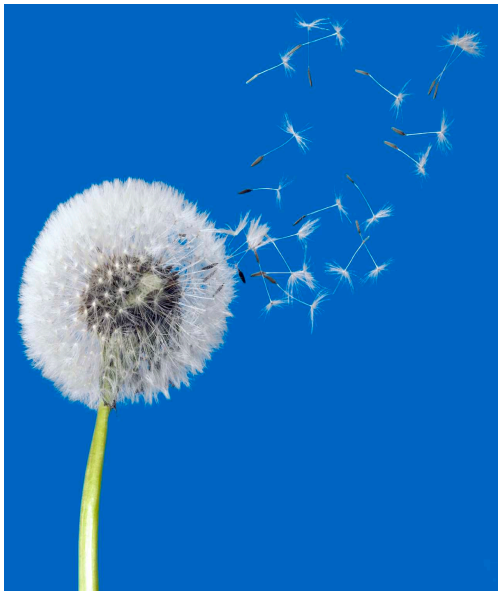
Go further, faster

CLUEBOX: A Performance Log Analyzer for Automated Troubleshooting

Niranjan Thirumale et al.
NetApp, Inc.

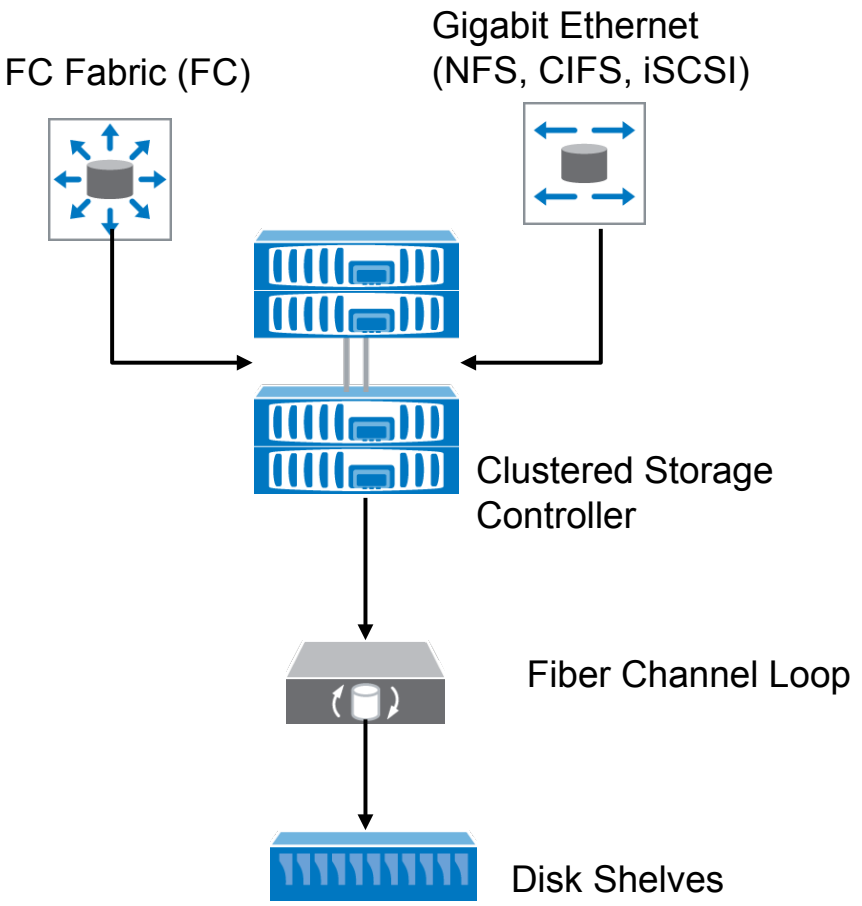
WASL 2008



The Problem

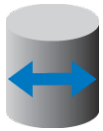
- Complex system deployments have ad-hoc but effective mechanisms for:
  - Data collection
  - Monitoring
  - Alerting
- …but do not have good analysis tools
- What's available: detailed performance counters. Can we mine that to:
  - Characterize and identify workload?
  - help in the performance troubleshooting workflow?

The Environment

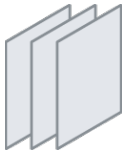


File System Software

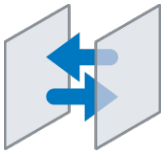
Aggregates
RAID groups



Flexible Volumes

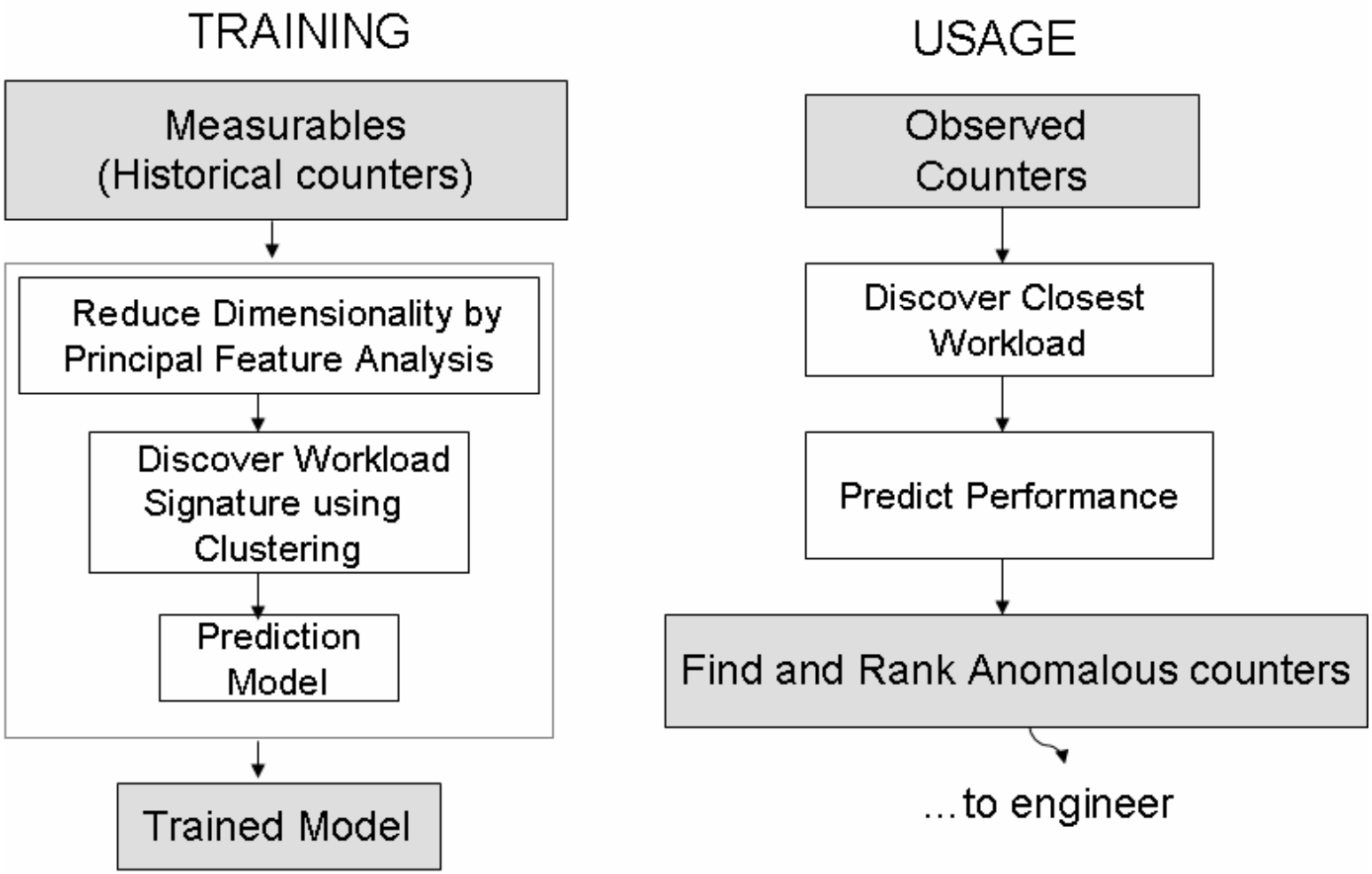


Snapshot Copies

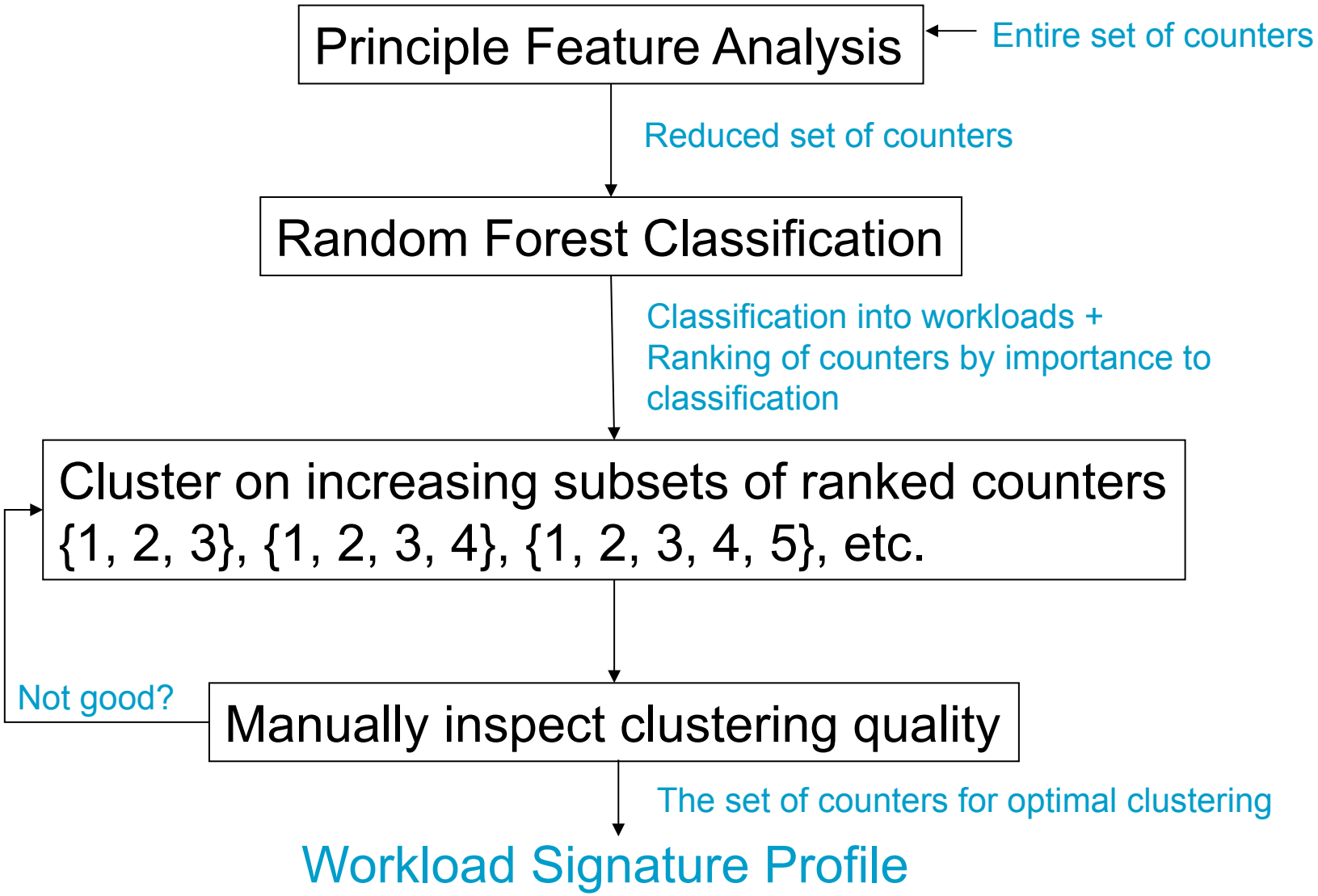


Replica Services

CLUEBOX Workflow



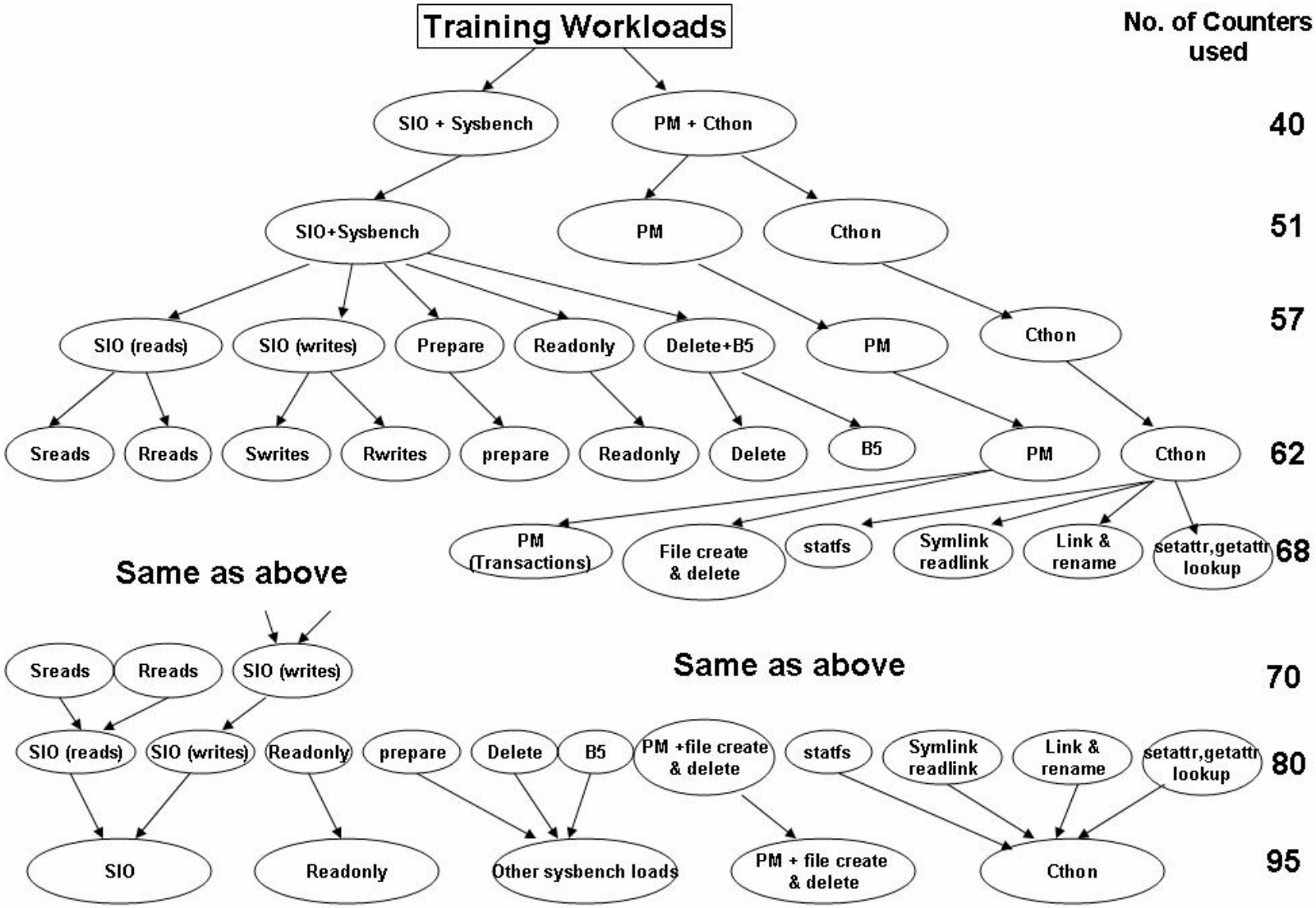
Workload Identification



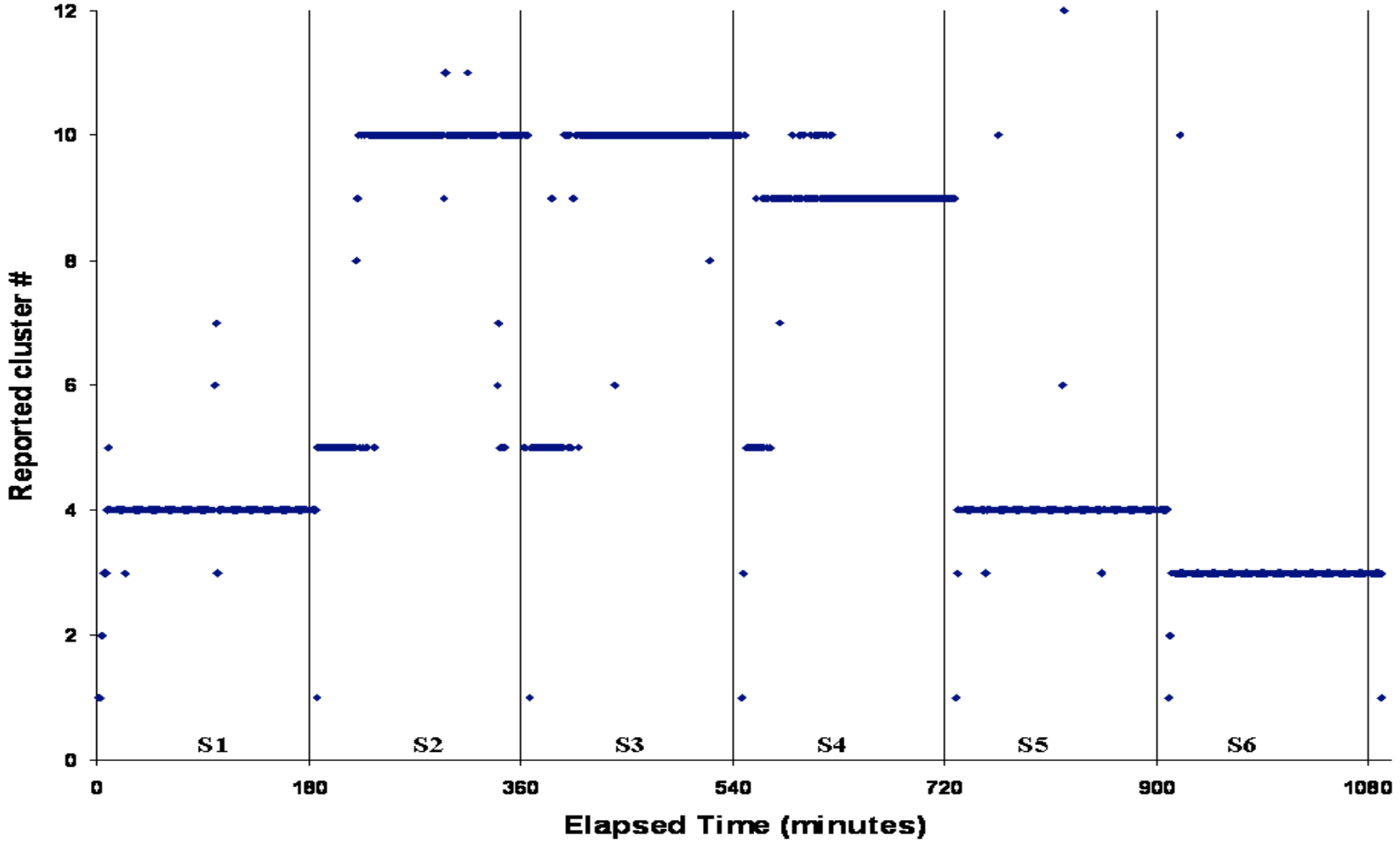
Workload Identification: Evaluation

- SIO
  - 6 workloads, each with varying read/write ratios and sequential/random I/O ratios
- Cthon
  - 5 workloads, each exercising different sets of file system metadata operations
- PostMark
  - 2 workloads, each operating on different number of files and different file size ranges
- Sysbench
  - 5 workloads, each exercising database-like workloads

Workload Identification: How Many Counters Do We Really Need?

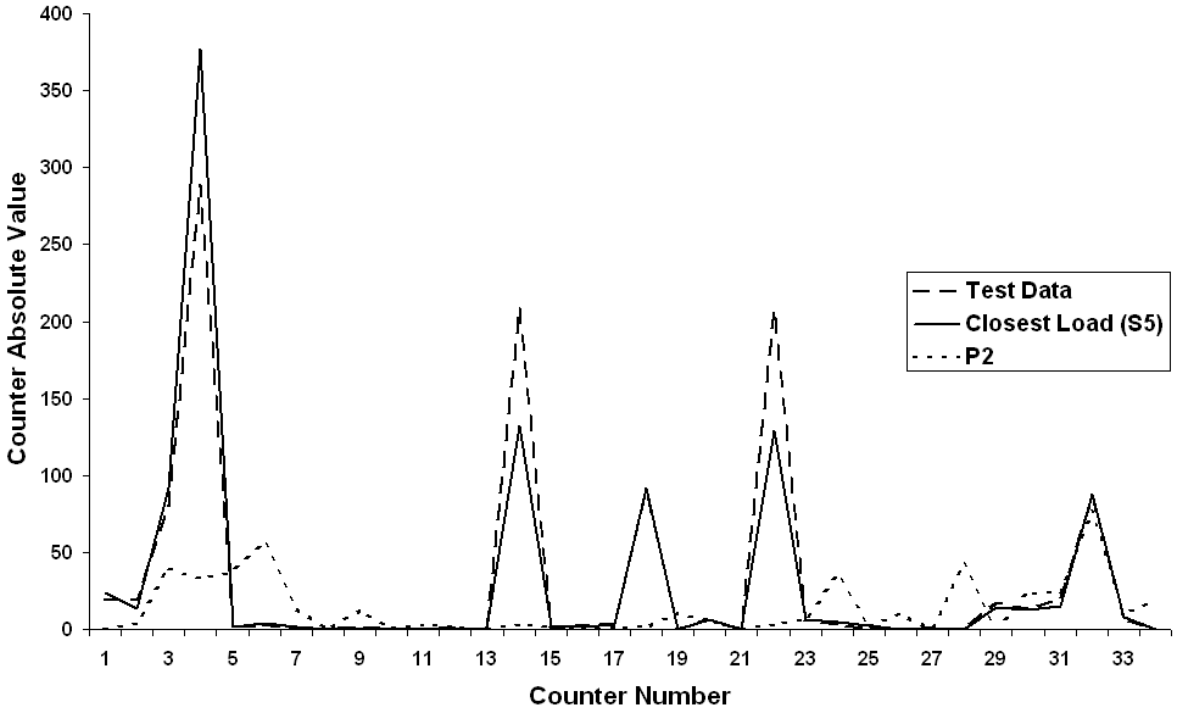


Workload Identification: Results



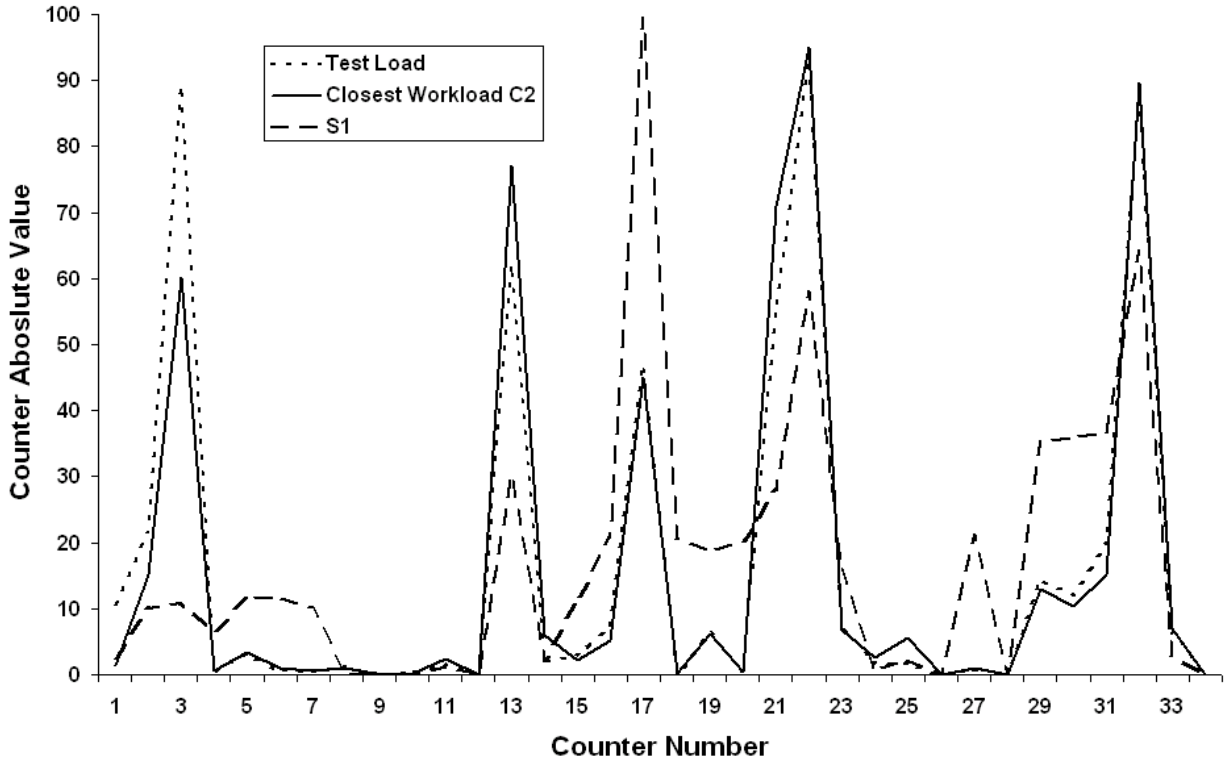
S1: Read 80%, Seq 80%
S2: Read 20%, Seq 80%
S3: Read 30%, Seq 90%
S4: Read 30%, Seq 10%
S5: Read 80%, Seq 90%
S6: Read 80%, Seq 10%

Workload Identification: Results



- Known workloads training data contains:
  - S5: SIO, Seq Reads = 72%, Seq Writes = 18%, Random Reads = 8%, Random Writes = 2%
  - P2: PostMark, # Files = 10^5, File Size = 1KB to 100KB
- Test workload: SIO, Seq Reads = 72%, Seq Writes = 8%, Random Reads = 18%, Random Writes = 2%

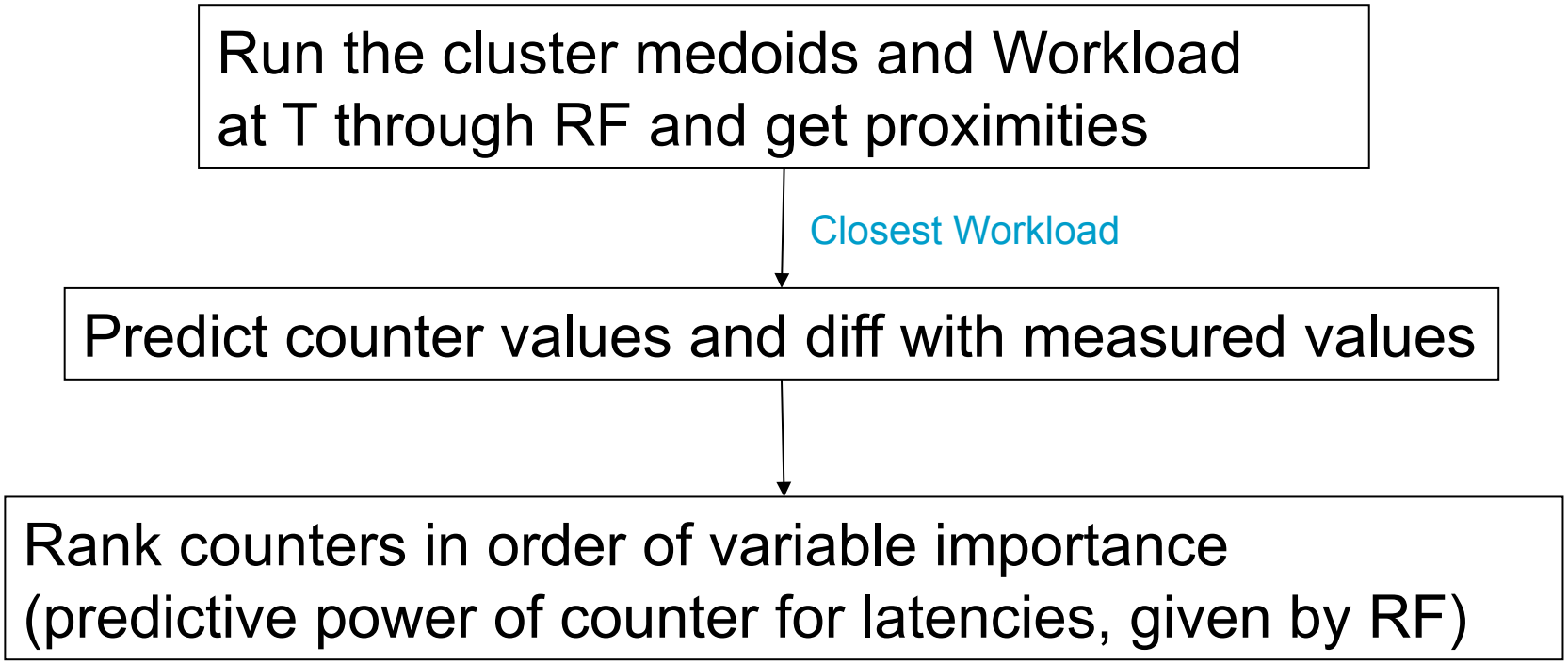
Workload Identification: Results



- Known workloads training data contains
  - C2: Cthon Test 8 (symlink, readlink), # files = 1000, #symlinks = 1000
  - S1: SIO, Seq Reads = 64%, Seq Writes = 16%, Random Reads = 16%, Random Writes = 4%
- Test workload: Cthon Test8 with different parameters

Anomaly Detection

- User: "the performance of the system was fine at Time T. I haven't changed a thing since then, but now the latencies are very high"



Anomaly Detection

- A handful of top-ranked counters were enough
- Anomalies in counters + Non-anomalous counters were both important
- An expert scanning this list diagnosed the root-cause rapidly
  - Conducive to a rule-based system at this point?

Anomaly Detection: fsck-like load

Counter	Deviation from Prediction
Ifnet:e0:total_packets	-99.4%
Processor0:hard_switches	-63.6%
System:cpu_busy	305.6%
wafl:restart_msg_cnt:BACK DOOR	99947%
Readahead:total_read_reqs	-100%
Wafl:buf_hash_hit	7088%
Wafl:new_msg_cnt:BACKD OOR	972.9%

- Readahead counters have dropped, so not a replication load
- Absence of anomaly in cp_phase_times counter, so no high amount of destaging from cache
- Buf_hash_hit is extremely high, which cannot be explained by repeated mounts and unmounts
- Only an fsck or similar scanner can explain the data

Anomaly Detection: Local I/O load

Counter	Deviation from Prediction
Testaggr:cp_read_blocks	129%
Ifnet:e0:total_packets	8.2%
System:cpu_busy	359.9
Wafl:buf_hash_hit	3045
Testaggr:total_transfers	164.2%

- system:cpu_busy higher than predicted indicative of some other load
- ifnet:e0:total_packets is not different, so not a network I/O load
- No other significant anomalies, so not a bookkeeping activity
- testaggr:total_transfers higher imples significant amount of new I/O
- Only a separate I/O workload local to the storage controller can explain the symptoms

What Next

- Test CLUEBOX on customer environments
  - Real performance problems
  - Real workloads
- Incorporate CLUEBOX into NetApp's performance log analysis workflows
  - AutoSupport Database
  - Support Case Resolution
  - Duplicate case/bug report Identification
- Figure out how to avoid re-training if system configurations are changed (e.g., a few more disks added)