

Federation

Welcome to

- [Welcome to](#)
- [Introduction](#)
- [Scope and Interfaces](#)
- [Release Plan](#)
- [Test Plan](#)
- [Documentation](#)
- [Useful Links](#)

Introduction

In order to allow multiple ODL clusters to cooperate (federate) for a given application, there is a need to allow state exchange between ODL clusters. While a lot of this exchange may be application specific, there are common infrastructure elements that may be required to allow such federation. These include:

1. A means for initial (full) synchronization between remote clusters
2. A means to publish state updates between clusters while preserving event ordering and possibly allowing data transformation and filtering
3. A means to subscribe to state updates from remote clusters

The purpose of this project is to implement such infrastructure which would listen on events in MD-SAL, allow for their transformation/manipulation and filtering by an application specific plugin, interface with one of a number of messaging infrastructure candidates ("The Federation Platform") and publishing event updates for other ODL clusters to consume by means of a federation service subscriber interface, possibly after transforming the updates into new events to be inserted into the remote MD-SAL.

The Federation Service is intended to complement other services such as [messaging4transport](#) and the [Conceptual Data Tree](#) which are focused on state sharing between ODL clusters (and potentially other systems) without the need for transformation, filtering or event ordering.

As a first application use case, NetVirt federation for the purpose of Neutron network federation is to be implemented using this infrastructure.

Below is a high level diagram of the relationship between the Federation Service, other ODL services, MD-SAL and state sharing mechanisms such as [messaging4transport](#) and the [Conceptual Data Tree](#)

[blocked URL](#)

As depicted in the block diagram above, the Federation service will have API's with application specific Federation Plugins, will interact with MD-SAL for the purpose of configuration state as well as listening on designated events and will have a North bound API towards an off-the-shelf federation platform, such as RabbitMQ or Akka Pub Sub.

Scope and Interfaces

It is planned that NetVirt will use this service as a first example for federation if configured to share state with remote ODL clusters.

[blocked URL](#)

Within the Carbon release, the Federation Service will implement a the above architecture allowing the insertion of plugins adhering to the plugin API's. Full sync functionality will be within the scope of the specific plugin, as this is application specific, while using the federation guidelines regarding messages ID sequencing.

Release Plan

[Carbon Release Plan](#)

Test Plan

[Federation Test Plan](#)

Project Facts

Project Creation Date: October 27, 2016

Lifecycle State: Incubation

Type: Kernel

Primary Contact: Gideon Kaempfer <gidi@hpe.com>

Project Lead: Gideon Kaempfer <gidi@hpe.com>

Committers:

- Lior Baram <lior.baram@hpe.com>
- Gideon Kaempfer <gidi@hpe.com>
- Dovev Liberman <dovev.liberman@hpe.com>
- Guy Sela <guy.sela@hpe.com>

IRC: [freenode.net](#) #opendaylight

Mailing List: federation-dev@lists.opendaylight.org
Archives: [mailing list archives](#)

Meetings: See [Community Meetings](#)

Repository: git clone <https://git.opendaylight.org/gerrit/federation>

Jenkins: [jenkins silo](#)

Gerrit Patches: [code patches /reviews](#)

Bugs:

- [open bugs](#)

Documentation

[Developer Guide](#)

[Installation Guide](#)

Useful Links

[Federation Service Weekly Meeting Invite](#)

[Project proposal page](#)