

2.– 5. September 2013
in Nürnberg



Herbstcampus

Wissenstransfer
par excellence

Atmosphere Framework

Einheitliche API für Asynchrone Web Kommunikation

Walery Strauch

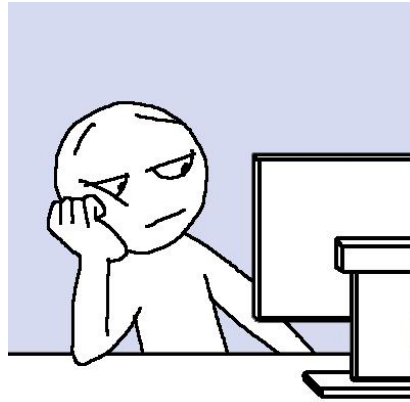
Freiberuflicher Softwareentwickler

HTTP

- Request
- Response
- Request
- Response
- Request
- Response
- Request
- Response
- Request
- Response
- Request
- Response

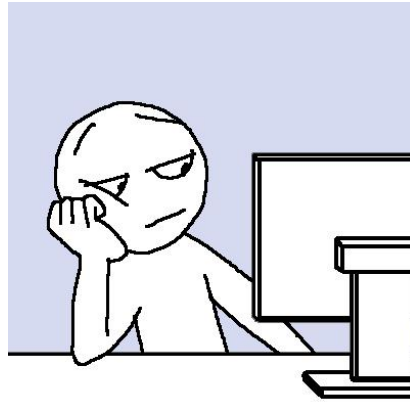
HTTP

- Request
- Response
- Request
- Response
- Request
- Response
- Request
- Response
- Request
- Response
- Request
- Response



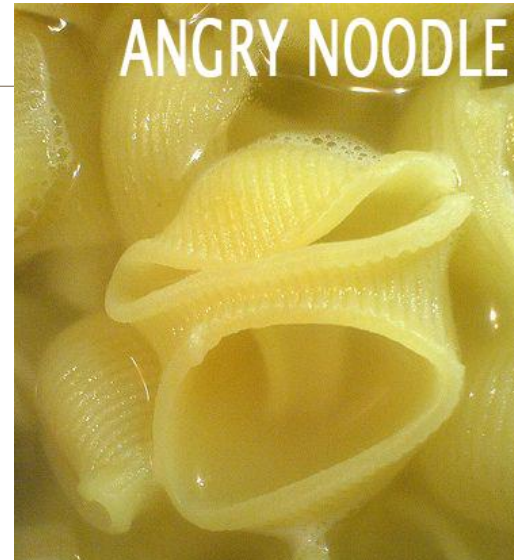
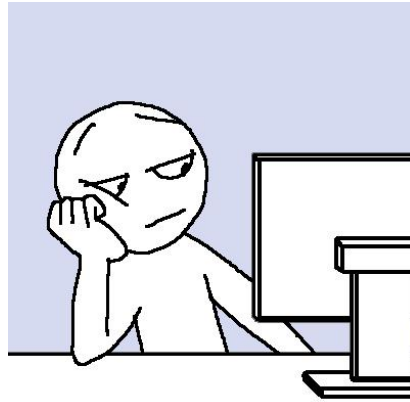
HTTP

- Request
- Response
- Request
- Response
- Request
- Response
- Request
- Response
- Request
- Response
- Request
- Response



HTTP

- Request
- Response
- Request
- Response
- Request
- Response
- Request
- Response
- Request
- Response
- Request
- Response



Lösungen

Lösungen - Im „BullshitBingoStyle“

Lösungen - Im „BullshitBingoStyle“

- SSE (Server Sent Events)
- Http-Streaming
- JSONP
- Long-Polling
- Ajax-Push
- Reverse Ajax
- Two-way-web
- Hidden iframe
- Java Applets
- Flash

Resultat => Websockets

Websockets

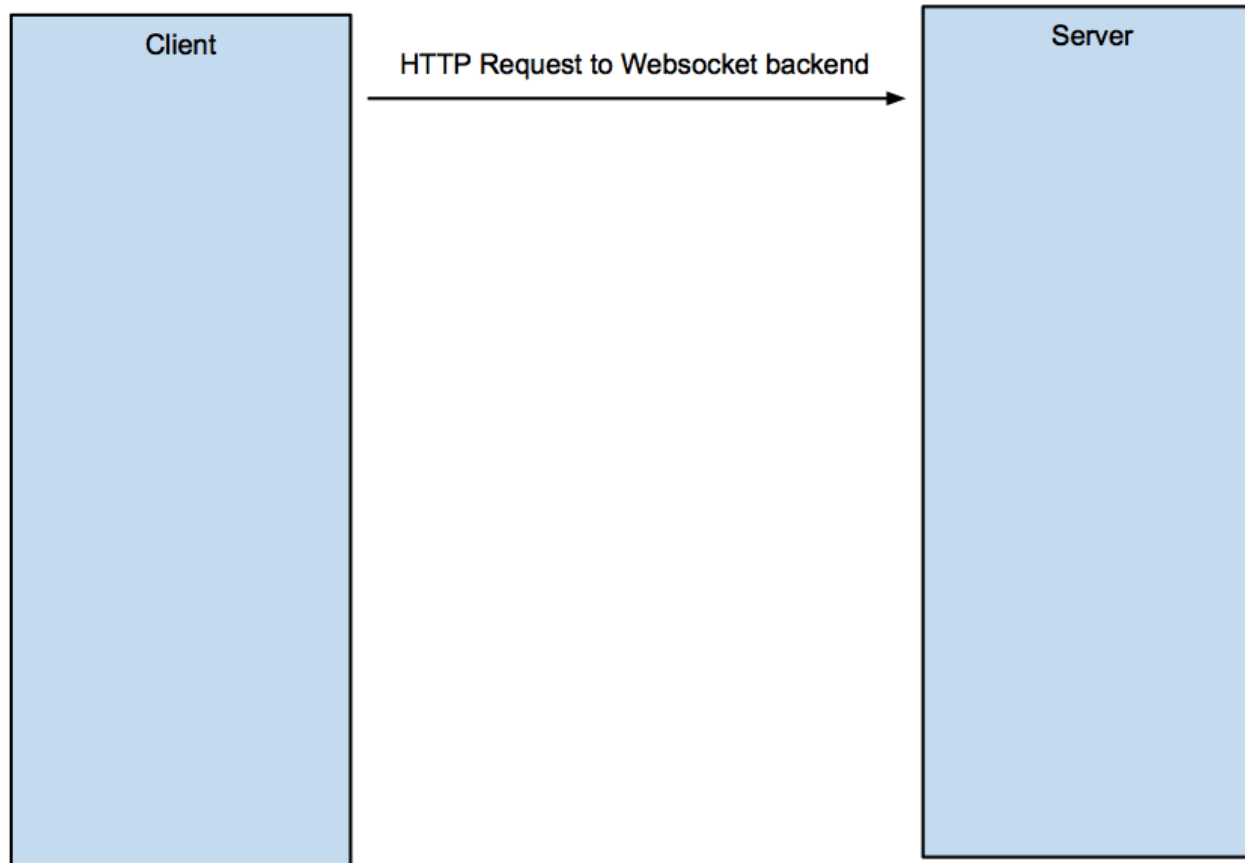


Websockets - Im „KlugscheisserStyle“

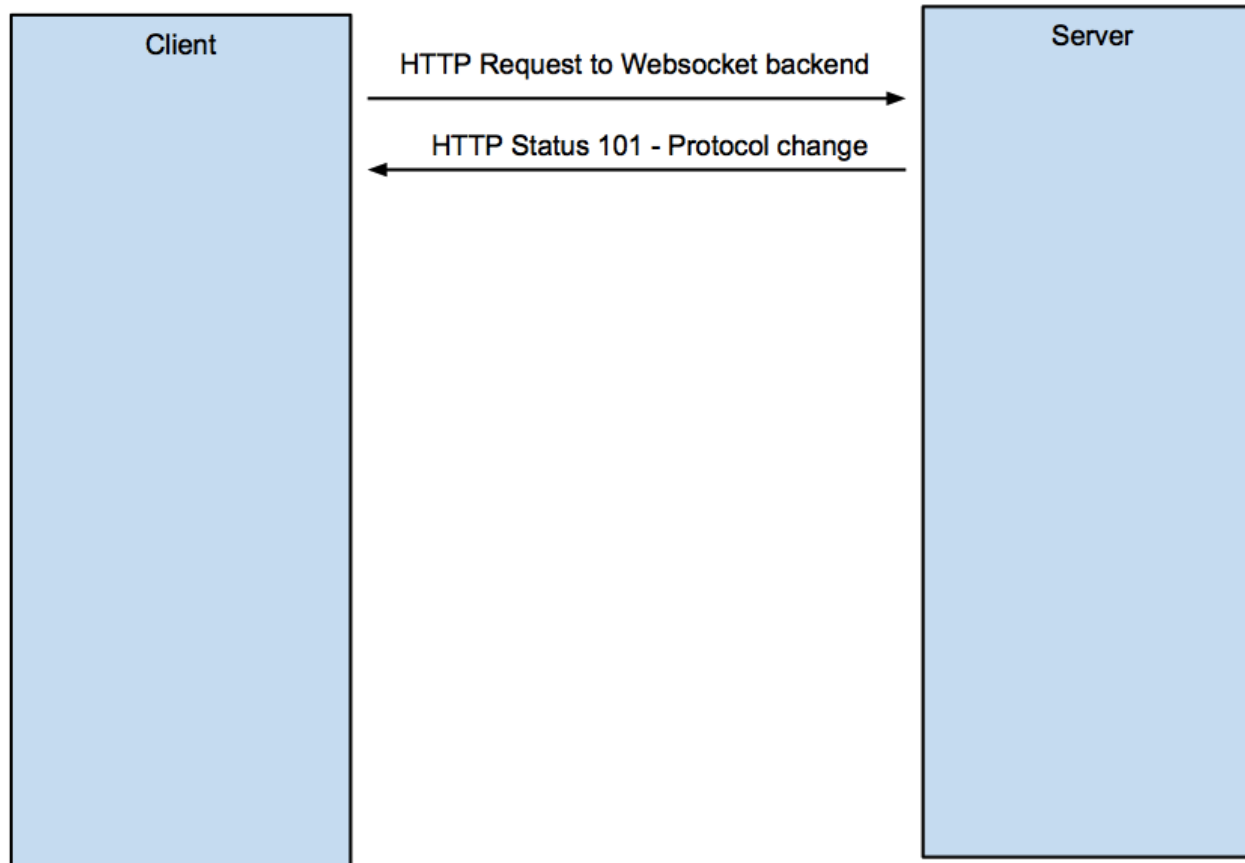
- RFC 6455 - Dez 2011



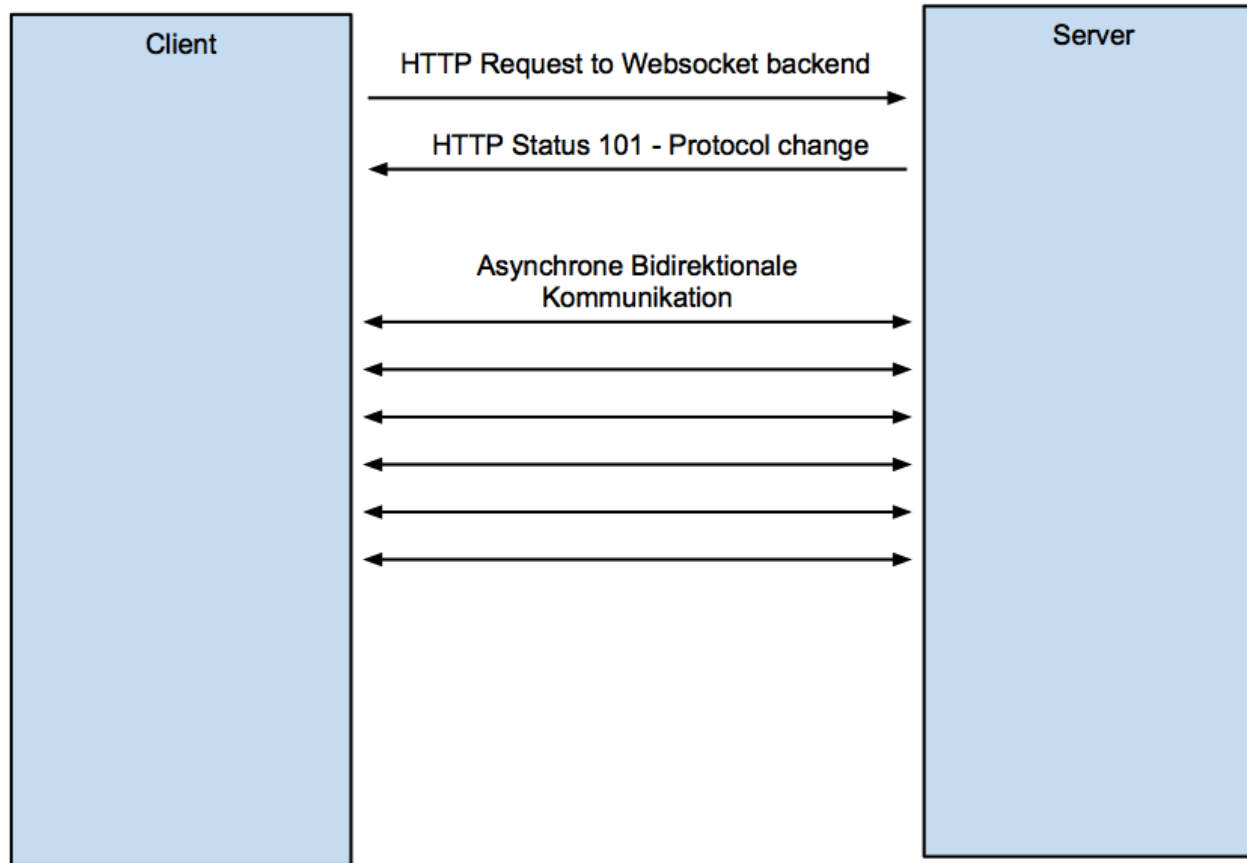
Websockets



Websockets



Websockets



Websockets – Im „TroubleStyle“

Websockets – Im „TroubleStyle“

- Jetty $\geq 7.x$
- Tomcat $> 7.0.26$
- Glassfish $\geq 3.1.2$
- JBoss AS $\geq 6.x$

Websockets – Im „TroubleStyle“

- Jetty $\geq 7.x$
- Tomcat $> 7.0.26$
- Glassfish $\geq 3.1.2$
- JBoss AS $\geq 6.x$

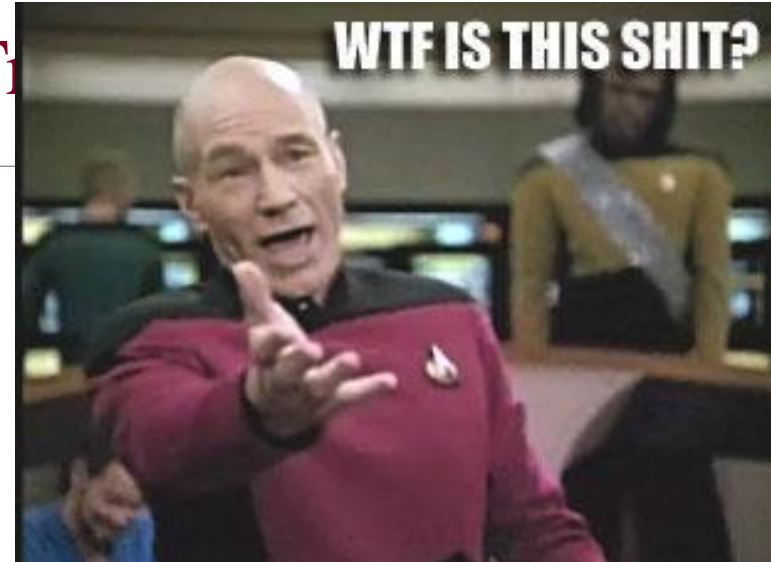
- Firefox $\geq 9.x$
- Chrome $\geq 13.x$
- IE $\geq 10.x$
- Opera $\geq 11.x$
- Safari $\geq 5.x$

ApplicationServer – Im „TroubleStyle“

- Jetty 7.0 – Websocket Support
 - Jetty 7.2.x – API break
 - Jetty 7.4.5 – API break
 - Jetty 8.0.2 – API break
-
- Glassfish 3.0 – Websocket API Support
 - Glassfish 3.1.2 – API break

ApplicationServer – Im „T

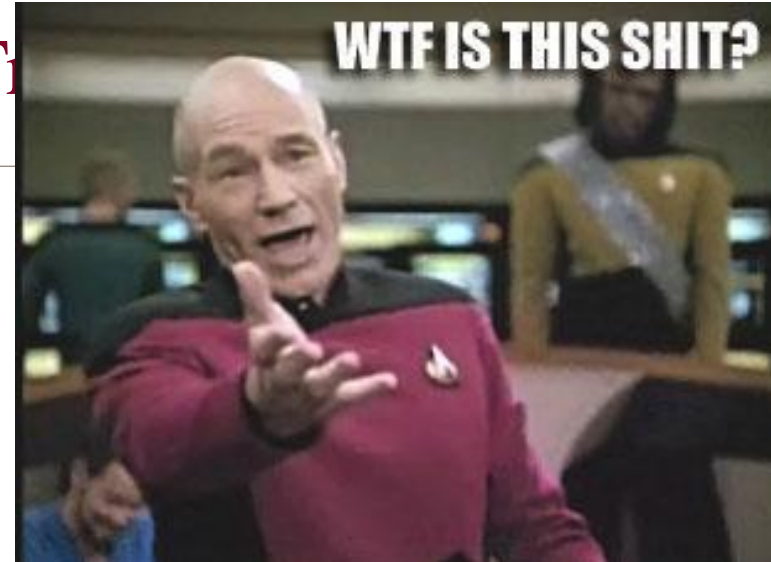
- Jetty 7.0 – Websocket Support
 - Jetty 7.2.x – API break
 - Jetty 7.4.5 – API break
 - Jetty 8.0.2 – API break
-
- Glassfish 3.0 – Websocket API Support
 - Glassfish 3.1.2 – API break



ApplicationServer – Im „T

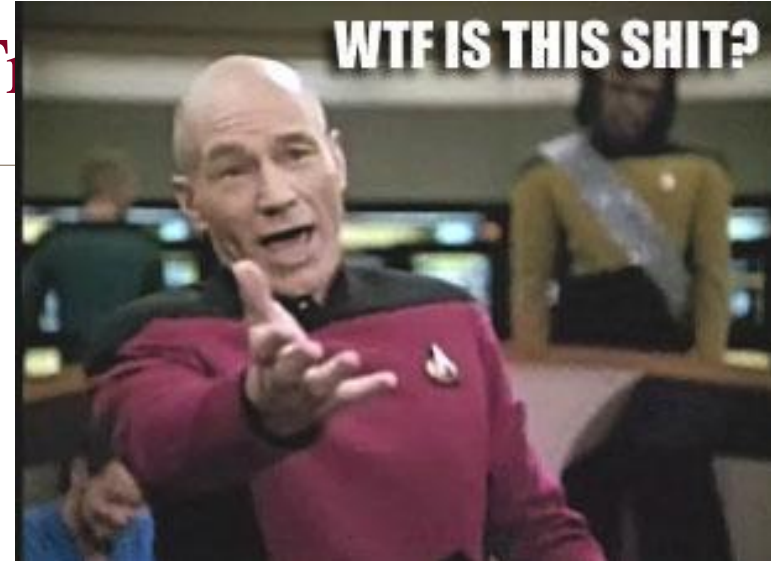
- Jetty 7.0 – Websocket Support
- Jetty 7.2.x – API break
- Jetty 7.4.5 – API break
- Jetty 8.0.2 – API break

- Glassfish 3.0 – Websocket API Support
- Glassfish 3.1.2 – API break



ApplicationServer – Im „T

- Jetty 7.0 – Websocket Support
 - Jetty 7.2.x – API break
 - Jetty 7.4.5 – API break
 - Jetty 8.0.2 – API break
-
- Glassfish 3.0 – Websocket API Support
 - Glassfish 3.1.2 – API break



Atmosphäre

- Einheitliche API
 - Auf dem Server
 - Auf dem Client
- Fallback falls verfahren nicht unterstützt wird
 - Auf dem Server
 - Auf dem Client
- Läuft als Servlet und unter Netty, Vert.x und Play!
- Viele weitere sonstige Features
 - Die nicht gezeigt werden weil das ein Lightning Talk ist ;)

Atmosphäre

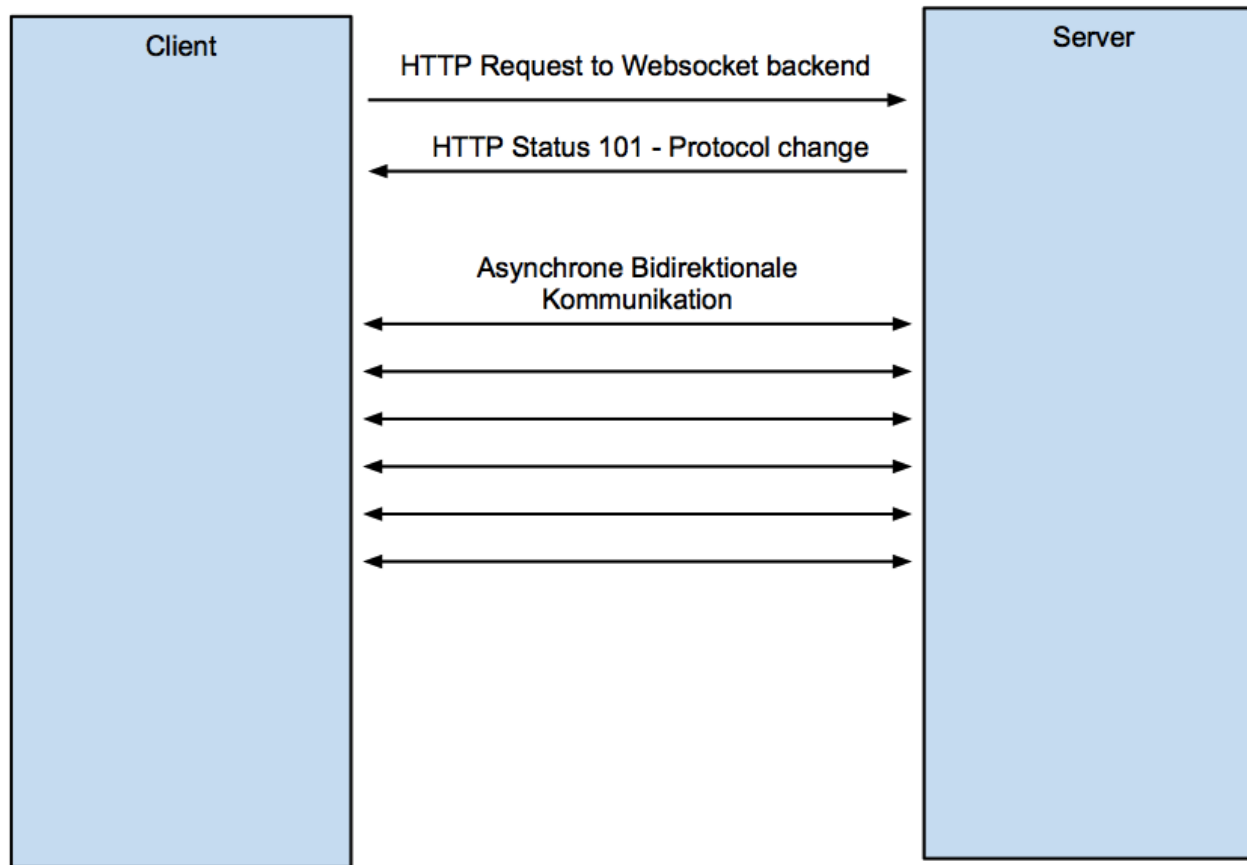
- Einheitliche API
 - Auf dem Server
 - Auf dem Client
- Fallback falls verfahren nicht unterstützt wird
 - Auf dem Server
 - Auf dem Client
- Läuft als Servlet und unter Netty, Vert.x und Play!
- Viele weitere sonstige Features
 - Die nicht gezeigt werden weil das ein Lightning Talk ist ;)



Das will ich sehen



Atmosphere – nochmal ganz kurz:



Atmosphere – Server (Java Servlet)

Atmosphere – Server (Java Servlet)

```
class Echo {
```

```
}
```

Atmosphere – Server (Java Servlet)

```
@ManagedService(path = '/ws/echo')  
class Echo {
```

```
}
```

Atmosphere – Server (Java Servlet)

```
@ManagedService(path = '/ws/echo')
class Echo {

    @Ready
    void onReady(AtmosphereResource r) {

    }

    @Disconnect
    void onDisconnect(AtmosphereResourceEvent event) {

    }

    @Message
    String onMessage(String msg) {

    }
}
```

Atmosphere – Server (Java Servlet)

```
@ManagedService(path = '/ws/echo')
class Echo {
    def broadcaster

    @Ready
    void onReady(AtmosphereResource r) {
        broadcaster = r.getBroadcaster()

        broadcaster.broadcast("user established new connection");
    }

    @Disconnect
    void onDisconnect(AtmosphereResourceEvent event) {
        // ...
    }

    @Message
    void onMessage(String message) {
        "echo: ${msg}"
    }
}
```

Atmosphere – Server (web.xml)

```
<servlet>
  <servlet-name>AtmosphereServlet</servlet-name>
  <servlet-class>
    org.atmosphere.cpr.AtmosphereServlet
  </servlet-class>

  <async-supported>true</async-supported>
</servlet>

<servlet-mapping>
  <servlet-name>AtmosphereServlet</servlet-name>
  <url-pattern>/ws/*</url-pattern>
</servlet-mapping>
```

Atmosphere – Client (JavaScript)

Atmosphere – Client (JavaScript)

```
var subsc = $.atmosphere.subscribe(
```

```
);
```

Atmosphere – Client (JavaScript)

```
var subsc = $.atmosphere.subscribe({  
  url: 'http://localhost:8080/lt/ws/echo',  
  transport: 'websocket',  
  trackMessageLength: true,  
  logLevel: 'debug`  
  
});
```

Atmosphere – Client (JavaScript)

```
var subsc = $.atmosphere.subscribe({
  url: 'http://localhost:8080/lt/ws/echo',
  transport: 'websocket',
  trackMessageLength: true,
  logLevel: 'debug',

  onMessage: function (response) { //...
  },
  onOpen: function (response) { //...
  },
  onReopen: function (response) { //...
  },
  onClose: function (response) { //...
  },
  onError: function (response) { //...
  },
  onReconnect: function (response) { //...
  }
});
```

Atmosphere – Client (JavaScript)

```
var subsc = $.atmosphere.subscribe({
  url: 'http://localhost:8080/lt/ws/echo',
  transport: 'websocket',
  trackMessageLength: true,
  logLevel: 'debug',

  onMessage: function (response) { //...
  },
  onOpen: function (response) { //...
  },
  onReopen: function (response) { //...
  },
  onClose: function (response) { //...
  },
  onError: function (response) { //...
  },
  onReconnect: function (response) { //...
  }
});

subsc.push('send some message to server');
```

Atmosphäre



DAS WAR EINFACH!

DEMO

Donnerstag, 5. September

Zeit	.NET	HTML & Co.	Java	Qualität	Verschiedenes
08:30 – 09:20	K30 Dies & das Allerlei in aller Kürze Michael Wiedeking et al.				
Pause					
09:40 – 10:50	N31 From Zero to Azure Hero Professionelle Cloud Entwicklung mit Windows Azure und C# Robert Eichenseer	H31 Zur Hölle mit Responsive! Ein Streitgespräch Florian Hirsch	J31 Lambdas I Funktionale Programmierung in Java mit Lambdas Angelika Langer	Q31 Test-Driven Development - Möge der Test beginnen Entwickler vs. Architekt Gregor Biswanger et al.	V31 Identitätskrise Zeit, Zustand und das Reinheitsgebot Andreas Flierl
Pause					
11:20 – 12:30	N32 Das [C]Kreuz mit den Lambdas Eine erweiterbare generisch-parallele Suche mit C#-PLinq Dr. Ulrich Hilburger	H32 Script Factory Method Builder Fortgeschrittene Techniken mit JavaScript Frank Goras	J32 Lambdas II Funktionale Erweiterungen der Java-Collections Klaus Kreft	Q32 Akzeptanztests und Regression Komplexe Schnittstellen und Prozesse mit FitNesse testen Frank Ganske	V32 Maßnahmen gegen das Code-Chaos in Standard-Software Erweiterung von Standard-Software um kundenindividuelle Funktionalitäten Martin Aschoff
Mittagessen					
14:00 – 15:10	N33 Flaschenhalse eliminieren Performance Optimierung von .NET-Applikationen Markus Müller	H33 <Superheld/> Web-Applikationen mit AngularJS Joachim Weinbrenner	J33 Schwarze 8 Änderungen in Java 8 Michael Wiedeking	Q33 Sichere Software Vermeidung von Angriffspunkten bei der Software-Entwicklung Andreas Vombach	V33 NoSQL hoch drei 3 NoSQL Datenbanken in 70 Minuten Eberhard Wolff
Pause					
15:40 – 16:50	N34 Dynamisch und gefährlich? C# Dynamics in freier Wildbahn Timothée Bourguignon	H34 Die ultimative Herausforderung Responsive Webdesign und Parallax Scrolling Uwe Frieser	J34 Die Ecke Das vert.x Framework – modular, polyglott, asynchron Eberhard Wolff	Q34 Selenium WebDriver Automatisches Testen von Webapps auf gängigen Browsern Walery Strauch	V34 ALM konkret Die Aspekte des Application Lifecycle Management entdecken, verstehen und nutzen Sebastian Rölz
Kleine Pause					
17:00 – 17:15	Das Beste kommt zum Schluss Verabschiedung Das Herbstcampus-Team				

Donnerstag, 5. September

Zeit	.NET	HTML & Co.	Java	Qualität	Verschiedenes
08:30 – 09:20			K30 Dies & das Allerlei in aller Kürze Michael Wiedeking et al.		
	Pause				
09:40 – 10:50	N31 From Zero to Azure Hero Professionelle Cloud Entwicklung mit Windows Azure und C# Robert Eichenseer	H31 Zur Hölle mit Responsive! Ein Streitgespräch Florian Hirsch	J31 Lambdas I Funktionale Programmierung in Java mit Lambdas Angelika Langer	Q31 Test-Driven Development - Möge der Test beginnen Entwickler vs. Architekt Gregor Biswanger et al.	V31 Identitätskrise Zeit, Zustand und das Reinheitsgebot Andreas Flierl
	Pause				
11:20 – 12:30	N32 Das [C]Kreuz mit den Lambdas Eine erweiterbare generisch-parallele Suche mit C#-PLinq Dr. Ulrich Hilburger	H32 Script Factory Method Builder Fortgeschrittene Techniken mit JavaScript Frank Goras	J32 Lambdas II Funktionale Erweiterungen der Java-Collections Klaus Kreft	Q32 Akzeptanztests und Regression Komplexe Schnittstellen und Prozesse mit FitNesse testen Frank Ganske	V32 Maßnahmen gegen das Code-Chaos in Standard-Software Erweiterung von Standard-Software um kundenindividuelle Funktionalitäten Martin Aschoff
	Mittagessen				
14:00 – 15:10	N33 Flaschenhalse eliminieren Performance Optimierung von .NET-Applikationen Markus Müller	H33 <Superheld/> Web-Applikationen mit AngularJS Joachim Weinbrenner	J33 Schwarze 8 Änderungen in Java 8 Michael Wiedeking	Q33 Sichere Software Vermeidung von Angriffspunkten bei der Software-Entwicklung Andreas Vombach	V33 NoSQL hoch drei 3 NoSQL Datenbanken in 70 Minuten Eberhard Wolff
	Pause				
15:40 – 16:50	N34 Dynamisch und gefährlich? C# Dynamics in freier Wildbahn Timothée Bourguignon	H34 Die ultimative Herausforderung Responsive Webdesign und Parallax Scrolling Uwe Frieser	J34 Die Ecke Das vert.x Framework modular, polyglott Eberhard Wolff	Q34 Selenium WebDriver Automatisches Testen von Webapps auf gängigen Browsern Walery Strauch	V34 ALM konkret Die Anwendung des Application Lifecycle Management entdecken, verstehen und nutzen Sebastian Rölz
	Kleine Pause				
17:00 – 17:15	Das Beste kommt zum Schluss Verabschiedung Das Herbstcampus-Team				

Vielen dank!



2.– 5. September 2013
in Nürnberg



Herbstcampus

Wissenstransfer
par excellence

Vielen Dank!

Walery Strauch

@walery

walery@walery.org

Firma

- Hier können Sie, wenn Sie möchten, Informationen über Ihre Firma einfügen