Seasar Conference
2007 Spring

今から役立つ! Teeda入門

2007.5.27

大谷 晋平

本日のアジェンダ

- Teeda概要
- Teedaの歩んできた道
- Teedaの特徴
- Teeda入門
- Teedaの歩んでいく道
- まとめ

自己紹介1

- 名前
  - 大谷 晋平(HN:shot)
- コミッタ活動
  - Teeda/S2JSF/S2Struts
  - Seasar2
- Blog
  - http://d.hatena.ne.jp/shot6/
- メールアドレス
  - shinpei.ohtani@gmail.com

自己紹介2

- 最近の執筆活動・メディア紹介記事
  - **JavaExpert 創刊号**
    特集2:「Seasar2.4/Teedaの正体を探る」(共同)
  - **Software Design 2007年05月号**
    - OSCJ Times 第9回「Teeda」(共同)

始める前に

- 本日のセッションは、
  - 前半後半で言うところの前半です。
    - 難しいことより、基礎知識をしっかり説明
    - 詳細な記述の部分は後で振り返ってみてください。
  - 「現場で役立つ実践Teeda」が後半戦
    - Teedaユーザとして共に歩んできたたかのりさんの発表
    - 具体的な事例、活用のTips
  - ひがさんのセッションの説明を受けて・・・
    - Teedaの詳細をもう一歩つっこむ

Teeda概要

- Teedaとは
  - JavaのWebアプリケーションフレームワーク
    - プレゼンテーションの開発をよりシンプルに。
      - **JSF meets DI x AOP**
    - Javaで**Page駆動開発**を実現
      - POJO(Plain Old Java Object)ベースでの開発
    - Seasar2.4ベース
  - http://teeda.seasar.org/ja/
  - Teedaとは沖縄の言葉で「太陽」

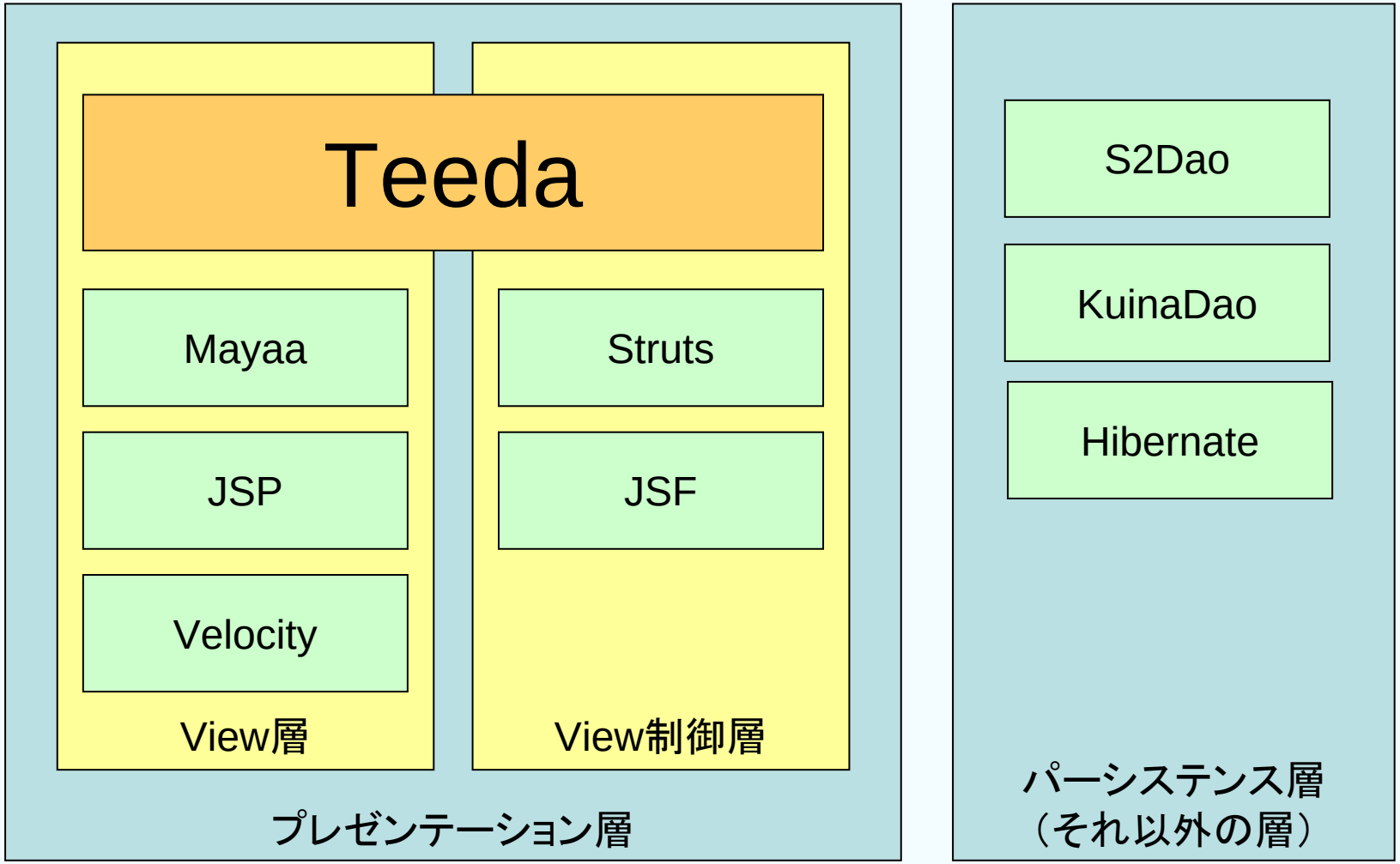
Teedaの歩んできた道

- リリース状況
  - 2006/11/11 1.0正式リリース
  - 現在最新版は**1.0.7**。1.0.8も着々と進行中
    - 平均して1ヶ月に1度の安定したリリース状況
  - **現在は機能追加よりも安定化を最優先**
- 案件適用状況
  - 適用事例は順調に増加中（感謝！）
  - **フィードバックによって更に改善・改良**
    - **Seasar-user ML**
    - **Blog**
    - **案件からの直接フィードバック**

Teedaの特徴

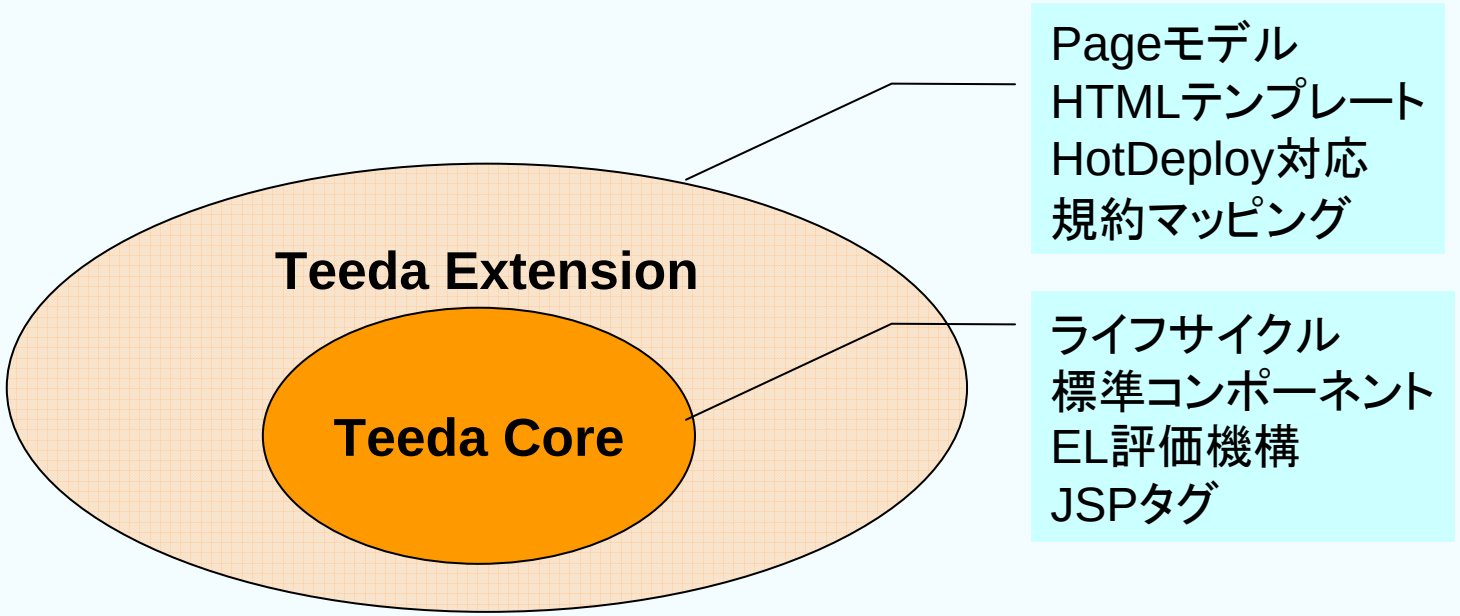
- Teedaの特徴
  - JSFベースの2層構造
    - **Core**と**Extension**
    - Add-onとして、**Ajax機能**、**Testing機能**
  - Javaで**Page駆動開発**を実現
  - Seasar2.4ベース
    - HotDeploy対応
  - 設定ファイルレス
  - Viewをシンプルな**HTML**で記述可能
  - 同名プロパティの状態受け継ぎ

Teedaのテクノロジ的な位置づけ



Teedaの特徴（詳細1）

- JSFベースの2層構造
  - Core → JSF1.1の実装
  - Extension → よりユーザへの利便性提供を実現



Teedaの特徴（詳細2）

- Page駆動開発とは
  - HTMLをベースとした開発スタイル
  - HTMLにJavaクラスをマッピングする
    - **対応関係が明確なのでムリが少ない**
      - 1Page = 1HTML + 1POJOクラス
        - » このPOJOを**Page**クラスと呼びます
    - マッピングの設定が必要なの？
      - **Teedaではシンプルに名前をあわせるだけ。**
      - add.html → AddPage.java
      - Strutsのように設定ファイルが必要ない

Teedaの特徴（詳細2続き）

- Pageモデル
  - PageクラスはHTMLに対応するデータ構造
  - Pageクラスは、プロパティとアクションメソッドを持つ

hoge.html

```
<html>
 <body>
  <form id="addForm">
   <input type="text" id="arg1" />
   <input type="text" id="arg2" />
   <input type="submit" id="doHoge"
     value="execute" />
  </form>
 </body>
</html>
```

```
public class HogePage {
 private Integer arg1;
 private Integer arg2;
 public String doHoge() {
  ・・・
 }
 //setter/getter
}
```

Teedaの特徴（詳細3）

- Seasar2.4ベース
  - **HOT deploy ready!**
    - **LLのようなStep by Stepな開発が可能**
      - アプリケーションサーバを稼動したまま、開発可能
  - 規約によって開発を効率化
    - 設定ファイルを書く時間を極小化
  - Seasar2.4のフルスタック「**Chura**」のWeb部分
    - SuperAgileではTeeda+S2.4+S2Dao
    - EasyEnterpriseではTeeda+S2.4+KuinaDao(+JPA)

Teedaの特徴（詳細4）

- 設定ファイルレス
  - ゼロでは無いが、最小限。
  - **XML設定地獄からの脱却**
    - 画面遷移の登録設定
    - 使用するモデルの登録設定
  - **あるべき設定はあるべき場所へ**
    - 画面遷移の設定は、HTMLへ記述する
      - 驚き最小限の法則
    - モデルの登録は本質的でない追加作業。

Teedaの特徴（詳細4続き）

- 設定ファイル比較

	画面遷移の設定	モデルの登録
Teeda	HTMLに簡単な規約で記述するだけなので直感的。 <input type="button“ id=“goAddResult"/>	初期設定時に規約での探索の基点となるパッケージ名を登録するのみ。
従来型WebFW	外出しのXMLに遷移ごとに記述する。一極集中で管理しやすい反面、設定量の膨大によって、保守性低下。	外出しのXMLに記述する。一極集中で管理しやすい反面、モデル名の変更などに追従できない。

Teedaの特徴（詳細5）

- Viewをシンプルなhtmlで記述可能
  - JSPなどの代わりにHTMLで記述
    - HTMLテンプレートと呼びます
  - 設計の時に使ったHTMLのMockをそのまま開発に使う
    - Teedaでは適切なidを振るのみで開発可能
    - add.htmlで下記のような記述をすると、AddPageのarg1プロパティとバインディングされる。

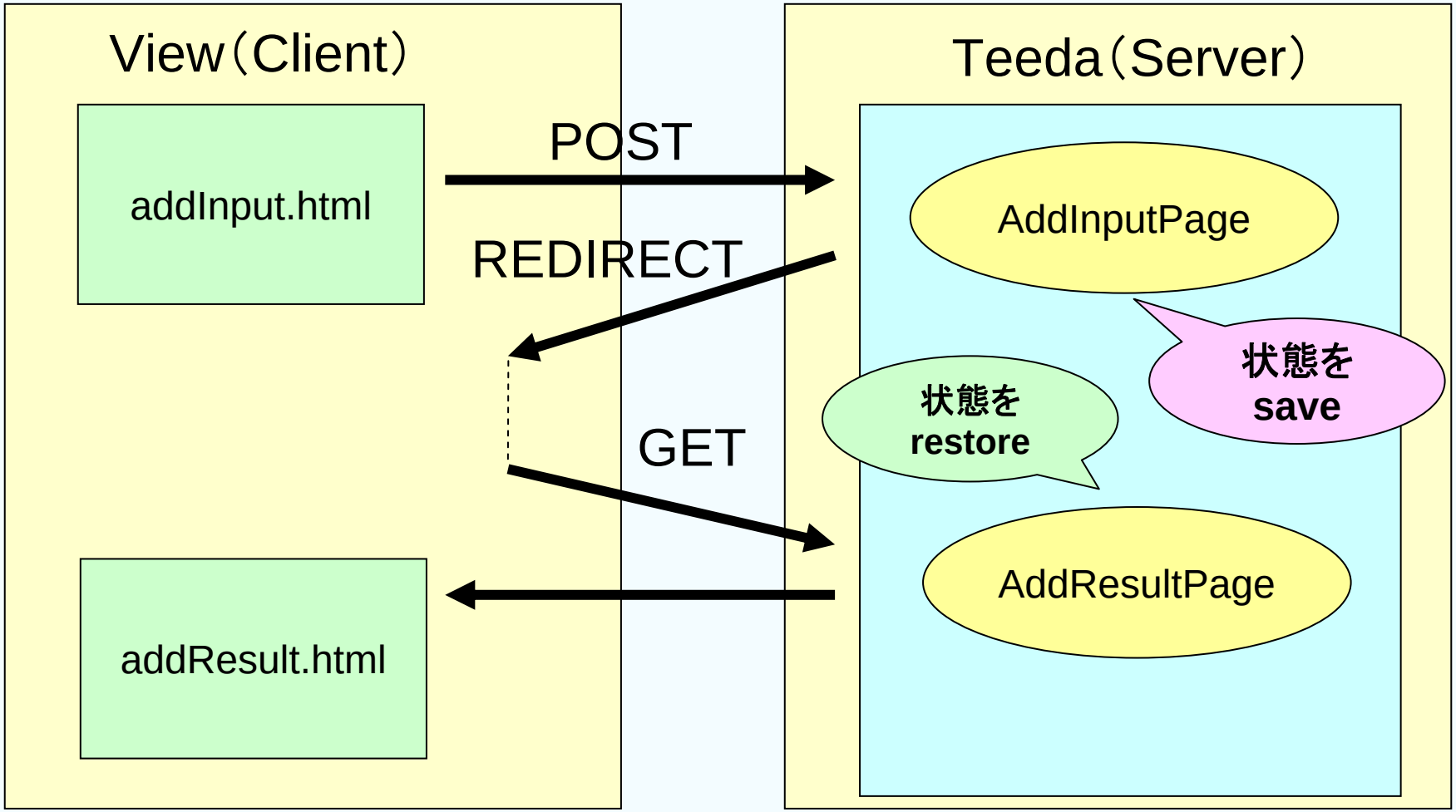
```
add.html
<html>
<body>
<form id=“addForm”>
 <input type="text" id="arg1"/>
```

```
public class AddPage {

private int arg1;
```

Teedaの特徴（詳細6）

- 同名プロパティの状態受け継ぎ



Teedaの特徴ふりかえり

- Teedaを使った開発者は、
  - HTMLとPageクラスだけ書けばよい
    - 対応関係は1対1（**add.html ←→ AddPage**）
  - HTMLは設計時に使ったMockを流用できる
  - 開発時に、
    - 設定ファイルに費やす時間を規約により削減できる
    - HOT deployでサーバ再起動の無駄な待ち時間を削減できる

これさえ覚えれば、
Teedaの特徴については完璧です。

Teeda入門概要

- Teeda入門では以下の事をご紹介します
  - 基本コンポーネントの使い方
  - Teedaのライフサイクルメソッド
  - Validator/Converterのかけかた
  - Demo

Teeda入門(outputText)

- outputText
  - 文字列を出力するコンポーネント
  - Spanタグでidあり

```
hello.html
 Hello <span id=“name">World!</span>
```

```
HelloPage
public class HelloPage {
 private String name = “Teeda”;
 //getter/setter
```

```
結果は
 Hello Teeda
```

Teeda入門(inputText)

- inputText
  - 入力コンポーネント
  - Inputタグ（タイプはtext）で、idあり

```
add.html
<form id="addForm">
 <input type="text" id="arg1" title="IN" />
```

```
AddPage
public class AddPage {
 private Integer arg1;
 //getter/setter
```

Teeda入門(button1)

- button（do）
  - Pageの任意のメソッド（引数なし）を呼び出す
  - Inputタグ（タイプはsubmit）で、idが”do*”

```
add.html
<form id=“addForm”>
 <input type=“submit” id=“doCalculate”
value=“submit”/>
```

```
AddPage
public class AddPage {
 public String doCalculate() {
   result = arg1 + arg2;
   return “addResult”;//nullを返すと自画面遷移
 }
```

Teeda入門(button2)

- button（go）
  - Pageのメソッドを呼び出さない
  - Validationはかかる
  - 状態維持はされる
  - Inputタグ（タイプはsubmit）で、idが”go*”

```
add.html
<form id=“addForm”>
 <input type=“submit” id=“goAddResult”
value=“submit”/>
```

Teeda入門(button3)

- button（jump）
  - Pageのメソッドを呼び出さない
  - Validationもかからない
  - 状態維持もされない
  - Inputタグ（タイプはsubmit）で、idが”jump*”

```
add.html
<form id=“addForm”>
 <input type=“submit” id=“jumpAddResult”
value=“submit”/>
```

Teeda入門(forEach)

- forEach
  - tbodyタグかdivタグで繰り返しの配列/Listを指定

```
<table>
 <tbody id="aaaItems">
  <tr>
    <td><span id="foo">hoge</span></td>
    <td><span id="bar">huga</span></td>
</tr>
 </tbody>
</table>
```

```
public class FooPage {
 private String foo;
 private String bar;
 private FooItem[] aaaItems;//FooItemは単なるDTO
```

Teeda入門(ライフサイクルメソッド1)

- initialize
  - Pageのインスタンス生成時に1度だけ呼ばれる
    - 画面の初期状態を設定
  - 戻り値はString、Class、voidのどれかを選択可能
    - Stringの場合はHTML名を指定
    - Classの場合はPageクラスを指定
    - Voidの場合は自画面にのみ遷移

```
public Class initialize() {
 //初期化ロジック
  return null;
 }
```

Teeda入門(ライフサイクルメソッド2)

- prerender
  - 画面の描画直前に毎回呼ばれる
    - 表示上の見栄えを整理する
  - 戻り値はString、Class、voidのどれかを選択可能
    - Initializeと同様

AddPage

```
public class AddPage {
 public Class prerender() {
 //表示上の設定項目を整える
  return null;
 }
}
```

Teeda入門(Validator/Converter)

- Validator
  - ValidatorはPageクラスにアノテーションで指定
  - メソッドかフィールドに指定

```
@Required
 @Length(minimum = 2, maximum = 5)
 private Integer arg1;
```

- Converter
  - ConverterもValidator同様

```
@DateTimeConverter
public Date getLimitDate() {
  return limitDate;
}
```

Teeda入門(デモ)

- Demoアプリ

Teedaの歩んでいく道

- Teeda Roadmap
  - **Teeda1.0**
    - **Stableとして、Policyも今後変更なし**
      - HTMLテンプレートとのマッピングはidベース
      - 1HTMLに対し、1Pageクラス
    - 地道な修正・改善を繰り返す
      - ForEachのネスト対応、faces-configの順番制御など
  - **Teeda1.1**
    - **Devとして、基本は踏襲しつつも積極的に変更**
    - よりコード量の削減を目的とした変更を取り入れる
      - publicフィールド対応（Seasar2.5）
    - カスタマイザビリティと開発生産性の強化
    - JSF1.2対応

Teeda1.1 Roadmap1

- publicフィールド対応
  - getter/setterの記述量削減
  - Seasar2.5レベルで対応

AddPage

```
public class AddPage {
 public Integer arg1;//getter/sessterなし
 public Integer arg2;
 public Integer result;
 public String doCalculate() {
   result = arg1 + arg2;
   return “addResult”;//nullを返すと自画面遷移
 }
}
```

Teeda1.1 Roadmap2

- カスタマイザビリティの強化
  - 規約拡張機能の強化
  - カスタムコンポーネント開発支援
    - できるだけ作成するリソースを減らす
- 開発生産性の強化
  - Bluescreen機能
    - 開発時にどこで何のエラーが発生したかが瞬時にわかる
  - Configurationの状況を可視化
    - Pageクラス、各内部的なクラスなどの状態を可視化

Teeda1.1 Roadmap3

- JSF1.2対応
  - 段階的にJSF1.2の仕様を取り入れていく
    - EL式の解釈をJSP2.1に委譲
    - Ajaxサポートの強化
    - 標準コンポーネントの強化
      - カスタムコンポーネントの作成
- Teeda1.1はTiger前提

Teeda Roadmap

- ドキュメント作成
  - JSF自体のドキュメント
  - Teeda extensionのドキュメント
    - リファレンスの強化
    - チュートリアルの作成
    - コンポーネントショウケースの作成
      - コンポーネントの使い方、ソース、詳細ドキュメントのセット

コミッタ・協力者募集中です

それ以外のオモシロRoadmap

- それ以外のオモシロRoadmap（何の役に立つかはおいておいて・・・）
  - Java Web StartでTeedaを配布？
  - Webコンテナと同梱されて配布？

Teeda リリースプラン

- Teeda リリースプラン
  - Teeda1.0は基本2ヶ月に1度リリース
  - Teeda1.1のリリースプラン
    - 2007/07 Beta1
    - 2007/08 RC1
    - 2007/09 RC2
    - 2007/10 **1.1リリース**

Teeda Roadmapまとめ

- Teeda1.0系
  - 安定版として、これからもバグ修正・改善
  - 単体テスト、結合テストを増やしていく
  - ドキュメントの充実
- Teeda1.1系
  - 開発版。より新機能の追加に重きを置く
  - より開発の生産性を挙げる

TeedaとS2Presentationの棲み分け

- S2Presentation？
  - S2.5に同梱される
  - 開始はS2.5、S2Persistence以後なので流動的
  - Seasarの仕様（SSR）は各issueごとでそれが決まり次第実装。先に仕様決定、Wikiに掲載
- Teeda
  - S2.4で稼動
  - 枯れた安定版への階段を徐々に上る

まとめ

- 本セッションでは、
  - Teedaの特徴
  - Teedaの入門
  - TeedaのRoadmap
    - 今後の方向性、リリースプラン

各種リソース1

- Teeda各種リソース
  - http://teeda.seasar.org/ja/
  - JavaExpert 創刊号特集2
  - もっとdeepなネタが知りたければ、、、
    - ちょっと古いけど、「Inside of Teeda」
      - https://www.seasar.org/svn/sandbox/learnings/trunk/donbiki/donbiki-view/1st/InsideOfTeeda.ppt
  - サンプルを動かしたいなら、
    - DoltengのScaffold
    - JavaExpertのTODO管理
      - http://www.gihyo.co.jp/book/2007/236471/download/todo-manage.zip

各種リソース2

- Teeda各種リソース2
  - Seasar-user
  - 各コミッタのblog
    - http://d.hatena.ne.jp/shot6/
    - などなど。。。

ありがとう
ございました