

Armazenamento S3

Modelos de Armazenamento (vamos voltar a isso mais adiante)

São muitos os modelos de armazenamento em nuvem, então vamos nos concentrar em alguns modelos principais, como os sistemas de arquivos gerais, os de armazenamento de objetos, os modelos de bancos de dados e sistemas de arquivos distribuídos para grandes volumes de dados.

Sistemas de arquivos

Você certamente está familiarizado com sistemas de arquivos de sistemas operacionais, como os do Windows ou Linux. Neles os dados são organizados em uma árvore de diretórios ou pastas. Este modelo é uma abstração de armazenamento de dados extremamente intuitiva e útil, e a API padrão para a versão derivada do Unix do sistema de arquivos é chamada de Interface de Sistema Operacional Portable (POSIX), sendo este o sistema que você usa no seu Windows, Linux ou mesmo Apple, podendo por linhas de comando, interfaces gráficas ou APIs, criar, ler e manipular arquivos e pastas. Esse modelo permite o acesso direto por programas na maior parte das linguagens (Python, Java, C++, R etc.) e fornece um mecanismo direto para representar relacionamentos hierárquicos dos dados (os diretórios). **Google Drive** e **DropBox** são exemplos de implementação desses sistemas em nuvem.

Armazenamento de objetos

Um armazenamento de objetos, ou **blobs** (*binary large objects*, como arquivos de imagens, som e multimídia), simplificam o modelo do sistema de arquivos eliminando a hierarquia dos objetos e proibindo atualizações dos objetos após terem sido criados. Diferentemente de arquivos texto como documentos, programas e páginas web, em geral você não atualiza uma imagem ou um vídeo, mas substitui o arquivo binário (o *blob*) inteiro, o que justifica essa simplificação nos sistemas de objetos. Isso também traz outras vantagens como garantia de autenticidade e, em muitos casos, sistemas de objetos são a solução preferencial

para o armazenamento de documentos como contratos, documentos de identificação de pessoas, documentos financeiros e de pagamentos etc.

Cada objeto possui um identificador único e, em geral, as soluções permitem serem associados vários metadados ao objeto. No caso de uma imagem fotográfica por exemplo, você pode pensar dados que informam o local da foto, o tipo de resolução, data e hora, etc. Uma vez carregados os objetos não podem ser modificados podem ser apenas excluídos e alguns sistemas dão suporte a controle de versão dos objetos.

Essas características trazem dão aos sistemas de armazenamento de objetos vantagens de simplicidade, desempenho e confiabilidade para lidar com dados tipo *blob* quando comparados com os sistemas de arquivos tradicionais, mas fornecem pouco suporte para pesquisa dos documentos sendo necessário que o usuário crie e conheça corretamente o esquema de identificação dos arquivos.

O **S3**, ou *Simple Storage Service*, da AWS ou o **Blob Storage Service** da Microsoft Azure são exemplos de serviços em nuvem para o armazenamento de objetos.

Model	Amazon	Google	Azure
Files	Elastic File System (EFS), Elastic Block Store (EBS)	Google Cloud attached file system	Azure File Storage
Objects	Simple Storage Service (S3)	Cloud Storage	Blob Storage Service
Relational	Relational Data Service (RDS), Aurora	Cloud SQL, Spanner	Azure SQL
NoSQL	DynamoDB, HBase	Cloud Datastore, Bigtable	Azure Tables, HBase
Graph	Titan	Cayley	Graph Engine
Warehouse analytics	Redshift	BigQuery	Data Lake

Amazon S3

O **Amazon S3** armazena os dados em objetos, que consistem em **um arquivo e seus metadados**. **Lembre-se, na internet não são reconhecidas as extensões dos arquivos que precisam ser fornecidas como metadados!** Cada objeto é armazenado em um bucket, que é um contêiner para armazenar

objetos. (Cuidado não confunda o contêiner com um 'docker'! aqui contêiner denomina uma 'estrutura' escalável de armazenamento)

Diferentemente dos sistemas de arquivos tradicionais, o S3 é **altamente escalável e tolerante a falhas**. Os dados são distribuídos automaticamente em vários servidores para garantir a disponibilidade e durabilidade.

Acesso. O acesso pode ser feito via API RESTful da AWS para que os desenvolvedores possam integrar o armazenamento de objetos nos aplicativos. Claramente **não podemos empregar file read() ou file write()**. Adicionalmente, pode ser acessado através de uma interface de console da AWS e ferramentas de linha de comando (CLI).

Segurança. Há vários recursos de segurança disponíveis como criptografia dos dados e de trânsito, controle de acesso baseado em políticas, autenticação multifator e monitoramento de integridade de dados.

Durabilidade e Recuperação. Há vários recursos de recuperação de desastres, como a replicação de dados entre regiões e o backup de dados em fita.

Há ainda vários recursos adicionais, como a capacidade de hospedar sites estáticos, a integração com outros serviços da AWS (**por exemplo um Lambda, um programa, pode ser ativado sempre que um arquivo é carregado. Para que você empregaria isso?**), capacidade de definir políticas de ciclo de vida para automatizar a exclusão ou movimentação de objetos após um determinado período de tempo, etc.

Exemplo de acesso RESTful

```
import boto3

# Configuração das credenciais de acesso
aws_access_key_id = 'SUA_ACCESS_KEY'
aws_secret_access_key = 'SUA_SECRET_KEY'
region_name = 'NOME_DA_REGIAO'

# Criando uma conexão com o S3
```

```
s3 = boto3.client('s3',
                  aws_access_key_id=aws_access_key_id,
                  aws_secret_access_key=aws_secret_access_key,
                  region_name=region_name)

# Criando um cliente do S3
s3 = boto3.client('s3')

# Fazendo o download de um objeto do S3
s3.download_file('nome_do_bucket',          'nome_do_objeto',
                'nome_do_arquivo_local')

# Upload de um objeto para o S3
s3.upload_file('nome_do_arquivo_local',     'nome_do_bucket',
                'nome_do_objeto')
```

Todas essas facilidades são encontradas de forma similar nos principais provedores de serviços de nuvem.

Referências

Ian Foster and Dennis B. Gannon. **Cloud Computing for Science and Engineering** (1st. ed.), 2017. The MIT Press. Alternativamente Disponível em: <https://cloud4scieng.org/> Acesso: 10.03.2021.