

MicroProfile (Config) Future Directions

Jan Westerkamp 2026-02-05

History

- Java EE (8) development stuck
- MicroProfile was created at EF
- Java EE was donated to EF by Oracle
- Java EE development continued as Jakarta EE
- CN4J was created to align Jakarta EE and MicroProfile
- MicroProfile Config transition was planned within CN4J
- Jakarta Config (MVP) was created
- CN4J was declared as deprecated
- Config required for Jakarta NoSQL
- Discussions to move the MicroProfile WG into Jakarta EE WG started
- IBM proposal to move MPWG including keeping the org.eclipse.microprofile namespace was rejected by Jakarta Spec Committee

Reasons for new Approach

- A members majority would like to align the namespace within Jakarta EE
- A members majority still would like to merge MPWG into JEEWG
- To add Jakarta NoSQL into the Platform/Profile, MP Config (alike) is still required
- A Config spec within Jakarta Core Profile would be beneficial to almost all Jakarta Component Specs that require configuration

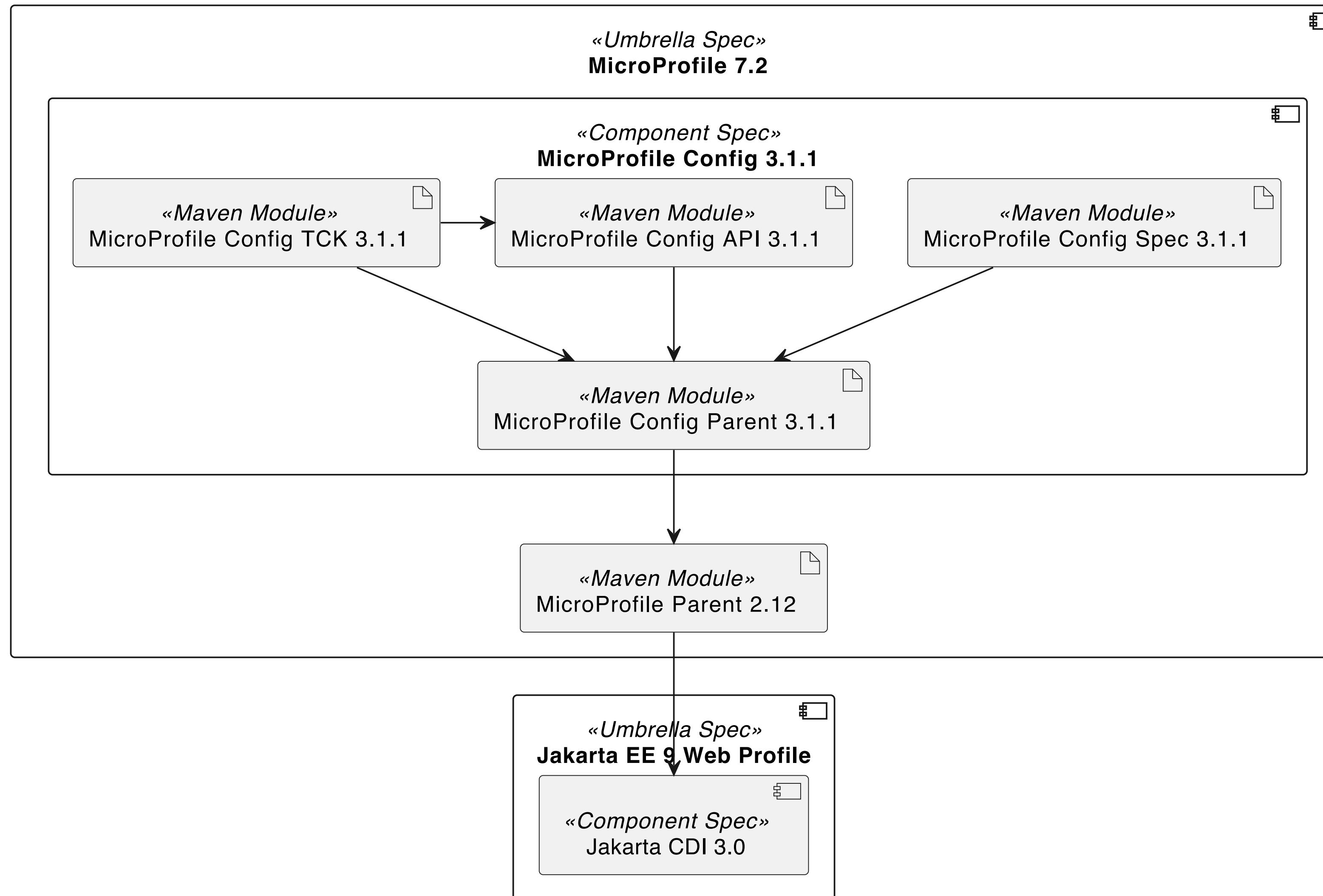
2 Ways

- Migrate MPWG into JEEWG as separate (Jakarta Micro) Profile
 - Namespace might change
 - Pattern like Java EE to Jakarta EE transition
 - Migrate Jakarta Config into Jakarta Core Profile
- Start with migration MP Config into Jakarta Config (only)
 - Namespace will change
 - Profile will change
 - New Release of MP Config depending on Jakarta Config

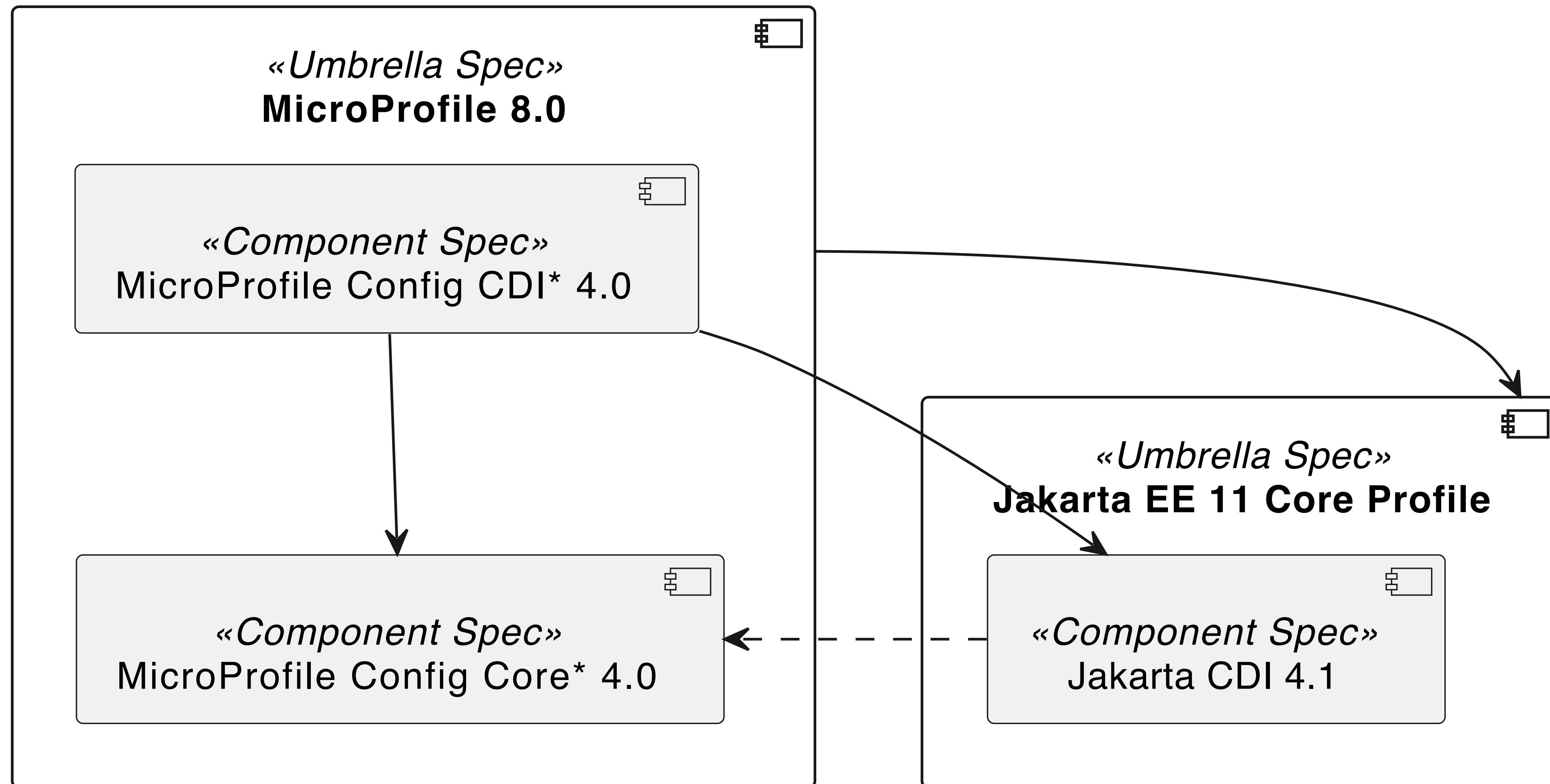
Technical Challenges

- MicroProfile Parent equivalent is missing in Jakarta
 - Single Git repo, multi maven module Spec setup required
 - Current MP Config dependency to MP Parent is outdated
 - MP Config 3.1 -> MP Parent -> 2.8 (-> Jakarta CDI 3.0.0 & Java SE 8)
 - MP Config 3.1.1-RC1 -> MP Parent -> 2.12 (-> Jakarta CDI 3.0.0 & Java SE 8)
- MP Config dependency to Jakarta CDI prevents general configuration (i.e. to CDI)
 - Split up MP Config into CDI dependent (Jakarta Config CDI*) and independent (Jakarta Config Core*) part
 - Decide in which WG the split will be done
- Design migration path for implementers and users
 - Following proven Java EE to Jakarta EE pattern
 - Define the Jakarta EE version to start with
 - May create new MP Config release depending on Jakarta Config for transition
 - Alternatively support both namespaces for transition

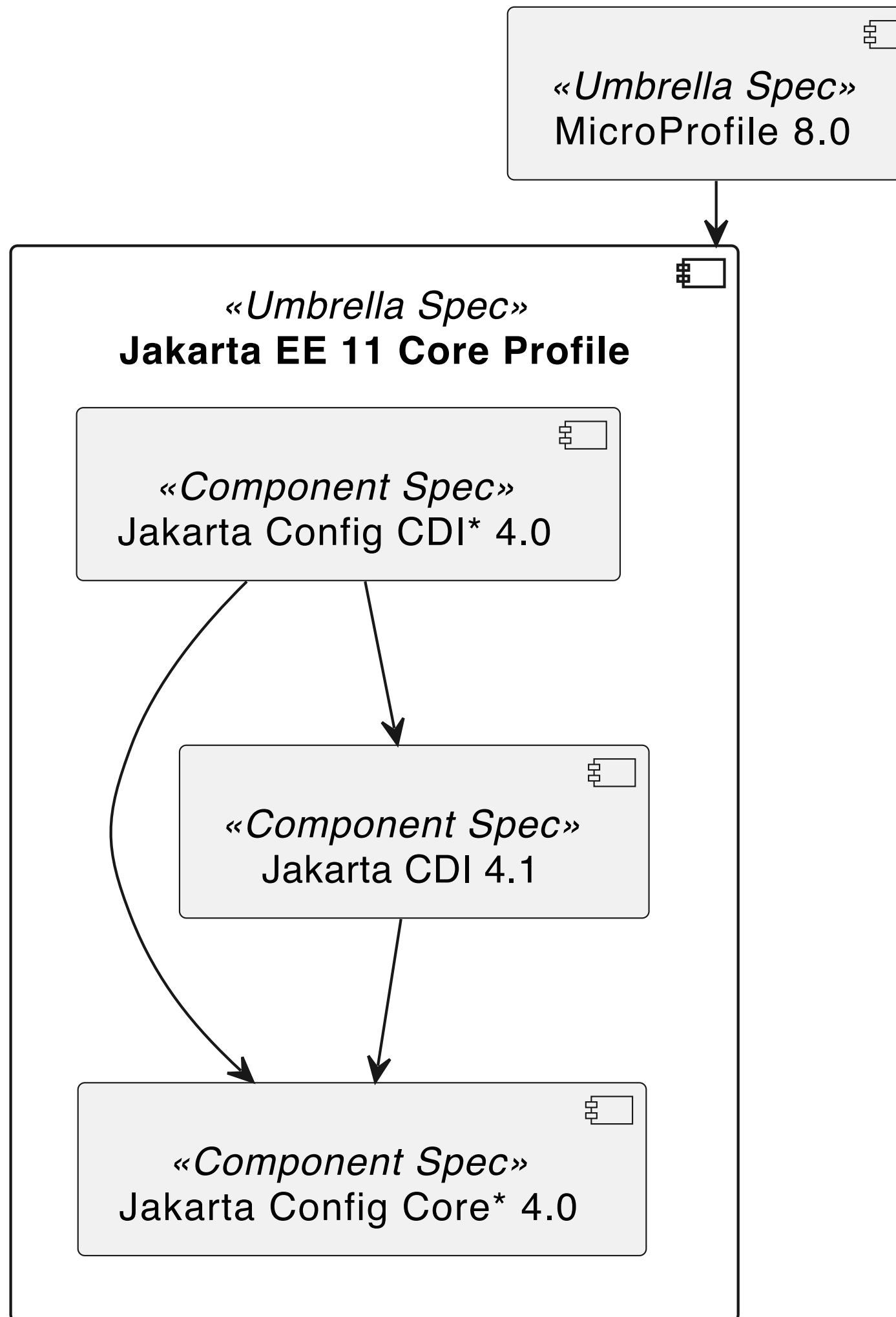
MP Config Spec



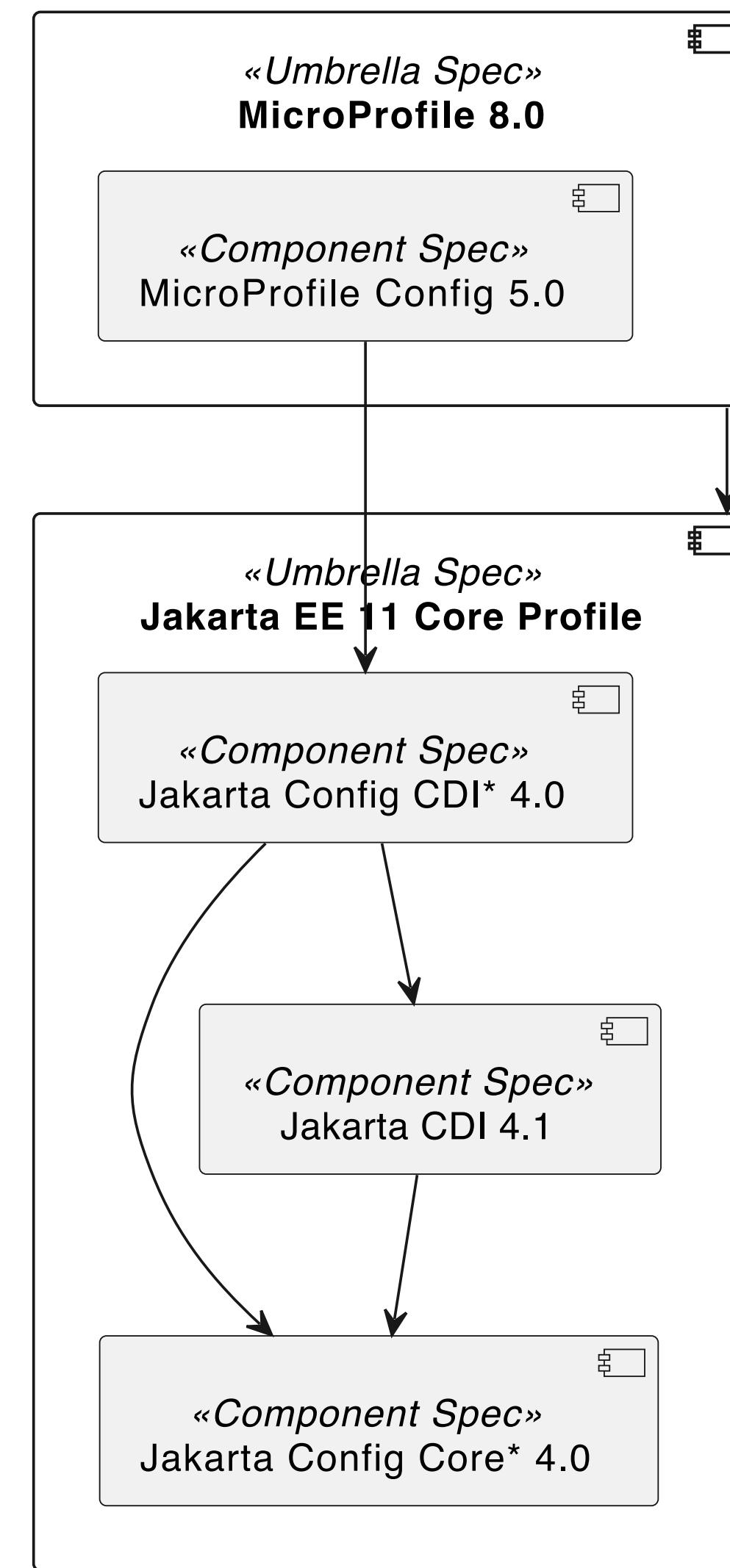
MP Config Split



Jakarta Config Split



Jakarta Config Split & MP Config



Organisational Challenges

- Majorities in both WGs are required
- Compatible Implementation (CI) in both WGs are required
- Jakarta Spec Committee constraint to use deviating namespace for API (jakarta.*) and TCK (!)
 - Exception for MP specs?
- Prepare MP (Config) major release depending on latest Jakarta EE release (JEE 11) for transition in MPWG?
- Meeting transparency and openness
 - Recorded meetings
 - Guest access
- Community involvement
 - Additional elected Community Member in committees?