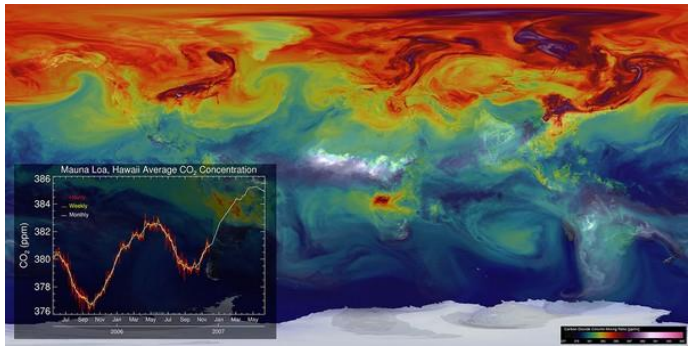
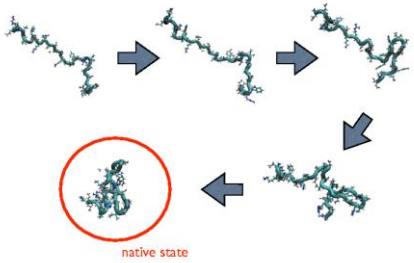
Placement of Virtual Containers on NUMA Systems

Justin Funston*, Maxime Lorrillere†, and Alexandra Fedorova, University of British Columbia
Baptiste Lepers, EPFL
David Vengerov and Jean-Pierre Lozi, Oracle Labs
Vivien Quéma, IMAG

*** Currently Huawei R&D**
† Currently Arista Networks

Motivation



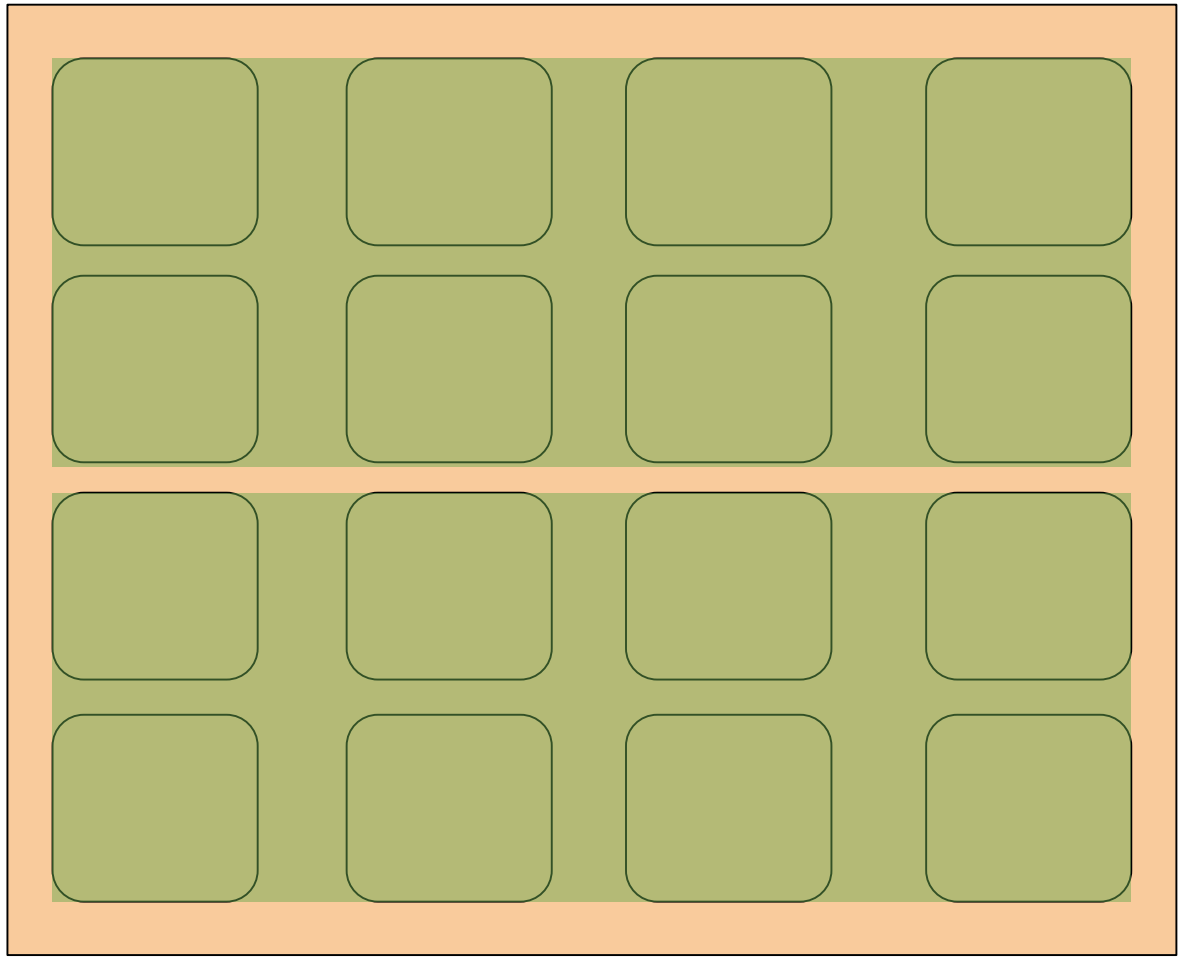


Data Centers:

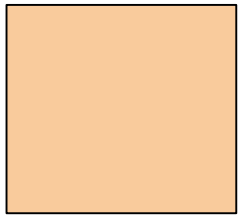
3% of global electricity usage

86% used by servers+cooling

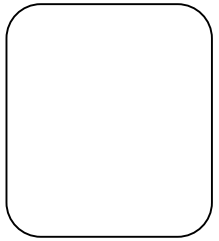
Motivation – Server Packing



Container/Application

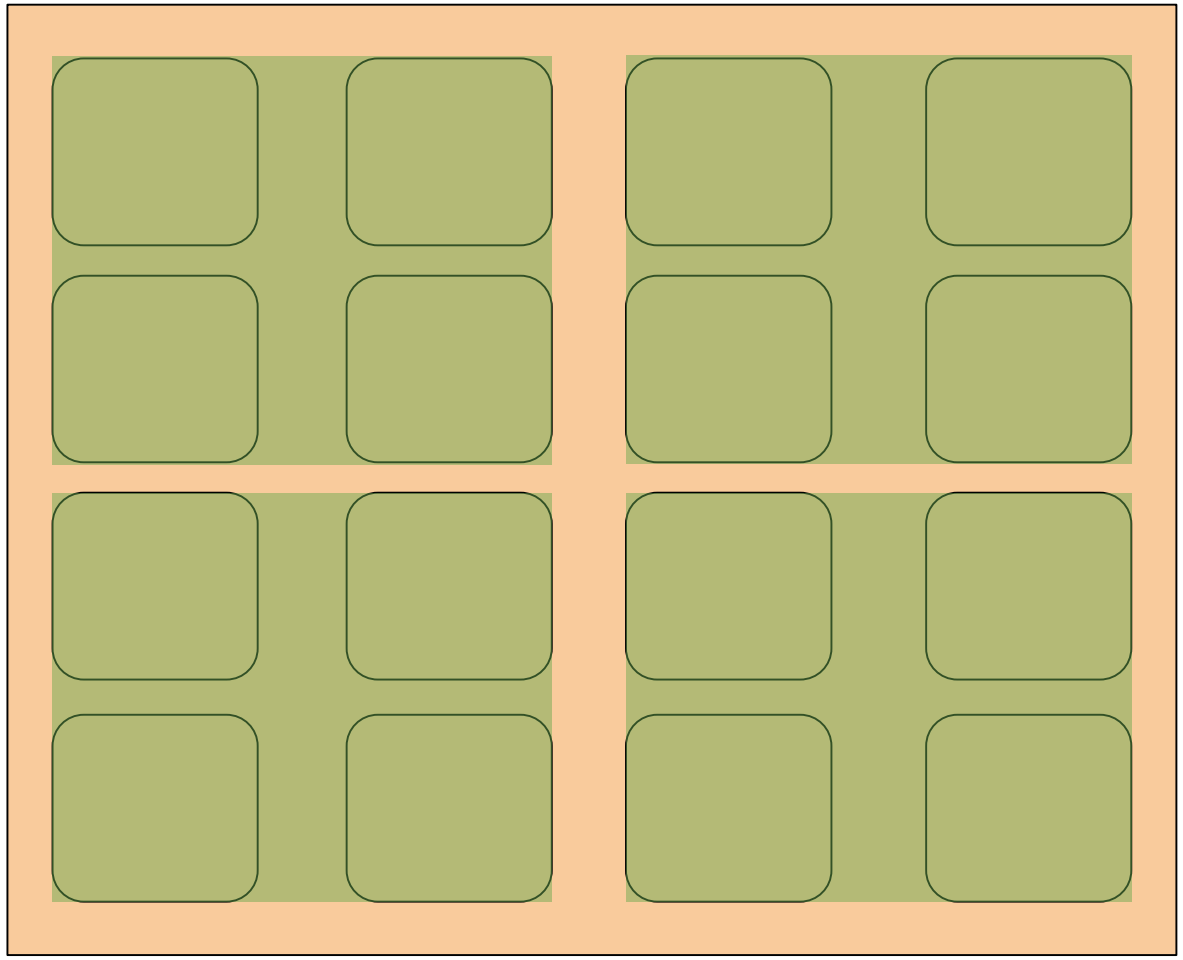


Server

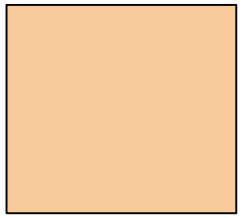


Resource Node

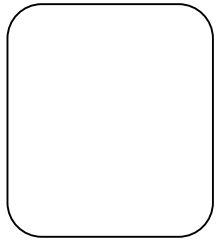
Motivation – Server Packing



Container/Application

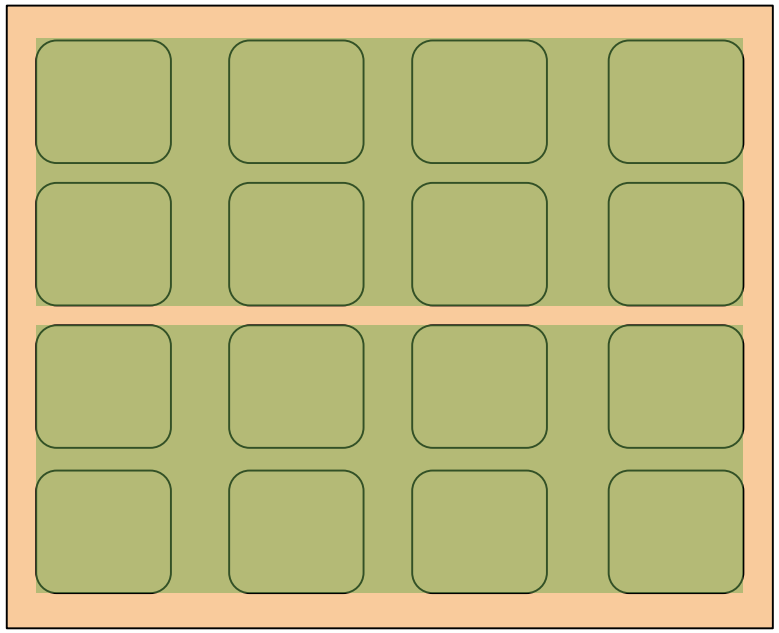


Server

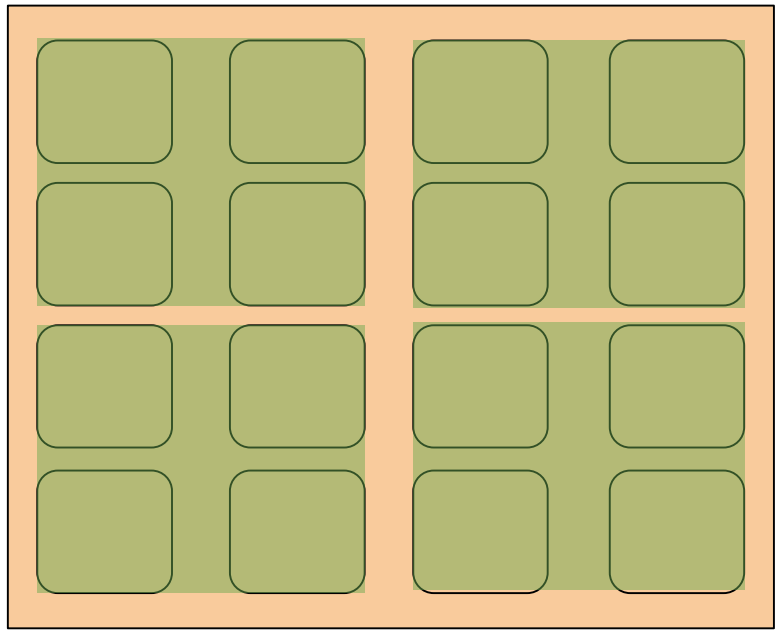


Resource Node

Motivation – Server Packing

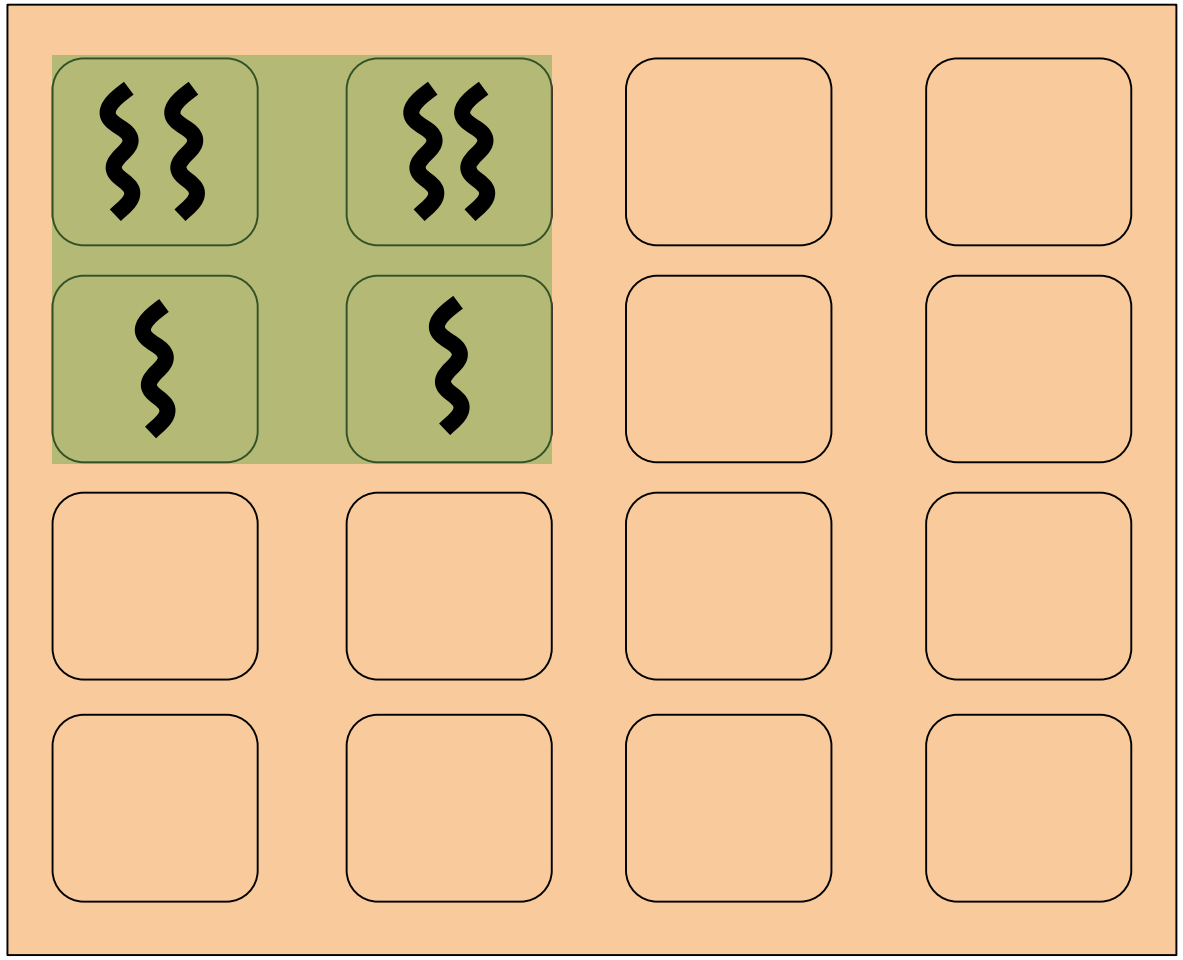


VS.



- **Half as many servers!**
- **Half as much energy!**
- **Half as much infrastructure!**
- **Performance?**

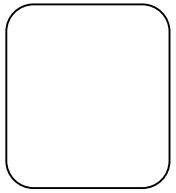
Motivation – Server Packing



Container/Application

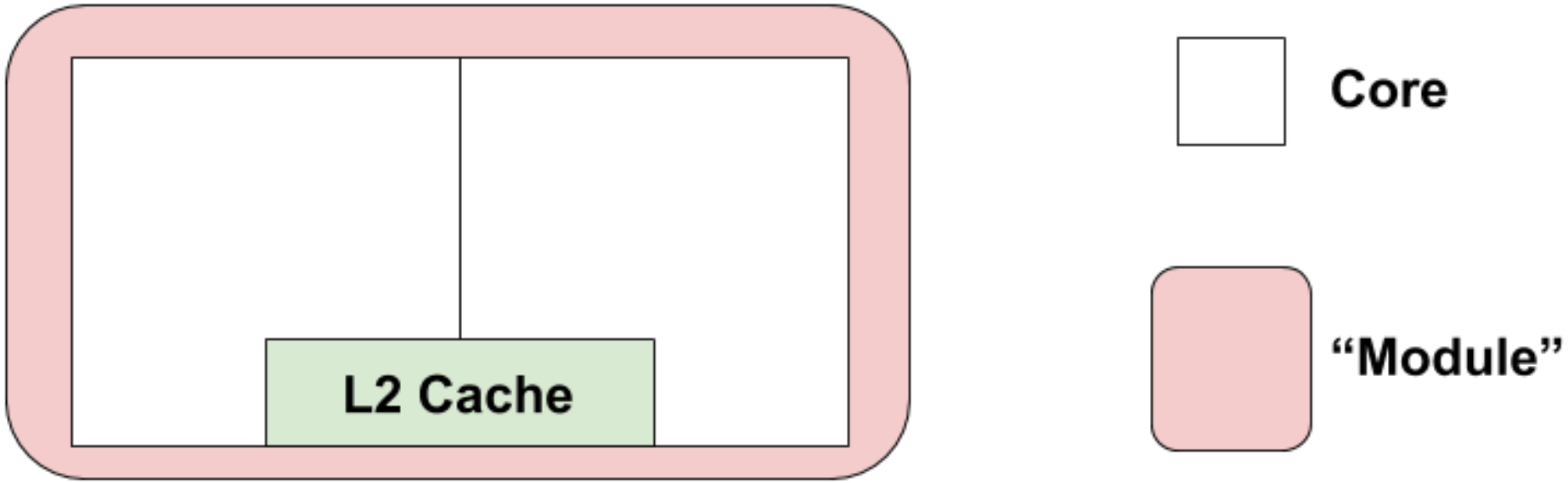
Application Thread

Server

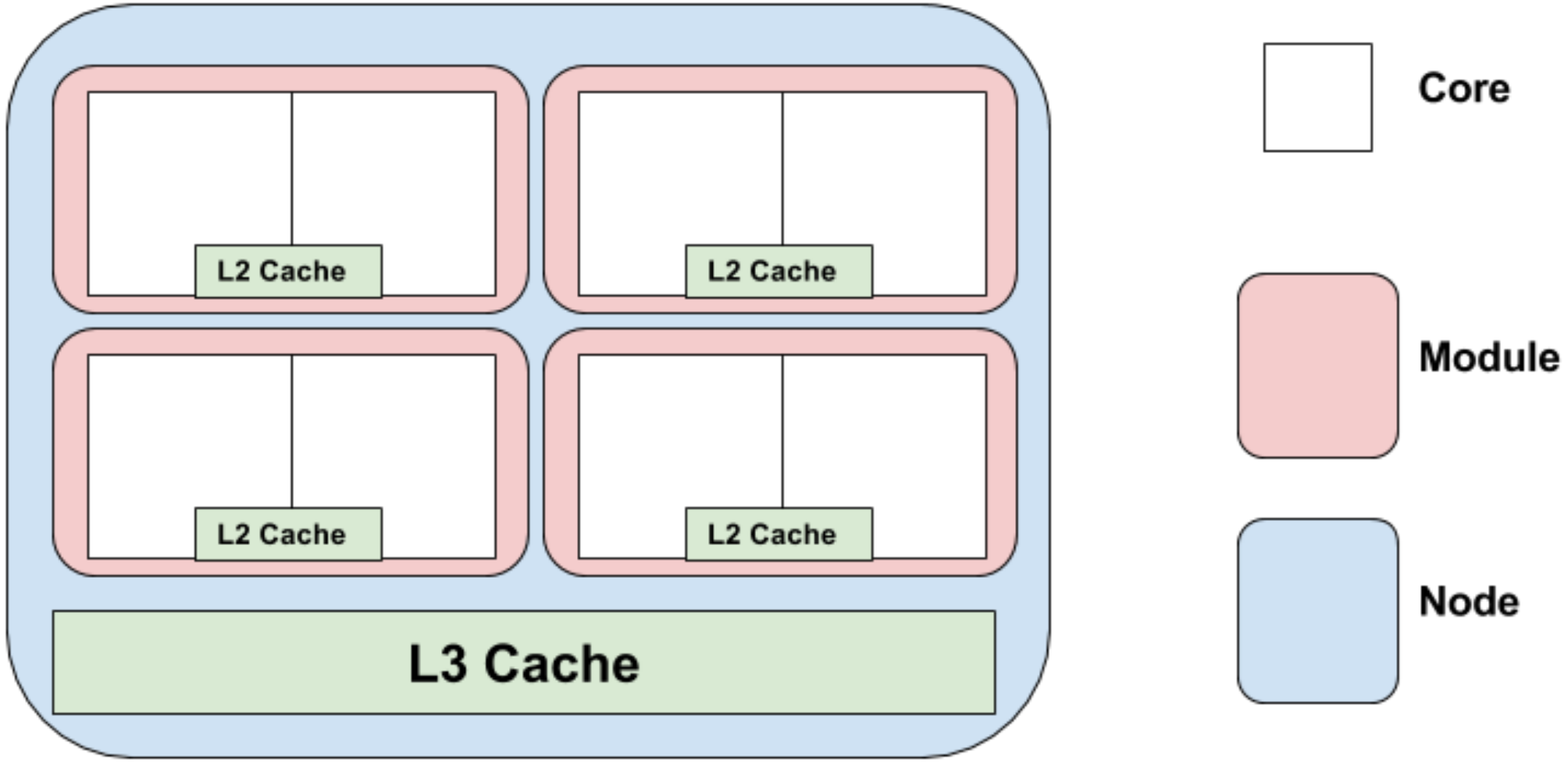


Resource Node

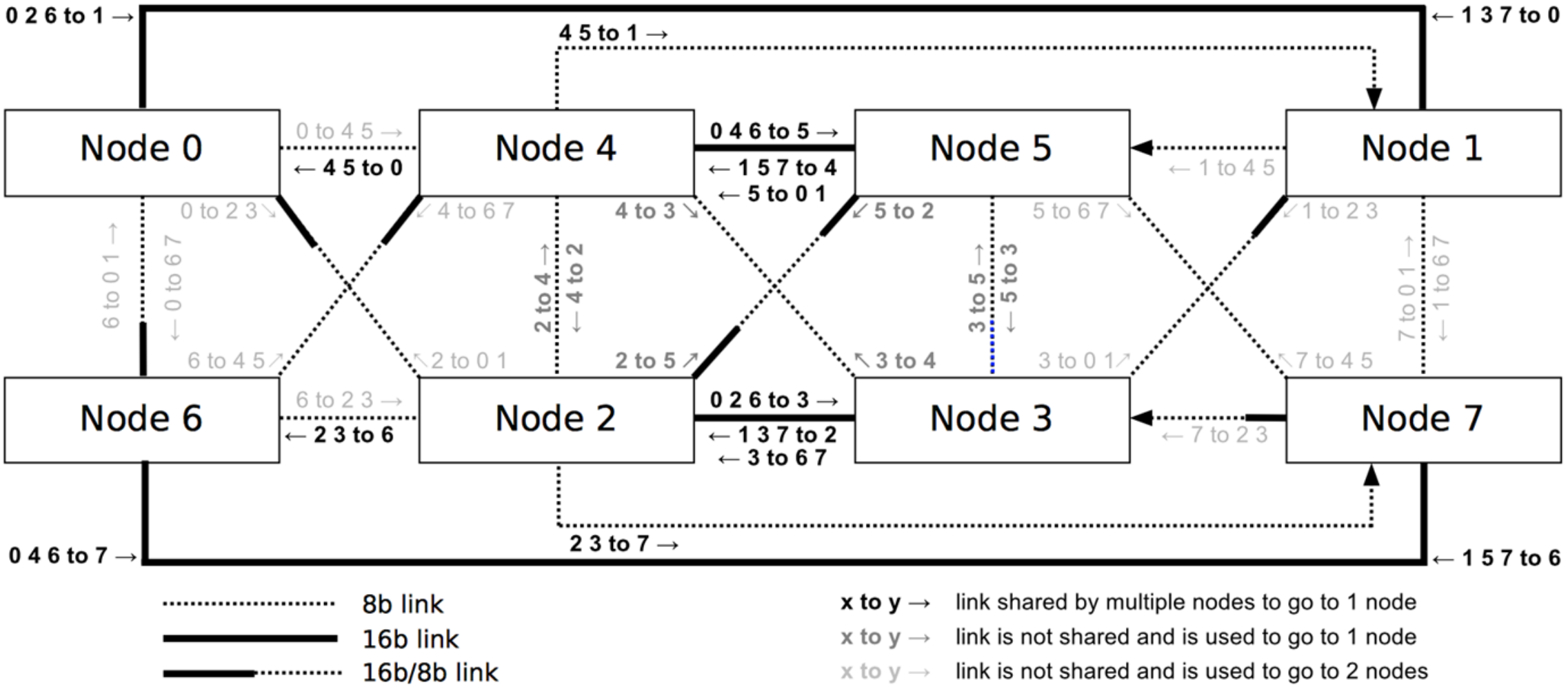
Background – “Module”



Background – NUMA Node



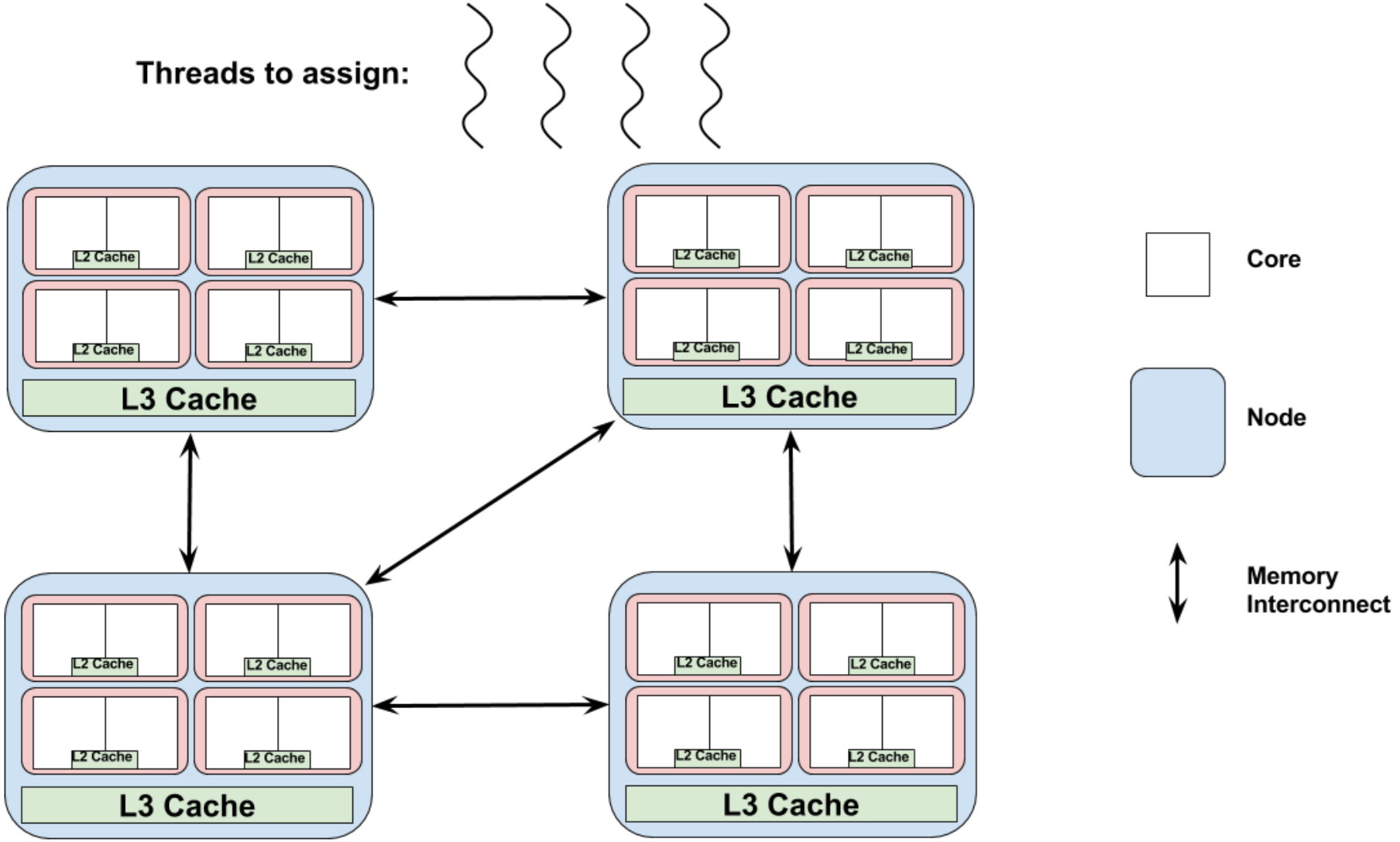
Background – Interconnect Topology



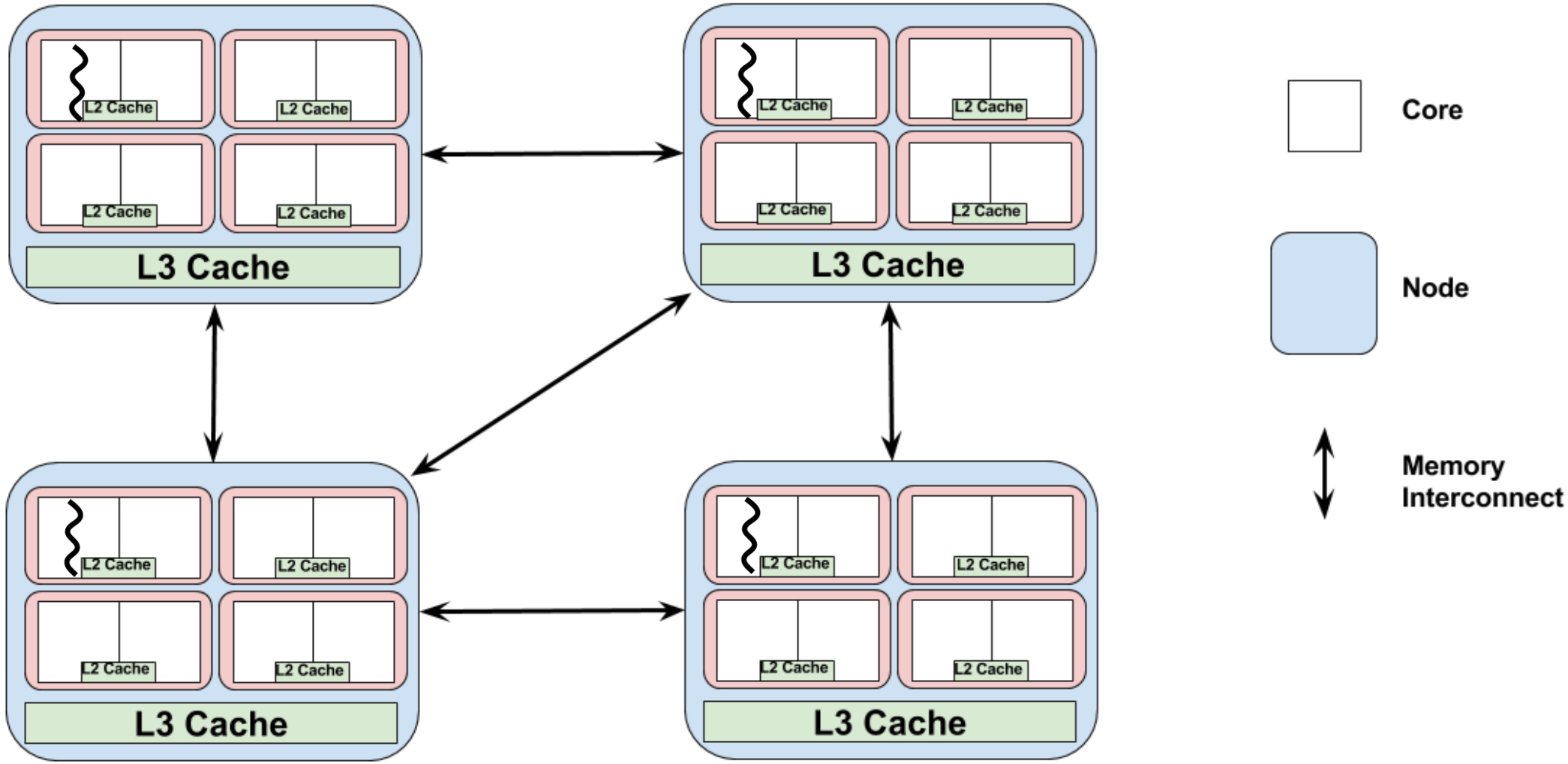
Assumptions

- The number of vCPUs a container uses is fixed
- Max of one vCPU per core
- Containers packed together should not interfere with each other – containers will not share NUMA nodes

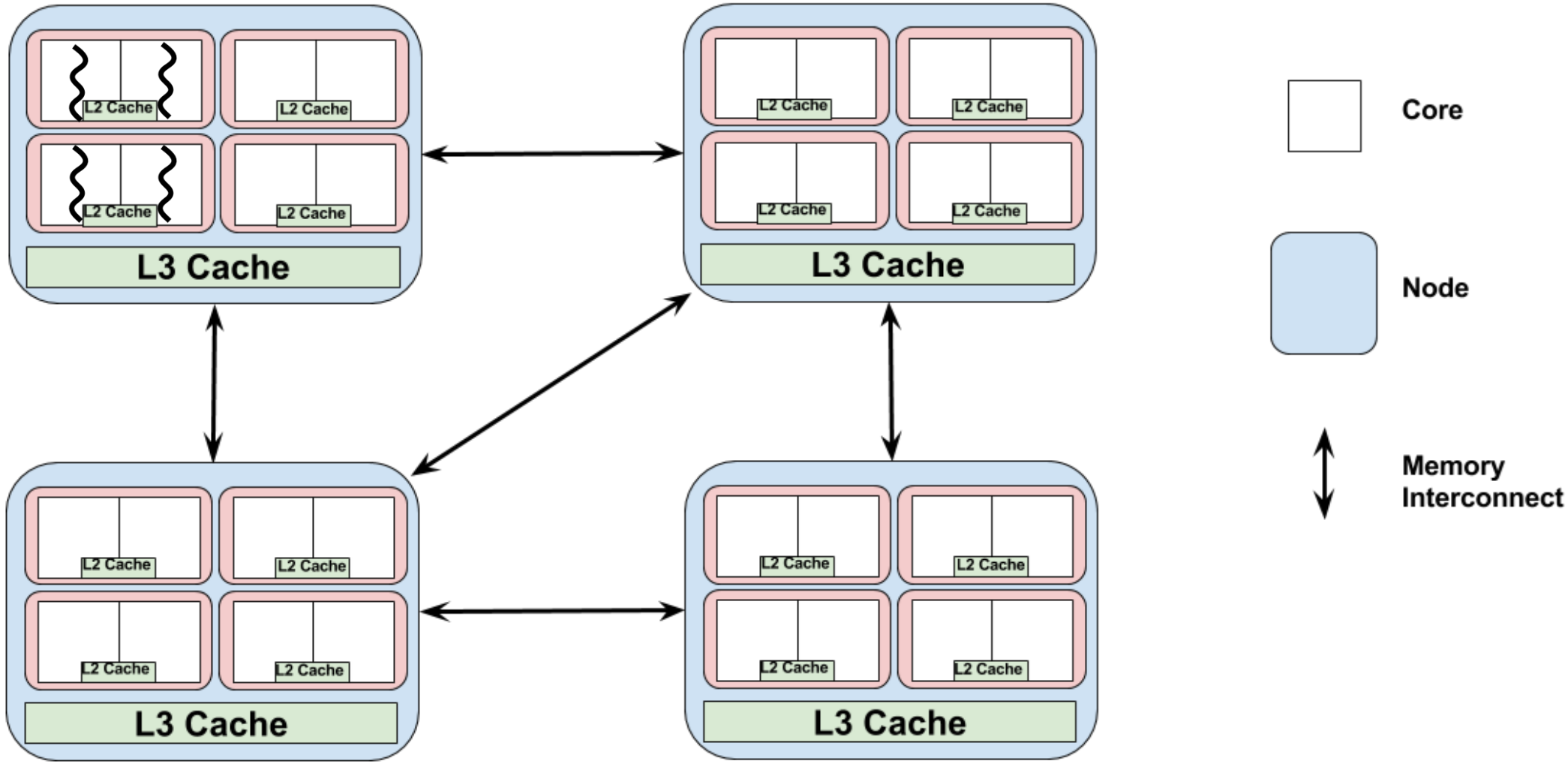
Placements


Threads to assign:
L2 Cache
L3 Cache
Core
Node
Memory
Interconnect

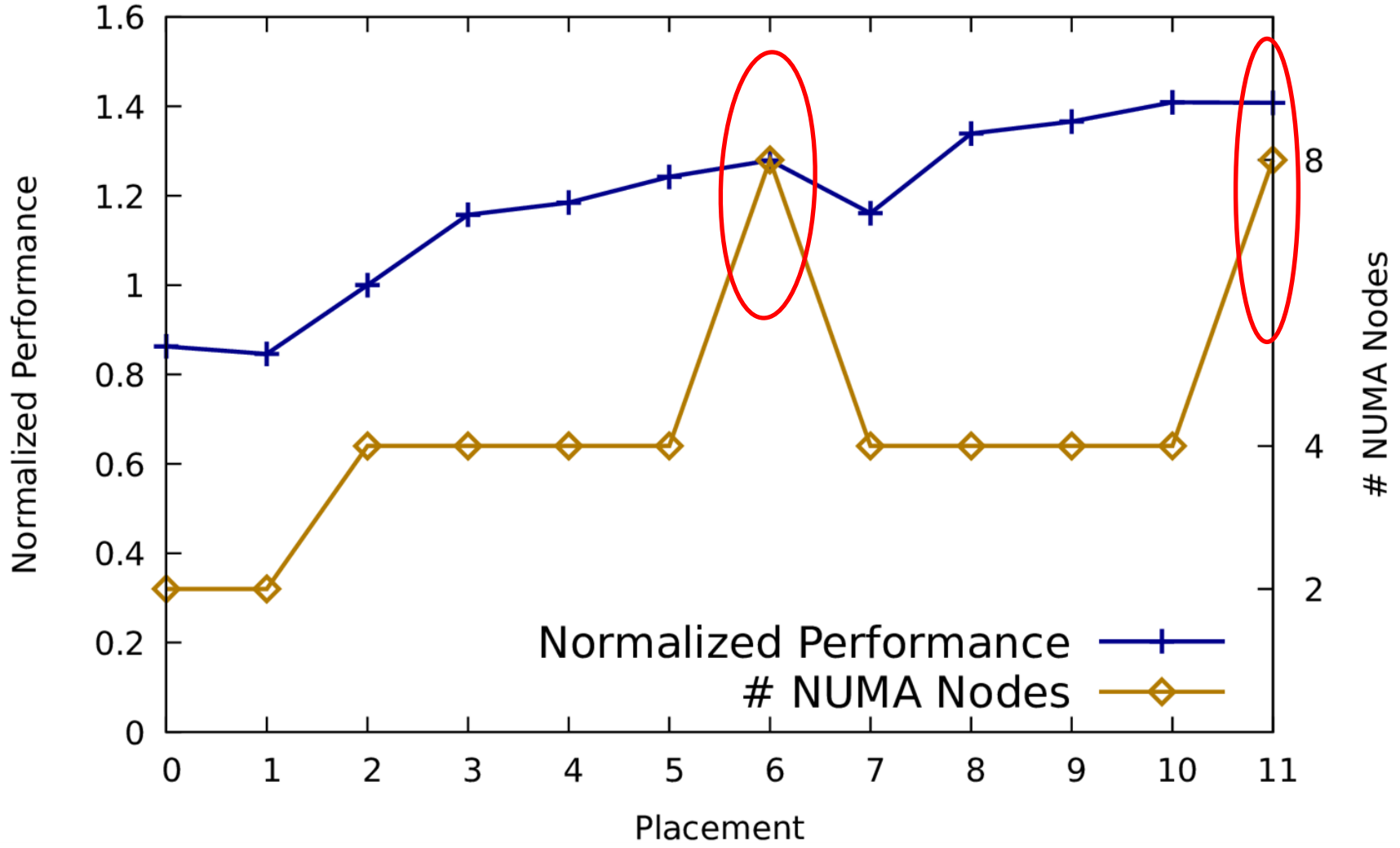
Placements



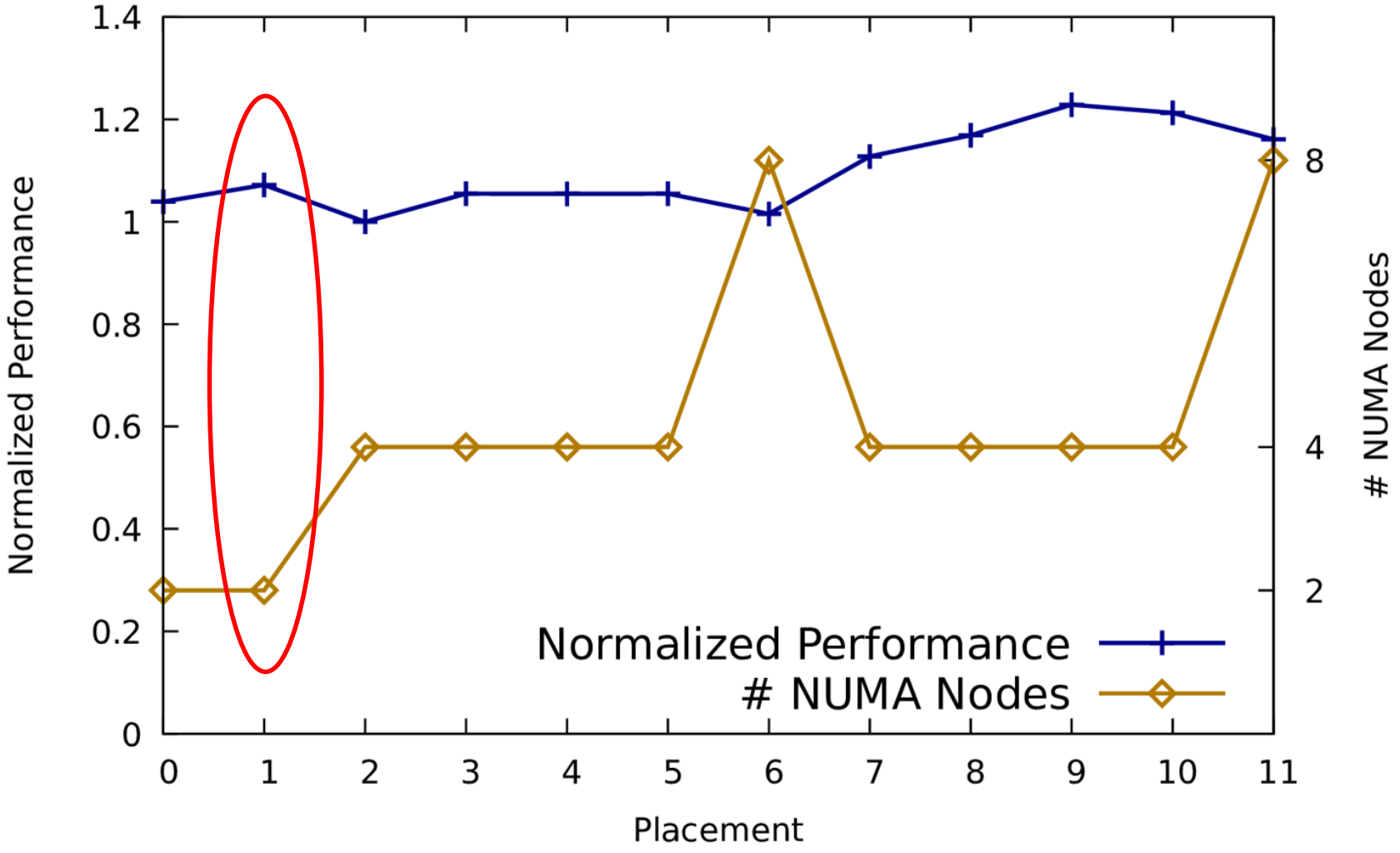
Placements



Motivation – ua.B (scientific simulation)

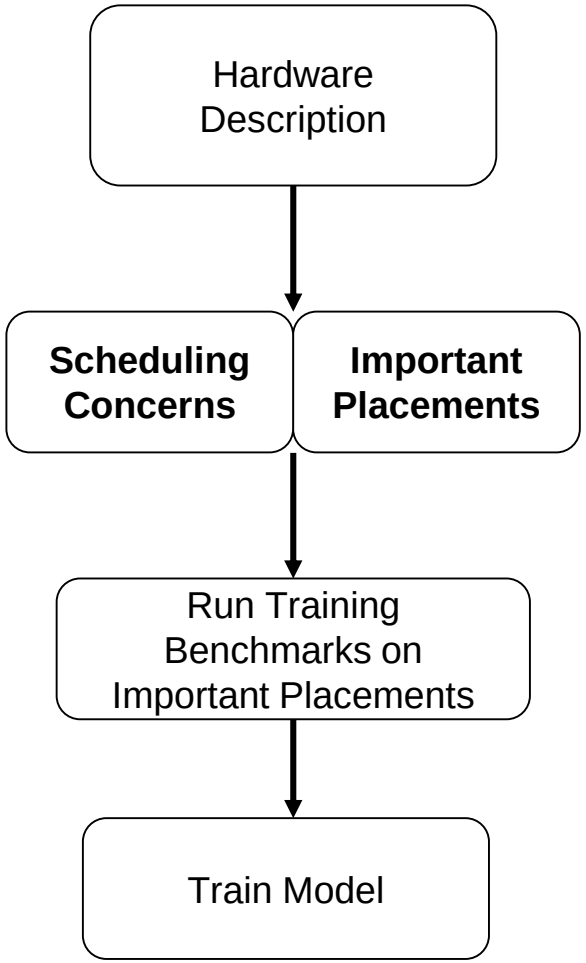


Motivation – Spark Pagerank

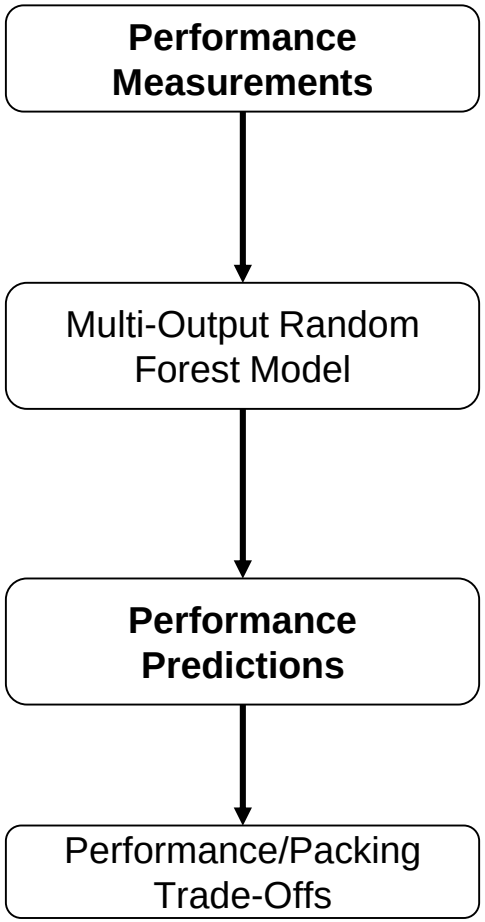


Workload Placement Overview

Abstract Machine Model (Offline)



Performance Prediction Model (Online)

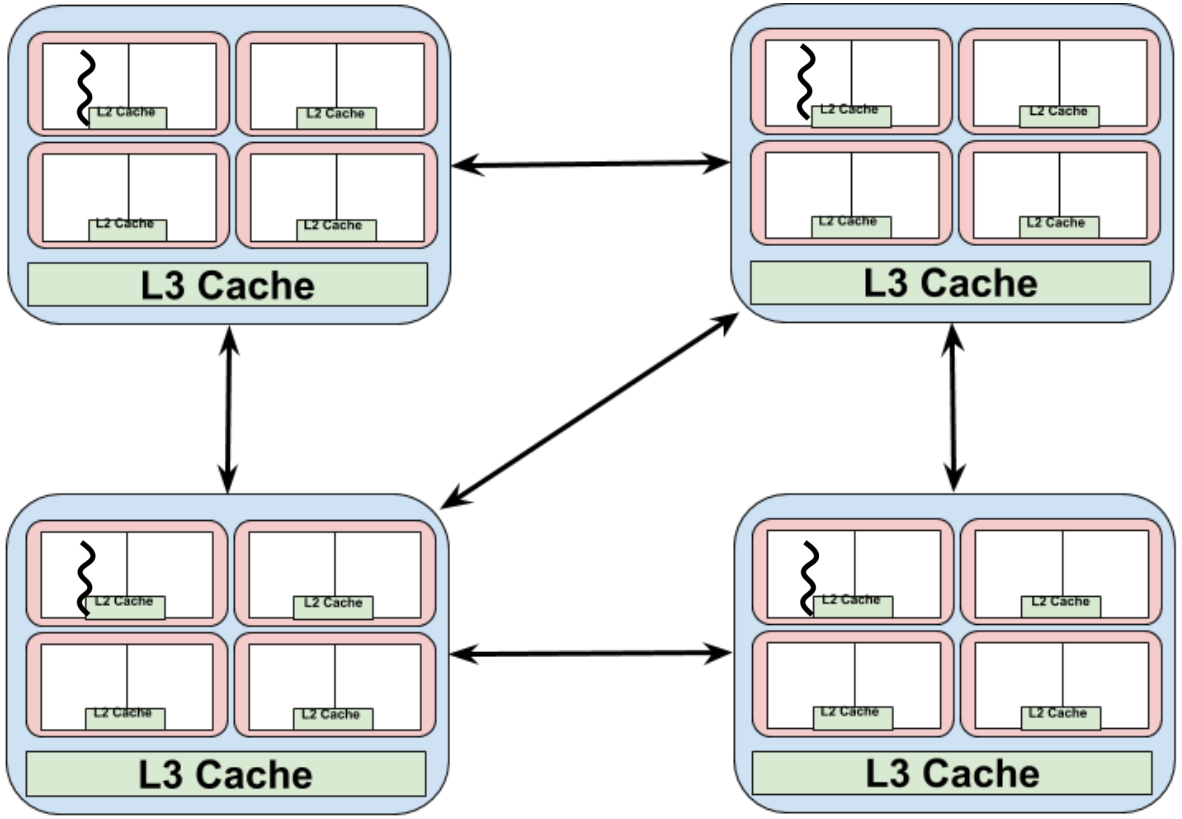


Abstract Machine Model

- How to represent placements?
  - Too many (e.g. trillions for 16 threads on 64 cores) for a naïve approach
  - Need to exploit symmetry

Scheduling Concerns

Placement:



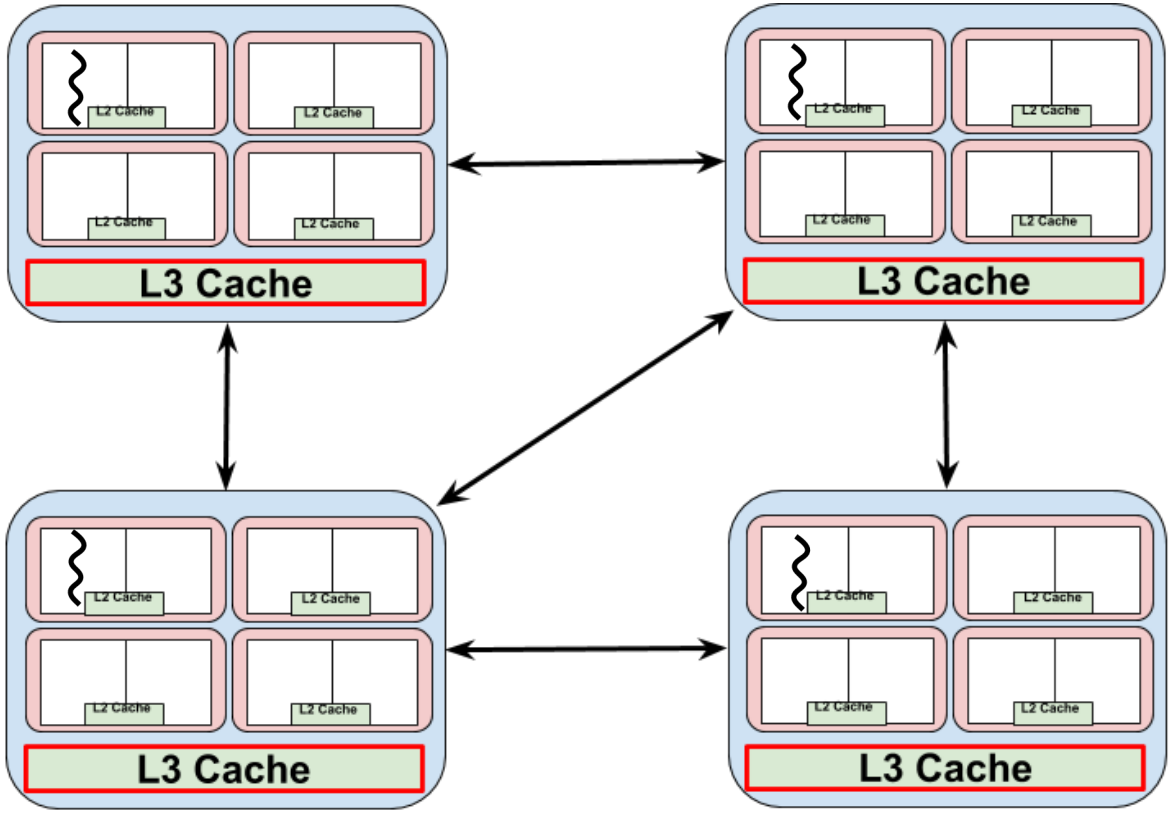
Scheduling Concern:

Score:

4

Scheduling Concern Example – L3

Placement:



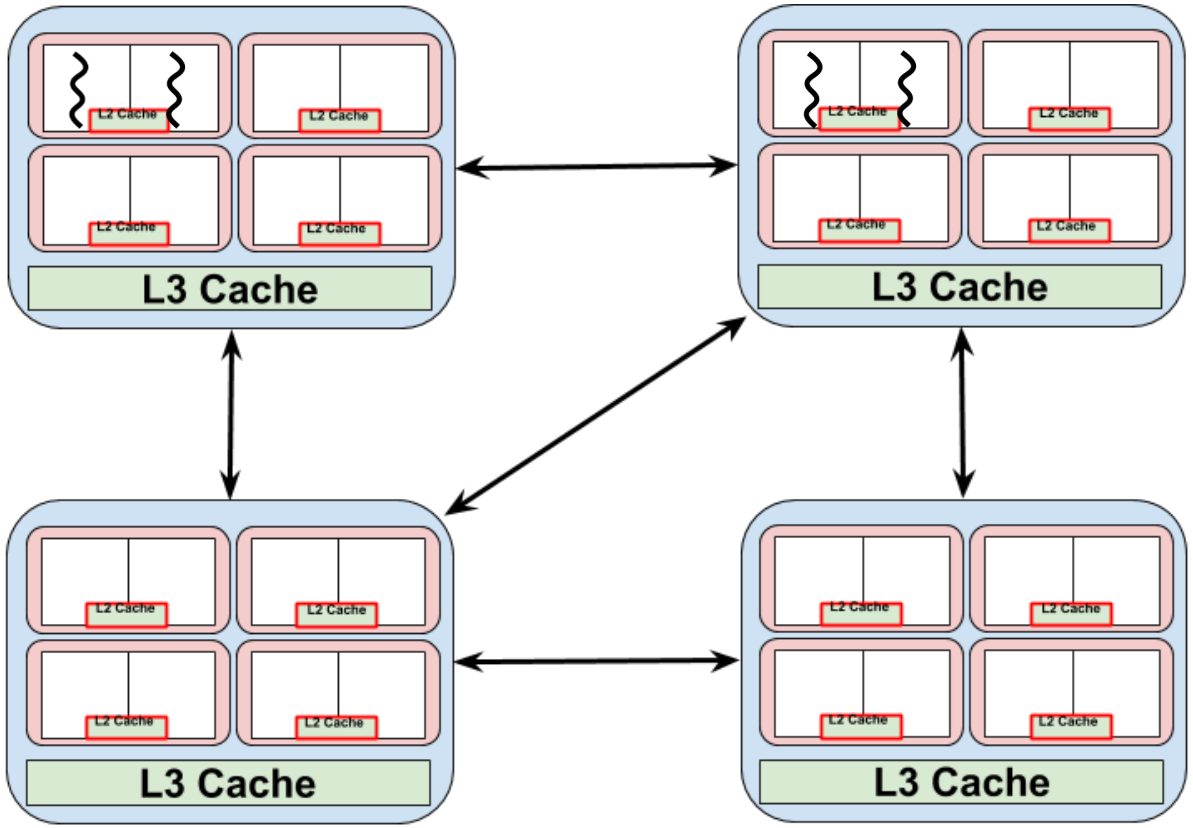
Scheduling Concern:

Score = # L3 Caches In-Use

4

Scheduling Concern Example – L2

Placement:



Scheduling Concern:

Score = # L2 Caches In-Use

2

Abstract Machine Model – Important Placements

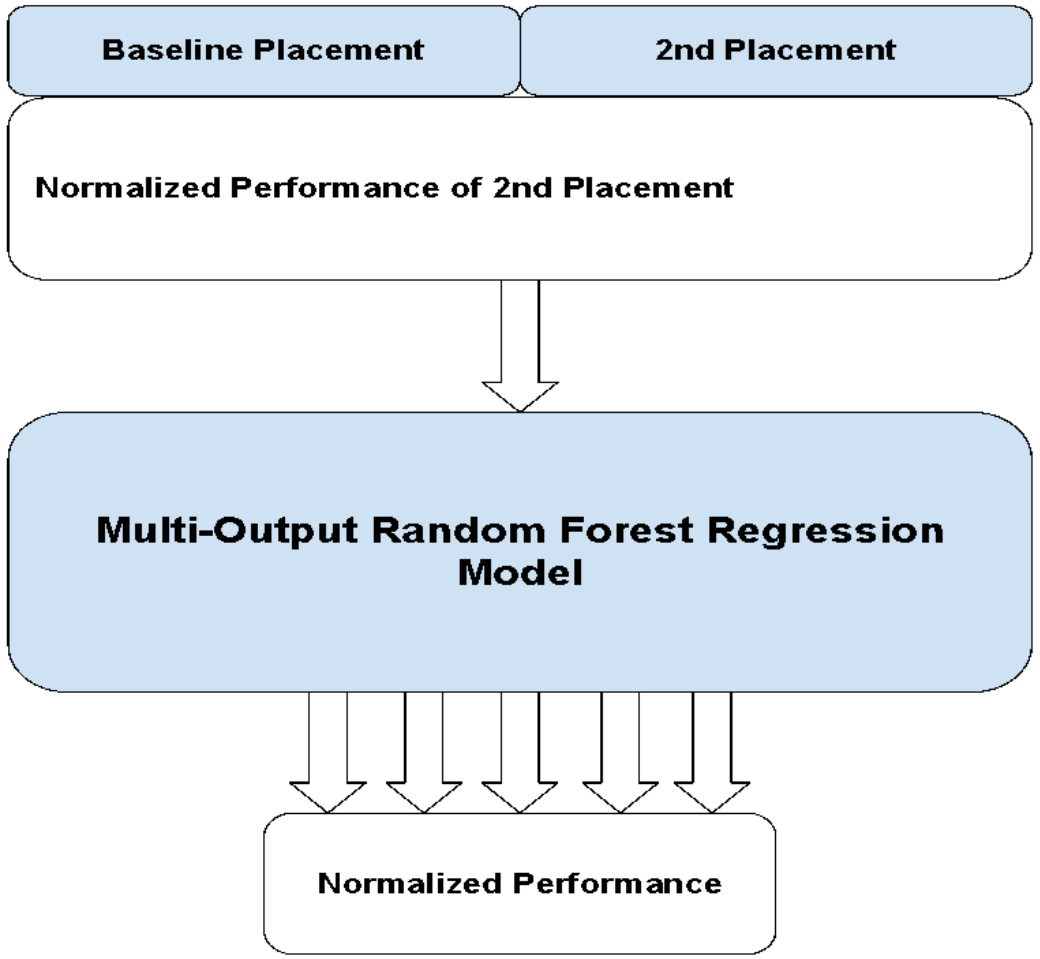
- **Scheduling concerns + Important placements:**
- **~10^{14} Placements → 12 Placements**
  - 16 threads on 64 cores
- Can train on all important placements

Performance Prediction Model – Features/Inputs

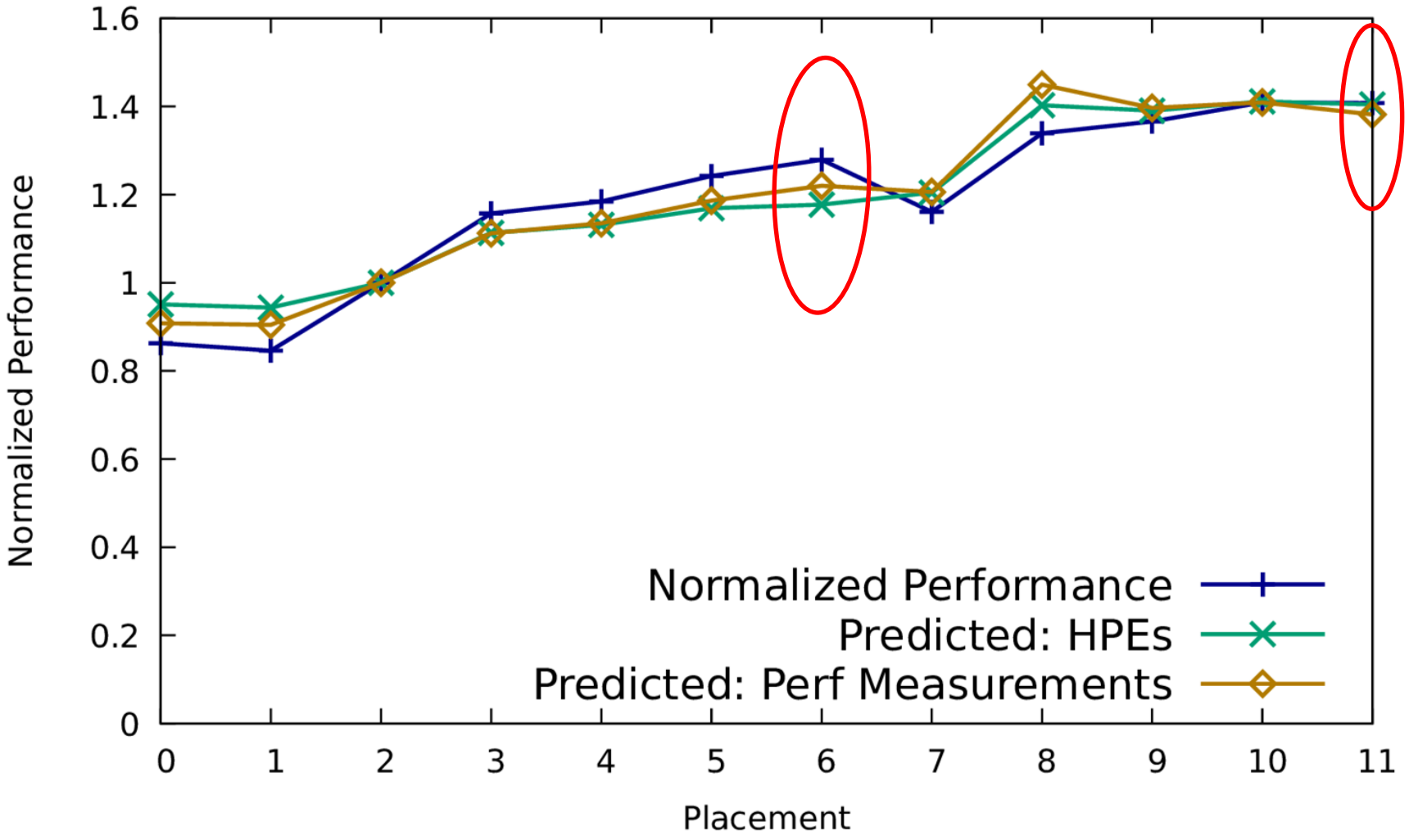
- **Hardware Performance Events (HPEs)**
  - Standard in existing work
  - Surprisingly, poor predictive performance!
  - Excessive training time

- **Performance Measurements**

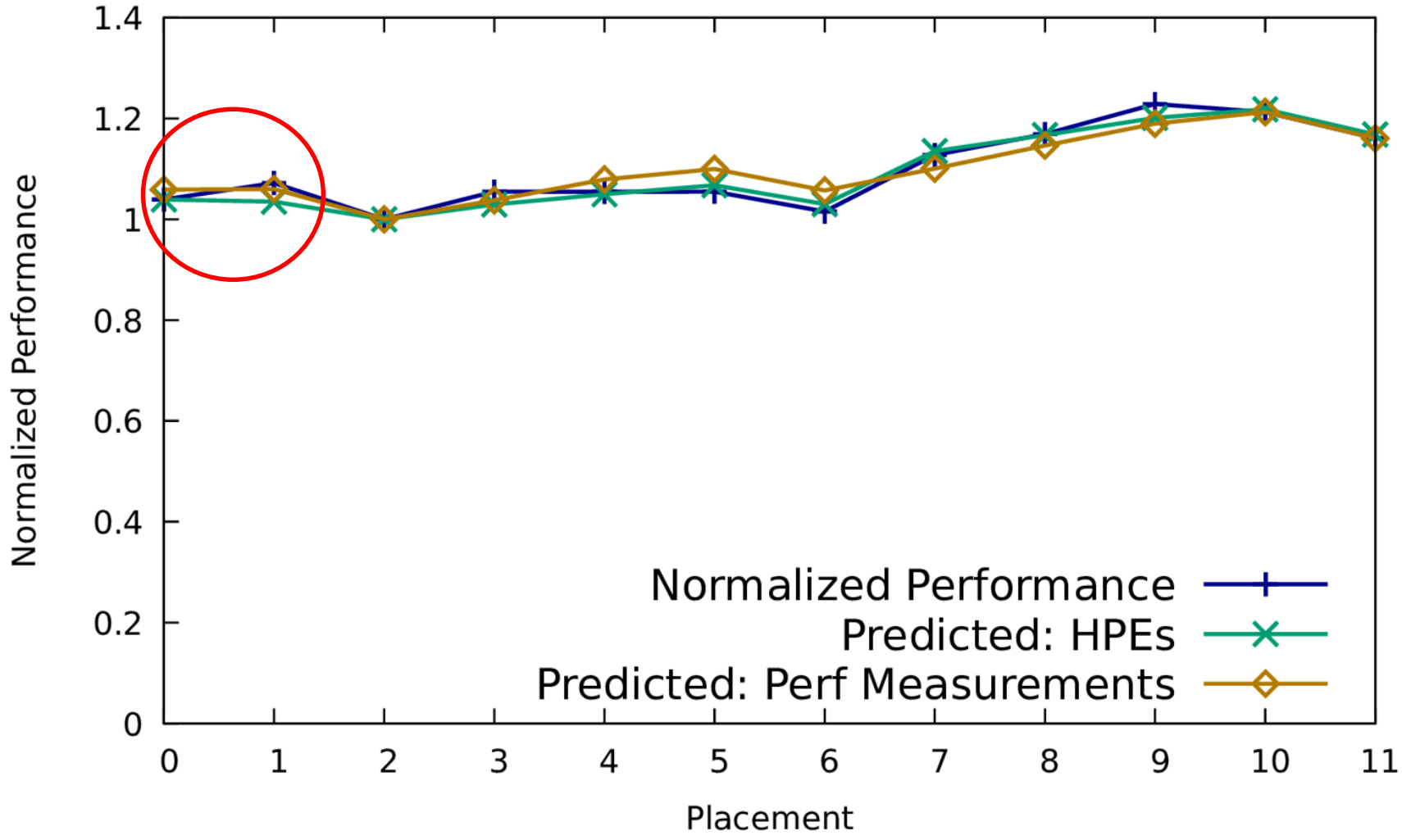
Performance Predictions, Online Inference



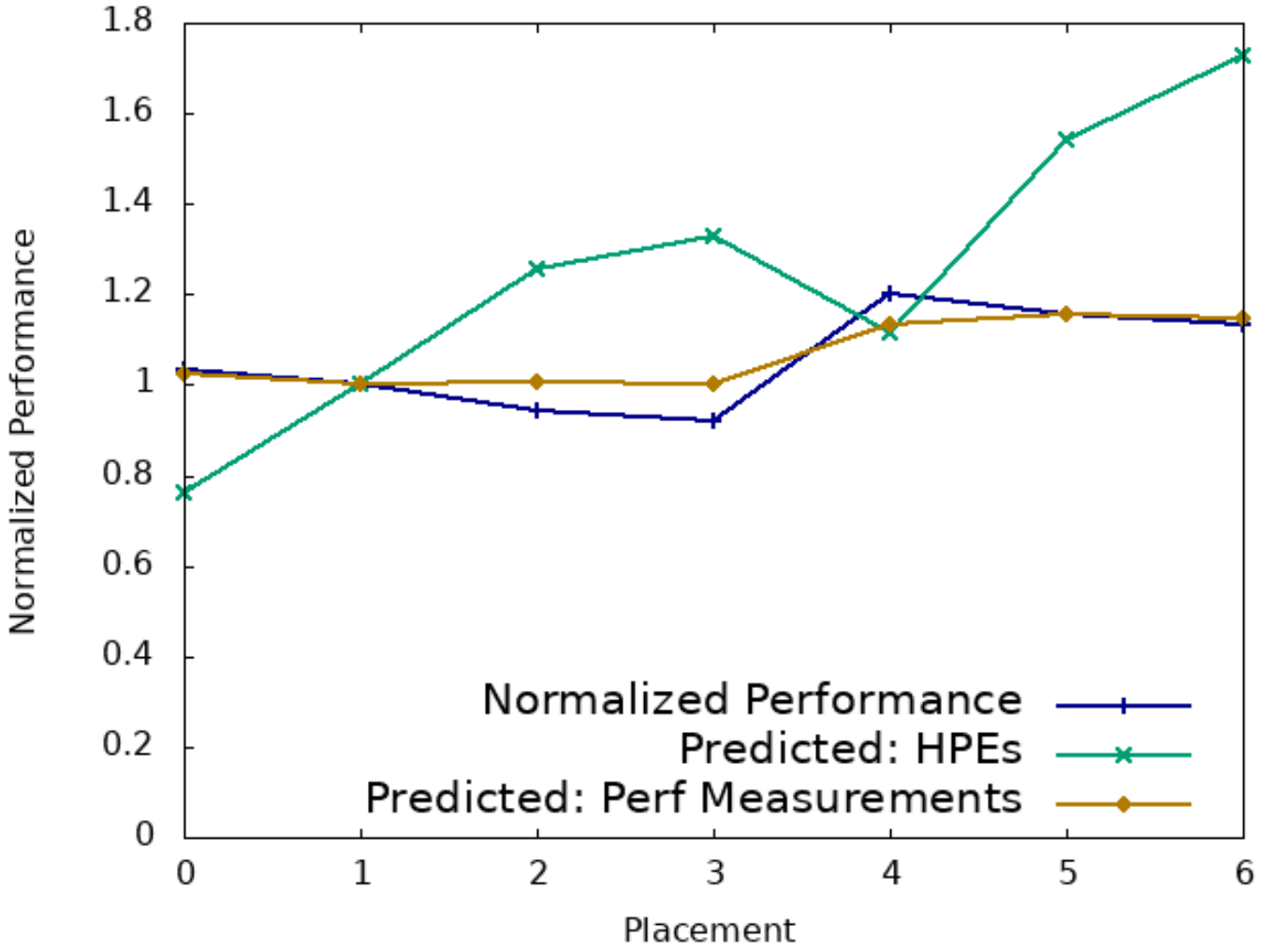
Performance Predictions – ua.B (scientific sim.)



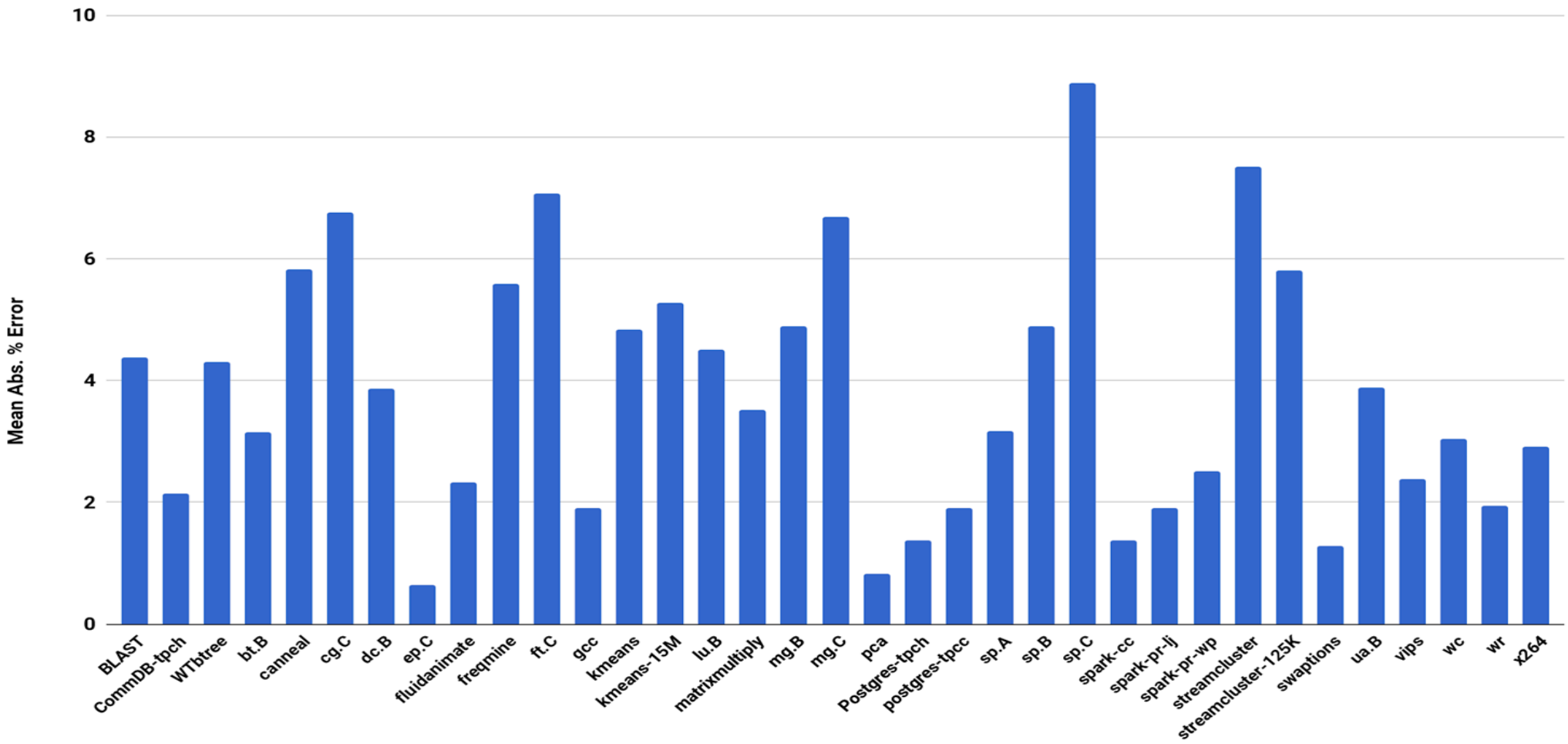
Performance Predictions – Spark Pagerank



Performance Predictions – is.D (sort)



Machine Learning – Prediction Accuracy



Conclusion

- Data centers: **3%** world's electricity (86% servers+cooling)
- Packing containers onto servers
  - Increase efficiency
  - Maintain performance goals
- **2-4×** better utilization in many cases!

- **Abstract machine model**
- **Performance prediction model**

Questions?

Workload Placement – Related Work

	Predicts Performance	Multiple Hardware Resources	Easily Adapted	Deployable Online
Our Solution	✓	✓	✓	✓
Pandia (EuroSys ‘17)	✓	✓	✗	✗
SMiTe (Micro ‘14)	✓	✓	✗	✓
Bubble-Flux (ISCA ‘13)	✓	✓	✗	✓
Asymsched (ATC ‘15)	✗	✗	✓	✓
DINO (ASPLOS ‘10)	✗	✗	✓	✓
Thread Clustering (EuroSys ‘07)	✗	✗	✓	✓