MPLS 2010
13TH ANNUAL CONFERENCE

MPLS Openflow and the Split Router Architecture: A Research Approach

Howard Green, James Kempf, Sonny Thorelli, Attila Takacs

Ericsson Research
howard.green@ericsson.com

www.mpls2010.com

Ericsson Research Silicon Valley

Part of our global research organization

Ericsson Research

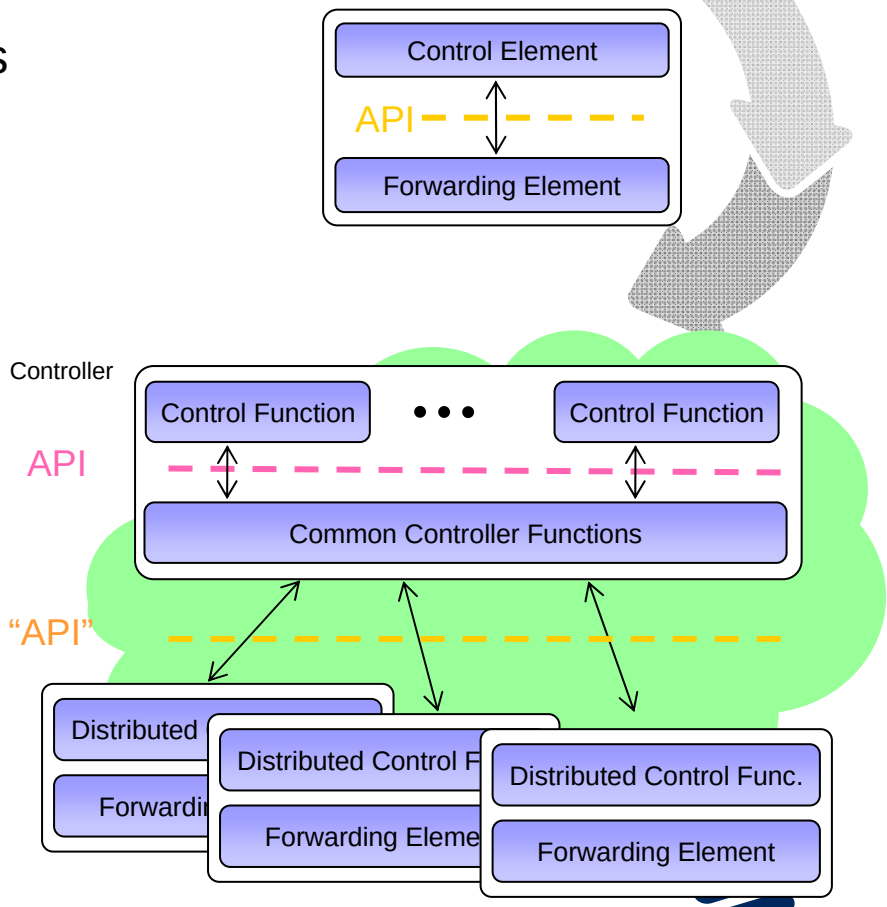
- ~600 people
- GSM, 3G and LTE technologies were invented here
- Delivers concepts and pre-commercial prototypes
- Files over 50 % of all Ericsson patents

ER Silicon Valley

- Packet Networking
- Open Application environment
- Radio Access Tech & Signal processing

Research trend in Routing & Transport

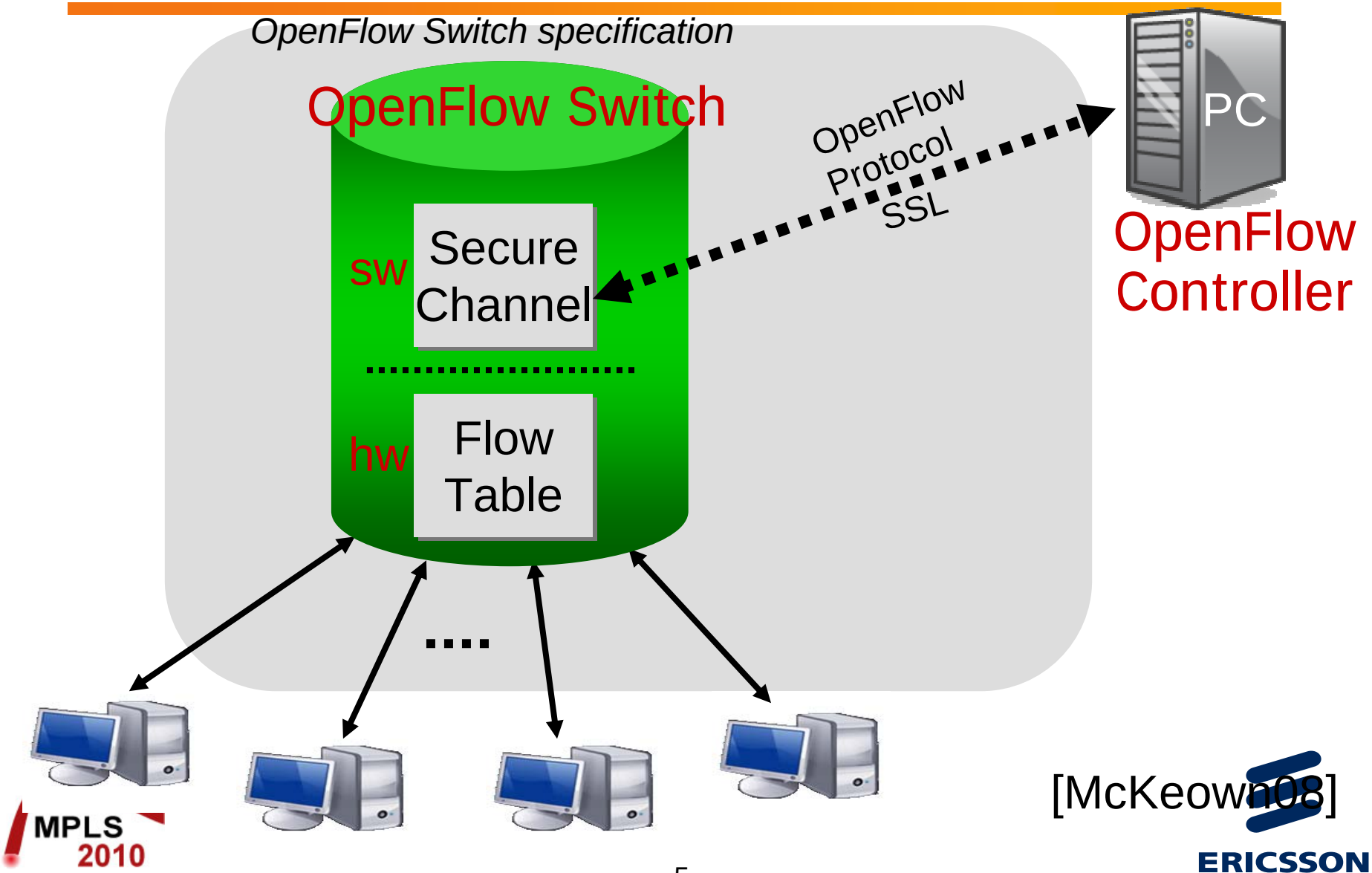
- Splitting the monolithic router
  - HW and SW tightly integrated with proprietary interfaces → box products
  - Difficult to innovate
  - Long lead time for features
- Moving intelligence out of the box
  - Network wide innovation still difficult
  - Define an open "API" between network and control functions
  - **OpenFlow and Centralized Network Controller**
    - Simple forwarding abstraction
    - Control functions run as "network applications" on centralized controller



Openflow-academic motivation

- **”Openflow: Enabling innovation in campus networks”** Mckeown, Anderson, Balakrishnan, Parulkar, Peterson, Rexford, Shenker, Turner; Sigcomm 2008
- Experiments at scale with new protocols
- Mixing experimental and normal traffic
- Intention to deploy at many large campuses
  - Currently underway
- Basis for experimental work in National Science Foundation GENI program
- http://www.openflowswitch.org

Openflow-what it is



Openflow now

- Openflow 1.0 delivered as basis of GENI activity end 2009
  - Open source controller platform NOX
  - Research switches from HP, NEC, Quanta
- Very limited functionality: IPv4, Ethernet, simple VLANs

OpenFlow Flow Table Entry

Rule:Flow match on 10-tuple; wildcards

In Port	VLAN ID	Ethernet			IPv4			Tranport	
		SA	DA	Type	SA	DA	Proto	Src	Dst

Action: forward to port, send to controller, rewrite header fields

Stats: Packet and byte counters

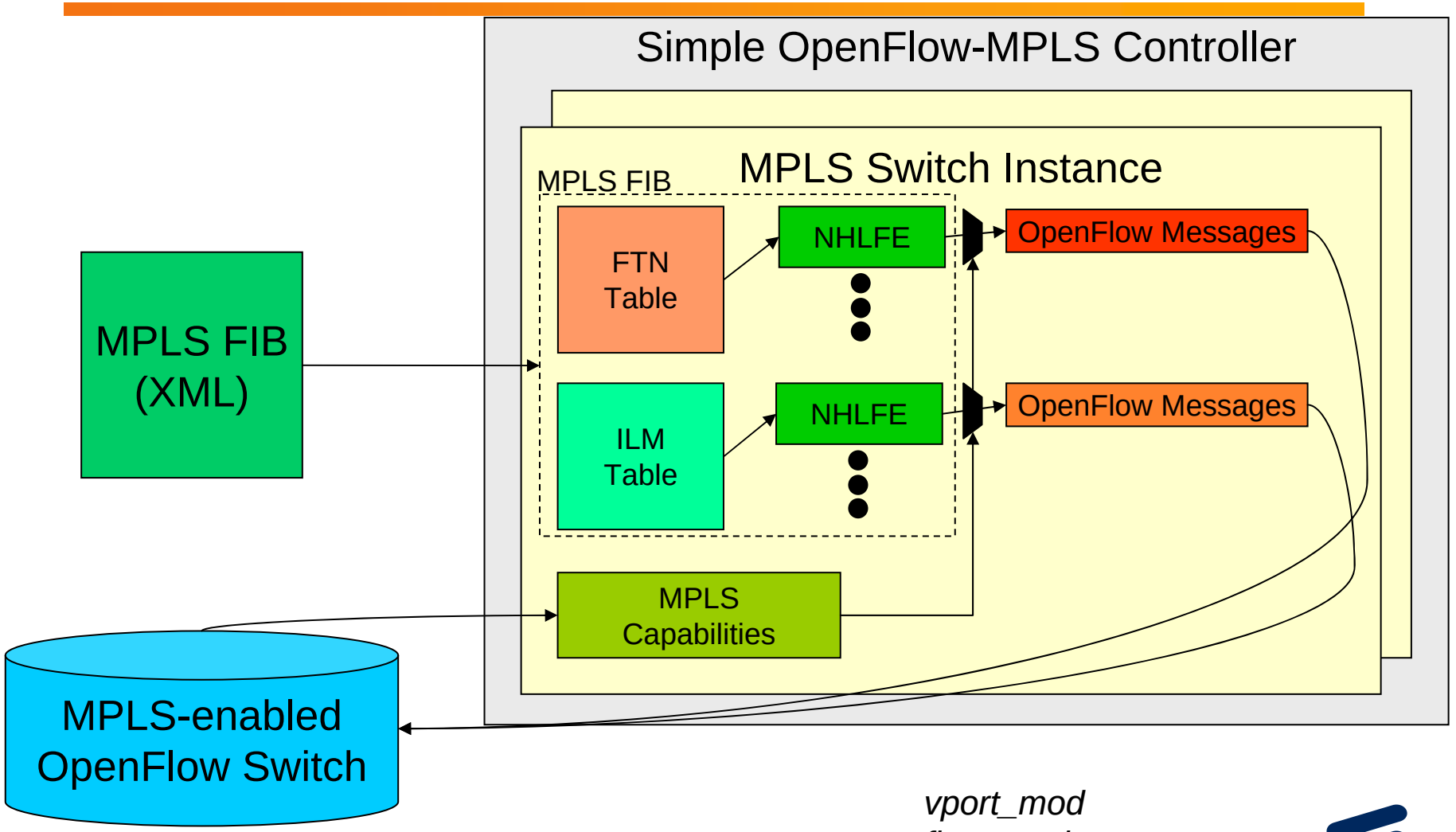
Openflow-who and why?

- Stanford Cleanslate project
  - Google, Deutsche Telekom, Cisco, Juniper, Ericsson, NEC, Xilinx, Docomo, Ericsson
- Other participants
  - HP, Nicira …..
- Current industrial interest from the data center
  - Scalable architectures for “virtual switching”
  - Support for “multi-tenancy” virtualisation
  - Separation of control software platform on standard servers
  - Use of high volume switching elements to build Clos networks
    - multiple paths, flow steering
  - Common management of many switches as “single fabric”
- Related work: IETF FORCES
  - Model and protocol for control/forwarding separation

Experiment: Openflow FOR MPLS

- Started as summer intern project
  - thanks to Mart Haitjema (WUSTL), Peyman Kazemian (Stanford)
- Experimental extensions to Openflow supporting MPLS
- Implement switch functions in software and hardware
  - Stanford reference Ethernet software switch
  - NetFPGA hardware platform
- Notes and links to opensource code at http://www.openflowswitch.org/wk/index.php/OpenFlowMPLS

Openflow MPLS Experiment



vport_mod
flow_mod
…

OpenFlow Modifications

Virtual Ports

Handle protocol/port specific actions with a protocol-independent interface

May be nested

Virtual port table: Virtual port number instead of 10-tuple: Action List

Port No	Parent Port	Actions	Stats

Matching

Extend 10-tuple to include top 2 labels in label stack

Actions

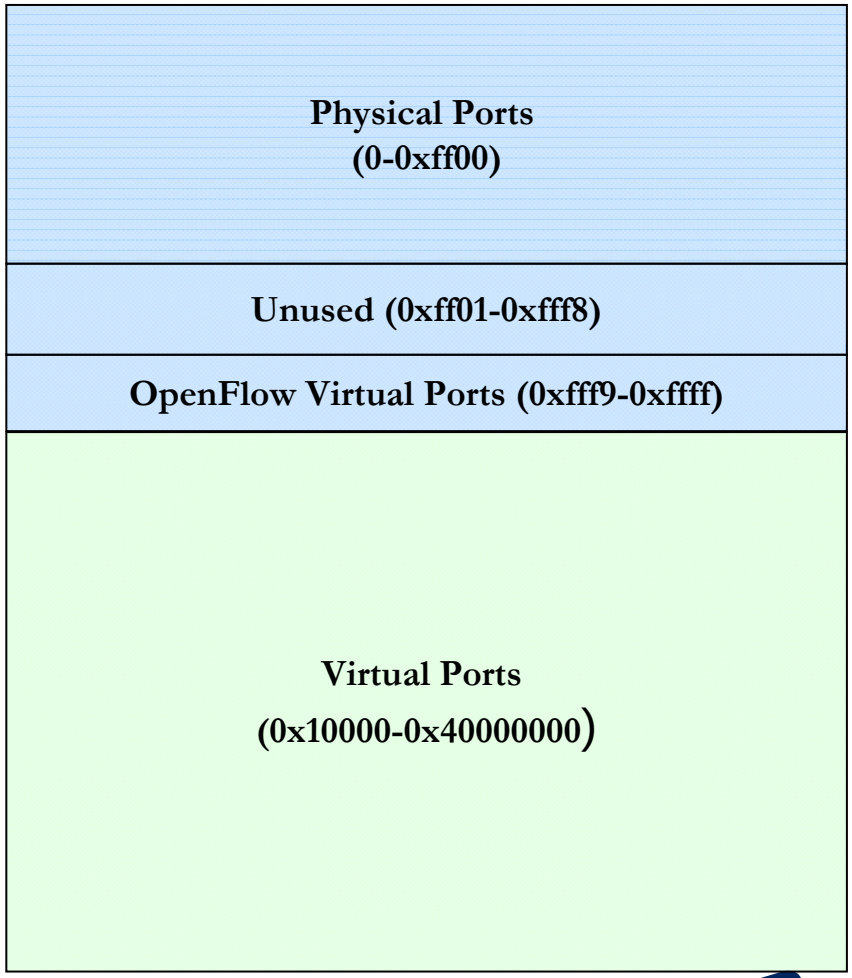
Send packet to virtual port to perform protocol specific actions (i.e. MPLS pop, push, swap)

Drop TTL expired packets & add mpls_ttl_exp stat

In Port	VLAN ID	MPLS		Ethernet			IP			Tranport	
		Label 1	Label 2	SA	DA	Type	SA	DA	Proto	Src	Dst

Virtual Port Table and Actions

- *push_mpls* action
  - Push a 32-bit MPLS label onto top of label stack
  - Copies TTL and Exp bits from IP header if required
  - Parameters:
    - 20-bit label to push
    - 8-bit ttl value (optional)
    - 3-bit exp value (optional)
    - Flags (to indicate special behavior)
- *pop_mpls* action
  - Pop the top 32-bit MPLS label off the label stack
  - Copies TTL and Exp bits to IP header if required
  - Parameters
    - Eth_type (if popping last label)
    - Flags (to indicate special behavior)



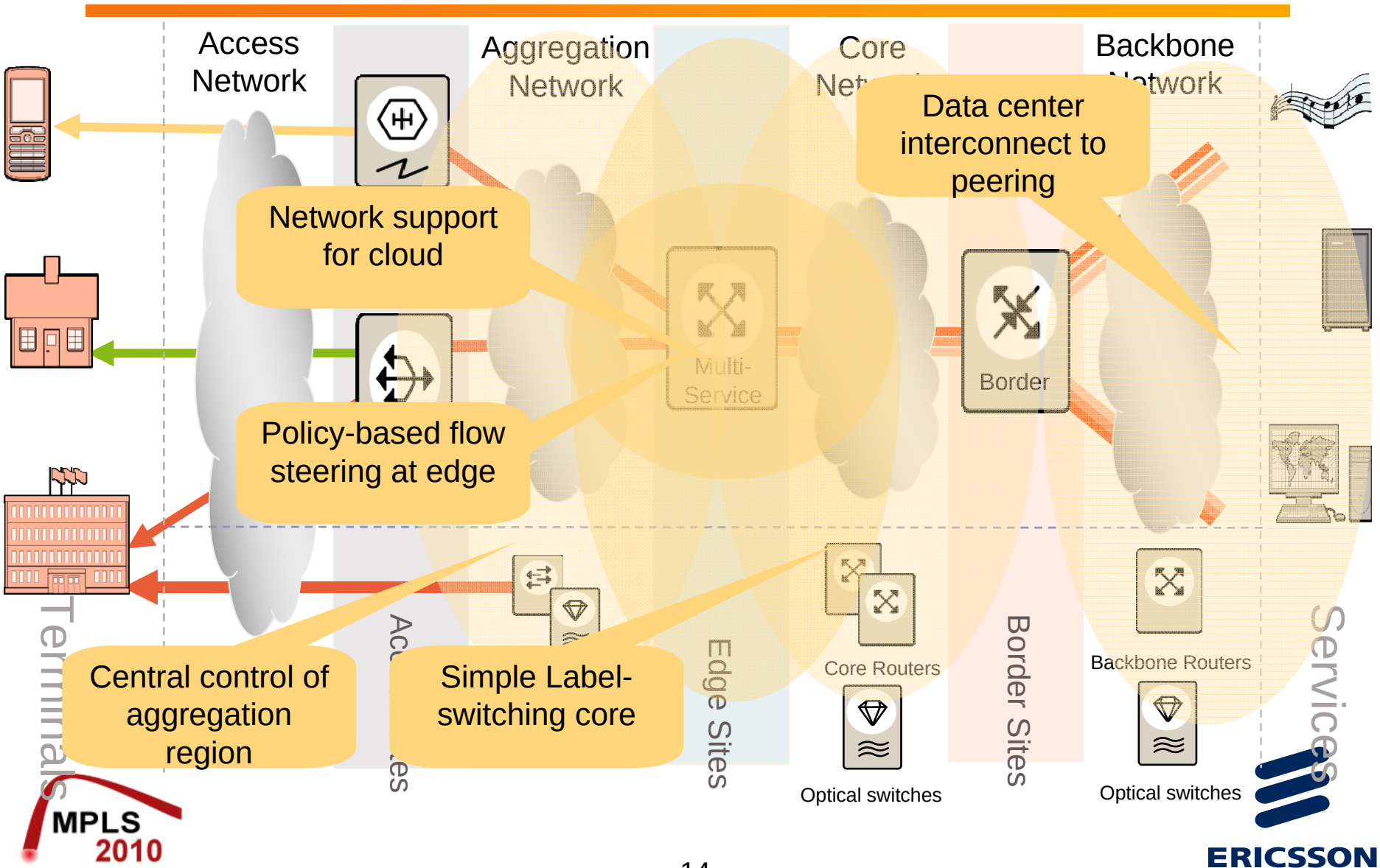
Next Step: OpenFlow V1.1 Specification

- Key additions to OpenFlow V1.0 for “large scale” usage
  - Targetted for late 2010
- Introduction of multiple tables
  - The current Flow Table model does not scale well
  - A topology of n untyped tables (matches to full n-tuple)
  - Introduce “meta-data” as further matching element
- Introducing multi-path support
  - Revises the current port model by adding “logical ports” ~ virtual ports and groups
- Tags and tunnels (includes MPLS support)

Why is this interesting?

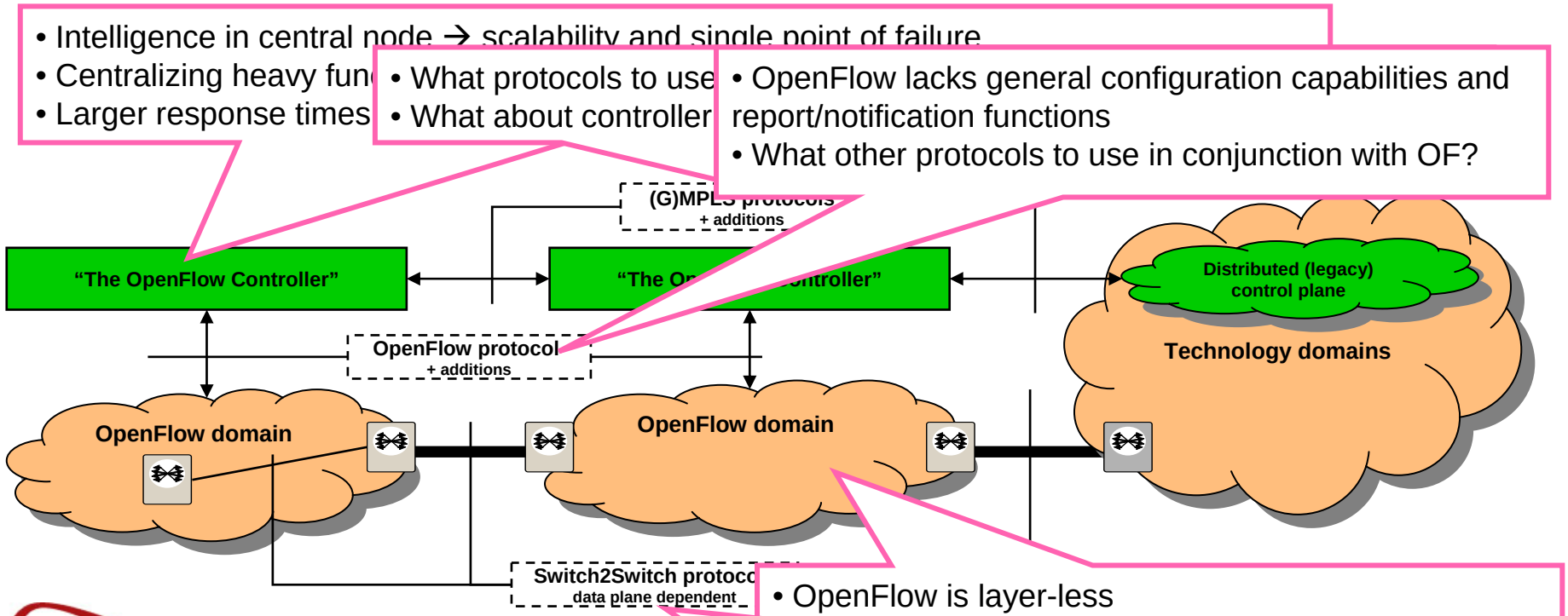
- High volume low cost technology from the data center
  - Hundreds of thousands of nodes connected at 10Gb/s
- "Split router"
  - Could the router market evolve to
    - Distributed control software platform in standard servers
    - Standardised switching platforms controlled over an open i/f
- What are the key things that need to be added to make "data center" solutions suitable for carriers?
- Currently a long list
  - Resilience, OAM, flow scale, MPLS, IPv6,….
- But this is why we do research!

Split router Use Cases



SPARC Project

- European research project
  - DT, Ericsson, IBBT, Acreo, EICT
  - 2010 July – 2012 July
- First studied use-case: Central control of aggregation...



MPLS OPENFLOW Controller Architecture

(Current Work)