



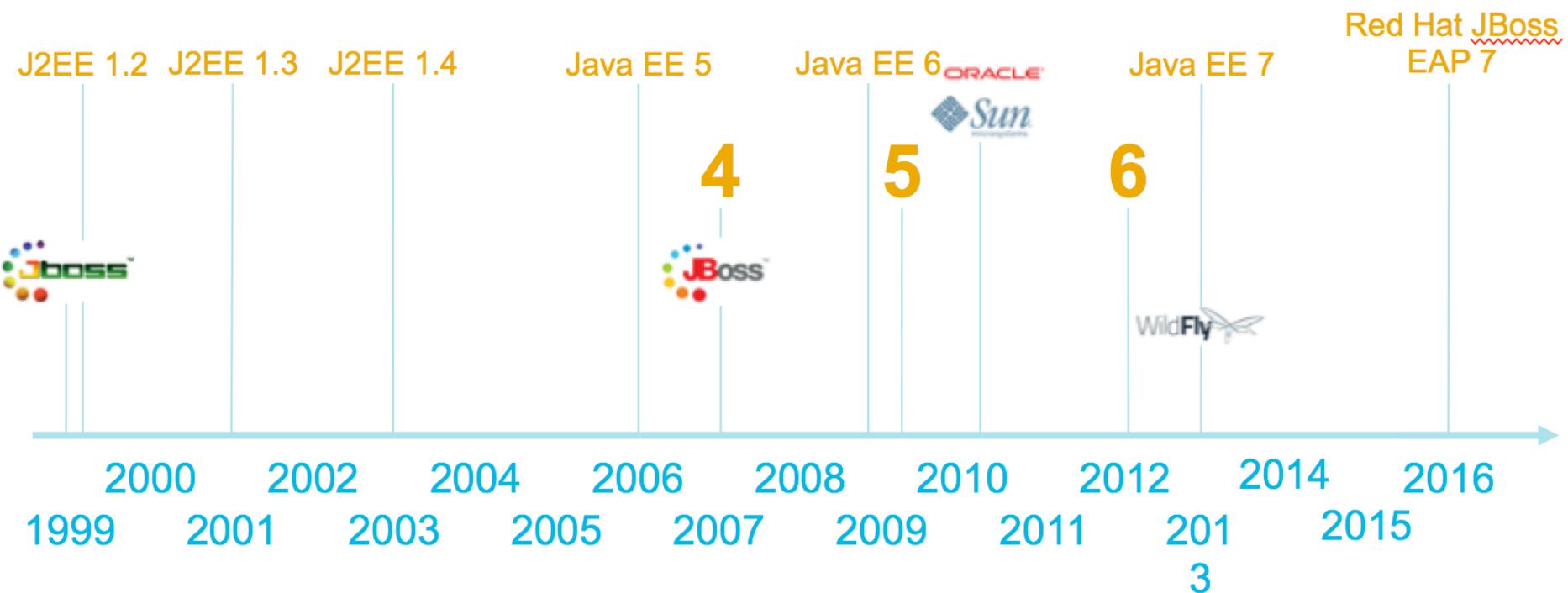
redhat.

## The Future of Java EE

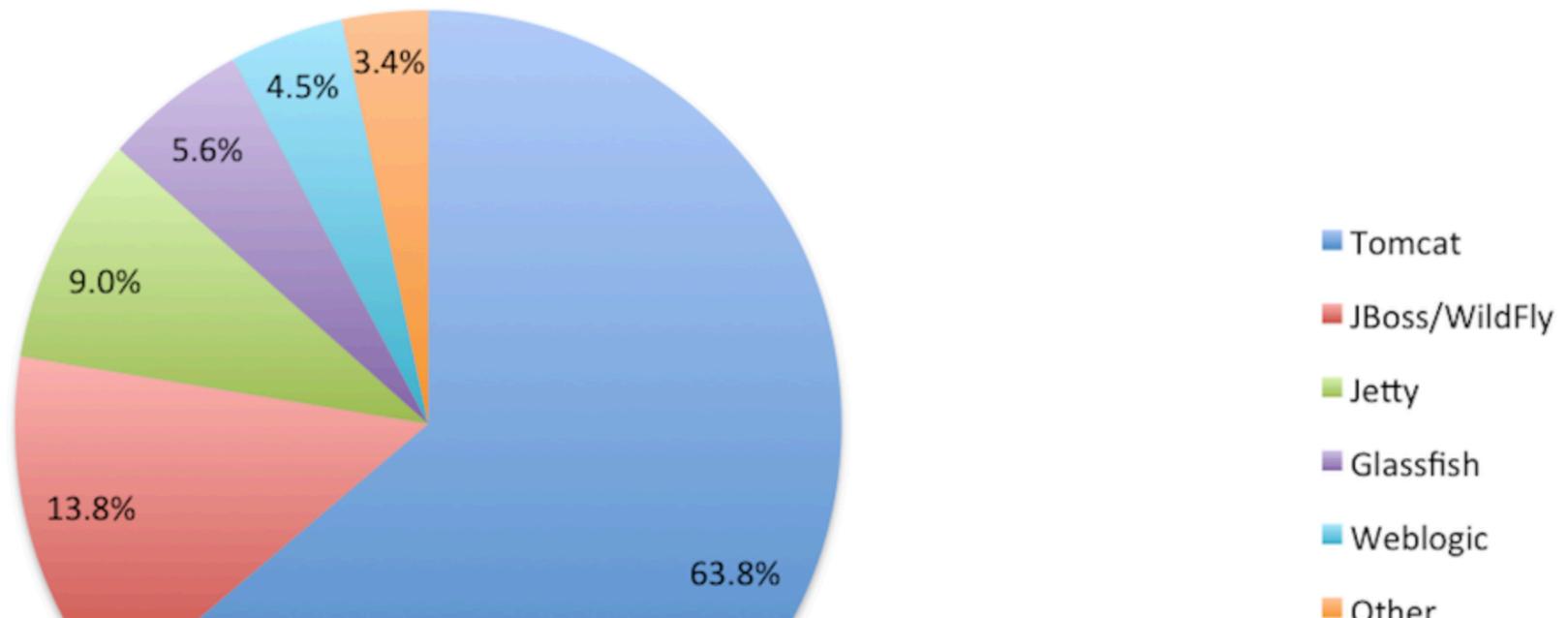
### JCP EC F2F Meeting, January 2018

Dr Mark Little, VP Engineering

# JBoss EAP – Enterprise Open Source Pioneer



## Java application server market 2017

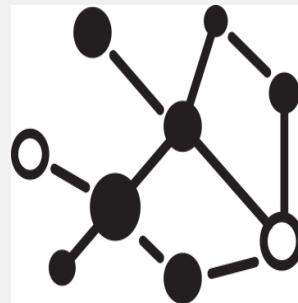
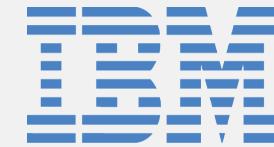


# Java EE and 2017





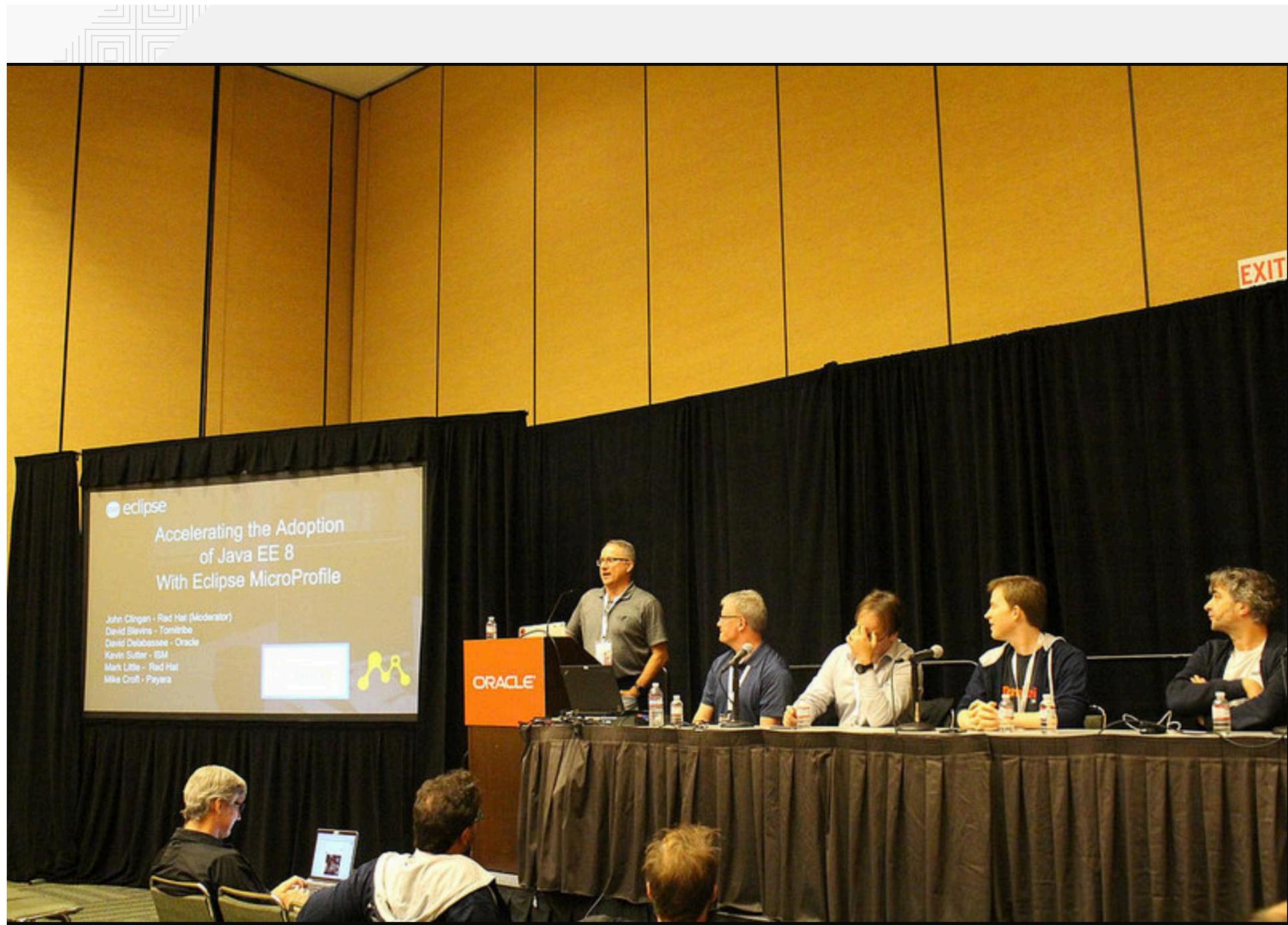
redhat.



MicroProfile

Tomitribe



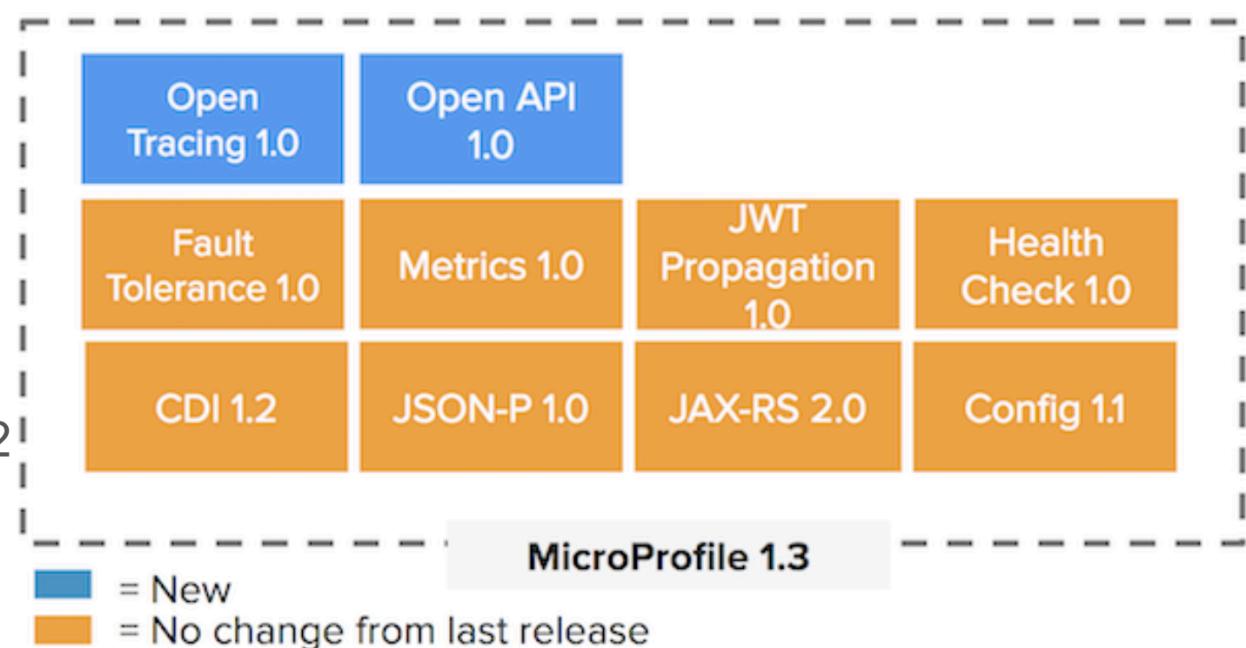


# Eclipse MicroProfile Evolution

- MP 1.0 - Sep/16
- EMP 1.1 - Jul/17
- EMP 1.2 - Oct/17
- EMP 1.3 - Jan/18

## WF-Swarm

- v2017.12.1 => EMP v1.2
- v1.3 by Summit



# Eclipse EE4J

Eclipse Enterprise for Java (EE4J)  
Moving Java EE to Eclipse Foundation



Community  
and  
Vendors

Technology

Sponsorship



- ✓ Agile
- ✓ Flexible
- ✓ Open
- ✓ Compatible

*Join the discussion at ee4j-community@eclipse.org*

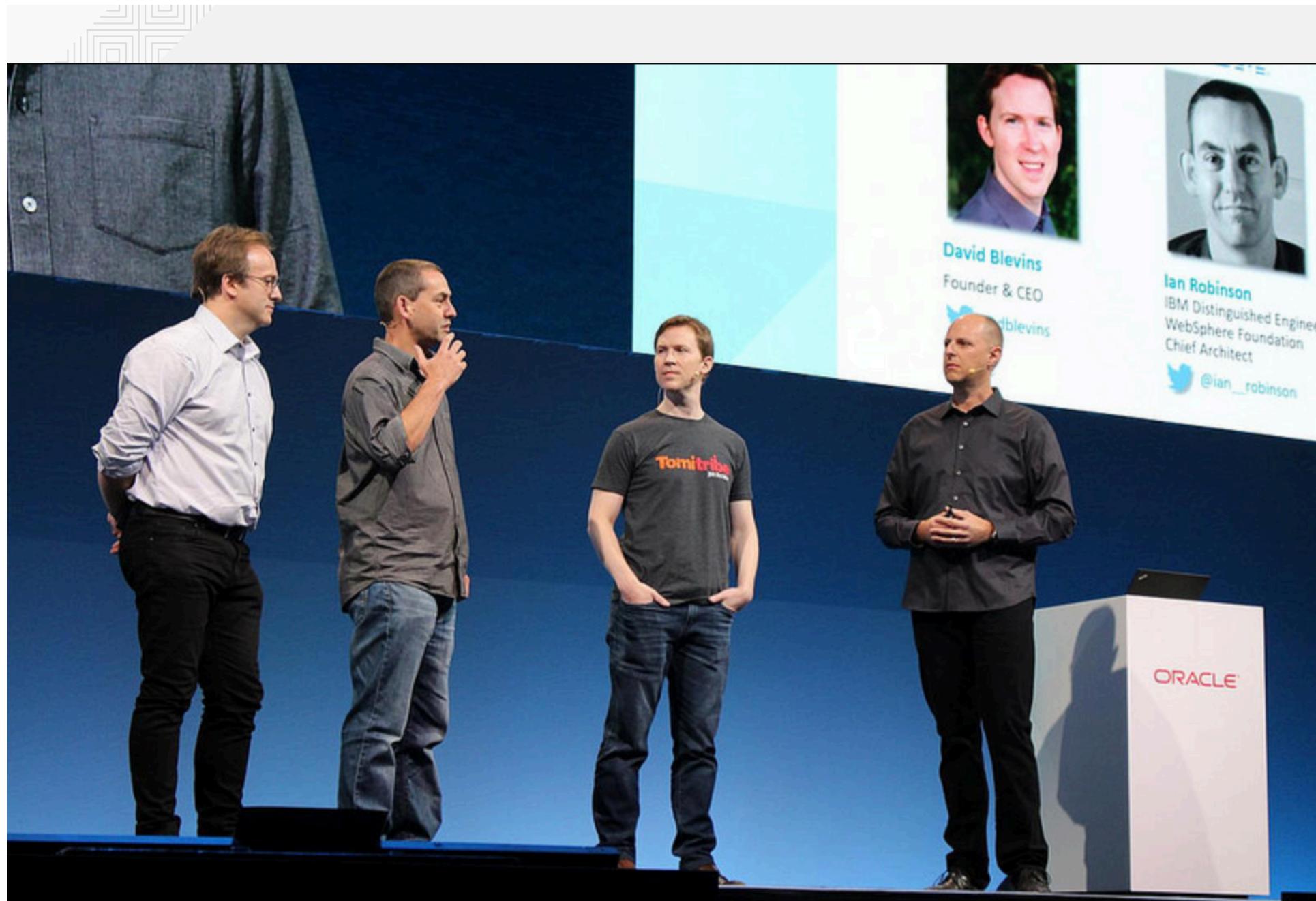


Copyright © 2012, Oracle and/or its affiliates. All rights reserved. |

46

E

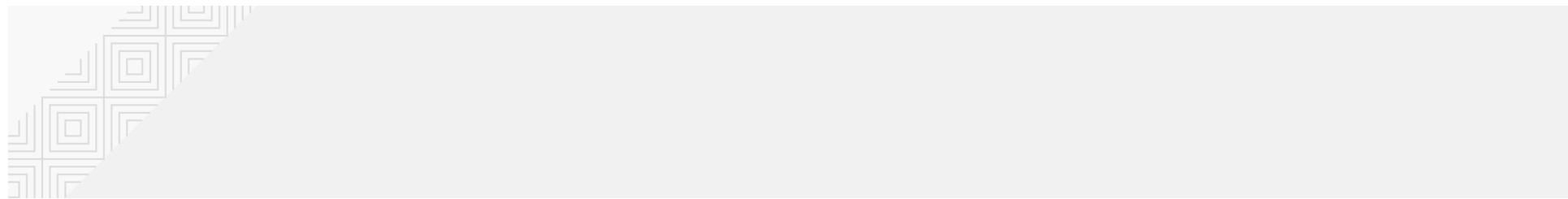




**David Blevins**  
Founder & CEO  
[@dblevins](https://twitter.com/dblevins)



**Ian Robinson**  
IBM Distinguished Engineer  
WebSphere Foundation  
Chief Architect  
[@ian\\_\\_robinson](https://twitter.com/ian__robinson)



Google Custom Search



GETTING STARTED    MEMBERS    PROJECTS    MORE ▾

★ DONATE

[HOME](#) / [PROJECTS](#) / THE ECLIPSE ENTERPRISE FOR JAVA PROJECT TOP LEVEL PROJECT CHARTER

# The Eclipse Enterprise for Java Project Top Level Project Charter

Please see the [Frequently Asked Questions](#).

## Overview

Eclipse Enterprise for Java (EE4J) is an open source initiative to create standard APIs, implementations of those APIs, and technology compatibility kits for Java runtimes that enable development, deployment, and management of server-side and cloud-native applications. EE4J is based on the Java™ Platform, Enterprise Edition (Java EE) standards, and uses Java EE 8 as the baseline for creating new standards.

Although Java EE was developed in open source, often the Java EE development process was not perceived as nimble, flexible or open enough when compared to other open source projects and processes. EE4J enables the use of more nimble processes, more flexible licensing, and a more open governance process for evolution of the platform.

This charter was developed in accordance with the [Eclipse Development Process](#) and will outline the mission of the EE4J Project. This document extends the Eclipse [Standard Top-Level Charter v1.2](#), and includes the required content and overrides that follow. It is anticipated that as the standard charter is updated, this charter will incorporate the changes and make adjustments as seen fit by the PMC, and with approval from the EMO and board of directors.



# Future of Java EE and JCP

- Java EE evolving to Eclipse EE4J is only part of the solution
- Open standards are not sufficient
- Need Eclipse EE4J to adopt standards processes
- Future of EE4J and JCP tied
- <https://www.infoq.com/news/2018/01/no-jcp-for-javeee>
- Potentially huge future for EE4J
  - Asynchronous programming
  - Cloud-native enterprise Java
  - Community and experience driven
  - Rapid releases and innovation

# Conclusion

- The future of Java EE looks bright but it is not assured
  - <https://middlewareblog.redhat.com/2017/09/11/the-future-of-java-ee/>
- Java and Java EE dominate the non-Microsoft software landscape and no reason that shouldn't continue
- We must rally the vendors and communities behind Eclipse EE4J and Eclipse MicroProfile
  - If it falls on the main Java EE vendors to drive then it won't succeed
- The JCP has a role to play in helping to evolve Eclipse processes