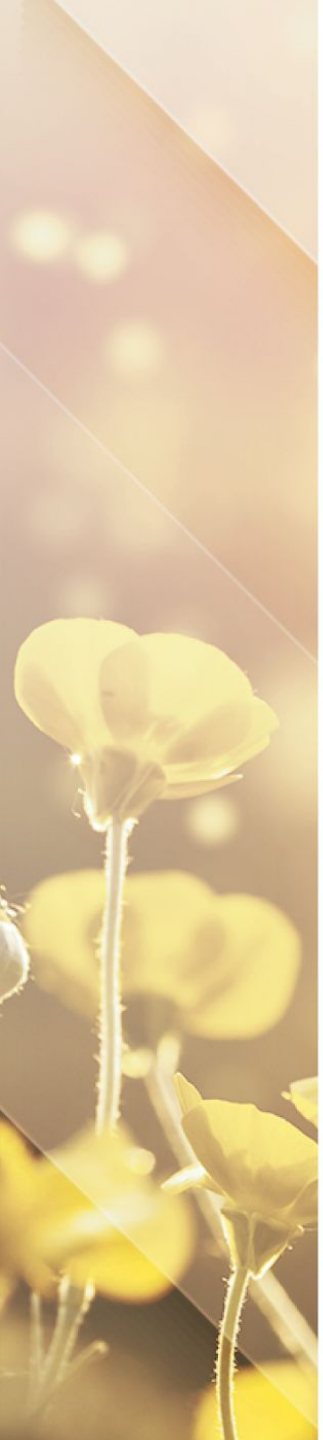

OPEN
DAYLIGHT

OpenDaylight Technical Architecture and key technologies

Brady Johnson
ODL SFC Project Technical Lead
brady.allen.johnson@ericsson.com



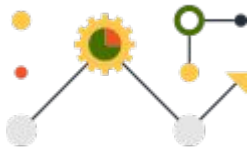
Agenda

- OpenDaylight Architecture
  - MD-SAL, SouthBound, Karaf
  - Key OpenDaylight projects
- OpenDaylight Governance and Leadership
  - The Board of Directors
  - The Technical Steering Committee
  - Projects and committers
- Getting involved in OpenDaylight
  - Local User Groups
  - User Advisory groups
  - Email lists and meetings
  - Committing code
- OpenDaylight release planning

OpenDaylight Architectural Overview

OpenDaylight (ODL) is an Open Source platform to enable networking services and technologies.

The OpenDaylight platform provides a flexible, common platform underpinning a wide breadth of applications and Use Cases.



Automated Service Delivery

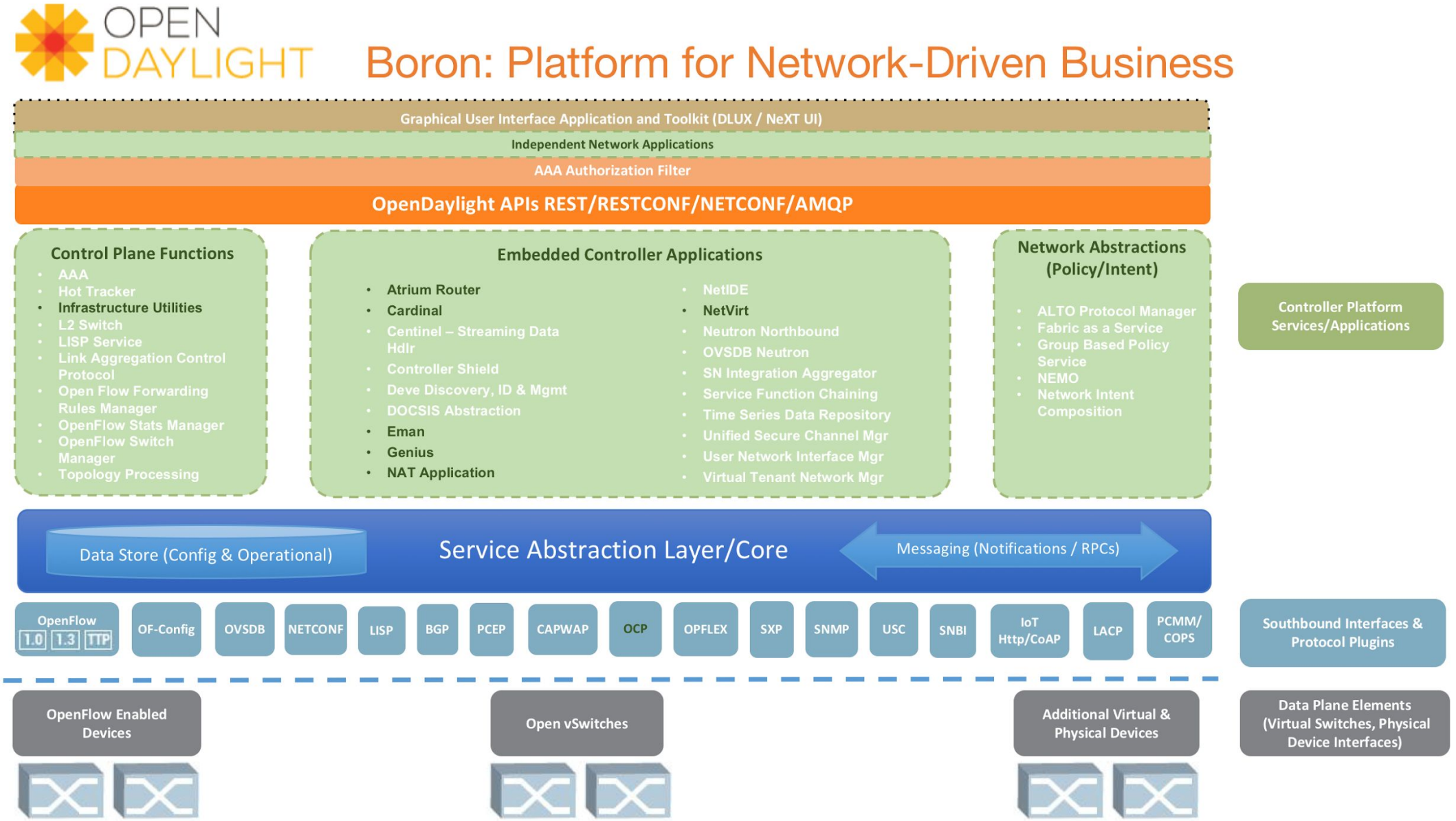
Cloud and NFV

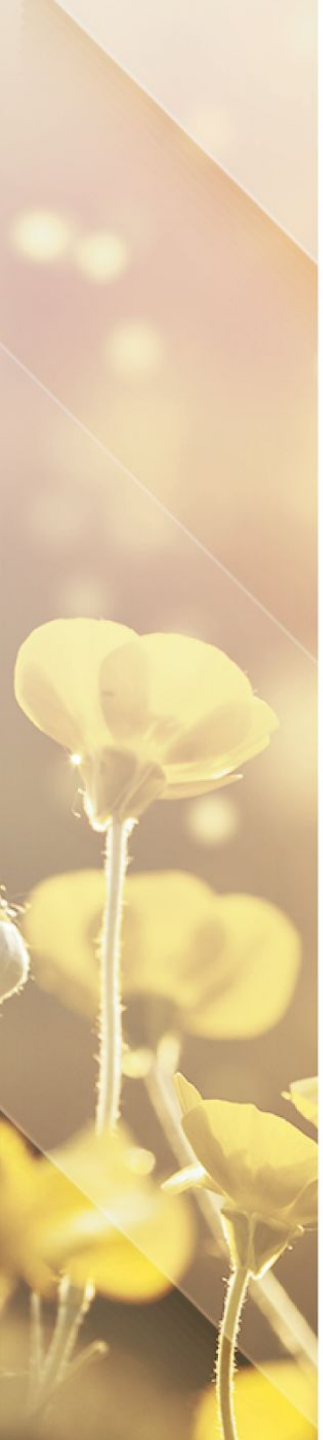
Network Resource Optimization

Network Visibility and Control

Research, Education, and Government

OpenDaylight Boron Architecture





OpenDaylight Architecture details

Key OpenDaylight Architectural details

- Model Driven Development (YANG)
- MD-SAL
  - **M**odel **D**riven **S**ervice **A**bstraction **L**ayer
  - Provides messaging and data storage functionality
- Modular plugin SouthBound interface approach
- Karaf OSGi container
  - Runtime environment

MD-SAL and YANG

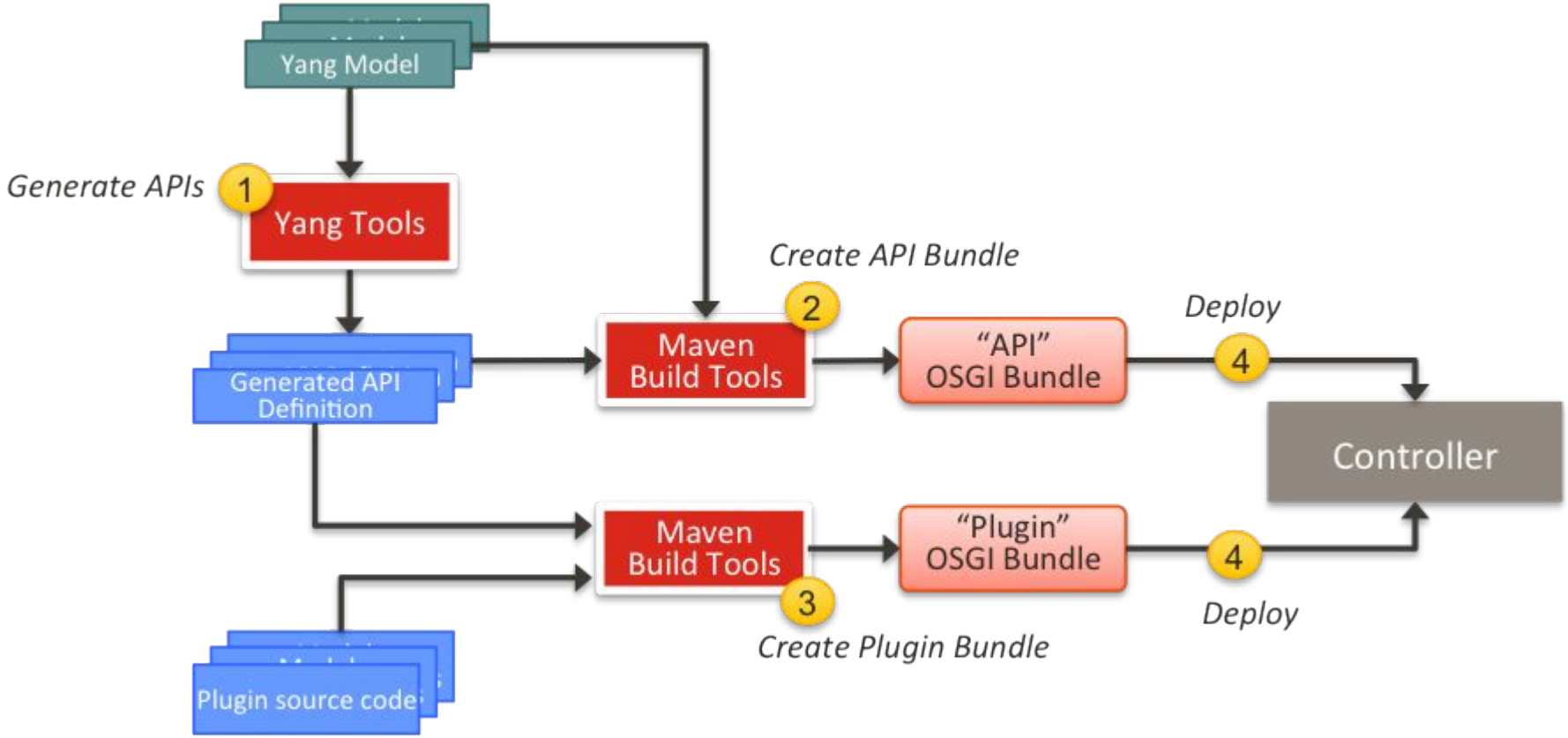
The Model-Driven SAL (MD-SAL) is an infrastructure component that provides messaging and data storage functionality based on user-defined (application developers) data and interface models

- Model Driven Architecture
  - Runtime and Compile time code generation
  - YANG modeling language (rfc6020)
  - Different ODL application communicate via the Data Store as opposed to using traditional APIs ⇒ ***Decoupled***

Basic concepts are building blocks, from which MD-SAL derives its services, behavior is based on a mapping of basic concepts to developer-supplied YANG models

- Data Tree
  - All state-related data are modeled and represented as a data tree
  - Possible to address any element / subtree
  - Configuration Data Tree - Intended state of the system or network, populated by consumers, which expresses their intention.
  - Operational Data Tree - Reported system state, published by providers using MD-SAL. Feedback loop for apps to observe system state
- Instance Identifier
  - Unique identifier of node/subtree in data tree, provides unambiguous info, how to retrieve node/subtree from conceptual data trees
- Notification
  - Asynchronous transient event (from perspective of provider) which may be consumed by consumers and they may act upon it
- RPC
  - Asynchronous request-reply message pair, when request is triggered by consumer, sent to provider, which in future sends reply message

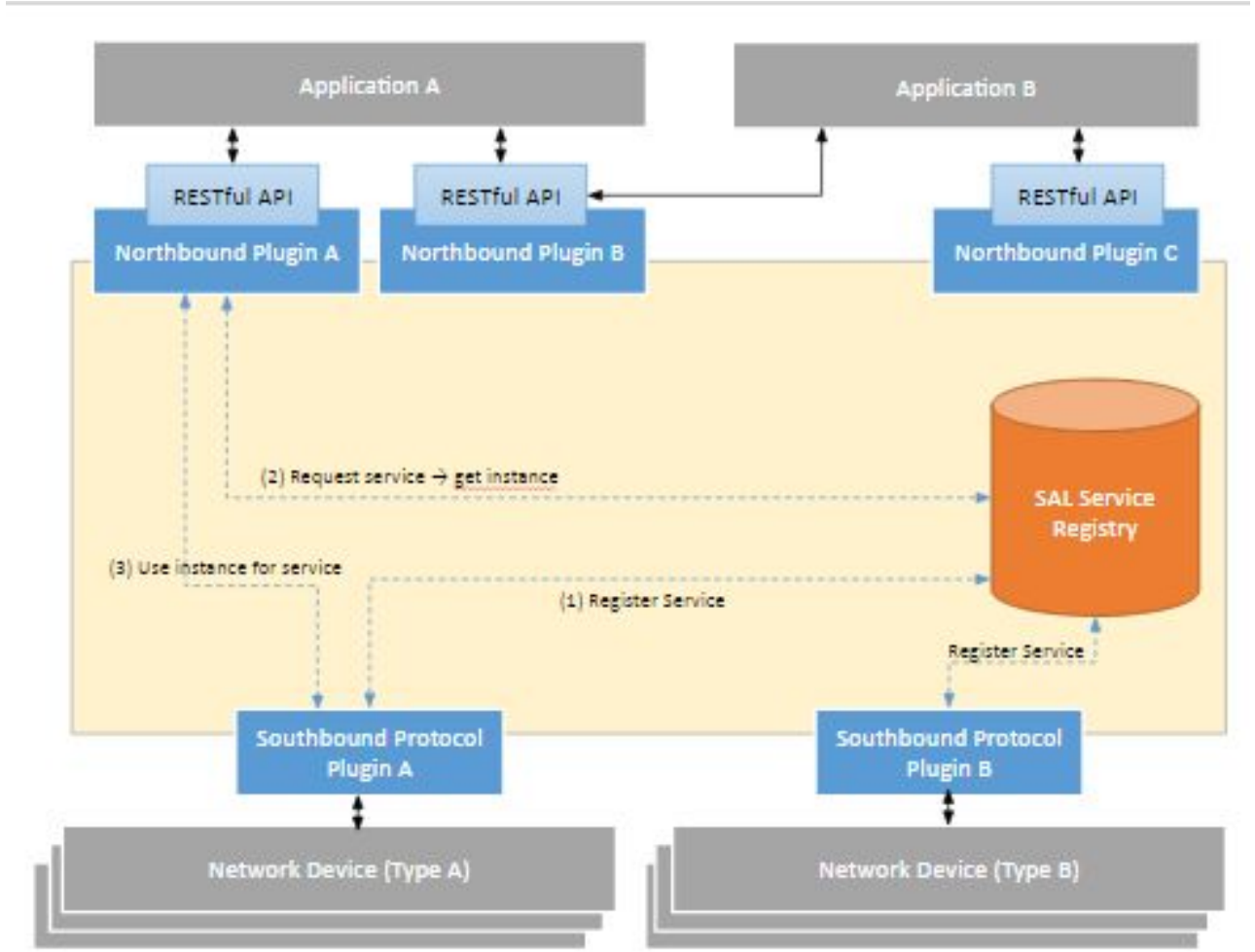
Application Development with YANG



OpenDaylight Modular SouthBound

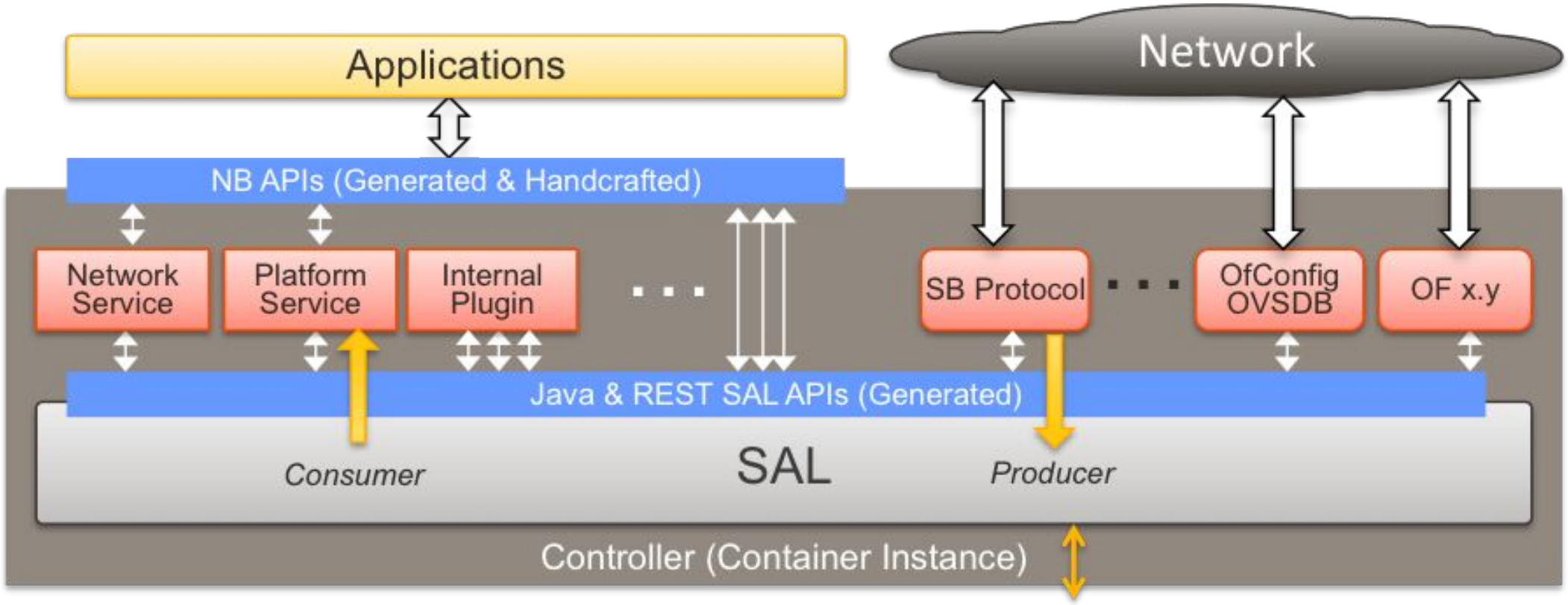
The SouthBound plugins are designed with a Modular plugin interface

- Extensive support for standard network management interfaces



OpenDaylight Modular SouthBound
continued

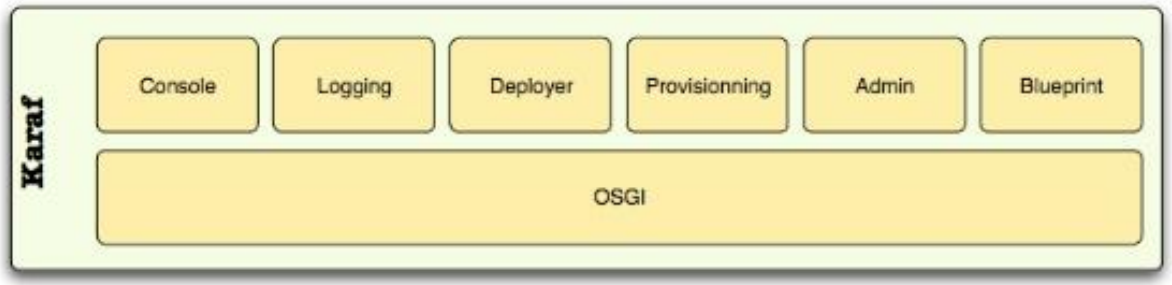
Another example



OpenDaylight Runtime Environment

Karaf: an OSGi container

Karaf is a small OSGi based **runtime** which provides a lightweight container onto which various components and applications can be deployed.

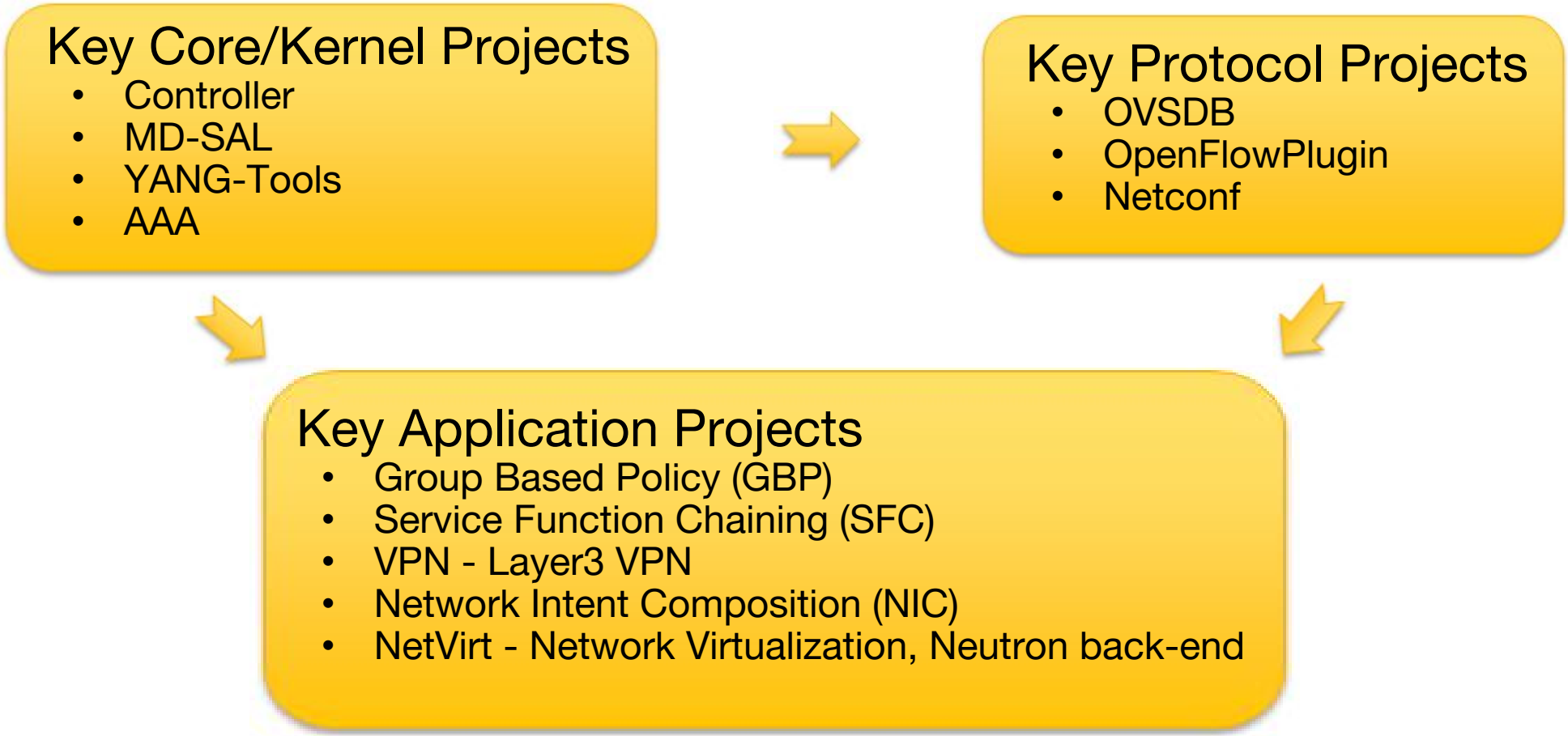


Apache Karaf can trace it's origins back to the Apache ServiceMix project's Kernel.

OSGi (Open Service Gateway Initiative)

A Java **framework** for developing and deploying modular software programs and libraries. Each bundle is a tightly coupled, **dynamically loadable** collection of classes, jars, and configuration files that explicitly declare their external dependencies (if any)

Key OpenDaylight Projects



Complete Project list

- https://wiki.opendaylight.org/view/Simultaneous_Release:Boron_Release_Plan#Participating_Projects

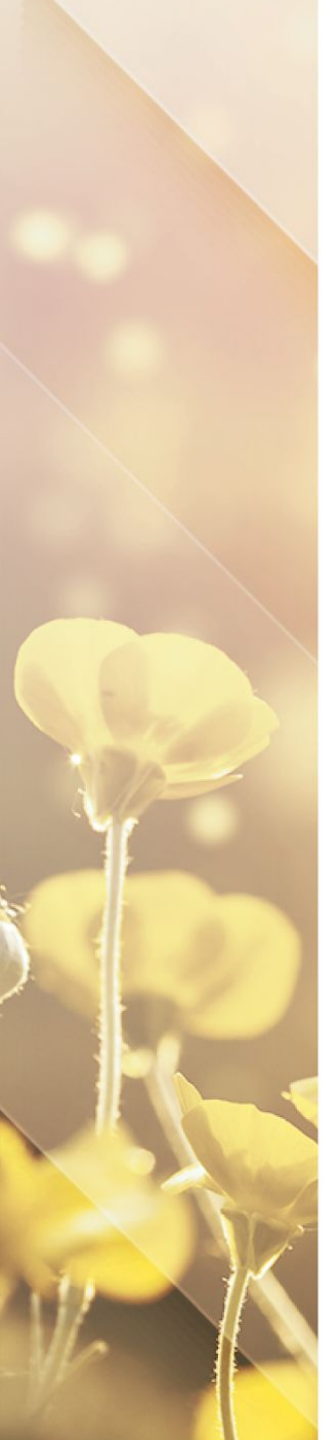
OpenDaylight Governance and Leadership

OpenDaylight Governance

OpenDaylight is funded and driven by the Linux Foundation
http://www.linuxfoundation.org/

- OpenDaylight Board of Directors
  - Manages the business leadership for OpenDaylight including finances, governance, marketing and operational decisions
  - Does not get directly involved in technical decisions

- OpenDaylight Technical Steering Committee (TSC)
  - Comprised of elected ODL members
  - Provide technical leadership and direction for the OpenDaylight projects and community



OpenDaylight Projects and committers

- OpenDaylight projects
  - Each project has a Project Tech Lead (PTL) that is responsible for the technical aspects of the project
  - Project interface to the ODL TSC
  - Project Leaders are elected by project committers

- Contributor versus Committer
  - Anybody can be a contributor
  - Committers are per project, and are elected by the project and the TSC
  - A contributor can ***write and submit code***, but only a committer can ***merge*** code into the code base

Getting involved in OpenDaylight

Getting involved in OpenDaylight

OpenDaylight User Groups (ODLUG)

- https://www.opendaylight.org/odlug
- Regional, self-organized, informal associations that meet globally to discuss OpenDaylight
- Currently there are 28 global ODLUGs with 1000s of members
- The Madrid ODL UG
  - http://www.meetup.com/OpenDaylight-Madrid-User-Group/

OpenDaylight Advisory Group

- https://www.opendaylight.org/advisors
- The Advisory Group has been established to assist and support ODL in its objectives by providing technical and strategic guidance to the ODL Technical Steering Committee and ODL Developers based on challenges of running a real-world network

OpenDaylight mailing lists

- https://lists.opendaylight.org/pipermail/
- The ODL mailing lists are very active
- Each project has an ODL mailing list
  - Great way to follow what's happening in a project
- Interesting email lists: TSC, Release, Discuss

Getting involved in OpenDaylight, continued

Weekly Meetings

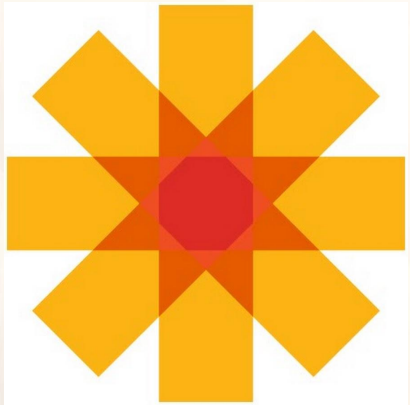
- All ODL meeting information and links
  - https://wiki.opendaylight.org/view/Meetings
- Each project has a weekly meeting
- TSC weekly meetings
  - Thursdays, 17:00 UTC (19:00 Madrid)
  - All PTLs should attend
- Technical Workshop (TWS) weekly meetings
  - Mondays, 17:00 UTC (19:00 Madrid)
  - Forum for new ideas, technology, etc

Contributing code

- Code contributions are always very welcome
- Bug fixes and Unit Tests are a great place to get started
- Gerrit code reviews and Jenkins CI is used to facilitate code submissions

MD-SAL Toaster example

- Great examples of how to start using ODL
- https://wiki.opendaylight.org/view/OpenDaylight_Controller:MD-SAL:Toaster_Step-By-Step



OpenDaylight Releases

OpenDaylight Release Planning

Milestones

- M0 – Official project start
- M1 – Declare Project intention to participate
- M2 – Release plan complete
- M3 – Feature/Scope freeze
- M4 – API Freeze
- M5 – Code Freeze

There is usually 1 month between Milestones

Project Offsets

- Each project is one of: Offset 0, 1, or 2 (Core, Protocol, Application projects respectively)
- Each project has its Milestones ***offset*** according to the downstream dependencies it may have.
  - add 2 weeks to offset 1 milestones, and another 2 weeks for offset 2 milestones

Release Candidates

- RC0-RC3, builds are tested for the Formal Release
- Taken weekly

Formal Release

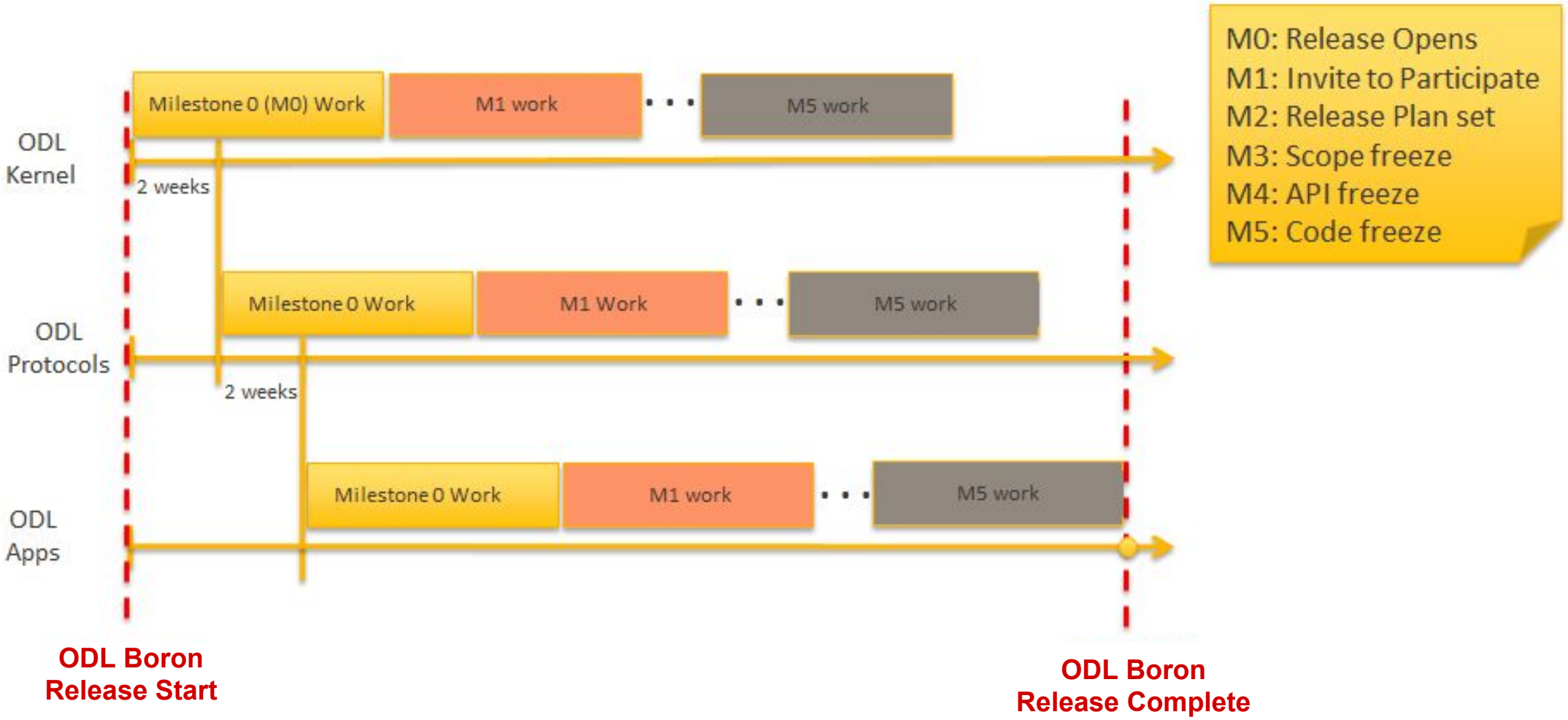
- The official OpenDaylight release

Service Releases

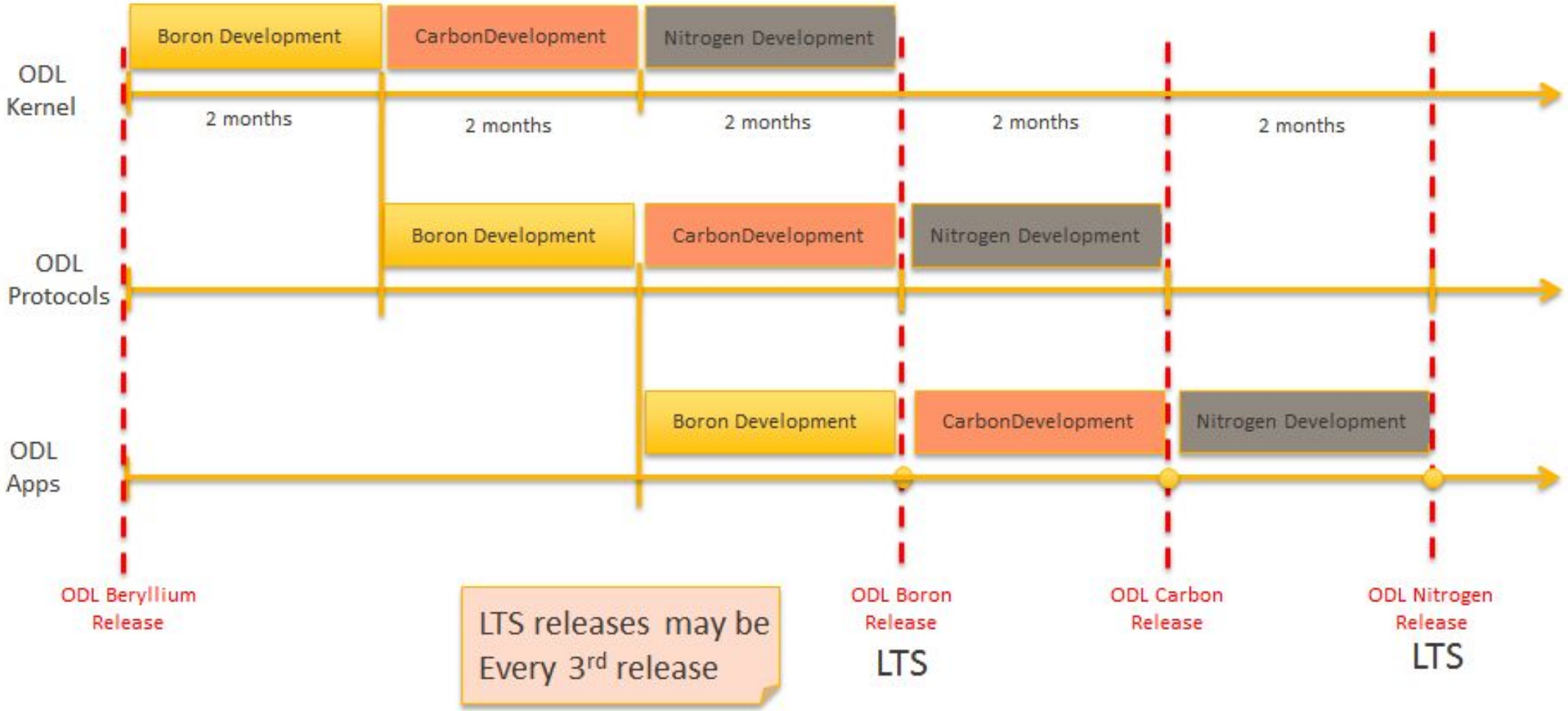
- Bug Fixes on the Formal Release

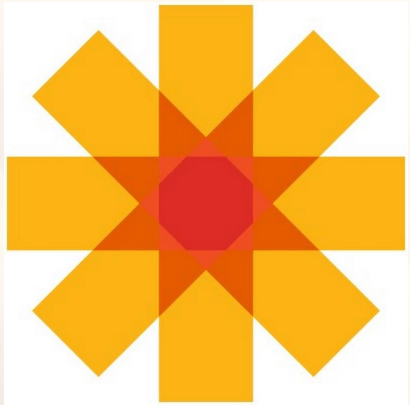
https://wiki.opendaylight.org/view/Simultaneous_Release:Carbon_Release_Plan#Schedule

Current OpenDaylight Releases



Proposed Fast Phased Release

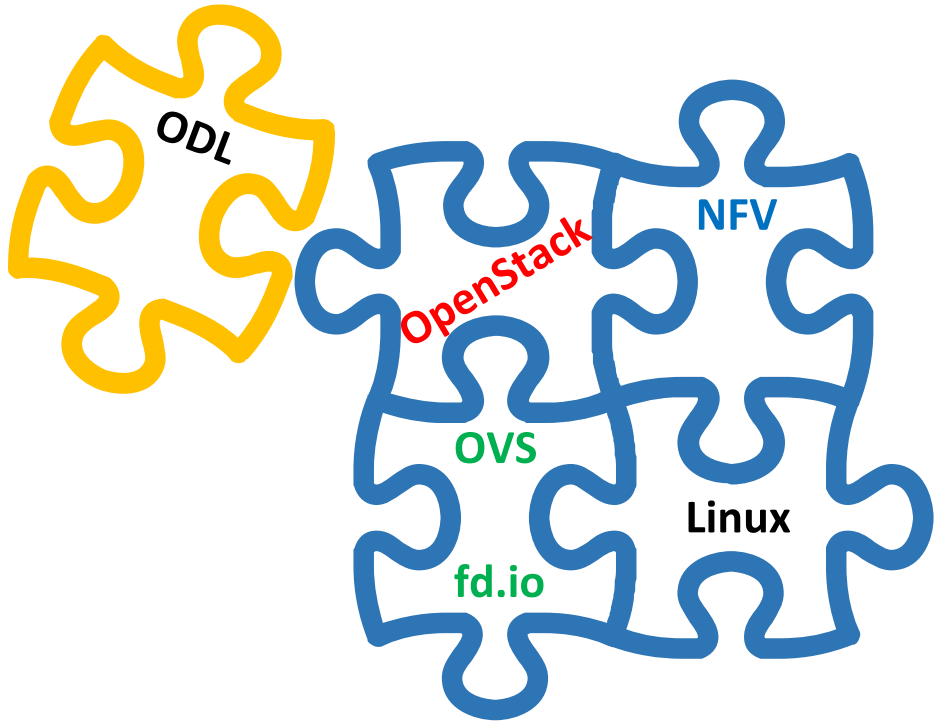




Questions & Answers


opendaylight.org/odlboron
5
B
Boron
Now available — what will you produce?
#OpenSDN

OpenDaylight: Just 1 piece of the puzzle



OPNFV

- OPNFV: Integrating it all together
  - https://www.opnfv.org/
  - https://wiki.opnfv.org/display/sfc/Service+Function+Chaining+Home