NVIDIA®

Xen GPU

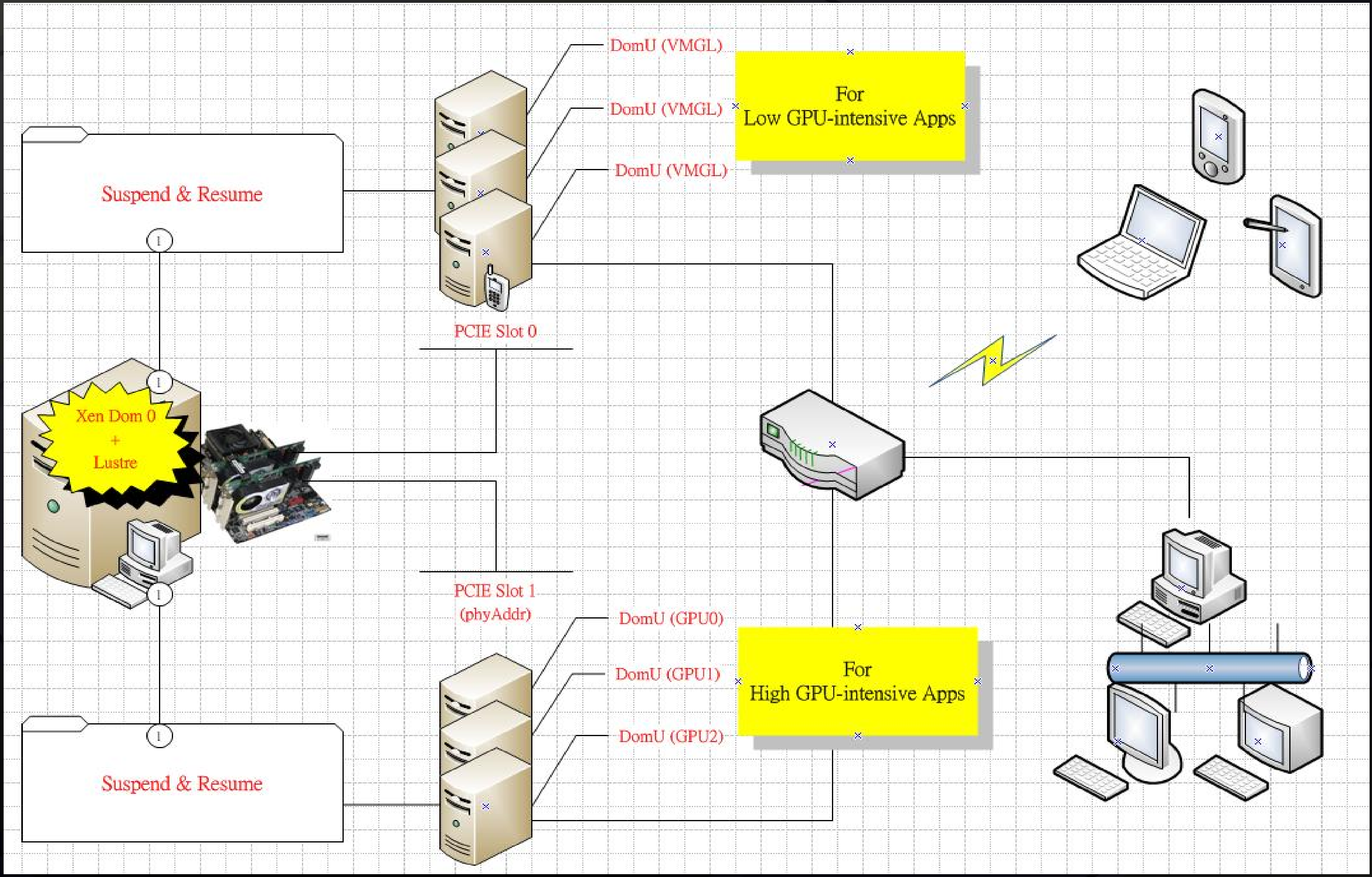
Rider

NCHC

Outline

- Target & Vision
- GPU & Xen
- CUDA on Xen
- GPU Hardware Acceleration On VM - VMGL

Target: Virt-GPU for HPC



Vision

- Cost Down – Higher C/P value
- Green – Energy saved
- Anytime , Anywhere – More convenient

Xen

+

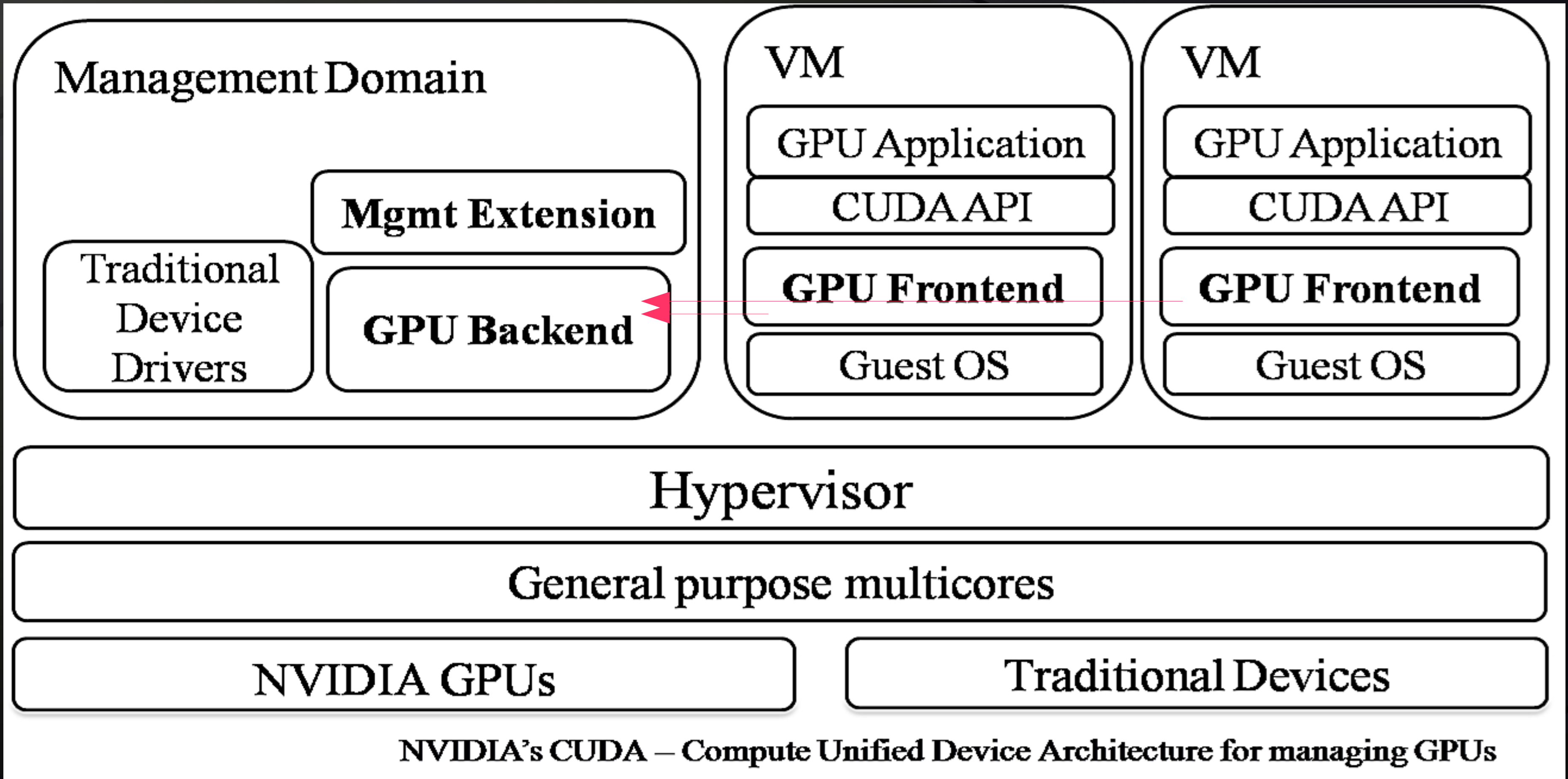
GPU

GPGPU On Virtual Machine

- GPGPU
- GViM
  - Split-driver model
  - VMM-bypass mechanism
  - Lower-level memory management mechanism
- Xen privileged domain
  - Memory management
  - Communication methods

Reference: http://www.cc.gatech.edu/~vishakha/files/GViM.pdf

Virtualization of GPUs



Reference: http://www.cc.gatech.edu/~vishakha/files/GViM.pdf

CUDA Intro

- "Compute Unified Device Architecture"
- General purpose programming model
  - User kicks off batches of threads on the GPU
  - GPU = dedicated super-threaded, massively data parallel co-processor
- Targeted software stack
  - Compute oriented drivers, language, and tools
- Driver for loading computation programs into GPU
  - Standalone Driver - Optimized for computation
  - Interface designed for compute – graphics-free API
  - Data sharing with OpenGL buffer objects
  - Guaranteed maximum download & readback speeds
  - Explicit GPU memory management

GPU Virtualization - Components

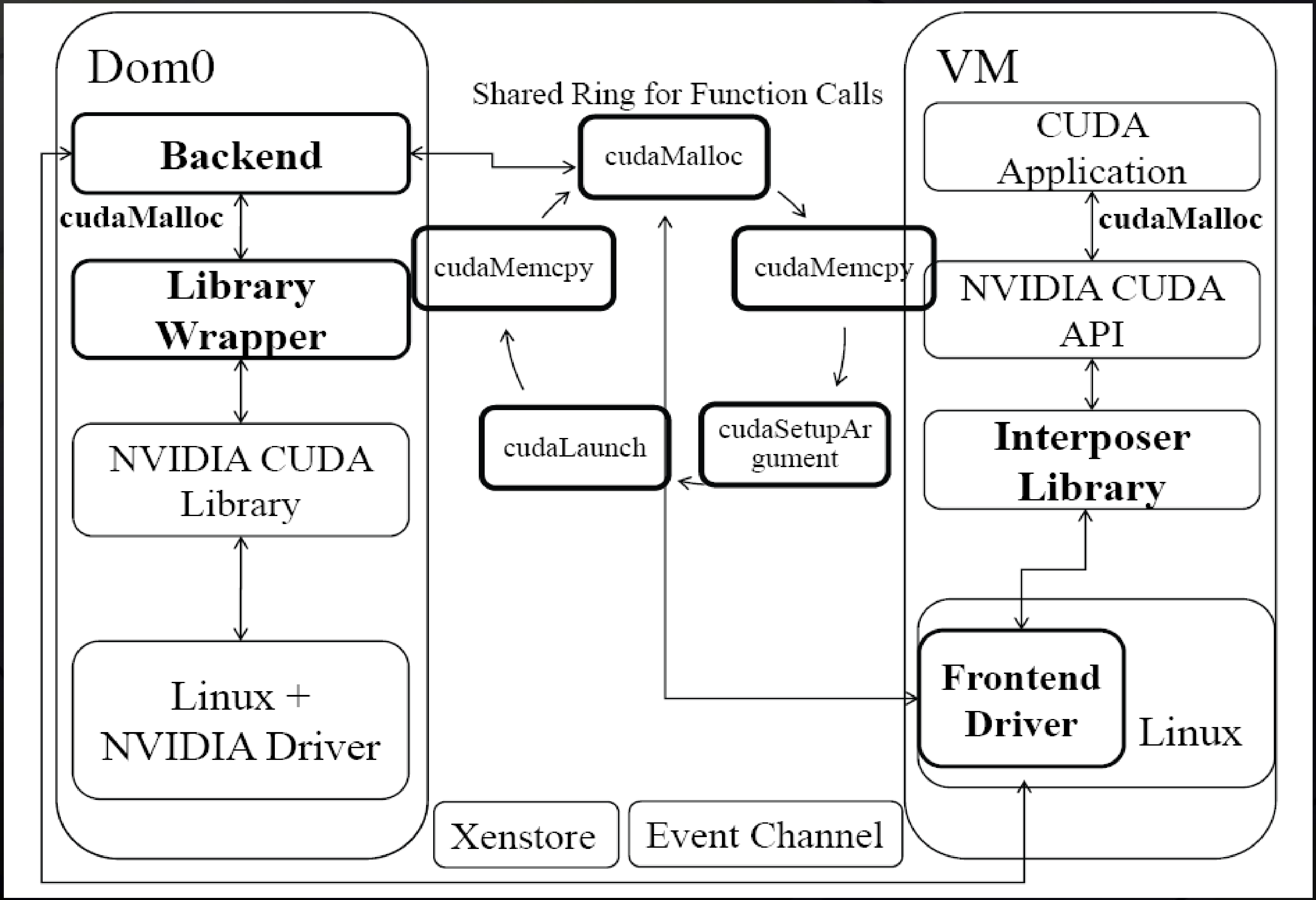
- Xen privileged domain
  - Memory management
  - Communication methods
- Using CUDA API
- The CUDA interposer library
- Fronted driver
  - Xen-bus
  - Xenstore
  - Shared call buffer

GPU Virtualization - Components

- The backend driver
  - User-level module
  - CUDA calls → execution results
- Library Wrapper function
  - Call packets → CUDA function calls
- CUDA library & driver
  - CUDA Driver
  - SDK
  - Toolkit

Reference: http://www.cc.gatech.edu/~vishakha/files/GViM.pdf

Virtualization components for GPU



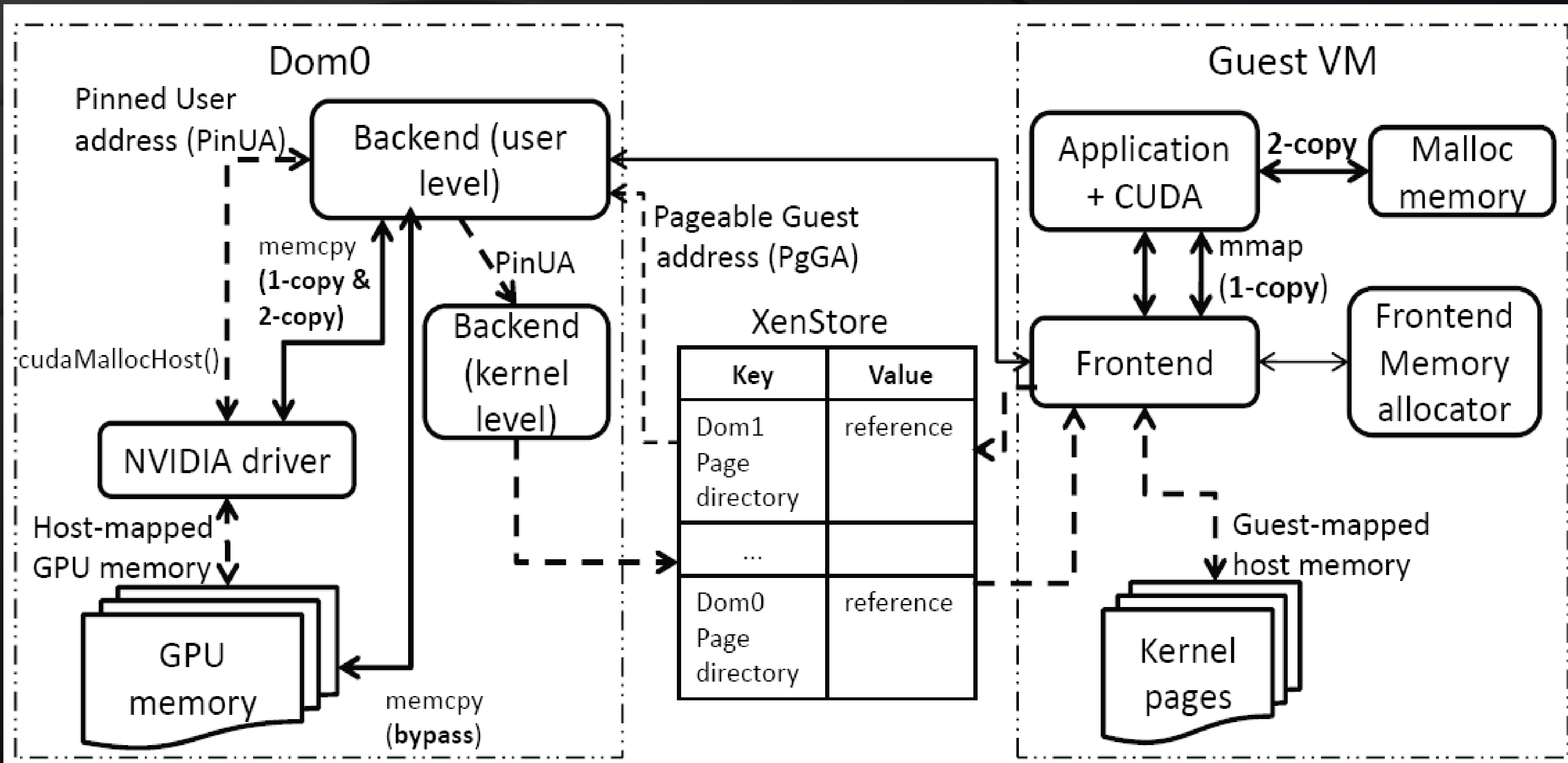
Reference: http://www.cc.gatech.edu/~vishakha/files/GViM.pdf

Memory Management

- 2-copy
  - malloc()
  - Host memory → GPU memory
- 1-copy
  - mmap()
  - Xenstore
- Zero-copy bypass
  - Limited data size

Reference: http://www.cc.gatech.edu/~vishakha/files/GViM.pdf

Memory Management



Reference: http://www.cc.gatech.edu/~vishakha/files/GViM.pdf

Conclusion

- Strengths
  - Arch of foresight, Optimized interface
- Weaknesses
  - Resource Mg, Scheduling, Dedicated Platform
- Opportunities
  - Green Computing – Power Mg
- Threats
  - Intel VT-d, NVIDIA Multi-OS

VMGL

Why Virtualize 3D ?

- Virtualization
  - Cost
  - Green
  - Convenience
- 3D
  - Interaction
  - Reality
  - Recreation
  - User-Interface

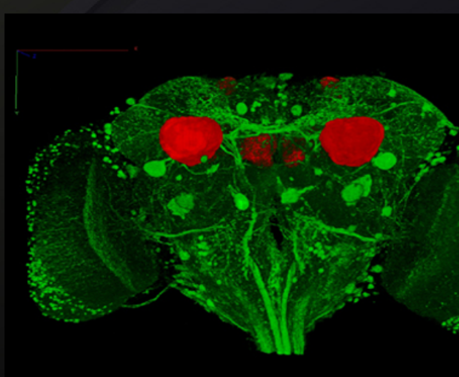
Virtualizing GPUs is difficult ?

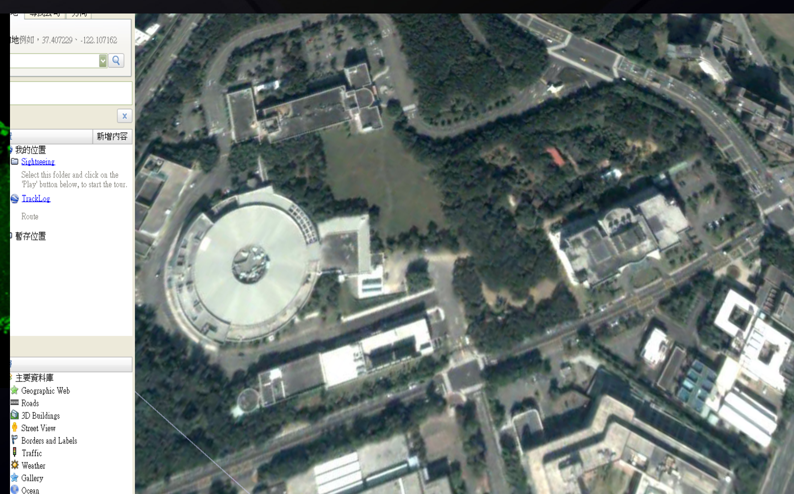
- Closed information
  - HW interface
  - Technical specification
  - Device driver
- Lack of standard
  - uncertain interface – AGP ? PCI ? PCI-X ? PCI – E ?
  - common standard – x86, IDE, SCSI ...

High-level API ?

Demand

- Graphics-intensive applications
- Excellent rendering performance
- Cross-Platform – VMs & GPUs
- Suspend & Resume across different GPUs
- Thin client User-Interface

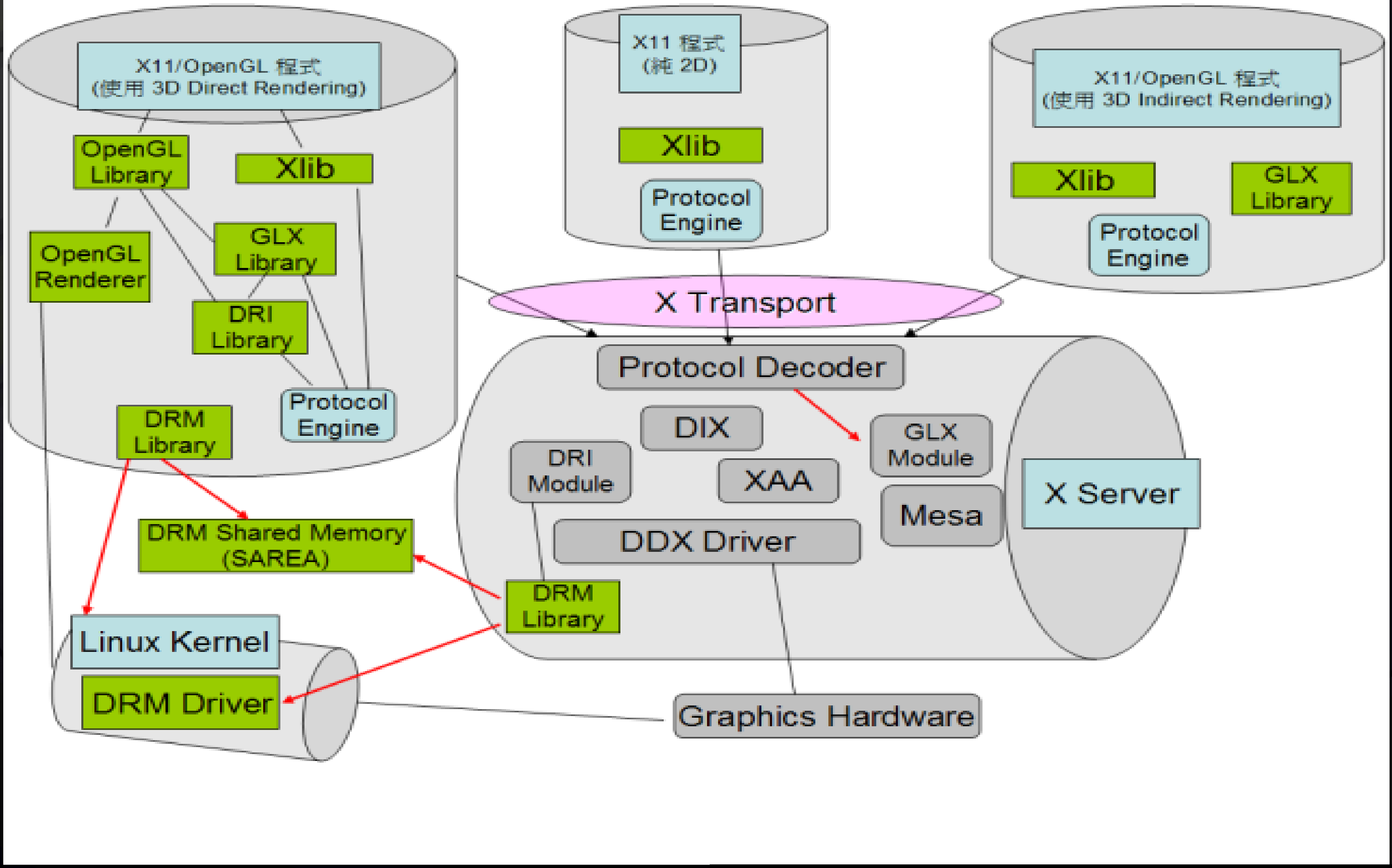




3D Graphics API

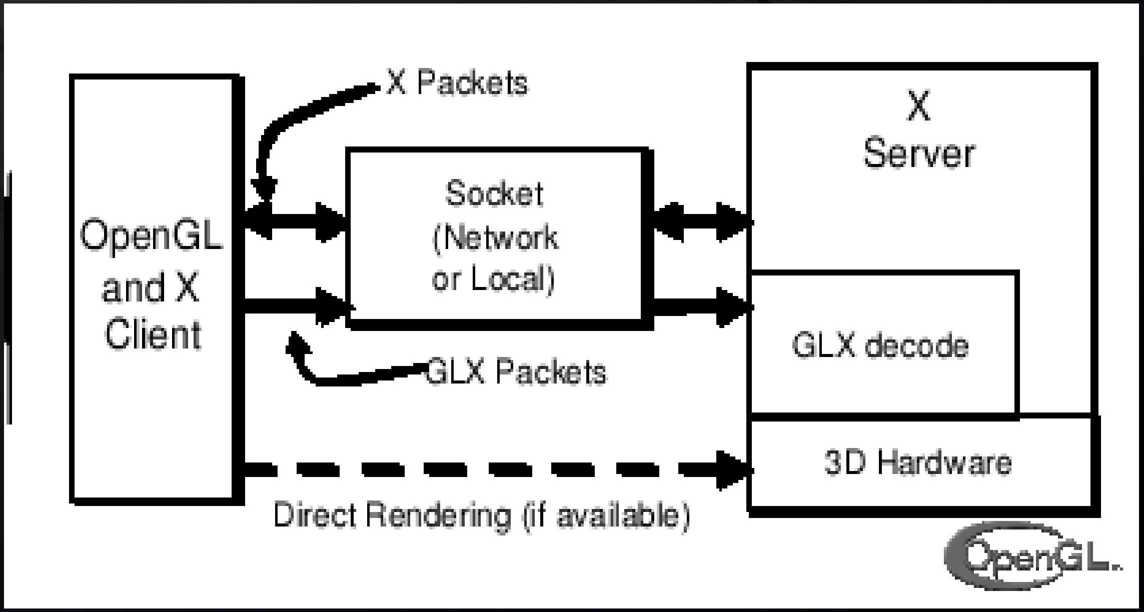
- Direct3D
- OpenGL

X11 / OpenGL



Reference: http://gallery.debian.org.tw/2004-12-02/dri_gram

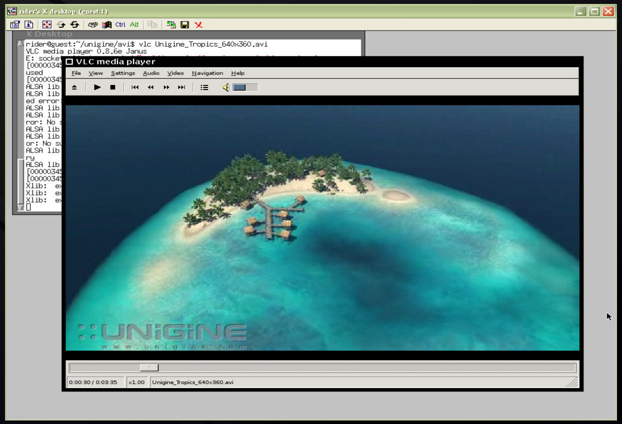
OpenGL extension – GLX
X Window – 3D Rendering



Reference: http://gallery.debian.org.tw/2004-12-02/dri_gram

Virtualized OpenGL - VMGL

- HW accelerated rendering
- VM Host
  - vendor-supplied GPU driver
  - OpenGL library
- Cross-Platform
- 87% or better of native HW acceleration
- Suspend & Resume



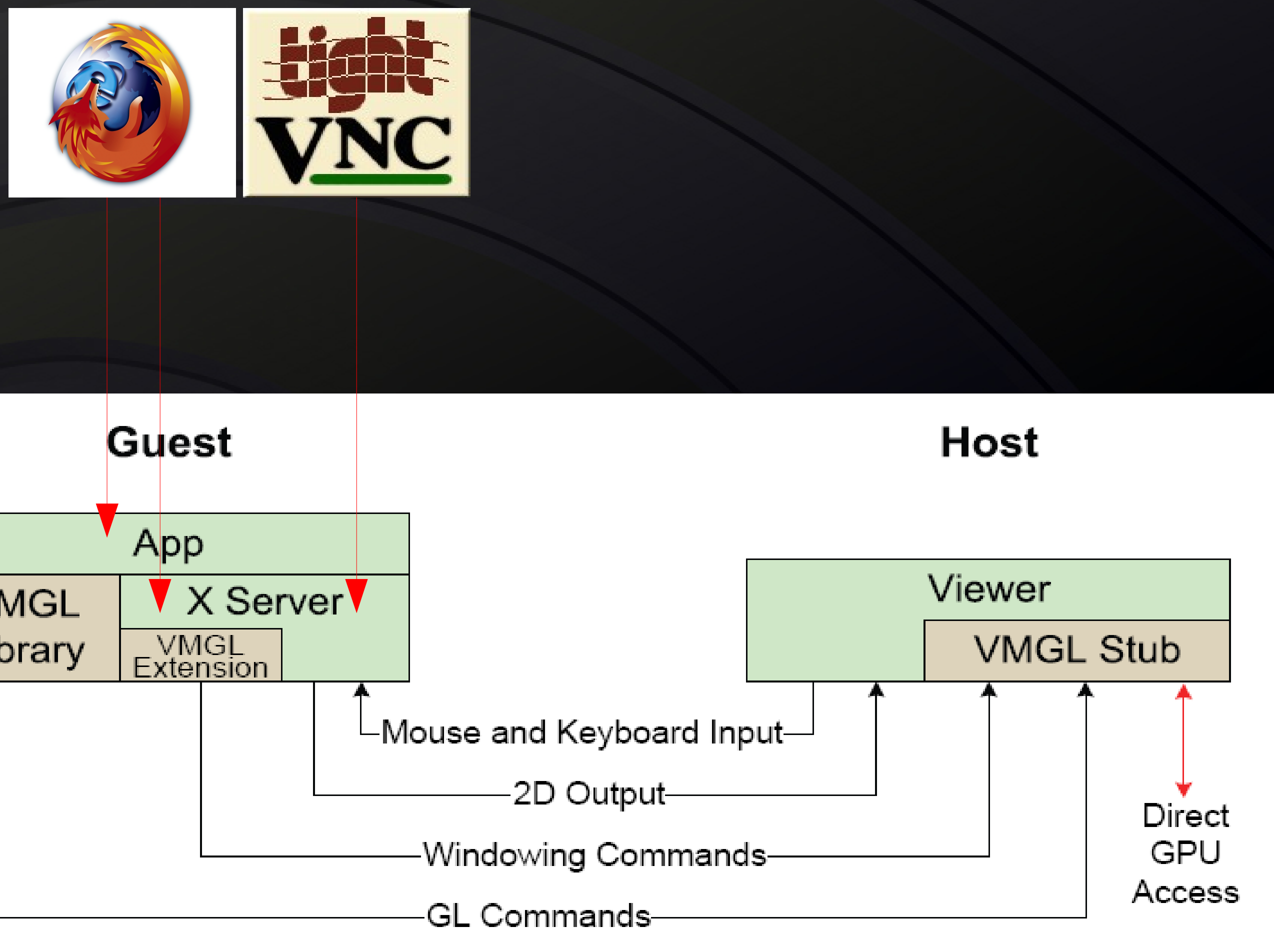
Reference: http://www.cs.toronto.edu/~andreslc/publications/slides/Xen-Summit-2007/vmgl.pdf

Architecture features - VMGL

- Virtualizing the OpenGL API
  - portability
  - compatibility
- Use a Network Transport
  - Cross-VM

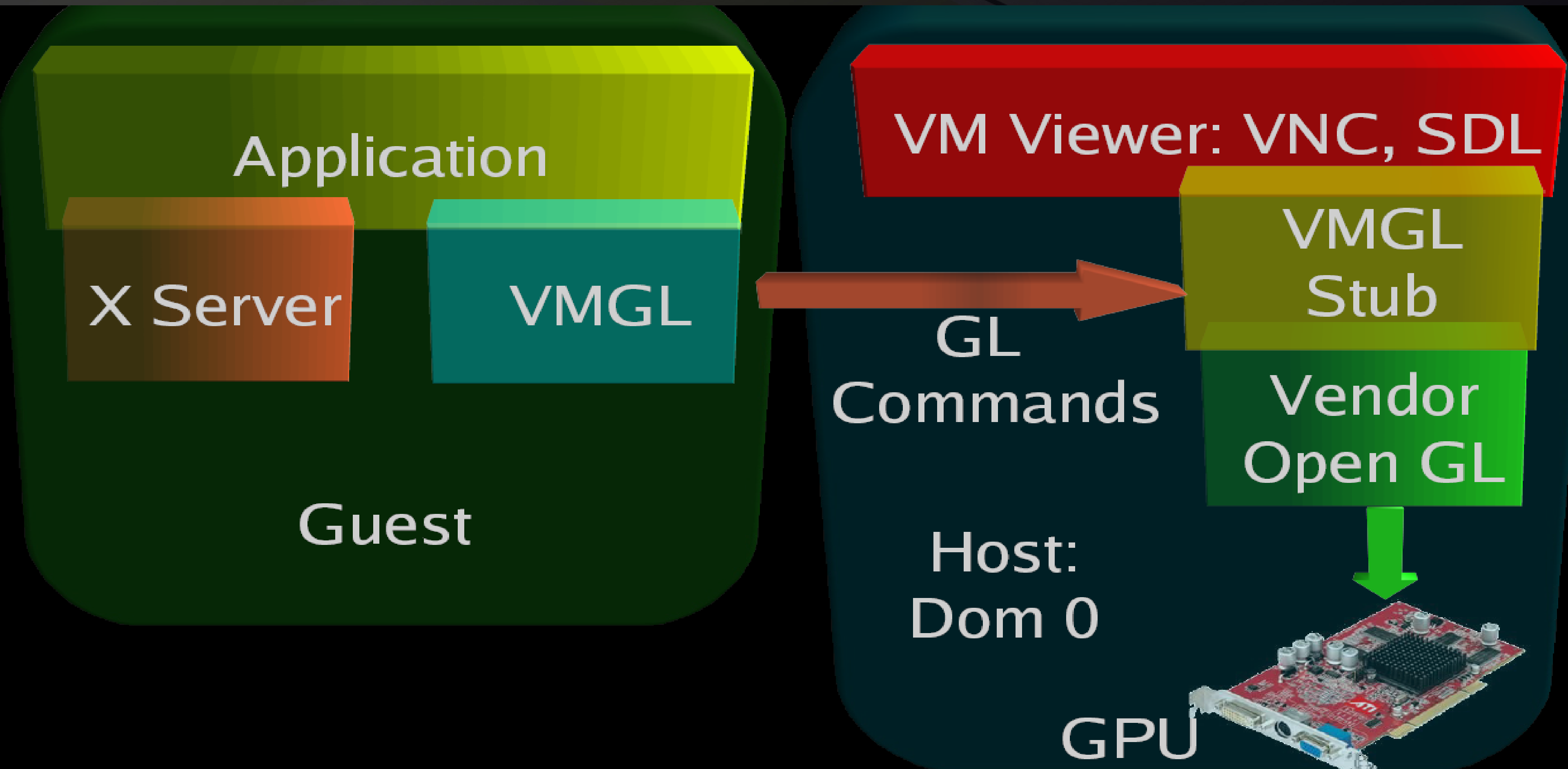
Reference: http://www.cs.toronto.edu/~andreslc/publications/slides/Xen-Summit-2007/vmgl.pdf

Architecture



Reference: http://www.cs.toronto.edu/~andreslc/publications/LagarCavillaVEE07.pdf

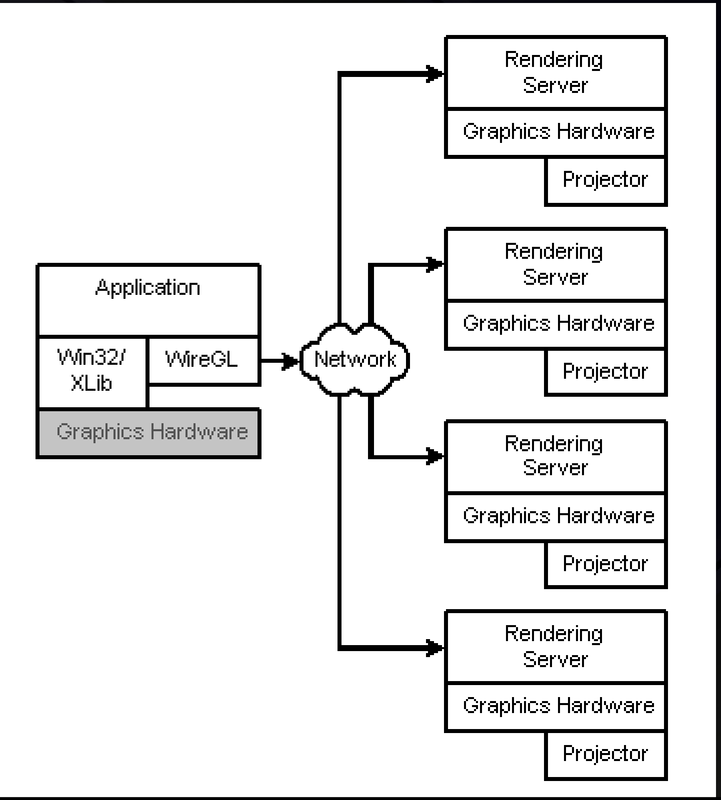
VMGL Apps in X11 Guest VMs



Reference: http://www.cs.toronto.edu/~andreslc/publications/slides/Xen-Summit-2007/vmgl.pdf

OpenGL transport (Problem?)

- Direct rendering PATH from Apps to Graphics Cards
  - GL stub
- WireGL
  - screen-visible state
  - OpenGL command queues



Reference: http://www.cs.toronto.edu/~andreslc/publications/LagarCavillaVEE07.pdf

Suspend & Resume

- Shadow driver
  - keep OpenGL state
- OpenGL contexts
  - Global Context State
  - Texture State
  - Display Lists

Reference: http://www.cs.toronto.edu/~andreslc/publications/LagarCavillaVEE07.pdf

Conclusion

- Strengths
  - Cross-Platform, Lightweight, X11 Forwarding Performance
- Weaknesses
  - Performance via VNC, GL extensions, Shared Memory
- Opportunities
  - Thin client, WebOS
- Threats
  - Intel VT-d, NVIDIA Multi-OS

Reference

- NVIDIA
  - http://www.nvidia.com/object/cuda_home.html
  - http://forums.nvidia.com/index.php?
  - David Kirk/NVIDIA and Wen-mei W. Hwu, 2007 ECE 498AL, University of Illinois, Urbana-Champaign
- GViM
  - http://www.cc.gatech.edu/~vishakha/files/GViM.pdf
- VMGL
  - http://www.cs.toronto.edu/~andreslc/xen-gl/
  - http://www.cs.toronto.edu/~andreslc/publications/LagarCavillaVEE07.pdf
  - http://www.cs.toronto.edu/~andreslc/publications/slides/Xen-Summit-2007/vmgl.pdf
- Xen
  - http://trac.nchc.org.tw/grid/wiki/Reading/XenP
  - http://www.virtuatopia.com/index.php/Xen_Virtualization_Essentials

Q & A