

# SERVER-SIDE FRONTENDS MIT MVC 1.0

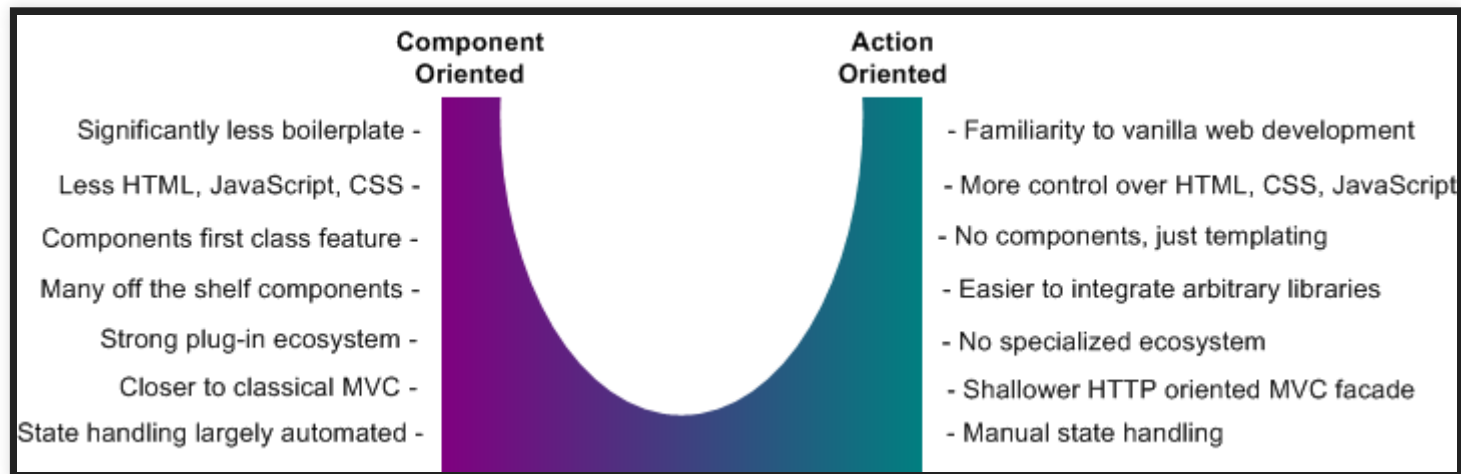
Gregor Tudan - @cofinpro\_greg



*Sind Frontends vom Server nicht tot?*

*Aber wir haben doch JSF und JSP!*

# ACTION VS. COMPONENT BASED



Quelle: Oracle

*Dann nimm doch Play, Spring MVC oder Struts!*

# ZIELE VON MVC

1. Aufsetzen auf existierende Java-EE Technologien
2. *Basisfunktionen* zum Bauen von MVC-Applications
3. JAX-RS als Matching- & Binding-Layer
4. Support für JSP und Facelets als View-Languages

# DIE KURZE VERSION DER GESCHICHTE

# 2014

- JSR 371 wird vorgestellt
- Referenzimplementierung "Ozark"
- Community Ballot: MVC auf Platz 5 der gewünschten Features für JavaEE8



A pixelated landscape with a path leading to a horizon. The path is light-colored and leads from the bottom center towards the horizon. The landscape is composed of various shades of brown, tan, and yellow, suggesting a dry or autumnal environment. The sky is a uniform light gray. The overall style is reminiscent of a low-resolution digital painting or a heavily pixelated photograph.

2015

# 2016

- JavaOne: Community-Umfrage zu JavaEE 8
- MVC: Platz 21
- wird aus JavaEE 8 gestrichen

# JANUAR 2017

*Die Community soll's richten*

*Java EE ohne MVC: Oracle überträgt  
JSR 371 an die Community*

*Jax-Center*

# 2017

- Oracle übergibt JavaEE an Eclipse
- aus JavaEE wird EE4J
- MVC-Spec und Ozark wechseln zur Eclipse Foundation

# 2018

- JSR 371 wird verabschiedet
- Ozark soll umbenannt werden

Schluss jetzt! Zeig mir Code!

# CONTROLLER SIND REST- ENDPUNKTE

```
@Path("/")
@Controller
public class Controller {

    @GET
    public String view(String name) {
        return "hello.html";
    }
}
```

# MODEL BINDING

```
@Path("/")
@Controller
public class Controller {

    @Inject
    private Models model;

    @GET
    public String view(@QueryParam("name") String name) {
        model.put("name", name);
        return "hello.html";
    }
}
```



# TEMPLATES

```
<html>
<head>
  <title>Hello World</title>
</head>
<body>
  <h1>Hello ${name}</h1>
</body>
</html>
```

# TEMPLATE ENGINES

- Java-Server-Pages
- Facelets (hello JSF-Developers!)
- Thymeleaf
- Handlebars
- AsciiDoc
- ...

# WEITERE FEATURES

- Validation
- Internationalization
- Exception Mappers & Error Bindings
- Cross-site Request Forgery Protection
- XSS-Escaping
- Redirect-Handling
- Events
- URL-Resolving

# WO KANN ICH DAS BENUTZEN?

- JAX-RS Extension
  - Jersey, Resteasy & CXF
- Wildfly-Extension
- Thorntail/WF-Swarm-Fraction

# GETTING STARTED

- [MVC-Spec.org/learn](https://mvc-spec.org/learn)
- Wildfly-Swarm

**COFINPRO**

**CONTRIBUTE!**