



LL(軽量言語)によるアジャイル・エンタープライズ開発

– Ruby on Rails –

2007.5.14

XML Consortium Week

Web2.0部会
メターデータ株式会社
大場寧子 ohba@metadata.co.jp

© XML Consortium

はじめに



挙手おねがいします

ご存知の方？

開発経験のある方？

企業で採用経験のある方？

© XML Consortium

自己紹介



- C++ 3年
- Java 7年 うち Servlet/JSP 5年
- Ruby on Rails 1年

現在、Ruby on Rails 2年生 (^o^)

© XML Consortium

Ruby on Railsとの出会い



Drecom Award on Rails (2006年)

- Web家計簿「小槌」を開発
- 大賞・審査員賞を受賞

<http://www.kozuchi.net/>



以後、RoR一色に

© XML Consortium

本日の構成



- Ruby on Rails について
- デモ
- 企業で採用する際のポイント

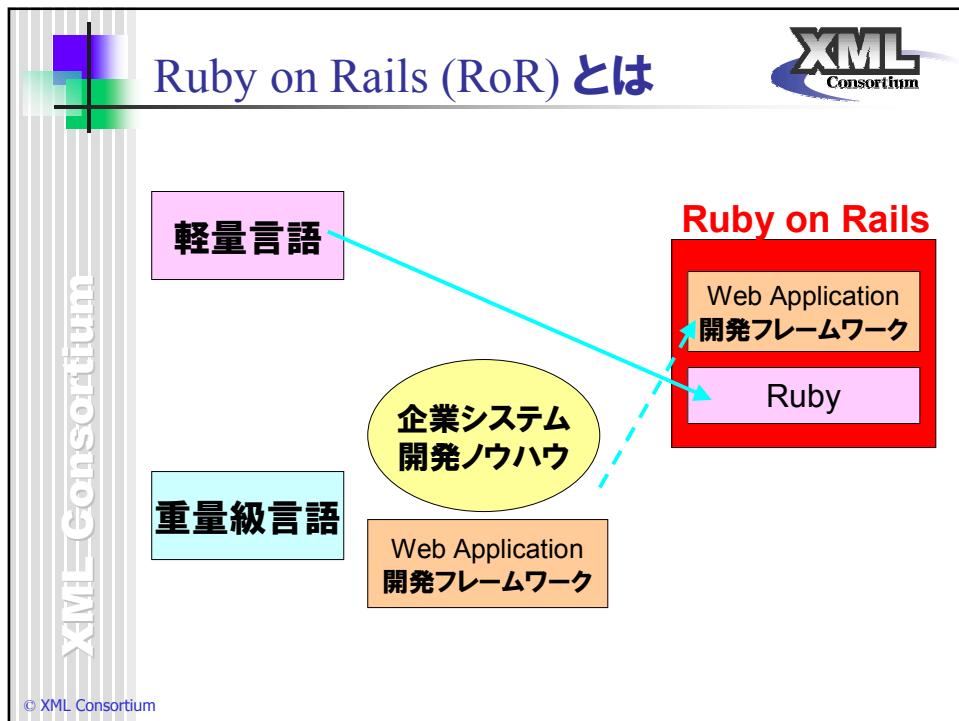
© XML Consortium

軽量言語台頭の2大背景



- ハードウェアの進化
 - 軽量言語は比較的処理速度が遅い
 - ハードウェアが安く、速く、大容量に
 - 軽量言語の実用性 Up !
- ニーズの変化
 - 短寿命化
 - 短納期、低成本、高品質

© XML Consortium



-
- RoRの特色** (Features of RoR)
- **Convention over Configuration**
(規約は設定に勝る)
 - **DRY**
Don't Repeat Yourself
(重複したコードを書かない)
 - **フルスタック**
 - **開発・テスト・本番の3種類の環境を用意**
- © XML Consortium

気に入っているところ



個人的に一番ラブな部分は

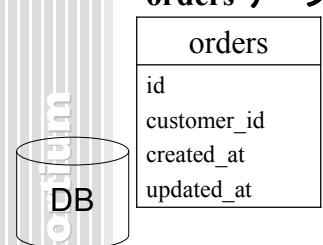
ActiveRecord (O/Rマッピング)

© XML Consortium

強力な ActiveRecord



orders テーブル



Orderクラス (order.rb)

```
class Order < ActiveRecord::Base  
end
```

RoRは自動的にカラムに対応するメソッドを用意してくれる。

```
id  
id =  
customer_id  
customer_id =  
....
```

- ・テーブル←→クラスの対応
- ・アクセスメソッド
- ・テーブル間のリレーション
- ・コールバック

© XML Consortium

強力な ActiveRecord

XML Consortium

最小限のコードでできる！

DB 設定ファイル不要！

orders テーブル

id	customer_id	created_at
1	1	2008-01-01 12:00:00

Orderクラス (order.rb)

```
class Order < ActiveRecord::Base
end
```

RoRは自動的にテーブルに対応するメソッドを用意してくれる。

id
id =
order_id
is_order_id
is_order_ids
....

• テーブル←→クラスの対応
• セフメソッド
• テーブル間のリレーション
• コールバック

© XML Consortium

RoRでAjax

XML Consortium

- prototype.js などのライブラリを同梱
- Ajax Helper が充実
- 典型的なAjax処理がとても簡単に書ける
 - イベントでHTMLの一部を書き換える
 - 特定フィールドの変化をタイマーをつかって監視
 - etc..
- JavaScriptを直接書かずにAjaxで作れる

ただし

- API Documentだけでは、どこまでHelperができるか把握しきれない面も
- JavaScript やライブラリの知識はやはりほしい

© XML Consortium



RoRでWeb Service



	REST	XML-RPC	SOAP
利用	簡単 HTTP接続 + XML解析	努力？ 標準ライブ ラリ	複雑？ 添付ライブ ラリ
提供	超簡単 XMLを返す 普通のWeb アプリ	少し努力？ ActiveWebService (RoR) を使ってサービス記述	

※青字は個人的印象です。？は直接体験なし

© XML Consortium

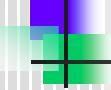


RoRとセキュリティ



- SQLインジェクション
 - ActiveRecord のプレースフォルダを利用
- クロスサイトスクリプティング(XSS)
 - 危険のある出力文字列を h メソッドでサニタイ
ジング
- クロスサイトスクリプティングフォージェリ
(CSRF)
 - security_extensions プラグインを利用

© XML Consortium



Ruby の力



なぜ Rails は Ruby で生まれたか？

Ruby は

- ものすごく柔軟である
- オブジェクト指向
- 簡潔に記述できる

© XML Consortium



デモ



詳細は当日のお楽しみで… ☺

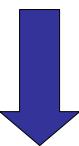
© XML Consortium



RoRを採用するメリット



- 高い生産性
- Web2.0技術が簡単に利用できる
- 変化に強いアプリケーション開発



アジャイル開発

ユーザーニーズを素早くシステムに反映

© XML Consortium

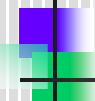


これまでの企業での利用経験



- 海外向けECサイト構築
- コミュニティサイト構築
- 医事会計ソフト構築
 - 1000以上の煩雑な個別ルール
 - オフショア開発とのコンビネーション
 - RoR では複雑なビジネスロジックがきれいに書ける印象

© XML Consortium



RoR採用に際しての課題



XML Consortium

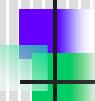
- パフォーマンス
- スケーラビリティ

現時点での代表的な運用環境は

- FastCGI + Lighttpd
- Apache + Mongrel

流動的な状況

© XML Consortium



JRuby

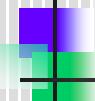


XML Consortium

- <http://www.jruby.org>
- JavaVMでRubyを実行するRuby実装
- Java で培われた堅固なステージング環境で Ruby を運用できる
- 課題となる実行速度も上がってきた
 - インタプリタモードのJRubyはRuby1.8の3~4倍遅い
 - コンパイル済モードのJRubyはRuby1.8の1.5~2倍速い

※JRuby開発者の一人、Charles Nutter氏のブログ
<http://headius.blogspot.com/2007/03/behind-scenes-jruby-098-released.html>

© XML Consortium



人材の確保



- Ruby技術者は少ない
- Java経験者なら数日で書けるようになる
- うまく書くには 1～3ヶ月はかかる
- 学習しながら開発するなら、レビューとリファクタリングの時間をしっかり確保

© XML Consortium



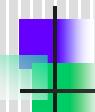
品質の確保



自動テスト

IDEの進化にも期待

© XML Consortium



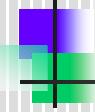
RoRでうまく開発するために



- ほどよい人数
- 設計やコードの書き方についての合意
- コミュニケーション

良い名前をつける

© XML Consortium



ご清聴ありがとうございました。

© XML Consortium