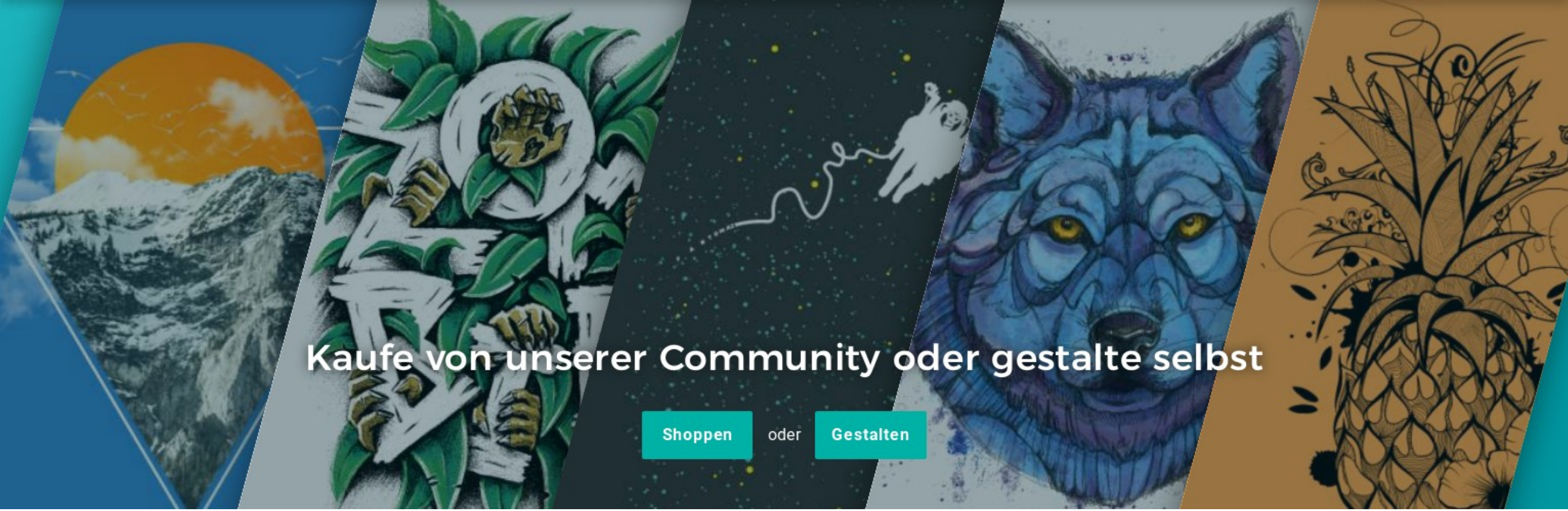
Hyper Converged Infrastructure

How we did it @ Spreadshirt



sprd.net AG

Gießerstraße 27
04229 Leipzig
Deutschland

Ansgar Jazdzewski
Senior System Engineer

ansgar.jazdzewski@spreadshirt.net

Bernd Naumann
System Engineer

bernd.naumann@spreadshirt.net

HYPER-CONVERGED INFRASTRUCTURE (HCI)

Hyper-Converged is a Software-Defined-Approach to manage Virtualazion (IaaS), Storage (SDS) and Network (SDN) within the same nodes/computers,
and scale them horizontal as a Unit.

Preperation

Hardware, and Rack-Layout
Spine-Leaf-Architecture
Provisioning

SDN with Bird & VxLAN
IaaS with Ganeti
SDS with Ceph

Preperation

Preparation

- How many computers do we have now and will we need?
- What kind of storage do we allocate now and in the future?
- Do we have special purpose hardware which we need to include?
- Compare your definitions and imaginations of HCI and SDN with your vendors. (White paper, articles, ...)
- Set your goals about feature-sets and calibrate your expectations.
- Take your time in the lab and build your setup as near to reality as you can! Do training and test your setup well.

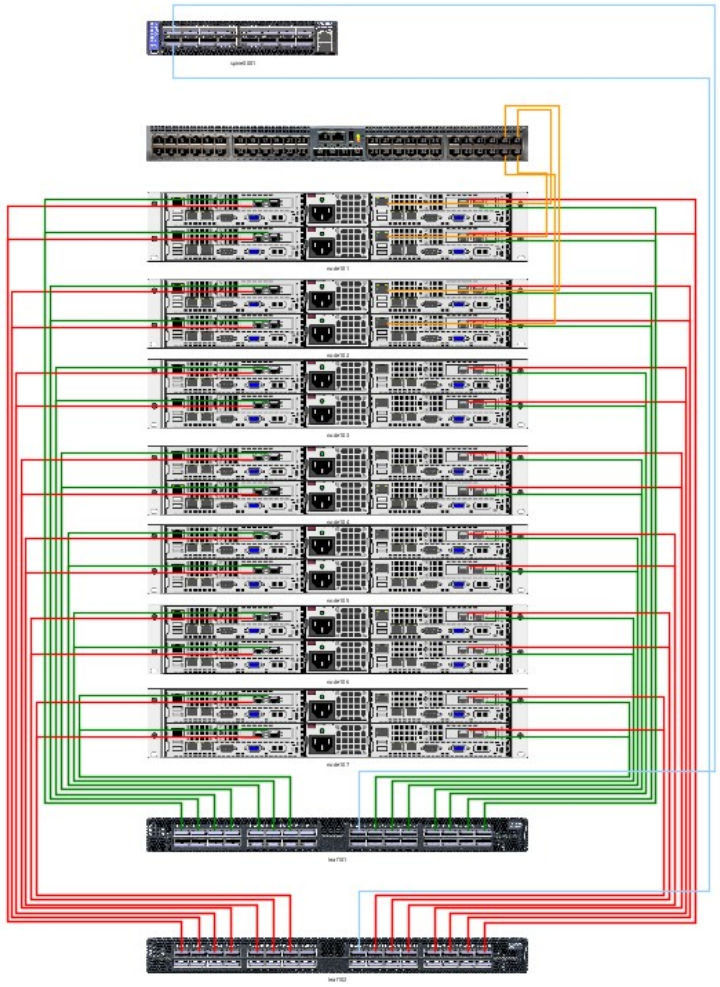
Hardware, and Rack-Layout

Hardware, and Rack-Layout

- How many sockets, cores, threads, and how many VMs should be served? What kind of VMs?
- CPU/RAM ratio and Threads/VM & Memory/VM?
- Do you allocate extension-slots? i.e. PCIe-Passthrough
- Network-Cards, -Cables, and -Switches
- *To be continued...*

Hardware, and Rack-Layout – Planing

- Put leaf in the middle of the Rack (shorter cables)
- Put management on the top and bottom (less cables ends at the same spot in the rack)
- Fill up with nodes – As long as you can or have ports...



Hardware, and Rack-Layout – in practice and reality

- Not as clean as we hoped
- Main cause:
  - Cables are not available in all length (0,7m)
  - Even the widest rack they could offer us seams tight
  - Cable-management can make it even worst!

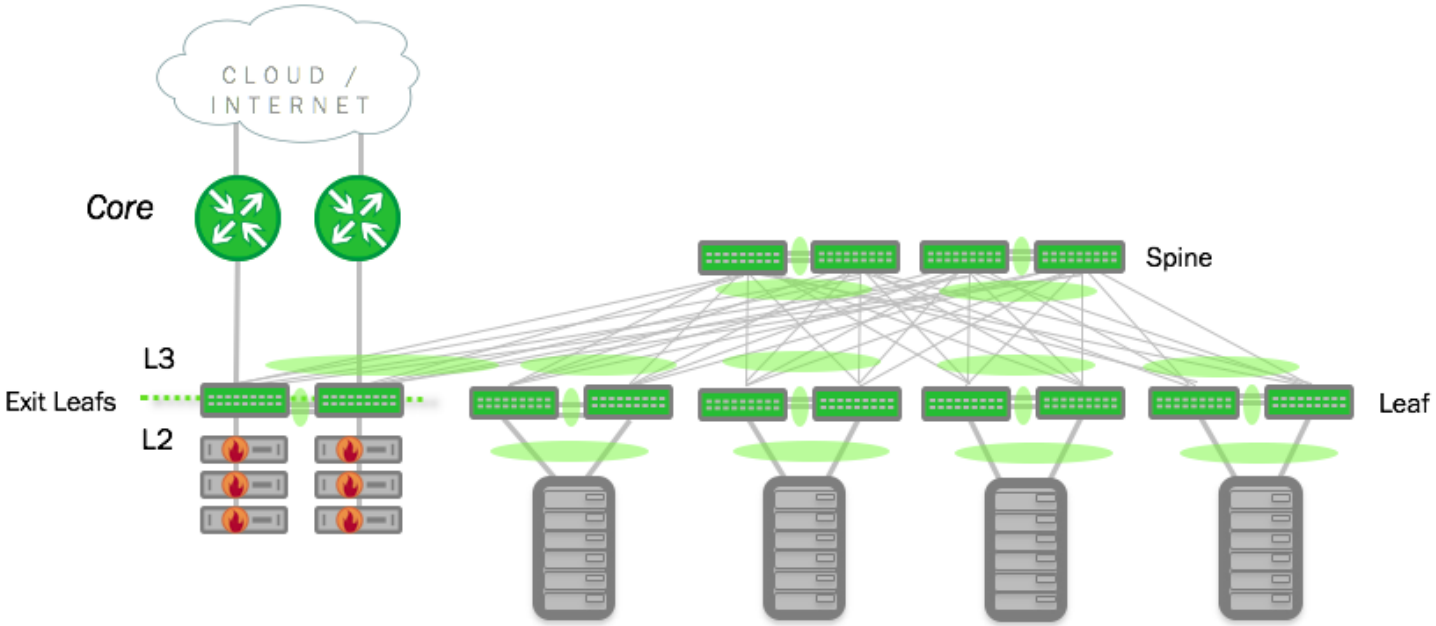
Hardware, and Rack-Layout – lesson learned

- Keep rack-width in mind (go as wide as you can)
- Keep cables as short as possible
- Take care of the “Air-Flow” (C2P or P2C)
- Use colors (port based)
- Always keep your remote-hands-service in mind
- Try to keep access easy

Spine-Leaf
(IP-Fabric)

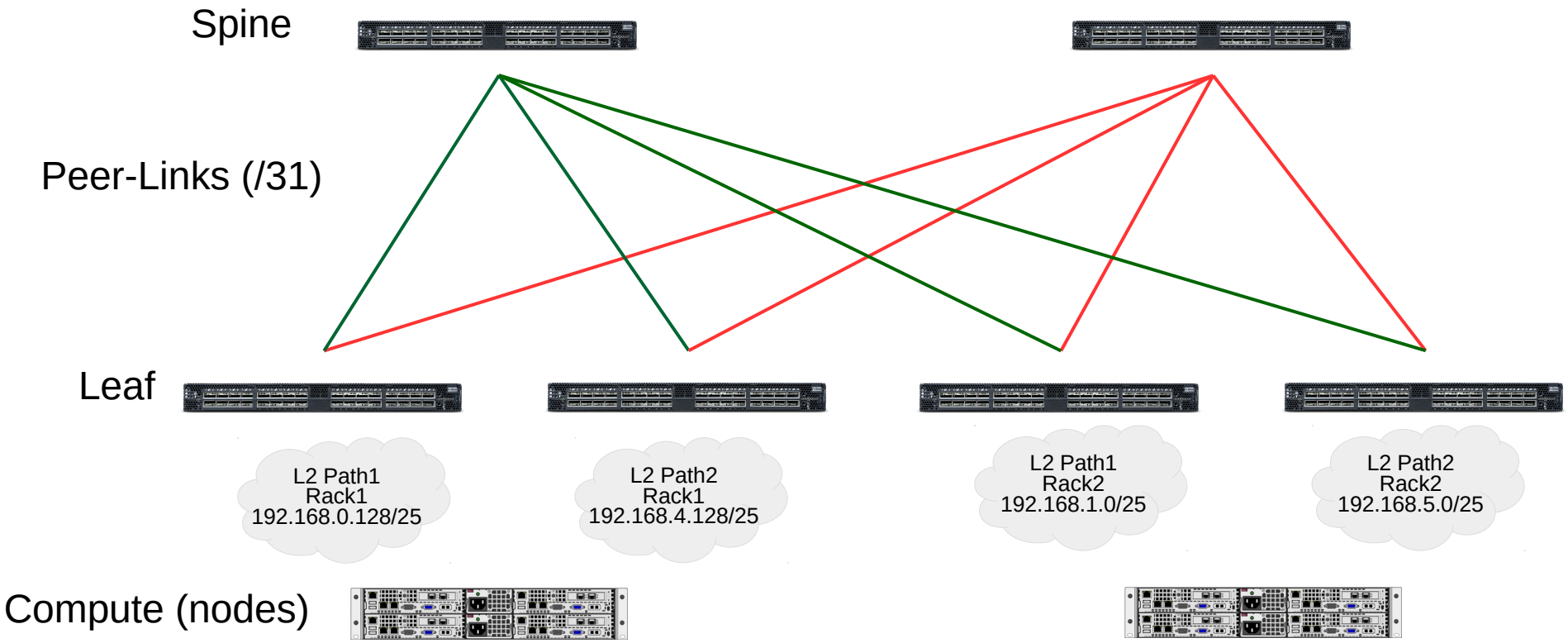
Spine-Leaf

- We think of a tree of routers (or switch-routers)
- Spine – Leaf – Node: in theory full L3-routed



https://support.cumulusnetworks.com/hc/en-us/article_attachments/205509927/l2alltheway.png

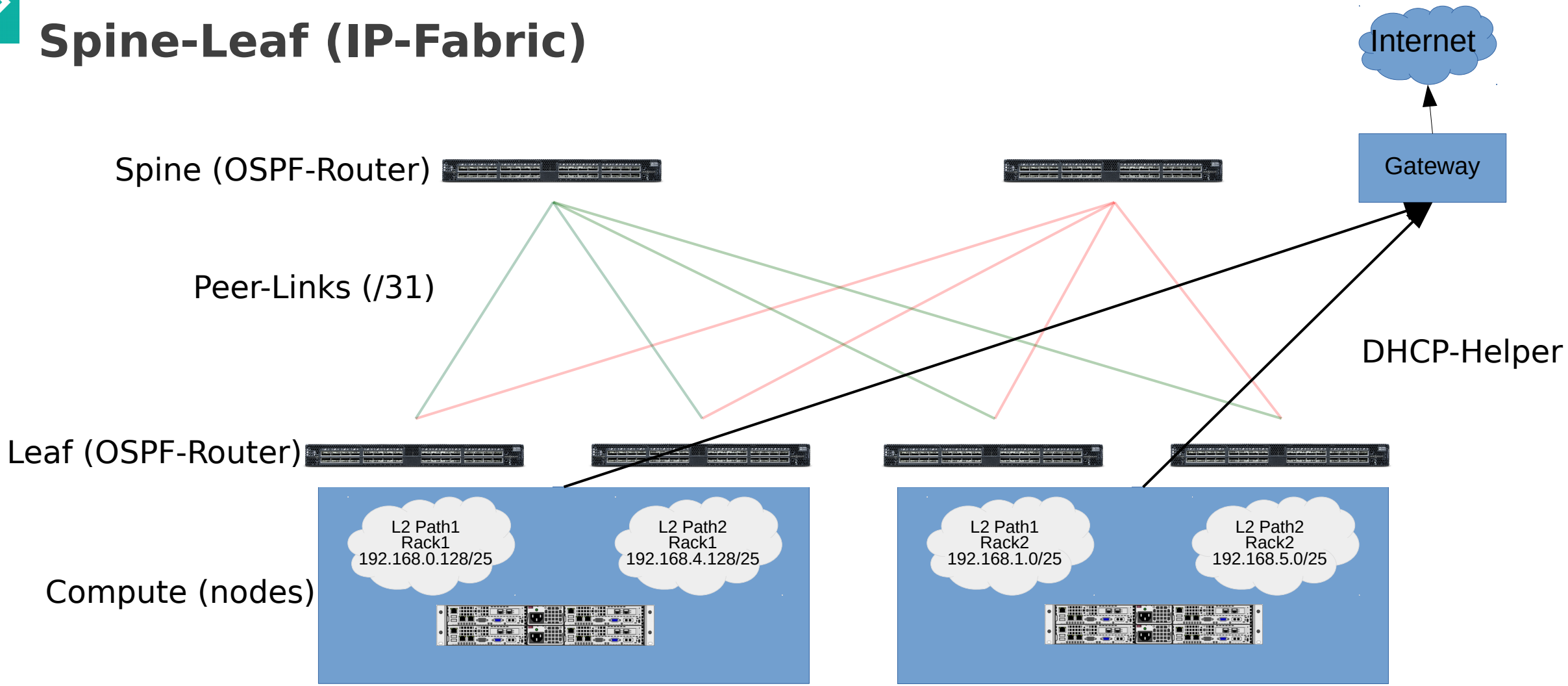
Spine-Leaf (IP-Fabric)



Spine-Leaf – some modifications to our needs

- We like to have a `pressed/d-i` based installation, so we need DHCP.
- Because of this decision we have a `Layer-2 Braodcast-Domain` between Leaf-Switches and Compute-Nodes
- In consequence, we need to route the path networks (used for ECMP) to our gateway and installation-server (iPXE, etc), which is not intended by design.

Spine-Leaf (IP-Fabric)



Provisioning

Provisioning – Preseed and d-i pitfalls

- Buy hardware of the same kind

```
# This command is run immediately before the partitioner starts. It may be
# useful to apply dynamic partitioner preseeding that depends on the state
# of the disks (which may not be visible when preseed/early_command runs).

d-i partman/early_command string \
DISKA=$(realpath /dev/disk/by-id/ata-SuperMicro_SSD_* /dev/disk/by-id/ata-SATA_SSD_*
/dev/disk/by-id/ata-INTEL_SSD*|grep -o '/dev/sd[a-z]'|uniq|sort|head -n1);\
DISKB=$(realpath /dev/disk/by-id/ata-SuperMicro_SSD_* /dev/disk/by-id/ata-SATA_SSD_*
/dev/disk/by-id/ata-INTEL_SSD*|grep -o '/dev/sd[a-z]'|uniq|sort|head -n2|tail -1);\
if [ "${DISKA}" == "" ]; then DISKA="/dev/sda"; fi;\
if [ "${DISKB}" == "" ]; then DISKB="/dev/sdb"; fi;\
debconf-set partman-auto/disk "$DISKA $DISKB";\
debconf-set partman-auto-raid/recipe "1 2 0 ext4 / ${DISKA}2#${DISKB}2 . 1 2 0 lvm - $
{DISKA}4#${DISKB}4 .";\
debconf-set grub-installer/bootdev "$DISKA $DISKB";
```

Provisioning – Preseed and d-i pitfalls

- Special Network Config & Bugs you hit

```
d-i                 preseed/late_command                string      \
cp /target/etc/network/interfaces /etc/network/interfaces ; \
    in-target /usr/bin/rm /etc/apt/sources.list ; \
    in-target /usr/bin/apt update ; \
    in-target /usr/bin/apt-get install -y linux-image-4.15.0-13-generic ; \
        sed -i -e 's/\(HashKnownHosts\) yes/\1 no/' /target/etc/ssh/ssh_config ; \
        rm -f /target/etc/cron.weekly/fstrim ; \
        echo 'path-exclude /etc/cron.weekly/fstrim' >
/target/etc/dpkg/dpkg.cfg.d/exclude_etc_cron.weekly_fstrim ; \
        echo 'server = puppet4ca.sprd.net' >> /target/etc/puppetlabs/puppet/puppet.conf
;
```

Provisioning – DHCP & DNS

How to solve the dependency on DHCP and DNS and bring up the data-center from scratch?!

Provisioning – Preseed – First boot DHCP

- Try not to depend on it to bring up your environment! – 1

```
source /etc/network/interfaces.d/*

auto lo
iface lo inet loopback

allow-hotplug ens1f0
iface ens1f0 inet manual
  pre-up ip link set ens1f0 up
  pre-up /usr/bin/sprd-network ens1f0
  mtu 9000

allow-hotplug ens1f1
iface ens1f1 inet manual
  pre-up ip link set ens1f1 up
  pre-up /usr/bin/sprd-network ens1f1
  mtu 9000
```

Provisioning – Preseed – First boot DHCP

- Try not to depend on it to bring up your environment! – 2

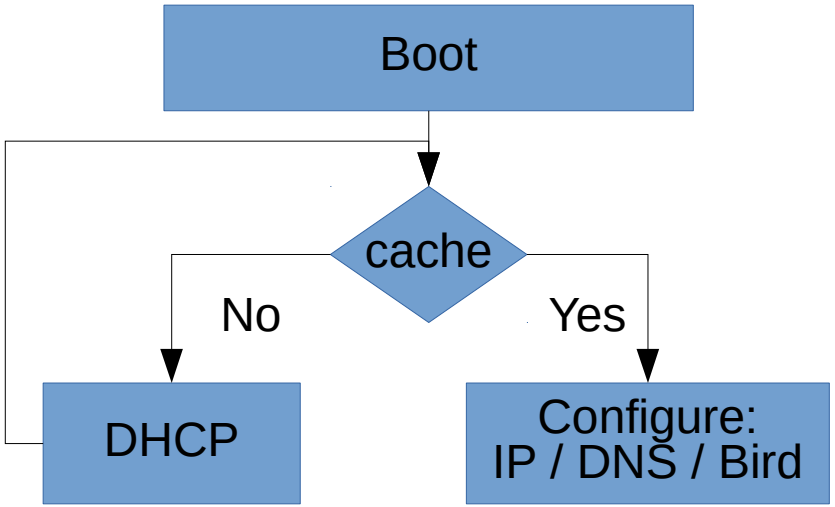
```
case "${1}" in
        enp*|ens*|eth*)
                # load cache if present
                if [ -f "/var/cache/sprd-network.cache" ]; then
                        # shellcheck source=/dev/null
                        . "/var/cache/sprd-network.cache"
                else
                        dhcpclient "${1}"
                        # shellcheck source=/dev/null
                        . "/var/cache/sprd-network.cache"
                fi

                modprobe tcp_bbr
                if grep --quiet --only-matching --regexp="bbr" "/proc/sys/net/ipv4/tcp_available_congestion_control"; then
                        echo "bbr" > "/proc/sys/net/ipv4/tcp_congestion_control"
                fi

                setInterface lo "${CIDR_LO}"
                setInterface "${NIC0}" "${CIDR_PATH1}"
                setInterface "${NIC1}" "${CIDR_PATH2}"
                setHostname "${NODE_NAME}"
                setDNS "${DNS_SERVER}" "${DNS_SEARCH}"
                configureBird "${BIRD_IP}" "${BIRD_AREA}" "${NIC0}" "${NIC1}"
                configureDNSmasq "${DNS_SERVER}" "${DC}"
                configurePuppet "${DC}" "${CONTEXT}" "${BIRD_IP}" "${IP_PATH1}" "${IP_PATH2}" "${BIRD_AREA}"
  ;;
```

Provisioning – Preseed – First boot DHCP

- udhcpc (busybox) can be used to pass DHCP-results into a script, so we use it to configure our network.



Provisioning – Preseed – First boot DNS

- `/etc/hosts` and `/etc/dnsmasq.conf` are statically installed by `sprd-network`
  - Sad, but true. In the end: How often do we deploy new nodes?
- DNS mostly with SRV-Records and some CNAMEs
  - apt-repository-url, key-value-store, puppet, ...

  ..., which are served by a “special” dnsmasq setup.

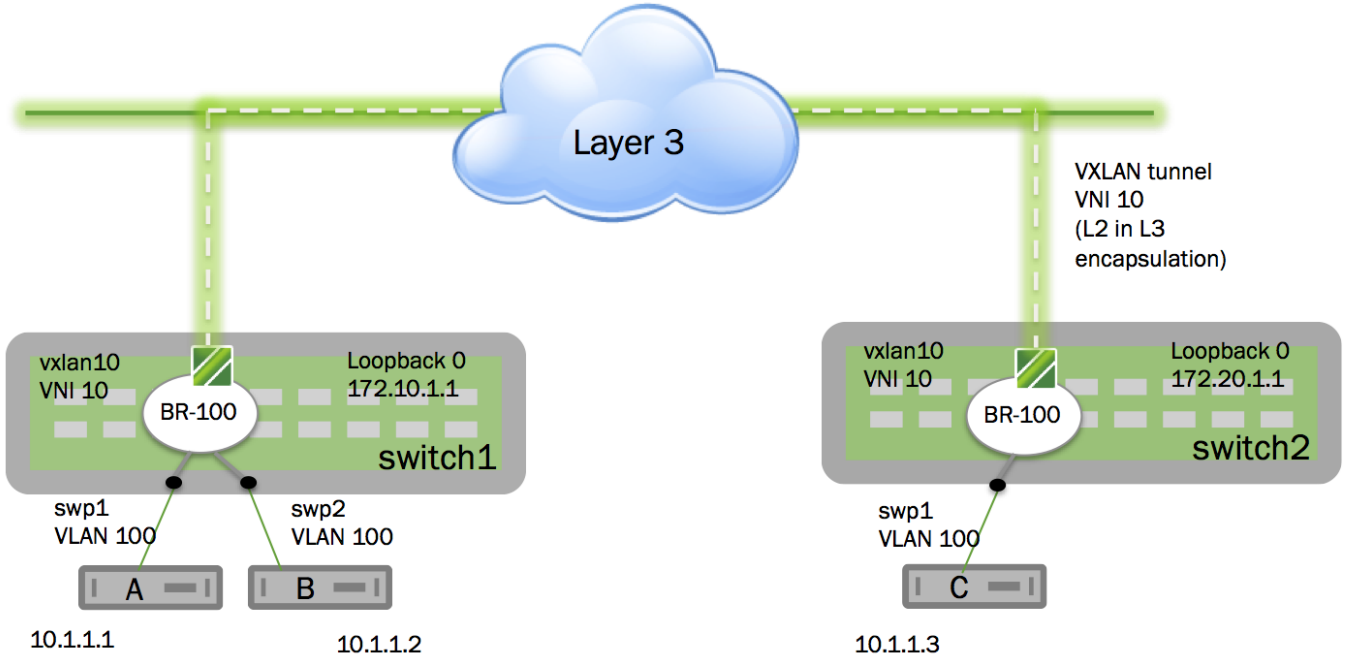
  (VMs will not use this DNS-Server, but a dedicated cluster-setup.)
- `/etc/hosts` on each node
  - Every other compute-node or server
  - Cluster-Managers virtual IP (VRRP)

SDN
(VxLAN & Bird)

SDN – VxLAN

- Almost everyone speaks about using it, but why and how?
  - Actually its plain stupid simple, and beats many other encapsulation/tunnel solutions, but VxLAN support is some how… *tricky*.
- L2 in L3 encapsulation
  - Ansgar: “A bit like VLAN and VPN without encryption, maybe...”
  - Virtual Layer-2 over L3/L4
- 24 bit VNIs (VXLAN Network Identifier or VXLAN Segment ID) in VxLAN-Header, which can hold 12 bit VLANs each
- https://tools.ietf.org/html/rfc7348

SDN – VxLAN

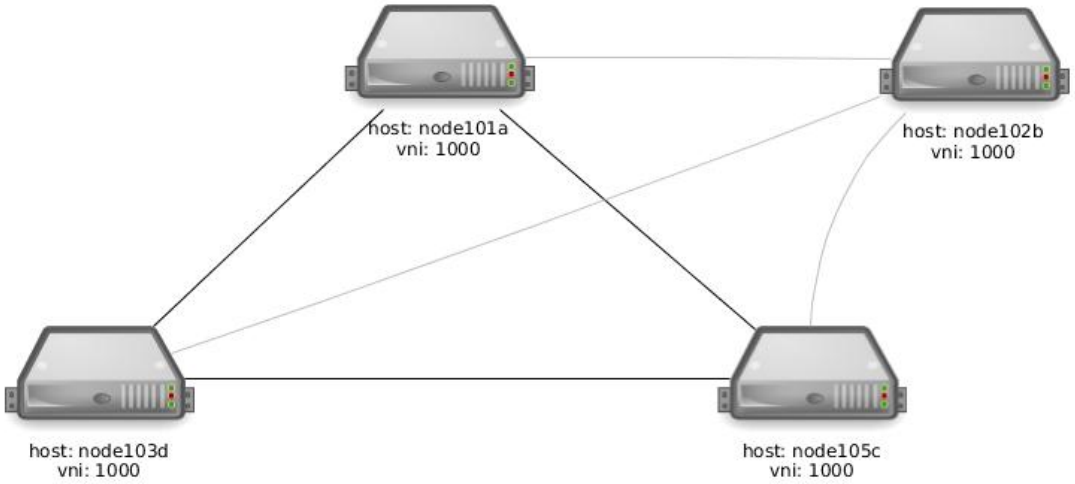


https://docs.cumulusnetworks.com/

SDN – VxLAN – a lot of Questions

- How does VxLAN handle Brodcast-Unknown-Multicast?
  - How do VMs in a shared VNI find each other?
- How is the Tunnel provided?
  - How is the tunnel established?
- How fast is it? / How big is the latency added by VxLAN?
  - Do you even recognize it?

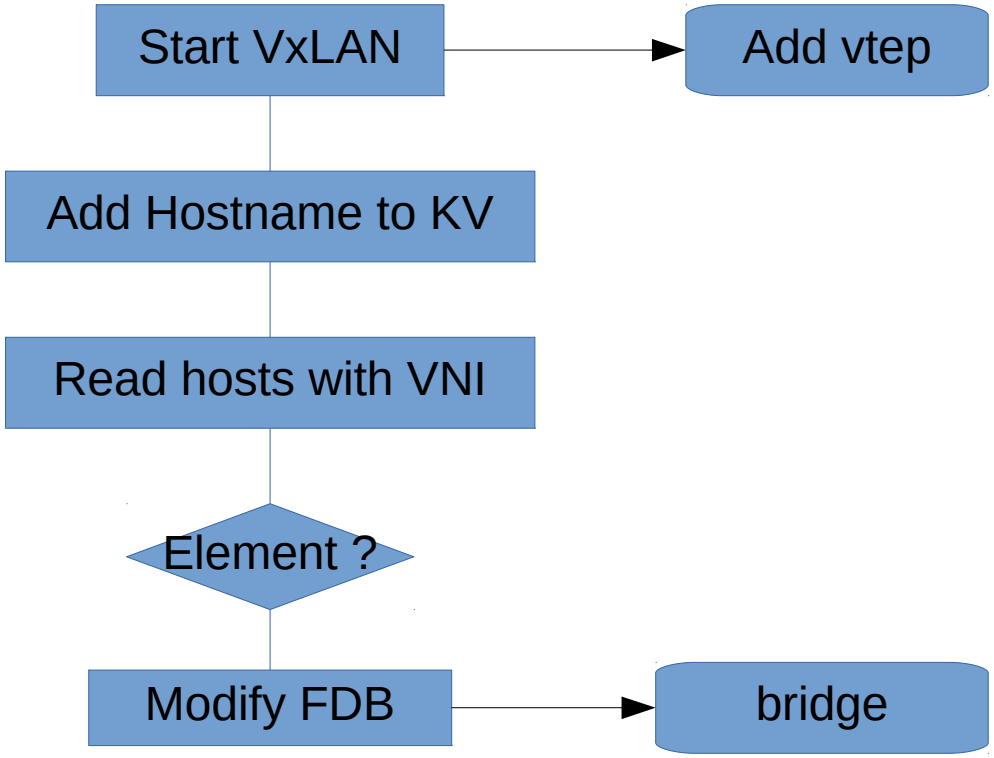
SDN – VxLAN – BUM?



Nodes share FDB information by

- Multicast (IGMP / PIM), or
- SDN Controller, or
- Static, or
- FDBDD, or
- …

VxLAN – What’s happening in the end…



- ip link add <vtep> type vxlan id <vni> proxy local <localip> dstport 4789
- bridge fdb append to 00:00:00:00:00:00 dev <vtep> dst <node>

Hint: https://vincent.bernat.im/en/blog/2017-vxlan-linux

SDN – VxLAN – FDBDD?

- FDB Distributed Daemon
- fdbdd is a simple python-service handling registration of a node for an VxLAN-ID and to find other nodes providing this VNI.
- fdbdd is triggered by the cluster-management.

SDN
(VxLAN & Bird)

SDN – Routing with Bird

- http://bird.network.cz/ (OSPF and BGP)
- We use it for OSPF (easy discovery)
- To be able to use ECMP (EqualCostMultiPath)

```
protocol kernel {
    learn;
    persist;
    scan time 10;
    import all;
    export all;
    merge paths yes;
    export filter {
      krt_prefsrc = <HostIP>;
      accept;
    };
}
```

SDN – Routing with Bird (ECMP)

- This is how it looks like, each host can be reached over two independent Layer-2 connections

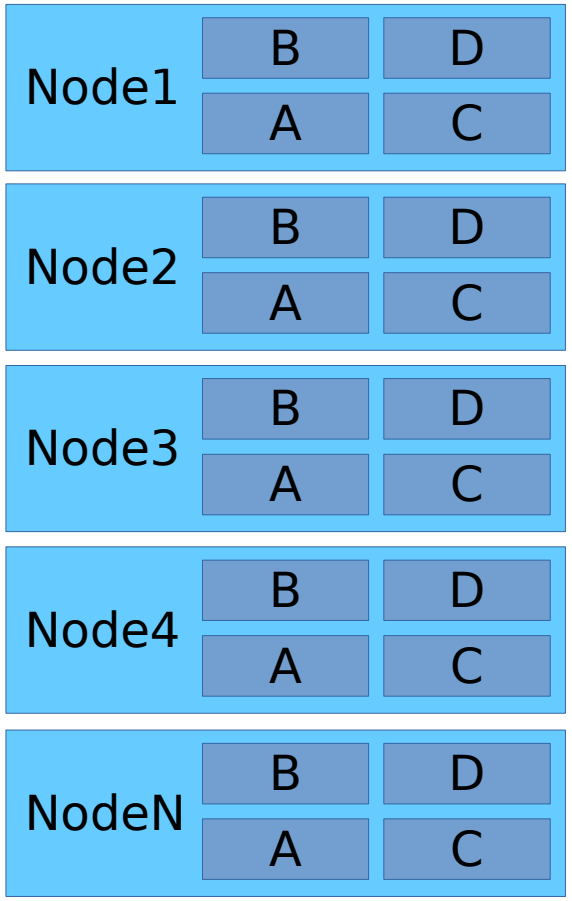
```
172.16.160.65  proto bird  src 172.16.160.6
        nexthop via 192.168.0.4  dev ens1f0 weight 1
        nexthop via 192.168.4.5  dev ens1f1 weight 1
172.16.160.66  proto bird  src 172.16.160.6
        nexthop via 192.168.0.4  dev ens1f0 weight 1
        nexthop via 192.168.4.5  dev ens1f1 weight 1
172.16.160.67  proto bird  src 172.16.160.6
        nexthop via 192.168.0.4  dev ens1f0 weight 1
        nexthop via 192.168.4.5  dev ens1f1 weight 1
```

IaaS with Ganeti

Iaas with Ganeti – Features

- Ganeti is a virtual machine cluster management tool built on top of existing virtualization technologies such as Xen or KVM and other open source software.
- Ganeti is designed to facilitate cluster management of virtual servers and to provide fast and simple recovery after physical failures using commodity hardware.

IaaS with Ganeti – Data-Center Zones / Failure-Domains



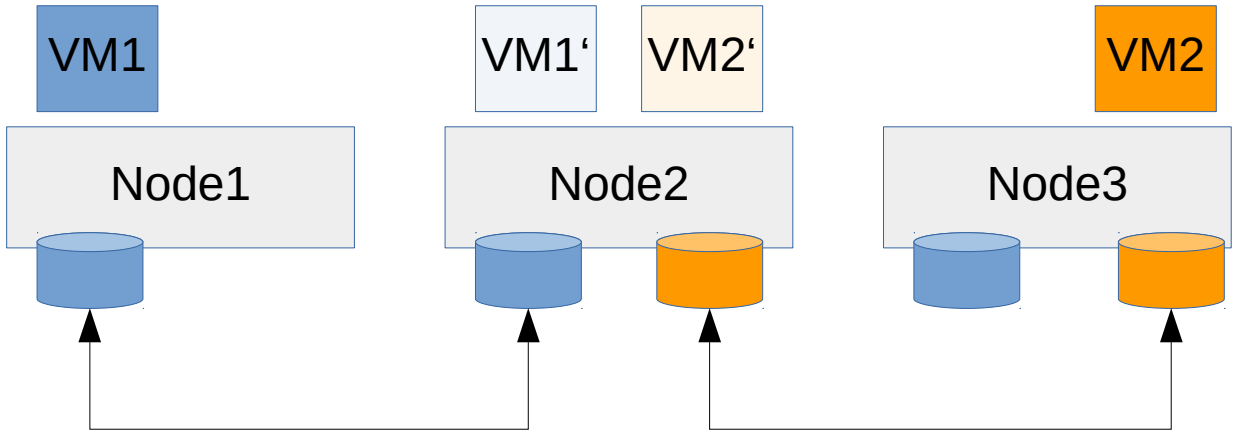
- All nodes with the same letter will join one group (zone_a..d) by adding a tag
- A service have to be deployed in different zones
- Each zone is n+1 redundant
- `hbal` + `hail` will take care of VM-placment

Iaas with Ganeti – Compromises and Issues

- We are still using tap- and bridge-interface
  - VMs will stuck below 10 Gbit/s
  - vhost with multi-queue could be one option
  - But do we miss it (yet)?
- But, because we are not using i.e. SR-IOV, live migration of VMs is possible, and ECMP is fast.
- Setup some nodes with special purpose hardware for special purpose VMs, forces you to make compromises for the provided redundancy. We have these with NVMe-Cards.

SDS with Ceph

SDS with Ceph – Hyper-Converged



- You will need VMs when you do not can rely on our storage, yet.
- ceph-monitor in VM with DRBD-Storage
- Rados-Gateway in VM (Could use RBD)
- metadata service – Not sure if we will deploy it yet

SDS with Ceph – Deployment

- `cephdeploy` do not like /32 IP-Addresses
  - But we have them assigned to our loopback-device as our preferred src-IP
- But we know how to deploy Ceph the old way
  - See http://docs.ceph.com/docs/master/install/manual-deployment/ for details

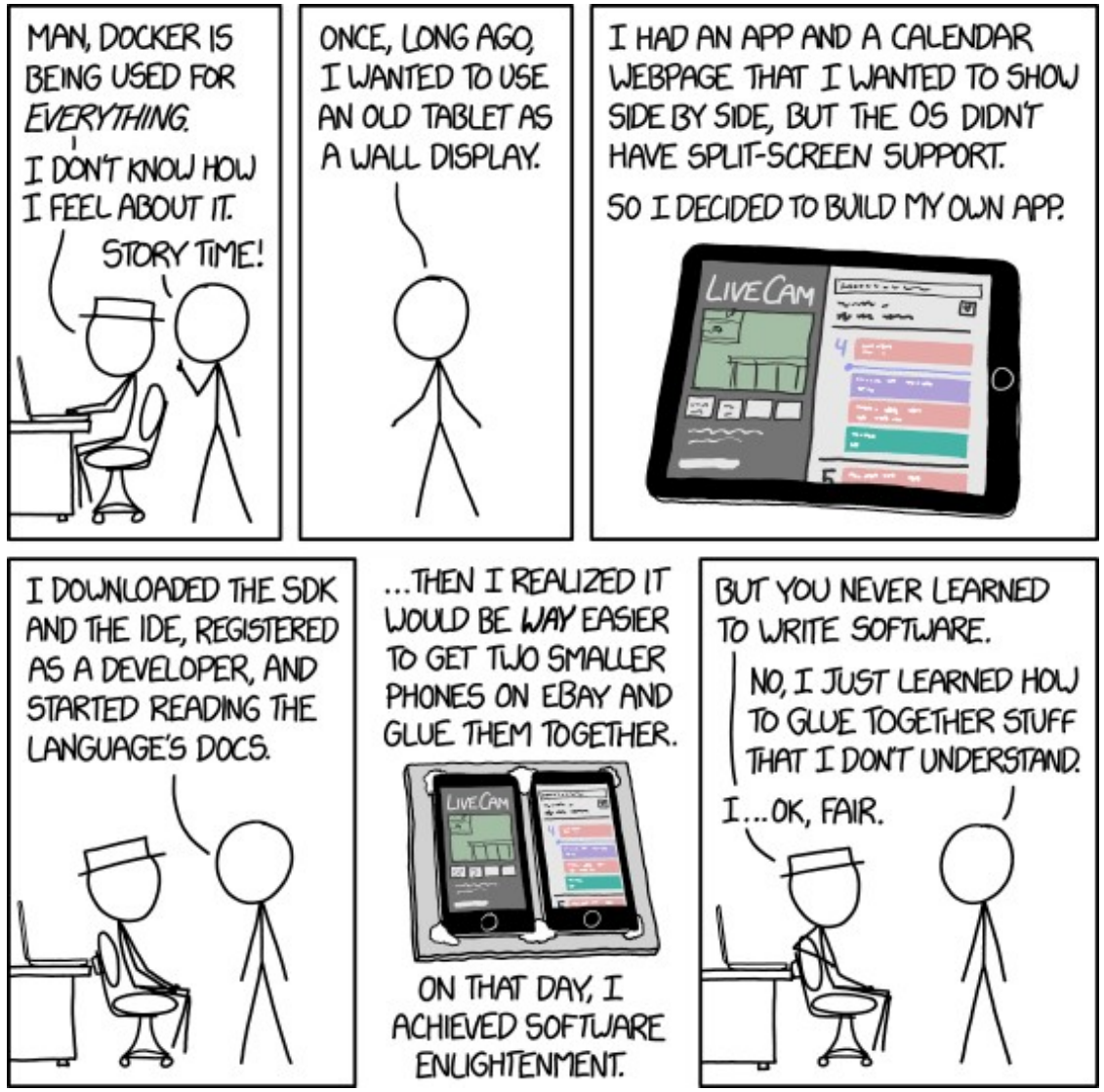
SDS with Ceph – Automate it

- Puppet only installs ceph and places config-files and keys
- We start the OSD by hand, as we like to decide when replication traffic and therefor the impact will happen
- host=”hostname”; chassis=”hostname without last character”; rack=”OSPF Area” root=”datacenter”

```
…
osd crush update on start = true
osd crush location hook = /etc/ceph/osd-location-hook.sh
…
```

```
if [ -z "`echo ${NODE} | egrep '^node[0-9][0-9][0-9][abcd]$'`" ]; then
    echo "host=${NODE} root=default"
else
    eval `echo ${NODE} | perl -nle '/^node([0-9])([0-9][0-9])([abcd])$/; print
"RACK=area$1"; print "CHASSIS=chassis$1$2";'`
    echo "host=${NODE} chassis=${CHASSIS} rack=${RACK} root=${ROOT}"
fi
```

Closing joke



https://imgs.xkcd.com/comics/containers.png

Thank You

Ansgar Jazdzewski <ansgar.jazdzewski@spreadshirt.net>
Bernd Naumann <bernd.naumann@spreadshirt.net>

Addon

Subneting

- Plan your L3 networks well.
  - Subnet-sizes; cut/split them well to fit your needs
  - If you have not done it yet, take the opportunity and start to write an allocation plan, if you start to host “tenants” on your HCI, the need for subnets increases heavily.
  - Infrastructure, Management/OOB, ECMP-Transport, VMs and tenants
- Prepare to have enough space in each subnet even for the yet unknown.
- Think about routing everything and all the time.

KSMtuned and NUMA

- Ksmtuned try to save some memory by deduplicate the memory-pages
- But by default it ignore your NUMA Units → bad performance

```
…

echo 0 > /sys/kernel/mm/ksm/merge_across_nodes

…
```