CacheRewinder:

Revoking Speculative Cache Updates Exploiting Write-Back Buffer

Jongmin Lee*, Junyeon Lee†, Taeweon Suh*, Gunjae Koo*

*Korea University

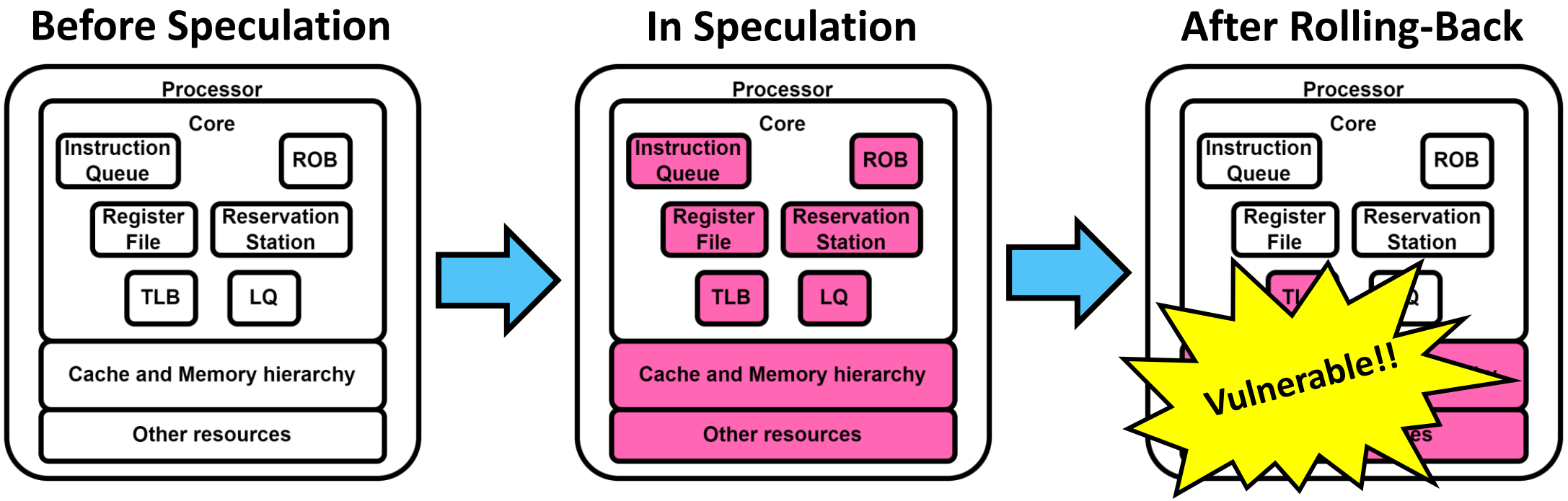
†Samsung Electronics

KOREA UNIVERSITY

Outline

- *Backgrounds & Motivation*
- *Architecture of CacheRewinder*
- *Evaluation*
- *Conclusion*

Attacks Exploiting Speculative Executions

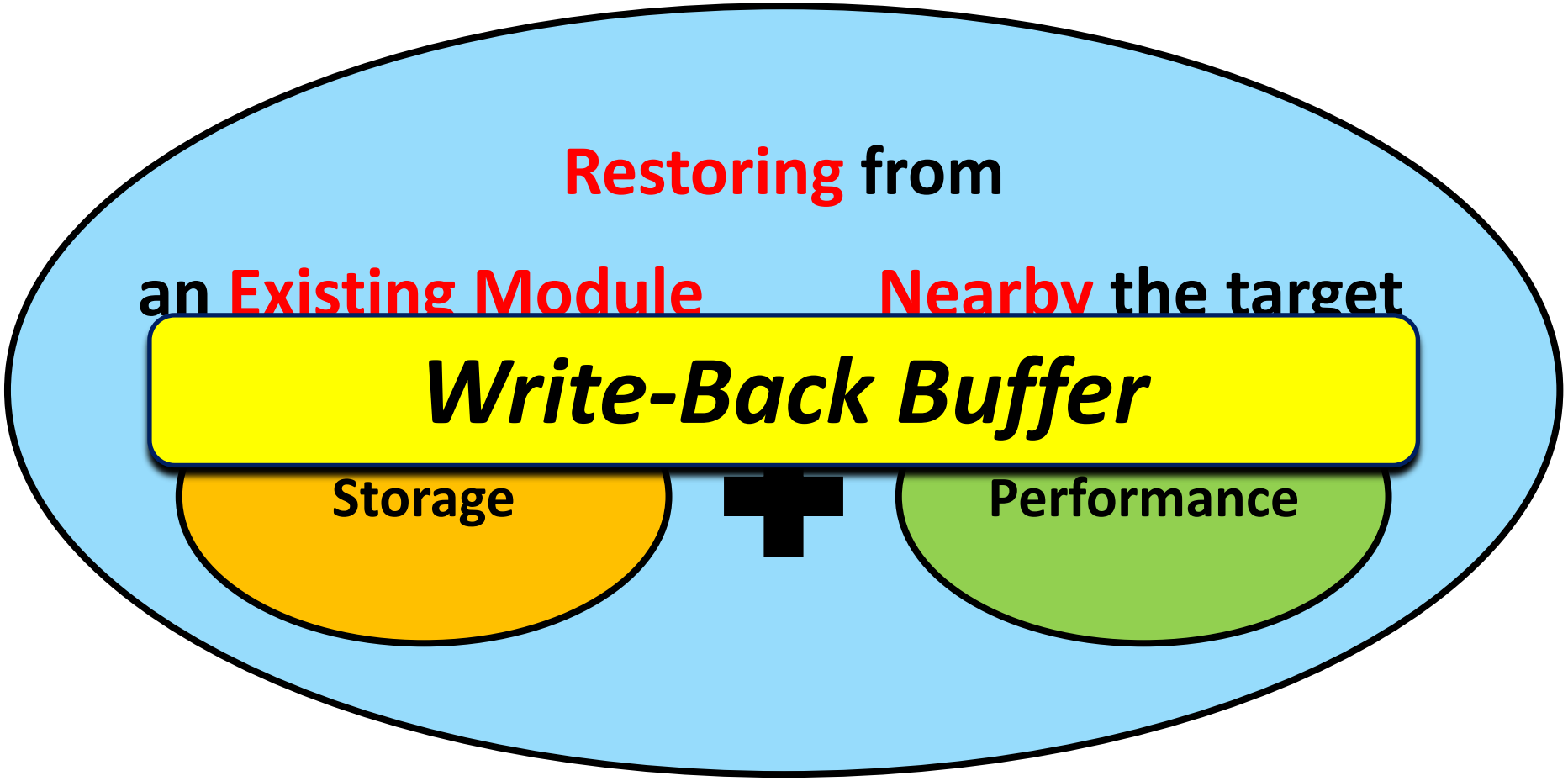


Inefficiency of Previous Mitigations

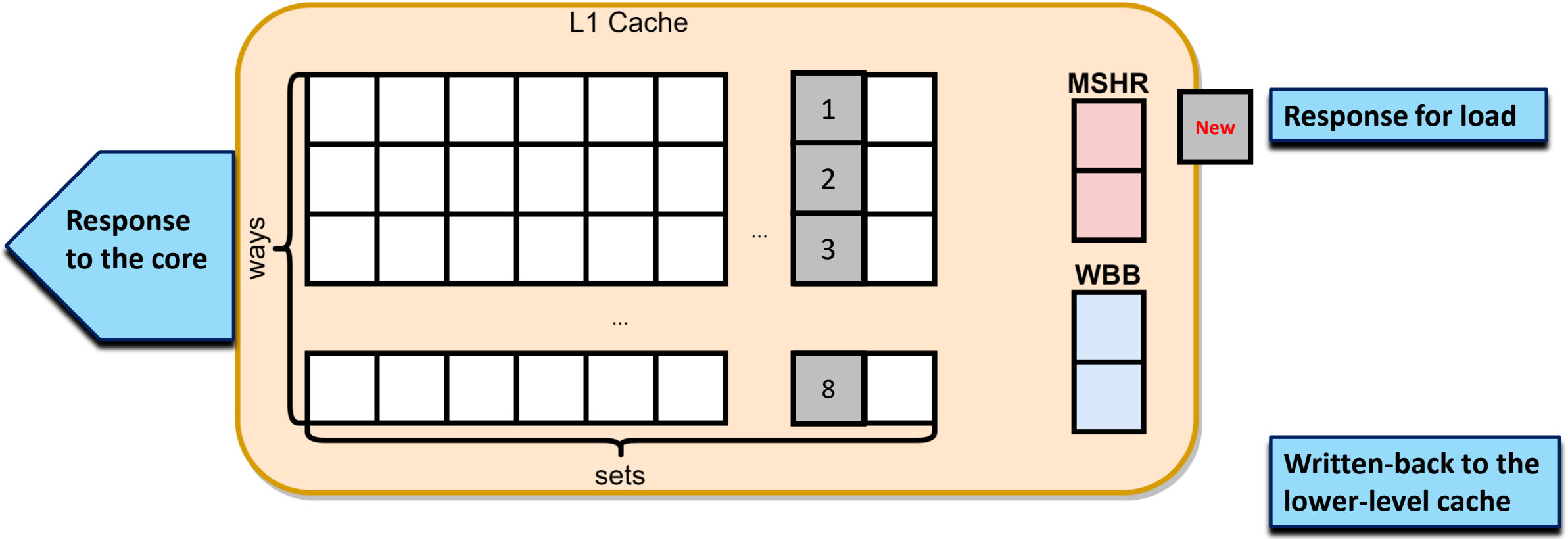
Software Approaches	Hardware Approaches	
	Delaying	*Restoring*

• **Static**
  • **By a user**
  • **By a compiler**
• Significant **performance drop**
	• **Performance degradation** by correct speculations
	• **Additional buffer** for delayed data
		• **Performance degradation** by restorations of cache states
		• **Additional buffer** for data that may be restored

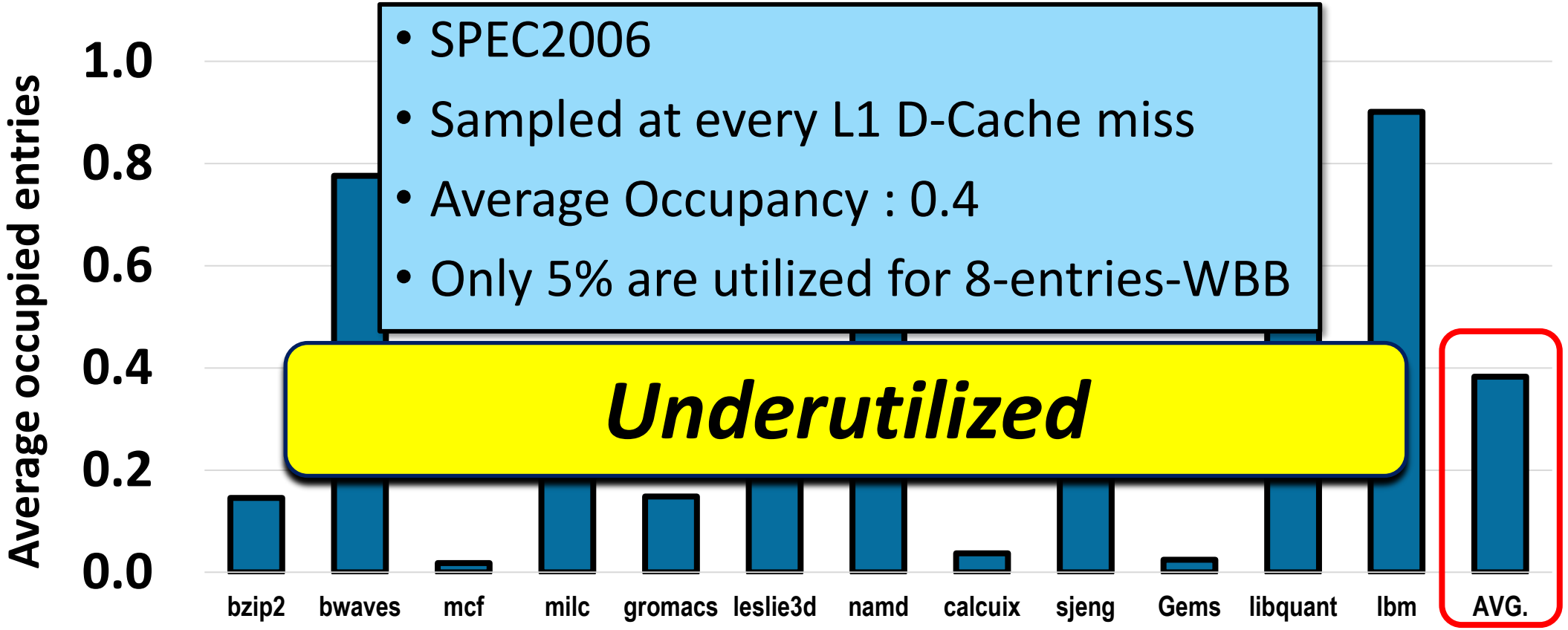
The Solution



Write-Back Buffer



Underutilized Write-Back Buffer

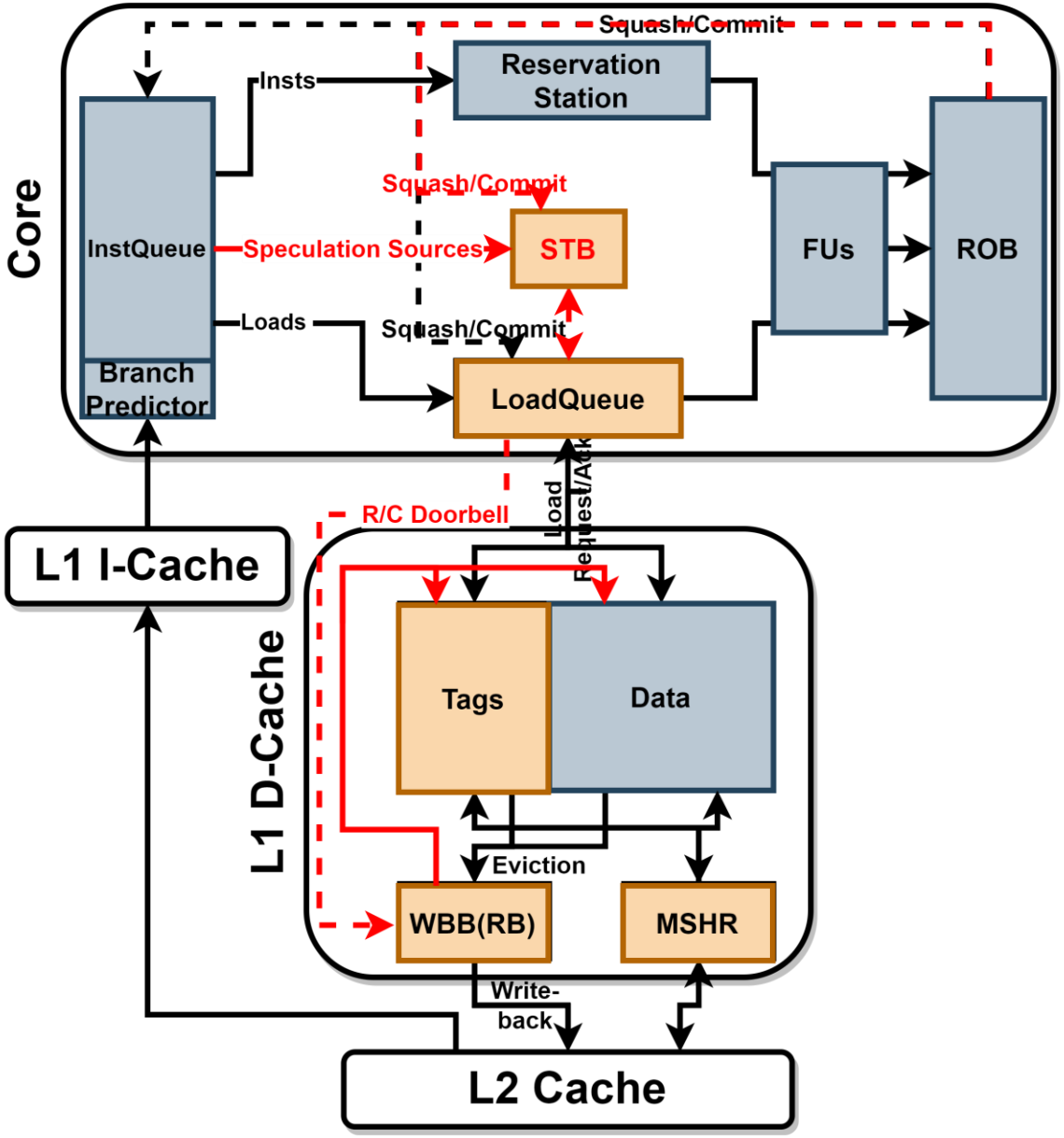


- SPEC2006
- Sampled at every L1 D-Cache miss
- Average Occupancy : 0.4
- Only 5% are utilized for 8-entries-WBB

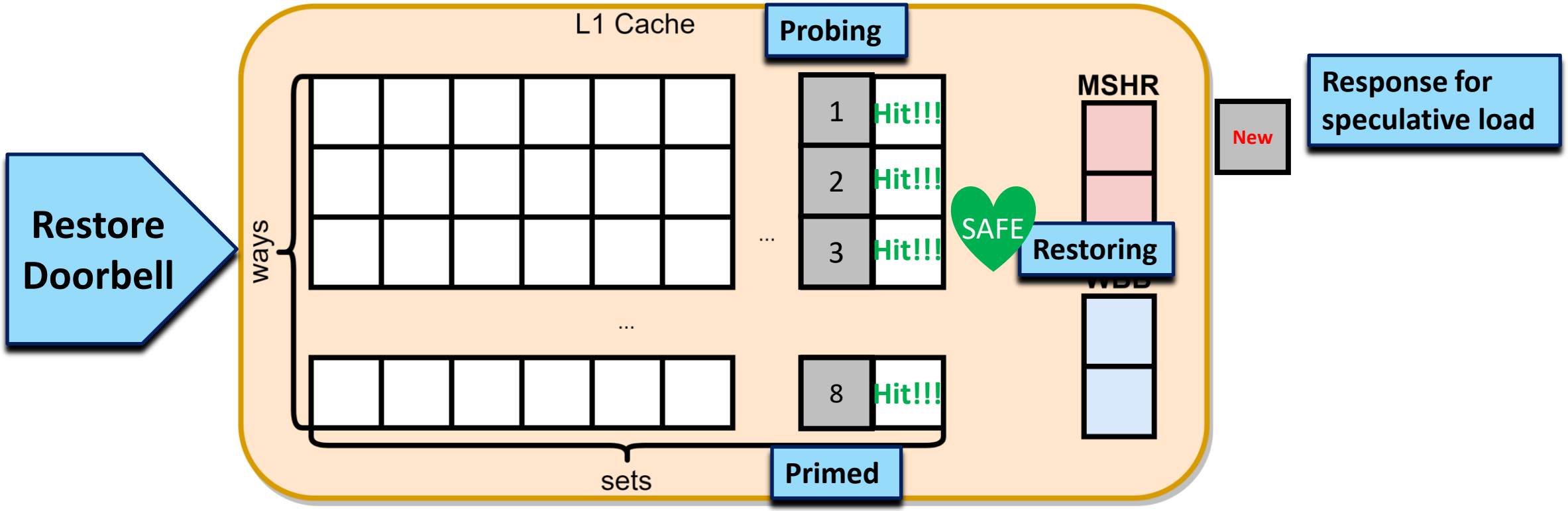
Underutilized

CacheRewinder

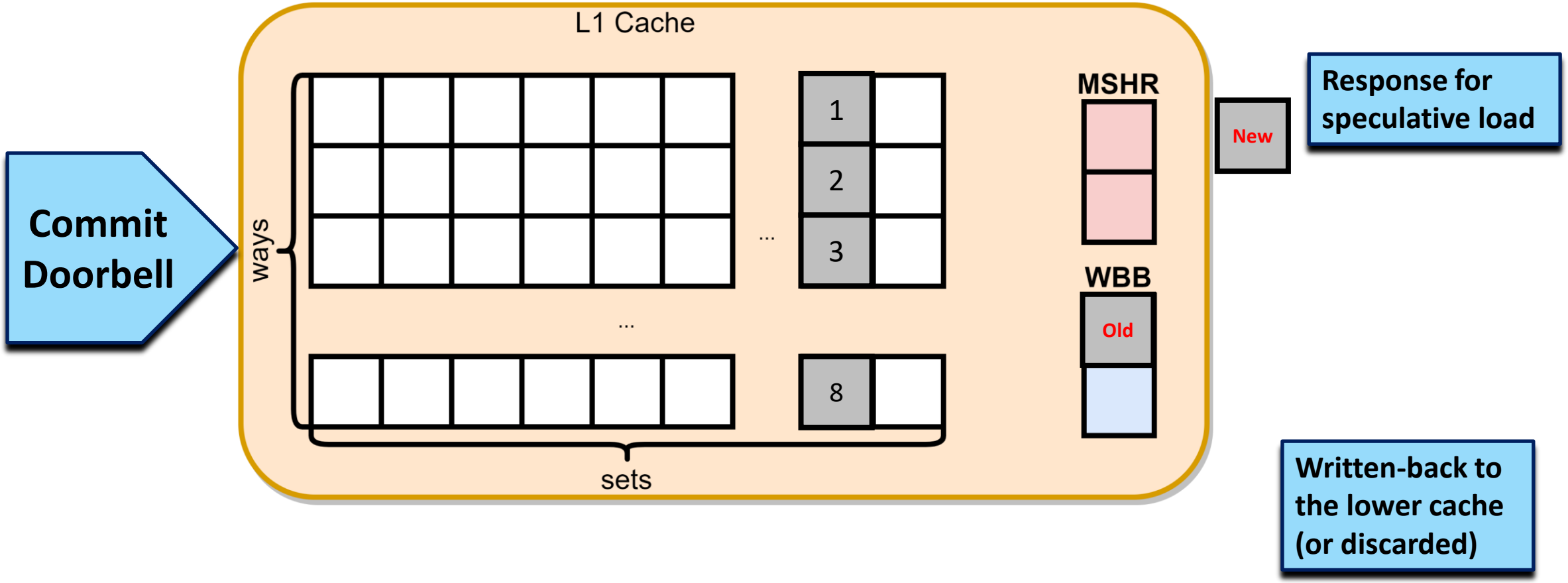
- *For Core*
  - *Newly added STB (Speculation Tracking Buffer)*
  - *Modified LQ*
- *R/C Doorbell*
- *For L1 D-Cache*
  - *Modified WBB (as a RB)*
  - *Modified MSHR*
  - *Modified Tag Memory*



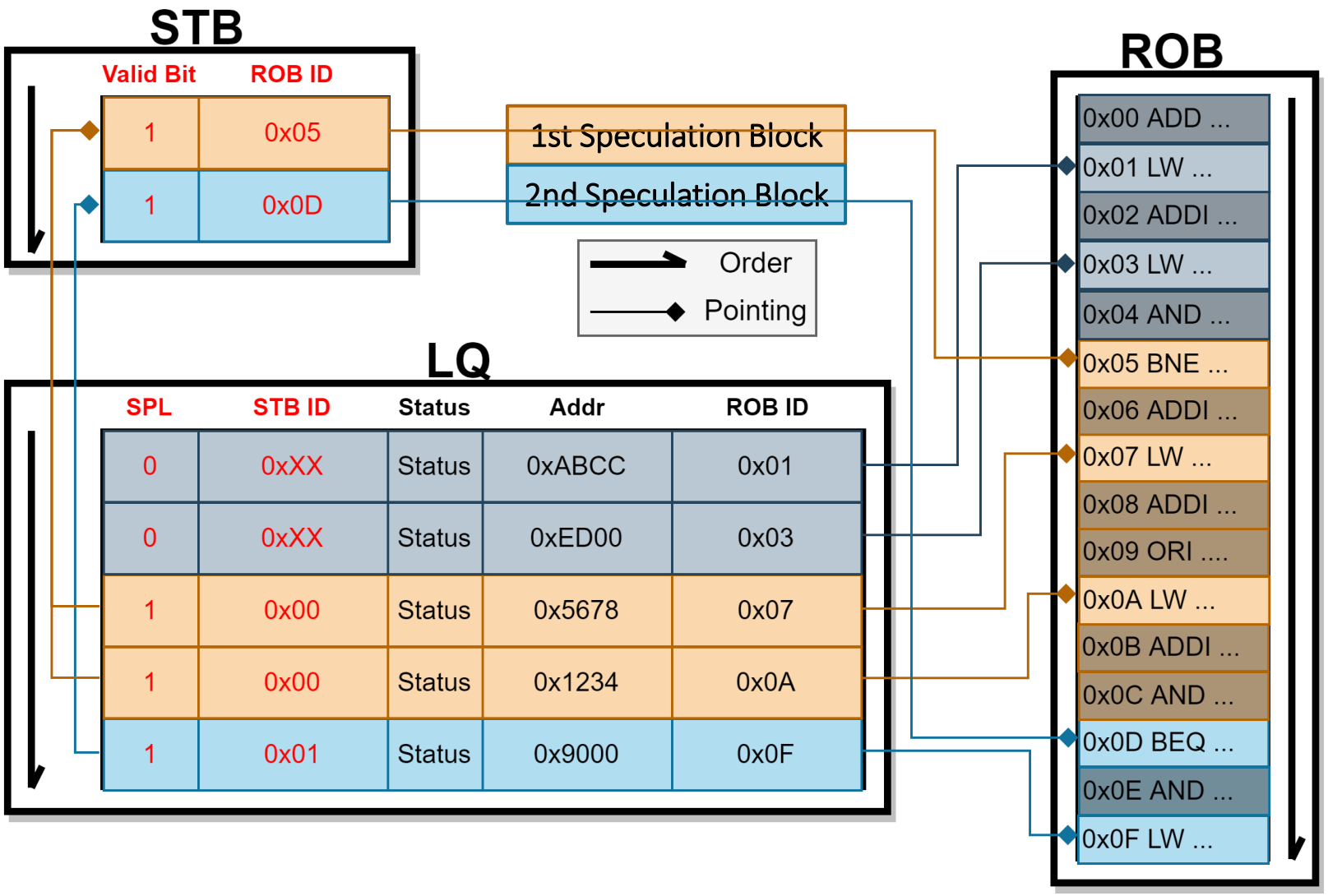
Restore Operation against Spectre



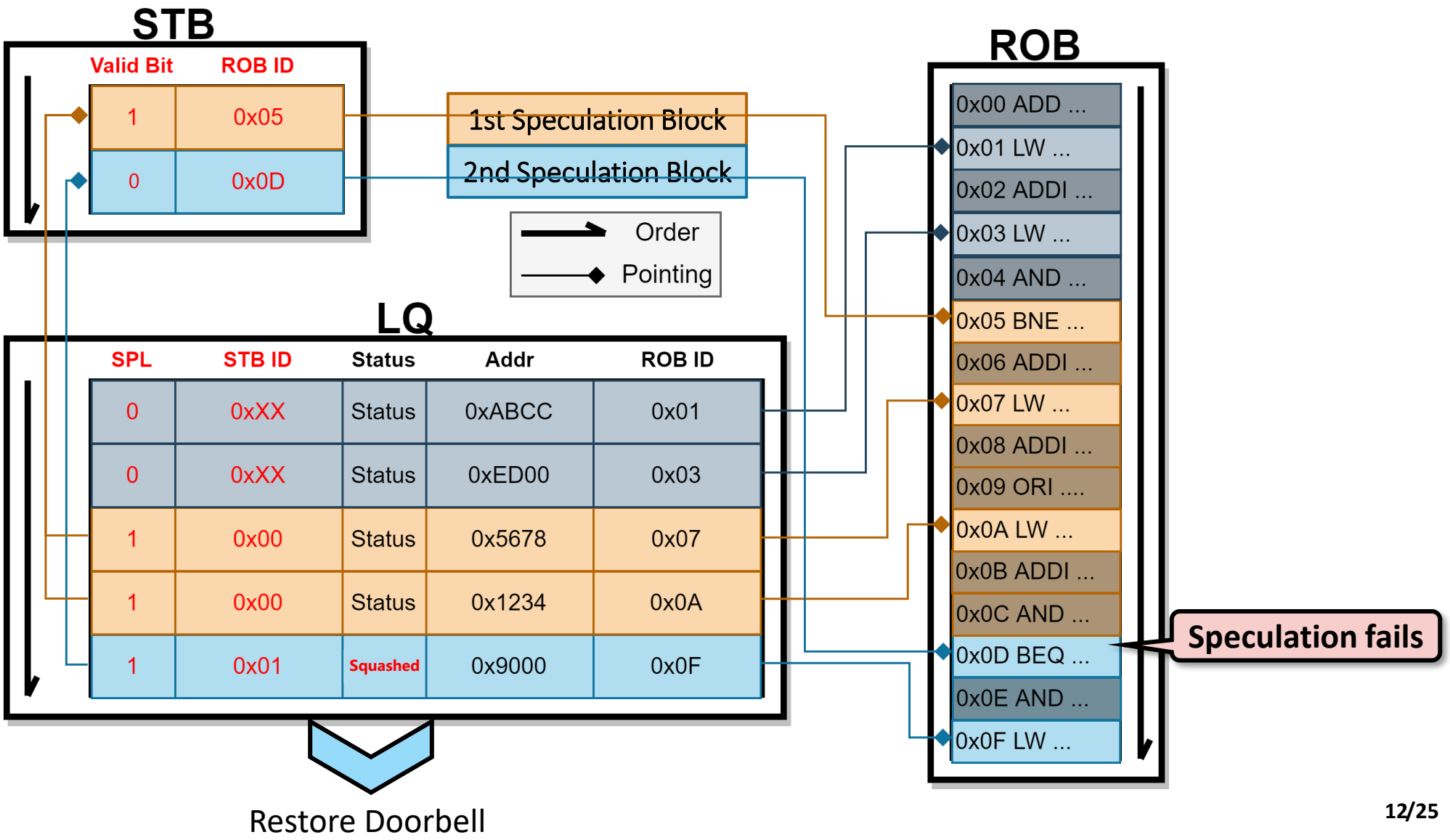
Commit Operation



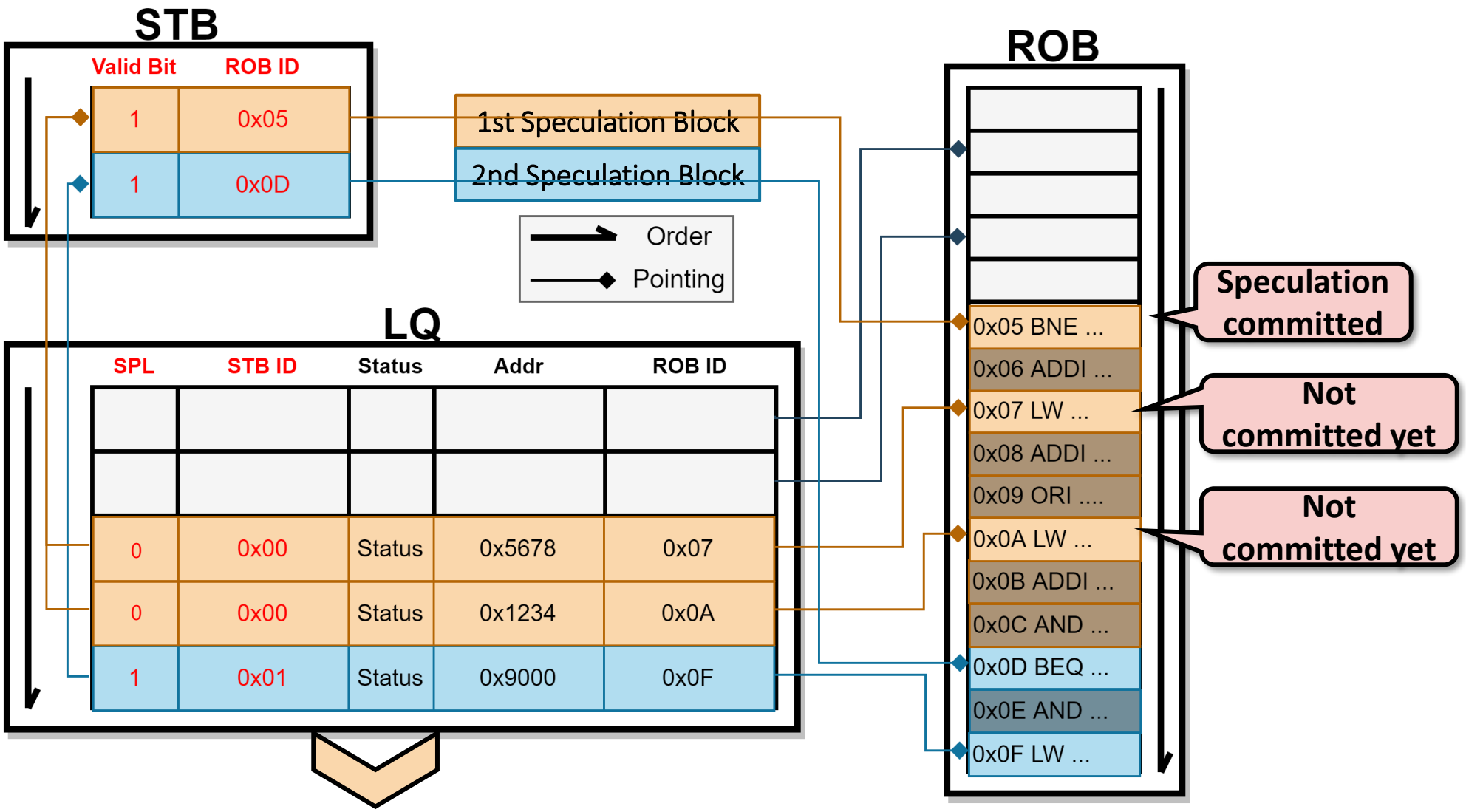
Speculation Tracking Buffer (STB)



Restore Doorbell



Commit Doorbell



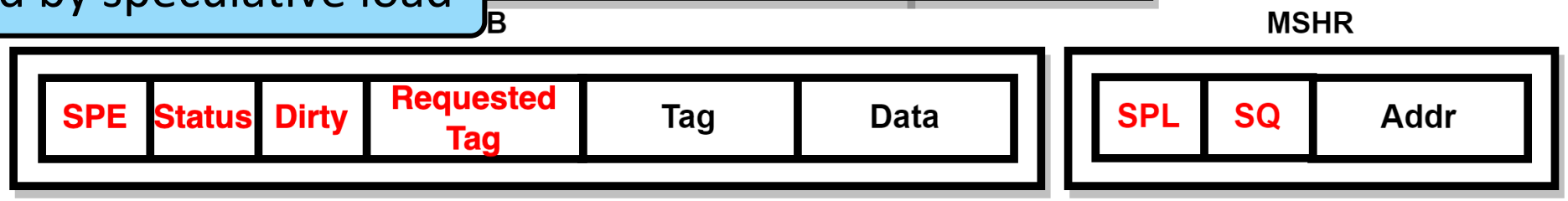
2 Commit Doorbells

Modification of L1 D-Cache

SPI : installed by a speculative load

SQ : canceling the in-flight case

SPE : evicted by speculative load

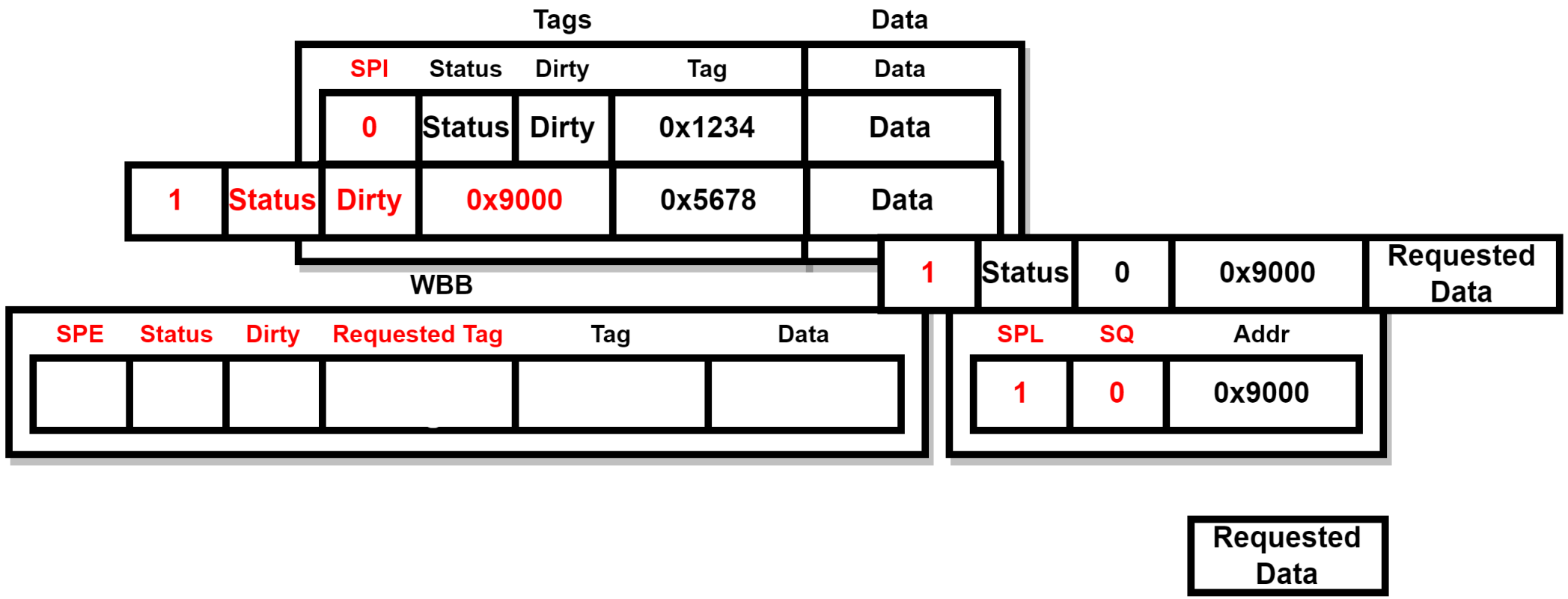


Requested Tag : the address of the replacer

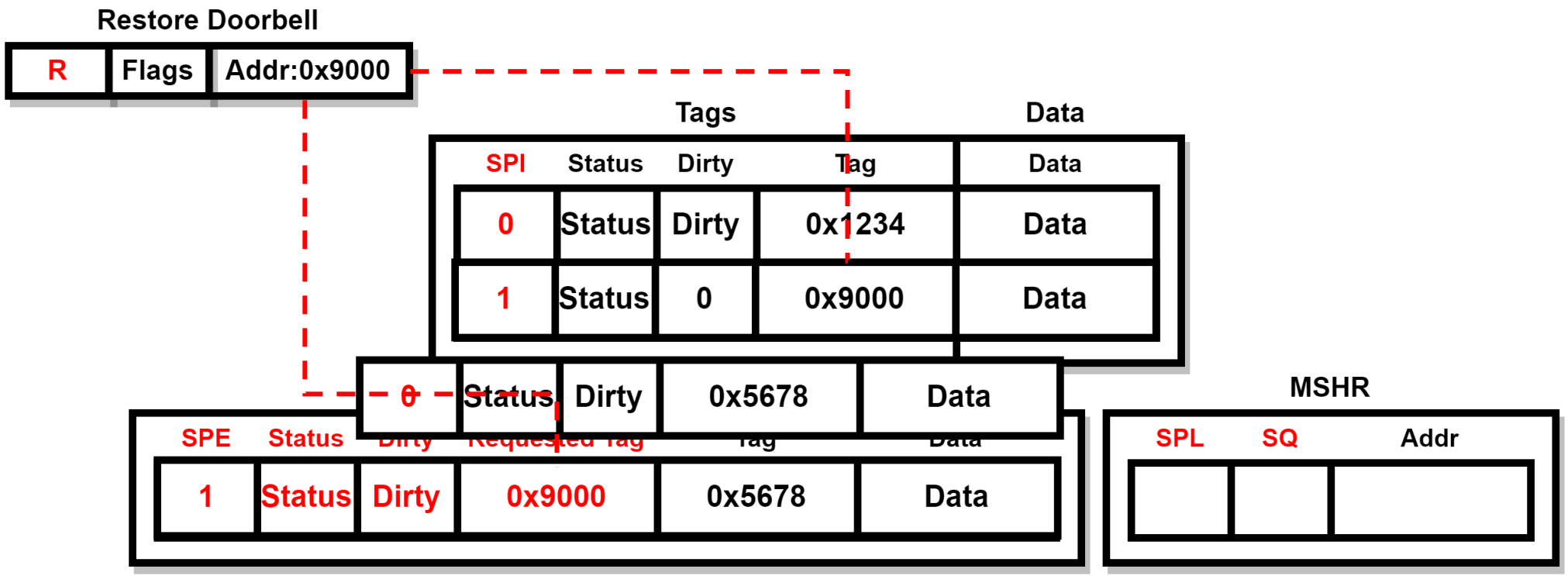
SPL : checking speculative load miss

Status Flags including dirty bit : data to be restored

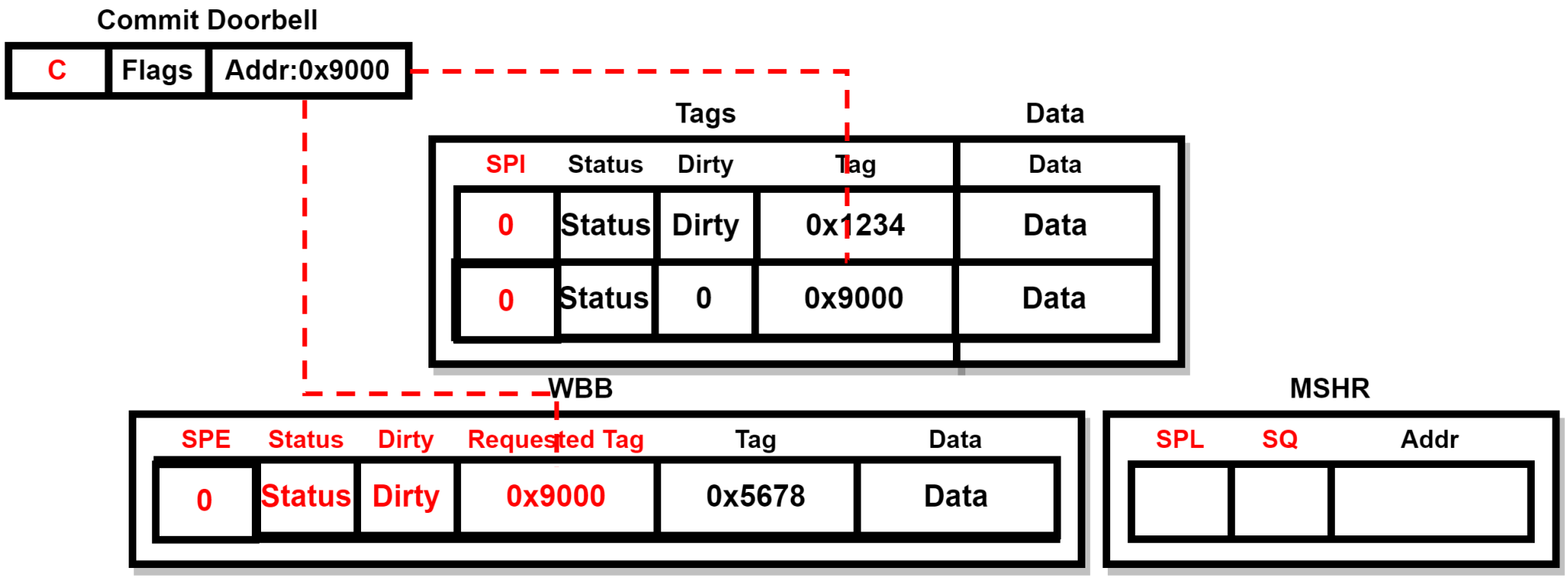
Speculative Load of Modified Cache



Restore Operation of Modified Cache



Commit Operation of Modified Cache

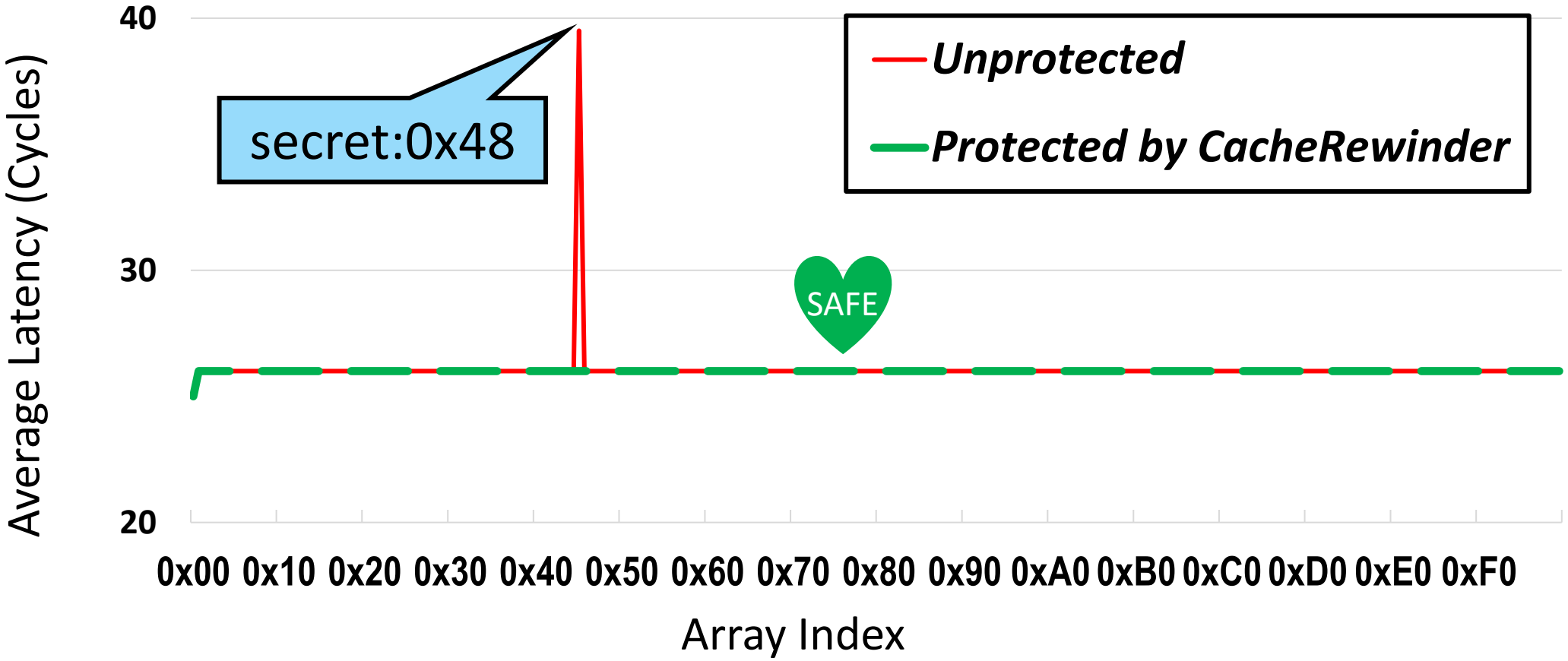


Evaluation

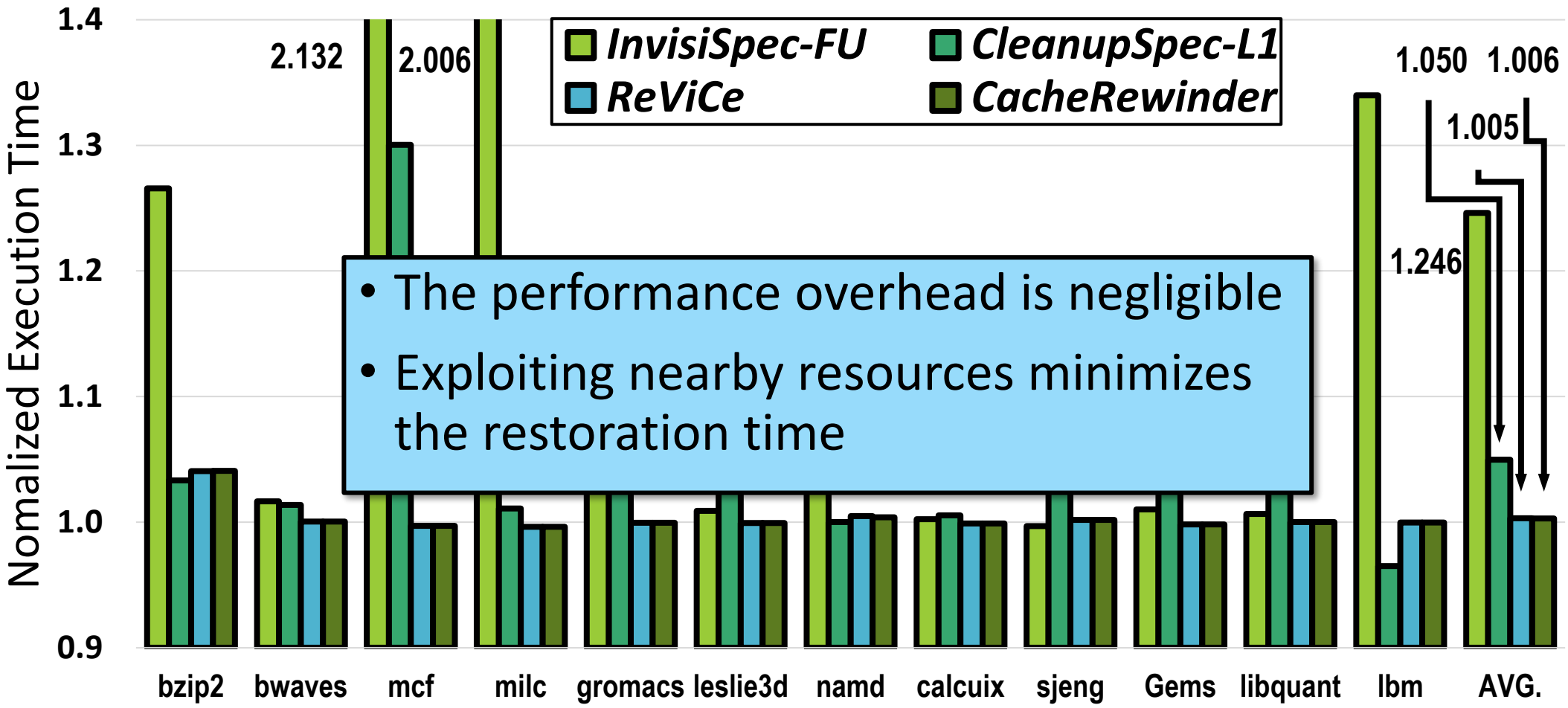
- *Gem5 with SE mode*
- *SPEC2006 benchmarks and modified Spectre PoC code*

Parameter	Configuration
Core	x86 ISA, out-of-order, no SMT, 64 IQ entries, 192 ROB entries, 32 LQ entries, 32 SQ entries, 32 STB entries
Branch Predictor	L-TAGE
L1-I cache	32 KB, 64B line, 8-way
L1-D cache	32 KB, 64B line, 8-way, 8 MSHR entries
L2 cache	256 KB, 64B line, 8-way
Write-back buffer	8 entries
Coherence protocol	MESI

Defense Evaluation

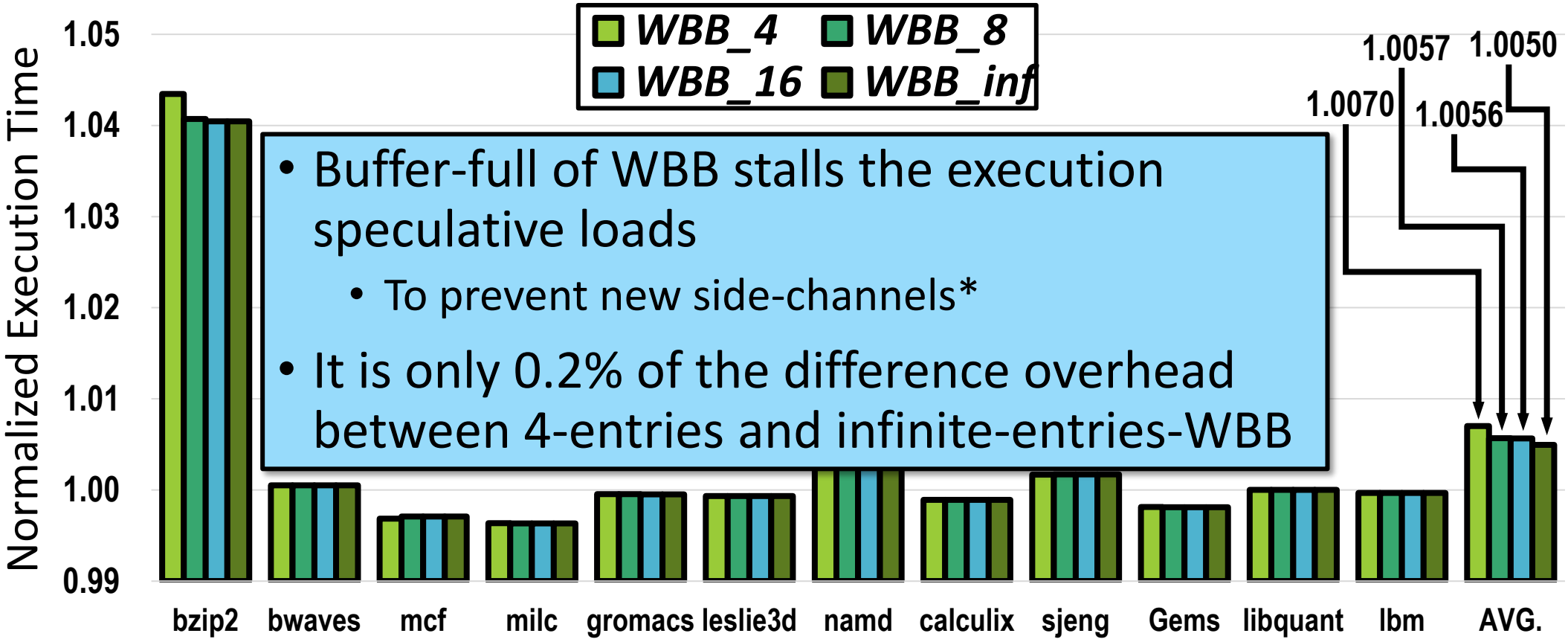


Performance Comparison



- The performance overhead is negligible
- Exploiting nearby resources minimizes the restoration time

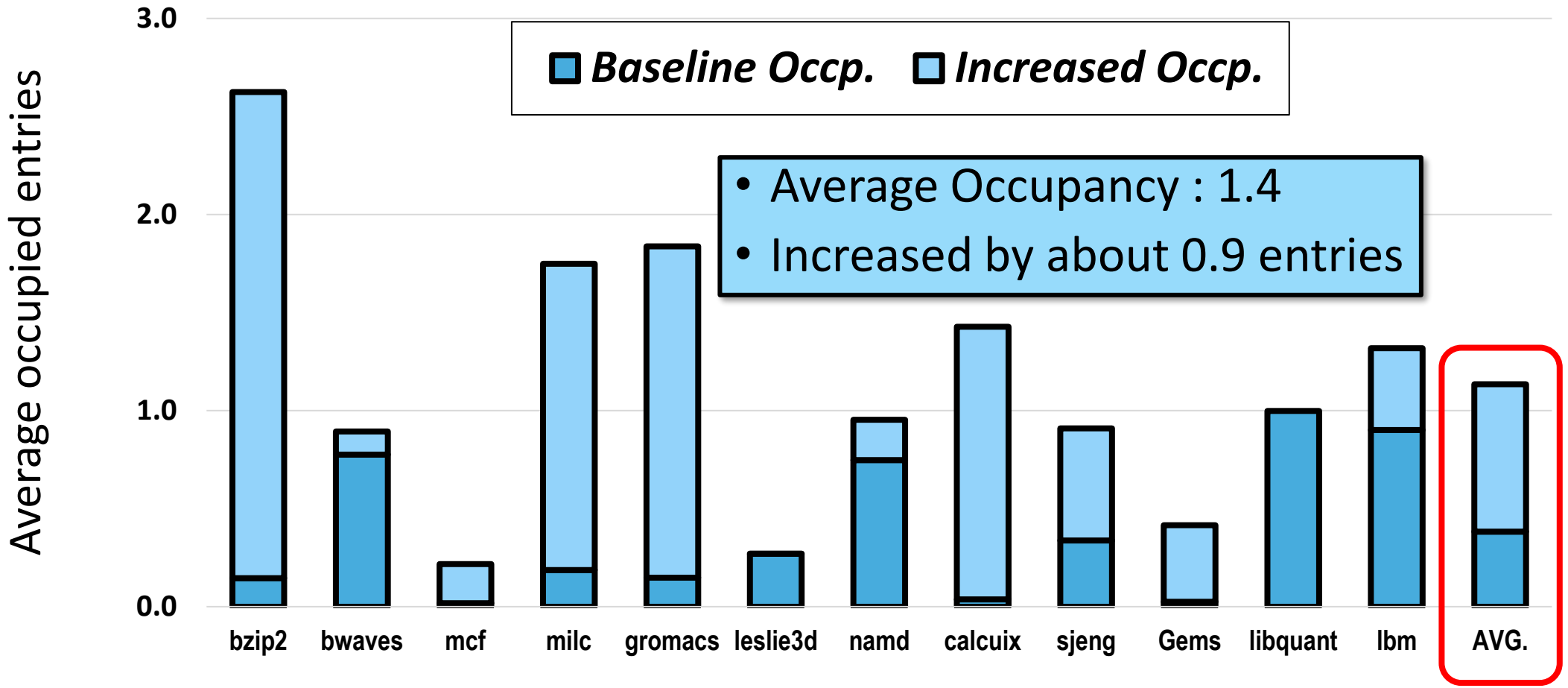
Performance w.r.t the Length of WBB



- Buffer-full of WBB stalls the execution speculative loads
  - To prevent new side-channels*
- It is only 0.2% of the difference overhead between 4-entries and infinite-entries-WBB

Jongmin Lee and Gunjae Koo, "Restore Buffer Overflow Attack; Breaking Undo-Based Defense Schemes", The 36th International Conference on Information Networking (ICOIN 2022)

Utilization of WBB



Storage Overhead

- *Comparison for additional storage*
- *Focus on only Core and L1-D Cache, except L2 Cache*
- *Total overhead of CacheRewinder is minimum, on the same setting.*

Solutions	Core	L1-D Cache	Total
InvisiSpec	28 B	2072 B	2100 B
CleanupSpec	224 B	56 B	280 B
ReViCe	24 B	2500 B	2524 B
CacheRewinder	60 B	131 B	191 B

Conclusion

- ***CacheRewinder – an efficient architectural defense solution***
  - ***Exploiting WBB (write-back buffer) as a restore buffer***
  - ***Implementing STB (speculation tracking buffer) for precise restore/commit operations***
- ***Extremely low performance/cost overhead***
  - ***Negligible performance overhead (0.6%)***
  - ***Low storage overhead (191 Bytes)***

Thank you

CacheRewinder: Revoking Speculative Cache Updates Exploiting Write-Back Buffer

Jongmin Lee, Junyeon Lee, Taeweon Suh, Gunjae Koo

✉ **flackekd@korea.ac.kr**