IS02 – Virtualisation Options with DPM, z/VM and KVM on IBM z Systems

Arwed Tschoeke
Client Center Böblingen

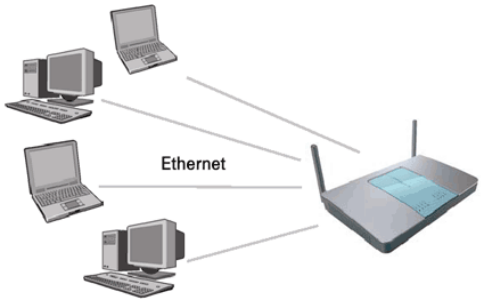
Agenda

- Virtualization basics
- PR/SM and DPM
- z/VM
- KVM
- Docker

Virtualization in your daily life

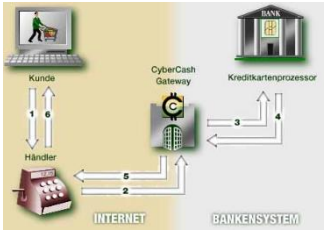
- Ideas!?

Car Sharing



Internet Access Sharing

Bathroom Sharing



Money and virtual Money?

=> Sharing „a few“ real resources with „many“ users

Type 1 Hypervisor

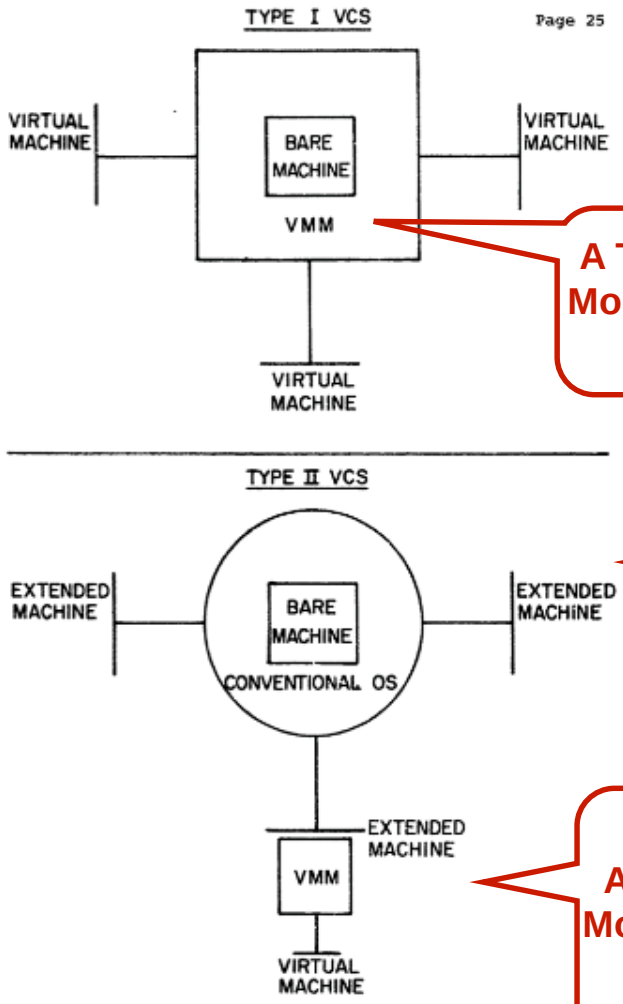


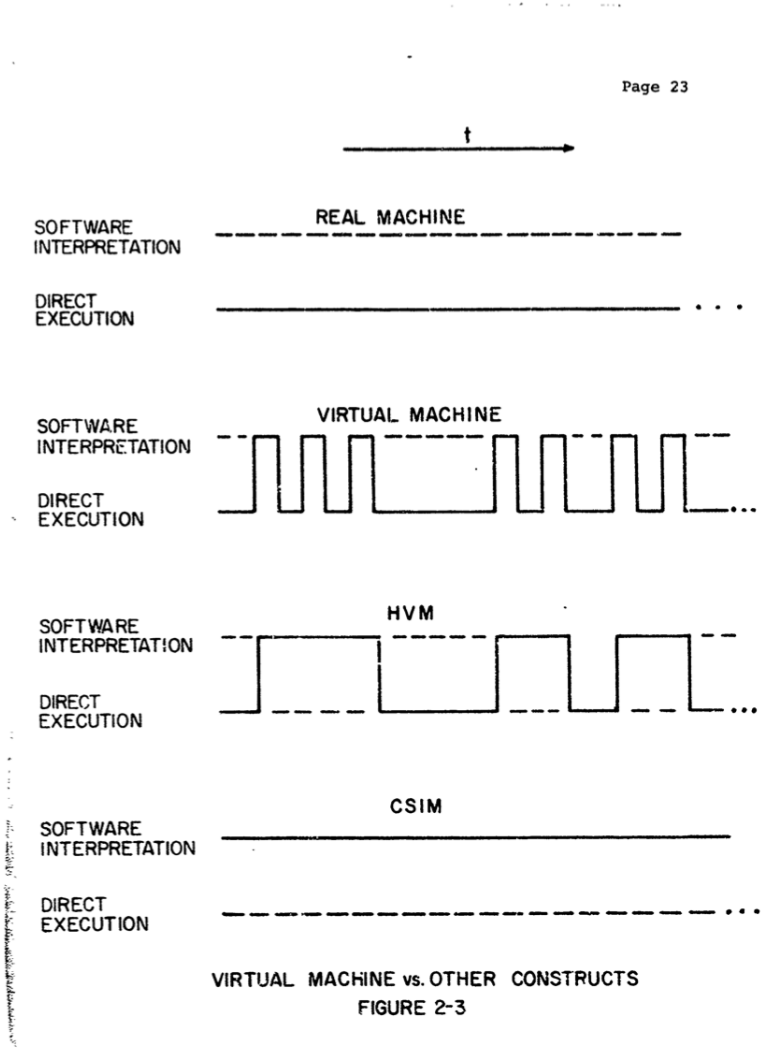
FIGURE 2-4 TYPE I vs TYPE II VCS

A Type 1 Virtual Machine Monitor runs as part of the Kernel

"Extended Machine" is the environment running a time-shared program. The EM evolved into a modern OS Process.

A Type II Virtual Machine Monitor runs as a standard OS Process

Software Interpretation



Non-Virtualized

KVM, VMware, Xen, PowerVM, z/VM

VMware Pre-VMX

Qemu

Hypervisors and Virtualization for z Systems

PR/SM-LPARs

- ***Virtualization capabilities built into the system***
- PR/SM manages and virtualizes all the installed and enabled system resources as a single large SMP system
- ***Full sharing of the installed resources*** with high efficiency and very low overhead
- ***High scalability*** with support for up to 40 (for z13s) or 85 (for z13) logical partitions
- IBM Dynamic Partition Manager ***simplifies management experience***
- Ensured ***workload separation*** based on highest EAL5+ security certification

z/VM v6.4 (preview)

- ***Enables extreme scalability, security and efficiency*** creating cost savings opportunities
- ***Ease Migration*** with upgrade in place infrastructure provides a seamless migration path from previous z/VM releases (z/VM 6.2 and z/VM 6.3) to the latest version
- ***Operational improvements*** by enhancing z/VM to provide ease of use
- ***Improved SCSI support*** for guest attachment of disk and other peripherals, and hypervisor attachment of disk drives

IBM Wave for z/VM

- IBM Wave for z/VM ***simplifies the management*** of virtual Linux servers from a single user interface
- Provides the foundation for cognitive computing on z Systems

KVM on z Systems v1.1.1

- ***Support new analytics workloads*** with Single Instruction Multiple Data (SIMD) for competitive advantage
- ***Deliver higher compute capacity*** with support for Simultaneous Multithreading (SMT) to meet new business requirements
- ***RAS support enhanced for problem determination and high availability*** setup to reduce down time and quickly react to business needs
- ***Secure and protect*** business data with Crypto exploitation

PR/SM and DPM

PR/SM or LPAR Hypervisor

- 'Processor Resource/System Manager' (PR/SM) and 'LPAR hypervisor' are commonly used synonymously.
- However the 'LPAR Hypervisor' is the program itself and 'PR/SM' is the facility of the whole
- So PR/SM aka LPAR hypervisor is a Type-1 Hypervisor that manages logical partitions:
  - Each partition owns a defined amount of physical storage
  - Strictly no storage shared across partitions
  - No virtual storage management / paging done by LPAR hypervisor
  - Zone relocation lets each partition start at address 0
  - CPUs may be dedicated to a partition or may be shared by multiple partitions
  - I/O channels may be dedicated to a partition or may be shared by multiple partitions (Multiple image facility, MIF)
  - Each LPAR has its own architecture mode (ESA/390 or z/Architecture)
- PR/SM is shipped with z Systems (considered as part of the firmware)
- PR/SM was initially introduced in 1988 with the IBM 3090 processors
- Beginning with z990, the PR/SM is always loaded (no Basic Mode anymore)
- Separation of logical partitions is considered as good as having each partition on a separate physical machine (Evaluation Assurance Level 5)

Dynamic Partition Manager (DPM) – At a Glance

Simplification
Provide simplified, consumable, enhanced Partition life-cycle and integrated dynamic I/O management capabilities.

DPM Mode
A CPC can be in non-DPM mode or DPM mode. Enable DPM mode with first IML.

Cloud
Provides the technology foundation that enables IaaS and secure, private Clouds.

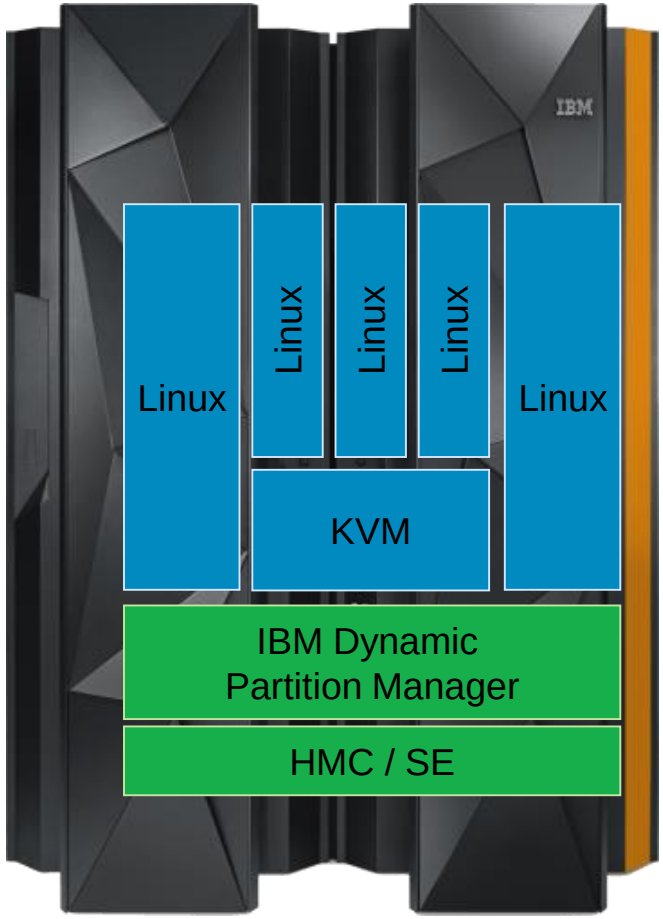
FIE
Initial focus on First In Enterprise (FIE) customers with support for existing clients in a later stage.

Linux only
A CPC running in DPM mode is Linux only. No z/OS, z/VM, zVSE, zTPF support in Stage 1. FCP Storage only.

"DPM provides simplified z Systems hardware and virtual infrastructure management including integrated dynamic I/O management for FIE customers that run KVM on z as a hypervisor and/or Linux on z as a Partition-hosted operating system."

Partition and I/O device management at the HMC

- Linux partitions or KVM partitions correspond to LPARs under standard PR/SM
- Supports only Linux and Linux based hypervisors
- Creation of I/O Configuration Data Set (IOCDS) is “under the covers”
  - Supports dynamic updates of I/O
- Hardware and operating system message displays are unchanged
- Problem determination and maintenance continues to exist on the System Element (SE)
- On/Off Capacity on Demand (OOCoD) and Customer Initiated Upgrade (CIU) supported for Linux



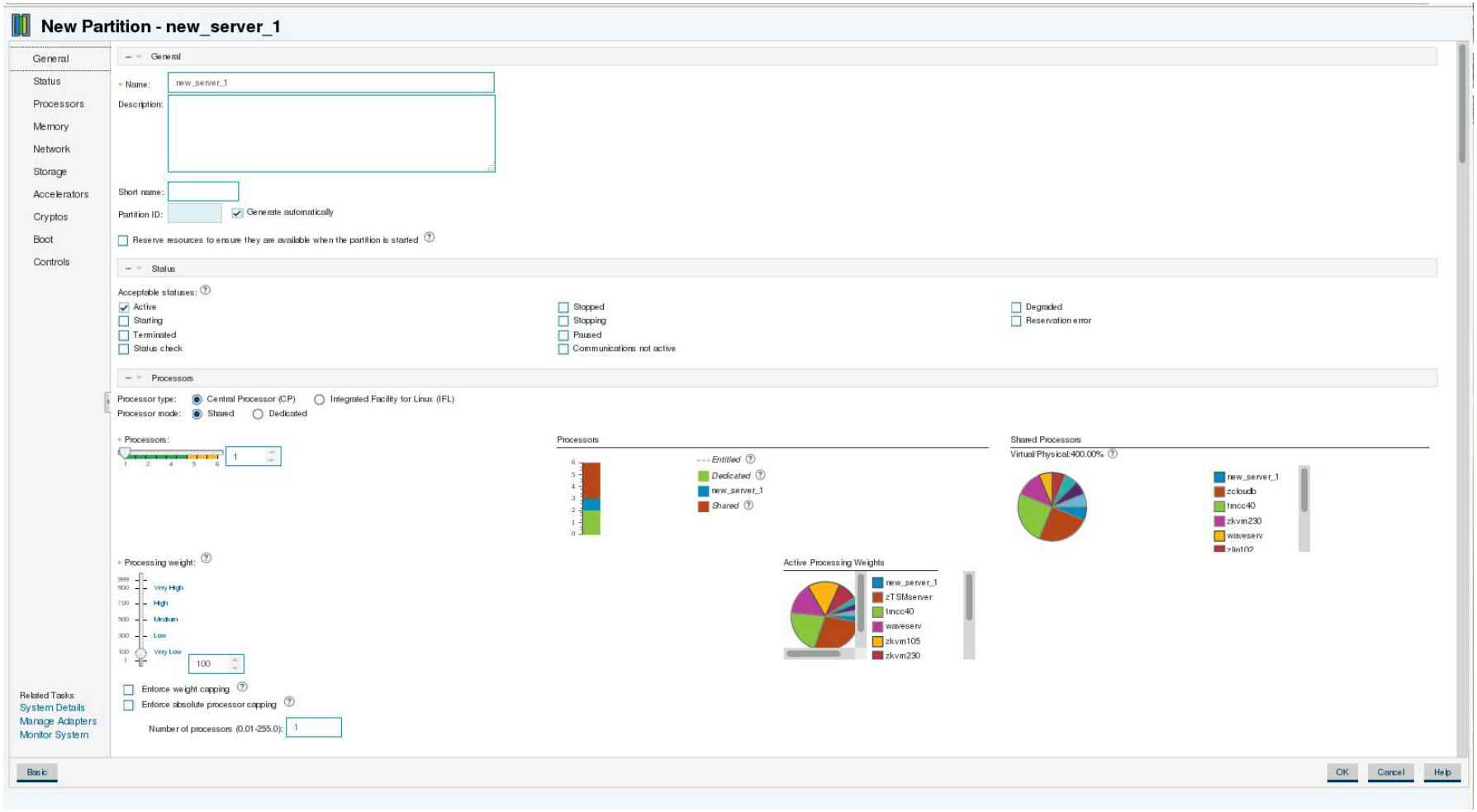
How DPM helps in a new Linux Environment

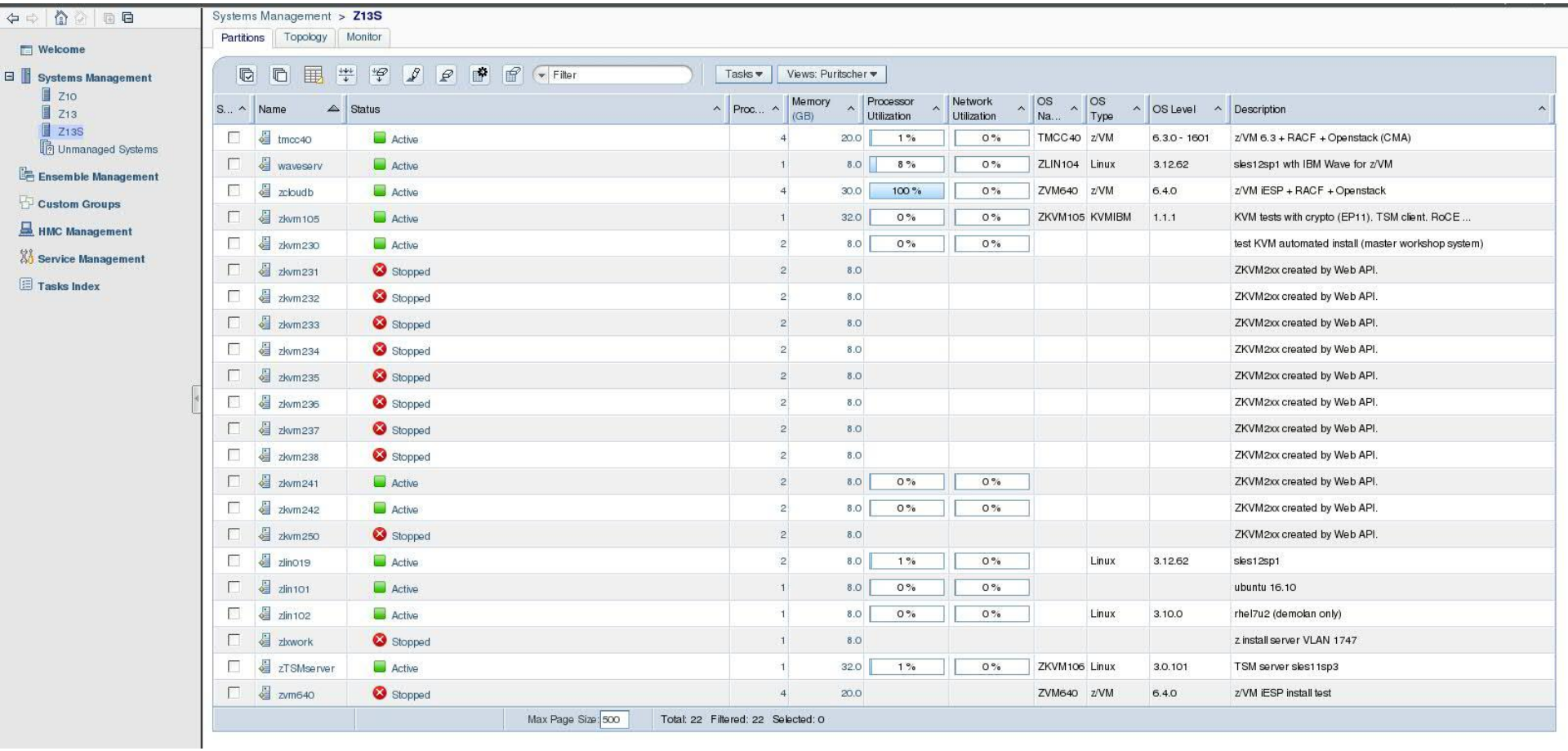
- z Systems and PR/SM require a HW definition
- Dynamic IO – one of the key differentiator of the platform would be nice
- Having the option to have a GUI-based administration
- Overcome the prejudice: z is old school and complicated
- Everything is scripted (that‘s why we need GUIs ;-))

```
*ICP  ICP070I SEARCH FOR '*ICP' TO FIND EACH IOCP MESSAGE
        ID    MSG1='IODF00',MSG2='SYS1.IODF00 - 2014-07-22 16:19',    *
              SYSTEM=(2828,1),LSYSTEM=P008B857,                       *
        RESOURCE PARTITION=((CSS(0),(ZOS1,2),(ZVM1,1),(*,3),(*,4),(*,5*
              ),(*,6),(*,7),(*,8),(*,9),(*,A),(*,B),(*,C),(*,D),(*,E),*
              (*,F)),(CSS(1),(*,1),(*,2),(*,3),(*,4),(*,5),(*,6),(*,7)*
              ,(*,8),(*,9),(*,A),(*,B),(*,C),(*,D),(*,E),(*,F)))
        CHPID PATH=(CSS(0),25),SHARED,PARTITION=((ZOS1,ZVM1),(=)),    *
              PCHID=160,TYPE=FC
        CHPID PATH=(CSS(0),26),SHARED,PARTITION=((ZOS1,ZVM1),(=)),    *
              PCHID=11C,TYPE=FC
        CHPID PATH=(CSS(0),27),SHARED,PARTITION=((ZOS1,ZVM1),(=)),    *
              PCHID=161,TYPE=FC
        CHPID PATH=(CSS(0),28),SHARED,PARTITION=((ZOS1,ZVM1),(=)),    *
              PCHID=11D,TYPE=FC
        CHPID PATH=(CSS(0),F8),SHARED,PARTITION=((ZOS1,ZVM1),(=)),    *
              PCHID=17C,TYPE=OSD
        CHPID PATH=(CSS(0),F9),SHARED,PARTITION=((ZOS1,ZVM1),(=)),    *
              PCHID=104,TYPE=OSD
        CNTLUNIT CUNUMBR=0030,PATH=((CSS(0),F8)),UNIT=OSA
        IODEVICE ADDRESS=(030,064),UNITADD=00,CUNUMBR=(0030),UNIT=OSA
        CNTLUNIT CUNUMBR=0070,PATH=((CSS(0),F9)),UNIT=OSA
        IODEVICE ADDRESS=(070,064),UNITADD=00,CUNUMBR=(0070),UNIT=OSA
        CNTLUNIT CUNUMBR=0300,PATH=((CSS(0),25,26)),                  *
              UNITADD=((00,032)),CUADD=0,UNIT=2105
        IODEVICE ADDRESS=(300,032),CUNUMBR=(0300),STADET=Y,UNIT=3390
        CNTLUNIT CUNUMBR=0320,PATH=((CSS(0),25,26)),                  *
              UNITADD=((00,032)),CUADD=2,UNIT=2105
        IODEVICE ADDRESS=(320,032),UNITADD=00,CUNUMBR=(0320),STADET=Y,*
              UNIT=3390
        CNTLUNIT CUNUMBR=0340,PATH=((CSS(0),25,26)),                  *
              UNITADD=((00,032)),CUADD=4,UNIT=2105
        IODEVICE ADDRESS=(340,032),UNITADD=00,CUNUMBR=(0340),STADET=Y,*
              UNIT=3390
        CNTLUNIT CUNUMBR=0360,PATH=((CSS(0),25,26)),                  *
              UNITADD=((00,032)),CUADD=6,UNIT=2105
        IODEVICE ADDRESS=(360,032),UNITADD=00,CUNUMBR=(0360),STADET=Y,*
              UNIT=3390
        CNTLUNIT CUNUMBR=0380,PATH=((CSS(0),25,26)),                  *
              UNITADD=((00,032)),CUADD=8,UNIT=2105
        IODEVICE ADDRESS=(380,032),UNITADD=00,CUNUMBR=(0380),STADET=Y,*
              UNIT=3390
        CNTLUNIT CUNUMBR=0400,PATH=((CSS(0),25,26)),                  *
              UNITADD=((00,032)),CUADD=A,UNIT=2105
        IODEVICE ADDRESS=(400,032),CUNUMBR=(0400),STADET=Y,UNIT=3390
```

short version: No Texteditor required to get started, dynamic IO available

How DPM looks like





Partition Details - zTSMserver

- General
- Status
- Controls
- Processors
- Memory
- Network
- Storage
- Accelerators
- Cryptos
- Boot

Related Tasks
Stop
System Details
Manage Adapters
Monitor System

Storage

HBAs

Actions | Search

Name	WWPN	Type	Adapter Name	Device Number	Card Type	Description
sana	C05076D7D2000019	FCP	SAN-A	7000	FICON Express16s	
sanb	C05076D7D200001A	FCP	SAN-B	7100	FICON Express16s	

Total: 2 Selected: 0

Accelerators

Accelerator Virtual Functions

Partition Details - zTSMserver

- General
- Status
- Controls
- Processors
- Memory
- Network
- Storage
- Accelerators
- Cryptos
- Boot

Related Tasks
- Stop
- System Details
- Manage Adapters
- Monitor System

Network

NICs

Actions | Search

Name	Type	Adapter Name	Adapter Port	Device Number	Card Type	Description
demolan	OSA	DemoLAN	0	EA00	OSA-Express5s 1000Base-T	
intranet	OSA	Intranet	0	EC00	OSA-Express5s 1000Base-T	

Total: 2 Selected: 0

Storage

HBAs

OK | Apply | Cancel | Help

Partition Details - zkvm230

- General
- Status
- Controls
- Processors
- Memory
- Network
- Storage
- Accelerators
- Cryptos
- Boot

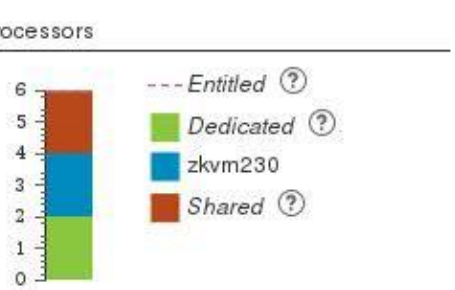
Related Tasks
- Stop
- System Details
- Manage Adapters
- Monitor System

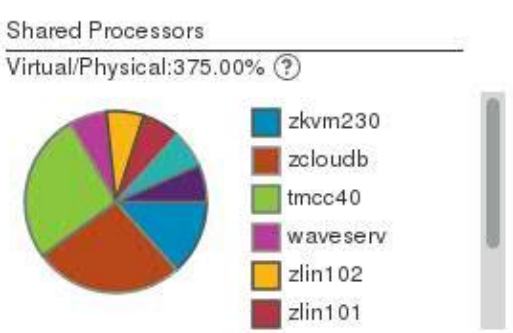
Processors

Processor type: (•) Central Processor (CP) ( ) Integrated Facility for Linux (IFL)

Processor mode: (•) Shared ( ) Dedicated

* Processors: 1 2 4 5 6 — 2

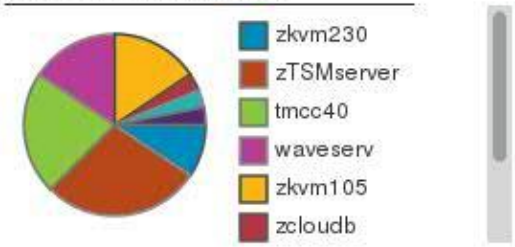




* Processing weight:

999 / 900 Very High
700 High
500 Medium
300 Low
100 / 1 Very Low

300



[ ] Enforce weight capping

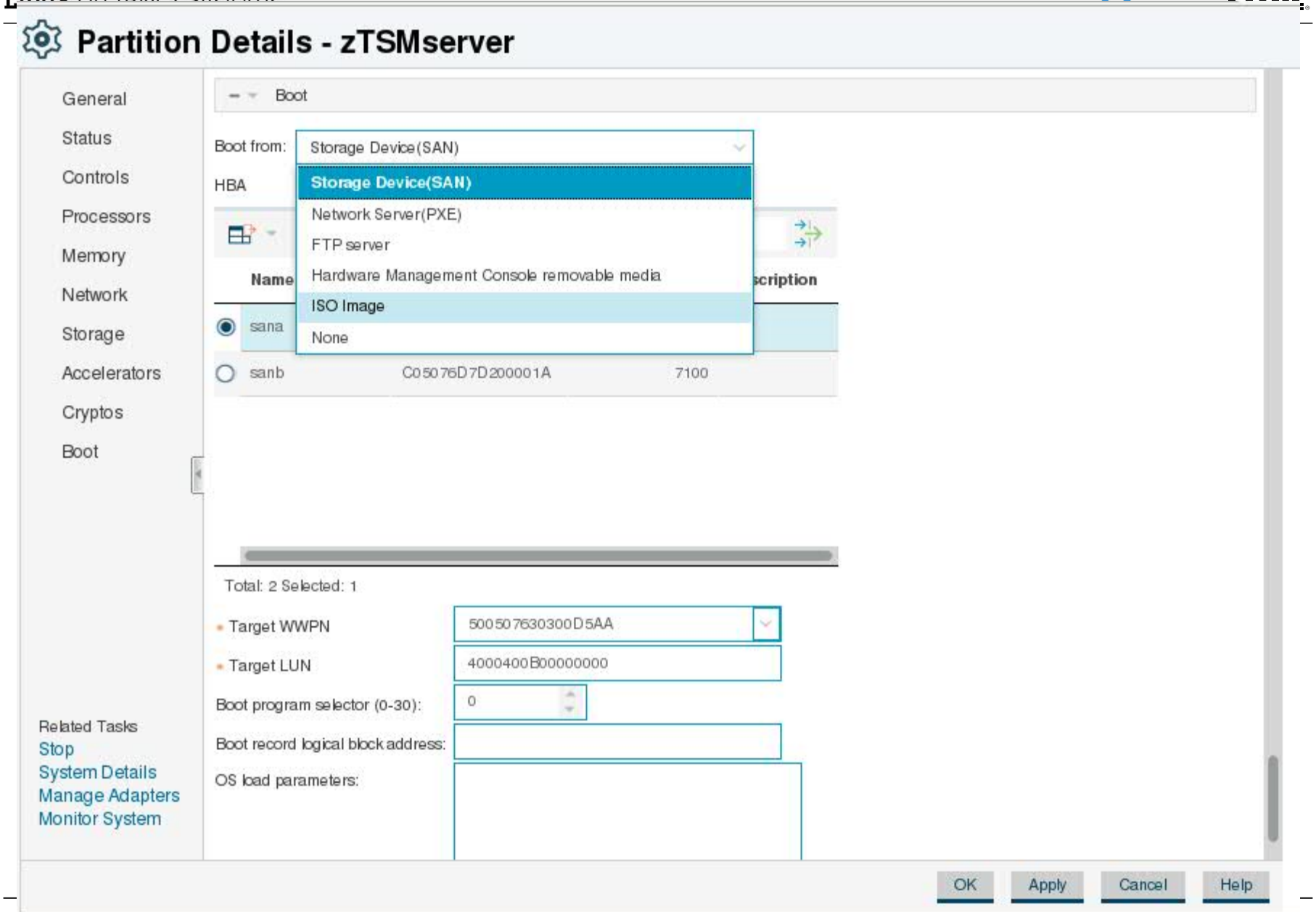
[ ] Enforce absolute processor capping

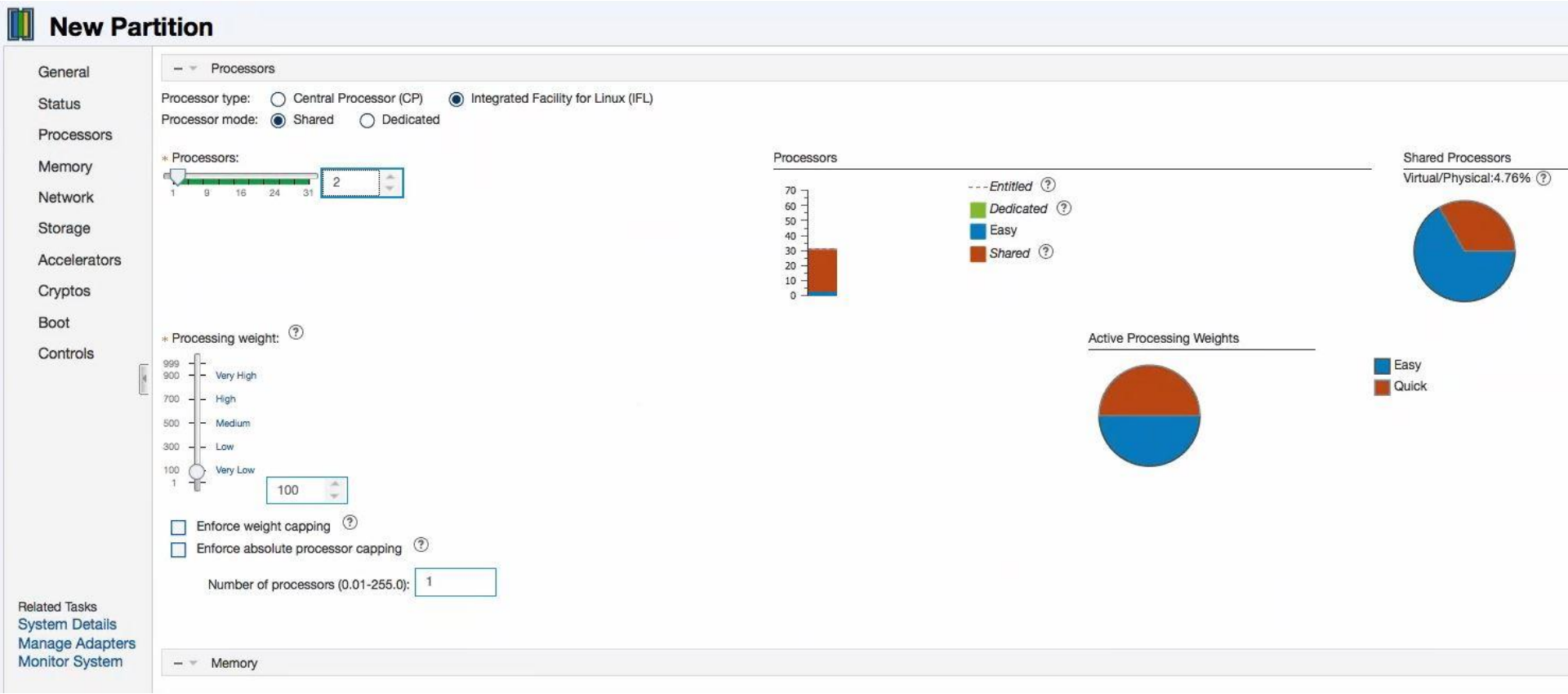
Number of processors (0.01-255.0): 1

Manage Processor Sharing

Memory

OK | Apply | Cancel | Help





z/VM

IBM z/VM Hypervisor

- z/VM is the product name of a Type-1 Hypervisor
- z/VM
  - virtualizes the architecture:
    - Guests definitions are completely virtual (and do not necessarily be consistent with physical HW)
  - support DASD and FCP
    - Offers the possibility to choose the solution with the largest convenience factor
  - SSI Clustering for increased avaialbility
  - Integration into GDPS
- Since z990 (with the removal of the Basic Mode), z/VM always runs either in an LPAR or nested on another z/VM systems

z/VM Version 6 Release 3
Making Room to Grow Your Business

Product General Availability

z/VM support for zEDC Express and 10GbE RoCE Express features Available, CPU Pooling

January 14

z13 and z/VM Enhancements Announcement

Feb 13

Base z13 & Crypto support Available

March 13

SMT and Scalability Support Available

June 26

Multi-VSwitch Link Aggregation Available

September 15

RACf enhancements,

Prorated Core time

January 14

dynamic PDR migration,

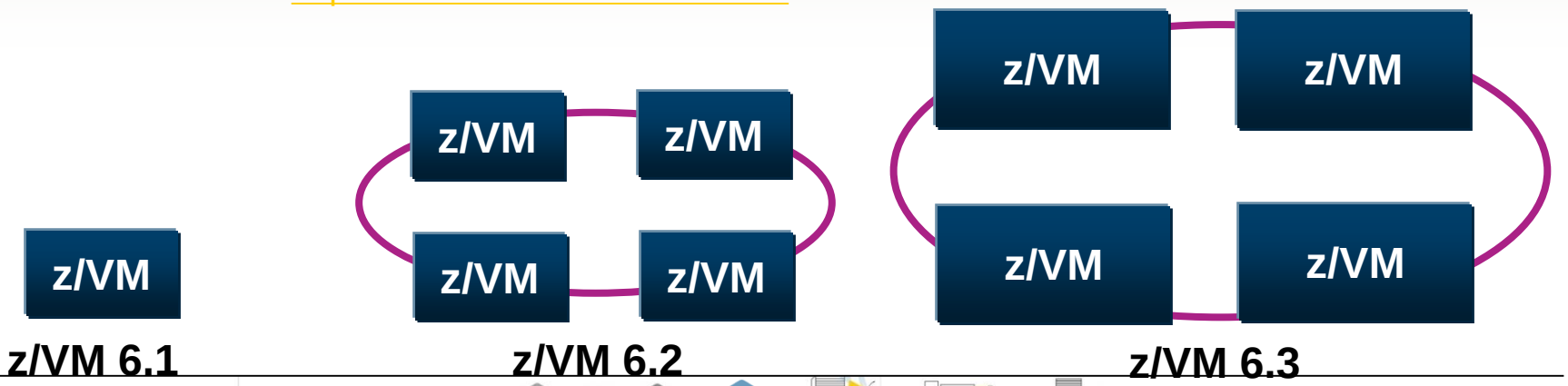
SIMD support

Q4/14

z/VM6.4 GA

2013/14 | 2015 | 2016

See http://www.vm.**ibm.com**/zvm630/



System Programmer & Management Capability

- Upgrade In Place migration enhancements
  - Upgrade In Place migration was introduced in z/VM 6.3
  - Enhanced to allow migration to z/VM 6.4 from
    - z/VM 6.2 or z/VM 6.3 (but not both at same time in cluster)
    - Supports migration for clustered or non-clustered systems

KVM

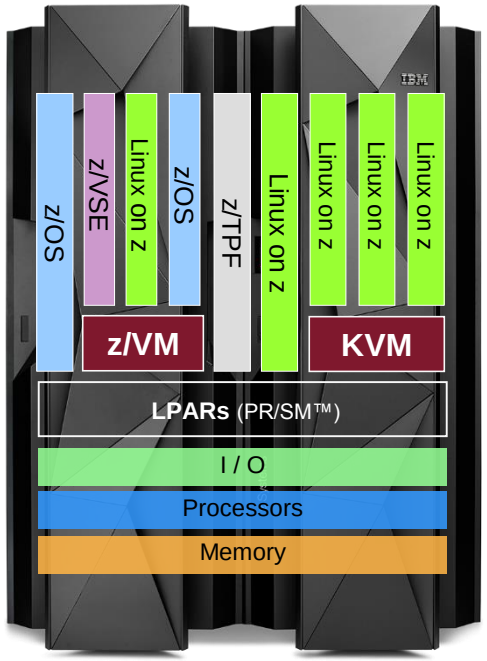
KVM Overview

- KVM (Kernel Virtual Machine) is a Linux kernel-based hypervisor
- Developed and maintained by Avi Kivity / Qumranet, recently acquired by Red Hat
- KVM turns the Kernel into a hypervisor by loading a kernel module and opening a device node. The main parts of KVM are:
  - Kernel module kvm.ko
  - Hardware specific modules
  - Device node /dev/kvm (to create/run VMs from userspace with a set of ioctl()s)
- Virtual machines (or guests or domains) appear as normal Linux processes and integrate seamlessly into the rest of Linux
- A VM has its own memory, that is separated from the user space process
- Virtual CPUs are not scheduled on it's own (vCPUs are realized as Linux threads, and are still scheduled by the Linux Kernel process scheduler)
- In full virtualization mode it's possible to run multiple unmodified guest OSes in parallel, with each having private virtual hardware (network, disk, graphics etc.)
- Exploits 'SIE' hardware instruction on z Systems

KVM for z Systems

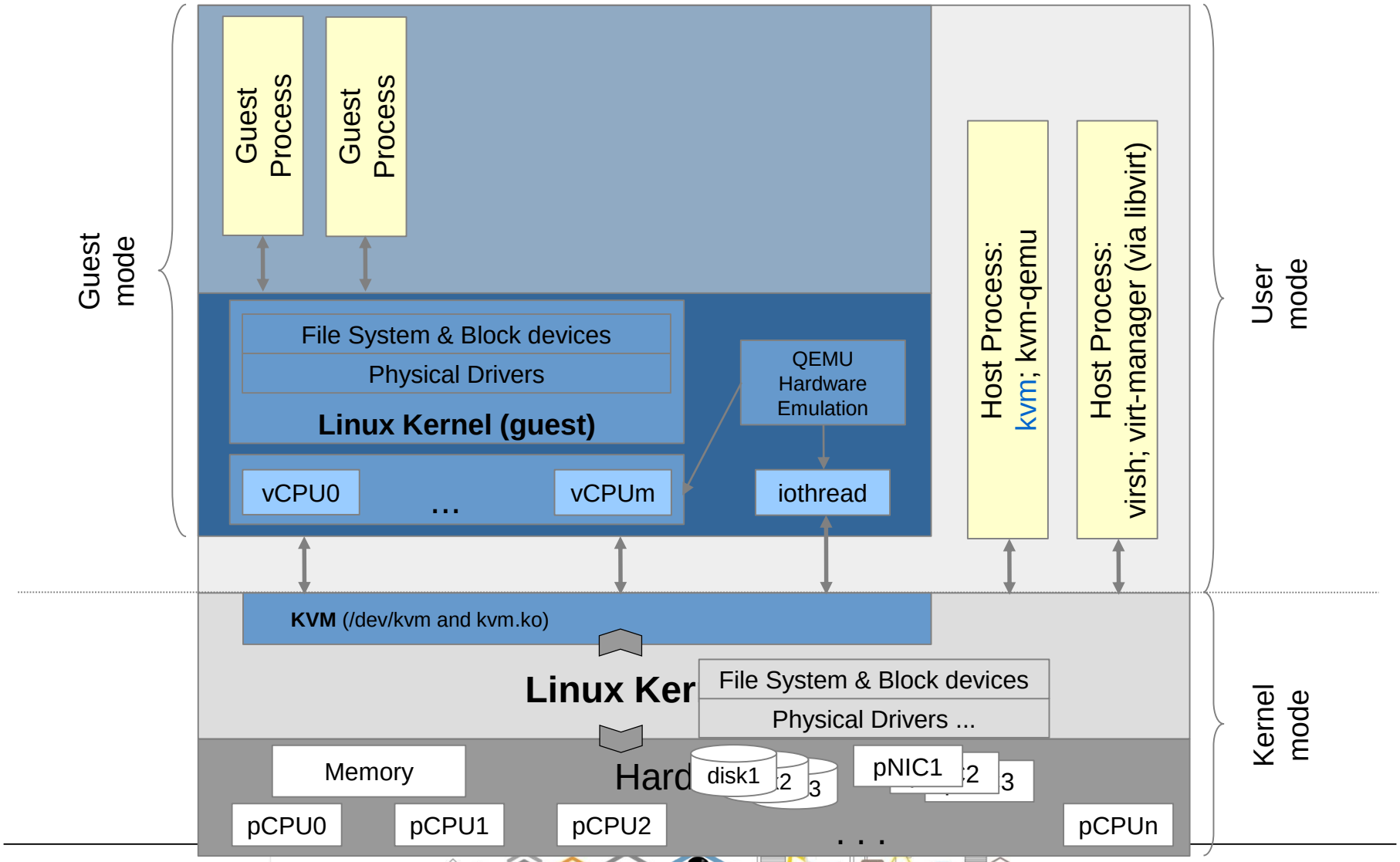
In addition to z/VM, IBM supports a Kernel-based Virtual Machine (KVM) offering for z Systems that hosts Linux on z Systems guest virtual machines.

- The KVM can be installed on z Systems processors.
- The KVM offering co-exists with z/VM virtualization environments, z/OS, Linux on z Systems, z/VSE and z/TPF.
- The KVM offering is optimized for the z Systems architecture and provides standard Linux and KVM interfaces for operational control of the environment.
  - Enterprises will be enabled to easily integrate Linux servers into their existing infrastructure and cloud offerings.



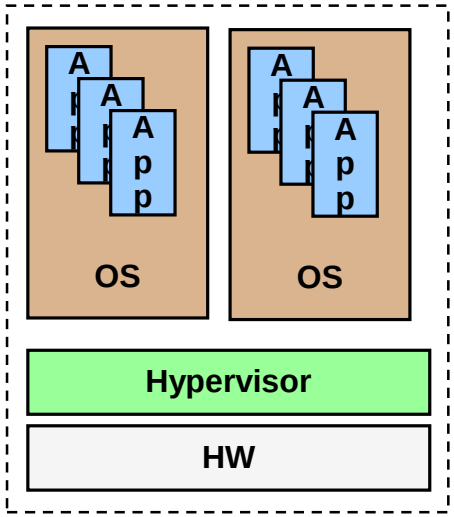
* All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. Any reliance on these Statements of General Direction is at the relying party's sole risk and will not create liability or obligation for IBM.

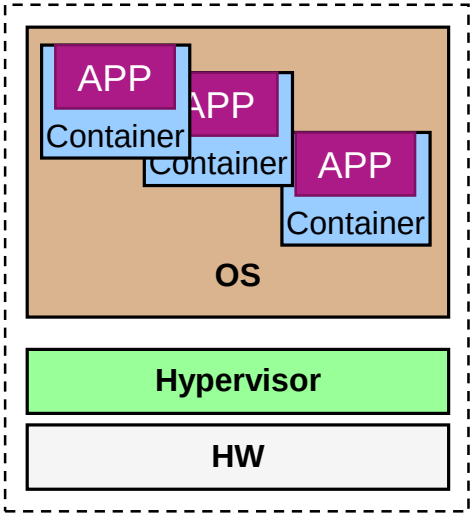
Qemu/KVM Component Diagram



Others

Virtualization vs. Containers



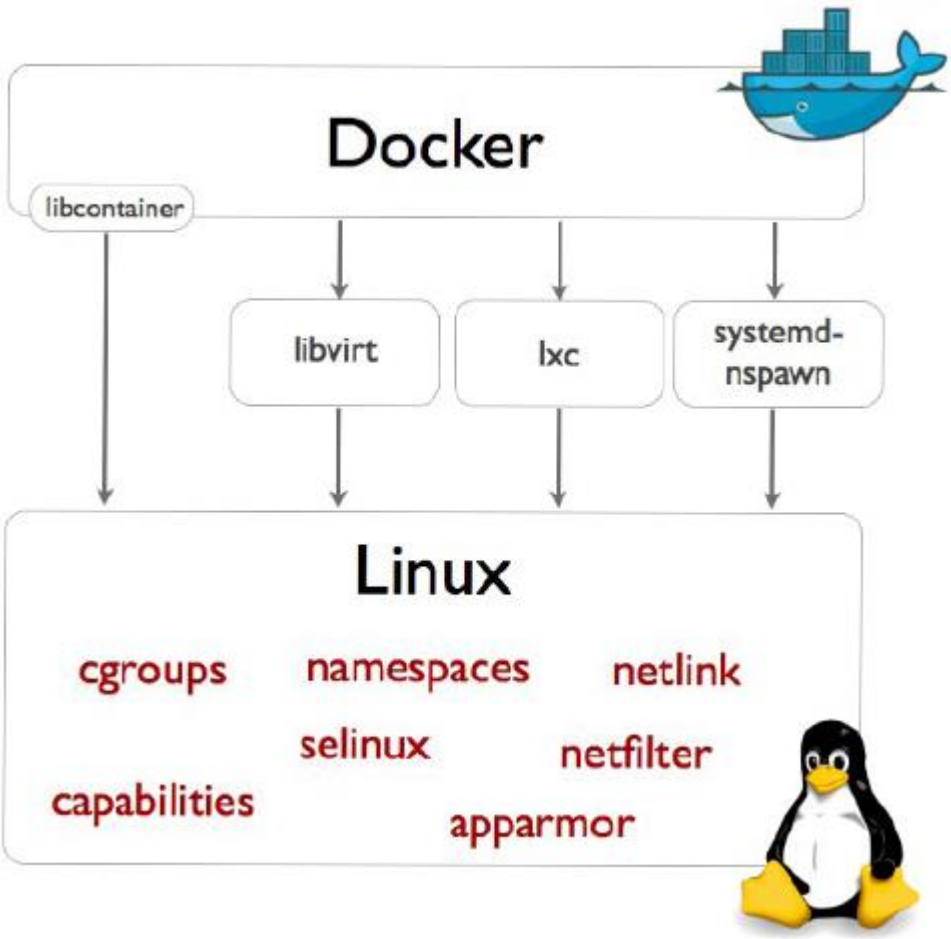


- Infrastructure oriented:
- coming from servers, now virtualized
- several applications per server
- isolation
- Separation between tenants

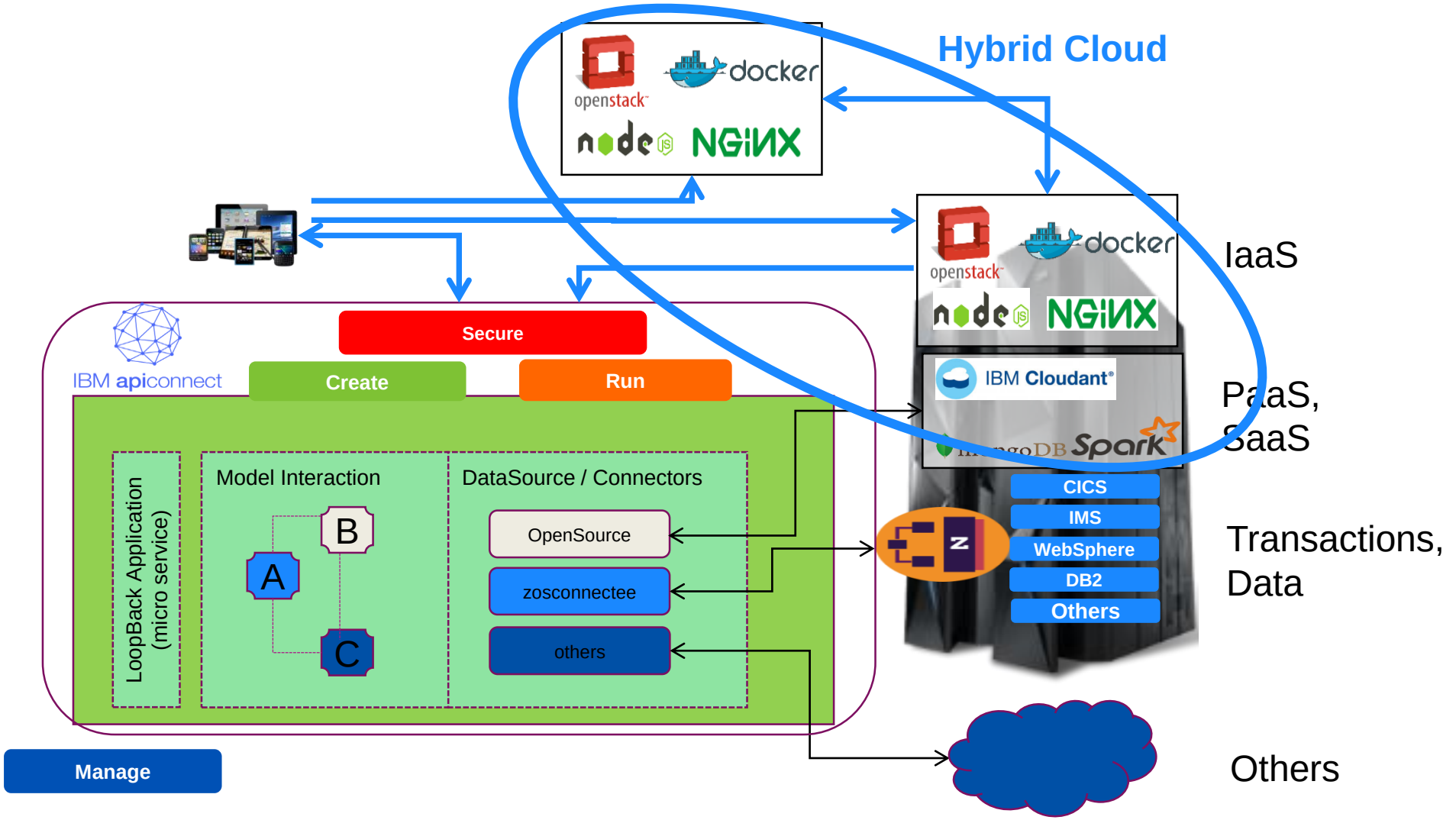
- Service oriented:
- application-centric
- solution decomposed
- DevOps
- separations between the apps of a tenant

Docker and Containers

- Docker in general offers different ways to access the Linux Kernel and software resources that allows to constitute and form Containers:
  - libvirt
  - systemd-nspawn
  - lxc and
  - libcontainer
- The recently introduced 'libcontainer' library – Dockers own way to access these resources, like namespaces and cgroups – seems to prevail and to become accepted.



z Systems: connecting data and transaction to Hybrid clouds



No architectural limits for integration with other platforms

Platform as-a-Service

- **OpenShift** — Red Hat
- **Cloud Foundry** — SUSE, Ubuntu
- **BlueMix (Public)** (Based on Cloud Foundry) — IBM

Workload Orchestration

- **VMware vRealize Automation**
- **IBM Cloud Orchestrator**

Deployment Management

- **Chef**
- **Ansible**
- **Puppet**
- **SaltStack**
- **Juju**
- **Urban Code Deploy**

Infrastructure as-a-Service

- **OpenStack**
  - **Nova**
  - **Neutron**
  - **Cinder**
  - **Trove**

Container Management

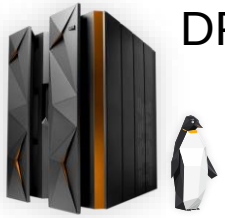
- **Docker**
- **Kubernetes**
- **Mesos**
- **LXC LXD**

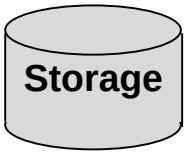
Virtual Infrastructure

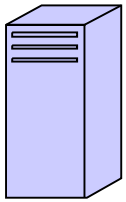
- **SLES**
- **Ubuntu**
- **RHEL**
- **z/VM**
- **KVM for IBM z**

Physical Infrastructure


PR/SM


DPM

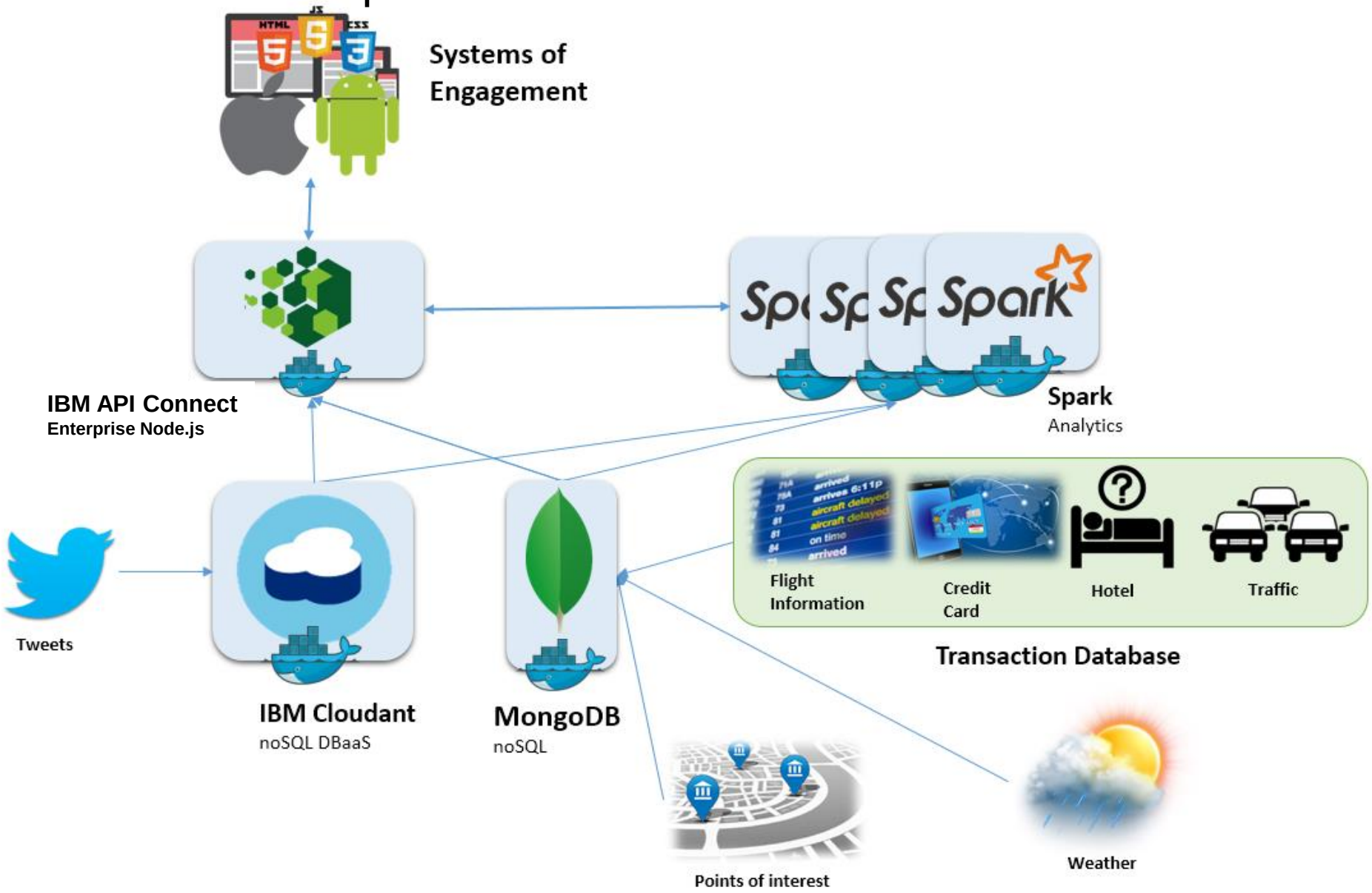


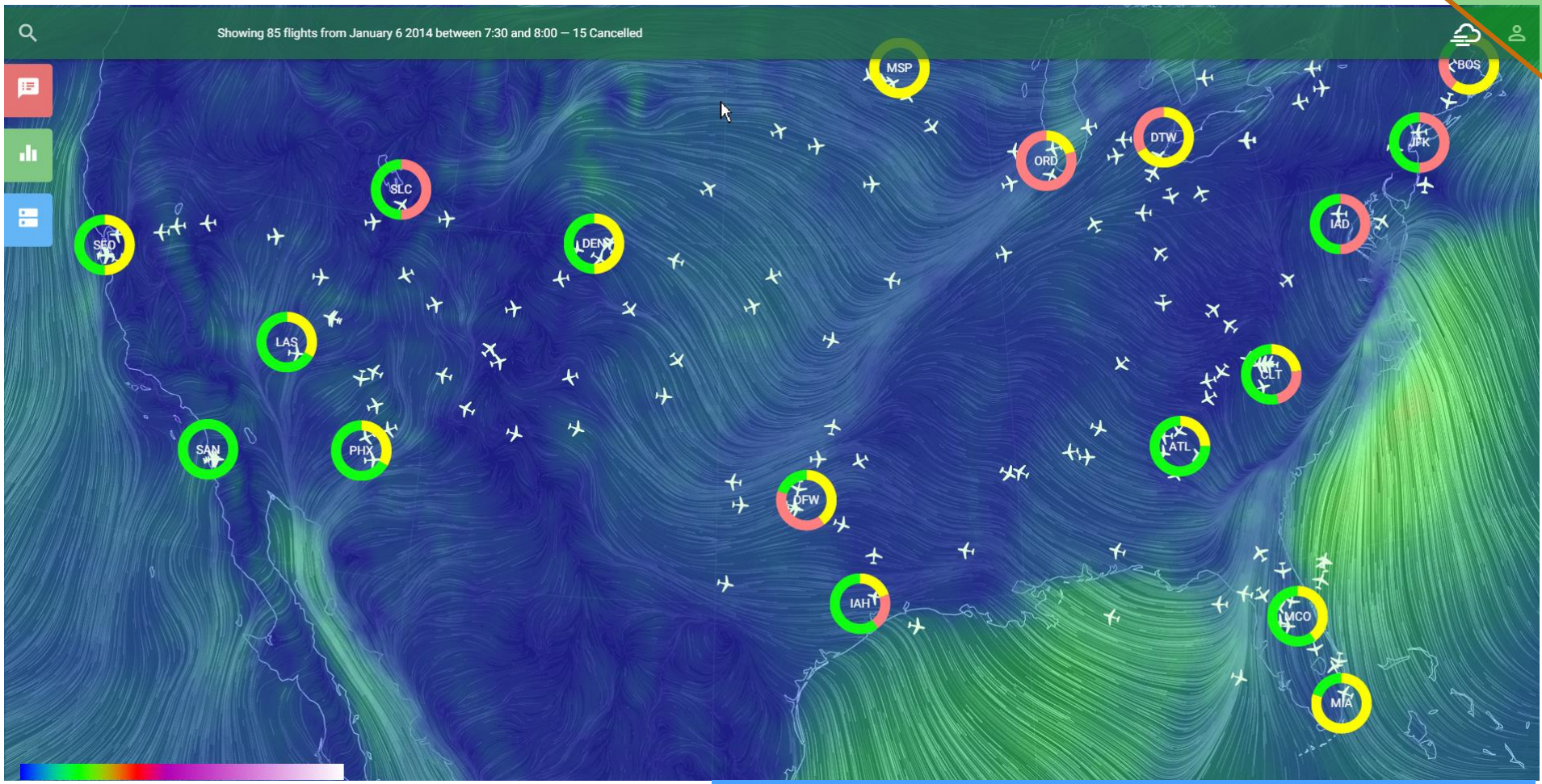

Switches

Legend:
Delivered by IBM

Summary

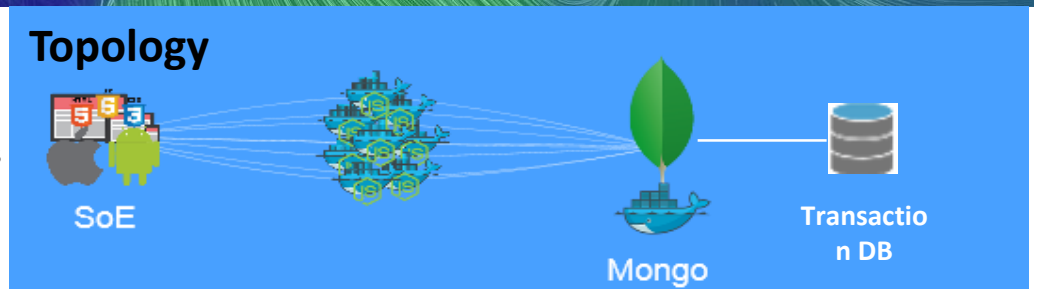
Architecture of a µservice Based Solution





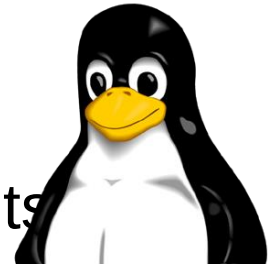
Why LinuxONE

- In 2014, there were nearly 850 Million US Airline passengers or ~ 2.3 millions passengers per day.
- On a bad travel day, an average user could generate ~20 page loads with each page load generating ~100 web events.
- This drives a server volume of ~ 4.6 billion web events per day!



Things to Remember

- HW-accelerated virtualization
  - integrates into the existing world
  - efficient, secure, scalable
  - large benefits in current environemnts (consolidation, efficiency) and future (µ services, highly scalable, on requirement)
- Choices:
  - take what ever is suited best
    - What is standard in your IT
    - What provides the best match to the requirements
    - Or both
  - Remember: OR, not XOR



Arwed Tschoeke

IBM Client Center –
Systems and Software –
z ATS
IBM Germany Lab

Schoenaicher Str. 220
D-71032 Boeblingen

Phone +49 (0) 171 863 7780

arwed.tschoeke@de.ibm.com