ExDe: Design Space Exploration of Scheduler Architectures and Mechanisms for Serverless Data-processing

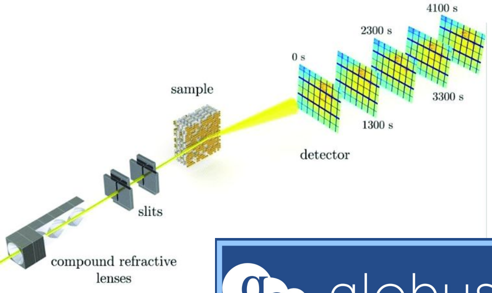
Sacheendra Talluri[1], Nikolas Herbst[2], Cristina Abad[3], Tiziano De Matteis[1], Alexandru Iosup[1]

[1]VU Amsterdam, [2]Würzburg University, [3]ESPOL

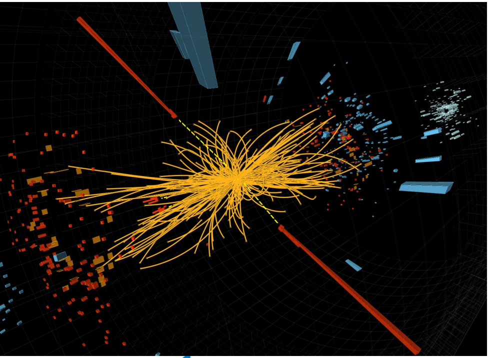
https://atlarge-research.com/pdfs/exde_fgcs23.pdf

@Large Research

Serverless Supports A Variety of Workloads



globus
a uchicago non-profit service



Text Search

Business Analytics

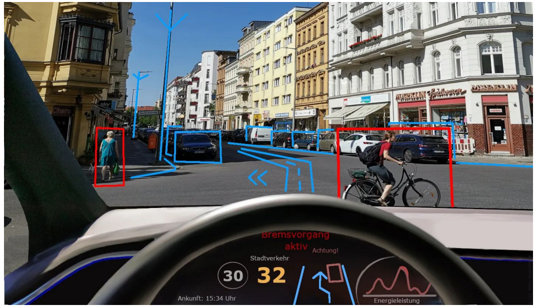
Scientific Computing

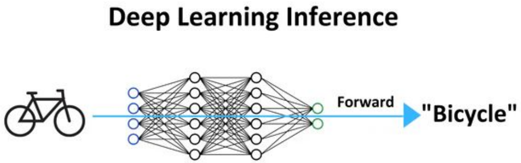
Multimedia Search

Video Analytics

ML Inference

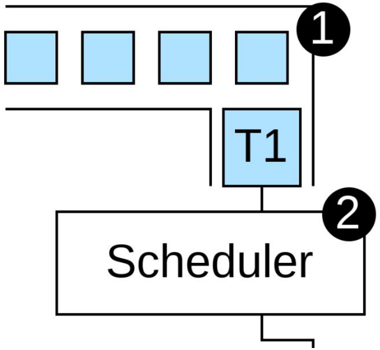
…



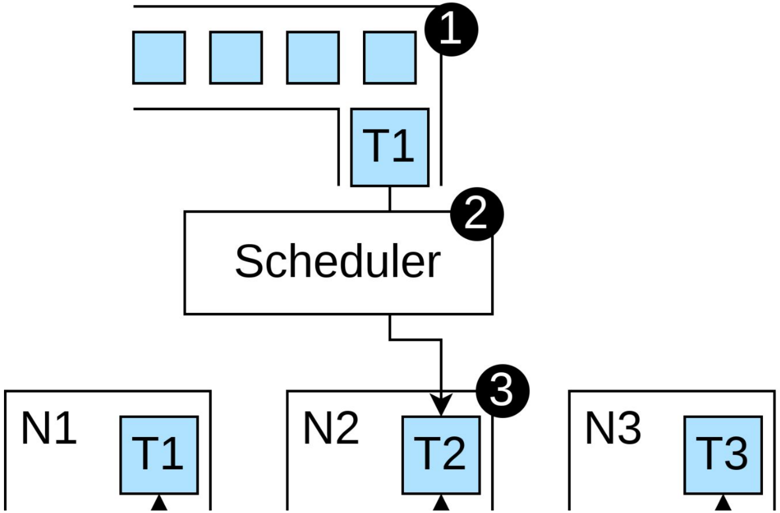


https://atlarge-research.com/pdfs/exde_fgcs23.pdf

Serverless Data-processing

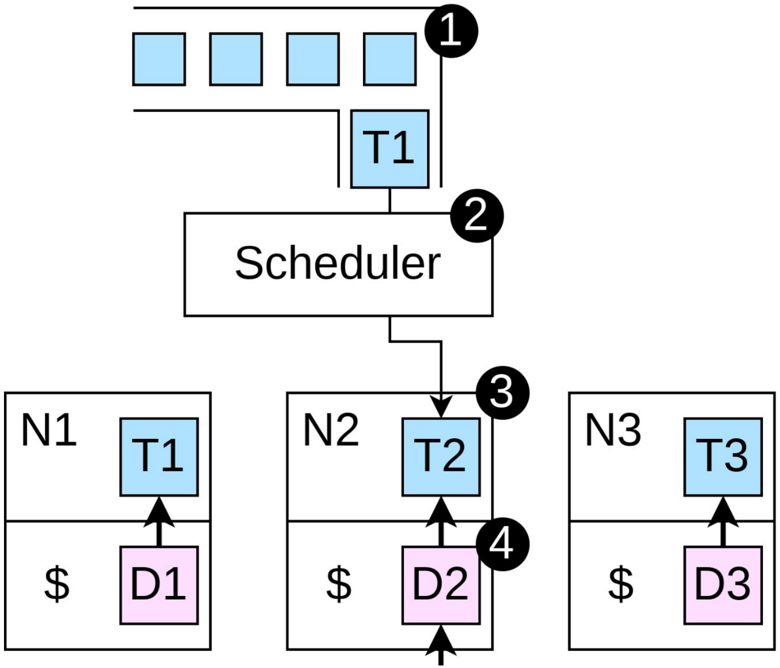


Serverless Data-processing



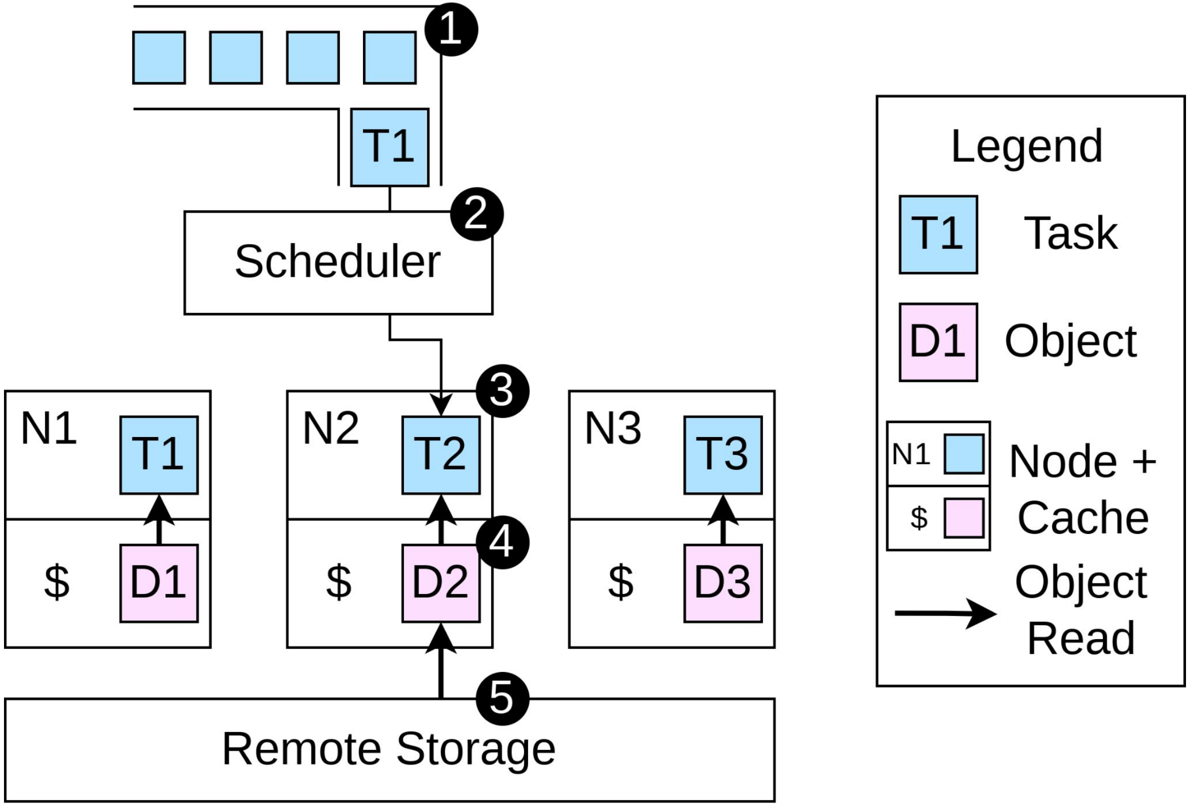
https://atlarge-research.com/pdfs/exde_fgcs23.pdf

Serverless Data-processing



https://atlarge-research.com/pdfs/exde_fgcs23.pdf

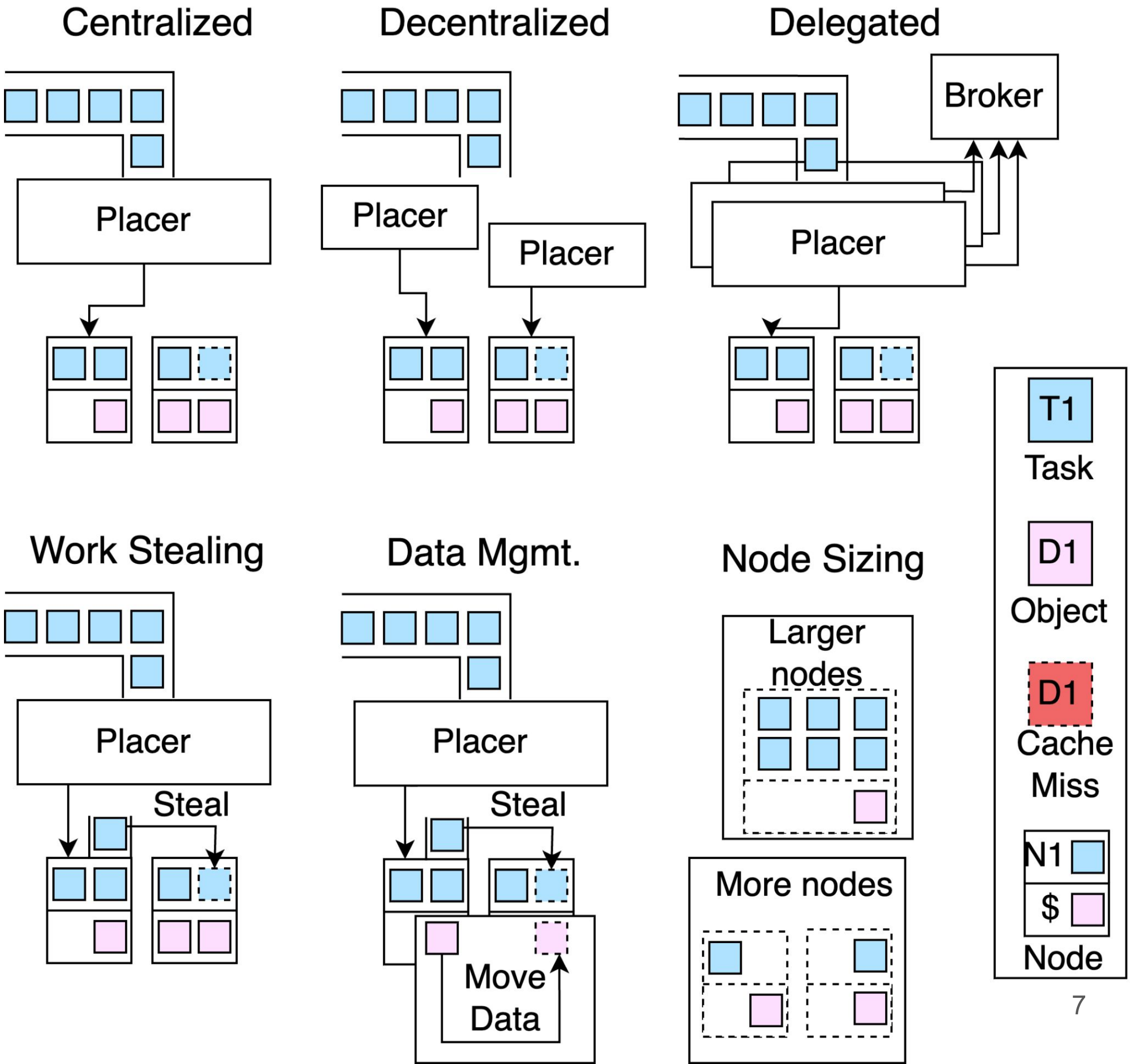
Serverless Data-processing



https://atlarge-research.com/pdfs/exde_fgcs23.pdf

Serverless Requires A Variety of Schedulers & Mechanisms

Large Design Space



https://atlarge-research.com/pdfs/exde_fgcs23.pdf

Research Questions

How do we compare schedulers across the design space?

How do we characterize scheduler performance?

Research Questions (Challenges)

How do we compare schedulers across the design space?

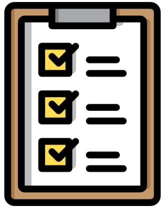
- A lot of schedulers are quite close to each other

How do we characterize scheduler performance?

- Wide workload and design variety

How do we compare schedulers across the design space?

Exhaustively list all possible features



Only look at extreme points

Look at what is implemented

https://atlarge-research.com/pdfs/exde_fgcs23.pdf

Scheduler Frames

The **scheduler frame** is the set of all mechanisms in the scheduler that enable actions **not possible by any local modification** of the scheduler algorithm and policy. Instead, a frame requires coordination between multiple scheduler components.

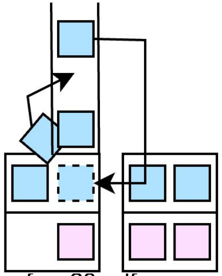
Each scheduler frame is an identifiable point in the design space.

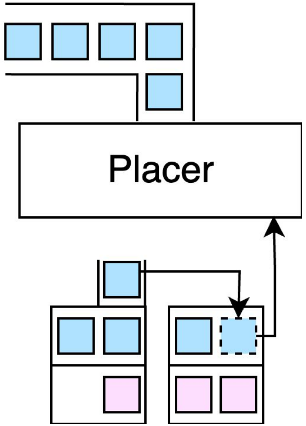
Scheduler Frames

The **scheduler frame** is the set of all mechanisms in the scheduler that enable actions **not possible by any local modification** of the scheduler algorithm and policy. Instead, a frame requires coordination between multiple scheduler components.

Each scheduler frame is an identifiable point in the design space.

Preemption -
local modification





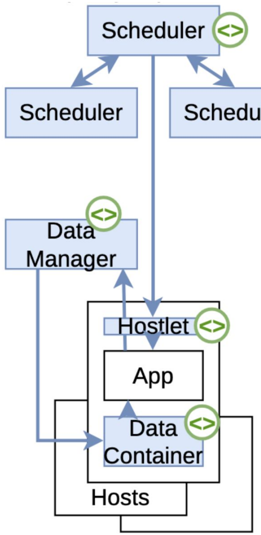
Work stealing - placer
and host software
modification

https://atlarge-research.com/pdfs/exde_fgcs23.pdf

Existing Mechanisms Characterized Using Frames

Mechanism	Components used						Implementations
	Placer(s)	Broker	Host	Client	Metadata	DM	
Architecture	✓	✓		✓			Centralized [28, 29], decentralized [30, 4], delegated [31, 13], hybrid [32, 33]
Preemption	✓		✓				Threshold-based [34], fair sharing [24]
Control-flow	✓	✓	✓				Push/pull [35], speculative exec. [36]
Data placement	✓				✓	✓	Shuffle [27], intermediate data [21]
Fault tolerance	✓			✓	✓	✓	Checkpoint [37], retry [38]
Networking	✓				✓		NetHint [39]
Barriers	✓	✓					Gang scheduling [18, 40]

How do we characterize scheduler performance?



Using simulation

- Isolate performance impact
- Evaluate 10s of designs
- Across many workloads
- Time-efficient

Fine-grained modeling of components and communication between them as identified in the scheduler frame

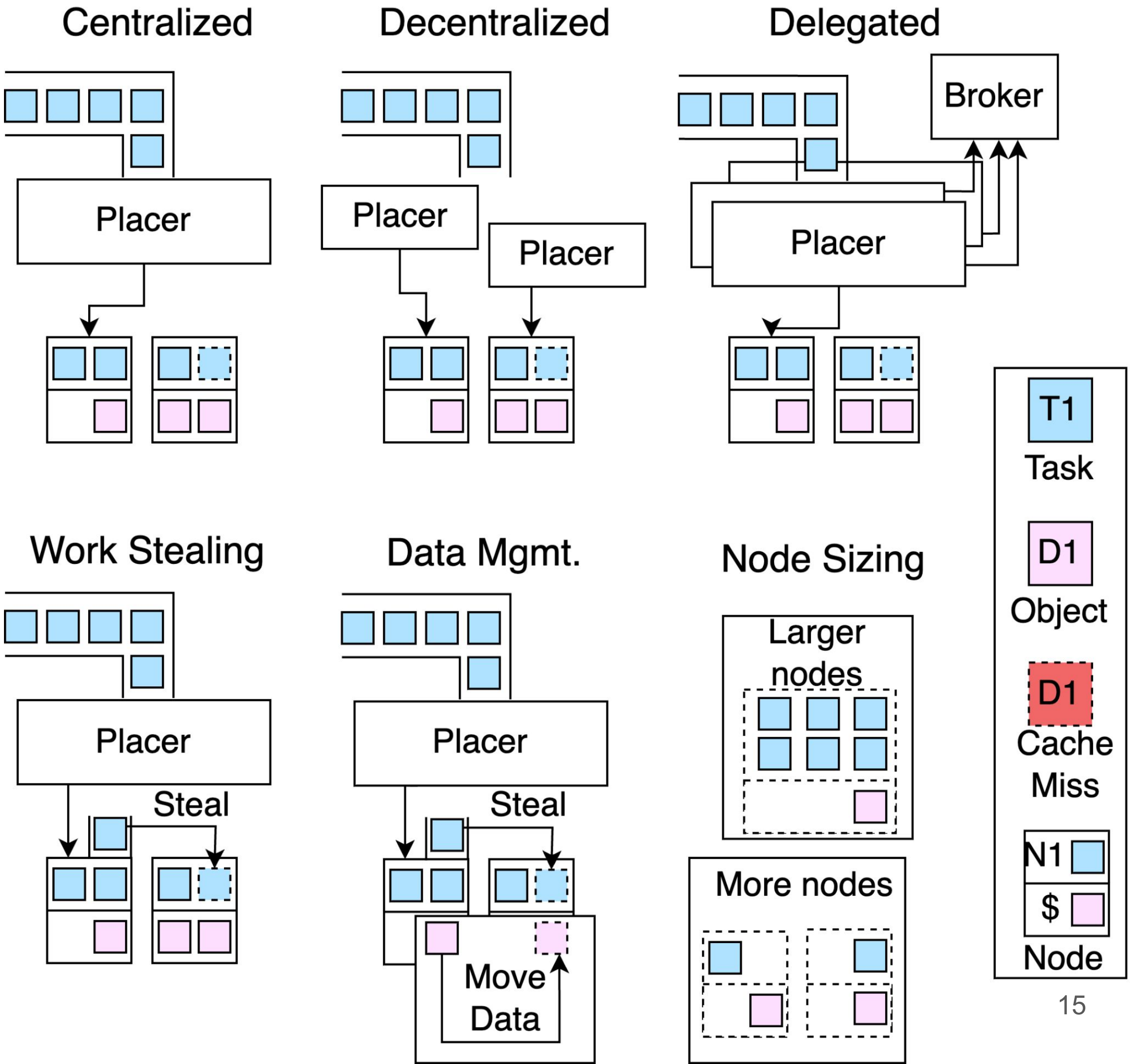
- Take into account system's distributed nature in simulation

Simulator Setup

OpenDC Simulator

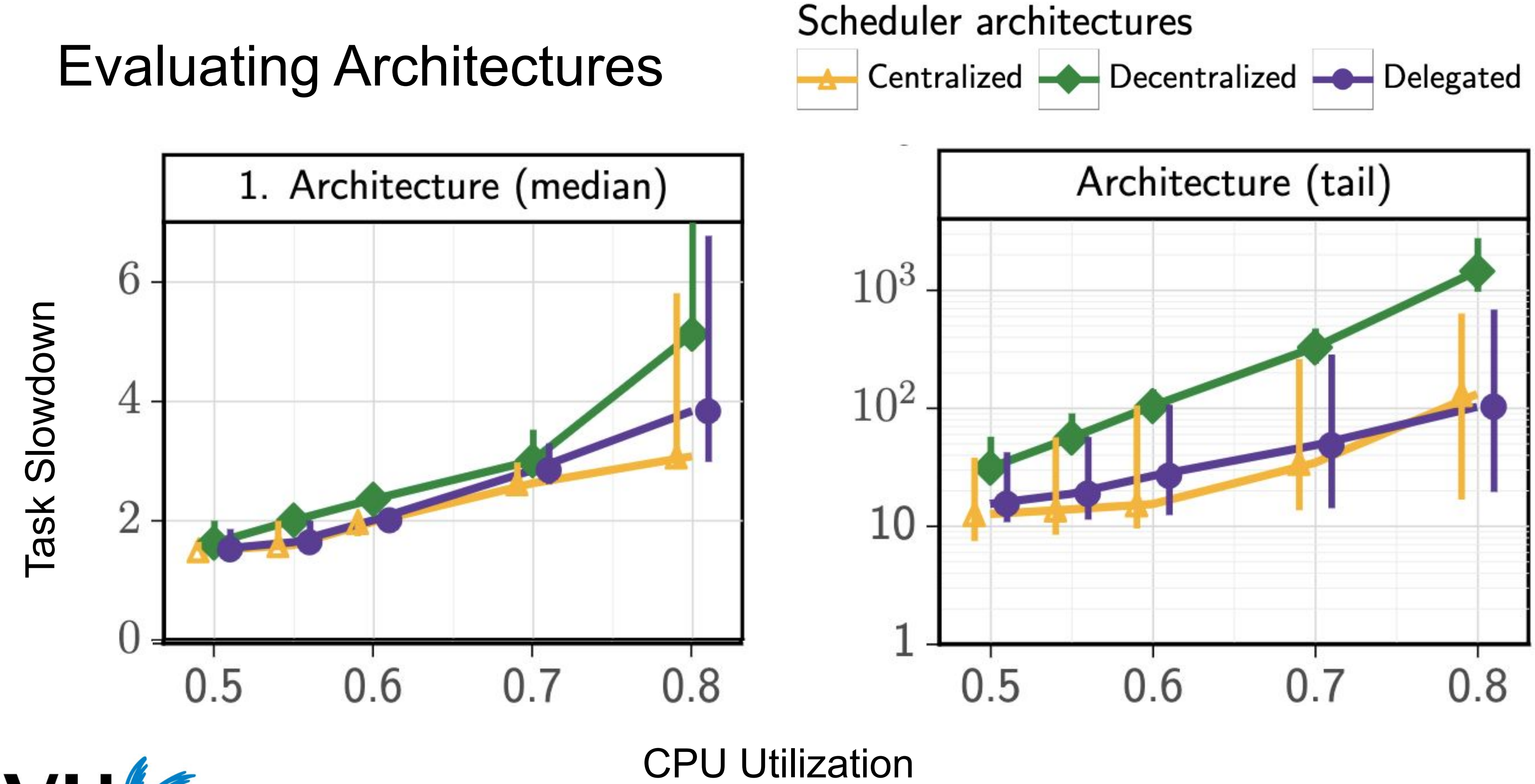
- 54 traces from IBM
- 3 million tasks per trace
- 15-31 node clusters based on the trace
- 4 CPUs per node
- FIFO + Least loaded node placement policy

Metric:
Slowdown = executime time / ideal execution time



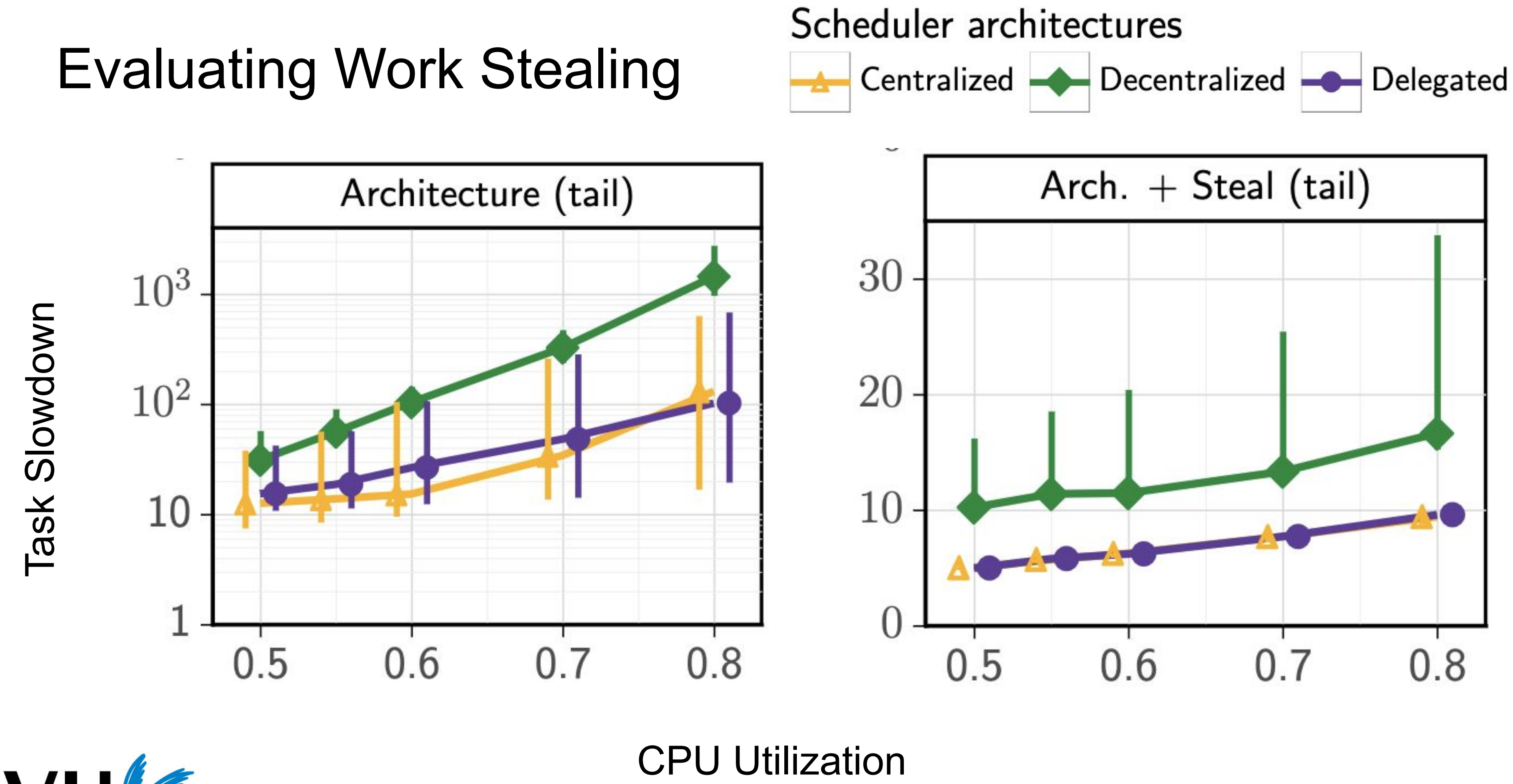
https://atlarge-research.com/pdfs/exde_fgcs23.pdf

Evaluating Architectures



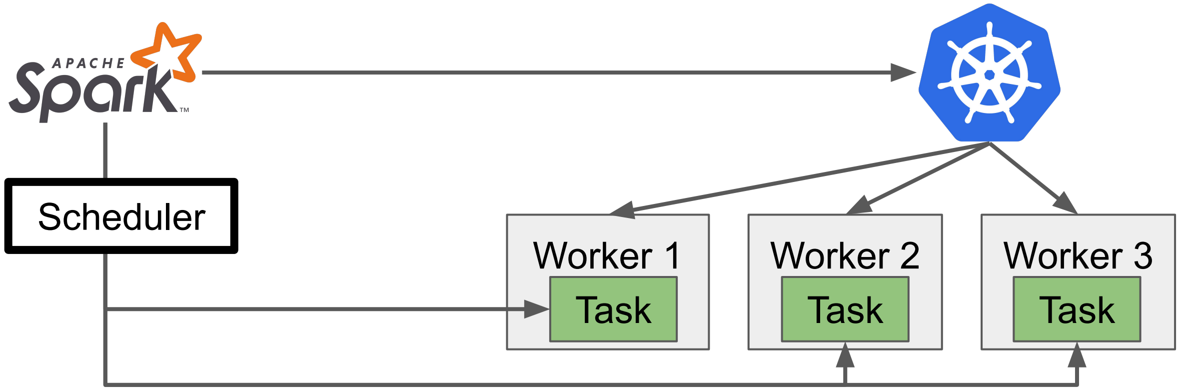
https://atlarge-research.com/pdfs/exde_fgcs23.pdf

Evaluating Work Stealing

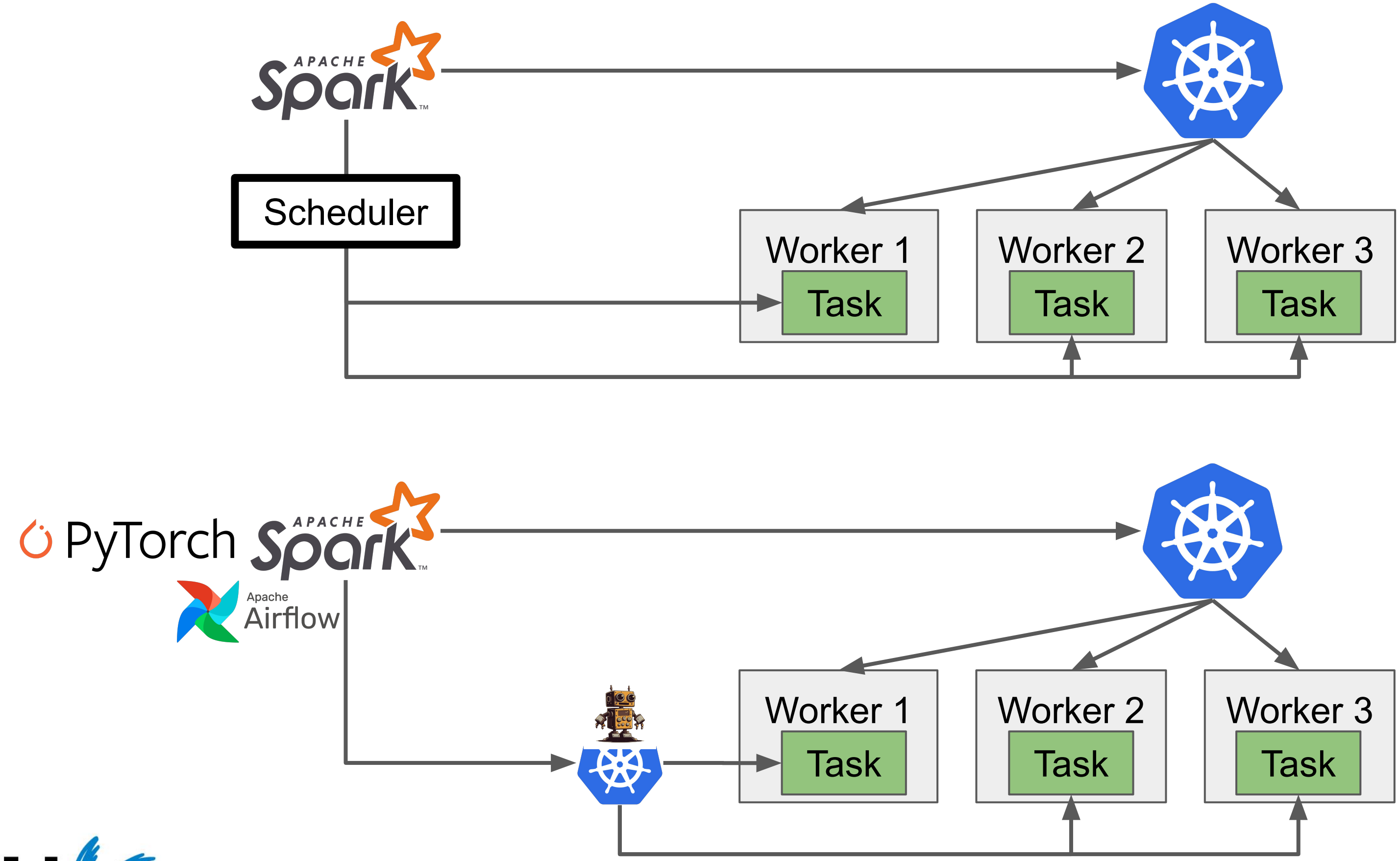


https://atlarge-research.com/pdfs/exde_fgcs23.pdf

Realisation For Real Systems [WIP]
- Kubernetes-based Scheduler as a Service



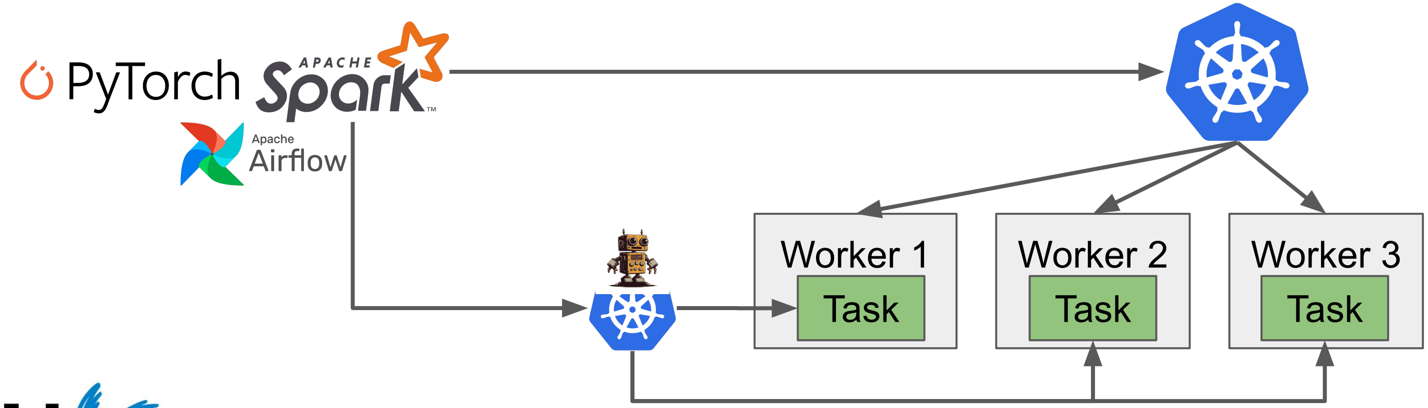
Kubernetes-based Scheduler as a Service [WIP]



https://atlarge-research.com/pdfs/exde_fgcs23.pdf

Kubernetes-based Scheduler as a Service [WIP]

- Distributed component support based on Scheduler Frames
- Plug-n-play for multiple systems
- Advanced scheduling policy and mechanism library

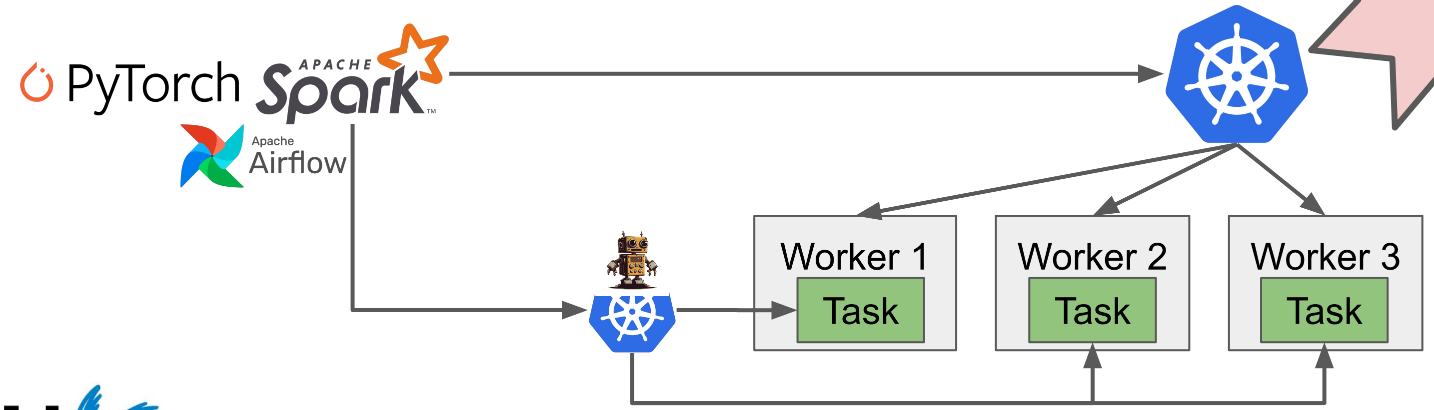


https://atlarge-research.com/pdfs/exde_fgcs23.pdf

Kubernetes-based Scheduler as a Service [WIP]

- Distributed component support based on Scheduler Frames
- Plug-n-play for multiple systems
- Advanced scheduling policy and mechanism library

Join us!



https://atlarge-research.com/pdfs/exde_fgcs23.pdf

Key Takeaways

1. Scheduler architecture and mechanism design space is large
2. Scheduler frames help identify unique points in the design space
   a. Difference emphasized by non-local modification
3. Implemented in OpenDC for design space characterization
   a. Work stealing tames tail performance
4. Kubernetes-based Scheduler as a Service in progress

https://github.com/atlarge-research/opendc

https://atlarge-research.com/pdfs/exde_fgcs23.pdf

Further Reading

[this work] Talluri, S., Herbst, N., Abad, C., De Matteis, T., & Iosup, A. (2024). ExDe: Design space exploration of scheduler architectures and mechanisms for serverless data-processing. Future Generation Computer Systems, 153, 84-96.

[related work on scheduler APIs] Manterola Lasa, A., Talluri, S., De Matteis, T., & Iosup, A. (2024, May). The Cost of Simplicity: Understanding Datacenter Scheduler Programming Abstractions. In Proceedings of the 15th ACM/SPEC International Conference on Performance Engineering (pp. 166-177).

[using the tools] Talluri, S., Herbst, N., Abad, C., Trivedi, A., & Iosup, A. (2023, May). A Trace-driven Performance Evaluation of Hash-based Task Placement Algorithms for Cache-enabled Serverless Computing. In Proceedings of the 20th ACM International Conference on Computing Frontiers (pp. 164-175).

[reference architecture] Andreadis, G., Versluis, L., Mastenbroek, F., & Iosup, A. (2018, November). A reference architecture for datacenter scheduling: design, validation, and experiments. In SC18: International Conference for High Performance Computing, Networking, Storage and Analysis (pp. 478-492). IEEE.

https://atlarge-research.com/pdfs/exde_fgcs23.pdf