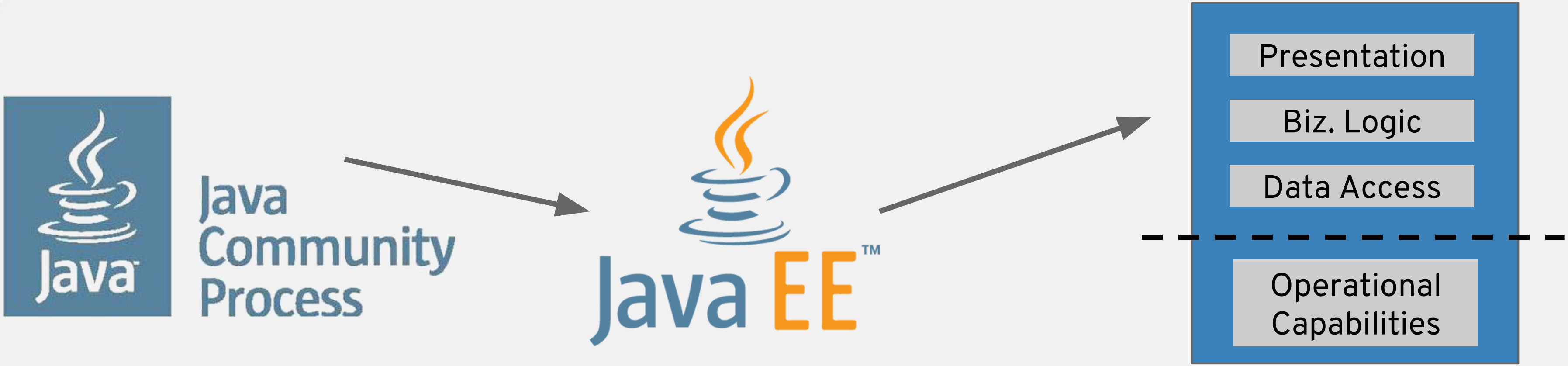
redhat.

The Future of Enterprise Java

MicroProfile & Jakarta EE Roadmap

Overview



Java Community Process

Java EE™

Presentation

Biz. Logic

Data Access

Operational Capabilities

Oracle-led community
IBM, Red Hat, ...

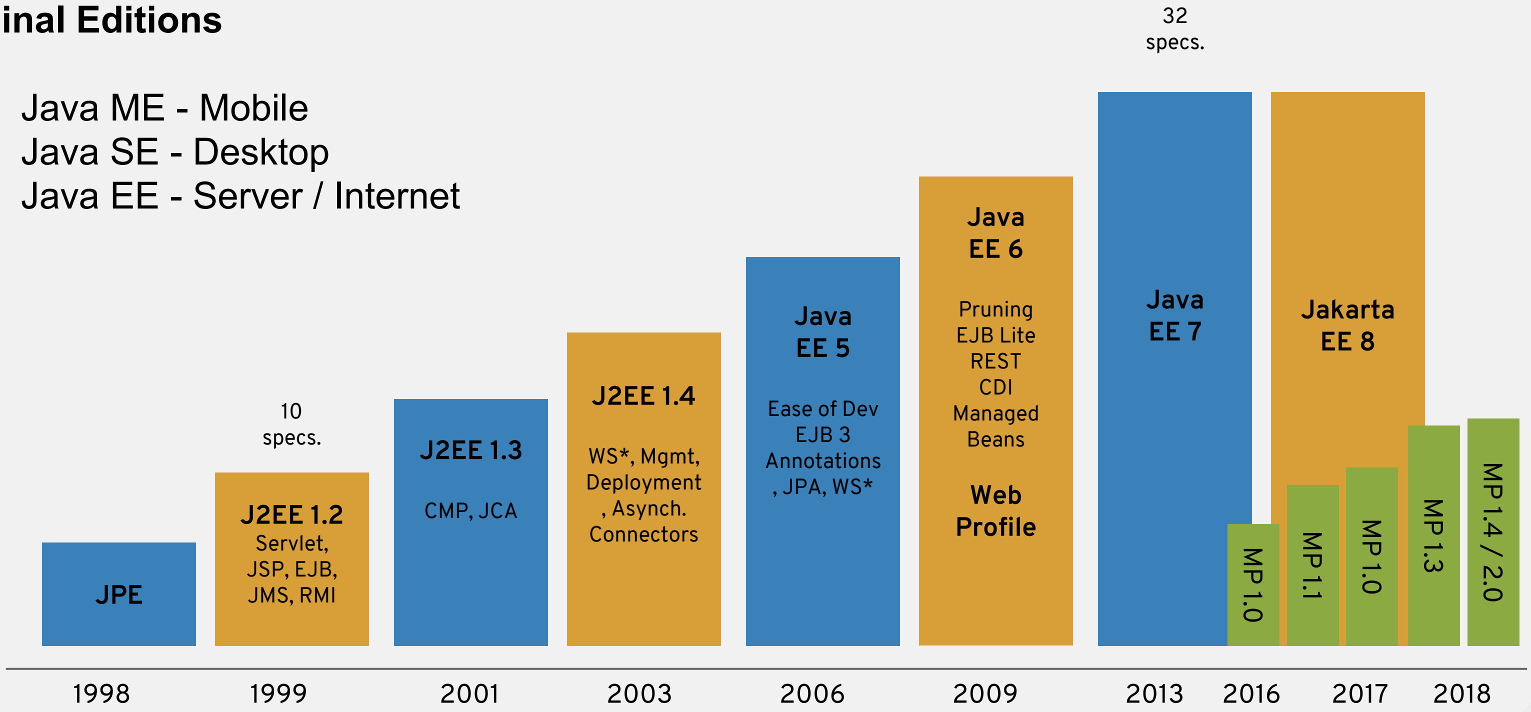
Standard Specification
Certification Test Suite
Trademark / Brand Usage
Licensed by vendors

Many different implementations, business and licensing models

A Brief History of Java EE

Original Editions

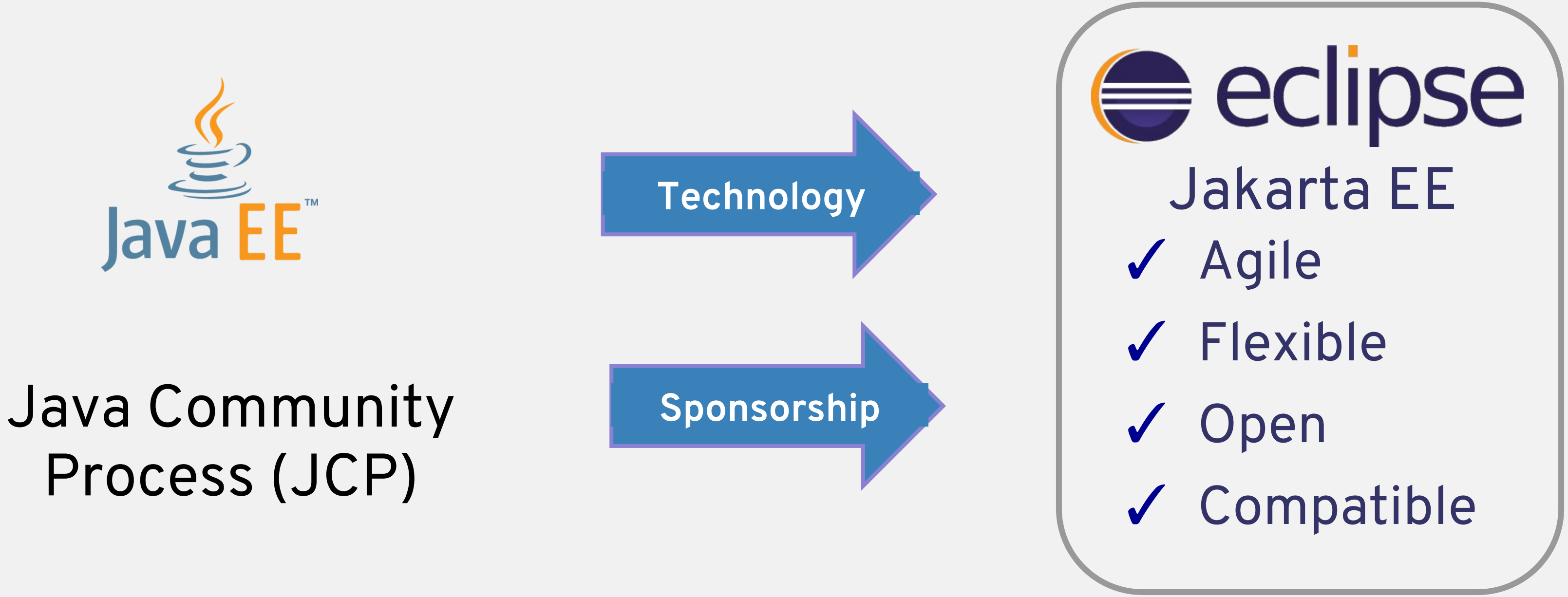
- Java ME - Mobile
- Java SE - Desktop
- Java EE - Server / Internet



Java EE - the challenges

- Big, bloated and expensive
  - mostly perception based on older, proprietary WAS / WLS
  - new breed of lightweight, modular servers (JBoss, Liberty, Glassfish, TomEE) hasn’t done much to change perception
  - the new lightweight is measured in KB (Go, node.js)
- Slow Moving
  - Missed Mobile, IoT, MSA, etc.
- Not fully open
  - Dependence on single vendor
  - Complex licensing, licensed from competing vendor

Moving Java EE to the Eclipse Foundation



Java EE™

Java Community Process (JCP)

Technology

Sponsorship

eclipse

Jakarta EE

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

ORACLE® payara® IBM Tomitribe™ join the tribe redhat. FUJITSU

JBoss EAP remains Red Hat’s Only Java EE (Jakarta EE) compatible implementation

Java EE specification		Community project		Product
J2EE 1.4	→	JBoss AS 4	→	JBoss EAP 4
Java EE 5	→	JBoss AS 5	→	JBoss EAP 5
Java EE 6	→	JBoss AS 7	→	JBoss EAP 6
Java EE 7	→	WildFly 8,9,10 / 11	→	JBoss EAP 7 / 7.1
Java EE 8	→	WildFly 12,13,14	→	JBoss EAP 7.2
Jakarta EE 8	→	WildFly 15,16	→	JBoss EAP 7.3
Jakarta EE 9	→	WildFly 17,18,19	→	EAP-NEXT(The Runtime)

MICROPROFILE™

OPTIMIZING ENTERPRISE JAVA

eclipse

payara FUJITSU Tomitribe join the tribe
IBM redhat.
LJC HAMMOCK SOUJava sociedade de usuários java Lightbend
Microsoft hazelcast ORACLE

Vendors, Organizations, Individuals

Open Tracing 1.1	Open API 1.0	Rest Client 1.1	Config 1.3
Fault Tolerance 1.1	Metrics 1.1	JWT Propagation 1.1	Health Check 1.0
CDI 2.0	JSON-P 1.1	JAX-RS 2.1	JSON-B 1.0

MicroProfile 2.0

- Microservices specifications for Enterprise Java
- Industry collaboration within the Eclipse Foundation
- 6+ implementations
- Rapid release cadence

Thorntail (WildFly Swarm) is our MicroProfile Implementation

- uber-jar - build the server around your application / service
- generators for easy getting started
- built on battle-tested technology from WildFly (JBoss EAP)
- integrations with KeyCloak, istio, Netflix,

Develop Monoliths and Microservices with JBoss EAP and Thorntail

JBoss EAP

- Extend existing apps
- Full Java EE / Jakarta EE

- Deploy on or off OpenShift
- Domain mode, high availability clustering

Thorntail

- Develop new microservices
- Web Profile and MicroProfile

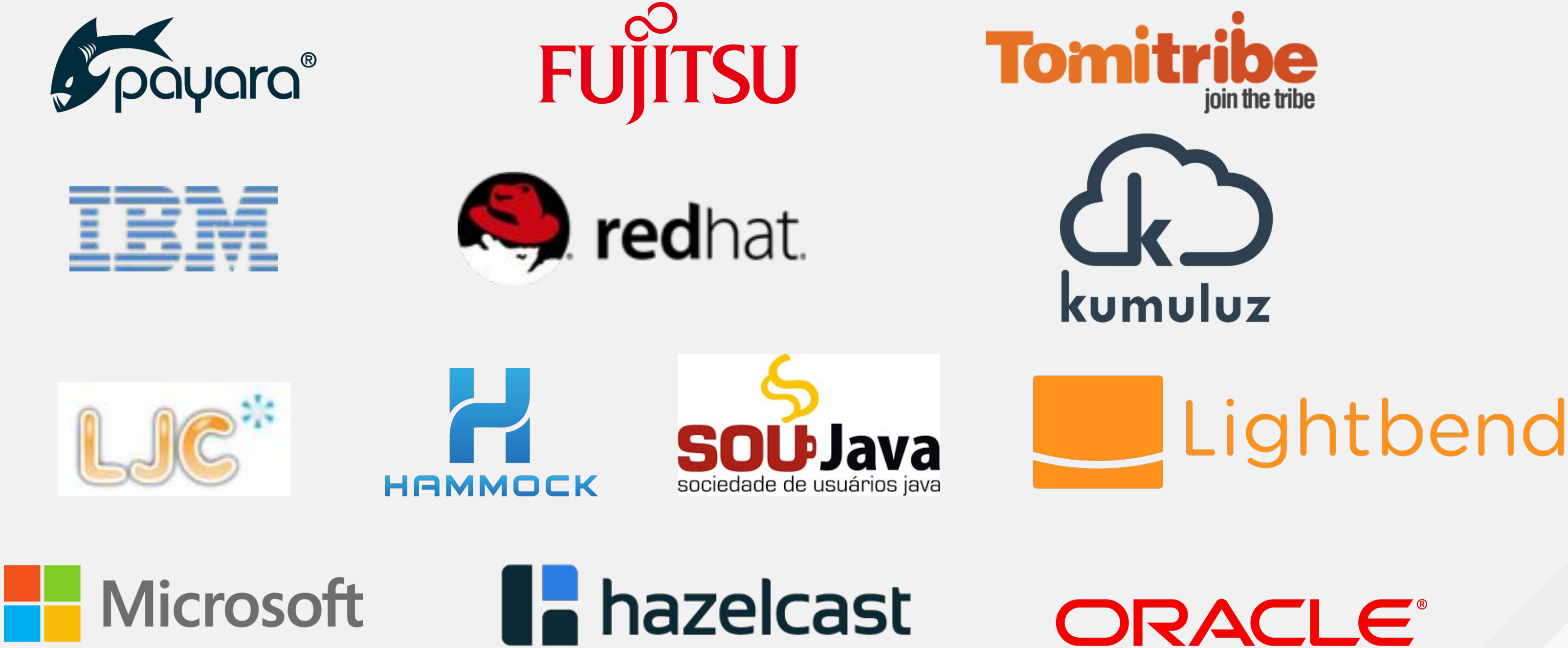
- Optimized for OpenShift
- Self-contained uber-jar deployments

Eclipse MicroProfile

What is Eclipse MicroProfile?

- Eclipse MicroProfile is an open-source community specification for Enterprise Java microservices
- A community of individuals, organizations, and vendors collaborating within an open source (Eclipse) project to bring microservices to the Enterprise Java community

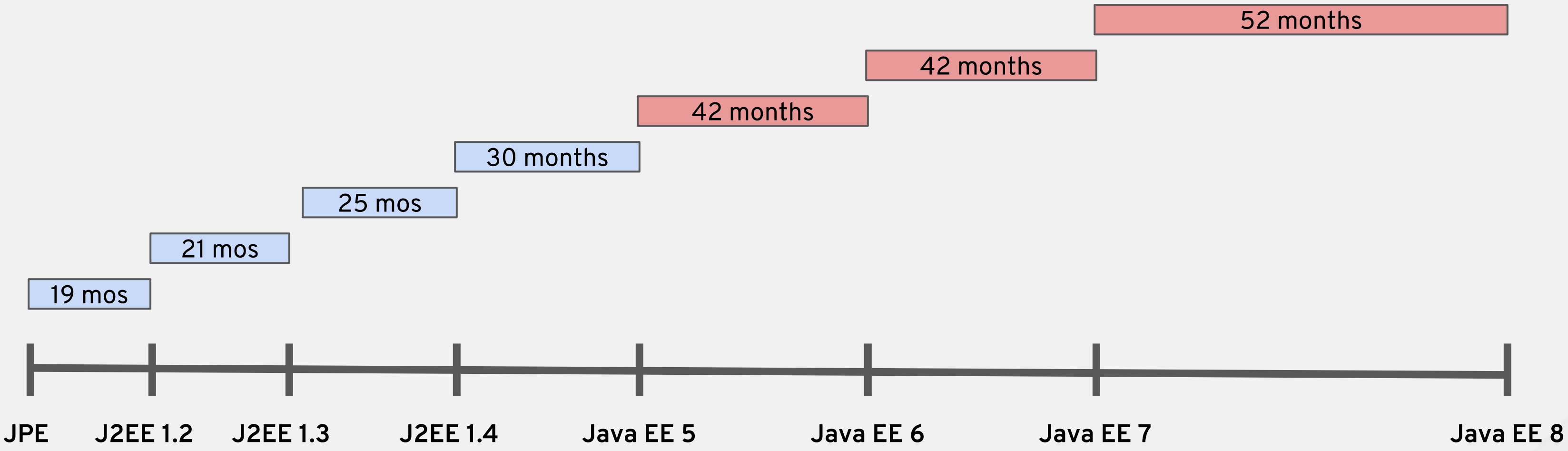
Community - individuals, organizations, vendors



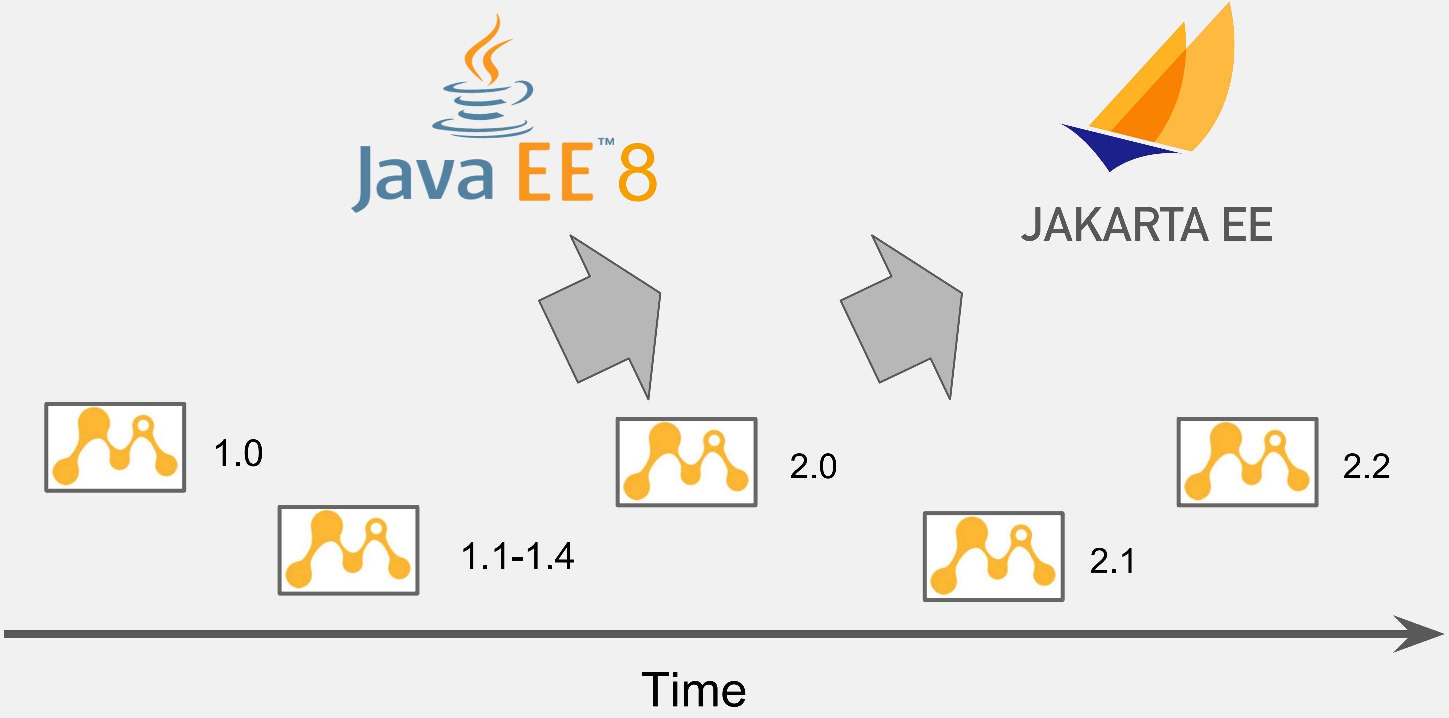
Innovation vs. Standardization

Eclipse MicroProfile	Java Community Process
(Open Source) Project	Standards organization
Incremental feature release	Large multi-feature releases
Community controls pace	Spec Lead controls pace

Time between Java EE releases



Accelerating Adoption of Microservices



1.0

1.1-1.4

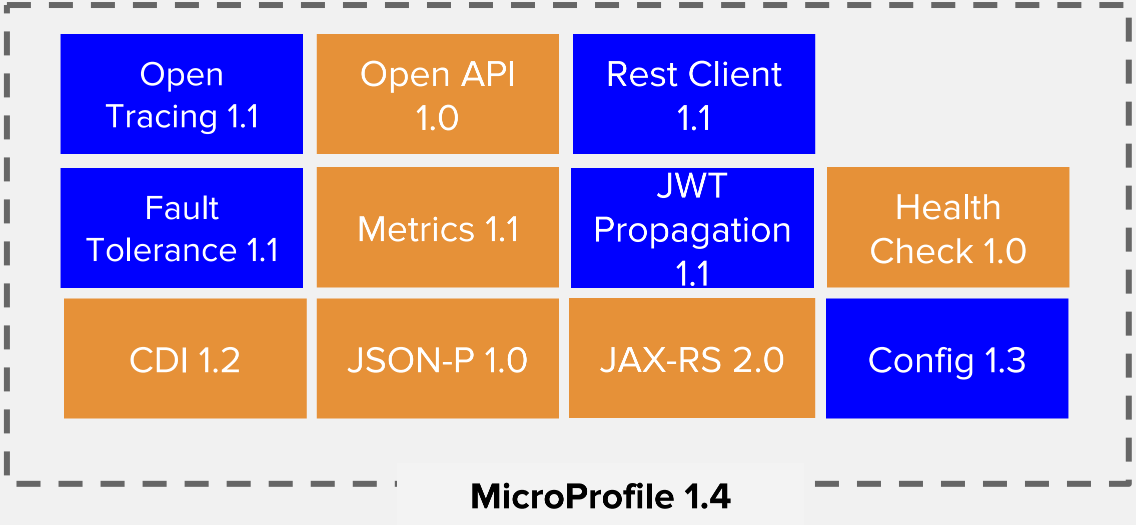
2.0

2.1

2.2

Time

Eclipse MicroProfile 1.4 (Jun, 2018)

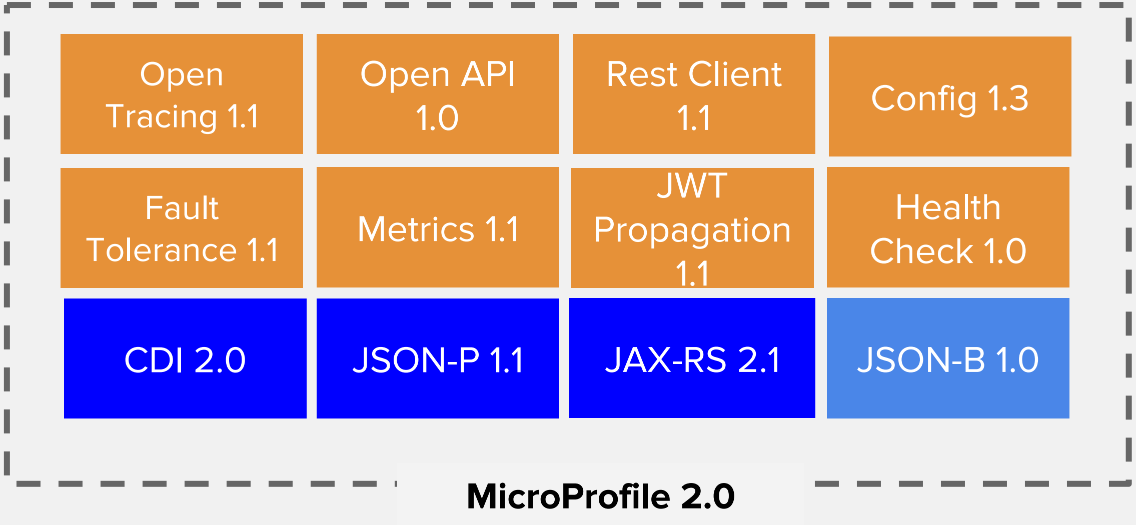


🟦 = New

🟦 = Updated

🟧 = No change from last release

Eclipse MicroProfile 2.0 (Jun, 2018)



Open Tracing 1.1	Open API 1.0	Rest Client 1.1	Config 1.3
Fault Tolerance 1.1	Metrics 1.1	JWT Propagation 1.1	Health Check 1.0
CDI 2.0	JSON-P 1.1	JAX-RS 2.1	JSON-B 1.0

MicroProfile 2.0

🟦 = New

🟦 = Updated

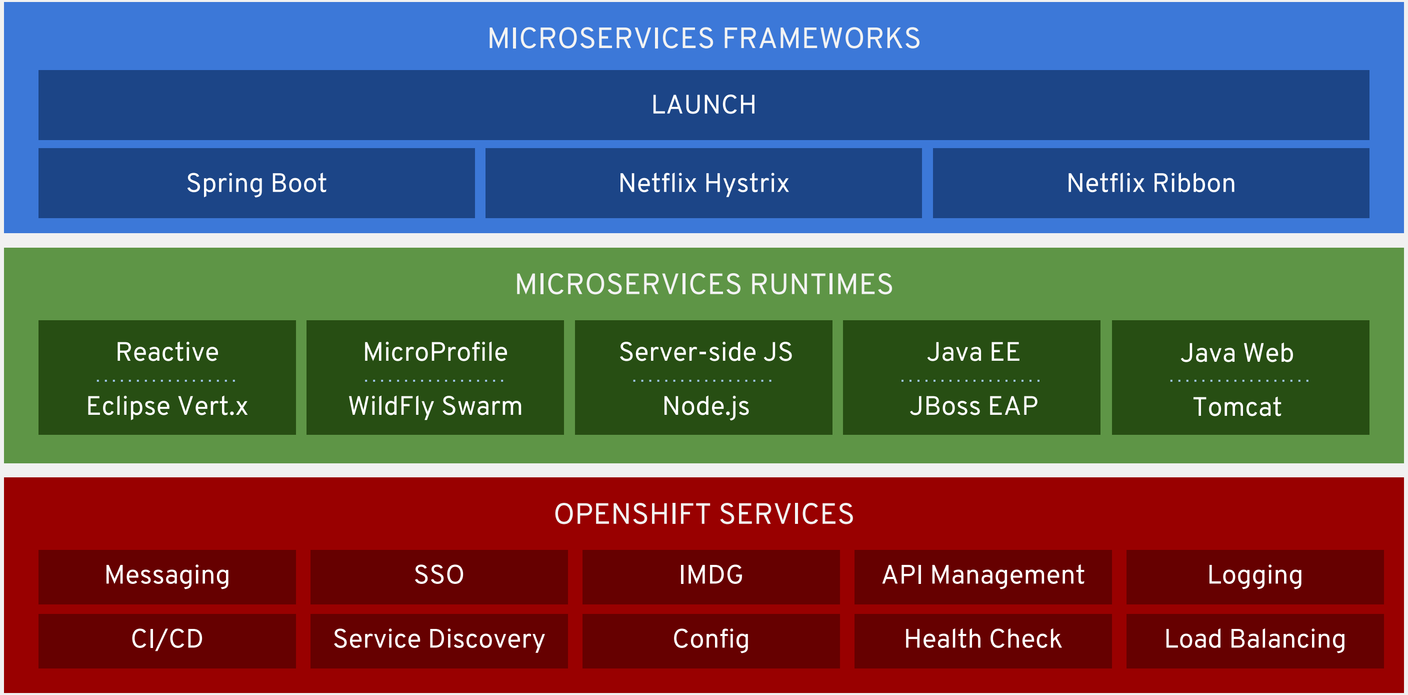
🟧 = No change from last release (MicroProfile 1.4)

OPENSHIFT

APPLICATION RUNTIMES

Application runtimes and a prescriptive developer experience for organizations that are moving beyond Enterprise Java and embracing cloud-native application development.

RED HAT® OPENSHIFT
Application Runtimes



MICROSERVICES FRAMEWORKS

LAUNCH

Spring Boot

Netflix Hystrix

Netflix Ribbon

MICROSERVICES RUNTIMES

Reactive	MicroProfile	Server-side JS	Java EE	Java Web
Eclipse Vert.x	WildFly Swarm	Node.js	JBoss EAP	Tomcat

OPENSHIFT SERVICES

Messaging	SSO	IMDG	API Management	Logging
CI/CD	Service Discovery	Config	Health Check	Load Balancing

WildFly Swarm (Thorntail)

- Drives our Eclipse Microprofile implementation efforts
- WildFly Swarm (Thorntail) a runtime included in RHOAR
- Promoting shared API implementations across projects (SmallRye)
- Researching further runtime reduction techniques
- Will first add MicroProfile 1.4 support, then MicroProfile 2.0
- WildFly Swarm (Thorntail) is targeting Java EE Web Profile technologies and MicroProfile and is strictly focused on Cloud Native / microservices development

Jakarta EE

Transition Java EE 8 to Jakarta EE in CY2018

- Jakarta EE Working Group Charter published
- Open process
- Collaboration: community, vendors, Eclipse
- Key deliverables:
  - Eclipse Glassfish 5.1
  - Java EE 8 RIs and TCKs contributed
  - Process for existing and new specs
  - Compatibility process
- Better integration with Eclipse MicroProfile
- Continuity for Java EE community

eclipse

Jakarta EE

✓ Agile

✓ Flexible

✓ Open

✓ Compatible

Jakarta EE Working Group

- Jakarta EE WG is an industry consortium that:
  - Approves specifications
  - Manages the Jakarta EE brand
  - Establishes technical roadmap
  - Ensures compatibility via open source TCKs
- All stakeholders (enterprises, vendors, SI’s, community) have seat at the table
- New, open processes to enable rapid open innovation
- Compatibility ensured through certification

Jakarta EE Working Group Founding Members

Strategic Members

FUJITSU

IBM

ORACLE®

payara®

redhat.

Tomitribe™ join the tribe

Participating Members

Lightbend

Pivotal®

w://

Microsoft

SAP®

Jakarta EE Technical Roadmap - 2018

- ~40 new Eclipse projects being created for existing reference impl’s

  Culminates in release of **Eclipse Glassfish 5.1**
- Two major milestones:
  - Q3-2018: Java EE 8 certified
  - Q4-2018: Jakarta EE 8 certified
  - This is a new certification process being defined
- All new development will be done under Jakarta EE brand moving forward from this point

Crowdsource logo/brand

Technical Vision

As the future of Cloud Native Java, Jakarta EE will:

- Help developers create portable cloud native applications
- Deliver faster innovation
- Lower barriers to participation

To support this vision we will value:

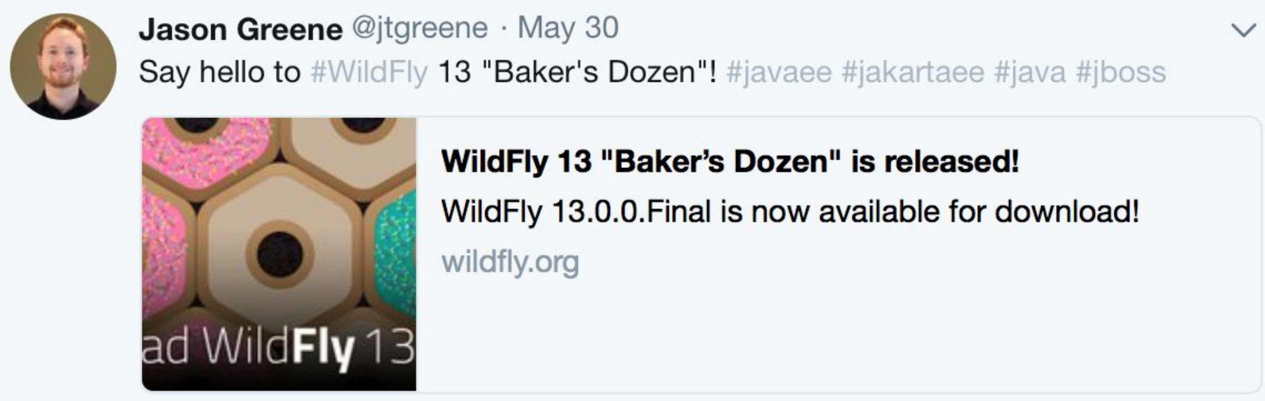
- Open standards
- A community-based specification process, building on the experience of real world deployments and developers
- Adopting Java innovations from open source communities like Eclipse MicroProfile into new versions of the platform

WildFly.Next

- Moving to Quarterly timeboxed releases
- Merge only completed features to main branch
- Formalize and enforce topic-branch testing
- Evolve our Kanban continuous feature delivery process
- Emphasis on fast delivery of Java EE 8 and refocus on Jakarta EE

WildFly 13 - May 2018

- Second release following our new quarterly delivery model
- Java EE 8 functionality is fully completed
- WildFly 14 will focus on achieving formal certification for Java EE 8.
- Since Java EE 8 is fully backwards compatible, Java EE 7 deployments will still function as they do today



Jason Greene @jtgreene · May 30

Say hello to #WildFly 13 "Baker's Dozen"! #javaee #jakartaee #java #jboss

WildFly 13 "Baker's Dozen" is released!

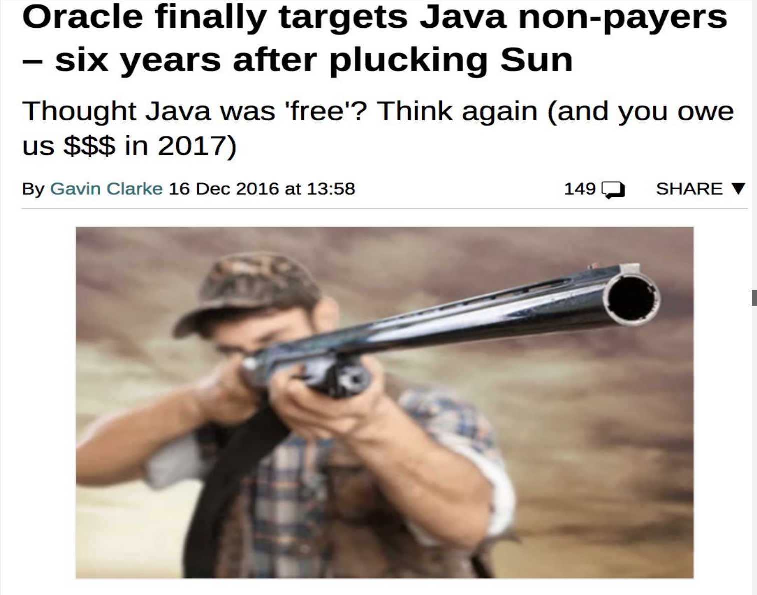
WildFly 13.0.0.Final is now available for download!

wildfly.org

WildFly Tentative Roadmap

- WildFly 14 - anytime soon :)
  - Java EE 8 Certification
  - Improved package granularity
- WildFly 15, 16 - 1st Half 2019
  - Jakarta EE 8
- WildFly and JBoss EAP will support Java EE 8 / Jakarta EE 8

Open JDK : the safer and better choice



Oracle finally targets Java non-payers – six years after plucking Sun

Thought Java was 'free'? Think again (and you owe us $$$ in 2017)

By Gavin Clarke 16 Dec 2016 at 13:58 149 SHARE ▼

OpenJDK

https://www.theregister.co.uk/2016/12/16/oracle_targets_java_users_non_compliance/

OTHER QUESTIONS ?