

# DELL EMC POWERSCALE ONEFS: EINE TECHNISCHE ÜBERSICHT

## Zusammenfassung

Dieses Whitepaper enthält technische Details zu den wichtigsten Merkmalen und Funktionen des OneFS-Betriebssystems, das für alle Dell EMC PowerScale-Scale-out-NAS-Storage-Lösungen verwendet wird.

September 2021

# Überarbeitungen

Version	Datum	Comment
1.0	November 2013	Erstausgabe für OneFS 7.1
2.0	Juni 2014	Aktualisiert für OneFS 7.1.1
3.0	November 2014	Aktualisiert für OneFS 7.2
4.0	Juni 2015	Aktualisiert für OneFS 7.2.1
5.0	November 2015	Aktualisiert für OneFS 8.0
6.0	September 2016	Aktualisiert für OneFS 8.0.1
7.0	April 2017	Aktualisiert für OneFS 8.1
8.0	November 2017	Aktualisiert für OneFS 8.1.1
9.0	Februar 2019	Aktualisiert für OneFS 8.1.3
10.0	April 2019	Aktualisiert für OneFS 8.2
11.0	August 2019	Aktualisiert für OneFS 8.2.1
12.0	Dezember 2019	Aktualisiert für OneFS 8.2.2
13.0	Juni 2020	Aktualisiert für OneFS 9.0
14.0	September 2020	Aktualisiert für OneFS 9.1
15.0	April 2021	Aktualisiert für OneFS 9.2
16.0	September 2021	Aktualisiert für OneFS 9.3

## Mitwirkung

Dieses Whitepaper wurde erstellt von:

Autor: Nick Trimbee

Die Informationen in dieser Veröffentlichung werden ohne Gewähr zur Verfügung gestellt. Dell Inc. macht keine Zusicherungen und übernimmt keine Haftung jedweder Art im Hinblick auf die in diesem Dokument enthaltenen Informationen und schließt insbesondere jedwede implizierte Haftung für die Handelsüblichkeit und die Eignung für einen bestimmten Zweck aus.

Für die Nutzung, das Kopieren und die Verbreitung der in dieser Veröffentlichung beschriebenen Software ist eine entsprechende Softwarelizenz erforderlich.

Copyright © Dell Inc. oder ihre Tochtergesellschaften. All Rights Reserved. Dell, EMC, Dell EMC und andere Marken sind Marken von Dell Inc. Oder ihren Tochtergesellschaften. Alle anderen Marken können Marken ihrer jeweiligen Inhaber sein.

# INHALTSVERZEICHNIS

Einleitung.....	4
Übersicht über OneFS .....	4
PowerScale-Nodes.....	5
Netzwerk .....	6
Überblick über die OneFS-Software .....	7
Struktur des Dateisystems .....	10
Datenlayout .....	11
Dateischreibvorgänge .....	12
OneFS-Caching.....	15
OneFS-Cachekohärenz .....	17
Level-1-Cache .....	18
Level-2-Cache .....	19
Level-3-Cache .....	19
Lesen von Dateien .....	20
Sperrungen und gleichzeitiger Zugriff .....	22
I/O-Vorgänge mit Multithreading .....	23
Data Protection.....	23
Kompatibilität.....	31
Unterstützte Protokolle .....	32
Unterbrechungsfreie Vorgänge – Protokollunterstützung .....	33
Dateifilter .....	33
Datendeduplizierung – SmartDedupe .....	33
Storage-Effizienz kleiner Dateien.....	34
Inline-Datenreduzierung.....	35
Schnittstellen .....	37
Authentifizierung und Zugriffskontrolle.....	38
Active Directory .....	39
Zugriffszonen.....	39
Rollenbasierte Verwaltung .....	40
OneFS-Auditing.....	40
Softwareupgrade .....	40
OneFS-Data-Protection- und Managementsoftware.....	41
Fazit.....	43
MACHEN SIE DEN NÄCHSTEN SCHRITT.....	43

# Einleitung

Die drei Ebenen des herkömmlichen Speichermodells (Dateisystem, Volume Manager und Data Protection) wurden im Laufe der Zeit auf die Anforderungen kleinerer Speicherarchitekturen hin weiterentwickelt, sind heute aber hochgradig komplex und nicht gut an Systeme mit einer Skalierung im Petabytebereich angepasst. Das Betriebssystem OneFS ersetzt alle diese Schichten und bietet ein vereinheitlichtes Clusterdateisystem mit integrierter skalierbarer Data Protection, für das kein Volume-Management erforderlich ist. OneFS ist ein grundlegender Baustein für Scale-out-Infrastrukturen und ermöglicht eine umfassende Skalierbarkeit und enorme Effizienz und wird zur Unterstützung aller Dell EMC PowerScale NAS-Storage-Lösungen verwendet.

Entscheidend ist, dass OneFS nicht nur bei Computern, sondern auch im Hinblick auf den menschlichen Faktor skalierbar ist. So können große Systeme von einem Bruchteil der für herkömmliche Speichersysteme erforderlichen Mitarbeiter gemanagt werden. OneFS sorgt für weniger Komplexität und umfasst automatische Fehlerkorrektur und automatische Managementfunktionen, durch die der Arbeitsaufwand für das Speichermanagement drastisch reduziert wird. OneFS bietet außerdem Parallelität auf einer sehr tiefgehenden Betriebssystemebene, sodass nahezu jeder wichtige Systemservice auf mehrere Hardwareeinheiten verteilt wird. Dadurch kann OneFS bei einer Erweiterung der Infrastruktur in nahezu jede Richtung skaliert werden. Was heute funktioniert, wird also auch bei einem zukünftigen Anwachsen des Datenvolumens weiterhin funktionsfähig sein.

OneFS ist ein vollständig symmetrisches Dateisystem ohne Single-Point-of-Failure, wobei Clustering nicht nur für die Skalierung von Performance und Kapazität eingesetzt wird, sondern auch für das n:n-Failover und für mehrere Redundanzlevel, die weit über die Möglichkeiten von RAID hinausgehen. Der Trend zu Festplattensubsystemen hat sich nach und nach positiv auf die Performance ausgewirkt, wobei zugleich die Speicherdichten zunehmen. OneFS reagiert auf diese Gegebenheiten, indem die Menge an Redundanz sowie die Geschwindigkeit der Reparaturen bei Ausfällen skaliert werden. Somit kann OneFS auf Systeme von mehreren Petabyte skaliert werden und ist gleichzeitig zuverlässiger als kleine herkömmliche Speichersysteme.

PowerScale-Hardware stellt die Appliance bereit, auf der OneFS ausgeführt wird. Dabei handelt es sich um Best-of-Breed-, aber gleichzeitig Standardhardwarekomponenten, so dass die Vorteile der sich ständig verbessernden Kosten- und Effizienzkurven von Standardhardware profitiert. Mit OneFS kann Hardware dem Cluster jederzeit hinzugefügt oder daraus entfernt werden, sodass eine Abstraktion von Daten und Anwendungen von der Hardware besteht. Die Daten haben uneingeschränkte Langlebigkeit und sind vor den Unbeständigkeiten zukünftiger Hardwaregenerationen geschützt. Kosten und Probleme im Zusammenhang mit Datenmigration und Hardwareaktualisierung werden vermieden.

OneFS ist die ideale Lösung für File-basierte und unstrukturierte „Big Data“-Anwendungen in Unternehmensumgebungen, darunter umfangreiche Stammverzeichnisse, Dateifreigaben, Archive, Virtualisierung und Geschäftsanalysen. Daher ist OneFS heute in vielen datenintensiven Branchen weit verbreitet, zum Beispiel Energie, Finanzdienstleistungen, Internet- und Hostingservices, Business Intelligence, Engineering, Fertigung, Medien und Unterhaltung, Bioinformatik, wissenschaftliche Forschung und andere leistungsfähige Computing-Umgebungen.

## Zielgruppe

Dieses Whitepaper enthält Informationen zur Bereitstellung und zum Management eines Dell EMC PowerScale-Clusters und bietet umfassende Hintergrunderläuterungen zur OneFS-Architektur.

Das Zielpublikum für dieses Whitepaper sind alle Personen, die eine PowerScale-Cluster-Storage-Umgebung konfigurieren und managen. Es wird davon ausgegangen, dass der Leser ein grundlegendes Verständnis von Speicher, Netzwerk, Betriebssystemen und Datenmanagement hat.

 Weitere Informationen zu OneFS-Befehlen und zur Funktionskonfiguration finden Sie im [OneFS-Administrationsleitfaden](#).

## Übersicht über OneFS

OneFS führt die drei Schichten herkömmlicher Speicherarchitekturen – Dateisystem, Volume Manager und Data Protection – in einer einzigen Softwareschicht zusammen und schafft auf diese Weise ein einziges intelligentes, verteiltes Dateisystem, das auf einem OneFS-Storage-Cluster ausgeführt wird.

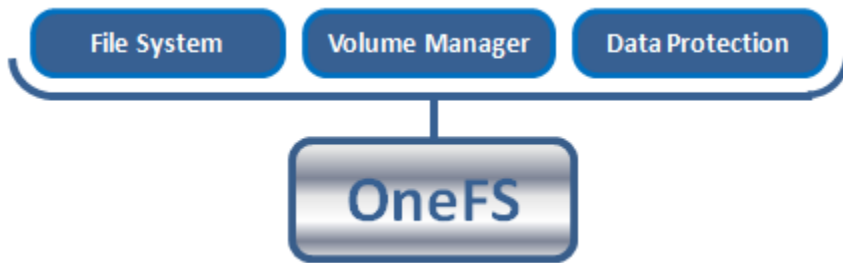


Abbildung 1: OneFS führt Dateisystem, Volume Manager und Data Protection in einem einzigen intelligenten, verteilten System zusammen.

Dies ist die Kerninnovation, mit der Unternehmen Scale-out-NAS in ihren heutigen Umgebungen direkt erfolgreich nutzen können. Die Software entspricht den Grundprinzipien von Scale-out: intelligente Software, handelsübliche Hardware und verteilte Architektur. OneFS ist nicht nur das Betriebssystem, sondern auch das zugrunde liegende Dateisystem, das die Daten im Cluster speichert.

### PowerScale-Nodes

OneFS arbeitet ausschließlich mit dedizierten Plattform-Nodes, die als „Cluster“ bezeichnet werden. Ein einziger Cluster besteht aus mehreren Nodes, die rackmontierbare Enterprise-Appliances mit Arbeitsspeicher, CPU, Netzwerk, Ethernet- oder InfiniBand-Verbindungen mit niedriger Latenz, Festplatten-Controller und Speichermedien sind. Jeder Node im verteilten Cluster verfügt somit über Compute- als auch Storage- oder Kapazitätsfunktionen.

Bei der Gen6-Architektur ist ein einziges Gehäuse mit 4 Nodes in einem 4-HE-Formfaktor (Rackeinheiten) erforderlich, um ein Cluster zu erstellen, das auf bis zu 252 Nodes in OneFS 8.2 und höher skaliert werden kann. Einzelne Node-Plattformen benötigen mindestens drei Nodes und 3RU Rack-Stellfläche zur Bildung eines Clusters. Es gibt verschiedene Arten von Nodes, die alle in einem einzigen Cluster integriert werden können, in dem verschiedene Nodes unterschiedliche Kapazitätsraten für Durchsatz oder Eingabe-/Ausgabevorgänge pro Sekunde (IOPS) bereitstellen. Sowohl das herkömmliche Gen6-Gehäuse als auch die eigenständigen PowerScale-All-Flash-Nodes F900, F600 und F200 sind glücklicherweise im selben Cluster vorhanden.

Jeder einem Cluster hinzugefügte Node oder jedes hinzugefügte Gehäuse erhöht die aggregierte Kapazität für Laufwerk, Cache, CPU und Netzwerk. OneFS nutzt alle Hardwarebausteine auf eine Weise, dass das Endergebnis größer als die Summe seiner Teile ist. Der RAM wird in einem einzigen kohärenten Cache gruppiert, sodass für die I/O-Vorgänge in einem beliebigen Teil des Clusters Daten genutzt werden können, die an beliebigen Speicherorten zwischengespeichert sind. Ein Dateisystem-Journal sorgt dafür, dass Schreibvorgänge bei Stromausfällen sicher sind. Spindeln und CPU werden kombiniert, um Durchsatz, Kapazität und IOPS bei wachsenden Clustern zu steigern, um auf eine oder mehrere Dateien zuzugreifen. Die Storage-Kapazität eines Clusters kann zwischen dutzenden TB und dutzenden PB reichen. Die maximale Kapazität erhöht sich mit steigender Dichte der Speichermedien und Node-Gehäuse.

Die von OneFS betriebenen Plattform-Nodes sind je nach ihrer Funktionalität in verschiedene Klassen oder Ebenen unterteilt:

Tier	I/O Profile	Drive Media	Nodes
<b>Performance</b>	High Perf, Low Latency	Flash NVMe/SAS	F900 F810 F600 F800 F200
<b>Hybrid / Utility</b>	Concurrency & Streaming Throughput	SATA/SAS & SSD	H700 H600 H7000 H5600 H500 H400
<b>Archive</b>	Nearline & Deep Archive	SATA	A300 A200 A3000 A2000

Tabelle 1: Hardware-Tiers und -Node-Typen

## Netzwerk

Es gibt zwei Typen von Netzwerken, die mit einem Cluster verbunden sind: intern und extern.

### Back-end-Netzwerk

Die gesamte Kommunikation zwischen den Nodes in einem Cluster erfolgt über ein dediziertes Back-end-Netzwerk, das entweder aus 10, 40 oder 100 GB Ethernet oder aus QDR InfiniBand (IB) mit niedriger Latenz besteht. Dieses Back-end-Netzwerk, das mit redundanten Switches zur Erzielung hoher Verfügbarkeit konfiguriert ist, dient als Rückwandplatine für den Cluster. So kann jeder Node als Mitwirkender im Cluster fungieren und die Kommunikation zwischen den einzelnen Nodes mit einem privaten, Hochgeschwindigkeitsnetzwerk mit niedriger Latenz isolieren. Für dieses Back-end-Netzwerk wird zur Kommunikation zwischen den einzelnen Nodes IP (Internet Protocol) verwendet.

### Front-end-Netzwerk

Clients stellen über Ethernetverbindungen (10 GbE, 25 GbE 40 GbE oder 100 GbE), die auf allen Nodes verfügbar sind, eine Verbindung mit dem Cluster her. Da jeder Node eigene Ethernetports bietet, kann die im Cluster verfügbare Netzwerkbandbreite je nach Performance und Kapazität linear skaliert werden. Der Cluster unterstützt Standardprotokolle für die Netzwerkcommunication für ein Kundennetzwerk, einschließlich NFS, SMB, HTTP, FTP, HDFS und S3. Darüber hinaus bietet OneFS eine vollständige Integration in IPv4- und IPv6-Umgebungen.

### Ansicht des gesamten Clusters

Das gesamte Cluster mit Hardware, Software und Netzwerken wird in der folgenden Ansicht zusammengestellt:

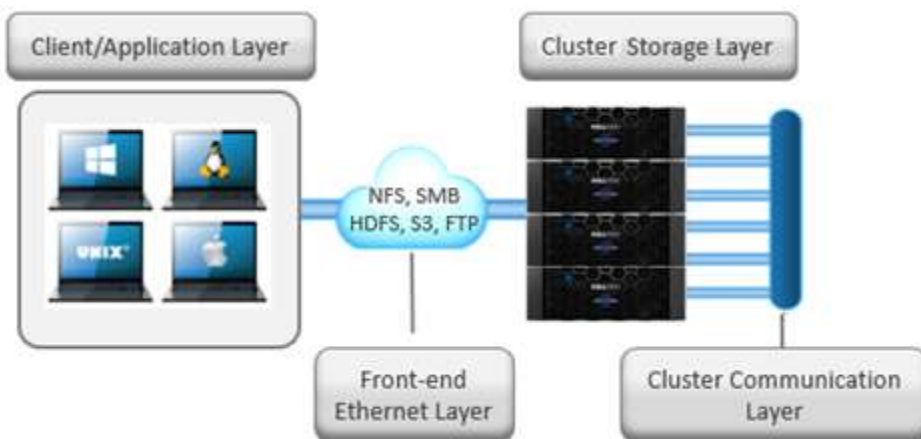


Abbildung 2: Alle Komponenten von OneFS

Die obige Abbildung stellt die gesamte Architektur dar: Software, Hardware und Netzwerk arbeiten in Ihrer Umgebung mit Servern zusammen, sodass ein einziges, vollständig verteiltes Dateisystem entsteht, das je nach Workloads und Änderungen bei Kapazität oder Durchsatz in einer Scale-out-Umgebung dynamisch skaliert werden kann.

OneFS SmartConnect ist ein Load Balancer, der auf der Front-end-Ethernet-Schicht arbeitet, um Clientverbindungen gleichmäßig über den Cluster zu verteilen. SmartConnect unterstützt dynamisches NFS-Failover und -Failback für Linux- und UNIX-Clients und kontinuierliche SMB3-Verfügbarkeit für Windows-Clients. Damit ist dafür gesorgt, dass bei einem Node-Ausfall oder bei präventiven Wartungen, alle In-Flight-Lese- und Schreibvorgänge an einen anderen Node im Cluster weitergegeben werden, um den Vorgang ohne Nutzer- oder Anwendungsunterbrechung abzuschließen.

Während des Failover werden Clients gleichmäßig über alle verbleibenden Nodes im Cluster verteilt, sodass die Auswirkung auf die Performance äußerst gering bleibt. Wenn ein Node aus einem beliebigen Grund, einschließlich eines Ausfalls, abgeschaltet wird, werden die virtuellen IP-Adressen auf diesem Node nahtlos zu einem anderen Node im Cluster migriert. Wenn der Offline-Node wieder online geschaltet wird, gleicht SmartConnect die NFS- und SMB3-Clients im gesamten Cluster automatisch aus, um maximale Storage- und Performanceauslastung sicherzustellen. Bei regelmäßigen Systemwartungsarbeiten und Softwareupdates ermöglicht diese Funktion fortlaufende Upgrades pro Node, die für die Dauer der Wartung vollständige Verfügbarkeit bieten.

# Überblick über die OneFS-Software

## Betriebssystem

OneFS ist auf einem BSD-basierten UNIX-Betriebssystem aufgebaut. Es bietet nativen Support für die Semantik von Linux bzw. UNIX sowie Windows, einschließlich Hardlinks, Löschvorgang beim Schließen, atomares Umbenennen, ACLs und erweiterte Attribute. Als grundlegendes Betriebssystem wird BSD verwendet, da es sich um ein ausgereiftes und bewährtes Betriebssystem handelt, das von den Innovationen der Open-Source-Community profitiert. Ab OneFS 8.2 ist die zugrunde liegende Betriebssystemversion FreeBSD 11.

## Client Services

Die von Clients zur Interaktion mit OneFS verwendeten Front-end-Protokolle werden als Client Services bezeichnet. Im Abschnitt „Unterstützte Protokolle“ finden Sie eine detaillierte Liste der unterstützten Protokolle. Um zu verstehen, wie OneFS mit Clients kommuniziert, teilen wir das I/O-Subsystem in zwei Hälften auf: in die obere Hälfte, oder den „Initiator“, und in die untere Hälfte, oder den „Teilnehmer“. Jeder Node im Cluster ist Teilnehmer an einem bestimmten I/O-Vorgang. Der mit dem Client verbundene Node ist der Initiator. Dieser Node fungiert als „Kapitän“ für den gesamten I/O-Vorgang. Die Lese- und Schreibvorgänge werden weiter unten genauer beschrieben

## Clustervorgänge

In einer Clusterarchitektur gibt es Clusterjobs, die für die Integrität und Verwaltung des Clusters selbst verantwortlich sind. Diese Jobs werden durch die OneFS-Job-Engine gesteuert. Die Job-Engine wird im gesamten Cluster ausgeführt und dient zum Aufteilen und Durchführen umfangreicher Speichermanagement- und Schutzaufgaben. Hierbei werden die Aufgaben in kleinere Arbeitselemente aufgeteilt und diese Teile des Gesamtjobs dann mehreren Worker Threads auf den einzelnen Nodes zugewiesen. Der Fortschritt wird während der Jobausführung überwacht und gemeldet. Nach Abschluss oder Abbruch wird ein detaillierter Bericht mit Statusangabe bereitgestellt.

Die Job-Engine enthält ein umfassendes Kontrollpunktsystem, sodass Jobs nicht nur gestartet und beendet, sondern auch angehalten und fortgesetzt werden können. Das Job Engine-Framework umfasst außerdem ein adaptives Auswirkungsmanagementsystem.

Die Job-Engine führt Jobs im gesamten Cluster in der Regel im Hintergrund aus, wobei freie oder speziell reservierte Kapazitäten und Ressourcen eingesetzt werden. Die Jobs selbst können in drei Hauptklassen eingeteilt werden:

## Dateisystemwartungsjobs

Diese Jobs führen im Hintergrund Dateisystemwartungen durch und erfordern in der Regel Zugriff auf alle Nodes. Diese Jobs müssen in Standardkonfigurationen und oft im heruntergestuften Cluster ausgeführt werden. Beispiele: Dateisystemschutz und erneuter Aufbau von Laufwerken

## Funktionsunterstützungsjobs

Die Supportjobs für Funktionen umfassen gewisse erweiterte Funktionen für das Speichermanagement und werden in der Regel nur dann ausgeführt, wenn die Funktion konfiguriert wurde. Beispiele hierfür sind Deduplizierung und Virenüberprüfungen.

## Nutzeraktionen

Diese Jobs werden direkt vom Storage-Administrator ausgeführt, um bestimmte Datenmanagementziele zu erreichen. Beispiele: Parallele Strukturlöschvorgänge und Berechtigungswartung

Die folgende Tabelle enthält eine umfassende Liste der verfügbaren Job-Engine-Jobs, Informationen zu den durchgeführten Vorgängen sowie den jeweiligen Methoden für den Dateisystemzugriff:

Jobname	Jobbeschreibung	Zugriffsmethode
AutoBalance	Gleicht freien Speicherplatz im Cluster aus.	Laufwerk + LIN

Jobname	Jobbeschreibung	Zugriffsmethode
<b>AutoBalanceLin</b>	Gleicht freien Speicherplatz im Cluster aus.	<b>LIN</b>
<b>AVScan</b>	Virenüberprüfung, die vom Virenschutzserver ausführt wird.	<b>Struktur</b>
<b>ChangelistCreate</b>	Erstellen einer Liste der Änderungen zwischen zwei aufeinanderfolgenden SyncIQ-Snapshots	<b>Änderungsliste</b>
<b>CloudPoolsLin</b>	Archiviert Daten zu einem Cloud-Anbieter gemäß einer Dateipool-Richtlinie.	<b>LIN</b>
<b>CloudPoolsTreewalk</b>	Archiviert Daten zu einem Cloud-Anbieter gemäß einer Dateipool-Richtlinie.	<b>Struktur</b>
<b>Collect</b>	Wiedergewinnung von Speicherplatz, der aufgrund eines nicht verfügbaren Node oder Laufwerks nicht freigegeben werden konnte, da diese verschiedene Ausfallbedingungen aufweisen	<b>Laufwerk + LIN</b>
<b>ComplianceStoreDelete</b>	SmartLock-Job für die Garbage Collection im Compliance-Modus.	<b>Struktur</b>
<b>Dedupe</b>	Dedupliziert identische Blöcke im Dateisystem.	<b>Struktur</b>
<b>DedupeAssessment</b>	Probewertung der Vorteile der Deduplizierung	<b>Struktur</b>
<b>DomainMark</b>	Verknüpft einen Pfad und seine Inhalte mit einer Domain.	<b>Struktur</b>
<b>DomainTag</b>	Verknüpft einen Pfad und seine Inhalte mit einer Domain.	<b>Struktur</b>
<b>EsrsMftDownload</b>	ESRS-verwaltete Dateiübertragung für Lizenzdateien.	
<b>FilePolicy</b>	Effiziente Richtlinien-Job für SmartPools-Dateipools.	<b>Änderungsliste</b>
<b>FlexProtect</b>	Stellt das Dateisystem wieder her und schützt es erneut, um sich von einem Fehlerszenario zu erholen.	<b>Laufwerk + LIN</b>
<b>FlexProtectLin</b>	Schützt das Dateisystem wieder.	<b>LIN</b>
<b>FSAnalyze</b>	Erfasst Dateisystem-Analysedaten, die im Zusammenhang mit InsightIQ verwendet werden.	<b>Änderungsliste</b>
<b>IndexUpdate</b>	Erstellt und aktualisiert einen effizienten Dateisystemindex für den FilePolicy- und FSAnalyze-Jobs.	<b>Änderungsliste</b>
<b>IntegrityScan</b>	Führt eine Online-Überprüfung und Korrektur aller Dateisysteminkonsistenzen durch.	<b>LIN</b>
<b>LinCount</b>	Scannt und zählt die logischen Inodes des Dateisystems (LINs).	<b>LIN</b>
<b>MediaScan</b>	Scannt Laufwerke nach Fehlern auf Medienebene.	<b>Laufwerk + LIN</b>
<b>MultiScan</b>	Gleichzeitige Ausführung von Collect- und AutoBalance-Jobs	<b>LIN</b>
<b>PermissionRepair</b>	Korrektur von Berechtigungen für Dateien und Verzeichnisse	<b>Struktur</b>



Jobname	Jobbeschreibung	Zugriffsmethode
<b>QuotaScan</b>	Aktualisiert die Quota-Berechnung für Domains, die unter einem vorhandenen Verzeichnispfad erstellt werden.	<b>Struktur</b>
<b>SetProtectPlus</b>	Wendet die Standard-Dateirichtlinie an. Dieser Job ist deaktiviert, wenn SmartPools auf dem Cluster aktiviert ist.	<b>LIN</b>
<b>ShadowStoreDelete</b>	Gibt Speicherplatz frei, der einem Schattenspeicher zugeordnet ist.	<b>LIN</b>
<b>ShadowStoreProtect</b>	Schützen von Schattenspeichern, die von einem Lin mit höherem angefordertem Schutz referenziert werden.	<b>LIN</b>
<b>ShadowStoreRepair</b>	Reparieren von Schattenspeichern.	<b>LIN</b>
<b>SmartPools</b>	Job, der Daten zwischen den Tiers von Nodes innerhalb desselben Clusters ausführt und verschiebt. Führt außerdem die CloudPools-Funktion aus, sofern sie lizenziert und konfiguriert ist.	<b>LIN</b>
<b>SmartPoolsTree</b>	Erzwingt SmartPools-Datei-Policies in einer Unterstruktur.	<b>Struktur</b>
<b>SnapRevert</b>	Setzt einen gesamten Snapshot auf den Head zurück.	<b>LIN</b>
<b>SnapshotDelete</b>	Schaffen von freiem Speicherplatz durch gelöschte Snapshots	<b>LIN</b>
<b>TreeDelete</b>	Löscht einen Pfad im Dateisystem direkt aus dem Cluster selbst.	<b>Struktur</b>
<b>Undedupe</b>	Entfernt die Deduplizierung identischer Blöcke im Dateisystem.	<b>Struktur</b>
<b>Durchführen eines Upgrades von</b>	Upgrade des Clusters auf eine spätere OneFS-Version.	<b>Struktur</b>
<b>WormQueue</b>	Scannen der SmartLock-LIN-Warteschlange.	<b>LIN</b>

Abbildung1: OneFS Job Engine-Jobbericht

Die Wartungsjobs für das Dateisystem werden zwar nach einem festen Zeitplan oder als Reaktion auf ein bestimmtes Dateisystemereignis standardmäßig ausgeführt, aber für jeden Job der Job-Engine können sowohl das Prioritätslevel (im Verhältnis zu anderen Jobs) als auch die Auswirkungen-Policy konfiguriert werden.

Eine Auswirkungen-Policy kann ein oder mehrere Auswirkungsintervalle umfassen, die Zeitblöcke in einer bestimmten Woche definieren. Jedes Auswirkungsintervall kann auf ein vordefiniertes Auswirkungslevel konfiguriert werden, durch das die Menge der für einen bestimmten Clustervorgang zu verwendenden Clusterressourcen angegeben wird. Für die Job-Engine verfügbare Auswirkungslevel:

- Angehalten
- Niedrig
- Mittel
- Hoch

Dieser Grad an Granularität ermöglicht die Konfiguration von Auswirkungsintervallen und -stufen pro Job, um einen reibungslosen Clusterbetrieb zu erreichen. Die resultierenden Auswirkungs-Policies geben vor, wann ein Job ausgeführt wird und welche Ressourcen dafür genutzt werden können.

Zusätzlich werden die Jobs der Job-Engine auf einer Skala von eins bis zehn priorisiert, wobei ein niedrigerer Wert eine höhere Priorität bedeutet. Dieses Konzept ähnelt dem der UNIX Scheduling Utility „nice“.

Mit der Job-Engine können bis zu drei Jobs gleichzeitig ausgeführt werden. Diese gleichzeitige Jobausführung unterliegt den folgenden Kriterien:

- Jobpriorität
- Ausschlusssätze: Jobs, die nicht gemeinsam ausgeführt werden können (d. h. FlexProtect und AutoBalance)
- Clusterintegrität: Die meisten Jobs können nicht ausgeführt werden, wenn sich das Cluster in einem heruntergestuften Status befindet.

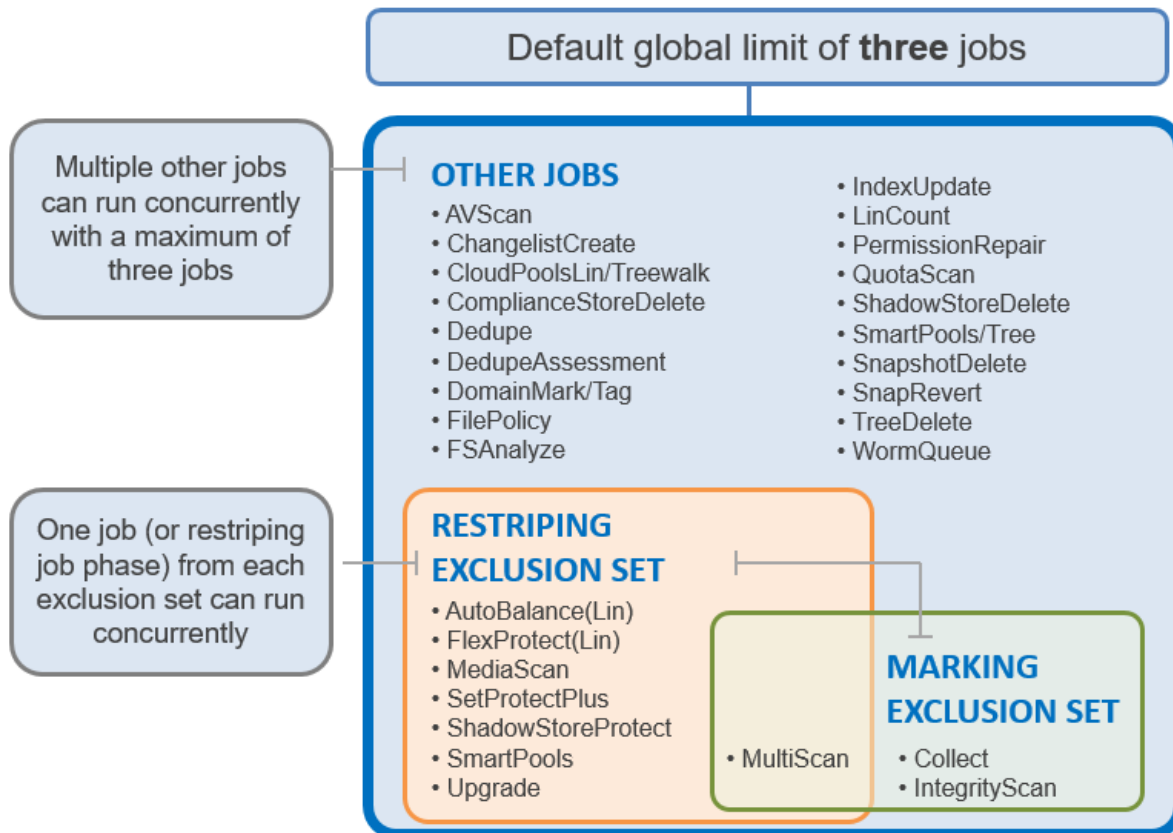


Abbildung 4: Ausschlusssätze der OneFS Job Engine

📖 Weitere Informationen finden Sie im Whitepaper [OneFS Job Engine](#).

## Struktur des Dateisystems

Das OneFS-Dateisystem basiert auf dem UNIX-Dateisystem (UFS). Es handelt sich daher um ein äußerst schnelles, verteiltes Dateisystem. Jedes Cluster erstellt einen einzigen Namespace und ein Dateisystem. Somit wird das Dateisystem über alle Nodes im Cluster verteilt und ist für Clients über eine Verbindung zu einem beliebigen Node im Cluster zugänglich. Es gibt keine Partitionierung und es ist keine Volume-Erstellung erforderlich. Statt den Zugriff auf freien Speicherplatz und nicht autorisierte Dateien auf Ebene des physischen Volume zu beschränken, bietet OneFS dieselbe Funktion in der Software über Freigabe- und Dateiberechtigungen und über den SmartQuotas-Service, der ein Quotamanagement auf Verzeichnisebene bereitstellt.

📖 Weitere Informationen finden Sie im Whitepaper [OneFS SmartQuotas](#).

Da alle Informationen über das interne Netzwerk von den Nodes gemeinsam verwendet werden, können Daten auf einen beliebigen Node geschrieben bzw. von einem beliebigen Node gelesen werden. Dies führt zu einer optimierten Performance, wenn mehrere Nutzer zur gleichen Zeit denselben Datensatz lesen bzw. darin schreiben.

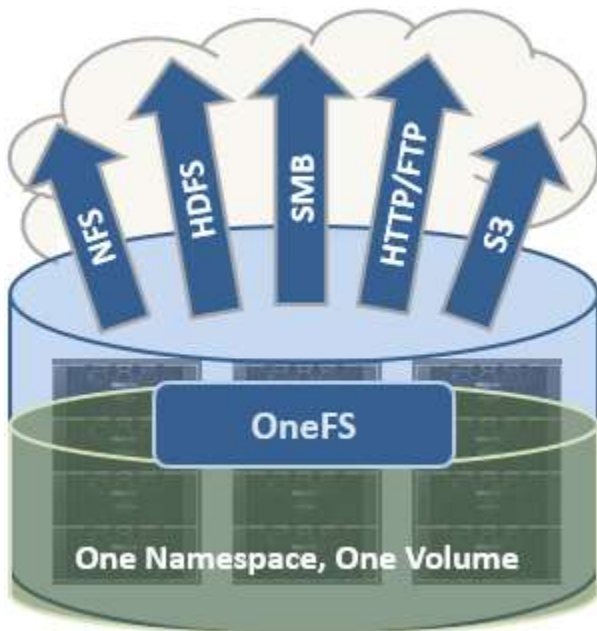


Abbildung 5: Ein einziges Dateisystem mit mehreren Zugriffspunkten

OneFS ist tatsächlich ein einziges Dateisystem mit einem globalen Namespace. Daten und Metadaten werden im Hinblick auf Redundanz und Verfügbarkeit auf mehrere Nodes in Stripes verteilt. Der Storage wurde für NutzerInnen und AdministratorInnen komplett virtualisiert. Die Dateistruktur kann ohne Planung oder Überwachung der Nutzerverwendung organisch wachsen. Die AdministratorInnen brauchen sich keine Gedanken über das Tiering von Dateien auf der entsprechenden Festplatte zu machen, da dies automatisch von OneFS SmartPools gemanagt wird, ohne die einzige Struktur zu unterbrechen. Es muss ebenfalls nicht berücksichtigt werden, wie solch eine große Struktur repliziert werden kann, denn der OneFS-SyncIQ-Service sorgt automatisch und unabhängig von der Form oder Tiefe der Dateistruktur für eine parallele Übertragung der Dateistruktur auf einen oder mehrere alternative Cluster.

Dieses Design sollte mit der Namespace-Aggregation verglichen werden. Dies ist eine allgemein verwendete Technologie, um einen herkömmlichen NAS so aussehen zu lassen, als hätte er einen einzigen Namespace. Mit der Namespace-Aggregation müssen Dateien nach wie vor in separaten Volumes verwaltet werden, aber eine einfache „Furnier“-Ebene ermöglicht das „Verkleben“ einzelner Verzeichnisse in Volumes über symbolische Links mit einer „obersten“ Struktur. In diesem Modell sind nach wie vor LUNs und Volume-Begrenzungen vorhanden. Im Hinblick auf den Lastenausgleich müssen die Dateien manuell von Volume zu Volume verschoben werden. AdministratorInnen müssen die Dateistruktur umsichtig gestalten. Das Tiering ist bei Weitem nicht nahtlos, sodass erhebliche und kontinuierliche Eingriffe erforderlich sind. Beim Failover muss eine Spiegelung von Dateien zwischen Volumes erfolgen, was die Effizienz beeinträchtigt und die Kosten für Anschaffung, Strom und Kühlung erhöht. Insgesamt ist der Aufwand für AdministratorInnen bei der Namespace-Aggregation höher als der für ein einfaches herkömmliches NAS-Gerät. Daher ist ein großes Wachstum für solche Infrastrukturen nicht möglich.

## Datenlayout

In OneFS werden physische Pointer und Extents für Metadaten und Speicherdateien sowie Verzeichnismetadaten in Inodes verwendet. Logische Inodes (LINS) von OneFS sind in der Regel 512 Byte groß, sodass sie in die nativen Sektoren passen, mit denen die Mehrheit der Festplatten formatiert ist. Unterstützung wird für 8-KB-Inodes bereitgestellt, um die dichteren Klassen von Festplatten zu unterstützen, die jetzt mit 4-KB-Sektoren formatiert sind.

B-Strukturen werden häufig im Dateisystem eingesetzt und ermöglichen Skalierbarkeit auf Milliarden von Objekten und ein nahezu sofortiges Abfragen von Daten oder Metadaten. OneFS ist ein vollständig symmetrisches und hochgradig verteiltes Dateisystem. Daten und Metadaten sind immer über mehrere Hardwaregeräte redundant. Alle Daten sind anhand des Erasure Codings über die Nodes im Cluster geschützt. Dadurch entsteht ein hocheffizientes Cluster mit einem Verhältnis von mindestens 80 % Rohkapazität zur nutzbaren Kapazität auf Clustern von fünf Nodes oder mehr. Metadaten (normalerweise weniger als 1 % des Systems) werden im Cluster auf Performance und Verfügbarkeit gespiegelt. Da OneFS nicht von RAID abhängt, kann die Menge an Redundanz durch AdministratorInnen auf Datei- oder Verzeichnisebene über die Standards des Clusters hinaus ausgewählt werden. Metadaten und Sperraufgaben werden von allen Nodes gemeinsam und gleichgestellt in einer Peer-to-Peer-Architektur gemanagt. Diese Symmetrie ist der Schlüssel für diese einfache und ausfallsichere Architektur. Es gibt keinen einzigen Metadatenserver, Sperrmanager oder Gateway-Node.

Da OneFS gleichzeitig auf Blöcke von verschiedenen Geräten zugreifen muss, wird das für Daten und Metadaten verwendete Adressierungsschema auf der physischen Ebene über einen Tupel von {Node, Laufwerk, Offset} indiziert. Beispiel: Wenn 12345 eine Blockadresse für einen Block auf Festplatte 2 von Node 3 ist, wird {3,2,12345} angezeigt. Alle Metadaten innerhalb des Clusters werden zur Data Protection mehrfach gespiegelt, mindestens auf das Maß an Redundanz der zugehörigen Datei. Wenn beispielsweise eine Datei einen Löschcodeschutz von „+2n“ hat, sie also zwei gleichzeitige Ausfälle überstehen könnte, werden alle für den Dateizugriff erforderlichen Metadaten dreimal gespiegelt, sodass diese ebenfalls zwei Ausfällen standhalten könnten. Das Dateisystem ist so angelegt, dass jede Struktur grundsätzlich alle Blöcke auf allen Nodes im Cluster verwenden kann.

Andere Speichersysteme senden Daten über RAID und Volume-Managementebenen, was zu Ineffizienz beim Datenlayout und einem nicht optimierten Blockzugriff führt. OneFS steuert die Platzierung von Dateien direkt, bis hin zur Sektorebene jedes Laufwerks an einem beliebigen Ort im Cluster. Dies ermöglicht eine optimierte Datenplatzierung und optimierte I/O-Muster. Nicht erforderliche Lese-/Änderungs-/Schreibvorgänge werden vermieden. Da die Daten Datei für Datei auf Festplatten abgelegt werden, steuert OneFS die Art des Striping sowie die Redundanz des Speichersystems auf flexible Weise auf System-, Verzeichnis- und sogar Dateiebene. Bei herkömmlichen Speichersystemen müsste ein ganzes RAID-Volume einer bestimmten Performancekategorie- und Sicherungseinstellung zugewiesen werden. Beispielsweise kann eine Reihe von Festplatten für eine Datenbank in einer RAID 1+0-Struktur angeordnet werden. Dadurch lässt sich der Einsatz von Spindeln über den gesamten Speicherbestand nur schwer optimieren (da inaktive Spindeln nicht ausgeliehen werden können). Weiterhin führt dies zu unflexiblen Designs, die sich nicht auf die jeweiligen geschäftlichen Anforderungen anpassen lassen. OneFS ermöglicht jederzeit individuelles Tuning und flexible Änderungen, und zwar vollständig online.

## Dateischreibvorgänge

Die OneFS-Software wird auf allen Nodes auf gleiche Weise ausgeführt. So entsteht ein einziges Dateisystem, das über alle Nodes hinweg ausgeführt wird. Das Cluster wird nicht von einem einzelnen Node oder „Master“ gesteuert. Alle Nodes sind völlig gleichwertig.

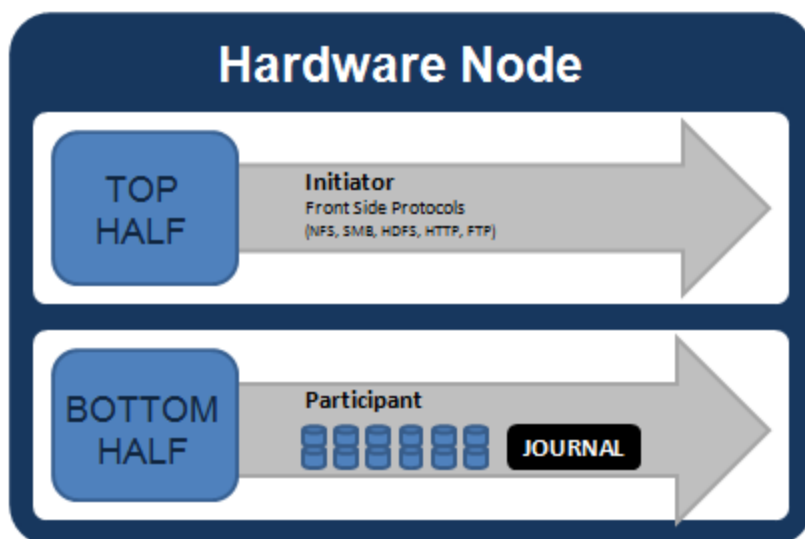


Abbildung 6: Modell von an I/O-Vorgängen beteiligten Node-Komponenten

Wenn Sie alle Komponenten innerhalb jedes Node eines Clusters in Betracht ziehen, die allgemein an I/O-Vorgängen beteiligt sind, sähe dies wie Abbildung 6 oben aus. Wir haben den Stapel in eine „oberste“ Ebene, den Initiator, und eine „untere“ Ebene, den Teilnehmer, aufgeteilt. Diese Aufteilung wird als „logisches Modell“ für die Analyse aller Lese- oder Schreibvorgänge verwendet. Auf physischer Ebene bearbeiten CPUs und RAM-Cache in den Nodes gleichzeitig die Initiator- und Teilnehmeraufgaben für I/O-Vorgänge, die im Cluster stattfinden. Es gibt Caches und einen verteilten Sperrmanager, die aus Gründen der Einfachheit nicht im Diagramm oben aufgeführt sind. Sie werden in den späteren Abschnitten dieses Dokuments behandelt.

Wenn ein Client eine Verbindung mit einem Node herstellt, um in eine Datei zu schreiben, wird eine Verbindung mit der oberen Hälfte oder dem Initiator dieses Node hergestellt. Die Dateien werden in kleinere logische Segmente namens Stripes unterteilt, bevor sie in die untere Hälfte oder den Teilnehmer eines Node (Festplatte) geschrieben werden. Durch das ausfallsichere Puffern mithilfe eines Schreib-Coalescer wird dafür gesorgt, dass die Schreibvorgänge effizient sind und Lese-Änderungs-Schreibvorgänge vermieden werden. Die Größe der einzelnen Dateisegmente wird als Größe der Stripe-Einheiten bezeichnet.

OneFS verteilt die Daten in Stripes auf alle Nodes – nicht einfach auf Festplatten – und schützt die Dateien, Verzeichnisse und die damit verbundenen Metadaten über Softwarelöschcodes oder Spiegelungstechnologie. Für Daten verwendet OneFS (je nach Ermessen der AdministratorInnen) entweder das Reed-Solomon-Erasure-Coding-System für Data Protection oder (weniger häufig) die Spiegelung. Das Anwenden von Spiegelungen auf Nutzerdaten erfolgt eher in Fällen mit hoher Transaktionsperformance. Für die meisten Nutzerdaten wird in der Regel Erasure Coding eingesetzt, da sie eine extrem hohe Performance ohne Beeinträchtigung der Festplatteneffizienz bietet. Das Erasure Coding kann mehr als 80 % Effizienz auf unformatierten Festplatten mit fünf Nodes oder mehr bereitstellen, auf großen Clustern sogar gleichzeitig mit einer Vierfachredundanz. Die Stripe-Breite einer bestimmten Datei ist die Anzahl der Nodes (nicht Laufwerke), über die eine Datei geschrieben ist. Sie wird von der Anzahl der Nodes im Cluster, der Dateigröße und der Schutzeinstellung bestimmt (z. B. +2n).

OneFS verwendet erweiterte Algorithmen, um ein Datenlayout mit maximaler Effizienz und maximalem Schutz festzulegen. Wenn ein Client eine Verbindung mit einem Node herstellt, tritt der Initiator dieses Node als „Kapitän“ für das Schreibdatenlayout dieser Datei auf. Daten, Erasure Code (ECC)-Schutz, Metadaten und Inodes sind auf mehrere Nodes innerhalb eines Clusters und sogar auf mehrere Laufwerke innerhalb der Nodes verteilt.

Abbildung 7 unten zeigt einen Dateischreibvorgang, der über alle Nodes in einem Cluster mit drei Nodes erfolgt.

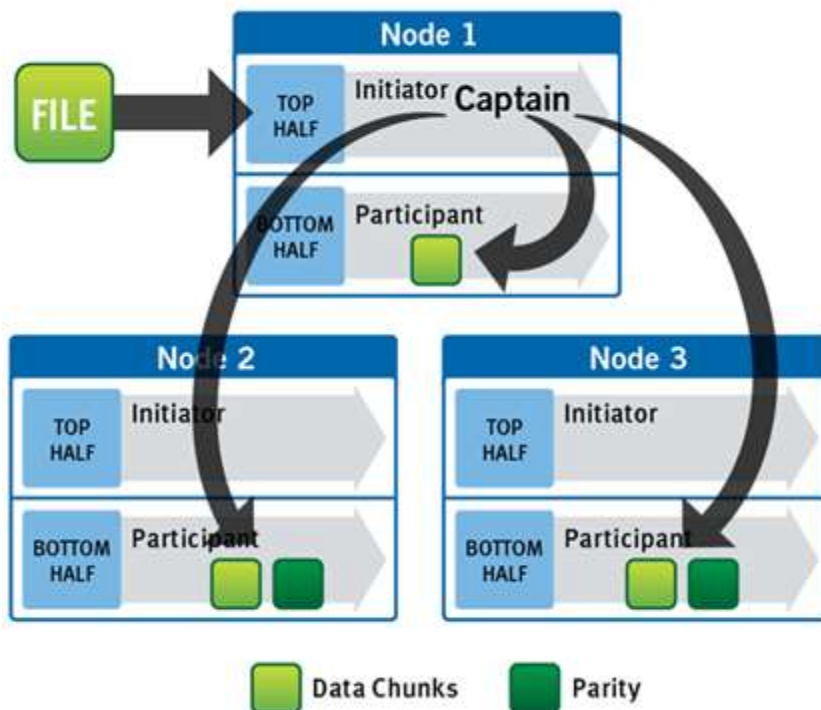


Abbildung 7: Ein Dateischreibvorgang auf einem Cluster mit drei Nodes

OneFS verwendet das Back-end-Netzwerk, um Daten automatisch auf alle Nodes im Cluster zu verteilen, sodass keine weitere Verarbeitung erforderlich ist. Wenn Daten geschrieben werden, werden sie auf der angegebenen Ebene geschützt. Wenn Schreibvorgänge stattfinden, teilt OneFS die Daten in atomare Einheiten namens Schutzgruppen auf. Schutzgruppen verfügen über integrierte Redundanz. Wenn alle Schutzgruppen geschützt sind, ist auch die gesamte Datei geschützt. Für Dateien, die mit Löschkodes geschützt sind, besteht die Schutzgruppe aus einer Reihe von Datenblöcken sowie einer Reihe von Löschkodes für diese Datenblöcke. Bei gespiegelten Dateien besteht eine Schutzgruppe aus allen Spiegelungen eines Satzes an Blöcken. OneFS ist in der Lage, die in einer Datei verwendete Schutzgruppe dynamisch während des Schreibvorgangs zu ändern. Dies ermöglicht viele zusätzliche Funktionen, z. B. kann das System angewiesen werden, in Situationen ohne Blockierung weiterzuarbeiten, in denen temporäre Node-Ausfälle im Cluster verhindern würden, dass die gewünschte Anzahl von Löschkodes verwendet wird. In solchen Fällen kann vorübergehend die Spiegelung verwendet werden, damit die Schreibvorgänge fortgesetzt werden können. Wenn Nodes im Cluster wiederhergestellt werden, werden diese gespiegelten Schutzgruppen nahtlos und automatisch, also ohne Eingriff durch den Administrator, wieder in den Löschkodeschutz umgewandelt.

Die Blockgröße im OneFS-Dateisystem beträgt 8 KB. Eine Datei, die kleiner als 8 KB ist, verwendet einen vollständigen Block von 8 KB. Je nach Data-Protection-Level verwendet diese Datei von 8 KB möglicherweise mehr als 8 KB Datenspeicherplatz. Die Einstellungen für die Data Protection werden in einem späteren Abschnitt dieses Dokuments näher behandelt. OneFS kann Dateisysteme mit Milliarden kleiner Dateien bei hoher Performance unterstützen, da alle Festplattenstrukturen für eine Skalierung auf solche Größen entwickelt wurden. Darüber hinaus bietet OneFS unabhängig von der Gesamtanzahl der Objekte einen nahezu sofortigen Zugriff auf beliebige Objekte. Bei größeren Dateien nutzt OneFS mehrere zusammenhängende Blöcke von 8 KB. In diesen Fällen können bis zu 16 zusammenhängende Blöcke auf die Festplatte eines einzigen Node verteilt werden. Wenn eine Datei eine Größe von 32 KB aufweist, werden vier zusammenhängende Blöcke von 8 KB verwendet.

Bei noch größeren Dateien kann OneFS die sequenzielle Performance maximieren, indem eine Stripe-Einheit aus 16 zusammenhängenden Blöcken eingesetzt wird, sodass sich insgesamt 128 KB pro Stripe-Einheit ergeben. Während eines Schreibvorgangs werden die Daten in Stripe-Einheiten unterteilt und als Schutzgruppe über mehrere Nodes verteilt. Während die Daten über das gesamte Cluster hinweg abgelegt werden, werden je nach Bedarf Löschkodes oder Spiegelungen verteilt, um dafür zu sorgen, dass die Dateien jederzeit geschützt sind.

Eine der wichtigsten Funktionen der in OneFS integrierten AutoBalance-Funktion besteht darin, die Daten neu zuzuweisen und abzugleichen. So wird der Speicherplatz so optimal wie möglich genutzt. In den meisten Fällen kann die Stripe-Größe von großen Dateien erhöht werden, um neuen freien Speicherplatz zu nutzen (wenn Nodes hinzugefügt werden) und das Striping auf der Festplatte effizienter zu gestalten. AutoBalance sorgt für hohe Festplatteneffizienz, wodurch „Hotspots“ auf der Festplatte automatisch eliminiert werden.

Die obere Initiatorhälfte des „Kapitän“-Node verwendet eine modifizierte Zweiphasen-Commit-Transaktion, um die Schreibvorgänge sicher auf mehrere NVRAM im Cluster zu verteilen, wie in Abbildung 8 gezeigt.

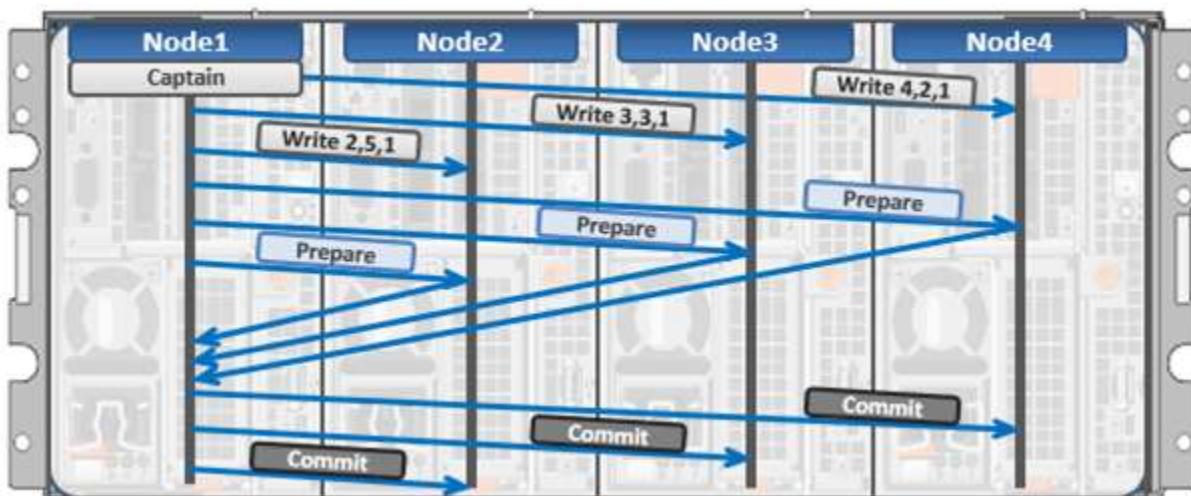


Abbildung 8: Verteilte Transaktionen und Zweiphasen-Commit-Transaktion

Jeder Node, der über Blöcke in einem bestimmten Schreibvorgang verfügt, ist an einer Zweiphasen-Commit-Transaktion beteiligt. Der Mechanismus beruht zur Protokollierung aller Transaktionen, die über jeden Node im Storage-Cluster erfolgen, auf NVRAM. Die Verwendung mehrerer NVRAM parallel ermöglicht Schreibvorgänge mit hohem Durchsatz bei gleichzeitiger Wahrung der Datensicherheit vor jeder Form von Ausfällen, einschließlich Stromausfällen. Wenn ein Node während einer Transaktion ausfällt, wird die Transaktion sofort ohne den betroffenen Node neu gestartet. Wenn der Node wieder einsatzbereit ist, muss er lediglich das entsprechende Journal von NVRAM wiedergeben – was Sekunden oder höchstens einige Minuten in Anspruch nimmt – und gelegentlich für die AutoBalance-Funktion Dateien neu abstimmen, die an der Transaktion beteiligt waren. Teure „fscck-Prozesse“ oder „Festplattenprüfprozesse“ sind in keinem Fall erforderlich. Zudem muss nie eine langwierige Neusynchronisierung durchgeführt werden. Schreibvorgänge werden nie aufgrund eines Ausfalls blockiert. Dieses patentierte Transaktionssystem ist eine der Methoden, mit denen OneFS Single-Points-of-Failure oder sogar mehrere Points-of-Failure ausschaltet.

Während eines Schreibvorgangs orchestriert der Initiator das Layout von Daten und Metadaten, die Erstellung von Erasure Codes und den Normalbetrieb für Sperrmanagement und die Berechtigungskontrolle. AdministratorInnen können über das Webmanagement oder die Befehlszeilenschnittstelle jederzeit die von OneFS getroffenen Layoutentscheidungen für den jeweiligen Workflow optimieren. AdministratorInnen können auf Datei- oder Verzeichnisebene eine Auswahl aus den unten aufgeführten Zugriffsmustern treffen:

- Nebenläufigkeit: Optimierung auf die aktuelle Last auf dem Cluster, Möglichkeit für mehrere Clients gleichzeitig. Diese Einstellung bietet das beste Verhalten für gemischte Workloads.
- Streaming: Optimierung für das Hochgeschwindigkeitsstreaming einer einzigen Datei, sodass beispielsweise sehr schnelle Lesevorgänge mit einem einzigen Client ermöglicht werden.
- Zufällig: Optimierung für unvorhersehbaren Zugriff auf die Datei durch Feinabstimmung des Striping und Deaktivieren aller vorabgerufenen Caches.

OneFS beinhaltet auch einen adaptiven Pre-Fetch in Echtzeit, der die optimale Leseperformance für Dateien mit einem erkennbaren Zugriffsmuster ohne administrativen Eingriff bereitstellt.

① Die größte Dateigröße, die OneFS derzeit unterstützt, wird in OneFS 8.2.2 und höher auf 16 TB erhöht, von maximal 4 TB in früheren Versionen.

## OneFS-Caching

Das Design der Cachinginfrastruktur in OneFS basiert auf einer Aggregation des auf jedem Node in einem Cluster vorhandenen Caches in einen global zugänglichen Speicherpool. Dazu verwendet OneFS ein effizientes Messaging-System, das dem NUMA (Non-Uniform Memory Access) ähnelt. So kann der Arbeitsspeicher aller Nodes für jeden Node im Cluster bereitgestellt werden. Der Zugriff auf Remotespeicher erfolgt über eine interne Verbindung, was zu viel geringeren Latenzen führt als bei einem Zugriff auf Festplattenlaufwerke.

Für den Remotespeicherzugriff nutzt OneFS ein redundantes, unzureichend abonniertes Flach-Ethernet-Netzwerk, das im Wesentlichen als verteilter Systembus dient. Auch wenn der Zugriff nicht so schnell wie bei lokalem Speicher ist, ist der Remotespeicherzugriff aufgrund der niedrigen Latenz von 40 GB Ethernet immer noch sehr schnell.

Das OneFS-Cachesubsystem ist im gesamten Cluster kohärent. Wenn also die gleichen Inhalte in privaten Caches mehrerer Nodes vorhanden sind, sind diese zwischengespeicherten Daten über alle Instanzen hinweg konsistent. OneFS verwendet das MESI-Protokoll, um die Cachekohärenz aufrechtzuerhalten. Dieses Protokoll implementiert eine „Invalidate-on-Write“-Policy, die Daten bei einem Schreibvorgang ungültig macht, um dafür zu sorgen, dass alle Daten im gesamten gemeinsam genutzten Cache konsistent sind.

OneFS verwendet bis zu drei Lesecachestufen sowie einen NVRAM-unterstützten Schreibcache oder Coalescer. Diese Elemente und ihre allgemeine Interaktion sind im folgenden Diagramm dargestellt.

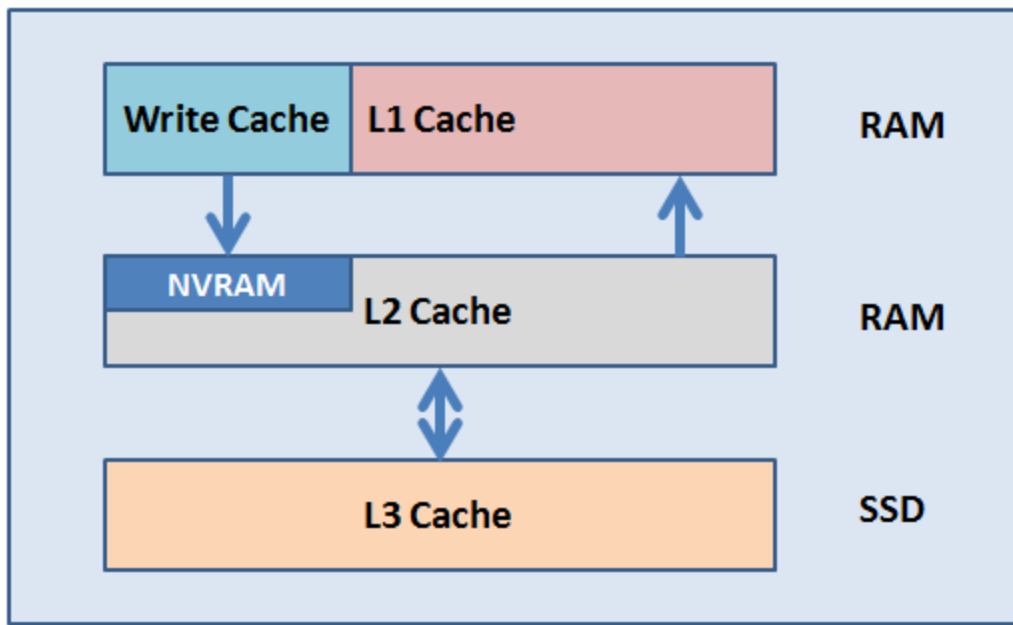


Abbildung 9: OneFS-Caching-Hierarchie

Die ersten beiden Lesecachearten, Stufe 1 (L1) und Stufe 2 (L2), sind speicherbasiert (RAM-basiert) und mit dem in Prozessoren (CPUs) verwendeten Cache vergleichbar. Diese beiden Cachestufen sind in allen Plattformen der Storage Nodes vorhanden.

Name	Typ	Persistenz	Beschreibung
<b>L1-Cache</b>	RAM	Flüchtig	Auch Front-end-Cache genannt. Der Cache enthält bereinigte, clusterkohärente Kopien von Dateisystemdaten und Metadatenblöcke, die von Clients über das Front-end-Netzwerk angefordert wurden.
<b>L2-Cache</b>	RAM	Flüchtig	Back-end-Cache. Der Cache enthält bereinigte Kopien von Dateisystemdaten und Metadaten auf einem lokalen Node.
<b>SmartCache/ Schreib-Coalescer</b>	NVRAM	Nicht flüchtig	Persistenter, batteriegestützter NVRAM-Journalcache, der alle ausstehenden Schreibvorgänge an Front-end-Dateien puffert, die nicht auf die Festplatte geschrieben wurden.



<b>SmartFlash</b> <b>L3-Cache</b>	SSD	Nicht flüchtig	Enthält filebasierte Daten und Metadatenblöcke, die aus dem L2-Cache entfernt wurden, und erhöht praktisch die Kapazität des L2-Caches.
--------------------------------------	-----	----------------	---

## OneFS-Cachekohärenz

Das OneFS-Cachesubsystem ist im gesamten Cluster kohärent. Wenn also die gleichen Inhalte in privaten Caches mehrerer Nodes vorhanden sind, sind diese zwischengespeicherten Daten über alle Instanzen hinweg konsistent. Betrachten wir beispielsweise den folgenden Anfangsstatus und die anschließende Reihenfolge der Ereignisse:

1. Node 1 und Node 5 haben jeweils eine Kopie der Daten unter einer Adresse im gemeinsam genutzten Cache.
2. Node 5 erklärt als Reaktion auf eine Schreibanforderung die Kopie in Node 1 für ungültig.
3. Node 5 aktualisiert dann den Wert. (Siehe unten).
4. Node 1 muss die Daten aus dem gemeinsam genutzten Cache erneut lesen, um den aktualisierten Wert zu erhalten.

OneFS verwendet das MESI-Protokoll, um die Cachekohärenz aufrechtzuerhalten. Dieses Protokoll implementiert eine „Invalidate-on-Write“-Policy, die Daten bei einem Schreibvorgang ungültig macht, um dafür zu sorgen, dass alle Daten im gesamten gemeinsam genutzten Cache konsistent sind. Die folgende Abbildung zeigt die verschiedenen Status, die im Cache befindliche Daten aufweisen können, und die Übergänge dazwischen. Die verschiedenen Status in der Abbildung lauten:

- M – Modifiziert: Die Daten existieren nur im lokalen Cache und wurden im Vergleich zum Wert im gemeinsamen Cache geändert. Geänderte Daten werden in der Regel als „unbereinigt“ bezeichnet.
- E – Exklusiv: Die Daten existieren nur im lokalen Cache, stimmen aber mit dem gemeinsamen Cache überein. Diese Daten werden oft als „bereinigt“ bezeichnet.
- S – Freigegeben: Die Daten im lokalen Cache können auch in anderen lokalen Caches im Cluster vorhanden sein.
- I – Ungültig: Eine Sperre (exklusiv oder freigegeben) der Daten ist verloren gegangen.

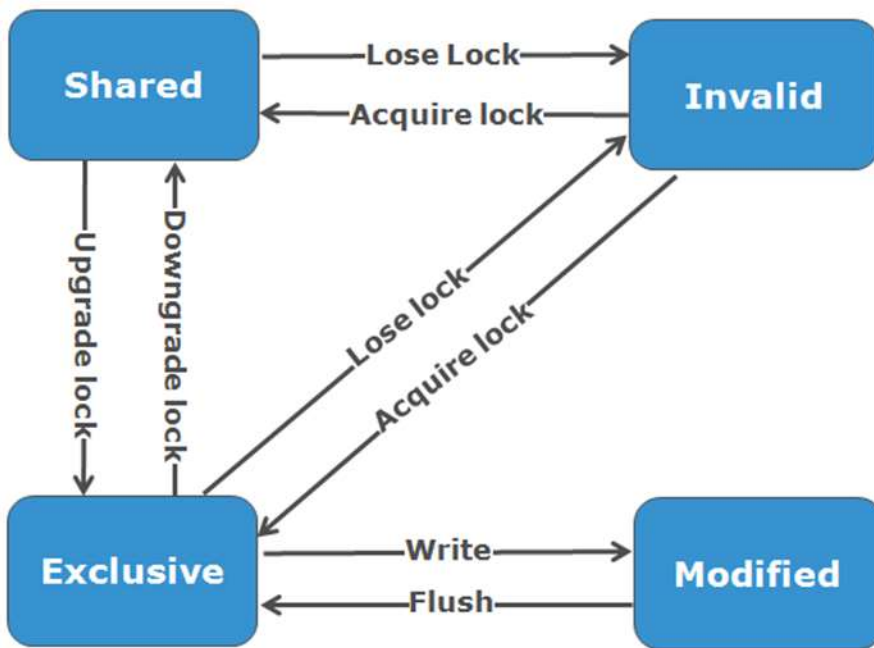


Abbildung 10: Statusdiagramm der OneFS-Cachekohärenz

## Level-1-Cache

Der Level-1-Cache (L1) oder Front-end-Cache ist der Arbeitsspeicher, der sich am nächsten an den Protokollschichten (z. B. NFS, SMB usw.) befindet, die von Clients oder Initiatoren verwendet werden, die mit diesem Nodes verbunden sind. Der Hauptzweck des L1-Caches besteht im Vorabrufen der Daten aus den Remote-Nodes. Die Daten werden pro Datei vorabgerufen, und dies wird optimiert, um die Latenz des Back-end-Netzwerks der Nodes zu reduzieren. Da die Back-end-Verbindungslatenz relativ gering ist, sind die Größe des L1-Caches und die typische pro Anforderung gespeicherte Datenmenge geringer als beim L2-Cache.

L1 wird auch als Remotecache bezeichnet, da er Daten enthält, die von anderen Nodes im Cluster abgerufen wurden. Er ist im gesamten Cluster kohärent, wird aber nur von dem Node verwendet, auf dem er sich befindet, und kann nicht von anderen Nodes abgerufen werden. Die Daten im L1-Cache auf Storage Nodes werden offensiv verworfen, nachdem sie verwendet wurden. Der L1-Cache verwendet eine dateibasierte Adressierung, bei der die Daten über ein Offset in ein Dateiojekt abgerufen werden.

Der L1-Cache bezieht sich auf Arbeitsspeicher auf demselben Node wie der Initiator. Er ist nur für den lokalen Node zugänglich und enthält in der Regel keine Masterkopie der Daten. Dies entspricht dem L1-Cache auf einem CPU-Kern, der ungültig werden kann, wenn andere Kerne in den Hauptspeicher schreiben.

Die Kohärenz des L1-Caches wird über ein MESI-ähnliches Protokoll mit verteilten Sperren verwaltet, wie oben beschrieben.

OneFS verwendet zudem einen dedizierten Inode-Cache, in dem kürzlich angeforderte Inodes aufbewahrt werden. Der Inode-Cache hat häufig große Auswirkungen auf die Performance, da Clients häufig Daten zwischenspeichern und viele I/O-Aktivitäten im Netzwerk in erster Linie Anforderungen von Dateiattributen und Metadaten sind, die schnell vom zwischengespeicherten Inode zurückgegeben werden können.

① L1-Cache wird in den Cluster Accelerator Nodes anders verwendet, da diese keine Festplattenlaufwerke enthalten. Stattdessen ist der gesamte Lesecache L1-Cache, da alle Daten von anderen Storage Nodes abgerufen werden. Außerdem basiert die Cachealterung auf einer LRU-Leerungs-Policy (Least Recently Used) für am seltensten verwendete Elemente, im Gegensatz zum Rückstandsalgorithmus, der typischerweise in der Regel im L1-Cache eines Storage Node verwendet wird. Da der L1-Cache eines Accelerator groß ist und die darin enthaltenen Daten viel wahrscheinlicher erneut angefordert werden, werden Datenblöcke nicht unmittelbar nach der Verwendung aus dem Cache gelöscht. Allerdings profitieren Metadaten und upgradeintensive Workloads nicht so stark davon, und der Cache eines Accelerator ist nur für Clients sinnvoll, die direkt mit dem Node verbunden sind.

## Level-2-Cache

Der Level-2-Cache (L2) oder Back-end-Cache bezieht sich auf lokalen Arbeitsspeicher auf dem Node, in dem ein bestimmter Datenblock gespeichert wird. L2-Cache ist global von jedem Node im Cluster zugänglich und wird verwendet, um die Latenz eines Lesevorgangs zu senken, da keine Suche direkt von den Festplattenlaufwerken erforderlich ist. Daher ist die Menge der Daten, die zur Verwendung durch Remote-Nodes in den L2-Cache vorabgerufen werden, deutlich höher als die im L1-Cache.

L2-Cache wird auch als lokaler Cache bezeichnet, da er die Daten enthält, die von Festplattenlaufwerken abgerufen wurden, die sich auf diesem Node befinden. Diese werden dann für Anforderungen von Remote-Nodes zur Verfügung gestellt. Die Daten im L2-Cache werden gemäß einem LRU-Algorithmus (Least Recently Used) geleert.

Die Daten im L2-Cache werden vom lokalen Node mithilfe eines Offset in ein Festplattenlaufwerk abgerufen, das für diesen Node lokal ist. Da der Node weiß, wo sich die von den Remote-Nodes angeforderten Daten auf der Festplatte befinden, ist dies eine sehr schnelle Methode zum Abrufen von Daten für Remote-Nodes. Ein Remote-Node greift auf den L2-Cache zu, indem er eine Suche nach der Blockadresse für ein bestimmtes Dateiojekt durchführt. Wie oben beschrieben, ist hier keine MESI-Ungültigerklärung erforderlich, und der Cache wird automatisch während der Schreibvorgänge aktualisiert und über das Transaktionssystem und den NVRAM kohärent gehalten.

## Level-3-Cache

Eine optionale dritte Lesecachestufe namens SmartFlash- oder Stufe-3-Cache (L3) ist ebenfalls auf Nodes konfigurierbar, die SSDs (Solid-State-Laufwerke) enthalten. SmartFlash (L3) ist ein Leerungscache, der durch die vom Speicher gelöschten L2-Cacheblöcke aufgefüllt wird. Es gibt einige Vorteile bei der Verwendung von SSDs für das Caching anstelle von herkömmlichen Dateisystem-Speichergeräten. Beispiel: Wenn ein SSD für das Caching reserviert ist, wird das gesamte SSD verwendet, und die Schreibvorgänge erfolgen auf sehr lineare und zuverlässige Art und Weise. Dies bietet eine weitaus bessere Auslastung und führt ebenfalls zu deutlich geringerer Abnutzung und einer höheren Haltbarkeit bei regelmäßiger Dateisystemnutzung, insbesondere bei zufälligen Schreib-Workloads. Im Vergleich zu einer Verwendung von SSDs als Storage Tier wird mit dem Einsatz von SSDs für den Cache auch die Dimensionierung der SSD-Kapazität sehr viel überschaubarer und weniger fehleranfällig.

Das folgende Diagramm zeigt, wie Clients mit der OneFS-Lesecacheinfrastruktur und dem Schreib-Coalescer interagieren. Der L1-Cache interagiert weiterhin auf allen benötigten Nodes mit dem L2-Cache, und der L2-Cache interagiert mit dem Speichersubsystem und dem L3-Cache. Der L3-Cache wird innerhalb des Node auf einem SSD gespeichert, und auf jedem Node im selben Node-Pool ist der L3-Cache aktiviert.

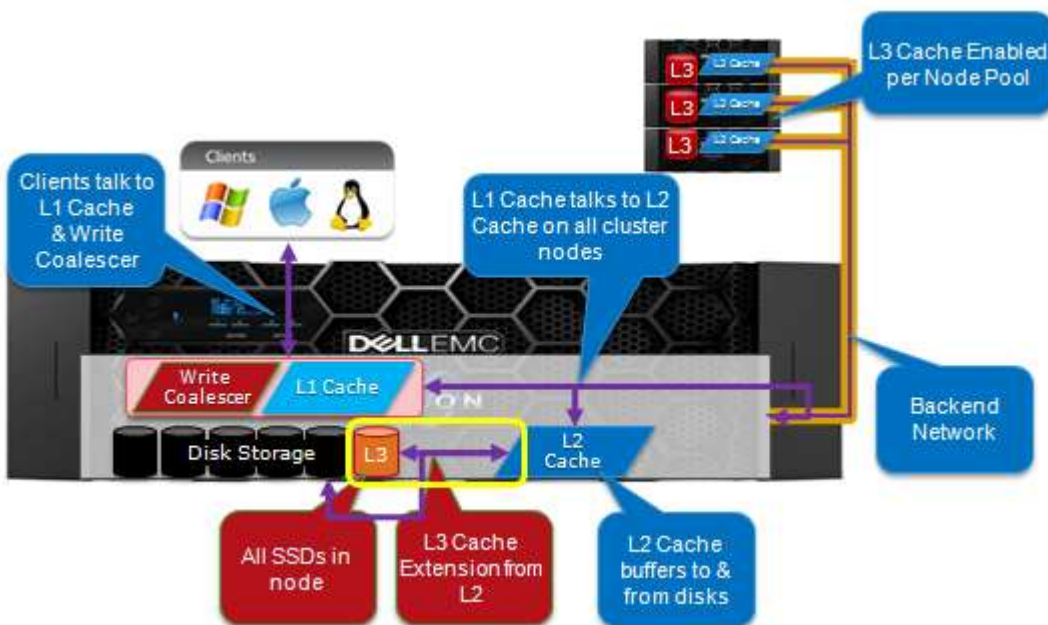


Abbildung 11: Architektur des L1-, L2- und L3-Caches in OneFS

OneFS schreibt vor, dass eine Datei über mehrere Nodes im Cluster und möglicherweise mehrere Laufwerke in einem Node geschrieben wird. Daher werden bei allen Leseanforderungen Remotedaten (und möglicherweise lokale Daten) gelesen. Wenn eine Leseanforderung von einem Client eintrifft, bestimmt OneFS, ob die angeforderten Daten im lokalen Cache enthalten sind. Alle Daten, die sich im lokalen Cache befinden, werden sofort gelesen. Befinden sich die angeforderten Daten nicht im lokalen Cache, werden diese von der Festplatte gelesen. Für Daten, die sich nicht auf dem lokalen Node befinden, wird eine Anforderung von den Remote-Nodes gestellt, auf denen sie sich befinden. Auf jedem der anderen Nodes wird eine weitere Cacheabfrage durchgeführt. Alle Daten im Cache werden sofort zurückgegeben, und alle Daten, die sich nicht im Cache befinden, werden von der Festplatte abgerufen.

Wenn die Daten vom lokalen und Remotecache (und ggf. der Festplatte) abgerufen wurden, werden sie an den Client zurückgegeben.

Die allgemeinen Schritte für die Erfüllung einer Leseanforderung auf einem lokalen und Remote-Node lauten:

Auf dem lokalen Node (der Node, der die Anforderung empfängt):

1. Bestimmen, ob sich ein Teil der angeforderten Daten im lokalen L1-Cache befindet. Falls ja, Rückgabe an den Client.
2. Wenn die Daten sich nicht im lokalen Cache befinden, Daten von den Remote-Node anfordern.

Auf Remote-Nodes:

1. Bestimmen, ob sich die angeforderten Daten im lokalen L2- und L3-Cache befinden. Falls ja, Rückgabe an den anfordernden Node.
2. Wenn die Daten sich nicht im lokalen Cache befinden, von der Festplatte lesen und an den anfordernden Node zurückgeben.

Schreibcaching beschleunigt den Prozess des Schreibens von Daten in ein Cluster. Dies wird dadurch erreicht, dass kleinere Schreibenforderungen in einem Batch zusammengefasst und in größeren Segmenten an die Festplatte gesendet werden, was einen Großteil der Latenz beim Schreiben auf Festplatten vermeidet. Wenn Clients an das Cluster schreiben, schreibt OneFS die Daten vorübergehend in einen NVRAM-basierten Journalcache auf dem Initiator-Node statt sofort auf die Festplatte. OneFS kann diese zwischengespeicherten Schreibvorgänge dann zu einem späteren, passenderen Zeitpunkt an die Festplatte weiterleiten. Darüber hinaus werden diese Schreibvorgänge auch auf den NVRAM-Journalen der Teilnehmer-Nodes gespiegelt, um die Anforderungen für die Dateisicherung zu erfüllen. Im Falle einer Clusterteilung oder eines unerwarteten Node-Ausfalls sind daher selbst nicht übernommene zwischengespeicherte Schreibvorgänge vollständig geschützt.

Der Schreibcache arbeitet wie folgt:

- Ein NFS-Client sendet eine Schreibenforderung für eine Datei mit +2n-Schutz an Node 1.
- Node 1 nimmt die Schreibvorgänge in seinen NVRAM-Schreibcache auf (schneller Pfad) und spiegelt dann die Schreibvorgänge in die Protokolldateien der Teilnehmer-Nodes, um sie zu schützen.
- Bestätigungen für die Schreibvorgänge werden sofort an den NFS-Client zurückgegeben, sodass die Latenz beim Schreiben auf Festplatten vermieden wird.
- Der nach und nach anwachsende Schreibcache von Node 1 wird regelmäßig geleert und die Schreibvorgänge werden unter Anwendung des entsprechenden Löscheschutzes (ECC) (+2n) über den Zweiphasen-Commit-Prozess (siehe Beschreibung weiter oben) auf die Festplatte übertragen.
- Der Schreibcache und die Protokolldateien des Teilnehmer-Node werden geleert und können neue Schreibvorgänge aufnehmen.

 Weitere Informationen finden Sie im Whitepaper [OneFS SmartFlash](#).

## Lesen von Dateien

Daten, Metadaten und Inodes sind auf mehreren Nodes innerhalb eines Clusters und sogar auf mehreren Laufwerken innerhalb der Nodes verteilt. Beim Lesen oder Schreiben in das Cluster fungiert der Node, an den ein Client angebunden ist, als „Kapitän“ für den Vorgang.

Bei einem Lesevorgang erfasst der „Kapitän“-Node alle Daten aus den verschiedenen Nodes im Cluster und stellt sie der anfordernden Stelle auf zusammenhängende Weise bereit.

Dank kostenoptimierter Hardware nach Branchenstandard bietet das Cluster bei der Übertragung vom Cache auf die Festplatte ein hohes Verhältnis (mehrere GB pro Node), das den Lese- und Schreibvorgängen je nach Bedarf dynamisch zugewiesen wird. Dieser RAM-basierte Cache ist über alle Nodes im Cluster einheitlich und kohärent, sodass eine Leseanforderung von einem Client an einen bestimmten Node die bereits an einen anderen Node weitergeleiteten I/O-Vorgänge nutzen kann. Diese zwischengespeicherten Blöcke sind über die Rückwandplatine mit niedriger Latenz von beliebigen Nodes aus schnell zugänglich, was einen großen, effiziente RAM-Cache und somit eine erheblich schnellere Leseperformance ermöglicht.

Je mehr das Cluster anwächst, desto größer werden die Vorteile beim Caching. Aus diesem Grund ist die Anzahl der I/O-Schreibvorgänge an die Festplatte in einem Cluster in der Regel deutlich niedriger als auf herkömmlichen Plattformen, was zu niedrigerer Latenz und einem besseren Nutzererlebnis führt.

Für Dateien, die mit einem Zugriffsmuster von „gleichzeitig“ oder „Streaming“ gekennzeichnet sind, kann OneFS die Vorteile des Vorabrufs von Daten auf der Grundlage von Heuristiken nutzen, die von der SmartRead-Komponente verwendet werden. SmartRead kann eine „Datenpipeline“ aus dem L2-Cache lesen und über einen lokalen L1-Cache auf dem „Kapitän“-Node vorabrufen. Dadurch verbessert sich die sequenzielle Leseperformance über alle Protokolle hinweg erheblich und die Lesevorgänge werden innerhalb von Millisekunden direkt vom RAM bereitgestellt. In Fällen mit hohem sequenziellem Zugriff kann SmartRead eine äußerst große Datenmenge vorabrufen, sodass Lese- oder Schreibvorgänge für einzelne Dateien mit extrem hohen Datenraten erfolgen können.

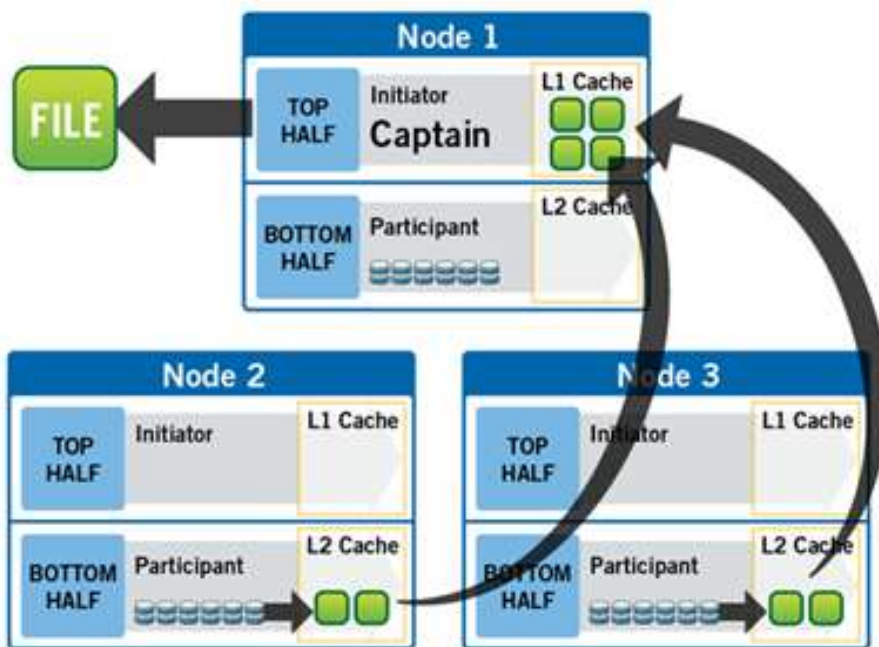


Abbildung 12: Ein Dateilesevorgang auf einem Cluster mit 3 Nodes

Abbildung 10 zeigt, wie SmartRead eine nicht im Cache gespeicherte Datei mit sequenziellem Zugriff liest, die Datei von einem Client angefordert wird, der in einem Cluster mit drei Nodes an Node 1 angebunden ist.

1. Node 1 liest Metadaten, um zu ermitteln, an welchem Ort sich die verschiedenen Blöcke von filebasierten Daten befinden.
2. Node 1 prüft außerdem seinen L1-Cache, um festzustellen, ob die angeforderten filebasierten Daten enthalten sind.
3. Node 1 erstellt eine Lesepipeline und sendet gleichzeitig Anforderungen an alle Nodes, die einen Teil der filebasierten Daten enthalten, um diese filebasierten Daten von der Festplatte abzurufen.
4. Die einzelnen Nodes rufen die Blöcke der filebasierten Daten von der Festplatte in ihren L2-Cache (oder, falls verfügbar, in den L3-SmartFlash-Cache) ab und übertragen die filebasierten Daten an Node 1.
5. Node 1 speichert die eingehenden Daten im L1-Cache und stellt die Datei gleichzeitig für den Client bereit. Dabei wird der Pre-Fetch-Prozess fortgesetzt.
6. In Fällen mit hohem sequenziellem Zugriff werden die Daten im L1-Cache optional „zurückgelassen“, um RAM für andere L1- oder L2-Cacheanforderungen freizugeben.

Das intelligente Caching mit SmartReads ermöglicht eine sehr hohe Leseperformance mit einem hohen gleichzeitigen Zugriff. Noch wichtiger ist, dass Node 1 die filebasierten Daten vom Node 2-Cache (über den Cluster-Interconnect mit niedriger Latenz) schneller abrufen kann, als auf die eigene lokale Festplatte zuzugreifen. Die Algorithmen in SmartReads steuern, wie aggressiv der Pre-Fetch-Vorgang ausgeführt wird (Deaktivierung von Pre-Fetch für zufälligen Zugriff) und wie lange die Daten im Cache verbleiben. Außerdem optimieren sie das Caching von Daten, indem sie den entsprechenden Cache auswählen.

## Sperrungen und gleichzeitiger Zugriff

OneFS verfügt über einen vollständig verteilten Sperrmanager, mit dem die Sperren für Daten in allen Nodes in einem Cluster gesteuert werden können. Der Sperrmanager ist hochgradig erweiterbar und ermöglicht unterschiedliche „Sperrarten“, um sowohl Dateisystemsperren als auch clusterkohärentes Sperren auf Protokollebene wie SMB-Freigabesperren- oder wahlfreies NFS-Sperren zu unterstützen. Darüber hinaus bietet OneFS Support für delegierte Sperren wie CIFS Oplocks und NFSv4-Delegierungen.

Jeder Node in einem Cluster ist ein Koordinator für das Sperren von Ressourcen. Ein Koordinator wird sperrbaren Ressourcen auf Grundlage eines erweiterten Hashing-Algorithmus zugewiesen. Der Algorithmus wurde so entwickelt, ist, dass der Koordinator sich am Ende fast immer auf einem anderen Node befindet, als der Initiator der Anforderung. Wenn eine Sperre für eine Datei angefordert wird, kann es sich um eine gemeinsam genutzte Sperre handeln (sodass mehrere NutzerInnen die Sperre gleichzeitig nutzen können, in der Regel für Lesevorgänge) oder eine exklusive Sperre (nur jeweils ein Nutzer, in der Regel für Schreibvorgänge).

Abbildung 13 zeigt ein Beispiel dafür, wie Threads von verschiedenen Nodes eine Sperre vom Koordinator anfordern können.

1. Node 2 wurde als Koordinator dieser Ressourcen festgelegt.
2. Thread 1 von Node 4 und Thread 2 von Node 3 fordern gleichzeitig eine gemeinsame Sperre für eine Datei von Node 2 an.
3. Node 2 überprüft, ob eine exklusive Sperre für die angeforderte Datei vorhanden ist.
4. Wenn keine exklusive Sperre besteht, gewährt Node 2 Thread 1 von Node 4 und Thread 2 von Node 3 gemeinsame Sperren für die angeforderte Datei.
5. Node 3 und Node 4 führen jetzt einen Lesevorgang für die angeforderte Datei durch.
6. Thread 3 von Node 1 fordert eine exklusive Sperre für dieselbe Datei an, die von Node 3 und Node 4 gelesen wird.
7. Node 2 überprüft bei Node 3 und Node 4, ob die gemeinsame Sperre aufgehoben werden kann.
8. Node 3 und Node 4 haben den Lesevorgang noch nicht abgeschlossen, sodass Node 2 Thread 3 von Node 1 auffordert, einen Augenblick zu warten.
9. Thread 3 auf Node 1 wird so lange blockiert, bis die exklusive Sperre durch Node 2 vergeben wird, und schließt den Schreibvorgang ab.

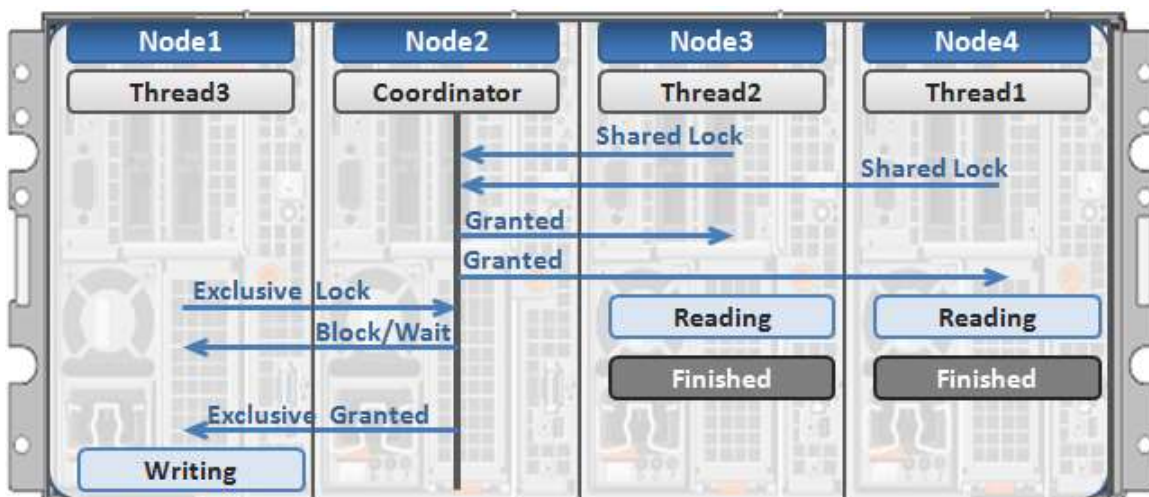


Abbildung 13: Verteilter Sperrmanager

## I/O-Vorgänge mit Multithreading

Angesichts der zunehmenden Nutzung riesiger NFS-Datstores für die Servervirtualisierung und den Support von Anwendungen der Enterprise-Klasse sind für große Dateien ein hoher Durchsatz und geringe Latenz erforderlich. Aus diesem Grund unterstützt der OneFS Multi-Writer mehrere Threads gleichzeitig für Schreibvorgänge auf einzelne Dateien.

Im obigen Beispiel kann der gleichzeitige Schreibzugriff auf eine große Datei durch den exklusiven, auf der gesamten Dateiebene angewendeten Sperrmechanismus eingeschränkt werden. Um diesen potenziellen Engpass zu vermeiden, bietet der OneFS Multi-Writer eine feiner abgestimmte Schreibsperre, bei der nicht die gesamte Datei gesperrt, sondern die Datei in separate Bereiche unterteilt wird, denen dann exklusive Schreibsperren zugewiesen werden. Auf diese Weise können mehrere Clients gleichzeitig Schreibvorgänge für unterschiedliche Bereiche derselben Datei durchführen.

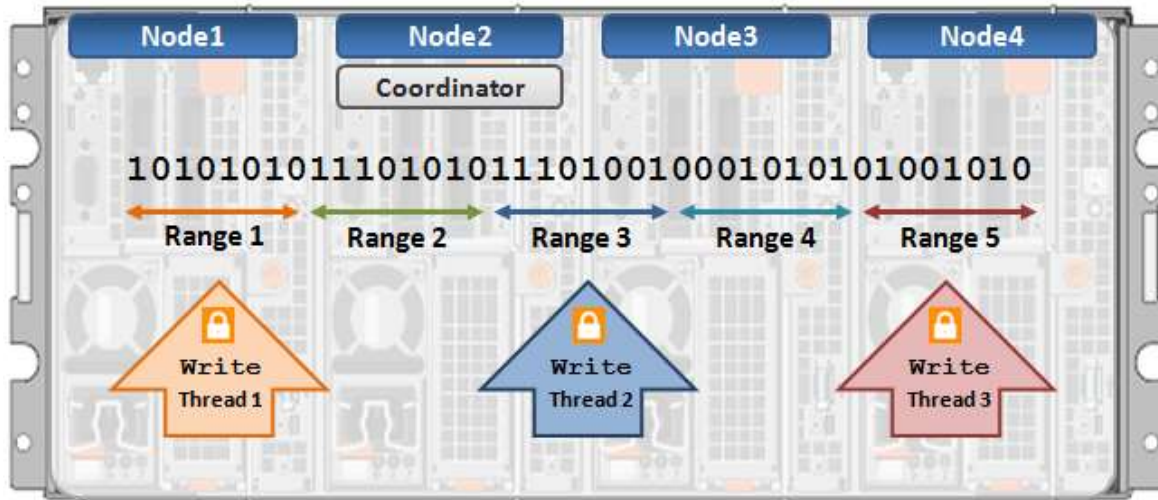


Abbildung 14: IO-Writer mit Multithreading

## Data Protection

### Stromausfall

Ein Dateisystemjournal, in dem Informationen zu Änderungen am Dateisystem gespeichert werden, ist auf eine schnelle, konsistente Recovery nach Systemausfällen oder Abstürzen ausgelegt, z. B. bei einem Stromausfall. Das Dateisystem gibt die Journaleinträge erneut wieder, wenn ein Node oder Cluster nach einem Stromausfall oder einem anderen Ausfall wiederhergestellt wird. Ohne ein Journal müsste das Dateisystem nach einem Ausfall alle potenziellen Änderungen einzeln untersuchen und überprüfen („fsck“- oder „chkdsk“-Vorgang), was in einem großen Dateisystem viel Zeit in Anspruch nehmen kann.

OneFS ist ein Journaldateisystem, in dem jeder Node eine batteriegestützte NVRAM-Karte enthält, mit der nicht in das Dateisystem übernommene Schreibvorgänge geschützt werden. Die Batterieladung der NVRAM-Karte reicht für mehrere Tage, ohne dass ein Aufladen erforderlich ist. Wenn ein Node gestartet wird, prüft er sein Journal und gibt selektiv Transaktionen an die Festplatten wieder, wenn das Journalsystem dies für erforderlich hält.

OneFS wird nur dann gemountet, wenn sichergestellt werden kann, dass alle noch nicht im System gespeicherten Transaktionen aufgezeichnet wurden. Wenn beispielsweise nicht die richtigen Verfahren zum Herunterfahren eingehalten wurden und die NVRAM-Batterie entladen wurde, sind möglicherweise Transaktionen verloren gegangen. Um potenzielle Probleme zu vermeiden, wird das Dateisystem nicht durch den Node gemountet.

### Hardwarefehler und Quorum

Damit das Cluster korrekt funktioniert und Datenschreibvorgänge akzeptieren kann, muss ein Quorum von Nodes aktiv sein und reagieren. Ein Quorum wird als einfache Mehrheit definiert: ein Cluster mit Nodes muss  $\lfloor 2 \rfloor + 1$  Nodes online haben, um Schreibvorgänge zu ermöglichen. Beispiel: In einem Cluster mit sieben Nodes sind vier Nodes für ein Quorum erforderlich.

Wenn ein Node oder eine Gruppe von Nodes aktiv und reaktionsfähig, aber nicht Mitglied eines Quorums ist, wird der Node oder die Gruppe von Nodes in einen schreibgeschützten Status versetzt.

OneFS verwendet ein Quorum, um „Split-Brain“-Bedingungen zu verhindern, die entstehen können, wenn das Cluster vorübergehend in zwei Cluster aufgeteilt wird. Durch Befolgen der Quorum-Regel sorgt die Architektur dafür, dass immer ein Schreibvorgang stattfindet, unabhängig davon, wie viele Nodes ausfallen oder wieder online sind, und dass der Schreibvorgang auf alle vorherigen Schreibvorgänge abgestimmt werden kann. Das Quorum gibt außerdem die Anzahl der Nodes vor, die zum Umstieg auf ein bestimmtes Data-Protection-Level erforderlich sind. Für das auf Erasure-Codes-basierende Schutzlevel von + muss der Cluster mindestens 2+1 Nodes umfassen. Beispielsweise sind für eine +3n-Konfiguration mindestens sieben Nodes erforderlich. So kann bei einem gleichzeitigen Verlust von drei Nodes auch weiterhin das Quorum von vier Nodes aufrechterhalten werden, damit der Cluster voll funktionsfähig bleibt. Wenn ein Cluster unter das Quorum fällt, wird das Dateisystem automatisch in einen gesicherten schreibgeschützten Status versetzt. Schreibvorgänge sind nicht mehr möglich, aber die verfügbaren Daten können nach wie vor gelesen werden.

### **Hardwarefehler – Hinzufügen/Entfernen von Nodes**

Ein System namens Gruppenmanagementprotokoll (GMP) ermöglicht jederzeit globale Kenntnisse des Clusterstatus und ermöglicht eine konsistente Ansicht des Status aller anderen Nodes über das gesamte Cluster. Wenn ein oder mehrere Nodes im gesamten Cluster-Interconnect nicht erreichbar sind, wird die Gruppe vom Cluster „geteilt“ bzw. aus dem Cluster entfernt. Alle Nodes werden in einer neuen konsistenten Ansicht ihres Clusters aufgelöst. (Stellen Sie sich das so vor, als ob der Cluster in zwei separate Node-Gruppen aufgeteilt worden wäre, aber beachten Sie, dass nur eine Gruppe das Quorum haben kann.) In diesem Split-Status sind alle Daten im Dateisystem erreichbar und in der Gruppe, die das Quorum enthält, änderbar. Alle auf dem „nachgeschalteten“ Gerät gespeicherten Daten werden mithilfe der im Cluster integrierten Redundanz wiederhergestellt.

Wenn der Node wieder verfügbar ist, erfolgt ein „Zusammenführen“ oder Hinzufügen, um die Nodes zurück in das Cluster zu bringen. (Die beiden Gruppen werden wieder zu einer Gruppe zusammengeführt.) Der Node kann dem Cluster wieder beitreten, ohne wiederhergestellt und neu konfiguriert zu werden. Dies steht im Gegensatz zu RAID-Arrays für Hardware, bei denen die Laufwerke wiederhergestellt werden müssen. Die AutoBalance-Funktion führt möglicherweise zur Steigerung der Effizienz für bestimmte Dateien ein erneutes Striping durch, wenn einige ihrer Schutzgruppen überschrieben und während der Teilung auf einen niedrigeren Stripe transformiert wurden.

Die OneFS-Job-Engine umfasst auch einen Prozess namens „Collect“, der alle verwaisten Blöcke aufsammelt. Wenn ein Cluster während eines Schreibvorgangs geteilt wird, müssen möglicherweise einige der Datei zugewiesenen Blöcke auf der Quorum-Seite neu zugewiesen werden. Dadurch entstehen „verwaiste“ zugeordnete Blöcke auf der Seite, die nicht das Quorum hält. Wenn das Cluster wieder zusammengeführt wird, sucht der Job „Collect“ diese verwaisten Blöcke über einen parallelen Mark-and-Sweep-Scan, sodass sie als freier Speicherplatz für das Cluster zurückgewonnen werden können.

### **Skalierbare Wiederherstellung**

OneFS nutzt für die Zuweisung oder Wiederherstellung von Daten nach einem Ausfall kein Hardware-RAID. Stattdessen wird der Schutz von filebasierten Daten in OneFS direkt gemanagt. Bei einem Ausfall stellt die Software die Daten auf parallele Weise wieder her. OneFS kann zeitlich konstant bestimmen, welche Dateien von einem Ausfall betroffen sind, indem die Inode-Daten auf lineare Weise direkt von der Festplatte gelesen werden. Der betroffene Dateisatz wird einem Satz Worker Threads zugewiesen, die von der Job-Engine auf die Cluster-Nodes verteilt werden. Die Worker Nodes reparieren die Dateien auf parallele Weise. Mit zunehmender Clustergröße verringert sich somit die Zeit für die Wiederherstellung nach Ausfällen. Dadurch steigert sich die Effizienz enorm, da die Ausfallsicherheit von Clustern auch dann beibehalten wird, wenn ihre Größe ansteigt.

### **Virtuelle Hot Spares**

Für die meisten herkömmlichen Speichersysteme, die auf RAID basieren, ist das Provisioning eines oder mehrerer Hot-Spare-Laufwerke erforderlich, um eine unabhängige Recovery fehlgeschlagener Laufwerke zu ermöglichen. Das Hot-Spare-Laufwerk ersetzt das fehlgeschlagene Laufwerk in einem RAID-Set. Wenn diese Hot Spares nicht ersetzt werden, bevor weitere Ausfälle stattfinden, besteht das Risiko eines katastrophalen Datenverlusts im System. OneFS vermeidet die Verwendung von Hot-Spare-Laufwerken und borgt für eine Recovery nach einem Ausfall einfach verfügbaren freien Speicherplatz vom System aus. Diese Methode wird als virtueller Hot Spare bezeichnet. Auf diese Weise wird eine automatische Fehlerkorrektur des Clusters ohne menschliches Eingreifen ermöglicht. AdministratorInnen können eine virtuelle Hot-Spare-Reserve schaffen, sodass auch bei laufenden Schreibvorgängen durch die EndnutzerInnen eine automatische Fehlerkorrektur durchgeführt werden kann.



## Data Protection auf Dateiebene mit Erasure Coding

Ein Cluster ist so ausgelegt, dass ein oder mehrere gleichzeitige Komponentenausfälle möglich sind, ohne dass sich dies auf die Datenbereitstellung durch das Cluster auswirkt. Zu diesem Zweck setzt OneFS für den Schutz von Dateien einen auf Erasure-Code-basierenden Schutz, die Reed-Solomon-Fehlerkorrektur (N+M-Schutz) oder ein Spiegelungssystem ein. Die Data Protection wird in der Software auf Dateiebene angewendet. So kann sich das System ganz auf die Wiederherstellung der Dateien konzentrieren, die durch den Ausfall infiziert sind, statt eine gesamte Dateigruppe oder ein gesamtes Volume zu prüfen und zu reparieren. OneFS-Metadaten und Inodes werden immer durch Spiegelung und nicht durch die Reed-Solomon-Codierung geschützt, und zwar mit mindestens dem Sicherheitslevel wie die Daten, auf die sie sich beziehen.

Da alle Daten, Metadaten und Schutzinformationen über die Nodes des Clusters verteilt sind, benötigt ein Cluster zum Managen von Metadaten keinen dedizierten Paritäts-Node und kein dediziertes Laufwerk bzw. kein dediziertes Gerät oder keinen Satz von Geräten. Damit wird verhindert, dass ein einzelner Node einen Single-Point-of-Failure darstellt. Die auszuführenden Aufgaben werden gleichmäßig auf alle Nodes verteilt, sodass eine perfekte Symmetrie und ein perfekter Lastenausgleich in einer Peer-to-Peer Architektur entsteht.

OneFS bietet mehrere Level von konfigurierbaren Data-Protection-Einstellungen, die Sie jederzeit ändern können, ohne das Cluster oder Dateisystem offline nehmen zu müssen.

Für eine mit Löschkodes geschützte Datei gilt beispielsweise, dass jede ihrer Schutzgruppen auf Level  $N+M/b$  geschützt wird, wobei  $N > M$  und  $M \geq b$ . Die Werte  $N$  und  $M$  stehen jeweils für die Anzahl der Laufwerke, die innerhalb der Schutzgruppe für Daten und für Löschkodes verwendet wird. Der Wert „b“ bezieht sich auf die Anzahl der Daten-Stripes, auf die die Schutzgruppe abgelegt wurde, und wird nachfolgend erläutert. Ein üblicher und leicht verständlicher Fall ist  $b=1$ , was bedeutet, dass eine Schutzgruppe  $N$  Laufwerke mit Daten und  $M$  Laufwerke mit Redundanz umfasst, die in Löschkodes gespeichert sind, und dass die Schutzgruppe über genau einen Stripe über eine Reihe von Nodes gelegt werden sollte. Somit können  $M$  Mitglieder der Schutzgruppe gleichzeitig ausfallen und es wird nach wie vor eine Datenverfügbarkeit von 100 % zur Verfügung gestellt. Die  $M$  Löschkodemitglieder werden aus den  $N$  Datenmitgliedern berechnet. Abbildung 13 zeigt den Vorgang für eine reguläre Schutzgruppe mit  $4+2$  ( $N = 4$ ,  $M = 2$ ,  $b = 1$ ).

Da OneFS-Stripes über Nodes verteilt werden, können Dateien, die an  $N+M$  verteilt sind, ohne Verlust von Verfügbarkeit gleichzeitige Node-Ausfälle überstehen. Somit bietet OneFS Stabilität für beliebige Arten von Ausfällen, ob es sich um ein Laufwerk, einen Node oder eine Komponente innerhalb eines Node handelt (z. B. eine Karte). Darüber hinaus gilt ein Node unabhängig von der Anzahl oder Art von Komponenten, die darin fehlschlagen, als ein einziger Ausfall. Wenn also fünf Laufwerke in einem Node ausfallen, wird dies im Hinblick auf den  $N+M$ -Schutz lediglich als ein einziger Ausfall gezählt.

OneFS bietet sogar die einzigartige Möglichkeit,  $M$  variabel auf einen Wert bis zu vier festzulegen und somit für vierfachen Ausfallschutz zu sorgen. Das geht weit über das maximale, heute üblicherweise verwendete RAID-Level, den Schutz vor doppelten Ausfällen mit RAID-6, hinaus. Da die Ausfallsicherheit des Speichers mit dem Umfang an Redundanz geometrisch zunimmt, kann der  $+4n$ -Schutz um Größenordnungen zuverlässiger als ein herkömmliches Hardware-RAID sein. Dieser zusätzliche Schutz bedeutet, dass SATA-Laufwerke mit großer Kapazität, wie z. B. 4-TB- und 6-TB-Laufwerke, problemlos hinzugefügt werden können.

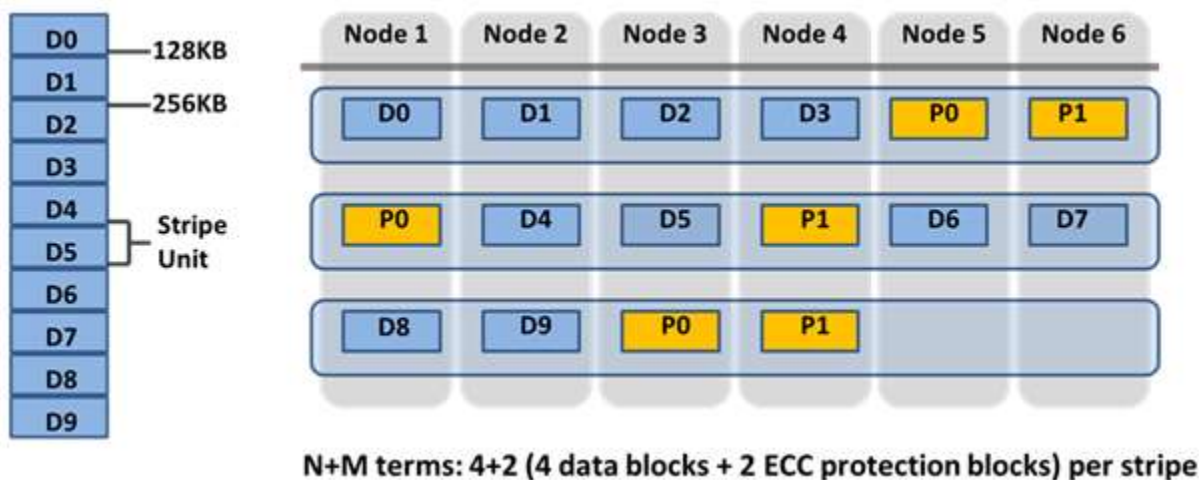


Abbildung 15: OneFS-Redundanz – N M-Löschkodeschutz

Kleinere Cluster können mit +1n-Schutz gesichert werden, was aber Folgendes bedeutet: Ein einziges Laufwerk oder ein Node kann zwar wiederhergestellt werden, zwei Laufwerke in zwei unterschiedlichen Nodes jedoch nicht. Laufwerksausfälle sind sehr viel wahrscheinlicher als Node-Ausfälle. Für Cluster mit großen Laufwerken ist es empfehlenswert, Schutz für mehrere Laufwerksausfälle einzurichten, obwohl Single-Node-Wiederherstellbarkeit akzeptabel ist.

Um für eine Situation vorzusorgen, bei der doppelte Festplattenredundanz und Redundanz eines einzelnen Nodes gefordert wird, können Sie Schutzgruppen von doppelter oder dreifacher Breite erstellen. Diese Schutzgruppen mit doppelter oder dreifacher Breite „umhüllen“ sozusagen dieselbe Node-Gruppe ein- oder zweimal bei deren Auslegung. Da jede Schutzgruppe genau zwei Festplatten für Redundanz enthält, ermöglicht dieser Mechanismus einem Cluster entweder, zwei oder drei Laufwerksausfälle oder einen vollständigen Node-Ausfall ohne Beeinträchtigung der Datenverfügbarkeit zu überstehen.

Ein wichtiger Punkt bei kleinen Clustern besteht darin, dass diese Striping-Methode äußerst effizient ist, da die Effizienz auf dem Laufwerk  $M/(N+M)$  beträgt. Beispiel: Wenn auf einem Cluster mit fünf Nodes und doppeltem Ausfallschutz  $N = 3$ ,  $M = 2$  verwendet wird, ergibt sich eine Schutzgruppe von 3+2 mit einer Effizienz von  $1-2/5$  oder 60 %. Wenn aber im selben Cluster mit fünf Nodes jede Schutzgruppe über zwei Stripes abgelegt wird, ist  $N$  jetzt 8 und  $M = 2$ , sodass eine Effizienz auf der Festplatte von  $1-2/(8+2)$  oder 80 % erreicht und gleichzeitig der doppelte Ausfallschutz für das Laufwerk erhalten wird und nur eine Schutzfunktion des doppelten Node-Ausfallschutzes geopfert werden muss.

OneFS unterstützt verschiedene Schutzschemata. Dazu gehört das weit verbreitete +2d:1n-Schema, das vor zwei Laufwerksausfällen oder einem Node-Ausfall schützt.

① Die Best Practice ist, das für eine bestimmte Clusterkonfiguration empfohlene Schutzlevel zu verwenden. Dieses empfohlene Schutzlevel ist auf den Konfigurationsseiten für die OneFS-WebUI-Storage-Pools deutlich als „Suggested“ (Vorgeschlagen) gekennzeichnet und wird in der Regel standardmäßig konfiguriert. Für alle aktuellen Hardwarekonfigurationen der 6. Generation ist das empfohlene Schutzlevel „+2d:1n“.

Die Hybrid-Schutzschemata sind besonders nützlich für das Gehäuse der 6. Generation mit Node-Konfigurationen mit hoher Dichte, bei denen die Wahrscheinlichkeit, dass mehrere Laufwerke ausfallen, bei weitem die Wahrscheinlichkeit des Ausfalls eines gesamten Nodes übersteigt. Im unwahrscheinlichen Fall, dass mehrere Geräte gleichzeitig ausgefallen sind, sodass die Datei keinem Schutzlevel mehr unterliegt, schützt OneFS alles, was möglich ist, erneut und nimmt Fehler für die einzelnen betroffenen Dateien in die Clusterprotokolle auf.

OneFS bietet außerdem eine Vielzahl von Spiegelungsoptionen von 2x bis 8x und ermöglicht zwei bis acht Spiegelungen des angegebenen Inhalts. Metadaten werden beispielsweise standardmäßig auf einer Ebene über FEC gespiegelt. Wenn eine Datei z. B. mit +2n geschützt ist, wird das zugehörige Metadatenobjekt dreifach gespiegelt.

Die folgende Tabelle enthält einen Überblick über alle OneFS-Schutzlevel:

Schutzlevel	Beschreibung
+1n	Toleriert den Ausfall von 1 Laufwerk ODER 1 Node
+2d:1n	Toleriert den Ausfall von 2 Laufwerken ODER 1 Node
+2n	Toleriert den Ausfall von 2 Laufwerken ODER 2 Nodes
+3d:1n	Toleriert den Ausfall von 3 Laufwerken ODER 1 Node
+3d:1n1d	Toleriert den Ausfall von 3 Laufwerken ODER 1 Node 1 Laufwerk
+3n	Toleriert den Ausfall von 3 Laufwerken ODER 3 Nodes
+4d:1n	Toleriert den Ausfall von 4 Laufwerken ODER 1 Node
+4d:2n	Toleriert den Ausfall von 4 Laufwerken ODER 2 Nodes

<b>+4n</b>	Toleriert den Ausfall von 4 Nodes
<b>2x bis 8x</b>	Spiegelung über 2 bis 8 Nodes nach Konfiguration

OneFS ermöglicht AdministratorInnen, die Schutz-Policies in Echtzeit zu ändern, während Clients angebunden sind und Daten lesen und schreiben.

ⓘ Beachten Sie, dass durch Erhöhen des Schutzlevels eines Clusters sich möglicherweise der Speicherplatz erhöht, die durch die Daten auf dem Cluster belegt wird.

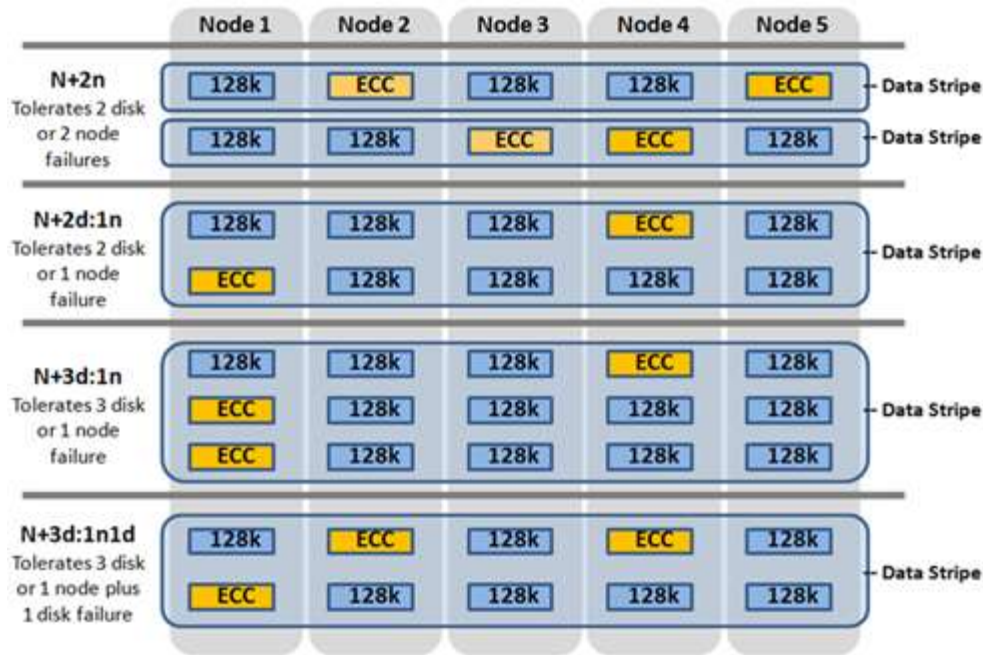


Abbildung 16: OneFS – Hybride Löscheschutzschemata

ⓘ OneFS bietet außerdem Warnmeldungen bei unzureichendem Schutz für neue Clusterinstallationen. Wenn das Cluster nicht geschützt ist, erzeugt das Cluster Event Logging System (CELOG) Warnmeldungen, mit denen AdministratorInnen auf den mangelnden Schutz hingewiesen werden. Außerdem wird eine Änderung des entsprechenden Schutzlevels für die Konfiguration dieses konkreten Clusters vorgeschlagen.

📖 Weitere Informationen finden Sie im Whitepaper [OneFS hohe Verfügbarkeit und Data Protection](#).

### Automatische Partitionierung

Daten-Tiering und -management in OneFS erfolgen durch das SmartPools-Framework. Im Hinblick auf die Effizienz von Data Protection und Datenlayout erleichtert SmartPools die Unterteilung vieler homogener Nodes mit hoher Kapazität in kleinere Laufwerkpools mit einem geringeren MTTDL (Mean Time to Data Loss, Durchschnittliche Zeit bis zu Datenverlust). Zum Beispiel wird ein H500-Cluster mit 80 Nodes normalerweise mit einem Schutzlevel von +3d:1n1d ausgeführt. Durch eine Partitionierung in vier Laufwerkpools mit jeweils 20 Nodes kann jeder Pool mit +2d:1n ausgeführt werden. Dadurch wird der Schutz-Overhead verringert und die Speicherauslastung verbessert wird, ohne dass eine Nettozunahme beim Management-Overhead zu verzeichnen ist.

Um das Ziel einer Vereinfachung des Speichermanagements umzusetzen, berechnet OneFS das Cluster automatisch und partitioniert es in Pools von Laufwerken oder „Node-Pools“, die für MTTDL und eine effiziente Speicherauslastung optimiert sind. Daher werden Entscheidungen zum Schutzlevel, wie im obigen Beispiel mit dem Cluster mit 80 Nodes angeführt, nicht dem Kunden überlassen.

Dank automatischer Bereitstellung wird jede Gruppe kompatibler Node-Hardware automatisch in Laufwerkpools unterteilt, die bis zu 40 Nodes und sechs Laufwerke pro Node enthalten. Diese Node-Pools werden standardmäßig mit +2d:1n geschützt und mehrere Pools können dann in logischen Tiers kombiniert und anhand der Dateipool-Policies von SmartPools verwaltet werden. Durch die Unterteilung der Laufwerke eines Node in mehrere, separat geschützte Pools sind die Nodes bei mehreren Festplattenausfällen sehr viel robuster, als dies bisher möglich war.

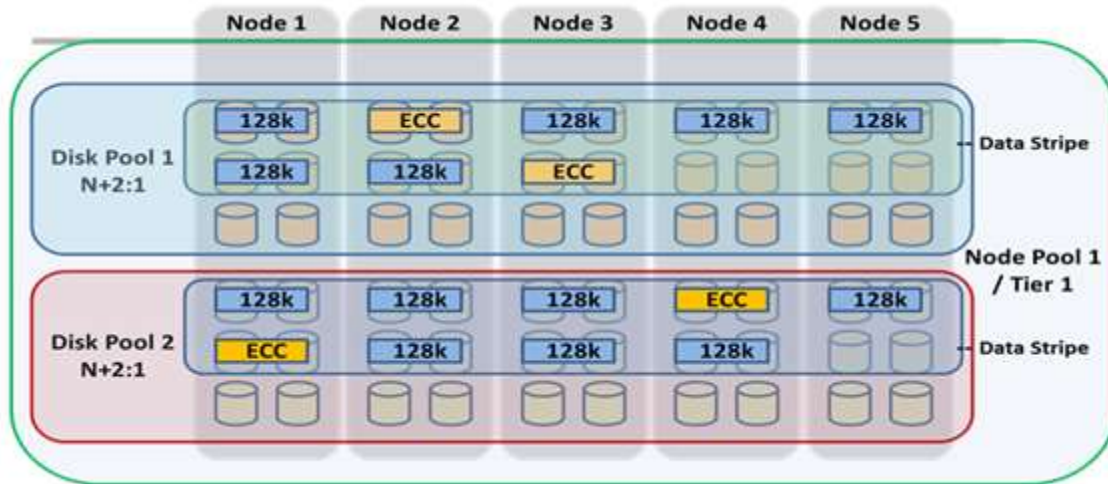


Abbildung 17: Automatische Partitionierung mit SmartPools

📖 Weitere Informationen finden Sie im Whitepaper [SmartPools](#).

Modulare PowerScale-Hardwareplattformen der 6. Generation verfügen über ein hochgradig dichtes, modulares Design, bei dem vier Nodes in einem einzigen 4-HE-Gehäuse enthalten sind. Dieser Ansatz verbessert das Konzept von Laufwerkpools, Node-Pools und „Nachbarschaften“, wodurch ein weiteres Level an Resilienz im OneFS Fehler-Domain-Konzept erreicht wird. Jedes Gehäuse der 6. Generation enthält vier Compute-Module (eins pro Node) und fünf Laufwerkscontainer oder Schlitten pro Node.

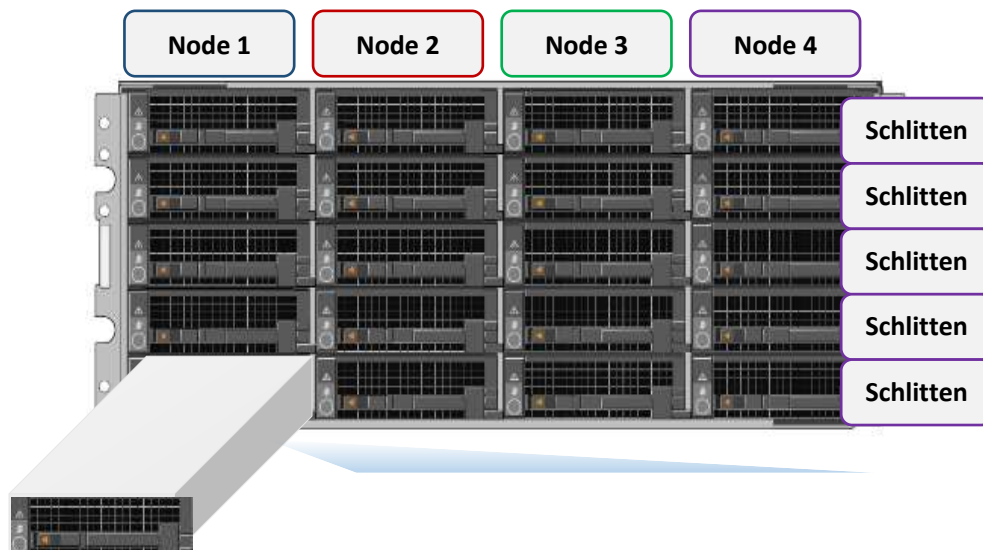


Abbildung 18. Vorderansicht des Plattformgehäuses der 6. Generation mit Laufwerkschlitten

Jeder Schlitten ist ein Fach, das in die Vorderseite des Gehäuses gleitet und je nach Konfiguration eines bestimmten Gehäuses zwischen drei und sechs Laufwerken enthält. Laufwerkpools sind die kleinste Einheit in der Storage-Pool-Hierarchie. Die OneFS-Bereitstellung funktioniert unter der Voraussetzung, dass Laufwerke ähnlicher Nodes in Gruppen oder Laufwerkpools aufgeteilt werden, wobei jeder Pool eine separate Ausfall-Domain darstellt. Diese Laufwerkpools sind standardmäßig durch +2d:1n (oder die Fähigkeit, zwei Laufwerkausfällen oder einen gesamten Node-Ausfall standzuhalten) geschützt.

Laufwerkpools werden in jedem Gen6-Node auf allen fünf Schlitten angeordnet. Beispielsweise hat ein Node mit drei Laufwerken pro Schlitten die folgende Laufwerkpool-Konfiguration:

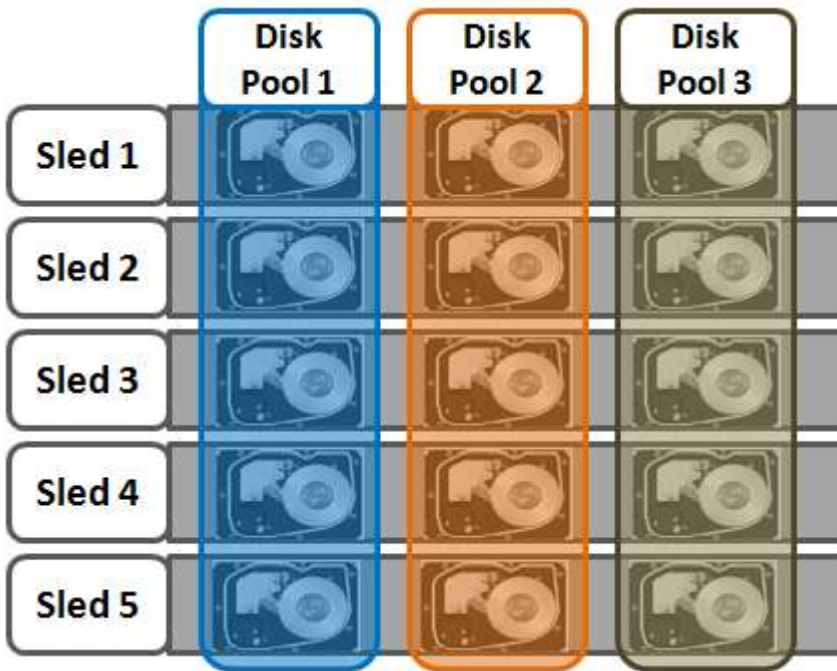


Abbildung 19. OneFS-Laufwerkpools

Node-Pools sind Gruppen von Laufwerkpools, die auf ähnlichen Storage Nodes verteilt sind (Kompatibilitätsklassen). Dies ist unten in Abbildung 20 dargestellt. Mehrere Gruppen verschiedener Node-Typen können in einem einzigen, heterogenen Cluster zusammenarbeiten. Beispiel: Ein Node-Pool mit Nodes der F-Serie für I/Ops-intensive Anwendungen, ein Node-Pool mit Nodes der H-Serie, der hauptsächlich für hochgradig gleichzeitige und sequenzielle Workloads genutzt wird, und ein Node-Pool mit Nodes der A-Serie, der hauptsächlich für Nearline- und/oder Deep-Archiv-Workloads genutzt wird.

Auf diese Weise kann OneFS einen einzigen Storage-Ressourcenpool präsentieren, der mehrere Laufwerkdatenträgertypen umfasst – SSD, Hochgeschwindigkeits-SAS, SATA mit großer Kapazität etc. –, und eine Reihe unterschiedlicher Performance-, Schutz- und Kapazitätsmerkmale bietet. Dieser heterogene Storage-Pool kann wiederum ein breites Spektrum an Anwendungen und Workload-Anforderungen durch ein zentrales, einheitliches Management unterstützen. Ältere und neuere Hardware können zudem kombiniert werden, sodass Investitionen auch über Produktgenerationen hinweg geschützt und nahtlose Hardwareaktualisierungen erzielt werden können.

Jeder Node-Pool enthält nur Laufwerkpools vom gleichen Storage-Node-Typ und ein Laufwerkpool kann zu genau einem Node-Pool gehören. Beispiel: Nodes der F-Serie mit SSD-Festplatte mit 1,6 TB befinden sich in einem Node-Pool, während sich Nodes der A-Serie mit SATA-Laufwerken mit 10 TB in einem anderen Pool befinden. Heute sind mindestens 4 Nodes (ein Gehäuse) pro Node-Pool für Gen6-Hardware wie PowerScale H700 oder drei Nodes pro Pool für eigenständige Nodes wie PowerScale F900 erforderlich.

OneFS-„Nachbarschaften“ sind Fehler-Domains in einem Node-Pool, deren Zweck darin besteht, die Ausfallsicherheit im Allgemeinen zu verbessern und vor der Nichtverfügbarkeit von Daten durch versehentliches Entfernen von Laufwerkschlitzen zu schützen. Für in sich geschlossene Nodes wie den PowerScale F200 hat OneFS eine ideale Größe von 20 Nodes pro Node-Pool und eine maximale Größe von 39 Nodes. Beim Hinzufügen des 40. Nodes teilen sich die Nodes in zwei Nachbarschaften mit je 20 Nodes auf.

Bei der Gen6-Plattform wurde die ideale Größe einer Nachbarschaft von 20 auf 10 Nodes geändert. Dies schützt vor dem gleichzeitigen Auftreten von Fehlern im Node-Paarjournal und vollständigen Gehäuseausfällen.

Partner-Nodes sind Nodes, deren Journale gespiegelt werden. Bei der Gen6-Plattform werden die Node-Journale auf SSDs gespeichert und jedes Journal verfügt über eine Spiegelungskopie auf einem anderen Node, anstatt dass jeder Node sein Journal im NVRAM wie auf früheren Plattformen speichert. Der Node, der das gespiegelte Journal enthält, wird als Partner-Node bezeichnet. Durch die Änderungen am Journal wurden mehrere Vorteile in Bezug auf die Ausfallsicherheit erreicht. SSDs sind beispielsweise persistenter und ausfallsicherer als NVRAM, der erfordert, dass eine geladene Batterie den Status aufrechterhält. Außerdem müssen bei gespiegelten Journalen beide Journalaufwerke komplett ausfallen, bevor ein Journal als verloren betrachtet wird. Wenn also nicht beide gespiegelten Journalaufwerke ausfallen, können beide Partner-Nodes wie gewohnt funktionieren.

Beim Partner-Node-Schutz werden Nodes möglichst in unterschiedlichen Nachbarschaften und damit auch in unterschiedlichen Fehlerdomains platziert. Der Schutz von Partner-Nodes ist möglich, sobald das Cluster fünf vollständige Gehäuse (20 Nodes) erreicht und OneFS nach der ersten Aufteilung der Nachbarschaft Partner-Nodes in unterschiedlichen Umgebungen platziert:

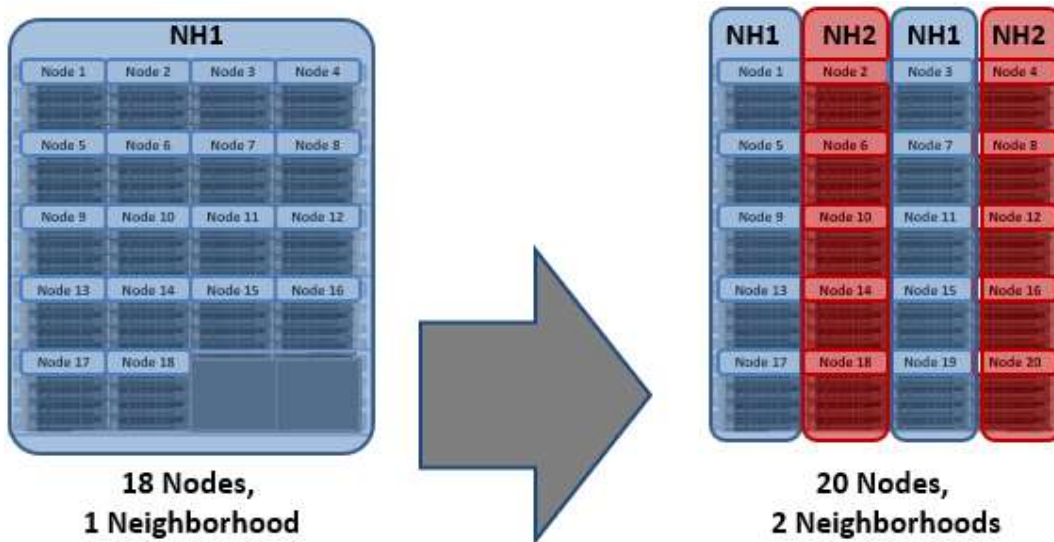


Abbildung 20. Aufteilung auf zwei Nachbarschaften bei 20 Nodes

Der Schutz von Partner-Nodes erhöht die Ausfallsicherheit, denn wenn beide Nodes ausfallen, befinden sie sich in unterschiedlichen Fehlerdomains, sodass ihre Fehlerdomains nur den Verlust eines einzelnen Nodes zu verzeichnen haben.

Beim Gehäuseschutz werden die vier Nodes in einem Gehäuse möglichst in einer separaten Nachbarschaft platziert. Der Gehäuseschutz wird ab 40 Nodes möglich, da bei der Aufteilung der Nachbarschaft bei 40 Nodes jeder Node in einem Gehäuse in einer anderen Nachbarschaft platziert werden kann. Wenn ein Gen6-Cluster mit 38 Nodes auf 40 Nodes erweitert wird, werden die beiden bestehenden Nachbarschaften in vier Nachbarschaften mit jeweils 10 Nodes aufgeteilt:

Der Gehäuseschutz sorgt dafür, dass bei einem Ausfall eines gesamten Gehäuses jede Ausfall-Domain nur einen Node verliert.

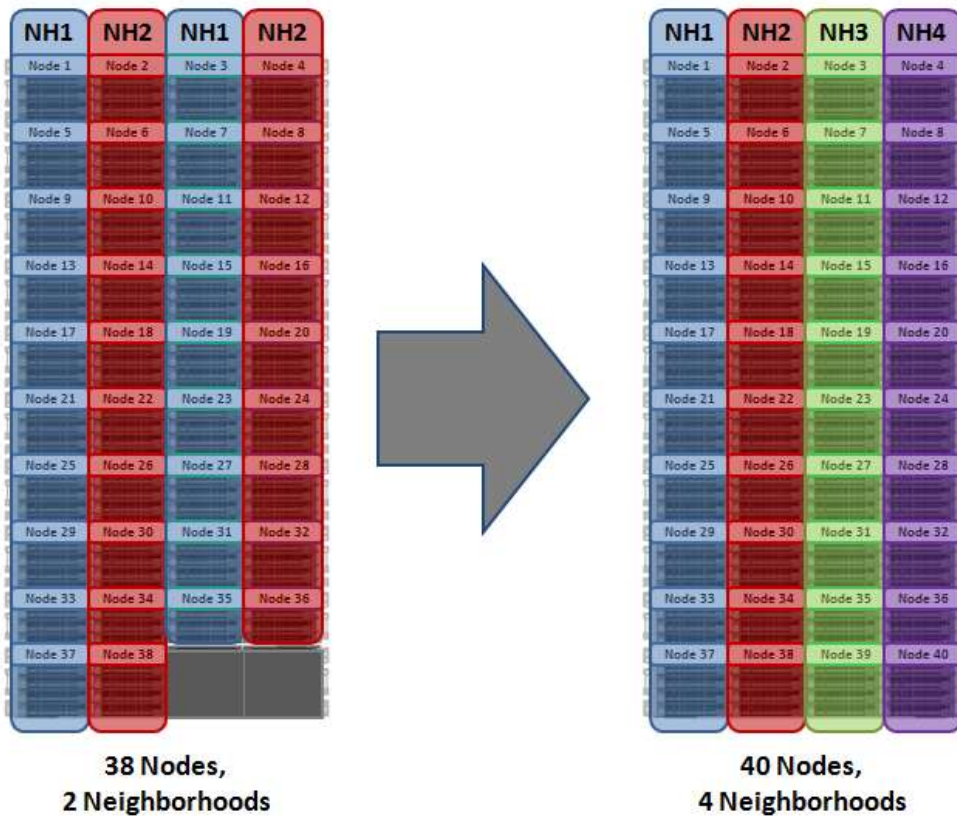


Abbildung 21. OneFS Nachbarschaften – Aufteilung in vier Nachbarschaften

① Ein Cluster mit 40 Nodes oder mehr und vier Nachbarschaften, der mit dem Standardlevel +2d:1n geschützt ist, kann einem Ausfall eines einzelnen Nodes pro Nachbarschaft standhalten. Dadurch ist der Cluster vor einem einzigen Gen6-Gehäuseausfall geschützt.

Insgesamt weist ein Cluster der Gen6-Plattform eine Ausfallsicherheit von mindestens einer Größenordnung größer als die Cluster der Vorgängergeneration mit einer ähnlichen Kapazität als direktes Ergebnis der folgenden Verbesserungen auf:

- Gespiegelte Journale
- Kleinere Nachbarschaften
- Gespiegelte Boot-Laufwerke

## Kompatibilität


Bestimmte ähnliche, aber nicht identische Node-Typen können nach Node-Kompatibilität in einem vorhandenen Node-Pool bereitgestellt werden. OneFS erfordert, dass ein Node-Pool mindestens drei Nodes enthalten muss.

① Aufgrund erheblicher Architekturunterschiede gibt es keine Node-Kompatibilitäten zwischen den Gen6-Plattform, frühere Hardwaregenerationen oder die PowerScale-Nodes.

OneFS enthält auch eine Option für die SSD-Kompatibilität, über die Nodes mit unterschiedlicher Kapazität der SSDs für einen einzigen Node-Pool bereitgestellt werden können.

Die SSD-Kompatibilität wird in der Liste „OneFS WebUI SmartPools Compatibilities“ erstellt und beschrieben und zudem in der Liste „Tiers & Node Pools“ angezeigt.

① Bei der Erstellung dieser SSD-Kompatibilität überprüft OneFS automatisch, ob die beiden zusammenzuführenden Pools über die gleiche Anzahl von SSDs sowie über die gleichen Tier-, Schutz- und L3-Cacheeinstellungen verfügen. Wenn diese Einstellungen unterschiedlich sind, fordert die OneFS-WebUI Sie zur Konsolidierung und Anpassung dieser Einstellungen auf.

 Weitere Informationen finden Sie im Whitepaper [SmartPools](#).

## Unterstützte Protokolle

Clients mit den richtigen Anmeldedaten und Berechtigungen können Daten mithilfe einer der unterstützten Standardmethoden für die Kommunikation mit dem Cluster erstellen, ändern und lesen:

- NFS (Network File System)
- SMB/CIFS (Server Message Block/Common Internet File System)
- FTP (File Transfer Protocol)
- HTTP (Hypertext Transfer Protocol)
- HDFS (Hadoop Distributed File System)
- REST-API (Representational State Transfer Application Programming Interface)
- S3 (Objektspeicher-API)


Für das NFS-Protokoll unterstützt OneFS sowohl NFSv3 als auch NFSv4 sowie NFSv4.1 in OneFS 9.3. Darüber hinaus bieten OneFS 9.2 und höher Unterstützung für NFSv3overRDMA.

Für Microsoft Windows wird das SMB-Protokoll bis zu Version 3 unterstützt. Als Teil des SMB3-Dialekts unterstützt OneFS die folgenden Funktionen:

- SMB3 Multipath
- SMB3 Kontinuierliche Verfügbarkeit und Zeuge
- SMB3-Verschlüsselung

Die SMB3-Verschlüsselung kann pro Share, Zone oder Cluster konfiguriert werden. Nur Betriebssysteme mit Unterstützung für die SMB3-Verschlüsselung können mit verschlüsselten Shares arbeiten. Diese Betriebssysteme können auch mit unverschlüsselten Shares arbeiten, wenn der Cluster so konfiguriert ist, dass nicht verschlüsselte Verbindungen zugelassen werden. Andere Betriebssysteme können nur auf nicht verschlüsselte Shares zugreifen, wenn der Cluster so konfiguriert ist, dass nicht verschlüsselte Verbindungen zugelassen werden.

Das Stammverzeichnis des Dateisystems für alle Daten im Cluster ist /ifs (das OneFS-Dateisystem). Dies kann über das SMB-Protokoll als eine „ifs“-Share (\\<cluster\_name>\ifs) und über das NFS-Protokoll als ein „ifs“-Export (<cluster\_name>:/ifs) dargestellt werden.

 Alle Protokolle verfügen über die Daten, sodass Änderungen am Dateiinhalte über ein Zugriffsprotokoll sofort für alle anderen Protokolle sichtbar sind.

OneFS bietet vollständige Unterstützung für IPv4- und IPv6-Umgebungen für die Front-end-Ethernet-Netzwerke, SmartConnect und die vollständige Palette an Storage-Protokollen und Managementtools.

Darüber hinaus unterstützt OneFS CloudPools die folgenden Speicher-APIs von Cloud-Anbietern, sodass Dateien auf eine Reihe von Speicherzielen verteilt werden können, darunter:

- Amazon Web Services S3
- Microsoft Azure
- Google Cloud Service
- Alibaba Cloud
- Dell EMC ECS
- OneFS RAN (RESTful Zugriff auf Namespace)

 Weitere Informationen finden Sie im [CloudPools – Administrationshandbuch](#).



## Unterbrechungsfreie Vorgänge – Protokollunterstützung

OneFS trägt zur Datenverfügbarkeit durch die Unterstützung von dynamischem NFSv3- und NFSv4-Failover und -Failback für Linux- und UNIX-Clients und kontinuierliche SMB3-Verfügbarkeit für Windows-Clients bei. Damit ist dafür gesorgt, dass bei einem Node-Ausfall oder bei präventiven Wartungen, alle In-Flight-Lese- und Schreibvorgänge an einen anderen Node im Cluster weitergegeben werden, um den Vorgang ohne Nutzer- oder Anwendungsunterbrechung abzuschließen.

Während des Failover werden Clients gleichmäßig über alle verbleibenden Nodes im Cluster verteilt, sodass die Auswirkung auf die Performance äußerst gering bleibt. Wenn ein Node aus einem beliebigen Grund, einschließlich eines Ausfalls, abgeschaltet wird, werden die virtuellen IP-Adressen auf diesem Node nahtlos zu einem anderen Node im Cluster migriert.

Wenn der Offline-Node wieder online geschaltet wird, gleicht SmartConnect die NFS- und SMB3-Clients im gesamten Cluster automatisch aus, um maximale Storage- und Performanceauslastung sicherzustellen. Bei regelmäßigen Systemwartungsarbeiten und Softwareupdates ermöglicht diese Funktion fortlaufende Upgrades pro Node, die für die Dauer der Wartung vollständige Verfügbarkeit bieten.

## Dateifilter

Die OneFS-Dateifilterung kann für NFS- und SMB-Clients verwendet werden, um Schreibvorgänge in eine Export-, Share- oder Zugriffzone zuzulassen oder nicht zuzulassen. Diese Funktion verhindert, dass bestimmte Dateierweiterungstypen für Dateien blockiert werden, die Probleme hinsichtlich Sicherheit, Produktivität, Durchsatz oder Speicherordnung verursachen können. Die Konfiguration kann entweder über eine Ausschlussliste erfolgen, die bestimmte Dateierweiterungen blockiert, oder über eine Einschlussliste, die ausdrücklich nur das Schreiben bestimmter Dateitypen erlaubt.

## Dateneduplizierung – SmartDedupe

Das Produkt SmartDedupe maximiert die Storage-Effizienz eines Clusters, indem es den für die Speicherung von Unternehmensdaten erforderlichen physischen Speicher reduziert. Effizienz wird durch das Scannen von Festplattendaten auf identische Blöcke und das anschließende Löschen von Duplikaten erreicht. Dieser Ansatz wird in der Regel als nachgeordnete oder asynchrone Deduplizierung bezeichnet.

Nachdem doppelte Blöcke erkannt wurden, verschiebt SmartDedupe eine einzelne Kopie dieser Blöcke in einen speziellen Satz von Dateien, die als Schattenspeicher bezeichnet werden. Während dieses Prozesses werden doppelte Blöcke aus den tatsächlichen Dateien entfernt und durch Zeiger auf die Schattenspeicher ersetzt.

Bei der nachgeordneten Deduplizierung werden neue Daten zuerst auf dem Speichergerät gespeichert. Anschließend analysiert ein nachfolgender Prozess die Daten, um nach Gemeinsamkeiten zu suchen. Das bedeutet, dass die anfängliche Performance bei Dateischreib- oder -änderungsvorgängen nicht beeinträchtigt wird, da im Schreibpfad keine zusätzliche Berechnung erforderlich ist.

## Architektur von SmartDedupe

Die Architektur von OneFS SmartDedupe besteht aus fünf Hauptmodulen:

- Control Path für die Deduplizierung
- Deduplizierungsjob
- Deduplizierungs-Engine
- Schattenspeicher
- Deduplizierungsinfrastruktur

Der SmartDedupe Control Path umfasst das OneFS-Webmanagementschnittstelle (WebUI), die Befehlszeilenoberfläche (CLI) sowie die RESTful-Plattform-API und ist für das Management von Konfiguration, Planung und Steuerung des Deduplizierungsjobs zuständig. Der Job selbst ist ein hochgradig verteilter Hintergrundprozess, der die Orchestrierung der Deduplizierung für alle Nodes im Cluster managt. Die Jobsteuerung umfasst das Scannen von Dateisystemen, die Erkennung und die gemeinsame Nutzung von übereinstimmenden Datenblöcken in Zusammenarbeit mit der Deduplizierungs-Engine. Die Infrastrukturschicht der Deduplizierung ist das Kernelmodul, das die Konsolidierung von gemeinsam genutzten Datenblöcken in den Schattenspeicher, den Dateisystemcontainern, die sowohl physische Datenblöcke als auch Verweise, oder Zeiger, auf gemeinsam genutzte Blöcke enthalten, durchführt. Diese Elemente werden im Folgenden ausführlicher beschrieben.



Abbildung 22: Modulare Architektur von OneFS SmartDedupe

[📖](#) Weitere Informationen finden Sie im Whitepaper [OneFS SmartDedupe](#).

### Schattenspeicher

OneFS-Schattenspeicher sind Dateisystemcontainer, die die Speicherung von Daten zur gemeinsamen Nutzung ermöglichen. Daher können Dateien auf OneFS sowohl physische Daten als auch Zeiger, oder Verweise, auf gemeinsam genutzte Blöcke in Schattenspeichern enthalten.

Schattenspeicher ähneln regulären Dateien, enthalten aber meist nicht alle Metadaten, die normalerweise regulären Datei-Inodes zugeordnet sind. Insbesondere werden zeitbasierte Attribute (Erstellungszeit, Änderungszeit usw.) explizit nicht beibehalten. Jeder Schattenspeicher kann bis zu 256 Blöcke enthalten, wobei jeder Block von 32.000 Dateien referenziert werden kann. Wenn diese Referenzgrenze von 32.000 überschritten wird, wird ein neuer Schattenspeicher erstellt. Darüber hinaus verweisen Schattenspeicher nicht auf andere Schattenspeicher. Und Snapshots von Schattenspeichern sind nicht zulässig, da Schattenspeicher über keine Hardlinks verfügen.

① Schattenspeicher werden zusätzlich zur Deduplizierung auch für OneFS-Datei-Clones und Small File Storage Efficiency (SFSE) verwendet.

### Storage-Effizienz kleiner Dateien

Ein anderer Hauptverbraucher von Schattenspeichern ist OneFS Small File Storage Efficiency (Storage-Effizienz kleiner Dateien). Diese Funktion maximiert die Speicherplatzauslastung eines Clusters, indem die Menge des physischen Speichermediums reduziert wird, die erforderlich ist, um die kleinen Dateien aufzunehmen, die häufig ein Archiv-Datenvolumen umfassen, z. B. in PACS-Workflows im Gesundheitswesen.

Effizienz wird für kleine Dateien durch das Scannen der Daten auf der Festplatte erreicht, die durch vollständige Kopiespiegelungen geschützt sind und in Schattenspeichern platziert werden. Diese Schattenspeicher werden dann durch Parität geschützt, anstatt gespiegelt zu werden, und bieten in der Regel eine Storage-Effizienz von 80 % oder mehr.

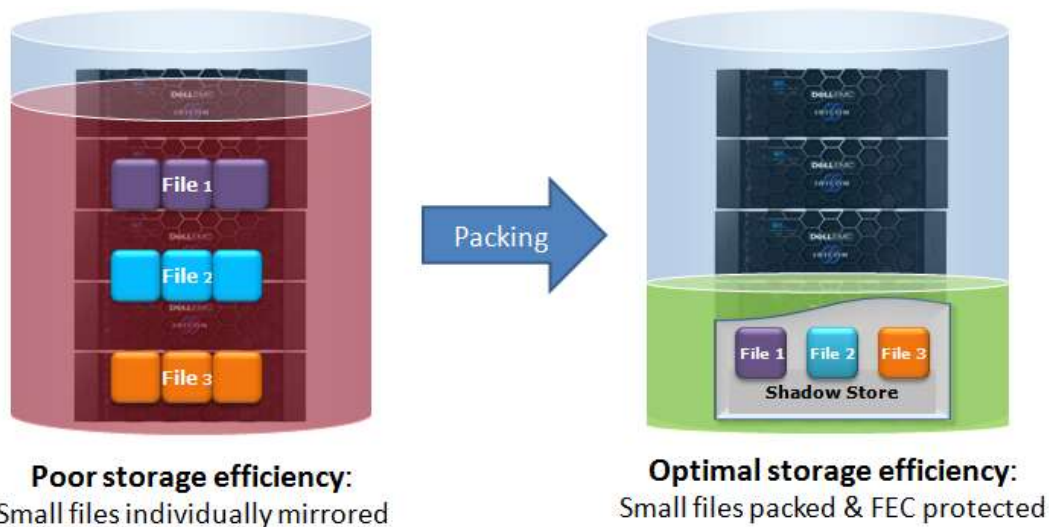


Abbildung 23: Containerisierung kleiner Dateien

Bei der Storage-Effizienz kleiner Dateien wurde eine geringe Performanceeinbuße bei der Leselatenz zugunsten einer besseren Speicherauslastung in Kauf genommen. Die archivierten Dateien bleiben offensichtlich beschreibbar, wenn aber containerisierte Dateien mit Schattenverweisen gelöscht, gekürzt oder überschrieben werden, können nicht referenzierte Blöcke in Schattenspeichern verbleiben. Diese Blöcke werden später freigegeben und können zu Löchern führen, die die Storage-Effizienz reduzieren.

Der tatsächliche Effizienzverlust hängt vom Layout des Schutzlevels ab, das vom Schattenspeicher verwendet wird. Kleinere Schutzgruppen sind anfälliger als containerisierte Dateien, da alle Blöcke in Containern höchstens eine Verweisdatei haben und die gepackten Größen (Dateigröße) klein sind.

Ein Defragmentierer wurde hinzugefügt, um die Fragmentierung von Dateien aufgrund von Überschreibungen und Löschungen zu reduzieren. Dieser Schattenspeicher-Defragmentierer ist in den ShadowStoreDelete-Job integriert. Beim Defragmentierungsprozess wird jede containerisierte Datei in logische Segmente (jeweils bis zu etwa 32 MB) unterteilt und jedes Segment wird für die Fragmentierung bewertet.

Wenn die Storage-Effizienz eines fragmentierten Segments unterhalb des Ziels liegt, wird dieses Segment verarbeitet, indem die Daten an einen anderen Speicherort verlegt werden. Die standardmäßige Zieleffizienz ist 90 % der maximalen Storage-Effizienz, die mit der vom ShadowStore verwendeten Schutzstufe verfügbar ist. In größeren Schutzgruppengrößen ist eine höhere Fragmentierungsstufe möglich, bevor die Storage-Effizienz unter diesen Schwellenwert fällt.

## Inline-Datenreduzierung

Die OneFS-Inline-Datenreduzierung ist auf den F900-, F810-, F600- und F200-All-Flash-Nodes, den Hybridgehäusen H700/7000 und H5600 sowie der A300/3000-Archivplattform verfügbar. Die OneFS-Architektur besteht aus den folgenden Hauptkomponenten:

- Datenreduzierungsplattform
- Komprimierungs-Engine und Blockzuordnung
- Nullblockentfernungs-Phase
- Deduplizierung von In-Memory-Index und Schattenspeicherinfrastruktur
- Datenreduzierungswarnmeldungs- und -reporting-Framework
- Datenreduzierungs-Kontrollpfad

Der Schreibpfad für die Inline-Datenreduzierung besteht aus 3 Hauptphasen:

- Nullblockentfernung
- Inline-Deduplizierung
- Inline-Komprimierung

Wenn Inline-Komprimierung und -Deduplizierung in einem Cluster aktiviert sind, wird die Nullblockentfernung zuerst durchgeführt, gefolgt von der Deduplizierung und dann der Komprimierung. Diese Reihenfolge ermöglicht es jeder Phase, den Arbeitsumfang in jeder nachfolgenden Phase zu reduzieren.



Abbildung 24: Inline-Datenreduzierungsworkflow

Die F810 beinhaltet eine Komprimierungsfunktion für Hardware, bei der jeder Node in einem F810-Gehäuse mit einem Mellanox InnoVa-2 Flex-Adapter ausgestattet ist. Das bedeutet, dass Komprimierung und Dekomprimierung transparent durch den Mellanox-Adapter mit minimaler Latenz durchgeführt werden, wodurch die teuren CPU- und Arbeitsspeicherressourcen eines Node nicht verbraucht werden müssen.

Die OneFS-Hardwarekomprimierungs-Engine verwendet zlib mit einer Softwareimplementierung von igzip für die Nodes von PowerScale F900, F810, F600, F200, H700/7000, H5600 und A300/3000. Die Softwarekomprimierung wird auch als Fallback im Falle eines Komprimierungshardwarefehlers und in einem gemischten Cluster zur Verwendung in Nicht-F810-Nodes ohne Hardwarekomprimierungsfunktion und als Fallback im Falle eines Komprimierungshardwarefehlers verwendet. OneFS verwendet für die Komprimierung eine Segmentgröße von 128 KB, wobei jedes Segment aus 16 8-KB-Datenblöcken besteht. Das ist optimal, da es auch dieselbe Größe ist, die OneFS für seine Data-Protection-Stripe-Einheiten verwendet. Da kein Overhead durch zusätzliches Packen von Segmenten entsteht, ist die Lösung einfach und effizient.

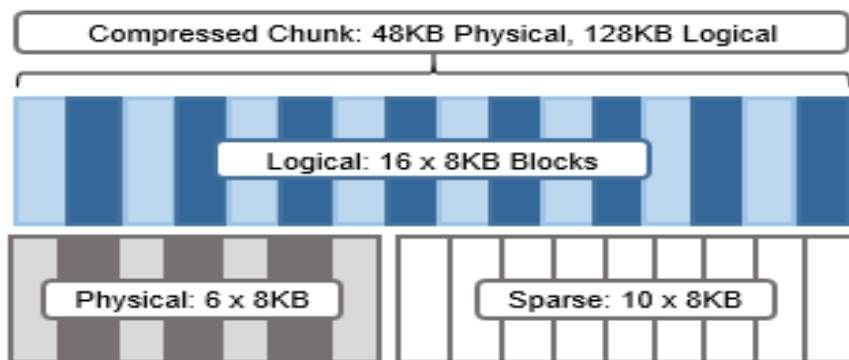


Abbildung 25: Komprimierungssegmente und transparente OneFS-Überlagerung

Sehen Sie sich das obige Diagramm an. Nach der Komprimierung wird dieser Block von 16 auf 6 Blöcke mit jeweils 8 KB reduziert. Das bedeutet, dass dieser Block jetzt eine Größe von 48 KB aufweist. OneFS bietet ein transparentes logisches Overlay der physischen Attribute. Mit diesem Overlay wird beschrieben, ob die Backup-Daten komprimiert sind oder nicht und welche Blöcke im Block physisch oder verstreut sind, so dass die Verbraucher des Dateisystems von der Komprimierung nicht betroffen sind. Somit wird der komprimierte Block logischerweise mit einer Größe von 128 KB dargestellt, unabhängig von seiner tatsächlichen physikalischen Größe.

Einsparungen bei der Effizienz müssen mindestens 8 KB (einen Block) betragen, damit die Komprimierung durchgeführt werden kann. Andernfalls wird dieses Segment oder diese Datei übergeben und verbleibt im ursprünglichen, nicht komprimierten Status. Beispielsweise würde eine Datei mit 16 KB, die 8 KB (einen Block) an Einsparungen ergibt, komprimiert werden. Sobald eine Datei komprimiert wurde, wird sie durch FEC geschützt.

Komprimierungssegmente überschreiten niemals Node-Pool-Grenzen. Dadurch wird müssen Daten weder dekomprimiert noch erneut komprimiert werden, um Schutzlevel zu ändern, wiederhergestellte Schreibvorgänge durchzuführen oder auf andere Weise Schutzgruppengrenzen zu verschieben.

## Dynamische Skalierung/Skalierung nach Bedarf

### Performance und Kapazität

Im Gegensatz zu herkömmlichen Speichersystemen, die „hochskaliert“ werden müssen, wenn zusätzliche Performance oder Kapazität erforderlich ist, ermöglicht OneFS einem Speichersystem ein „Scale-out“ und damit die nahtlose Erweiterung des vorhandenen Dateisystems oder Volume auf eine Kapazität mit mehreren Petabyte. Gleichzeitig wird die Performance linear erhöht.

Das Hinzufügen von Kapazität und Performance zu einem Cluster ist erheblich einfacher als bei anderen Speichersystemen. Die Storage-AdministratorInnen brauchen nur drei einfache Schritte durchzuführen: einen weiteren Node im Rack hinzufügen, den Node an das Back-end-Netzwerk anbinden und das Cluster anweisen, den zusätzlichen Node hinzuzufügen. Der neue Node bietet zusätzliche Kapazität und Performance, da jeder Node CPU, Speicher, Cache, Netzwerk, NVRAM und I/O-Kontrollpfade umfasst.

Die in OneFS integrierte AutoBalance-Funktion verschiebt Daten automatisch und auf kohärente Weise über das Back-end-Netzwerk, sodass auf dem Cluster vorhandene Daten in diesen neuen Storage Node verschoben werden. Dank des automatischen Ausgleichs wird dafür gesorgt, dass der neue Node kein Hot Spot für neue Daten wird und dass vorhandene Daten von den Vorteilen eines leistungsfähigeren Speichersystems profitieren können. Die AutoBalance-Funktion in OneFS ist auch für die EndnutzerInnen vollständig transparent und kann so eingestellt werden, dass die Auswirkung auf leistungsfähige Workloads minimiert wird.

Diese Funktion allein ermöglicht eine transparente Skalierung in OneFS von TB auf PB bei laufendem Betrieb, ohne zusätzliche Managementzeit für die AdministratorInnen oder erhöhte Komplexität innerhalb des Speichersystems.

Ein umfangreiches Speichersystem muss die für verschiedene Workflows erforderliche Performance bereitstellen, unabhängig davon, ob diese Workflows sequenziell, gleichzeitig oder zufällig sind. Zwischen und innerhalb einzelner Anwendungen existieren verschiedene Workflows. OneFS erfüllt dank intelligenter Software all diese Anforderungen gleichzeitig. Noch wichtiger ist aber, dass in OneFS Durchsatz und IOPS linear mit der Anzahl der in einem einzigen System vorhandenen Nodes skaliert werden. Aufgrund der ausgeglichenen Datenverteilung, des automatischen Ausgleichs und der dezentralen Verarbeitung können Sie mit OneFS weitere CPUs, Netzwerkports und Arbeitsspeicher bei der Skalierung des Systems nutzen.

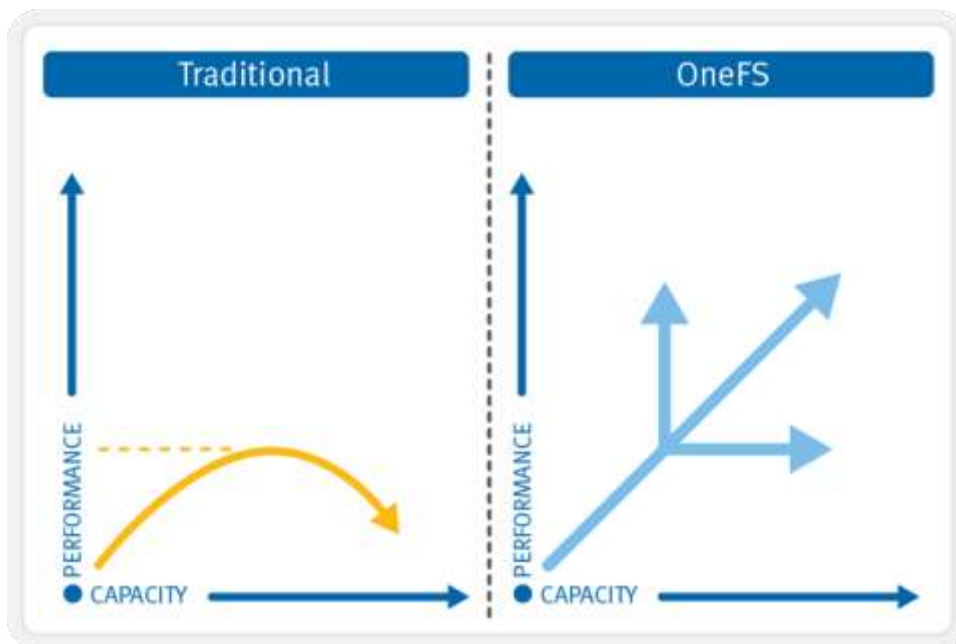


Abbildung 26: Lineare OneFS-Skalierbarkeit

### Schnittstellen

AdministratorInnen können zur Verwaltung eines Storage-Clusters in ihrer Umgebung mehrere Schnittstellen verwenden:

- Webverwaltungsschnittstelle (WebUI)
- Befehlszeilenoberfläche über SSH-Netzwerkzugriff oder einen seriellen RS232-Anschluss
- LCD-Bereich auf den Nodes selbst für einfache Funktionen zum Hinzufügen/Entfernen
- RESTful-Plattform-API zur programmatischen Steuerung und Automatisierung von Clusterkonfiguration und -management

Dashboard

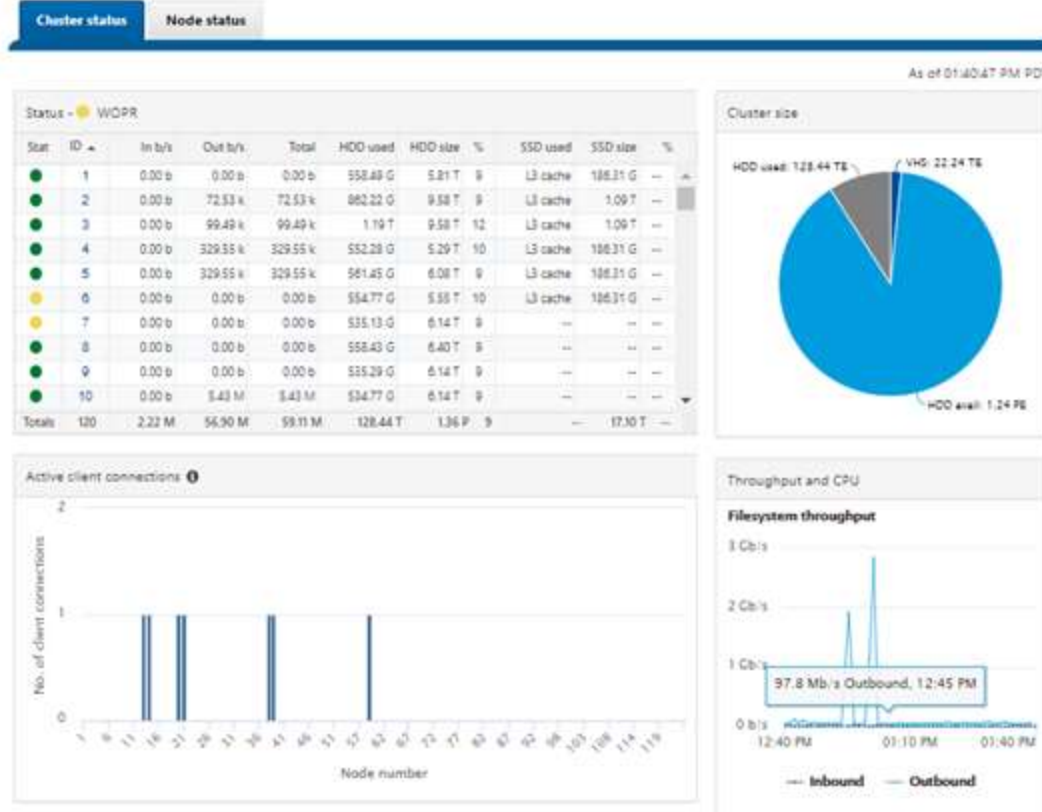


Abbildung 27: Webbenutzeroberfläche von OneFS

📖 Weitere Informationen zu OneFS-Befehlen und zur Funktionskonfiguration finden Sie im [OneFS-Administrationsleitfaden](#).

### Authentifizierung und Zugriffskontrolle

Authentifizierungsservices stellen eine Sicherheitsmaßnahme dar, da sie die Anmeldedaten von NutzerInnen überprüfen, bevor diese auf Dateien zugreifen und Dateien verändern können. OneFS unterstützt vier Methoden für die Authentifizierung von NutzerInnen:

- Active Directory (AD)
- LDAP (Lightweight Directory Access Protocol)
- NIS (Netzwerkinformationsservice)
- Lokale NutzerInnen und Gruppen

OneFS unterstützt die Verwendung mehrerer Authentifizierungstypen. Es wird jedoch empfohlen, dass Sie sich mit den Interaktionen zwischen Authentifizierungstypen gründlich vertraut machen, bevor Sie mehrere Methoden für das Cluster aktivieren. Detaillierte Informationen zur richtigen Konfiguration mehrerer Authentifizierungsmodi finden Sie in der Produktdokumentation.

## Active Directory

Active Directory, eine LDAP-Implementierung von Microsoft, ist ein Verzeichnisservice, der Informationen über die Netzwerkressourcen speichern kann. Active Directory bietet viele Funktionen, aber der Hauptgrund für die Verbindung des Clusters mit der Domain besteht in der Authentifizierung von NutzerInnen und Gruppen.

Sie können die Active Directory-Einstellungen eines Clusters über die Webverwaltungsschnittstelle oder die Befehlszeilenoberfläche konfigurieren und verwalten. Es wird jedoch empfohlen, nach Möglichkeit die Webverwaltungsschnittstelle zu verwenden.

Jeder Node im Cluster verwendet dasselbe Active Directory-Rechnerkonto, wodurch die Verwaltung sehr vereinfacht wird.

## LDAP

Das LDAP (Lightweight Directory Access Protocol) ist ein Netzwerkprotokoll zum Definieren, Abfragen und Ändern von Services und Ressourcen. Der Hauptvorteil von LDAP besteht in der offenen Gestaltung der Verzeichnisservices und der Fähigkeit, LDAP plattformübergreifend zu verwenden. Mithilfe von LDAP kann das Clusterspeichersystem von NutzerInnen und Gruppen authentifizieren, bevor diesen Zugriff auf das Cluster gewährt wird.

## NIS

NIS (Network Information Service) von Sun Microsystems ist ein Protokoll für Verzeichnisservices, mit dem das OneFS NutzerInnen und Gruppen, die auf das Cluster zugreifen, authentifizieren kann. NIS, auch als „Gelbe Seiten“ bezeichnet, unterscheidet sich vom Protokoll NIS+, das von OneFS nicht unterstützt wird.

## Lokale NutzerInnen

OneFS unterstützt lokale Nutzer- und Gruppenauthentifizierung. Sie können Nodes für lokale NutzerInnen und Gruppen direkt mithilfe der WebUI auf dem Cluster erstellen. Die lokale Authentifizierung kann nützlich sein, wenn Verzeichnisservices – Active Directory, LDAP oder NIS – nicht verwendet werden, oder wenn ein bestimmter Nutzer bzw. eine bestimmte Anwendung Zugriff auf das Cluster benötigt.

## Zugriffszonen

Zugriffszonen bieten eine Methode zur logischen Partitionierung des Clusterzugriffs und zum Zuweisen von Ressourcen zu eigenständigen Einheiten, wodurch eine gemeinsam genutzte, oder mehrmandantenfähige, Mandantenumgebung bereitgestellt wird. Zur Vereinfachung dieses Vorgangs werden die drei wichtigsten externen Zugriffskomponenten in Zugriffszonen zusammengefasst:

- Clusternetzwerkconfiguration
- Dateiprotokollzugriff
- Authentifizierung


SmartConnect-Zonen sind mit einer Reihe von SMB-Shares, NFS-Exporten, HDFS-Racks und einem oder mehreren Authentifizierungsanbietern pro Zone für die Zugriffskontrolle verknüpft. Dies bietet die Vorteile eines zentral gemanagten einzelnen Dateisystems, das für mehrere Mandanten bereitgestellt und gesichert werden kann. Dies ist besonders in Unternehmensumgebungen nützlich, in denen mehrere separate Geschäftsbereiche über eine zentrale IT-Abteilung bedient werden. Ein weiteres Beispiel: Während eines Projekts zur Serverkonsolidierung werden mehrere Windows-Dateiserver zusammengeführt, die mit separaten, nicht vertrauenswürdigen Active Directory-Strukturen verbunden sind.

Beim Einsatz von Zugriffszonen umfasst die integrierte Systemzugriffszone standardmäßig eine Instanz jedes unterstützten Authentifizierungsanbieters, alle verfügbaren SMB-Shares und alle verfügbaren NFS-Exporte.

Diese Authentifizierungsanbieter können mehrere Instanzen von Microsoft Active Directory, LDAP, NIS und lokale Nutzer- oder Gruppendatenbanken umfassen.

## Rollenbasierte Verwaltung

Bei der rollenbasierten Administration handelt es sich um ein Zugriffskontrollsystem auf Grundlage von Clustermanagementrollen (Roles Based Access Control, RBAC), das die Berechtigungen von „Root“- und „Administrator“-NutzerInnen in detailliertere Berechtigungen aufteilt und die Zuweisung dieser Berechtigungen zu bestimmten Rollen ermöglicht. Diese Rollen können dann anderen, nicht privilegierten NutzerInnen zugeteilt werden. So kann beispielsweise RechenzentrumsmitarbeiterInnen Lesezugriff für das gesamte Cluster zugewiesen werden, wodurch sie umfassenden Monitoringzugriff erhalten, aber keine Konfigurationsänderungen vornehmen können. OneFS bietet eine integrierte Sammlung von Rollen, einschließlich Audit, System- und Sicherheitsadministrator sowie die Möglichkeit, benutzerdefinierte Rollen zu erstellen, entweder pro Zugriffszone oder im gesamten Cluster. Die rollenbasierte Administration ist in die OneFS-Befehlszeilenoberfläche, die WebUI und die Plattform-API integriert.

 Weitere Informationen zu Identitätsmanagement, Authentifizierung und Zugriffskontrolle in Multiprotokoll-Umgebungen finden Sie im [OneFS Multiprotocol Security Guide](#).

## OneFS-Auditing

OneFS bietet die Möglichkeit, die Systemkonfiguration und die NFS-, SMB- und HDFS-Protokollaktivität auf einem Cluster zu überwachen. So können Unternehmen verschiedene Vorgaben für Data Governance und zur Einhaltung behördlicher Auflagen erfüllen, an die sie möglicherweise gebunden sind.

Alle Auditdaten werden im Dateisystem des Clusters gespeichert und geschützt und nach Auditthemen organisiert. Von hier aus können Auditdaten über das Dell EMC Common Event Enabler (CEE)-Framework in Anwendungen von Drittanbietern wie Varonis DatAdvantage und Symantec Data Insight exportiert werden. Das OneFS-Protokoll-Auditing kann pro Zugriffszone aktiviert werden, sodass eine fein abgestimmte Steuerung im gesamten Cluster möglich ist.

Ein Cluster kann Auditereignisse auf bis zu fünf CEE-Server pro Node in einer parallelen Konfiguration mit Lastenausgleich schreiben. Somit stellt OneFS eine End-to-End-Auditösung der Enterprise-Klasse dar.

 Weitere Informationen finden Sie im Whitepaper [OneFS Audit](#).

## Softwareupgrade

Durch die Durchführung eines Upgrades auf die neueste Version von OneFS nutzen Sie alle neuen Funktionen, Fehlerkorrekturen und Merkmale. Cluster können mit zwei Methoden aktualisiert werden: Simultanes oder unterbrechungsfreies Upgrade

### Simultanes Upgrade

Bei einem simultanen Upgrade wird das neue Betriebssystem installiert. Ferner werden alle Nodes im Cluster gleichzeitig neu gestartet. Für ein simultanes Upgrade muss der Service bei Neustart der Nodes vorübergehend für einen Zeitraum von weniger als zwei Minuten unterbrochen werden.

### Fortlaufendes Upgrade

Bei einem fortlaufenden Upgrade wird das Upgrade einzeln durchgeführt und die Nodes im Cluster werden nacheinander neu gestartet. Während eines fortlaufenden Upgrades bleibt das Cluster online und stellt Clients weiterhin ohne Serviceunterbrechungen Daten bereit. Bei Versionen vor OneFS 8.0 kann ein fortlaufendes Upgrade nur innerhalb einer OneFS-Produktreihe mit einer Codeversion und nicht zwischen unterschiedlichen Hauptversionen der Codeversion von OneFS durchgeführt werden. Ab OneFS 8.0 kann für jede neue Version ein fortlaufendes Upgrade aus der vorherigen Version erfolgen.

### Unterbrechungsfreie Upgrades

Unterbrechungsfreie Upgrades (Non Disruptive Upgrades, NDUs) ermöglichen ClusteradministratorInnen das Upgrade des Storage-Betriebssystems, während ihre EndnutzerInnen ohne Fehler oder Unterbrechungen weiterhin auf Daten zugreifen können. Bei der Aktualisierung des Betriebssystems auf einem Cluster handelt es sich um ein einfaches unterbrechungsfreies Upgrade. Während dieses Prozesses wird jeweils ein Node auf den neuen Code aktualisiert und die aktiven, damit verbundenen NFS- und SMB3-Clients werden automatisch zu anderen Nodes im Cluster migriert. Ein partielles Upgrade ist ebenfalls zulässig, wobei für eine Teilmenge der Cluster-Nodes ein Upgrade erfolgen kann. Die Teilmenge der Nodes kann auch während des Upgrades erweitert werden. Ein Upgrade kann angehalten und fortgesetzt werden, sodass Kunden Upgrades über mehrere kleinere Wartungsfenster erstrecken können. Darüber hinaus bieten OneFS 8.2.2 und höher parallele Upgrades, wobei Cluster gleichzeitig ein Upgrade einer gesamten Nachbarschaft oder Fehlerdomain durchführen können, wodurch die Dauer von Upgrades großer Cluster deutlich verkürzt wird. OneFS 9.2 und höher kombinieren Betriebssystem- und Firmwareupgrades, wodurch die Auswirkungen und die Dauer von Upgrades deutlich reduziert werden, indem sie parallel durchgeführt werden können. 9.2 und höher umfassen auch drain-basierte Upgrades, wobei Nodes daran gehindert werden, Protokollservices neu zu booten oder neu zu starten, bis alle SMB-Clients vom Node getrennt wurden.



## Rollback-fähig

OneFS unterstützt Upgrade-Rollback, bei dem die Möglichkeit besteht, einen Cluster mit einem nicht übernommenen Upgrade auf seine vorherige OneFS-Version zurückzusetzen.

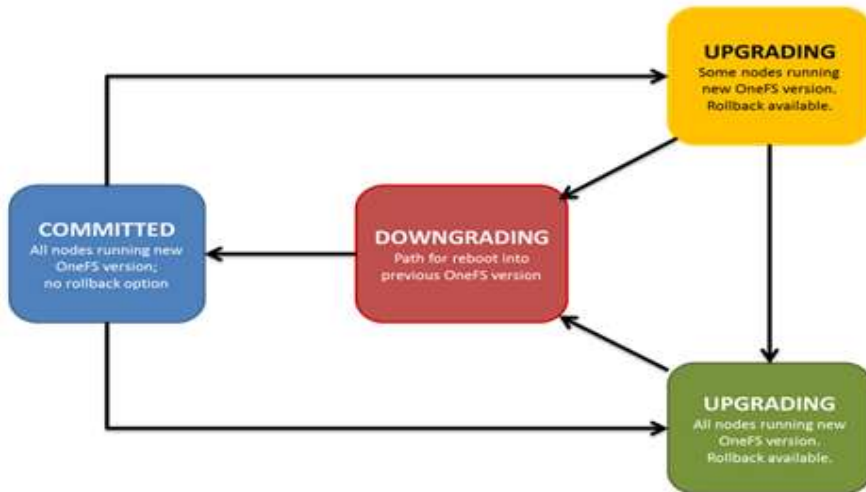


Abbildung 28: Status unterbrechungsfreier OneFS-Upgrades

## Automatische Firmwareupdates

Mit OneFS betriebene Cluster unterstützen automatische Updates der Laufwerksfirmware für neue und Ersatzlaufwerke als Teil des unterbrechungsfreien Update-Prozesses der Firmware. Firmwareupdates werden über Laufwerks-Supportpakete bereitgestellt, die sowohl das Management vorhandener als auch neuer Laufwerke im gesamten Cluster vereinfachen und rationalisieren. Auf diese Weise wird sichergestellt, dass die Laufwerksfirmware auf dem neuesten Stand ist und die Wahrscheinlichkeit von Fehlern aufgrund bekannter Laufwerksprobleme minimiert wird. Daher ist das automatische Update der Laufwerksfirmware eine wichtige Komponente der OneFS-Strategie zur Erzielung hoher Verfügbarkeit und unterbrechungsfreier Vorgänge. Laufwerks- und Node-Firmware können entweder als fortlaufendes Upgrade oder über einen vollständigen Cluster-Neustart übernommen werden.

Vor OneFS 8.2 mussten Firmwareupdates der Nodes jeweils pro Node installiert werden. Dies war ein zeitaufwendiger Vorgang, insbesondere bei großen Clustern. Firmwareupdates der Nodes können jetzt für einen gesamten Cluster durch Bereitstellung einer Liste der Nodes, die gleichzeitig aktualisiert werden sollen, vorgenommen werden. Mit dem Upgrade-Hilfstooll kann eine gewünschte Kombination von gleichzeitig aktualisierbaren Nodes und eine Liste der explizit nicht gemeinsam zu aktualisierenden Nodes (z. B. Nodes in einem Node-Paar) ausgewählt werden.

## Durchführen des Upgrades

Im Rahmen eines Upgrades führt OneFS automatisch eine Überprüfung vor der Installation aus. Dabei wird überprüft, ob die Konfiguration in der aktuellen Installation von OneFS mit der Version von OneFS kompatibel ist, für die das Upgrade beabsichtigt ist. Wenn eine nicht unterstützte Konfiguration gefunden wird, wird das Upgrade angehalten und Troubleshooting-Anweisungen werden angezeigt. Eine installationsvorbereitende Upgradeprüfung vor dem Start des Upgrades trägt dazu bei, Unterbrechungen aufgrund von nicht kompatiblen Konfigurationen zu vermeiden.

## OneFS-Data-Protection- und Managementsoftware

OneFS bietet ein umfassendes, auf Ihre Anforderungen abgestimmtes Softwareportfolio für Data Protection und Management an:

Softwaremodul	Funktion	Beschreibung
<a href="#"><u>CloudIQ™</u></a>	Überwachung der Clusterintegrität	Nutzen Sie intelligente und vorausschauende Analysen, um die Integrität Ihres Clusters proaktiv zu überwachen.
<a href="#"><u>InsightIQ™</u></a>	Performancemanagement	Maximiert die Performance Ihres Clusters mit innovativen Tools für Performance-Monitoring und -reporting
<a href="#"><u>DataIQ™</u></a>	Datenanalyse und -management	Suchen, Abrufen und Managen von Daten innerhalb von Sekunden, unabhängig davon, wo sich diese befinden – in Datei- und Objektspeichern, lokal oder in der Cloud. Verschaffen Sie sich einen ganzheitlichen Überblick über heterogene Speichersysteme in einer einzigen Ansicht, wodurch Daten, die in Silos gefangen sind, effektiv transparent gemacht werden.
<a href="#"><u>SmartPools™</u></a>	Ressourcenmanagement	Implementiert eine hocheffiziente, automatisierte Tiered-Storage-Strategie zur Optimierung der Storage Performance und Kosten
<a href="#"><u>SmartQuotas™</u></a>	Datenmanagement	Managt und weist Quotas zu für eine nahtlose Partitionierung und Thin Provisioning von Speicher in einfach zu managende Segmente auf Cluster-, Verzeichnis-, Unterverzeichnis-, Nutzer- und Gruppenebene
<a href="#"><u>SmartConnect™</u></a>	Datenzugriff	Ermöglicht einem Lastenausgleich für Clientverbindungen und ein dynamisches NFS-Failover und -Failback von Clientverbindungen zwischen Storage Nodes zur Optimierung der Nutzung von Clusterressourcen
<a href="#"><u>SnapshotIQ™</u></a>	Data Protection	Effizienter und zuverlässiger Schutz Ihrer Daten mit sicheren, nahezu sofortigen Snapshots bei geringem bis gar keinem Performance-Overhead Beschleunigte Recovery wichtiger Daten mit nahezu sofortigen Snapshot-Wiederherstellungen nach Bedarf Erstellen Sie speicherplatz- und zeiteffiziente, änderbare Kopien eines schreibgeschützten Snapshots mit beschreibbaren OneFS-Snapshots.
<a href="#"><u>SynclQ™</u></a>	Datenreplikation	Asynchrone Replikation und Verteilung großer, geschäftskritischer Datenvolumen an mehrere freigegebene Speichersysteme an mehreren Standorten für eine zuverlässige Disaster-Recovery-Fähigkeit Einfaches Failover und Failback mit einem Tastendruck zur erhöhten Verfügbarkeit geschäftskritischer Daten
<a href="#"><u>SmartLock™</u></a>	Datenaufbewahrung	Schutz Ihrer wichtigen Daten vor versehentlichem, vorzeitigem oder böswilligem Ändern oder Löschen mit unserem softwarebasierten WORM-Ansatz (Write Once Read Many) und Einhaltung strenger Compliance- und Governance-Anforderungen wie SEC 17a-4
<a href="#"><u>SmartDedupe™</u></a>	Dateneduplizierung	Maximierte Storage-Effizienz durch Scannen des Clusters auf identische Blöcke und nachfolgendes Eliminieren von Duplikaten, wodurch die Menge an erforderlichen physischen Speichermedien verringert wird.
<a href="#"><u>CloudPools™</u></a>	Cloud-Tiering	Mit CloudPools können Sie festlegen, welche Daten auf Ihrem Cluster im Cloud-Storage archiviert werden sollen. Cloud-Anbieter sind u. a. Microsoft Azure, Google Cloud, Amazon S3, Dell EMC ECS und native OneFS.

Tabelle 3: Dell EMC Power-Scale-Datendienstportfolio

Weitere Informationen finden Sie in der Produktdokumentation.

## Fazit

Mit Dell EMC Scale-out-NAS-Lösungen, die vom OneFS-Betriebssystem unterstützt werden, können Unternehmen von TB auf PB in einem einzigen Dateisystem, einem einzigen Volume und mit einem zentralen Verwaltungspunkt skalieren. OneFS bietet hohe Performance und hohen Durchsatz oder beides, und das ohne zusätzliche Komplexität beim Management.

Rechenzentren der nächsten Generation müssen auf nachhaltige Skalierbarkeit ausgelegt werden. Sie müssen die Vorteile der Automatisierung und Kommerzialisierung von Hardware nutzen, die vollständige Auslastung der Netzwerk-Fabric ermöglichen und maximale Flexibilität für Unternehmen bieten, die darauf bedacht sind, sich ständig ändernde Anforderungen erfüllen zu können.

OneFS ist das Dateisystem der Zukunft, mit dem sich alle diese Herausforderungen meistern lassen. OneFS bietet die folgenden Vorteile:

- Vollständig verteiltes, einziges Dateisystem
- Leistungsfähiges, vollständig symmetrisches Cluster
- Datei-Striping über alle Nodes in einem Cluster hinweg
- Weniger Komplexität dank automatisierter Software
- Dynamische Inhaltsverteilung
- Flexibler Schutz von Daten
- Hohe Verfügbarkeit
- Webbasierte und Befehlszeilenadministration

OneFS ist ideal für dateibasierte und unstrukturierte „Big Data“-Anwendungen in Data-Lake-Unternehmensumgebungen – wie umfangreiche Home Directories, Dateifreigaben, Archive, Virtualisierung und Geschäftsanalysen – sowie für vielfältige datenintensive und leistungsfähige Computing-Umgebungen geeignet, z. B. Energieversorgung, Finanzdienstleistungen, Internet- und Hostingservices, Business Intelligence, Technik, Fertigung, Medien und Unterhaltung, Bioinformatik und wissenschaftliche Forschung.

## MACHEN SIE DEN NÄCHSTEN SCHRITT

Wenn Sie mehr darüber erfahren möchten, wie Ihr Unternehmen von PowerScale-NAS-Storage-Lösungen profitieren kann, wenden Sie sich an Ihren Dell EMC Vertriebsmitarbeiter oder einen autorisierten Reseller.

Auf der Website von [Dell EMC PowerScale](#) können Sie Funktionen vergleichen und weitere Informationen erhalten.



Weitere Informationen  
zu Dell EMC  
PowerScale-Lösungen



Kontakt zu einem  
Dell EMC Experten



Weitere Ressourcen



Reden Sie mit:  
[#DellEMCStorage](#)