Lecture Presentation

Autonomic Fail-over for Software-Defined Container Computer Network

Chien-Yung Lee, Yu-Wei Lee, Cheng-Chun Tu, Pai-Wei Wang, Yu-Cheng Wang, Chih-Yu Lin , and Tzi-cker Chiueh

Agenda

- Introduction
- Terminologies
- What is SDN?
- Peregrine Architecture
- Working of Peregrine
- Fault Tolerance Support
- Relation to course work
- Performance Evaluation
- Conclusion

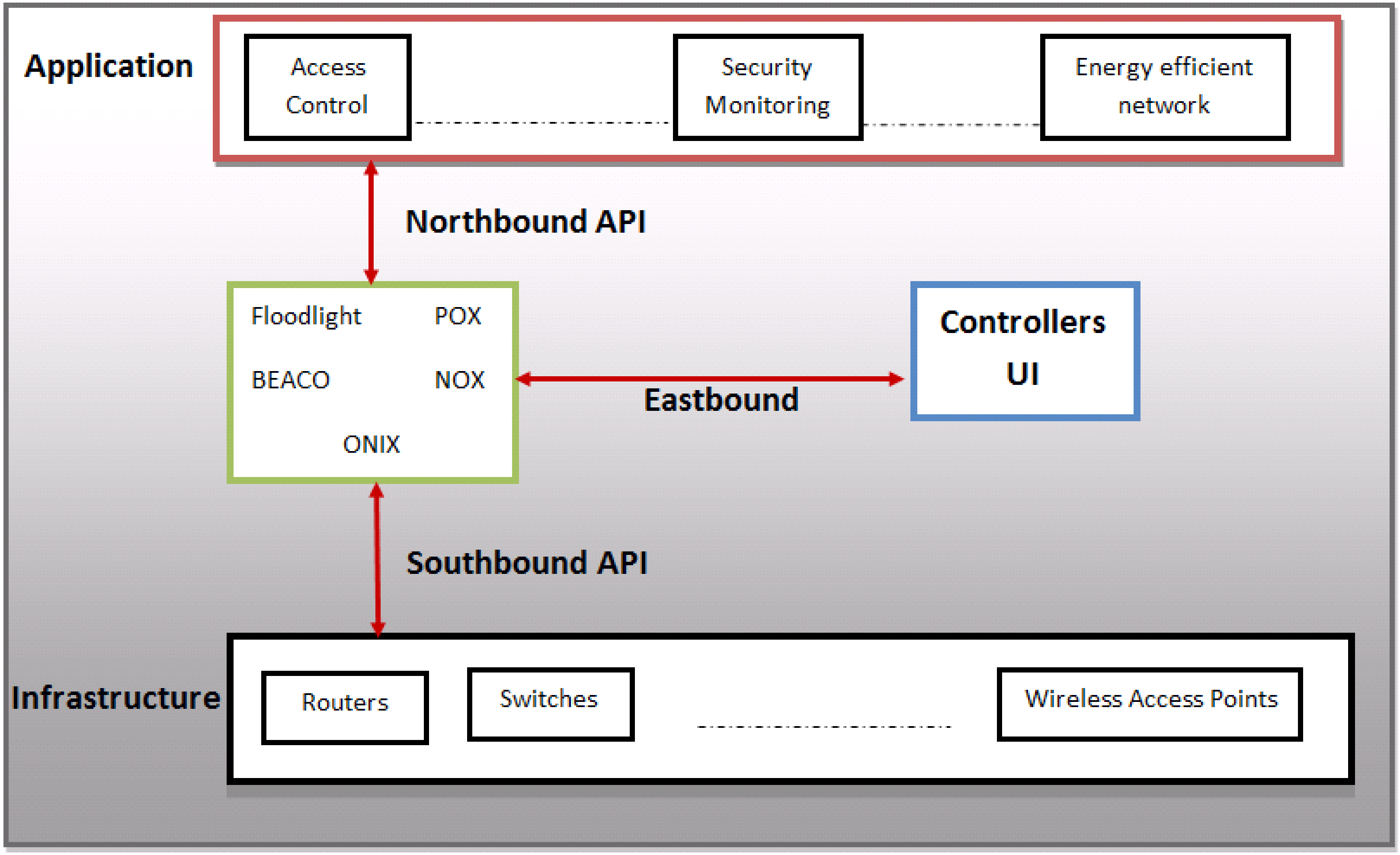
Introduction

- Designing ITRI Cloud Data Center.
- Using Peregrine to create required network:
  - Centralized control.
  - Efficient use of physical links.
  - Reduce fail-over latency.
- Using off-the-self Ethernet switches as basic building blocks.
- Various fail-over strategies used by Peregrine

Terminologies

- ITRI: Industrial Technology Research Institute
- SDN: Software Defined Network
- TOR: Top-of-Rack
- DS: Directory Server (centralized)
- RAS: Route Algorithm Server (centralized)
- ARP: Address Resolution Protocol
- DHCP: Dynamic Host Configuration Protocol

What is SDN?


Application
Access Control
Security Monitoring
Energy efficient network
Northbound API
Floodlight
POX
BEACO
NOX
ONIX
Eastbound
Controllers UI
Southbound API
Infrastructure
Routers
Switches
Wireless Access Points

Peregrine Architecture

- Housed in 20-foot container
- 96 X86 CPU with 3 TB DRAM
- 12 JBOD storage (1PT storage)
- Every rack
  - 48 servers nodes
    - 4 TOR switches
      - 48 1GE ports
      - 4 10GE ports
- Off-the-self Ethernet switches with all build in control plane functionality removed such as source learning, flooding, etc.
- It uses centralized control plane which manages the forwarding tables of the Ethernet switches

Arch. Continued...

Software Arch.

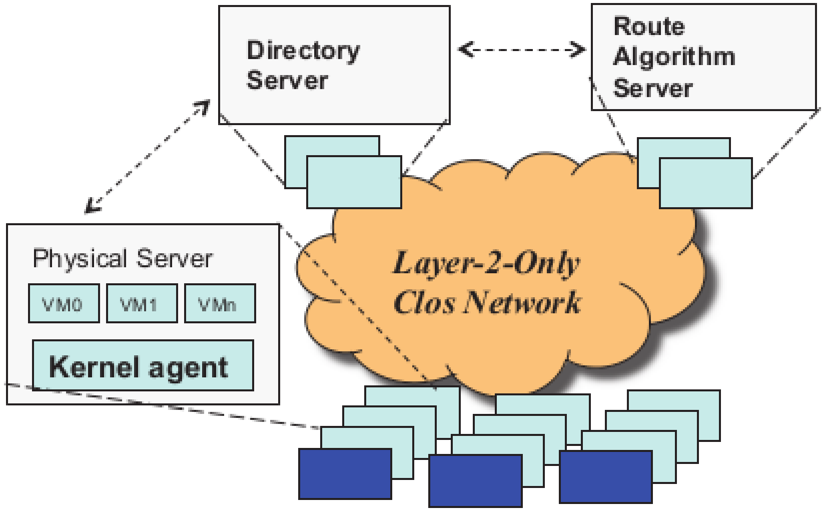


Figure 1: *The software architecture of the current Peregrine prototype, which consists of a kernel agent installed in the Dom0 VM of every physical machine, a centralized directory server (DS) for IP to MAC address look-up, and a centralized route algorithm server (RAS) for route computation and forwarding table population.*

- Kernel agent performing ARP query packet intercept and transformation installed on every physical base Xen Server
- A centralized DS that perform generalized IP to MAC look-up
- A centralized RAS that
  - Constantly collects the network's traffics matrix
  - Runs a load-based routing algorithm based on traffic matrix
  - Populate switches with with forwarding tables with routes
- RAS also build inverse map associated with every link

- <u>Directory Server (DS):</u>
  - Generalized ARP (GARP) map between IP and MAC (primary/secondary)
  - Each GARP map entry keeps a list of caching clients and their expiration time.
  - Directory clients cache GARP entries using a lease-based cache consistency protocol.
- <u>Routing Algorithm Server (RAS):</u>
  - Monitor and collect congestion events and failures.
  - Run time traffic matrix
  - Route engine to compute routes between pairs.
  - Inverse map to associate with network links.

Working of Peregrine

- Centralized IP Address Resolution:
  - Peregrine discourage broadcast protocols such as ARP, DHCP.
  - It replace it with client-server architecture.
  - When VM send ARP query:
    - Peregrine agent on same server intercept it and convert the query into unicast packet and sent it to DS
    - DS sent reply to Peregrine agent and agent converts it into ARP response and send to original VM
    - Agent also cache the DS response for future ARP queries
  - Lease-based stateful cache is used to maintain consistency of ARP and do unicast based invalidation notification to VMs if they expire.
  - This helps in:
    - Scalling up network size
    - Redirection of VM migration
    - Fail-over in network

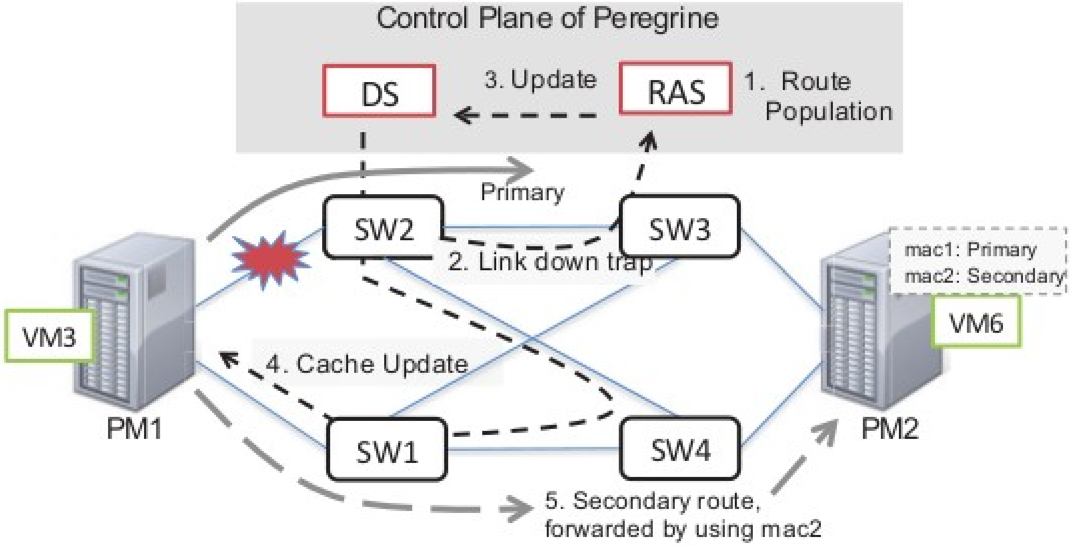
- Primary/Secondary Routing: (X:physical server)
  - Main goal is to reduce fail-over time to 100ms.
  - To do this Pre-computation of primary and secondary route from other physical servers are done at X
  - To support switch from primary to seconday:
    - Assigning multiple MAC address to physical servers
    - So each MAC created distinct paths to reach X
    - Peregrine install pre-computed primary/secondary routes to every server and switche's forwarding table
  - By default primary path is used.

Working of Peregrine

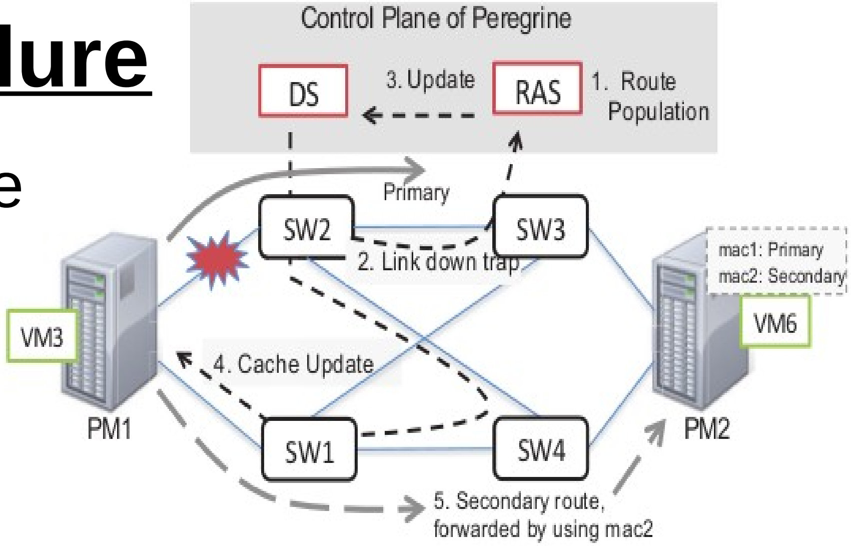
- Centralized IP Address Resolution:
  - Peregrine discourage broadcast protocols such as ARP, DHCP.
  - It replace it with client-server architecture.
  - When VM send ARP query:
    - Peregrine agent on same server intercept it and convert the query into unicast packet and sent it to DS
    - DS sent reply to Peregrine agent and agent converts it into ARP response and send to original VM
    - Agent also cache the DS response for future ARP queries
  - Lease-based stateful cache is used to maintain consistency of ARP and do unicast based invalidation notification to VMs if they expire.
  - This helps in:
    - Scalling up network size
    - Redirection of VM migration
    - Fail-over in network

Fault Tolerance Support

- Broad classification:
  - Fail-over for network
  - Fail-over for DS/ RAS
  - Messaging on fail-over
  - Broadcast support

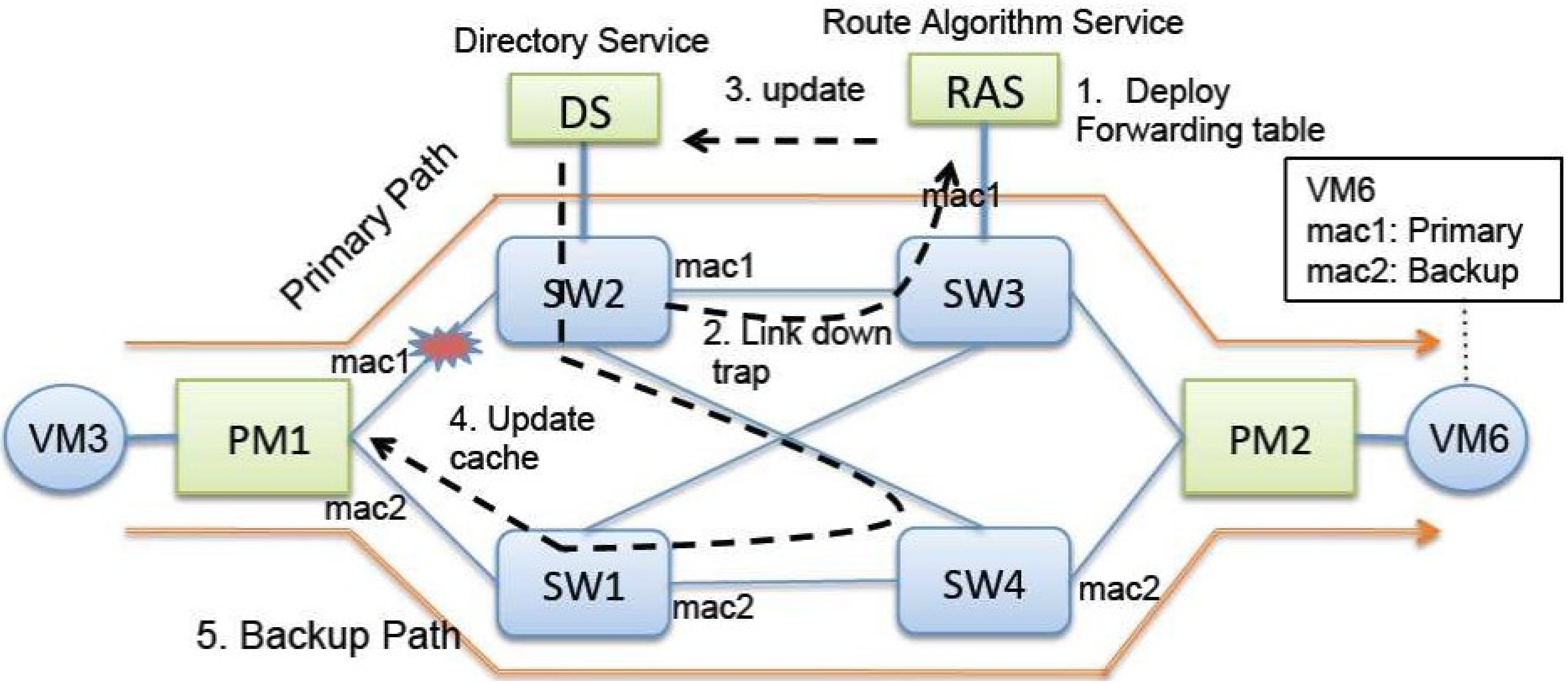


Fast fail-over for network failure



- On switch failure RAS receive multiple SNMP traps
- RAS verify it by ping response
- On detection of failure RAS:
  - Check inverse map for paths which includes failed switch and update DS
  - If DS check any primary route is effected notify all pairs (servers) and turn of primary routes
  - RAS activate secondary routes in forwarding tables

Network failure:



http://conf.ncku.edu.tw/icpads/File/KeynoteSpeach-II.pdf

Fast fail-over for DS/RAS failure

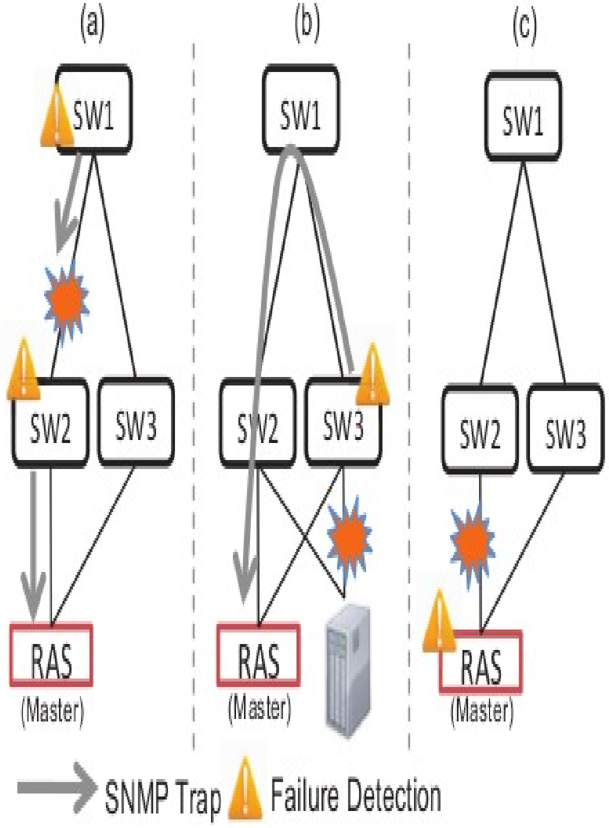
- DA/RAS are important part of centralized control therefore should be available in spit of failure.
- All data structures related to DS and RAS are stored on disk.
- Active master and passive slave architecture is used.
  - Master state is first logged into memory-resident logs
  - Synchronously replicated to slave
  - Asynchronously written on disk and synchronously updated on slave disk
- Slaves take over if masters dies.
- Pacemaker tools are used to monitor status of DS and RAS masters.

Fast fail-over for DS/RAS failure

- DA/RAS are important part of centralized control therefore should be available in spit of failure.
- All data structures related to DS and RAS are stored on disk.
- Active master and passive slave architecture is used.
  - Master state is first logged into memory-resident logs
  - Synchronously replicated to slave
  - Asynchronously written on disk and synchronously updated on slave disk
- Slaves take over if masters dies.
- Pacemaker tools are used to monitor status of DS and RAS masters.

Resilient Messaging during fail-over

- Peregrine set two MAC address for DS and RAS and creates two disjoint paths.
- Every switch is configured to send SNMP packet twice.
- Kernel agent keep track of the IP and MAC address of DS and RAS, which are used in case of ARP timeout.
- On startup RAS connect to DS and list all address using UDP.



Broadcast support

- Peregrine is designed to minimize broadcast-based protocols.
- Some cases broadcast messages are supported such as commercial switches or routers on which Peregrine agent is not installed.
- To avoid Ethernet storms:
  - Uses tree structure spans all nodes
  - Allowing broadcast to flow only in tree
  - Disabling all other node's port not in tree
- Tree is recreated in case of link/switch failure.

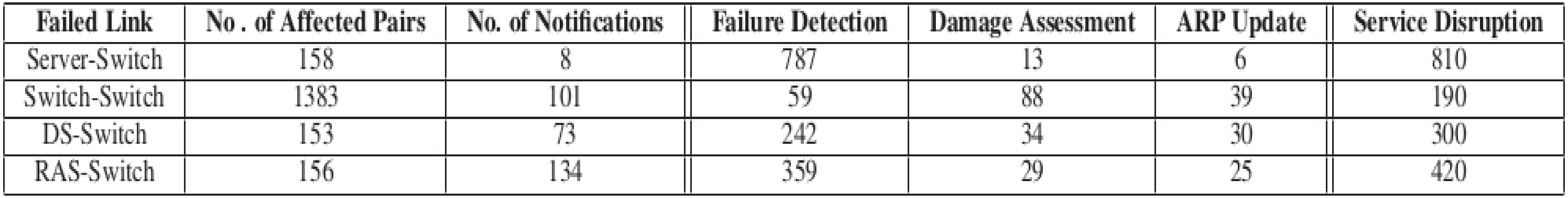
Relation to course work

- Architecture is created to support fault tolerance based on SNMP feeds from nodes and switches.
- Primary/secondary path are added as fail-safe.
- Route recreation based on feedback from switches.
- Network controlled using centralized DS/RAS controller.

Performance Evaluation

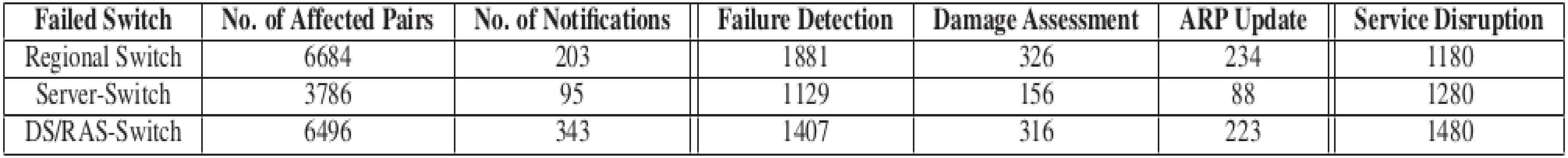
- Service disruption divided into four broad sections:
  - Failure detection time
  - Damage assessment time
  - ARP update time
  - Switch-over time
- Evaluation is done by sending UDP packets from source to RAS every msec.

Link and Switch failure data



Failed Link	No . of Affected Pairs	No. of Notifications	Failure Detection	Damage Assessment	ARP Update	Service Disruption
Server-Switch	158	8	787	13	6	810
Switch-Switch	1383	101	59	88	39	190
DS-Switch	153	73	242	34	30	300
RAS-Switch	156	134	359	29	25	420

Table 1: *The average service disruption times of four different types of link failure and their detailed breakdowns. All time measurements are in terms of ms.*



Failed Switch	No. of Affected Pairs	No. of Notifications	Failure Detection	Damage Assessment	ARP Update	Service Disruption
Regional Switch	6684	203	1881	326	234	1180
Server-Switch	3786	95	1129	156	88	1280
DS/RAS-Switch	6496	343	1407	316	223	1480

Table 2: *The average service disruption times of three different types of switch failures and their detailed breakdowns. All time measurements are in terms of ms.*

Conclusion

- Peregrine is SDN implementation on a very broader scale and uses off-the-self Ethernet switches.
- Its more scalable then with high availability then traditional networks.
- Centralized control plane and distributed data plane.
- Self-adaptive and learning architecture.
- No broadcast flooding, source learning
-

Thank You