JavaBLEND

# Seam

Pete Muir
JBoss, a Division of Red Hat

http://in.relation.to/Bloggers/Pete

pete.muir@jboss.org

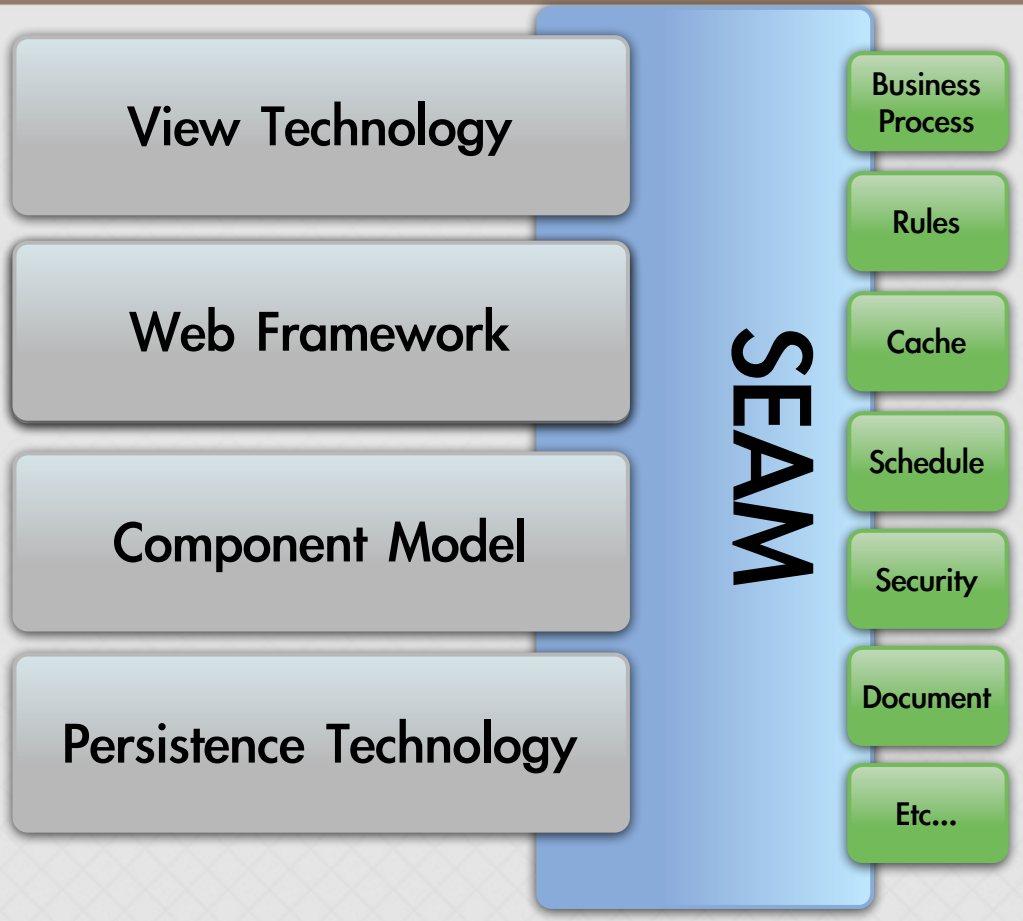
JavaBLEND

## Road Map

- What is Seam?
- Why should I care about atomic conversations?
- How do I quickly build an application with Seam?
- What tools are available?
- The future

JavaBLEND

# Application Stack



JavaBLEND

# Where can I run my app?

WebSphere

Web Logic

Tomcat

Glassfish

JBoss Application Server

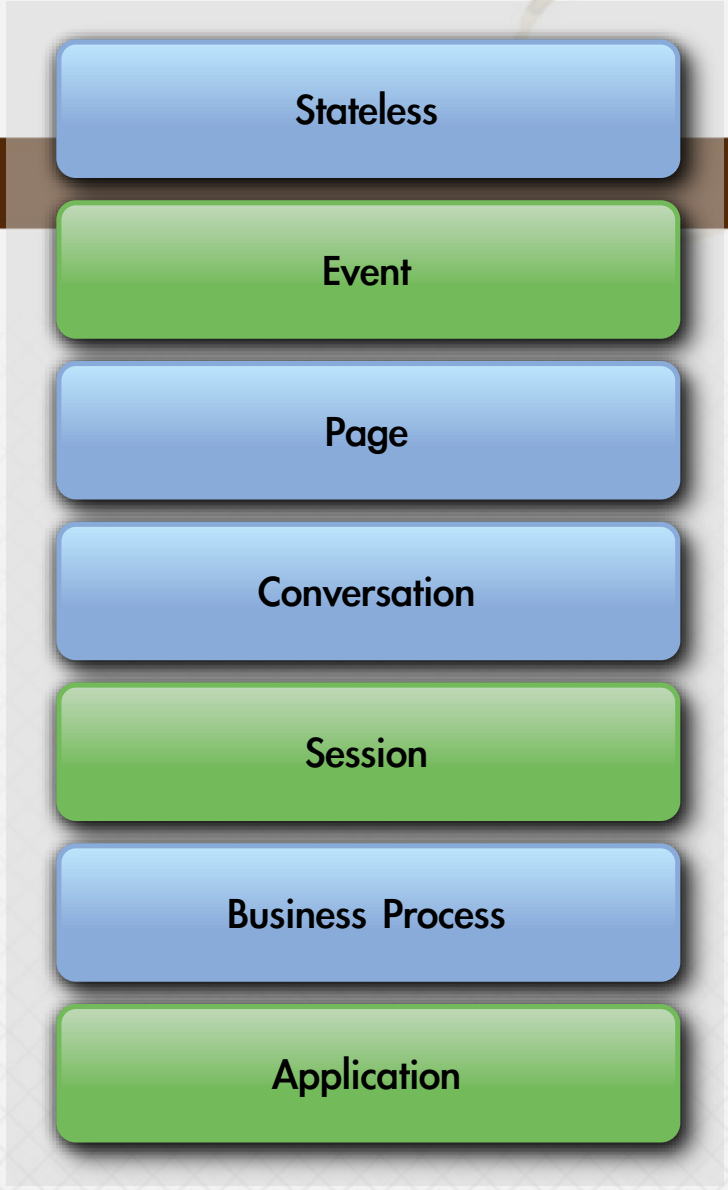
JBoss Enterprise Application Platform

JBoss SOA Platform

JavaBLEND

# Seam is contextual

- Stateful
  - store objects for as long as you need them
  - Seam manages all interactions with the context
  - dependencies are injected and updated every time a component is accessed



JavaBLEND

# The unified component model

```
@Name("discEditor") @Scope(CONVERSATION)
public class DiscEditor {

   @In EntityManager entityManager;

   @In Session mailSession;

   @In @Out Item disc;

   @In(scope=SESSION) User user;

   // Use the components in some way ;

}
```

Seam will store the component in the conversation context

Inject the EntityManager

Inject a JavaMail session

Alias the item from the context, and update the context if the object changes

Specify the context to inject from

JavaBLEND

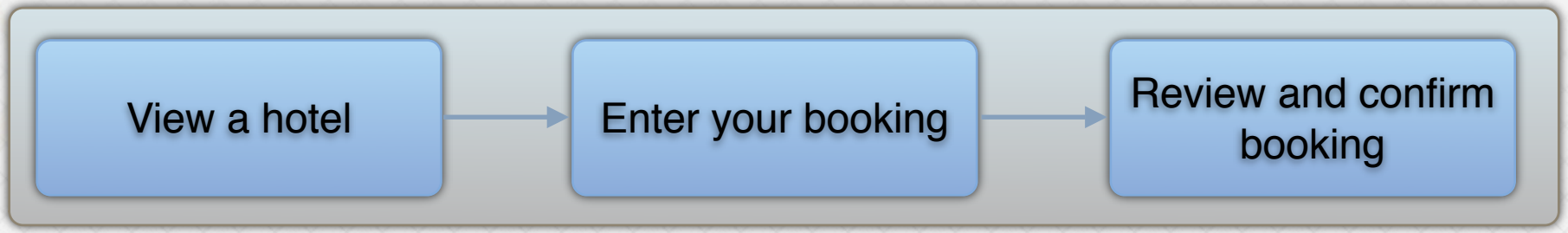
## Road Map

- What is Seam?
- Why should I care about atomic conversations?
- How do I quickly build an application with Seam?
- What tools are available?
- The future

JavaBLEND

# Why do I want an atomic conversation?

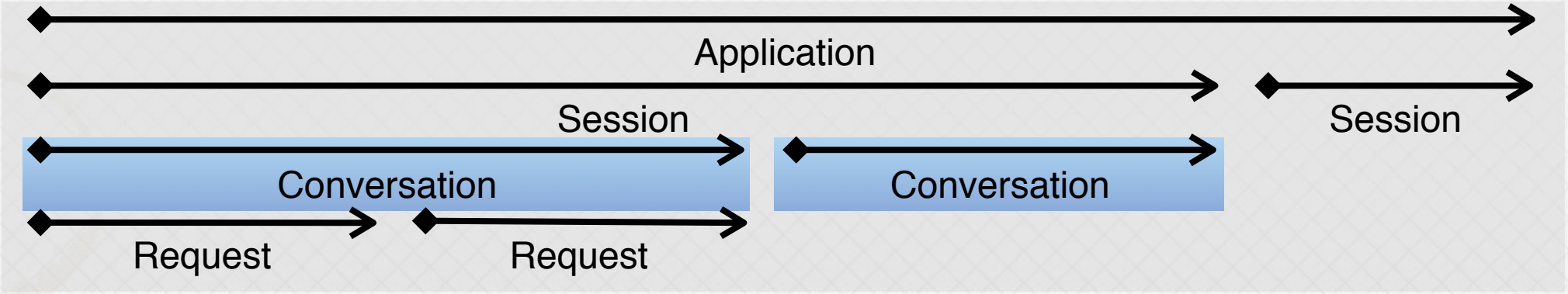
- Here's a common scenario:
  - A user has a task to complete which:
    - spans multiple pages
    - should be able to click cancel or back at any time, no changes made until the user is finished
    - should be able to do the same task in multiple windows



JavaBLEND

# What does Seam provide?

- A conversation scope
  - shorter than the session, longer than a request
  - demarcated by the application developer
  - a conversation per window/tab



JavaBLEND

## What does Seam provide?

- A conversation scoped persistence context keeps entities attached for the entirety of the user's task
  - guarantees object equality
  - allows lazy loading
- An atomic conversation needs to only flush changes at particular points
  - Only flush the persistence context when explicitly instructed to

JavaBLEND

# What does Seam provide?

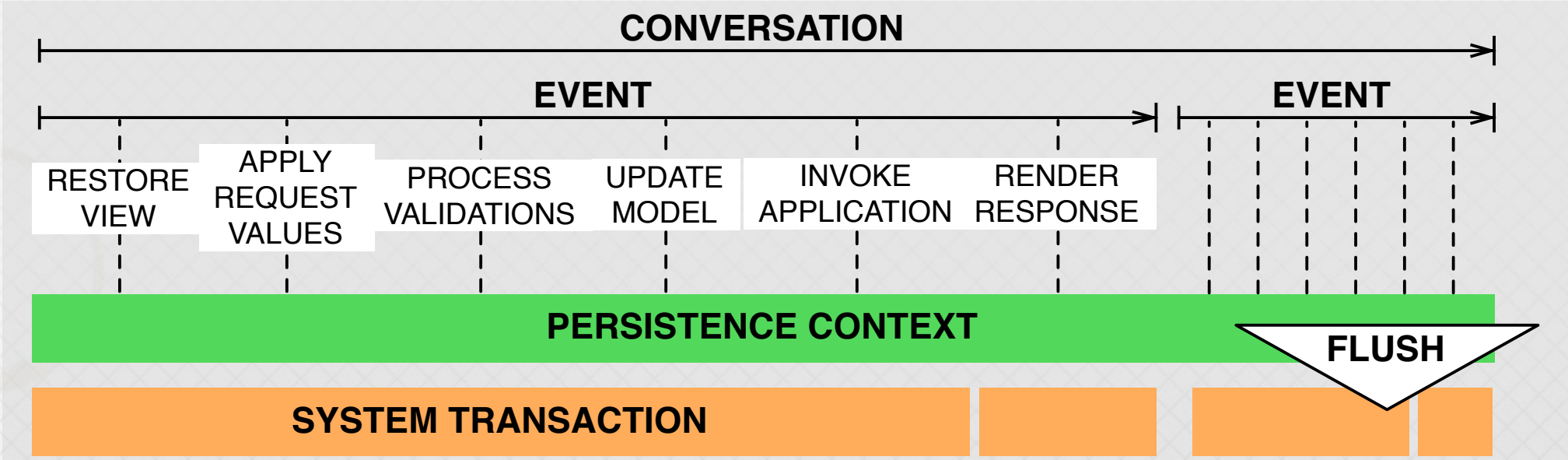
- An atomic conversation needs to only flush changes at particular points
  - Need to use a manual flush mode from Hibernate

```
@Begin(flushMode=MANUAL)
public void editDisc() {
    // Load the item to edit
}

@End
public void saveDisc() {
   entityManager.flush();
}
```

# How do I manage the system transaction then?

- Seam manages the system transaction for you
  - A read-write transaction
  - A read only transaction for rendering the page (slightly better than Open Session in View)



JavaBLEND

## Road Map

- What is Seam?
- Why should I care about atomic conversations?
- How do I quickly build an application with Seam?
- What tools are available?
- The future

JavaBLEND

# Application Framework

- UI orientated controller components
- EntityHome for CRUD

Can define in XML or Java for custom behaviour

```
<fwk:entity-home entity-class="com.acme.Disc" name="discHome"/>
```

```
<s:decorate template="/edit.xhtml">
  <h:inputText value="#{disc.name}" required="true" />
</s:decorate>
<h:commandButton action="#{discHome.update}" value="Save" />
<h:commandButton action="#{discHome.remove}" value="Delete" />
```

Seam provides JSF controls for easy decoration of fields

Bind directly to the entities, no need for DTOs

JavaBLEND

# Application Framework

- EntityQuery for search

```
<fwk:entity-query name="discs" ejbql="select d from Disc d" order="d.name"
max-results="5">
  <fwk:restrictions>
    <value>lower(d.name) like concat(#{exampleDisc.name}, '%'))</value>
  </fwk:restrictions>
</fwk:entity-query>
```

Basic queries can be specified in XML

```
<component name="exampleDisc" class="com.acme.Artist" scope="session" />
```

A prototype, used to bind query parameters between UI and query

JavaBLEND

# Application Framework

- EntityQuery for search

Search criteria

```
<h:form>
  Filter by name:
  <h:inputText value="#{exampleDisc.name}">
    <a:support reRender="artists" event="onkeyup" />
  </h:inputText>
</h:form>
```

Output the results

```
<h:table value="#{discs.dataModel}" var="d" id="discs">
  <h:column>
    <s:link action="disc" value="#{disc.name}">
      <f:param name="discId" value="#{disc.id}" />
    </s:link>
  </h:column>
</h:table>
```

JavaBLEND

```
<h:inputText value="#{disc.name}" required="true">
  <s:validate />
</h:inputText>
```

## Validation

- Need to report validation errors back to the user on the correct field
- BUT normally need to enforce same constraints at the persistence layer and the database

```
@Entity public class Item {

  @Id @GeneratedValue Long id;

  @Length(min=3, max=1000, message="Must be between 3 & 1000 chars")
  String description;
}
```

JavaBLEND

## Road Map

- What is Seam?
- Why should I care about atomic conversations?
- How do I quickly build an application with Seam?
- What tools are available?
- The future

JavaBLEND

## Tooling

- `seam-gen` - command line tool for generating skeleton project and reverse engineering a database schema using Seam Application Framework
- JBoss Developer Studio - Eclipse based IDE
  - For $99 you get a full installer + JBoss EAP
  - Based on the freely available JBoss Tools Eclipse plugins

JavaBLEND

# Demo

JavaBLEND

## Road Map

- What is Seam?
- Why should I care about atomic conversations?
- How do I quickly build an application with Seam?
- What tools are available?
- The future

JavaBLEND

## Flex as a view layer

- A community effort
- Uses Granite Data Services or Blaze Data Services
- Check out a couple of demos at

http://www.rationaldeveloper.com

JavaBLEND

## JSF 2

- Easy Component Creation & Templating
  - Standardizes Facelets
  - No XML needed to create a component
- Built in Ajax support
- Many improvements to JSF
  - lifecycle (performance!)
  - error handling
  - navigation

JavaBLEND

## What else?

- Seam 2.1 release candidate in the next week or two
  - Friendly URLs
  - Identity Management
  - ACL style permissions
  - Wicket support
  - Excel reporting module
  - Support for JAX-RS (REST)

JavaBLEND

# Q&A

http://in.relation.to/Bloggers/Pete

http://www.seamframework.org