

**本科毕业设计开题报告**

**RDMA性能在多租户环境下的测量**

**研究 - 虚拟机及容器**

**学生姓名: 朱庄翟**

**学生学号: 1100720119**

**指导教师: 张 琨**

**专 业: 计算机科学与技术**

**所在学院: 教育实验学院**

目 录

本科毕业设计开题报告 1

1. 项目背景 1

1.1. 高速网络与RDMA技术 1

1.1.1. RDMA的实现 2

1.1.2. 优势 2

1.2. 云计算与虚拟机 2

1.2.1. 云计算的优越性 2

1.2.2. 虚拟机 3

1.3. Linux容器 3

2. 目标和任务 4

2.1. TCP与RDMA 的网络性能实验 4

2.1.1. 主机间性能实验 5

2.1.2. 虚拟机间性能实验 5

2.1.3. Linux容器间性能实验 5

2.1.4. 多租户环境下的网络性能实验 6

2.2. 结果分析 6

2.2.1. TCP与RDMA的性能比较 6

2.2.2. 主机、虚拟机以及Linux容器的性能比较 7

2.2.3. 虚拟环境下多租户性能研究 7

3. 实验方法 7

3.1. 硬件需求 7

3.2. 软件配置 8

3.3. 实验方法 8

3.4. 并行编程 8

4. 关键技术考虑 9

4.1. 实验环境的搭建 9

4.1.1. Linux 的安装与配置 9

4.1.2. 虚拟机的安装与配置 9

4.1.3. Linux容器的安装与配置 9

4.1.4. RDMA的安装与设置 9

4.1.5. MPI通信库的安装与环境配置 10

4.2. 关键技术 10

4.2.1. RDMA的软件依赖关系 10

4.2.2. 多进程通信程序的开发 10

4.2.3. 网络性能测量方法 10

5. 预期工作结果 11

6. 进度计划 11

7. 参考文献 11

# 本科毕业设计开题报告

RDMA性能在多租户环境下的测量研究 - 虚拟机及容器

1. 项目背景

当代社会已经进入信息爆炸时代，数据产生的速率每天都在增长。据预计，到2020年，互联网上的数据将以每年4300%的速率增长。如何高效传输数据，减小通信延迟一直是计算机领域研究的重要课题。同时，云计算和虚拟技术在互联网的大潮中逐渐成为了影响信息产业格局的重要技术力量，越来越多的企业和个人开始使用虚拟机代替本地的服务器，使计算系统的使用、维护、扩展和升级成本大幅度降低，且更加方便。如何在虚拟技术系统中降低网络通信开销，特别是利用先进的高速网络技术加速云计算服务，在当今的信息产业发展中具有十分重要的研究价值和现实意义。

* 1. 高速网络与RDMA技术

通信延迟主要分为两个部分：处理消息的延迟和网络延迟。现代高速网络已经达到很高的传输速度，如Gigabit Ethernet，10G Ethernet 以及Myrinet和InfiniBand等高性能计算机通信网络，这样通信中的瓶颈就从以前的网络转移到处理消息收发任务的软件上。在传统的系统中，由内核处理消息的情况导致多次的数据拷贝和许多内容的转换，这样大大增大了端到端的延迟。在传统的基于处理器到处理器的互联中，主机端对于数据的复制操作是低效的，所有的数据都需要在用户缓冲区与内核缓冲区间传输，这样的数据复制操作需要极大的内存带宽。数据到达网络接口时，它更新内核缓冲区，然后由主机处理器将数据从内核缓冲区移动到用户缓冲区，在这过程中，需要进行大量的读写操作，反向亦然。这种方法降低了 CPU 的效率，也消耗了大量的内存带宽。

RDMA (Remote Direct Memory Access) 技术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](http://baike.baidu.com/view/3314.htm" \t "_blank)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](http://baike.baidu.com/view/3314.htm" \t "_blank)的处理功能。它消除了外部存储器复制和文本交换操作，因而能解放内存带宽和CPU周期用于改进应用系统性能。

* + 1. RDMA的实现

RDMA目前主要有三种协议实现，包括InfiniBand，RoCE和iWARP。

* InifiniBand

InifiniBand 是一种用于进程与I/O设备数据传输的通信链路,它提供2.5GB/s的吞吐量，可供64000个可寻址设备使用。由于它支持 质量服务（QoS)和故障转移(failover), InfiniBand经常用于高性能计算机(HPC)环境中的服务器连接。

许多PC和服务器系统的内部数据传输都相对较慢，随着系统内部流入和输的数据逐渐增加，现有的总线系统成为高速传输的瓶颈。相比于通过底盘总线并行传送数据，InfiniBand 使用的是串行总线, 串行总线能够在一个多路复用信号中同时承载多种数据信息。同时，InfiniBand还支持多重内存区域，每个内存区域都可通过进程和存储设备寻址。

* RoCE

RoCE(RDMA over Converged Ethernet) 是用于以太网上的远程直接内存访问的网络协议。目前RoCE有两种版本: RoCE v1和RoCE v2。RoCE是链路层的协议，因此它允许在同一以太网广播域中的任意两个主机通信。RoCE v2是网络层的协议，因此它的网络层数据包是可路由的，该协议可同时用于传统的非聚合以太网和聚合以太网。

* iWARP

iWARP允许在TCP层上实现RDMA。一些InfiniBand和RoCE中的特性在iWARP中并不支持。iWARP允许在标准以太网设施上（交换机）实现RDMA，但需要特定的网卡（NIC）来支持。

* + 1. 优势

RDMA具有以下优点：

* 零拷贝(Zero-copy)，
* 绕过系统内核(Kernel bypass)
* 无需对方CPU参与(No CPU involvement)
* 基于消息的传输(Message based transactions)
* 支持数据发散/聚集 (Scatter/gather entries)

RDMA 技术基于远程内存直接访问，可以显著降低数据处理延迟从而改进网络系统性能。但如何在虚拟化环境中利用RDMA技术是一项研究课题。​

* 1. 云计算与虚拟机

云计算（Cloud Computing），是一种基于互联网的计算方式，通过这种方式，共享的软硬件资源和信息可以按需提供给计算机和其他设备。云计算拥有复杂的后台控制，却可以留给用户简单、明了的应用接口，用户不需要了解云计算背后的技术细节，不需要知道云计算服务的地理位置，无需计算机相关的知识，也不必直接控制云计算资源。对于用户来说，云资源是趋于无限大的，当然这得益于云计算服务采用的虚拟化技术。

* + 1. 云计算的优越性

云计算具有常规计算方式所不具备的优势。总体来讲，云计算具有地理位置无关性、设备无关性、时效性、可靠性等特点，这使得它迅速建立来了一个全新的市场领域，新的服务模式。

位置无关性主要表现在只要有网络的地方就可以计算。云计算本身是一种基于互联网的服务，在有网络访问的地放就可以访问相应的服务。设备无关性在于所有的设备上的应用都是基于同一种标准开发，无论客户端设备是什么，均能得到同等的服务。时效性一方面在于所有的服务都已经现有，而且随时可用；另一方面也在于云计算本身强大的计算能力，其性能瓶颈基本上在网络传输上而不是计算本身。云计算一般都会在一定范围内集群大量计算机，任何一部分的计算机故障都不会导致整个系统的崩溃，可靠性比小型计算机系统好的多。

* + 1. 虚拟机

虚拟机的基本原理

虚拟机的几种目前流行的实现

* VirtualBox

VirtualBox 是一款开源[虚拟机软件](http://baike.baidu.com/view/2690045.htm" \t "_blank)。VirtualBox 是由德国 Innotek 公司开发，由Sun Microsystems公司Sun Microsystems公司出品的软件，使用[Qt](http://baike.baidu.com/view/23681.htm" \t "_blank)编写，在 Sun 被 [Oracle](http://baike.baidu.com/view/15020.htm" \t "_blank) 收购后正式更名成 Oracle VM VirtualBox。Innotek 以 GNU General Public License (GPL) 释出 VirtualBox，并提供二进制版本及 OSE 版本的代码。使用者可以在VirtualBox上安装并且执行[Solaris](http://baike.baidu.com/view/329359.htm" \t "_blank)、[Windows](http://baike.baidu.com/view/4821.htm" \t "_blank)、[DOS](http://baike.baidu.com/subview/365/7971327.htm" \t "_blank)、[Linux](http://baike.baidu.com/view/1634.htm" \t "_blank)、OS/2 Warp、[BSD](http://baike.baidu.com/view/209692.htm" \t "_blank)等系统作为客户端操作系统。它的主要特点如下：

- 支持64位客户端操作系统，即使主机使用32位CPU  
　　- 支持SATA硬盘NCQ技术  
　　- 虚拟硬盘快照  
　　- 无缝视窗模式（须安装客户端驱动）  
　　- 能够在主机端与客户端共享剪贴簿（须安装客户端驱动）  
　　- 在主机端与客户端间建立分享文件夹（须安装客户端驱动）  
　　- 内建远端桌面服务器，实现单机多用户　- 支持VMware VMDK磁盘档及Virtual PC VHD磁盘档格式  
　　- 3D虚拟化技术支持OpenGL（2.1版后支持）、Direct3D（3.0版后支持）、WDDM（4.1版后支持）  
　　- 最多虚拟32颗CPU（3.0版后支持）

* VMware

VMware（中文名[威睿](http://baike.baidu.com/view/2315382.htm" \t "_blank) ） [虚拟机](http://baike.baidu.com/view/1132.htm" \t "_blank)软件可以使用户在一台机器上同时运行二个或更多[Windows](http://baike.baidu.com/view/4821.htm" \t "_blank)、[DOS](http://baike.baidu.com/view/365.htm" \t "_blank)、[LINUX](http://baike.baidu.com/view/1634.htm" \t "_blank)系统。与“多启动”系统相比，VMWare采用了完全不同的概念。多启动系统在一个时刻只能运行一个系统，在系统切换时需要重新启动机器。VMWare是真正“同时”运行，多个[操作系统](http://baike.baidu.com/view/880.htm" \t "_blank)在主系统的平台上，就象标准Windows[应用程序](http://baike.baidu.com/view/330120.htm" \t "_blank)那样切换。而且每个[操作系统](http://baike.baidu.com/view/880.htm" \t "_blank)都可以进行虚拟的分区、配置而不影响真实硬盘的数据，甚至可以通过网卡将几台[虚拟机](http://baike.baidu.com/view/1132.htm" \t "_blank)用网卡连接为一个局域网，极其方便。安装在VMware[操作系统](http://baike.baidu.com/view/880.htm" \t "_blank)性能上比直接安装在硬盘上的系统低不少。

* OpenStack

OpenStack是一个由NASA（[美国国家航空航天局](http://baike.baidu.com/view/85211.htm" \t "_blank)）和Rackspace[合作研发](http://baike.baidu.com/view/1506447.htm" \t "_blank)并发起的，以[Apache](http://baike.baidu.com/view/28283.htm" \t "_blank)许可证授权的[自由软件](http://baike.baidu.com/view/20965.htm" \t "_blank)和[开放源代码](http://baike.baidu.com/view/1708.htm" \t "_blank)项目。它是一个旨在为公共及私有云的建设与管理提供软件的开源项目。它覆盖了网络、虚拟化、操作系统、服务器等各个方面。

OpenStack云计算平台，帮助服务商和企业内部实现类似于 Amazon EC2 和 S3 的云基础架构服务(Infrastructure as a Service, IaaS)。OpenStack 包含两个主要模块：Nova 和 Swift，前者是 NASA 开发的虚拟服务器部署和业务计算模块；后者是 Rackspace开发的分布式云存储模块，两者可以一起用，也可以分开单独用。

* KVM

KVM：就是Keyboard Video Mouse的缩写。KVM [交换机](http://baike.baidu.com/view/1077.htm" \t "_blank)通过直接连接键盘、视频和鼠标 (KVM) 端口，让用户能够访问和控制计算机。KVM 技术无需目标服务器修改软件。这就意味着可以在 Windows 的 BIOS 环境下，随时访问目标计算机。KVM 提供真正的主板级别访问，并支持多平台服务器和串行设备。

* 1. Linux容器

Linux 容器 (LXC) 是一种类似虚拟机，但更轻量级的虚拟实现技术，LXC为Linux Container的简写。Linux Container容器是一种内核[虚拟化技术](http://baike.baidu.com/view/13605.htm" \t "_blank)，可以提供轻量级的虚拟化，以便隔离进程和资源，而且不需要提供指令解释机制以及全虚拟化的其他复杂性。相当于C++中的NameSpace。容器有效地将由单个操作系统管理的资源划分到孤立的组中，以更好地在孤立的组之间平衡有冲突的资源使用需求。与传统[虚拟化技术](http://baike.baidu.com/view/13605.htm" \t "_blank)相比，它的优势在于：

（1）与宿[主机](http://baike.baidu.com/view/23880.htm" \t "_blank)使用同一个内核，性能损耗小；

（2）不需要指令级模拟；

（3）不需要即时(Just-in-time)编译；

（4）容器可以在CPU核心的本地[运行指令](http://baike.baidu.com/view/2290127.htm" \t "_blank)，不需要任何专门的解释机制；

（5）避免了准虚拟化和系统调用替换中的复杂性；

（6）轻量级隔离，在隔离的同时还提供共享机制，以实现容器与宿主机的资源共享。

总结：Linux Container是一种轻量级的虚拟化的手段。

Linux Container提供了在单一可控[主机](http://baike.baidu.com/view/23880.htm" \t "_blank)节点上支持多个相互隔离的server container同时执行的机制。Linux Container有点像chroot，提供了一个拥有自己进程和[网络空间](http://baike.baidu.com/view/420399.htm" \t "_blank)的虚拟环境，但又有别于[虚拟机](http://baike.baidu.com/view/1132.htm" \t "_blank)，因为lxc是一种操作系统层次上的资源的虚拟化。截止2014年底LXC的最新版本为1.0。

1. 目标和任务

随着计算机网络技术和虚拟化技术的发展，在虚拟机上利用高速网络技术提高网络传输性能是十分重要和有意义的研究课题。本次毕业设计旨在了解并掌握现代高速网络技术，特别是RDMA技术，降低数据处理延迟，改进网络系统性能。本毕业设计着重研究虚拟化技术对RDMA技术的影响，通过对比在虚拟化多租户环境中，标准TCP协议和RDMA技术，研究当前主流RDMA技术在虚拟机上的性能。

* 1. TCP与RDMA 的网络性能实验

本次毕业设计的主要内容是就是测量不同环境下的TCP和RDMA的网络通信性能，对比相同大小的数据传输在主机和不同虚拟技术下，使用TCP和RDMA进行通信的性能差异。

网络性能实验的基本技术要求包括熟练掌握Linux操作系统，网络配置，MPI多进程编程模型等。为了系统地准确地进行测试实验并方便之后的数据分析对比，我们的性能实验设计总共有10项四组，显示在Table 1。下面我们分别描述各组实验的基本设计思路。

**TABLE 1. TCP和RDMA网络性能实验设计**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test** | **Sender** | **Receiver** | **Protocol** | **Transfer** |
| 1 | host | host | TCP | 1 and 4 transfers |
| 2 | host | host | RDMA | 1 and 4 transfers |
| 3 | VM | VM | TCP | 1 and 4 transfers |
| 4 | VM | VM | RDMA | 1 and 4 transfers |
| 5 | LXC | LXC | TCP | 1 and 4 transfers |
| 6 | LXC | LXC | RDMA | 1 and 4 transfers |
| 7 | 4 VMs | 4 VMs | TCP | 1 transfer |
| 8 | 4 VMs | 4 VMs | RDMA | 1 transfer |
| 9 | 4 LXCs | 4 LXCs | TCP | 1 transfer |
| 10 | 4 LXCs | 4 LXCs | RDMA | 1 transfer |

* + 1. 主机间性能实验

主机间实验就是测量两台通过网络直连(或通过一个交换机相连)的Linux系统计算机相互文件传输的网络延迟和带宽。这组实验是其他实验的基础，通过这组实验建立起来基本的网络通信性能数据，用来和使用虚拟技术进行网络通信的实验数据进行对比。第一组实验包括以下两项四个测试。

1. 建立两个Linux系统的主机，(a) 每个主机上各使用一个进程，让其用TCP 协议相互传送1GB 数据；(b) 每个主机上各使用四个进程，让其用TCP 协议两两相互传送1GB 数据。
2. 建立两个Linux系统的主机，(a) 每个主机上各使用一个进程，让其用RDMA 协议（ifiniBand/RoCE/ iWARP）相互发送1GB 文件；(b) 每个主机上各使用四个进程，让其用RDMA 协议（ifiniBand/RoCE/ iWARP） 两两相互传送1GB 数据。
   * 1. 虚拟机间性能实验

第二组实验是在两台主机上安装虚拟机软件，并建立虚拟机进行网络传输性能测试。第二组实验包括以下两项四个测试。

1. 在两台主机上各建立一个虚拟机，(a) 在每个虚拟机上各使用一个进程用TCP协议相互传送1GB 数据；(b) 在每个虚拟机上各使用四个进程，用TCP协议两两相互传送1GB数据。
2. 在两台主机上各建立一个虚拟机，(a) 在每个虚拟机上各使用一个进程用RDMA协议（ifiniBand/RoCE/ iWARP）相互传送1GB 数据；(b) 在每个虚拟机上各使用四个进程，用RDMA协议（ifiniBand/RoCE/ iWARP）两两相互传送1GB数据。
   * 1. Linux容器间性能实验

第三组实验使用Linux容器，在两台主机上安装Linux容器软件，并建立Linux容器来进行网络传输性能测试。第三组实验包括以下两项四个测试。

1. 在两台主机上各建立一个Linux容器，(a) 在每个Linux容器上各使用一个进程用TCP协议相互传送1GB 数据；(b) 在每个Linux容器上各使用四个进程，用TCP协议两两相互传送1GB数据。
2. 在两台主机上各建立一个Linux容器，(a) 在每个Linux容器上各使用一个进程用RDMA协议（ifiniBand/RoCE/ iWARP）相互传送1GB 数据；(b) 在每个Linux容器上各使用四个进程，用RDMA协议（ifiniBand/RoCE/ iWARP）两两相互传送1GB数据。
   * 1. 多租户环境下的网络性能实验

以上的三组实验都是在单租户环境下地网络测试，而虚拟技术的一大特点就是同一台物理主机上会同时运行多个虚拟环境，多个虚拟系统会共享网络资源，这种情形就被称为多租户(multi-tenant)环境。如下四项实验是测量多租户环境下TCP和RDMA网络传输性能。

1. 在两台主机上各建立四个虚拟机，在两台位于不同主机的虚拟机之间两两用TCP协议相互传送1GB 数据。
2. 在两台主机上各建立四个虚拟机，在两台位于不同主机的虚拟机之间两两用RDMA协议（ifiniBand/RoCE/ iWARP）相互传送1GB 数据。
3. 在两台主机上各建立四个Linux容器，在两台位于不同主机的Linux容器之间两两用TCP协议相互传送1GB 数据。
4. 在两台主机上各建立四个Linux容器，在两台位于不同主机的Linux容器之间两两用RDMA协议（ifiniBand/RoCE/ iWARP）相互传送1GB 数据。
   1. 结果分析

如果能够顺利完成上述实验，得到可靠的结果数据，我们将针对数据就行分析，以期待得到一些TCP和RDMA在主机和不同虚拟环境下，以及多租户环境下的结论。并通过对比初步获得进一步研究和优化的建议。预计的分析将从以下三方面展开。

* + 1. TCP与RDMA的性能比较

首先我们要对传统TCP协议和RDMA协议性能进行比较。可预期的时RDMA将会提供更小的延迟和更大得带宽。本项分析将着重在在同一配置（主机、虚拟机和Linux容器，及多租户环境）内的性能呢比较：

* 实验1和实验2对比：主机间TCP和RDMA网络性能比较
* 实验3和实验4对比：虚拟机间TCP和RDMA网络性能比较
* 实验5和实验6对比：Linux容器间TCP和RDMA网络性能比较
* 实验7和实验8对比：虚拟机多租户环境下TCP和RDMA网络性能比较
* 实验9和实验10对比：Linux容器多租户环境下TCP和RDMA网络性能比较
  + 1. 主机、虚拟机以及Linux容器的性能比较

本项分析着重在同一协议下进行，比较不同虚拟技术对网络性能的影响。

* 实验1a，3a，5a的对比：TCP协议在主机、虚拟机和Linux容器中的网络性能比较
* 实验2a，4a，6a的对比：RDMA协议在主机、虚拟机和Linux容器中的网络性能比较
  + 1. 虚拟环境下多租户性能研究

本项分析在前两项实验基础上，分析多租户环境下网络共享在不同虚拟技术上的性能影响。

* 实验1b，3b，5b，7，9的对比： TCP协议协议在主机多进程共享网络、单一虚拟机多进程共享网络、单一Linux容器多进程共享网络、多个虚拟机共享网络、多个Linux容器共享网络中的网络性能比较
* 实验2b，4b，6b，8，10的对比：RDMA协议协议在主机多进程共享网络、单一虚拟机多进程共享网络、单一Linux容器多进程共享网络、多个虚拟机共享网络、多个Linux容器共享网络中的网络性能比较

1. 实验方法
   1. 硬件需求

为了完成设计的所有实验，需要至少两台服务器：

* CPU至少四核处理器
* 服务器至少8GB内存
* 支持Giga Ethernet，10G Ethernet或InfiniBand网络的网卡
* 网络：用于两台服务器连接的直连网线，或用于互连的网线和交换机
  1. 软件配置

为了完成设计的所有实验，需要以下相关软件：

* Linux操作系统，目前暂定为CentOS 6.6
* RDMA的网络支持: OFA OFED 或 Mellanox OFED
* RDMA相关网络驱动: RoCE, iWARP 或 InfiniBand
* RDMA的软件实现: SoftRoCE 或 SoftiWARP
* MPI通信库: 目前暂定为 OpenMPI-1.8.1
  1. 实验方法

预计的网络性能实验将通过自己设计开发MPI多进程并行程序进行通信来实现，预计的编译器为Linux标准的gcc。

因为网络通信的性能与硬件、软件系统的很多行为相关，为了保证实验的稳定性和可靠性，实验将采取以下方法：

* 每个程序将循环 （如20次）进行多次通信，最终计算平均时间；
* 为避免网卡等启动开销，每个程序将把前若干次通信（如5次）排除在计时之外；
* 同一组实验将重复10次以上，对10次实验结果取平均。
  1. 并行编程

由于实验设计要同时操纵多个进程进行通信，我们预计选取MPI编程模型进行并行编程来实现通信和计时实验。

MPI，全称Message Passing Interface，是并行计算和高性能分布式计算的事实工业标准，被广泛地应用在科学计算、大规模数据处理和超级计算机等领域。由于RDMA等高性能网络技术在超级计算机中广泛地应用，MPI主要的实现库都普遍支持RDMA的通信，并能自由选择切换TCP与高速通信网络。目前MPI的实现最主要的有开源实现MPICH和OpenMPI两种，以及众多的商业化实现版本（Intel，HP，IBM，Cray等）。在本毕业设计中，我计划采用OpenMPI作为通信库来进行实验。

1. 关键技术考虑

这部分简单介绍本毕业设计的主要技术难点和关键技术，毕业设计的顺利完成需要逐一解决以下技术问题并在实际系统中正确配置实现。

* 1. 实验环境的搭建

实验环境的搭建是本设计一个难点，因为配置能够实现在不同虚拟环境下使用不同网络协议通信的软硬件系统，其中的工作需要涉及Linux操作系统、网络协议、不同虚拟机实现、基础通信库版本、Linux容器以及多种RDMA实现的工作，涉及知识面广，工作量大，需要在实验过程中逐渐摸索并找到正确的设置和实验方法。

* + 1. Linux 的安装与配置

需要掌握Linux操作系统基本安装和配置。预计选用CentOS 6.6系统，并使用yum安装程序安装Linux基本开发所用的工具。由于涉及到在虚拟环境下重复安装操作系统并做同一性的配置，我们计划使用snapshot软件对安装好的Linux系统进行镜像，并将同一镜像应用到多台主机/虚拟机中。

* + 1. 虚拟机的安装与配置

虚拟机目前在Linux下主要的实现有KVM, VM Ware, Virtual Box和OpenStack等。将在毕业设计进行中结合相关文档进行实验和配置，安装选用最方便的虚拟机。由于多租户环境下的实验需要在同一主机上启动4台虚拟机，对主机系统的内存压力较大。且传输文件实验中，需要每台虚拟机都同时进行大数据传送，所以虚拟机的内存配置非常关键，需要在设计实现过程中逐渐研究解决。另外虚拟机软件对网络配置上需要认真研究虚拟网络环境的设置。

* + 1. Linux容器的安装与配置

Linux容器是相对轻量级的虚拟技术，目前有开源的LXC-1.0软件，需要对Linux容器在CentOS系统上的安装和配置进行实验。

* + 1. RDMA的安装与设置

RDMA的几种实现技术对网络的要求不同，比如InfiniBand网络一般要求有支持InfiniBand的网络适配器支持。如果能够得到支持相应技术的网络设备，实验将更加方便，否则软件实现，如SoftRoCE等将被考虑采用作为替代。

* + 1. MPI通信库的安装与环境配置

MPI通信库将采用开源的OpenMPI实现，一种方案是使用yum系统直接rpm安装，另一种是下载源码并通过configure, make编译生成，后者有利于灵活地系统配置，但需要更多的时间实验研究。其中MPI环境配置需要在多个主机或虚拟机之间建立ssh信任连接。

* 1. 关键技术
     1. RDMA的软件依赖关系

在前期的实验中发现，主要RDMA实现都需要依赖OFED库的实现。OFED是Open Fabric Enterprise Development的简称，该软件提供了网络通信自上而下啊所需的多层API（应用程序编程接口），。。目前主要的OFED实现有开源的OFA OFED和商业用的Mellanox OFED实现。每个OFED对Linux系统内核版本，相关软件的依赖都很敏感，这需要相当多的实验来验证OFED软件对操作系统内核的兼容性。此外，RDMA依赖的软件包众多，都需要通过逐一查找安装配置来达到实验所需的环境。

* + 1. 多进程通信程序的开发

MPI并行编程开发是本设计的一个关键技术，并行编程不同于串行编程，需要统一考虑多个进程的行为，在通信和同步过程中都需要调用MPI\_开头的通信函数来实现。目前首先考虑的时最基本的阻塞式MPI\_Send和MPI\_Recv的实现，如果有必要，将考虑非阻塞式通信以及单边通信等作为测量程序的实现。

* + 1. 网络性能测量方法

网络性能的主要评估指标包括网络延迟和网络带宽，很多标准测试程序(benchmark)能够用来测量，比如OSU benchmark, Intel micro benchmark等。本设计将参考并结合相关标准测试程序来测量多租户环境下虚拟技术中得网络性能。

1. 预期工作结果

当整个设计完成时，将有如下预期工作结果和贡献：

* TCP与RDMA在主机、虚拟机及Linux容器下网络性能结果及结论；
* 在多租户环境下，TCP与RDMA在主机、虚拟机及Linux容器下网络性能结果及结论；
* Linux虚拟机上RDMA的安装和配置流程；
* Linux 容器上RDMA的安装和配置流程；
* 多租户环境下RDMA性能优化的建议。

1. 进度计划

|  |  |
| --- | --- |
| 时间安排 | 任务 |
| 2014.12 | 学习基本Linux，RDMA等基础知识，提交开题报告 |
| 2015.01 | 主机上Linux系统实验平台的搭建和实验完成 |
| 2015.01~2015.02 | 虚拟机平台的搭建和实验完成 |
| 2015.02~2015.04 | LXC平台的搭建和实验完成 |
| 2015.05 | 实验全部完成，毕业论文基本完成 |
| 2015.06 | 毕业论文提交，毕业答辩 |

1. 参考文献
2. Hines, Michael. Features/RDMA Live Migration - QEMU
3. Koning, Paul Black, David. RFC7146 - Securing Block Storage Protocols over IP: RFC 3723 Requirements Update for IPsec v3. tools.ietf.org, April 2014
4. Xiaoyi Lu, Md. Wasi-ur-Rahman, Nusrat Islam, Dipti Shankar, Dhabaleswar K. (DK) Panda, Accelerating Spark with RDMA for Big Data Processing: Early Experiences. IEEE Computer Society, 2014
5. Metzler, Bernard Culley, Paul R Recio, Renato J Garcia, Dave Hilland, Jeff. RFC5040- A Remote Direct Memory Access Protocol Specification. tools.ietf.org
6. Talpey, Thomas Juszczak, Chet. RFC5532 - Network File System (NFS) Remote Direct Memory Access (RDMA) Problem Statemen. tools.ietf.org.
7. 沈利. Infiniband网络接口的研究与实现. [学位论文], 国防科学技术大学, 2010