

# 大数据库系统

## 第1章 大数据库系统课程概述

主讲人：苏立超

# 第1章 大数据数据库系统课程概述

## ◆ 主要内容:

- 1、大数据数据库系统课程简介及要求
- 2、大数据数据库系统课程内容
- 3、一些重要的概念、工具概述

Redis

MongoDB

数据仓库

Hadoop下的数据仓库Hive

云数据库

# 1.1 课程简介及要求

## ◆ 大数据数据库系统理论课

共40个课时

考核方式：笔试70%+平时30%

## ◆ 大数据数据库系统实践课

共24个课时

总共6次实验课

## ◆ 老师联系方式：

大数据数据库课程群：859409244

Email: [651424071@qq.com](mailto:651424071@qq.com)

QQ: 651424071



## 1.2 大数据数据库系统课程内容

### ◆ 大数据系统理论课（共40课时）

第1章 大数据数据库系统课程概述（2课时）

第2章 NoSQL与NewSQL（4课时）

第3章 键值数据库Redis（10课时）

第4章 文档数据库MongoDB（3课时）

第5章 数据仓库（3课时）

第6章 数据仓库Hive（10课时）

第7章 数据仓库Impala（2课时）

第8章 云数据库（4课时）

总复习（2课时）

共计40课时

# 1.2 大数据数据库系统课程内容

## ◆ 大数据系统理论课

### 1. NoSQL数据库

- 理论概念、Redis、MongoDB

### 2. 数据仓库

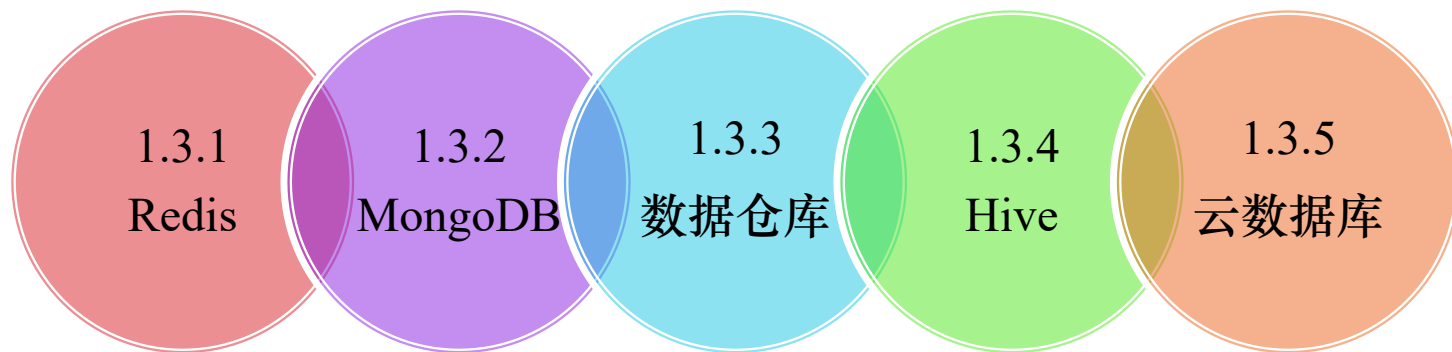
- 理论概念、Hive、Impala

### 3. 云数据库

- 理论概念、云数据库产品

## 1.3 重要的概念、工具

### ◆ 本节内容



## 1.3.1 Redis

### ◆ Web开发过程数据存储

日常的Web开发中，通常会使用数据库来进行数据的存储

### ◆ 涉及大数据量时

商品抢购

活动开放

秒杀、预约

等

主页访问量瞬间较大

该怎么办？

## 1.3.1 Redis

### ◆ 传统数据库性能弊端

- 数据库需要涉及磁盘的读写，然而磁盘读/写速度比较慢
- 成千上万的请求到来，需要系统在极短的时间内完成成千上万次的读写操作

### ➤ 最终导致

- 数据库无法承受
- 容易造成数据库系统瘫痪
- 最终导致服务器宕机等严重生产问题

造成经济损失等



## 1.3.1 Redis

### ◆ 引入NoSQL

尤其是基于内存的数据库

提供一定的持久化功能

Redis和MongoDB是当前使用最广泛的NoSQL

Redis是一种键值数据库

MongoDB是一种文档数据库

## 1.3.1 Redis

### ◆ Redis

- 性能优越：支持每秒十几万次读写操作，远超传统关系型数据库
- 支持分布式集群
- 支持持久化
- 主从复制，原则上可以无限扩展，让更多的数据存储在内存中
- 支持一定的事务能力等

保证了高并发的场景下数据的安全和一致性

## 1.3.1 Redis

### ◆ Redis在Web中的应用

存储缓存用的数据

需要高速高并发读写的场合使用它快速读写

等等



## 1.3.1 Redis

### ◆ Redis应用之一：做缓存

➤ 日常对数据库的访问中，读操作的次数远超写操作

- 比例大概在 1:9 到 3:7，需要读的可能性比写大得多

➤ 使用SQL语句去数据库进行读写操作时比较慢

- 数据库会去磁盘把对应的数据索引取回来，这是一个相对较慢的过程

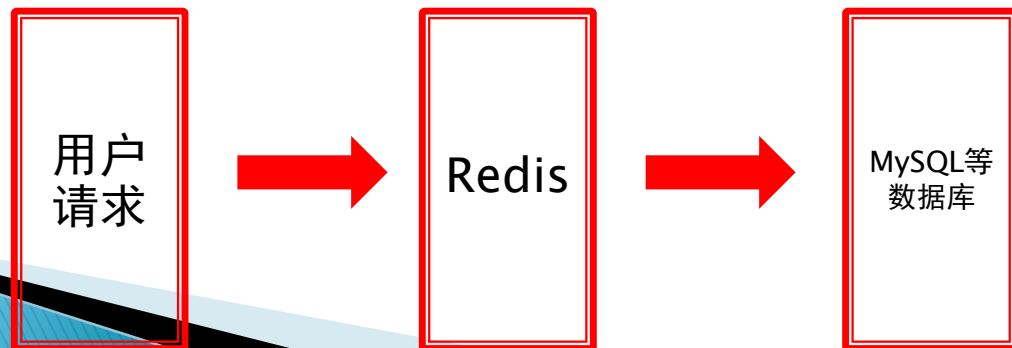
## 1.3.1 Redis

### ◆ Redis应用之一：做缓存

把数据放在 Redis 中，也就是直接放在内存之中，让服务端直接去读取内存中的数据

- 速度快
- 极大减小数据库的压力

使用内存进行数据存储价格比较高，限于成本的原因



## 1.3.1 Redis

◆ 使用内存进行存储的时候，需要从以下几个方面来考虑

➤ 业务数据常用吗？命中率如何？

- 如果命中率很低，就没有必要写入缓存；

➤ 该业务数据是读操作多，还是写操作多？

- 如果写操作多，频繁需要写入数据库，也没有必要使用缓存；

## 1.3.1 Redis

### ◆ Redis读流程

**当第一次读取数据的时候：**读取 Redis 的数据就会失败，此时就会触发程序读取数据库，把数据读取出来，并且写入 Redis 中；

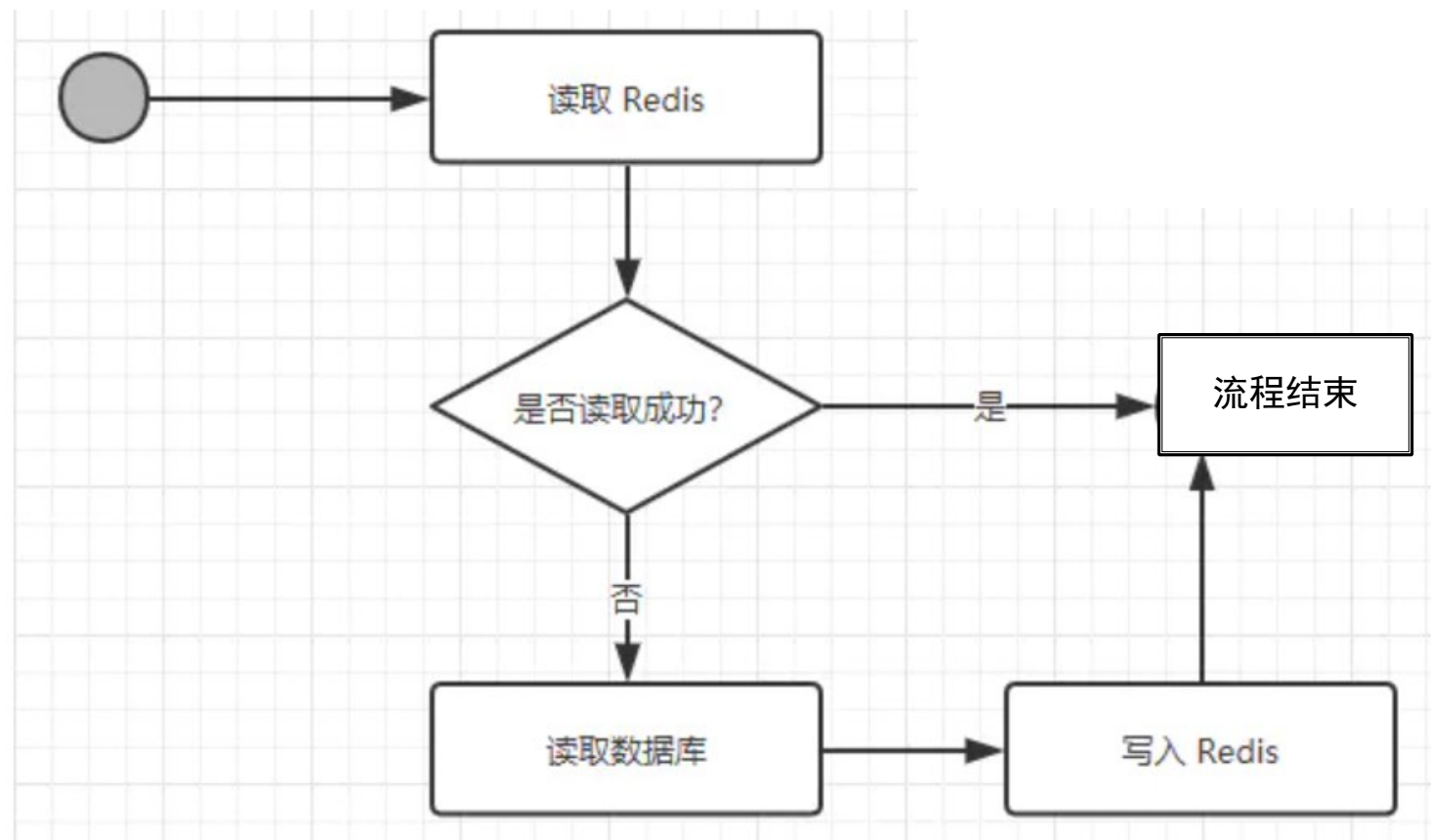
**当第二次以及以后需要读取数据时：**就会直接读取 Redis，读到数据后就结束了流程，这样速度就大大提高

这样做的好处

- ✓ 读操作的可能性是远大于写操作的，所以使用 Redis 来处理日常中需要经常读取的数据，速度提升是显而易见的
- ✓ 同时也降低了对数据库的依赖，使得数据库的压力大大减少

## 1.3.1 Redis

◆ 使用 Redis 作为缓存的读取逻辑如下图所示



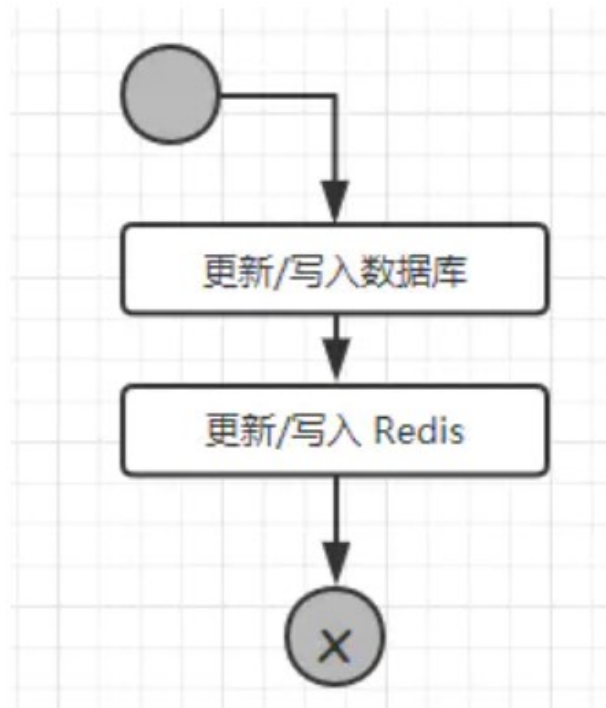


## 1.3.1 Redis

### ◆ Redis写流程

如果业务数据写次数远大于读次数  
就没有必要使用 Redis

谷歌把所有互联网的数据都存储在内存条  
所以才会有如此高质量、高效的搜索  
但它毕竟是谷歌

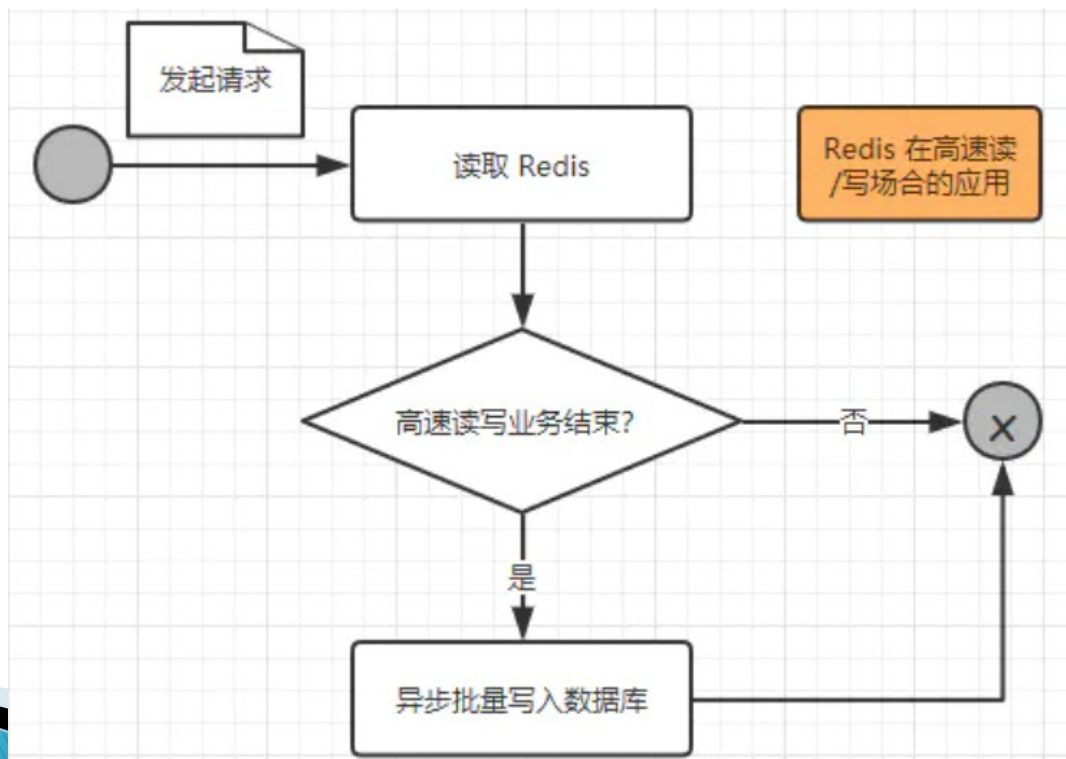


## 1.3.1 Redis

### ◆ Redis应对高速读/写的场合

高并发的情况，比如天猫双11、抢红包、抢演唱会门票等

- 某一个瞬间或者是某一个短暂的时刻有成千上万请求到达服务器



## 1.3.1 Redis

### ◆ Redis应对高速高并发读/写的场合

- 当一个请求到达服务器时，只是把业务数据在 Redis 上进行读写，而没有对数据库进行任何的操作
  - 大大提高读写的速度，从而满足高速响应的需求
- 在一个请求操作完 Redis 的读/写之后，会去判断该高速读/写的业务是否结束，这个判断通常会在秒杀商品为0，红包金额为0时成立，如果不成立，则不会操作数据库；如果成立，则触发事件将 Redis 的缓存的数据以批量的形式一次性写入数据库

## 1.3.1 Redis

### ◆ Redis不仅仅当缓存

支持五大数据类型

集群

持久化

事务

哨兵模式等等

### ◆ 国内外许多大型企业均在使用Redis

twitter、github、美团、搜狐、知乎、新浪微博等

## 1.3.1 Redis

### ◆ Redis学习安排（10课时）

#### 1、通用key操作及五大数据类型的使用

String、List、Set、Sorted Set、Hash

#### 2、Redis两种持久化方式

AOF、RDB

#### 3、Redis集群：主从复制

#### 4、Redis运维、哨兵模式

#### 5、Redis事务

课时所限，Redis还有很多功能，虽无法全部涵盖，也算是入门到深入

## 1.3.2 MongoDB

### ◆ MongoDB是什么

MongoDB并非芒果的意思，而是源于 Humongous（巨大）一词



## 1.3.2 MongoDB

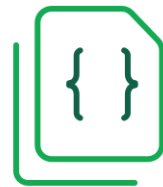
### ◆ MongoDB是什么

- MongoDB是一个介于关系数据库和非关系数据库之间的产品
- 是非关系数据库当中功能最丰富，最像关系数据库的非关系数据库
- MongoDB 是一种文档数据库，它所具备的可扩展性和灵活性可以满足查询和索引的需求

## 1.3.2 MongoDB

- ◆ MongoDB 将数据存储名为BSON的灵活文档中，这意味着字段可能因具体文档而异，并且数据结构可能随着时间的推移而变化

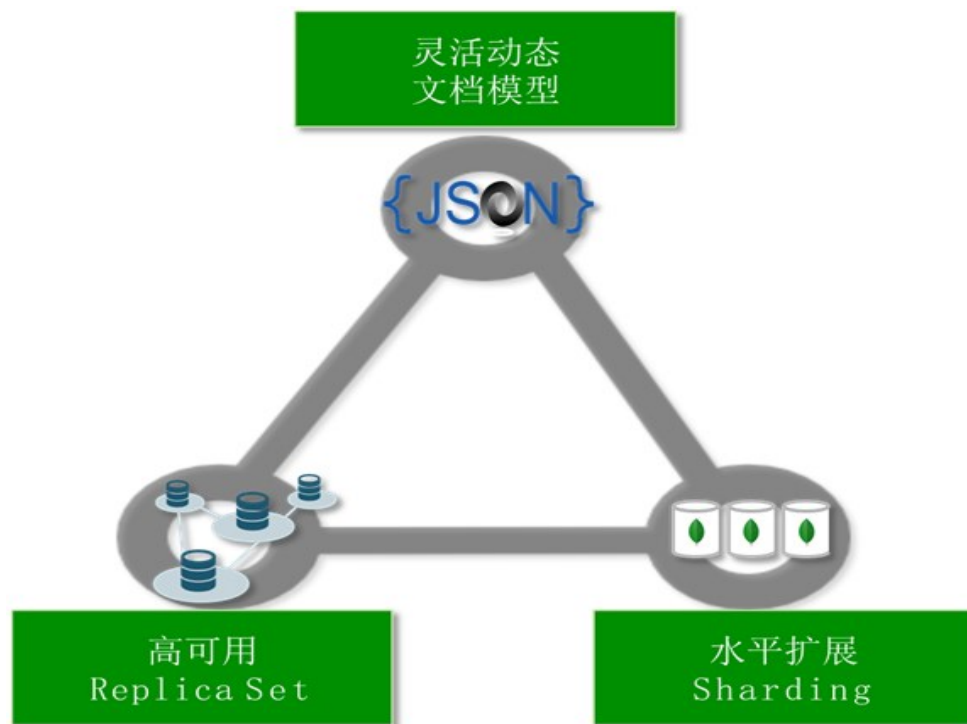
```
1  {  
2    _id: "5cf0029caff5056591b0ce7d",  
3    firstname: 'Jane',  
4    lastname: 'Wu',  
5    address: {  
6      street: '1 Circle Rd',  
7      city: 'Los Angeles',  
8      state: 'CA',  
9      zip: '90404'  
10   }  
11 }
```





## 1.3.2 MongoDB

### ◆ MongoDB的技术特色



## 1.3.2 MongoDB

### ◆ 对于用户而言

- 不断地添加磁盘容量和内存容量是不现实的
- 手工的分库分表又会带来非常繁重的工作量和技术复杂度

### ◆ MongoDB的技术特色

- 自带Mongos集群，只需要在适当的时候继续添加Mongo分片，就可以实现自动水平扩展
  - ✓ 缓解单个节点的读写压力
  - ✓ 有效地均衡磁盘容量的使用情况
  - ✓ 整个mongos集群对应用层完全透明，并可完美地做到各个Mongos集群组件的高可用性

## 1.3.2 MongoDB

### ◆ MongoDB的技术特色

- 即席查询、索引和实时聚合提供了访问数据和分析数据的强大方式
- Licensed under the AGPL，有开源的社区版本
- 起源& 赞助by MongoDB公司，提供商业版licenses 许可

其他：二级索引、动态查询、全文搜索、聚合框架、MapReduce、GridFS、地理位置索引、内存引擎、地理分布等一系列的强大功能。

- 即席查询：在每一个查询操作被执行之前，查询的目标对象是不明确的
- "SELECT \* FROM table WHERE id = " + std\_name

## 1.3.2 MongoDB

### ◆ 关系型数据库与MongoDB对比

存储方式是以表的形式存放，而在MongoDB中，以文档的形式存在

#### 关系型

PERSON

Pers_ID	Surname	First_Name	City
0	Miller	Paul	London
1	Ortega	Alvaro	Valencia
2	Huber	Urs	Zurich
3	Blanc	Gaston	Paris
4	Bertolini	Fabrizio	Rome

CAR

Car_ID	Model	Year	Value	Pers_ID
101	Bentley	1973	100000	0
102	Rolls Royce	1965	330000	0
103	Peugeot	1993	500	3
104	Ferrari	2005	150000	4
105	Renault	1998	2000	3
106	Renault	2001	7000	3
107	Smart	1999	2000	2

NO RELATION

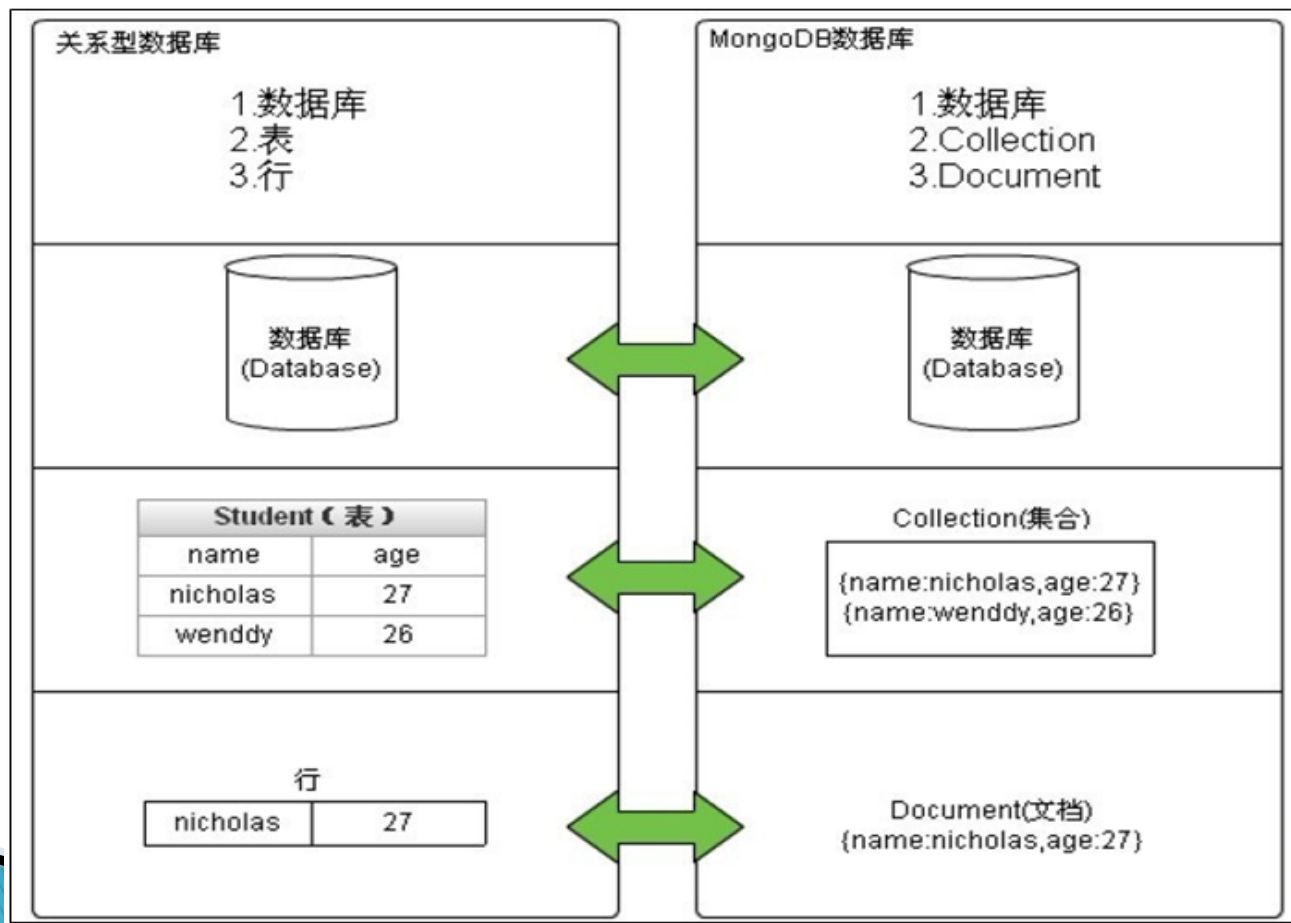
#### MongoDB

```
{
  first_name: 'Paul',
  surname: 'Miller',
  city: 'London',
  location:
    [45.123, 47.232],
  cars: [
    { model: 'Bentley',
      year: 1973,
      value: 100000, ... },
    { model: 'Rolls Royce',
      year: 1965,
      value: 330000, ... }
  ]
}
```

## 1.3.2 MongoDB

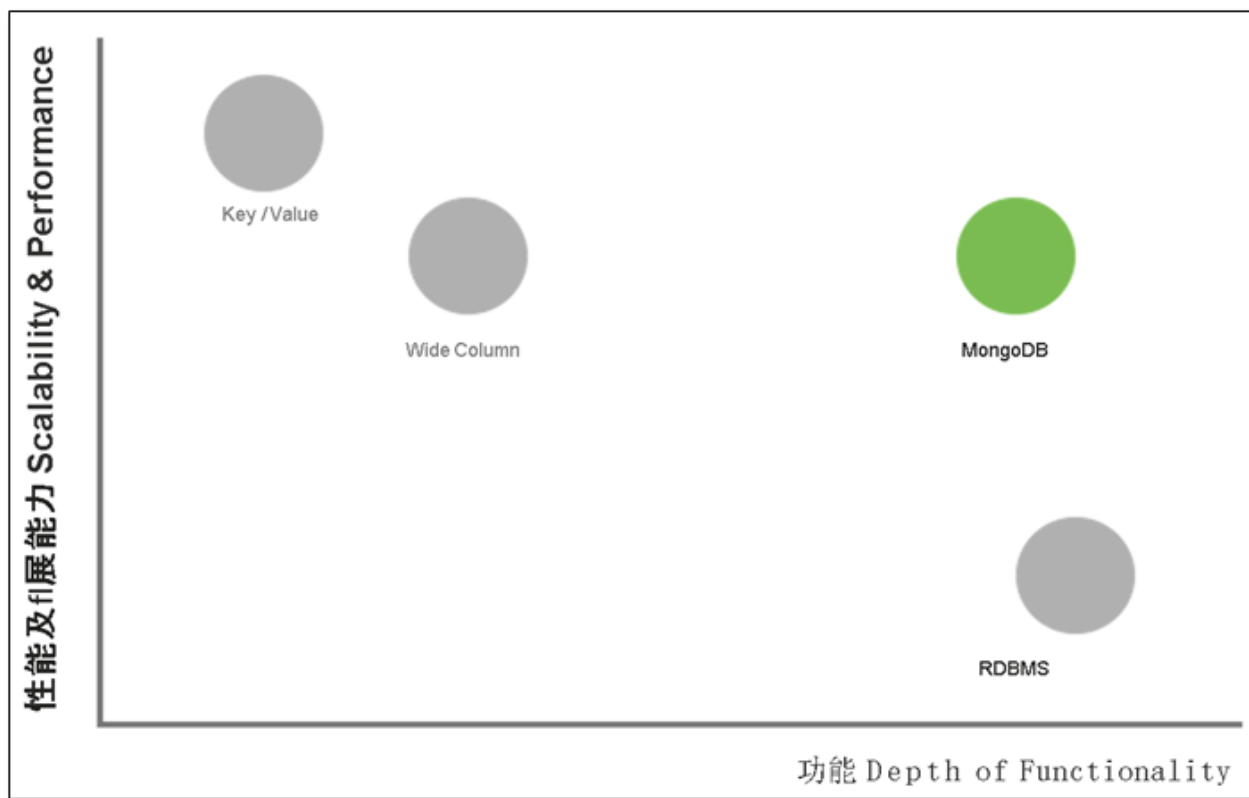
### ◆ 关系型数据库与mongodb对比

数据库中的对应关系，及存储形式的说明



## 1.3.2 MongoDB

### ◆ 性能对比



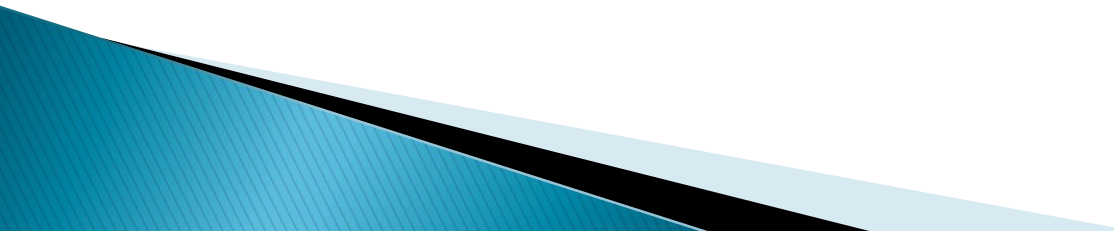
MongoDB数据库的性能扩展能力及功能都较好，都能够在数据库中，站立一席之地

## MongoDB与Redis的对比

指标	MongoDB(v2.4.9)	Redis(v2.4.17)	比较说明
实现语言	C++	C/C++	--
协议	BSON、自定义二进制	类Telnet	--
性能	依赖内存，吞吐量较高	依赖内存，吞吐量非常高	Redis优于MongoDB
可操作性	丰富的数据表达、索引；最类似于关系数据库，支持丰富的查询语言	数据丰富，简单的查询，查询依赖于用空间换取查询效率	MongoDB优于Redis
内存及存储	适合大数据量存储，依赖系统虚拟内存管理，采用镜像文件存储，内存占有率较高，官方建议独立部署在64位系统（32位有最大2.5G文件限制，64位没有）	Redis2.0后增加虚拟内存特性，突破物理内存限制，数据可以设置时效性，类似于memcache	--
一致性	不支持事务，靠客户端自身保证	支持部分事务	Redis优于MongoDB
数据分析	内置一定的数据分析功能（MapReduce）	不支持	MongoDB优于Redis

## 1.3.2 MongoDB

### ◆ MongoDB的学习（3课时）

- 1、 MongoDB起源、简介、特点、适用场景
  - 2、 文档、集合、数据库概念与注意事项
  - 3、 MongoDB的部署与常用交互式命令
  - 4、 MongoDB数据库的操作、集合操作
  - 5、 MongoDB文档的增、删、改、查命令
  - 6、 使用Java程序访问MongoDB
- 



## 1.3.3 数据仓库

### ◆ 应用需求

很多人知道数据库，但不知道数据仓库

- 1、如果你要的数据分别存放在很多个不同的数据库，甚至存在文本文件，excel中，你要如何获取这些数据？
- 2、如果你从这些数据源中取出了你要的数据，但是发现格式不一样，或者数据类型不一样，你要怎么规范？
- 3、如果你是一个只会SQL查询的人，你想从复杂的海量数据中分析查询数据，应该怎么办？
- 4、如果你有一个关于人口的数据，你想知道“某个省份学历分布情况”，要怎么快速高效地得知呢？

为了解决上面几个问题，数据仓库就诞生了

## 1.3.3 数据仓库

### ◆ 数据仓库是什么

官方解释：

- 数据仓库，英文名称为Data Warehouse，可简写为DW或DWH
- 数据仓库（Data Warehouse）是一个面向主题的（Subject Oriented）、集成的（Integrated）、相对稳定的（Non-Volatile）、反映历史变化（Time Variant）的数据集合，用于支持管理决策(Decision Making Support)的数据集合

## 1.3.3 数据仓库

### ◆ 数据仓库是什么

- 从逻辑上理解，数据库和数据仓库没有区别，都是通过数据库软件实现存放数据
- 从数据量来说，数据仓库要比数据库更庞大得多
- 数据仓库里的数据通常不需要改动，但是会一定时间批量更新
- 数据仓库主要用于数据挖掘和数据分析，辅助做决策

## 1.3.3 数据仓库

### ◆ 数据仓库的用途

- 数据仓库是企业所有级别的决策制定过程，提供**所有类型数据**支持的战略集合
- 数据仓库是单个数据存储，出于**分析性报告和决策支持**目的而创建
- 数据仓库为需要业务智能的企业，提供指导业务流程改进、监视时间、成本、质量以及控制

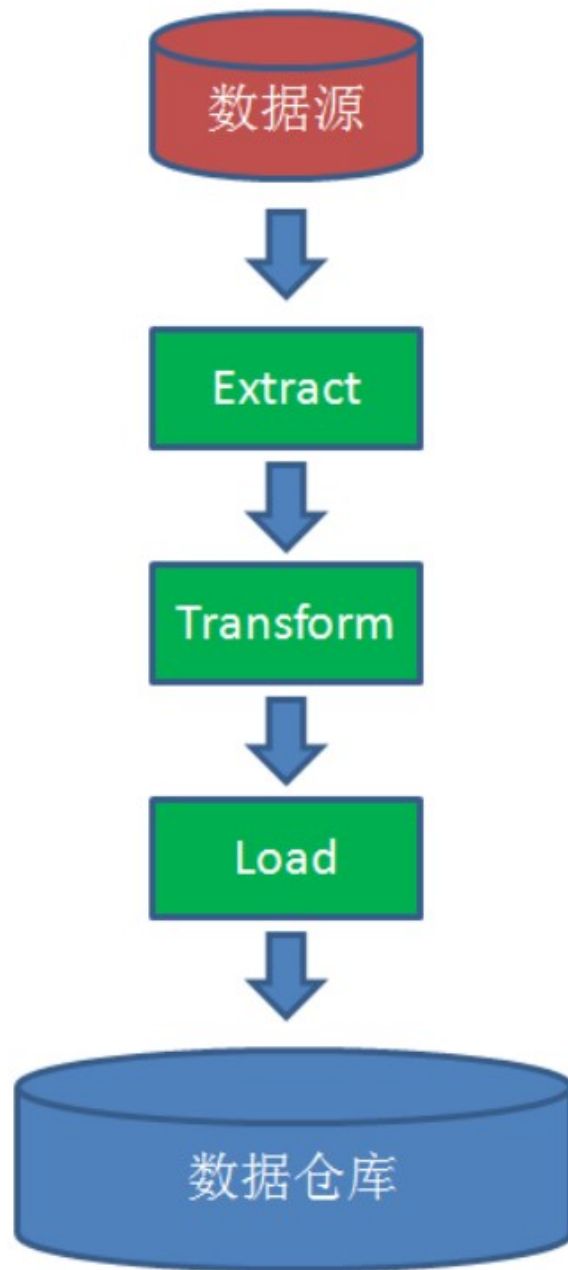
数据仓库具有改变业务的威力，它能帮助公司深入了解客户行为，预测销售趋势，确定某一组客户或产品的收益率

## 1.3.3 数据仓库

### ◆ 什么是ETL?

ETL (Extract-Transform-Load) 描述将数据从来源迁移到目标的过程

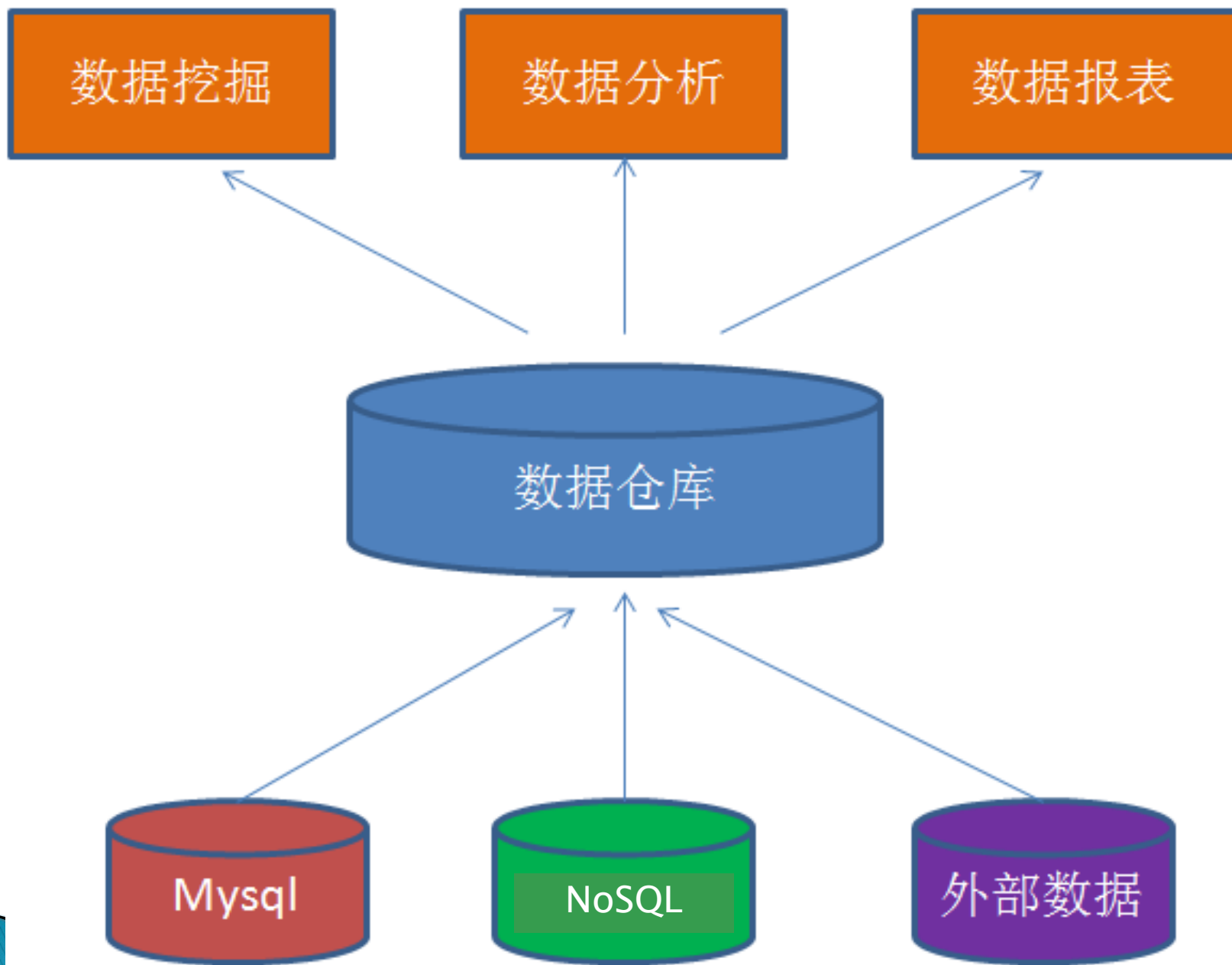
- **Extract**, 数据抽取, 也就是把数据从数据源读出来
- **Transform**, 数据转换, 把原始数据转换成期望的格式和维度
- **Load** 数据加载, 把处理后的数据加载到目标处



## 1.3.3 数据仓库

### ◆ 构建数据仓库的过程

- 将不同数据源的数据整合起来，通过对数据进行清洗，规范化数据等步骤，根据需求围绕一个主题进行构建
- 构建好的数据仓库不用于UPDATE，用于查询、数据分析、数据挖掘等





## 1.3.3 数据仓库

### ◆ 数据仓库与数据挖掘（3课时）

#### 1、数据仓库简介

- 数据仓库的产生
- 数据仓库的定义
- 数据仓库的特征

#### 2、数据仓库的架构及相关概念

- ETL
- 元数据（MetaData）
- 数据集市（Data Market）

#### 3、数据库与数据仓库的联系与区别

#### 4、数据仓库与数据挖掘的联系

## 1.3.4 数据仓库Hive

### ◆ Hive的起源

在很久很久以前

有一个叫facebook的公司，内部搭建了数据仓库（理解成把一大堆数据放到一个地方，用来分析以决定决策）是基于mysql

后来随着数据量的不断增加，这种传统的数据库扛不住了

Hadoop出现了，于是经过一系列的折腾换到了hadoop上

## 1.3.4 数据仓库Hive

### ◆ 问题来了

- 以前基于数据库的数据仓库用SQL就能做查询，现在换到HDFS上面，得跑MapReduce任务去做分析
- 以前做分析的人还得学MapReduce，并且去写大量的MapReduce代码，花时间费力气

## 1.3.4 数据仓库Hive

### ◆ 问题来了

- 1、 Hadoop是个好工具，但是学习难度大，人员成本太高
- 2、 项目周期要求太短
- 3、 MapReduce 实现复杂查询逻辑开发难度太大

### ◆ 解决问题

Facebook开发了一套用SQL语句来做HDFS的分析查询的工具

- 用户输入的是SQL，通过这个工具把SQL转成MapReduce的任务，然后再去执行分析

## 1.3.4 数据仓库Hive

### ◆ Hive是什么

- ✓ Hive是由Facebook开源用于解决海量结构化日志的数据统计
- ✓ Hive是一个成功的Apache项目，很多组织把它用作一个通用的、可伸缩的数据处理平台
- ✓ Hive是一个基于Hadoop的工具，可用来对数据进行提取/转化/加载(ETL)
- ✓ Hive提供了一种可以存储、查询和分析存储在HDFS中的大规模数据的机制

## 1.3.4 数据仓库Hive

### ◆ HQL语言

Hive定义了一种类似SQL的查询语言，被称为HQL

- ✓ 对于熟悉SQL的用户可以直接利用Hive来查询数据
- ✓ 这个语言也允许熟悉 MapReduce 开发者们开发自定义的mappers和reducers来处理内建的mappers和reducers无法完成的复杂的分析工作

## 1.3.4 数据仓库Hive

### ◆ Hive的作用

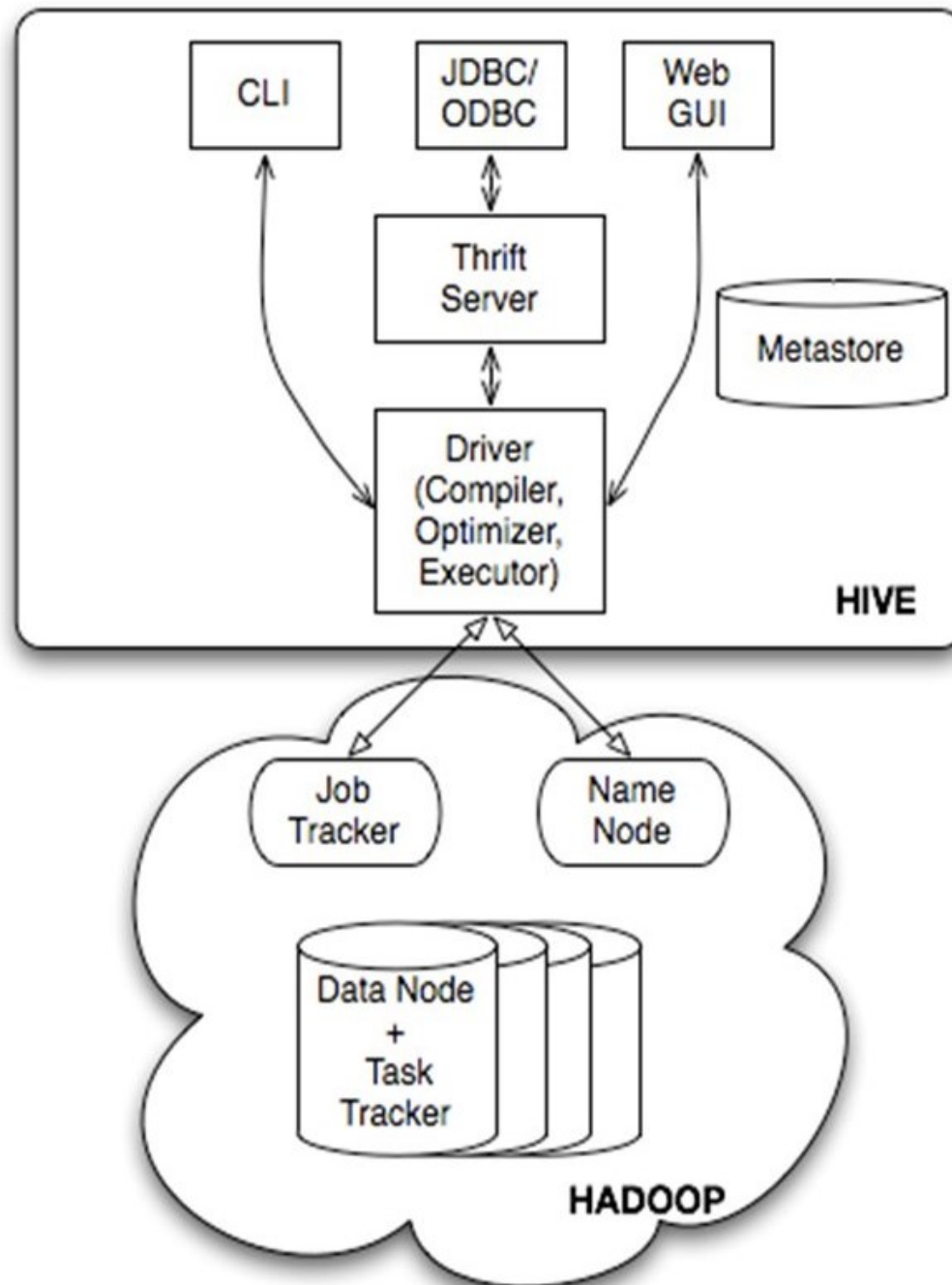
- 1.把SQL语句转化成Map Reduce代码
- 2.可以对数据进行存储，使用 HDFS
- 3.可以对数据使用MapReduce进行转化、计算、分析

### ◆ Hive的意义

降低程序员使用hadoop的难度

降低学习成本







## 1.3.4 数据仓库Hive

### ◆ Hive工作流程

- 1、Hive通过给用户提供的系列交互接口，接收到用户的指令(HQL)
  - 支持各种命令，比如dfs的命令、脚本的执行
- 2、接收指令后，会交给Driver的组件，结合元数据(MetaStore)，编译解析成MapReduce
- 3、把编译出来的结果交给hadoop去执行，将执行返回的结果输出到用户交互接口

可以将Hive理解成搭建在Hadoop(HDFS和Map Reduce)之上的语言壳子

## 1.3.4 数据仓库Hive

### ◆ Hive的应用

- Hive可以使用HQL(Hive SQL)很方便的完成对海量数据的统计汇总
- Hive使用Hadoop作为执行引擎，有批处理，高延迟、高可扩展性和高容错性的特点
  - 在数据量很小的时候，Hive执行也需要消耗较长时间来完成，这时候，就显示不出它与Oracle，Mysql等传统数据库的优势。
- Hive擅长非实时的、离线的、对响应及时性要求不高的海量数据批量计算，网络日志分析等

## 1.3.4 数据仓库Hive

### ◆ Hive的学习（10课时）

#### 1、Hive简介

- Hive的起源、特性、架构、Hive的运行机制
- Hive的应用场景

#### 2、SQL（HQL）转换成MapReduce的工作原理

#### 3、Hive的安装

#### 4、Hive常用属性配置、交互指令以及数据类型简介

#### 5、Hive的基本操作

- Hive中数据库的操作
- Hive中表的基本操作

#### 6、HQL的基本使用

## 1.3.4 数据仓库Hive

### ◆ Hive的学习（10课时）

#### 7、Hive中数据导入与导出

- 数据导入的几种方法介绍
- 数据导出的几种方法介绍
- 清空表中的数据

#### 8、Hive内部表与外部表

- 创建管理表和外部表
- 管理表和外部表的区别
- 适用场景

#### 9、Hive分区表

- 分区表介绍
- 分区表的创建及使用

#### 10、Hive窗口函数

- 窗口函数的介绍
- 窗口函数的使用

目前很多大数据开发的公司面试时常问到Hive的相关问题，说明Hive在数据仓库占据比较重要的地位

## 1.3.5 云数据库

### ◆ 主流的数据库的构建方式

#### ➤ 线下自己构建数据库

MySQL、DB2、ACCESS或者Oracle等

- MySQL是免费的，其他的像IBM的DB2，Microsoft的ACCESS还有Oracle的数据库，下载镜像，之后在自己的购买的服务器上安装之后就可以使用

#### ➤ 使用在线的云数据库

## 1.3.5 云数据库

### ◆ 线下构建数据库遇到的问题

商业用途的话就需要购买许可（向数据库公司付服务费用）

数据库构建完以后需要专门的运维人员

保证数据中心不能断电（各种不可预料因素），还要做好备份，这些工作往往是非常消耗人力和物力

安全性问题

- 外部的破坏者DDOS或者暴力破解或者SQL注入等
- 被自己的运维人员删掉了（有删库到跑路）

对于开发者或者初创公司而言会有很多的不便之处

## 1.3.5 云数据库

### ◆ 云数据库应运而生

- ✓ 云计算服务提供商，提供的云数据库服务
- ✓ 看不到运行数据库的实体主机
- ✓ 随时访问自己的云数据库中的数据并使用
- ✓ 不需要担心数据的安全性（如果你能信赖云数据库服务商）云数据库服务商往往会提供冗余算法保障数据的安全（但无法保证我们自己操作问题导致安全性问题）
- ✓ 节省了用于数据库运维的大量的人力和物力

## 1.3.5 云数据库

### ◆ 云数据库

- 云数据库是指被优化或部署到一个虚拟计算环境中的数据库，可以实现按需付费、按需扩展、高可用性以及存储整合等优势
- 根据数据库类型一般分为
  - 关系型数据库
  - 非关系型数据库（NoSQL数据库）



## 1.3.5 云数据库

### ◆ 云数据库提供的服务

- ✓ 用户按照存储容量和带宽的需求付费
- ✓ 可以将数据库从一个地方移到另一个地方（云的可移植性）
- ✓ 可以按需扩展
- ✓ 其他：实例创建快速、支持只读实例、读写分离、故障自动切换、数据备份、Binlog备份、SQL审计、访问白名单、监控与消息通知等

## 1.3.5 云数据库

### ◆ 国内的云数据库服务提供商

阿里、腾讯、网易、华为等

### ◆ 国外的云数据库服务提供商

Amazon、Microsoft Azure、谷歌云等

## 1.3.5 云数据库

### ◆ 阿里云数据库的优点

- 3层安全防护体系，通过十项安全合规认证，能抵御90%以上的网络攻击
- 3重高可用（容灾）架构，提供99.95%的业务可用性保障
- 弹性扩展，实现100%资源利用率
- 内网外网同时连接，方便本地化管理
- 自动备份，两年内数据恢复，解决90%以上的系统故障
- 自动监控预警，定期性能巡检，可以分担60%以上的运维工作

## 1.3.5 云数据库

### ◆ 云数据库（4课时）

#### 1、云数据库简介

- 云数据库概念
- 云数据库的特点
- 云数据库产品

#### 2、云数据库系统架构

#### 3、Amazon AWS和云数据库

#### 4、微软云数据库SQL Azure

#### 5、以阿里云RDS为实例介绍云数据库操作实践

谢谢大家！