JAKARTA EE

Update - January 2019

ECLIPSE FOUNDATION

Jakarta EE Working Group Members

Strategic Members

Participating Members

ECLIPSE FOUNDATION

JAKARTA EE

Working Group

An Industry Consortium

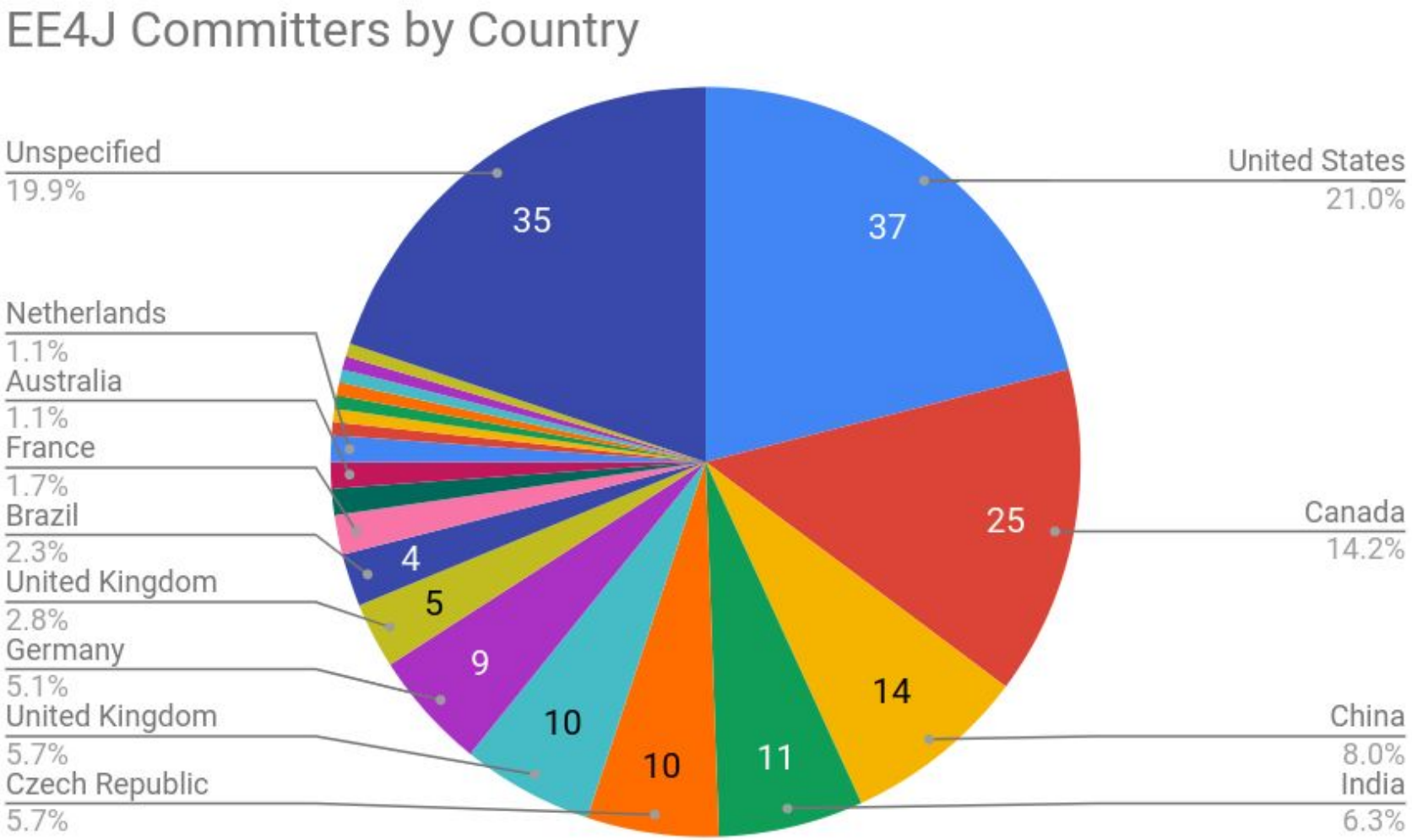
Approves Specifications

Manages the Jakarta EE brand

Establish technical roadmap

ECLIPSE FOUNDATION

An International Community



Accomplishments

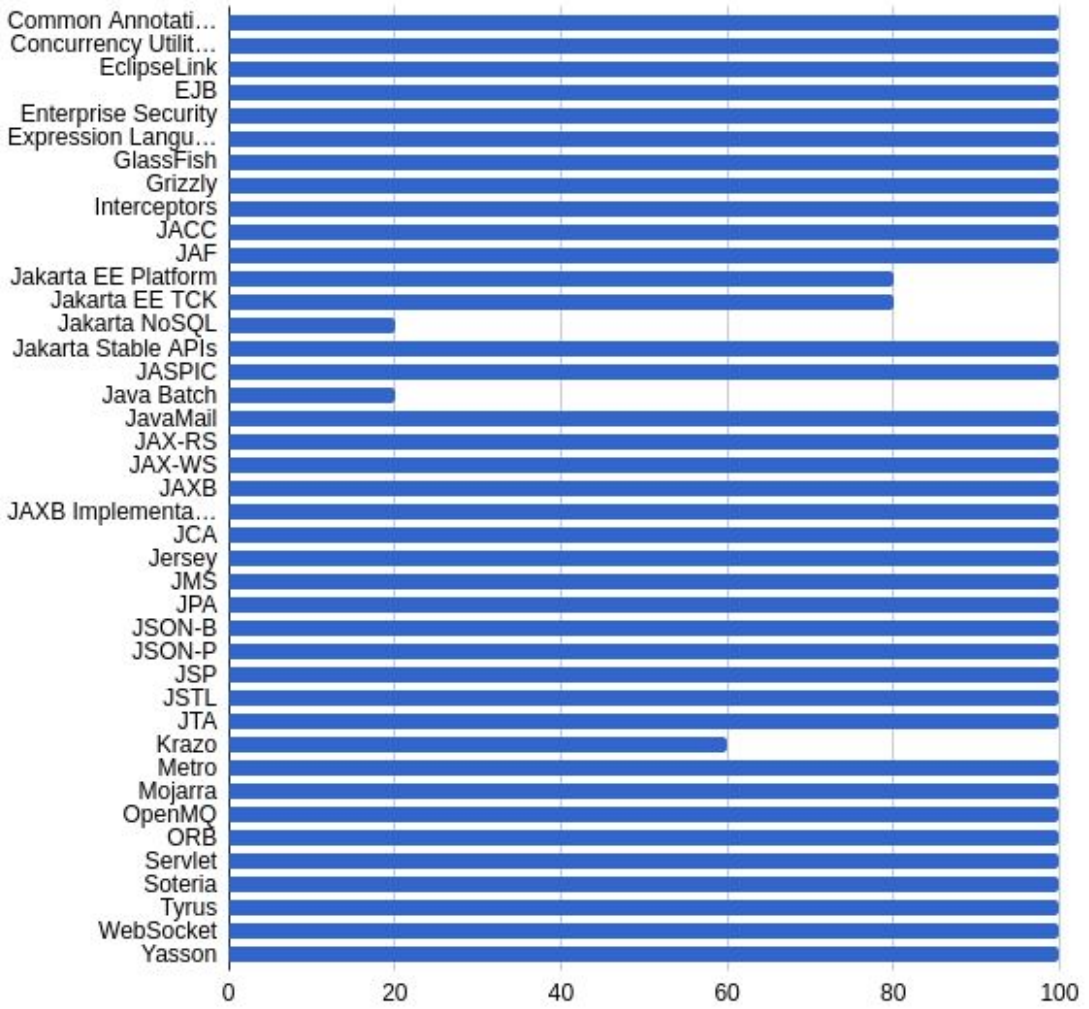
- 39 new Eclipse projects
  - Java EE 8 RIs, TCKs are all contributed
  - 160 new committers
  - IP team has already processed 30% more CQs than in all of 2017
  - Thank you Oracle! The Java EE team has been great to work with
- Java EE 8 TCK agreement completed with Oracle
- Eclipse Glassfish 5.1 was released on Jan 29, 2019 certified as Java EE 8 compatible

What does it mean?

TCK is now open sourced!

- **Transparency**
  - insight into tests
  - the community participation
- **Openness**
  - greater pool of contributors
  - equal opportunity with established process and governance
- **Shared burden**
  - spread responsibility for building and maintaining the TCKs
  - no dependency on a single organization or group
- **Vendor neutrality and continuity**
  - continuity in the case single entity reduces their investment.

Proposal → Creation → Initial Contribution → Working in an EF Repository → Release Review



Done

JAKARTA EE

Migration status

http://jakarta.ee/

#jakartaee

ECLIPSE FOUNDATION

JAKARTA EE

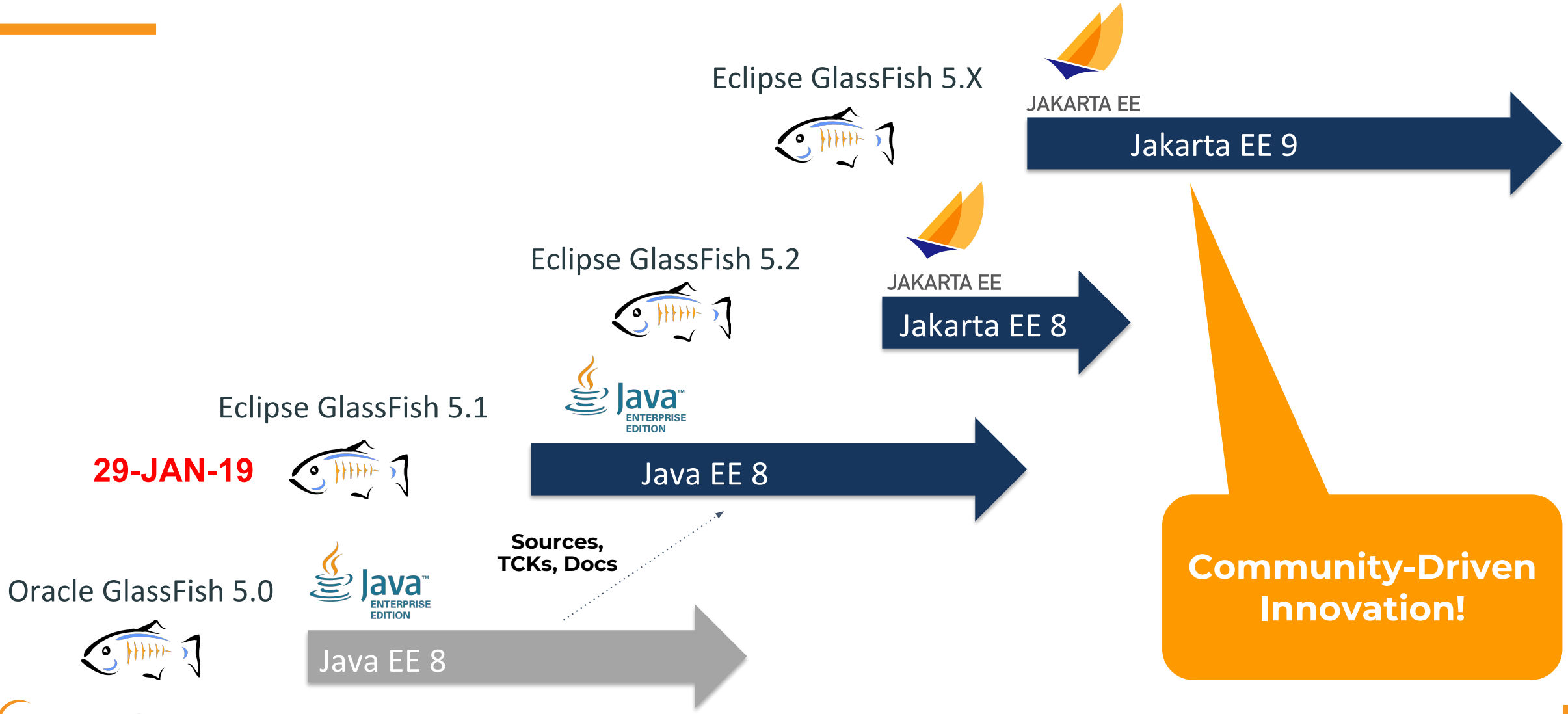
Open for Innovation

Technical Direction Guiding Principles

- Open for Innovation
- Split stand-alone Jakarta EE TCK into individual projects
- Embrace JPMS
- Standardise on the Maven build system
- Deprecate old technologies and provide optional modules
- Prefer soft dependencies
- Integration with CDI and Config
- Release Cadence
- Focus on testing
- Standard formatting of Specification and Documentation

ECLIPSE FOUNDATION

High Level Roadmap



ECLIPSE FOUNDATION

9

JAKARTA EE

So Where Are We?

- The code and TCKs are contributed
- We have the first Eclipse GlassFish 5.1 official release
- EF Specification Process adopted
- Jakarta EE Specification Process being defined as an adaptation of the EF process
- A proposal for the very first specification to follow the EF Specification Process
  - *Jakarta NoSQL*

ECLIPSE FOUNDATION

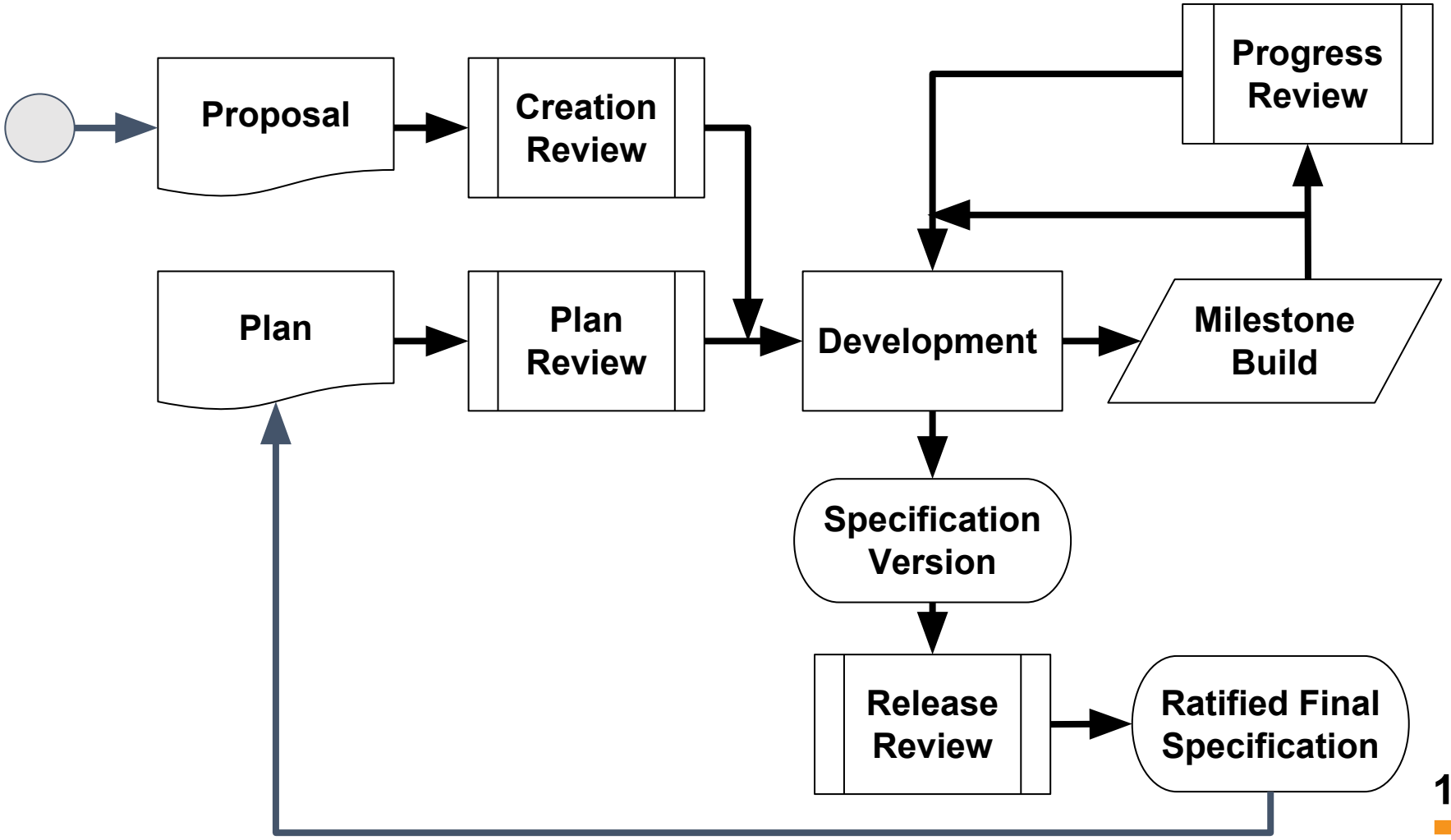
The Jakarta EE Specification Process

JCP vs. EFSP

Java EE	**Specification** First	→	**Code** First
Java EE	**Led** by Specification Lead	→	**Collaborative**
Java EE	Documents and TCKs are **closed source**	→	Documents and TCKs are **open source**
Java EE	One normative "**Reference Implementation**"	→	One or more "**Compatible Implementations**"
Java EE	**Oracle certification process**	→	**Self certification**

The Overall Specification Process

EFSP Community review completed



EFSP vs. JESP

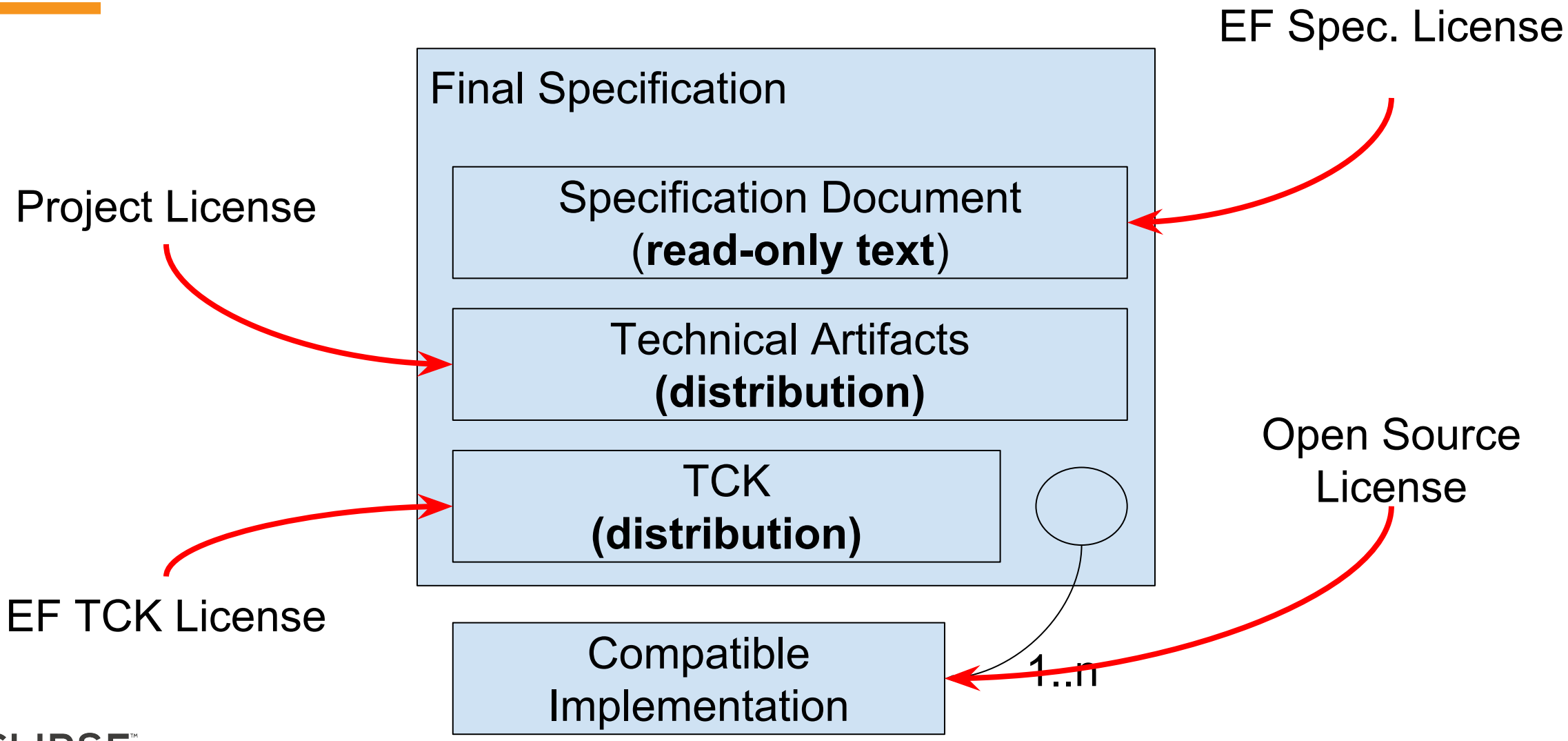
EFSP = “Eclipse Foundation Specification Process”

- Template to be used by any working group that wants to implement a spec process
- e.g. Sparkplug, Jakarta EE

JESP = “Jakarta EE Specification Process”

- Specialization of EFSP by the Jakarta EE Working Group
- Still under discussion
- Example change: specifying a longer review period for each vote

Final Specification



Specification Licenses

- Eclipse Foundation Specification License
  - Allows implementers to create implementations of the spec under whatever license they would like
- Eclipse Foundation TCK License
  - Allows implementers to verify that they are compatible with a specific version of a ratified final specification
- Eclipse Foundation Trademark License
  - Allows compatible implementation to use the logo/trademark (e.g. Jakarta EE)

Specification Licenses

Defines compatibility
$0

- Eclipse Foundation Specification License
  - Allows implementers to create implementations of the spec under whatever license they would like
- Eclipse Foundation TCK License
  - Allows implementers to verify that they are compatible with a specific version of a ratified final specification
- Eclipse Foundation Trademark License
  - Allows compatible implementation to use the logo/trademark (e.g. Jakarta EE)

Specification Licenses

- Eclipse Foundation Specification License
  - Allows implementers to create implementations of the spec under whatever license they would like
- Eclipse Foundation TCK License
  - Allows implementers to verify that they are compatible with a specific version of a ratified final specification
- Eclipse Foundation Trademark License
  - Allows compatible implementation to use the logo/trademark (e.g. Jakarta EE)

Permits use of logo
Must be a member of Jakarta EE, but no license fee

Self Certification

"Compatible Implementation"

- Implements a Final Specification
- Fulfills all of the requirements of the Ratified TCK
- Must publicly post TCK results

Brand

- Compatible Implementations of Profiles
- Eclipse Foundation Trademark License
- Must be an Eclipse Foundation Member
  - *…..but no license fees or royalties*

ECLIPSE FOUNDATION

Thank You!
Any Questions?

www.eclipse.org

ECLIPSE FOUNDATION