December 2007

Mellanox
TECHNOLOGIES

InfiniBand Software and Protocols Enable Seamless Off-the-shelf Applications Deployment

1.0 Introduction

InfiniBand architecture defines a high-bandwidth, low-latency clustering interconnect that is used for high-performance computing (HPC) and enterprise data center (EDC) class applications. It is an industry standard developed by the InfiniBand Trade Association (www.InfiniBandTA.org).

Since the release of the specification, the InfiniBand community has been active in developing software for all major OS platforms including Linux open source software for the InfiniBand architecture. The Linux software community, comprising of major suppliers to the HPC and EDC markets collaborate on joint open source software development as part of the OpenFabrics alliance (www.OpenFabrics.org, previously called OpenIB.org).

The InfiniBand software stack is designed ground up to enable ease of application deployment. IP and TCP socket applications can avail of InfiniBand performance without requiring any change to existing applications that run over Ethernet. The same applies to SCSI, iSCSI and file system applications. Upper layer protocols that reside over the low level InfiniBand adapter device driver and device independent API (also called verbs) provide industry standard interfaces to enable seamless deployments of off-the-shelf applications.

Some extremely high performance applications require very low latency and such applications typically use industry standard message passing interface (MPI) or interface directly with the verbs layer. Such applications need to be ported and tuned specifically to the InfiniBand specific semantics. These applications and protocols are not discussed in this article.

1.1 Software Architecture

Figure 1 below shows the Linux InfiniBand software architecture. The software consists of a set of kernel modules and protocols. There are associated user-mode shared libraries as well which are not shown in the figure. Applications that operate at the user level stay transparent to the underlying interconnect technology. The focus of this article is to discuss what application developers need to know to enable their IP, SCSI, iSCSI, sockets or file system based applications to operate over InfiniBand.

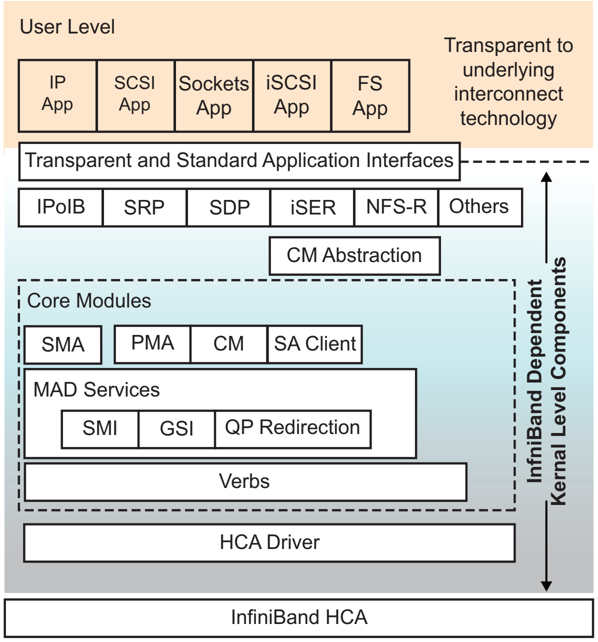

Figure 1

A detailed discussion on the operation of the protocols and the underlying Core and HCA Driver modules is beyond the scope of this article. However, for the sake of completeness, below is a brief overview of the kernel level and InfiniBand specific modules and protocols is presented below.

The kernel code divides logically into three layers: the HCA driver(s), the core InfiniBand modules, and the upper level protocols. The core InfiniBand modules comprise the kernel level mid-layer for InfiniBand devices. The mid-layer allows access to multiple HCA NICs and provides a common set of shared services. These include the following services:

- User-level Access Modules – The user-level access modules implement the necessary mechanisms to allow access to InfiniBand hardware from user-mode applications.
- The mid-layer provides the following functions:
  - Communications Manager (CM) – The CM provides the services needed to allow clients to establish connections.

- SA Client – The SA (Subnet Administrator) client provides functions that allow clients to communicate with the subnet administrator. The SA contains important information, such as path records, that are needed to establish connections.
- SMA – The Subnet Manager Agent responds to subnet management packets that allow the subnet manager to query and configure the devices on each host.
- PMA – The Performance Management Agent responds to management packets that allow retrieval of the hardware performance counters.
- MAD services – Management Datagram (MAD) services provide a set of interfaces that allow clients to access the special InfiniBand queue pairs (QP), 0 and 1.
- GSI – The General Services Interface (GSI) allows clients to send and receive management packets on special QP 1.
- Queue pair (QP) redirection allows an upper level management protocol that would normally share access to special QP 1 to redirect that traffic to a dedicated QP. This is done for upper level management protocols that are bandwidth intensive.
- SMI – The Subnet Management Interface (SMI) allows clients to send and receive packets on special QP 0. This is typically used by the subnet manager.
- Verbs – The mid-layer provides access to the InfiniBand verbs supplied by the HCA driver. The InfiniBand architecture specification defines the verbs. A verb is a semantic description of a function that must be provided. The mid-layer translates these semantic descriptions into a set of Linux kernel application programming interfaces (APIs).
- The mid-layer is also responsible for resource tracking, reference counting, and resource cleanup in the event of an abnormal program termination or in the event a client closes the interface without releasing all of the allocated resources.

- The lowest layer of the kernel-level InfiniBand stack consists of the HCA driver(s). Each HCA device requires an HCA-specific driver that registers with the mid-layer and provides the InfiniBand verbs.

The upper level protocols such as IPoIB, SRP, SDP, iSER etc., facilitate standard data networking, storage and file system applications to operate over InfiniBand. Except for IPoIB, which provides a simple encapsulation of TCP/IP data streams over InfiniBand, the other upper level protocols transparently enable higher bandwidth, lower latency, lower CPU utilization and end-to-end services using field proven RDMA (Remote DMA) and hardware based transport technologies available with InfiniBand. The following is a discussion of those upper level protocols and how existing applications can be quickly enabled to operate over them and InfiniBand.

1.2 Supporting IP Applications

The easiest path to evaluating any IP-based application over InfiniBand is to use the upper layer protocol called IP over IB (IPoIB). IPoIB running over high bandwidth InfiniBand adapters can provide an instant performance boost to any IP-based applications. IPoIB supports tunneling of Internet Protocol (IP) packets over InfiniBand hardware. See Figure 2 below. In Linux, the protocol is implemented as a standard Linux network driver, and this allows any application or kernel driver that uses standard Linux network services to use the InfiniBand transport without modification. Linux kernel 2.6.11 and above includes support of the IPoIB protocol, the InfiniBand Core and a HCA driver for HCA NICs based on Mellanox Technologies' HCA silicon.

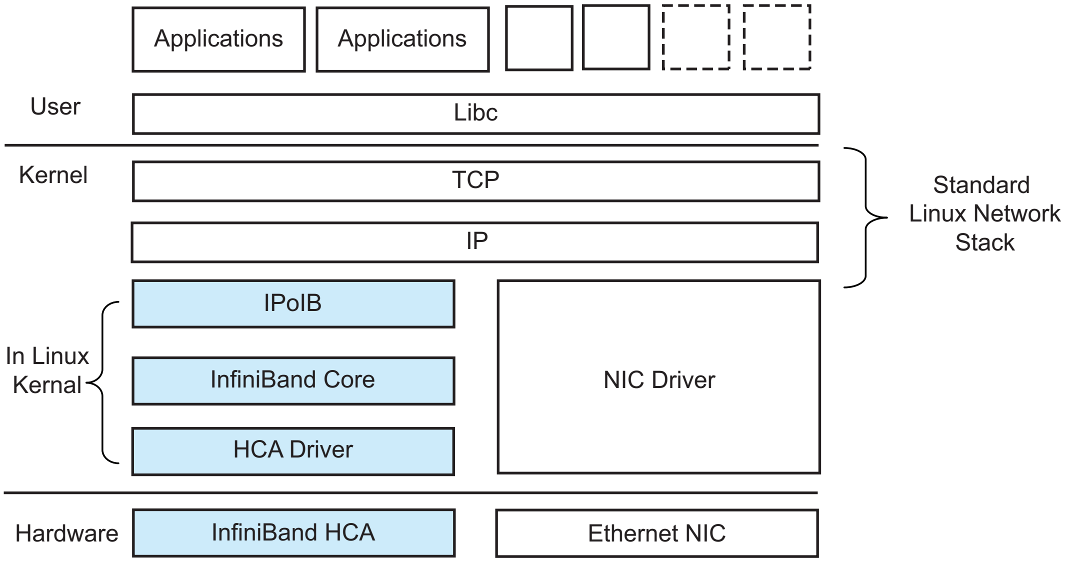

Figure 2

This method of enabling IP applications over InfiniBand is effective for management, configuration, setup or control plane related data where bandwidth and latency are not critical. Because the application continues to run over the standard TCP/IP networking stack, the applications are completely unaware of the underlying I/O hardware. However, to attain full performance and take advantage of some of the advanced features of the InfiniBand architecture, application developers may want to use the sockets direct protocol (SDP) and related sockets based API.

1.3 Supporting Sockets-based Applications

For applications that use TCP sockets, SDP or sockets direct protocol delivers a significant boost to performance while reducing CPU utilization and application latency. The SDP driver provides a high-performance interface for standard socket applications and provides a boost in performance by bypassing the software TCP/IP stack, implementing zero copy and asynchronous I/O, and transferring data using efficient RDMA and hardware based transport mechanisms.

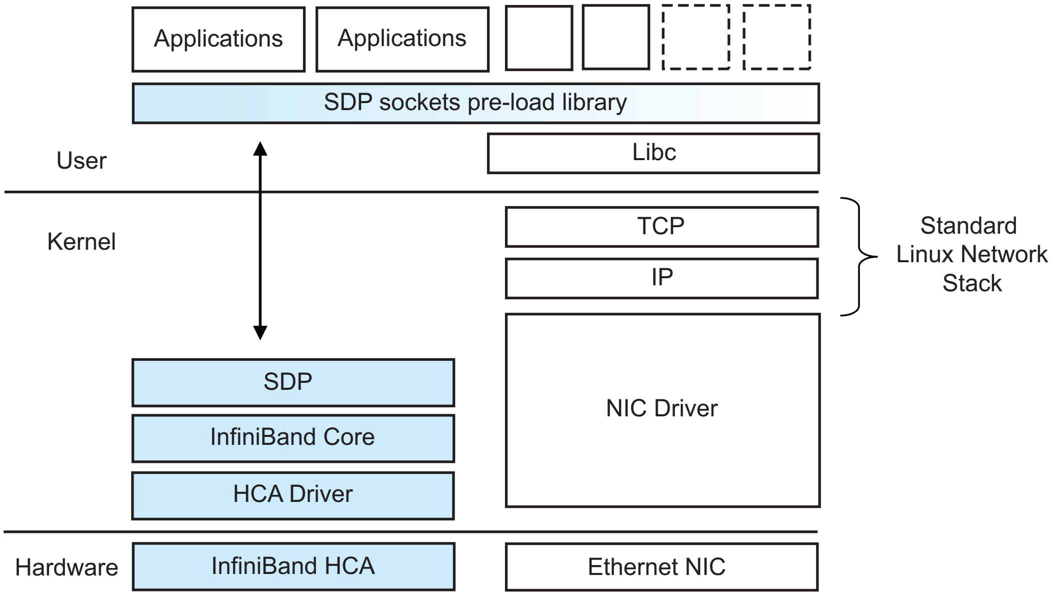

Figure 3

InfiniBand hardware provides a reliable and hardware based transport. As such, the TCP protocol becomes redundant and can be bypassed, saving valuable CPU cycles. See Figure 3 above which depicts a Linux based implementation of SDP. Zero-copy SDP implementations can save on expensive memory copies and use of RDMA can help save on expensive context switch penalties on, CPU utilization, performance and latency. The SDP protocol is implemented as a separate network address family. For example, TCP/IP provides the AF_INET address family and SDP provides the AF_SDP (27) address family. To allow standard sockets applications to use SDP without modification, SDP provides a preloaded library that traps the libc sockets calls destined for AF_INET and redirects them to AF_SDP. As is obvious, applications do not need to change except to interface with the preloaded library.

1.4 Supporting SCSI and iSCSI Protocol-based Applications

SCSI RDMA Protocol (SRP) was defined by the ANSI T10 committee to provide block storage capabilities for the InfiniBand architecture. SRP is a protocol that tunnels SCSI request packets over InfiniBand hardware using this industry-standard wire protocol. This allows one host driver to use storage target devices from various storage hardware vendors.

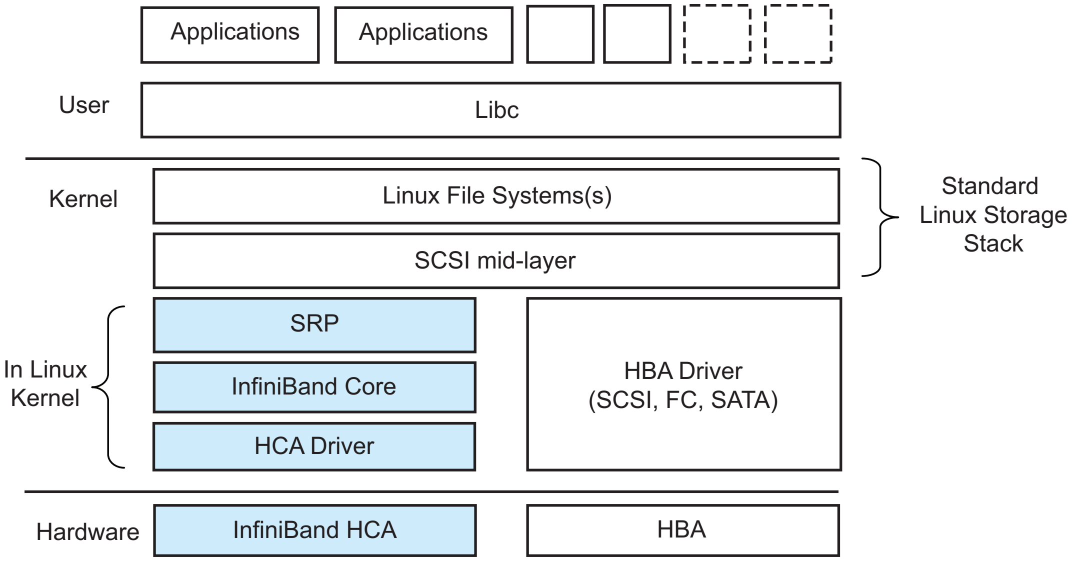

Figure 4

As shown in Figure 4 above, the SRP upper level protocol plugs into Linux using the SCSI mid-layer. Thus, to the upper layer Linux file systems and user applications that use those file systems, the SRP devices appear as any other locally attached storage device, even though they can be physically located anywhere on the fabric. It is worthwhile to mention that SRP is part of the latest Linux kernel versions.

iSER (iSCSI RDMA) eliminates the traditional iSCSI and TCP bottlenecks by enabling zero-copy RDMA, offloading CRC calculations in the transport layer to the hardware and by working with message boundaries instead of streams. It leverages iSCSI management and discovery facilities and uses SLP and iSNS global storage naming. The iSER specification for InfiniBand and other RDMA fabrics is driven within the Internet Engineering Task Force (IETF) (http://www.ietf.org/internet-drafts/draft-ietf-ips-iser-05.txt) and IBTA has created an annex for support of iSER over InfiniBand.

iSER follows the same methods of plugging into Linux using the SCSI mid-layer. However, as seen in Figure 5 below, iSER works over an extra layer of abstraction (the CMA, Connection Manager Abstraction layer) to enable transparent operation over InfiniBand and iWARP based RDMA technologies.

Applications that interface with LibC at the user level and the Linux File Systems at the kernel level work transparently, unaware of what interconnect technology being used underneath.

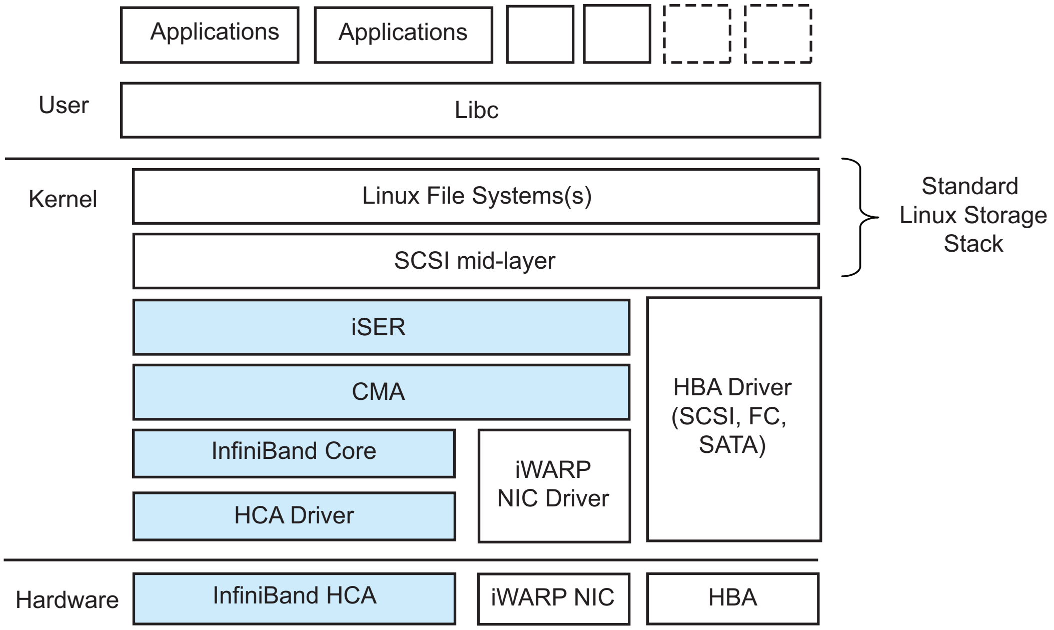

Figure 5

1.5 Supporting NFS-based Applications

Network File System (NFS) over RDMA is a protocol being developed by the Internet Engineering Task Force (IETF). This effort is extending NFS to take advantage of the RDMA features of the InfiniBand architecture and other RDMA enabled fabrics.

As shown in Figure 6 below, the NFS-RDMA client plugs into the RPC switch layer and the standard NFS v2/v3 layer in the Linux kernel. The RPC switch directs NFS traffic either through the NFS-RDMA client or the TCP/IP stack. Like the iSER implementation, NFS-RDMA client works over the CMA to provide transparent support over InfiniBand and iWARP based RDMA technologies. File system applications interface to the standard Linux file system layer and are unaware of the underlying interconnect.

An open source project is underway to develop an NFS-RDMA client at http://sourceforge.net/projects/nfs-rdma/. Mellanox provides a Linux NFS-RDMA package (see: http://www.mellanox.com/products/nfs_rdma_sdk.php) that is ready for production usage and is compliant with OpenFabrics InfiniBand software shipped in major Linux OS distributions.

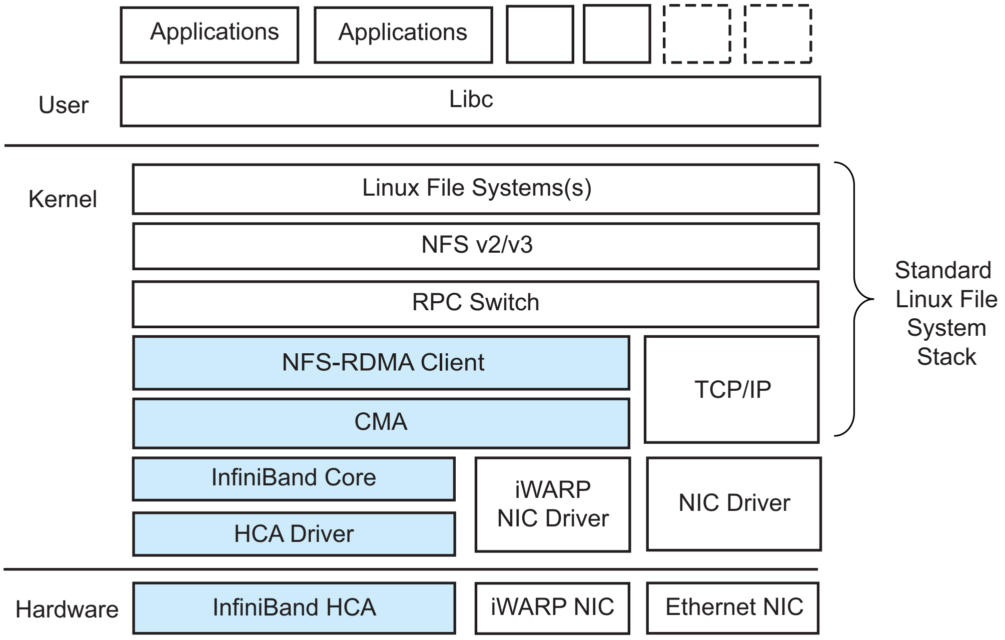

Figure 6

1.6 Software Ecosystem

InfiniBand software and protocols are supported and available in major Linux, Windows and Virtual Machine Monitor (Hypervisor) platforms. This includes Red Hat Enterprise Linux, SUSE Linux Enterprise Server, Microsoft Windows Server and Windows CCS (Compute Cluster Server), and VMware Virtual Infrastructure platforms.

1.7 Conclusion

InfiniBand offers advanced features that are attractive for high performance computing and data center applications that require high performance and no-compromise I/O services in the areas of throughput, latency and end-to-end service level guarantees. Using the protocols discussed in this article, applications that run today over lower performing interconnects can be transparently migrated, without any changes, to work over InfiniBand and avail of the advanced features. The InfiniBand RDMA based software ecosystem is the most advanced one in the industry today, allowing for multiple and reliable software sourcing options.