

Construindo aplicações HTML5 em Java

Eder Ignatowicz
Sr. Software Engineer
JBoss by Red Hat

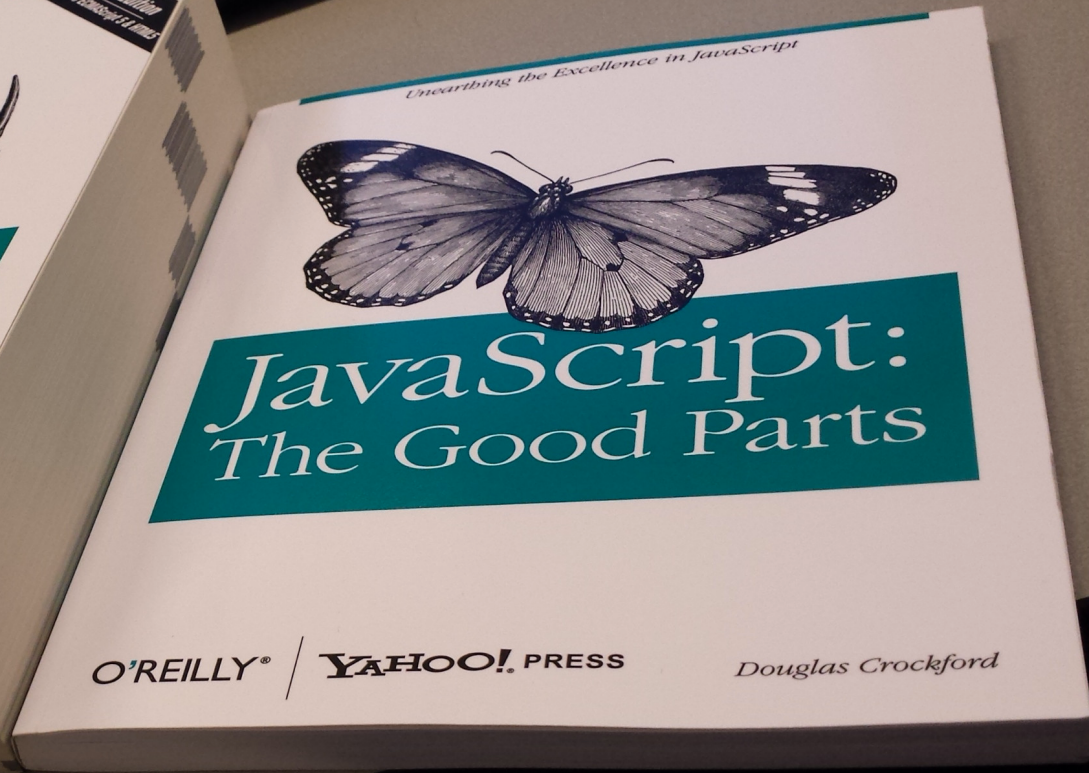
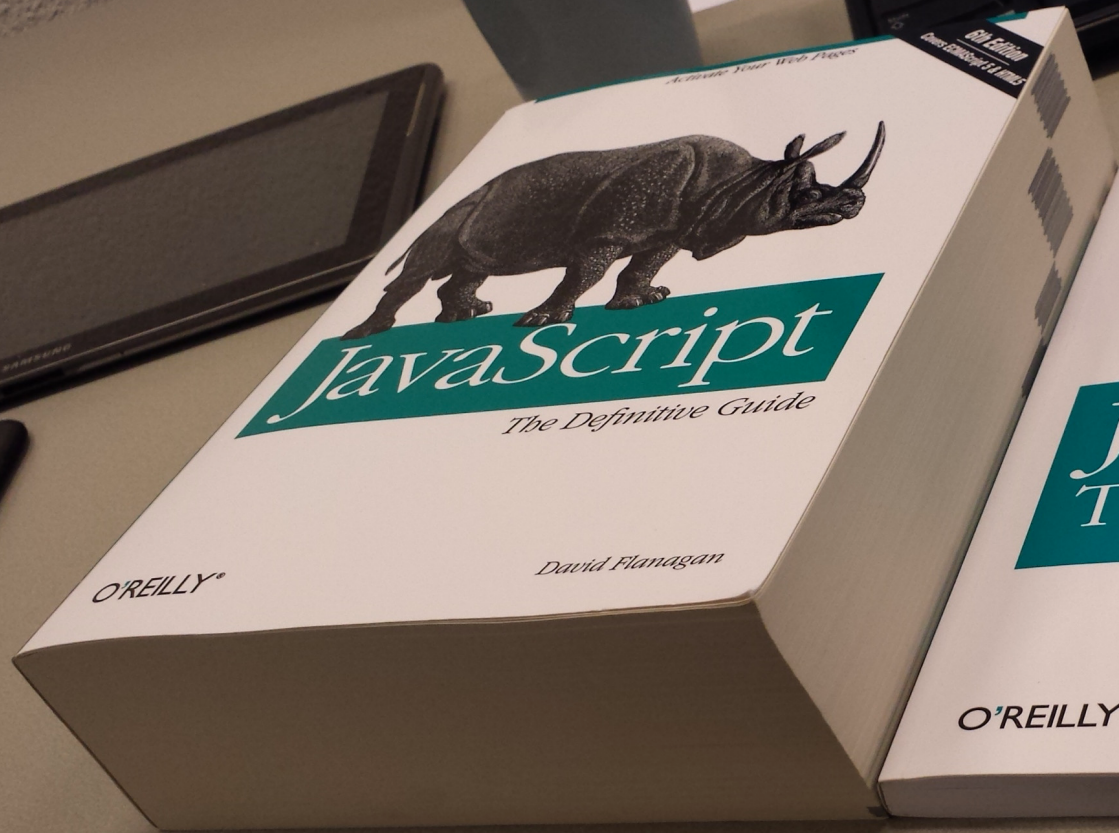
Java <3

GWT < 3

YODA SAID,

"YOU MUST UNLEARN, WHAT YOU HAVE LEARNED."





Como construimos os workbenches do Drools e do jBPM

Eder Ignatowicz
Sr. Software Engineer
JBoss by Red Hat



~600k linhas de client code
~20 repositórios



uberfire



UX + Dev

```
<html>
<head>
  <title>First JSP</title>
  <link rel="stylesheet" href="number-styles.css">
</head>
<body>
  <%
double num = Math.random();
if (num > 0.95) {
%>
  <h2 class="response">You'll have a lucky day!</h2>
  <p class="number lucky">(<%= num %>)</p>
  <%
} else {
%>
  <h2 class="response">Well, life goes on ... </h2>
  <p class="number unlucky" >(<%= num %>)</p>
  <%
}
%>
  <a href="<%= request.getRequestURI() %>"><h3>Try Again</h3></a>
</body>
</html>
```

```

<header id="header">
  <h1>todos</h1>
  <form id="todo-form" ng-submit="addTodo()">
    <input id="new-todo" placeholder="What needs to be done?" ng-model="newTodo" autofocus>
  </form>
</header>
<section id="main" ng-show="todos.length" ng-cloak>
  <input id="toggle-all" type="checkbox" ng-model="allChecked" ng-click="markAll(allChecked)"> <label
for="toggle-all">Mark all as complete</label>
  <ul id="todo-list">
    <li ng-repeat="todo in todos | filter:statusFilter track by $index"
ng-class="{completed: todo.completed, editing: todo == editedTodo}">
    <div class="view">
    <input class="toggle" type="checkbox" ng-model="todo.completed"> <label ng-dblclick="editTodo(todo)">{{todo.title}}
</label>
    <button class="destroy" ng-click="removeTodo(todo)"></button>
    </div>
    <form ng-submit="doneEditing(todo)">
    <input class="edit" ng-trim="false" ng-model="todo.title" todo-escape="revertEditing(todo)"
ng-blur="doneEditing(todo)" todo-focus="todo == editedTodo">
    </form>
    </li>
  </ul>
</section>
<footer id="footer" ng-show="todos.length" ng-cloak>
  <span id="todo-count"><strong>{{remainingCount}}</strong> <ng-pluralize count="remainingCount"
when="{ one: 'item left', other: 'items left' }"></ng-pluralize> </span>
  <ul id="filters">
    <li><a ng-class="{selected: status == ''}" href="#/">All</a></li>
    <li><a ng-class="{selected: status == 'active'}" href="#/active">Active</a></li>
    <li><a ng-class="{selected: status == 'completed'}" href="#/completed">Completed</a></li>
  </ul>
  <button id="clear-completed" ng-click="clearCompletedTodos()" ng-show="completedCount">Clear completed
({{completedCount}})</button>
</footer>
</section>
<footer id="info">
<p>Double-click to edit a todo</p>
<p>
Part of <a href="http://todomvc.com">TodoMVC</a>
</p>
</footer>

```

```
// https://github.com/facebook/react/blob/master/docs/docs/tutorial.md
```

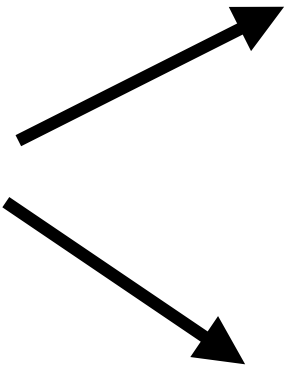
```
var Comment = React.createClass({
  rawMarkup: function() {
    var rawMarkup = marked(this.props.children.toString(), {sanitize: true});
    return { __html: rawMarkup };
  },

  render: function() {
    return (
      <div className="comment">
        <h2 className="commentAuthor">
          {this.props.author}
        </h2>
        <span dangerouslySetInnerHTML={this.rawMarkup()} />
      </div>
    );
  }
});
```

ERRAI UI

```
<form id="login">
  <legend>Log in to your account</legend>
  <label for="username">Username</label>
  <input id="username" type="text" placeholder="Username">
  <label for="pass">Password</label>
  <input id="pass" type="password" placeholder="Password">
  <button id="submit">Log in</button>
  <button>Cancel</button>
</form>
```


.java



javac



.class

src/main/java

HelloWorld.gwt.xml

client

HelloWorldClient.java

shared

HelloWorldEntity.java

server

HelloWorldService.java

src/main/webapp

HelloWorld.html

WEB-INF

web.xml

src/main/resources

ErraiApp.properties

Shared Codebase

```
@Bindable
@Entity
@NamedQuery(
    name = "allComplaints",
    query =
        "SELECT c FROM UserComplaint c ORDER BY c.id" )
public class UserComplaint {

    @Id
    @GeneratedValue( strategy = GenerationType.AUTO )
    private Long id;

    @Version
    private Long version;

    @NotNull
    private String name;

    @Email
    private String email;

    private String text;

    ...
}
```

JPA in the Browser

```
@Templated
@Page
public class ComplaintForm {

    @Inject
    private EntityManager em;

    @EventHandler("submit")
    private void onSubmit(ClickEvent e) {
        em.persist(complaint);
        em.flush();
    }
}
```

Data Binding

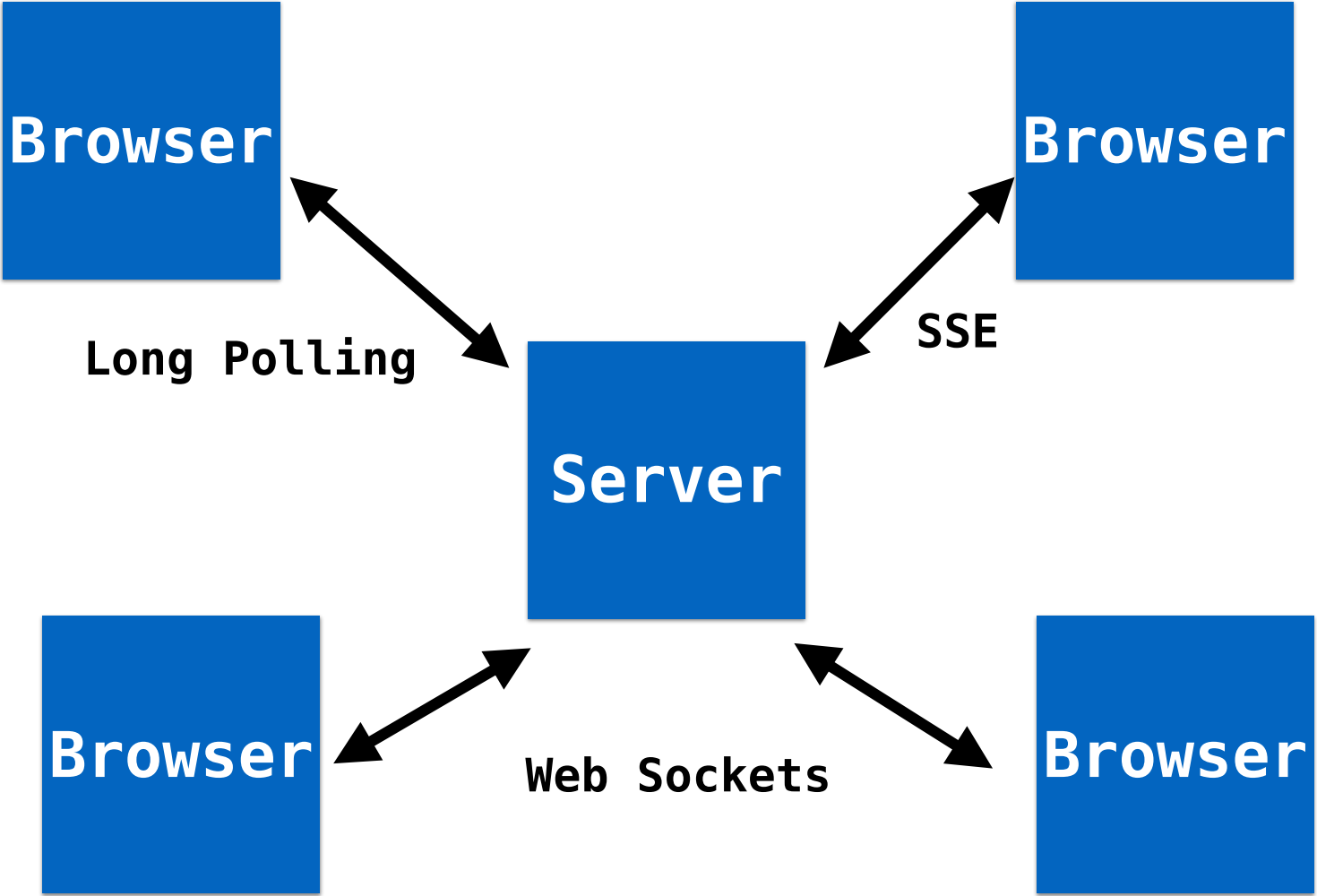
```
@Templated
@Page
public class ComplaintForm {
    @Inject
    @Bound
    @DataField
    private TextBox email;

    @Inject
    @Bound
    @DataField( "name" )
    private TextBox username;

    @Inject
    @Bound
    @DataField
    private TextArea text;

    @Inject
    @Model
    private UserComplaint complaint;

    @EventHandler( "submit" )
    private void onSubmit( ClickEvent e ) {
        sendToServer( complaint );
    }
}
```

JAX-RS

Client

@Inject

```
private Caller<ContactStorageService> contactService;
```

```
contactService.call( (List<Contact> contacts)  
-> addAll(contacts) ).getAllContacts();
```

Shared

@Path("/**contact**")

```
public interface ContactStorageService {
```

@GET

@Produces("**application/json**")


```
List<Contact> getAllContacts();
```

IDE Support








Refactoring

```
@Templated("#main")
@Page(path="share")
public class SharePage extends Composite {
    @Inject private @DataField Label overallErrorMessage;
    @Inject private @DataField TextBox email;
    @Inject private @DataField Button shareButton;
    @Inject private Caller<ShareService> shareService;
    @Inject private TransitionTo<ToDoListPage> todoListPageLink;

    @PostConstruct
    private void init() {
        overallErrorMessage.setVisible(false);
    }
}
```

 overallErrorMessage cannot be resolved

7 quick fixes available:

-  [Create local variable 'overallErrorMessage'](#)
-  [Create field 'overallErrorMessage'](#)
-  [Create parameter 'overallErrorMessage'](#)
-  [Create class 'overallErrorMessage'](#)
-  [Create constant 'overallErrorMessage'](#)
-  [Change to 'overallErrorMessage'](#)
-  [Fix project setup...](#)

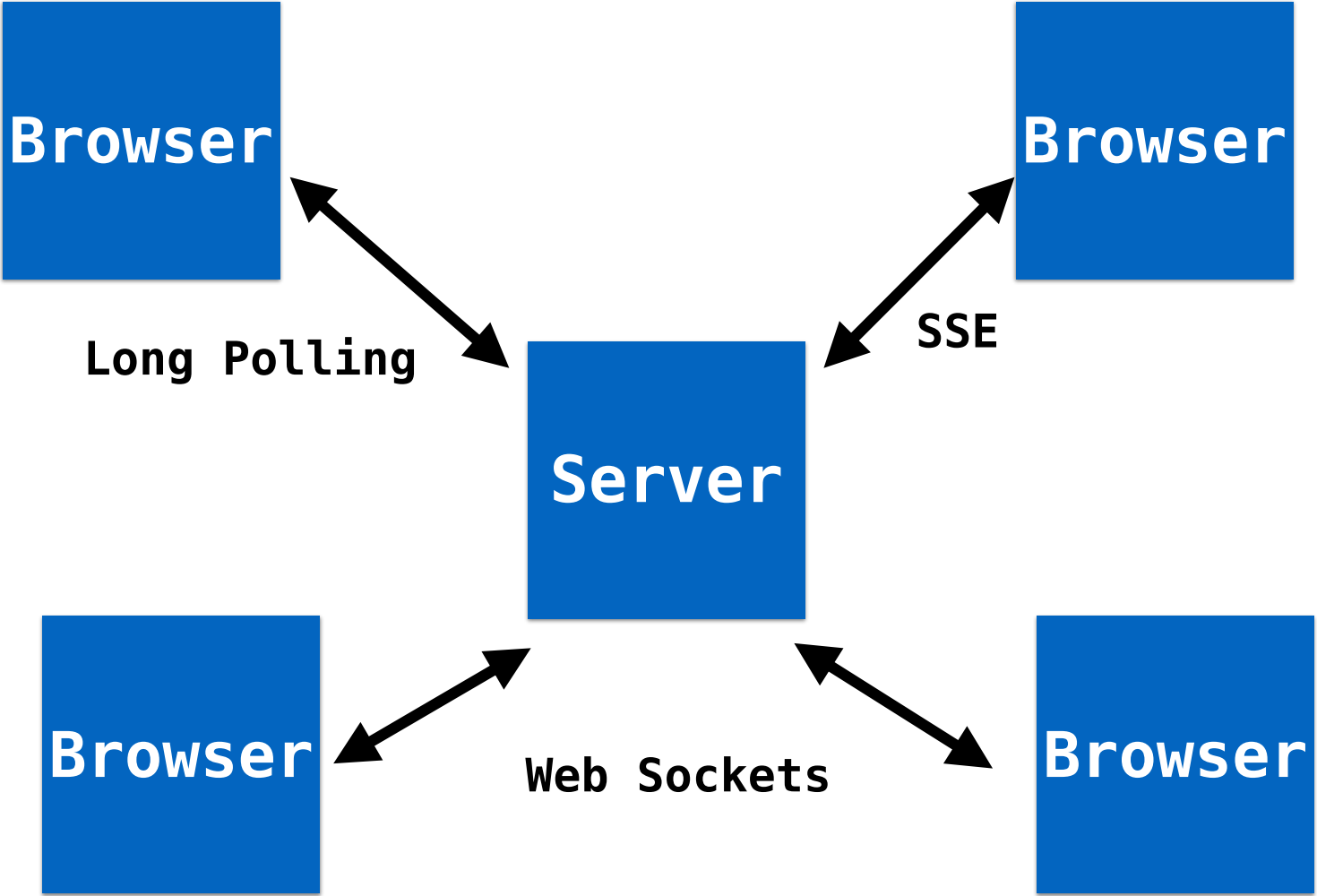


CDI

<3

CDI

no browser



@Dependent

@Templated

```
public class ProjectsView implements ProjectsPresenter.View {
```

```
    private ProjectsPresenter presenter;
```

```
    @Inject
```

```
    Document document;
```

```
    @Inject
```

```
    @DataField( "projects-view" )
```

```
    Div view;
```

```
    @Inject
```

```
    @DataField( "new-project" )
```

```
    Button newProject;
```

```
    @Inject
```

```
    @DataField( "projects-list" )
```

```
    UnorderedList projectsList;
```

```
    @Inject
```

```
    Instance<ProjectItem> projects;
```

Observando eventos no
Client
Server
Ambos

```
public void onRemoteCreated(final @Observes
@Operation(CREATE) ContactOperation contactOperation) {
    if (sourceIsNotThisClient(contactOperation)) {
        binder.getModel().add(contactOperation.getContact());
    }
}
```

JSInterop

```
package com.acme;
```

```
@JsType
```

```
class Foo {
```

```
    public int x;
```

```
    public int y;
```

```
    public int sum() {
```

```
        return x + y;
```

```
    }
```

```
}
```

```
var foo = new com.acme.Foo();
```

```
foo.x = 40;
```

```
foo.y = 2;
```

```
foo.sum(); // will return 42!
```

```
// in JavaScript
com = { acme: {} };
    com.acme.Foo = function() {
        this.x = 40;
        this.y = 2;
    };
    com.acme.Foo.prototype.sum = function()
        { return this.x + this.y; };
```

```
// in Java
package com.acme;

@JsType(isNative = true)
class Foo {
    public int x;
    public int y;

    public native int sum();
}

class FooMain {
    public static void main() {
        Foo foo = new Foo();
        foo.sum(); // will return 42!
        foo.x = 50;
        foo.y = 5;
        foo.sum(); // will return 55!
    }
}
```



uberfire



Framework para construção de grandes aplicações
estilo “workbench” e consoles web

“Eclipse Like”

Modelo de Programação “Opinativo”

model

runtime plugin

window mgmt

cdi

nsew

template

...

bus

ui

mvp

errai

gwt

cluster

index / search

vfs

metadata

preferences

security

***RUNTIME
PLUGINS***

APPS

WIRES

**UBERFIRE
EXTENSIONS**

***FORM
MODELER***

core

GUVNOR

**BUILD &
PROVISIONING**

**KIE
EXTENSIONS**

SOCIAL

DASHBUILDER

DEMO

(feature preview)

Java EE + GWT

<3

Open Source

<3



RED HAT DEVELOPERS

LEARN MORE.

CODE MORE.

SHARE MORE.

Obrigado <3

<http://erraiframework.org>

<http://uberfireframework.org>

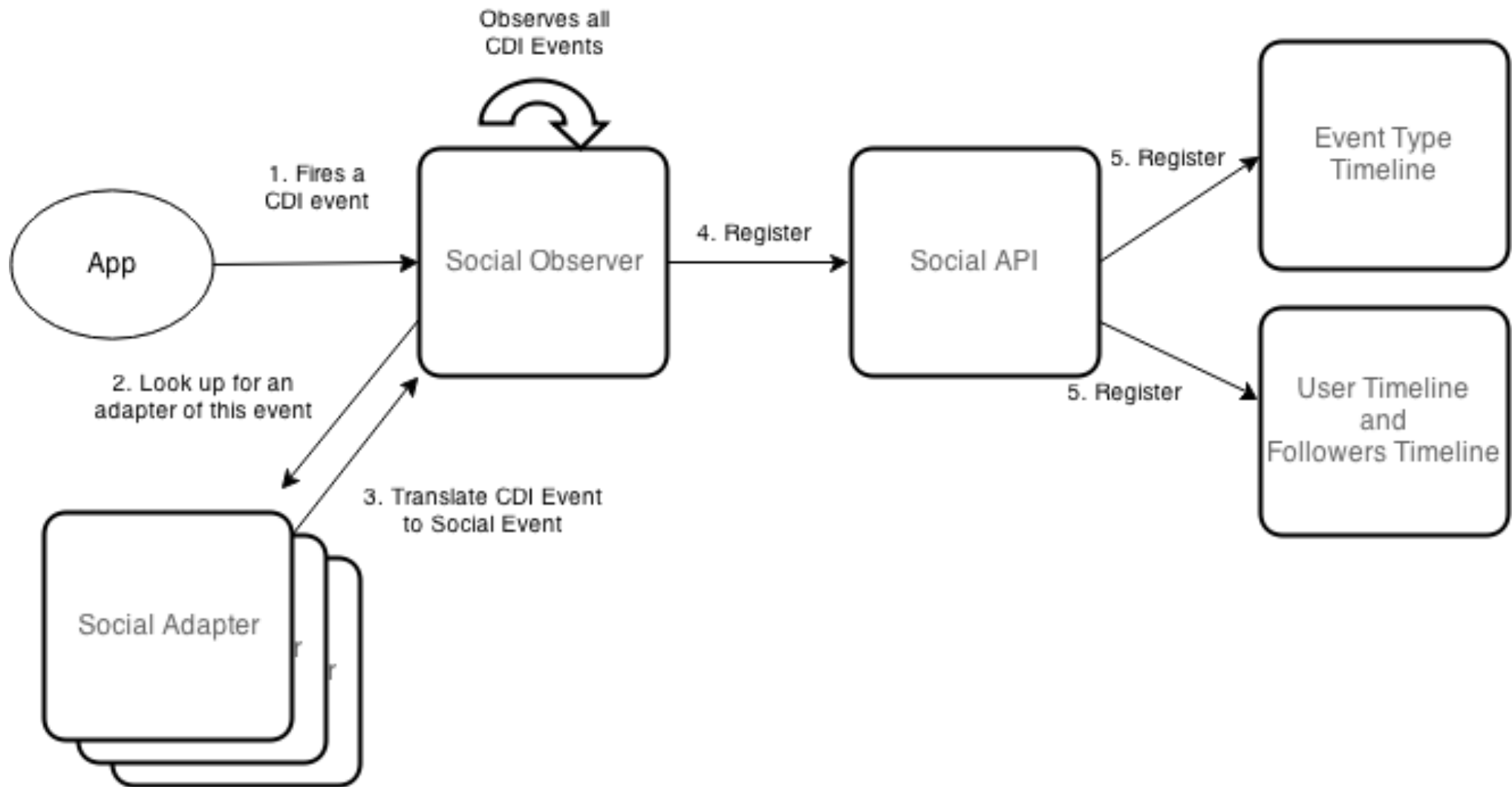
IRC: #errai #uberfire na freenode

Eder Ignatowicz
@ederign



redhat®

Social + CDI



Social Activities Feed

UberFire Showcase x Social Activities Feed x +

127.0.0.1:8888/org.uberfire.UberfireShowcase/social/NEW_REPOSITORY_EVENT ☆ Google



Subscribe to this feed using

Always use Live Bookmarks to subscribe to feeds.

Subscribe Now

Social Activities Feed

NEW_REPOSITORY_EVENT

July 16, 2014 at 2:29 PM

system SocialActivitiesEvent{timestamp=Wed Jul 16 14:29:36 BRT 2014, user=system, type=NEW_REPOSITORY_EVENT, created repository uf-playground }

NEW_REPOSITORY_EVENT

July 16, 2014 at 3:43 PM

system SocialActivitiesEvent{timestamp=Wed Jul 16 15:43:47 BRT 2014, user=system, type=NEW_REPOSITORY_EVENT, created repository sample1 }

NEW_REPOSITORY_EVENT

July 16, 2014 at 3:44 PM

system SocialActivitiesEvent{timestamp=Wed Jul 16 15:44:00 BRT 2014, user=system, type=NEW_REPOSITORY_EVENT, created repository sample2 }