「主流」あるいは「カッコいい」と考える言語が使われる。そこでCOBOLやBASICが使われることは決してない(失礼)。かつてはAloglやPascalが多かった。その後、Adaが使われた時期もあったが、現在はCやC++が使われることが多い。何年かたてば流行は次の言語に移り、今まで主流だった言語はもはや古臭いものとみなされるようになる。そのうちJavaだって広く使われるようになるかもしれない。

#### Java?

そう、Java。CやC++にとって代わったとまではいえないし、Javaでアルゴリズムを説明する論文はまだ少ないが、少なくとも研究対象としてならJavaはすでにニッチを確保

したといつてよい。情報処理学会のWebページで検索してみると、Javaをキーワードに含む論文や記事が400件以上ヒットする。ACMのdigital libraryで検索すると、700件近く見つかる。

1995年にJavaが発表された当座<sup>1)</sup>、Javaの処理系は実行速度が遅く、またクラスライブラリもごく非力なものだったにもかかわらず、多くの人がJavaに関心を持った。実際のアプリケーション開発に使われるより早く飛び付いたのは、学生や研究者たちであり、中でも早くから興味を持ったのはプログラミング言語の研究者たちだった。

研究者にとってみれば、格好の飯の種を見つけたというところかもしれない。なにせ研究者だってお金と無縁ではいられない。研究費をとってくるには産業界が関心を持つ言語の方が簡単だし、解説書を書くにしてもポピュラーな言語の方がよい。研究するにしてもメジャーな言語の方がインパクトがあるし、大学で教えるにしても、ほかではお目にかかれなようなマイナーな言語よりは、どこでも使える言語の方が望ましいに決まっている。

しかし、研究者たちが飛び付いたのは、単に「広まりそうだから」という理由だけではないと思うのだ。そもそも「広まりそうだ」と思うにはそれなりの理由があったはずだ。また、メジャーだというだけでは論文にしたり学生に教えたりする気にはならないだろう。VisualBasicの処理系に関する論文など見たことがないし、情報系の学生にVBでプログラミングを教えている大学は聞いたことがない(きっとあるのだろうけど...)

まず言語自体がそこそこモダンな仕様であったのが重要だと思う。たとえばオブジェクト指向というパラダイムが役に立つということは、専門家にとってはとくに常識になっていたのだが、ポピュラーで・いろんな環境で使えて・安価な(できればフリーな)処理系が手に入る言語で



は、なぜか1970年代そのままの「こーぞーかってすごいでしょ」と言ってるレベルのものしか存在しなかった。新しい概念を取り込んでいて、落とし穴が少なく、手近な環境でも動く言語がやっと出てきたか、という喜びが大きかったのではなからうか。

手近な環境とは、WindowsだったりMacだったりするのかもしれないが、私のまわりでは一般にUnix系のOSが好まれているようだ。内部構造がよく見えて、しかもシステムの働きに手を加えることが容易であることがその理由だろう。自動車の構造を教えたり、「新たなエンジンのチューニング法」を研究したりするのに、ボンネットを空けることのできない自動車を使ったのでは都合が悪いのと同じことだ。

中身が見えない・触れない環境の方が世の中では多くなってきて、性能も使い心地もどんどん向上しているのに対して、自分たちの環境でプログラミングの教育に使える言語が相変わらずCか、せいぜいC++くらいにとどまっていることにいらだちを感じていた人は多いだろう。そのうち世の中は中身が見えない、コンピュータの仕組みを教えたり研究したりするのにきわめて不都合な環境だけになってしまっただけで、その他の環境は滅びてしまうのではないかという不安だってあったはずだ。そういう環境をふたたびモダンなものにしてくれるという意味でも、Javaはありがたい選択肢だったのではないか。

## C#

一方、「中身が見えない環境」で世の中を覆いつくそうとしていた側にとっては、そうでない環境の有用性が増すのは面白いことではないから、当然いろいろな対抗手段をとってくる。自製品のシェアを高めようとするのは商行為としては当然であるから、別にそれ自体を非難する理由はないのかもしれない。

いろいろあった対抗手段のうちの1つが、昨年発表されたC#言語である。プログラミング言語オタクとしては仕様のディテールをあげつらって、どちらが良いかのディベートをやってみたり、後知恵の優位な立場から設計者たちをけなしてみたりするのも楽しいと思うのだが、大多数の読者にとって、細かな仕様上の差異などに興味はないだろう。

ここでは、両者の言語仕様のコア部分はほとんどそっくりだとだけ言っておく（仕様書<sup>3)</sup>にはJavaへの言及は皆無だが）。個人的な印象としては、Javaがどちらかといえばストイックで、多少書くのが面倒でも言語の単純さを保とうとする傾向が強いのに対して、C#の方は、少しくらい不統一でも便利なものはどんどん採り入れようとする傾向があるように感じる。その程度の違いである。

さて、自由経済の世の中では、同じニッチを奪い合う企業間の競争によって技術が進歩し、製品の価格性能比が向

上し、全体の生産性が向上するということになっている。この2つの言語も健全な競争によって使う側が利益を得るということになるのだろうか？

徹底してOS非依存を狙ったJavaに対して、C#は特定のプラットフォームの機能をなるべく引き出すことに主眼を置いている。しかし、C#の言語自体はプラットフォームに依存したものではなく、ECMAに言語仕様を提出して規格化するのだという<sup>2)</sup>。また、C#が前提とする.NET Frameworkも、ネットワーク透過で、OSに依存しないプラットフォームになるのだという。つまり、Javaが目指している世界とほとんど変わるところはないように思える。やはりプログラマはJavaとC#をどちらでも自由に選べて、競争の利益を享受できるのだろうか？

## 不安

そうはいかない気が強くするのはなぜだろう？ もし競争を仕掛けたC#が勝ち残って、Javaを使う開発者がほとんどいなくなったとすると、開発者のコミュニティは競争がなかったときに比べて利益を得たことになるのだろうか？ それよりも、コミュニティがそもそも分裂せず、負けた言語から乗り換える苦勞などしないですむ方が利益になると思うのは、競争原理に反する「抵抗勢力」の考え方か？

もしJavaが開発現場から消えたと仮定してみても、.NET FrameworkとC#を使った「オープンな環境」でプログラミングを教える自分が想像できないのは単なる偏見、あるいは想像力不足だろうか？ 「中身が見えるかどうか」などにこだわらず、勝ち残って世の中で広く使われているものを教えるべきなのだろうか？

C#が勝ってJavaが消えると世界が闇に閉ざされるように感じるのは、私が滅びつつある環境にしがみついているだけのオールドタイプだからか？ もしそうであっても実は教育にも研究にも何の害もなくて、今Javaを対象としている研究者たちは、フリーなC#コンパイラのソースコードを改変して新しい研究を行うようになるのだろうか？

いったいこの不安が根拠のないものなのかどうか。どなたか教えてください。

## 参考文献

- 1) Byous, J.: Java™ Technology: An Early History.
- 2) Wong, W. and Ricciuti, M.: Microsoft Strikes at Sun's Java with New standard.
- 3) Microsoft Corporation: C#標準リファレンス, 日経BPソフトプレス (2001).
- 4) Archer, T.: プログラミングC#, 日経BPソフトプレス (2001).
- 5) 後藤弘茂: .NETの正体はバーチャルWindows構想.

(2001.9.15)