

Virtualization Overview



Joel Jaeggli
AFNOG SS-E 2013

What are we using this Year?

- Mac-mini servers
- Intel core i7 quad core 8 hyperthreads
- 16GB of ram
- 2 x 256GB SATA SSD
- A pretty hefty server
- Less than \$2k
- Drawbacks
 - One psu
 - OOB is kind of a pain
- Ubuntu / KVM

What is it?

- Virtualization of is the abstraction of the manifestation of a resource from the actual physical instance of that resource.
- What Computing/Network resources can be virtualized?
 - Virtually anything! :)

Anything?

- In the context of this course. We're interested in virtualization along two dimensions:
 - Services
 - Hosts

Resource/Service virtualization

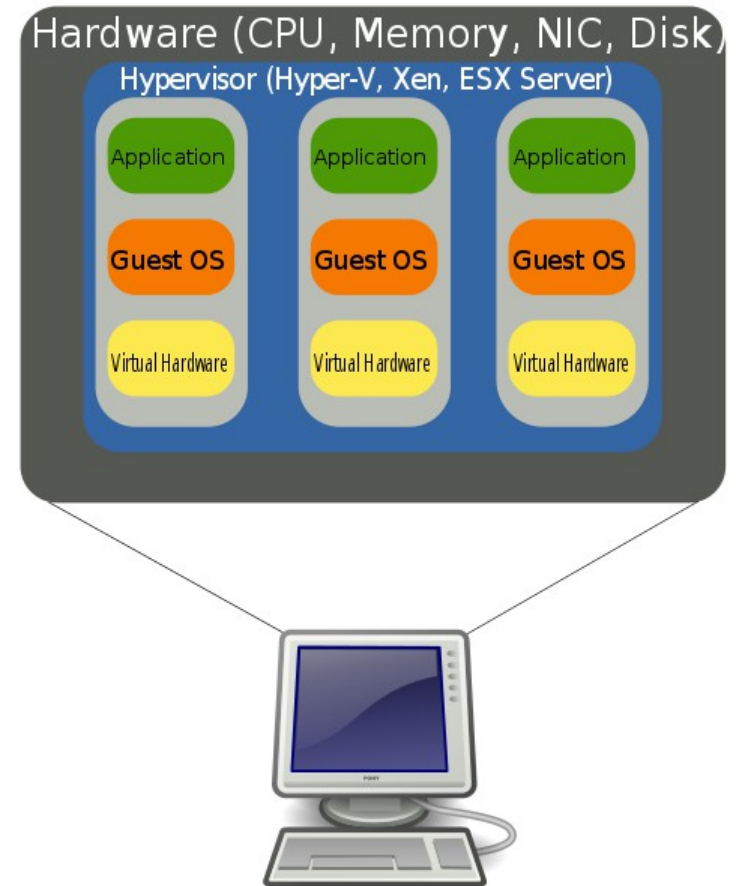
- Examples:
 - Load-balancers
 - DNS Based GLB
 - HTTP(S) Virtual Hosting
 - MX records
 - Virtual Switches
 - Virtual Routers
 - Virtual Firewalls

Resource Virtualization - Continued

- HTTP virtual hosts
 - Multiple websites on one system
- Load Balancing
 - One (or many sites or applications) across many systems
 - Can be done at Layer-3/4/7

Host Virtualization

- Examples
 - VMware
 - Virtual-Box (used in class)
 - KVM
 - XEN
 - FreeBSD and Linux Jails
 - Windows Hyper-V



What problem are we attempting to solve with host virtualization.

- Problem 1 – Idle capacity.
 - Most of the machines in your datacenter are idle most of the time.
 - Capacity you're not using:
 - Cost money up front
 - Cost money to operate
 - Reduces your return on capital
 - Packing discreet systems into a smaller number of servers provides savings along virtually every dimension.

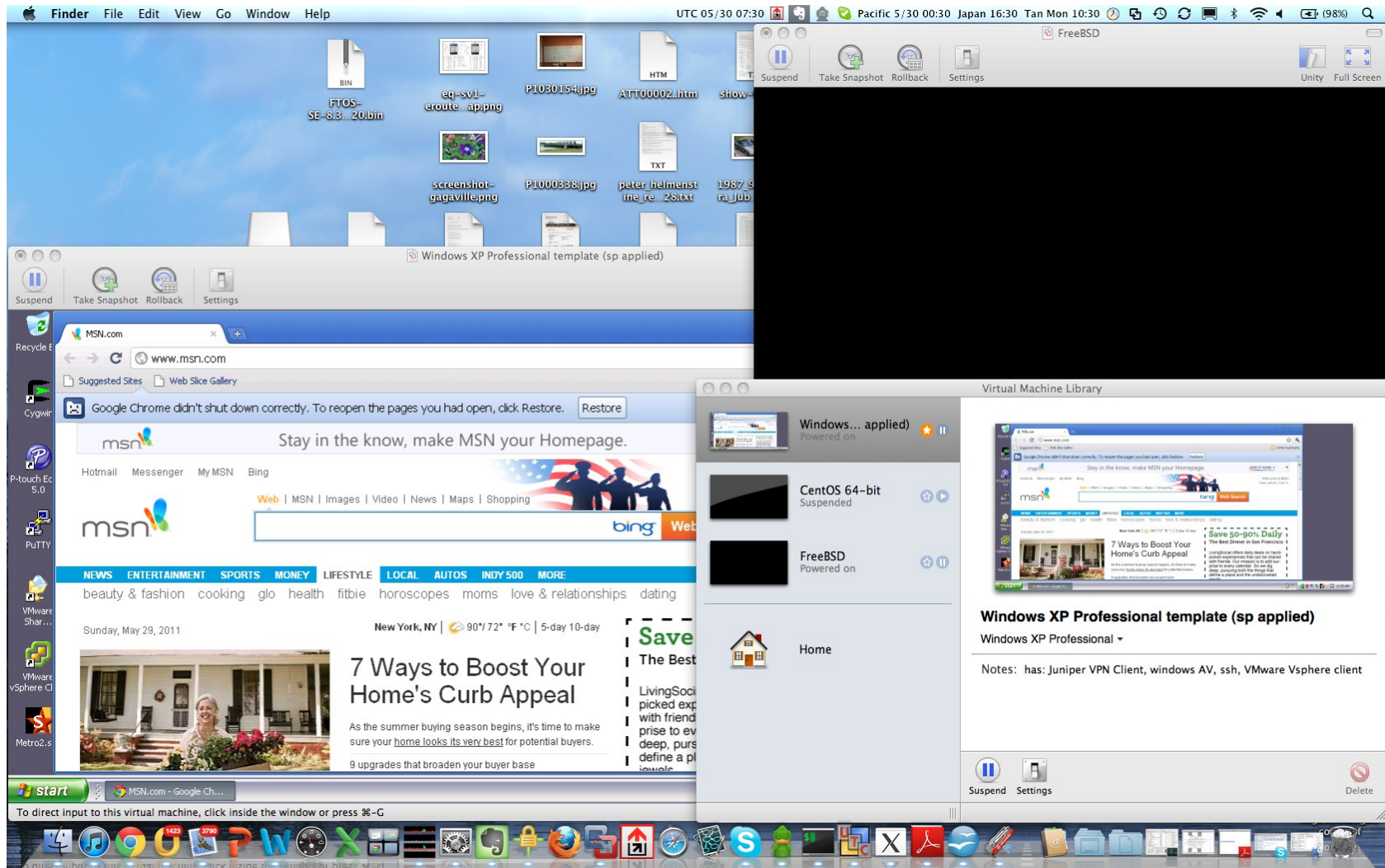
Problems - Continued

- Problem 2 – Provisioning
 - Spinning up a new service involves:
 - Acquiring the hardware
 - Building the server
 - Integration with existing services
 - With virtualization we're aiming to short-circuit that
 - Capacity is a resource
 - Machine instances may be cloned or provisioned from common basic images
 - Resources are purchased in bulk and assigned to applications as necessary.

Problems - Continued

- Problem 3 – Hardware abstraction
 - Operating systems, servers, and applications evolve at different rates.
 - Providing a common set of infrastructure resources means, virtualized systems are portable across servers
 - Hardware failure can more easily be managed.
- Abstraction may come at a performance cost however. (some workloads are more expensive than others)
 - See:
<http://blog.xen.org/index.php/2011/11/29/baremetal-vs-xen-vs-kvm-redux/>

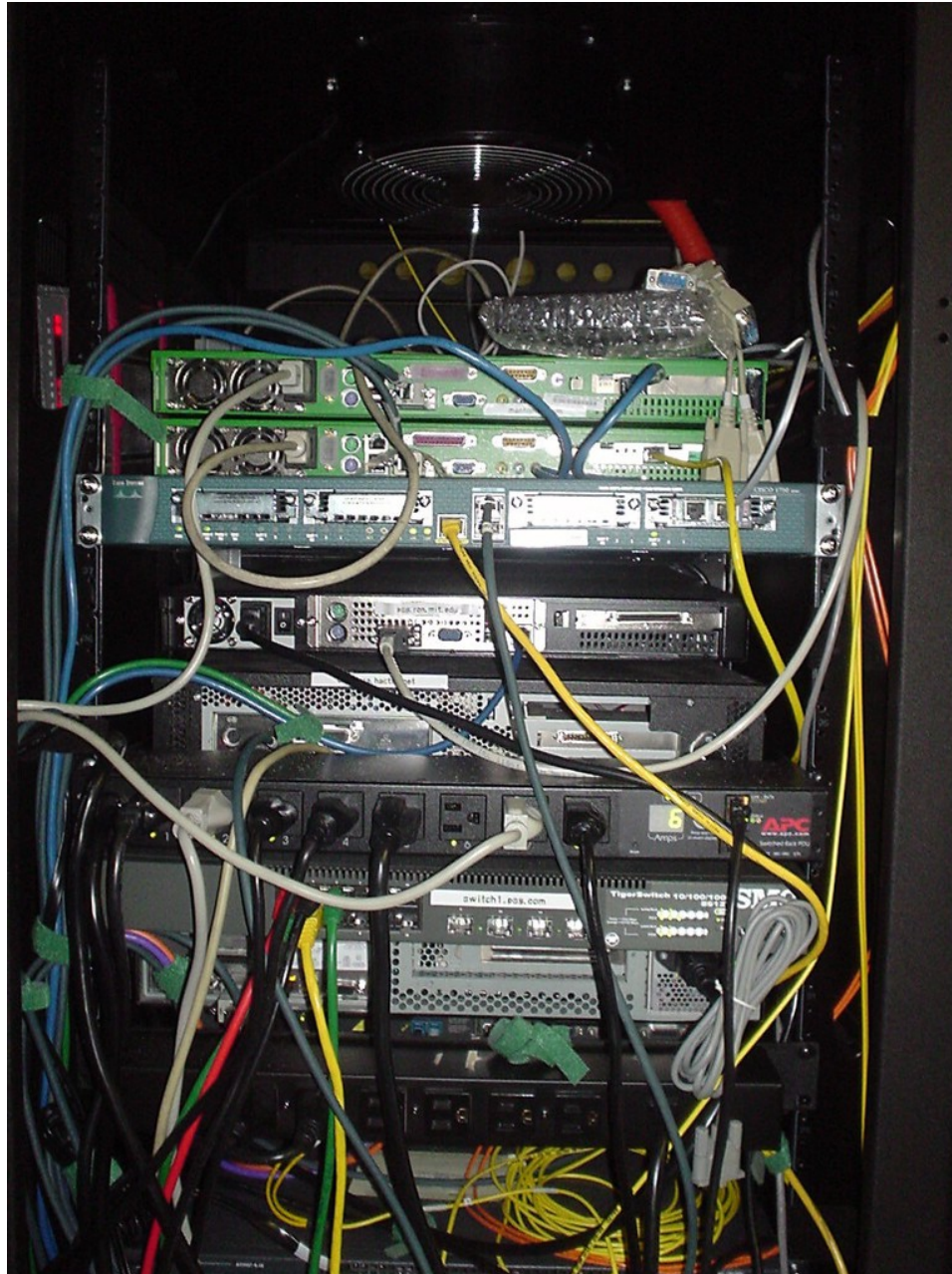
Examples – Desktop Virtualization



Desktop Virtualization

- Uses
 - Prototyping services or applications before deployment
 - Utilities that don't run on your operating system
 - Isolation of sandbox environments from your desktop
 - Maintaining multiple versions of an environment for support purposes.
 - Staying familiar with unix while running windows (consider compared to the alternative (dual-booting))
- Issues
 - Emulating multiple computers on your laptop/desktop is somewhat resource intensive
- Vmware player and VirtualBox are free.
 - <http://www.virtualbox.org/wiki/Downloads>
 - <https://my.vmware.com/web/vmware/downloads>

Examples – Server Virtualization



Server Virtualization - Continued

The screenshot displays the vSphere Client interface for a VMware ESXi host. The left sidebar shows a tree view of the inventory, including the host and several virtual machines. The main pane shows the 'Virtual Machines' tab for the host 'vm0.sea.rg.net'. A table lists the virtual machines with their names, states, and resource usage. Below the table is a 'Recent Tasks' section with a search filter and a table header.

vm0.sea.rg.net VMware ESXi, 4.1.0, 260247

Getting Started Summary **Virtual Machines** Resource Allocation Performance Configuration Local Users & Groups Events Permissions

Name, State or Guest OS contains: Clear

Name	State	Provisioned Space	Used Space	Host CPU - MHz	Host Mem - MB	Guest Mem - %	Notes
ssh.rpki.net	Powered On	104.00 GB	104.00 GB	39	4165	3	
turing.wps.com	Powered On	102.00 GB	102.00 GB	5	2105	0	
ran.psg.com	Powered On	154.00 GB	154.00 GB	39	4187	3	
cent0.psg.com	Powered On	102.00 GB	102.00 GB	51	1453	1	
apnicrpki.rand.apnic.net	Powered On	5.29 GB	5.29 GB	57	717	9	
rip1.psg.com	Powered On	38.00 GB	38.00 GB	41	2085	4	
hiroshima.bogus.com	Powered Off	254.00 GB	250.00 GB	0	0		
nagasaki.bogus.com	Powered On	1023.00 GB	439.14 GB	124	4257	9	

Recent Tasks Name, Target or Status contains: Clear X

Name	Target	Status	Details	Initiated by	Requested Start Ti...	Start Time	Completed Time
------	--------	--------	---------	--------------	-----------------------	------------	----------------

Tasks root

Server Virtualization

The screenshot displays the vSphere Client interface for a VMware ESXi host. The window title is "vm0.sea.rg.net - vSphere Client". The main content area is titled "vm0.sea.rg.net VMware ESXi, 4.1.0, 260247" and shows the "Summary" tab. The interface is divided into several sections:

- General:** Manufacturer: Supermicro, Model: X7DWU, CPU Cores: 8 CPUs x 2 GHz, Processor Type: Intel(R) Xeon(R) CPU E5405 @ 2.00GHz, License: vSphere 4 Essentials Licensed for 2 physical CPU..., Processor Sockets: 2, Cores per Socket: 4, Logical Processors: 8, Hyperthreading: Inactive, Number of NICs: 2, State: Connected, Virtual Machines and Templates: 8, vMotion Enabled: N/A, VMware EVC Mode: N/A, Host Configured for FT: N/A, Active Tasks: 0, Host Profile: N/A, Profile Compliance: N/A.
- Resources:** CPU usage: 750 MHz (Capacity: 8 x 2 GHz), Memory usage: 20163.00 MB (Capacity: 32766.17 MB).
- Datstore:** A table showing the capacity and free space of the datstore1.
- Network:** A table showing the network configuration for the VM Network.
- Fault Tolerance:** Fault Tolerance Version: 2.0.1-2.0.0-2.0.0, Total Primary VMs: 0, Powered On Primary VMs: 0, Total Secondary VMs: 0, Powered On Secondary VMs: 0.

The "Recent Tasks" section at the bottom shows a table with columns: Name, Target, Status, Details, Initiated by, Requested Start Time, Start Time, and Completed Time. The table is currently empty.

Virtualized Servers as a Service (Amazon Web Services)

- Much as collocated servers, are available from a hosting provider, virtual servers are also available.
- Model is:
 - You pay for what you use.
 - Flexibility, need fewer servers today than you used, yesterday.
 - Leverage other amazon tools (storage/map-reduce/load-balancing/payments etc)

AWS

The screenshot shows the AWS Management Console interface for the 'My Instances' page. The browser address bar shows the URL: <https://console.aws.amazon.com/ec2/home?region=us-west-1#s=Instances>. The top navigation bar includes links for 'Welcome, Joel Jaeggli', 'Settings', and 'Sign Out'. The left sidebar contains navigation options for 'EC2 Dashboard', 'INSTANCES', 'IMAGES', 'ELASTIC BLOCK STORE', and 'NETWORKING & SECURITY'. The main content area is titled 'My Instances' and features a 'Launch Instance' button and 'Instance Actions' dropdown. Below this, there are filters for 'Viewing: All Instances' and 'All Instance Types'. A table lists two instances:

Name	Instance	AMI ID	Root Device	Type	Status	Security Groups	Key Pair Name	Monitoring	Virtualization
<input checked="" type="checkbox"/>	i-7a255b3e	ami-c7643482	ebs	t1.micro	terminated	default	joelja_amazon_key	basic	paravirtual
<input type="checkbox"/>	empty	i-ac1213e8	ebs	t1.micro	running	bare	amazon	basic	paravirtual

Below the table, a message states: 'You have selected an instance that is not running. Amazon CloudWatch Detailed Monitoring can only be enabled for running instances.' Four monitoring charts are displayed for the selected instance:

- Avg CPU Utilization (Percent):** A line chart showing utilization over time, with values near 0%.
- Avg Disk Reads (Bytes):** A line chart showing disk reads over time, with values near 0.0.
- Avg Disk Writes (Bytes):** A line chart showing disk writes over time, with values near 0.0.
- Max Network In (Bytes):** A line chart showing network input over time, with a sharp spike to approximately 5,000 bytes.

The footer of the console contains the copyright notice: '© 2008 - 2011, Amazon Web Services LLC or its affiliates. All right reserved.' and links for 'Feedback', 'Support', 'Privacy Policy', 'Terms of Use', and 'An amazon.com company'.

AWS Steps

- Select availability zone
- Launch new instance
- Select appropriate ami
- Associate with ssh key
- Launch instance
- Add ip
- SSH into new machine instance.
- t1-micro-instances run \$54 a year + bandwidth

Try it for free...

- Free tier for the first Calendar year is (per month):
 - 750 hours of EC2 running Linux/Unix Micro instance usage
 - 750 hours of Elastic Load Balancing plus 15 GB data processing
 - 10 GB of Amazon Elastic Block Storage (EBS) plus 1 million IOs, 1 GB snapshot storage, 10,000 snapshot Get Requests and 1,000 snapshot Put Requests
 - 15 GB of bandwidth in and 15 GB of bandwidth out aggregated across all AWS services
- Which is not to say that, at scale EC2 is particularly cheap, (It isn't)
 - Limited capital at risk is in the context of prototyping or experimentation however.

AWS - Continued

- For provisioning purposes cli interaction is possible:
 - <http://aws.amazon.com/developertools/351>
- Along with tools to support the provisioning and destruction of virtual machines.

Provisioning and management

- Is the glue that makes virtualization usable
- In commercial virtualization environments the provisioning/management toolkits represent the bulk of the licensing cost (VMware) and the secret sauce (VMotion, disaster recovery, backup, etc)
- Examples:
 - XEN tools – a collection of perl scripts for spinning VMs
<http://www.xen-tools.org/software/xen-tools/>
 - KVM tools - http://www.linux-kvm.org/page/Management_Tools
 - Cloud.com/cloud-stack (orchestration) - <http://www.cloudstack.org/>
 - Rightscale – (orchestration multiple public/private clouds)
<http://www.rightscale.com>
 - Puppet (host / configuration management) -
<http://puppetlabs.com/puppet/>
 - PDSH – (Parallel Shell execution) <http://code.google.com/p/pdsh/>

Variation In virtualized environments

- Enterprise and Government virtualized environments may tend towards heterogeneity.
 - e.g. the applications (servers) that are being virtualized have accumulated over time
 - Are different enough that management may be depressingly manual
- ASP/Internet services environments may be more homogenous.
 - Leverage a common set infrastructure primitives
 - Thousands of like-systems providing overlapping functionality across hundreds of servers
 - Traditional network elements (e.g. loadbalancers/firewalls) may be virtualized along with the application.

Can you spot the...

- Web-node?
- Database-node?
- Load-balancer?
- Nameserver?
- DHCP Server?
- Email cluster?
- Devnodes?



Complimentary technologies

- NIC teaming or Link aggregation
- Network attached storage and network centric filesystems
 - NFS
 - Hadoopfs
 - GFS2
- Distributed databases
 - Example mysql cluster
 - Couchbase/Membase
 - OracleRAC