北京交通大学
BEIJING JIAOTONG UNIVERSITY

大数据概论

数据管理

- 数据管理概述
- 关系数据库
- 分布式文件系统
- 新型数据管理与查询系统

数据管理概述

- 本章重点介绍数据存储与管理技术的概念与发展过程，选择经典的关系数据库技术以及大数据时代的分布式文件系统技术、NoSQL与Sql on Hadoop技术新型大数据存储与查询技术进行介绍。

数据管理的内涵

- 数据管理技术
  - 数据管理技术是指对数据进行分类、编码、存储、索引和查询，是大数据处理流程中的关键技术，负责数据从落地存储（写）到查询检索（读）的核心系统。数据管理技术从最早人们使用文件管理数据，到数据库、数据仓库技术的出现与成熟，再到大数据时代NoSQL等新型数据管理系统的涌现，一直是数据领域研究和工程领域的热点。
- 数据库
  - 数据库(Database)是按照数据结构来组织、存储和管理数据的建立在计算机存储设备上的仓库。简单来说是本身可视为电子化的文件柜，用户可以对文件中的数据进行新增、截取、更新、删除等操作。严格来说，数据库是长期储存在计算机内、有组织的、可共享的数据集合。

数据管理历史

- 关系数据库
  - 上世纪70年代，IBM公司的E.F.Codd开创了关系数据库理论，80年代随着事务处理模型的完善，关系数据管理在学术届和工业界取得主导地位，并一直保持到今天。关系数据库的核心是将数据保存在由行和列组成的简单表中，而不是将数据保存在一个层次结构中。Codd开创了关系数据库和数据规范化理论研究，获得了1981年的图灵奖，关系数据库也很快成为数据库市场的主流。
- 新型数据管理与查询系统
  - 2010年前后，美国谷歌公司为满足搜索业务的需求，推出了以分布式文件系统GFS（Google File System）、分布式计算框架MapReduce、列族数据库BigTable为代表的新型数据管理与分布式计算技术。Doug Cutting领衔的技术社区研发了对应的开源版本，在Apache开源社区推出，形成了Hadoop大数据技术生态，不断迭代发展出一系列大数据时代的新型数据管理技术，例如面向内存计算的Spark大数据处理软件栈，MangoDB、Cassandra等各类型NoSQL数据库，Impala、SparkSQL等分布式数据查询技术（Sql on Hadoop）。

关系数据库

- 关系数据库建立在关系数据模型之上，是主要用来存储结构化数据并支持数据的插入、查询、更新、删除等操作的数据库。

关系模型

- 关系数据模型是以集合论中的关系概念为基础发展起来的。关系数据模型中无论是实体还是实体间的联系均由单一的数据结构——关系来表示。关系数据模型中对的数据操作通常由关系代数和关系演算两种抽象操作语言来完成，此外关系数据模型中还通过实体完整性、参照完整性和自定义完整性来确保数据的完整一致
- 关系数据模型的基本数据结构就是关系（Relation），一个关系对应着一个二维表，二维表的名字就是关系名。

关系名

表3-1 学生表

学号	姓名	性别	年龄	图书证号	所在系
S3001	张明	男	22	B20050101	外语
S3002	李静	女	21	B20050102	外语
S4001	赵丽	女	21	B20050301	管理

关系模型：数据结构

- 从横向看，二维表中的一行被称为是关系中的一个元组（Tuple），关系本质上就是由同类元组构成的集合。
- 从纵向看，二维表由很多列构成，列被称为关系的属性（Attribute），同一个集合中的元组都由同样的一组属性值组成。
- 属性的取值范围被称为域（Domain），它也可以被理解为属性中值的数据类型。

属性域：男，女

表3-1 学生表

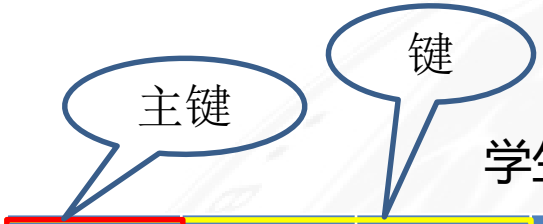
学号	姓名	性别	年龄	图书证号	所在系
S3001	张明	男	22	B20050101	外语
S3002	李静	女	21	B20050102	外语
S4001	赵丽	女	21	B20050301	管理

一个属性

一个元组

关系模型：数据结构

- 如果在一个关系中存在唯一标识一个元组的属性集合（可以是单一属性构成的集合），则称该属性集合为这个关系的键或码。
- 用来唯一标识一个元组的最小属性集合，称为主键（主码）。



学生表

学号	姓名	性别	年龄	图书证号	所在系
S3003	张磊	男	22	B20050101	外语
S3002	李静	女	21	B20050102	外语
S4006	张磊	女	21	B20050301	管理

关系模型：数据操作

- 关系数据模型的数据操作分为查询和更新两类。
  - 关系更新可细分：插入（Insert）、修改（Update）、删除（Delete）；
  - 关系查询包括：选择（Select）、投影（Project）、并（Union）、差（Except）以及连接（Join）等。

    - 插入：将一个新的元组（行）加入到现有的关系中。
    - 修改：对关系中已有的数据进行修改，特指对各种属性值进行修改。
    - 删除：如果关系中的一行或多行数据已经不再需要，可以用删除操作将它们从关系中彻底去掉。

关系模型：数据操作

学号	姓名	性别	年龄	图书证号	所在系
S3001	张明	男	22	B20050101	外语
S3002	李静	女	21	B20050102	外语
S4001	赵丽	女	21	B20050301	管理

学号	课程号	成绩
S3001	C1	90
S3001	C2	95
S3002	C1	84
S4001	C3	50

学号	姓名	性别	年龄	图书证号	所在系	课程号	成绩
S3001	张明	男	22	B20050101	外语	C1	90
S3001	张明	男	22	B20050101	外语	C2	95
S3002	李静	女	21	B20050102	外语	C1	84
S4001	赵丽	女	21	B20050301	管理	C3	50

投影

学号	姓名	所在系	课程号	成绩
S3001	张明	外语	C1	90
S3001	张明	外语	C2	95
S3002	李静	外语	C1	84
S4001	赵丽	管理	C3	50

- 连接：把分散在不同关系中的数据关联在一起查看。我们希望查看学生选课的情况，但选课表中只有学生的学号，没有学生的姓名及所在系等信息（在学生表中），此时就需要将学生表和选课表连接起来。
- 选择：从关系中选取满足条件的元组，例如从学生表中选出所有的男学生。
- 投影：从关系中抽取出若干属性形成一个新的关系，例如我们只抽取学生的姓名和所在系。
- 并和差：并操作就是将两个同类关系中的元组集合合并起来形成新的关系，差操作则是从两个同类关系的元组集合中找出不同的元组集合。

结构化查询语言

- 结构化查询语言（Structured Query Language）简称SQL，是一种数据库查询和程序设计语言，用于查询、更新和管理关系数据库系统。
- 结构化查询语言是高级的非过程化编程语言，允许用户在高层数据结构上工作。它不要求用户指定对数据的存放方法，也不需要用户了解具体的数据存放方式，所以即使是具有完全不同底层结构的不同数据库系统，也可以使用相同的结构化查询语言作为数据输入与管理的接口。结构化查询语言语句可以嵌套，这使它具有极大的灵活性和强大的功能。

结构化查询语言：SQL历史

- SQL是IBM在其System R系统中首次提出的。1979年ORACLE公司首先提供商用的SQL，其后IBM公司在DB2和SQL/DS数据库系统产品中也实现了SQL。
- 1986年10月，美国ANSI采用SQL作为关系数据库管理系统的标准语言（ANSI X3. 135-1986），后为国际标准化组织（ISO）采纳为国际标准。
- 1989年，美国ANSI采纳在ANSI X3.135-1989报告中定义的关系数据库管理系统的SQL标准语言，称为ANSI SQL 89，该标准替代ANSI X3.135-1986版本。
- 之后每隔一定时间ISO都会更新新版本的SQL标准，目前最新的版本已经演进到2016。

结构化查询语言：SQL历史

- 按照不同的用途，SQL语言通常被分成三个子集（子语言）：
  - 数据定义语言（DDL：Data Definition Language），用于操纵数据库模式，例如数据库对象（表、视图、索引等）的创建和删除。数据定义语言的语句包括动词CREATE和DROP，之后用数据库对象的类型名词区分要定义的数据库对象，例如TABLE、VIEW、INDEX。
  - 数据操作语言（DML：Data Manipulation Language），用于对数据库中的数据进行各类操作，包括读取和修改，其语句包括动词SELECT、INSERT、UPDATE和DELETE。它们分别用于查找、增加、修改和删除表中的行。
  - 数据控制语言（DCL：Data Control Language），包括除DDL和DML之外的其他杂项语句，这些语句包括对访问权限和安全级别的控制、事务的控制、连接会话的控制等。

数据库事务

- 为防止不同用户同时操作同一数据时产生的不良影响，现代的数据库管理系统中都引入了事务（Transaction）的概念。事务由一系列的数据库操作构成，它必须满足四个特性（被简称为ACID特性）：
  - （1）原子性（Atomicity）：事务所包含的所有操作要么全部正确地反映在数据库中，要么全部不反映；
  - （2）一致性（Consistency）：事务的执行会使数据库从一种一致性的状态达到另一种一致性状态，即事务的执行不会让数据库出现不一致；
  - （3）隔离性（Isolation）：事务之间是隔离的，每个事务都感觉不到系统中有其他事务在并发地执行；
  - （4）持久性（Durability）：一个事务成功完成后，它对数据库的改变是永久的，即使系统出现故障也是如此。

关系数据库管理系统

- 关系数据库管理系统（Relational Database Management System：RDBMS）是管理、操作和维护关系型数据库的一种软件程序。

分布式文件系统

- 分布式文件系统建立在通过网络联系在一起的多台价格相对低廉的服务器上，将要存储的文件按照特定的策略划分成多个片段分散放置在系统中的多台服务器上。

分布式文件系统分类

- 从用途来看，目前主流的分布式文件系统主要有两类：
  - 第一类分布式文件系统主要面向以大文件、块数据顺序读写为特点的数据分析业务，其典型代表是Apache旗下的HDFS。
  - 另一类主要服务于通用文件系统需求并支持标准的可移植操作系统接口（Portable Operating System Interface of UNIX，缩写为 POSIX），其代表包括Ceph和GlusterFS。
- 这种分类仅表示各种分布式文件系统的专注点有所不同，并非指一种分布式文件系统只能用于某种用途。

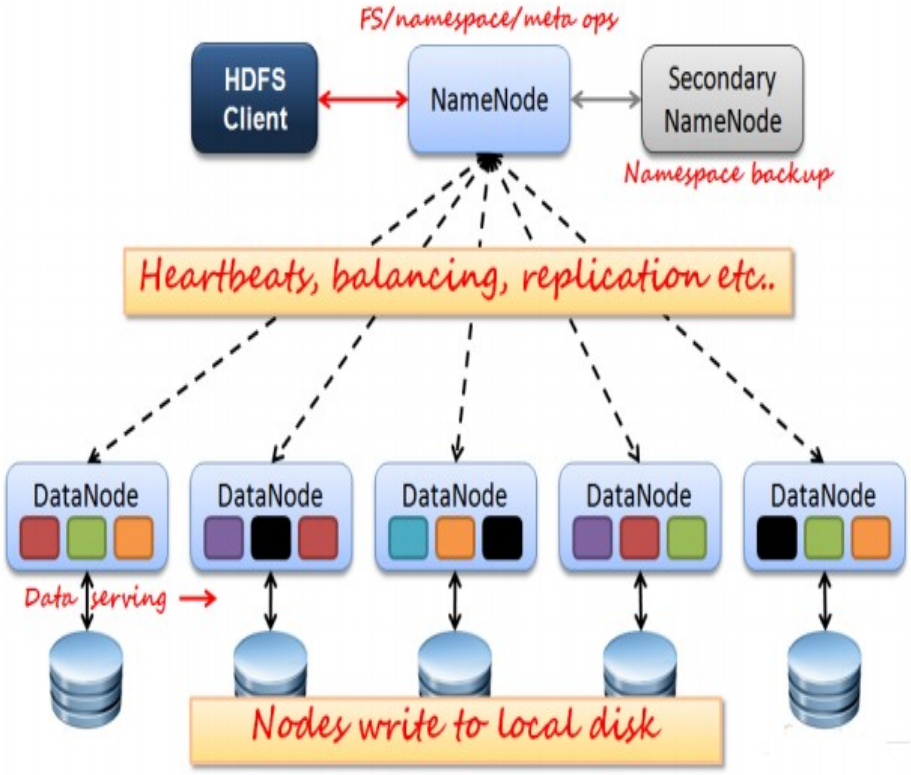
Hadoop分布式文件系统（HDFS）：特点

- HDFS作为Hadoop的分布式文件系统，其功能为数据的存储、管理和出错处理。它是类似于GFS的开源版本，设计的目的是用于可靠地存储大规模的数据集，并提高用户访问数据的效率。

A	B	C	D
适合大文件存储和处理	集群规模可动态扩展	能有效保证数据一致性	数据的吞吐量大，跨平台移植性好
可处理的文件规模可达到百MB乃至数百TB，目前应用已到PB级。	存储节点可在运行状态下加入到集群中，集群仍然可以正常地工作。	基于“一次写入，多次读取”设计，简化处理文件访问方式，当一个文件创建、写入并关闭后就不能再修改。	采用数据流式读写的方式，用以增加数据的吞吐量。具有很好的跨平台移植性，源代码开放。

Hadoop分布式文件系统（HDFS）：架构和操作

- HDFS采用的是单一主服务器的主从结构，一个HDFS集群通常由一台主服务器和若干台数据服务器构成，有一台后备主服务器用于定期对主服务器存储的元数据进行备份，保障名称空间、元数据等系统信息的完整性。这台后备主服务器只与主服务器进行交互，对系统中的其他节点不可见。



Active Namenode

1.主Master(只有一个)
2.管理HDFS的名称空间
3.管理数据块映射信息
4.配置副本策略
5.处理客户端读写请求

Standy Namenode

1.NameNode的热备
2.定期合并fsimage和fsedit，推送给NameNode
3.当Active NameNode出现故障时，快速切换为新的Active NameNode

Datanode

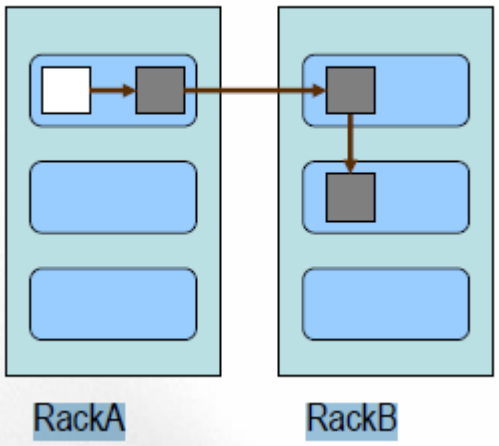
1.Slave（有多个）
2.存储实际的数据块
3.执行数据块的读写

Client

1.文件切分
2.与NamdeNode交互，获取文件位置信息
3.与DataNode交互，读取或者写入数据;
4.管理HDFS 和访问HDFS

Hadoop分布式文件系统（HDFS）：副本管理

- 为了提高系统中文件数据的可靠性，HDFS系统提供了一种副本机制：默认情况下，每一个文件块都会在HDFS系统中拥有三个副本，副本数可以在部署集群时手动设置。通常这三个副本会被放置在不同的数据服务器上，这样就保证了即便其中某一个副本丢失或者损坏，都可以保证该文件块可以继续使用，甚至还可以利用其他两个副本来恢复丢失或者损坏的那个副本。



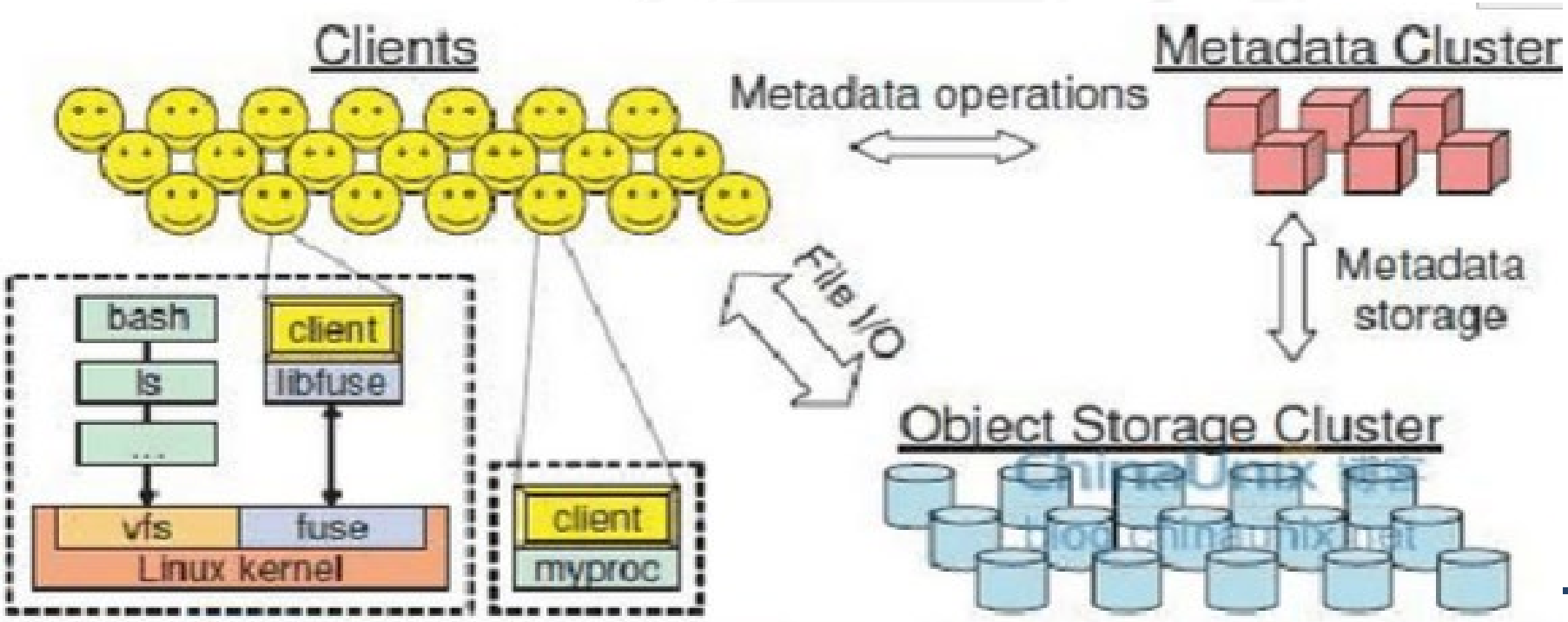
- 从应用场景来看，HDFS是专门为Hadoop这样的计算引擎而生，更适合离线批量处理大数据，例如电商网站对于用户购物习惯的分析。由于HDFS本身设计的特点，它不适合于经常要对文件进行更新、删除的在线业务。

Ceph

- Ceph是一个高可用、易于管理、开源的分布式存储系统，可以同时提供对象存储、块存储以及文件存储服务，其优势包括统一存储能力、可扩展性、可靠性、性能、自动化的维护等等。
- Ceph优势均来源于其先进的核心设计思想，可其概括为八个字——“无需查表，算算就好”。基于这种设计思想，Ceph充分发挥存储设备自身的计算能力，同时消除了对系统单一中心节点的依赖，从而实现了真正的无中心结构。
- Ceph项目起源于其创始人Sage Weil在加州大学圣克鲁兹分校攻读博士期间的研究课题。

Ceph

- 客户端通过与OSD（Object Storage Device）的直接通讯实现文件操作。在打开一个文件时，客户端会向MDS（Metadata storage ）发送一个请求。MDS把请求的文件名翻译成文件节点（inode），并获得节点号、访问模式、大小以及文件的其他元数据。如果文件存在并且客户端可以获得操作权，则MDS向客户端返回上述文件信息并且赋予客户端操作权。



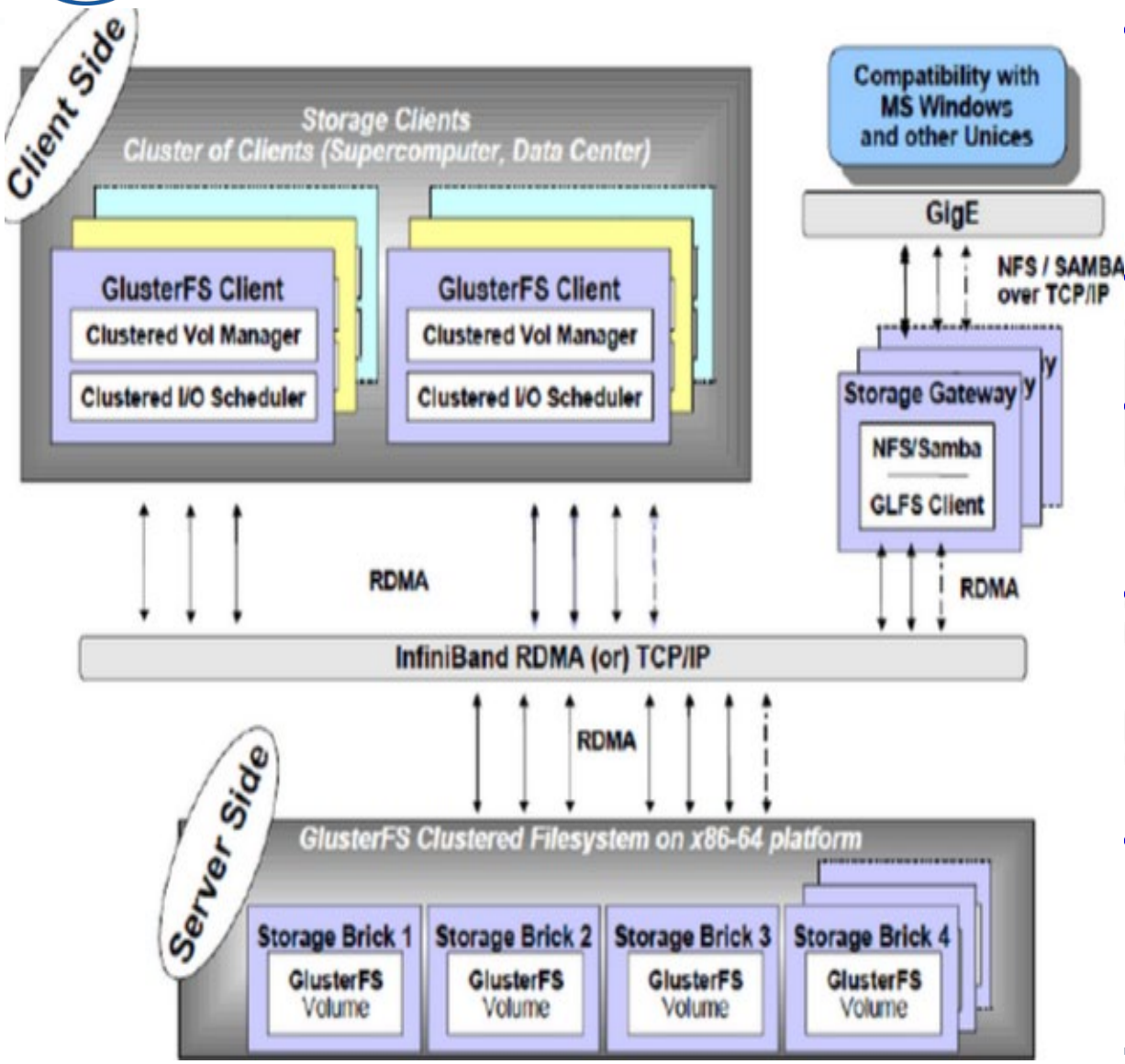
Ceph

- 相对于面向离线批处理的HDFS来说，Ceph更偏向于成为一种高性能、高可靠、高扩展性的实时分布式存储系统，其对于写入操作特别是随机写入的支持要更好。

GlusterFS

- GlusterFS是Scale-Out存储解决方案Gluster的核心，它是一个开源的分布式文件系统，具有强大的横向扩展能力，通过扩展能够支持数PB存储容量和处理数千客户端。
- GlusterFS借助TCP/IP或InfiniBand RDMA网络将物理分布的存储资源聚集在一起，使用单一全局命名空间来管理数据。GlusterFS基于可堆叠的用户空间设计，可为各种不同的数据负载提供优异的性能。

GlusterFS



- Storage Brick: GlusterFS中的存储单元，可以通过主机名和目录名来标识。
- Storage Client: 挂载了GlusterFS卷的设备
- RDMA:远程直接内存访问，支持不通过双方的OS进行直接内存访问。
- RRDNS: round robin DNS，是一种通过DNS轮转返回不同的设备以进行负载均衡的方法。
- Self-heal: 用于后台运行检测副本中文件和目录的不一致性并解决这些不一致。

GlusterFS

- GlusterFS支持运行在任何标准IP网络上标准应用程序的标准客户端，用户可以在全局统一的命名空间中使用NFS/CIFS等标准协议来访问应用数据。
- GlusterFS使得用户可摆脱原有的独立、高成本的封闭存储系统，能够利用普通廉价的存储设备来部署可集中管理、横向扩展、虚拟化的存储池，存储容量可扩展至TB/PB级。
- GlusterFS由于缺乏一些关键特性，可靠性也未经过长时间考验，还不适合应用于需要提供 24 小时不间断服务的产品环境。目前适合应用于大数据量的离线应用。

分布式文件系统对比

特性	HDFS	Ceph	GlusterFS
元数据服务器	单个 存在单点故障风险	多个 不存在单点故障风险	无 不存在单点故障风险
POSIX兼容	不完全	兼容	兼容
配额限制	支持	支持	不详
文件分割	默认分成64MB块	采用RAID0	不支持
网络支持	仅TCP/IP	多种网络，包括 TCP/IP、Infiniband	多种网络，包括TCP/IP、 Infiniband
元数据	元数据服务器管理全量元数据	元数据服务器管理少量元数据	客户端管理全量元数据
商业应用	大量，国内包括中国移动、百度、网易、淘宝、腾讯、华为等	非常不成熟，尚不适合生产环境	测试和使用案例多为欧美，国内用户很少

NoSQL数据库

NoSQL（Not only SQL）数据库是对于非关系型的一类数据库系统的统称。它针对关系型数据库在管理键值对、文档、图等类型数据上的不足，针对各个类型数据的存储和访问特点而专门设计的数据库管理系统。

- NoSQL数据库设计原则：
  - 采用横向扩展（Scaling Out）的方式，通过对大量节点的并行处理，获得包括读性能和写性能在内的极高数据处理性能和吞吐能力。NoSQL数据库需要对数据进行划分，以便进行并行查询处理。
  - 放弃严格的ACID一致性约束，采用放松的一致性约束条件，允许数据暂时出现不一致的情况，并接受最终一致性。
  - 对数据进行容错处理，一般对数据块进行适当备份，以应对结点失败状况，保证在普适服务器组成的集群上稳定高可靠地运行。

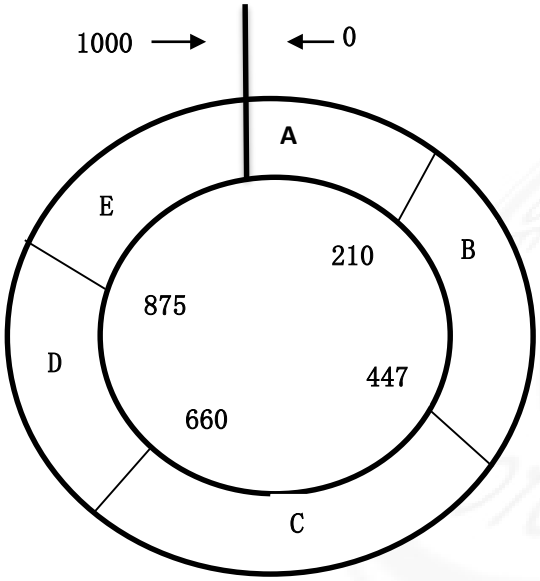
NoSQL数据库

- 四类常用NoSQL数据库技术对比

分类	相关产品	典型应用场景	数据模型	优点	缺点
键值对数据库	Tokyo Cabinet/Tyrant, Redis, Voldemort, Oracle BDB	内容缓存，主要用于处理大量数据的高访问负载	Key 指向 Value 的键值对，通常用hash table来实现	查找速度快	数据无结构化
列族数据库	Cassandra, HBase, Riak	分布式的文件系统	以列簇式存储，将同一列数据存在一起	查找速度快，可扩展性强，更容易进行分布式扩展	功能相对局限
文档数据库	CouchDB, MongoDb	Web应用（与Key-Value类似，Value是结构化的）	Key-Value对应的键值对，Value为结构化数据	数据结构要求不严格，表结构可变	查询性能不高，缺乏统一的查询语法
图数据库	Neo4J, InfoGrid, Infinite Graph	社交网络，推荐系统等，专注于构建关系图谱	图结构	利用图结构相关算法	需对整个图做计算，不容易做分布式集群方案

NoSQL数据库

- 键值对（Key-Value）数据库
  - 这一类数据库主要会使用到一个哈希表，这个表中有一个特定的键和一个指针指向特定的数据。Key/value模型对于IT系统来说的优势在于简单、易部署。如：TokyoCabinet/Tyrant, Redis, Voldemort, Oracle BDB,Memcached。



一致性哈希划分数据

Tokyo Cabinet 8192PiB

membase

redis

Storage Engine ORACLE BERKELEY DB

Project Voldemort
A distributed database.
zoie · bobo · cleo · decomposer · norbert

NoSQL数据库

- 文档数据库
  - 文档数据库技术是以键值对存储模型作为基础模型的NoSQL技术。文档存储数据库以不同标准（如JSON，XML，BSON或YAML）编码的文档形式，来存储半结构化数据。每个文档都由唯一的键key或ID来标识。在将文档存储到数据库中之前，无需为文档定义任何模式。

NoSQL数据库

- 文档数据库
  - Mongodb是一款分布式文档数据库，它为大数据量、高度并发访问、弱一致性要求的应用而设计。Mongodb具有高扩展性，在高负载的情况下，可以通过添加更多的节点，保证系统的查询性能和吞吐能力。
  - Mongodb数据库支持增加、删除、修改、简单查询等主要的数据操作以及动态查询，并且可以在复杂属性上建立索引，当查询包含该属性的条件时，可以利用索引获得更高的查询性能。

使用文档数据库来存储商品记录

ID	Document

34fd459fs523f3f34d43325	{
	“标题”：“IPhone 8 Plus”
	“特点”：[
	“屏幕尺寸 5.5英寸”
	“后置摄像头 1200万”
	“存储容量 64GB ”
	“运行内存 6GB ”
	“操作系统 iOS ”
	]
	“价格”：5999元
	}

NoSQL数据库

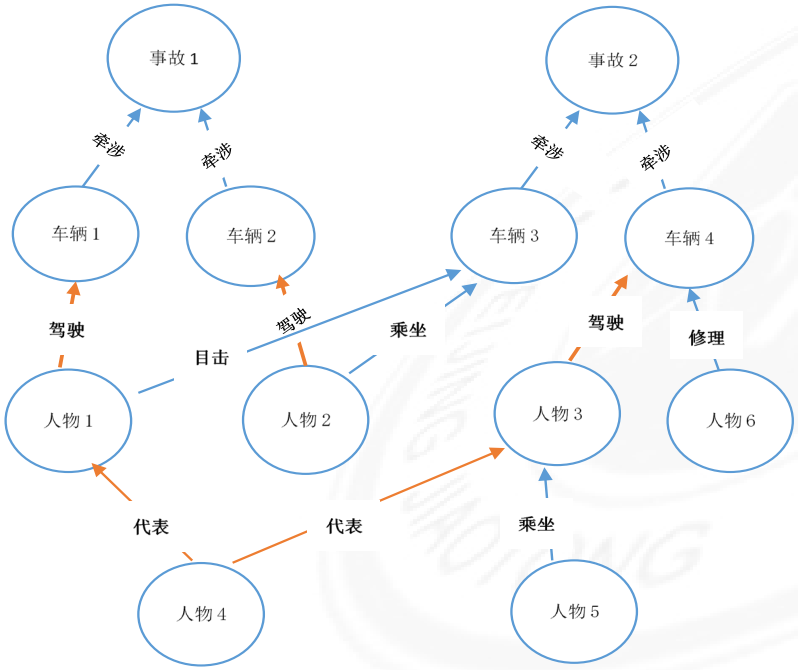
- 列族（Column Family）数据库
  - 通常是用来应对分布式存储海量数据。数据存储的基本单位是一个列，它具有一个名称和一个值。由列的集合组成的每一行，通过行-键标识来标示，列组合在一起成为列族。与关系数据库不同，列族数据库不需要在每行中都有固定的模式和固定数量的列。

HBase表存储结构

HBase表		
RowKey	ColumnFamily-1	ColumnFamily-2
记录1	列1……列n	列1，列2，列3
记录2	列1，列2	
记录3	列1……列5	列1

NoSQL数据库

- 图数据库
  - 图数据库是用于专门存储具有节点和边的图结构数据的一类数据库，并以节点和边作为基本数据模型。节点可以代表数据模型中的重要的实体或信息条目，节点之间的关系以边的形式表示。



某交通保险欺诈关系图

SQL on Hadoop系统

- 互联网公司最先遇到大数据难题，需要为海量互联网网页构建倒排列表。2004年，Google公司提出 MapReduce技术，作为面向大数据分析和处理的并行计算模型，引起了工业界和学术界的广泛关注。 Hadoop技术很快也影响了数据库研究领域，有面向简单的键值对读写事务型负载的NoSQL系统（如HBase等），也有面向数据分析任务的Hive系统。
- Hive系统的出现，一改传统的OLAP只能在关系数据仓库中运行的局面，从而可以对HDFS中存储的结构化数据，基于一种类似SQL的HiveQL语言，进行ROLAP方式的数据分析。
- 来自互联网领域或者其他领域很多大数据创新公司，并没有止步于Hive。最近五六年间做出了很多努力，开发了多个SQL on Hadoop系统，以提升这些系统的性能。这些系统借鉴了20世纪90年代以来在并行数据库方面所积累的一些先进技术，大幅度提升了SQL on Hadoop系统的性能。

SQL on Hadoop系统

- **Hive**
  - 自从Facebook在2007年推出Apache Hive系统及其HiveQL语言以来，已经成为Hadoop平台标准的SQL实现。Hive把HiveQL查询首先转换成MapReduce作业，然后在Hadoop集群上执行。某些操作（如连接操作）被翻译成若干个MapReduce作业，依次执行。
  - 近年来，开源社区对Hive进行持续改进，主要包括以下几个方面：
    - 在SQL接口方面，增加了新的数据类型、子查询支持、更加完备的Join语法等。
    - 在文本类型、RCFile列存储格式之外，增加了具有更高效率的列存储格式ORCFile。
    - 和Tez紧密集成，以便执行更通用的任务，获得更高的性能。
    - 增加初步的查询优化能力，能根据数据特点，进行表连接顺序调整和连接算法选择。
    - 新的向量化的查询执行引擎，通过更好地利用现代CPU的特点，提高查询性能。

SQL on Hadoop系统

- Impala
  - Impala是由Cloudera公司推出的一个支持交互式（实时）查询的SQL on Hadoop系统。Impala放弃使用效率不高的MapReduce计算模型，设计专有的查询处理框架，把执行计划分解以后，分配给相关节点运行，而不是把执行计划转换为一系列的MapReduce作业。
  - Impala不把中间结果持久化到硬盘上，而是使用MPP数据库惯用的技术，即基于内存的数据传输，在各个操作之间传输数据。 在连接操作的处理方面，Impala根据表的绝对和相对大小，在不同的连接算法之间进行选择。
  - 根据Cloudera的评测结果，对于I/O限制的查询，相对于老版本的Hive，Impala有3-4倍的性能提升。
  - Impala令人印象深刻的性能使人们相信，只要充分利用各种优化措施，包括存储优化、执行引擎优化、查询优化等技术，Hadoop平台上的SQL查询也能达到交互式的性能要求。

SQL on Hadoop系统

- Spark SQL
  - Spark SQL是美国加州大学伯克利分校提出的大数据处理框架BDAS（Berkeley data analytics stack）的一个重要组成部分，包括资源管理层、存储层、核心处理引擎、存取接口、应用层等层次和部件。
  - Spark SQL是实现大数据交互式SQL查询的处理系统，包括接口Spark SQL和处理引擎Spark Core。Spark是一个分布式容错内存集群，通过基于血统关系的数据集重建技术，实现内存计算的容错。

SQL on Hadoop系统

- Spark SQL
  - Spark SQL使用内存列存储技术支持分析型应用。在复杂查询执行过程中，中间结果通过内存进行传输，无需持久化到硬盘上，极大地提高了查询的执行性能。
  - Spark SQL在设计上实现了和Apache Hive在存储结构、序列化和反序列化方法、数据类型、元信息管理等方面的兼容。此外，BDAS还支持流数据处理和图数据的计算，并通过迭代计算支持各种机器学习算法。
  - 新版本的Spark SQL还计划支持数据并置以及部分DAG执行技术，允许系统根据运行时搜集的统计信息，动态改变执行计划，以获得更高的性能。

Thank You