

# Introduction to Virtualization

Kun Suo

# About Me

Kun Suo (索坤), Tony

- 2013～Now, Ph.D in Computer Science in UCCS
- 2012～2013, Kyoto University
- 2008～2012, Bachelor in Software Engineering in Nanjing University (NJU)

## Research Interests

- Cloud computing
- Virtualization
- Operating System
- Email: [ksuo@uccs.edu](mailto:ksuo@uccs.edu)

# Schedule



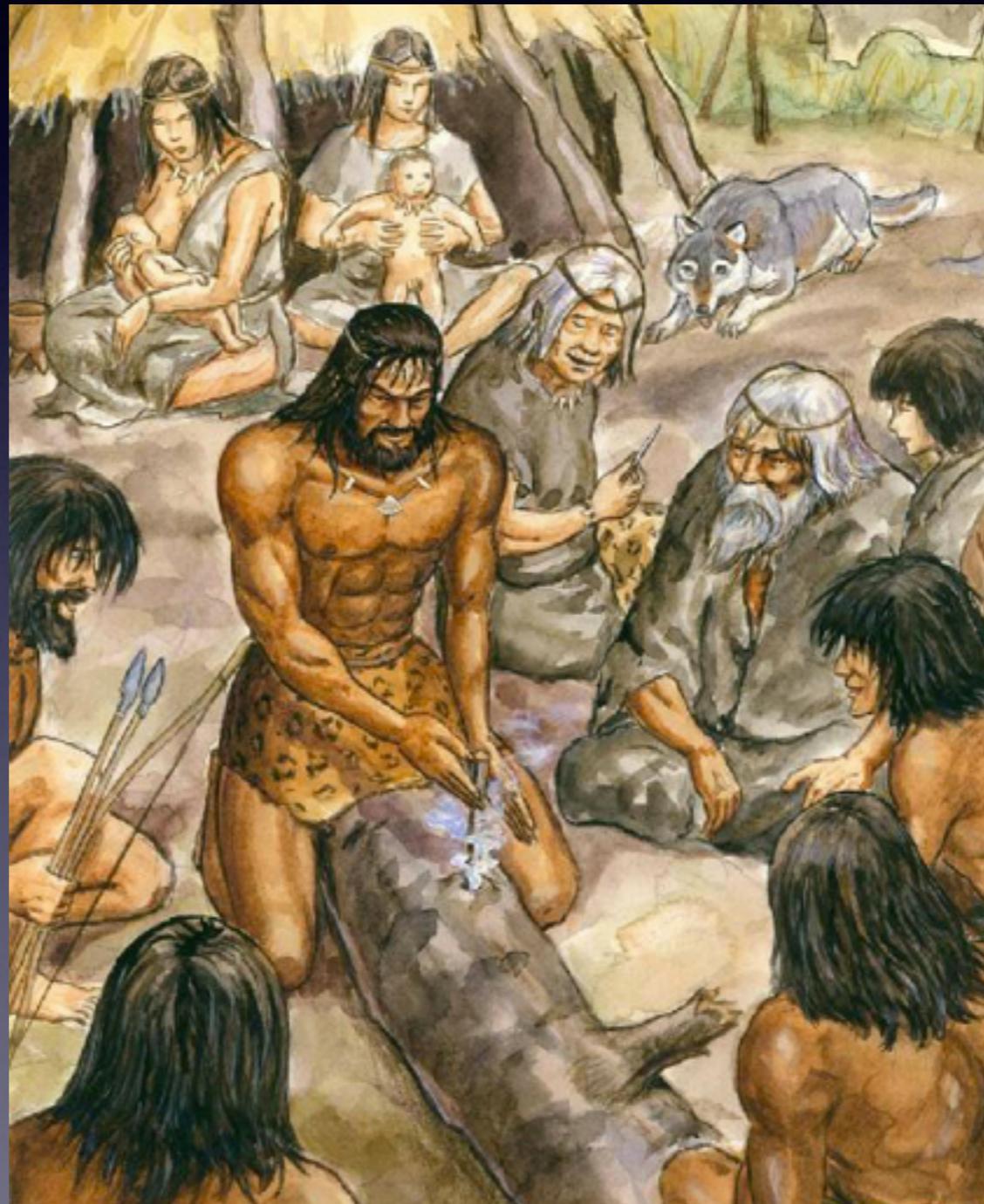
- Monday: Introduction
- Tuesday: Xen
- Wednesday: Xen
- Thursday: Docker
- Friday: Conclusion/homework solution

# Topic Today

- Why needs virtualization ?
- What is virtualization ?

Why need virtualization ?

# The energy to people in ancient



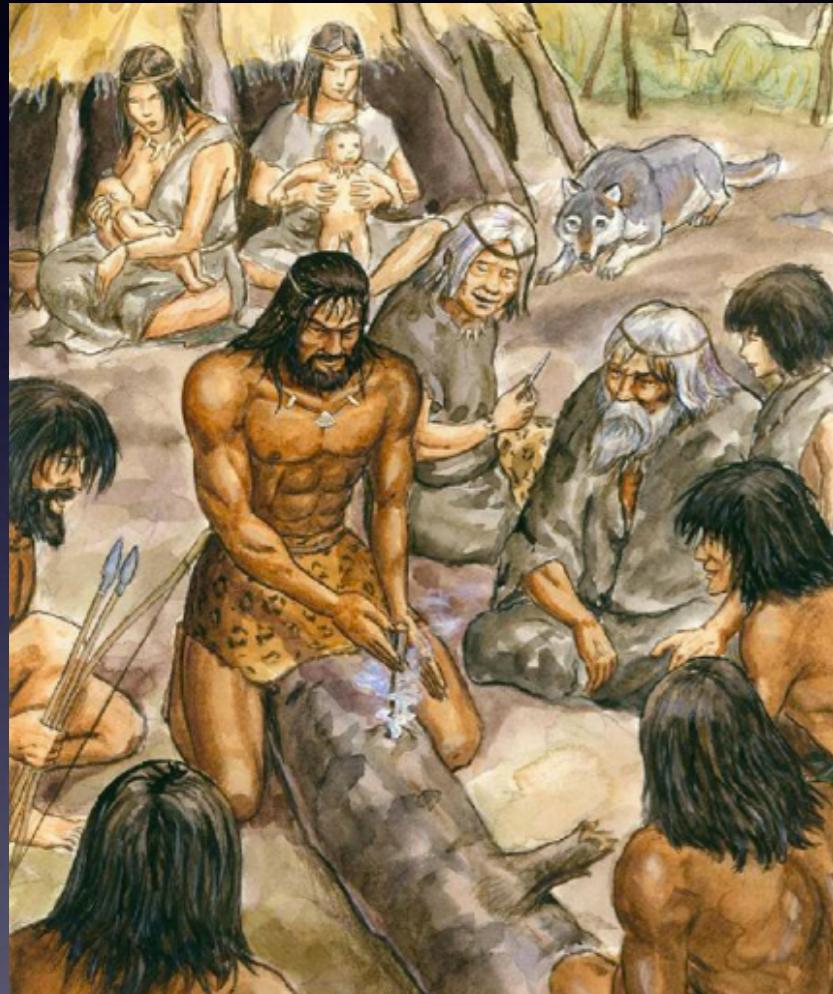
# The energy to people in 21 century



nipic.com/nipic

# The energy to people in 21<sup>st</sup> century

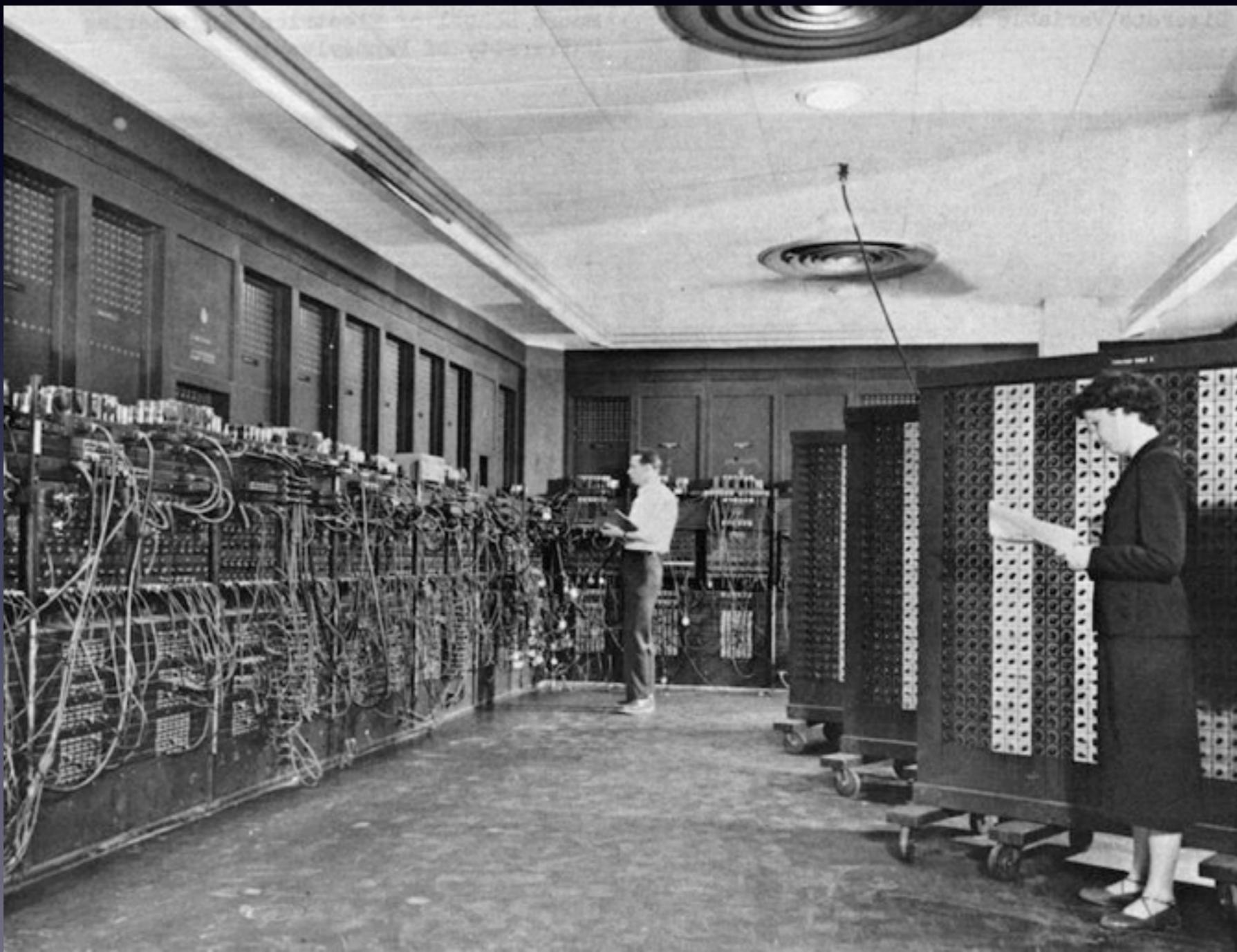
云的发展就像电的发展。一百年前，家家户户都有自己的发电机，后来当时的农场和公司都逐渐关闭了自己的发电机，转而向更加高效的发电厂购买电力。



- 0, buy, not self-produce
- 1, unlimited
- 2, low-cost
- 3, pay as you go

- Through energy “virtualization”, you cannot feel the difference of energy sources, energy distance, energy intensity, etc.

# The computation to people in 1940s



# The computation to people in nowadays



# The computation to people in nowadays



- Through computation “virtualization”, you cannot feel the difference of physical resources, resources distance, resources quantity, etc.
- 0, buy, not self-produce
- 1, unlimited
- 2, low-cost
- 3, pay as you go

# Why need virtualization?

- abstract resources
- high utilization
- reduced cost
- faster redeploy
- greener energy
- elastic computing
- better testing
- better disaster recovery
- easier migration
- higher security

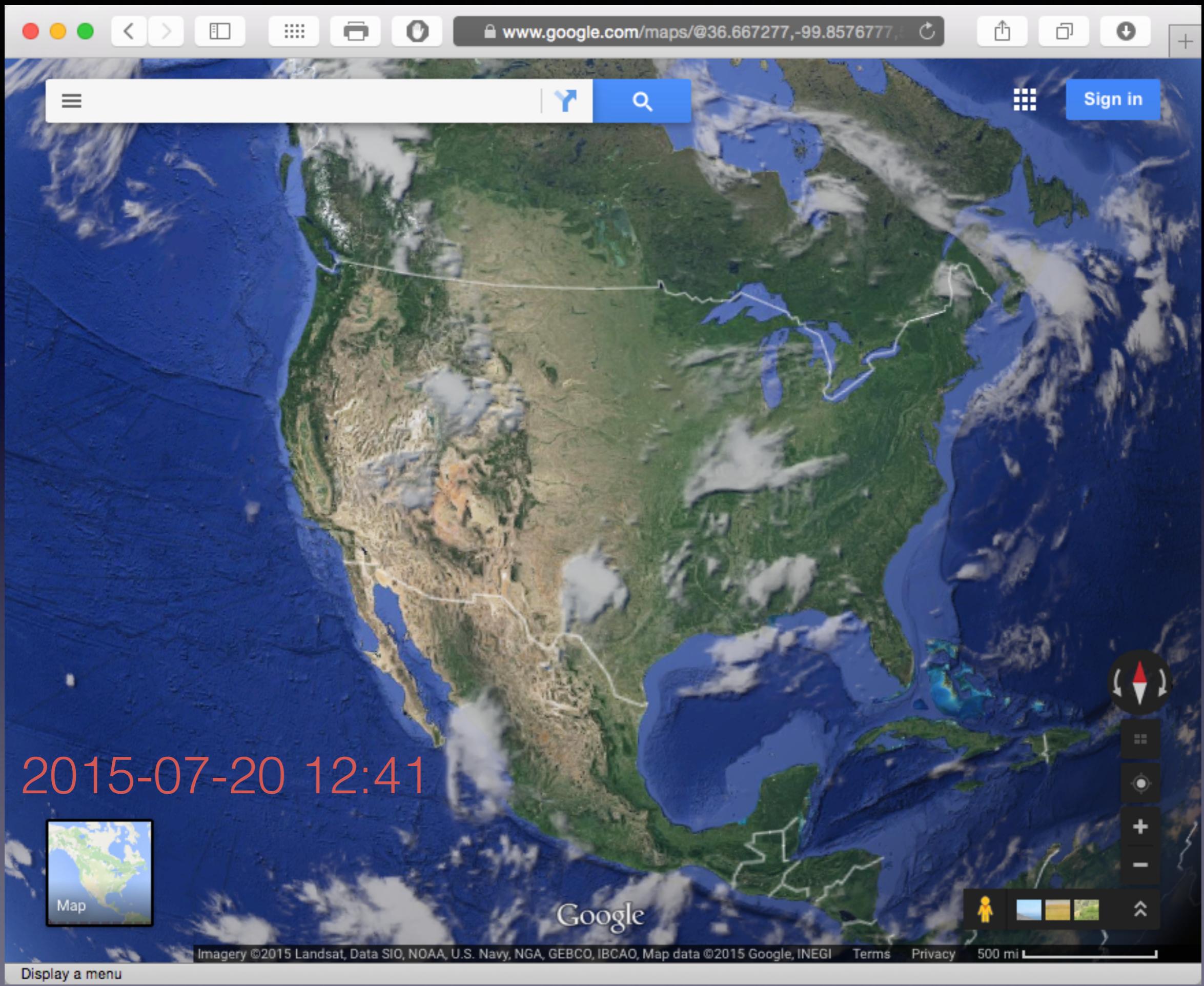
# App Example

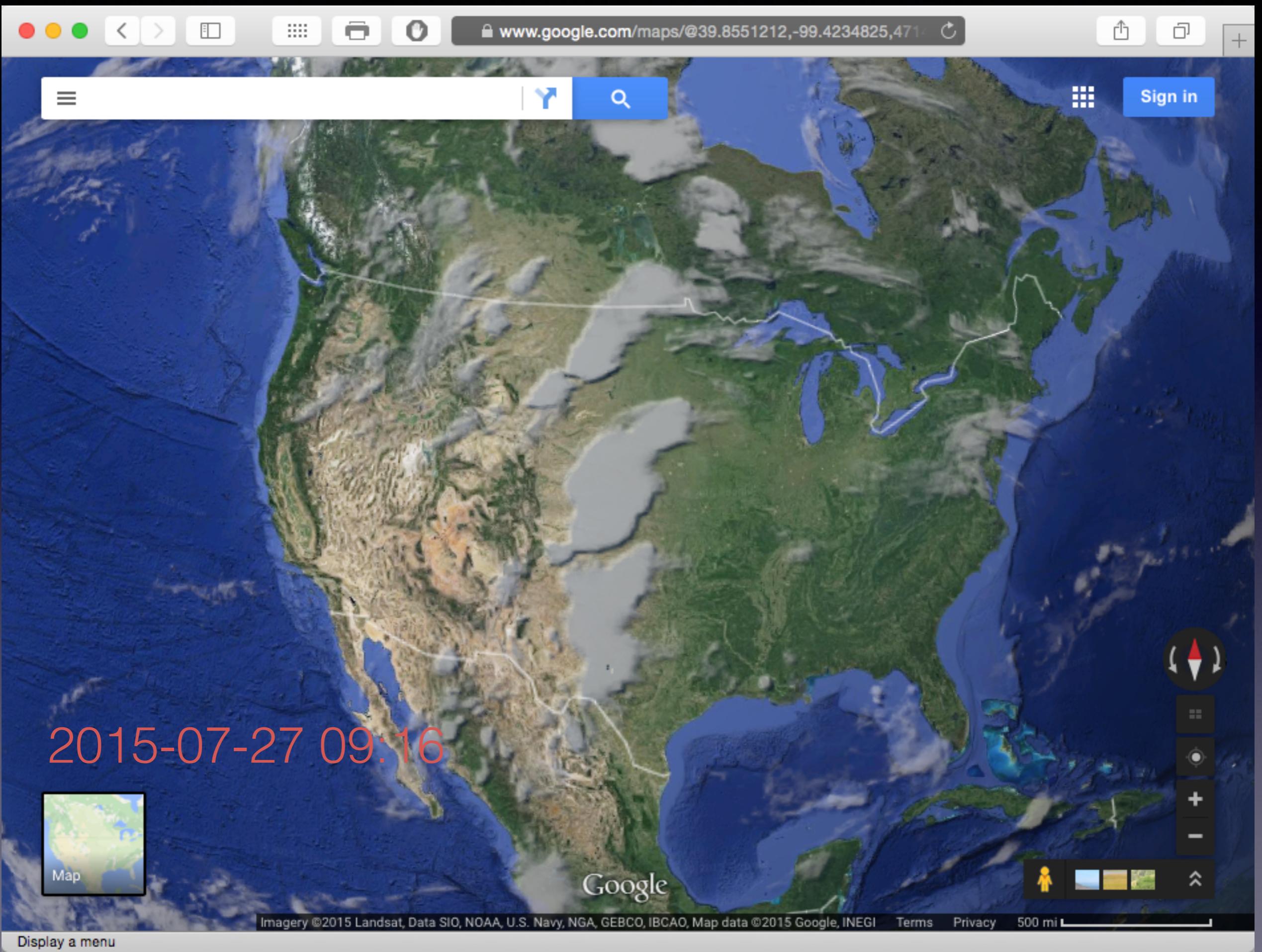
- Huge computation
- Dynamic load
- High reliability
- .
- ... ...

<http://im.qq.com/online/>

<http://onemilliontweetmap.com>

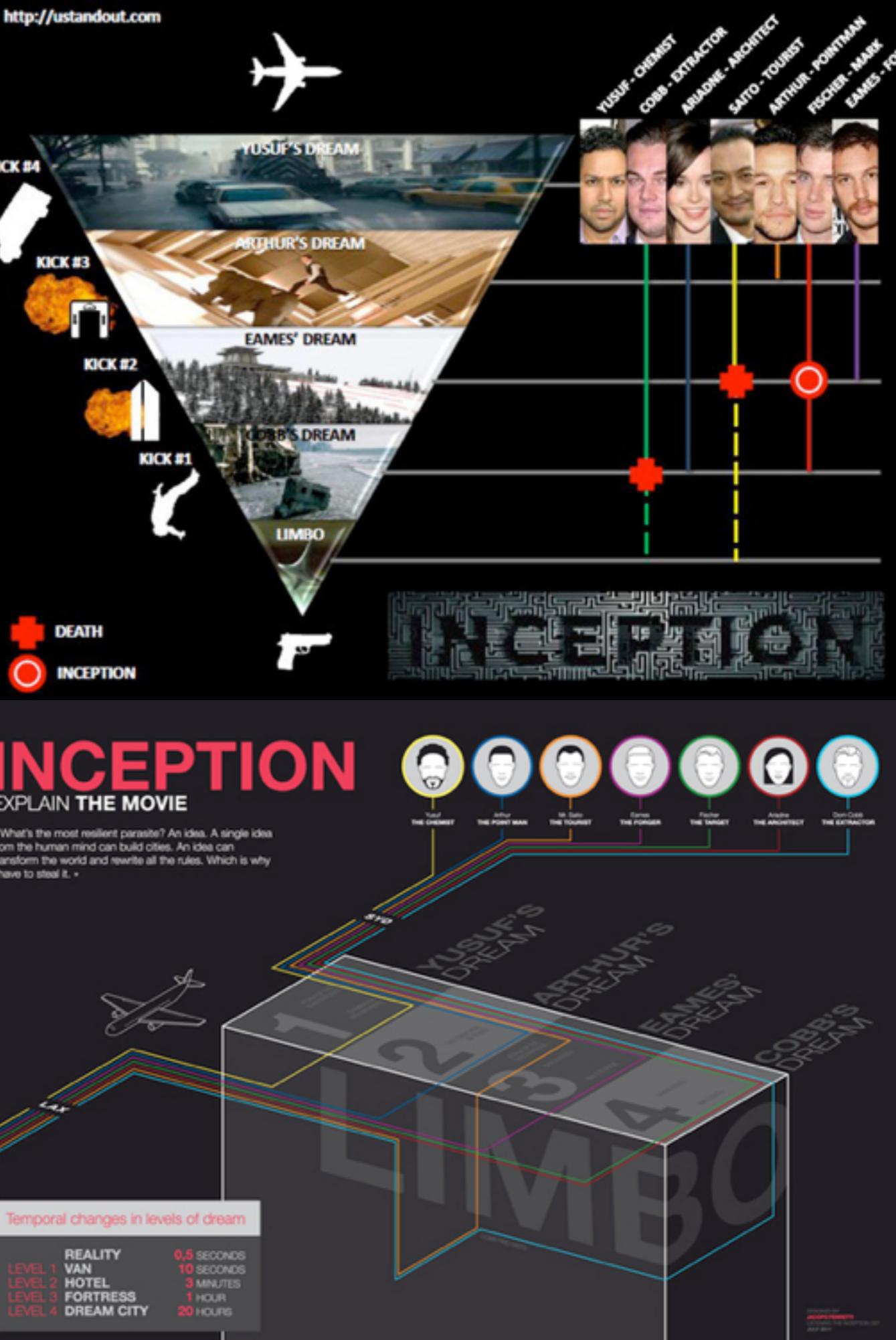
<http://maps.google.com>





# What is virtualization ?





# Actually you have already touched virtualization

- Operating System: time sharing
- Computer Architecture: virtual memory, virtual address
- Java: JVM
- Build VM by VMware, etc
- more... ...

# Looking Back into History



open source,  
software,  
internet,



1991



virtualization,  
cloud computing  
Amazon EC2,

2002

...



?

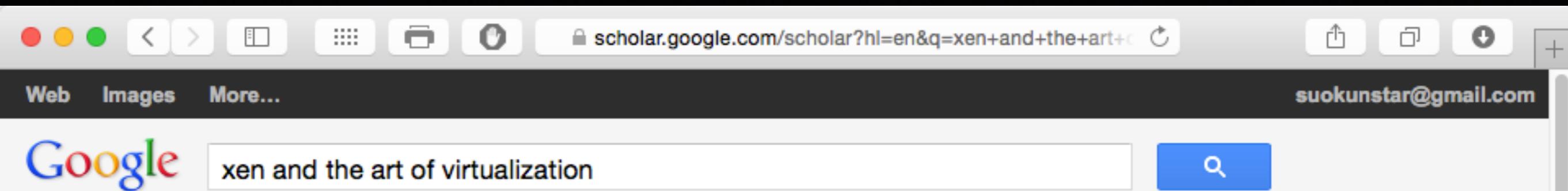
?

...

Today



20年前linux诞生，影响至今有目共睹；  
10之前xen诞生，《xen and the art of virtualization》虚拟化如火如荼大局基本定型；那今天呢，会有哪个开源项目将会用大约十年的时间证明对世界的价值呢？



**Xen and the art of virtualization**

P Barham, B Dragovic, K Fraser, S Hand... - ACM SIGOPS ..., 2003 - dl.acm.org

Abstract Numerous systems have been designed which use virtualization to subdivide the ample resources of a modern computer. Some require specialized hardware, or cannot support commodity operating systems. Some target 100% binary compatibility at the ...

Cited by 6319 Related articles All 247 versions Cite Save

usb.ve [PDF]  
Full-Text @ UCCS

**Xen and the art of virtualization**

B Dragovic, K Fraser, S Hand, T Harris, A Ho, I Pratt... - 2003 - citeulike.org

Abstract Numerous systems have been designed which use virtualization to subdivide the ample resources of a modern computer. Some require specialized hardware, or cannot support commodity operating systems. Some target 100% binary compatibility at the ...

Cited by 405 Related articles All 2 versions Cite Save More

[HTML] **Xen and the Art of Repeated Research.**

B Clark, T Deshane, EM Dow, S Evanchik... - USENIX Annual ..., 2004 - usenix.org

... 7. Conclusions. We were able to repeat the performance measurements of **Xen** published in "**Xen and the Art of Virtualization**" from SOSP-03. We find that **Xen** lives up to its claim of high performance **virtualization** of the x86 platform. ...

Cited by 192 Related articles All 16 versions Cite Save More

usenix.org [HTML]

[PDF] **Xen 3.0 and the art of virtualization**

I Pratt, K Fraser, S Hand, C Limpach... - Linux ..., 2005 - ols.fedoraproject.org

Abstract The **Xen** Virtual Machine Monitor will soon be undergoing its third major release, and is maturing into a stable, secure, and full-featured **virtualization** solution for Linux and other operating systems. **Xen** has attracted considerable development interest over the ...

Cited by 174 Related articles All 15 versions Cite Save More

fedoraproject.org [PDF]

[BOOK] **Running Xen: a hands-on guide to the art of virtualization**

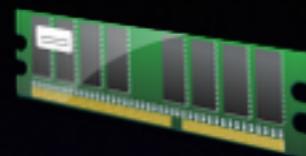
JN Matthews, EM Dow, T Deshane, W Hu, J Bongio... - 2008 - dl.acm.org

Abstract This accessible and immediately useful book expertly provides the **Xen** community

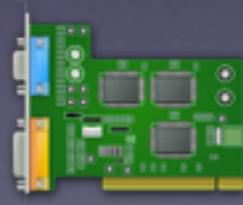
Display a menu

What is virtualization ?

What is computer ?



What we can see...



What we cannot see...

abstract hardware

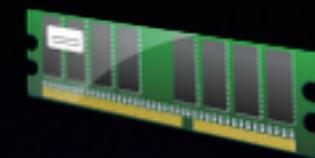
# What is Virtualization?

- Virtualization refers to the act of creating a virtual (rather than actual) version of something, including (but not limited to) a virtual computer hardware platform, operating system (OS), storage device, or computer network resources. ----Wikipedia
- Virtualization is a logic view of resources, not limited to physical.  
---- IBM

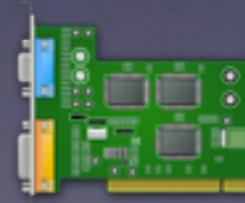
# What is virtualization?

- (1) The object of virtualization is all kinds of hardware resources (like CPU, memory, storage, etc) and software resources (like OS, file system, etc)
- (2) The resources after virtualization hide unnecessary details to the users
- (3) Usually, users can get part or all functionalities in virtualized environment compared to real world

# Virtualization in Different Architecture Levels

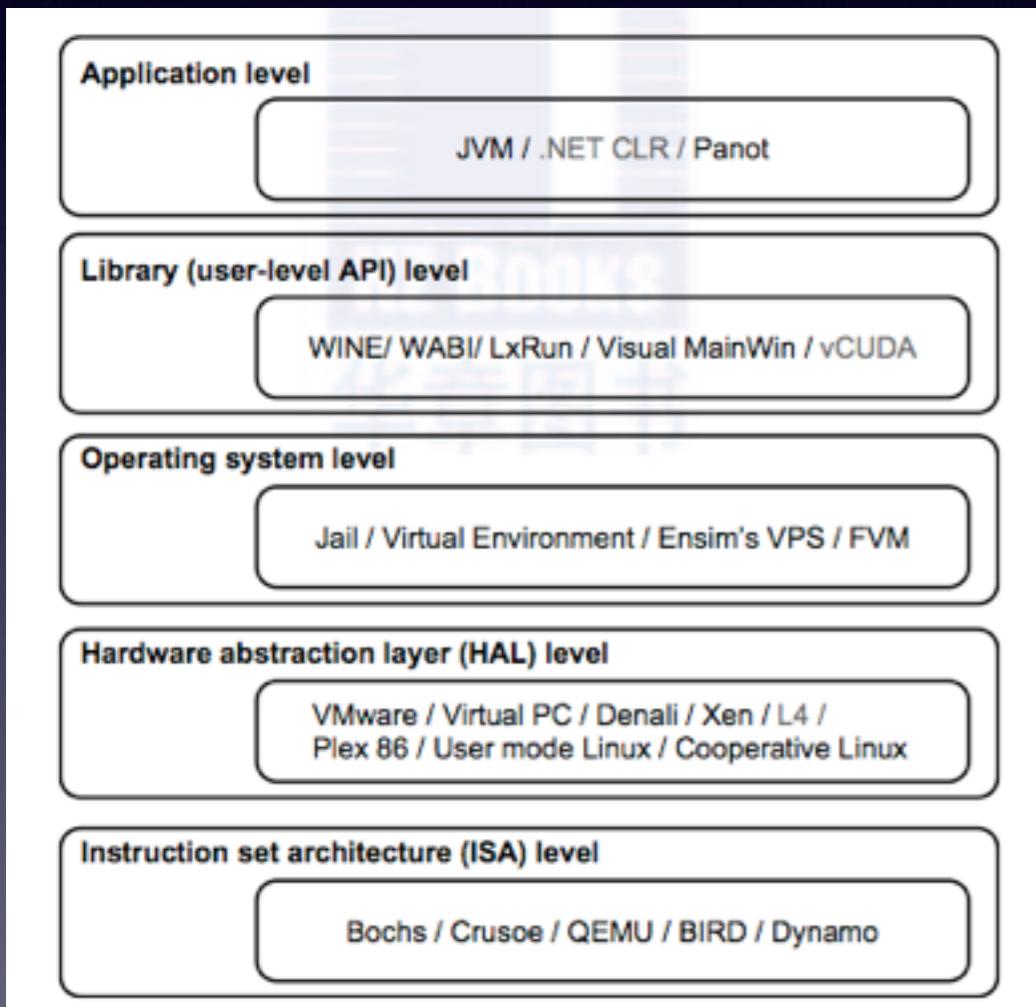


What we can see...



What we cannot see...

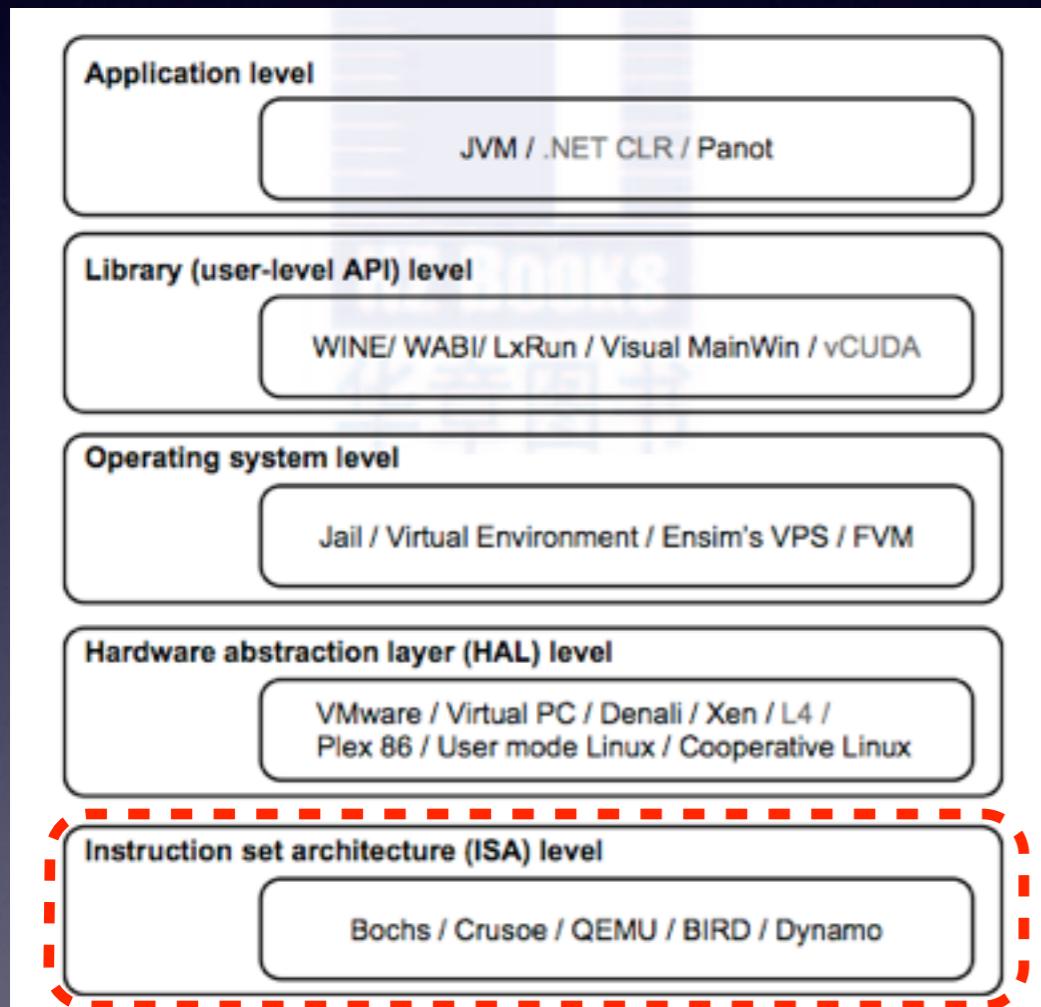
# Virtualization in Different Levels



**Table 3.1** Relative Merits of Virtualization at Various Levels (More "X"'s Means Higher Merit, with a Maximum of 5 X's)

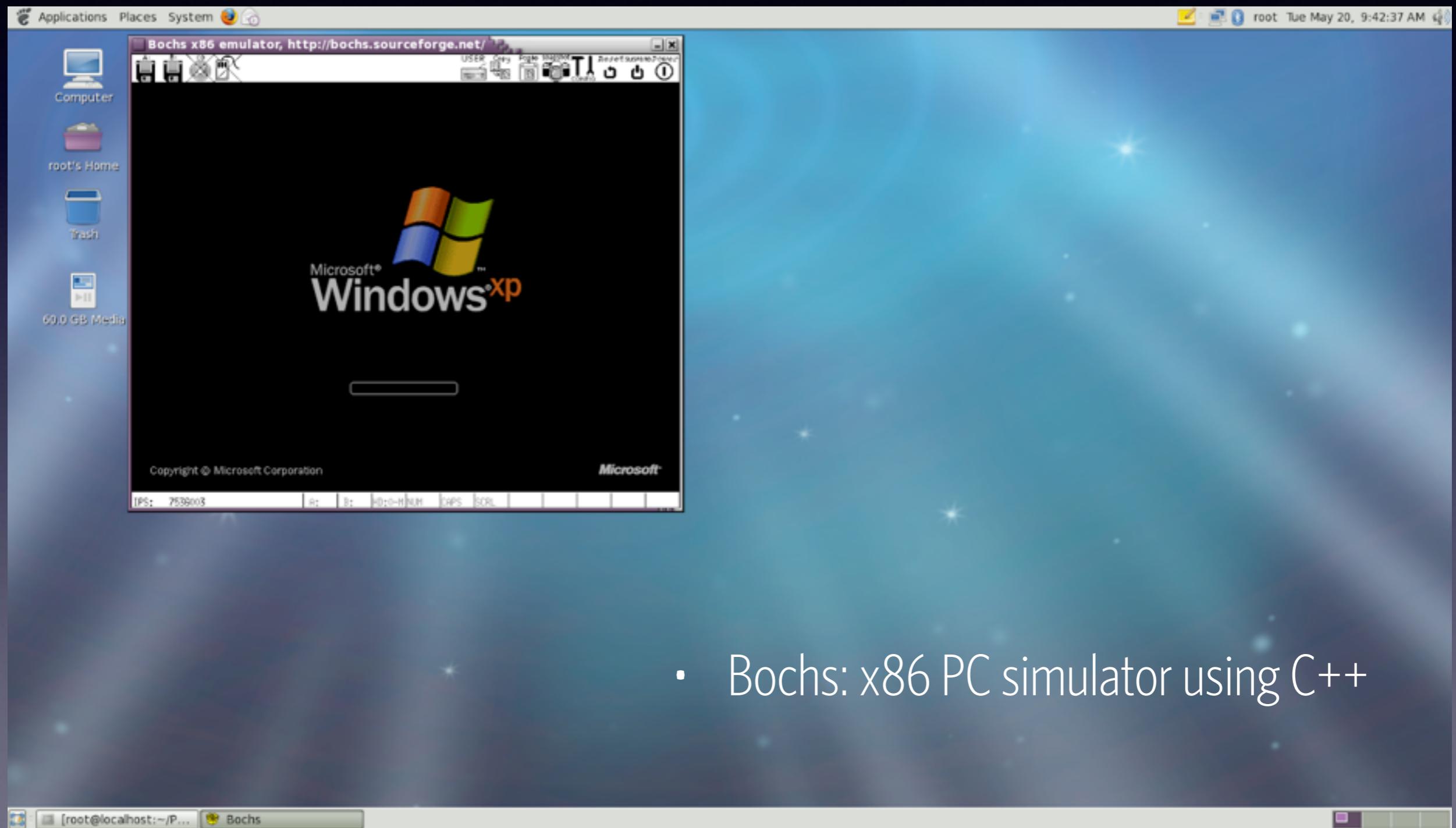
Level of Implementation	Higher Performance	Application Flexibility	Implementation Complexity	Application Isolation
ISA	X	XXXXX	XXX	XXX
Hardware-level virtualization	XXXXX	XXX	XXXXX	XXXX
OS-level virtualization	XXXXX	XX	XXX	XX
Runtime library support	XXX	XX	XX	XX
User application level	XX	XX	XXXXX	XXXXX

# Virtualization in ISA Levels



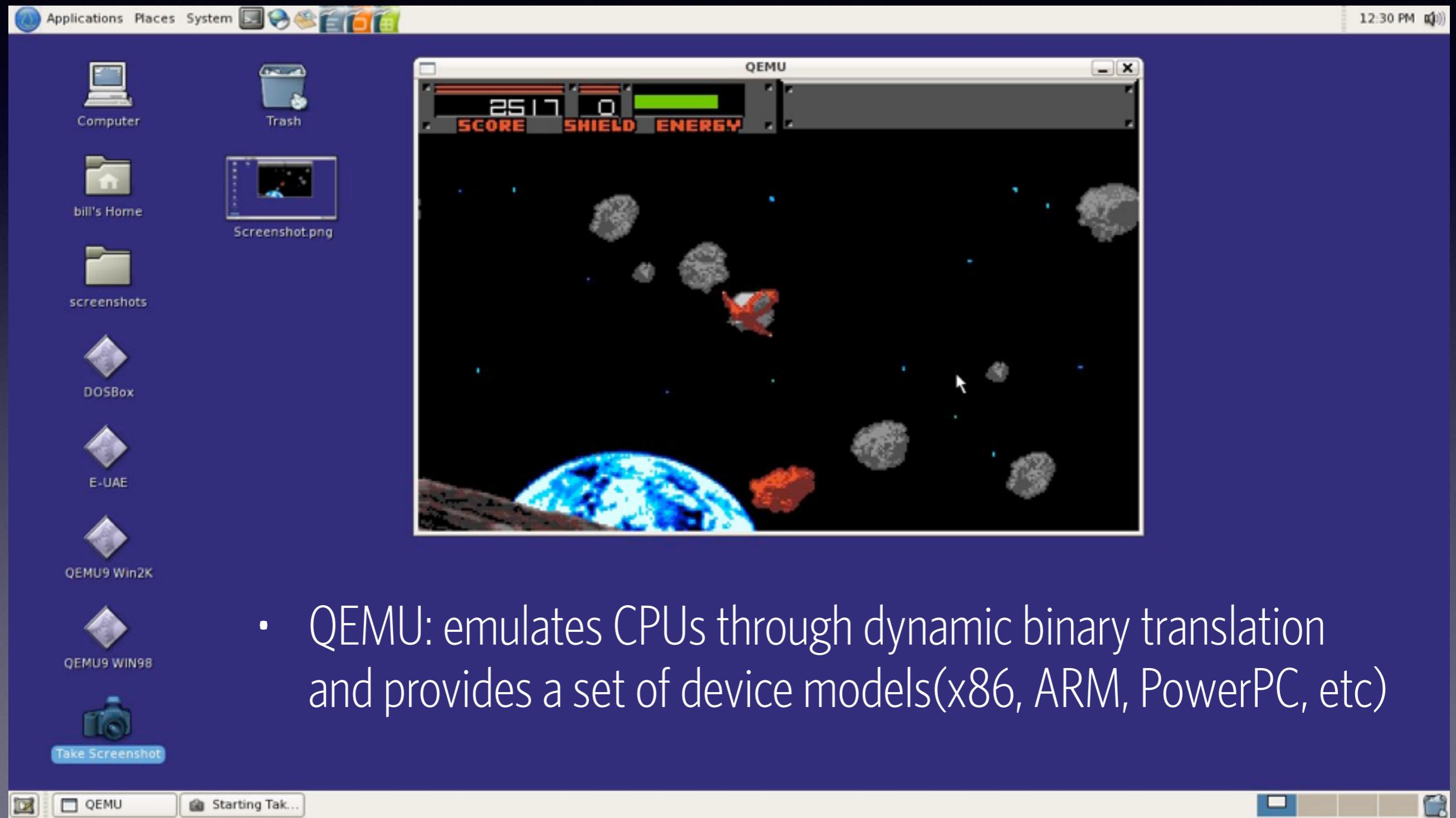
- Using software, simulate instruction set which is different from the real applications use
- Example: Bochs, QEMU

# Virtualization in ISA Levels



- Bochs: x86 PC simulator using C++

# Virtualization in ISA Levels



- QEMU: emulates CPUs through dynamic binary translation and provides a set of device models(x86, ARM, PowerPC, etc)

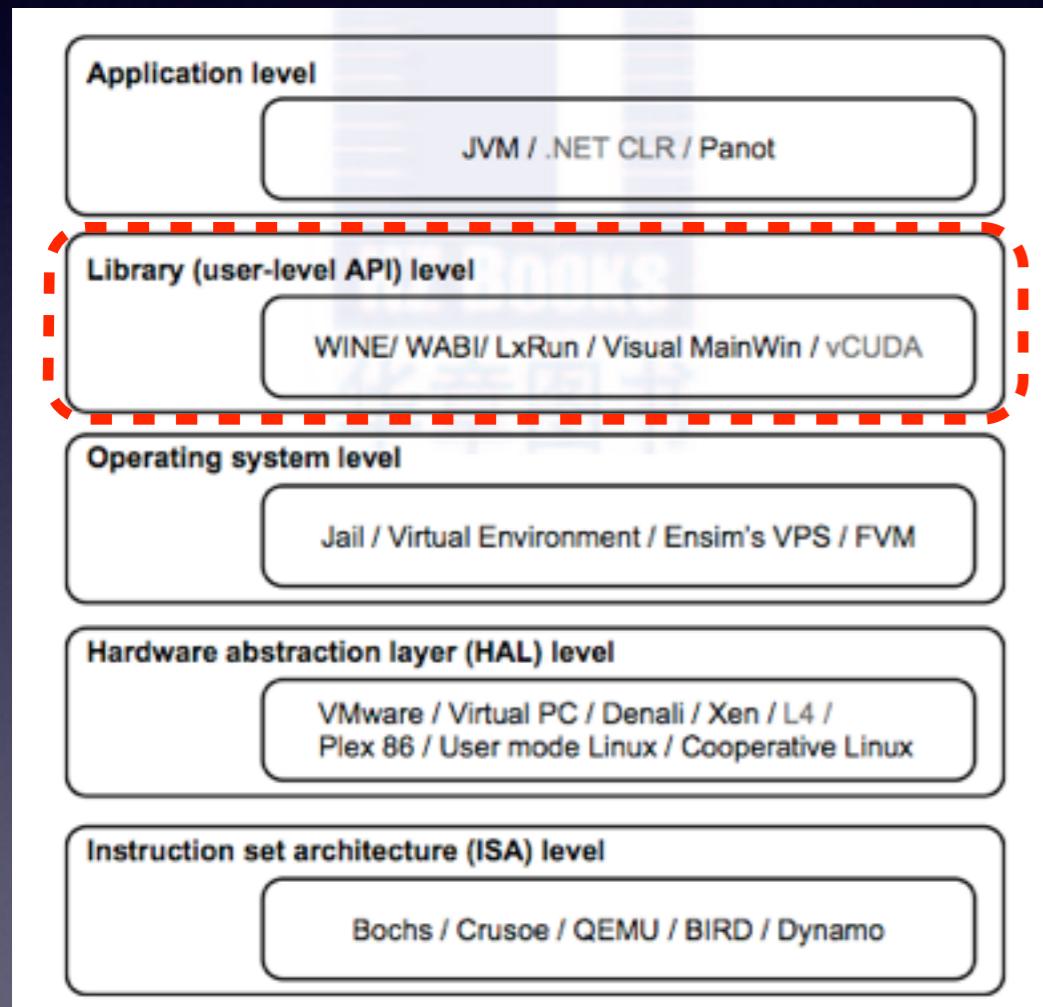
# Virtualization in ISA Levels

- Pro & Con

**Table 3.1** Relative Merits of Virtualization at Various Levels (More "X"'s Means Higher Merit, with a Maximum of 5 X's)

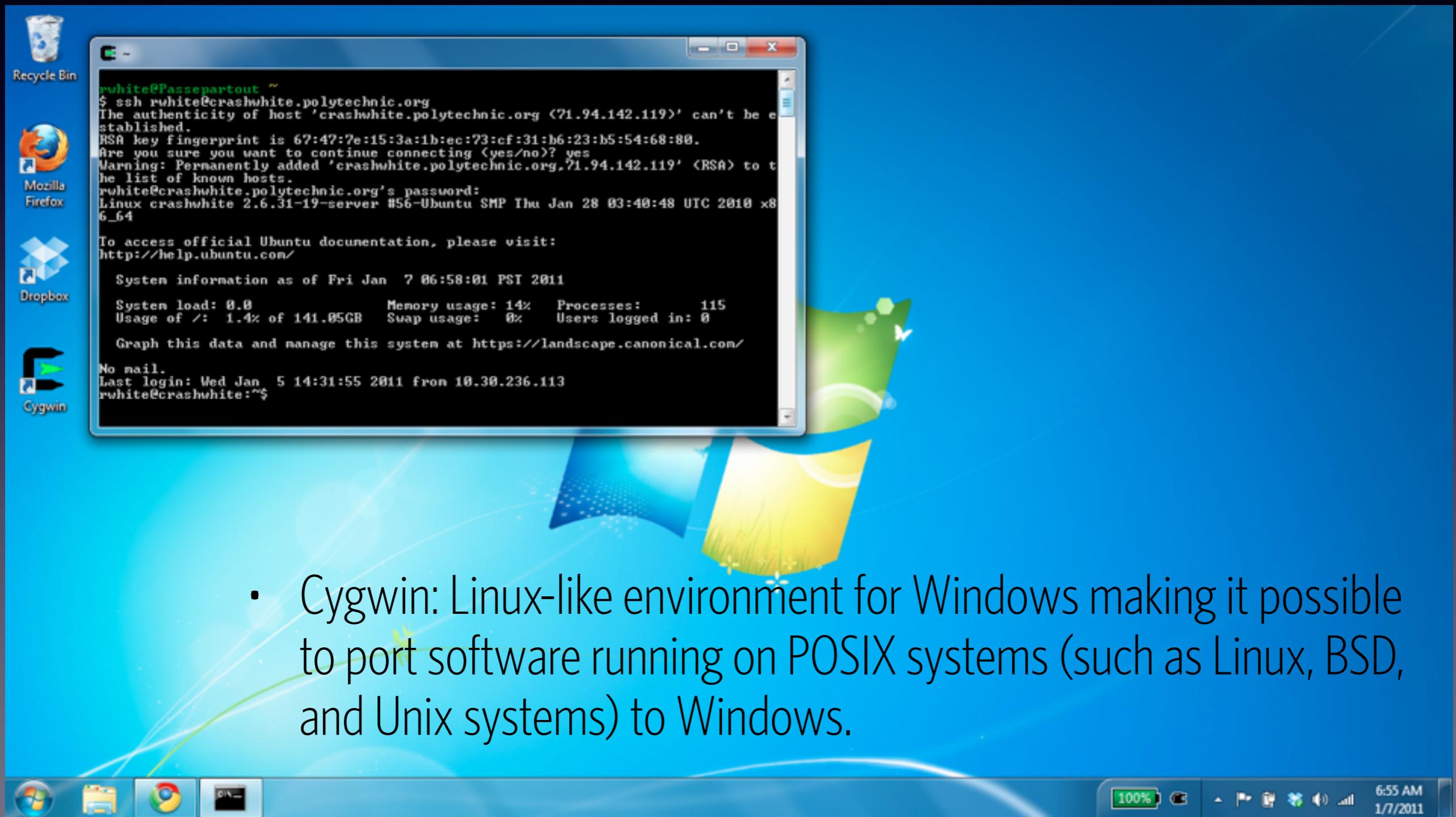
Level of Implementation	Higher Performance	Application Flexibility	Implementation Complexity	Application Isolation
ISA	X	XXXXX	XXX	XXX
Hardware-level virtualization	XXXXX	XXX	XXXXX	XXXX
OS-level virtualization	XXXXX	XX	XXX	XX
Runtime library support	XXX	XX	XX	XX
User application level	XX	XX	XXXXX	XXXXX

# Virtualization in Library Levels



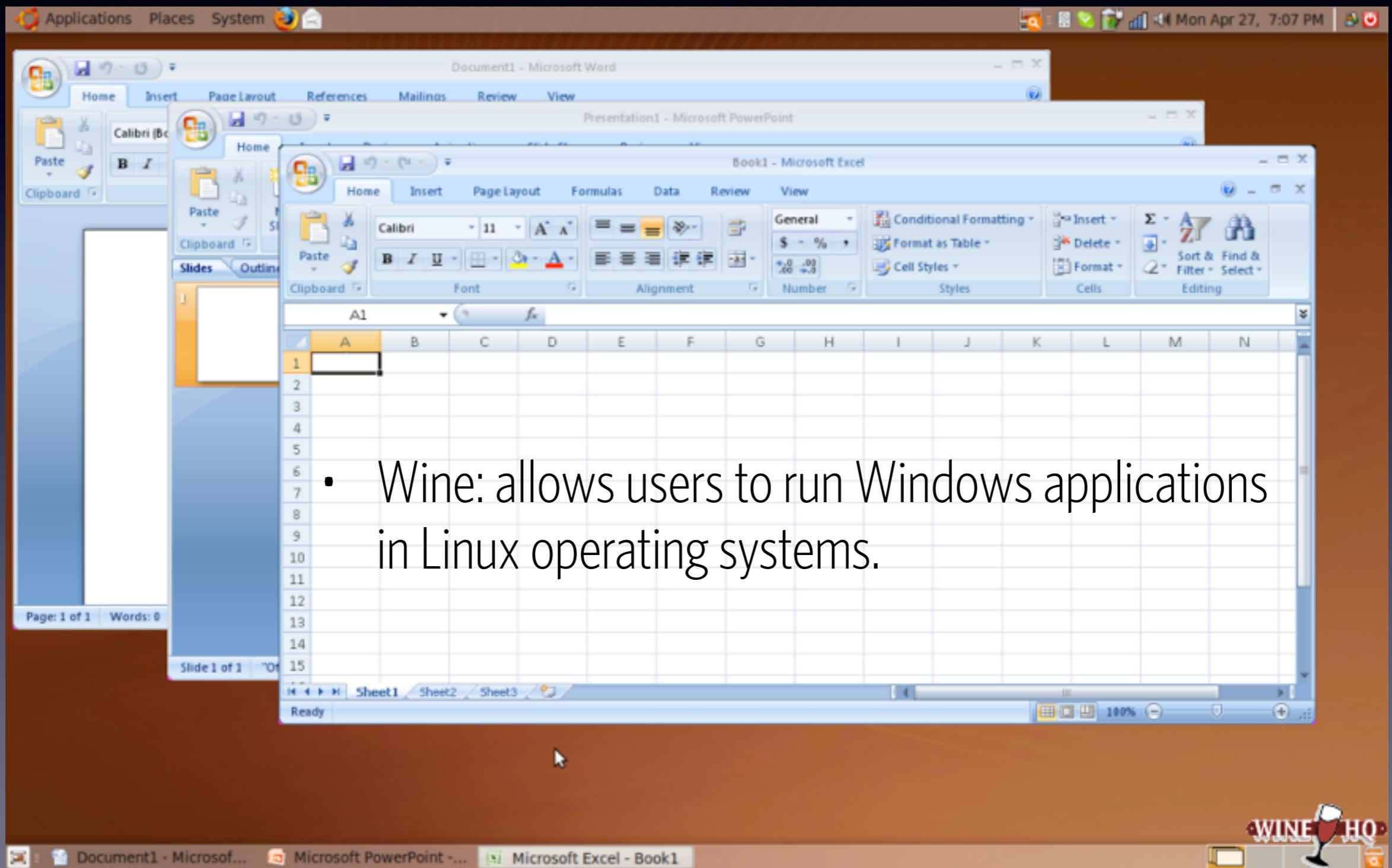
- Adding middleware to simulate library functions / APIs between applications and running libraries
- Example: Cygwin, wine

# Virtualization in Library Levels



- Cygwin: Linux-like environment for Windows making it possible to port software running on POSIX systems (such as Linux, BSD, and Unix systems) to Windows.

# Virtualization in Library Levels



- Wine: allows users to run Windows applications in Linux operating systems.

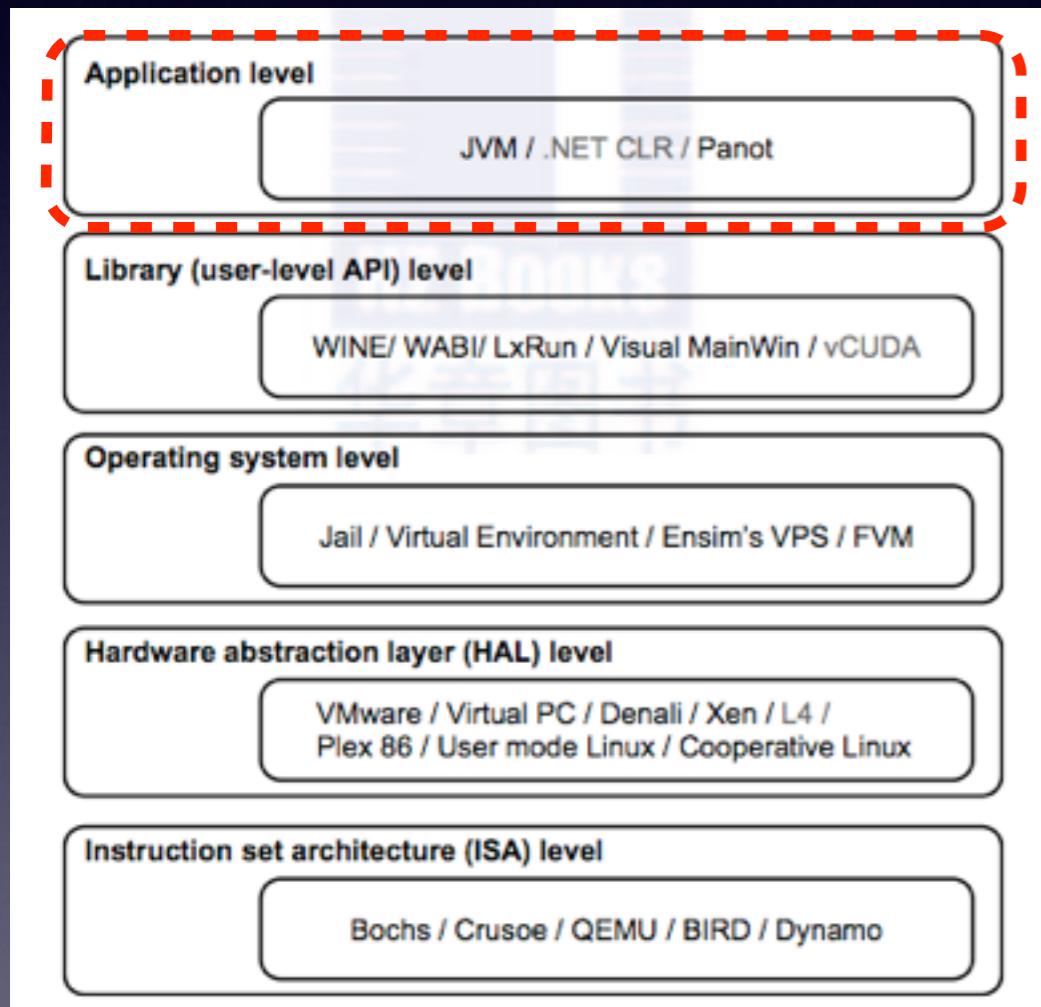
# Virtualization in Library Levels

- Pro & Con

**Table 3.1** Relative Merits of Virtualization at Various Levels (More "X"'s Means Higher Merit, with a Maximum of 5 X's)

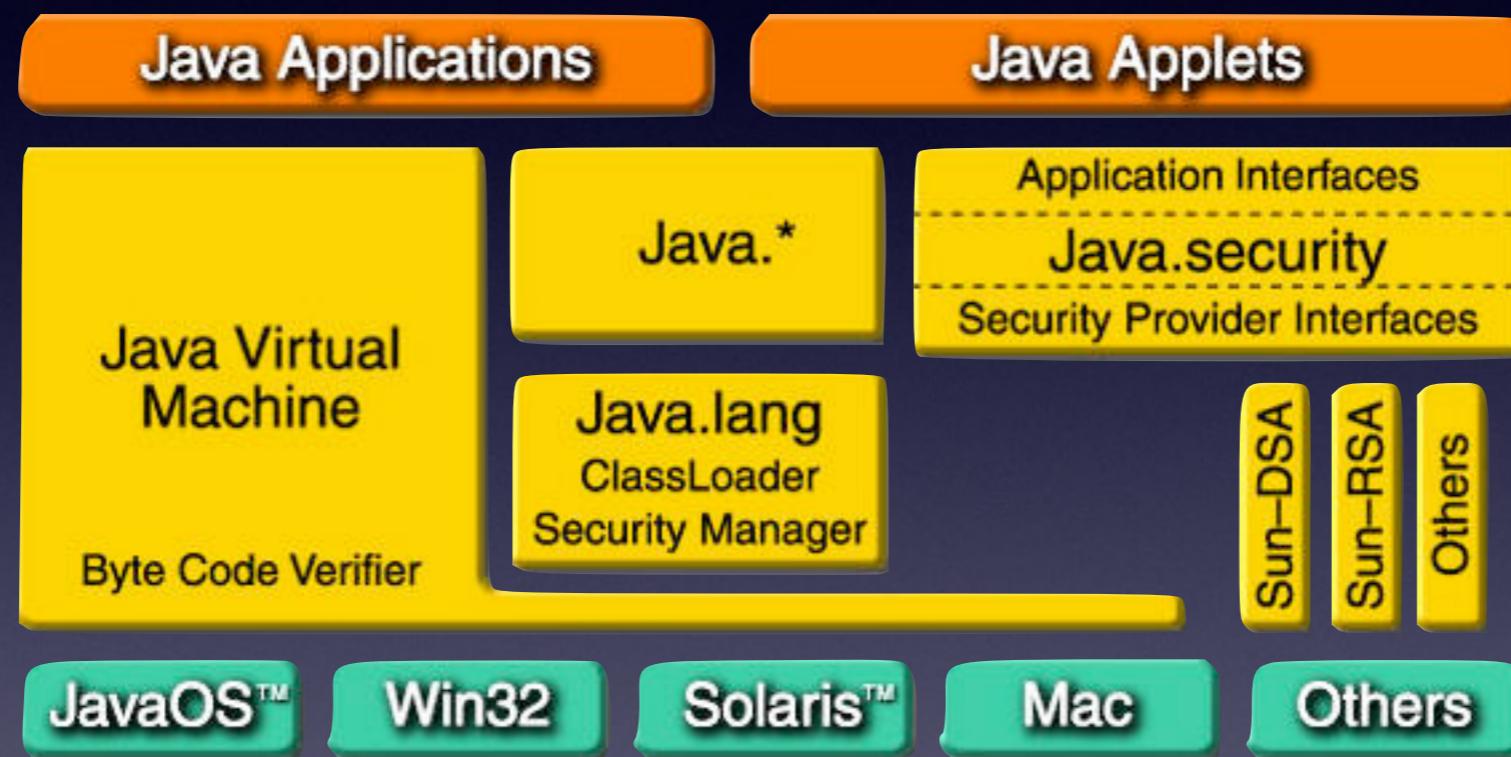
Level of Implementation	Higher Performance	Application Flexibility	Implementation Complexity	Application Isolation
ISA	X	XXXXX	XXX	XXX
Hardware-level virtualization	XXXXX	XXX	XXXXX	XXXX
OS-level virtualization	XXXXX	XX	XXX	XX
Runtime library support	XXX	XX	XX	XX
User application level	XX	XX	XXXXX	XXXXX

# Virtualization in App Levels



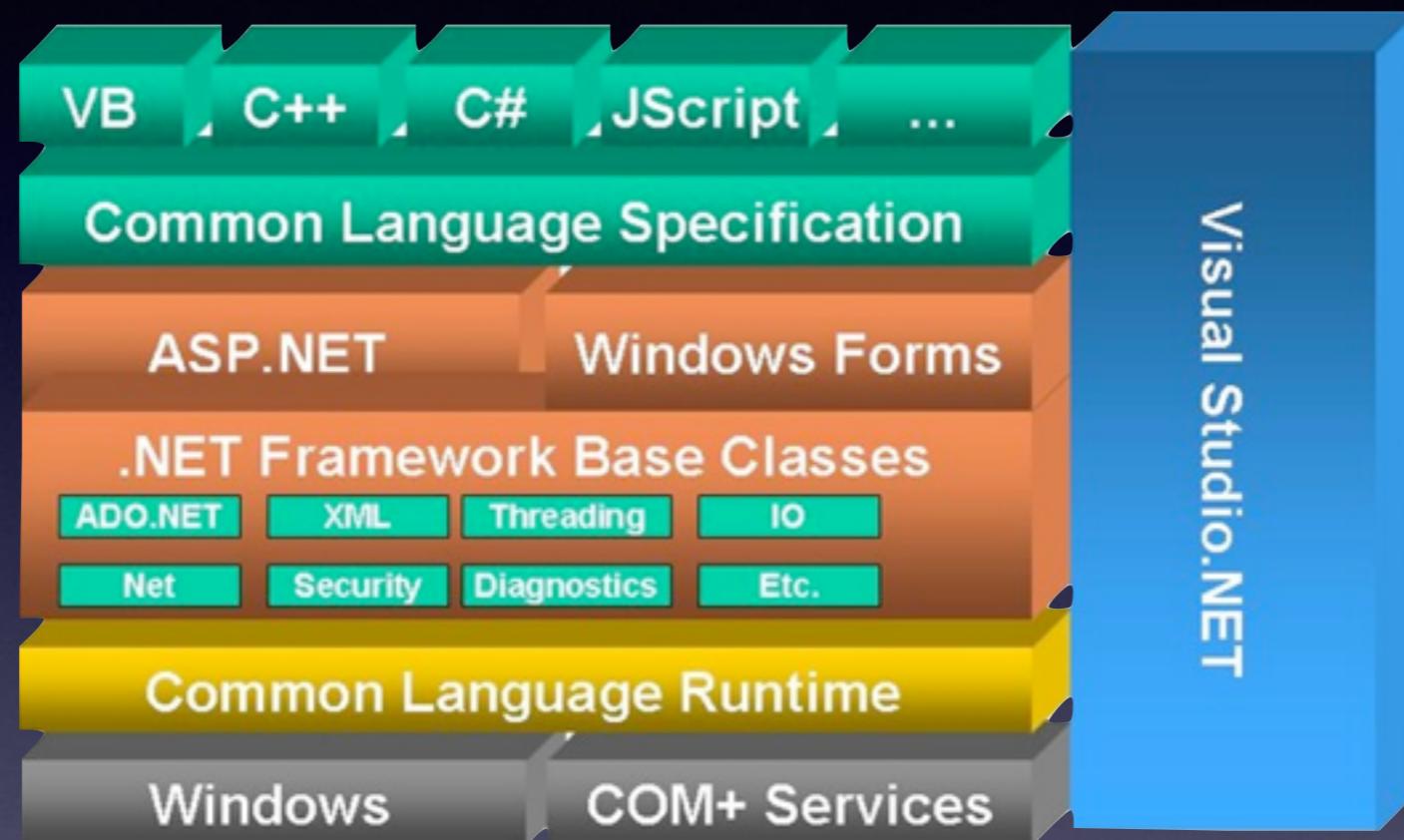
- Building a VM on the application level and support a new self-defined instruction set
- Example: JVM, .NET

# Virtualization in App Levels



- Java: based on JVM, write once, run anywhere

# Virtualization in App Levels



- .NET: provides language interoperability (each language can use code written in other languages) across several programming languages based on an application virtual machine called Common Language Runtime

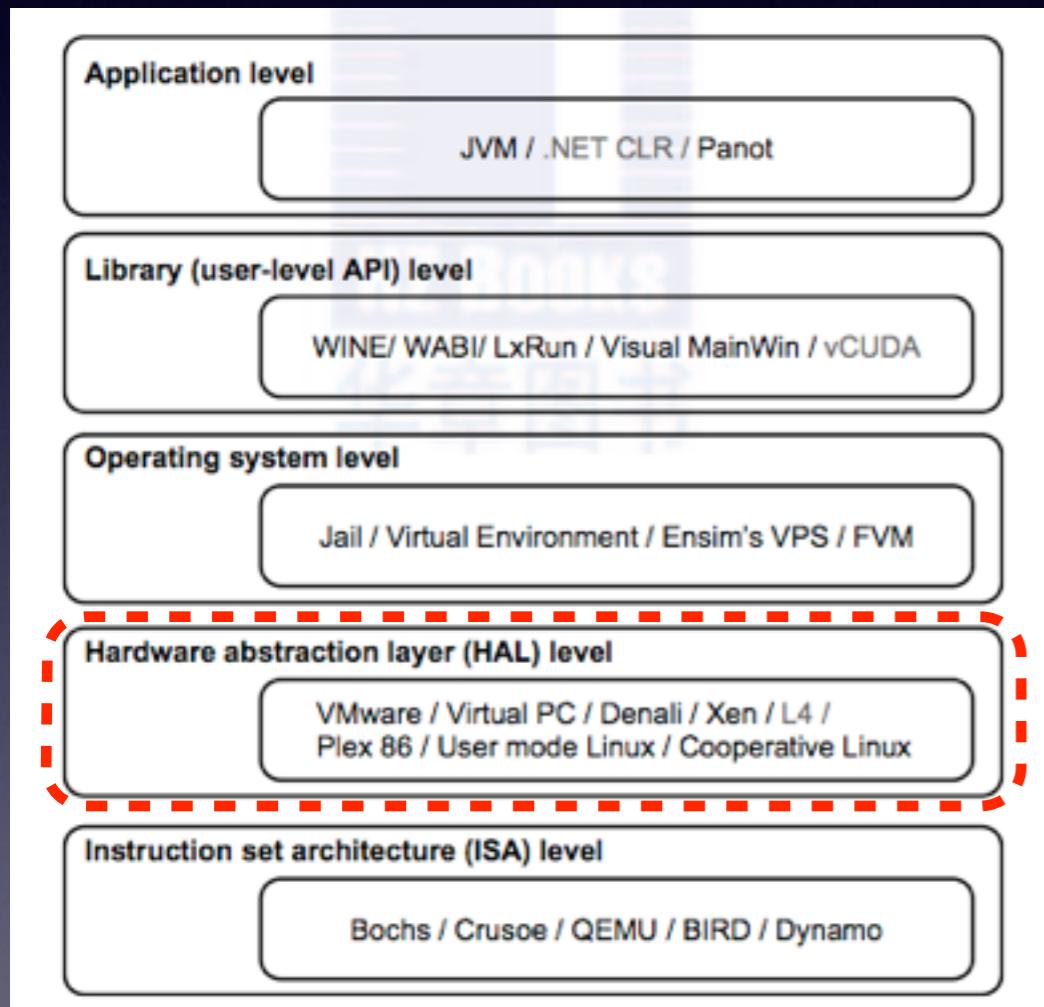
# Virtualization in App Levels

- Pro & Con

**Table 3.1** Relative Merits of Virtualization at Various Levels (More "X"'s Means Higher Merit, with a Maximum of 5 X's)

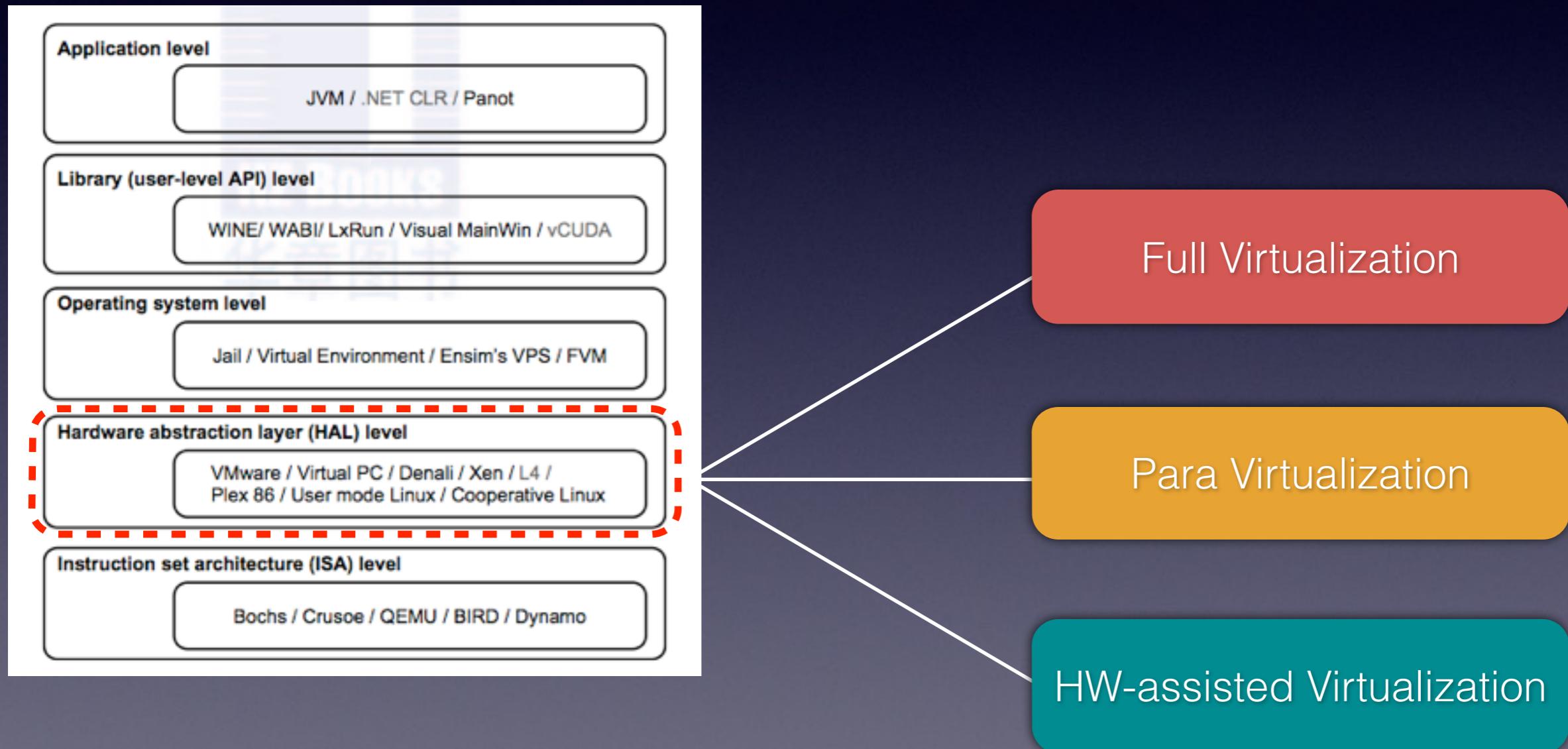
Level of Implementation	Higher Performance	Application Flexibility	Implementation Complexity	Application Isolation
ISA	X	XXXXX	XXX	XXX
Hardware-level virtualization	XXXXX	XXX	XXXXX	XXXX
OS-level virtualization	XXXXX	XX	XXX	XX
Runtime library support	XXX	XX	XX	XX
User application level	XX	XX	XXXXX	XXXXX

# Virtualization in Hardware Levels



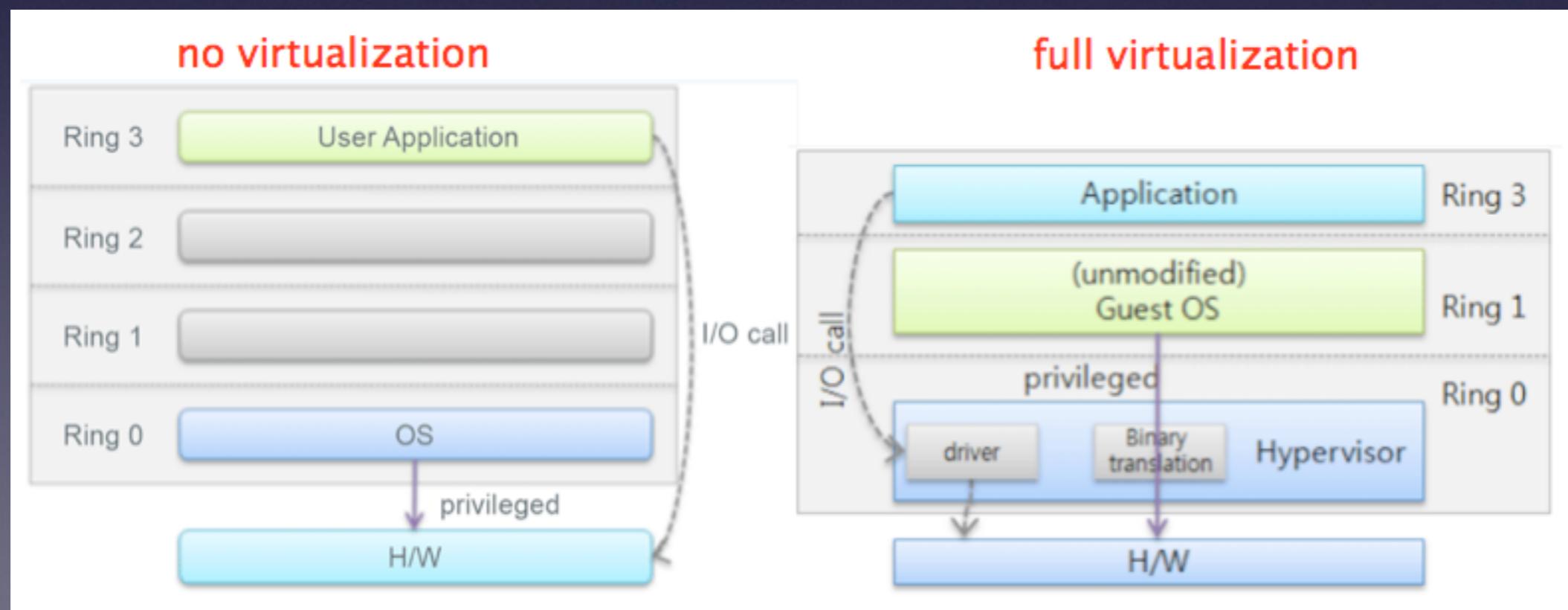
- Mapping virtual resources to physical resources and using local hardware in the virtual machine
- Example: Xen, VMware, KVM, Virtual PC
- The most widely used virtualization technology

# Virtualization in Hardware Levels



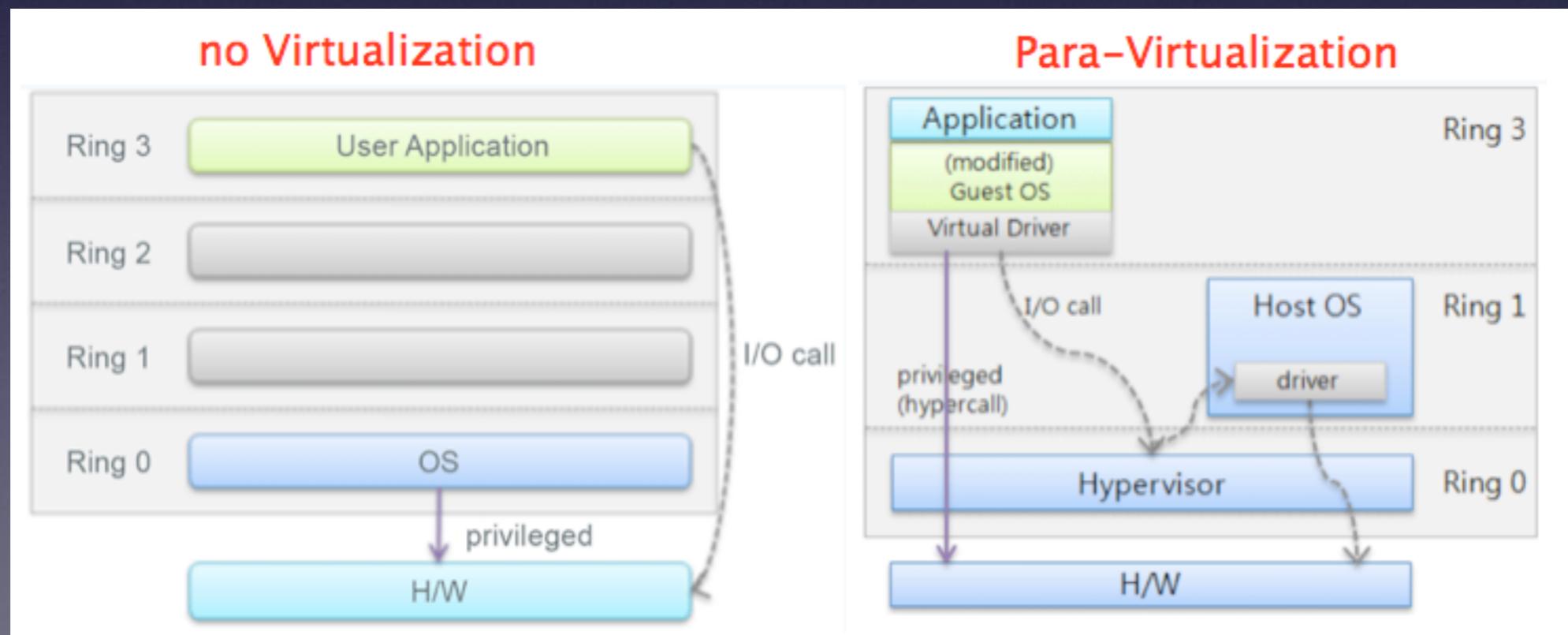
# Virtualization in Hardware Levels

- Full-virtualization: hypervisor simulate hardware resources and fully hardware services(like virtual device, virtual memory management) to virtual machines without assist from hardware and OS



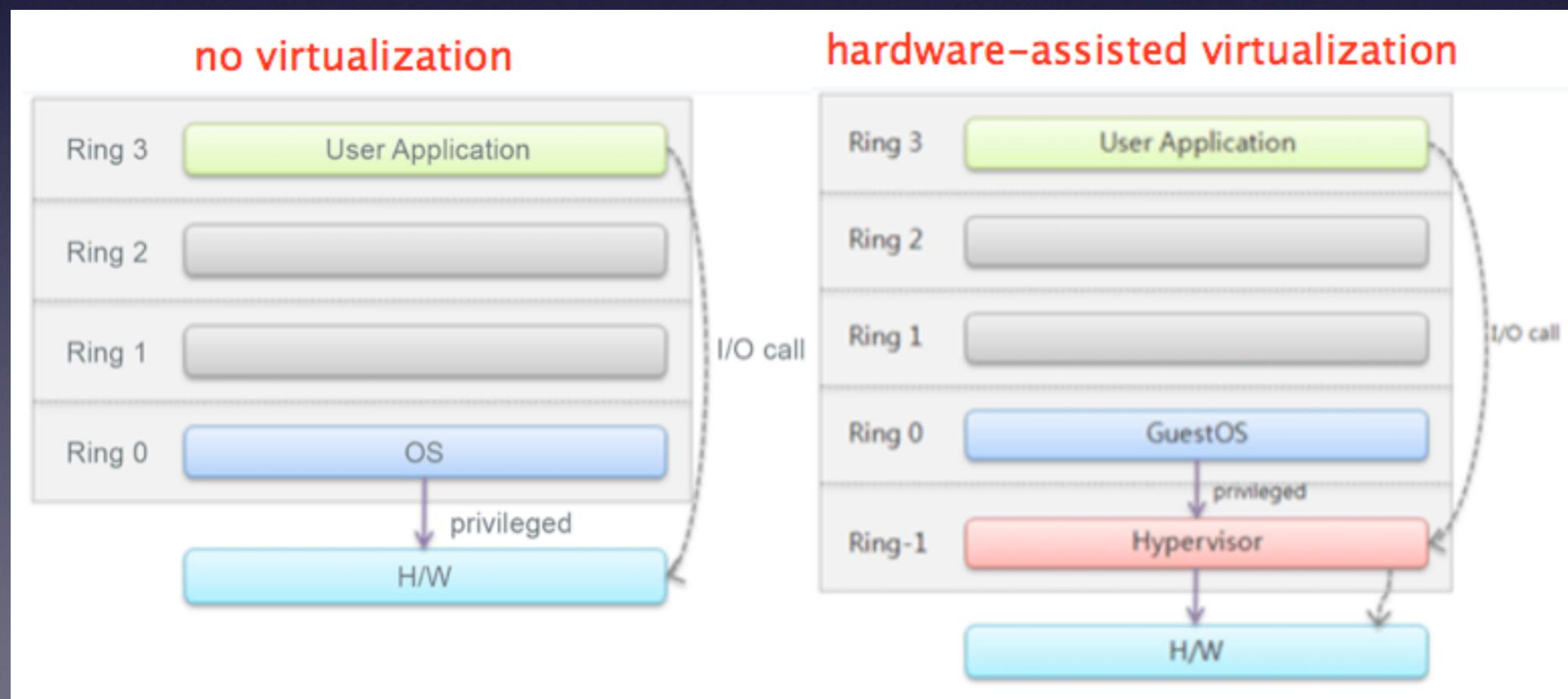
# Virtualization in Hardware Levels

- Para-virtualization: hypervisor needs assists from OS to simulate x86 privilege instructions; OS needs to be modified; also called OS-assisted virtualization



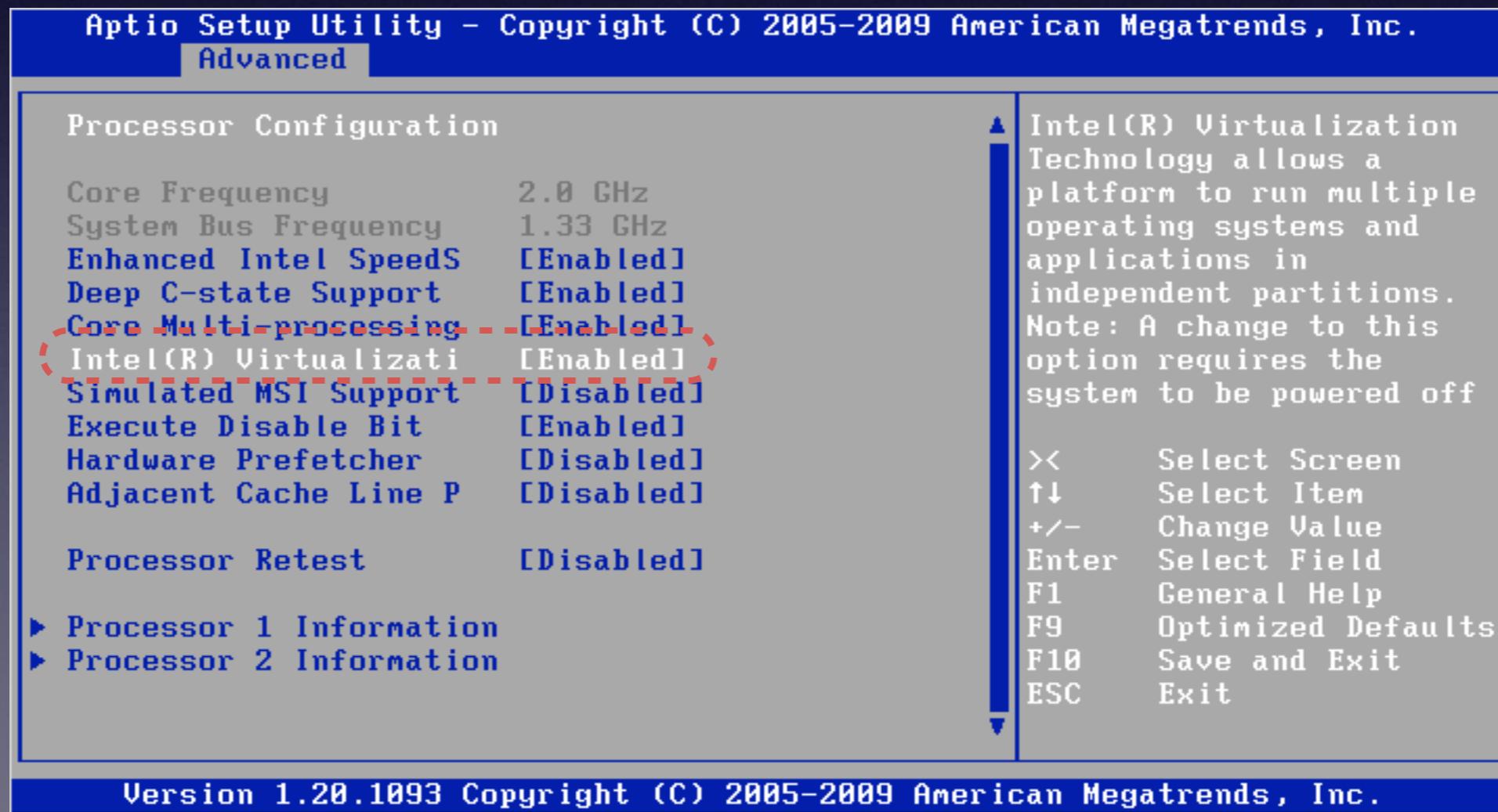
# Virtualization in Hardware Levels

- Hardware-assisted virtualization: hypervisor needs assists from hardware to simulate hardware resources; e.g.: Intel-VT, AMD-V

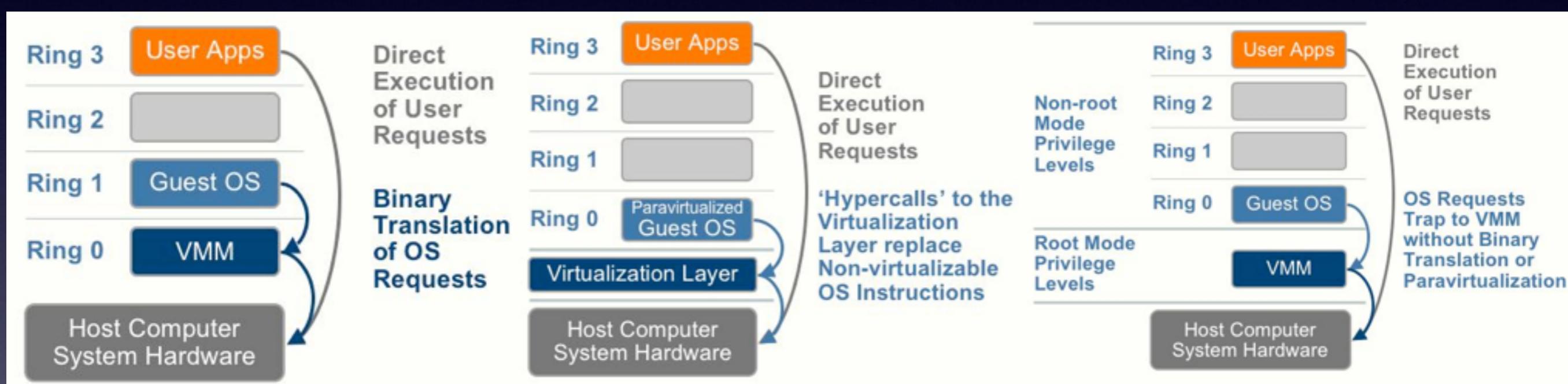


# Virtualization in Hardware Levels

- Hardware-assisted virtualization: better performance than the dynamic binary translation today



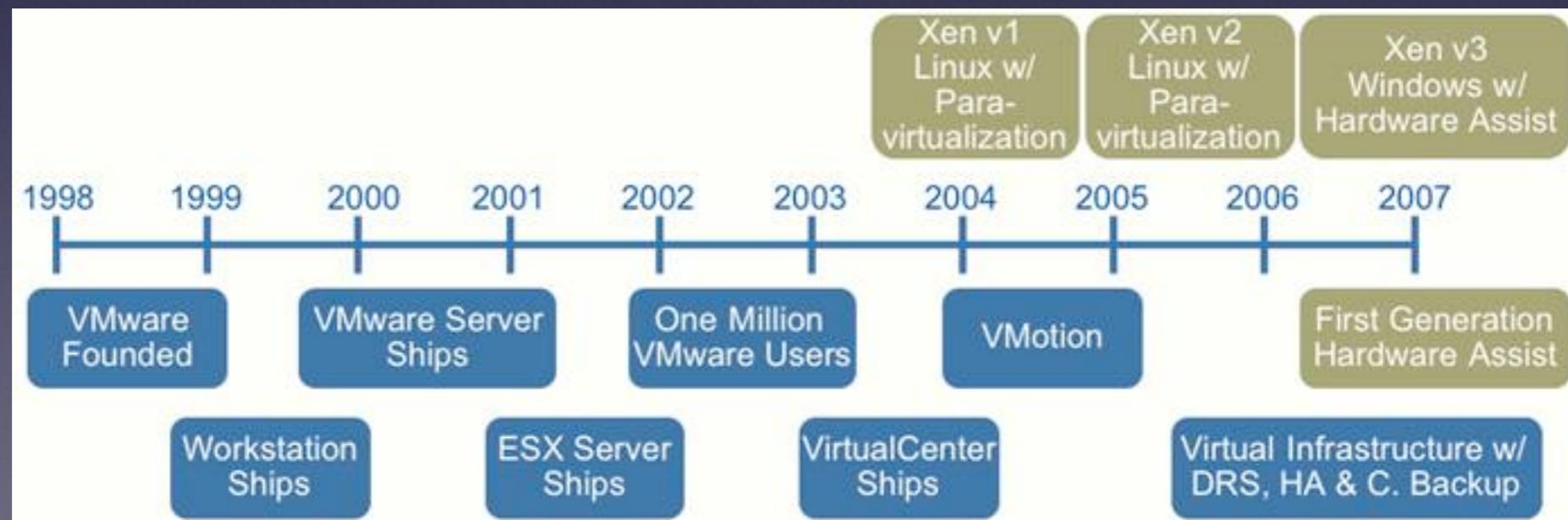
# Virtualization in Hardware Levels



- Full virtualization
- OS-assisted virtualization
- Hardware-assisted virtualization

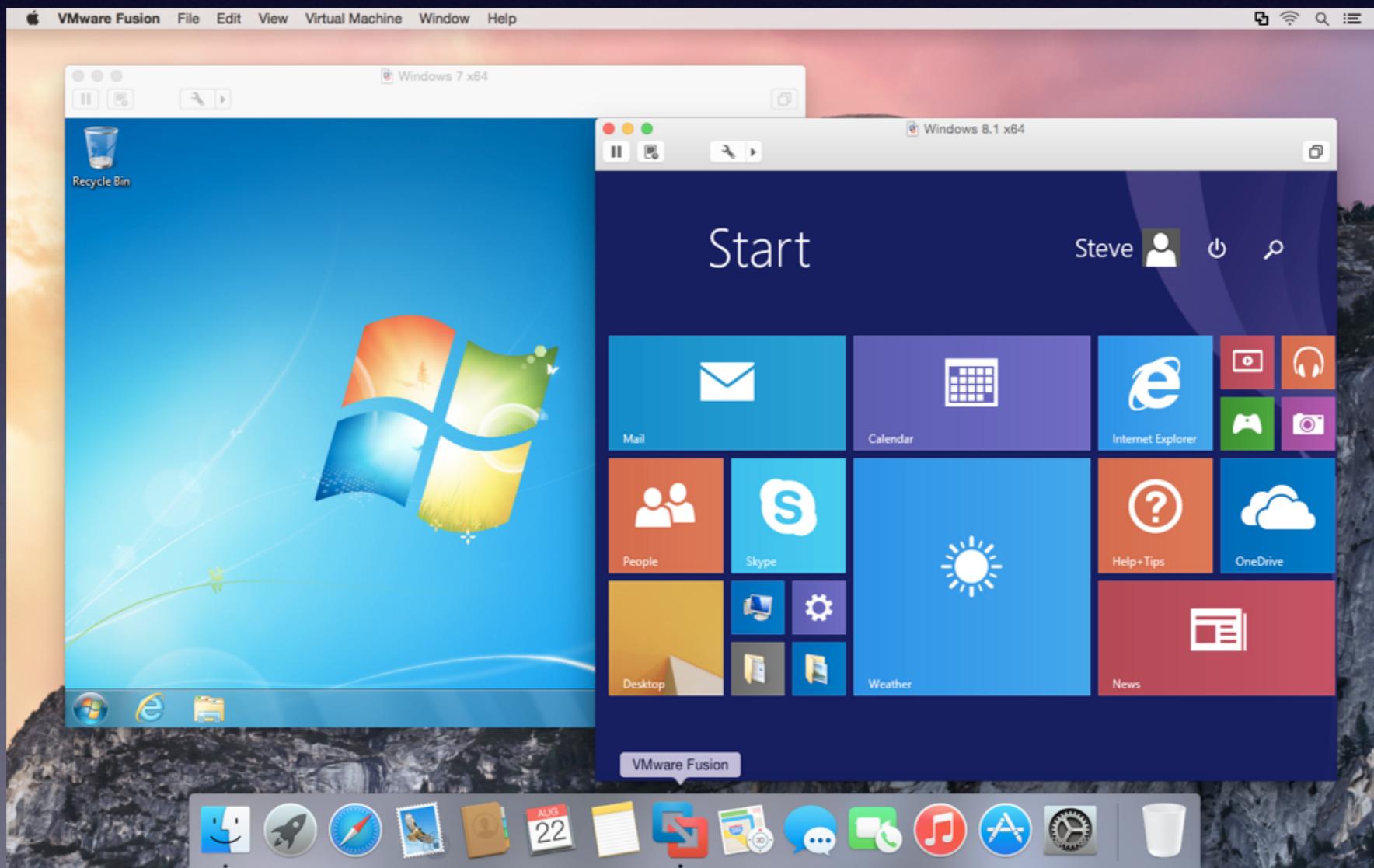
# Virtualization in Hardware Levels

- Full virtualization
- OS-assisted virtualization
- Hardware-assisted virtualization



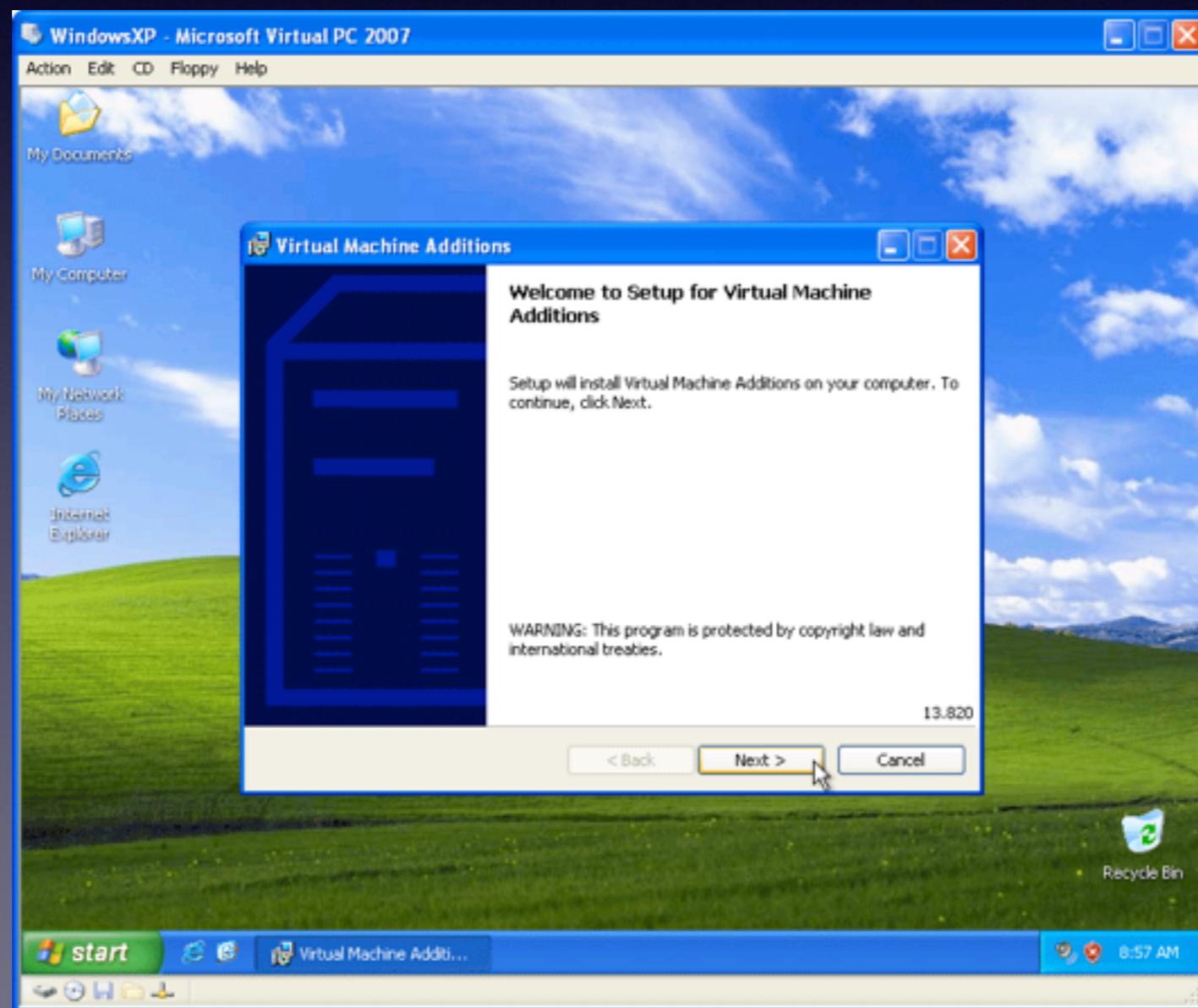
# Virtualization in Hardware Levels

- VMware Workstation: full virtualization



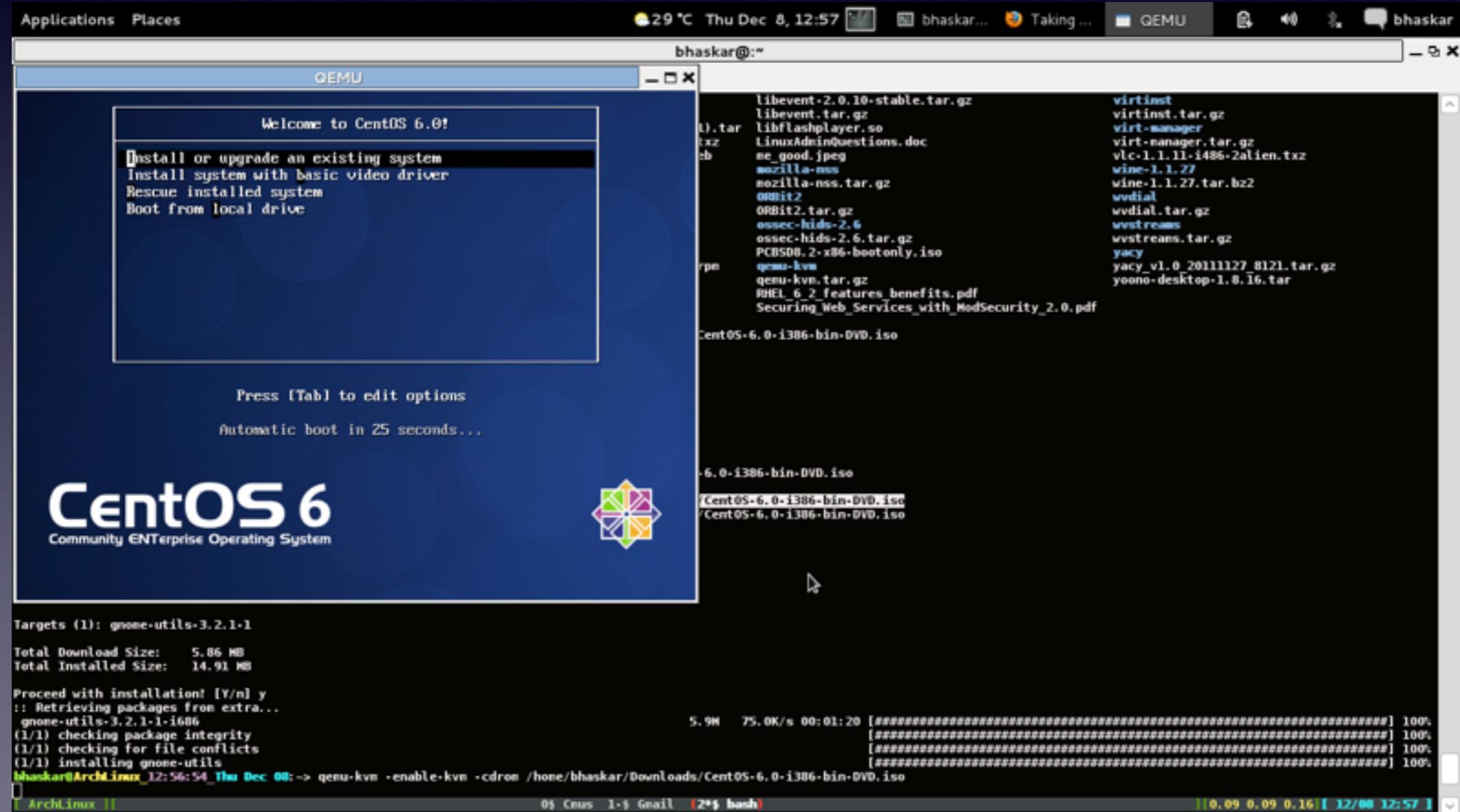
# Virtualization in Hardware Levels

- Virtual PC: full virtualization



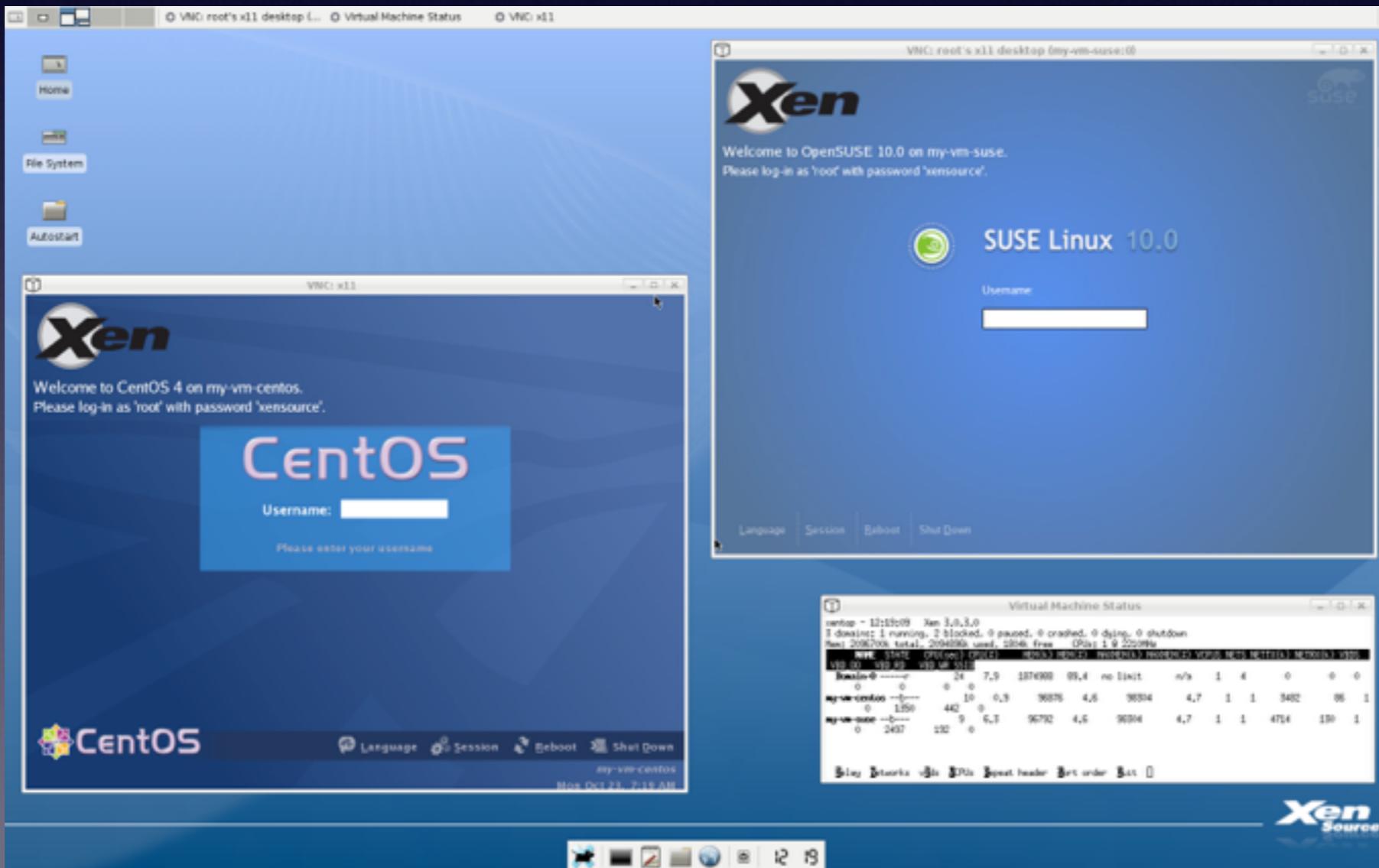
# Virtualization in Hardware Levels

- KVM: full virtualization



# Virtualization in Hardware Levels

- Xen: para virtualization



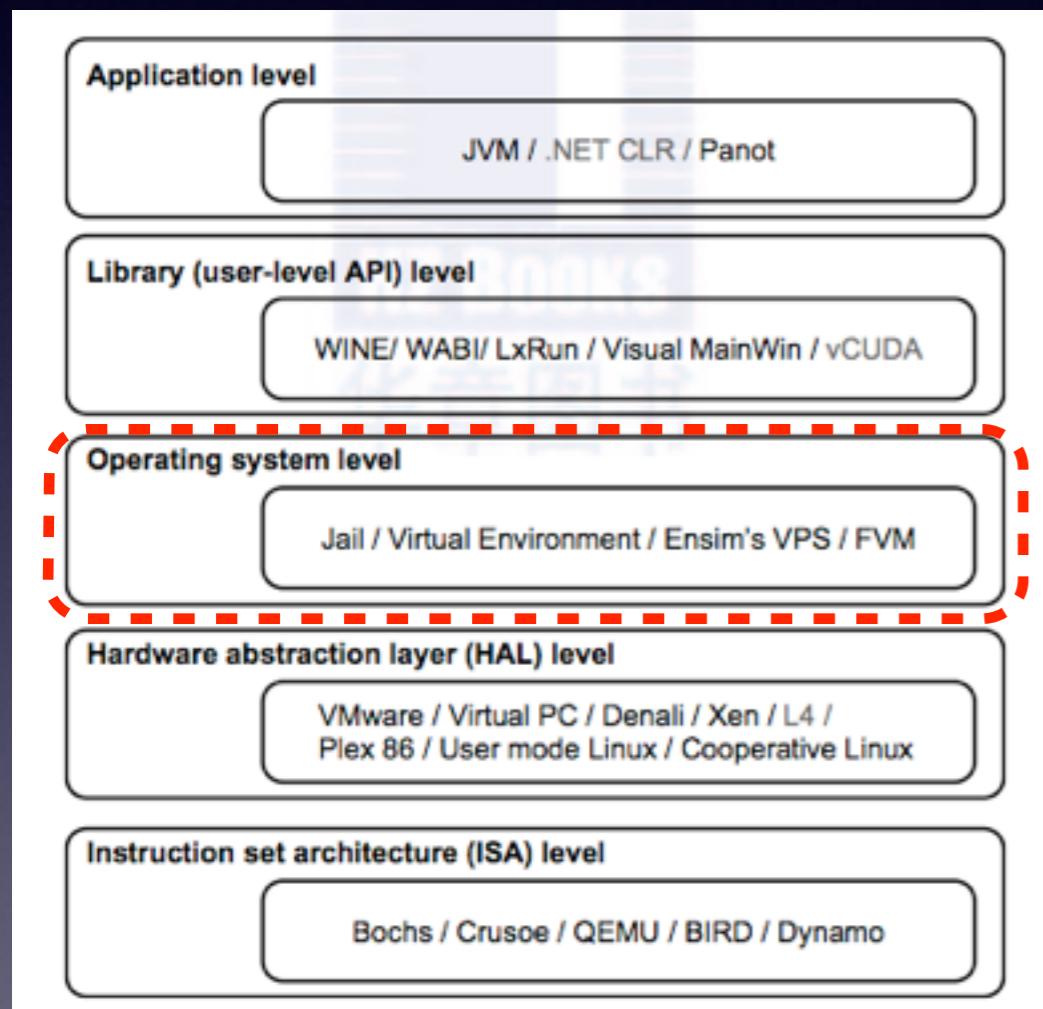
# Virtualization in Hardware Levels

- Pro & Con

**Table 3.1** Relative Merits of Virtualization at Various Levels (More "X"'s Means Higher Merit, with a Maximum of 5 X's)

Level of Implementation	Higher Performance	Application Flexibility	Implementation Complexity	Application Isolation
ISA	X	XXXXX	XXX	XXX
Hardware-level virtualization	XXXXX	XXX	XXXXX	XXXX
OS-level virtualization	XXXXX	XX	XXX	XX
Runtime library support	XXX	XX	XX	XX
User application level	XX	XX	XXXXX	XXXXX

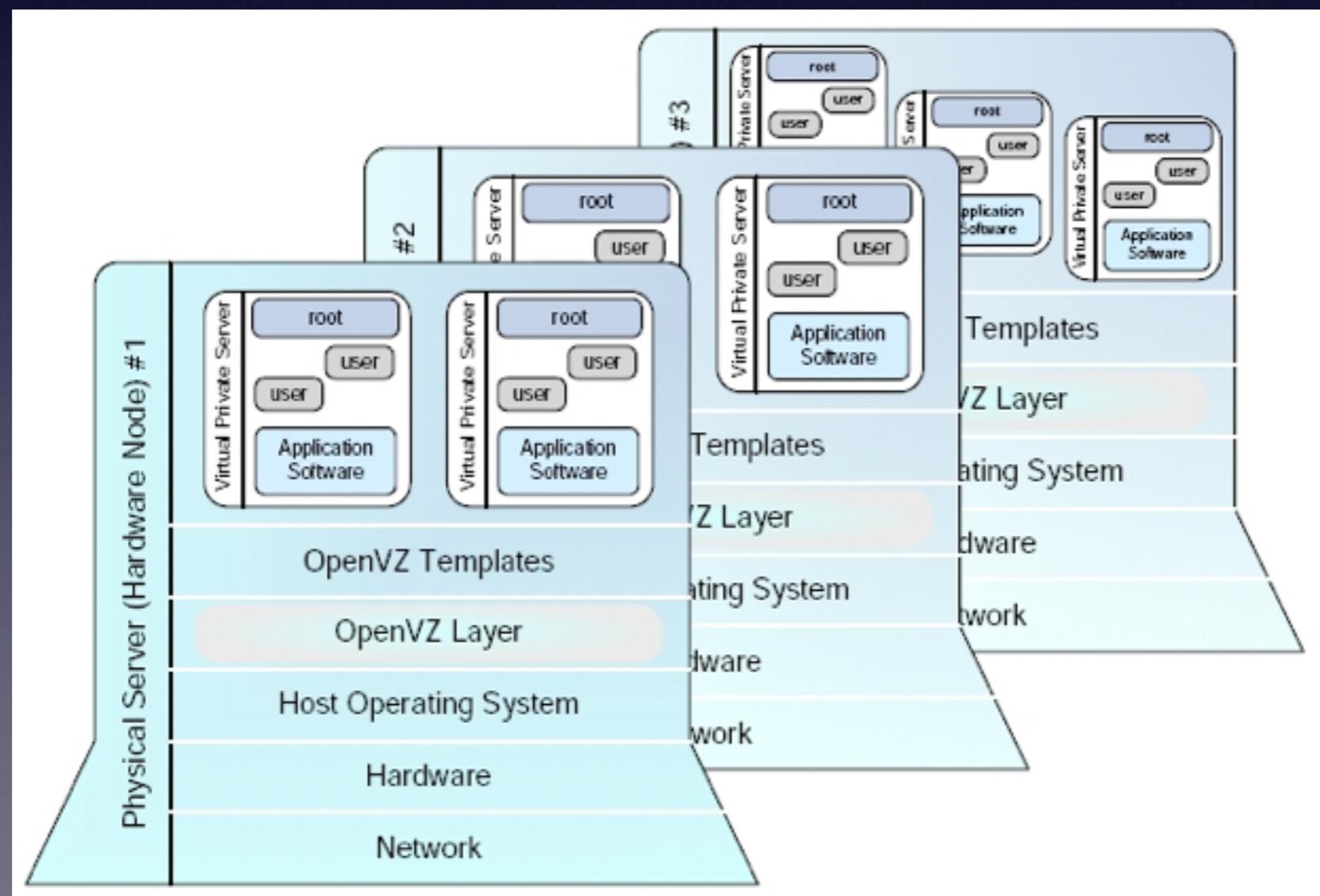
# Virtualization in OS Levels



- The operating system is virtualized into multiple isolated partitions. The OS kernel will run a single operating system and provide that operating system functionality to each of the partitions.
- Example: OpenVZ, Linux container

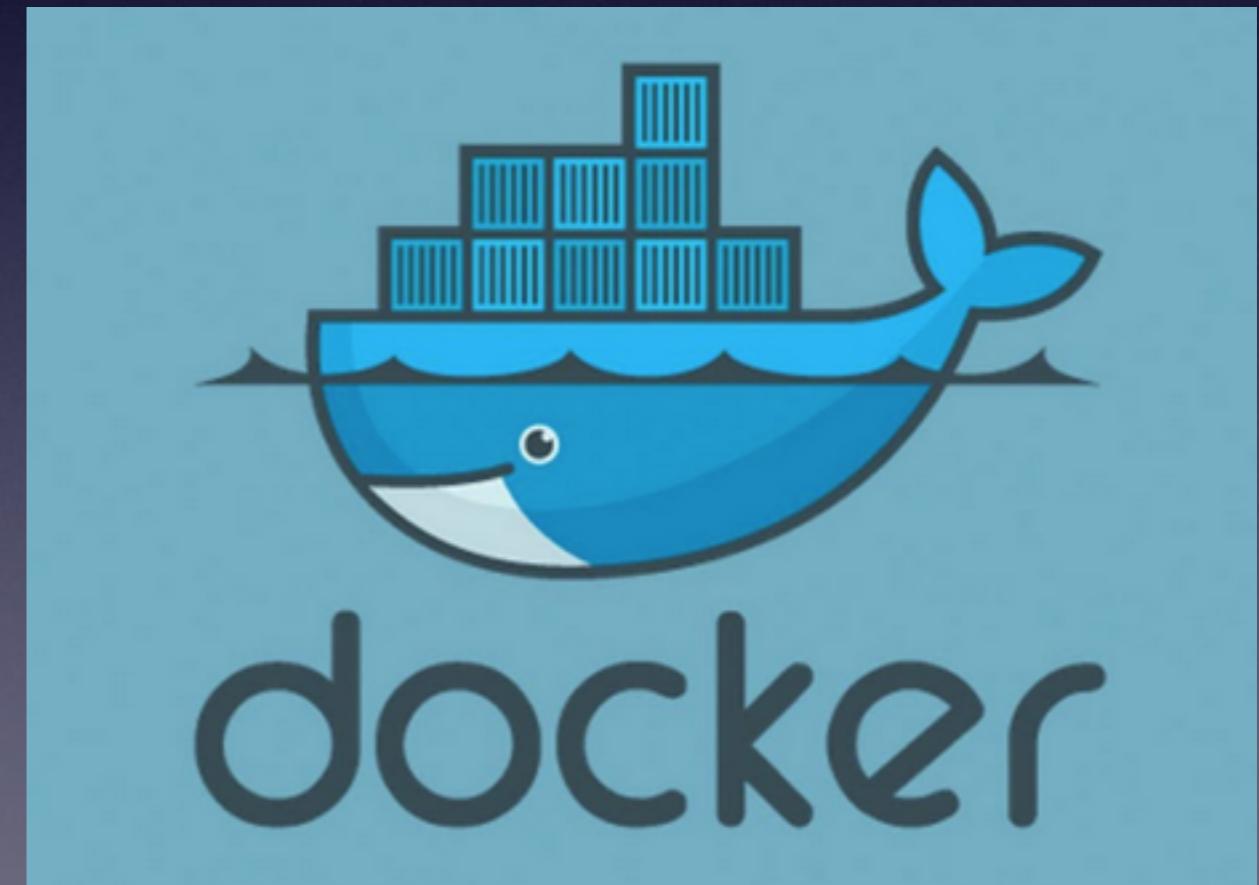
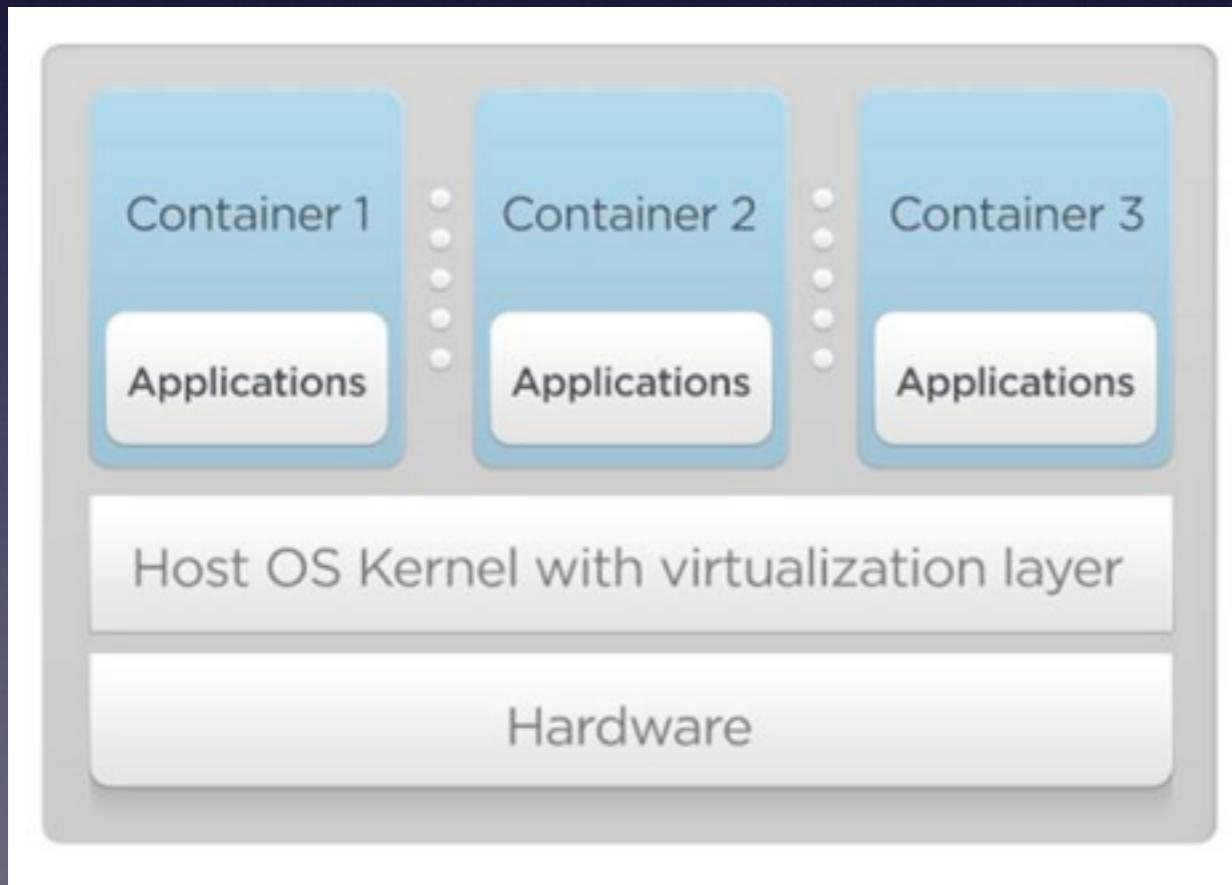
# Virtualization in OS Levels

- OpenVZ: creates multiple secure, isolated Linux containers on a single physical server



# Virtualization in OS Levels

- Linux container: runs multiple isolated Linux systems (containers) on a single Linux control host.



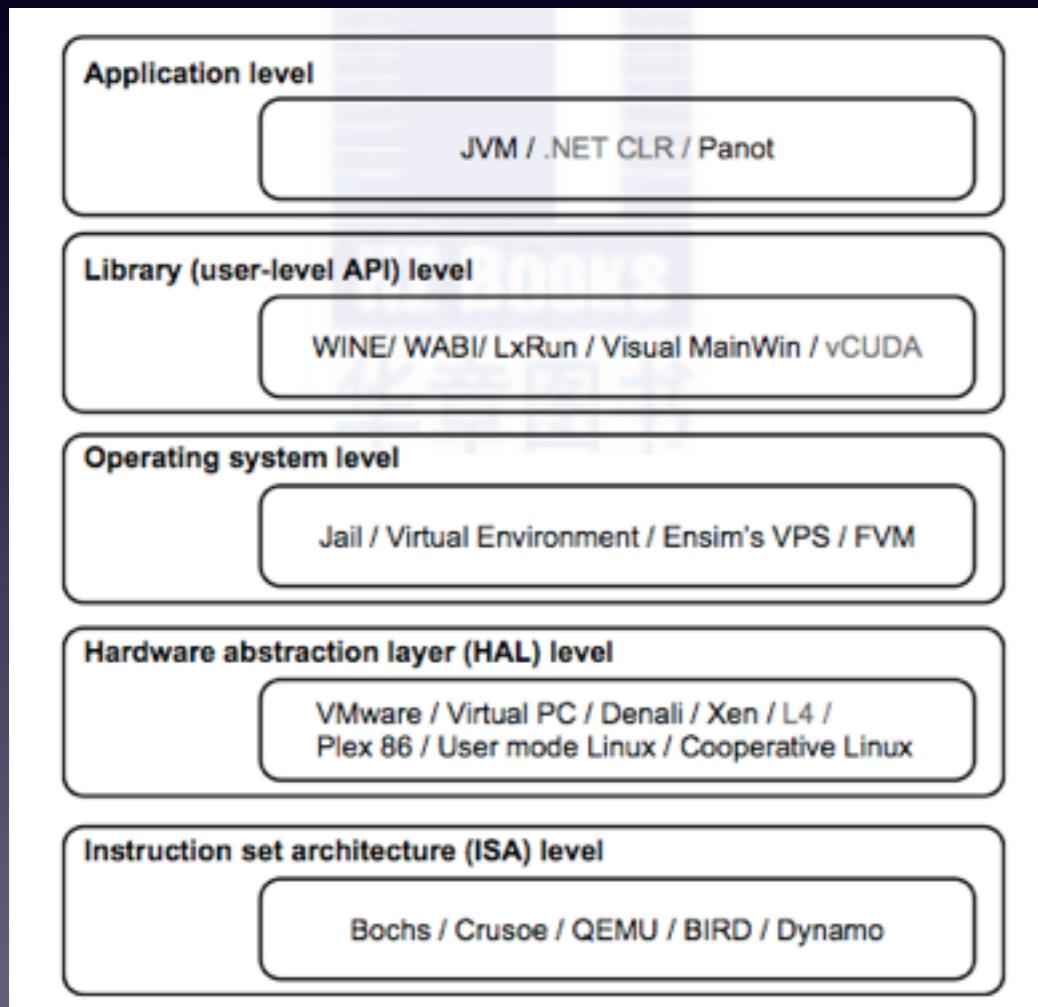
# Virtualization in OS Levels

- Pro & Con

**Table 3.1** Relative Merits of Virtualization at Various Levels (More "X"'s Means Higher Merit, with a Maximum of 5 X's)

Level of Implementation	Higher Performance	Application Flexibility	Implementation Complexity	Application Isolation
ISA	X	XXXXX	XXX	XXX
Hardware-level virtualization	XXXXX	XXX	XXXXX	XXXX
OS-level virtualization	XXXXX	XX	XXX	XX
Runtime library support	XXX	XX	XX	XX
User application level	XX	XX	XXXXX	XXXXX

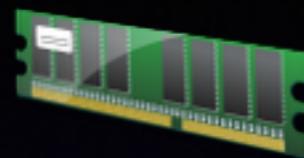
# Virtualization in Different Levels



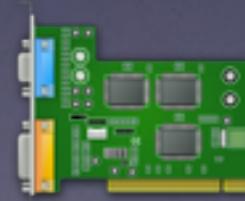
**Table 3.1** Relative Merits of Virtualization at Various Levels (More "X"'s Means Higher Merit, with a Maximum of 5 X's)

Level of Implementation	Higher Performance	Application Flexibility	Implementation Complexity	Application Isolation
ISA	X	XXXXX	XXX	XXX
Hardware-level virtualization	XXXXX	XXX	XXXXX	XXXX
OS-level virtualization	XXXXX	XX	XXX	XX
Runtime library support	XXX	XX	XX	XX
User application level	XX	XX	XXXXX	XXXXX

# Virtualization in Different Resources Levels

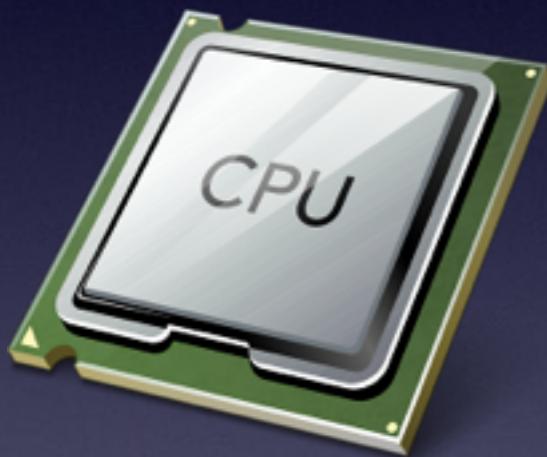


What we can see...

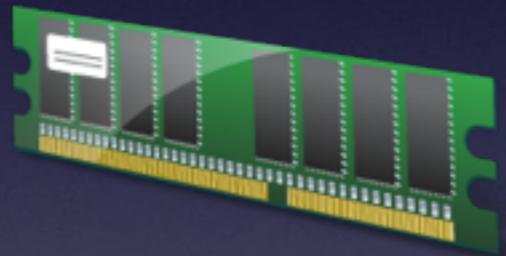


What we cannot see...

# Virtualization in Resources Level



- CPU



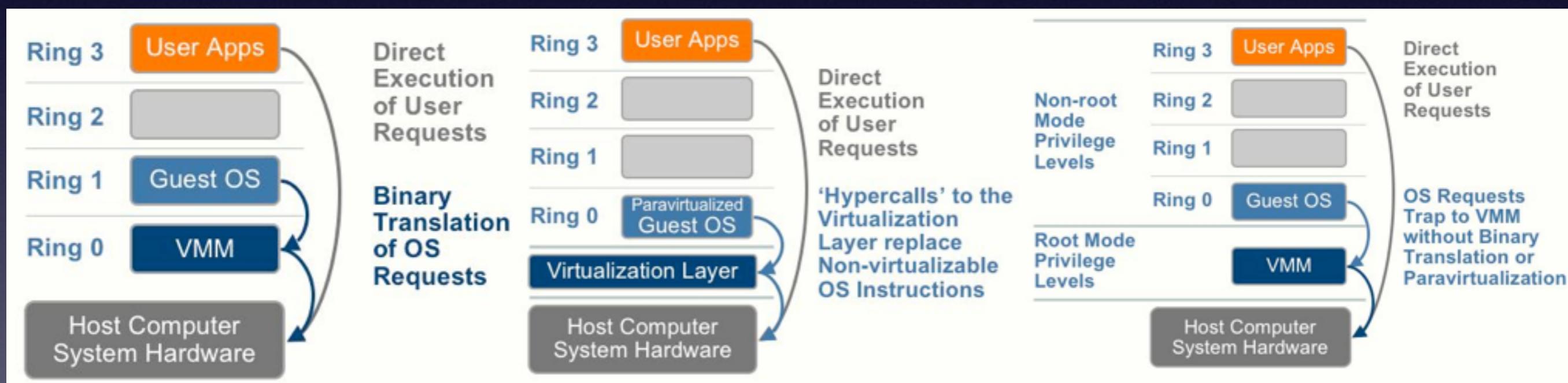
- Memory



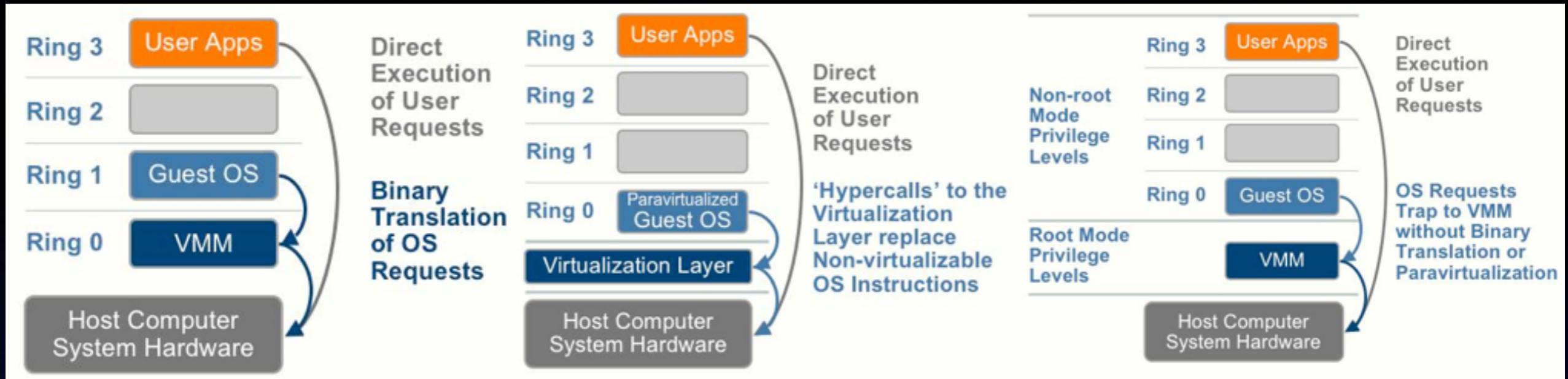
- I/O

# Virtualization in CPU Level

- CPU virtualization: abstract physical CPU into virtual CPU and supply those vCPUs for virtual machines to use



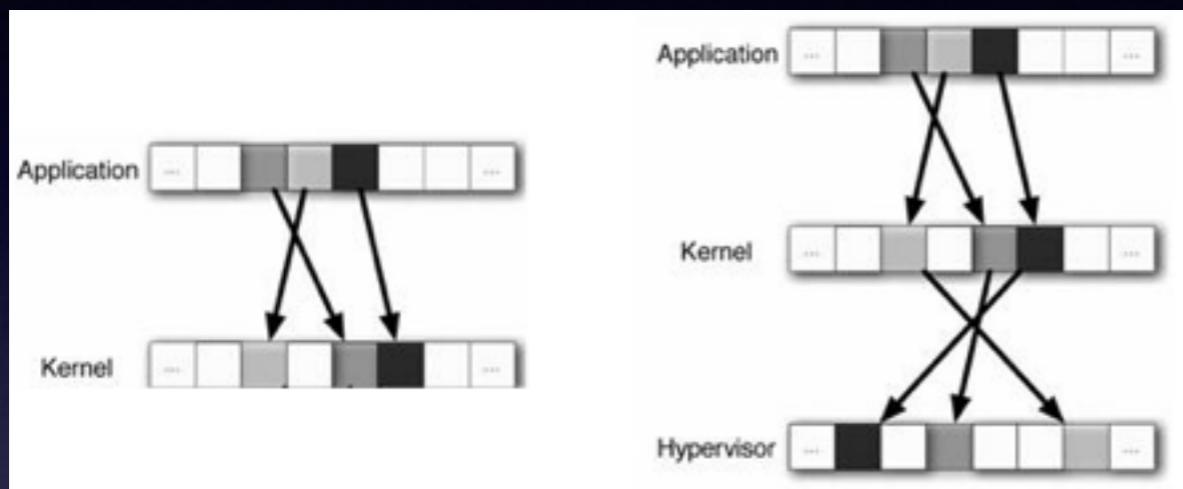
- Full virtualization
- OS-assisted (Para) virtualization
- Hardware-assisted virtualization



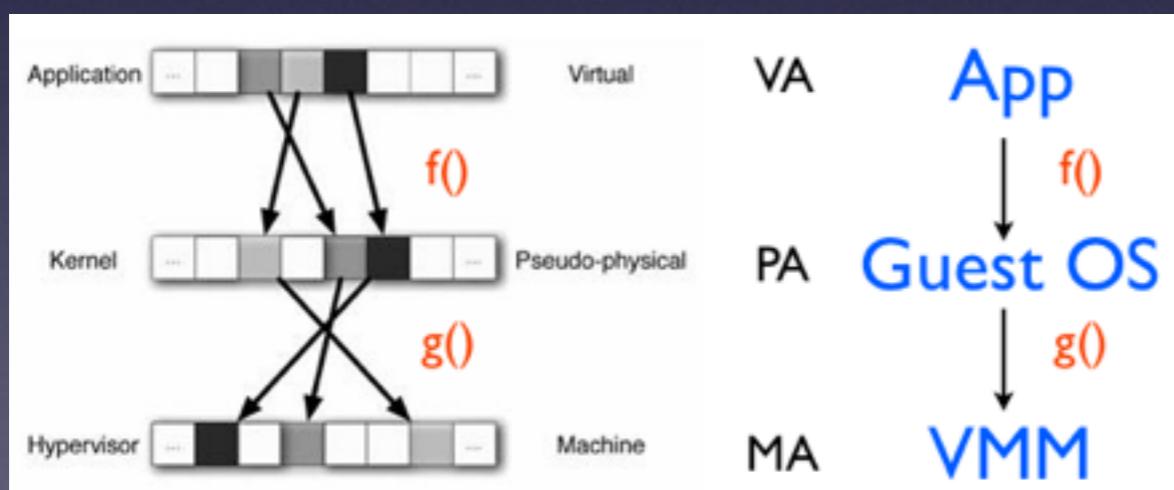
- Full virtualization
- OS-assisted (Para) virtualization
- Hardware-assisted virtualization

dynamic binary translation	hypercall	new hardware
no modified OS kernel	high performance	high performance, no modified OS
low performance	modified OS kernel	hardware limitation
Microsoft Virtual PC, Microsoft Virtual Server, VMware WorkStation(earlier version)	Xen, VMware ESX Server, Microsoft Hyper-V	Para-virtualization benefit from it

# Virtualization in Memory Level



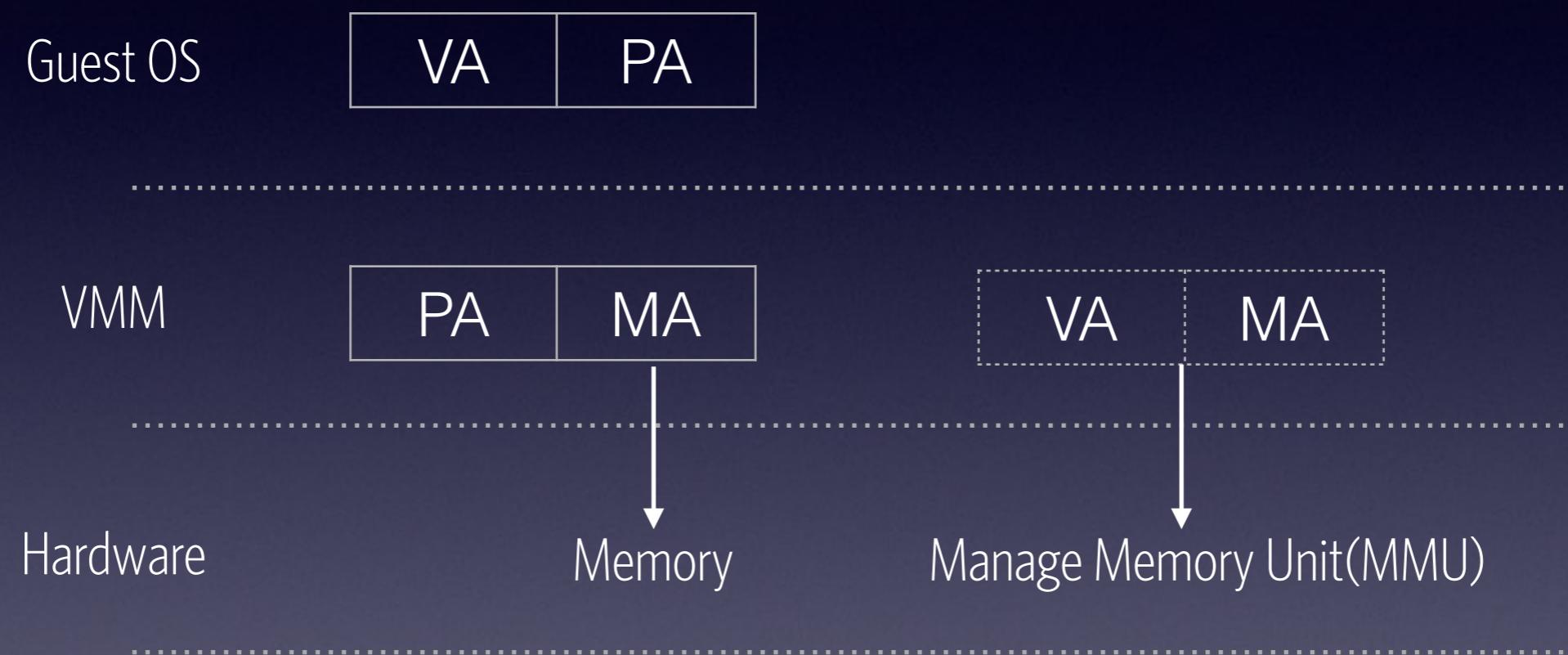
- One map by the Manage Memory Unit(MMU) in physical machines



- Two maps by the Manage Memory Unit(MMU) in virtual machines

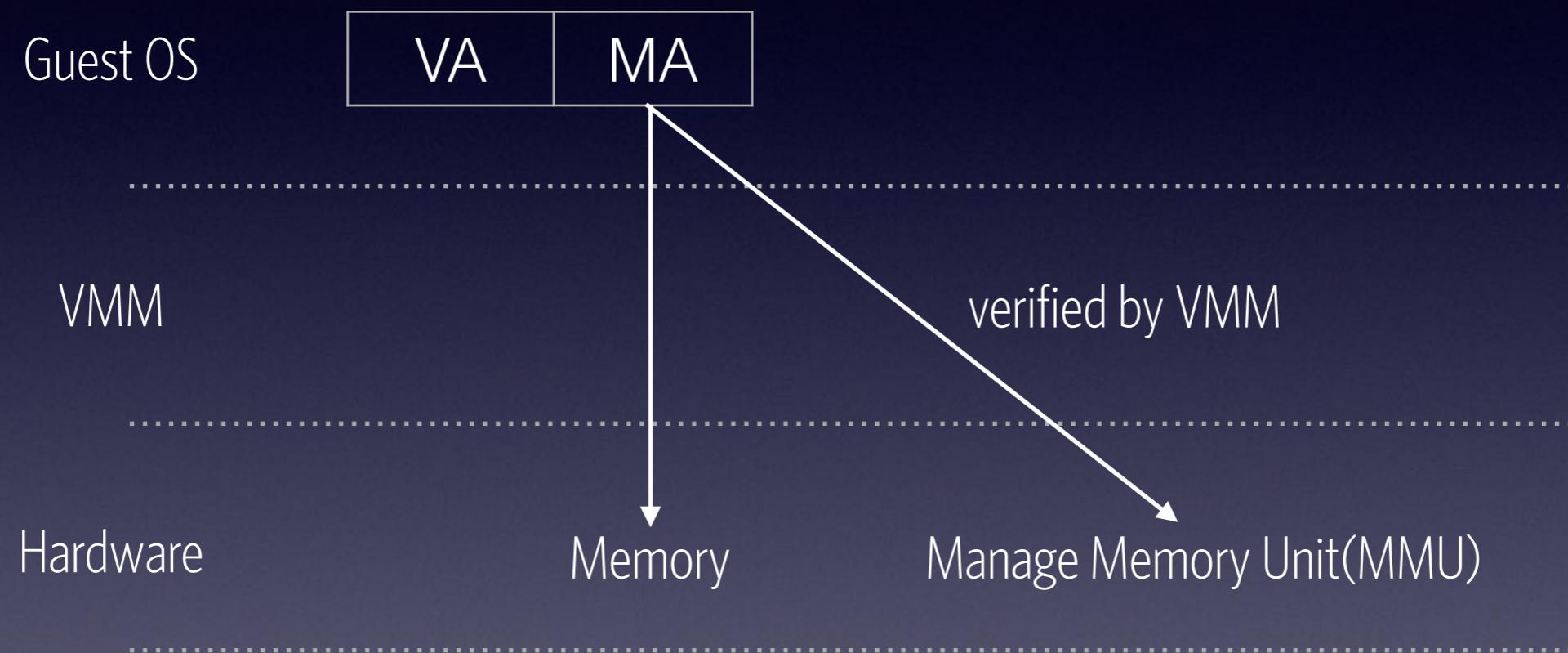
**Low efficiency!!!**

# Virtualization in Memory Level



- Shadow Page Table

# Virtualization in Memory Level



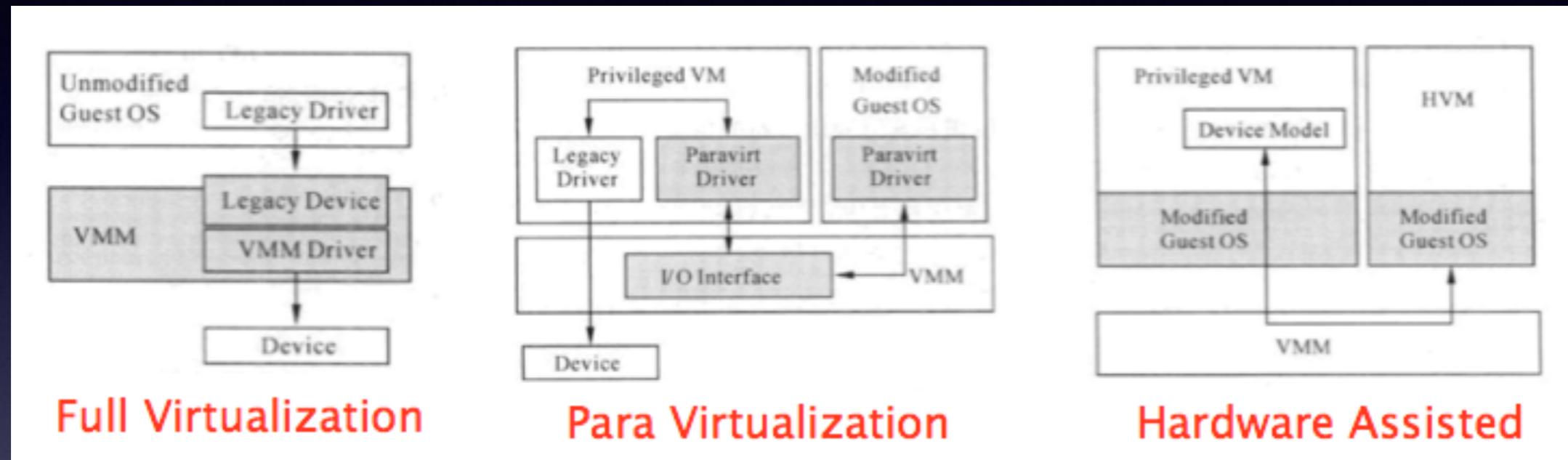
- MMU Para-virtualization

# Virtualization in Memory Level



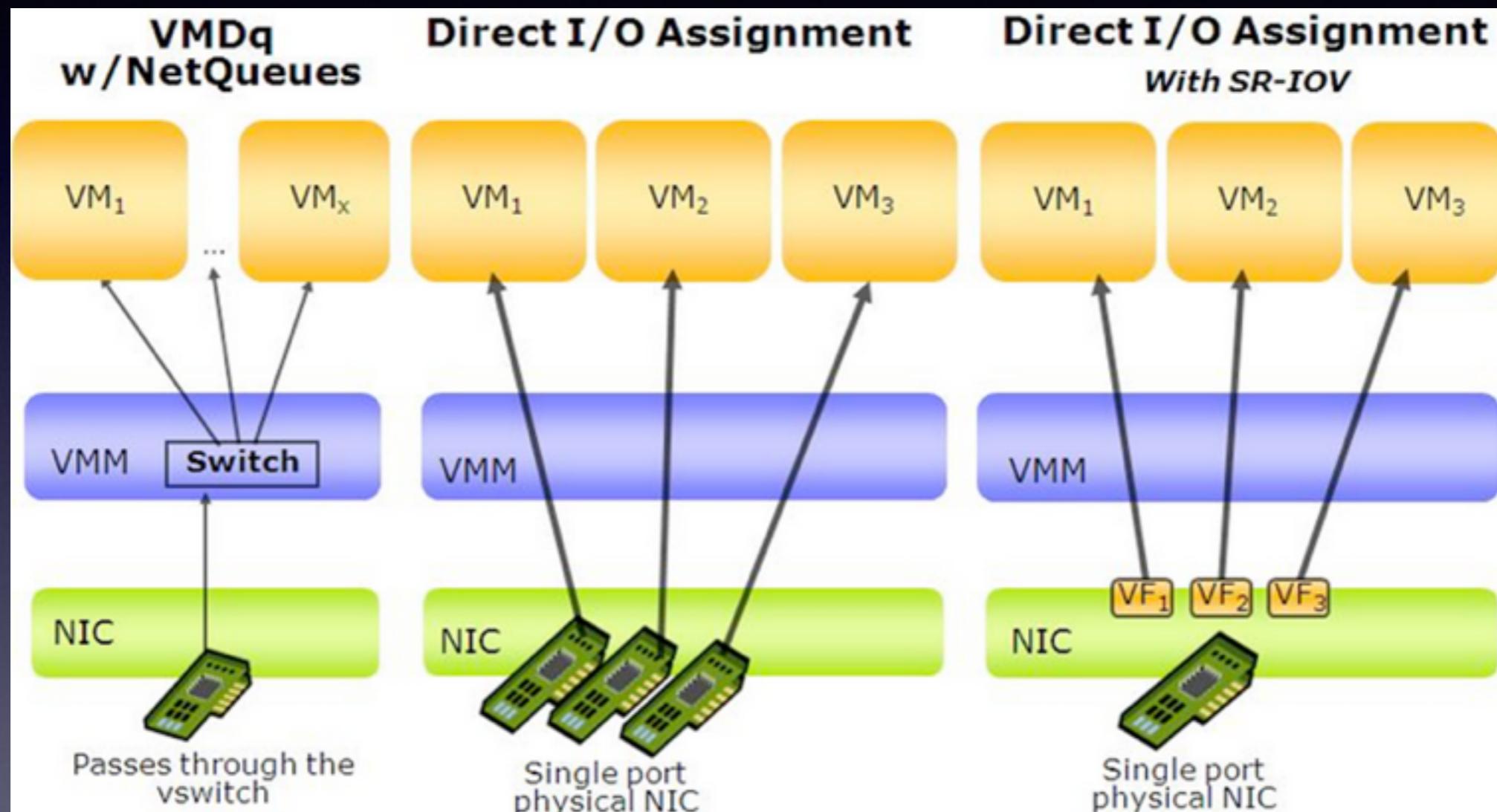
Shadow Page Table	MMU Para-virtualization
Time & Space overhead	modified OS kernel
Full virtualization, Hardware-assist virtualization	Para virtualization
VMWare Workstation, VMWare ESX Server, KVM	Xen

# Virtualization in I/O Level



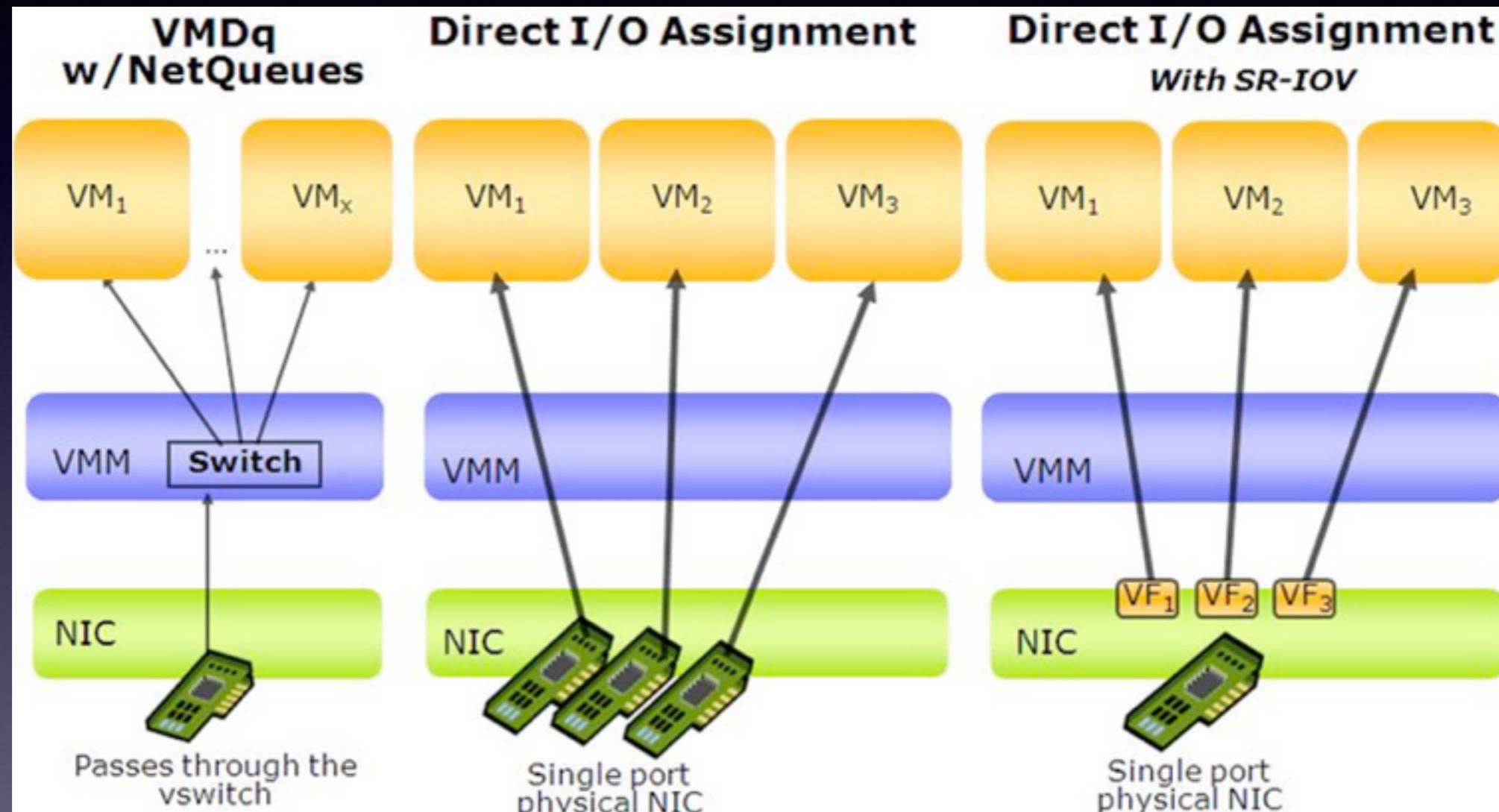
no modified OS kernel	high performance, low overhead	high performance, no modified OS
complicated, low performance	modified OS kernel, front-end/back-end	hardware limitation
VMware ESX Server, VMware Workstation	Xen	Para-virtualization benefit from it

# Virtualization in I/O Level



- shared virtual device
- dedicated devices, direct I/O
- single devices, multi virtual devices

# Virtualization in I/O Level



- low performance
- money...
- hardware support(e.g. SR-IOV)

# Virtualization in Resources Level

	Full virtualization	Para virtualization	Hardware-assisted virtualization
CPU virtualization	Dynamic binary translation	Hypercall	new instructions from hardware
Memory virtualization	Shadow Page Table	MMU para virtualization	Shadow Page Table
I/O virtualization	Device simulation	Para virtualization driver	direct visit

Go back to definition...

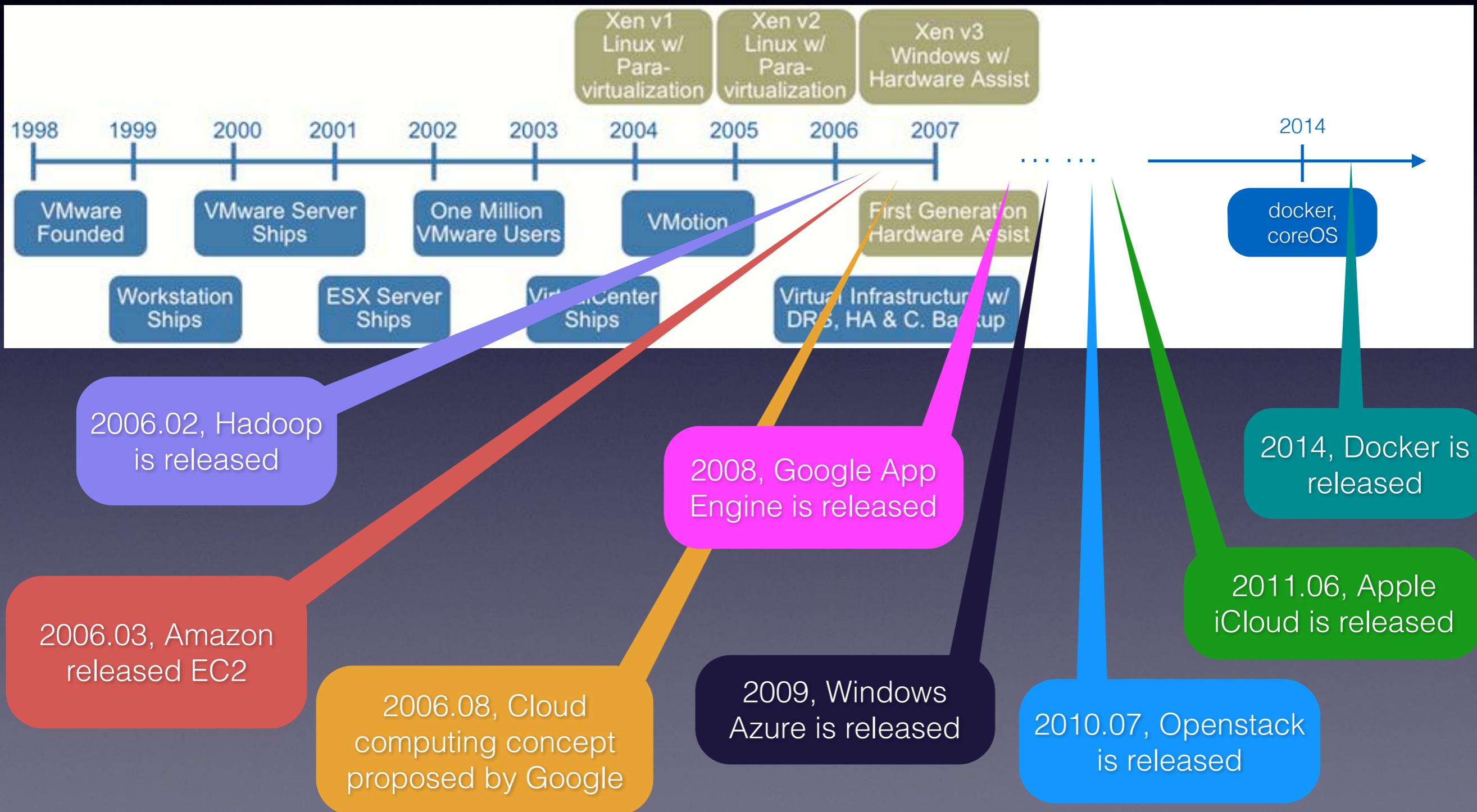
# What is virtualization?

- Virtualization refers to the act of creating a virtual (rather than actual) version of something, including (but not limited to) a virtual computer hardware platform, operating system (OS), storage device, or computer network resources. ----Wikipedia
- Virtualization is a logic view of resources, not limited to physical.  
---- IBM

# What is virtualization?

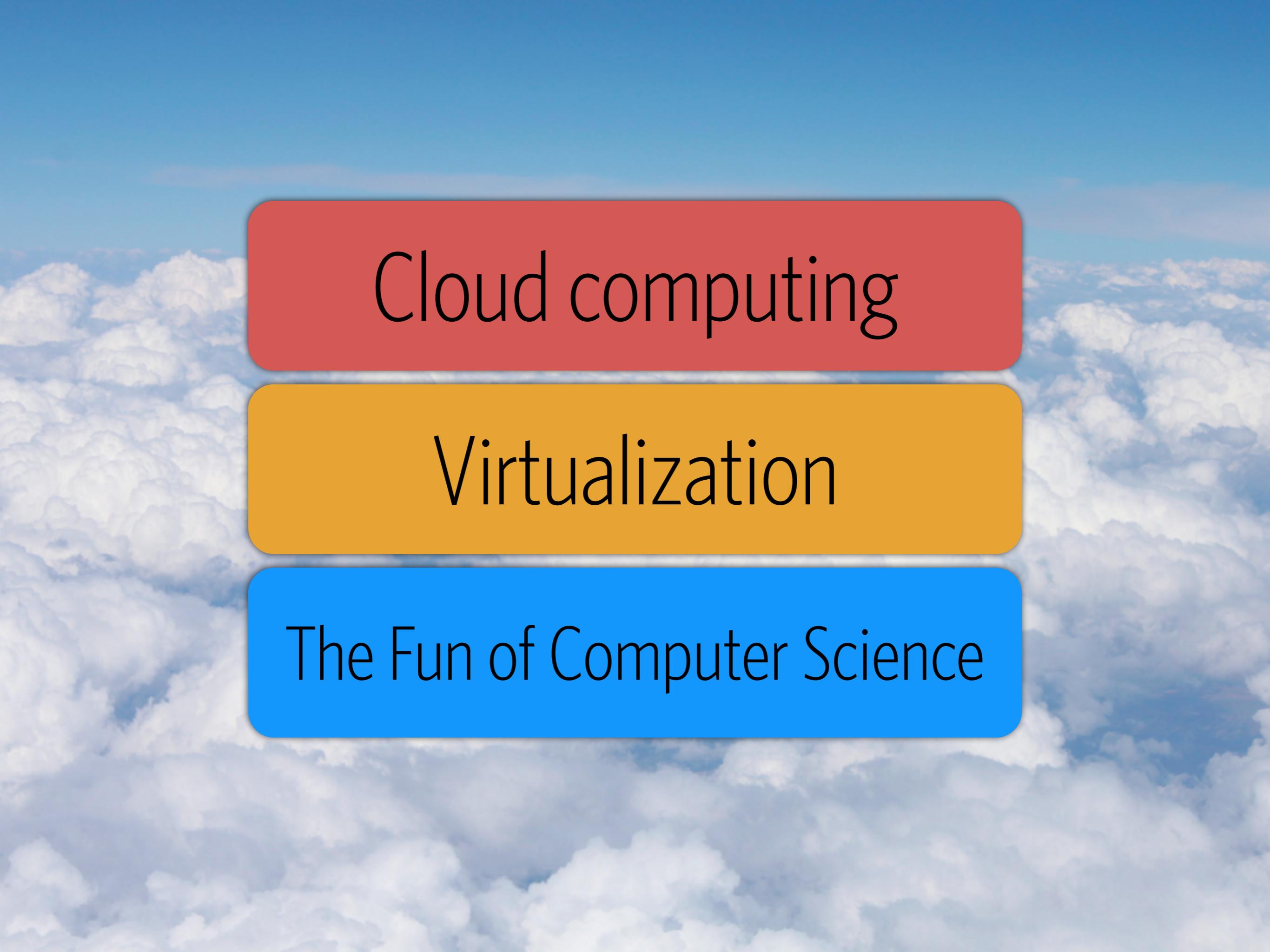
- (1) The object of virtualization is all kinds of hardware resources(like CPU, memory, storage, etc) and software resources(like OS, file system, etc)
- (2) The resources after virtualization hide unnecessary details to the users
- (3) Usually, users can get part or all functionalities in virtualized environment compared to real world

# Virtualization History



The background of the image is a clear, vibrant blue sky. It is heavily populated with large, white, puffy cumulus clouds. Some of these clouds are bright and well-defined against the blue, while others are more grey and overcast. The clouds vary in size and density, creating a textured, layered appearance across the entire frame.

Cloud computing



Cloud computing

Virtualization

The Fun of Computer Science

# Topic Today

- Why needs virtualization?
- What is virtualization ?

# Schedule



- Monday: Introduction
- Tuesday: Xen
- Wednesday: Xen
- Thursday: Docker
- Friday: Conclusion/homework solution

Thank you