

I/O 情報を利用したオブジェクトストレージ向けの 高速化プロキシの開発

内田 哲彰[†] 亀井 仁志[†]
香川大学[†]

1. はじめに

近年、デジタルデータの急激な増加に伴い、データ保存の効率化が重要な課題となっている。特に、大規模なデータセンターやクラウドストレージでは、データの保存とアクセス速度の最適化が求められている。本稿では、I/O 情報を利用したオブジェクトストレージ向けの高速化プロキシを開発し、データアクセスの効率化を図る方法を提案する。

2. 大規模データの保存と課題

現代の情報社会では、データ量は指数関数的に増加し、2025 年には全世界で生成されるデータ量は約 163 ゼタバイトに達すると予測されている [1]。このデータの大部分は非構造化データで、従来のファイルベースのストレージシステムでは効率的な管理が困難である。そこで、オブジェクトストレージが用いられる。

オブジェクトストレージは、非構造化データの保存に適した技術として広く採用されているが、データ量の増加に伴い、性能に関する問題が起きている。特に、データの読み書き速度とアクセス効率の向上は、大規模データ管理における不可欠な要件である。

3. 提案システム

3.1. ファイルシステムの性能差の利用

ファイルシステム(FS)は種類ごとに機能や性能が異なる。オブジェクトストレージのバックエンド FS によって、オブジェクトストレージ性能が異なることが考えられる。

例えば、ext3 は、小さいサイズのファイルを扱う場合に適する。これは、ext3 のファイルのメタデータ管理構造が単純であり効率的に管理できるためである。

一方、ext4 は、大きいサイズのファイルを扱う場合には、ext3 より性能が高い。これは、ext4 が大きなファイルに適するメタデータ管理構造を採用しているためである。メタデータ管理オーバーヘッドを低減し、大きいサイズのファイルを使用する際のパフォーマンスを向上している。

3.2. システム構成

本稿で提案するオブジェクトストレージは、I/O を高速化させるプロキシと、複数のバックエンド FS を持つオブジェクトストレージから構成される(図 1)。

書き込み時は、クライアントからの I/O リクエスト情報を用いて、プロキシの振り分けモジュールが適切な FS を選択する。その際、操作対象 FS の情報をデータベースに保存する。FS の選択は閾値に基づいて決定する。閾値は、これまで処理したオブジェクトサイズの平均値として、一方の FS に処理が偏らないようにする。

読み込み時と削除時は、クライアントからの I/O リクエスト情報を基にデータベースに保存された情報を参照し、適切な FS を選択する。選択した FS に格納されたオブジェクトに対して、操作を行う。

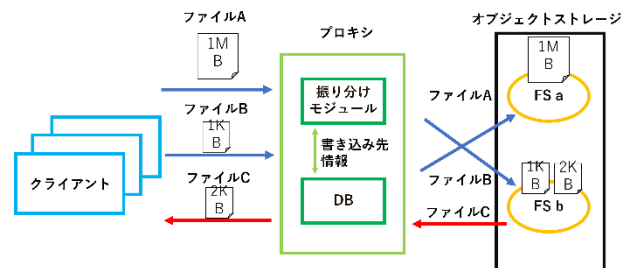


図 1 提案するオブジェクトストレージ

4. 評価

4.1. 評価環境

図 2 に評価構成を示す。ストレージノード、クライアントノード、プロキシノードとして、Microsoft Azure に 3 台の仮想マシンを設置した。また、表 1 に評価環境の詳細を示す。

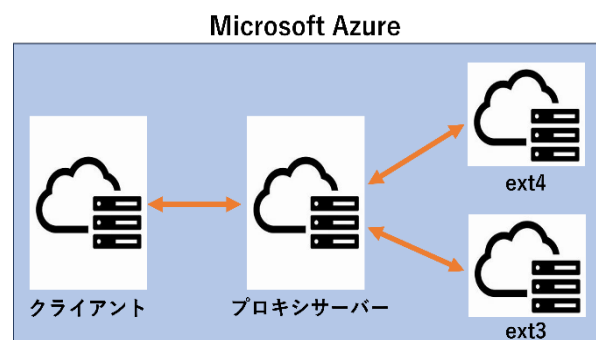


図 2 評価構成

Development of Acceleration Proxy for Object Storage using I/O Information

[†]Tetsuaki Uchida, Hitoshi Kamei · Kagawa University

表 1 評価環境

| | オブジェクト ストレージ | クライアント | プロキシサーバー |
|--------|---|--------|----------|
| OS | ubuntu 22.04 | | |
| CPU | Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz (Standard D2s v3 (2 vcpu 数、8 GiB メモリ)) | | |
| メモリ | 8GiB | | |
| ネットワーク | 960 Mbits/sec | | |
| ディスク | Standard HDD LRS 128GB | | |

ストレージノードではオブジェクトストレージである MinIo[2]を、2つの Docker コンテナ内で動作させている。それぞれ、ext3, ext4 を使用したディスクを使用している。

クライアントノードでは、オブジェクトストレージを性能評価するためのベンチマークツールである CosBench[3]と、自作した計測用のスクリプトを動作させ、評価した。

4.2. 評価結果

(1) FS 別評価

ext4 と ext3 を対象に、CosBench を用いてオブジェクトストレージである MinIo の性能を計測した。この測定では、プロキシを使用せず、FS の性能差を明らかにした。

図 3 に、オブジェクト書き込み時のスループットを示す。縦軸はスループットを表し、横軸は書き込みオブジェクトのサイズを表す。1KB, 2KB では、わずかに ext3 の性能が良く、128KB 以上では、ext4 の性能が高かった。

(2) 基本評価 (CosBench)

CosBench を用いて、オブジェクト振り分けプロキシを介し、MinIo の性能を計測した。100KB 以下のオブジェクトは ext3, 1000KB のオブジェクトを ext4 で操作した。図 4 にプロキシを使用した場合と使用しなかった場合のスループットを示す。縦軸はスループットを表し、横軸は書き込みオブジェクトのサイズを表す。プロキシを使用した場合、性能が大きく低下した。

(3) 書き込み評価 (w-bench)

スクリプトを用い、プロキシを介して、30 秒間、書き込むオブジェクトサイズを、1 KB, 10KB, 100KB, 1000KB, 2000KB, 3000KB のうちからランダムに選択し、MinIo に格納し、スループットを計測した。図 5 にプロキシを使用した場合と使用しなかった場合のスループットを示す。縦軸はスループットを表し、横軸はプロキシの有無と種類を表す。プロキシを使用した場合、性能が低下した。また、オブジェクト振り分けの閾値を動的に設定した場合、さらに性能が低下した。

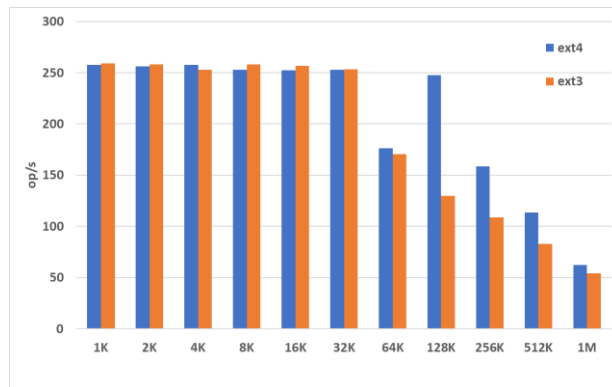


図 3 オブジェクト書き込み時のスループット

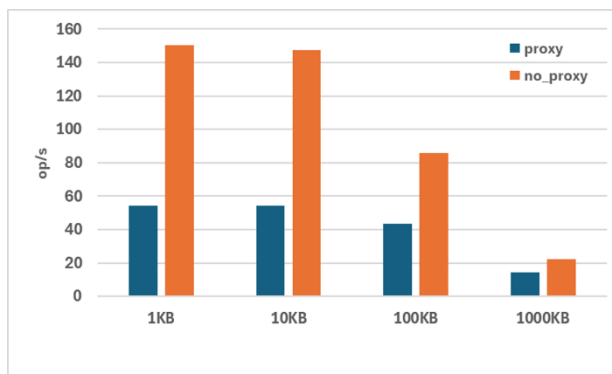


図 4 プロキシ使用時のスループット

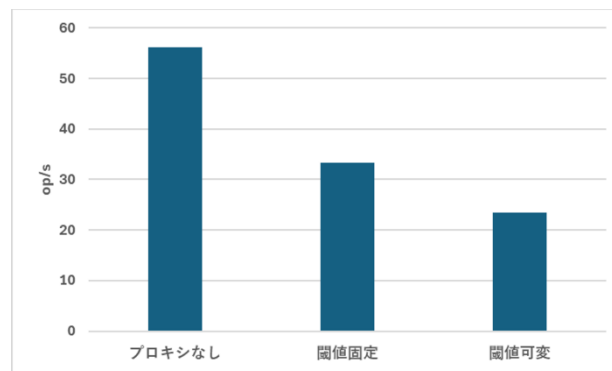


図 5 ランダムサイズ時のスループット

5. 考察とまとめ

本稿は、オブジェクトストレージの高速化のため、適切な FS を選択するプロキシを提案し、実装して評価した。その結果、プロキシのオーバーヘッドによる性能低下がみられた。今後、オブジェクトストレージに提案機能を組み込むことで、性能の向上を目指す。

参考文献

[1] “IoT/非 IoT データ生成量は 2025 年 163 兆ギガバイトに増大--IDC 調査”, ZDNET Japan, <https://japan.zdnet.com/article/35110412/>, (参照 2024-01-03)
 [2] MinIo : <https://min.io/>, (参照 2024-01-03)
 [3] CosBench : <https://github.com/intel-cloud/cosbench>, (参照 2024-01-03)