

ナイスステップな研究者から見た変化の新潮流

# Ruby アソシエーション まつもと ゆきひろ 代表理事理事長インタビュー

聞き手：企画課長 三木 清香

科学技術予測センター 研究員 小柴 等、特別研究員 中島 潤

昨今、社会のあらゆる分野で人工知能やIoT（モノのインターネット）など高度な情報技術がもたらす発展の兆しが見られ、第5期科学技術基本計画でも、ICTを最大限に活用し、サイバー空間とフィジカル空間とを融合させた取組により人々に豊かさをもたらす「超スマート社会」の実現に向けた一連の取組を、「Society 5.0」として強力に推進することを掲げている。

科学技術・学術政策研究所（NISTEP）では、ICT分野で活躍し、“Ruby”というプログラミング言語を創作してオープンソースソフトウェアとして発信した世界的なコンピュータ・プログラマー、一般財団法人Rubyアソシエーション代表理事理事長“まつもと ゆきひろ（Matz）”氏を、2010年に「ナイスステップな研究者」に選定した。今回の「ナイスステップな研究者から見た変化の新潮流」では、まつもと氏にインタビューし、活動の軌跡や超スマート社会を迎える中での課題について、お話を伺った。

1. まつもと氏が創り出されたプログラミング言語Rubyについてお伺いします。Rubyは日本発のプログラミング言語としては唯一ISOの標準も取得し、開発者・利用者が世界に広がるメジャーな言語になっています。Rubyとは具体的に、どのようなものでしょうか。

Rubyはプログラミング言語ですので、コンピュータに何かをさせようというときに使います。世の中には数千とも数万とも言われるプログラミング言語がありますが、広く知られる言語は、ごく一部です。その中であってRubyは、その生産性や柔軟性が高く評価され、世界中で使われています。他の言語と比



“まつもと ゆきひろ”氏

較しても、Rubyでソフトウェアを開発すると、簡潔で保守しやすく、また将来の変更にも強いという傾向があります。また、Rubyを気に入っている技術者のコミュニティが存在しているので、困ったときにインターネットを介した助け合いが行いやすいというのもメリットと言えそうです。

具体的なRubyの利用例としては、料理レシピ検索の「クックパッド」やグルメサイトの「食べログ」、ビジネス系では会計サービスの「Free」などが挙げられます。

元々は私が自身の趣味として作成していたRubyですが、おかげさまで広くビジネスでも御活用いただくようになりました。

2. そのような利便性や柔軟性が基礎となって、プログラミング言語の人気度合いを示す指標TIOBE Index<sup>注1</sup>でも上位にランクインしているのですね。趣味で始められたRubyが世界的に普及し、Rubyアソシエーションという法人活動に至った経緯について、御紹介をお願いします。

注1 TIOBE社が月1回の頻度で更新する、プログラミング言語の人気度合いを示す指標（<http://www.tiobe.com/tiobe-index/>）。複数の検索サイトを用い、各言語の検索結果が何件出現するかで算出される。

私自身は、元々は、ソフトウェア会社に勤務するプログラマーでした。当初は、会社の勤務時間外に、趣味でプログラミング言語である Ruby を作っていました。これをオープンソースソフトウェア<sup>注2</sup>としてインターネット上で公開したところ、好評を受けて世界中で使われるようになりました。Ruby の公開は 1995 年で、2005 年頃からは、ベンチャー企業を中心にビジネス用のインターネットサービスを構築するツールとしても徐々に使われるようになりました。

趣味の間は技術的に面白いものを作ることに注力していられますが、利用者が広がり、ビジネスにも使われるようになるにつれ「趣味の人たちばかりだと不安だ」、「作っている人たちが飽きたらどうするんだ」と思われる方も出てきました。そういう「仕事で使う人」と「趣味で作る人」のギャップを埋める方策として 2007 年に Ruby アソシエーションを設立したというのが法人活動に至った経緯です。

オープンソースの流れでソフトウェアの作り方も変化してきています。「オープンソースソフトウェアとそれを利用するビジネスの人たちのギャップ」を埋めることが私ども Ruby アソシエーションの活動の大きなテーマの一つだと思っています。

— 当初から今のように独立した法人の姿を目指されていたのですか？

思えば遠くまで来たと思います。

学生の頃からソフトウェア開発に非常に興味があり、ソフトウェアの開発に携わりたいと考えていました。また、フリーソフトウェアやオープンソースソフトウェアにも強い関心を持っていました。加えて、プログラミング言語を自分で作りたいとも思っていました。結果として、こうした学生の頃からの漠然とした興味や考えを全部組み合わせることができ、現在の活動に至った、というところですよ。

切り拓くという綿密な設計はありませんでしたが、学生のときに他の言語などを見て、いいなと思っていたものをどんどん取り込んでいたら、諦めなかったことや運の良さも重なり、予想外にうまく組み合わせさせて今の姿になっています。

— ナイスステップな研究者に選ばれた 2010 年頃は、趣味に発する技術発展とビジネスのギャップを

つなぐ活動をされていた時期でしょうか。その後は、Ruby をどう育ててこられましたか？

Ruby を使われる企業の方々の不満をできるだけ少なくするとともに、Ruby そのものを技術的に発展させてきました。例えば、インターネットサイトの維持や各種ツールの分析、古いバージョンの補修、セキュリティ問題など、最新でないところは研究者はやりたがりませんが、ビジネス利用には大切です。そこで、Ruby アソシエーション参加企業から頂いた費用で、仕事として技術者に発注して確実に Ruby を維持して信頼性を高める活動をしています。

また、Ruby を使っている技術者が「自分は Ruby に詳しいですよ」と証明できるツールとして、技術者認定試験なども用意しています。さらに、この会社にお願すれば Ruby を使った開発をやってもらえるとか、この教育機関にお願すると Ruby の技術者を育ててもらえるという目安になるように、会社や組織の認定もしています。このような形で企業の人事の方や経営の方々が安心して Ruby を利用できるように、目安を提供するようなこともやっております。

— Ruby の普及に伴う事業拡大では、どのような変遷があったのでしょうか？

やっていることは変わらないのですけれども、使う方が増えるに従い、ミスなどに不安を感じる方も増えてきたので、ミス防止の手順を踏む担当や組織が必要になってきた、というのはありました。

ソフトウェアの場合は、新しいバージョンを出す「リリース」という作業があります。これをミスなく行うのは私自身も余り得意ではなくて、過去には「一つファイルを入れ忘れまして」とか、「最後に直した修正が含まれていませんでした」と慌てて次を出すこともありました。利用者が 100 人ぐらいのときは「ああそうか、しょうがないね」で済みますけれども、Ruby の普及が進んで影響が大きくなってくると「なんか新しいバージョン、ダウンロードしたんだけど、だめなんだけど」みたいに大騒ぎになるのです。事業が拡大するにつれて、そういう手順の部分をおのづかやが助けてくださるようになったというのがあります。

一人で回らなくなったというのも事実です。ソフトウェアを作る場合には、そのソフトウェアがどう動く

注2 オープンソースソフトウェア（英：Open-source software、OSS）とは、ソースコードが利用可能で、かつ、コピー、改変、再配布などの権利についても、著作権者の指定する特定の条件下（例えば同様の条件で公開する、商用利用しない、など）で認める、というようなライセンスに基づいて公開されているソフトウェアを指す。

べきかという仕様を決める面、それから決めた仕様に対して実際にソフトウェアを作る面の両面がありますが、ここ何年かは、私は、仕様を決める、設計をする、デザインをするところに注力してきました。実際にソフトウェアを作る活動は余りしていませんね。

3. Rubyは海外でも広く使われています。日本発でこれほど世界中に広く受け入れられているプログラミング言語は余りないようですが、Rubyの成功要因は、どこにあるのでしょうか。

一番難しいのが、多分、日本と海外の間にある意識の壁ですね。日本は十分に豊かで、日本人と日本語だけで生きていけますから、「日本だけでいいや」という意識は、日本人の誰もが持っていると思います。これが国際展開が進まない要因だと思います。海外では、例えば、母国語でコンピュータサイエンスの教育を受けられる国はごく限られ、「プログラミングに関心があります、職業にします」と言った時点で英語が必須になる国がたくさんあります。母国語だけでは新しい技術は学べない、仕事ができない、というので、ソフトウェアの説明文書も英語で作ることになります。結果的に英語を使う人々がターゲットになり、潜在的顧客層が広がってビジネスで成功する余地が広がると思います。もちろん、広くても失敗するときは失敗するのですけれども。

私の場合は、運が良かったところもありました。私の趣味の一つに、他者が作ったソフトウェアを読んで勉強するというものがありますが、学生のときにダウンロードして調べたソフトウェアにコメントや説明書が私には読めない言語で書かれていたものがあったのです。すごく残念だと思い、自分でソフトウェアを作ったときには、説明書やマニュアルを下手なりに英語で用意するようになりました。それが最初のラッキーで、二番目のラッキーは、そうやって英語で公開したRubyを海外の方が見つけて、これは面白いと本を書いて紹介してくださった。インターネットの検索エンジンなどが動き始めた頃です。その人がたまたまインターネットで見つけて、興味を持って調べてくださり、こんなプログラミング言語があると紹介する本を書いてくださり、その本でRubyを知った方が使ってくださった。そういう意味では、大分運の良い経歴と言えると思います。

Rubyは、日本で広がる方が当然早かったのですが、自分の会社のシステムをRubyで作るようなビジネス利用は、当初は日本には、ほとんどありませんで

した。海外でビジネス利用が増えたのを見て、日本でも採択された経緯があります。ですので、日本で閉じていたら、今のRubyはなかったらと思います。新しい技術は不安なもので、特に日本の大企業は使わないですね。ところが、海外の場合は、うまくいかなければやり直せばいいじゃない、と考える人たちが結構多くいます。そのノリはすごいと思いますけれども。海外の成功を見て日本で採用されたケースもあります。逆輸入ですよ。

Rubyアソシエーションを作った背景には、日本の保守的な傾向の人たちにもアピールできるように、組織基盤を作り、証明になる認定制度を作ることを考えた、というのがあります。日本社会を相手にしなければ、Rubyアソシエーションの活動の半分くらいは要らなかつたらと思いますね。

4. 松江市を拠点にしようとお考えになったきっかけは何でしょうか？

20年ほど前、私の知人の知人であった井上<sup>注3</sup>が島根県で起業するというので、声をかけてもらったことがきっかけです。1997年当時、Linuxやオープンソースソフトウェアを主たる事業とするような会社は東京にも存在しませんでしたから、田舎でそんな仕事ができるなら素晴らしいと思いました。ちょうど名古屋で会社に勤めていた頃で、転職を考えていたタイミングでした。当時、私の所属部署が東京事業所に移転になったのですが、私だけ、転勤拒否をしたので、一人置いていかれて、別のプロジェクトの片隅に机を借りていました。インターネットを使って仕事を進め、週1回は東京に行って打合せをしていましたが、さすがに居心地が悪いなと思っていたところ、島根のお話がありました。それで、知人を通じて話をし、



左から“まつもとゆきひろ”氏、三木、小柴

注3 株式会社ネットワーク応用通信研究所 (NaCl) 代表取締役。まつもと氏は NaCl のフェローでもあり、Ruby アソシエーションは NaCl オープンラボのサテライトオフィス内に所在を置くなど、現在も密接な関係を有する。



勤務先の仕事を一段落させて、1997年3月の会社創立に少し遅れた8月から参加する形で引っ越してきました。彼らが松江に起業しなかったら、私はここに来なかったと思います。

5. ITが加速的に発達しています。今後はどう変わっていくのか、あるいは、どう変えていきたいと考えていらっしゃるのか、御意見をお聞かせください。

未来のことを言うのは難しいのですが、プログラミングに限定すると、プログラミングの在り方みたいなものが、どんどん変わっています。昔は、1台のコンピュータの計算能力も、処理できる容量も低いところに限界がありました。今はCPUという計算する部分を複数載せて、同時に複数の計算をさせるなど、Rubyを作り始めた20数年前とは、コンピュータの性質が変わってきています。それから、1台のコンピュータだけで処理が終わることも減っています。例えば、手元にスマートフォンというコンピュータがあっても、これ1台で動くことはめったになく、インターネットの向こう側の何十台ものコンピュータとお話をして結果を得るといった形が増えています。1台1台のコンピュータの性質も変わってきているし、使い方も、何台、何十台というコンピュータ全体で一つのシステムを構成するようになってきているのですね。プログラミングの世界でもそういう変化に対応していくことになると思います。

— プログラミング言語がますます発達して、より平易で簡便に扱えるようになると、専門知識を持たない人々がプログラミング活動を行うケースが増えることも予想されます。どこまで変化しそうですか？

レベルによるとと思いますが、一番ありそうなのが、プログラミングをしない人がどんどん増えていく姿だと思います。本当は万人がプログラミングする未来が欲しいですけど。

昔、パソコンが出始めた頃はコンピュータを使うこととプログラミングをすることがほぼイコールでしたが、現在は、コンピュータを使う方の中でプログラミングをされない方が圧倒的に多いと思います。一通りツール、アプリがそろっていて、プログラムなんか知らなくてもコンピュータは使える、という時代が既に来ているわけですね。その上、コンピュータを使ってやりたいこと、やらせたいこと、新しいことの大半が、特定のソフトを使うというよりも検索で済んでしまいます。あるいは、音声アシスタントにしゃべったら応えてくれて済んでしまいます。

もちろん、プログラミングする人、音声アシスタント

を作る人、検索エンジンを作る人、というのが、当然ある一定の割合で必要ですけれども、比率としてどんどん少なくなって、アシスタントとしてコンピュータを使う大多数の人と、プログラミングをする一部の人に分かれる、というのが一番ありそうな未来と考えます。

コンピュータの使い方では、今の企業の多くは、作るべきでないものを作って、作るべきものを作っていないと思います。例えば、給与計算や在庫管理などは、多くの会社でほとんど同じことをしているはずですが、それぞれの会社でそれぞれのソフトウェアを作っていたりします。我が社には何とかの都合とか、我が社独自のルールが、と言われますけれども、そうであれば独自のルールをやめたら良いですよ。同じルールにしてしまえばソフトウェアは一つ作ればよい。あとは、それをコピーすれば会社ごとに作らなくてもいいわけです。ありもののソフトを使えば投資しなくて済むのですから、浮いたお金で、ウチしかできないことをITでやろう、というところに投資すべきだと思います。投資すべきは投資し、投資しないで済むところは浮いたお金で新しいIT化に取り組む、という判断が必要になっていると思います。

経営判断をして、リソースを有効に投資しなければいけないと思います。結構多くの企業が簡単なIT化に無駄にお金を使って、価値ある新しいIT化の取組にお金を使っていないように見えます。

6. ITに関する経営判断について、もう少し詳しく問題意識をお聞かせください。

実際問題として、そう遠くない将来、全ての企業がIT企業になると思います。ITを活用するかどうかという選択が生命線になりつつあるわけですね。更に進んで、ITに対してどのような投資をするのか、あるいはどのような戦略的経営判断をするのかが、どの業種にあっても重要になります。この変化が既に起きているか、又は近い状態になっていると思います。

例えば、企業経営者の中には、「私はITは分からないから」とITを使えそうな部下にお願いして、お願いされた方もやっぱり分からないので、更にその部下に丸投げになっているような状況が確かにありますが、まずいですよね。経営に直結する判断は役員でなければできないという原則が当然あるわけですが、結果的に経営者でない人たちに判断が任されてしまうので「まずい」と思います。まずい企業の集まりが日本社会を構成すると考えると、日本の国そのものも結構「やばい」わけですよ。

そう思うと、ITとはどういう性質を持っているのか、ソフトウェアを作るとはどういうものかということ、

いろんな方々に理解していただきたいと思います。やっぱり日本の得意なものづくりとソフトウェア開発の性質は大分違うので。特に、ものづくりとソフトウェア開発との間に比喩を使うのは危なくて、例えば、「ちょっと仕事が遅れているので人を足しましょう」みたいな話も起こりますが、ソフトウェア開発に人を足すと遅くなる、というのは昔から言われていることです。その辺を理解してくれる人が増えてほしいな、と思います。

## 7. IT 社会で活躍する専門家として、どのような人材が望まれるとお考えですか？

「この人がいれば、何でも解決」と言うような人材はなかなかいないと思います。個人的には、自分の技術を武器に活躍して、自分の行く末を自分で決められるような生き方をする人が増えてほしいな、と思いますね。日本では、スタートアップ企業やベンチャー企業が弱いのですが、少なくとも東京近辺ではかなり改善されて多くのスタートアップが始まっていると思います。もちろん、必要とされている技術的な専門分野とか、あるいは態度みたいなものは、それぞれ違うと思います。しかしながら、プログラミングができるというのは、自分の選択肢を広げる大きな武器になるので、そういう武器を持って自分の選択肢を広げて、自分の行く末を自分で決められる人が増えてほしいと個人的には希望しています。

— 技術は持っているけれども活躍しきれない方を見かけたり、また、もう少し社会が支えれば、という思いを持たれることはありますか？

例えば、経済産業省の未踏プロジェクト<sup>注4</sup>や総務省の異能ベーション<sup>注5</sup>で将来の方向性が決まった、選ばれたことそのものを価値として自分のバリューを上げた、という方がたくさんいらっしゃいます。このようなプロジェクトの規模がより大きくなって、更にもっと言うと、民間からも支援が出るといいなと思います。税金を投入すると、失敗が何割だとか、失敗すると無駄金みたいな言われ方もしてしまいますが、むしろ、民間がベンチャー 10 件に投資した中で 1 件成功したら大丈夫です、と言うような形で、インキュベーション・プロジェクトを大規模にやって才能を見い

だせればいいと思います。

## 8. 児童・生徒を対象にしたプログラミング教育が活発になってきている状況を、どのように見ていらっしゃいますか？

そうですね、私も政府の審議会委員としてプログラミング教育の話もしますが、教育というと、いろんな人がいろんなイメージを持たれるのです。例えば IT 系の企業では常に人材が足りないわけです。そうすると早いうちから教育しておくことと未来のプログラマーが増えて、うちの会社に優秀なプログラマーが採用できるかもしれないというような思惑を持たれる方もいらっしゃいます。あるいは、日本は IT が弱いと言われているのはとがった人材が少ないのが理由だから小さいころからの教育で、と思われる方もいらっしゃいます。

ただ、私は、現実的に手が届くという意味で、学校教育での育成はちょっと無理だろうと思います。教材が要る、テキストが要る、先生も要るわけで、現実的に、すぐには難しいと思います。

したがって、プログラミングを体験してもらって、早いうちから興味を持つことができる人を発掘できればいいなと思います。理想を言うと、その先に、例えば少年団とか、部活動などの形で、発掘した子供を伸ばしていくという枠組みまでできるといいな、というふうには思います。

## 9. 最後に、今後の活動についてどのようにお考えですか？

技術者を続けられたらいいな、というのが究極のゴールです。やはり、年齢が高くなると、ソフトウェア開発の一線から外れてマネジメントに回りましょうとか、経営に回りましょうという話があるので、そうではなく、ずっと、ソフトウェアをデザインする人でいたいなと思いますね。そういうロールモデルになりたいと強く思います。オープンソースだけで生活している人はまだまだ少ないですし、それからシニアになってもずっと技術者でいるというのは、日本ではまだまだ少数派です。社会に対して、技術者のままでいいよ、というモデルを示すことを目指しています。

注 4 IT を駆使してイノベーションを創出できる独創的なアイデアと技術を有するとともに、これらを活用する優れた能力を持つ突出した若い人材を発掘・育成することを目的とした、情報処理推進機構の事業。

注 5 ICT 成長戦略の一つとして平成 26 年度に総務省が開始したプログラム。「破壊的イノベーション」の種になるような技術課題に挑戦する人材を支援する。