

关系型数据库与 NoSQL 数据库

张华强

(安徽南瑞继远软件有限公司,安徽 合肥 230088)

摘要:通过对传统关系型数据库以及他目前存在的问题的说明,进而引出了 NoSQL 数据库,并指出 NoSQL 数据库的特点以及目前应用状况,最后总结出传统关系型数据库和 NoSQL 数据库在某种场景下结合使用,并给出具体实例加以说明。

关键词:NoSQL 数据库;关系型数据库;CAP 理论

中图分类号:TP311 **文献标识码:**A **文章编号:**1009-3044(2011)20-4802-03

Relational Database and NoSQL Database

ZHANG Hua-qiang

(Anhui Nari Jiyuan Software CO.,LTD., Hefei 230088,China)

Abstract: This paper introduces a database called NoSQL,throughout the descriptions of the traditional relational database and its present problems, meanwhile,points out the characteristics of NoSQL database and the current application situations; finally, summarizes how to use in combination with NoSQL database and the traditional relational database in some scenes and illustrates with some examples.

Key words: relational database; NoSQL database; CAP theory

回顾数据库的发展历程,数据库技术从 60 年代末开始,经历了层次数据库、网状数据库和关系数据库而进入数据库管理系统(DBMS)阶段至今,数据库技术的研究也不断取得进展。最近几十年,关系型数据库成为发展的主流,几乎所有新推出的 DBMS 产品都是关系型的。关系型数据库在计算机数据管理的发展史上是一个重要的里程碑。但最近 NoSQL 数据库却风声鹤起,引起了人们的极大关注。NoSQL 数据库并不是最近才出现的,很多 NoSQL 数据库实现都已经存在了十多年了,有很多成功案例,是什么原因让它们比以前更受欢迎了呢? NoSQL 数据库会不会替代现有的关系型数据库呢? 本文将一一为你做出解答。

1 关系型数据库

1.1 关系型数据库概述

关系型数据库是支持关系模型的数据库系统,他是目前各类数据库中最重要,也是使用最广泛的数据库系统。关系型数据库从诞生到现在经过几十年的发展,已经变的比较成熟,目前市场上主流的数据库都为关系型数据库,比较知名的如 Sybase, Oracle, Informix, SQL Server, DB2 等。

1.2 关系型数据库的优势

关系型数据库相比其他模型的数据库而言,有着以下优点:

容易理解:关系模型中的二维表结构非常贴近逻辑世界,相对于网状、层次等其他模型来说更容易理解。

使用方便:通用的 SQL 语言使得操作关系型数据库非常方便,只需使用 SQL 语言在逻辑层面操作数据库,而完全不必理解其底层实现。

易于维护:丰富的完整性(实体完整性、参照完整性和用户定义的完整性)大大降低了数据冗余和数据不一致的概率。

1.3 关系型数据库存在的问题

传统的关系型数据库具有不错的性能,高稳定型,久经历史考验,而且使用简单,功能强大,同时也积累了大量的成功案例。在 90 年代的互联网领域,一个网站的访问量一般都不大,用单个数据库完全可以轻松应付。在那个时候,更多的都是静态网页,动态交互类型的网站不多。可是最近几年,互联网 Web2.0 网站开始快速发展。火爆的论坛、博客、微博逐渐引领 web 领域的潮流。传统的关系型数据库在应付这些超大规模和高并发的纯动态网站显得力不从心,暴露了很多难以克服的问题。

数据库高并发读写:高并发的纯动态网站一般都是根据用户个性化信息来实时生成动态页面和提供动态信息,所以基本上无法使用动态页面静态化技术,因此数据库并发负载非常高,往往要达到每秒上万次读写请求。关系数据库应付上万次 SQL 查询还勉强顶得住,但是应付上万次 SQL 写数据请求,硬盘 IO 就已经无法承受了。

海量数据的高效率存储和访问:上述提到的 Web2.0 网站,每天用户会产生海量的动态信息,对于关系数据库来说,在一张数以亿计条记录的表里面进行 SQL 查询,效率是极其低下,难以忍受的。

数据库的高可扩展性和高可用性:基于 web 的架构当中,数据库无法通过添加更多的硬件和服务节点来扩展性能和负载能力,对于很多需要提供 24 小时不间断服务的网站来说,数据库系统升级和扩展只能通过停机来实现,这无疑是一个艰难的决定。

收稿日期:2011-05-12

作者简介:张华强(1983-),男,安徽肥东人,工程师,本科,研究方向为计算机应用技术。

2 NoSQL 数据库

2.1 NoSQL 数据库概述

NoSQL 数据库是非关系型数据存储的广义定义,它打破了长久以来关系型数据库与 ACID 理论大一统的局面。NoSQL 数据存储不需要固定的表结构,通常也不存在连接操作。NoSQL 数据库不使用传统的关系数据库模型,而是使用如 key-value 存储、文档型的、列存储、图型数据库、xml 等方式存储数据模型。其中用的最多的是: key-value 存储。

2.2 NoSQL 数据库优势

易扩展:NoSQL 数据库种类繁多,但是一个共同的特点都是去掉关系型数据库的关系型特性。数据之间无关系,这样就非常容易扩展。也无形之间,在架构的层面上带来了可扩展的能力。

大数据量,高性能:同样由于数据之间无关系,数据库的结构简单,在大数据量下,NoSQL 数据库表现出非常高的读写性能。

灵活的数据模型:NoSQL 数据库无需事先为要存储的数据建立字段,随时可以存储自定义的数据格式。而在关系型数据库里,增删字段是一件非常麻烦的事情。如果是非常大数据量的表,增加字段简直难以想象。

2.3 NoSQL 数据库应用现状

NoSQL 数据库并不是最近才出现的,很多 NoSQL 数据库实现都已经存在了十多年了,有很多成功案例,是什么原因让它们比以前更受欢迎了呢?首先是由于社会化网络和云计算的发展,一些原先只有很高端的组织才会面临的问题,如今已经成为普遍问题了。其次,已有的方法已经被发现无法跟随需求一起扩展了。并且成本的压力让很多组织需要去寻找更高性价比的方案,并且研究证实基于普通廉价硬件的分布式存储解决方案甚至比现在的高端数据库更加可靠。所有这些导致了对 NoSQL 数据库的需求。

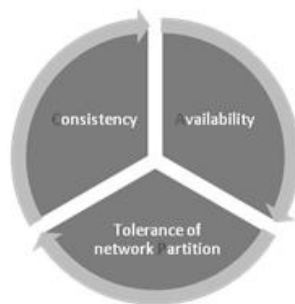
Web2.0 技术在网络中的广泛应用为 NoSQL 数据库发展提供了充足的动力,市场上先后出现了十多种比较流行的 NoSQL 产品,例如:Redis,Tokyo Cabinet,Cassandra,Voldemort,MongoDB,Dynomite,HBase,CouchDB,Hypertable,Riak,Tin,Flare,Lightcloud,KiokuDB,Scalaris,Kai,ThruDB,..... 这些 NoSQL 数据库都有自己的独到之处。有满足极高读写性能需求的 Kye-Value 数据库:Redis,Tokyo Cabinet,Flare;还有满足海量存储需求和访问的面向文档的数据库:MongoDB,CouchDB;以及满足高可扩展性和可用性的面向分布式计算的数据库:Cassandra,Voldemort;在此就不详细介绍每款 NoSQL 数据库了。

3 关系型数据库与 NoSQL 数据库结合

伴随着越来越多的 NoSQL 产品涌现出来,NoSQL 数据库会不会替代现有的关系数据库?在说明之前,我们先简单了解下 CAP 理论,以及 ACID、BASE。

3.1 CAP 理论

CAP:



C: Consistency 一致性

A: Availability 可用性 (指的是快速获取数据)

P: Tolerance of network Partition 分区容忍性 (分布式)

ACID:

ACID 事务提供以下几种保证:

Atomicity(原子性),事务中的所有操作,要么全部成功,要么全部不做。

Consistency(一致性),在事务开始与结束时,数据库处于一致状态。

Isolation(隔离性),事务如同只有这一个操作在被数据库所执行一样。

Durability(持久性),在事务结束时,此操作将不可逆转。

BASE:

Basically Available(基本可用)

Soft-state(软状态/柔性事务)

Eventual Consistency(最终一致性)

BASE 模型是传统 ACID 模型的反面,不同与 ACID,BASE 强调牺牲高一致性,从而获得可用性,数据允许在一段时间内的不一致,只要保证最终一致就可以了。

互联网 Web2.0 网站由于数据库存在高并发读写、高可扩展性、高可用性,所以要求设计成分布式存储,而在设计一个分布式存

储系统时,根据 CAP 理论,一致性(C),可用性(A),分区容错性(P),三者不可兼得,最多只能同时满足其中的两个。而关系型数据库保证了强一致性(ACID 模型)和高可用性,所以要想实现一个分布式数据库集群非常困难,这也解释了为什么关系型数据库的扩展能力十分有限。而 NoSQL 数据库则是通过牺牲强一致性,采用 BASE 模型,用最终一致性的思想来设计分布式系统,从而使得系统可以达到很高的可用性和扩展性。

对 Web2.0 网站来说,可用性与分区容忍性优先级要高于数据一致性,一般会尽量朝着 A、P 的方向设计,但事实上,数据库系统最大的优势就对一致性的保证,单纯为了 P(分布式)而放弃 C(一致性)也是不可取的,所以需要通过其它手段来保证对于一致性的商务需求。

3.2 NoSQL 数据库实际应用缺陷

缺乏强有力的商业支持:大部分的 NoSQL 数据库都是开源项目,没有世界级的数据库厂商提供完善的服务,如果出现故障,只能自己解决,风险较大。

成熟度不高:NoSQL 数据库的实际应用不多,大部分 NoSQL 产品都在小范围应用。成熟度不高,目前还很难在企业中广泛应用。

NoSQL 数据库在设计时难以体现实际,关系型数据库中的关系模型对于数据库设计是很有帮助的,而 NoSQL 数据库缺乏这种关系,难以体现业务的实际情况,对于数据库的设计与维护都增加了很大难度。

3.3 关系型数据库与 NoSQL 数据库结合

从上面的 CAP 理论来看,分布式存储系统更适合用 NoSQL 数据库,现有的 Web2.0 网站遇到的性能以及扩展性瓶颈也会迎刃而解,但是目前 NoSQL 数据库的实际应用缺陷又让我们难以放心。这时我们考虑是否可以将 NoSQL 数据库与关系型数据库结合使用,在强一致性(C),高可用性场景我们采用 ACID 模型,在高可用性和扩展性场景,我们就采用 BASE 模型,答案是肯定的,目前的 NoSQL 数据库还难以与关系型数据库一争高下,但它却可以对关系数据库在性能和扩展性上进行弥补,所以我们可以把 NoSQL 和关系数据库进行结合使用,各取所长,需要使用关系特性的时候我们使用关系数据库,需要使用 NoSQL 特性的时候我们使用 NoSQL 数据库,各得其所。

下面举个典型的例子加以说明:

在 Web2.0 网站中比较典型就是用户评论的存储,评论表大致可分为评论表主键 ID、被评论用户 ID、评论用户 ID、评论时间、评论内容等字段。结合关系型数据库与 NoSQL 数据库的特点,我们可将需要查询的字段,比如评论表主键 ID、被评论用户 ID、评论用户 ID、评论时间等数据、时间类型的小字段存储于关系型数据库中,根据查询建立相应的索引,而评论内容是个大文本字段,我们肯定不会通过文本内容进行查询,所以我们把评论内容存储在 NoSQL 数据库中。这种让关系型数据库专门负责处理擅长的关系存储, NoSQL 数据库作为数据的存储的结合使用方式,首先节省了关系型数据库的 IO 开销,提高了数据库的数据备份与恢复速度;其次由于 NoSQL 数据库的 Cache 往往都是行级别的,所以对评论内容字段也更容易做 Cache,最后由于 NoSQL 数据库天生就容易扩展,经过这种结合优化,关系型数据库的性能也能得到提高。这种结合方式实现起来比较容易,却能取得不错的效果。关系型数据库与 NoSQL 数据库结合并不局限于这种方式,应该根据具体的应用场景灵活组合使用。

4 结束语

关系数据库已经流行了几十年了, NoSQL 数据库想在短期内取而代之显然是不可能的,但是 NoSQL 数据库的发展势头也不容小觑。在当前阶段的某些场景下,可以将 NoSQL 数据库与关系型数据库结合使用,相互弥补各自的缺陷,这种数据库组合对解决目前 Web2.0 网站所遇到的性能、扩展性等问题具有指导意义。

参考文献:

- [1] 李莉莎.关于 NOSQL 的思考[J].中国传媒科技,2010(4):40-41.
- [2] 黄贤立.NOSQL 非关系型数据库的发展及应用初探[J].福建电脑,2010(7):30.