# AIST Super Green Cloud

## A build-once-run-everywhere high performance computing platform


Takahiro Hirofuchi, Ryosei Takano, Yusuke Tanimura, Atsuko Takefusa, and Yoshio Tanaka

Information Technology Research Institute (ITRI), National Institute of Advanced Industrial Science and Technology (AIST), Japan



HPC2014, Cetraro, Italy, July 7-11, 2014

# Outline

1. AIST Super Green Cloud
   - A new super computer for us
   - Fully virtualized
   - HPC benchmarks on PCI-pass-through VM
   - Dynamic consolidation
2. VM Live Migration for HPC
   - Hybrid (Pre- & Post-copy) live migration
   - Interconnect-agnostic live migration
   - SAN-assisted fast live migration

# About AIST


AIST Booth@SC13


The National Institute of Advanced Industrial Science and Technology (AIST) is Japan's largest public research organization, with about 3100 employees. It is an independent administrative institution associated with the Ministry of Economy, Trade and Industry (METI). AIST conducts research and development in six fields: Life Science and Technology; Information Technology; Nanotechnology, Materials and Manufacturing; the Environment and Energy; Geological Survey and Applied Geosciences; and Meteorology and Measurement Technology.

# History of AIST Clusters

| | AIST Super Cluster (ASC) | AIST Green Cloud (AGC) | AIST Super Cloud (ASC) | AIST Super Green Cloud (ASGC) |
|---|---|---|---|---|
| Year | 2004 | 2009 | 2011 | **2014** |
| Cost (Million USD) | 18 | 0.76 | 0.66 | **2** |
| # of nodes | 1408 | 128 | 192 | **155** |
| # of total cores | 2816 | 1024 | 1536 | **3100** |
| Peak performance (TFlops) | 14.6 | 10.4 | 14.6 | **70** |
| Actual energy consumption (KW) | 460 | 40 | 48 | **13-70** |
| GFlops/KW | 32 | 260 | 304 | **1000** |
| Provided resource | PMs | PMs | PMs (Experimentally VMs) | **Only VMs** |
| Target applications | HPC | HPC | HPC | **HPC and non-HPC** |

# Hardware Specification

| | AIST Super Green Cloud |
|---|---|
| Year | 2014 July |
| Processors | Intel Xeon E5-2680v2 2.8GHz 10core x2 |
| CPU of a Node (# of Cores / Processors) | 20/2 |
| Memory of a Node (GB) | 128 |
| Storage of a Node | 600GB SSD |
| **# of Nodes** | **155** |
| **# of Total Cores** | **3100** |
| Interconnection | Infiniband FDR (56Gbps, non-blocking) 10G Ethernet (non-blocking) |
| Speed (TFlops) | 70 |
| Power Consumption (KW) | 13～70 |
| GFlops/KW | 1000 |


Cray H2312

# ASGC Features

- For users, ASGC will provide
  - Build once, run everywhere
    - Use VM technology
    - Portable to other cloud services
  - Pay as you go
    - Use as your budget can afford ☺
  - High performance
    - Support both HPC and non-HPC users
- For admins, ASGC will provide
  - Green
    - Power off unused nodes
  - Research platform for cloud computing
    - Collect actual resource use
    - Try crazy ideas of resource management

# Apache CloudStack & Qemu/KVM

- Provide VMs, not PMs
- Support HPC and non-HPC applications
  - E.g., scientific simulations and big data processing
  - E.g., web server, intranet server, and desktop
- Extension by AIST
  - Support PCI Pass-through & SR-IOV
  - VM image export to Amazon EC2 and others
  - Dynamic VM consolidation with our migration technologies

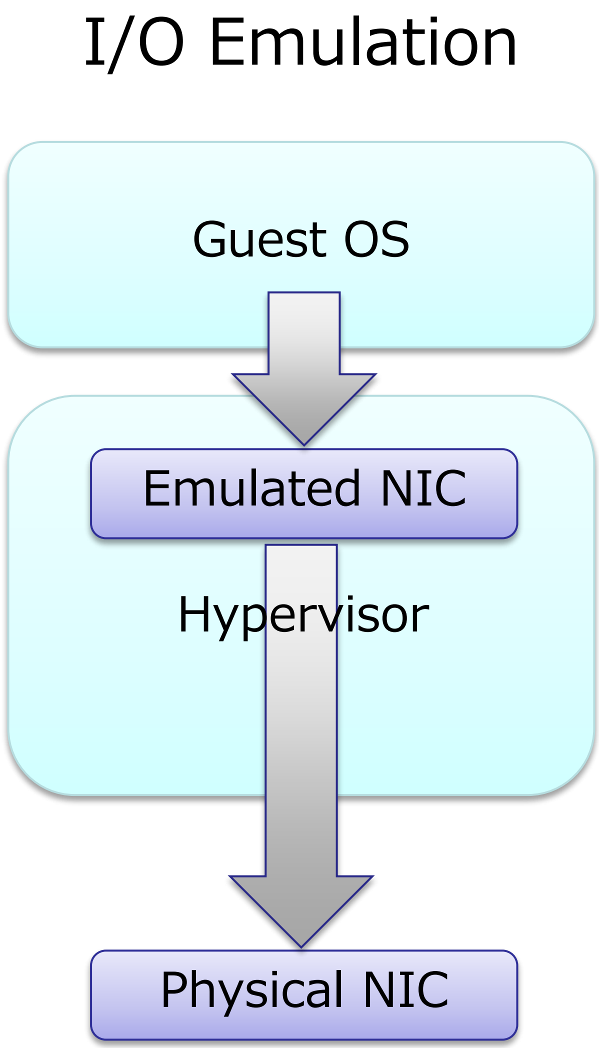
# VM Instance Types

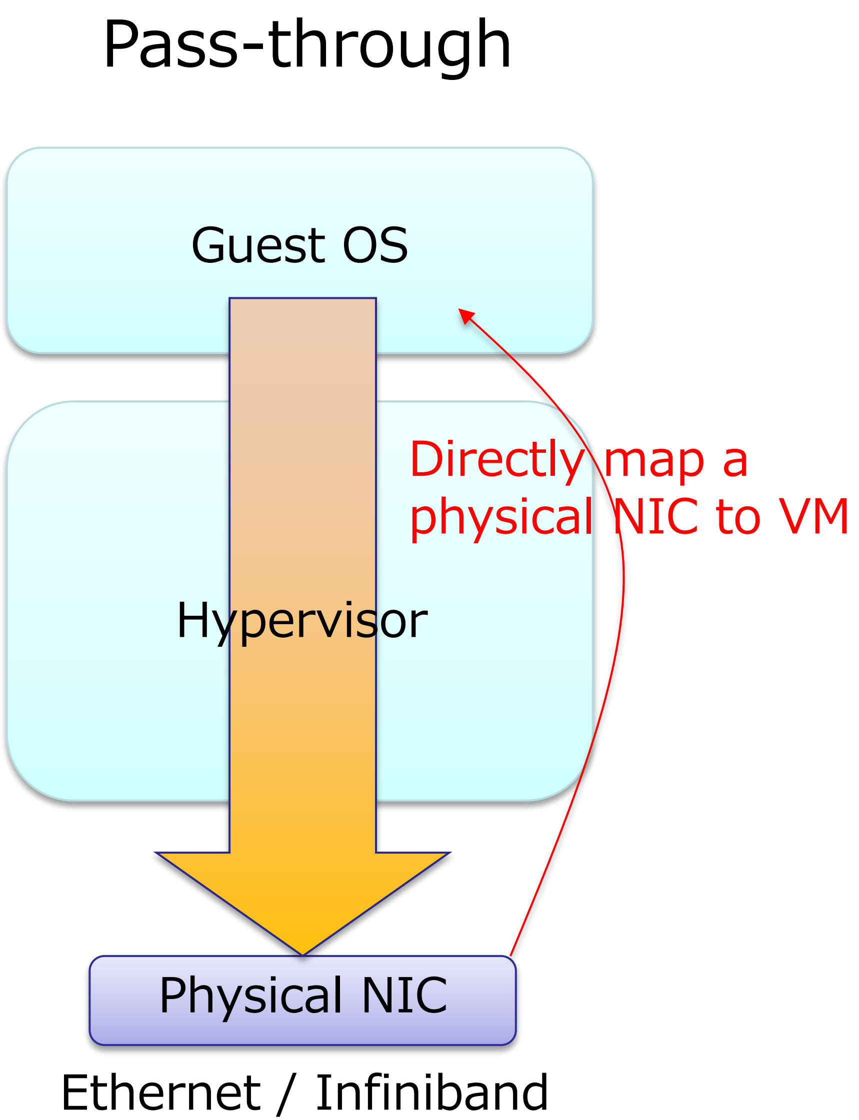
- HPC VM
  - Occupy an entire PM
    - 20-cores VCPUs, 120GB RAM, 472GB HDD
    - InfiniBand and Ethernet
    - PCI Pass-through and SR-IOV
  - 1 VM*hour: $0.16 (i.e., $120 per month)
  - A template VM image has pre-installed HPC programs (TORQUE, OpenMPI, Intel Compiler, etc)
  - Use-cases: bio/geo/nao simulations, big data processing
- Standard VM
  - Share a PM with other VMs
    - 1-core VCPU, 6GB RAM, 22GB HDD
  - 1 VM*hour: $0.08 (i.e., $6 per month)
  - Use-cases: web servers, intranet servers, desktop

# Challenge of Virtualization

- Benefits of using VMs
  - Encapsulate user's environment
  - Share a single PM with other VMs
    - E.g., standard VM in ASGC
  - Safely manage physical resource
    - Do not allow users to do something stupid un-/intentionally
    - E.g., flush BIOS, kill important daemon
  - Consolidation through live migration

- Challenge
  - Performance overhead due to virtualization
  - **We aims at providing users with the almost same performance as physical machines**
    - Pin VCPUs to physical CPUs
    - Use PCI pass-through and SR-IOV
      - E.g., HPC VM in ASGC


I/O Emulation
Guest OS
Emulated NIC
Hypervisor
Physical NIC


Pass-through
Guest OS
Hypervisor
Directly map a physical NIC to VM
Physical NIC
Ethernet / Infiniband

# Preliminary Experiments

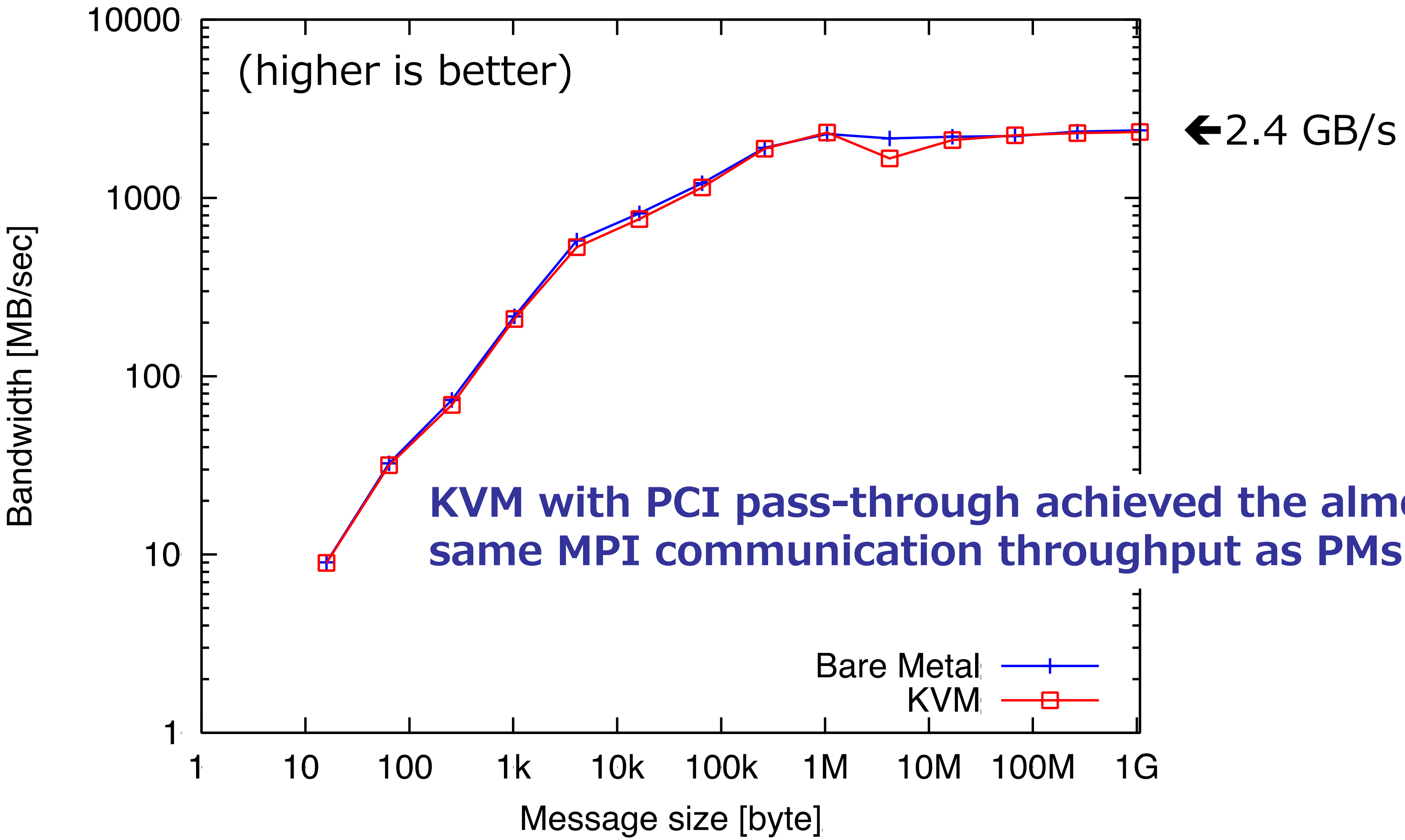
Blade server for experiments

| | Dell PowerEdge M610 |
|---|---|
| Year | 2013 |
| Processors | Intel Quad-core Xeon E5540/2.53GHz x2 |
| CPU of a Node<br>(# of Cores / Processors) | 8/2 |
| # of Nodes | 16 |
| Memory of a Node (GB) | 48 (DDR3) |
| Interconnection | InfiniBand QDR<br>Mellanox ConnectX (MT26428)<br>Connected via Mellanox M3601Q (QDR 16 port) |

- Assign one VM（8 VCPUs, 45GB RAM）to each PM
- Run a benchmark program on VMs or PMs

- Debian 6.0.1, Linux Kernel 2.6.32-5-amd64,
  KVM-0.12.50, gcc/gfortran-4.4.5, Open MPI 1.4.2

# MPI Point-to-Point Communication




**KVM with PCI pass-through achieved the almost same MPI communication throughput as PMs.**

# HPL (High-Performance LINPACK)

- Compute intensive
- Setting NUMA affinity is effective
  - For PMs and VMs
- Virtualization overhead is 8%

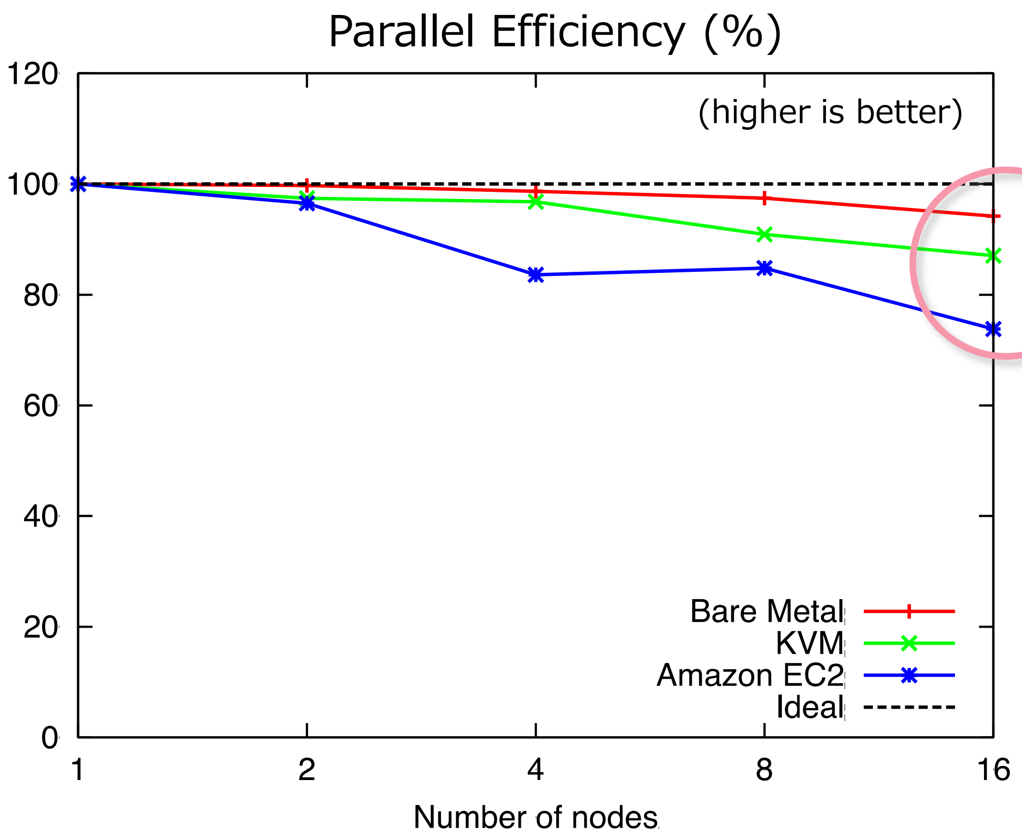
[GFlops]

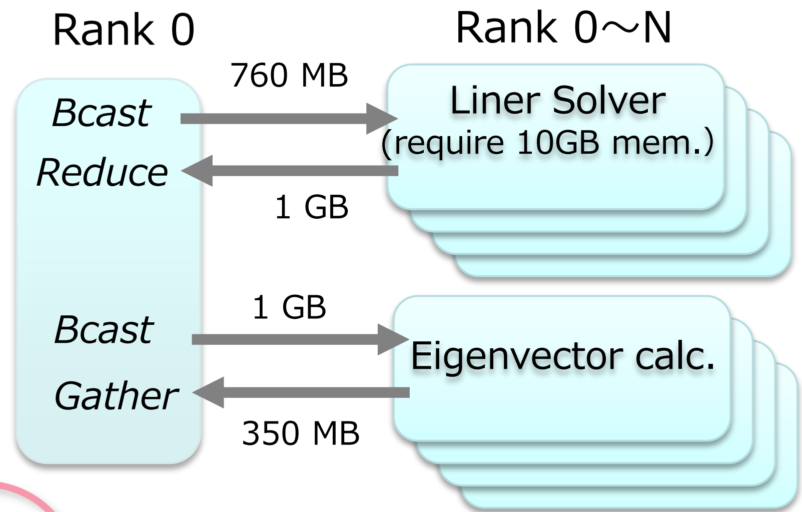
| Configuration | 1 node | 16 nodes |
|---|---|---|
| PM | 50.24 (0.98) | 706.21 (0.94) |
| PM w/ affinity | 51.07 (1.00) | **747.88 (1.00)** |
| VM | 48.03 (0.94) | 671.97 (0.90) |
| VM w/ affinity | 49.33 (0.97) | **684.96 (0.92)** |

The LINPACK efficiency is 57.7% in the 16 nodes of PMs (63.1% in a single node)

# Bloss: Non-linear Eigensolver

Hierarchical parallel program
by MPI and OpenMP




coarse-grained MPI comm.

Degradation of parallel efficiency is 8% in KVM with pass-through.

# Thoughts

- Using PCI pass-though and enabling CPU affinity, a VM got the *almost* same performance as a physical machine.
- There still remains small performance degradation (e.g., 8%)
  - For ASGC, it's a trade-off!
  - Perhaps, this would be a research topic?

# ASGC Implementation

- CloudStack 4.3.0 + many bug fix patches + our extension
  - Our extension
    - PCI Pass-through & SR-IOV support
    - Advanced accounting support
      - Bind billing data to each research project
    - ASGC/Amazon EC2 image converter
    - Utility programs to make a virtual cluster easily
- Zabbix
  - Monitor the usage of all physical and virtual hardware resources
    - CPU, memory, network interface, disk
    - PMs and VMs
- RADOS (Reliable Autonomic Distributed Object Store)
  - Store virtual machine images
    - Currently just for template images
    - In future for on-line virtual disks to support live migration
  - 10 storage nodes (160TB in total)
  - 3 copy replicas
  - Export storage via an Amazon-S3 compatible service (i.e., RADOS Gateway)

# Practice & Experiences, so far

- Playing with CloudStack is uphill battle
  - Lack of a stable release?
  - Lack of documentation of API?
  - Should migrate to OpenStack?
  - Does anyone share experience with us?
- Enlightening non-IT guys about cloud computing is still on-going
  - They do not want to migrate to ASGC yet
  - Why?
    - They do not know how their cluster is inefficient

# Current Status

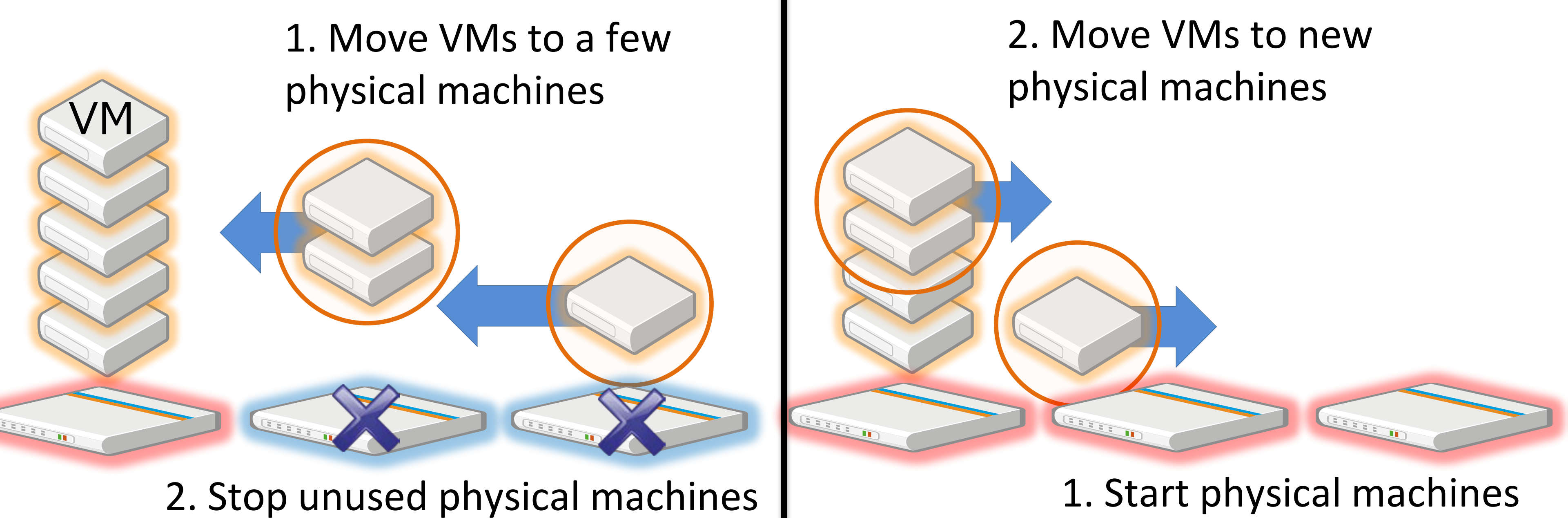
- Phase 0: 2013-
  - Obtain machines (delivered at the end of FY2013)
  - Start developing the system
- **Phase 1: 2014/07-**
  - **Start service with basic features**
  - **Collect resource usage**
- Phase 2: 2015/xx-
  - Support image export to external clouds
  - **Start dynamic VM consolidation using our live migration technologies**

# Reduce Excessive Energy Consumption by Dynamic VM Consolidation

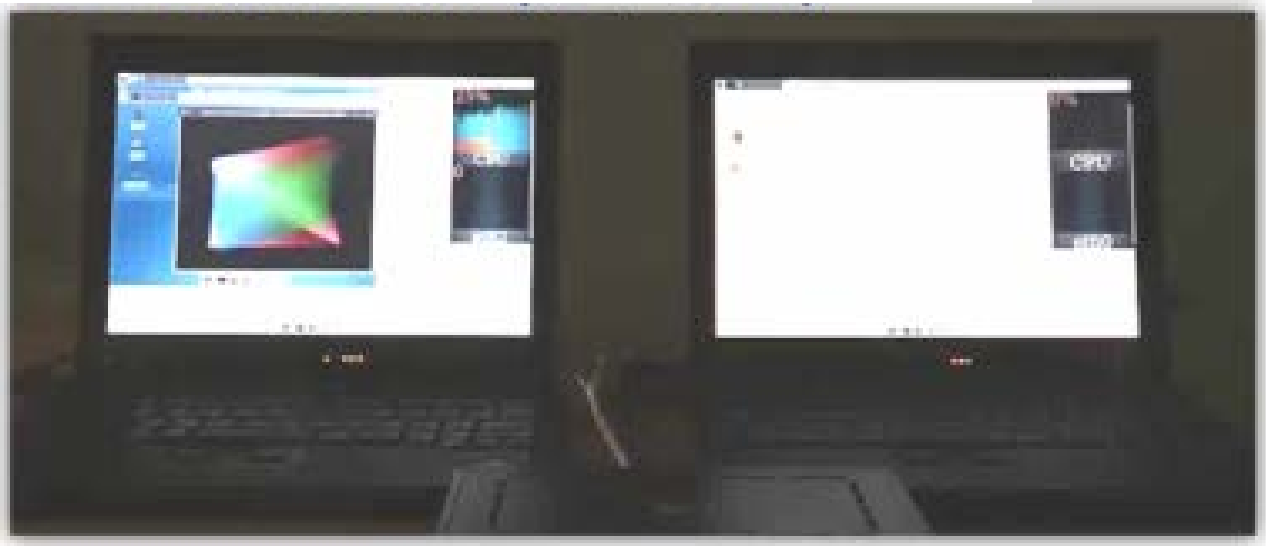
- Dynamically optimize the placement of standard VMs on ASGC, and power off excessive hardware resource
- Use live migration of VM

When the load of ASGC gets low,

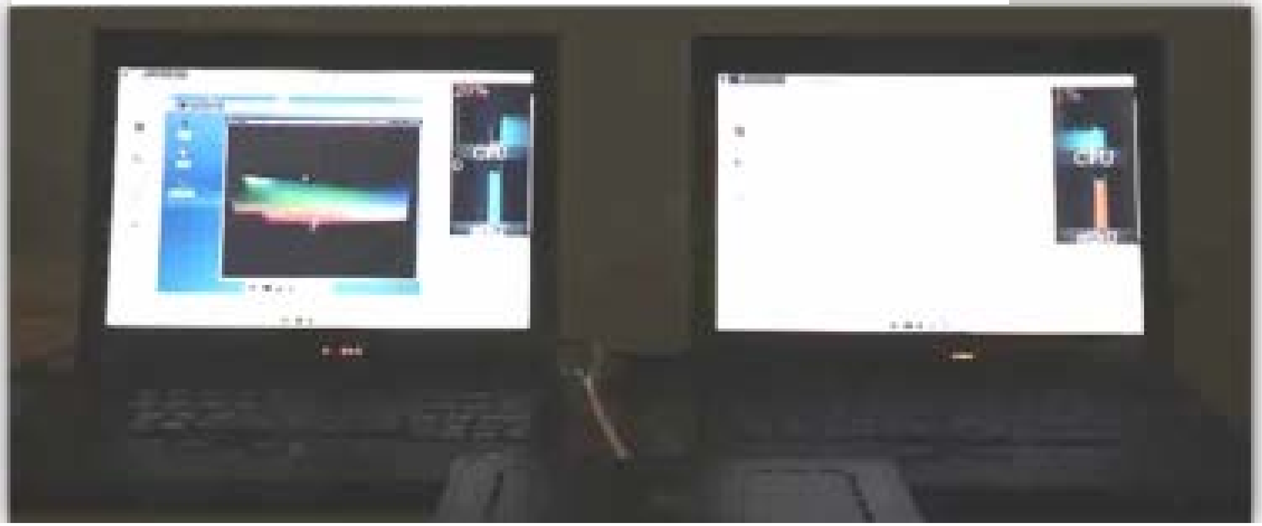
When the load of ASGC gets high,



# Quick Live Migration by AIST



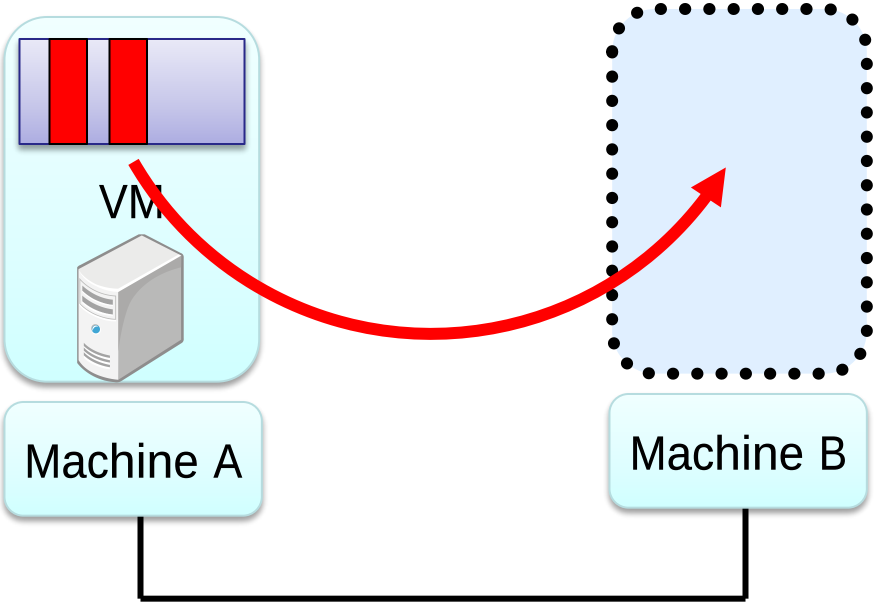

# Normal Live Migration by KVM



# Normal Live Migration

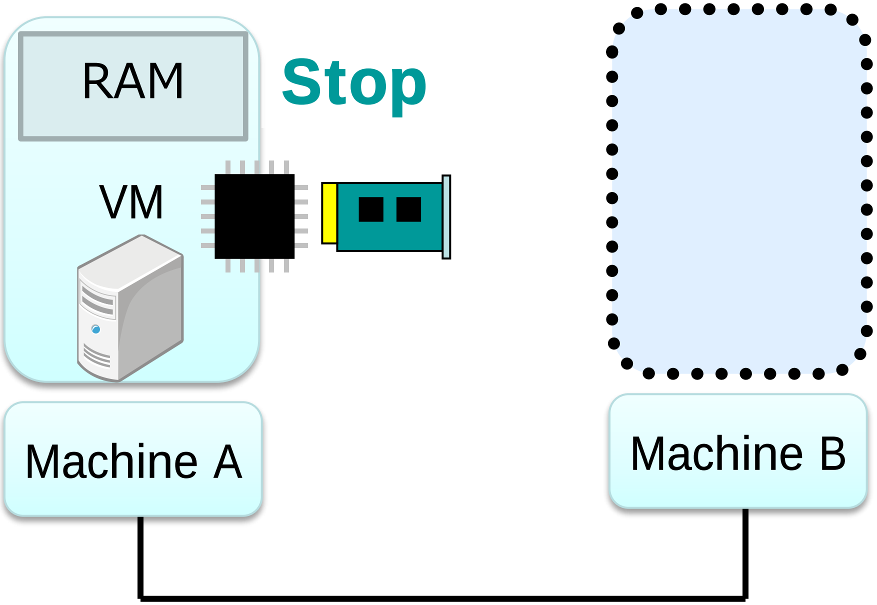
*Copy VM memory before relocation (Precopy)*

1. Copy all memory pages to destination
2. Copy memory pages updated during the previous copy again
3. Repeat the 2nd step until the rest of memory pages are enough small
4. Stop VM
5. Copy CPU registers, device states, and the rest of memory pages.
6. Resume VM at destination



# Postcopy Live Migration (1)

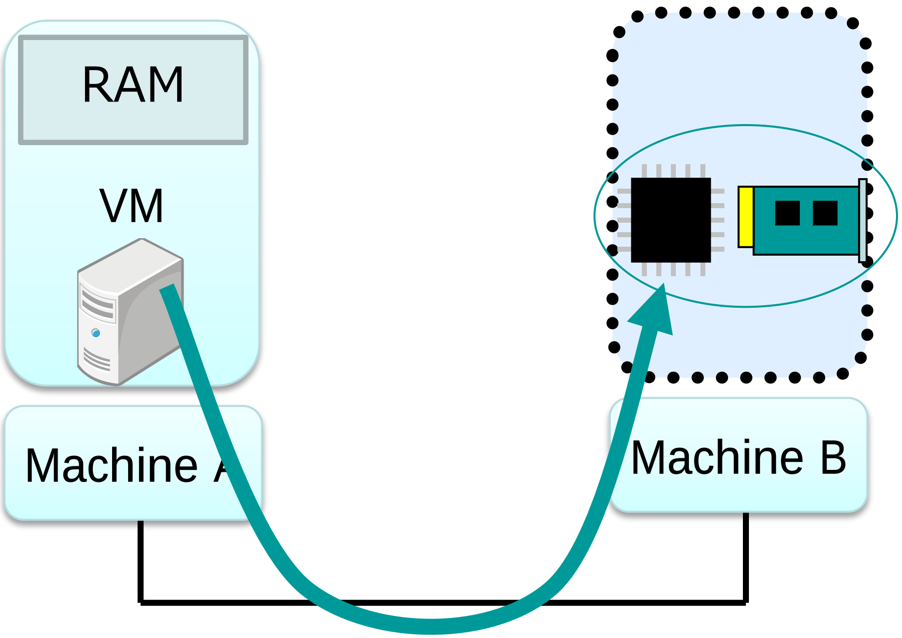
*Copy VM memory after relocation*




1. Stop VM
2. Copy CPU and device states to destination
3. Resume VM at destination
4. Copy memory pages

# Postcopy Live Migration (2)

*Copy VM memory after relocation*




1. Stop VM
2. Copy CPU and device states to destination
3. Resume VM at destination
4. Copy memory pages

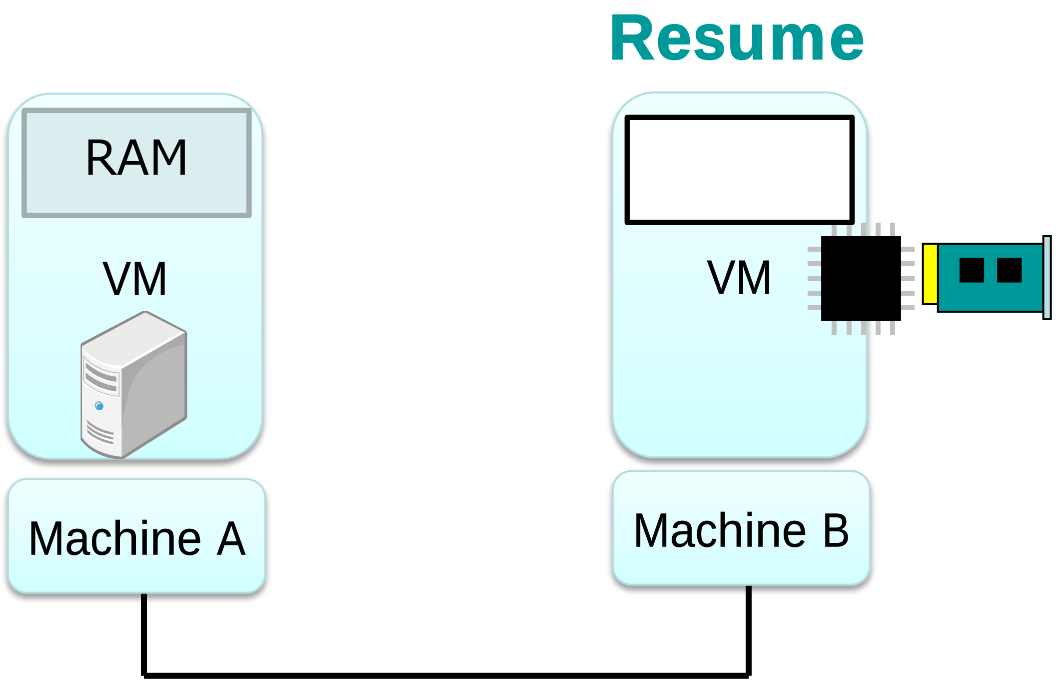
**Copy CPU and device states**

**Only 256KB w/o VGA**

**=> Less than 1 sec for relocation**

# Postcopy Live Migration (3)

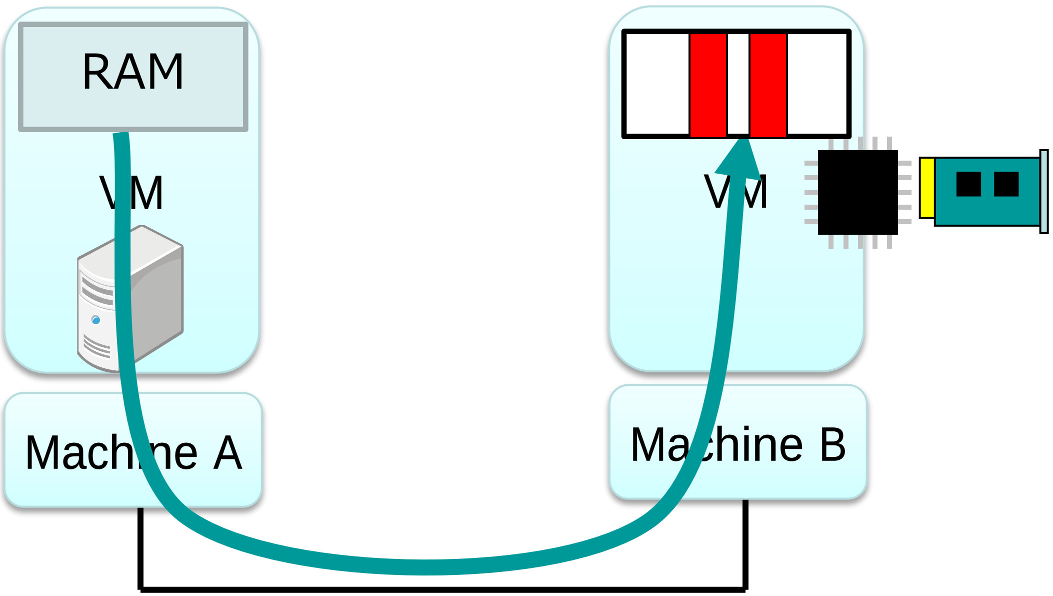
*Copy VM memory after relocation*




1. Stop VM
2. Copy CPU and device states to destination
3. Resume VM at destination
4. Copy memory pages

# Postcopy Live Migration (4)

*Copy VM memory after relocation*



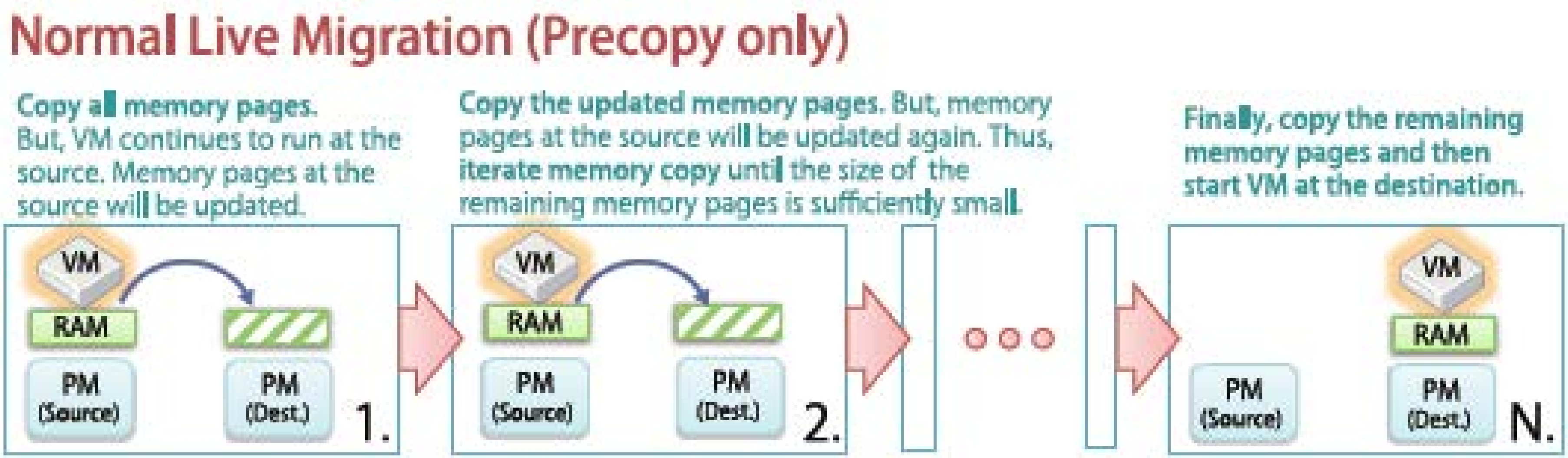

1. Stop VM
2. Copy CPU and device states to destination
3. Resume VM at destination
4. Copy memory pages

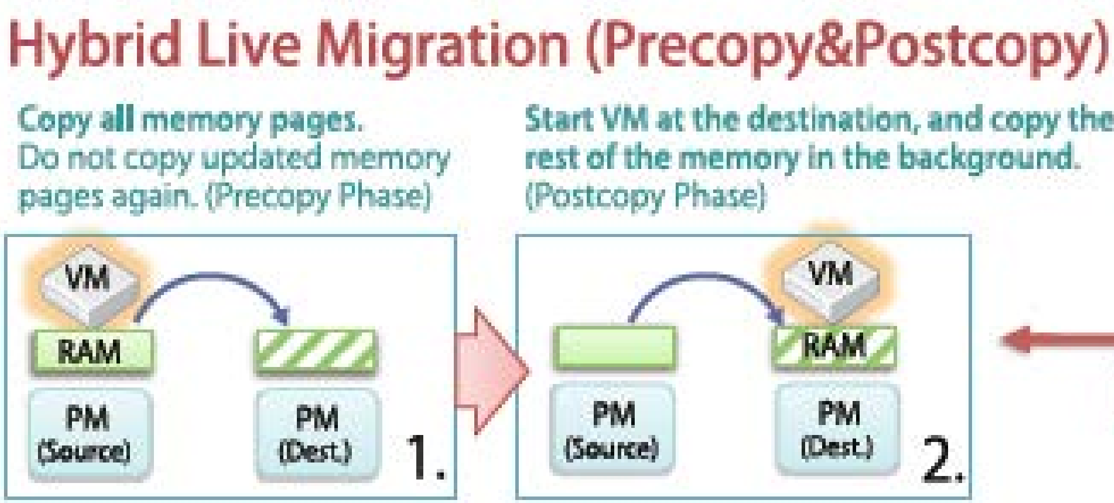
**Copy memory pages**

- **On-demand**
- **Background**

CCGrid 2012, Hirofuchi, et.al

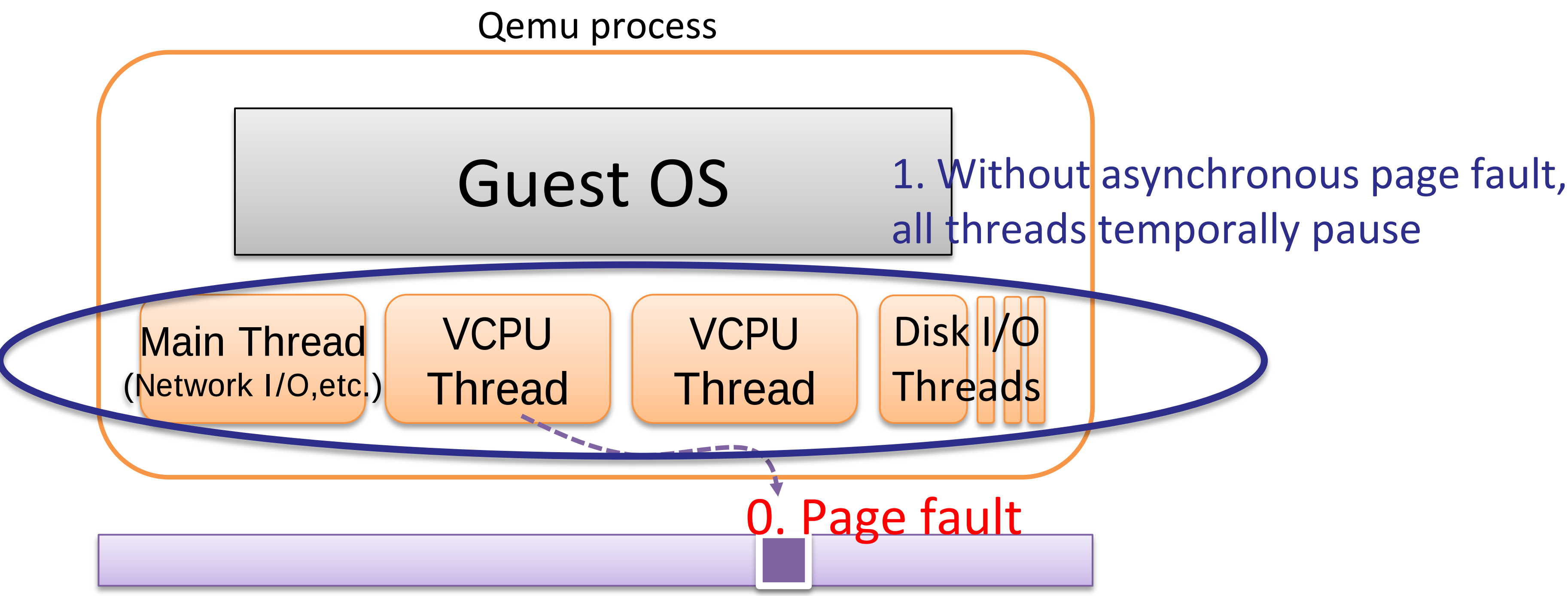
# Hybrid Live Migration for Heavy Workloads



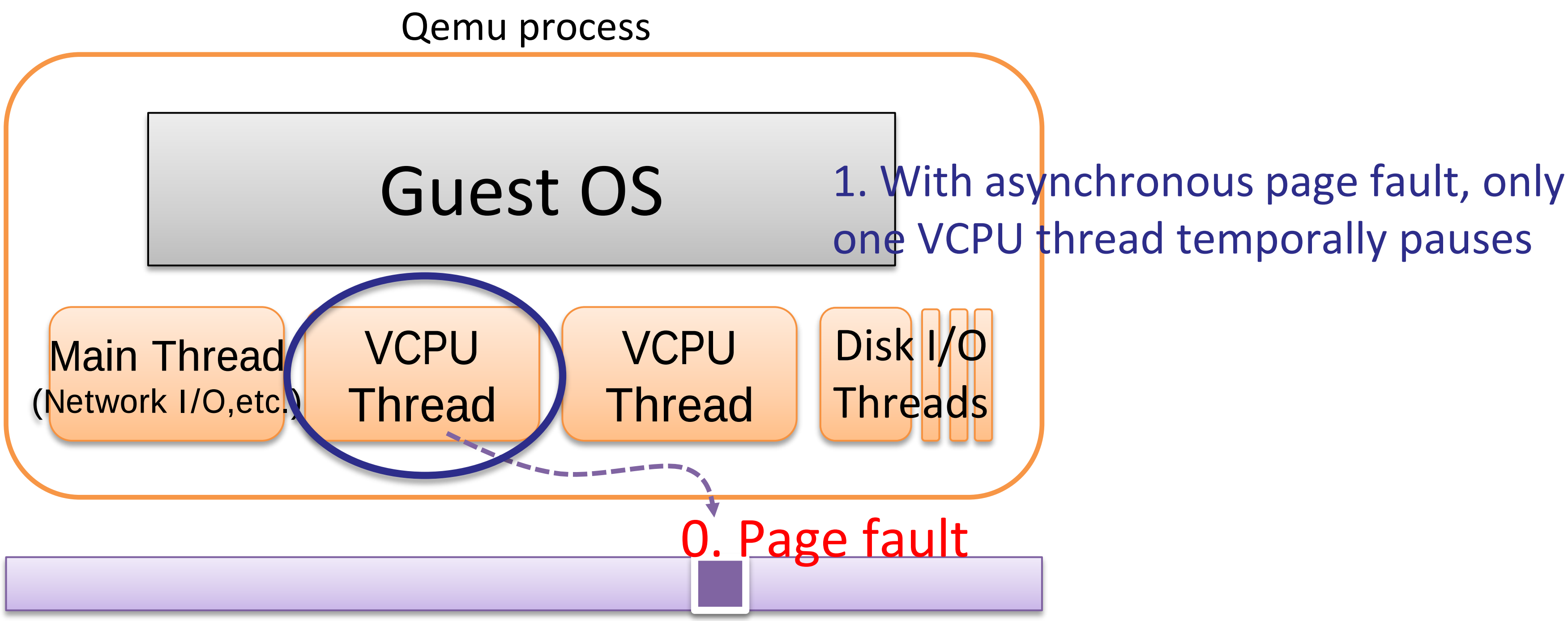

Presented in KVM Forum 2012 etc.

# Asynchronous Page Fault for Post-Copy Live Migration (1)




In the postcopy phase, if the VM accesses a not-yet-transferred memory page, the entire guest OS needs to temporally pause until that page is transferred. This causes performance degradation.

# Asynchronous Page Fault for Post-Copy Live Migration (2)

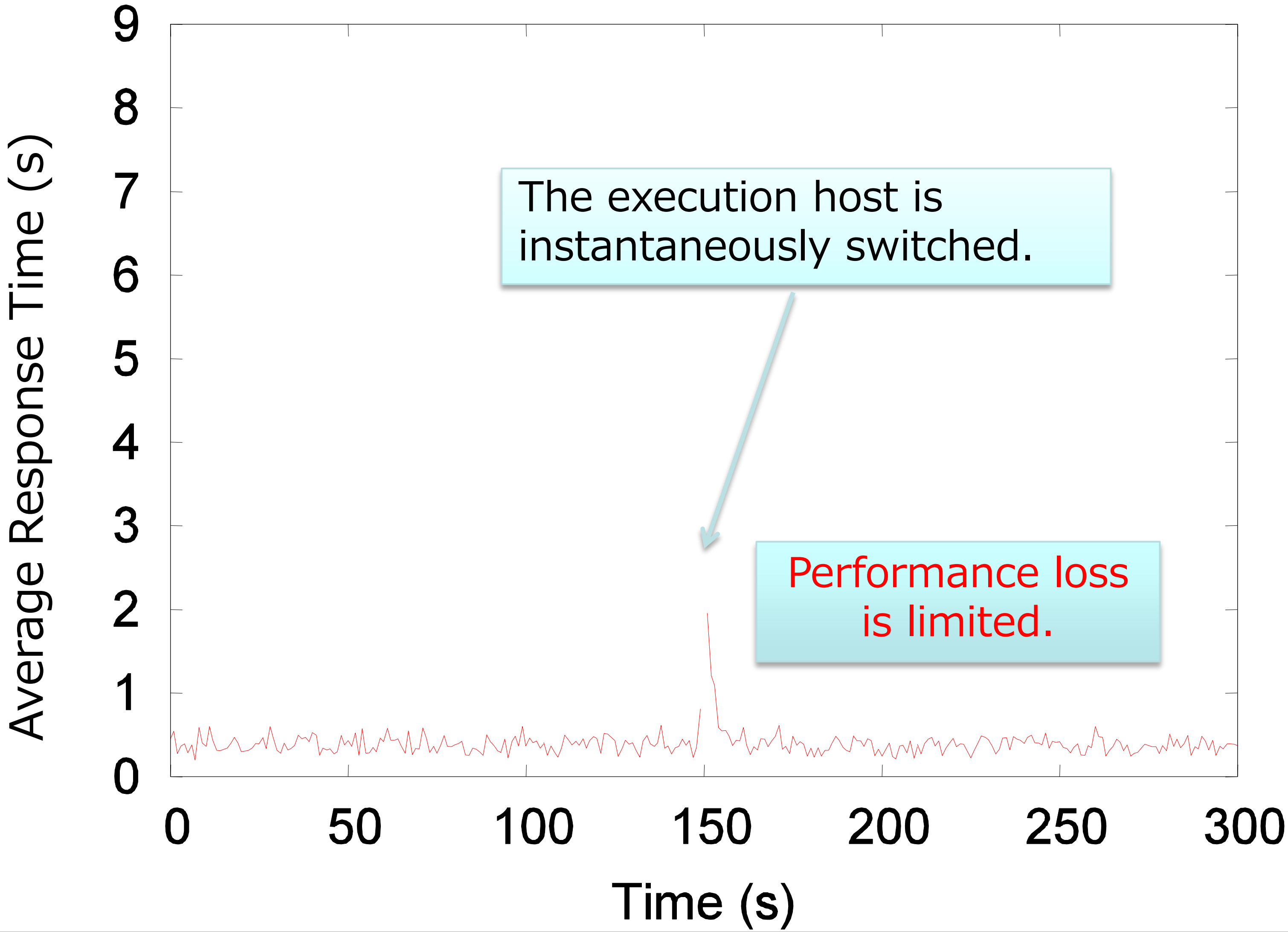



With asynchronous page fault, only the VCPU thread that caused page fault temporally pauses. The other threads keep running.

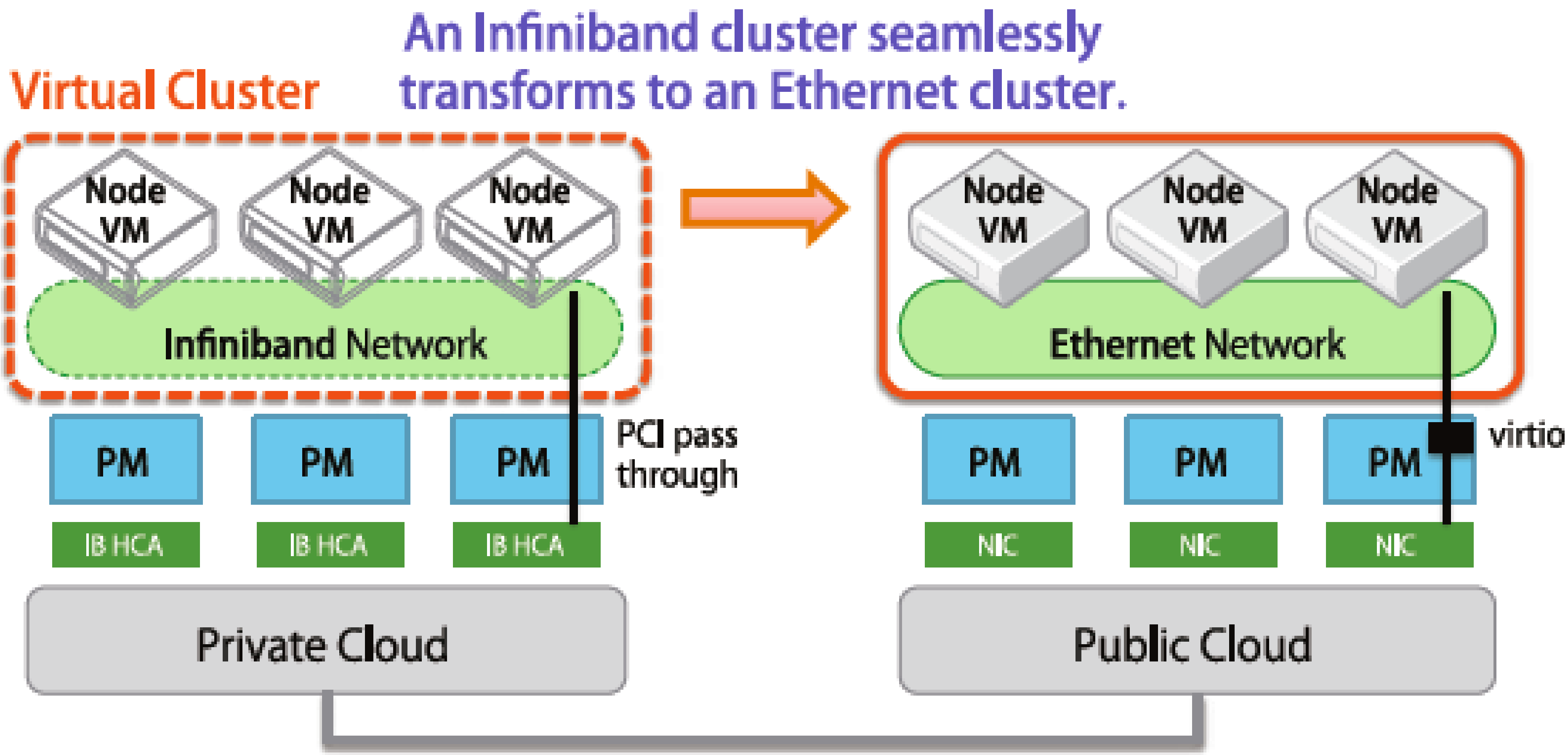
KVM Forum 2012, Yamahata, et.al

# Migrate an heavily-loaded Web Server VM

Precopy cannot migrate this VM with the default setting.



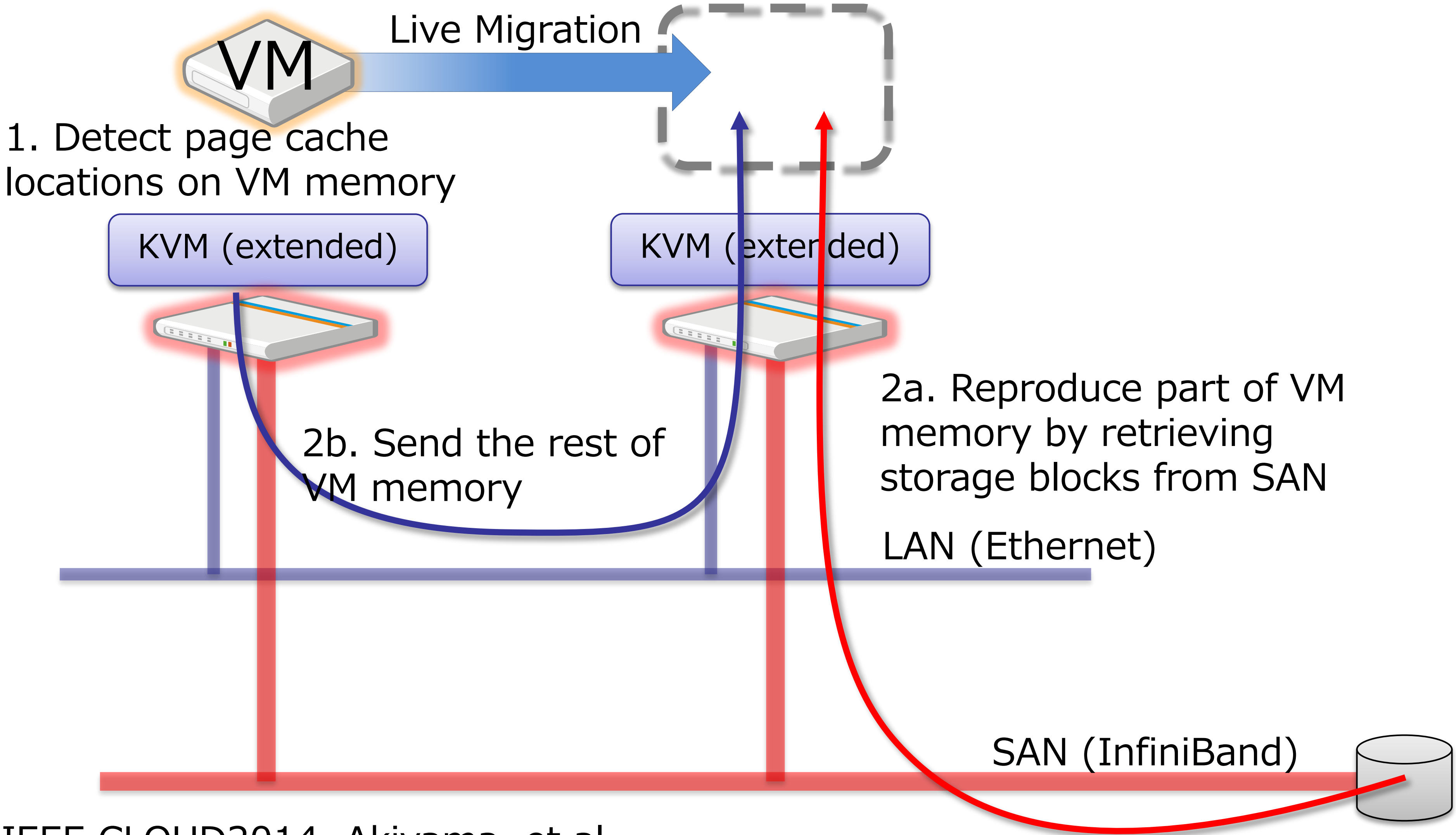
# Interconnect-agnostic Live Migration




Upon live migration, the extended KVM and MPI library hides the difference of interconnect among cloud services

IEEE eScience 2012, Takano, et.al

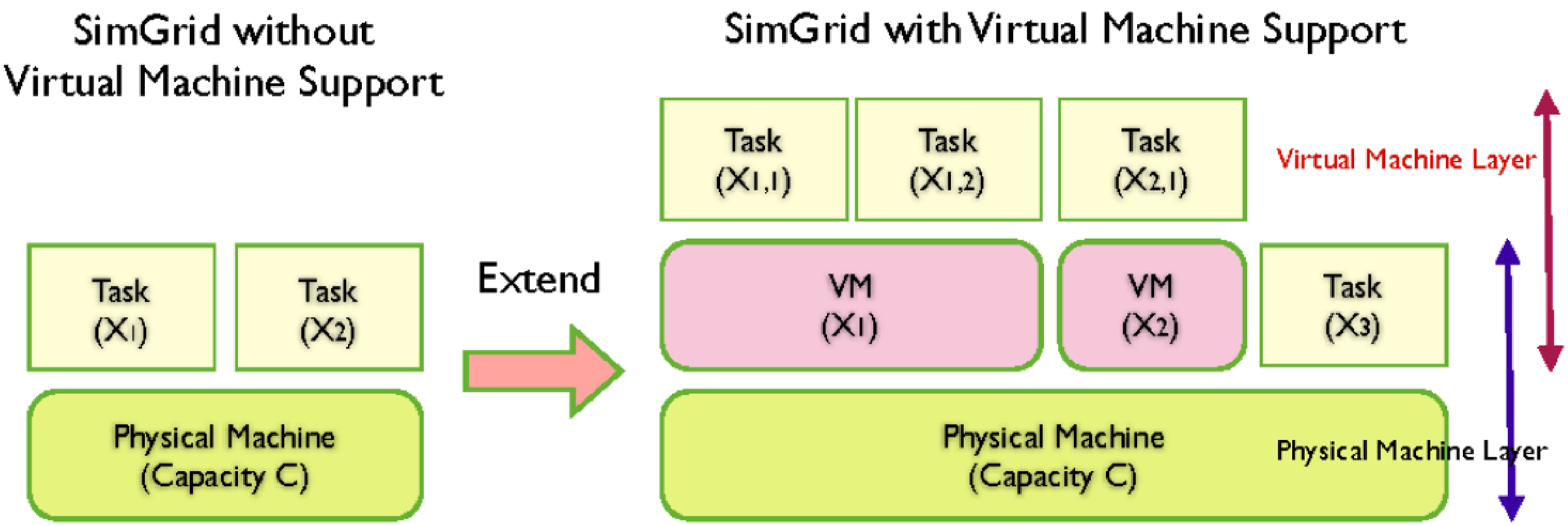
# SAN-assisted Fast Live Migration




IEEE CLOUD2014, Akiyama, et.al

# Adding Virtual Machine Model for SimGrid

(My work in Ecole des Mines of France)




1. Solve all the constraint problems at once.

   Eq1: $X_1 + X_2 < C$

1. Solve the constraint problems at the physical machine layer.

   Eq1: $X_1 + X_2 + X_3 < C$

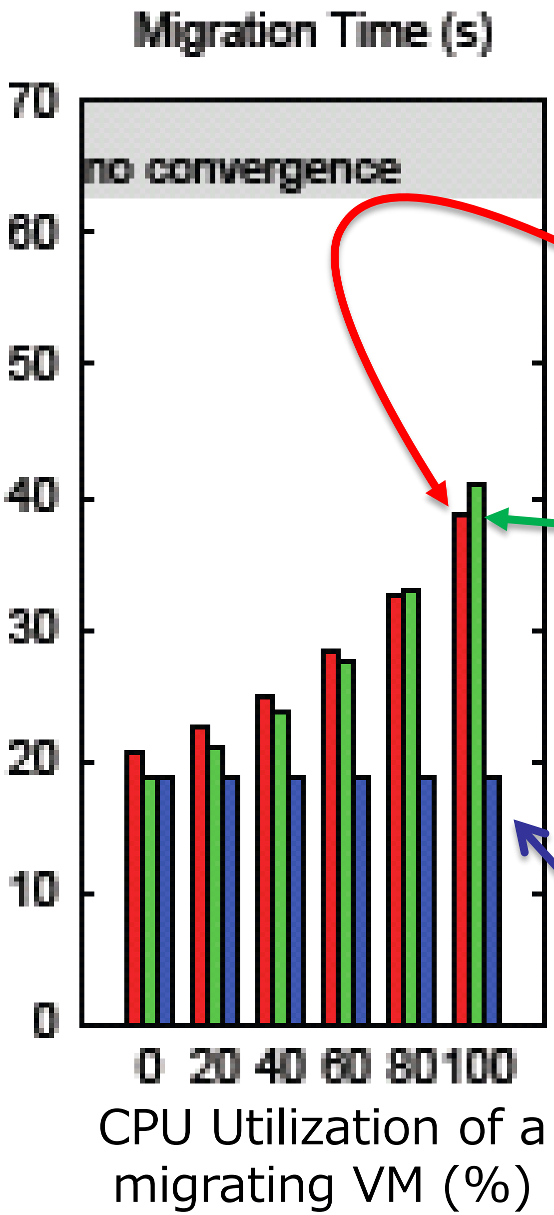
2. Solve the constraint problems at the virtual machine layer.

   Eq2: $X_{1,1} + X_{1,2} < X_1$

   Eq3: $X_{2,1} < X_2$

IEEE CloudCom2013, Hirofuchi, et.al

Acknowledgement: French Research Project (ANR-SONGS)




Migration time in reality

**The proposed migration model accurately estimates live migration time in a simulation.**

An existing migration model under-estimates migration time for an actively-running VM.

# Conclusion

- AIST Super Green Cloud (ASGC)
  - IaaS platform built on a 155-nodes cluster
  - Support HPC and non-HPC applications in AIST
  - Apache CloudStack & Qemu/KVM
    - Provide only VMs, but achieve high performance
    - Extension by AIST
      - Support PCI Pass-through & SR-IOV
      - VM image export to Amazon EC2 and others
      - Dynamic VM consolidation with our migration technologies
- Current status and future work
  - Just started service since July 2014
  - Save energy by dynamic consolidation