

2.– 5. September 2013
in Nürnberg



Herbstcampus

Wissenstransfer
par excellence

Apache TomEE:

Was braucht man mehr für JavaEE?

Frank Pientka

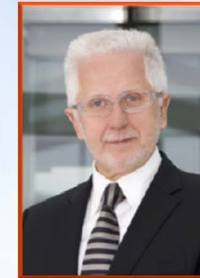
MATERNA GmbH

MATERNA-Unternehmensgruppe



Dr. Winfried Materna

Helmut an de Meulen



Gesellschafter

155 Mio. € Umsatz 2012

1.300 Mitarbeiter

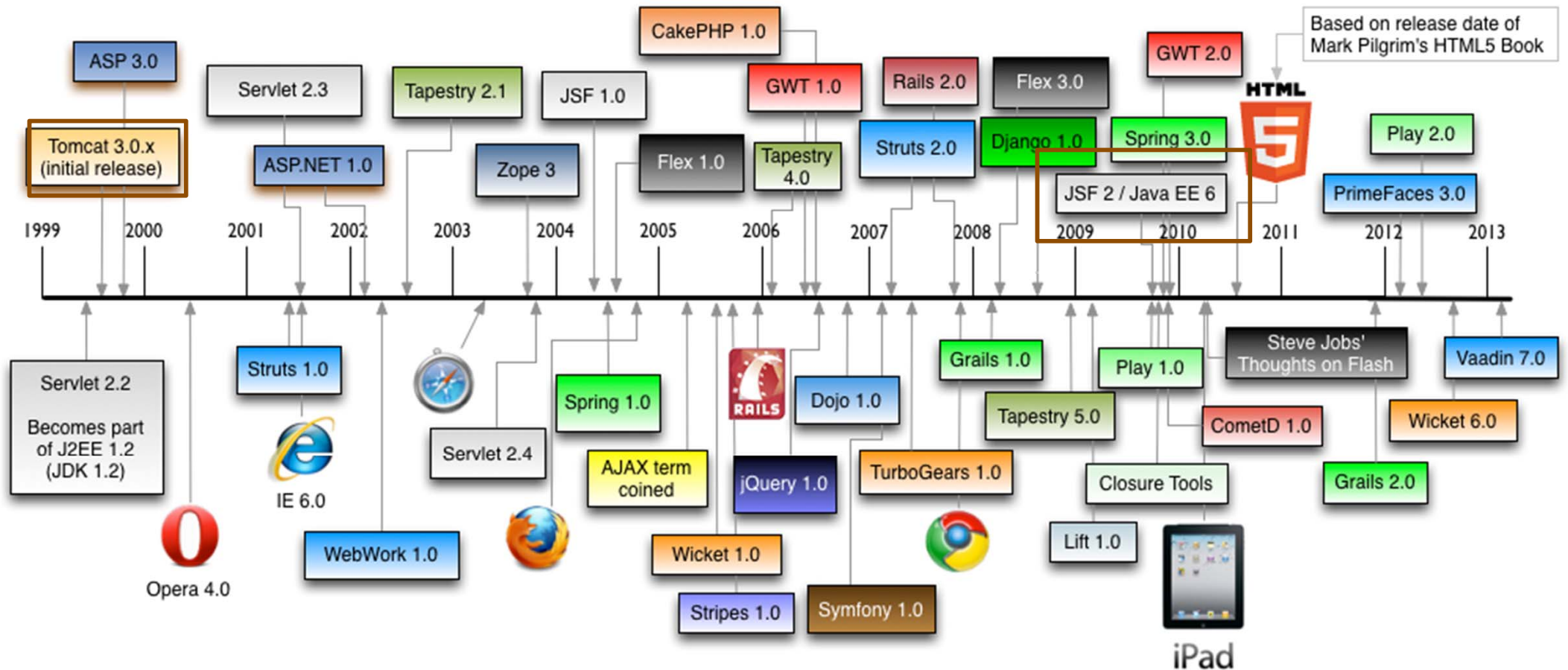
Lünendonk 2012 TOP 19

Gegründet 1980

Inhalt

- Java-Webentwicklung heute
- Darf ich vorstellen: Tomcat, TomEE
- Beispiel
- Warum wechseln?
- Ausblick

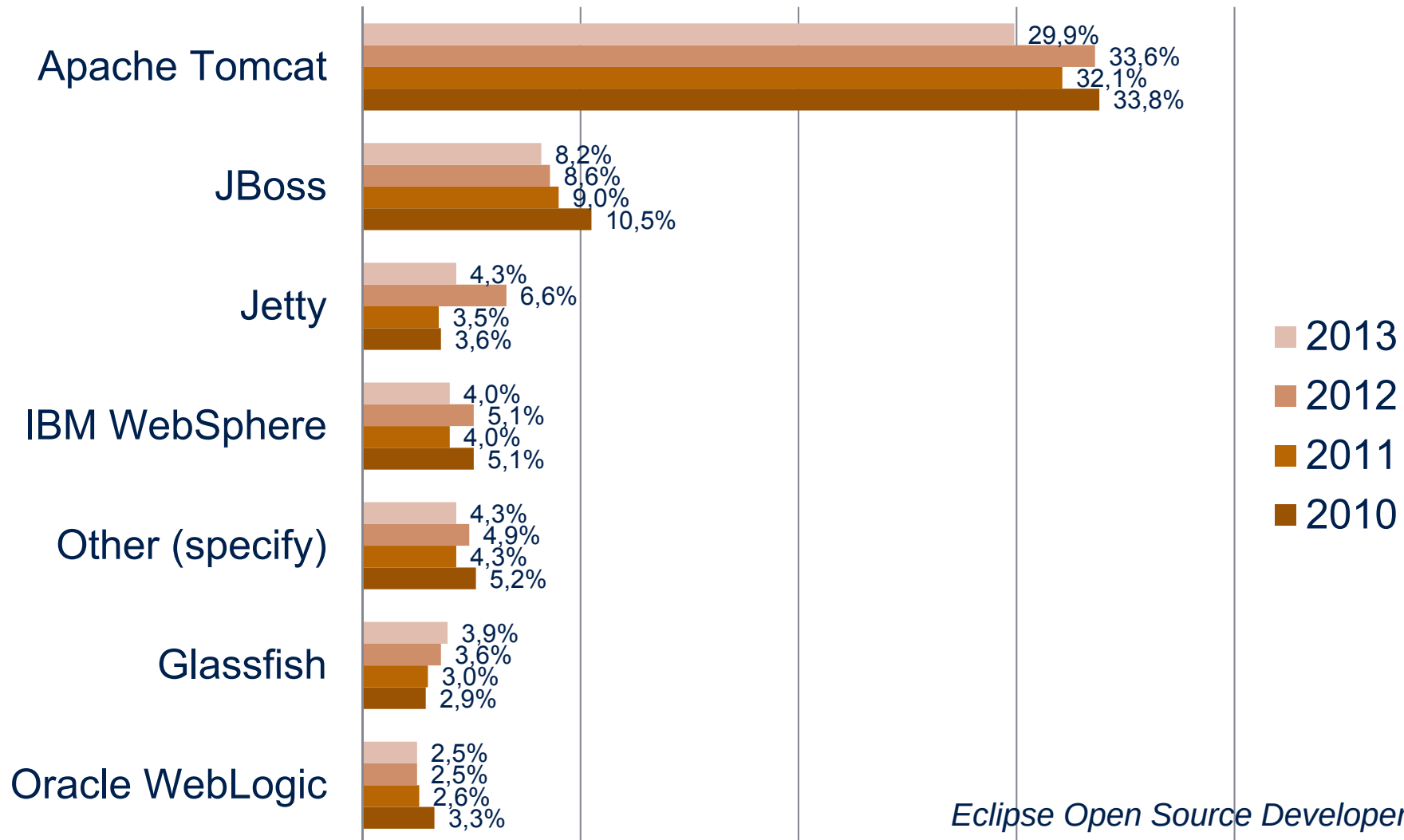
Historie von Webframeworks



<https://github.com/mraible/history-of-web-frameworks-timeline>

Primary Application Server

What is the primary application server you typically use for deployed applications? (Choose one.)



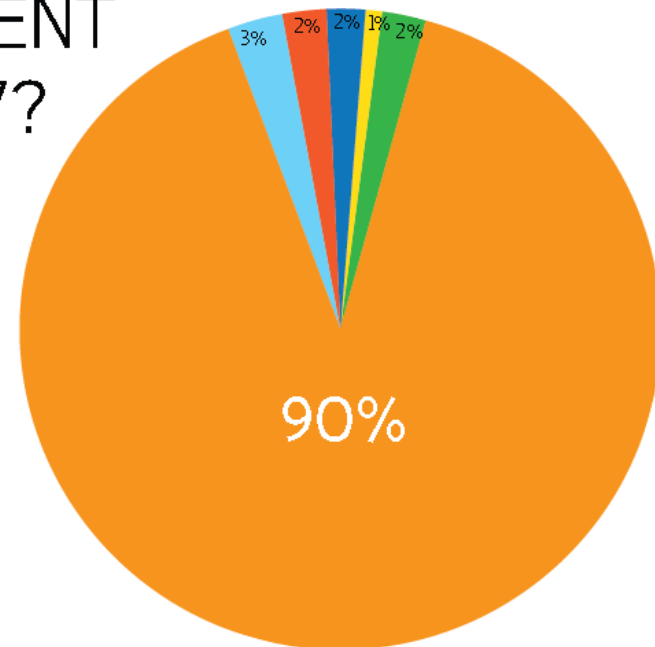
Eclipse Open Source Developer Report 2013

Was ist das wichtigste an JavaEE7?

WHAT'S THE MOST IMPORTANT ENHANCEMENT IN JAVA EE 7?

JavaMagazin MAY/JUNE 2013

In a recent [Java.net](#) poll, the Java developer community overwhelmingly selected HTML5 support as being the most important enhancement in Java EE 7. A total of 1,631 votes were cast during the two-week survey. The final voting shares were



90%

HTML5 support, including WebSocket and JSON Processing ([JSR 356](#)) and [JSR 353](#))

3%

JavaServer Faces 2.2 ([JSR 344](#))

2%

JAX-RS 2.0 (the Java API for RESTful Web Services, [JSR 339](#))

2%

Expression Language 3.0 ([JSR 341](#))

1%

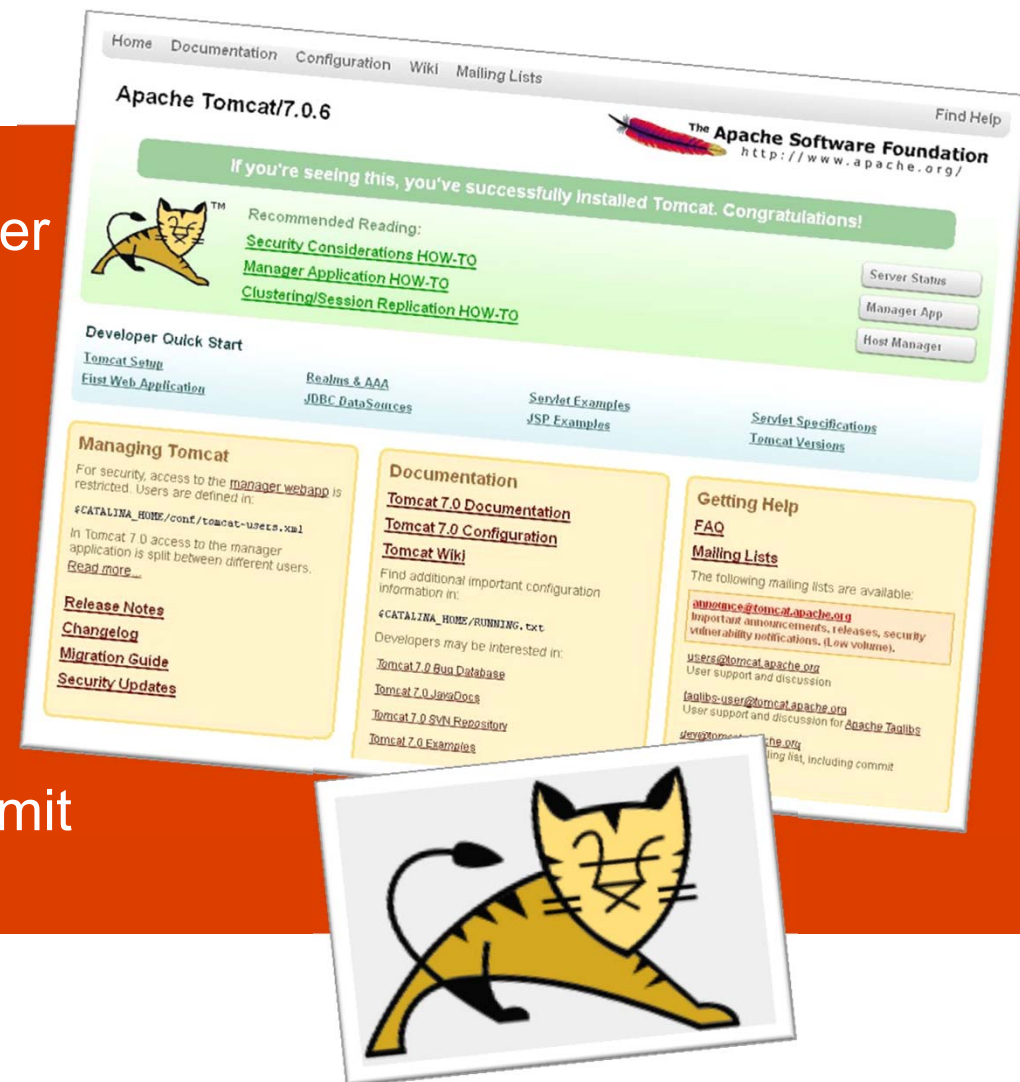
Java Message Service 2.0 ([JSR 343](#))

2%

Something else

Apache Tomcat

- Am weitesten verbreiteter Servlet-Container
- Niemals JavaEE-zertifiziert
- Spartanische GUI, Doku
- Lange Release-Zyklen, kein Support
- Manuelle Erweiterungen nötig TomEE
- Mit Heraku, Jelastic, Elastic Beanstalk in die Cloud
- TomEE für JavaEE6-Profil zertifiziert, 1.5 mit JAX-RS, Plus



<http://tomee.apache.org/comparison.html>

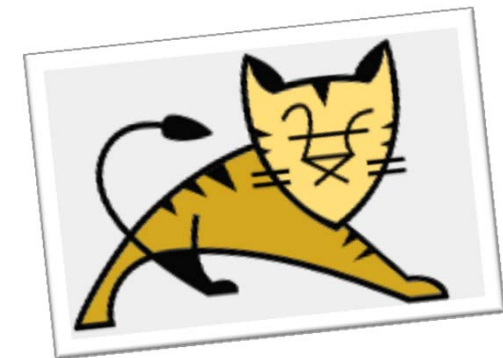
Tomcat Historie: Meilensteine

- 2002 Tomcat 4 wird zu beliebtesten Web Container
- 2004 Tomcat 5 mit Unterstützung von Servlet API 2.4 und JSP 2.0
- 2005 Tomcat wird Top Level Projekt der Apache Foundation
- 2006 Tomcat 5.5 EOL Oktober 2012
- 2007 Tomcat 6
- 2010 Tomcat 7
- 2013 Tomcat 8

<http://tomcat.apache.org/whichversion.html>

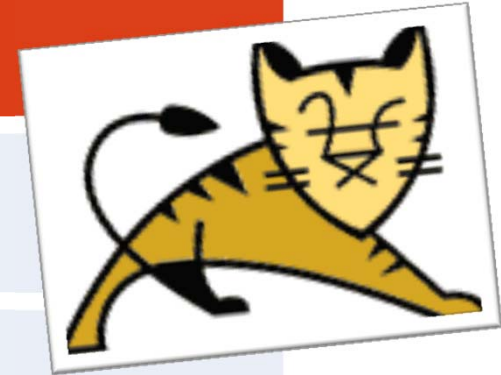
<http://wiki.apache.org/tomcat/TomcatVersions>

<http://tomcat.apache.org/migration-8.html>



Apache Tomcat Versionen

Servlet/ JSP Spec		Tomcat Version	JDK, EL Version
3.1	2.3	8.0.x	JDK 1.7+, EL 3.0
3.0	2.2	7.0.x	JDK 1.6+, EL 2.2
2.5	2.1	6.0.x	JDK 1.5+, EL 2.1
2.4	2.0	5.0.x, 5.5.x	JDK 1.4+, EL 1.0
2.3	1.2	4.1.x	JDK 1.3+, EL 1.0
2.2	1.1	3.3.x	JDK 1.2+



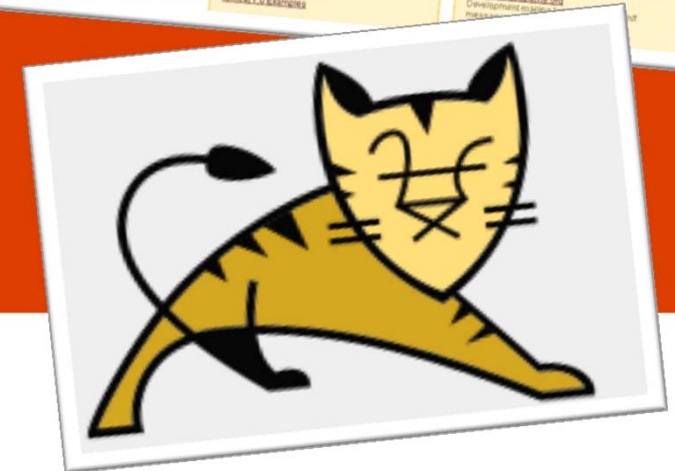
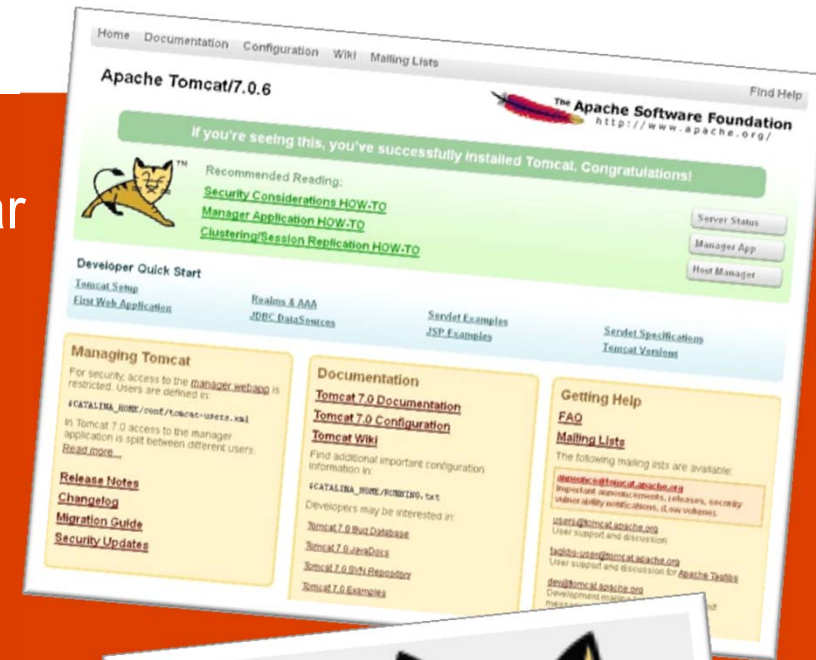
Tomcat 8 RC1

- catalina-storeconfig.jar (Generation of XML configuration from current state)
- el-api.jar (EL 3.0 API)
- jasper-el.jar (Jasper 2 EL implementation)
- jsp-api.jar (JSP 2.3 API)
- servlet-api.jar (Servlet 3.1 API)
- tomcat-jdbc.jar (Tomcat's database connection pooling solution)
- tomcat-jni.jar (Interface to the native component of the APR/native connector)
- tomcat-spdy.jar (SPDY implementation)
- tomcat-util.jar (Various utilities)
- tomcat-websocket.jar (WebSocket 1.0 implementation)
- websocket-api.jar (WebSocket 1.0 API)
- tomcat-spdy.jar (SPDY implementation)
- tomcat-util.jar (Various utilities)
- tomcat-websocket.jar (WebSocket 1.0 implementation)
- websocket-api.jar (WebSocket 1.0 API)



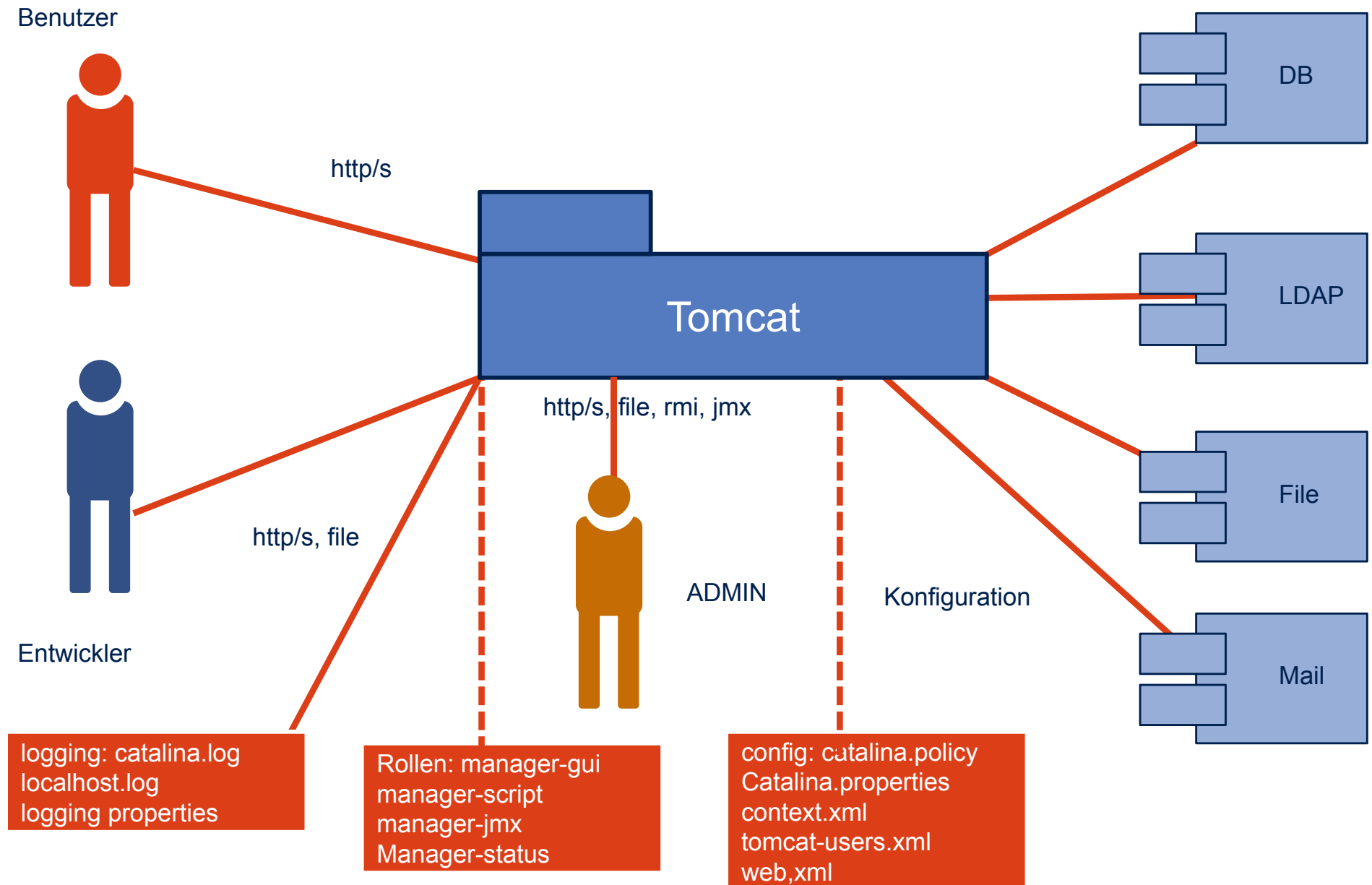
Was kostet Tomcat ?

Kostenlose Online Manuals und Tutorials verfügbar
Mailinglisten und Foren für alle Niveaus
Wenig Einarbeitungsaufwand
Angebote für Coaching und Consulting / Support
Der “Rest“ kostet NICHTS !!!
Apache Software Licence Version 2.0



<http://www.apache.org/LICENSE>

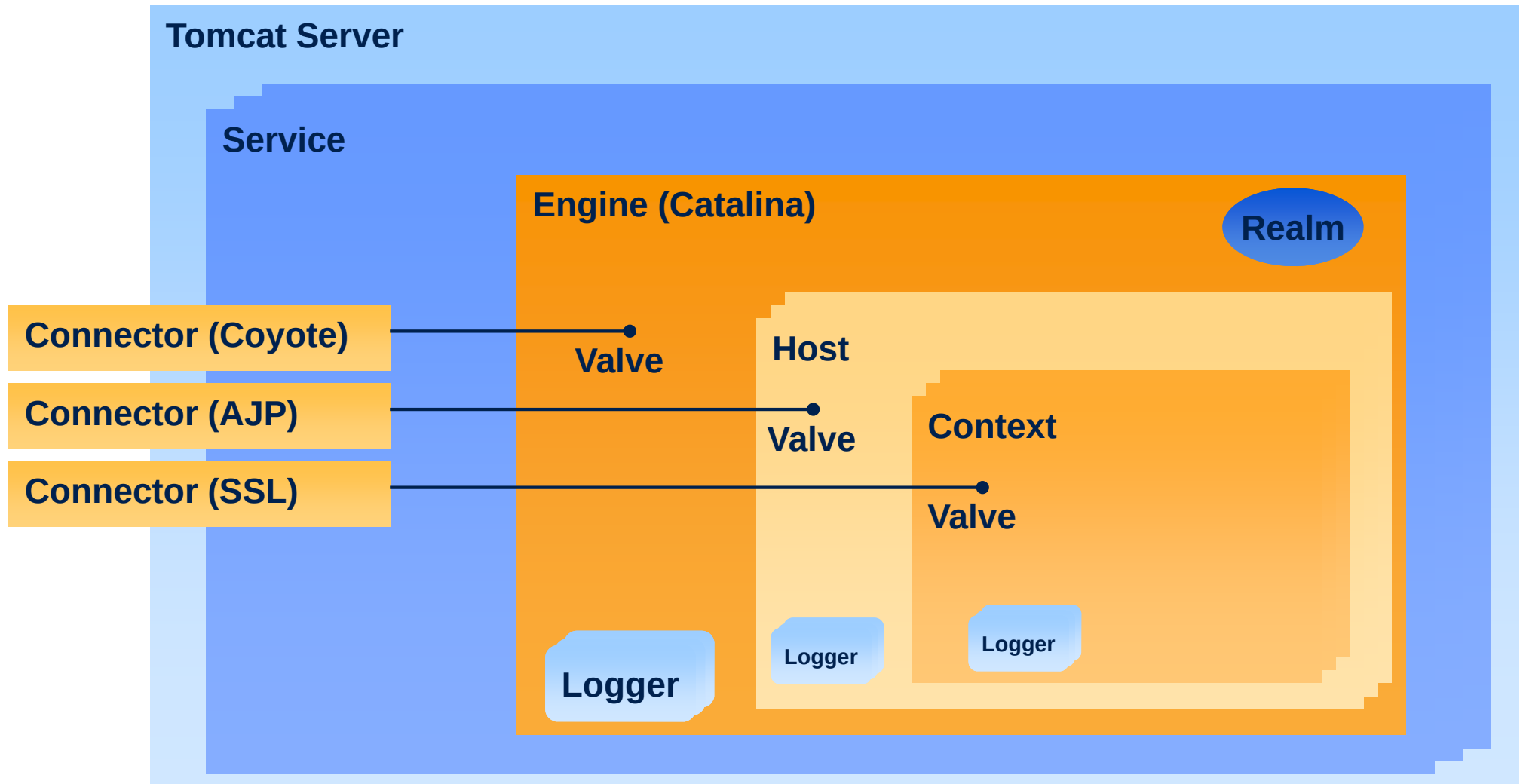
Tomcat-Überblick



Tomcat Deploymentarten

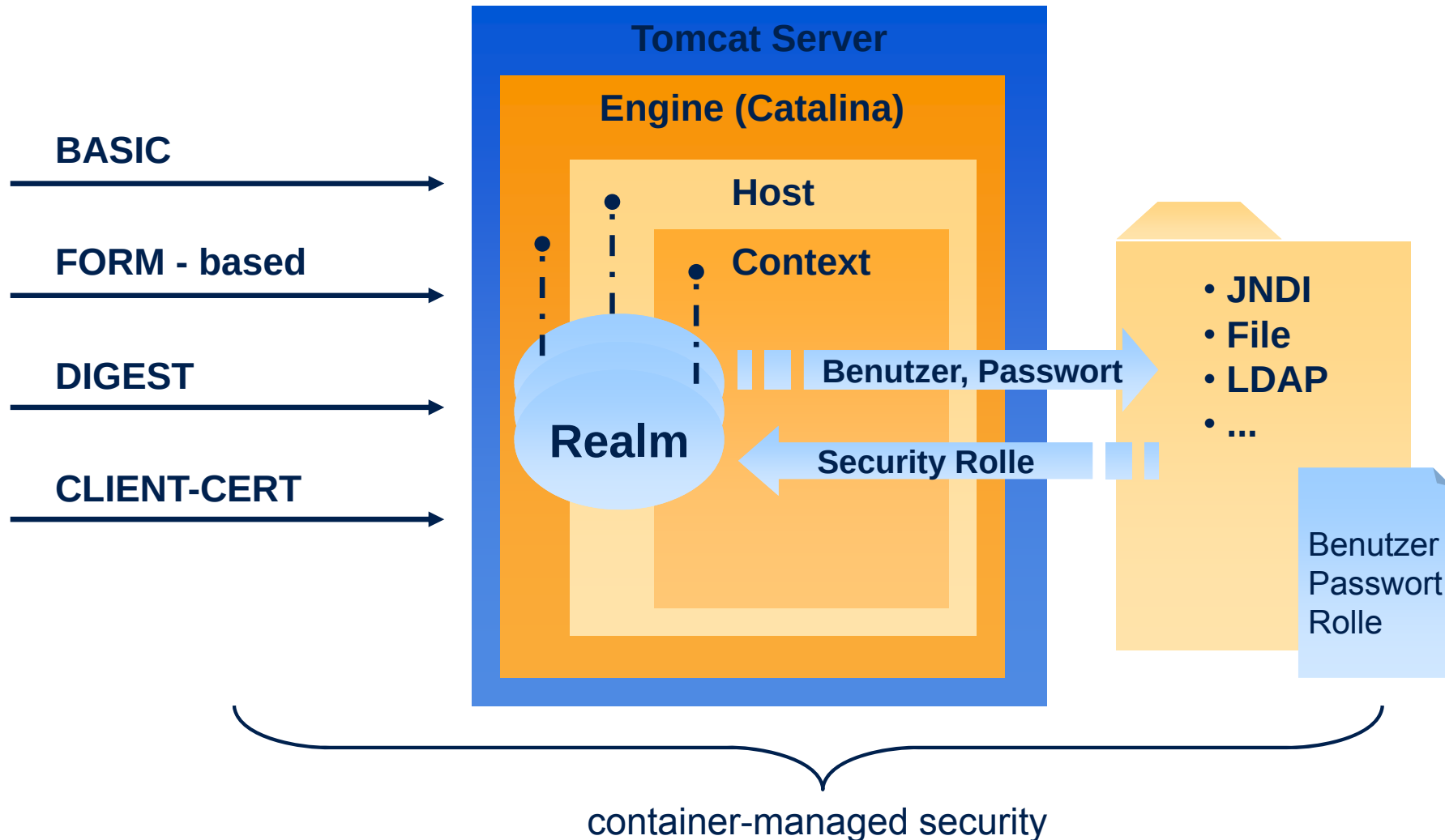


Architektur - Tomcat Komponenten



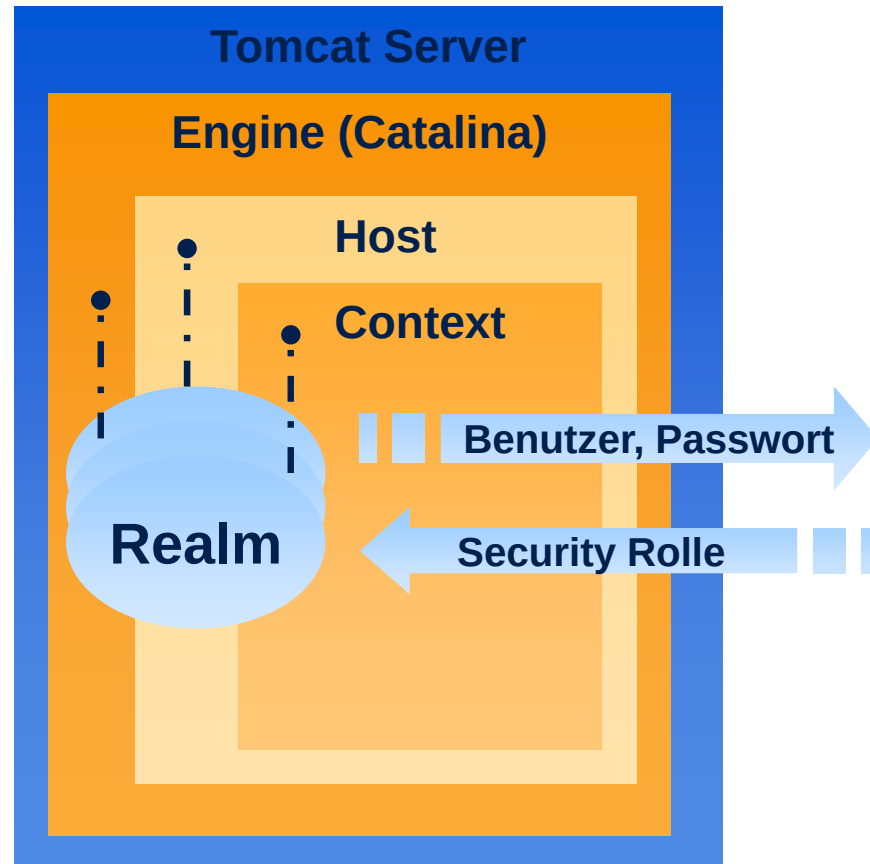
Authentifizierungsverfahren bei Webanwendungen

Zugriffsmethoden:

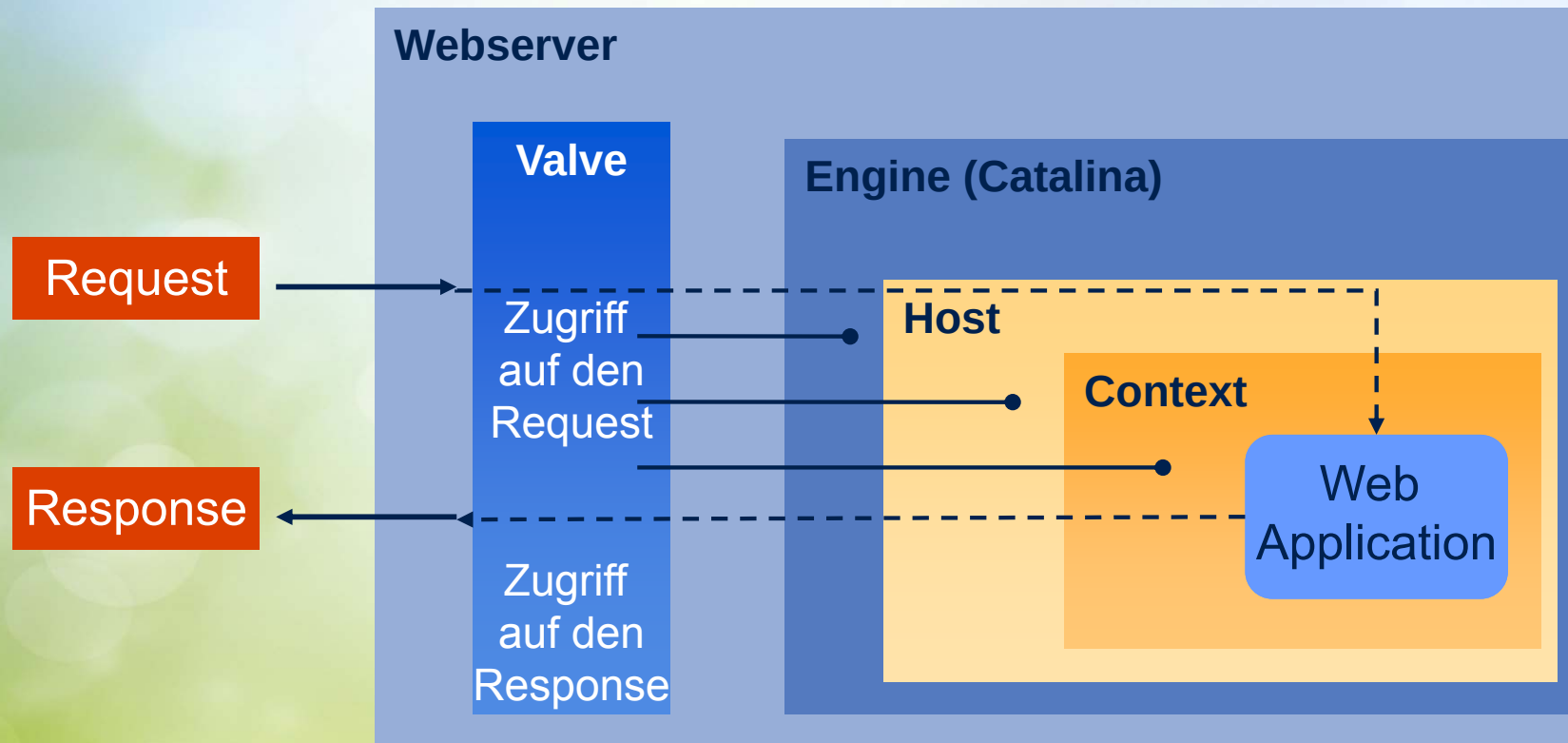


Definitionsebenen von Realms

- Engine (**<Engine>**)
 - für alle Web Applikationen auf allen Hosts
 - überschreibbar durch Realms innerhalb von **<Host>** und **<Context>**
- Host (**<Host>**)
 - für alle Web Applikationen auf diesem Host
 - überschreibbar durch Realms innerhalb von **<Context>**
- Context (**<Context>**)
 - für diese Web Applikation



Filter und Ventile



Arten von Valve Komponenten

Access Logging

Ermöglicht
erweitertes Logging
(vgl. *Logger*)

Single Sign-On

Authentifizierung
erfolgt nur einmal im
Host

Request Filtering

Zugriffssteuerung
auf Basis von IP /
Hostnamen

Request Dumping

Schreibt sämtliche
Daten aus dem
Request



weitere Valves:
z.B. FormAuthenticatorValve,
JDBCRemoteAccessValve, ...

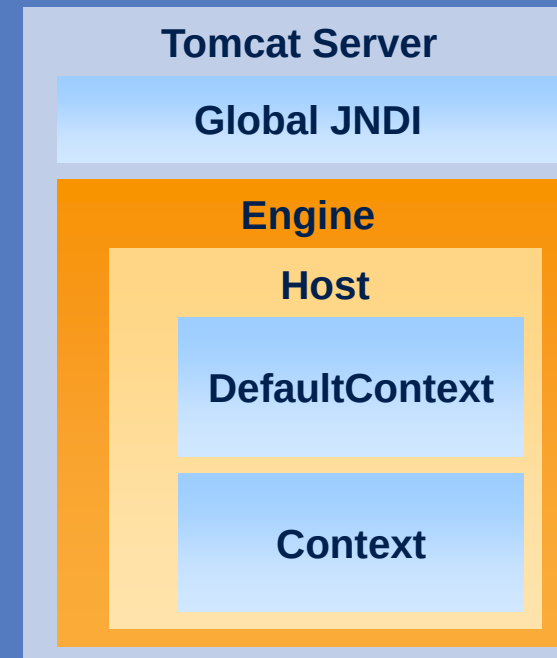
Tomcat JNDI Konfiguration

Innerhalb folgender Elemente:

- **<GlobalNamingResources>**
 - global innerhalb von Tomcat
- **<DefaultContext>**
 - global innerhalb des Hosts
- **<Context>**
 - nur für Application

Für Globale Ressourcen

- **<ResourceLink>**
 - jeder Context kann durch Link auf globale Elemente zugreifen



Monitoring via JMX

- Erweitern der JAVA_OPTS für die Verfügbarkeit von Remote JMX
 - -Dcom.sun.management.jmxremote
 - Dcom.sun.management.jmxremote.port=9004
 - Dcom.sun.management.jmxremote.authenticate=false
 - Dcom.sun.management.jmxremote.ssl=false
 - optional: Verwendung von Security, Definition von Rollen
- JMX Client ermöglicht Monitoring und Administration der MBeans im JDK in JAVA_HOME\bin - JConsole.exe
 - Open Source JMX Clients, z.B. MC4J (<http://www.mc4j.org>)
 - via HTTP Commands

Höhere Sicherheit mit SSL-Verschlüsselung:

- -Dcom.sun.management.jmxremote.ssl=true
- -Djavax.net.ssl.keyStore=keystore
- -Djavax.net.ssl.keyStorePassword=password

Farbige Logdateien

<http://code.google.com/p/jlibs>

```
import java.util.logging.ConsoleHandler;
import jlibs.core.util.logging.AnsiFormatter;
public class ColorConsoleHandler extends ConsoleHandler {
    public ColorConsoleHandler() {
        setFormatter(new AnsiFormatter());
    }
}
```

```
INFO      :OpenWebBeans Container is starting...
INFO      :Adding OpenWebBeansPlugin : [CdiPlugin]
INFO      :Adding OpenWebBeansPlugin : [OpenWebBeansJsrfPlugin]
INFO      :All injection points were validated successfully.
INFO      :OpenWebBeans Container has started, it took [20] ms.
using context file D:\apache-tomee-plus-1.0.0\webapps\host-manager\META-INF\context.xml
INFO      :Deployed Application(path=D:\apache-tomee-plus-1.0.0\webapps\host-manager)
WARNING   :No mappings of FacesServlet found. Abort initializing MyFaces.
INFO      :Deploying web application directory D:\apache-tomee-plus-1.0.0\webapps\manager
TomcatWebApplicationBuilder.start /manager
INFO      :faces config file is null
```

Farbige Logdateien

WINDOWS **ansicon** –i <http://adoxa.110mb.com/ansicon/index.html>

%CATALINA_BASE%/

conf/logging.properties

- handlers = 1catalina.org.apache.juli.FileHandler, ColorConsoleHandler
 - .handlers = 1catalina.org.apache.juli.FileHandler, ColorConsoleHandler
- bin/catalina.bat
- set CATALINA_OPTS=-DAnsi=true
 - set "CLASSPATH=%CLASSPATH%;%CATALINA_HOME%\bin\tomcat-juli.jar;%CATALINA_BASE%\bin\jlibs-core.jar;%CATALINA_BASE%\bin\colorloghandler.jar,,
 - catalina.bat run

<http://www.jcabi.com/jcabi-log/multicolor.html>

log4j.properties:

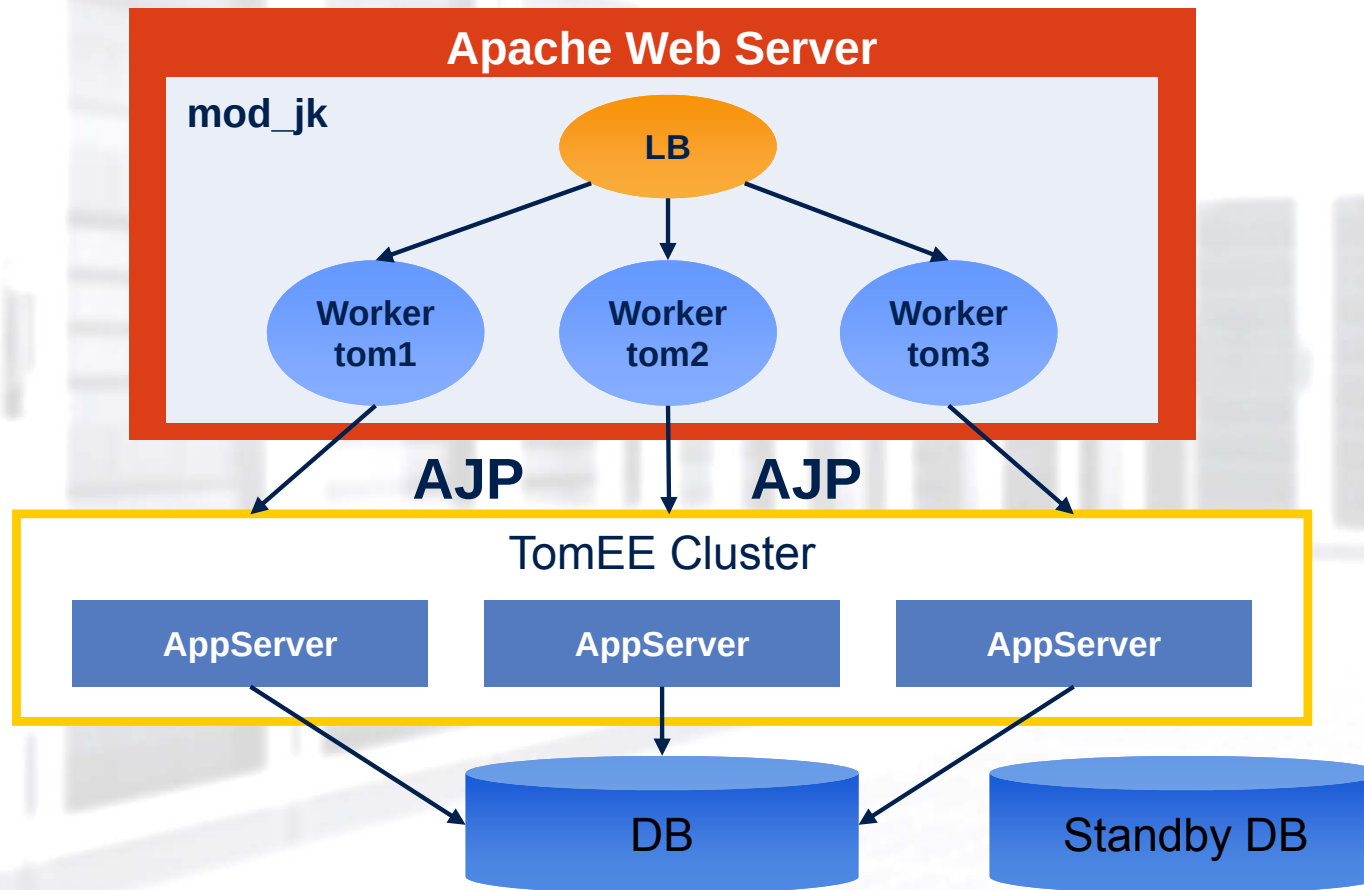
log4j.rootLogger=INFO, CONSOLE

log4j.appender.CONSOLE=org.apache.log4j.ConsoleAppender

log4j.appender.CONSOLE.layout=com.jcabi.log.MulticolorLayout

log4j.appender.CONSOLE.layout.ConversionPattern=[%color{%p}] %c: %m%n

Apache Loadbalancer



TomEE/OpenEJB

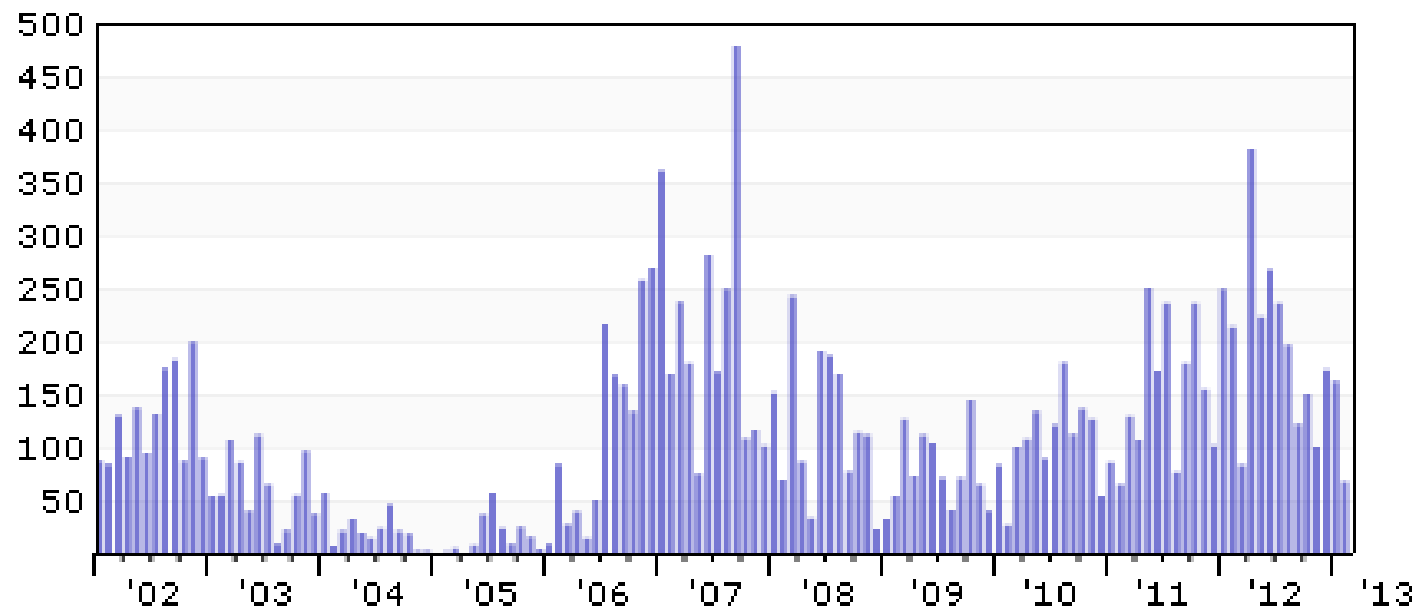
- Dezember 1999: OpenEJB 0.01 Richard Monson-Haefel, David Blevins
- Januar 2002 OpenEJB 0.8 SourceForge.net
- March 2004 OpenEJB 0.9 Codehaus
- September 2006 OpenEJB 2.0 Apache Incubator, Geronimo 1.x
- Juni 2007 Apache OpenEJB 3.0 TLP, Geronimo 2.x
- **April 2012 TomEE 1.0 TLP, OpenEJB 3.1.3, Java EE 6**
- Oktober 2012 TomEE 1.5.0 , OpenEJB 4.5.1
- **April 2013 TomEE 1.5.2, OpenEJB 4.5.2**
- Q3/2013 TomEE 1.5.3/1.6.0, OpenEJB 4.5.3/4.6.0 (JavaEE 6)
- <http://tomee.apache.org/download/index.html>
- <http://ci.apache.org/projects/tomcat/tomcat8/docs/RELEASE-NOTES.txt>

TomEE Varianten

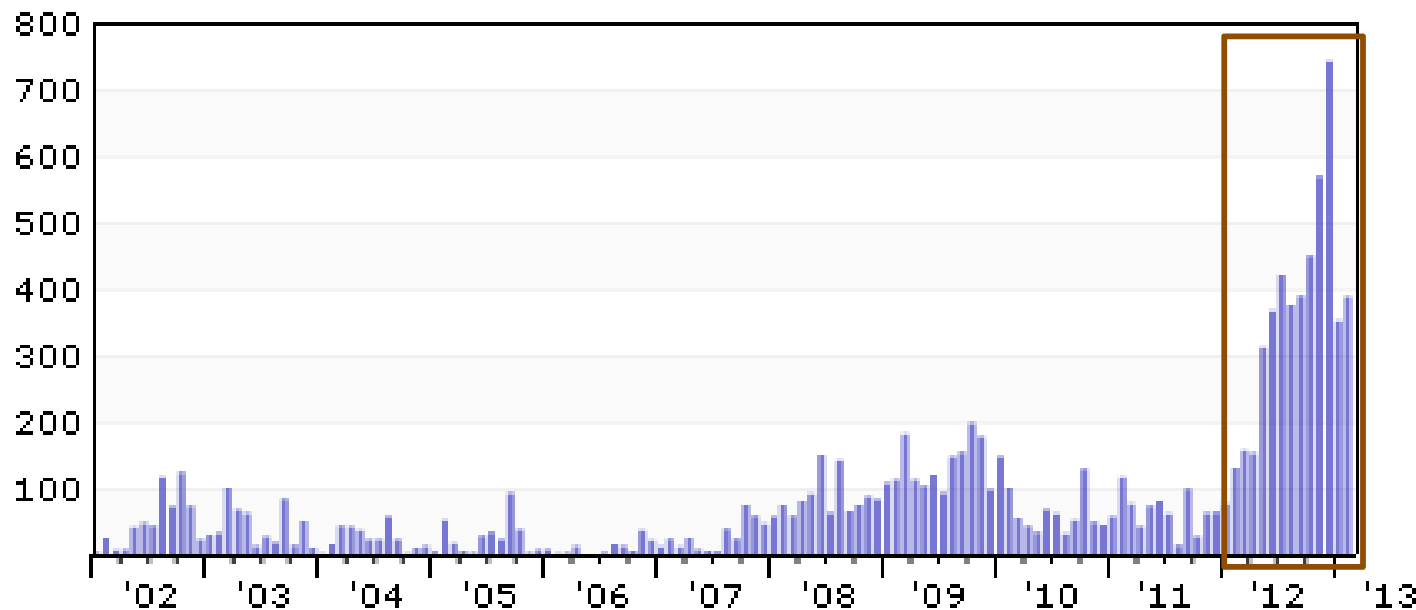
- TomEE Web Profile certification
 - CDI - Apache OpenWebBeans
 - EJB - Apache OpenEJB
 - JPA - Apache OpenJPA
 - JSF - Apache MyFaces
 - JSP - Apache Tomcat
 - JSTL - Apache Tomcat
 - JTA - Apache Geronimo Transaction
 - Servlet - Apache Tomcat
 - Javamail - Apache Geronimo JavaMail
 - Bean Validation - Apache BVal
- TomEE JAXRS
 - JAX-RS - Apache CXF
- TomEE+
 - JAX-WS - Apache CXF
 - JMS - Apache ActiveMQ
 - Connector - Apache Geronimo Connector
- KarafEE Multicontainer OSGI runtime



TomEE ist im Kommen: InfoWorld 2013-03-21 Can TomEE save Java EE?



dev@tomee.apache.org



users@tomee.apache.org

Tomcat, TomEE, OpenEJB Vergleich

	Tomcat	TomEE	TomEE JAXRS	TomEE+	OpenEJB
Servlets	X	X	X	X	
JSP	X	X	X	X	
JSF		X	X	X	
JTA		X	X	X	X
JPA		X	X	X	X
CDI		X	X	X	X
JAAS		X	X	X	X
JACC		X	X	X	X
JavaMail		X	X	X	X
Bean Validation		X	X	X	X
Enterprise JavaBeans		X	X	X	X
JAX-RS			X	X	X
JAX-WS				X	X
Connector Architecture				X	X
JMS				X	X

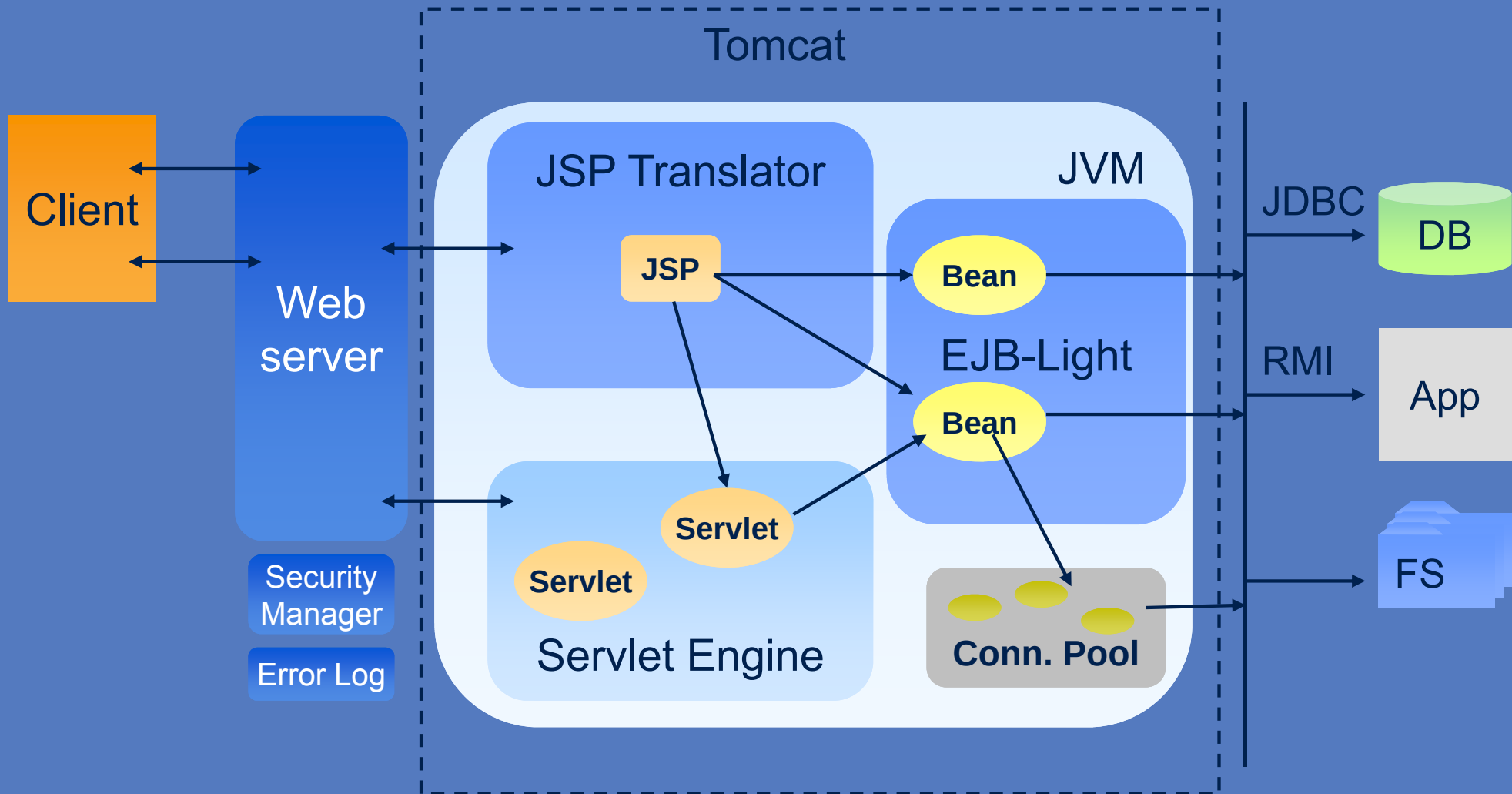
Meet TomEE in the Cloud! Jelastick

The screenshot displays the Jelastic console interface for configuring a new environment. The top navigation bar includes tabs for JAVA and PHP. The main configuration area is divided into several sections:

- Stack Selection:** A vertical flowchart shows the selection of components: NGINX (green box), TomEE+ 1.5 (blue box with cat icon), MySQL 5.5 (orange box with leaf icon), VDS (grey box), SSL (grey box), and a gear icon for additional settings.
- Virtual Dedicated Server Settings:**
 - A toggle switch for "Virtual Dedicated Se.." is set to OFF.
 - "Vertikale Skalierung pro Knoten" (Vertical scaling per node) is configured with:
 - Reserviert: 1 Cloudlet(s)
 - Skalierungsgrenze: bis zu 8 Cloudlet(s)
 - "Horizontale Skalierung" (Horizontal scaling) is shown with a slider from 0 to 64 nodes.
 - The operating system is set to Centos 6.3.
 - "Öffentliche IPv4" (Public IPv4) is toggled ON.
- Ressourcen (cloudlets) Summary:**
 - Reservierte Cloudlets: Jetzt: 9, Nächste: 16
 - rabatt: 25%, rabatt: 30%
 - Insgesamt reservierte Cloudlets: €47 pro Monat
 - Sie sparen €16/Monat durch die reservierten Cloudlets
 - monatliche Kosten: €47* → €255*
 - *Gebühren entfallen für die Test-Phase.
 - Buttons: Details anzeigen, Quote & Preis
 - Umgebung Name: env-3279591.jelastic.dogado.eu

An inset window at the bottom left provides a detailed view of the TomEE+ 1.5 component, showing its monthly cost as €25* and providing links for details and pricing.

TomEE-Aufbau



OpenEJB



Application Server

OpenEJB CPI

OpenEJB

Stateful Bean
Containers

Stateless Bean
Containers

Entity Bean
Containers

OpenEJB SPI

JTA

Transaction Manager

Security Manager

Resource Managers

Datenquellen-Fallback: TOMEE-912 HA datasources + jta handling TomEE 1.5.3

<http://rmannibucau.wordpress.com/2013/04/26/tomeeopenejb-and-failover-of-datasources>

<http://tomee.apache.org/dynamic-datasource.html>

<http://tomee.apache.org/examples-trunk/dynamic-datasource-routing/>

```
<Resource id="jdbc/main" type="DataSource">
```

```
    JdbcUrl = jdbc:mysql://main:3306/ds
```

```
    ...
```

```
</Resource>
```

```
<Resource id="jdbc/fallback" type="DataSource">
```

```
    JdbcUrl = jdbc:mysql://fallback:3306/ds
```

```
    <Resource id="failover-router" class-name="org.apache.openejb.resource.jdbc.router.FailOverRouter">
```

```
        datasourceNames = jdbc/main,jdbc/fallback
```

```
</Resource>
```

```
    ...
```

```
</Resource>
```

```
<Resource id="jdbc/ds" type="DataSource" provider="RoutedDataSource">
```

```
    router = failover-router
```

```
</Resource>
```

TomEE-Werkzeuge

- configtest.bat
- shutdown.bat
- startup.bat
- tomee.bat
- mvn tomee:help
- IDE Eclipse, NetBeans, IDEA, ANT ...
- Webconsole



Apache TomEE wichtigste Verzeichnisse

- **bin** – unveränderte Skripte
- **conf** – server.xml mit einem extra Listener (ports, hosts, engines, threadpools Konfiguration etc). Extra file **openejb.xml** (global resources). Tomcat-users.xml für Rollen, Rechteverwaltung
- **logs** – Extra-Datei openejb.log
- **webapps** – wie bei Tomcat, Hot-Deploymnet WAR, EAR-Dateien oder Verzeichnisse hineinkopieren

Files Added

conf/

system.properties tomee.xml

bin/

TomEE.*.exe service*.bat tomee.sh

endorsed/

annotation-api.jar jaxb-impl.jar
jaxb-api.jar

lib/

bval-*.jar	myfaces-*.jar
commons-*.jar	openejb-*.jar
geronimo-javamail.jar	openjpa-*.jar
geronimo-transaction.jar	openwebbeans-*.jar
gson.jar	quartz.jar
howl.jar	serp.jar
hsqldb.jar	slf4j-*.jar
jansi.jar	swizzle-stream.jar
javaee-api.jar	tomee-*.jar
javassist.jar	wsdl4j.jar
mbean-annotation-api.jar	xbean-*.jar

Geänderte Dateien

- `conf/server.xml`
`<!-- TomEE plugin for Tomcat -->`
`<Listener className="org.apache.tomee.catalina.ServerListener" />`
 - `bin/catalina.sh`
 - Optional, statt build-time enhancement, runtime-time enhancement von OpenJPA oder EclipseLink verwenden
- ```
Add OpenEJB javaagent
if [-r "$CATALINA_HOME"/lib/openejb-javaagent.jar]; then
 JAVA_OPTS=""-javaagent:$CATALINA_HOME/lib/openejb-javaagent.jar$JAVA_OPTS"
fi
```



## ApacheTomEE: Security

- Tomcat's Security Realm
- TomEE Erweiterungen für
  - JAAS
  - JACC
  - WS Security

## Apache TomEE: Resources Configuration

- In \$CATALINA\_BASE/conf/openejb.xml  
«Resource id="moviesDatabase" type="DataSource"»  
@Resource DataSource moviesDatabase
- JDBC-Treiber in tomcat.home/lib kopieren

```
<Resource id="moviesDatabase" type="DataSource">
 JdbcDriver com.mysql.jdbc.Driver
 JdbcUrl jdbc:mysql:localhost:3306/moviedb
 UserName user
 Password secret
 JtaManaged true
</Resource>
```

## Apache TomEE: JMS-Resources

- In \$CATALINA\_BASE/conf/openejb.xml (Connection Factory, Topics, Queues)
- ActiveMQ default port 61616
- Resources injizieren mit
  - `@Resource ConnectionFactory connectionFactory;`
  - `@Resource Queue queue;`
  - `@Resource Topic topic;`

## Apache TomEE: Web Services

- @Webservice verwendbar in
  - POJO
  - Session Bean
- Web Service Security supported
- Web Service Clients also work
  - @WebServiceRef CalculatorWs service;

## Apache TomEE: EJB Client

- HTTP for EJB remote communication (<http://localhost:8080/openejb/ejb>)
- Great for firewalled systems
- User/Pass supplied with InitialContext params
- HTTP or HTTPS

```
Properties p = new Properties();
p.setProperty(Context.INITIAL_CONTEXT_FACTORY, "org.apache.openejb.client.RemoteInitialContextFactory");
p.setProperty(Context.PROVIDER_URL, "http://localhost:8080/openejb/ejb");

InitialContext ic = new InitialContext(p);
MoviesRemote movies = (MoviesRemote) ic.lookup("MoviesRemote");
```

# OpenEJB-JavaEE 80 Beispiele

## Session Beans

- simple-singleton \*
- simple-stateful \*
- simple-stateless \*
- simple-stateless-with-descriptor

## EntityManagers

- injection-of-entity-manager \*
- jpa-eclipselink
- jpa-hibernate
- jpa-enumerated \*

## CDI

- cdi-basic \*
- cdi-request-scope \*
- cdi-application-scope \*
- simple-cdi-interceptor \*
- cdi-produces-disposes
- decorators
- cdi-alternative-and-stereotypes

## EJB

- access-timeout \*
- async-methods \*
- schedule-expression \*
- schedule-methods
- interceptors

## REST

- simple-rest
- rest-example
- rest-example-with-application
- rest-on-ejb
- rest-xml-json

## Web Services

- simple-webservice \*
- webservice-handlerchain \*
- webservice-holder \*
- webservice-attachments
- webservice-inheritance
- webservice-security
- webservice-ws-security

## JMS and MDBs

- injection-of-connectionfactory
- simple-mdb-with-descriptor
- simple-mdb \*

## Transactions

- testing-transactions
- transaction-rollback
- applicationexception

## Security

- testing-security-3
- testing-security-2
- testing-security

examples-4.5.2-src.zip

## Data Sources

- injection-of-datasource
- datasource-ciphered-password \*
- dynamic-datasource-routing \*
- resources-declared-in-webapp

## Referencing EJBs

- injection-of-ejbs \*
- lookup-of-ejbs-with-descriptor
- lookup-of-ejbs

## Environment Entries

- injection-of-env-entry
- custom-injection

## Java EE Connectors

- quartz-app \*

## Testing Techniques

- alternate-descriptors \*
- application-composer \*
- testcase-injection
- ear-testing \*

## Frameworks

- spring-integration
- spring-data-proxy
- struts

## Meta-Annotations

- access-timeout-meta \*
- schedule-methods-meta
- testing-security-meta
- movies-complete-meta
- movies-complete

## Proxy Beans

- dynamic-dao-implementation \*
- dynamic-implementation
- dynamic-proxy-to-access-mbean
- spring-data-proxy

## EJB Legacy

- cmp2-entity-beans \*
- component-interfaces

## Hallo Servlet 3.0-Welt-Beispiel

```
WebServlet(value="/hello",
 initParams = {
 @WebInitParam(name="greeting", value="Hello "),
 @WebInitParam(name="to", value=" Servlet world 3.0!")
 })

public class HelloServlet extends GenericServlet {

 public void doGet (ServletRequest req, ServletResponse res)
 throws IOException, ServletException
 {
 PrintWriter out = res.getWriter();
 out.println(getInitParameter("greeting"));
 out.println(getInitParameter("to"));
 }
}
```

## Hello TomEE mit Maven

- HSQldb, EJB 3.1, JAAS Authentication, JSF 2.1
- <http://werpublogs.blogspot.co.at/2012/11/tomee-and-maven.html>
- Sourcen holen: <https://github.com/werpu/tomeemaven>
- mvn install
- User anlegen in tomcat-users.xml

```
<role name="authenticated-user" />
<user name="test" password="user" roles="authenticated-user" />
</tomcat-users>
```
- Deployen mvn tomee:help mvn tomee:run
- <http://localhost:8080/tomeemaven>
- Datenbankquelle anlegen in tomee.xml oder web.xml

## Hello TomEE: Web-Schicht

**@Named @RequestScoped**

class HelloView

{

**@EJB**

    HelloEJB helloEjb;

    //----- getter and setter -----

    public String getHelloWorld()

    {

        return helloEjb.getHello();

    }

}

## Hello TomEE: DAO-Schicht

**@Stateless**

class HelloEJB

{

    @PersistenceContext(name = "**Demo\_Unit**")

    EntityManager em;

    public String getHello()

    {

        Query query = em.createQuery("select hello from HelloEntity hello");

        if (query.getResultList().size() == 0)

        {

            HelloEntity entity = new HelloEntity();

            em.persist(entity);

            return entity.getHelloWorld();

        }

        HelloEntity entity = (HelloEntity) query.getResultList().get(0);

        return entity.getHelloWorld();

    }

}

## Hello TomEE Datenbankzugriff: tomee.xml oder web.xml

```
<tomee>
 <Resource id="MyDataSource"
 type="DataSource">
 JdbcDriver org.hsqldb.jdbcDriver
 JdbcUrl
 jdbc:hsqldb:hsql://localhost:9001/testdb2
 UserName sa
 Password
 JtaManaged true
 </Resource>
</tomee>
```

```
<data-source>
 <name>MyDataSource</name>
 <class-name>org.hsqldb.jdbcDriver</class-
name>
 <url>jdbc:hsqldb:hsql://localhost:9001/testdb2</url>
 <user>sa</user>
 <password></password>
</data-source>
</web-app>
```

# Hello TomEE Persistenz: persistence.xml

```
<persistence-unit name="Demo_Unit">
 <provider>org.apache.openjpa.persistence.PersistenceProviderImpl</provider>
 <jta-data-source>MyDataSource</jta-data-source>
 <class>com.github.werpu.tomeedemo.orm.HelloEntity</class>
 <properties>
 <property name="openjpa.jdbc.DBDictionary" value="hsql"/>

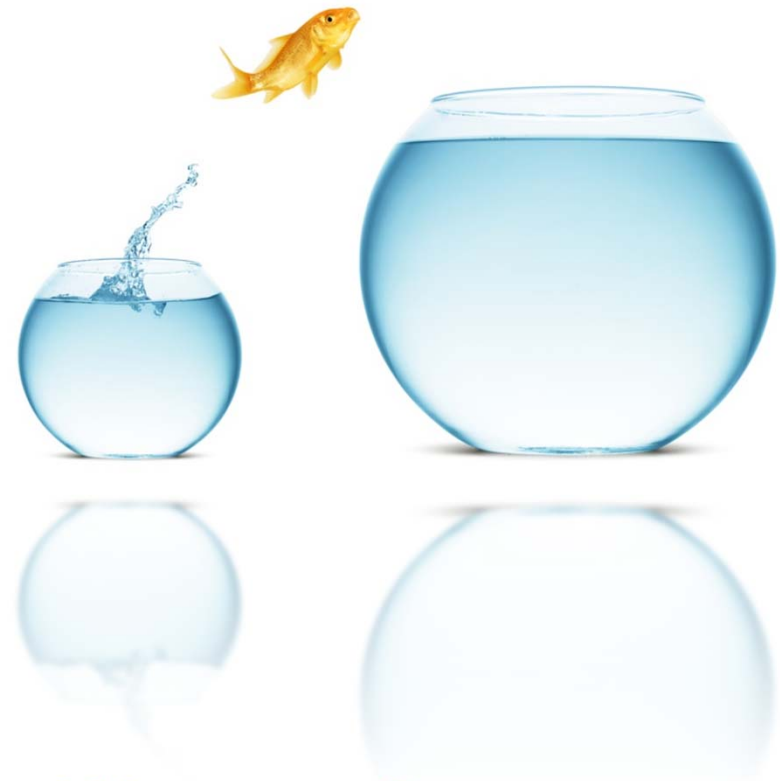
 <property name="openjpa.jdbc.SynchronizeMappings" value="buildSchema(ForeignKeys=true)"/>
 </properties>
 <property name="openjpa.DynamicEnhancementAgent" value="false"/>
 <property name="openjpa.Log" value="DefaultLevel=WARN, Runtime=INFO, Tool=INFO, SQL=TRACE"/>
 <property name="openjpa.DetachState" value="loaded(DetachedStateField=true)"/>
 <property name="openjpa.Compatibility" value="IgnoreDetachedStateFieldForProxySerialization=true"/>
 <property name="openjpa.jdbc.MappingDefaults"
 value="ForeignKeyDeleteAction=restrict, JoinForeignKeyDeleteAction=restrict"/>
 </properties>
</persistence-unit>
```

## Migration und Modernisierung

Es wird zu eng  
Der Support läuft aus



Mehr Flexibilität im neuen Hause

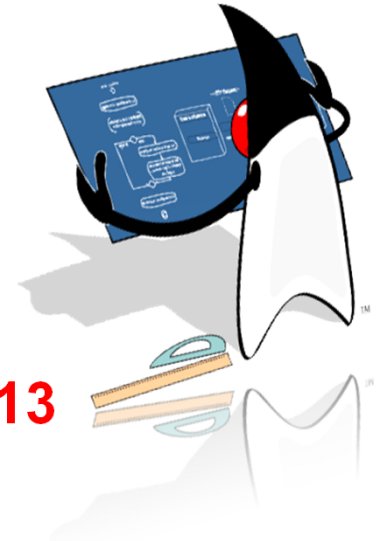


Doch wie sieht der Weg aus?

## Häufige Migrationsgründe: Support, Wartungskosten

### End-Off-Life Zyklus

- **ORACLE JSE 6.0.45 April 2013**
- **Apache Struts 1.3.10 April 2013**
- **Hibernate 3.2 Oktober 2011**
- **IBM WebSphere Application Server 6.1 September 2013**
- **JBoss Enterprise Application Server 4.3 Januar 2011**
- **JBoss Enterprise Application Server 5.x November 2013**
- **Apache Tomcat 5.5.36 Oktober 2012**
- **Apache Geronimo 2.1 Februar 2013**
- **ORACLE WebLogic Server 9.x Premier November 2011**
- **ORACLE WebLogic Server 10.3 Premier Januar 2014**
- **ORACLE GlassFish Server 3.0.x Dezember 2014**
- **ORACLE OC4J 10g R3 Premier Juni 2014**
- **JavaSE 8 ist für 2014 geplant**
- **Apache HTTP Server 2.0.65 Juli 2013**



## Was brauche ich wirklich?



## Ausblick

- TomEE ist für JavaEE zertifiziert und enthält viele bewährte Komponenten
- Der Ein-/Umstieg auf TomEE gelingt, schnell, da Tomcat-Know-How/-Werkzeuge weiterverwendet werden können
- Integrationsaufwände werden reduziert
- Etwas langsamere Releasezyklen, als bei Tomcat

## Weitere Informationen

- Apache Geronimo: Serverhandbuch, Frank Pientka, DPunkt-Verlag, 01/09
- Boss der Apachen, Frank Pientka, iX kompakt, 01/09
- Status quo: GlassFish, JBoss und Geronimo, Frank Pientka, JM 12/10
- Apache Tomcat 7: Katzenleben, Frank Pientka, JS 05/11
- Dreikampf, Frank Pientka, Java aktuell, 02/2012
- Geronimo 3.0: modulare Hybride fahren gut, Frank Pientka, Java aktuell, 01/2013
- <http://tomee.apache.org/refcard/refcard.html>
- TomEE Website: <http://tomee.apache.org>



2.– 5. September 2013  
in Nürnberg



# Herbstcampus

Wissenstransfer  
par excellence

**Vielen Dank!**

Frank Pientka

MATERNA GmbH