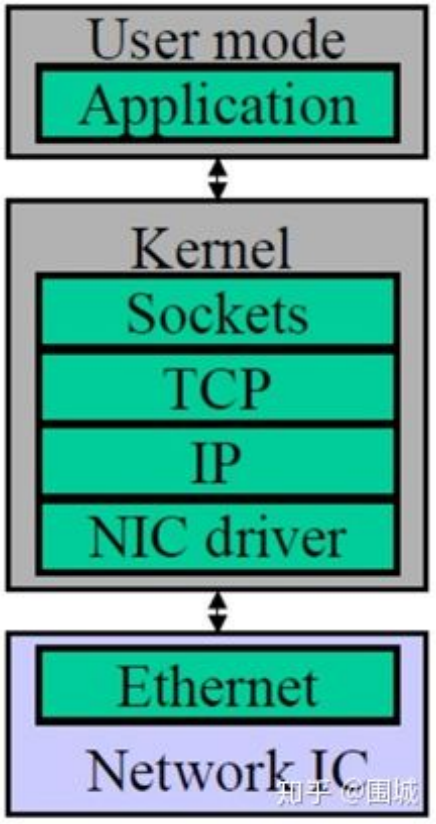
深入浅出全面解析RDMA

围城
欢迎大家follow：https://github.com/Tjcug

RDMA(RemoteDirect Memory Access)技术全称远程直接内存访问，就是为了解决网络传输中服务器端数据处理的延迟而产生的。它将数据直接从一台计算机的内存传输到另一台计算机，无需双方操作系统的介入。这允许高吞吐、低延迟的网络通信，尤其适合在大规模并行计算机集群中使用。RDMA通过网络把资料直接传入计算机的存储区，将数据从一个系统快速移动到远程系统存储器中，而不对操作系统造成任何影响，这样就不需要用到多少计算机的处理能力。它消除了外部存储器复制和上下文切换的开销，因而能解放内存带宽和CPU周期用于改进应用系统性能。

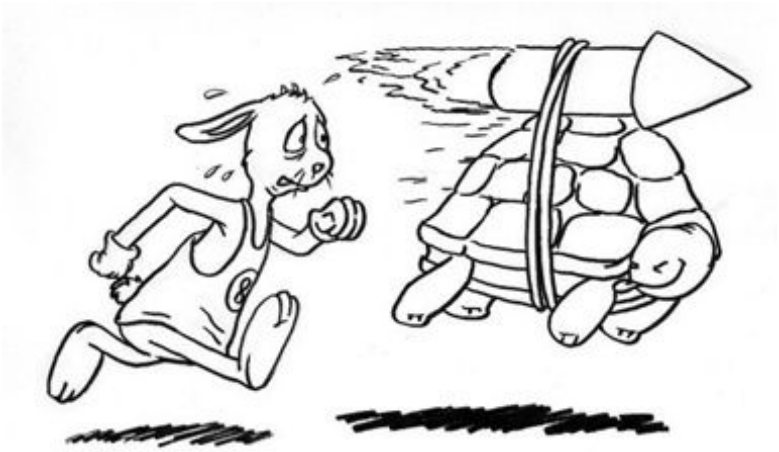
本次详解我们从三个方面详细介绍RDMA：RDMA背景、RDMA相关工作、RDMA技术详解。

一、背景介绍



1.1 传统TCP/IP通信模式

传统的TCP/IP网络通信，数据需要通过用户空间发送到远程机器的用户空间。数据发送方需要讲数据从用户应用空间Buffer复制到内核空间的Socket Buffer中。然后Kernel空间中添加数据包头，进行数据封装。通过一系列多层网络协议的数据包处理工作，这些协议包括传输控制协议（TCP）、用户数据报协议（UDP）、互联网协议（IP）以及互联网控制消息协议（ICMP）等。数据才被Push到NIC网卡中的Buffer进行网络传输。消息接受方接受从远程机器发送的数据包后，要将数据包从NIC buffer中复制数据到Socket Buffer。然后经过一些列的多层网络协议进行数据包的解析工作，解析后的数据被复制到相应位置的用户应用空间Buffer，这个时候再进行系统上下文切换，



如今随着社会的发展，我们希望更快和更轻量级的网络通信。

1.2 通信网络定义

计算机网络通信中最重要两个衡量指标主要是指高带宽和低延迟。通信延迟主要是指：处理延迟和网络传输延迟。处理延迟开销指的就是消息在发送和接收阶段的处理时间。网络传输延迟指的就是消息在发送和接收方的网络传输时延。如果网络通信状况很好的情况下，网络基本上可以 达到高带宽和低延迟。

1.3 当今网络现状

当今随着计算机网络的发展。消息通信主要分为两类消息，一类是Large messages，在这类消息通信中，网络传输延迟占整个通信中的主导位置。还有一类消息是Small messages，在这类消息通信中
场景中

位。具体来说，处理开销指的是buffer管理、在不同内存空间中消息复制、以及消息发送完成后的系统中断。

1.4 传统TCP/IP存在的问题

传统的TPC/IP存在的问题主要是指I/O bottleneck瓶颈问题。在高速网络条件下与网络I/O相关的主机处理的高开销限制了可以在机器之间发送的带宽。这里感兴趣的高额开销是数据移动操作和复制操作。具体来讲，主要是传统的TCP/IP网络通信是通过内核发送消息。Messaging passing through kernel这种方式会导致很低的性能和很低的灵活性。性能低下的原因主要是由于网络通信通过内核传递，这种通信方式存在的很高的数据移动和数据复制的开销。并且现如今内存带宽性相较如CPU带宽和网络带宽有着很大的差异。很低的灵活性的原因主要是所有网络通信协议通过内核传递，这种方式很难去支持新的网络协议和新的消息通信协议以及发送和接收接口。

二、相关工作

高性能网络通信历史发展主要有以下四个方面：TCP Offloading Engine（TOE）、User-Net Networking(U-Net)、Virtual interface Architecture（VIA）、Remote Direct Memroy Access(RDMA)。U-Net是第一个跨过内核网络通信的模式之一。VIA首次提出了标准化user-level的网络通信模式，其次它组合了U-Net接口和远程DMA设备。RDMA就是现代化高性能网络通信技术。

2.1 TCP Offloading Engine

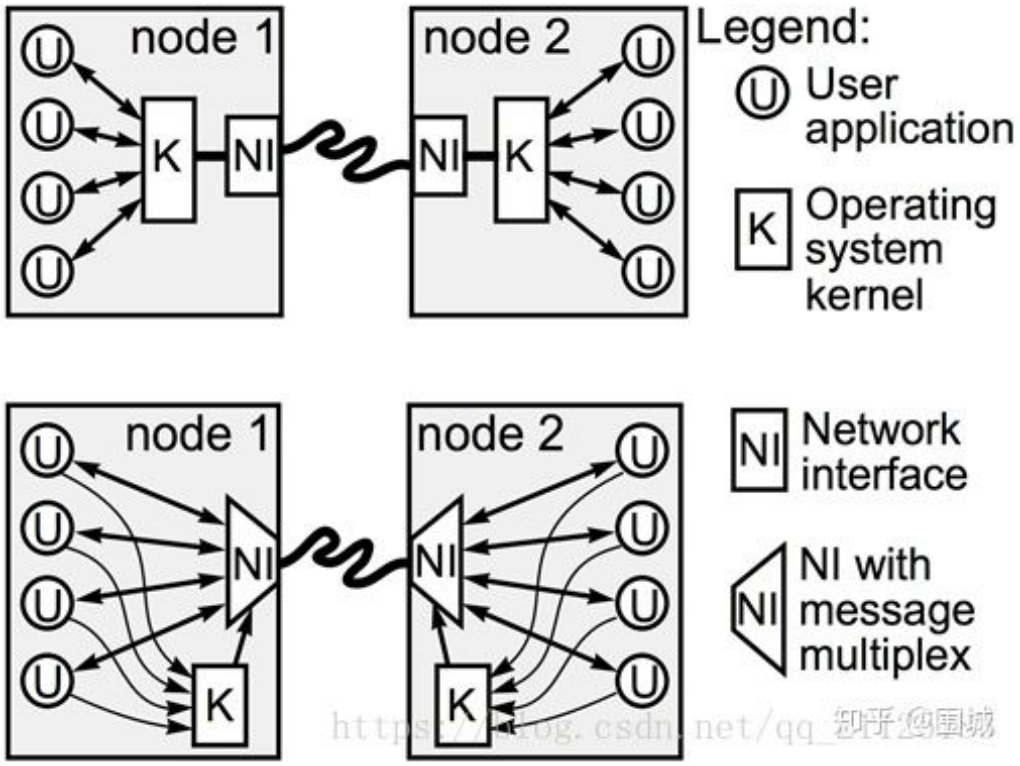
在主机通过网络进行通信的过程中，主机处理器需要耗费大量资源进行多层网络协议的数据包处理工作，这些协议包括传输控制协议（TCP）、用户数据报协议（UDP）、互联网协议（IP）以及互联网控制

部分主机处理器资源解放出来专注于其他应用，人们发明了TOE（TCP/IP Offloading Engine）技术，将上述主机处理器的工作转移到网卡上。

这种技术需要特定网络接口-网卡支持这种Offloading操作。这种特定网卡能够支持封装多层网络协议的数据包，这个功能常见于高速以太网接口上，如吉比特以太网（GbE）或10吉比特以太网（10GbE）。

2.2 User-Net Networking(U-Net)

U-Net的设计目标是将协议处理部分移动到用户空间去处理。这种方式避免了用户空间将数据移动和复制到内核空间的开销。它的设计宗旨就是移动整个协议栈到用户空间中去，并且从数据通信路径中彻底删除内核。这种设计带来了高性能的提升和高灵活性的提升。

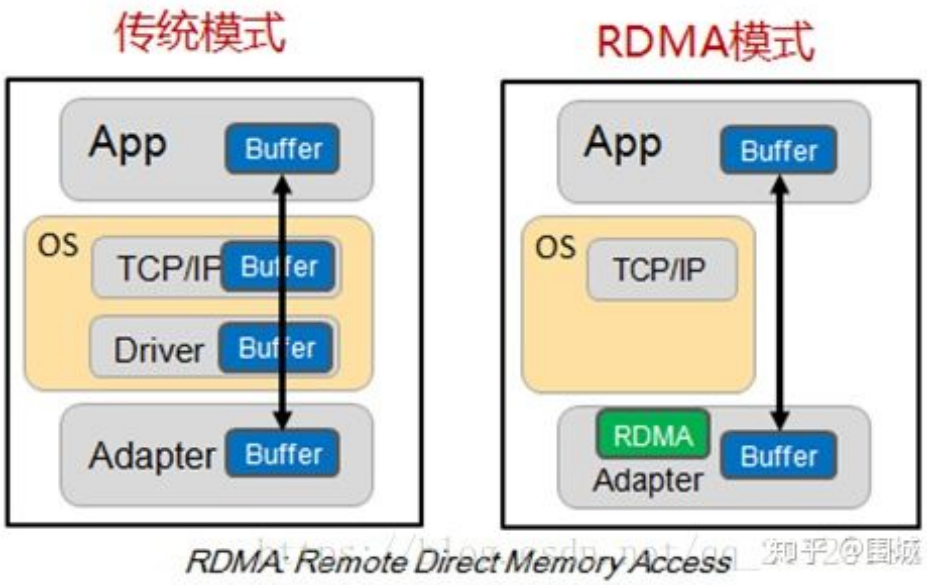


U-Net的virtual NI 为每个进程提供了一种拥有网络接口的错觉，内核接口只涉及到连接步骤。传统上的网络，内核控制整个网络通信，所有的通信都需要通过内核来传递。U-Net应用程序可以通过MUX直接访问网络，应用程序通过MUX直接访问内核，而不需要将数据移动和复制到内核空间中去。

三、RDMA详解

RDMA(Remote Direct Memory Access)技术全称远程直接内存访问，就是为了解决网络传输中服务器端数据处理的延迟而产生的。RDMA通过网络把资料直接传入计算机的存储区，将数据从一个系统快

机的处理功能。它消除了外部存储器复制和上下文切换的开销，因而能解放内存带宽和CPU周期用于改进应用系统性能。



RDMA主要有以下三个特性：1.Low-Latency 2.Low CPU overhead 3. high bandwidth

3.1 RDMA 简介

Remote：数据通过网络与远程机器间进行数据传输

Direct：没有内核的参与，有关发送传输的所有内容都卸载到网卡上

Memory：在用户空间虚拟内存与RNIC网卡直接进行数据传输不涉及到系统内核，没有额外的数据移动和

Access：send、receive、read、write、atomic操作

3.2 RDMA基本概念

RDMA有两种基本操作。

1. Memory verbs: 包括RDMA read、write和atomic操作。这些操作指定远程地址进行操作并且绕过接收者的CPU。
2. Messaging verbs:包括RDMA send、receive操作。这些动作涉及响应者的CPU，发送的数据被写入由响应者的CPU先前发布的接受所指定的地址。

RDMA传输分为可靠和不可靠的，并且可以连接和不连接的（数据报）。凭借可靠的传输，NIC使用确认来保证消息的按序传送。不可靠的传输不提供这样的保证。然而，像InfiniBand这样的现代RDMA实现使用了一个无损链路层，它可以防止使用链路层流量控制的基于拥塞的损失[1]，以及使用链路层重传的基于位错误的损失[8]。因此，不可靠的传输很少会丢弃数据包。

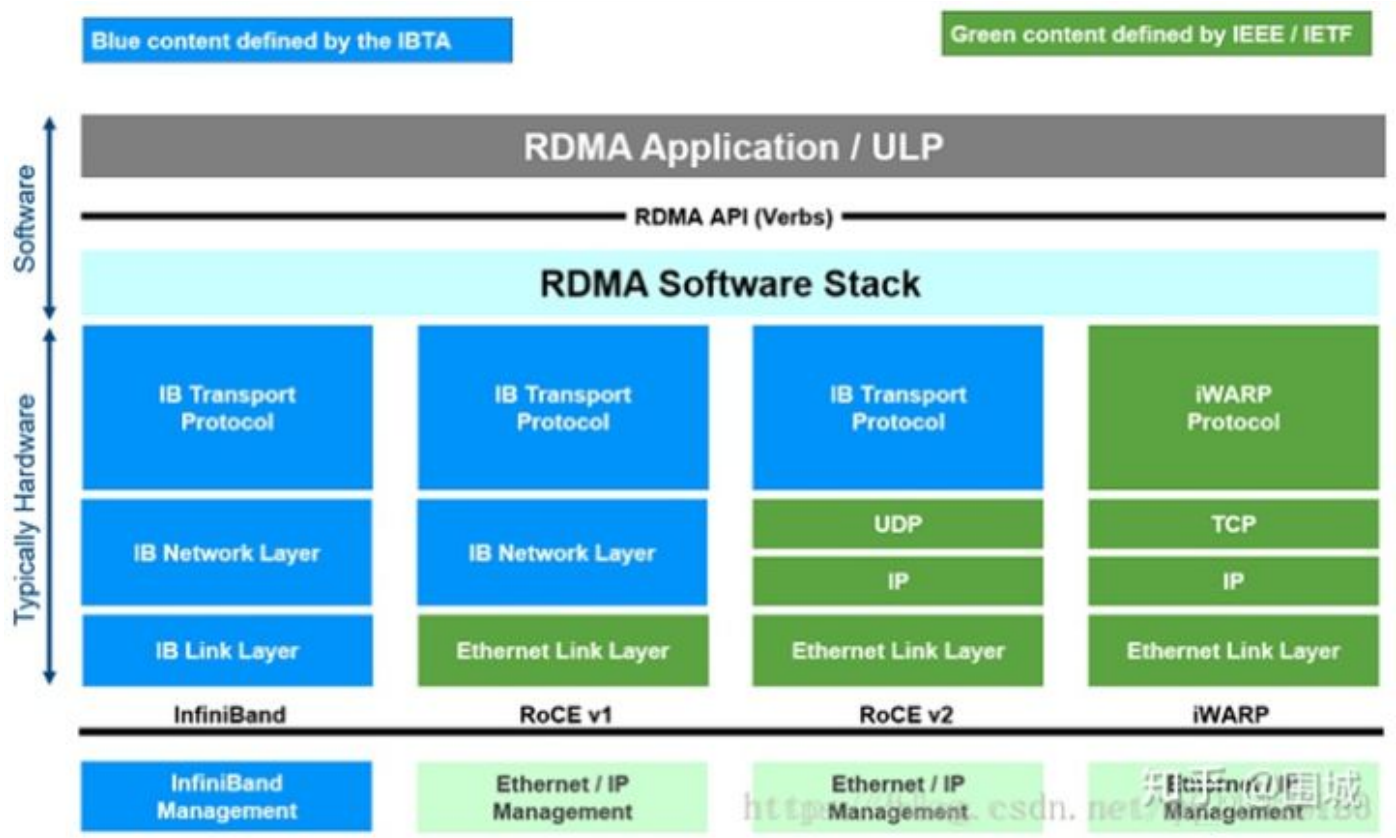
目前的RDMA硬件提供一种数据报传输：不可靠的数据报（UD），并且不支持memory verbs。

	SEND/RECV	WRITE	READ	WQE header
RC	✓	✓	✓	36 B
UC	✓	✓	✗	36 B
UD	✓	✗	✗	68 B

Table 1: Operations supported by each transport type, and their Mellanox WQE header size for SEND, WRITE, and READ. RECV WQE header size is 16 B for all transports.

3.3 RDMA三种不同的硬件实现

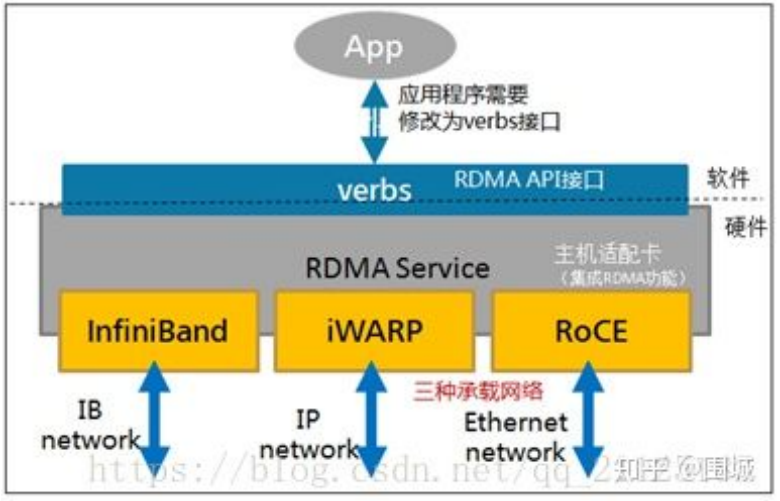
目前RDMA有三种不同的硬件实现。分别是InfiniBand、iWarp（internet Wide Area RDMA Protocol）、RoCE(RDMA over Converged Ethernet)。



目前，大致有三类RDMA网络，分别是Infiniband、RoCE、iWARP。其中，Infiniband是一种专为RDMA设计的网络，从硬件级别保证可靠传输 ， 而RoCE 和 iWARP都是基于以太网的RDMA技术，支持相应的verbs接口，如图1所示。从图中不难发现，RoCE协议存在RoCEv1和RoCEv2两个版本，主要区别RoCEv1是基于以太网链路层实现的RDMA协议(交换机需要支持PFC等流控技术，在物理层保证可靠传输)，而RoCEv2是以太网TCP/IP协议中UDP层实现。从性能上，很明显Infiniband网络最好，但网卡和交换机是价格也很高，然而RoCEv2和iWARP仅需使用特殊的网卡就可以了，价格也相对便宜很多。

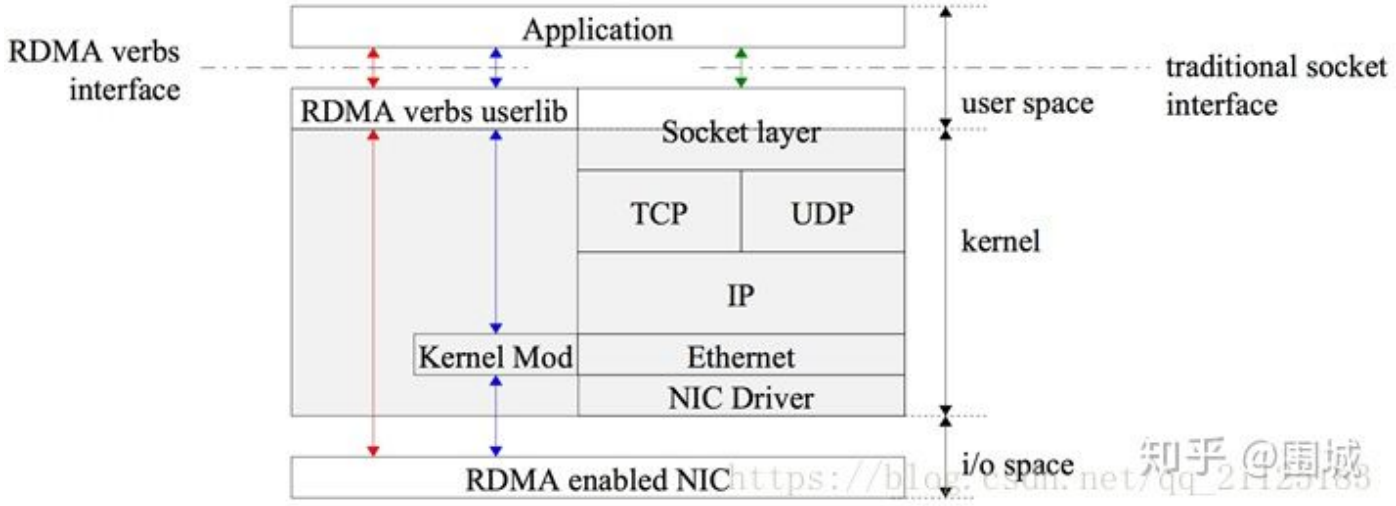
1. Infini
术的I

2. RoCE，一个允许在以太网上执行RDMA的网络协议。 其较低的网络标头是以太网标头，其较高的网络标头（包括数据）是InfiniBand标头。 这支持在标准以太网基础设施（交换机）上使用RDMA。 只有网卡应该是特殊的，支持RoCE。
3. iWARP，一个允许在TCP上执行RDMA的网络协议。 IB和RoCE中存在的功能在iWARP中不受支持。 这支持在标准以太网基础设施（交换机）上使用RDMA。 只有网卡应该是特殊的，并且支持iWARP（如果使用CPU卸载），否则所有iWARP堆栈都可以在SW中实现，并且丧失了大部分RDMA性能优势。



	InfiniBand（IB）	iWARP	RoCE
标准组织	IBTA	IETF	IBTA
性能	最好	稍差（受TCP影响）	与IB相当
成本	高	中	低
网卡厂商	Mellanox 40Gbps	Chelsio 10Gbps	Mellanox-40Gbps Emulex –10/40Gbps

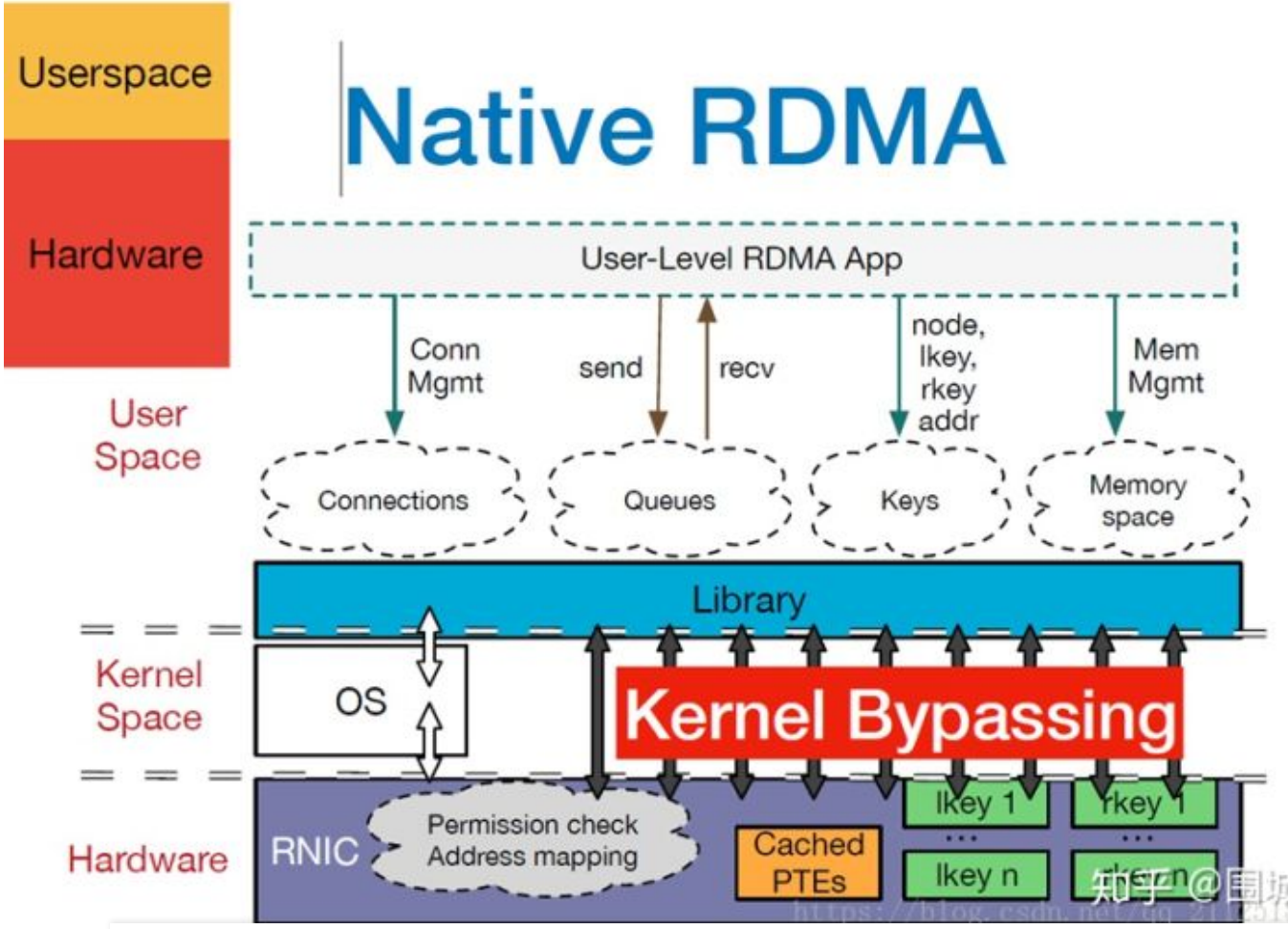
3.4 RDMA技术



传统上的
RDMA的

的verbs interface而不是传统的TCP/IP Socket interface。要使用RDMA首先要建立从RDMA到应用程序内存的数据路径 ，可以通过RDMA专有的verbs interface接口来建立这些数据路径，一旦数据路径建立后，就可以直接访问用户空间buffer。

3.5 RDMA整体系统架构图



上诉介绍的是RDMA整体框架架构图。从图中可以看出，RDMA在应用程序用户空间，提供了一系列verbs interface接口操作RDMA硬件。RDMA绕过内核直接从用户空间访问RDMA 网卡(RNIC)。RNIC网卡中包括Cached Page Table Entry，页表就是用来将虚拟页面映射到相应的物理页面。

3.6 RDMA技术详解

RDMA 的工作过程如下:

1)当一个应用执行RDMA 读或写请求时，不执行任何数据复制.在不需要任何内核内存参与的条件下，RDMA 请求从运行在用户空间中的应用中发送到本地NIC(网卡)。

2) NIC 读取缓冲的内容，并通过网络传送到远程NIC。

3) 在网络上传输的RDMA 信息包含目标虚拟地址、内存钥匙和数据本身.请求既可以完全在用户空间中处理(通过轮询用户级完成排列)，又或者在应用一直睡眠到请求完成时的情况下通过系统中断处理.RDMA 操作使应用可以从一个远程应用的内存中读数据或向这个内存写数据。

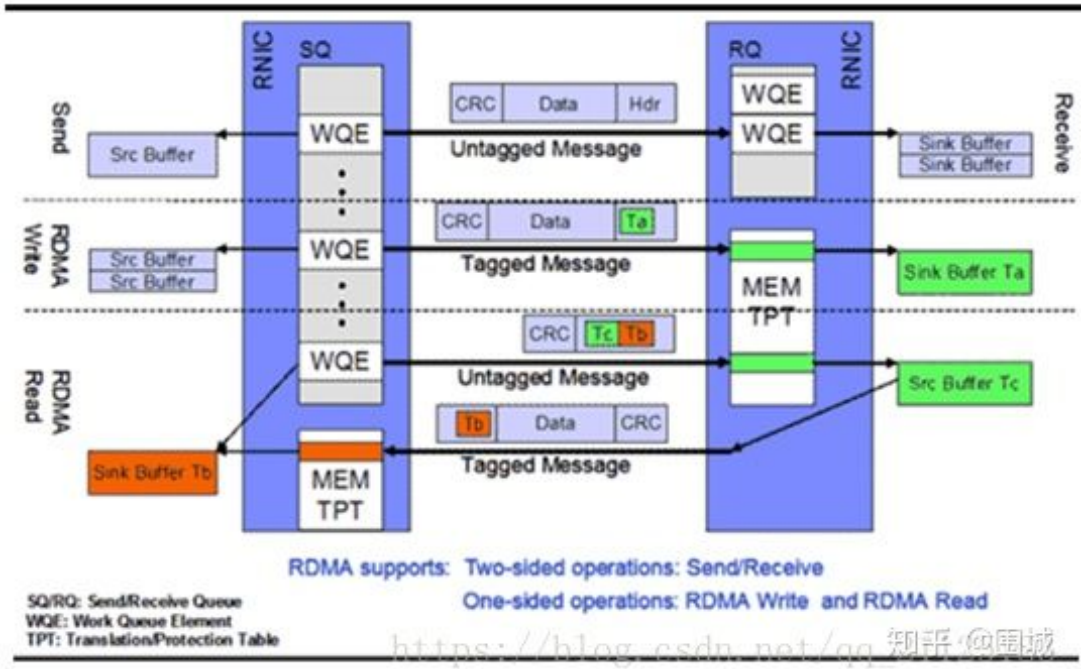
4) 目标NIC 确认内存钥匙，直接将数据写人应用缓存中.用于操作的远程虚拟内存地址包含在RDMA 信息中。

3.7 RDMA操作细节

RDMA提供了基于消息队列的点对点通信，每个应用都可以直接获取自己的消息，无需操作系统和协议栈的介入。
消息服务建立在通信双方本端和远端应用之间创建的Channel-IO连接之上。当应用需要通信时，就会创建
Send Q

被映射到应用的虚拟地址空间，使得应用直接通过它访问RNIC网卡。除了QP描述的两种基本队列之外，RDMA还提供一种队列Complete Queue（CQ），CQ用来知会用户WQ上的消息已经被处理完。

RDMA提供了一套软件传输接口，方便用户创建传输请求Work Request(WR），WR中描述了应用希望传输到Channel对端的消息内容，WR通知QP中的某个队列Work Queue(WQ)。在WQ中，用户的WR被转化为Work Queue Element（WQE）的格式，等待RNIC的异步调度解析，并从WQE指向的Buffer中拿到真正的消息发送到Channel对端。



3.7.1 RDAM单边操作 (RDMA READ)

READ和WRITE是单边操作，只需要本端明确信息的源和目的地址，远端应用不必感知此次通信，数据的读或写都通过RDMA在RNIC与应用Buffer之间完成，再由远端RNIC封装成消息返回到本端。

对于单边操作，以存储网络环境下的存储为例，数据的流程如下：
1. 首先A、B建立连接，QP已经创建并且初始化。
2. 数据被存档在B的buffer地址VB，注意VB应该提前注册到B的RNIC (并且它是一个Memory Region) ，并拿到返回的local key，相当于RDMA操作这块buffer的权限。
3. B把数据地址VB，key封装到专用的报文传送到A，这相当于B把数据buffer的操作权交给了A。同时B在它的WQ中注册进一个WR，以用于接收数据传输的A返回的状态。
4. A在收到B的送过来的数据VB和R_key后，RNIC会把它们连同自身存储地址VA到封装RDMA READ请求，将这个消息请求发送给B，这个过程A、B两端不需要任何软件参与，就可以将B的数据存储到A的VA虚拟地址。
5. A在存储完成后，会向B返回整个数据传输的状态信息。

单边操作传输方式是RDMA与传统网络传输的最大不同，只需提供直接访问远程的虚拟地址，无须远程应用的参与其中，这种方式适用于批量数据传输。

3.7.2 RDMA 单边操作 (RDMA WRITE)

对于单边操作，以存储网络环境下的存储为例，数据的流程如下：
1. 首先A、B建立连接，QP已经创建并且初始化。
2. 数据remote目标存储buffer地址VB，注意VB应该提前注册到B的RNIC(并且它是一个Memory Region)，并拿到返回的local key，相当于RDMA操作这块buffer的权限。
3. B把数据地址VB，key封装到专用的报文传送到A，这相当于B把数据buffer的操作权交给了A。同时B在它的WQ中注册进一个WR，以用于接收数据传输的A返回的状态。
4. A在收到B的送过来的数据VB和R_key后，RNIC会把它们连同自身发送地址VA到封装RDMA WRITE请

5. A在发送数据完成后，会向B返回整个数据传输的状态信息。
单边操作传输方式是RDMA与传统网络传输的最大不同，只需提供直接访问远程的虚拟地址，无须远程应用的参与其中，这种方式适用于批量数据传输。

3.7.3 RDMA 双边操作 (RDMA SEND/RECEIVE)

RDMA中SEND/RECEIVE是双边操作，即必须要远端的应用感知参与才能完成收发。在实际中，SEND/RECEIVE多用于连接控制类报文，而数据报文多是通过READ/WRITE来完成的。
对于双边操作为例，主机A向主机B(下面简称A、B)发送数据的流程如下：
1. 首先，A和B都要创建并初始化好各自的QP，CQ
2. A和B分别向自己的WQ中注册WQE，对于A，WQ=SQ，WQE描述指向一个等到被发送的数据；对于B，WQ=RQ，WQE描述指向一块用于存储数据的Buffer。
3. A的RNIC异步调度轮到A的WQE，解析到这是一个SEND消息，从Buffer中直接向B发出数据。数据流到达B的RNIC后，B的WQE被消耗，并把数据直接存储到WQE指向的存储位置。
4. AB通信完成后，A的CQ中会产生一个完成消息CQE表示发送完成。与此同时，B的CQ中也会产生一个完成消息表示接收完成。每个WQ中WQE的处理完成都会产生一个CQE。
双边操作与传统网络的底层Buffer Pool类似，收发双方的参与过程并无差别，区别在零拷贝、Kernel Bypass，实际上对于RDMA，这是一种复杂的消息传输模式，多用于传输短的控制消息。

编辑于 2018-06-04

书籍推荐

网络与SDN

进入专栏

RDMA

RDMA

RDMA(RemoteDirect Memory Access)技术全称远程直接内存访问，就是为了解决...

进入专栏

赞同 212

推荐阅读

分享

有哪些值得一看能够提升自我的好书推荐？

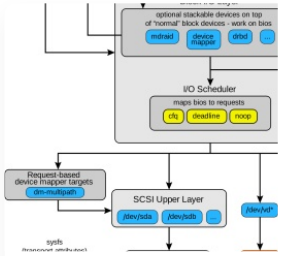
墨叶轻舞　发表于跨界知青

2018年推荐书单（140本）

kimmking

做出好决策，我推荐这5本书

靳小凡　发表于小凡聊书



Linux 的IO栈

丁凯

19 条评论

切换为时间排序

写下你的评论...

赞同 212　19 条评论　分享　收藏

赞

围城 (作者) 回复 FredZhou 1 年前

RDMA消除了数据包在用户空间和内核空间中的数据移动和复制的开销，并且数据的发送完全由RDMA特定的网卡进行数据发送，RDMA这种零拷贝的技术以及不用CPU干涉极大的降低了传输延迟。凭借可靠的传输RDMA支持ACK重传机制重发数据包，来保证数据的传输是可靠的。像InfiniBand这样的现代RDMA实现使用了一个无损链路层，它可以防止使用链路层流量控制的基于拥塞的损失

3

Starr Wang 回复 围城 (作者) 1 年前

感觉类似pcie的mac层

赞

展开其他 1 条回复

偷得浮生半日闲 11 个月前

可否提供一些深入学习 RDMA 的相关资料？谢谢

赞

偷得浮生半日闲 11 个月前

或者说明一下文章的插图可以在哪里找到原始资料。谢谢楼主

赞

围城 (作者) 回复 偷得浮生半日闲 11 个月前

你邮箱私信给我吧 我发你资料

赞

赞同 212

分享

邮箱已发私信，谢谢楼主

赞

知乎用户 11 个月前

您好，麻烦您能提供一些学习RDMA的资料给我吗？刚给您私信了，谢谢！

赞

朱君鹏 10 个月前

写的挺清楚的

赞

知乎用户 10 个月前

很详尽的资料

赞

「已注销」 7 个月前

需要专门的网卡？那好遗憾。

赞

前端交友 7 个月前

没看懂 谁规定一定要queue pair的？

赞

Rix Tox 7 个月前

支持 TLS 嗎？

赞

RDMA和U-net的区别在哪？都已经避开内核复制了

赞

苏文强 回复 风澈 2 个月前

区别在于U-net的数据，还是要从内存->cpu->NIC，数据经过cpu，这样速度会变慢，增加cpu的负载压力，cpu拷贝大数据时，存在cache命中率，context切换的缺点，而DMA不存在这个问题。

赞

风澈 4 个月前

楼主，RDMA单边操作不是很清楚，A和B是谁读谁应该写一下吧，不知道那边是远端

赞

我就呵呵了 24 天前

楼主，你的rdma分析非常棒，我最近在分析mlx5驱动，需要为另一个驱动添加roce的功能，奈何一直缺少资料，对于roce的理解很浅显，能否提供一些学习rdma或者roce的资料，原本查到了ib specification的资料，但太多了，一时半会读不完。以后也希望能够跟楼主多交流一下。

赞

王哲 3 天前

可以提供完整的Reference嘛？我看到reference只有标号

赞