

Jakarta EE 8 Release

The purpose of this document is to capture the high level goals and success criteria for the Jakarta EE 8 release.

Assumption: Jakarta EE 8 == Java EE 8 ie. no additional specifications, API, behavioral changes or new profiles.

This release will consist of the following:

- **Specification** (documents and APIs) for the full platform including full and web profiles
- **TCKs** (source and binaries); part of the TCKs are documents on how to use TCKs
 - the Jakarta EE 8 will not attempt refactoring of the current monolith of the Java EE 8 TCK
 - Specifications that already have a separate TCK will remain as is
- **A Compatible Implementation of the full and web profiles**, at least one, available under open source license;

What is required for the release:

- **Jakarta EE Specification Process** needs to be completed
- **Jakarta EE TCK process** needs to be defined
- Means to publish Final Specifications and related TCKs
- Branding process and logo usage guidelines

With the above understanding the intent is to document and identify work required for Jakarta EE 8 release, and start on actions that can be carried out right away, while we are still waiting for legal agreements to be completed.

Specification: Overall Assumptions, Objectives and Projects

To declare success with Jakarta EE 8, we must have a Final Specification for the Full and Web Profiles, and for each of the referenced Specifications.

Assumptions

- No specification content will be refactored as part of this effort (e.g. TCK project will remain in its current form, and no APIs will change);
- A single Specification Project may host the work for multiple Specifications;
- All or most of the TCKs for the Specifications are hosted by a single TCK Project; and
- This applies only to specifications hosted by the Eclipse Foundation.

Objectives

- ✓ Create Eclipse Foundation Agreements and licenses
 - Eclipse Foundation Specification License / Eclipse Foundation TCK License
- Create a Final Specification for every Specification;
- Create a distribution channel for the Final Specifications and TCKs;
- Acquire required Paperwork for everyone involved in specification project work:
 - Eclipse Foundation Membership Agreement ([EFMA](#))
 - Member Committer and Contributor Agreement ([MCCA](#)) / Individual Committer Agreement ([ICA](#))
 - Working Group Participation Agreement ([WGPA](#))
- Develop a scheme to matching versions of Specifications with their TCK.

To create Final Specifications, we need to have a **Specification Project** and a **TCK Project** for every Specification hosted at the Eclipse Foundation.

Specification Projects

In order to create Final Specifications, all existing projects concerned with Specification work must be converted to Specification Projects (right now all the projects are created as Eclipse Projects, not specification projects).

Assumptions:

- Specification Projects will remain as EE4J subprojects;

Objectives (for each Specification Project):

- Finalize Specification name(s);
- Finalize corresponding project name;
- Acquire Specification Document(s);
- Update Specification Documents to take new Specification names into account;
- Acquire required paperwork for Specification Project Committers;
- Get Specification Committee Approval to convert existing project into a Specification Project; and
- Engage in a Release Review.

Details for the project **name**:

- We need to agree upon naming convention for Specifications and Specification Projects
 - Proposal on the Jakarta EE Specification names and Project names needs consensus by the Steering Committee;

- Trademark review of the naming proposal by Eclipse Foundation Trademark Team needs to be done

Details for the project **scope**:

- Per the EFSP, all Specification Projects must have a clear scope.
 - This is, ultimately, the responsibility of the individual Specification Teams;
 - Assistance of the PMC is needed and involvement all the project teams to ensure the project scope is well-defined and then follow the regular process for approval (EMO/ED approval, Specification Committee approval)

TCKs and TCK Project

Working on the TCK project for the Jakarta EE 8 release will mean:

- TCK project will stay as is (e.g. “Eclipse Jakarta EE TCK”)
 - For TCK project name we are currently considering dropping “Eclipse” to become “Jakarta EE TCK”
 - No changes are expected on this project for Jakarta EE 8 release
- Need to assess implication of specification renaming to TCK
- GlassFish is included in the CTS to enable interoperability testing, and at this point integral part of TCK / CTS. For the Jakarta EE 8 release, we can not run CTS without releasing Eclipse GlassFish 5.2
- Need to discuss making available TCK binaries and their distribution for consumers

Compatible Implementation

Compatible Implementation will implement the full and web profiles

- At least one compatible implementation needs to be available under open source license
- Community set up to run Jakarta TCK

Supporting Processes

- **Jakarta EE Specification Process** needs to be completed
 - IP Bootstrapping
- **Jakarta EE TCK process** needs to be defined
 - This is being worked on by Specification Committee
- Means to publish Final Specifications and related TCKs
- Branding process and logo usage guidelines
 - This is currently being worked on by Eclipse Foundationwe are not

Dependency Hierarchy - Summary

Think of the sub bullets as blockers on completing the task.

- Specification
 - Distribution Channel (for Final Specification and TCK)
 - [Eclipse Foundation Specification License](#)□
 - [Eclipse Foundation TCK License](#)□
 - TCK Project
 - Binary build and distribution
 - TCK process
 - TCK Project name
 - Updates to content based on changed spec names
 - Specification Project(s)
 - Spec name/new project name
 - Brainstorming new names for specs and projects
 - Oracle Agreement
 - IP (essential claims) Bootstrapping
 - Specification Documents
 - Oracle Agreement
 - Other vendor agreements
 - Scope
 - Specification Committee Approval
 - Committers
 - Signed Working Group Participation Agreement(s)
 - Signed MCCA(s) (for employees of Member companies)
 - Signed ICA(s) and Eclipse Membership Agreement (for individuals)
 - Specification Committee Approval of Release Review
 - JESP