NoSQL(NoSQL = Not Only SQL )，意即"不仅仅是SQL"。

在现代的计算系统上每天网络上都会产生庞大的数据量。

这些数据有很大一部分是由关系数据库管理系统（RDBMS）来处理。

NoSQL 是一项全新的数据库革命性运动，早期就有人提出，发展至2009年趋势越发高涨。NoSQL的拥护者们提倡运用非关系型的数据存储，相对于铺天盖地的关系型数据库运用，这一概念无疑是一种全新的思维的注入。

## 关系型数据库遵循ACID规则

**1、A (Atomicity) 原子性**

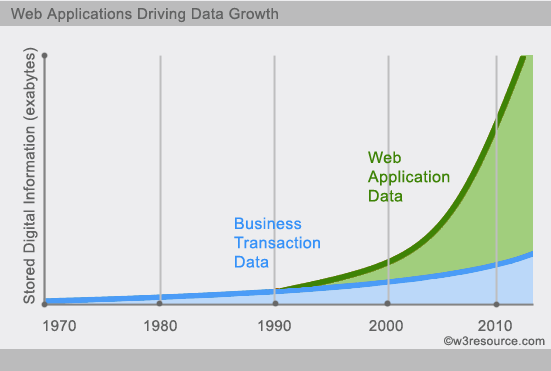
**2、C (Consistency) 一致性**

**3、I (Isolation) 独立性**

**4、D (Durability) 持久性**

持久性是指一旦事务提交后，它所做的修改将会永久的保存在数据库上，即使出现宕机也不会丢失。

## 为什么使用NoSQL ?



## CAP定理（CAP theorem）

在计算机科学中, CAP定理（CAP theorem）, 又被称作 布鲁尔定理（Brewer's theorem）, 它指出对于一个分布式计算系统来说，不可能同时满足以下三点:

* **一致性(Consistency)** (所有节点在同一时间具有相同的数据)
* **可用性(Availability)** (保证每个请求不管成功或者失败都有响应)
* **分隔容忍(Partition tolerance)** (系统中任意信息的丢失或失败不会影响系统的继续运作)

## 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。 |