



AWS
Black Belt
Online Seminar

【AWS Black Belt Online Seminar】
Amazon Simple Storage Service
(Amazon S3)

アマゾンウェブサービスジャパン株式会社
ソリューションアーキテクト 焼尾 徹
2017.04.19

自己紹介

名前： 焼尾 徹 (やきお とおる)

所属： アマゾンウェブサービスジャパン
ソリューションアーキテクト

ロール： インサイドSA

経歴： ハードウェア、ソフトウェアメーカー
ストレージ商材の啓蒙・訴求

好きなサービス： Amazon Wind Farm (サービスじゃないけど)



内容についての注意点

- 本資料では2017年4月18日時点のサービス内容および価格についてご説明しています。最新の情報はAWS公式ウェブサイト(<http://aws.amazon.com>)にてご確認ください。
- 資料作成には十分注意しておりますが、資料内の価格とAWS公式ウェブサイト記載の価格に相違があった場合、AWS公式ウェブサイトの価格を優先とさせていただきます。
- 価格は税抜表記となっております。日本居住者のお客様が東京リージョンを使用する場合、別途消費税をご請求させていただきます。
- AWS does not offer binding price quotes. AWS pricing is publicly available and is subject to change in accordance with the AWS Customer Agreement available at <http://aws.amazon.com/agreement/>. Any pricing information included in this document is provided only as an estimate of usage charges for AWS services based on certain information that you have provided. Monthly charges will be based on your actual use of AWS services, and may vary from the estimates provided.

Agenda

- Amazon S3の位置付け
- 概要
- 機能
- パフォーマンス最適化
- ユースケース
- 料金
- まとめ



Amazon S3 の位置付け

AWSストレージ関連サービス



Amazon EBS
(不揮発性)

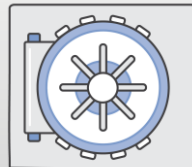


Amazon EC2
インスタンスストア
(揮発性)

ブロック



Amazon S3



Amazon Glacier

オブジェクト



Amazon EFS

ファイル

移行、取り込み、バッチ、ストリーム



AWS Snowball



AWS
Storage
Gateway



AWS Direct
Connect



3rd Party
Connectors



Transfer
Acceleration



Amazon Kinesis
Firehose

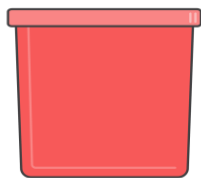
AWSストレージサービス



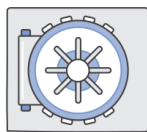
Amazon EBS
(不揮発性)



Amazon EC2
インスタンスストア
(エフェメラル)



Amazon S3



Amazon Glacier



Amazon EFS

ブロックストレージ

仮想サーバ(**Amazon EC2**)にマウントして活用
するディスクサービス

オブジェクトストレージ

安価かつ高い耐久性を持つ**オンライン**
ストレージサービス

安価かつ高い耐久性を持つ**アーカイブ**ストレージ
サービス

ファイルストレージ

複数の仮想サーバ(**Amazon EC2**)から同時マウント
できる**共有**ストレージサービス

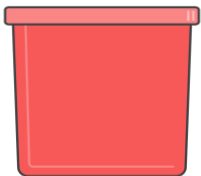
AWSストレージサービス（本日の主題部分）



Amazon EBS
(不揮発性)



Amazon EC2
インスタンスストア
(エフェメラル)



Amazon S3



Amazon Glacier



Amazon EFS

ブロックストレージ

仮想サーバ(Amazon EC2)にマウントして活用
するディスクサービス

オブジェクトストレージ

安価かつ高い耐久性を持つ**オンライン**
ストレージサービス

安価かつ高い耐久性を持つ**アーカイブ**ストレージ
サービス

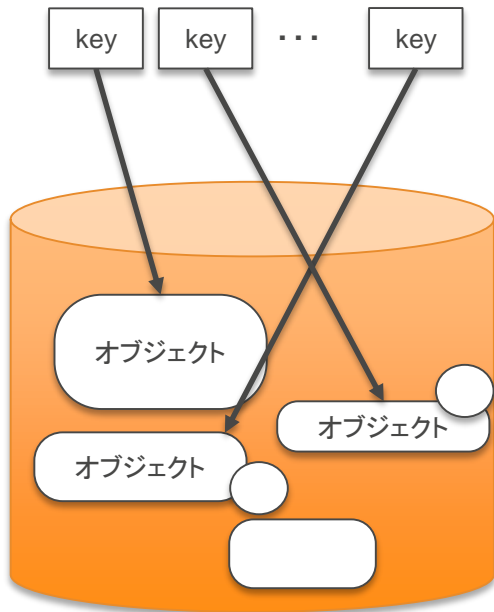
ファイルストレージ

複数の仮想サーバ(Amazon EC2)から同時マウント
できる**共有**ストレージサービス

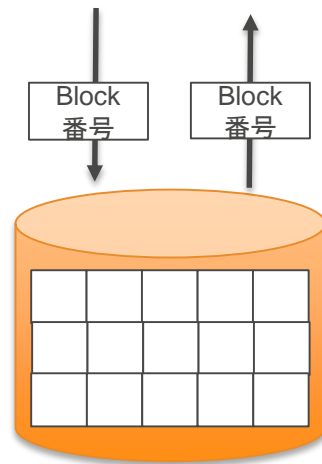
オブジェクトストレージとは

オブジェクトストレージ

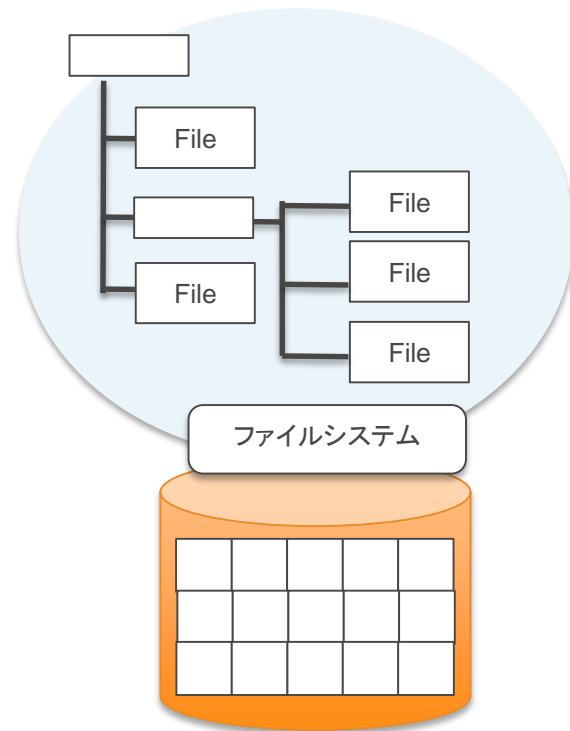
オブジェクト、それに付随するメタデータ、そのオブジェクトにアクセスするためのユニークなIDで構成されるデータの倉庫



ブロックストレージ



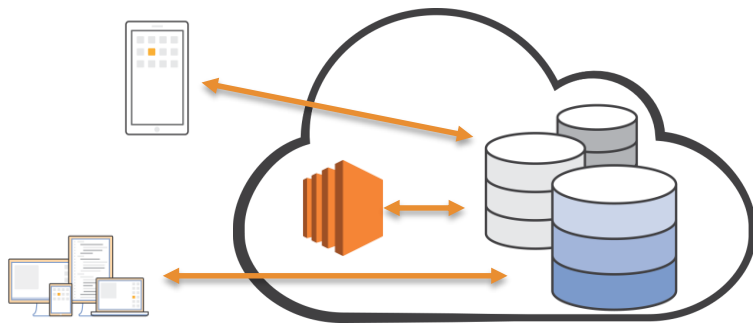
ファイルストレージ



概要

Amazon S3 概要

- Amazon Simple Storage Service (S3)はWeb時代のオブジェクトストレージ
- ユーザはデータを**安全に、どこからでも、容量制限なく**保存可能
- 最大限のスケーラビリティを利用者やデベロッパーに提供



Amazon S3 特徴

- **容量無制限**

- 1ファイル最大5TBまで

- **高い耐久性**

- 99.999999999%

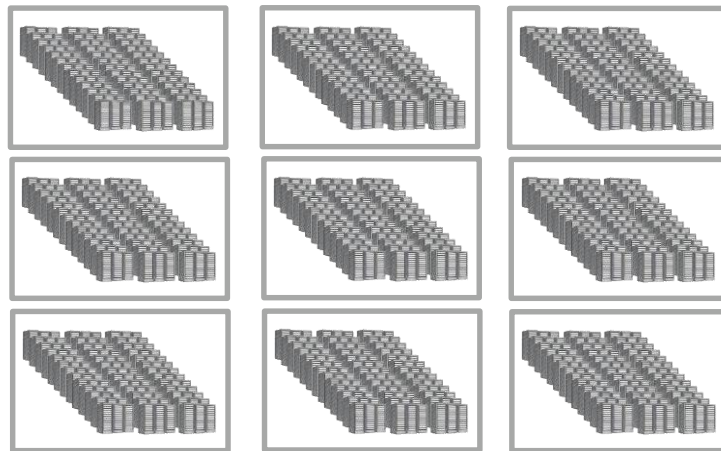
- **安価なストレージ**

- 容量単価:月額1GB / 約3円※

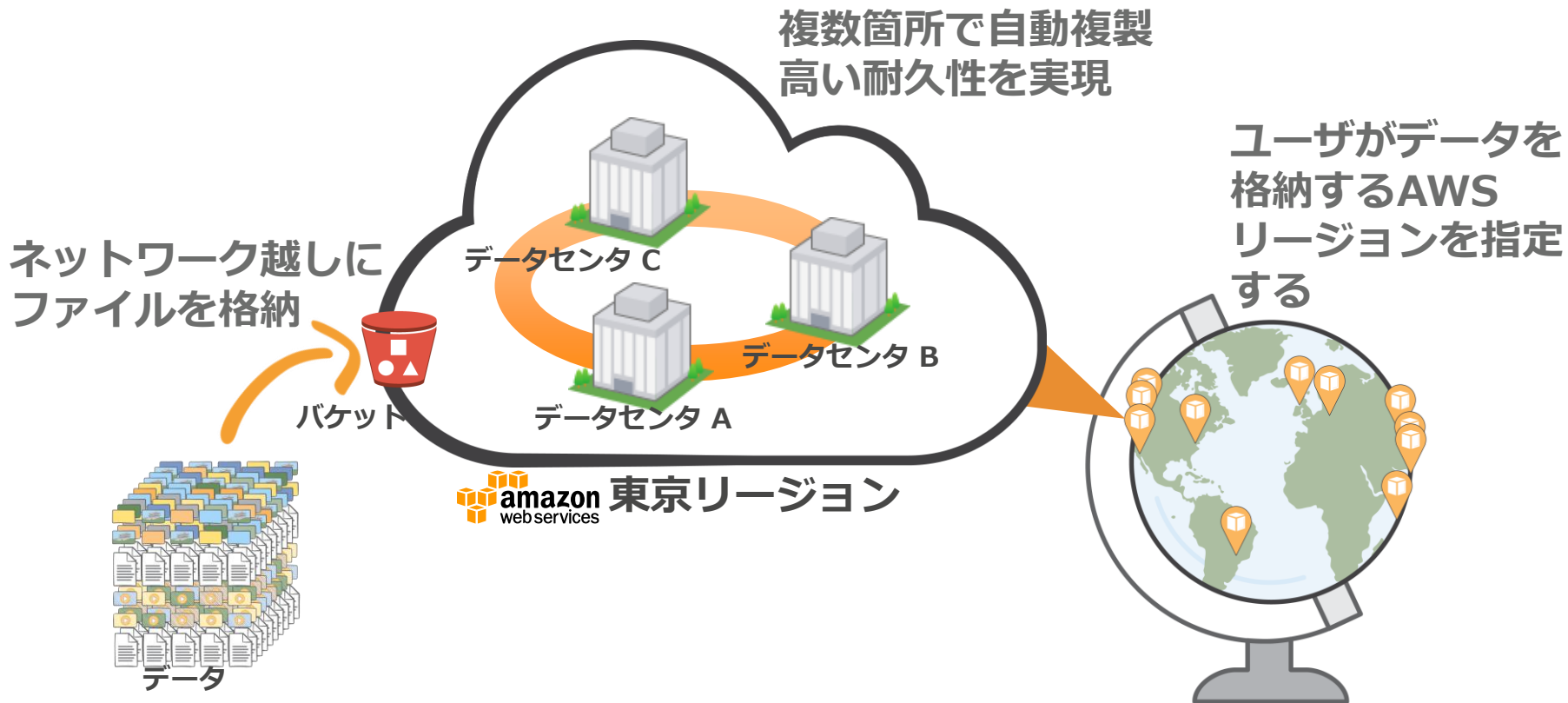
- **スケーラブルで安定した性能**

- データ容量に依存しない性能（ユーザが、サーバ台数、媒体本数やRAID、RAIDコントローラを考慮する必要がない）

AWS



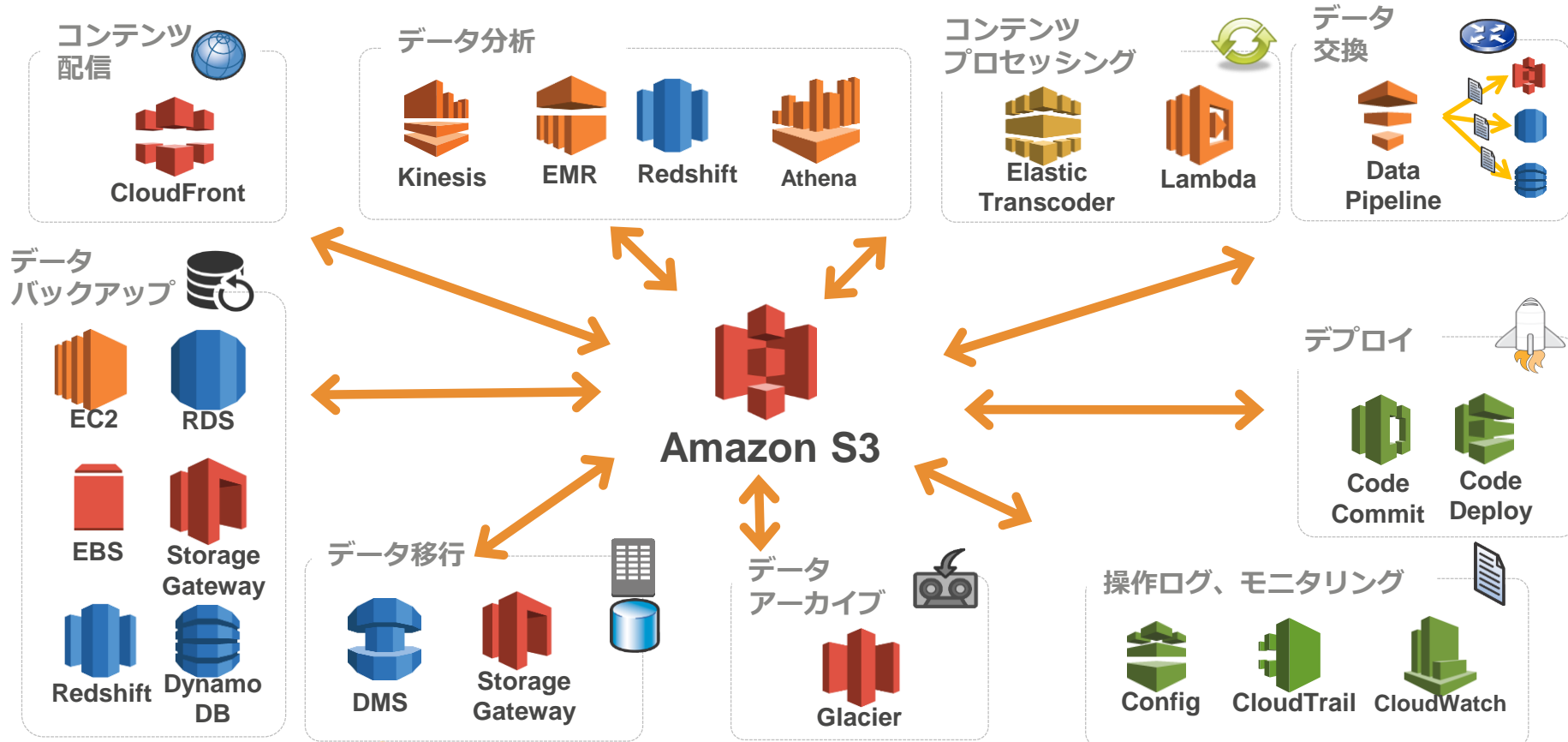
Amazon S3のデータ配置



Amazon S3の用途

- **コンテンツ配信や保管サーバ**
 - Web・画像・動画などのメディアコンテンツ
 - JavaScriptを活用した2Tier Webシステム
- **ログ&データハブストレージ**
 - ログや分析データ保管用ストレージ
 - データロード元
- **バックアップやディザスタリカバリ**
 - データバックアップストレージ
 - 拠点間レプリケーション

データレイクに代表されるS3によるデータ「ハブ」



Amazon S3 用語

• バケット

- オブジェクトの保存場所。各AWSアカウントにてデフォルト100個まで作成可能。**名前はグローバルでユニークな必要**あり。上限緩和申請で100以上も利用可能(*)に。

• オブジェクト

- データ本体。S3に格納されるファイルでURLが付与される。バケット内オブジェクト数は無制限。1オブジェクトサイズは0から5TBまで(1つのPUTでアップロード可能なオブジェクトの最大サイズは5GB)。

• キー

- オブジェクトの格納URLパス。「バケット+キー+バージョン」が必ず一意になる。

• メタデータ

- オブジェクトに付随する属性の情報。システム定義メタデータ、ユーザ定義メタデータがある。

• リージョン

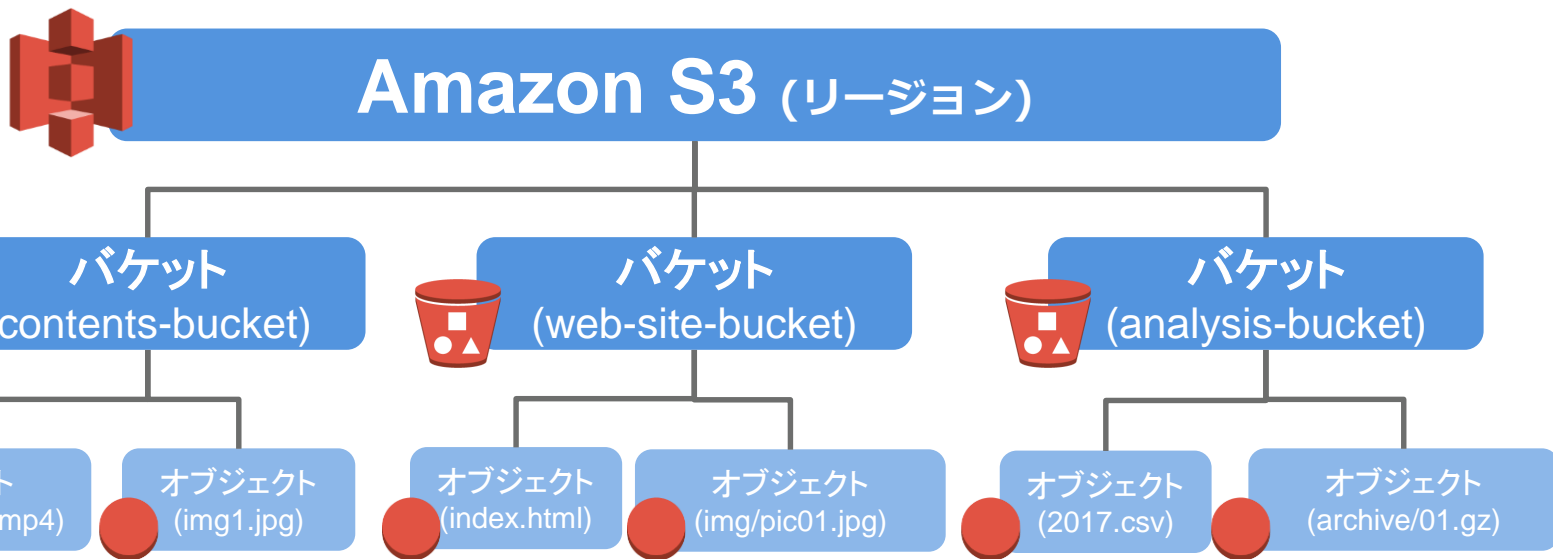
- バケットを配置するAWSのロケーション。目的のアプリケーションと同じリージョンであると有利。

• アクセスコントロールリスト(ACL)

- バケットやオブジェクトのアクセス管理。

(*) 2015.8月

Amazon S3 用語(続き)



- オブジェクトはバケット内にフラットに格納される。
- キーのパス指定でフォルダ階層のように表示も可能。「/」が区切り記号として、マネジメントコンソールでは、フォルダ構造を表現する。
- 例) video/sample.mp4 の「video/」の部分を**プレフィックス**という。

Amazon S3用語（続き）

・ ストレージクラス

- 用途に応じて、オブジェクトを格納するS3の場所の使い分け。

ストレージクラス	特徴	耐久性(設計上)
スタンダード	複数箇所にデータを複製。デフォルトのストレージクラス。	99.999999999%
STANDARD-IA (標準低頻度アクセスストレージ)	スタンダードに比べ格納コストが安価。いつでもアクセス可能だが、データの読み出し容量に対して課金。IAはInfrequent Accessの略。	99.999999999%
Glacier (アーカイブ)	最も低コストだが、データの取り出しにコストと時間を要する。ライフサイクルマネジメントにて指定する。	99.999999999%
低冗長化ストレージ(RRS)	RRS はReduced Redundancy Storageの略。Glacierから取り出したデータの置き場所として利用。	99.99%

Amazon S3用語(続き)

• S3のData Consistency モデル

- Amazon S3はデータを複数の場所に複製することで高い可用性を実現するため、データの更新・削除にはEventual Consistency Readモデル（結果整合性）が採用されている。

オペレーション	Consistencyモデル	挙動
新規登録 (New PUTs)	Consistency Read(*)	登録後、即時データが参照できる
更新 (Overwrite PUTs)	Eventual Consistency Read(結果整合性)	更新直後は、以前のデータが参照される可能性がある
削除 (DELETE)	Eventual Consistency Read (結果整合性)	削除直後は、削除前のデータが参照される可能性がある

- 同じオブジェクトへの複数同時書き込み制御のためのロック処理は行われず、タイムスタンプが更新される。
- （ロック処理があるような仕組みと比べて）読み込みの待ち時間が小さくなるのがメリット




(*) 2015.8月 new put!について、read-after-write consistencyがUS Standard regionでもサポートされるようになり、全てのリージョンにてread-after-write-consistencyとなりました。

Amazon S3 の操作

オペレーション	処理	特徴
GET	S3から任意のファイルをダウンロード	<ul style="list-style-type: none">• RANGE GETに対応。• Glacierにアーカイブされ、RestoreされていないオブジェクトへのGETリクエストはエラー
PUT	S3に対してファイルをアップロード(新規・更新)	<ul style="list-style-type: none">• シングルPUTオペレーションでは最大5GBまで、Multipart Uploadを利用すると5TBまで格納可能。
LIST	S3バケット内のオブジェクト一覧を取得	<ul style="list-style-type: none">• Prefixによるパス指定での取得一覧のフィルタリングが可能。• 1回のリクエストでは1,000オブジェクトまで情報を取得可能。それ以上の場合は再帰的にリクエストを実施する必要がある
COPY	S3内でオブジェクトの複製を作成	<ul style="list-style-type: none">• シングルCOPYオペレーションでは最大5GBまで、Multipart Uploadを利用すると5TBまでのファイルの複製が可能
DELETE	S3から任意のファイルを削除	<ul style="list-style-type: none">• シングルDELETEオペレーションで最大1,000個のオブジェクトを削除可能• MFA(Multi Factor Authentication)と連携した削除制御が可能
HEAD	オブジェクトのメタデータを取得	<ul style="list-style-type: none">• オブジェクトそのものをGETオペレーションで取得しなくても、メタデータだけを取得可能
RESTORE	アーカイブされたオブジェクトを一時的にS3に取り出し	<ul style="list-style-type: none">• Glacierからのデータの取り出し。• 低冗長化ストレージに指定期間オブジェクトがコピーされ、その指定期間中、ダウンロードが可能になる

Amazon S3の操作（続き）

全てのオペレーションがSDK, CLI, Management Consoleや3rd Party Toolで実行でき、用途に合わせて使い分け可能

操作		利用イメージ
アプリケーション連携	AWS SDK	<pre>PutObjectRequest putObjectRequest = new PutObjectRequest(bucketName, key, file); PutObjectResult result = this.client.putObject(putObjectRequest)</pre> 
コマンドラインやシェル	AWS CLI	<pre>\$ aws s3 cp xxx.mp4 s3://bucketName/ \$ aws s3api get-object --bucket-name <bucket-name> --key <prefix/file-name></pre>
手動、人間の操作	Management Console 3rdパーティツール	 <p>AWS Management Console</p>  <p>3rd Party Tools</p>

Amazon S3の操作(続き)

New

• S3の新コンソール

以前のコンソール

S3の新コンソール

お知らせ: オブジェクトギングと新しいストレージ管理機能が新しいコンソール使用可能です。
オブジェクトギングとストレージ管理を試みるためにオプトインをしてください。

お知らせ: S3 Transfer Accelerationによりデータのロード速度を最大 300% 引き上げ。
試してみる。

バケット名	リージョン	作成日
aws-toryakio-logs	アジアパシフィック (東京)	2016/12/19 16:14:20
cf-templates-3t9f4yn57gef-ap-northeast-1	アジアパシフィック (東京)	2017/01/05 14:22:17
cf-templates-3t9f4yn57gef-ap-northeast-2	アジアパシフィック (ソウル)	2017/02/15 8:55:58
cloudtrail-awslogs- engard-do-not-delete	米国東部 (バージニア北部)	2016/12/02 17:22:25
connect-3e10c554904e	米国東部 (バージニア北部)	2017/04/13 8:06:52
learnlambda-inputs3bucketfortransactionfiles-67df828u8549	アジアパシフィック (東京)	2017/01/05 14:24:35
my-bucket20161220	アジアパシフィック (東京)	2016/12/21 1:19:26
redshift-bucket-toryakio	米国西部 (オレゴン)	2017/02/10 22:43:00
sample-bucket-analytics-oregon	米国西部 (オレゴン)	2017/02/14 10:03:17
sample-bucket-analytics-virginia	米国東部 (バージニア北部)	2017/02/14 10:03:50
sample-bucket-cf	アジアパシフィック (東京)	2017/01/28 14:43:10

機能

アクセス管理

きめ細やかなバケットもしくはオブジェクトへのアクセス権の設定

デフォルトでは、S3のバケットやオブジェクトなどは全てプライベートアクセス権限 (Owner:作成したAWSアカウント)のみに設定

IAMユーザ、クロスアカウントユーザ、匿名アクセスなどバケット/オブジェクト単位で指定可能

- **ユーザポリシー**
 - IAM Userに対して、S3やバケットへのアクセス権限を設定
 - 複数バケットやS3以外のものも含めて一元的にユーザ権限を指定する場合など
- **バケットポリシー**
 - S3バケット毎に、アクセス権限を指定
 - クロスアカウントでのS3バケットアクセス権を付与する場合など
- **ACL**
 - 各バケットおよびオブジェクトのアクセス権限を指定
 - バケット単位やオブジェクト単位で簡易的に権限を付与する場合など

アクセス管理（続き）

ユーザポリシーサンプル



```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:ListAllMyBuckets"
      ],
      "Resource": "arn:aws:s3:::*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:ListBucket", "s3:GetBucketLocation"
      ],
      "Resource": "arn:aws:s3:::examplebucket"
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject", "s3:GetObject", "s3>DeleteObject"
      ],
      "Resource": "arn:aws:s3:::examplebucket/*"
    }
  ]
}
```

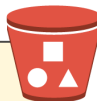
- ユーザポリシーを利用して、IAM ユーザに対して任意のバケットへのアクセス権限を付与
- Condition要素を利用することで、接続元IPアドレス制限なども指定することが可能

主に、IAMへのアクセス管理を軸とする場合

アクセス管理(続き)

バケットポリシーサンプル

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AddPerm",
      "Effect": "Allow",
      "Principal": "*",
      "Action": ["s3:GetObject"],
      "Resource": ["arn:aws:s3:::examplebucket/*"]
    }
  ]
}
```



- バケットポリシーを利用して、全てのユーザに対して、任意のバケットへのGETリクエストを許可

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "IPAllow",
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:*",
      "Resource": "arn:aws:s3:::examplebucket/*",
      "Condition": {
        "IpAddress": {"aws:SourceIp": "54.240.143.0/24"}
      }
    }
  ]
}
```



- バケットポリシーを利用して、任意のIPアドレスレンジからバケットへのアクセスを許可
- Conditionを利用してIAM User、クロスアカウント、IPアドレス制限、HTTP Referrer制限、CloudFront、MFA制限なども指定可能

主に、そのバケットへそのものへのアクセス管理を軸とする場合

その他サンプルは下記URLを参照

http://docs.aws.amazon.com/ja_jp/AmazonS3/latest/dev/example-bucket-policies.html

アクセス管理（続き）

ACLはバケット単位のACLとオブジェクト単位のACLが存在

- バケットACLはバケット内のオブジェクトにも影響を与えるが、オブジェクトが個別にACLを設定している場合、オブジェクトACLが優先させる
- ACLよりも、ユーザポリシーやバケットポリシーが優先される
- ACLはバケットやオブジェクトに対して100個まで指定可能
 - Grantee : Everyone, Authenticated Users, Log Delivery, Me
 - Permission: READ, WRITE, READ_ACP, WRITE_ACP, FULL_CONTROLL
- 例えば、違うアカウントが所有するバケット上のオブジェクトのアクセス許可を管理する場合に、ACLが有用



主に、そのオブジェクトだけのアクセス管理を考える場合
(バケットACLは、アクセスログの取得への許可で使用)

Pre-signed Object URL (署名付きURL)

Pre-signed URLを利用することで、セキュアにS3とのデータのやり取りが可能

- AWS SDKを利用して生成される署名されたURLを利用し、S3上のプライベートなオブジェクトに対して**一定時間**アクセスを許可
- GETとPUTオペレーションで利用可能
 - 任意のユーザへの一時的なオブジェクト共有
 - 任意のユーザからの一時的なS3へのオブジェクトアップロード権限の付与



Pre-signed Object URL (署名付きURL、続き)

署名URLの生成ソースサンプル (Java SDK)

```
AmazonS3 s3Client = new AmazonS3Client(new ProfileCredentialsProvider());

java.util.Date expiration = new java.util.Date();
long msec = expiration.getTime();
msec += 1000 * 60 * 60; // 1 hour.
expiration.setTime(msec);

GeneratePresignedUrlRequest generatePresignedUrlRequest =
    new GeneratePresignedUrlRequest(bucketName, objectKey);

// GETの場合
generatePresignedUrlRequest.setMethod(HttpMethod.GET);

//PUTの場合
generatePresignedUrlRequest.setMethod(HttpMethod.PUT);

generatePresignedUrlRequest.setExpiration(expiration);
URL url = s3client.generatePresignedUrl(generatePresignedUrlRequest);

// 以降でPUTもしくはGET処理を実装
:
```

URL有効期間の指定

対象バケットおよびオブジェクトの指定

GET/PUTのいずれかの処理を指定

署名付きURL生成



Pre-signed Object URL (署名付きURL、続き)

署名URLの生成ソースサンプル (Python)

```
# Get the service client.
s3 = boto3.client('s3')

# Generate the URL to get 'key-name' from 'bucket-name'
url = s3.generate_presigned_url(
    ClientMethod='get_object',
    Params={
        'Bucket': 'sample-bucket-cf',
        'Key': 'contents/test.txt'
    },
    ExpiresIn=3600
)
```

```
# 以降でPUTもしくはGET処理を実装
:
```

署名付きURL生成

GET/PUTのいずれかの処理
を指定

対象バケットおよびオブジェ
クトの指定

URL有効期間の指定



Webサイトホスティング機能

静的なWebサイトをS3のみでホスティング可能

- バケット単位で指定
 - Management Consoleで設定可能
 - パブリックアクセスを許可するため別途バケットポリシーで全ユーザにGET権限を付与
- 独自ドメインの指定
 - ドメイン名をバケット名として指定(www.example.com)
 - 通常は `http://バケット名.s3-website-ap-northeast-1.amazon.com`
 - Route53のAlias設定でドメイン名とS3のバケット名を紐付けたレコードを登録
- リダイレクト機能
 - 任意のドメインにリダイレクト設定が可能
 - `x-amz-website-redirect-location`(メタデータの一つ)にセットされる

Webサイトホスティング機能(続き)

- CORS(Cross-origin Resource Sharing)の設定
 - AJAXなどを利用して、異なるドメインからのS3アクセス時に利用
 - Management Console の場合Bucket PropertiesのPermissionより設定

```
<CORSConfiguration>
<CORSRule>
  <AllowedOrigin>http://www.example.com</AllowedOrigin>
  <AllowedMethod>PUT</AllowedMethod>
  <AllowedMethod>POST</AllowedMethod>
  <AllowedMethod>DELETE</AllowedMethod>
  <AllowedHeader>*</AllowedHeader>
</CORSRule>
</CORSConfiguration>
```

設定例

クロスドメインがwww.example.comの場合、
全てのリクエストを許可

- CloudFrontとの連携

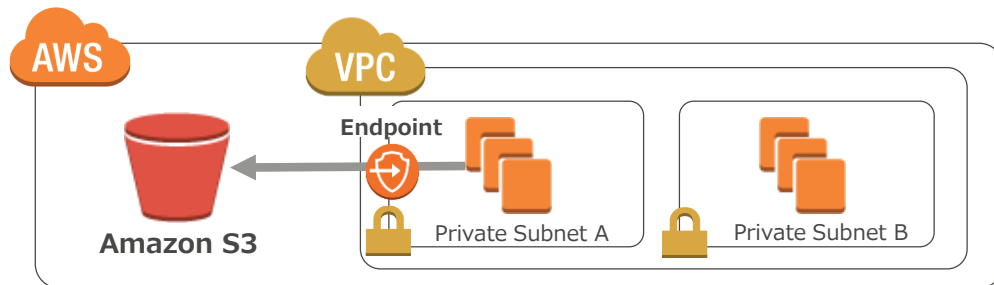
- WebサーバとしてS3を利用する場合は、CloudFront経由で配信することを推奨
- バケットポリシーを利用してCloudFrontからのHTTP/HTTPSリクエストのみを許可することも可能
 - バケットポリシーのPrincipalにCloudFrontのCanonicalUserを指定 (CloudFrontの「Origin Access Identity」のコンフィグレーション)

VPC Endpoint

(*) 2015.5月より

VPC内の**Private Subnet**上で稼働するサービスから、NAT GatewayやNATインスタンスを経由せずに直接S3とセキュアに通信させることが可能

- 通信可能なのは**同一リージョン**のS3のみ
- VPC管理画面のEndpointで作成し、S3と通信したいSubnetの**ルートテーブル**に追加
- Endpoint作成時にアクセスポリシーを定義し、通信可能なBucketや通信元のVPCの指定が可能 (バケットポリシーやIAMポリシーを利用したSource IPやVPC CIDRによる制限は利用不可)
- 別のVPCやSubnetを跨いだ直接のEndpointの利用は不可



S3 support for IPv6

New
2016.8月より

IPv4とIPv6の両方を、デュアルスタックエンドポイントにてサポート

- 追加の費用はなし
- アクセス先のエンドポイントを以下のように変更する
 - 仮装ホストスタイルの場合
<http://bucketname.s3.dualstack.aws-region.amazonaws.com>
 - パススタイルアドレスの場合
<http://s3.dualstack.aws-region.amazonaws.com/bucketname>
- AWS SDK/CLIでの利用
 - http://docs.aws.amazon.com/ja_jp/AmazonS3/latest/dev/dual-stack-endpoints.html
- 静的ウェブホスティングでは利用できない
- IPv6ベースでのアクセス制御が可能

S3 support for IPv6(続き)

- IPアドレスによるアクセス権限の例

IPv4を指定しているバケットポリシー例

```
{ "Version": "2012-10-17",  
  "Id": "S3PolicyId1",  
  "Statement": [  
    { "Sid": "IPAllow",  
      "Effect": "Allow",  
      "Principal": "*",  
      "Action": "s3:*",  
      "Resource": "arn:aws:s3:::examplebucket/*",  
      "Condition": {  
        "IpAddress": {"aws:SourceIp": "54.240.143.0/24"}  
        "NotIpAddress": {"aws:SourceIp": "54.240.143.188/32"} } } ] }
```

IPv6を加えたバケットポリシーの例

```
{ "Version": "2012-10-17",  
  "Id": "S3PolicyId1",  
  "Statement": [  
    { "Sid": "IPAllow",  
      "Effect": "Allow",  
      "Principal": "*",  
      "Action": "s3:*",  
      "Resource": "arn:aws:s3:::examplebucket/*",  
      "Condition": {  
        "IpAddress": "aws:SourceIp":  
          [ "54.240.143.0/24", "2001:DB8:1234:5678::/64" ]  
        "NotIpAddress": {"aws:SourceIp":  
          [ "54.240.143.128/30", "2001:DB8:1234:5678:ABCD::/80" ] } } ] }
```

暗号化によるデータ保護

保管時(Amazon S3 データセンター内のディスクに格納されているとき)のデータを暗号化して保護するもの

• サーバサイド暗号化

- AWSのサーバリソースを利用して格納データの暗号化処理を実施
- 暗号化種別
 - SSE-S3 : AWSが管理する鍵を利用して暗号化
 - SSE-KMS : Key Management Service(KMS)の鍵を利用して暗号化
 - SSE-C : ユーザが提供した鍵を利用して暗号化(AWSで鍵は管理しない)

• クライアントサイド暗号化

- 暗号化プロセスはユーザ管理
- クライアント側で暗号化したデータをS3にアップロード
- 暗号化種別
 - AWS KMSで管理されたカスタマーキーを利用して暗号化
 - クライアントが管理するマスターキーを利用して暗号化



暗号化

なし

AES-256
Amazon S3 では、サーバー側の暗号化でデータを暗号化します。

AWS-KMS

キャンセル 保存



異なるリージョン間のS3バケットオブジェクトのレプリケーションを実施

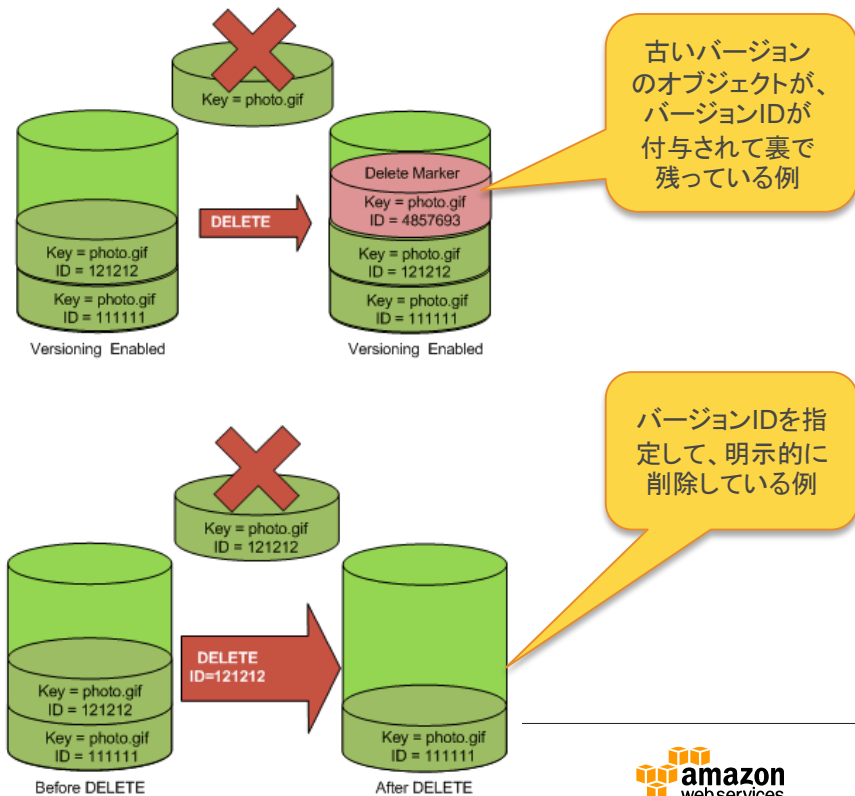
- バケットに対するオブジェクトの作成、更新、削除をトリガーに非同期でレプリケーションが実行
 - 対象元バケットはバージョンングの機能を有効にする必要がある
 - バケットはそれぞれ異なるリージョンでなければならない
 - 双方向レプリケーションも可能
 - レプリケーション時は、リージョン間データ転送費用が発生



バージョン管理機能

ユーザやアプリケーションの誤操作による削除対策に有効

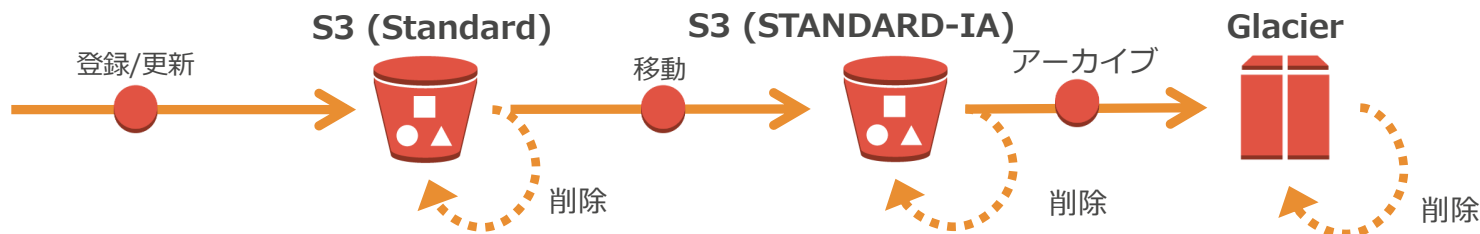
- バケットに対して設定
- バージョン保管されている任意のオブジェクトを参照可能
- バージョニングにより保管されているオブジェクト分も課金
- ライフサイクル管理(後述)と連携し、保存期間(有効期限)も指定可能
- バケットを削除したい場合は、古いバージョンのオブジェクトも削除する
 - ここでも、ライフサイクル管理が便利



ライフサイクル管理

バケット内のオブジェクトに対して、ストレージクラスの変更や、削除処理に関する自動化

- バケット全体もしくはPrefixに対して、オブジェクトの更新日をベースに日単位での指定が可能
- 最大1,000までLifecycleのルールを設定可能
- **毎日0:00UTCに処理がキューイングされ順次実行**
- Lifecycleを利用してIAに移動できるのは128KB以上のオブジェクトのみでそれ以外はIAに移動されない
- STANDARD-IA・アーカイブおよび削除の日程をそれぞれ指定した組み合わせも可能(後述)
- マルチアップロード処理で完了せず残った分割ファイルの削除にも対応
- MFA delete が有効なバケットにはライフサイクル設定は不可



ライフサイクル管理（続き）

- ライフサイクル管理の設定パターン

	選択可能オプション		処理
通常 (複数選択可能)	STANDARD-IAストレージクラスへの移行		128KB以上のオブジェクトを指定日(30日以上)にSTANDARD-IAに移動
	Glacier ストレージクラスへのアーカイブ		オブジェクトを指定日にGlacierへ移動 (STANDARD-IAを併用する場合は、STANDARD-IAにて30日以上経過している指定日を設定する必要あり)
	完全に削除(expire)		指定日にオブジェクトを削除(STANDARD-IAやGlacierと併用する場合は、それぞれの指定日より後に設定する必要あり)
バージョン有効 (複数選択可能)	最新のバージョン	STANDARD-IAストレージクラスへの移行	既存のオブジェクトを指定日にSTANDARD-IAへ移動 ※要件は通常と同様
		Glacier ストレージクラスへのアーカイブ	既存のオブジェクトを指定日にGlacierへ移動 ※要件は通常と同様
		失効	既存のオブジェクトを指定日に削除されバージョン
	以前のバージョン	STANDARD-IAストレージクラスへの移行	バージョンオブジェクトを指定日にSTANDARD-IAへ移動 ※要件は通常と同様
		Glacier ストレージクラスへのアーカイブ	バージョンオブジェクトを指定日にGlacierへ移動 ※要件は通常と同様
		完全に削除	バージョンオブジェクトを指定日に削除
マルチパート	不完全なマルチパートアップロードのアクション実行		マルチパートアップロードで完了せず残ったファイルを指定日に削除

ライフサイクル管理(続き)

Amazon S3 > my-bucket20161220

オブジェクト プロパティ アクセス権限 **管理**

ライフサイクル 分析 メトリクス インベントリ

+ ライフサイクルルールの追加 編集 削除 さらに ↓

ライフサイクルルール	適用対象:	移行先	以前のバージョンの移行
LifeCycle	バケット全体	標準 IA	Permanently Delete

開始もしくは
は編集方法

ライフサイクルルール

名前とスコープ ② 移行 ③ 有効期限 ④ 確認

移行の設定

現行バージョン 以前のバージョン

オブジェクトの移行

オブジェクト作成 対象: オブジェクト作成からの日数

+ 移行を追加する

標準 IA への移行の期限 30 x

標準 IA への移行の期限

Amazon Glacier への移行の期限

戻る 次へ

移行先ストレージ
クラスと、オブ
ジェクト作成から
の日数を指定

ライフサイクルルール

名前とスコープ 移行 ③ 有効期限 ④ 確認

設定の失効

現行バージョン 以前のバージョン

前のバージョンを完全に削除する ⓘ

次の: 10 日間の以前のバージョンになってからの日数

期限切れのオブジェクト削除マーカーおよび不完全なマルチパートアップロードをクリーンアップする。 ⓘ

期限切れのオブジェクト削除マーカーをクリーンアップする ⓘ

不完全なマルチパートアップロードをクリーンアップする ⓘ

戻る 次へ

以前のバージョン
を削除

アーカイブ及び復元

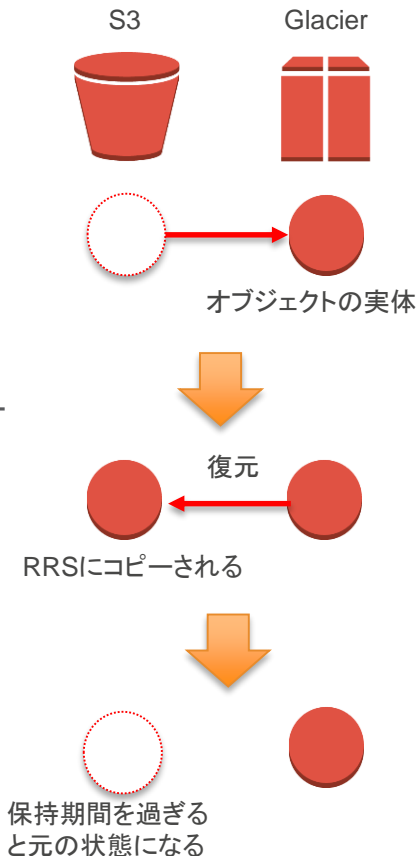
• アーカイブ

- オブジェクトのデータはGlacierに移動(アーカイブ後、マスターは Glacier)
- S3上のデータを削除することで、Glacier側のデータも削除される
- S3には8KBのオブジェクト名とメタデータのみが保管
- Glacierには32KBのインデックスおよび関連メタデータが追加で保管
- アーカイブしたオブジェクトを90日以内に削除しても、90日間アーカイブされたのと同じ課金対象

• オブジェクトの復元(restore)

- オブジェクト毎に復元
- データは一時的にS3の低冗長化ストレージに指定日数間複製される
- 復元後の、S3上での保持期間の変更も可能
- **復元にかかる時間について、3種類から選択可能に**
- 復元期間中は、S3の低冗長化ストレージとGlacier双方で課金

New



アーカイブ及び復元（続き）

New

2016.11月

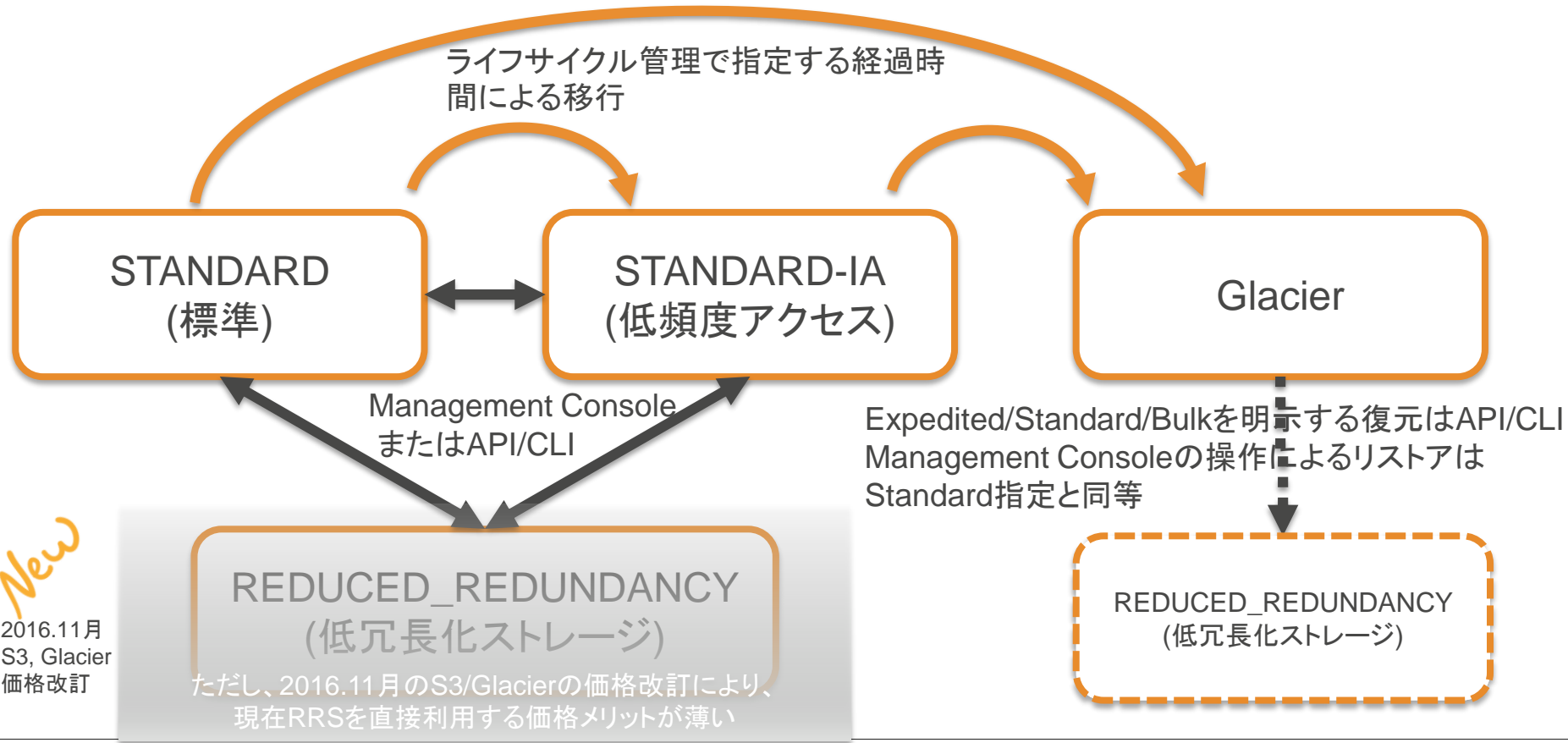
復元リクエスト時に指定できる3つの選択肢

- **Expedited:** 少ない数のファイルについて、緊急のアクセスを要する場合の取得
- **Standard:** 3-5時間の間にファイルを取得する標準的な取得
- **Bulk:** 5-12時間の間にファイルを取得する最も低価格で、大量のデータを取得

	迅速(Expedited)	標準(Standard)	大容量(Bulk)
データアクセス時間	1~5分	3~5時間	5~12時間
データ復元容量	\$0.033 / GB	\$0.011 / GB	\$0.00275 / GB
復元リクエスト	オンデマンド: \$0.011 リクエストごと プロビジョンド: \$110プロビジョンドキャパシ ティユニットごと	\$0.0571 : 1,000 リクエストあたり	\$0.0275 : 1,000 リクエストあたり

プロビジョンド=あらかじめデータを取り出すリソースを購入できる考え方
1プロビジョンドキャパシティユニット=5分間に、3回までのExpedited
復元リクエスト、かつ、復元時スループットが150MB/sec以内

ストレージクラス間のオブジェクト移動の整理



S3 分析

New

2016.11月

「STANDARD-IAとGlacierどちらににいつ移動すればいいのだろうか？」この疑問に応える、データのアクセスパターンの簡易可視化


開始方法

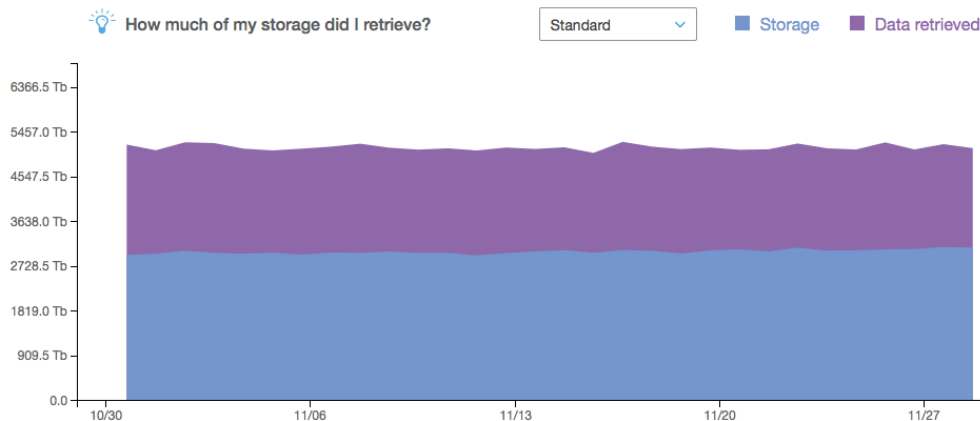
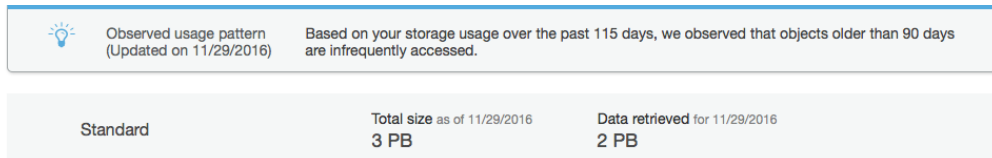
- 目的のバケットに対して、分析フィルターを定義する
- 結果がが出来るまで、フィルター作成してから24~48時間ほど待つ

CSVでも結果を出力する場合→

The screenshot displays the Amazon S3 console's 'Analysis' (分析) page. At the top, the 'Management' (管理) tab is selected. Below it, the 'Analysis' (分析) button is highlighted. The 'Filter Selection' (フィルターの追加) panel is open, showing a filter named 'whole' with a monitoring prefix/tag. The 'Data Export' (データのエクスポート) option is highlighted, indicating that CSV output is possible. The main area shows the 'Storage Class Analysis' (ストレージクラス分析) interface with a trash can icon and a bar chart.

S3 分析 (続き)

- バケットの分析を行う
 - バケット内のプレフィックスやタグで分類されたデータのみを分析 (フィルターで指定)
 - CSVでバケットに出力することも可能 (出力先にバケットポリシーを仕込む)
 - アクセス頻度の低いデータは、どのくらい経過したデータなのか、を把握
- 
- **ライフサイクルポリシーの設定値の元ネタとなる**



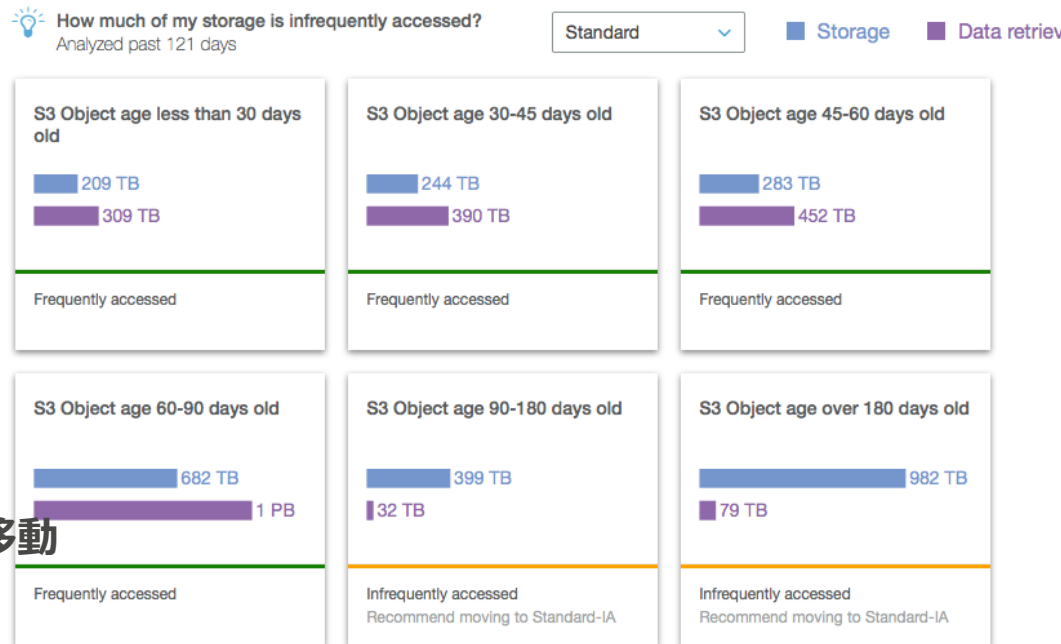
S3 分析 (続き)

青が格納量、紫がどれだけそのデータが読まれたか？

- この例の場合は、
 - 90日までのオブジェクトは、そこそこアクセスがある。
 - 90日以降のオブジェクトのニーズが急に減っている。
 - 90日以降でも、全くアクセスがないわけではない。
 - 他のコンプライアンス要件などを加味したとして、...



90日経過したデータをSTANDARD-IAへ移動
365日経過したデータをGlacierへ移動(例)
5年経過したデータは削除(例)



S3 インベントリ

New

S3に入っているオブジェクトのリストを、一気にcsvファイルで取得する

2016.11月


- オブジェクトのリストを取得するにあたって、List Bucketの処理に時間や手間がかかる場合に有益
- スケジュール化（日単位・週1回）してレポートを取得
- 初回の結果が出るまで、48時間待つ
- ある時点のsnapshotとしてのPUT/DELETE（結果整合性）結果のインベントリリストとなる



The screenshot shows the Amazon S3 console interface for a bucket named 'sample-bucket-cf'. The 'Management' tab is selected, and the 'Inventory' button is highlighted with a red box. Below the navigation bar, there is a message: 'インベントリレポートが有効になっていません。' (Inventory report is not enabled). Below this message is an icon of a bucket with a list symbol, and the text 'インベントリ' (Inventory). At the bottom, there is a note: 'バケット全体または共有プレフィックスについて、毎日または毎週、オブジェクトインベントリのCSVファイルを受け取ることができます。' (You can receive CSV files of object inventory for the entire bucket or shared prefixes, daily or weekly.) and a '+ 新規追加' (Add new) button.

S3 インベントリ(続き)

例) インベントリを取得したいバケット: sample-bucket-analytics-oregon

インベントリ名	フィルター	送信先バケット 	送信先プレフィックス	頻度
sample-inventory	プレフィックスでフィルター(省略)	redshift-bucket-toruyakio	s3inventory	週1回

マニフェストファイルの吐き出し先

`destination-prefix/source-bucket/config-ID/YYYY-MM-DDTHH-MMZ/manifest.json`
`destination-prefix/source-bucket/config-ID/YYYY-MM-DDTHH-MMZ/manifest.checksum`

この例の場合:

`s3://redshift-bucket-toruyakio/s3inventory/sample-bucket-analytics-oregon/sample-inventory/2017-02-14T15-02Z/manifest.json`
`s3://redshift-bucket-toruyakio/s3inventory/sample-bucket-analytics-oregon/sample-inventory/2017-02-14T15-02Z/manifest.checksum`

インベントリリストの吐き出し先

`destination-prefix/source-bucket/data/example-file-name.csv.gz`

この例の場合

`s3://redshift-bucket-toruyakio/s3inventory/sample-bucket-analytics-oregon/sample-inventory/data/0042fc70-0dee-4e0a-9fb5-92c639d1d93c.csv.gz`

Bucket	Key	VersionID	IsLatest	IsDeleteMaker	Size	Last modified date	Etag	Storage Class	Replication Status
sample-bucket-analytics-oregon	bad_keys/00000002/2017-	d2fPieFQm7	TRUE	FALSE	1024	2017-02-15T00:23:43.000Z	0f343b09311	STANDARD	COMPLETED
sample-bucket-analytics-oregon	bad_keys/00000002/2017-	fnS18W.tD4f	TRUE	FALSE	1024	2017-02-15T00:24:05.000Z	0f343b09311	STANDARD	COMPLETED
sample-bucket-analytics-oregon	bad_keys/00000002/2017-	sTM3kb7E5	TRUE	FALSE	1024	2017-02-15T00:24:15.000Z	0f343b09311	STANDARD	COMPLETED
sample-bucket-analytics-oregon	bad_keys/00000002/2017-	UZkrmkdrqZH	TRUE	FALSE	1024	2017-02-15T00:23:54.000Z	0f343b09311	STANDARD	COMPLETED
sample-bucket-analytics-oregon	bad_keys/00000002/2017-	kkxyNpUDpl	TRUE	FALSE	1024	2017-02-15T00:25:53.000Z	0f343b09311	STANDARD	COMPLETED
sample-bucket-analytics-oregon	bad_keys/00000002/2017-	Dhe37pgyHs	TRUE	FALSE	1024	2017-02-15T00:24:55.000Z	0f343b09311	STANDARD	COMPLETED
sample-bucket-analytics-oregon	bad_keys/00000002/2017-	Oo8UJgBwcJ	TRUE	FALSE	1024	2017-02-15T00:25:09.000Z	0f343b09311	STANDARD	COMPLETED
sample-bucket-analytics-oregon	bad_keys/00000002/2017-	ShIIN9n_agC	TRUE	FALSE	1024	2017-02-15T00:25:53.000Z	0f343b09311	STANDARD	COMPLETED
sample-bucket-analytics-oregon	bad_keys/00000002/2017-	zxO8Q9dLe_	TRUE	FALSE	1024	2017-02-15T00:25:11.000Z	0f343b09311	STANDARD	COMPLETED
sample-bucket-analytics-oregon	bad_keys/00000002/2017-	WaaQPOaJq	TRUE	FALSE	1024	2017-02-15T00:26:17.000Z	0f343b09311	STANDARD	COMPLETED

S3 イベント通知

バケットにてイベントが発生した際に、Amazon SNS, SQS, Lambdaに対して通知することでシームレスなシステム連携が可能

イベントタイプ	概要
s3:ObjectCreated:*	S3バケットにオブジェクト作成された時（PUT/POST/COPYのAPIがコールされた時）
s3:ObjectCreated:Put	
s3:ObjectCreated:Post	
s3:ObjectCreated:Copy	
s3:ObjectCreated:CompleteMultipartUpload	
s3:ObjectRemoved:*	S3バケットから、オブジェクトが削除された時 Delete = バージョニングされていないオブジェクトの削除、またはバージョンングされているバケットのオブジェクトの完全な削除 DeleteMarkerCreated = バージョニングされているオブジェクトの削除マーカー作成
s3:ObjectRemoved:Delete	
s3:ObjectRemoved:DeleteMarkerCreated	
s3:ReducedRedundancyLostObject	低冗長化ストレージにてデータロストが発生した時

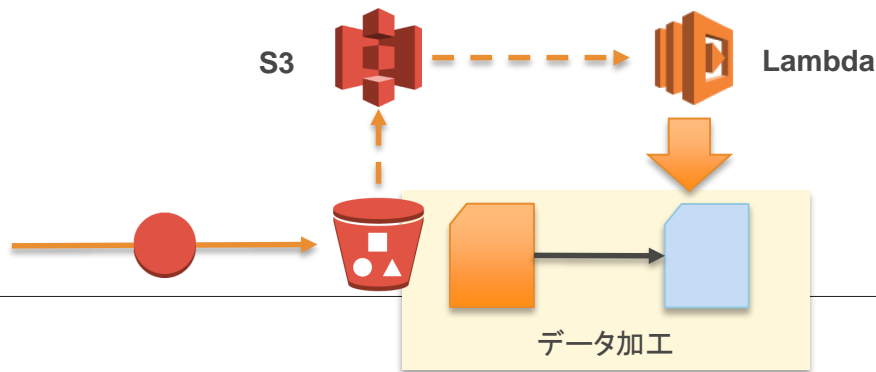
• 通知、または連携システム

- Amazon SNS: メール送信, HTTP POST, モバイルPushなどのTopics実行
- Amazon SQS: キューメッセージの登録
- Amazon Lambda: 指定Lambda Functionの実行

S3 イベント通知 (続き)

- S3からの実行権限の付与
 - SNSおよびSQSはそれぞれのPolicyに対してS3からの実行権限を付与
 - Lambdaに関しては、Lambdaが利用するIAM RoleにS3の権限を付与
- イベントでの通知内容

	通知項目
共通情報	リージョン, タイムスタンプ, Event Type
リクエスト情報	Request Actor Principal ID, Request Source IP, Request ID, Host ID
バケット情報	Notification Configuration Destination ID, バケット名, バケットARN, バケットOwner Principal ID
オブジェクト情報	オブジェクトキー, オブジェクトサイズ, オブジェクトETag, オブジェクトバージョンID



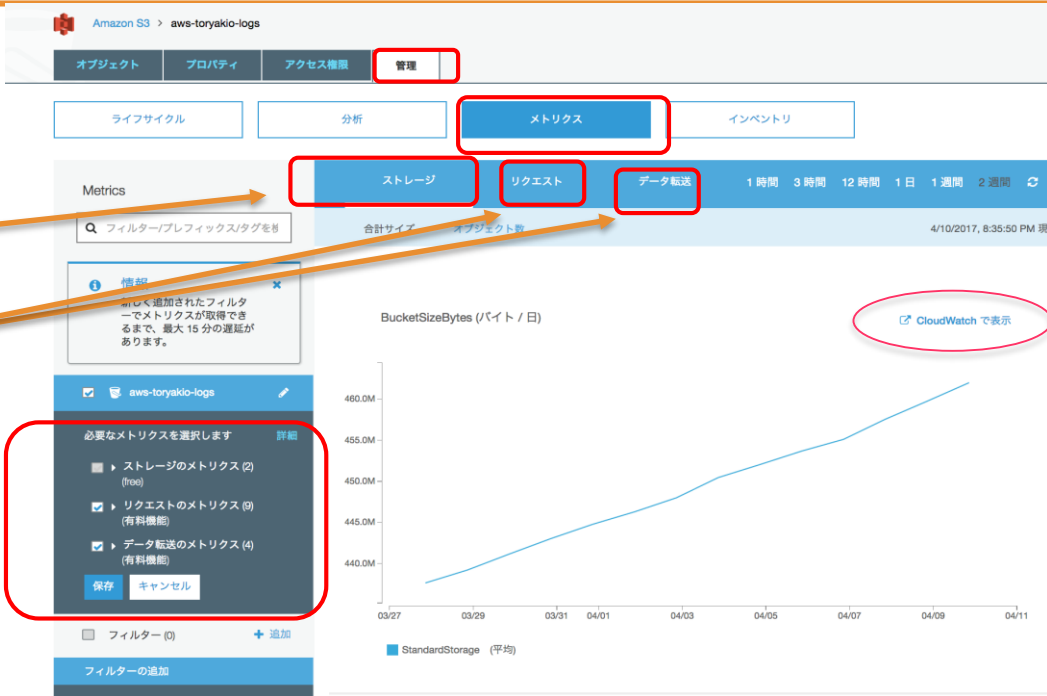
Amazon CloudWatchによるメトリクス管理

1. バケットに対するストレージメトリクス → 日単位
2. オブジェクトに対するリクエストメトリクス → 分単位

ストレージメトリクス

リクエストメトリクス

そのバケットで、リクエストメトリクスを利用する際に設定→



Amazon CloudWatchによるメトリクス管理(続き)

ストレージメトリクス

- バケット単位および、Storage Type(Glacierを除く) ごとにメトリクスを把握する
- 1日間隔でのレポート、状況把握 (追加料金なし)

メトリクス	詳細
BucketSizeBytes	標準ストレージクラス、STANDARD IAストレージクラス、または低冗長化ストレージ (RRS) クラスのバケットに保存されたバイト単位のデータ量
NumberOfObjects	GLACIER ストレージクラスを除く、すべてのストレージクラスのバケットに保存されたオブジェクトの総数

リクエストメトリクス

- タグやプレフィックスの指定にて細かい粒度での把握も可能
- 1分間隔でのメトリクスで、通常のCloudWatchの料金

New

2016.11月

メトリクス	単位	メトリクス	単位
AllRequests	Count	BytesDownloaded	MB
PutRequests	Count	BytesUploaded	MB
GetRequests	Count	4xxErrors	Count
ListRequests	Count	5xxErrors	Count
DeleteRequests	Count	FirstByteLatency	ms
HeadRequests	Count	TotalRequestLatency	ms
PostRequests	Count		

AWS CloudTrailによるAPI管理

CloudTrailを有効にすることでS3への操作ログ(API Call)を収集することが可能

- いつ、どこから、誰がS3の操作を行ったか、コンプライアンスの目的で把握可能(S3 イベント通知との使い分けを意識)

CloudTrailでのイベント	操作
データイベント	GetObject, DeleteObject, PutObjectなどのS3のオブジェクトに対するAPI操作
管理イベント	S3のバケット操作はもちろん、その他のすべてのAPI操作



New

2016.11月

- 監査対象とは別のS3バケットを用意することを推奨
- 100,000イベントごとに、\$0.1の料金
- ログに記録されるS3オペレーションは下記を参照
http://docs.aws.amazon.com/ja_jp/AmazonS3/latest/dev/cloud-trail-logging.html

その他のモニタリングや管理に有効な機能

• Logging

- バケット単位でバケットに対するアクセスログの出力設定が可能
- 出力先としてS3バケットを指定
- ログフォーマットは下記を参照

http://docs.aws.amazon.com/ja_jp/AmazonS3/latest/dev/LogFormat.html

• Tag管理

- バケットに対してタグの指定が可能
- タグ指定によりリソースグループにて関連するAWSサービスとの紐付けが可能
- **オブジェクトに対してのタグ**もできるようになり、ここまで紹介したライフサイクル、分析、モニタリング機能で活用可能


New

2016.11月

パフォーマンス最適化

パフォーマンスの最適化

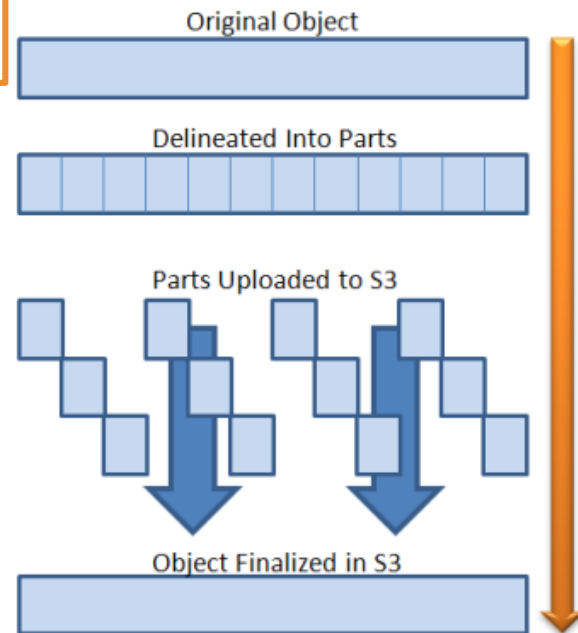
大きなサイズのファイルを快適に、ダウンロード、アップロード

- GETリクエストについて、**RANGE GETを活用**することで、マルチスレッド環境では高速にダウンロードが可能
 - マルチパートアップロード時と同じチャンクサイズを利用する
- 
- マルチパートアップロードの活用によるアップロード(PUT)オペレーションの高速化
 - チャンクサイズと並列コネクション数のバランスが重要
 - 帯域が太い場合は**20MB-50MB**チャンクサイズから調整
 - モバイルや帯域が細かい場合は**10MB程度**から調整

パフォーマンス最適化(続き)

目安100MB以上のファイルのアップロードを快適にしたい場合のマルチパートアップロード機能

- S3にアップロードする際に、ファイルを複数のチャンクに分割して並列アップロードを実施
 - ファイルが100MBを超える場合、利用することを推奨
 - 各チャンクは5GB以下に設定(5MB-5GB)
 - 全てのチャンクがアップロードされるとS3側で結合
 - Multipart Uploadを利用することで単一オブジェクトで5TBまで格納可能
- 各SDKにてMultipart Uploadの機能は実装済み
AWS CLIの場合は、ファイルサイズを元に自動的に判別
- PUT処理を並列化することでのスループット向上を期待→広帯域ネットワークが重要



パフォーマンスの最適化(続き)

1 秒あたり 100 個以上のリクエストを定常的に処理している場合

定常的にS3バケットへのPUT/LIST/DELETEリクエストが100RPSを超える、もしくはGETリクエストが300RPSを超える場合、キー名先頭部分の文字列をランダムにすることを推奨

例 (ハッシュ文字列の追加)

```
examplebucket/2013-26-05-15-00-00/cust1234234/photo1.jpg  
examplebucket/2013-26-05-15-00-00/cust3857422/photo2.jpg  
...  
examplebucket/2013-26-05-15-00-01/cust1248473/photo4.jpg  
examplebucket/2013-26-05-15-00-01/cust1248473/photo5.jpg  
...
```



```
examplebucket/232a-2013-26-05-15-00-00/cust1234234/photo1.jpg  
examplebucket/7b54-2013-26-05-15-00-00/cust3857422/photo2.jpg  
...  
examplebucket/9810-2013-26-05-15-00-01/cust1248473/photo4.jpg  
examplebucket/c34a-2013-26-05-15-00-01/cust1248473/photo5.jpg  
...
```

- 大量のGETリクエストが発生するワークロードの場合は、Amazon CloudFrontを併用することを推奨



Amazon S3 Transfer Acceleration

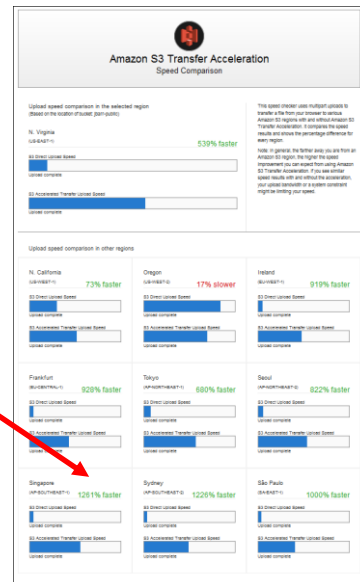
2016.4月より

AWSのマネージドバックボーンネットワークを活用した 高速ファイル転送サービス

- 全世界73箇所(*)にあるAWSのエッジネットワークから、最適化されたAWSのネットワークを經由して、高速にAmazon S3とのデータ転送を実現
 - 利用者は自動的に最短のエッジネットワークに誘導
- S3 Bucketに対してAccelerationを有効化
 - S3へのアクセスエンドポイントを変更するだけで利用可能
 - Acceleration有効後、転送速度が高速化されるまでに最大30分かかる場合がある
 - バケット名はピリオド"."が含まれない名前にする必要がある
 - IPv6 (dualstack)エンドポイントも指定可能
- 利用している端末からの無料スピード測定ツールも提供



マネジメントコンソール
からも起動可能



ユースケース

Amazon S3の用途（再掲）

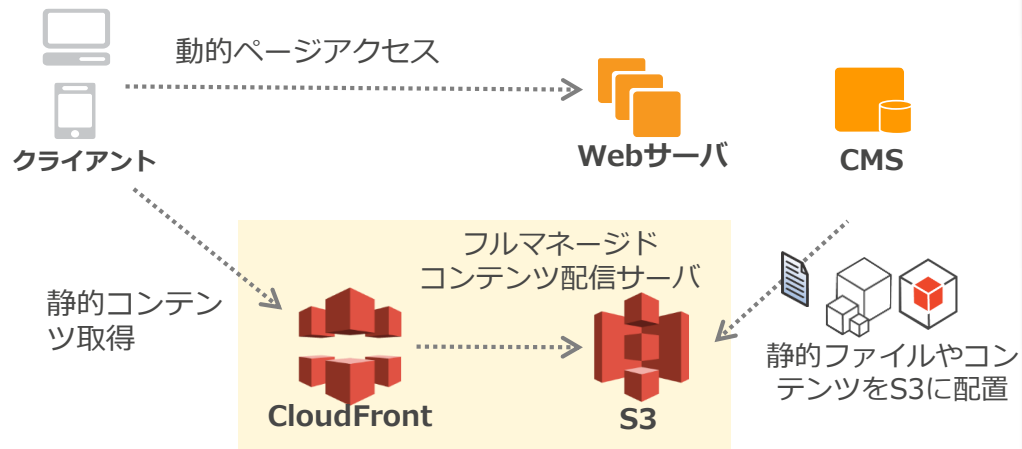
- **コンテンツ配信や保管サーバ**
 - Web・画像・動画などのメディアコンテンツ
 - JavaScriptを活用した2Tier Webシステム
- **ログ&データハブストレージ**
 - ログや分析データ保管用ストレージ
 - データロード元
- **バックアップやディザスタリカバリ**
 - データバックアップストレージ
 - 拠点間レプリケーション

コンテンツ配信サーバ

転送量が多くなりがちなデータをS3にオフロードし、CloudFrontでキャッシュさせる

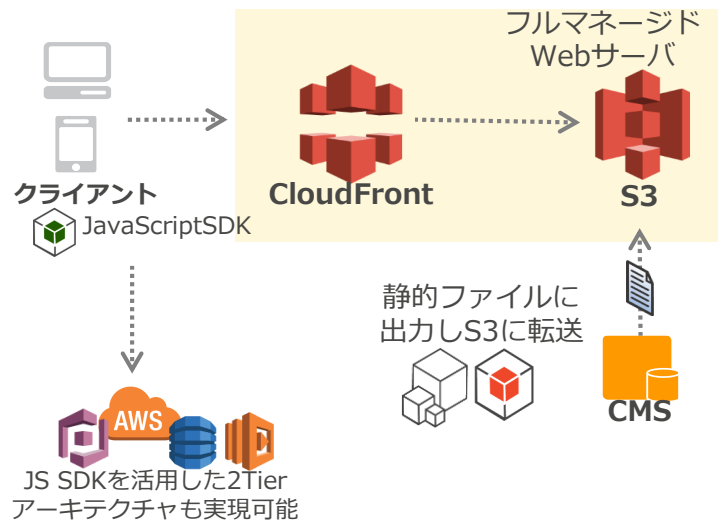
写真などのコンテンツを含む動的Webサイト

フルマネージドコンテンツ配信サーバとして配信負荷をオフロード
写真や動画などのファイルサイズの大きなものをS3に配置する



静的コンテンツ中心のサイト

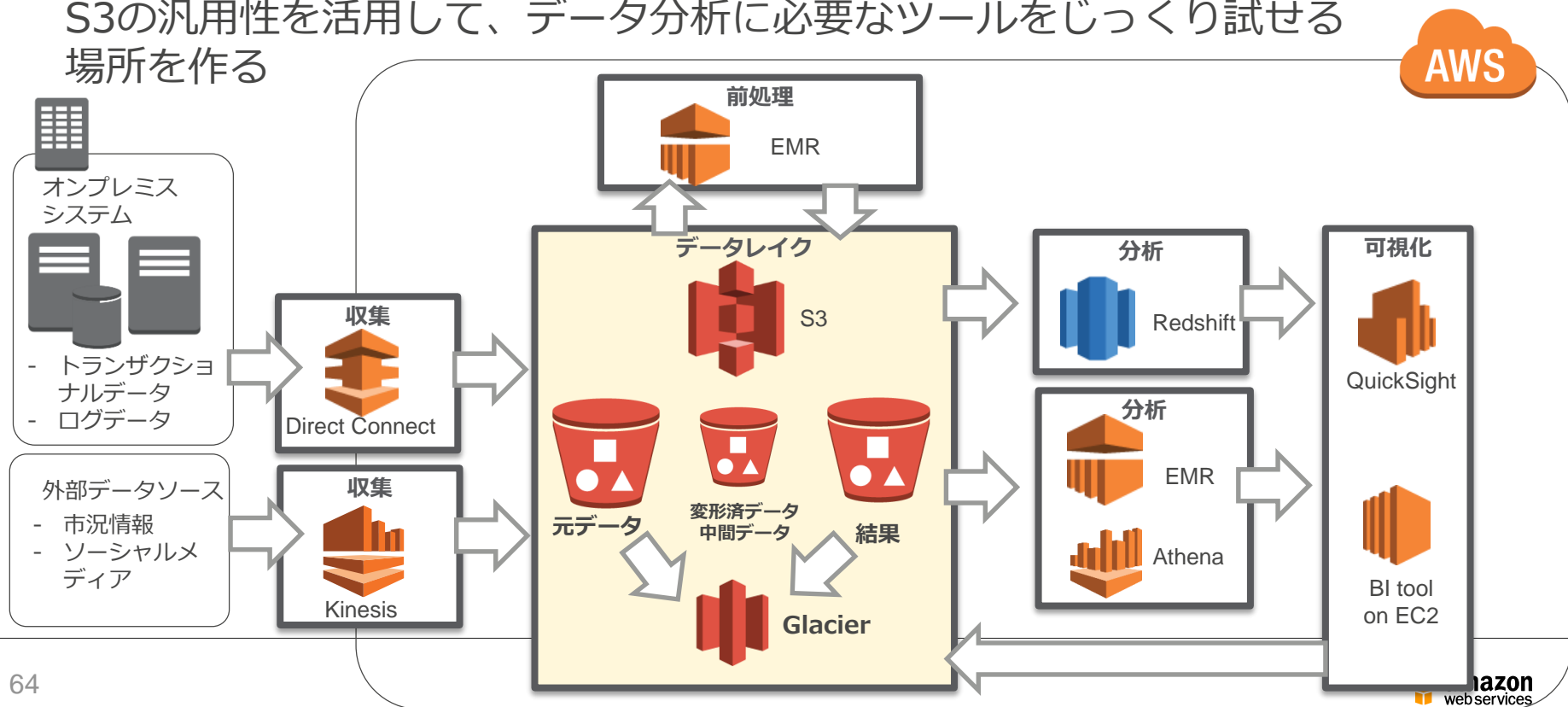
フルマネージドWebサーバとして活用
Java Scriptで動的部分がクライアントで処理されるシステム



ログ&データハブストレージ

データレイクを中心としたデータ処理基盤の例

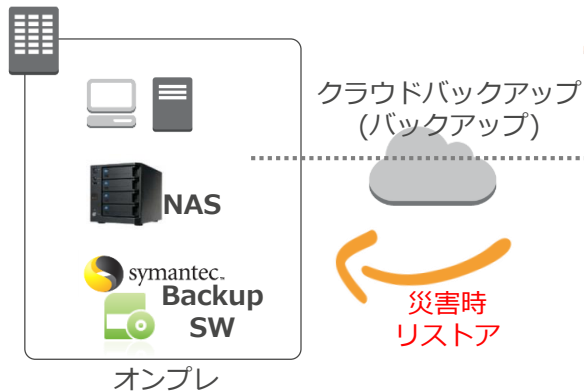
S3の汎用性を活用して、データ分析に必要なツールをじっくり試せる場所を作る



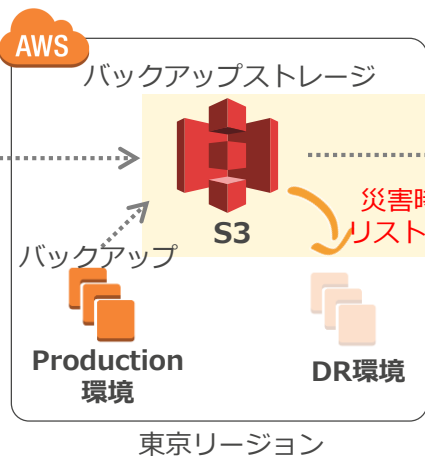
バックアップ/DRストレージ

用途に合わせてバックアップ/DR環境を安価に構築可能

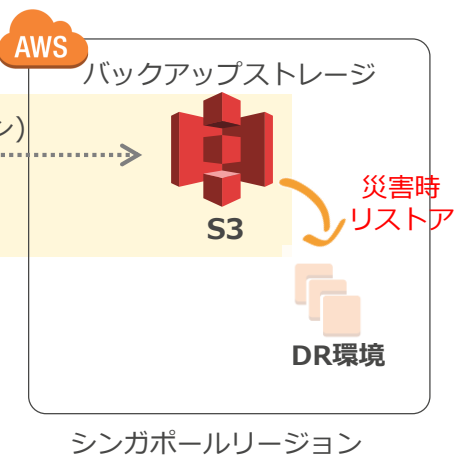
クラウドバックアップ用途



VPCでのデータ保護先



クロスリージョンでデータの複製を保持(DR等)



料金

Amazon S3の料金

ストレージ料金

New

	スタンダード	STANDARD-IA(*)	Glacier
最初の50TB/月	\$0.025 / GB	\$0.019 / GB	\$0.005 / GB
次の450TB/月	\$0.024 / GB	\$0.019 / GB	\$0.005 / GB
500TB月以上	\$0.023 / GB	\$0.019 / GB	\$0.005 / GB

(*) STANDARD-IAの請求対象となる最小オブジェクトサイズは 128 KB です。128 KB より小さいサイズのオブジェクトは、128 KBとして課金されます。

リクエスト料金

	スタンダード	STANDARD-IA	Glacier
PUT、COPY、POST、またはLIST リクエスト	\$0.0047 : 1,000 リクエストあたり	\$0.01 : 1,000 リクエストあたり	-
GET および他のすべてのリクエスト	\$0.0037 : 10,000 リクエストあたり	\$0.01 : 10,000 リクエストあたり	-
ライフサイクル移行リクエスト	-	\$0.01 : 1,000 リクエストあたり	\$0.0571 : 1,000 リクエストあたり
取り出し (容量)		\$0.01 GB あたり	Glacier取り出し料金(slide 45)

Amazon S3の料金

ストレージマネジメント料金

管理	料金
S3 Inventory	リストされるオブジェクト 100 万個ごとに \$0.0028
S3 Analytics Storage Class Analysis	モニターされるオブジェクト 100 万個ごとに月あたり \$ 0.10
S3 Object Tagging	10,000 タグごとに月あたりUS\$0.01
CloudWatch リクエストメトリクス	CloudWatch 料金
CloudTrail データイベント	100,000 件のイベントあたり \$0.1

New

データ転送料金

転送方向		価格
IN	全てのデータ転送「IN」	\$0.000/GB
OUT (AWS Network)	同じリージョンのAmazon EC2	\$0.000/GB
	別のAWSリージョン	\$0.090/GB
	Amazon CloudFront	\$0.000/GB
OUT (Internet)	最初の1GB/月	\$0.000/GB
	10TBまで/月	\$0.140/GB
	次の40TB/月	\$0.135/GB
	次の100TB/月	\$0.130/GB
	次の350TB/月	\$0.120/GB
	350TB/月以上	お問い合わせ

Amazon S3の料金

S3 Transfer Acceleration料金

転送方向		価格
S3へのデータIN	米国、欧州、日本のエッジロケーションによる高速化	\$0.04/GB
	その他の国のエッジロケーションによる高速化	\$0.08/GB
S3からのデータOUT (Internet)	エッジロケーションによる高速化	\$0.04/GB
S3と別のAWSリージョン間	エッジロケーションによる高速化	\$0.04/GB

S3 Transfer Accelerationの費用は、S3のデータ転送コストとは別に加算されることに注意

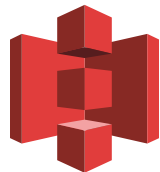
S3 Transfer Accelerationを利用してデータをやり取りする場合、通常のS3との転送よりも高速であるかを確認します。通常の転送に比べTransfer Accelerationが高速でないと判断した場合は、Transfer Accelerationの料金は請求されず、Transfer Accelerationシステムをバイパスする可能性があります。

S3 無料枠(1年)

- 標準ストレージ 5GB
- 20,000 GETリクエスト / 2,000 PUTリクエスト
- 15GBデータ送信

まとめ

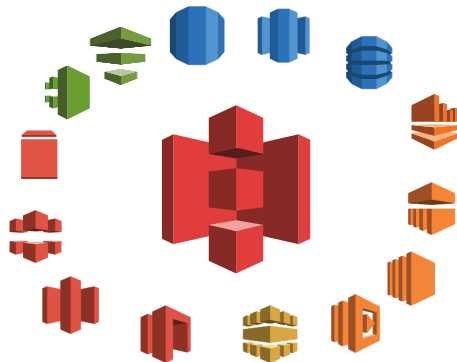
Amazon Simple Storage Service (S3)



マネージドオンラインストレージサービス



Amazon S3



- **特徴** (<http://aws.amazon.com/jp/s3/>)
 - 高い耐久性 99.999999999%
 - 格納容量無制限、利用した分のみの課金
 - 様々なAWSサービスと連携するセンターストレージ
- **価格体系** (<http://aws.amazon.com/jp/s3/pricing/>)
 - データ格納容量
 - データ転送量(OUT)
 - APIリクエスト数

まとめ

- **Amazon S3を活用することで**

- 想定が難しいストレージサイジングからの解放
- 堅牢性が高くセキュアにデータを保管
- コンテンツ配信などWeb負荷のオフロード
- お手元の「データ」の活用を促進することができる

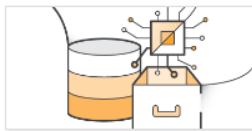
参考資料

- Amazon S3
 - <http://aws.amazon.com/jp/s3/>
- Amazon S3 開発者ガイド
 - http://docs.aws.amazon.com/ja_jp/AmazonS3/latest/dev/Welcome.html
- Amazon S3 FAQ
 - <http://aws.amazon.com/jp/s3/faqs/>
- Amazon S3 Pricing
 - <http://aws.amazon.com/jp/s3/pricing/>
- Amazon Web Services ブログ
 - <https://aws.amazon.com/jp/blogs/news/>

オンラインセミナー資料の配置場所

- AWS クラウドサービス活用資料集

- <http://aws.amazon.com/jp/aws-jp-introduction/>



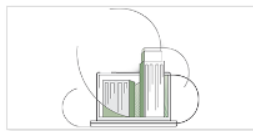
サービス別資料

無料オンラインセミナー「Black Belt Online Seminar」のサービスカット資料他、AWSのTechメンバーによる各サービスの解説資料がご覧いただけます。



ソリューション別資料

無料オンラインセミナー「Black Belt Online Seminar」のソリューションカット資料他、特定のソリューションについてのAWS活用方法がご覧いただけます。



業種別資料

無料オンラインセミナー「Black Belt Online Seminar」のインダストリーカット資料他、特定の業界のユースケースがご覧いただけます。



その他の資料

イベントに関する資料やアップデート情報などがご覧いただけます。

- AWS Solutions Architect ブログ

- 最新の情報、セミナー中のQ&A等が掲載されています

- <http://aws.typepad.com/sajp/>

AWSの導入、お問い合わせのご相談

AWSクラウド導入に関するご質問、お見積り、資料請求をご希望のお客様は以下のリンクよりお気軽にご相談ください

<https://aws.amazon.com/jp/contact-us/aws-sales/>

<p>お問い合わせ</p> <p>日本担当チームへのお問い合わせ ></p> <p>関連リンク</p> <p>フォーラム</p>	<h2>日本担当チームへのお問い合わせ</h2> <p>AWS クラウド導入に関するご質問、お見積り、資料請求をご希望のお客様は、以下のフォームよりお気軽にご相談ください。平日営業時間内に日本オフィス担当者よりご連絡させていただきます。</p> <p>※ご請求金額またはアカウントに関する質問はこちらからお問い合わせください。 ※Amazon.com または Kindle のサポートにお問い合わせはこちらからお問い合わせください。</p> <p>アスタリスク (*) は必須情報となります。</p> <p>姓* <input type="text"/></p> <p>名* <input type="text"/></p>
---	--

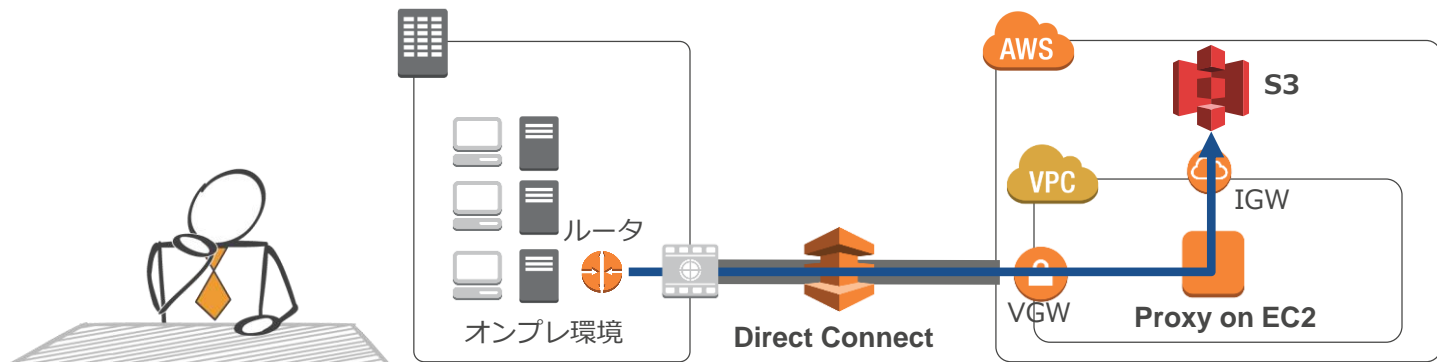
※「AWS お問い合わせ」で検索してください



**付録：前回のBlackBeltオンラインセミナーS3
のスライドで、今回取り上げなかったスライド**

AWS Direct Connect経由でのS3アクセス

- Direct Connectを利用してAWSに接続する場合、多くはPrivate ASを利用しVPCへの接続となるため、オンプレミス環境からS3への通信は、VPC内のEC2で構築したProxyサーバを経由するパターンが多い



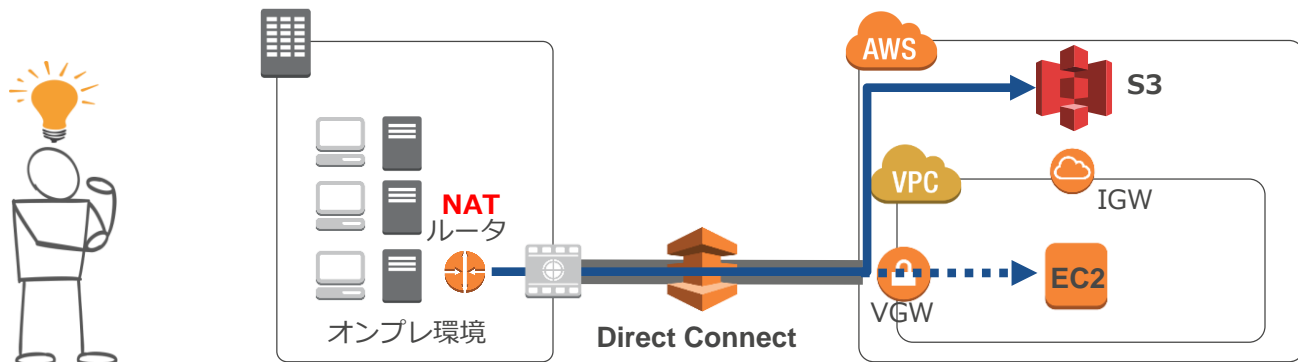
Amazon S3はAWSのPublicセグメントに存在するため

EC2の運用負荷と経由することでのオーバーヘッドによるDirect Connectの回線スピードの有効活用に課題

AWS Direct Connect経由でのS3アクセス

- AWSからパブリックIPの割り当てを受け、オンプレミスルータ側のIPアドレスでNATさせることで、VPCを経由せずオンプレミス環境からDirect Connect回線を利用して直接S3と通信が可能

(詳細は AWS SA Blog参照: <http://aws.typepad.com/sajp/2014/12/aws-direct-connect-public.html>)



Direct Connectの回線スピードをフル活用
VPC内のEC2インスタンス含むAWSリソースへのアクセスも可能