Lustre

Julius Plehn

Arbeitsbereich Wissenschaftliches Rechnen
Fachbereich Informatik
Fakultät für Mathematik, Informatik und Naturwissenschaften
Universität Hamburg

01.06.2016

Gliederung (Agenda)

Grenzen herkömmlicher Technologien

- Natürliche Begrenzungen eines einzelnen Servers
  - Speicher:
    - HDD: ~200 MB/s
    - SSD: ~500 MB/s
    - Idealfall: Limitierung durch Schnittstelle (S-ATA, SAS: 6, 12, (22.5) GBit/s [1]
  - Anbindung:
    - Ethernet: ~40 GBit/s
    - Infiniband: ~100 GBit/s

Problemlösung: Lustre

- Verteiltes, paralleles Dateisystem
- Beschränkungen eines einzelnen Systems entfallen
- Bandbreite: 2.5 TB/s (real)
- >55 PB Speicher (max. 512 PB)
- POSIX Interface
- Von 60 der schnellsten 100 verwendet (TOP500)
- Open Source (GPLv2)

Komponenten

- Metadata Target (MDT):
  - Verwaltet Metadaten: Dateien und Ordnernamen, Rechte, Stripes, Locks, ...
- Metadata Server (MDS):
  - Stellt diese in MDT gespeicherte Metadaten den Clients zur Verfügung
  - Nicht am eigentlichen IO beteiligt
- Object Storage Target (OST)
  - Daten werden in (mehreren) Objekten gespeichert
  - Jedes Objekt wird auf unterschiedlichen OSTs gespeichert
  - Max. Größe: 128 TB
- Object Storage Servers (OSS)
  - Verwaltet Zugriff auf bis zu 32 OST

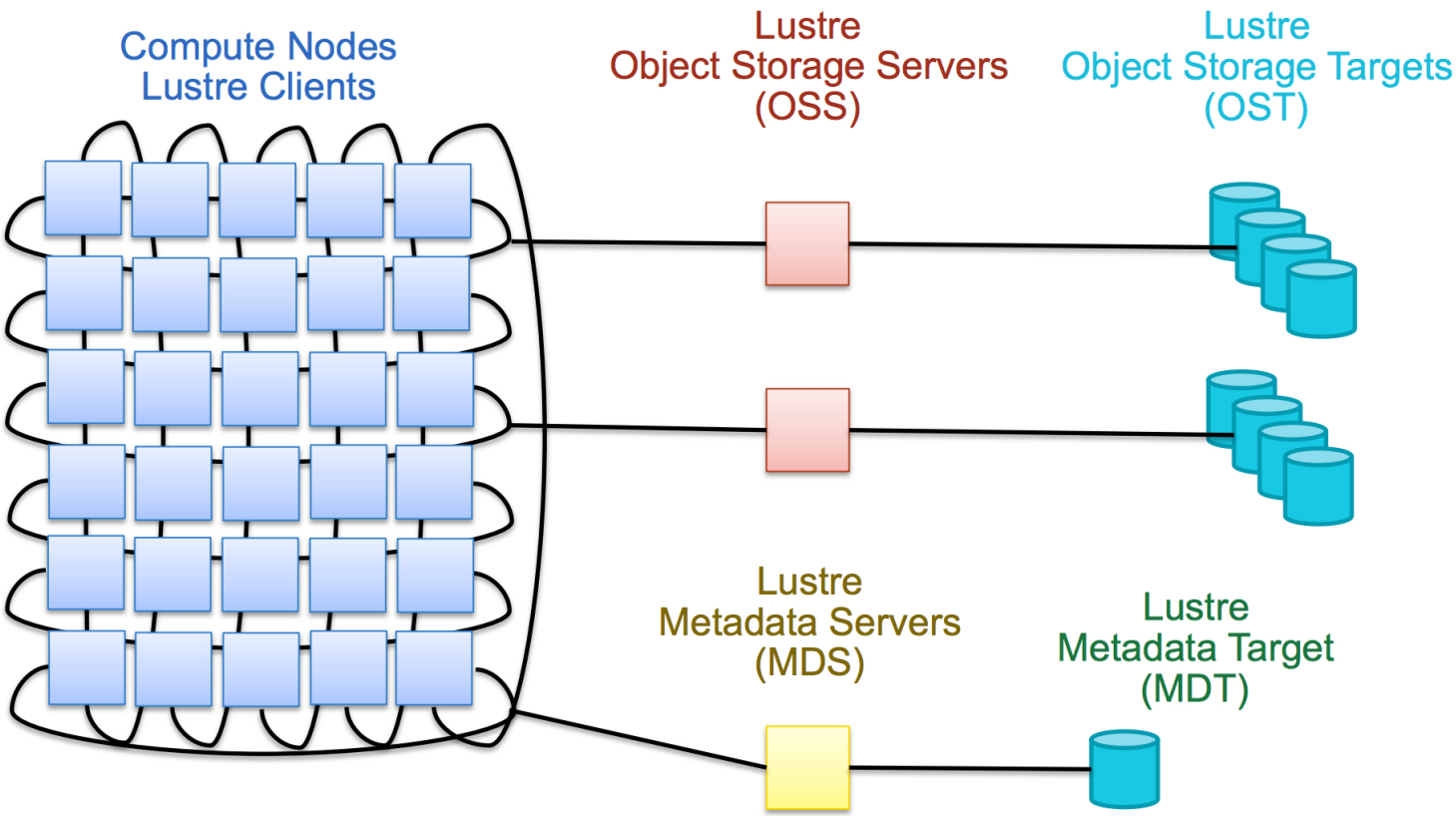

Abbildung: Architektur [2]

Frage: Was würde der Befehl ls -l in einem Lustre Verzeichnis bewirken?

ls -l: Auflistung aller Dateien zuzüglich ihrer Besitzer, Rechte, Größen, ...

Frage: Was würde der Befehl ls -l in einem Lustre Verzeichnis bewirken?

- Rechte und Besitzer: MDT
- Dateiinformationen: OST
- Für jede Datei/Verzeichnis müssen verschiedene Server kontaktiert werden

Dateisysteme

- Austauschbare Object Storage Devices (OSD)
- Abstrahieren den Zugriff auf das Dateisystem
- ldiskfs: optimiertes ext4, max. 4 Milliarden Dateien pro MDT
- zfs: neuer, integriertes RAID, größere Dateien, 256 Milliarden Dateien pro MDT, wichtige Features (Snapshots, Prüfsummen, ...)

Frage: Wie kann man eine Datei möglichst effektiv speichern?

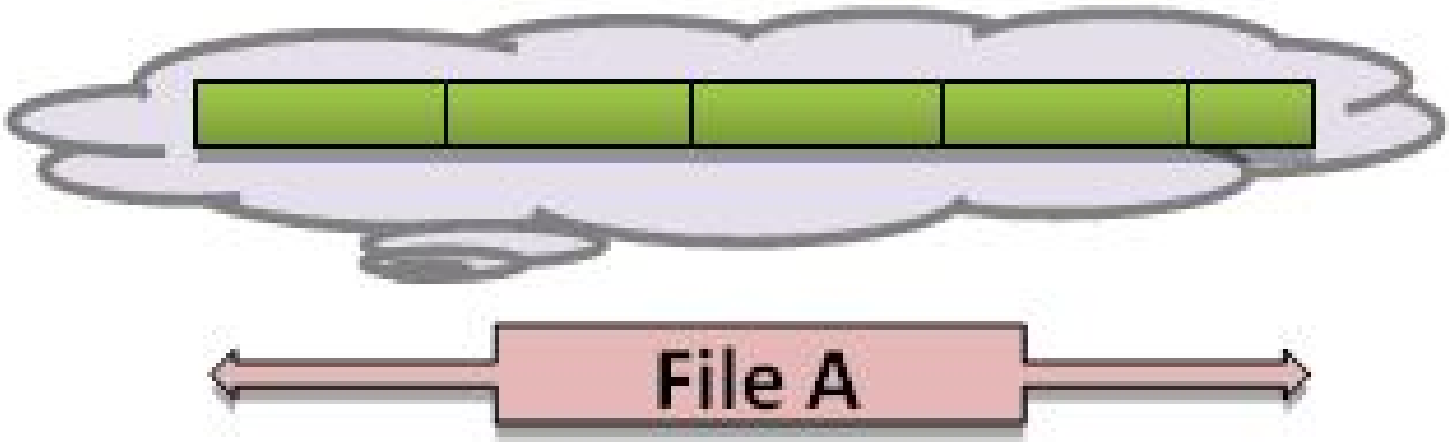

Abbildung: Dateisicht [3]

File Striping

- Verteilung der Dateien über mehrere OSTs
- Stripe Count: Anzahl über wie viele OSTs die Datei verteilt werden soll
- Stripe Size: Anzahl der Bytes pro Stripe
- Einstellung für einzelne Dateien, Ordner oder gesamtes Filesystem möglich
- Verteilung der Stripes durch Round Robin oder gewichteten Algorithmus

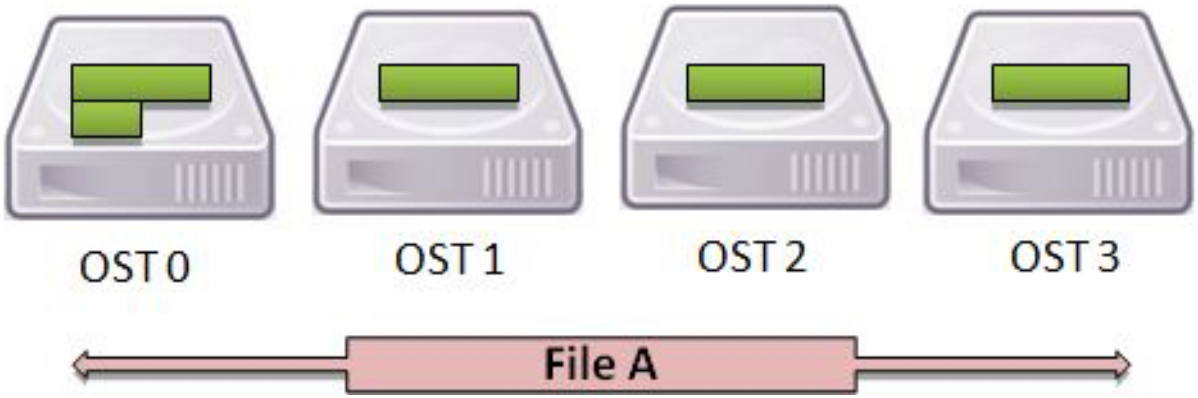

Abbildung: physische Sicht [3]

Gleichmäßige Ausrichtung

- Dateigröße: 9 MB
- Stripe Count: 4
- Stripe Size: 1 MB

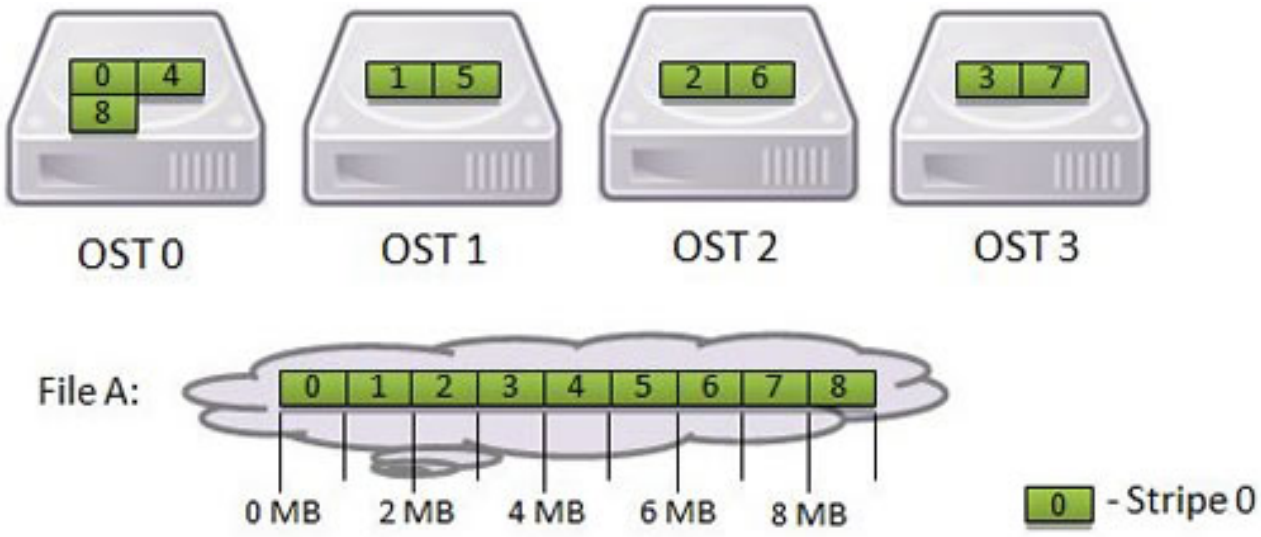

Abbildung: Aligned Stripes [3]

Ungleichmäßige Ausrichtung

- Dateigröße: 5 MB
- Stripe Count: 4
- Stripe Size: 1 MB

Prozesse schreiben unregelmäßige Daten: starke Fragmentierung

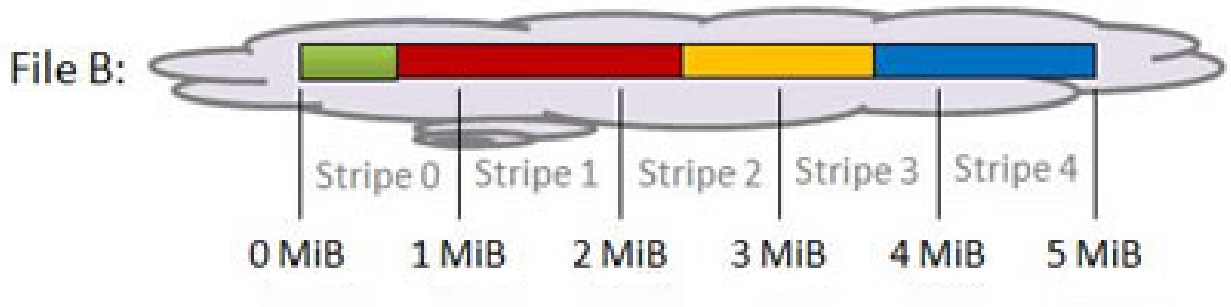

Abbildung: Datei mit verschiedenen Offsets [3]

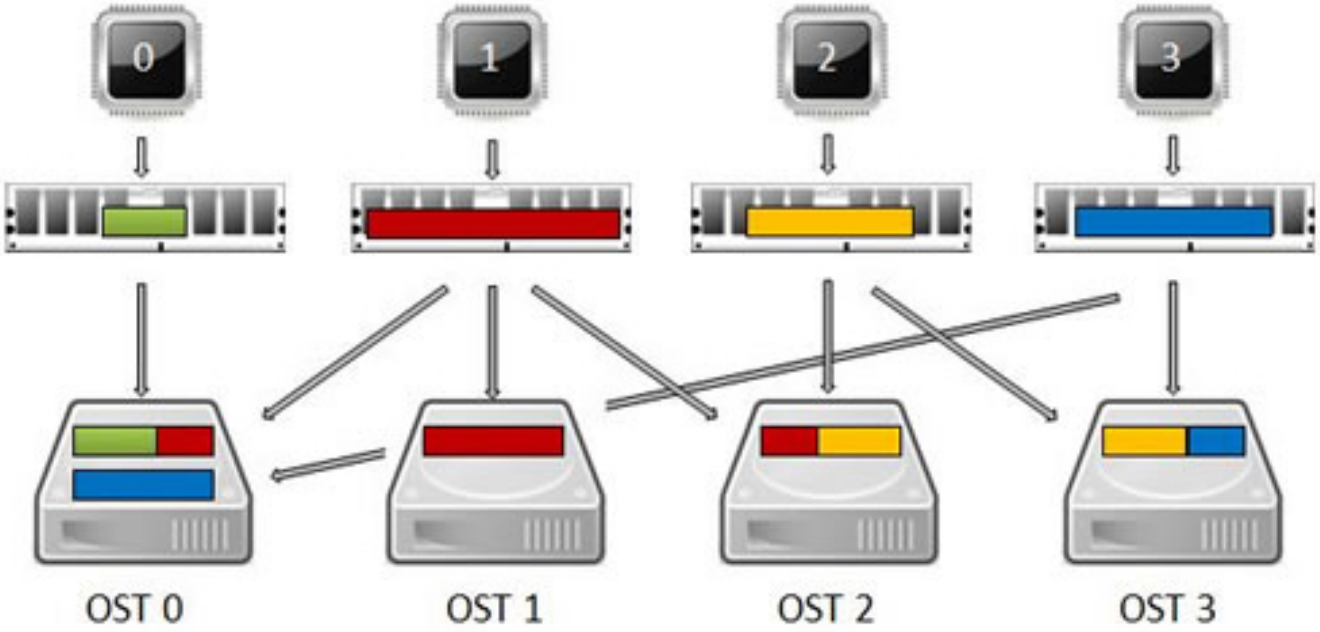

Abbildung: Non-Aligned Stripes [3]

Wahl des Size und Count Parameters

- Min. 512 KB, Versand von 1 MB Chunks
- Empfohlen: 1 MB bis 4 MB: Lock Zeiten beachten
- Bei kleinen Dateien: Stripe Count 1
- Wenn möglich: Stripe Size an Operationen des Programms anpassen (Aligned Stripes) [4]

Vorteile:

- Einzelne große Dateien können über mehrere Server verteilt werden
- Erhöhung der Bandbreite

Schwierigkeiten:

- Balance zwischen Stripe Size und Stripe Count finden

File Identifiers (FIDs)

- FIDs repräsentieren Geräte-unabhängig die klassischen Inode Kennziffern
- 128-bit Länge:

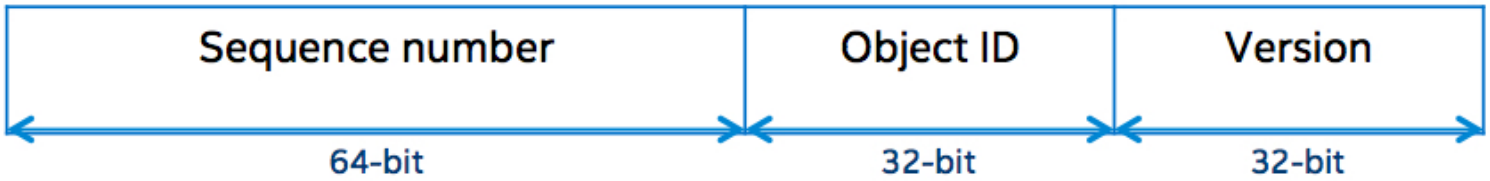

Abbildung: FID [5]

- Sequence Number: Identifizierung des MDT
- Object ID: Identifizierung des Objekts
- Version Number: Versionierung (derzeit ungenutzt)
- FID Location Database (FLDB) löst die FID zum Target auf

File Layout EA

- Nach MDT-Identifizierung durch FLDB kann auf das File Layout zugegriffen werden

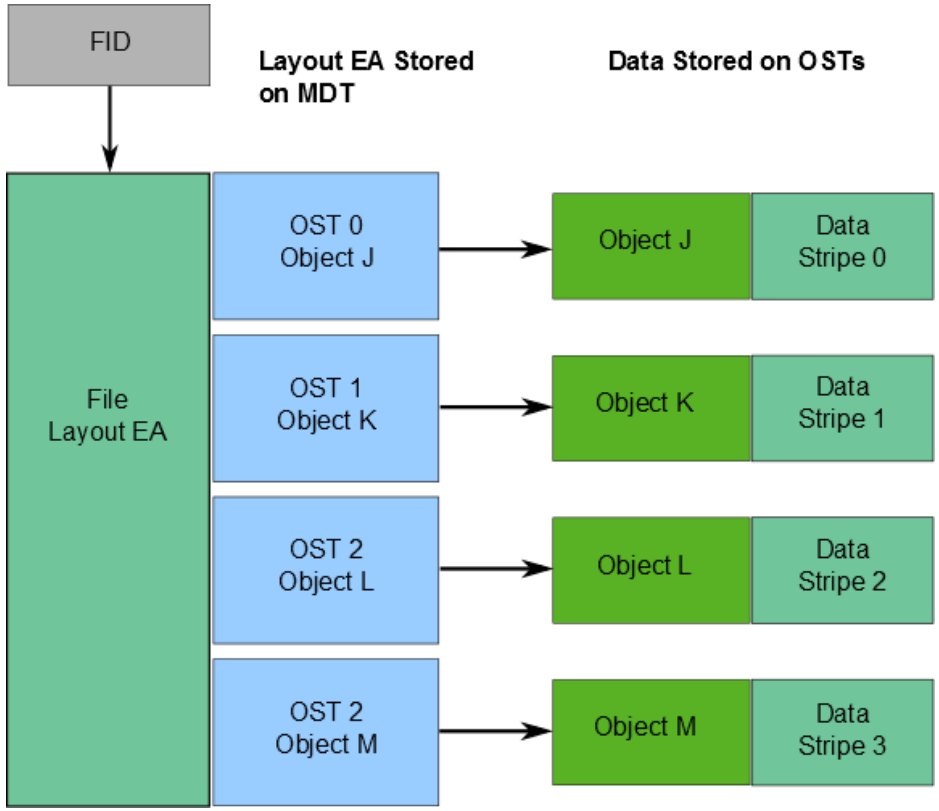

Abbildung: File Layout [6]

Locking

- Lustre distributed lock manager (LDLM) sichert Kohärenz zwischen allen Servern und Clients
- MDS LDLM: Metadata Locks für Inode Modifikation auf den MDT
- OST LDLM: Locks für Stripes in Blöcken (4096 Bytes)
- Read Locks unproblematisch, starke Parallelisierung möglich

Lustre Networking (LNET)

- Kommunikationsprotokolle werden durch LNET bereitgestellt
- Lustre Network Driver (LND) stellen die Treiber bereit
- Netzwerktypen: TCP (10 GigE, IPoIB), InfiniBand, Cray, ...
- Lustre $\rightarrow$ Network RPC API $\rightarrow$ LNET $\rightarrow$ LND $\rightarrow$ NIC Driver [7]
- Payloads: Daten lesen/schreiben, Metadaten und Locks
- Unterstützt (wenn durch NIC möglich) RDMA
- Je nach Anwendung sind die IOPS entscheidend [8]
- Latenz: [9]
  - 100 Gb/s IB: 0.5 μ s
  - Intel 40Gb/s: 1.2 μ s

RDMA

- **R**emote **D**irect **M**emory **A**ccess
- NIC kopiert Daten direkt in den RAM des Ziels
- Zero-copy: Vermeidung von Kopien zwischen Betriebssystem und Anwendung
- Datenaustausch ohne Involvierung der CPU, Caches oder Kontextwechsel

Failover

- Downtime durch Redundanz minimieren
- Active/Passiv: passiver Server übernimmt bei Problemen des Aktiven
- Active/Active: jeder Server kann bei Ausfall des Anderen dessen Aufgaben übernehmen
- Fehlererkennung von Lustre beschränkt sich auf das Dateisystem und kann durch externe Mechanismen erweitert werden (Corosync, Pacemaker, ...)

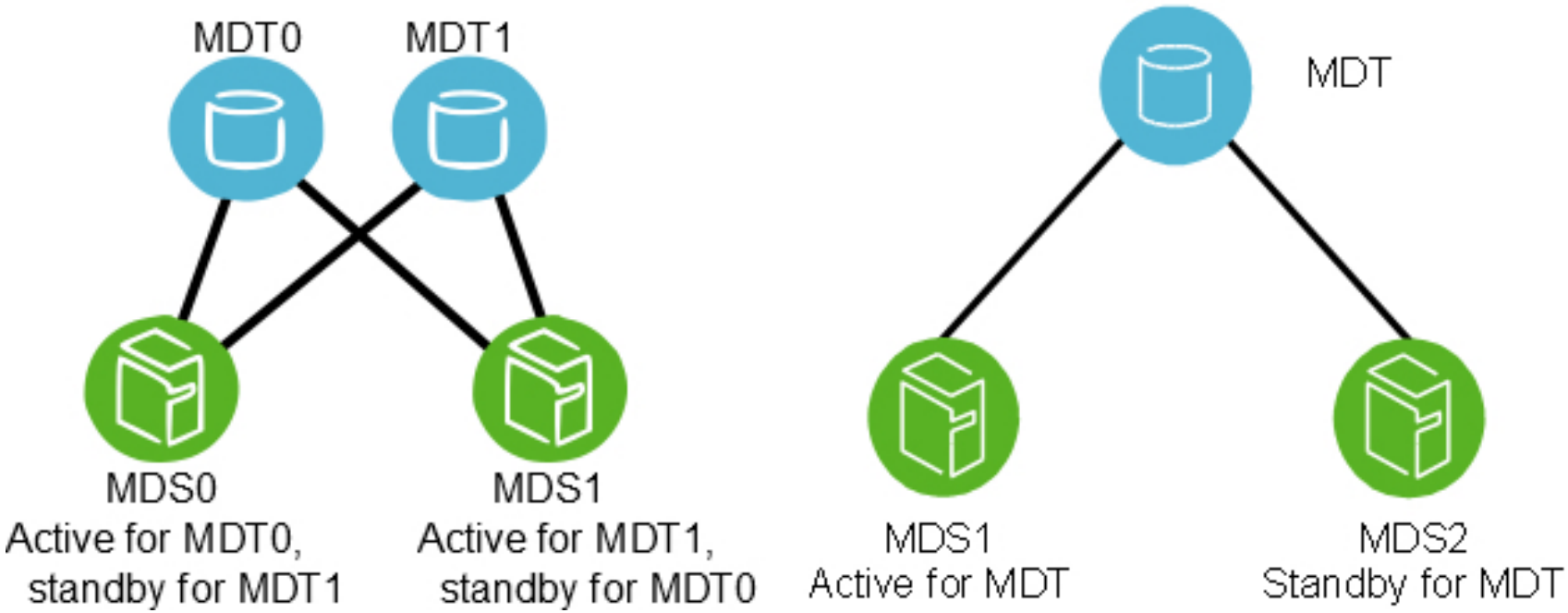

Abbildung: Failover [6]

Quotas

- Ermöglicht den Speicherplatz eines Nutzers oder einer Nutzergruppe zu beschränken
- Beschränkungen für die Anzahl von Inodes (MDT) und Blöcken (OSTs) [10]
- Softlimit:
  - Limit darf für eine bestimmte Zeit (Grace Period) überschritten werden
  - Nach Grace Period verhält sich das Soft Limit wie das Hard Limit
- Hardlimit:
  - Limit kann nicht überschritten werden

Hierarchical Storage Management (HSM)

- HSM Storage:
  - Langsamer Storage (Tape Archive, ...)
  - Günstiger und lange Haltbarkeit
- Lustre:
  - Lustre als Cache für Dateien, auf die oft zugegriffen wird
  - Schnelle Anbindung an das Archiv
- Archivierung und Wiederherstellung wird von einem Client koordiniert: Agent
- „Coordinator“ muss auf jedem MDT aktiviert sein

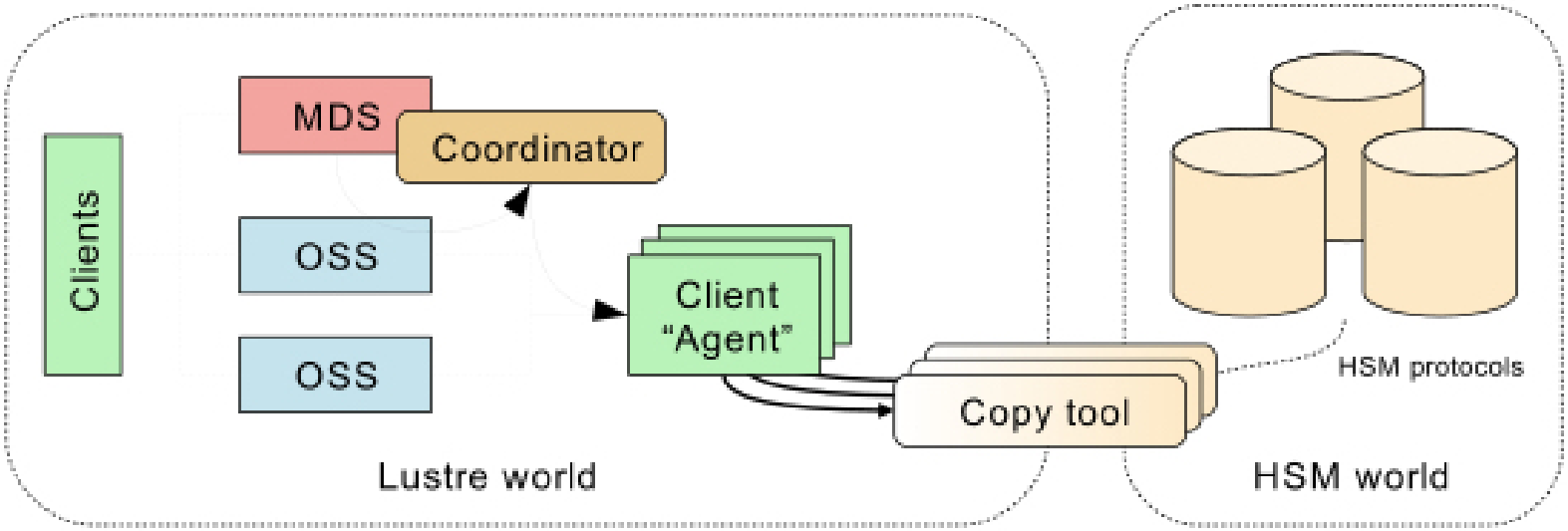

Abbildung: HSM [6]

Zugriff

- Lustre Client

```
1 mkdir -p /mnt/lustre/client
2 mount -t lustre $(MDS):/lustre /mnt/lustre/client
```

Listing 1: Mount

- Einbindung in Windows und OS X durch NFS und CIFS Schnittstellen
- Parallele Programmierung: llapi, MPI-IO

Single Writer I/O

System: Cray XT5 (Jaguar)

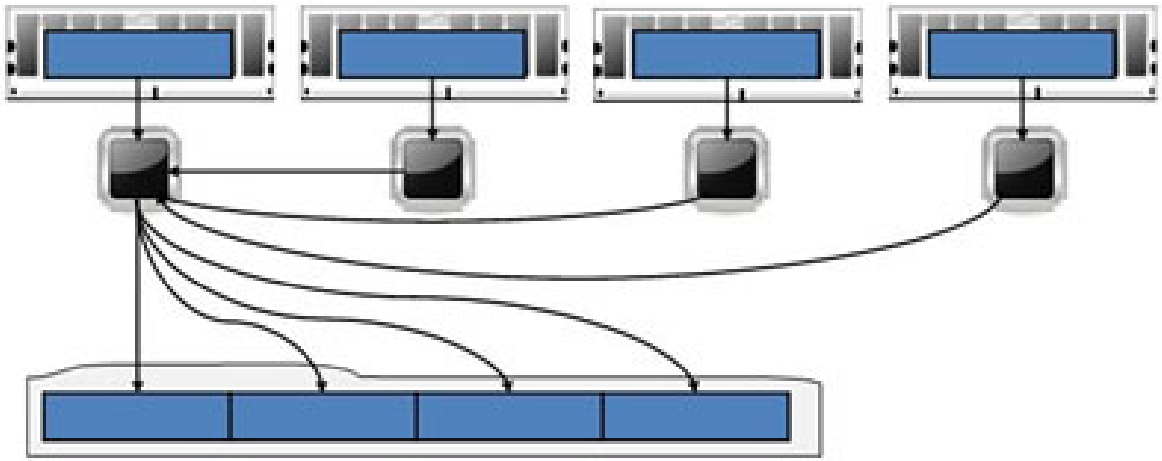

Abbildung: Single Writer I/O [3]

Single Writer I/O

- Dateigröße: 32 MB bis 5 GB

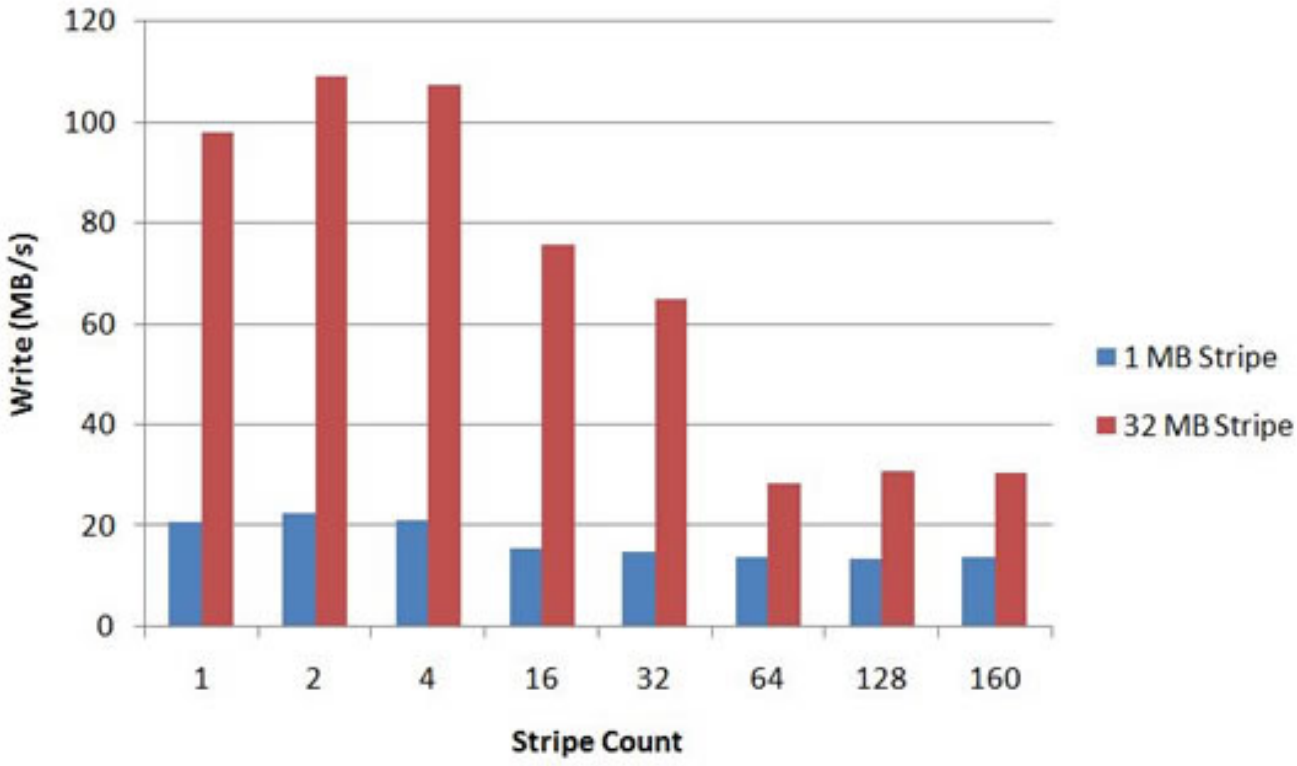

Abbildung: Single Writer I/O [3]

Single Shared File I/O

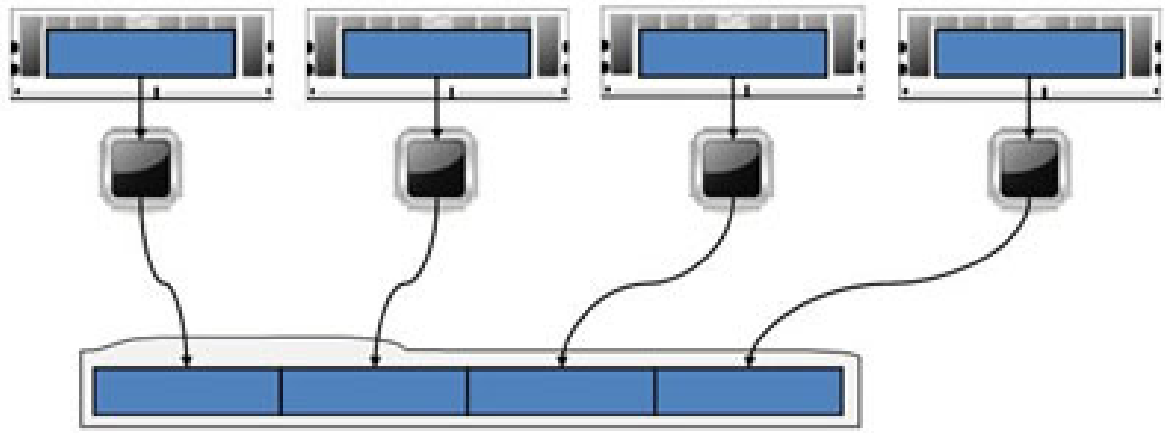

Abbildung: Single Shared File I/O [3]

Single Shared File I/O

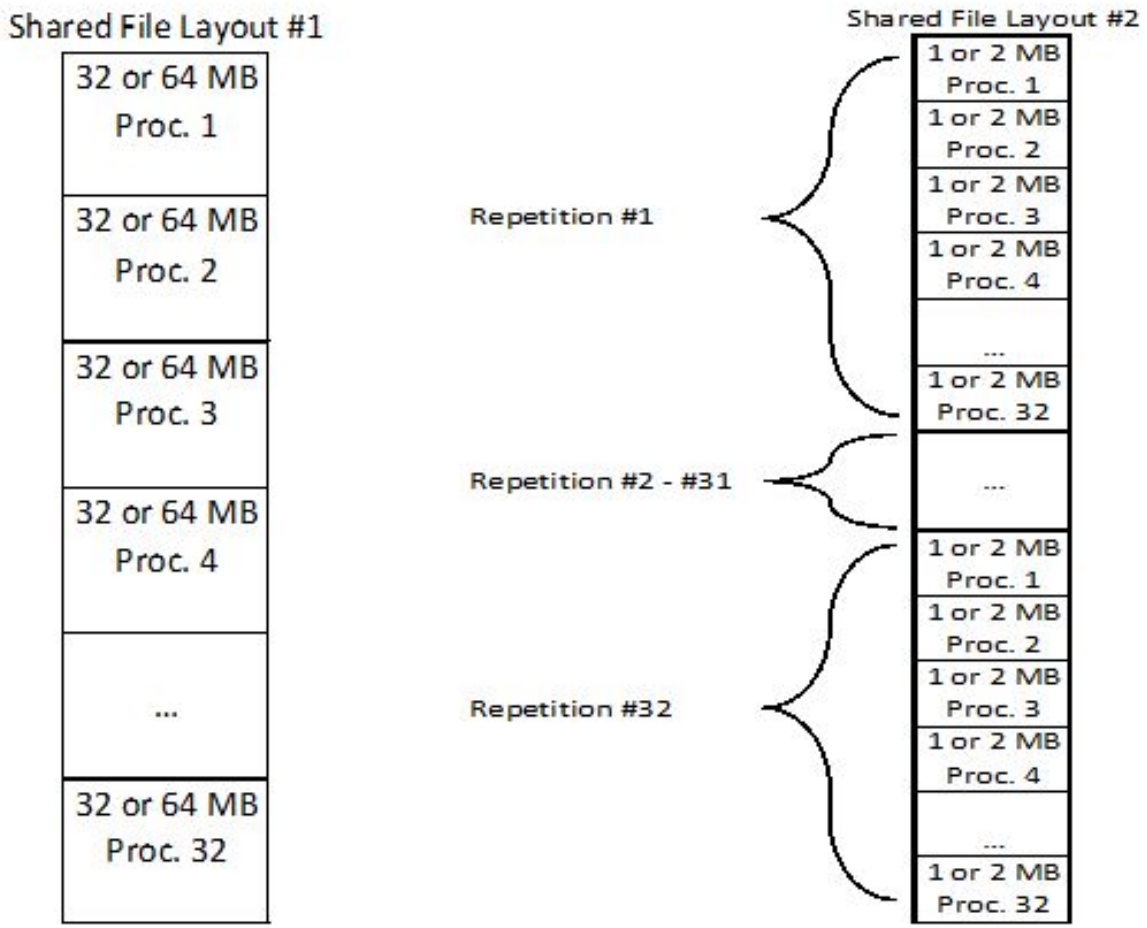

Abbildung: Single Shared Layout [3]

Single Shared File I/O

- Dateigröße: 1 GB, 2 GB
- Stripe Count: 32, 64

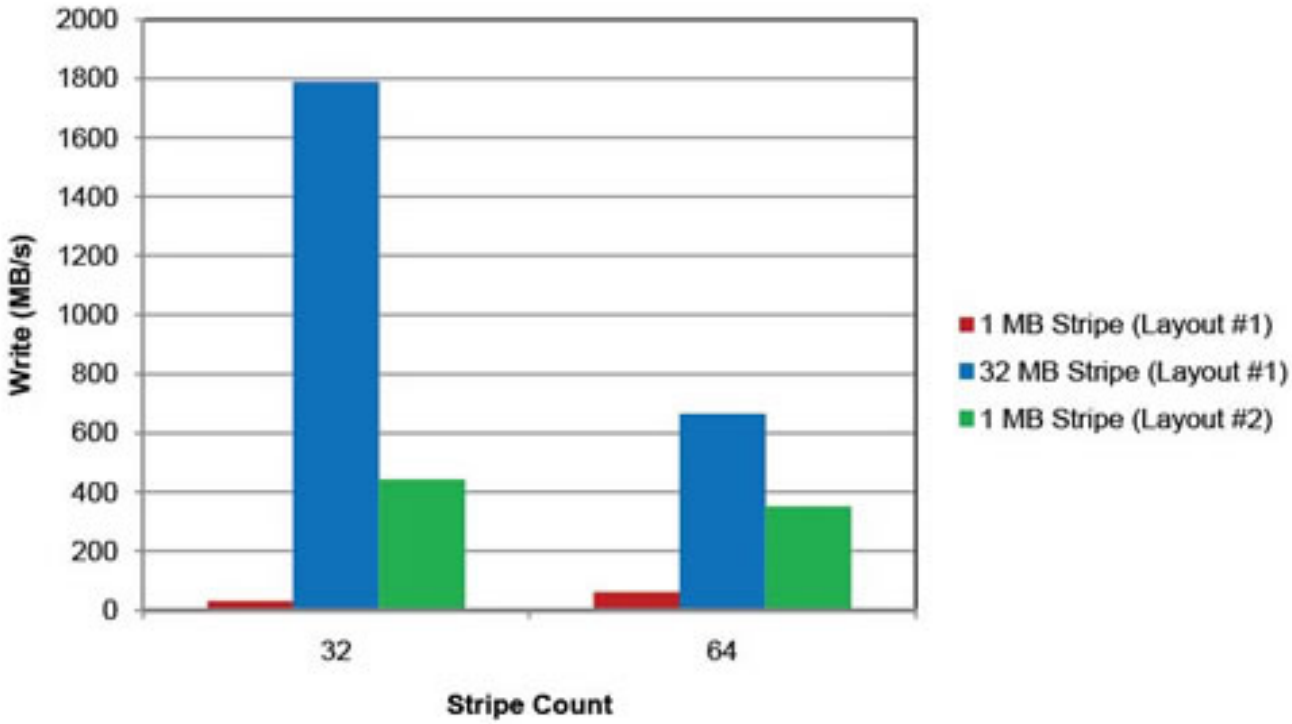

Abbildung: Single Shared File I/O [3]

File Per Process I/O

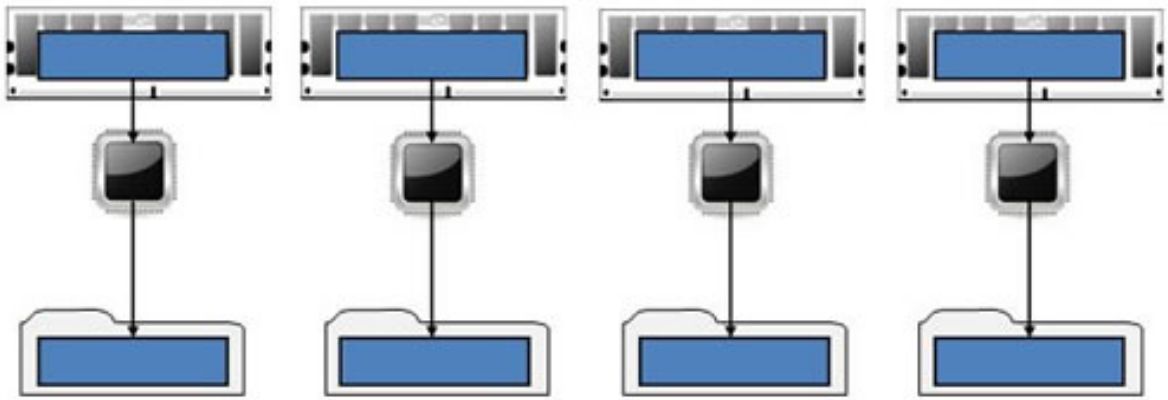

Abbildung: File Per Process I/O [3]

File Per Process I/O

- Dateigröße: 128 MB
- Stripe Count: 1

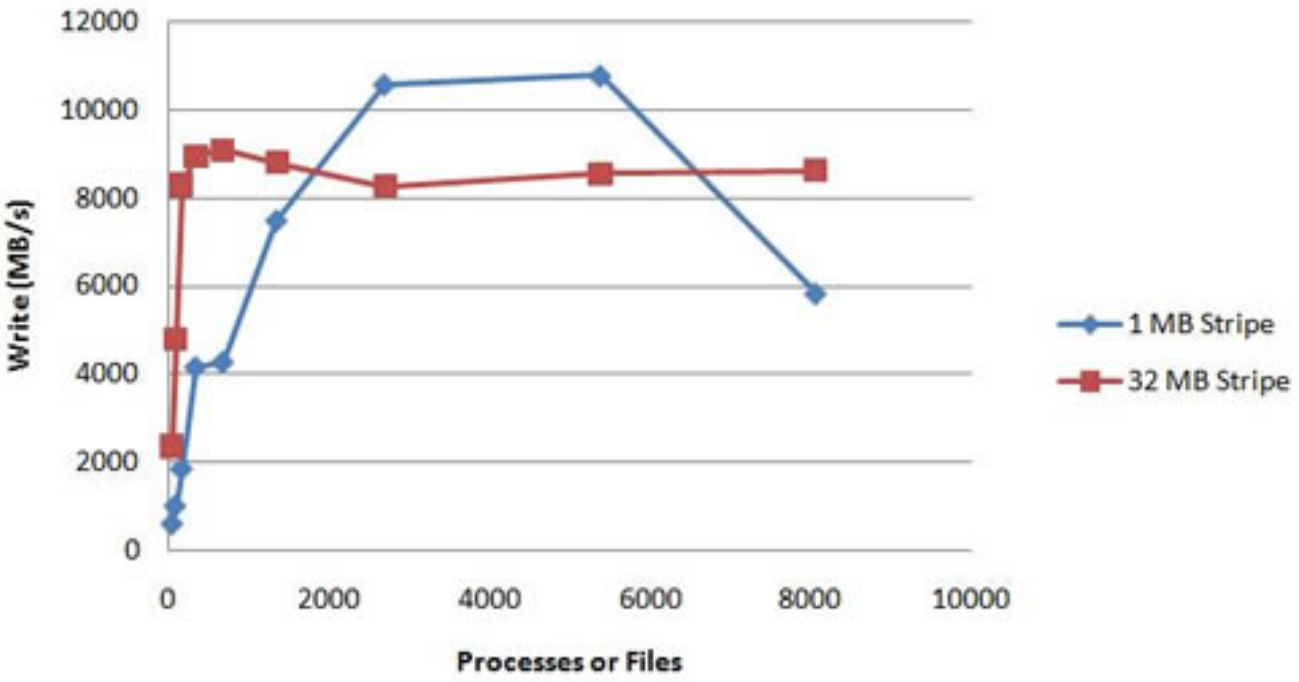

Abbildung: File Per Process I/O [3]

Zusammenfassung

- Paralleler Zugriff auf jeweils beschränkte Ressourcen steigern die Performance enorm
- Parameter müssen sorgsam gewählt werden und variieren je nach Anwendungsfall (Stripe Size, Count)
- Lustre bietet viele Funktionen, die insbesondere für das HLR unabdingbar sind
- Komplexe Konfiguration und lange Einarbeitung nötig

[1] Serial Attached SCSI. Mai 2016. https://de.wikipedia.org/wiki/Serial_Attached_SCSI.

[2] https://www.ornl.gov/. Lustre Architektur. Mai 2016. http://lustre.ornl.gov/lustre101-courses/content/C1/L1/LustreIntro.pdf.

[3] citutor. Dateisicht. Mai 2016. https://www.citutor.org/.

[4] NASA. Lustre Best Practices. Mai 2016. http://www.nas.nasa.gov/hecc/support/kb/lustre-best-practices_226.html.

[5] Malcolm Cowe. AN INTRODUCTION TO LUSTRE ARCHITECTURE. August 2015. https://nci.org.au/wp-content/uploads/2015/08/01-Introduction-to-Lustre-Architecture.pdf.

[6] Lustre. Lustre* Software Release 2.x. Mai 2016. http://doc.lustre.org/lustre_manual.xhtml.

[7] Intel. Lustre system and network administration. Mai 2016. https://nci.org.au/wp-content/uploads/2015/08/02-Introduction-LNET.pdf.

[8] HPC Advisory Counci. Interconnect Analysis: 10GigE and InfiniBand inHigh Performance Computing. Mai 2016. http://www.hpcadvisorycouncil.com/pdf/IB_and_10GigE_in_HPC.pdf.

[9] Mellanox. InfiniBand Performance. Mai 2016. http://www.mellanox.com/page/performance_infiniband.

[10] Intel. Lustre system and network administration - Lustre quota. Juli 2015. https://nci.org.au/wp-content/uploads/2015/08/04-Quota.pdf.