

# OS Support for Virtualizing Hardware Transactional Memory

Michael M. Swift, Haris Volos,  
Luke Yen, Neelam Goyal,  
Mark D. Hill and David A. Wood

University of Wisconsin—Madison

# The Virtualization Problem of HTM

---

What should happen to a transaction when the Operating System virtualizes a processor or memory?

1. Abort the transaction
2. Go non-speculative
3. Switch to software
4. Virtualize the transaction

# Why virtualize?

---

1. Not virtualizing requires undoing the transaction
  - Aborts take time
  - Blocks other threads until complete
  - Example: aborting a 2048-page TX takes 1,000  $\mu$ s
2. Invoking OS services may require blocking
  - Locks, reversible I/O
3. Limits generality of TM as compared to locks

# Importance of Virtualization

---

Profile results from Sun T1000 32 thread machine, 256 MB/process

| Application |             | Locked Context Switches |        | Page Faults |             |
|-------------|-------------|-------------------------|--------|-------------|-------------|
| Name        | Workload    | Voluntary               | Invol. | Rate        | Locked Rate |
| BIND 9.0    | QueryPerf   | 1494.0                  | 26.0   | 0.00        | 0.000       |
| Apache 2.0  | SpecWeb99   | 555.0                   | 0.5    | 0.50        | 0.000       |
| AOLServer   | ApacheBench | 0.1                     | 2.0    | 0.00        | 0.000       |
| Firefox     | Browsing    | 12.0                    | 1.5    | 0.23        | 0.007       |
| OpenOffice  | Editing     | 0.1                     | 2.0    | 0.25        | 0.130       |

## Virtualizing HTM

---

- We implemented TVM, an OpenSolaris kernel module that:
  - Virtualizes a variant of LogTM-SE
  - Supports context switching and paging
  - Hooks the kernel in 9 places
  - Comprises 1120 lines of code
  - Adds less than 2% runtime overhead

# Outline

---

- Introduction
- Hardware Overview
- OS support for virtualization
- Evaluation
- Conclusion

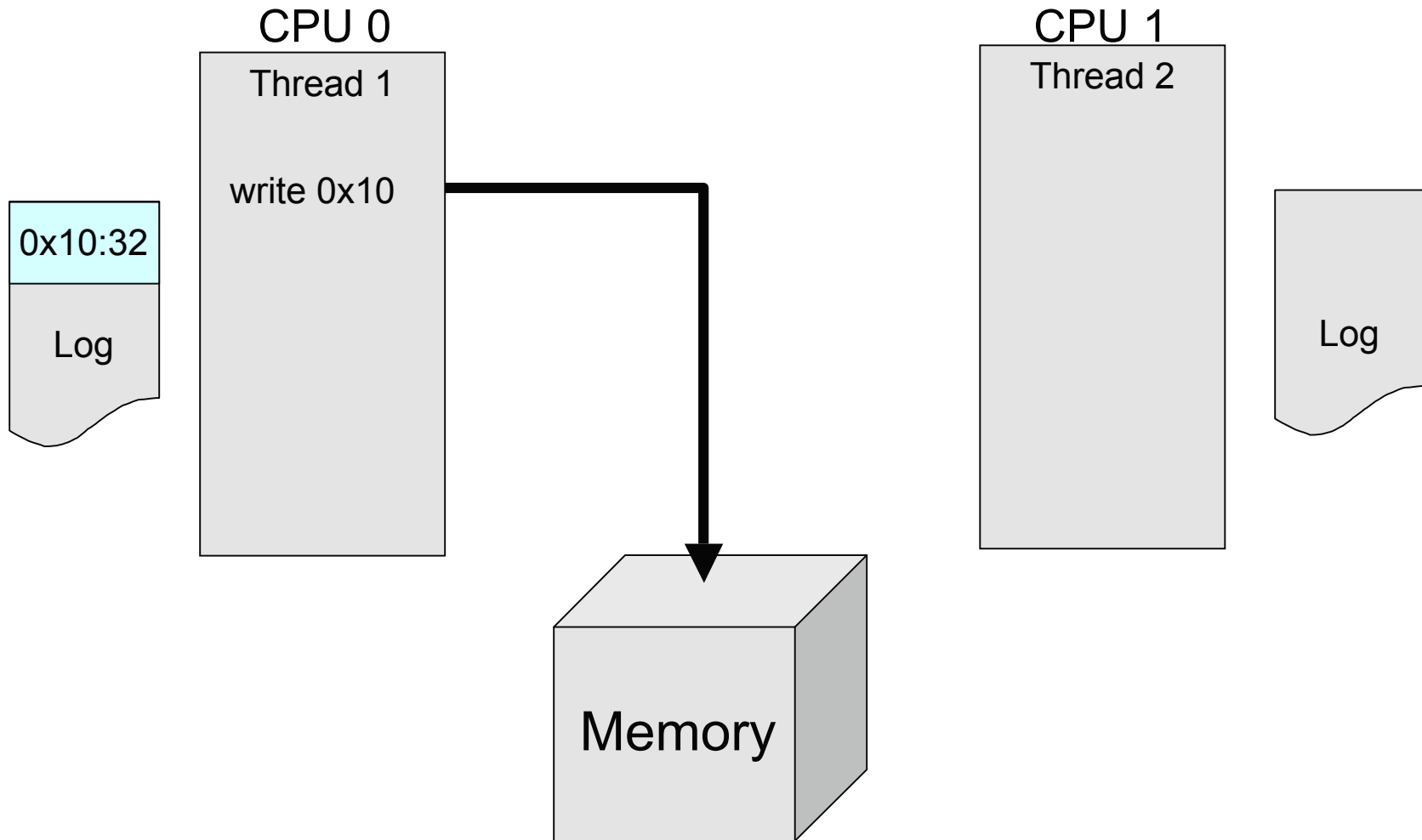
# LogTM-VSE Hardware Overview

---

- Version management:
  - Update in place, old values logged in VM

# LogTM-VSE Regular Operation

---

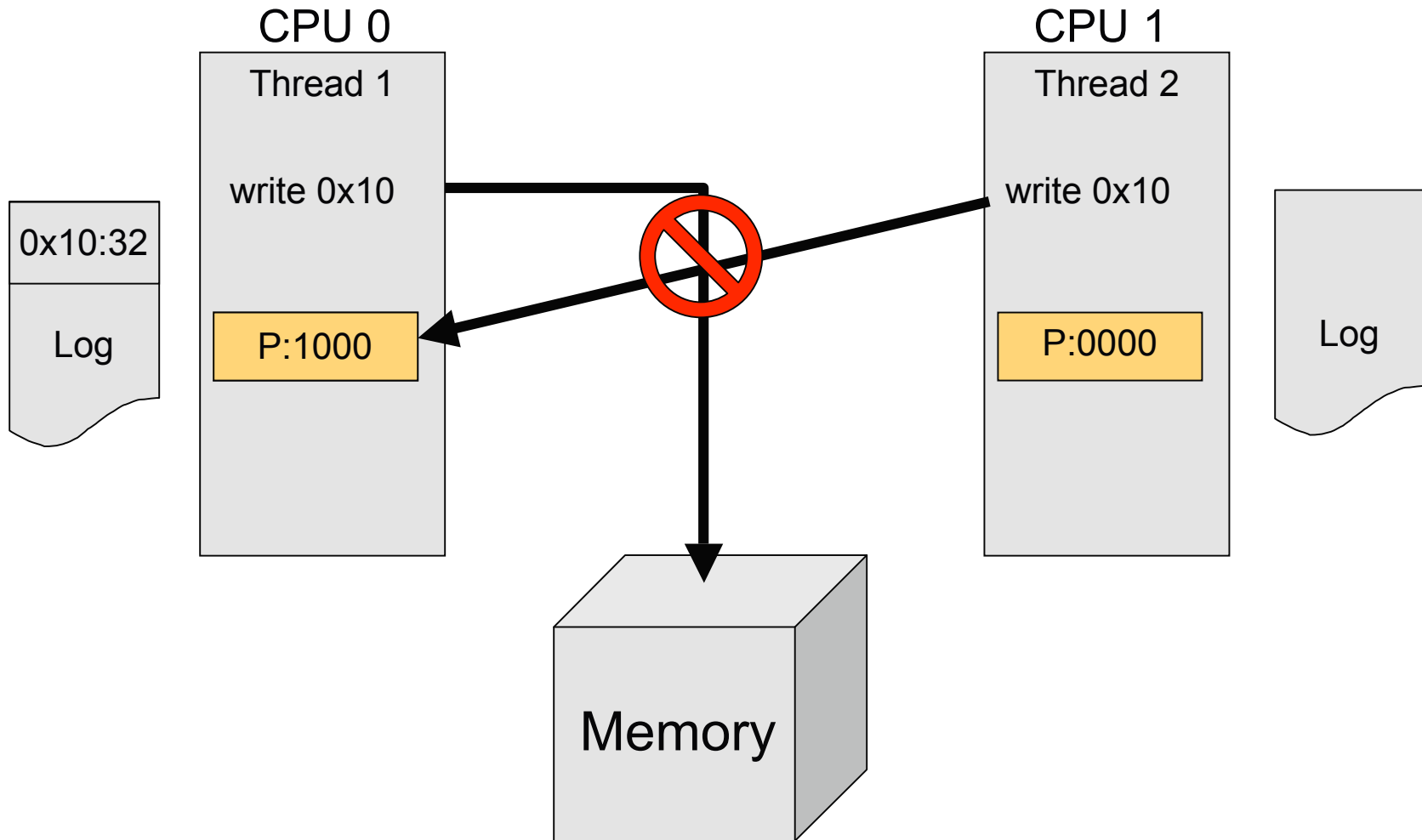


# LogTM-VSE Hardware Overview

---

- Version management:
  - Update in place, old values logged in VM
- Conflict detection:
  - Read/write addresses hashed into signature
  - Coherence reqs. check signature for conflicts

# LogTM-VSE Regular Operation

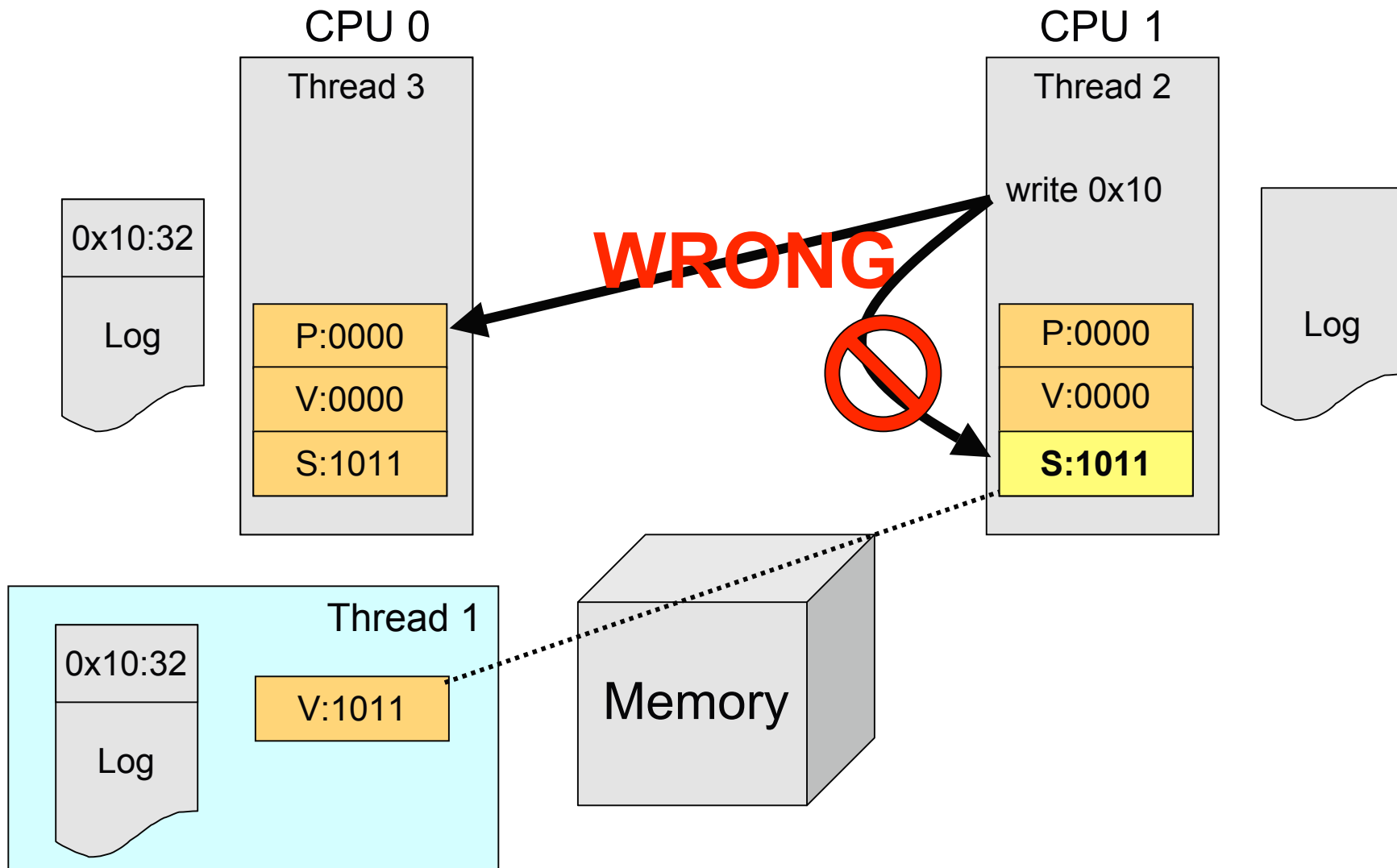


# LogTM-VSE Hardware Overview

---

- Version management:
  - Update in place, old values logged in VM
- Conflict detection:
  - Read/write addresses hashed into signature
  - Coherence reqs. check signature for conflicts
- Virtualization
  - Adds virtual signatures for paging
  - Summary signature for conflict detection on suspended transactions

# Virtualizing a Transaction



# Outline

---

- Introduction
- Hardware Overview
- OS support for virtualization
  - Context switching
  - Paging
- Evaluation
- Conclusion

## Virtualizing TX with TVM

---

- Transaction Virtualization Manager (TVM) enforces isolation of virtualized TX
  - Hooks OpenSolaris in 9 places
  - Computes & distributes summary signatures

# Context Switching

---

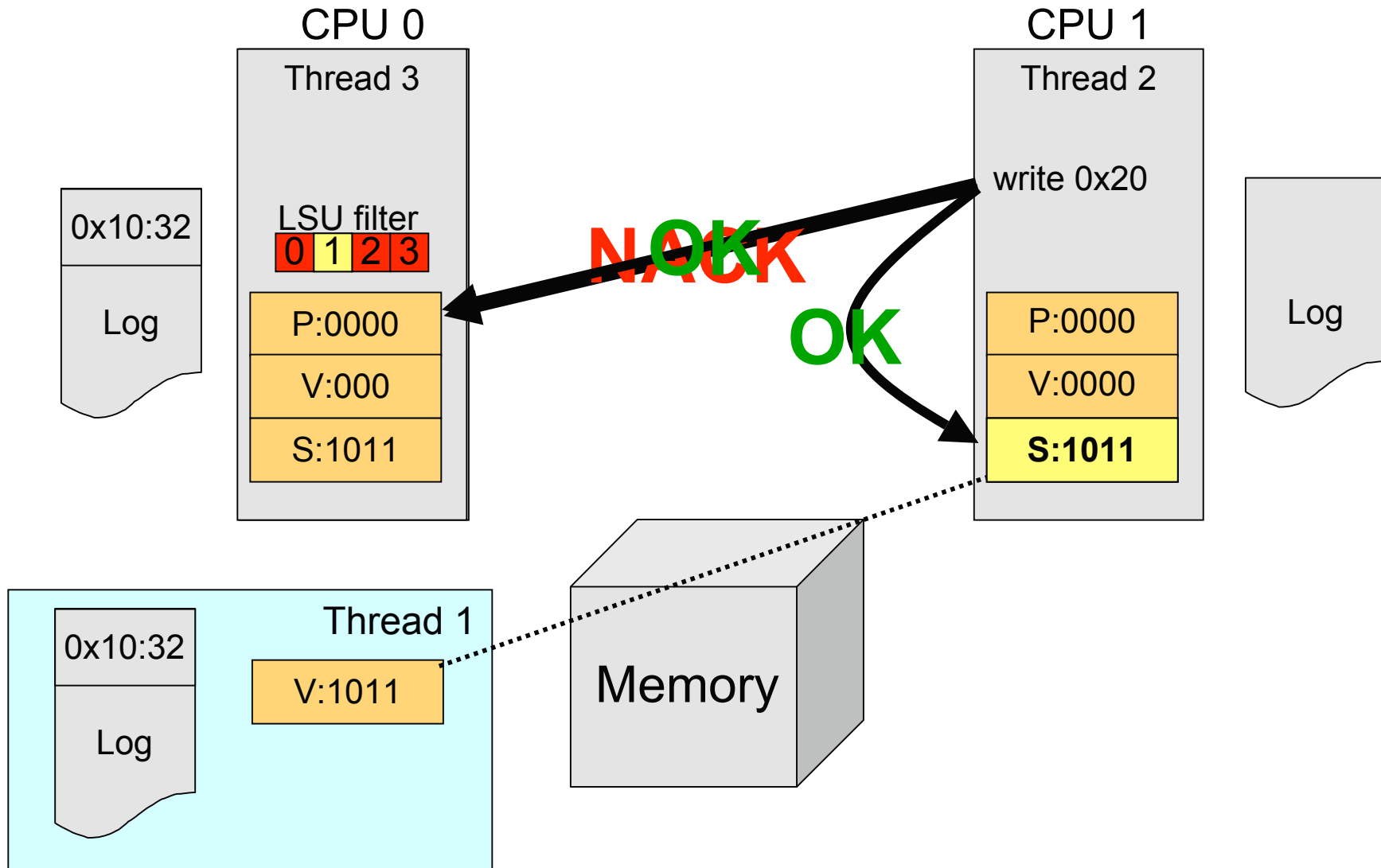
- TVM hooks the OpenSolaris kernel in 4 places
  - `savectx()`: save virtual signature, distribute new summary
  - `restorectx()`: restore virtual signature
  - **Complete virtual transaction**: distribute new summary
  - **Summary conflict**: forward to contention manager

# Optimizations

---

- Why distribute new summaries synchronously?
  - **Lazy completion** defers update on transaction complete
  - **Lazy summary update** defers update on thread suspend to reduce latency

# Lazy Summary Update



# Paging Transactional Data

---

- What happens when the kernel changes the address mapping for a page?
  - On paging
  - On copy-on-write

## Virtualize Transactions!

- Details in paper

## TVM Summary

---

- TVM manages summary signatures to virtualize transactions
- TVM is implemented as an OpenSolaris kernel module
  - 1120 lines of code
    - Context Switch: 325 lines
    - Paging: 265 lines
    - Common: 530 lines
  - TVM invoked from only 9 locations

# Outline

---

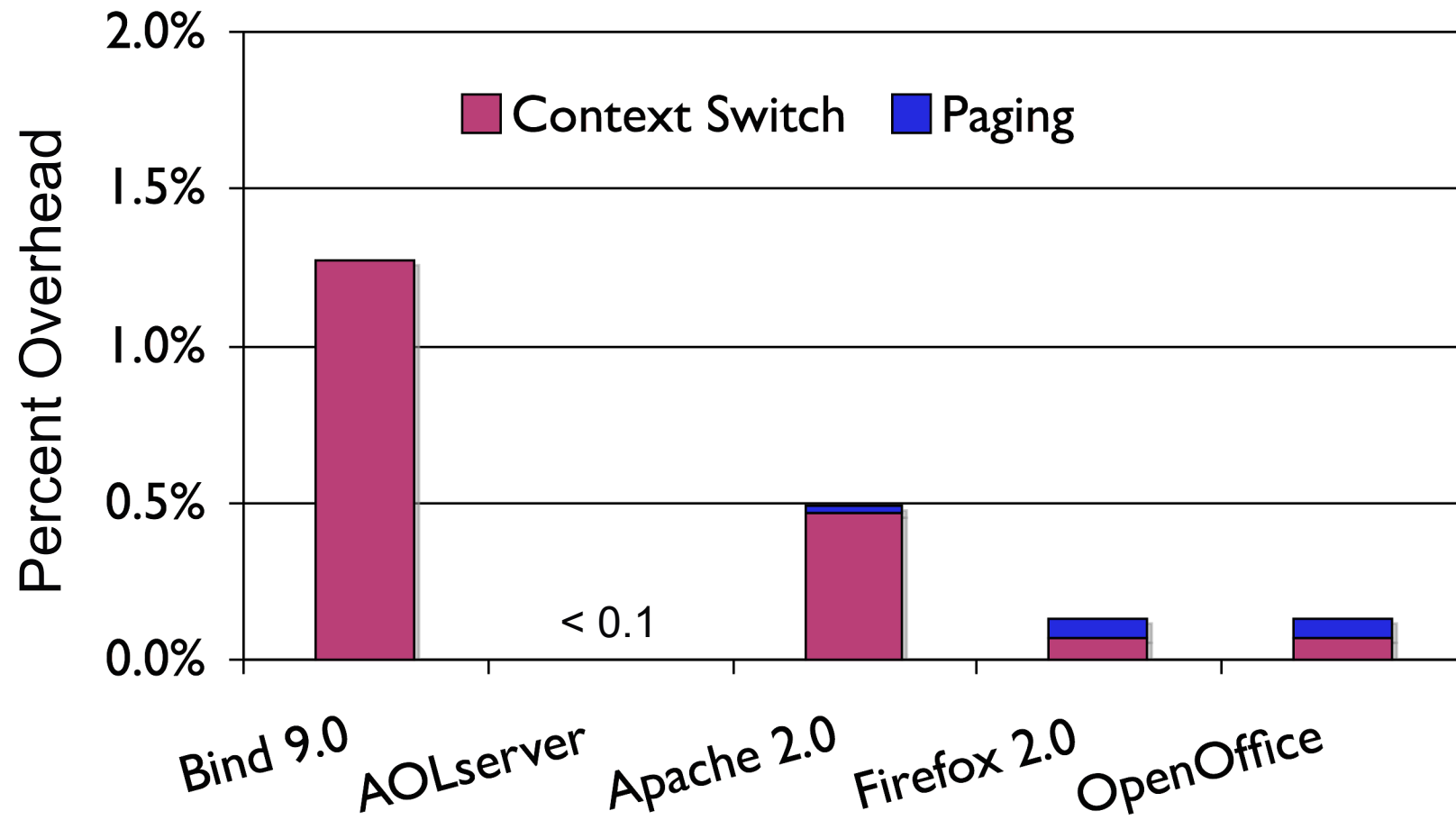
- Introduction
- Hardware Overview
- OS support for virtualization
- Extending virtualization to a VMM
- Evaluation
- Conclusion

# Evaluation

---

- Profile lock-based critical sections
  - Sun T1000 platform - Niagara (8-core 4-way SMT)
- Microbenchmark LogTM-VSE/TVM in simulation
  - GEMS/LogTM-VSE full-system simulation
    - 32 in-order SPARC cores
    - Memory latencies match T1000
    - 2048-byte signatures
- Predict overhead
  - Profiled event count x simulator measured time

# Virtualization Overhead



## Summary

---

- We implemented TVM, an OpenSolaris kernel module that:
  - Supports context switching and paging
  - Hooks the kernel in 9 places
  - Comprises 1120 lines of code
  - Adds less than 2% runtime overhead
- Our design supports execution in a VMM and virtualization by a VMM

# Questions?

---

For more information:

[swift@cs.wisc.edu](mailto:swift@cs.wisc.edu)

[www.cs.wisc.edu/multifacet/logtm](http://www.cs.wisc.edu/multifacet/logtm)