SPRING-OPEN - An ONF TAG Project

Introduction

Overall, the Open Networking Foundation has seen considerable progress, as the key forum for SDN, promoting physical separation of data and control planes, and producing public vendor-agnostic APIs for both. Accordingly, membership continues to grow, working groups continue to expand in number and charter, and R&D investment continues to grow.

That said, traction for OpenFlow scenarios in *physical switching* could be improved. Part of the problem could be perception, stemming from the evolution of very simple abstractions in OF1.0 to the richer abstractions in OF1.x. As a result, today, a commercial OF-like solution might use proprietary extensions in some areas, and simplifications or subsets in other areas. Part of the problem could be real, in that the without the pressure of real scenarios the specs and work products emerging from OF might miss the mark on where to draw the line between core functionality and extensibility.

To settle these questions, to get better traction, and to get it more quickly, the ONF Technical Advisory Group (TAG) is setting out to seed scenario-focused efforts, with the intent to produce open-source solutions based on stable versions of the OpenFlow spec. Contributions from the working group leadership and membership will be vital. Contributions from ONF member institutions are also critical. To the degree that ONF specs and work products fall short, this should come to light in pursuing these scenarios, and get fed back to the ONF community to fix. Importantly, while the open-source solutions produced must deliver on the core requirements, the solutions must also provide hooks for extensibility to foster innovation and differentiation in the marketplace.

Proposed Solution:

The TAG believes that the best way forward is to *demonstrate* the technical maturity of the ONF work in a networking scenario that matters to the stakeholders both in ONF and beyond, namely, the network operators.

By identifying the core set of requirements for the chosen networking scenario, the TAG can then create a small team of engineers and developers that can execute on the requirements to build a small but useful artifact as an open-source effort, leveraging ONF protocols and mechanisms. The team would comprise a technical project-lead chosen by the TAG, and engineering/developer-time contributed by member companies. With a bounded budget, timeline, and clearly defined deliverables, the open-source effort would be expected to jump-start innovation in OpenFlow-based SDN mechanisms that involve physical switching and routing. If successful, such an effort can be repeated with other networking scenarios of interest to the ONF stakeholders.

The project requirements and development work must adhere to the following six guidelines.

- First, the chosen networking scenario must use physical-switching hardware, as the TAG believes that SDN based software (or virtual) switching scenarios have been successfully demonstrated (especially in the datacenter), but hardware-switching based SDN has lagged behind.
- Second, the solution implementing the networking scenario must use ONF protocols and mechanisms, namely OF-Wire, OF-Config, SDN Controllers and the like. The TAG realizes that there may be other SDN or non-SDN (traditional) networking means to achieve the goals of the networking scenario. Nevertheless, the focus of this project is to use ONF mechanisms.
- Third, the project must provide feedback to the TAG, Board, ONF WGs, and ONF DGs, about the standards and mechanisms used – what worked, what didn't, what was not implemented and why, what the gaps or inefficiencies in the protocols are, what the hardware/software limitations are, and so on.
- Fourth, the solution must use available, commodity switching parts, as the goal of the work is to be open-source, reproducible, and extensible. The TAG believes that the use of merchant silicon and white-box hardware is beneficial towards meeting this requirement.
- Fifth, wherever possible, the project should be designed such that a diversity of parts is admissible either as part of the project deliverables or as a potential extension. For example, use of a certain white-box part should not preclude the use of other white-box parts, nor should it preclude the use of commercial vendor equipment, as long as they meet the other project requirements.
- Finally, the solution must be designed to be reproducible and extensible for value-add. The TAG believes that this requirement will spur innovation by allowing the project-solution to serve as the core-kernel for future, more fully featured open-source and commercial solutions. Indeed, the TAG invites all members of the SDN community to contribute compatible adjacent capabilities as the means for doing so becomes evident.

Chosen Scenario: SDN Based WAN Control

A wide range of potential scenarios and approaches were discussed in the TAG for months, including data-center, WAN, and NFV-oriented scenarios. Ultimately, the SDN based WAN control scenario was chosen. The focus is on demonstrating the minimum set of capabilities necessary for the chosen networking scenario. In this particular case, besides basic unicast routing, it involves supporting capabilities for policy-routing, traffic-engineering and steering, as well as recovering from failures in both data and control planes, supporting horizontal-scaling, supporting a visibility & diagnostics framework, and supporting configuration and consistent-updates.

In any networking scenario, the data plane needs to follow some high level rules. In the chosen WAN scenario, the data plane routes unicast IPv4 packets with standard MPLS operations that abide by Segment Routing principles of globally-significant labels and source routing [1]. Segment Routing shares a lot of goals in common with SDN,

including the elimination of complex distributed protocols and the use of source-routing via controllers or head-end routers. In addition, SR requires no change to the existing MPLS data plane [2], making it an attractive-candidate for rapid prototyping via available commodity merchant-silicon ASICs and white-box hardware, and enabling telecom operators to relate to the scenario with high interest.

The project will deliver an Open Segment Router (OSR) that routes unicast IPv4 packets using the MPLS data plane. In keeping with ONF SDN principles, the project will also deliver a controller as part of the solution, responsible for operating an island of OSRs (Fig. 1).

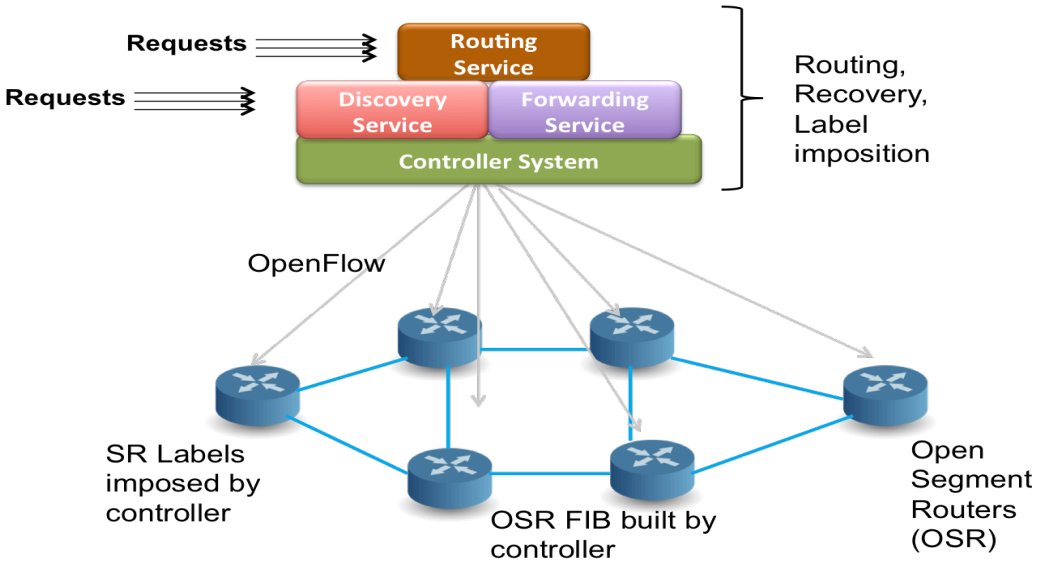

Fig. 1 WAN SDN Control of Open Segment Routers

The controller should support an externally visible Discovery Service and a Routing Service. The Discovery Service will be responsible for bootstrapping and configuring the network, discovering node-capabilities, discovering and maintaining the topology graph, providing statistics and troubleshooting services, and finally implementing an API for the Routing Service as well as external requests. The Routing Service will be responsible for default routing on the configured network using Segment Routing principles like Node Segments and ECMP. It should also support capabilities allowing for Policy Routing, Traffic Engineering and Steering. The project will build a reproducible system prototype and demonstrate all of these features.

The TAG invites early and open involvement of the ONF Working Groups in this project. ONF plug-fests and interoperability events have proven invaluable in figuring out the limitations, bugs, gaps, and deficiencies of the protocol-semantics and usage [3]. These have been fed back to the Extensibility WG, and have made their way into subsequent versions of the protocol. The TAG project will go one step further by implementing a networking scenario on top of the same protocols, thereby potentially increasing the scope and significance of the feedback. By centering the effort with a networking scenario-based approach, while keeping it open and vendor-generic, the TAG

expects high involvement of the Extensibility, Config Management, and Testing & Interoperability WG efforts.

Similarly, the project expects to develop and contribute to the FAWG by defining a new TTP for an MPLS data plane. The TAG also believes that the project presents an incredible opportunity for contributions to the NBI WG from a realistic use-case that is implemented in open-source. Similarly there is much that can be exposed to the Migration WG in figuring out how such an SDN controlled, segment-routed island can interoperate with traditional routers. Likewise, the TAG also believes that the Optical and Wireless WGs could benefit from an example of SDN-based WAN control, as the use cases of those WGs are typically in WANs. Finally, the project presents an invaluable tool for the Market Education WG to exploit towards spreading the ONF SDN message via a complete implementation that highlights the entire SDN stack, and embodies all that ONF stands for.

Project Goals

In the chosen networking scenario – *SDN based WAN control* – the project must demonstrate a small but functional prototype using open-source, readily available, reproducible parts using ONF mechanisms and protocols. The project has the following set of goals:

1. Demonstrate maturity and scale of the ONF work product in hardware readily available today using the latest stable versions of ONF protocols – e.g., OF 1.3.4.
2. Provide feedback to ONF WGs on their work product from an implementation of the chosen networking scenario.
3. Promote adoption by creating a core-kernel that is extensible for value-add towards deployment, interoperability and differentiation.

The success of the project will be measured on successfully delivering the Project Deliverables detailed below.

Non-Goals

The project does not aim to do the following:

1. It does not aim to create a Generally Available (GA) product. It will not undergo typical product Quality Assurance (QA), nor will it be ready for production or interoperate with other networks and network control planes. Instead the project will support certain elements that help in the 'productization' of SDN ideas, like ease-of-configuration, visibility and troubleshooting.
2. The project also does not aim to deliver any specific service like a full-blown bandwidth-managed TE service, a VPN/VPLS/VPWS service, or an NFV service. It will however, support the core capabilities required to build such services (and others) on top – see extensibility choices below.
3. Finally, the project does *not* aim to exclude the use of other parts, in the data plane as well as the control plane. In the interest of time and limited resources, choices will be made for building the system. However such choices should be replaceable

by other parts, both commercial and open-source, as long as they conform to the requirements.

Project Deliverables

The project has the following deliverables.

1. An Open Segment Router (OSR) on 1 hardware platform that meets the SPRING-OPEN Data Plane Requirements (outlined below).
2. A WAN Controller that meets the SPRING-OPEN Control Plane Requirements (outlined below).
3. A System Prototype and Demonstration of WAN control of an island of OSRs, which demonstrates several discovery and routing scenarios, and is extensible towards interoperability and deployment (see options for extensibility below).
4. Feedback (written documentation) to ONF WGs and larger community on - what worked, what didn't, gaps and inefficiencies in the protocols, what was not implemented and why, hardware/software limitations, common/best practices and so on.

Project Timeline

The expected project timeline for this work is six months, with a rough breakup as follows:

May – June'14: Independent development of OSR and WAN Controller
July – August'14: OSR and WAN Controller integration
Sept – Oct'14: System Prototype and Demonstration

If necessary, the project may extend a few more weeks, but all deliverables need to be met before the holiday season at the end of the year.

Checkpoints: The project team is expected to provide the following updates

Status update each week at the TAG meeting
Once a month, provide an update at the CoC meeting
On demand, provide updates at individual WG meetings

Team Composition / Roles

The project team comprises a project lead, designated by the TAG and under contract with the ONF, as well as engineer/developer time contributed by member companies. Saurav Das, an independent consultant [4], has been designated as the project-lead. The roles of the team members are as follows:

Project Lead

- Architect the System/Network with assistance and guidance from the TAG
- Manage team members, roles and responsibilities, dev tasks, monitoring progress, taking corrective actions etc.
- Responsible for interactions with TAG and ONG WGs
- Responsible for project deliverables and timelines

- Develop controller and/or switch code

Project Switch Team (at least 2 devs)

- Responsibilities include delivering an OSR on at least 1 platform, according to the SPRING-OPEN Data Plane Requirements
- Task may require integrating a number of open source parts, and/or creating open and closed-source components
- Skills:
    - Switch ASIC SDK engineer / C developer
    - Switch software engineer, Linux/C/Networking developer
    - Must have prior experience developing switch/router software
    - Experience with OpenFlow a big plus
- Responsible for working with the Controller team to deliver the System Prototype and Demonstration

Project Controller Team (at least 2-4 devs)

- Responsibilities include delivering a WAN Controller, according to the SPRING-OPEN Control Plane Requirements
- Task will require integrating a number of open-source parts, and creating more open-source parts from scratch
- Skills:
    - Distributed Systems / Java/Python/C++ developer
    - Network Systems / Java/Python/C++ developer
    - Prior experience developing controller software a big plus
- Responsible for working with the Switch team to deliver the System Prototype and Demonstration

Project Contributions from Member Companies

- Controller and Switch team members dedicated for the project timeline (six months)
- The Prototype lab is expected to be hosted by a member company - includes rack/power/dev-space/monitors etc. Free lunch a big plus :)
- Equipment to be donated by member companies
    - bare-metal switches - at least 6
    - servers - at least 10

The TAG is currently working on data and control plane requirements for the project. A preliminary set of data plane requirements is indicated below. These are intended to represent the scope that the TAG team will undertake but are not intended to restrict any auxiliary contributions by other ONF members.

SPRING-OPEN Data Plane Requirements

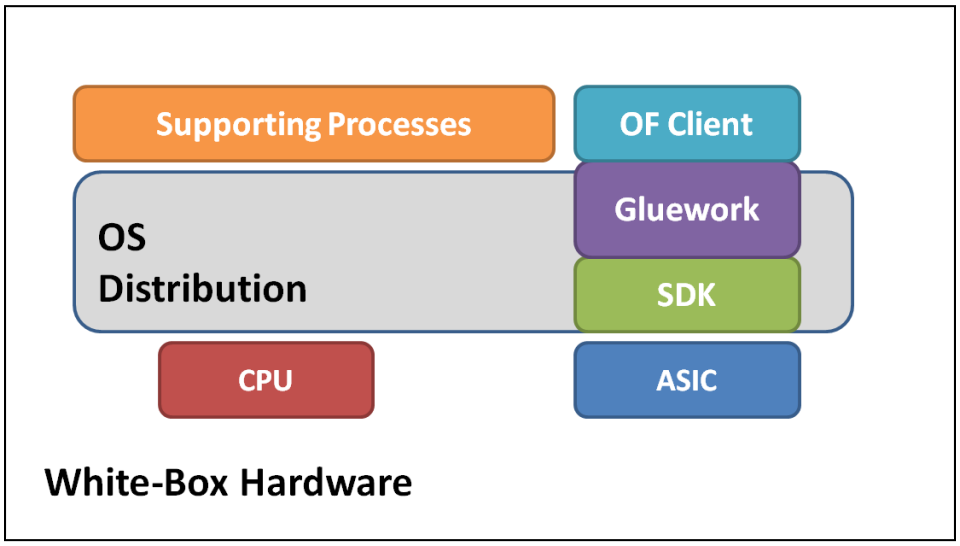

Fig. 2 SPRING-OPEN Data Plane

1. Merchant Silicon
   - No custom ASICS, no FPGA
   - Widely available/deployed
   - Must support MPLS operations
     - push/pop/swap labels
     - label stack – BOS indicator/PHP
     - Pushing multiple labels
     - Popping a label, and pushing multiple labels
     - Copying IP TTL to MPLS and back/ TTL decr.
     - Copying IP ToS to MPLS EXP/ DSCP optional
     - ECMP in the presence of labels / Entropy label
     - Fast Reconvergence
     - Optional DPI capabilities and user-defined FEC classification bits
   - One company/one chip preferred

2. White-Box Hardware
   - Std Config: 48 X 10G + 4 X 40G or 64 X 10G
   - Widely available/deployed
   - Usually with PowerPC CPU/ consider x86 option

3. Operating System
   - Open-source Linux distribution
   - Preferably OCP compliant (e.g., ONIE bootable)

4. Supporting Processes
   - Must be open-source
   - CLI/SNMP/OFConfig/ LED, SFP control etc.

5. OpenFlow Client
   - Must be open-source
   - Must be OF 1.3+
   - Preferably use a single OF client even if box/OS are diverse

6. ASIC SDK + HAL + Gluework (to OF Client)
   - No choice but to be closed-source
   - Must be freely downloadable as binary

SPRING-OPEN Control Plane Requirements

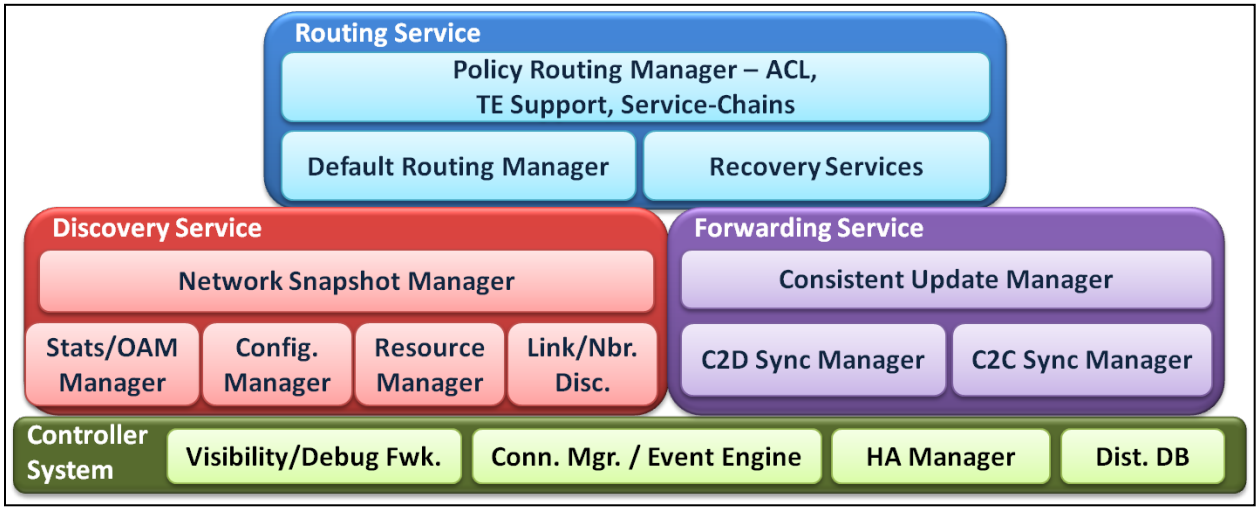

Fig. 3 SPRING-OPEN Control Plane

1. Controller System
   - Should support horizontal scaling via persistent distributed storage and notifications
   - Connection manager for data-plane switches; event engine supporting pub/sub mechanism for subscribing to and propagation of events
   - High Availability manager performing leader election
   - Visibility and Debug framework for the control plane – GUI/CLI/Troubleshooting

2. Discovery Service

- Resource manager that discovers node/link characteristics, capabilities and constraints
- Link and Neighbor discovery
- Configuration manager that handles all network configuration including the scope for identifiers and namespaces, as well as their association with nodes and interfaces; provides proxy services like ARP, ICMP etc.
- Statistics/OAM Manager for data plane statistics and troubleshooting – eg. intf counters, LSPing, BFD config etc.
- Network Snapshot Manager – provides network wide view of topology, traffic, capabilities and resource limits; maintains API for requests from Routing and Forwarding services as well as external requests

3. Routing Service
   - Default Routing Manager: performs SPF routing with Node segments, ECMP and PHP.
   - Recovery Services for convergence and protection
   - Policy Routing Manager: connectivity management using ACLs, policies for avoiding links and nodes, TE support via strict explicit paths, load balancing over non-equal-cost paths, service function chaining

4. Forwarding Service
   - Responsible for all forwarding state in the network –all table entries in all routers
   - Manages all consistency requirements when updating forwarding state
   - Control-to-Data plane Sync Manager – performs state-synchronization between controller view and individual switch view
   - Control-to-Control plane Sync Manager -- performs state-synchronization between controller instances.

5. Generic Requirements
   - Must be completely open-source
   - Programmer friendly language, library maturity
   - Multi-threaded performance
   - NBI – support for REST API, extensible
   - SBI – support for OF1.3

Options for Extensibility:

The TAG project for the chosen networking scenario can be extended in several ways:

- Extend the controller for hierarchical, geographically distributed control
- Add E-BGP on the controller (aka RCP[5], IRSCP[6] or NTT[7] work) for exchanging reachability information, route selection and more
- Provide L3VPN/VPLS/VPWS services

- Provide full blown TE solution with bandwidth optimization, calendaring etc.
- Extend control plane to work with optical switches / networks
- Interoperability with traditional LDP/IGP control plane
- In-band control
- Add FRR to data plane recovery
- Deeper buffers & QoS in white-box platform
- Scale-out Segment Routers with white-boxes
- More OAM / troubleshooting features
- Security features
- Multicast/IPv6 … and much more

Action Item:

The ONF TAG invites comments from member companies and WGs on solutions that satisfy the SPRING-OPEN Data and Control Plane Requirements, and welcomes offers of resources (both human and otherwise) from those who want to help bring the project to a successful conclusion.

References

[1] http://www.ietf.org/id/draft-filsfils-rtgwg-segment-routing-01.txt

[2] http://www.ietf.org/id/draft-filsfils-spring-segment-routing-mpls-00.txt

[3] https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow-test/onf-testing-interop-june-2013-tech-issues-v0.4.pdf

[4] http://www.linkedin.com/pub/saurav-das/24/80a/b88

[5] M. Caesar, et.al, Design and implementation of a routing control platform, NSDI'05.

[6] J. van der Merwe, et.al, Dynamic connectivity management with an intelligent route service control point, SIGCOMM' 06 workshop.

[7] https://www.opennetworking.org/images/stories/downloads/sdn-resources/use-cases/Migration-WG-Use-Cases.pdf