OpenStack for KVM with Ubuntu on IBM z Systems

OpenStack and Ubuntu

- **Overview**
  - 45min Content + 15min Q&A
  - Notes also contain the information talked about
- **Abstract:**
  - First, a recap what Infrastructure as a Service (IaaS) is about.
  - After that, an overview of the most important capabilities of OpenStack are shown.
  - Eventually, after having enough context, we take a look at Ubuntu and OpenStack.

Target Audience

- **new(-ish) to OpenStack**

- **evaluating IaaS solutions**

- **user/ops-view (no development)**

- **less technical, more capabilities**

  You will be bored when:

  – you want to see code or actual configuration

  – you want to know how to contribute to OpenStack

Outcome

- **After this session you should know:**

    – enough to ask more questions

        - basic capabilities / use cases
        - project names and their focus
        - basic topologies
        - basic architecture

About

- **Markus Zoeller**

- **OpenStack developer since 04/2014**

- **Mostly working on a project called Nova**

  - Bug report management for Nova for >1y

  - Config option documentation cleanup

  - Libvirt KVM driver support for z Systems

  - 3rd party CI temp

  - ...

https://de.linkedin.com/in/markus-zoeller

Part I

Cloud Recap

Context

Average Joe users care about
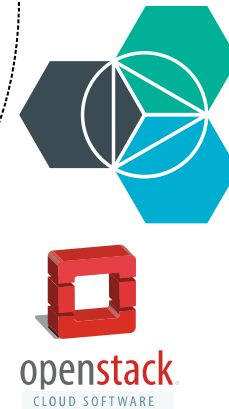
Application developers care about

Software as a Service (SaaS)

Platform as a Service (PaaS)

Infrastructure as a Service (IaaS)

Providers care about

IaaS vs. Traditional

	IaaS	Traditional IT
Consumption Model	On Demand Self Service	Request to Resource Owner
Scaling	out	up
Life cycle	"Cattle"	"Pets"
Reliability	via software	via hardware (+ firmware)
Applications	expect outage	rely on HW HA
Hardware	mass-produced/cheap	premium/expensive

* That's the black/white view. Things can be mixed.

Part II

OpenStack
Overview

Terms and Definitions

Glossary (1/6)

Instance | *A (virtualized) operating system booted from an image. Synonyms are guest, virtual machine, virtual server.*

Glossary (2/6)

Image

A file which packages a bootable operating system in one of many supported formats (*.iso, *.raw, *qcow2, ...).

Glossary (3/6)

Flavor | *Some kind of "T-shirt" size to pre-define buckets of CPU, Memory, disk resource an instance is allowed to consume.*

Glossary (4/6)

Volume

A slice of persistent block storage. Images can be deployed into them. After that, you can launch from that volume.

Glossary (5/6)

Driver | *A (hypervisor or storage or network) backend specific implementation. This enables a common behavior for the user without knowing backend details.*

Glossary (6/6)

Node

An operating system (like) entity which is capable of executing processes. Synonyms could be host or server. That includes bare metal, virtual machines, containers (lxc, lxd, ...).

Disclaimers

- **Whenever I use the term "supports", it means "it is technically working" (if not stated otherwise).**

Demo Video

watch the video at: https://vimeo.com/214680377


```
$ openstack server create MyInstance \
    --image ubuntu-zesty-s390x \
    --flavor m1.tiny
```

Capabilities

OpenStack Capabilites (1/6)

- **Well-defined, stable, versioned, REST API**
- **Role Based Access Control (per resource)**
- **(Nested) multi-tenancy**
- **Cached authentication tokens**
- **Common CLI + API bindings (Python)**
- **Event notifications (Push: CRUDL)**
- **Supports different authentication backends (LDAP, …)**
- **Quotas (limit consumption)**

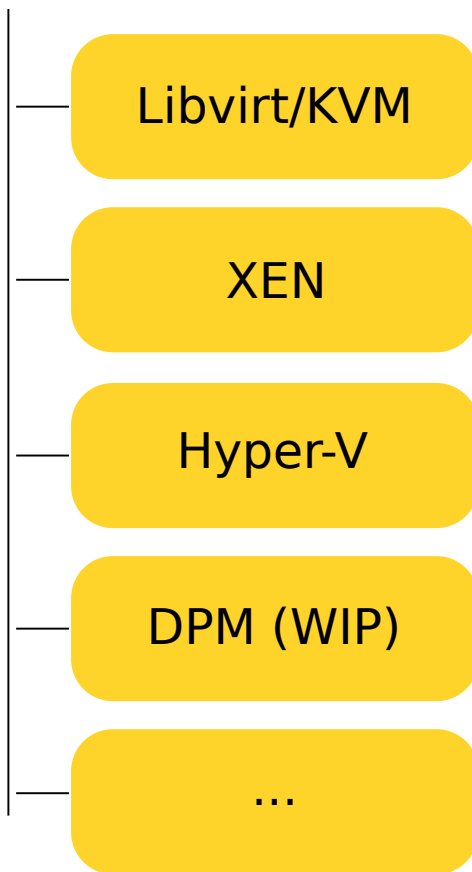# OpenStack Capabilites (2/6)

- **Allows many different configurations but uses sane defaults**

- **2 major releases per year (+ fix releases)**

- **Rolling upgrades to new releases**

- **Defined upstream processes**

  - deprecation policies

  - release notes with upgrade impact

  - governance, milestones, feature freeze, …

  - Vulnerability Management Team

OpenStack Capabilites (3/6)

- **Software Defined Networking (with many backends)**
- **Software Defined Storage (with many backends)**
- **Object Storage (Swift)**
- **Image Repository (Glance)**
- **Key Manager (Barbican)**
- **Advanced Scheduling/Placement**
- **Instance customization on-the-fly with a config-drive**
- **Many more projects which build upon the base**
  - DBaaS (*Trove*),
  - Application Lifecycle Orchestration Engine (*Heat*),
  - ...

The *Drivers* concept

Nova
- Libvirt/KVM
- XEN
- Hyper-V
- DPM (WIP)
- ...

Neutron
- Open vSwitch
- Linux Bridge
- ...

Cinder
- LVM
- IBM Storwize
- EMC VMAX
- ...
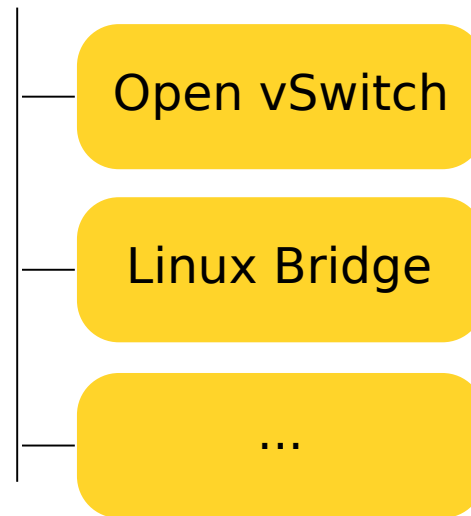
OpenStack Capabilites (4/6)
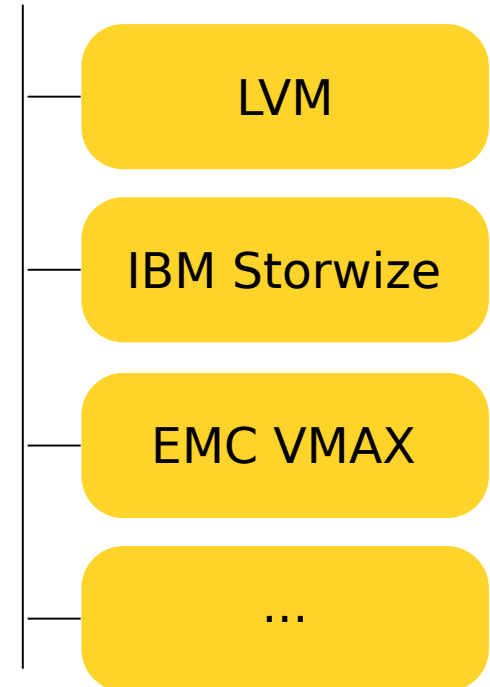
- **Supports all major virtualizer (VMware, Xen, KVM/qemu, zVM, PowerKVM, PowerVM, ...)**

- **Supports bare metal and containers too**

- **Heterogeneous (arch, virt) setups supported**

- **Designed for distributed setups (incl. multi-site)**

- **No Vendor-lockin / strong community**

- **Big company involvement (IBM, Rackspace, RedHat, Canonical, Mirantis, ...)**

OpenStack Capabilites (5/6)

- **You can take influence (summits, mid-cycle, PTG, operators WG, User group, ...)**

- **Discussions are 100% in the open (there are no secrets)**

- **Heavy focus on quality (reviews, automated tests, ...)**

- **Many user docs (HA guide, Config guide, Architecture guide, Ops guide, Security guide, ...)**

OpenStack Capabilites (6/6)

- **Packaged in distros Ubuntu, RHEL, SUSE, Debian, Fedora**

- **Support in common CCM tools: Puppet, Chef, Ansible, ...**

- **Support for common ops tools: syslog, sosreport**

The OpenStack core projects
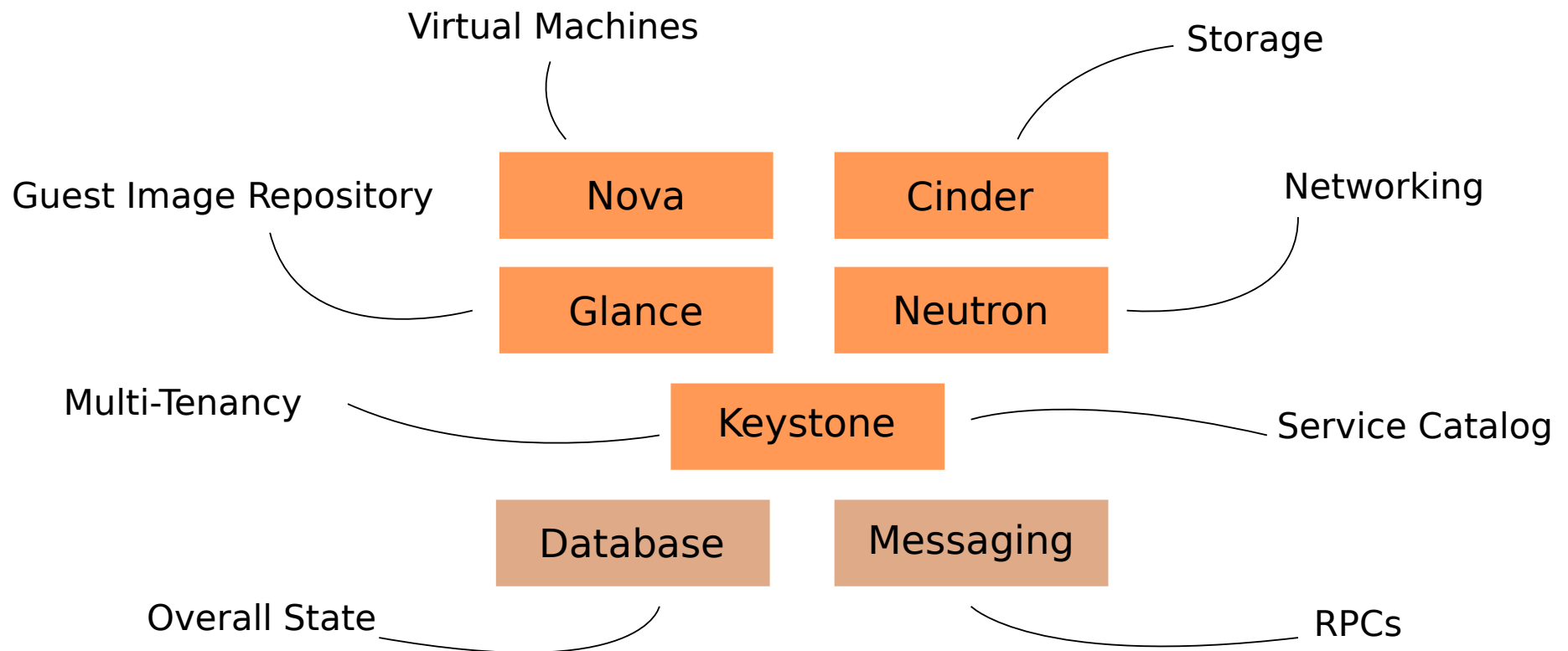
Topology (nodes and their roles)

Controller Node

Networking Node

glance-api

nova-api

...

glance-...

nova-conductor

nova-scheduler

Database

Messaging

neutron-api

neutron-...

cinder-api

cinder-...

cinder-volume

Storage

Storage Node

nova-compute

nova-compute

· · ·

Compute Nodes
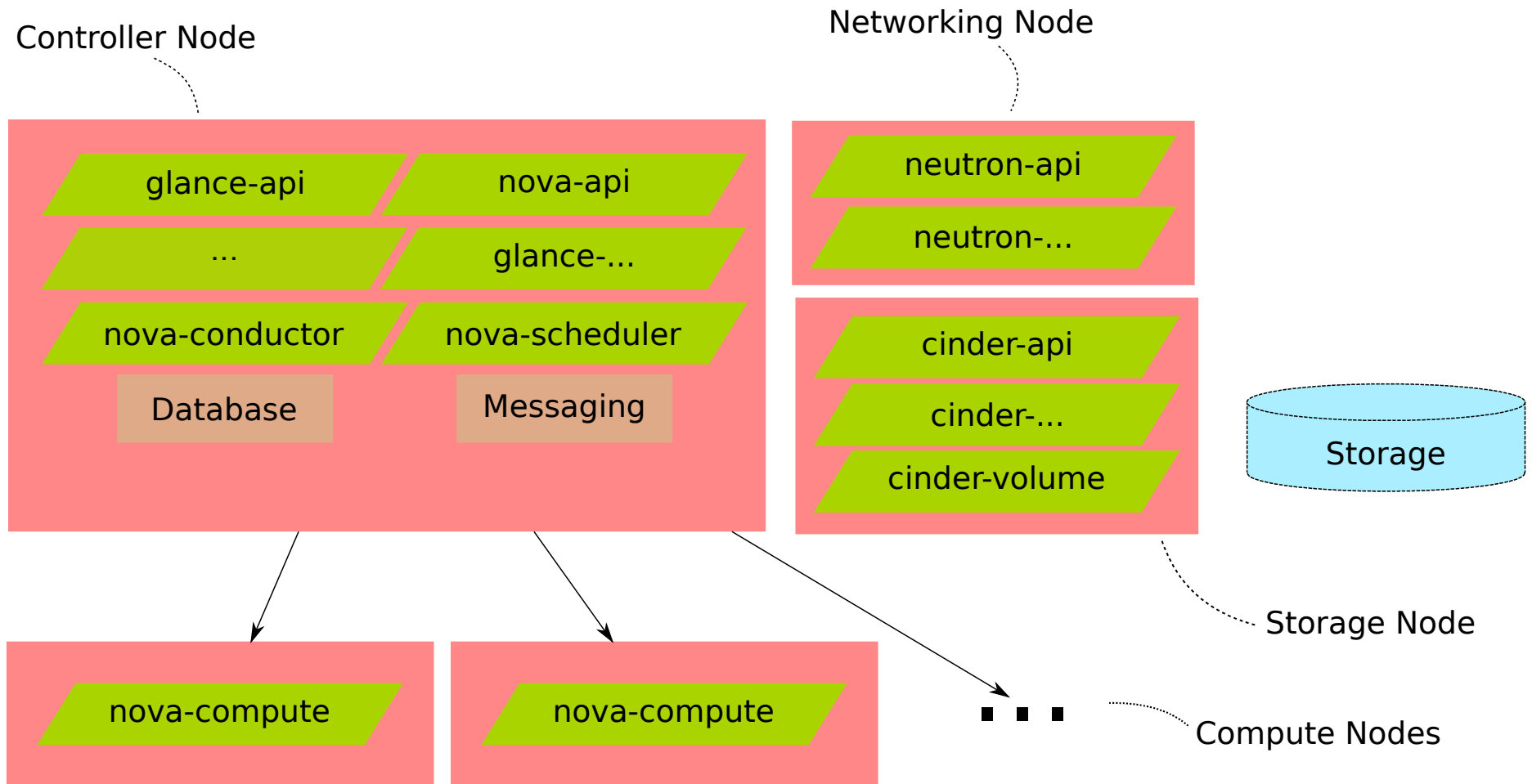
Pain Points

"OpenStack (is perceived) as …

- **… complicated"**
  - mostly true, it's complex (for ops and devs)
  - but very easy for users to consume
- **… hard to upgrade"**
  - yes, but since *Mitaka* it's a lot easier
  - gets improved with every release
- **… having many bugs"**
  - yes, *low* and *medium* prio (patches need a trade-off)
  - but (the very few) *high* and *criticals* get fixed very fast
- **… dead"**
  - the hype might be
  - but there are more and more production clouds

side note: adoption of releases

Which releases are deployments at all stages using?

Adoption of OpenStack's releases shows more deployments using Mitaka or Newton than older releases (Liberty and prior). Note that more than one answer was accepted per deployment so the percentage of users on each release adds to more than 100%.

Figure 3.4

October 2015 n = 348   April 2016 n = 318   October 2016 n = 233   April 2017 n = 479

source: https://www.openstack.org/assets/survey/April2017SurveyReport.pdf
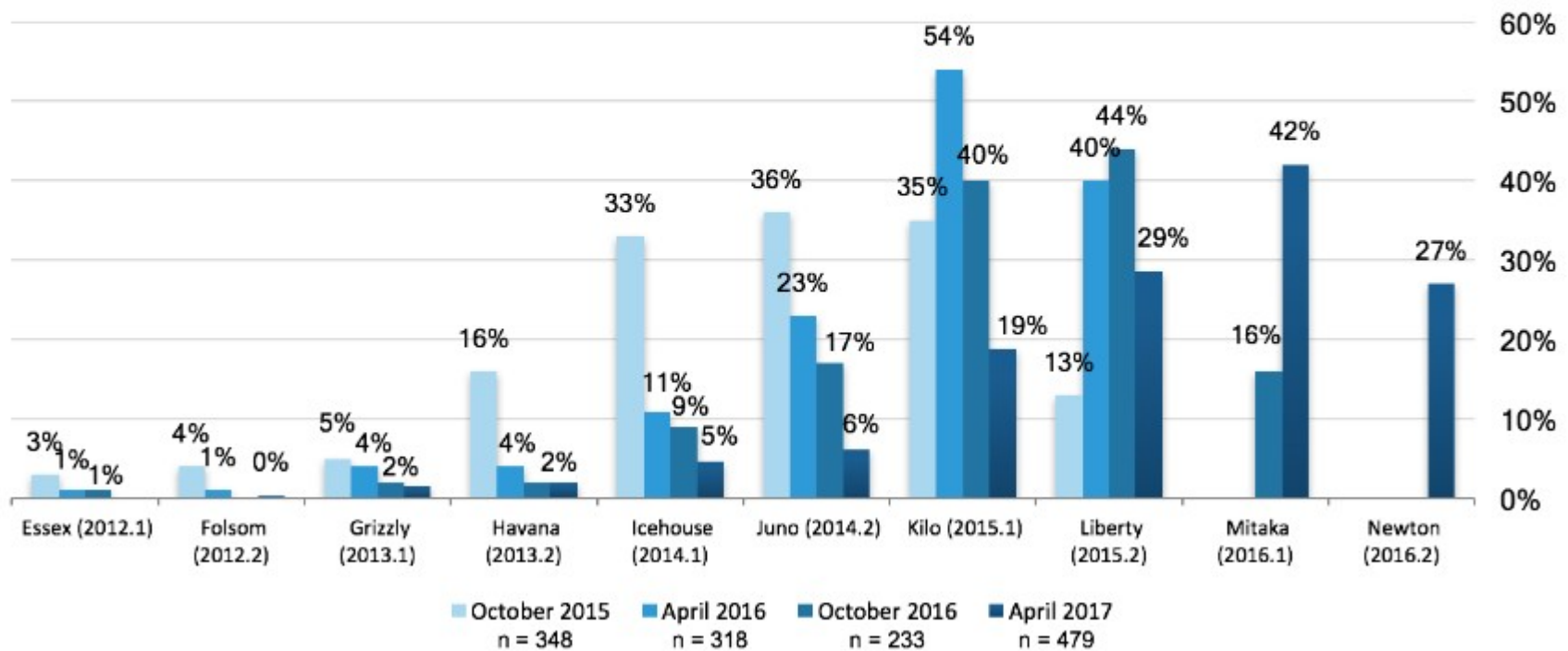
side note: hype cycle
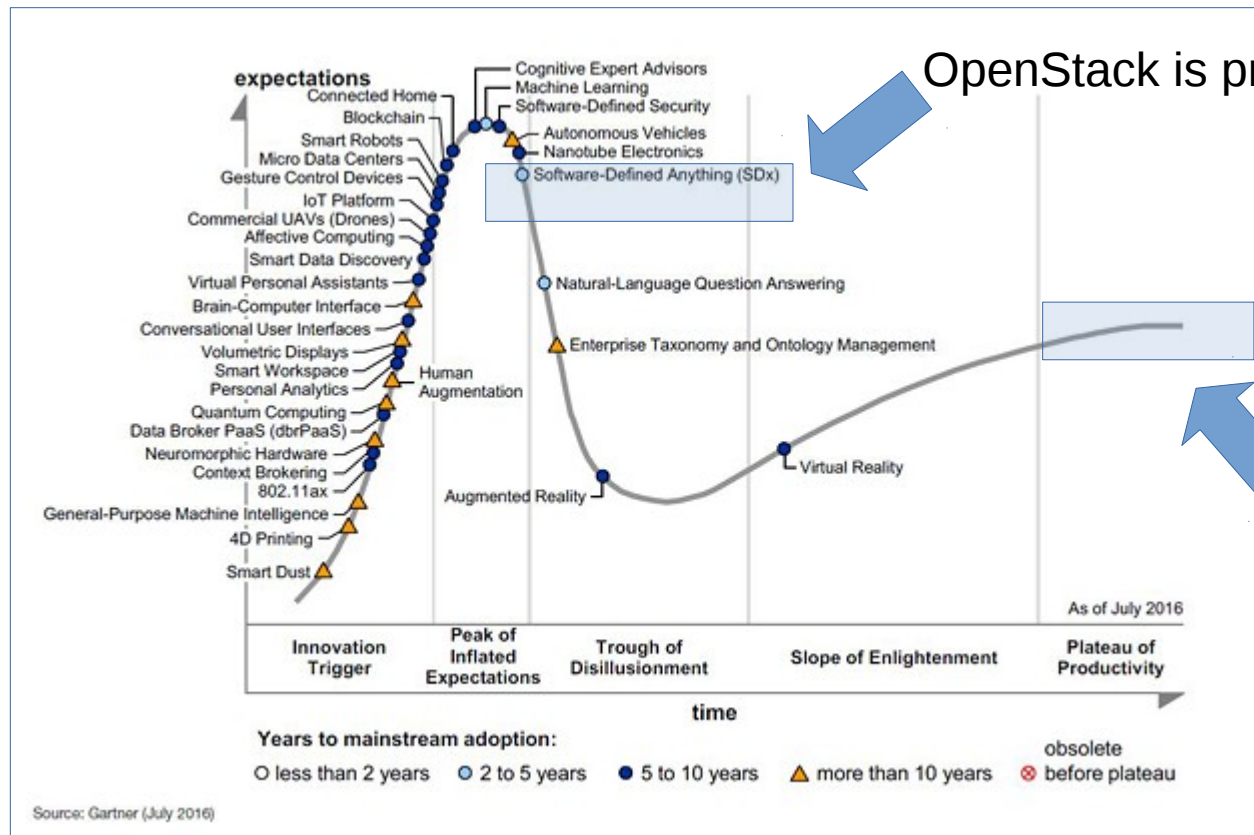
OpenStack is probably here

With a high chance off getting here

source: http://www.gartner.com/newsroom/id/3412017

Part III

Ubuntu + OpenStack

Ubuntu + z Systems

- **Ubuntu is supported as a:**
  - KVM instance within *KVM for z Systems**
  - native install in an LPAR
  - Container (Docker, LXC, LXD)
  - z/VM instance
  - KVM instance within Ubuntu on an LPAR

source: Frank Heimes (canonical)
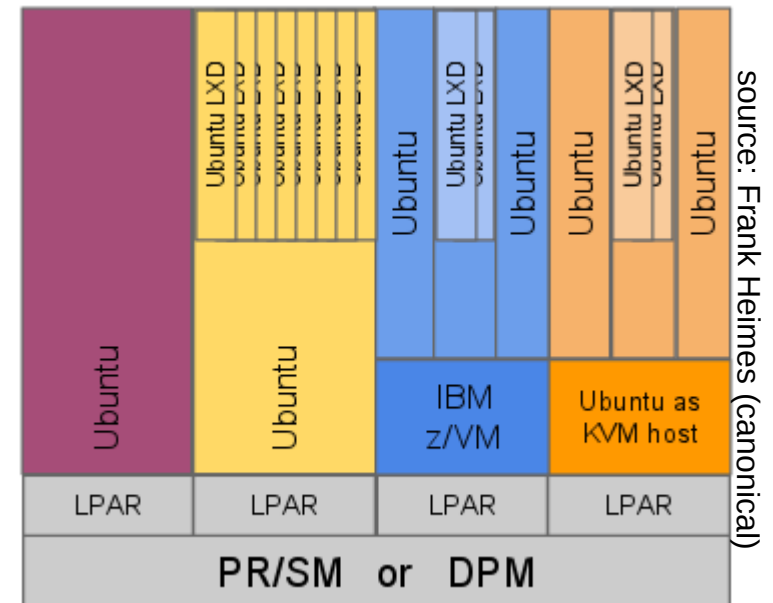
* KVM for IBM z Systems Version 1.1.2 will be the last release delivered by IBM.
  https://www-03.ibm.com/systems/z/solutions/virtualization/kvm/

z Systems + OpenStack

Hypervisor	OpenStack	Supported OpenStack products
KVM on z	>= Mitaka	Ubuntu OpenStack (Ubuntu)
z/VM	>= Liberty	CMA* (IBM) + SUSE OpenStack Cloud
PR/SM in DPM mode	>= Ocata	N/A (opensource)

*This time I'm talking about
a service & support structure.*

 * *Cloud Manager Appliances* (CMA):
   component of IBM z/VM 6.3 or higher with CMA host OS embedded
   http://www.vm.ibm.com/sysman/osmntlvl.html

Upstream OpenStack release EOL

- **2 stable releases + "master" branch**
- **EOL at ~12 Monate after release**

Series	Status	Initial Release Date	Next Phase	EOL Date
Queens	*Future*	TBD		TBD
Pike	Under Development	TBD		TBD
Ocata	Phase I – Latest release	2017-02-22	Phase II – Maintained release on 2017-08-28	2018-02-26
Newton	Phase II – Maintained release	2016-10-06	Phase III – Legacy release on 2017-08-28	2017-10-11
Mitaka	Phase III – Legacy release	2016-04-07	End of Life	2017-04-10
Liberty	EOL	2015-10-15		2016-11-17
Kilo	EOL	2015-04-30		2016-05-02
Juno	EOL	2014-10-16		2015-12-07

source: https://releases.openstack.org/

Ubuntu OpenStack release EOL

KVM on z
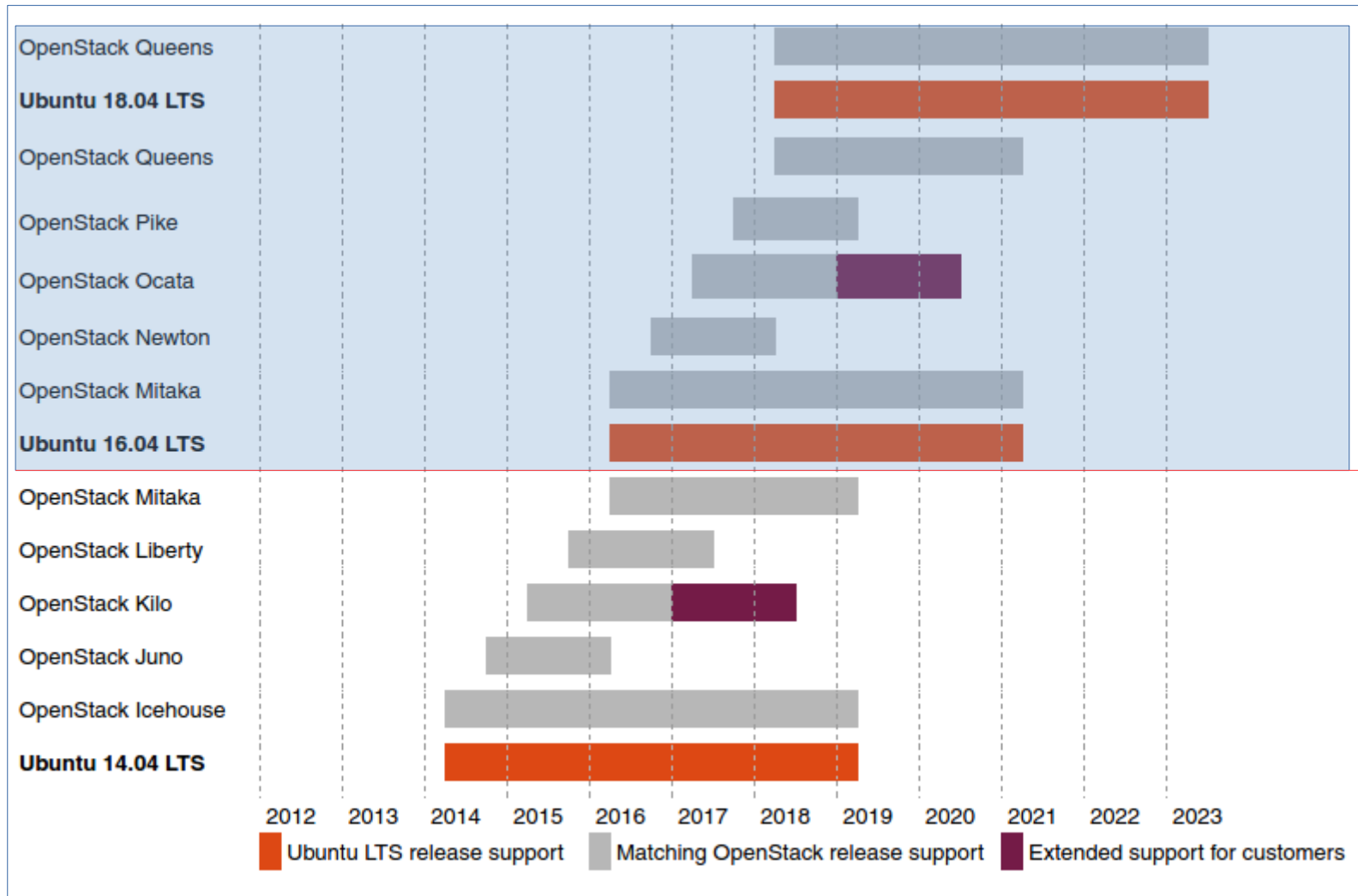
source: https://www.ubuntu.com/info/release-end-of-life

Ubuntu Automation Tools

- **LXD (v2.x)**
  - Container technology built on top of LXC
  - Package your (micro-)services
- **Juju**
  - Service modeling (with LXD) across servers
  - Deploy and configure with *"Charms"*
- **MaaS (TBD for z)**
  - "Metal as a Service"
  - Physical servers management

Complementary Technologies
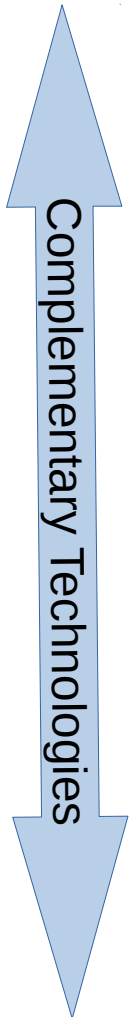
Benefits

- **Use x and z mixed setups with the same control plane**
  - x: cattles
  - z: pets / cattles
- **Hide z complexity for users but use the features under the cover**
- **Enable IaaS with z resources**
- **Rely on strong partners: Canonical + IBM**

Q&A

The next slides contain the questions and answers of the session from 2017-04-26.
I slightly rephrased the questions to make them easier to read and understand but tried to keep them as close to the original as possible.

Question 1

Q: *"Is the reason this session mentions Ubuntu, because that is the only distro with a KVM that is 'supported' (in the support & service sense)? I think SLES and RHEL on z also have a working KVM but it may be considered a 'technical preview' without support by the distro."*

A: *Ubuntu is the only distro which support OpenStack with KVM on an IBM z Systems machine. That's why this session is focused on Ubuntu. SUSE supports KVM (without OpenStack) on IBM z Systems machines with SLES12 SP2[1]. There is no support for either KVM or OpenStack in RHEL on IBM z Systems machines at the moment.*

References:
[1] https://www.suse.com/communities/blog/virtualization-suse-kvm-support-ibm-z-systems-linuxone/

Question 2

Q: *"What are the minimal requirements to set up an OpenStack environment? How many LPARs, OSA ports, Storage, IFLs do I need to start. Can it all run on z System?"*

A: Assumed you want to create a minimal PoC and only want to spent the minimal amount of resources to get something running, I'd recommend to have 1 LPAR (which means we have an "all-in-one" topology), 8GB Memory, 20GB storage, 2 IFLs, 1 OSA port. The Ubuntu OpenStack base [1] gives a few hints. Document [2] gives hints about upper limits (for the sake of completeness). You may also be interestet in capacity planning[3] and install guidance[4]. OpenStack on LXD needs more resources[5]. It can run all on z Systems. More info about a setup at github[6].

References:
[1] https://jujucharms.com/openstack-base/
[2] http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/PRS5331
[3] https://docs.openstack.org/ops-guide/ops-capacity-planning-scaling.html
[4] https://docs.openstack.org/mitaka/install-guide-ubuntu/environment.html
[5] https://docs.openstack.org/developer/charm-guide/openstack-on-lxd.html
[6] https://github.com/ubuntu-openstack/zopenstack/blob/master/README-lpar-single.md

Question 3

Q: *"KVM on z base on Ubuntu - is this the KVM product that's shipped by IBM?"*

A: The names can easily be confused. Whenever I used "KVM on z" in the slides, I meant the KVM technology running in *Linux for IBM z Systems*. The IBM product name is "*KVM for IBM z Systems*"[1]. When I referenced the KVM in an Ubuntu distro, I try to consistently use the term "Ubuntu KVM host".

References:
[1] https://www-03.ibm.com/systems/z/solutions/virtualization/kvm/

Question 4

Q: *"Is the OpenStack Controller running on System z?"*

A: Yes, it works on both, x and z (no difference in the usage).

Question 5

Q: *"I am just wondering on how OpenStack gets implemented for handling z Systems. Are you using XCAT API or a Linux LPAR Installation with OpenStack enabled? How do you get OpenStack to provision the Linux guests on z?"*

A: The "Nova z/VM virt driver"[1] does all the XCAT calls to the underlying z/VM hypervisor and is mainly used by the z/VM Cloud Managed Appliance (CMA). However, the "Nova Libvirt driver"[2] uses Libvirt to communicate with KVM. Libvirt and KVM have upstream support for z Systems. That's the abstraction done by the OpenStack "driver" concept. For the sake of completeness, for PR/SM DPM virtualization, OpenStack communicates with the HMC REST API[3] (by using the "zhmcclient" library[4]).

References:
[1] https://github.com/openstack/nova-zvm-virt-driver/blob/master/nova_zvm/virt/zvm/driver.py
[2] https://github.com/openstack/nova/blob/master/nova/virt/libvirt/driver.py
[3] https://github.com/openstack/nova-dpm/blob/master/nova_dpm/virt/dpm/driver.py
[4] https://pypi.python.org/pypi/zhmcclient/

Question 6

Q: *"Is OpenStack a prereq for Docker?"*

A: No. But it's useful to have IaaS (for example "OpenStack") to scale out your SaaS solutions (for example delivered with "Docker"). For the sake of completeness, there is also an OpenStack project called "Magnum"[1] which handles (among others) "Docker Swarm".

References:
[1] https://wiki.openstack.org/wiki/Magnum

Question 7

Q: *"Should we separate the compute node and the controller node (on different LPARs)?"*

A: For production, yes, definitely. One LPAR as control plane and one for every new compute node. See slide 28, the pink boxes would be LPARs. For PoC environments, you're good to go with 1 LPAR where the control plane and 1 compute node is combined.

Question 8

Q: *"In the demo video, was the OpenStack controller running on z or x?"*

A: It was on x because (in my role as developer) I have that setup laying around on my laptop when developing patches, before I test that patch in an IBM z Systems environment. But you won't see any difference when it runs on z.

Question 9

Q: *"Using VRealize as the web front end, I understand it would need to interface with the OpenStack controller...is this right?"*

A: Yes. To be precise, the interface is the REST API[1] running on the controller node.

References:
[1] https://developer.openstack.org/api-guide/quick-start/index.html

Question 10

Q: *"Every future deployment of Openstack Mitaka on Ubuntu should be done with opensource KVM and not with 'KVM for IBM z Systems'… right ?"*

A: You can order the product "KVM for IBM z Systems"[1] until August 2017. End-of-life will be March 2018[2]. You get patches until the EOL. Personally, I wouldn't recommend to use a product which has an EOL in the near future.
Good to know, the Qemu-kvm code level that is used in latest "KVM for IBM z Systems" (version 1.1.2) got incorporated into Ubuntu 17.04. That kernel will soon also be available for 16.04 as 16.04.3 Ubuntu.

References:
[1] https://www-03.ibm.com/systems/z/solutions/virtualization/kvm/
[2] https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=897/ENUS917-048&infotype=AN&subtype=CA

Question 11

Q: *"Can Ubuntu OpenStack provision LXD containers?"*

A: Yes, OpenStack can instantiate LXD, too. An introduction can be found at [1]. The latest release "16.0.0.0b1" is from end of April 2017[2]. The combination of OpenStack and LXD is supported for all architecture, incl. s390x.

References:
[1] https://insights.ubuntu.com/2015/05/06/introduction-to-nova-compute-lxd/
[2] https://github.com/openstack/nova-lxd/releases

Question 12

Q: "This solution is supported by canonical?"

A: "Ubuntu OpenStack" is commercially supported by Canonical, incl. Ubuntu Server with KVM, all on z. More about plans and pricing at [1].

References:
[1] https://www.ubuntu.com/cloud/plans-and-pricing