<http://blog.csdn.net/longxingzhiwen/article/details/53896702>

当前主流的关系型[**数据库**](http://lib.csdn.net/base/mysql)有[**Oracle**](http://lib.csdn.net/base/oracle)、DB2、Microsoft SQL Server、Microsoft Access、**[MySQL](http://lib.csdn.net/base/mysql" \o "MySQL知识库" \t "_blank)**等。

非关系型数据库有 NoSql、Cloudant。

nosql和关系型数据库比较？  
优点：  
1）成本：nosql数据库简单易部署，基本都是开源软件，不需要像使用oracle那样花费大量成本购买使用，相比关系型数据库价格便宜。  
2）查询速度：nosql数据库将数据存储于缓存之中，关系型数据库将数据存储在硬盘中，自然查询速度远不及nosql数据库。  
3）存储数据的格式：nosql的存储格式是key,value形式、文档形式、图片形式等等，所以可以存储基础类型以及对象或者是集合等各种格式，而数据库则只支持基础类型。  
4）扩展性：关系型数据库有类似join这样的多表查询机制的限制导致扩展很艰难。  
缺点：  
1）维护的工具和资料有限，因为nosql是属于新的技术，不能和关系型数据库10几年的技术同日而语。  
2）不提供对sql的支持，如果不支持sql这样的工业标准，将产生一定用户的学习和使用成本。

3）不提供关系型数据库对事物的处理。

非关系型数据库的优势：1. 性能NOSQL是基于键值对的，可以想象成表中的主键和值的对应关系，而且不需要经过SQL层的解析，所以性能非常高。2. 可扩展性同样也是因为基于键值对，数据之间没有耦合性，所以非常容易水平扩展。

关系型数据库的优势：1. 复杂查询可以用SQL语句方便的在一个表以及多个表之间做非常复杂的数据查询。2. 事务支持使得对于安全性能很高的数据访问要求得以实现。对于这两类数据库，对方的优势就是自己的弱势，反之亦然。

关系型数据库把所有的数据都通过行和列的二元表现形式表示出来。

关系型数据库的优势：

1. 保持数据的一致性（事务处理）

2.由于以标准化为前提，数据更新的开销很小（相同的字段基本上都只有一处）

3. 可以进行Join等复杂查询

其中能够保持数据的一致性是关系型数据库的最大优势。

关系型数据库的不足：

不擅长的处理

1. 大量数据的写入处理

2. 为有数据更新的表做索引或表结构（schema）变更

3. 字段不固定时应用

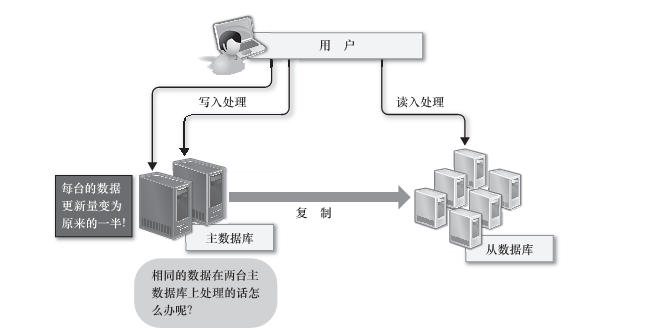
4. 对简单查询需要快速返回结果的处理

--大量数据的写入处理

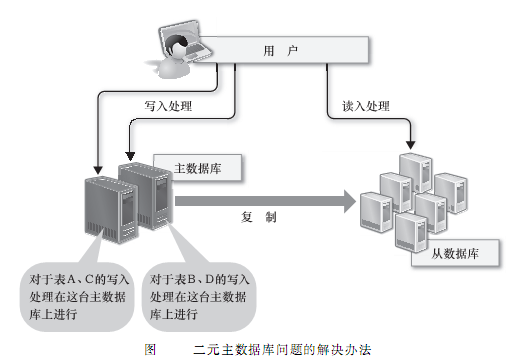
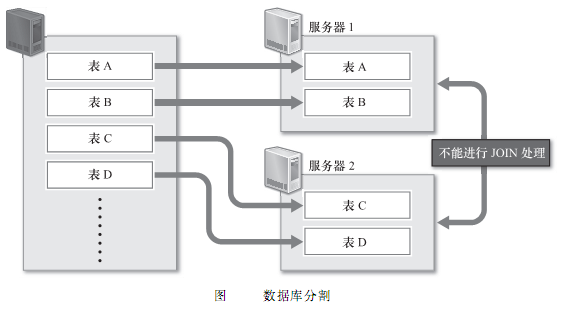
读写集中在一个数据库上让数据库不堪重负，大部分网站已使用主从复制技术实现读写分离，以提高读写性能和读库的可扩展性。

所以在进行大量数据操作时，会使用数据库主从模式。数据的写入由主数据库负责，数据的读入由从数据库负责，可以比较简单地通过增加从数据库来实现规模化，但是数据的写入却完全没有简单的方法来解决规模化问题。

第一，要想将数据的写入规模化，可以考虑把主数据库从一台增加到两台，作为互相关联复制的二元主数据库使用，确实这样可以把每台主数据库的负荷减少一半，但是更新处理会发生冲突，可能会造成数据的不一致，为了避免这样的问题，需要把对每个表的请求分别分配给合适的主数据库来处理。

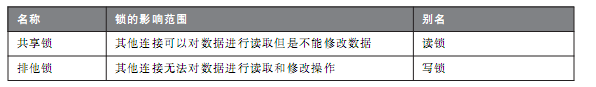


第二，可以考虑把数据库分割开来，分别放在不同的数据库服务器上，比如将不同的表放在不同的数据库服务器上，数据库分割可以减少每台数据库服务器上的数据量，以便减少硬盘IO的输入、输出处理，实现内存上的高速处理。但是由于分别存储字不同服务器上的表之间无法进行Join处理，数据库分割的时候就需要预先考虑这些问题，数据库分割之后，如果一定要进行Join处理，就必须要在程序中进行关联，这是非常困难的。

--为有数据更新的表做索引或表结构变更

在使用关系型数据库时，为了加快查询速度需要创建索引，为了增加必要的字段就一定要改变表结构，为了进行这些处理，需要对表进行共享锁定，这期间数据变更、更新、插入、删除等都是无法进行的。如果需要进行一些耗时操作，例如为数据量比较大的表创建索引或是变更其表结构，就需要特别注意，长时间内数据可能无法进行更新。



--字段不固定时的应用

如果字段不固定，利用关系型数据库也是比较困难的，有人会说，需要的时候加个字段就可以了，这样的方法也不是不可以，但在实际运用中每次都进行反复的表结构变更是非常痛苦的。你也可以预先设定大量的预备字段，但这样的话，时间一长很容易弄不清除字段和数据的对应状态，即哪个字段保存有哪些数据。

--对简单查询需要快速返回结果的处理  （这里的“简单”指的是没有复杂的查询条件）

这一点称不上是缺点，但不管怎样，关系型数据库并不擅长对简单的查询快速返回结果，因为关系型数据库是使用专门的sql语言进行数据读取的，它需要对sql与越南进行解析，同时还有对表的锁定和解锁等这样的额外开销，这里并不是说关系型数据库的速度太慢，而只是想告诉大家若希望对简单查询进行高速处理，则没有必要非使用关系型数据库不可。

---------------------------

NoSQL数据库

关系型数据库应用广泛，能进行事务处理和表连接等复杂查询。相对地，NoSQL数据库只应用在特定领域，基本上不进行复杂的处理，但它恰恰弥补了之前所列举的关系型数据库的不足之处。

优点：

 易于数据的分散

各个数据之间存在关联是关系型数据库得名的主要原因，为了进行join处理，关系型数据库不得不把数据存储在同一个服务器内，这不利于数据的分散，这也是关系型数据库并不擅长大数据量的写入处理的原因。相反NoSQL数据库原本就不支持Join处理，各个数据都是独立设计的，很容易把数据分散在多个服务器上，故减少了每个服务器上的数据量，即使要处理大量数据的写入，也变得更加容易，数据的读入操作当然也同样容易。

**典型的NoSQL数据库**

临时性键值存储（memcached、Redis）、永久性键值存储（ROMA、Redis）、面向文档的数据库（MongoDB、CouchDB）、面向列的数据库（Cassandra、HBase）

一、 键值存储

它的数据是以键值的形式存储的，虽然它的速度非常快，但基本上只能通过键的完全一致查询获取数据，根据数据的保存方式可以分为临时性、永久性和两者兼具 三种。

（1）临时性

      所谓临时性就是数据有可能丢失，memcached把所有数据都保存在内存中，这样保存和读取的速度非常快，但是当memcached停止时，数据就不存在了。由于数据保存在内存中，所以无法操作超出内存容量的数据，旧数据会丢失。总结来说：

      。在内存中保存数据

      。可以进行非常快速的保存和读取处理

      。数据有可能丢失

 （2）永久性

       所谓永久性就是数据不会丢失，这里的键值存储是把数据保存在硬盘上，与临时性比起来，由于必然要发生对硬盘的IO操作，所以性能上还是有差距的，但数据不会丢失是它最大的优势。总结来说：

       。在硬盘上保存数据

       。可以进行非常快速的保存和读取处理（但无法与memcached相比）

       。数据不会丢失

（3） 两者兼备

       Redis属于这种类型。Redis有些特殊，临时性和永久性兼具。Redis首先把数据保存在内存中，在满足特定条件（默认是 15分钟一次以上，5分钟内10个以上，1分钟内10000个以上的键发生变更）的时候将数据写入到硬盘中，这样既确保了内存中数据的处理速度，又可以通过写入硬盘来保证数据的永久性，这种类型的数据库特别适合处理数组类型的数据。总结来说：

       。同时在内存和硬盘上保存数据

       。可以进行非常快速的保存和读取处理

       。保存在硬盘上的数据不会消失（可以恢复）

       。适合于处理数组类型的数据

二、面向文档的数据库

   MongoDB、CouchDB属于这种类型，它们属于NoSQL数据库，但与键值存储相异。

   （1）不定义表结构

　　   即使不定义表结构，也可以像定义了表结构一样使用，还省去了变更表结构的麻烦。

   （2）可以使用复杂的查询条件

　　   跟键值存储不同的是，面向文档的数据库可以通过复杂的查询条件来获取数据，虽然不具备事务处理和Join这些关系型数据库所具有的处理能力，但初次以外的其他处理基本上都能实现。

三、 面向列的数据库

   Cassandra、HBae、HyperTable属于这种类型，由于近年来数据量出现爆发性增长，这种类型的NoSQL数据库尤其引入注目。

   普通的关系型数据库都是以行为单位来存储数据的，擅长以行为单位的读入处理，比如特定条件数据的获取。因此，关系型数据库也被成为面向行的数据库。相反，面向列的数据库是以列为单位来存储数据的，擅长以列为单位读入数据。



面向列的数据库具有搞扩展性，即使数据增加也不会降低相应的处理速度（特别是写入速度），所以它主要应用于需要处理大量数据的情况。另外，把它作为批处理程序的存储器来对大量数据进行更新也是非常有用的。但由于面向列的数据库跟现行数据库存储的思维方式有很大不同，故应用起来十分困难。

总结：关系型数据库与NoSQL数据库并非对立而是互补的关系，即通常情况下使用关系型数据库，在适合使用NoSQL的时候使用NoSQL数据库，让NoSQL数据库对关系型数据库的不足进行弥补。