Managing Unmaintained Component Dependencies in ONAP

THE LINUX FOUNDATION

LF NETWORKING

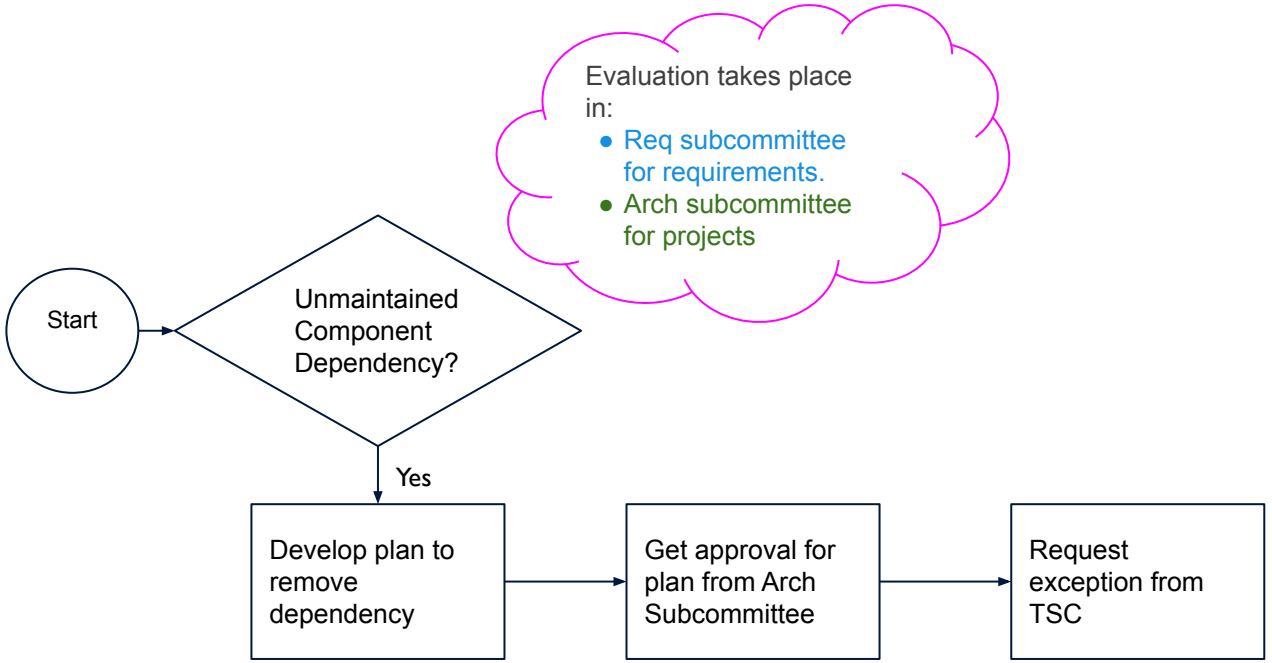
Introduction

- Dependencies on unmaintained components represents a security and logistical risk for ONAP
- Although we have developed processes for lifecycle management of unmaintained projects, we haven't really dealt with the issue of ongoing dependencies
- This week, a group of interested ONAP members began meeting to discuss how to handle these dependencies

Role of Requirements and Arch Subcommittees

- One possibility is to ask the requirements and arch subcommittees to update their review processes to include evaluating dependencies
- If a project or proposed requirement has a dependency on an unmaintained component, then in order to continue:
  - The PTL or requirement owner must develop a plan and get approval from the arch subcommittee to remove the dependency
  - The PTL or requirement owner must request an exception from the TSC

Flow


Evaluation takes place in:
Req subcommittee for requirements.
Arch subcommittee for projects
Start
Unmaintained Component Dependency?
Yes
Develop plan to remove dependency
Get approval for plan from Arch Subcommittee
Request exception from TSC

Benefits

- Makes issue more transparent to the TSC
- Requires the development of a credible plan, including a review process, for removing the dependency
- Creates “friction” for PTLs or requirement owners with dependencies on unmaintained components and incentivizes them to remove the dependency

Issues

- Plan depends on self-identification of dependencies. A tool would be preferable.
- What about legacy use cases that no longer have an "owner" but have dependencies on unmaintained components?
- The arch subcommittee has struggled to complete reviews by Milestone 2. Can they handle this additional workload?

Interested?

- Please join us on Mondays at 8 a.m. Pacific.
- Let me know and I will add you to the invitation.