

# 教学内容如何跟上数据库 系统的演进

周烜

华东师范大学 数据科学与工程学院



**DaSE**  
Data Science  
& Engineering

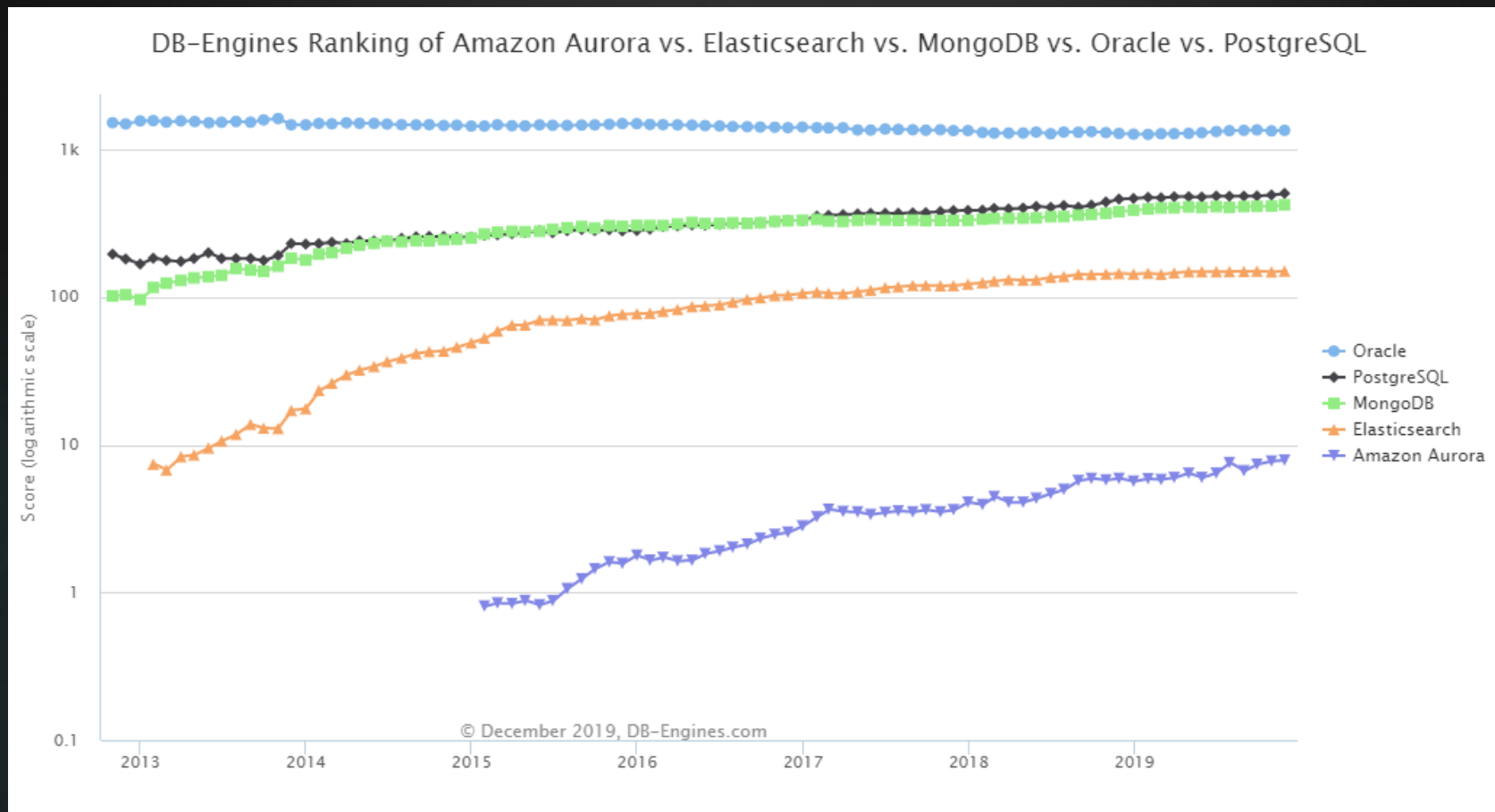
# 系统使用方法 vs 系统工作原理

- ▶ 绝大部分学生未来只是系统的使用者，而非开发者。
- ▶ 了解工作原理是为了更好的使用系统。

# 数据库系统的形态已经发生了变化

SQL & NOSQL & NewSQL DB

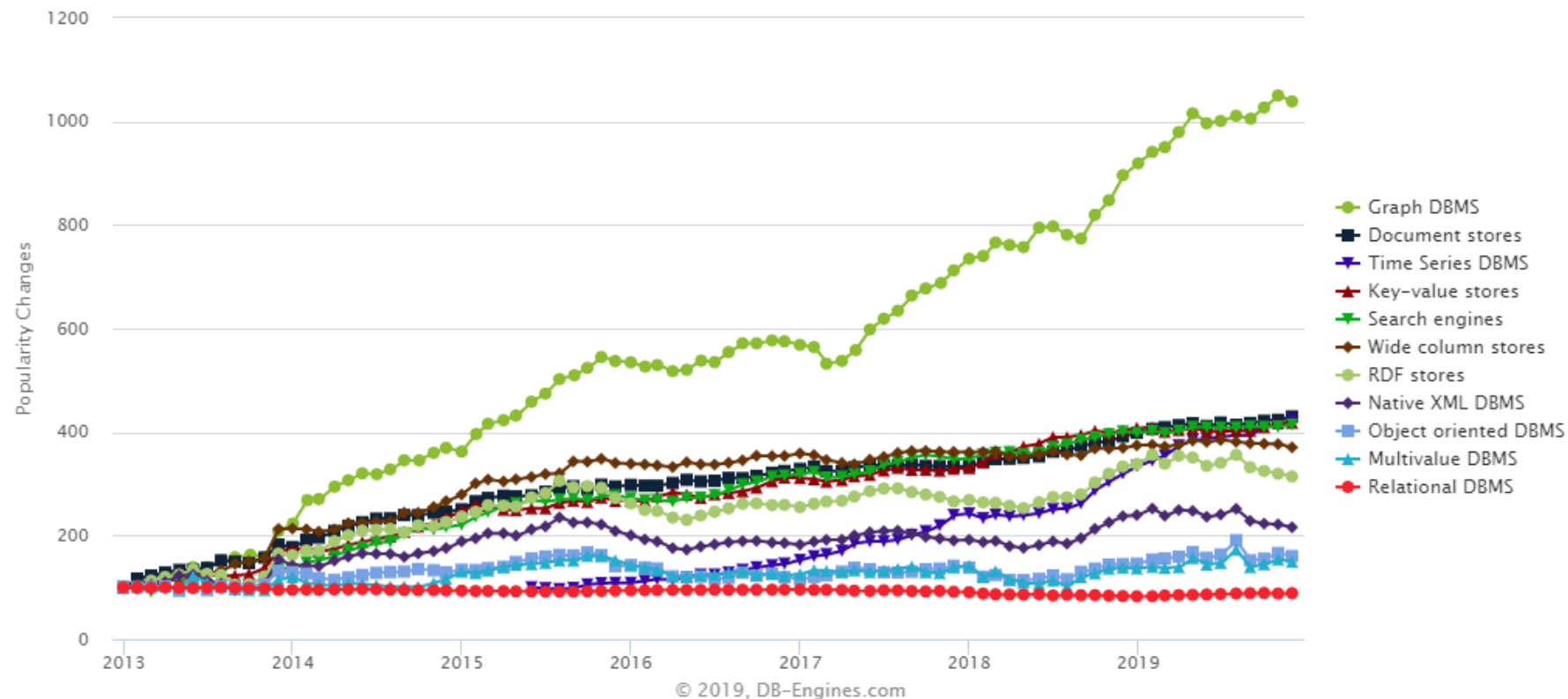
# 数据库系统的形态已经发生了变化



典型数据库系统的流行趋势

# 数据库系统的形态已经发生了变化

Complete trend, starting with January 2013



各类数据库系统的流行趋势



“One Size does not Fit All.”  
– Michael Stonebraker

# 教学内容如何安排？

系统设计者的想法



SQL & NOSQL & Scalable DB



如何在众多系统中做出正确选择

# What is wrong with SQL?

- ▶ Not Scalable
- ▶ Not Usable (in some cases)



# Uncle Bob: No DB



“As the lead developer, responsible for the quality of the software, my view of a relational database was that it would be a big, stogy, slow, expensive pain in the rear. We didn’t have complex queries. We didn’t need massive reporting capabilities. We certainly didn’t need a process with a multi-megabyte footprint sitting in memory and burning cycles. (Remember, this was the ‘80s). So I fought against this idea with everything I had; because it was the wrong technical solution.”

“The center of your application is not the database. Nor is it one or more of the frameworks you may be using. The center of your application are the use cases of your application.”

# SQL is not intuitive.

```
{  
  name: Bob  
  dateofbirth: 12/3/1954  
  gender: male  
  hobby: [basketball, reading,  
         guitar]  
}
```

uid	name	dateofbirth	gender
001	Bob	12/3/1954	male

uid	hobby
001	basketball
001	reading
001	guitar

# Merits of SQL?

- ▶ Expressive Query Language (Declarative)
  - ▶ Decoupling App and DB
- ▶ De facto Standard

# Comparison

## ▶ SQL

- ▶ Powerful functionality
- ▶ Clean Schema
- ▶ Good for Analytics

## ▶ Document DB (MongoDB)

- ▶ More control to APP
- ▶ Intuitive
- ▶ Object Oriented

# 课程大纲（数据管理系统）

知识点	学时
数据管理系统的概念	4
文档数据库功能和原理	6
文档数据库设计	4
关系数据库功能和原理	10
关系数据库设计	8
事务处理	10
数据分析系统	4
系统比较	2
数据管理系统的扩展	8~10*

## ▶ 传统课程

知识点	学时
数据库系统的概念	2
关系数据库功能和原理	18
关系数据库设计	12
事务处理	10
数据仓库	6

# 课程大纲（数据管理系统）

- ▶ 削减关系数据库的内容
  - ▶ SQL的部分细节
  - ▶ 范式
  - ▶ 部分功能的介绍：游标、触发器、存储过程、访问控制
- ▶ 增强的内容
  - ▶ 关系数据库的历史和设计思想

# 课程大纲展开

- ▶ 数据管理系统的概念
  - ▶ 系统和系统的性质 2
  - ▶ 数据管理系统 1
  - ▶ 数据模型 1
- ▶ 文档数据库功能和原理
  - ▶ 文档模型 1
  - ▶ CRUD操作 2
  - ▶ 存储与索引 3
- ▶ 文档数据库设计
  - ▶ 文档数据库设计实例 2
  - ▶ 文档数据库设计基本原则 2
- ▶ 关系数据库功能和原理
  - ▶ 关系数据库的历史和设计思想 2
  - ▶ 关系模型 2
  - ▶ SQL 5
  - ▶ 关系的存储 1
- ▶ 关系数据库设计
  - ▶ ER图 4
  - ▶ 范式 2
  - ▶ 物理设计 2
- ▶ 事务处理
  - ▶ 操作原子性/日志/锁 2
  - ▶ 高可用协议 2
  - ▶ 无事务的数据一致性 1
  - ▶ 事务功能 1
  - ▶ ACID的实现原理 2
  - ▶ 事务使用的基本原则 2
- ▶ 数据分析系统
  - ▶ 分析应用简介 2
  - ▶ 分析平台简介 2
- ▶ 系统比较 2

# 什么引起了数据库系统的形态变化？

- ▶ 应用需求的变化
- ▶ 软件开发模式的转变
- ▶ 硬件平台的革新



# 应用需求的变化

## ▶ 应用规模的扩展



前互联网时代



互联网时代



万物互联时代

## ▶ 应用领域的拓展

▶ 金融、销售、物流、传媒、医疗、教育

# 应用需求变化带来的影响

- ▶ 负载增加→对扩展能力的需求

- ▶ 分布式数据库成为主流：Spanner、Oceanbase、Impala、TiDB、...

- ▶ 架构的变化

- ▶ 折中点的重新考量：

易用性

功能性

通用性



功能性

扩展性

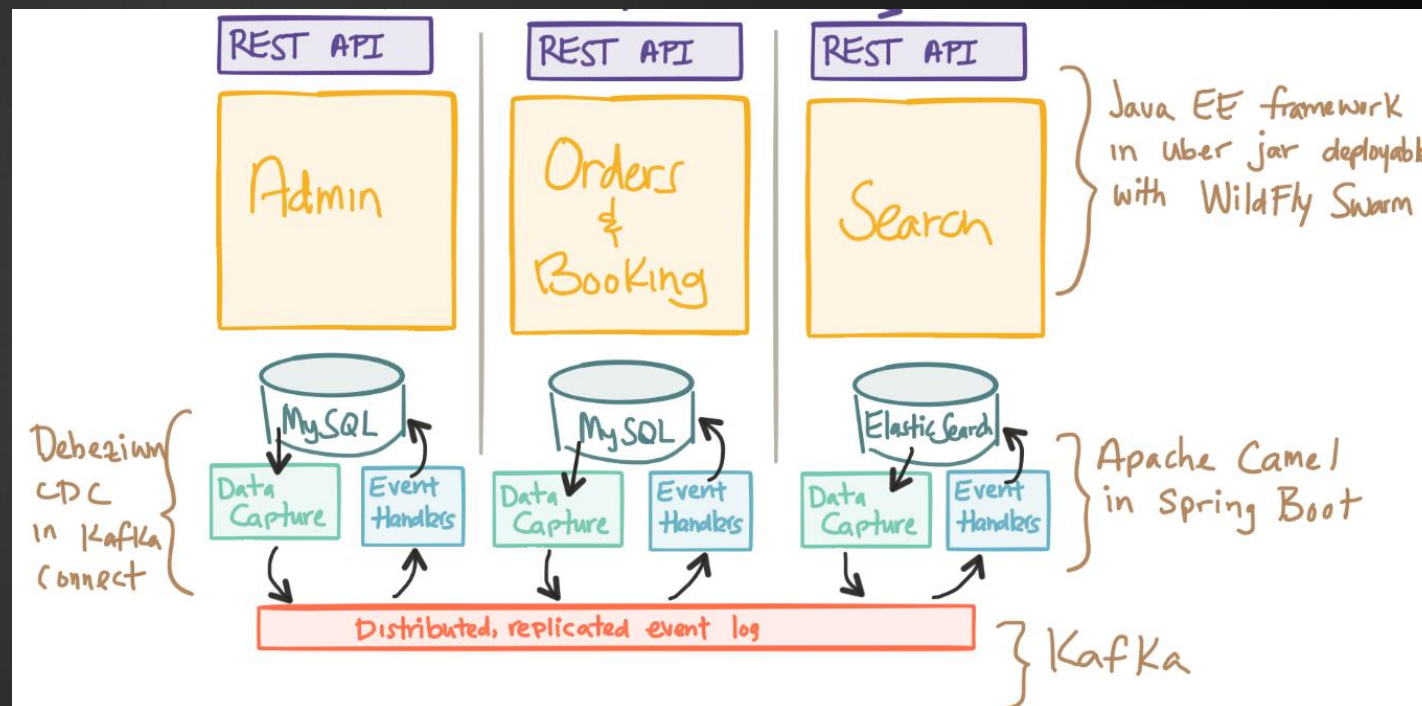
适配性

- ▶ 多样化：SQL、NoSQL、Search Engine、Hadoop

- ▶ "One Size does not Fit All." – Michael Stonebraker

# 软件开发模式的转变

- ▶ 敏捷开发
- ▶ 微服务
- ▶ 事件驱动架构
- ▶ 框架的广泛应用



流行的微服务架构

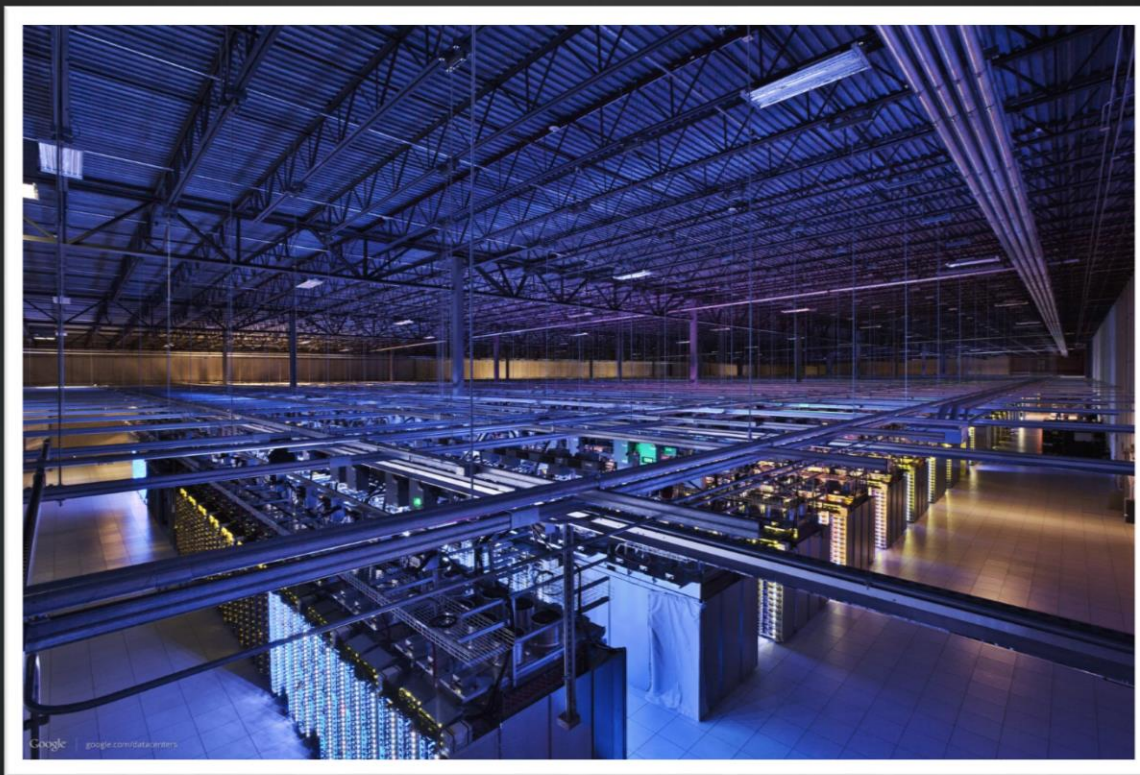
# 开发模式变化带来的影响

- ▶ 同一个软件，多种数据库产品并存
  - ▶ 不同微服务使用不同的数据库产品
- ▶ 程序员和DBA的界限模糊化
  - ▶ 应用设计和数据库设计的界限被打破
- ▶ 事务处理模式的变化
  - ▶ 消息队列和事件驱动架构替代传统事务处理



# 硬件平台的革新

- ▶ 云计算
- ▶ 新硬件



GPU



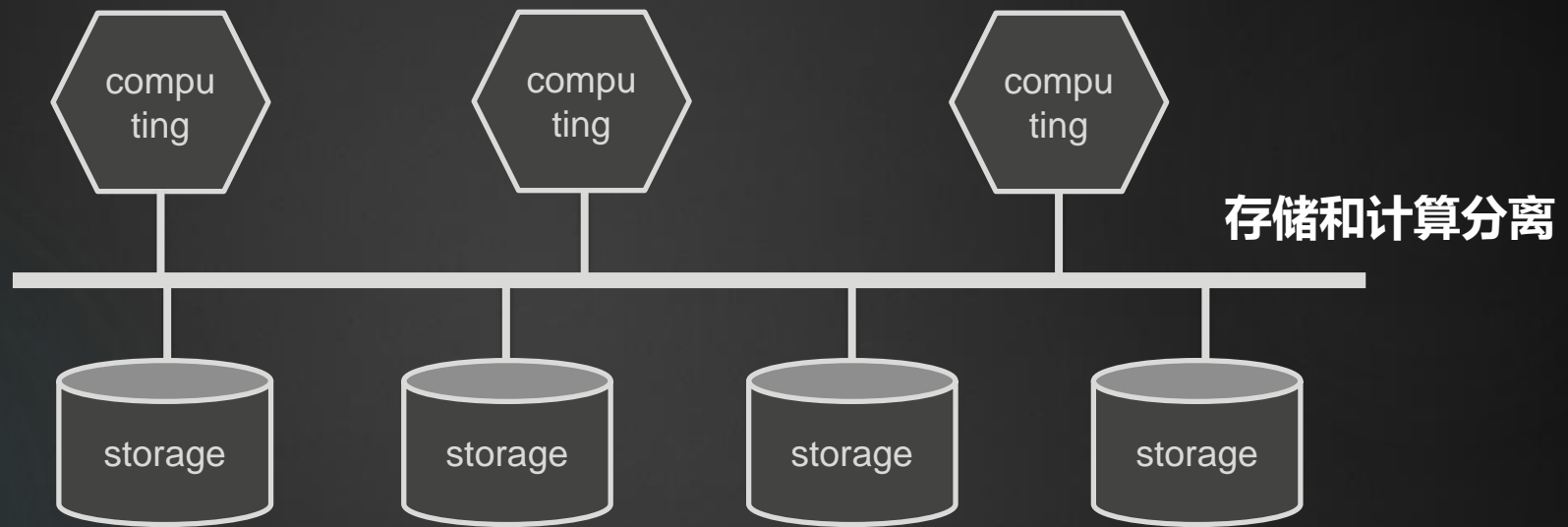
TPU



NVM

# 硬件平台革新带来的影响

- ▶ 云平台上的数据库 → 弹性



- ▶ 软硬件结合优化

$$\text{data/cost} = \text{data/hardware} \times \text{hardware/cost}$$

# 未来发展趋势

- ▶ One Size Fits a Bunch
  - ▶ 为不同应用构建不同系统，为不同系统配置不同硬件
- ▶ 功能分解，多系统协作中间件，多模数据库
  - ▶ SQL + NoSQL + Cache + MQ + DW + ...
- ▶ 云化 (DBaaS)
  - ▶ Serverless, Auto-scaling, AI+DB

谢谢！

