KVM Forum 2008
Nested paging hardware and software

Benjamin Serebrin
Jörg Rödel

Advanced Micro Devices

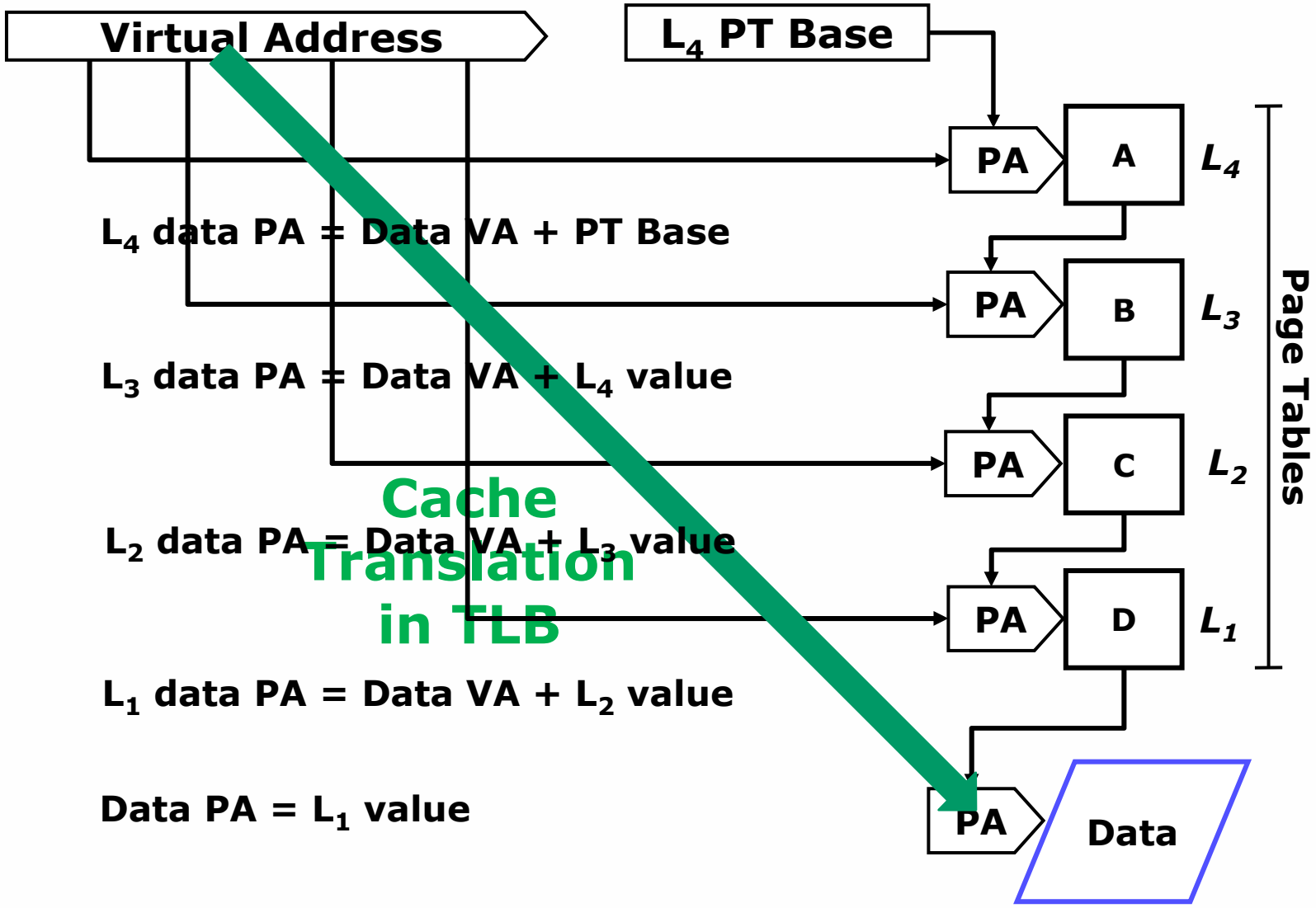
June 13, 2008

Outline

AMD
Smarter Choice

1. Background
   - AMD64 Page Walks and Caching
   - Virtualization Terminology
   - Memory Management in Virtualized Systems
2. Two-Dimensional Page Walks
   - Nested Paging + Current Paging = 2D Page Walk
   - 2D Page Walk Caching
   - Hardware and Software 2D Page Walk Acceleration
3. KVM Implementation and Results
   - KVM Software Implementation
   - Results

Four-Level AMD64 Page Walk

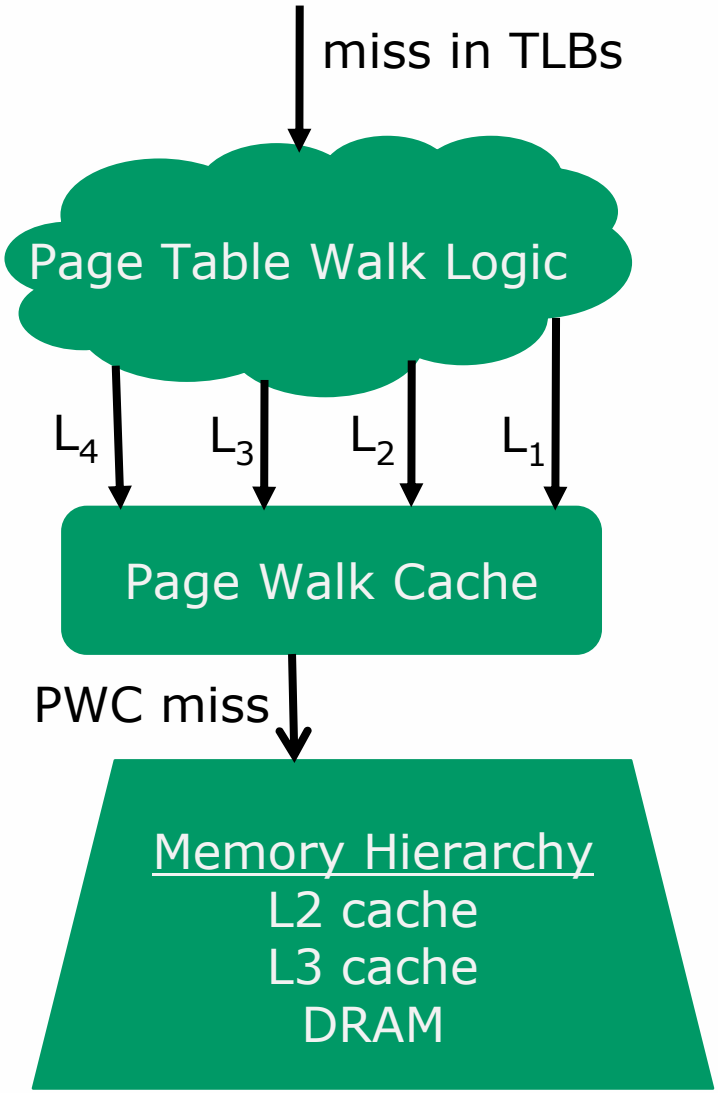


AMD64 Processor Page Walk Caching

AMD
Smarter Choice

Page Walk Cache (PWC)

- In all generations of AMD64 processors
- Stores intermediate page table values
- Low-latency access

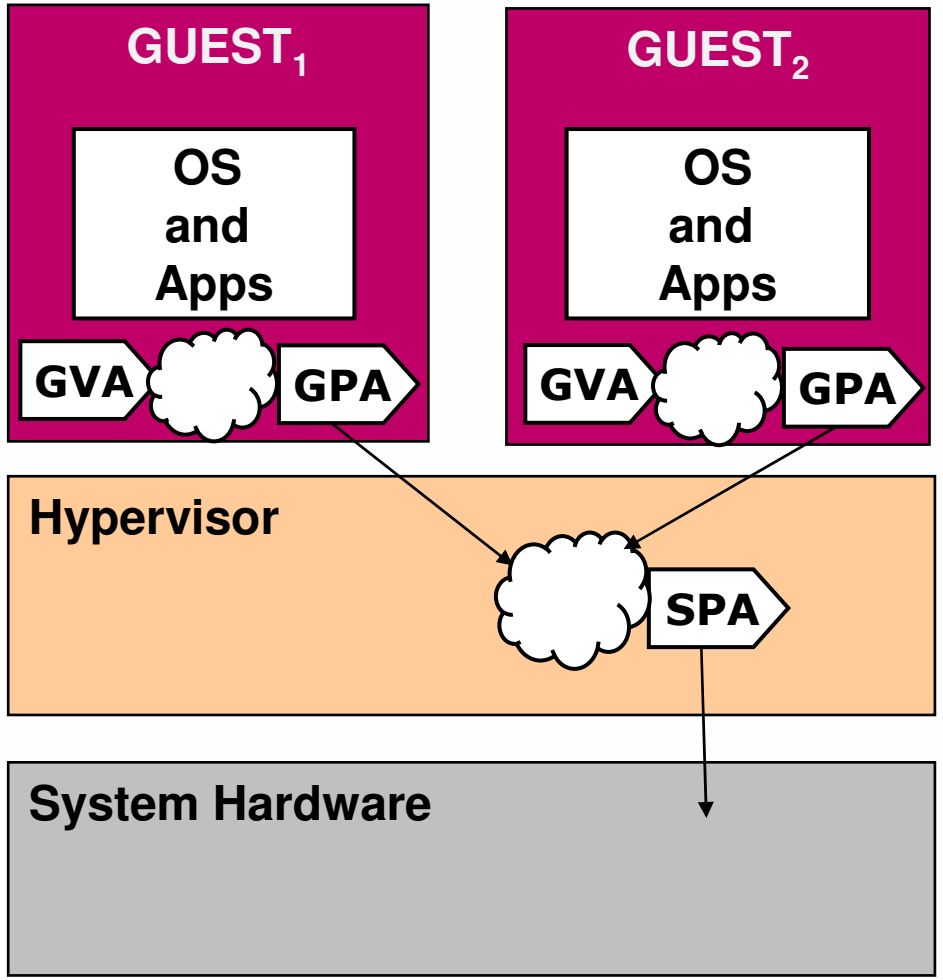


Address Terminology

AMD
Smarter Choice

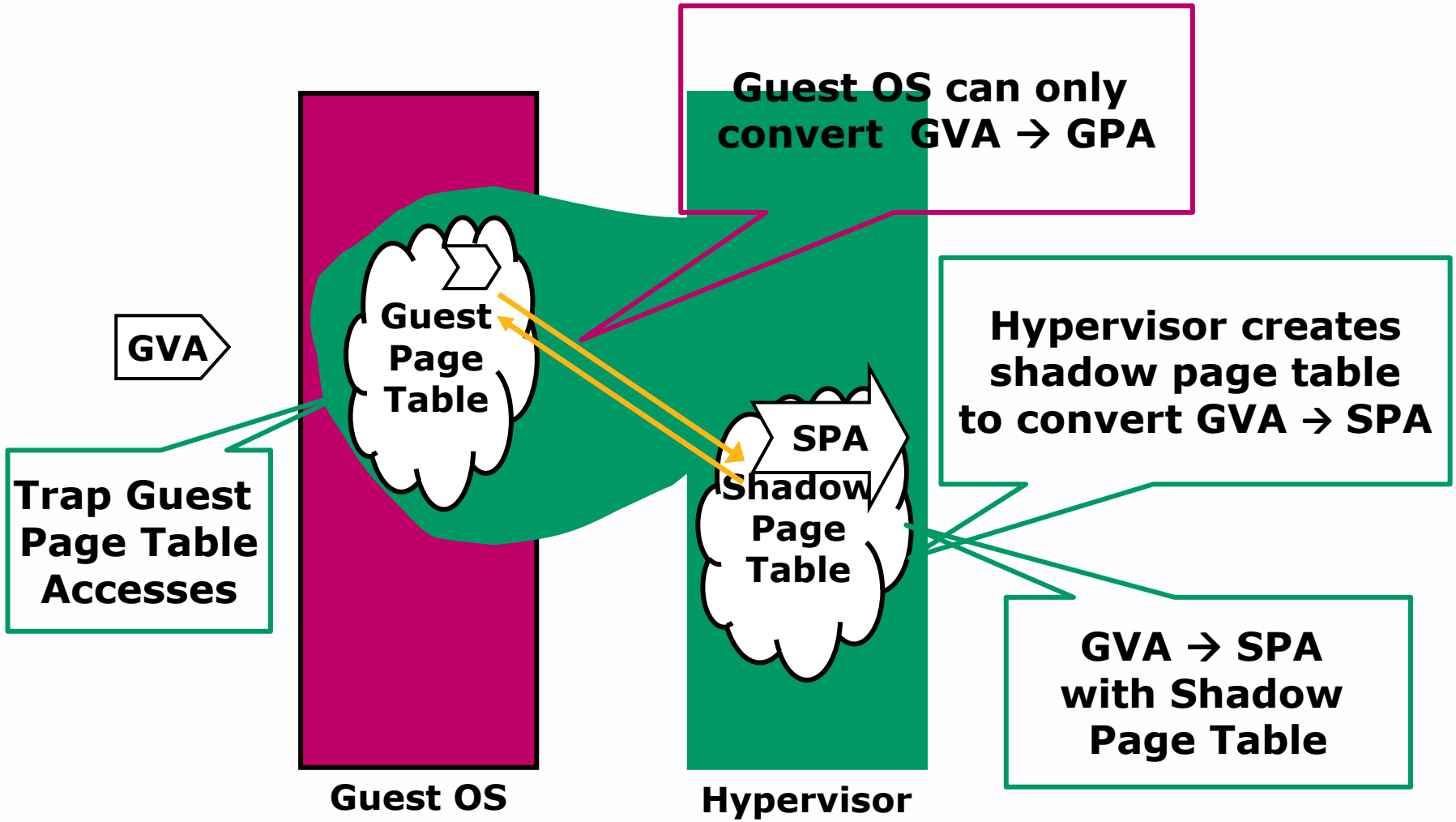
Addresses

- GVA: guest virtual address
- GPA: guest physical address
- SPA: system physical address



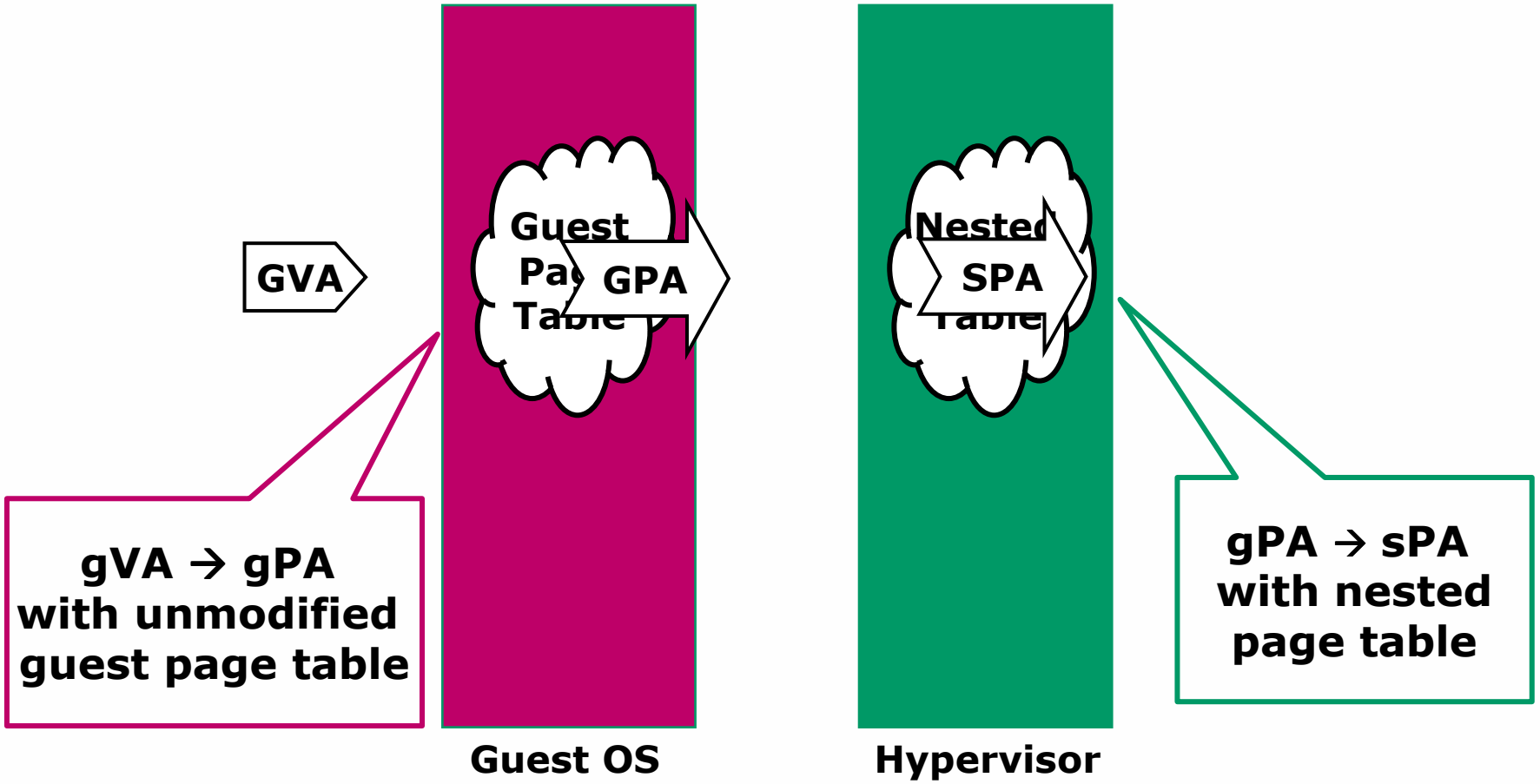
Virtualization Memory Management:
No hardware support: Shadow Paging

AMD
Smarter Choice



Virtualization Memory Management:
Hardware support: Nested Paging

AMD
Smarter Choice



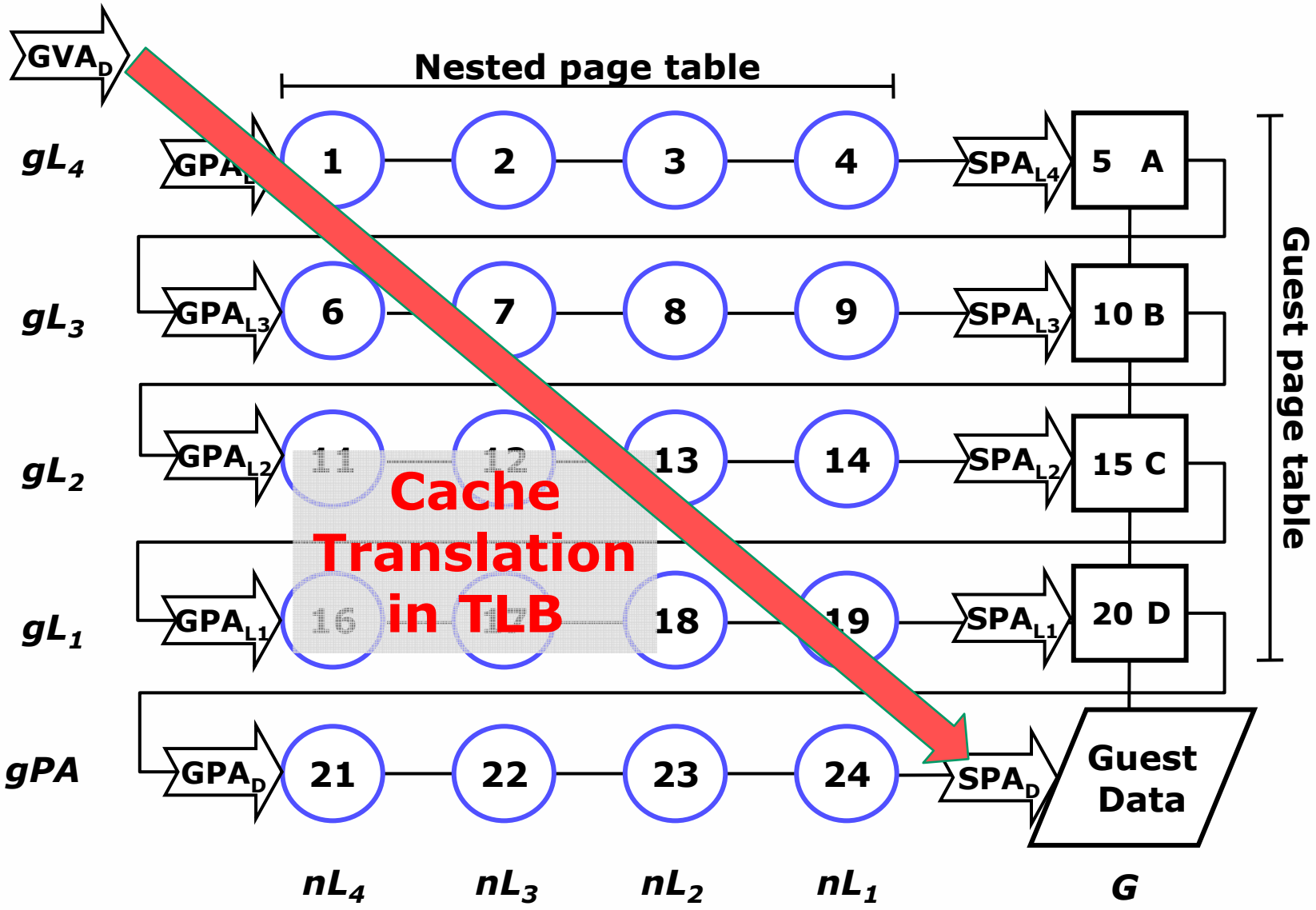
- **Benefits**: No more traps on Guest Page Table accesses
- **Drawback**: Extra page table steps add latency to TLB miss

Outline

AMD
Smarter Choice

1. Background
   - AMD64 Page Walks and Caching
   - Virtualization Terminology
   - Address Translations in Virtualized Systems
2. Two-Dimensional Page Walks
   - Nested Paging + Native Paging = 2D Page Walk
   - 2D Page Walk Caching
   - Hardware and Software 2D Page Walk Acceleration
3. KVM Implementation and Results
   - KVM Software Implementation
   - Results

Two-Dimensional (2D) Page Walk



Two-Dimensional Page Walk Caching

AMD Smarter Choice

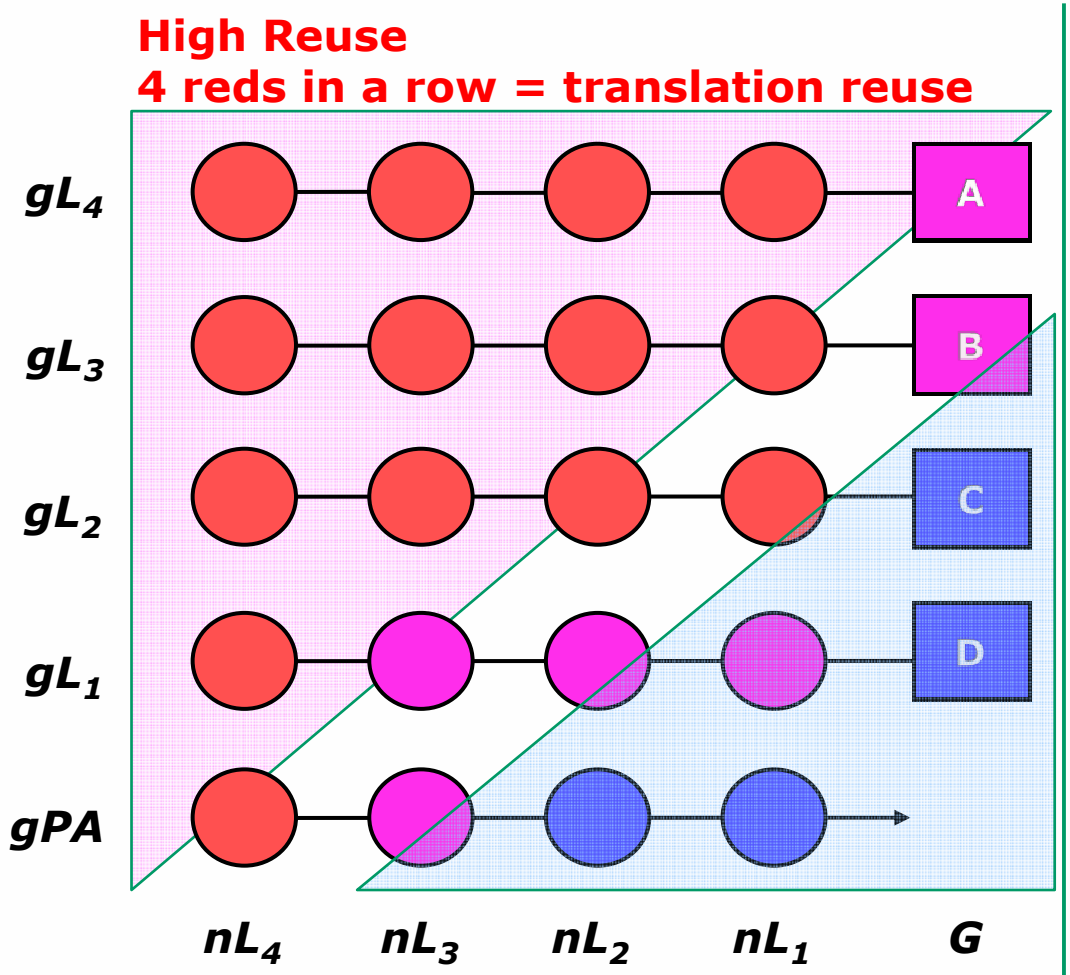
Average 2D Walk

- **14** PWC hits
  - 5% of PWC misses
- **6** mixed PWC hit/miss
  - 25% of PWC misses
- **4** PWC misses
  - 70% of PWC misses

BLUE = PWC MISS

PURPLE = PWC MIXED

RED = PWC HIT



Two-Dimensional Page Walk Caching:
with the Nested TLB (NTLB)

AMD Smarter Choice

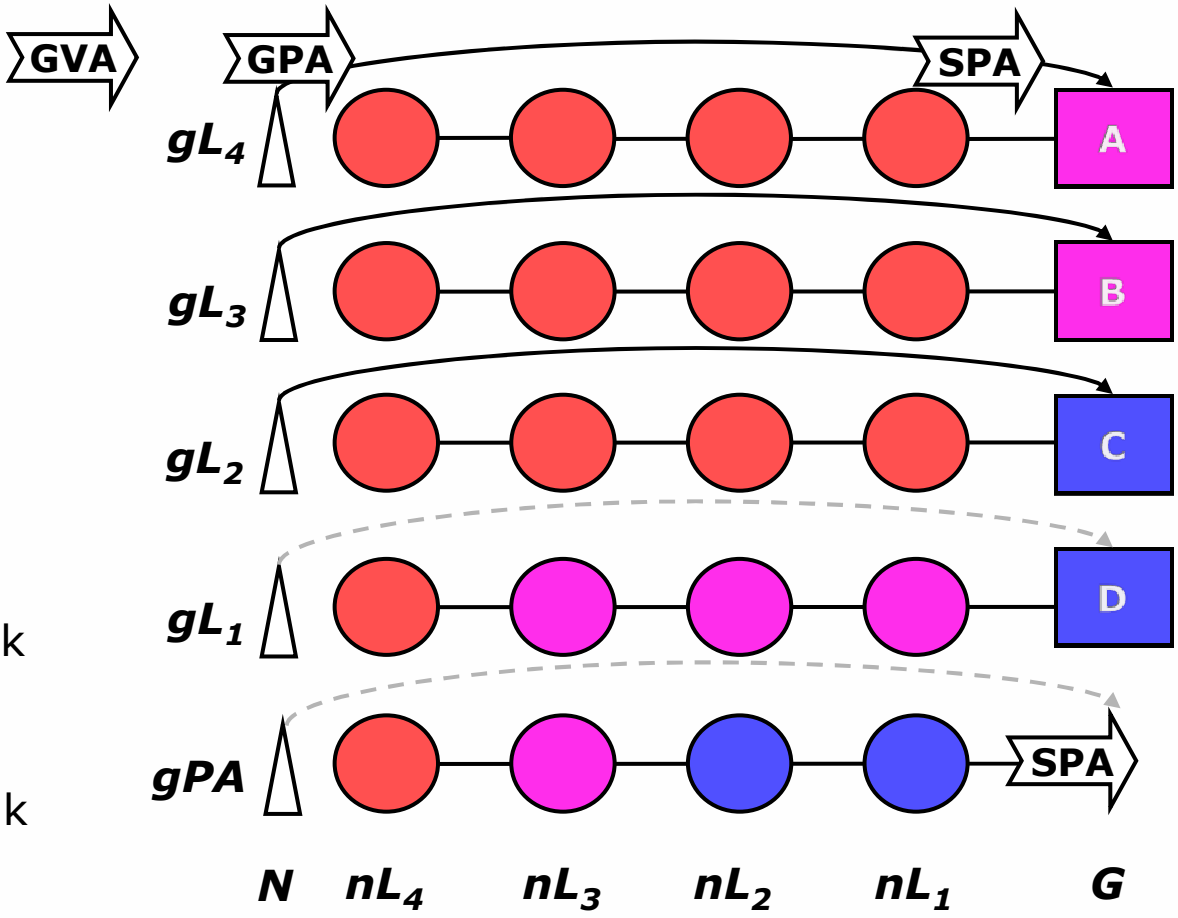
gL_4 NTLB Hit:
Skip Nested Page Walk

gL_3 NTLB Hit:
Skip Nested Page Walk

gL_2 NTLB Hit:
Skip Nested Page Walk

gL_1 NTLB Miss:
Perform Nested Page Walk

gPA NTLB Miss:
Perform Nested Page Walk

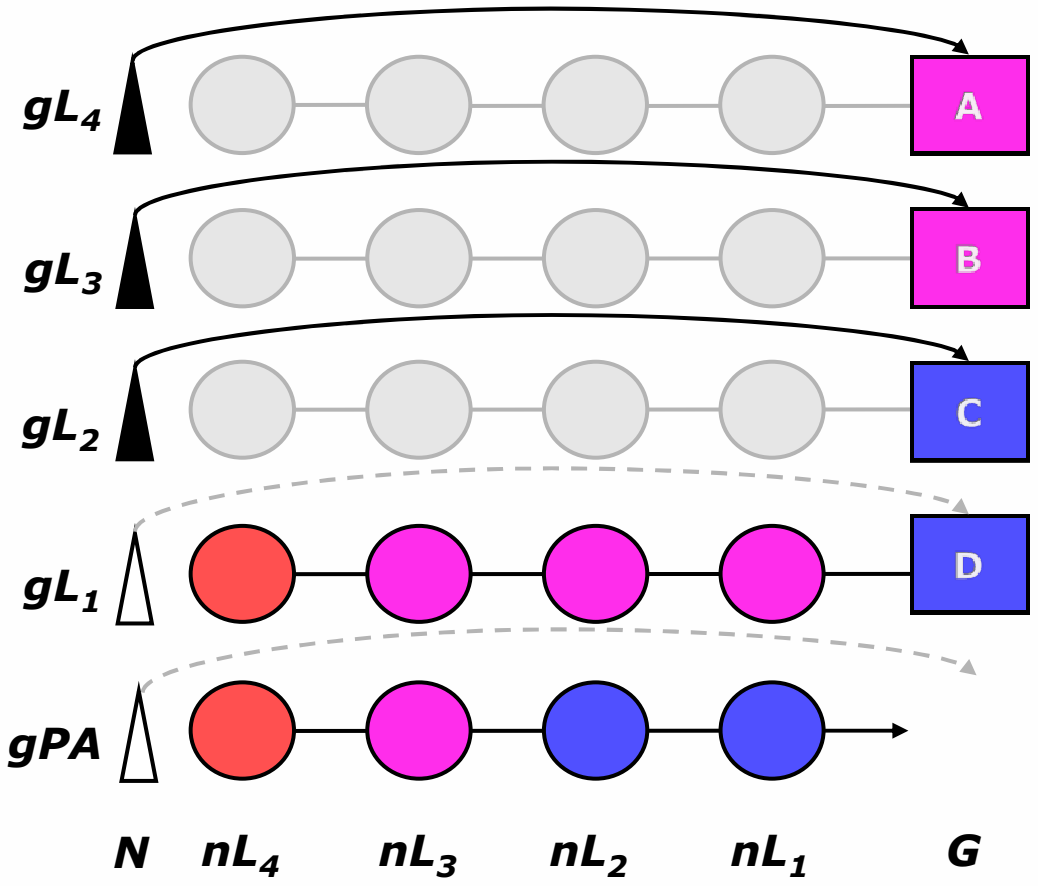


Simulated Results of not-exactly-real hardware – see ASPLOS08 paper

Sources of 2D Walk Overhead:
L2 Cache Misses

AMD Smarter Choice

- Many PWC misses become L2 cache misses
- 1 of 4 PWC misses also miss in L2 cache



Simulated Results of not-exactly-real hardware – see ASPLOS08 paper

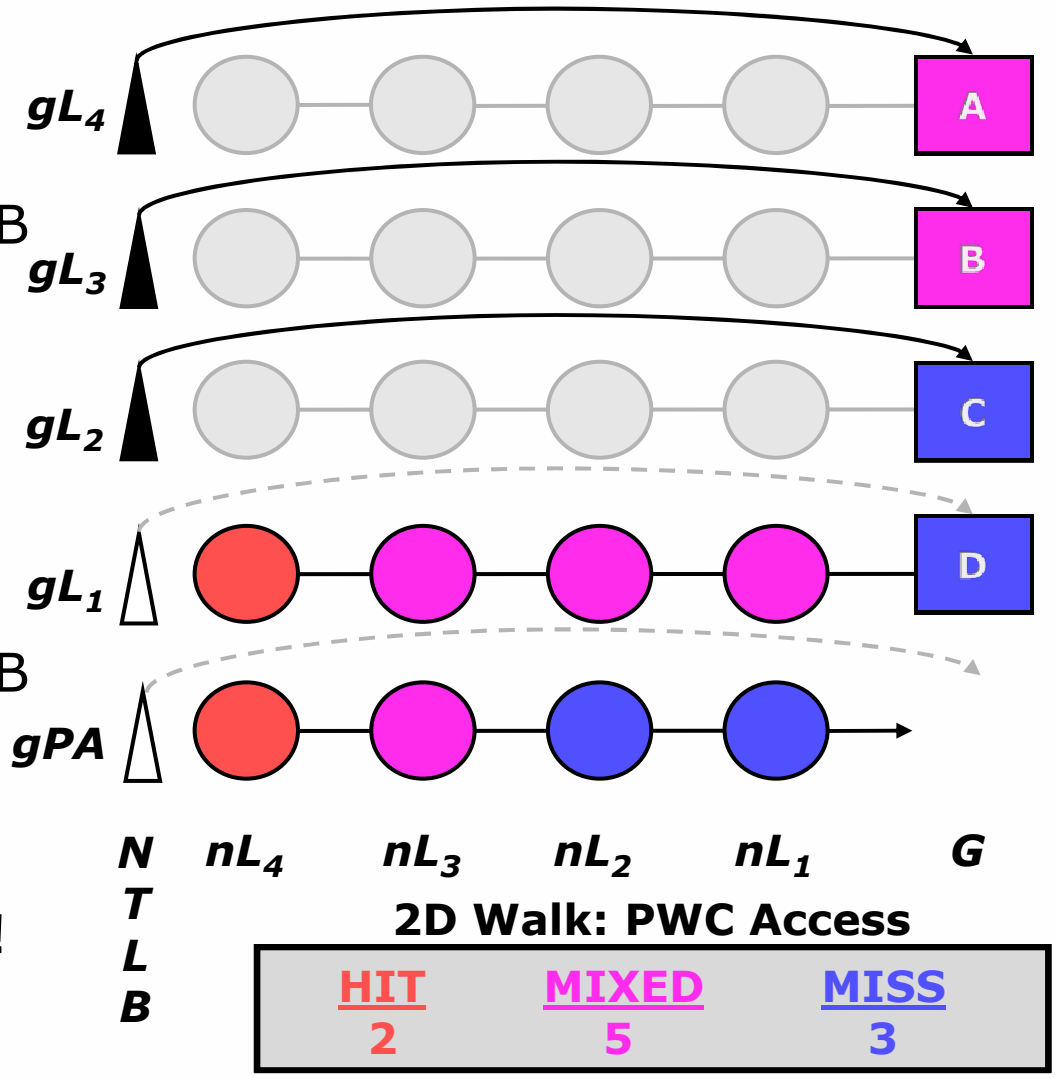
Large Nested Pages & Large Guest Pages

AMD Smarter Choice

Large Nested Pages
- Created by hypervisor
- Map 2MB instead of 4KB
- Eliminate nL_1 column
- Fewer PWC misses

Large Guest Pages
- Created by guest OS
- Map 2MB instead of 4KB
- Eliminate gL_1 row
- Fewer PWC misses

Large pages are good!



Outline

AMD
Smarter Choice

1. Background
   - AMD64 Page Walks and Caching
   - Virtualization Terminology
   - Address Translations in Virtualized Systems
2. Two-Dimensional Page Walks
   - Nested Paging + Native Paging = 2D Page Walk
   - 2D Page Walk Caching
   - Hardware and Software 2D Page Walk Acceleration
3. KVM Implementation and Results
   - KVM Software Implementation
   - Results

Nested Paging Support in KVM

AMD
Smarter Choice

- Direct mapped page table is the same as the nested page table
- Shadow Paging Code for Real Mode creates a direct mapped page table
- Nested Paging support utilizes Shadow Paging Code
- This kept implementation very simple `(5 files changed, 190 insertions(+), 12 deletions(-))`
- Live Migration and Guest Swapping work out of the box
- Real performance boost for KVM on AMD processors

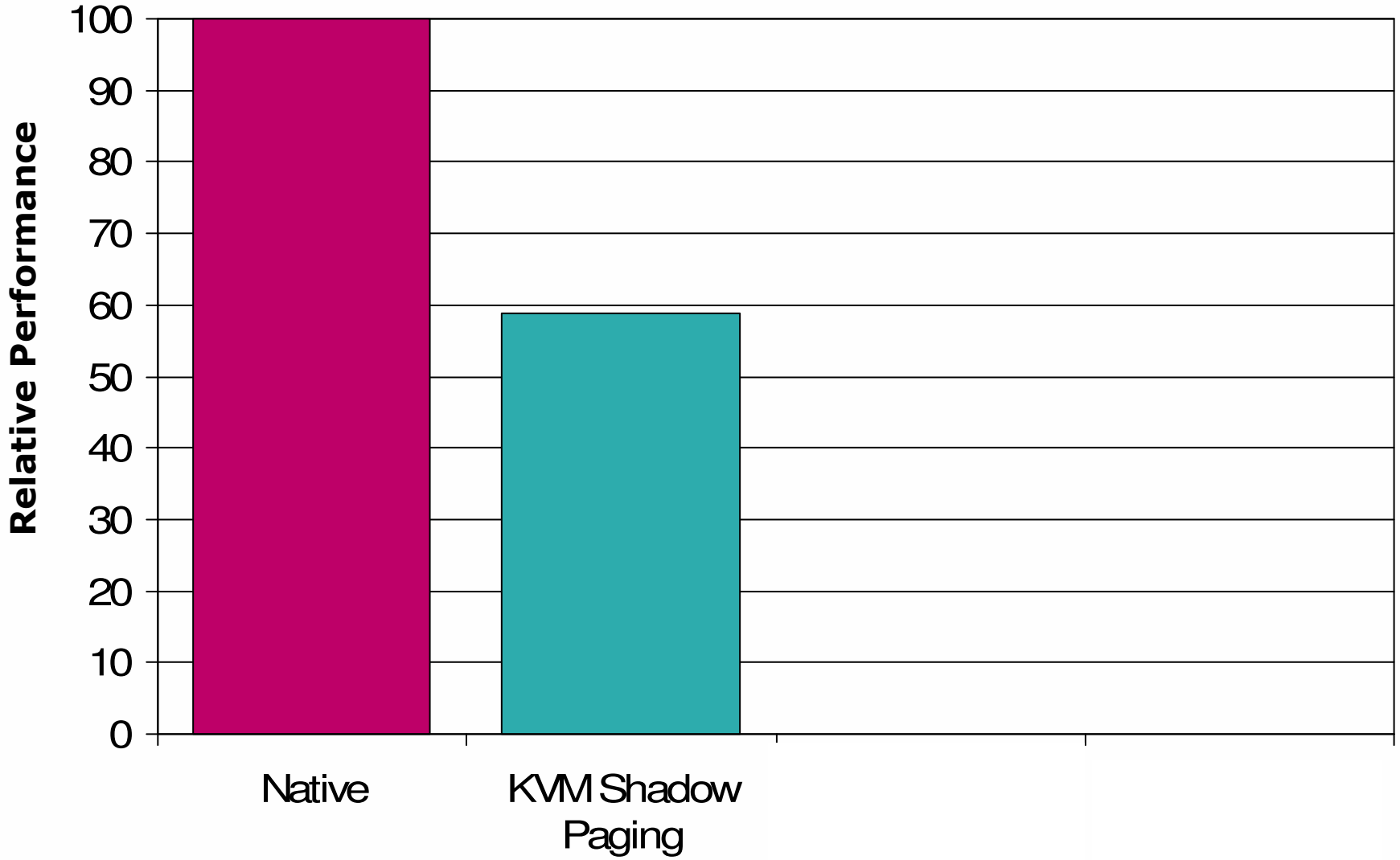
Benchmarking - Environment

AMD
Smarter Choice

- Hardware: AMD Phenom™ 9550 2.2 GHz B3 silicon with 4GB RAM
- Host OS: Redhat Enterprise Linux 5.2
  - KVM-69
  - Xen-unstable 17731
- Guest OS: Redhat Enterprise Linux 5.1
- Guest: 2 VCPUs and 2 GB Memory
- For benchmarks on bare metal (Native) host was booted with “maxcpus=2 mem=2G”

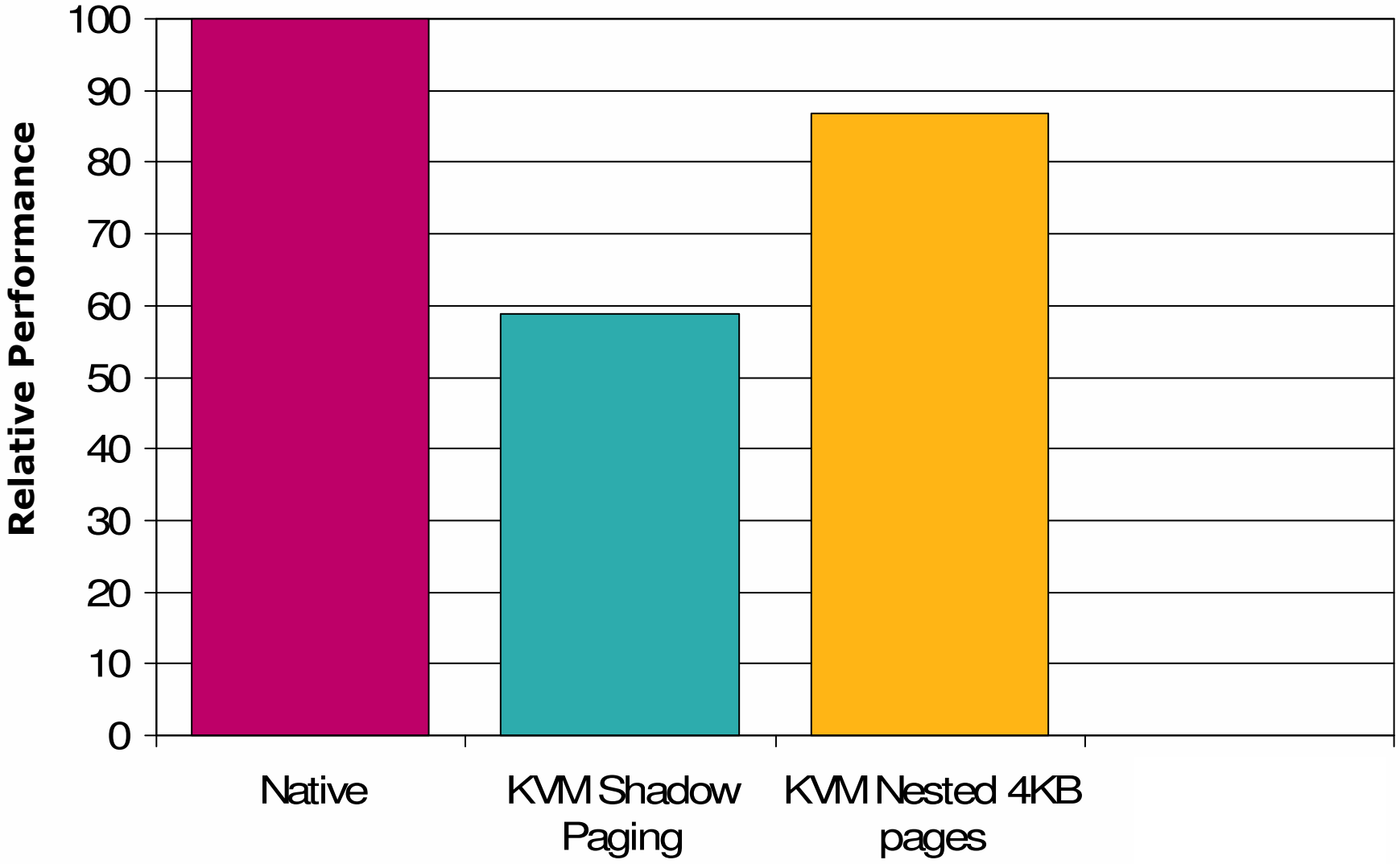
Kernbench Performance: Shadow Paging vs. Native

AMD
Smarter Choice



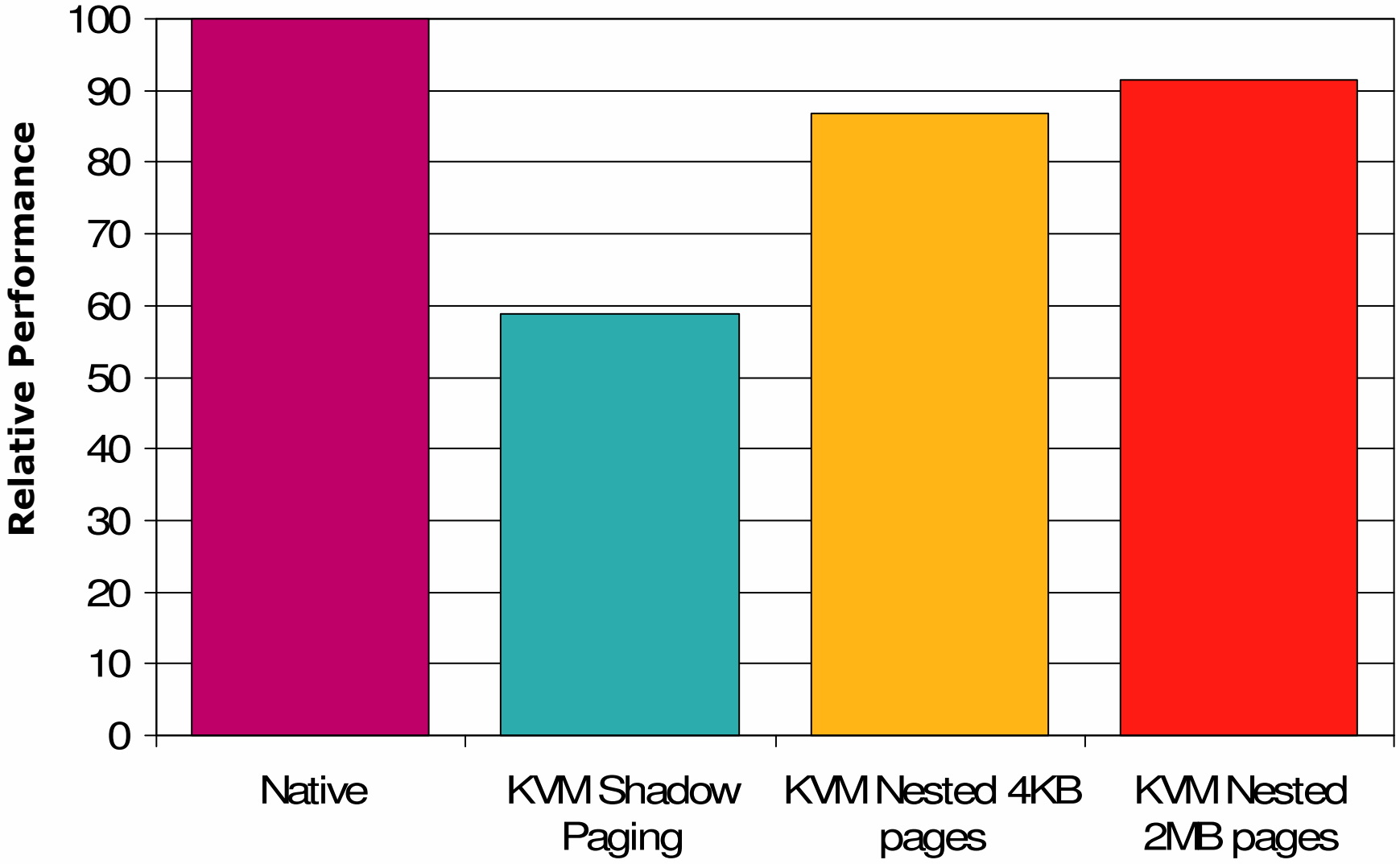
Kernbench Performance: Nested paging
KVM Nested 4KB pages

AMD
Smarter Choice



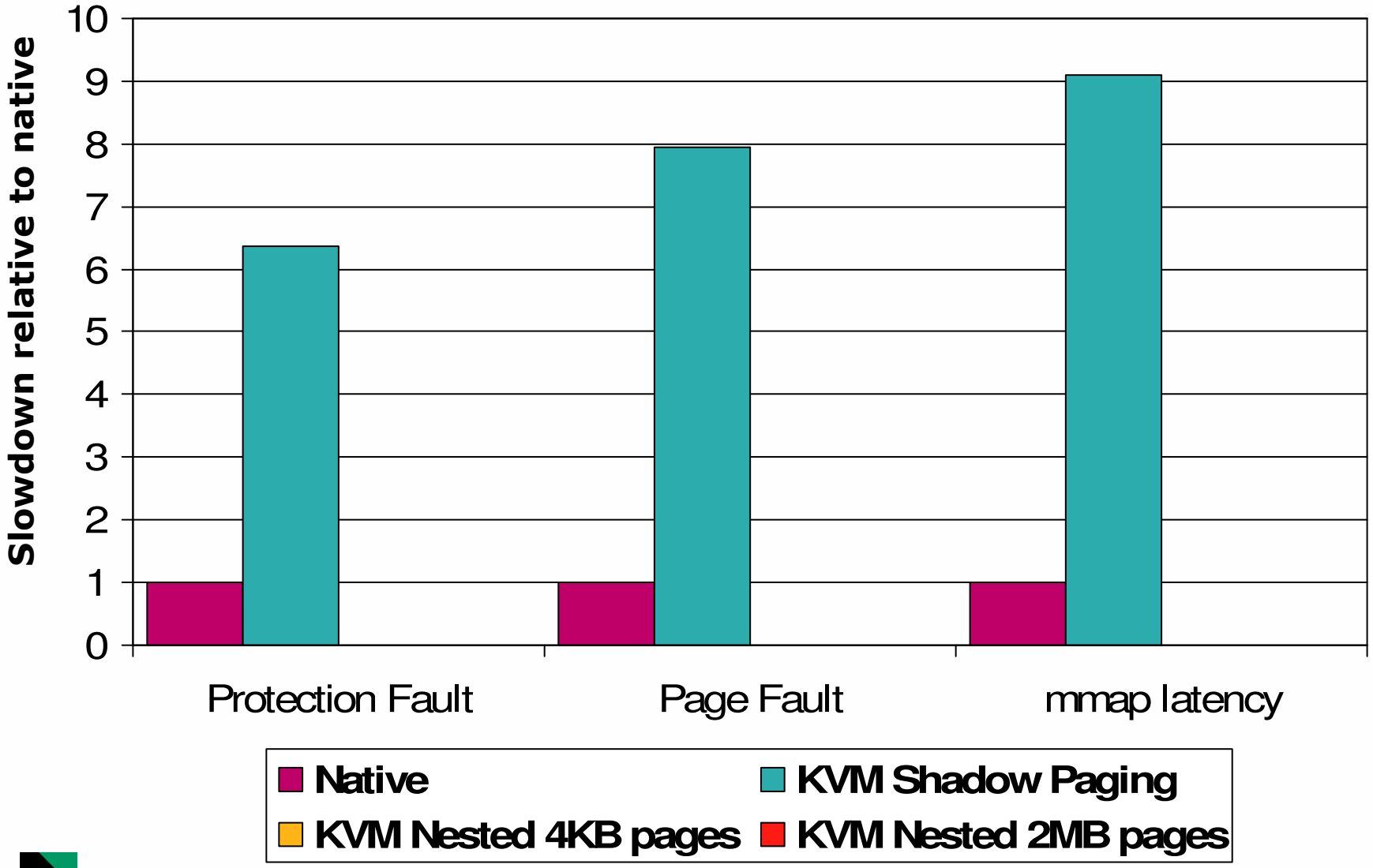
Kernbench Performance: Nested paging
Performance benefits from large pages

AMD
Smarter Choice



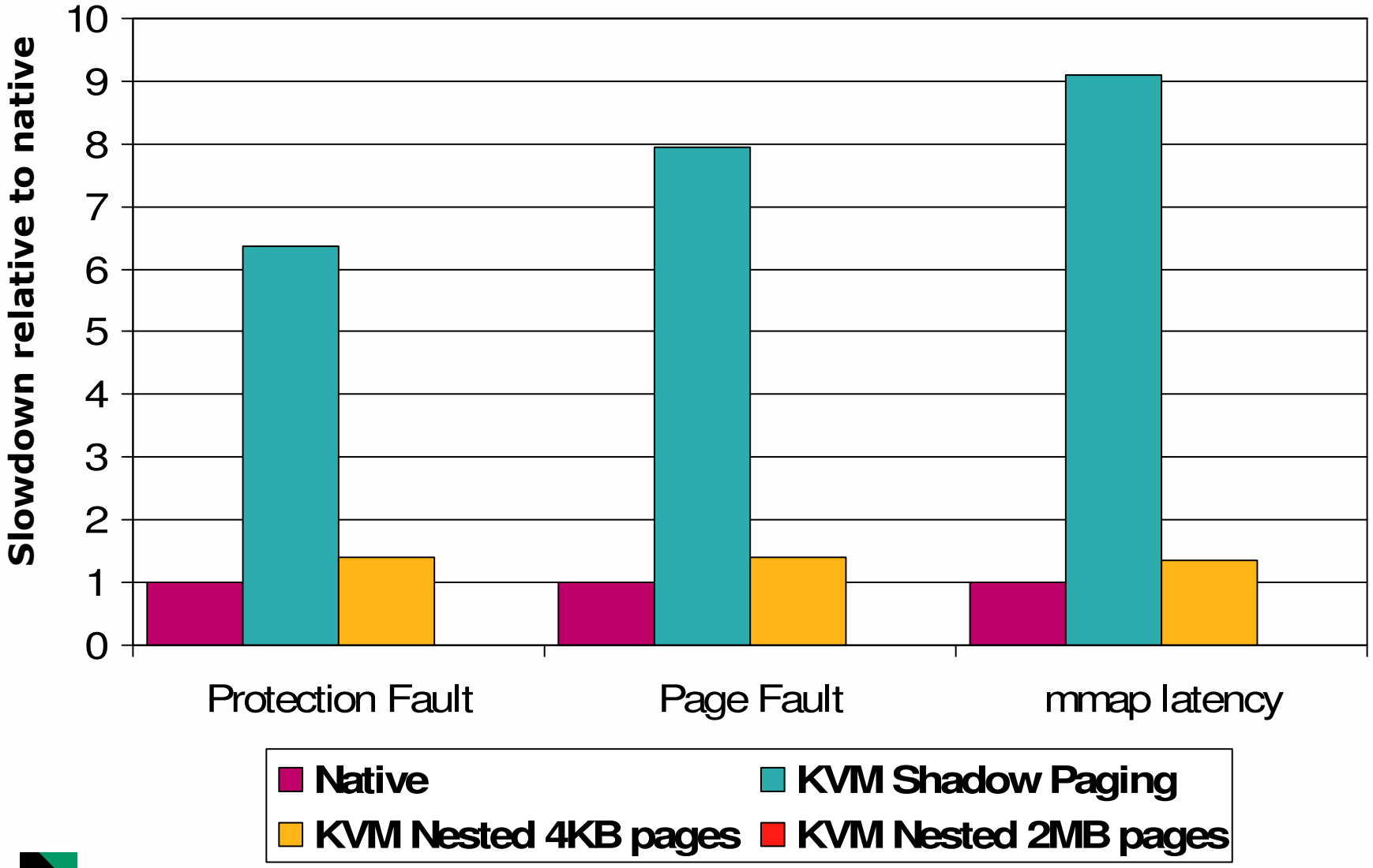
LMBench Performance: Shadow Paging vs. Native

AMD
Smarter Choice



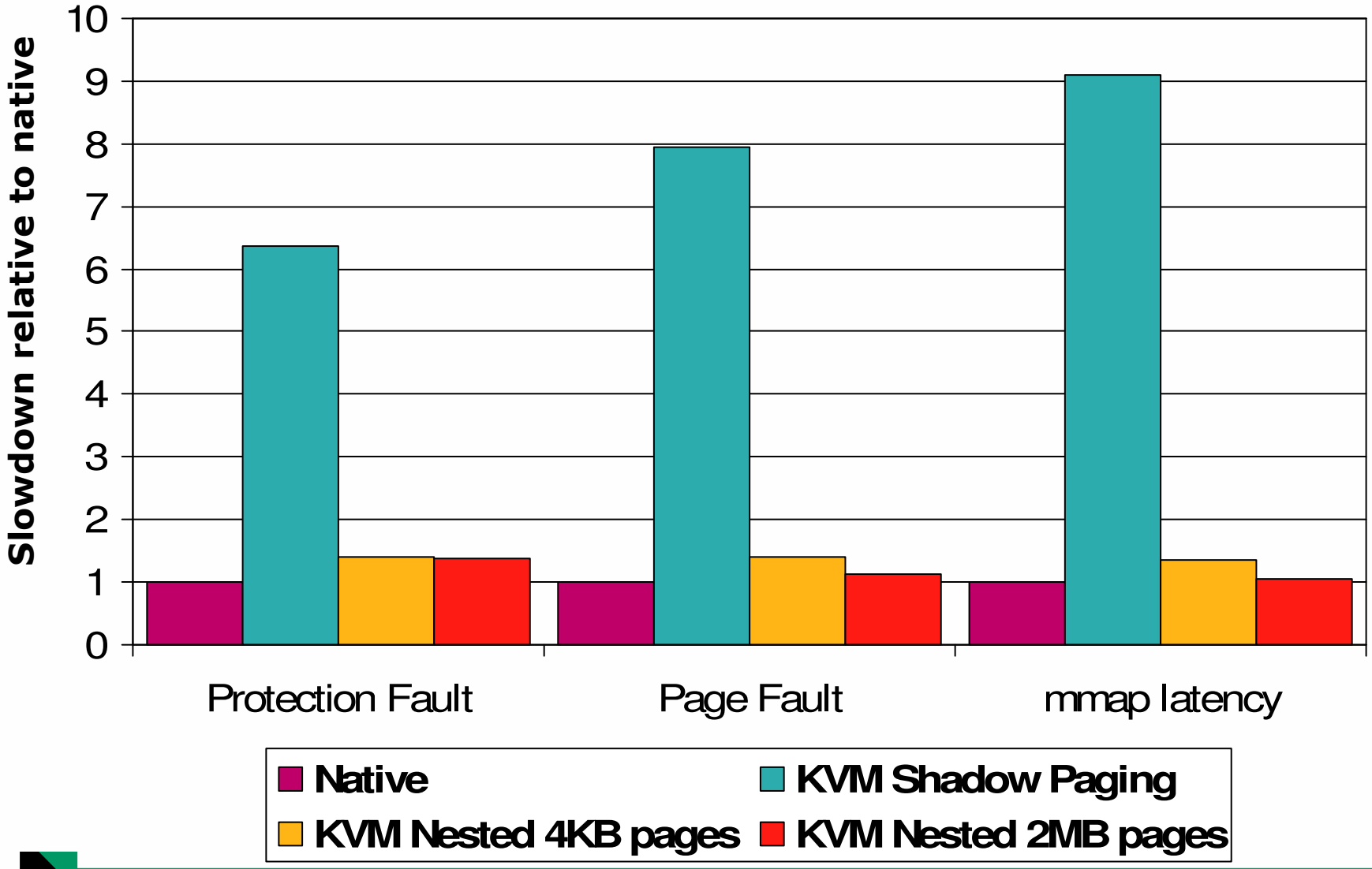
LMBench Performance: Nested paging
KVM Nested 4KB pages

AMD
Smarter Choice



LMBench Performance: Nested paging
Performance benefits from large pages

AMD
Smarter Choice



Conclusion

AMD
Smarter Choice

Nested Paging

- A HW solution to reduce memory management overhead
- Also introduces overhead on TLB misses

Hardware overhead can be significantly reduced

- *Nested TLB* to skip nested page walks and *Page walk cache*
- Approach native speed with these techniques

Overhead *elimination* more difficult

- Some 2D walk references always miss in PWC and L2 cache
- Exclusive use of 2MB pages in hypervisor is difficult

KVM Implements Nested Paging

- Performance improves and memory footprint shrinks
- Best performance from use of large nested page sizes

Trademark Attribution