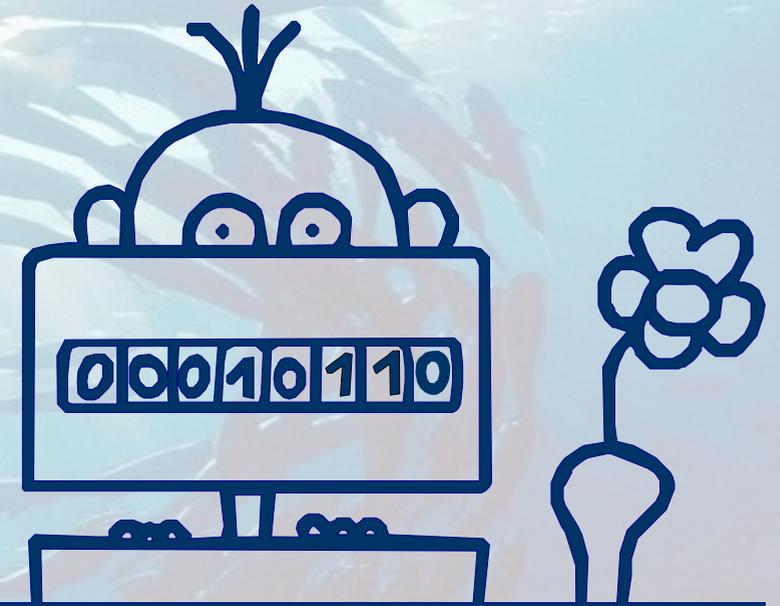


Oft geht mehr als man denkt

Backup, Storage Thin Provisioning, Datenkompression
mit der OSL Data Center Engine 4.8

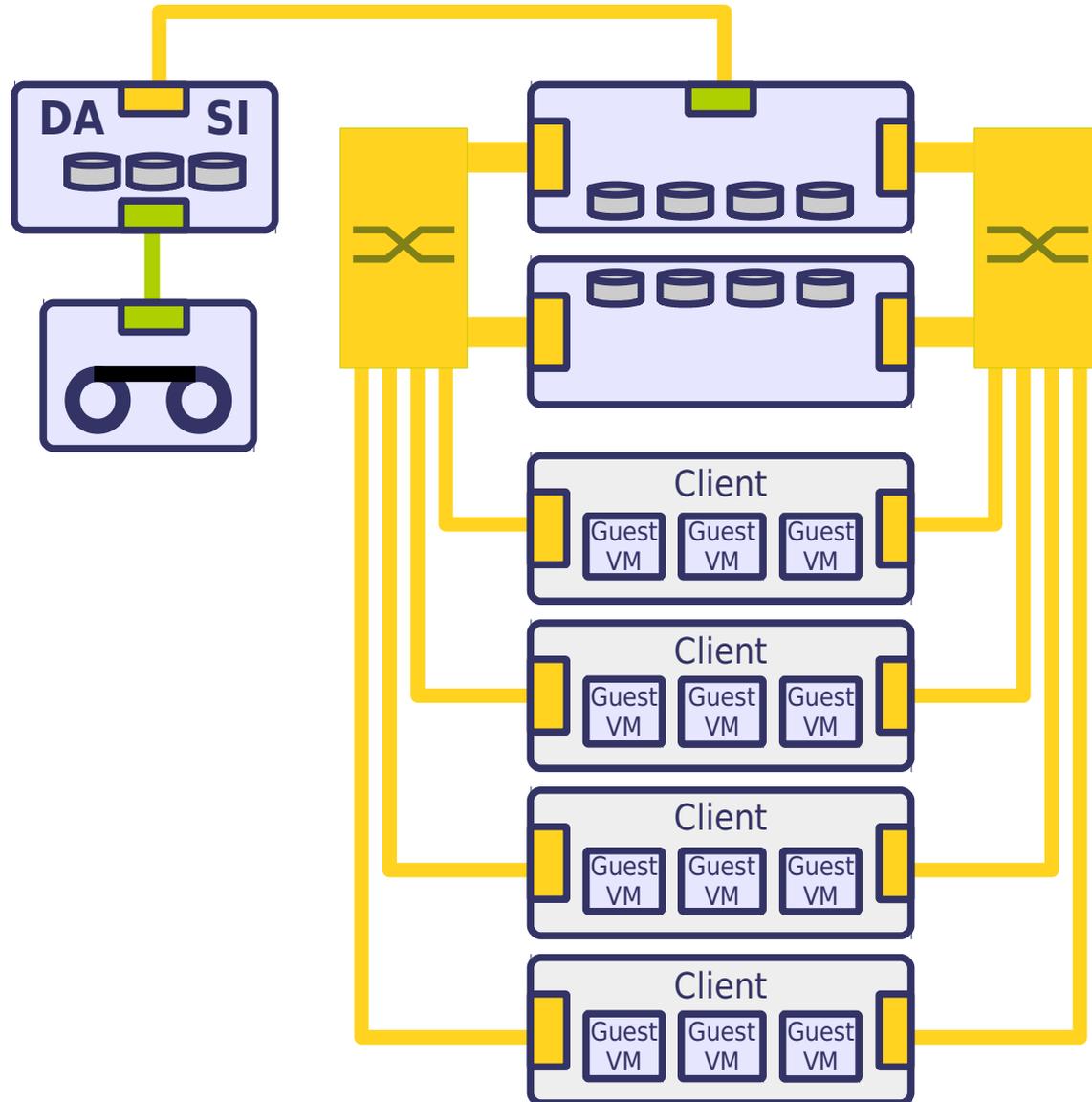


OSL aktuell

Schöneiche / Berlin • 29. Mai 2024

Konventionelles Backup über Netzwerk

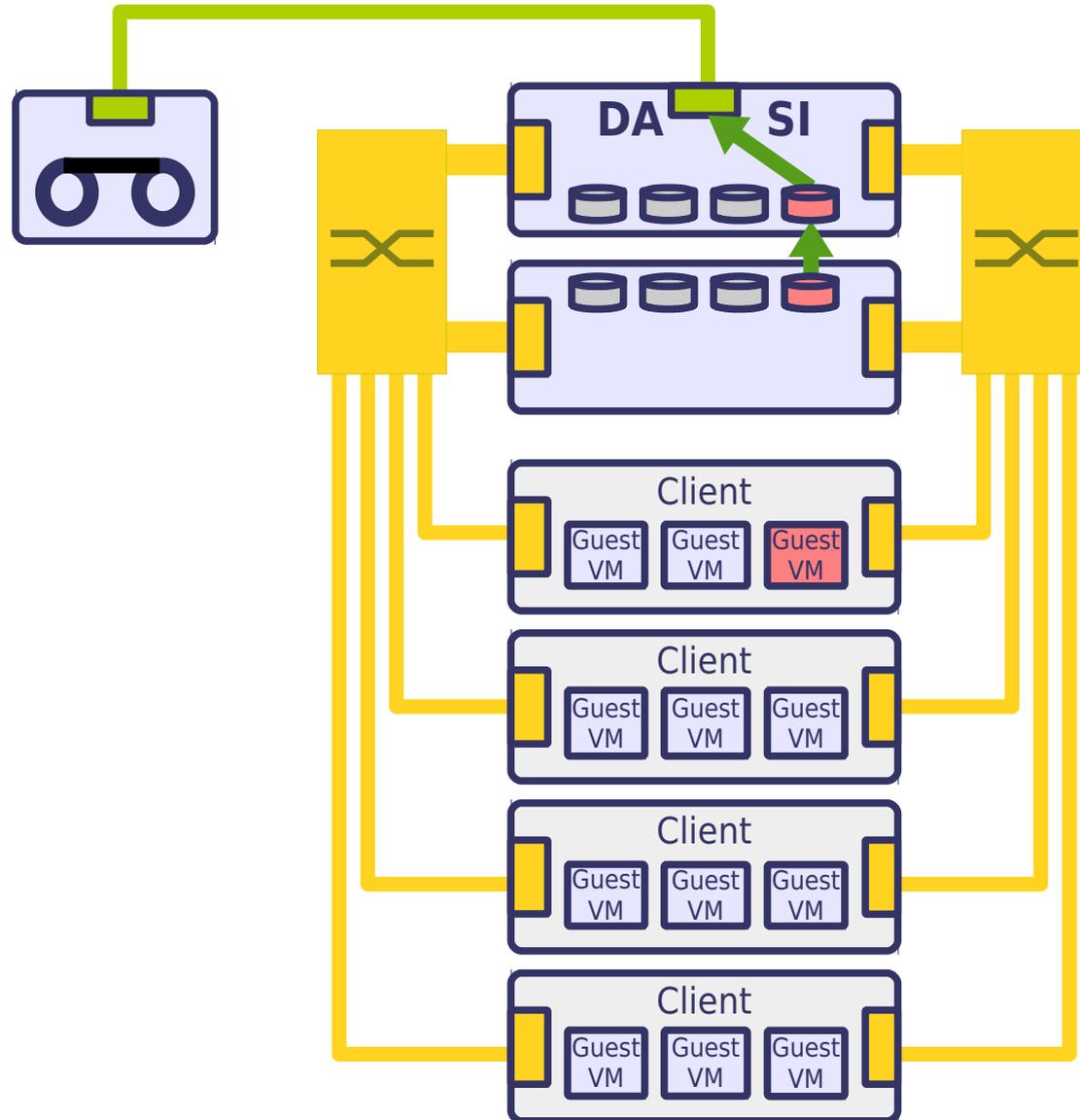
wie mit klassischer Hardware



- Nutzung der Netz-Backup-Lösung
 - VM ist an Dateneinlieferung beteiligt
→ Backup-Clients auf den VMs
 - Daten laufen über das Netz
 - Disk-Image-Backup etwas schwierig
 - File-Backup
→ hängt von Möglichkeiten der Backup-Lösung ab
- **geht immer**

Zentrales Backup to Disk ...

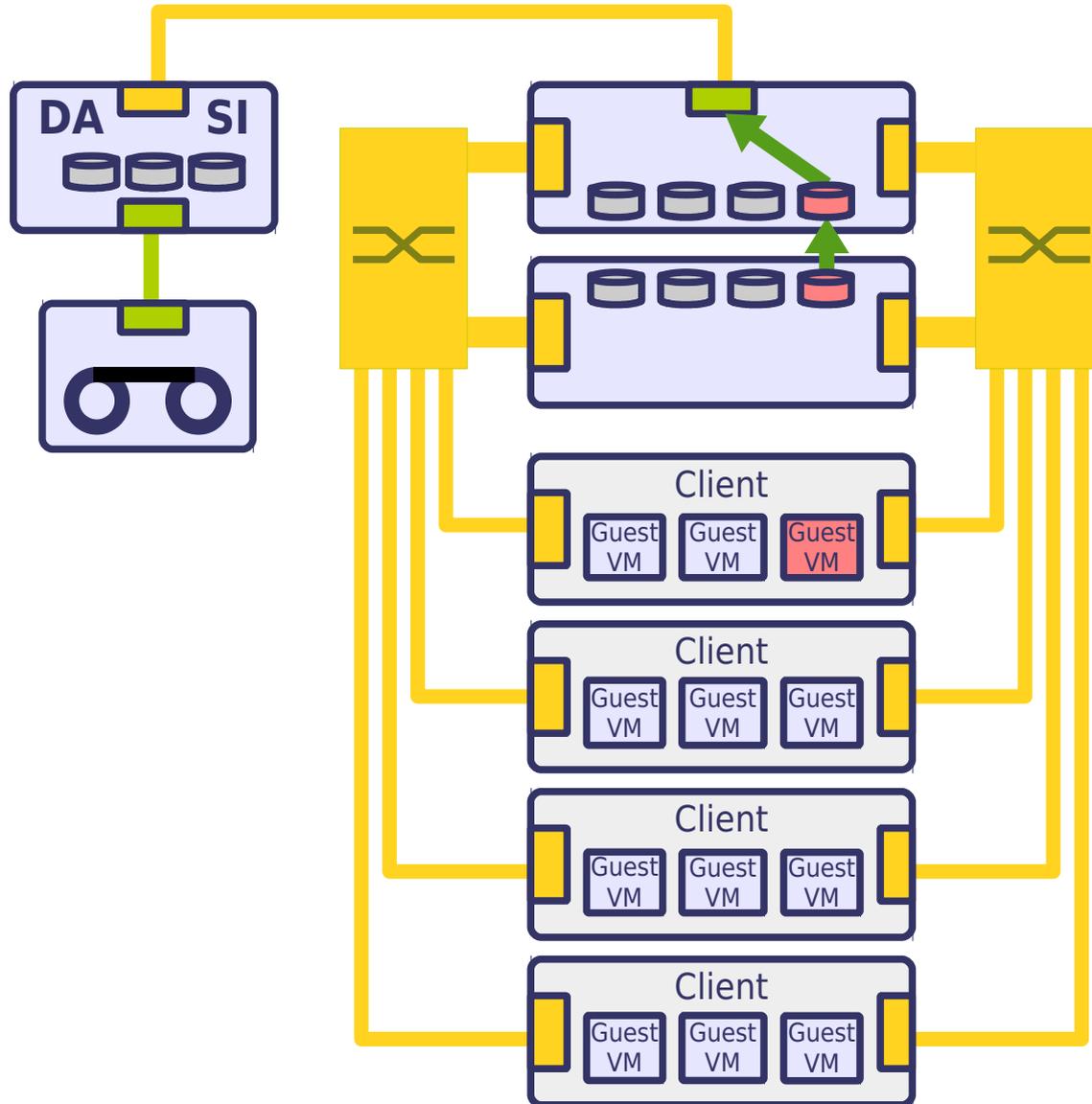
und dann weiter



- Nutzung der Spiegelung / Universen auf UVS
 - Backup-to-Disk - dann weiter auf Tape
 - inkrementelle Verfahren bereits bei der Spiegelung möglich
 - VM ist an Dateneinlieferung nicht beteiligt
 - keine Backup-Clients auf den VMs
 - nur eine Maschine am Backup/'Restore beteiligt
 - Daten laufen nicht mehr über das Netz
 - Disk-Image-Backup z. B. für
 - Datenbanken / komplette Landschaften (atomarer Snapshot)
 - proprietäre Formate
 - File-Backup
 - DASI-Server muß Dateisystem lesen (auch schreiben) können
- **hat bestimmte Voraussetzung**

Zentrales Backup to Disk ...

geht auch so



- DASI-Server greift auf Spiegelvolumen zu
 - DASI-Server auch als VM möglich
- **hat bestimmte Voraussetzung**

Grundsätzliche Überlegungen

Die Implementierung wird sehr individuell sein



- Was soll gesichert werden
 - Datenbank bzw. Verfahrenslandschaften vs. Datei(server)
 - Natur der Daten bestimmt Verfahren der Datensicherung / -Wiederherstellung
- Wie soll wiederhergestellt werden?
 - Dateien oder ganze Systeme?
- Wie oft und in welchen Situationen wird wiederhergestellt?
- Aufbewahrungsfristen, Ablaufsystem für Ansicht historischer Daten u. a.
- Storage-Universen bieten viele Möglichkeiten → Denkanstöße
 - ggf. auch mehrere Versionen können On-Disk vorgehalten werden → restorefreier Wiederanlauf!
 - Volumes können der VM (ggf. RO) präsentiert werden
 - Wiederherstellung einzelner Dateien anhand des Backups von gestern in der VM möglich
 - Wiederherstellung einzelner Dateien aus der Bandsicherung auf spezielle Restore-Container möglich

u. v. m.

LFS - Local File Storage

Die Idee

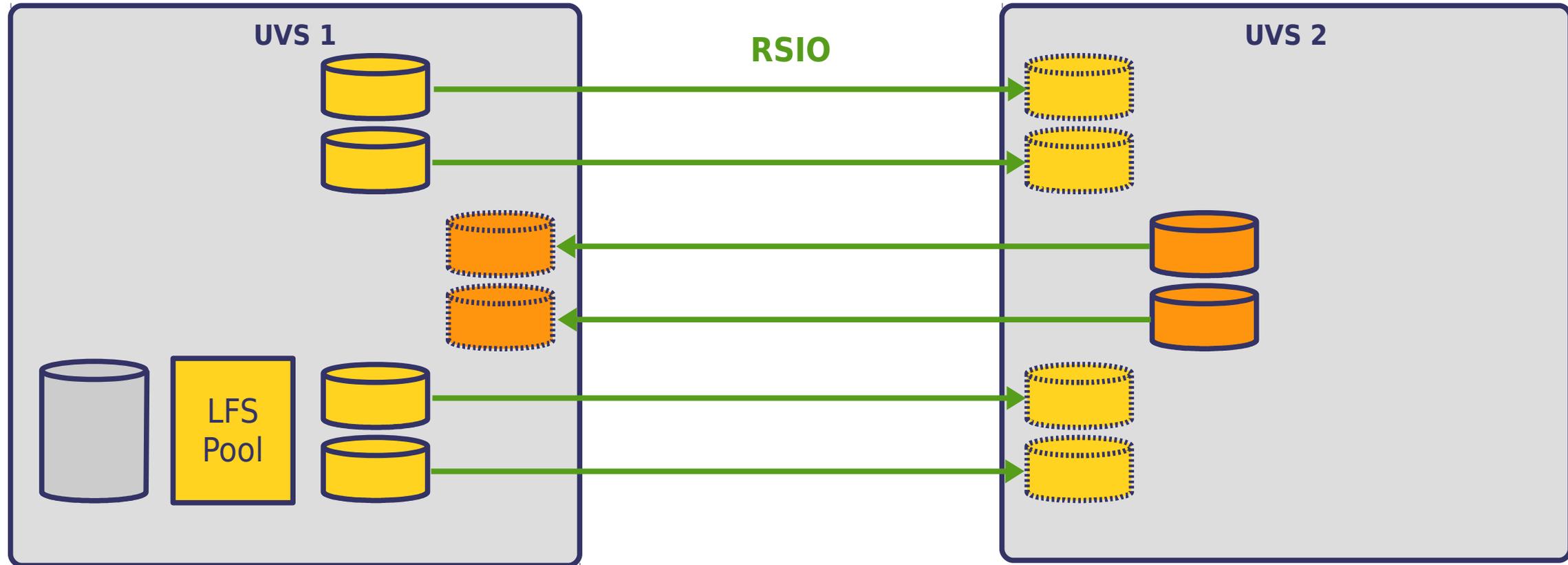
Build your work on top of others



- bestimmte nachgefragte Funktionen sind Dateisystemfunktionen
- bisheriges Backend im Storage Cluster und UVE: Entire Disks
- VSD beherrscht weitere Backends: Disk, Partition, Block Range, File, RAM, RSIO Device
- Lokale Dateien können als Teile eines lokalen Storage-Pools dargestellt werden
→ ermöglicht Überprovisionierung und Datenkompression
- Clusterweiter Zugriff dennoch möglich → RSIO

So funktioniert das - Beispiel UVE

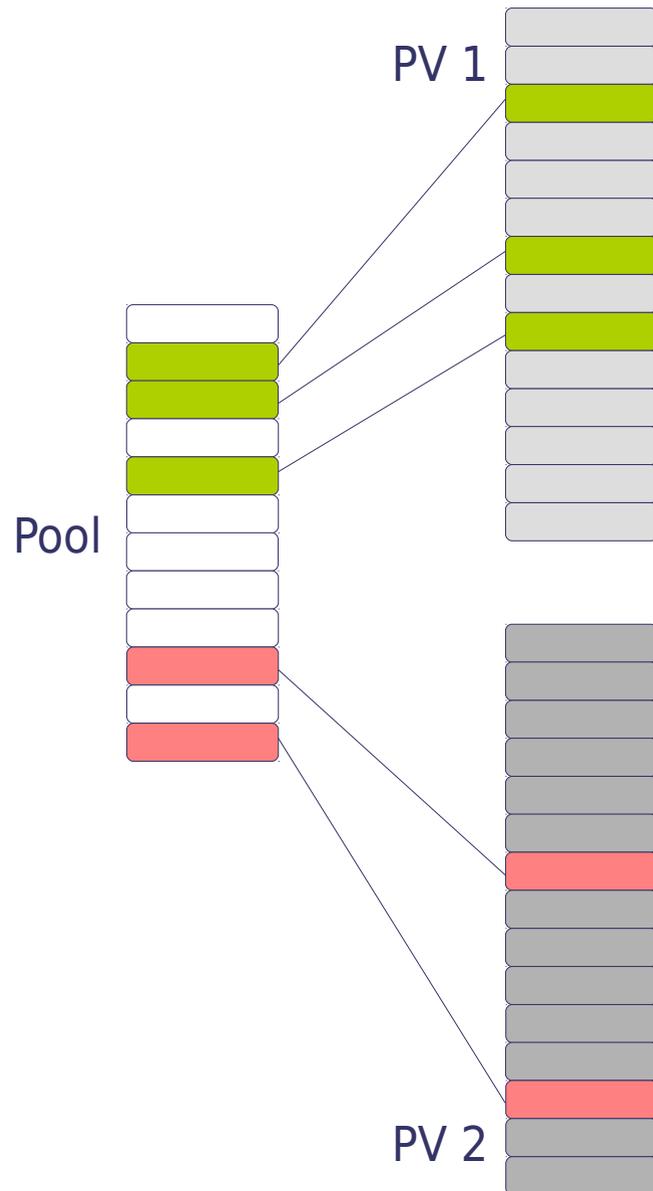
Lokale Pools global verfügbar



- VSD präsentiert wieder ein Block-Device
- Das Gerät steht im gesamten Cluster zur Verfügung (auch auf den UVC)

So funktioniert das - Thin Provisioning

Mit "Sparse"-Volumes Spielräume schaffen



- Volumes werden zunächst aus "heißer Luft" erzeugt
- Erst beim Schreiben werden Blöcke allokiert (verbraucht)
- Unbeschriebene Blöcke liefern "heiße Luft" - NULL-Blöcke
- VSD implementiert Freigabe von "NULL"-Blöcken
- "Scratch" gibt Blöcke eines Volumes frei

- Wichtig: Monitoring der Pools!

- Request-Registration und WOW-Protection im VSD-Treiber verbessern OS-Mechanismen
- LFS-basierte Volumes können gespiegelt werden
- “avmove” kann LFS in reguläre Block-Devices überführen und umgekehrt (auch online)
- Derzeitige File-Backends: - regular File
- sparse File
- Auch Block-Devices (fremde Treiber) können so eingeschleust werden
 - sofortiger Zugriff bei Migrationen
 - Konvertierung in OSL-Storage-Formate später und im laufenden Betrieb möglich
- Weitere Backends denkbar: - QEMU-Images (qcow2, vmdk, vdi, ...)
- ggf. auch Zusatzfunktionalitäten (Snapshots usw.)



virtualization and clustering - made simple