TRILL/SPB Working Group

RONALD VAN DER POL - SURFNET

LHCOPN and LHCONE joint meeting – Paris (FR)
17 June 2013

TRILL/SPB Working Group Objectives

Investigate alternatives for spanning tree protocol (STP)

First attempt of L2 service two years ago resulted in spanning loop problems

VRF was deployed as temporary solution

Several (innovation) working groups were started to work on "next generation" service
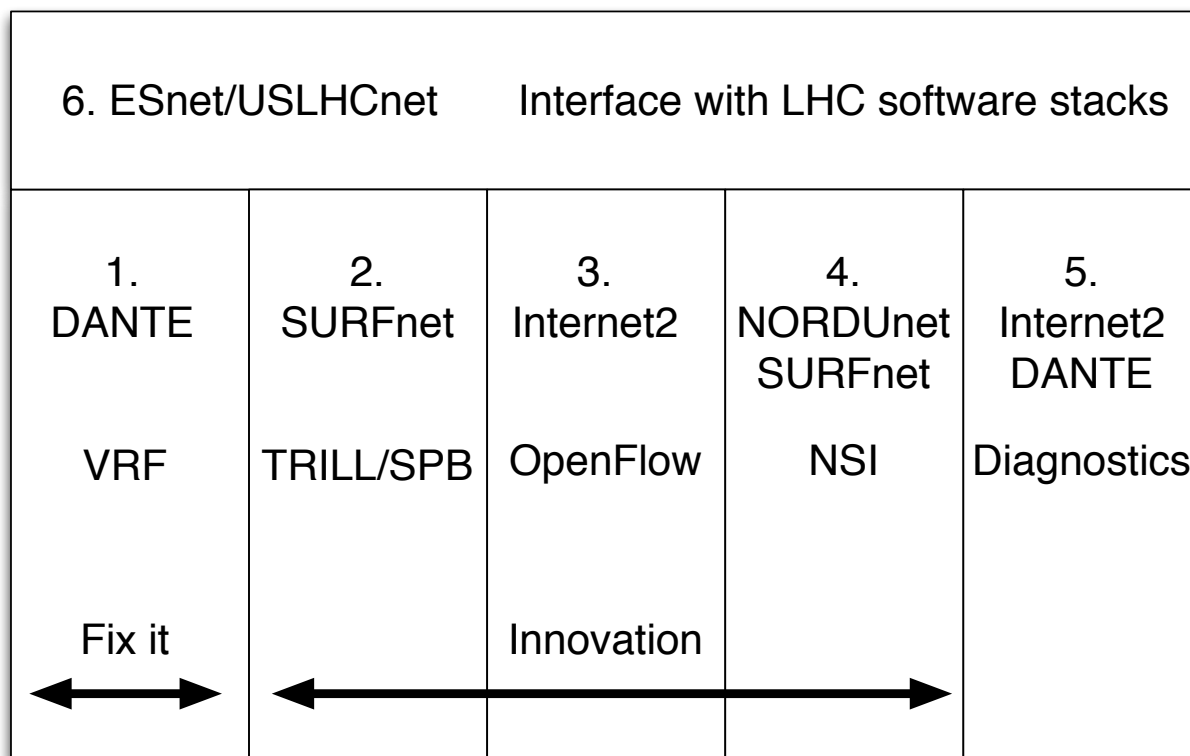
• TRILL/SPB

• OpenFlow

• NSI

TRILL/SPB WG was chartered to work on STP alternatives

Make better use of multiple paths in the infrastructure

There are multiple links between T1/T2 sites, multiple transatlantic links

Can we make better use of that capacity?

SURF NET

LHCONE Working Groups

6. ESnet/USLHCnet	Interface with LHC software stacks			
1. DANTE VRF Fix it	2. SURFnet TRILL/SPB	3. Internet2 OpenFlow Innovation	4. NORDUnet SURFnet NSI	5. Internet2 DANTE Diagnostics

SURF NET

Innovation Working Group Schedule

In the LHCONE meeting in Amsterdam in December 2011 we agreed:

Activity leaders report on what can be achieved in:
a) July 1, 2013
b) January 1, 2014

Late 2014: LHC restart

SURF NET

SPB/TRILL WG Proposed Schedule

Proposed schedule presented in January 2012:

1Q2012: document about 802.1aq and TRILL (done)

Rest of 2012: setup testbed with equipment that supports SPB/TRILL
Very few switches support SPB/TRILL
We decided to investigate if OpenFlow could be used to implement SPB/TRILL features
Testbed setup with iCAIR/Caltech/SURFnet

End of 2012: choose one of the protocols
Taken over by events; we have moved to OpenFlow
Provisioning of paths by OLiMPS is alternative for STP
OLiMPS maps application flows to forwarding paths: looks like a useful load balancing solution
MPTCP could also be part of the load balancing solution

1H2013: early deployment testbed (our works is not production ready)

SURF NET

Multipath Testbed

Partners: iCAIR, Caltech, SURFnet (and NLnet Labs, TU Delft, UCL)

Testbed between CERN, NetherLight and StarLight

Demos at GLIF (2012), SC12 (2012) and TNC (2013)

Not production ready:
Not enough operational experience yet with OpenFlow switches
(Too) fragmented OpenFlow controller community
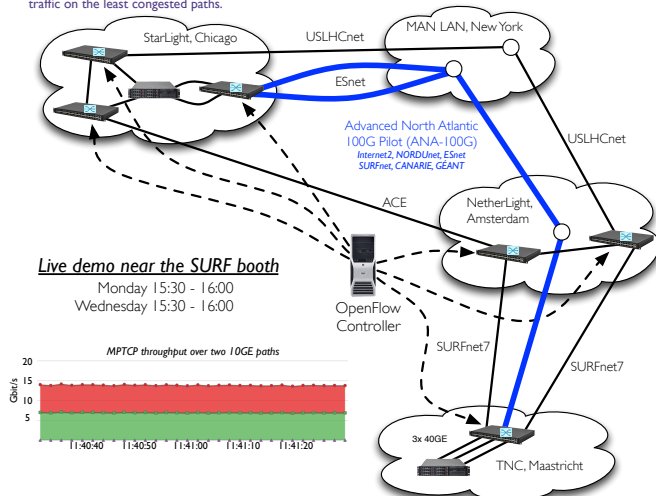No (large) OpenFlow WAN community yet
OLiMPS developed by 1 person (Michael)

SURF NET

100G Multipath Demo at TNC 2013

https://tnc2013.terena.org/core/poster/9

100G Multipath Demo at TNC 2013

https://tnc2013.terena.org/core/presentation/13
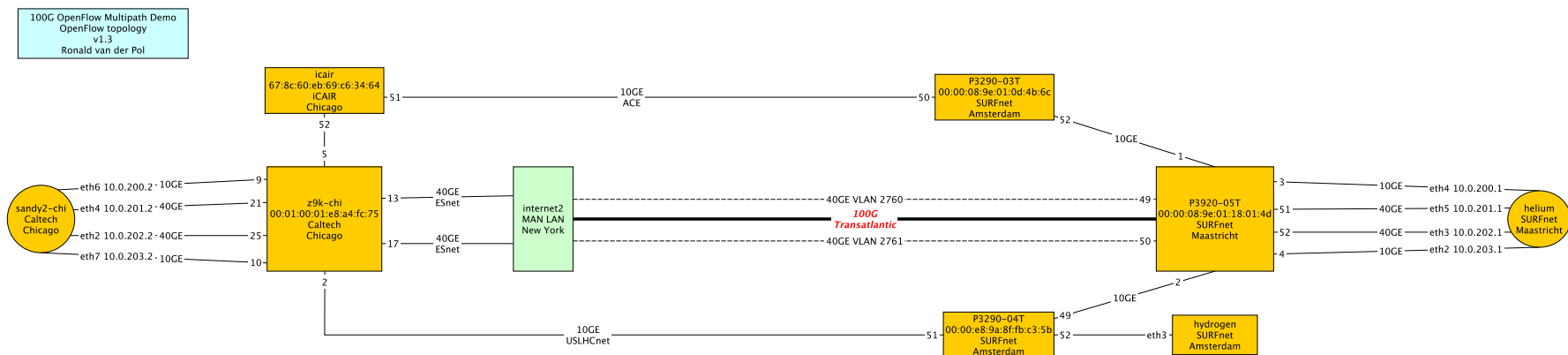
SPB/TRILL Working Group Status

Investigate alternatives for STP (✔)
OpenFlow forwarding entries are a good alternative for spanning tree

Make better use of multiple paths in the infrastructure (✔)
OpenFlow and MPTCP are useful tools to make better use of the available network capacity

Production Ready (✖)

SURF NET

SPB/TRILL WG Challenges

How can we move to a production level service?

Start small and simple

Start with single domain? Transatlantic load balancing?

Include fallback scenario

How can we create an OpenFlow development community for the research networks needs?

When we take OpenFlow seriously, we need a broad, long term stable development community

No niche, adopt and cooperate with other development communities

We do not know how to do multi-domain OpenFlow

LHCONE is a multi-continental network, we cannot use 1 controller for that

There is no production ready virtualisation solution yet

Flowvisor does network segmentation, not virtualisation

Future development of FlowVisor is uncertain

Encapsulation comes closest to network virtualisation similar to computing MMU

SURF NET

SPB/TRILL WG Possible Actions

Merge with OpenFlow Working Group?
TRILL/SPB WG activities have fully moved to OpenFlow solutions
It seems logical to merge the two groups

Identify where OpenFlow can play a role in LHCONE
The demos gave us insight of what is possible
Traffic engineering with OpenFlow seems useful
Maybe start on the transatlantic load balancing problem?

Work on research networks OpenFlow community
Which research networks are interested?
Universities?
ONLAB?
Open Daylight?

SURF NET

Ronald van der Pol
Ronald.vanderPol@SURFnet.nl
www.surf.net

WHAT **SURF** CAN DO