

---

# Arbeitsprotokoll

## Vergleich verteilter Dateisysteme

---

Systemtechnik Theorie  
5BHIT 2017/18

Pierre Rieger, Martin Wölfer

Note:  
Betreuer: SABM

Version 0.2  
Begonnen am 23. Januar 2018  
Beendet am 24. Januar 2018

# Inhaltsverzeichnis

1	Aufgabenstellung	1
2	Lustre	1
3	XtreemFS	1
4	Ceph	2

# 1 Aufgabenstellung

Wählen Sie 3 verteilte Dateisysteme und vergleichen Sie deren Funktionsumfang, Funktionsweise, Kompatibilitäten sowie - sofern möglich - die Performance. Versuchen Sie auch die größten Supporter/Entwickler sowie Nutzer der jeweiligen Implementierungen zu finden.

Zur Auswahl u.a.: BeeGFS, Ceph, GlusterFS, Hadoop DFS, HAST, Lustre, OpenAFS, ...

## 2 Lustre

Lustre, eine sprachliche Kombination aus den Wörtern "Linux" und "Cluster", ist besonders für sehr große Systeme mit großen Mengen an Dateien konzipiert. Dabei können Systeme mit Datenmengen über **100 petabyte** unterstützt werden, mit einem Datenfluss über **mehrere gigabyte pro Sekunde**. [1]

Es ist POSIX-konform und läuft auf allen Linux Betriebssystemen. Wie schon bereits erwähnt, hebt sich Lustre vor allem darin von anderen verteilten Dateisystemen ab, dass es unabhängig von Hardware-Einschränkungen den Speicher und Performance problemlos skalieren kann. [2]

Wichtig ist auch noch, dass Metadaten auf anderen Servern gespeichert werden, um für jeweils verschiedene Arbeitsauslastung optimiert werden zu können:

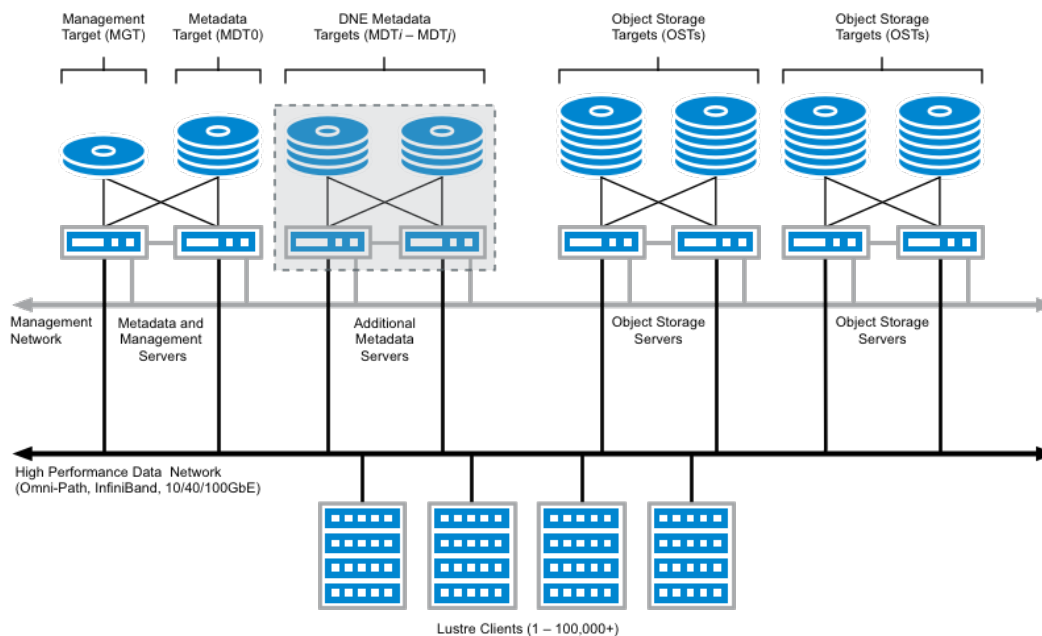


Abbildung 1: Lustre File System Architecture

## 3 XtreamFS

XtreamFS ist ebenfalls ein Dateisystem, welches aber nicht nur verteilte Dateisysteme unterstützt, sondern "general purpose". Es wird keine besondere Hardware oder Software-Kernels benötigt, und es läuft auf Linux, Windows und MAC OS. Im Gegensatz dazu, läuft Lustre nativ nur auf Linux Distributions, bzw. wird ein eigenes Projekt benötigt, welches Lustre in Windows portiert. Zusätzlich ist es ebenfalls "fault-tolerant" und stark skalierbar. [3]

Was nun XtreamFS von anderen verteilten Dateisystemen stark unterscheidet, ist der Fakt dass es dafür entwickelt wurde in "Wide-Area-Networks" eingesetzt zu werden, d.h. es ist überall mountbar und überall zugänglich, auch in sehr großen Netzen. Zusätzlich sind noch Sicherheitsmaßnahmen gegeben, welche sicherstellen, dass wenn über das Netz verteilt wird, auch alle Pakete sicher, vertraulich und **komplett** ankommen. Dies wurde umgesetzt mit verschiedenen **RAID-Konfigurationen** sowie stetig überwachte Prüfsummen. Dies führt zu einem System, welches weniger anfällig auf Serverausfälle bzw. Server-downtime ist. [4]

## 4 Ceph

Der Source-code für Ceph ist für jeden Entwickler einsehbar und nur durch die Community wurde Ceph zu dem, was es heute ist. Das System arbeitet mit Objektspeicherung, welches wieder REST interfaces verwendet um mit vielen Applikationen kompatibel zu sein. Besonders mit S3 und Swift funktioniert Ceph einwandfrei. Zusätzlich wird mit der "Block-Storage" Technik gearbeitet, welche im Endeffekt die Daten welche gespeichert werden soll in Blöcke aufteilt auf welche zugegriffen werden kann. Zusätzlich wird wie bei den anderen Systemen ein POSIX-komformes Dateisystem angeboten, mit welchem hohe Leistung, viel Speicher und leichte Skalierung versprochen wird. [5]

Ein Problem bei Ceph besteht darin, im Vergleich zu den anderen verteilten Dateisystemen ist, dass es nicht in einer Docker-Umgebung verwendet werden kann, dafür müsste CephFS verwendet werden, welche noch nicht ausgereift ist. [6]

Ein klarer Vorteil an Ceph ist, dass es den sogenannten **CRUSH-Algorithmus** verwendet. Dies bedeutet, dass kein zentraler Metadata Service gebraucht wird um Daten zu lokalisieren. Nachteil allerdings an dieser Vorgehensweise, ist das Daten herumgeschoben werden damit der CRUSH-Algorithmus funktioniert. [7]

## Literatur

- [1] Lustre. [http://wiki.lustre.org/Main\\_Page](http://wiki.lustre.org/Main_Page).
- [2] Lustre. <http://wiki.lustre.org/images/6/64/LustreArchitecture-v4.pdf>.
- [3] XtreamFS. <http://www.xtreamfs.org/>.
- [4] XtreamFS. <http://www.xtreamfs.org/faq.php>.
- [5] Ceph. <https://ceph.com/>.
- [6] Jason Feriante. <https://www.quora.com/What-do-you-think-of-Ceph-which-is-claiming-to-be-a-scalable-alternative-to-HDFS>.
- [7] Maksym Yehorov. <https://www.quora.com/What-are-the-advantages-of-using-ceph>.

## Abbildungsverzeichnis

1	Lustre File System Architecture . . . . .	1
---	---	---