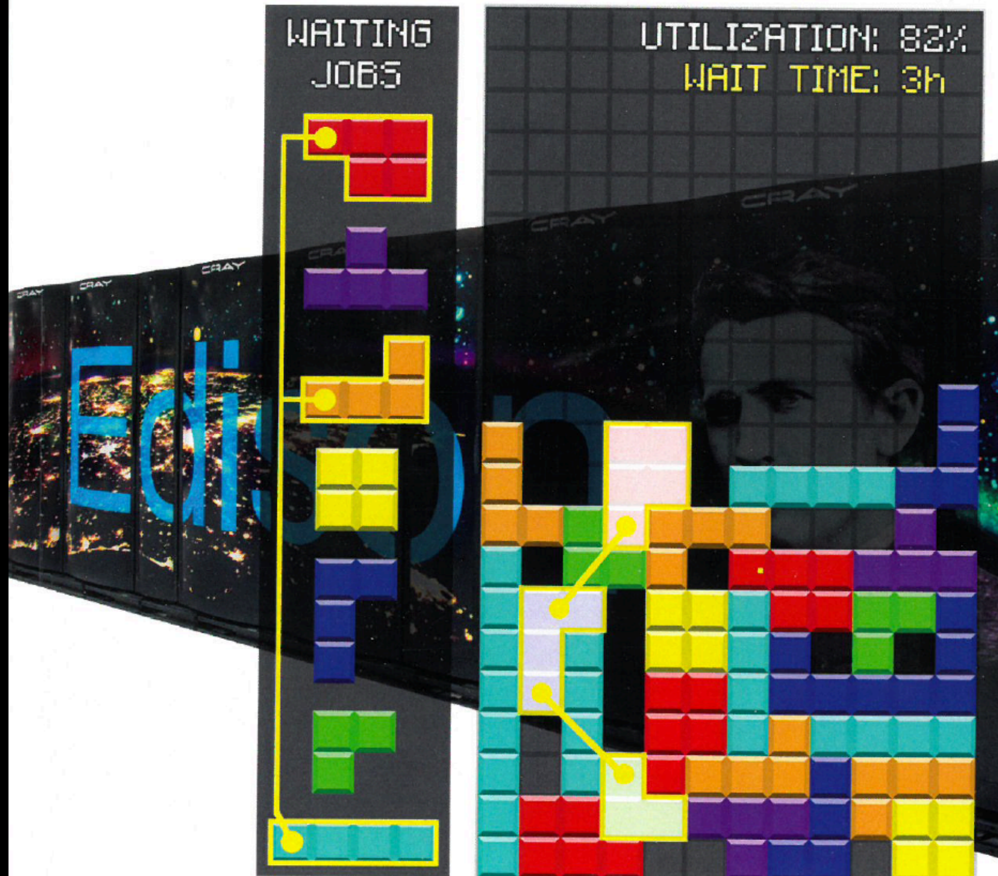


HPC Scheduling in a Brave New World

Gonzalo P. Rodrigo Álvarez

UMEÅ UNIVERSITY

PhD Defense
in Computing Science

Public defense of academic thesis
for the degree of
Doctor of Philosophy

PhD Defense

Respondent
MSc *Gonzalo P. Rodrigo Álvares*

Faculty Opponent
Professor Ewa Deelman,
University of Southern California,
CA, USA

PhD Defense

Examination Board
Docent Henrik Björklund, Department of Computing Science, Umeå University

Docent Emanuel Rubensson, Department of Information Technology, Uppsala University

Docent Oxana Smirnova, Department of Physics, Lund University

Chairman
Professor Erik Elmroth
Department of Computing Science,
Umeå University

PhD Defense Procedure

1. Presentation of the respondent, the faculty opponent, the examination board, and the chairman
2. The respondent – comments, addendum, errata
3. Presentations (the respondent and the opponent)
4. The respondent – possible addendum
5. The opponent and the respondent disputes
6. The examination board and the respondent disputes
7. Open floor – the audience and the respondent disputes
8. The defense is closed
9. The examination board convenes

HPC Scheduling in a Brave New World

Doctoral thesis defense

Gonzalo P. Rodrigo Álvarez
gonzalo@cs.umu.se
gprodrigoalvarez@lbl.gov

21st April 2017 – Umeå


UMEÅ UNIVERSITET
BERKELEY LAB

All papers and work

Paper I: **Towards Understanding HPC Users and Systems: A NERSC Case Study**. Submitted to JPDC (Journal of Parallel and Distributed Computing)

> **Paper Ia: Towards Understanding Job Heterogeneity in HPC: A NERSC Case Study**. 6th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID 2016)
>
> **Paper Ib: HPC System Lifetime Story: Workload Characterization and Evolutionary Analyses on NERSC Systems**. In Proceedings of the 24th International Symposium on High-Performance Parallel and Distributed Computing (HPDC 2015)

Paper II: Priority Operators for Fairshare Scheduling. 18th Workshop on Job Scheduling Strategies for Parallel Processing (JSSPP 2014) co-located with the IPDPS 2014 conference.

Paper III: A2L2: An Application Aware Flexible HPC Scheduling Model for Low-Latency Allocation. In Proceedings of the 8th International Workshop on Virtualization Technologies in Distributed Computing (VTDC 2015)

Paper IV: ScSF: A Scheduling Simulation Framework. 21th Workshop on Job Scheduling Strategies for Parallel Processing (JSSPP 2017) co-located with the IPDPS 2017 conference

Paper V: Enabling workflow aware scheduling on HPC systems. 26th International Symposium on High-Performance Parallel and Distributed Computing (HPDC 2017)

TR I: Establishing the equivalence between operators: theorem to establish a sufficient condition for two operators to produce the same ordering in a Fairshare prioritization system. January 2014

TR II: **Proof of compliance for the relative operator on the proportional distribution of unused share in an ordering fairshare system**. January 2014

TR III: **Theoretical analysis of a workflow aware scheduling algorithm**. March 2017

Open Source Project: WoAS (Workflow Aware scheduling) for Slurm
Open Source Project: ScSF, Scheduling Simulation Framework
Open Source Project: qdo, a many task workflow framework

All papers and work: In this presentation

Paper I: **Towards Understanding HPC Users and Systems: A NERSC Case Study**. Submitted to JPDC (Journal of Parallel and Distributed Computing)

- **Paper Ia: Towards Understanding Job Heterogeneity in HPC: A NERSC Case Study**. 6th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID 2016)
- **Paper Ib: HPC System Lifetime Story: Workload Characterization and Evolutionary Analyses on NERSC Systems**. In Proceedings of the 24th International Symposium on High-Performance Parallel and Distributed Computing (HPDC 2015)

Paper II: Priority Operators for Fairshare Scheduling. 18th Workshops on Job Scheduling Strategies for Parallel Processing (JSSPP 2014) co-located with the IPDPS 2014 conference.

Paper III: A2L2: An Application Aware Flexible HPC Scheduling Model for Low-Latency Allocation. In Proceedings of the 8th International Workshop on Virtualization Technologies in Distributed Computing (VTDC 2015)

Paper IV: ScSF: A Scheduling Simulation Framework. 21th Workshops on Job Scheduling Strategies for Parallel Processing (JSSPP 2017) co-located with the IPDPS 2017 conference

Paper V: Enabling workflow aware scheduling on HPC systems. 26th International Symposium on High-Performance Parallel and Distributed Computing (HPDC 2017)

TR I: Establishing the equivalence between operators: theorem to establish a sufficient condition for two operators to produce the same ordering in a Fairshare prioritization system. January 2014

TR II: **Proof of compliance for the relative operator on the proportional distribution of unused share in an ordering fairshare system**. January 2014

TR III: **Theoretical analysis of a workflow aware scheduling algorithm**. March 2017

Open Source Project: WoAS (Workflow Aware scheduling) for Slurm
Open Source Project: ScSF, Scheduling Simulation Framework
Open Source Project: qdo, a many task workflow framework

All papers and work: Also relevant

Paper I: Towards Understanding HPC Users and Systems: A NERSC Case Study. Submitted to JPDC (Journal of Parallel and Distributed Computing)

Paper Ia: Towards Understanding Job Heterogeneity in HPC: A NERSC Case Study. 6th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID 2016)

Paper Ib: HPC System Lifetime Story: Workload Characterization and Evolutionary Analyses on NERSC Systems. In Proceedings of the 24th International Symposium on High-Performance Parallel and Distributed Computing (HPDC 2015)

Paper II: Priority Operators for Fairshare Scheduling. 18th Workshops on Job Scheduling Strategies for Parallel Processing (JSSPP 2014) co-located with the IPDPS 2014 conference.

Paper III: A2L2: An Application Aware Flexible HPC Scheduling Model for Low-Latency Allocation. In Proceedings of the 8th International Workshop on Virtualization Technologies in Distributed Computing (VTDC 2015)

Paper IV: ScSF: A Scheduling Simulation Framework. 21th Workshops on Job Scheduling Strategies for Parallel Processing (JSSPP 2017) co-located with the IPDPS 2017 conference

Paper V: Enabling workflow aware scheduling on HPC systems. 26th International Symposium on High-Performance Parallel and Distributed Computing (HPDC 2017)

TR I: Establishing the equivalence between operators: theorem to establish a sufficient condition for two operators to produce the same ordering in a Fairshare prioritization system. January 2014

TR II: **Proof of compliance for the relative operator on the proportional distribution of unused share in an ordering fairshare system**. January 2014

TR III: **Theoretical analysis of a workflow aware scheduling algorithm**. March 2017

Open Source Project: WoAS (Workflow Aware scheduling) for Slurm
Open Source Project: ScSF, Scheduling Simulation Framework
Open Source Project: qdo, a many task workflow framework

All papers and work: Corrections

Paper I: Towards Understanding HPC Users and Systems: A NERSC Case Study. Submitted to JPDC (Journal of Parallel and Distributed Computing)

- **Paper Ia: Towards Understanding Job Heterogeneity in HPC: A NERSC Case Study**. 6th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID 2016)
- **Paper Ib: HPC System Lifetime Story: Workload Characterization and Evolutionary Analyses on NERSC Systems**. In Proceedings of the 24th International Symposium on High-Performance Parallel and Distributed Computing (HPDC 2015)

Paper II: Priority Operators for Fairshare Scheduling. 18th Workshops on Job Scheduling Strategies for Parallel Processing (JSSPP 2014) co-located with the IPDPS 2014 conference.

Paper III: A2L2: An Application Aware Flexible HPC Scheduling Model for Low-Latency Allocation. In Proceedings of the 8th International Workshop on Virtualization Technologies in Distributed Computing (VTDC 2015)

Paper IV: ScSF: A Scheduling Simulation Framework. 21th Workshops on Job Scheduling Strategies for Parallel Processing (JSSPP 2017) co-located with the IPDPS 2017 conference

Paper V: Enabling workflow aware scheduling on HPC systems. 26th International Symposium on High-Performance Parallel and Distributed Computing **(HPDC 2017)**

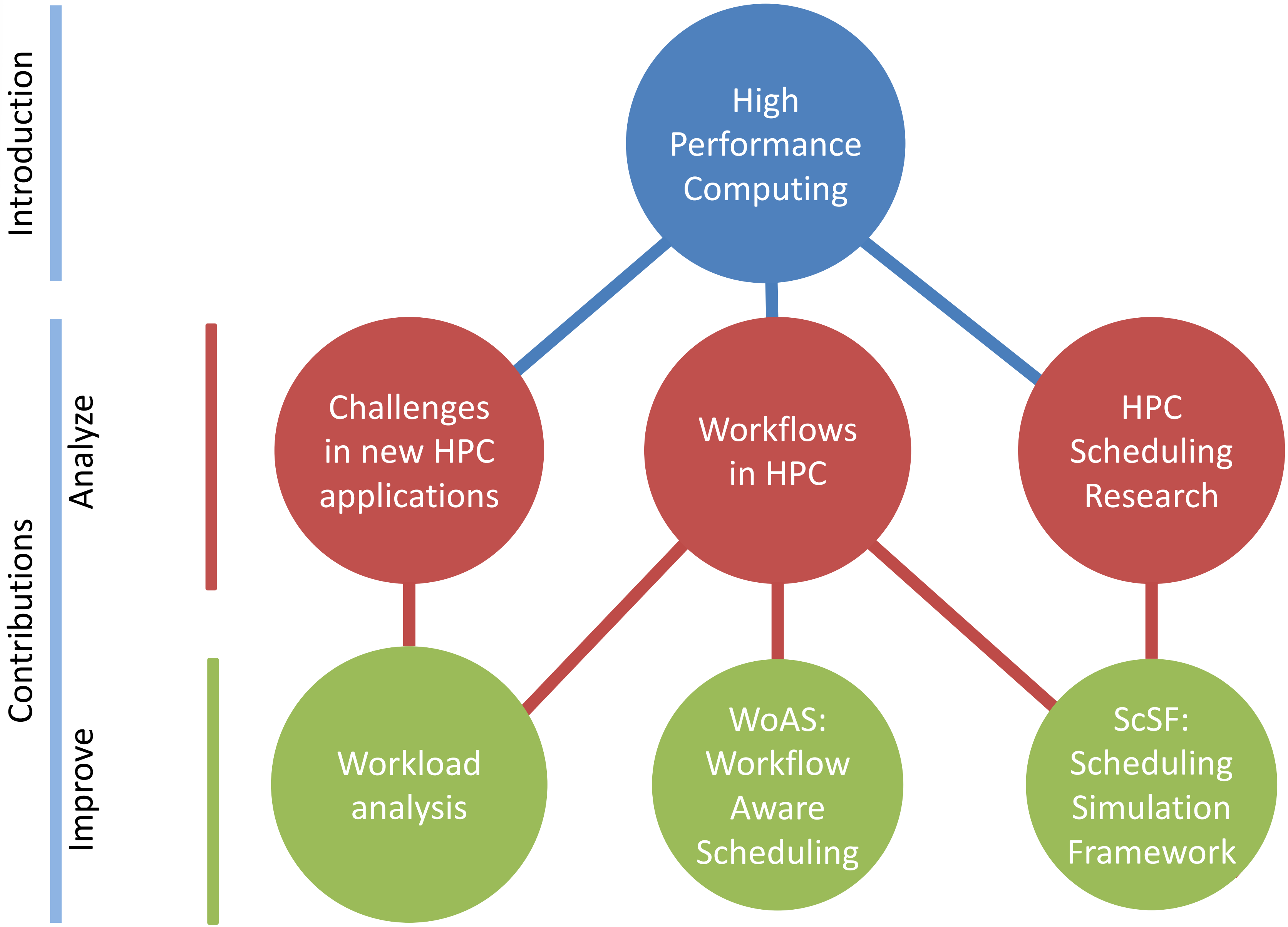
TR I: Establishing the equivalence between operators: theorem to establish a sufficient condition for two operators to produce the same ordering in a Fairshare prioritization system. January 2014

TR II: Proof of compliance for the relative operator on the proportional distribution of unused share in an ordering fairshare system. January 2014

TR III: Theoretical analysis of a workflow aware scheduling algorithm. March 2017

Open Source Project: WoAS (Workflow Aware scheduling) for Slurm
Open Source Project: ScSF, Scheduling Simulation Framework
Open Source Project: qdo, a many task workflow framework

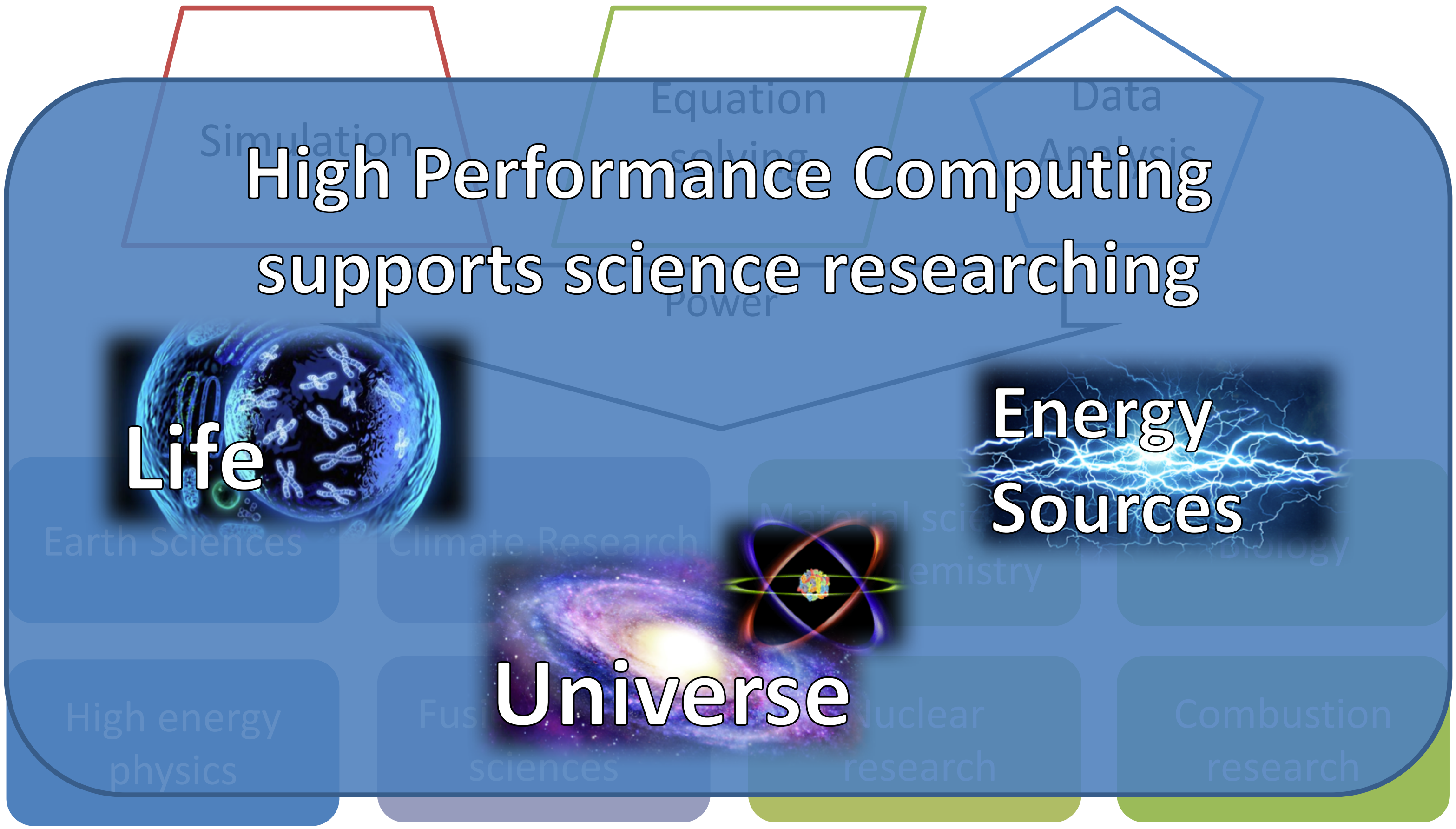
Outline


Introduction
Contributions
Analyze
Improve
High Performance Computing
Challenges in new HPC applications
Workflows in HPC
HPC Scheduling Research
Workload analysis
WoAS: Workflow Aware Scheduling
ScSF: Scheduling Simulation Framework

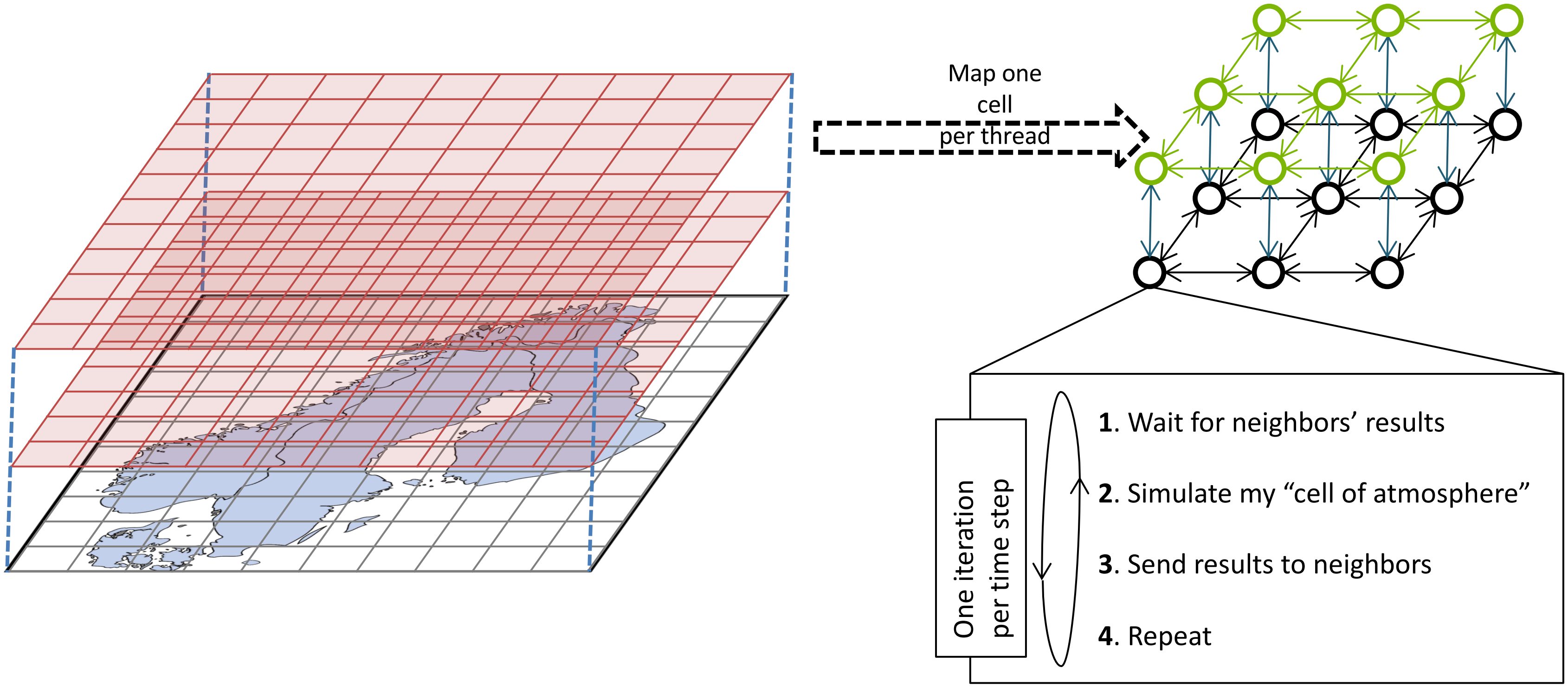
High Performance Computing

PRIMER

High Performance Computing: Uses



HPC Example application: weather forecast



Extreme parallelism

Fast processing

Large memory and I/O concurrency

Homogeneity

Low predictable latency networks

HPC economics 101: Scientific work

Demand Excess

More science than compute hours

Sync. with human research

↓

Get more work done… On time

"Scientific" finances

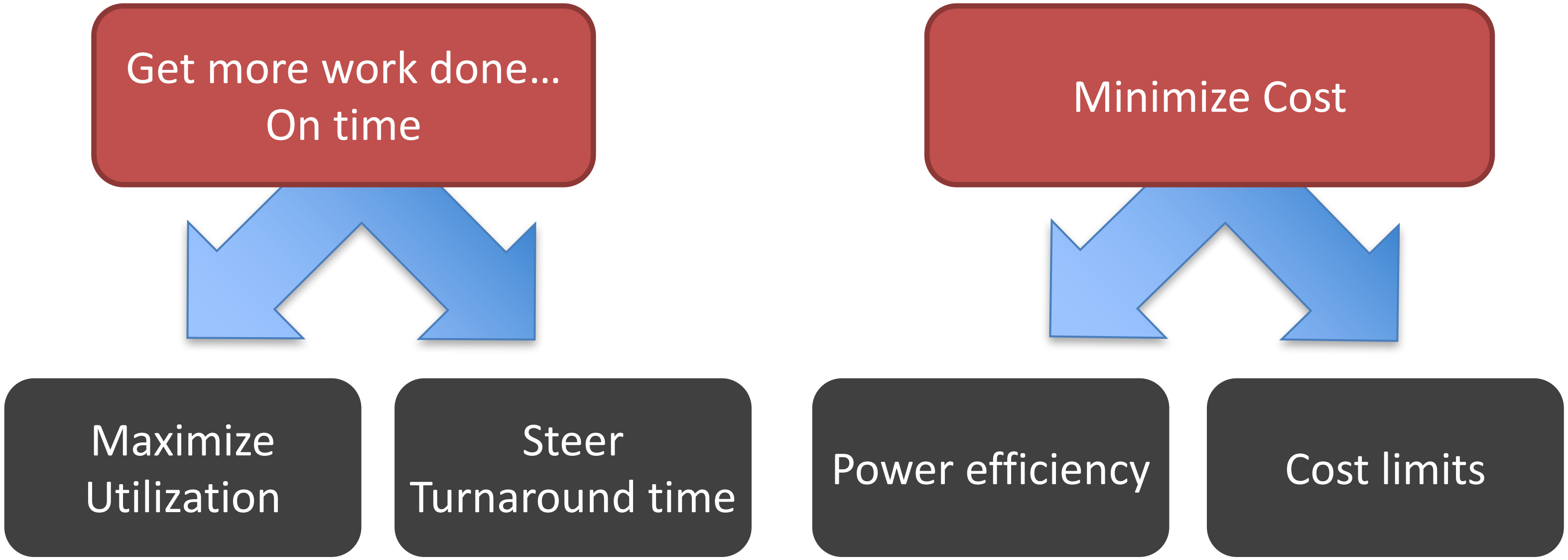
HPC System revenue independent work done

Fixed operational budget

↓

Minimize Cost

HPC economics 101: Scientific work



HPC Systems: Final Requirement

Maximize Utilization	Steer Turnaround time	Power efficiency	Cost limits
Extreme parallelism	Fast processing	Large memory and I/O concurrency	Homogeneity

Low latency, synchronous network

… a computing system should be considered high performance if it supports the execution of large-scale, performance-oriented applications, at the smallest possible cost, with the shortest possible runtime, within some time constraint…

HPC Systems: Examples

Edison @ NERSC

HPC Systems: Edison in detail

HPC Systems: Edison in detail

First Cray XC30: #1

Peak: **2.57 Petaflops/s**.

357 Terabytes of memory

133.824 cores (x2 with HT)

5586 compute nodes (24 cores)

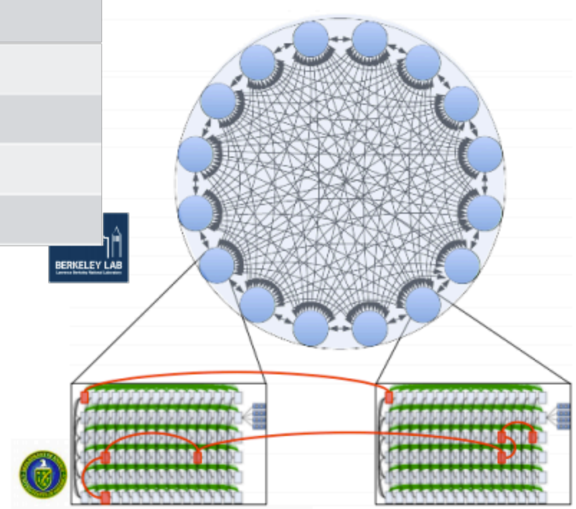
File Systems: up **to 700GB/s**

Power: **3,747.07 kW**

Custom interconnect: **Aries**

MPI	Latency (us)	Bandwidth (GB/s)
Socket	0.3	
Node	0.7	
Blade	1.3	14.9
Rank-1	1.5	15.4
Rank-2	1.5	15.4
Rank-3	2.2	15.3
Farthest	2.3	15.3

Aries Topology



Top500: #18 (2014)-> #60 (2016)

On a normal operation:
more than 500 apps run at the same time

Characterization of the Cray Aries Network – NERSC - https://www.nersc.gov/assets/pubs_presos/NUG2014Aries.pdf

HPC Systems: Edison in detail

NERSC

BERKELEY LAB

First Cray XC30: #1

Peak: **2.57 Petaflops/s.**

357 Terabytes of memory

133.824 cores (x2 with HT)

5586 compute nodes (24 cores)

File Systems: up **to 700GB/s**

Power: **3,747.07 kW**

Custom interconnect: **Aries**

MPI	Latency (us)	Bandwidth (GB/s)
Socket	0.3	
Node	0.7	
Blade	1.3	14.9
Rank-1	1.5	15.4
Rank-2		
Rank-3		
Farthes		

Low latency, synchronous network

Aries Topology

Power efficiency

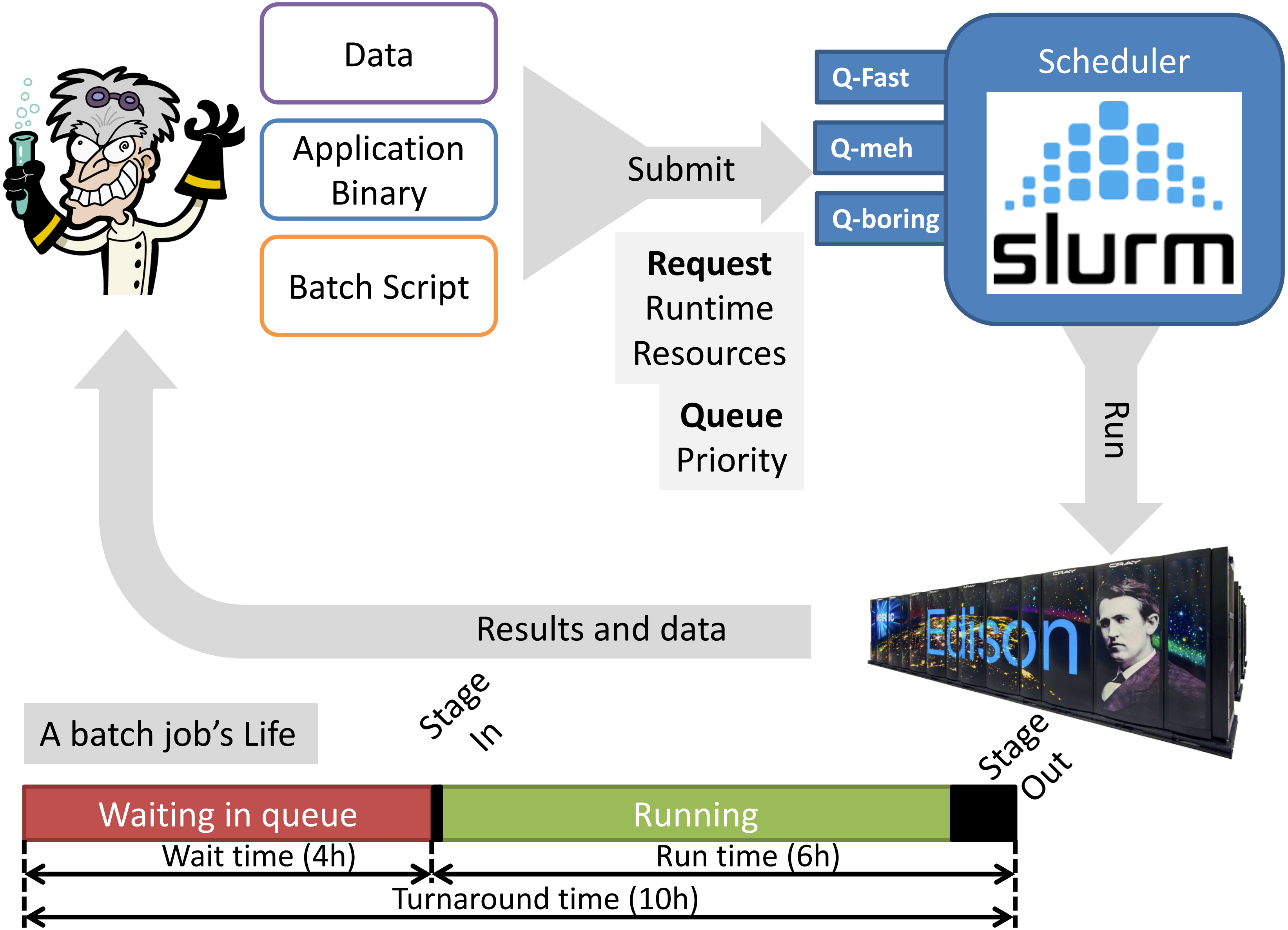
Extreme parallelism

Fast processing

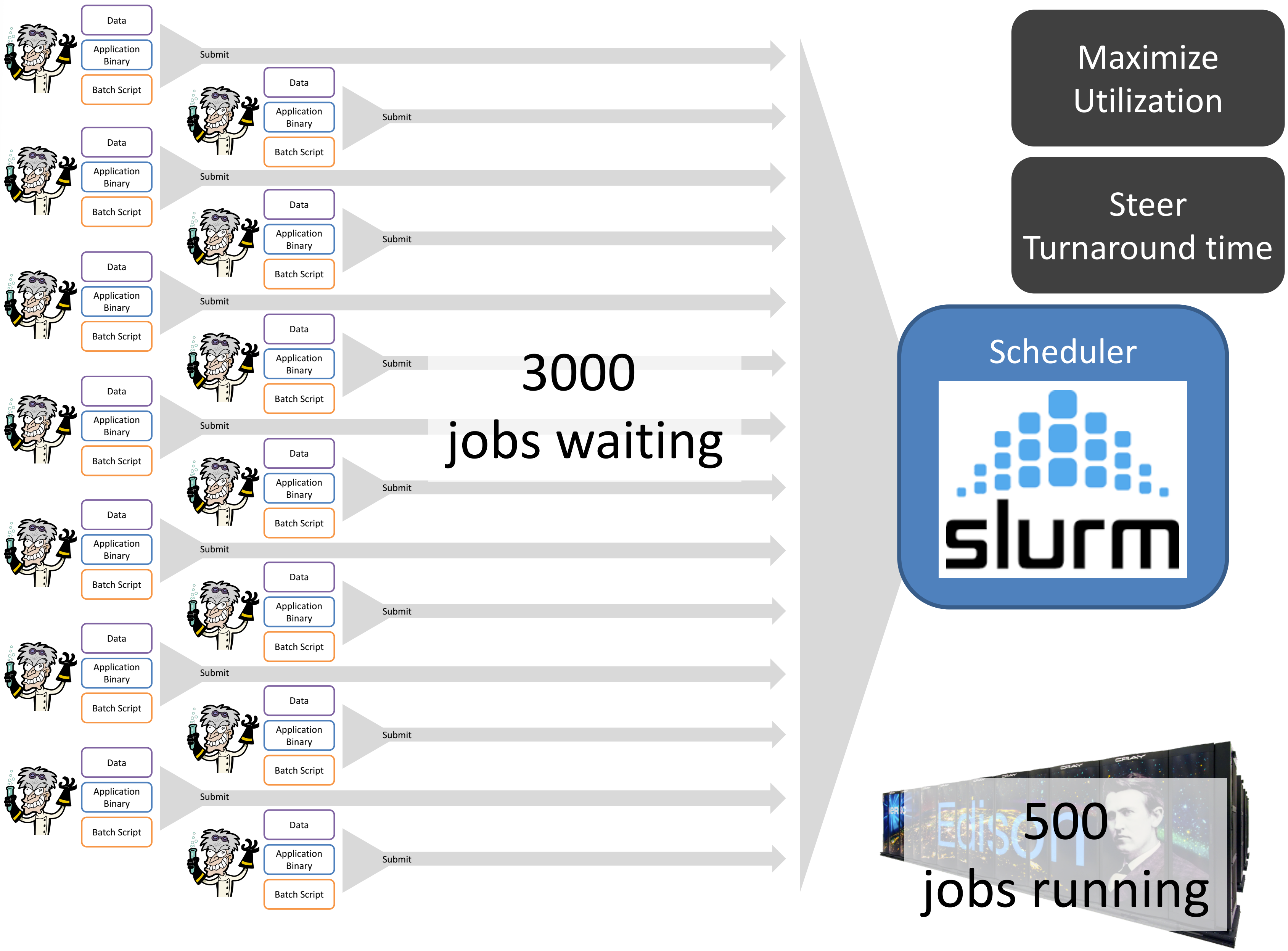
Large memory and I/O concurrency

Homogeneity

Using an HPC systems: User perspective



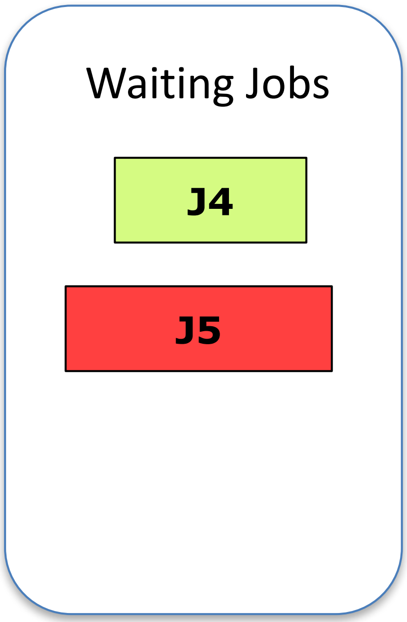
HPC Jobs and scheduling

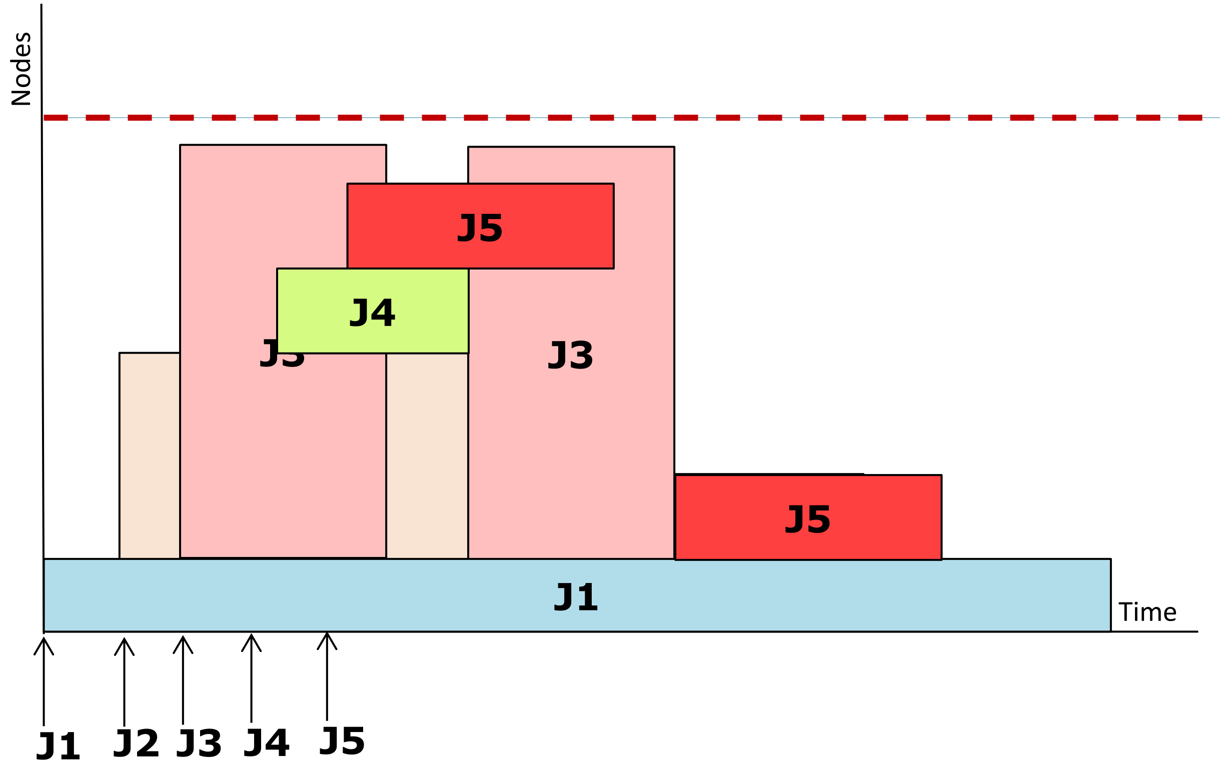


HPC Scheduling

First Come First Serve (FCFS): Run jobs in arrival order

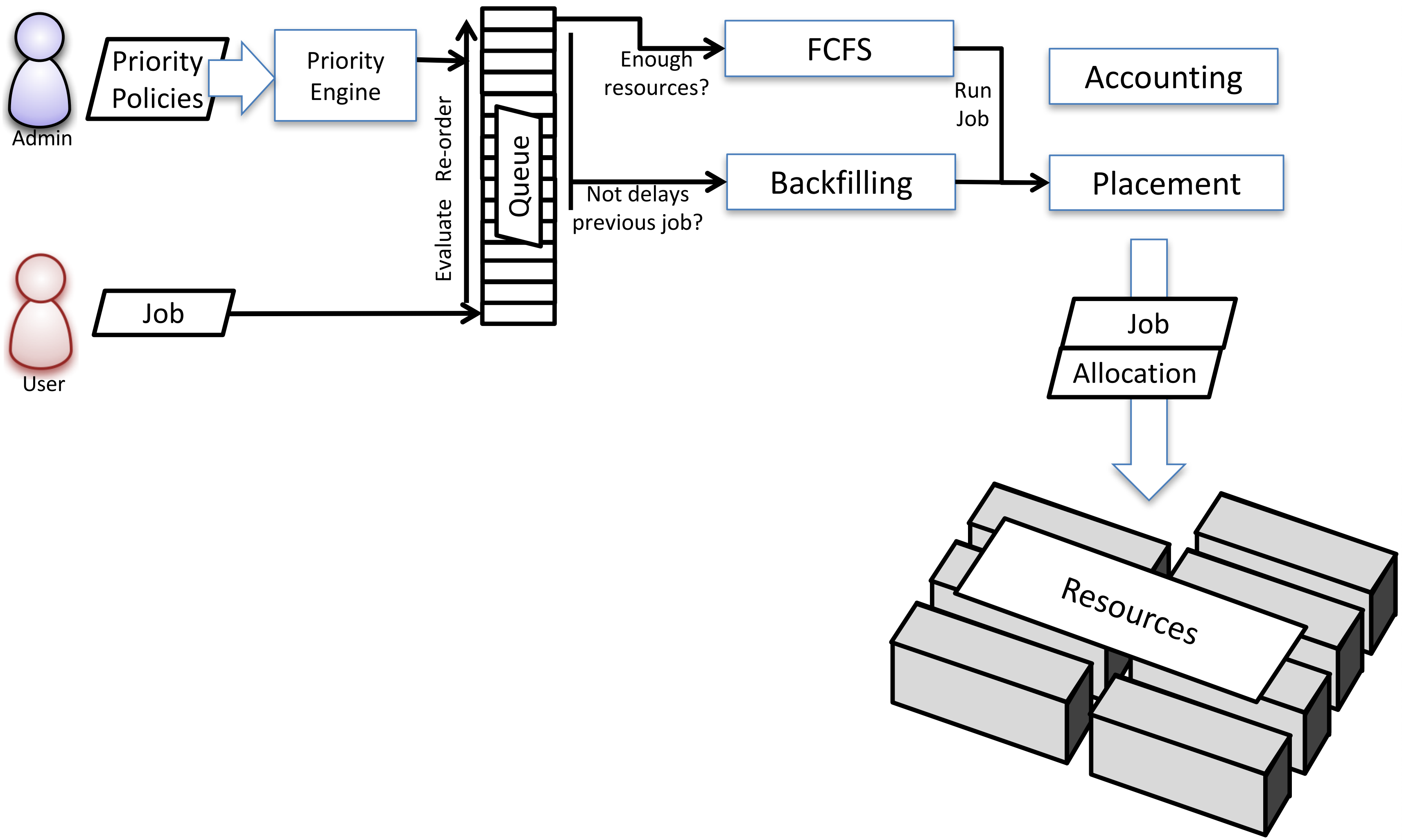
Backfill: Run jobs that will not delay previous ones



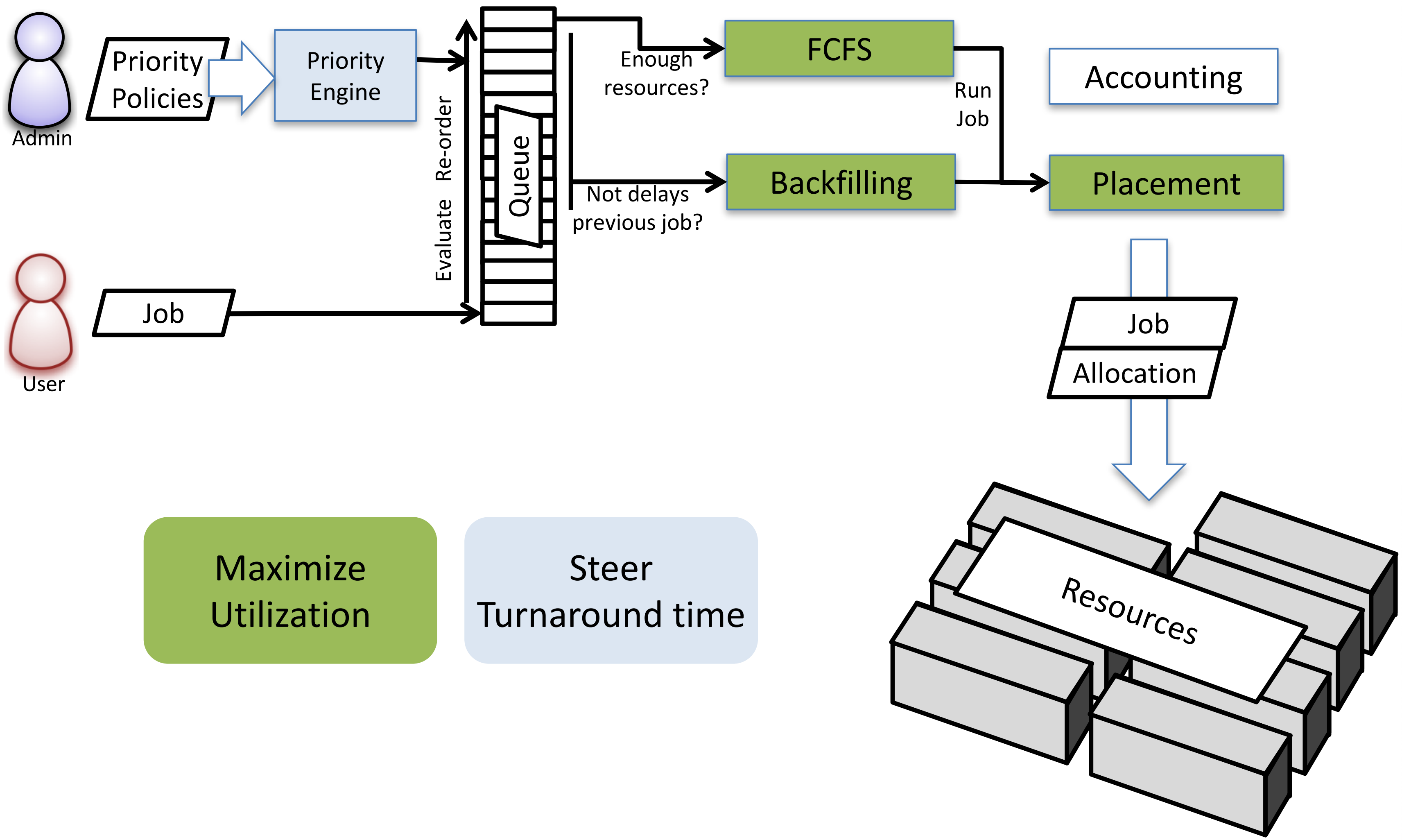


Maximize Utilization

Generic HPC Scheduler



Generic HPC Scheduler

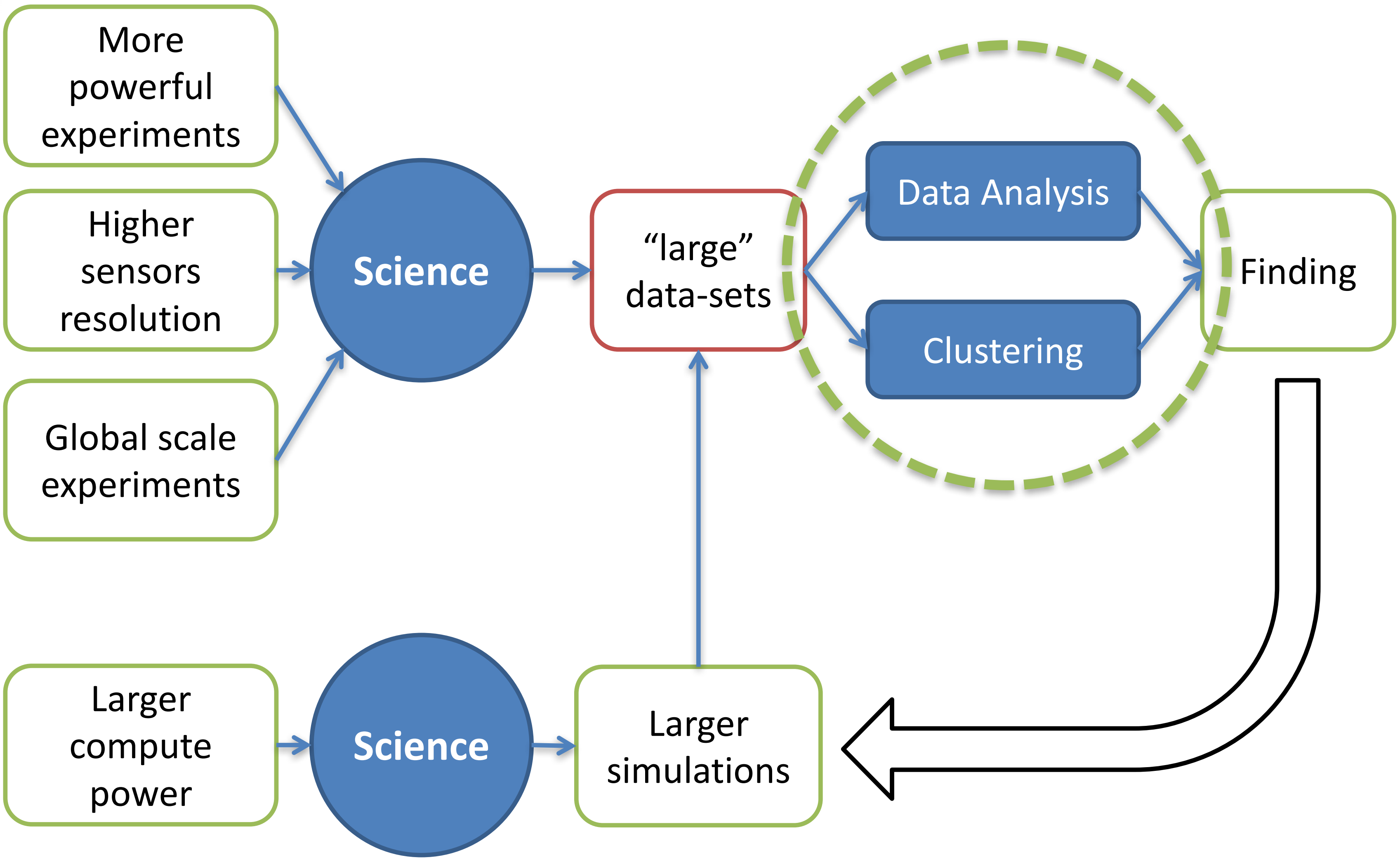


Challenges in High Performance Computing

NEW HPC APPLICATIONS

(AND THEIR BATCH JOBS)

Welcome to the 4th Paradigm of Science: Big Data



Tansley, Stewart, and Kristin Michele Tolle, eds. The fourth paradigm: data-intensive scientific discovery. Vol. 1. Redmond, WA: Microsoft research, 2009.

Paper III: A2L2: An Application Aware Flexible HPC Scheduling Model for Low-Latency Allocation. In Proceedings of the 8th International Workshop on Virtualization Technologies in Distributed Computing (VTDC 2015)

Data more important in HPC workloads

Data intensive applications

Application Diversity

High-throughput applications

Bad	Good
I/O and MEM more impact on performance	Might allow dynamic resource allocation.
Higher network load	Existence data intensive frameworks
Application Diversity: Different performance model	
Increased resource heterogeneity	
Workflows: data movement and coordination	

Paper III: A2L2: An Application Aware Flexible HPC Scheduling Model for Low-Latency Allocation. In Proceedings of the 8th International Workshop on Virtualization Technologies in Distributed Computing (VTDC 2015)

Applications are changing, and batch jobs?

Is job geometry changing? Does it matter?

↓

Analysis on workload evolution

↓

Detailed analysis of one year

↓

Job heterogeneity Analysis

↓

Performance Analysis

Paper I: Towards Understanding HPC Users and Systems: A NERSC Case Study. Submitted to JPDC (Journal of Parallel and Distributed Computing)

Source Dataset: NERSC Systems

Supercomputers

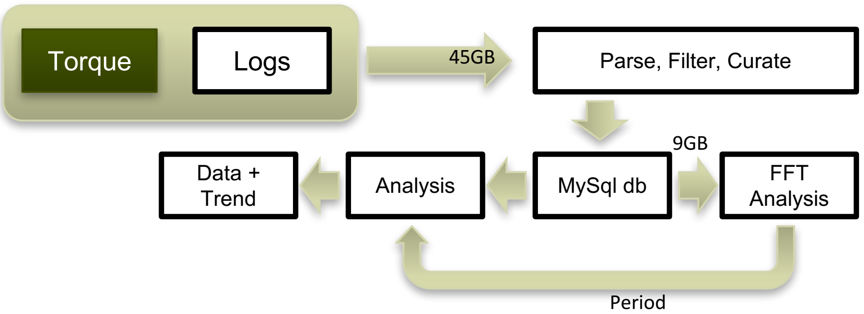
Cluster

Deployed January 2010	Deployed January 2014	Deployed in 2010
6,384 Nodes, 24 cores/node 154,216 cores	5,576 Nodes, 24 cores/node 133,824 cores	1,120 Nodes, 8/12/32 co/node 9,984 cores
1.28 Pflops/s	2.57 Pflops/s	106.5 Tflops

Torque Scheduler logs

Edison 2014, 1.3M jobs
Hopper 2010-2014, 4.5M Jobs
Carver 2010-2014, 9.3M Jobs

Analysis Pipeline



Paper I: Towards Understanding HPC Users and Systems: A NERSC Case Study. Submitted to JPDC (Journal of Parallel and Distributed Computing)

General conclusions

	Trend Analysis	Detailed Analysis
Jobs' Geometry / Runtime / CPU Cores	Hopper: Longer, Narrower. Carver: Shorter, Wider.	Smaller than expected. Mixed.
Wall clock time accuracy	Lower through time.	Jobs run < 50% req. time: Edison: 60% Hopper: 66% Carver: 93% Low quality prediction.
Jobs' Wait time	Increases through lifetime steadily. Pressure increases.	Jobs wait <=3h: Edison: 67% Hopper: 61% Carver: 80% Users adapt to system.
Utilization	Stable and high.	Real / Available Edison: 0.87 / 0.91 Hopper: 0.90 / 0.80 Carver: 0.88 / N/A Quite some downtime. Missing logging info.

Job Distribution	Edison	Hopper	Carver
%Jobs Wall Clock ≤ 2h	88%	86%	87%
%Jobs Width < 240 cores	69%	75%	99%
%Jobs Width ≤ 1 Node	39%	37%	92%
%Jobs Alloc. ≤ 1 core-h.	19%	26%	77%
%Jobs Alloc. ≥ 1K core-h.	7%	8%	~8%

Reference of current HPC systems workload

Paper I: Towards Understanding HPC Users and Systems: A NERSC Case Study. Submitted to JPDC (Journal of Parallel and Distributed Computing)

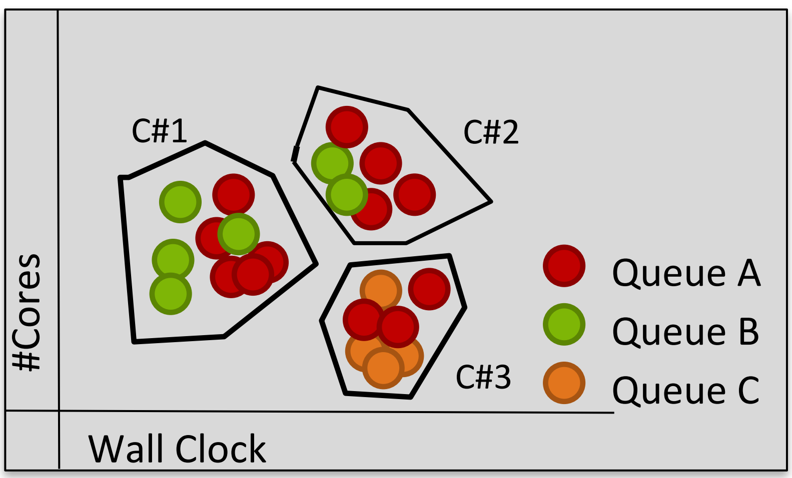
Job heterogeneity and performance

First observation of job geometry diversity

Measure | Performance

Job Heterogeneity

1) Overall: Search minimum k-means clusters in job geometry values (runtime, #cores)
2) Per queue: Map clusters on Queues



Q	Dom. C	Hom. Idx
A	1	41%
B	1	71%
C	3	100%

Paper I: Towards Understanding HPC Users and Systems: A NERSC Case Study. Submitted to JPDC (Journal of Parallel and Distributed Computing)

Heterogeneity Vs. Wait Time?

Wait time expectation

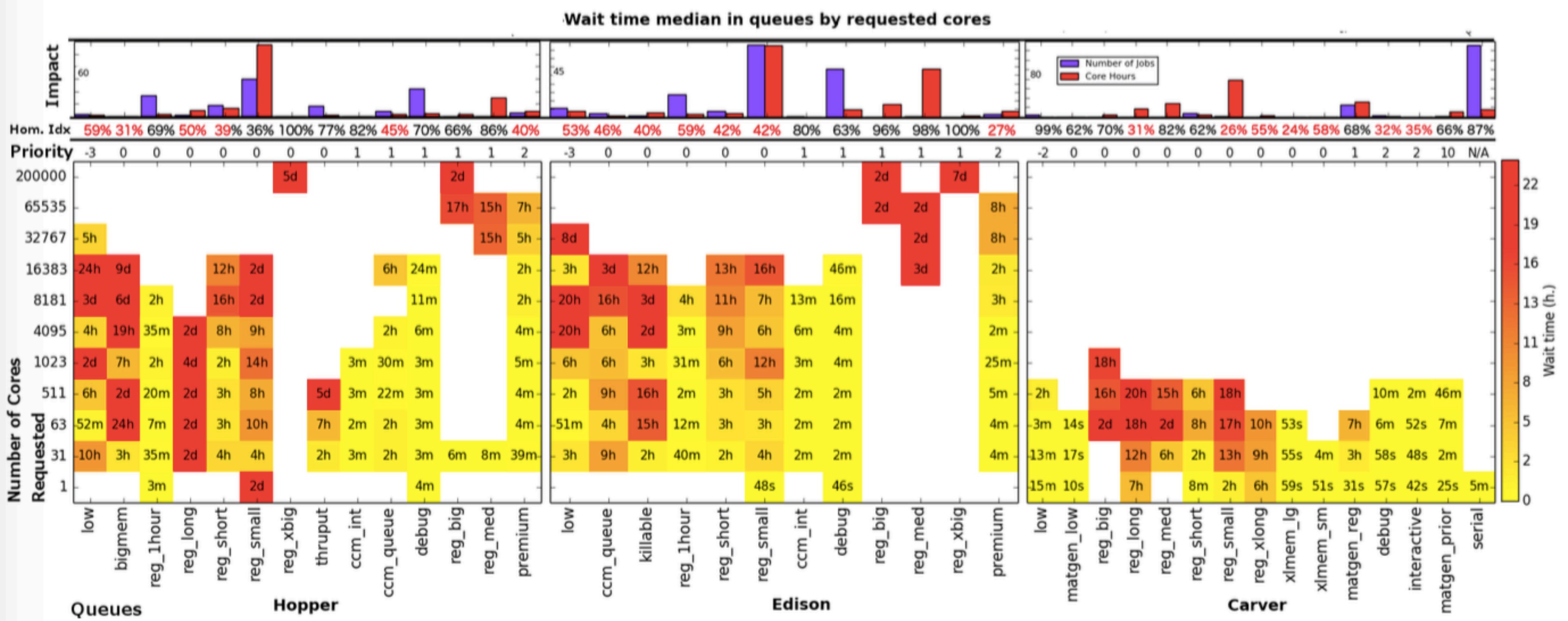
Job Geometry	Bigger = Longer Wait
Job Priority	Higher = Shorter Wait
Queue busy	Higher = Longer Wait

Observation

Queue Homog.	Low = Predictable?

Paper I: Towards Understanding HPC Users and Systems: A NERSC Case Study. Submitted to JPDC (Journal of Parallel and Distributed Computing)

Performance + Queues + Homogeneity



Paper I: Towards Understanding HPC Users and Systems: A NERSC Case Study. Submitted to JPDC (Journal of Parallel and Distributed Computing)

Conclusions on job analysis

Diversity also present in job geometries

Job heterogeneity affect queue's wait time predictability

What about re-shuffling queue?

What about extra schedulers?
Scheduler for smaller "opportunistic" jobs: Hawk*

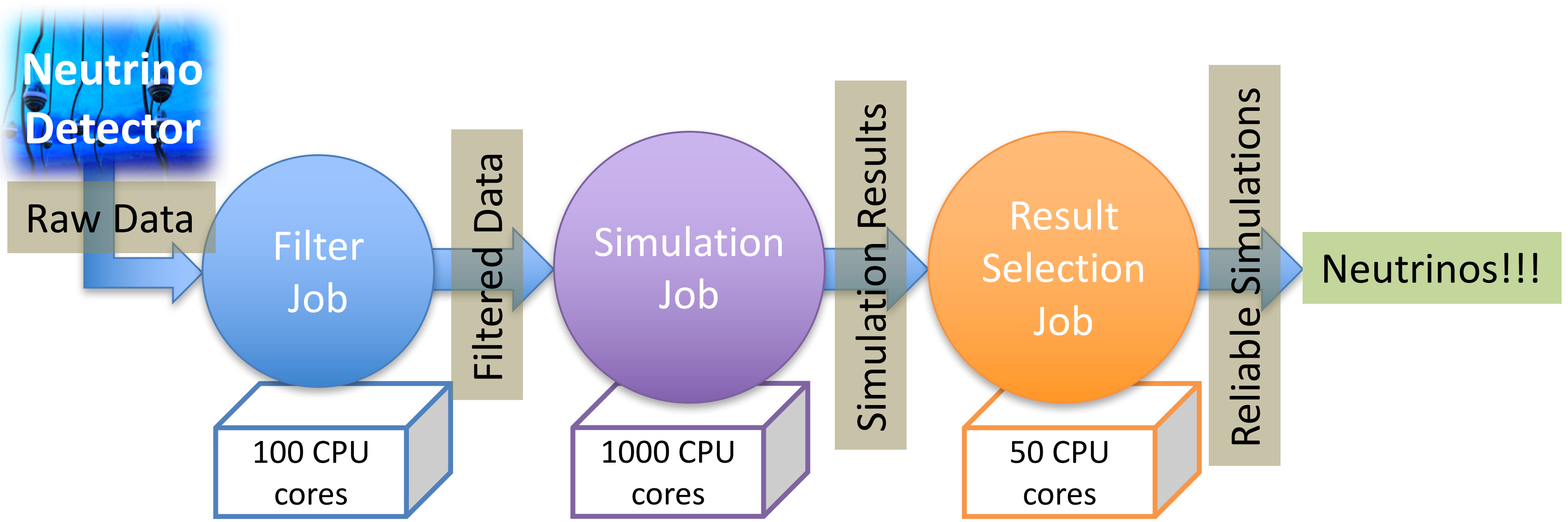
* Delgado, P., Dinu, F., Kermarrec, A. M., & Zwaenepoel, W. (2015, July). Hawk: Hybrid Datacenter Scheduling. In USENIX Annual Technical Conference
Paper I: Towards Understanding HPC Users and Systems: A NERSC Case Study. Submitted to JPDC (Journal of Parallel and Distributed Computing)

Challenges in High Performance Computing

SCHEDULING WORKFLOWS

What is a workflow?

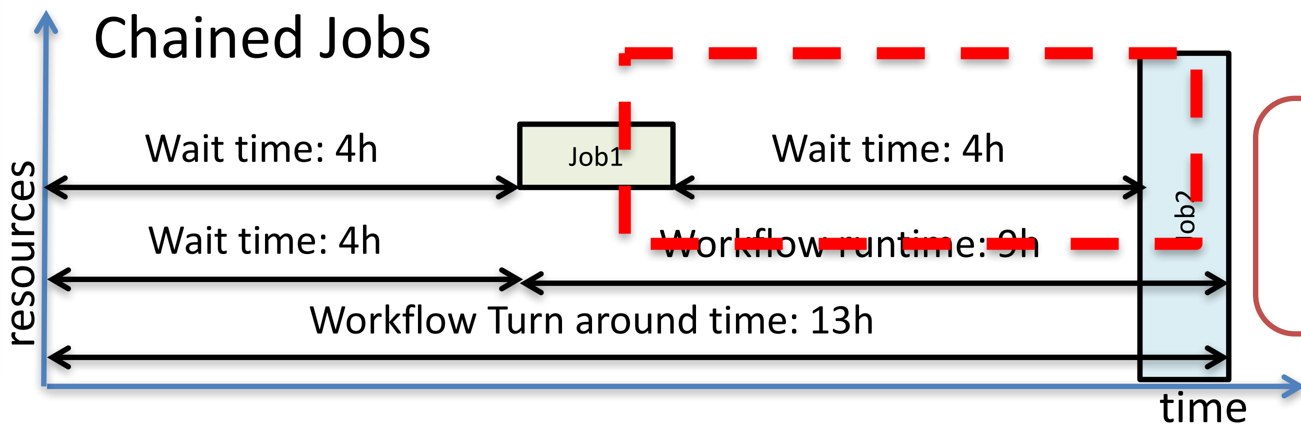
"… a composition of jobs with data or control dependencies…"



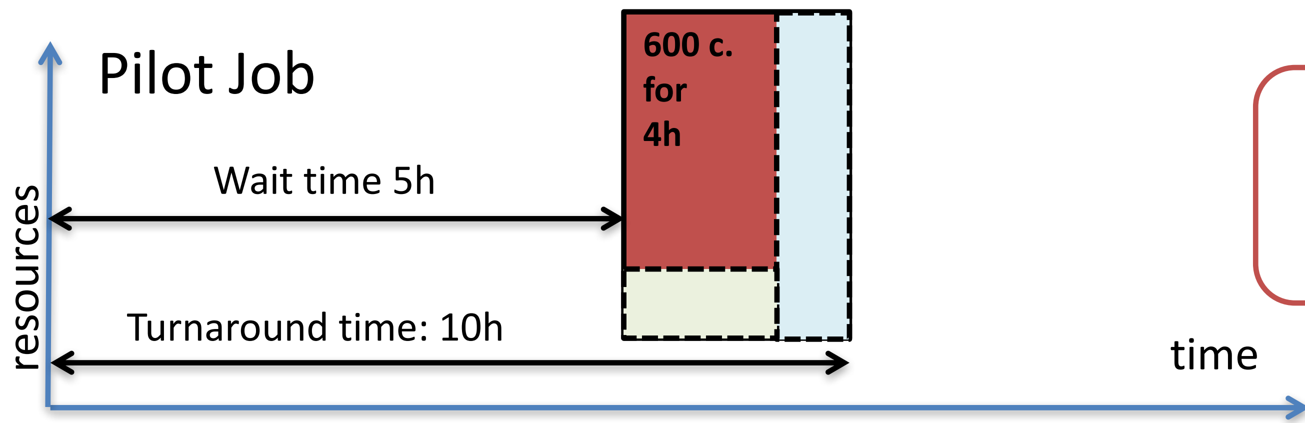
IceCube Neutrino telescope data pipeline (simplified)

Workflows and HPC Schedulers

Schedulers are not aware of the workflows



Long turnaround times

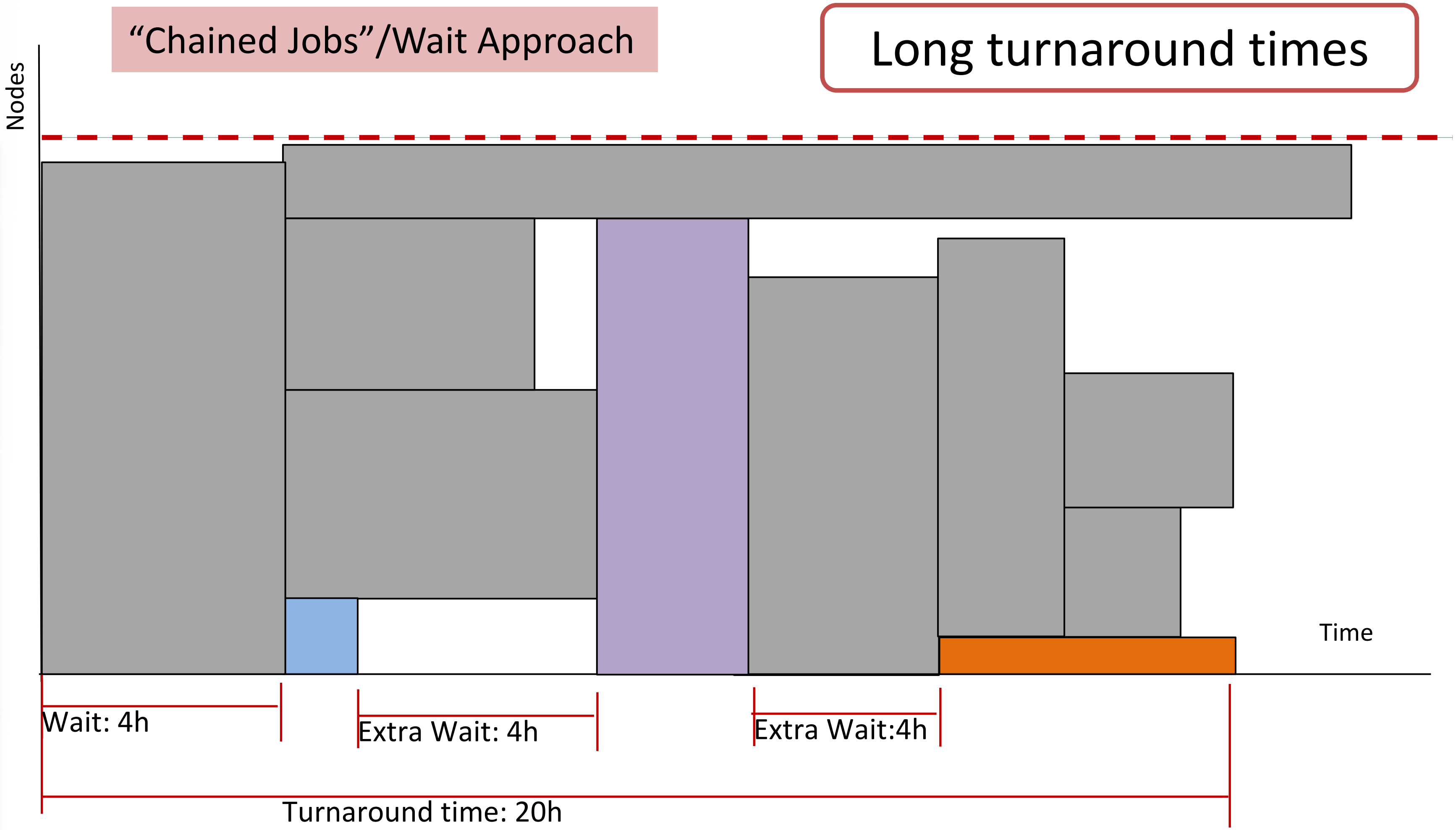


Resource Waste

Paper V: A new scheduling algorithm for workflows in HPC systems. 26th International Symposium on High-Performance Parallel and Distributed Computing (HPDC 2017)

Workflows and HPC Schedulers

Schedulers are not aware of workflows



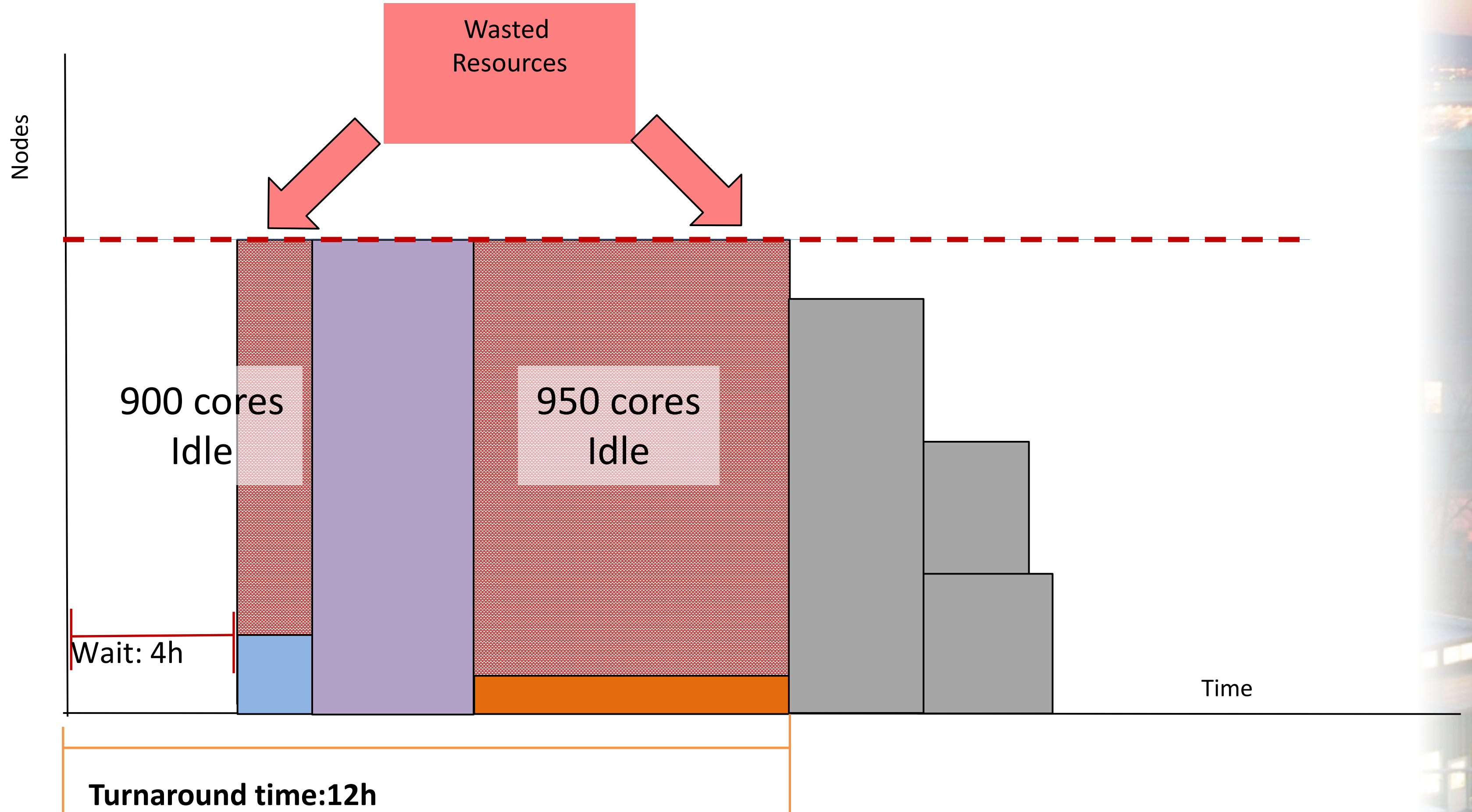
Paper V: A new scheduling algorithm for workflows in HPC systems. 26th International Symposium on High-Performance Parallel and Distributed Computing (HPDC 2017)

Workflows and HPC Schedulers

Schedulers are not aware of workflows

"Pilot job"/Waste Approach

Idles resources wasted



Paper V: A new scheduling algorithm for workflows in HPC systems. 26th International Symposium on High-Performance Parallel and Distributed Computing (HPDC 2017)

Improving Workflow Scheduling

Minimize workflow turnaround time without wasting resources

Define algorithm

"Does it work better?"

"How much better does it work?"

"Does it break anything?"

Minimize changes to scheduler

Paper V: A new scheduling algorithm for workflows in HPC systems. 26th International Symposium on High-Performance Parallel and Distributed Computing (HPDC 2017)

WoAS: Workflow Aware Scheduling

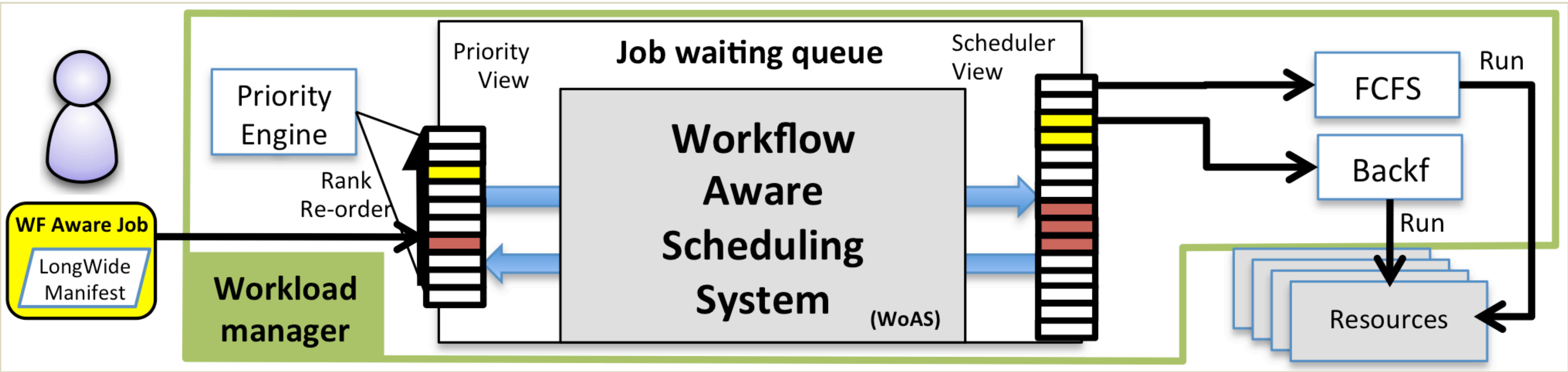
Best of both worlds

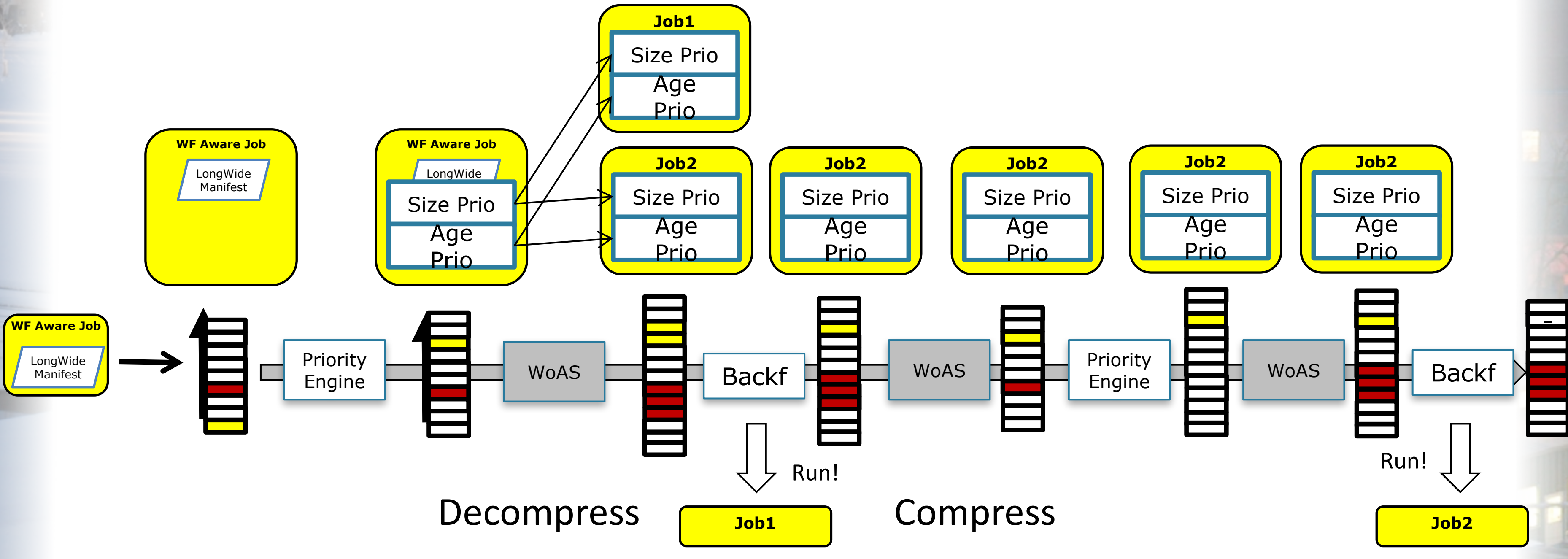
- Pilot Job
- Scheduler aware of "idle resources"

Paper V: A new scheduling algorithm for workflows in HPC systems. 26th International Symposium on High-Performance Parallel and Distributed Computing (HPDC 2017)

WoAS: In a real Scheduler

The "views" system

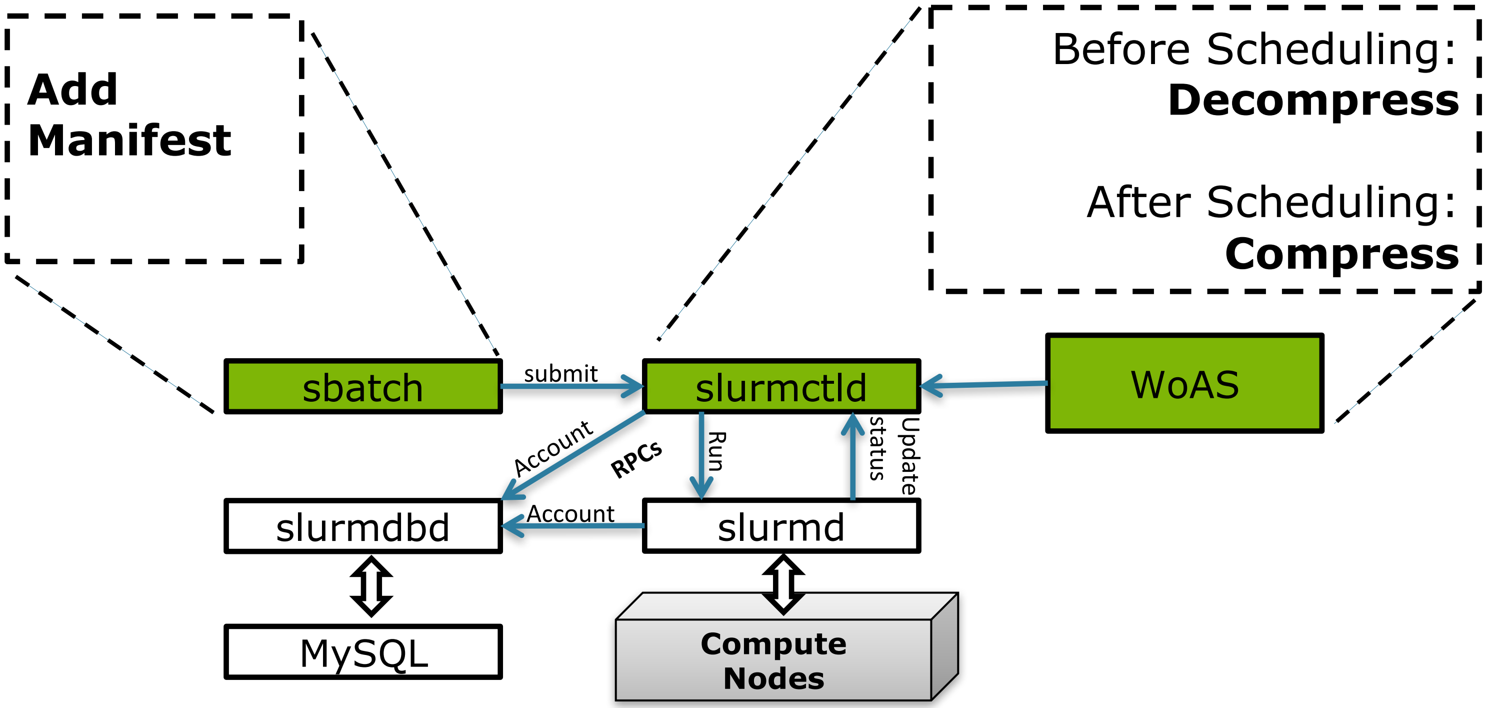




Paper V: A new scheduling algorithm for workflows in HPC systems. 26th International Symposium on High-Performance Parallel and Distributed Computing (HPDC 2017)

WoAS for Slurm

Main HPC scheduler and Open Source



Open Source Patch for Slurm 14.8.3

Paper V: A new scheduling algorithm for workflows in HPC systems. 26th International Symposium on High-Performance Parallel and Distributed Computing (HPDC 2017)

WoAS Evaluation

Theoretical analysis → **Turnaround time**
Depends on eligibility and intermediate waits

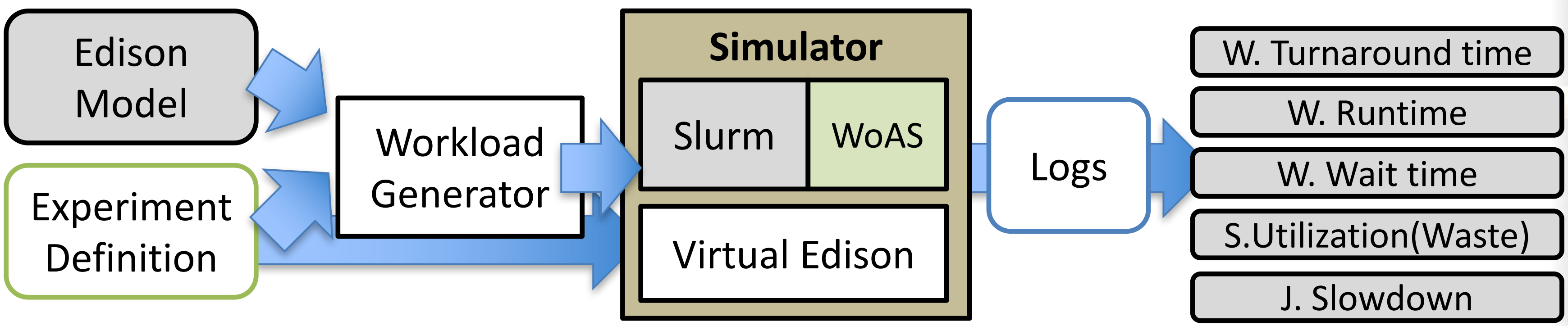
Experiments Simulation → **Works, Better, Breaks?**

Paper V: A new scheduling algorithm for workflows in HPC systems. 26th International Symposium on High-Performance Parallel and Distributed Computing (HPDC 2017)

WoAS Evaluation: Simulations

Workload and system modeled after Edison

Workload
- Edison-like synthetic jobs
- Experiment specific workflows
- 5 +1 Days of simulated system

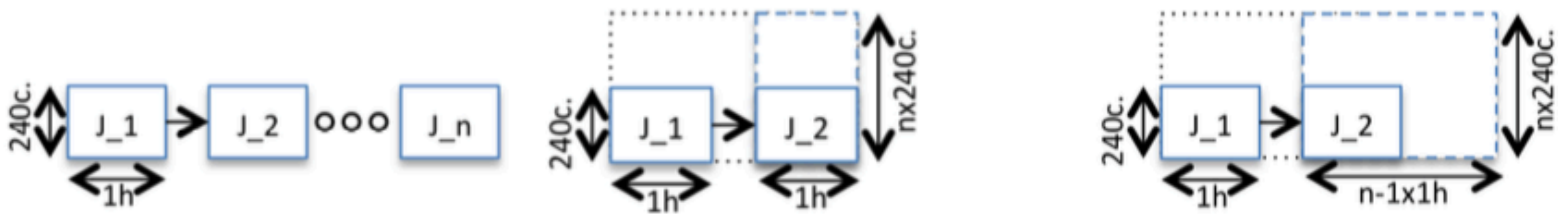


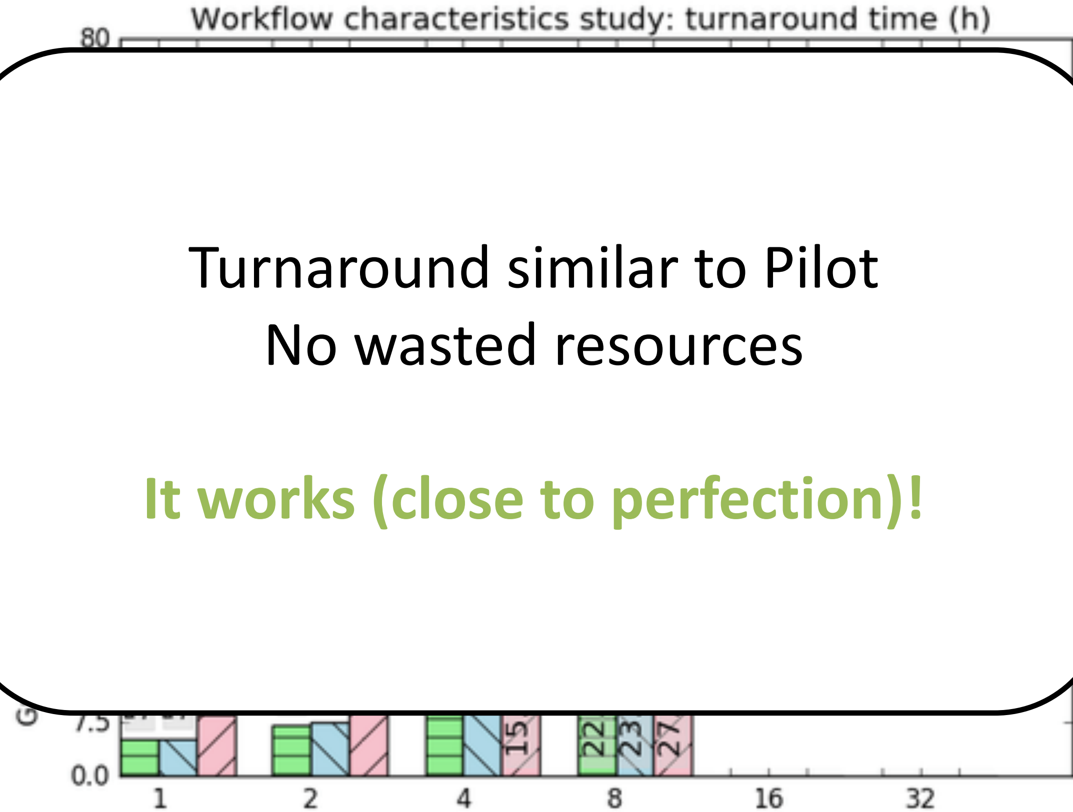
271 Scenarios, 1626 Experiments. 29 years of Edison: 3.8 Million Core-Years

Paper V: A new scheduling algorithm for workflows in HPC systems. 26th International Symposium on High-Performance Parallel and Distributed Computing (HPDC 2017)

Results: Does WoAS work?

Type A Type B Type C





Turnaround similar to Pilot
No wasted resources

It works (close to perfection)!

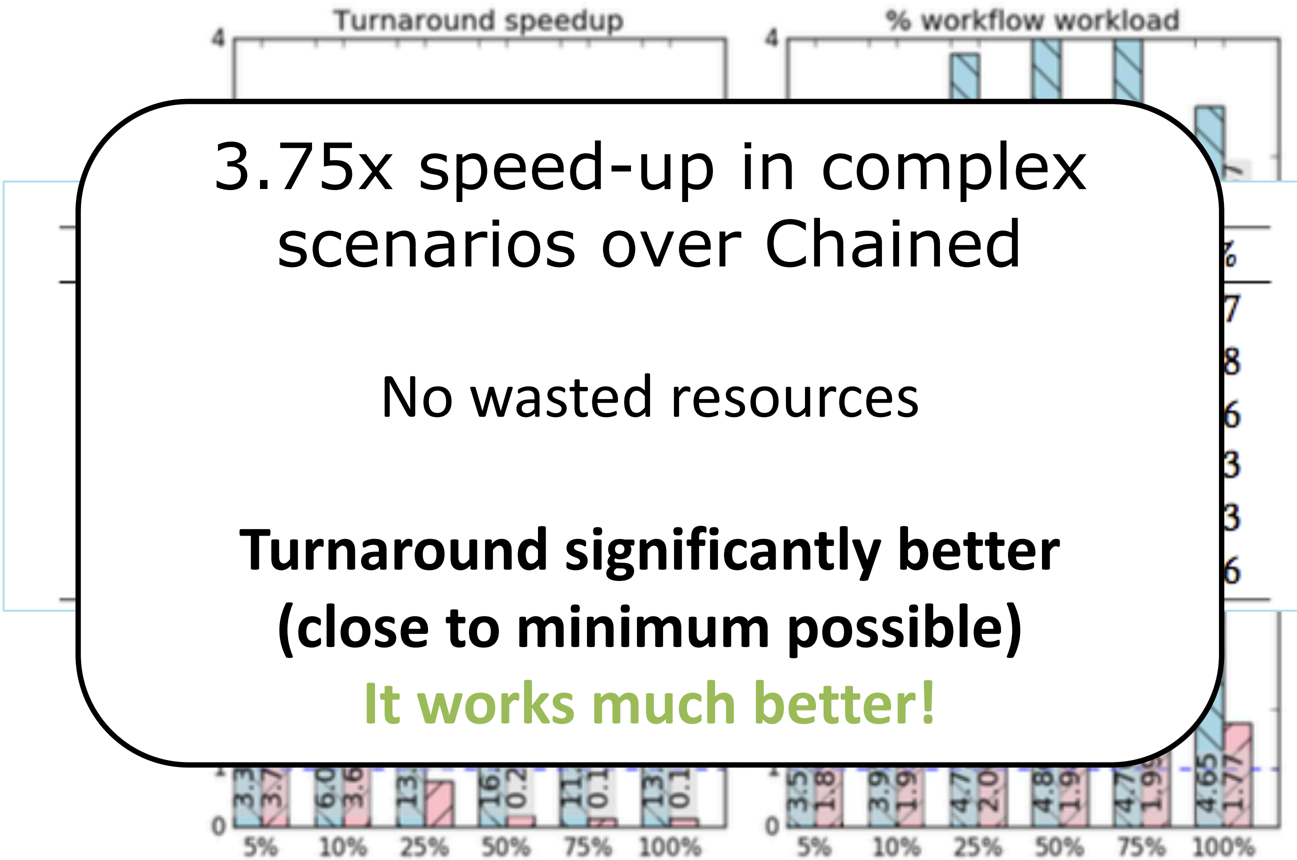
Paper V: A new scheduling algorithm for workflows in HPC systems. 26th International Symposium on High-Performance Parallel and Distributed Computing (HPDC 2017)

Results: How much does WoAS work better?

Different core-hours % for workflows

Workflows

- LongWide
- WideLong
- FloodPlain
- Montage
- Cybershake
- Sipht



3.75x speed-up in complex scenarios over Chained

No wasted resources

Turnaround significantly better (close to minimum possible)

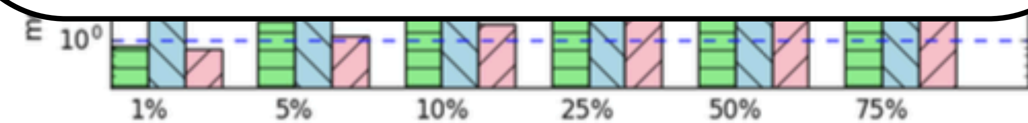
It works much better!

Paper V: A new scheduling algorithm for workflows in HPC systems. 26th International Symposium on High-Performance Parallel and Distributed Computing (HPDC 2017)

Evaluation: Does WoAS break the schedule?

Regular Jobs Slowdown Analysis

No significant effect
on jobs' slowdown

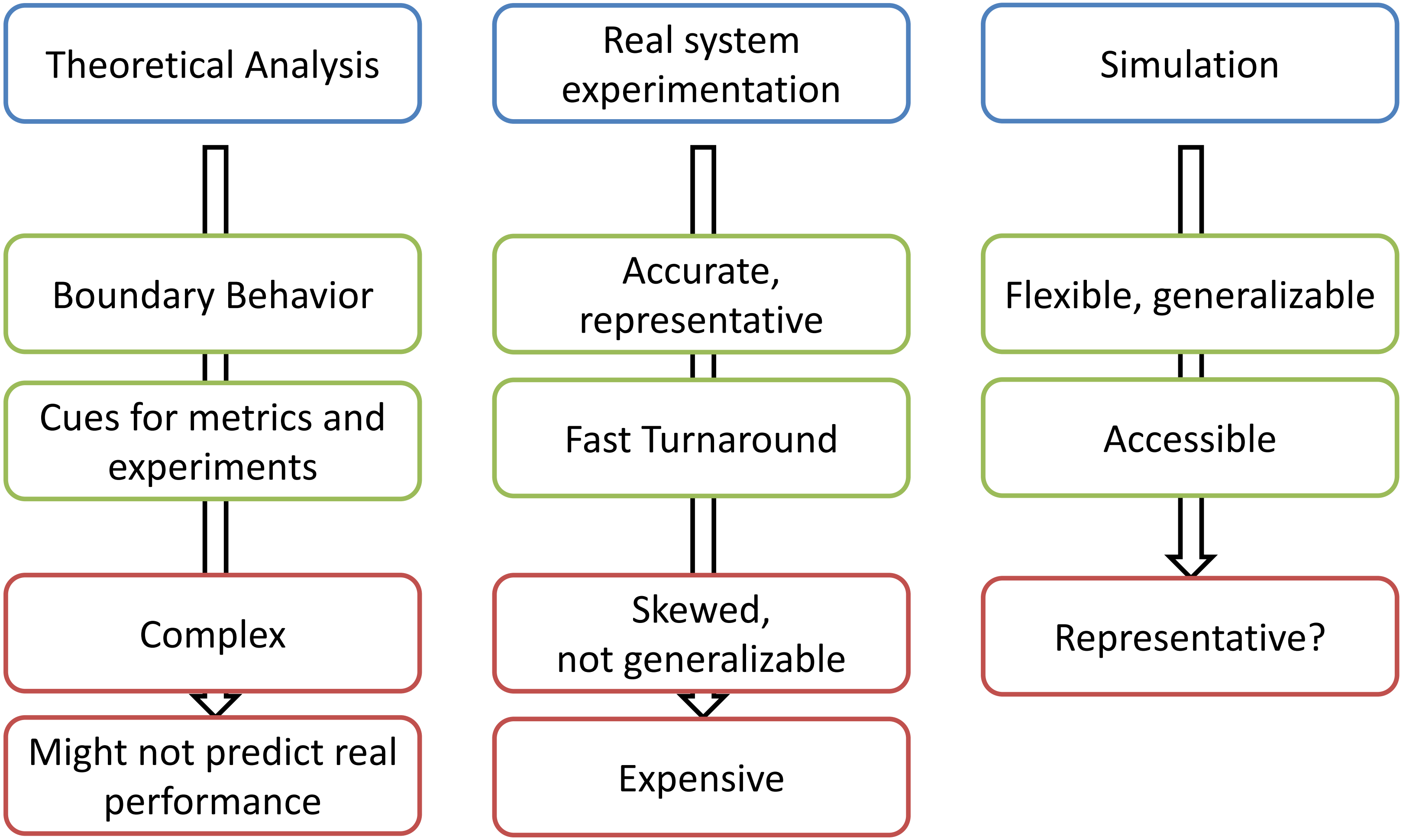


Paper V: A new scheduling algorithm for workflows in HPC systems. 26th International Symposium on High-Performance Parallel and Distributed Computing (HPDC 2017)

Challenges in High Performance Computing

HPC SCHEDULING RESEARCH

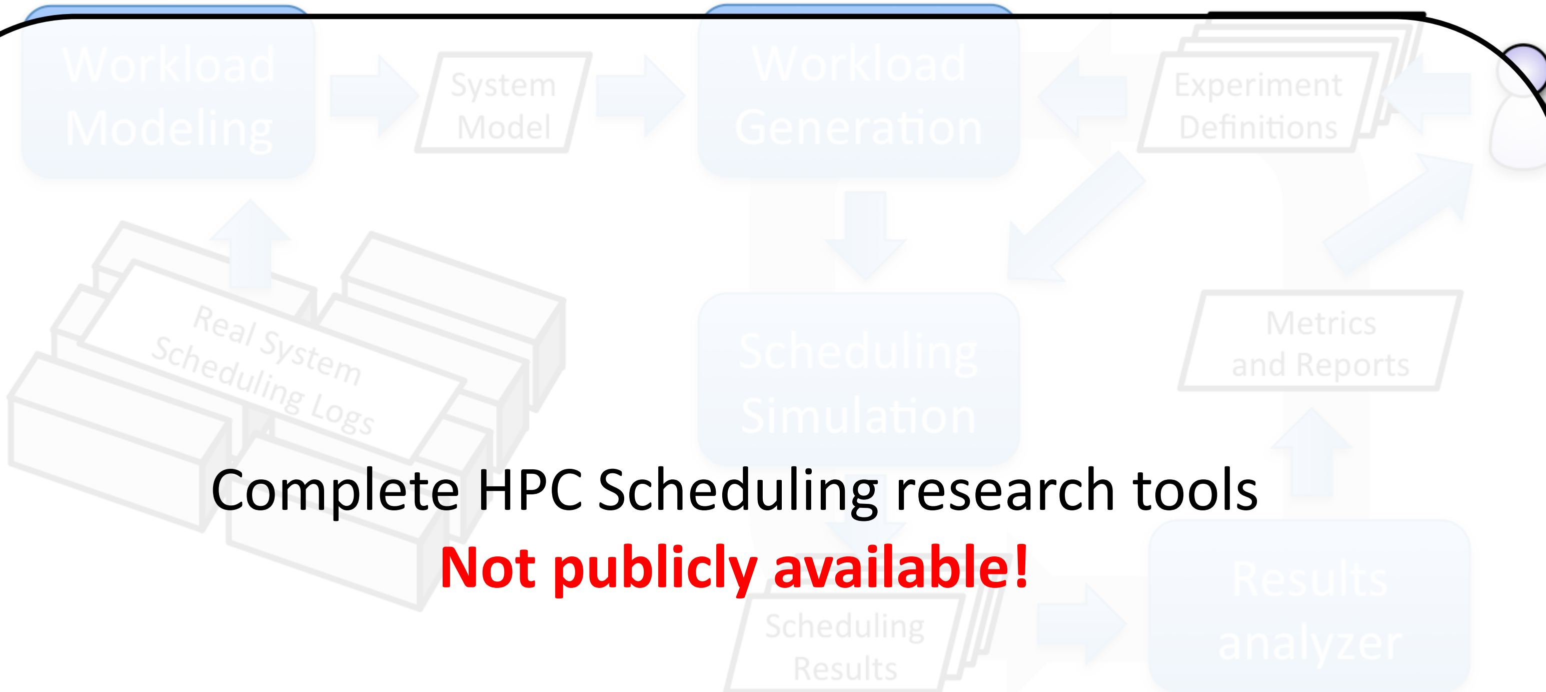
HPC Scheduling Research approaches



Schwiegelshohn, U.: How to design a job scheduling algorithm. In: Workshop on Job Scheduling Strategies for Parallel Processing. pp. 147–167. Springer (2014)

Paper IV: ScSF: A Scheduling Simulation Framework. 21th Workshops on Job Scheduling Strategies for Parallel Processing (JSSPP 2017)

HPC Scheduling Simulation: Research cycle



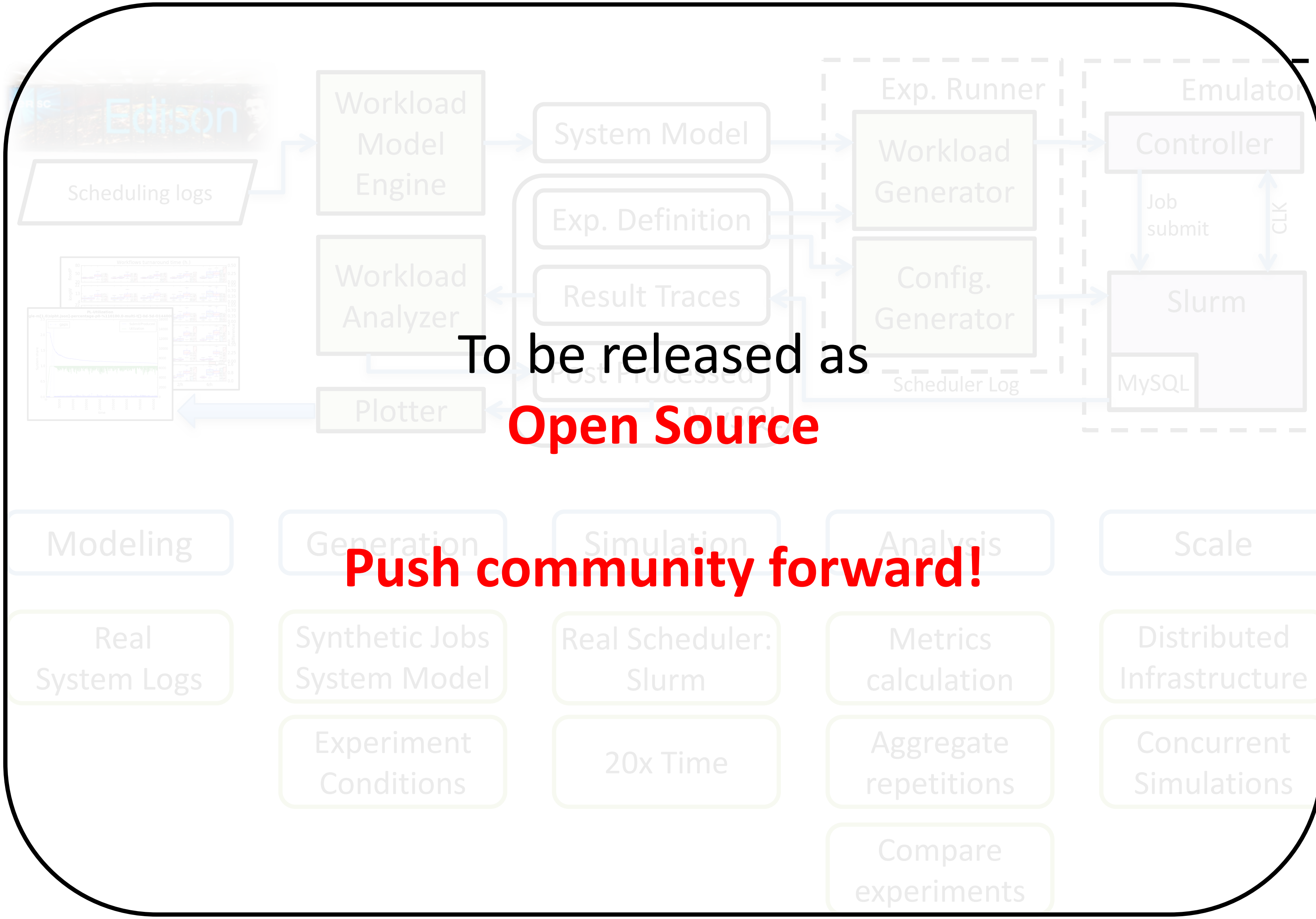
Complete HPC Scheduling research tools

Not publicly available!

	Modeling	Generation	Simulation	Analysis
Alea	✗	✗	✓ Not real scheduler	✗
Slurm Simulator	✗	✗	✓ Slow	✗
Parallel Archive	✗	✓ Old Small	✗	✗

Paper IV: ScSF: A Scheduling Simulation Framework. 21th Workshops on Job Scheduling Strategies for Parallel Processing (JSSPP 2017)

ScSF: Scheduling Simulation Framework



To be released as

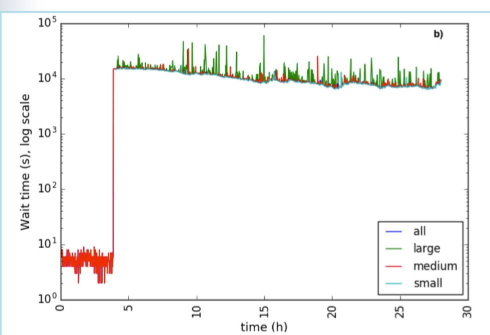
Open Source

Push community forward!

Paper IV: ScSF: A Scheduling Simulation Framework. 21th Workshops on Job Scheduling Strategies for Parallel Processing (JSSPP 2017)

ScSF: Workload Modeling & Generation

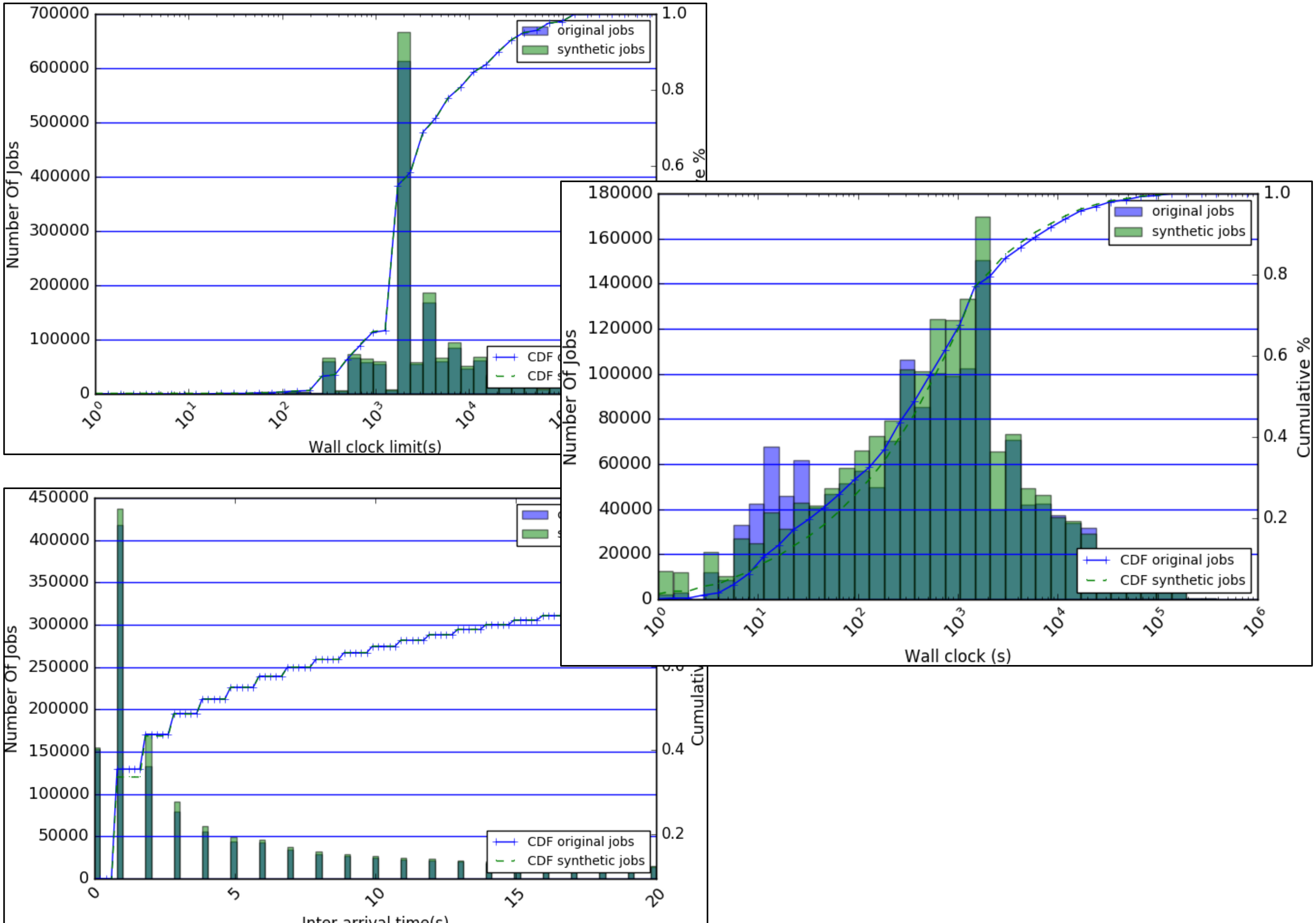
- Empirical Distribution
- Open Loop
- Job Pressure Control
- Cold start stabilization
- Baseline wait time
- Workflows




```
{"tasks": [
  {"id":"SWide", "cmd":"./W.py", "cores":480, "rtime":360.0},
  {"id":"SLong", "cmd":"./L.py", "cores":48, "rtime":1440.0,
    "deps": ["SWide"]}]}
```

Paper IV: ScSF: A Scheduling Simulation Framework. 21th Workshops on Job Scheduling Strategies for Parallel Processing (JSSPP 2017)

ScSF: Workload Modeling & Generation

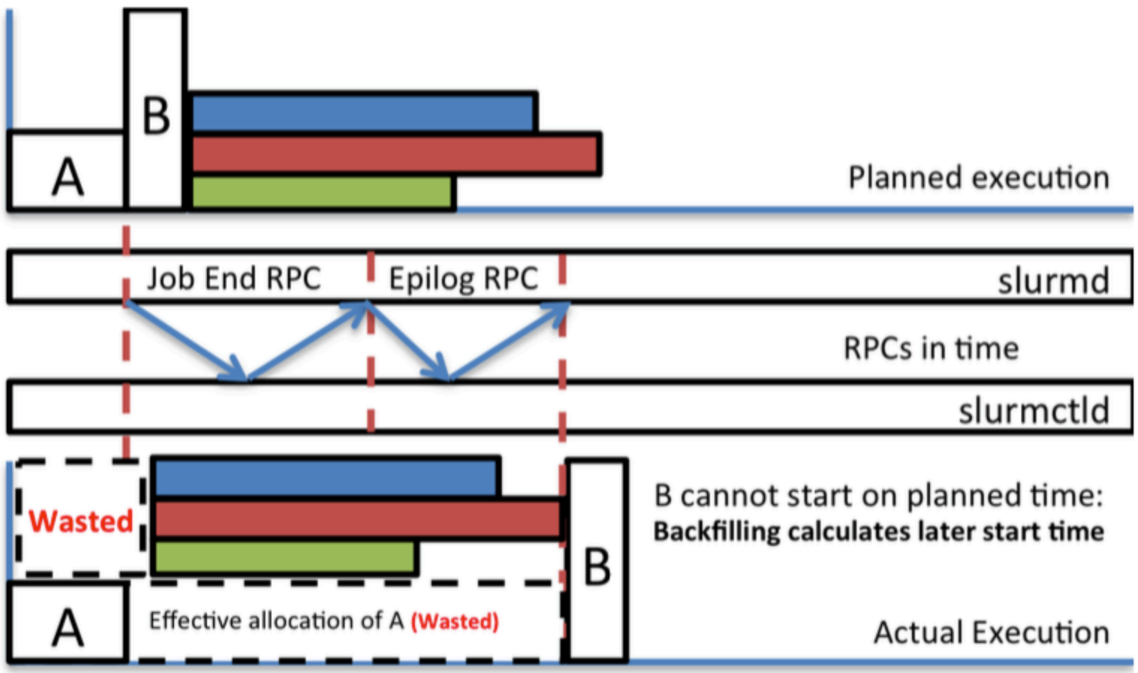


Paper IV: ScSF: A Scheduling Simulation Framework. 21th Workshops on Job Scheduling Strategies for Parallel Processing (JSSPP 2017)

ScSF: Slurm Simulator

Wraps real Slurm Scheduler | Emulates system and job execution | Emulates job submission (replay)

Original Implementation: Slow (1 to 1), no determinism



Slurm simulator improved by synchronizing scheduling threads

Faster (20x speed-up)

Time consistent

Achieve good utilization with out-of-the-box scheduler

Paper IV: ScSF: A Scheduling Simulation Framework. 21th Workshops on Job Scheduling Strategies for Parallel Processing (JSSPP 2017)

ScSF: Running experiments in scale

170 Worker VMs
17 Hosts
Two continents

Simulate
30 years
of Edison's Life

ScSF

1 month

Paper IV: ScSF: A Scheduling Simulation Framework. 21th Workshops on Job Scheduling Strategies for Parallel Processing (JSSPP 2017)

ScSF: Lessons learned

Slurm is a complex old-fashion-SWE package:
expensive to modify

Loss-less experiment restart is needed
Specially if experiment runtime are long (e.g. 5 days)

HPC scheduling requires a lot of simulation

Think big from the beginning!

Loaded systems network fail
So harden your comms

The system is as weak as its weakest link
Single point of failure

Paper IV: ScSF: A Scheduling Simulation Framework. 21th Workshops on Job Scheduling Strategies for Parallel Processing (JSSPP 2017)

Summary and Conclusions

In this work… … we covered the complete research cycle of HPC scheduling.

Understanding new applications and workloads and their impact on schedulers

Created tools to perform HPC scheduling Research

Improved HPC workflow scheduling

Insights on new systems, applications challenges, and scheduling models

Current systems and applications conform

a *Brave New World*

that requires new scheduling models and algorithms!

#hpcmatters!

THANKS

UMEÅ UNIVERSITET

BERKELEY LAB