为什么关系型数据库不可替代

杨哲涵

Michael Stonebraker 和 Andrew Pavlo 认为,抛弃 SQL 和关系模型(RM)的想法不会成功,尽管在这个过程中可以为 SQL 引入新的特性,但关系型数据库的基本概念是不可替代的[1]. NoSQL 可以看成数据库发展上的一段弯路,如果用戶期望比文档模型,键值存储更多的功能,这些专有领域数据库便不得不加入 SQL 和 RM 的支持.

例如,在过去 20 年间,出现过设计糟糕,但因为响应了市场需求而成功的 DBMS(好营销对烂产品的影响),Stonebraker 提到 MySQL, MongoDB 的例子,说明尽管不遵循关系型数据库的设计原则,但是这些系统在特定的场景下是有用的.它们在商业上的成功为它们赢得时间弥补欠下的技术债务,例如 MongoDB 重新添加了 SQL 支持.这从另一个角度说明了 RDBMS 的优势,它们的设计是经过深思熟虑的,并且 RDBMS 经过扩展后可以覆盖更多的场景.

当前,由于 AI 的发展,数据库系统的需求也在发生变化,例如,在大数据分析中,用戶希望数据库能够支持向量嵌入功能. DB for AI 是一个新兴领域,但如作者所说,“What goes around comes around”,这些新的需求也可以通过扩展关系型数据库来实现.

文章最后,作者对数据库发展提出了展望.下一代数据库应该变得更加“模块化”,解析器这样的部件显然不应该重复开发.在模块化设计中,数据库的各个功能组件(如查询处理器,存储引擎,索引管理器等)可以独立开发和优化.这不仅能提高系统的可维护性和可扩展性,还能够通过组件化来满足不同应用场景的需求.

此外,这种模块化方法还促进了数据库技术的“去重复化”,减少了开发资源的浪费.通过共享和复用成熟的模块,如 SQL 解析器或者事务管理系统,可以更快地为数据库开发专有功能,向量嵌入,地理信息,时序数据,消息队列,全文检索等,而无需从零开始构建所有功能.这样,关系型数据库的核心优势(如 ACID 特性,SQL 支持)可以在新的应用场景下得到更好的发挥.

对于硬件的发展,GPU 硬件加速在数据库领域的作用主要体现在大幅提升数据查询和处理速度上.CPU 在面对大规模数据分析或复杂查询时,性能瓶颈显而易见.GPU 专门用于并行处理,在处理大量数据时更为高效,可以将数据聚合,排序和复杂的 SQL 查询并行化执行.此外,网络一直是数据库系统的瓶颈之一,数据库的大部分的 CPU 时间都花费在了 TCP/IP 协议栈上.一种新颖的解决方案是使用 RDMA(远程直接内存访问)技术,绕过 CPU,直接在网卡和内存之间传输数据,这样也就避免了 CPU 的内核网络栈的开销. 然而,云服务厂商并不会跟进最新的硬件技术,如果用戶希望应用硬件进步,可能遇到相当大的障碍,部署新硬件需要人力和时间,而且可能会导致应用程序的不稳定.因此,硬件的发展对数据库系统的增益是有限的.

最后,在技术之外,开源数据库正面临来自云服务厂商的巨大挑战.最初,开源数据库如 PostgreSQL 和 MySQL 通过免费的软件授权和低廉的硬件成本,挑战了以 Oracle 为代表的高昂商业数据库.然而,随着云计算的兴起,云服务厂商通过将开源数据库内核包装成云数据库服务(如 RDS),大幅提高了服务价格,垄断了开源数据库的使用和维护市场,还通过建立专家池攫取了软件生命周期中大部分价值,而将开发成本转嫁给开源社区.这样的做法不仅削弱了开源社区的人才供给和流动,还破坏了开源软件的生态循环,威胁到开源软件的可持续发展.而在国内,数据库的发展还面临开源社区和氛围不成熟,国产数据库重复开发,缺乏创新,缺乏社区支持等问题.

RMS 时代的自由软件与非自由软件之争,在今天以云服务与本地部署之争取代.云服务厂商通过

提供便捷的服务,吸引了大量用戶,但是用戶在使用云服务的同时,也失去了对软件的控制权.云服务厂商可以随时更改服务条款,提高价格,甚至停止服务,用戶无法控制.因此,用戶需要权衡云服务的便利性和软件的自由性,并在云服务和本地部署之间取得平衡.数据库系统的发展也需要关注这一点,保持数据库系统为任何用戶提供自由选择的权利.

参考文献

[1] M. Stonebraker and A. Pavlo, What Goes Around Comes Around... And Around..., SIGMOD Rec. **53**, 21 (2024)