OpenShift Container Platform on Z: Performance, Quality & Best Practices

Dr.-Ing. Axel Busch
axel.busch@ibm.com
Technical Lead OpenShift Performance on Z
Linux on IBM z Performance

Trademarks

The following are trademarks of the International Business Machines Corporation in the United States, other countries, or both.

Not all common law marks used by IBM are listed on this page. Failure of a mark to appear does not mean that IBM does not use the mark nor does it mean that the product is not actively marketed or is not significant within its relevant market.

Those trademarks followed by ® are registered trademarks of IBM in the United States; all others are trademarks or common law marks of IBM in the United States.

For a more complete list of IBM Trademarks, see www.ibm.com/legal/copytrade.shtml:

*CICS®, DataPower®, DB2®, e business(logo)®, ESCON, eServer, FICON®, IBM®, IBM (logo)®, IMS, MVS, OS/390®, POWER6®, POWER6+, POWER7®, Power Architecture®, PowerVM®, PureFlex, PureSystems, S/390®, Sysplex Timer®, System p®, System p5, System x®, System z®, System z9®, System z10®, WebSphere®, X-Architecture®, z13®, z13s®, z Systems®, z9®, z/Architecture®, z/OS®, z/VM®, z/VSE®, zEnterprise®, zSeries®, IBM Z®, IBM z Systems®, IBM z13®, IBM z13s®, IBM z14®, IBM LinuxONE

The following are trademarks or registered trademarks of other companies.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both. Open vSwitch and OvS are trademarks of The Linux Foundation.

*** All other products may be trademarks or registered trademarks of their respective companies.**

Notes:

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.

IBM hardware products are manufactured Sync new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

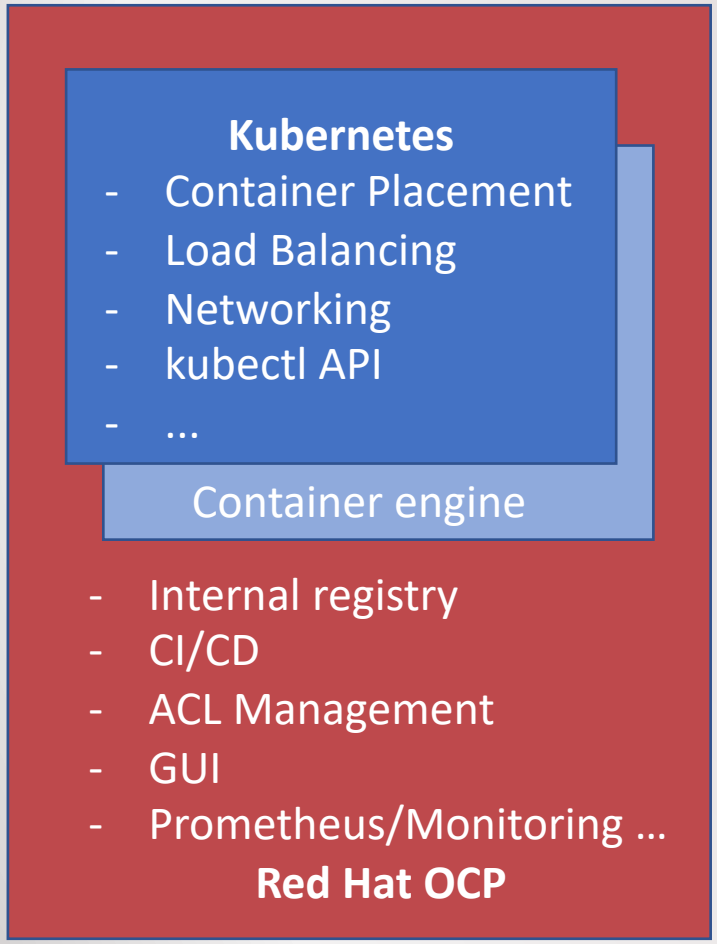
This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Information about non-IBM products is obtained Sync the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

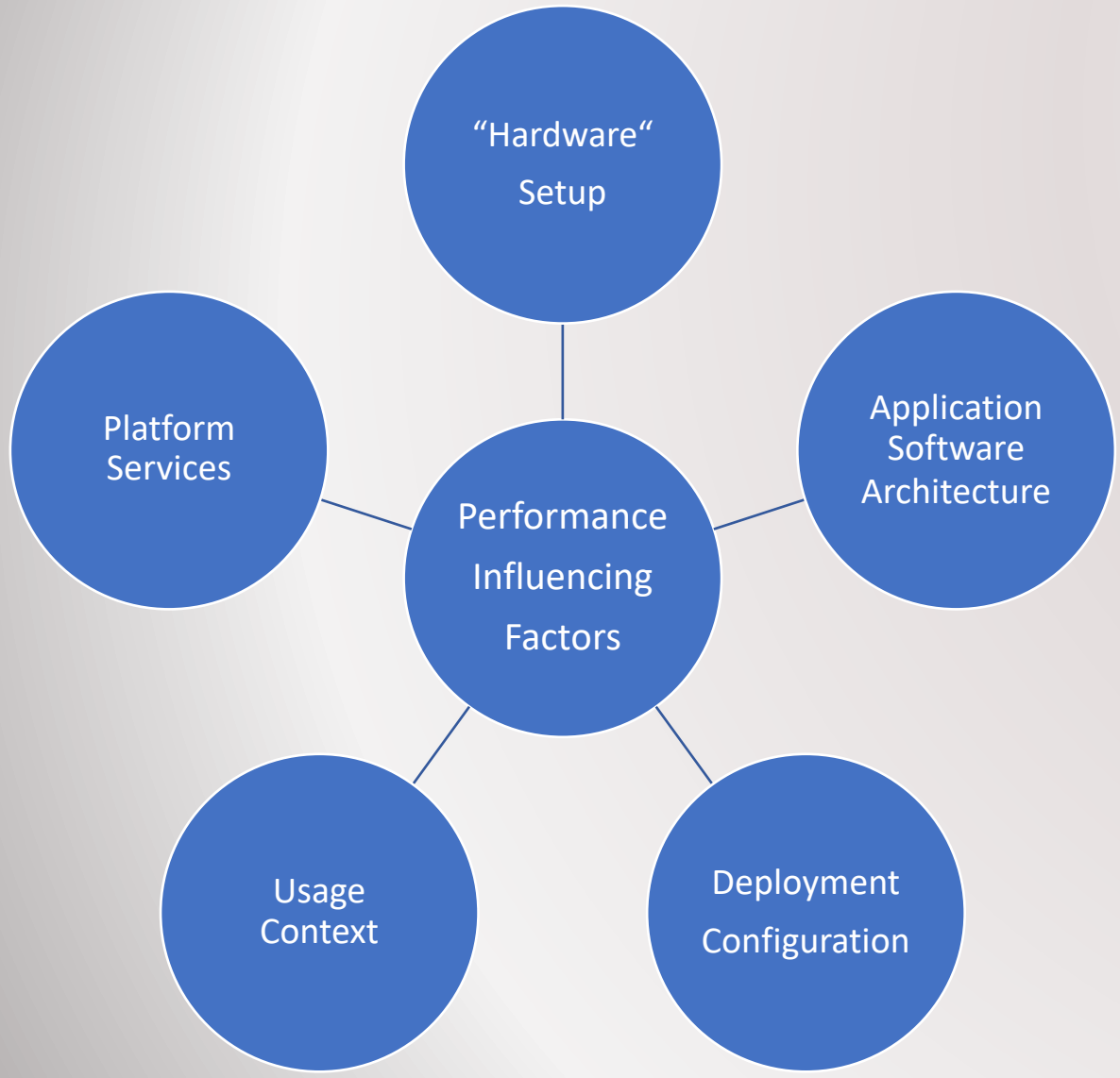
OpenShift container platform in a nutshell



- Kubernetes-based automated container orchestration platform
- Supports DevOps for more efficient digital transformation
  - Tools for agile development of software
  - Automated build and deployment pipeline
- Monitoring stack and security features
- Can be run on physical, virtual and cloud infrastructures

- Kubernetes core features
- Additional tools that can make life easier for daily work of developers
- Supports modern (agile) software development processes

Why do we talk about (OCP) performance?



- CPU/memory/.. resources
- Hypervisor, i.e. z/VM, KVM,…

- Monolythic Application
- Service oriented
- Microservices

- Trade-off decision for deployment
- High locality vs. high distribution

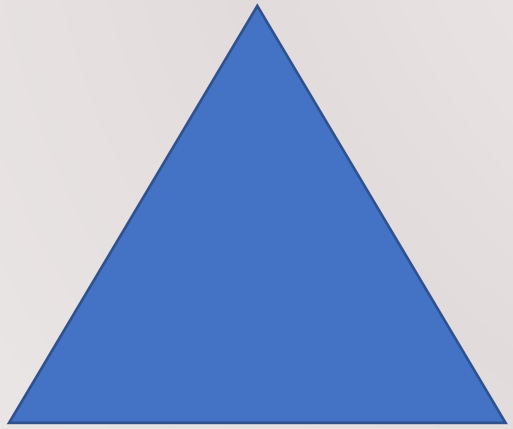
- Workload pattern
- # concurrent users
- Data amount

- OpenShift Layers/Operators
- Software defined network
- Prometheus
- etcd,…

Performance as one of many attributes…

- Requirements often competing
- More requirements = more trade-off decisions

Performance



Cost

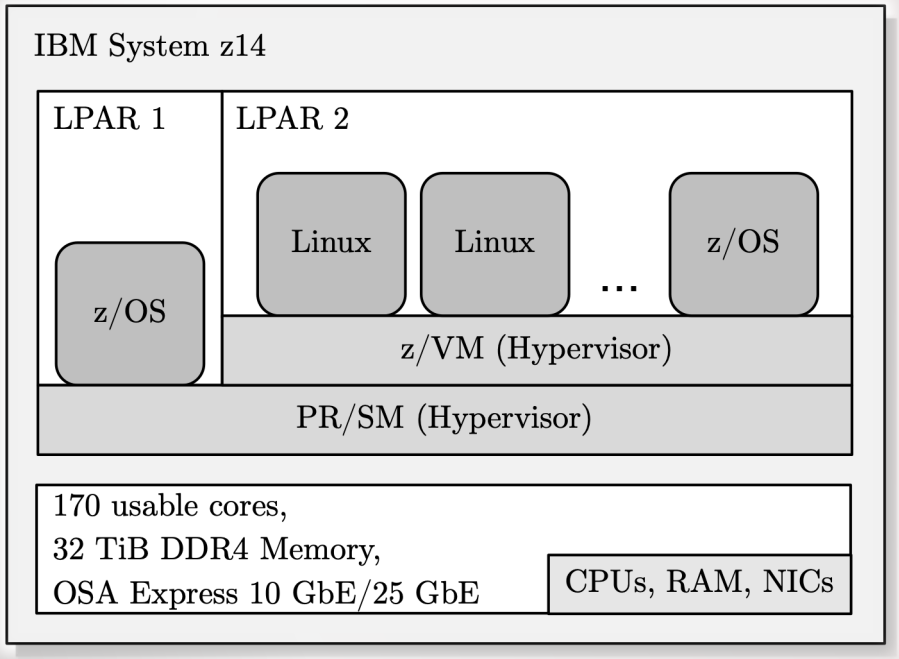
Security

… many more

Based on *Roger Atkinson - Project management: cost, time and quality, two best guesses and a phenomenon, its time to accept other success criteria, International Journal of Project Management, 1999*

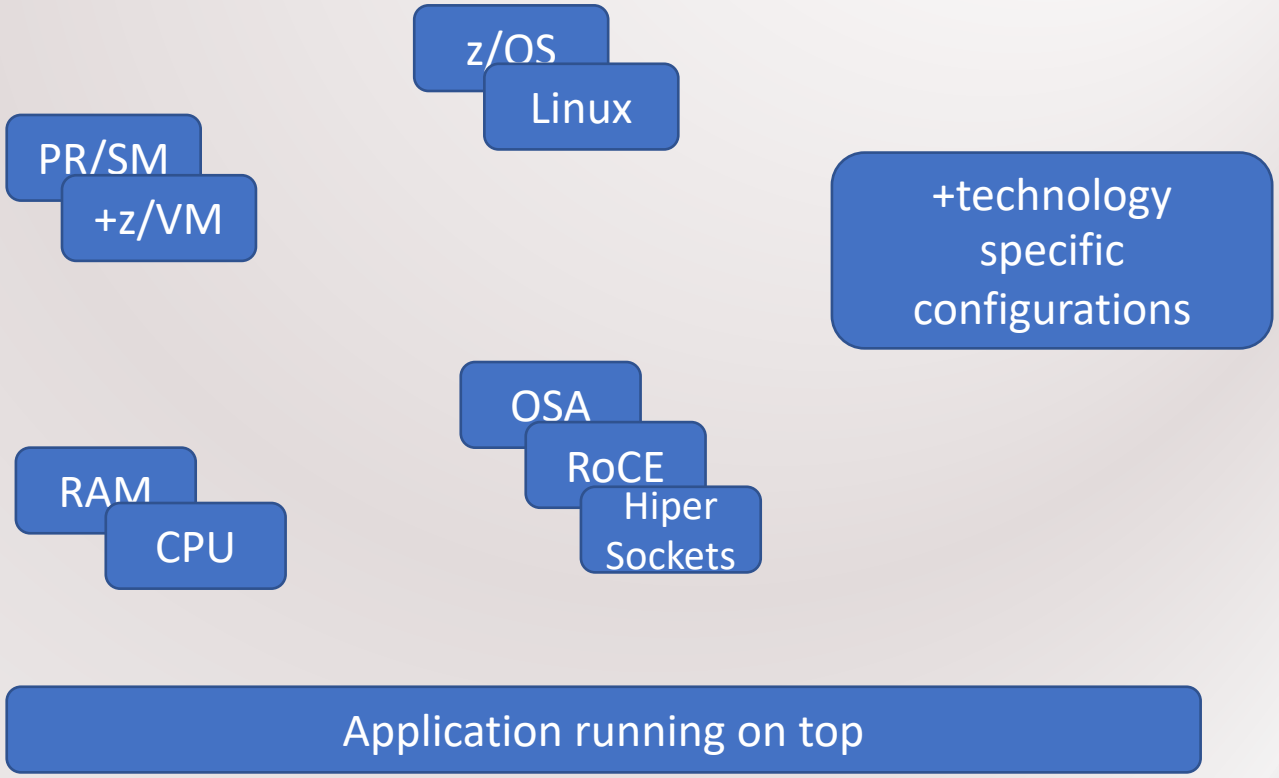
… and what is different to what we do since decades?

IBM z14 as an example:

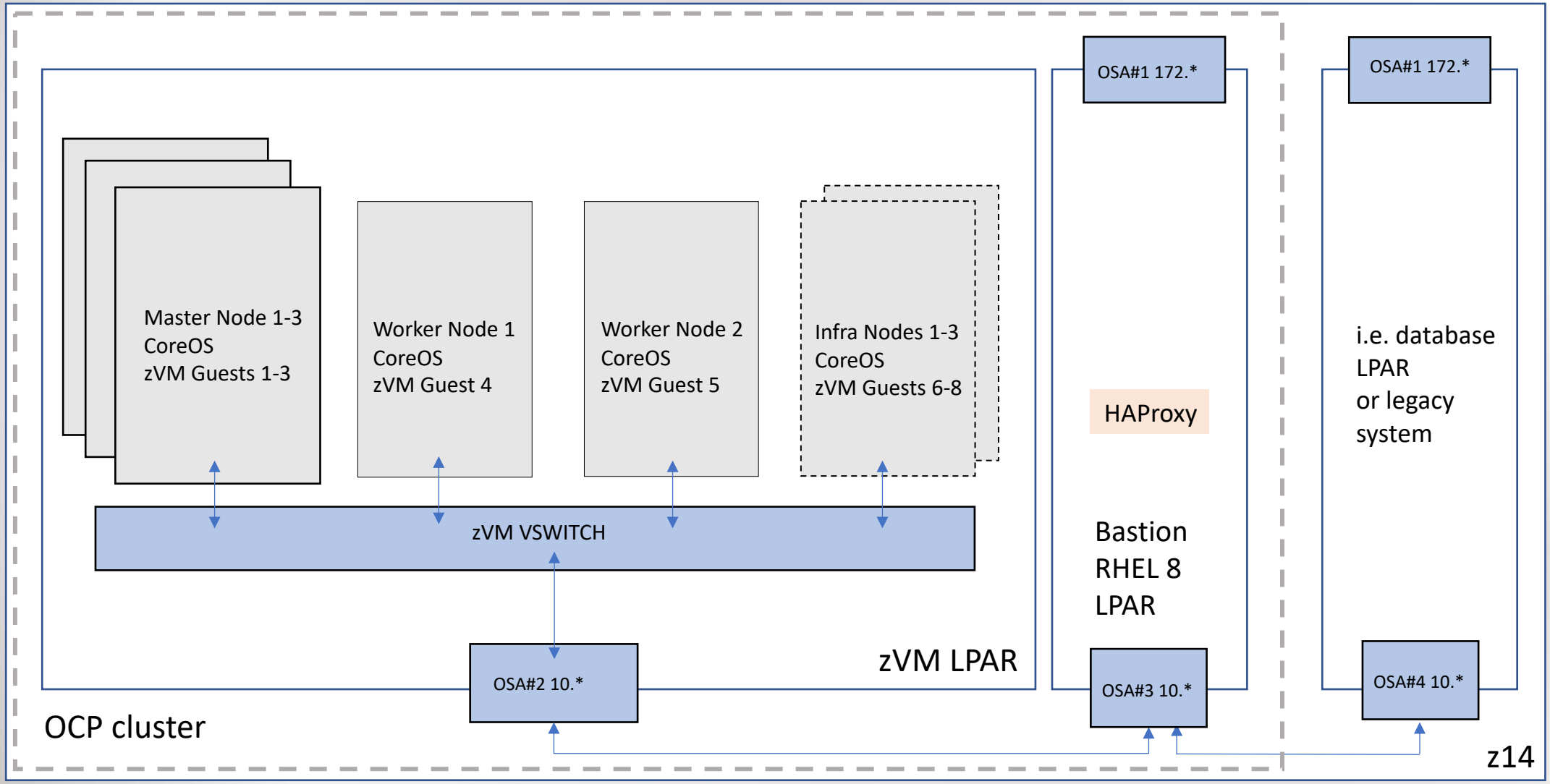


Degree of freedom space

Configuration options



OpenShift (on z) Container Platform: System Architecture example



Again: what is different to what we do since decades?

Before:
- Choose if you want to run a monolythic application
- Create shortcuts to increase speed
- Disable (or just do not enable) logging to decrease CPU consumption

Deciding whether to optimize a quality attribute at the expense of others

Now: OCP prevents to make these decisions in a way that protects other quality attributes, such as
- Maintainability
- Security
- Reliability

But: Quality by design of OCP introduces other degrees of freedom
- Node types, i.e. master, worker, infra nodes
- Operator placement
- Network tunings, i.e. Receive packet steering, receive flow steering, node port,…

And now? More degrees of freedoms? More decisions to be made? Less flexibility?

OCP tends to guide you to **make decisions more thoughtfully**
You **still can optimize** things

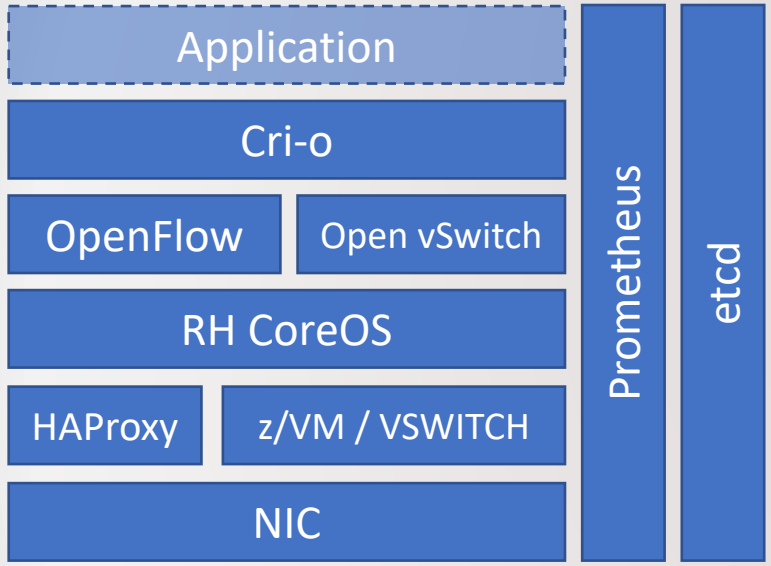
- Optimize the size of nodes and cluster (more workers/bigger workers)
- Optimize network throughput and latency by consuming some CPU cycles
- Define storage, e.g. NFS, OCS/Ceph,…

But it is **harder to accidentally bypass the fundamental quality by design** architecture principles by

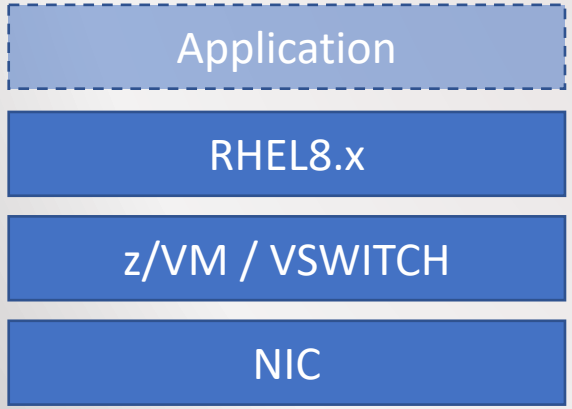
- avoiding to bypass the entire security architecture
- Prometheus logging stack always activated

Comparing what cannot be compared: steady state load of OCP vs. z/VM+RHEL8.1 installation

OCP



RHEL8.x



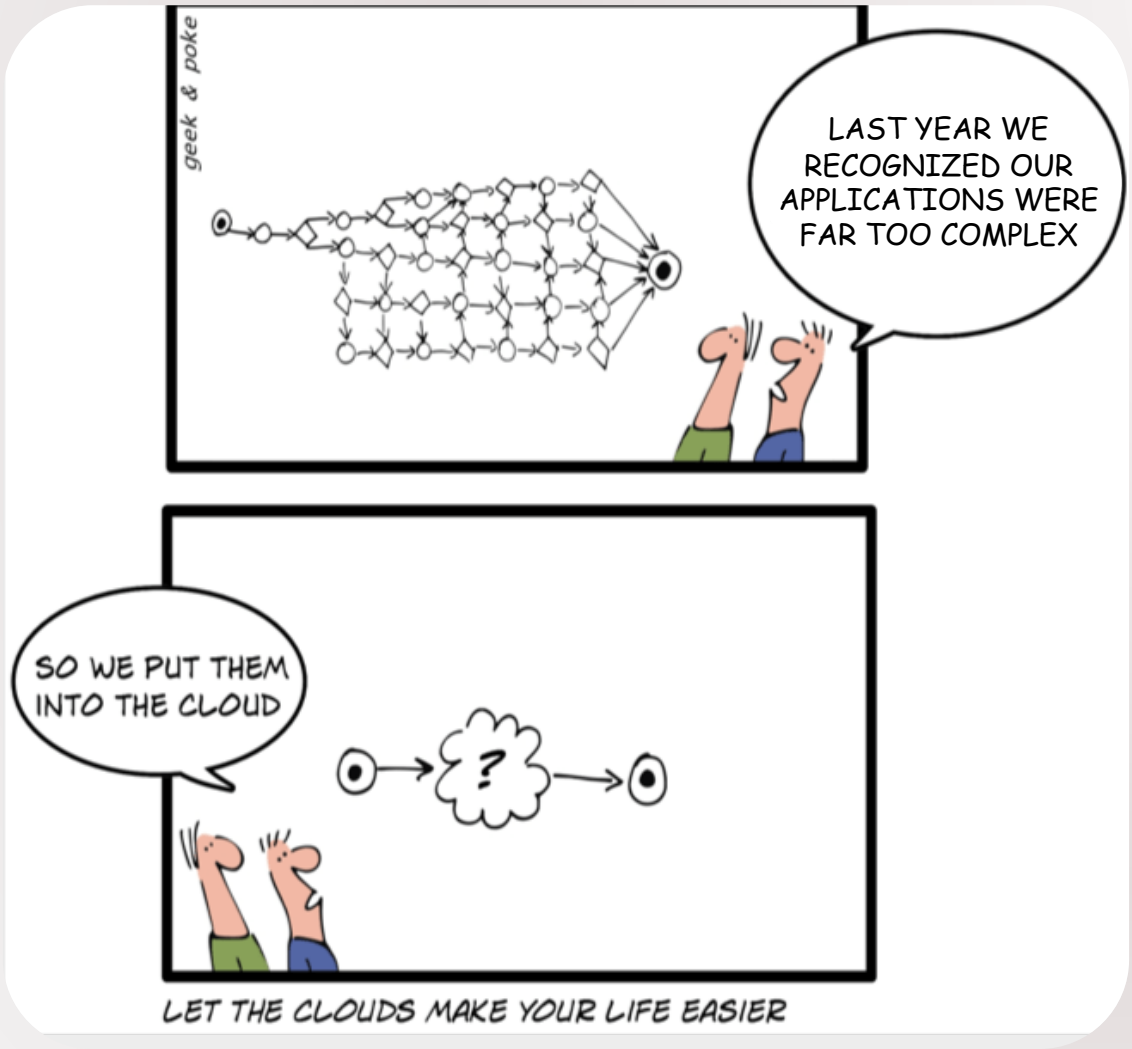
NAMESPACE	NAME
openshift-monitoring	prometheus-k8s-1
openshift-kube-apiserver	kube-apiserver-master-0.boet4244ocp.lnxne.boe
openshift-kube-apiserver	kube-apiserver-master-1.boet4244ocp.lnxne.boe
openshift-monitoring	prometheus-k8s-0
openshift-etcd	etcd-master-0.boet4244ocp.lnxne.boe
openshift-etcd	etcd-master-1.boet4244ocp.lnxne.boe
openshift-marketplace	certified-operators-9lgv5
openshift-operator-lifecycle-manager	packageserver-86598d8494-lk2cn
openshift-marketplace	redhat-operators-jjsw8
openshift-operator-lifecycle-manager	packageserver-86598d8494-25qfg
openshift-etcd	etcd-master-2.boet4244ocp.lnxne.boe
openshift-marketplace	community-operators-2cqtl
openshift-kube-apiserver	kube-apiserver-master-2.boet4244ocp.lnxne.boe

Top consumers:

- **openshift-monitoring**
- **openshift-kube-apiserver**
- **openshift-etcd**

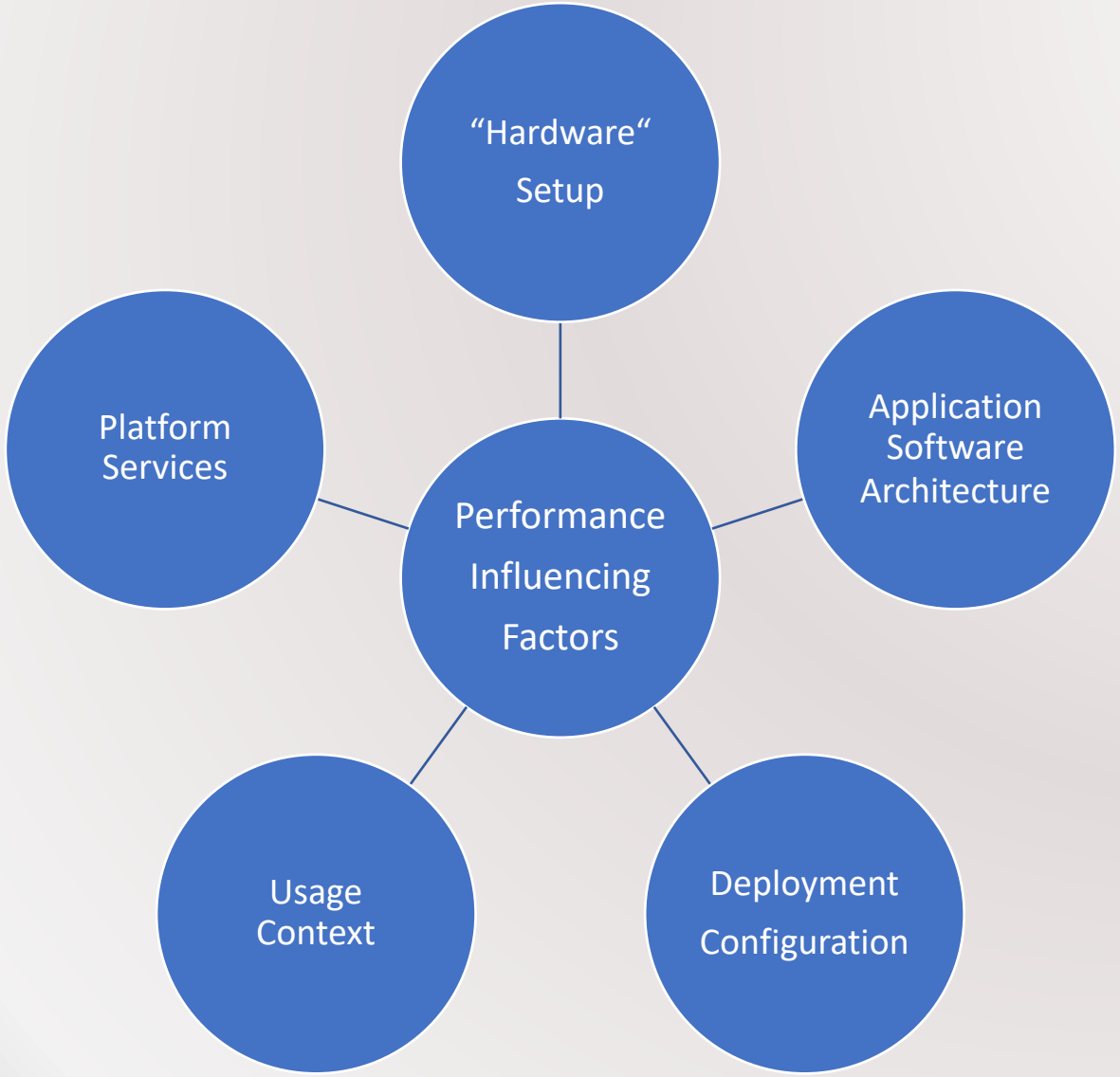
Operators for Reliability, Security, Logging… Keeping service quality high

Cloud Complexity

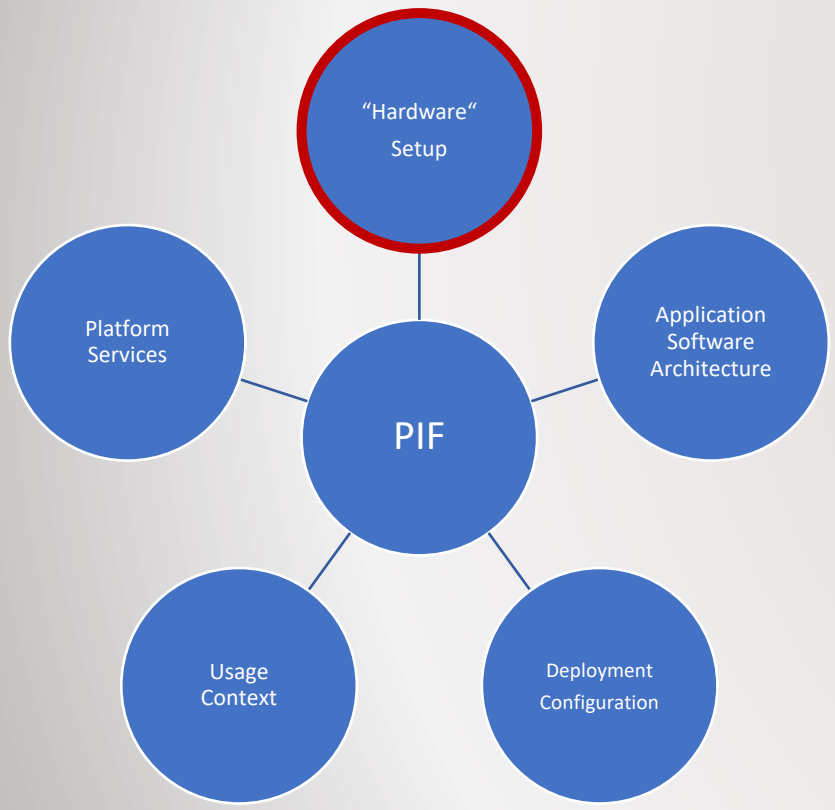


http://geek-and-poke.com/

Tunings addressing performance influencing factors



Tunings addressing performance influencing factors (1)



HyperPAV

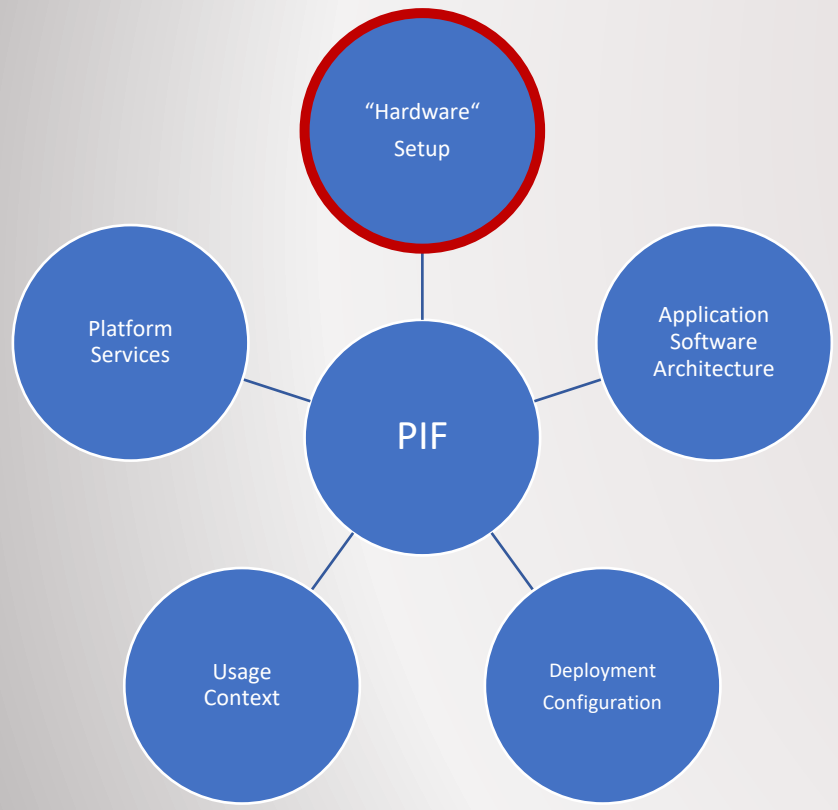
- DASD/ECKD devices
- Useful if workload has many parallel disk accesses
- OCP operators do have many parallel disk accesses
  - Prometheus
  - etcd
- If workloads access overlay filesystem of containers in parallel

- Channel subsystem processes one I/O operation per device
- HyperPAV devices allow parallel processing of I/O operations

- We recommend 5-8 HPAV devices per node

J. Doelle - Scaling HyperPAV alias devices on Linux guests on z/VM

Tunings addressing performance influencing factors (2)

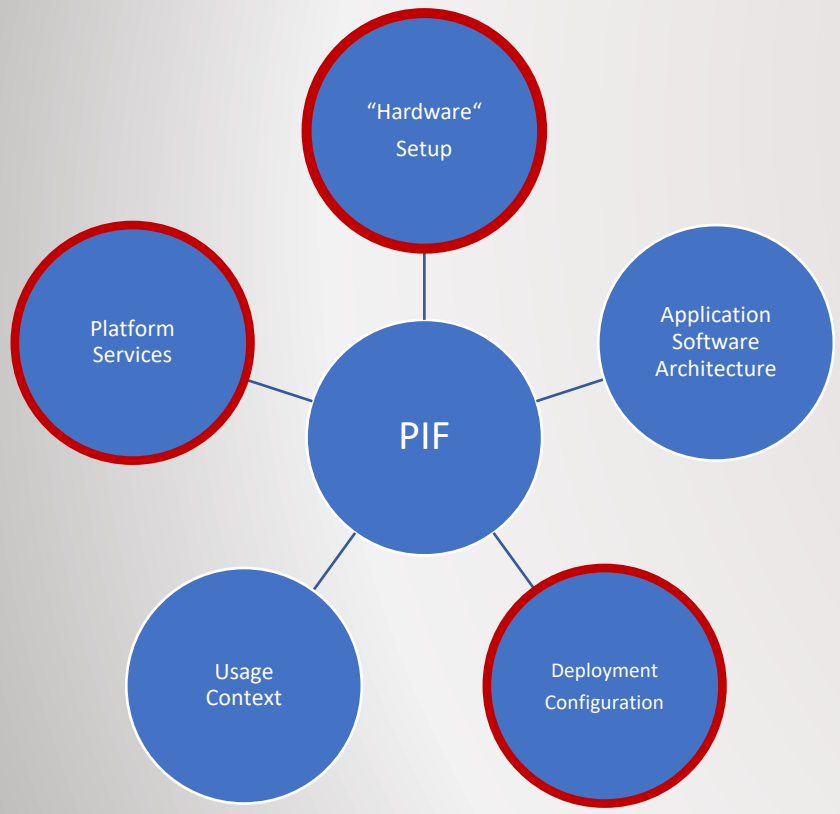


z/VM CPU SHARE

- Control the percentage of processor time a user receives
- With z/VM, a virtual machine receives its proportion of processor time according to its SHARE setting
- Some nodes might need more CPU proportion than others

- Adjusting can improve response time and throughput of workloads

Tunings addressing performance influencing factors (3)

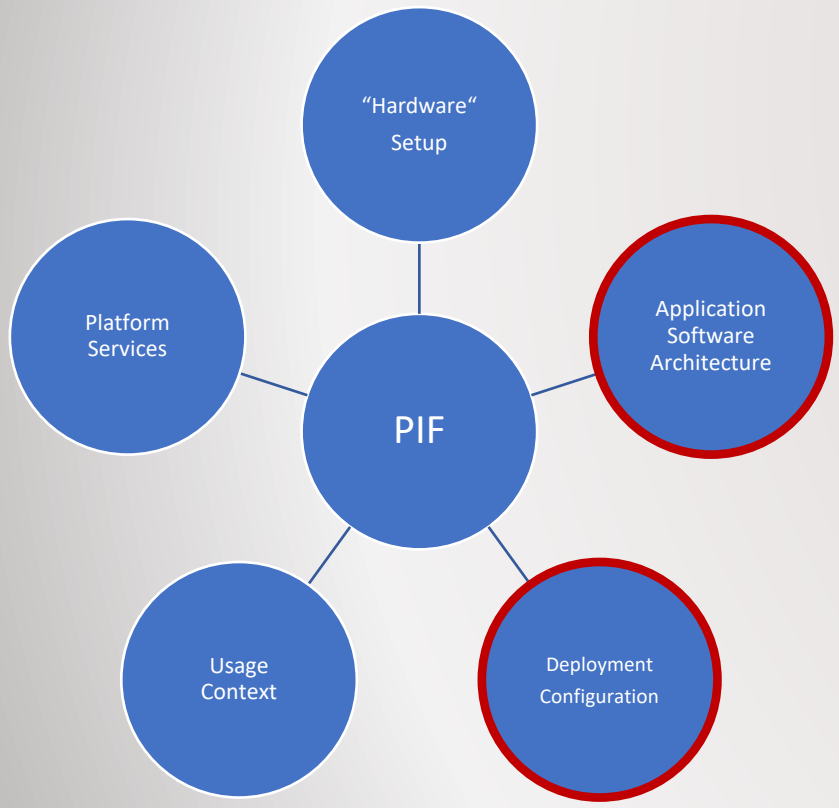


Cluster size #workers

- Trade-off decision between
  - number of workers
  - number of cpus per worker
  - amount of steady state load
- To what extent OCP can schedule pods on workers
- Scalability in high load scenarios (performance)
- Re-deployment in downtime scenarios of single nodes (reliability)

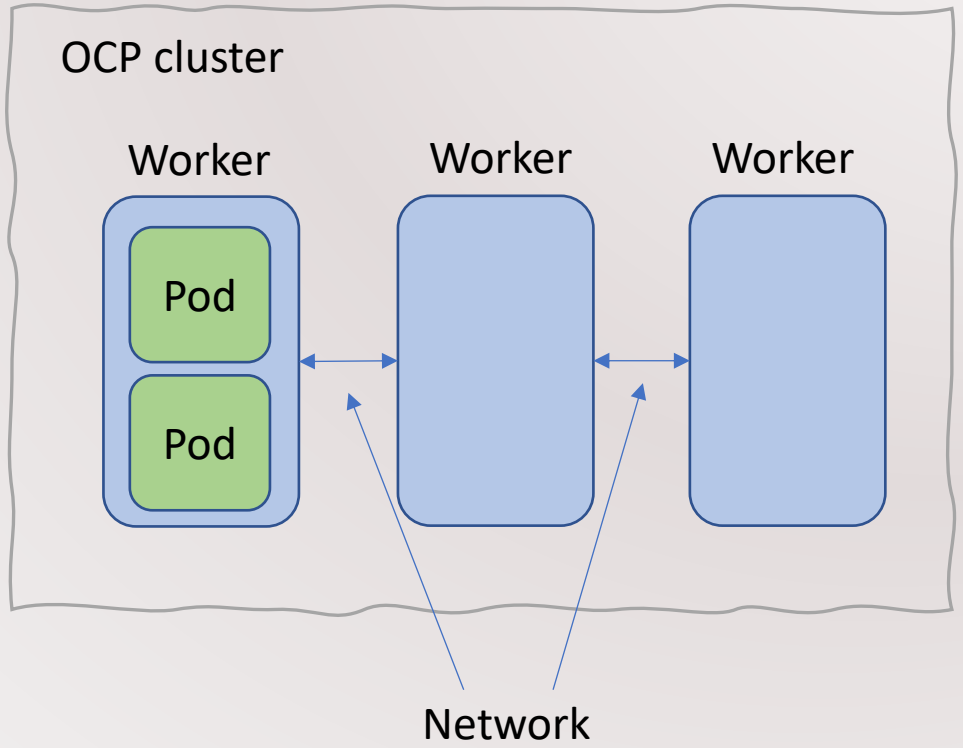
- Both, more workers and more cpus per worker increase steady state load
  - More logging effort
  - More network traffic between nodes

Tunings addressing performance influencing factors (4)

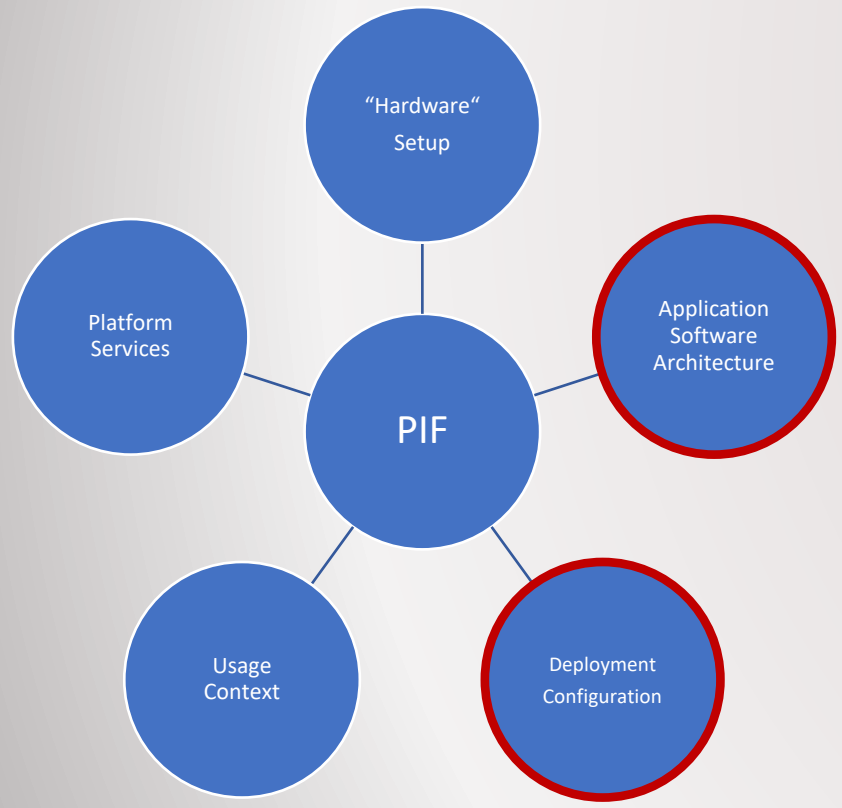


Service locality

1. Co-located

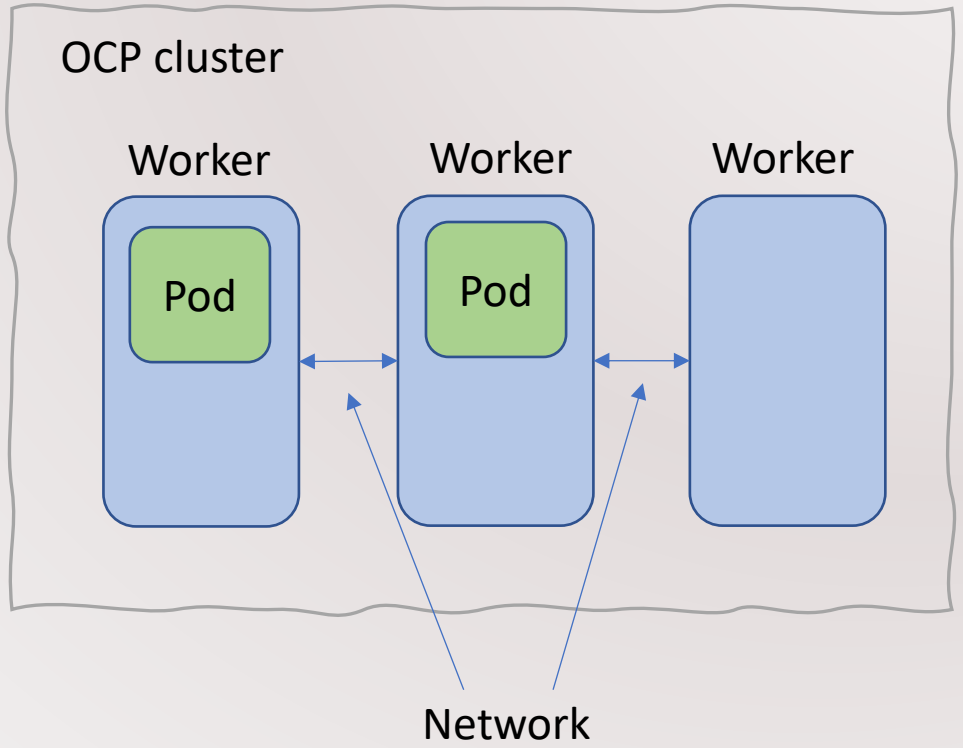


Tunings addressing performance influencing factors (4)

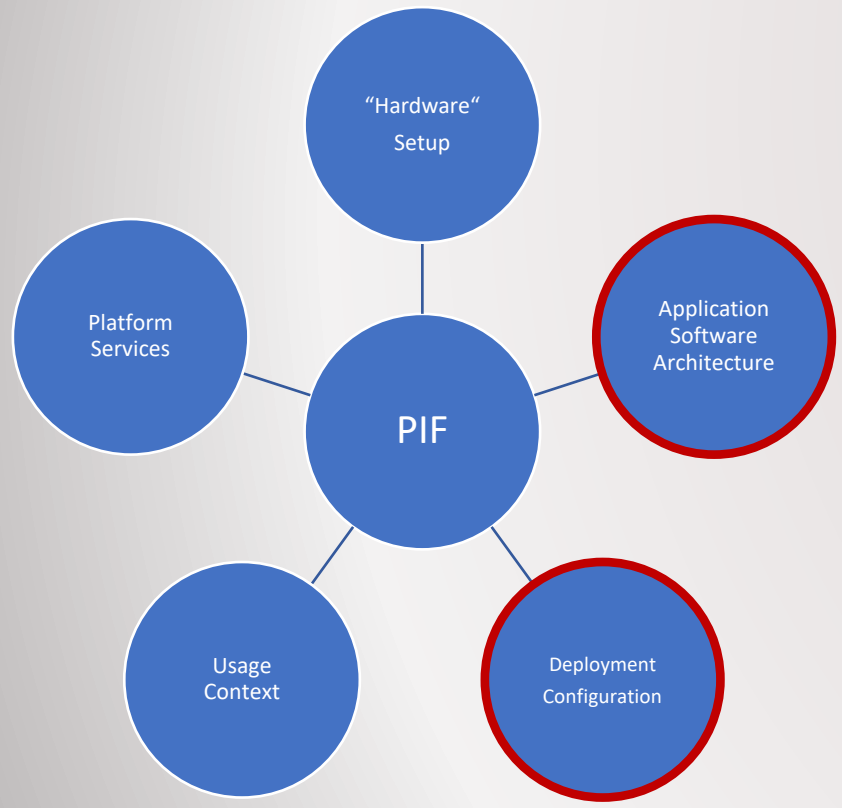


Service locality

2. Non-colocated

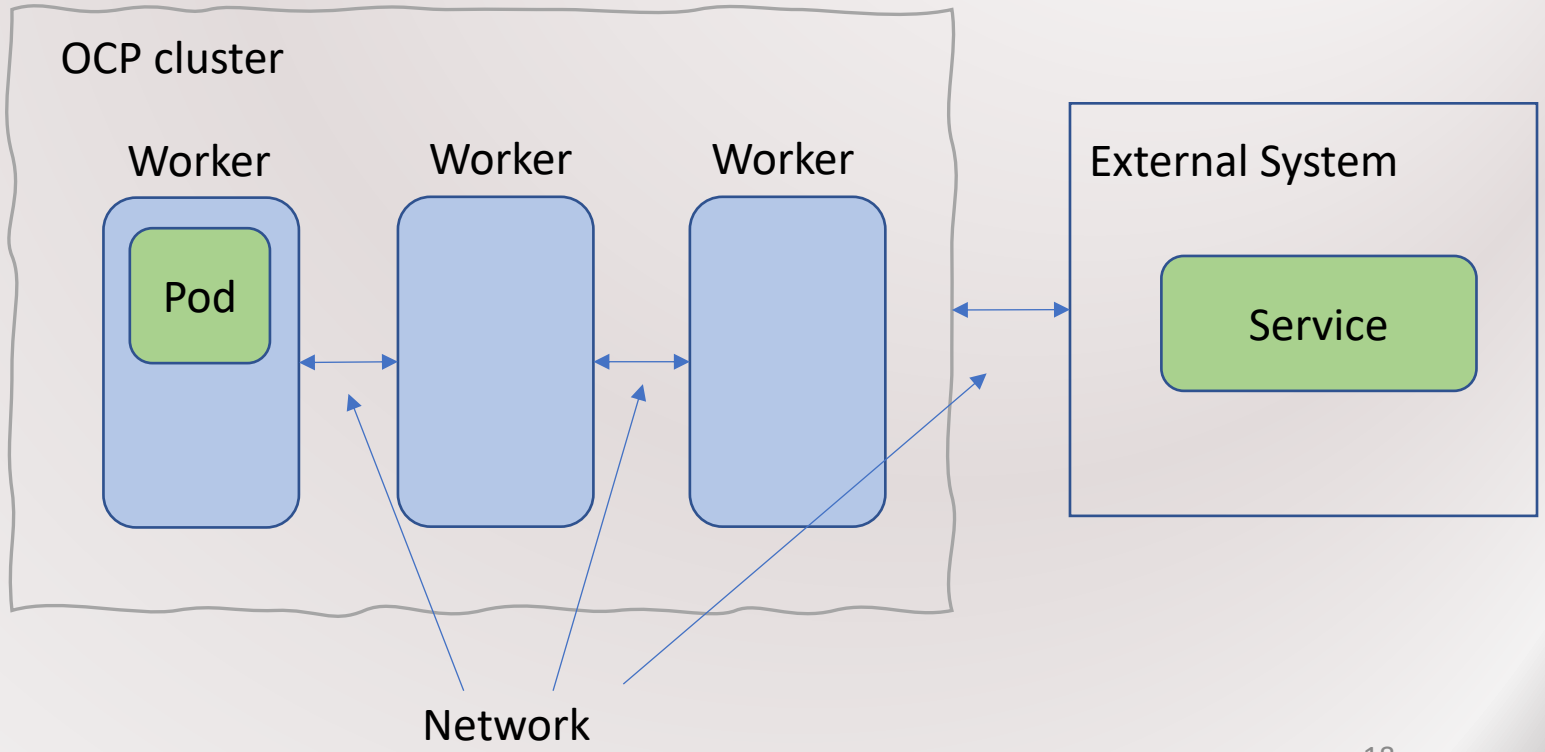


Tunings addressing performance influencing factors (4)

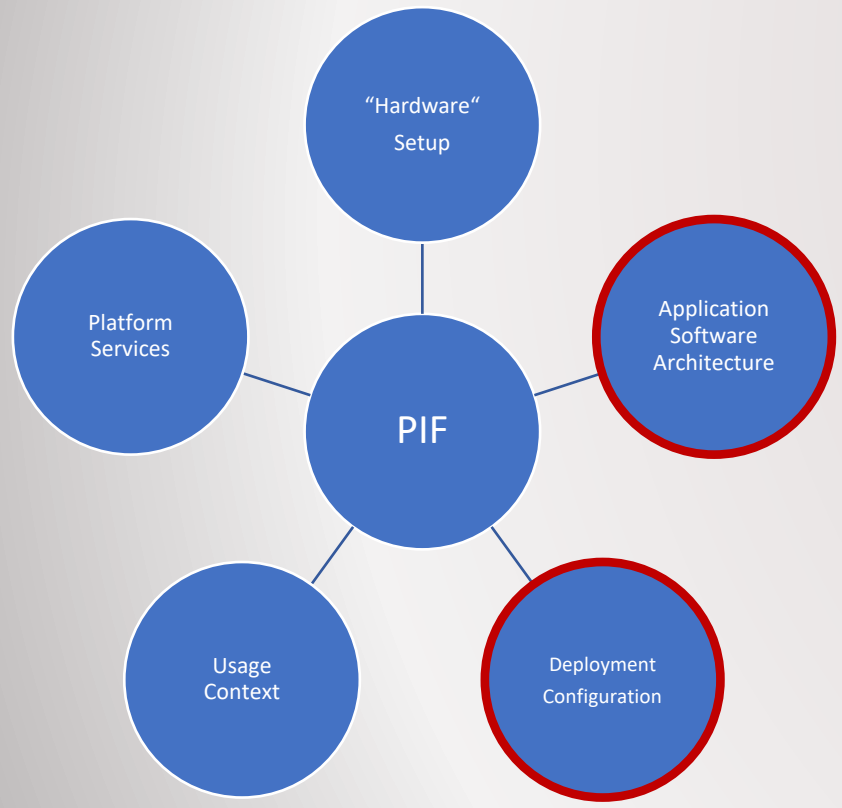


Service locality

3. External service



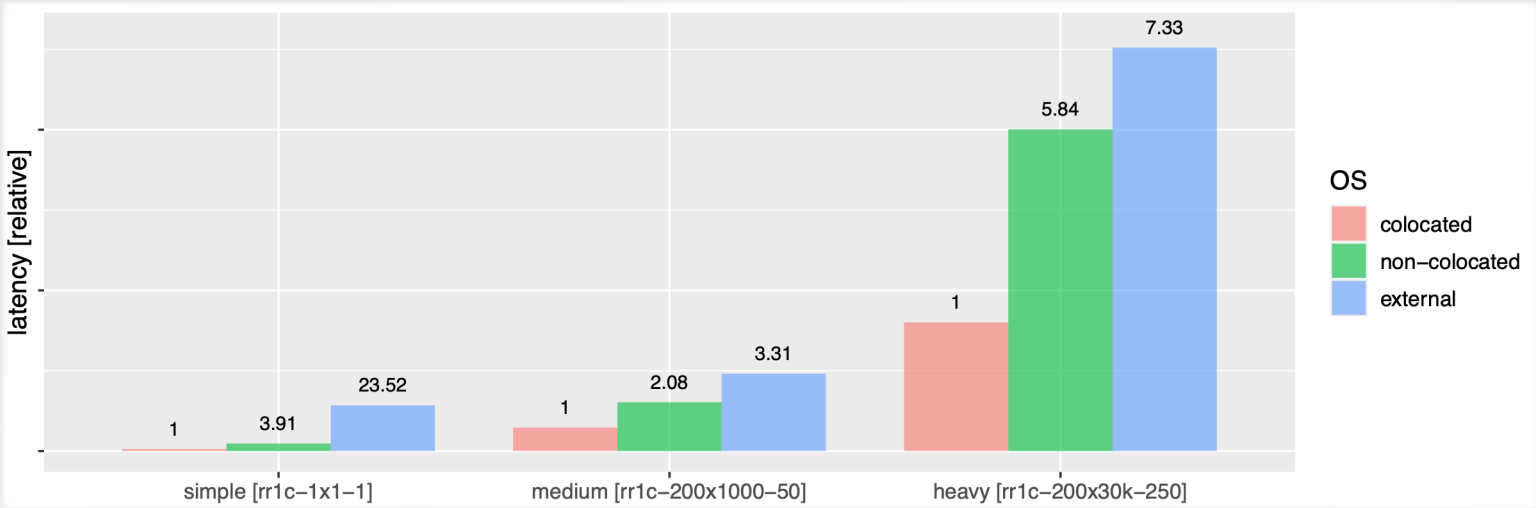
Tunings addressing performance influencing factors (4)



Service locality

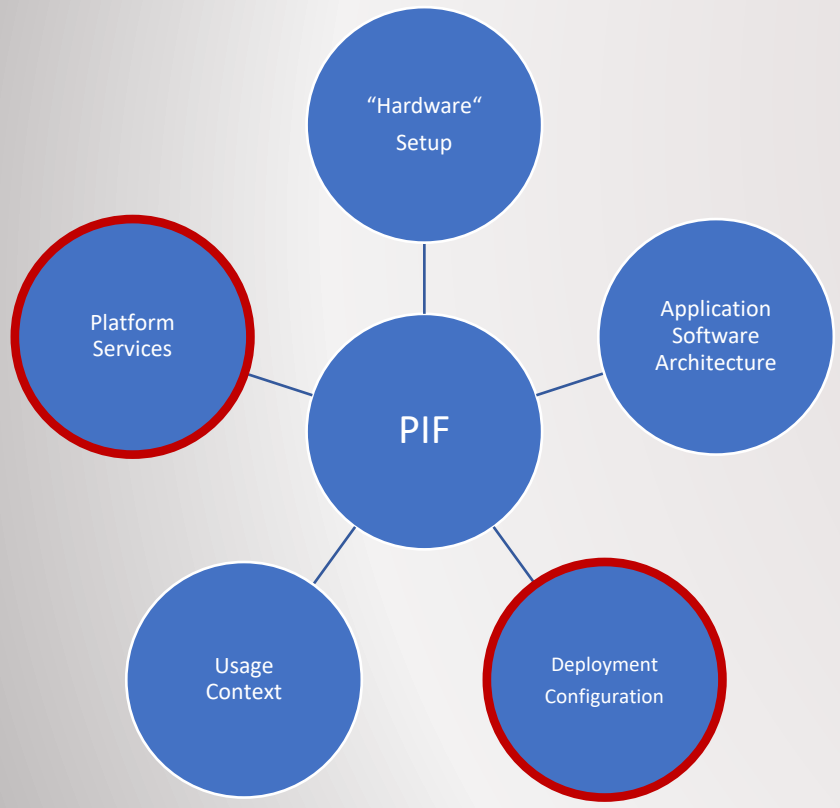
- Assumption: Service-oriented architecture/microservice architecture
- Each deployment strategy comes with different performance

- Longer network distances can impact both, latency and throughput



- Deploy services with frequent interaction locally
- Again trade-off decision required between CPU demand and locality

Tunings addressing performance influencing factors (5)



Nodes and node types

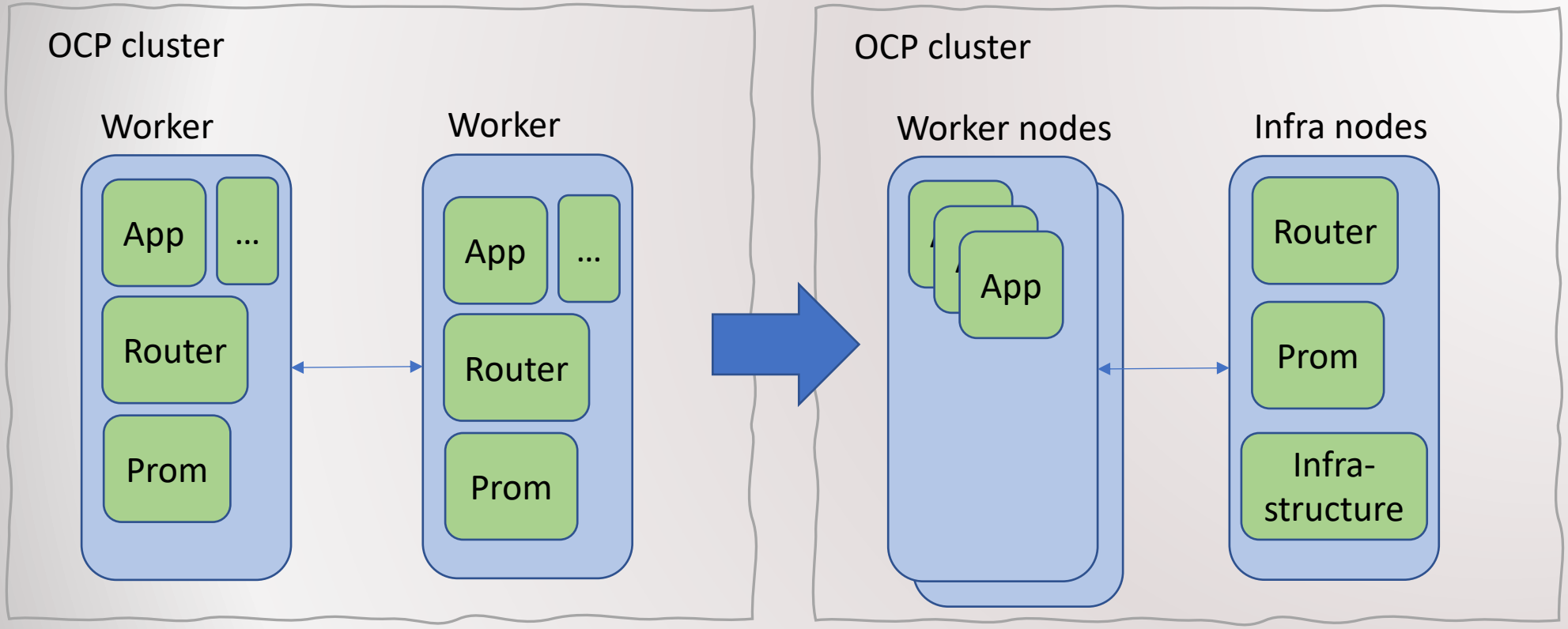
- Infrastructure nodes optional node types in OCP
- Can be used to offload worker nodes from infrastructure pods
  - Cluster monitoring
  - Ingress-router

- Using infra nodes and moving pods can lead to more stable workload performance
- Infrastructure pods consume cycles
- Can influence workload performance noticable
- Per default deployed on workers randomly

- Drawback: Less worker nodes available

How to setup infra nodes:

Tunings addressing performance influencing factors: Infra nodes (1)



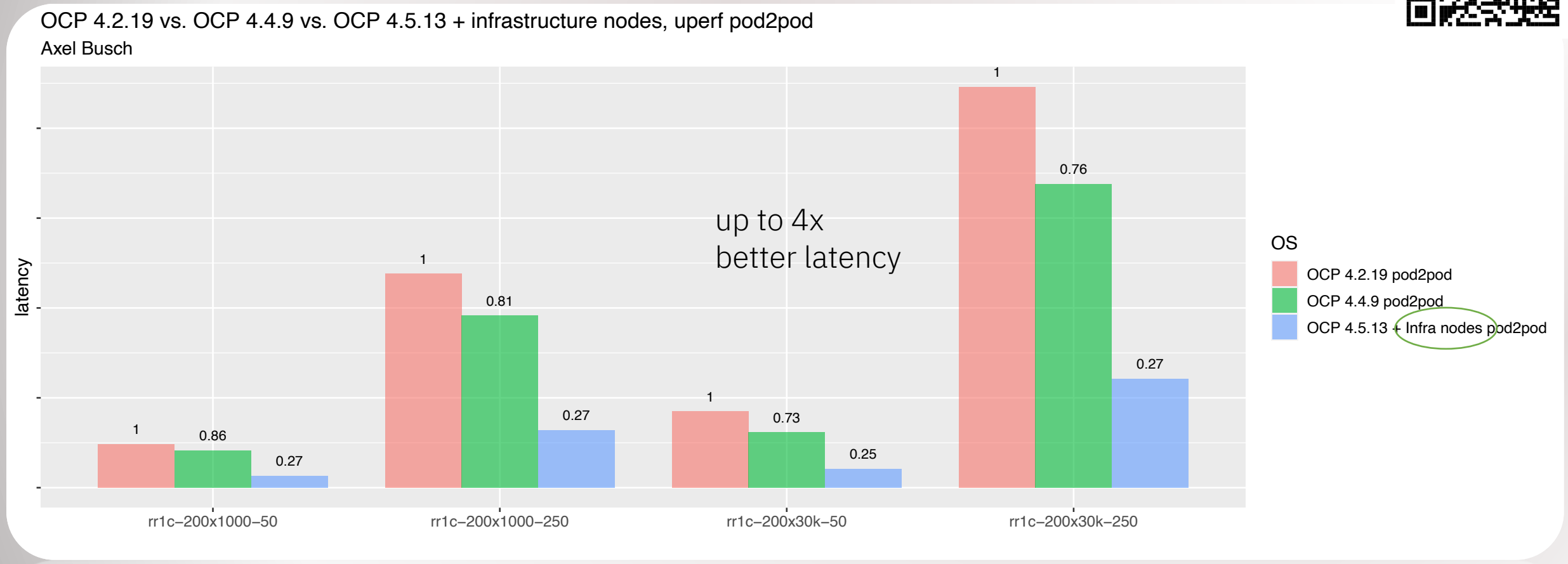
- OCP services, such as monitoring solutions (e.g. Prometheus) and router run on worker nodes per default
- Consume resources of workers and slow down applications

- Move all infrastructure services to infrastructure nodes to keep workers for application workloads exclusively
- Can improve performance significantly

https://www.linkedin.com/pulse/boosting-performance-using-infrastructure-nodes-your-cluster-miranda/

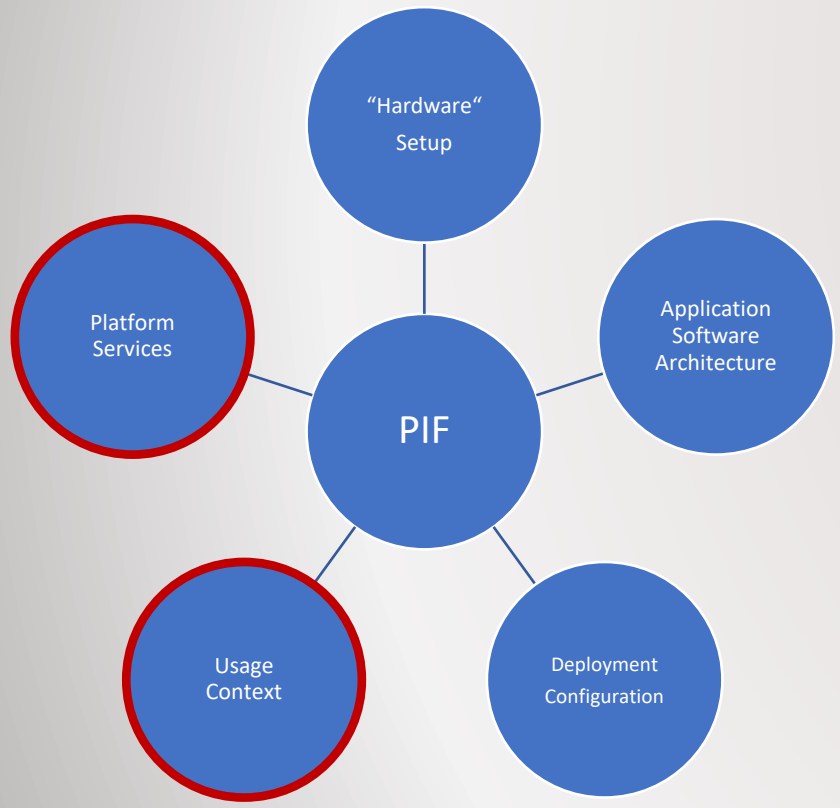
How to setup infra nodes:

Tunings addressing performance influencing factors: Infra nodes (2)



How to setup RFS:

Tunings addressing performance influencing factors (6)



RPS & RFS

- Receive Packet Steering (RPS) & Receive Flow Steering (RFS)
- Prevents hardware queue of network card beeing bottleneck
  - ksoftirq/0
  - ksoftirq/1
  - …
- Directs packets to specific CPUs
- Implemented on software level (kernel)

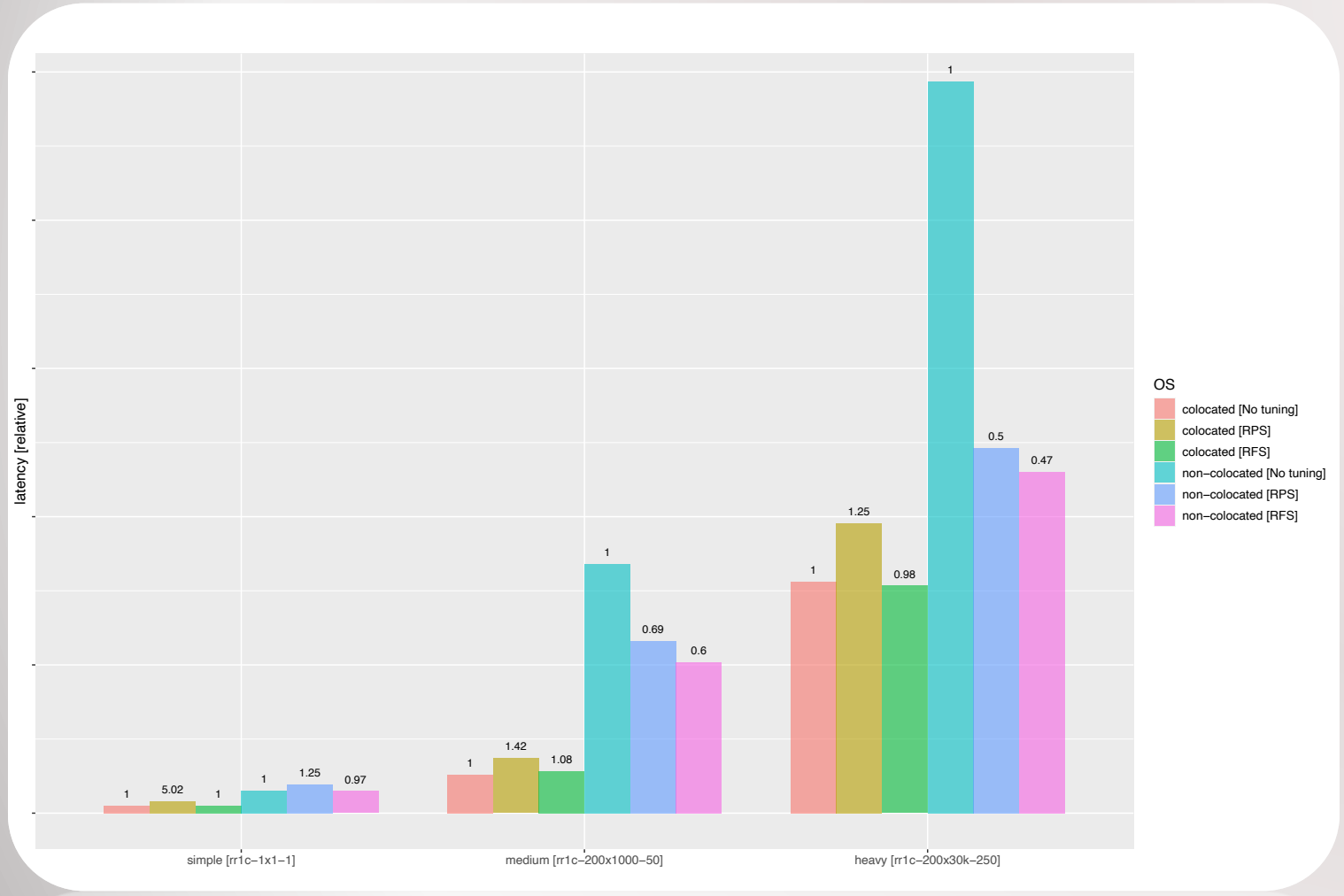
- Can improve latency and throughput
- Especially in high load networking scenarios

- Consumes CPU cycles
- Depends on workload whether to use or not

https://developer.ibm.com/components/ibmz/tutorials/red-hat-openshift-on-ibm-z-tune-your-network-performance-with-rfs/

How to setup RFS:

Tunings addressing performance influencing factors: Results



- RFS seems to outperform RPS (at least in our scenarios)
- RFS beneficial in non-colocated scenarios
- RPS can be harmful in co-located scenarios

RFS can improve performance up to 2x

Summary

Quality

Performance

Trade-off decisions to be made

Performance Influencing Factors

Tunings

RPS & RFS

infra nodes

...

Thank you!

Dr.-Ing. Axel Busch
Technical Lead OCP Performance on Z
axel.busch@ibm.com
Linux on IBM z Performance