

大数据的信息基础设施

存储模型

陈一帅

yschen@bjtu.edu.cn

北京交通大学电子信息工程学院

内容

- 基本概念
- 分布式存储
- 存储模型

基本概念

数据操作事务

- 事务
 - Transaction
 - 一系列数据操作组成的一个完整的逻辑过程
- 例
 - 银行转帐，从原账户扣除金额，以及向目标账户添加金额
 - 这两个数据库操作的总和，构成一个完整的逻辑过程，不可拆分
 - 这个过程被称为一个事务

存储系统 ACID 特性

- 数据库系统在写入或更新资料过程中，为保证事务正确可靠，必须具备的四个特性
- 原子性 (atomicity) : 不可分割性
- 一致性 (consistency)
- 隔离性 (isolation) : 独立性
- 持久性 (durability)

内容

- 基本概念
- 分布式存储
- 存储模型

分布式存储

- 扩展数据库的明显解决方案是在多台计算机之间分布和/或复制数据
- 例如，通过分布不同的表或同一表的不同行
- 但分布和复制也带来挑战

云计算中的分布式存储

- 存储作为一种服务
- 分布式存储提供大数据存储服务
- 6 种存储模型

Table 2.1: Storage as a service options from major public cloud vendors.

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

1、文件系统

- 以 Amazon AWS 文件系统为例
 - 提供两种文件系统
- Elastic Block Store - 弹性块存储 (EBS)
 - EC2 实例低延迟访问
 - 存储工作数据，支持频繁读取和写入，但太大而无法容在内存中
- Elastic File System - 弹性文件系统 (EFS)
 - 通用文件存储
 - 为 EC2 实例提供文件系统访问语义（强一致性，文件锁），并发进程可同时访问（读取和写入）

2、对象存储

- 存储非结构化二进制对象
 - 二进制大型对象通常被称为 blob
- 对象由唯一的标识符标识
- 对象可具有与之关联的元数据
- 对象一旦上传就无法修改，只能删除
 - 或在支持版本控制的对象存储中进行版本升级

对象存储模型的优势

- 在简单，性能和可靠性方面，具有重要优势
- 简化了文件系统模型
 - 消除了层次结构
- 一旦创建就无法修改的特性使构建高可靠、可扩展的实现变得容易
 - 可在多个物理存储设备之间复制每个对象，以提高弹性和（当有多个并发读取器时）性能
 - 无需专门的同步逻辑来处理并发更新
- 可在具有不同性能和成本参数的存储类别之间手动或自动移动对象

对象存储模型的局限

- 几乎不支持对数据进行组织，不支持搜索
 - 用户必须知道对象的标识符才能访问它
- 不提供结构化处理数据的机制
 - 可能必须将整个对象下载到计算机上以对其内容进行计算
- 无法以文件系统的方式将对象作为文件系统挂载，或使用现有工具进行访问

对象存储操作示例

- 以 Amazon AWS 为例
 - 采用 AWS boto3 Python API
- 第一步：创建 S3 资源
 - 设定资源的访问密钥

```
import boto3
s3 = boto3.resource('s3',
                    aws_access_key_id='YOUR ACCESS KEY',
                    aws_secret_access_key='your secret key' )
```

对象存储操作示例

- 第二步：创建 S3 存储桶 (Bucket)
 - 在资源中创建 Bucket, 名称: datacont
 - 指定其地理区域
- 将在其中存储数据 objects

```
import boto3
s3 = boto3.resource('s3')
s3.create_bucket(Bucket='datacont', CreateBucketConfiguration={
    'LocationConstraint': 'us-west-2'})
```

对象存储操作示例

- 第三步：将数据对象加载到桶里
 - 上传图片到 Bucket 中

```
# Upload a file, 'test.jpg' into the newly created bucket
s3.Object('datacont', 'test.jpg').put(
    Body=open('/home/mydata/test.jpg', 'rb'))
```

3、关系数据库

- 适用于中等大小、高度结构化数据集
- MySQL

4、NoSQL 数据库

- NoSQL 数据库
 - 支持大规模数据量和用户量
 - 处理不容易以表格形式表示的非结构化数据
- 键值存储
 - 可组织大量记录
 - 每个记录将任意键与任意值相关联

NoSQL DB 的 ACID 特性

- 云中的 NoSQL 数据库通常分布在多个服务器上，并且还复制到不同的数据中心
- 因此，它们通常无法满足所有的 ACID 属性
- 一致性通常被最终的一致性所取代
- 这意味着跨副本的数据库状态可能暂时不一致

NoSQL DB 操作示例

- 以 AWS DynamoDB 为例
 - AWS 的大数据平台 EMR 基于 DynamoDB
- 基于可扩展键值模型
- 通过行分配数据
 - 一行中的所有元素都映射到同一设备
 - 要确定数据所在设备，只需查找 PartitionKey
 - 该哈希将散列到行所在的物理存储设备的索引
- 可通过 Elasticsearch 进行文本搜索

其它存储模型

- 图数据库
 - 存储关系
- 数据仓库
 - 优化查询

小结

- 基本概念
 - 事务
 - ACID
- 分布式存储
- 存储模型
 - 文件系统
 - Object 存储
 - 关系数据库
 - NoSQL 数据库

练习

- 利用云计算账号，练习至少一种数据模型的操作
 - 界面操作
 - 编程操作