

Virtuelle Infrastrukturen mit Linux ...



... und deren Integration in OSL SC

Christian Schmidt
Systemingenieur

● **Virtualisierung**

- *"Aufteilung oder Zusammenfassung von Ressourcen"*
- *Unterschiedliche Bereiche für die Virtualisierung in der IT*
 - *Hardware, Desktop, Software, Memory, Storage, Network*
- *In jedem Bereich auch noch implementierungsspezifische Unterschiede*
- *Keine neue Technik:*
 - *Hardware-Virtualisierung seit 1967 (IBM CP-40)*
 - *Anfänge der Softwarevirtualisierung mit chroot (seit 1979 in Unix)*
- *Viele Möglichkeiten, viele Produkte*
 - *über 50 Softwarepakete, die vollständige Hardware-Virtualisierung ermöglichen*
 - *großer Aufschwung in der x86-Welt seit 2000*
 - *seit 2005 hardwareunterstützte Virtualisierung bei x86-Prozessoren (Intel VT-x AMD-V)*

- **Hardware-Emulation**

- *Hardwarekomponenten werden durch Software simuliert*
- *unmodifizierte Programme und Betriebssysteme können so gestartet werden*
- *sehr langsam, da die VM keinen Zugriff auf reelle Hardware hat*
- *Unterschiedliche CPU Architekturen sind kein Problem (x86-Programme auf Sparc, Gameboy-Spiele auf x86 PCs)*



Qemu

Virtual PC (x86 on Mac)

- **Hardware-Virtualisierung**

- Teile der physischen Hardware werden einem Gastbetriebssystem zur Verfügung gestellt
- Gast hat vollständigen Zugriff auf seine physische Hardware
- Jedes Programm und Betriebssystem, das auf der realen Hardware läuft muss ohne Modifikation in der virtuellen Maschine laufen
- Für x86 erst seit 2005 mit Einführung von AMD-V bzw. Intel VT-x möglich



- **Container (Betriebssystem-virtualisierung)**
 - *Funktion des Hostbetriebsystems*
 - *Kein zusätzliches Betriebssystem in der virtuellen Maschine notwendig*
 - *Prozesse werden in isolierten Instanzen gestartet*
 - *Ressourcen können gemeinsam genutzt werden*
 - *In den Containern können keine Treiber geladen werden*



Solaris Zonen

Linux Container (LXC)

● **Serverpartitionierung**

- *Aufteilung der vorhandenen Hardware in Teilmengen und Zuordnung zu autonomen Betriebssysteminstanzen*
- *feste oder dynamische Ressourcenzuordnung*
- *Hardwarepartitionen, Softwarepartitionen, Blades*

● **Hypervisor**

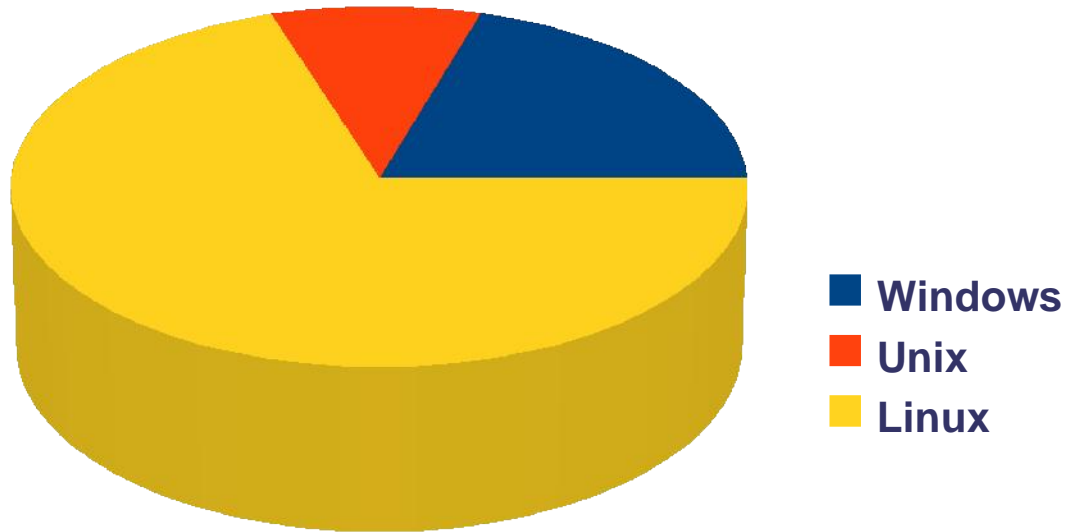
- *schafft die Ablaufumgebung für die virtuellen Maschinen (VMM – Virtual Maschine Monitor)*
- *nimmt spezielle Systemcalls entgegen und leitet sie an die Hardware weiter oder emuliert sie*
- *Typ 1: setzt direkt auf der Hardware auf (XEN, VMware)*
- *Typ 2: läuft als Applikation innerhalb eines Betriebssystems (KVM, VirtualBox)*

- **Paravirtualisierung**

- *Nutzung spezieller APIs im Gastbetriebssystem für den Zugriff auf die Hardware*
- *Anpassung des Gastbetriebssystems ist notwendig*
- *Teilweise auch nur einige Treiber paravirtualisiert (Netzwerk, Storage, Grafik ...)*
- *Hohe Leistung, da optimierte Verfahren genutzt werden können*

Virtualisierte Infrastrukturen mit Linux

Der Rolle von Linux im Rechenzentrum



- *Mehr als 70% aller Webserver laufen unter Linux*
- *Linux ist aus dem tägl. IT-Betrieb nicht mehr wegzudenken*
- *Es gibt fast alle Serverapplikationen für Linux*
- *hervorragende Serverhardwareunterstützung*

Marktanteil Serverbetriebssysteme (Webserver)
Stand: Juli 2009



- **Mehr als 20 kommerzielle und nicht-kommerzielle Virtualisierungslösungen für Linux vorhanden**
 - *Qual der Wahl*
 - *Teilweise zusammsetzbare Bausteine*
 - *sowohl spezielle als auch universelle Lösungen*
 - *Distributionen bieten teilweise eigene Frameworks, setzen meistens jedoch konsequent auf KVM oder XEN*
 - *günstige Lösungen – doch wo bleibt der Komfort?*

- **Distributionen nutzen oft eigene Virtualisierungsframeworks**

- *Red Hat Enterprise Virtualization*

- *basiert auf KVM*
- *zusätzliches Management Interface für eine einfache Handhabung von Deployment, Migration, High-Availability*

- *SUSE Linux Enterprise Server*

- *XEN oder KVM Virtualisierung*
- *SUSE Linux Enterprise Virtual Machine Driver Pack – Paravirtualisierte Treiber*

- **libvirt ist die gemeinsame Grundlage**

- *Bibliothek zur Administration von virtuellen Maschinen*
- *Unterstützt KVM, Qemu, XEN, LXC, OpenVZ, VMware, Hyper-V*



- ***KVM wird wahrscheinlich die bevorzugte Virtualisierungslösung unter Linux***
 - *Teil des Linux Kernels*
 - *Hervorragende Performance*
 - *SPECvirt Performance Benchmark -> sehr gute Ergebnisse*
 - *skaliert sehr gut auf großen Systemen*
 - *max. 4069 Cores, 64TB RAM (64vCPUs, 2TB RAM pro Gast)*
 - *Open Virtualization Alliance*
 - *über 150 Unternehmen unterstützen KVM und die Entwicklung von KVM weltweit*
 - *Paravirtualisierte Treiber für Linux und Windows*

- **Vorteile für den Einsatz von Linux in virtuellen Infrastrukturen**

- *günstiger Anschaffungspreis*
- *große Auswahl an Hypervisoren (auch spezielle)*
- *gemeinsame Schnittstelle mit libvirt*
- *flexible Umgebung*
- *sehr gute Unterstützung aktueller Serverhardware*
- *große Community*

- **Nachteile**

- *viele unterschiedliche Tools für die verschiedenen Aufgaben*
- *schlechte Gesamtintegration (Speichermanagement, Hochverfügbarkeit, VM-Management)*

Virtualisierte Infrastrukturen mit Linux

Einsatzmöglichkeiten von RSIO und OSL SC

- **Mit RSIO und OSL SC für Linux können einige Nachteile wettgemacht werden**
 - *Hochverfügbarkeit, Applikationsmanagement und Speicherverwaltung aus einer Hand*
 - *Einfaches Speichermanagement*
 - *Volumes anlegen, moven, clonen*
 - *Backups ganzer VMs erzeugen*
 - *globaler Storagepool*
 - *günstige, schnellste und einfachste Speichieranbindung über RSIO*
 - *Unterstützung von P2V Migrationen ganzer (OSL SC) Applikationen*

Virtualisierte Infrastrukturen mit Linux

Verbesserte Unterstützung von VMs in OSL SC 4.0

Mit OSL SC 4.0 gibt es Virtual Nodes und Virtual Maschine Applications

- **Virtual Nodes**

- *virtuelle Maschine, welche als vollwertiger Node im Cluster betrachtet wird*
- *haben Zugriff auf virtual Volumes und die CCF*
- *kann Applikationen des Clusters starten und XDM Operationen durchführen*
- *Solaris (Zonen) oder Linux (benötigt den OSL-Treiber)*
- *tauchen in der Knotenliste des Clusters auf – keine Applikation*
- *Anschalten und Ausschalten der Virtual Nodes über NPC Methoden*

Virtualisierte Infrastrukturen mit Linux

Verbesserte Unterstützung von VMs in OSL SC 4.0

- **Virtual Maschine Applications**

- *Virtuelle Maschine, welche als Applikation im Cluster gesehen wird*
- *Erstellbar über Applikationemplates*
- *Kann auf physikalischen Nodes mit installiertem Hypervisor gestartet werden*
- *Start und Stopp der VM mit ACO-Methoden (appstart/appstop)*
- *Migration und Hochverfügbarkeit mit der ACO*
- *Keine Begrenzung auf bestimmte Betriebssysteme*
- *Backupintegration – Point-In-Time Backup der gesamten VM*

Virtualisierte Infrastrukturen mit Linux

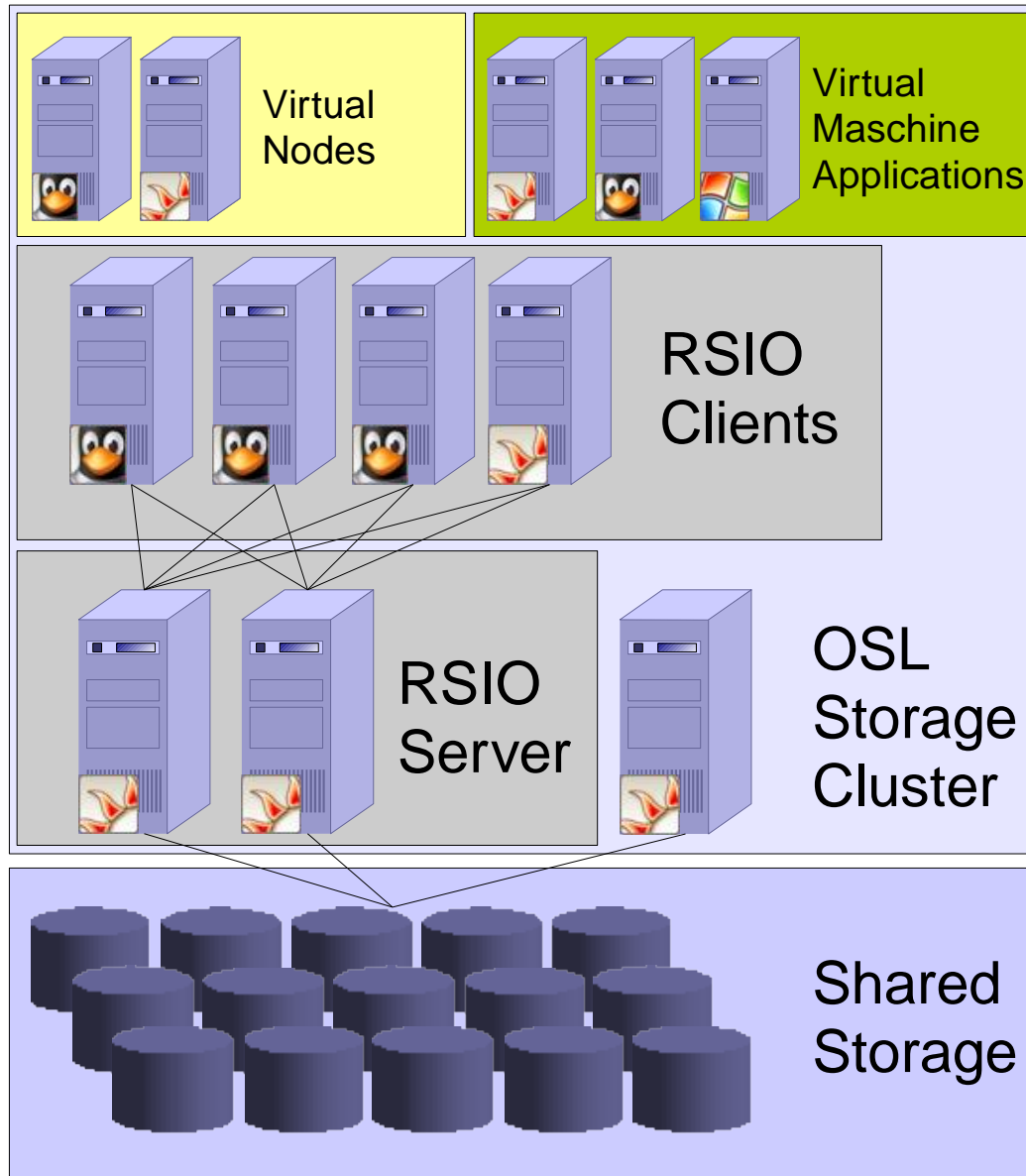
Verbesserte Unterstützung von VMs in OSL SC 4.0

- **Einfache OS-Migrationen mit OSL Software**

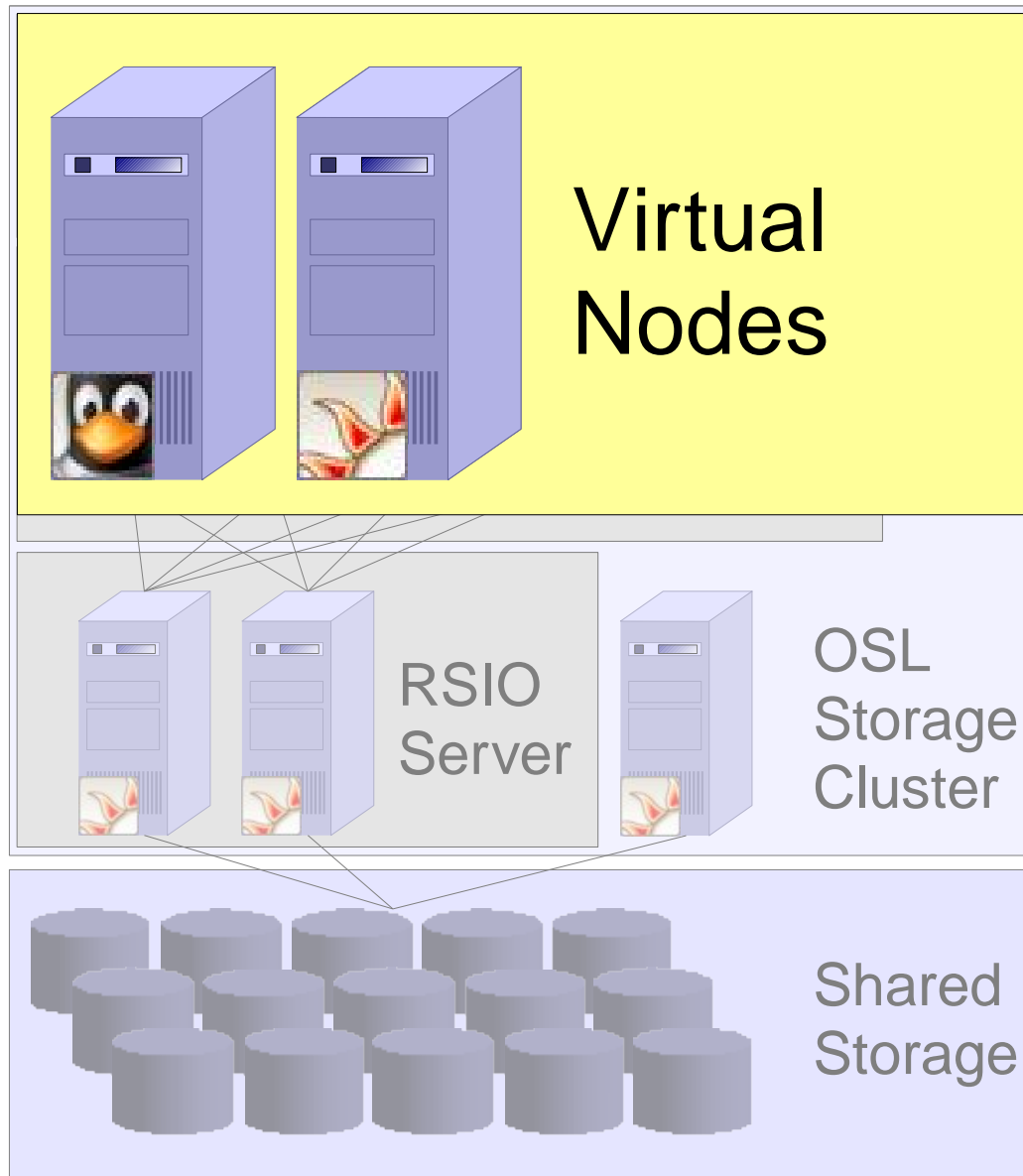
- *Applikationen können von physikalischen Nodes in virtuelle Nodes verschoben werden*
 - *Betriebssystem muss gleich bleiben*
- *Konsolidierung kann vereinfacht werden*
- *Mischcluster (Solaris/Linux) sind möglich*
- *Applikationen können sogar unter verschiedenen Betriebssystemen und Betriebssystemreleases geschwenkt werden (z.B. NFS oder Samba)*

Virtualisierte Infrastrukturen mit Linux

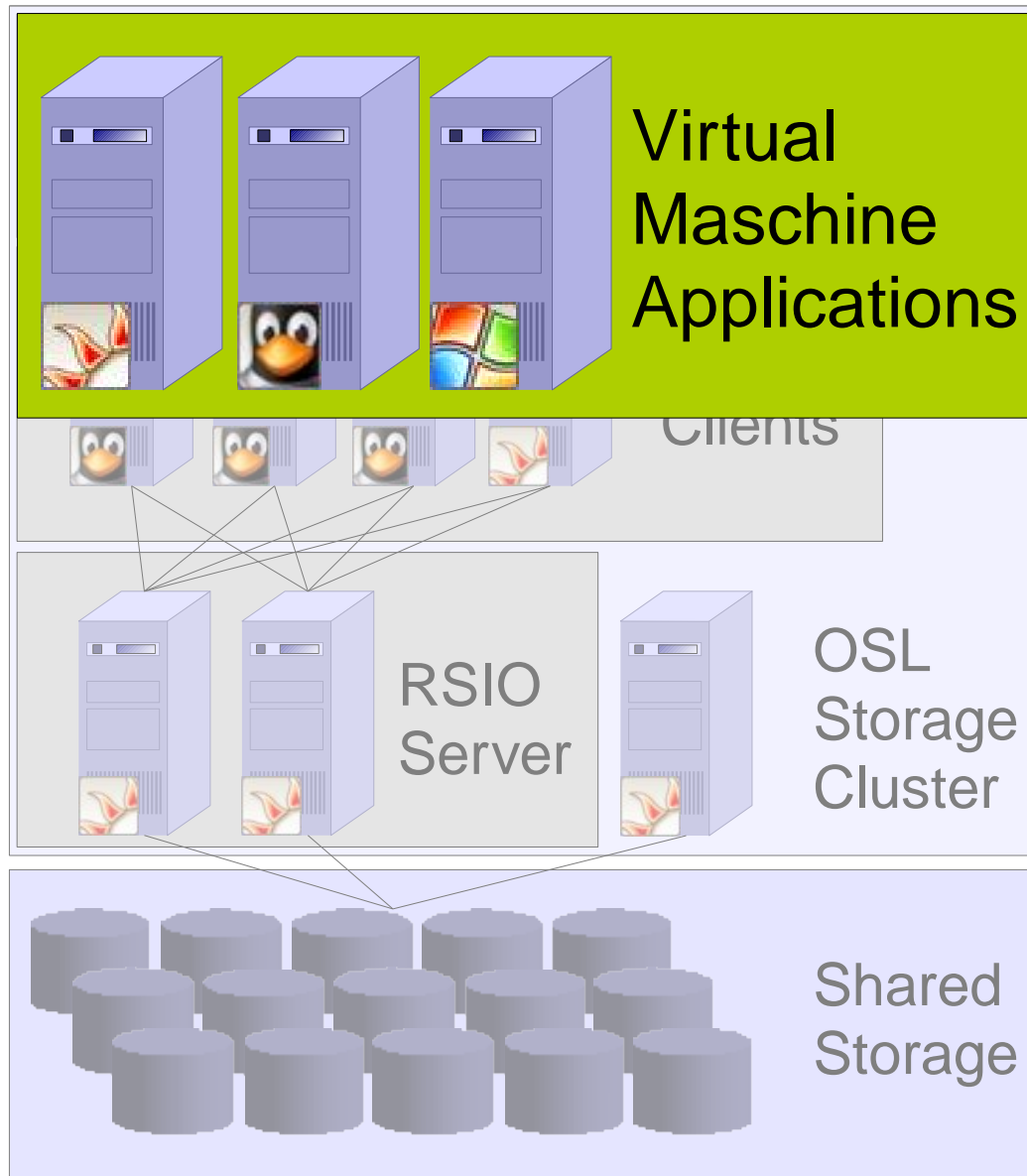
Mögliche Setups mit OSL SC 4.0



- **Cluster gebildet mit shared Storage, RSIO Servern und RSIO Clients**
- **Shared Storage ist auf allen Nodes verfügbar**
 - gleiche Namensraum
 - globale Storageadministration
- **Run Application everywhere**



- *Laufen auf physischen Nodes mit installiertem Hypervisor*
- *Benötigen OSL-Treiber (Linux oder Solaris)*
- *Haben Zugriff auf Application Volumes und CCF*
- *können Applikationen starten oder Applikationen von ausgefallenen physischen Nodes übernehmen*
- *laufen immer auf physischen Nodes des Clusters*
- *mögliche Hypervisor: Solaris Zonen, KVM, XEN*



- *Laufen auf physischen Nodes mit installiertem Hypervisor*
- *Applikation läuft in der VM*
- *kann im laufenden Betrieb zwischen den physikalischen Nodes geschwenkt werden*
- *unabhängig vom Gastbetriebssystem*
- *VM ist auf Application Volumes installierte, hat aber keinen Zugriff auf weitere Application Volumes*
- *mögliche Hypervisor: KVM, XEN*

- **Virtualisierte Infrastrukturen mit Linux und OSL Storage Cluster**
 - *Professionelle Virtualisierungslösungen und innovative Speicherverwaltung*
 - *Management von virtuellen Maschinen über Distributionsgrenzen*
 - *Clustering, Load-Balancing, Hochverfügbarkeit aus einer Hand*
 - *Integration der virtuellen Infrastruktur in den bestehenden Cluster*
 - *Veränderungen im Speicherbackend können ohne Down-Time durchgeführt werden*
 - *höchste Flexibilität im Bereich der Server- und Storagehardware*

Was noch fehlt:

- **Tool für das Management von VMs im Cluster**

- *derzeit über die ACO mit Start- und Stoppskripten*
- *nur teilweise automatisches Deployment und Cloning*
- *Schnittstellen sind vorhanden (libvirt, ACO, apptool)*
- *Tool ist in Planung :)*

- *Welche Features sind wichtig?*

(Stroagemanagment, automatische Migration, Cloning, Automatische Installation, Monitoring, VM-Management, SNMP, HA, Spezielle Keyssequenzen (STRG_ALT_ENTF, RESET, POWER ON, POWER OFF, POWER_SEQ))