



RSIO Tutorial

Data Center Block I/O over Ethernet

OSL Technologietage
Berlin • 12./13. September 2012

Bert Miemietz

OSL Gesellschaft für
offene Systemlösungen mbH

RSIO - Remote Storage I/O

Eckdaten der neuen Technologie für LAN-attached (Shared) Block Devices



- *neues, von OSL entwickeltes Protokoll*
- *direkter Transport aller relevanten IO-Aufrufe (read, write, ioctl)*
- *integriert Verbindungsaufbau, Überwachung, Path-Multiplexing, Trunking*
- *fähig zu Selbstkonfiguration und Error Recovery*
- *kann alle modernen Storage-Szenarien abbilden:*
 - *einfache Server und Clients, ggf. mit Multipathing*
 - *Cluster von Storage-Servern (Targets)*
 - *Cluster von Storage Clients (Initiators)*
 - *integrierte Cluster von Servern und Clients*
 - *Storage Server Farms*
 - *Cloud-Konzepte*
- *besondere Eignung für Kombination mit Speichervirtualisierung*
 - *eingängige Namen*
 - *fdisk (Partitionierung) auf Clientseite entfällt*
 - *On-Demand-Allokation und Online-Rekonfiguration*
 - *viele weitere Sonderfunktionen*
 - *ermöglicht Administration vom Client aus*

OSL Gesellschaft für offene Systemlösungen mbH
www.osl.eu

RSIO – das Performance-Beast

Hohe Performance und beeindruckende Skalierbarkeit bereits mit 1GBit Ethernet



Server-Performance bei Cache Read / 8k

<i>iSCSI</i>	<i>10 Clients</i>	<i>100 Threads</i>	<i>7,6 Cores</i>	<i>31.000 IOPS</i>
<i>iSCSI / comstar</i>	<i>10 Clients</i>	<i>100 Threads</i>	<i>10,0 Cores</i>	<i>85.000 IOPS</i>
<i>RSIO</i>	<i>4 Clients</i>	<i>64 Threads</i>	<i>5,6 Cores</i>	<i>98.000 IOPS</i>
<i>RSIO</i>	<i>4 Clients</i>	<i>128 Threads</i>	<i>6,3 Cores</i>	<i>102.000 IOPS</i>

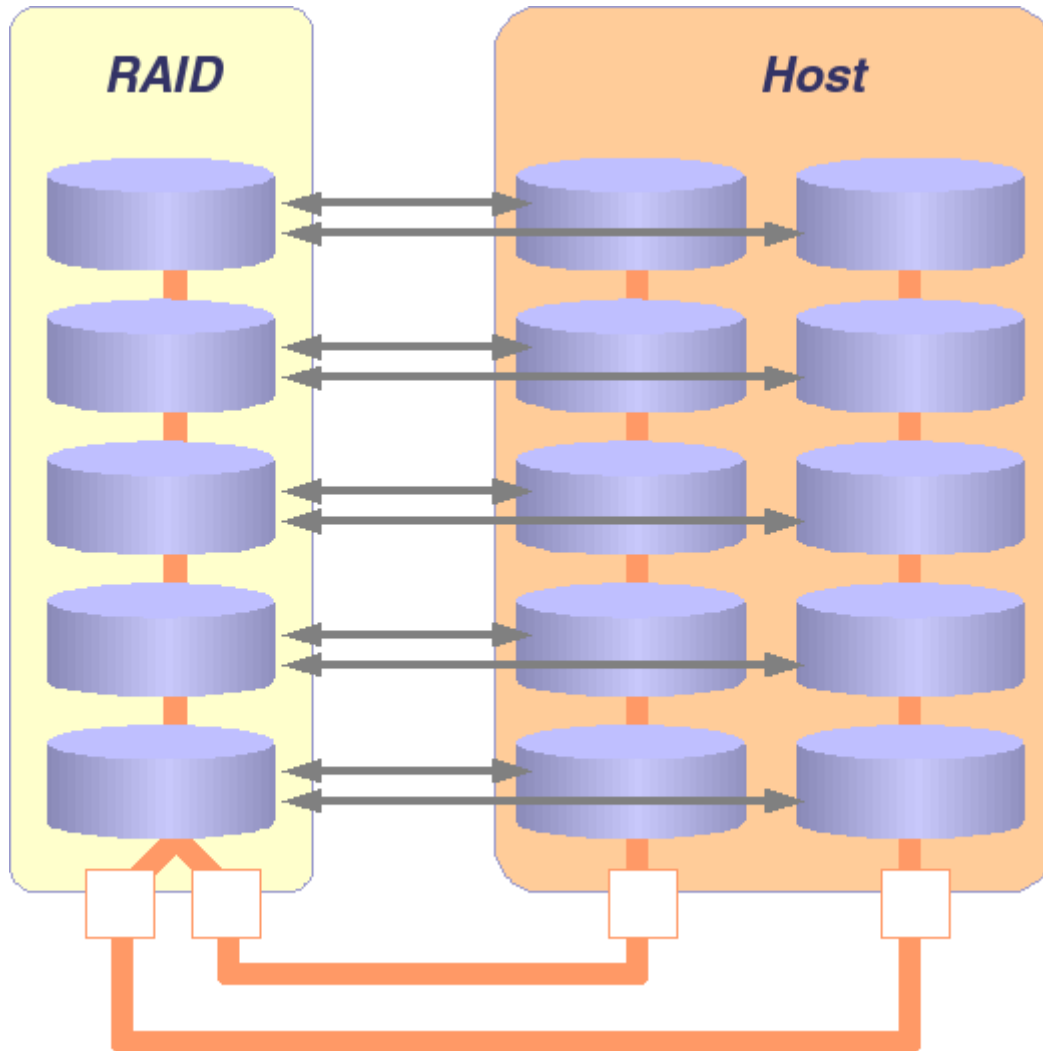
Client-Performance Throughput

<i>RSIO</i>	<i>1 x 1 GBit</i>	<i>ca. 0,5 Cores</i>	<i>> 110 MByte/s</i>
<i>RSIO</i>	<i>2 x 1 GBit</i>	<i>ca. 1,0 Cores</i>	<i>> 220 MByte/s</i>
<i>RSIO</i>	<i>4 x 1 GBit</i>	<i>ca. 2,0 Cores</i>	<i>> 440 MByte/s</i>
<i>RSIO</i>	<i>8 x 1 GBit</i>	<i>> 4,0 Cores</i>	<i>bis > 900 MByte/s</i>

OSL Gesellschaft für offene Systemlösungen mbH
www.osl.eu

Das Wichtigste: Ein völlig anderes Konzept

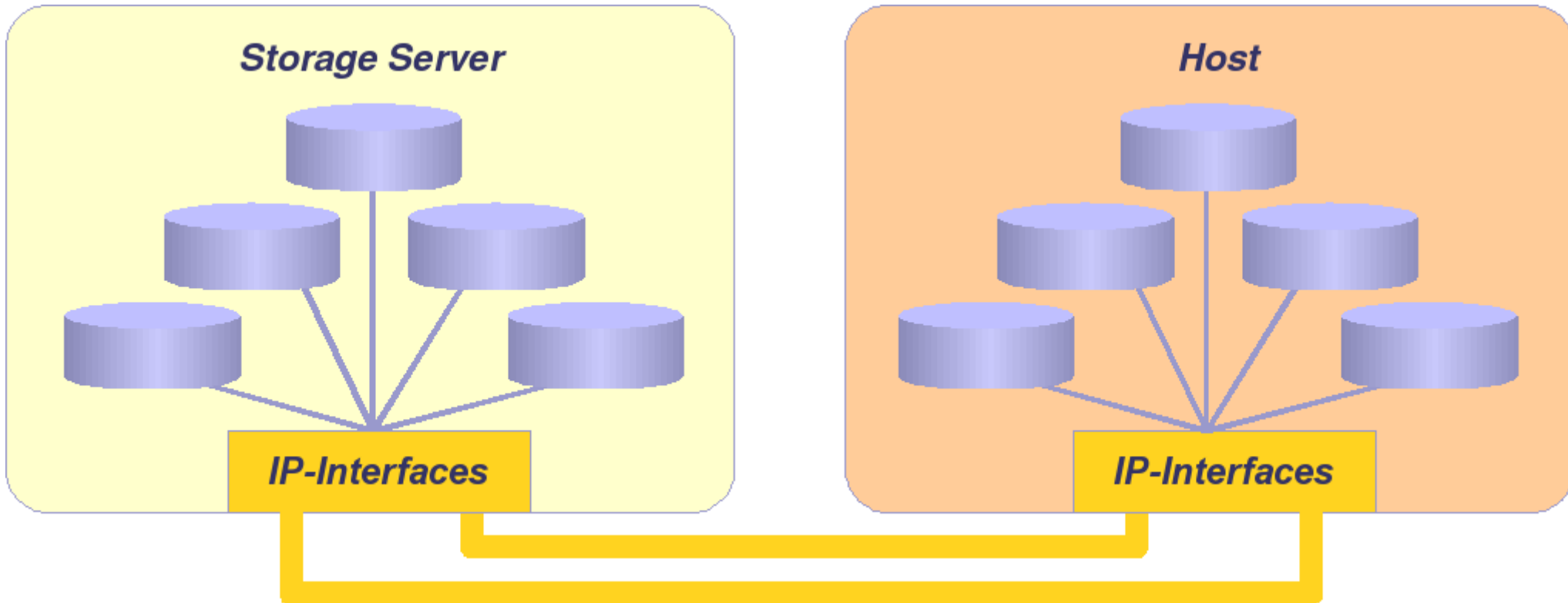
Warum die Welt mit SCSI / Fibre Channel so komplex sein kann



- SCSI über FC, iSCSI etc. emulieren die SCSI Daisy Chain
- Verbindungsstatus pro Lun, bei Multipath entspr. mehrfach -> Zuordnung zum Controller wichtig, daneben hunderte logischer Verbindungen
- SCSI-Layer hat kein Verständnis von der Netztopologie

Das Wichtigste: Ein völlig anderes Konzept

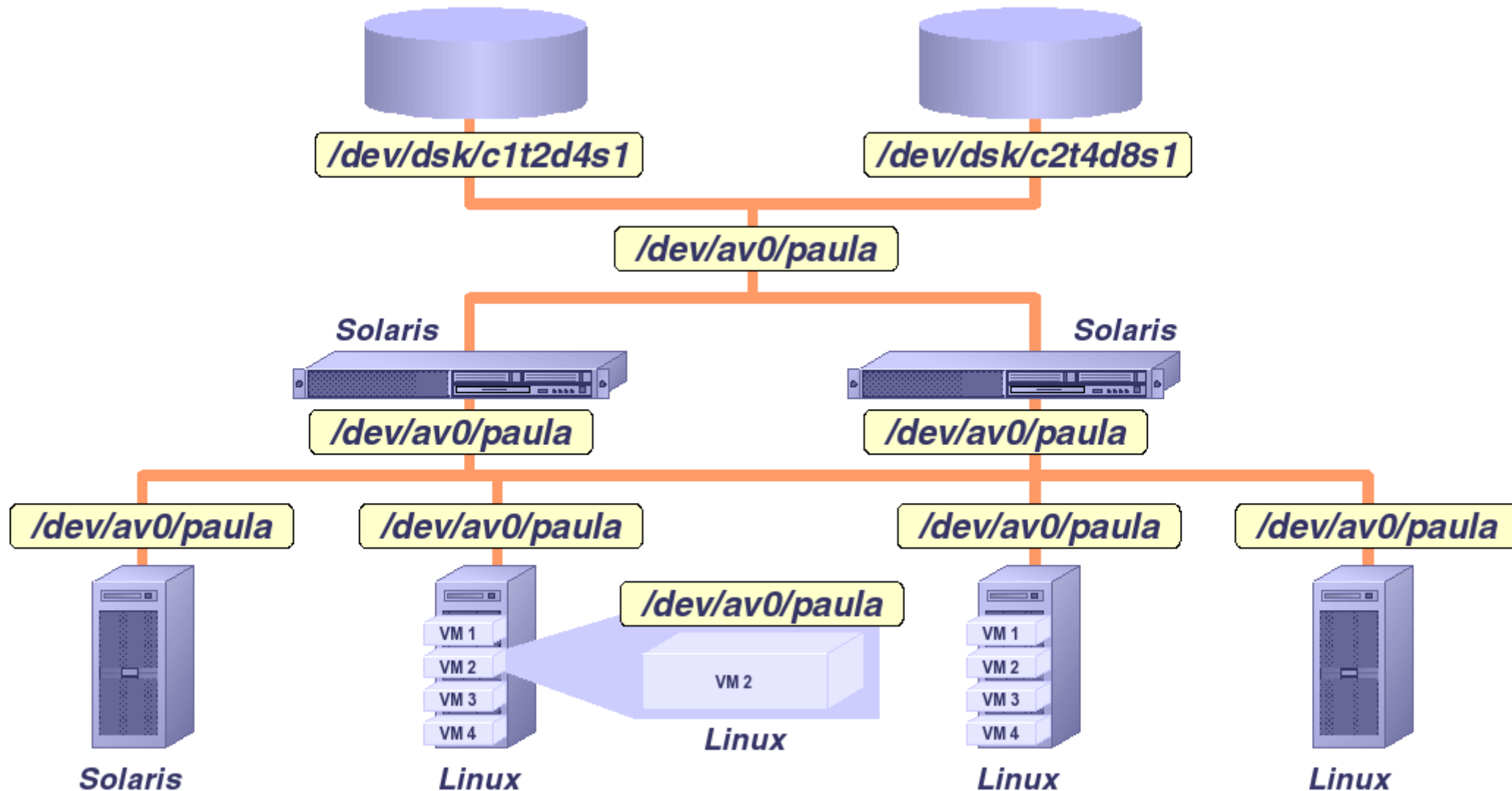
Mehr Übersichtlichkeit und völlig neue Möglichkeiten



- *Mit RSIO werden die Geräte über IP “getunnelt”*
- *Admin und Software interessieren sich nur noch für die logische Verbindung*
- *Die Geräte entstehen aus dem IP-Interface heraus und stehen damit überall zur Verfügung, wo ich ein IP-Interface habe.*

RSIO und Speichervirtualisierung

Was bringt die Verknüpfung mit einer clusterfähigen Speichervirtualisierung?

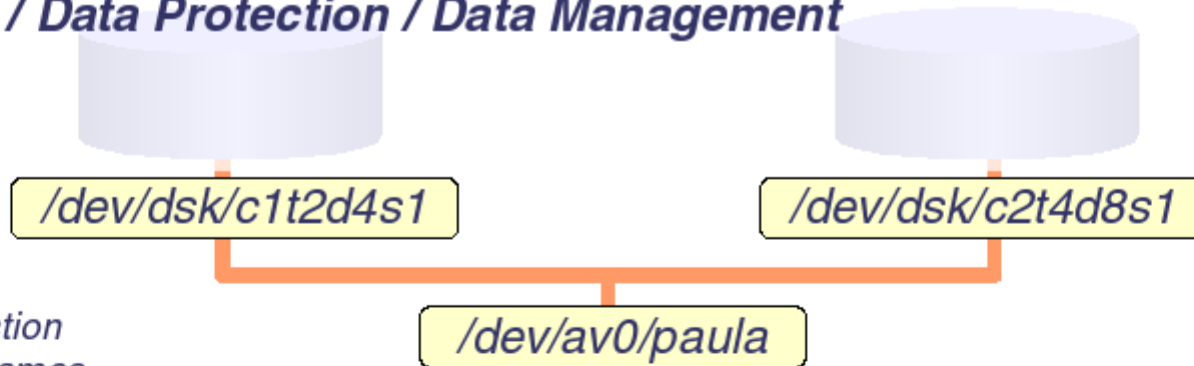


RSIO und Speichervirtualisierung

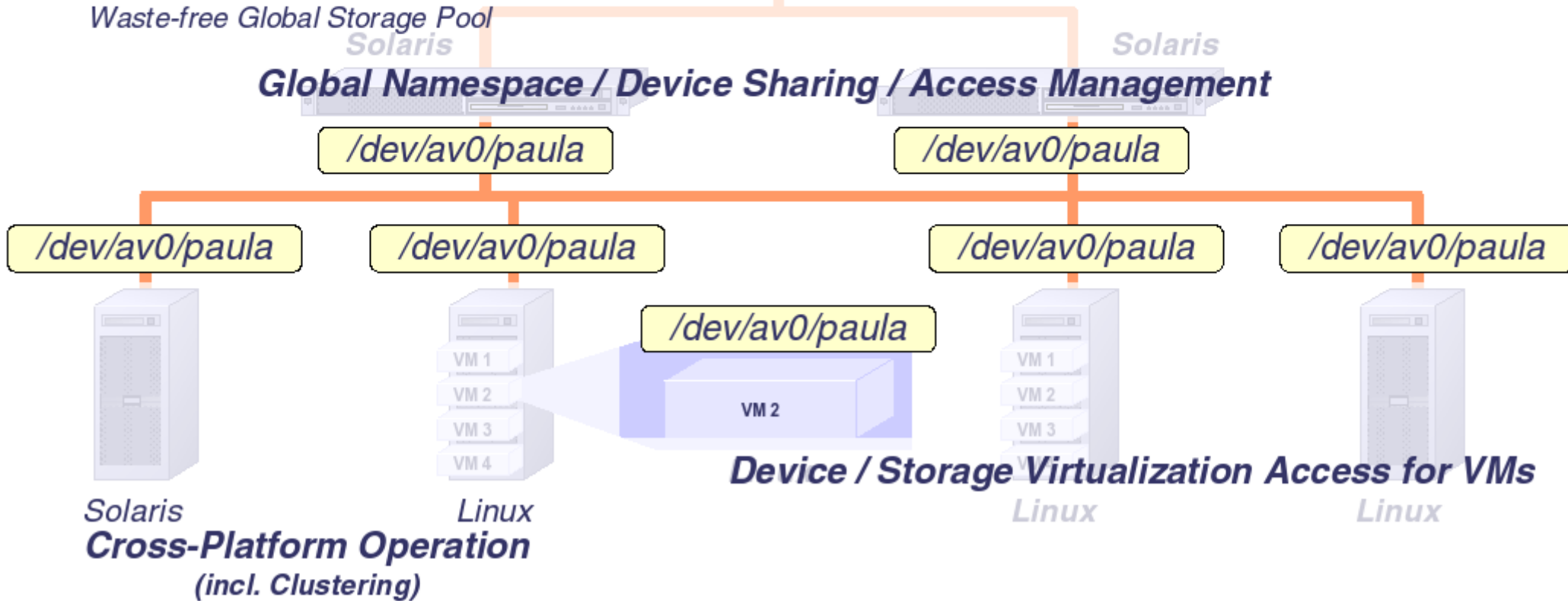
Was bringt die Verknüpfung mit einer clusterfähigen Speichervirtualisierung?

Performance / Data Protection / Data Management

- Virtual Partitions
- Concatenation
- Striping
- Mirroring
- Data Mobility
- Ease of Use
- Hardware Abstraction
- Custom Device Names
- Waste-free Global Storage Pool



Global Namespace / Device Sharing / Access Management



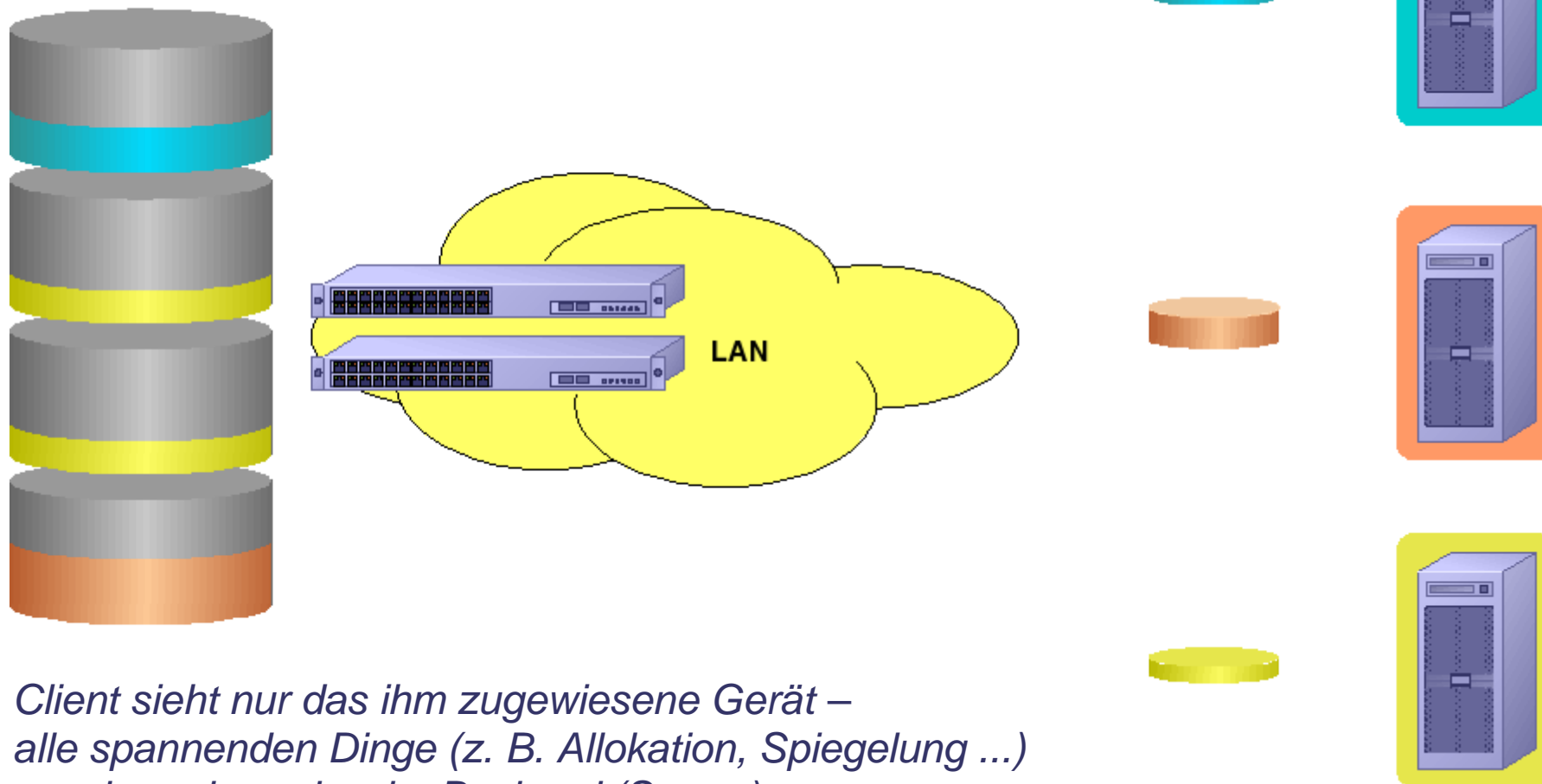
Zusammenfassung

Was ist anders mit RSIO?

- *Berücksichtigung der Netzwerkarchitektur*
- *Geräte entstehen aus dem IP-Interface heraus*
- *Verbindungen und Geräte sind administrativ getrennt*
- *Multipath- und Trunkingfunktionen integriert*
- *Verknüpfung mit Speichervirtualisierung (Gerätenamen, kein fdisk, Spiegelung, Clonen ...)*
- *Speicher überall da, wo IP ist*
- *überzeugendes Gerätekonzept auch für Linux*

RSIO mit und ohne Storage Cluster

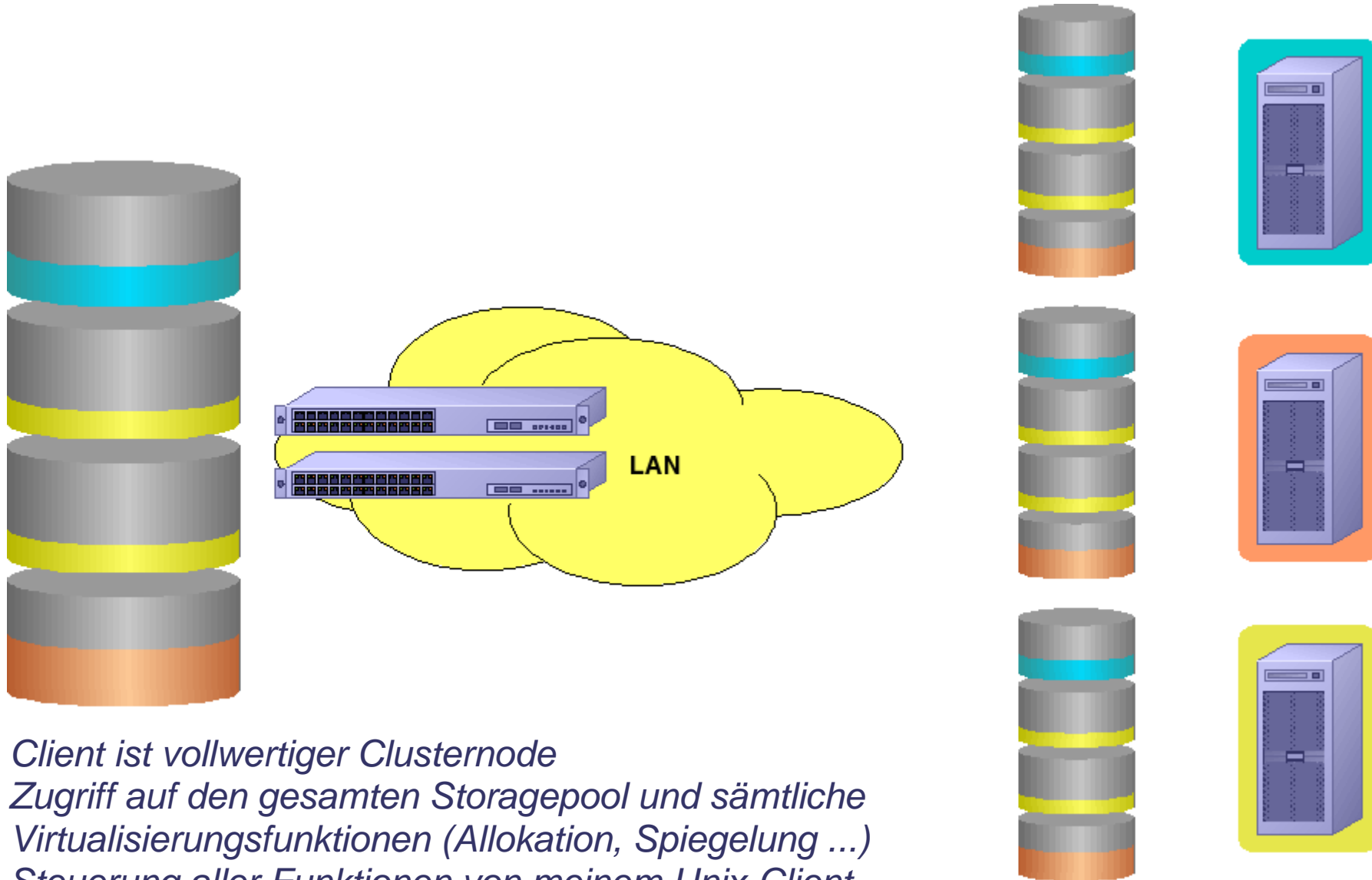
Zwei grundsätzliche Betriebsarten - 1. Einfacher RSIO-Client



*Client sieht nur das ihm zugewiesene Gerät –
alle spannenden Dinge (z. B. Allokation, Spiegelung ...)
passieren irgendwo im Backend (Server)*

RSIO mit und ohne Storage Cluster

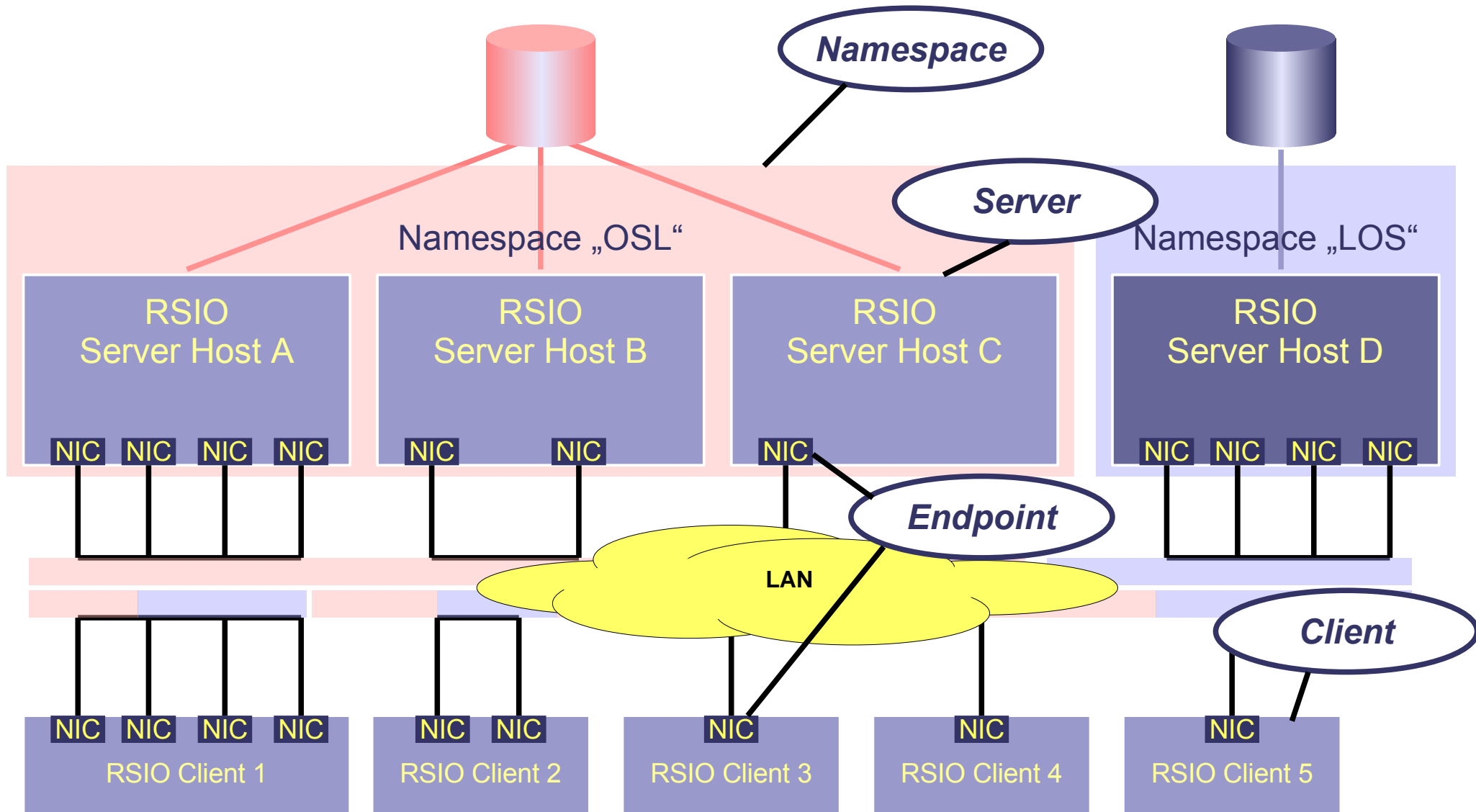
Zwei grundsätzliche Betriebsarten - 2. Betriebsart Storage Cluster



*Client ist vollwertiger Clusternode
Zugriff auf den gesamten Storagepool und sämtliche
Virtualisierungsfunktionen (Allokation, Spiegelung ...)
Steuerung aller Funktionen von meinem Unix-Client*

RSIO – Architektur im Überblick

Klar gegliedertes und flexibles administratives Konzept



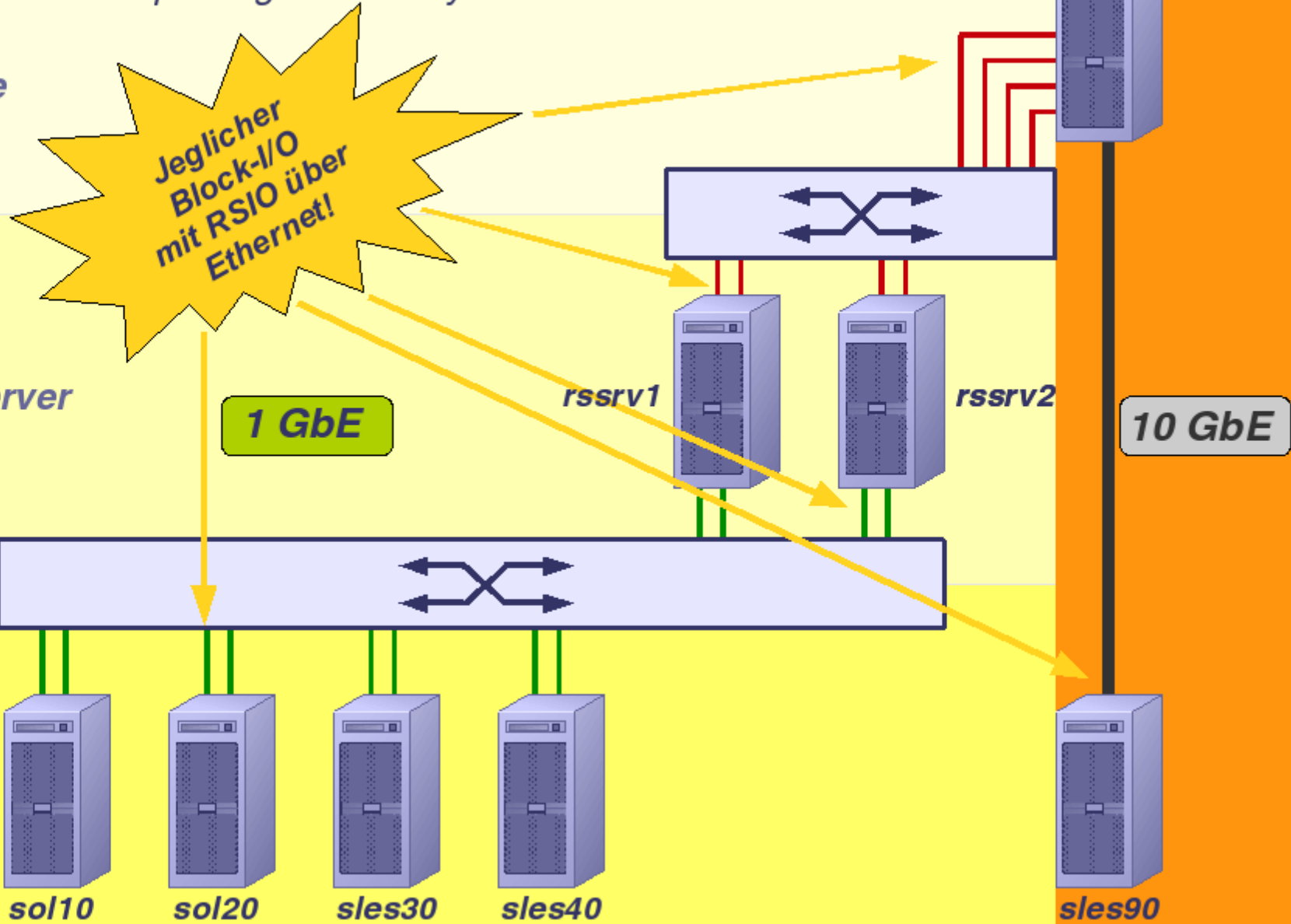
Ein Blick auf die Praxis

Die Vorführkonfiguration in praxisgerechter Systematik

Backend Storage

RSIO Storage Server

RSIO Clients



Tutorial

Was ist zur Speichieranbindung mit RSIO zu tun (praktische Vorführung)



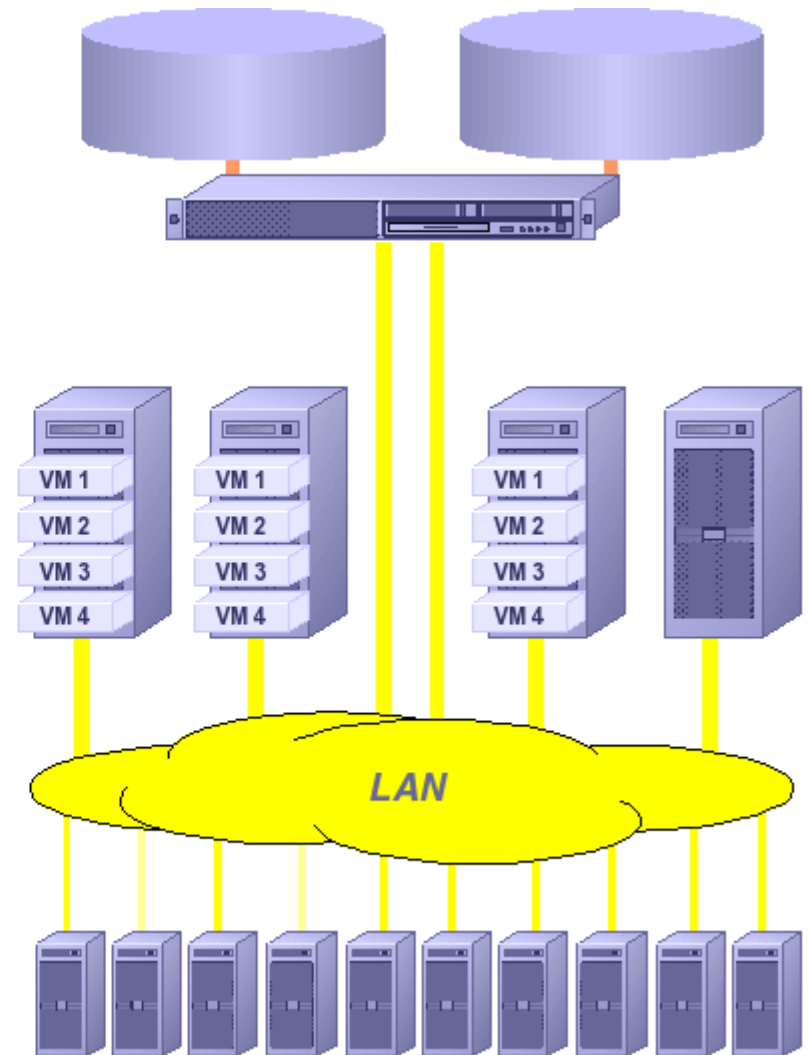
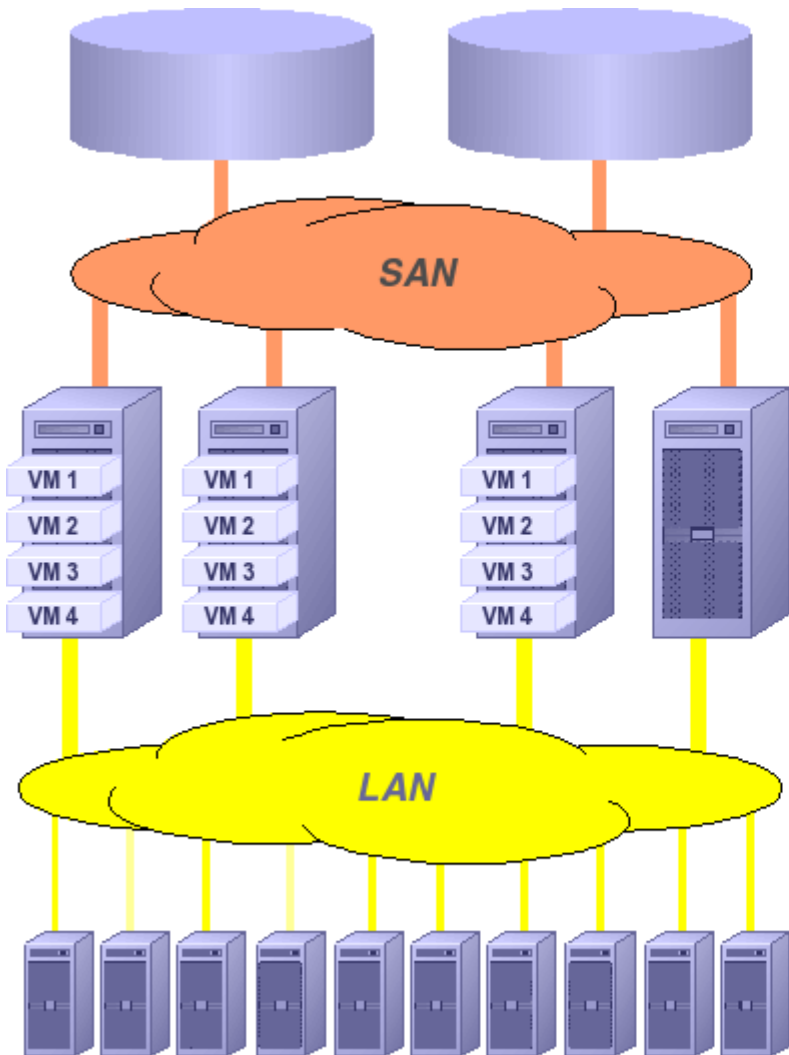
- *Aufbau der physikalischen Infrastruktur*
- *Installation RSIO-Software*
- *Konfiguration:* - *Zuweisen der zu benutzenden Interfaces*
(Menüsystem) - *Auswahl des Namespaces*
- *Auswahl der angebotenen Server-Interfaces*
- *Registrierung des Clients auf dem Server (Schlüssel)*
- *Volumes attachen (automatisch) und benutzen*
- ***fertig in 5 Minuten!*** (inkl. Multipathing, Trunking ...)

Bei Bedarf:

- *Installation OSL Storage Cluster Base*
- *Freischaltung des SC-Modes für den Client auf dem Server*
- *damit voller Zugriff auf die Speichervirtualisierung vom Client aus!*

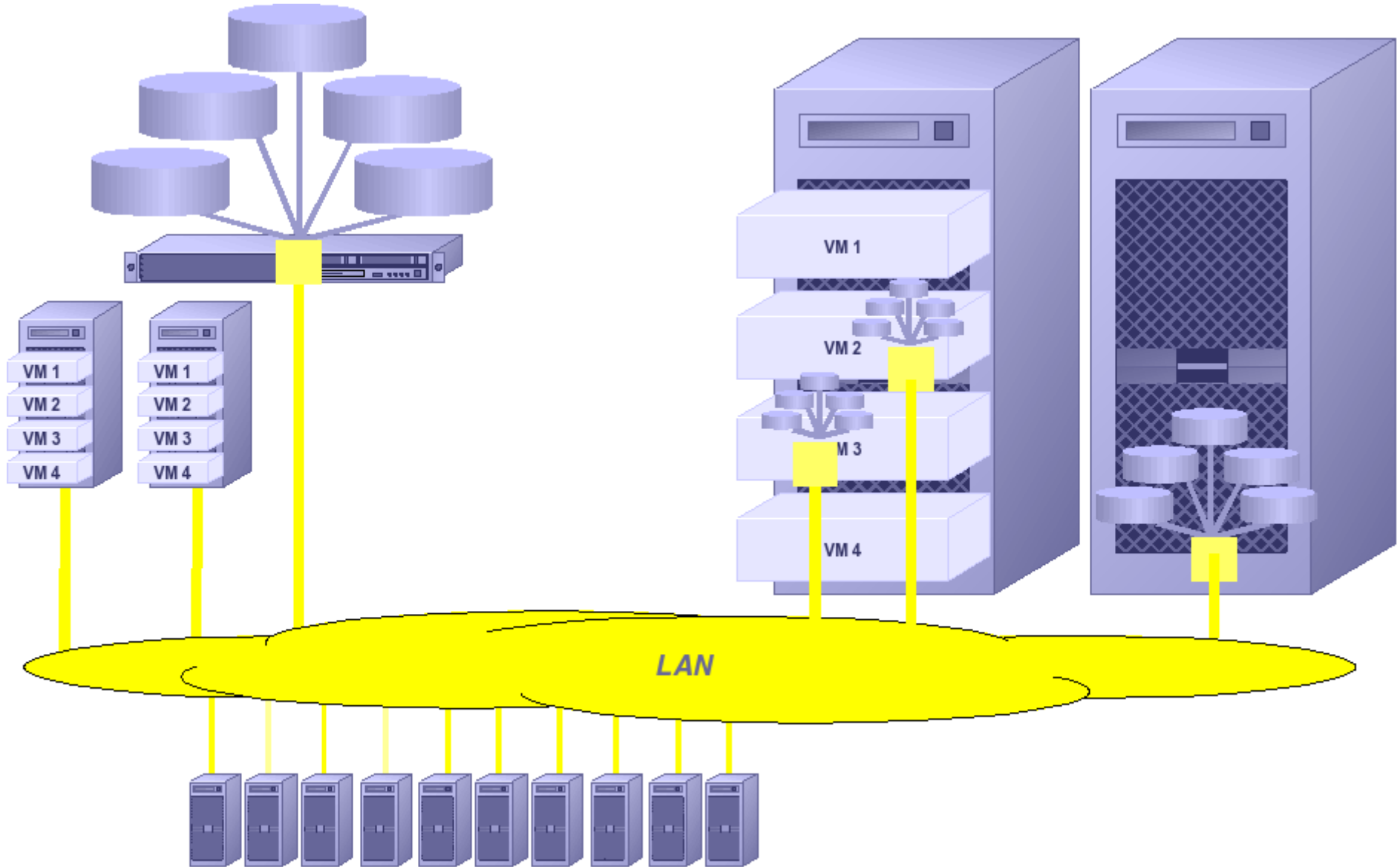
Was folgt daraus? Einsatzkonzepte

Erste Möglichkeit: Eine Netzwerktechnologie statt zweier!



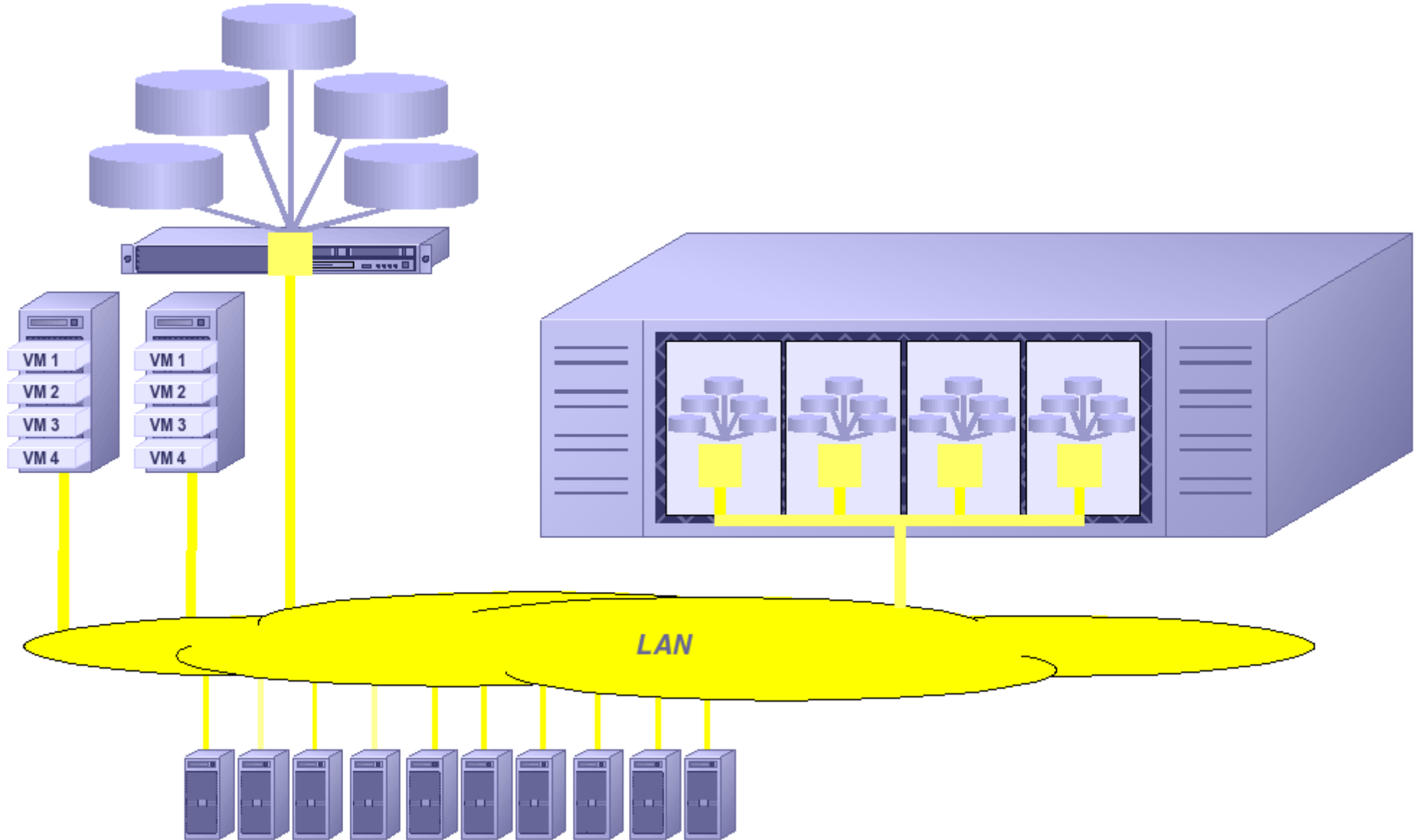
Im Detail

Überall wo IP ist: Storage in 5 Minuten!



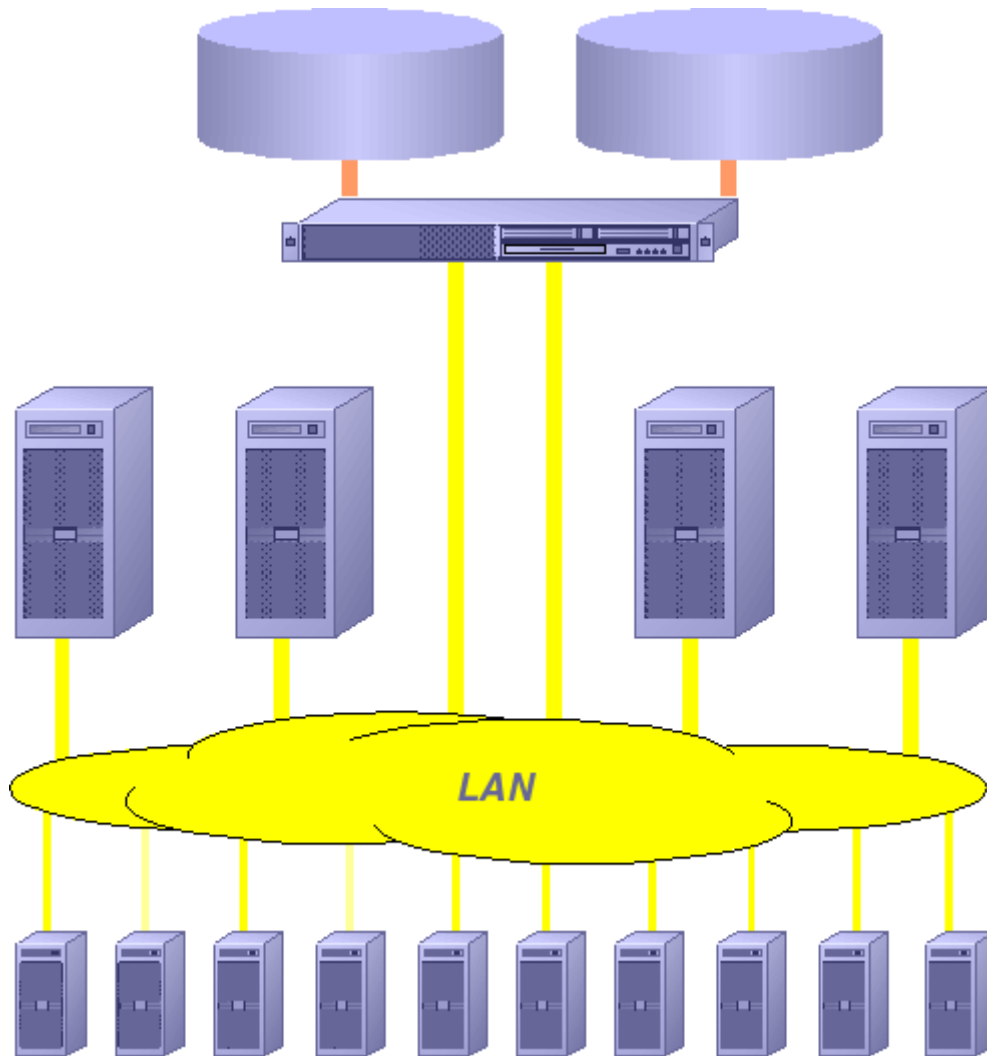
Das gilt natürlich auch für Blade-Server

Flexible Softwarelösung anstelle proprietärer Hard- und Firmware



Ein einfaches RZ-Szenario

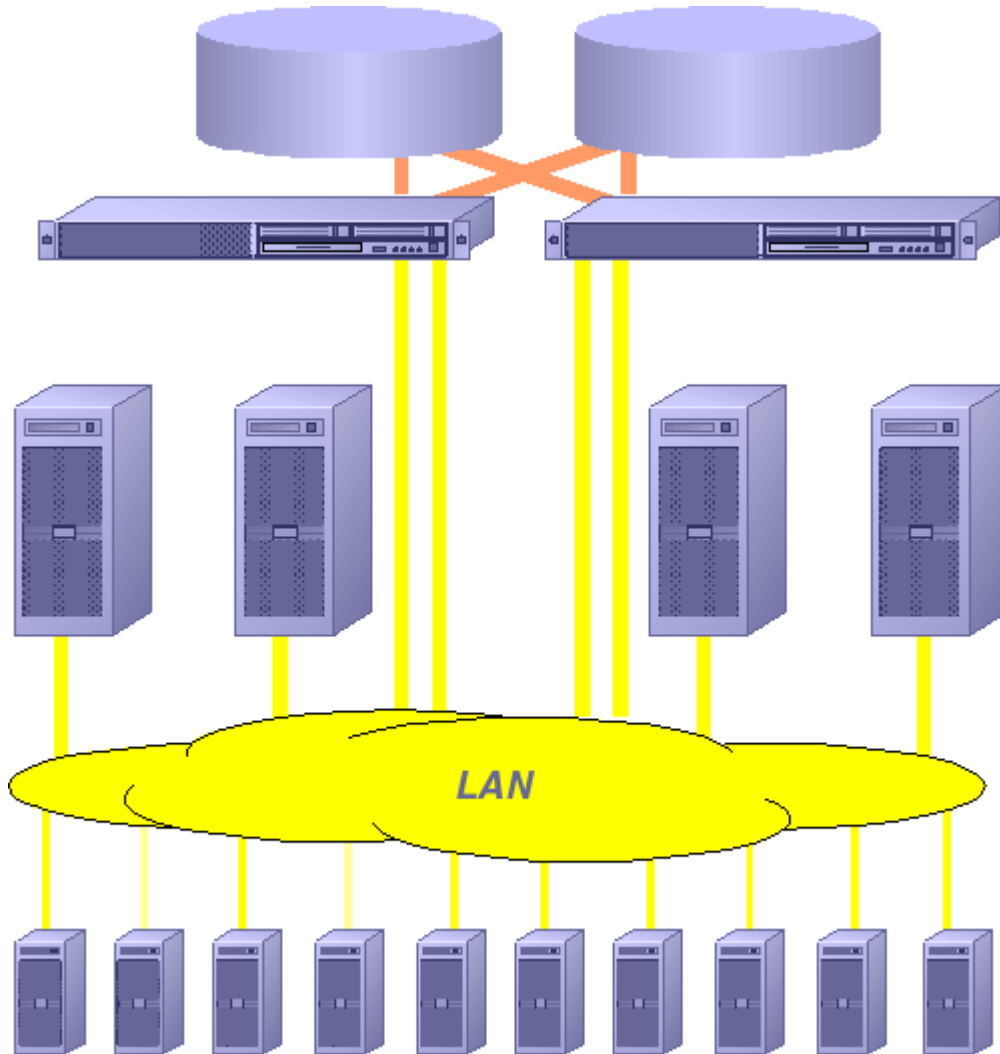
Einbeziehung von FC-Devices - RSIO-Server als LAN-to-FC-Gateway



- *bestmögliches Fanout-Verhältnis für die FC-Ports der RAID-Systeme*
- *preiswerteste und trotzdem performante und zuverlässige Storage-Anbindung*

Für höhere Ansprüche

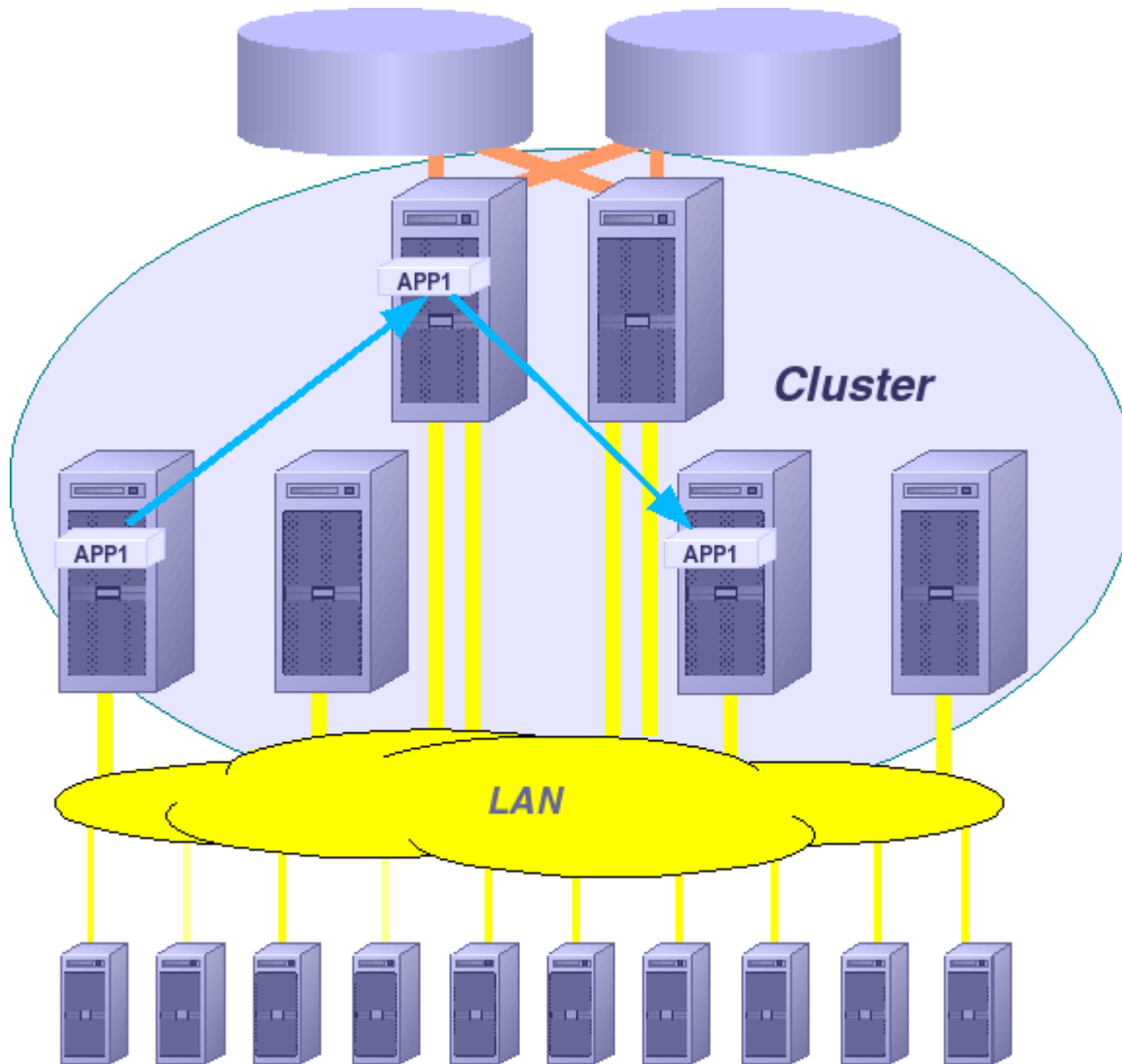
Geclusterte Storage-Server – Skalierbar in Redundanz und Durchsatz



- Multipathing für die Clients über mehrere Storage-Server
- Bei nicht zu vielen Storage-Servern weiterhin Verzicht auf FC-Switches möglich

Ausbau zum multifunktionalen Storage-Cluster

Storage-Server und Storage-Clients in einem Cluster

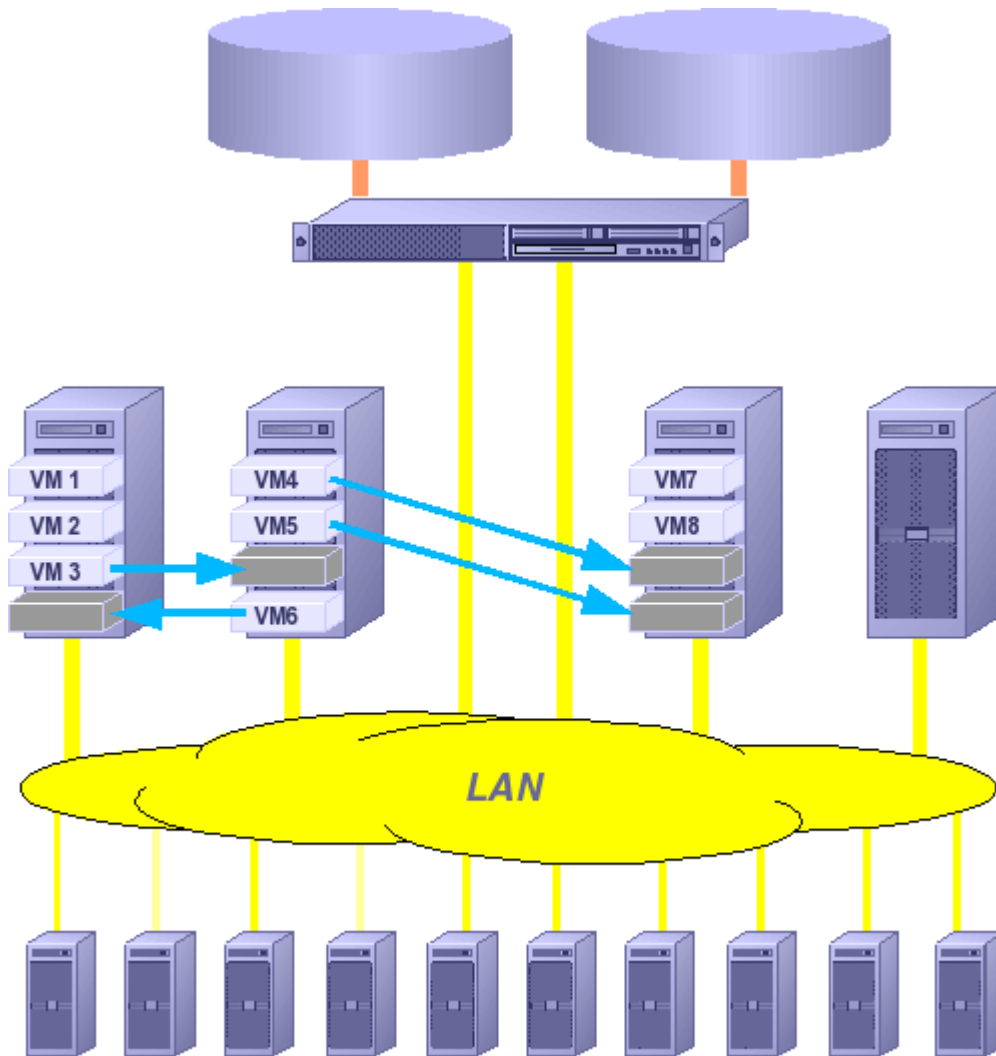


Mit Installation der Storage-Cluster-Pakete lässt sich ein RSIO-Client zum vollwertigen Cluster-Mitglied machen, d.h.:

- Storage-/Device-Allokation vom Client aus
- Spiegeloperationen vom Client aus
- automatisches Access-Management
- identische Gerätesicht auf Storage-Servern und -Clients
- optionale HV für Applikationen (Applikationen auf jedem Clusternode lauffähig, gleich ob Storage-Server oder -Client)

Storage Cluster, RSIO und KVM

Einheitliche Netzwerk-Infrastruktur, einfaches Handling



In Verbindung z.B. mit KVM erhalte ich einen VM-Cluster:

- Storage, LAN, Live-Migration etc. über die gleiche Infrastruktur: Ethernet
- Hochverfügbarkeit
- kein Trennung von Storage- und Control-Verbindungen
- kein Split Brain
- zentrale Administration möglich
- Alle Funktionen der Speichervirtualisierung für VM-Abbilder nutzbar
- via IP Zugriff auf die gesamte Storage-Welt einschl. Storage-Virtualisierung aus der VM heraus installierbar / konfigurierbar

Noch einmal: Zusammenfassung

Was ist anders mit RSIO?

- *Berücksichtigung der Netzwerkarchitektur*
- *Geräte entstehen aus dem IP-Interface heraus*
- *Verbindungen und Geräte sind administrativ getrennt*
- *Multipath- und Trunkingfunktionen integriert*
- *Verknüpfung mit Speichervirtualisierung (Gerätenamen, kein fdisk, Spiegelung, Clonen ...)*
- *Speicher überall da, wo IP ist*
- *überzeugendes Gerätekonzept auch für Linux*