“架构师技术联盟”微信公众号专注技术架构和行业解决方案，构建专业交流平台，分享一线技术实践，洞察行业前沿趋势，内容覆盖云计算、大数据、超融合、软件定义网络、数据保护和解决方案等。

《InfiniBand 架构和技术实战总结(第二版)》

公众号作者牺牲业余休息时间，把历史原创文章进行了总结、归类和细化，梳理成电子书(含免费)，方便相关从业者参阅。很多读者读完反馈受益良多，作者也倍感欣慰。核心电子书收少许整理费，一则可把书传递给真正需要的读者手中，二则算是对作者的略微肯定，重点书目如下所示(持续更新…)：

《数据备份和副本管理技术全面解析》
《容器技术架构、网络和生态详解》
《闪存技术、产品和发展趋势全面解析》
《虚拟化技术最详细解析》
《传统企业存储知识完全解析》
《IO 知识和系统性能深度调优全解》
《业界主流数据中心存储双活完全解析》
《Ceph 技术架构、生态和特性详细对比分析》
《数据中心大二层交换技术详解》
《VMware 云数据中心(私有云)解决方案详解》
《大数据时代数据重删技术详解》
《高性能计算 HPC 技术、方案和行业全面解析》
《RDMA 原理分析和技术对比》

说明：免费电子书下载地址(实时更新防止链接失效)->请关注“架构师技术联盟”微信号或在“架构师电子书店”首页，按照提示语或说明获取下载地址。

“架构师技术联盟”微信公众号

架构师技术联盟书店

目录

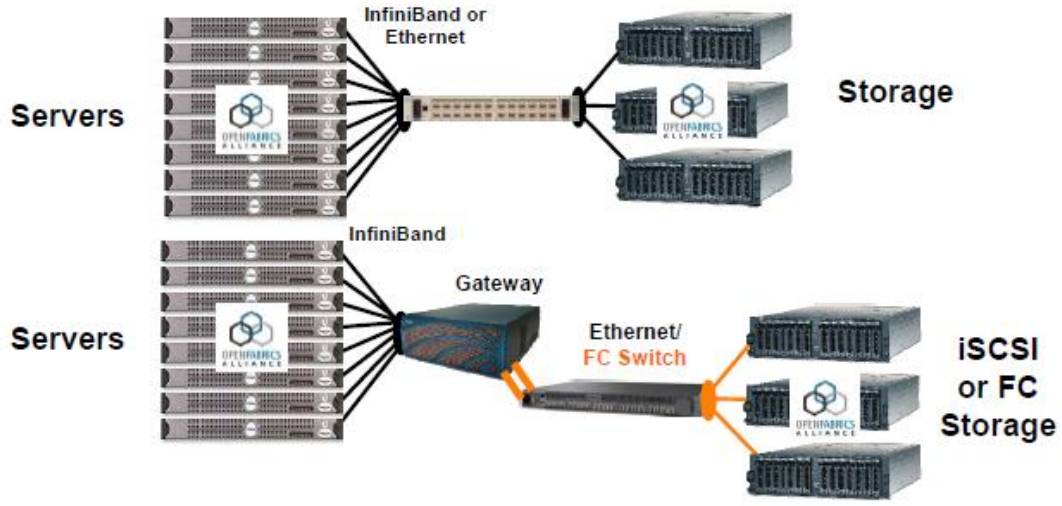
第一章 InfiniBand 关键技术和概念解析

传统的 TCP/IP 协议的多层次结构使得复杂的缓冲管理带来很大的网络延迟和操作系统的额外开销，随着网络技术的发展，网络需要一种开放、高带宽、低延迟、高可靠以及满足集群无限扩展能力的以交换为核心的体系架构，在这种技术背景下，InfiniBand（简称 IB）应运而生。

根据 IB 高带宽、低延时、高可靠以及满足集群无限扩展能力的特点，IB 主要定位于存储网络和计算网络的应用。IB 使用 RDMA(Remote Direct Memory Access 远程直接内存存取)技术，通过一个虚拟的寻址方案，让服务器知道和使用其他服务器的部分内存，无需操作系统的内核干预，既直接继承了总线的高带宽和低时延，又降低了 CPU 的处理负担，这对于像存储这样的集群来说很合适。

1.1 什么是 InfiniBand (IB)?

InfiniBand 是一种网络通信协议，它提供了一种基于交换的架构，由处理器节点之间、处理器节点和输入/输出节点(如磁盘或存储)之间的点对点双向串行链路构成。每个链路都有一个连接到链路两端的设备，这样在每个链路两端控制传输(发送和接收)的特性就被很好地定义和控制了。



InfiniBand 通过交换机在节点之间直接创建一个私有的、受保护的通道，进行数据和消息的传输，无需 CPU 参与远程直接内存访问(RDMA)和发送/接收由 InfiniBand 适配器管理和执行的负载。适配器通过 PCI Express 接口一端连接到 CPU，另一端通过 InfiniBand 网络端口连接到 InfiniBand 子网。与其他网络通信协议相比，这提供了明显的优势，包括更高的带宽、更低的延迟和增强的可伸缩性。

InfiniBand Trade Association (IBTA)成立于 1999 年，负责 InfiniBand 商业

产品的符合性和互操作性测试。IBTA 比任何其他互连解决方案更积极地推动了高性能的开发，确保了为 21 世纪设计的体系结构。作为 IBTA 指导委员会的积极成员，Mellanox 感到自豪。

但是 IBTA 的 9 个主要董事成员 CRAY、Emulex、HP、IBM、intel、Mellanox、Microsoft、Oracle、Qlogic 中只有 Mellanox 和 Emulex 专门在做 InfiniBand，其他成员只是扮演了使用 InfiniBand 的角色。而 Emulex 由于业务不景气也在 2015 年的 2 月被 Avago 收购，Qlogic 的 infiniband 业务在 2012 年也全部卖给 Intel 了。

有关 InfiniBand 的更多信息，请参见:
https://cw.infinibandta.org/document/dl/7268

1.2 InfiniBand 与传统的网络协议有何不同?

InfiniBand 是为实现最高效的数据中心而设计的。它本机支持服务器虚拟化、覆盖网络和软件定义网络(SDN)。

InfiniBand 采用以应用程序为中心的方式进行消息传递，找到在一点到另一点之间传输数据的阻力最小的路径。这与传统的网络协议(如 TCP/IP 和光纤通道)不同，后者使用更以网络为中心的方法进行通信。

直接访问意味着应用程序不依赖于操作系统来传递消息。在传统的互连中，操作系统是共享网络资源的唯一所有者，这意味着应用程序不能直接访问网络。相反，应用程序必须依赖操作系统将数据从应用程序的虚拟缓冲区传输到网络堆栈并传输到网络上，而接收端的操作系统必须具有类似的功能，只是反向操作。
相比之下，InfiniBand 则通过绕过网络栈，在两端为应用程序之间的通信创建直接通道，从而避免了操作系统的介入。InfiniBand 的简单目标是为应用程序提供消息服务，让其直接与另一个应用程序或存储通信。一旦建立起来，InfiniBand 体系结构的其余部分将确保这些通道能够将大小不一的消息传输到虚拟地址空间，跨越物理距离，具有隔离性和安全性。

1.3 InfiniBand 与 TCP 有什么不同?

InfiniBand 架构，InfiniBand Architecture(IBA)是为硬件实现而设计的，而 TCP 则是为软件实现而设计的。因此，InfiniBand 是比 TCP 更轻的传输服务，因为它不需要重新排序数据包，因为较低的链路层提供有序的数据包交付。传输层只需要检查包序列并按顺序发送包。进一步,因为 InfiniBand 提供以信用为基础的流控制(发送方节点不给接收方发送超出广播 “信用“大小的数据包),传输层不需要像 TCP 窗口算法那样的包机制确定最优飞行包的数量。这使得高效的产品能够以非常低的延迟和可忽略的 CPU 利用率向应用程序交付 56、100Gb/s 的数据速率。

1.4 InfiniBand 严格意义上是 I/O Fabric 吗?

不，InfiniBand 提供更多能力。在最低层，InfiniBand 提供了高性能、低延迟、可靠的交换机架构，作为可伸缩的 I/O 互连。然而，InfiniBand 提供了更高层次的功能，支持应用程序集群、虚拟化和 san(存储区域网络)。

1.5 InfiniBand 是分层协议吗?

是的。InfiniBand 规范基于 OSI 7 层模型松散地定义了模块化层中的协议，覆盖了 1-4 层。规范定义了给定层与上面和下面的层之间的接口。因此，最低的物理层只对上面的链接层进行接口。InfiniBand 链路层定义了下面物理层的接口和上面网络层的另一个接口。

1.6 InfiniBand 的优势是什么?

InfiniBand 相对于其他互联技术的主要优势包括:
更高的吞吐量: 每台服务器和存储连接 56Gb/s,与 40Gb 以太网和光纤通道相比，很快达到 100Gb/s

更低的延迟: RDMA 零拷贝网络减少了操作系统的开销，所以数据可以在网络中快速移动

•增强的可伸缩性: InfiniBand 只需要添加额外的交换机，就可以在理论上适应基于相同交换机组件的无限规模的平面网络
•更高的 CPU 效率: 随着数据移动的减少，CPU 可以在其应用程序上花费更多的计算周期，这将减少运行时间并增加每天的作业数量

1.7 可用的 InfiniBand 数据速率是多少?

InfiniBand 串行链路可以在不同的信令速率下运行，然后可以捆绑在一起实现更高的吞吐量。原始信令速率与编码方案耦合，产生有效的传输速率。编码将通过铜线或光纤发送的数据的错误率降至最低，但也增加了一些开销(例如，每 8 位数据传输 10 位)。

典型的实现是聚合四个链接单元(4X)。目前，InfiniBand 系统提供以下吞吐量速率:

Table 1: InfiniBand Data Rates

Name	Abbreviation	Raw Signaling Rate	Applied Encoding	Effective Data Rate	Aggregated (4x) Throughput
Single Data Rate	SDR	2.5 Gb/s	8b/10b	2 Gb/s	**8 Gb/s**
Double Data Rate	DDR	5 Gb/s	8b/10b	4 Gb/s	**16 Gb/s**
Quad Data Rate	QDR	10 Gb/s	8b/10b	8 Gb/s	**32 Gb/s**
Fourteen Data Rate	FDR	14.1 Gb/s	64b/66b	13.64 Gb/s	**54.5 Gb/s**
Enhanced Data Rate	EDR	25.8 Gb/s	64b/66b	25 Gb/s	**100 Gb/s**
High Data Rate	HDR	51.6 Gb/s	64b/66b	50 Gb/s	**200 Gb/s**
Next Data Rate	NDR	TBD	TBD	TBD	**TBD**

有关 InfiniBand 数据速率的更多信息，请参见:
http://www.infinibandta.org/content/pages.php?pg=technology_overview

1.8 什么是 RDMA ?它的优点是什么?

InfiniBand 使用远程直接内存访问(Remote Direct Memory Access, RDMA)，将数据从通道的一端传输到另一端。RDMA 能够在网络上直接在应用程序之间传输数据，而不涉及操作系统，同时消耗可忽略的 CPU 资源(零拷贝传输)。另一端的应用程序只是直接从内存读取消息，消息已经成功传输。
这减少了 CPU 开销，增加了网络快速移动数据的能力，并允许应用程序更快地接收数据。给定数量的数据从源传输到目标的时间间隔称为延迟，延迟越低，应用程序作业完成的速度越快。

Figure 1: Traditional Interconnect

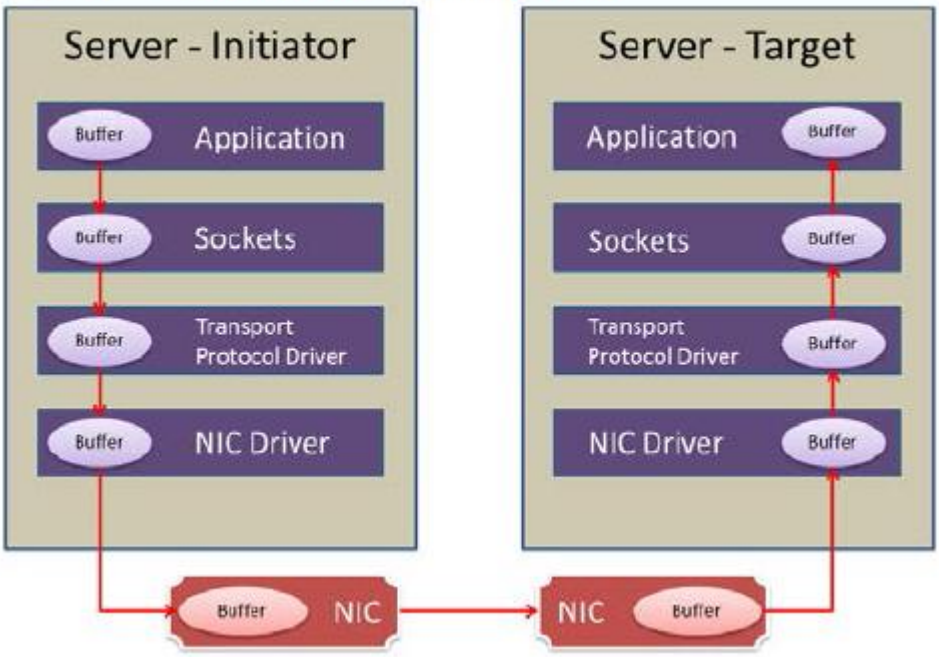
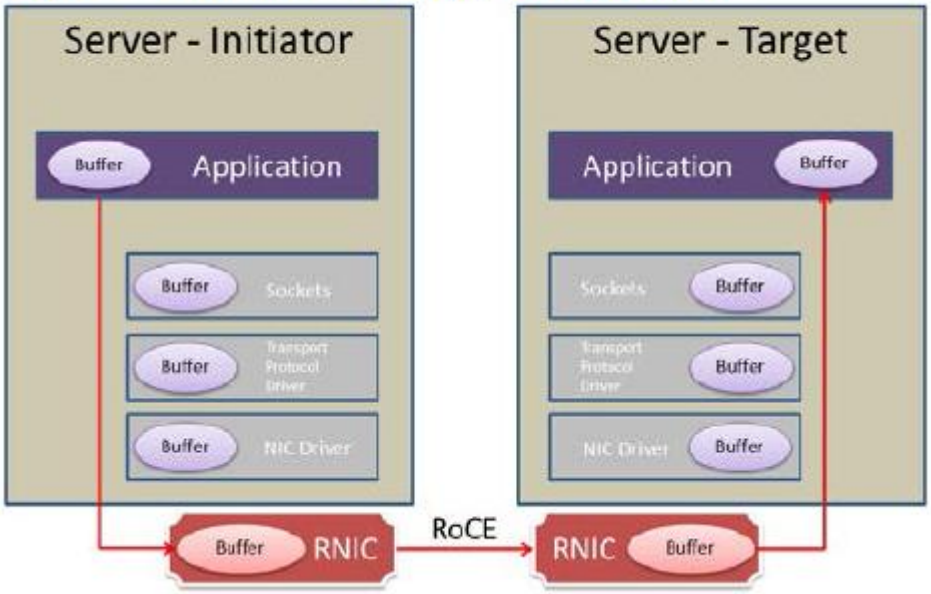


Figure 2: RDMA Zero-Copy Interconnect



Mellanox FDR InfiniBand 的延时最低可达 0.7 微秒，是数据传输的最低延时。有关 RDMA 的更多信息，请参见:http://www.mellanox.com/blog/tag/rdma/

1.9 InfiniBand 架构的主要元素是什么?

InfiniBand 网络的基本组成部分是:
•主机通道适配器(HCA)
•交换机
•子网经理(SM)
•网关

- Host Channel Adapter (HCA)
  - Device that terminates an IB link and executes transport-level functions and support the verbs interface

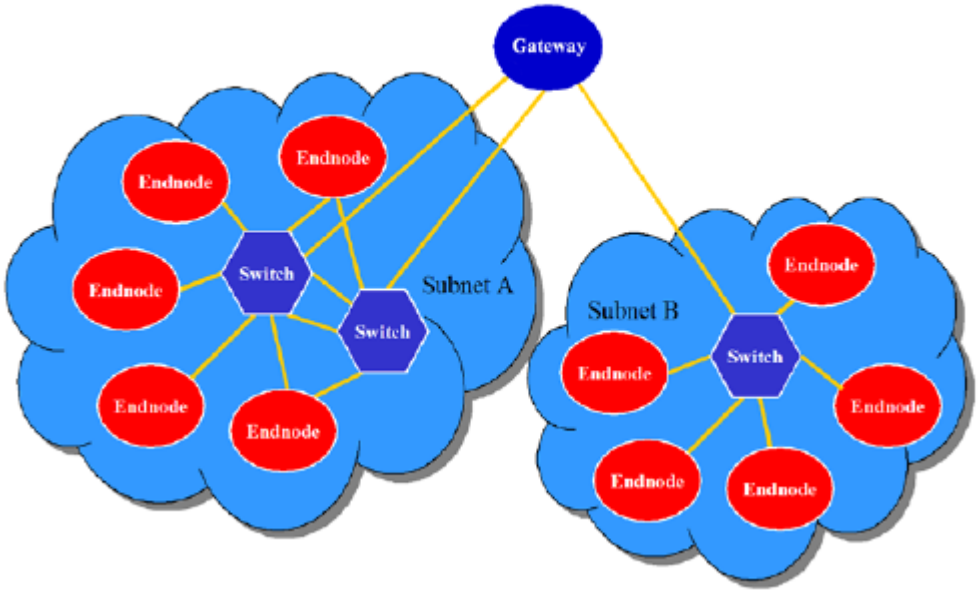
- Switch
  - A device that moves packets from one link to another of the same **IB** Subnet

- Router
  - A device that transports packets between different IBA subnets

- Bridge/Gateway
  - **InfiniBand** to **Ethernet**

每个终端节点必须有一个主机通道适配器来设置和维护与主机设备的链接。交换机包含多个 InfiniBand 端口，将数据包从一个端口转发到另一个端口，以在子网中继续传输数据包。路由器用于将包从一个子网转发到另一个子网(如果必要的话)。子网管理是通过软件定义的网络来处理的，它控制网络的物理元素，并通过开放的、行业标准的接口引入流量工程特性。

Figure 3: Basic InfiniBand Architecture



有关 InfiniBand 架构的更多详细信息，请参见:http://www.mellanox.com/related-docs/solutions/deploying-hpc-cluster-with-mellanox-infiniband-interconnect-solutions.pdf

1.10 什么是主机通道适配器(HCA)?

主机通道适配器(HCA)是一种接口卡或控制器，它连接着 InfiniBand 导线和主机系统总线。每个终端节点都必须有一个 HCA，它设置和维护主机设备和网络上其他实体之间的链接。
HCAs 提供到其他设备的端口连接。HCA 可以连接到另一个 HCA、目标设备或交换机。

1.11 什么是交换机?在 InfiniBand 中如何工作?

交换机用于物理连接网络中的设备，并将传入的数据流量转发到目的地。交换机有多个端口，可以通过电缆将数据处理和转发到特定设备，从而控制网络内的流量。
在 InfiniBand 网络中，交换机是 InfiniBand 结构的一个关键部分。实际上，InfiniBand 被称为“交换结构”或“基于交换的互连”，因为当流量被转发时，就有一个从一个端口到另一个端口的逻辑连接，类似于老式的电话交换机。随着更多的交换机被添加到系统中，设备之间可能有更多的路径。

1.12 什么是子网管理器(SM)?

子网管理器(SM)是一个软件实体，配置其本地子网并确保其运行。它在每个端点

之间设置主路径和次要路径，以便预先编程流转发决策并以最少的时间传递数据。子网中必须至少有一个 SM，以便管理所有的交换机和路由器设置，并在链路断开或出现新链路时重新配置子网。SM 可以驻留在子网中的任何设备上。
子网中的所有设备必须包含专用子网管理代理(SMA)， SM 使用它与各种 InfiniBand 组件通信。

一个子网中可以有多个 SMs，只要在任何时刻只有一个是活动的。非活动 SMs(称为备用子网管理器)保留活动 SM 的转发信息的副本，并验证活动 SM 是可操作的。如果活动 SM 发生故障，备用 SM 将接管其职责，以确保整个结构继续运行。
软件定义网络(SDN)已经成为子网管理的主要手段，它不仅控制网络设备和它们之间的数据流量，而且增加了网络的灵活性和可伸缩性。

1.13 在 InfiniBand 网络中路由器是必需的吗?

路由器用于在不同的计算机网络之间设备连接。当数据到达路由器时，它读取数据包中的地址信息并确定最终目的地。
在以太网网络中，由于泛洪机制和生成树，大型子网的效率不高。因此，需要一个路由器来连接各种较小的子网。
这些问题在 InfiniBand 中不存在，在 InfiniBand 中，多达 40,000 个节点的大型子网可以高效运行。没有必要划分更小的子网，所以路由器不是必需的。
然而，随着对数据中心的需求不断增长，如果需要超过 40000 个节点的子网，就可以使用 InfiniBand 路由器技术，如 Mellanox 的交换机 IBTM。

1.14 什么是网关?它如何在 InfiniBand 网络中工作?

网关是子网中的设备，充当两个协议之间的桥梁。例如，它允许 InfiniBand 集群访问以太网网络或存储接口。它允许公司实现一个 InfiniBand 网络，通过有一个专用接口，可以连接到运行他协议上的现有遗留设备。
有关 InfiniBand 网关的更多信息，请参见：http://www.mellanox.com/page/gateway_overview

1.15 VPI 与 InfiniBand 有什么关系 ？

Virtual Protocol Interconnect (VPI)是一个 Mellanox 技术,为客户提供互连同时确保可伸缩性。VPI 允许 HCA 或交换机中的任何端口运行 InfiniBand 或以太网，并根据需要在两个协议之间进行切换。对于服务器和存储系统，这是最高的连接灵活性。对于交换网络，这创建了一个开箱即用的完美网关，支持 InfiniBand 和以太网网络以及集群之间的集成。

Mellanox VPI 提供端口灵活性，可以根据需要调整端口。此外，Mellanox VPI 提供了 InfiniBand 的所有优点，同时维护了与现有以太网的连接。最重要的是，这是可以实现性能无损失，因为 Mellanox 在其 InfiniBand 和以太网产品中确保

了最高的带宽和最低的延迟。
有关 Mellanox VPI 的更多信息，请参见:http://www.mellanox.com/related-docs/case_studies/CS_VPI_GW.pdf

1.16 什么是 LID，GID 和 GUID?

子网中的所有设备都有一个本地标识符(LID)，这是子网管理器分配的 16 位地址。子网内发送的所有包都使用 LID 作为在链路级别转发和交换包的目标地址。LIDs 支持在单个子网内最多 48,000 个结束节点。当一个子网被重新配置时，新的盖子被分配到子网中的各个端点。
不同子网之间的路由是基于全局标识符(GID)完成的，这是一个 128 位地址，模仿 IPv6 地址，这允许 InfiniBand 本质上无限的可伸缩性。GID 标识终端节点、端口、交换机或多播组。
全局惟一标识符(GUID)是子网中所有元素的 64 位定义，包括 Chassis、HCAs、交换机、路由器和端口。GUID 不会改变，它是创建 GID 的地址的一部分。
GID 和 GUIDs 独立于 LIDs，因此对子网重构不感知。

1.17 InfiniBand 支持 IP 流量吗?IPoIB 是什么?

通过网络接口将 IP 包封装在 InfiniBand 包中，Internet 协议(IP)包可以通过 InfiniBand 接口发送。这就是 IP over IB (IPoIB)。只要 InfiniBand 网络安装了必要的驱动程序，它就会使用分区键(PKEYs)为每个端口创建一个接口，然后可以无缝地在 InfiniBand 网络上传输 IP 数据包。
有关 IPoIB 的更多信息，请参阅 Mellanox OFED 用户手册或 Mellanox WinOF VPI 用户手册。
http://www.mellanox.com/related-docs/prod_software/Mellanox_OFED_Linux_User_Manual_v2.2-1.0.1.pdf
http://www.mellanox.com/related-docs/prod_software/MLNX_VPI_WinOF_User_Manual_v4.70.pdf

1.18 什么是可靠和不可靠的传输方式?

当信息包从一个节点传输到另一个节点时，它们可以可靠或不可靠地发送。可靠的传输就像它听起来的那样——没有丢失或重复的包，包必须按顺序发送，并且接收节点向发送方提供确认(正面或负面)包已经被正确接收。通过确保通过硬件协议将数据传递给远程对等点，应用程序本身就免除了这种责任。

不可靠的传输不会提供这样的确认，并且将尽最大努力按顺序交付数据包(在使用不可靠传输时称为数据报)。

可靠的传输需要保留额外的资源，以确保数据包被正确地发送和接收，以及处理

此类确认。因此，在某些情况下，数据中心可能会选择使用不可靠的数据报而不是可靠的连接，以便将这些资源分配到其他地方。
IP 通过 InfiniBand (IPoIB)可以运行在不可靠传输的不可靠数据报(UD)模式，或连接模式(CM)，这允许可靠传输。在运行时使用一个命令就可以在两种模式之间切换。默认的 InfiniBand 传输模式是 UD，但是可以添加一个脚本，将新的接口自动配置为 CM，以便可靠传输。

1.19 IPoIB 支持绑定吗?

绑定指的是将两个端口连接在一起，供单个应用程序使用的过程，这为发送数据提供了更大的灵活性。IPoIB 接口的绑定只在 Linux 中可用，可以用与以太网接口的绑定相同的方式完成，即通过 Linux 绑定驱动程序。Windows 还不支持绑定。

1.20 InfiniBand 支持多播吗?

开关可以配置为转发单播包(到一个位置)或多播包(到多个设备)。这些完全热的可切换连接由子网管理器管理，可以包括发送到子网上的所有系统或这些系统的子集。InfiniBand 的高带宽为这种多播能力提供了主干，无需二级互连链路。

1.21 InfiniBand 支持服务质量吗?

服务质量(QoS)是指网络能够为应用程序提供不同的优先级，并在数据流到这些端点时保证一定程度的性能。
InfiniBand 通过创建虚拟专用道(VL)来支持 QoS。这些 VLs 是单独的逻辑通信链路，共享一个物理链路。每个链路最多可支持 15 个标准 VLs(通过 VL14 指定 VL0)和一个管理通道(指定 VL15)。VL15 优先级最高，VL0 最低。
用服务级别(SL)定义数据包，以确保其 QoS 级别。SL 为每个链路提供了理想的通信优先级。通信路径上的每个交换机或路由器都有一个映射表，它由子网管理器设置以将 SL 转换为 VL，以便在该链接上支持的 VLs 中保持适当的优先级。
在主机端，可以为每个流(队列对)分配 QoS 参数，基本上每个端点提供 1600 万个 QoS 级别。
当应用程序的带宽需求达到定义的 QoS 限制时，应用程序在主机上使用的资源将不再增长。降低已达到其定义限制的应用程序的 QoS 限制将释放主机上的资源，从而使该主机上的其他工作负载受益。这对于流量共享具有很高的价值。
有关服务质量的更多信息，请参阅 Mellanox OFED 用户手册或 Mellanox WinOF VPI 用户手册。

1.22 InfiniBand 是无损网络吗?

无损 Fabric 是一种不定期丢包的网络。以太网被认为是一种有损的网络结构，因为它经常丢弃数据包。TCP 传输层检测丢失的数据包并进行调整。

相比之下，InfiniBand 则使用链路级流量控制，以确保数据包不会在 Fabric 中丢失。这种无损流量控制使得数据中心内的带宽得到了非常有效的利用，使得 InfiniBand 非常适合于数据中心之间的远程通信。

1.23 InfiniBand 如何处理安全问题?

InfiniBand 的适配器支持高级虚拟化特性,比如网络功能虚拟化(NFV)。通过 sdn 优化的 InfiniBand 交换机，我们可以构建最高效、可伸缩、高性能的网络。InfiniBand 解决方案允许数据比任何其他解决方案更快地移动和分析，从而为数据中心管理人员提供了收集网络流量信息、分析流量行为(如检测异常)和保护流量的能力。这些解决方案支持为防火墙和入侵检测解决方案构建基础设施。此外，InfiniBand 解决方案的网络流量控制是灵活的、可实时重新编程的。
有关使用 SR-IOV 适配器的基于 sdn 的安全解决方案的示例，请参阅下面的博客文章：http://www.mellanox.com/blog/2013/05/connectx-3-leverages-network-services-in-sdn-era/。

1.24 基于信用的流量控制如何工作?

流控制用于管理链路之间的数据流，以保证无损耗结构。一个链路(虚拟通道)的每个接收端向发送设备提供信用,以指定在不丢失数据的情况下可以接收多少数据。专用链路包管理设备之间的信用传递，以更新接收设备可以接收的数据包数量。除非接收端宣布的信用指示有足够的缓冲空间以接收整个消息，否则不会传输数据。

1.25 Infiniband 有生成树吗?

生成树被认为是以太网中最好的和最差的元素之一。它允许以太网在节点之间存在循环时进行诊断，并禁用冗余或并行链接，以防止重复发送数据包。
虽然这在 10-15 年前是非常有用的，但是生成树现在被认为是一种负担，因为它限制了网络工程师构建高性能网络的能力,因为它限制了节点之间可以有多少并行路径。
另一方面，Infiniband 则通过一个能看到网络中所有路径的中央代理来管理流量，从而实现了所有端点之间具有全带宽的大规模链路结构。今天，软件定义网络(SDN)作为中央子网管理，分布和分配流量，以利用端点之间的所有并行链接。更好的是，这种管理不会对性能产生影响，因为 SDN 在连接设置时确定配置中的可能路径。

1.26 InfiniBand 中 Verbs 是什么?

Verbs 是应用程序从 InfiniBand 消息传输服务请求操作的一种方法。Verbs 集合

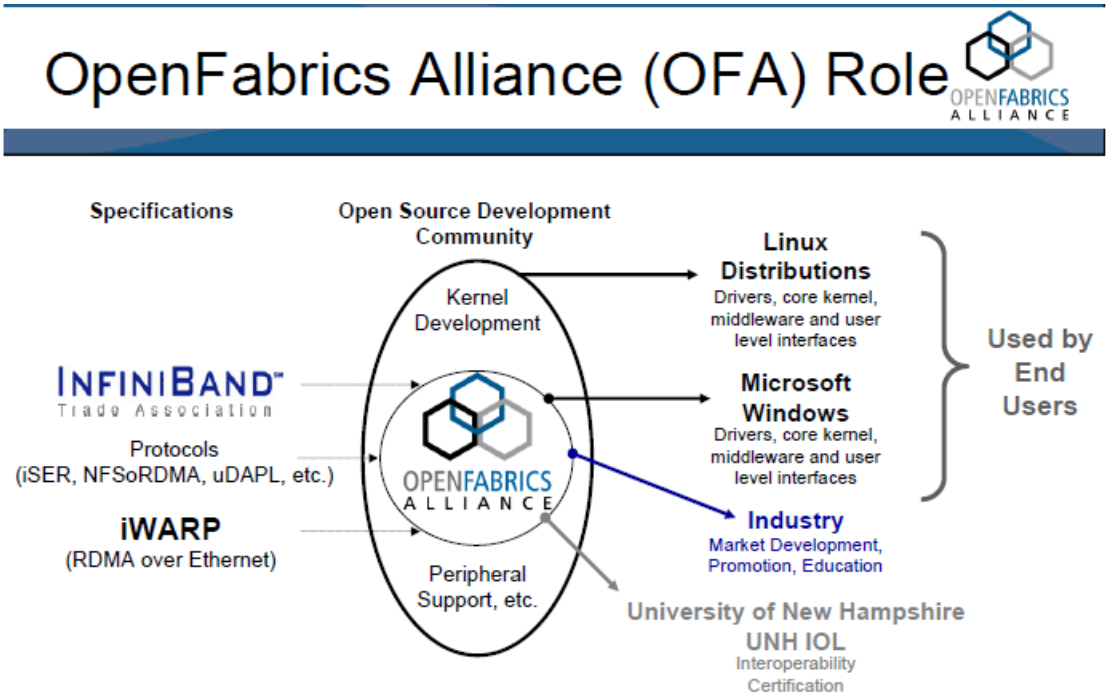
是应用程序在无限带宽网络中交互的各种动作的语义描述。Verbs 完全定义在 InfiniBand 软件传输接口规范中。
这些 Verbs 是指定应用程序使用的 api 的基础，但是 InfiniBand 体系结构没有定义 api。其他 Fabric，如 openfabric 联盟，提供了一套完整的 api 和软件，它们实现了与 InfiniBand 硬件无缝协作的 Verbs。

有关 InfiniBand Verbs 用法的更多信息，请参见：
https://cw.infinibandta.org/document/dl/7268
http://www.mellanox.com/related-docs/prod_software/RDMA_Aware_Programming_user_manual.pdf

1.27 如何监控 InfiniBand 网络的带宽、拥塞和健康状况?

OpenFabrics Enterprise Distribution(OFED)是一组开源软件驱动、核心内核代码、中间件和支持 InfiniBand Fabric 的用户级接口程序，2005 年由 OpenFabrics Alliance(OFA)发布第一个版本。Mellanox OFED 用于 Linux，Windows (WinOF)，包括各种诊断和性能工具，用于监视 InfiniBand 网络的运行情况，包括监视传输带宽和监视 Fabric 内部的拥塞情况。

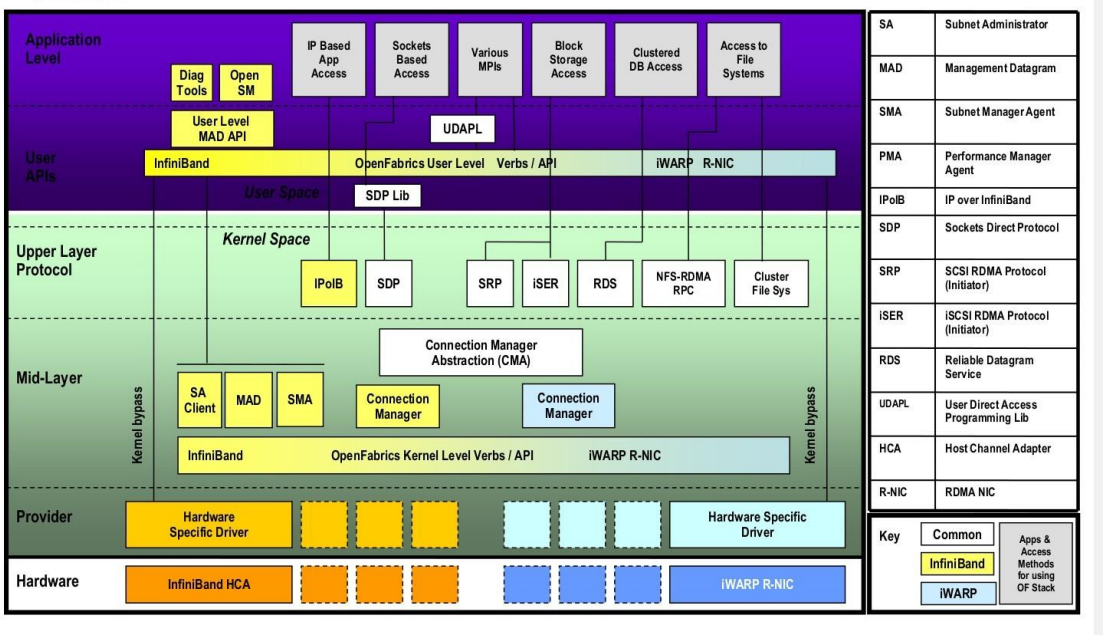
OpenFabrics Alliance(OFA)是一个基于开源的组织，它开发、测试、支持 OpenFabrics 企业发行版。该联盟的任务是开发并推广软件，通过将高效消息、低延迟和最大带宽技术架构直接应用到最小 CPU 开销的应用程序中，从而实现最大应用效率。



该联盟成立于 2004 年 6 月，最初是 OpenIB 联盟，致力于开发独立于供应商、基于 Linux 的 InfiniBand 软件栈。2005，联盟致力于支持 Windows，此举将使软件栈真正跨平台。2006 年，该组织再次扩展其章程，包括对 iWARP 的支持，在 2010 年增加了对 RoCE(RDMA over Converged)支持通过以太网交付高性能 RDMA 和内核旁路解决方案。2014 年，随着 OpenFabrics Interfaces 工作组的建立，联盟再次扩大，实现对其他高性能网络的支持。

OFED 包括内核态驱动程序，面向通道的 RDMA 和发送/接收操作，操作系统的内核旁路，用于并行消息传递（MPI）的内核态/用户态应用程序编程接口（API）和服务，套接字数据交换（如 RDS，SDP），NAS 和 SAN 存储（例如 iSER，NFS-RDMA，SRP）和文件系统/数据库系统。

OFED 支持的网络架构包括：10Gb Ethernet，iWARP，RoCE(RDMA over Converged Ethernet，InfiniBand.



今天，OpenFabrics 联盟的愿景是为 RDMA 和内核旁路提供一个统一的、跨平台的、独立于传输的软件栈。传输独立性意味着用户可以利用相同的 RDMA 和内核旁路 API 在无限带宽、iWARP、RoCE 或其他结构上不知不觉地运行他们的应用程序。为此，OFA 管理工具和开发资源，以编码、改进和发布基于标准的开源软件，这些软件具有支持关键任务应用程序所需的可伸缩性和性能。联盟成员和其他贡献者参与到工作组中，以确保 RDMA/Advanced Networks 软件和 OFED 功能、易于使用和部署、以及与网络硬件和主要操作系统的互操作性。
更多 OpenFabrics 信息：https://www.openfabrics.org/

Mellanox 还提供 Unified Fabric Manager(UFM)的软件，这是一个强大的平台来管理 InfiniBand 计算环境。UFM 使数据中心操作员能够有效地提供、监视和操作他们的 fabric，同时提高应用程序性能并确保 fabric 始终处于启动和运行状

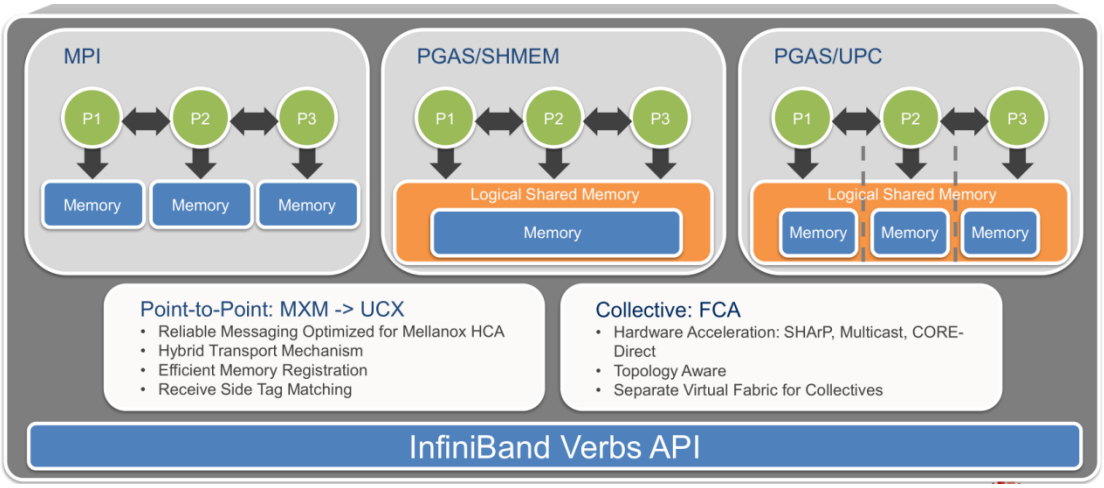
态。

有关监视 InfiniBand Fabric 的更多信息，请参见:Mellanox OFED For Linux:
http://www.mellanox.com/page/products_dyn?
Mellanox
WinOF:http://www.mellanox.com/page/products_dyn?product_family=32
UFM:http://www.mellanox.com/page/products_dyn?product_family=100&mtag=unified_fabric_manager

1.28 HPC-X 并行计算软件包

HPC-X 是 Mellanox 公司推出的一款并行计算软件包，支持 MPI、PGAS（OpenSHMEM,UPC）等编程模型。HPC-X 针对 Mellanox 公司的产品提供了一系列的并行库和调测工具，极大地方便了并行程序的开发和优化。

Component	Description
OpenMPI	An Open-source impletion of Message Passing Model
OpenSHMEM/BUPC	Impletion of Partitioned Address Space Model
UCX	Unified Communication X, a framework that complete point-to-point communication. UCX hide the bottom network layer(IB, RoCE, iWARP, etc) and provide two/one-sided operations for MPI/PGAS.
MXM	Mellanox Message Accelerator. Not available on ARM.
FCA/HCOLL	Fabric Collective Accelerator/Hierarchical Collectives. Collective operation accelerator.
SHARP	Scalable Hierarchical Aggregation and Reduction Protocol. Offload collective operations to from CPU to switch network. Used by HCOLL.
KNEM	A Linux kernel module that improves intra-node communication.
Libibprof/IPM	Profiling Tools.
OSU/IMB	Standard Benchmarks.



Notes:
1） HPC-X 的 MPI 不支持多线程，根据实际应用需求可自行重编。
2） MXM 库只对 x86 有效，对 ARM 不提供支持

1.29 InfiniBand 和 Mellanox 更多学习资源

InfiniBand 贸易协会:http://www.infinibandta.org

Mellanox 网站: http://www.mellanox.com

Mellanox 社区: https://community.mellanox.com/welcome

Mellanox 学院: http://www.mellanox.com/academy/

Mellanox 会议圆桌: http://www.mellanox.com/page/webinars

Mellanox 博客: http://www.mellanox.com/blog/

InfiniBand 白皮书: http://www.mellanox.com/page/white_papers

第二章 InfiniBand 发展和技术原理

与其他网络协议（如 TCP/IP）相比，IB 协议具有更高的传输效率。原因在于许多网络协议如 TCP/IP 具有转发损失的数据包的能力，但是由于要不断地确认与重发，基于这些协议的通信也会因此变慢，极大地影响了性能。与之相比，IB 使用基于信任的、流控制的机制来确保连接的完整性，数据包极少丢失。使用 IB 协议，除非确认接收缓存具备足够的空间，否则不会传送数据。接受方在数据传输完毕之后，返回信号来标示缓存空间的可用性。通过这种办法，IB 协议消除了由于原数据包丢失而带来的重发延迟，从而提升了效率和整体性能。
IB 是以通道（Channel）为基础的双向、串行式传输，在连接拓朴中是采用交换、

切换式结构(Switched Fabric),所以会有所谓的 IBA(Infiniband Architecture)交换器(Switch),此外在线路不够长时可用 IBA 中继器(Repeater)进行延伸。而每一个 IBA 网络称为子网(Subnet),每个子网内最高可有 65,536 个节点(Node),IBA Switch、IBA Repeater 仅适用于 Subnet 范畴,若要通跨多个 IBA Subnet 就需要用到 IBA 路由器(Router)或 IBA 网关器(Gateway)。至于节点部分,Node 想与 IBA Subnet 接轨必须透过配接器(Adapter),若是 CPU、内存部分要透过 HCA(Host Channel Adapter),若为硬盘、I/O 部分则要透过 TCA(Target Channel Adapter),之后各部分的衔接称为联机(Link)。上述种种构成了一个完整的 IBA。

IB 的传输方式相当灵活,若在设备机内可用印刷电路板的铜质线箔传递(特别是用在工控、电信设备的 Backplane 背板上),若在机外可用铜质缆线传递,或需要更远的传递也可改用光纤,若用铜箔、铜缆最远可至 17m,而光纤则可至 10km,同时 IBA 也支持热插拔,及具有自动侦测、自我调适的 Active Cable 活化智能性连接机制。

Infiniband 开放标准技术简化并加速了服务器之间的连接,同时支持服务器与远程存储和网络设备的连接。

2.1 InfiniBand 技术的发展

1999 年开始起草规格及标准规范,2000 年正式发表,但发展速度不及 Rapid I/O、PCI-X、PCI-E 和 FC,加上 Ethernet 从 1Gbps 进展至 10Gbps。所以直到 2005 年之后,InfiniBand Architecture(IBA)才在集群式超级计算机上广泛应用。全球 Top 500 大效能的超级计算机中有相当多套系统都使用上 IBA。

随着越来越多的大厂商正在加入或者重返到它的阵营中来,包括 Cisco、IBM、

HP、Sun、NEC、Intel、LSI 等。InfiniBand 已经成为目前主流的高性能计算机互连技术之一。为了满足 HPC、企业数据中心和云计算环境中的高 I/O 吞吐需求，新一代高速率 56Gbps 的 FDR (Fourteen Data Rate) 和 EDR InfiniBand 技术已经出现。

2.2 InfiniBand 技术的优势

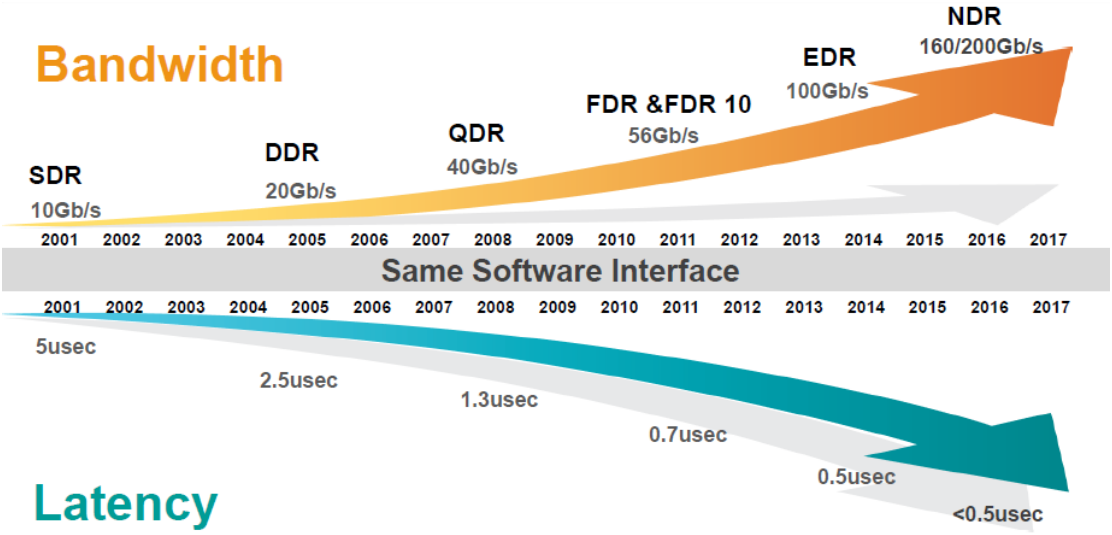
Infiniband 大量用于 FC/IP SAN、NAS 和服务器之间的连接,作为 iSCSI RDMA 的存储协议 iSER 已被 IETF 标准化。目前 EMC 全系产品已经切换到 Infiniband 组网，IBM/TMS 的 FlashSystem 系列，IBM 的存储系统 XIV Gen3，DDN 的 SFA 系列都采用 Infiniband 网络。

相比 FC 的优势主要体现在性能是 FC 的 3.5 倍，Infiniband 交换机的延迟是 FC 交换机的 1/10，支持 SAN 和 NAS。

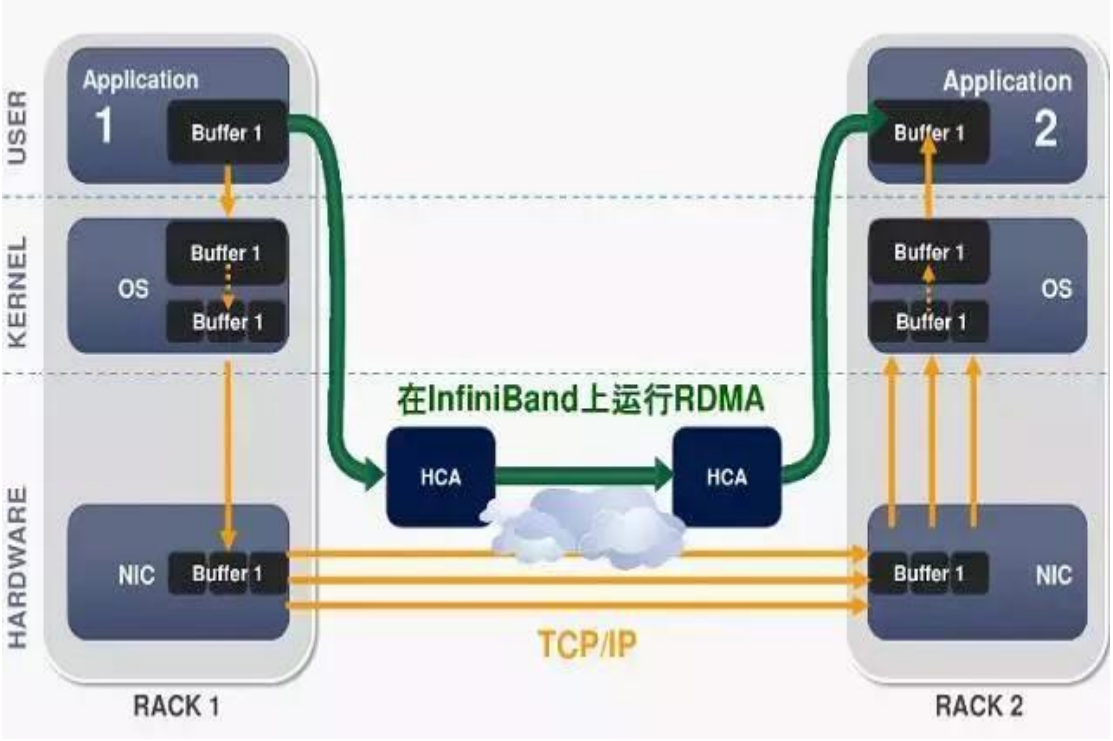
存储系统已不能满足于传统的 FC SAN 所提供的服务器与裸存储的网络连接架构。HP SFS 和 IBM GPFS 是在 Infiniband fabric 连接起来的服务器和 iSER Infiniband 存储构建的并行文件系统，完全突破系统的性能瓶颈。

Infiniband 采用 PCI 串行高速带宽链接，从 SDR、DDR、QDR、FDR 到 EDR HCA 连接，可以做到 1 微妙、甚至纳米级别极低的时延，基于链路层的流控机制实现先进的拥塞控制。

InfiniBand 采用虚通道(VL 即 Virtual Lanes)方式来实现 QoS，虚通道是一些共享一条物理链接的相互分立的逻辑通信链路，每条物理链接可支持多达 15 条的标准虚通道和一条管理通道(VL15)。



RDMA 技术实现内核旁路，可以提供远程节点间 RDMA 读写访问，完全卸载 CPU 工作负载，基于硬件传出协议实现可靠传输和更高性能。



相比 TCP/IP 网络协议，IB 使用基于信任的、流控制的机制来确保连接的完整性，数据包极少丢失，接受方在数据传输完毕之后，返回信号来标示缓存空间的可用性，所以 IB 协议消除了由于原数据包丢失而带来的重发延迟，从而提升了效率和整体性能。

TCP/IP 具有转发损失的数据包的能力，但是由于要不断地确认与重发，基于这些协议的通信也会因此变慢，极大地影响了性能。

2.3 InfiniBand 组网方式和相关概念

IB 是以通道为基础的双向、串行式传输，在连接拓朴中是采用交换、切换式结构(Switched Fabric)，在线路不够长时可用 IBA 中继器(Repeater)进行延伸。每一个 IBA 网络称为子网(Subnet)，每个子网内最高可有 65,536 个节点(Node)，IBA Switch、IBARepeater 仅适用于 Subnet 范畴，若要通跨多个 IBASubnet 就需要用到 IBA 路由器(Router)或 IBA 网关器(Gateway)。

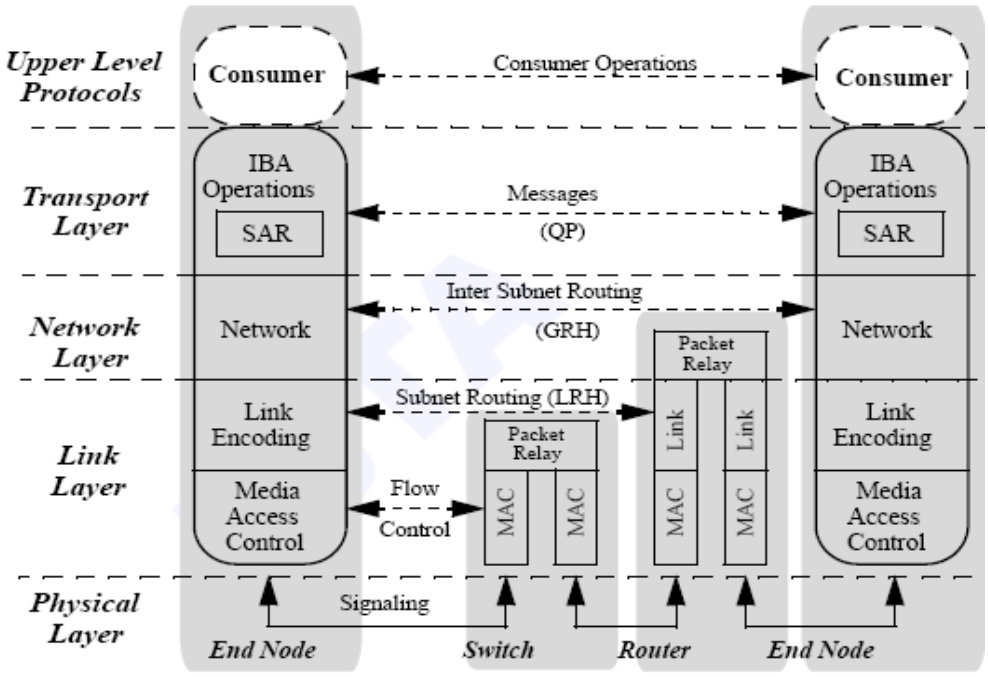
每个节点(Node) 必须透过配接器(Adapter)与 IBA Subnet 连接，节点 CPU、内存要透过 HCA(Host Channel Adapter)连接到子网；节点硬盘、I/O 则要透过 TCA(TargetChannel Adapter)连接到子网，这样的一个拓扑结构就构成了一个完整的 IBA。

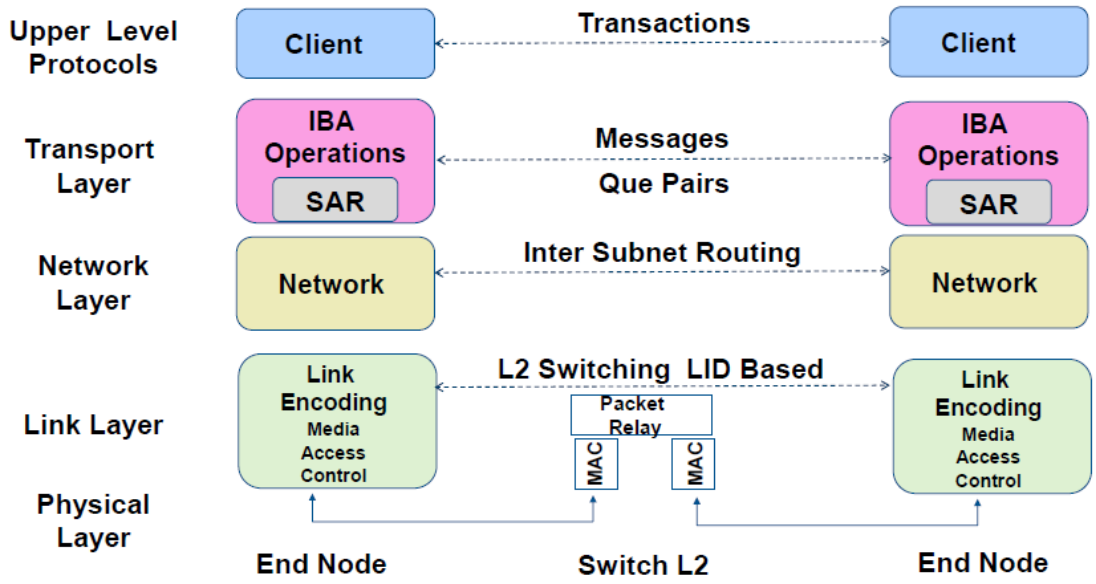
IB 的传输方式和介质相当灵活，在设备机内可用印刷电路板的铜质线箔传递

(Backplane 背板)，在机外可用铜质缆线或支持更远光纤介质。若用铜箔、铜缆最远可至 17m，而光纤则可至 10km，同时 IBA 也支持热插拔，及具有自动侦测、自我调适的 Active Cable 活化智能性连接机制。

2.4 InfiniBand 协议简介

InfiniBand 也是一种分层协议(类似 TCP/IP 协议)，每层负责不同的功能，下层为上层服务，不同层次相互独立。IB 采用 IPv6 的报头格式。其数据包报头包括本地路由标识符 LRH，全局路由标示符 GRH，基本传输标识符 BTH 等。





2.4.1 物理层协议

物理层定义了电气特性和机械特性，包括光纤和铜媒介的电缆和插座、底板连接器、热交换特性等。定义了背板、电缆、光缆三种物理端口。

并定义了用于形成帧的符号(包的开始和结束)、数据符号(DataSymbols)、和数据包直接的填充(Idles)。详细说明了构建有效包的信令协议，如码元编码、成帧标志排列、开始和结束定界符间的无效或非数据符号、非奇偶性错误、同步方法等。

负责数据帧在线缆上的比特传输。IB提供的是一种无损网络传输，物理层保证满足$10e^{-12}$的误码率要求。

物理层定义了电气特性和机械特性，包括光纤和铜媒介的电缆和插座、底板连接器、热交换特性等；
定义了三种物理端口：背板端口、电缆端口、光缆端口。电缆端口为铜线，传输距离可达 100m，光纤端口最远可以到 10 公里；

定义了比特位是如何在线路上形成符号的，并定义了用于形成帧的符号(包的开始和结束)、数据符号(Data Symbols)、和数据包直接的填充(Idles)。它详细说明了构建有效包的信令协议，如码元编码、成帧标志排列、开始和结束定界符间的无效或非数据符号、非奇偶性错误、同步方法等。

2.4.2 链路层协议

链路层描述了数据包的格式和数据包操作的协议，如流量控制和子网内数据包的路由。链路层有链路管理数据包和数据包两种类型的数据包。

负责解析报文格式，和报文操作，例如流控和子网内的报文交换；
链路层描述了数据包的格式和数据包操作的协议，如流量控制和子网内数据包的路由。链路层有两种类型的数据包：

链路管理数据包—用于运行和维持链路操作的数据包。在链路层产生和消耗且不与流量控制冲突。用于链路每个末端端口之间协商比特率、链路宽度等操作参数，同时也用于传送流量控制信用和维持链路完整。

2.4.3 网络层协议

网络层是子网间转发数据包的协议，类似于 IP 网络中的网络层。实现子网间的数据路由，数据在子网内传输时不需网络层的参与。

数据包中包含全局路由头 GRH，用于子网间数据包路由转发。全局路由头部指明了使用 IPv6 地址格式的全局标识符(GID)的源端口和目的端口，路由器基于 GRH 进行数据包转发。GRH 采用 IPv6 报头格式。GID 由每个子网唯一的子网 标示符和端口 GUID 捆绑而成。

2.4.4 传输层协议

传输层负责报文的分发、通道多路复用、基本传输服务和处理报文分段的发送、接收和重组。传输层的功能是将数据包传送到各个指定的队列(QP)中，并指示队列如何处理该数据包。当消息的数据路径负载大于路径的最大传输单元(MTU)时，传输层负责将消息分割成多个数据包。

接收端的队列负责将数据重组到指定的数据缓冲区中。除了原始数据报外，所有的数据包都包含 BTH，BTH 指定目的队列并指明操作类型、数据包序列号和分区信息。

负责分发报文到期望的目的端，并负责对超过 MTU 的报文进行分段和重组；主要负责报文的分发、通道多路复用和基本传输服务，此外还负责处理报文分段的发送、接收和重组。传输层的功能是将数据包传送到各个指定的队列队(QP)中，并指示 QP 如何处理该数据包。当消息的数据路径负载大于路径的最大传输单元(MTU)时，传输层负责将消息分割成多个数据包。接收端的 QP 负责将数据重组到指定的数据缓冲区中。

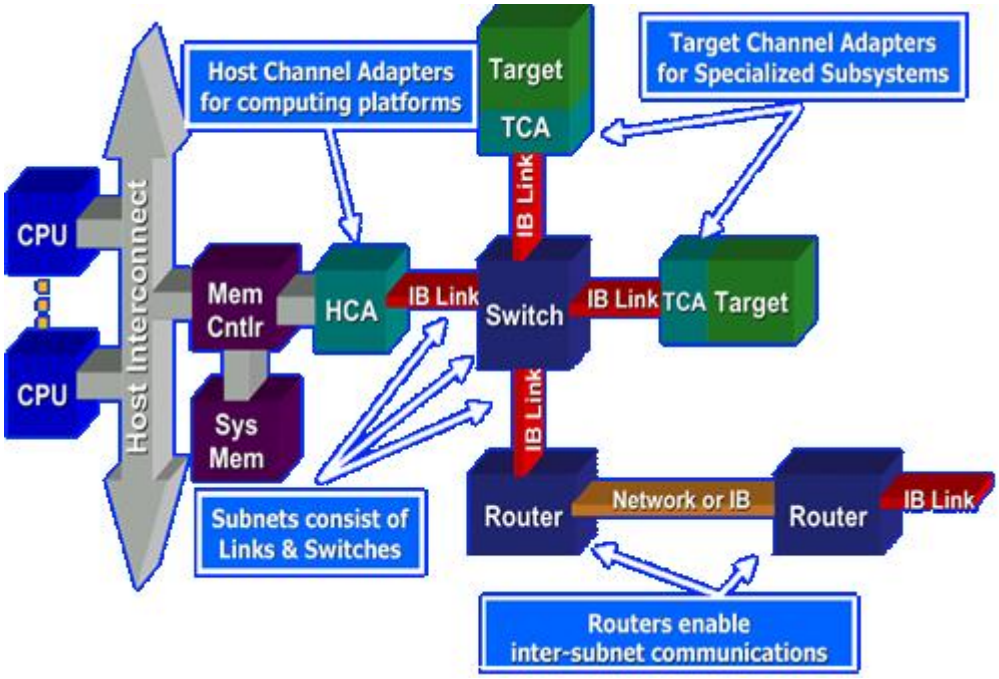
2.4.5 上层网络协议

InfiniBand 为不同类型的用户提供了不同的上层协议，提供应用程序与硬件驱动之间的 Verbs 接口，允许上层应用基于 Verbs 接口进行 RDMA 编程；并为某些管理功能定义了消息和协议。InfiniBand 主要支持 SDP、SRP、iSER、RDS、IPoIB 和 uDAPL 等上层协议。

- SDP(SocketsDirect Protocol)是 InfiniBand Trade Association (IBTA)制定的基于 infiniband 的一种协议，它允许用户已有的使用 TCP/IP 协议的程序运行在高速的 infiniband 之上。
- SRP(SCSIRDMA Protocol)是 InfiniBand 中的一种通信协议，在 InfiniBand 中将 SCSI 命令进行打包，允许 SCSI 命令通过 RDMA(远程直接内存访问)在不同的系统之间进行通信，实现存储设备共享和 RDMA 通信服务。
- iSER(iSCSIRDMA Protocol)类似于 SRP(SCSI RDMA protocol)协议，是 IB SAN 的一种协议 ，其主要作用是把 iSCSI 协议的命令和数据通过 RDMA 的方式跑到例如 Infiniband 这种网络上，作为 iSCSI RDMA 的存储协议 iSER 已被 IETF 所标准化。
- RDS(ReliableDatagram Sockets)协议与 UDP 类似，设计用于在 Infiniband 上使用套接字来发送和接收数据。实际是由 Oracle 公司研发的运行在 infiniband 之上，直接基于 IPC 的协议。
- IPoIB(IP-over-IB)是为了实现 INFINIBAND 网络与 TCP/IP 网络兼容而制定的协议，基于 TCP/IP 协议，对于用户应用程序是透明的，并且可以提供更大的带宽，也就是原先使用 TCP/IP 协议栈的应用不需要任何修改就能使用 IPoIB。
- uDAPL(UserDirect Access Programming Library)用户直接访问编程库是标准的 API，通过远程直接内存访问 RDMA 功能的互连（如 InfiniBand）来提高数据中心应用程序数据消息传送性能、伸缩性和可靠性。

2.5 InfiniBand 体系架构

图表 1 InfiniBand 体系架构图



IB 标准定义了一套用于系统通信的多种设备，包括通道适配器（Channel Adapter)、交换机（Switch）和路由器（Router)：

1、 通道适配器（Channel Adapter）用于同其他设备的连接，包括主机通道适配器（HCA）用于主控NODE，和目标通道适配器（TCA）用于外设NODE，使IO设备脱离主机而直接置于网络中，通道适配器实现物理层，链路层，网络层和传输层的功能。

通道适配器是 IB 网络接口的一个重要组成部分,是带有特定保护特性的可编程 DMA 器件，允许本地和远端的 DMA 操作。

2、 交换机（Switch）是IB结构中的基本组件，负责在IB子网里转发报文；

3、 路由器（Router）也是IB结构中的基本组件，负责在不同的IB子网间转发报文。

2.6 InfiniBand 工作原理

与其他网络协议（如 TCP/IP）相比，IB 具有更高的传输效率。原因在于许多网络协议具有转发损失的数据包的能力，但是由于要不断地确认与重发，基于这些协议的通信也会因此变慢，极大地影响了性能。

需要说明的是，TCP 协议是一种被大量使用的传输协议，从冰箱到超级计算机等各种设备上都可以看到它的身影，但是使用它必须付出高昂的代价：TCP 协议极其复杂、代码量巨大并且充满了各种特例，而且它比较难卸载。

与之相比，IB 使用基于信任的、流控制的机制来确保连接的完整性，数据包极少丢失。使用 IB，除非确认接收缓存具备足够的空间，否则不会传送数据。接受方在数据传输完毕之后， 返回信号来标示缓存空间的可用性。通过这种办法，IB 消除了由于原数据包丢失而带来的重发延迟，从而提升了效率和整体性能。

IB 是以通道（Channel）为基础的双向、串行式传输，在连接拓朴中是采用交换、切换式结构（Switched Fabric)，所以会有所谓的 IBA 交换器（Switch)，此外在线路不够长时可用 IBA 中继器（Repeater）进行延伸。而每一个 IBA 网络称为子网（Subnet)，每个子网内最高可有 65,536 个节点（Node)，IBA Switch、IBA Repeater 仅适用于 Subnet 范畴，若要通跨多个 IBA Subnet 就需要用到 IBA 路由器（Router）或 IBA 网关器（Gateway)。

至于节点部分，Node 想与 IBA Subnet 接轨必须透过配接器（Adapter），若是 CPU、内存部分要透过 HCA（Host Channel Adapter），若为硬盘、I/O 部分则要透过 TCA（Target Channel Adapter），之后各部分的衔接称为联机（Link）。上述种种构成了一个完整的 IBA。

IB 的传输方式相当活化弹性，若在设备机内可用印刷电路板的铜质线箔传递（特别是用在工控、电信设备的 Backplane 背板上），若在机外可用铜质缆线传递，或需要更远的传递也可改用光纤，若用铜箔、铜缆最远可至 17m，而光纤则可至 10km，同时 IBA 也支持热插拔，及具有自动侦测、自我调适的 Active Cable 活化智能性连接机制。

2.7 Infiniband 技术特点

1. 高带宽

Infiniband 通过 1、4、8、12 线并行来扩展通道带宽，并采用 SDR，DDR，QDR，FDR 技术，使带宽进一步提升。采用不同并行通道数和不同技术的带宽如下表所示：

通道数	SDR模式	DDR模式	QDR模式	FDR模式
通道数 1X	2.5Gbps	5Gbps	10Gbps	14Gbps
通道数 4X	10Gbps	20Gbps	40Gbps	56Gbps
通道数 8X	20Gbps	40Gbps	80Gbps	112Gbps
通道数 12X	30Gbps	60Gbps	120Gbps	168Gbps

2. 低延时

作为高性能计算机的互连最重要的指标，基于 IB 通道的交换机时延小于 100ns，应用程序时延小于 1-3us。

3. 高可扩展性

采用点到点的交换结构，连接数万个终端设备实现无拥塞的 IB 网络。

4. Qos

提供了 16 级可映射到 16 个服务层的虚拟通道，通过对不同的虚拟通道设定优先级来实现不同服务等级 SL 的服务质量管理，基于信用的流控机制及注入速率控制机制，实现了拥塞控制。

5. 支持RDMA

IB服务器和存储网络中的服务器可以通过RDMA技术与其它服务器中的内存或者存储器高速地交换数据。

6. 专用协议卸载引擎

IB 由硬件实现高效可靠的传输层点到点连接，支持在线路上的消息传递和内存映像技术，并具有旁路OS核心的能力，分担了CPU大量负荷，提高了整体性能。

7. IO子系统与主机系统分离

通道适配器（CA）提供到IO控制器的链路和传输服务，使IO设备可以脱离主机而直接置于网络中，子网管理(一个主SM+多SMA)的模式，使管理模式更加安全高效。节省了机箱空间，有了更好的扩展性，打破了主机与IO系统的距离限制：铜线-17m，光纤-10公里。

8. 支持分区

将IB子网划分为多个分区，提供了更好的性能和更高的安全性。

9. 容错功能

通过在主机系统和IO设备之间建立多个物理通道（各物理通道相互独立），来达到容错的目的。

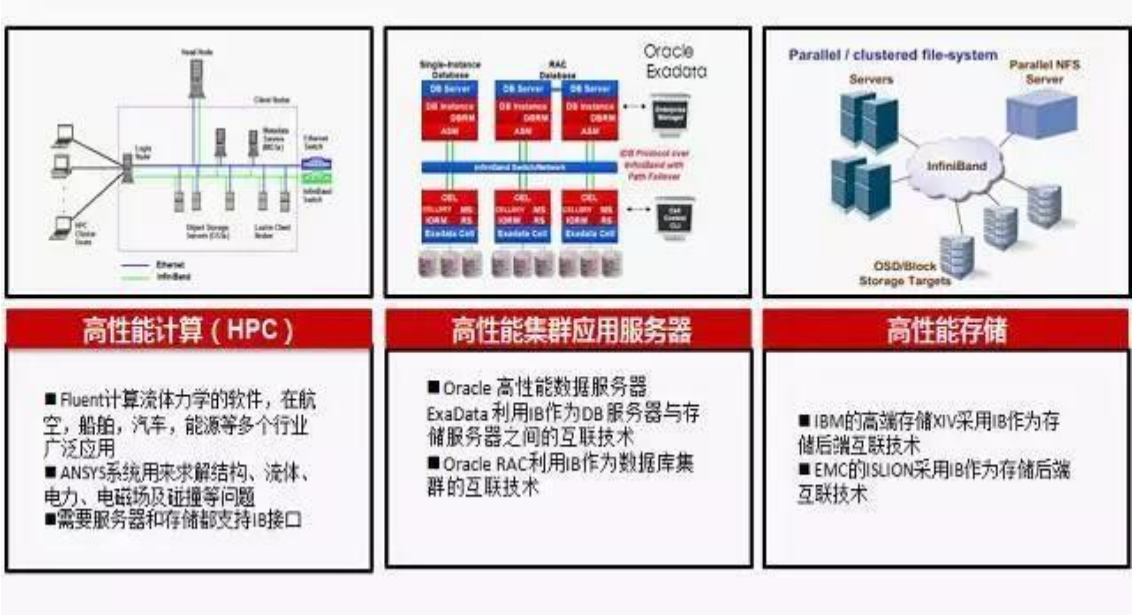
10. IPV6编址

IB采用IPv6报头格式，其传输的数据包中包含了数据的源地址和目的地址，这些地址使得IB交换机和路由器可以根据自己的转发表（由SM-SMP-SMA配置）将数据包直接转发到正确的设备。

2.8 InfiniBand应用场景

Infiniband灵活支持直连及交换机多种组网方式，主要用于HPC高性能计算

场景，大型数据中心高性能存储等场景，HPC 应用的共同诉求是低时延(<10 微秒)、低 CPU 占有率（<10%）和高带宽(主流 56 或 100Gbps)



一方面 Infiniband 在主机侧采用 RDMA 技术释放 CPU 负载，可以把主机内数据处理的时延从几十微秒降低到 1 微秒；另一方面 InfiniBand 网络的高带宽(40G、56G 和 100G)、低时延(几百纳秒)和无丢包特性吸取了 FC 网络的可靠性和以太网的灵活扩展能力。

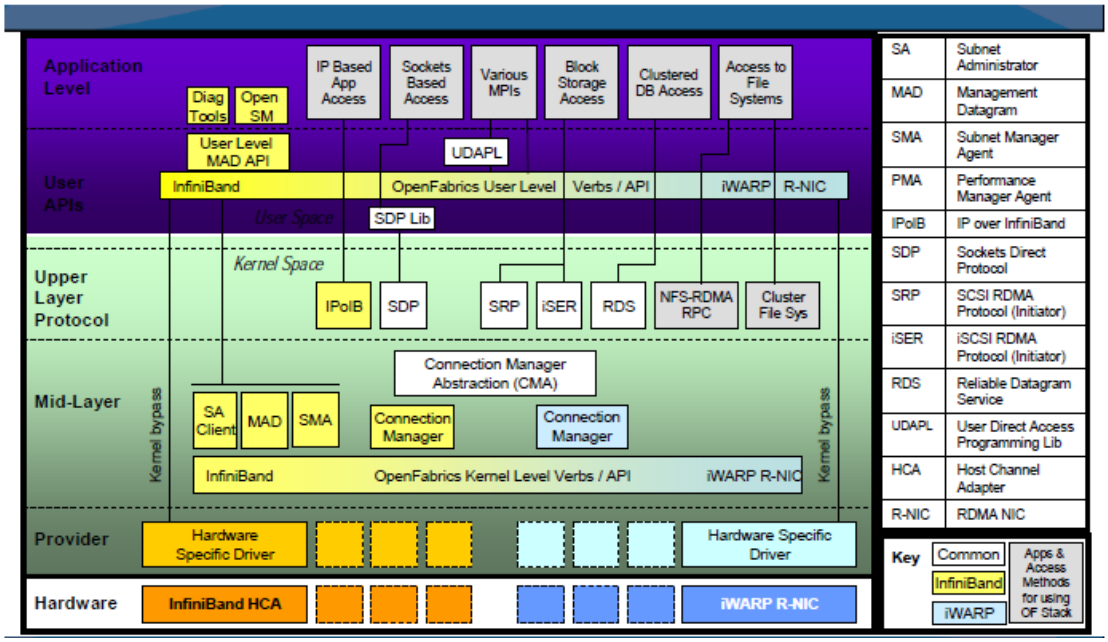
还有一家 InfiniBand 技术厂商就是 Intel，Intel 拿出 1.25 亿美元收购 QLogic 的 InfiniBand 交换机和适配器产品线发力于高性能计算领域，但今天我们重点讨论 Mellanox 的产品、技术和趋势。

第三章 InfiniBand 架构解析

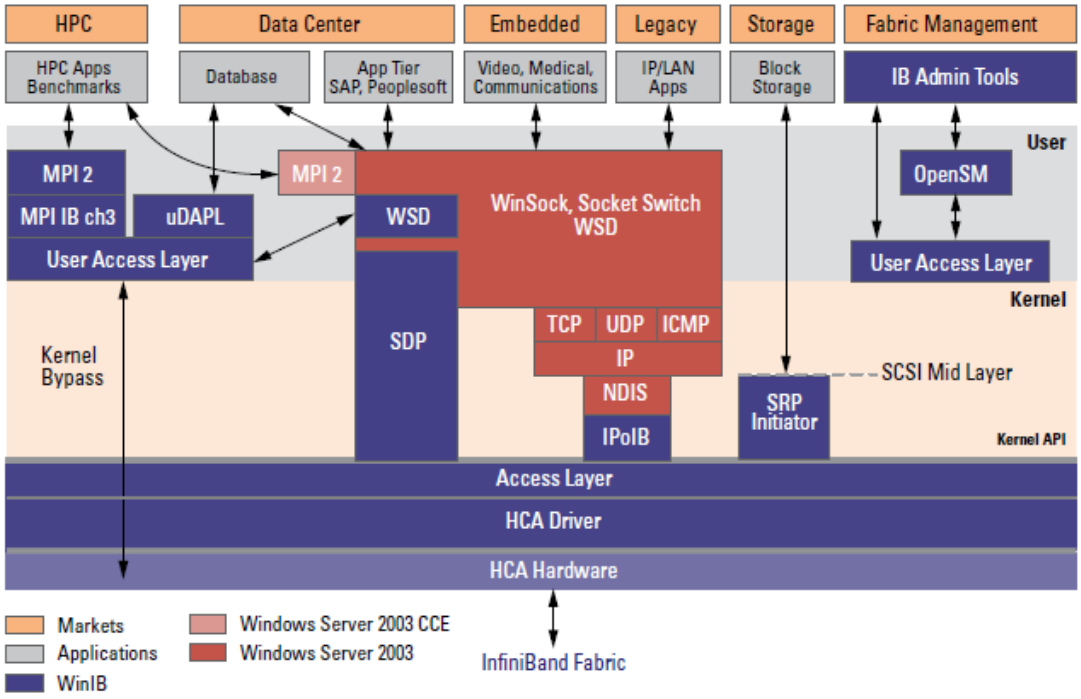
3.1 软件协议栈 OFED 介绍

为服务器和存储集群提供低延迟和高带宽的企业数据中心(EDC)，高性能计算(HPC)和嵌入式应用环境。 Mellanox 所有适配卡与基于 Open Fabrics 的 RDMA 协议和软件兼容。2004 年 OpenFabrics Alliance 成立，该组织致力于促进 RDMA 网络交换技术的发展。2005 年，OpenFabrics Alliance 发布了第一个版本

的 OFED(OpenFabrics Enterprise Distribution)。



Mellanox OFED 是一个单一的软件堆栈，包括驱动、中间件、用户接口，以及一系列的标准协议 IPoIB、SDP、SRP、iSER、RDS、DAPL(Direct Access Programming Library)，支持 MPI、Lustre/NFS over RDMA 等协议，并提供 Verbs 编程接口；Mellanox OFED 由开源 OpenFabrics 组织维护。



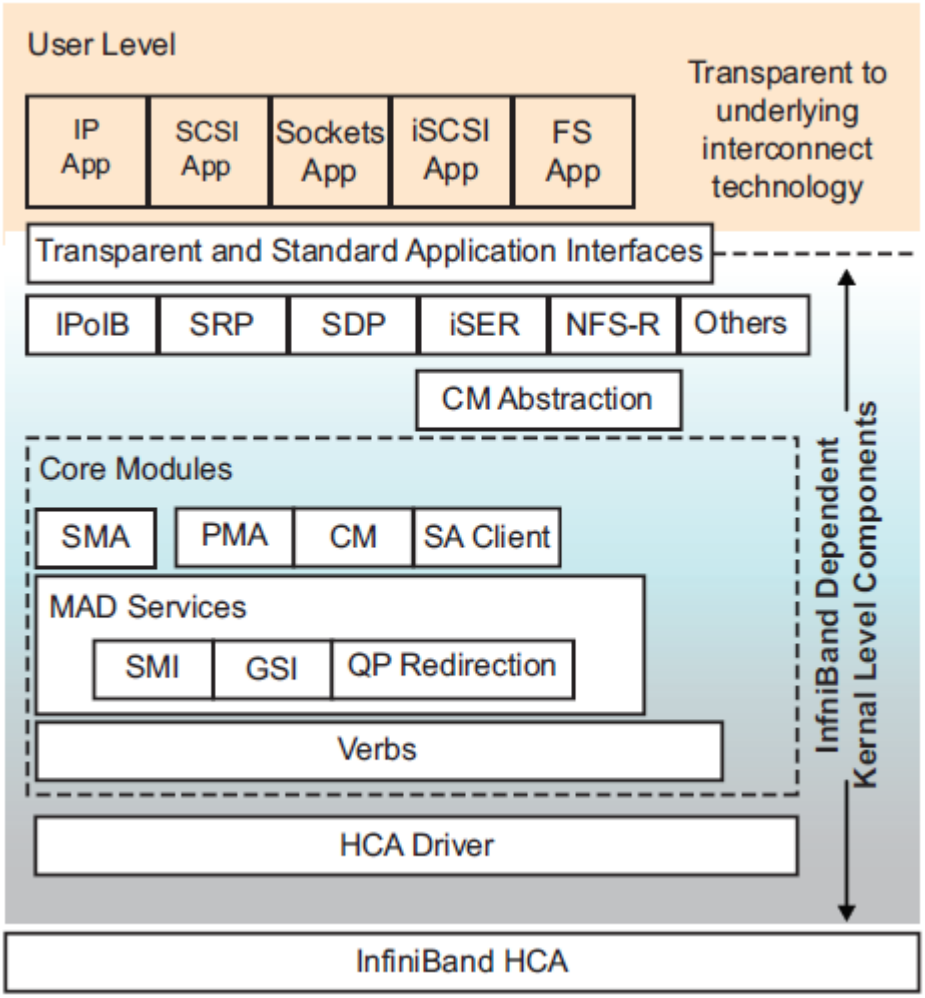
如果前面的软件堆栈逻辑图过于复杂，可以参考上面的简明介绍图。Mellanox OFED for Linux (MLNX_OFED_LINUX) 作为ISO映像提供，每个Linux发行版，包

括源代码和二进制RPM包、固件、实用程序、安装脚本和文档。
下面我们站在应用开发架构师或开发者的角度，分析、解读下 InfiniBand的架构和服务能力(简化的InfiniBand架构)。

3.2 InfiniBand 的软件架构

InfiniBand 软件栈的设计是为了简化应用部署。IP 和 TCP 套接字应用程序可以利用 InfiniBand 性能，而无需对运行在以太网上的现有应用程序进行任何更改。这同样适用于 SCSI、iSCSI 和文件系统应用程序。位于低层 InfiniBand 适配器设备驱动程序和设备独立 API(也称为 verbs)之上的上层协议提供了行业标准接口，可以无缝部署现成的应用程序。

Linux InfiniBand 软件架构。该软件由一组内核模块和协议组成。还有一些关联的用户模式共享库，这些库在图中没有显示。在用户级操作的应用程序对底层互连技术保持透明。本文的重点是讨论应用程序开发人员需要知道什么，才能使他们的 IP、SCSI、iSCSI、套接字或基于文件系统的应用程序在 InfiniBand 上运行。



对协议的操作、底层核心和 HCA 驱动程序的详细讨论超出了本文的范围。但是，

为了完整起见，下面是内核级别的简要概述，下面将介绍 InfiniBand 特定模块和协议。

内核代码逻辑上分为三层:HCA 驱动程序、核心 InfiniBand 模块和上层协议。核心 InfiniBand 模块包括 InfiniBand 设备的内核级中间层。中间层允许访问多个 HCA NICs，并提供一组公共共享服务。这些服务包括:

-用户级访问模块——用户级访问模块实现了必要的机制，允许从用户模式应用程序访问 InfiniBand 硬件。

-中间层提供以下功能:
•通信经理(CM) - CM 提供了允许客户建立连接所需的服务。
•SA 客户端——SA(子网管理员)客户端提供了允许客户端与子网管理员通信的功能。SA 包含建立连接所需的重要信息，如路径记录。
•SMA-子网管理器代理响应子网管理包，允许子网管理器在每个主机上查询和配置设备。
•PMA -性能管理代理响应允许检索硬件性能计数器的管理包。
•MAD 服务——管理数据报(MAD)服务提供一组接口，允许客户端访问特殊的 InfiniBand 队列对(QP)， 0 和 1。
•GSI -通用服务接口(GSI)允许客户端在特殊 QP1 上发送和接收管理包。
•队列对(QP)——重定向高层管理协议，通常将共享对特殊 QP 1 的访问重定向到专用 QP。这是为带宽密集型的高级管理协议所需要的。
•SMI -子网管理接口(SMI)允许客户端在特殊 QP0 上发送和接收数据包。这通常由子网管理器使用。
•Verbs-对中间层提供由 HCA 驱动程序提供的 Verbs 访问。InfiniBand 体系结构规范定义了 Vbers。Vbers 是必须提供的函数的语义描述。中间层将这些语义描述转换为一组 Linux 内核应用程序编程接口(API)。
•中间层还负责在异常程序终止或客户端关闭后，对没有释放的已分配资源的资源跟踪、引用计数和资源清理。

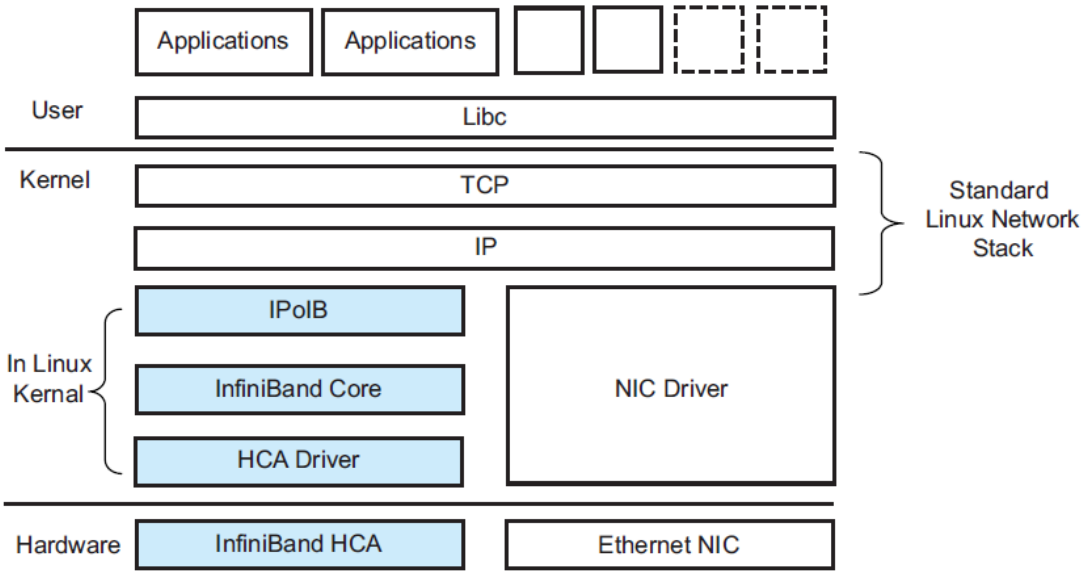
-InfiniBand 堆栈的最低层由 HCA 驱动程序组成。每个 HCA 设备都需要一个特定于 HCA 的驱动程序，该驱动程序注册在中间层，并提供 InfiniBand Verbs。

如 IPoIB, SRP, SDP, iSER 等高级协议，采用标准数据网络，存储和文件系统应用在 InfiniBand 上操作。除了 IPoIB 提供了 InfiniBand 上 TCP/IP 数据流的简单封装外，其他更高级别的协议透明地支持更高的带宽、更低的延迟、更低的 CPU 利用率和端到端服务，使用经过现场验证的 RDMA(远程 DMA)和 InfiniBand 硬件的传输技术。下面将讨论这些高级协议，以及如何快速启用现有的应用程序对 InfiniBand 进行操作。

3.2.1 IB 对基于 IP 的应用支持

在InfiniBand上评估任何基于IP的应用程序的最简单方法是使用上层协议IP

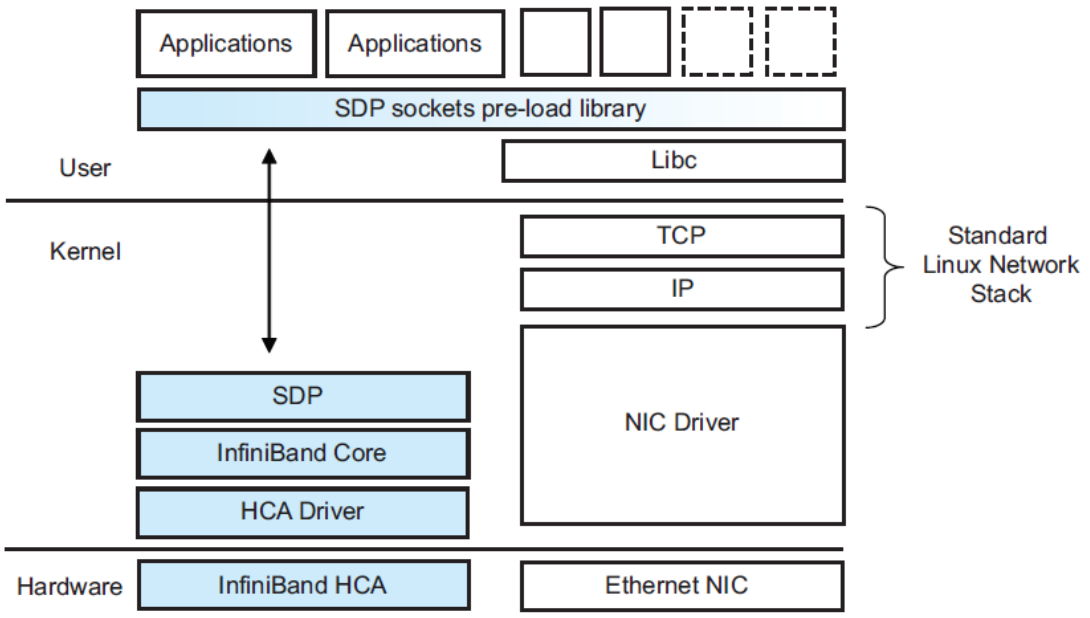
over IB (IPoIB)。在高带宽的InfiniBand适配器上运行的IPoIB可以为任何基于ip的应用程序提供即时的性能提升。IPoIB支持在InfiniBand硬件上的(IP)隧道数据包。如下图，在Linux中，协议是作为标准的Linux网络驱动程序实现的，这允许任何使用标准Linux网络服务的应用程序或内核驱动程序在不修改的情况下使用InfiniBand传输。Linux内核2.6.11及以上版本支持IPoIB协议，并对InfiniBand核心层和基于Mellanox技术公司HCA的HCA驱动程序的支持。



这种在InfiniBand上启用IP应用程序的方法对于带宽和延迟不重要的管理、配置、设置或控制平面相关数据是有效的。由于应用程序继续在标准TCP/IP网络栈上运行，应用程序完全不知道底层I/O硬件。然而，为了获得充分的性能并利用InfiniBand体系结构的一些高级特性，应用程序开发人员也可以使用套接字直接协议(SDP)和相关的基于套接字的API。

3.2.2 IB 对基于 Socket 的应用的支持

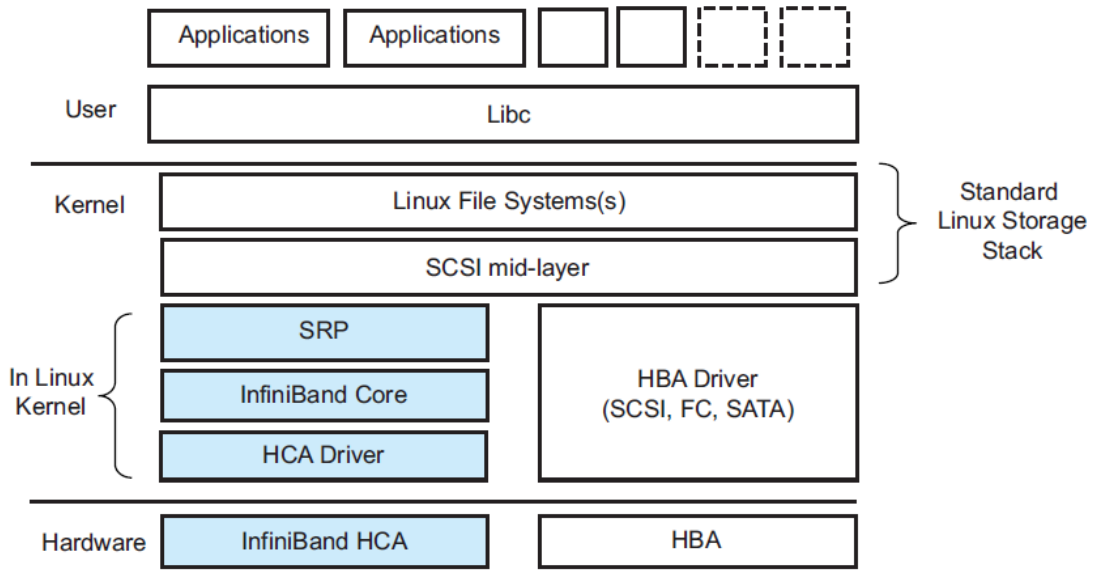
对于使用TCP套接字的应用程序，SDP或socket direct协议可以显著提高性能，同时减少CPU利用率和应用程序延迟。SDP驱动程序为标准套接字应用程序提供了高性能接口，并通过绕过软件TCP/IP栈、实现零拷贝和异步I/O以及使用高效的RDMA和基于硬件的传输机制传输数据，从而提高了性能。



InfiniBand硬件提供可靠的硬件传输基础。因此，TCP协议变得冗余，可以绕过，从而节省宝贵的CPU周期。上图描述了一个基于Linux的SDP实现。零拷贝SDP的实现可以节省内存拷贝，使用RDMA可以帮助节省昂贵的上下文切换开销，CPU利用率，性能和延迟。SDP协议采用单独的网络地址族实现，例如，TCP/IP提供AF_INET地址族，而SDP提供AF_SDP(27)地址族。为了允许标准套接字应用程序在不进行修改的情况下使用SDP，SDP提供了一个预加载库用于捕获libc套接字调用，并将其重定向到AF_INET地址族。很明显，除了预加载库的接口之外，应用程序不需要更改。

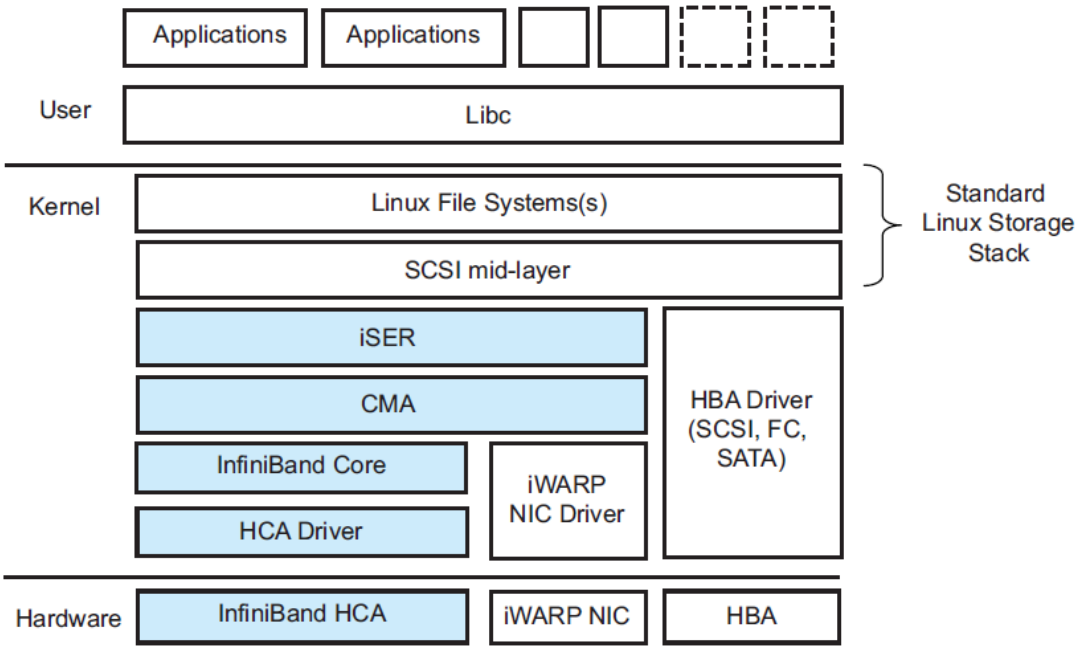
3.2.3 IB 对基于 SCSI 和 iSCSI 应用的支持

SCSI RDMA协议(SRP)由ANSI T10委员会定义，为InfiniBand体系结构提供块存储能力。SRP是一种协议，使用这种行业标准协议在InfiniBand硬件上通过SCSI请求数据包。这允许一个主机驱动程序使用来自不同存储硬件供应商的存储目标设备。



如图所示，SRP上层协议使用SCSI中间层插入Linux。因此，对于使用这些文件系统的上层Linux文件系统和用户应用程序来说，SRP将作为本地附加存储设备出现(当然，它可以物理地位于fabric上的任何位置)。SRP也作为最新的Linux内核版本的一部分。

iSER (iSCSI RDMA)通过启用零拷贝RDMA、将传输层中的CRC计算转移到硬件上、使用消息边界技术替代流来消除传统的iSCSI和TCP瓶颈。它利用iSCSI管理和发现工具，使用SLP和iSNS全局存储命名。在互联网工程专责小组(IETF)和IBTA的指导下，制定了一个附件来支持iSER over InfiniBand。



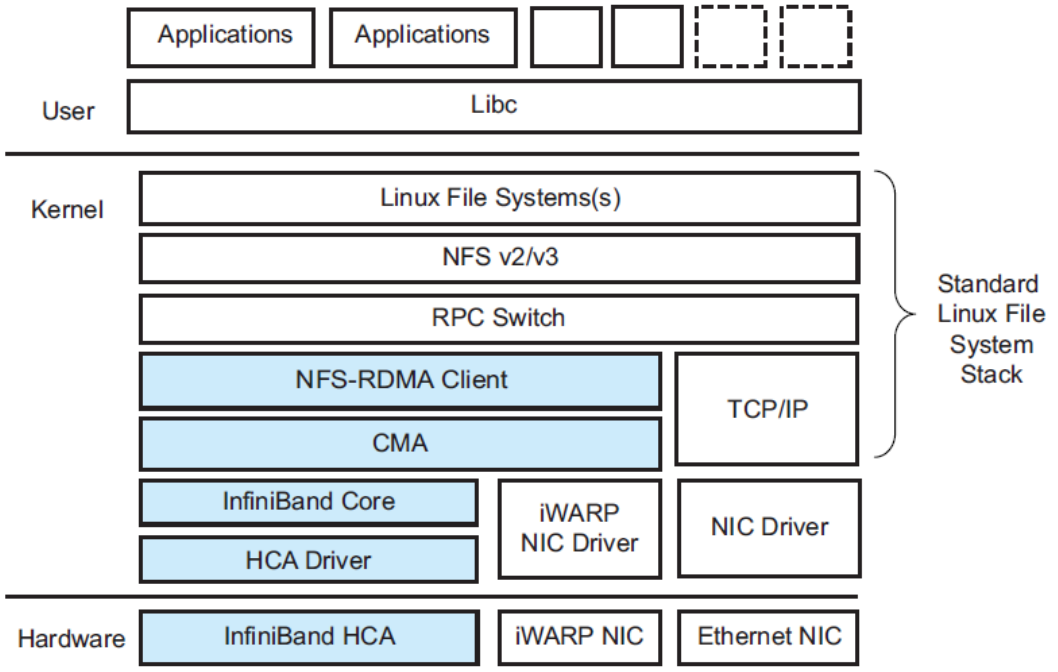
iSER也采用了使用SCSI中间层的方法插入到Linux。但是，如面图所示，iSER在额外的抽象层(CMA, Connection Manager Abstraction layer)上工作，实现对基于InfiniBand和iWARP的RDMA技术的透明操作支持。

采用LibC接口的用户应用程序和内核级采用Linux文件系统接口的应用程序是透明的，不会感知底层使用的是什么互连技术。

3.2.4 IB 对 NFS 应用的支持

基于RDMA的网络文件系统(NFS)是互联网工程工作组(IETF)正在开发的协议。这项工作的目的是扩展NFS，使其可以利用InfiniBand体系结构的RDMA特性和其他支持RDMA的Fabric。

如下图所示，NFS-RDMA客户端插入到Linux内核中的RPC交换层和标准NFS v2/v3层。RPC交换层通过NFS-RDMA客户端或TCP/IP堆栈来引导NFS通信。与iSER的实现类似,NFS-RDMA客户端通过CMA实现对基于InfiniBand和iWARP的RDMA技术的透明支持。文件系统应用程序接口映射到标准的Linux文件系统层，不会感知底层使用的是什么互连技术。



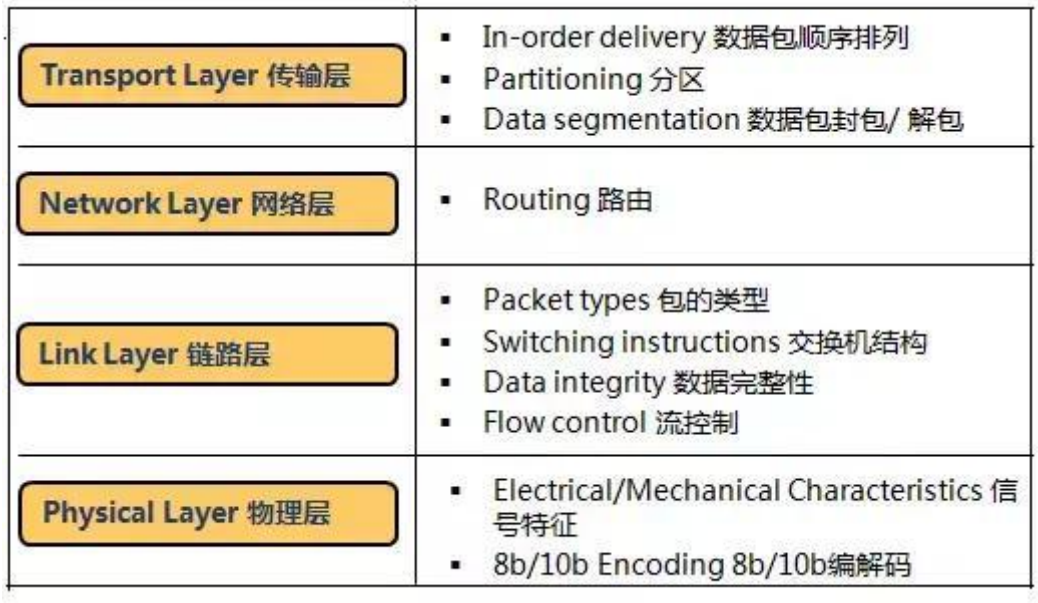
一个开源项目正在开发NFS-RDMA客户端，项目具体参见 http://sourceforge.net/projects/nfs-rdma/ 。Mellanox 提供了一个Linux NFS-RDMA 软件包（参见:http://www.mellanox.com/products/nfs_rdma_sdk.php)，可用于生产使用，并且兼容主要Linux操作系统发行版的openfabric InfiniBand软件。

InfiniBand 软件和协议主要的 Linux、Windows 版本和虚拟机监控程序(Hypervisor)平台上得到了支持和支持。这包括 Red Hat Enterprise Linux、SUSE Linux Enterprise Server、Microsoft Windows Server 和 Windows CCS(计

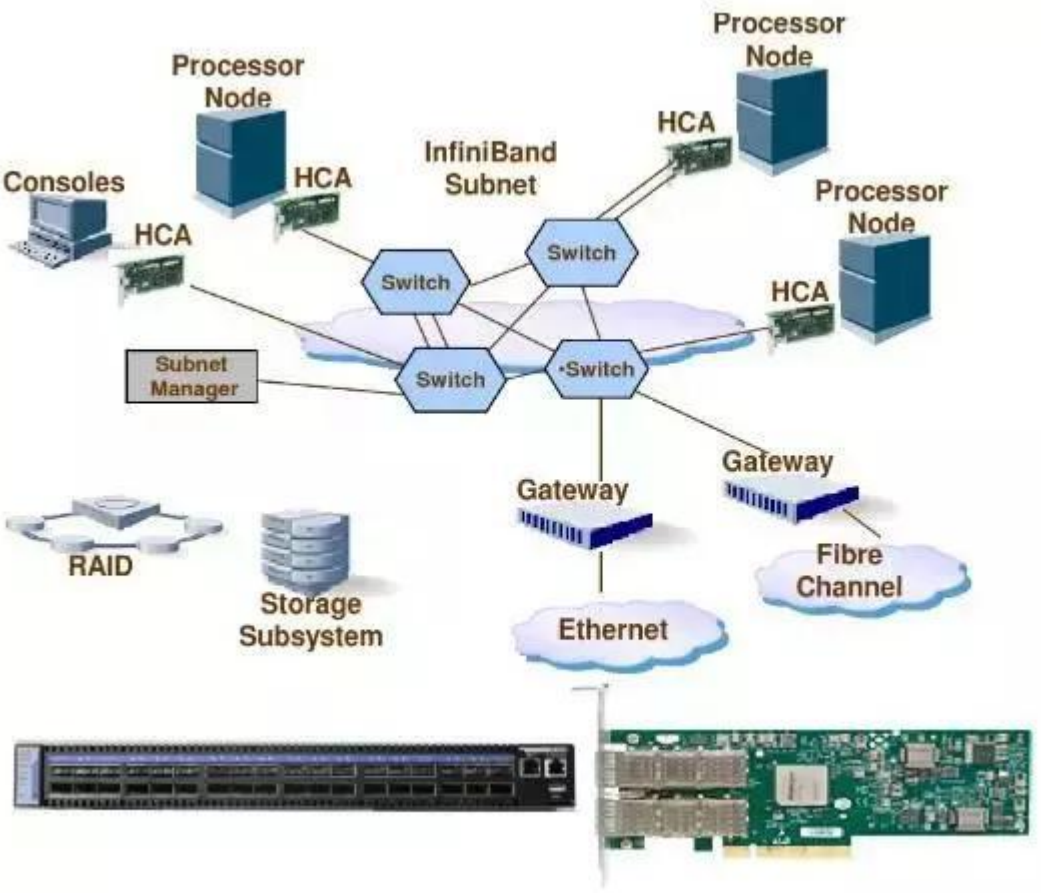
算集群服务器)以及 VMware 虚拟基础设施平台。

3.3 InfiniBand 网络和拓扑组成

InfiniBand 结构基于信道的串口替代共用总线，从而使 I/O 子系统和 CPU/内存分离。所有系统和节点可通过信道适配器逻辑连接到该结构，它们可以是主机、适配器(HCA)或目标适配器(TCA)，还包括 InfiniBand 交换机和路由器扩展，从而满足不断增长的需求。



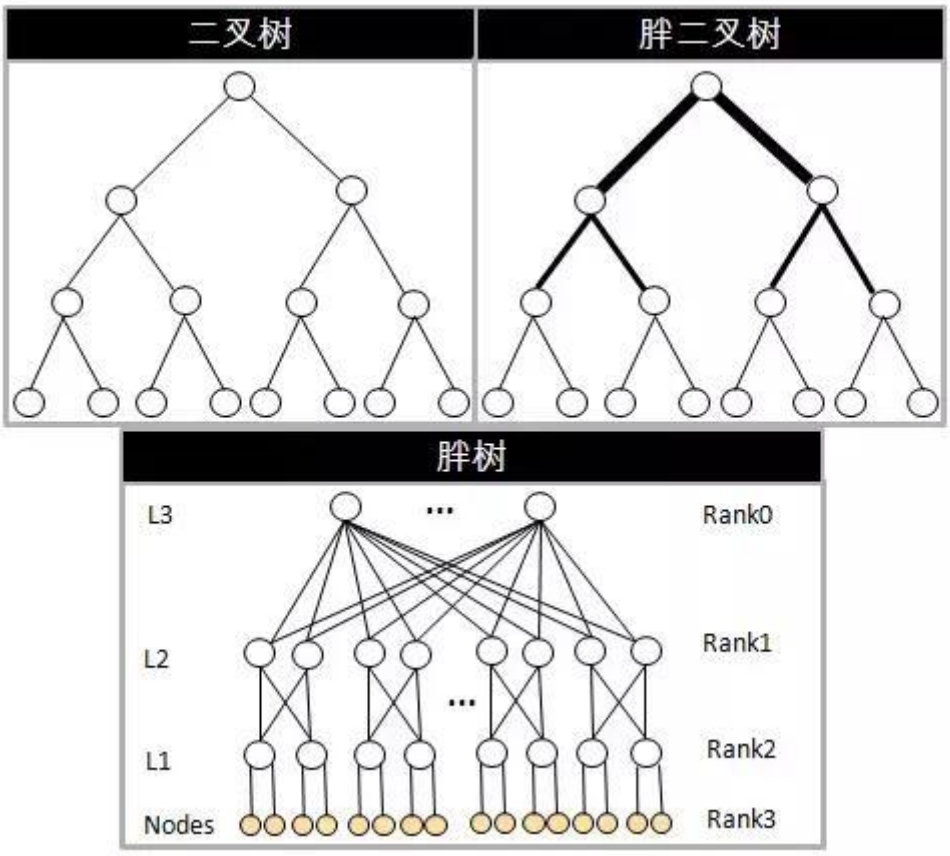
InfiniBand 也是一种分层协议(类似 TCP/IP 协议)，每层负责不同的功能，下层为上层服务，不同层次相互独立，每一层提供相应功能。InfiniBand 协议可满足各种不同的需求，包括组播、分区、IP 兼容性、流控制和速率控制等。



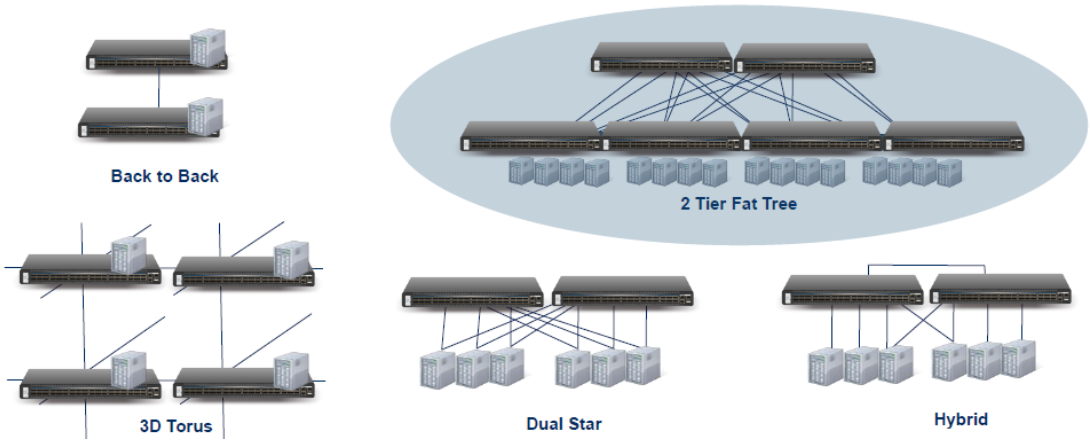
InfiniBand 网络路由算法包括最短路径算法、基于 Min Hop 的 UPDN 算法和基于 Fat Tree 组网 FatTree 算法。

算法在一定程度上也决定了 InfiniBand 网络拓扑结构，尤其在高性能计算、大型集群系统，必须要考虑网络之间的拓扑结构，网络上行和下行链路阻塞情况也决定着整个网络性能。由于树形拓扑结构具备清晰、易构建和管理的有点，故而胖树网络拓扑结构常常被采用，以便能够发挥出 InfiniBand 网络优势，也通常应用在无阻塞或阻塞率很低的应用场景，所以我们下面我们重点讨论下。

在传统的三层组网架构中(二层架构也经常用到)，由于接入层节点数量庞大，所以要求汇聚层或核心层的网络带宽和处理能力与之匹配，否则设计出来的网络拓扑结构就会产生一定的阻塞比。



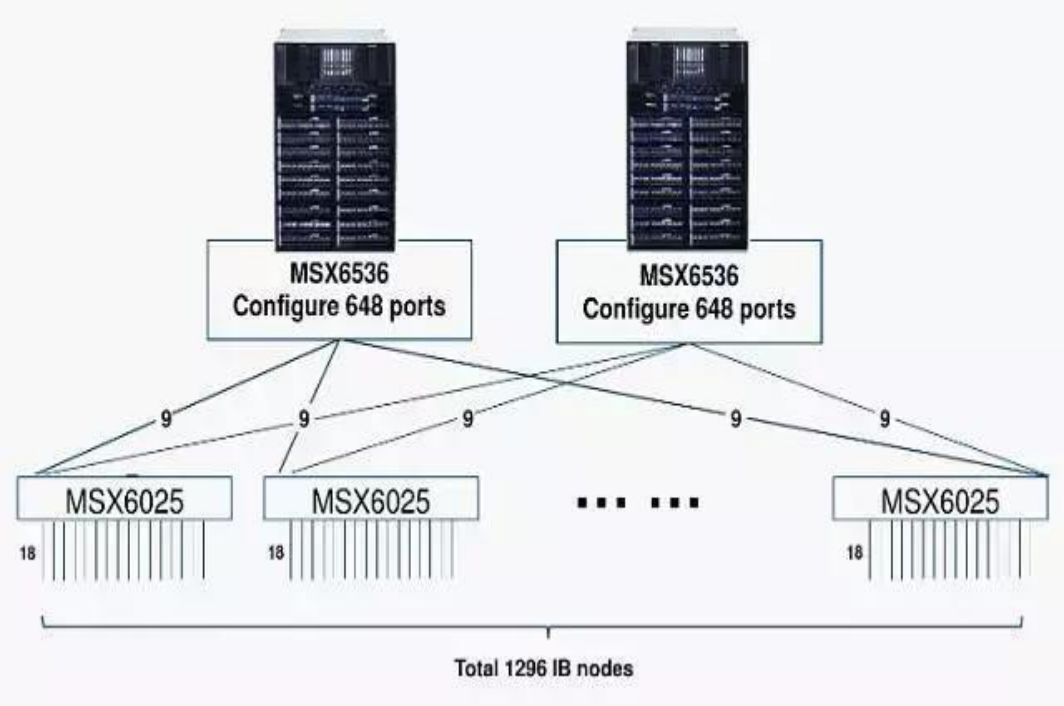
为了解决这一问题，在汇聚层和核心层就要采用胖节点组网(如果采用瘦节点就一定发生阻塞，且三层组网阻塞比二层组网更加严重)，如上图胖二叉树事例，胖节点(Fat Tree)必须提供足够的网络端口和带宽与叶子节点匹配。



采用胖树拓扑网络的结构一般由叶子(Leaf)和主干(Spine)交换机组成，叶子

交换机与服务器或存储等信道适配卡相连，分配一部分端口给节点，另一部分端口被接入网络中。在 InfiniBand 网络中 Fat Tree 组网结构具有下面几个特点。

- 连接到同一端 Switch 的端口叫端口组，同一 Rank 级别的 Switch 必须有相同的上行端口组，且根 Rank 没有上行端口组；除了 Leaf Switch，同一 Rank 的 Switch 必须有相同的下行端口组。
- 同一 Rank 的每个上行端口组中端口个数相同;且同一 Rank 的每个下行端口组中端口个数也相同。
- 所有终端节点的 HCA 卡都在同一 Rank 级别上。



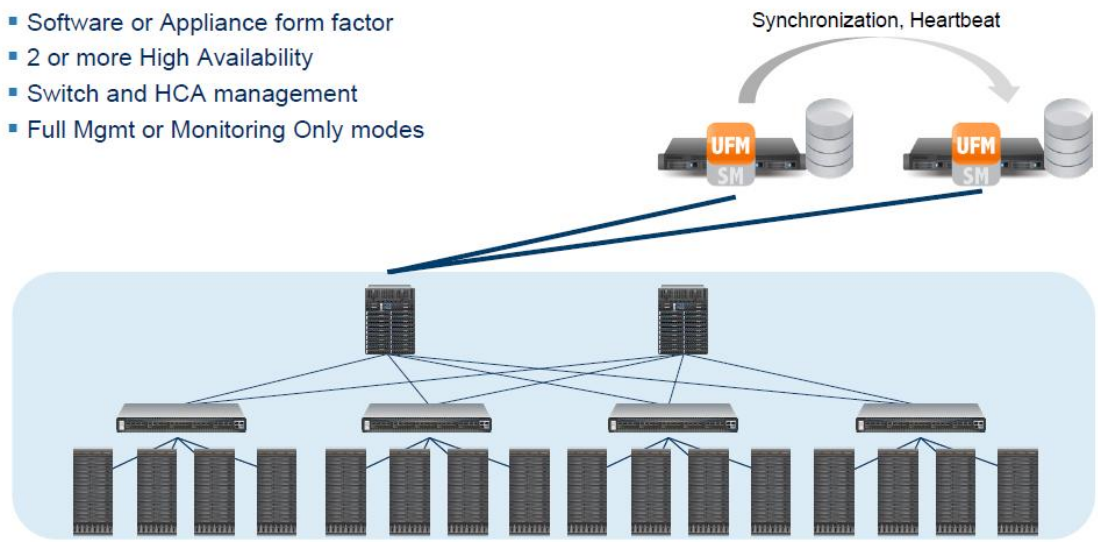
上图是一个采用二层架构的无阻塞 Fat Tree 组网事例，接入层下行提供 1296 个 IB 端口给服务器或存储适配卡，上行也提供适配器给汇聚层。但从一个接入 IB 交换机来看，上行和下行分别提供 18 个接口实现无阻塞组网。胖树拓扑结构一方面提供费阻塞数据传输，另一方面提供网络冗余增强网络可靠性。

3.4 InfiniBand 网络管理

OpenSM 软件是符合 InfiniBand 的子网管理器(SM)，运行在 Mellanox OFED 软件堆栈进行 IB 网络管理，管理控制流走业务通道，属于带内管理方式。

OpenSM 包括子网管理器、背板管理器和性能管理器三个组件，绑定在交换机

内部的必备部件。提供非常完备的管理和监控能力，如设备自动发现、设备管理、Fabric 可视化、智能分析、健康监测等等。



Mellanox 还提供 Unified Fabric Manager(UFM)的软件，这是一个强大的平台来管理 InfiniBand 计算环境。

3.5 InfiniBand 并行计算集群

MPI (Message Passing Interface) 用于并行编程的一个规范，并行编程即使用多个 CPU 来并行计算，提升计算能力。Mellanox OFED for Linux 的 InfiniBand MPI 实现包括 Open MPI 和 OSU MVAPICH。

Open MPI 是基于 Open MPI 项目的开源 MPI-2 实现，OSU MVAPICH 是基于俄亥俄州立大学的 MPI-1 实施。下面列出了一些有用的 MPI 链接。

MPI Standard	http://www-unix.mcs.anl.gov/mpi
Open MPI	http://www.open-mpi.org
MVAPICH MPI	http://nowlab.cse.ohio-state.edu/projects/mpi-iba/
MPI Forum	http://www.mpi-forum.org

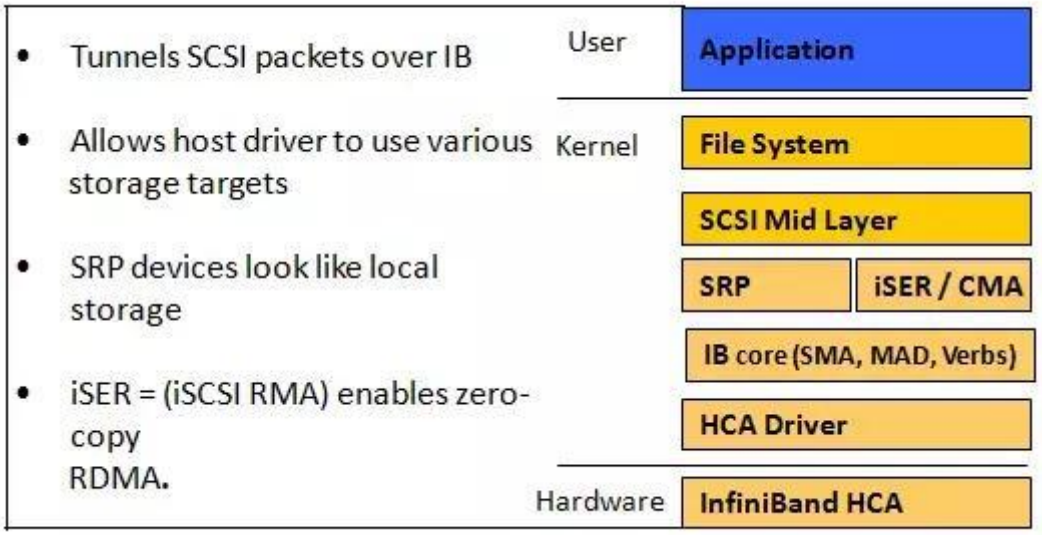
RDS (Reliable Datagram Socket)是一种套接字 API，在 sockets over RC or TCP/IP 之间提供可靠的按顺序数据报传送，RDS 适用于 Oracle RAC 11g。

IPoIB/ EoIB (IP/Eth over InfiniBand) 是通过 InfiniBand 实现的网络接口实现，IPoIB 封装 IP 数据报通过 InfiniBand 连接或数据报传输服务。

SDP (Socket Direct Protocol) 是提供 TCP 的 InfiniBand 字节流传输协议流语义，利用 InfiniBand 的高级协议卸载功能，SDP 可以提供更低的延迟更高带宽。

3.6 InfiniBand 的存储支持能力

支持 iSER (iSCSI Extensions for RDMA)和 NFSoRDMA (NFS over RDMA)，SRP (SCSI RDMA Protocol) 是 InfiniBand 中的一种通信协议，在 InfiniBand 中将 SCSI 命令进行打包，允许 SCSI 命令通过 RDMA(远程直接内存访问)在不同的系统之间进行通信，实现存储设备共享和 RDMA 通信服务。

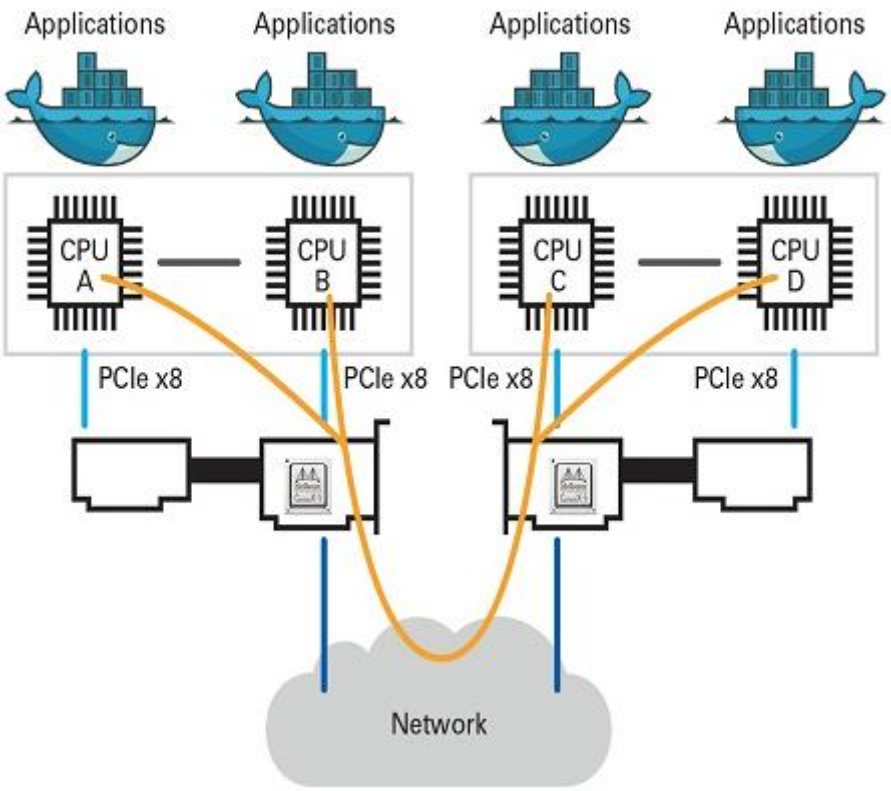


3.7 InfiniBand 对 RDMA 技术支持

RDMA (Remote Direct Memory Access)技术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 通过网络把数据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功能。它消除了外部存储器复制和文本交换操作，因而释放内存带宽和 CPU 周期用于改进应用系统性能。

第四章 Mellanox Socket Direct 技术

本节涉及的 Socket Direct 特指 Mellanox 公司的 Socket Direct HCA 卡，不是 OpenFabric 协议栈中的 Socket Direct Protocol。 Mellanox 公司针对当前服务器中普遍应用的 dual socket 结构提供 Socket Direct 的方案。



其基本原理如上图所示,将 PCIe x16 的 HCA 卡分成 2 张 PCIe x8 卡(Main Card 和 Auxilary Card),并连接到不同 Socket 上,原本需要通过 inter-processor bus 的通信可以直接通过 HCA 卡进行，从而减少 CPU 间的通信，提升系统性能。

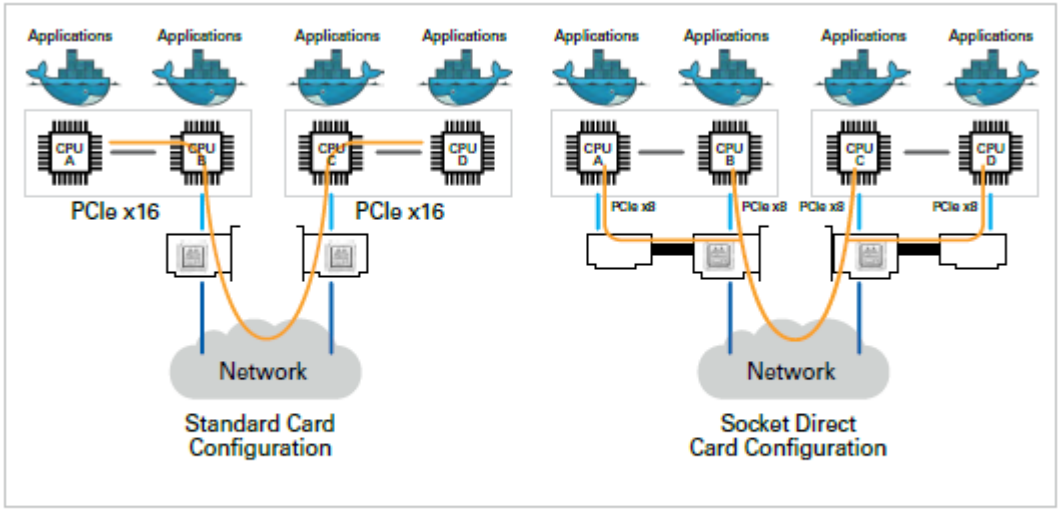
4.1 Socket Direct 技术原理

Mellanox Socket Direct 可以把两张 PCIe 卡通过一种独特网络组网形态，实现把 PCIe 通道分割在两张 PCIe 卡之间网络技术。PCIe 适配器卡为多路服务器带来的一个关键好处是消除了多路 CPU 之间通过内部总线进行的网络流量,从而显著降低了开销和延迟。

下图显示了 Mellanox Socket Direct 适配器的图片，该方案不但有效地集成了主板上的单个网络适配器，同时集成了一个辅助的 PCIe 连接卡和连接二者的 SAS 线缆。

Socket-Direct Adapter (Front and Back Angles)

Socket Direct 如何工作？当把两个 PCIe 插槽直接连接到两个 CPU 插槽，并启用 Socket Direct 功能时，该方案允许每个 CPU 通过其专用的 PCIe 接口直接访问网络。



Standard Card Configuration versus Socket Direct Card

4.2 Socket Direct 和标卡测试对比

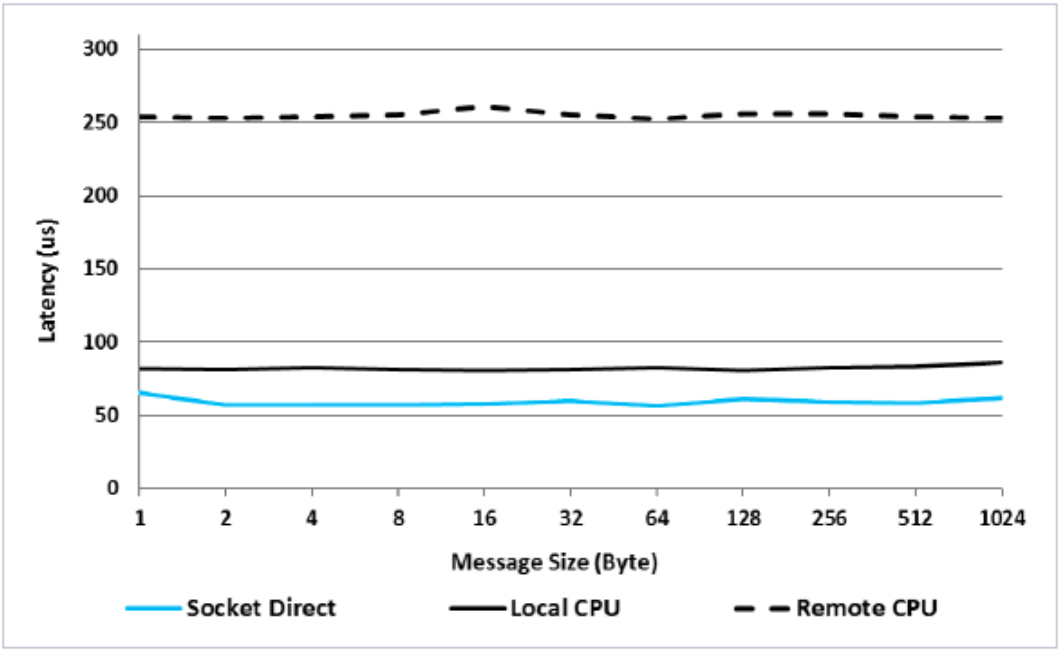
测试比较了基于 ConnectX 的 Socket Direct (安装在双路服务器中)和标准 PCIe x16 100Gb/s 适配器卡的性能(仅连接到一个 CPU)。测试范围覆盖 TCP 吞吐量、延迟和 CPU 利用率，以及 RDMA 基准测试。

标准卡配置:上图(左)中显示了一个由双路服务器组成的基本网络，其中安装了一张网卡。金线显示了在服务器之间跨网络数据包的流量路径。流量路径包含多个数据处理节点(CPU-A、CPU-B、CPU-C 和 CPU-D)。流量可以在流路径上的任意两个节点之间传递。

Socket Direc 卡配置:上图(右)中显示，CPU-A 和 CPU-D 可以绕过处理器间总线，直接访问网络控制器。

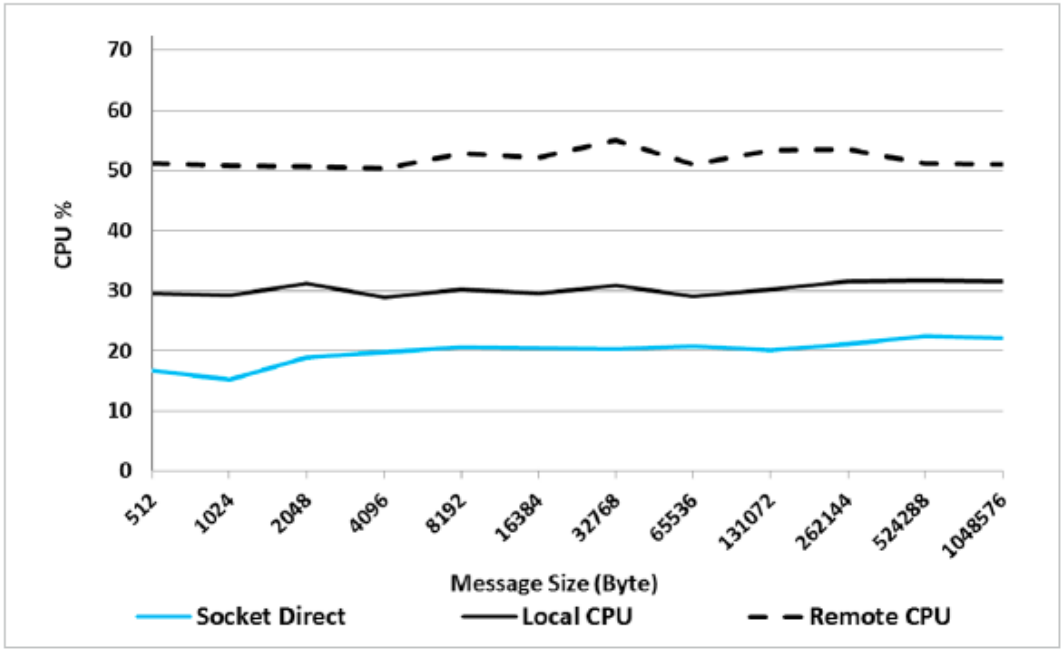
对比测试用例：

- Test1 、本地 CPU(上图左):数据流量路径从 CPU- B 开始，到 CPU- C 结束。
- Test2 、远程 CPU(上图左)：数据流量路径从 CPU- A 中启动，通过处理器间总线到达 CPU- B，通过网络到达 CPU-Cc，然后再次通过处理器间总线到达 CPU- D。
- Test3 、Socket Direct(上图右)：数据流量路径从 CPU-A 和 CPU-B 通过网络到达另一个服务器上的 CPU-C 和 CPU-D。



Average Latency

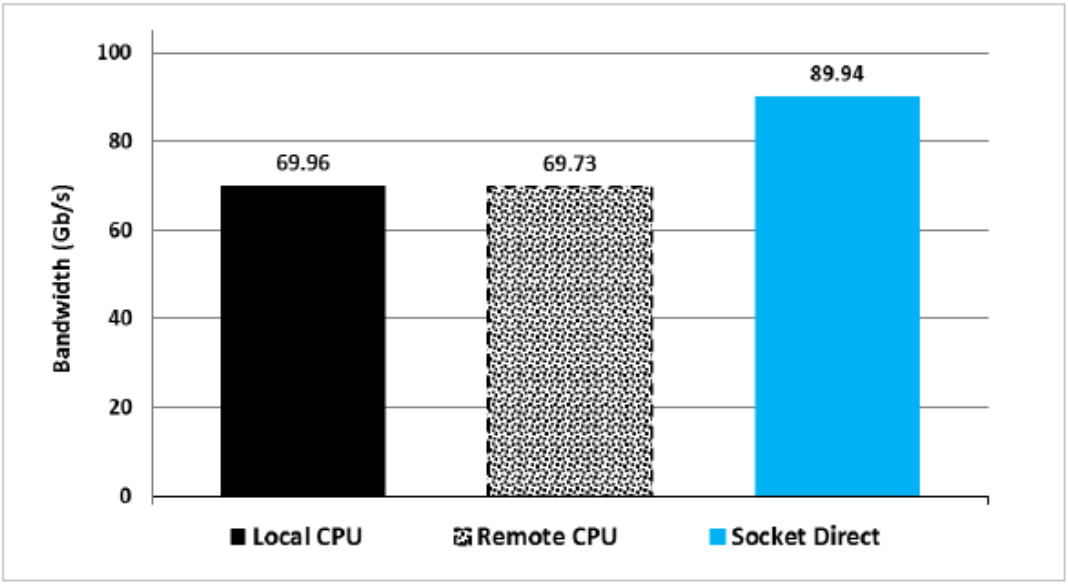
上图比较了 Socket Direct 适配器与标准网络适配器的平均延迟。该图显示，与标准适配器组网相比，使用 ocket Direct 适配器时，延迟减少了 80%。由于 CPU 传输的数据流量套接字都采用了直接路径来访问网络，并且在 CPU 之间均匀地分布 TCP 流，所以降低了网络时延。



Host CPU Utilization (Inter-processor Load Applied)

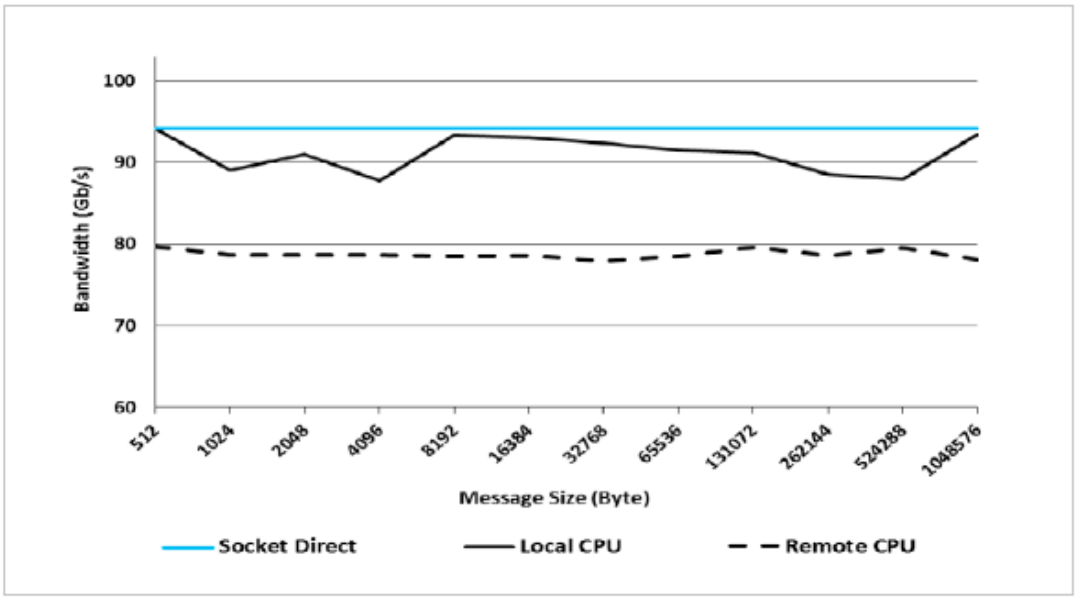
上图显示了 CPU 利用率，很明显，使用 Socket direct 直接访问网络也可以提高 50%的 CPU 利用率。此外，与标准网络适配器相比，TCP 数量流量均匀分布减少了两个 CPU 上的平均缓存计数操作，这进一步提高了 CPU 利用率。

在实际的场景中，在双路服务器上运行的应用程序遍历 CPU(通过处理器间通信总线)的数据。为了更真实地测量网络性能，我们在处理器间总线上增加一个人工负载，然后测量这个负载对服务器外部数据流量的影响。我们两种类型的适配器(标准适配器和 Socket Direct)进行测量和对比。



Load on the Inter-processor Bus

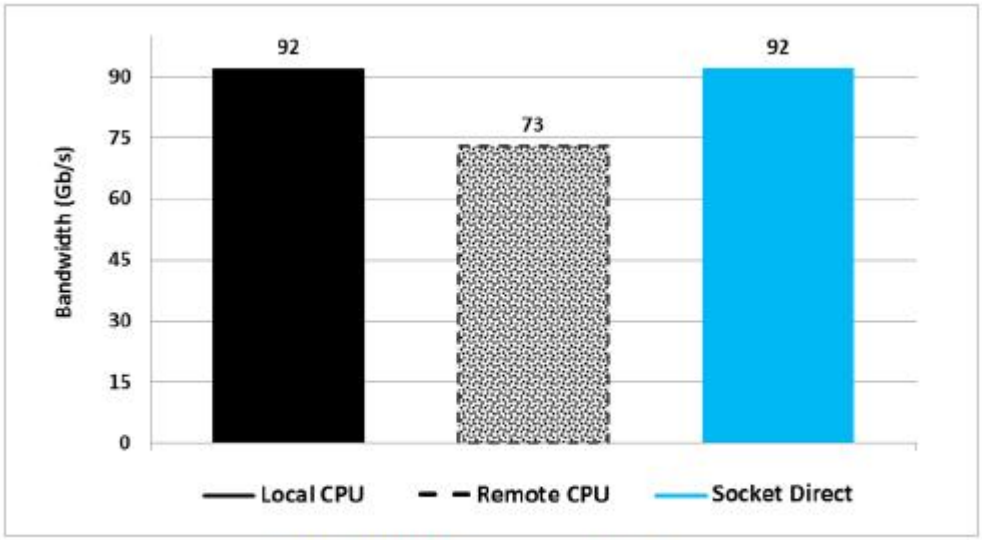
测试表明，CPU 之间可以通过的流量越多，CPU 之间的通信效率就越高。上图显示了服务器处理器间通信数据流量。很明显，当使用 Socket Direct 适配器时，就有更多的数据流量设法在 CPU 之间传递。



Ethernet Throughput (Inter-processor Load Applied)

在应用处理器间总线上增加一个人工负载负载时，比较服务器的外部吞吐量，很明显发现(如上图)，相比标准的适配器连接，Socket Direct 适配器组网方式下吞吐量提高了 16%-28%。

我们测试了使用 Socket Direct 适配器对 RDMA 工作负载的影响。下图显示了测试的结果，相比远程 CPU 上的吞吐量，Socket Direct 适配器方案高出 25%的吞吐量。



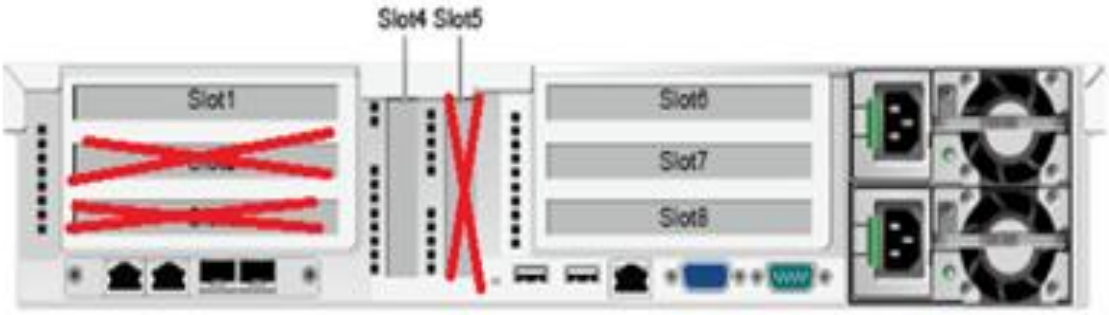
RoCE RDMA Throughput

Mellanox Socket Direct 适配器为最苛刻的应用提供了最高的性能和最灵活的解决方案。Socket Direct 通过最大的吞吐量来扩展服务器性能和利用率。多路或双路服务器中的 Socket Direct 适配器允许两个 CPU 直接连接到网络，从而提供更低的延迟、降低 CPU 利用率，提供更高的网络吞吐量。

4.3 Socket Direct 硬件安装

Socket Direct HCA card

服务器 Server PCIe slot



注意 Main Card 插在 Slot4，Auxilary Card 插在 Slot6. 插反会导致 OpenSM 无法正常建立 subnet。使用 EDR 3m IB Cable 连接 2 张 IB 卡，注意需连接同一端口。安装成功后， 过一段时间端口会亮起来。也可以通过 lspci 命令查看。

```
[root@localhost runfiles]# lspci |grep Mellanox
0002:e9:00.0 Infiniband controller: Mellanox Technologies MT27800
Family [ConnectX-5]
0002:e9:00.1 Infiniband controller: Mellanox Technologies MT27800
Family [ConnectX-5]
000a:11:00.0 Infiniband controller: Mellanox Technologies MT27800
Family [ConnectX-5]
000a:11:00.1 Infiniband controller: Mellanox Technologies MT27800
```

```
Family [ConnectX-5]
```

4.4 MLNX_OFED 安装

```
tar zxvf MLNX_OFED_* && cd MLNX_OFED_*
./mlnxofedinstall --add-kernel-support
重启机器
service openibd start
service opensm start
```

Notes:

1) 检查 kernel 版本是否与 MLNX_OFED 版本（.supported_kernels）一致，若不一致，需加上-add-kernel-support 选项，通过 src RPM 包重新编译各模块。

2) MLNX_OFED 无法制定安装目录，很多包安装在非共享目录下，所以需要给各个子节点单独安装。为方便安装，可解压到共享目录下，再通过以下脚本安装：

3)

```
#!/bin/bash

for node in $ALLNODE
do
  ssh node “cd $PWD && ./mlnxofedinstall --add-kernel-support &”
done
```

4) 可通过/etc/infiniband/openib.conf 调整 openibd 加载的各个模块。

5) 只需一个节点运行 service opensm start.

6) 实际只要在一个子节点安装好驱动即可，其他子节点都可以公用这个驱动。（**即只要在主节点和任意一个子节点上各安装一次驱动**）。

安装完成后，可使用 mst,ibdev2netdevice，ibstat，ibhosts 等命令查看 IB 状态。

```
[root@localhost runfiles]# ibdev2netdev -v
0002:e9:00.0 mlx5_0 (MT4119 - MCX556M-ECAT-S25SN) CX556M - ConnectX-5 QSFP28 fw 16.22.1002 port 1 (ACTIVE) ==> ib0 (Down)          //未配置 IPoIB
0002:e9:00.1 mlx5_1 (MT4119 - MCX556M-ECAT-S25SN) CX556M - ConnectX-5 QSFP28 fw 16.22.1002 port 1 (DOWN  ) ==> ib1 (Down)
000a:11:00.0 mlx5_2 (MT4119 - MCX556M-ECAT-S25SN) CX556M - ConnectX-5 QSFP28 fw 16.22.1002 port 1 (ACTIVE) ==> ib2 (Down)
000a:11:00.1 mlx5_3 (MT4119 - MCX556M-ECAT-S25SN) CX556M - ConnectX-5 QSFP28 fw 16.22.1002 port 1 (DOWN  ) ==> ib3 (Down)
[root@localhost runfiles]# mst status -v
```

```
MST modules:
------------
    MST PCI module is not loaded
    MST PCI configuration module is not loaded
PCI devices:
------------
DEVICE_TYPE                MST         PCI                   RDMA
NET                        NUMA
ConnectX5(rev:0)             NA          0002:e9:00.1        mlx5_1
net-ib1                    0
ConnectX5(rev:0)             NA          000a:11:00.0        mlx5_2
net-ib2                    2
ConnectX5(rev:0)             NA          000a:11:00.1        mlx5_3
net-ib3                    2
ConnectX5(rev:0)             NA          0002:e9:00.0        mlx5_0
net-ib0                    0
          //Main Card 和 Auxilary Card 连接在不同的 CPU Socket 上
```

通过 ib_send_bw, ib_send_lat 测试带宽和时延。

```
[root@localhost runfiles]# ib_send_bw -a -d mlx5_0 --report_gbits & ssh
126.26.136.114 "ib_send_bw -a -d mlx5_0 126.26.136.113 > /dev/null"
[1] 43769

************************************
* Waiting for client to connect... *
************************************
---------------------------------------------------------------------------
------------------
                    Send BW Test
 Dual-port       : OFF          Device         : mlx5_0
 Number of qps   : 1            Transport type : IB
 Connection type : RC           Using SRQ      : OFF
 RX depth        : 512
 CQ Moderation   : 100
 Mtu             : 4096[B]
 Link type       : IB
 Max inline data : 0[B]
 rdma_cm QPs     : OFF
 Data ex. method : Ethernet
---------------------------------------------------------------------------
------------------
 local address: LID 0x02 QPN 0x134f PSN 0x99f898
 remote address: LID 0x01 QPN 0x0ccb PSN 0xe6ec07
---------------------------------------------------------------------------
```

```
---------------------
 #bytes       #iterations      BW peak[Gb/sec]      BW average[Gb/sec]
MsgRate[Mpps]
 2              1000               0.000000                 0.012035
0.752209
 4              1000               0.000000                 0.026141
0.816918
 8              1000               0.000000                 0.052748
0.824182
 16             1000                  0.00                  0.10
0.812434
 32             1000                  0.00                  0.21
0.819731
 64             1000                  0.00                  0.42
0.820113
 128            1000                  0.00                  0.84
0.819873
 256            1000                  0.00                  1.61
0.784673
 512            1000                  0.00                  3.35
0.817817
 1024           1000                  0.00                  6.70
0.817390
 2048           1000                  0.00                 13.29
0.811104
 4096           1000                  0.00                 26.39
0.805417
 8192           1000                  0.00                 52.42
0.799865
 16384          1000                  0.00                 57.52
0.438832
 32768          1000                  0.00                 58.62
0.223609
 65536          1000                  0.00                 58.97
0.112485
 131072         1000                  0.00                 59.12
0.056377
 262144         1000                  0.00                 59.16
0.028211
 524288         1000                  0.00                 59.18
0.014109
 1048576        1000                  0.00                 59.19
0.007057
 2097152        1000                  0.00                 59.19
```

```
0.003528
 4194304       1000                    0.00                          59.17
0.001764
 8388608       1000                    0.00                          59.12
0.000881
-----------------------------------------------------------------------
------------------
[1]+  Done                    ib_send_bw -a -d mlx5_0 --report_gbits
```

单端口的带宽为 59Gbs 左右。

4.5 HPC-X 软件包安装

```
cd /opt/ohpc/pub/mpi/ $$ tar zxvf /path/to/hpcx-v2.1.0*
module use hpcx-v2.1.0*/modulefiles
module load hpcx
```

Notes:

1) HPC-X 解压后即可使用，所以建议放在共享目录下。

2) HPC-X 提供的 OpenMPI 不支持多线程，若有该需求，需重新编译。编译过程如下

```
tar zxvf $HPCX_HOME/sources/openmpi*.tar.gz && cd openmpi*
./configure --prefix=${HPCX_HOME}/hpcx-ompi \
   --with-knem=${HPCX_HOME}/knem \
   --with-ucx=${HPCX_HOME}/ucx \
   --with-hcoll=${HPCX_HOME}/hcoll \
   --with-platform=contrib/platform/mellanox/optimized
make -j9 all && make -j9 install
```

3) HPC-X 默认使用 hcoll 和 ucx（可通过 ompi_info 查看优先级）。

```
mpirun -allow-run-as-root -mca coll_hcoll_enable 1 -mca pml ucx -x
UCX_NET_DEVICES=mlx5_0:1,mlx5_2:1 -n 2 myapp
```

4.6 安装常见问题解答

1. IB 卡未识别

   检查硬件安装，使用 lspci 查看.

   注意 IB 卡接口和线的接口是否松动,如果松动要重新插紧。如果重新插紧后，IB 卡仍然未被识别(端口指示灯不亮)，可将此 IB 卡换到 IB 卡状态正常的机器上。如果 IB 卡工作正常的机器换上此卡后 IB 卡不能被识别，则说明此 IB 卡很可能坏了，建议更换新的 IB 卡。

2. IB 卡状态一直是 Link Down

1） 确保连线正确，端口指示灯亮。

2） 使用 service openibd status 检查节点是否加载驱动模块。

4) 使用 service opensmd restart 重启 subnet manager

3. IB 卡带宽很低

```
mlxconfig -d /dev/mst/mt4119_pciconf0 set PCI_WR_ORDERING=1
```

4. 安装驱动时提示”MLNX_OFED_LINUX does not have drivers available for this kernel”

1)执行 mlnxofedinstall 时加上选项 –add-kernel-support
2)如果加上参数后安装过程很慢，可先执行 mlnx_add_kernel_support.sh 脚本以生成适合当前内核的安装包，再使用新生成的安装包安装。

1. 驱动安装过程很慢，或安装进度长时间卡住
驱动正常安装完成的时间应在半小时之内。
如果安装时间过长，则可能是安装所需的存放临时文件路径的磁盘空间不够（可通过 df -h 查看）。驱动安装默认的生成临时文件的路径为/tmp,若该路径所在磁盘的可用空间小于 5G，则应改变临时文件存放路径。
更改临时文件存放路径：执行 mlnxofedinstall 时加上选项 --tmpdir <tmp dir>

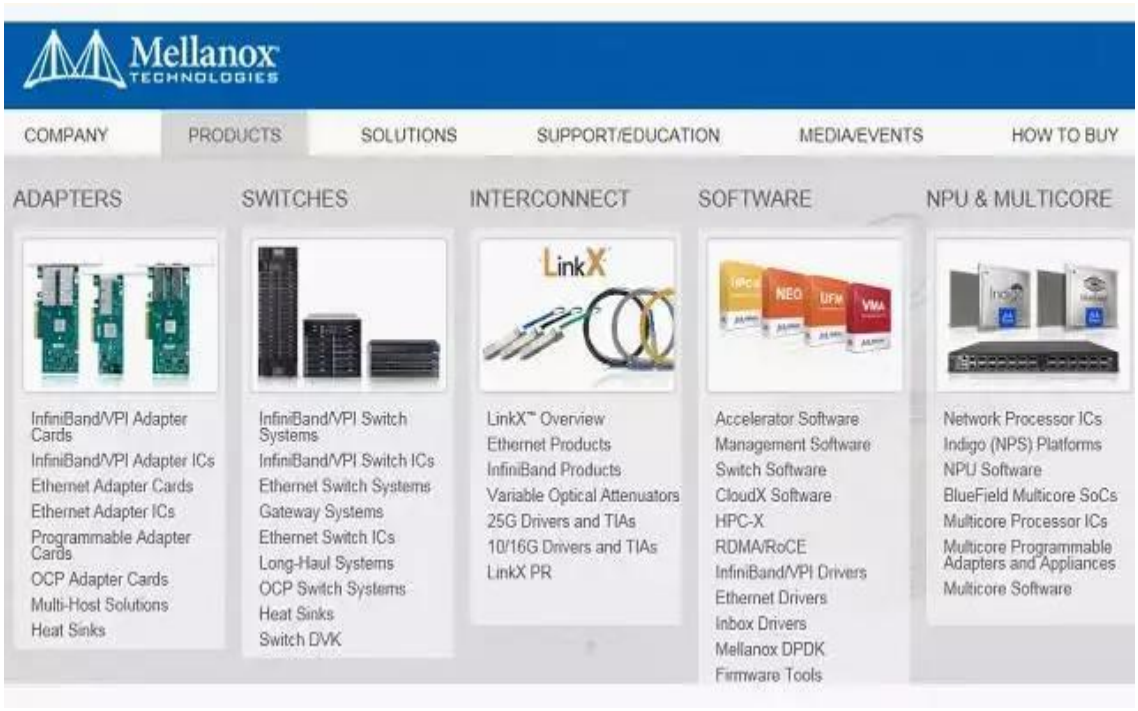
2. 在使用 ib_send_bw 测试时提示 Socket 初始化失败
可能是 ib_send_bw 使用的默认端口被占用，可加上-p 选项指定端口号。

3. KNEM 组件编译的结果是内核模块，与 linux 内核版本强依赖。当前 knem 的 1.1.3 最新版本还未支持内核>=4.15 版本，需要通过修改代码后编译。

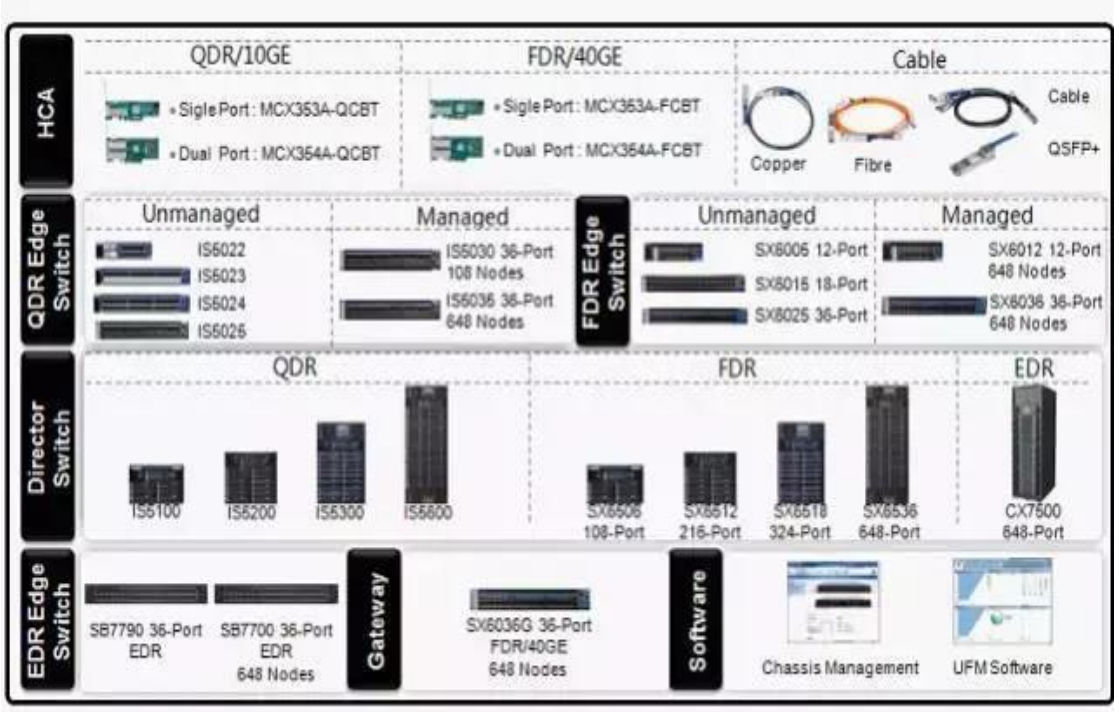
第五章 InfiniBand 主要产品和特性

5.1 Mellanox 主要产品介绍

Mellanox 是服务器和存储端到端连接解决方案的领先供应商，一直致力于 InfiniBand 和以太网互联产品的研发工作，也是业界公认的超高速网络典型代表。下面我们重点看看 InfiniBand 和相关产品介绍。



InfiniBand 产品搭配先进的 VPI 技术使得单端口适配业务需求，主要产品包括 VPI 系列网卡、交换机。芯片产品也是保障所有系列产品的可靠基石。种类丰富的线缆是实现高速互联网络的重要保证。除了硬件外，InfiniBand 配套加速软件和统一管理软件丰富整个产品家族。

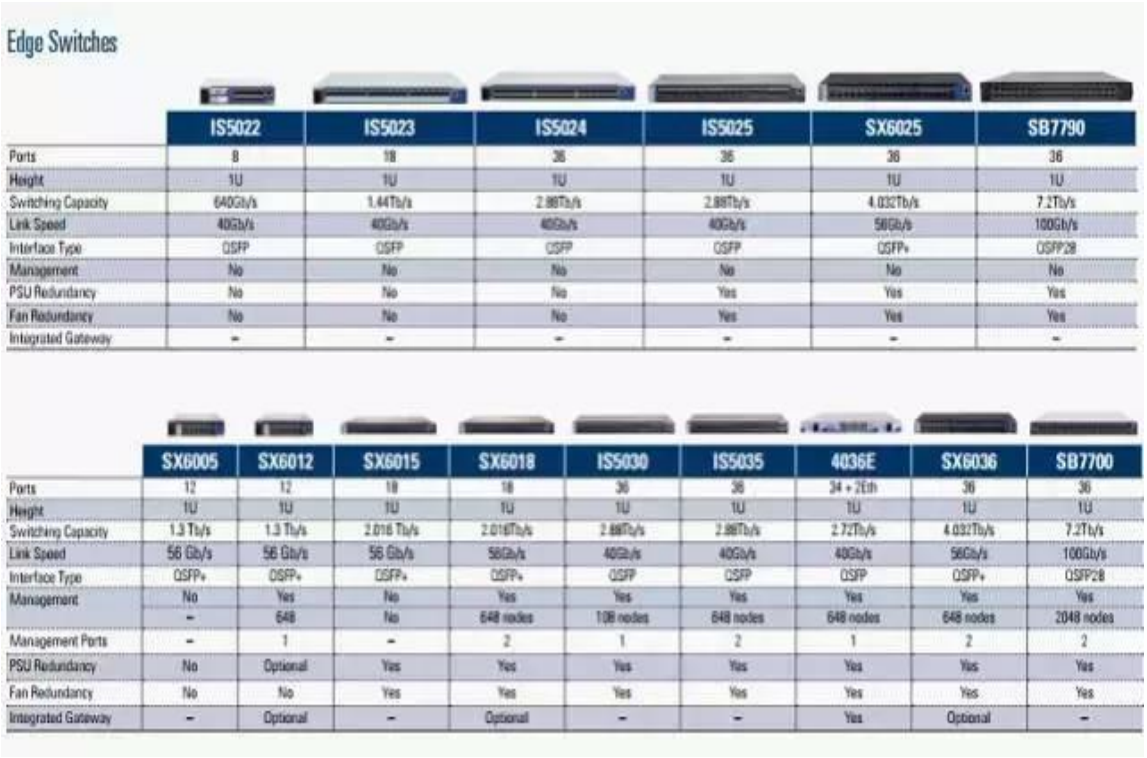


4.2 Infiniband 交换机

在 IB 网络内提供点到点高速通信;基于 LID 技术将数据从一个端口送到另外一个端口 ，当前单个交换机支持从 18 到 864 节点等规模不等，支持 SDR(10Gbps)、DDR(20Gbps)、QDR(40Gbps)、FDR10（40Gbps)、FDR(56Gbps)等。

从 SwitchX 到 Switch IB，SwitchX 是支持 10、20、40、56 G IB 主流的芯片，下一代芯片 Switch IB 支持 IB EDR 100Gb/s,并且向前兼容，后面还有 SwitchX3 支持 100G 和 IB EDR。

基于 ConnectX 系列网卡和 SwitchX 交换机可以实现以太网和 IB 网络的虚拟协议互联(VPI)，实现链路协议显示或自动适配，一个物理交换机实现多种技术支持。虚拟协议互联支持整机 VPI、端口 VPI 和 VPI 桥接，整机 VPI 实现交换机所有端口运行在 InfiniBand 或以太网模式，端口 VPI 实现交换机部分端口运行 InfiniBand、部分端口运行以太网模式，VPI 桥接模式实现 InfiniBand 和以太网桥接。



Edge Switches

	IS5022	IS5023	IS5024	IS5025	SX6025	SB7790
Ports	8	18	36	36	36	36
Height	1U	1U	1U	1U	1U	1U
Switching Capacity	640Gb/s	1.44Tb/s	2.88Tb/s	2.88Tb/s	4.032Tb/s	7.2Tb/s
Link Speed	40Gb/s	40Gb/s	40Gb/s	40Gb/s	56Gb/s	100Gb/s
Interface Type	QSFP	QSFP	QSFP	QSFP	QSFP+	QSFP28
Management	No	No	No	No	No	No
PSU Redundancy	No	No	No	Yes	Yes	Yes
Fan Redundancy	No	No	No	Yes	Yes	Yes
Integrated Gateway	-	-	-	-	-	-

	SX6005	SX6012	SX6015	SX6018	IS5030	IS5035	4036E	SX6036	SB7700
Ports	12	12	18	18	36	36	34 + 2Eth	36	36
Height	1U	1U	1U	1U	1U	1U	1U	1U	1U
Switching Capacity	1.3 Tb/s	1.3 Tb/s	2.016 Tb/s	2.016Tb/s	2.88Tb/s	2.88Tb/s	2.72Tb/s	4.032Tb/s	7.2Tb/s
Link Speed	56 Gb/s	56 Gb/s	56 Gb/s	56Gb/s	40Gb/s	40Gb/s	40Gb/s	56Gb/s	100Gb/s
Interface Type	QSFP+	QSFP+	QSFP+	QSFP+	QSFP	QSFP	QSFP	QSFP+	QSFP28
Management	No	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes
	-	648	No	648 nodes	108 nodes	648 nodes	648 nodes	648 nodes	2048 nodes
Management Ports	-	1	-	2	1	2	1	2	2
PSU Redundancy	No	Optional	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Fan Redundancy	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Integrated Gateway	-	Optional	-	Optional	-	-	Yes	Optional	-

边缘(机架) InfiniBand 交换机系统支持 8 到 36 端口，提供非阻塞 40 到 100Gb 端口，在 1U 的空间可提供 7.2Tb 的带宽，这些边缘交换机是组成中小型费阻塞网络集群 Leaf 节点的理想选择。边缘交换机使用先进的 InfiniBand 交换技术(如自适应路由、拥塞控制和服务质量等)旨在构建最有效的交换矩阵。

Director Switches

	SX6506	SX6512	CS7520	SX6518	SX6536	CS7500
Ports	108	216	216	324	648	648
Height	6U	9U	12U	16U	29U	28U
Switching Capacity	12.12Tb/s	24.24Tb/s	43.2Tb/s	36.36Tb/s	72.52Tb/s	130Tb/s
Link Speed	56Gb/s	56Gb/s	100Gb/s	56Gb/s	56Gb/s	100Gb/s
Interface Type	QSFP+	QSFP+	QSFP28	QSFP+	QSFP+	QSFP28
Management	648 nodes	648 nodes	2048 nodes	648 nodes	648 nodes	2048 nodes
Management HA	Yes	Yes	Yes	Yes	Yes	Yes
Console Cables	Yes	Yes	Yes	Yes	Yes	Yes
Spine Modules	3	6	6	9	18	18
Leaf Modules (Max)	6	12	6	18	36	18
PSU Redundancy	YES (N+N)	YES (N+N)	YES (N+N)	YES (N+N)	YES (N+N)	YES (N+N)
Fan Redundancy	Yes	Yes	Yes	Yes	Yes	Yes

核心 InfiniBand 交换机系统支持 108 至 648 端口，提供全双向 40 至 100Gb 端口， InfiniBand 核心交换机系统提供高密的解决方案，在一个机框内带宽可以 8.4Tb 至 130Tb 之间灵活扩展，可达数千个端口。针对关键任务应用，InfiniBand 核心交换机提供核心级可用性，系统所有部件都采用冗余技术设计。

4.3 InfiniBand 适配卡 HCA

Inifiniband 的主机信道适配器 HCA(网络接口卡)，通常通过 PCIE 接口与主机连接，插在或集成在服务器内;支持 PCI-E 8X 插槽(双端口和单端口)。提供 Inifiniband 的网络链路接入能力。等同于以太网的 NIC。HCA 包含三代芯片：目前主流的 QDR，FDR 使用的芯片为 ConnectX3，OSCA 使用的也是 ConnectX3

Channel Adapter(CA)
分为Host Channel Adapter(HCA)和Target Channel Adapter(TCA)

Host Channel Adapter(HCA)
主机通道适配器

如：**Mellanox 产品**

Target Channel Adapter(TCA)
目标通道适配器
用于IB交换机、存储系统的IO接口

型 号	速 率	系统说明	接 口
Mellanox MH 系列	10-40Gbps	PCIe 2.0	2个 IB 铜口 / QSHP

目标信道适配器(TCA)提供 InfiniBand 到 I/O 设备的连接，绑定在存储或网关设备等外设。

4.4 Infiniband 路由器和网关设备

Infiniband 路由器完成不同子网的 infiniband 报文的转发。Mellanox 的 SB7780 是基于 Switch-IB 交换机 ASIC 实现的 InfiniBand 路由器，提供 EDR 100Gb s 端口可以连接不同类型的拓扑。因此，它能够使每个子网拓扑最大化每个应用程序的性能。例如，存储子网可以使用 Fat Tree 拓扑，而计算子网可以使用最适合本地应用程序的环路拓扑。

SX6036G 是采用 Mellanox 第六代 SwitchX 2 InfiniBand 构建的交换机网关设备，提供高性能、低延迟的 56Gb FDR Infiniband 到 40Gb 以太网的网关，支持 InfiniBand 和以太网连接的虚拟协议互连(VPI)技术，VPI 通过一个硬件平台能够在同一机箱上运行 InfiniBand 和以太网网络协议。

4.5 Infiniband 线缆和收发器

Mellanox LinkX 互连产品包括 10、25、40、50 和 100 Gb/s 丰富铜缆、有源光缆以及针对单模光纤和多模光纤应用的收发器。

LinkX 系列提供 200Gb/s 和 400Gb/s 电缆和收发器等关键组件，对于 InfiniBand 互连基础设施来说，让端到端的 200Gb/s 解决方案成为可能。