

Experiences with SDN & OpenDaylight

OPENDAYLIGHT MINI-SUMMIT DÜSSELDORF, OCTOBER 2014



SDN
OpenFlow
WORLD
CONGRESS

Ronald van der Pol <Ronald.vanderPol@SURFnet.nl>



Overview

- **SURFnet, academic and research networking and the changing environment**
- **Building an SDN prototype application with OpenDaylight**
- **What we learned**

SURF eScience Services for Research and Higher Education



SURFnet Dutch National Research Network

Nationwide dark fiber infrastructure

DWDM & Carrier Ethernet

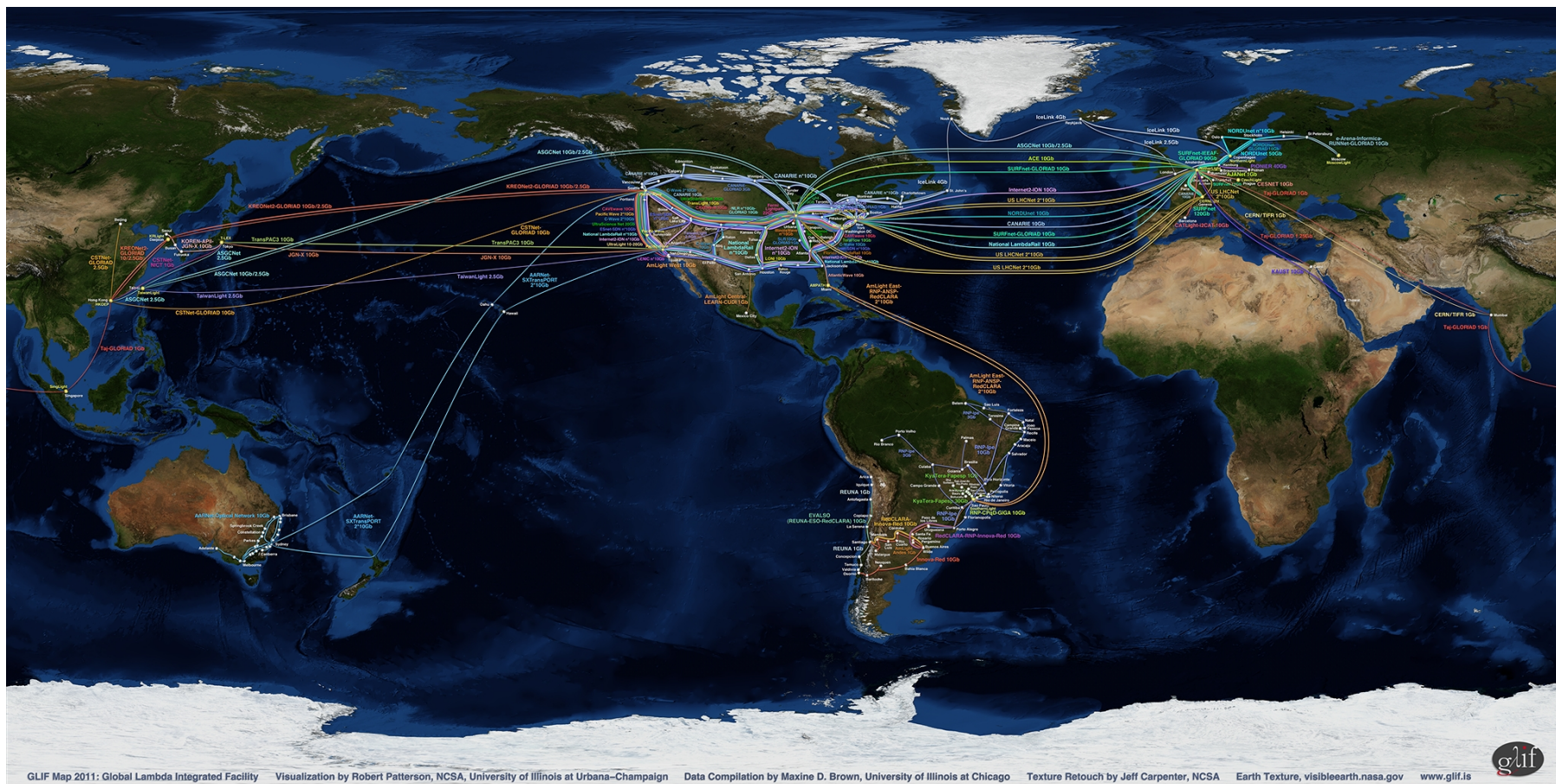
Around 165 connected institutions (universities, university medical centres, research institutes)

IPv4/IPv6 unicast/multicast + (dynamic) high speed P2P circuits

Federated ID, collaboration, security, wireless services & innovation

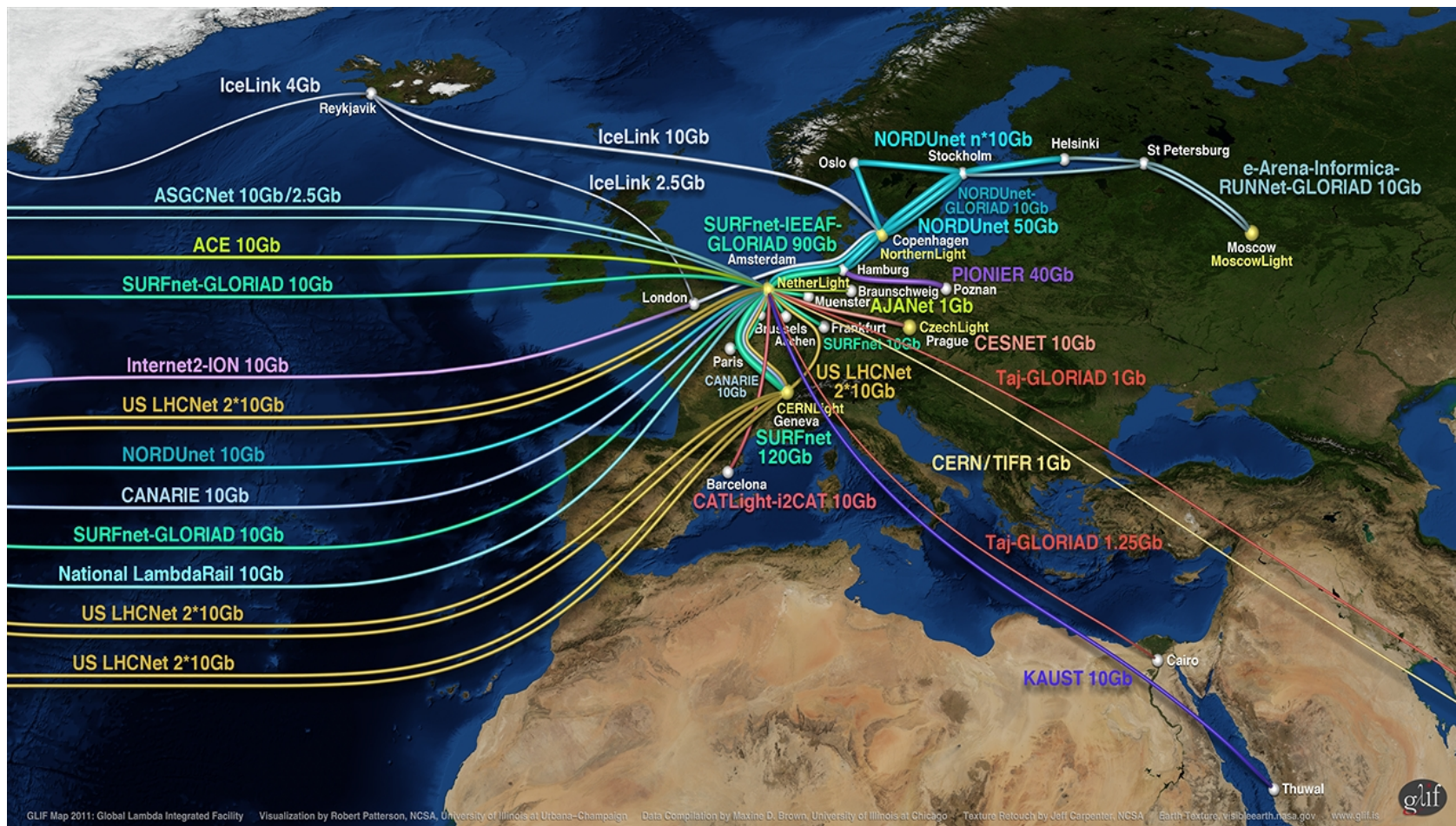


GLIF Worldwide Research Infrastructure



GLIF Map 2011: Global Lambda Integrated Facility Visualization by Robert Patterson, NCSA, University of Illinois at Urbana-Champaign Data Compilation by Maxine D. Brown, University of Illinois at Chicago Texture Retouch by Jeff Carpenter, NCSA Earth Texture, visibleearth.nasa.gov www.glif.io

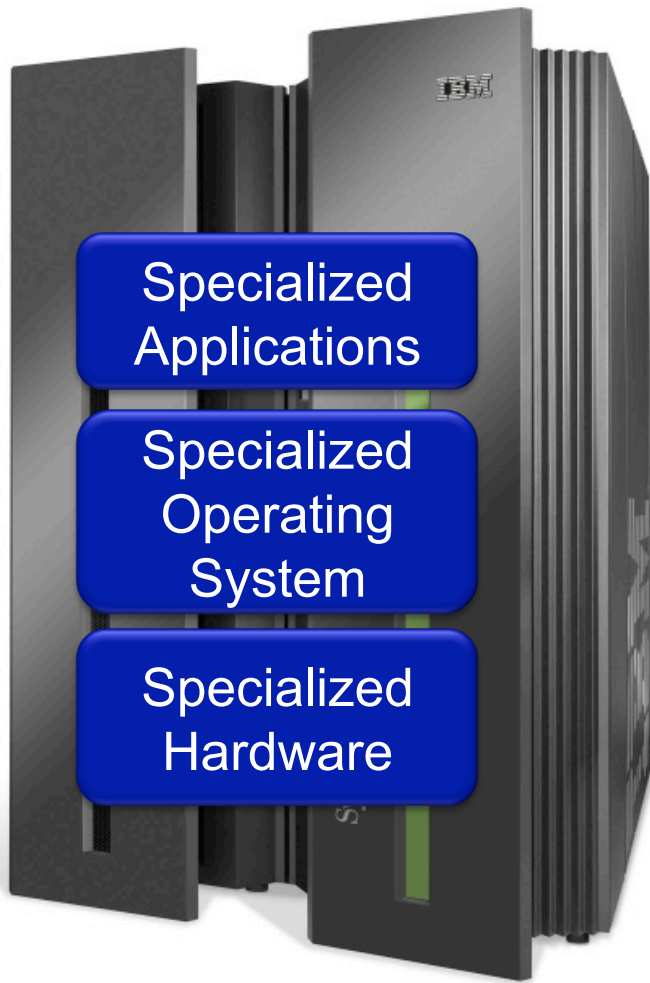
GLIF European Part



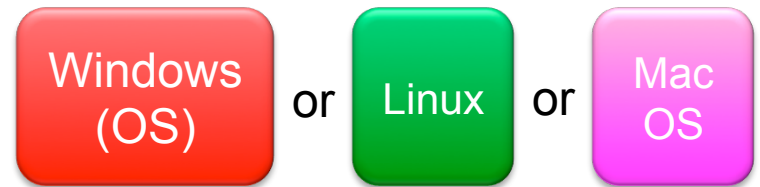
Changing Environment: SDN

- **Network innovation slow because we are dependent on vendors and their roadmaps.**
- **Vendors decide which features will be implemented and when.**
- **They offer mainstream features that are most requested.**
- **However, research networks must be leading in new technology.**
- **SDN, open APIs and moving features from routers and switches to the control plane offers NRENs the opportunity to be in control of innovation (again).**

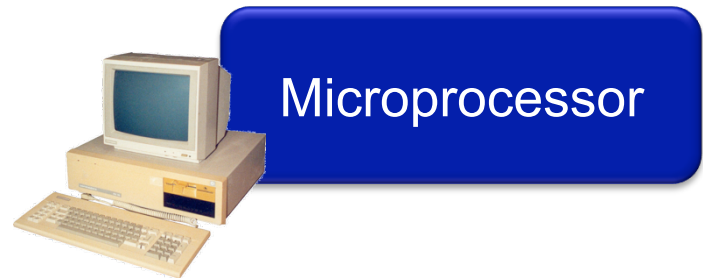
(slide by Nick McKeown, Stanford University)



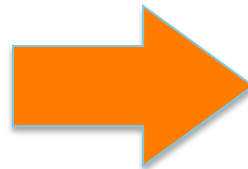
— Open Interface —



— Open Interface —

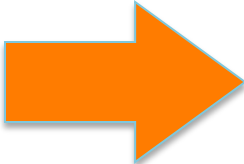


Vertically integrated
Closed, proprietary
Slow innovation
Small industry



Horizontal
Open interfaces
Rapid innovation
Huge industry

(slide by Nick McKeown, Stanford University)



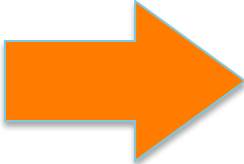
Open Interface



Open Interface

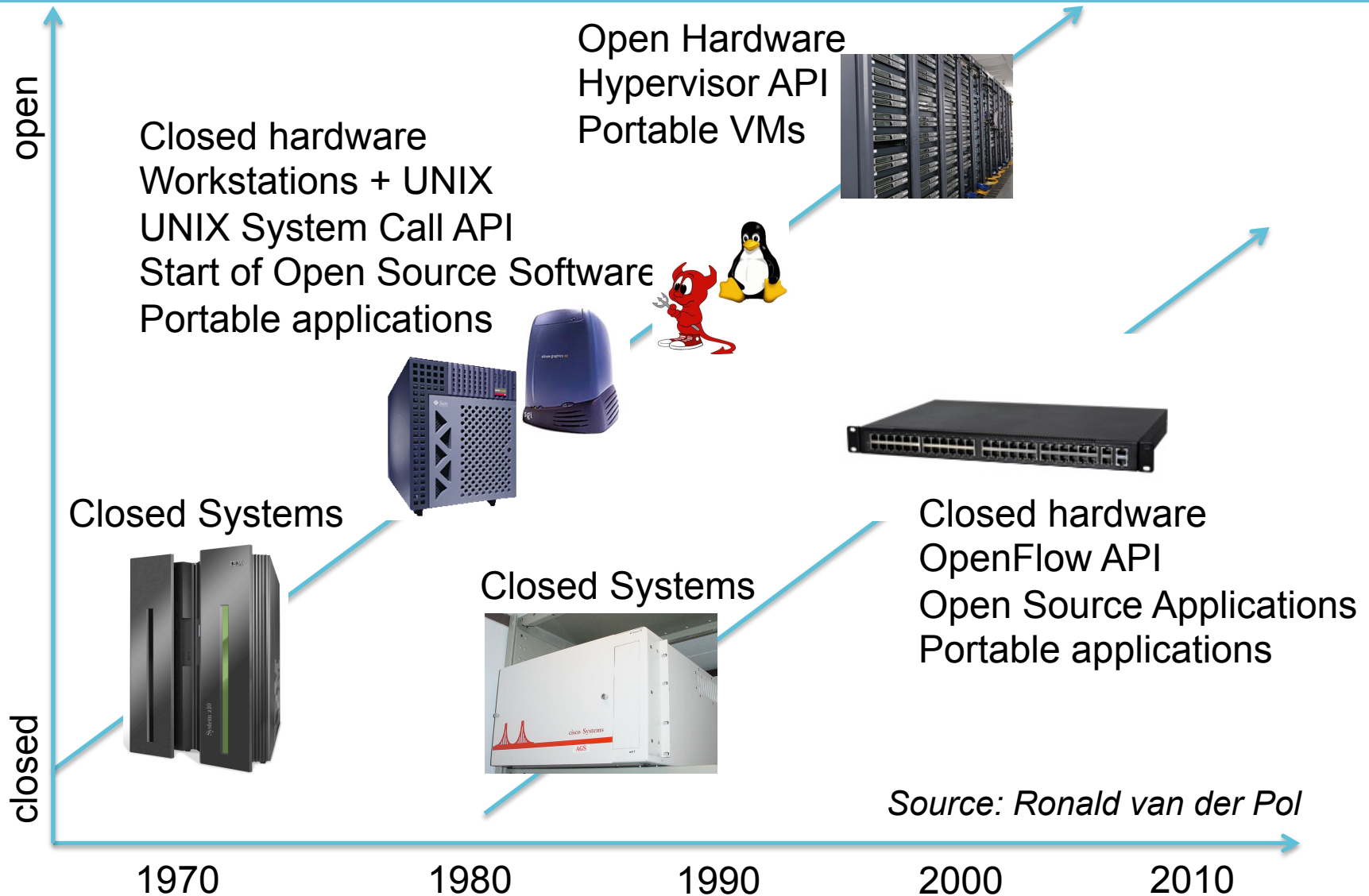


Vertically integrated
Closed, proprietary
Slow innovation



Horizontal
Open interfaces
Rapid innovation

Computing vs Networking

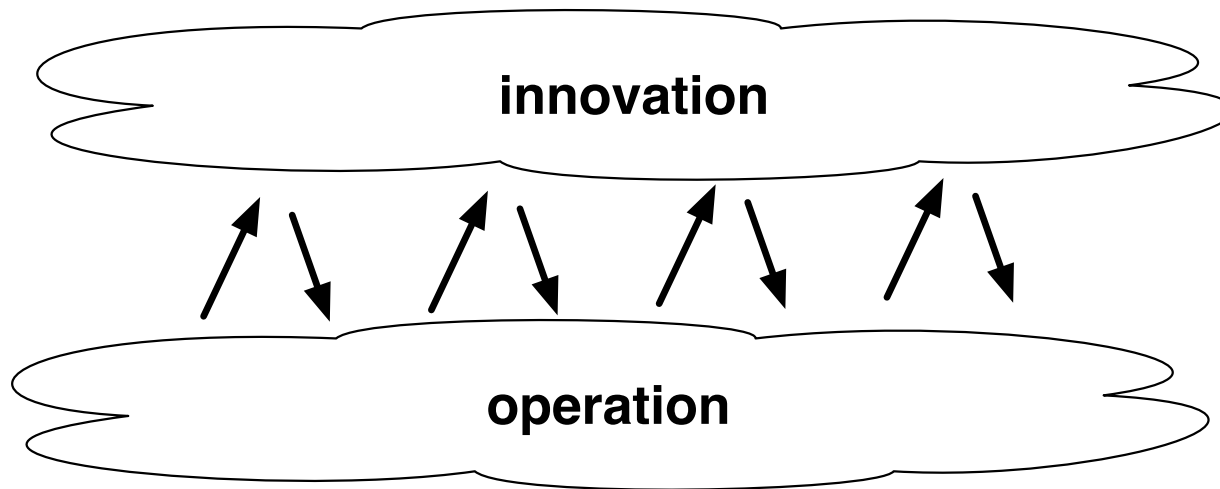
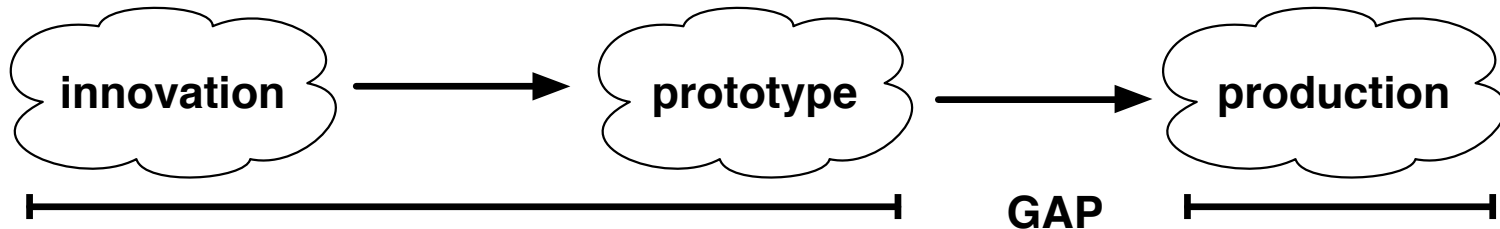


Source: Ronald van der Pol

Network Innovation at Research Networks

- **With SDN and Open APIs we can work on network innovation again by working on the control plane and networks applications.**
- **Cooperate with other research networks and academia.**
- **Join large communities in these areas, like OpenDaylight. Work with vendors and other users and service providers.**
- **But culture change needed.**
 - Few or no software developers at NRENs.
 - Most are CLI and protocol thinkers, not abstraction and API thinkers.
 - Innovation projects usually big isolated projects.

Culture and Organisational Changes



SURFnet SDN Goals for 2014

- **Build nationwide SDN testbed with hardware OpenFlow switches and OpenStack mini clouds for NFV**
- **Look at all operational aspects of such a network (OAM, monitoring, resilience, centralised versus distributed functions)**
- **Write a “real” SDN application to get experience with network programming (what does it take?, which frameworks do exist?, which abstractions are useful? etc)**
- **How mature is SDN?**

SURFnet Nationwide OpenFlow Testbed

- 6 Pica8 P5101 Trident II switches
- Looped multi-stage topology
- Mini OpenStack cloud at each site for NFV experiments
- OpenDaylight controlled
- Best way to slice to be investigated



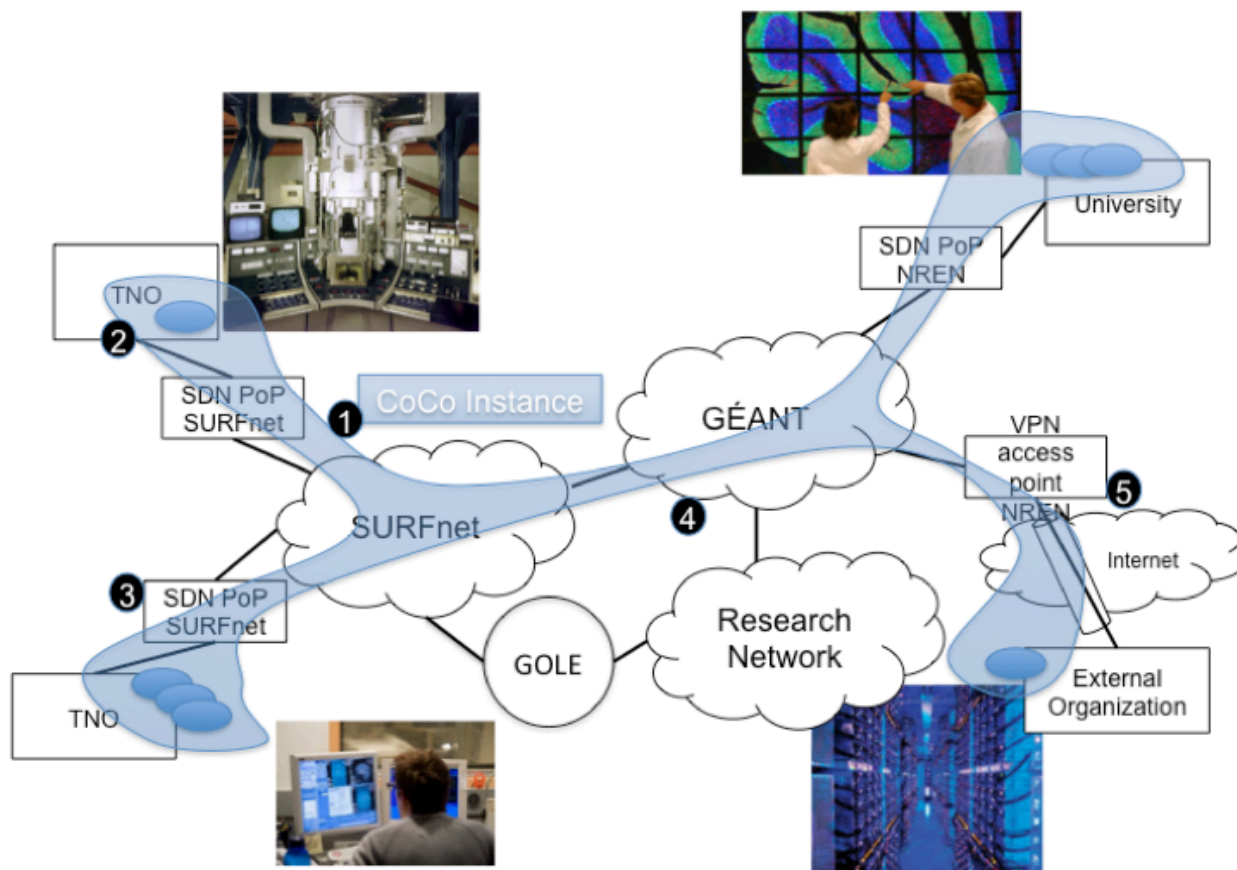
Write a “Real” Network Application

- **EU funded GN3plus Open Call Project (CoCo).**
- **<http://www.geant.net/opencall/SDN/Pages/CoCo.aspx>**
- **October 2013 – March 2015 (18 months).**
- **Budget Eur 216K; 16.4 person months.**
- **Partners: SURFnet (NL) & TNO (NL) – 50/50 split in effort.**
- **Five work packages:**
 - WP1: use cases & market demand
 - WP2: architecture, design & development
 - WP3: experimental validation
 - WP4: dissemination
 - WP5: project management

Community Connection (CoCo) Prototype

- **Goal of CoCo prototype:**
 - On-demand multi-domain, multipoint private L2/L3 network prototype.
 - Intended users: closed (eScience) community groups.
 - CoCo instances interconnect users, compute, storage, instruments, etc.
 - Each closed eScience community group can easily setup their own private CoCo instance via web portal without the help of network engineers.
- **Based on OpenFlow & OpenDaylight.**

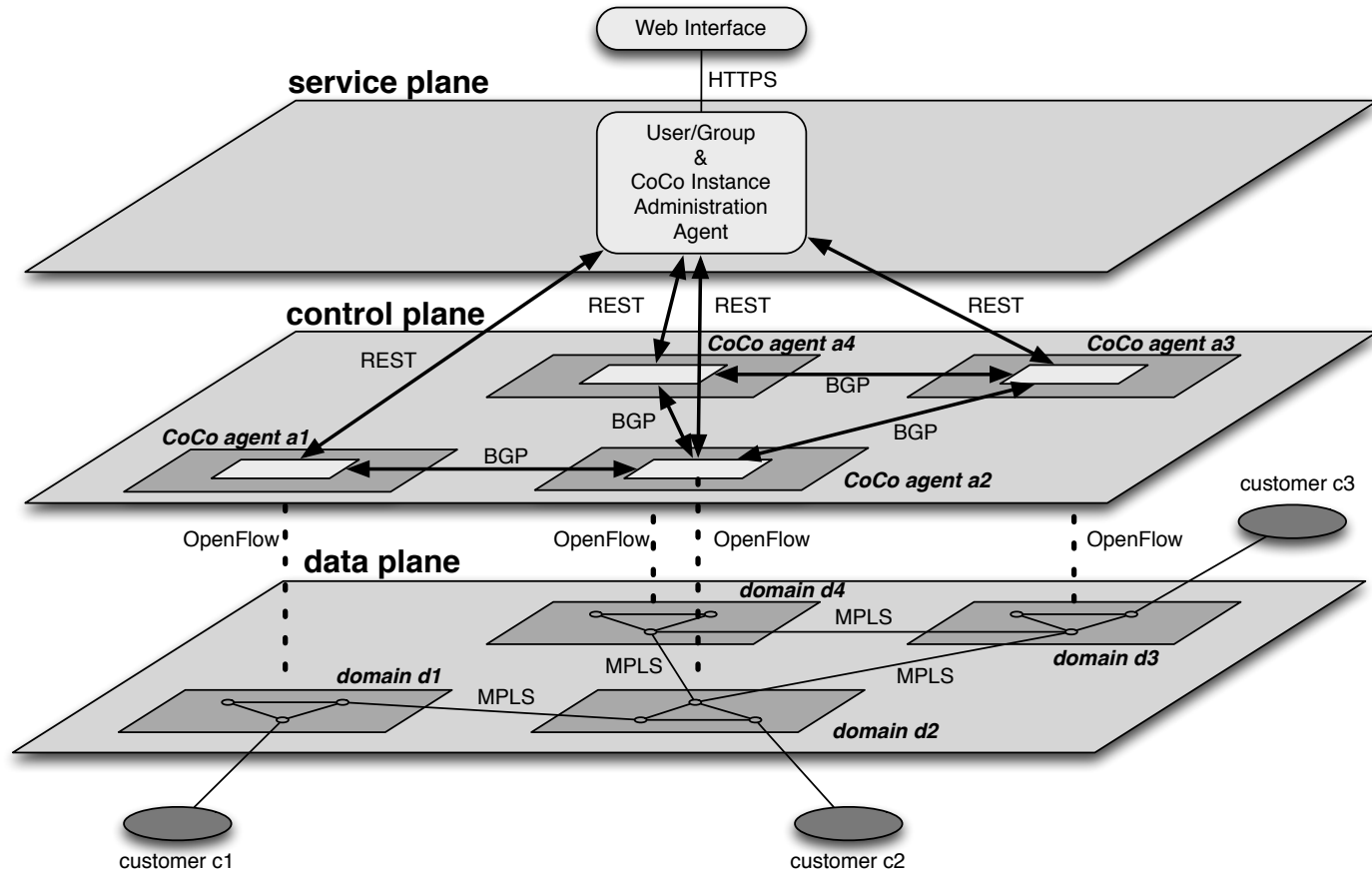
CoCo Instance



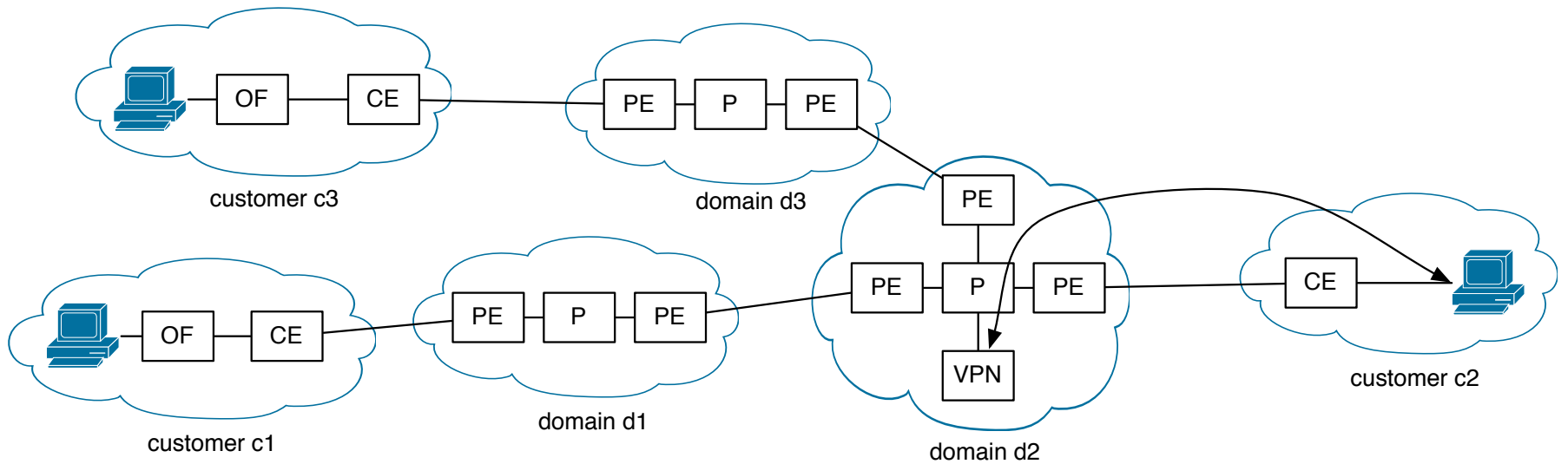
CoCo Architecture Choices

- **Use as much existing code, frameworks and protocols as possible.**
- **Use MPLS labels for aggregation and forwarding in the core.**
- **Centralised agent chooses label, no label swapping within domain, no label distribution protocol needed on the data plane.**
- **Use BGP messages to exchange information between domains.**
- **Use much on the BGP MPLS VPN (RFC 4364) architecture.**
- **BGP only used in the control plane. BGP not used for RIB to FIB.**

CoCo Control, Data and Service Planes



MPLS Forwarding



CoCo Data Plane

- **Data plane forwarding based on MPLS labels**
 - Outer MPLS label used to forward to destination PE switch.
 - Inner MPLS label identifies CoCo instance.
- **MPLS encapsulation and decapsulation done at PE.**
- **At PE the customer traffic is aggregated onto MPLS paths.**

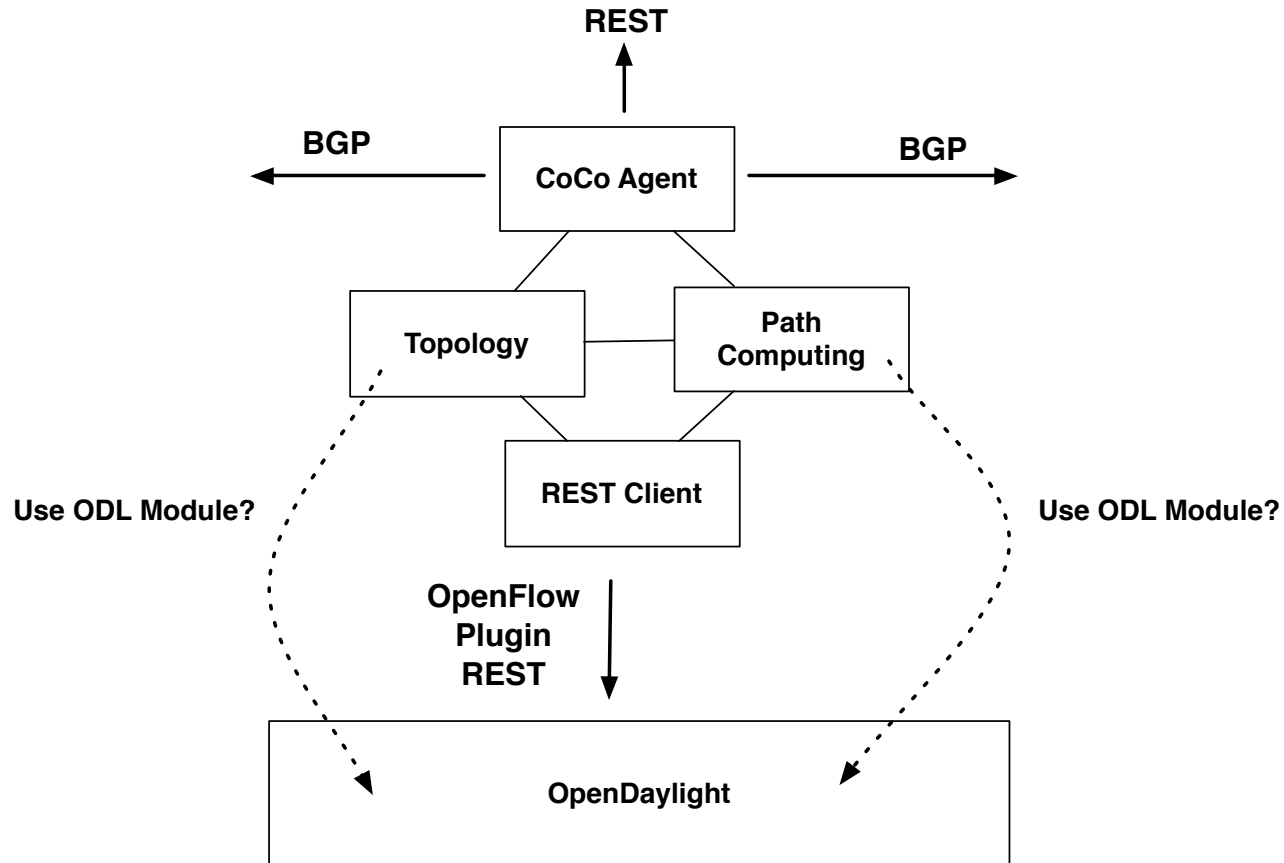
CoCo Control Plane

- **Control plane consists of federated CoCo agents.**
- **Each domain runs its own CoCo agent.**
- **Tasks CoCo agent:**
 - inserts MPLS forwarding rules in the core.
 - Inserts MPLS encap/decap rules in the PE switches.
 - Exchanges information with neighbours via BGP peering model
- **Information exchanged between CoCo agents:**
 - Customer IP prefixes.
 - MPLS label used for CoCo instance.
 - Information about end points participating in CoCo (for web portal).

Project Progress

- **January 2014: workshop with Dutch eScientists to define uses cases**
- **June 2014: agreement on architecture**
- **September 2014: picked use case (VPNs to interconnect genome sequencers, scientists, storage and compute)**
- **October 2014: first single domain prototype based on OpenDaylight**

First prototype setup



What have we learned?

1. Limitations in the OpenFlow protocol.

- We wanted to have a CE – PE service port with QinQ.
- Campus/enterprise would use 1 dedicated VLAN for the CoCo service. The outer VLAN tag.
- The inner VLAN tag would be used to map to a VPN instance.
- But, OpenFlow does not have a feature to match on inner tags.

What have we learned?

2. Limitations in hardware.

- Workaround for the previous problem is multiple tables and apply-action.
- But the Trident II (probably) does not support this.
- Some ASICs do not support MPLS (data centre vs transport ASICs).

What have we learned?

3. Bugs and unimplemented features.

- Mininet does not have full MPLS support yet.

Experiences with OpenDaylight

- **Lots of progress in one year!**
- **We needed OF 1.3, and it took some time to setup an environment with mininet and Hydrogen that worked with OF 1.3.**
- **“Old” and “new” OpenFlow plugins and finding the correct documentation.**
- **Documentation (up to date!) was hard to find, improved with Helium.**
- **But very good support on mailing lists and IRC!**
- **Much has improved in Helium!**
- **OpenDaylight has a very nice architecture, but also a large learning curve.**

Ronal van der Pol
Ronald.vanderPol@SURFnet.nl



WHAT **SURF** CAN DO