

# Cubby

**100% Pure JSP**

**Web Application Framework**

2006/8/19@Nulab Jaian Recital

Toshitaka Agata(agata@nulab.co.jp)

# Cubbyって何？

- 100% Pure JSP Web Application Framework
- Zero Configuration
- ベリーシンプル&スモール
  - ・ クラス数約50個
- RESTっぽいURIをサポート
- 直感的な名前のタグライブラリ
- Cubby = 整理棚

# なぜにJSP？

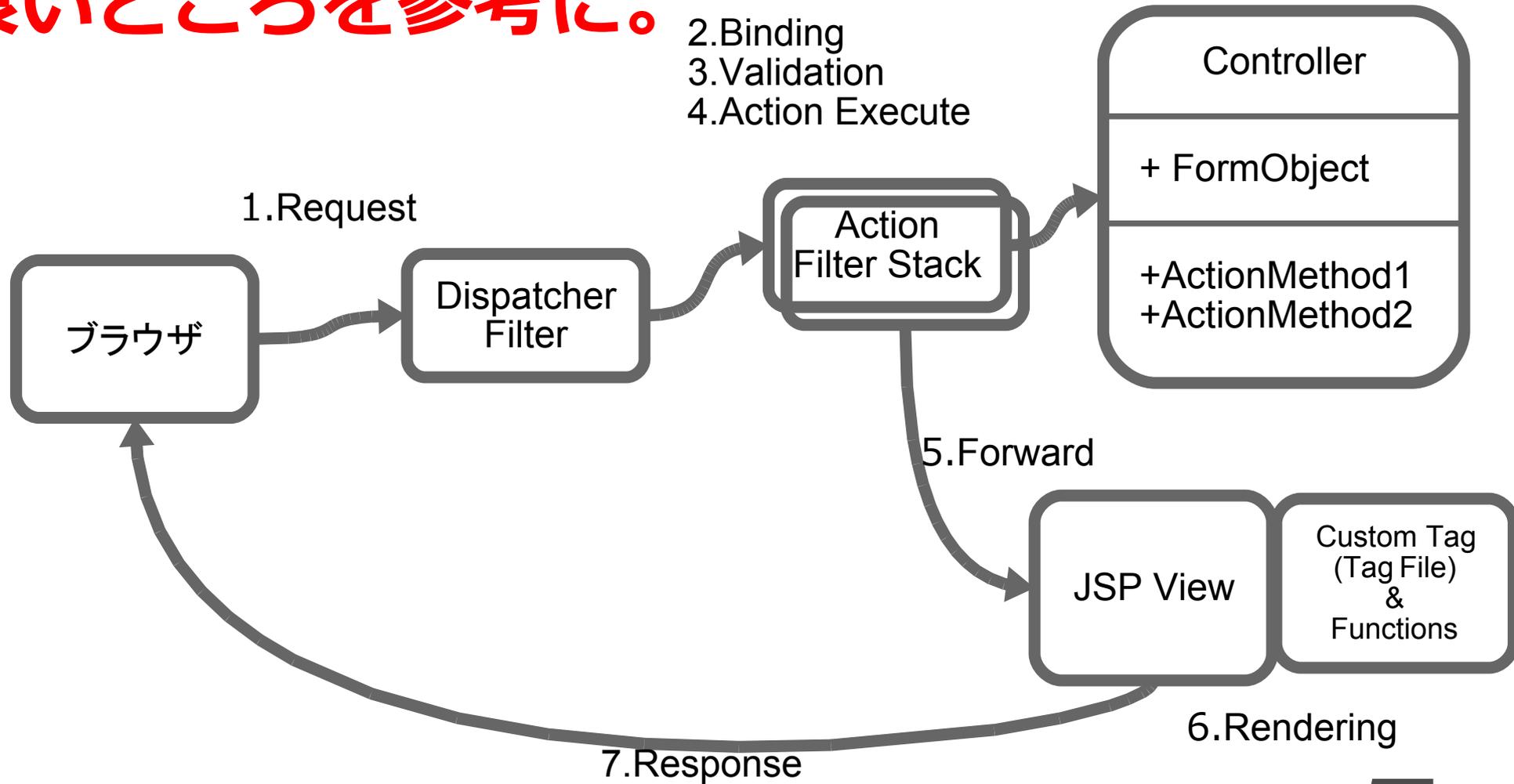
- 最近嫌われ者のJSP・・・
  - ・ スクリプトレットだし、
  - ・ エラーがわかりにくいし、
  - ・ Pure HTMLテンプレートじゃないし、
  - ・ 流行りではないし、
  - ・ etc...

# なぜにJSP？

- 学習容易性
  - 新しい知識はいりません。
- 最近のJSP2.0はいい感じですよ。
  - EL式(Expression Language)
  - JSTL(JSP Standard Tag Library)
  - Functions
  - タグファイル
    - フラグメント
    - Dynamic Attributes
  - エラー出力の改善

# Cubbyの概要

Struts, WebWork, Ruby on Railsの  
良いところを参考に。



Strutsとだいたい同じ範囲をカバーします。  
やり方はだいぶ異なります

# View (JSP)

- 基本はHTMLのタグ。

Cubbyに必要な属性を追記します。

## Pure HTML

```
<input name="userId" value="" />
<select name="typeId">
  <option value="1">仕事</option>
  <option value="2">個人</option>
  <option value="3">その他</option>
</select>
```

## Pure JSP

```
<%@ taglib tagdir="/WEB-INF/cubby/tags" prefix="t" %>
<t:input name="userId" />
<t:select name="typeId"
  items="${todoTypes}" valueProperty="id" labelProperty="name">
  <option value="1">仕事</option>
  <option value="2">個人</option>
  <option value="3">その他</option>
</t:select>
```

# View (JSP)

- 全てのカスタムタグはタグファイルでできています！ (WEB-INF/tags/\*.tag)

input.tag

```
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://www.nulabco.jp/cubby" prefix="f" %>
<%@ tag dynamic-attributes="dyn" %>
<%@ attribute name="type" type="java.lang.String" rtexprvalue="true" required="true" %>
<%@ attribute name="checkedValue" type="java.lang.Object" rtexprvalue="true" required="false" %>
<input type="${type}" ${f:toAttr(dyn)} ${f:checked(dyn['value'],checkedValue)}/>
```

- Dynamic Attributesで、定義していない属性も出力可能です！

# Controller&Action

- 1つのControllerに複数Actionメソッドを定義

## Controller

```
@Action
public class TodoController extends Controller {
    ...
    public String show() {
        ...
        return "show.jsp";
    }
    public String edit() {
        ...
        return "edit.jsp";
    }
    public String confirm() {
        ...
        return "confirm.jsp";
    }
    public String save() {
        ...
        return "@list";
    }
}
```

return “show.jsp”  
show.jspにフォワード

return “@list”  
listにリダイレクト

return “/common/error.jsp”  
/common/error.jspに  
フォワード

return “@/login/logout”  
/login/logoutにリダイレクト

# Controller&Action

- メタ情報はアノテーションで（ちょっとだけ）

## Controller

```
@Action
@Form("loginDto")
public class LoginController extends Controller {
    public static TodoLogic todoLogic = new MockTodoLogic();
    @Session
    public User user;
    public LoginDto loginDto = new LoginDto();
    public String login() {
        return "login.jsp";
    }
    @Validation(errorPage="login.jsp")
    public String login_process() {
        if(todoLogic.login(loginDto)){
            user = new User(1, "agata");
            return "@/todo/list";
        } else {
            errors.add("ユーザIDかパスワードが違います。");
            return "login.jsp";
        }
    }
    public String logout() {
        user = null;
        return "@login";
    }
}
```

## @Action

publicメソッドをアクションメソッドとみなす  
(メソッド単位でも設定可能)

## @Form(プロパティ名)

パラメータの自動バインディング対象を指定  
(メソッド単位でも設定可能)

## @Validation(errorPage="エラー時の遷移先")

アクションの実行前にバリデーションを実行

## @Session

セッションスコープの変数として、  
自動バインディング  
(その他publicなフィールドはリクエストスコープ)

# Validation

- コードで入力検証を記述

Model

```
public class Todo implements Validatable {  
  
    public static Validators VALIDATORS = new Validators();  
    static {  
        VALIDATORS.add("text", new Required(), new MaxLength(10));  
        VALIDATORS.add("memo", new Required(), new MaxLength(100));  
        VALIDATORS.add("typeId", new Required());  
        VALIDATORS.add("limitDate", new DateFormat());  
    }  
  
    public Validators getValidators() {  
        return VALIDATORS;  
    }  
    ...  
}
```

# その他の機能

- URI自動マッピング(Plugin化)
  - Rails風
  - /todo/list→TodoController#list
  - /todo/confirm→TodoController#confirm
  - /agata/todo/1→TodoController#show?id=1
- HTMLプレビュー
  - JSPでもブラウザによるプレビューが（ある程度）  
できます！
- @ActionFilter WebWork2 & Rails風

# ロードマップ

- 2006年11月 Ver0.9
- 2006年12月 Ver1.0-RC
  - リファクタリング
  - ドキュメントの整備
- 2007年1月 Ver1.0
  - サンプルの整備
  - JavaDocコメント

一緒に開発してくれる方、  
募集中です!!

仕様に対するご意見も、  
お待ちしております。