Software Defined Networking for access networks

ALIEN

$$\text{Richard G. Clegg} \begin{cases} \text{University College London} & t <= 1406851200 \\ \text{Imperial College} & t > 1406851200 \end{cases}$$

Talk to Coseners/MSN 2014

(Prepared using LaTeX and beamer.)

The ALIEN project

ALIEN – Expanding the availability of OpenFlow

Get OpenFlow working on new types of devices. Expand use of OpenFlow by making it useful for new platforms.

- Programmable platforms – NetFPGA and similar.
- Access networks – e.g. GEPON and DOCSIS.
- Optical devices – require OpenFlow extensions.

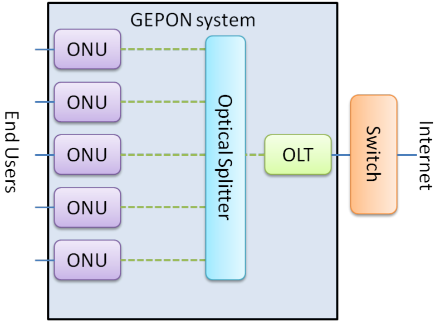
Advertising break

MoN13: Mathematics of Networks 13

Free meeting, practitioners who use applied mathematics to study networks. Good chance to learn about the wider field and to present your work to a wider audience.

- Imperial College 10th September 2014.
- Abstracts by 1st August 2014.
- All the fun of mathematics and networks, together!
- `http://www.monmeetings.org/meeting13`.

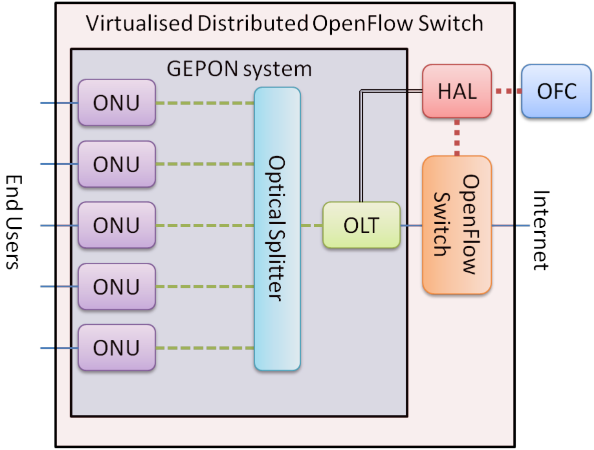
The GEPON (Gigabit Ethernet Passive Optical Network)



Getting OpenFlow on GEPON

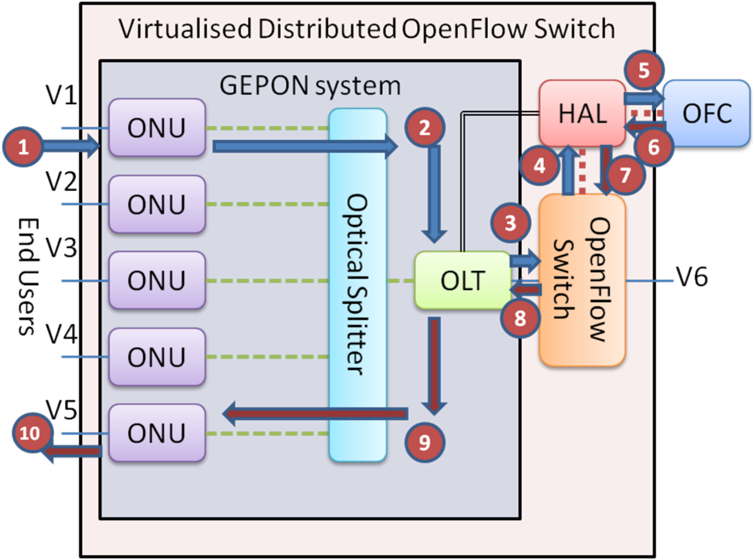
- Want to make whole system of switch, OLT, splitter ONU present as single massively distributed OF switch.
- Problem with making GEPON OpenFlow capable:
  - Proprietary device, no knowledge of chips or drivers.
  - Not intended as programmable.
  - ONU cheap low power consumer unit.
- Solution:
  - Make use of fact that OLT usually needs switch before it anyway.
  - Give OLT Open Flow capable front end switch (xdpd on NetFPGA).
  - OLT can switch to ONU on VLANs.
  - Use a mapping to VLANs in lower level open flow switch.
  - Controller sees only higher level abstract switch.

The GEPON with OpenFlow



- Note – approach is generic – could work for many access devices.
- Another partner is working on the generic approach of surrounding the non OF device with OF switches.

The packet's big adventure



Results

ROFL libraries

The Revised OpenFlow Library abstracts OpenFlow concepts as C++ objects and methods. Not your standard OFC northbound – targetted at data path implementations not controllers. `https://github.com/bisdn/rofl-core`

Unit tests

We pass them – lots of them. Oftest both a standard and substandard. `http://www.projectfloodlight.org/oftest/`

- Map vlans to interfaces – sounds easy, isn't.
- Lots of corner cases (flood packet to all interfaces, match on a physical output port).
- Some things still require "digging into the hardware" – modifying a port's status doesn't work with VLANs you need to send a command to the hardware.

Conclusions: Lessons learned

- Implementing the whole protocol is hard.
- OpenFlow commands in this approach group to:
  - Just work – no changes.
  - Work with tags – VLAN tags added/removed.
  - Just don't work – need hardware commands.
  - Will never work – VLAN tags when no QinQ available.
- ROFL is an excellent place to start if you need more than "just another OpenFlow northbound". https://github.com/bisdn/rofl-core.
- Provides sensible abstractions not just for messages to/from controller.
- OpenFlow is actually really fun to play with despite the pain.
- More details in EWSDN paper, preprint: http://www.richardclegg.org/access_sdn.
- Our code is available: https://github.com/richardclegg/xcpd