ODL Root Parent: Lithium: Release Plan

Contents

Introduction

Release Deliverables

Name	Description
Deliverable Name	Deliverable Description

Release Milestones

- **Offset:** <Should be 0, 1 or 2. The dates in the table and the number in the column header should be adjusted to match>

Milestone	Offset 0 Date	Deliverables		
M1	2014-12-11	**Name**	**Description**	
		Release Plan	Candidate Release Plan	
		Deliverable Name	Deliverable Description	
M2	2015-01-22	**Name**	**Description**	
		Release Plan	Final Release Plan	
		Deliverable Name	Deliverable Description	
M3	2015-02-19	**Name**	**Description**	
		Feature Freeze		
		Deliverable Name	Deliverable Description	
		BUG-2680	netty version to 4.0.26	
		Xtend	Completely remove references to xtend	
		pluginRepositories	Add pluginRepositories section to help resolving Maven plugins	
		BUG-2159	bump karaf version used in tests	
		pax-exam	Bump pax-exam to 4.4.0	

M4	2015-03-19		
		Name	**Description**
		API Freeze	
		Deliverable Name	Deliverable Description
			Bug fixes and enhancements

M5	2015-04-16		
		Name	**Description**
		Code Freeze	
		Deliverable Name	Deliverable Description

RC0	2015-05-28		
		Name	**Description**
		Deliverable Name	Deliverable Description

RC1	2015-06-11		
		Name	**Description**
		Deliverable Name	Deliverable Description

RC2	2015-06-18		
		Name	**Description**
		Release Review	Release Review Description
		Deliverable Name	Deliverable Description

Formal Release	2015-06-25		
		Name	**Description**
		Deliverable Name	Deliverable Description

Externally Consumable APIs

- Please list and describe all externally consumable APIs or point to a document that does so.
- This need not be a list of all java interfaces and yang files, but should be a list of high-level functionality, e.g., flow programming.
  - For each such API, there should be a list of the bundles providing the API, but not necessarily the implementation.
- Each PROVISIONAL or TENTATIVE API must be listed as project deliverable
  - Must be declared as "IN" or "OUT" at the M3 Milestone

Expected Dependencies on Other Projects

Currently not known dependent project.

Providing Project	Deliverable Name	Needed By	Acknowledged?	Description
Project Name	Deliverable Name	Milestone Needed By	Yes/No/Tentative (as link to Other Project Release Plan)	Description of the new deliverable or changes to a deliverable

Expected Incompatibilities with Other Projects

Currently not known any incompatible project.

Compatibility with Previous Releases

Removed APIs and/or Functionality

- Please include list of a APIs and/or functionality that existed the previous release and will be removed.
  - For each such API/functionality, discuss any suggestions for how those who are using it should adapt.
  - In order for an API/functionality to be removed, it must have been deprecated in a previous release.
    - Functionality and/or APIs that were correctly tagged with the @Deprecated annotation in Java before the Helium release can be considered deprecated and thus removed in Lithium.

Deprecated APIs and/or Functionality

- Please include list of a APIs and/or functionality that existed the previous release and will be deprecated.
  - For each such API/functionality, discuss any suggestions for how those who are using it should adapt.
  - If possible, e.g., for all human-generated Java, add the @Deprecated annotation
  - If not possible, please note clearly things which are deprecated in a clearly visible manner

Changed APIs and/or Functionality

- Please include list of a APIs and/or functionality that existed the previous release and will be changed.
  - For each such API/functionality, provide guidance about who will be affected and how they should adapt.
  - In general, project's should strive to be backward compatible with the previous release and note what functionality will be removed by deprecating it and noting that with the @Deprecated annotations wherever possible.

Themes and Priorities

Requests from Other Projects

For each API request, the requesting project should create an entry like the example below. After creating the entry, the requesting project should send an e-mail to release@lists.opendaylight.org, and both projects' dev lists using this template:

```
Subject: [REQUEST FOR NEW OR EXTENDED API] ${API name}

Note: This email is a request from ${requesting project} for a new or
extended API in ${providing project}.

API Name: ${API name}
Request: ${link to the request in the providing project's release plan}

Please let us know if you will be able to provide this new
functionality by the listed milestone. If you need clarifications or
help in providing the API, let us know so we can reach an agreement.

If you feel that providing this API is a bad idea regardless of where
the resources are coming from, please let us know why and ideally,
suggest and alternative.
```

Requesting Project	API Name	Needed By	Acknowledged?	Description

Example Request

- **Requesting Project:**
- **Providing Project:** <should be this project, i.e., the project whose release plan this page is>
- **Requested Deliverable Name:**
- **Needed Milestone:** <e.g., M3, assumed to be at that milestone at the offset of the providing project unless otherwise stated>
- **Requested Deliverable Description:** <description of the change or new API needed>
- **Response:** <one of Yes, No, Tentative>
  - **Description:** <provides details of the response, e.g., "Yes, we can do that", "No, that's a bad idea", "No, we don't have the staff", "Tentaive, it's a good idea, but we don't know if we can">
  - **Resources From:** <should be either the providing project if they agree to do the work or the requesting project if the agree>
  - **Link to Section in Requesting Project Release Plan:** <if No or Tentative, link to the adjustments in the requesting project's release plan>
  - **Link to Section in Providing Project Release Plan:** <if Yes or Tentative, link to the deliverable in this release plan>
- **Negotiation:**
  - <this should be a back and fort until a response of Yes, No, or Tentative is reached>
  - <it should probably start with a response from the providing project that can be one of Yes, No, or Tentative, OR asking for clarification>
  - <after that it should go back and forth reaching a conclusion and then documented in the Response and Description above>
  - <if the negotiation fails or stalls, the TSC will do it's best to help>

Test Tools Requirements

Other