October 2nd, 2018

Ecole Jeunes Chercheurs sur l'Efficacité Énergétique des Réseaux et Systèmes Distribués (E3RSD)

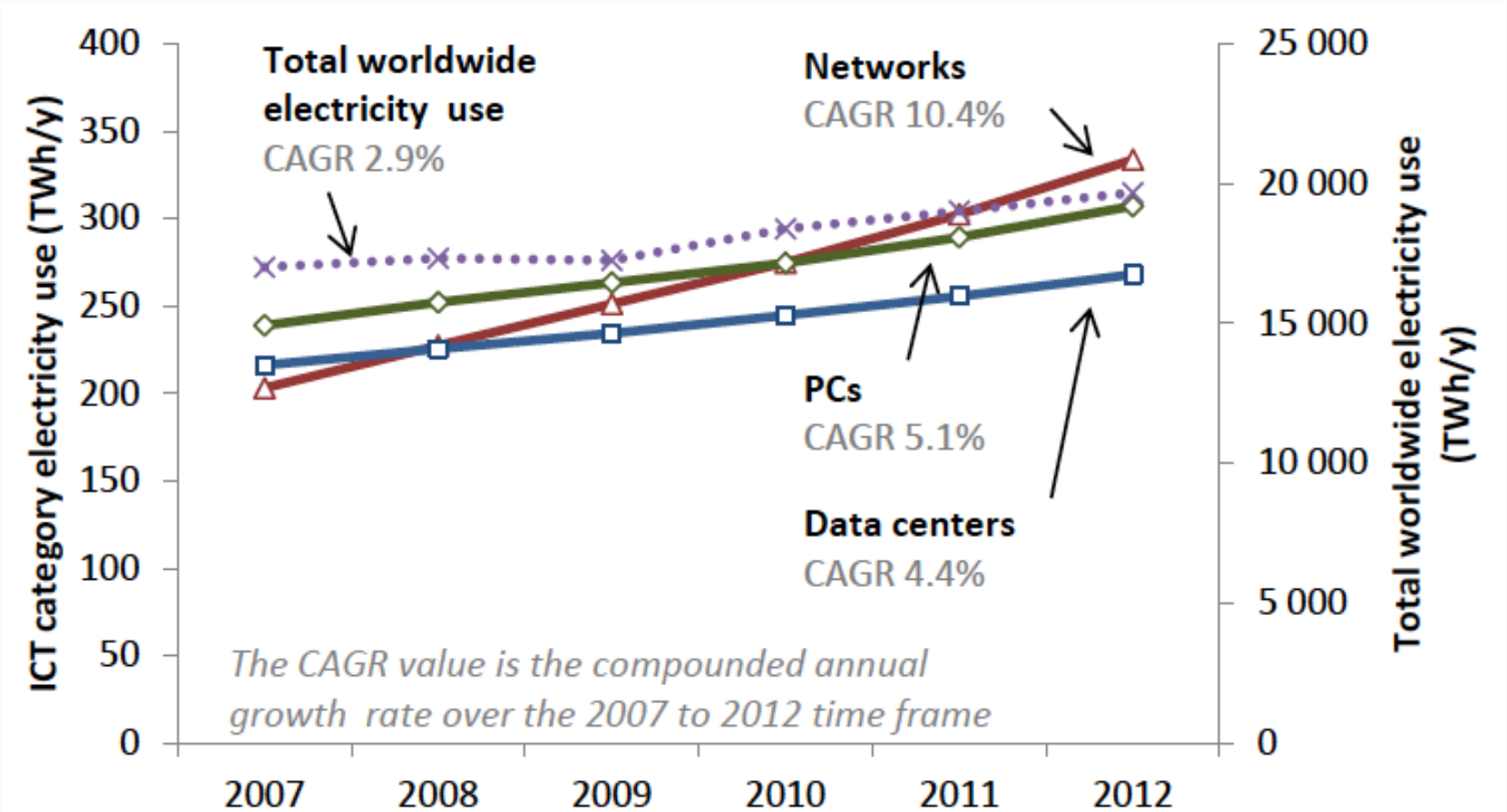
Towards Realistic Devices Shutdown Strategies in ISP and Data Center networks

Dino Lopez Pacheco
dino.lopez@univ-cotedazur.fr

SigNet team
Université Côte d'Azur, CNRS, I3S, France

Introduction & Context

The Worldwide ICT Energy Consumption



Estimated energy consumption of the ICT between years 2007 and 2012, from [Van Heddeghem *et al.* 2014]

Energy optimization through shutdown solutions

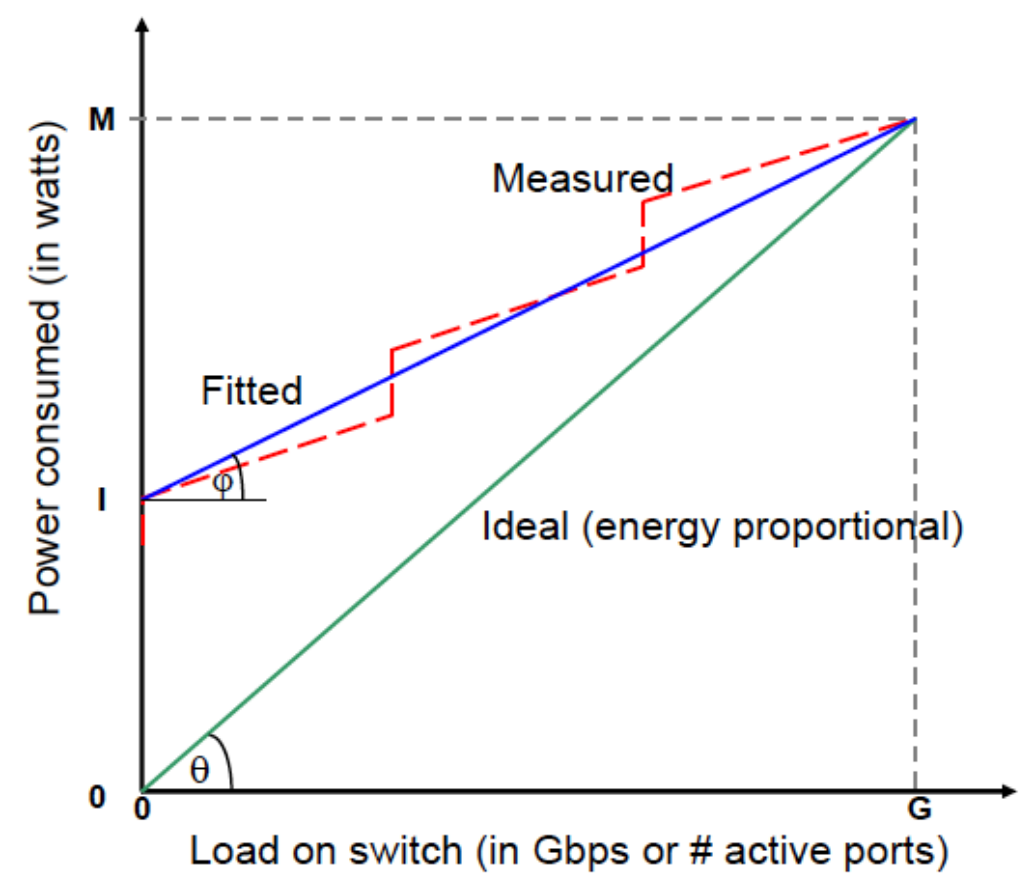
- Networks are dimensioned to support peak loads $\rightarrow$ underutilization of resources
- Tackling the energy problem on both ISP and DC networks with shutdown strategies
  - ISP: Shutdown (sleep) routers $\rightarrow$ optimal usage of resources
  - DC: Shutdown idle VMs $\rightarrow$ massive consolidation
- Why not Energy Proportionality-based solutions?
  - Low Energy Proportionality Index (EPI)

Going from idle to active might need considerable power!!

Introduction & Context

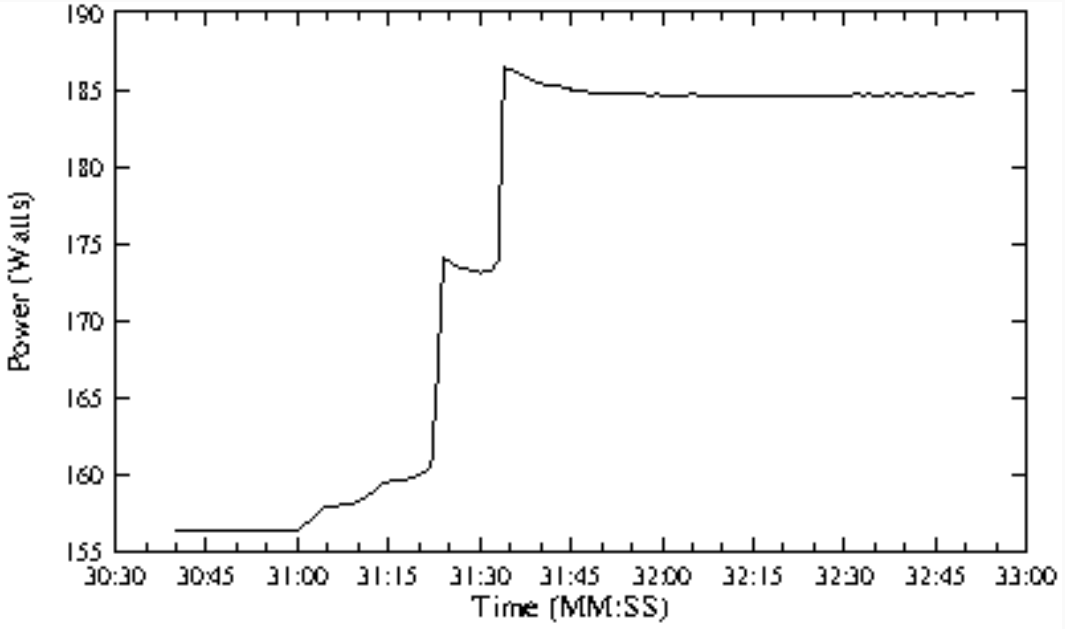
Energy Proportitonality and Energy Proportionality Index

Network Energy Proportionality, from [Mahadevan et al. 2009]



- $EPI = (M - I)/M$
- EPI of 25% for Edge switches and near 0% for core switches

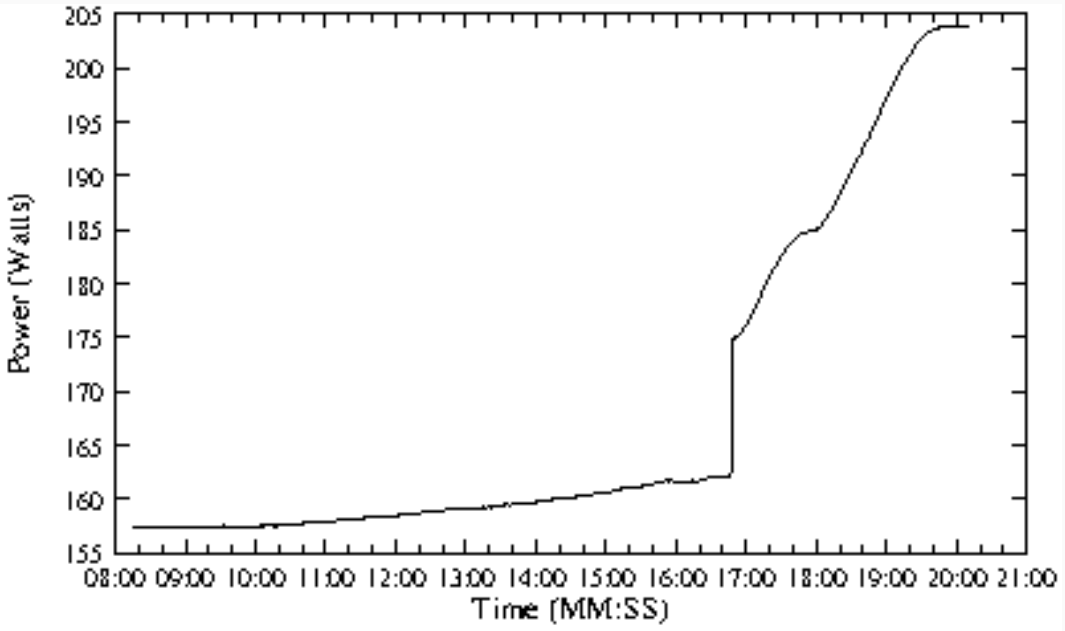
Idle $\rightarrow$ Active - Our experiences (1)



Turning on one module every 10s, all ports at once.

- HP 5400zl switch
- 4 modules, 24 GigaEthernet ports per module
- $I_{pdis} = 158 Watts$, $M = 210 Watts$, $EPI \approx 25\%$

Idle $\rightarrow$ Active - Our experiences (2)



Turning on one port every 5s.

- HP 5400zl switch
- 4 modules, 24 GigaEthernet ports per module
- $I_{pdis} = 158 Watts$, $M = 210 Watts$, $EPI \approx 25\%$

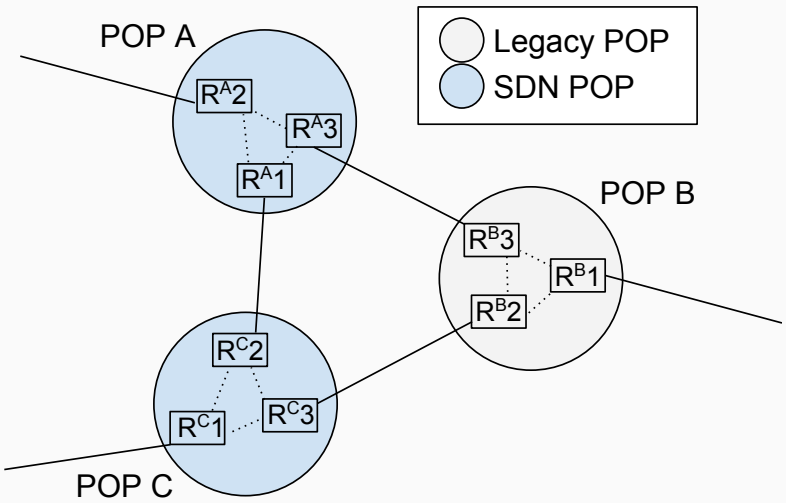
Shutting Routers Down in ISP Networks

Energy Optimization in Networks

- Energy Aware Routing [Rouzic *et al.* 2013; De Rango *et al.* 2012; Shang *et al.* 2010]
- Energy Aware Routing + Software Defined Networks [Giroire *et al.* 2014; "ElasticTree: Saving Energy in Data Center Networks."]
- Hybrid Software Defined Networks [Vissicchio *et al.* 2014; Agarwal *et al.* 2013]

Energy Models and Hybrid Network Models

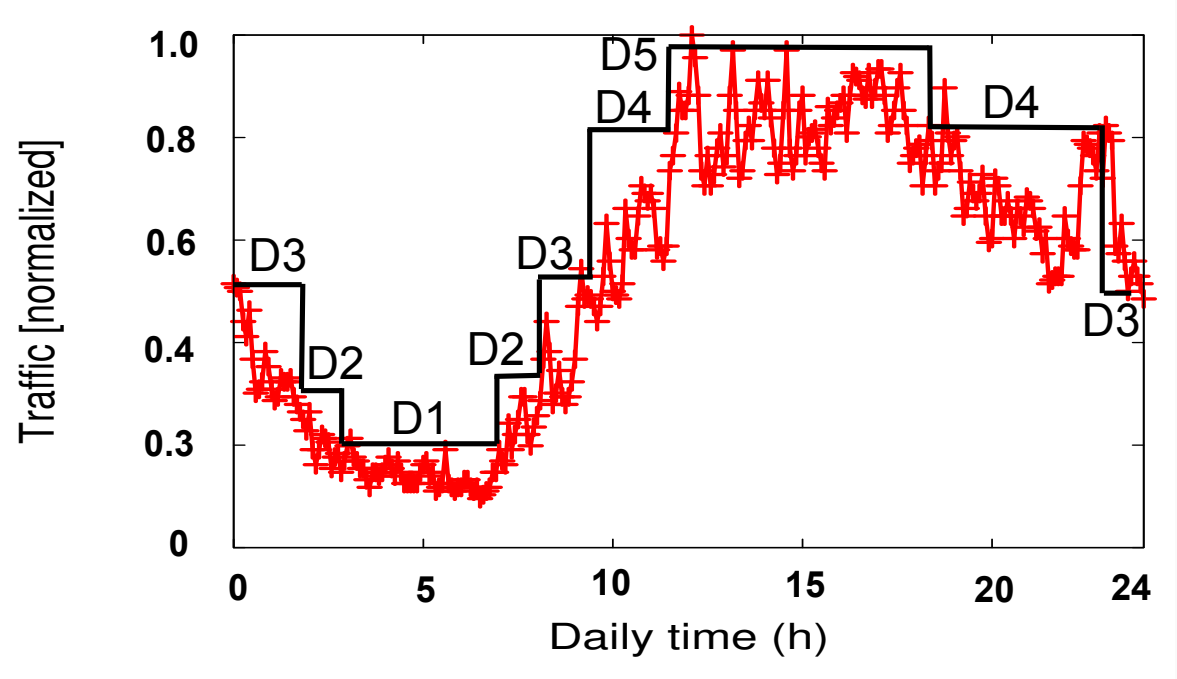
- Full SDN POPs and legacy POPs
- POP networks $\rightarrow$ full mess topology
- turning off routers in SDN POPs only



- Link power model between 2 hosts u and v [Idzikowski *et al.* 2016]
  - $P_{l(u,v)} = x_{u,v}U_{u,v} + F_{u,v}L_{u,v}$
- Forwarding node power model
  - $P_{n(u)} = B_u + A_u + \sum_{v \in N^+(u)} P_{l(u,v)}$

Network Traffic Observations

- Traffic roughly stable over time
  - stable pattern per day, per week.
- A few configurations are enough to adapt the network resources to the spared amount traffic
  - No need for frequent network reconfigurations



Shutting Routers Down in ISP Networks

The Smooth ENergy Aware Routing (SENAtoR) Solution

SENAtoR: Off-line side

Goal: Maximize the number of links to be turned off

- Limited budget $\Longrightarrow$ not all POPs can be SDN-capable
- SDN POPs placement
  - chose the k first POPs with the highest degree
- Off link selection + tunnels for fast rerouting
  - inter POP links only $\Longrightarrow$ 1 SDN device
  - For every link in a SDN POP
    - mark the link as disabled and reroute the traffic
    - if the traffic can be rerouted, set the link as *removable*
  - For i in 1 up to $l^r_{u,v}$ *removable* links
    - declare the first i removable links to be turned off
    - create a tunnel between the concerned SND POP and the closest POP allowing to reach the traffic destination

SENAtoR: On-line side

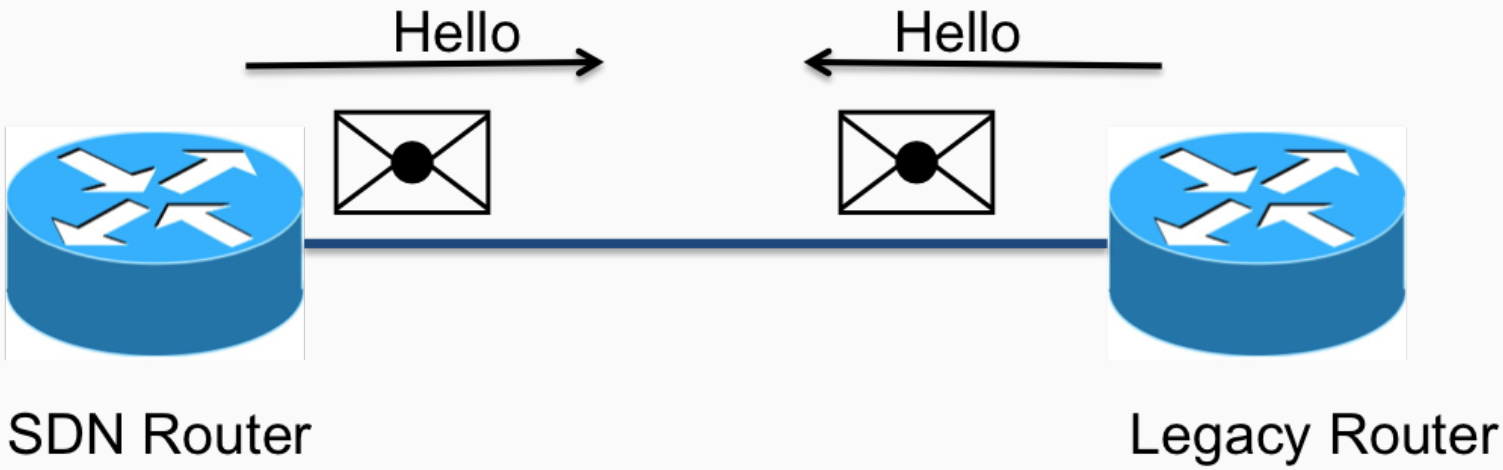
Goal: Turn off links, put routers in sleep mode and deal with unexpected network dynamics

- Link failure detection $\rightarrow$ sudden traffic decrease
  - Turn on all routers and links if $E_i(t) \leq \beta * E_i^{ES}(t)$, $\beta = 0.5$
- Traffic spike mitigation
  - Turn on all routers and links if $E_i(t) \geq \alpha * E_i^{ES}(t)$, $\alpha = 1.5$
- Safe link shutdown
  - Reroute traffic through tunnels (during legacy routing protocol convergence)
  - Trigger legacy routing convergence before actual turn off

Triggering Legacy Routing Protocol Convergence

Goal: Smooth Router Shutdown

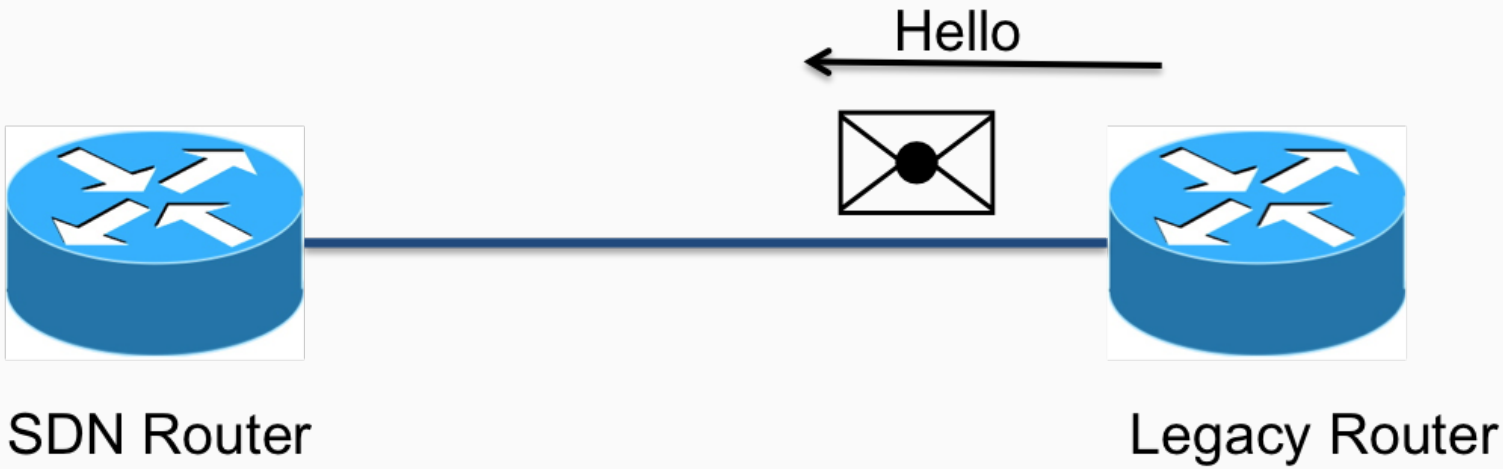
- Neighbor discovery through HELLO packet exchanges
- Link/Network failures detected after missing HELLO packets



Triggering Legacy Routing Protocol Convergence

Goal: Smooth Router Shutdown

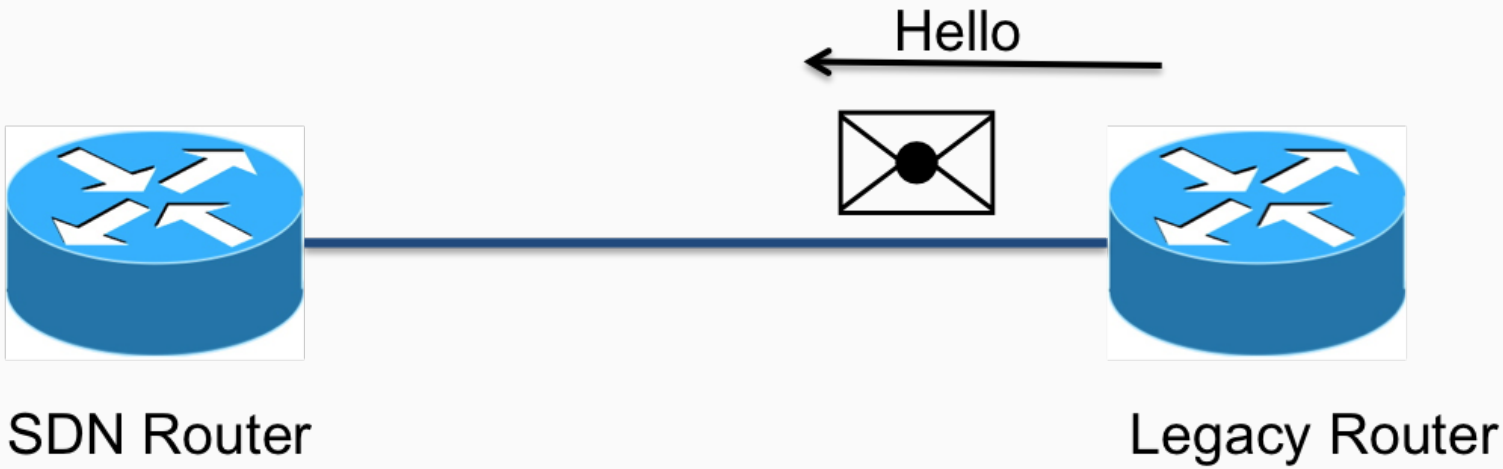
- Neighbor discovery through HELLO packet exchanges
- Link/Network failures detected after missing HELLO packets
  - SDN routers stop sending HELLO packets



Triggering Legacy Routing Protocol Convergence

Goal: Smooth Router Shutdown

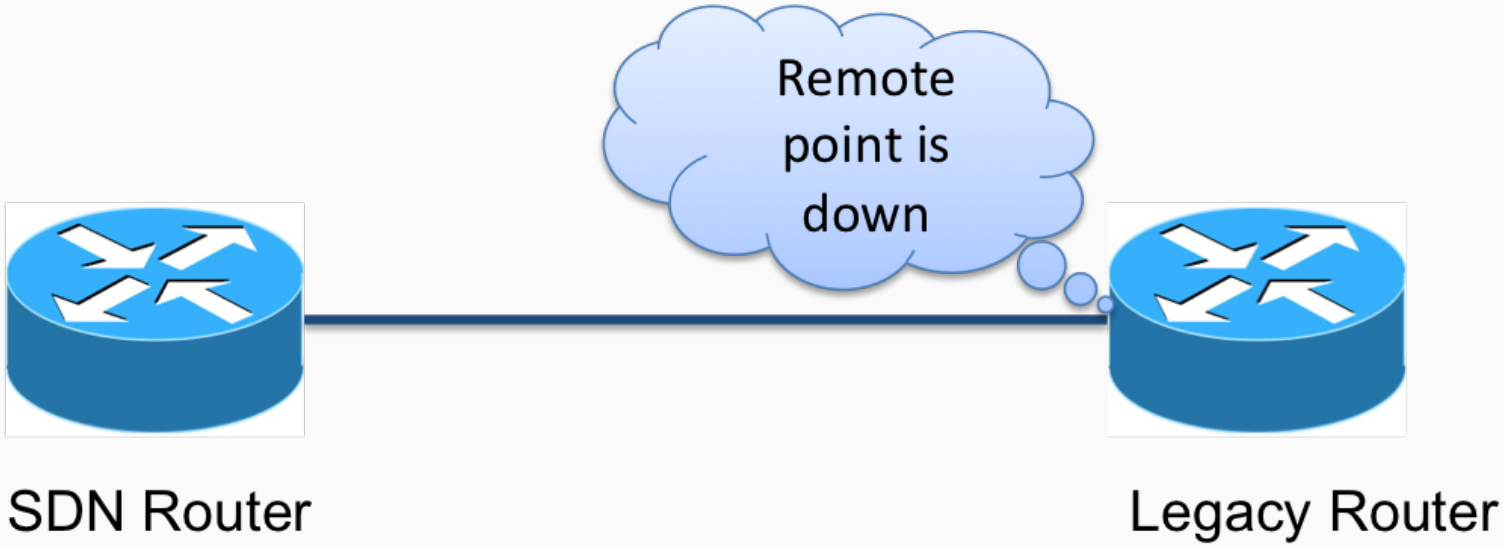
- Neighbor discovery through HELLO packet exchanges
- Link/Network failures detected after missing HELLO packets
  - SDN routers stop sending HELLO packets
  - Forward any incoming packet



Triggering Legacy Routing Protocol Convergence

Goal: Smooth Router Shutdown

- Neighbor discovery through HELLO packet exchanges
- Link/Network failures detected after missing HELLO packets
  - SDN routers stop sending HELLO packets
  - Forward any incoming packet
  - Turn off a port and router when no more traffic is received



Shutting Routers Down in ISP Networks

Evaluation

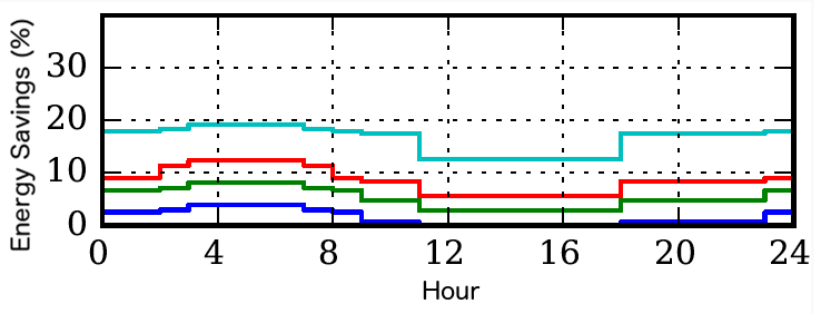
Numerical Evaluation

- Tests on different network size from the SDNLib [Orlowski *et al.* 2007] database
  - *atlanta* - 15 nodes and 33 links
  - *germany50* - 50 nodes and 88 links
  - *zib54* - 54 nodes and 81 links
  - *ta2* - 65 nodes and 108 links
- Several parameters
  - Stretch ratio
  - Energy savings
  - and more

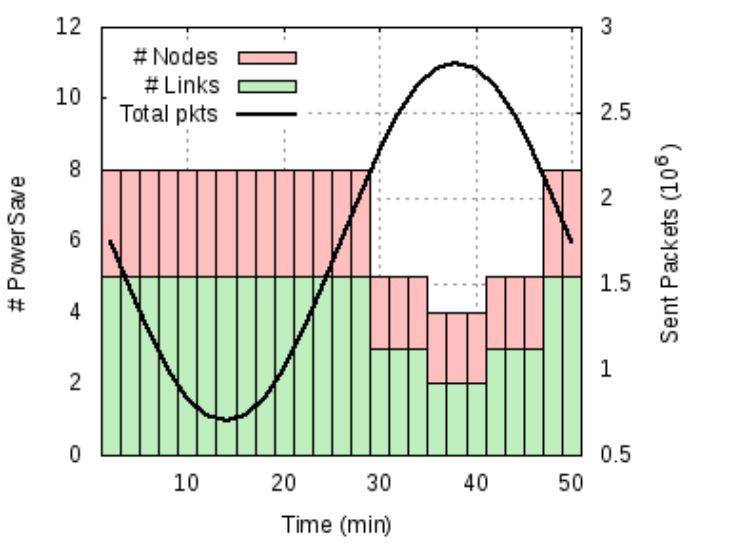
Experimental evaluation on *atlanta*

Experimental Evaluation: Energy Savings

Realistic Traffic Pattern $\Longrightarrow$ realistic energy savings

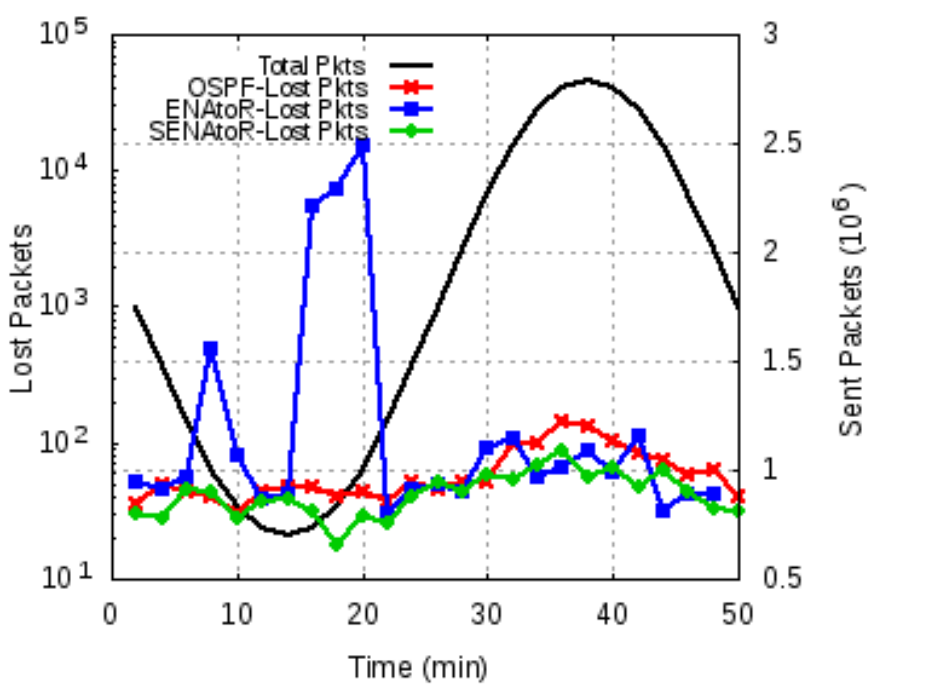


Sinusoidal Traffic Pattern $\Longrightarrow$ Impact of traffic on the number of inactive nodes and links



Experimental Evaluation: Packet Losses

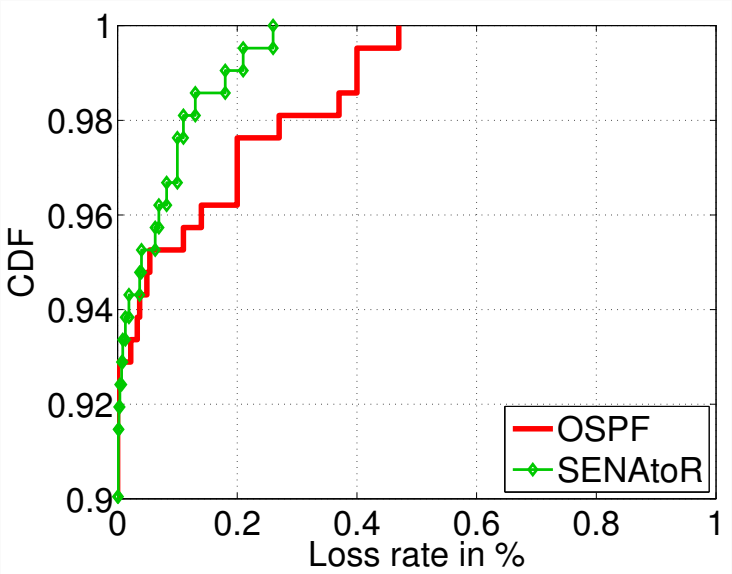
- OSPF - No device shutdown - baseline
- ENAtoR - Router shutdown - No smooth extinction
- SENAtoR - Smooth router extinction



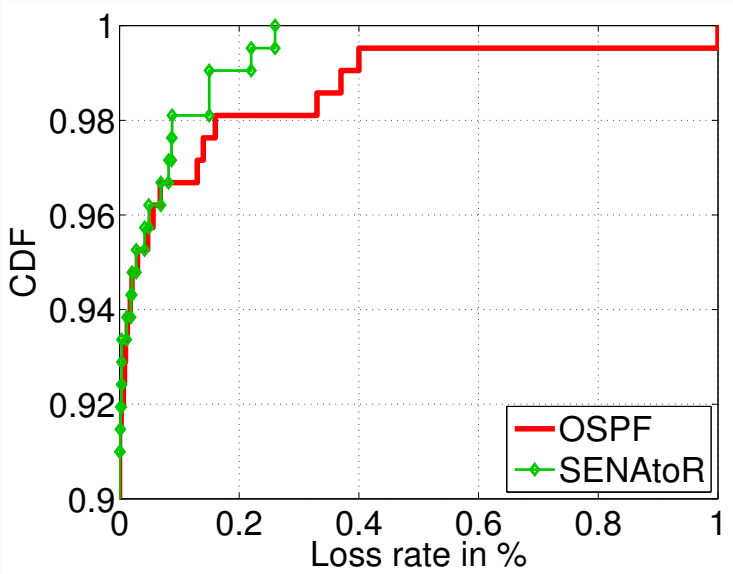
Takeaway: SENAtoR = OSPF without disabled devices!

Experimental Evaluation: Traffic Spikes

Spike at a OSPF – SDN link



Spike at a OSPF – OSPF link



- SENAtoR successfully handle sudden traffic increases
  - Re-enabling links and routers
  - Fast load balancing

Takeaway: SENAtoR outperforms OSPF even in presence of disabled devices!

Shutting Routers Down in ISP Networks

Conclusions

SENAtoR for real networks

- Designing SENAtoR $\rightarrow$ Real network constraint
- SENAtoR can deliver complex green solutions to reality
  - Sleep mode for routers
  - Link disabling
- SENAtoR's green services does not degrade the Quality of Service offered by current networks

SENAtoR for real networks

- Designing SENAtoR $\rightarrow$ Real network constraint
- SENAtoR can deliver complex green solutions to reality
  - Sleep mode for routers
  - Link disabling
- SENAtoR's green services does not degrade the Quality of Service offered by current networks
  - Can do better

SENAtoR for real networks

- Designing SENAtoR $\rightarrow$ Real network constraint
- SENAtoR can deliver complex green solutions to reality
  - Sleep mode for routers
  - Link disabling
- SENAtoR's green services does not degrade the Quality of Service offered by current networks
  - Can do better

Nicolas Huin, Myriana Rifai, Frédéric Giroire, Dino Lopez, Guillllaume Urvoy-Keller and Joanna Moulierac. *Bringing Energy Aware Routing closer to Reality with SDN Hybrid Networks*. To appear in IEEE Transactions on Green Communications and Networking. May 2018.

Massive Resources Consolidation in DCN

Smart VM Placement

Virtual Machines placement to globally optimize the resource utilization and decrease costs:

- Server Consolidation
- Server Reconsolidation
  - Cold/Hot MV migration

Idle virtual machines in the cloud

Presence of Idle VMs has been reported in [Koomey *et al.* 2017]

- At least 30% of VMs are in an idle state
  - Long running services like mail or web servers [Zhang *et al.* 2016]
  - Testing VMs instantiated and rarely shut down

The Problem

Idle VMs lock the physical resources assigned

- Instances need to serve sudden requests $\Longrightarrow$ VMs cannot be powered off
- Memory overcommitment $\Longrightarrow$ swapping $\Longrightarrow$ Performance degradation
- Memory is wasted and new tasks cannot be instantiated

The Challenge

Idle machines can be identified as in [Franzini, 2012]

- Indicators: *Network transfer*, *disk R/W*, *CPU*, *page dirtying rate*

What after? How to release resources?

- Suspended to RAM the VM to release *CPU*, but not *RAM*
- Suspended to Disk the VM to release *CPU*, and *RAM*

This is service disruptive and might hurt users' Quality of Experience!

Massive Resources Consolidation in DCN

Related Work

Related Work

Application proxies: DreamServer [Knauth *et al.* 2014]

$\Longrightarrow$ Each application protocol needs a custom proxy implemented

Redesigning the platform: Picocenter [Zhang *et al.* 2016]

$\Longrightarrow$ Incompatible in scenarios where users want IaaS and VMs

Remote memory: Oasis [Zhi *et al.* 2016]

$\Longrightarrow$ Specific hardware (low power mode) and patches for the hypervisor (partial VM migration)

Snapshoting: CloudGC [Zhang *et al.* 2017]

$\Longrightarrow$ Snapshot creation of VMs will increase IO operations and from disk restoring might be too slow

Massive Resources Consolidation in DCN

Our Solution: SEaMLESS

Our Solution: SEaMLESS

SEaMLESS key idea: transform a fully fledged idle VM in a resourceless lightwieght VNF.

SEaMLESS pros

- Release resources locked by idle VM instances
- Full transparency and generic application support
- Fast response time upon user activity

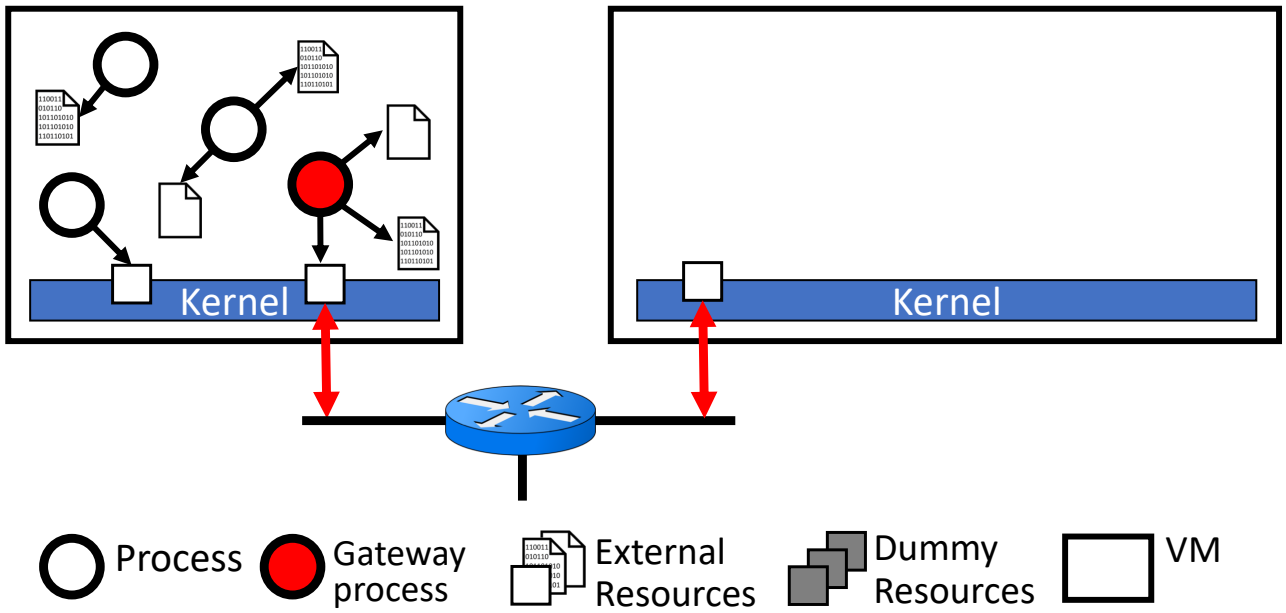
SEaMLESS comes as a

- *User-space solution* (Linux) compatible with existing technologies
- *No patch* for the VM *kernel*
- *No patch* for the *hypervisor* (based on KVM/QEMU)

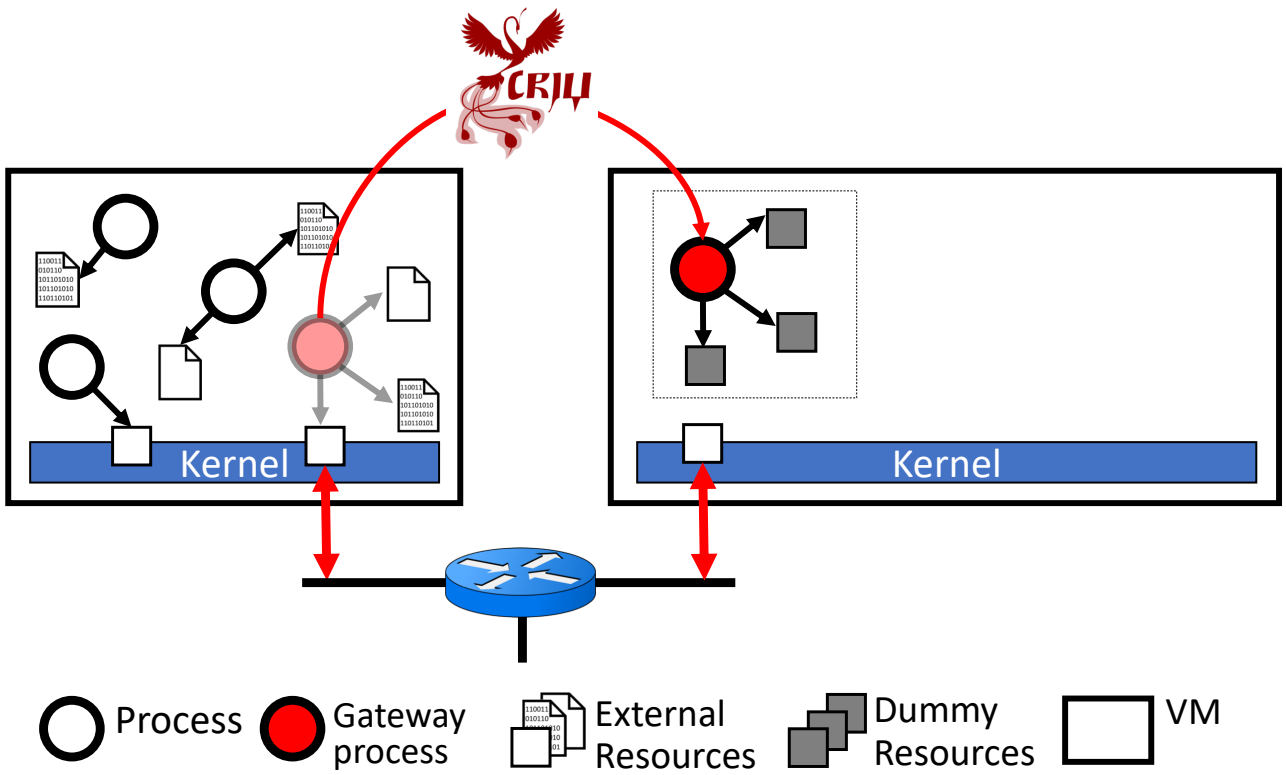
Creating lightweight resourceless VNFs

Gateway Process

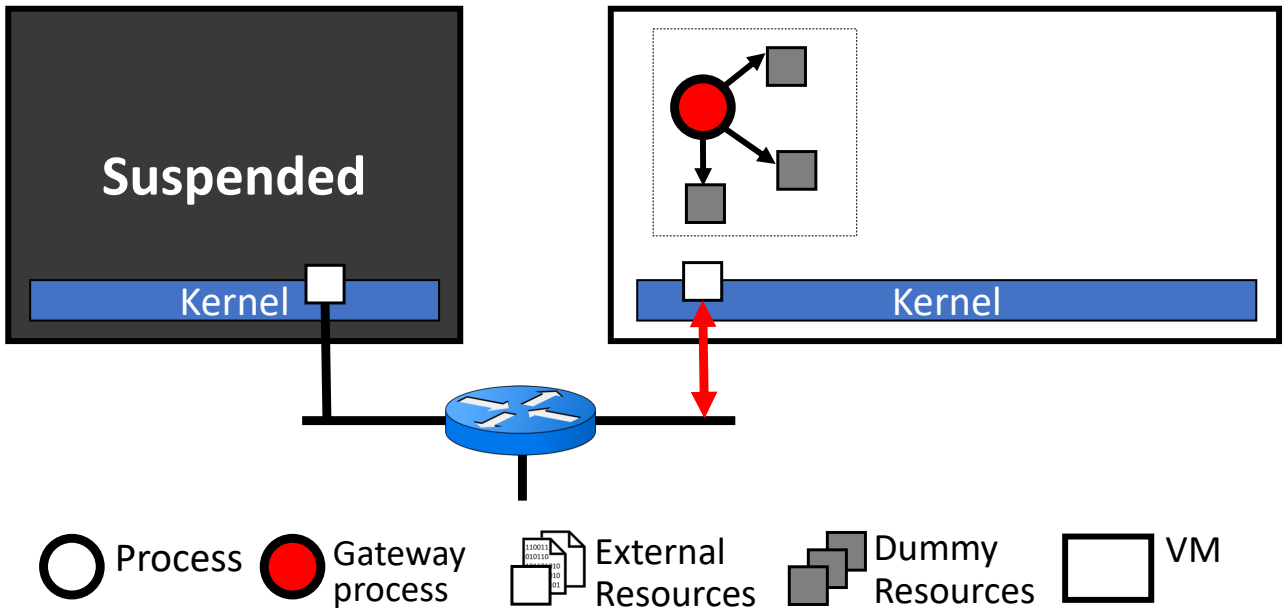
- Processes in a VM accepting end-user' requests
  - Web server, SSH server, FTP server, etc.
- Entry points for the VM



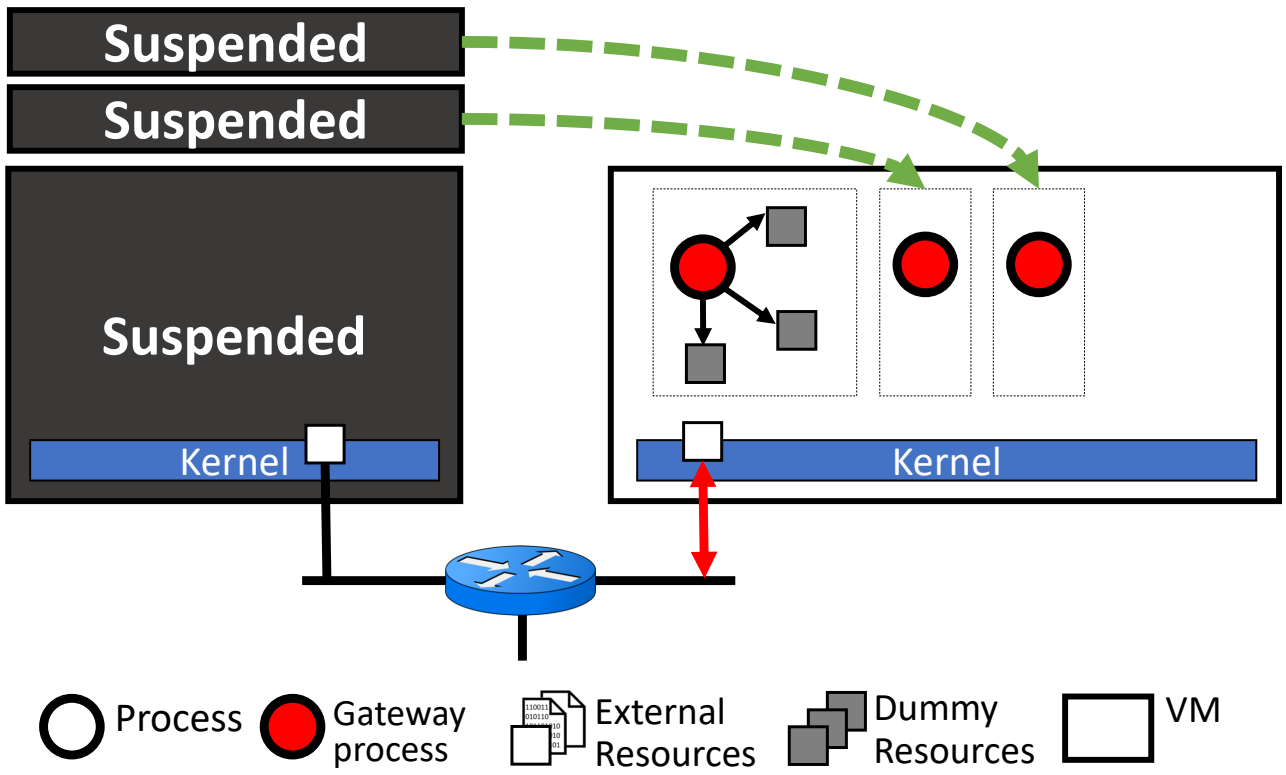
Creating lightweight resourceless VNFs



Creating lightweight resourceless VNFs

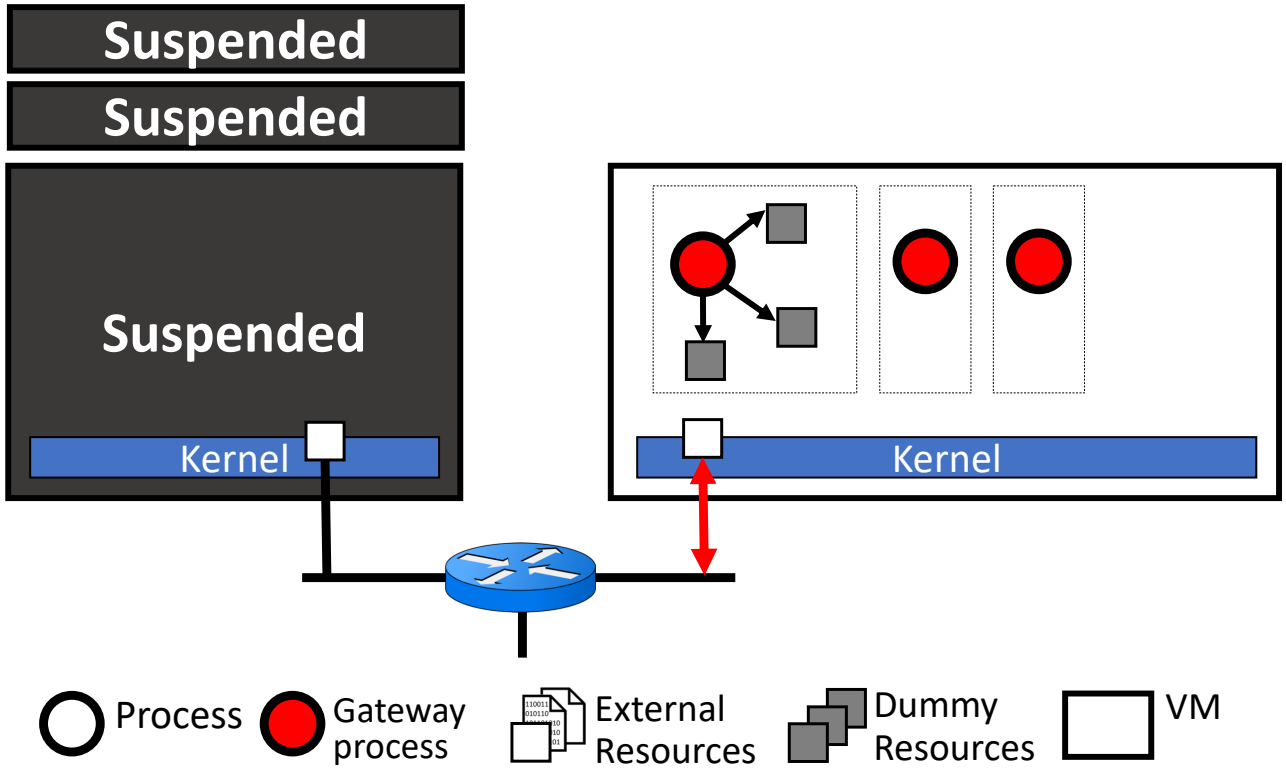


Creating lightweight resourceless VNFs



Creating lightweight resourceless VNFs

- Idle VMs transformed into resourceless VNFs (Sink VNFs)
- Hundreds of Sink VNFs consolidated on the Sink Server



Massive Resources Consolidation in DCN

Permanent Service Availability

User Activity Detection

Not all the requests need to resume the VM

- Trivial requests can be replied by the VNF
  - Network-, Transport- and Application-Layer KeepAlive messages
  - Network Control packets
- Non-trivial requests, Gateway Process has to be inside the VM

How to detect non-trivial user requests?

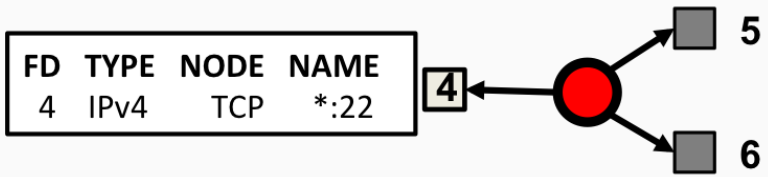
- Gateway Processes interact with resources (read/write or open/close) via *System calls (Syscalls)*
- Monitor syscalls to detect users' activity

If `write()`, `read()`, `open()`, `etc.` on a resource outside the VNF
$\Longrightarrow$ **Resume the VM**

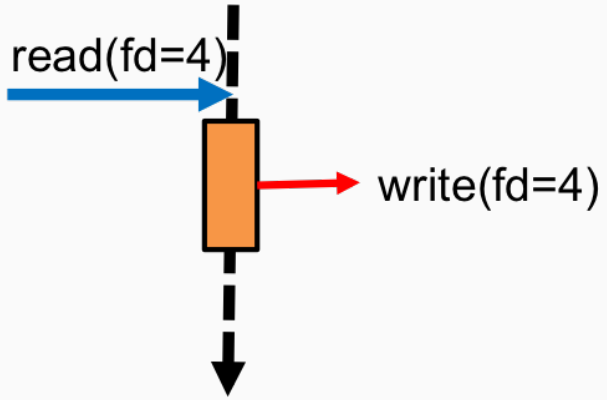
User Activity Detection

Classical Gateway Process at a Sink Container

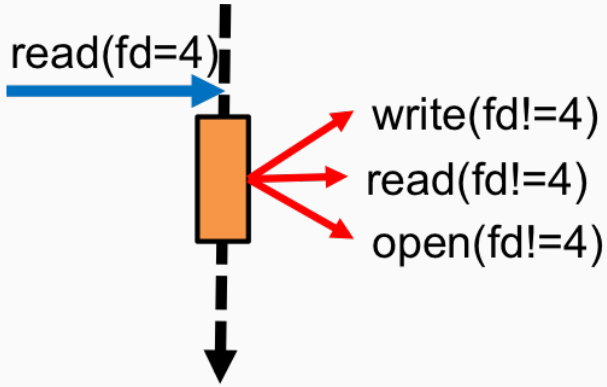
- Gateway Process with a network socket (syscalls on $fd = 4$)
- Other file descriptors pointing to empty/dummy resources



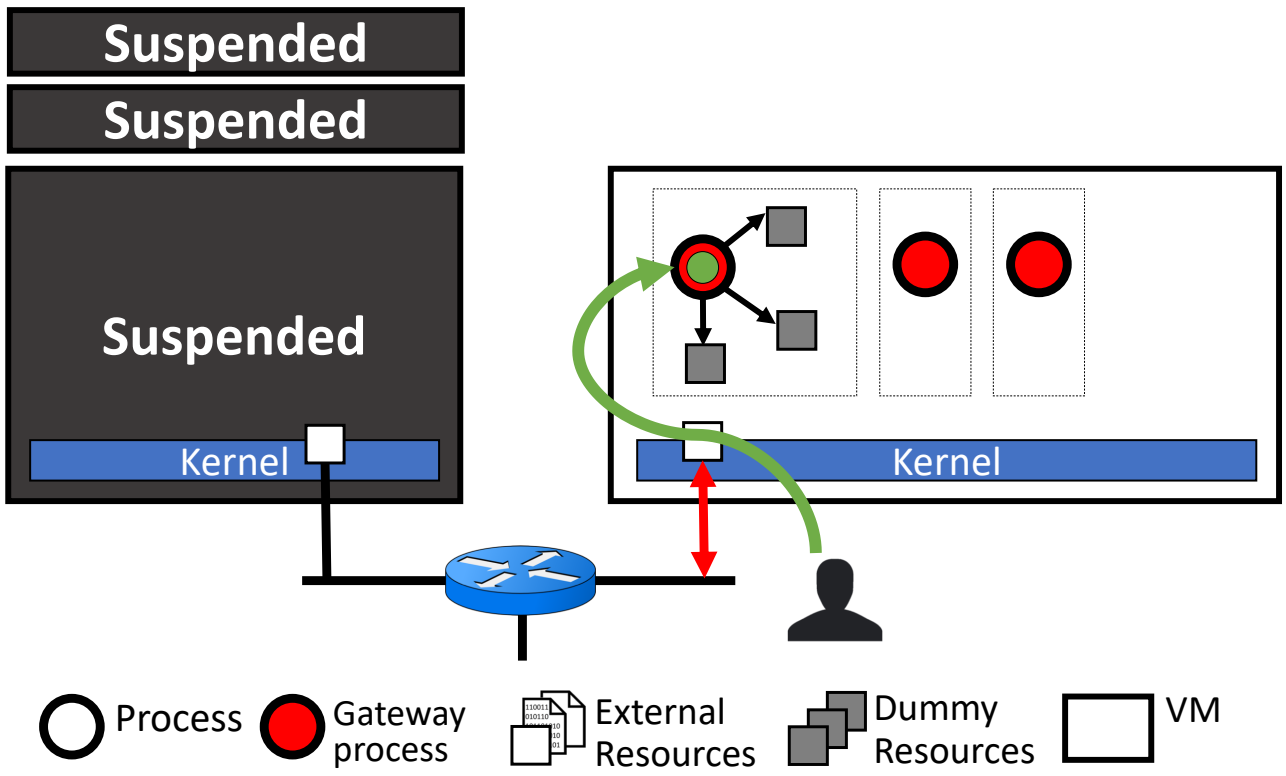
Trivial user request (e.g. KeepAlive)



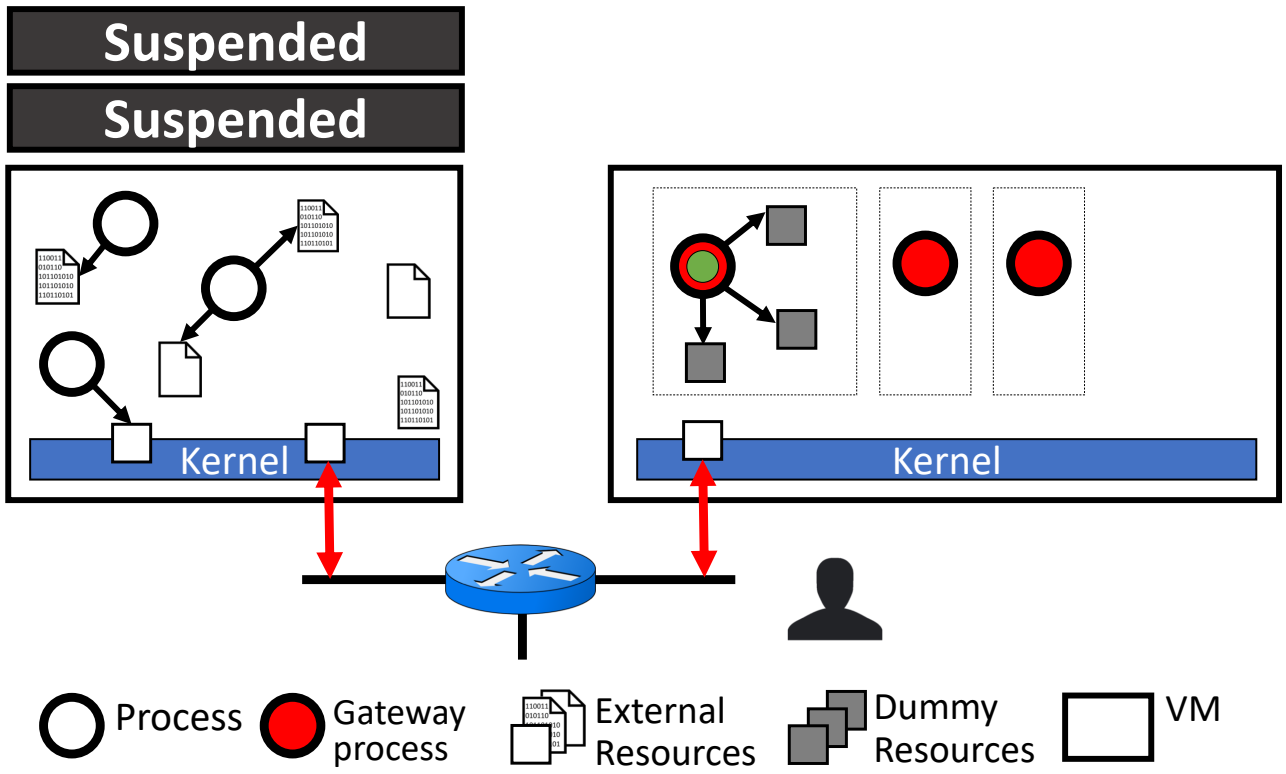
Non-Trivial User request



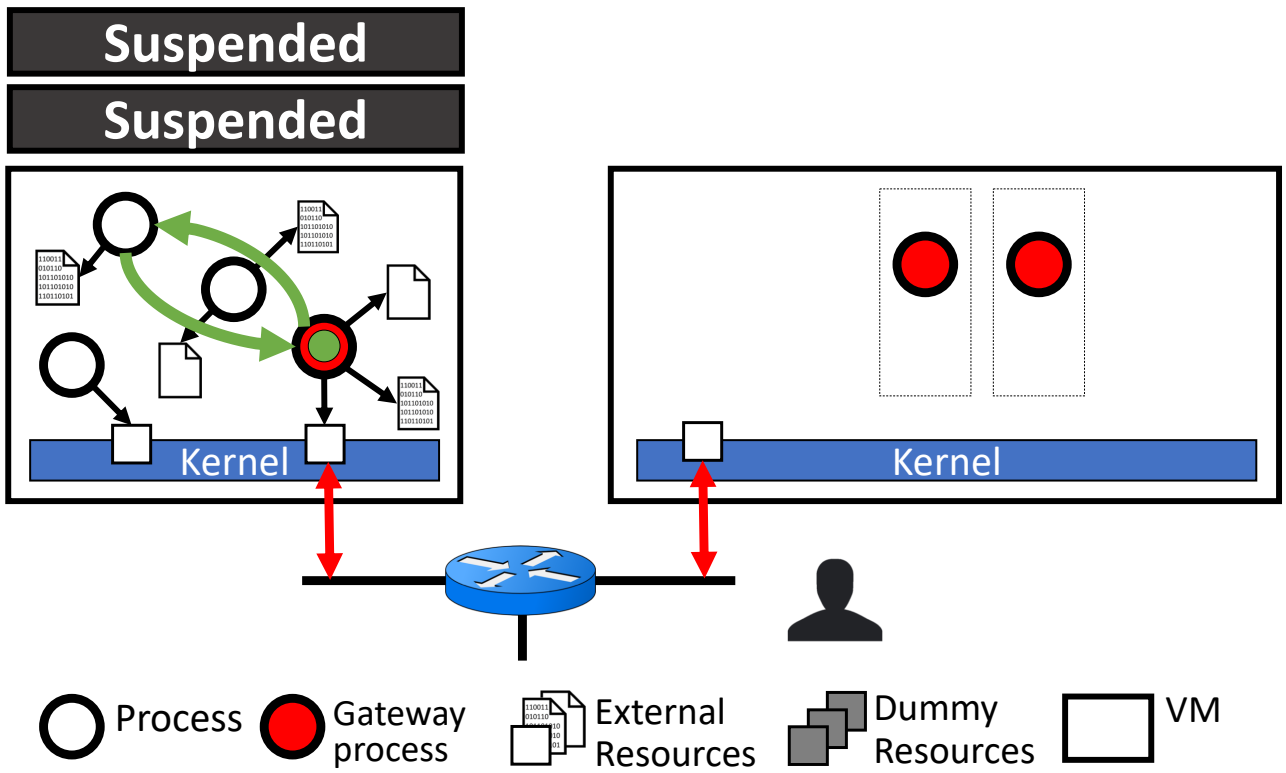
Replying to Users' Requests



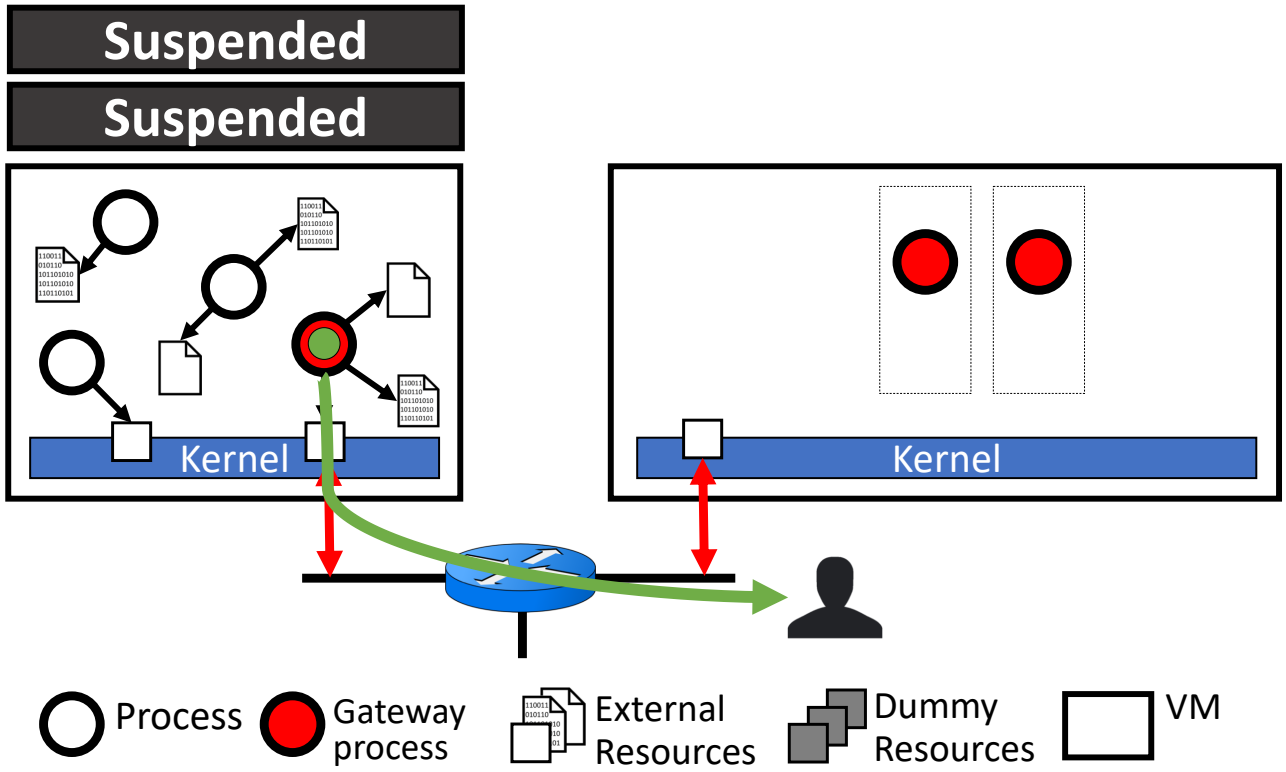
Replying to Users' Requests



Replying to Users' Requests



Replying to Users' Requests



Massive Resources Consolidation in DCN

Resource Releasing and Fast Service Restoration

Release the Resources

Different strategies to reclaim the resources allocated by a VM

Suspend to RAM

+ Fast
- Only CPU released

Suspend to Disk

+ RAM and CPU released
- Size dependent

Release the Resources

Different strategies to reclaim the resources allocated by a VM

Suspend to RAM

+ Fast
- Only CPU released

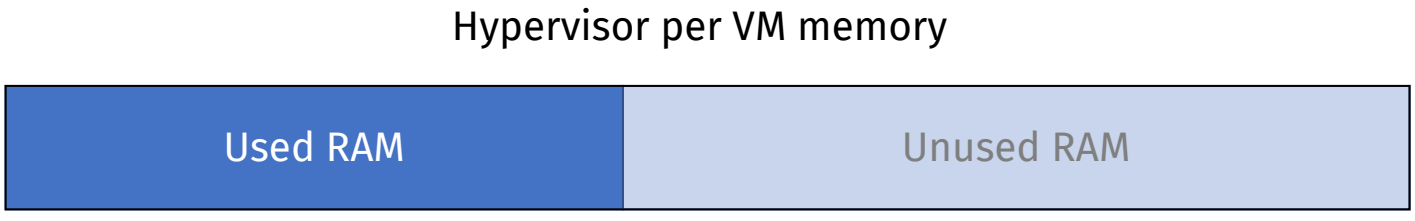
Suspend to Disk

+ RAM and CPU released
- Size dependent

Suspend to Swap

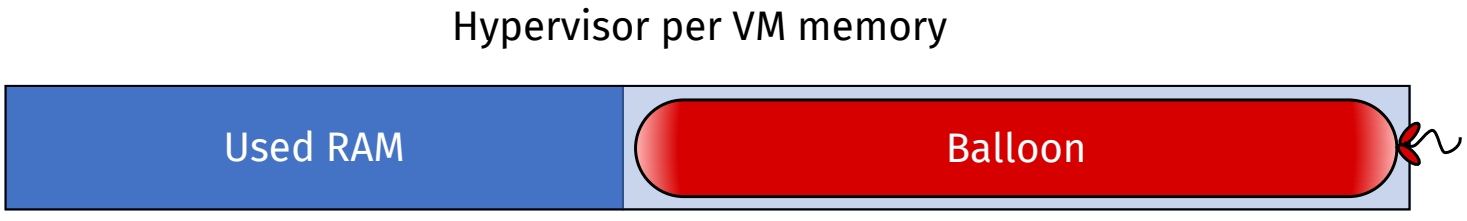
+ Immediate restart
+ Size independent
+ releases CPU
+ releases most of RAM

Suspending to Swap

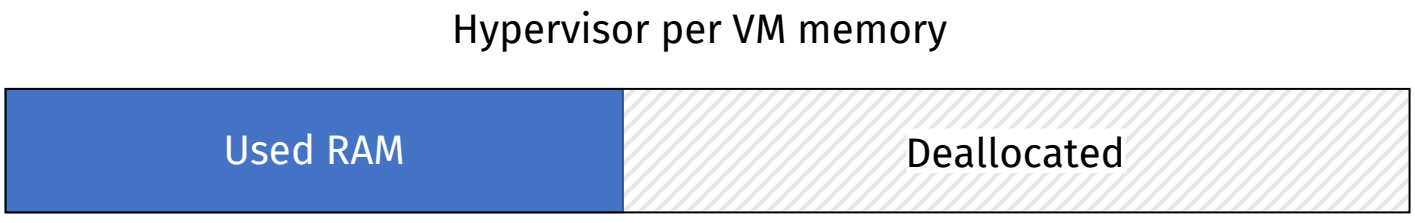


Suspending to Swap

Ballooning to retrieve free space

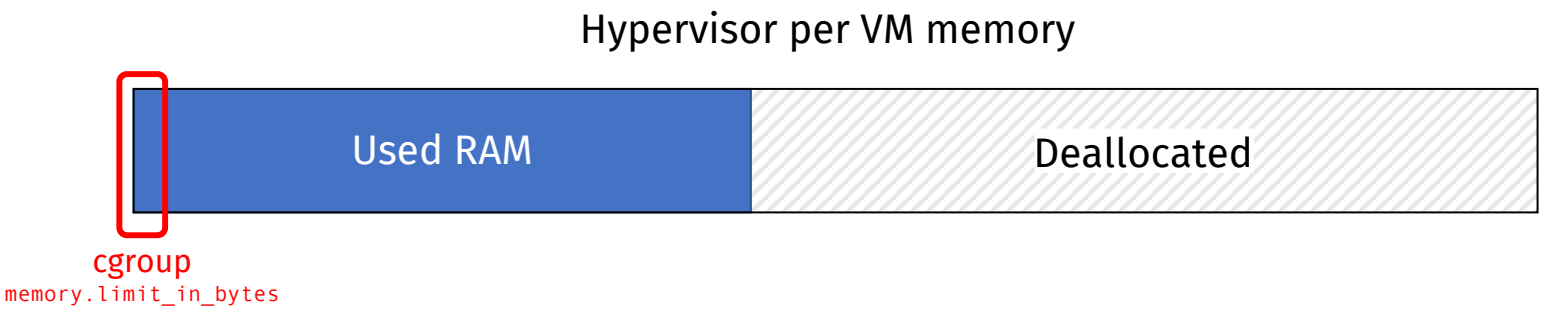


Suspending to Swap



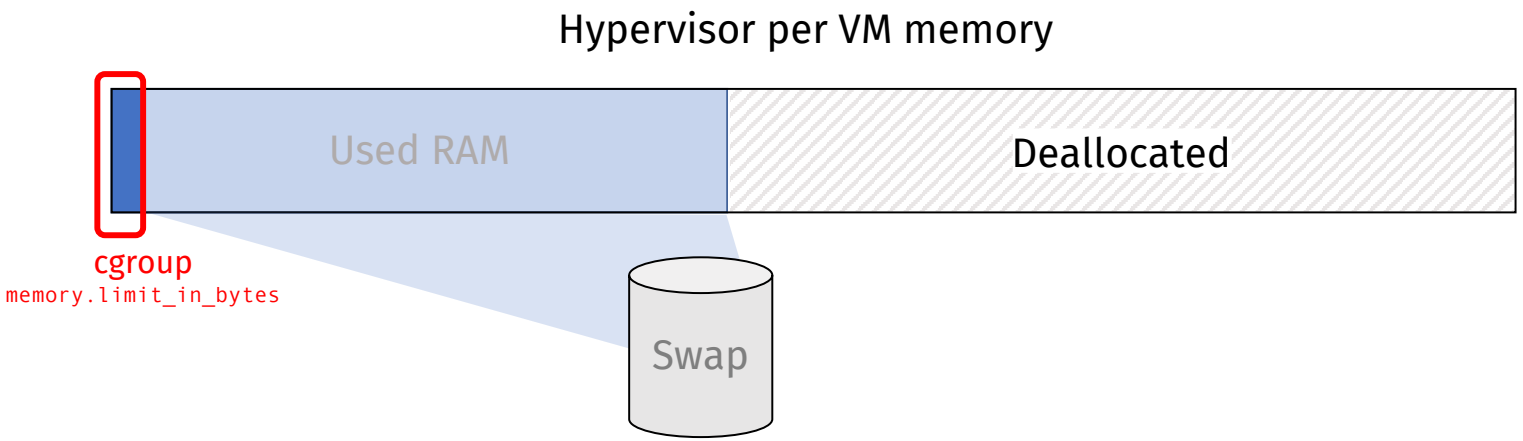
Suspending to Swap

cgroups to constraint the hypervisor VM memory into a small size

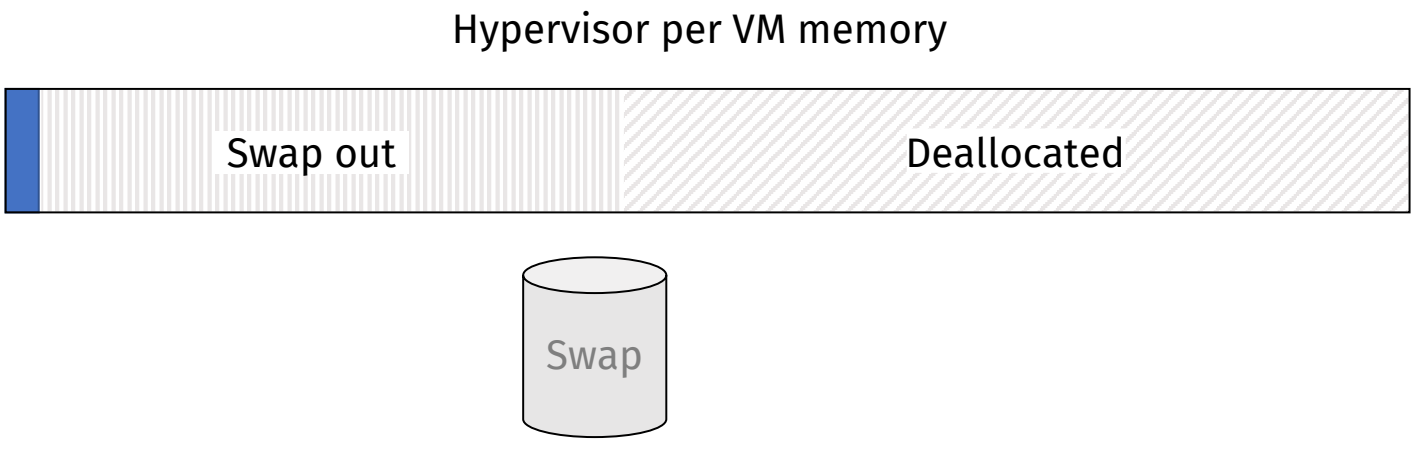


Suspending to Swap

Swap out the memory to satisfy the cgroup constraint

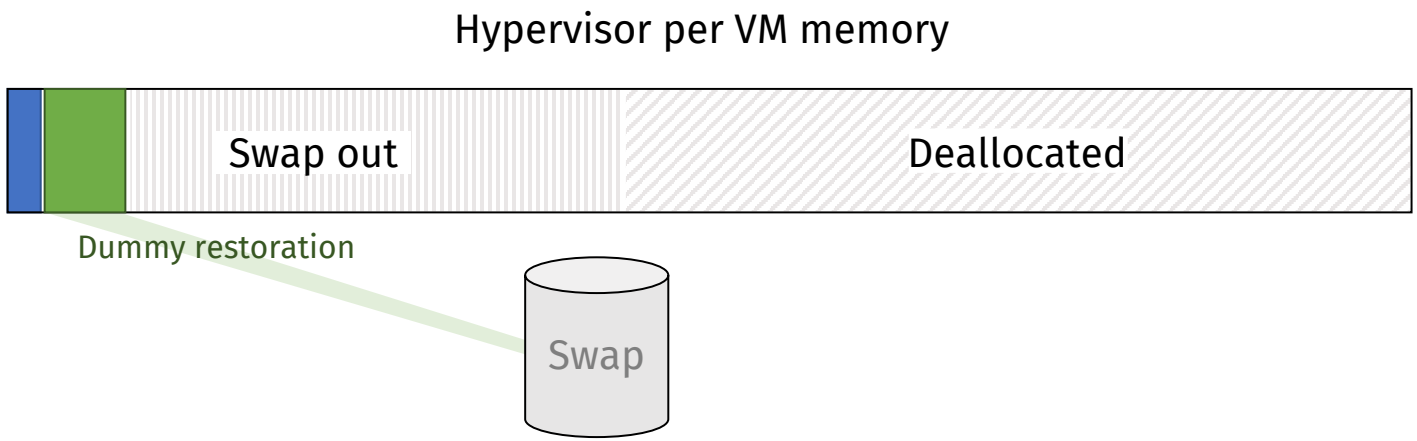


Suspending to Swap



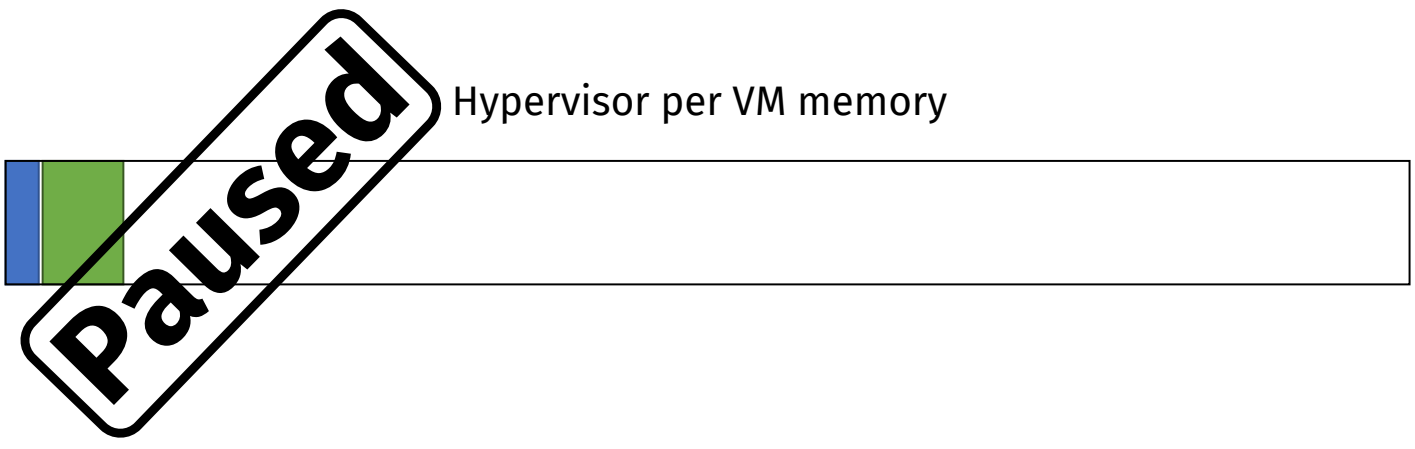
Suspending to Swap

Dummy Restoration fetches pages likely to be used during a real restoration process $\Longrightarrow$ Fast VM reply



Suspending to Swap

VM suspended to RAM



Summary

1. *Idle VMs* are turned into *VNFs*
2. *Suspend to Swap* releases *CPU* and *memory* allocated
3. New users' *non-trivial requests* are transparently intercepted, the original VM is then resumed quickly

Massive Resources Consolidation in DCN

Performance Evaluation

Performance Evaluation

SEaMLESS prototype tested with regard to

- Perceived end-user Quality of Experience
- Resulting memory savings

All tests were conducted on Grid5000 [Balouek *et al.* 2013]

Machine Specs
Dell PowerEdge R430
2 CPU Intel Xeon
32 GB RAM
10 GB Ethernet NICs

Impact over the QoE

The VM's resuming delays the processing of the request

- **Migration delay**, time to restore the Gateway Process
- **Resume the VM**, time to restore the VM

Impact over the QoE

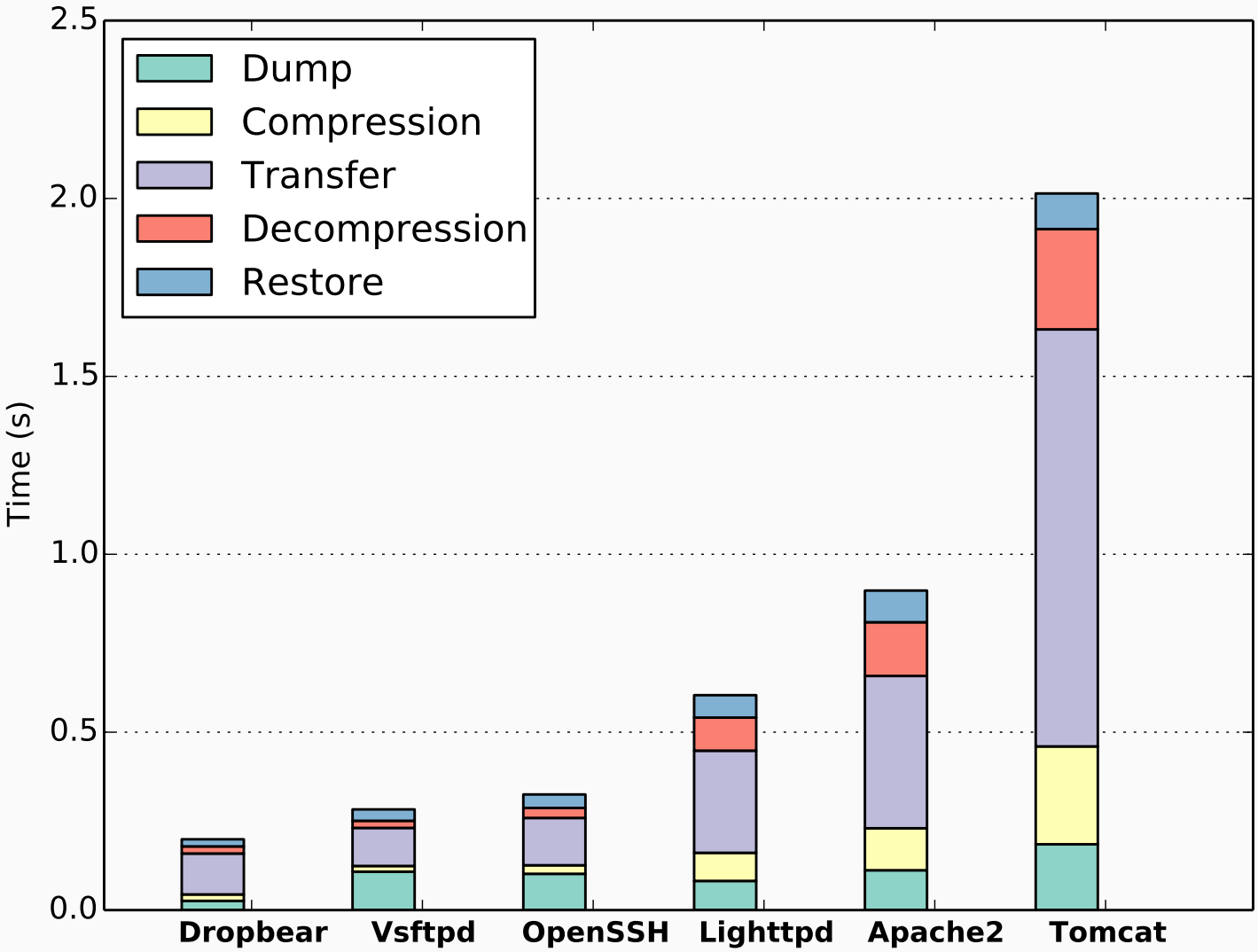
The VM's resuming delays the processing of the request

- **Migration delay**, time to restore the Gateway Process
- **Resume the VM**, time to restore the VM

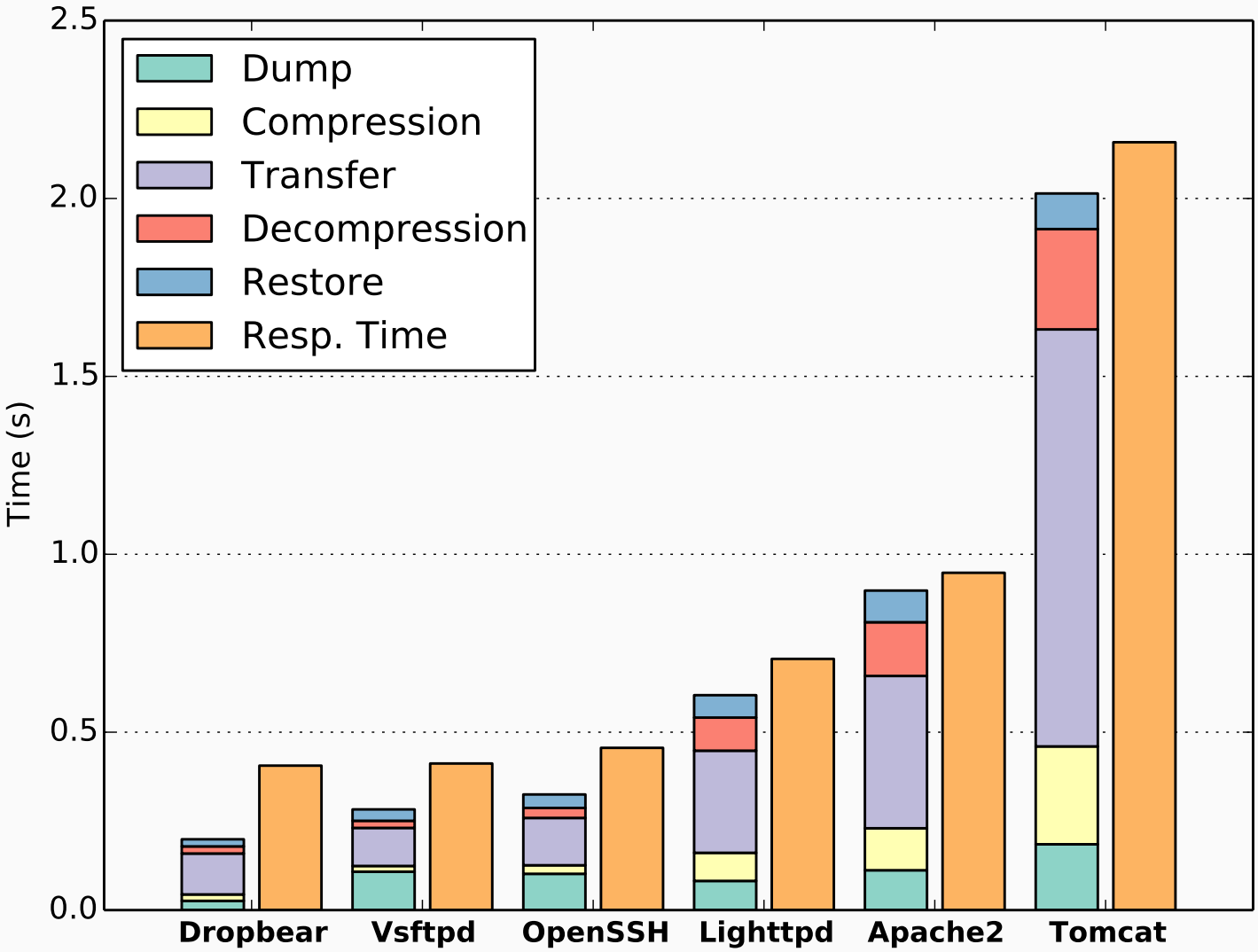
Restoring different Gateway Process depending on size and service

Application	Image Size (MB)	VNF Size (MB)
Dropbear	0.115	11.18
Vsftpd	0.107	7.81
OpenSSH	0.133	15.93
Lighttpd	0.287	46.43
Apache2	0.428	67.52
Tomcat	1.172	206.96

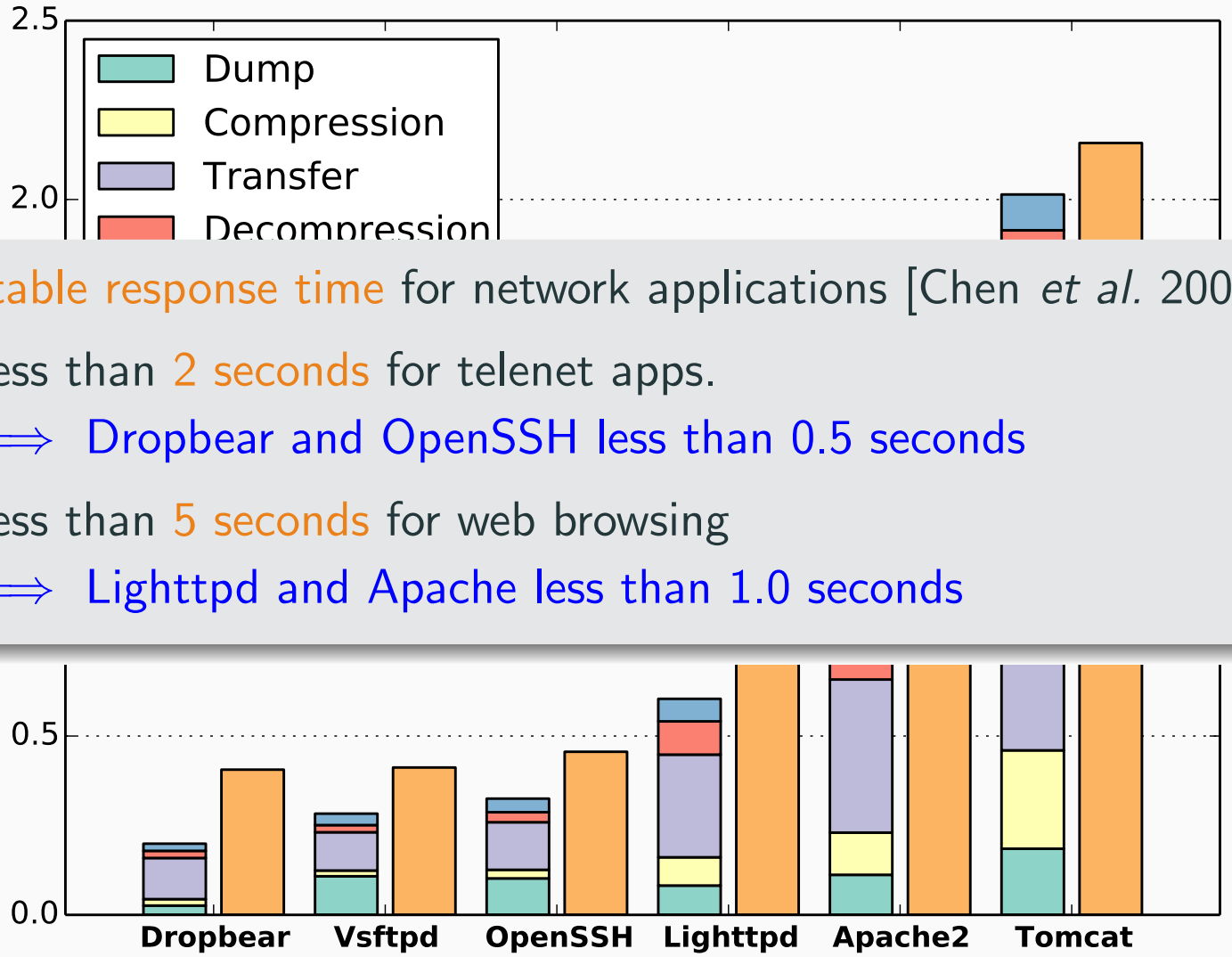
Migration Delay



Migration Delay



Migration Delay



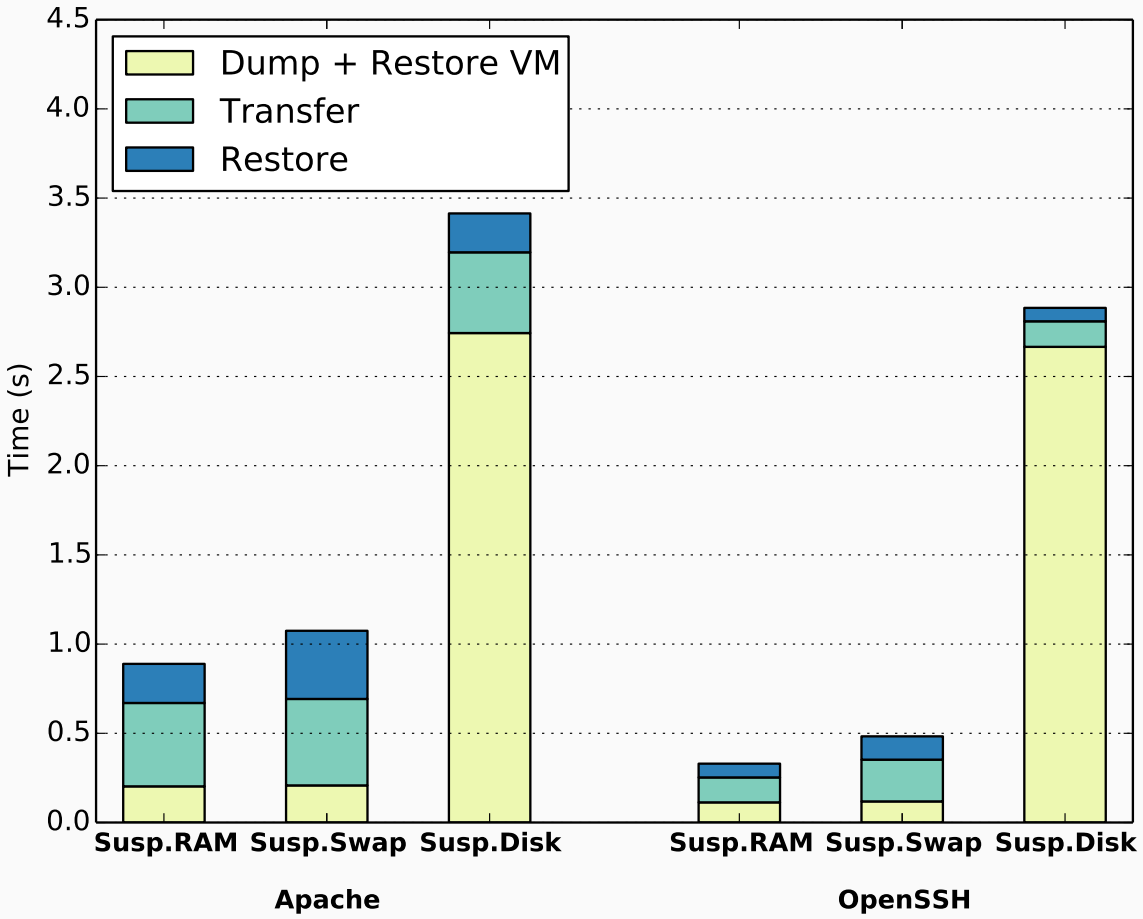
Acceptable response time for network applications [Chen *et al.* 2004]

- Less than 2 seconds for telenet apps.
  $\Longrightarrow$ Dropbear and OpenSSH less than 0.5 seconds
- Less than 5 seconds for web browsing
  $\Longrightarrow$ Lighttpd and Apache less than 1.0 seconds

Resuming the VM

Different strategies to reclaim the resources allocated by a VM

Suspend to RAM **Suspend to Swap** **Suspend to Disk**



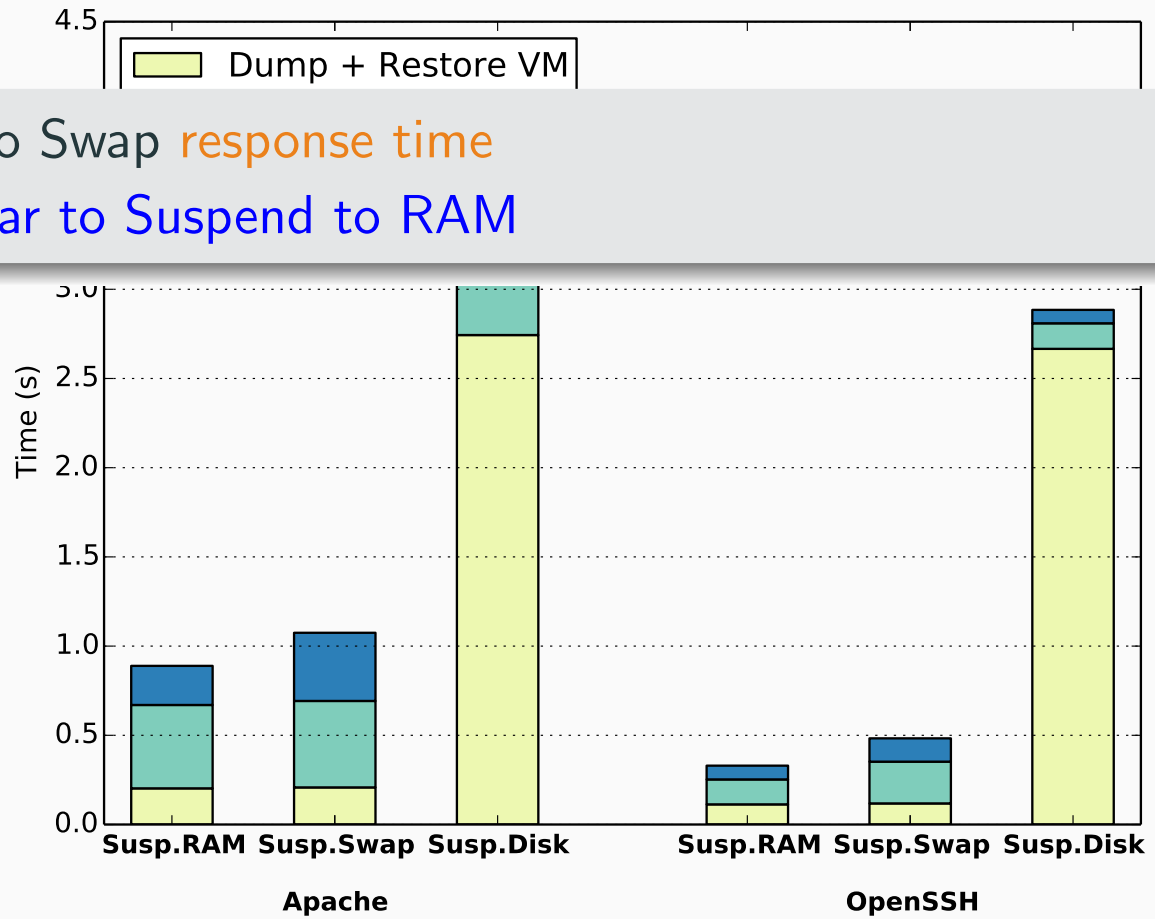
Resuming the VM

Different strategies to reclaim the resources allocated by a VM

Suspend to RAM **Suspend to Swap** **Suspend to Disk**

Suspend to Swap response time

$\Longrightarrow$ Similar to Suspend to RAM



Memory Savings

Tested Suspend to Swap with popular VM sizes from AWS

AWS Instances	Size (GB)	Reduced VM (GB)	Memory Saving (GB)
t1.micro	0.61	0.501	0.109
c1.medium	1.70	0.474	1.226
m1.small	1.70	0.474	1.226
c3.large	3.75	0.496	3.254
m1.medium	3.75	0.496	3.254
m3.medium	3.75	0.496	3.254
c1.xlarge	7.00	0.515	6.485
m1.large	7.50	0.511	6.989
m1.xlarge	15.00	0.527	14.473
m2.xlarge	17.10	0.560	16.540
m2.2xlarge	34.20	0.598	33.602

$\Longrightarrow$ Reduced between 500 MB and 600 MB no matter the initial size

Massive Resources Consolidation in DCN

Conclusions

Limitations

Some limitations are inherited from the underlying technologies

- e.g., cannot handle Unix socket stream (CRIU)
- e.g., cannot intercept writes/reads to shared memory (ptrace)

Some workloads and applications are not suitable

- *Monolithic* processes with big memory footprint

SEaMLESS Pros

- Release resources locked by idle VM instances
  $\Longrightarrow$ Suspend to Swap deallocates CPU and the majority of RAM
- Full transparency and generic application support
  $\Longrightarrow$ Sink VNF intercepts new requests and restarts VM
- Fast response time upon new activities
  $\Longrightarrow$ Response time between 0.5 and 2 seconds

SEaMLESS Pros

- Release resources locked by idle VM instances
  $\Longrightarrow$ Suspend to Swap deallocates CPU and the majority of RAM
- Full transparency and generic application support
  $\Longrightarrow$ Sink VNF intercepts new requests and restarts VM
- Fast response time upon new activities
  $\Longrightarrow$ Response time between 0.5 and 2 seconds

A. Segalini, D. Lopez Pacheco, Q. Jacquemart, M. Rifai, G. Urvoy-Keller and M. Dione. *Towards Massive Consolidation in Data Centers with SEaMLESS*. In Proceedings of CCGrid 2018, 18th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computin. Washington, DC, USA. May 2018.

Final Thoughts

General Learned Lessons

Energy Optimization, but

- Need to provide high QoS
  - Keep SLA in mind
- Integrate real constraints... little by little
- SDN is your friend
  - be reasonable

Questions?

References i

[Agarwal *et al.* 2013]
S. Agarwal, M. Kodialam, and T. Lakshman, "Traffic engineering in software defined networks," in *IEEE INFOCOM*, 2013.

[Balouek *et al.* 2013]
D. Balouek, A. Carpen Amarie, G. Charrier, F. Desprez, E. Jeannot, E. Jeanvoine, A. Lèbre, D. Margery, N. Niclausse, L. Nussbaum, O. Richard, C. Pérez, F. Quesnel, C. Rohr, and L. Sarzyniec, "Adding virtualization capabilities to the Grid'5000 testbed," in *Communications in Computer and Information Science – Cloud Computing and Services Science*, vol. 367, Springer, 2013.

[Chen *et al.* 2004]
Y. Chen, T. Farley, and N. Ye, "Qos requirements of network applications on the internet," *Inf. Knowl. Syst. Manag.*, vol. 4, no. 1, pp. 55–76, Jan. 2004.

[CRIU]
CRIU, *Checkpoint/Restore In Userspace*, `https://criu.org/Main_Page`.

References ii

[De Rango *et al.* 2012]

F. De Rango, F. Guerriero, and P. Fazio, "Link-stability and energy aware routing protocol in distributed wireless networks," *IEEE Transactions on Parallel and Distributed systems*, vol. 23, no. 4, 2012.

[Franzini, 2012]

Giovanni Franzini, "Idle virtual machine detection in fermicloud," Fermi National Accelerator Laboratory Scientific Computing Division Grid and Cloud Computing, Tech. Rep., 2012.

[Giroire *et al.* 2014]

F. Giroire, J. Moulierac, and T. K. Phan, "Optimizing rule placement in software-defined networks for energy-aware routing," in *IEEE GLOBECOM*, 2014.

[Heller *et al.*]

B. Heller, S. Seetharaman, P. Mahadevan, Y. Yiakoumis, P. Sharma, S. Banerjee, and N. McKeown, "Elastictree: Saving energy in data center networks.," in *USENIX NSDI 2010*.

References iii

[Idzikowski *et al.* 2016]
F. Idzikowski, L. Chiaraviglio, A. Cianfrani, J. L. Vizcaíno, M. Polverini, and Y. Ye, "A survey on energy-aware design and operation of core networks," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, 2016.

[Knauth *et al.* 2014]
T. Knauth and C. Fetzer, "Dreamserver: Truly on-demand cloud services," in *Proceedings of International Conference on Systems and Storage*, ACM, 2014, pp. 1–11.

[Koomey *et al.* 2017]
J. Koomey and J. Taylor, "Zombie/comatose servers redux", 2017. [Online]. Available: `anthesisgroup.com/wp-content/uploads/2017/03/Comatsoe-Servers-Redux-2017.pdf`.

References iv

[Maaloul *et al.* 2018]
R. Maaloul, L. Chaari, and B. Cousin, "Energy saving in carrier-grade networks: A survey," *Computer Standards & Interfaces*, vol. 55, pp. 8 –26, 2018, ISSN: 0920-5489. DOI: https://doi.org/10.1016/j.csi.2017.04.001. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0920548916301817.

[Mahadevan *et al.* 2009]
P. Mahadevan, P. Sharma, S. Banerjee, and P. Ranganathan, "A power benchmarking framework for network devices," in *Proceedings of the 8th International IFIP-TC 6 Networking Conference*, ser. NETWORKING '09, Aachen, Germany: Springer-Verlag, 2009, pp. 795–808, ISBN: 978-3-642-01398-0. DOI: 10.1007/978-3-642-01399-7_62. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-01399-7_62.

References v

[Orlowski *et al.* 2007]

S. Orlowski, M. Pióro, A. Tomaszewski, and R. Wessäly, "SNDlib 1.0–Survivable Network Design Library," English, in *Proceedings of the 3rd International Network Optimization Conference (INOC 2007), Spa, Belgium*, http://sndlib.zib.de, extended version accepted in Networks, 2009., 2007. [Online]. Available: `http://www.zib.de/orlowski/Paper/OrlowskiPioroTomaszewskiWessaely2007-SNDlib-INOC.pdf.gz`.

[BtrPlace]

T. B. Project, *An open-source flexible virtual machine scheduler*, `http://www.btrplace.org/`.

[Rouzic *et al.* 2013]

E. L. Rouzic, E. Bonetto, L. Chiaraviglio, F. Giroire, F. Idzikowski, F. Jimenez, C. Lange, J. Montalvo, F. Musumeci, I. Tahiri, A. Valenti, W. V. Heddeghem, Y. Ye, A. Bianco, and A. Pattavina, "Trend towards more energy-efficient optical networks," in *ONDM*, Brest, France, 2013, pp. 211–216.

References vi

[Shang *et al.* 2010]
Y. Shang, D. Li, and M. Xu, "Energy-aware routing in data center network," in *ACM workshop on Green networking*, 2010.

[Van Heddeghem *et al.* 2014]
W. Van Heddeghem, S. Lambert, B. Lannoo, D. Colle, M. Pickavet, and P. Demeester, "Trends in worldwide ict electricity consumption from 2007 to 2012," *Comput. Commun.*, vol. 50, pp. 64–76, Sep. 2014, ISSN: 0140-3664. DOI: 10.1016/j.comcom.2014.02.008. [Online]. Available: http://dx.doi.org/10.1016/j.comcom.2014.02.008.

[Vissicchio *et al.* 2014]
S. Vissicchio, L. Vanbever, and O. Bonaventure, "Opportunities and research challenges of hybrid software defined networks," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 2, 2014.

References vii

[Zhang *et al.* 2017]
B. Zhang, Y. Al-Dhuraibi, R. Rouvoy, F. Paraiso, and L. Seinturier, "Cloudgc: Recycling idle virtual machines in the cloud," in *2017 IEEE International Conference on Cloud Engineering, IC2E 2017, Vancouver, BC, Canada, April 4-7, 2017*, 2017, pp. 105–115. DOI: 10.1109/IC2E.2017.26. [Online]. Available: https://doi.org/10.1109/IC2E.2017.26.

[Zhang *et al.* 2016]
L. Zhang, J. Litton, F. Cangialosi, T. Benson, D. Levin, and A. Mislove, "Picocenter: Supporting long-lived, mostly-idle applications in cloud environments," in *EuroSys*, ACM, 2016.

[Zhi *et al.* 2016]
J. Zhi, N. Bila, and E. de Lara, "Oasis: Energy proportionality with hybrid server consolidation," in *EuroSys*, ACM, 2016.