1 Evaluation

RDMA(remote direct memory access), is a new type of high-performance network technology. RDMA is currently widely used in artificial intelligence, data processing, and high-performance computing. For examples, TensorFlow, Spark, and Hadoop [21] all have supported RDMA. With hardware protocol stack and zero copy technologies, RNICs(physical RDMA network cards) can bypass the kernel to read/write remote memory data according to the work requests of applications, without the participation of remote CPU. Therefore, RDMA has high throughput, low latency and low CPU load.

The core technology of cloud computing is virtualization, mainly include container and virtual machine. The container is a lightweight isolated runtime environment, does not need device emulation, and has low performance loss. The virtual machine has strong isolation and is more secure, but the performance loss is large. The advantages of virtual machines and containers have made them widely used, and the trend has become the unified deployment and management for hybrid virtual environments. For example, VMware's virtualization platform vSphere[22] and ReadHat's container cloud platform OpenShift[23], both clearly support the unified deployment and management of virtual machines and containers in the latest version iteration.

RDMA virtualization is necessary for cloud applications to exploit RDMA. RDMA virtualization not only needs to maintain high performance and manageability, but also needs to have generality to adapt to the trend of unified management in hybrid virtual environments. Therefore, our RDMA virtualization goals are as follows:

- `Generality`: To form unified RDMA virtualization, single centralized virtual layer should be set up, which is provided to virtual machines and containers with general interfaces.
- `High performance`: Virtual RDMA should be close to native RDMA in terms of throughput, latency, and CPU load. Meanwhile it should also suit for large-scale virtual cluster.
- `High manageability`: In RDMA virtualization, container and virtual machine characteristics should be maintained to meet the needs of portability, isolation and network management.

RDMA has different hardware characteristics and working mechanisms. Therefore, RDMA virtualization is different from traditional network virtualization. Current RDMA virtualization work mainly includes hardware virtualization and software virtualization. However, none of the existing solutions can meet the above goals.

The representative of hardware virtualization is SR-IOV. Its virtual layer is located in the hardware. Although the isolation and high performance are maintained, SR-IOV lacks portability and other manageability without software virtual layer. In software virtualization, existing work treats virtual machines and containers differently. For containers, FreeFlow forwards all RDMA commands to the virtual layer, and that is ineffective unlike native RDMA's kernel by-pass. For virtual machines, HyV avoids forwarding data commands by mapping RDMA resources to achieve high performance, but lacks the management of RDMA networks; although MasQ makes up for this problem, its virtual layer is located in the kernel space. Migrating to the container environment will lose the lightweight characteristics of user space management orchestration.

We proposes a unified RDMA virtualization framework for containers and virtual machines, uniRDMA, which achieves high performance and high manageability without losing the advantages of container and virtual machine. UniRDMA is mainly composed of single centralized uniRDMA virtual layer and general uniVerbs interfaces. All virtual RDMA network managements are concentrated in the user space, which can flexibly and safely realize the unified management of RDMA of containers and virtual machines; the UniVerbs interface is universal to the RDMA applications of virtual machines and containers, and at the same time, the high level of RDMA is realized through resource mapping. performance.

In the design process of uniRDMA, there are mainly two challenges: First, the challenge of versatility; virtual machines and containers are essentially different virtualization technologies, but virtual machines are not ordinary processes. They interact with other processes on the host. It requires the participation of a virtual machine monitor, how the user space virtualization layer interacts with virtual machines and containers; second, high performance challenges; maintaining the high performance of RDMA under the premise of versatility and high manageability. The key to high performance lies in the mapping of RDMA resources, but at present, the existing RDMA virtualization work only implements the RDMA mapping operation in the virtual machine scenario. In this paper, the virtualization layer is in the user space, which is different from the virtual machine and the container and belongs to another host process. How to complete the mapping resource operation to achieve the goal of high-performance RDMA is a big challenge.

In order to solve the challenge of versatility, uniRDMA separates the construction and use of RDMA virtualization, builds a virtual vRNIC in a unified user space virtualization layer, encapsulates RDMA services, and isolates them

with the help of hardware virtualization. vRNIC is no different for virtual machines and containers. Further, through a common file-based memory sharing queue, it serves as a common interface between virtual machines and containers and vRNIC; in order to solve performance problems, this article uses shared memory and other mechanisms to RDMA resources are mapped. First, in the design of vRNIC, virtual RDMA resources are mapped to physical network cards, and then mechanisms such as shared memory are used to ensure that RDMA resources in RDMA applications are mapped to vRNICs in the virtual layer, thereby realizing data Zero copy and bypass the virtual layer.

We implements the prototype of the uniRDMA framework according to the design, and tests the operating effects of uniRDMA, hardware virtualization, software virtualization FreeFlow, and native RDMA from multiple dimensions such as throughput, latency, scalability, and real application effects. From the test data, uniRDMA achieves high performance close to native RDMA in both virtual machine and container scenarios. The overall performance is equivalent to hardware virtualization and is significantly better than FreeFlow. The throughput can reach up to 6 times that of FreeFlow. At the same time, the minimum delay is only 40% of FreeFlow; uniRDMA has high scalability in various scenarios and still maintains the same performance as native RDMA; uniRDMA adapts to real RDMA applications in various scenarios, among which uniRDMA The performance difference between the performance and hardware virtualization technology is less than 10%.

The main contributions of this article are: (1)Based on the RDMA virtualization goal, a universal RDMA virtualization framework uniRDMA was designed, while maintaining high performance and high manageability, and achieving a prototype of the framework. (2)Tested the performance effects of the uniRDMA framework in network benchmark performance, scalability, and real RDMA application scenarios. It shows that uniRDMA maintains network performance close to native RDMA while satisfying generality and high manageability.