Red Hat Summit

Connect

Die Storage-Frage für OpenShift Virtualization

- alles was Sie wissen müssen -

Matthias Münch

Principal Specialist Solution Architect
Red Hat

Agenda

- Overview
- Solution approaches
- Dependencies (how to get and access storage)
- Evaluation graph
- Scoring
- Future fitness

Storage for OpenShift Virtualization

All types of storage providing block or file is suitable

- you can mix & match -
- you can reuse existing more "traditional" storage systems -
- you can use SDS -
- fast growing ecosystem from multiple storage solution vendors -

Check functionality

- some basic requirements must be met -
- scalability must match the environment -

Check costs

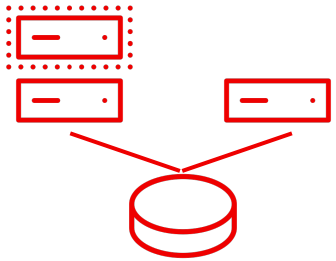
- timeline of a solution -
- stacking elements -

Check complexity

- multiple layers -
- reuse of storage solutions vs expectations -
- learning curve -

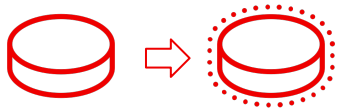
Storage layer functionality recommended

CSI plugin features



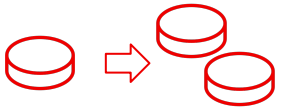
Access attribute:
Read-Write-Many

VM live migration &
VM restart



Volume snapshots

CSI:
VM cloning; VM backup



Volume cloning

VM cloning & VM templates
(aka golden images)

Availability

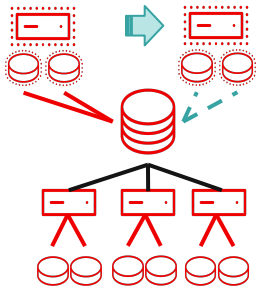
Data redundancy across
failure domains

Container Storage Interface (CSI)

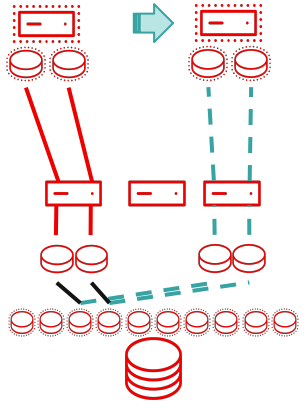
Solution Approaches

Different ways of providing storage to VMs

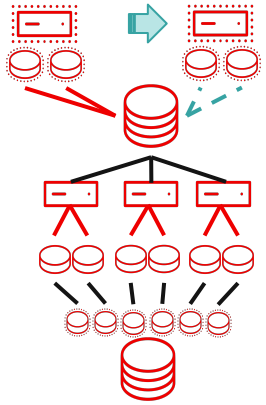
Full SDS



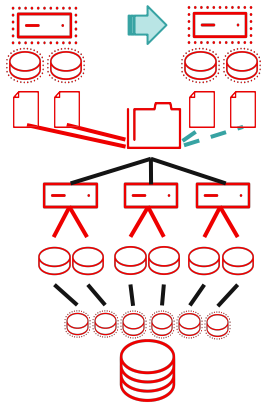
Traditional Storage with CSI



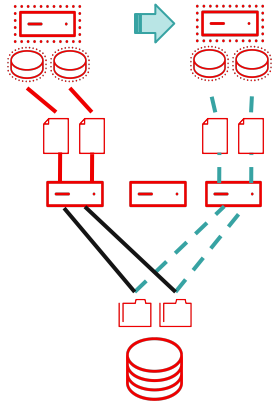
Traditional Storage w/o CSI



Traditional storage with Virtual File System



Shared File System



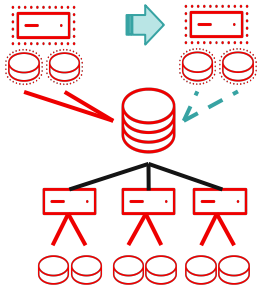
Native block storage

Abstraction layers

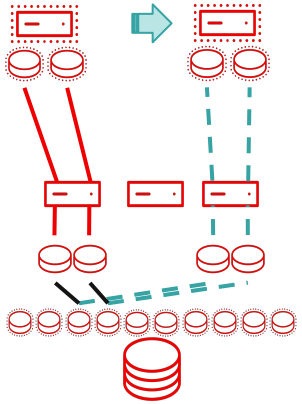
Native File storage

How to get storage?

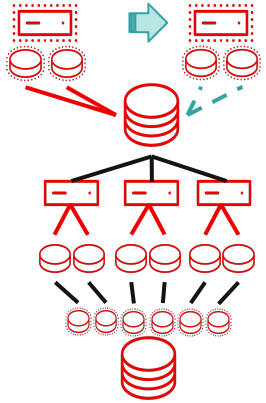
Full SDS



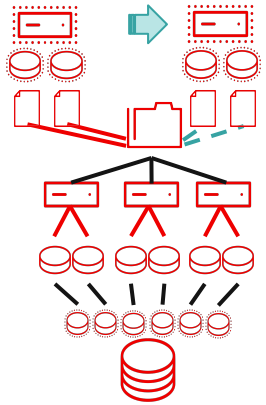
Traditional Storage with CSI



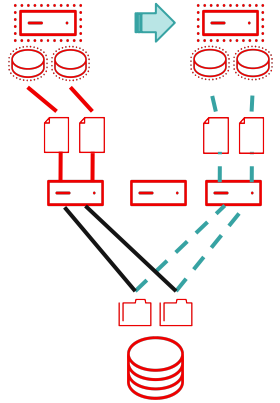
Traditional Storage w/o CSI



Traditional storage with Virtual File System



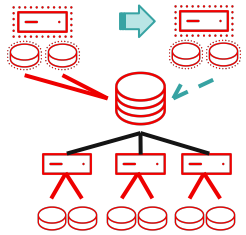
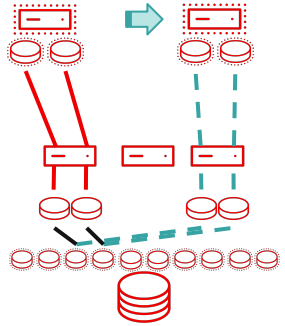
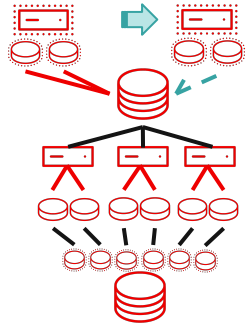
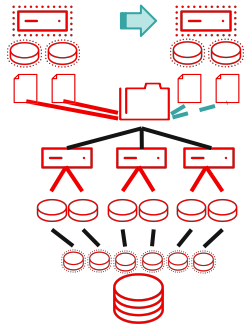
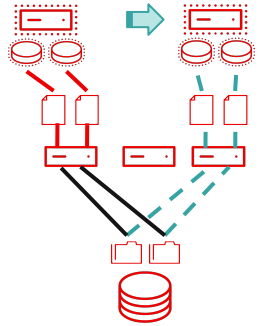
Shared File System



design	full **SDS** (native media)	**native storage** subsystems (enterprise storage)	**SDS** on top of **native storage** subsystems	**virt FS** layer on top of **native storage** subsystems	**shared FS** service provided by vendor
CSI	native	to array manager	CSI to SDS layer	CSI to virtual FS layer	CSI to suitable layer
topology	any (AZs; HCI, separate)	limited by sync replication	any / depends on sync repl.	depends on vendors	single / multi: vendor
host storage	block	block	block	block	virtual filesystem share (ex: NFS)
#devices	many, no multipath need	limited, dynamic multipath need	any; a few, static multipath need	any; a few, static multipath need	any; limit on shares
PV backing	block	block	block	file	file
footprint	replica / EC per media	media redundancy	replica * media redundancy	replica * media redundancy; files	native/media redundancy (vendor)

SDS => Software Defined Storage; CSI => Container Storage Interface; vols => volumes; EC => Erasure Coding; FS => file system

How to access storage?

	Full SDS	Traditional Storage with CSI	Traditional Storage w/o CSI	Traditional storage with Virtual File System	Shared File System
					
design	full **SDS** (native media)	**native storage** subsystems (enterprise storage)	**SDS** on top of **native storage** subsystems	**virt FS** layer on top of **native storage** subsystems	**shared FS** service provided by vendor
security	CSI within cluster	CSI to external storage	CSI within cluster	CSI within cluster	CSI to external storage; shares
perf/latency	SDS specific	native storage	native storage & SDS specific	native storage & FS specific	native FS
storage network	east-west, lossless	north-south / intra DC: east-west	east-west, lossless	north-south (east-west >solution); intra DC: east-west	north-south (east-west >solution); intra DC: east-west
PV access	via SDS nodes	all nodes need direct storage access	via SDS nodes; SDS nodes need storage access	via FS nodes; FS nodes need storage access	all nodes need direct storage access
risks	capacity, latency; network; layered org, internal media (blade), CPU/mem resources	#vols (traffic; host), multipathing, speed of CSI, snapshot & cloning; automated SAN zoning, overwhelmed storage admins	high capacity, latency, changed traffic -> other workloads, network; complexity, costs, additional layer	latency, network; complexity, costs, additional layer	#shares (storage, host); network; security; snapshot & cloning; possible bottleneck based on protocol (e.g. NFS)

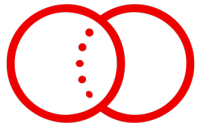
Integration into Data Services ecosystem

Encryption

Data Privacy and Isolation

Backup / Restore

Integration with Enterprise Ecosystem Backup Landscape



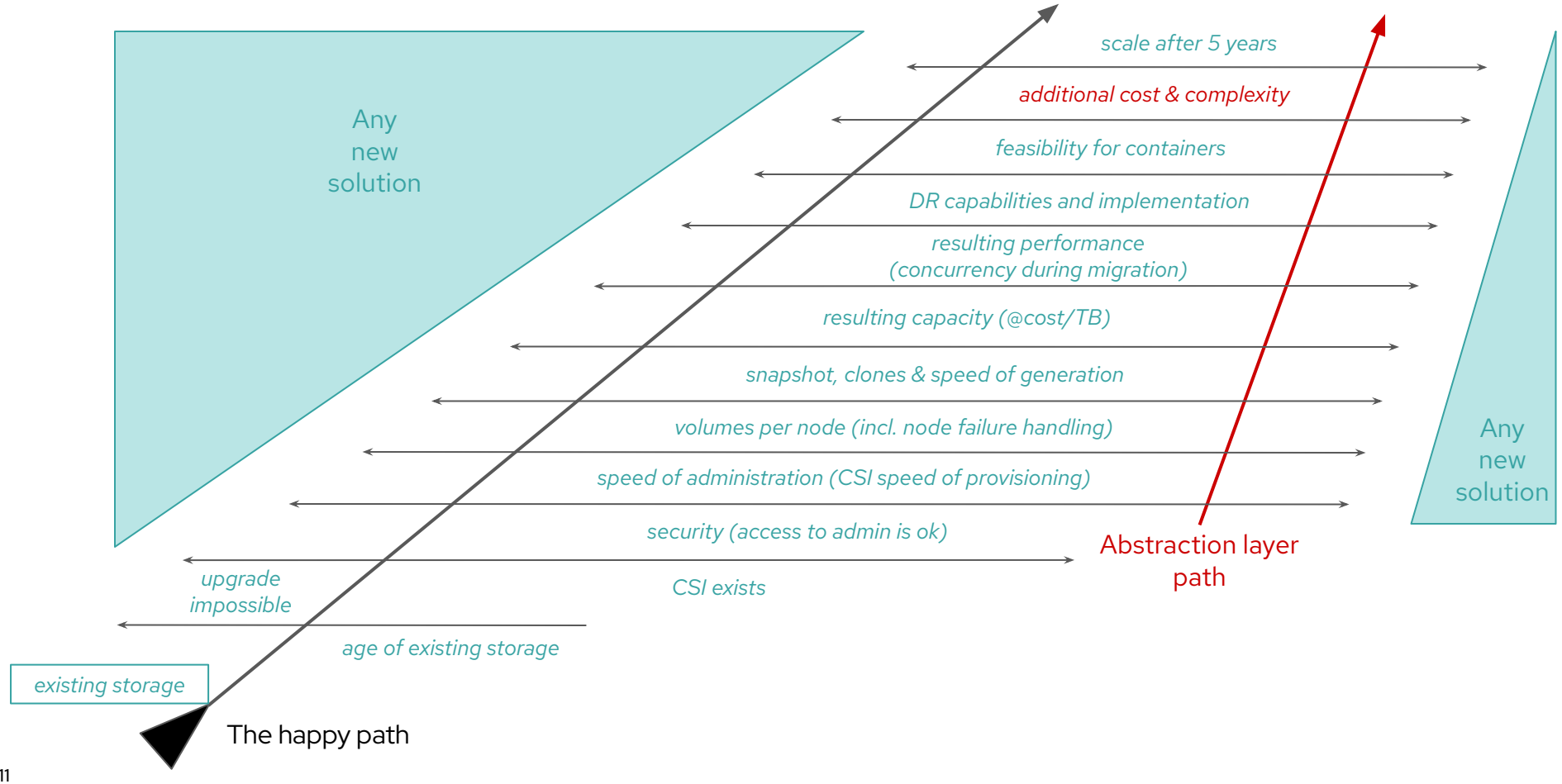
Disaster Recovery

Automation of Disaster Recovery handling

Maintenance & Scale

Facilitation of Infrastructure Maintenance and Scale of Workloads

Evaluation Graph



Scoring

Need=> / Approach:	Scalability [1]	Performance	Speed of administration [2]	Security (bridging boundaries)	Time to live (solution life time)	Cost	Complexity (integration effort)
classic SAN/NAS	-	++	--	-	-	+	-
abstracted SAN *	o	o**	++	++	mixed	--	--
SDS	++	-	++***	++	++	+	+
special SDS ****	+	++	++	++	+	++	o

The way out:

- mix & match - multiple solutions per cluster are ok
- on point solutions
- evaluate what works best for use cases
- avoid slowly reacting solutions for scale & focus on massive needs
- limit # VMs with needs for special storage per node

* Several limitations may apply, like access from all nodes to same storage, splitting of capacity into multiple file systems
** Change of traffic pattern through file access handling and filesystem structure
*** Individual capabilities may vary
**** Specialized high performance / low latency software defined storage solutions

1) #volumes per node and #volumes per subsystem
2) Ability to apply changes needed for adding & remapping upon node failover

Lessons learned

- storage is your **foundation - understand options and potential challenges**

- **starting** with the **right solution is key** to project success
- not all storage is good **if tested in a PoC - test for scale and speed** - understand the **limits early**

- know your enemy and **train**, try to understand the new technology

- **network is not a given** - plan for capacity and performance
- ideally, **separate networks** for SDS and use large MTU size

- **understand workloads density and patterns** - do not assume

- **collect data if you fail - your not alone**

Future Fitness

Avoid blocks by central single solutions

- single central solution & gravity
- version interdependency
- maintenance interdependency
- performance interdependency
- availability of new options

Plan for changes

- new technology with different requirements
- business process evaluation

Embrace the future

- build processes around your goals, not the actual technology
- technology and customer demand change quickly

Red Hat and Partner solutions

A fast growing ecosystem - examples only

Full SDS

- Red Hat OpenShift Data Foundation
- portworx by Pure Storage
- IBM Fusion
- VAST
- DELL Technologies
- lightbits

Traditional Storage with CSI

- NetApp
- HITACHI Inspire the Next
- DELL Technologies
- INFINIDAT
- Hewlett Packard Enterprise

Traditional Storage w/o CSI

- Red Hat OpenShift Data Foundation
- portworx by Pure Storage

Traditional storage with Virtual File System

- arctera
- IBM Fusion

Shared File System

- NetApp

Red Hat Summit

Connect

Thank you


linkedin.com/company/red-hat


facebook.com/redhatinc


youtube.com/user/RedHatVideos


twitter.com/RedHat

Jetzt Session bewerten!

Einfach QR-Code scannen, Session aus der Liste wählen und bewerten. **Vielen Dank!**

red.ht/rhsc-darmstadt-feedback