

Open Source and Standards

Similar
but
Different

Scott K. Peterson
Red Hat, Inc.
speterso@redhat.com

ONF Connect 2019



Open Source and Standards: Similar

- Both: collaboration on development of technology.
- Overlap: fostering software interoperability
 - **Complementary:** development of open source implementations can inform development of specifications and/or can become reference implementations.
 - **Alternatives:** each can be used to facilitate software interoperability.
 - **Other roles:** Test suites and other tools that facilitate use of a standard.

Open Source and Standards: Different

- Not arbitrary points on a spectrum of combinations of features.
- Rather than seeking a hybrid, let's understand how to advantageously use each.
- The consequences of the differences are not obvious.
 - Implementations v. Specifications
 - Continuous development v. Version-driven development
 - Potential for forking has implications for governance

[Standards and Open Source: Why are patents treated differently?](#)

[Governance without rules: How the potential for forking helps projects](#)

[links to opensource.com]

Why is Open Source Important?

- Over decades, demonstrated to be a remarkably effective way to build software.
- Features work together to achieve remarkably powerful collaborative results:
 - **Ongoing work product (software) is visible to everyone**
 - **Broad openness of participation**
 - **Simple IP rules: contribution-based; no-strings**
 - You choose what to contribute.
 - Contributions are without strings attached - without expectation of subsequent fees for use of what you have contributed.
- "I just want to change it a little bit." → Don't expect the benefits of open source development.
 - more control over access to the ongoing work product ??
 - more formality for participation ??
 - adjust the patent rules ??