

Resource Virtualization and the Enterprise

Syllabus

Web Page

<http://pdinda.org/virt-msit>

Instructor

Peter A. Dinda

Technological Institute, Room L463

847-467-7859

pdinda@northwestern.edu

Office hours: Thursdays, 2-4pm or by appointment

I am an associate professor of computer science at Northwestern University, as well as the head of the computer engineering and systems division of the EECS department. I hold a BS degree in ECE from U.Wisconsin and a Ph.D. degree in CS from Carnegie Mellon University. My general area of research is distributed and parallel systems. I have been involved in virtualization research since 2002, leading several NSF- and DOE-funded projects in the area. A 2003 paper of mine, *A Case for Grid Computing on Virtual Machines*, is an often-cited work on what is these days called “infrastructure-as-a-service cloud computing.” In my current virtualization project, my group has built an open source virtual machine monitor for modern x86 hardware to support virtualization in high performance computing.

Prerequisites

I recommend that students will have familiarity with the general principles of operating systems, although we will review these in class. Students should also have had some practical experience in installing operating systems and managing infrastructure. Students should have been exposed to the concepts of an instruction set architecture and a machine architecture.

Materials

Students will be provided with the following materials:

- Hess and Newman, *Practical Virtualization Solutions: Virtualization from the Trenches*, Prentice Hall, 2010. This is the “IT” book for the class. It is targeted at a working IT professional and it contains both low-level

detail on how to use virtualization tools such as VMware and Xen, and high-level material on designing and managing virtualized infrastructure.

- Smith and Nair, *Virtual Machines: Versatile Platforms for Systems and Processes*, Morgan Kaufman, 2005. This is a reference book on various aspects of virtualization from an architect's point of view. It contains details of how a real machine and a virtual machine actually work.
- Students will have access to the Wilkinson Lab, an educational lab with Linux computers that can be used to talk to the VLab.
- Students will also have access to the VLab, a cluster that supports VMware Server. <http://pdinda.org/vlab>.
- Students can download their own free copies of VMware, Xen, Virtual Box, and other virtualization software
- Students can examine the source code of Palacios, a virtual machine monitor that my group is building, by visiting <http://v3vee.org>

Lab Accounts

Students should already have accounts in the Wilkinson Lab, with their account names being their netids.

If you have never used your Wilkinson Lab account, you will need to reset your password, which you can do here:

https://selfserv.eecs.northwestern.edu/temp_password/

The list of Wilkinson Lab machines you can use is given here:

<http://www.eecs.northwestern.edu/support/techsupport/>

If you don't have physical access to the Wilkinson Lab, you can still access the machines using SSH. Northwestern has a site-licensed SSH client, available from here: <http://charlotte.at.northwestern.edu/bef/SSHdist.html>

The Wilkinson Lab machines have VMware Server Console installed, which you will need to use to interact with VMware Server running on the VLab.

The VLab is described here: <http://pdinda.org/vlab>. We will provide you with your account information for the VLab during the first lecture, and you will be able to use it for the duration of the class.

Related non-MITP Course

I teach a quarter-long graduate research seminar on resource virtualization, EECS 441, from which this course is partially derived. 441 currently focuses on the design and implementation of virtual machine monitors (e.g., kernel development), using our Palacios VMM codebase. For more information see <http://pdinda.org/virt>.

Objectives, framework, philosophy, and caveats

Indirection is a concept that's used throughout computer science to simplify problems and systems, and to add functionality. Currently, there is considerable excitement among IT practitioners, IT managers, and the systems research community over a form of indirection that we shall call *resource virtualization*. The basic idea is to add a software layer that provides virtual machines, virtual networks, virtual services, and even virtual infrastructures that are implemented on top of the existing physical resources and services in the network. Because these resources are virtual, we can potentially create a great many of them, make them private to their users, customize them to particular purposes, simplify their administration by making them user- or group-specific, and even inspect them from the *outside* to monitor their performance or detect intrusions.

This course will examine resource virtualization from a practical point of view, focusing for the most part on the current state of the art in practice. We will also touch on the history of this technology and on current research results that may become practice in the short run.

As a result of this course, students will be able to:

- Differentiate among the various forms of CPU virtualization: language-level, OS-level, virtual servers, and emulators.
- Explain the differences between “pure” and paravirtualized virtual machine monitors (VMMs) and between type-I and type-II VMMs.
- Understand why VMMs interact with architecture and why the Intel and AMD hardware extensions are changing the face of modern VMMs.
- Have hands-on experience with using a modern VMM, VMWare Server.
- Have a good high-level understanding of the structure and execution behavior of a modern operating system and its interaction with the hardware and/or the VMM.
- Explain at a high-level the rough structure of a VMM and its theory of operation.
- Understand what is meant by network virtualization.
- Understand what is meant by storage virtualization and how it differs from network file systems.
- Explain some of the common scenarios in which virtualization is used today, including server consolidation, software development, debugging, fault tolerance, and security.
- Explain the concept of virtual infrastructure.
- Gain some insight into the futures that industry and researchers expect from virtualization, including specialized high-level service providers, low-level service marketplaces, and adaptive/autonomic computing.

Homework

Students are expected to make use of their access to the VLab to gain some hands on experience with using VMware Server.

Students may also want to use their own computers to install and play with other VMMs.

Students will investigate some topics in depth and write short essays about them.

Teaching Methods

I lecture using chalk and a blackboard, with digressions (e.g., showing software or a web site) using my laptop. I don't use class notes or slides. For this course, I want students to ask lots of questions so that I can dynamically adjust the content of the course to their interests.

Exams

There will be no exams.

Grading

50 % Class participation
50% Homework

Schedule

Week 1 (3 hours) --- Introduction to Virtual Machines

- Introduction and history
- Industry background (VMware, Xen, Microsoft, Sun, IBM, etc)
- Uses of virtualization
 - The security argument (isolation argument, size argument)
 - Development, debugging, and testing
 - IT configuration testing and incremental rollout
 - Server consolidation
 - Live migration and replication of VMs and their consequences for reliability and scaling
 - Desktop replacement via migration or consolidation
 - Attestation and capsule computing
 - Computing marketplaces/clouds (and their current limitations)
 - Adaptive and/or autonomic systems
- **Readings**
 - Hess and Newman, *Practical Virtualization Solutions: Virtualization from the Trenches*. Chapters 1-3, Appendix
- **Assignment**
 - We will go into the lab and you will use your VLab account to create a VMware virtual machine that boots a Linux live distribution.

- Outside of lab, you will use your VLab account to install an operating system in your VM and configure it with NAT networking.
- You'll then use your VM to ping and traceroute www.northwestern.edu, and hand in the ping/traceroute report.

Week 2 (3 hours) --- Review of hardware and operating systems concepts needed to understand a virtual machine monitor.

- Basic machine execution model (x86)
- Privileged and unprivileged modes, instructions, and machine state
- Interrupts
- Paging
- Process abstraction
- Forms of CPU virtualization
 - Language level (abstract virtual machines (JVM, .NET CLR))
 - OS level (Classic “pure” type-I and type-II VMMs (VM/VMware/VirtualPC/KVM/VBox), Paravirtualization (Xen), virtual servers (VServer, Zones), emulators (QEMU))
- **Readings**
 - Hess and Newman, *Practical Virtualization Solutions: Virtualization from the Trenches*. Chapter 5 (on Xen)
 - Chapter 1 (“Introduction to Virtual Machines”) and Appendix A (“Real Machines”) from Smith and Nair, *Virtual Machines: Versatile Platforms for Systems and Processes*
- **Assignment**
 - Install VMware or any other VMM on your personal machine. Create a VM, configure it with networking, install a guest, and then ping a known host. Hand in the ping/traceroute report.

Week 3 (3 hours) --- Overview of how a virtual machine monitor works

- Trap-and-emulate model
- VMexits and VMentries
- Hardware virtualization extensions (e.g. Intel VT, AMD SVM)
- Software-based approaches: code rewriting
- Paravirtualization
- (maybe) Nested and shadow paging
- (maybe) Interrupt virtualization
- Virtual devices
- **Readings**
 - Chapter 8 (“System Virtual Machines”) from Smith and Nair, *Virtual Machines: Versatile Platforms for Systems and Processes*
- **Assignment:**
 - None this week, focus on reading.

Note: Week 4 will be dynamically adjusted based on the experience of students in the class. What is described below is planned material. If it turns out that students do not

have enough background to have it make sense to dive into the codebase, we will instead spend time looking at other forms of virtualization (e.g., Jails, language VMs, hardware/software co-designed VMs) at a high-level.

Week 4 (3 hours) --- Diving into the source code of a real VMM

- Demonstration of OS kernel development
- We will take a high-level walk through the codebase of the Palacios VMM
- **Readings**
 - J. Lange, P. Dinda, *An Introduction to the Palacios Virtual Machine Monitor---Release 1.0*, Technical Report NWU-EECS-08-11, Department of Electrical Engineering and Computer Science, Northwestern University. (<http://v3vee.org/papers/NWU-EECS-08-11.pdf>). In particular, read Section 1 and Sections 6-10).
 - Browse the source code repository for Palacios (<http://v3vee.org/download>, Release 1.2 branch→main development branch)
- **Assignment**
 - Browse the Palacios source code repository via the web. Find the first commit of the PCI passthrough feature and the latest commit related to it. Note the commit dates and numbers, then send those, along with your own explanation of what was changed to me via email.)

Week 5 (3 hours) – Virtual Infrastructure and Clouds

- Datacenters and Clusters
- Virtual networks (overlays)
- Storage area networking
- Clouds (maybe Amazon EC2 example) and Grids
- Adaptive computing
- Virtualization of high performance computing resources
- **Readings**
 - Hess and Newman, *Practical Virtualization Solutions: Virtualization from the Trenches*. Chapters 9-11
 - Chapter 10 (“Emerging Applications”) from Smith and Nair, *Virtual Machines: Versatile Platforms for Systems and Processes*
- **Assignment**
 - Research and then write a short (3-4 page) essay comparing and contrasting IaaS, SaaS, and PaaS clouds.