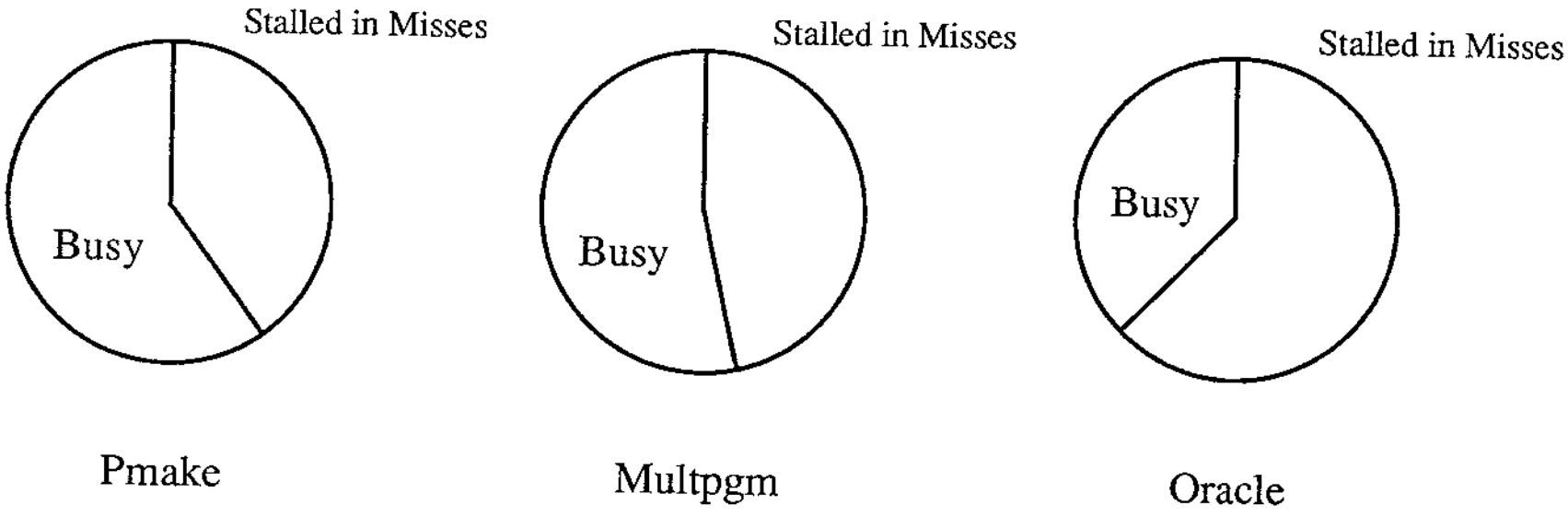
Losses Due to Misses in Shared-Memory Multiprocessors



What We Found Out

- OS Misses: Instructions, Block Ops., Migration
- OS Self-Interference in the I-Cache
- OS Synchronization: High Locality, Low Contention

Outline

1. Experimental Environment

2. Characterization of OS Cache Performance

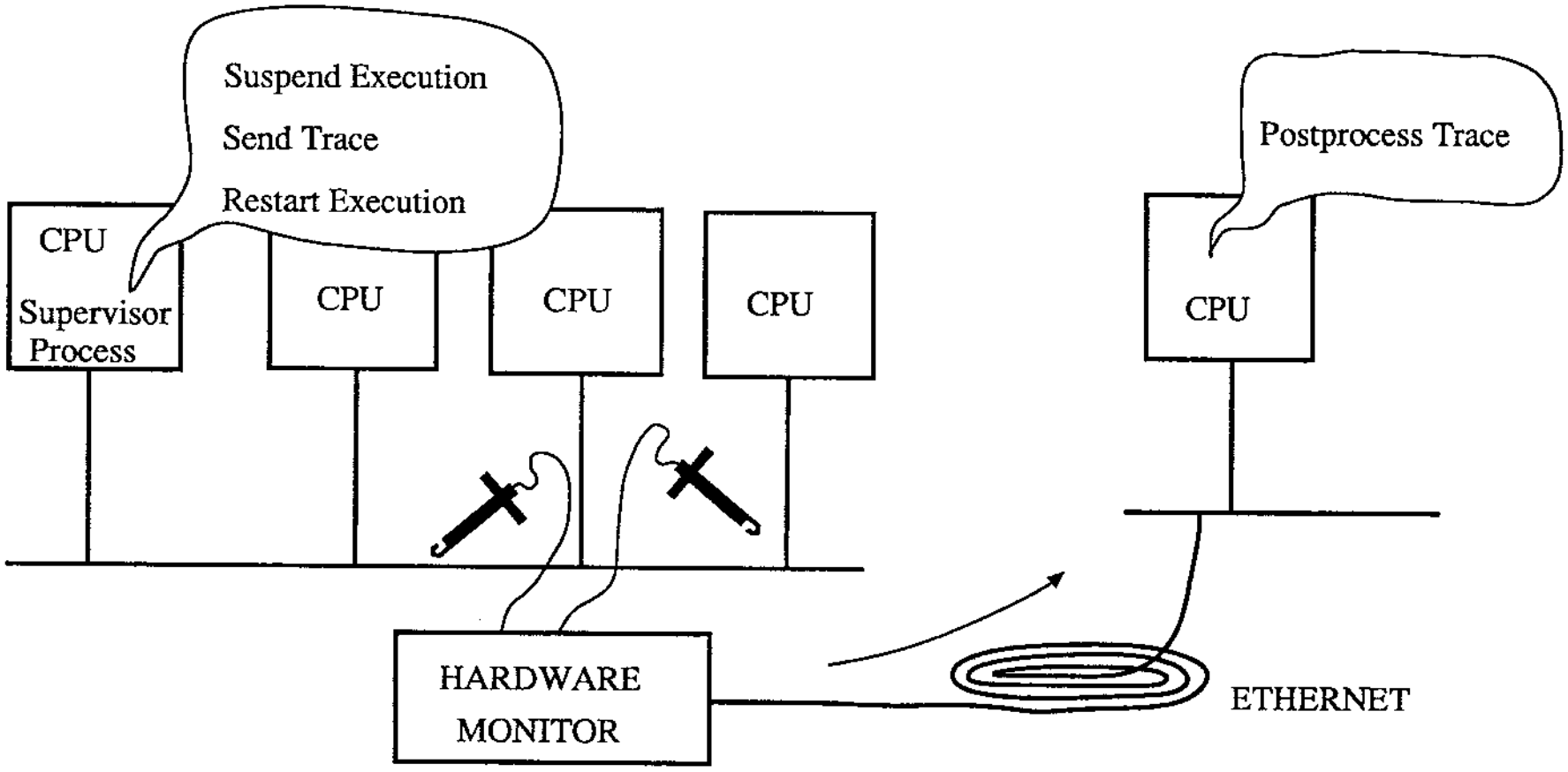
3. Characterization of OS Synchronization

Real Multiprocessor + Hardware Monitor

- 33-MHz 4-CPU Silicon Graphics Station (R3000 CPU)
  - Shared Memory
  - 64 KB I-Cache
  - 64 KB + 256 KB D-Cache
  - 30-Cycle Cache Miss Penalty
  - IRIX (UNIX System V): Multithreaded/Symmetric

- **Total Info: Hardware Monitor**

Collecting Address Traces

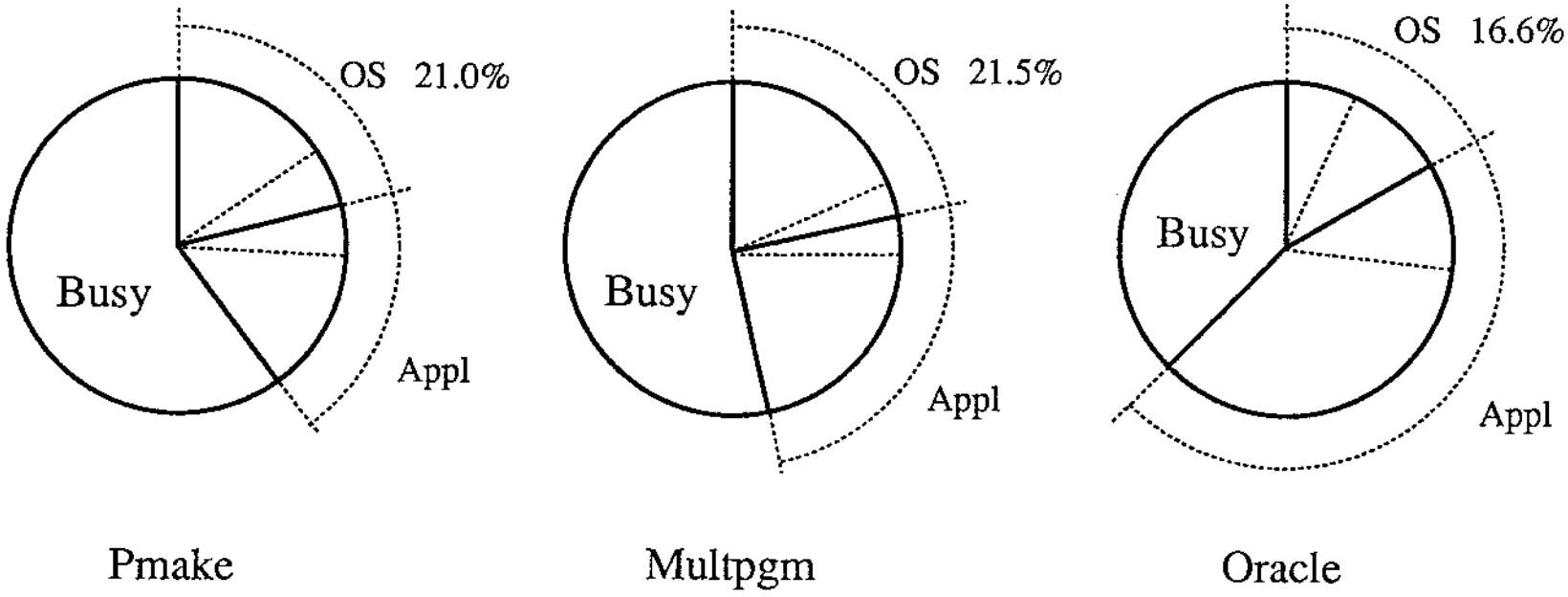


- 1-2 Minutes Worth of Uninterrupted Trace

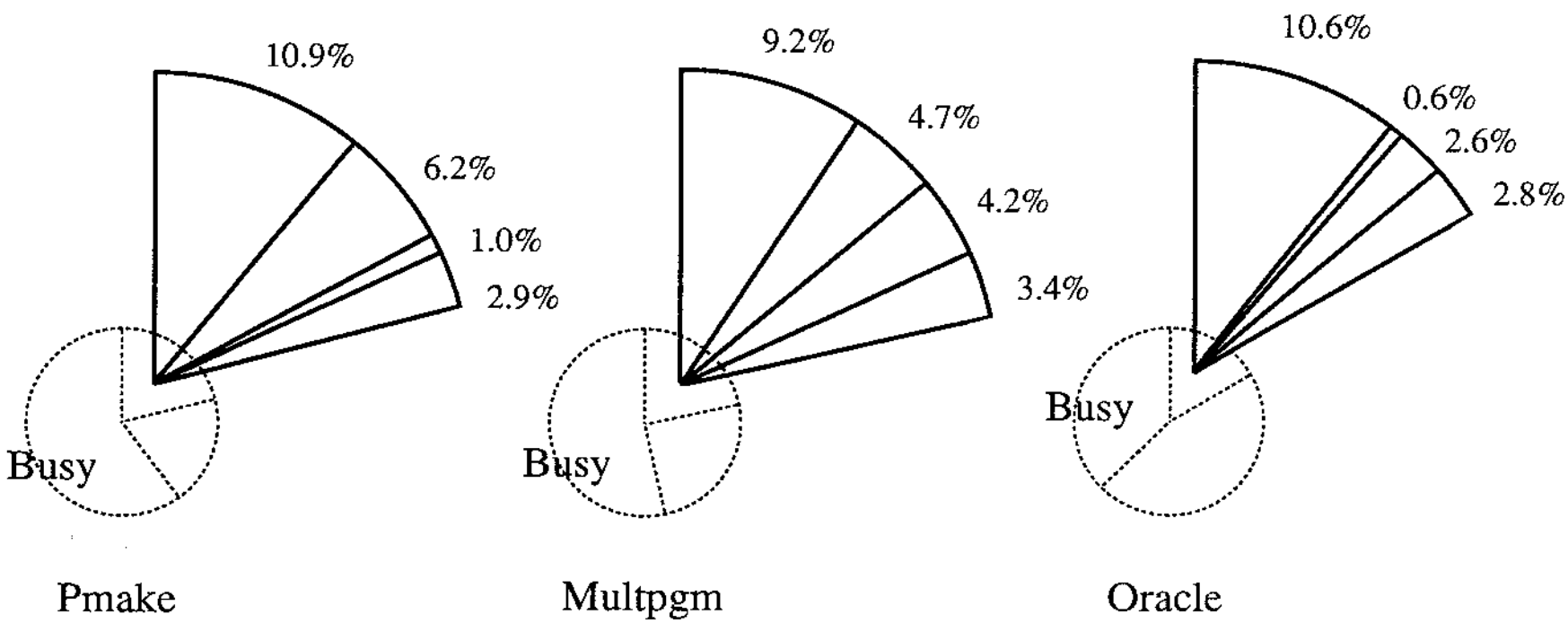
Parallel Workloads Traced

- **Pmake** Parallel Compile of 56 C Files / 480 Lines Each
- **Multpgm**
  - Pmake
  - +
  - 5 Screen Edit Sessions
  - +
  - Mp3d: 3-D Particle Simulator from Aeronautics
- **Oracle** Cached TP1 Benchmark on an Oracle Database

OS Misses: Stall Time



OS Misses: Components of the Stall Time



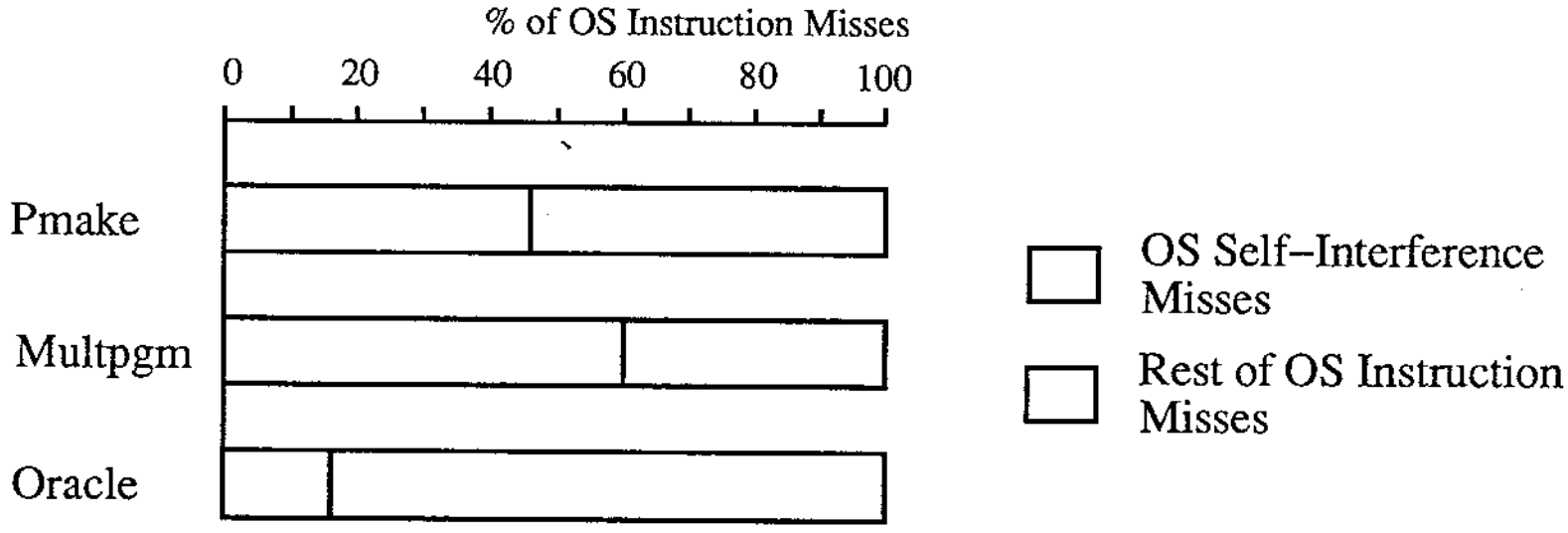
I–Misses

D–Misses: Block Ops.

D–Misses: Proc. Migration

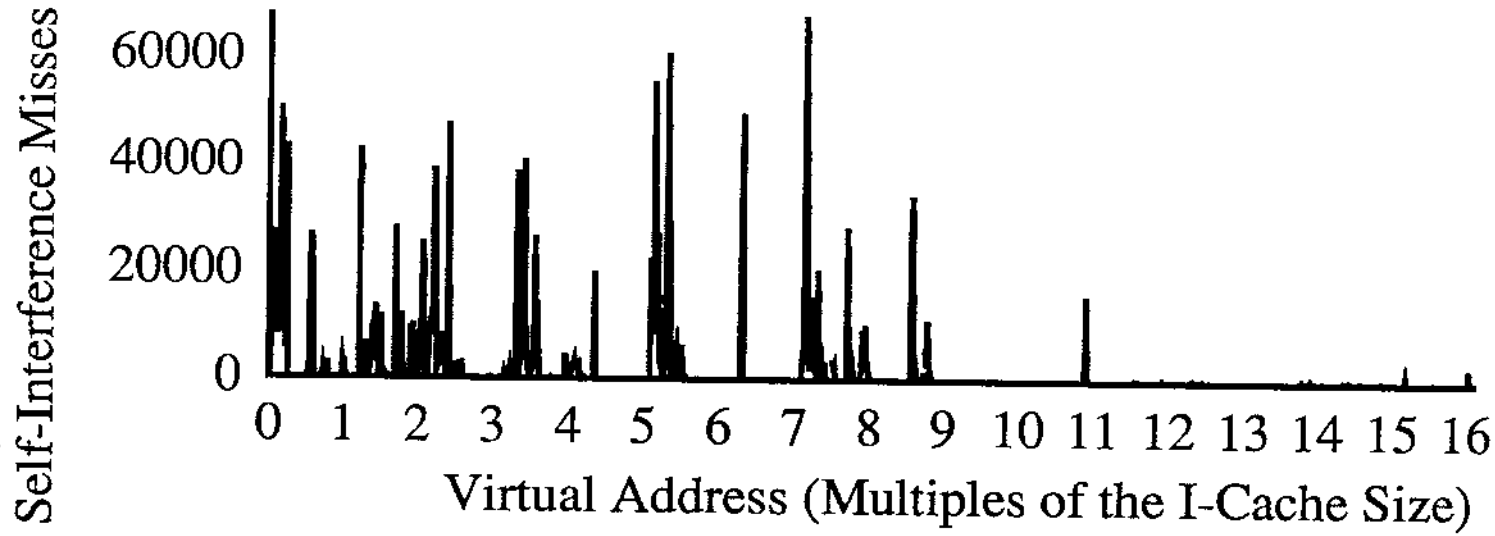
D–Misses: Other

A Closer Look at OS Instruction Misses



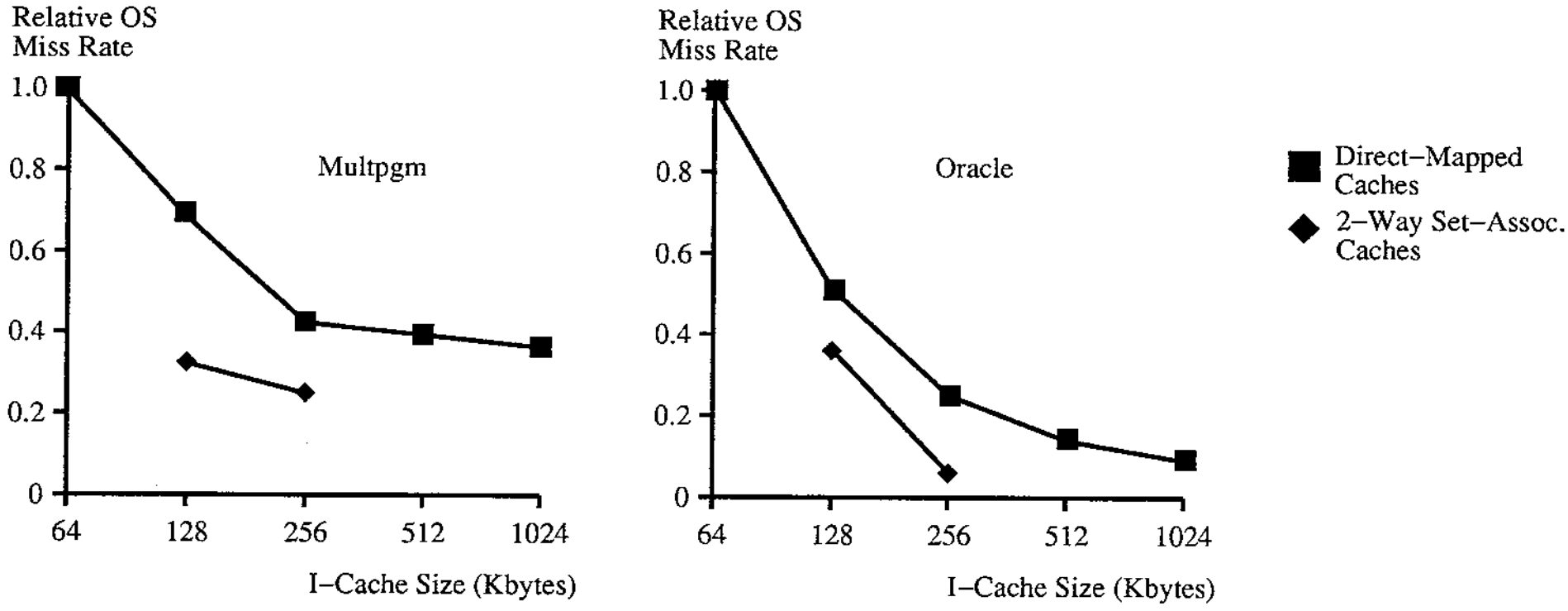
- OS Code often Self-Interferes in the Cache

OS Self-Interference Misses: Spatial Distribution



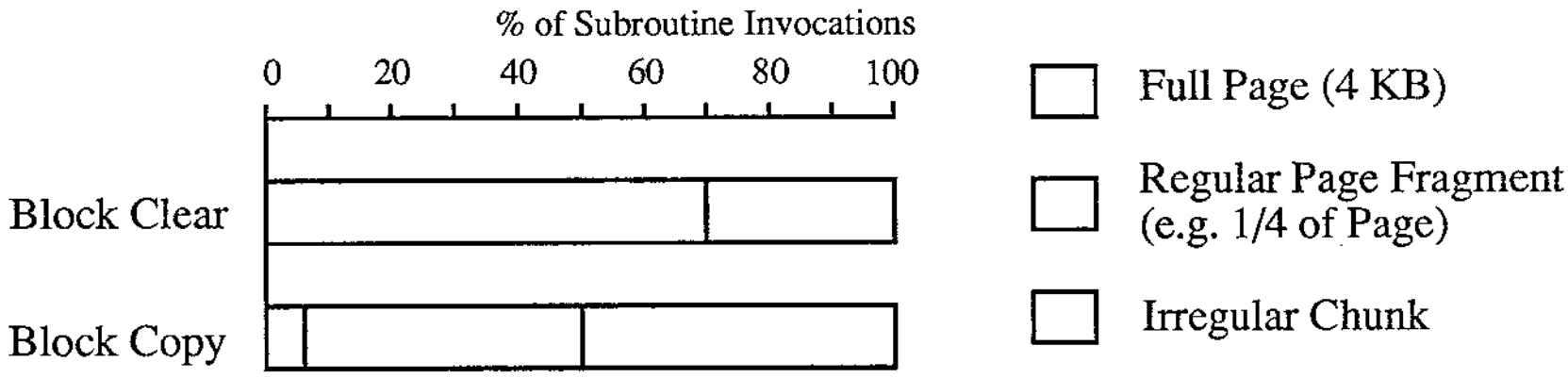
- Concentrated in Certain Routines: Try Placement Opts.

Eliminating I-Misses by Changing Cache Parameters



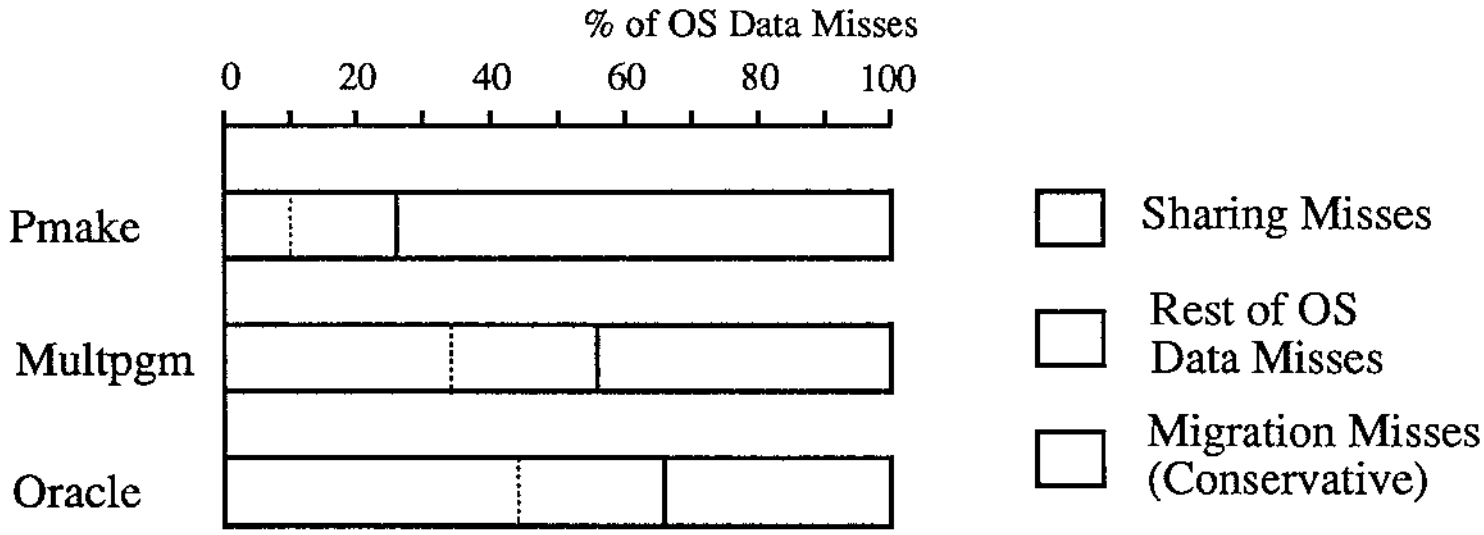
- Associativity Has Noticeable Impact

OS Block Operations: Sizes of the Blocks



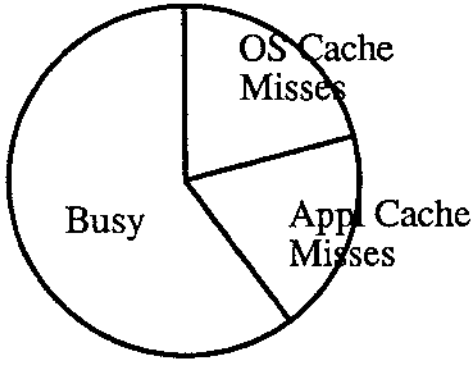
- Large Regular Blocks Dominate: Try Data Prefetching
- Try Cache Bypassing

OS Data Misses: Sharing and Migration

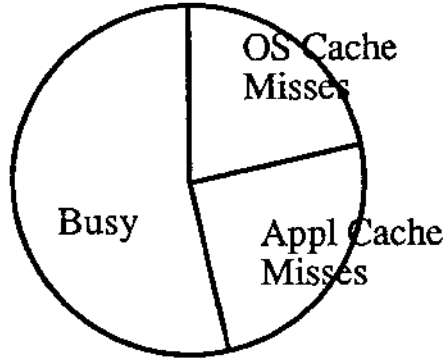


- Sharing Misses => Big Caches less Effective than Expected
- Migration Misses => Try Cache Affinity Scheduling

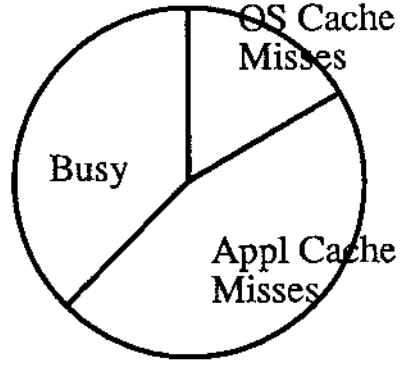
OS Synchronization Accesses: Stall Time



Pmake



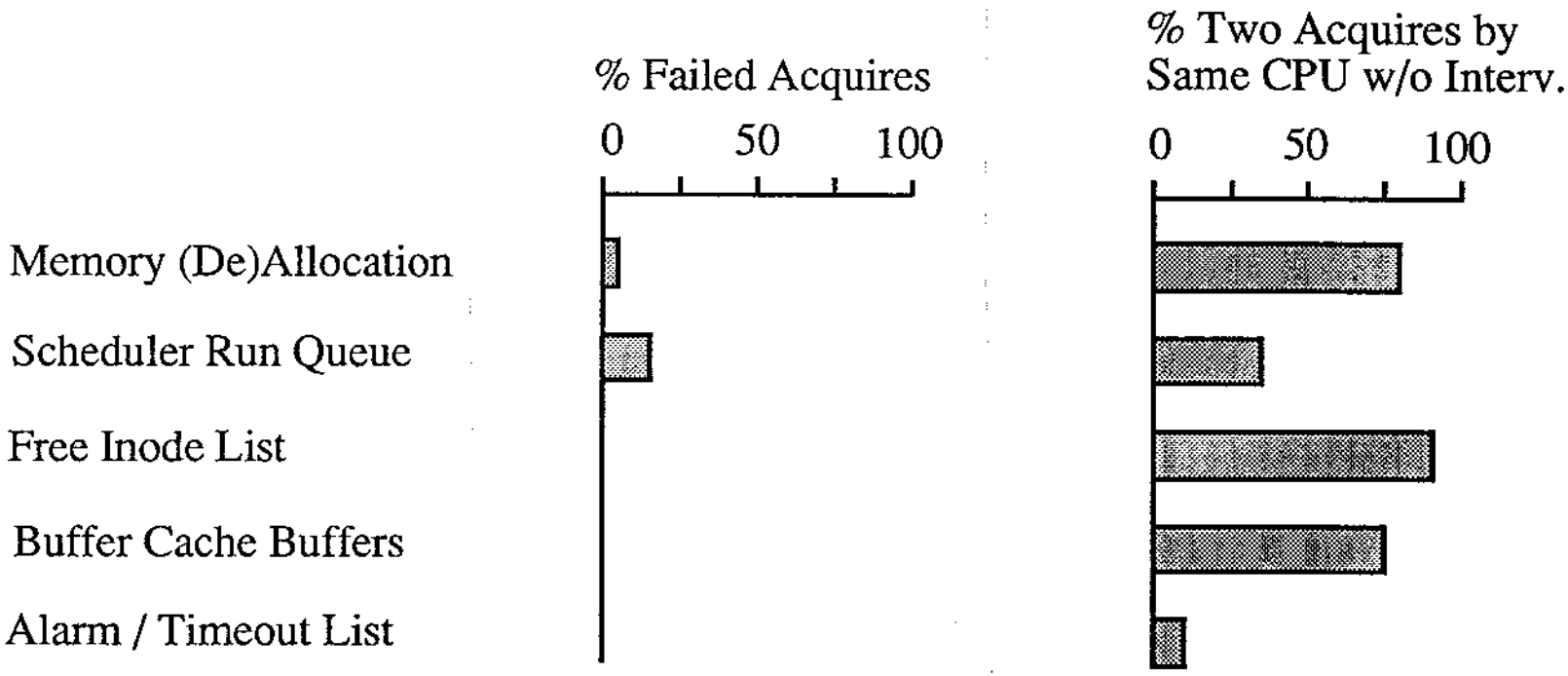
Multpgm



Oracle

Patterns of Access to OS Locks

What the Lock Protects:



- Low Contention
- High Locality

Summary

- **OS Misses:** Instructions → Code Layout / Associat.

  Block Ops. → Prefetching / Bypassing

  Migration → Affinity Scheduling

- **OS Synch.:** Small Cost → Cache Support for Locks

Implications for Hierarchical Machines

- Replicate OS Executable across Clusters
- Support Efficient Intercluster Block Transfers
- Distribute Run Queue across Clusters
- Distribute Popular Locks across Clusters