



分层

Enterprise applications

NetApp
February 11, 2026

目录

- 分层 1
 - 概述 1
 - 架构 1
 - 对象存储提供程序 1
 - 数据和元数据 2
 - 备份 2
 - 分层策略 2
 - 分层策略 2
 - 检索策略 3
- 层策略 4
 - 完整文件层 4
 - 部分文件分层 4
 - 归档日志层 5
 - Snapshot层 5
 - 备份层 6
- 对象存储访问中断 7

分层

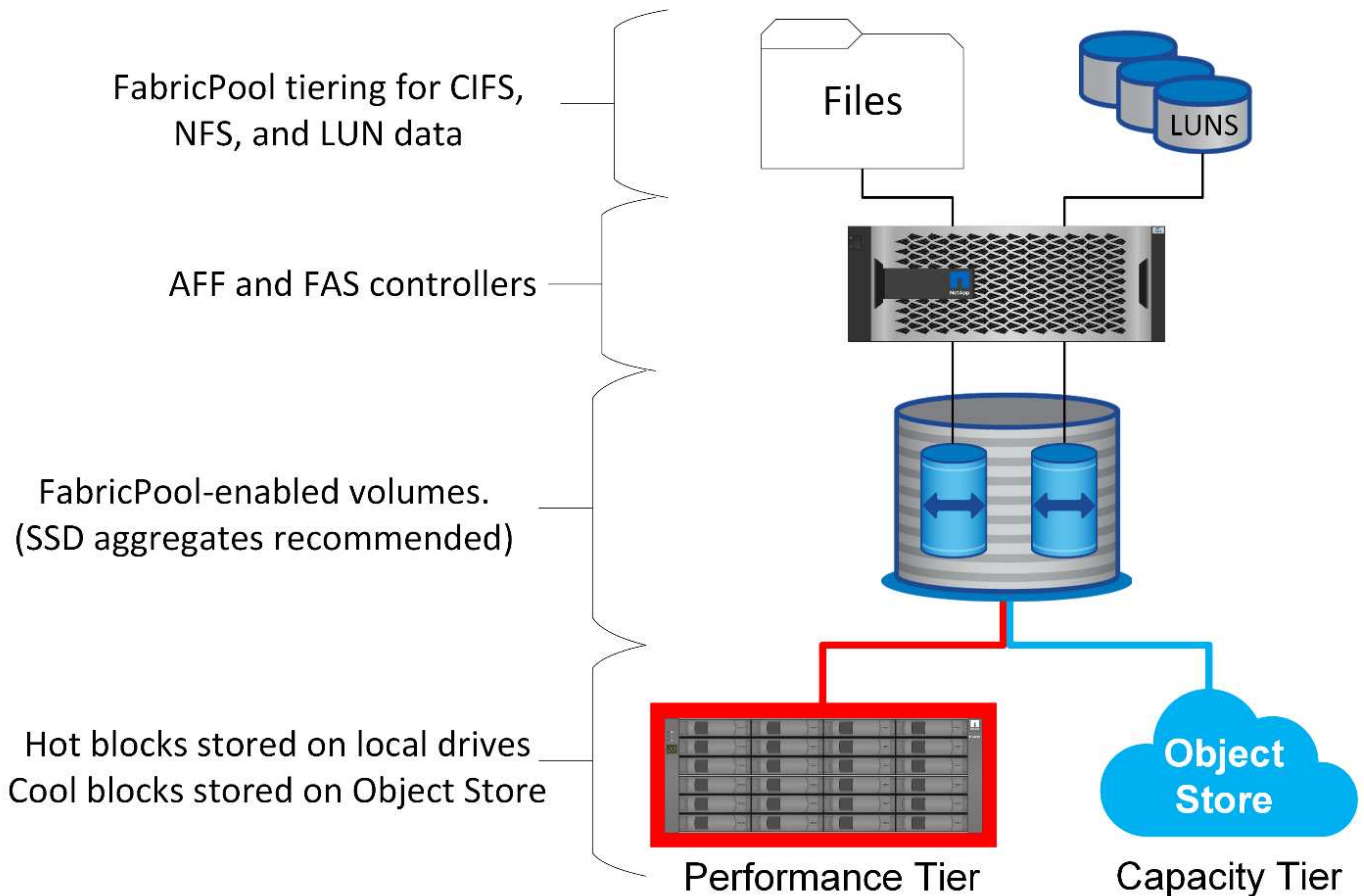
概述

要了解FabricPool层如何影响Oracle和其他数据库、需要了解低级别FabricPool架构。

架构

FabricPool是一种分层技术、可将块分为热块或冷块、并将其放置在最合适的存储层中。性能层通常位于SSD存储上、并托管热数据块。容量层位于对象存储上、用于托管超酷数据块。对象存储支持包括NetApp StorageGRID、ONTAP S3、Microsoft Azure Blob存储、阿里云对象存储服务、IBM云对象存储、Google云存储和Amazon AWS S3。

可以使用多个分层策略来控制块的分类方式、这些策略可以按卷设置、也可以根据需要进行更改。在性能层和容量层之间仅移动数据块。定义LUN和文件系统结构的元数据始终保留在性能层上。因此、可在ONTAP上集中管理。文件和LUN的显示与任何其他ONTAP配置中存储的数据没有任何不同。NetApp AFF或FAS控制器会应用定义的策略将数据移动到相应的层。



对象存储提供程序

对象存储协议使用简单的HTTP或HTTPS请求来存储大量数据对象。对对象存储的访问必须可靠、因为从ONTAP进行数据访问取决于请求的及时处理。选项包括Amazon S3 Standard和Infrequent Access选项、以及Microsoft Azure Hot and Cool Blob Storage、IBM Cloud和Google Cloud。不支持Amazon Glacier和Amazon

Archive等归档选项、因为检索数据所需的时间可能会超出主机操作系统和应用程序的容错范围。

NetApp StorageGRID也受支持、是最佳的企业级解决方案。它是一个高性能、可扩展且高度安全的对象存储系统、可以为FabricPool数据以及越来越可能成为企业应用程序环境一部分的其他对象存储应用程序提供地理冗余。

StorageGRID还可以通过避免许多公有云提供商为从其服务中读取数据而收取的传出费用来降低成本。

数据和元数据

请注意、此处的术语“数据”适用场景是指实际数据块、而不是元数据。仅对数据块进行分层、而元数据保留在性能层中。此外、只有读取实际数据块时、数据块的状态才会受到热或冷的影响。仅读取文件的名称、时间戳或所有权元数据并不会影响底层数据块的位置。

备份

虽然FabricPool可以显著减少存储占用空间、但它本身并不是一个备份解决方案。NetApp WAFL元数据始终保留在性能层上。如果灾难性灾难破坏了性能层、则无法使用容量层上的数据创建新环境、因为该环境不包含WAFL元数据。

但是、FabricPool可以成为备份策略的一部分。例如、可以为FabricPool配置NetApp SnapMirror复制技术。镜像的每一半都可以与对象存储目标建立自己的连接。结果是生成两个独立的数据副本。主副本由性能层上的块以及容量层中的关联块组成、副本是第二组性能和容量块。

分层策略

分层策略

ONTAP提供了四个策略、用于控制性能层上的Oracle数据如何成为重新定位到容量层的候选对象。

仅快照

。 `snapshot-only tiering-policy` 仅适用于未与活动文件系统共享的块。实际上、它需要对数据库备份进行层化。创建快照后、块将成为分层的候选块、然后将其覆盖、从而导致块仅存在于快照中。之前的延迟 `snapshot-only` 块被视为冷却由控制 `tiering-minimum-cooling-days` 卷的设置。从ONTAP 9.8开始、此范围为2到183天。

许多数据集的更改率较低、因此此策略节省的空间极少。例如、在ONTAP上观察到的典型数据库每周的更改率小于5%。数据库归档日志可能会占用大量空间、但它们通常仍存在于活动文件系统中、因此不适合在此策略下进行分层。

自动

。 `auto` 层划分策略可将层划分扩展到快照特定的块以及活动文件系统中的块。块被视为冷却之前的延迟由控制 `tiering-minimum-cooling-days` 卷的设置。从ONTAP 9.8开始、此范围为2到183天。

此方法可启用在中不可用的层选项 `snapshot-only` 策略。例如、数据保护策略可能需要保留90天的某些日志文件。如果将冷却期设置为3天、则会将超过3天的任何日志文件从性能层中分层出来。此操作可释放性能层上的大量空间、同时仍允许您查看和管理整个90天的数据。

无

。 none 分层策略可防止从存储层对任何其他块进行分层、但容量层中的任何数据仍会保留在容量层中、直到被读取为止。如果随后读取该块、则会将其移回并放置在性能层上。

使用的主要原因 none 分层策略可防止对块进行分层、但随着时间的推移更改策略可能会很有用。例如、假设某个特定数据集已广泛分层到容量层、但却出现了对全部性能功能的意外需求。可以更改此策略、以防止任何其他分层、并确认随着IO增加而读回的任何块仍保留在性能层中。

全部

。 all 层策略将取代 backup 自ONTAP 9.6起的策略。。 backup 策略仅应用于数据保护卷、即SnapMirror或NetApp SnapVault目标。。 all 策略的功能相同、但不限于数据保护卷。

使用此策略、块将立即视为冷数据块、并有资格立即分层到容量层。

此策略尤其适用于长期备份。它还可用作分层存储管理(HSM)的一种形式。过去、HSM通常用于将文件的数据块分层到磁带、同时使文件本身在文件系统中保持可见。使用的FabricPool卷 all 通过策略、您可以将文件存储在一个可见且易于管理的位置、但几乎不会占用本地存储层上的空间。

检索策略

分层策略用于控制将哪些Oracle数据库块从性能层分层到容量层。检索策略控制读取已分层的块时发生的情况。

Default

所有FabricPool卷的初始设置为 default，这意味着该行为由`云检索策略控制。`具体行为取决于所使用的层策略。

- auto-仅检索随机读取的数据
- snapshot-only-检索所有按顺序或随机读取的数据
- none-检索所有按顺序或随机读取的数据
- all-不从容量层检索数据

读取时

正在设置 ... cloud-retrieval-policy 到读取时会覆盖默认行为、因此读取任何分层数据都会将该数据返回到性能层。

例如、某个卷在下可能已长时间使用不多 auto 分层策略和大多数块现在已分层。

如果业务需求发生意外变化、需要重复扫描某些数据以准备特定报告、则可能需要更改 cloud-retrieval-policy to on-read 以确保读取的所有数据(包括按顺序读取的数据和随机读取的数据)都返回到性能层。这样可以提高卷的顺序I/O性能。

提升

提升策略的行为取决于层策略。如果此层策略为 auto，然后设置 cloud-retrieval-policy `to`promote 在下次分层扫描时从容量层恢复所有块。

如果此层策略为 `snapshot-only`，则返回的唯一块是与活动文件系统关联的块。通常、这不会产生任何影响、因为只有数据块在下进行了分层 `snapshot-only` 策略将是专门与快照关联的块。活动文件系统中不会存在分层块。

但是、如果卷上的数据是通过卷SnapRestore或文件克隆操作从快照还原的、则活动文件系统现在可能需要一些因仅与快照关联而分层出的块。可能需要临时更改 `cloud-retrieval-policy` 策略为 `promote` 以快速检索所有本地所需的块。

从不

请勿从容量层检索块。

层策略

完整文件层

虽然FabricPool分层在块级运行、但在某些情况下、可用于提供文件级分层。

许多应用程序数据集都按日期进行组织、随着数据老化、访问这些数据的可能性通常越来越小。例如、银行可能有一个PDF文件存储库、其中包含五年的客户对账单、但只有最近几个月处于活动状态。FabricPool可用于将旧数据文件重新定位到容量层。冷却期为14天、可确保最近14天的PDF文件仍保留在性能层上。此外、至少每14天读取一次的文件将保持热状态、因此仍保留在性能层上。

策略

要实施基于文件的分层方法、您必须拥有已写入且随后未修改的文件。。 `tiering-minimum-cooling-days` 策略应设置得足够高、以便可能需要的文件仍保留在性能层上。例如、如果某个数据集需要最近60天的数据且性能最佳、则需要设置 `tiering-minimum-cooling-days` 期限为60。根据文件访问模式、也可以实现类似的结果。例如、如果需要最近90天的数据、而应用程序正在访问这90天的数据、则数据将保留在性能层上。通过设置 `tiering-minimum-cooling-days` 从2开始、当数据变得不太活跃后、您会收到分层提示。

。 `auto` 要对这些块进行层化、需要使用策略、因为只有 `auto` 策略会影响活动文件系统块中的块。



任何类型的数据访问都会重置热图数据。病毒扫描、索引编制甚至是读取源文件的备份活动会阻止分层、因为需要分层 `tiering-minimum-cooling-days` 从未达到阈值。

部分文件分层

由于FabricPool在块级别工作、因此可能会更改的文件可以部分分层到对象存储、同时也可以部分保留在性能层上。

这在数据库中很常见。已知包含非活动块的数据库也是FabricPool层的候选数据库。例如、供应链管理数据库可能包含历史信息、这些信息在需要时必须可用、但在正常操作期间不会访问。可以使用FabricPool有选择地重新定位非活动块。

例如、使用的FabricPool卷上运行的数据文件 `tiering-minimum-cooling-days` 90天期限将在性能层上保留前90天访问的任何块。但是、任何在90天内未访问的内容都会重新定位到容量层。在其他情况下、正常应用程序活动会将正确的块保留在正确的层上。例如、如果数据库通常用于定期处理前60天的数据、则要低得多 `tiering-minimum-cooling-days` 可以设置期限、因为应用程序的自然活动可确保不会过早重新定位块。



。 `auto` 对数据库使用策略时应谨慎。许多数据库都定期开展活动、例如季度末流程或重新编制索引操作。如果这些操作的期限大于 `tiering-minimum-cooling-days` 可能会发生性能问题。例如、如果季度末处理需要1 TB的数据、而这些数据在其他情况下未被触及、则这些数据现在可能位于容量层上。从容量层读取的速度通常非常快、可能不会出现发生原因性能问题、但具体结果取决于对象存储配置。

策略

。 `tiering-minimum-cooling-days` 策略应设置得足够高、以保留性能层上可能需要的文件。例如、如果数据库中可能需要最新60天的数据且性能最佳、则需要设置 `tiering-minimum-cooling-days` 期限为60天。根据文件的访问模式、也可以实现类似的结果。例如、如果需要最近90天的数据、而应用程序正在访问这90天的数据、则数据将保留在性能层上。设置 `tiering-minimum-cooling-days` 在数据变得不太活跃后、将立即对数据进行分层。

。 `auto` 要对这些块进行层化、需要使用策略、因为只有 `auto` 策略会影响活动文件系统中的块。



任何类型的数据访问都会重置热图数据。因此、数据库完整表扫描甚至读取源文件的备份活动都会阻止分层、因为需要分层 `tiering-minimum-cooling-days` 从未达到阈值。

归档日志层

FabricPool最重要的用途或许是提高已知冷数据(如数据库事务日志)的效率。

大多数关系数据库都在事务日志归档模式下运行、以提供时间点恢复。通过记录事务日志中的更改来提交对数据库的更改、事务日志将保留而不被覆盖。因此、可能需要保留大量归档事务日志。许多其他应用程序工作流也存在类似的例子、这些工作流生成的数据必须保留、但极不可能被访问。

FabricPool通过提供具有集成层的单个解决方案解决了这些问题。文件会存储在通常的位置并始终可访问、但在主阵列上几乎不会占用任何空间。

策略

使用 `tiering-minimum-cooling-days` 如果策略设置为几天、则会在性能层上保留最近创建的文件(即近期最可能需要的文件)中的块。然后、旧文件中的数据块将移至容量层。

。 `auto` 在达到冷却阈值时强制执行提示分层、而不管日志是已删除还是仍位于主文件系统中。将所有可能需要的日志存储在活动文件系统中的位置也可以简化管理。没有理由通过搜索快照来查找需要还原的文件。

某些应用程序(如Microsoft SQL Server)会在备份操作期间会对事务日志文件进行节段、以便日志不再位于活动文件系统中。可以使用节省容量 `snapshot-only` 分层策略、但 `auto` 策略对日志数据没有用处、因为活动文件系统中的日志数据很少会冷却下来。

Snapshot层

FabricPool的初始版本针对备份用例。唯一可以分层的块类型是不再与活动文件系统中的数据关联的块。因此、只能将快照数据块移至容量层。当您需要确保性能不会受到影响时、这仍然是最安全的一种层选项。

Policies—本地快照

可通过两种方法将非活动快照块分层到容量层。首先是 `snapshot-only` 策略仅针对快照块。虽然 `auto` 策略包括 `snapshot-only` 块、它还会对活动文件系统中的块进行分层。这可能并不可取。

。 `tiering-minimum-cooling-days` 值应设置为一个时间段、以便在性能层上提供还原期间可能需要的数据。例如、关键生产数据库的大多数还原方案都包括前几天某个时间的还原点。设置 `tiering-minimum-cooling-days` 值为3可确保对文件进行任何还原都能使文件立即实现最高性能。活动文件中的所有块仍位于快速存储上、而无需从容量层中恢复。

Policies—复制的快照

使用 `SnapMirror` 或 `SnapVault` 复制的快照仅用于恢复、通常应使用 `FabricPool all` 策略。使用此策略、可以复制元数据、但所有数据块都会立即发送到容量层、从而实现最高性能。大多数恢复过程都涉及顺序 I/O、这本身就很高效。应评估从对象存储目标恢复的时间、但在设计完善的架构中、此恢复过程不需要比从本地数据恢复明显慢。

如果复制的数据也要用于克隆、则 `auto` 策略更合适、使用 `tiering-minimum-cooling-days` 包含预计在克隆环境中定期使用的数据的价值。例如、数据库的活动工作集可能包括前三天读取或写入的数据、但也可能包括另外6个月的历史数据。如果是、则 `auto SnapMirror` 目标上的策略可使工作集在性能层上可用。

备份层

传统应用程序备份包括 Oracle Recovery Manager 等产品、这些产品可在原始数据库位置之外创建基于文件的备份。

```
`tiering-minimum-cooling-days` policy of a few days preserves the most recent backups, and therefore the backups most likely to be required for an urgent recovery situation, on the performance tier. The data blocks of the older files are then moved to the capacity tier.
```

。 ``auto``

策略是最适合备份数据的策略。这样可以确保在达到冷却阈值时及时分层、而不管这些文件是已删除还是仍位于主文件系统中。将所有可能需要的文件存储在活动文件系统中的位置也可以简化管理。没有理由通过搜索快照来查找需要还原的文件。

。 `snapshot-only` 可以使策略有效、但该策略仅适用于不再位于活动文件系统中的适用场景块。因此、必须先删除 NFS 或 SMB 共享上的文件、然后才能对数据进行分层。

对于 LUN 配置、此策略的效率甚至会更低、因为从 LUN 中删除文件只会从文件系统元数据中删除文件引用。LUN 上的实际块将一直保留在原位、直到被覆盖为止。这种情况可能会在删除文件和覆盖块并成为可进行层的候选块之间造成长时间延迟。移动有一些好处 `snapshot-only` 块到容量层、但总体而言、`FabricPool` 备份数据管理最适合与结合使用 `auto` 策略。



这种方法有助于用户更高效地管理备份所需的空间、但 `FabricPool` 本身并不是一种备份技术。将备份文件分层到对象存储可简化管理、因为这些文件在原始存储系统上仍然可见、但对象存储目标中的数据块依赖于原始存储系统。如果源卷丢失、则对象存储数据将不再可用。

对象存储访问中断

使用FabricPool对数据集进行分层会导致主存储阵列与对象存储层之间存在依赖关系。有许多对象存储选项可提供不同级别的可用性。请务必了解主存储阵列与对象存储层之间可能断开连接的影响。

如果向ONTAP发出的I/O需要容量层中的数据、而ONTAP无法访问容量层来检索块、则此I/O最终会超时。此超时的影响取决于所使用的协议。在NFS环境中、ONTAP会根据协议使用EJUKEBOX或EDELAY响应进行响应。某些较早的操作系统可能会将此错误视为错误、但Oracle Direct NFS客户端的当前操作系统和当前修补程序级别会将此错误视为可检取的错误、并继续等待I/O完成。

适用场景SAN环境超时时间更短。如果对象存储环境中需要某个块、但该块在两分钟内仍不可访问、则会向主机返回读取错误。ONTAP卷和LUN会保持联机、但主机操作系统可能会将文件系统标记为处于错误状态。

对象存储连接问题 `snapshot-only` 策略不太值得关注、因为只有备份数据是分层的。通信问题会使数据恢复速度变慢、但不会影响正在使用的数据。。 `auto` 和 `all` 策略允许对活动LUN中的冷数据进行分层、这意味着对象存储数据检索期间出现错误可能会影响数据库可用性。采用这些策略的SAN部署只能与专为实现高可用性而设计的企业级对象存储和网络连接结合使用。NetApp StorageGRID是一个更好的选择。

版权信息

版权所有 © 2026 NetApp, Inc.。保留所有权利。中国印刷。未经版权所有者事先书面许可，本文档中受版权保护的任何部分不得以任何形式或通过任何手段（图片、电子或机械方式，包括影印、录音、录像或存储在电子检索系统中）进行复制。

从受版权保护的 NetApp 资料派生的软件受以下许可和免责声明的约束：

本软件由 NetApp 按“原样”提供，不含任何明示或暗示担保，包括但不限于适销性以及针对特定用途的适用性的隐含担保，特此声明不承担任何责任。在任何情况下，对于因使用本软件而以任何方式造成的任何直接性、间接性、偶然性、特殊性、惩罚性或后果性损失（包括但不限于购买替代商品或服务；使用、数据或利润方面的损失；或者业务中断），无论原因如何以及基于何种责任理论，无论出于合同、严格责任或侵权行为（包括疏忽或其他行为），NetApp 均不承担责任，即使已被告知存在上述损失的可能性。

NetApp 保留在不另行通知的情况下随时对本文档所述的任何产品进行更改的权利。除非 NetApp 以书面形式明确同意，否则 NetApp 不承担因使用本文档所述产品而产生的任何责任或义务。使用或购买本产品不表示获得 NetApp 的任何专利权、商标权或任何其他知识产权许可。

本手册中描述的产品可能受一项或多项美国专利、外国专利或正在申请的专利的保护。

有限权利说明：政府使用、复制或公开本文档受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中“技术数据权利 — 非商用”条款第 (b)(3) 条规定的限制条件的约束。

本文档中所含数据与商业产品和/或商业服务（定义见 FAR 2.101）相关，属于 NetApp, Inc. 的专有信息。根据本协议提供的所有 NetApp 技术数据和计算机软件具有商业性质，并完全由私人出资开发。美国政府对这些数据的使用权具有非排他性、全球性、受限且不可撤销的许可，该许可既不可转让，也不可再许可，但仅限在与交付数据所依据的美国政府合同有关且受合同支持的情况下使用。除本文档规定的情形外，未经 NetApp, Inc. 事先书面批准，不得使用、披露、复制、修改、操作或显示这些数据。美国政府对国防部的授权仅限于 DFARS 的第 252.227-7015(b)（2014 年 2 月）条款中明确的权利。

商标信息

NetApp、NetApp 标识和 <http://www.netapp.com/TM> 上所列的商标是 NetApp, Inc. 的商标。其他公司和产品名称可能是其各自所有者的商标。