Ready for vendor neutral Cloud Native Java!

July 2019 / Gaël Blondelle / @gblondelle
If by 50, you have not deployed a microservice architecture

You are a loser!
Conference Driven Development
Jump into Open Source Cloud
Native Java
Agenda

> Jakarta EE and MicroProfile: What’s new?
> Vendor neutral microservices
> With which tools?
> Future development
The Big Move

In September 2017, Java EE technologies moved to the Eclipse Foundation, where they now continue to evolve under the Jakarta EE brand.
An Industry Consortium

- Builds the community
- Approves Specifications
- Drives the Jakarta EE brand
- Establishes the technical roadmap
- Ensures compatibility
Jakarta EE Migration Process

Contribution Process

1. Proposal
2. Creation
3. Initial Contribution
4. Work in EF Repository
5. Release Review
Jakarta EE Migration Process

Contribution Process

1. Initial Contribution
2. Work in EF Repository
3. Release Review

DONE
Jakarta EE at the Eclipse Foundation

39
Projects

~160
New Committers

99
Git Repositories

53
Build Pipelines
TCK Now Open Sourced!

- Transparency
- Openness
- Shared Burden
- Vendor neutrality and continuity
Status Summary

> Code and TCKs contributed
  • Builds running on EF infra
> Eclipse GlassFish 5.1 Java EE 8 certified
  • Glassfish certification running on EF infra
> Eclipse Foundation Specification Process (EFSP) delivered: EFSP v1.1
> Jakarta EE SP (based on EFSP) delivered: JESP v1.0
## JCP vs Eclipse Foundation Specification Process (EFSP)

<table>
<thead>
<tr>
<th>Specification First</th>
<th>Code First</th>
</tr>
</thead>
<tbody>
<tr>
<td>Led by Specification Lead</td>
<td>Collaborative</td>
</tr>
<tr>
<td>Documents and TCKs are <strong>closed source</strong></td>
<td>Documents and TCKs are <strong>open source</strong></td>
</tr>
<tr>
<td>One normative “Reference Implementation”</td>
<td>One or more “Compatible Implementations”</td>
</tr>
<tr>
<td>Oracle certification process</td>
<td>Self certification</td>
</tr>
</tbody>
</table>
The Overall Specification Process

Just for a moment forget about the reviews

Proposal
Plan
Development
Milestone Build
Specification Version
Ratified Final Specification

JESP Jakarta EE Specification Process
&
EFSP Eclipse Foundation Specification Process
Status Summary

> New Specification Project Proposals
  • Jakarta NoSQL
  • Jakarta Batch
> Oracle agreement:
  • All future changes must happen in a new namespace
    • Jakarta vs. javax
  • Specs need to be renamed
What if you change ...

- javax.servlet
  Forces 9 package renames, 899 classes total
- javax.el
  Forces 12 package renames, 1115 classes total

See https://github.com/eclipse-ee4j/jakartaee-platform/blob/master/namespace/transitive.adoc
By David Blevins, Tomitribe
What's Next?

Eclipse GlassFish 5.1 ➔ Java EE 8

Eclipse GlassFish 5.2 ➔ Jakarta EE 8

Eclipse GlassFish 5.X ➔ Jakarta EE 9

GlassFish 5.0 ➔ Java EE 8
What’s Next?

Eclipse GlassFish 5.X ➔ Jakarta EE 9

Eclipse GlassFish 5.2 ➔ Jakarta EE 8

Eclipse GlassFish 5.1 ➔ Java EE 8

GlassFish 5.0 ➔ Java EE 8

Sources, TCKs, Docs

COPYRIGHT (C) 2019, ECLIPSE FOUNDATION, INC. | MADE AVAILABLE UNDER THE ECLIPSE PUBLIC LICENSE 2.0 (EPL-2.0)
What's Next?

- **Sources, TCKs, Docs**
- **Java EE 8**
- GlassFish 5.0 -> Java EE 8
- Eclipse GlassFish 5.1 -> Java EE 8
- Eclipse GlassFish 5.2 -> Jakarta EE 8
- Eclipse GlassFish 5.X -> Jakarta EE 9

- javax namespace
- New process
- New licensing
- New logo
“Jakarta EE 8 will NOT break existing Java EE 8 applications that rely upon javax APIs”

See
https://eclipse-foundation.blog/2019/05/08/jakarta-ee-8-faq/
What’s Next?

- New namespace
- Java SE 11
- Removal of deprecated specs
- ....maybe more
What’s Next?

- **Eclipse GlassFish 5.0** to **Java EE 8**
- **Eclipse GlassFish 5.1** to **Java EE 8**
- **Eclipse GlassFish 5.2** to **Jakarta EE 8**
- **Eclipse GlassFish 5.X** to **Jakarta EE 9**

Join the conversation at jakarta-platform-dev@eclipse.org
Community Engagement

- jakarta.ee-community@eclipse.org
- Jakarta Tech Talks
- Monthly Jakarta EE Update calls
- Monthly email updates
- Ad hoc town halls
- Blog aggregator on jakarta.ee
- Deep technical discussions on future of the platform at jakarta-platform-dev@eclipse.org

https://jakarta.ee/
Specifications for Java Microservices

Eclipse MicroProfile

- Collection of open source specifications for developing enterprise Java microservices
- Open source approach removes reliance on single vendor release cycles
- Natural extension of a Java EE developers’ existing skill set
Eclipse MicroProfile Specifications

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

MicroProfile 3.0

= New
= Updated
= No change from last release (MicroProfile 2.2)
Agenda

＞ Jakarta EE and MicroProfile: What’s new?
＞ Vendor neutral microservices
＞ With which tools?
＞ Future development
Open Source Foundations enable vendor-neutral open collaboration
Redis has a license to kill: Open-source database maker takes some code proprietary

It’s ‘cuz cloud giants use tools without giving anything back

By Thomas Claburn in San Francisco 23 Aug 2018 at 06:05

• Control by a single company
• Companies pivot!
• Need for a trusted third party

Database maker Redis Labs this week moved the Redis Modules developed by the company from the AGPL to a license that combines Apache v2.0 with Commons Clause, which restricts the sale of covered software.

The licensing change means that house-made Redis Modules – RediSearch, Redis Graph, ReJSON, ReBloom and Redis-ML – are no longer open-source software, as the term is generally defined. Instead, they become "source available."
There is only one definition of Free Software (since 1984)

**Freedom 0** - to run the program, for any purpose

**Freedom 1** - to study how the program works, and change it to make it do what you wish

**Freedom 2** - to redistribute copies

**Freedom 3** - to distribute copies of your *modified* versions to others
Foundations ensure sustainability by...

Securing copyright

Ensuring diversity

https://projects.eclipse.org/projects/eclipse
Foundations ensure sustainability by community diversity

Contribution Activity:
Commits on this project (last 12 months).

Individual Contribution Activity:
Commits on this project by individuals over the last three months.
Open Source principles are beneficial to the ecosystem!

Transparency

Openness

Meritocracy
Current MicroProfile implementations
Innovation happens!

Going Native with Eclipse MicroProfile and Quarkus

A primary value of Eclipse MicroProfile is the ability to write portable microservices that can run on multiple MicroProfile implementations. This offers developers a superb selection of runtimes to choose from and decide on one (or more) based on technical or business requirements. In case you haven’t heard, there is a new MicroProfile implementation in town called Quarkus!
Agenda

> Jakarta EE and MicroProfile: What’s new?
> Vendor neutral microservices
> With which tools?
> Future development
Command-line Interface

- Soon to be released
- CLI will be used for the creation of major IDE extensions
Eclipse Wild Web Developer
Eclipse CodeWind

- Tools for easier microservices development and test
- On the desktop: starting projects in Docker
- Deploying applications to Kubernetes
- Incremental update of applications running in Docker or Kubernetes
- Application performance testing tools
Eclipse CodeWind

VS Code

Eclipse IDE

Eclipse Che
Agenda

> Jakarta EE and MicroProfile: What’s new?
> Vendor neutral microservices
> With which tools?
> Future development
Jakarta EE 9
Community Wishlist

> CDI Alignment
> Modularity
> MicroProfile, Microservices
> Cloud-native deployments
> Reactive streams
> Java 11
> + ... your inputs ...
Eclipse MicroProfile roadmap

MicroProfile 3.1

- Open Tracing 1.3
- Open API 2.0
- Rest Client 1.3
- Config 1.4
- Fault Tolerance 2.0
- Metrics 2.0
- JWT Propagation 1.1
- Health Check 2.0
- CDI 2.0
- JSON-P 1.1
- JAX-RS 2.1
- JSON-B 1.0

- = New
- = Updated
- = No change from last release (MicroProfile 3.0)

Outside umbrella
- Reactive Messaging 1.0
- Reactive Streams Operators 1.1
- GraphQL 1.0
- LRA 1.0
- Context Propagation 1.0

Standalone
- Reactive Messaging 1.0
- Reactive Streams Operators 1.1
- GraphQL 1.0
- LRA 1.0
- Context Propagation 1.0
Going back to my Cloud Native Java Rpi based cluster
Cool techno for developers
Sustainable for companies
Really open source

Transparent  Open  Meritocratic
The Jakarta One Livestream CFP is open!
Participate and submit your talk by July 15: https://www.papercall.io/jakartaone-cfp
PARTICIPATE AT ECLIPSECON EUROPE 2019!

SUBMIT A TALK OR TUTORIAL

WWW.ECLIPSECON.ORG

@ECLIPSECON