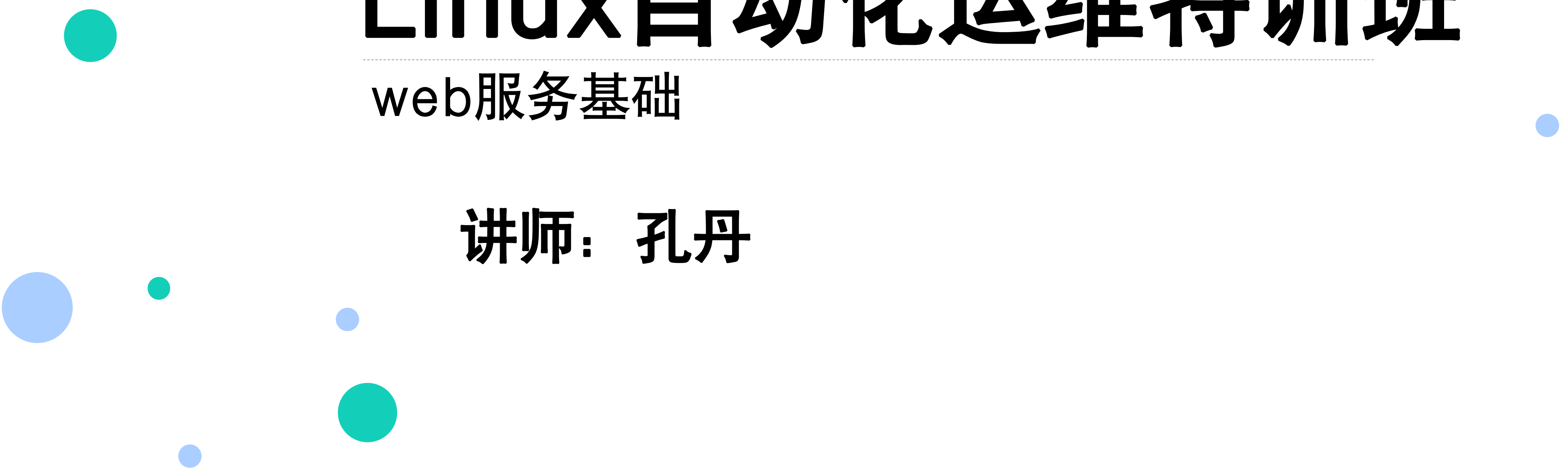




Linux自动化运维特训班

NoSQL简介

讲师：孔丹



大纲

- 什么是NoSQL?
- 为什么使用NoSQL?
- RDBMS vs NoSQL
- NoSQL 简史
- NoSQL 产品
- NoSQL 数据库分类

什么是NoSQL?

- ❑ NoSQL，指的是非关系型的数据库。NoSQL有时也称作 Not Only SQL的缩写，是对不同于传统的关系型数据库的数据库管理系统的统称。
- ❑ NoSQL用于超大规模数据的存储。（例如谷歌或Facebook每天为他们的用户收集万亿比特的数据）。这些类型的数据存储不需要固定的模式，无需多余操作就可以横向扩展。

为什么使用NoSQL？

- 今天我们可以通过第三方平台（如：Google, Facebook 等）可以很容易的访问和抓取数据。用户的个人信息，社交网络，地理位置，用户生成的数据和用户操作日志已经成倍的增加。我们如果要对这些用户数据进行挖掘，那SQL数据库已经不适合这些应用了，NoSQL数据库的发展却能很好的处理这些大的数据。
- 随着互联网 web2.0 网站的兴起，传统的关系数据库在应付 web2.0 网站，特别是超大规模和高并发的 SNS 类型的 web2.0 纯动态网站已经显得力不从心，暴露了很多难以克服的问题，而非关系型的数据库则由于其本身的特点得到了非常迅速的发展。NoSQL 数据库的产生就是为了解决大规模数据集合多重数据种类带来的挑战，尤其是大数据应用难题。

RDBMS vs NoSQL

□ RDBMS

- 高度组织化结构化数据
- 结构化查询语言 (SQL)
- 数据和关系都存储在单独的表中。
- 数据操纵语言, 数据定义语言
- 严格的一致性
- 基础事务

□ NoSQL

- 代表着不仅仅是SQL
- 没有声明性查询语言
- 没有预定义的模式
- 键-值对存储, 列存储, 文档存储, 图形数据库
- 最终一致性, 而非ACID属性
- 非结构化和不可预知的数据
- CAP定理
- 高性能, 高可用性和可伸缩性

NoSQL 简史

- NoSQL一词最早出现于1998年，是Carlo Strozzi开发的一个轻量、开源、不提供SQL功能的关系数据库。
- 2009年，Last.fm的Johan Oskarsson发起了一次关于分布式开源数据库的讨论，来自Rackspace的Eric Evans再次提出了NoSQL的概念，这时的NoSQL主要指非关系型、分布式、不提供ACID的数据库设计模式。
- 2009年在亚特兰大举行的"no:sql(east)"讨论会是一个里程碑，其口号是"select fun, profit from real_world where relational=false;"。因此，对NoSQL最普遍的解释是"非关联型的"，强调Key-Value Stores和文档数据库的优点，而不是单纯的反对RDBMS

NoSQL 产品

□ 常见的NoSQL产品：redis、memcache、mongodb等

□ NoSQL产品的显著特点：


1、NoSQL产品一般不使用严格的表关系；

2、NoSQL产品的数据查询一般不用在sql上；

<http://nosql-database.org/>

[nosql-database.org](#)

it is out there here since 2009!


ArangoDB
the multi-model NoSQL DB

LIST OF NOSQL DATABASE MANAGEMENT SYSTEMS

[currently >225]

Core NOSQL Systems: [Mostly originated out of a Web 2.0 need]

Wide Column Store / Column Families

Hadoop / HBase API: **Java / any writer**, Protocol: **any write call**, Query Method: **MapReduce Java / any exec**, Replication: **HDFS Replication**, Written in: **Java**, Concurrency: **?**, Misc: **Links: 3 Books [1, 2, 3]**, [Guru99 Article >>](#)

MapR, Hortonworks, Cloudera Hadoop Distributions and professional services .

Cassandra massively scalable, partitioned row store, masterless architecture, linear scale performance, no single points of failure, read/write support across multiple data centers & cloud availability zones. API / Query Method: **CQL and Thrift**, replication: **peer-to-peer**, written in: **Java**, Concurrency: **tunable consistency**, Misc: built-in data compression, MapReduce support, primary/secondary indexes, security features. Links: [Documentation](#), [PlanetC*](#), [Company](#).

Scylla Cassandra-compatible column store, with consistent low latency and more transactions per second. Designed with a thread-per-core model to maximize performance on modern multicore hardware. Predictable scaling. No garbage collection pauses, and faster compaction.

Hypertable API: **Thrift** (Java, PHP, Perl, Python, Ruby, etc.), Protocol: **Thrift**, Query Method: **HQL, native Thrift API**, Replication: **HDFS Replication**, Concurrency: **MVCC**, Consistency Model: **Fully consistent** Misc: High performance C++ implementation of Google's Bigtable. [» Commercial support](#)

NoSQL FORUMS

- [Global NOSQL Forum >>](#)
- [Forum Berlin >>](#)
- [Forum France >>](#)
- [Forum Japan >>](#)

NoSQL NEWS FEEDS

- [MyNoSQL by Alex P >>](#)
- [On Twitter: nosqlupdate >>](#)
- [NoSQL Weekly >> * new *](#)
- [HighScalability Blog >>](#)

Select the right Database Management System [»](#)

FUN

- [A NoSQL parody >>](#)

» A SQL-mimicking NoSQL language

NoSQL 数据库分类

类型	部分代表	特点
列存储	Hbase Cassandra Hypertable	顾名思义，是按列存储数据的。最大的特点是方便存储结构化和半结构化数据，方便做数据压缩，对针对某一系列或者某几列的查询有非常大的IO优势。
文档存储	MongoDB CouchDB	文档存储一般用类似json的格式存储，存储的内容是文档型的。这样也就有机会对某些字段建立索引，实现关系数据库的某些功能。
key-value存储	Tokyo Cabinet / Tyrant Berkeley DB MemcacheDB Redis	可以通过key快速查询到其value。一般来说，存储不管value的格式，照单全收。（Redis包含了其他功能）
图存储	Neo4J FlockDB	图形关系的最佳存储。使用传统关系数据库来解决的话性能低下，而且设计使用不方便。
对象存储	db4o Versant	通过类似面向对象语言的语法操作数据库，通过对象的方式存取数据。
xml数据库	Berkeley DB XML BaseX	高效的存储XML数据，并支持XML的内部查询语法，比如XQuery,Xpath。

总结

- NoSQL简介

- 常用NoSQL数据库



谢谢观看

更多好课，请关注[万门大学APP](#)

