

DISCO

Quiz Time

1. Disco is a ____ ?
 - a) Hardware Architecture
 - b) Operating System
 - c) Virtual Machine
 - d) Virtual Machine Manager

Quiz Time

2. How does Disco virtualize memory? What would be held in TLB? What happens on a TLB miss w/o and w/ Disco? What data structure does Disco add?

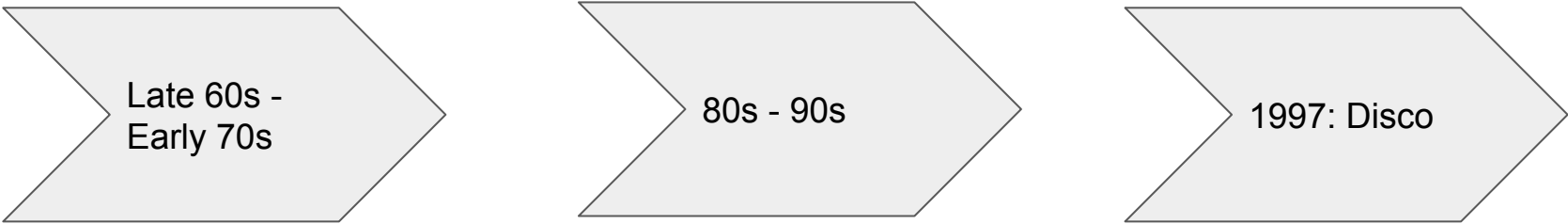
Fun Fact

Disco was published in 1997

VMWare was founded in 1998 by authors of Disco, now one of the biggest virtualization and cloud computing software providers

Timeline

https://en.wikipedia.org/wiki/Timeline_of_virtualization_development



Late 60s -
Early 70s

80s - 90s

1997: Disco

- Hardware was expensive
 - Multiplex an expensive machine
- IBM initiated VM idea to support legacy binary code
 - Support an old machine's on a newer machine
- Hardware got cheaper
 - Could deploy new OS on different machine
- OS got powerful
 - no need to use VMs to provide multi-user support
- People's attentions were drawn back to VMs

Motivation

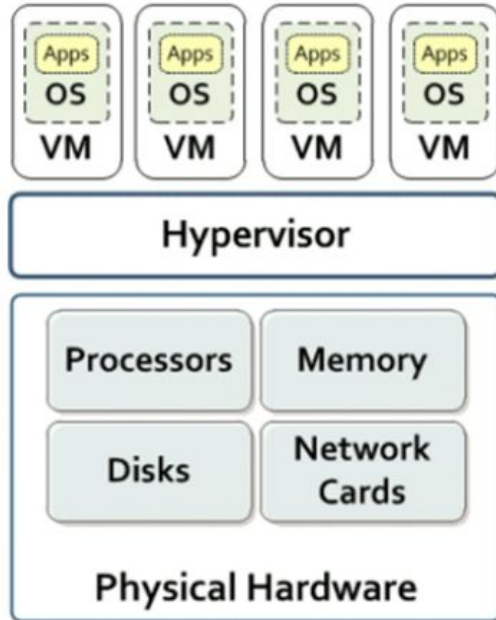
- Innovative Hardware in the market
 - Existing commodity OSes are not suited for a scalable multiprocessor.
- Gap between hardware innovations and the adaptation of system software
 - Hardware development faster than system software
 - System software for these machines has often trailed hardware in reaching the functionality and reliability expected by modern computer users.
- Customized OS are late, incompatible, unscalable, and possibly buggy
 - OS cannot leverage the potential of hardware
 - Extensive modifications to OS are required to efficiently support scalable machines
 - Do not scale due to unawareness of system architecture
 - Error Prone

Why existing OS not scalable?

Which one is Disco's architecture?

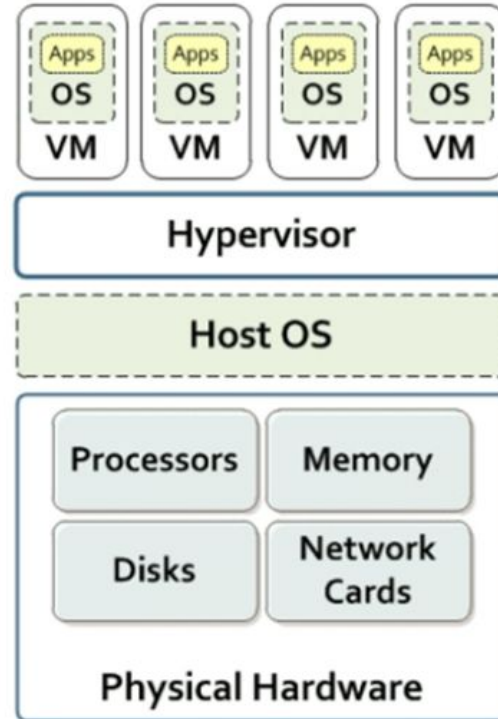
2 Types of VMMs

native / bare-metal
hypervisors



Type 1

hosted
hypervisors



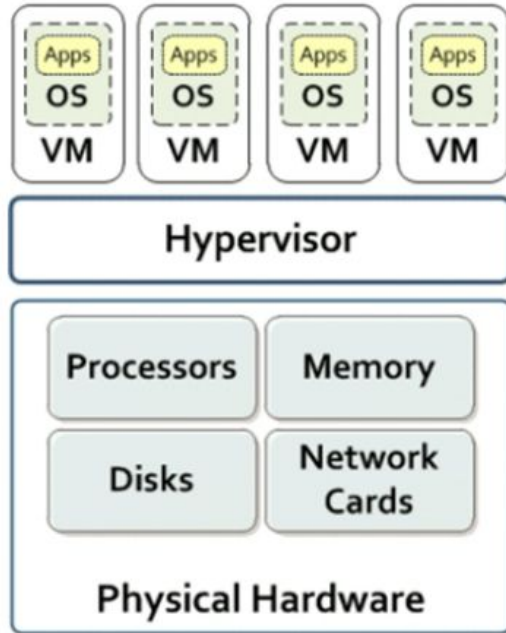
Type 2

Pros & Cons of these two types

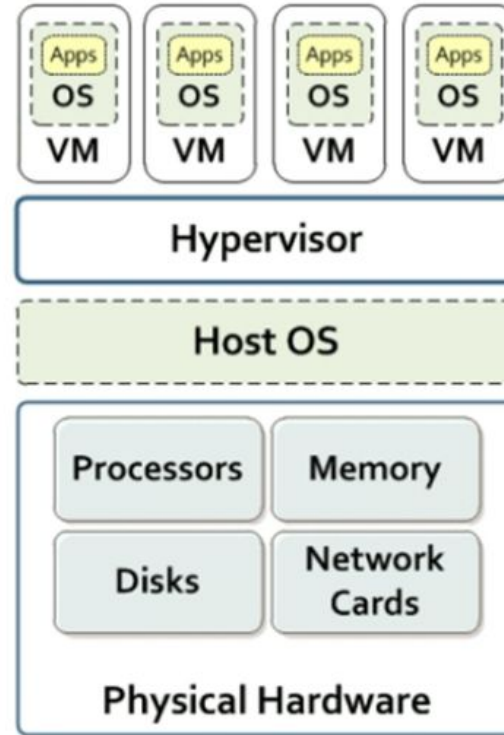
2 Types of VMMs

Eg:

IBM VM/370,
Disco,
VMware's ESXi,
Xen



Type 1



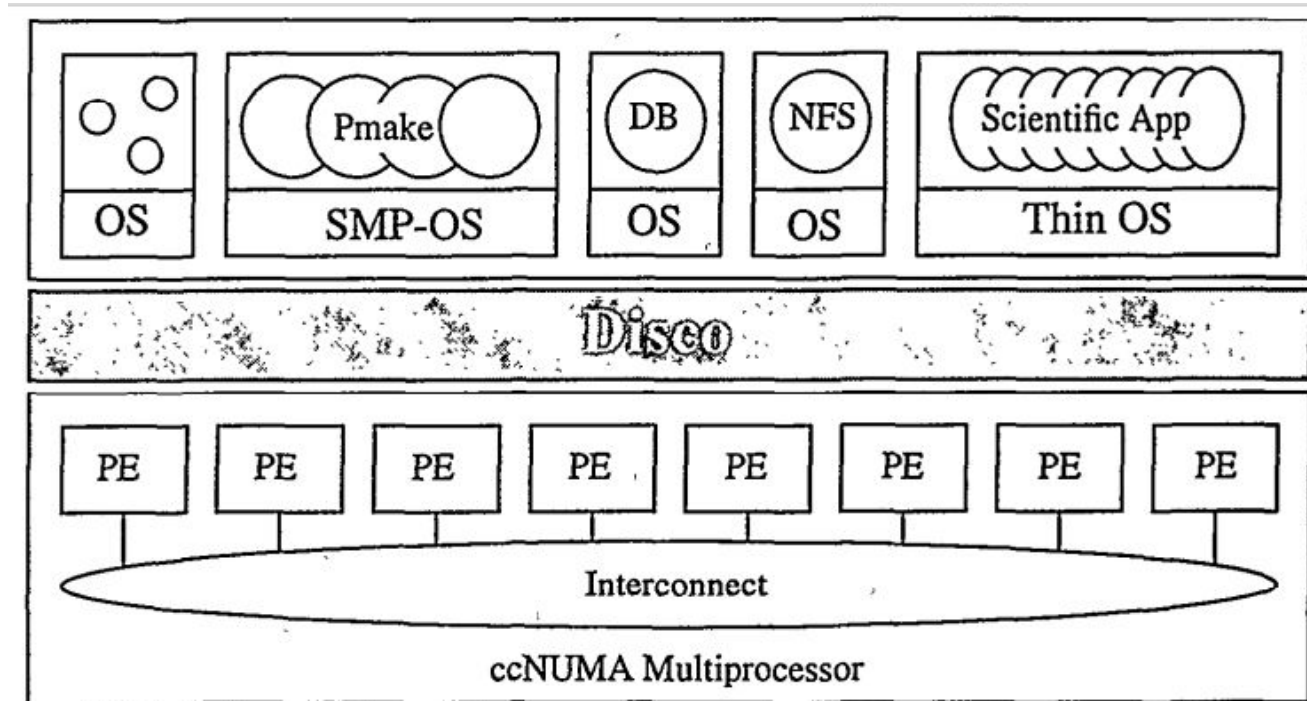
Type 2

Eg:

VirtualBox
VMWare
Workstation,
VMWare Server,
Kernel-Based
Virtual Machine
(KVM)

Disco's Idea

Virtual Machine Monitor between hardware and VM running commercial OS



Disco's Goal

Run multiple commodity operating systems on a scalable multiprocessor (large-scale shared-memory multiprocessors), with a small implementation effort

Trading off between performance and development costs

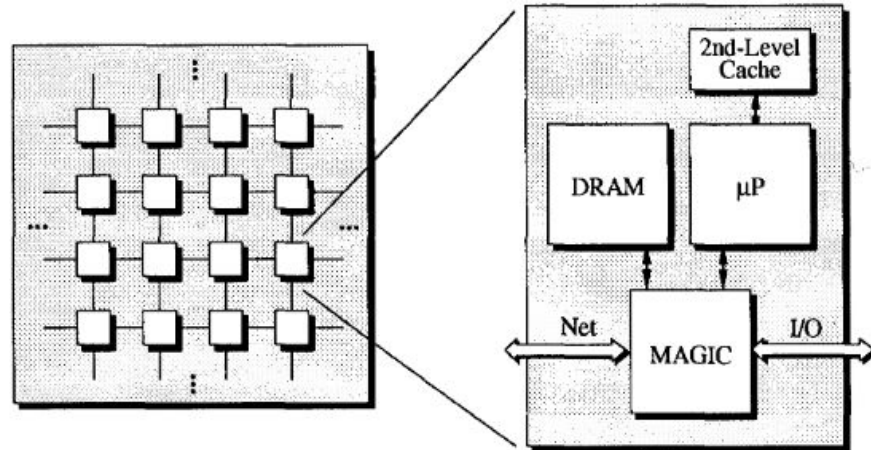
make multiple resources appear like a single resource

hides NUMA-ness from non-NUMA aware OSes

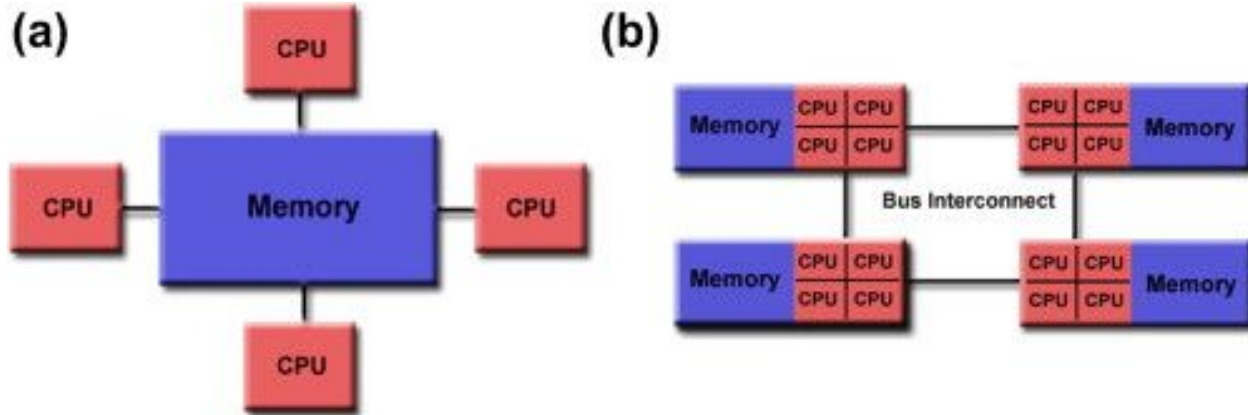
The Stanford FLASH Multiprocessor

The FLASH multiprocessor consists of a collection of nodes each containing a processor, main memory, and I/O devices.

Each Node has its own RAM and MMU associated. With ccNUMA systems, all memory is visible to and accessible from any node.



UMA VS. NUMA

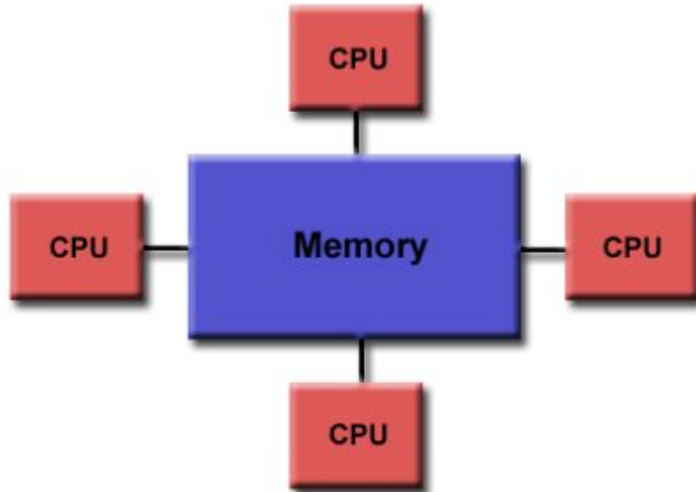


Which one is the architecture used in your laptop?

UMA / SMP (symmetric multiprocessors)

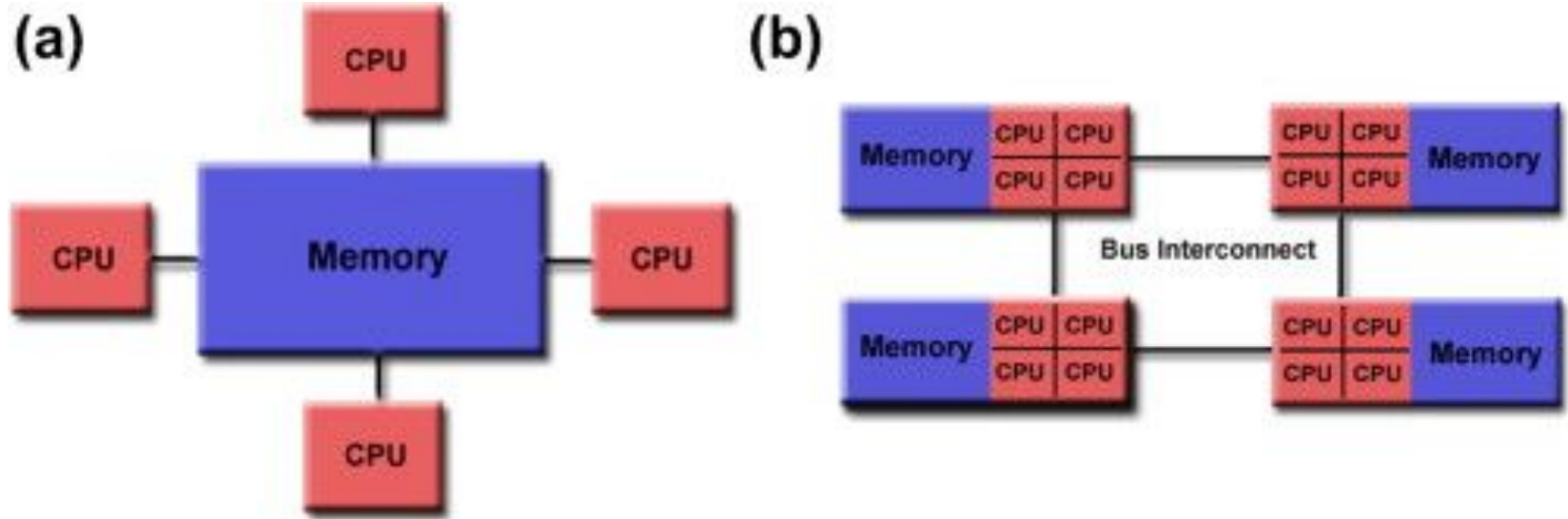
Each processor accesses the **same** physical memory:

Uniform Memory Access (UMA)



- UMA typically features few cores. E.g., your laptop or desktop at home.
- This allows for hardware implementations with symmetric access to memory

Evolution from UMA to NUMA

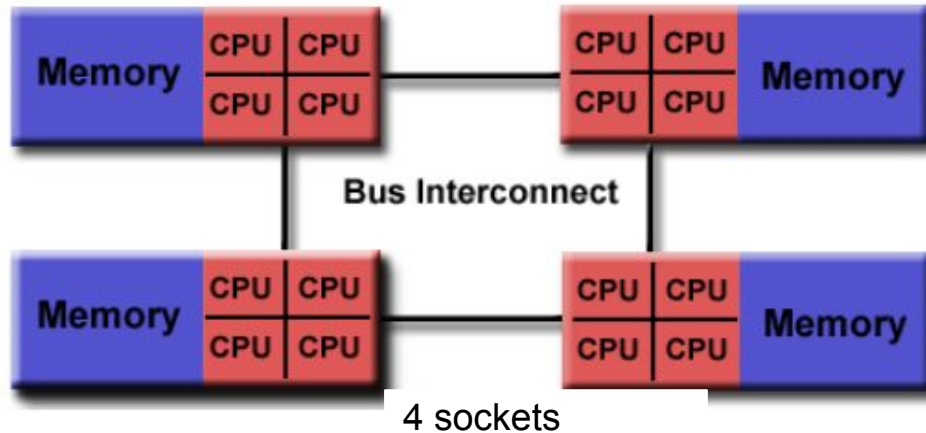


What drives Mainframe computers (or even Supercomputers) switch from UMA to NUMA pattern?

NUMA / DSM (Distributed Shared Memory)

A set of processors have direct access to their **own** physical memory:

Non-Uniform Memory Access (NUMA)



- Support large processor counts requires distributed shared memory.
- Have multiple sockets (separate units) to release the pressure on the bus. (avoid high mem access latencies)


```
[yw118@bc10u19n4 workplace]$ numactl --hardware
```

```
available: 2 nodes (0-1)
```

```
node 0 cpus: 0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46
```

```
node 0 size: 96965 MB
```

```
node 0 free: 81636 MB
```

```
node 1 cpus: 1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31 33 35 37 39 41 43 45 47
```

```
node 1 size: 98304 MB
```

```
node 1 free: 92031 MB
```

```
node distances:
```

```
node    0    1
```

```
  0:   10   21
```

```
  1:   21   10
```

How to Build a VMM?

Disco exports a more conventional hardware interface, so that each virtual machine can run a standard OS that manages its virtualized resources independently.

What should be inside such a hardware interface for a general purpose computer?

Disco's Interface

- Processors
 - emulation of a MIPS R10000 processor, optimized for trap emulation.
- Physical Memory
 - an abstraction of main memory
- I/O Devices
 - disks, network interfaces, periodic interrupt timers, clock, and a console. Disco must intercept all communication to and from I/O devices to translate or emulate the operation.

Disco Implementation - Virtual CPUs

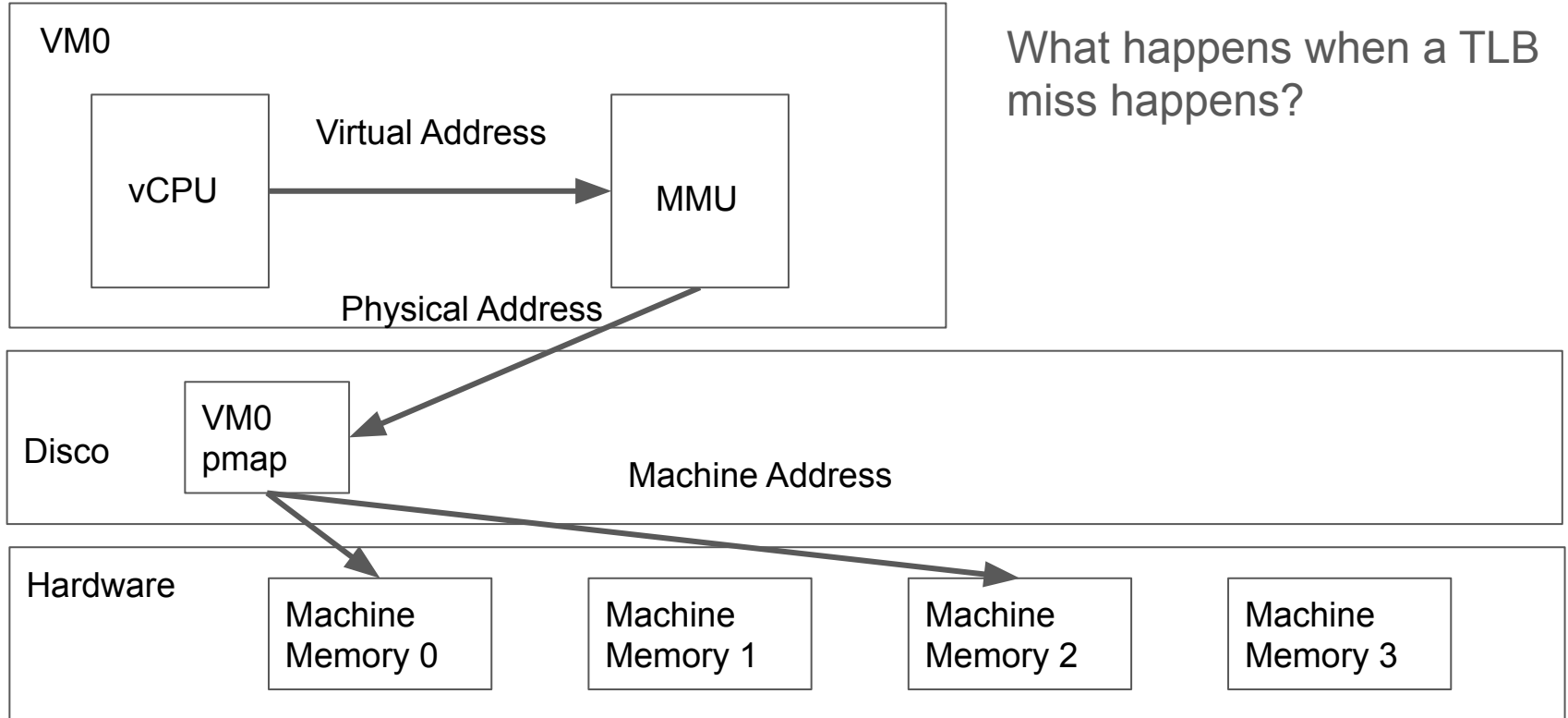
- Register File
- PC Reg

- Disco runs most of instructions directly on the hardware to achieve reasonable performance. Is there any Security Issues?

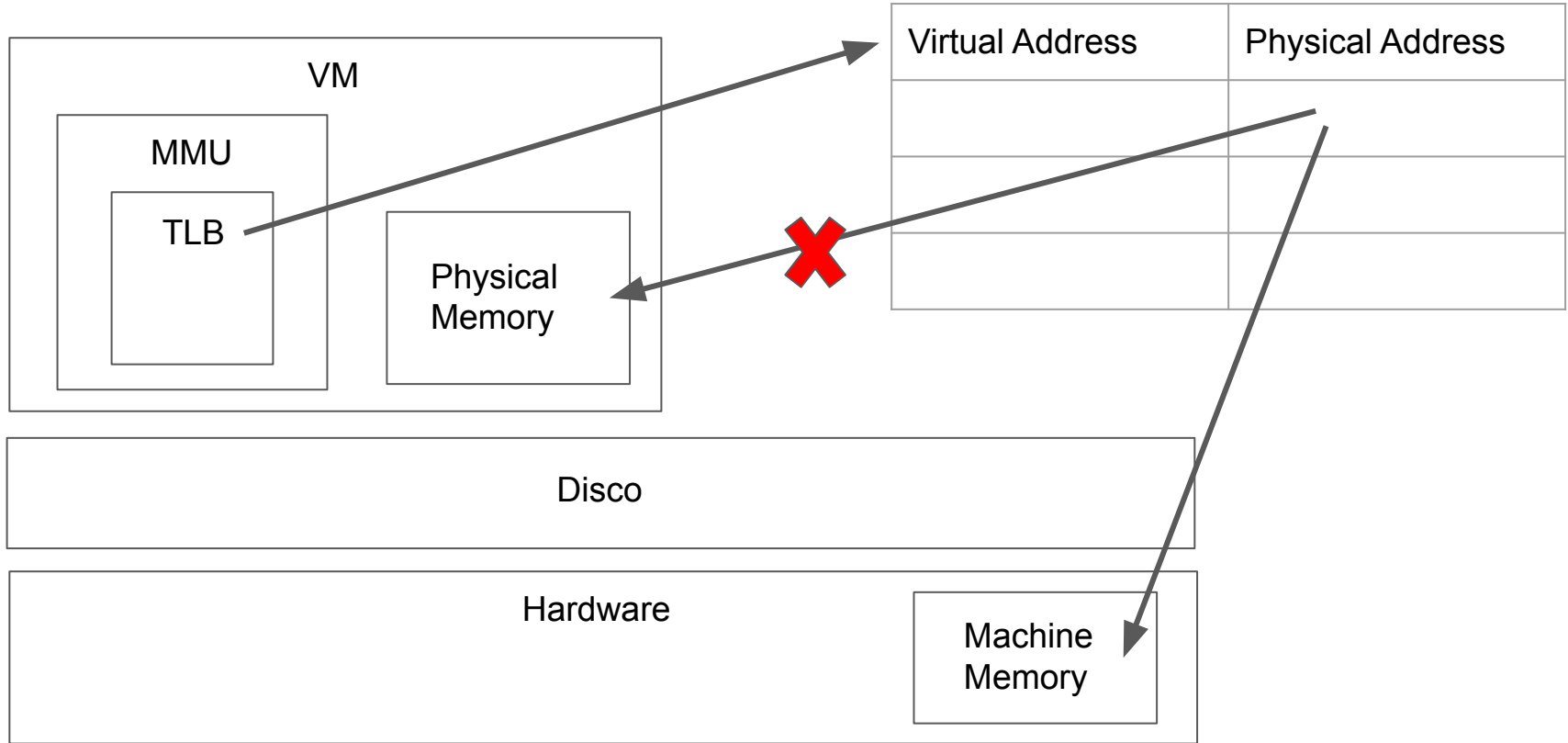
Disco Implementation - Virtual CPUs

- Disco runs in kernel mode
- All OS code runs in supervisor mode which does not allow execution of privileged instructions.
- Attempts to execute privileged instructions trap to Disco. Disco executes them on behalf of the OS, limiting access to that OS's VM resources

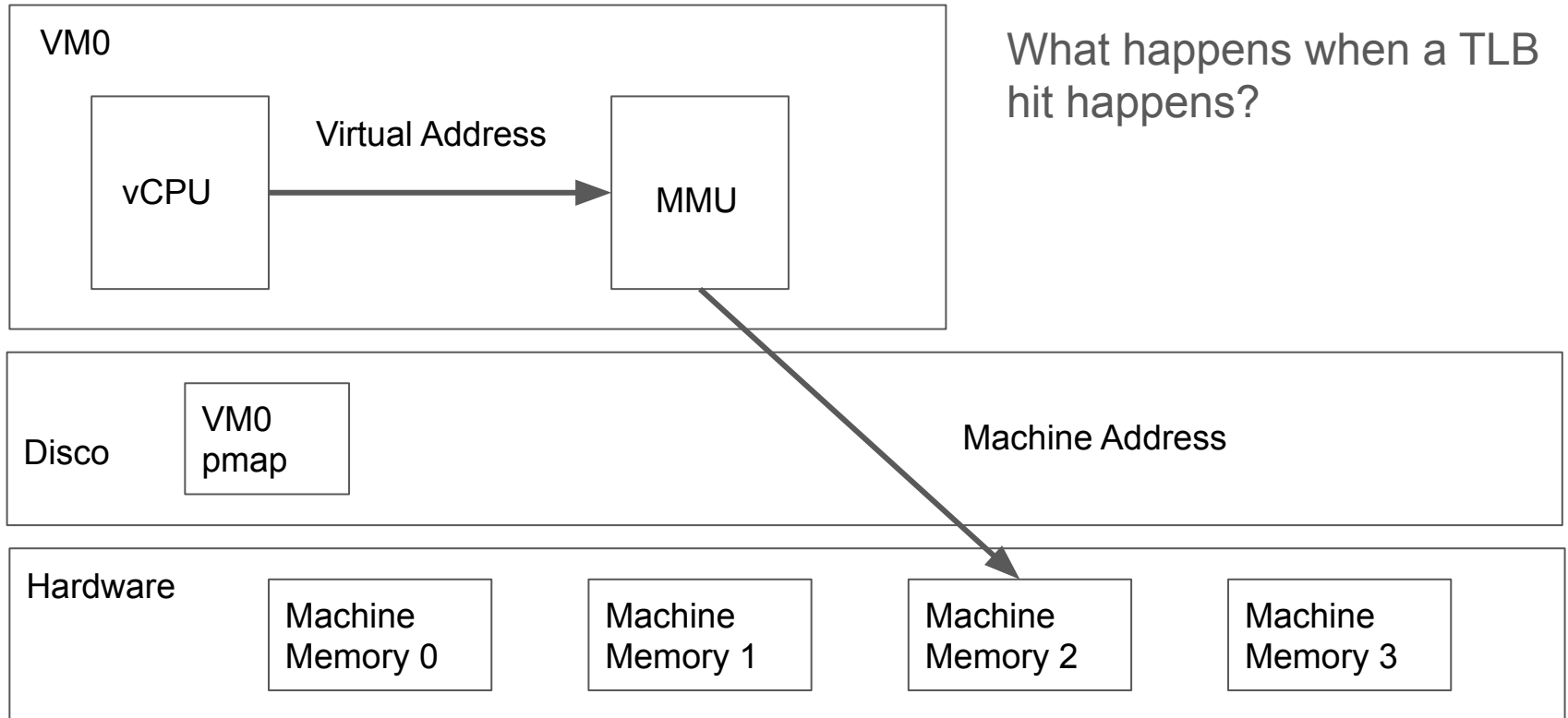
Disco Implementation - Memory Translation



Disco Implementation - Memory Translation



Disco Implementation - Memory Translation

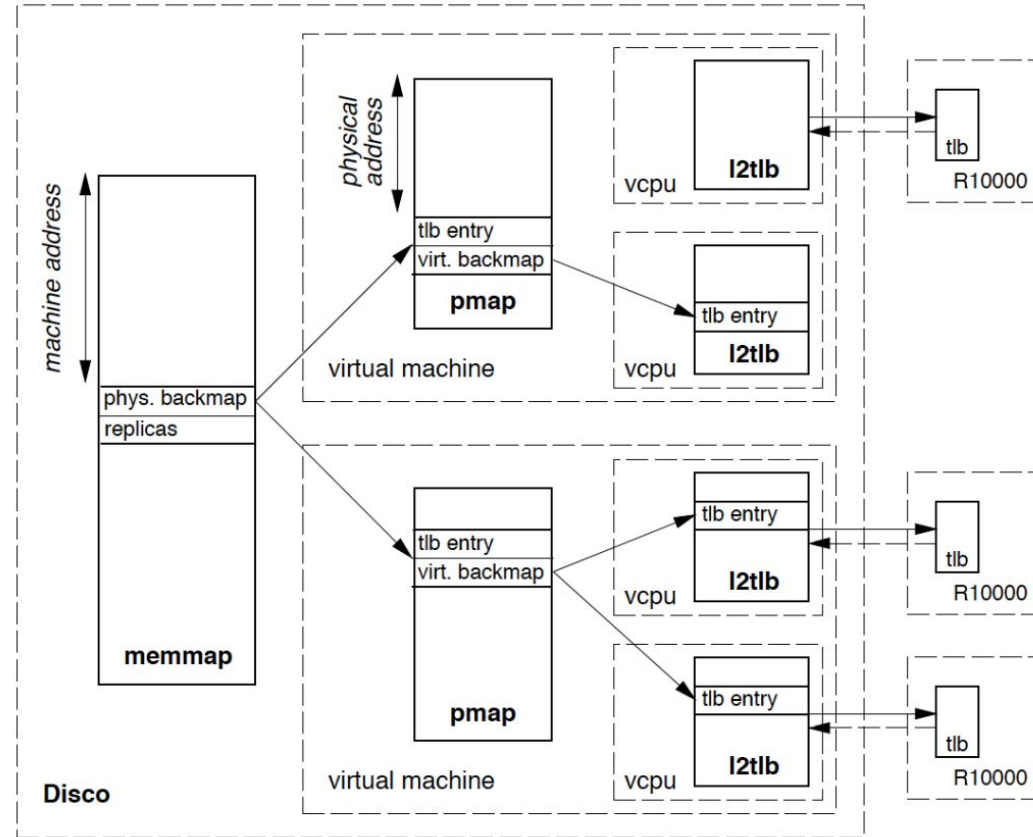


Disco Implementation - Virtual Physical Memory

vCPU Switch

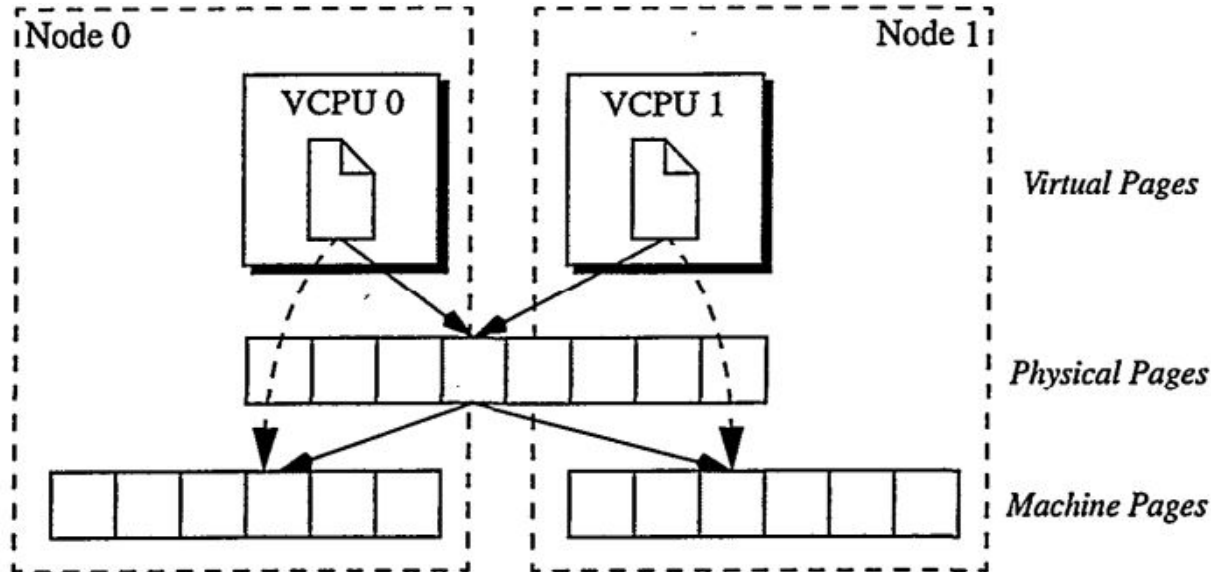
-> Flush TLB

-> secondary page table



Disco Implementation - NUMA Memory Management

dynamic page migration and page replication
to maintain locality



Evaluations

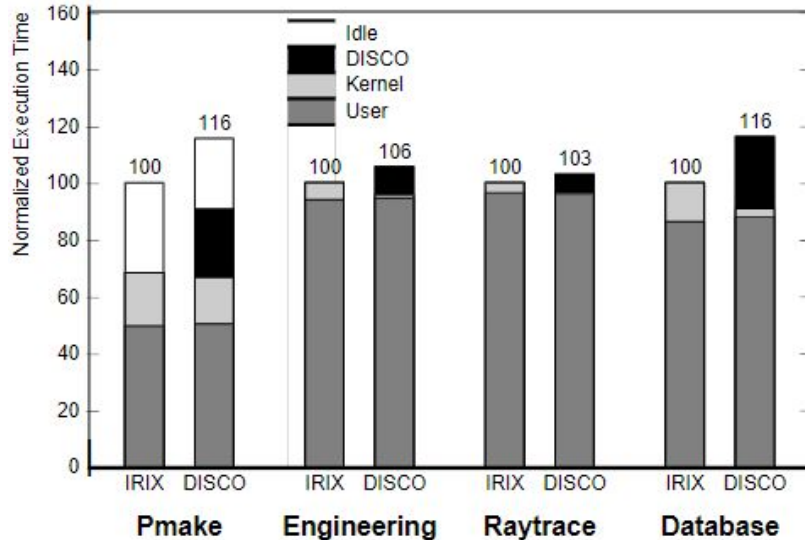


FIGURE 5. Overhead of Virtualization. The figure compares, for four uniprocessor workloads, the execution time when running IRIX directly on the simulated hardware with IRIX running in a Disco virtual machine. The execution time is separated between the time spent in user programs, the IRIX kernel, Disco, and the idle loop.

- Experimented on a uni-processor, once running IRIX directly on the h/w and once using disco running IRIX in a single virtual machine.
- Overhead of virtualization ranges from 3% - 16%.

Contributions

- VMWare
- Start of modern research on VM
- Novel way of hiding NUMA-ness