Resource Virtualization

Syllabus

Web Page

http://pdinda.org/virt

Instructor

Peter A. Dinda
Technological Institute, Room L463
847-467-7859
pdinda@northwestern.edu
Office hours: Thursdays, 2-4pm or by appointment

Location and Time

Lecture: Tech M164 (we may move)
Tuesdays and Thursdays, 9:30-10:50am

Prerequisites

Required	EECS 213 or (EECS 205 and EECS 231/211) or equivalent computer systems course
Required	Familiarity with C
Recommended	CS 343 or equivalent operating systems course

Readings

For the most part, readings for the course will be in the form of research papers. A separate reading list is provided. **There is no textbook, but you should buy the following reference book**:

J. Smith, R. Nair, *Virtual Machines: Versatile Platforms For Systems And Processes*, Morgan Kaufmann, 2005.

Code

We will spend a substantial fraction of the class examining the code of Palacios, a virtual machine monitor operating system under development in my group. You can download the public release of Palacios, and the technical report describing it from v3vee.org.

Objectives, framework, philosophy, and caveats

A basic principle in computer science is that of indirection, creating a new layer between two existing layers. By introducing a new layer into a software system, it often becomes straightforward to do many things and the system becomes more flexible. However, there is a tension: a new layer *may* make the system slower. A classic example is adding a programming language to a system. For example, the EMACS text editor has a core data manipulation library written in C coupled with a Lisp interpreter, through which the library can be used. Much of the editor, and the many extensions people have added to it are written in Lisp, which is arguably much easier than writing them in C and sufficiently fast for text editing.

Currently, there is considerable excitement in the operating systems, networking, and distributed systems communities over *resource virtualization*. Beyond academic interest, resource virtualization has become a substantial market, led by companies like VMware, XenSource (owned by Citrix), Microsoft, Sun, and others. Two implementations of virtualization have been integrated into the Linux kernel. The basic idea is to add a software layer that provides virtual machines, virtual networks, and even virtual services 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 examines resource virtualization, from the highly influential early work in the 1970s to the present.

About half of the course will focus on the *virtual machine monitor (VMM)*, which is an operating system whose abstraction is the same (or very similar to) the underlying hardware. That is, it is an operating system that is able to run other operating systems. VMMs are integral to resource virtualization. We will carefully examine the source code (about 30K lines, mostly C with some assembler) of the Palacios VMM being developed in my group. Palacios is a type-I, non-paravirtualized, whole system VMM that runs on raw x86 hardware using the AMD and Intel architectural extensions for hardware virtualization support. This experience will not only introduce you to how a particular kind of VMM works, but, more generally, how the x86 architecture works and how operating systems interact with it.

In addition to examining a VMM at this deep level, we will also cover the following aspects of resource virtualization by considering original research literature:

- Architectural support for virtualization (in VMM content)
- Traditional OS-level virtualization (in VMM content)
- Virtual devices (in VMM content)

- Paravirtualization
- Virtual servers (if time)
- Emulation and binary translation
- Virtual networking and overlays
- Virtual machine migration
- Virtualization-based computing environments
- Security and virtual machines (if time)

Although this course has been taught several times before, this is the first time it will have the Palacios VMM component. I will adjust the rate of coverage of Palacios to make sure that everyone is on board, and compensate by adjusting the coverage of other materials. At this point, I see us spending the first half of the course reading code, and the second half reading papers.

Scribes

For the first half of the course, where we will be examining the Palacios VMM in depth, the class will collectively be writing a detailed treatment of it. The hope is that this will be published as a technical report with all students being authors.

For each class, two students will be assigned as **scribes**, meaning that they will be responsible for writing the treatment of the lecture. Later, a second group of two students (and I) will be responsible for editing. To simplify this, I will video-tape each of these lectures, and bring a camera along to capture high-resolution images when needed.

Paper Presentations

For the second half of the course, we will spend most of the time reading and discussing papers. Almost all of the readings for the course will be in the form of research papers, with some experience report papers added as well. We will generally read 1-3 papers or equivalent materials for each session, covering fundamental ideas and important recent results. For each session, a student will be assigned to formally present the papers to the group, and then lead a discussion about them. The goal of this is to give you the experience of presenting someone else's work.

Before each class, every student will be responsible for posting a paragraph-long comment on the papers to the class discussion group. The goal of this is to assure that every student is prepared for discussion. Further, such short comments are what you should be writing anyway as you read research papers.

Graduate Students and Undergraduates

I will grade graduate students and undergraduates separately. However, whether you are a graduate student or undergraduate, you should have high expectations of yourself! Projects in the previous iterations of the course resulted in several papers published at high quality workshops and conferences. Three Ph.D. dissertations also got their start in this course… so far.

Whether you're a graduate student or undergraduate, realize that, beyond academia, this course gives you the opportunity to learn how an OS and a VMM work, *and to work on them yourself.* You can learn *kernel development skills* in this class, and such skills are rare.

Tools

Students will be given accounts in the Tlab and the Wilkinson Lab. Note that the Wilkinson Lab has recently been remodeled and is a very comfortable place to work as a team. If needed, students in the course will also be given account on the VLab. The VLab consists of a high end cluster that support virtual machines using VMware Virtual Server. You can find out more about the VLab at http://pdinda.org/vlab/.

The code we will examine in the class can run under the free QEMU full system emulator, which is available for many operating systems, including Windows and the Mac.

Project

Over the course of the quarter, you will apply what you learn to a project of your choice, and then document your project in a high quality paper and open presentation. Project topics will be chosen in consultation with me, but there will be essentially two kinds. The first kind will involve writing extensions or components for Palacios. These kinds of projects give the student the opportunity to test their kernel development skills, and create something that can ship (and certainly be part of a portfolio). The second kind of project will be more specifically research focused, but will also involve a substantial implementation aspect (if you don't build it, it's not systems research). I will hand out a list (on paper) of possible projects 1-2 weeks into the course.

Projects may be done individually or in groups. Project complexity and expectations will be tied to group size. I will expect weekly project reports.

All projects will be presented at a public colloquium at the day/time of the final exam.

Exams

There will be no exams

Grading

30 % Scribe and editing work (class graded collectively!)
10 % Paper presentations
50 % Project, **including weekly progress reports**
10 % Project paper and presentation

Schedule

The schedule for the course will be given on the course web page. Since this is the first time I'm including the large-scale coverage of Palacios, I will adapt the pace and content of coverage to the class.