Introducing the Eclipse Foundation Specification Process

Agenda

- Background
- Creating the EFSP
- What is a Specification?
- Eclipse Foundation Specification Process
- EFSP and the JCP
- Certification

ECLIPSE™ FOUNDATION

Background

Why are we doing this?

- Opportunity meets necessity
- Java EE migration to Eclipse Foundation requires a spec process to replace the JCP
- We expect that this process will be used elsewhere

ECLIPSE™ FOUNDATION

What's the Big Deal?

Specifications

- Guides you to implement collectively developed idea
- Support multiple implementations
- Allow for interoperability

ECLIPSE™ FOUNDATION

Guiding Principles

- “Code First”
- No more “Spec Lead”
- Specifications run as open source projects
- “Compatible” implementations, rather than one “Reference” implementation
- Self-certification
- Branding for compatible implementations of Profiles

Jakarta EE Spec Process: 2018 Key deliverables

- Establish spec process for existing (JCP) and new specs
- Compatibility process
- Brand licensing

Creating the EFSP

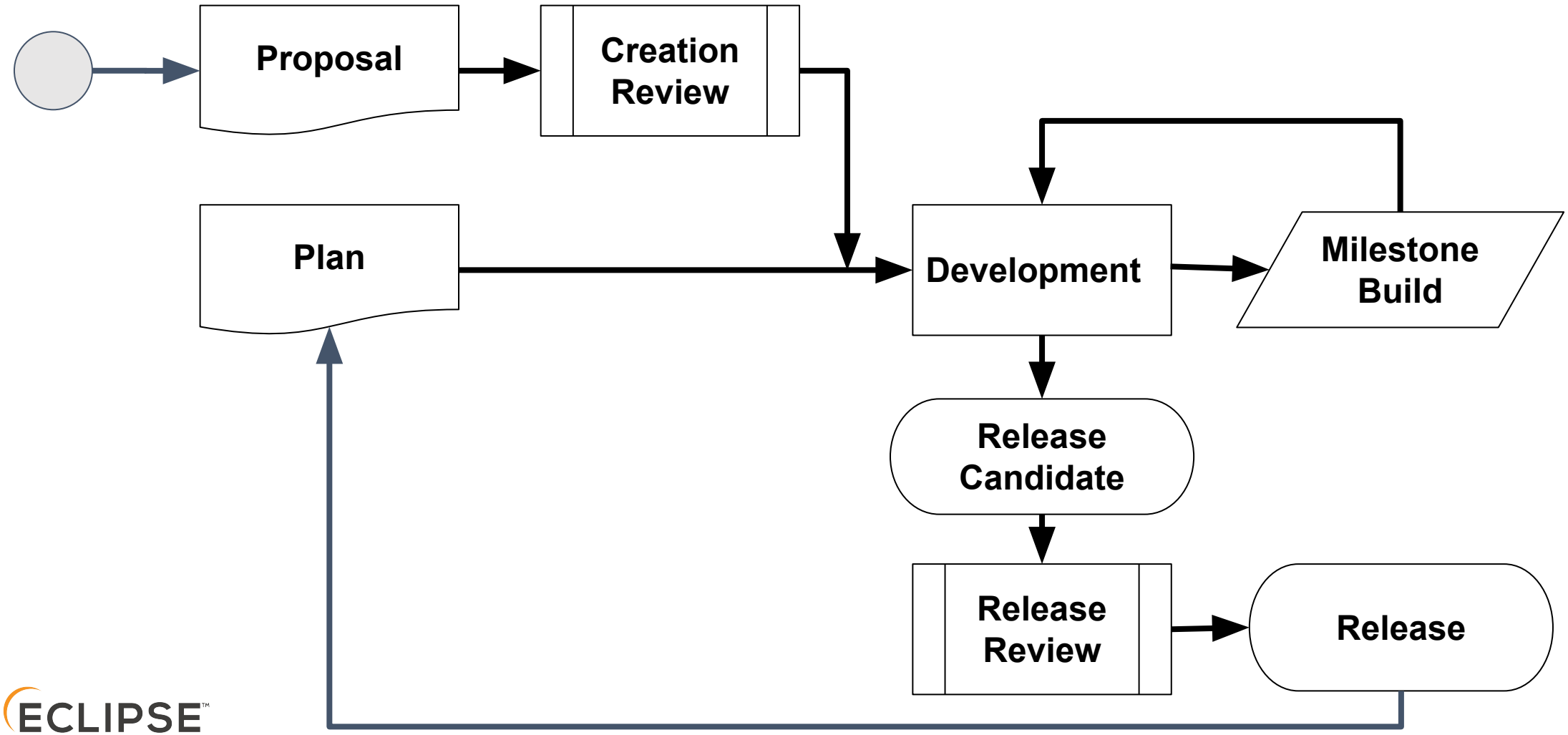
Start with The Eclipse Development Process

- Open source rules of engagement
- Governance, structure, definitions, reviews
- General framework for projects
- Day-by-day development rules/process is defined by the project

Open Source Rules of Engagement

- Transparency
- Openness
- Meritocracy
- Vendor neutrality

Development Process



EDP 2018

- “Progress Review”
  - Release Review becomes a kind of Progress Review
- Projects may release within one year of engaging in a successful Progress Review
  - IP Policy must be followed at all times
- Formalize the list of terms

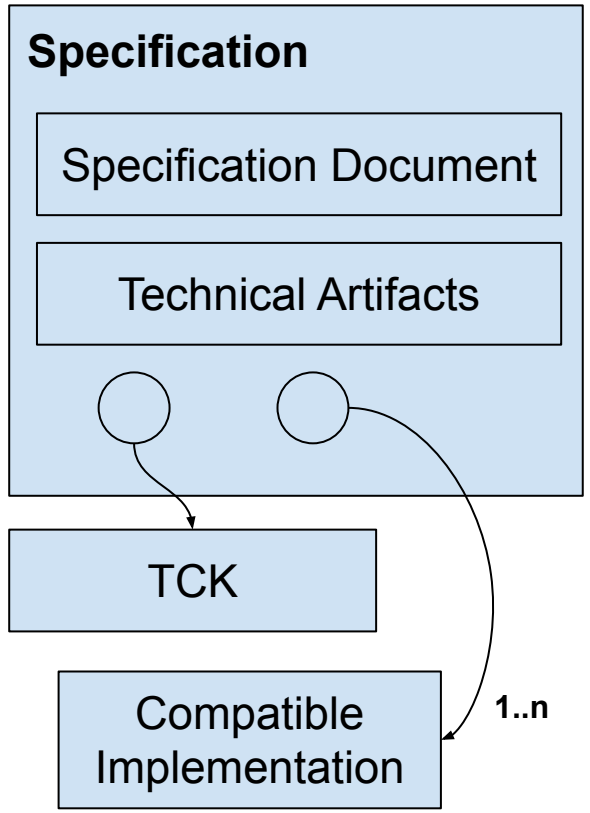
Eclipse Foundation Specification Process

- Spec. development in open source
  - Extends the EDP
  - “Just enough” process
- “Specification Project”
  - Formal alignment of Specification Projects with Working Groups
  - Specification Committee approval
- Participants and Participant Representatives

What is a Specification?

A Specification is...

...a collection of APIs, descriptions of semantic behavior, data formats, and/or protocols intended to enable the development of independent compatible implementations.



Specification Document

- Textual description of the obligations and rules
- May specify optional parts
  - Must be possible to implement all optional parts
- Must not override rules in referenced specifications

ECLIPSE FOUNDATION

TCK

- Specification must designate a TCK
  - May be different for different versions
- Must be developed under an approved Open Source License

ECLIPSE FOUNDATION

Compatible Implementations

- A Specification must reference at least one Compatible Implementation
- Must be developed under an approved Open Source License
- At least one Compatible Implementation must implement all optional features

ECLIPSE™ FOUNDATION

Open Source License

A “Compatible Implementation” must exist under an “Open Source License”, which is one of:

- Eclipse Public License - v 2.0 (+ Secondary Licenses)
- Eclipse Distribution License - v 1.0 (BSD-3-Clause)
- Apache License - v 2.0.

This list may be augmented by a Working Group with the approval of the EF Board

Profiles and Platforms

- A specification may be designated as a “Profile”
  - Profiles aggregate other specifications
- A Profile may be designated a “Platform”

ECLIPSE FOUNDATION

The Eclipse Foundation Specification Process

Eclipse Specification Process

- “Just enough” process
- Based on/extends the EDP
- IP flows are similar to the EDP process
- Patent rights need to be addressed
- Customizable

ECLIPSE™ FOUNDATION

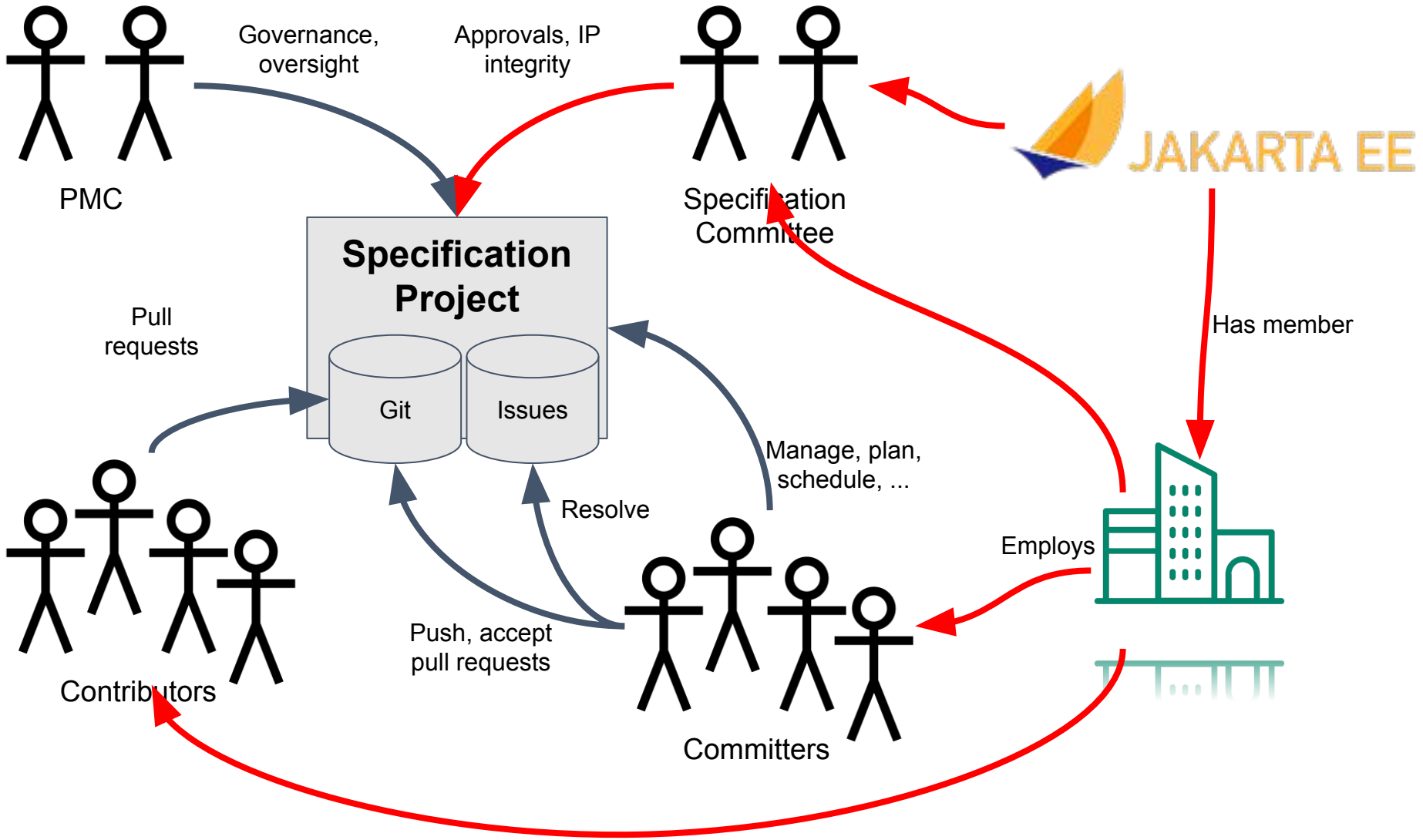
Specification Project

An Eclipse Project that…

- Is designated as a “Specification Project”
- Is “owned” by a Working Group
- Requires Specification Committee approvals
- Has a special class of committer

ECLIPSE™ FOUNDATION

Who's Who?



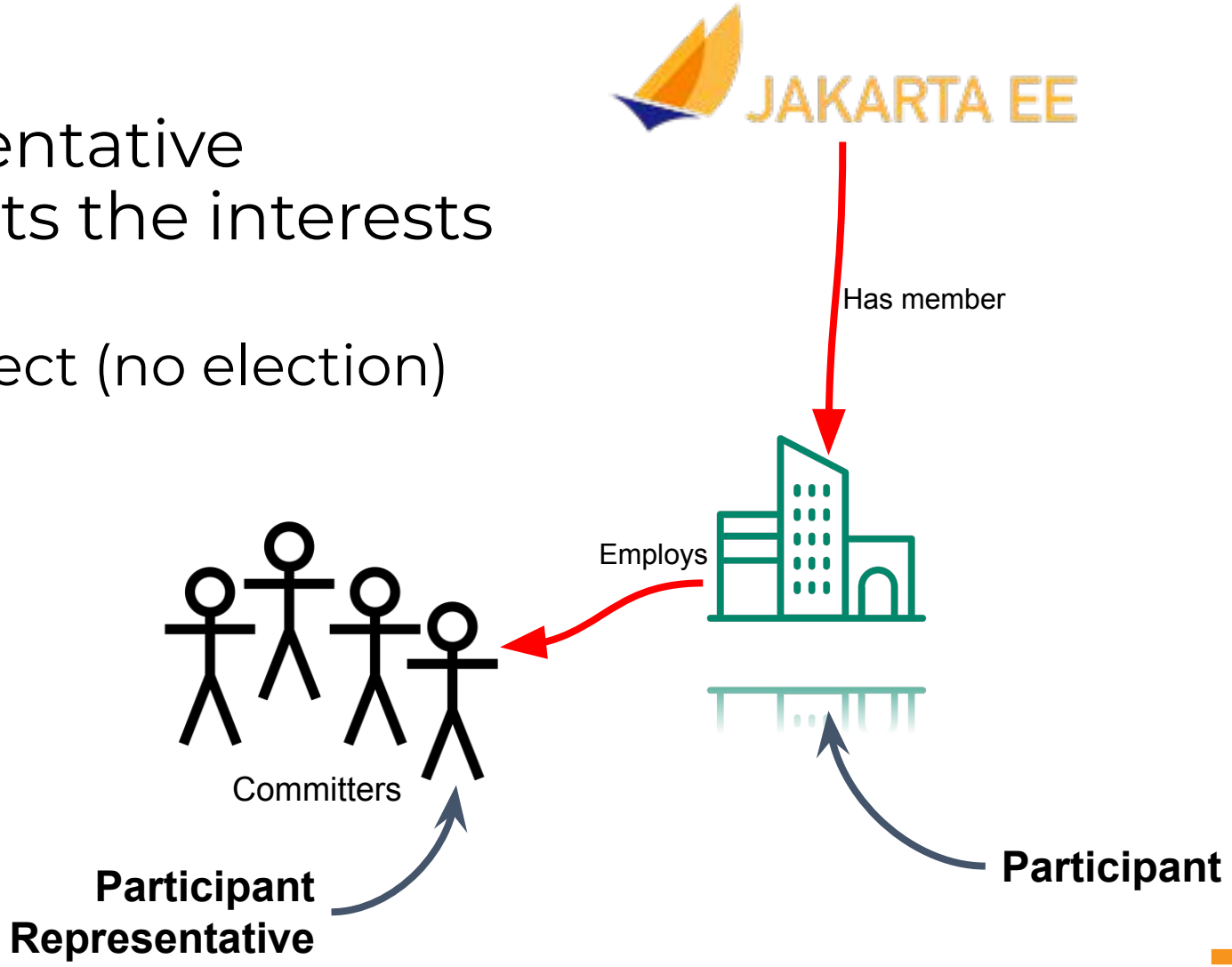
Specification Committer

- Must be a Member
  - Employed by an Eclipse Foundation Member Company, or
  - Committer Member
- Must be covered by a Working Group Participation Agreement

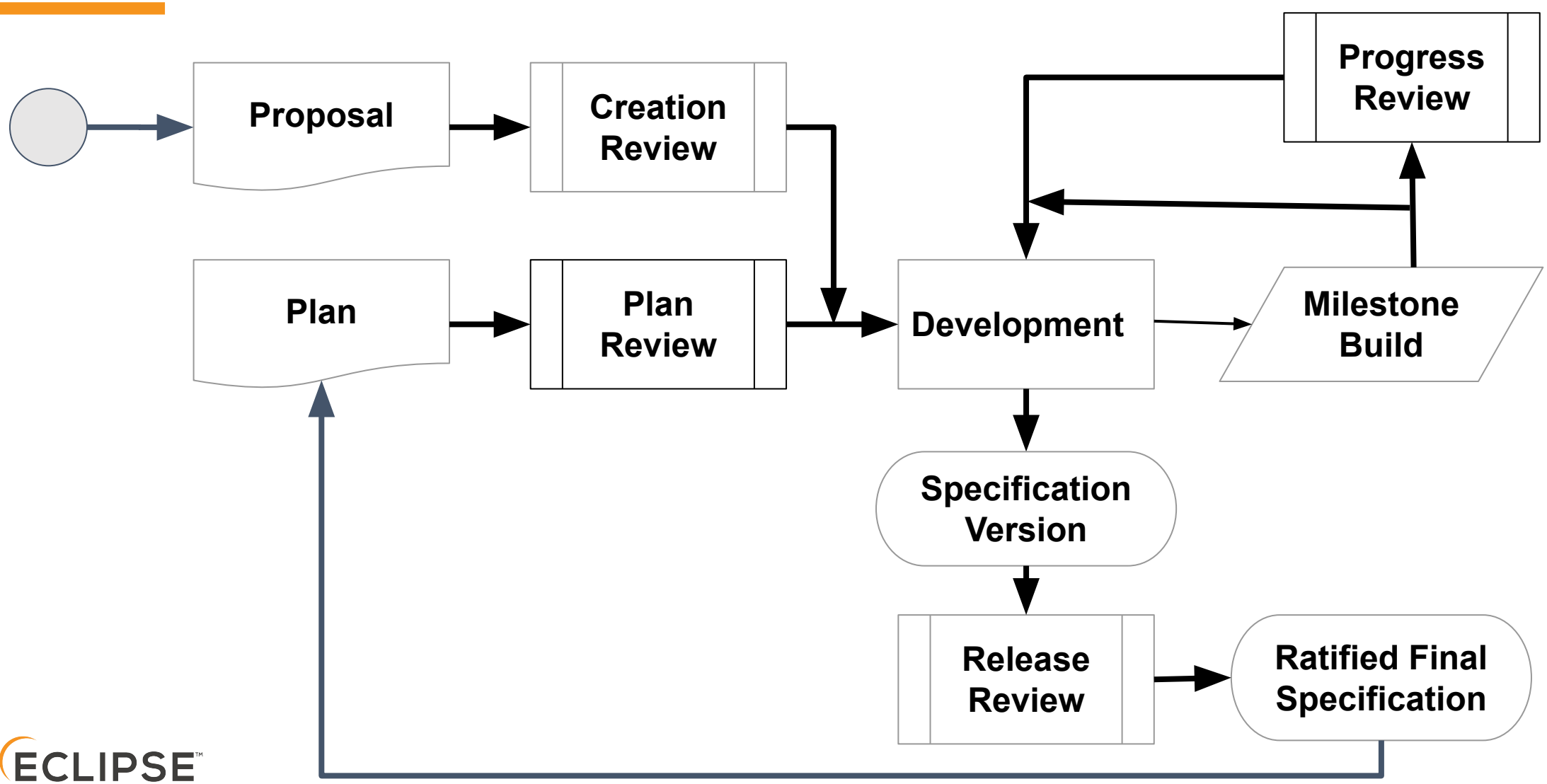
ECLIPSE™ FOUNDATION

Participant/Participant Representative

- A Participant Representative (committer) represents the interests of a Participant
  - Appointed to the project (no election)
- Participant
  - Individual Participant
  - Member Participant



Specification Process



Specification Committee Approvals

- Specification Project creation
- Release Plan
- Revision to the Scope
- Progress and Release Reviews
- Designation of a Profile or Platform
- Service Releases.

Revised Committer/Contributor Agreements

- Grant a license that says that if you contribute to an Eclipse project we can use your contributions to create a specification
  - Eclipse Contributor Agreement (ECA)
  - Individual Committer Agreement (ICA)
  - Member Committer Agreement (MCA)
  - Terms of Use
- State that if you post an idea on a Spec Project mailing list that you won't sue us later if we use your idea
  - Terms of Use

EFSP and JCP

EFSP compared to the JCP…

Code first	Specification First
Specification Committee	Executive Committee
Specification Project	Expert Group
IP Flows through Participants	IP Flows to and through Specification Lead
Collaborative	Led by Spec Lead
Public Communication	Public Communication
Customizable by Working Group	

… EFSP compared to the JCP…

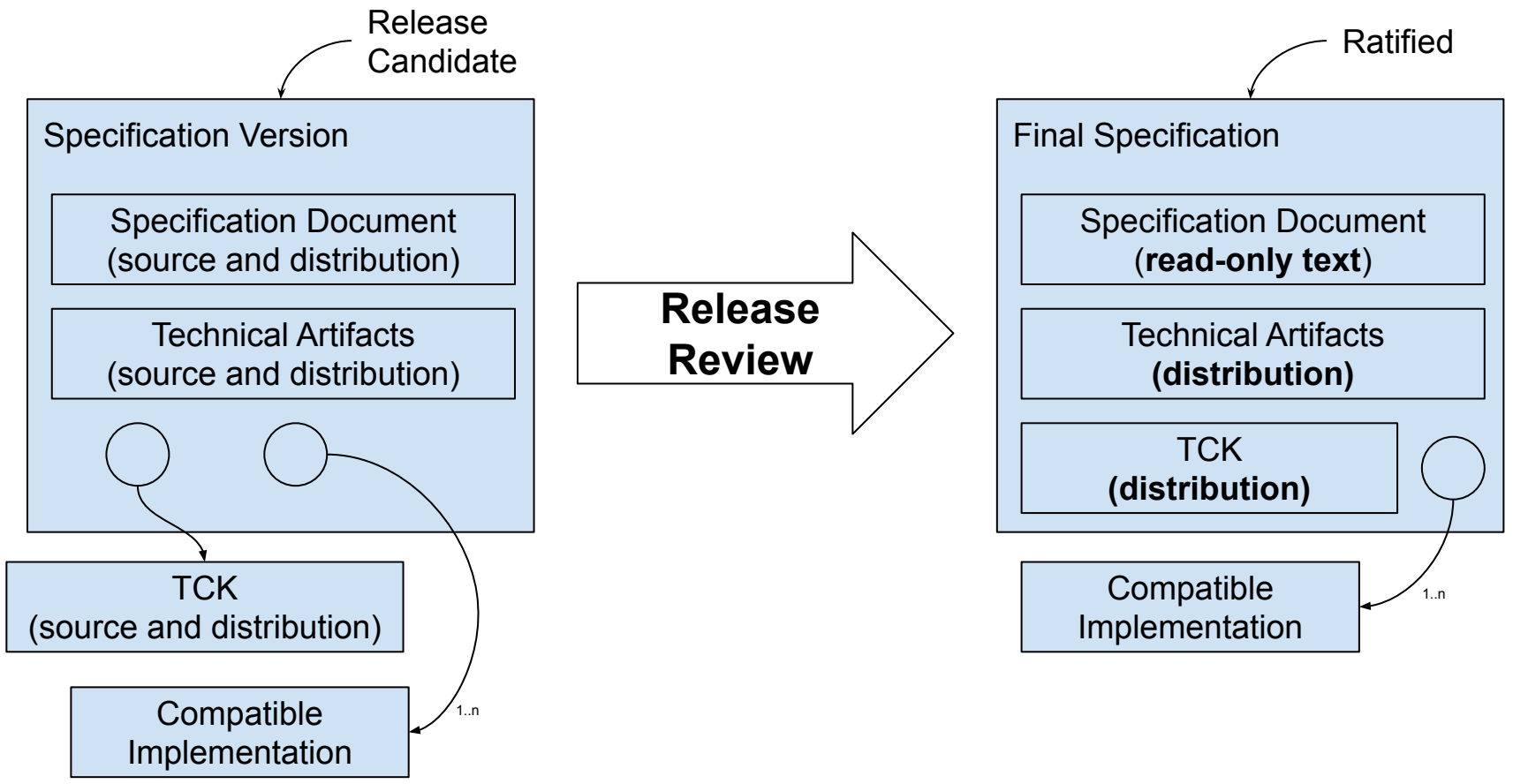
Document is open source	Document is closed source
Eclipse Foundation Specification License	Chosen by Spec Lead
TCK is open source	TCK is closed source
Eclipse Foundation TCK License	Proprietary License & NDA
One or more “Compatible Implementations”	One normative “Reference Implementation”

… EFSP compared to the JCP

Profiles	Profiles
Platforms	Platforms

Compatibility Claims

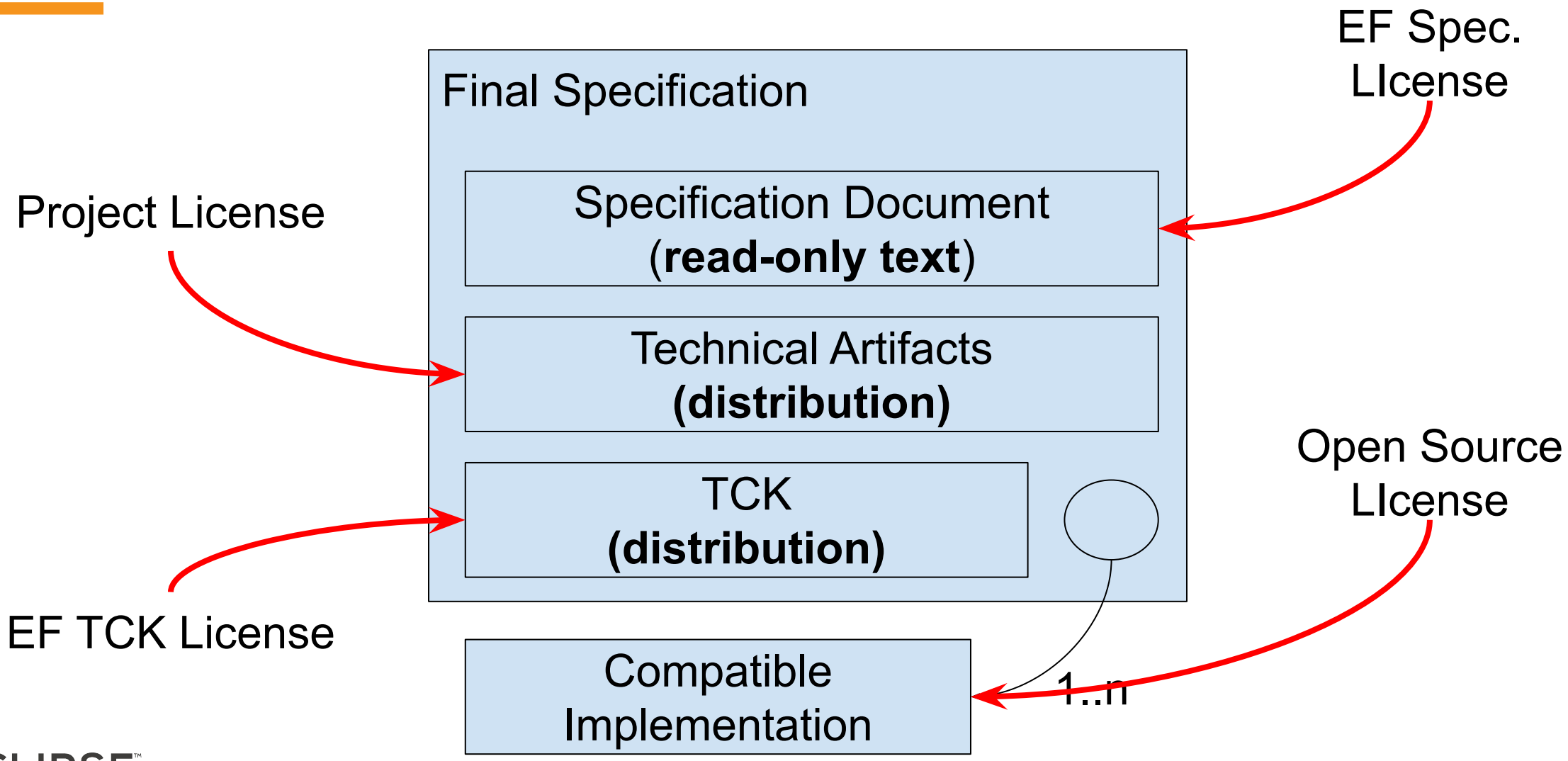
Final Specification



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)

Final Specification



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

The Take Away

The EFSP is...

- Code first
- Open and Transparent
- Community-oriented
- Light(er) weight
- Rigorous IP management
- Built on the success of the EDP

ECLIPSE™ FOUNDATION

Questions?