

開発者による開発者のためのカンファレンス

# Seasar Conference 2009 Autumn

Presented by  
The Seasar Foundation  
and the others



## ゆみるはこうげきりよくが2あがった！

- Ymirアプリケーションでの周辺機能の実現方法 -

Ymirプロジェクト プロジェクトリーダー  
(株)アークシステム

横田 健彦 (id:skirnir)



# Ymirのセッションにようこそ！



前回のカンファレンスでは  
聴講者も少なく



# 満席率で



約10%



# がらーん



としていて  
若干寂しくもあったんですが



# 今回はなんと前回の





約5.7倍



(満席率換算)




の方が聴講に来てくださって  
本当に感謝です！



# じこしょうかい





- 横田 健彦(よこた たけひこ)
    - ネット上ではSkirnir(すきーるにる)と名乗っています
      - (株)アークシステム所属
        - Javaや.NETのお仕事募集中!
    - プラグイン機構がウリのCMS「Kvasir/Sora」を開発
    - 北欧神話好き
      - Skirnir、Kvasir、Ymirも北欧神話のキャラクターの名前です
    - 阪神タイガース好き
      - 前回「カーネルサンダースの呪いが解けて優勝できるか!？」と発言
- 熾烈な3位争い中 



# きょう おはなしすること



# きょう おはなし すること



- ゆみるのおさらい
- ゆみるのそうび1
- ゆみるのそうび2



# ゆみるのおさらい







- フィルタ指向のシンプルなWebアプリケーションフレームワーク
  - ゆみる = 北欧神話に出てくる巨人。血肉が世界の元となった。
  - 公式サイト: <http://ymir.seasar.org/>
- MVCモデルのCの部分を担当
  - お勧め構成はYmir+FreyjaのZPT実装+DBFlute
    - Freyja...XMLテンプレートエンジン構築F/W。ZPT実装を持っている。
      - <http://www.skirnir.net/product/freyja/>

## • Seasar2.4のHOT deploy対応

Seasar Conference 2009 Autumn



- 構築事例も着々と増加
- 保育所・託児所のポータルサイト「たくほい」
  - <http://www.takuhoi.net/>
  - 9/14(月)オープン予定!

たくほい TAKUHOI 保育所・託児所のポータルサイト Beta version

たくほいは日本全国の保育所・託児所のポータルサイトです。地名、路線、キーワードからあなたとお子様にピッタリの保育所・託児所をお探しいただけます。

○ トップページ ○ たくほいへ掲載する ○ たくほいについて ○ お問合せ ○ プライバシーポリシー

保育所・託児所検索

地名から条件を入力して検索できます。

地名: 東京都 市区町村を選択してください

営業時間:  時間指定 0 時 0 分 ~ 0 時 0 分  
 時間指定なし  朝7時から営業  夜11時以降も営業  24時間可能  時間指定

一時預かり: 1時間あたり

受入年齢: ~

マイページログイン

ログイン  
パスワード  
 次回からログインを省略  
ログイン  
\* 新規会員登録  
\* ID・パスワード忘れ

たくほいモバイル

たくほいを携帯電話でもご利用いただけます。



- ViliとVe
- Vili(ヴィリ)...T2 Projectで公開されているプロジェクト作成支援Eclipseプラグイン
  - <http://eclipse.t2framework.org/updates/3.3/>
  - プロジェクトの雛形(スケルトン)とプログラム部品(フラグメント)からプロジェクトを作成
- Ve(ヴェー)...Ymirアプリケーション開発支援Eclipseプラグイン
  - <http://eclipse.seasar.org/updates/3.3/>



# ゆみるのそうび1





# ばち



# バッチ



- 実案件ではWebアプリケーション以外にバッチプログラムを作ることが多い
- Webアプリケーションで使われているコンポーネントをバッチからも利用したい
  - Logicクラス、Daoクラス、...
- バッチプログラムでS2のコンポーネントをきちんと読み込むのってメンドクサイ



- バッチプログラム作成のためのモジュール「ymir-batch」を新たにリリース
- ymir-batchを簡単に使うためのViliフラグメントも同時リリース





- バッチプログラムの作り方

- 1.プロジェクトにymir-batchフラグメントを追加
- 2.Batchインタフェースの実装クラスをbatchパッケージに作成



```
public interface Batch {  
    boolean init(String[] args) throws Exception;  
  
    int execute() throws Exception;  
  
    void destroy() throws Exception;  
}
```



- `init()`
  - 初期処理を行なう
  - 処理を中断する場合は`false`を返すこと
- `execute()`
  - 実際の処理を行なう
  - 返り値がバッチプログラムの終了コードになる
- `destroy()`
  - 終了処理を行なう



- バッチプログラムの動かし方  
– 「run.bat バッチクラス名」とする



# ゆみるのそうび2





# 鎖かたびら



mail<sup>1</sup>

—— n. 鎖かたたびら, よろしい;

---



# メール





- メール送信処理が必要なアプリケーションは多い
- JavaMailを直接ハンドリングするのはしんどい
- そこでozacc-mail library
  - <http://spring-ext.sourceforge.jp/oim/>
  - S2との相性も良い



```
// Mailインスタンスの生成
Mail mail = new Mail();
mail.setFrom("shop@example.com", "XXXオンラインショップ");
mail.addTo("taro@foo.com", "山田太郎さま");
mail.addBcc("order@example.com");
mail.setSubject("山田太郎さま:ご注文の確認");
mail.setText("山田太郎さま、商品を購入いただきありがとうございます。");

// メールの送信
sendMail.send(mail);
```

(<http://spring-ext.sourceforge.jp/oml/> のサンプルを改変)



- 件名や本文を動的に組み立てるのが若干手間



- mailsender

- SeasarプロジェクトのCMSサブプロジェクト (sandboxプロジェクト内) で開発されているメール送信ライブラリ
- ozacc-mail libraryベース
- 現在のバージョンは0.0.2



- mailsenderの特徴

- 件名と本文をFreemarkerテンプレートから生成
- メール送信処理をインターセプタ方式で実現
- Gmailのメールサーバを使ったメール送信に対応



```
public interface ShopMailSender {  
    @Subject("${name}様:ご注文の確認")  
    @BodyTemplate("customer.ftl")  
    void sendToCustomer(Mail mail, CustomerDto dto);  
}
```

- インタフェースまたはabstract classを作成
- メール送信メソッドの実装は不要
- アノテーションでテンプレートを指定
  - Subjectアノテーション...件名のテンプレート
  - BodyTemplateアノテーション...本文のテンプレートパス
    - ShopMailSenderがあるパッケージの「ShopMailSender.customer.ftl」という名前のリソース
- テンプレートに埋め込む値を引数 (CustomerDto) で指定
- 件名、本文以外はMailオブジェクトで指定

# ゆみるのそらび2



CustomerDto:

```
public class CustomerDto {
    private String name;

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }
}
```

customer.ftl:

```
#{name}さま、商品を購入いただき  
ありがとうございました。
```



```
// DTOの準備
CustomerDto dto = new CustomerDto();
dto.setName("山田太郎");

// Mailインスタンスの生成
Mail mail = new Mail();
mail.setFrom("shop@example.com", "XXXオンラインショップ");
mail.addTo("taro@foo.com", dto.getName() + "さま");
mail.addBcc("order@example.com");

// メールの送信
shopMailer.sendToCustomer(mail, dto);
```





- プロジェクトへのメール送信機能の追加方法

1. mailsenderフラグメントをプロジェクトに追加

2. メールサーバの設定をmailsender+sendMail.diconに記述

- 開発時用設定 : src/main/resources-develop/
- 統合テスト用設定 : src/main/webapp-it/WEB-INF/classes/
- 本番用設定 : src/main/webapp-release/WEB-INF/classes/

3. mailsenderパッケージにMailsenderインタフェースを作成

4. 本文のためのテンプレートリソースを作成



- Gmailのメールサーバを使う場合の設定例

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE components PUBLIC "-//SEASAR//DTD S2Container 2.4//EN"
    "http://www.seasar.org/dtd/components24.dtd">
<components>
  <component name="sendMail"
    class="org.seasar.cms.mailsender.impl.SendMailImpl">
    <property name="charset">"ISO-2022-JP"</property>
    <property name="host">"smtp.gmail.com"</property>
    <property name="port">587</property>
    <property name="protocol">"smtp"</property>
    <property name="starttlsEnabled">true</property>
    <property name="username">"(ユーザ名)"</property>
    <property name="password">"(パスワード)"</property>
  </component>
</components>
```



てき が あらわれた！





おわりに



- Ymirアプリケーションへの機能追加の方法として、バッチプログラムの追加方法とメール送信機能の追加方法についてご紹介しました
- これからも、実案件で使えるフレームワークを目指して(歩みは遅いかもかもしれませんが)頑張っていきたいと思っています
- 皆さんもYmirを使って「れべる255 ゆみる」を目指してみませんか？



ご清聴ありがとうございました