RDMA技术研究综述

刘志锋,叶志伟,蔡敦波,钱　岭

(中移(苏州)软件技术有限公司 创新中心,江苏 苏州 215153)

摘　要: RDMA技术是一种直接内存访问技术,其可以实现将数据从一台计算机直接传递到另一台计算机而无需操作系统的介入,从而实现高吞吐、低延迟的网络通信。近年来,为了充分发挥RDMA的优势,RDMA通常与各类技术相结合以提升其可扩展性。将RDMA技术应用于数据库中,有助于提升数据库的高可用性。基于RDMA的新型存储技术可提高网络性能、存储扩展能力及易用性。在介绍RDMA实现原理的基础上,针对当前RDMA技术行业应用的最新研究进展,分析其实现的具体思路以及存在的优势,探索未来RDMA可进一步拓展的方向。

关键词: RDMA;分布式存储;数据库;新型存储

DOI: 10. 11907/rjdk. 212568　　开放科学(资源服务)标识码(OSID):

中图分类号: TP393　　**文献标识码:** A　　**文章编号:** 1672-7800(2022)012-0266-06

RDMA Technology Research: A Survey

LIU Zhi-feng, YE Zhi-wei, CAI Dun-bo, QIAN Ling

(*Innovation Center, China Mobile(Suzhou) Software Technology Co., Ltd., Suzhou* 215153, *China*)

Abstract: RDMA technology is a direct memory access technology, which can realize the direct transfer of data from one computer to another without the intervention of the operating system, thereby realizing high-throughput and low-latency network communication. In recent years, in order to give full play to the advantages of RDMA and improve scalability, RDMA is usually combined with various technologies and is widely used. The combination of RDMA technology and database technology can greatly improve the high availability performance of the database; new RDMA-based storage technology can improve network performance, Storage expansion capacity, improve ease of use. On the basis of introducing the principles of RDMA implementation, aiming at the latest research progress in the application of the current RDMA technology industry, analyzing the specific ideas and existing advantages of the research, and exploring the direction in which RDMA can be further expanded in the future.

Key Words: RDMA; distributed storage; database; novel storage

0　引言

随着高性能计算、人工智能、大数据分析以及物联网技术的高速发展,各类行业应用对于网络传输性能的要求越来越高,传统的TCP/IP协议架构存在数据处理和网络传输延迟较大、TCP/IP协议处理复杂等问题。RDMA(远程直接内存访问)技术是一种用于解决网络传输中服务器数据处理延迟的技术,其支持远程直接读写异地内存,且无需双方CPU和操作系统的介入[1-2]。相比传统基于TCP/IP协议的网络传输技术,RDMA具有高吞吐、低延迟的传输性能,占用系统资源更少,适用于大规模并行计算机集群。将RDMA与传统TCP/IP进行比较,如图1所示。在传统的TCP/IP模式下,两台服务器进行应用之间的数据传输,首先需要把数据从应用缓存拷贝到内核中的TCP协议栈进行缓存,然后到达驱动层,最后拷贝到网卡缓存中,利用网络传输实现服务器间的数据包收发。在RDMA模式下,用户应用中的数据直接传入本机的存储区,数据通过网络从本地系统快速传输到远程系统内存,全程对操作系统没有任何影响,因此不需要太多的计算机处理能力。其消除了

收稿日期: 2021-11-16

基金项目: 中国移动应用基础研究项目(R21101H8)

作者简介: 刘志锋(1994-),男,硕士,中移(苏州)软件技术有限公司创新中心工程师,研究方向为数据库;叶志伟(1987-),男,硕士,中移(苏州)软件技术有限公司创新中心工程师,研究方向为数据库;蔡敦波(1981-),男,博士,中移(苏州)软件技术有限公司创新中心副教授,研究方向为知识工程、智能规划与调度;钱岭(1975-),男,博士,中移(苏州)软件技术有限公司创新中心高级工程师,研究方向为云计算、大数据。本文通讯作者:钱岭。

外部内存复制和上下文切换的开销,从而释放了内存带宽,并缩短了CPU周期,以提高应用程序系统性能。

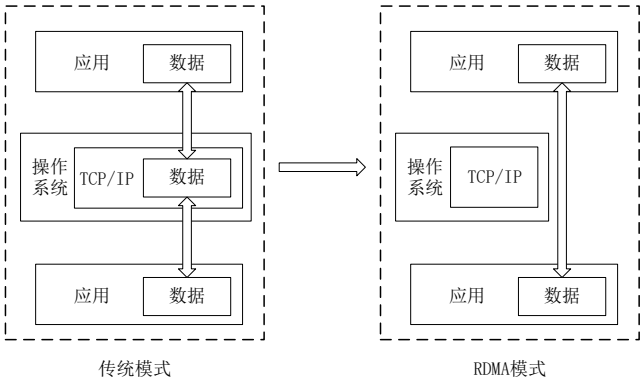

Fig. 1　Comparison of RDMA and traditional TCP/IP

图1　RDMA与传统TCP/IP比较

RDMA技术的发展大大降低了机器之间数据传输的代价。近年来,RDMA技术得到了广泛研究与应用,基于RDMA的分布式存储技术为满足分布式键值存储,已广泛用于可扩展的行业数据管理解决方案设计中。RDMA通过与分布式存储相结合,一方面在硬件方面加强资源的高效管理,另一方面在软件层面加强软硬件的耦合设计,大大提升了分布式存储性能[3]。在键值存储方向,FaRM基于RDMA构建分布式共享内存,使用RDMA直接访问共享地址空间中的数据,从而实现消息的快速传递,最后通过构建RDMA感知键值和图形存储以获得最佳RDMA性能[4];Pilaf基于RDMA提供了一种旨在实现高性能和低CPU使用的键值分布式内存键值存储方式[5];HERD同样提供了一个旨在充分利用RDMA网络的键值系统,区别于FaRM和Pliaf设计的键值系统,HERD的设计重点在于减少网络往返次数,同时使用高效的RDMA原语,从而显著降低延迟和吞吐量[6]。DrTM设计了一种内存事务处理系统,其利用HTM的强原子性和RDMA的强一致性,提供更高数量级的内存事务处理吞吐量和更低的延迟[7]。在DrTM的基础上,DrTM+R提供了一个快速通用的分布式事务处理系统[8]。DrTM+R通过利用电池后备内存作为数据库记录的主要存储,并结合HTM和RDMA实现快速分布式处理,从而支持内存中的事务交易。与DrTM不同的是,DrTM+R对事务性工作负载没有限制,为高可用性提供了完整的复制支持。FaSST是基于FaRM的分布式事务协议模型,其可提供具有可序列化和持久性的分布式内存事务,性能远优于FaRM和DrTM[9]。

当前,RDMA技术同样被引入到数据库设计中,利用高速网络和RDMA进行数据传输,让I/O性能不再成为瓶颈,从而大大提升了数据库性能[10]。Lu等[11]提出一种支持RMDA的MongoDB设计方案,考虑到通信部分占整体延迟的比例很高,基于RDMA的设计方案提供了更低的延迟和更高的吞吐量,利用InfiniBand等高性能网络来提高MongoDB的性能。在POLARDB中,计算节点与存储节点的通信设计机制引入高速RDMA网络实现计算节点与存储节点的高速通信[12]。Fent等[13]提出一种低等级、低延迟的消息库(L5),其是一种用于数据库系统的高性能通信层,通过使用RDMA(InfiniBand)、RoCE(以太网)和共享内存(IPC)等进行互连,L5可在很大程度上消除数据库系统的网络瓶颈。

此外,随着新型存储设备的逐渐普及并逐步替代传统磁盘,为充分发挥PM(Persistent Memory)、NVMe SSD等新型存储设备的新特性,将新型存储技术与RDMA相结合构建更高速的存储系统是当前新的研究方向[14]。FlatStore中设计了基于PM的KV存储引擎,设计的关键思想是将KV存储解耦为快速索引的易失性索引和高效存储的持久日志结构[15]。Octopus是一个支持RDMA的分布式持久内存文件系统,通过抽象共享持久内存池来减少数据传输中的冗余内存副本[16]。Hotpot提供了一种可将来自不同节点的PM在全局地址空间中进行管理的技术,并结合数据复制来支持容错[17]。AsymNVM是非对称分解非易失性存储器的通用框架,其实现了构建远程数据结构的基本原语,包括空间管理、并发控制、崩溃一致性和复制[18]。Liu等[19]提出的连续性哈希是一种基于RDMA与PM的合并哈希解决方案,连续性哈希支持通过单个单边RDMA操作进行高效远程读取,并为PM上的所有写操作提供无日志一致性保证。Erda是一种零拷贝日志结构内存设计方法,其具备数据远程高效传输的原子性,可用于解决在远程数据传输中需要消耗额外的网络往返、远程CPU参与和双NVM写入等问题[20]。

本文首先对RDMA技术的实现形式及通信原理作简要叙述,然后介绍最新的RDMA行业研究进展,针对几种应用场景,分别对几个典型的适配和优化技术进行具体分析,最后给出技术总结,并指出今后可继续开展的研究方向,旨在为设计者和开发者进行RDMA技术拓展提供参考。

1　RDMA技术原理

1.1　RDMA实现方式

RDMA支持3种网络协议:InfiniBand(IB)[21]、RoCE[22]和iWARP[23-24]。基于IB架构的RDMA是一种较早提出的原生支持RDMA的新一代网络协议,其搭载在专用的IB网卡IB交换机上,提供了基于通道的点对点消息队列转发模型,每个应用都可通过创建的虚拟通道直接获取本应用的数据消息,而无需协议栈及操作系统的介入。RoCE是基于以太网的RDMA技术,即RDMA over Ethernet,支持在以太网上承载IB协议。RoCE与IB具有相同的传输控制层和软件应用层,两者的区别在于网络链路层和以太网链路层。RoCE协议分为v1和v2两个版本,v1基于以太网承载RDMA,不支持跨网络传输,v2由以太网TCP/IP协议中的

UCP层实现。iWARP是基于以太网和TCP/IP协议的RDMA技术，即RDMA over TCP，一种允许通过TCP执行RDMA的网络协议，可运行在标准以太网基础设施上，但IB和RoCE中有些功能在iWARP中不受支持。

1.2 RDMA通信原理

RDMA网卡内部维护了3个工作队列：发送队列(SQ)、接收队列(RQ)和完成队列(CQ)。RDMA数据传输示例如图2所示，以RC模式进行点对点建链过程为例，当RDMA应用程序开始工作时，通信双方首先在RDMA网卡中创建SQ、RQ和CQ 3个队列，然后对内存中用于处理的区域进行注册。RDMA为每个连接提供一个工作队列(QP)负责调度工作，由一个SQ和一个RQ构成，同在一个QP上的数据传输遵循相同路径。在QP中，工作队列元素(WQE)作为指令放置指向缓冲区的指针，放置在SQ上的WQE包含指向要发送消息的指针，放置在RQ上的WQE包含指向要接收消息的指针。完成队列(CQ)用于传输完成后通知应用程序。每一个WQE完成时，将创建完成队列元素(CQE)放入CQ中。

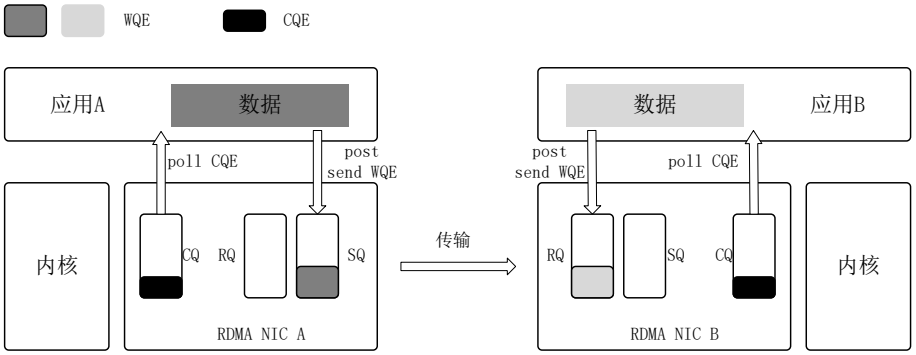

Fig. 2 RDMA data transmission example

图2 RDMA数据传输示例

在RDMA编程中，网卡驱动含有用户态编程接口和内核态编程接口，称为Verbs[25]。RDMA Verbs定义了两种通信原语：消息语义和内存语义。消息语义又称双边原语，用于消息通信。典型的消息语义包括Send和Recv指令，类似于套接字编程中的Send和Recv。接收方提前执行Recv指令用于提前指定接收发送端数据时存储的地址，发送方执行Send指令向接收方发送数据。在传输过程中，这些动作均涉及到响应者的CPU，通常适用于小数据传输。内存语义又称单边原语，是一个新的通信模型。典型的内存语义包括Read和Write指令，先在本地指定好要直接访问远端机器的内存地址，执行内存语义原语直接访问远端内存，远端机器CPU无需介入。通过将本地内存总线延伸到其它机器的方式可获得更高的传输效率，适用于较大数据的传输。

当前RDMA可支持有连接和无连接的网络传输，此外还支持可靠传输与不可靠传输。对于有连接传输，提供可靠连接(RC)和不可靠连接(UC)两种传输形式；对于无连接传输，只支持不可靠数据报(UD)这一种实现方式[26]。RC模式可确保报文正确传输到目的端，支持报文ack确认和重传。UC模式需要提前建链，报文不需要携带地址信息，不支持ack确认和重传，不保证目的端能正确接收，最大传输单元为2KB。UD模式不需要建链，不支持ack确认和重传，每个报文都含有目标地址和目标队列信息，最大传输单元限制为4KB。RC和UC模式都只支持点对点的面向连接的数据传输，UD模式不仅可支持点对点，而且可支持点对多的数据传输。

不同传输方式支持的RDMA verbs也所有区别，如表1所示。

Table 1 Comparison of RDMA Verbs under different transmission modes

表1 不同传输方式下的RDMA Verbs比较

	Send/Recv	Write	Read
RC	√	√	√
UC	√	√	×
UD	√	×	×

2 研究现状

随着大规模分布式系统和数据中心存在的网络性能瓶颈越来越明显，RDMA作为一种高效网络传输的硬件设备，能够帮助分布式存储和分布式数据库等网络密集型业务实现更高的吞吐和更低的时延。RDMA具有低延迟访问特性，该特性同样可在分布式系统设计中用于对本地与远程的性能权衡，为数据持久化提供了新思路。

2.1 基于RDMA的分布式存储技术

FaRM技术基于RDMA构建分布式共享内存，将应用程序数据存储在主内存中，此外还执行应用程序线程。集群中所有机器的内存都公开为一个共享地址空间，FaRM使用单侧RDMA读来直接访问数据，使用RDMA写来实现快速消息传递原语。FaRM的共享地址空间由许多2GB共享内存区域组成，这些区域是地址映射单元、恢复单元以及RDMA向NIC注册的单元。

FaRM支持分布式事务作为确保一致性的通用机制，使用乐观并发控制和两阶段提交以确保严格的可序列化。由于分布式事务对于性能关键操作来说可能过于昂贵，

FaRM提供了单机事务和无锁只读操作两种机制以实现良好的性能。应用程序可通过将事务访问对象并置在同一主节点和相同副本上,并将事务传送到主节点来使用单机事务。此外,在需要时可使用两种机制来提高性能,包括RDMA上的无锁读取,以及支持并置对象和函数传递以保证使用高效的单机事务。单机事务通过减少消息数量和进一步减少由于锁引起的延迟以提高性能。最终,通过构建RDMA感知键值和图形存储以证明这些技术的有效性。结果表明,FaRM表现良好,其始终相比在同一物理网络上使用TCP/IP的主内存系统能实现更好的吞吐量,并延迟一个数量级。

DrTM+R是一个快速通用的分布式事务处理系统,其保留了高级硬件功能带来的性能优势,同时通过复制支持一般事务的工作负载和高可用性。DrTM+R通过设计混合OCC和锁定方案解决了通用性问题,该方案利用HTM(硬件事务存储)的强原子性与RDMA的强一致性来保持严格的可序列化和高性能。为解决由HTM事务更新记录的即时可见性和此类记录未就绪复制之间的竞争条件问题,DrTM+R利用了一种乐观复制方案,该方案使用类似seqlock的版本控制来区分元组的可见性,并记录复制的准备情况。使用TPC-C和SmallBank等典型OLTP工作负载进行的评估表明,DrTM+R在6节点集群上的扩展性很好,并且在没有复制的情况下,TPC-C和SmallBank分别实现了每秒超过5.69及9 400万笔交易。在DrTM+R上启用3路复制,在达到网络瓶颈前最多只会产生41%的开销,并且仍然比最先进的分布式事务系统(Calvin)快一个数量级。

2.2　RDMA与数据库相结合

Lu等[11]提出一种支持RMDA的MongoDB设计,该技术在传输层中通过RDMA接口替换MongoDB的Boost.Asio API操作实现套接字兼容。服务器端和客户端的每个会话都保留一个已注册的内存块,并设计了一种新颖的动态注册内存管理策略,以应对消息长度在某个连接中不会发生急剧变化的现象,最终实现内存的高效使用。由于常见的消息大小一般小于1KB,每个会话在QP连接初始化时会分配并注册一个专用的1KB内存块。当要发送的消息大小超过现有的专用内存大小时,该设计方案会把内存块大小重新调整为大于或等于消息大小的最小2次方。当几个连续消息未使用现有内存的50%时,其将释放一半内存以避免浪费。基于该策略可获得与MongoDB原生方法近似的性能,以及更高的内存使用率。

L5是一种Low-Level、Low Latency消息传递库,其取代了传统的套接字,并且可以透明地使用RDMA(InfiniBand)、RoCE(以太网)或共享内存(IPC)作为通信通道。对于InfiniBand上的远程通信以及同一台机器上的隔离进程之间,L5将吞吐量和延迟提高了一个数量级以上。数据库服务器中的一个常见模式是一个数据库服务器处理来自多个客户端的小请求。在此模式下,数据库服务器可以有许多打开的连接,但只有少数处于活动状态。L5旨在减轻这种非对称模式对数据库服务器的压力。对于客户端请求数据,客户端需要两个RDMA写工作请求:第一个写消息(SQL语句),第二个设置标志。由于可靠的RDMA连接顺序可以确保在设置标志之前,消息数据已经完全写入,因此服务器永远不会看到不完整的消息,从而保证了一致性。L5提供了一个构建块,通过统一的基于消息的通信框架来加速类似ODBC的接口。结果表明,L5提供了小负载的高交易吞吐量,以及用于自适应选择最佳可用通信技术的统一接口,在少量客户端的业务场景下可获得较好性能。此外,L5提供了非常低的延迟,而不依赖于应用程序级别的批处理。

2.3　与新型存储技术结合

当前,随着新型存储技术的发展,新型存储设备如PM、NVMe SSD等逐渐替代了传统磁盘。将RDMA技术与新型存储介质的新特性结合起来,构建一个高速的存储系统是当前众多学者研究的热点方向。

在云计算领域,为了提供高性能的网络传输,网络中通常将RDMA(远程直接内存访问)与终端系统中的NVM(非易失性内存)相结合。由于没有CPU的参与,单边RDMA访问远程内存变得更加高效,而NVM技术具有非易失性、字节可寻址性以及类似DRAM的延迟等优势。为了实现端到端的高性能,许多研究旨在协同片面的RDMA和NVM。由于需要保证远程数据的原子性(RDA),必须消耗额外的网络往返,而且需要远程CPU参与和双NVM写入。因此,一种称为Erda的零拷贝日志结构内存设计方法被提出,其具备数据远程高效传输的原子性。在Erda中,客户端通过单侧RDMA写入,直接将数据传输到服务器的目标地址,而无需冗余复制和远程CPU消耗。为了检测获取数据的完整性,在没有客户端—服务器协调的情况下验证校验和。通过利用哈希表中的8字节原子更新进一步确保元数据的一致性,哈希表还包含陈旧数据的地址信息。当发生故障时,服务器正确恢复到一致版本。实验结果表明,与Redo Logging(CPU参与方案)和Read After Write(网络主导方案)相比,Erda减少了约50%的NVM写入,同时显著提高了吞吐量,并降低了延迟。

现有的分布式文件系统严格隔离文件系统和网络层,并且大量分层的软件设计使得高速硬件没有得到充分利用。因此,一个支持RDMA的分布式持久内存文件系统Octopus被提出,其通过紧密耦合NVM与RDMA功能来重新设计文件系统内部机制。对于数据操作,Octopus直接访问共享的持久内存池以减少内存复制开销,并在客户端主动获取和推送数据以重新平衡服务器与网络之间的负载。对于元数据操作,Octopus引入了自我识别的RPC用于文件系统与网络之间的即时通知,以及一致性的高效分布式事务机制。评估结果表明,Octopus几乎可实现大型I/

O 的原始带宽，并且相比于现有的分布式文件系统，其性能优势要高出几个数量级。

2.4 研究现状分析

上述基于 RDMA 的应用优化方案本质上是通过合理地选择通信原语、降低注册开销、减少直接内存访问次数等方法来实现。与传统以太网通信方式不同，RDMA 提供了 Read/Write 和 Send/Recv 等远程直接访问的单向原语。考虑到 CPU、内存资源消耗以及传输性能等各方面因素的影响，需要合理权衡 Read/Write 或 Send/Recv 通信原语的选择。对于无需内存注册的小数据传输场景，一般采用 Send/Recv 通信原语即可获得较好的性能。在通常情况下，使用基于 Read/Write 通信原语进行数据传输需要考虑对相关内存进行预注册，通过一次性注册较大的区域可有效降低内存注册的开销。由于 RDMA NIC 每次进行 DMA 操作需要占用主机总线带宽，当应用程序向一个 QP 发出多个 WQE 时，NIC 可通过一次 DMA 操作将多个 WQE 批量地放到缓存中，从而减少 DMA 次数。此外，上述对网络负载均衡性能和数据组织与索引方式的优化，同样也实现了对 RDMA 系统的优化。

当前 RDMA 网卡和处理器均具有独立的缓存系统，但受制于有限的 RDMA 网卡缓存空间，如何高效管理缓存空间将会很大程度上影响系统工作性能及其扩展性。此外，RDMA 网卡具有良好的并行性，在多核处理器环境下，如何处理好数据传输并实现性能的最大化，依然有很大的优化空间。

3 总结与展望

本文对 RDMA 技术的基本原理进行了简要叙述，并结合当前 RDMA 技术在当前行业应用的研究，详细阐述了 RDMA 在分布式存储、数据库和新型存储技术等方面具有代表性的研究成果。

目前 RDMA 技术大多应用于数据中心，为上层高性能计算提供服务。未来的分布式系统除直接利用 RDMA 技术外，还可依赖于定制化的硬件以进一步提升性能。如今部分 RDMA 网卡还提供了 NVMoF（NVMe over fabrics）新特性，可使用 RDMA 直接操作固态存储设备以加速分布式日志系统的日志传输。

未来，可基于 RDMA 和 NVM 设计新型的分布式事务系统，充分利用单边和双边 RDMA 强大功能来促进事务执行，同时在并发控制和复制协议设计中大量使用 RDMA 和 NVM，可为 RDMA-NVM 分布式数据库带来极大的性能改进空间。此外，还可使用乐观并发控制（OCC）来确保严格的序列化和主备份复制以实现高可用性。通过将 NVM 组织为分布式共享内存池，用于存储数据库记录和事务日志。采用四阶段来执行一个事务，每个阶段均使用 RDMA 与 NVM 交互：执行时使用单边 RDMA READ 读取存储在 NVM 中的记录；验证时使用单边 RDMA 比较和交换（CAS）来获取与记录位于同一位置的锁；日志记录时使用单边 RDMA WRITE 将事务更新复制到备份；Commit 时使用双边 RDMA 来更新和解锁记录。当前，基于 RDMA 网络和持久存储构建高效的分布式存储系统已成为新的热门研究方向，这也为大数据处理与存储带来了新机遇。

参考文献：

[1] JIN H, YANG H Z. Summary of research on RDMA network transmission technology[J]. Technology Wind, 2020(18):137.
金浩，杨洪章. RDMA 网络传输技术研究综述[J]. 科技风，2020(18):137.

[2] LI L F, SHI Y C, WANG J F, et al. Hardware accelerated dynamic RDMA method for gigabit Ethernet [J]. Journal of University of Electronic Science and Technology of China, 2018, 47(5):672-679.
李龙飞，史阳春，王剑峰，等. 面向千兆以太网的动态 RDMA 通信方法[J]. 电子科技大学学报，2018，47(5):672-679.

[3] WEI X, CHEN R, CHEN H, et al. Optimizing distributed systems with remote direct memory access[J]. Big Data Research, 2018, 4(4):3-14.

[4] DRAGOJEVIĆ A, NARAYANAN D, CASTRO M, et al. FaRM: fast remote memory[C]// Proceedings of the 11th USENIX Conference on Networked Systems Design and Implementation, 2014: 401-414.

[5] MITCHELL C, GENG Y, LI J. Using one-sided RDMA reads to build a fast, CPU-efficient key-value store. [C]// Proceedings of the 2013 USENIX Conference on Annual Technical Conference, 2013: 103-114.

[6] KALIA A, KAMINSKY M, ANDERSEN D G. Using RDMA efficiently for key-value services[J]. ACM SIGCOMM Computer Communication Review, 2014, 44(4):295-306.

[7] WEI X, SHI J, CHEN Y, et al. Fast in-memory transaction processing using RDMA and HTM[C]// Proceedings of the 25th Symposium on Operating Systems Principles, 2015:87-104.

[8] CHEN Y, WEI X, SHI J, et al. Fast and general distributed transactions using RDMA and HTM[C]//Proceedings of the Eleventh European Conference on Computer Systems, 2016: 1-17.

[9] KALIA A, KAMINSKY M, ANDERSEN D G. FaSST: fast, scalable and simple distributed transactions with two-sided (RDMA) datagram RPCs[C]// Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation, 2016:185-201.

[10] CAO W, ZHANG Y, YANG X, et al. PolarDB serverless: a cloud native database for disaggregated data centers[C]// SIGMOD/PODS '21: International Conference on Management of Data, 2021:2477-2479.

[11] LU F, FANG T, ZHANG Z, et al. Improving the performance of MongoDB with RDMA[C]//2019 IEEE 21st International Conference on High Performance Computing and Communications, 2019: 1004-1010.

[12] CAO W, LIU Y, CHENG Z, et al. POLARDB meets computational storage: efficiently support analytical workloads in cloud-native relational database[C]//18th USENIX Conference on File and Storage Technologies, 2020: 29-41.

[13] FENT P, VAN RENEN A, KIPF A, et al. Low-latency communication for fast DBMS using RDMA and shared memory[C]//2020 IEEE 36th International Conference on Data Engineering, 2020: 1477-1488.

[14] JIN H, MEI J J. Research on a method of RDMA transmission control [J]. Network Security Technology & Application, 2020(5): 19-20.
金浩,梅君君. 一种RDMA传输控制方法研究[J]. 网络安全技术与应用, 2020(5): 19-20.

[15] CHEN Y, LU Y, YANG F, et al. FlatStore: an efficient log-structured key-value storage engine for persistent memory [C]//Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems, 2020: 1077-1091.

[16] LU Y, SHU J, CHEN Y, et al. Octopus: an RDMA-enabled distributed persistent memory file system[C]//2017 USENIX Annual Technical Conference, 2017: 773-785.

[17] SHAN Y, TSAI S Y, ZHANG Y. Distributed shared persistent memory [C]//Proceedings of the 2017 Symposium on Cloud Computing, 2017: 323-337.

[18] MA T, ZHANG M, CHEN K, et al. AsymNVM: an efficient framework for implementing persistent data structures on asymmetric NVM architecture[C]//Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems, 2020: 757-773.

[19] LIU X, HUA Y, BAI R. Consistent RDMA-Friendly hashing on remote persistent memory[DB/OL]. https://arxiv.org/abs/2107.06836.

[20] LIU X, HUA Y, LI X, et al. Write-optimized and consistent RDMA-based NVM systems[DB/OL]. https://arxiv.org/abs/1906.08173.

[21] GUO C, WU H, DENG Z, et al. RDMA over commodity ethernet at scale [C]// Florianopolis: Proceedings of the 2016 ACM SIGCOMM Conference, 2016.

[22] MACARTHUR P, QIAN L, RUSSELL R D, et al. An integrated tutorial on InfiniBand, Verbs, and MPI[J]. IEEE Communications Surveys & Tutorials, 2017, 19(4):2894-2926.

[23] GUO Z, LIU S, ZHANG Z L. Traffic control for RDMA-enabled data center networks: a survey[J]. IEEE Systems Journal, 2019, 14(1): 677-688.

[24] RECIO R, METZLER B, CULLEY P, et al. A remote direct memory access protocol specification[J]. Journal of Neurophysiology, 2007, 93(1):467-480.

[25] DAI C, LIU Q, JIANG J H, et al. Survey on RDMA virtualization technology[J]. Computer Systems & Applications, 2020, 29(10):1-8.
代超,刘强,蒋金虎,等. RDMA虚拟化相关技术研究[J]. 计算机系统应用, 2020, 29(10):1-8.

[26] CHEN Y M, LU Y Y, LUO S M, et al. Survey on RDMA-based distributed storage systems[J]. Journal of Computer Research and Development, 2019, 56(2):227-239.
陈游旻, 陆游游, 罗圣美,等. 基于RDMA的分布式存储系统研究综述[J]. 计算机研究与发展, 2019, 56(2):227-239.

(责任编辑:黄　健)