

Welcome to ZFS

best Open Systems Day
Spring 2006

Unterföhring

Marco Kühn
best Systeme GmbH
kuehn@best.de



Agenda

- Überblick
- Daten Integrität
- Skalierbarkeit & Performance
- ZFS und Zonen
- Aussicht
- Administration - Live Demo

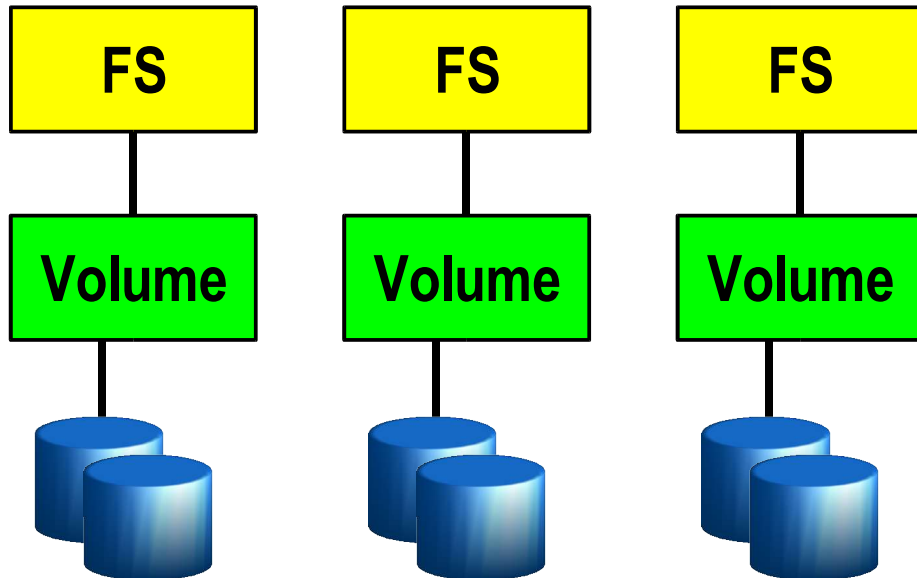
Geschichte der File Systeme

- 1984 – UFS
- 1988 - VxFS
- 1993 - ext2
- 1994 - WAFL
- 1996 - XFS
- 1998 – ReiserFS
- 2000 – JFS
- **2004 - ZFS**

FS+Volume Model vs. ZFS

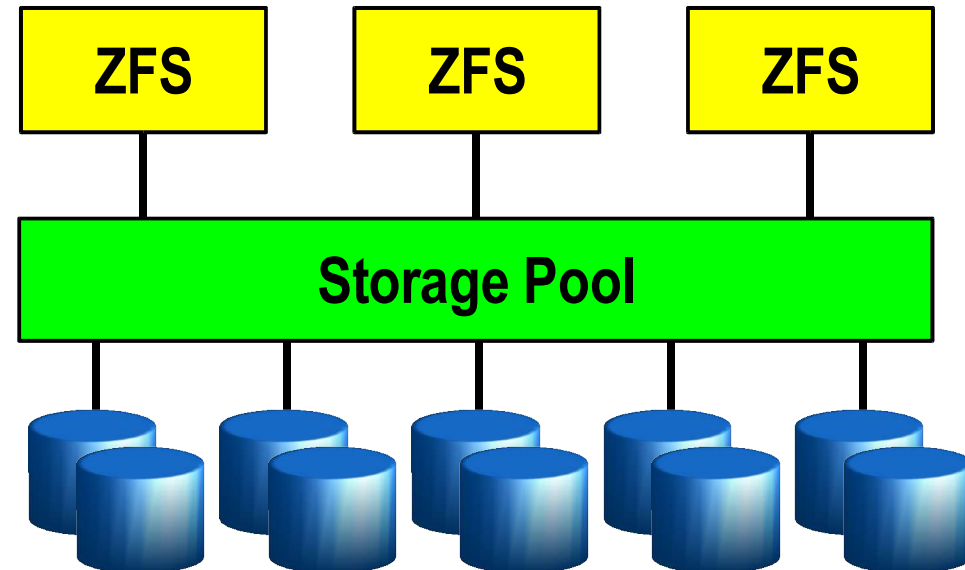
Traditionell:

- Partitions / Volumes
- Virtuelle Disks (LUNs)
- Fragmentiertes Storage



ZFS Pooled Storage:

- Keine Partitionierung
- Keine Volumes
- Shared Storage Pool



ZFS Features

- SnapShots – Read-only PIT copy of file system
- Clones – writeable copy of snapshot
- Backup / Restore (full & incrementell)
- Data Migration – Deport / Import
- Quotas
- Posix ACLs und NT-Style ACLs
- Compression
- Mount und Share ohne vfstab, dfstab
- Zones Support, ..

Agenda

- Überblick
- **Daten Integrität**
- Skalierbarkeit & Performance
- ZFS und Zonen
- Aussicht
- Administration - Live Demo

ZFS Data Integrity Model

■ Copy-on-write

- Live Daten werden nie überschrieben
- Disk Status ist immer gültig
- Kein „fsck“ File System Check notwendig

■ Transaktionen

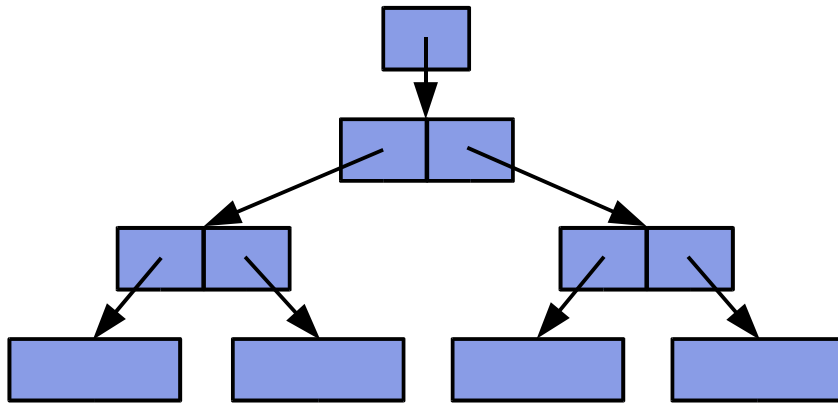
- Zusammengehörige Änderungen haben Erfolg oder schlagen als ganzes Fehl → Transaktionsklammer
- Kein Journaling notwendig

■ Prüfsummen

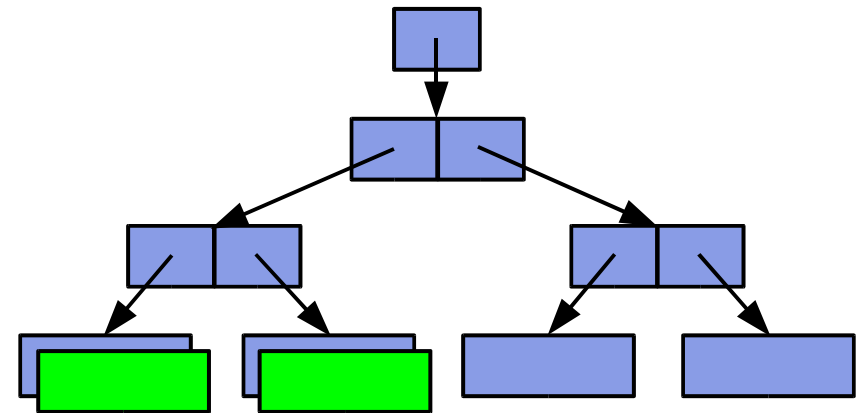
- Jeder Block ist „checksummed“
- Kein schleichende Daten Korruption

Copy-on-write Transaktionen

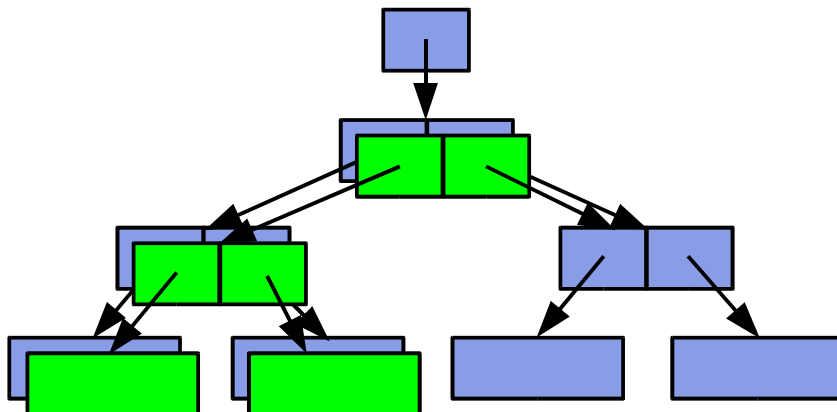
1. Initialer Block Baum



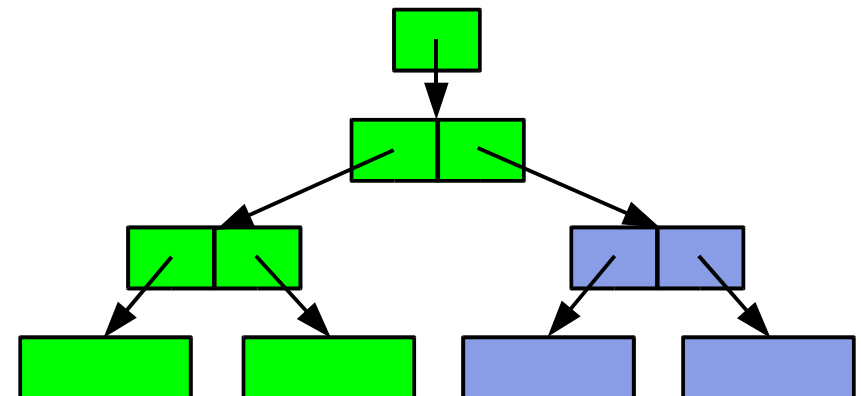
2. COW einige Blöcke



3. COW indirekte Blöcke

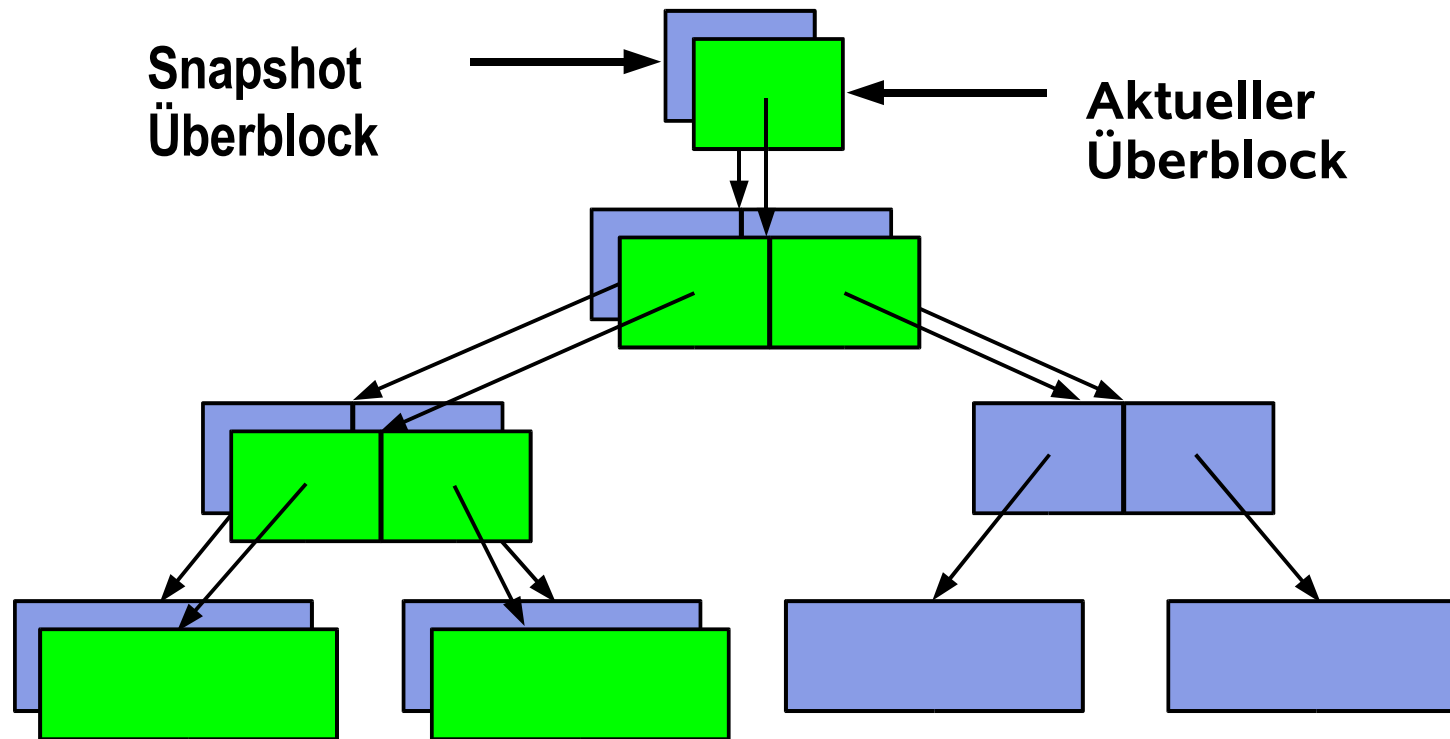


4. Rewrite Überblock (atomic)



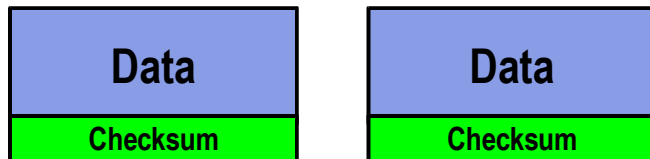
COW Bonus: SnapShots

- Nach Transaktion, COWed Blocks behalten
- „Günstiger“ SnapShots zu machen als es nicht zu tun !



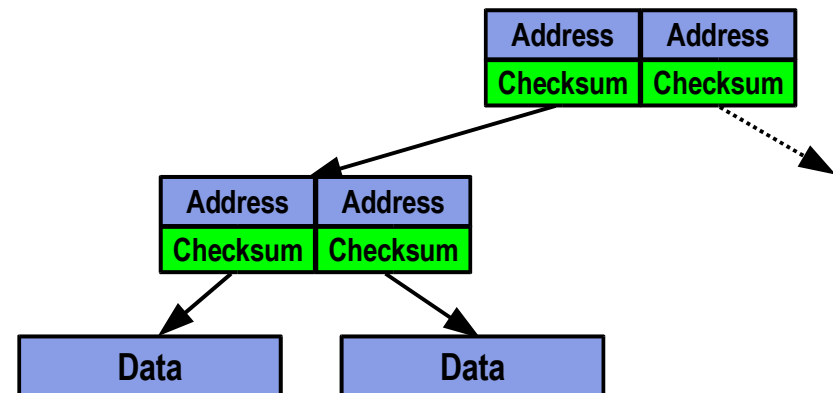
Disk Block Checksums

- Prüfsummen zusammen mit Daten Blöcken
- Kann keine „verirrten“ Writes erkennen
- Nur Media Fehler werden erkannt



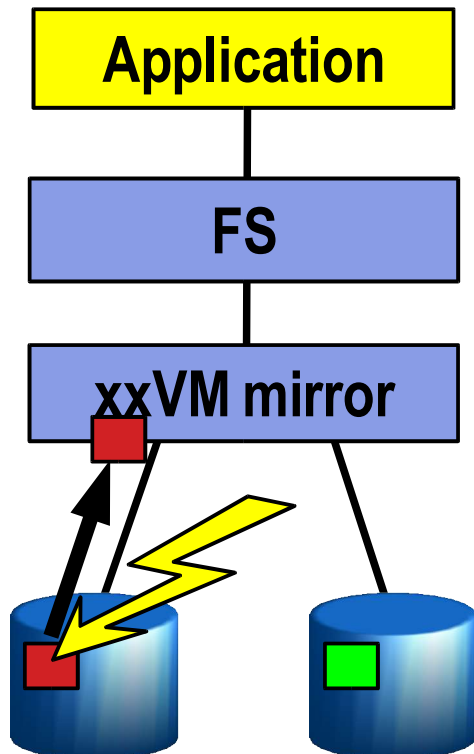
ZFS Data Checksum Trees

- Prüfsummen liegen im „Parent Block“
- Self Validation über den ganzen Block Baum

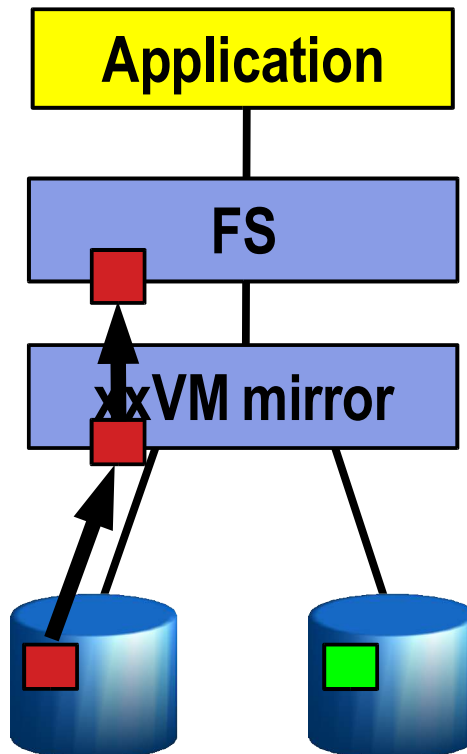


Traditionelles Spiegeln

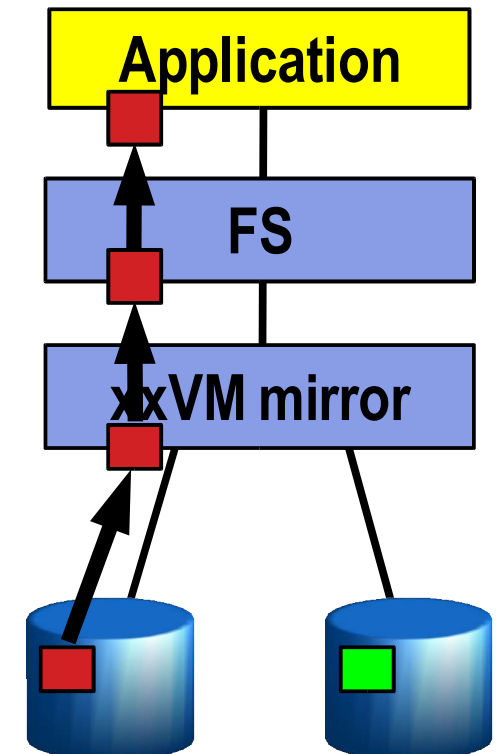
1. Applikation liest.
Submirror liefert
korrupten Block.
VM bemerkt nichts !



2. Volume Manager
gibt korrupten Block
ans File System.
Eventuell Panic ..

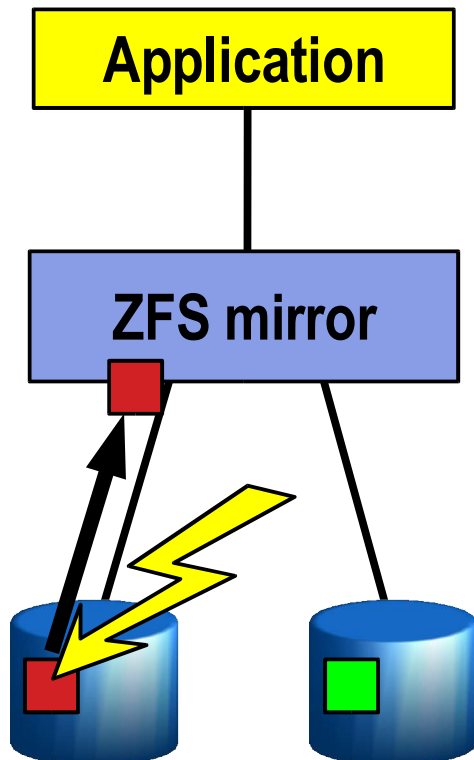


3. File System gibt
korrupte Daten an die
Applikation.

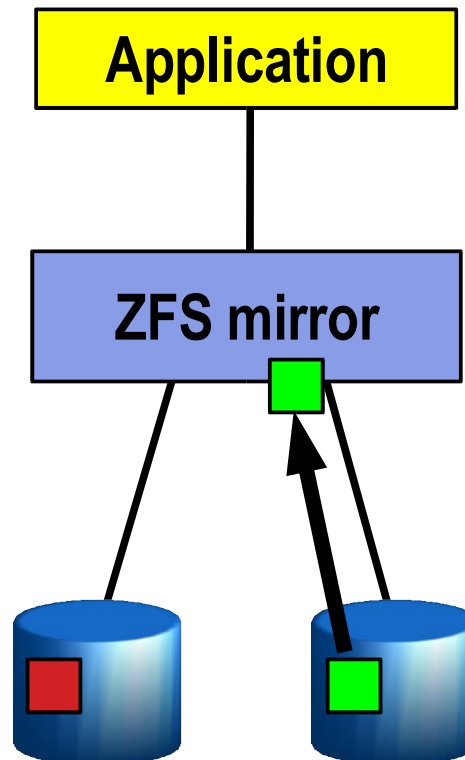


Self-Healing Data in ZFS

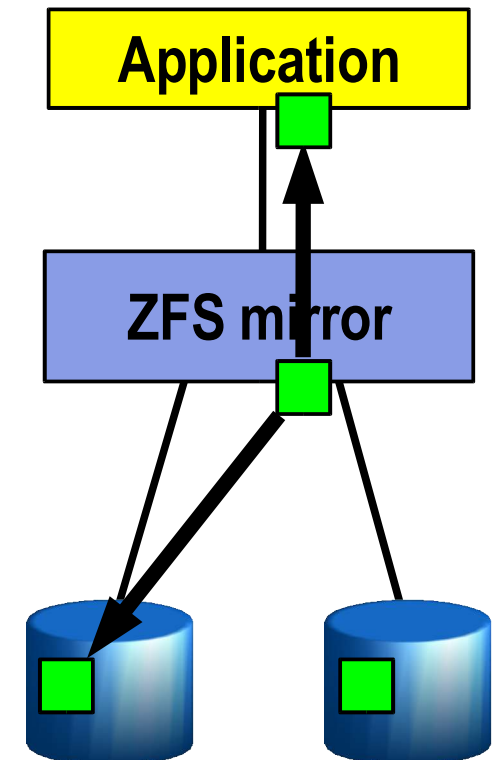
1. Applikation liest. ZFS bemerkt anhand der Checksumme den korrupten Block.



2. ZFS liest vom 2. Sub Mirror. Checksumm weist den Block als „gut“ aus.



3. ZFS reicht „gesunde“ Daten an die Applikation und korrigiert den korrupten Block.





RAID-Z

■ Dynamische Stripe Width

- Jeder logische Block ist sein eigener Stripe

 - Logischer Block = 3 Data Blocks + 1 Parity Block

- z.Z. Single-Parity; Double-Parity bereits in Entwicklung

■ Alle Schreibzugriffe sind full-stripe Schreibzugriffe

- Eliminiert Lesen-Ändern-Schreiben (Performance!)

■ Erkennt und korrigiert schleichende Datenkorruption

- Checksummen getriebene Rekonstruktion wie beim Spiegel

■ Keine HW RAID notwendig – ZFS liebt „billige“ Platten



Disk Scrubbing und Resilvering

- **Findet schlummernde Fehler, die korrigierbar sind**
 - ECC Memory Scrubbing für Platten
- **Alle Daten werden gelesen und ggf. korrigiert**
 - Mirror: Alle Spiegelhälften
 - RAID-Z: Daten- und Parity-Blöcke
- **Schnelles und sicheres Resilvering**
 - Traditionell: komplette Platte, d.h. alle Blöcke; keine Validierung
 - ZFS resilver: nur „live-data“; Checksummen Validierung

Agenda

- Überblick
- Daten Integrität
- **Skalierbarkeit & Performance**
- ZFS und Zonen
- Aussicht
- Administration - Live Demo

ZFS Skalierbarkeit

■ Immense Kapazität

- Tera Byte: 2^{40}

- Peta Byte: 2^{50}

- ...

- ZFS: 2^{128}

■ 100% dynamische Metadaten

- Keine Limits für:

- Dateigrößen (z.Z: Posix Standard 2^{64})

- Directories

- Inodes, ..

ZFS Performance

■ Copy-on-write Design

- Kehrt „random Writes“ in „sequential Writes“ um

■ Dynamisches Striping über alle Platten

- Maximiert den Durchsatz

■ Multiple Blöckgrößen

- Automatisches Anpassen nach Datenaufkommen

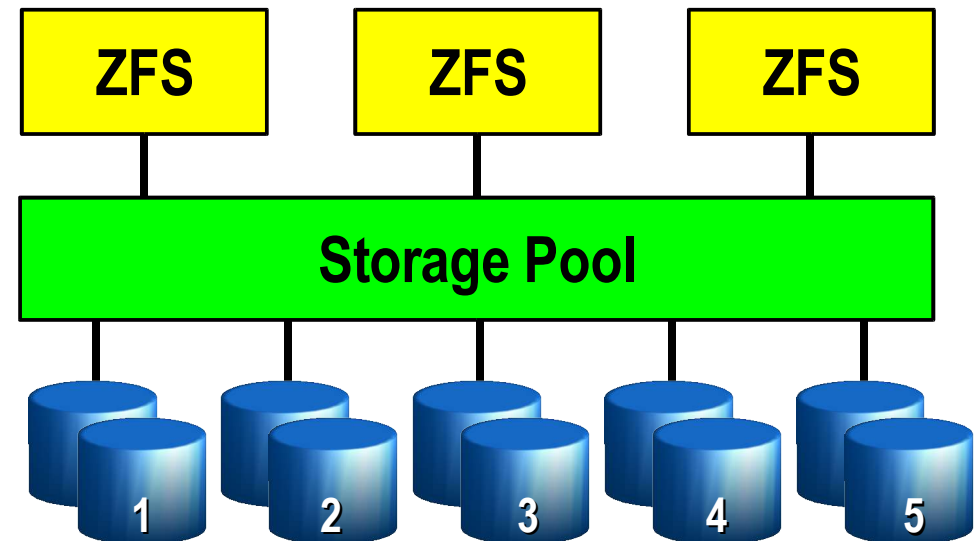
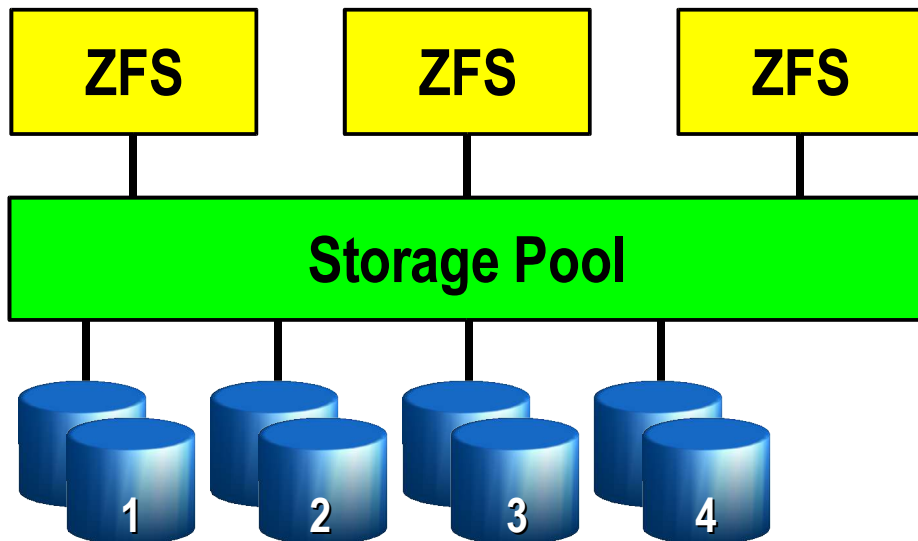
■ Pipelined I/O

- Maximale I/O Parallellisierung, Status, Sortierung, Aggregation, ..

■ Intelligentes Prefetching

Dynamic Striping

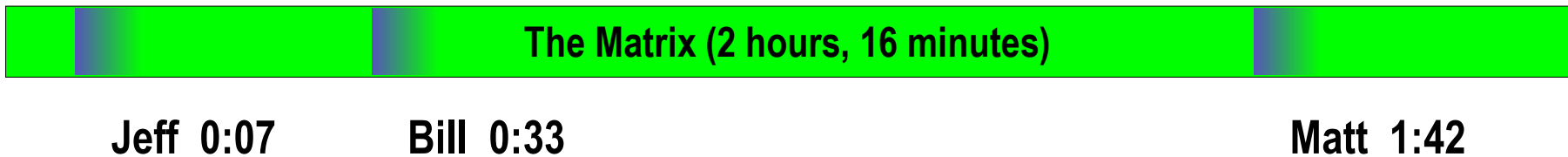
- Verteilt automatisch die Last über alle Platten
- Block Allocation Policy berücksichtigt:
 - Kapazität
 - Performance
 - Health (degraded Mirror)
- „alte“ Daten über 4 Platten verteilt
- „neue“ Daten über 5 Platten verteilt
- COW reallokiert sanft „alte“ Daten



Intelligent Prefetch

- Mehrere unabhängige „prefetch streams“

- Multi User Prefetch



- Automatische Erkennung der Länge und des Strides

- Ideal für HPC

- ZFS erkennt lineare Zugriffsmuster

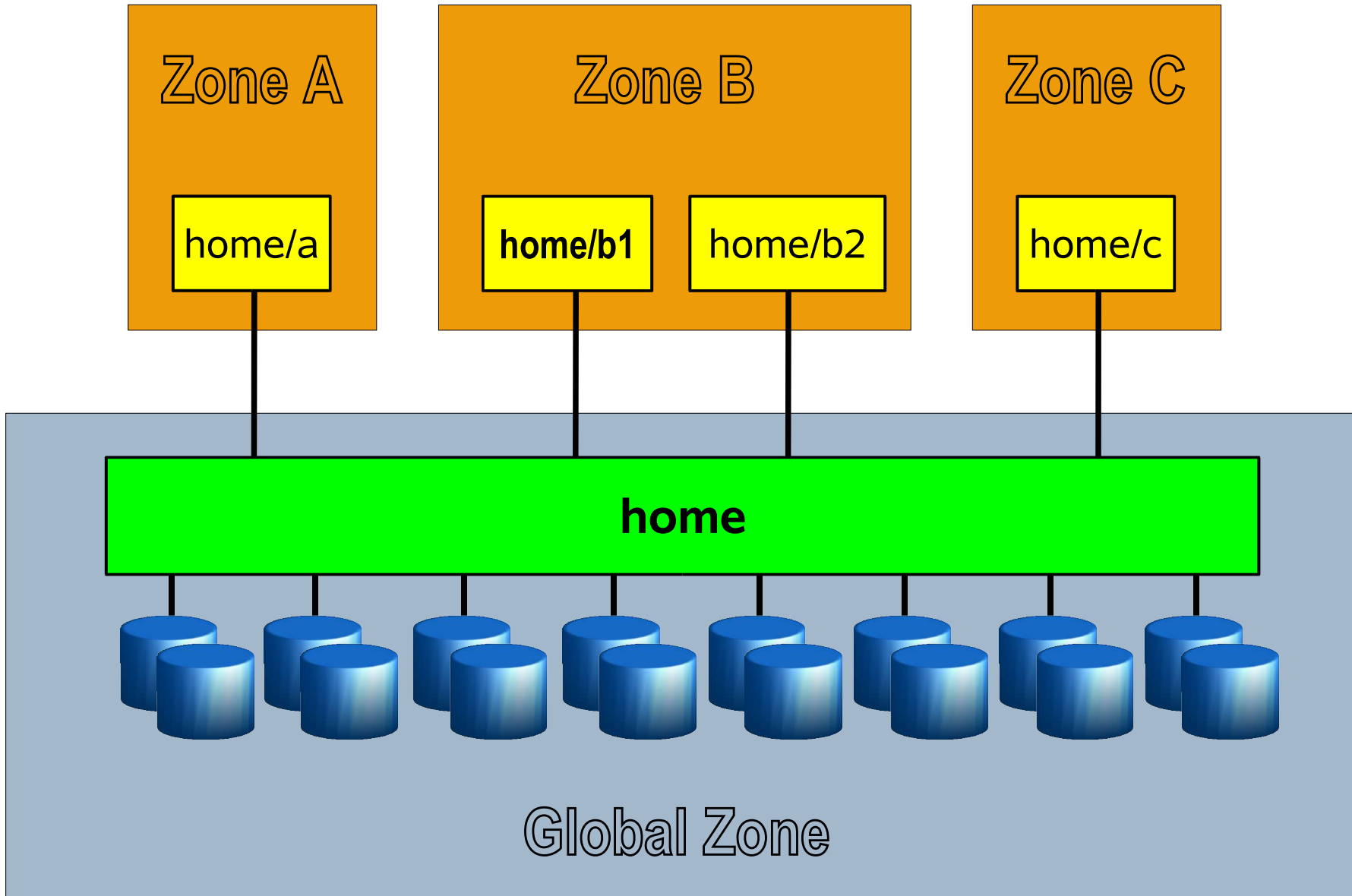
- Vorwärts als auch Rückwärts

Agenda

- Überblick
- Daten Integrität
- Skalierbarkeit & Performance
- **ZFS und Zonen**
- Aussicht
- Administration - Live Demo

- **ZFS und Zonen Virtualisierung passen perfekt zusammen**
 - Pool: physikalische Resource in der globalen Zone
 - DataSet: logische Resource in einer lokalen Zone
- **Präzise Kontrolle über DataSets**
 - Gemäß „add dataset“ in zonecfg (1M)
- **Strong Security Model**
 - Lokale Zonen sehen keine physikalischen Platten
- **Delegierte Administration**
 - Jede Zone kann Quotas setzen, SnapShots erzeugen, etc.
- **SnapShot und Cloning beschleunigen das Erstellen von Zonen**

ZFS und Zones



Agenda

- Überblick
- Daten Integrität
- Skalierbarkeit & Performance
- ZFS und Zonen
- **Aussicht**
- Administration - Live Demo

- **Ab 1. Juni 2006 in Solaris 10 Update2**
- **Datensicherheit**
 - Encryption
- **Boot**
 - Booten von jeglichen „Datasets“
 - Mehrere Boot Environments, von GRUB gemanaged
- **Einfaches Patchen**
 - SnapShot erzeugen, Patch einspielen, evtl. Rollback, ..
- **Live Upgrade**
 - Clone erzeugen, Upgrade durchführen, Booten vom Clone
 - Keine extra Partitionen

Agenda

- Überblick
- Daten Integrität
- Skalierbarkeit & Performance
- ZFS und Zonen
- Aussicht
- **Administration - Live Demo**

Fragen ?