Surpassing the TLB Performance of Superpages with Less Operating System Support

Madhusudhan Talluri, Mark D. Hill

University of Wisconsin-Madison

`{talluri,markhill}@cs.wisc.edu`

Motivation

Significant time in TLB miss handling (40 cycle miss penalty)

benchmark	% user time
coral	50%
compress	26%
mp3d	11%
spice2g6	7%
pthor	7%

Gets worse: kernel, multi-user/larger workloads, increased TLB miss penalty

How to improve TLB performance?

I) more TLB entries

➔ II) map more memory per TLB entry

Alternative TLB entries:

	Tag	Data	Area
Single-page-size	VPN	PPN | Attr | V	100%
Superpage	VPN | Size	PPN | Attr | V	103%
Complete-subblock	VPN (-2 bits)	PPN0 | Attr | V0 PPN1 | Attr | V1 PPN2 | Attr | V2 PPN3 | Attr | V3	170%
Partial-subblock	VPN (-2 bits) | V0 | V1 | V2 | V3	PPN | Attr	103%

Outline

Motivation

Superpages

Subblock TLBs

TLB Performance

Operating system implications

Conclusion

Superpages:

Map virtual address region with superpage if

- Starts at power of 2 aligned virtual AND physical addresses
- Length is a power of 2
- All base pages in region have same attributes
- Mapped to contiguous physical memory pages

Otherwise use base pages

+ Fewer TLB misses

- Increased OS overheads

- Increased TLB miss penalty

- Cannot always use superpages *E.g.,* a 60KB file

Superpages

Commodity microprocessors include superpage support in TLB

E.g., MIPS R4000 supports 4KB base page size and
Superpage sizes of 16KB, 64KB, 256KB, 1MB, 4MB, 16MB

I) large superpages (>= 256KB)

kernel, frame buffers, databases

Assume support for large superpages fully utilized

II) medium-sized superpages (32KB/64KB)

user text, heap, shared libraries

No Operating system support for superpages in user programs

Outline

Motivation

Superpages

Subblock TLBs

TLB Performance

Operating system implications

Conclusion

Subblock TLBs:

Each TLB entry maps “s” consecutive base virtual pages

(s is the subblock factor)

Complete-subblock : TLB stores full PTE for each base page

Partial-subblock : TLB shares some fields among base pages

Simpler restrictions than superpages

Applicable in more situations (*E.g.*, 60KB file)

Complete-subblock TLB:

TLB entry holds full PTEs for consecutive base virtual pages

Base virtual pages share a single TLB entry if

- **lie within same superpage region (trivial)**

+ No extra operating system support required

- Larger area than superpage or partial-subblock TLBs

For small subblock factors (2 or 4),

complete-subblock TLB better than single-page-size TLB

***E.g.,* MIPS R4000 : complete-subblock TLB, subblock factor 2**

Partial-subblock TLBs

TLB entry stores only one copy of PPN and Attributes

Base virtual pages share a single TLB entry if

- **lie within same superpage region (trivial)**
- **have same attributes (common)**
- **are mapped to aligned physical pages (OS support)**

Otherwise use separate TLB entries

+ less restrictive than superpages

+ fewer TLB misses than superpage TLBs

+ lesser area than complete-subblock TLB

- TLB design is slightly complicated

Outline

Motivation

Superpages

Subblock TLBs

TLB Performance

Operating system implications

Conclusion

TLB Parameters

base page size = 4KB

superpage size = 64KB

subblock factor = 64KB/4KB = 16

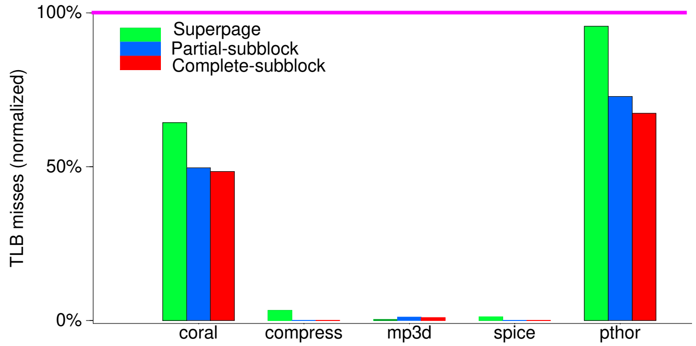
64-entry fully-associative TLBs :

single-page-size, superpage, complete and partial-subblock

Metric (TLB performance):

$$\frac{\text{\# of TLB misses for new TLB}}{\text{\# of TLB misses for single-page-size TLB}} * 100\%$$

TLB Performance



Superpages are very effective

Partial-subblock TLB is even better

Partial-subblock TLB comparable to complete-subblock TLB

OS support essential for Superpage and Partial-subblock TLBs

Outline

Motivation

Superpages

Subblock TLBs

TLB Performance

Operating system implications

Conclusion

Operating system support

Operation	Superpages	Partial-Subblocking
Variable sized memory allocation	Nearly identical (*e.g.*, page reservation)	
Page fault latency and I/O	Increased	Unchanged
Page promotion/Page size assignment	Explicit	Not required
Use for small/unaligned objects	No	Yes
Page Table/TLB miss handler	Complex	Unchanged
TLB misses to load superpage	One	Multiple

OS overheads may overwhelm TLB performance gains

Outline

Motivation

Superpages

Subblock TLBs

TLB Performance

Operating system implications

Conclusion

Which side are you on?

Bulls: Superpage OS support reasonable

Choose Partial-subblock TLBs

more efficient OS support + better TLB performance

Bears: Superpage OS support UNreasonable

Choose larger single-page-size or complete-subblock TLBs

Without OS support superpage TLB perf. is vaporware

Summary: have your cake and eat it too!

Relative to single-page-size TLB

	TLB Performance	OS overhead
Superpage TLB	Substantially better	Substantial
Subblock TLB	Even better	Even less