

ODL Root Parent: Beryllium: Release Review

Contents

- [Project Name](#)
- [Features](#)
- [Non-Code Aspects \(user docs, examples, tutorials, articles\)](#)
- [Architectural Issues](#)
- [Security Considerations](#)
- [Quality Assurance \(test coverage, etc\)](#)
- [End-of-life \(API/Features EOled in Release\)](#)
- [Bugzilla \(summary of bug situation\)](#)
- [Standards \(summary of standard compliance\)](#)
- [Schedule \(initial schedule and changes over the release cycle\)](#)

Project Name

odlparent

Features

ODL Root Parent provides three features in Beryllium release, which are repackaging of widely-used third party dependencies:

- odl-guava, which packages Google Guava libraries version 18.0
- odl-lmax, which packages LMAX Disruptor version 3.3.2
- odl-netty, which packages Netty version 4.0.33

Non-Code Aspects (user docs, examples, tutorials, articles)

N/A

Architectural Issues

None

Security Considerations

ODL Root Parent provides mostly build-related infrastructure and a centralized place for managing versions of third-party dependencies. While it does not introduce security risks itself, third-party artifacts may introduce security considerations, which need to be analyzed and taken into account.

Quality Assurance (test coverage, etc)

ODL Root Parent has a small Java component, which is modified Karaf plugin. This shows up in [Sonar](#). This component is exercised by odlparent-provided features and does not have unit tests. Reported Sonar issues are coding-style-related and will be addressed in the next release.

End-of-life (API/Features EOled in Release)

This is the last release declaring artifacts from eclipseSDK381. These components are outdated and at least some of them are not Java-8 compatible. Furthermore many of these are not used within OpenDaylight. Unused artifacts will be removed and the remaining ones will be replaced by their org. eclipse.tycho equivalents.

Bugzilla (summary of bug situation)

ODL Root Parent has currently [5 outstanding issues](#). None of these prevent the functionality provided from working. One of those deals with SingleFeatureTest coherence with user experience. Two track improvements to packaging collateral (CONTRIBUTING and similar). One deals with JaCoCo/Powermock integration. One tracks an ugly warning we get for packaging features.

Standards (summary of standard compliance)

Not applicable.

Schedule (initial schedule and changes over the release cycle)

We have started this release cycle with an empty release plan and all improvements have been contributed outside of planning. This means the project was always on schedule. While this has not lead to any wide-spread breakage, as the number of our downstreams increases and becomes more diversified, we will need to plan our major features and third-party upgrades in a structured way.