Vol.2017-HPC-160 No.25
2017/7/27

クラウドコールドストレージに対する大量データ格納の試行と評価

吉田 浩[†1]　合田 憲人[†1]　上田 郁夫[†2]　原 隆宣[†2]
小杉 城治[†3]　森田 英輔[†3]　中村 光志[†3]

概要：近年，アクセス頻度が比較的低いデータの長期保管を想定して，パブリッククラウドによるコールドストレージの提供が進んでおり，大量のデータの長期保管の TCO 低減やストレージ管理の負担軽減の観点から，その活用が注目されている．筆者らは，大量の研究データの長期保管にパブリッククラウドを活用するベストプラクティスを確立してゆくための基礎情報収集を目的として，複数の商用パブリッククラウドが提供するコールドストレージに対して，実際に科学研究で使われている実験データやシミュレーションデータ，観測／解析データを含め最大 1 ペタバイトのデータを格納することを試行した．本論文では，その結果について，性能や運用性などの観点から報告する．

キーワード：クラウド，コールドストレージ

An Experiment and Evaluation on Storing Large Amount of Data in Cloud Cold Storage Services

HIROSHI YOSHIDA[†1]　KENTO AIDA[†1]
IKUO UEDA[†2]　TAKANORI HARA[†2]
GEORGE KOSUGI[†3]　EISUKE MORITA[†3]　KOHJI NAKAMURA[†3]

Abstract: Recently major cloud providers have started providing cold storage services which target to store data with relatively low access frequency for a long period as a part of their public cloud services. The adoption of cold storage services should be considered in order to reduce the total cost of ownership and the labor of storage management of maintaining large amount of data for a long period. In order to acquire the fundamental information which helps establishing best practices of cold storage adoption for research data, the authors experimented on commercially available public cold storage services by storing large amount of data up to 1PB as well as storing and accessing the actual research data. This report describes the results of the experiments from the viewpoint of performance and manageability.

Keywords: Cloud, Cold storage.

1. はじめに

コールドストレージについて，ストレージ関連技術の国際標準化団体 SNIA は「プライマリストレージより少なくとも一桁低いコストで，コールドデータを格納するデータストレージデバイス，システム或いはサービス．コールドストレージの特長は,性能より低コストを実現するための，大容量，省エネルギー，長期データ保管である」と説明している[1]．なお，コールドデータとは，同じく SNIA の定義では，「頻繁にはアクセスされないデータ」を意味する．

上記の説明には「サービス」が含まれているが，パブリッククラウドにおいても，IaaS の一環として，コールドストレージサービスを提供することが一般化している．これは，アクセス頻度が比較的低いデータの長期保管を主な用途として想定した設計・運用・商品化がなされており，容量当たりの保管料金が標準仕様のクラウドストレージサービスと比較して相対的に低いことが特徴となっている．このようなサービスの用途の一つとして，大量の研究データの長期保管が考えられ，データの保管に関わる TCO の低減や，データを保管するストレージシステムの運用管理の負担軽減といった効果が期待される．しかし一方で，コールドストレージ（以下，特に断らない限り「コールドストレージサービス」を「コールドストレージ」と呼ぶ）には，データの読込み時間や課金の面で標準仕様のストレージサービスと異なる特有の仕様があり，現実的にどこまで使えるかという懸念が生じるのも事実である．

クラウドの科学技術計算分野への適用可能性について，計算資源を対象として評価した報告[2][3][4][5][6]はあるが，

†1 情報・システム研究機構 国立情報学研究所
National Institute of Informatics
†2 高エネルギー加速器研究機構 素粒子原子核研究所
Institute of Particle and Nuclear Studies
†3 自然科学研究機構 国立天文台
National Astronomical Observatory of Japan

a) 本論文に記載されている社名，商品またはサービスの名称等は，各社の商標または登録商標です．

コールドストレージの同分野への適用可能性については明らかではない．この状況をふまえて，本報告では，複数の商用パブリッククラウドで提供されるコールドストレージについて，基礎的なベンチマークテストと合わせて，高エネルギー物理学および天文学の分野における実際の研究データとアプリケーションを用いた実験を行った結果を述べる．実験を通して，研究データをコールドストレージに保管するかどうかの判断や，クラウドを含めたデータ保管のストレージアーキテクチャ設計の一助となる実際的な情報を得ることを目的としている．

2. クラウドにおけるコールドストレージ

クラウドにおける汎用のストレージサービスとしては，ハードディスクとして見えるブロックストレージと，任意長のバイト列の集合として見えるオブジェクトストレージが一般的であるが，コールドストレージは，オブジェクトストレージと論理構造（任意長バイト列であるオブジェクトとオブジェクトの入れ物の 2 階層構造），API，運用が同等あるいは類似性の高いサービスとして提供されることが多い．すなわち，コールドストレージと相対するオブジェクトストレージが必ず提供されている（以下「標準仕様のオブジェクトストレージ」と呼ぶ）．これとの比較において，コールドストレージは，以下のような特徴を持っている．

- 保管料金が標準仕様のオブジェクトストレージサービスよりも数分の一から一桁低い．
- サービスによっては，標準仕様のオブジェクトストレージサービスと比較して，アクセス性能が異なる，可用性が低いといった非機能要件における差異がある．
- サービスによっては，データの読込み前の復元処理が必要であり，1 時間から数時間の待ち時間を要する．
- データの書込み・検索・復元・読込みなどの料金が，標準仕様のオブジェクトストレージサービスより高く設定されている．サービスによっては，少量データの処理に対して特に課金されることもある．
- サービスによっては，データの最低保持期間が規定されている．最低保持期間満了前にデータを削除した場合，最低保持期間分の保管料金が課金される．

本実験は，次の 4 種類の商用パブリッククラウドのコールドストレージを対象とした．表 1 にその特徴をまとめる．

- Amazon Web Services (AWS) Glacier [7]
  および S3 Infrequent Access [8] （以下 S3 IA）
- Google Cloud Platform (GCP) Coldline Storage [9]
  （以下 Coldline）
- Microsoft Azure block BLOB Cool [10]（以下 Cool）
- Oracle Cloud Archive Storage Service [11]

表 1 に示したサービスの特徴から，クラウドのコールドストレージは，以下のように大別できることがわかる．

表 1　実験の対象としたコールドストレージの仕様の概要
（2017 年 5 月 1 日現在）

サービス名	AWS Glacier	AWS S3 Infrequent Access	GCP Coldline Storage	Azure block BLOB Cool	Oracle Archive Storage Service
保管料($/GiB･月)	0.005	0.019	0.010	0.015	0.001
対応する標準仕様オブジェクトストレージ	AWS S3		GCP Regional Storage	Azure Block BLOB Hot	Oracle Cloud Storage Service
[参考]標準仕様オブジェクトストレージの保管料（$/GiB･月）	0.025		0.023	0.024	0.020
書出しリクエスト料金の割増	あり	あり	あり	あり	あり注1
書出しデータ従量課金の割増	なし	なし	なし	あり	なし
読込み前の復元操作と所要時間	必要 標準注2 3～5 時間	不要	不要	不要	必要 最大 4 時間
読込みリクエスト料金の割増	あり	あり	あり	あり	あり
読込みデータ従量課金の割増	あり注2	あり	あり	あり	あり
最低保有期間	90 日	なし	90 日	なし	90 日
標準仕様オブジェクトストレージとの可用性の差	可用性の規定なし	あり	あり	あり	不明
標準仕様オブジェクトストレージとのAPI/操作互換	なし	あり	あり	あり	一部なし
備考			AWS S3 互換		Swift 互換

注 1: 10MiB 以下のオブジェクトについてのみ

注 2: 復元処理時間（高速，標準，低速）に応じ 3 段階の課金体系

- 標準のオブジェクトストレージとは別サービスとして提供されるもの．操作や API は基本的に別体系となり，互換性は小さい． 例．AWS Glacier
- 標準のオブジェクトストレージのオプションとして提供されるもの．操作や API は同等であり，多くの場合，オブジェクトあるいはコンテナ（AWS，GCP ではバケット，Azure，Oracle，OpenStack ではコンテナと呼ぶが，AWS や GCP 固有の記述を除いて，本報告ではコンテナに統一する）作成時にコールドストレージであることを指定する．例．AWS S3 IA，GCP，Azure
- 両者の中間的なもの． 例．Oracle

コールドストレージが標準のオブジェクトストレージのオプションである場合，課金以外の性能などの点で実質的にどのような差異が生じ得るのかを把握することが，コールドストレージを利用する上での一つの着眼点となる．

3. 実験方法

3.1 実験内容

実験は，以下の 3 つのシナリオに従って実施した．

(1) **シナリオ 1：コールドストレージの基礎データ収集**

最大 1TiB のデータの書出しと読込みを行い，スループットや復元操作の所要時間などの性能を測定する．

表 2　実験で使用した研究データとアプリケーション

データ	データ量	アプリケーション	提供元	備考
高エネルギー物理学（Belle II 実験の物理シミュレーションデータ）	633GiB 1000 ファイル（ほぼ同一サイズ，600～700MiB）	解析支援ソフトウェア環境 BASF2 (Belle II Analysis Frame-work)経由のファイル読込み	高エネルギー加速器研究機構	
天文学（ALMA 電波望遠鏡の観測／解析データ）	58.5TiB 138 万ファイル（1MiB 以下～100 GiB 以上に分布）	アーカイブシステム NGAS (Next Generation Archive System)	自然科学研究機構 国立天文台	他に Oracle DB 2.7TiB

合わせて，これらのアクセスの際の運用性や課金についても調査する．なお，標準仕様のオブジェクトストレージサービスについても同様の測定を行って比較する．

(2) シナリオ 2：大量データの格納

300TiB から最大 1PiB までのデータを実際にコールドストレージに書き出し，大量データを格納する際の性能，課金，運用性などを評価し，大規模利用に関する感触を得る．

(3) シナリオ 3：現実の研究データとアプリケーションによるコールドストレージ利用の試行

高エネルギー物理学分野および天文学分野において実際に研究に使われているデータをコールドストレージに格納し，さらに当該分野で使用しているアプリケーションまたはそのデータ入出力部を抽出したものを使ってアクセスする．対象のデータとアプリケーションの概要を表 2 に示す．

3.2 実験の実施方針

コールドストレージでは REST API が提供されており，これらを直接使うことによって，データ長，同期・非同期処理，並列度などの条件を柔軟に設定して実験を行うことも可能である．しかし，今回は，通常の利用者が最初の利用にあたってまず採用する方法という観点から，極力，プロバイダが提供する GUI, CLI, 言語ライブラリを使用することとした．これらの CLI や言語ライブラリでは，スレッド数などの並列度やデータアクセス時のセグメント長などのチューニングパラメタを指定できることが多いが，これも最初はデフォルト値を使用して実験を行い，何らかの不具合が生じた場合に変更を検討するという方針をとった．

3.3 実験の全体像

4 種のパブリッククラウドからなる実験環境を構築した．日本国内にデータセンタ（リージョン）を持つクラウドでは，それを利用した．ストレージに対するアクセス元としては，各クラウドが提供する同一リージョンの計算環境内の仮想マシン（以下 VM）と国立情報学研究所（以下 NII）のオンプレミスの物理サーバを用意し，必要なアプリケーションやデータを配置した．環境の全体像を図 1 に示す．

本実験では，NII のサーバとクラウド間の通信はインターネット経由で行った．通信速度の目安として，NII のサーバおよび同一リージョン内の VM から，クラウドストレージの API エンドポイントに ping を発行して測定した応答時間を表 3 に示す．全般的に NII からの通信よりもクラウド内からの通信のほうが速く，さらに，Oracle は米国データセンタへの通信となるため，時間が長くなっている．

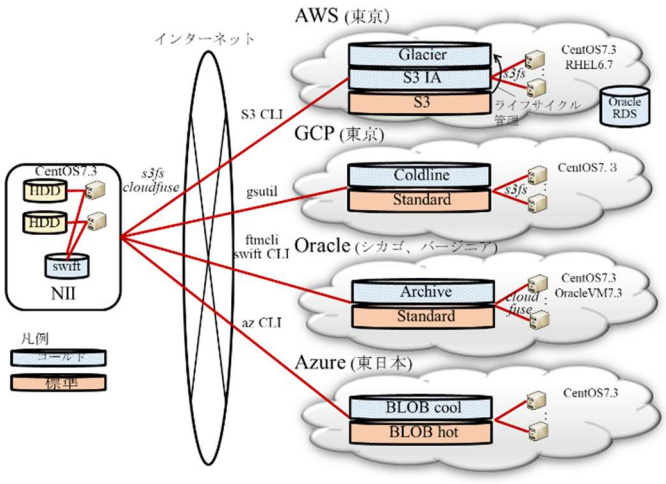

図 1　実験環境の全体像

表 3　実験環境における ping 応答時間（単位：ミリ秒）

発行元＼発行先	NII	AWS	GCP	Oracle
NII	0.17	3.7	2.8	150.7
AWS		2.8		
GCP			0.2	
Oracle				25.3

注 1：パケットデータサイズは 64 バイト

注 2：Azure のエンドポイントはセキュリティ上 ping を通さない設定となっていると見られ，測定対象からはずした．

4. 実験結果と考察

実験の大半は 2016 年 11 月から 2017 年 3 月にかけて実施し，一部の追加実験を 2017 年 4 月から 6 月に実施した．

4.1 シナリオ 1：基礎データ収集

オブジェクトストレージに対しては，Cosbench [12]などのベンチマークがしばしば使用される．しかし，コールドストレージまで対象範囲を拡大した場合，API 非互換への対処や復元処理の追加が必要となることから，本実験では，独自のベンチマークを実施した．すなわち，コールドストレージと，対応する標準仕様のストレージサービスに対し，8GiB と 8MiB のオブジェクトの書出しと読込みをクラウド内の計算環境および NII のサーバから行い，スループットを測定した．書出し・読込み処理は，各クラウドで提供されている 1 オブジェクト単位の書出し・読込みを行う CLI を呼び出すシェルスクリプトを作成して実施した．

(1) クラウド内におけるデータ書出し

8GiB，8MiB の 2 種類の長さのオブジェクトを別々の

表 4　データの書出し性能

クラウド	ストレージ	種別	8GiB	8MiB [注1]
			スループット (MiB/秒)	スループット (MiB/秒)
AWS	Glacier	Cold	21.3	7.0
	S3 IA	Cold	[注2)] 92.3	8.4
	S3	標準	92.5	8.7
GCP	Coldline	Cold	52.3	3.8
	Regional	標準	51.8	4.0
Azure	BLOB Cool	Cold	[注3)] 50.5	5.4
	BLOB Hot	標準	[注3)] 57.6	5.4
Oracle	Archive	Cold	88.2	2.7
	Cloud Storage	標準	99.2	2.6

注 1: Oracle と Azure の 8MiB オブジェクト書出し量は 32GiB

注 2: 値の変動あり （2 回目 83.2MiB/秒）

注 3: Azure は，1 オブジェクトに対する転送速度の最大値を 60MiB/秒とする仕様となっている（2017 年 6 月現在の値）.

表 5　データの復元・読込み性能

クラウド	ストレージ	種別	8GiB		8MiB	
			復元時間 (分)	スループット (MiB/秒)	復元時間 (分)	スループット (MiB/秒)
AWS	Glacier	Cold	242	80.6	240	10.2
	S3 IA	Cold	-	112.7	-	11.8
	S3	標準	-	112.8	-	12.1
GCP	Coldline	Cold	-	151.7	-	[注1)] 5.0
	Regional	標準	-	153.8	-	5.0
Azure	BLOB Cool	Cold	-	69.8	-	5.8
	BLOB Hot	標準	-	67.6	-	5.7
Oracle	Archive	Cold	63	29.7	60	3.9
	Cloud Storage	標準	-	25.6	-	6.2

注 1: 値の変動あり（最低 3.5）

コンテナに書き出す．コンテナサイズは 1TiB を標準としたが，時間の制約により縮小した場合もある．測定されたスループットの値を表 4 に示す．なお，測定を 2 回以上実施した場合はもっともよい値を記載し，毎回の変動が特に大きい場合は，その旨を注記した．

すべてのコールドストレージにおいて，オブジェクト長が大きい場合にスループットが向上している．書出しスクリプトの内部で呼び出している CLI は，長大オブジェクトを複数のセグメントに分割して並列に書き出すという実装になっているものもあり，その効果もあると考えられる．また，コールドストレージが標準オブジェクトストレージのオプションとして提供されているサービスにおいては，両者の性能はほぼ同等であると言える．

(2) クラウド内におけるデータ読込み

8GiB，8MiB の各コンテナのオブジェクトを読み込む．読み込むオブジェクト数は，8GiB は 16 個 ，4MiB は 512 個を標準としたが，時間の制約により削減した場合もある．復元処理が必要なサービスではその時間も合わせて測定した．復元時間とスループットに関する結果を表 5 に示す．

オブジェクト長が大きい場合のスループット向上と，コールドストレージが標準オブジェクトストレージのオプションであるサービスにおいて両者の性能がほぼ同等であることは，書出しの場合と同じである．

読込み前の復元処理が必要なサービス（Glacier, Oracle）において測定された所要時間は，プロバイダが提示する最大値を十分下回るものとなっている．この復元処理は，まず復元リクエストを発行し，応答として返された復元ジョブの識別子を使用して終了をポーリングするという一連の処理を API あるいは CLI で行うものである．しかし，CLI とは言え，curl コマンドを使用して http リクエストをほとんどそのまま入力するような仕様である場合もあり，手動で操作する運用には向かない．アーカイブシステムを上位に構築するための部品としての性格が強いと考えられる．

表 6　オンプレミスからのアクセス性能

(i)　書出し

クラウド	ストレージ	種別	8GiB	8MiB [注1]
			スループット (MiB/秒)	スループット (MiB/秒)
AWS	Glacier	Cold	14.7	6.7
	S3 IA	Cold	99.1	6.9
	S3	標準	99.1	6.6
GCP	Coldline	Cold	52.8	4.2
	Regional	標準	51.1	4.3

(ii)　読込み

クラウド	ストレージ	種別	8GiB		8MiB	
			復元時間 (分)	スループット (MiB/秒)	復元時間 (分)	スループット (MiB/秒)
AWS	Glacier	Cold	243	76.0	240	7.1
	S3 IA	Cold	-	112.8	-	7.8
	S3	標準	-	110.7	-	7.8
GCP	Coldline	Cold	-	77.3	-	4.7
	Regional	標準	-	76.4	-	6.1

(3) オンプレミスからのデータアクセス

AWS および GCP については，NII 内のサーバからのアクセス性能を(1)と同じ方法で測定した. 結果を表 6 に示す.

クラウド内の VM からのアクセスと比較して，スループットの低下が見られるサービスがあり，これはインターネット経由のアクセスの影響と考えられる．一方で，クラウド内からのアクセスとほぼ同等の性能と見なせる場合もある．これは，1）律速要因がネットワークよりもクラウド内部のストレージシステムに存在する，あるいは，2）リージョン内においても計算環境からのアクセス経路と外部のインターネット経由のアクセス経路が同等であるようなネットワーク構成となっている，といったクラウド基盤の実装が関わっていると推察される．

4.2 シナリオ 2：大量データ格納

本シナリオでは，8GiB のオブジェクトを，同一クラウド内の計算環境から 1 ないし複数個のコンテナに書き出す方法を標準とした. 書出しを加速するために, VM および VM 内の書出し処理を行うスレッドは可能な限り並列化した.

表 7　大量データの書出し性能

ストレージ	AWS Glacier	GCP Coldline	Azure Cool	Oracle Archive	注 1)
オブジェクト長	8GiB	8GiB	8GiB	8GiB	平均 648MiB
オブジェクト数	38,912	51,470	128,989	33,354	1,161,000
オブジェクト量(TiB)	304	402	1008	261	718
アップロード時間 (VM 数*時間)	792.6	252.3	2051.9	990.5	3832.9
サーバ当りスループット(MiB/秒)	37	155	36	19	27
スレッド当りスループット(MiB/秒)	5	19	12	19	-
VM 数	3	3	4	4	2
スレッド数	8	8	3	1	-

注 1: CLI による最大 8 多重（2VM）のファイル連続書出しを併用

並列書出し処理は，プロバイダが提供するストレージアクセス用 Python ライブラリあるいはストレージアクセスの REST API を直接呼び出すプログラムを作成して実施した．

このようにして，大量データの書出し性能を測定した結果を表 7 に示す．書き出した最大容量は，費用面の制約からクラウドごとに異なっており，304TiB から 1008TiB である．また，クラウドごとに VM 数，VM 内のスレッド数，書出し対象のコンテナ数が異なっているが，これは，予備的な試行を行った範囲でもっとも性能が高く，かつ安定して書出しが行える値を採用したためである．なお，クラウドによっては，書出しをさらに加速するために，他の並列書出し方法を併用したものもある．

AWS Glacier と GCP Coldline では，シナリオ 1 の 8GiB オブジェクトの書出し結果と比較して，サーバ当りのスループットが向上しており，並列化の効果が現れている．

一方 Azure は，シナリオ 1 の結果よりもスループットが低い．内部処理を分析したところ，Azure で提供されている Python 用ライブラリのデフォルト動作では，64MiB を超えるオブジェクトを 4MiB 単位に分割して http リクエストを発行するようになっていることが判明し，結果として 8MiB オブジェクト書出しのケースに近い性能で書出しが行われたと考えられる．なお，クラウドでは書出しリクエスト回数に応じた課金がなされるが，このケースでは，上記の動作によって，1008GiB の書出しに 2 億 5000 万回以上のリクエストが発行されており，課金額が非常に大きくなった．ペタバイトクラスの大量データを取り扱う場合には，デフォルトの動作がどう行われているかを十分理解して使用しないと，性能はもとより課金にも大きな影響があるというクラウド特有の状況を示していると言える．

4.3 シナリオ 3-1：高エネルギー物理学分野のデータに関する実験

(1) データのアップロード

高エネルギー加速器研究機構（以下 KEK）が保有する Belle II 実験の物理シミュレーションデータに対して，解析アプリケーションの 1 回の処理単位である 1000 ファイル 633GiB を，1 ファイルを 1 オブジェクトに対応させて NII 内のサーバからアップロードした．アップロード方法は，原則としてプロバイダが提供している一括アップロード CLI を使用した．これらの CLI では，アップロード時間を短縮するために，複数のオブジェクトや複数セグメントに分割された長大オブジェクトに対して，複数のスレッドによる並列処理を行っている．その並列度は，コマンドのパラメタとしてチューニング可能なものが多いが，今回の実験では，特に問題がない限り，デフォルト値を使用した．使用した一括アップロード用の CLI とデフォルトの並列度を表 8 に示す．なお，並列度に関しては，クラウドによって表現が異なっている．

表 8　コールドストレージの一括処理用 CLI

ストレージ	CLI 名	デフォルトの並列度
AWS S3 IA	S3 CLI	10 並列リクエスト
GCP Coldline	gsutil	10 スレッド
Azure Cool	Azure CLI 2.0	2 コネクション ※実験では性能向上を図って 16 コネクションで実行
Oracle Archive	ftmcli	15 スレッド
	swift CLI	オブジェクト処理 10 スレッド セグメント処理 10 スレッド

表 9　高エネルギー物理学データのアップロード性能

ストレージ	オブジェクト数	オブジェクト量 GiB)	アップロード時間(時:分)	スループット(MiB/秒)
AWS S3 IA	1000	633	2:25	74.5
GCP Coldline			2:09	84.0
Azure Cool			2:31	71.6
Oracle Archive			注 1) 5:23	33.4
			注 2) 1:39	109.5

注 1: ftmcli 使用　注 2: swift upload 使用

表 9 にアップロード所要時間およびスループットの結果を示す．なお，AWS，GCP，Azure は国内のリージョン（国内のデータセンタ）を使用したが，Oracle はデータセンタが米国にあり，このことがアップロード性能に何らかの影響を及ぼしている可能性がある．

(2) アプリケーションによるデータの読込み

解析アプリケーションのデータ読込み部分を抽出したプログラムにより，データ読込み性能（経過時間）を測定した．これは，KEK で使用している解析支援ソフトウェア環境 BASF2 のライブラリ経由でファイルを読み込むものであり，クラウド内の計算環境と NII のサーバで動作させた．

コールドストレージを含めたクラウドのオブジェクトストレージはコンテナとオブジェクトの 2 階層によって構成され，これらへのアクセスは REST API を使用して行う．一方，BASF2 を含めて既存の研究アプリケーションの多くは，Linux ファイルシステム（POSIX 互換ファイルシステム）を前提として作られている．当面は，このような既存のアプリケーションを修正せずにコールドストレージのオブジェクトにアクセスできるようにする手段が必要であり，本実験では，コンテナをファイルシステムとしてマウント

表 10　コールドストレージのファイルアクセス

ストレージ	方法
AWS S3 IA	s3fs [13]（OSS）を使用しバケットを FUSE でマウント
GCP Coldline	同上（S3 互換があるため）
Oracle Archive	cloudfuse [14]（OSS）を使用しコンテナを FUSE でマウント
Azure Cool	VM のローカルブロックストレージ内に構築した Linux ファイルシステムにダウンロード

表 11　高エネルギー物理学データの読込み性能

クラウド	ストレージ	種別	アクセス元	オブジェクト数	復元時間 (分)	読込み時間 (分)
AWS	Glacier	Cold	クラウド	10	312	3.5
	S3 IA	Cold	クラウド	1000	-	357
	S3 IA	Cold	NII	1000	-	435
	S3	標準	クラウド	10	-	3.4
	S3	標準	クラウド	1000	-	359
GCP	Coldline	Cold	クラウド	1000	-	455
	Coldline	Cold	NII	1000	-	541
Azure	BLOB Cool	Cold	クラウド	1000	151	530
Oracle	Archive	Cold	クラウド	10	67	5.9
	Archive	Cold	NII	10	67	34.6
	Cloud Storage	標準	クラウド	10	-	6.6
	Cloud Storage	標準	NII	10	-	27.9
参考: オンプレミス	KEK 中央計算機システム	-	KEK	1000	-	266
	NII サーバ HDD	-	NII	1000	-	349
	NII 所内 swift 注 1)	-	NII	1000	-	591

注 1: cloudfuse 経由で読込み

する OSS を主に使用した．使用した OSS を表 10 に示す．

復元処理が必要な Glacier と Oracle Archive では，オブジェクト（Glacier ではアーカイブと呼ぶ）に対して GUI（Glacier）や CLI（Oracle）で復元開始指示と完了待合せを手動で行った後に, s3fs または cloudfuse 経由で読み込んだ．そのため，測定対象を 10 個に限定した．なお，Glacier の本来の仕様では，書出し時に付与されたアーカイブ識別子を指定して復元ジョブを実行してからそのジョブ識別子を指定してデータを読み込む．元のファイルのパス名は保存されず，読込み時にも使えない（従って，必要ならば元のファイルとアーカイブ識別子の関係を外付けのデータベースなどで管理することが行われる）．しかし本実験では，復元処理後に s3fs でマウントして元のパス名でアクセスできる必要がある. そこで, 前項のアップロード結果とは別に，S3 に一旦アップロードした後に S3 ライフサイクル管理機能によって Glacier に自動的に移行されたオブジェクトを読み出すことにした．この場合，復元処理は元のパス名を使用して GUI から行うことができ，復元後は S3 オブジェクトとして元のパス名でアクセスできる．

アプリケーションの総データ読込み時間の測定結果を表 11 に示す．クラウド内の VM からアクセスする場合，AWS では，NII のオンプレミスサーバのローカルデイスクにデータを格納した場合とほぼ近い性能が出ている. また，復元処理後の Glacier，S3 IA，標準の S3 ともに，読込み時間はほぼ同等という結果が得られた．ただし，KEK の中央計算機システムの値には及ばず，実用性の点ではまだ課題があると考えられる．一方，GCP は AWS 互換であるものの，s3fs 経由のオブジェクト読込み性能は AWS より低い. Oracle に対して使用した cloudfuse 経由の読込み性能はさらに落ちる．NII 内の swift に対する cloudfuse 経由の読込みも同様の結果であることから，OSS 自体の完成度や，個々のクラウドサービスに対する OSS のチューニング状況が関係していると考えられる．

表 12　天文学データのアップロード性能

ストレージ	オブジェクト数	オブジェクト量 (TiB)	アップロード時間 (時:分)	スループット (MiB/秒)	分割数	標準偏差 (MiB/秒)
AWS S3 IA	1,380,791	58.5	281:06	60.6	8	15.4
GCP Coldline	1,380,791	58.5	174:19	97.8	8	40.5
Azure Cool	955,889	41.2	197:56	60.7	6	6.4
Oracle Archive	注 1) 86,531	8.4	24:02	101.9	2	4.1

注1) 1〜2 万オブジェクトに 1 回程度，タイムアウトが発生した．

クラウド内の VM からのアクセスと比較して，オンプレミスのサーバからのリモートアクセスは，すべての場合において性能が低下している．コールドストレージの実運用の設計においては，このような性能差を意識しながら，データをクラウド内で処理するか／オンプレミスシステムで処理するか，後者の場合，クラウドに対するリモートアクセスを行うか／オンプレミスシステムに一旦ダウンロードするか，といった選択を考える必要がある．

なお，本実験では，コンテナをファイルシステムとしてマウントする OSS を主に使用したが，Oracle では，オブジェクトストレージを NFS でマウントしてアクセスする NAS ソフトウェアアプライアンスが提供されており，クラウド内の計算環境の VM に配備して Archive のフロントエンドとして動作させることが可能である．同社は本機能の利用を推奨しており，今後試行する必要があると考える．

4.4 シナリオ 3-2：天文学分野のデータに関する実験

(1) データのアップロードと書出し

電波望遠鏡の観測／解析データのアーカイブから抽出した約 138 万ファイル 58.5TiB を, NII 内のサーバから 1 ファイルを 1 オブジェクトとしてアップロードした．高エネルギー物理学のデータと同様に，アップロードは表 8 に示す CLI を使用した．表 12 にアップロード所要時間およびスループットの結果を示す．なお，サービスによっては，データの一部分のアップロードにとどめたものもある．

高エネルギー物理学のデータのアップロードと比較してスループットが低いが，これは，平均オブジェクト長が，高エネルギー物理学の場合（648MiB）よりもかなり小さい（44.4MiB）ことに起因すると考えられる．

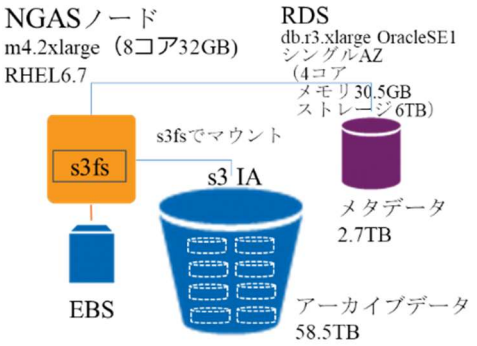

図 2　AWS 上に移植した NGAS の構成

なお，Azure に関しては，当初，実験開始時点で提供されていた Azure CLI 2.0 プレビュー版を並列度のデフォルト値である 2 コネクションで動作させたところ，あまり性能が上がらなかった．その後提供された正式版を使って最大コネクション数を 16 に増すチューニングを実施した結果，4 倍近い性能改善が得られ，上記の結果を得た．

(2) **アプリケーションによるデータの読込み**

観測／解析データのアップロードは 4 種のクラウドに対して実施したが，このデータを管理しアクセスするアプリケーションである NGAS（Next Generation Archive System）は AWS の上で動作させた．

NGAS は，複数のホストとそのホスト配下のアーカイブデータを格納した複数のディレクトリ群，およびメタデータ管理用の Oracle データベースによって構成される．本実験では，2 個のホストを同じ AWS EC2 のインスタンス上に配置し，アーカイブデータの各ディレクトリに対応したバケットを作成して，これにデータをアップロードした後に，表 10 で述べた s3fs を使用してファイルシステムとしてマウントした．また，データベースは AWS のサービスとして提供される RDS for Oracle Database を使用した．構成を図 2 に示す．

NGAS ソフトウェアは，通常のオンプレミスの場合とほぼ同じ方法でインストールすることができたが，クラウド特有の問題として，RDS のホスト名がオンプレミスのシステムで通常使われるホスト名より長い（53 文字）ために，データベースの対応するカラム長を拡大する必要があった．インストール後，アーカイブデータの単純な検索は正常に動作することが確認できた．今後，実際の研究シーンにおけるアーカイブデータの参照パターンに基づいて，アーカイブからデータを取り出す際の性能や，1 回の取出しに要する課金額を測定する予定である．

5. まとめと今後の取組み

本報告では，複数の商用パブリッククラウドで提供されるコールドストレージについての基礎的なベンチマークおよび実際の研究データによる実験結果を述べた．

コールドストレージに関しては，カタログから判断できる仕様はもとより，実際の使用を通じて把握できる性能や運用性に関してもクラウドごとにかなり特徴があり，このような情報の収集と整備を通じて，目的に適合したコールドストレージを的確に選択できるようにしてゆくことが改めて重要と考える．また，今回は，まずはクラウドが提供する機能をデフォルトのまま使ってみるという方針で実験を進めたが，本格的な利用においては相応のチューニングが必要であり，特に大量の研究データを格納して利用する場合には，その成否が性能や課金に大きく影響することが明らかとなった．

今後の取組みとしては，これまでの実験の延長として，より最適なアクセス・管理方法を見出して性能や運用性の改善を図ること，研究データの利用シーンにおける性能や課金という観点を強化した評価を継続することに加えて，以下のような展開が考えられる．

- SINET 接続サービスを利用したデータ転送の高速化の可能性を検討する．SINET 接続サービスは，クラウドプロバイダのデータセンタを SINET に直結するものであり，今回の実験の対象としたクラウドでは，AWS および Azure が対応している（2017 年 6 月現在）．特にシナリオ 3 のような実際の研究データの転送やリモートアクセスの局面で効果が得られるユースケースを探る．
- コールドストレージの特性に対するアプリケーションや運用の最適化を検討する．今回の実験では，研究データのファイルをそのまま 1 オブジェクトとして格納し，アプリケーションは無修正という前提でファイルシステム互換を実現する OSS を利用した．これに対して，ファイル API からオブジェクトストレージ API への変更による高性能・高機能化や，1 回のアクセス単位をまとめてコールドストレージに格納し復元処理の負担を軽減するといった可能性を検討する．
- クラウドコールドストレージ活用の支援情報やツールの蓄積・整備を進める．実験結果の事例化と共有，オンデマンドクラウド構築サービステンプレート[15]によるコールドストレージ利用環境構築（オブジェクトのファイルアクセス機能や操作の容易なデータ復元機能などを含む）の自動化などが考えられる．

今回の実験は，コールドストレージサービスを実際に利用することによって，研究データの長期保管におけるコールドストレージ活用の検討材料となる情報を得ることを目標として実施した．コールドストレージに限らず，クラウドサービスの機能・性能・品質の進化は非常に速い．実験期間中においても，大幅な性能や品質の向上によって，以前の結果が塗り替えられるということをたびたび経験している．従って，ここで述べた結果は，あくまでも今回の実験期間におけるスナップショットに近いものであると考えており，定点観測としてのデータの蓄積に関しても継続的

な取組みが必要であると認識している．このような情報整備の延長上にある取組みとして，研究データの長期保管にコールドストレージを活用してゆくためのベストプラクティスを積み上げ共有してゆくことにつなげてゆきたい．

謝辞 実験を進めるにあたりご指導・ご助言をいただいた各研究機関の皆様，結果の分析やチューニング方法の検討に関してご協力いただいた各プロバイダの皆様，実験に必要なクラウドサービスの調達をご支援いただいた国立情報学研究所の関係者の皆様に，謹んで感謝の意を表する．

参考文献

[1] “ストレージネットワーキング用語集：コールドストレージ”. http://www.snia-.org/dictionary/storage_network_keywords/234.html, (参照 2017-06-19).
[2] G. Juve, E. et al.. Scientific workflow applications on Amazon EC2. Proceedings of the 2009 5th IEEE International Conference on E-Science Workshops. 2009, p. 59-66.
[3] Jackson, K. et al.. Performance analysis of high performance computing applications on the Amazon Web Services Cloud. Proceedings of the 2010 IEEE Second International Conference on Cloud Computing Technology and Science (Cloud- Com 2012). 2010, p. 159-168.
[4] Wang, G. and Ng, T.S.E.. The impact of virtualization on network performance of Amazon EC2 data center. Proceedings of the 29th conference on Information communications (INFOCOM’19). 2010, p. 1163-1171.
[5] Knight, D. et al.. Evaluating the efficacy of the cloud for cluster computation. Proceedings of the 2012 IEEE Aerospace Conference. 2012, p. 1-10.
[6] Marathe, A. et al.. A comparative study of high-performance computing on the cloud. Proceedings of the 22nd international symposium on High-performance parallel and distributed computing (HPDC ’13). 2013, p. 239-250.
[7] “Amazon Glacier Developer Guide API Version 2012-06-01”. http://docs.aws.amazon.com/amazonglacier/latest/dev/glacier-dg.pdf, (参照 2017-06-19).
[8] “Working with Amazon S3 Objects - Storage Classes”. http://docs.aws.amazon.com/AmazonS3/latest/dev/storage-class-intro.html, (参照 2017-06-19).
[9] “ストレージクラス”. https://cloud.google.com/storage/docs/storage-classes, (参照 2017-06-19).
[10] “Azure Blob Storage: ホットストレージ層とクールストレージ層”. https://docs.microsoft.com/ja-jp/azure/storage/storage-blob-storage-tiers, (参照 2017-06-19).
[11] “Oracle Storage Cloud Service”. http://docs.oracle.com/en/cloud/iaas/storage-cloud/index.html, (参照 2017-06-19).
[12] “Cosbench”. https://github.com/intel-cloud/cosbench, (参照 2017-06-22).
[13] “s3fs”. https://github.com/s3fs-fuse/s3fs-fuse, (参照 2017-06-27).
[14] “cloudfuse”. https://github.com/redbo/cloudfuse, (参照 2017-06-27).
[15] 竹房あつ子, et al.. インタークラウド環境構築システムの開発. 信学技報 CPSY. 2017, 7 (掲載予定).