Unearthing inter-job dependencies for better scheduling

Andrew Chung[+]

Subru Krishnan*, Konstantinos Karanasos*,
Carlo Curino*, Greg Ganger[+]

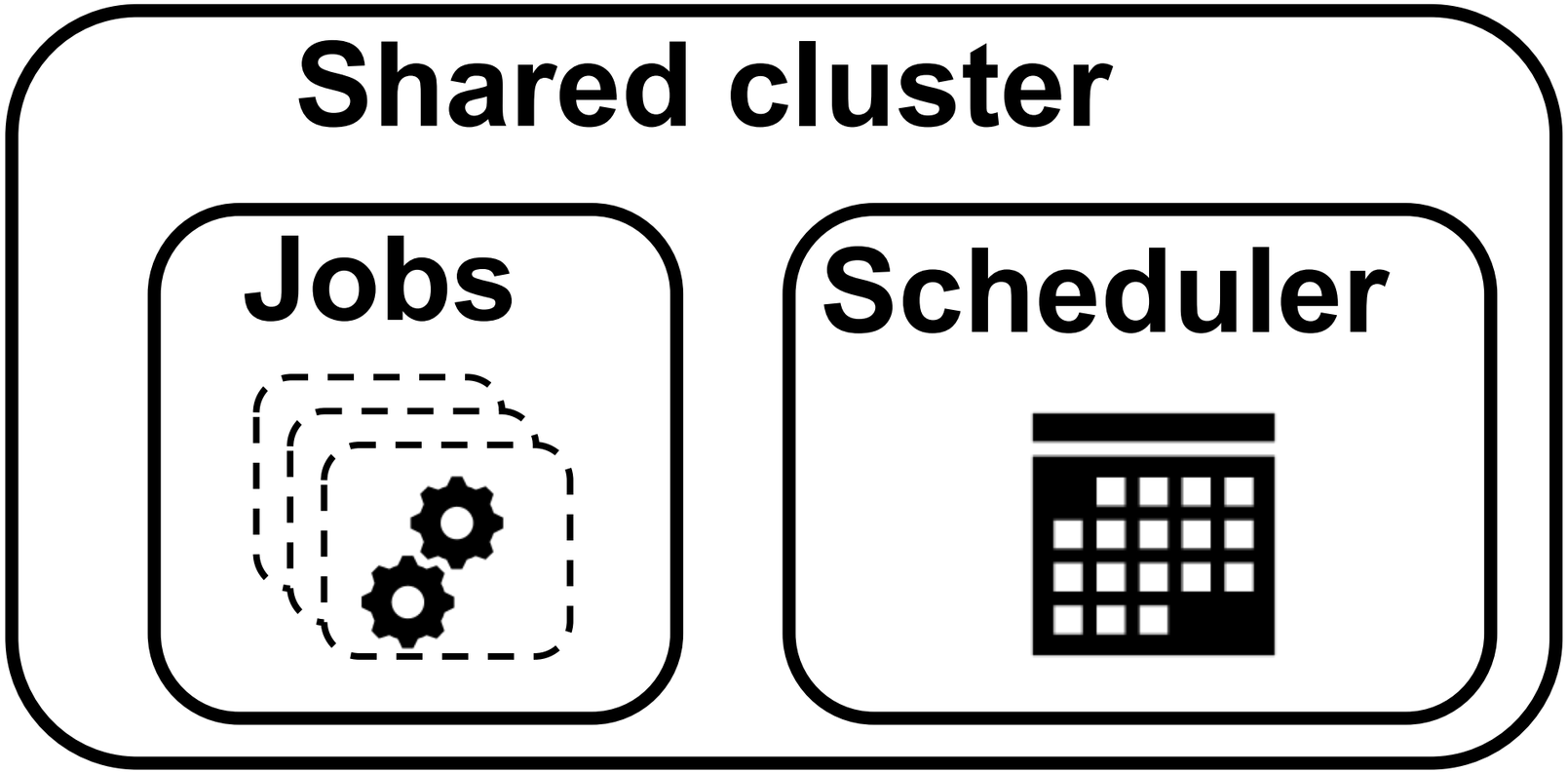
[+] Carnegie Mellon University

[*] Azure Data
Gray Systems Lab

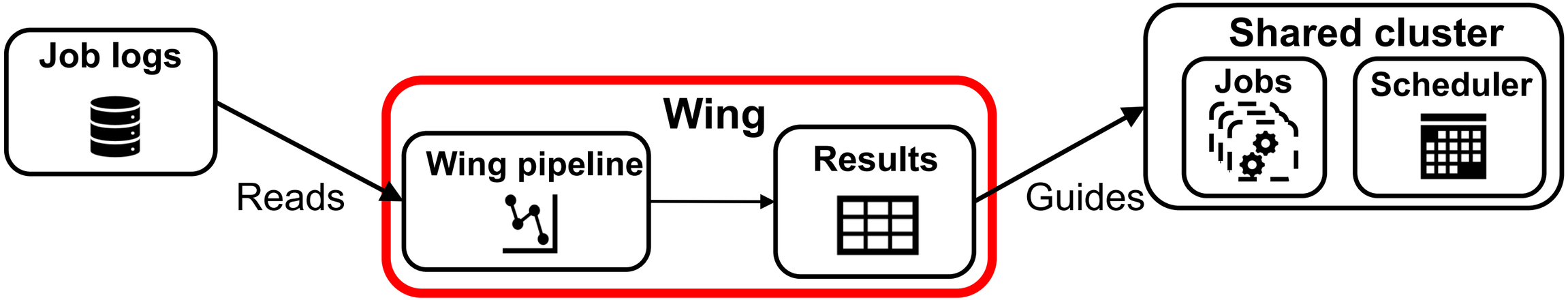
Carnegie Mellon
Parallel Data Laboratory

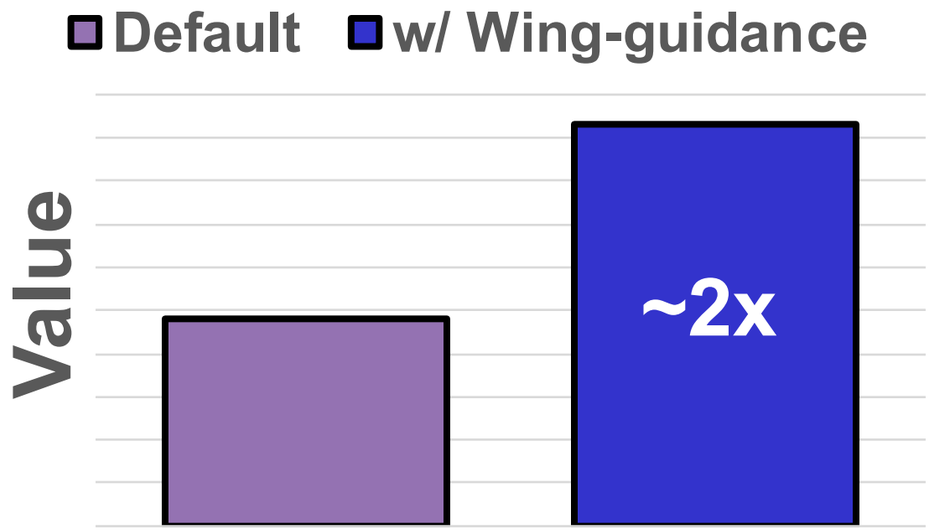
Microsoft

Wing summary



Wing summary





Understanding previously-ignored inter-job dependencies is important

Carnegie Mellon
Parallel Data Laboratory

Microsoft

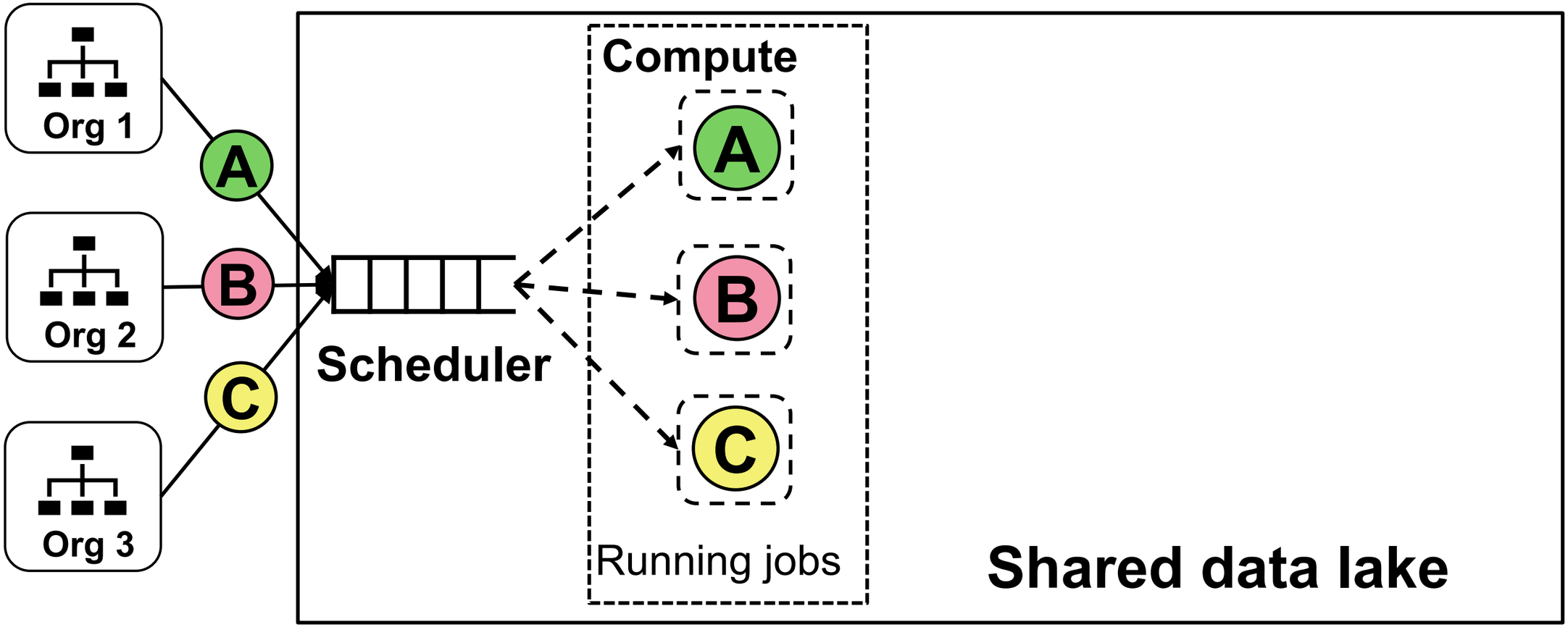
Outline

- Background: Clusters, scheduling, inter-job dependence
- Inter-job dependencies and the problems they bring
- The Wing inter-job dependency profiler
- Cluster resource scheduling with Wing
- Conclusion: Inter-job dependencies are important!

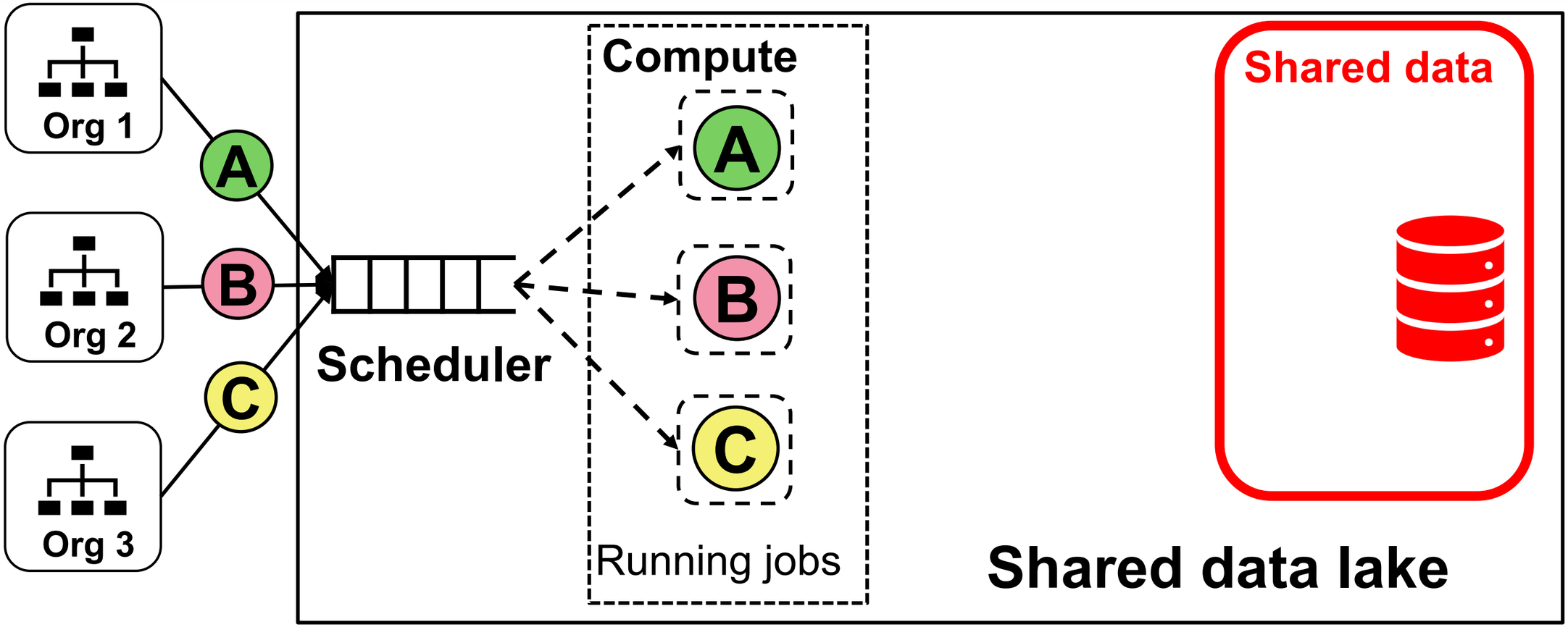
Outline

- Background: Clusters, scheduling, inter-job dependence
- Inter-job dependencies and the problems they bring
- The Wing inter-job dependency profiler
- Cluster resource scheduling with Wing
- Conclusion: Inter-job dependencies are important!

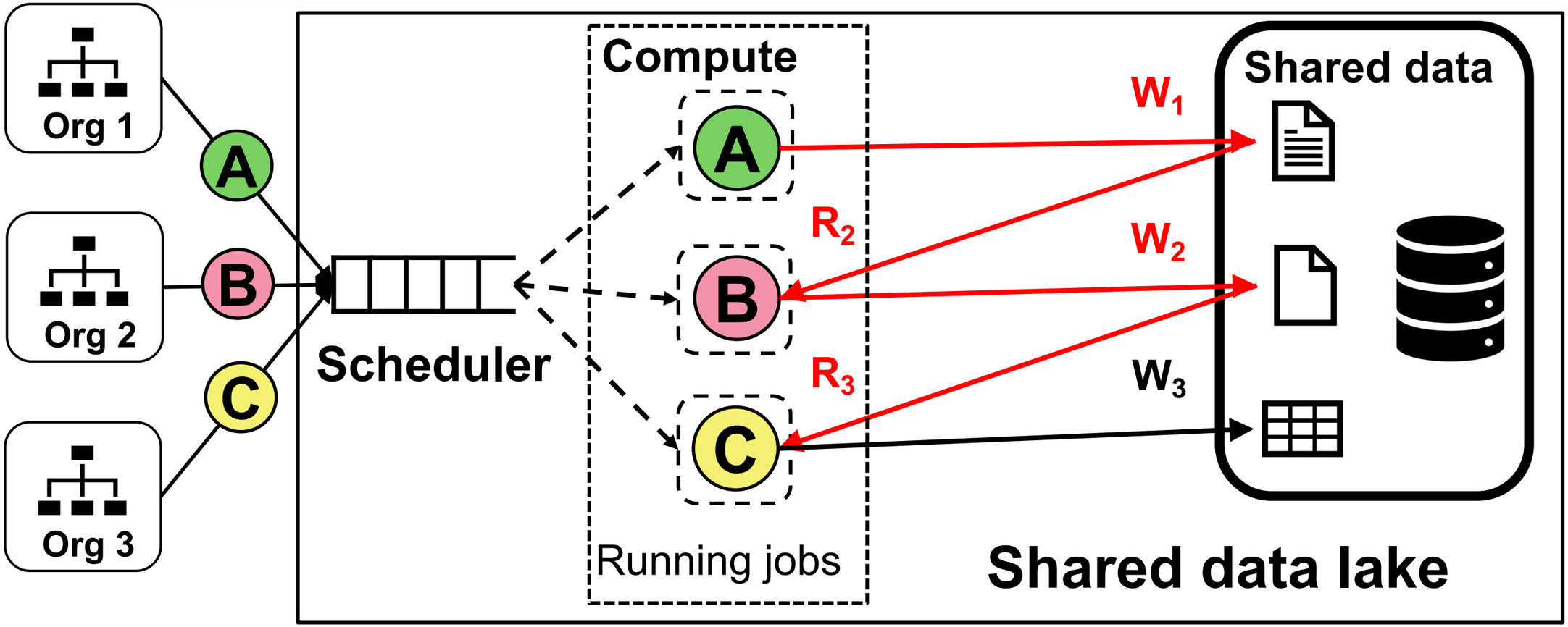
Shared data lakes



Shared data lakes



Shared data lakes

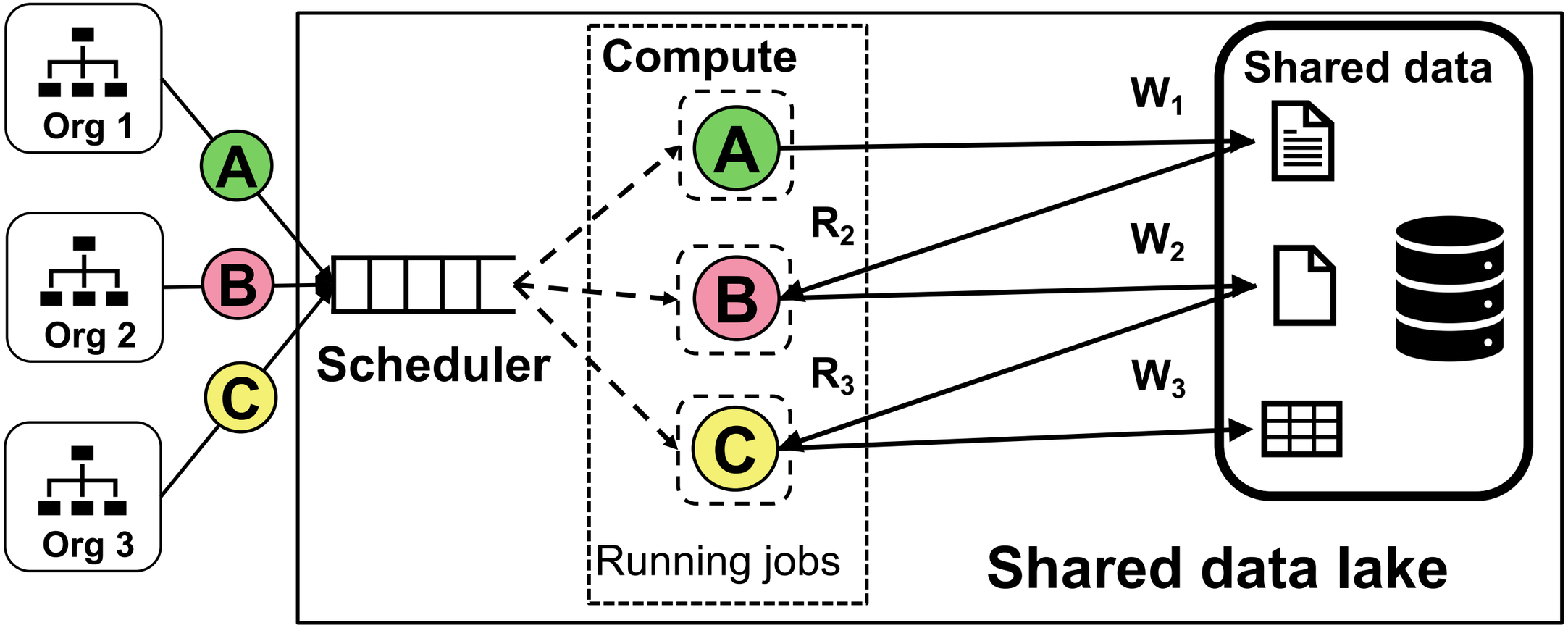


C depends on B depends on A

Outline

- Background: Clusters, scheduling, inter-job dependence
- Inter-job dependencies and the problems they bring
- The Wing inter-job dependency profiler
- Cluster resource scheduling with Wing
- Conclusion: Inter-job dependencies are important!

Shared data lakes

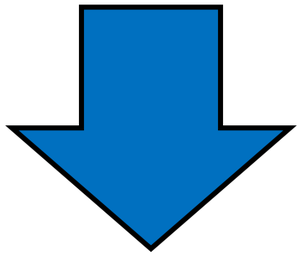


Problems when not considering deps

Inter-job dependencies pervade data lakes, but are ignored in resource management

Problems when not considering deps

Inter-job dependencies pervade data lakes, but are ignored in resource management



Missed deadlines, wasted resources, and untapped opportunities

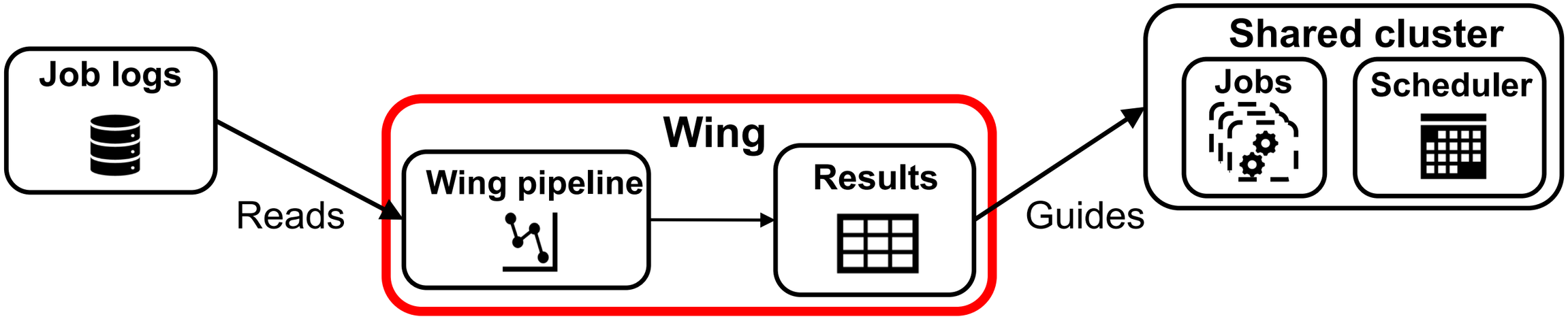
Towards addressing inter-job deps

Wing

Discovers + analyzes inter-job dependencies from data provenance

Scheduling with Wing guidance

Scheduling informed with historical inter-job dependencies



Outline

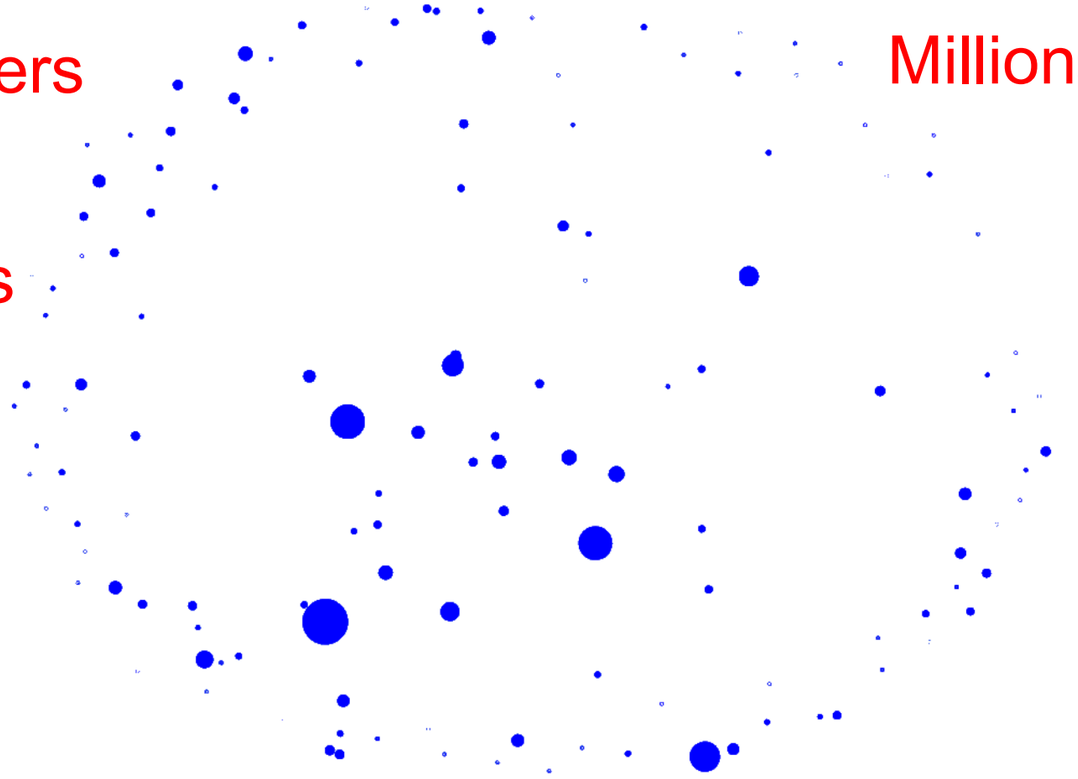
- Background: Clusters, scheduling, inter-job dependence
- Inter-job dependencies and the problems they bring
- The Wing inter-job dependency profiler
- Cluster resource scheduling with Wing
- Conclusion: Inter-job dependencies are important!

Data from a Cosmos cluster

40k+ daily jobs

50k+ servers

Millions of daily tasks



Hundreds of queues (organizations)

TBs of job + data prov logs daily

Data from a Cosmos cluster

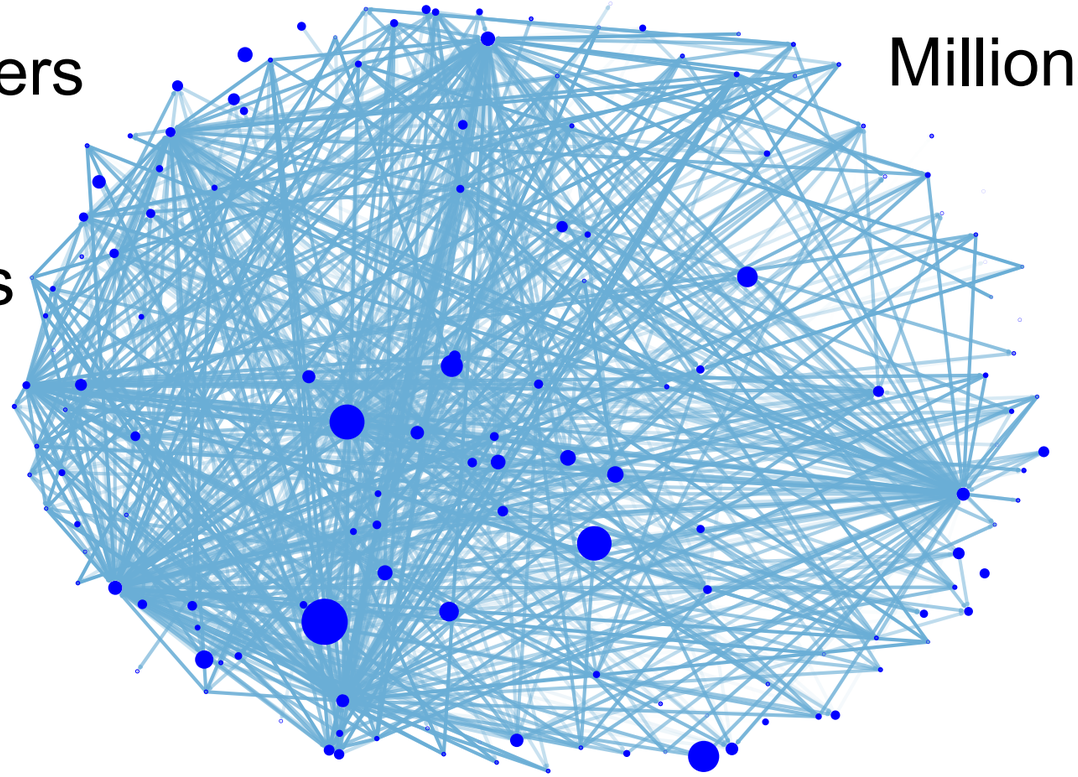
40k+ daily jobs

50k+ servers

Millions of daily tasks

Hundreds of queues (organizations)

TBs of job + data prov logs daily



160k+ daily inter-job dependencies

68% jobs recurring

95% of queues inter-dependent

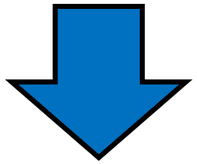
80% jobs depend on other jobs

Carnegie Mellon Parallel Data Laboratory

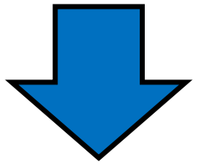
Microsoft

Scheduling and predicting the future

Better prediction of future jobs



Better planning for future jobs when scheduling



Better results

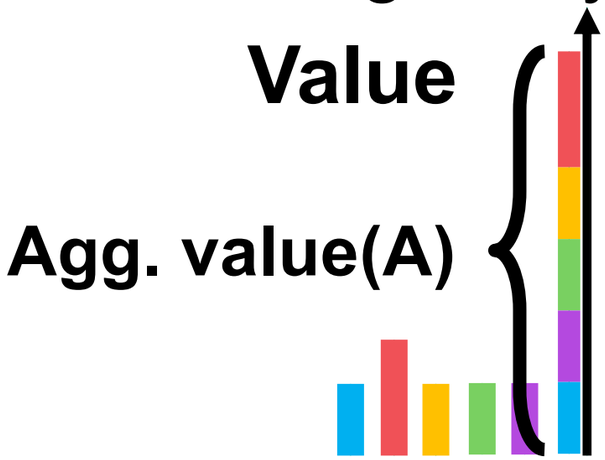
Recurring dependencies can help improve predictions

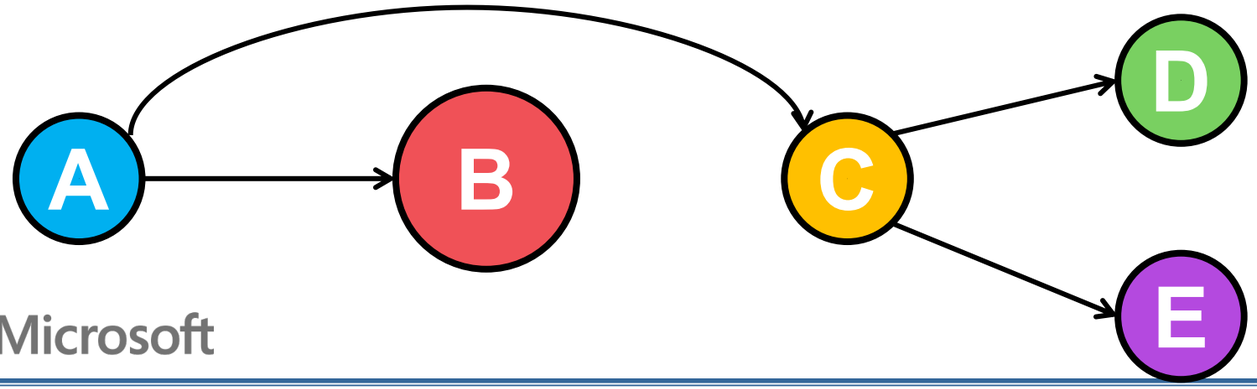
Job value & inter-job dependencies

- Failing/finishing jobs late can impact downstream jobs
- Wing analyzes the aggregate value (impact) of jobs

Job value & inter-job dependencies

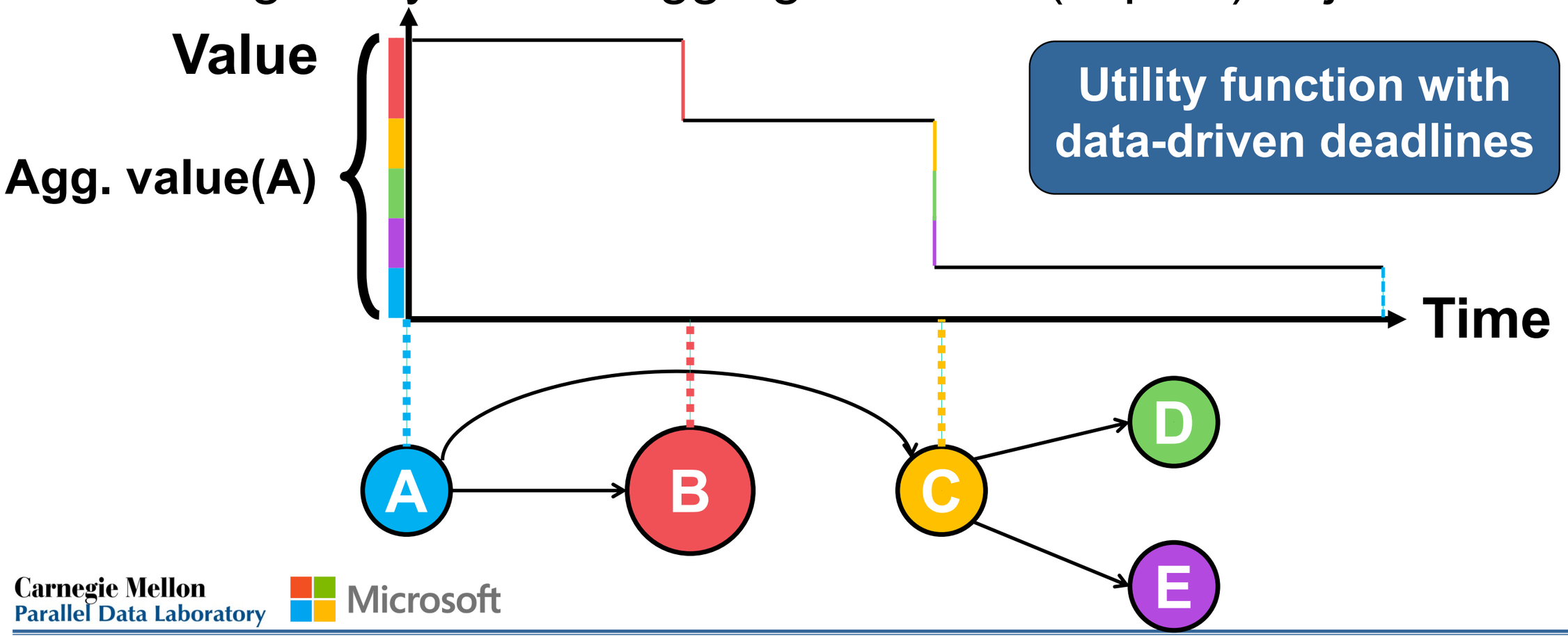
- Failing/finishing jobs late can impact downstream jobs
- Wing analyzes the aggregate value (impact) of jobs





Job value & inter-job dependencies

- Failing/finishing jobs late can impact downstream jobs
- Wing analyzes the aggregate value (impact) of jobs



Outline

- Background: Clusters, scheduling, inter-job dependence
- Inter-job dependencies and the problems they bring
- The Wing inter-job dependency profiler
- Cluster resource scheduling with Wing
- Conclusion: Inter-job dependencies are important!

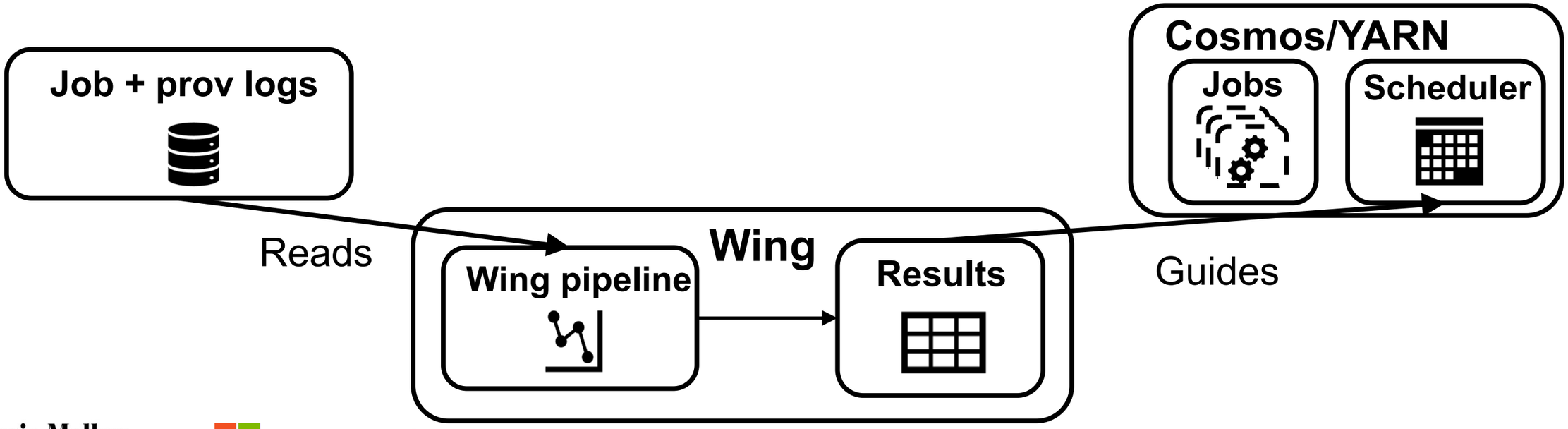
YARN, Cosmos, and value scheduling

- **YARN:** A resource management framework
  - Back-end of Cosmos resource management
  - Default scheduler: Resource decisions based on priorities
- **Value scheduling**
  - Complete jobs in a timely manner to achieve value
  - State-of-the-art: Considers each job independently

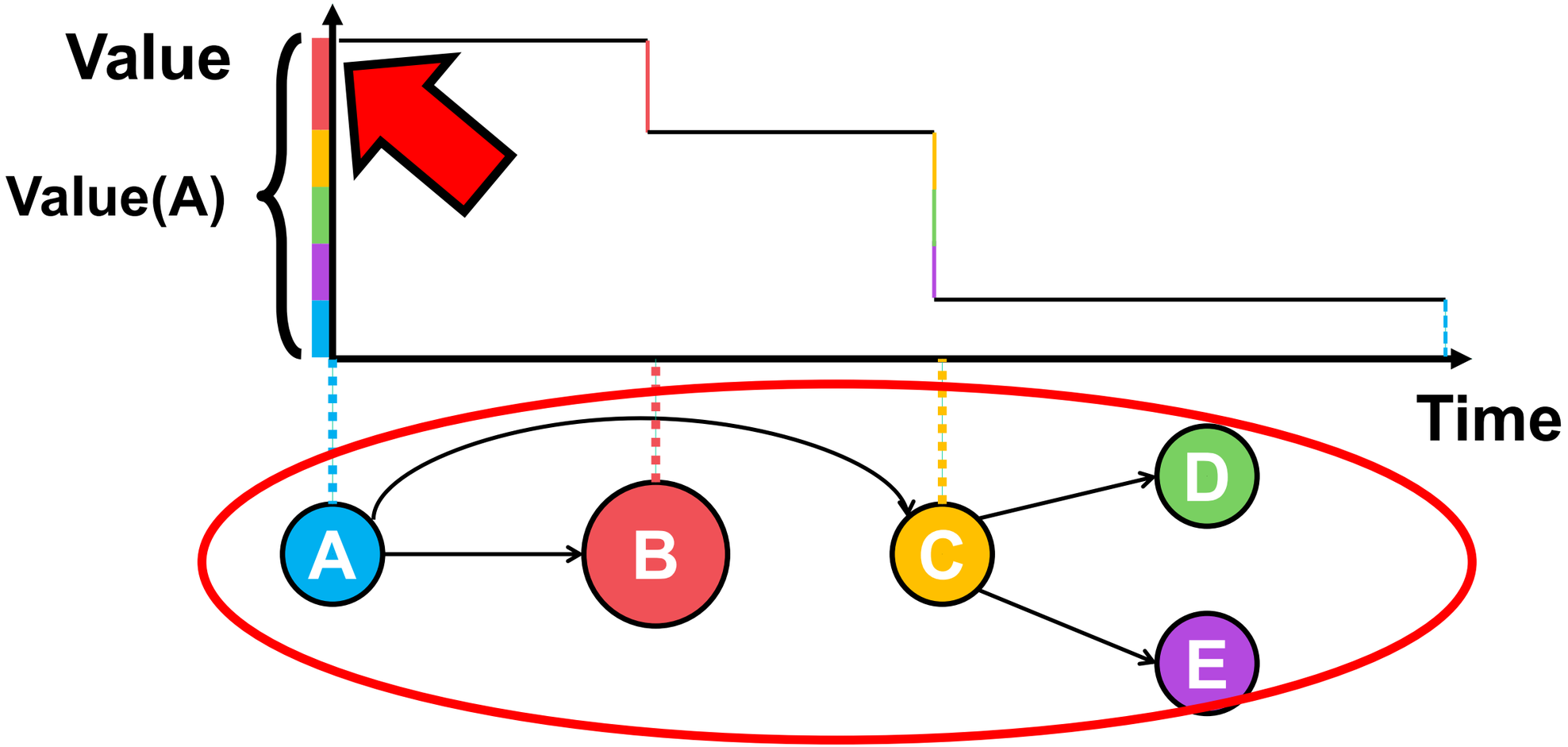
Inter-job dependencies to achieve more value

Wing-Agg: Wing-guided scheduling

- Exploit job + dependency recurrence to attain value
- **Wing-Agg**: YARN's prio-based sched + Wing-guidance
  - Prioritize recurring jobs with high aggregate value efficiency



Wing-Agg

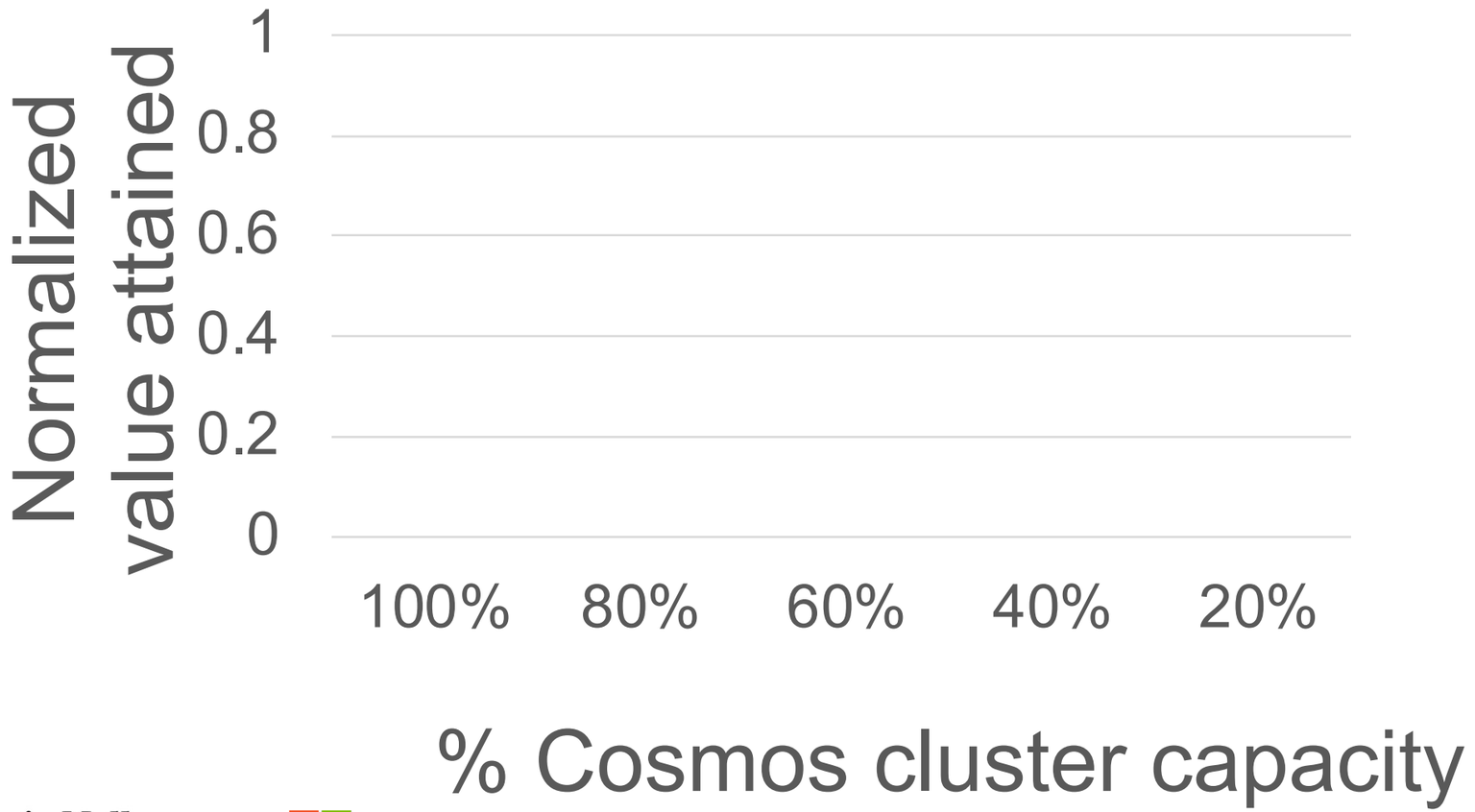


Experimental setup

- Trace-driven simulations on real cluster traces
  - Preserves inter-job dependencies and properties
- Goal: Attain more value from the same workload
  - Value metric: Total file output downloads attained
- Experiments at various cluster sizes (capacities)
  - To simulate resource-constrained clusters

Value-attainment

- **Wing-Agg**: Prio as historical **agg** value / **agg** compute

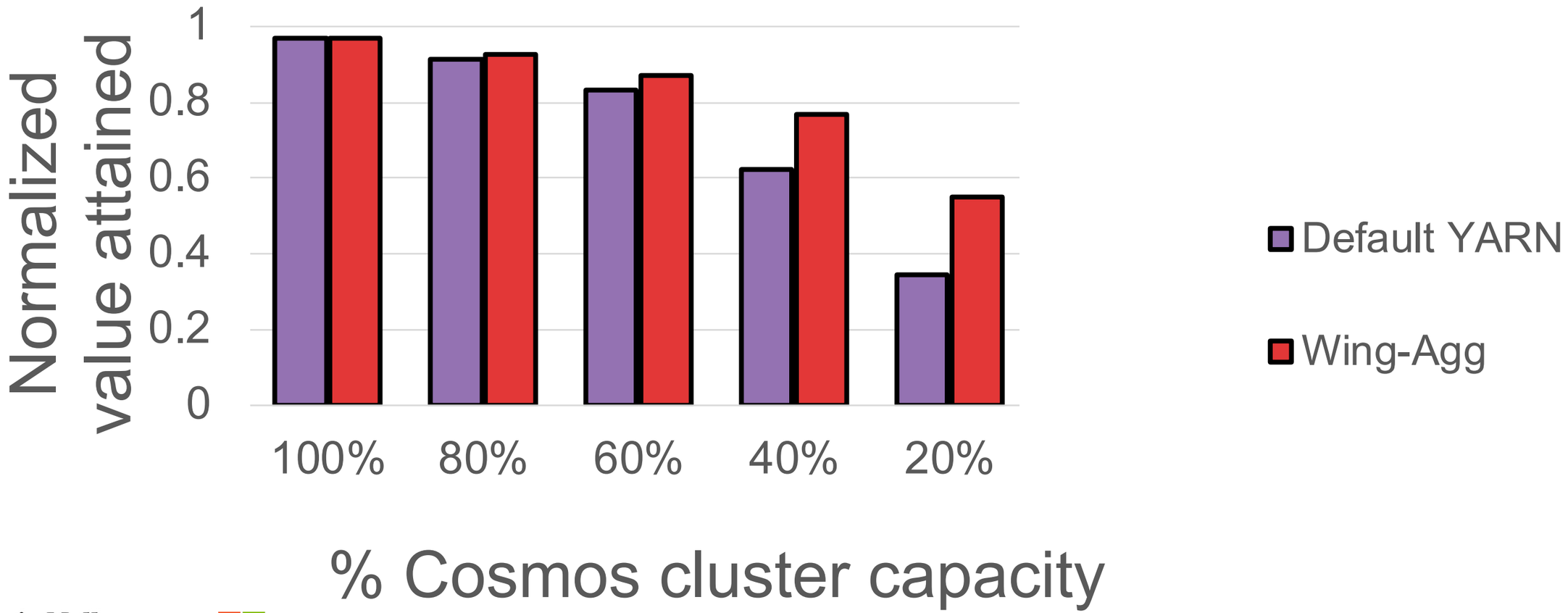


Carnegie Mellon
Parallel Data Laboratory

Microsoft

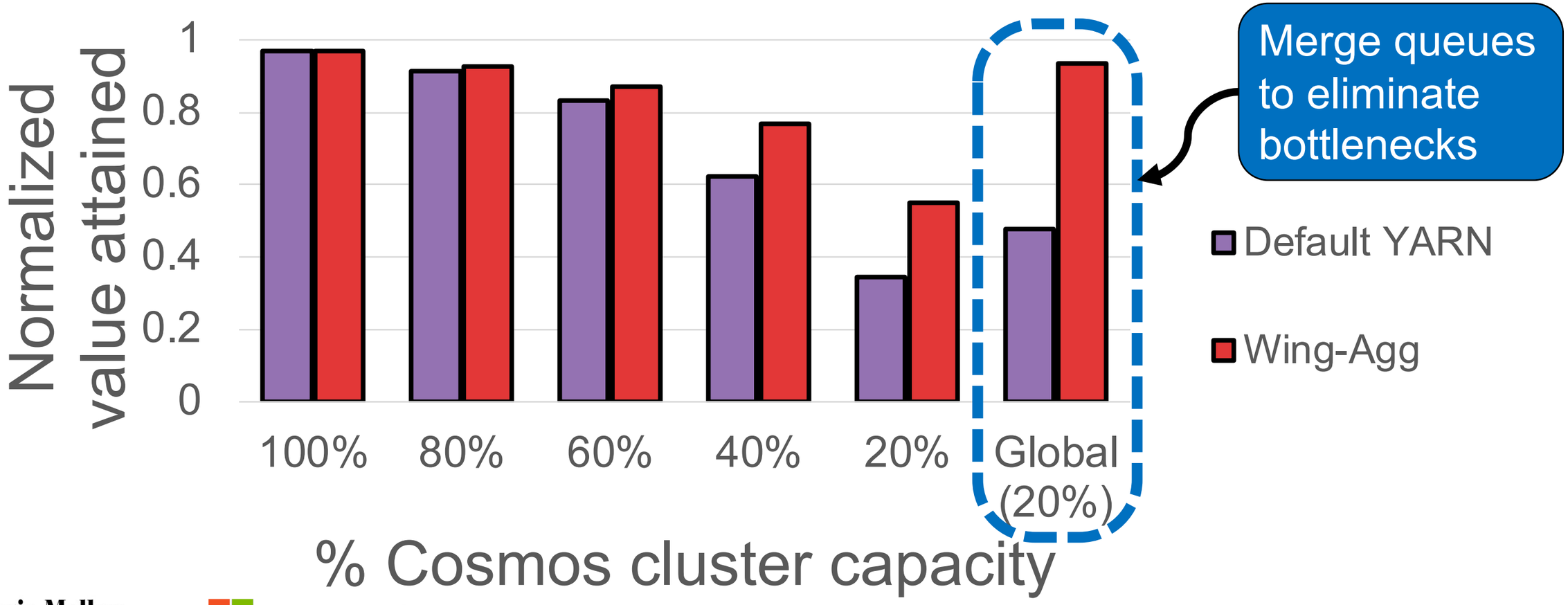
Value-attainment

- **Wing-Agg**: Prio as historical **agg** value / **agg** compute



Value-attainment

- **Wing-Agg**: Prio as historical **agg** value / **agg** compute



Takeaways

- Inter-job dependencies prevalent in real clusters
  - But, can be predictable with recurrence
- Inter-job dependencies need to be addressed
  - To ensure jobs meet their deadlines, reduce resource wastage, and improve value attained in shared clusters

Thank you!