



VMware Experience Report

Developing with VMware and Java

Overview

- ▶ Key Concepts
- ▶ Usage Scenarios
- ▶ Tips & Tricks
- ▶ VMware in the Real World
- ▶ Demo: Weblogic Cluster

Virtual Hardware

- ▶ VMWare is really a hardware platform
- ▶ Guest OSes never see the „real“ hardware
- ▶ Virtual Hardware is largely independant of Host OS and physical hardware

Virtual Networking

- ▶ Forget Bridged, Host-Only and Custom Networking: a Network is a Network
- ▶ Work just the same as physical networks: can be bridged, sniffed, routed through, etc.
- ▶ Can be “plugged into” by both physical and virtual machines

Properties of Virtual Machines

- ▶ Similar to OO: Encapsulation, Inheritance, Reusability, Extendability and Clonability
- ▶ Similar to Java Platform: Hardware Independence, OS Independence, Scalability
- ▶ Unique to VMWare: Virtual Hardware, Virtual Networking, Undoable State, Multiplicity

Usage Scenarios 1/6

- ▶ Install any platform in an enterprise without disrupting the managed environment.

Developers stay out of admin's hair

Encapsulation

- ▶ Test the same software on a variety of different platforms

Clonability, Undoable State

- ▶ Build a distributed test environment that mirrors a real target environment

Virtual Hardware and Virtual Networking

Usage Scenarios 2/6

- ▶ Keep software projects in separate VMs to reduce conflict potential in the environment.
DLLs, registry settings and JDK versions are kept separate.

Encapsulation

- ▶ Increase productivity by taking „snapshots“ of virtual machines at key stages of a build.
Easy „one file“ backup of an entire build, special builds are based on (inherited from) more general builds.

Clonability, Inheritance

Usage Scenarios 3/6

- ▶ Keep projects in a working and tested environment... forever ;-)
Encapsulation, Hardware Independence
- ▶ Deploy VMs instead of server applications to avoid server environment conflicts and dependancies
Clearer line of responsibility between the software provider and the organization or company that contracted the software
Encapsulation, Clonability

Usage Scenarios 4/6

- ▶ Run server components such as file servers, web/ftp servers and application servers in virtual machines to improve scalability and load balancing.

Hardware can be upgraded and resources can be adjusted with almost no disruption of the software environment.

Hardware Independence, Clonability, Scalability

- ▶ Make the most out of reboot time (windows) by working in another VM

Multiplicity

Usage Scenarios 5/6

- ▶ Kickstart software projects by giving developers VMs with preinstalled working environments.

Potentially hard to install/configure tools are easily distributed to developers.

If VM's guest OS or installed software becomes corrupted or unusable, the original can simply be re-cloned.

Clonability, Reusability

Usage Scenarios 6/6

- ▶ Try out new software or tools without the danger of interfering with a working production environment.

Encapsulation, Undoable State

Tips & Tricks 1/5

- ▶ Get to know the hot keys.
Particularly important for switching between VMs in full screen mode.
- ▶ Install both a bridged and host only virtual NIC when using Win2000 host OS.
Ip-stack „disappears“ on Win2000 when network cable is unplugged.
- ▶ Turn off „autorun“ in windows guest OSes or uncheck the start connected property.
Avoids all guest OSes „autorunning“ whenever a CD is inserted.

Tips & Tricks 2/5

- ▶ Uncheck the start connected property for floppy.
Not usually needed, avoids messages every time a VM is started/resumed.
- ▶ Enable the „use SHIFT in hotkey sequences“ option for Linux guest OSes.
Avoids the mouse-cursor-of-death in X.
- ▶ Set up file sharing on host OS, use as file server.
Makes exchanging and backuping data easier.

Tips & Tricks 3/5

- ▶ Beware of external dependances when deciding what data to store inside and what to store outside of the VM.
This can become critical when a VM is moved to a new host OS/machine.
- ▶ Do not keep your working source code tree on a host OS share.
Clock synchronization can lag enough to cause warnings and/or errors with project build tools, such as make and ant.

Tips & Tricks 4/5

- ▶ Try to use DHCP as much as possible.
Makes managing IP addresses easier.
- ▶ For mobile users with bridged networking and DHCP: keep a script handy in each VM that releases and renews the DHCP lease.
Use this when the physical network has changed.
- ▶ Most users should stick to virtual disks
Flexability vs. performance

Tips & Tricks 5/5

- ▶ Always completely shutdown the guest OS before moving a VM between OSes (same or different physical machine) or between physical machines.
Otherwise the memory layout of the suspended state will no longer match that of the new environment.
- ▶ After moving VMs between OSes and/or physical machines, DO NOT FORGET to change the copied configuration to point to the new virtual harddisk(s).
Easy mistake to make; results in chaos.

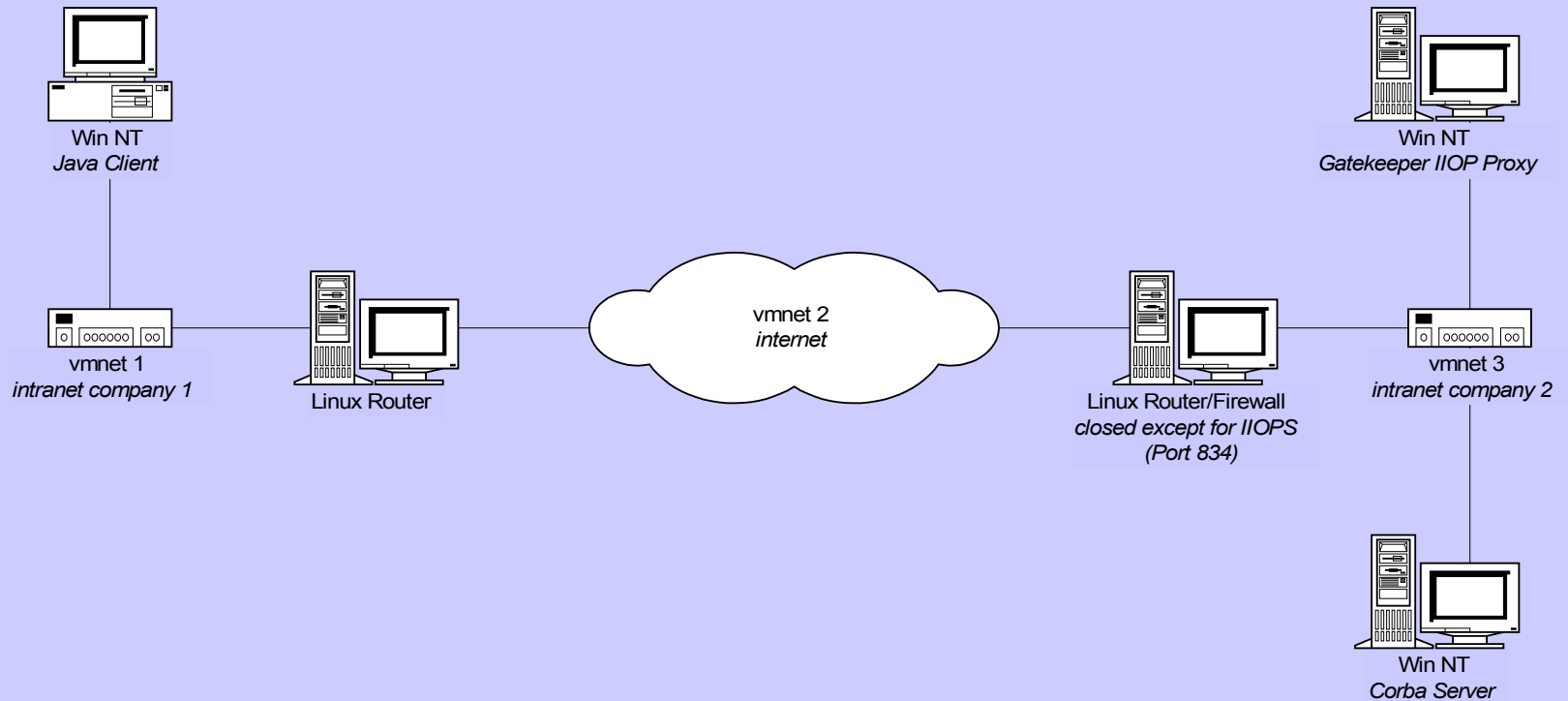
Wishlist 1/2

- ▶ MultipleWorlds Technology for desktop version
- ▶ extended support for virtual networks
 - more than four networks (preferably n)
 - bridging service talks to virtual NIC instead of proprietary method
- ▶ open architecture for adding third party hardware support
- ▶ open APIs for third party vendors
- ▶ full documentation for all command line tools (vnetconfig)

Wishlist 2/2

- ▶ fully documented specification of the config file format
 - one format for all platforms
 - use XML
- ▶ directx support
- ▶ support for all hardware platforms (solaris), even at the expense of performance trade-offs
- ▶ make vmtools more usable under Linux

Simulated Internet Application

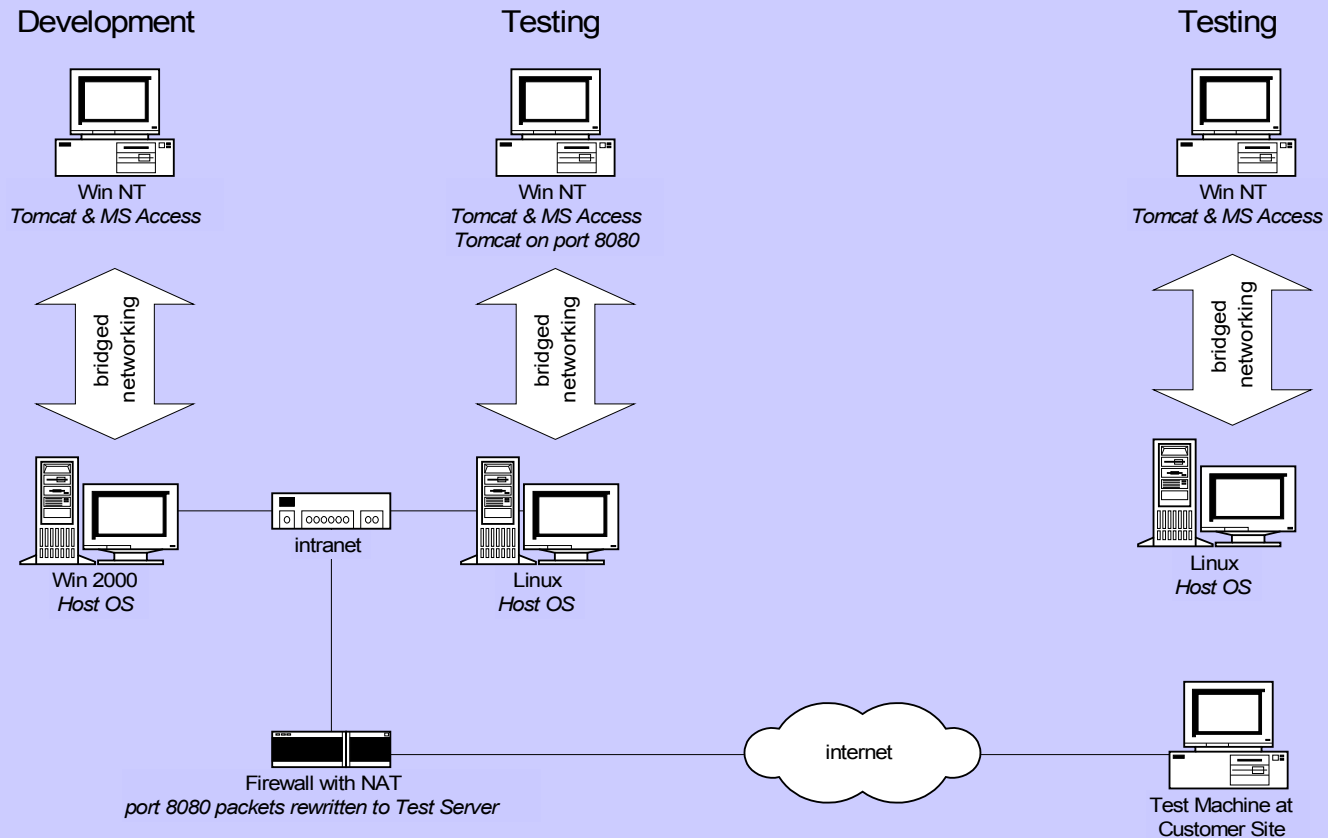


Host OS
Linux on 400MHz Intel PII, 256MB RAM

VM Deployment of Web Application

Ideartis

Customer



Weblogic Server Cluster

