When SDN meets legacy IP control planes

Dagstuhl Seminar on Software Defined Networking
Wadern, Germany, 7 Sep 2012
Agenda

- Intro and overview of the RouteFlow project
- Ideas of the HotSDN paper
- Raise the debate on:
  +++ Transitioning existing networks to Openflow/SDN
  +++ Hybrid Openflow/SDN approaches
     (integration with legacy control planes)
  +++ How OpenFlow direct FIB manipulation can help IP routing
     control applications and enable cost-effective architectures?
- **Shape the research agenda of RouteFlow!**
  [Happy to talk about the low-level details:
   e.g., how to do IP forwarding (match+actions) with OF1.0 and 1.X]
Software Defined IP Routing

- Specialized Features
- Specialized Control Plane
- Specialized Hardware

High cost
Specialized config.
Closed source
Slow innovation

Low cost (commodity)
Multi-vendor modularity
Open source
Rapid innovation

Source: McKeown
RouteFlow Project History

- Start Msc. Thesis work by Marcelo N.
  - First Prototype
  - First Short-Paper @ WPEIF
  - QuagFlow Poster @ SIGCOMM
- Aug / 2010
  - Open-Source Release
  - Evaluation on NetFPGA testbed
- Dez / 2010
  - Demos @ ONS11
  - Tutorial & Demo @ OFELIA/CHANGE SS
- May / 2011
  - Indiana University - Pronto OF switches + BGP peering with Juniper MX
- Oct / 2011
  - Demo @ SuperComputing 11
    - Demos @ ONS12
      - HotSDN Paper
        - Running on FIBRE / OFELIA testbed
        - Collaboration with NTT
... building a community

Visits: 18,000+ (8,000+ Unique)
From over 1,600 cities of 100+ countries all over the globe!

http://go.cpqd.com.br/routeflow/

493 days since Project Launch

1000s downloads!
Collaborations and community developments

- Web-based UI & Internet 2 HW pilot [C. Small, Indiana]
- Aggregated BGP Routing Service [C. Corrêa, Unirio]
- SNMP plugin [J. Stringer, Google]
- Optimal BGP best path reflection [R. Raszuk, NTT-MCL]
- Open Label Switched Router [OSRF; Google]
- OpenFlow v1.2 and v1.3 [w/ Ericsson]
- OpenFlow-enabled ROAD [EU/Brazil FIBRE Project]
Controller-Centric Hybrid Networking

- A migration path to roll out OpenFlow technology
- Not a revolution, but an evolution of current iBGP RRs to essentially eBGP Route Controllers
  - “BGP-free edge”: A cost-effective simplified edge for SW-driven innovations
Key Features

- Modular architecture
  - RF-Proxy
  - RF-Server
  - RF-Client

- Database layer
  - JSON-based IPC
  - Resilient core state
  - Programmer-friendly

- Multi-Controller support
  - NOX, POX, (Ryu)
  - Floodlight, Trema (planned)
Modes of operation

- From logical routers (akin VRFs) to single node abstractions over flexible virtual networks.
- New design choices on the distribution of the control nodes.
Research in scope and contribution

- Early work on Routing Control Platforms (RCP)
  - In operation at AT&T, considered a differentiator for "dynamic connectivity management".
- Research Question:
  - Re-examine the concept of RCP with the visibility (i.e., network-wide, multi-layer, flow and topology maps, full RIBs) and direct control capabilities (i.e., actual FIB installation, rich matching and instruction set) of the SDN abstraction set and the specifics of the OpenFlow choice
- RouteFlow glues virtualized IP routing stacks with OpenFlow
- RouteFlow acts as a new indirection layer for
  - routing protocol messages (e.g. BGP session terminates in servers)
  - RIB-(to-FIB)-to-OpenFlow transformations
Deployment

- 4 Virtual routers
- 10 Gig and 1 Gig connections
- 2 BGP connections to external networks
- Remote Controller
- New User Interface
Compare interfaces over the last 30 years

“PC” user interfaces

Network user interfaces

Demystifying Configuration Challenges and Tradoffs in Network Based ISP Services (Benson, Akella, Shaikh SIGCOMM 2011)

Source: Chris Small (Indiana)
RouteFlow User Interface

- How to make network administration:
  - Simpler to implement
  - More robust and consistent
  - Easier to manage

- Automation and Config Abstractions

- Can you build very different interfaces with SDN backends?
  E.g., type: http://netkarma.testlab.grnoc.iu.edu/rf/ or... http://goo.gl/T3Tqe

Source: Chris Small (Indiana)
Prototyped: Aggregated BGP routing service

- Single node abstraction of a domain-wide eBGP router
  - Think modern multi-chasis routing architectures with external route processors and OpenFlow switches acting as line cards
- Aggregation logic defined in the RF-Server
- NOX, MongoDB, LXC
Routing-centric use cases under research

- **Engineered path selection**
  - Think **Google WAN**, performance-based routing, etc.

- **Optimal best path reflection**
  - Per ingress/customer  [draft-ietf-idr-bgp-optimal-route-reflection-01]

- **Path protection with prefix independent convergence**
  - Hierarchical FIBs w/ OF 1.X Tables + LFA route-precomputation

- **Security**
  - Data plane blackholes and middlebox injections,
  - Secure Inter-domain routing ideas (crypto intense S*-BGP, etc..)

- **Simplifying customer multi-homing**
  - Easy to set and control cost/performance/policy-based routing

- **IPv6 migration**
  - Flow matching for service termination in v4-v6 migration solutions
Google Software Defined WAN Architecture
Fast convergence

Exploit OF 1.X group tables to store backup NHs per-prefix
Offline pre-computation of loop-free/converged alternate routes
- Use a “shadow” network to learn about future states

For every possible link failure:
- Force control plane failure events in the shadow network
- Let control plane converge
- Observe final state and store deltas
- (Rank failures according to “costs”)

When actual failure (state change happens)
--- Directly apply the pre-computed state changes (flow-mod deltas)
----- If combined with switch OAM: pre-install restoration state in group actions, triggered by the switch OAM (e.g BFD)
## Control Plane Distribution Options

<table>
<thead>
<tr>
<th></th>
<th>Vertically integrated (classic Router/Switch Model)</th>
<th>Decoupled (original OpenFlow model)</th>
<th>Hybrid (evolving model in ONF)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logically Centralized</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(“servers”)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fully distributed</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(“on box”)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Data Path jointly controlled by standard on-box control plane and centralized off-box controller

---

**Legend:**
- ![Data plane](image)
- ![Control plane function](image)

*Slide courtesy Frank Brockners*
Challenges

- **Centralized BGP**
  - Shown to scale well in modern CPU architectures
  - Centralized does not mean not distributed (but removal from edge)
- **Small OpenFlow table sizes**
  - Transient limitation?
  - Expose existing FIB data structures as an IP lookup OF table?
  - Smart RIB&FIB reduction (e.g., simple [draft-ietf-grow-simple-va-04]
  - HW/SW flow offloading (e.g. Fibium)
- **Limited OpenFlow processing in datapath**
  - Transient / Un-optimized implementations
- **High availability**
  - Previous ideas from distributed RCPs
  - Database-centric designs
  - Development in-progress of “BGP SHIM” for transparent eBGP redundancy
Conclusions

- RouteFlow is
  - a simple yet powerful (adaptable, inexpensive) routing architecture
  - a platform for real IP routing protocol experimentation
  - a tool for OpenFlow adoption via controller-centric hybrid networking

- Many open research questions and future work
  - OF 1.X, MPLS, OAM, GUI, policy languages, configuration mgm, etc.

- Opportunity for a community-driven development of competitive, deployable, open routing control solutions
Thank you!

Questions?

Christian Esteve Rothenberg, PhD
Diretoria de Redes Convergentes (DRC)
esteve@cpqd.com.br
Live DEMO

• Access:
  - http://go.cpqd.com.br/7API-demo

• Indiana University GUI demo:
  - http://goo.gl/T3Tqe
RouteFlow Platform research topics

- High availability
- Integration of OF v1.1, v1.2 and v1.3
- LDP / MPLS support towards open-source LSR
- Realizing the northbound SDN abstractions
  - Specification / Configuration
  - Network Information Base
  - Knowledge Information Base
- Troubleshooting, testing, debugging, ...
- ...