



厦门大学研究生课程 《大数据处理技术Spark》

<http://dblab.xmu.edu.cn/post/7659/>

温馨提示：编辑幻灯片母版，可以修改每页PPT的厦大校徽和底部文字

第1章 大数据技术概述

(PPT版本号：2017年春季学期)

林子雨

厦门大学计算机科学系

E-mail: ziyulin@xmu.edu.cn ▶▶

主页: <http://www.cs.xmu.edu.cn/linziyu>



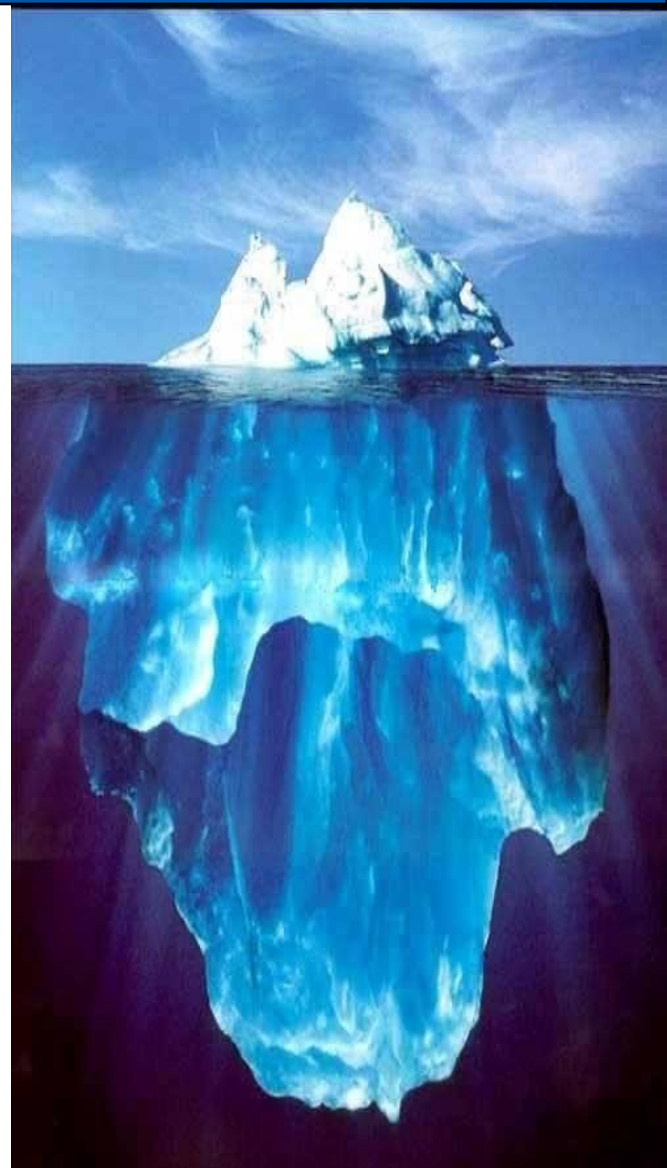
扫一扫访问班级主页





提纲

- 1.1 大数据时代
- 1.2 大数据概念
- 1.3 大数据的影响
- 1.4 大数据关键技术
- 1.5 大数据计算模式
- 1.6 代表性大数据技术





1.1.1 第三次信息化浪潮

- 根据IBM前首席执行官郭士纳的观点，IT领域每隔十五年就会迎来一次重大变革

表1 三次信息化浪潮

信息化浪潮	发生时间	标志	解决问题	代表企业
第一次浪潮	1980年前后	个人计算机	信息处理	Intel、AMD、IBM、苹果、微软、联想、戴尔、惠普等
第二次浪潮	1995年前后	互联网	信息传输	雅虎、谷歌、阿里巴巴、百度、腾讯等
第三次浪潮	2010年前后	物联网、云计算和大数据	信息爆炸	将涌现出一批新的市场标杆企业



1.1.2 信息科技为大数据时代提供技术支撑

1. 存储设备容量不断增加

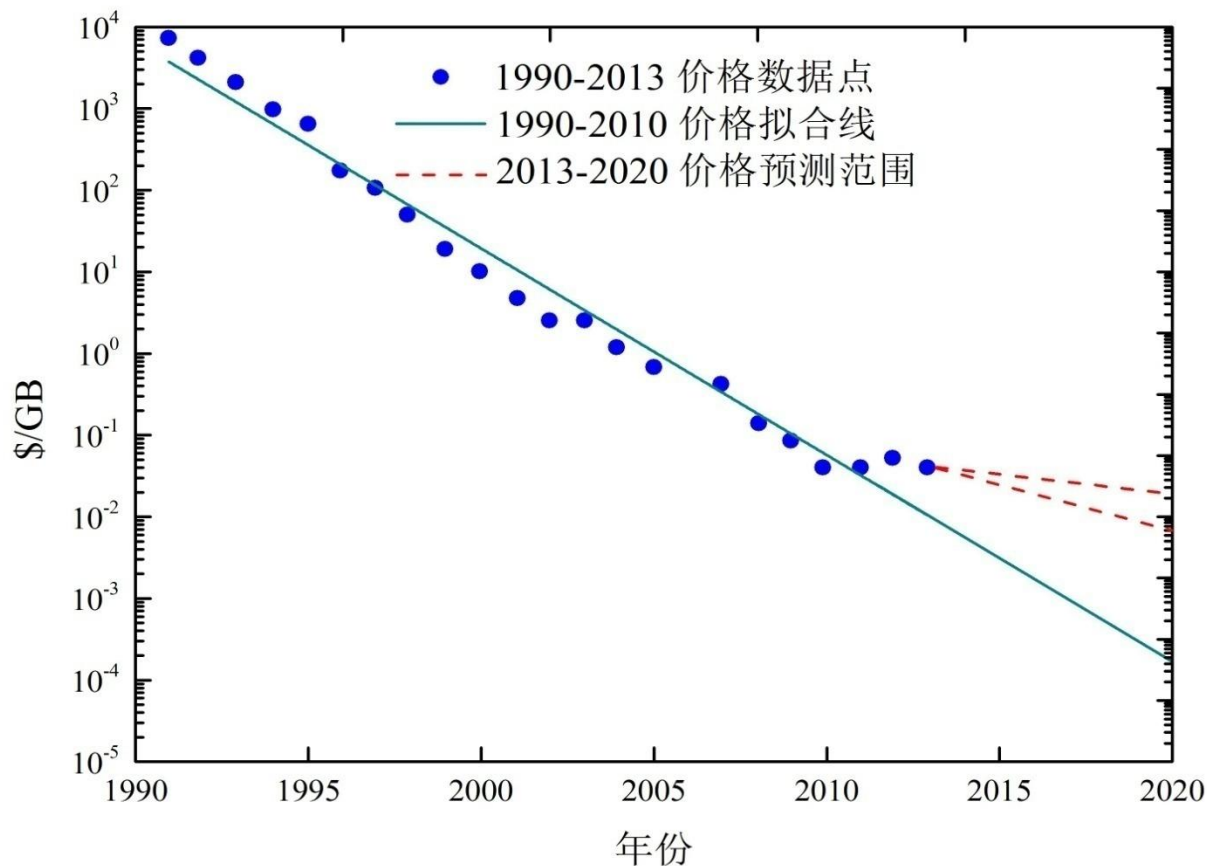


图1-1 存储价格随时间变化情况



1.1.2 信息科技为大数据时代提供技术支撑



来自斯威本科技大学（Swinburne University of Technology）的研究团队，在2013年6月29日刊出的《自然通讯（Nature Communications）》杂志的文章中，描述了一种全新的数据存储方式，可将1PB（1024TB）的数据存储到一张仅DVD大小的聚合物碟片上。



1.1.2 信息科技为大数据时代提供技术支撑

2. CPU处理能力大幅提升

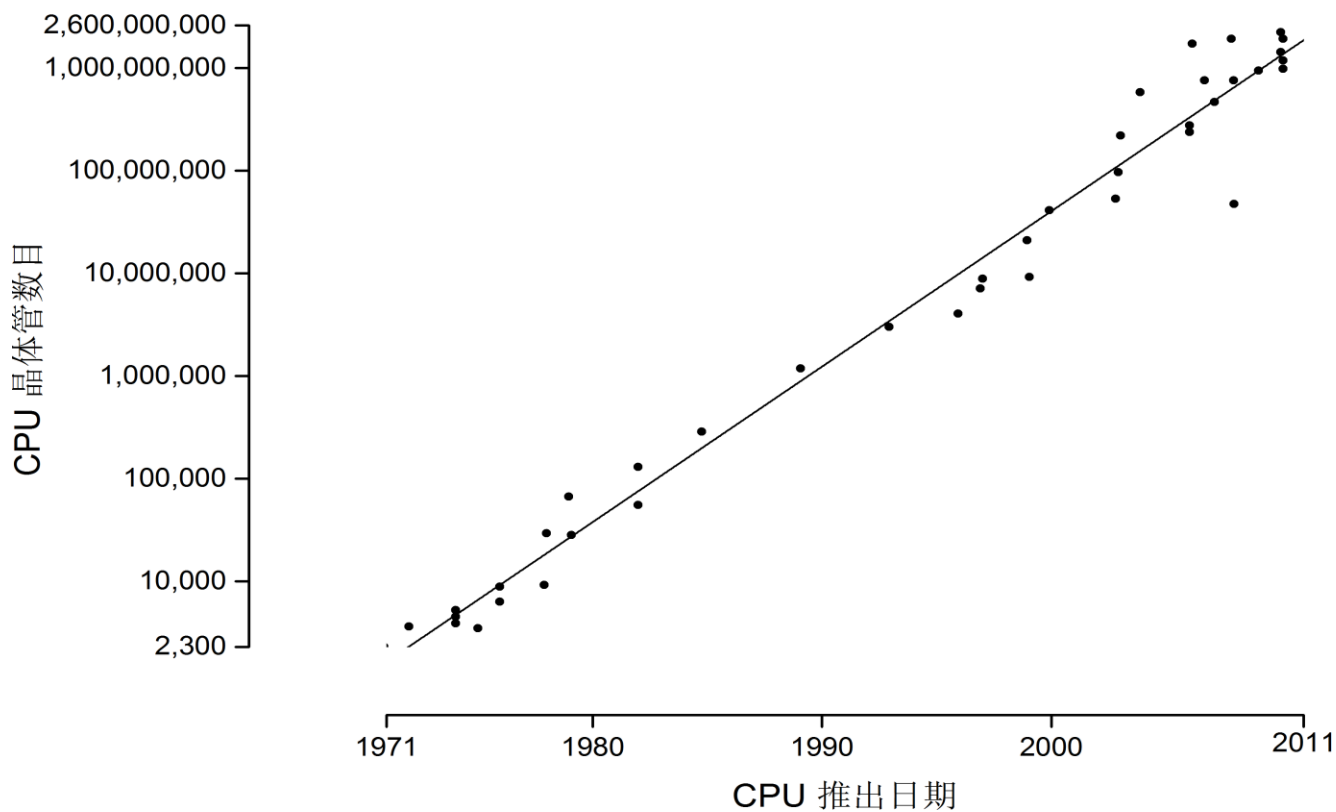


图 CPU晶体管数目随时间变化情况



1.1.2 信息科技为大数据时代提供技术支撑

3. 网络带宽不断增加

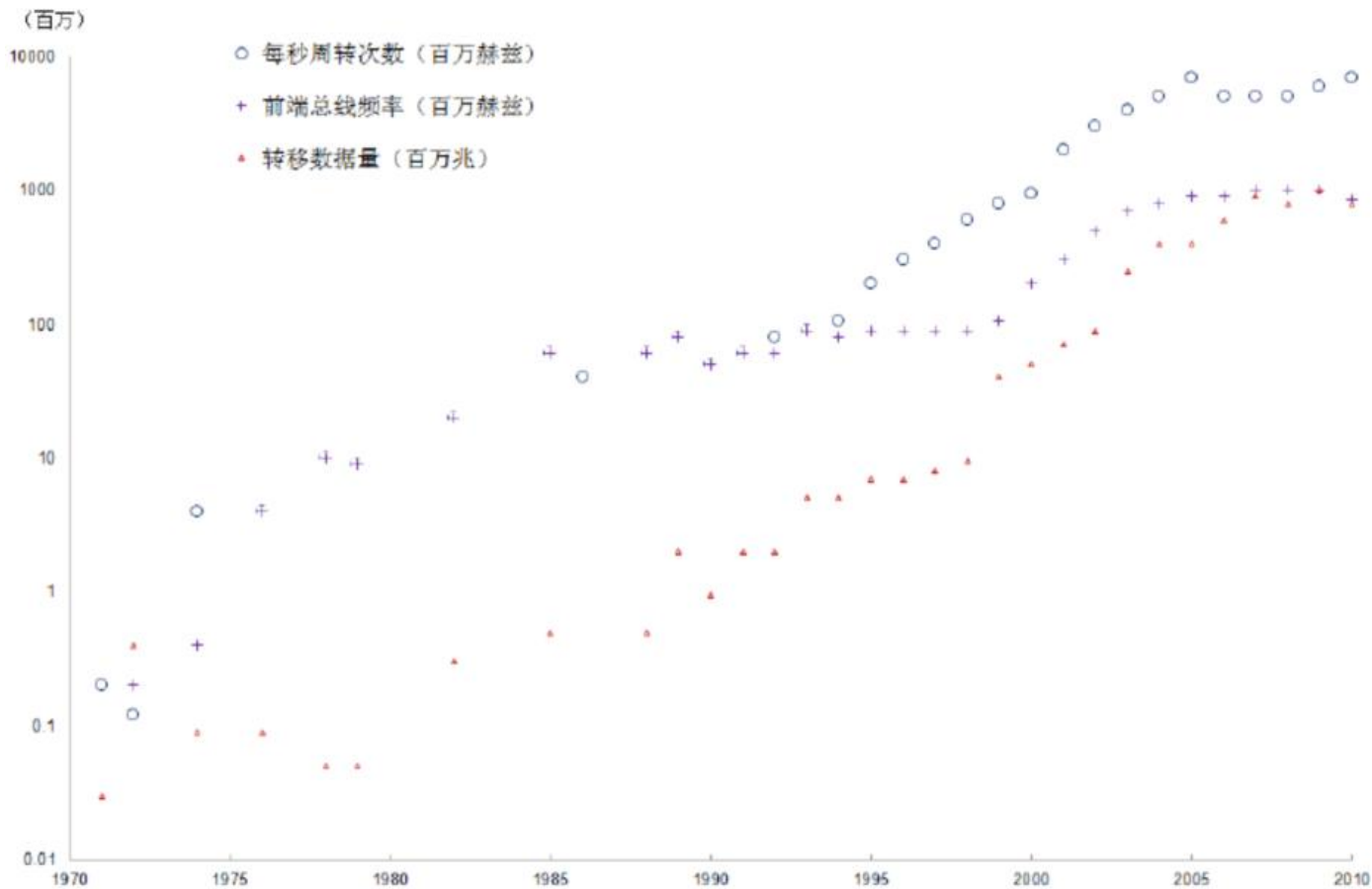


图 网络带宽随时间变化情况



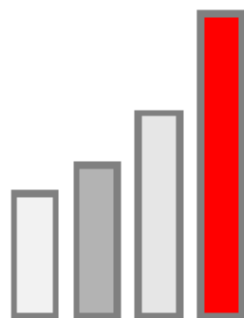
1.1.3 数据产生方式的变革促成大数据时代的来临



图 数据产生方式的变革



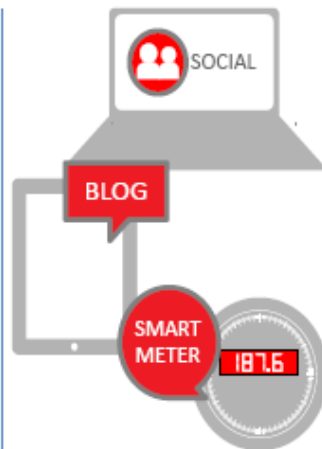
1.2 大数据概念



VOLUME
大量化



VELOCITY
快速化



VARIETY
多样化



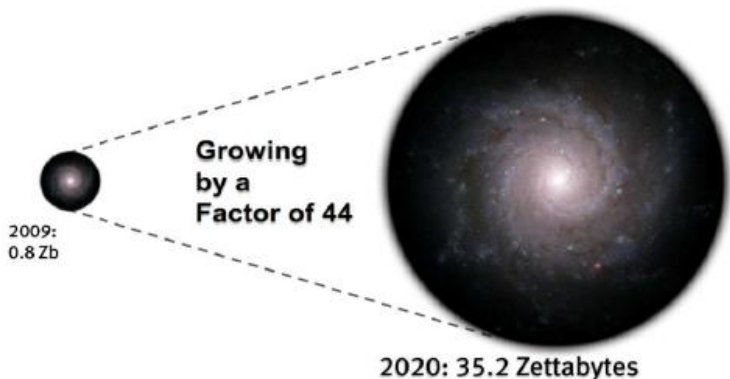
VALUE

大数据不仅仅是数据的“大量化”，而是包含“快速化”、“多样化”和“价值化”等多重属性。



1.2.1 数据量大

- 根据IDC作出的估测，数据一直都在以每年50%的速度增长，也就是说每两年就增长一倍（大数据摩尔定律）
- 人类在最近两年产生的数据量相当于之前产生的全部数据量
- 预计到2020年，全球将总共拥有35ZB的数据量，相较于2010年，数据量将增长近30倍

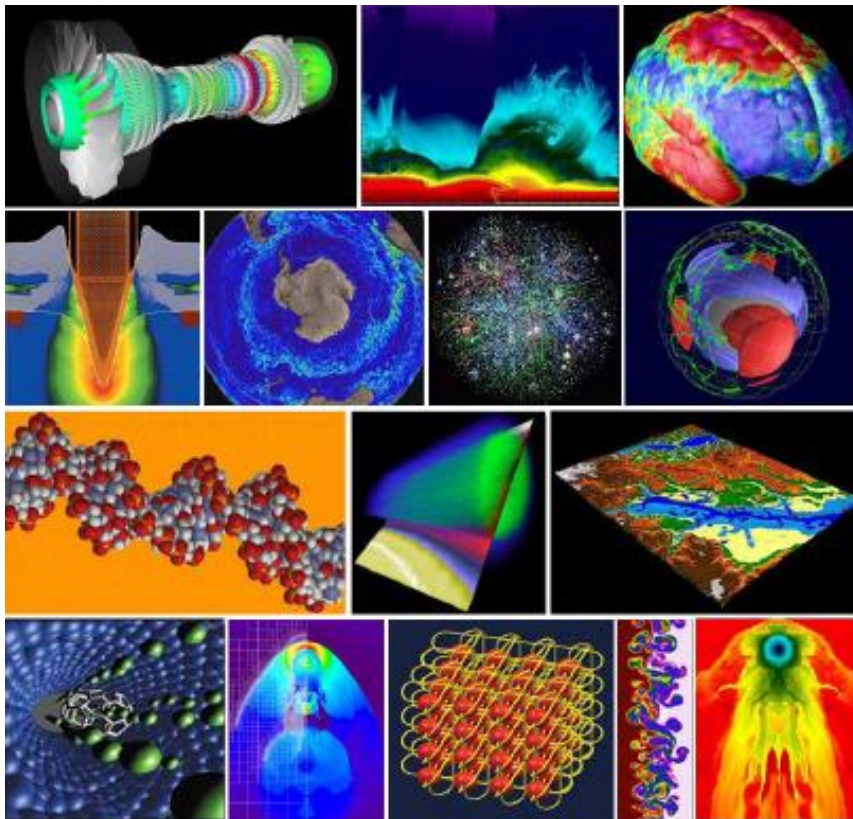


TERABYTE	10 的 12 次方	一块 1TB 硬盘		200,000 照片或 mp3 歌曲
PETABYTE	10 的 15 次方	两个数据中心机柜		16 个 Blackblaze pod 存储单元
EXABYTE	10 的 18 次方	2,000 个机柜		占据一个街区的 4 层数据中心
ZETTABYTE	10 的 21 次方	1000 个数据中心		纽约曼哈顿的 1/5 区域
YOTTABYTE	10 的 24 次方	一百万个数据中心		特拉华州和罗德岛州



1.2.2 数据类型繁多

- 大数据是由结构化和非结构化数据组成的
 - 10%的结构化数据，存储在数据库中
 - 90%的非结构化数据，它们与人类信息密切相关



- 科学研究
 - 基因组
 - LHC 加速器
 - 地球与空间探测
- 企业应用
 - Email、文档、文件
 - 应用日志
 - 交易记录
- Web 1.0数据
 - 文本
 - 图像
 - 视频
- Web 2.0数据
 - 查询日志/点击流
 - Twitter/ Blog / SNS
 - Wiki



1.2.3处理速度快

- 从数据的生成到消耗，时间窗口非常小，可用于生成决策的时间非常少
- 1秒定律：这一点也是和传统的数据挖掘技术有着本质的不同

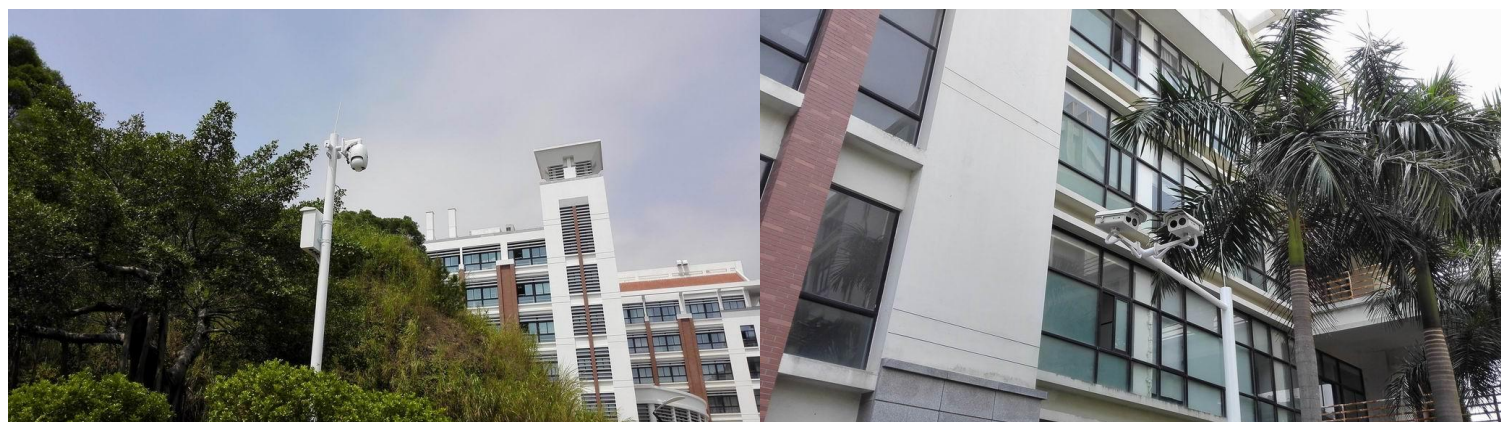




1.2.4 价值密度低

价值密度低，商业价值高

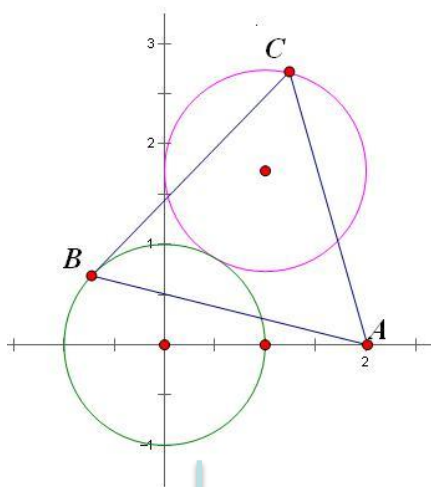
以视频为例，连续不间断监控过程中，可能有用的数据仅仅有一两秒，但是具有很高的商业价值





1.3 大数据的影响

图灵奖获得者、著名数据库专家Jim Gray 博士观察并总结人类自古以来，在科学研究上，先后历经了实验、理论、计算和数据四种范式



实验

理论

计算

数据



1.3大数据的影响

- 在思维方式方面，大数据完全颠覆了传统的思维方式：
 - 全样而非抽样
 - 效率而非精确
 - 相关而非因果





1.4 大数据关键技术

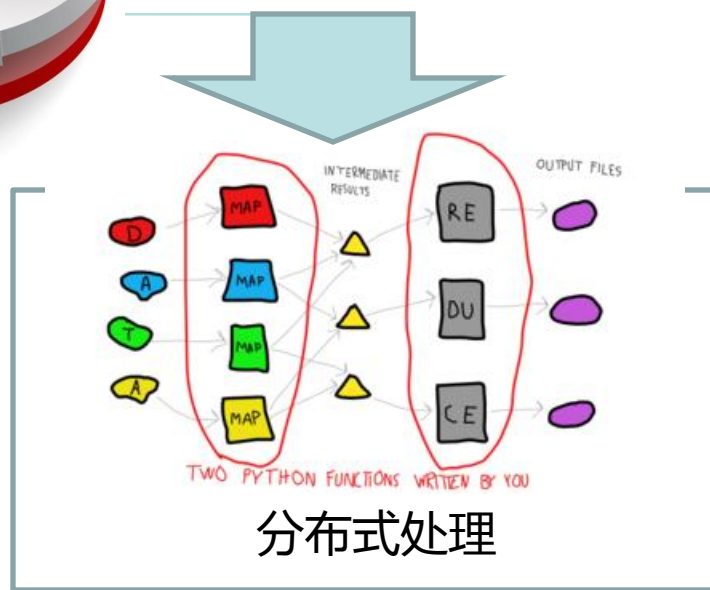
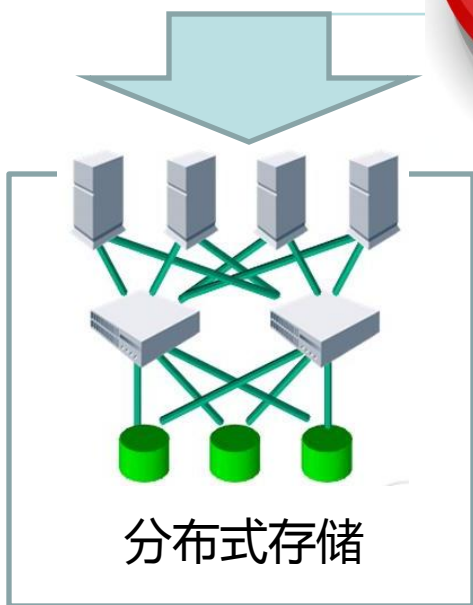
表1-5 大数据技术的不同层面及其功能

技术层面	功能
数据采集	利用ETL工具将分布的、异构数据源中的数据如关系数据、平面数据文件等，抽取到临时中间层后进行清洗、转换、集成，最后加载到数据仓库或数据集中，成为联机分析处理、数据挖掘的基础；或者也可以把实时采集的数据作为流计算系统的输入，进行实时处理分析
数据存储和管理	利用分布式文件系统、数据仓库、关系数据库、NoSQL数据库、云数据库等，实现对结构化、半结构化和非结构化海量数据的存储和管理
数据处理与分析	利用分布式并行编程模型和计算框架，结合机器学习和数据挖掘算法，实现对海量数据的处理和分析；对分析结果进行可视化呈现，帮助人们更好地理解数据、分析数据
数据隐私和安全	在从大数据中挖掘潜在的巨大商业价值和学术价值的同时，构建隐私数据保护体系和数据安全体系，有效保护个人隐私和数据安全



1.4 大数据关键技术

两大核心技术



GFS\HDFS

BigTable\HBase

NoSQL (键值、列族、图形、文档数据库)

NewSQL (如: SQL Azure)

MapReduce



1.5 大数据计算模式

表1-3 大数据计算模式及其代表产品

大数据计算模式	解决问题	代表产品
批处理计算	针对大规模数据的批量处理	MapReduce、Spark等
流计算	针对流数据的实时计算	Storm、S4、Flume、Streams、Puma、DStream、Super Mario、银河流数据处理平台等
图计算	针对大规模图结构数据的处理	Pregel、GraphX、Giraph、PowerGraph、Hama、GoldenOrb等
查询分析计算	大规模数据的存储管理和查询分析	Dremel、Hive、Cassandra、Impala等



1.6 代表性大数据技术

1.6.1 Hadoop

1.6.2 Spark

1.6.3 Flink

1.6.4 Beam



1.6.1 Hadoop

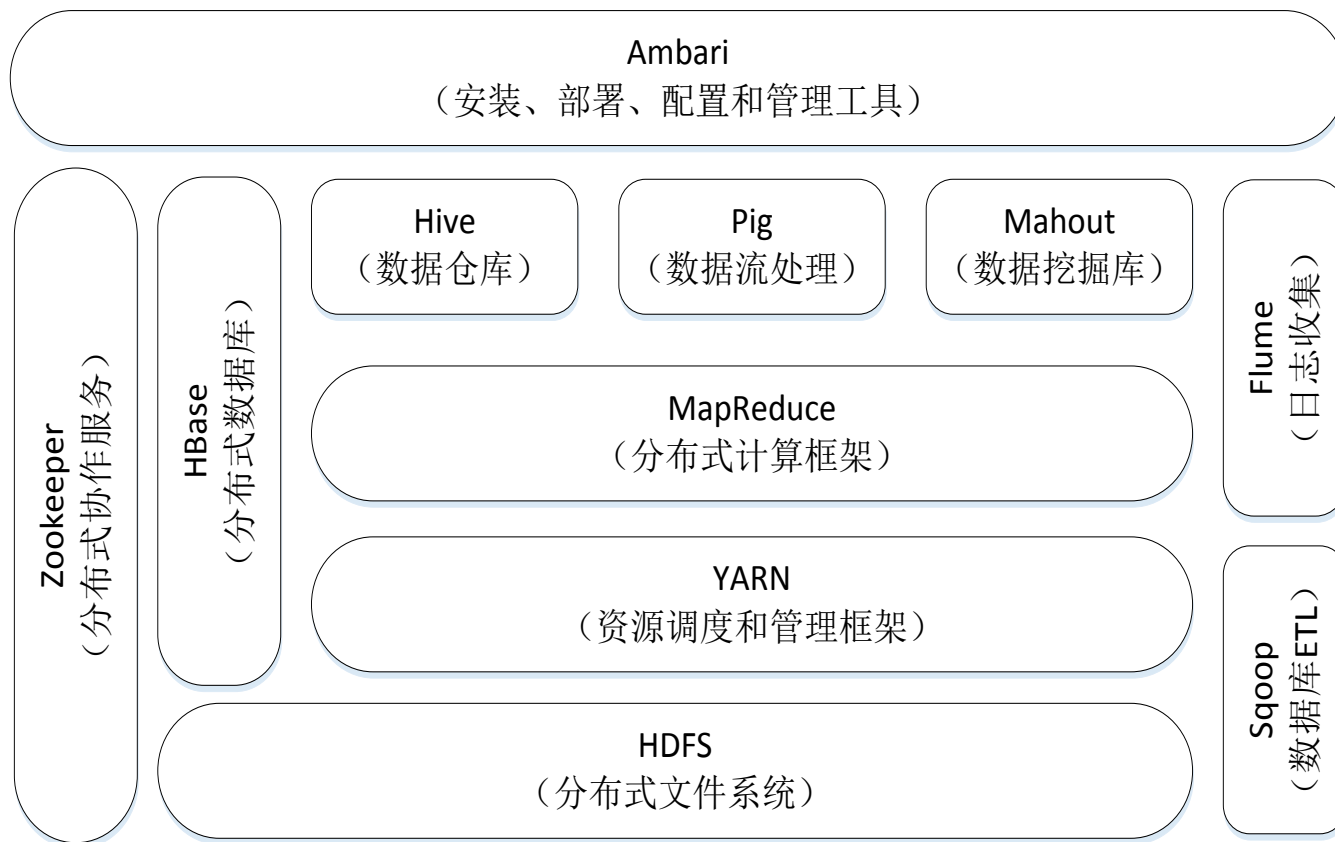


图 Hadoop生态系统



1.6.1 Hadoop——MapReduce

- MapReduce将复杂的、运行于大规模集群上的并行计算过程高度地抽象到了两个函数：**Map**和**Reduce**
- 编程容易，不需要掌握分布式并行编程细节，也可以很容易把自己的程序运行在分布式系统上，完成海量数据的计算
- MapReduce采用“**分而治之**”策略，一个存储在分布式文件系统的大规模数据集，会被切分成许多独立的分片（**split**），这些分片可以被多个**Map**任务并行处理

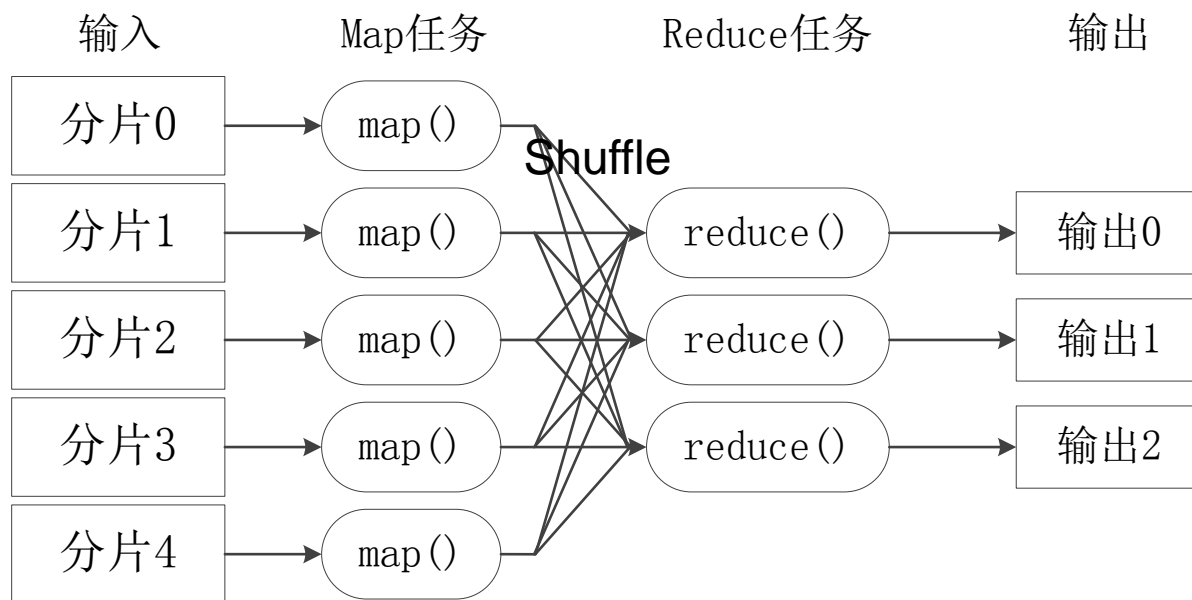


图 MapReduce工作流程



1.6.1 Hadoop——YARN

YARN的目标就是实现“一个集群多个框架”，为什么？

- 一个企业当中同时存在各种不同的业务应用场景，需要采用不同的计算框架
 - MapReduce实现离线批处理
 - 使用Impala实现实时交互式查询分析
 - 使用Storm实现流式数据实时分析
 - 使用Spark实现迭代计算
- 这些产品通常来自不同的开发团队，具有各自的资源调度管理机制
- 为了避免不同类型应用之间互相干扰，企业就需要把内部的服务器拆分成多个集群，分别安装运行不同的计算框架，即“一个框架一个集群”
- 导致问题
 - 集群资源利用率低
 - 数据无法共享
 - 维护代价高



1.6.1 Hadoop——YARN

- YARN的目标就是实现“一个集群多个框架”，即在一个集群上部署一个统一的资源调度管理框架YARN，在YARN之上可以部署其他各种计算框架
- 由YARN为这些计算框架提供统一的资源调度管理服务，并且能够根据各种计算框架的负载需求，调整各自占用的资源，实现集群资源共享和资源弹性收缩
- 可以实现一个集群上的不同应用负载混搭，有效提高了集群的利用率
- 不同计算框架可以共享底层存储，避免了数据集跨集群移动

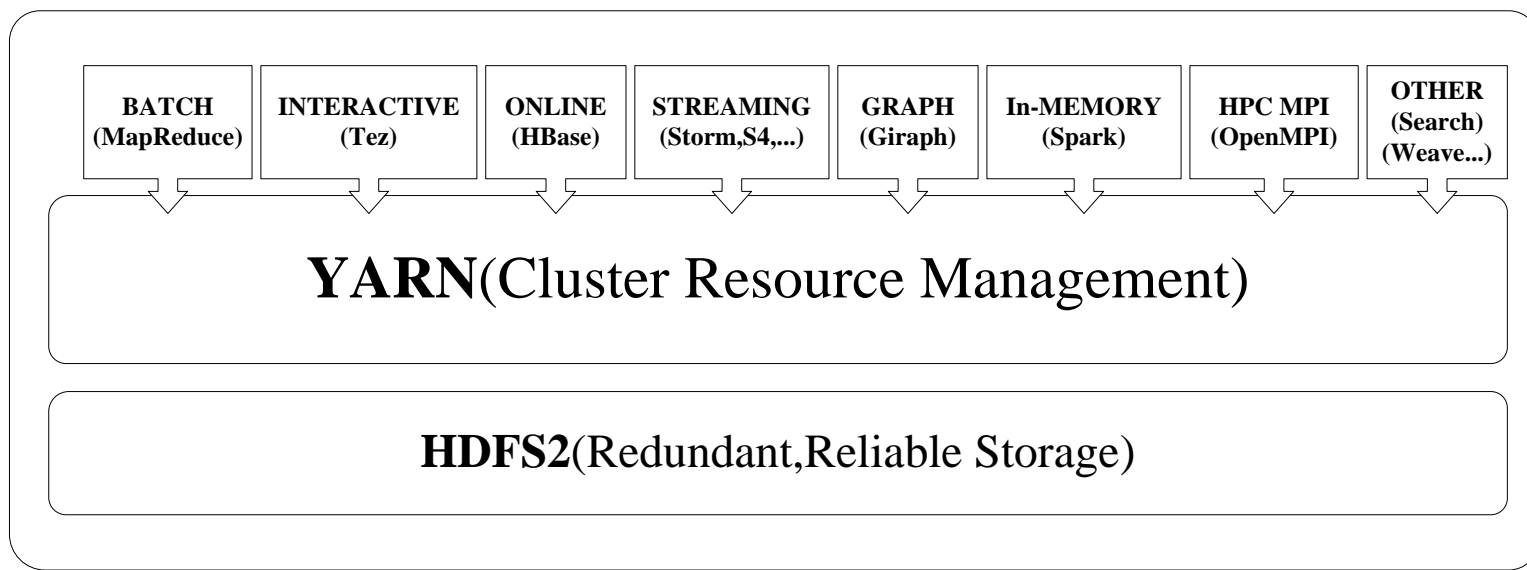
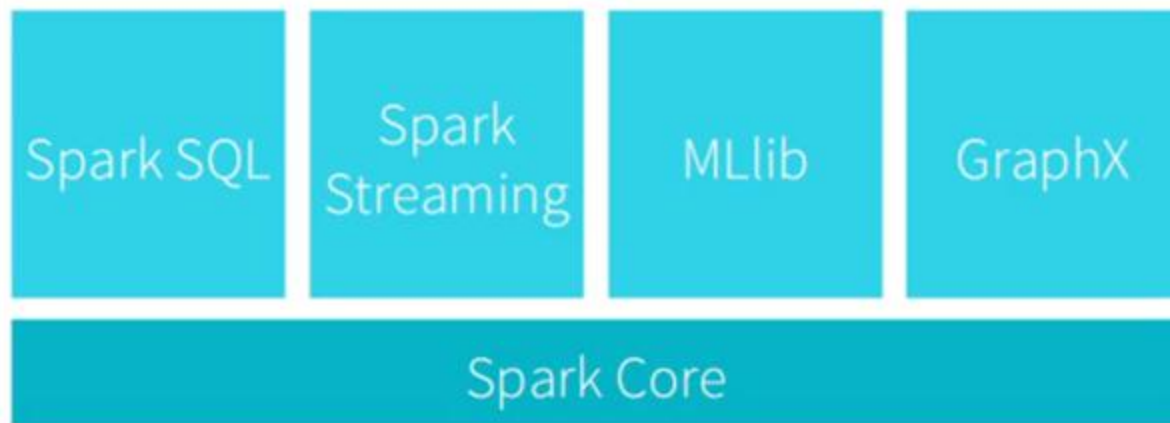


图 在YARN上部署各种计算框架



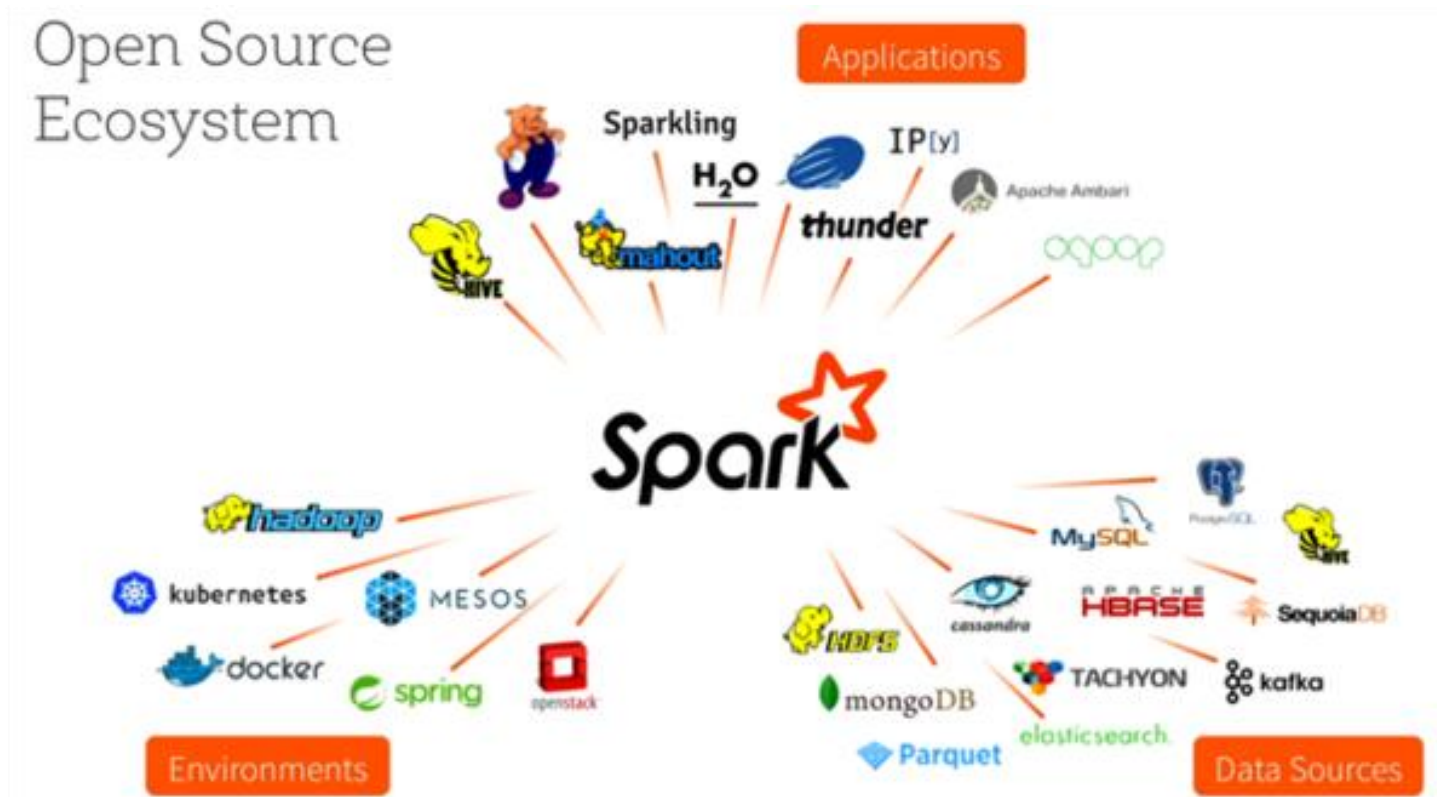
1.6.2 Spark



Spark架构图



1.6.2 Spark



Spark生态系统



1.6.2 Spark

Hadoop与Spark的对比

Hadoop存在如下一些缺点：

- 表达能力有限
- 磁盘IO开销大
- 延迟高
 - 任务之间的衔接涉及IO开销
 - 在前一个任务执行完成之前，其他任务就无法开始，难以胜任复杂、多阶段的计算任务



1.6.2 Spark

Hadoop与Spark的对比

Spark在借鉴Hadoop MapReduce优点的同时，很好地解决了MapReduce所面临的问题

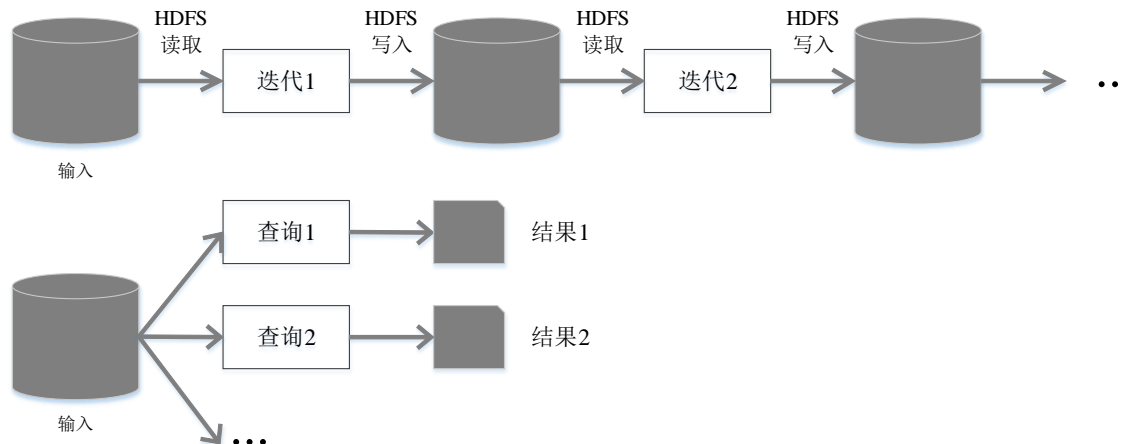
相比于Hadoop MapReduce，Spark主要具有如下优点：

- Spark的计算模式也属于MapReduce，但不局限于Map和Reduce操作，还提供了多种数据集操作类型，编程模型比Hadoop MapReduce更灵活
- Spark提供了内存计算，可将中间结果放到内存中，对于迭代运算效率更高

Spark基于DAG的任务调度执行机制，要优于Hadoop MapReduce的迭代执行机制



1.6.2 Spark



(a) Hadoop MapReduce执行流程

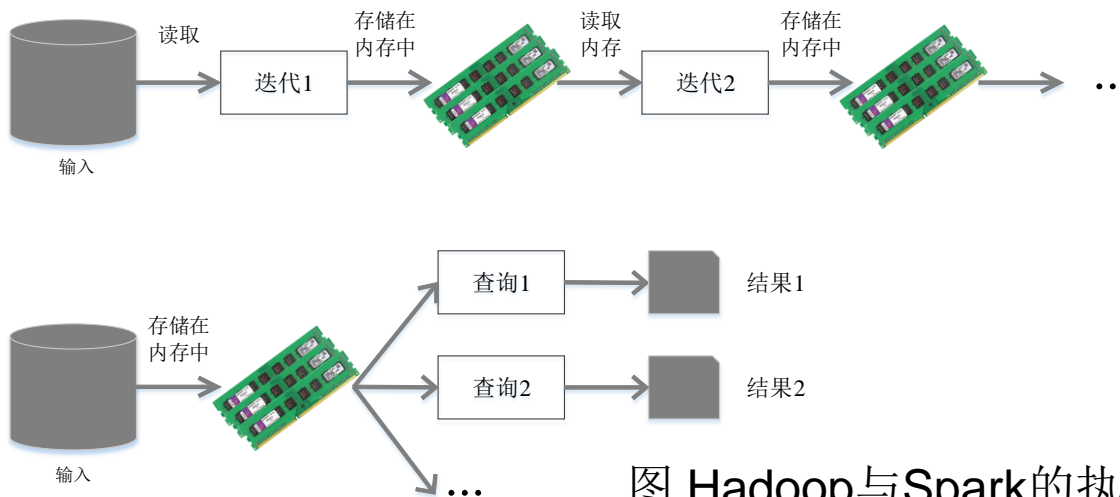


图 Hadoop与Spark的执行流程对比

(b) Spark执行流程



1.6.2 Spark

- 使用Hadoop进行迭代计算非常耗资源
- Spark将数据载入内存后，之后的迭代计算都可以直接使用内存中的中间结果作运算，避免了从磁盘中频繁读取数据

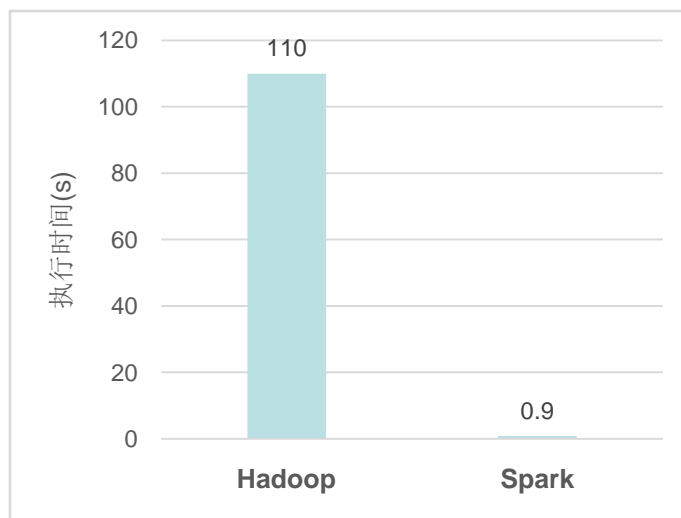
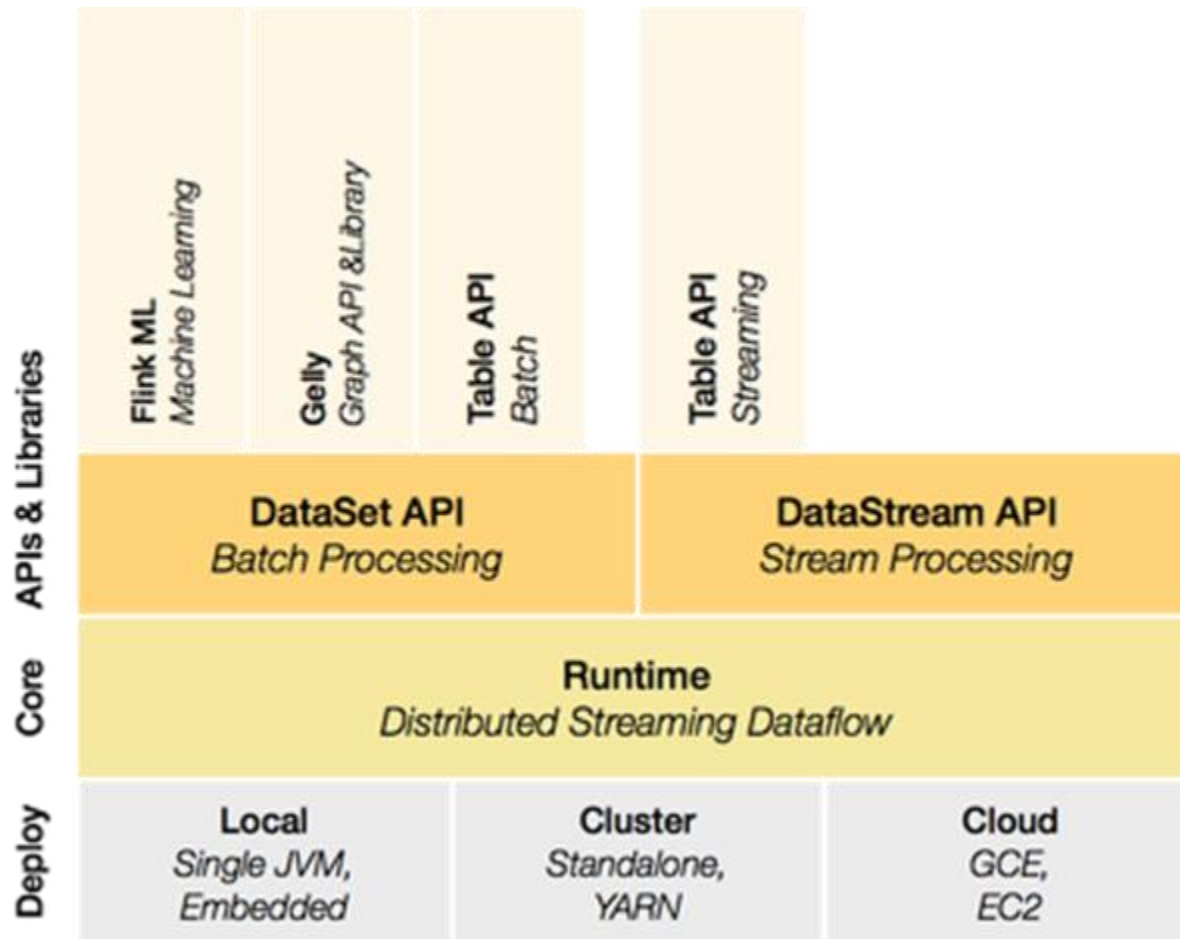


图 Hadoop与Spark执行逻辑回归的时间对比



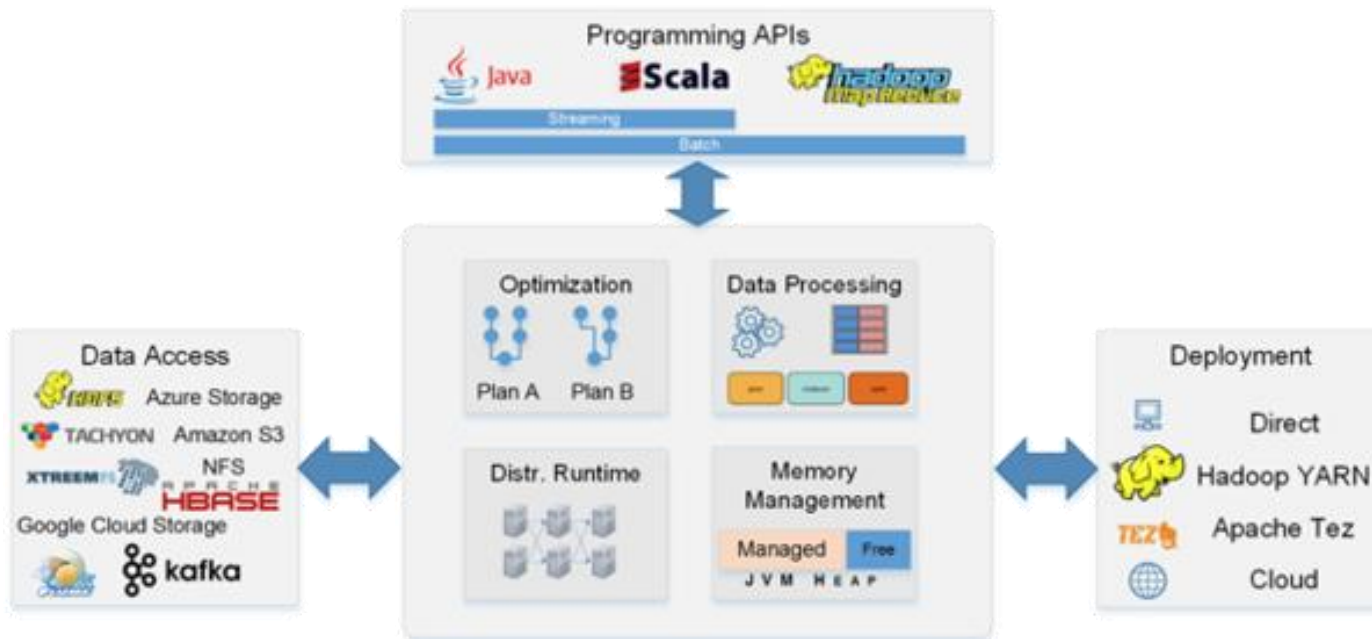
1.6.3 Flink



Flink架构图



1.6.3 Flink



Flink生态系统



1.6.3 Flink

Flink与Spark的比较

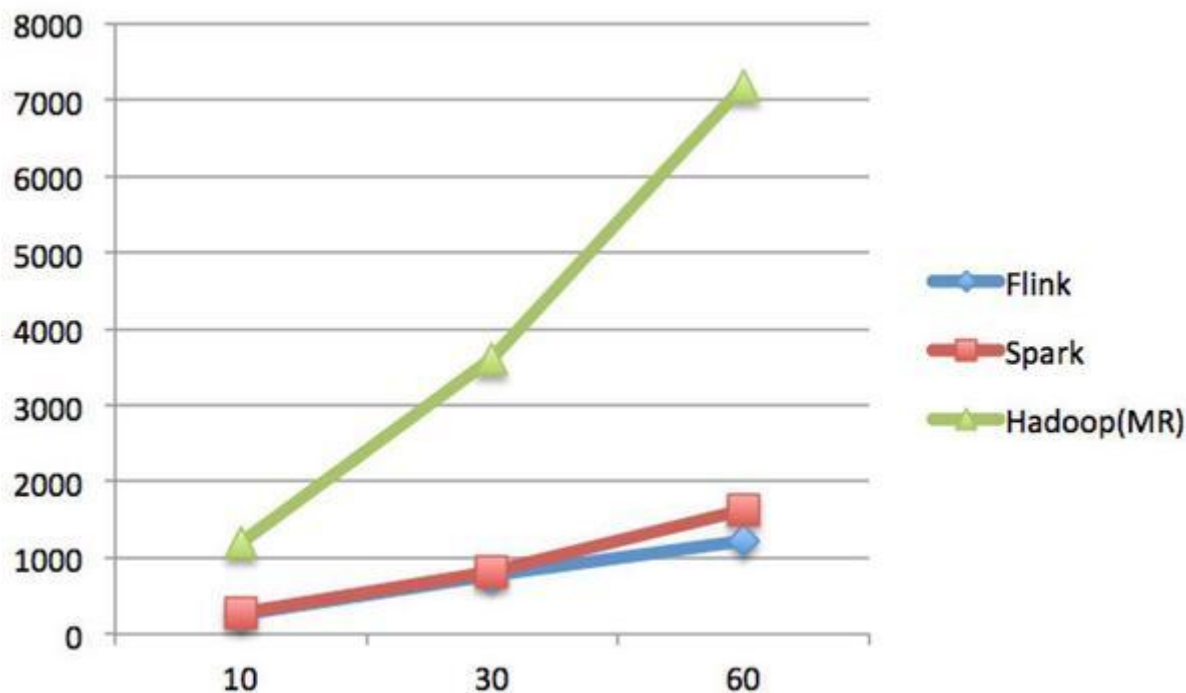
	Apache Spark	Apache Flink
核心实现	Scala	Java
编程接口	Java, Python 以及 R 语言	DataSet API 支持 Java、Scala 和 Python。 DataStream API 支持 Java and Scala
计算模型	Spark 是基于数据片集合 (RDD) 进行小批量处理, 采用了微批处理模型	Flink 是一行一行处理, 基于操作符的连续流模型。
优缺点	在流式处理方面, 不可避免增加一些延时, Spark 则只能支持秒级计算。	Flink 的流式计算跟 Storm 性能差不多, 支持毫秒级计算



1.6.3 Flink

性能对比

首先它们都可以基于内存计算框架进行实时计算，所以都拥有非常好的计算性能。经过测试，**Flink**计算性能上略好。



Spark和Flink全部都运行在Hadoop YARN上，性能为Flink > Spark > Hadoop(MR)，迭代次数越多越明显，性能上，Flink优于Spark和Hadoop最主要的原因是Flink支持增量迭代，具有对迭代自动优化的功能。



1.6.3 Flink

流式计算比较

它们都支持流式计算，**Flink**是一行一行处理，而**Spark**是基于数据片集合（**RDD**）进行小批量处理，所以**Spark**在流式处理方面，不可避免增加一些延时。**Flink**的流式计算跟**Storm**性能差不多，支持毫秒级计算，而**Spark**则只能支持秒级计算。

SQL支持

都支持**SQL**，**Spark**对**SQL**的支持比**Flink**支持的范围要大一些，另外**Spark**支持对**SQL**的优化，而**Flink**支持主要是对**API**级的优化。

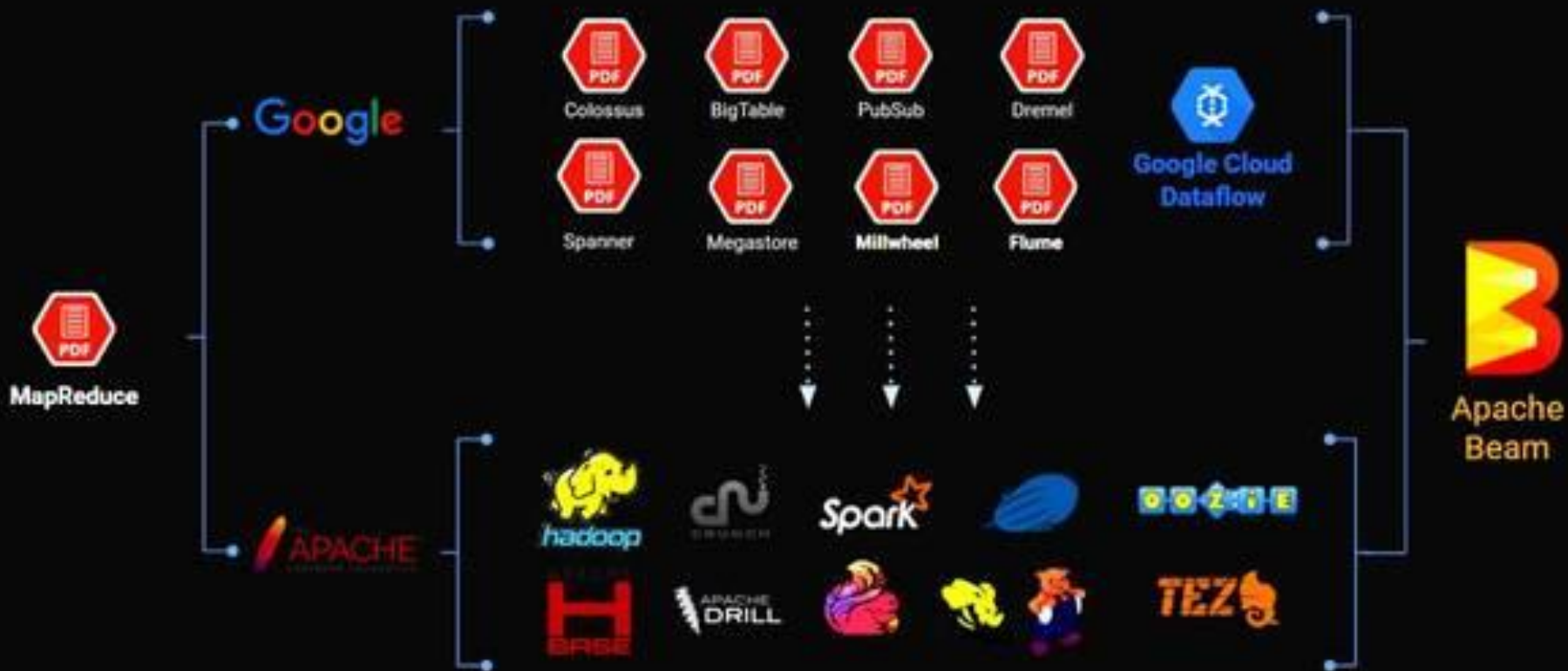
既生瑜，何生亮！



1.6.4 Beam

谷歌，Beam，一统天下？

The Evolution of Apache Beam

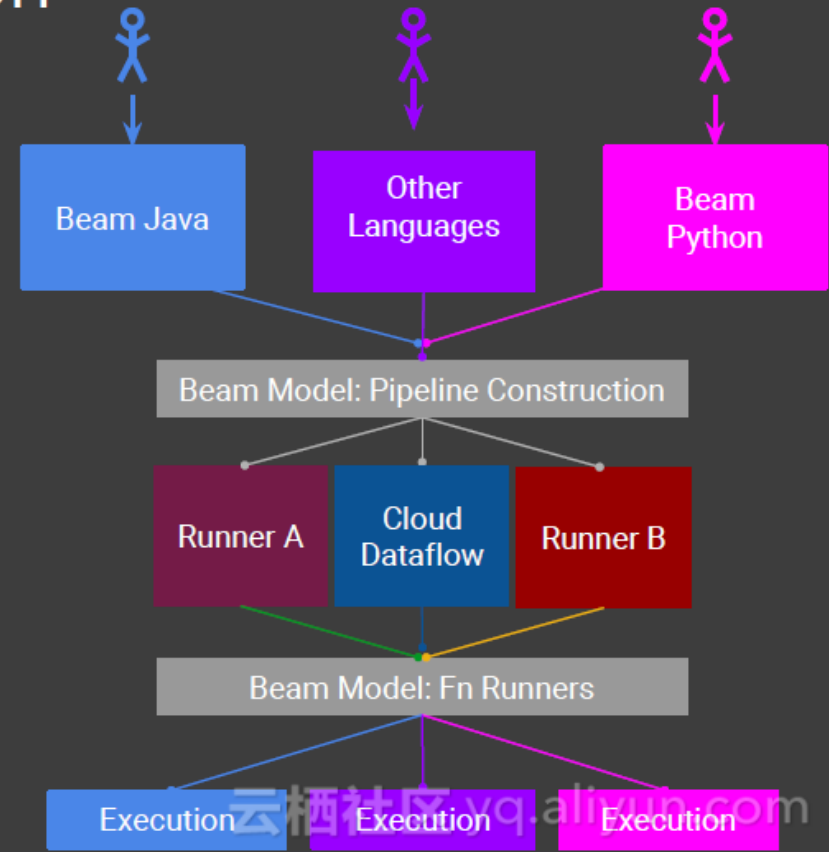




1.6.4 Beam

Apache Beam Technical Vision

1. **End users:** who want to write pipelines or transform libraries in a language that's familiar.
2. **SDK writers:** who want to make Beam concepts available in new languages.
3. **Runner writers:** who have a distributed processing environment and want to support Beam pipelines





附录：主讲教师林子雨简介



主讲教师：林子雨

单位：厦门大学计算机科学系

E-mail: ziyulin@xmu.edu.cn

个人网页: <http://www.cs.xmu.edu.cn/linziyu>

数据库实验室网站: <http://dblab.xmu.edu.cn>



扫一扫访问个人主页

林子雨，男，1978年出生，博士（毕业于北京大学），现为厦门大学计算机科学系助理教授（讲师），曾任厦门大学信息科学与技术学院院长助理、晋江市发展和改革局副局长。中国计算机学会数据库专业委员会委员，中国计算机学会信息系统专业委员会委员，荣获“2016中国大数据创新百人”称号。中国高校首个“数字教师”提出者和建设者，厦门大学数据库实验室负责人，厦门大学云计算与大数据研究中心主要建设者和骨干成员，2013年度厦门大学奖教金获得者。主要研究方向为数据库、数据仓库、数据挖掘、大数据、云计算和物联网，并以第一作者身份在《软件学报》《计算机学报》和《计算机研究与发展》等国家重点期刊以及国际学术会议上发表多篇学术论文。作为项目负责人主持的科研项目包括1项国家自然科学基金青年基金项目(No.61303004)、1项福建省自然科学基金项目(No.2013J05099)和1项中央高校基本科研业务费项目(No.2011121049)，同时，作为课题负责人完成了国家发改委城市信息化重大课题、国家物联网重大应用示范工程区域试点泉州市工作方案、2015泉州市互联网经济调研等课题。中国高校首个“数字教师”提出者和建设者，2009年至今，“数字教师”大平台累计向网络免费发布超过100万字高价值的研究和教学资料，累计网络访问量超过100万次。打造了中国高校大数据教学知名品牌，编著出版了中国高校第一本系统介绍大数据知识的专业教材《大数据技术原理与应用》，并成为京东、当当网等网店畅销书籍；建设了国内高校首个大数据课程公共服务平台，为教师教学和学生学习大数据课程提供全方位、一站式服务，年访问量超过50万次。具有丰富的政府和企业信息化培训经验，厦门大学管理学院EDP中心、浙江大学管理学院EDP中心、厦门大学继续教育学院、泉州市科技培训中心特邀培训讲师，曾给中国移动通信集团公司、福州马尾区政府、福建龙岩卷烟厂、福建省物联网科学研究院、石狮市物流协会、厦门市物流协会、浙江省中小企业家、四川泸州企业家、江苏沛县企业家等开展信息化培训，累计培训人数达3000人以上。



附录：《大数据技术原理与应用》教材



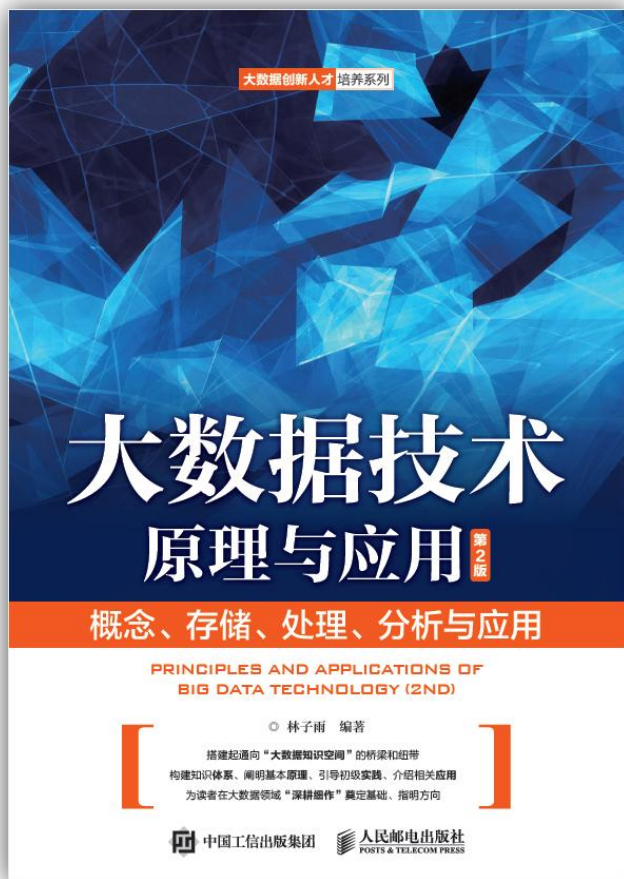
扫一扫访问教材官网

《大数据技术原理与应用——概念、存储、处理、分析与应用（第2版）》，由厦门大学计算机科学系林子雨老师编著，是中国高校第一本系统介绍大数据知识的专业教材。

全书共有15章，系统地论述了大数据的基本概念、大数据处理架构Hadoop、分布式文件系统HDFS、分布式数据库HBase、NoSQL数据库、云数据库、分布式并行编程模型MapReduce、Spark、流计算、图计算、数据可视化以及大数据在互联网、生物医学和物流等各个领域的应用。在Hadoop、HDFS、HBase、MapReduce和Spark等重要章节，安排了入门级的实践操作，让读者更好地学习和掌握大数据关键技术。

本书可以作为高等院校计算机专业、信息管理等相关专业的大数据课程教材，也可供相关技术人员参考、学习、培训之用。

欢迎访问《大数据技术原理与应用——概念、存储、处理、分析与应用（第2版）》教材官方网站：
<http://dblab.xmu.edu.cn/post/bigdata>





附录：中国高校大数据课程公共服务平台



中国高校大数据课程 公共服务平台

<http://dblab.xmu.edu.cn/post/bigdata-teaching-platform/>



扫一扫访问平台主页



扫一扫观看3分钟FLASH动画宣传片

The background is a solid blue color with faint, light blue silhouettes of people. At the top, there are two groups of people holding hands. On the right side, there is a silhouette of a person standing with their hand on their chin. At the bottom left, there are silhouettes of two people sitting at a table, one with their hand on their chin.

Thank You!

Department of Computer Science, Xiamen University, 2017