JBoss Seam

Ein JEE 5 Webframework

Jörg Wüthrich

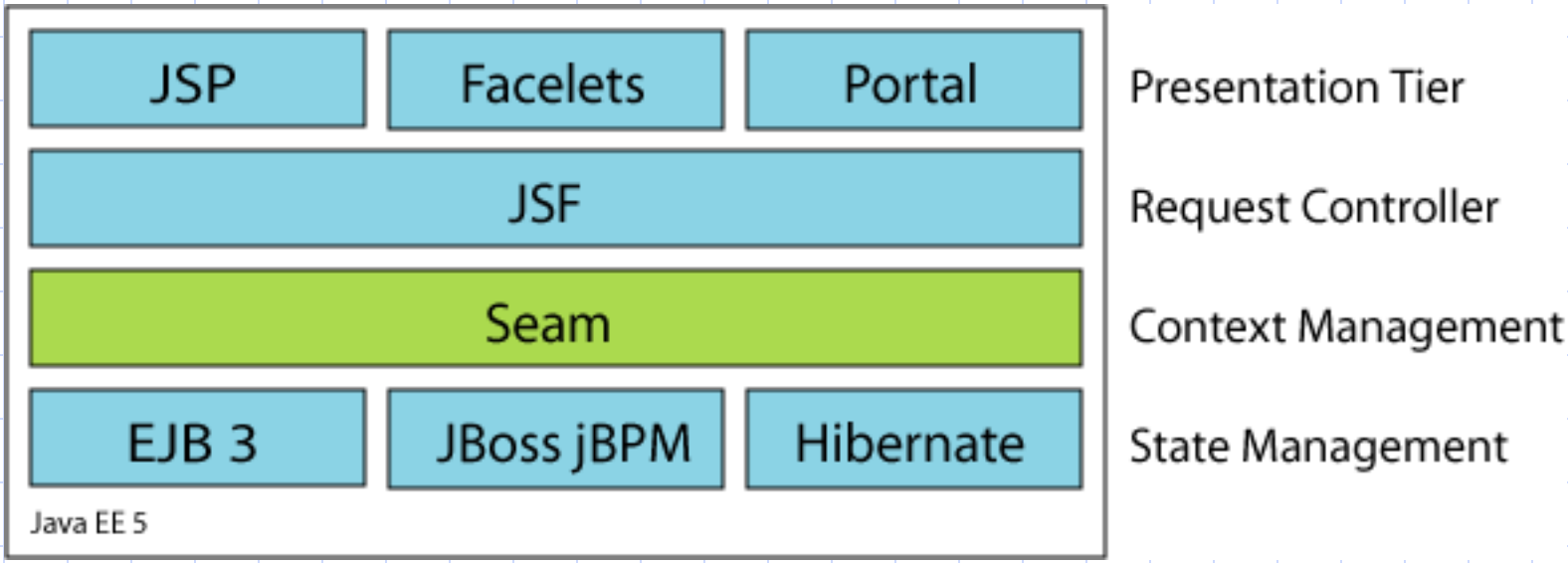
Infopoint, 4. Februar 2009

Inhalt

- Einführung
- Warum Seam?
- Zentrale Konzepte
- Demo
- Validierung
- Abschliessende Gedanken

Einführung

- Positionierung



Einführung

- Seam ist ein Applikations-Framework für Enterprise Java 5
- Integriert
  - Java Server Faces (JSF)
  - Enterprise JavaBeans (EJB3)
  - Asynchrones JavaScript und XML (AJAX)
  - Java Persistence API (JPA)
  - Business Process Management (BPM)
- Design-Fokus: Elimination von Komplexität auf Architektur- und API-Level

Warum Seam?

- Ein einfacher Weg, um mit EJB 3.0 zu beginnen
  - EJB 3.0 unterstützt die Entwicklung von transaktionalen Business Komponenten
  - Seam liefert die Integration dieser Komponenten vom GUI bis zur Persistenz-Schicht

Warum Seam?

- Der schnellste Weg um "rich" zu werden
  - AJAX Clients kommunizieren mit vielen asynchronen, konkurrenzierenden Detail-Anfragen mit dem Server
  - Seam wurde entwickelt mit guter Unterstützung für parallele Zugriffe und ausgereifter Zustands-Verwaltung
  - Richfaces und ICEFaces sind integriert

Warum Seam?

- Der beste Weg, JSF zum Fliegen zu bringen
  - Ersatz der vielen XML-Konfiguration durch ein paar Annotationen
  - Multi-Windowing Unterstützung
  - Modell-basierte Validierung
  - Saubere Integration von transaktionalen Ressourcen (JPA, JTA, EJB3)

Warum Seam?

- Gute Integration von BPM
  - Optimierung der Arbeits-Abläufe heute immer wichtiger
  - Voraussetzung, um optimieren zu können, ist Messbarkeit
  - Workflows verdeutlichen die Arbeits-Abläufe und ermöglichen Messungen
  - Seam integrierts ☺

Warum Seam?

- Persistenz wird zum Kinderspiel
  - Seam ist aus der Hibernate Community entstanden
  - Seams Konversations-Modell löst diverse Probleme der traditionell zustandslosen Web Applikationen

Warum Seam?

- Beste Unterstützung für CRUD Applikatikonen
  - Seam zu gross für eine einfache Datenbank-Applikation?
  - „seam-gen“ erstellt kleine Applikationen mit DB-Anbindung im Nu (vergleiche „ruby on rails“)

Warum Seam?

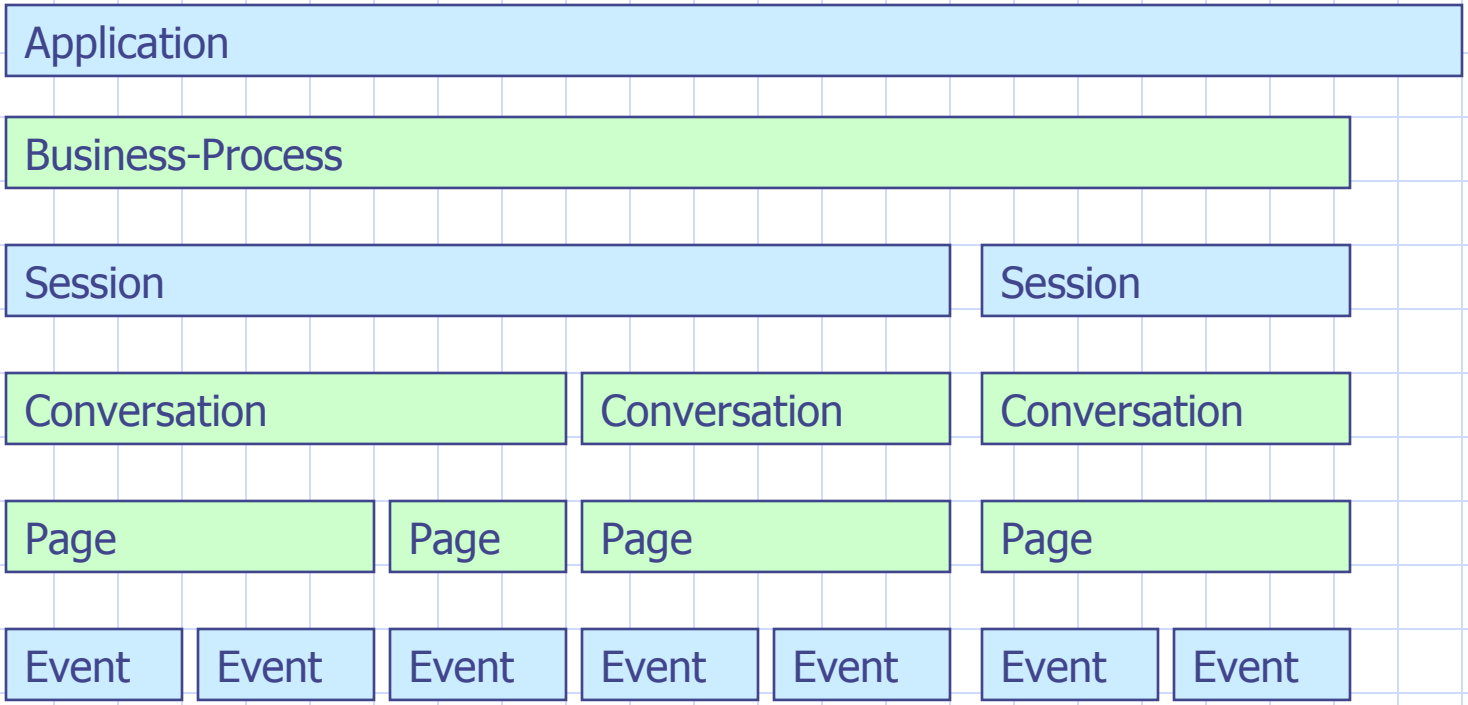
- Automatisierte Integrations-Test
  - Unittests können Interaktionen zwischen Komponenten nicht testen
  - Seam bietet einen Ansatz, um User-Interaktionen zu simulieren und so von UI bis zur Persistenz durchgängig zu testen

Zentrale Konzepte

- Komponente
  - beliebige POJOs, EJB3s (Session, Entity, MDB) oder Spring Beans
  - annotiert mit @Name(„...“)
  - lebt immer in einem Kontext (-> zuständig für Lifecycle)

Zentrale Konzepte

- Kontext



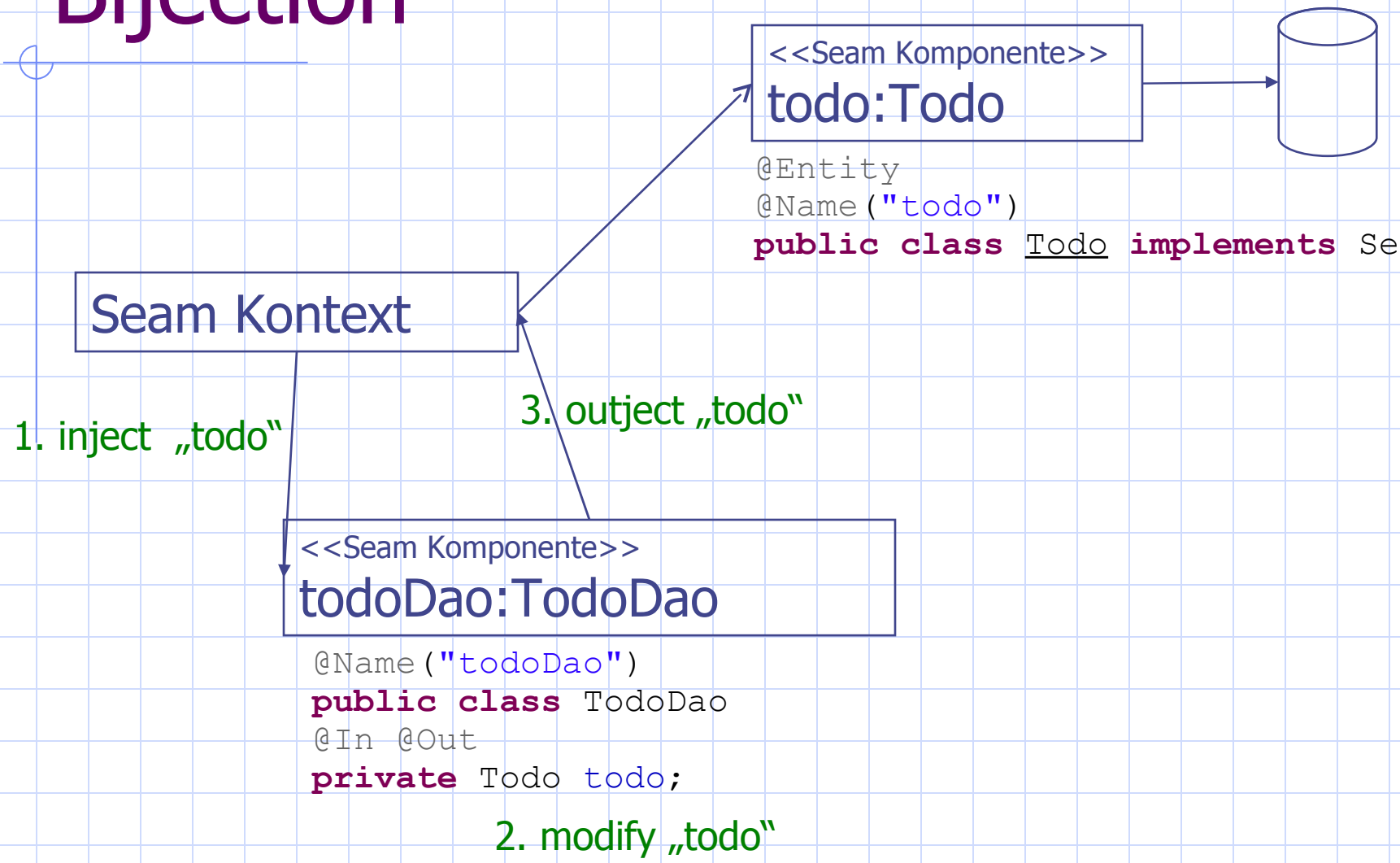
Zentrale Konzepte

- Kontext
  - Event: entspricht einem JSF-Request
  - Page: eine Instanz einer gerenderten Seite
  - Conversation: Arbeitseinheit aus Sicht des Users (kann mehrere Server-Requests beinhalten)
  - Session: Session-Scope des Servlet-APIs
  - Business-Process: hält Zustand über länger dauernden Geschäftsprozess, in welchen mehrere Akteure involviert sein können
  - Application: entspricht Application-Scope von JSF

Zentrale Konzepte

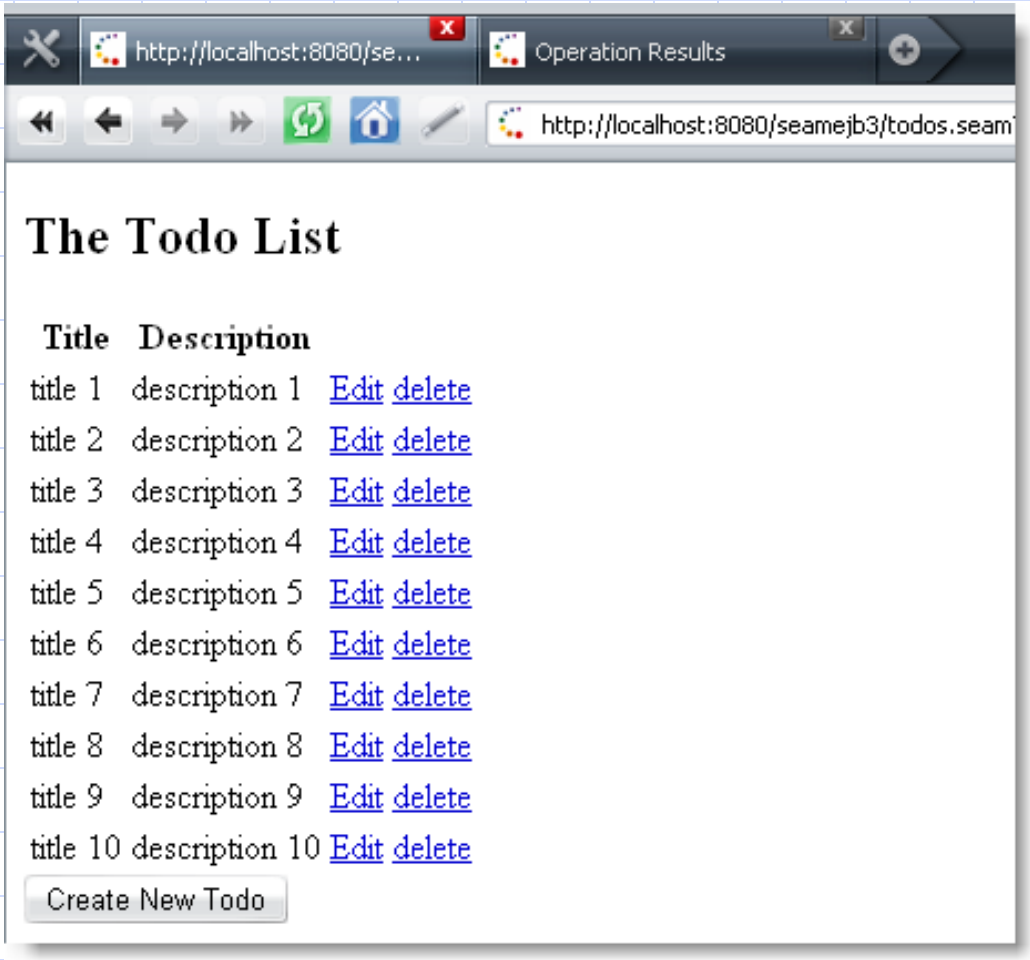
- Bijection
  - alle Seam Komponenten werden in einem Kontext gehalten
  - Injection: eine Komponente wird einer anderen aus dem Kontext zur Verfügung gestellt
  - Outjection: eine Komponente gibt eine andere nach Bearbeitung an den Kontext zurück
  - Bijection: Kunstbegriff von Seam, um In- und Outjection in ein Wort zu fassen

Bijection


<<Seam Komponente>>
todo:Todo
@Entity
@Name("todo")
public class Todo implements Seriali
Seam Kontext
1. inject „todo"
3. outject „todo"
<<Seam Komponente>>
todoDao:TodoDao
@Name("todoDao")
public class TodoDao
@In @Out
private Todo todo;
2. modify „todo"

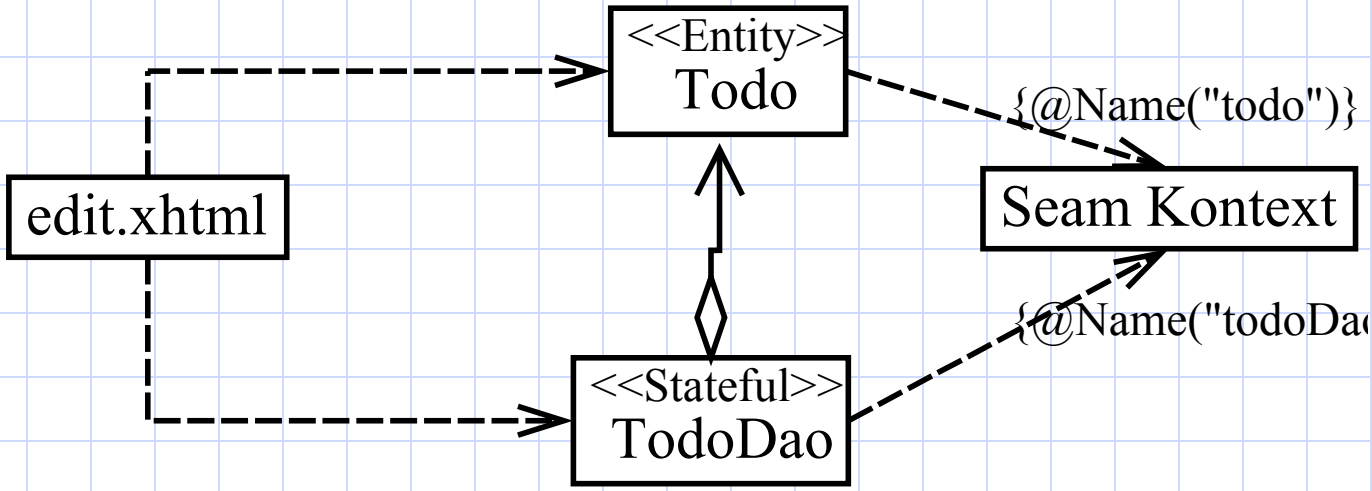
Beispiel-Anwendung

- Demo

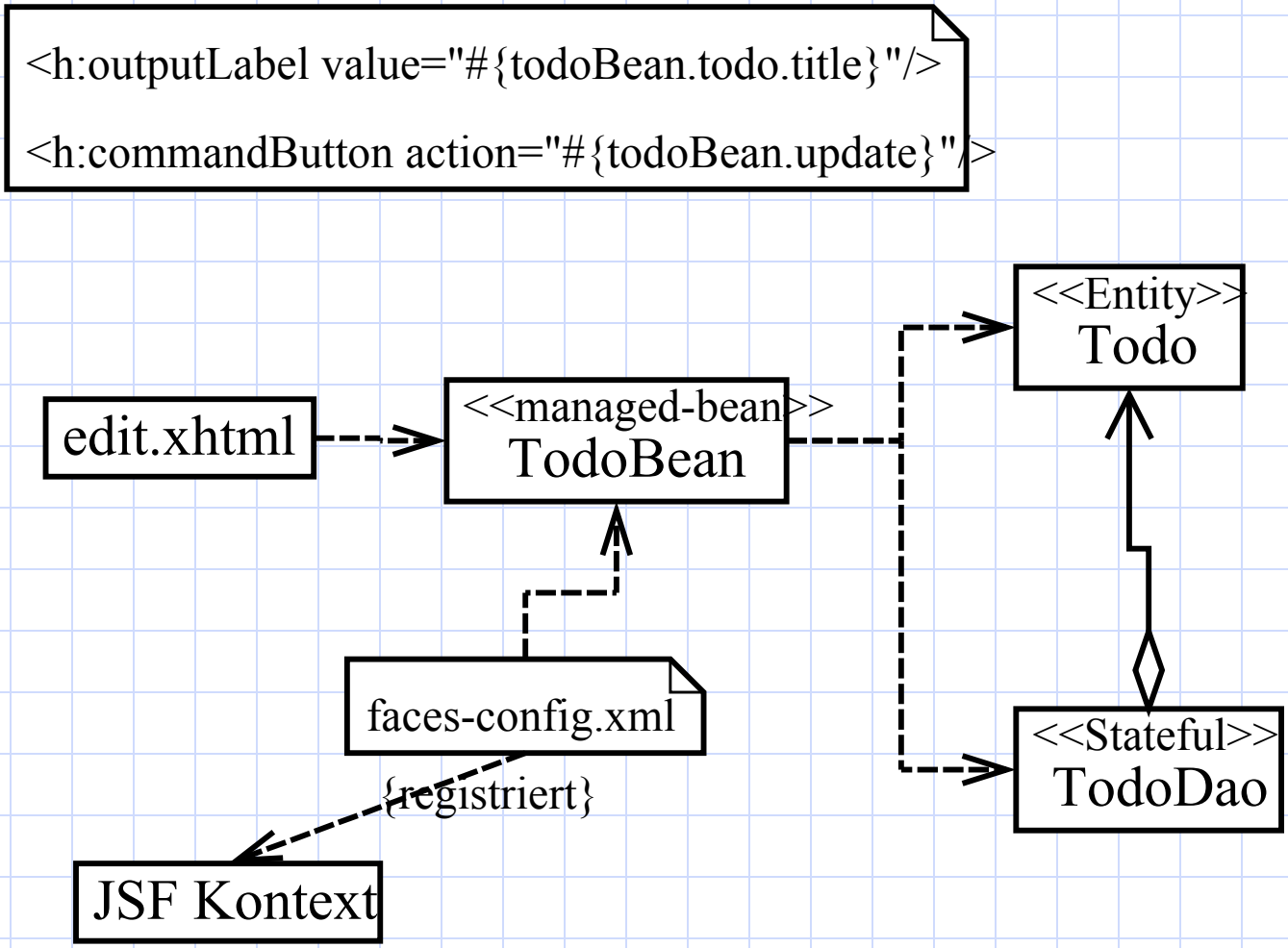


Beispiel-Anwendung Seam

```
<h:outputLabel value="#{todo.title}"/>

<h:commandButton action="#{todoDao.update}"/>
```



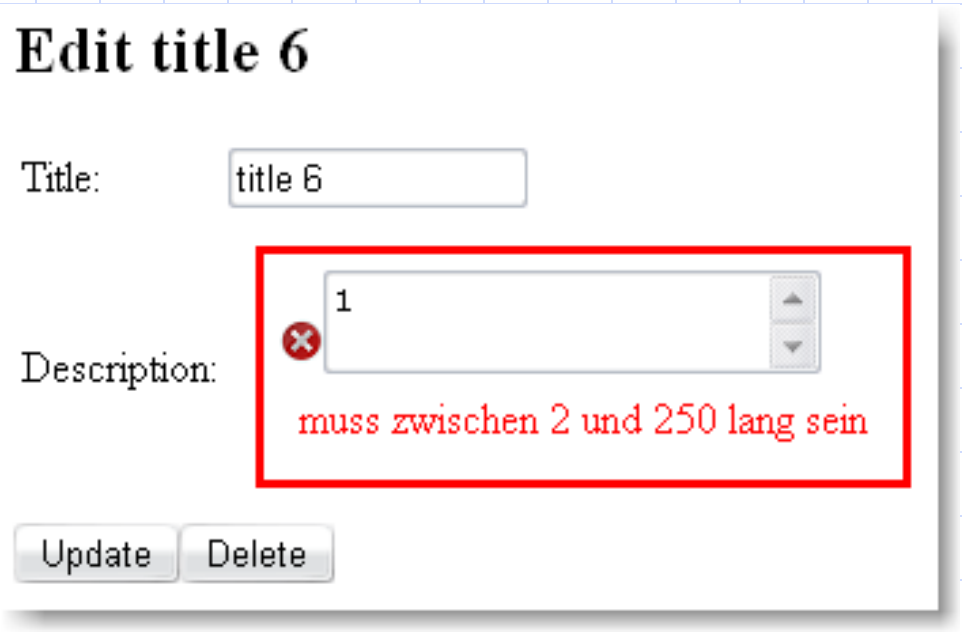
Beispiel-Anwendung JSF



Klasse TodoDao

```
@Stateful
@Name("todoDao")
public class TodoDao implements TodoDaoInt {

   @In(required=false) @Out(required=false)
   private Todo todo;

   @PersistenceContext(type=EXTENDED)
   private EntityManager em;

   @DataModel
   private List<Todo> todos;

   @Factory("todos")
   @Begin(join=true)
   public void findTodos() {
      todos = em.createQuery("select t from Todo t").getResultList();
   }

   …
```

Validierung



Validierung

```
@Entity
@Name("todo")
public class Todo implements Serializable {
   …
   private String description;

   public Todo () {
      description ="";
   }
   …
   @NotNull
   @Length(min=2, max=250)
   public String getDescription() {return description;}
}
```

Validierung

```
<tr>
    <td>Description:</td>
    <td>
        <h:inputTextarea id="description"
value="#{todo.description}">
            <s:validate />
        </h:inputTextarea>
        <h:message for="description" />
    </td>
</tr>
```

Validierung JSF

```
<tr>
    <td>Description:</td>
    <td>
      <h:inputTextarea id="description"
value="#{todoBean.todo.description}">
        <f:validateLength minimum="2" maximum="250"/>
      </h:inputTextarea>
      <h:message for="description">
  </td>
  </tr>
```

Validierung erweitert

```
<f:facet name="beforeInvalidField">
  <h:graphicImage styleClass="errorImg"
value="error.png"/></f:facet>
<f:facet name="afterInvalidField">
  <s:message styleClass="errorMsg" /></f:facet>
<f:facet name="aroundInvalidField">
  <s:div styleClass="error"/></f:facet>

<s:validateAll>

 <tr><td>Description:</td>
    <td>
      <s:decorate>
        <h:inputTextarea id="description"
value="#{todo.description}"
        cols="50" rows="10"/>
      </s:decorate>
    </td>
  </tr>

<s:validateAll>
```

Abschliessende Gedanken zu Seam

- Seam ist nicht alleine einsatzfähig – liefert Zusammenhalt für diverse Komponenten-Frameworks
- Setzt Kenntnis der zu integrierenden Technologien voraus
- Performance??

Referenzen

- Dokumentation zu Seam: http://www.seamframework.org/
- Seam Referenz-Dokumentation: http://docs.jboss.com/seam/2.1.1.GA/reference/en-US/html/
- Seam Tutorial mit JSF-Vergleich: http://www.redhat.com/docs/manuals/jboss/jboss-eap-4.2/doc/Getting_Started/index.html

Referenzen

- JBoss Seam – die Webbeans Implementierung (http://www.webbeans.eu/ ISBN 978-3-446-41190-6)
- Seam in Action (http://www.manning.com/dallen/ ISBN 978-1-933988-40-1)