

Ruby業務システム開発 in ホットスポット しまね

2010年8月29日

(株)マツケイ

CEO 吉岡 宏

1. 自己紹介(システム開発の経験)
2. ホットスポットしまねの動き(業務システム開発の実績)
3. 業務システムの開発パターン化(Ruby & Rails)
4. COBOLerの復活と若手エンジニアの育成
5. Rubyの広がりと期待

1. 自己紹介(システム開発の経験)



PROJECT-R
Ruby X TECHNO PROJECT

1976 (昭和51年) I T 業界入り

(国産メインフレームを通産省の支援の下、共同開発していた時代)

メインフレーム用ミドルウェアの開発(5年)

オンラインデータベースシステムの為のフレームワーク

テレビ局向け営業放送システム(メインフレーム、オフコン)

金融)勘定系オンライン

※各メーカーシステムのコンバージョン作業 ⇒ 多くのシステム構造を経験

1994~ ⇒ オープン化、C/Sシステム化

Access、VB 各種小規模システムの開発

1998~ ⇒ C/Sシステムの3階層開発 (VBとCOBOL)

自治体向け介護保険システム

2007~ ⇒ Ruby・Railsを用いたシステム開発

2010(昭和85年) 約35年間経過 Sier、請負契約中心

1. 自己紹介(システム開発の経験)
2. **ホットスポットしまねの動き**
(業務システム開発の実績)
3. 業務システムの開発パターン化(Ruby & Rails)
4. COBOLerの復活と若手エンジニアの育成
5. Rubyの広がりと期待

ホットスポットしまね

まつもとゆきひろ氏
“Ruby”

マスコミにも良く取り上げられる
市民レベル(半分くらい?)
“Ruby”=コンピュータ言語
&まつもと氏

しまね

産業振興
雇用拡大

松浦 松江市長

溝口 島根県知事

しまねソフト産業
ビジネス研究会
40社加盟、約1000名
代表 吉岡宏



2007年 IPA(情報処理推進機構)実証事業

オープンソフトウェアセンターがRuby/Railsを基幹業務に適用する実証事業を実施。

⇒ 松江市殿)医療・介護高額合算システム

(日本初:バッチ処理を含む基幹業務全体)

約50人月 (株)テクノプロジェクト、(株)マツケイ
(株)ネットワーク応用通信研究所

※業務システム:企業内・自治体内の基幹業務システム

販売管理、財務会計、顧客管理、税系、福祉系など

支給申請入力

窓口業務担当ユーザ

2008/01/22 16:13:25

ログアウト

被保険者状況照会

自己負担証明書申請入力

登録 修正 削除

状態:

受付 > 合算済 > 確認済 > 審査済 > 発行済 > 依頼中 > 支払済

制度: 2:後期 年度: 2008 被保険者番号: 3000003800 個人番号: 1000003340 被保険者番号検索 受付番号: 0000000006

被保険者名: 高額太郎(コカ^{タカ}タロウ) 区分: 合算 世帯集約番号: 3003120000 給付制限状況: 所得区分: 70未満:-- 70以上:現
 生年月日: 昭和5年4月25日 年齢: 77歳 性別: 男 住所: 松江市松江町217-1

主	個人番号	氏名 (氏名カナ)	生年月日	年齢	性別	制度	保険者	被保険者番号	資格加入期間情報			支給申請情報	
									取得	喪失	得喪事由	受付番号	申請日
	1000003340	高額太郎(コカ ^{タカ} タロウ)	S05.04.25	77歳	男	後期	後期島根県	3000003800	09/03/26		転入による資格取	0000000006	09/11/15
						介護	介護松江市	4000008200	09/03/26		転入による資格取		
						後期	鳥取県後期	3900000200					
						介護	境港市介護	4900000300					
	1000003350	高額花子(コカ ^{タカ} ハナコ)	S07.05.18	75歳	女	後期	後期島根県	3000003900	09/03/26		転入による資格取	0000000033	09/11/15
						介護	介護松江市	4000008300	09/03/26		転入による資格取		

自己負担額証明書登録:

登録

修正

削除

申請日: 2009/11/15 本人電話番号: 0852-12-3515

申請者: 郵便番号: 690-0041 住所: 松江市松江町217-1

氏名: 高額太郎 電話番号: 0852-12-3515

送付先: 1:本人 その他の送付先: 郵便番号: 住所:

氏名: 電話番号:

支払方法: 1:口座振替

口座情報: 金融機関: 0001 松手銀行 店舗: 001 本店 金融機関検索

口座種別: 1:普通 口座名義人: 高額太郎 口座名義人カナ: コカ^{タカ}タロウ 口座番号: 0000182

Railsでも、C/Sシステムと同様な画面IF (コンボボックス、オプションボタン、サブウィンドウ等)

窓口に市民が来た時の対応 ⇒ 1つの画面で完結 (画面遷移の少ない処理)

(上段: 申請者情報、中段: 受給履歴、下段: 申請内容)



2008年 小規模システム中心に開発事例の拡大

住民・コンシューマ向けシステム（ホームページ）
企業内業務システム
企業内情報系システム

5人月程度の小規模システム

ウォーターフォール&アジャイル

ベテランSE向け開発手法

（業務ノウハウ精通者 ⇔ COBOLer）

※島根県による「Ruby教育コース」開催



2009年 Ruby開発推進(半額補助 & 研究開発)

公募事業 (Ruby開発に1/2補助)

Ruby研究開発 (1/2補助)

島根県) 既存の部局システムのRuby化 (Web化)

民間での導入事例 (販売管理)

※島根県による「Ruby教育コース」開催

※RubyWorld Conference 2009 松江市で開催



2010年 新規業務はRubyが基本 既存システムのWeb化もRubyで！

自治体システムでの調達拡大

- ・松江市の新行政情報システム(住記・税・福祉)
(Ruby・地域貢献に2割の評価点)

- ・島根県の部局システム(全13業務Ruby等で調達)

民間へのシステム提案にRuby活用

パッケージ開発にRuby活用

※島根県による「Ruby教育コース」開催

※RubyWorld Conference 2010 松江市で開催

1. 自己紹介(システム開発の経験)
2. ホットスポットしまねの動き(業務システム開発の実績)
- 3. 業務システムの開発パターン化
(Ruby & Rails)**
4. COBOLerの復活と若手エンジニアの育成
5. Rubyの広がりと期待

業務システム開発の要件



大量データ
レスポンス

- ①アクティブレコード(DBアクセスがブラックボックス)
⇒ **見える化(SQLを直書き)**
⇒ 性能アップ & 生産性低下(ステップ数増加)

メンテナンス性

業務システム
10・20年稼動する
プログラマが何世代も交替

- ②多様な言語仕様(自由度が高い)
⇒ **コーディング規約とソースレビュー**
- ③アルファベット表記(Rails全体での日本語対応:非)
⇒ **一定条件の下で日本語項目名称使用
(ソース可読性向上)**

サポート

- ④安心できるサポート ⇒ ディストリビュータ
⇒ ネットワーク応用通信研究所
基本はリテラシー向上(豊富なネット情報)

バッチ処理

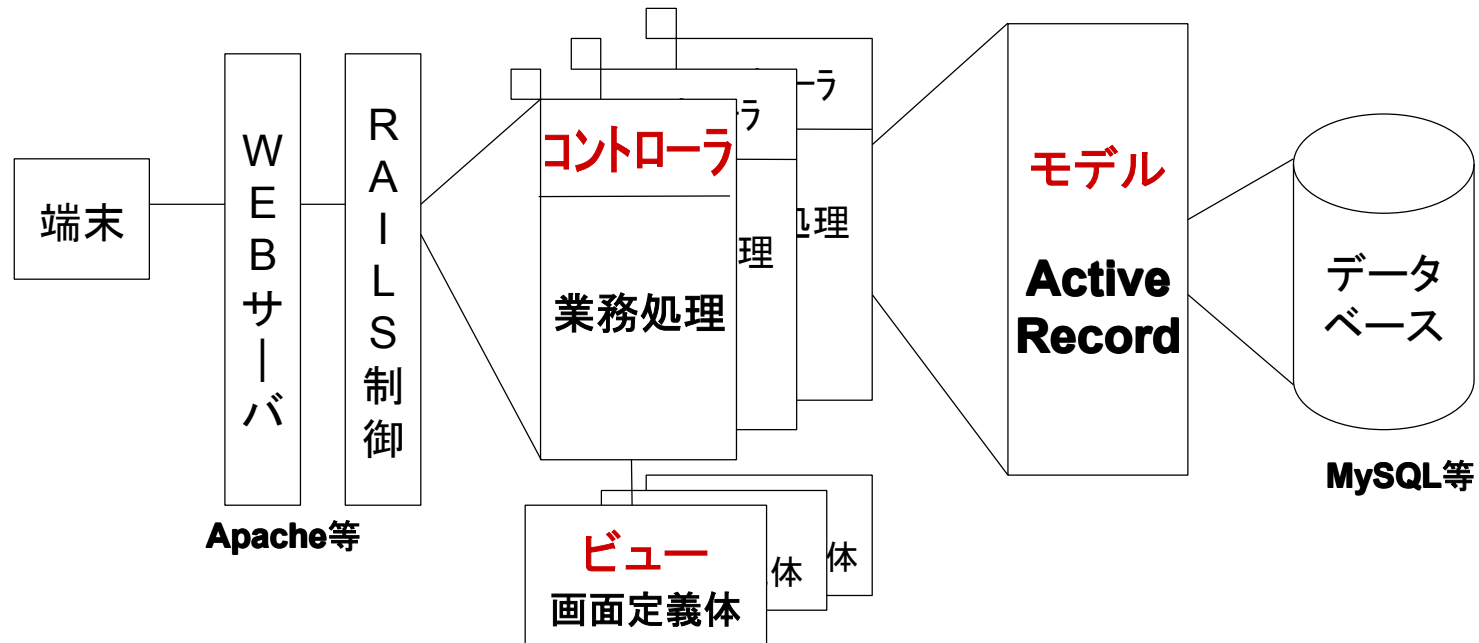
- ⑤バッチ処理 ⇒ バッチRails(DBアクセス)
⇒ **開発要員の共通化、部品の共通化**

開発方式

- ⑥オブジェクト指向言語=クラス設計から・部品化
⇒ **構造化設計(業務フロー、画面遷移図、設計書、テスト仕様書)**
- ⑦Railsオンライン構造=MVC(モデル・ビュー・コントローラ)
⇒ **C(コントローラ)に業務ロジックを集約する**
=》従来(手続き型)の開発方法と同じ=ロジックは1箇所
Railsテストツールの適用率低下

Webアプリ(Railsシステム構造)

Ruby on Rails (機能分割: モデル・ビュー・コントローラ)



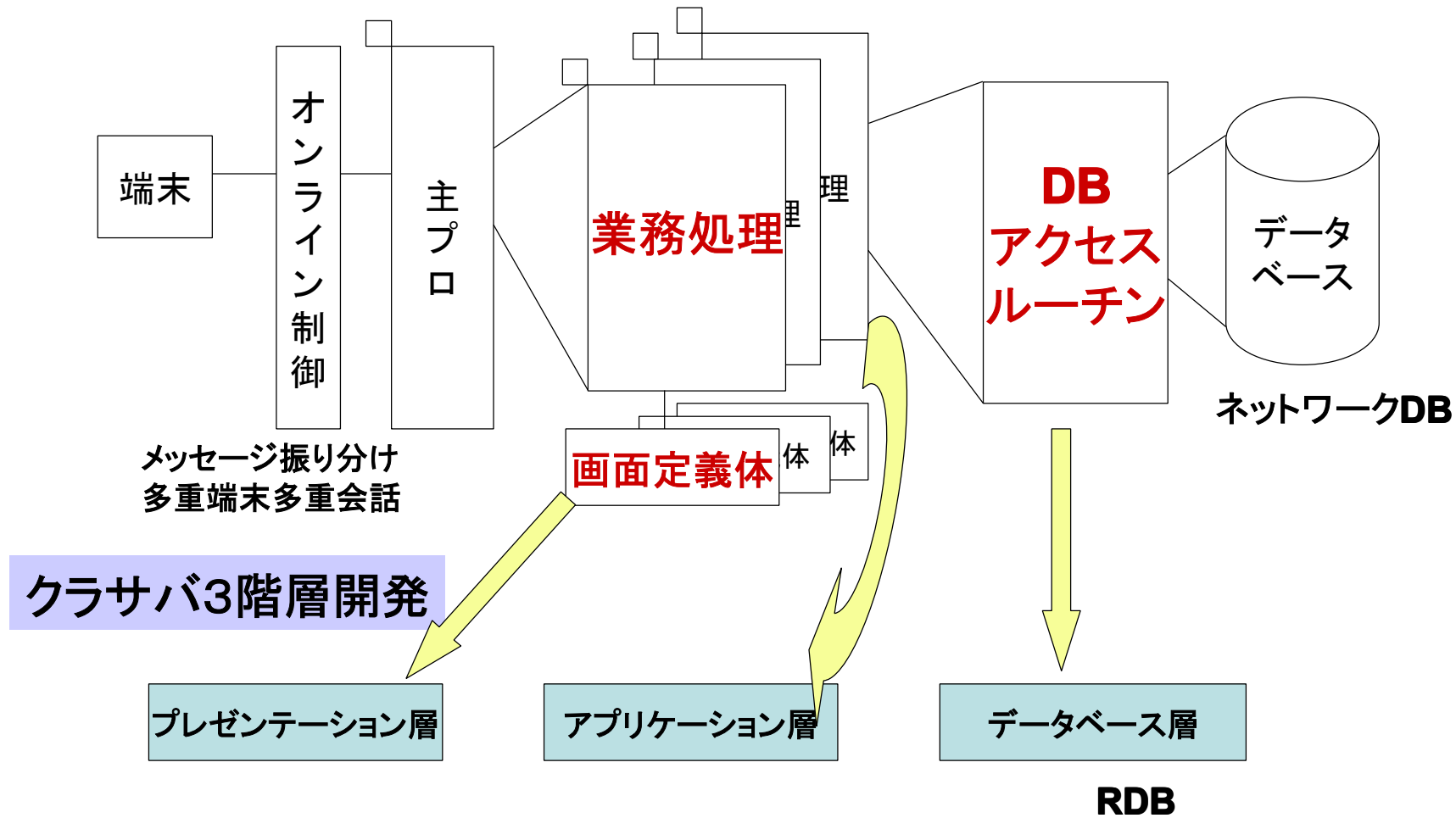
※Ruby on Rails

M:モデル(DBマクロ) 検索・挿入・更新・削除のSQL

V:ビュー(画面定義) DBの追加・更新・削除の画面HTML

C:コントローラ(オンライン制御・業務処理)

メインフレームのオンライン構造



業務処理に集中して開発できる環境

①プロジェクト管理・開発ツール

Redmine : 案件管理・ToDo

Subversion : ソース履歴管理 (Redmineと連携)

NetBeans : 開発ツール

②厳格なコーディング規約

ソースレビュー : 第三者の目に触れることが重要

③納品物と開発方式

(1) 従来型の納品物が要求される場合

⇒ **ウォーターフォール開発** (設計書・テスト仕様/成績書)

※ユーザ側の仕様決定プロセスや契約形態にも依存する

(2) 従来型の納品物が要求されない場合

⇒ **アジャイル開発**

初期 : 簡易な設計資料 ⇒ 終了時 : メンテナンス資料の整備
構成一覧・プログラム概要 (重要ロジックなど)

④開発チーム

短期開発(1年以内): **最初から全員投入**

⇒ チームワーク・コミュニケーション

PGの役割: SE補助作業・業務習得・先行開発

⑤データベース

日本語項目名称:活用 ⇒ ソースの可読性向上

メンテナンス性、専門用語

DBの複合キー ⇒ 設計の柔軟性(副次キー)

例) 年+月+種別+社員番号 : 給与データ

複合キー用プラグイン、主キー: **id**(一意な数値)

SQL直書き ⇒ レスポンス・スループット向上

特にSelect処理とバッチの繰り返し処理

更新系は**ActiveRecord(Validation活用)**

⑥ **テスト駆動開発** 抵抗を感じるエンジニアが多い
単体テスト(モデル) 活用: DBアクセス・拡張ロジック
機能テスト(コントローラ) 活用: 画面単位のテスト

⑦ 帳票ツール

オンライン・バッチ共通環境 & 共通要員 ⇒ コスト・品質に有利

Java用: JasperReports (iReport)・・・Rubyより利用

※次ページ参照

Ruby用: **ThinReports** (帳票エディタ、ビューア: PxDoc)

オープンソースで(株)マツケイ作成(2010.9リリース予定)

将来構想: ビューア開発・PDF化機能

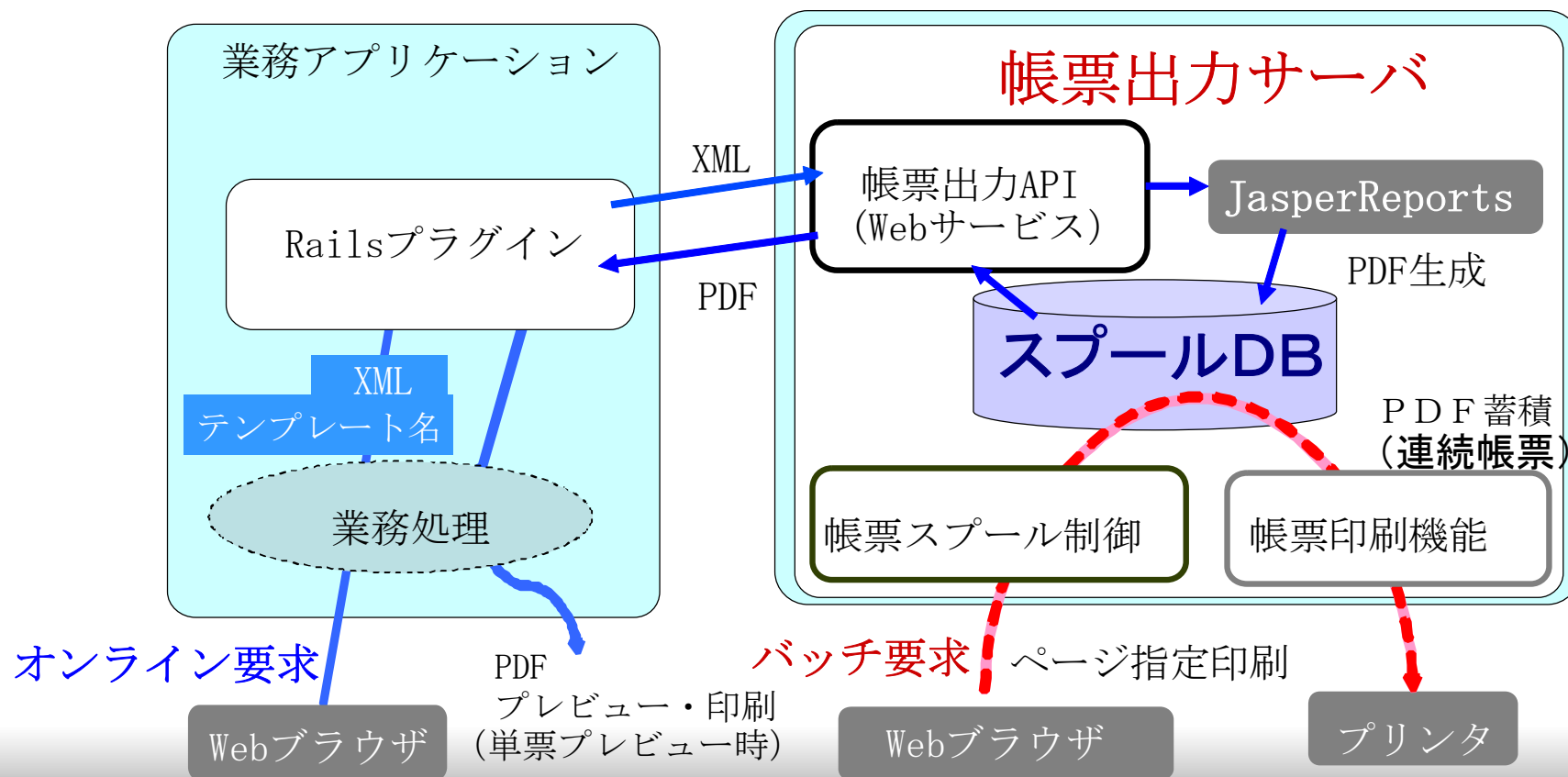
ベンダーのC/Sシステム用帳票ツール(リストクリエータ、・・・)

スプール機能 ⇒ 帳票DB: 夜間バッチ + 翌日印刷
プレビュー、再印刷、ページ指定印刷

スプール機能(帳票出力環境)

スプール機能

- ・プレビュー
- ・ページ指定印刷
- ・再印刷
- ・印刷中キャンセル
- ・検索
- ・削除



1. 自己紹介(システム開発の経験)
2. ホットスポットしまねの動き(業務システム開発の実績)
3. 業務システムの開発パターン化(Ruby & Rails)
- 4. COBOLerの復活と
若手エンジニアの育成**
5. Rubyの広がりと期待

ベテランSE(COBOLer)がWebシステムの開発に参加できる

- ①SQL記述(判り易さ)
- ②コーディング規約
- ③日本語のDB項目名称(設計書とソースの見易さ)
- ④ウォーターフォール開発(設計工程あり)
- ⑤バッチ処理も同一言語で開発
- ⑥構造化設計(業務ロジック重視、部品化は少量)
- ⑦従来のオンライン構造と同様(メインフレームと同じ設計)

COBOLer向けRuby/Rails教育コース by テクノプロジェクト

COBOLerの復活 ⇒ 設計重視



“業務ノウハウ”

業務知識に基づいたお客様とのコミュニケーション
特に、お客様の業務熟練者が現場から居なくなって来ている(2007年問題)
※Webアプリの技術より、業務ノウハウ(法令・業界ルールなど)が重要

“データベース設計”

複雑な業務も、スマートなテーブル設計により、シンプルなロジックになる
但し、**Rails**の規約にも合わせたテーブルやキーの設計が必要である
仕様変更や機能追加をある程度想定した作り
※大量データ処理: ディスクアクセスの削減が重要 ⇒ DBの直接アクセスを削減する。テキストファイルに出し、ソートやマッチングを活用。

“豊富な経験”

イレギュラー処理(赤伝・データの訂正処理)
入力ミスや処理ミスを想定した設計、大量データの処理方法
初期段階での設計が、そのシステムのライフサイクル全体に影響する



Webアプリ ⇒ オブジェクト指向言語 / 開発 ⇒ 若いエンジニア

経験豊富なベテランSEがWebシステムの現場復帰

“短期開発と業務ノウハウ”

開発期間が短く、業務知識を習得する余裕がない

“品質確保とテスト駆動開発”

顧客仕様を満たすために、テストツールに要件を記述する
⇒ 顧客が明に提示した要件を満足するシステム



“ベテランSEと若手エンジニアの融合”

⇒ **ベテランSE**が若手エンジニアに業務ノウハウを伝承する
業界の暗黙値、専門用語の理解不足、エラー時の処理など



Ruby/Railsの業務システム開発は若手エンジニアの育成の場

業務ノウハウに基づいた、品質の良いシステムの構築

1. 自己紹介(システム開発の経験)
2. ホットスポットしまねの動き(業務システム開発の実績)
3. 業務システムの開発パターン化(Ruby & Rails)
4. COBOLerの復活と若手エンジニアの育成
5. Rubyの広がりと期待

- ① **米国**で認められた製品の逆輸入
- ② 競争激化による**コストダウン**要求
有償ソフトから無償あるいは低価格ソフトへ
OSから言語まで、すべてのミドル製品が
オープンソースで揃う
- ③ **国の支援**（世界で通用する国産技術）
経済産業省（IPA）や総務省の支援
- ④ C/Sシステムを**Web化**するタイミング
クラウドサービスの時代
- ⑤ 中小ITベンダーが大手の下請けから**自立**
- ⑥ 国・地方自治体での**調達指針**
2006年総務省ガイドライン：オープンな標準製品

Rubyの広がりと期待



松江市

島根県

中国経済
産業局

福岡県

IPA

標準化

クラウド

国産技術



Ruby

A Programmer's Best Friend

政府調達



PROJECT-R

Ruby X TECHNO PROJECT

ありがとうございました