

平成 21 年度

ちゅうごく地域 Ruby ビジネス活用研究会報告

ちゅうごく地域 Ruby ビジネス活用研究会  
(中国経済産業局：ちゅうごく地域イノベーションパートナーシップ事業)

<はじめに>

IT 産業界において、システムの開發生産性の高さなどからオープンソースプログラミング言語「Ruby」が注目されており、また、開発者である「まつもとゆきひろ」氏が島根県松江市に在住することから、島根県を中心に IT ベンダーにその技術を持つ企業が増えてきています。

中国経済産業局では、平成 21 年度地域情報化基盤整備事業として島根県の IT ベンダーと広島県の IT ベンダーの連携による「Ruby 開発共通基盤」の構築プロジェクトや、Ruby によるクラウドコンピューティングでの業務システム開発プロジェクトなどを採択し、中国地域の IT ベンダーが連携することで新たなイノベーションを加速させていく支援を行っているところです。

こうした状況を踏まえ、生産性の高さ等から注目されている「Ruby」のビジネス活用を促進し、中国地域の IT 産業の活性化を図ることを目的として、以下のメンバーで構成する「ちゅうごく地域 Ruby ビジネス活用研究会」を立ち上げ、Ruby の優位性や課題等について議論してまいりました。

ここにその成果としてご報告いたします。

中国経済産業局

#### 「Ruby ビジネス活用研究会メンバー」

##### 《IT ベンダー》

###### ①情報産業団体

(社)鳥取県情報産業協会、(社)島根県情報産業協会、(社)システムエンジニアリング岡山、(社)広島県情報産業協会、(社)山口県情報産業協会

###### ②OSS 協議会

しまね OSS 協議会、  
NPO ひろしまオープンソフトウェアコンソーシアム、  
やまぐち OSS 協議会

##### 《IT ユーザー》

###### ①自治体

鳥取県、島根県、岡山県、広島県、山口県、松江市、広島市

###### ②企業など

中国経済連合会、中国電力(株)、広島銀行、山陰合同銀行

##### 《大 学》

島根大学(座長：島根大学総合情報処理センター長 野田哲夫教授)

##### 《事務局》

ちゅうごくイノベーションパートナーシップ事務局  
(株)広島ソフトウェアセンター、中国経済産業局)

## Ruby ビジネス活用研究会報告案

### 第 1 章 Ruby の現状（最近の活用事例なども盛り込んで Ruby のプロフィール）

- 1-1 クラウドコンピューティングとオープンソース・ソフトウェア
- 1-2 Ruby on Rails の登場と Ruby、Rails の評価
- 1-3 Ruby、Rails の普及と大規模化への対応
- 1-4 Ruby と地域産業振興の取組み（島根県・松江市）
- 1-5 Ruby と地域産業振興の取組み（中国地域）

### 第 2 章 IT ベンダー（開発側）から見た Ruby の優位性（利点）と課題について

- 2-1 Ruby 自体の言語特性による優位性（利点）と課題
- 2-2 Rails 活用事例を通じた優位性（利点）と課題
- 2-3 業務システム開発における Ruby、Rails の優位性（利点）と課題
- 2-4 小括

### 第 3 章 Ruby 開発が IT ユーザーにもたらすメリットについて

- 3-1 コスト削減効果の課題と可能性
- 3-2 調達における課題と可能性
- 3-3 クラウドコンピューティングと Ruby の可能性

### 第 4 章 Ruby 普及に向けた今後の取組み（提言）

- 4-1 Ruby、Rails のビジネス標準仕様に向けての取組み
- 4-2 ベンダー側の力量強化の取組み
- 4-3 ユーザー側の力量強化の取組み
- 4-4 Ruby 普及と中国地域競争力に向けた今後の取組み

## 第1章 Rubyの現状（最近の活用事例なども盛り込んでRubyのプロフィール）

### 1-1 クラウドコンピューティングとオープンソース・ソフトウェア

2006年ごろよりネットワーク経由でソフトウェアや情報サービスを利用する形態、いわゆるクラウドコンピューティング (Cloud Computing) が注目されるようになってきており、Google や Amazon、そしてセールスフォース・ドットコムなどの米国 IT 企業がこれを利用したサービスによって業績を拡大してきている<sup>1</sup>。クラウドコンピューティングは、情報の発信が企業から個人・グループへ移る側面を有するものの、むしろネットワークを活用して情報を集約している米 IT 企業に莫大な利益をもたらしているのが現実である。その結果、クラウドコンピューティングの進展は既存の IT ベンダーの衰退をもたらす可能性があるが、クラウドコンピューティングを活用する側にとっては地域格差や規模の格差を解消する可能性がある。これは地域の IT 産業や IT を活用する産業にとっても新たなビジネスチャンスをもたらすものであるが、これを現実化するためにはクラウドコンピューティングやオープンソースに関する技術的知識、そして情報を収集し発信する能力がより求められることになる。

一方、Linux に代表されるオープンソース・ソフトウェア (OSS) や、これによる新たなソフトウェアやシステムの開発はインターネットを利用して自主的に参加する人々が集まり (Bazaar 型の開発方式)、開発が続けられてきた。自由に利用できるソースコードと、その改修と機能追加の迅速な対応が可能となり、統一した規格や標準化もオープンな場で議論し、決めることが可能である<sup>2</sup>。これ自体クラウドコンピューティング的、あるいは集合知を活用した開発スタイルであり、クラウドコンピューティングのシステム自体にも多くの OSS が利用されている<sup>3</sup>。

---

<sup>1</sup> 情報化の新たな動きとしてのクラウドコンピューティングは中国経済連合会『提言 中国地方の新たな情報化戦略について』（2009年4月8日）を参照。

<sup>2</sup> 従来のソフトウェア開発の作業には膨大な時間、巨額の投資が必要であった。そこでソフトウェアが簡単にコピーされるなら企業も望むだけの収入を得ることができなくなるので、企業は OS などのソース・コード（ソフトウェアの設計図）の技術情報を隠すようになり、法的にコンピュータの内部情報は知的財産であるとして著作権で守られるようになる。また、コンピュータやソフトウェアの普及には互換性を進めていくために規格の標準化が必要である。これは公的な場で決められるのではなく、Windows に典型的に見られるように、よく売れたためにみなそれに従うというデファクト・スタンダードが力を持ち、結果として一つの企業が市場を占有するという覇権構造に直結してきた。

<sup>3</sup> Google は自らのシステムの大部分を自前で、OSS を組み合わせることで構築している。それ故、OSS のコミュニティを積極的に支援し、また毎夏には学生を対象にしたプロジェクト Summer of Code に約 200 万ドルの助成金を提供し、この中で OSS に関する開発者の育成と成果の獲得を効果的に行っている。また、OSS の呼び名が登場する以前よりオープンスタンダードを掲げ、1990 年代後半まで UNIX ベンダーとして成長を続けていた Sun Microsystems は、2006 年には OS の Solaris と開発言語の Java をオープンソース化するなど、OSS の開発コミュニティを活用した戦略を進めている。その OSS 化の象徴が Java で実装した Ruby の処理系である JRuby の開発であり、オープンソース・プロジェクトと

さらに、世界的な経済危機の結果、企業における IT 投資を含めた企業の投資活動において IT 投資に対する見方・評価が厳しくなる中で、オープンソースの活用によるコスト削減への期待が働いている。その結果、OSS 活用 IT ソリューション市場は 2007 年度から 2012 年度まで平均成長率 9.0% で推移するものと予測され、2012 年には IT ソリューション市場全体に占める割合は 13.6% まで拡大すると予測されている<sup>4</sup>。

そして、中央官庁では OSS の利用が比較的早くから進み、政府の施策で積極採用を推進しているため、オープンソース活用 IT ソリューション市場全体に対する割合が高く 2006 年度で 2270 億円（構成比 24.8%）、2007 年度で 2540 億円（構成比 24.2%）となっている<sup>5</sup>。今後は、中央政府はもちろん、自治体でもクラウドコンピューティングが進むことが予測されるため、当然自治体のシステムにおけるオープンソースの活用もより拡大するであろう。その結果、自治体のクラウドコンピューティングはオープンソースを活用する IT 産業の動向に大きな影響を与えることになる。

ただし、行政機関の情報化・ネットワーク化を進める電子政府の推進に合わせた日本政府のオープンソース活用政策や地方自治体への導入支援政策は、導入機関の側にも受注の側にもオープンソースの技術力・開発力、そして企画力が求められる。オープンソースの開発方式自体地域の情報サービス産業にとっても、新しいビジネス市場の拡大の可能性をもたらすが、これを実現していくためには、そのための技術力・開発力、そして企画力が必要となる。一方、地方自治体の中には、オープンソース活用政策を地域の情報サービス企業への支援と技術力向上、需要拡大につなげているケースも見られる<sup>6</sup>。

今後、クラウドコンピューティングの進展に対して、地域の情報サービス産業自身の対応が必要となるが、地方自治体も調達と産業振興の側面で大きな役割を担うことになる。

## 1-2 Ruby on Rails の登場と Ruby、Rails の評価

クラウドコンピューティングのシステム自体に多くの OSS が利用されているが、特にその中で、クラウドコンピューティングという言葉が登場する以前から、ネットワーク経由でのソフトウェアや情報サービスの利用を支える技術&サービスの一つとして

---

して開発が進んでいる。現在は Engine Yard が継承

<sup>4</sup> ミック経済研究所『OSS 活用 IT ソリューション市場の現状と展望 2008』、ミック経済研究所（2008）より。

<sup>5</sup> 同上書より。

<sup>6</sup> 代表的なのは長崎県における「電子県庁システム」の構築で、長崎県は 2001 年に民間から CIO を招聘し、文書管理、申請、入札の 3 機能を持つ電子県庁システムの基盤技術にオープンソースを採用し、地元の中小的情報サービス企業をシステム開発に参加させることを促進した。この方式によって従来の大手 IT ベンダーによる不透明な提案と見積もりを排除し、電子県庁システムを安いコストで導入することを可能にした。

2000年代半ばから注目されてきた Web2.0<sup>7</sup>の構築の中で、Ruby と Ruby on Rails が注目を集めてきた。

Ruby は、まつもとゆきひろ氏<sup>8</sup>により 1993 年に開発、1995 年に公開されたプログラミング言語であり、OSS としてその設計情報も公開されている。しかしながら、当初は Typo などのブログ構築システムの他は目立ったキラー・アプリケーションを持たなかったため、一部の技術者の間を除いてはビジネス分野では爆発的な普及はしなかった。

それが、2004 年にデンマーク人のプログラマである David Heinemeier Hansson により、Web アプリケーションフレームワーク (Web アプリケーション開発に共通する基本的なプログラム構造や機能セットをあらかじめ準備されたプログラム) である Ruby on Rails (以下 Rails と表記) としてリリースされ、これが上記の Web2.0 の流れの中で一気に注目を集めるようになったのである。そして、まず米国において、Twitter などの Web2.0 のサービスを行うサイトにおいて Rails が利用されるようになった。これは、Ruby が他のプログラミング言語に比べて記述量が少なくてすむ他、文法が英語に近く人間のイメージを表現しやすく、その結果開発の生産性が高いということが大きな理由である。Ruby はコンパイルを必要としないスクリプト言語であるためにプログラムを記述してすぐに実行→結果を確認できるが、その分実行速度は遅くなる。しかしながらハードウェアの性能の急速な向上によって処理の部分はハードウェアが担うことが可能になり、素早いリリースと頻繁な変更を求められる Web アプリケーションの開発においては Ruby の生産性が評価された。

Rails は Web アプリケーションフレームワークであるので、Web の開発を支援するものであるが、同じプログラミング言語である PHP や Java も同様のフレームワークを持っている。これらの他のフレームワークに対して Rails が注目されたのはその高い生産性である。これはまず、Rails がアプリケーションの開発を他のフレームワークより少ないコードで簡単に開発できるよう考慮し設計されている。Rails の基本理念は「同じことを繰り返さない」(DRY : Don't Repeat Yourself) と「設定よりも規約」

---

7 特定の技術やサービス、製品ではなく従来 (Web1.0) とは異なる新しいウェブの世界の特徴、環境変化、トレンドを総称したものである。テクノロジー関連のマニュアルや書籍の出版社である米国の O'Reilly Media の CEO、Tim O'Reilly によって提唱された。「2.0」という表現が使われたのは、1990年代半ば頃から普及・発展してきた従来型の Web の延長ではない、質的な変化が起きているという認識を込めたからである。従来の Web は製作者が作った状態で完結しており、利用者は単にそれを利用するだけの関係であったが、Web2.0 では先にあげたブログ、SNS、そして Wikipedia に代表されるように、多くのユーザーが参加して双方向で情報を出し合うことで、その蓄積が全体として巨大な「集合知」を形成するという点が象徴的である。

8 本名は「松本 行弘」だが、一般にはひらがな表記が定着している。英語圏では Matz の通称で知られる。現在は島根県松江市に在住し、同市内の株式会社ネットワーク応用通信研究所 (NaCl) にフェローとして勤務している。

(Convention over Configuration) であり、定義などの作業は一回だけですますことができるからである。

次に、Rails はユーザーからの検索が頻繁に行われる Web2.0 で欠かせない対話型 Web アプリケーションの実装形態である Ajax (Asynchronous JavaScript and XML)<sup>9</sup>に対応しており、Rails のインターフェース自体も Ajax を使った使いやすいユーザーインターフェースとして特徴的である。また、Rails ではプログラム本体よりも先にテストケースを書くテスト駆動開発(test-driven development; TDD)が可能で、データベースを用いたアプリケーションにおいてもテスト用のデータベースを簡単に生成できるなど、テストファーストを簡単に実践できる仕組みになっており、後述するソフトウェアを迅速かつ適応的に開発する開発手法＝アジャイルソフトウェア開発 (agile software development) に適している。

さらに、PHP や Java の Web アプリケーションフレームワークは多種多様であり<sup>10</sup>、そのため PHP や Java の開発エンジニアは多数であるが、それぞれのフレームワークに精通しているエンジニアは分散化されてしまう。一方、Ruby の開発エンジニアはまだこれらのプログラミング言語に比べれば少数であるがその増加速度は顕著であり (米国調査会社ガートナーの調査によれば 2013 年までに Ruby を使うエンジニアは 400 万人に達すると予測されている)、Web アプリケーションフレームワークは Rails にほぼ集約されるためにエンジニアの共通認識が得られやすいという利点がある。

その結果、Sun Microsystems や Microsoft、そして Apple などの米国の大手 IT 企業も、Ruby で作ったプログラムが自社製品で動作するようにその仕様を変え始めた。Sun Microsystems は既に Java 仮想マシン上で Ruby で書かれたプログラムを動作させることができる JRuby の開発を 2006 年から支援しており、Microsoft も Web アプリケーション開発技術である .NET に対応させるために IronRuby を導入、また Apple も同社の OS に対応させるために MacRuby を開発している。このような流れが、2006 年ごろから逆に日本でも注目されるようになった。

### 1-3 Ruby、Rails の普及と大規模化への対応

この結果、Web2.0 のサービスが日本でも普及していくのに伴い、日本においても、まず (2006 年ごろから) ビジネス分野では Web アプリケーション開発において Rails

---

<sup>9</sup> 画面遷移を伴わない動的な Web アプリケーションの製作が実現可能になる。例えば Web 検索に応用することで、従来は入力確定後に行っていた検索を、ユーザがキー入力をする間にバックグラウンドで行うことによってリアルタイムに検索結果を表示していくといったことが可能になる。

<sup>10</sup> PHP では動的な Web ページに向けた設計により、CakePHP、PRADO、Qcodo、Symfony、Zoop Framework、eZ publish Web 出版フレームワーク、Zend Framework などのフレームワークなどがある。

の導入が進んでいった。早くは TIS(株)が研究所レベルでの導入を開始し社内 SNS を Rails を使って構築 (2005 年 10 月)、(株)ドリコムがユーザー情報の管理やログイン・パスワードを一元化するドリコム・アカウントを Rails で構築 (2006 年 4 月)、ニフティ(株)が会員向け Web サービス「@niftyTimeline」を Rails で構築、同社のセキュリティツールを通して安全性をチェックしてサービスを開始 (2006 年 10 月)、楽天(株)が Web アプリケーションに Rails を採用し、まつもと氏も楽天技術研究所のフェローに (2007 年 5 月)、伊藤忠テクノソリューションズ(株)が社内トレーニングから Ruby、Rails の教育ビジネスへと発展 (2007 年) するなど、Web サービスの構築に Rails が次々と採用されるという大きな流れがあった。

そして現在、Web2.0 からクラウドコンピューティングへの転換、Web 上でのアクセスと処理の大規模化という流れの中で、Ruby、そして Rails も大量のアクセスへの対応が求められ始めた。前述のように、Web アプリケーションの分野における Ruby、Rails での開発の生産性は既に評価がされているが、インターネットの人気サイトでの大量のアクセスが発生した場合に処理能力に限界があり、様々な技術的な課題が表面化した。そこで、既に楽天(株)では上述の楽天技術研究所において大規模分散処理システムの Roma と Fairy<sup>11</sup>を開発し、Roma は 2009 年 10 月にオープンソースとして公開されている。カカコムやクックパッドなどのアクセスが大量に集中するサイトにおいても Rails でのサイト構築をすると同時に大量のアクセスに対する技術的な対処を行っている。(株)カカコムの運営するユーザーがレストランを評価しレーティングするクチコミ・サイト「食べログ.com」では大量のデータアクセスによって性能が低下する事態に対して<sup>12</sup>、Rails とデータベースの MySQL のチューニングによって対処し、新システムを稼動し (2007 年)、さらに新しいシステムを次々と追加することが可能になった。

また、Ruby や Rails が普及しつつあると言っても、その開発実績は他のプログラミング言語に比べてまだ少数であり、そして、Ruby による記述はプログラマの自由度が高い一方で、そのために大規模なシステム開発などの共同作業や作業の継続性に難点があるのも課題である。そこで、情報処理推進機構 (IPA) では Ruby アソシエーション (Ruby のビジネス分野での活用とその課題解決のために 2007 年設立) とともに 2008 年から Ruby を国際標準化機構 (ISO) へ登録する作業を行っており、Ruby の標準仕

---

<sup>11</sup> Roma はスケーラブル・ストレージに関する取り組みで、ディスクへのアクセス時間を短縮する技術。Fairy は、マルチタスク・ハンドリングに関する取り組みである。CPU のマルチコア化に対し並列処理時のリソース競合の問題をソフトウェアで解決する技術である。

<sup>12</sup> データベースへアクセスするための SQL 文を生成する機能は Rails の高い生産性をもたらす特徴のひとつだが、データが何十万件、何百万件という巨大な規模になった時、そのままでは実用的な性能が得られなかった。カカコムは MySQL に詳しいスマートスタイル社とネットワーク応用通信研究所に協力依頼し SQL の最適化を図り、性能改善を実現した。



様書の作成も進めている。

さらに島根大学では 2008 年度に Ruby を中心に、OSS の安定化・高度化のための手法の検討を進める研究プロジェクトを開始し、産学官と開発コミュニティの連携によるオープンソース・ソフトウェアの開発スタイルに関して、ビジネスモデルの構築と情報サービス産業の生産性に関する実証的・理論的研究を行っている<sup>13</sup>。これらの動向、ビジネス分野を中心とした Ruby の広がりや標準化の動向は 2009 年 9 月に松江市で開催された RubyWorld Conference で報告・議論され、国内外の注目を集めた。

#### 1-4 Ruby と地域産業振興の取組み（島根県・松江市）

このように Ruby、Rails のビジネス分野での普及、そしてその結果生じる開発の大規模化への対応などが国内外で進むのに対して、Ruby の開発者まつもとゆきひろ氏が在住し、また、まつもと氏が在籍する IT 企業・(株)ネットワーク応用通信研究所が存在する松江市では、Ruby を IT 産業振興のための「地域資源」として注目し、オープンソース Ruby を活用した地域の情報サービス産業振興政策として、Ruby City MATSUE Project を 2006 年度に開始した。これは Ruby 自体を「地域資源」とするというよりむしろ、Ruby によって開発者もユーザーもハッピーになれるというまつもと氏の想いそのものを体現したものであろう。

前述のように、オープンソースの開発方式自体は、地域の情報サービス産業にとってもその技術力を有していれば、新しいビジネス市場の拡大の可能性をもたらす。すでに(株)ネットワーク応用通信研究所 (NaCl) では設立当初 (1997 年) からオープンソース Linux を中心とした IT ソリューションのビジネスを進めており、2000 年には日本医師会のレセプトシステム ORCA プロジェクトに参加しその開発の中心を担うなど、特定のベンダーに依存しないシステム開発を行って全国的な市場を獲得してきた。そして、Rails の公開によって Ruby への注目が集まり始めることによって、Ruby、Rails による技術的優位性を活かした IT ソリューションビジネスや Ruby によるコンサルティングや教育などのビジネスを進めてきている。

Ruby はオープンソースのプログラミング言語であるので、何処で誰が Ruby を使って開発してもかまわないわけであるが、松江市はまつもと氏が在住する他、(株)ネットワーク応用通信研究所 (松江市) を中心に Ruby のエンジニアが集積するなどの地理的・技術的優位性を利用して、それを IT ソリューション市場の拡大、地域の産業振興につなげようとしたのである。その取り組みのスタートは 2006 年 7 月にオープンソースに関する情報交換・交流の場として、松江駅前ビルの 80 m<sup>2</sup>ほどの空きスペースに松

---

<sup>13</sup> OSS の導入と産業振興を進める米オレゴン州のオレゴン州立大学や国連大学マーストリヒト技術経済研究所など国内外の OSS 研究機関と連携して行っている。また技術的・工学的側面から Ruby 開発者のまつもとゆきひろ氏を客員教授として迎えている。

江オープンソースラボを開設したことであった。

そして、同年9月にはこの『松江オープンソースラボ』を拠点として、オープンソース・ソフトウェアに関わる民間企業、技術者、研究者、学生、ユーザーが交流を深めることで技術・競争力の向上と優れた人材の育成を図る組織として、しまねオープンソース・ソフトウェア協議会も立ち上がった。Rubyは世界的な、国籍を超えたコミュニティに参加する開発者の互恵的な精神によって支えられている技術であるが、松江市において開発者を中心とした地域のオープンソース開発コミュニティが立ち上がったのである。

このように、Ruby City MATSUE Project 自体は松江市という行政機関が主導したプロジェクトであるが、その主体は松江市内の民間企業や技術者、そして大学の研究者であり、Ruby という技術的優位性を活かしながら、産学官の連携によってオープンソース開発に必要な Communication と Collaboration を代替し、地域の情報サービス企業のプロジェクトマネジメント力を高めようとする計画であった。そしてオープンソースの開発スタイルに特徴的な Bazaar 型の開発スタイルを地域において実現しようという取組であった。

Ruby City MATSUE Project、オープンソース Ruby を活用した地域情報サービス産業振興政策のスタートアップにおいて、地域の企業に需要をもたらす点からも、行政(島根県や松江市)による開発発注は必要であった。島根県の CMS (2007年リリース)や、松江市の行政評価システム(2007年リリース)、松江市空き店舗物件情報サイト(2008年リリース)は、Ruby で開発が行われた。そしてこれらのシステムの多くはオープンソース化されていった。現在は松江 SNS (2010年リリース予定)や、島根大学の教員評価情報データベースシステム(2010年リリース予定)や、島根県の業務システムなど、地元企業によって Ruby、Rails を使った開発が進められている。

また、Ruby が注目されるようになったのは Rails の登場による Web アプリケーション開発市場の拡大であることが大きい。一方、今後の Ruby の市場拡大を目指して、Ruby による基幹業務の開発も進められてきた。(株)テクノプロジェクト(松江市)は2007年度に IPA の公募事業によって松江市の医療・介護の高額合算システムを Ruby で開発した。この過程で Ruby による基幹業務開発の信頼性・安定性とそこでの生産性の測定が行われ(後述)、同時に開発に COBOL エンジニアがあたることによって、基幹業務開発のノウハウを持つ COBOL エンジニアの「復活」を可能にした。

これらの流れは、行政による地域産業への直接支援=公共事業の側面を持つが、地元の IT 企業はこれらの事業・開発を進める中で、Ruby 開発の技術力・プロジェクトマネジメント力を高めていった<sup>14</sup>。そこで、島根県の IT 企業がしまねオープンソース・

---

<sup>14</sup>行政=発注側の業務システム開発をマネジメントする能力が弱かったことが、地元企業の能力向上につながった面もある。

ソフトウェア協議会に参加する他<sup>15</sup>、しまねソフト産業ビジネス研究会が設立され、OSS や Ruby を活用したビジネス拡大、市場獲得のための研究と、大規模開発の県内企業による共同受注を進めている。

そして、この過程で Ruby による技術力を高めていった地元企業や進出企業によるビジネス向けのソリューション開発も進んでいる。小松電機産業(株) (松江市) が i モードとインターネットによる上下水道施設・社会インフラの管理監視制御を行うソフトウェア「やくも水神」を Ruby で開発 (2008 年)、(株)マツケイ (松江市) が帳票印刷を伴う業務ソフト・日本下水道協会島根支部の工事責任者管理システムを Ruby で再構築 (2009 年) するなど、地元企業による Ruby の技術力向上を伴った業務システム開発が進んでいる。また、2007 年に松江市にソフトウェア開発拠点を開設したバブ日立ソフト(株) (広島県呉市) は Ruby を使った中小飲食企業向け販売システムを開発 (2009 年)、2008 年には日立ソフトウェアエンジニアリング(株)が同様に開発拠点を設けて地元企業と協力してビジネスソリューションの開発研究を進めるなど、島根県・松江市における Ruby 開発の優位性を示している。これらの地元企業の取組みも、2009 年 9 月に松江市で開催された RubyWorld Conference で報告され、国内外に Ruby City MATSUE を幅広くアピールした。

一方、地域産業振興のためには市場拡大と同時に、開発を担う人材の育成が必要である。島根県は 2007 年度から企業エンジニア向けの Ruby、Rails のエンジニア育成講座を開設して人材育成事業を進める他、地元の教育機関での OSS の利活用や Ruby の実践教育の導入を進めている。そこで島根大学では 2007 年度から Ruby、Rails の実習と Ruby に関わる開発者、エンジニアの講義を聞き、Ruby やオブジェクト指向言語の今後の発展、活用の可能性について探る「Ruby プログラミング」講座を開設、同様に松江高専・松江商業などの地元の教育機関においても Ruby、Rails の講義・授業などが進められてきた。人材育成に関してはすぐに効果が表れるものではないが、継続していくことによって地域産業振興につながることを期待される。

これらの取組みの結果、島根県の情報サービス企業 (その 70%は松江市に開発拠点がある) 全体の売上高や就業人数も 2006 年度から 2008 年度の間に全国を上回る伸びを示している (表 1・2、図 1・2 参照)。

表 1 全国と島根県の情報サービス企業の売上高推移 (単位：百万円)

	全国	伸び率	島根県	伸び率
2006年度	13,751,730		10,452	
2007年度	13,409,700	-2.5%	12,060	15.4%
2008年度	14,817,900	10.5%	13,241	9.8%

<sup>15</sup> 設立当初の参加企業は 10 社ほどであったが、現在 (2009 年 9 月現在) では 30 社の企業が参加し、毎月のように Ruby を中心とした技術研修だけでなく、オープンソースに関わる経営者、開発者、研究者を招いた研究会を開催している。

表2 全国と島根県の情報サービス企業の就業者数推移（単位：人）

	全国	伸び率	島根県	伸び率
2006年度	567,498		1,022	
2007年度	501,807	-11.6%	1,389	35.9%
2008年度	557,263	11.1%	1,537	10.7%

出所：経済産業省特定サービス産業実態調査、社団法人島根県情報産業協会調査

図1 全国と島根県の情報サービス企業の売上高比較（2006年度を1とした場合）

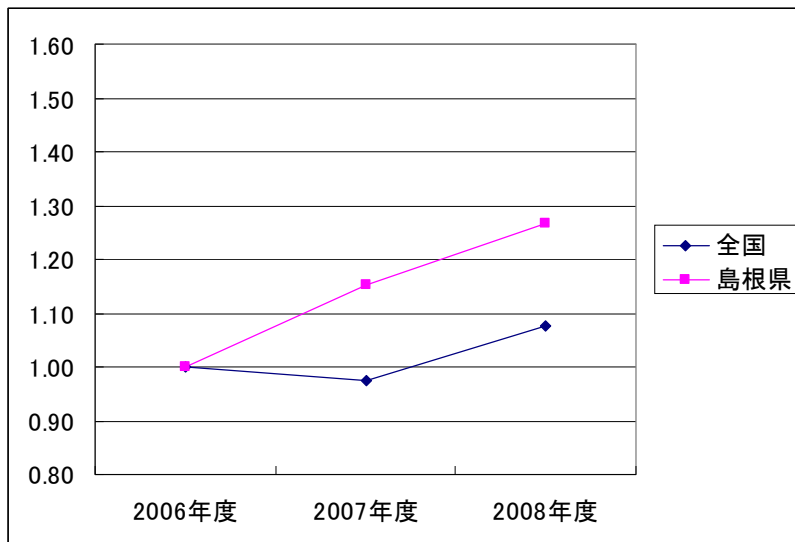
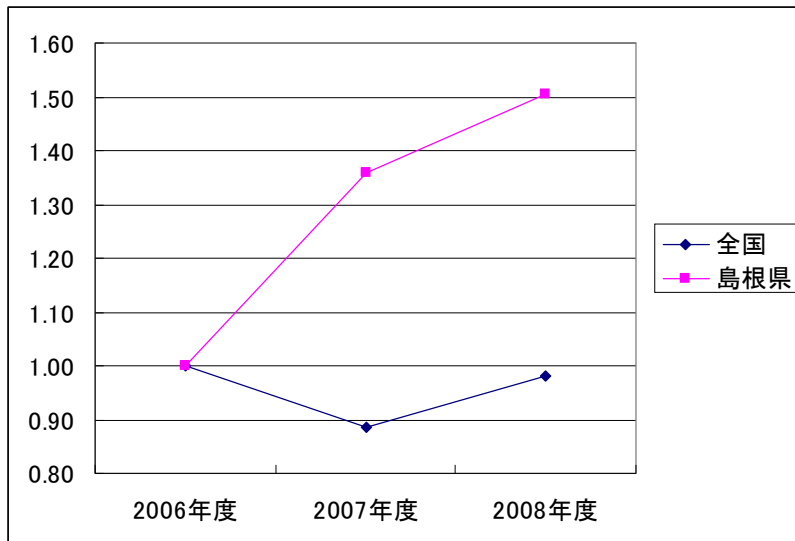


図2 全国と島根県の情報サービス企業の就業者数比較（2006年度を1とした場合）



#### 1-5 Ruby と地域産業振興の取組み（中国地域）

Ruby、Rails の普及は、島根県・松江市にのみ IT ソリューション市場の拡大をもたらすものではない。前述のように、Web アプリケーション開発を中心とした市場の拡大は、東京の IT ソリューションベンダーにも Ruby による開発、そのための人材育成・研究教育にも目を向けさせるようになっており、またこれはその他の地域でも同様である。既に 2007 年に Ruby のビジネス分野での活用とそのための人材育成を目指した組織 Ruby ビジネスコモンズが設立され、福岡を拠点に全国各地で勉強会が開かれているが、2008 年には福岡県を Ruby ビジネスの推進拠点に、ソフトウェア産業の更なる発展を目指す「福岡 Ruby ビジネス拠点推進会議」(F-Ruby) が設立されている。また、日本の Ruby コミュニティのカンファレンスとして 2006 年から毎年首都圏で日本 Ruby 会議が開催されているが、各地域での Ruby コミュニティの盛り上がり（その背景にある Ruby、Rails エンジニアの増加）とそれぞれの地域での Ruby、Rails のビジネス分野での拡大を背景に、2009 年からは毎月のように各地域（札幌、仙台、栃木、名古屋、関西、福岡、松江、広島など）で地域 Ruby 会議が開催されており、全国の「地域」での Ruby の普及を示している。

これらの動向は決して Ruby、Rails による IT ソリューション市場を奪い合うものではなく、むしろ Ruby、Rails のエンジニア、開発企業の拡大によって Ruby、Rails による開発の大規模化、継続性が保証されることにつながり、Ruby、Rail による IT ソリューション市場全体の拡大につながるものである。もちろん拡大した市場の中で、これを獲得していく部分は企業間・地域間の競争となり、そのためには技術力・開発力における優位性が必要となる。これは中国地域でも同様のことであり、前掲の中国経済連

合会の提言書においても、**Ruby**を活かしてプログラマを育成し競争力を高めることの必要性が謳われている。既に技術的な優位性を持つ松江において**Ruby**を中心としたIT産業の共同開発を通じたクラスターができあがりつつある中で、これを中国地域全体に拡大していくチャンスでもある。

すでに2009年度より中国地域においても**Ruby**を活用する共同開発プロジェクトが進んでいる。(株)マイティネット(広島市)は(株)アクトシステムズ(福山市)、(株)テクノプロジェクト(松江市)、(株)マツケイ(松江市)と共同で**Ruby**の開発のための支援システムを構築している。また(株)ミウラ(広島市)は(株)ネットワーク応用通信研究所(松江市)と共同で、**Ruby**を使った放送業界向けシステムの研究・開発を行っている。

また、岡山県においては、(社)システムエンジニアリング岡山の主催により**Ruby**をテーマとした「おかやまOSS研究会」が開催され、岡山県の関連サイトである「おかやまコンテンツスクエア」のホームページについても、開発言語に**Ruby**を指定して、地元企業による開発を行う動きが出ている。

そして、**Ruby**、**Rails**の活用は、後述するようにITベンダー(開発側)とITユーザー双方にメリットをもたらすものである。開発によるIT産業振興だけでなく、地域産業全体の高度化にもつながるものである。**Ruby**を中心としたIT分野の生産性・イノベーションが地域産業全体の生産性の高度化・イノベーションにつながり、閉塞感のある地域経済の打開の展望を示しうる。

## 第2章 ITベンダー（開発側）から見た Ruby の優位性（利点）と課題について

### 2-1 Ruby 自体の言語特性による優位性（利点）と課題

Ruby は他のプログラミング言語に比べて記述量が少なく、すむ他、文法が英語に近く人間のイメージを表現しやすく、その結果開発の生産性が高いということが言われてきた。そのため素早いリリースと頻繁な変更を求められるネットビジネスなどの Web アプリケーションの開発においては Ruby の生産性が評価された。その際に、Rails を利用して 10 分程度で Web アプリケーションを開発する動画が公開されたり、「Ruby は Java の数倍の生産性がある」と言われてきた。また、楽天技術研究所 所長の森正弥氏の「楽天市場の実際のサービスを開発してみた結果、Ruby の生産性は他言語の 1.6 ~ 3 倍だった」（2007 年 6 月 10 日、日本 Ruby 会議 2007）といった報告もある。実際にこのような生産性の優位が証明されるのであれば、Ruby を開発の生産現場に導入することは開発の生産性を上げ、エンジニアのストレスも減少させることができる。そこで、Ruby 自体の他の言語と比較した生産性を計測するため、中央情報システム(株) (広島)の協力を得て同じ開発経験年数(7 年)のあるエンジニアでそれぞれ Ruby と Java、Perl で同機能を有する Web システム<sup>16</sup>を、あえて Web アプリケーションフレームワークを使わずに開発し、双方の生産性の比較を行った。

表 4 各言語比較結果

使用言語	Java	Ruby	Perl
行数 (コメントを除く)	177 行	46 行	42 行
工数 (製造+テストの時間)	製造：8 時間 テスト：1 時間	製造+テスト：2 時間	製造+テスト：0.75 時間
読み込み モジュール数	(require 数) 19	(require 数) 2	(use 数) 4
動作条件	Servlet コンテナ必須	一般的な Http サーバ	一般的な Http サーバ
動作確認サーバ	Tomcat	Apache Anhttp (他 CGI 動作可能サーバ)	Apache Anhttp (他 CGI 動作可能サーバ)
使用言語 経験年数	7 年	0 年 (開発経験は 7 年)	5 年 (開発経験は 7 年)

この結果より、生産性においては Java に比較してコード量、製造時間共に大きく上

<sup>16</sup> 作成したシステムは名前、コメントの投稿できる伝言板システムで、多重投稿防止機能や必須チェック、古い投稿の自動削除機能を有するものである。

回ったことが分かる。製造時間だけみれば Ruby が Java の数倍の生産性がある（単純比較すれば 4.5 倍）ことが実証された。また、同じスクリプト言語である Perl と比べても、Ruby で初の製造であるにも関わらず、5 年経験した Perl とほぼ同じソースコード量で製造可能であることがわかった。製造時間に関しては倍以上かかっているが、コーディングに慣れれば Perl 以上の速度が実現可能であろう。

また、動作環境を比較しても、コーディングしたシステムを動作させるのに専用のコンテナを必要とせず、外部から読み込むモジュールも 2 モジュール（Java は 19 モジュール、Perl は 4 モジュール）であった。

一方、Ruby は書籍や Web 上の情報も充実してきたが Java と比べてコーディング初級者向けの情報が圧倒的に少なく、Perl、PHP と比べても実績が少ない点、また「簡単にコーディングできるコード」は一步間違えると「見づらいソース」になりかねない点、すなわち共同作業や継続性のある作業に向かない点などが指摘された。

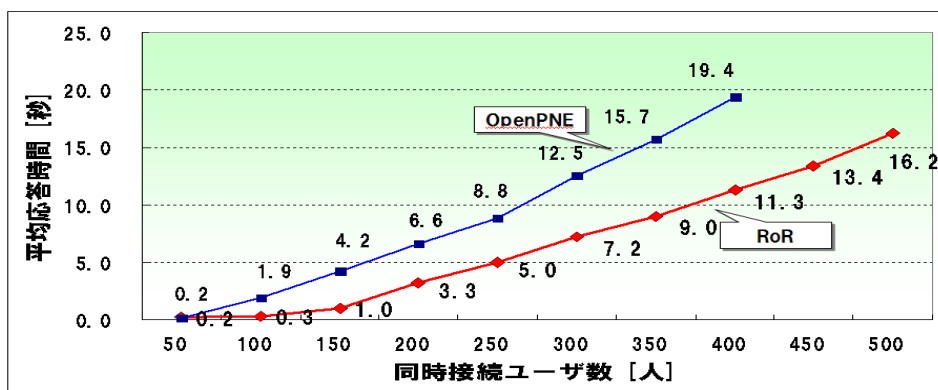
そこで、開発事例を踏まえた上で、開発環境や開発手法および標準化に関する課題をまず洗い出すことが必要である。そして、初級者用の、統一した仕様の情報の確立と提供が求められる。

## 2-2 Rails 活用事例を通じた優位性（利点）と課題

Ruby が注目されるようになったのは、Web2.0 による Web アプリケーション開発需要の拡大と軌を一にした Web アプリケーションフレームワーク、Rails の登場であり、Ruby 同様に高い生産性が言われている。

一方、Ruby がインタプリタであるために実用に耐えられるか、不良の発生状況、拡張性の検証、などの信頼性が問題として指摘されている。

日立ソフトウェアエンジニアリング(株)では、OpenPNE で開発されていた社内のコミュニケーションシステムを Rails で書き換える開発を通して、まず OpnePNE と比較した信頼性の検証を行った。



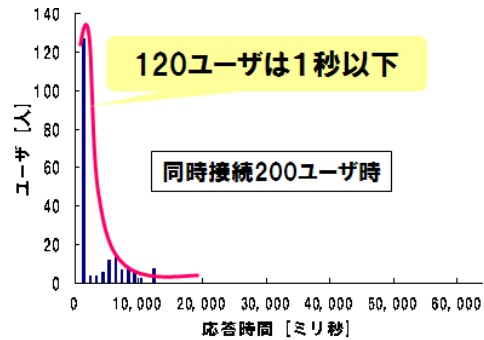
OpenPNEにおいてユーザ450、500は無応答エラーが発生し、測定不可



## 実用に耐えられるかの検証

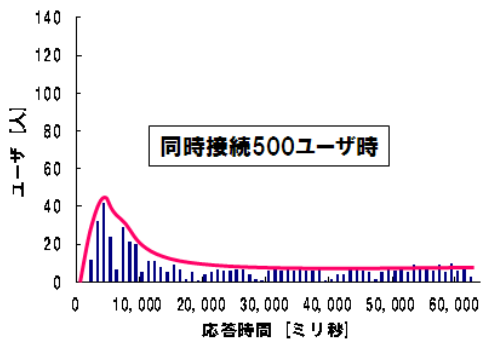
### ◆ 200ユーザ接続時

- ✓ 平均応答時間は、3.3秒であるが、応答時間1秒以内が120ユーザとなる
- ✓ 最大値は12秒
- ✓ 日々の利用には問題のない性能



### ◆ 500ユーザ接続時

- ✓ 平均応答時間は、16.2秒であるが、応答時間3秒以内が40ユーザとなる
- ✓ 1秒以内はゼロ
- ✓ 20秒以上は250ユーザとなる
- ✓ 最大値は60秒
- ✓ ハードウェアの増強が必要と考える



## 不良の発生状況

### ◆ テスト方法

- ✓ 開発メンバーによるモンキーテスト
- ✓ テスト環境を使用し、数十名によるユーザテスト

### ◆ 平均修正工数: 2.6 [h/件]

本番リリース

イテレーション	#1	#2	#3	#4	#5	#6	計
バグ [件]	31	8	1	1	16	0	57
修正工数 [h]	87.0	21.6	3.0	1.0	38.4	0.0	151
件数あたりの修正工数 [h/件]	2.8	2.7	3.0	1.0	2.4	-	2.6

### ◆ イテレーションの進捗で修正に掛かる工数に変化はない

(※イテレーションとは、反復の意)

次に Rails の開発による生産性であるが、これはまず、

- (1) Rails が、Ruby によって、アプリケーションの開発を他のフレームワークより少ないコードで簡単に開発できるよう考慮し設計されている。
- (2) Rails に用意された「設定よりも規約」(Convention over Configuration) に従

ってプログラミングすることでプログラミングの量を減らすことができる。

- (3) Rails はクラスを継承することによって、クラス定義のみで「同じことを繰り返さない」(DRY : Don't Repeat Yourself) でデータベースの操作が可能である。といった技術的な要因があげられている。

そこで、(株)トスコ (岡山市) の協力を得て以前 (2007 年) に Java および Web アプリケーションフレームワークの JBoss Seam<sup>17</sup>によって開発された勤怠管理システムの焼き直し+追加機能要件を Rails によって行った。まず、コード量を比較するために、ソフトウェアの設計モデルで処理の中核を担う Model、表示・出力を司る View、入力を受け取ってその内容に応じて View と Model を制御する Controller の 3 要素に対して、それぞれのステップ数を比較した。

表 5 コード量の比較

分類	ソース行数 (Rails)	ソース数 (前回開発)	ソース数 (前回開発) ÷0.7
Controller	5.1K	18.4K	26.3K
Model	1.2K	12.6K	38.1K
View	4.2K	4K	5.7K
合計	10.5K	35K	50K

※ 前回開発環境 : Java、JBoss Seam (JSF、EJB3.0)  
前回開発規模 : 7 機能、今回開発の約 6~7 割

単純にステップ数を比較すると 3 分の 1 であり、また前回の開発規模が今回の開発 (Rails) の約 6~7 割であることを考えると、同等開発規模レベルで 5 分の 1 であることが分かる。

また、プログラマの開発生産性を比較するためにファンクションポイント法<sup>18</sup>による測定を行った。

<sup>17</sup> JBoss Seam は、Java EE 5 アプリケーションにおいて EJB 3.0(Enterprise Java Beans 3.0)のコンポーネントモデルと JSF(JavaServer Faces)のコンポーネントモデルをシームレスに統合するための Web アプリケーションフレームワーク。

<sup>18</sup> ソフトウェアの規模を測定するための手法の一つで、ソフトウェアが備える機能の数や複雑度などをもとに、ファンクションポイント (FP) と呼ぶ値を算出する。FP 値はソフトウェアの機能や画面、帳票の数が増えると FP 値も増える。人月当たりの FP 値が高いということは 1 人あたりの生産性が高いことを示す。

その結果、FP/工数の比較では 1.4 倍の生産性（費用では約 7 割）であることが認められた。

この 2 つのケースは、いずれもプログラム開発と実テストを行うのみであり、設計書やテスト手順・成績書などのドキュメント作成を行っていない点はあるが<sup>19</sup>、すでに存在するシステムを Rails による Web アプリケーションに書き換える場合に、仕様を熟知している人が行えば驚異的な生産性を発揮することが証明された。

一方、Ruby の場合と同様にソフトウェアの資産や開発のための手引書が少ないことも課題となった。これらを払拭するためには、まず開発案件を増やすことによって事例・資産を増やすことが重要である一方、「自治体主導の積極的活用」と「ユーザメリットの見える化」（後述）も望まれる。

また、Web アプリケーション開発の際には、開発対象を多数の小さな機能に分割し、反復（イテレーション）開発によって機能を開発していくアジャイル開発の手法が適していると言われている。そして Rails は Web を素早く、見栄えよく作れる アプリケーションフレームワークであり、Ruby は読みやすく、変化を受け入れる言語であるために、アジャイル開発に最も適していると評価されている（(株)永和システムマネジメント平鍋健児氏談）。

前述の日立ソフトウェアエンジニアリング(株)では、Web アプリケーションを Rails によってアジャイル開発した場合の生産性の測定を、社内のコミュニケーションシステムを Rails で書き換える開発を通して行った。

アジャイル開発の場合は（今回の開発の場合は）仕様書は作成していないので、従来のソフトウェア開発手法＝ウォーターフォール型の開発と単純に比較することはできないが、生産性は

$$\begin{aligned} 8,119 \text{ [step]} / 638 \text{ [h]} &= 12.7 \text{ [step/h]} \\ 638 \text{ [h]} / 99 \text{ [画面]} &= 6.4 \text{ [h/画面]} \\ 8,119 \text{ [step]} / 99 \text{ [画面]} &= 82 \text{ [step/画面]} \end{aligned}$$

であった。（比較値になっていない）

また、人材の確保の面で、技術者の育成や作業の標準化が必要であった。

まず技術者の育成に関しては「Ruby プログラミング講座」（2 日間）・「RoR プログラミング講座」（2 日間）で基礎教育・応用教育を行い、実際に動作するプログラムを動かしながら学習した。作業の標準化は、開発に必要な各種ガイドライン（クラス設計、命名規則、コーディング規約など）を作成した。さらにライブラリを使いこなすことが困難であるので（オープンソースであるが故の課題、メンテナンスされていないライブ

<sup>19</sup> 通常ビジネスの現場ではドキュメント類が必須である。

ラリも多く、使用している環境で動作しないケースもある)、同社で有用なライブラリ(よく利用されているライブラリを検証し、有用なものを選別)を開発環境と共に提供した。

このように、Web アプリケーション開発に Ruby、Rails はその特性を発揮、生産性を向上させるとしても、これは Ruby、Rails の開発経験を有するエンジニアを前提としたものであり、Ruby の開発年数が少ないエンジニアを含むチームでの開発や、後述する開発の継続性を考えた場合、Ruby、Rails の人材育成と作業の標準化、ライブラリ整備などを合わせて行うことが必要となる。

### 2-3 業務システム開発における Ruby、Rails の優位性(利点)と課題

Ruby が注目されるようになったのは Rails の登場による Web アプリケーション開発における活用の拡大であることが大きい、一方で今後の Ruby のビジネスでの活用拡大のためには、基幹業務での採用も想定される。

(株)テクノプロジェクト(松江市)は2007年度にIPAの公募事業によって松江市の医療・介護の高額合算システムを Ruby で開発し、また、(株)マツケイ(松江市)が帳票印刷を伴う業務ソフト・日本下水道協会島根支部の工事責任者管理システムを Ruby で再構築した(2009年)。この過程で Ruby による基幹業務開発の信頼性・安定性とそこでの生産性の測定が行われた。

まず、医療・介護の高額合算システムは医療・介護の高額合算システムは、従来の5人前後で行われる Web アプリケーション開発とは異なり、比較的大規模な開発チームで、必ずしも Ruby に習熟していない開発者も加わった体制であり、開発を効率的に進めることができる手法を確立することが課題であった<sup>20</sup>。

最初に、信頼性の面であるが、基幹業務に採用する点で問題となる演算精度の点で、Ruby に標準で添付されているライブラリで固定小数点演算がサポートされており、消費税演算など問題はなかった。また安定稼動においても、Ruby および Rails+MySQL・Apache・Linux の Web アプリケーション開発での多くの実績があり、予期せぬ動きや落ちたりするといった問題はなく安定性が保証された。

次に生産性での計測であるが、今回の開発においては従来の業務システム開発のノウハウを活かす面からもウォーターフォールの開発手法を採用した。

---

<sup>20</sup> この開発プロジェクトは(株)テクノプロジェクトで13名(フルに関わるメンバーが7名)に加え、(株)ネットワーク応用通信研究所、伊藤忠テクノソリューションズ(株)などを含め全部で37名と、Ruby によるシステムとしてはかなり大規模な体制になっている。

	今回の構成（従来型）	Ruby/Rails で一般的な構成
開発プロセス	ウォーターフォール	アジャイル
データベース設計	自然キー	人工キー
プログラム構成	関数ベース（規約で決まってい ない部分）	クラスベース

そこで開発の前提条件として

①設計工程（基本設計、詳細設計）	30%
②製造工程（PG 設計、プログラミング、単体テスト）	40%
③テスト工程（結合テスト、総合テスト、運用テスト、操作研修）	30%

というように工程が分割され、設計工程での各種ドキュメントの作成、テスト工程でのテスト仕様書の作成は、従来の基幹開発業務と同様に顧客納品物として必要であるため、この工程での「改善」は行わなかった。一方製造工程においては Ruby、Rails の統合開発環境 IDE を利用することによってプログラム修正から単体テストがスムーズにできるなどの生産性の向上があり、25%の改善があった、その結果

設計工程（30%）+製造工程（40%）×0.75+テスト工程（30%） →全工程（90%）・・・10%の改善
---

という結果であった。

SIer として ((株)テクノプロジェクトとして) 基幹業務にはアジャイル開発は適さない（何回繰り返しが発生するか不明なので請負契約ができない、納品物として、設計書やテスト成績書が必要となる）との判断で、以下のような開発手法を採用した結果である。

<p>当初計画（設計・製造・テスト）に基づき、1回の開発でシステム構築を完了させる。納期厳守。</p> <p>① 開発要員：設計工程から PG 要員も参加させる。プロジェクト全体を通して同一メンバー</p> <p>② 設計書・結合/総合テスト手順書/成績書を作成する。プログラム設計書は作成しない。</p> <p>③ 設計工程での PG 要員の作業は、SE 業務のサポート業務ノウハウ・新技術・フレームワークの習得である。（要員教育の前倒し、SE 業務の効率化）</p>
---

この開発の結果、チームワークを保ったモチベーションの高い協力体制が構築され、

製品の品質にもつながった。また、工程間のドキュメントを最小限にし、製造工程での生産性上昇（仕様の理解度・技術の習熟度が高い）が達成されたとの評価である。

またセキュリティ面でも **Rails** が提供するセキュアな機能の効果と、開発規約を用いてセキュアな実装を徹底することにより、堅牢なシステムを構築することが実証されている。

基幹業務は変化に対応して 10 年 20 年の長期間に渡って利用され、メンテナンスにおいての継続性が必要とされる。**Ruby** は言語仕様が豊富であり、開発者が自由に組める言語であることが特徴的であるが、これは継続性とは両立し難い部分がある。このためこの基幹業務の開発においてはこれを大幅に制限し管理された開発を行った。

- ① 標準化＝コーディング規約を厳しく設ける。
- ② ソースレビューの徹底（スパゲッティ状態にならないこと）
- ③ 豊富な **Ruby** の言語仕様に制限して利用する。

これは（特にコーディング規約の作成は）**Ruby** の基幹業務での開発のみならず、今後のビジネス分野での活用においても課題になるであろう。これは同時に、**Ruby** の特性である多様性を制限してその利点を失わせる面もあり、今後は開発対象によって適性を的確に判断して、活用する必要があるであろう。

一方、この高額合算システムや工事責任者管理システムの開発で得られたノウハウは MIT ライセンス<sup>21</sup>として公開を行っており、バッチ処理機能と帳票出力環境のフレームワーク、業務フローや設計書、ソースコードといった業務システムは誰でも自由に利用できる状態にしてあり、ノウハウの蓄積も進んでいる。

#### 2-4 小括（人材育成・開発情報の標準化・業務システム開発における課題）

**Ruby**、**Rails** の開発における生産性の高さはある程度実証されたが、**Java** や **PHP** など **Web** アプリケーション開発で主流であるプログラミング言語に比べて開発経験者は未だ少なく、実績も少ない。**Ruby**、**Rails** 関連の書籍や **Web** 上の情報も充実してきたがこれも **Java** と比べてコーディング初級者向けの情報やビジネス向け教育コースが圧倒的に少ない。

しかしながら、プログラム開発の経験があれば、特に **Java** や **PHP** の経験があれば、**Ruby**、**Rails** の習得にはさほどの時間は必要ない。そこで、プログラミング経験者が

---

<sup>21</sup> マサチューセッツ工科大学を起源とするソフトウェアライセンスの代表的なものである。コピーレフトでは無いが、オープンソースであるか無いかに関わらず再利用を認めている。BSD ライセンスをベースに作成された BSD スタイルのライセンスの一つである。

Ruby、Rails を使って開発を行う場合は、その教育期間・コストを加味しても、全体の生産性は Java や PHP と比較して大幅に向上すると考えられる。また、システムエンジニアリングの経験に優れる COBOL エンジニアのモチベーションを上げ、「再生」させることも可能である。

一方、Ruby はコードの多様性がプログラマにとっての魅力ではあるが、それは同時に共同作業や継続性のある作業に向かないという面もある。またこれと関連して、Web アプリケーション開発に向くとされるアジャイル開発において、Ruby、Rails はその特性を発揮、生産性を向上させるとしても、ビジネス分野でのチームによる開発や、更新作業やメンテナンスも含めた開発の継続性を考えた場合、Ruby、Rails の人材育成と併せて Ruby、Rails 統一した仕様情報の確立、作業の標準化、ビジネス分野で活用可能なライブラリ整備などを合わせて行うことが必要となる。これらが平行して行われれば、Ruby、Rails の生産性も発揮ビジネス分野で発揮されることになる。

Ruby、Rails は OSS であるために開発がコミュニティによって行われており、ライブラリのメンテナンス、バージョン対応やマルチプラットフォームでの作動などで整備が不十分であるが、Ruby アソシエーションを中心にビジネス分野での Ruby 活用の対応が進められており、また日立ソフトウェアエンジニアリング(株)などの大手 IT 企業では自社内で作業の標準化やライブラリの整備などを進めている。また、Ruby、Rails を活用した島根県 CMS や松江市の SNS 開発とそのオープンソース化などによってライブラリ化が進んでいる。今後、各企業の企業努力によって進められていく部分もあるが、Ruby、Rails のビジネス拡大を進めるための共通基盤の整備も求められる。また、ビジネス分野や基幹業務に不可欠である帳票処理などのバッチ処理についても、開発の生産性やメンテナンス（他の言語との混在よりは Ruby で一貫したほうが生産性が高い）などを考えた場合、Ruby で開発して共通基盤化することが求められる。

さらに、基幹業務での Ruby、Rails による開発拡大を図る方法の一つとして、アジャイル開発の課題（何回繰り返しが発生するか不明なので請負契約ができない、納品物として、設計書やテスト成績書が必要となる）の克服があげられる。現状では製造工程をアジャイル開発の要素の一つである繰り返し開発で行い、設計工程とテスト工程を従来のウォーターフォール開発で行う方法も可能である。その中で Ruby、Rails の生産性をより高く発揮しようとした場合、マイルストーンを決めてリリースしていくなど、ユーザーとのコラボレーションを含んだ方策も考えられる。これはユーザー側のスキルや理解、契約関係とも関係する課題であり、ユーザー側からの分析が必要となる。

## 第3章 Ruby 開発が IT ユーザーにもたらすメリットについて

### 3-1 コスト削減効果の課題と可能性

Ruby、Rails はオープンソースのプログラミング言語であり、動的なウェブコンテンツを含むウェブサイトの構築がオープンソースを組み合わせた LAMP<sup>22</sup>が主流となり、スクリプト言語である Ruby、そして Web アプリケーションフレームワークの Rails も Perl、PHP、Python とともにこの中に位置づけられつつある。そして、Web2.0 やクラウドコンピューティングの流れ、また世界的な金融危機の結果、企業における IT 投資を含めた企業の投資活動において IT 投資に対する見方・評価が厳しくなる中で、OSS の活用によるコスト削減への期待が働いていることによって、オープンソースを活用した IT ソリューション市場が拡大する中で、Ruby、Rails への注目度が高まるのも必然である。

また、Ruby はプログラミング言語であるので Ruby、Rails で開発したアプリケーション、システム（Ruby 自体の開発ではない）を必ずしもオープンソース化する必要はないが、他のスクリプト言語同様に Ruby で開発されオープンソース化された CMS なども存在する。

そこで、Ruby、Rails を含むシステムを調達することは、従来のエンタープライズ系のシステムに比べてコストを削減することが可能であり、Rails というフレームワークがオープンソースであるためにシステムの更新などにおいて特定のベンダーにロックインされないというメリットも存在する。

ただし、これは前節（IT ベンダーにもたらすメリット）で課題となった Ruby、Rails 開発者の増加や統一した仕様の情報の確立、作業の標準化による開発の継続性、ビジネス分野で活用可能なライブラリ整備などが解決されることを前提としている。しかしながら、ベンダー側からの課題解決だけを待っているだけでは、既に LAMP の技術がオープンソースだといえども米国の大手 IT 企業の支援によって進むと同時に、その技術力・開発力の優位性も維持されている中で、新たなロックインを生み出す可能性があり、特にクラウドコンピューティングの進展はそれを加速化させる<sup>23</sup>。

---

<sup>22</sup> OS である Linux、Web サーバである Apache HTTP Server、データベースである MySQL、スクリプト言語である Perl、PHP、Python を総称したもの。

<sup>23</sup> OSS 導入に伴うサポートビジネスに関しては、早くから Linux を搭載したサーバを市場に投入した IBM や Hewlett Packard などのハードウェアベンダーそして前述のオラクルや Red Hat などによる OSS のサポートビジネスの供給が進んでいるが、また OSS によるトータルソリューションを掲げるベンチャー企業も現れてきている。代表的な企業が 2003 年に設立された SpikeSource で、Linux を含めた OSS を組み合わせたサービスを提供し、コンポーネント間の依存関係の解決や豊富なテストパターンによる信頼性の向上による付加価値によるソリューション=サポートビジネスを提供している。これらのサポートビジネスを支えているのが中国・インド系を中心としたシリコンバレーのエンジニア群であり、企業内とコミュニティを流動的に行き来する中で自らのキャリアをアップさせている。こ



これに対して Ruby はオープンソースであり、Ruby 自体の開発や利用は世界的に拡大しているが、コア開発技術やエンジニアの層などの面での優位性は日本に、そして中国地域に存在する。

そこで、OSS の積極的な採用によって IT ソリューション市場を先導してきた日本政府も情報処理推進機構（IPA）を中心に、Ruby アソシエーションとともに Ruby を国際標準化機構（ISO）へ登録する作業を行っており、Ruby の標準仕様書の作成も進めている。これは Ruby 自体のコア開発技術に焦点を当てたものであり、Ruby そして Rails による開発における仕様の確立、作業の標準化、そして人材の育成を、Ruby、Rails の積極的な採用によって進めていくことは、長期的にもコスト削減とベンダーロックインの解除、そして技術的な対米依存からの脱却につながる。

これは地方自治体についても同様であり、Web アプリケーション開発を中心に Ruby、Rails の需要が全国的に拡大しつつある中で、Ruby、Rails による調達を進めながら、コスト削減とベンダーロックインの解除、そして地域の情報サービス産業の市場拡大を進める可能性は存在する。

### 3-2 調達における課題と可能性

IT システムを構築するユーザーにとって、ビジネス分野でももちろんであるが、単なる導入コスト削減ではなく、ランニングコストも含めた全体最適化が課題である。その中にアウトソーシングやクラウドコンピューティング、そして企業内クラウドや自治体クラウドも位置づけられている。

地方自治体におけるオープンソース導入政策は、特に IPA・OSS センター設立（2006年）以降は自治体業務に OSS を導入する実証実験を行なうという形で積極的な支援政策を進められてきた。また、前述したように、長崎県は民間から CIO を招聘し、電子県庁システムの基盤技術にオープンソースを採用し、コスト削減とベンダーロックインの解除、さらに地域の IT 産業振興に取り組んだ。

一方、小規模の自治体では CIO（Chief Information Officer：組織において情報に関する資源を統括する最高責任者）を配置できない自治体が多く、オープンソース採用に関しては調達側のオープンソースに関する知識、要件定義や仕様書策定のための時間と労力は大きな負担となり、また後年度のサポートのための費用、初期費用も含めたトータルコストの換算によって逆に高コストとなってしまう場合もある<sup>24</sup>。

---

のように米国では開発系の企業から大手 IT ベンダー、そして中小のベンチャー企業に至るまで各段階において OSS の技術的开发やサポートビジネスが進んでおり、そして重要な点は OSS の普及とともにこれらの企業のビジネスが世界的にも市場を拡大する可能性があるということである。

<sup>24</sup> Linux サーバを中心とした LAMP の導入はサーバ導入に伴う構築サービスを中心としたシステムインテグレーションのコストを含めても初期の導入費用コストは低下させること

IPA が 2007 年度と 2008 年度に「オープンソフトウェア利用促進事業」として地方自治体における情報システム基盤の現状と方向性の調査を行っているが、調査結果の分析を IPA と協力して行った(株)三菱総合研究所 情報技術研究センター 主任研究員 谷田部智之氏によると、

「OSS を採用するメリットとして、特定の事業者や製品からの脱却をあげる団体が 80%を超え、中長期的な費用の削減、初期費用の削減を期待する回答が多かったことのほか、システム間連携や最適化、地元ベンダーや中小ベンダーの参入機会の増加をあげる例も多かった。また、デメリットとしては、調達仕様書を作成するスキルおよびノウハウが不足しているとした回答が 3 分の 2 を占めたこと、対応できるベンダーが少ないという回答が多かったことを示した。IT ガバナンスレベルが高い自治体ほど、オープン化レベルも高いが、その一方で、IT ガバナンスが同等の自治体でもオープン化への取り組みには大きな差がある。」<sup>25</sup>としている。

「システム間連携や最適化、地元ベンダーや中小ベンダーの参入機会の増加」が中長期的な費用の削減につながると思われるが、そのためにはベンダー側だけでなくユーザー側にも「調達仕様書を作成するスキルおよびノウハウ」が求められる。Ruby、Rails を含めた OSS を採用するメリットをユーザー側が発揮するためには、ユーザー側でも

ができるが、導入後のサポート費用がユーザーにとって大きな負担になる場合がある。例えば 2005 年度から「履修科目申請システム」の運用を開始した早稲田大学では初期導入コストを抑えるためにオープンソース・ソフトウェア（OS に MIRACLE LINUX、Web サーバに Apache、データベースソフトに PostgreSQL、開発環境に PHP）を採用した。その結果初期導入費用に関しては商用ソフトを採用した場合の試算（958 万 8000 円）に比べて低く抑えることができた。しかしながら、サポート費用に関しては年間 1067 万円となり、これに比べて商用ソフトによる試算が年間 197 万 7880 円であり、このため既に 2 年目以降から初期費用とサポート費用の合計で商用ソフトによるコストを上回るようになってしまった。

**オープンソース・ソフトウェア  
サポートサービス費用**

PostgreSQL	3,000,000
PHP	3,360,000
Apache	3,360,000
MIRACLE LINUX	950,000

<b>サポート費用合計/年</b>	<b>10,670,000</b>
-------------------	-------------------

<b>初期費用</b>	<b>1,140,000</b>
-------------	------------------

**商用製品ベンダーによる  
サポートサービス費用**

Oracle Database	} 604,200	1,373,680
.NET Framework		
IIS		
Windows Server 2003		

<b>サポート費用合計/年</b>	<b>1,977,880</b>
-------------------	------------------

<b>初期費用</b>	<b>9,588,000</b>
-------------	------------------

<sup>25</sup> 三菱総合研究所 情報技術研究センター 主任研究員 谷田部智之氏談。  
<http://journal.mycom.co.jp/articles/2009/05/29/ossaj/index.html>

この課題の克服が求められる。これを克服するのが CIO なのであるが、現実には CIO を配置できない自治体がほとんどである。

一方、CIO を配置できない小規模な自治体、あるいはビジネス分野にとって ITC (IT コーディネータ) の活用も有効であると考えられるが、調達仕様書を作成するスキルや業務ノウハウを保有している ITC も少数であるのが現実である。

そこで、CIO、ITC といった役職、資格で考えるのではなく、オープンソース、そして Ruby、Rails の導入に必要な技術力・企画力、調達仕様書を作成するスキルや業務ノウハウを有する人材が求められる。

また、Web アプリケーション開発の際には、開発対象を多数の小さな機能に分割し、反復 (イテレーション) 開発によって機能を開発していくアジャイル開発の手法が適していると言われている。そして Rails は Web を素早く、見栄えよく作れる アプリケーションフレームワークであり、Ruby は読みやすく、変化を受け入れる言語であるために、アジャイル開発に最も適していると評価されている。

### 3-3 クラウドコンピューティングと Ruby の可能性

地方自治体にとって、財政事情が厳しくなっている中で、クラウドコンピューティング化が進むことは、情報システムの調達をより最適に行える可能性を示している。また、LAMP を中心にクラウドコンピューティングのシステム自体にも OSS が採用される可能性も増大している。

一方、中国経済連合会提言書が指摘するとおり、「クラウドコンピューティングの時代においては、大規模なスーパーコンピュータを有する数ヶ所のデータセンタにより全国がカバーされ、そこから必要なだけアプリケーションが利用される状況が想定できる。」「今以上にソフト開発も ICT ビジネスも、さらにデータセンタさえも一極集中になる可能性すらある。」<sup>26</sup> クラウドコンピューティングが進むことは、IT 産業の東京一極集中がさらに強まり、地域・中国地域の IT 産業にとってはますます厳しい事態が想定される。

しかしながら、これも中国経済連合会提言書でも提言されているとおり、「グーグルのようなサービスの場合はデータセンタの所在地はあまり問題とならないが、企業の基幹業務を主たる目的とする場合には、データセンタは地域内にあることが心理面でも運用面でも望ましい」。また、「中国地域の企業・自治体・大学等が共同で利用する大規模な省エネデータセンタを電力事情 (送電ロス等)、ユーザの利便性、交通事情などに配慮しつつ、機能の一部を分散することを含め、適切な地域に構築することが考えられる」<sup>27</sup>。

<sup>26</sup>中国経済連合会『提言 中国地方の新たな情報化戦略について』2頁より。

<sup>27</sup> 同上書、8～9頁より。

このようなデータセンタ、情報基盤の中国地域での分散化に合わせて、クラウドコンピューティング、Web 上でのアクセスと処理の大規模化への対応が進められている Ruby、Rails によるシステム構築を進めていく可能性も存在する。前節の Ruby、Rails のビジネス仕様の標準化と合わせて、クラウドコンピューティングに対応したシステム構築を進めていくことは、Ruby、Rails の市場拡大とともに、中国地域の IT 産業の市場拡大をもたらす可能性がある。

## 第4章 Ruby 普及に向けた今後の取組み（提言）

### 4-1 Ruby、Rails のビジネス標準仕様に向けての取組み

独立行政法人情報処理推進機構（IPA）の支援の下、Ruby アソシエーションを中心に進められている Ruby の標準化の取組みは、オープンソースという誰もが開発に参加できる任意団体のコミュニティで作られたことを考慮すると、画期的なことであり、教育機関での教育教材としての採用や行政分野でのシステム調達に大いに期待できるものである。Ruby が日本国内を中心に開発されていることから、世界に通用する国産の IT 技術として経済産業省を中心に強力な支援を受けている。Ruby を核にした技術的優位性を早期に創り出し、国策として Ruby に関連する研究開発やプロジェクトを推進するタイミングである。

島根県では、2009 年度に Ruby 関連の研究開発費の補助を、500 万円を上限に 1/2 補助をしている。帳票エディターの開発や分散開発の研究、PHP から Ruby への変換ツールなどの取組みが行なわれており、着実に技術的優位性を創り出している。

一方、システム構築の面ではフレームワークとして、米国中心の開発コミュニティがサポートしている Ruby on Rails が圧倒的な強みを持っており、開発言語 Ruby においては標準的なフレームワークとなっている。システム構築においては、Web アプリの構造を支援するフレームワークとして、生産性の高い Rails をベースに考えることが望ましい。しかし、システム構築においては、Rails が万能なものではない。処理形態や業務の特性等から、さらなる工夫やツールが整備されると、開発の容易性、高品質、高メンテナンス性等の面で、一層の生産性向上に結びつく。これらの工夫やツールは、開発する対象システムの業務特性に依存するし、プロジェクトの規模や開発メンバーの習熟度などにも依存する部分であり、様々なものが考えられる。

Ruby・Rails を使った開発は、ホームページに簡単な業務を追加するようなネットビジネス分野、例えば会員登録、買い物、支払い、ブログ、情報提供など、短期開発を武器に普及してきた。システム構築の経験の少ない人が、これまで高度とされていた Web アプリの構造的な開発技術を易しくマスターできるようにしたものである。複雑なビジネスロジックが存在する多くの業務システムが Ruby・Rails で開発することにより、簡単になるものではない。複雑なビジネスロジックを判り易く画面に表示したり、帳票を見やすくするのが、システム設計である。複雑なチェックや計算をプログラムで行なうことにより、画面を操作する人が複雑な業務を易しく操作できるようになる。

このような業務システムを開発するための工夫やツールが、Ruby・Rails 開発をこれから始めようとするプロジェクトに提供されることにより、オープンソース開発や新しい技術に対する不安を取り除き、スムーズな開発を推進することに繋がる。その事例となるものがいくつか出来上がってきている。2007 年度 IPA の公募事業（松江市の医療・介護高額合算システム）では、自治体基幹業務開発の開発方法やその中で利用されたラ

イブラリが公表されている。2009年度経済産業省の地域ITベンダー間の連携強化を図る公募事業(Ruby業務システム開発基盤整備・開発技術向上プロジェクト)では、Rubyによる業務システム開発の技術向上を図るため、島根と広島ITベンダー連携により遠隔地共同開発基盤としてのSaaS基盤を構築し、Rubyによる業務アプリケーション開発プラットフォームを整備する。Rubyの強みである開発生産性をさらに向上させ、地域のIT供給力強化を図る取組みが行なわれている。また、島根県庁の情報システムでは、各部局のシステムをRuby化するための先行プロジェクトが実施され、課題の整理が進んでいる。

これらの事例を開発業務の特性やプロジェクトの規模、開発方式に応じた複数パターン化が必要とされている。中国地域で利活用できるRuby・Railsによる業務システム開発の基盤整備が推進されなければならない。

#### 4-2 ベンダー側の力量強化の取り組み

ITベンダーにとって、ビジネスでオープンソースを取り扱うことは、初めてというケースが殆どである。LinuxをOSとして活用するケースは増えているが少数である。それらも、大手ITベンダーのサポートの下で利用するケースが多い。Linuxは2000年からのe-JAPAN政策の下で、政府・行政機関が率先して活用したことにより、大手ITベンダーがサポートを強化した経緯がある。Ruby、RailsはLinuxの場合と違い、2006年に米国から逆輸入という形態で国内に普及して来た。それもネットビジネスを扱う分野が中心であり、システム開発の大部分を占める業務システムの分野には、極一部での取り組みにとどまっている。

従って、ビジネス的なサポートは一部の企業で取り組んでいるに過ぎなく、ビジネスとして確立された状態ではない。ITベンダーがRuby・Railsの開発を行なう場合、オープンソースの壁を乗り越えなければならない状況であることに変わりはない。書籍やネット上に公開された情報を調べ、多くのWebアプリ開発に関連するオープンソースを組合せて、問題なく動作することを自分達で検証する必要がある。Rubyの機能アップこそ年に1度であるが、Railsは3カ月に1度のペースで更新されている。その他のWebアプリに関するツールやミドルウェアがどのタイミングで機能アップされているか、正確に把握できている状態ではない。ネット上に、動作実績の情報が公開されているが、それは調査を行なった人の環境での実績であり、全ての機能が確認されている訳ではない。

このような状況で、ITベンダーがビジネスとしてRuby・Railsを活用していくためには、以下のパターンが想定される。

① 自社内（プロジェクト内）にオープンソースに精通した人材がいる

この場合は、その人を中心に開発環境を整備することが可能である。技術的な支援やトラブル時のネット情報による解決も可能である。開発メンバーに対して、**Ruby、Rails** の研修を行ない、**Ruby・Rails** 経験者の技術支援により開発が可能となる。

但し、業務システム開発におけるプロジェクト管理はオープンソースによる開発であろうと、従来の技術による開発であろうと変わるものではない。豊富なプロジェクト開発の経験を有するプロジェクトマネージャが必要であることに変わりはない。

② 自社内（プロジェクト内）にオープンソースに精通した人材がいない、かつオブジェクト指向言語の経験あり

**Java** のようなオブジェクト指向言語を経験した人にとって、**Ruby** を修得することは非常に容易である。2～3日の研修でマスターできる。**Rails** については、**Web** アプリの仕組みを理解する必要があり、**Java** 経験者であれば同様なシステム開発を多く行なっているため、これも短期間で修得できる。

開発環境の整備をスケジュール通りに進めるためには、**Ruby・Rails** 経験者の技術支援を受けることが望ましい。あるいは、社内で研究開発的な小規模な開発を **Ruby・Rails** で行ない、経験を積ませる方法もある。開発環境をクラウド型で提供するサービス（**PaaS**）も始められているので活用する方法もある。

③ 自社内（プロジェクト内）にオープンソースに精通した人材がいない、かつオブジェクト指向言語の経験なし

手続き型言語である **COBOL** や **Visual Basic** のエンジニア中心の場合、業務システムの開発方式も **Web** アプリの開発方式を取り入れようとしているため、積極的に **Ruby・Rails** の開発に取り組むことが望ましい。オブジェクト指向の特徴であるデータに着目した設計や部品化を推進する設計は、手続き型の設計を行ってきたエンジニアには困難さを伴うので、従来からの業務フローに着目した設計方法で支障なくシステム構築可能である。2007年度の松江市の医療・介護の高額合算システムのプロジェクトで実証されている。

**Ruby、Rails** の研修プログラムが整備され、地域 **IT** ベンダーによる **Ruby・Rails** の開発供給力アップを図り、技術的な先行有利を享受できるような取組みを推進する必要がある。

併せて、更に **Ruby** 開発のスピード・生産性を向上させ、競争力を強化し、その優位性を確保していくためには、中国地域の広域 **IT** ベンダー連携の下で、以下の取組みを今後進めていくことが重要である。

#### ○Ruby 開発基盤（プラットフォーム）構築とライブラリーの拡充

・生産性に優れ、多機能かつライブラリが充実した「Ruby 開発基盤（プラットフォーム）」を構築するとともに、それを使いこなすための人材の育成、作業標準化を進める。

#### ○Ruby の標準化

・Ruby の国際標準化を進めていくことにより、開発業務を標準化させ、大型案件の共同開発など様々な案件に対して、使いやすい環境を整えていく。

#### ○IT 専門家（IT コンサルタントや IT コーディネータ等）の活用

・お客様(発注者)の負担を軽減しつつ、開発を進めていくにあたっては、IT コンサルタントなどの専門家を介して仕様調整等を行うことが、1つの有効な手法であることから、これが行いやすい環境を整えていく。

あわせて、Ruby の信頼性・安心感を高め、更に普及・拡大していくためには、Ruby のサポート体制(セキュリティ、開発手引き書の整備等含む)を充実させていく必要がある。

### 4-3 ユーザー側の力量強化の取り組み

自治体のクラウドコンピューティングが進むことが予測され、オープンソースを活用する IT 産業の動向に大きな影響を与えることになる。この自治体クラウド化を特定の企業へのシステムの集中化として進めるのではなく、各自治体の具体的な住民サービスの提供と合わせた情報基盤の中国地域での分散化に対応して、Web 上でのアクセスと処理を Ruby、Rails によって行う可能性が存在する。

一方、Ruby、Rails を含めた OSS を採用するメリットをユーザー側が発揮するためには、調達仕様書を作成するスキルや業務ノウハウを保有している人材が求められる。これは CIO、ITC といった役職で考えるのではなく、オープンソース、そして Ruby、Rails の導入に必要な技術力・企画力、調達仕様書を作成するスキルや業務ノウハウを有する人材である。既存の ITC を含めて、調達側にこれらの能力を養成する教育が早急に求められる。

また、各ユーザーのレベルによって

#### ① 開発部門のあるユーザー

→ Ruby の生産性の高さを強調する。

IT ベンダー向け研修に参加を促す。



② 開発部門なし、システム企画・運用部門あり

→ 中国地域で推進する地域資源であることを理解してもらう。

Ruby が IT ユーザーにもたらすメリットを強調する（講演会等）

③ 開発部門なし、システム企画・運用部門なし

→ Ruby、Rails を推進・普及することは難しいが、

中国地域で推進する地域資源であることの理解は可能である。

といった対応が考えられる。

#### 4-4 Ruby 普及と中国地域競争力に向けた今後の取組み

中国経済産業局では、地域 IT ベンダー連携による地域の IT 供給力強化に重点を置き、地域の IT ベンダーとユーザーの連携を加速させていく「地域イノベーションパートナーシップ構想」をもって、地域 IT ベンダー連携による IT 供給力強化支援を行ってきた。

中でも、Ruby に関連した IT ベンダー連携による供給力強化を図るプロジェクトとして、島根県の IT ベンダーと広島県の IT ベンダーの連携による「Ruby 開発共通基盤」の構築プロジェクトや、Ruby によるクラウドコンピューティングでの業務システム開発プロジェクトなど、平成 21 年度地域情報化基盤整備補助事業の中で採択支援を行ってきたところである。

本研究会での議論の中で、Ruby が Web アプリケーション開発において生産性の高さから IT ベンダーにとって有利な武器となること、クラウドコンピューティングの時代に入り Ruby の活用場面が拡大することが予想され、またユーザーにとってもコスト削減効果や細かいニーズを取り込んだシステム開発や拡張性の高さなどのメリットをもたらすことが見いだされた。

しかしながら、現時点において Ruby はこれまで主に Web アプリケーション開発を中心に使われている状況であり、業務システム開発での実績はまだ少ない状況から、プログラミングの生産性の高さを持ちながらも、未だ爆発的な普及にいたっていない。

IT 産業界では、近年クラウドコンピューティングが主流になっていこうとしている中、地域 IT ベンダーは、大手ベンダーのクラウドソリューションとの差別化をいかに図っていきけるか、首都圏の大手ベンダーからの下請け脱却をいかにして図っていくか、クラウドコンピューティングの時代を迎え、IT ユーザーに近い距離にある地域 IT ベン

ダーこそビジネスチャンスの可能性を合わせもっている。

また、コーディング作業等の標準化が進めば大型案件での共同開発手法が確立できること、生産性を向上させる Ruby 開発共通基盤の構築と業務系での開発機能などの拡充、それを使いこなす Ruby 人材の育成と継続的供給体制の仕組み作り、それによって中国地域に Ruby 先進企業及び技術者の集積を図っていくことができれば、クラウドコンピューティング時代での IT 先端地域として中国地域の IT 産業の競争力強化につながる可能性を秘めている。

こうした観点から、この Ruby ビジネス活用研究会で見いだされた Ruby の優位性やユーザー企業にもたらすメリットの両面から普及を加速させ、IT 産業の地域競争力強化を図る取組みとして引き続き Ruby を核とした IT ベンダー連携や IT ベンダーとユーザーの連携を加速させ、需要の創出も併せて行っていくことが重要であり、産学官の連携の下中国地域での Ruby 人材の育成と集積を図るとともに、中国地域が Ruby 開発供給拠点を目指した施策の展開が求められる。需要の創出の面においては、行政が率先して導入することなどが望まれる。

産学官が一体となつての取り組みによって相乗効果を生みだし、中国地域が Ruby 開発供給拠点として将来に向かってさらに飛躍発展していくこと、きたるクラウドコンピューティング時代において、開発力、供給力、競争力をもって中国地域の IT ベンダーが優位なビジネスを展開していけるモデルを構築し、国内外の IT 利活用のベストプラクティスを中国地域から発信していくことが求められる。