# 技术报告

## How to enable StorageGRID in your environment

NetApp
April 22, 2024

# 目录

# 技术报告

# NetApp StorageGRID和大数据分析

## NetApp StorageGRID用例

NetApp StorageGRID对象存储解决方案可提供可扩展性、数据可用性、安全性和高性能。各种规模和各行各业的组织都在广泛的使用情形中使用StorageGRID S3。让我们来了解一些典型场景：

大数据分析： StorageGRID S3常用作数据湖、企业可在其中存储大量结构化和非结构化数据、以便使用Apache Spark、Splunk Smartstore和DREMIO等工具进行分析。

数据分层： NetApp客户使用ONTAP的FabricPool功能在高性能本地层之间自动将数据移动到StorageGRID。在将冷数据保留在低成本对象存储上的同时、将昂贵的闪存存储释放出来、以存储热数据。这样可以最大限度地提高性能并节省成本。

*数据备份和灾难恢复：*企业可以使用StorageGRID S3作为可靠且经济高效的解决方案来备份关键数据并在发生灾难时进行恢复。

应用程序的数据存储： StorageGRID S3可用作应用程序的存储后端，使开发人员能够轻松地存储和检索文件、图像、视频和其他类型的数据。

内容交付： StorageGRID S3可用于存储静态网站内容、媒体文件和软件下载并提供给全球用户、利用StorageGRID的地区分布和全局命名空间实现快速可靠的内容交付。

数据分层： NetApp客户使用ONTAP FabricPool功能在高性能本地层之间自动将数据移动到StorageGRID。在将冷数据从低成本对象存储中随时可用的同时、通过层化释放昂贵的闪存存储来存储热数据。这样可以最大限度地提高性能并节省成本。

数据归档： StorageGRID提供不同的存储类型、并支持分层到公共长期低成本存储选项、使其成为出于合规性或历史目的需要保留的数据归档和长期保留的理想解决方案。

对象存储用例

[StorageGRID用例图、宽度=396、高度=394]

在上述情形中、大数据分析是最热门的使用情形之一、其使用量呈上升趋势。

## 为什么选择StorageGRID解决数据湖问题?

- 增强协作—利用行业标准API访问实现大规模共享多站点、多租户
- 降低运营成本—通过一个自我修复型自动化横向扩展架构简化运营
- 可扩展性—与传统的Hadoop和数据仓库解决方案不同、StorageGRID S3对象存储可将存储与计算和数据分离、从而使企业能够随着增长扩展存储需求。
- 耐用性和可靠性—StorageGRID的耐用性高达99.9999999%、这意味着存储的数据能够高度抵御数据丢失。它还提供高可用性、确保数据始终可访问。
- 安全性—StorageGRID提供各种安全功能、包括加密、访问控制策略、数据生命周期管理、对象锁定和版本控制、以保护S3存储分段中存储的数据

- StorageGRID S3数据湖*

[StorageGRID数据报时示例、宽度=不等、高度= 345]

## 哪种数据仓库或数据湖最适合S3对象存储

NetApp通过三个数据仓库/湖屋生态系统(Hive、Delta湖和德雷米奥)对StorageGRID进行了基准测试。 "Apache iceberg：权威指南" 简要介绍了数据仓库和数据湖屋以及这两种架构的利弊。

- 基准测试工具- TPC-DS - https://www.tpc.org/tpcds/
- 大数据生态系统
  - 一个由5个VM组成的集群、每个VM具有128 G RAM和24个vCPU、SSD存储用于系统磁盘
  - 采用Hive 3.1.3的Hadoop 3.3.5 (1个名称节点+ 4个数据节点)
  - 采用Spark 3.2.0 (1个主服务器+ 4个员工)和Hadoop 3.3.5的Delta Lake
  - d不良V23 (1个主服务器+ 4个执行器)
- 对象存储
  - NetApp® StorageGRID® 11.5，带有3个SG6060 + 1个SG1000负载平衡器
  - 对象保护—2个副本
- 数据库大小为1000 GB
- 在所有3个生态系统上禁用缓存、以便在每个查询测试中获得一致的结果。

TPC-DS附带99个复杂的SQL查询、用于查询基准测试。我们测量了完成所有99个查询所需的总分钟数、并通过细分S3请求的类型和数量来深入分析结果。下面的第一个表显示了所有99个查询的总持续时间、第二个表汇总了每个生态系统发送到StorageGRID的S3请求的数量和类型。

**TPC-DS查询结果**

| 生态系统 | 配置 | 三角洲湖 | **Dremio** |
|---|---|---|---|
| 存储层 | NetApp® StorageGRID® | NetApp® StorageGRID® | NetApp® StorageGRID® |
| 驱动器类型 | HDD | HDD | HDD |
| 表格格式 | 镶木地板 | 镶木地板 | 镶木地板[1] |
| 数据库大小 | 1000 G | 1000 G | 1000 G |
| TPCDS 99查询+ 总分钟数 | 1084 [2] | 55 | 47.1. |

[1]测试了镶木地板和冰山一角表格式，结果相似。

[2] Hive无法完成查询编号72。

**TPC-DS查询- S3请求细分**

| **S3**请求 | 配置 | 三角洲湖 | **Dremio** |
|---|---|---|---|
| 获取 | 1、117184 | 2、074、610 | 4、414、227 |

| S3请求 | 配置 | 三角洲湖 | Dremio |
| --- | --- | --- | --- |
| 观察：+ 所有范围GET | 从32 MB对象中获取2 KB到2 MB的80%范围、每秒50到100个请求 | 73%的范围从32 MB对象开始低于100 KB、每秒1000到1400个请求 | 从256MB对象获取90% 1M字节范围、2000到2300个请求/秒 |
| 列出对象 | 312、053 | 24、158 | 240 |
| 头部+ (不存在的对象) | 156、027 | 12、103 | 192. |
| 头部+ (存在的对象) | 982、126 | 922732 | 1、845 |
| 请求总数 | 2. | 3、033、603 | 4、416、504 |

从第一张桌子上、我们可以看到Delta Lake和德雷米奥比Hive快得多。从第二个表中、我们注意到Hive发送了大量S3列表对象请求、这在所有对象存储平台中通常都很慢、尤其是在处理包含许多对象的分段时。这会显著增加整体查询持续时间。另一个观察结果是、在Hive中、德尔米奥能够并行发送大量GET请求、每秒2000到2、300个请求、而每秒50到100个请求。Hive和Hadoop S3A模拟标准文件系统会导致S3对象存储运行减速。

要将Hadoop (无论是在HDFS还是S3对象存储上)与Hive或Spark结合使用、需要具备有关Hadoop和Hive或Spark以及每个服务中的设置如何交互的广泛知识—它们共同具有1000多个设置。这些设置通常是相互关联的、不能单独更改。要找到要使用的设置和值的最佳组合、需要花费大量时间和精力。

dremio是一种数据湖引擎、它使用端到端Apache Arrow(阿帕奇箭头)来显著提高查询性能。Apache Arrow"提供标准化的列式内存格式、可实现高效的数据共享和快速分析。ARrow采用不受语言限制的方法、旨在消除数据序列化和反序列化的需求、从而提高复杂数据流程和系统之间的性能和互操作性。

在很大程度上、Mirio的性能取决于其集群的计算能力。虽然desmio会使用Hadoop的S3A连接器建立S3对象存储连接、但不需要使用Hadoop、并且desmio不会使用Hadoop的大多数FS.S3A设置。这样、无需花费时间学习和测试各种Hadoop S3A设置、即可轻松调整德米奥的性能。

根据此基准测试结果、我们可以得出结论、针对基于S3的工作负载进行优化的大数据分析系统是一个主要性能因素。在使用S3存储时、由于使用的是Hive、因此、使用此解决方案可以优化查询执行、高效利用元数据并提供对S3数据的无缝访问、从而获得比Hive更高的性能。请参见此部分 "页面。" 使用StorageGRID配置d不良S3数据源。

请访问以下链接、详细了解StorageGRID和德莱米奥如何协同工作来提供现代化且高效的数据湖基础架构、以及NetApp如何从Hive + HDFS迁移到德莱米奥+ StorageGRID来显著提高大数据分析效率。

- "借助NetApp StorageGRID提升大数据的性能"
- "借助StorageGRID和d处 米奥打造现代化、功能强大且高效的数据湖基础架构"
- "NetApp如何利用产品分析重新定义客户体验"

# Hadoop S3A调整

Hadoop S3A连接器有助于在基于Hadoop的应用程序和S3对象存储之间实现无缝交互。在使用S3对象存储时、要优化性能、必须调整Hadoop S3A Connector。在深入介绍调整详细信息之前、我们先大致了解一下Hadoop及其组件。

## 什么是Hadoop?

**Hadoop** 是一个功能强大的开源框架，专为处理大规模数据处理和存储而设计。它支持跨多个计算机集群进行分布式存储和并行处理。

Hadoop的三个核心组件是：

- **Hadoop HDFS (Hadoop分布式文件系统)**：用于处理存储、将数据拆分为块并在节点之间分布。
- **Hadoop MapReredget**：负责将任务划分为较小的区块并并行执行来处理数据。
- **Hadoop yar (Yet Another Resource Neotiator)**： "高效管理资源并计划任务"

## Hadoop HDFS和S3A连接器

HDFS是Hadoop生态系统的重要组成部分、在高效处理大数据方面发挥着关键作用。HDFS可实现可靠的存储和管理。它可确保并行处理和优化数据存储、从而加快数据访问和分析速度。

在大数据处理方面、HDFS在为大型数据集提供容错存储方面表现出色。它通过数据复制来实现这一点。它可以在数据仓库环境中存储和管理大量结构化和非结构化数据。此外、它还可以与领先的大数据处理框架无缝集成、例如Apache Spark、Hive、Pig和Flink、从而实现可扩展的高效数据处理。它与基于Unix (Linux)的操作系统兼容、因此对于更喜欢使用基于Linux的环境进行大数据处理的组织来说、它是理想的选择。

随着数据量逐渐增长、使用自己的计算和存储向Hadoop集群添加新计算机的方法变得效率低下。线性扩展为高效使用资源和管理基础架构带来了挑战。

为了应对这些挑战、Hadoop S3A连接器可针对S3对象存储提供高性能I/O。使用S3A实施Hadoop工作流有助于将对象存储用作数据存储库、并将计算和存储分开、进而使您能够独立扩展计算和存储。分离计算和存储还可以让您将适当数量的资源专用于计算作业、并根据数据集大小提供容量。因此、您可以降低Hadoop工作流的总体TCO。

## Hadoop S3A连接器调整

S3的行为与HDFS不同、某些尝试保留文件系统外观的行为也明显欠佳。要最高效地利用S3资源、必须仔细调整/测试/试验。

本文档中的Hadoop选项基于Hadoop 3.3.5、请参见 "Hadoop 3.3.5 core-site.xml" 所有可用选项。

注意—某些Hadoop FS.S3a设置的默认值在每个Hadoop版本中都不同。请务必查看特定于当前Hadoop版本的默认值。如果未在Hadoop core-site.xml中指定这些设置、则会使用默认值。您可以使用Spark或Hive配置选项在运行时覆盖此值。

您必须访问此页面 "Apache Hadoop页面" 了解每个FS.S3A选项。如果可能、请在非生产Hadoop集群中对其进行测试、以查找最佳值。

您应阅读 "在使用S3A连接器时最大限度地提高性能" 了解其他调整建议。

让我们来探讨一些关键注意事项：

- 。数据压缩*

请勿启用StorageGRID数据压缩。大多数大数据系统都使用字节范围GET、而不是检索整个对象。对压缩对象使用字节范围GET会显著降低GET性能。

- 。S3a提交人*

一般情况下、建议使用magic S3A提交器。请参见此部分 "通用S3A提交器选项页面" 更好地了解Magic committer及其相关的S3A设置。

魔力委员会：

Magic Commonter特别依靠S3Guard在S3对象存储上提供一致的目录列表。

借助一致的S3 (现在是这种情况)、Magic Comm귻 퓍 믨 可以安全地与任何S3存储分段配合使用。

选择和实验：

根据您的使用情形、您可以在Staging Commenter (依赖于集群HDFS文件系统)和Magic Commenter之间进行选择。

尝试这两种方法、确定哪种方法最适合您的工作负载和要求。

总之、S3A委员会为应对持续、高性能和可靠的S3输出承诺这一根本性挑战提供了解决方案。其内部设计可确保高效的数据传输、同时保持数据完整性。

[S3A选项表]

**3.**线程、连接池大小和块大小

- 与单个存储分段交互的每个*S3A*客户端都有自己的专用池，其中包含用于上传和复制操作的开放HTTP 1.1连接和线程。
- "您可以调整这些池大小、以便在性能与内存/线程使用量之间取得平衡"。
- 将数据上传到S3时、数据会划分为多个块。默认块大小为32 MB。您可以通过设置FS.S3a.block.size属性来自定义此值。
- 较大的块大小可通过减少上传期间管理多部件的开销来提高大型数据上传的性能。对于大型数据集、建议值为256 MB或以上。

[S3A选项表]

**4.**多部分上传

S3A提交者*始终*使用MPU (多部分上传)将数据上传到S3存储分段。这是在以下情况下所必需的：任务失败、任务的推测性执行以及提交前作业中止。以下是与多部件上传相关的一些关键规格：

- 最大对象大小：5 TiB (TB)。
- 每次上传的最大部件数：10、000。
- 部件号：范围为1到10、000 (含1到10、000)。
- 部件大小：介于5 MiB和5 GiB之间。值得注意的是、多部分上传的最后一部分没有最小大小限制。

对S3多部件上传使用较小的部件大小既有优点也有缺点。

优势：

- 从网络问题中快速恢复：当您上传较小的部分时、由于网络错误而重新启动失败的上传所产生的影响将降至最低。如果某个部件出现故障、您只需要重新上传该特定部件、而不是整个对象。
- 更好的并行处理：利用多线程或并发连接、可以并行上传更多部件。这种并行处理可提高性能、尤其是在处理大型文件时。

缺点：

- 网络开销：部件较小意味着要上传的部件较多、每个部件都需要自己的HTTP请求。HTTP请求越多、启动和完成单个请求的开销就越大。管理大量小部件可能会影响性能。
- 复杂性：管理订单、跟踪部件和确保上传成功可能会非常繁琐。如果需要中止上传、则需要跟踪并清除已上传的所有部件。

对于Hadoop、建议对fs.s3a.multipart.size使用256MB或以上的部件大小。请始终将FS.S3a.mutlpart.threshold"值设置为2 x FS.S3a.multipart.size值。例如、如果fs.s3a.multipart.size = 256M、则fs.s3a.mutlifpart.threshold"应为512M。

对大型数据集使用较大的零件大小。根据您的特定使用情形和网络条件、选择一个能够平衡这些因素的部件大小非常重要。

多部分上传是 "三步流程"：

1. 上传已启动、StorageGRID将返回一个上传ID.
2. 对象部件将使用上载-id进行上载。
3. 上传所有对象部件后、发送包含上传id的完整多部分上传请求。StorageGRID根据上传的部分构建对象、客户端可以访问该对象。

如果未成功发送完整的多部件上传请求、则这些部件将保留在StorageGRID中、不会创建任何对象。作业中断、失败或中止时会发生这种情况。这些部件将保留在网格中、直到多部件上传完成或中止、或者如果上传启动后15天、StorageGRID会清除这些部件。如果一个存储分段中有许多(几百到几百万个)正在进行的多部分上传、则当Hadoop发送'list-multipart-Uploads'(此请求不按上传ID筛选)时、此请求可能需要很长时间才能完成、或者最终超时。您可以考虑使用适当的FS.S3a.mutlpart.purge值将FS.S3a.multipart.purge.age设置为true (例如、5到7天、不要使用默认值86400、即1天)。或者联系NetApp支持部门调查情况。

[S3A选项表]

**5.**缓冲区写入数据存储在内存中

为了提高性能、您可以在将写入数据上传到S3之前将其缓冲在内存中。这样可以减少小型写入次数并提高效率。

[S3A选项表]

请记住、S3和HDFS的工作方式各不相同。要最有效地利用S3资源、必须仔细调整/测试/实验。