Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Proseminar: Speicher und Dateisysteme

Jan Harder – 08.01.2019
Betreuer: Dr. Michael Kuhn

Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Proseminar Speicher und Dateisysteme | Jan Harder

Lustre

Gliederung

1. Allgemein
    - 1.1 Was ist Lustre?
    - 1.2 Motivation
    - 1.3 Geschichte / Projekt
2. Architektur
    - 2.1 Komponenten
    - 2.2 Clients
    - 2.3 Server und Targets
    - 2.4 Objekt based storages
    - 2.5 LNET
3. Funktionen
    - 3.1 Failover
    - 3.2 Striping
4. Verwendung
5. Aktuelle Entwicklung

Was ist Lustre?

- Paralleles, verteiltes Dateisystem
- Name: Linux + Cluster
- Für Linux-Systeme entwickelt
- Für hohe Auslastung ausgelegt
- Open Source durch GNU GPL (v2)
- POSIX konform

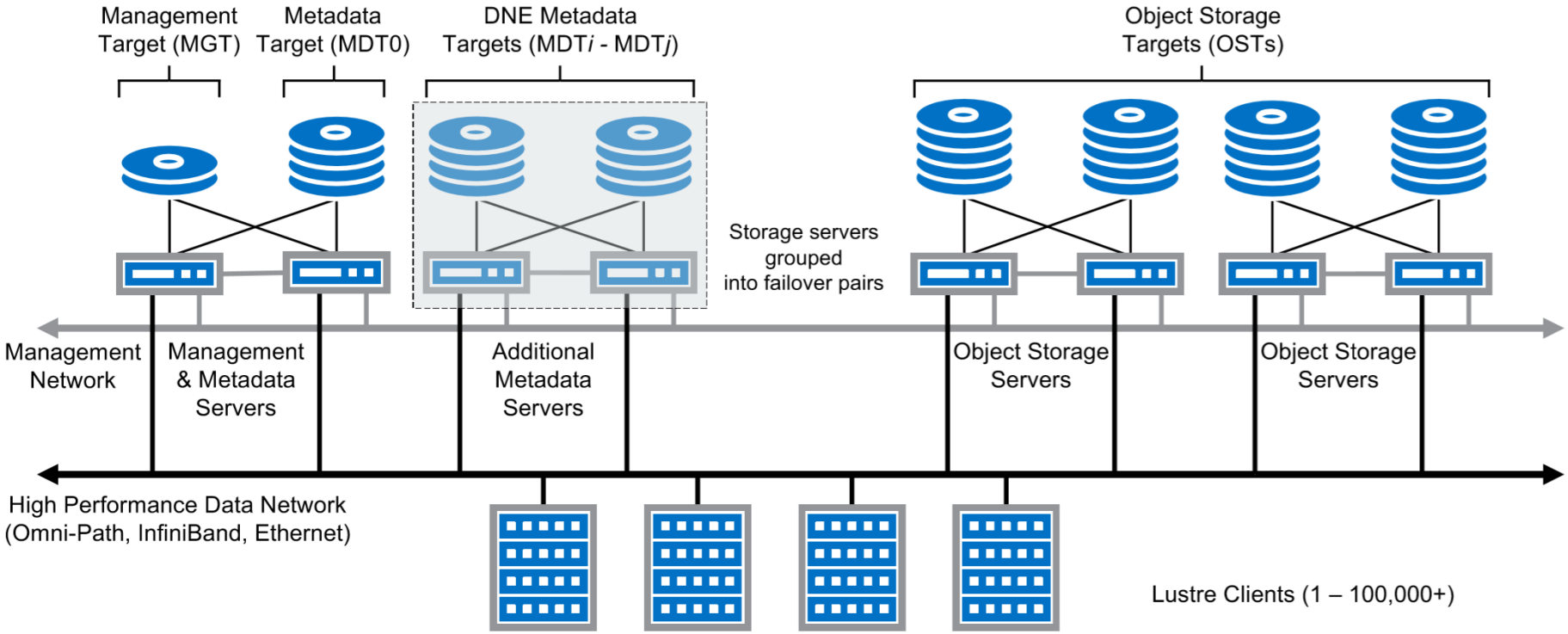
Motivation

- Immer größere Datenmengen
- Größere Dateien
- Server und Netzwerk haben jedoch beschränkten Durchsatz und Kapazität
- -> Dateien über mehrere Server verteilen

Geschichte von Lustre

- 1999: Forschungsprojekt von Peter J. Braam an der Carnegie Mellon University
- 2001: Cluster File Systems in 2001 gegründet
- 2007: Kauf durch Sun Microsystems
- 2010: Kauf durch Oracle - keine Weiterentwicklung
- Verschiedene Gruppen setzen Entwicklung fort (Whamcloud, Open SFS)

Architektur von Lustre



Aufbau eines Lustre Dateisystems von wiki.lustre.org [1]

Komponenten

- Clients
- Metadata Server / Metadata Target
- Management Server / Management Target
- Object Storage Server / Object Storage Target
- Object Storage Devices
- LNET

Clients

- Hohe Anzahl an Clients möglich (100.000+)
- Ein Namensraum für alle Clients
- Clients fassen Meta- und Objektdaten POSIX konform zusammen
- Anwendung muss nicht für Lustre geschrieben werden
- Clients greifen nicht direkt auf Speicher zu
- Oft ohne eigene Festplatten

Management Server / Management Target

- Management Server (MGS) stellt Konfigurationsinformationen bereit
- Daten werden auf Management Target (MGT) gespeichert
- Nur ein MGS für ganzes Lustre Dateisystem
- MGS ist nicht an eigentlichen Operationen beteiligt

Metadata Server / Metadata Target

- Metadata Server (MDS) stellt Metadaten bereit
- Dateiname, Zugriffsrechte, Sperren, Stripes…
- Metadata wird in inodes gespeichert
- Inodes werden auf Metadata Target(s) (MDT) gespeichert
- Kein Zugriff auf Dateien möglich ohne MDS/MDT
- Für das Löschen und Erstellen von Dateien zuständig

Object Storage Server / Object Storage Target

- Datei wird in Stripes auf mehrere Objekte verteilt
- Object Storage Server (OSS) verwaltet die gespeicherten Daten
- Führen I/O Operationen aus (read/write)
- Passiver Speicher (Verteilung durch MDS oder MGS)
- OST sollten gleichmäßig verteilt sein
- Kapazität lässt sich durch Hinzufügen von OST erhöhen

Object Storage Devices (OSD)

- Targets können zwei verschiedene Dateisysteme haben
- LDISKFS: Weiterentwicklung von ext4, Kernel muss angepasst werden
  - Berechnung der inodes Anzahl bei Formatierung
- ZFS: Kernel muss nicht angepasst werden, Installation aufwendiger
  - Dynamisches Berechnen der inodes Anzahl
- Auch Kombinationen von LDISKFS und ZFS möglich

Skalierbarkeit

	Value using LDISKFS backend	Value using ZFS backend	Notes
Maximum stripe count	2000	2000	Limit is 160 for ldiskfs if "ea_inode" feature is not enabled on MDT
Maximum stripe size	< 4GB	< 4GB	
Minimum stripe size	64KB	64KB	
Maximum object size	16TB	256TB	
Maximum file size	31.25PB	512PB*	
Maximum file system size	512PB	8EB*	
Maximum number of files or subdirectories per directory	10M for 48-byte filenames. 5M for 128-byte filenames.	2^{48}	
Maximum number of files in the file system	4 billion per MDT	256 trillion per MDT	
Maximum filename length	255 bytes	255 bytes	
Maximum pathname length	4096 bytes	4096 bytes	Limited by Linux VFS

Tabelle über die Skalierbarkeit von wiki.lustre.org [2]

Lustre Networking (LNET)

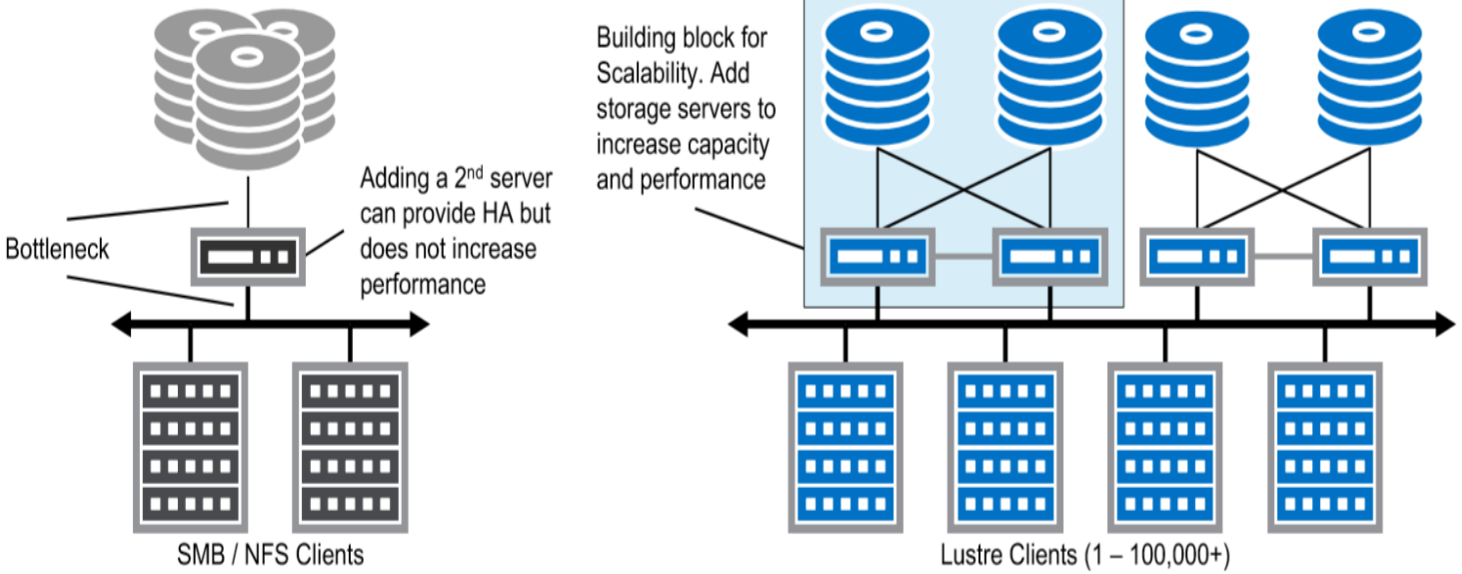
- Netzwerk API: LNET
- Vermittelt zwischen Clients und Server
- Verschiedene Typen unterstützt
  - Ethernet, Infiniband…

Konfiguration

- 1 MGS + MGT
- 1 MDS + MDT — Mindestanforderung
- 1 OSS + OST

- MGS und MDS verbunden
- 2 MDS + MDT — Hohe verfügbarkeit
- 2 OSS + OST

Konfiguration

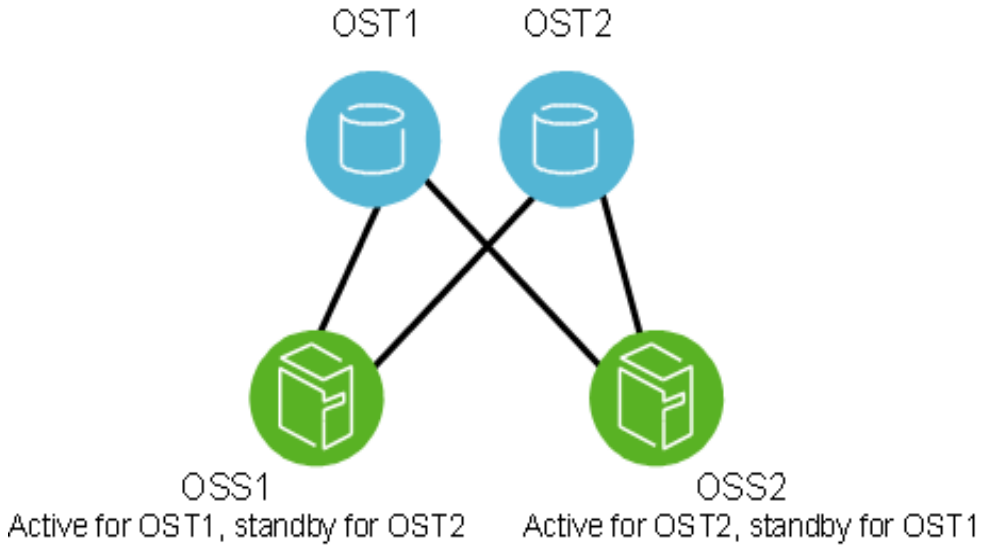


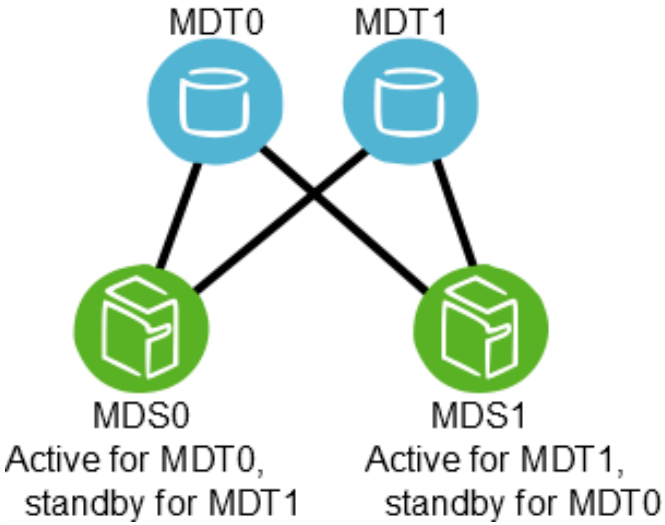
Vergleich von herkömmlichen Dateisystemen und Lustre von wiki.lustre.org [3]

Failover

- Failover-Konfiguration sichert Ausführung bei Serverausfall
- MDS, MGS und OSS unterstützen Failover
- Anordnung in Paaren
- Aktiv-Passiv für MDS und MGS
- Aktiv-Aktiv für OSS
- Transaction log

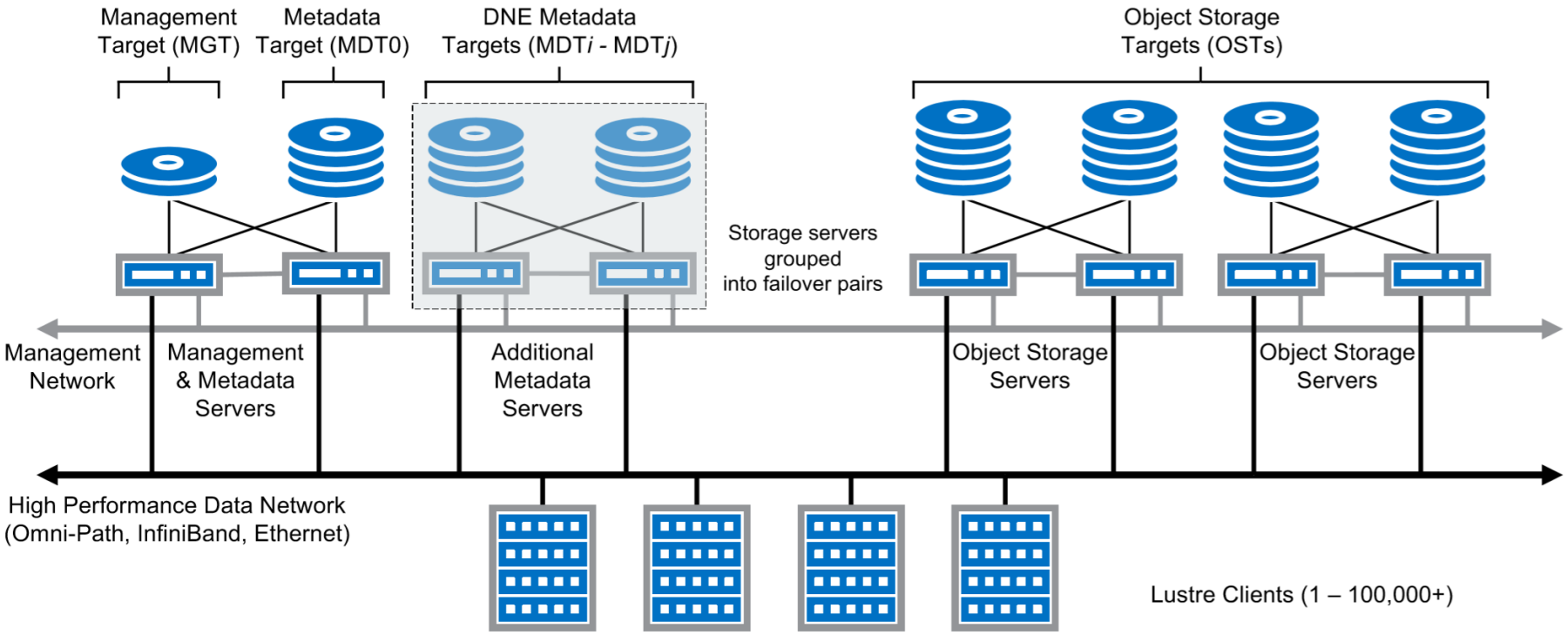
Failover





Aufbau von Failoverpaaren von doc.lustre.org [4]

Failover

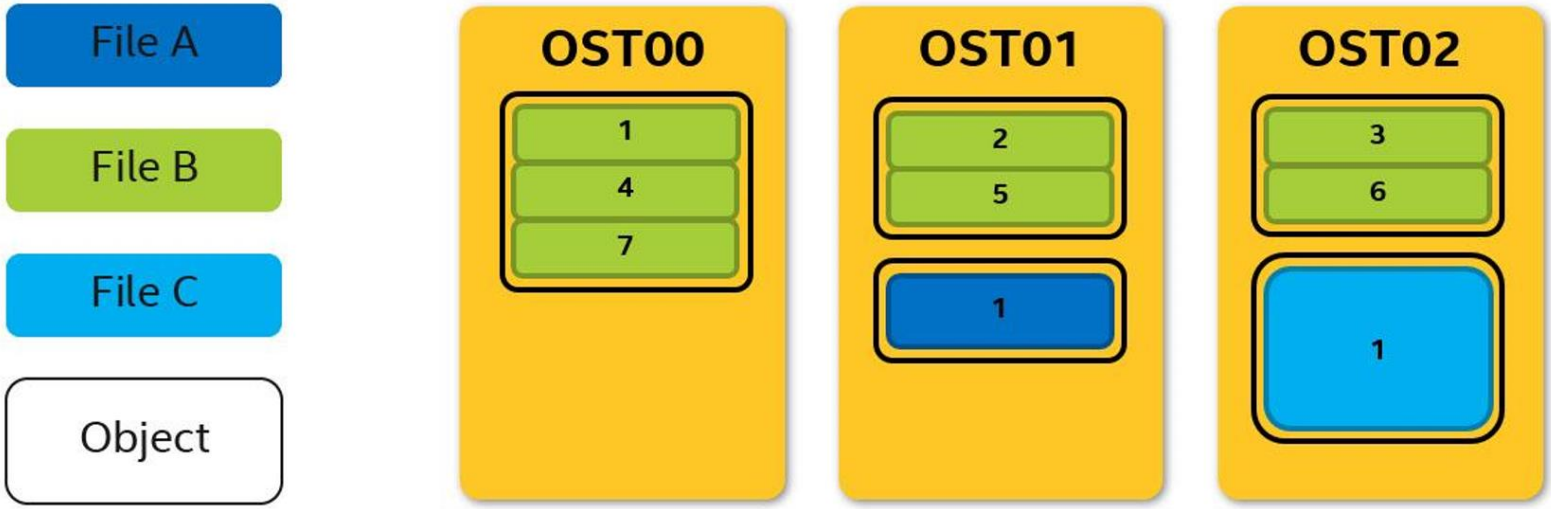


Aufbau eines Lustre Dateisystems von wiki.lustre.org [1]

Striping

- Dateien werden über mehrere OST verteilt
- Dateigröße wird nicht durch OST Kapazität limitiert
- Schnelleres Speichern oder Lesen durch paralleles Speichern
- Datei kann auf bis zu 2000 OST verteilt warden
- Round Robin Algorithmus verteilt Dateien fortlaufend

Striping



Schema von Striping von Intel [5]

Verwendung

- Supercomputer
- Aufwendige Berechnungen
  - Z.B. Wetterprognosen
- Zeitkritische Berechnungen
- Sehr weit verbreitet

Aktuelle Entwicklung

- Keine Lustre Kernel Patches mehr
  - Einfachere und Kernelunabhängige Installation
- Komprimierung von Daten auf Clients und Server

Zusammenfassung

- Hohe Verfügbarkeit durch Failover
- Erhöhte Kapazität und Durchsatz durch parallel Zugriff
- Open Source

Fragen?

Abbildungen:

http://wiki.lustre.org/images/6/64/LustreArchitecture-v4.pdf Seite 5 [1]

http://wiki.lustre.org/images/6/64/LustreArchitecture-v4.pdf Seite 8 [2]

http://wiki.lustre.org/images/6/64/LustreArchitecture-v4.pdf Seite 9 [3]

http://doc.lustre.org/lustre_manual.xhtml [4]

https://www.intel.com/content/dam/www/public/us/en/documents/training/lustre-file-striping.pdf Seite 13 [5]

Jeweils letzter Aufruf am 07.01.2019

https://en.wikipedia.org/wiki/Lustre_(file_system)

https://www.sysgen.de/lustre-parallel-filesystem.html

http://www.linux-magazin.de/ausgaben/2007/11/need-for-speed/7/

http://berrendorf.inf.h-brs.de/lehre/ss05/parsys/s_Lustre.pdf

https://www.intel.com/content/dam/www/public/us/en/documents/training/lustre-file-striping.pdf

https://www.nextplatform.com/2016/05/23/lustre-daos-machine-learning-intels-platform/

Jeweils letzter Aufruf am 07.01.2019

http://wiki.lustre.org/Projects

http://wiki.lustre.org/images/6/64/LustreArchitecture-v4.pdf

https://wr.informatik.uni-hamburg.de/_media/teaching/sommersemester_2015/hea-15-parallele_verteilte_dateisysteme.pdf

http://www.scc.kit.edu/scc/docs/Lustre/Riehm-Lustre.pdf

https://en.wikipedia.org/wiki/Round-robin_scheduling

Jeweils letzter Aufruf am 07.01.2019