

# Tutorial

## Dynamic Server Applications with Eclipse RT

```
GreetingsResource.java
package hello.jaxrs;

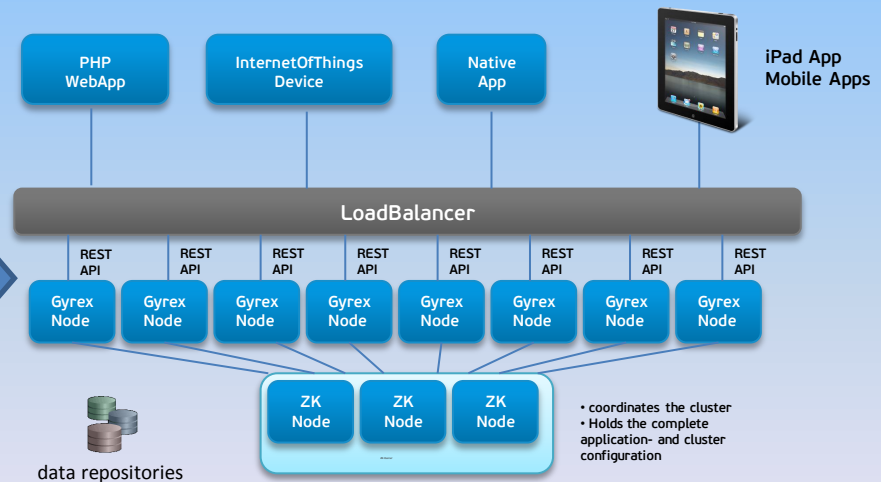
import hello.service.GreetingService;

/**
 * Hello World JAX-RS Resource
 */
@Path("/greetings")
public class GreetingsResource {

    // inject OSGi service using JAX-RS @Context
    @Context
    private GreetingService greetingService;

    // inject OSGi service using JSR 330 @Inject
    @Inject
    private GreetingService greetingService2;

    @Context
    private UriInfo uriInfo;
}
```



# The authors

Gunnar Wagenknecht  
[gunnar@wagenknecht.org](mailto:gunnar@wagenknecht.org)  
CTO  
AGETO, Germany



Andreas Mihm  
[andreas.mihm@ageto.net](mailto:andreas.mihm@ageto.net)  
Head of Research  
AGETO, Germany



Jochen Hiller  
[jo.hiller@googlemail.com](mailto:jo.hiller@googlemail.com)  
Developer Evangelist  
Deutsche Telekom AG, Germany



# In this tutorial you will

- **LEARN** what modern server applications need to feature
- **UNDERSTAND** the usage of selected EclipseRT technologies
- **BUILD** an EclipseRT server application based on Equinox and Eclipse Gyrex
- **SETUP** a private cluster connecting all attendees



# Modern Server Application Requirements

- Support high traffic use cases
- Support different frontend technologies and devices
- modular in development and deployment
- Easy to setup
- Open for new technologies → easy to extend AND to shrink
  - e.g. persistence

Million  
transactions  
per hour



# Eclipse RT

“EclipseRT is the collection of OSGi-based runtimes and frameworks built by the Eclipse open source projects. ”

Containers, Middleware, EnterpriseFrameworks

eclipse)link

*jetty://*

equinox  
OSGI



GYREX

eclipseRT



# Equinox

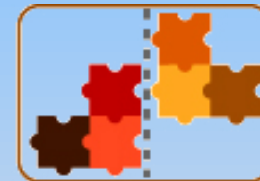
- reference implementation for OSGi spec
- foundation of EclipseRT
- component oriented development and assembly



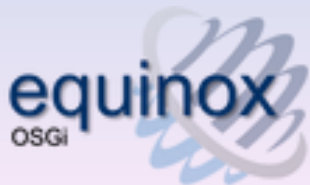
*Create*



*Extend*



*Assemble*



# Eclipse jetty

## features

- Asynchronous HTTP Server
- Standard based Servlet Container
- Web Sockets server
- Asynchronous HTTP Client
- OSGi, JNDI, JMX, JASPI, AJP support

Small foot print allows large scaling

## Jetty runs in

- Apache Hadoop
- Google AppEngine

The logo for Eclipse Jetty, featuring the word "jetty" in a bold, italicized, orange font with a black outline, followed by "://" in a similar style.


# EclipseLink

Comprehensive Java persistence solution  
addressing relational, XML, and database web services.



  
Databases

  
XML Data

  
Legacy Systems





# GYREX

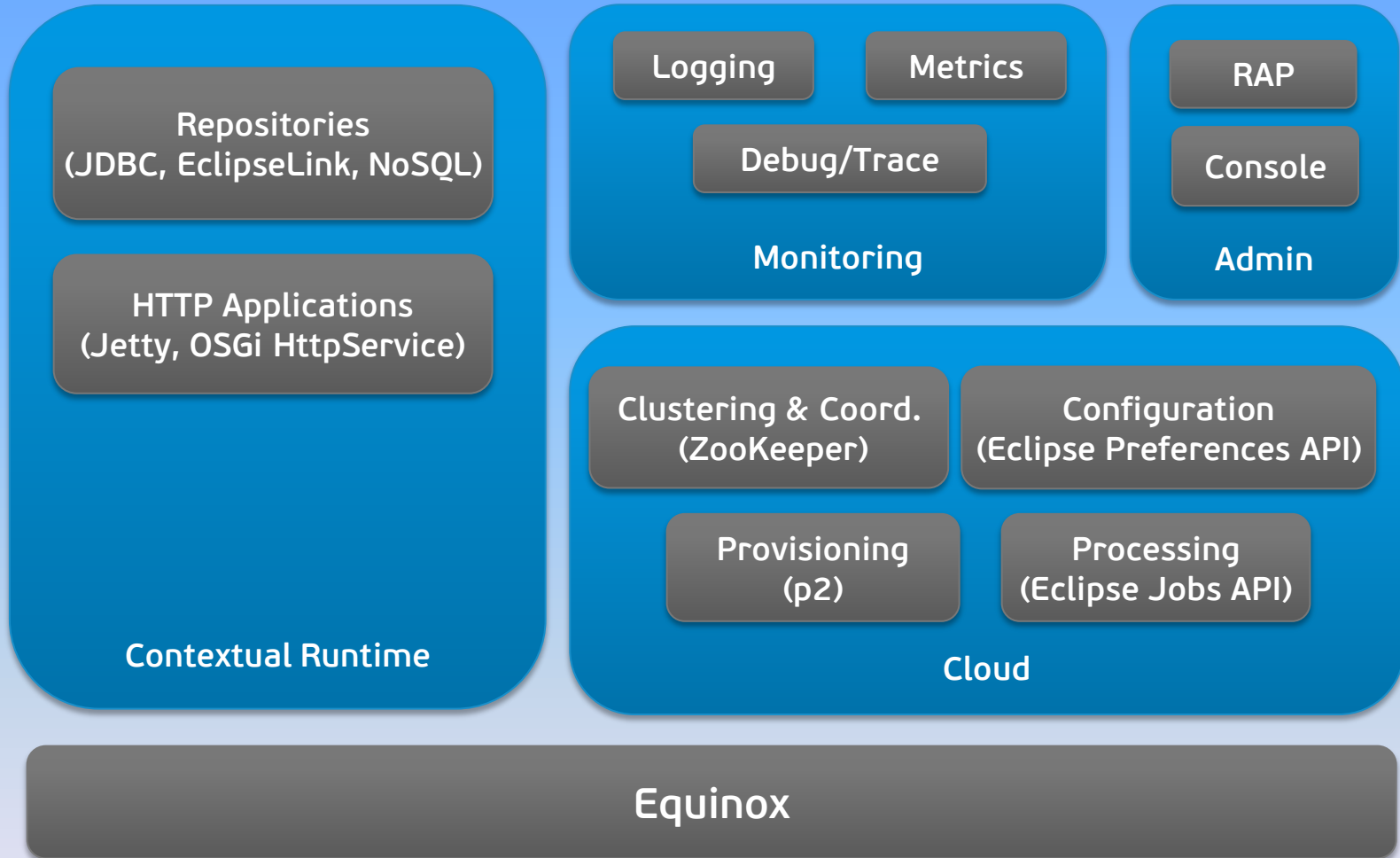
lightweight application stack for building server applications using EclipseRT technologies.

- built-in clustering
- built-in web-based administration UI
- built-in multi tenancy
- enhancements for professional maintenance
  - centralized logging
  - cluster provisioning UI

# GYREX



# GYREX Components

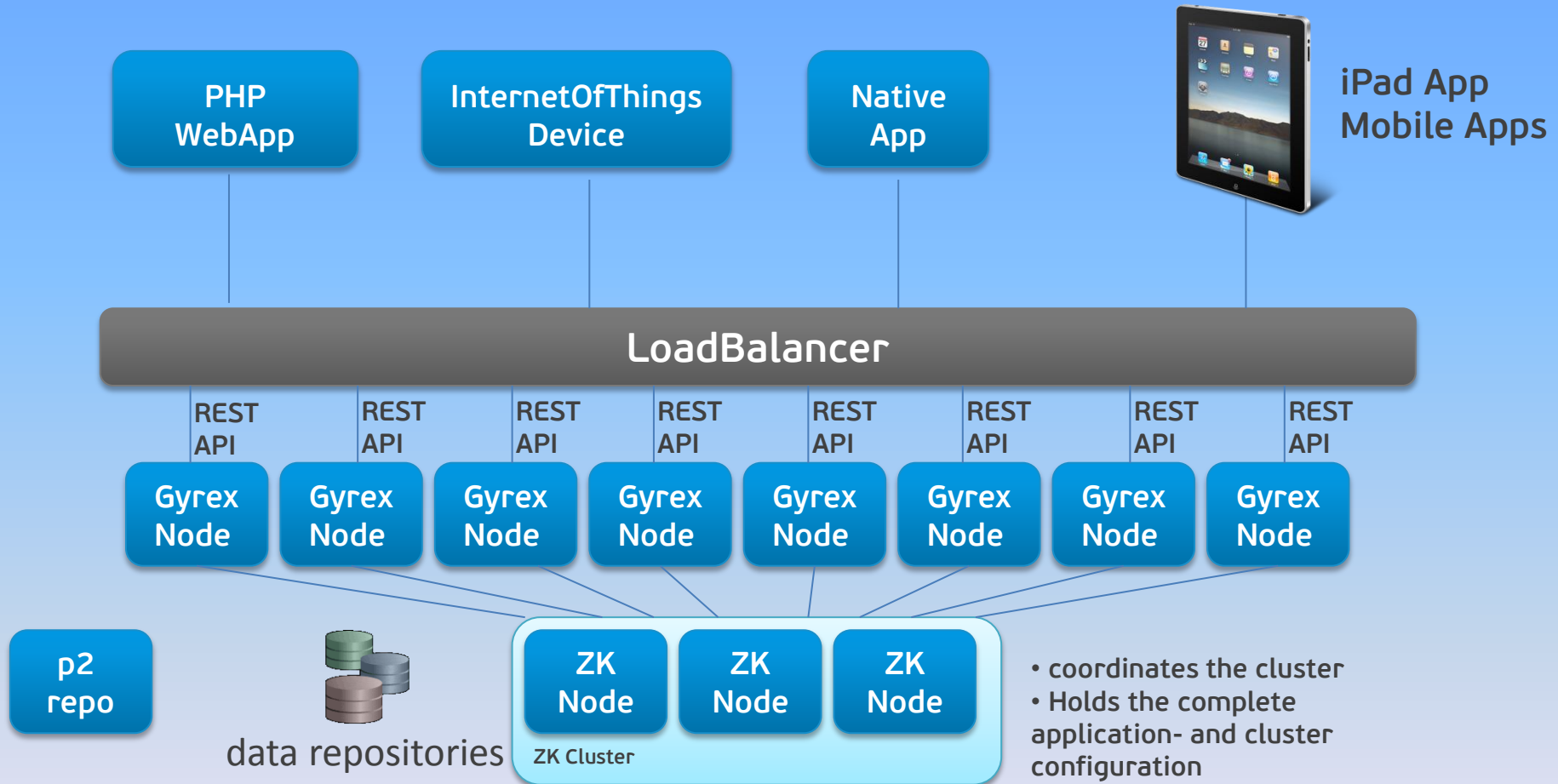


GYREX



# GYREX Infrastructure Setup

## For a High Traffic Application with different frontends

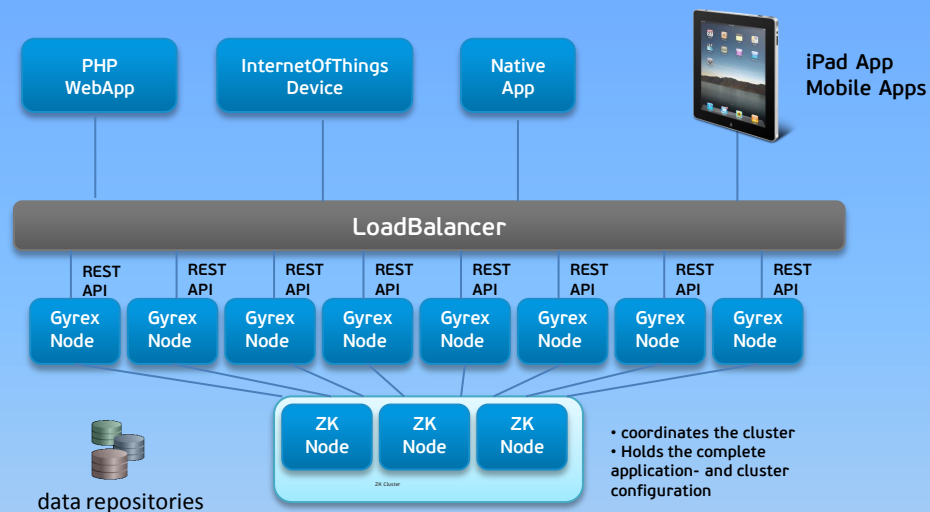


# GYREX



# GYREX Features

- light weight application stack
- fast 100% OSGi runtime
- central cluster configuration through zookeeper
- cluster aware job scheduling
- multi tenant application support
- automated deployment through p2
- support for cluster node roles, e.g. „job worker node“ and „api node“



# GYREX



# Exercises

- Setup workspace and target platform
- Simple OSGi Service implementation
- JAXRS example application
- Private cluster setup

# Setup Steps

- Copy USB-stick content to local disk
- Extract fresh Eclipse installation from archive
- Copy samples folder into eclipse/dropins
- Start Eclipse with new workspace
- Open Java Perspective
- Import targetplatform project  
(File->Import->Existing Projects)
- Set LOCAL TargetPlatform



# For more information

- <http://www.eclipse.org/gyrex/>
- <http://www.eclipse.org/equinox/>
- <http://www.eclipse.org/rt/>




# Give Feedback on the Sessions

1

Sign In: [www.eclipsecon.org](http://www.eclipsecon.org)

2

Select Session Evaluate

Making ALM Work -   
Transform your Application Lifecycle Management to Foster Innovation (presented by HP)  
Ronit [HP]  
**EVALUATE**

3

Vote





**Thank You  
for your attention**

