

ANALYSIS OF PORTFOLIO-STYLE PARALLEL SAT SOLVING ON CURRENT MULTI-CORE ARCHITECTURES

Martin Aigner¹, Armin Biere², Christoph M. Kirsch¹,
Aina Niemetz², Mathias Preiner²

¹University of Salzburg

²Johannes Kepler University Linz

Pragmatics of SAT 2013

July 8, 2013

PREVIEW

- Our Goal:
 - identify scalability bounds for Plain Parallel Portfolio (PPP) SAT solving
 - when and why does PPP scale?

OUR APPROACH

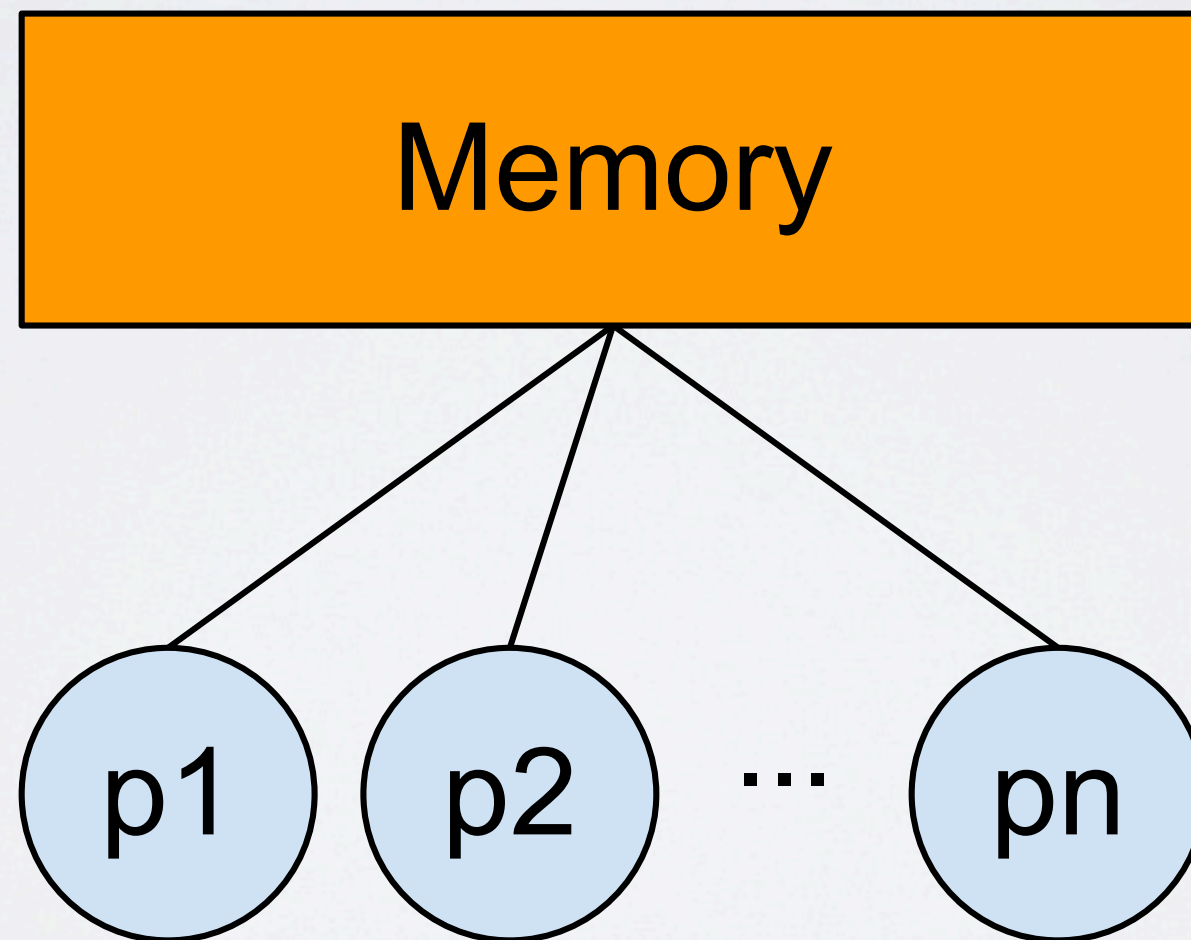
- We measure the *slowdown* of identical solvers
 - on the same instance
 - on shared-memory multi-core hardware
- And identify the cause of the slowdown

OVERVIEW

- Machine model
- Experimental setup and metrics
- Results
- Conclusion

MACHINE MODEL

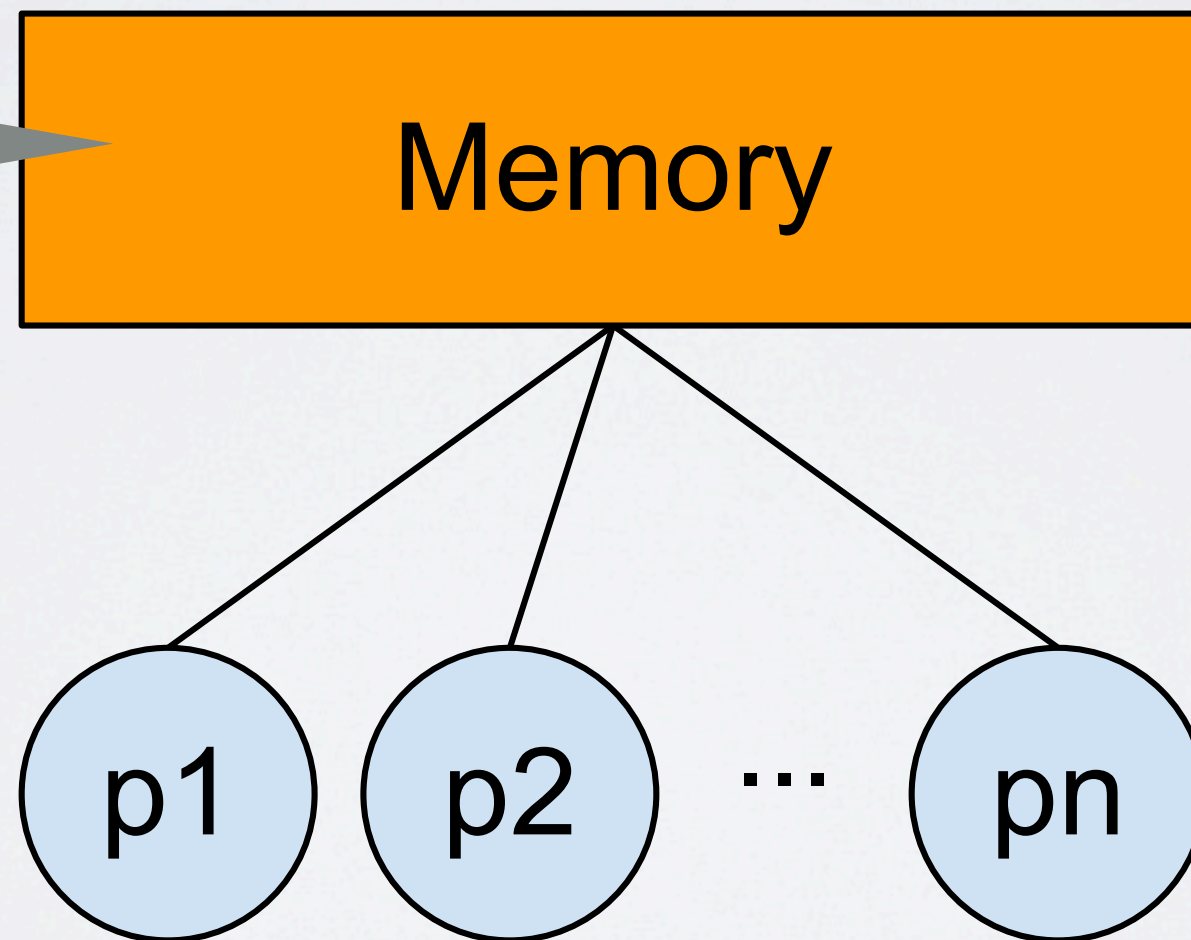
- Simplest form: PRAM model



MACHINE MODEL

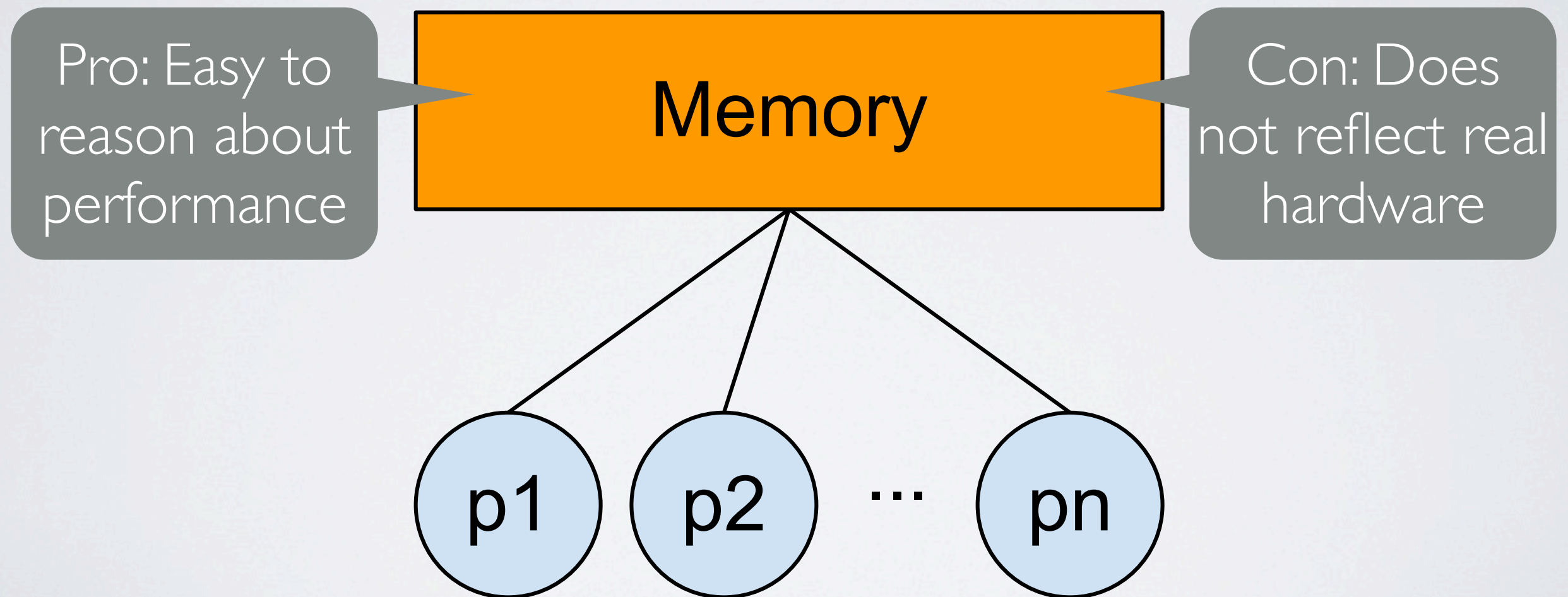
- Simplest form: PRAM model

Pro: Easy to
reason about
performance



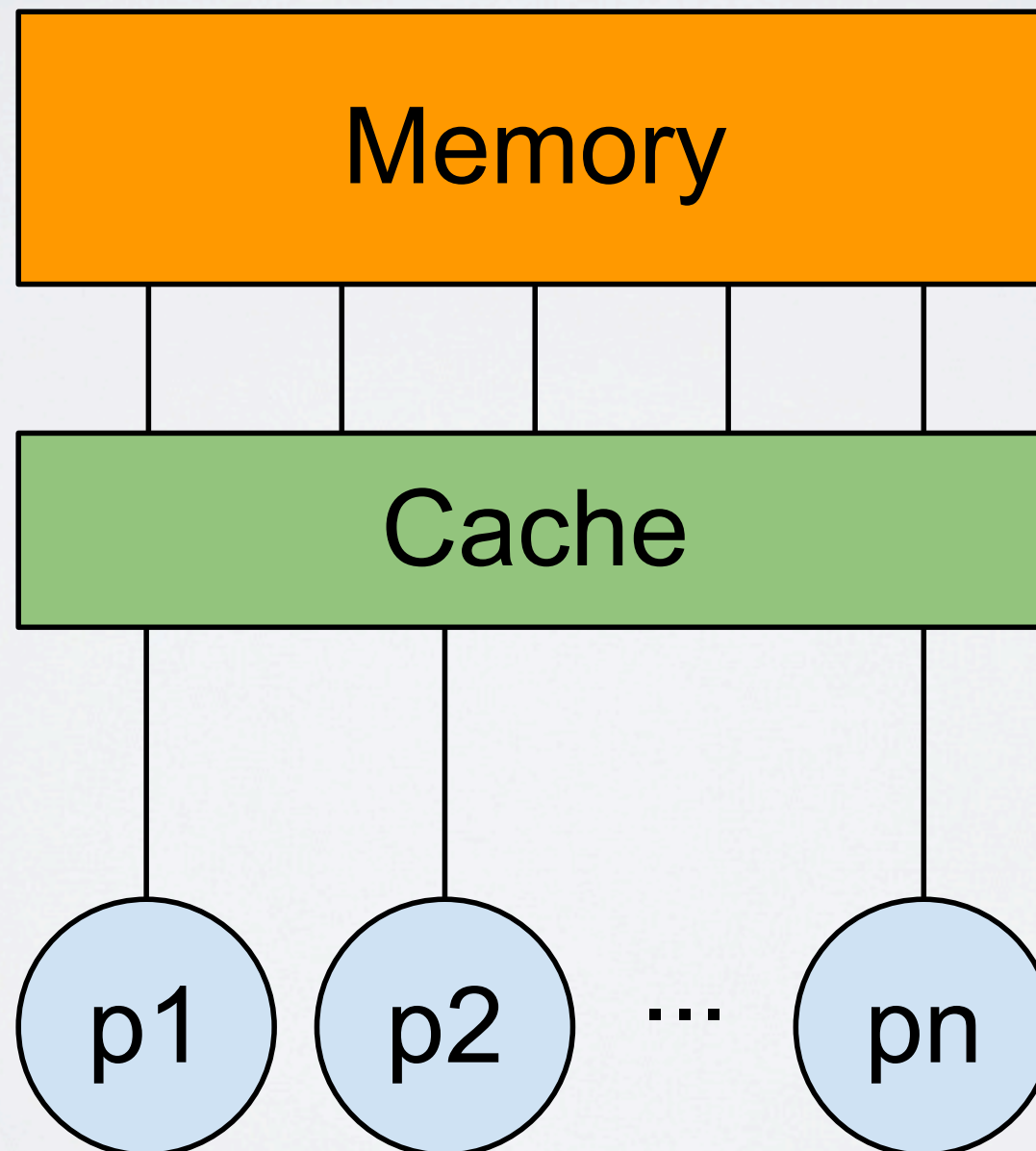
MACHINE MODEL

- Simplest form: PRAM model



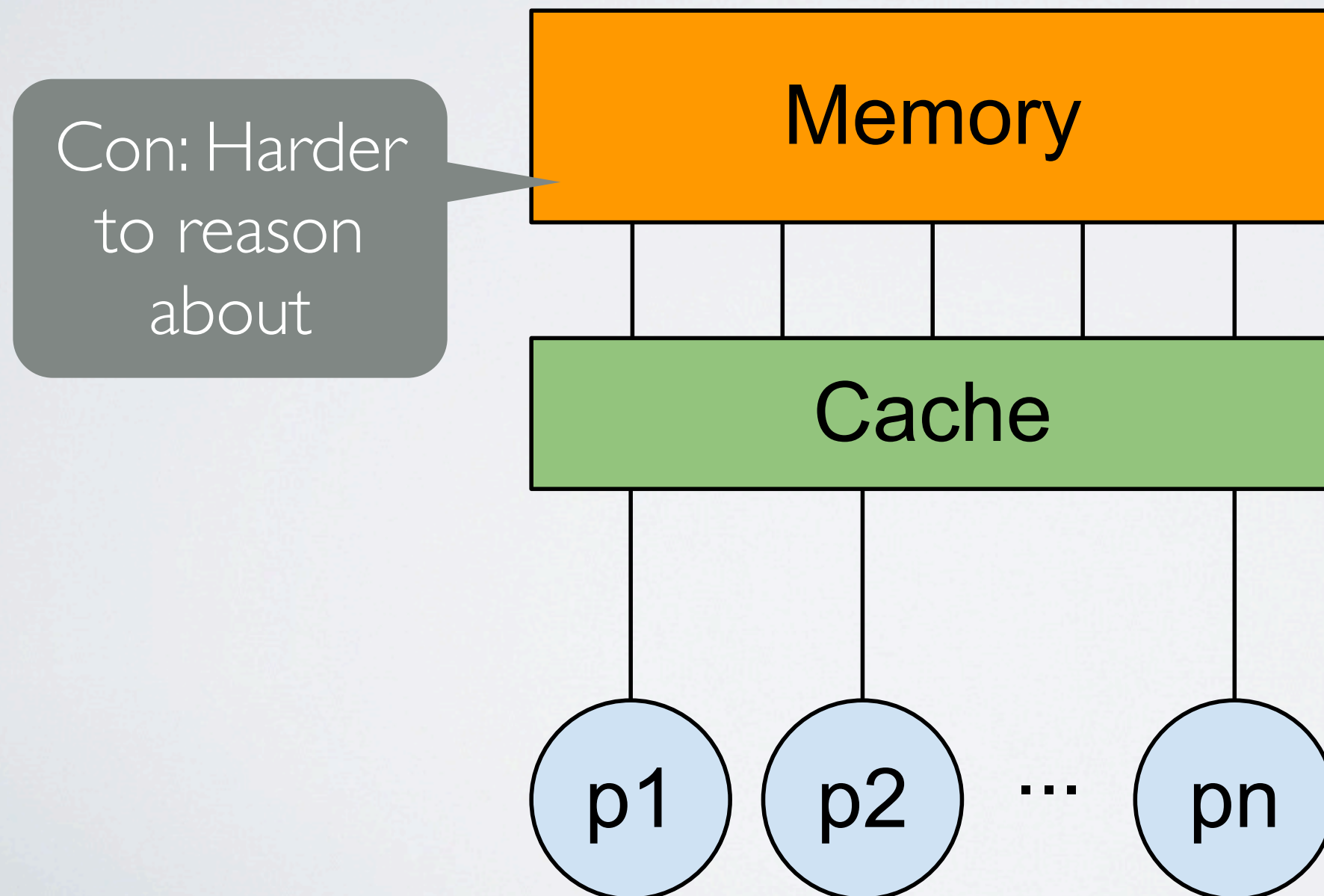
MACHINE MODEL

- Extended model: adding caches



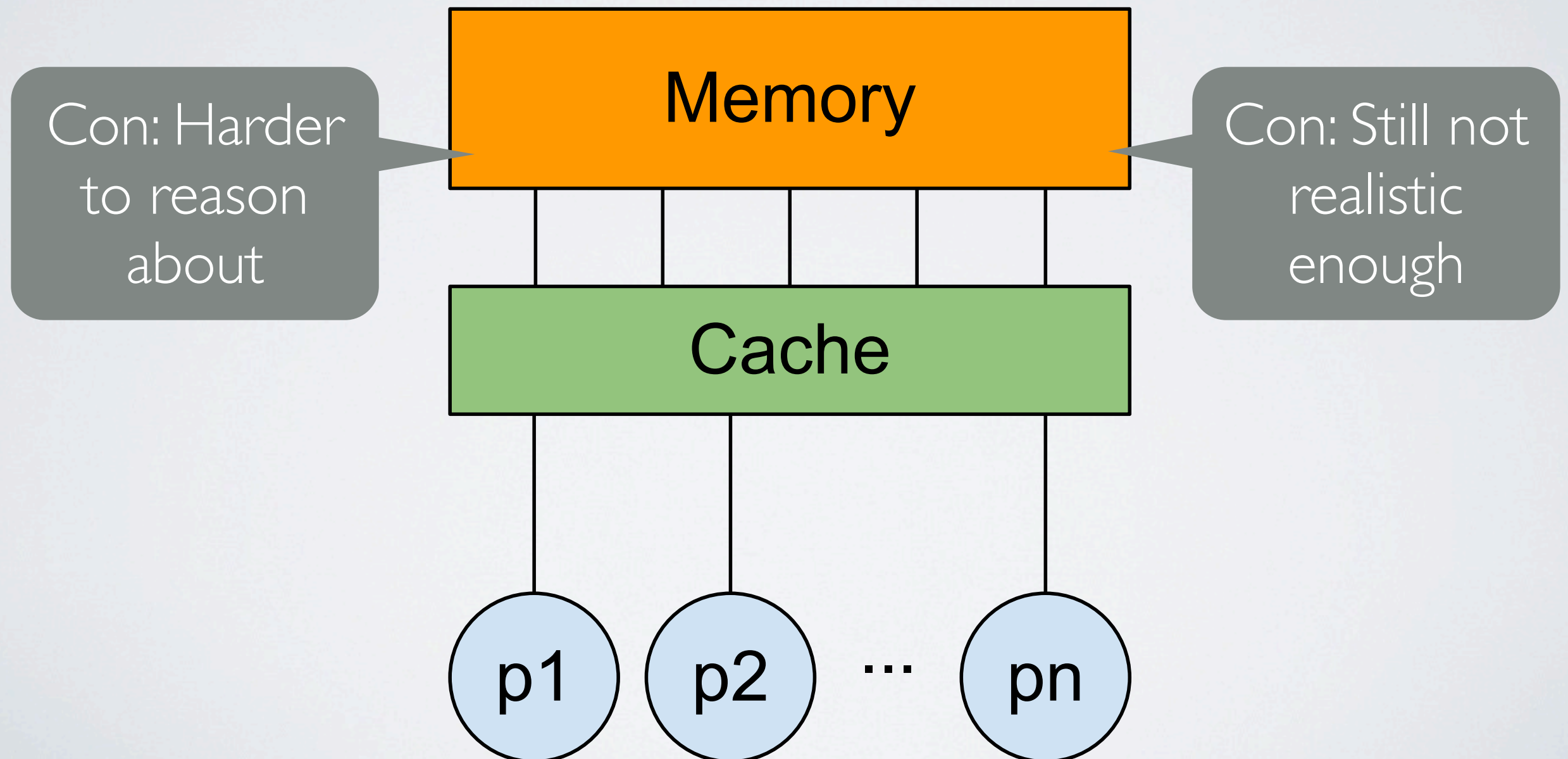
MACHINE MODEL

- Extended model: adding caches



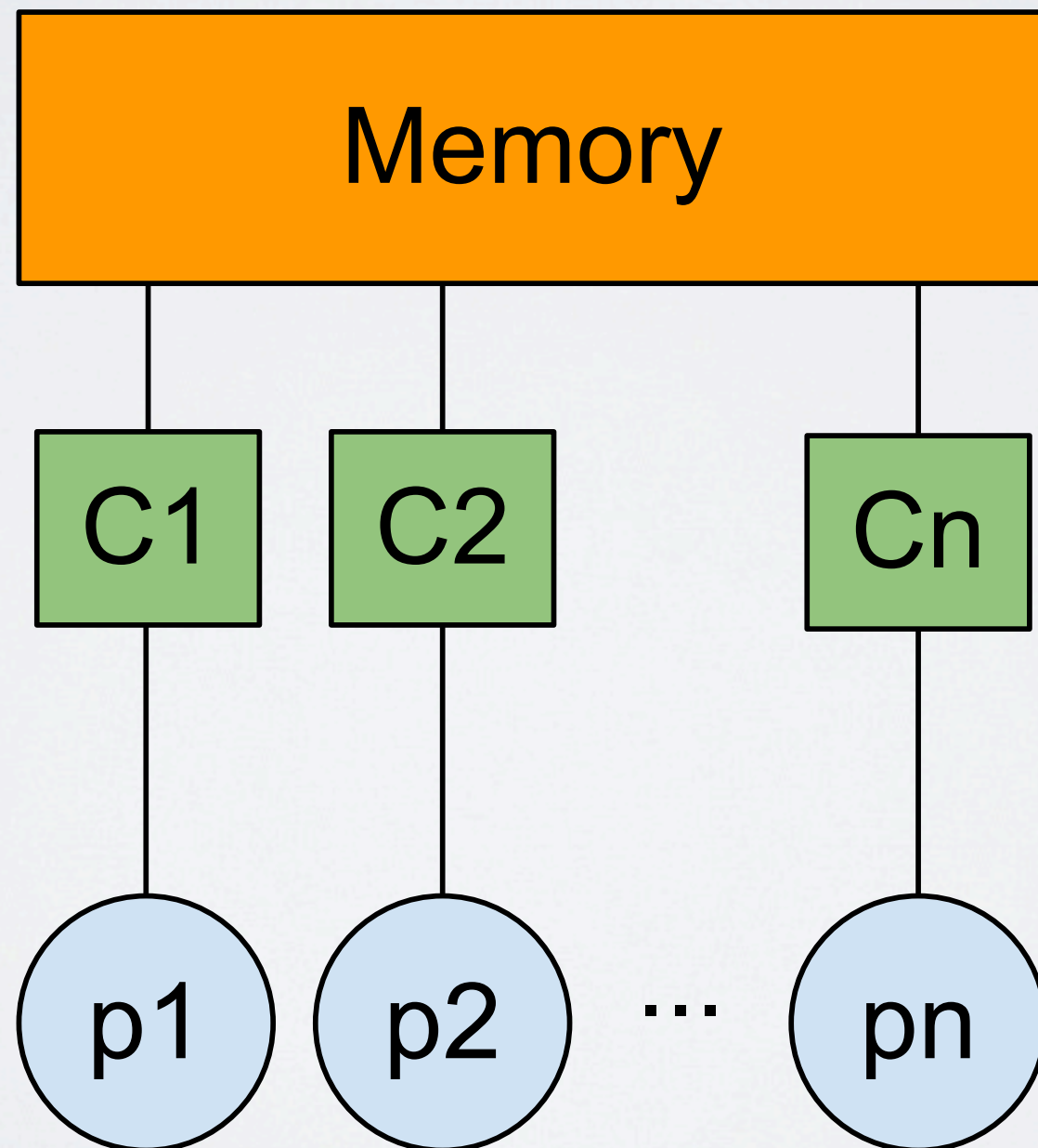
MACHINE MODEL

- Extended model: adding caches



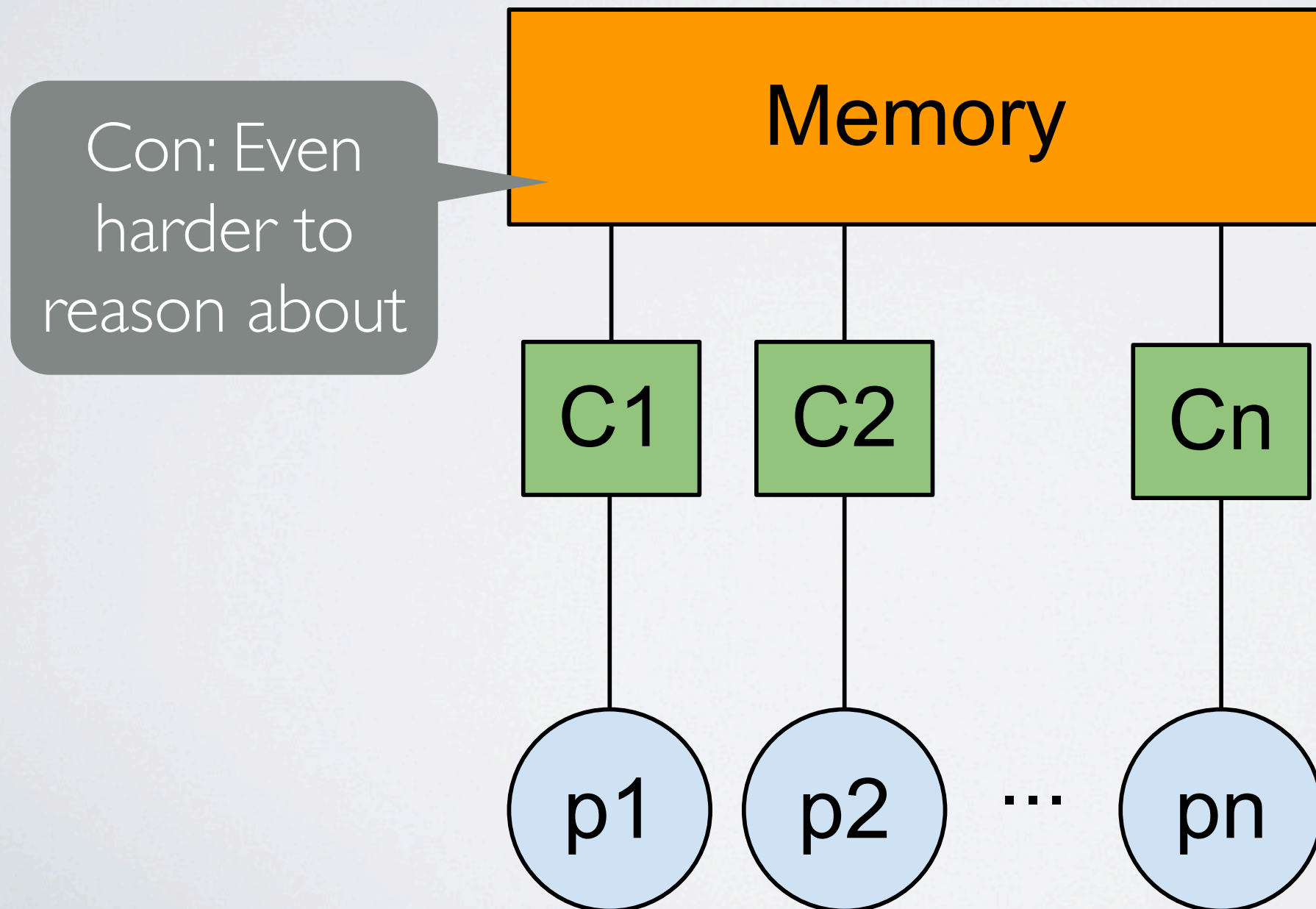
MACHINE MODEL

- Extended model: adding core-private caches



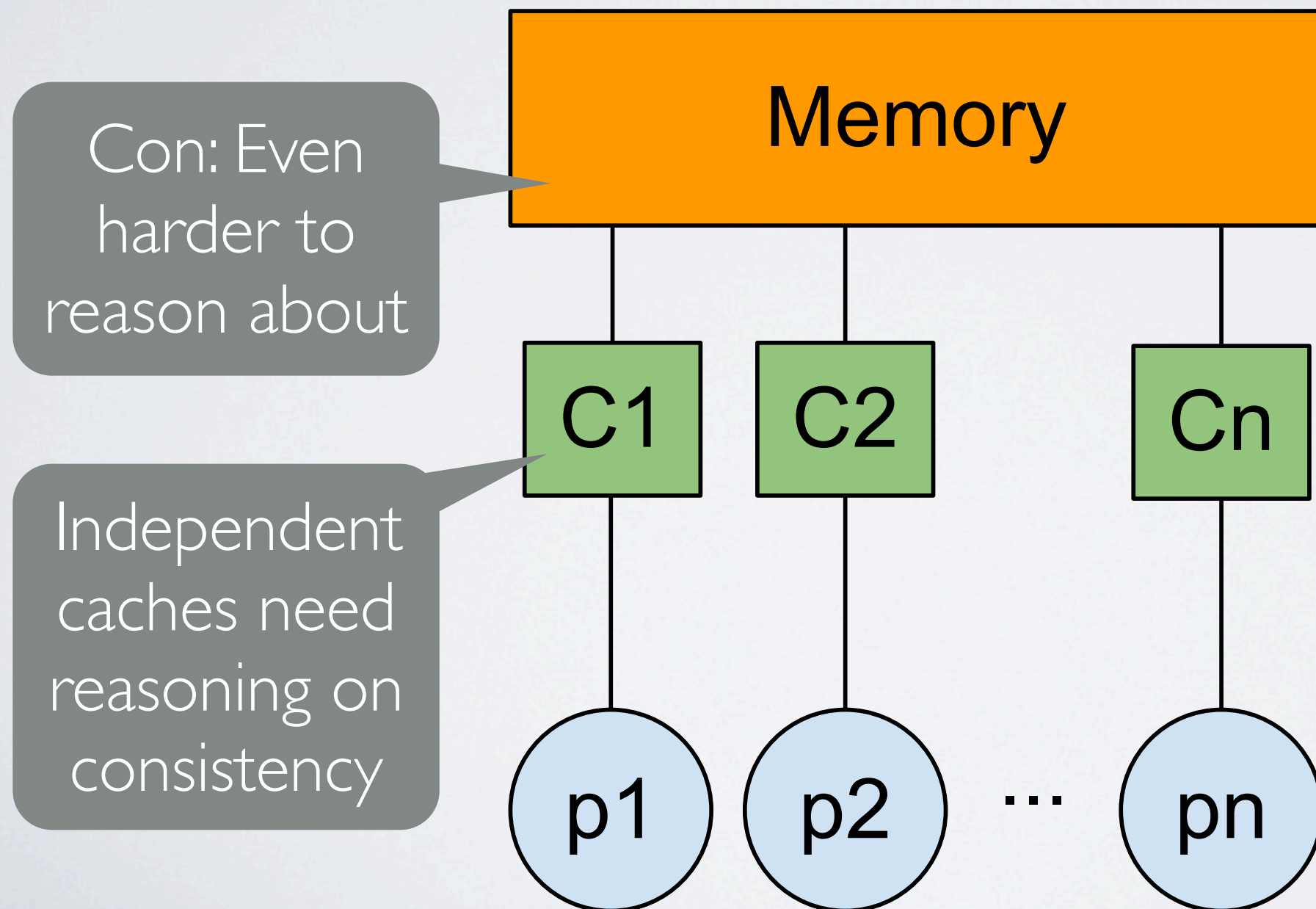
MACHINE MODEL

- Extended model: adding core-private caches



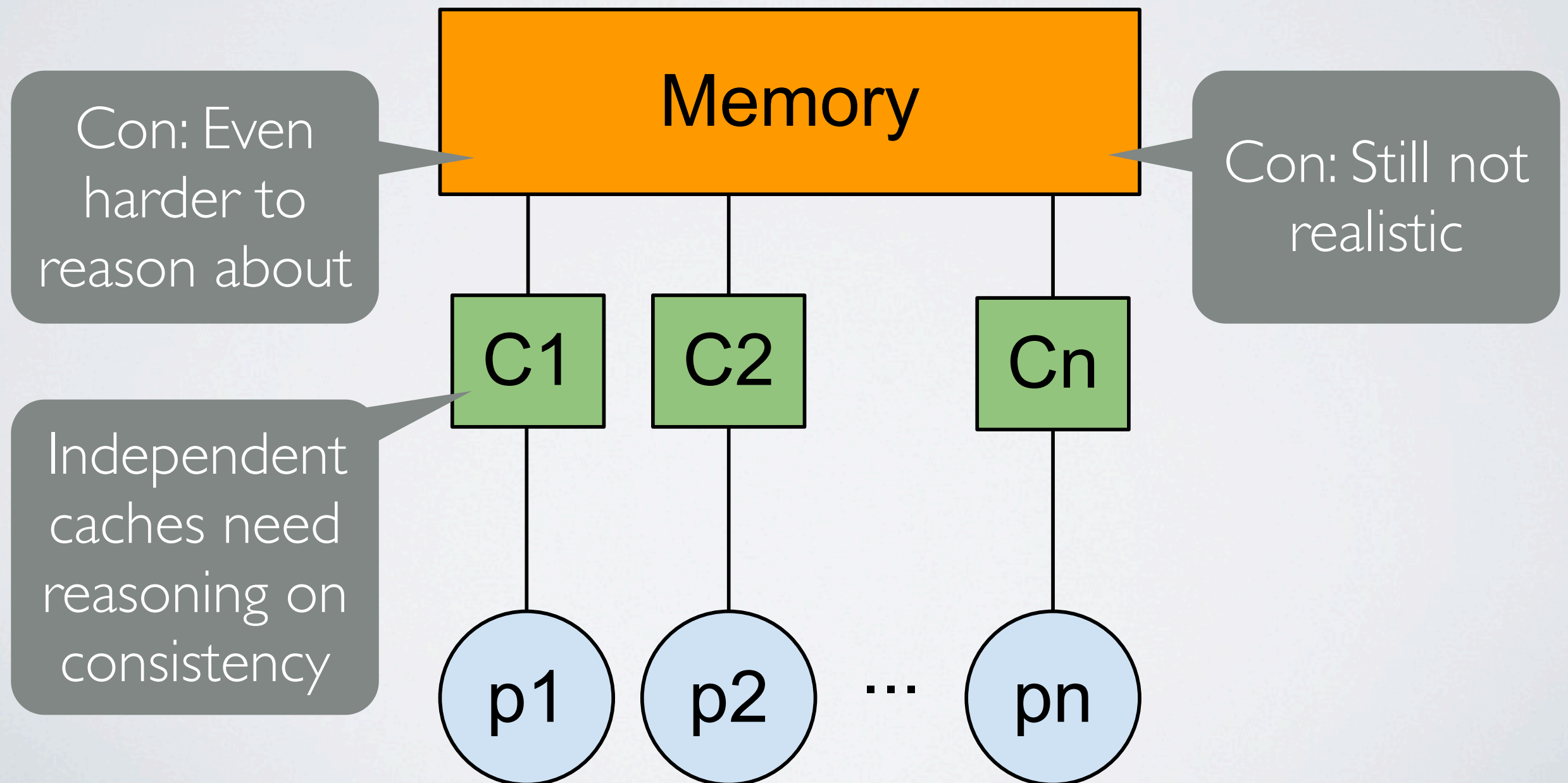
MACHINE MODEL

- Extended model: adding core-private caches



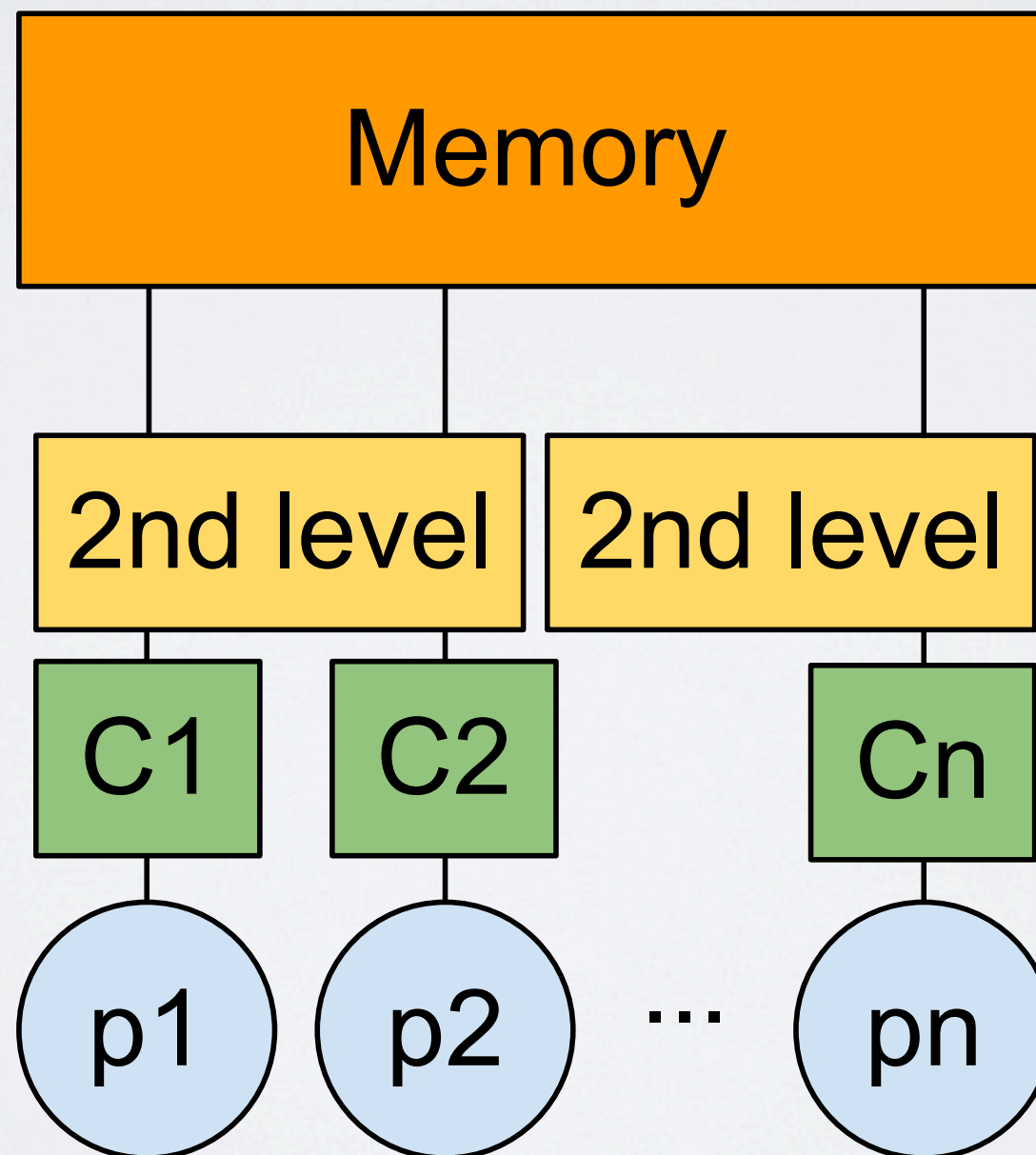
MACHINE MODEL

- Extended model: adding core-private caches



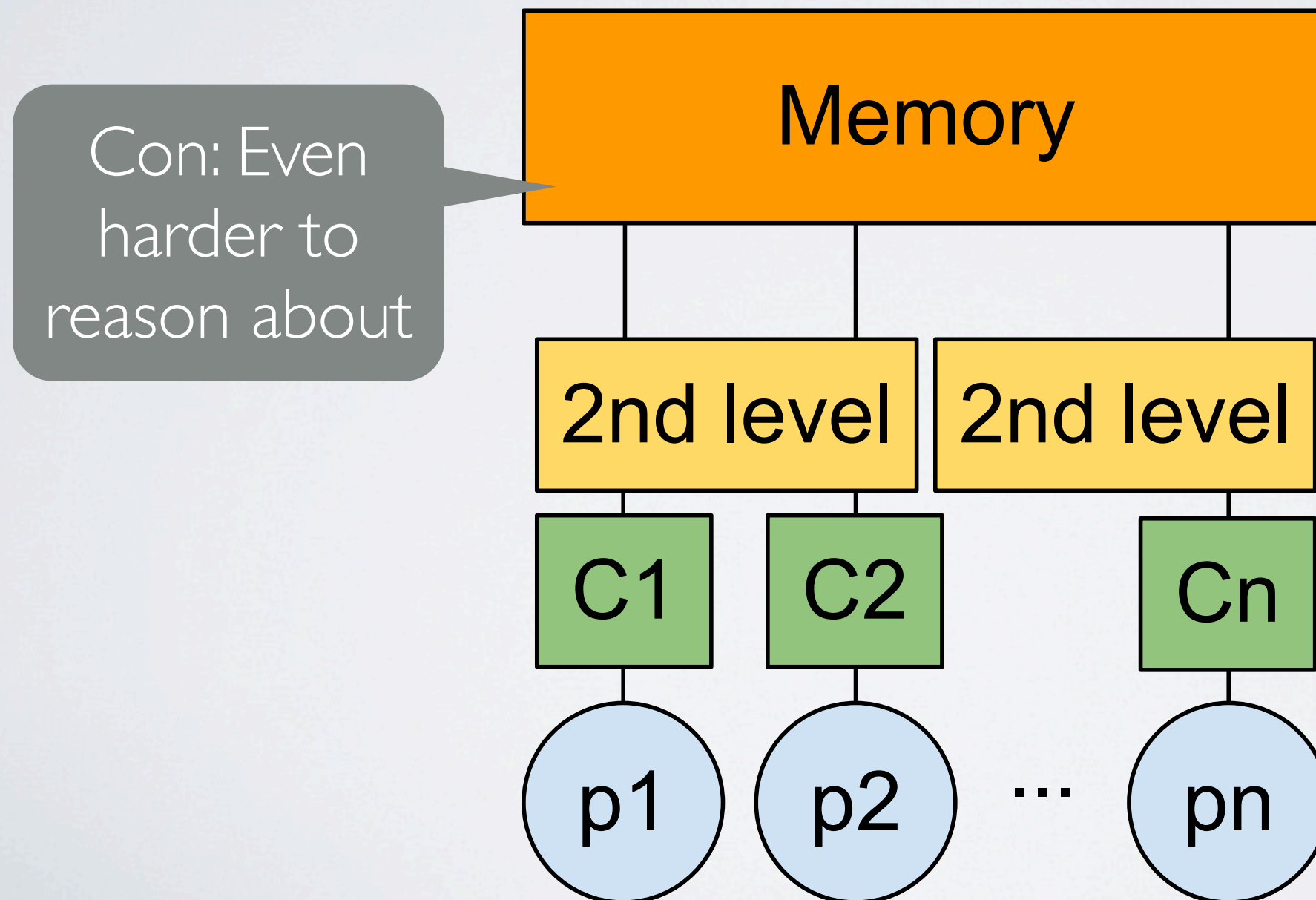
MACHINE MODEL

- Extended model: adding another cache level



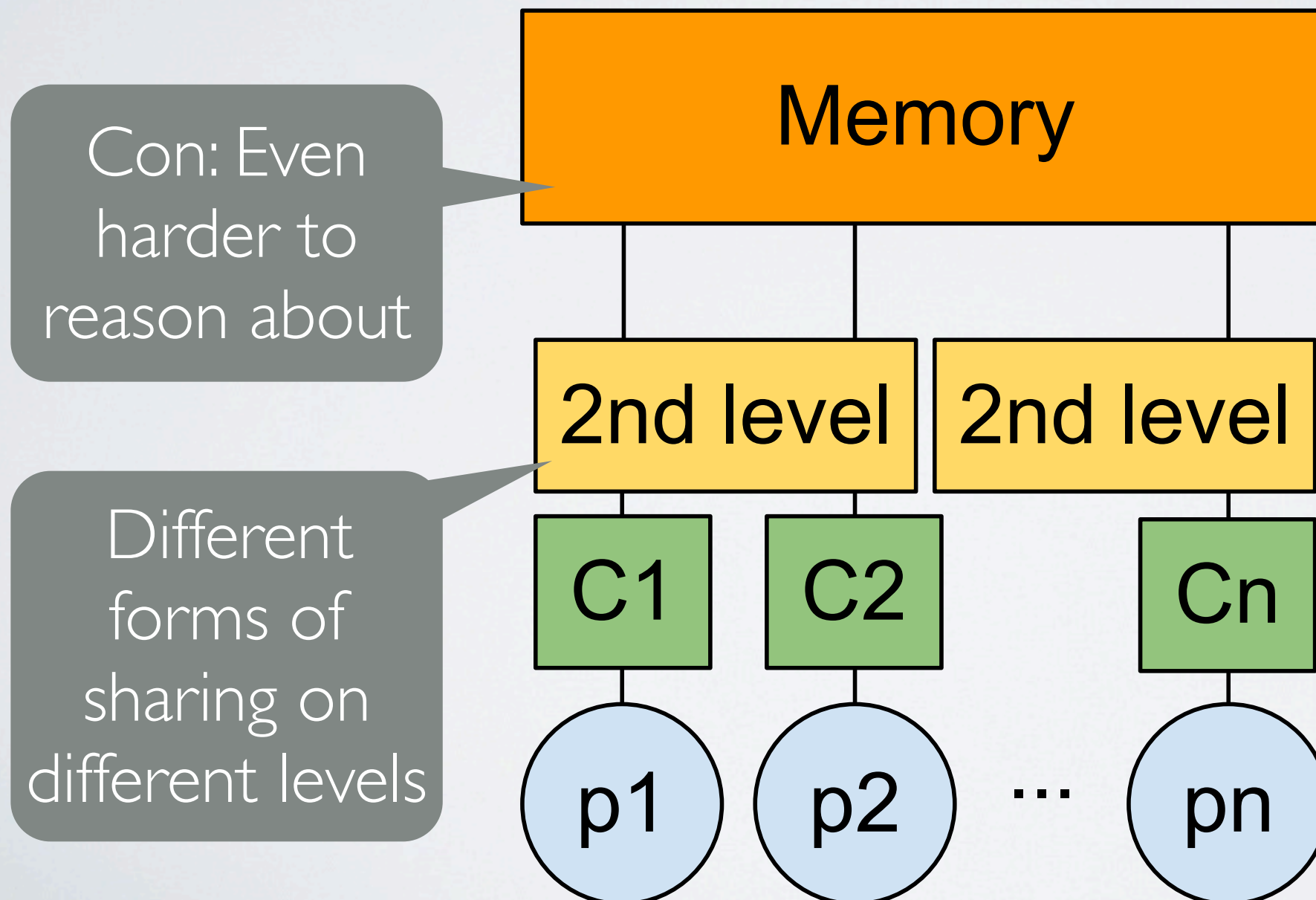
MACHINE MODEL

- Extended model: adding another cache level



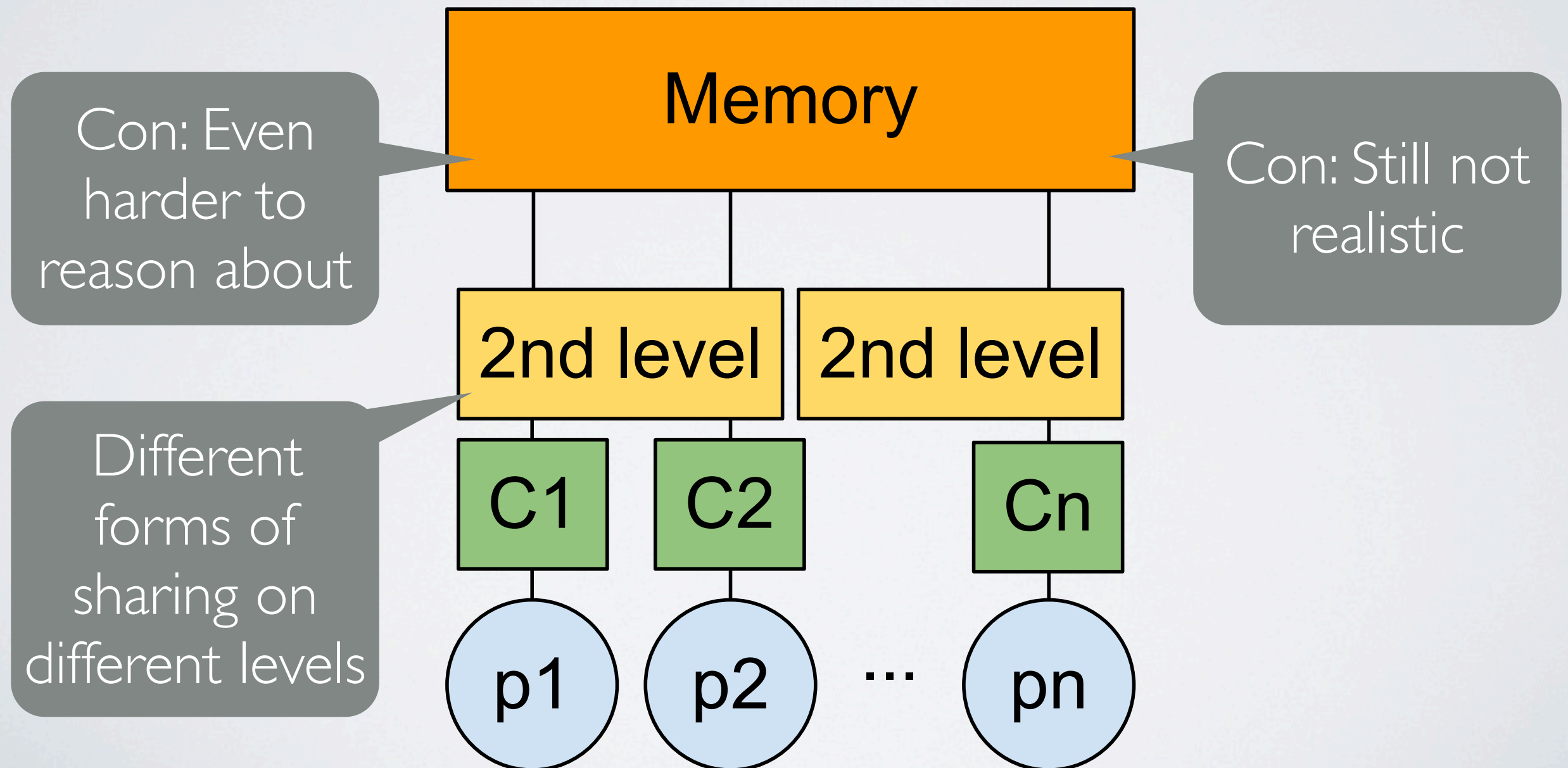
MACHINE MODEL

- Extended model: adding another cache level

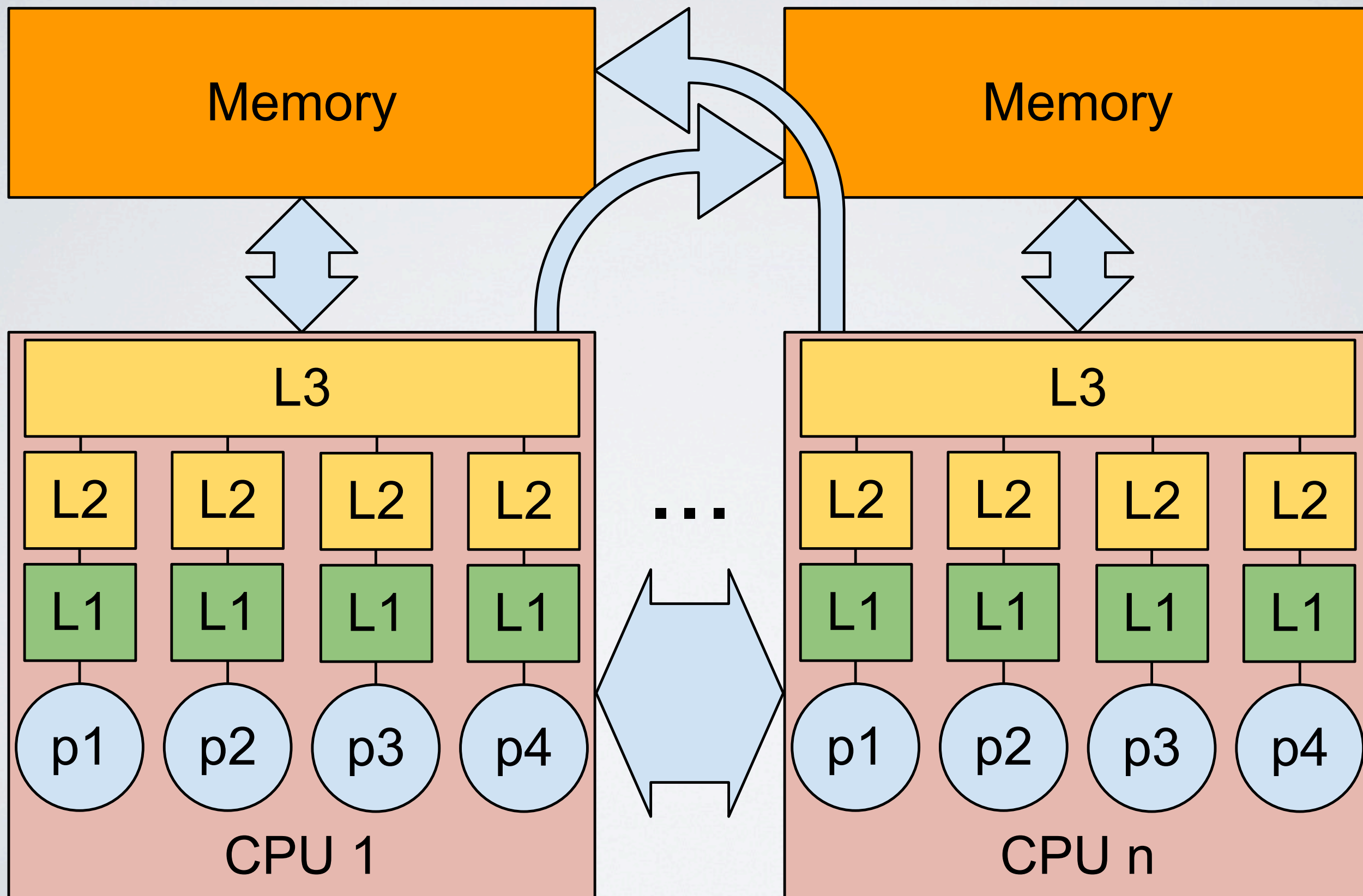


MACHINE MODEL

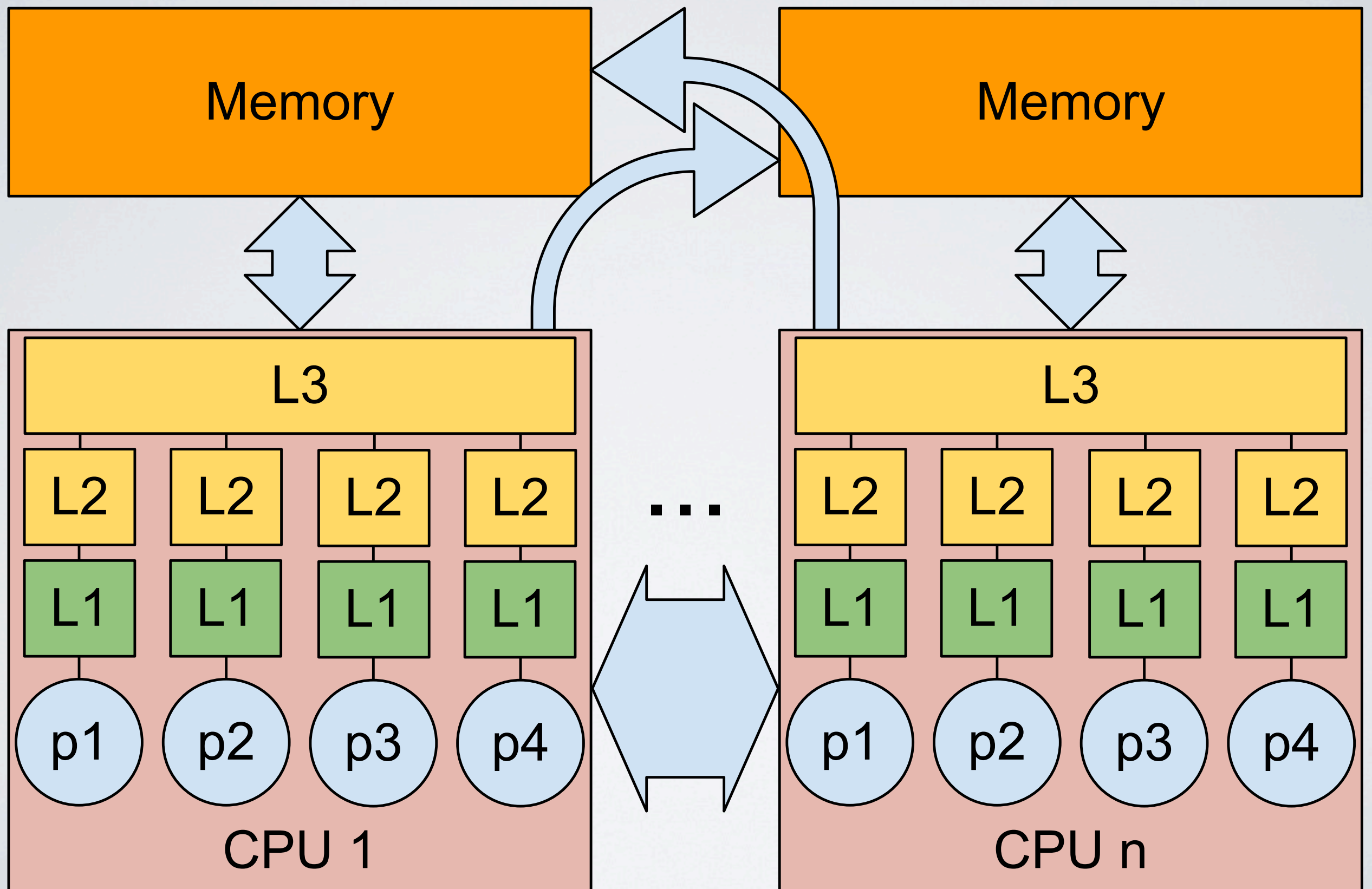
- Extended model: adding another cache level



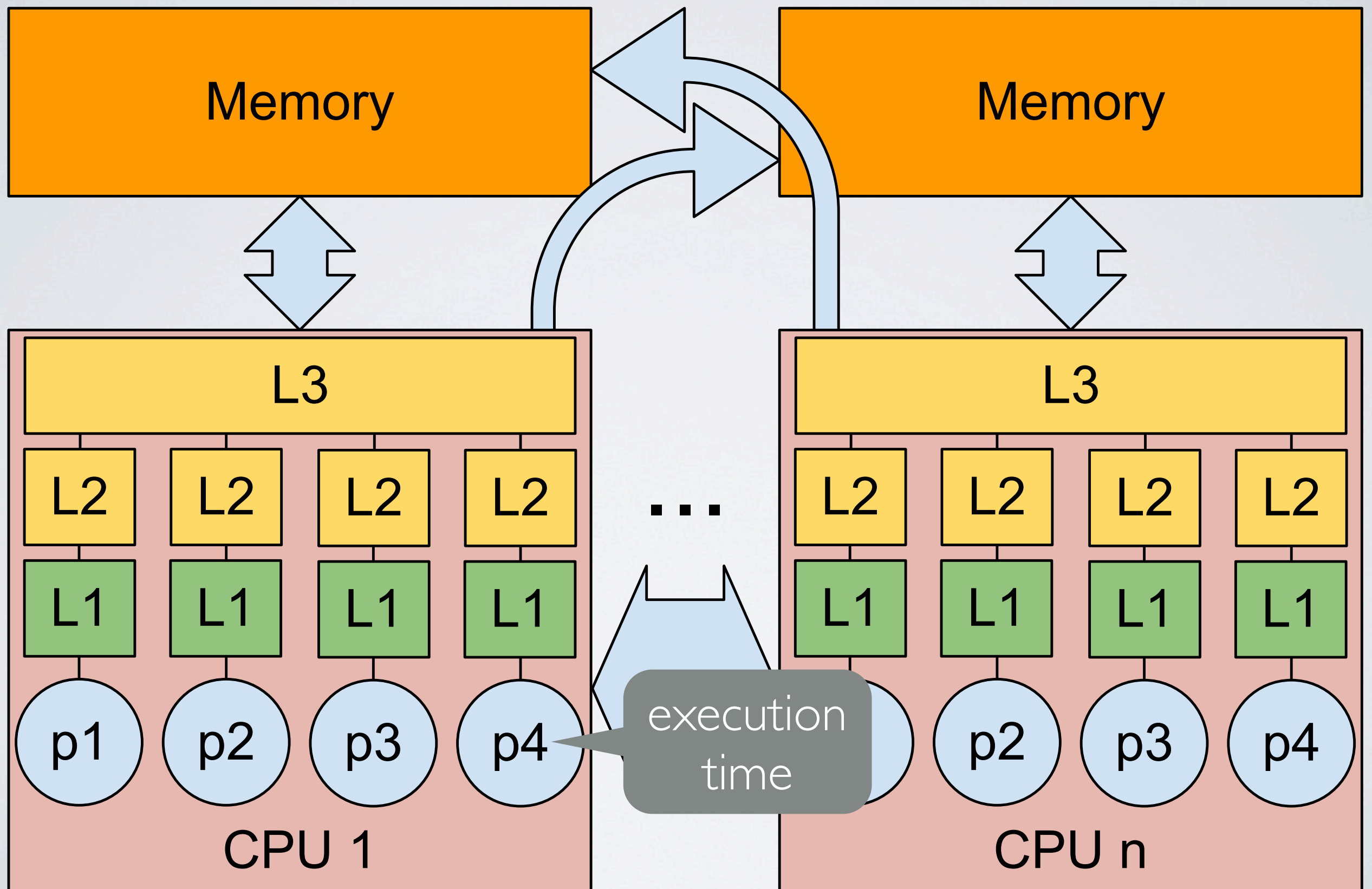
WHAT IS REALISTIC?



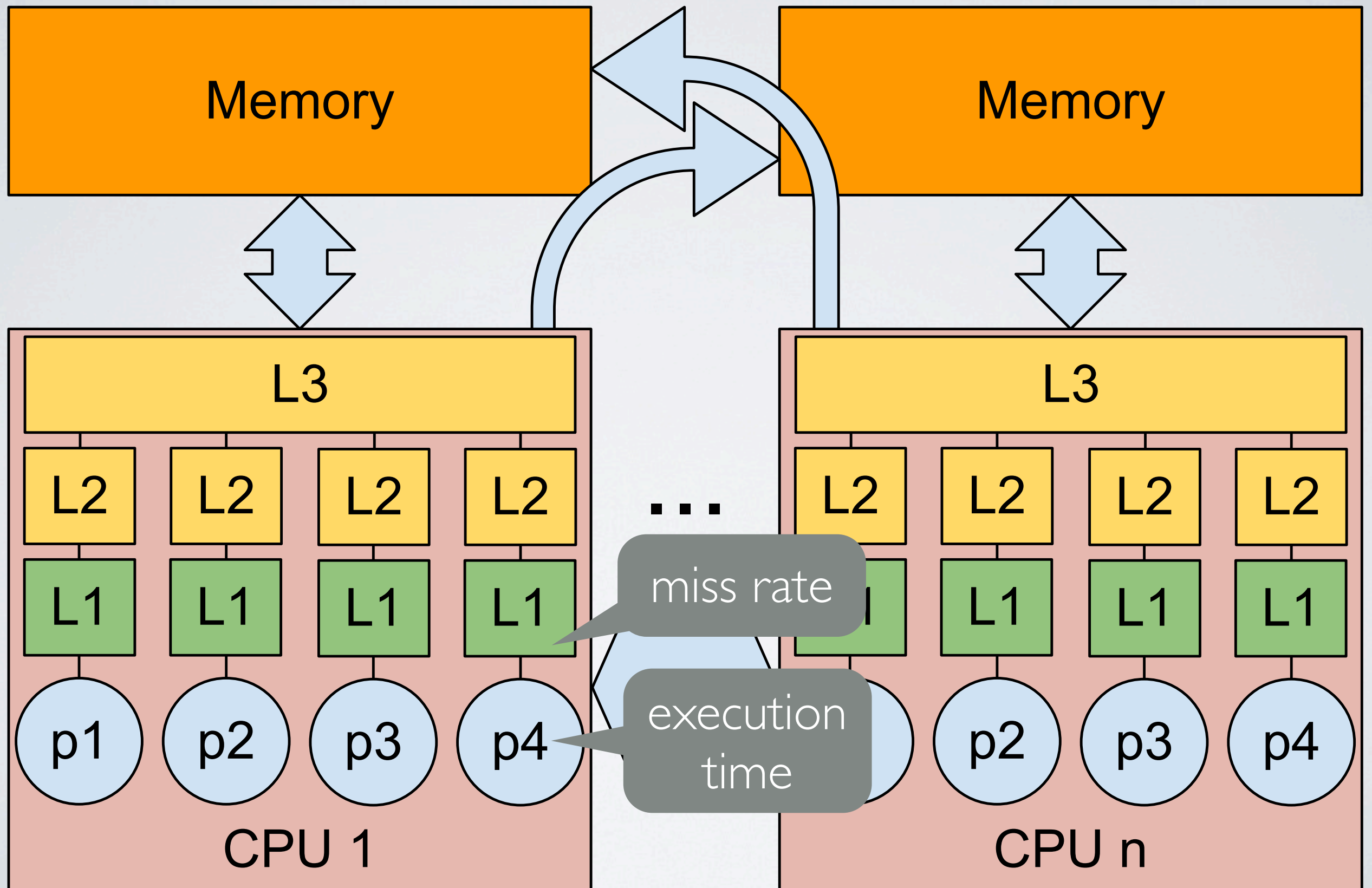
WHAT CAN WE MEASURE?



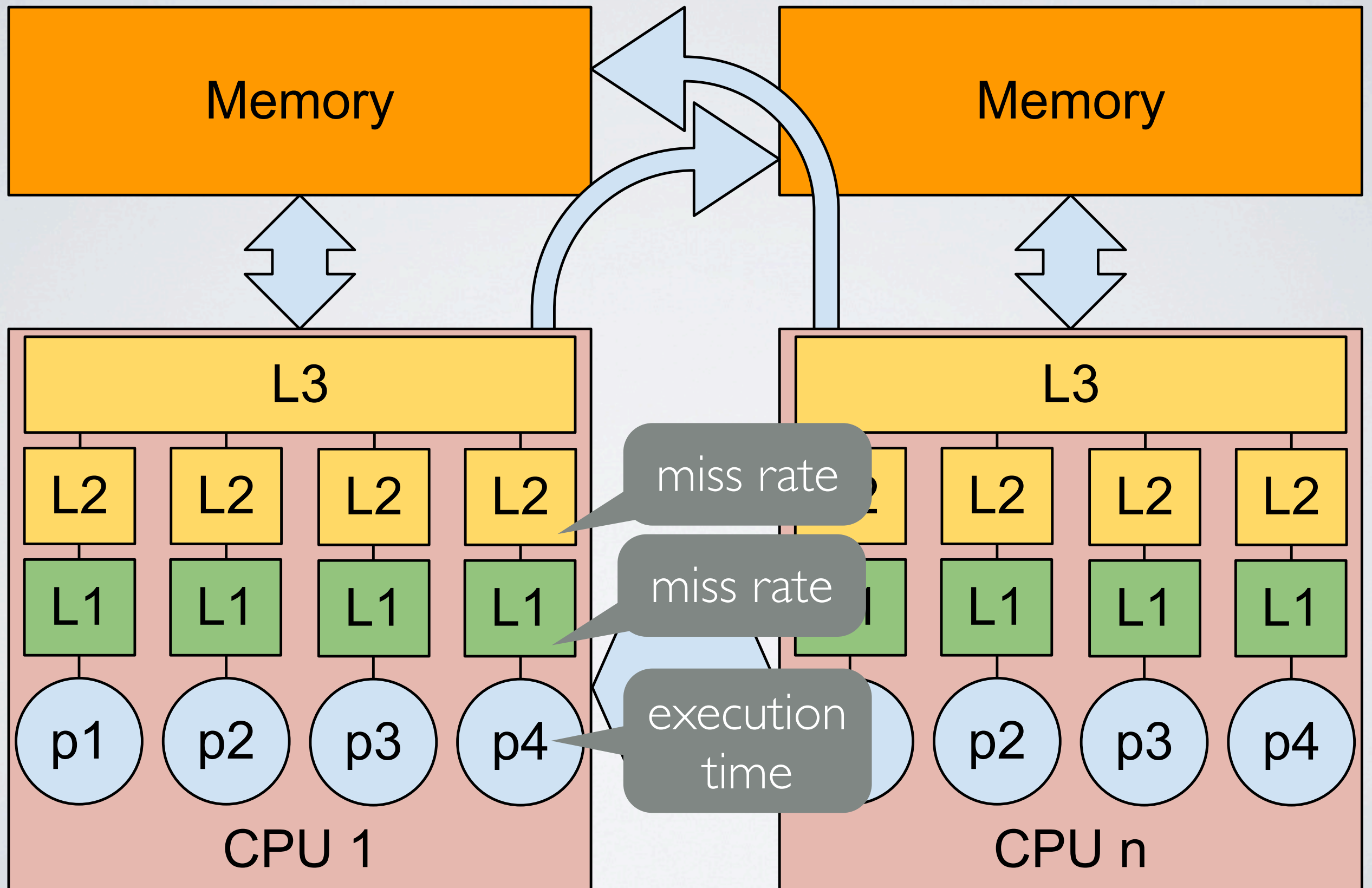
WHAT CAN WE MEASURE?



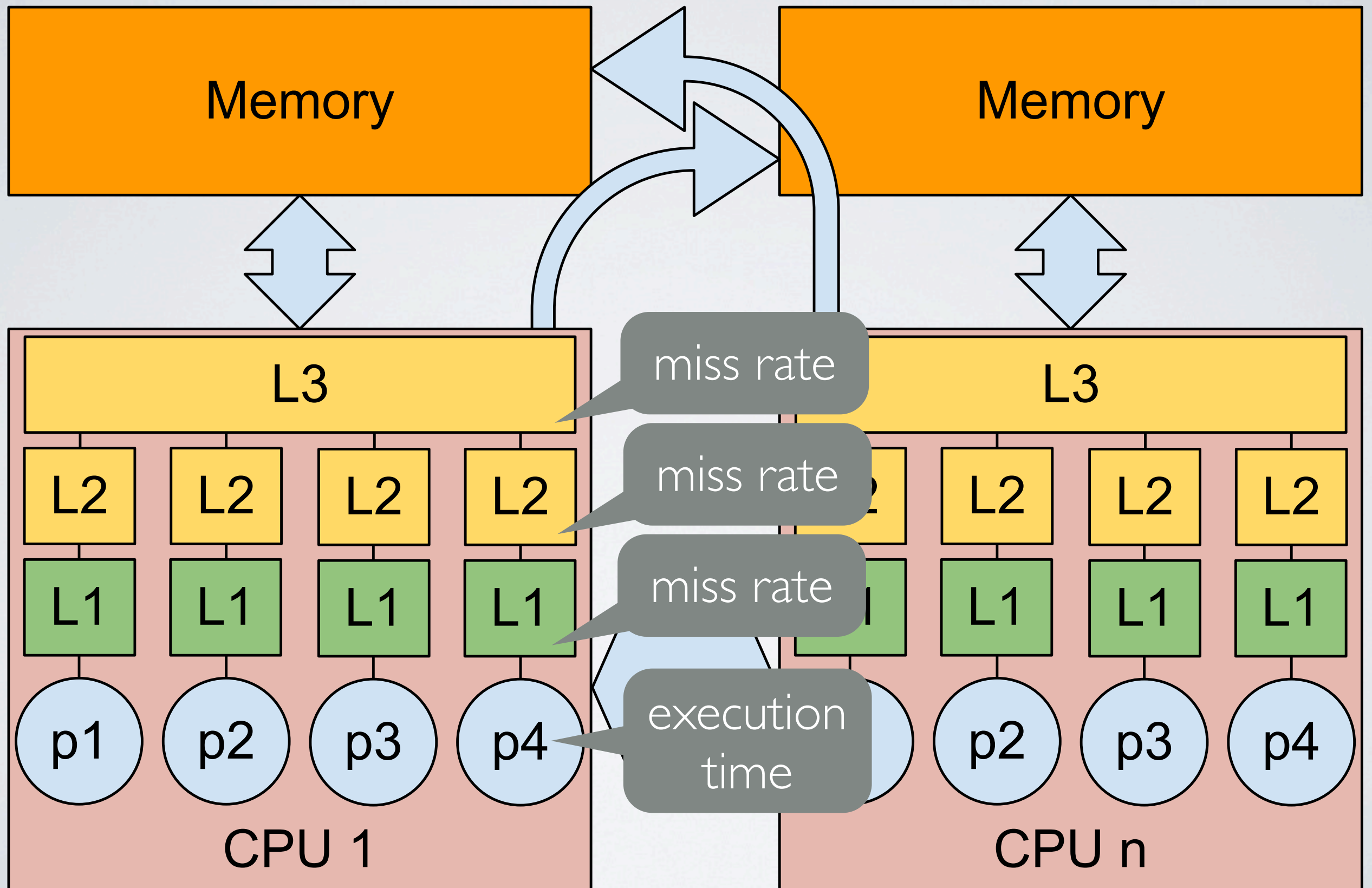
WHAT CAN WE MEASURE?



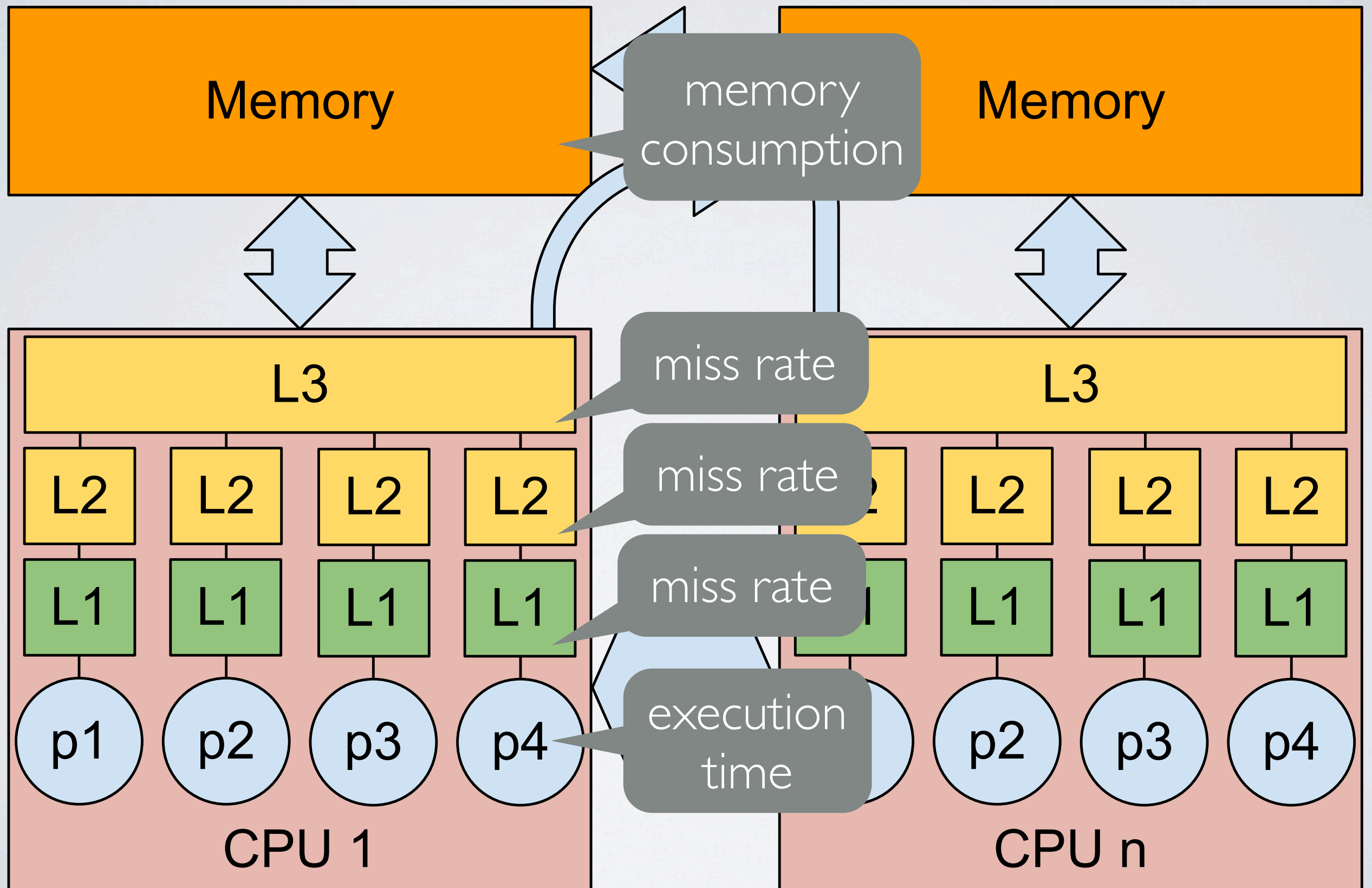
WHAT CAN WE MEASURE?



WHAT CAN WE MEASURE?



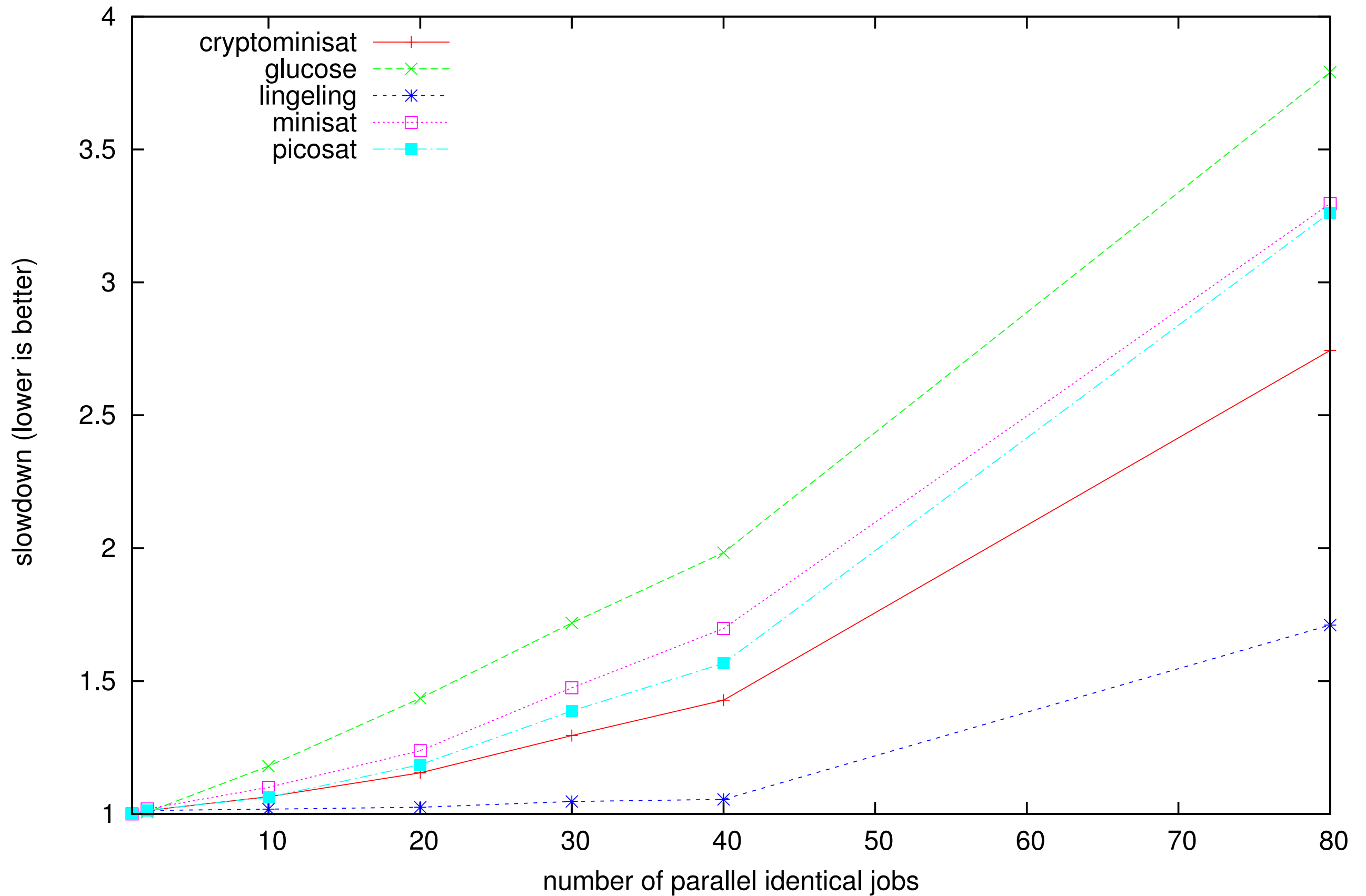
WHAT CAN WE MEASURE?



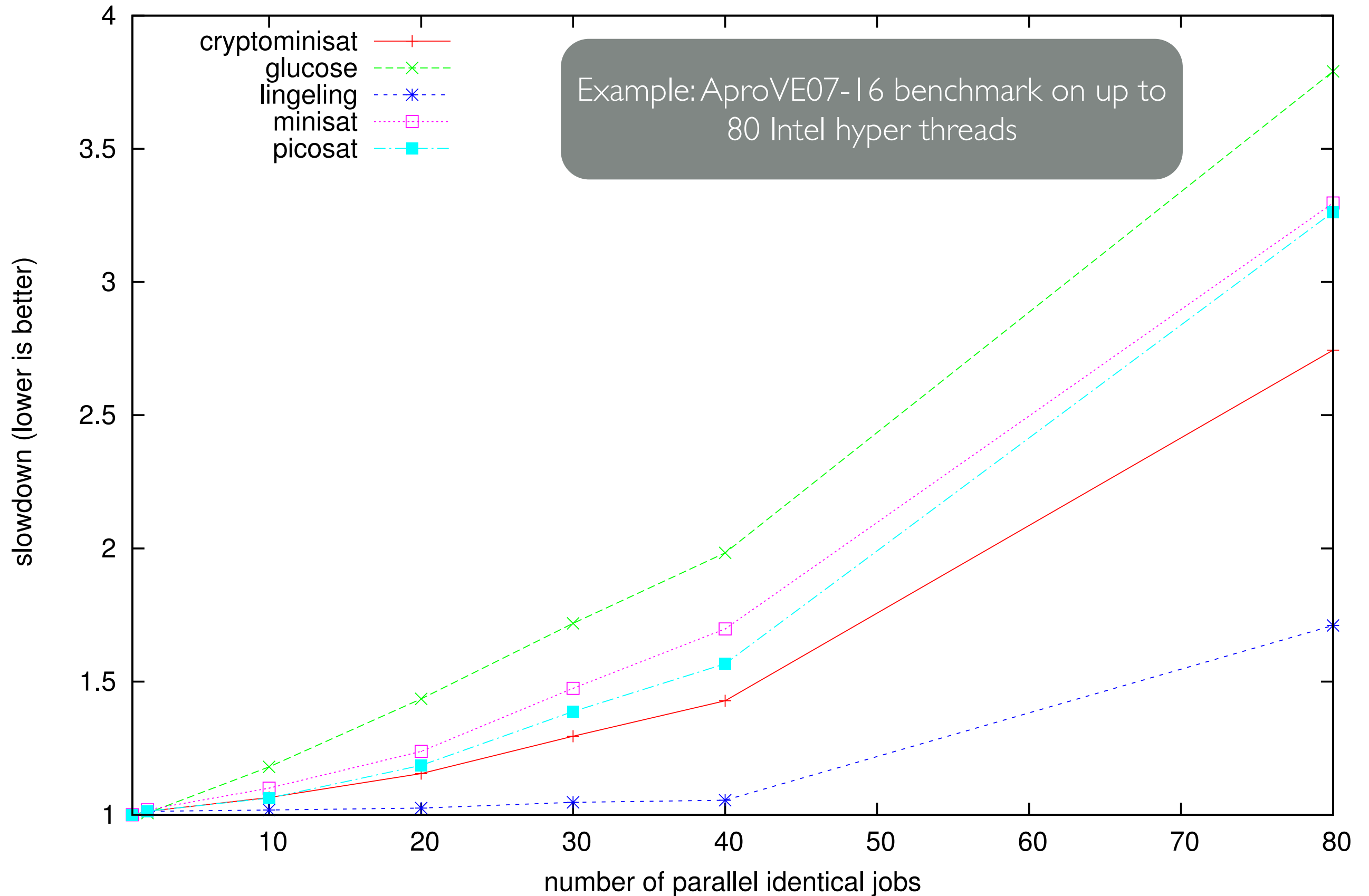
EXPERIMENTAL SETUP

- Five recent SAT solvers
 - Lingeling, MiniSAT, CryptoMiniSAT, PicoSAT, Glucose
- Ten benchmarks: “solvable in reasonable time”
 - listed in the paper
- Five shared-memory multi-core systems (Intel and AMD)
 - ranging from 8 to 80 cores

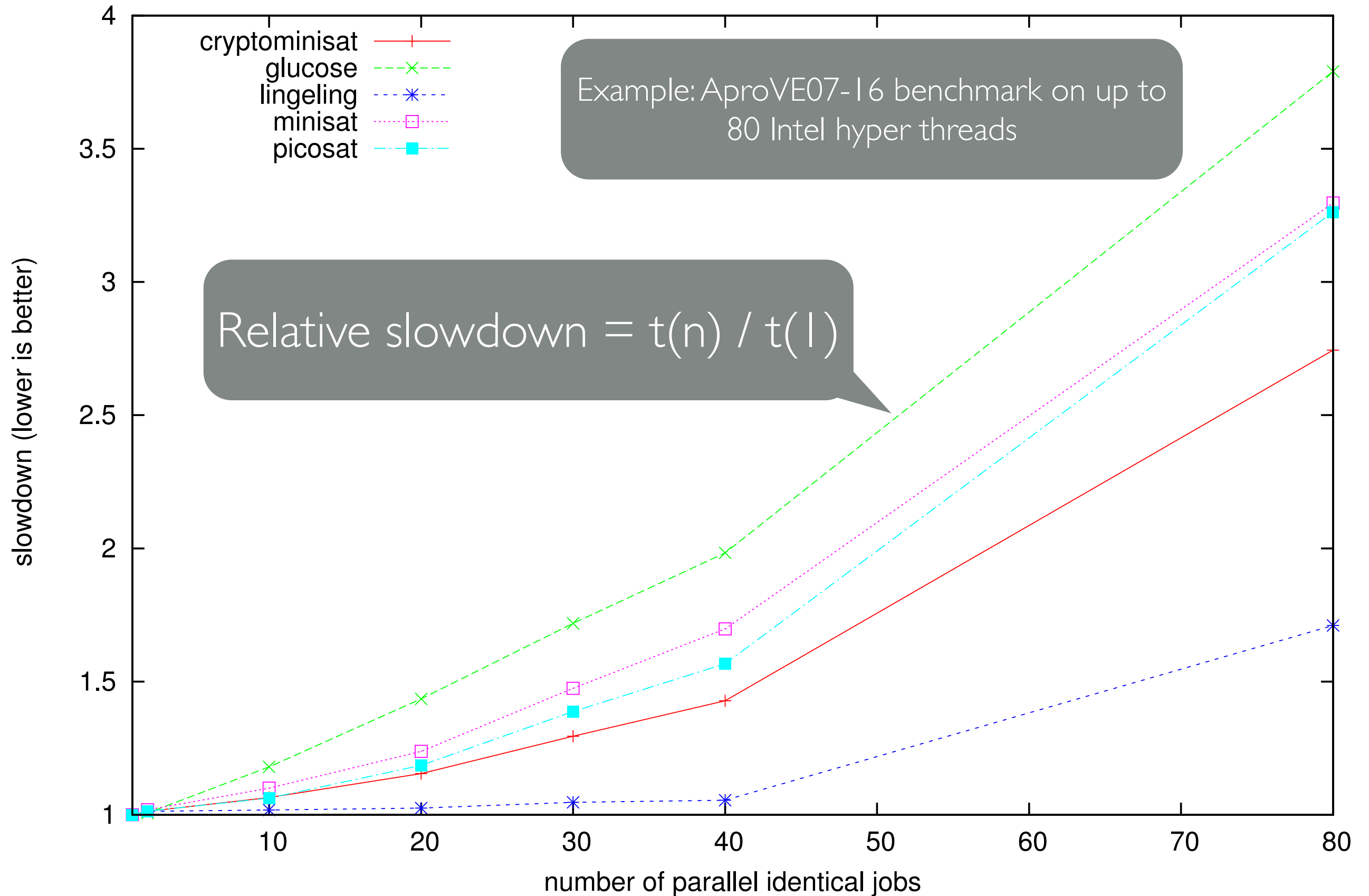
EXPERIMENT: SLOWDOWN



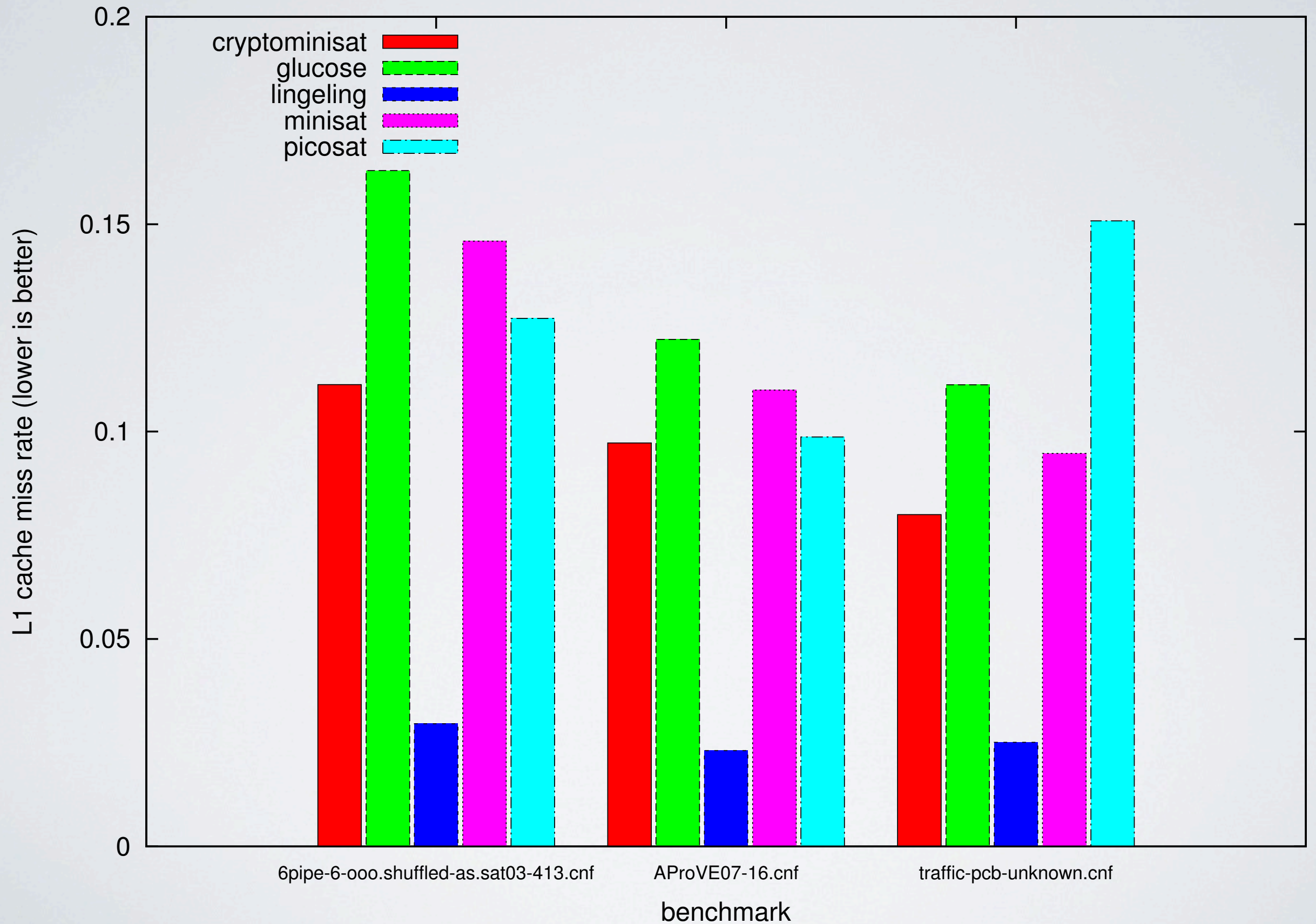
EXPERIMENT: SLOWDOWN



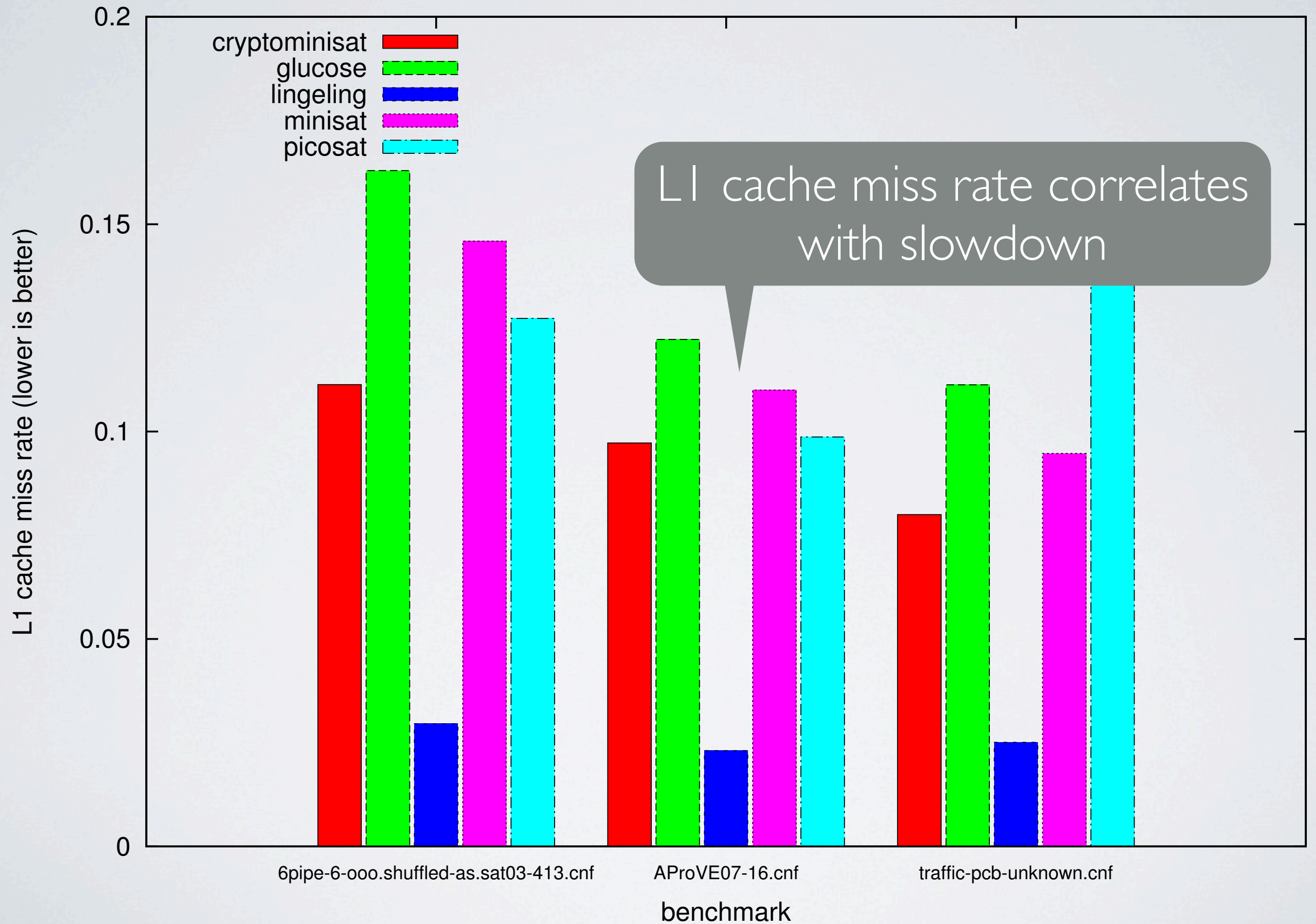
EXPERIMENT: SLOWDOWN



EXPERIMENT: SLOWDOWN



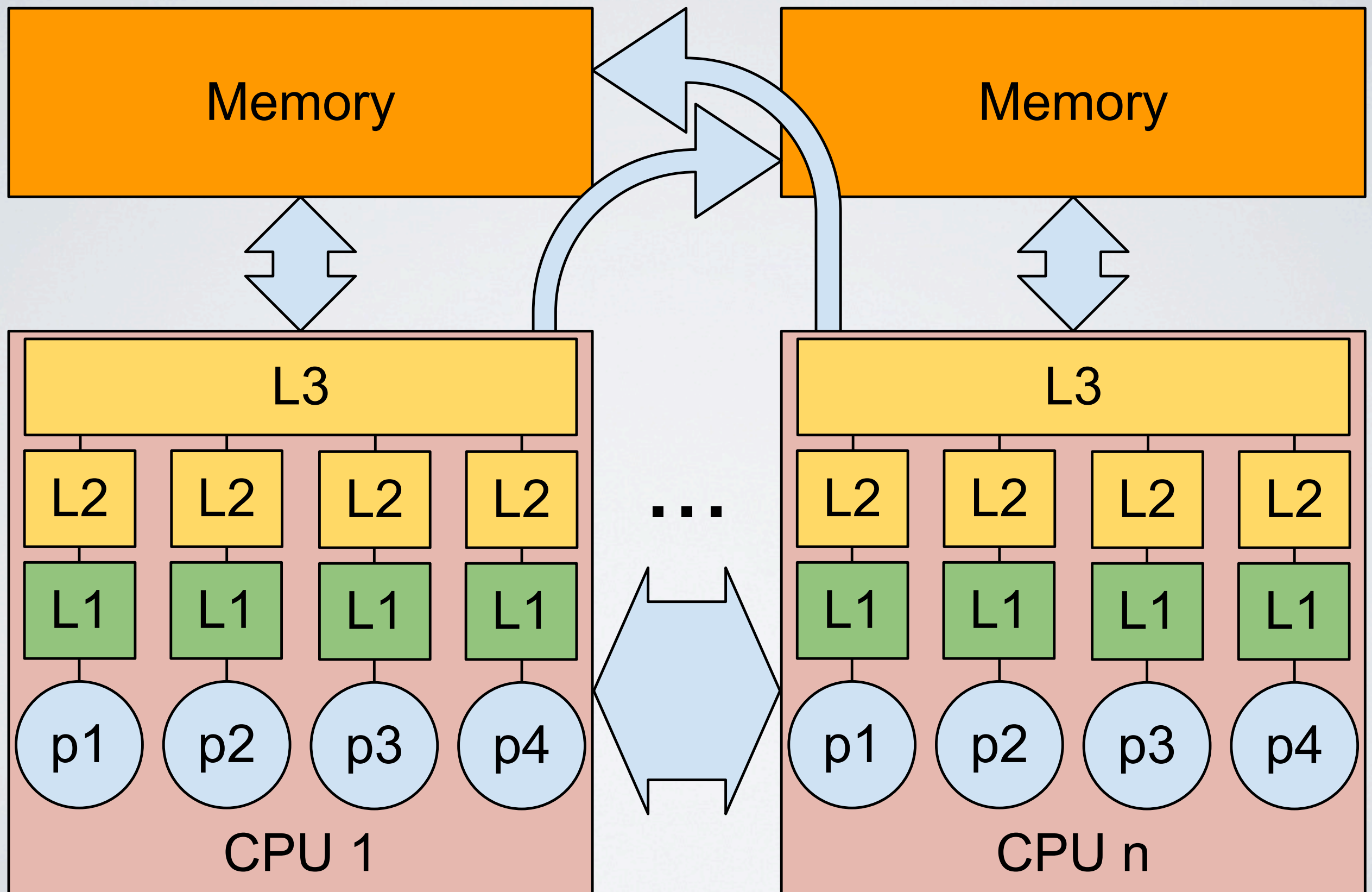
EXPERIMENT: SLOWDOWN



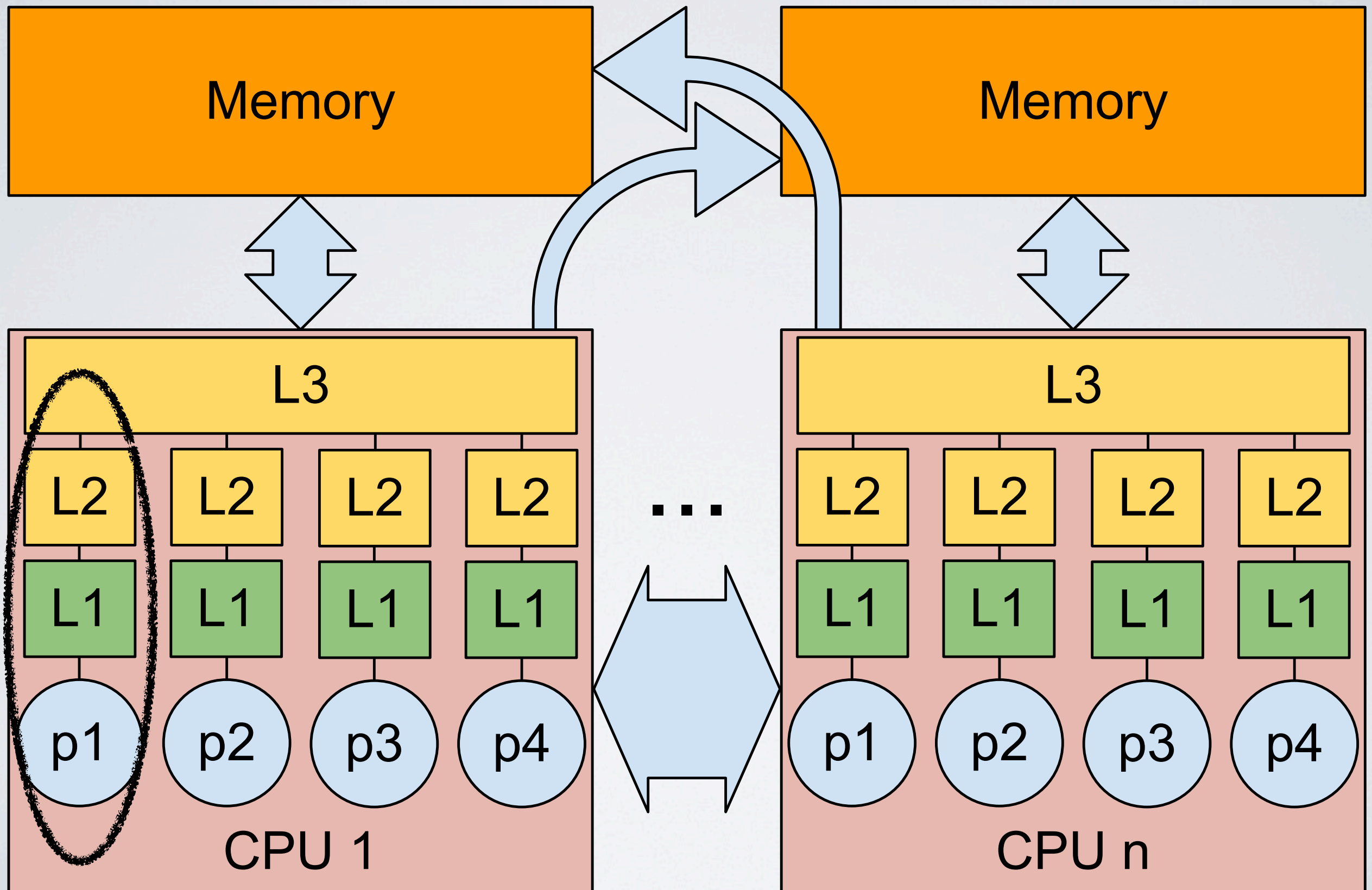
WHAT DOES THAT MEAN?

- Low cache miss rate suggests a small *working set* (WS)
 - the working set of a job is the memory that a job reads or writes during a given time interval
 - measuring the working set is tricky (choosing proper intervals, sampling is either expensive or coarse grained)
 - cache misses can be used as an indicator

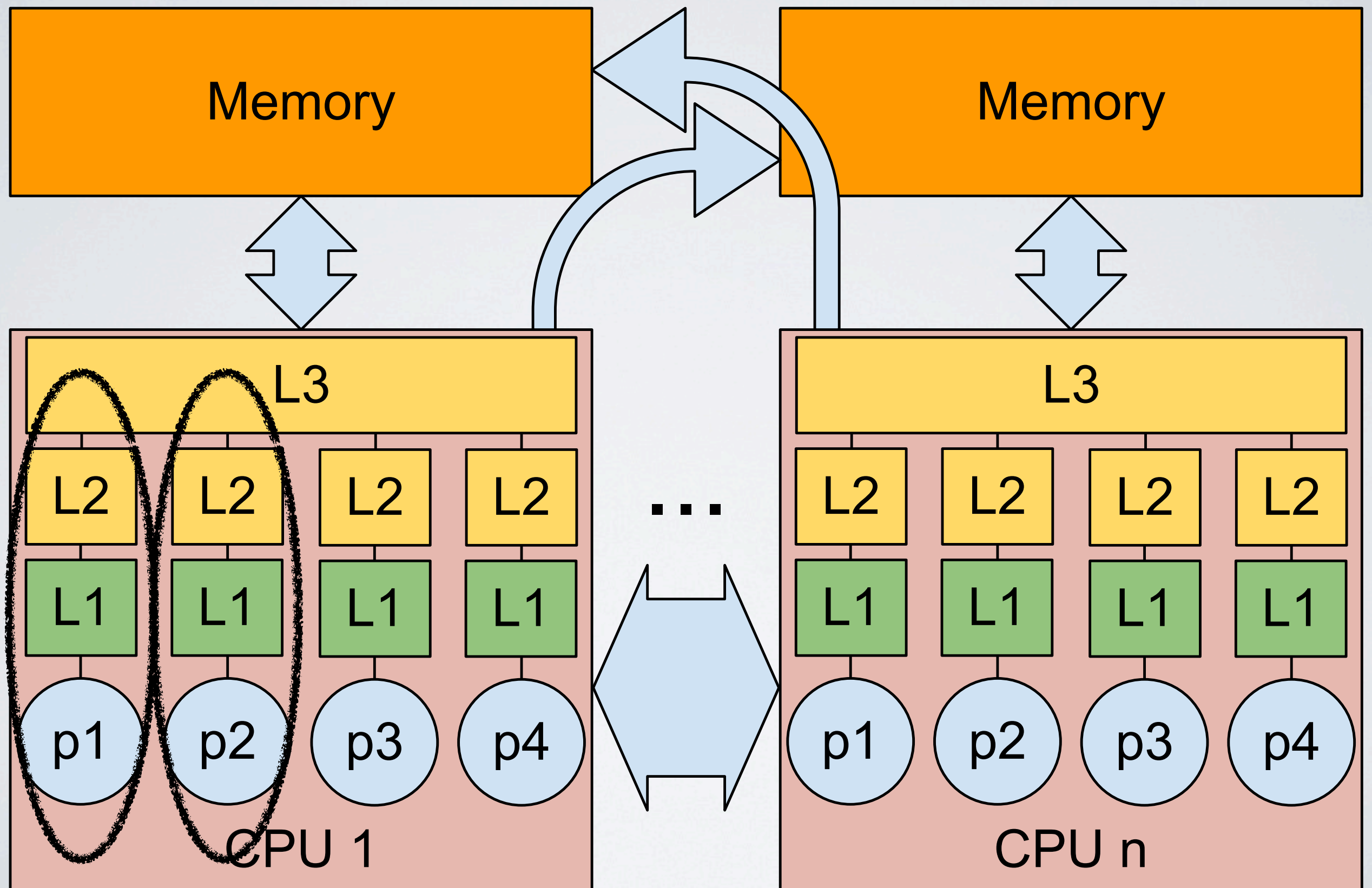
ADVANTAGE OF A SMALL WS



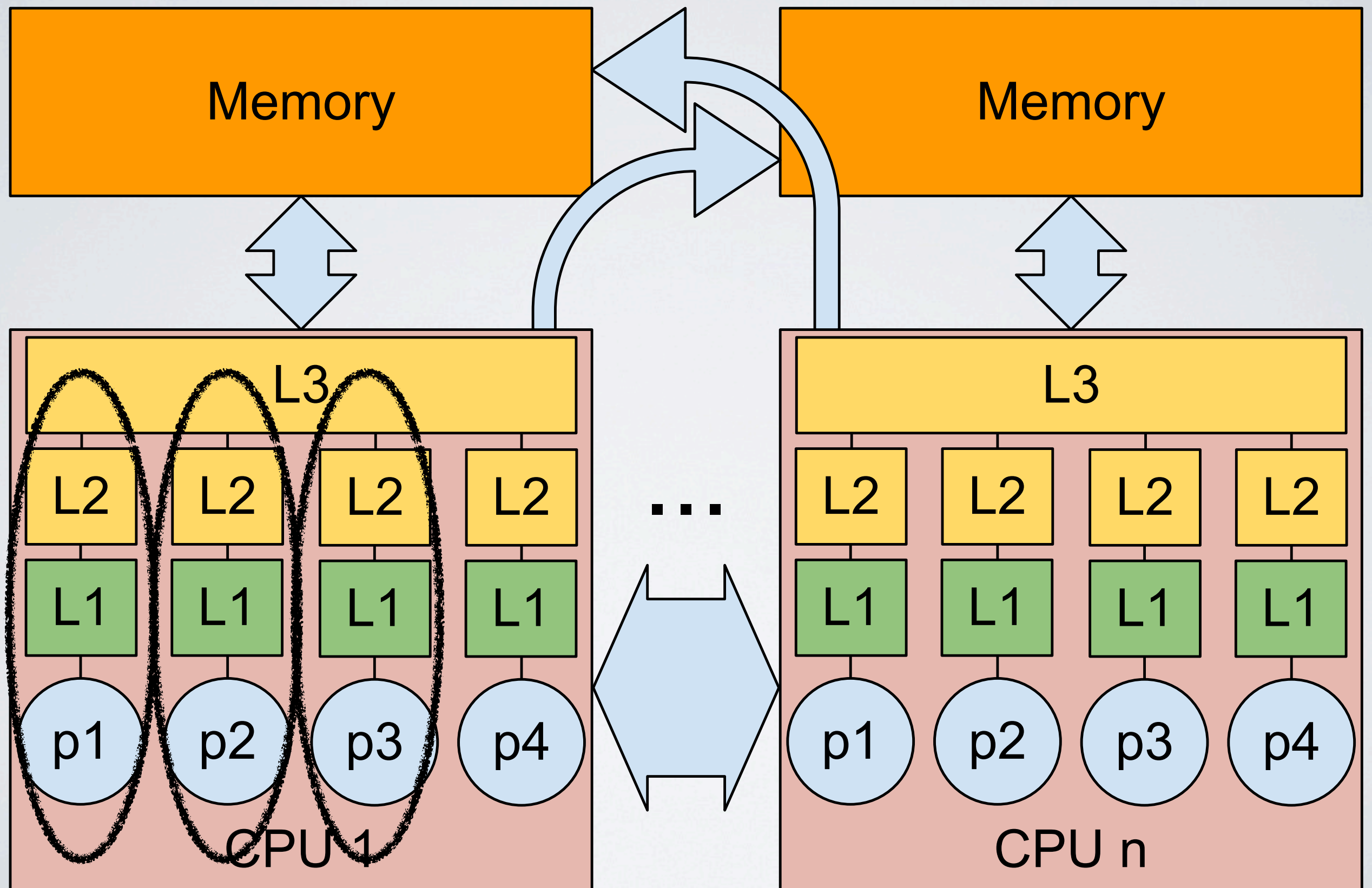
ADVANTAGE OF A SMALL WS



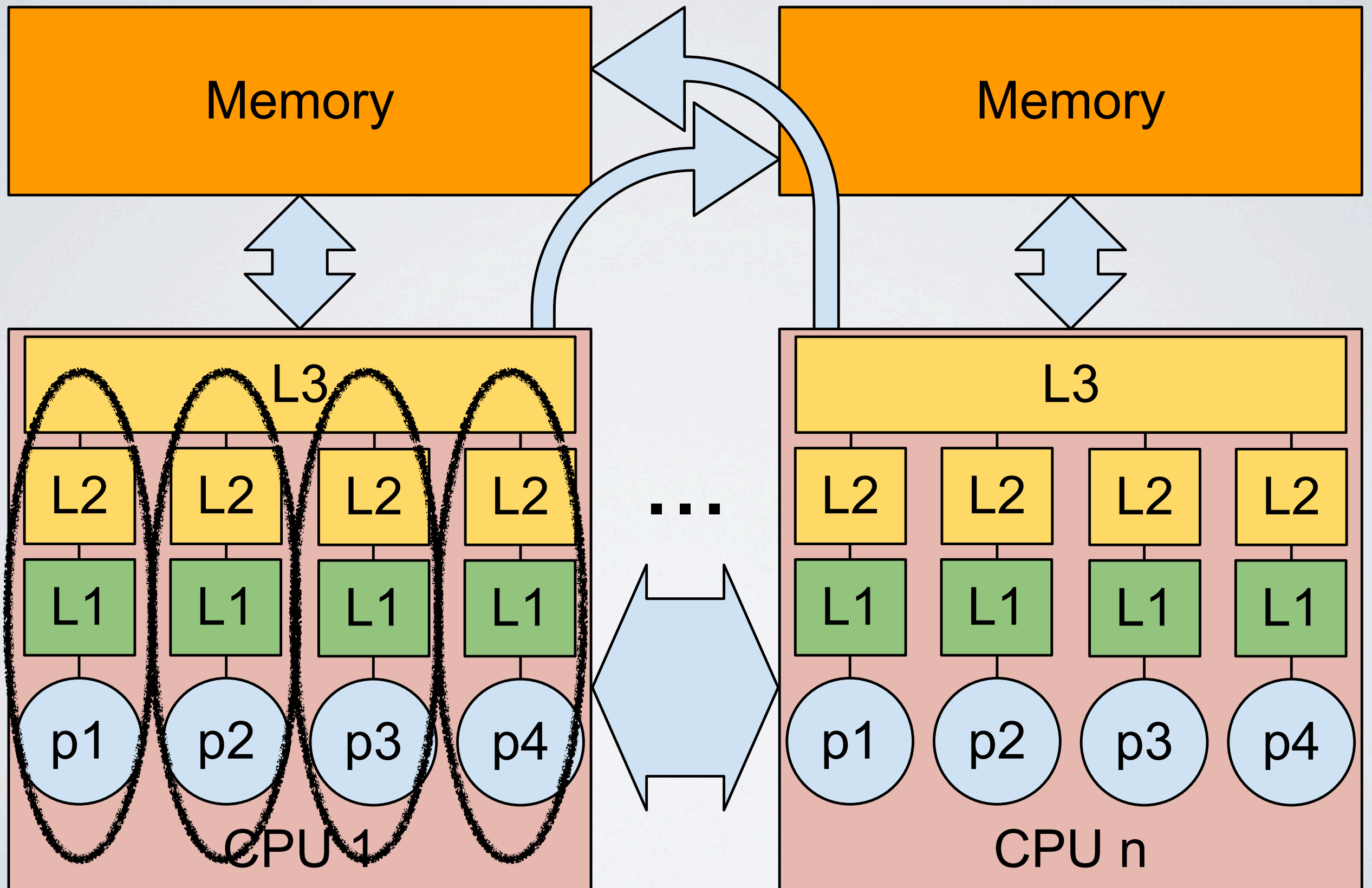
ADVANTAGE OF A SMALL WS



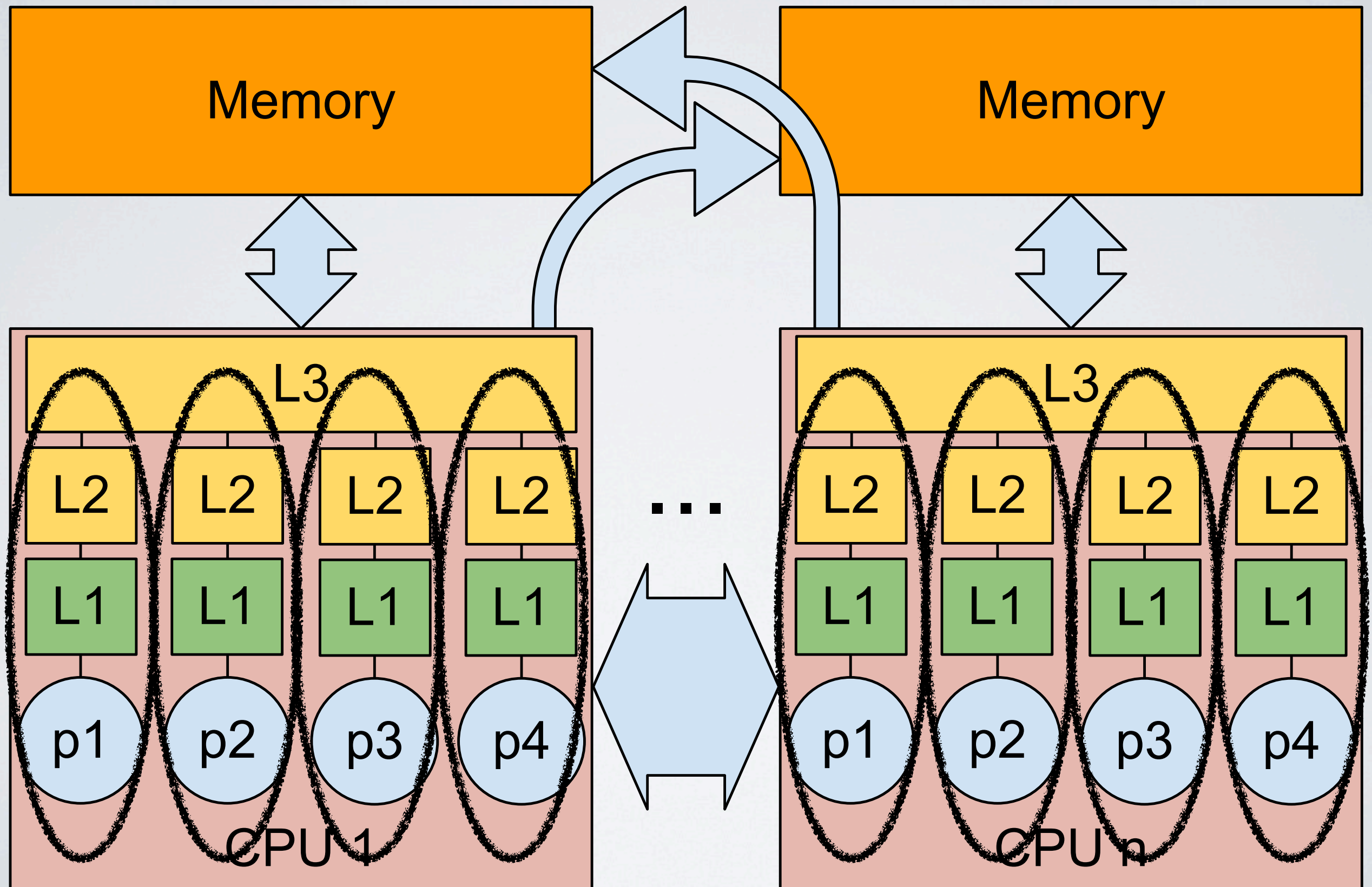
ADVANTAGE OF A SMALL WS



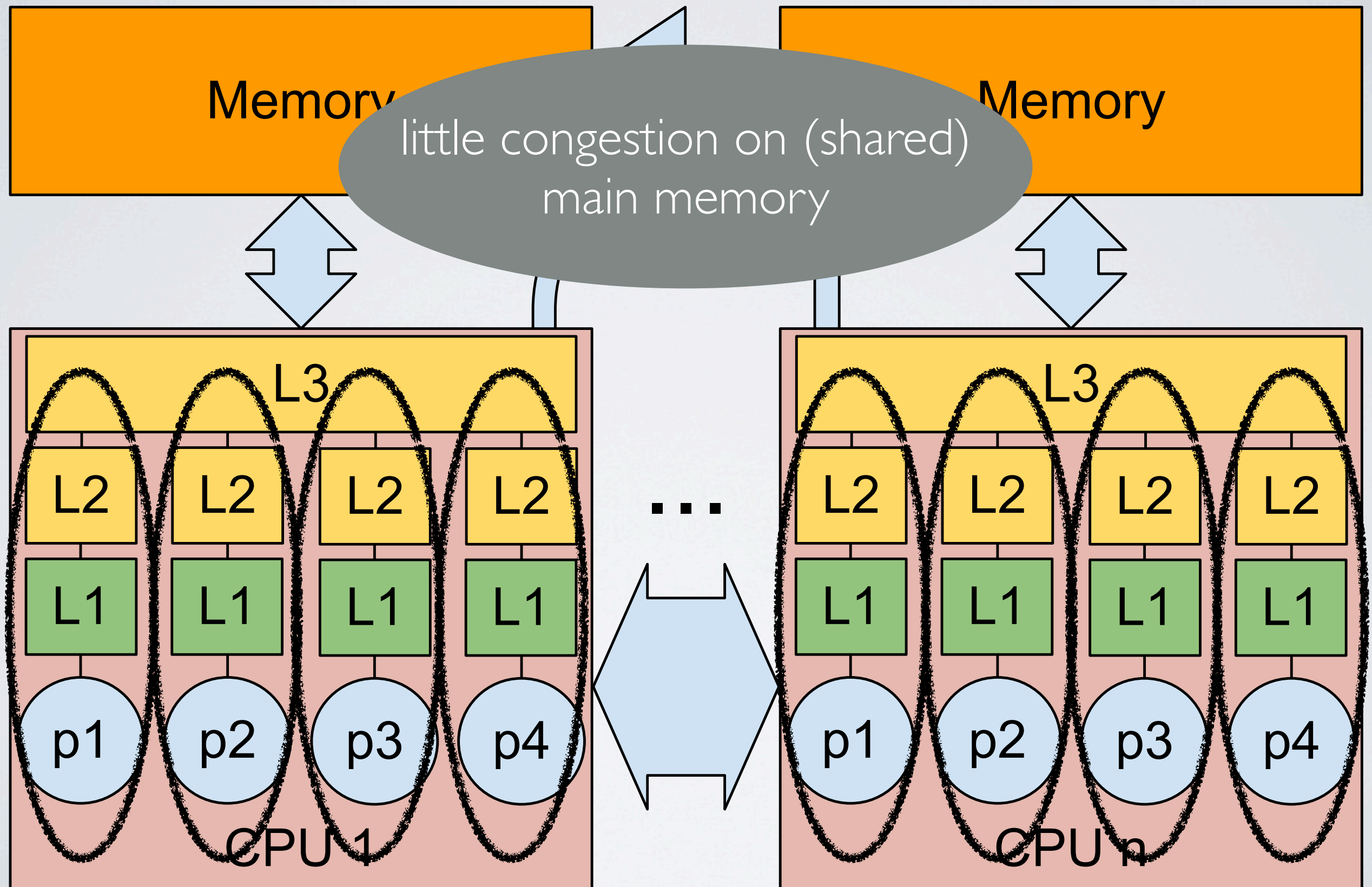
ADVANTAGE OF A SMALL WS



ADVANTAGE OF A SMALL WS



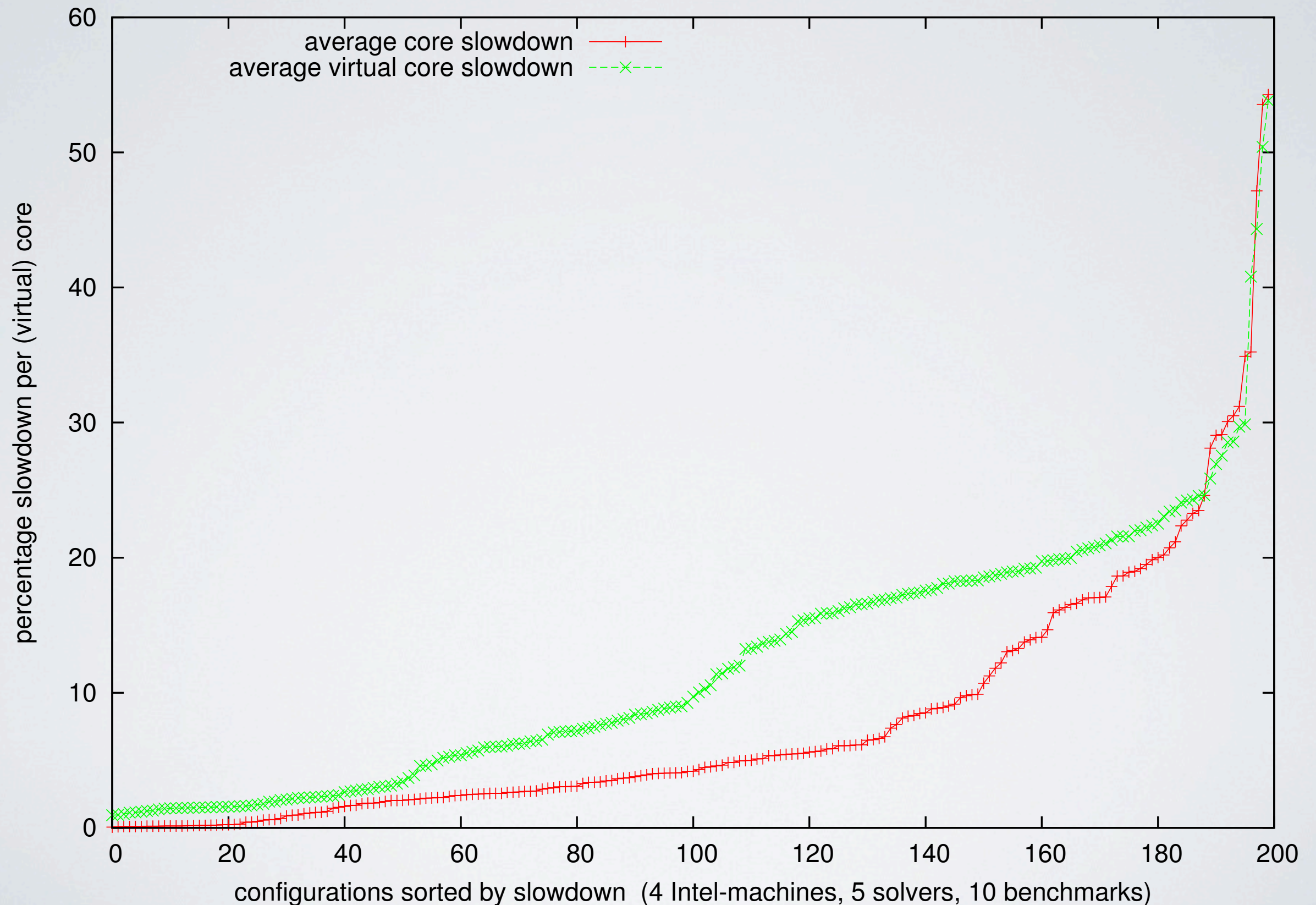
ADVANTAGE OF A SMALL WS



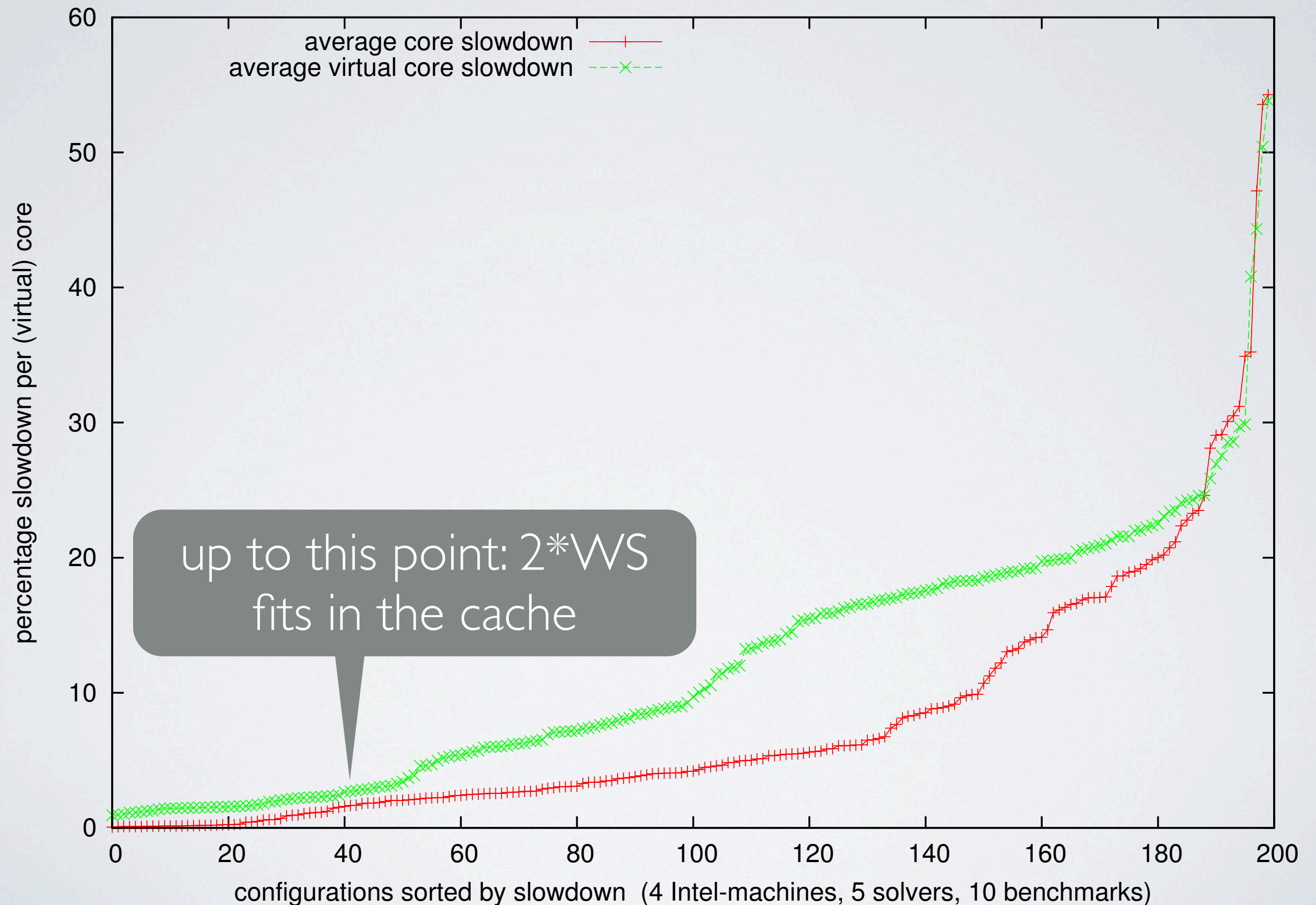
IMPACT OF HYPER THREADING

- Average core slowdown: average slowdown when adding one job and $\#jobs < \#cores$
- Average virtual core slowdown: average slowdown when adding one job and $\#cores < \#jobs \leq \#virtual\ cores$

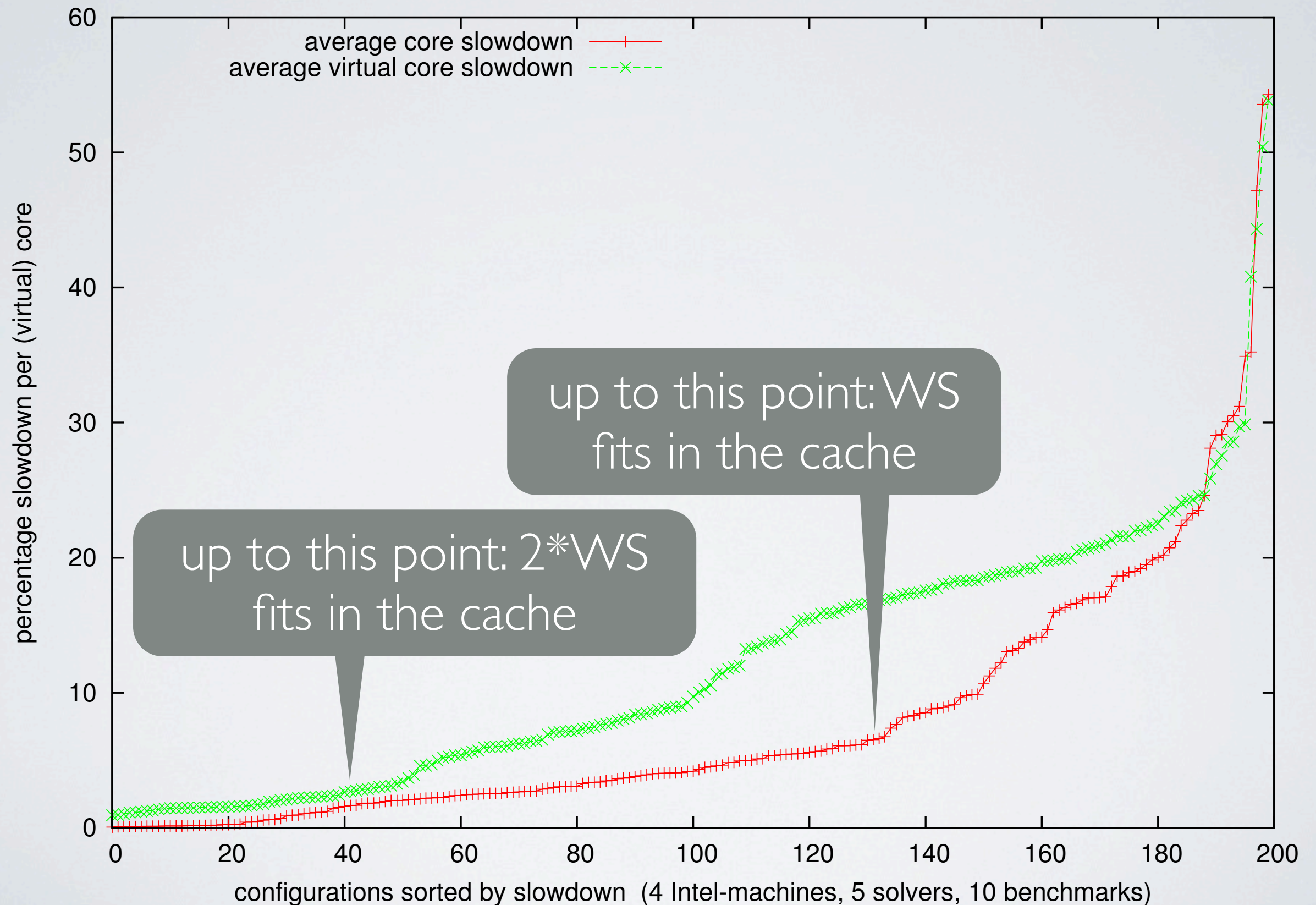
EXPERIMENT: ACSD v.s. AVCSD



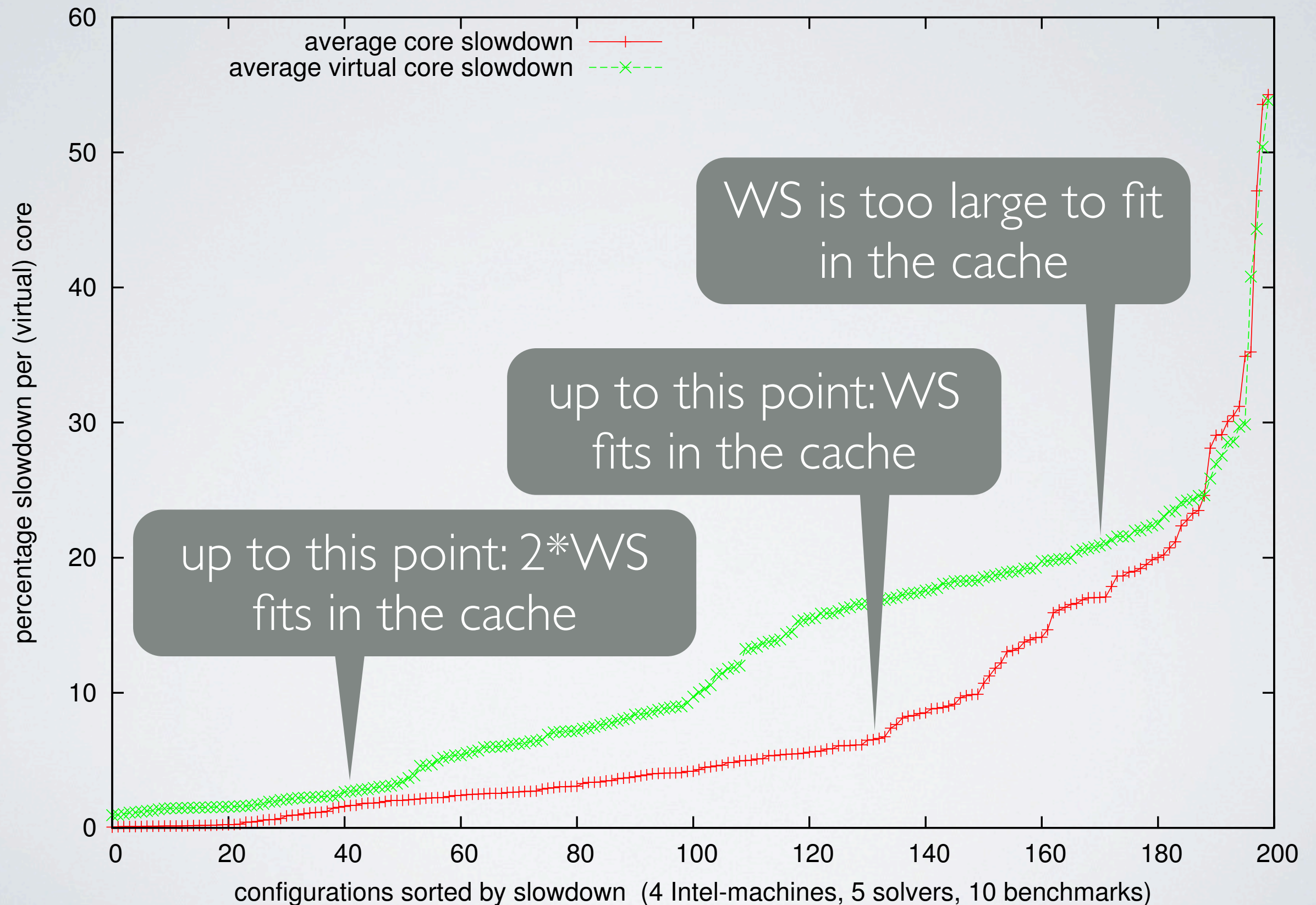
EXPERIMENT: ACSD v.s. AVCSD



EXPERIMENT: ACSD v.s. AVCSD



EXPERIMENT: ACSD v.s. AVCSD



CONCLUSION & FUTURE WORK

- We have analyzed the expected slowdown for PPP
- Results suggest employing low working set solvers for PPP
- Future work:
 - Analyze slowdown w.r.t. cache size (outlook for future systems)
 - Dynamic working set estimators to control the number of parallel SAT solvers

THANK YOU
QUESTIONS?