

CENTER OPEN MIDDLEWARE

Universidad Politécnica de Madrid (UPM)



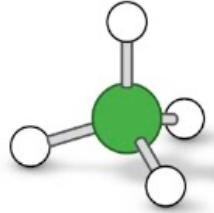
POLITÉCNICA

SoloWAN: open source WAN optimization

David Fernández, F. Javier Ruiz, Luis Bellido, Raúl Álvarez, German Martín, Carlos Vega, Roberto Montero, Mattia Peirano, Francisco Huertas, Antonio Celorio, Juan Carlos Dueñas

solowan@dit.upm.es

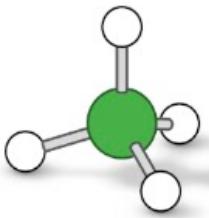
Center for Open Middleware
Universidad Politécnica de Madrid



Contents

- WAN optimization techniques
- SoloWAN project
- Use case scenarios and results
- Virtual testbeds scenarios based on VNX





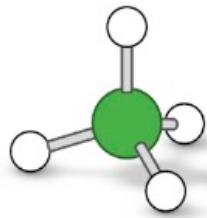
WAN Optimization Techniques

- **Objective:** improve WAN links performance
 - WAN link characteristics: lower bandwidth, higher delays, higher cost

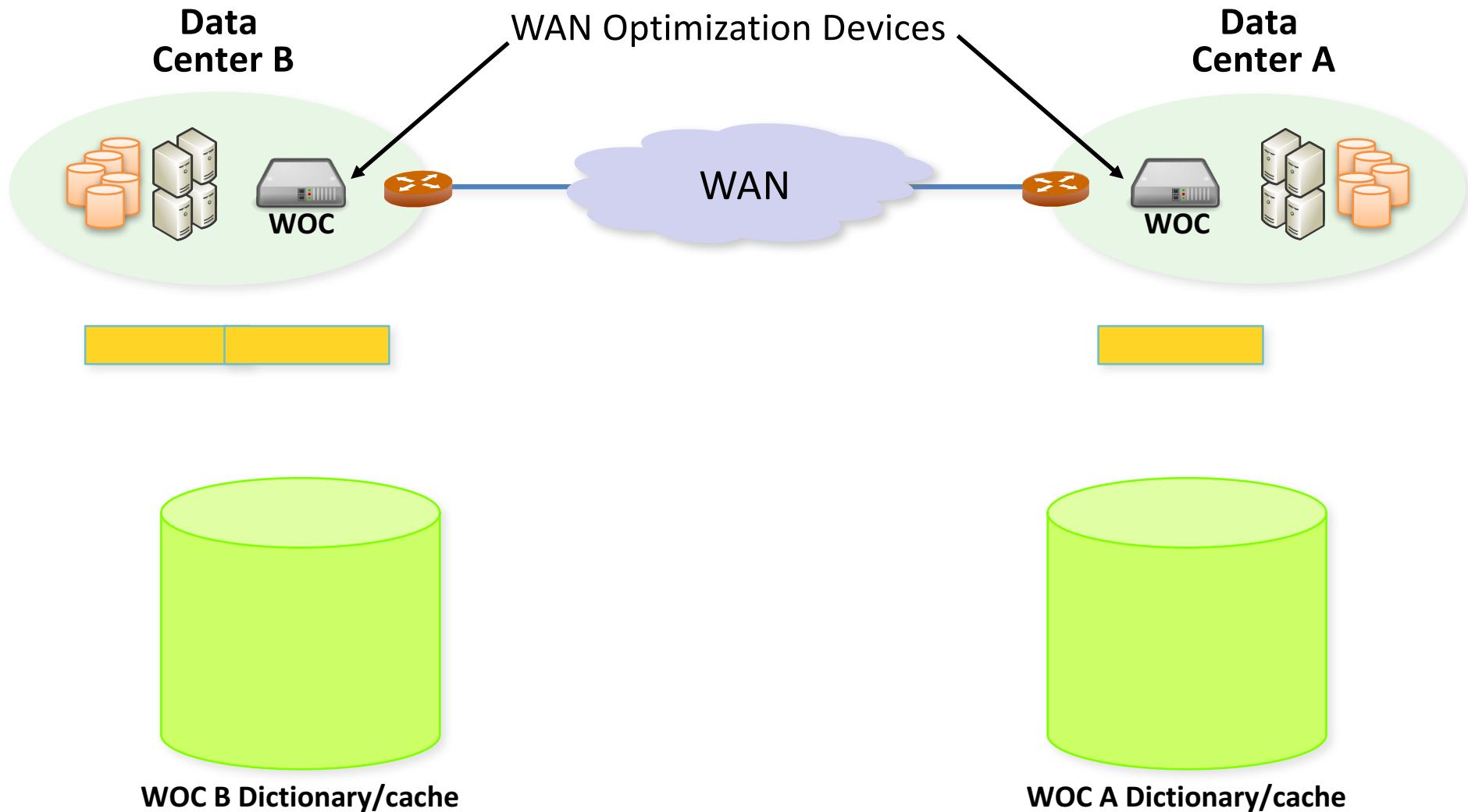


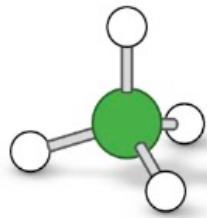
- **Techniques:**
 - Compression, Deduplication (or dictionary based compression), TCP optimization, Application caches, etc.



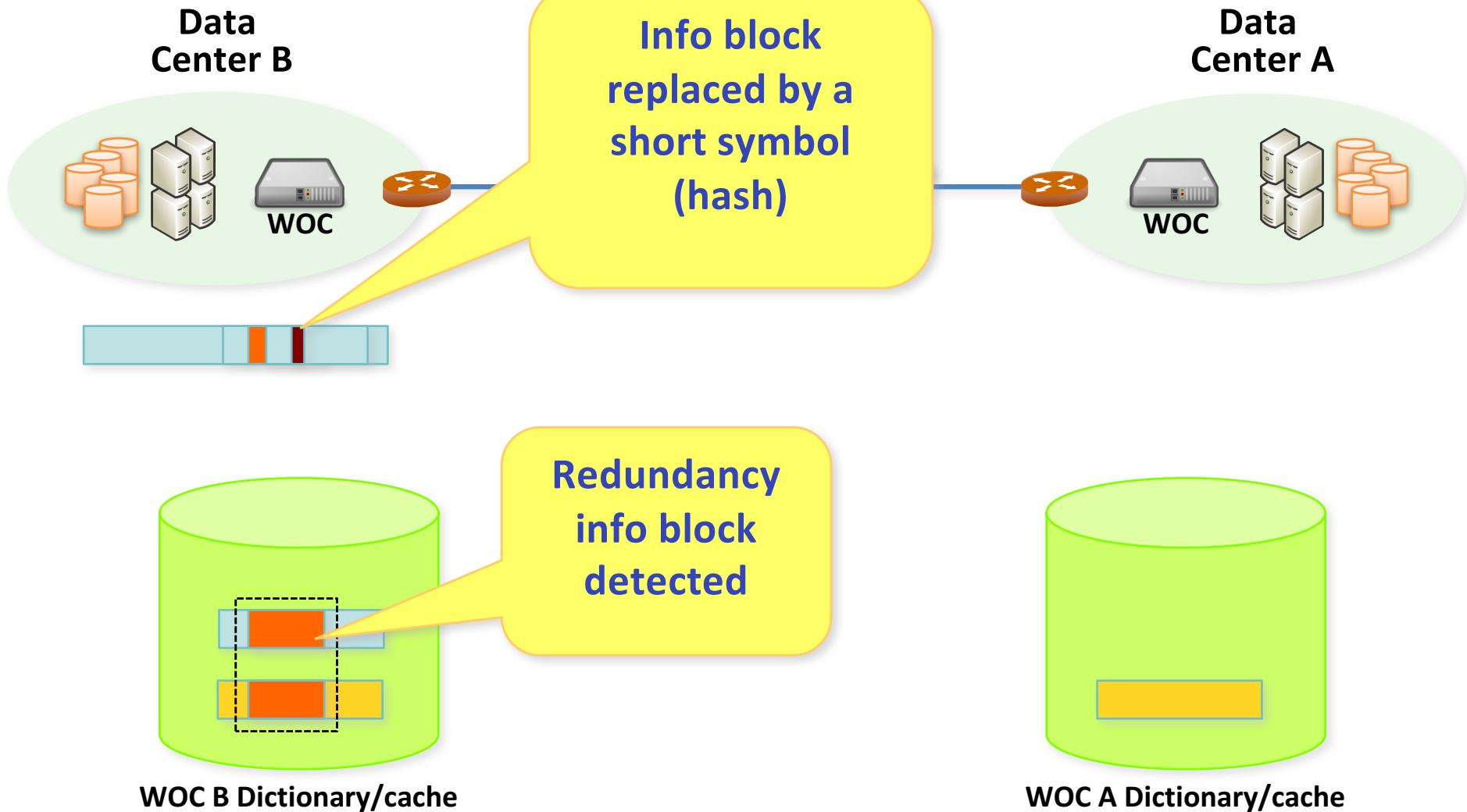


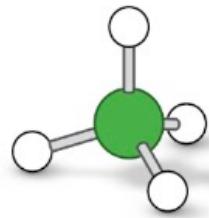
WAN optimization based on deduplication



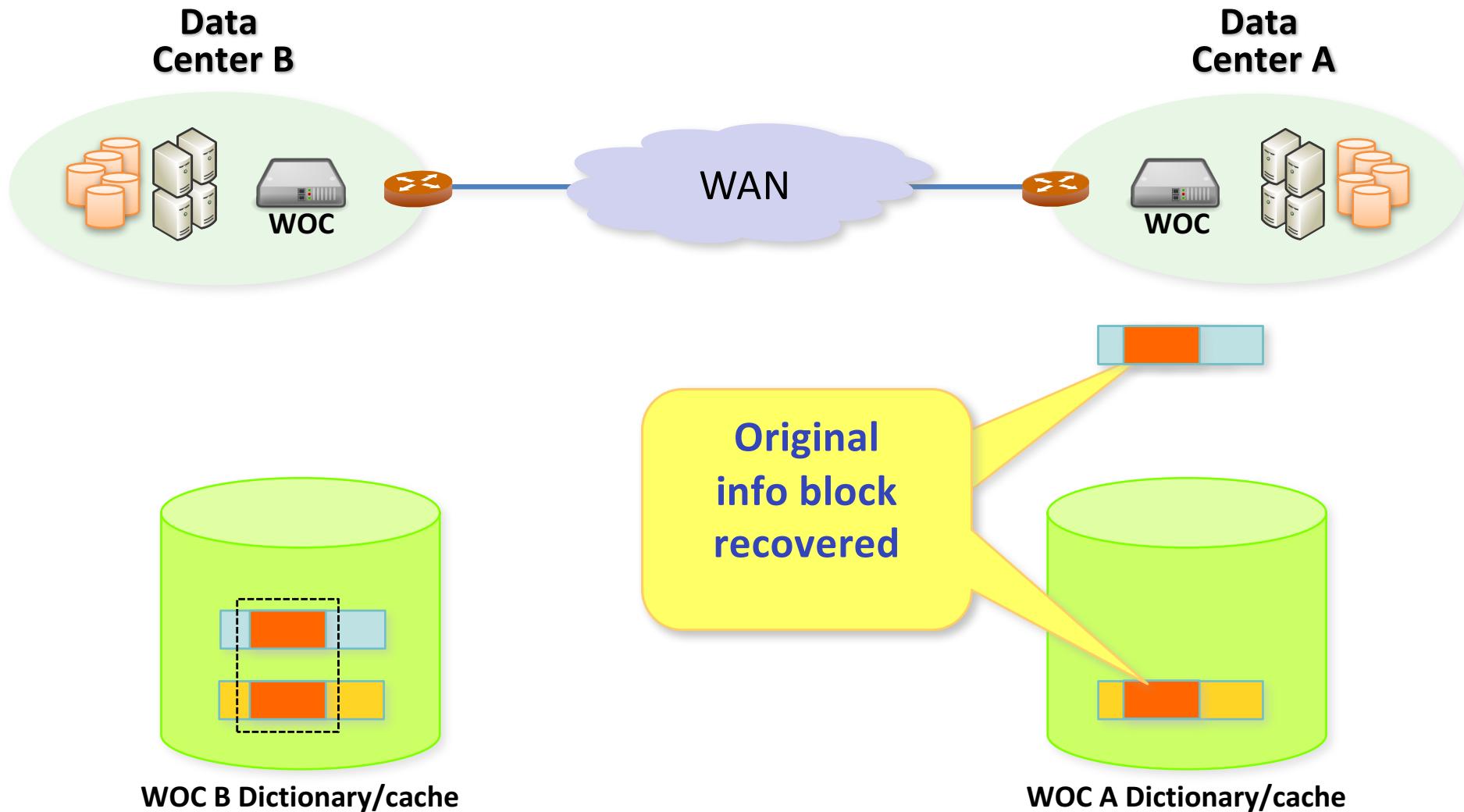


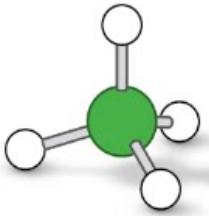
WAN optimization based on deduplication





WAN optimization based on deduplication



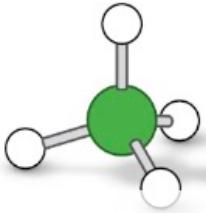


SoloWAN rationale

- Commercial WOC solutions:
 - Complex systems
 - Scalability and elasticity issues
 - High cost in terms of equipment and licensing

What about open source software (OSS) based WAN optimization solutions?





SoloWAN initial objective



WANProxy



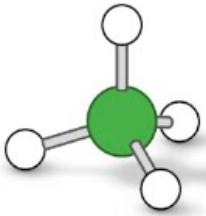
Any OSS mature solution found?

YES

NO

Objective reoriented to enhance OSS OpenNOP WAN optimization platform with:

- Efficient deduplication algorithm
- Scalable and elastic architecture running on standard hardware

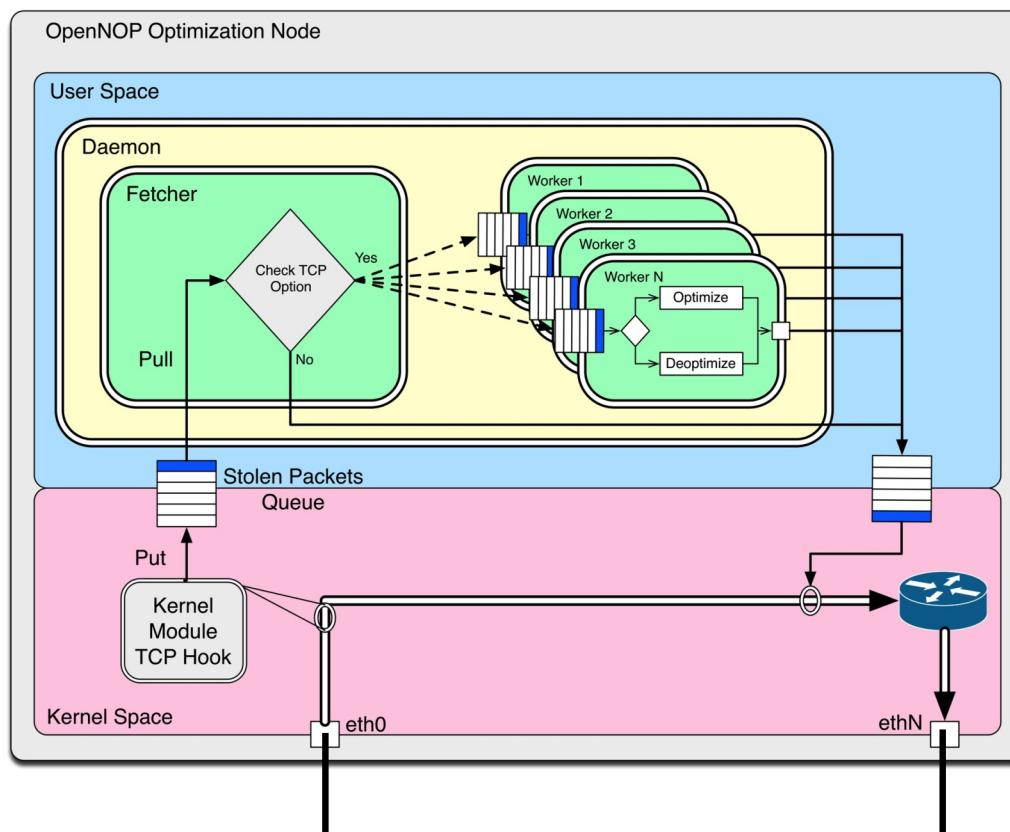


SoloWAN Architecture

- Based on Open Network Optimization Platform
(OpenNOP, www.opennop.org)

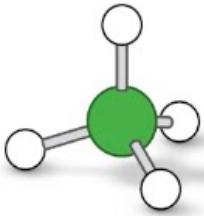
Linux
User
Space

Linux
Kernel



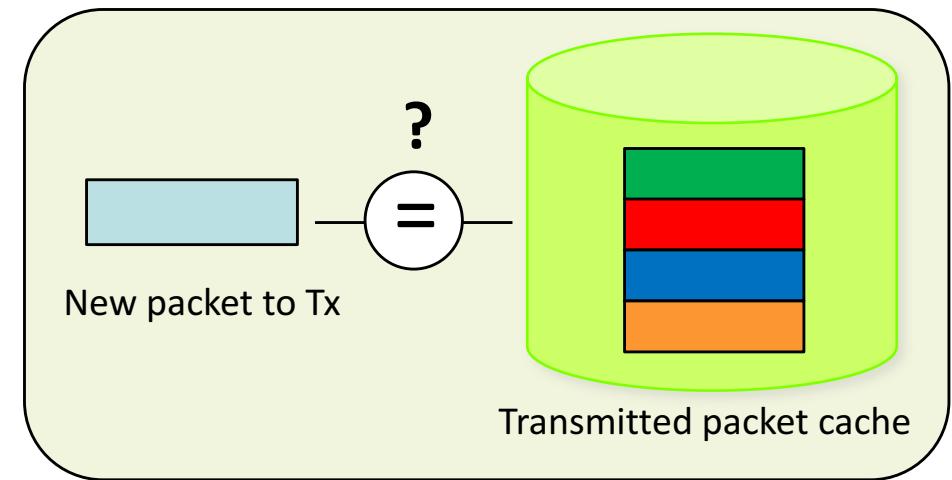
- Linux kernel module + application in user-space
- Use of netfilter to capture traffic
- Multithread (worker)
- Written in C

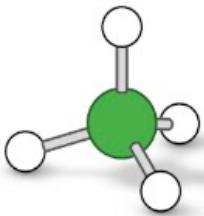




Deduplication Algorithm

- Deduplication functionality based on:
 - Neil T. Spring and David Wetherall. *A Protocol-Independent Technique for Eliminating Redundant Network Traffic*. SIGCOMM '00.
- Main problem: how to efficiently identify repeated content?
 - Computationally intensive
- Simplification:
 - Compare only fixed size blocks (64 bytes) at a reduced number of positions (32)
 - Calculate a hash (fingerprint) for each block inside a packet and store it in the dictionary together with the packet
 - Once a match is detected, look around matching block to see if more bytes match

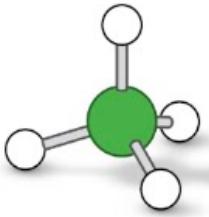




SoloWAN characteristics

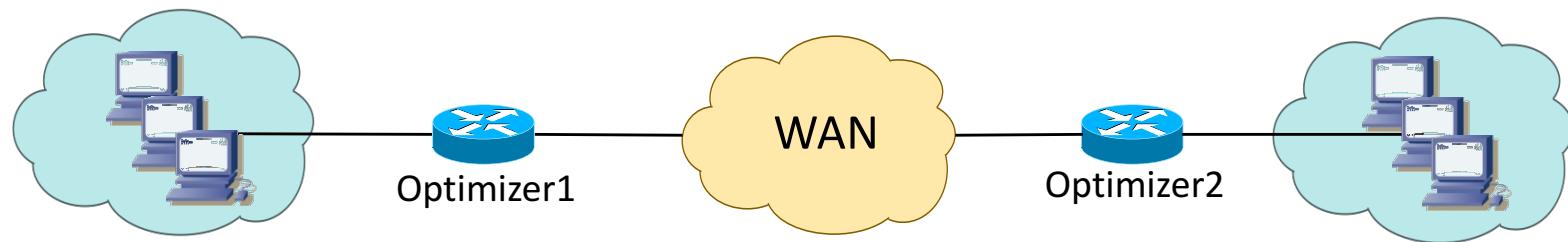
- **Algorithms:**
 - Compression (originally included in OpenNOP)
 - Deduplication
 - Deduplication + Compression
- Available as a linux daemon or as a docker container
- First version released in nov 2014
 - <https://github.com/solowan/solowan>
- Multithread architecture with dictionaries shared among threads
- Demo virtual machine available with VNX based virtual scenarios
- Wireshark Plugin to decode SoloWAN packet format



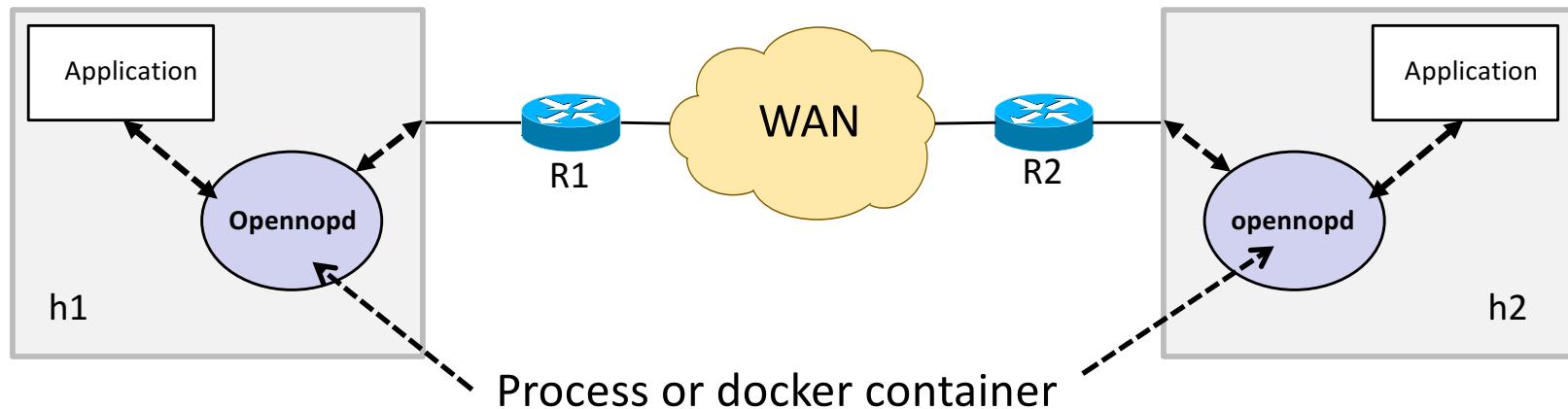


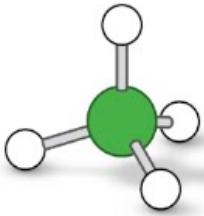
SoloWAN Scenarios

- Optimizer-in-the-Network (oithn)



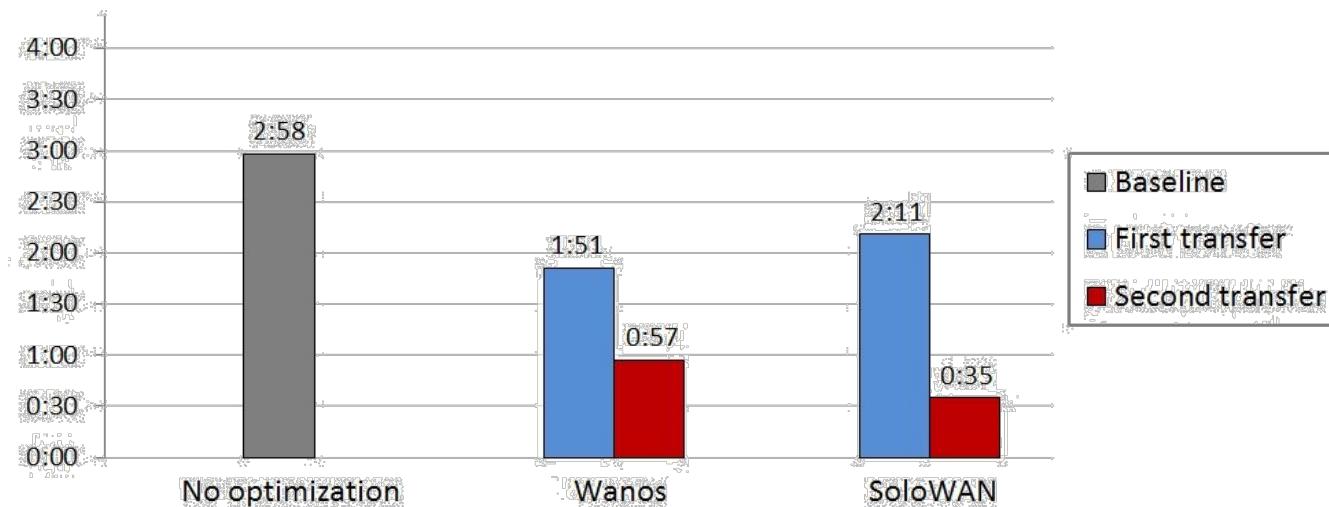
- End-to-End (e2e)

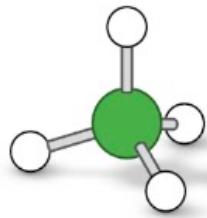




SoloWAN Tests

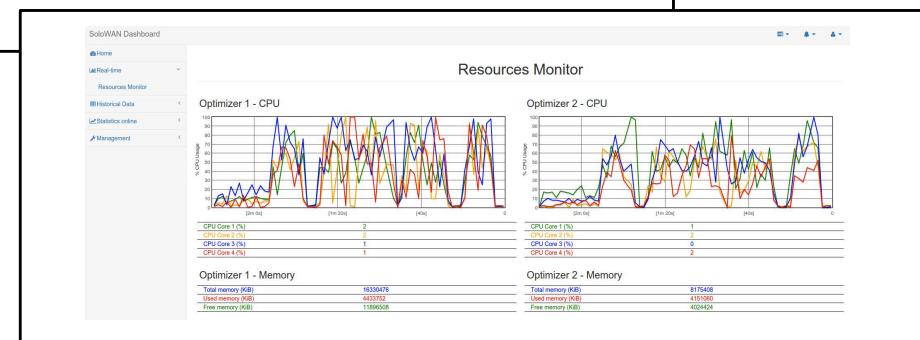
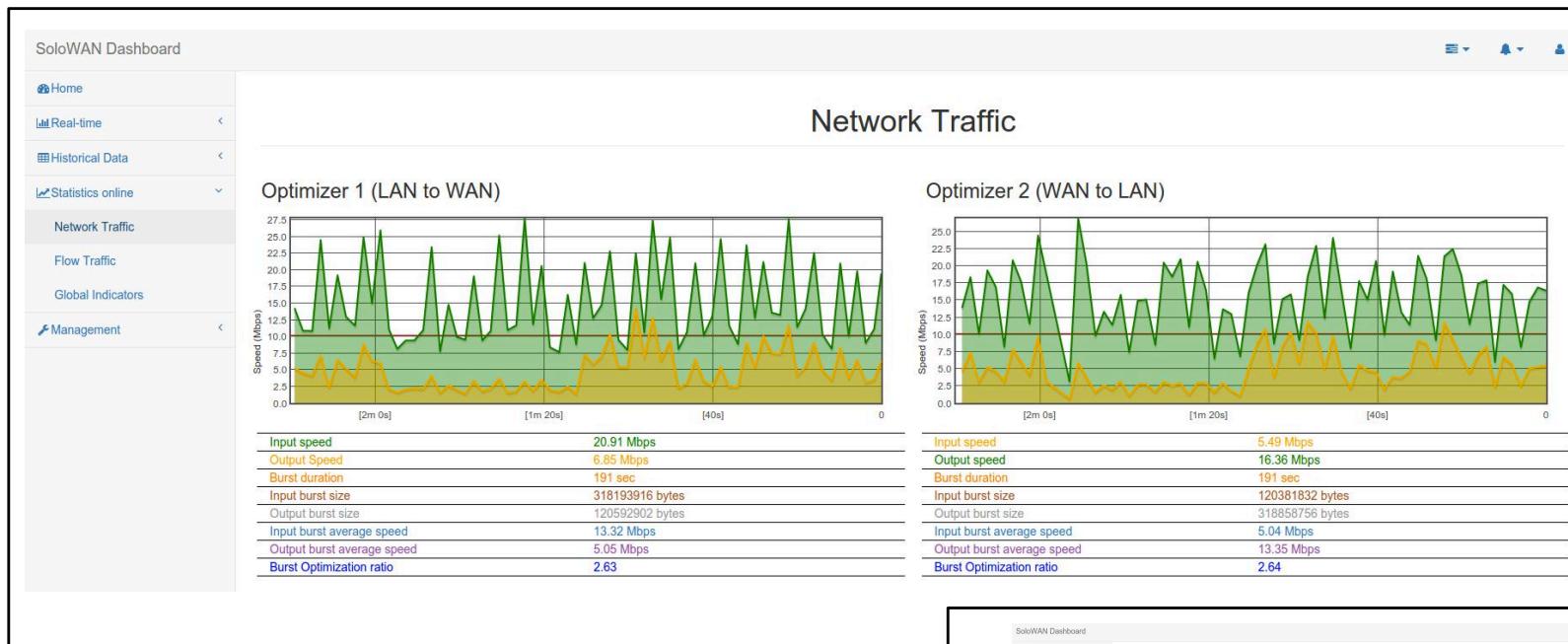
- Mainly tested with FTP, HTTP and access to mysql databases
 - Not suitable for encrypted traffic
- Using benchmark files from Silesia Corpus:
 - Set of representative files (de 6 a 51 M)
 - <http://www.data-compression.info/Corpora/SilesiaCorpus/>

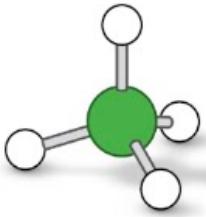




Web application interface

- Limited management and real time statistics:

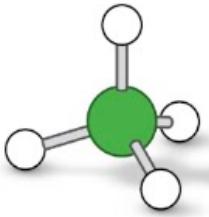




SoloWAN in German IT press

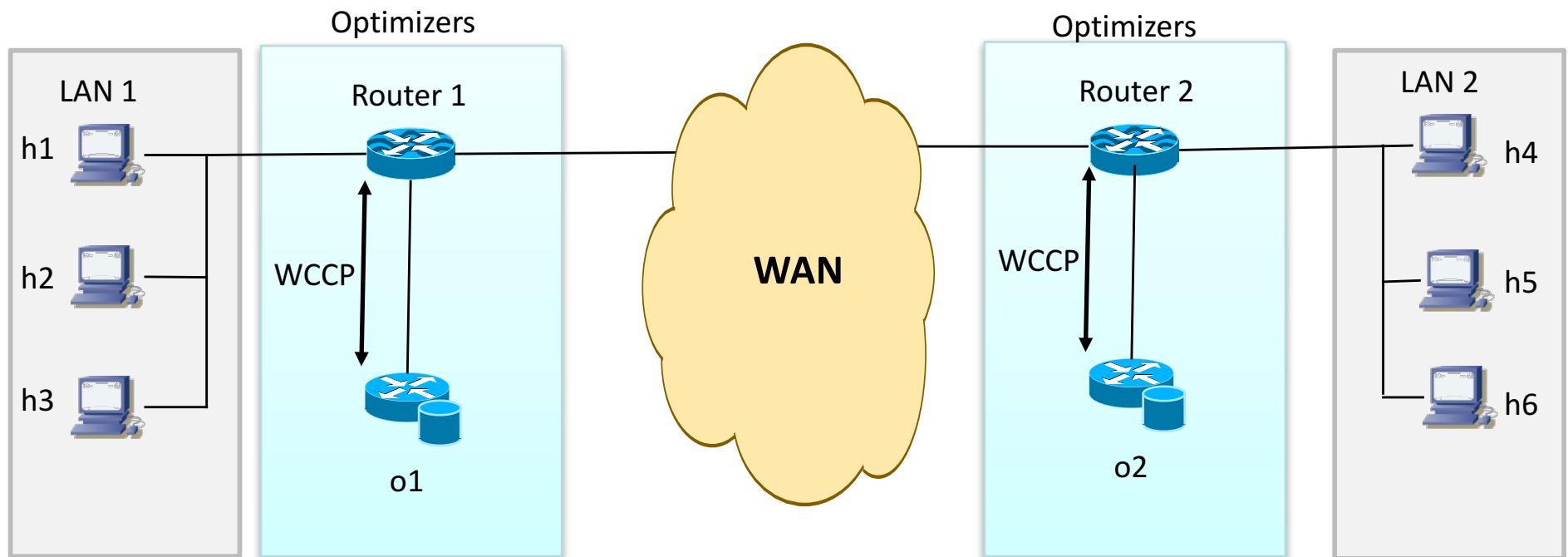
- SoloWAN in the popular German enterprise IT magazine:
iX - Magazin für Professionelle Informationstechnik
- Article about open source based optimization solutions:
 - OpenNOP
 - SoloWAN
 - TrafficSqueezer
 - Wanproxy
- Includes laboratory tests:
 - Highlights the easy installation and high compression rate provided by SoloWAN

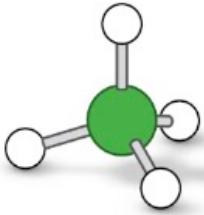




SoloWAN Scenarios

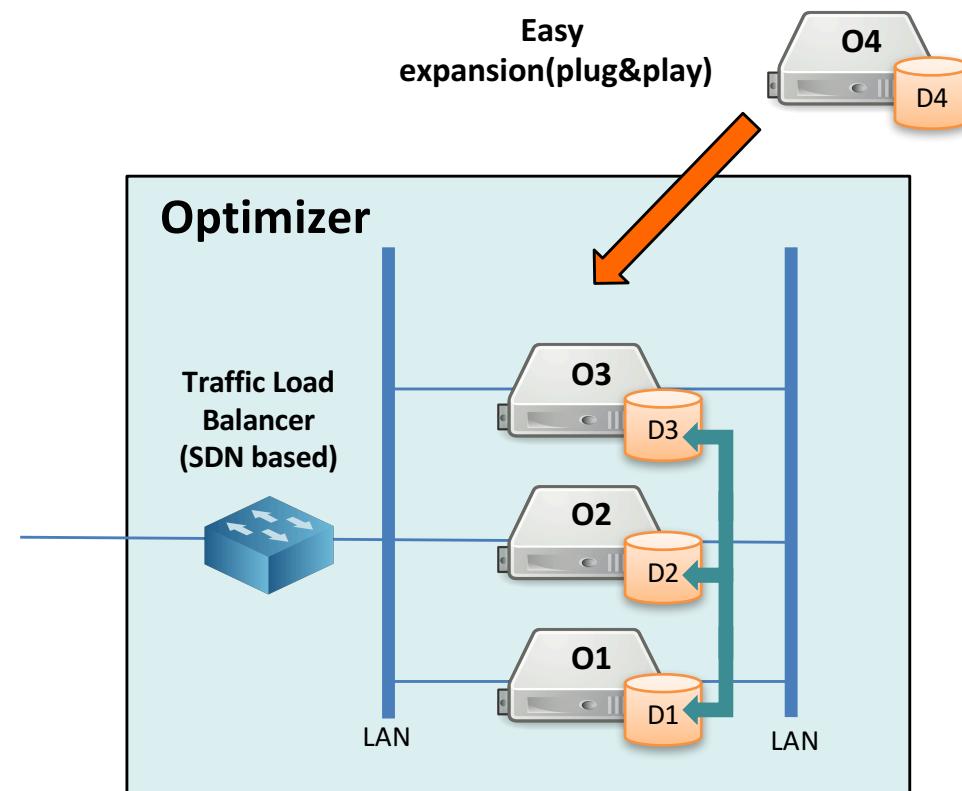
- Traffic redirection based on WCCPv2

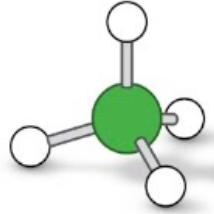




SoloWAN cloud architecture

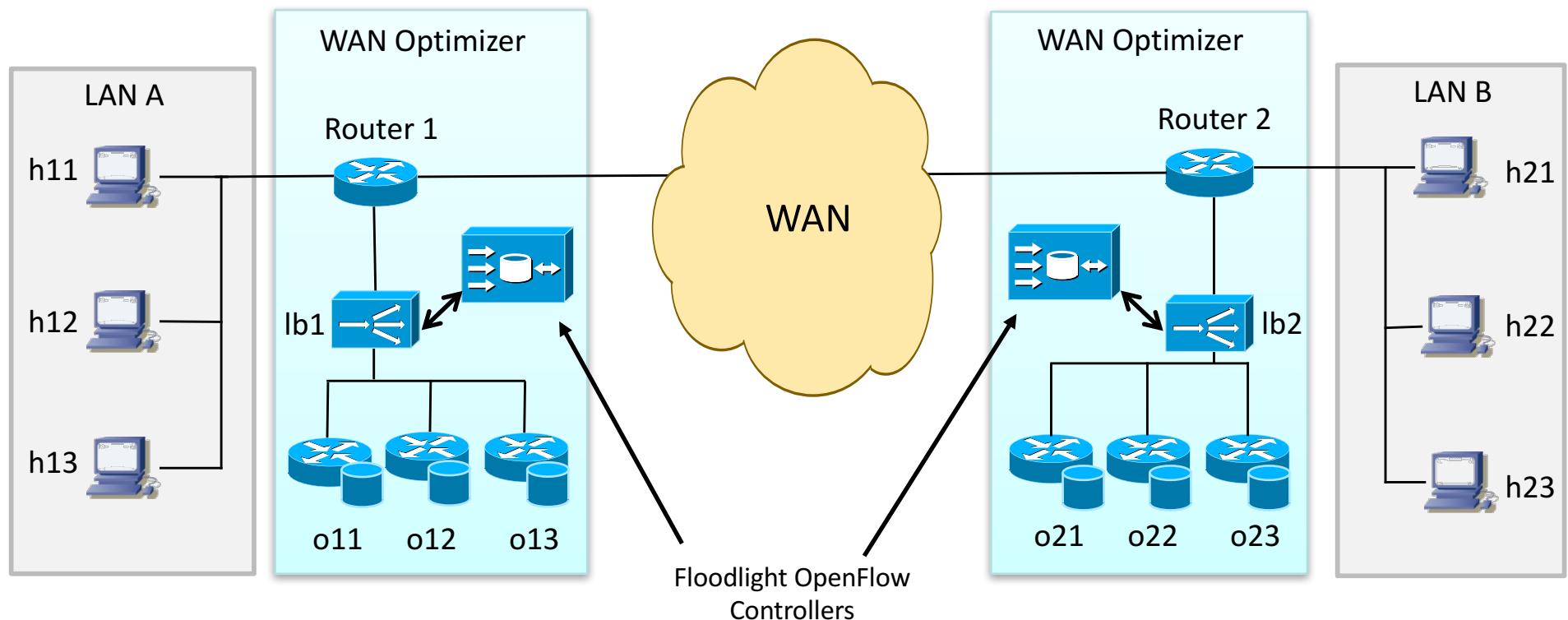
- Create a scalable architecture based in SoloWAN with support for multiple parallel optimizers
 - Multiple threads per machine
 - Multiple servers
- Use cloud computing paradigms to add optimizers as needed

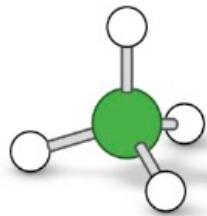




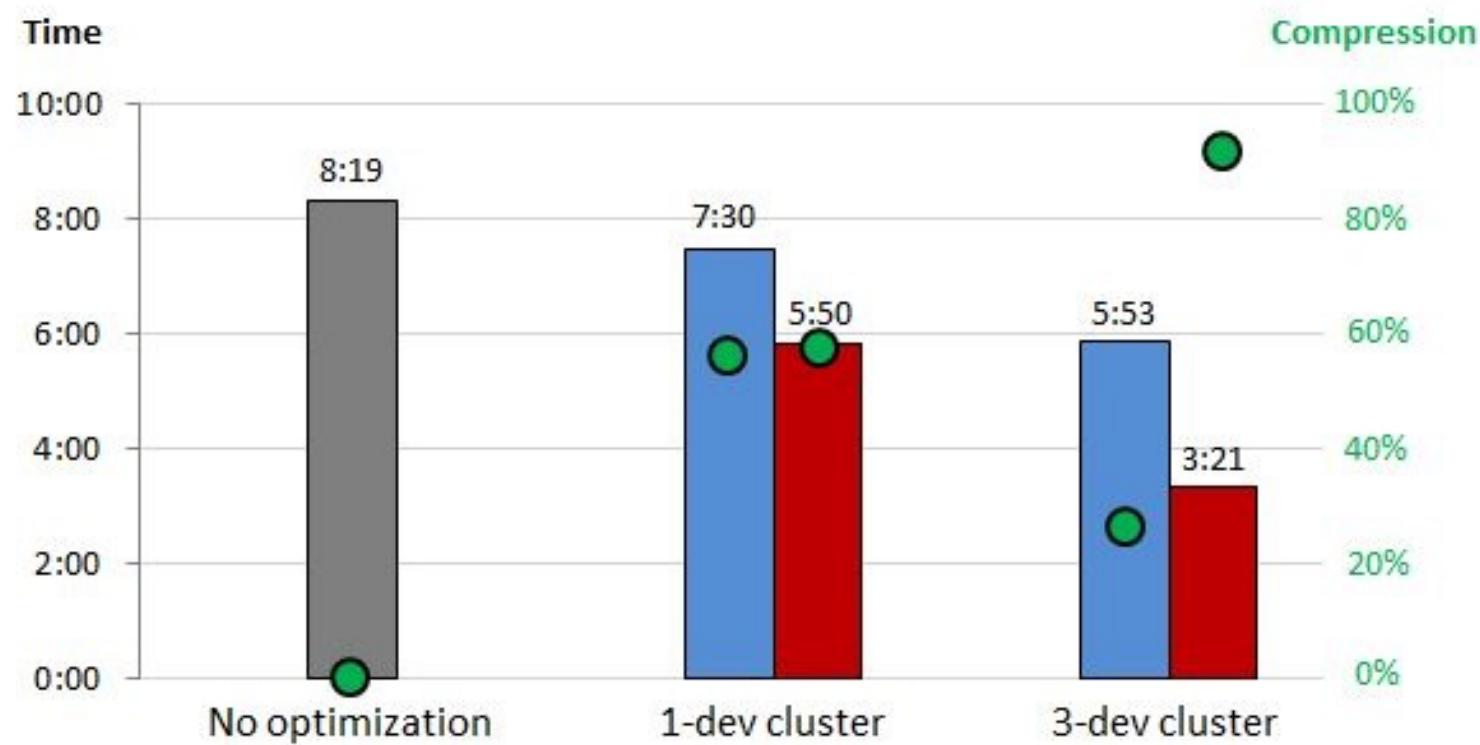
SDN based SoloWAN cloud architecture

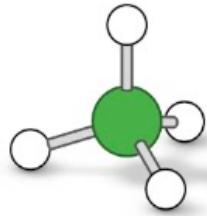
- Reference and testbed scenario:





Scalable architecture Tests

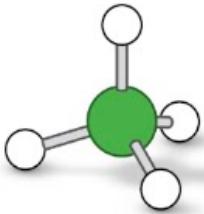




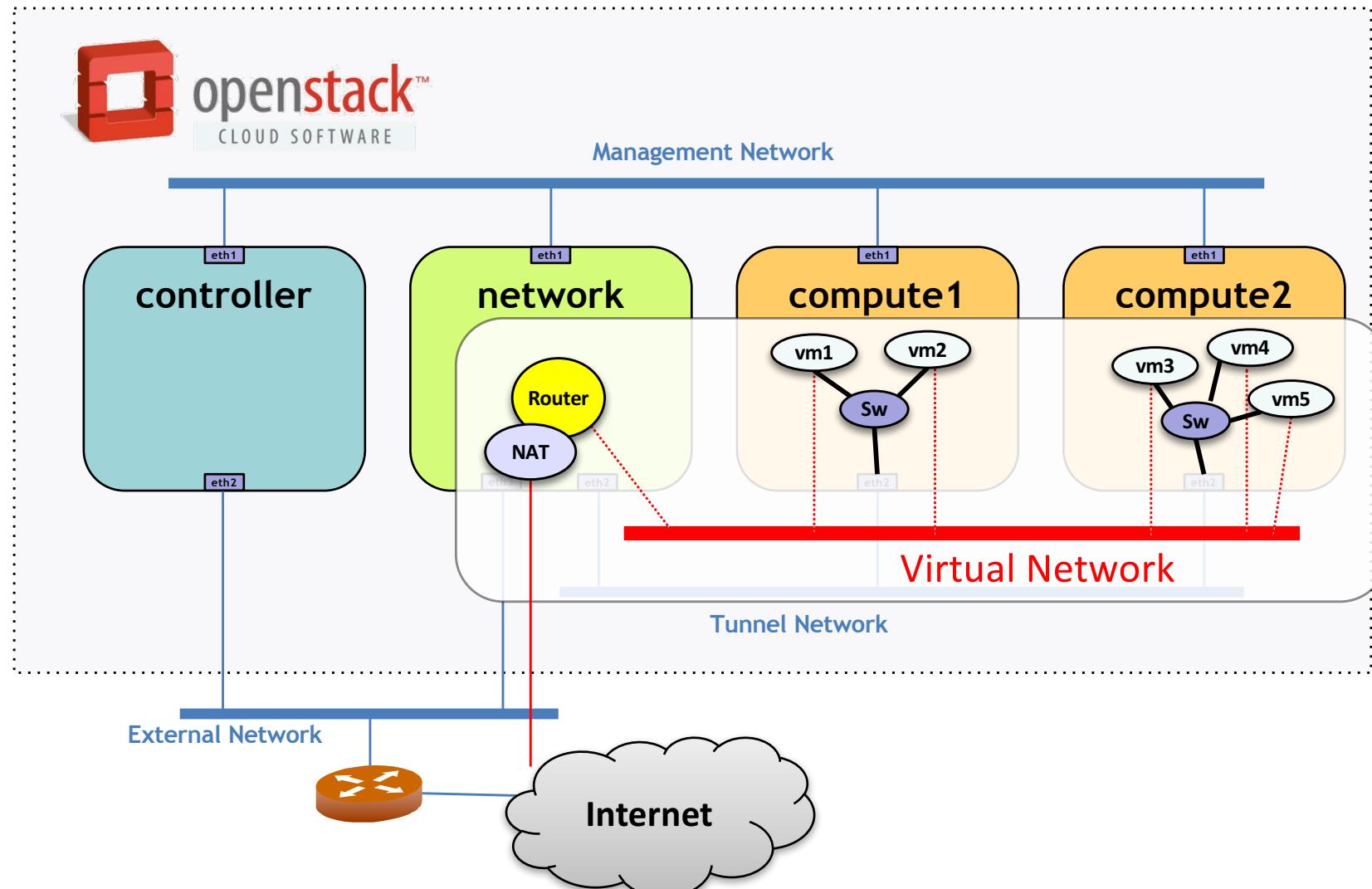
WAN Optimization as a Service

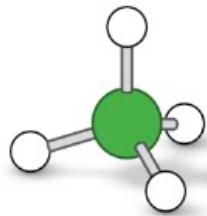
- Idea: Integrate WAN optimization as a service available in Openstack IaaS
- Proof of Concept implementation developed
 - <https://github.com/carlosv5/OaaS-network>
- Based on Firewall as a Service (FWaaS) Openstack code



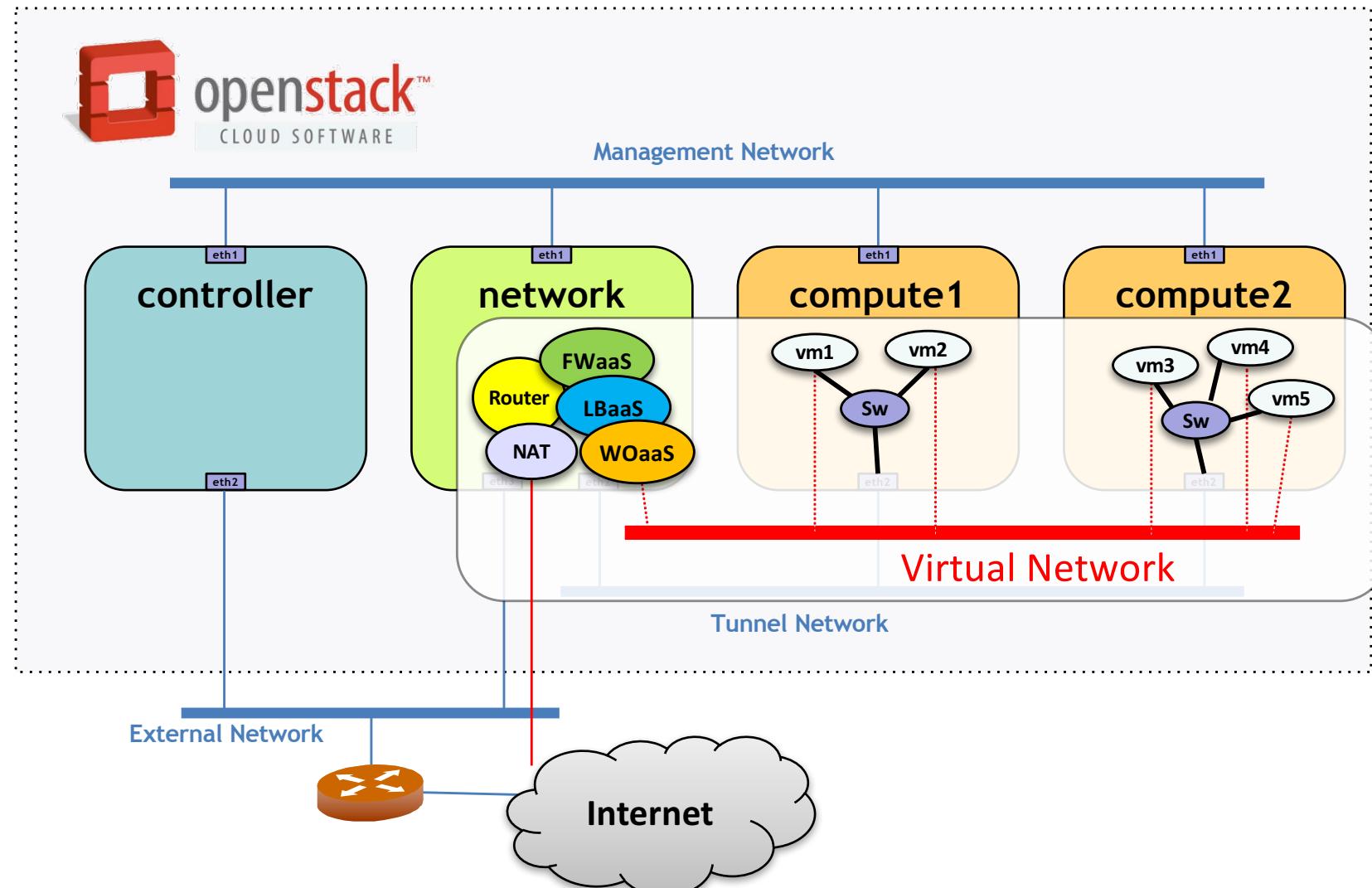


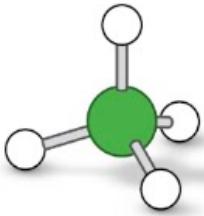
Typical Openstack Scenario





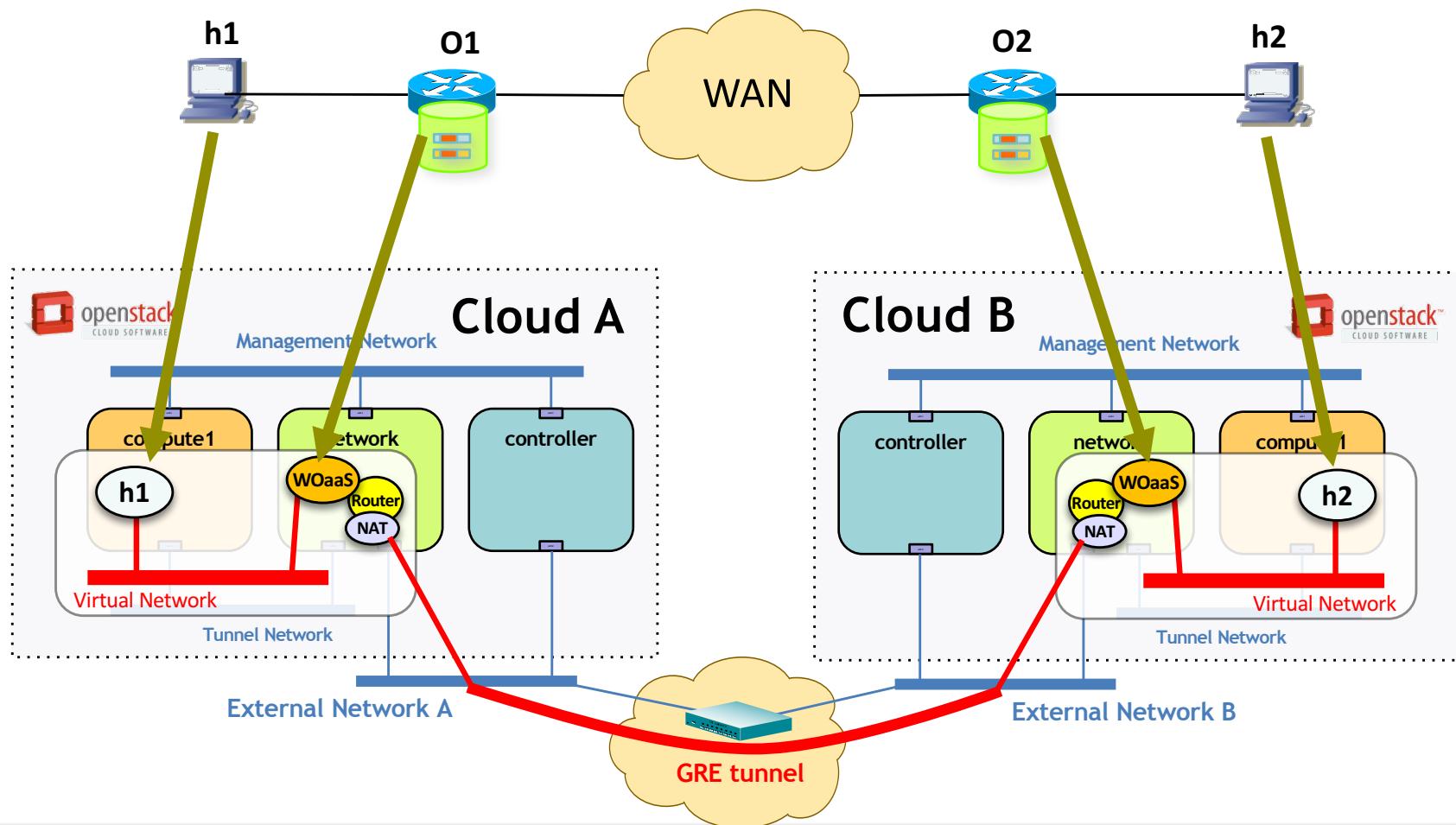
WAN Optimization as a Service

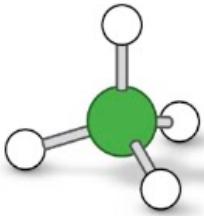




WOaaS Complete Test Scenario (I)

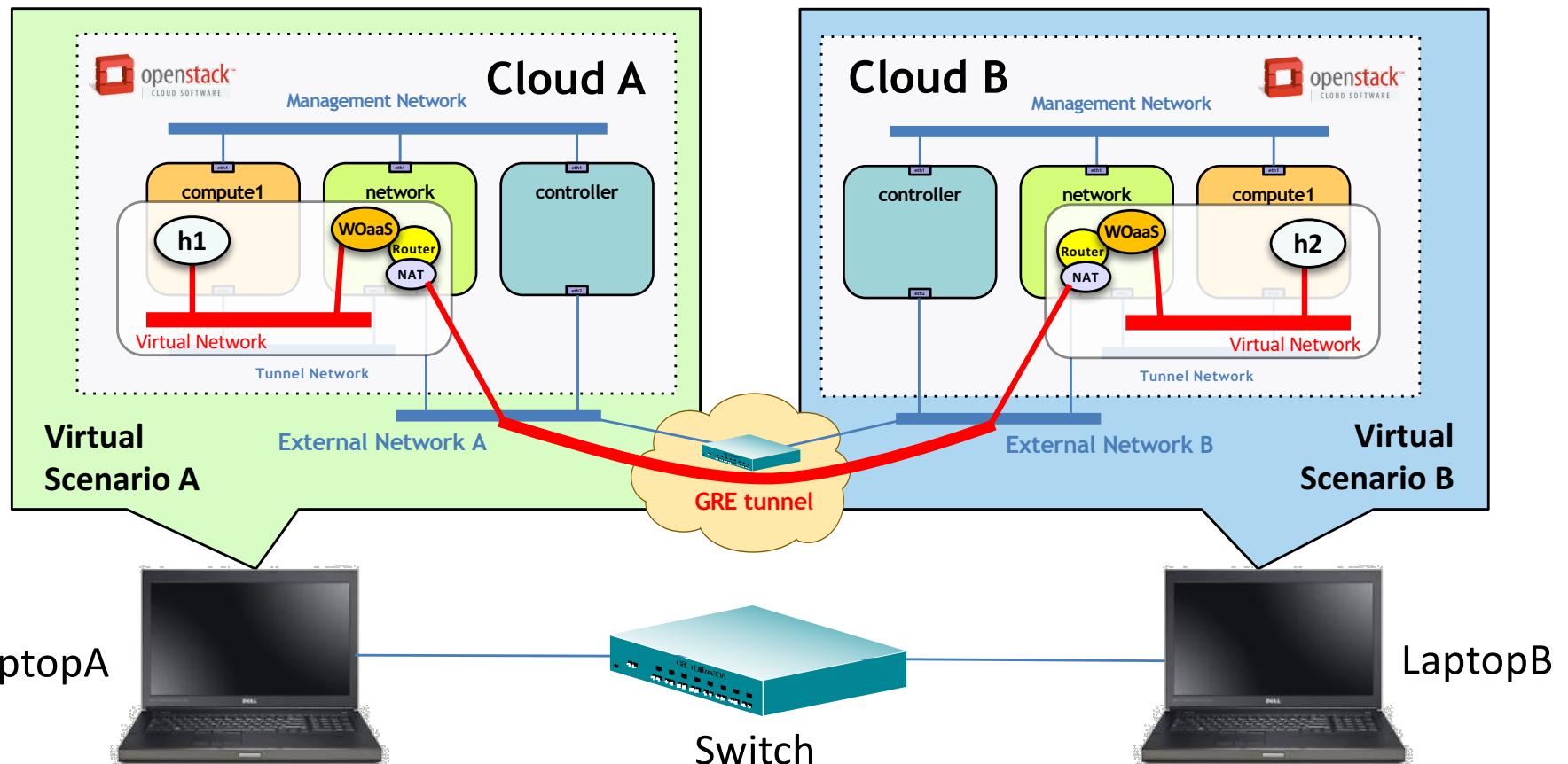
- Basic scenario tested over two Openstack clouds





WOaaS Complete Test Scenario (II)

- Testing infrastructure:



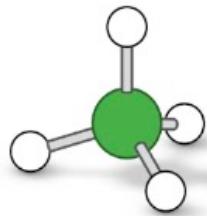
VNX @vnx_upm · Oct 23

New Openstack Mitaka scenario. Test a simple Openstack scenario made of controller, network and 2 compute nodes: goo.gl/Wtrx2J

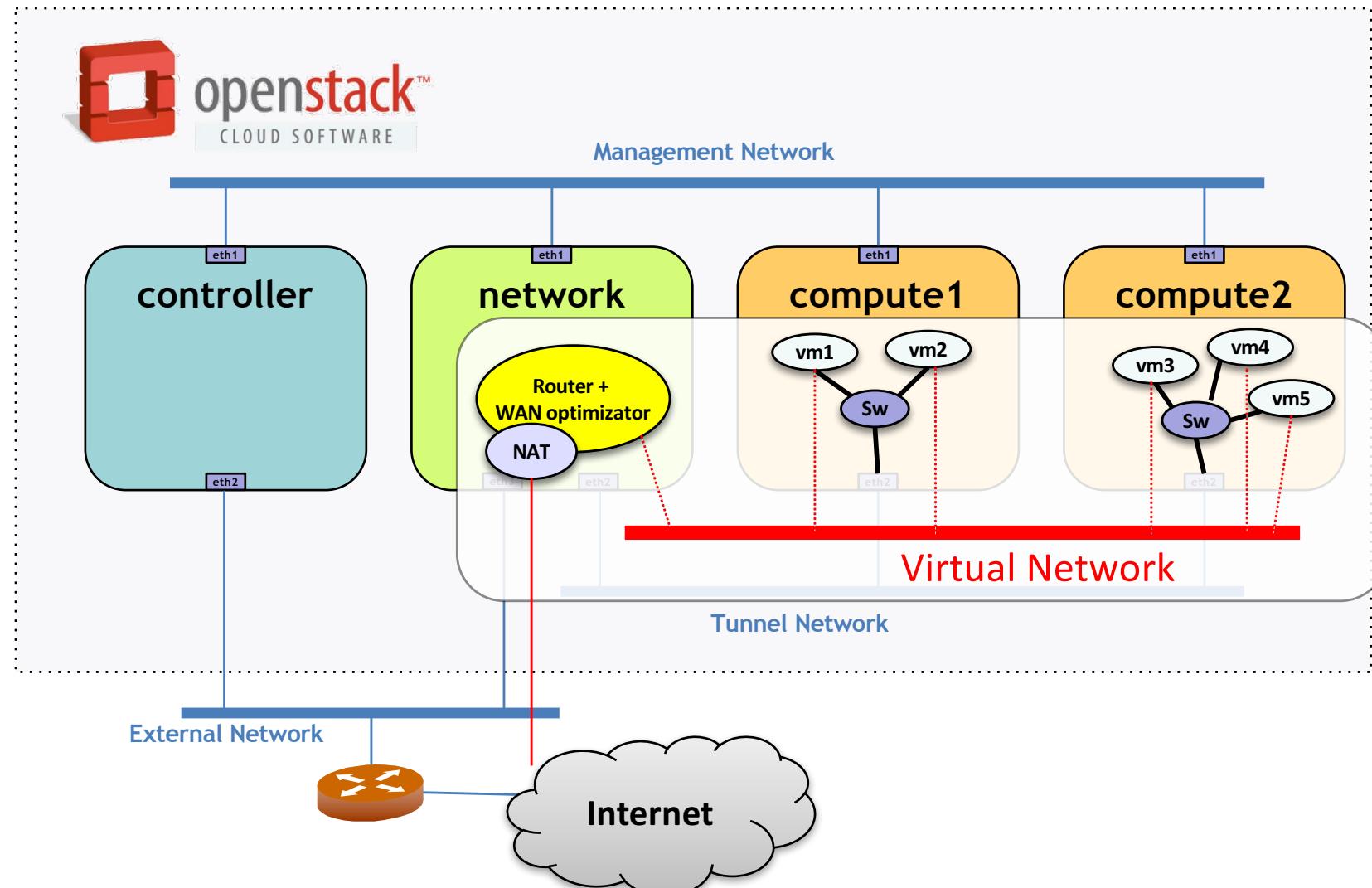


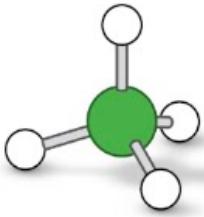
CENTER OPEN MIDDLEWARE





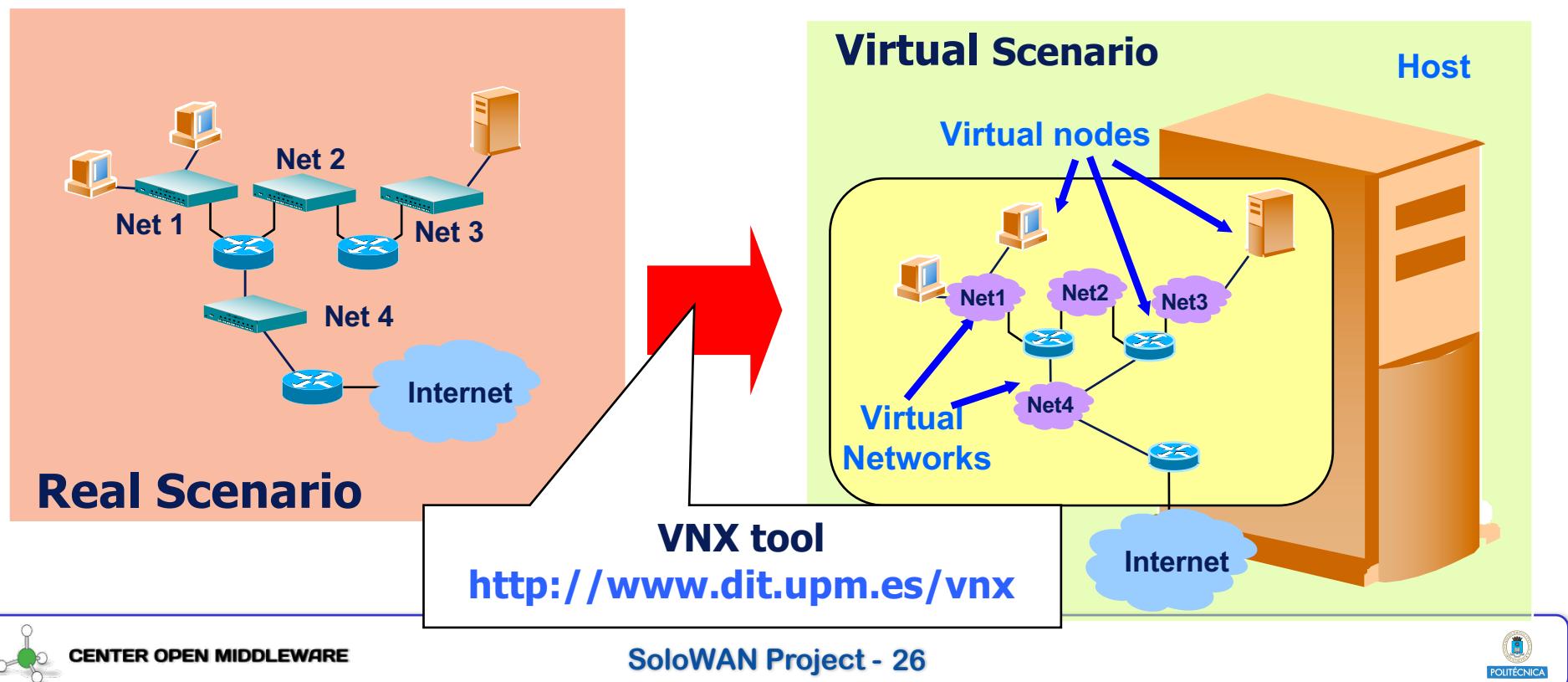
WAN Optimization as a Service

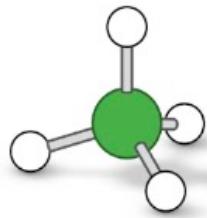




Virtual Testbeds

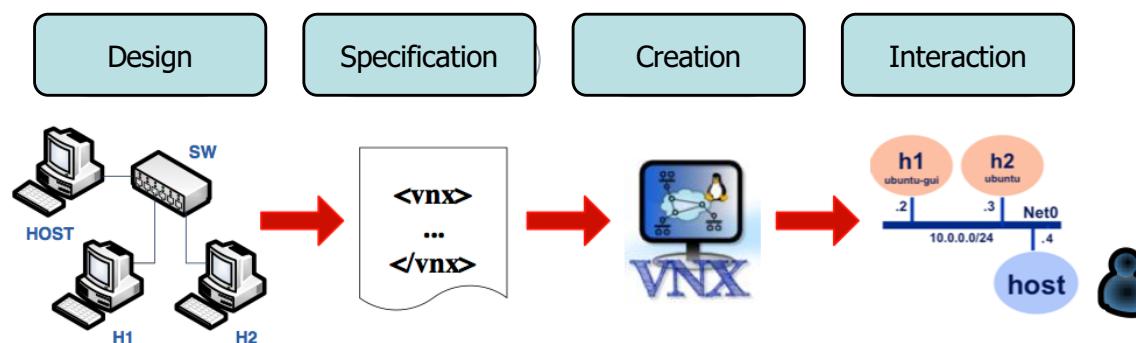
- Most SoloWAN testbeds developed using virtual testbeds based on VNX tool
- Combining VM and virtual network emulation on the host it is possible to run Virtual Network Scenarios





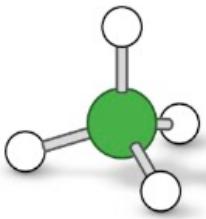
VNX: Virtual Networks over LinuX

- VNX is a general purpose open-source virtualization tool designed to help building virtual network testbeds automatically
- Characteristics:
 - Uses libvirt (Linux virtualization library)
 - Autoconfiguration for Linux, FreeBSD, OpenBSD and Windows
 - Support for Dynamips (CISCO) and Olive (Juniper)
 - LXC containers support
 - Open vSwitch support



<http://www.dit.upm.es/vnx>





SoloWAN Demo VM

- Ubuntu VM including SoloWAN virtual scenarios

SOLOWAN-v74 [Running]

h1 - console #1

```
File Edit View Search Terminal Help
Resolviendo h2 (h2)... 192.168.1.2
Conectando con h2 (h2)[192.168.1.2]:80... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 10485760 (10M)
Grabando a: 000/tmp/random1-10M.out000
100%[=====] 10.485.760 1,07MB/s en 9,4s
2015-07-15 02:19:00 (1,06 MB/s) - 000/tmp/random1-10M.out000 guardado [10485760/10485760]
]

-- Getting file random4-10M.out
2015-07-15 02:19:00 -- http://h2/random4-10M.out
Resolviendo h2 (h2)... 192.168.1.2
Conectando con h2 (h2)[192.168.1.2]:80... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 10485760 (10M)
Grabando a: 000/tmp/random4-10M.out000
66%[=====] 7.017.461 1,08MB/s T.E. 3s
```

solowan@solowan: ~/demo

```
File Edit View Search Terminal Help
-- SoloWAN test menu
fp_per_pkt 32
fps factor 4

Default values:
Proto://Server:Port
# of repetitions
Index filename
Filename

Scenario xml file: opennop-lxc-v3.xml

Select an option:
1) Get a file
2) Get files liste
3) Get a list of f
4) Change server n
5) Change server p
6) Change server p
7) Change number o
Type option number or 'q' to quit: 1
```

Solowan Dashboard - Mozilla Firefox

OpenNOP-SoloWAN de... x http://sta.../monitor/ x Solowan Dashboard x +

http://sta:8000

Solowan Dashboard

Real-time

Historical Data

Dashboard

Real-Time Optimizers:

- Optimizer 1: Decompressor
- Optimizer 2: Compressor
- Optimizer 1: Compressor
- Optimizer 2: Decompressor

Historical Data: Optimizer 1 - Decompression

Optimizer Decompression - by day

Historical Data: Optimizer 2 - Compression

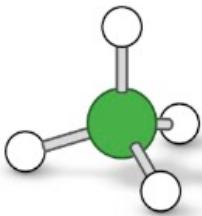
Optimizer Compression - by day

Taskbar:

- Solowan Dashboard
- o1 - console #1
- o2 - console #1
- h1 - console #1
- h2 - console #1
- wanem - console...
- sta - console #1
- [solowan@solowan...]
- [solowan@solowan...]

Available at: <https://goo.gl/YE6QXW>

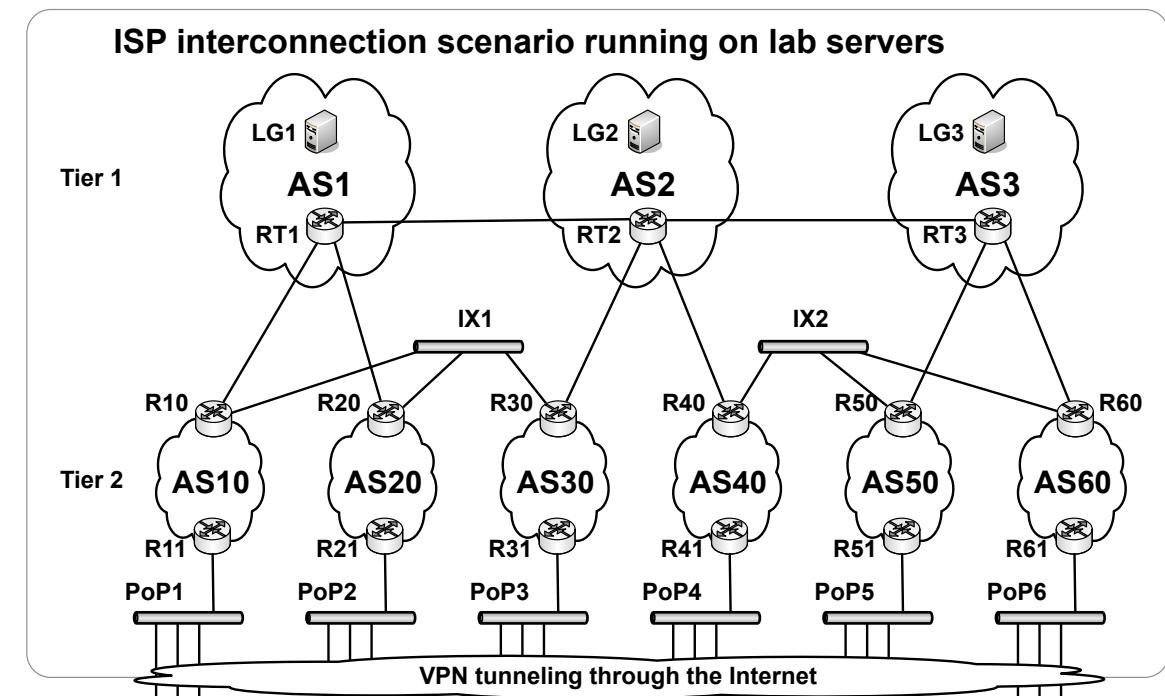
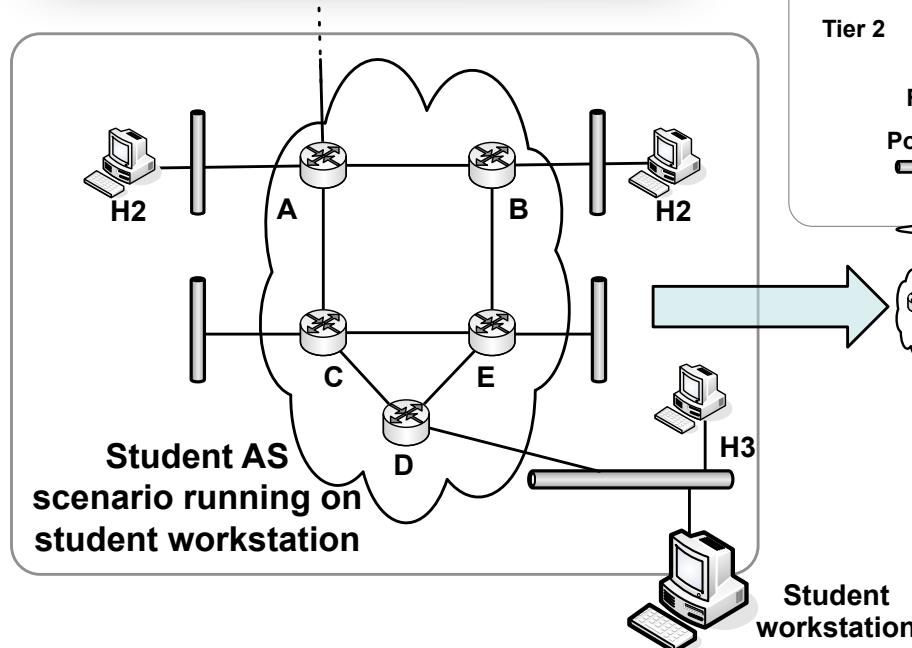


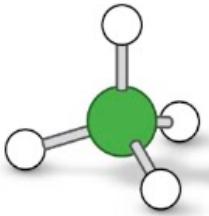


VNX Example scenario: Collaborative OSPF/BGP Practice

Requirements for ~400 students:

- Backbone: 15 dynamips routers, 5 servidores
- Client networks : 400×5 routers Linux with quagga routers, 4 PCs
- Switches
- Configuration management
 - 400 different configurations

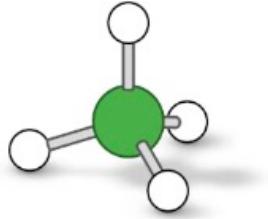




References

- **Center for Open Middleware (COM).**
 - <http://www.centeropenmiddleware.com/>
- **SoloWAN repository at Github.**
 - <https://github.com/solowan/solowan>
- **Virtual Networks over Linux (VNX).**
 - <http://vnx.dit.upm.es>
- **WOaaS Prof-of-concept implementation.**
 - <https://github.com/carlosv5/OaaS-network>





CENTER OPEN MIDDLEWARE

Universidad Politécnica de Madrid (UPM)



POLITÉCNICA

Thanks for your attention

Any questions?