Data Management to Promote Near-Data Processing

Francieli Zanon Boito

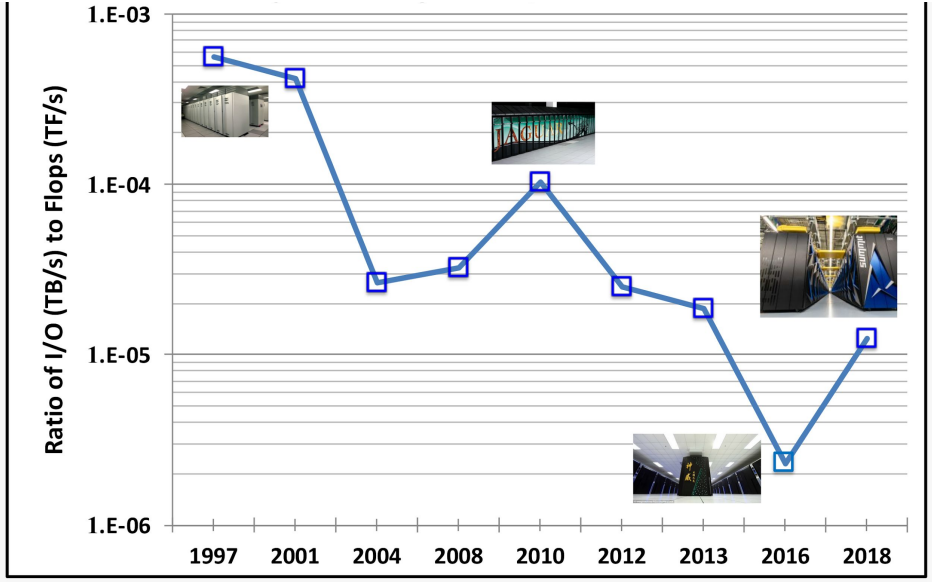
Marie Sklodowska-Curie Fellow, Inria Grenoble

DAMA project - "extreme scale data management" (since November 2018)

August 2019

extreme-scale DAta MAnagement

- High performance computing (**HPC**)
  - Scientific applications (compute intensive)
  - Data access by parallel file systems
  - Applications send requests (file, offset, length)
- **I/O is a bottleneck**
- Evolution (convergence of HPC and Big Data)
  - Scientific workflows (heterogeneous tasks)
  - NVRAM in the processing nodes



extreme-scale DAta MAnagement

- **Optimizations depend on applications' characteristics**
- Stateless I/O stack: difficult adaptation
- Multiple **applications share the machine**
  - Each application for itself: contention!
- **Our goal: global data management**
  - Remove the responsibility from applications
  - Provide high-performance data access

DAMA - Outline

- **WP1** - Data Management (**Performance**)
  - Reinforcement learning to adapt optimization techniques [Bez et al. 2019a]
  - Data replication to promote near-data processing
- **WP2** - **Intelligence**
  - Pattern matching to detect the current access pattern [Boito et al. 2019a]
  - Characterization from Darshan traces, classification techniques [Bez et al. 2019b] [Pavan et al. 2019] [Bez et al. 2019c]
- **The project is ending after 10 months** (I'm becoming an associate professor @ Université de Bordeaux)

Data Management to Promote Near-Data Processing

Francieli Zanon Boito

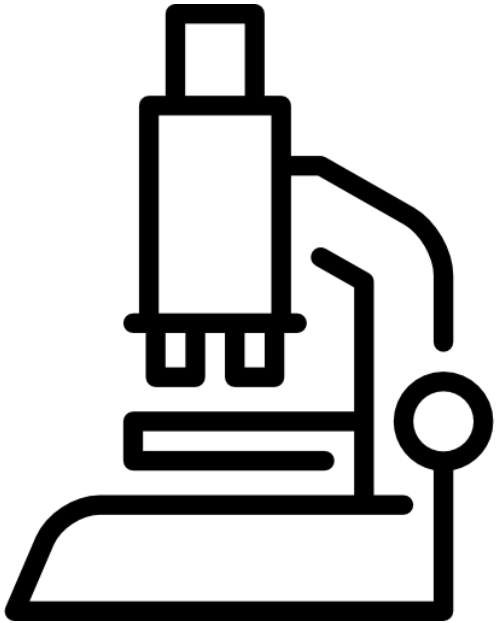
Marie Sklodowska-Curie Fellow, Inria Grenoble

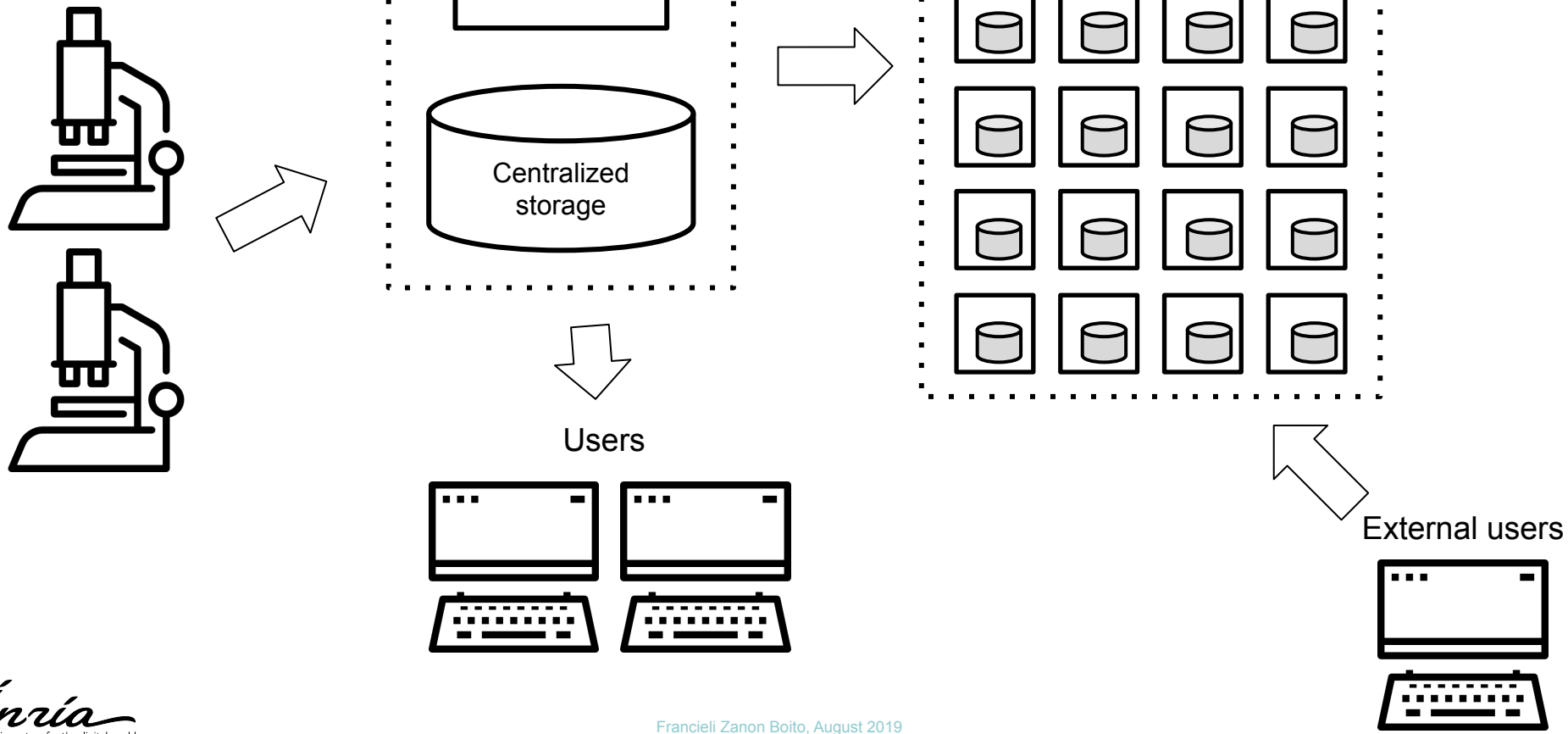
DAMA project - "extreme scale data management" (since November 2018)

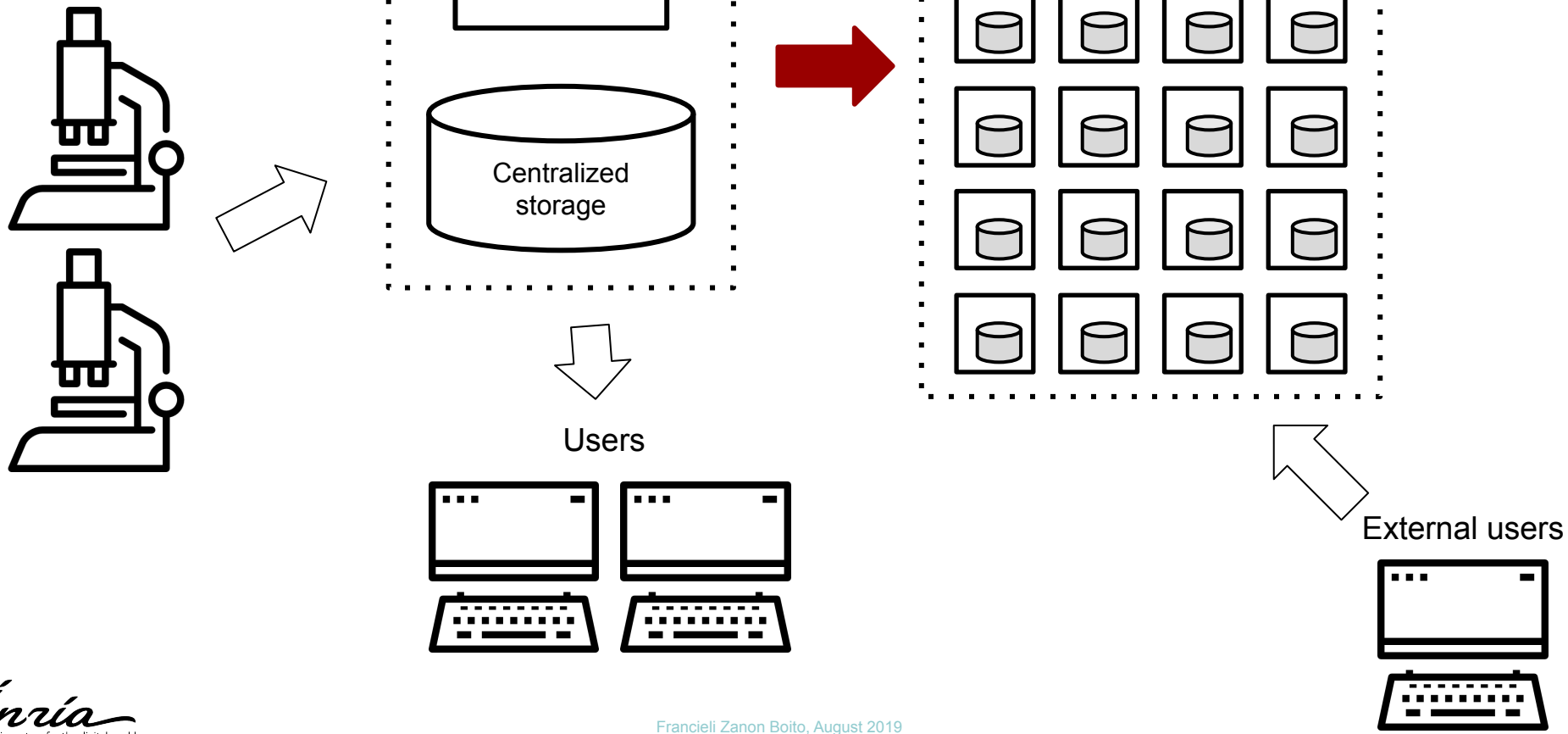
August 2019

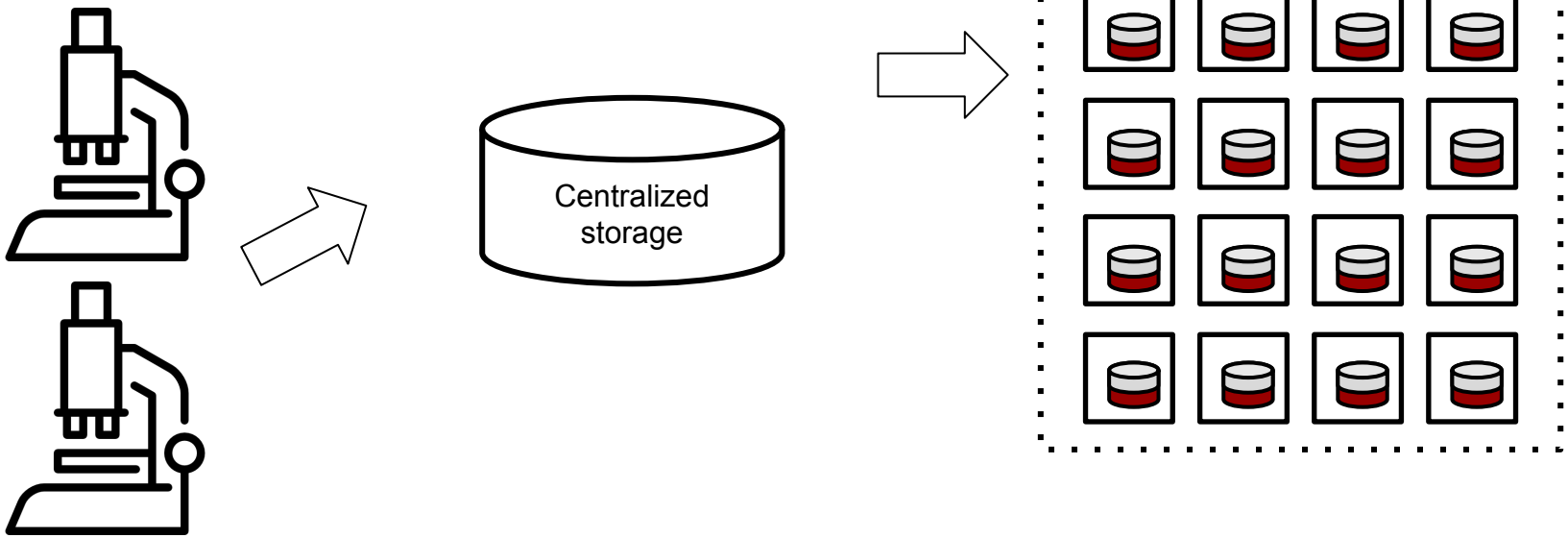
Motivation

- The context: **Materials Research** @CEA
- The problem: to manage data generated by lab equipment
- Researchers copy results to USB keys and **analyze them later**
  - Processing power can be a problem to some users
  - Results often guide future experiments (reaction delay)
  - **Reproducible research** is hard
- 1-year postdoc in the EoCoE project [Boito et al. 2019b]




"framework"
Centralized storage
Processing nodes
Users
External users


"framework"
Centralized storage
Processing nodes
Users
External users



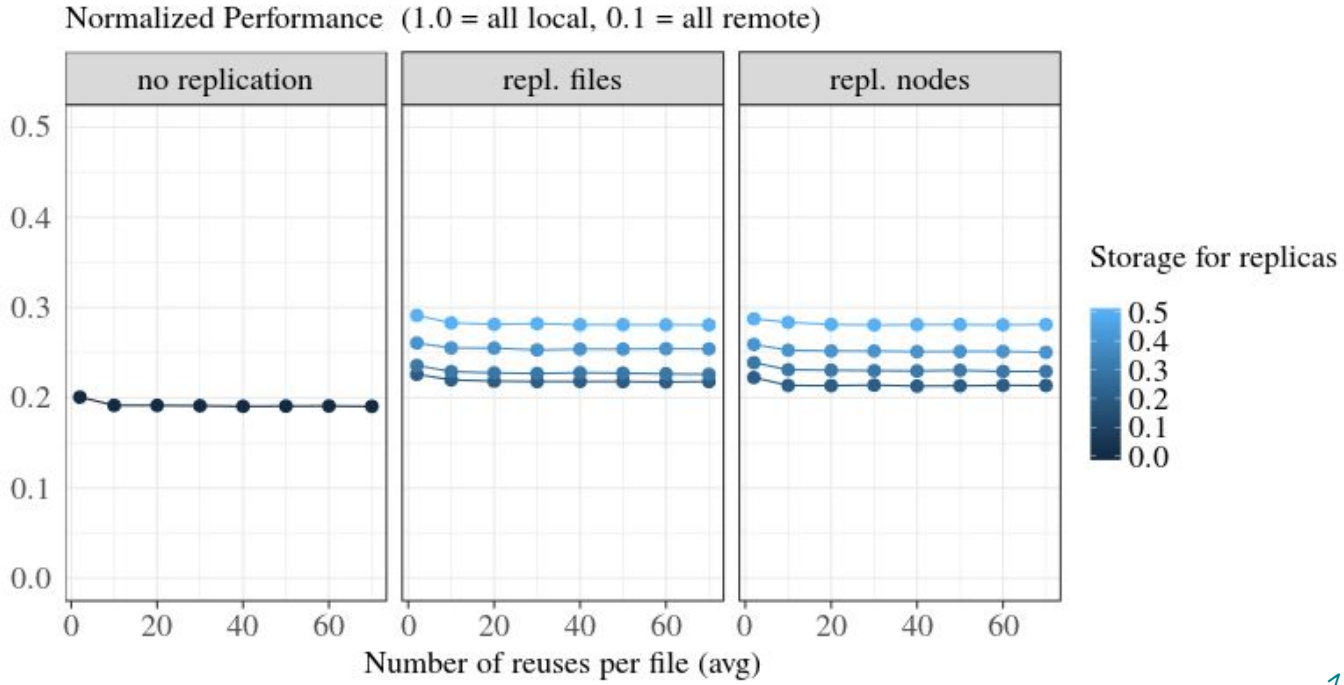
- Most accesses are to recent data
- Each file is going to be accessed a few times
- Files are immutable
- Local storage devices are not critical for users

In a nutshell

- Idea: to use a part of the local devices as a **cache** for the framework
  - Prefetch new files
- However: files in local devices are **not always available**
  - Node is being used by other processing task
  - Node is being used by other jobs outside of the framework
- **Replicate** to increase "hit ratio"
  - Increase data locality (hence performance)
  - But **cost** is important too!

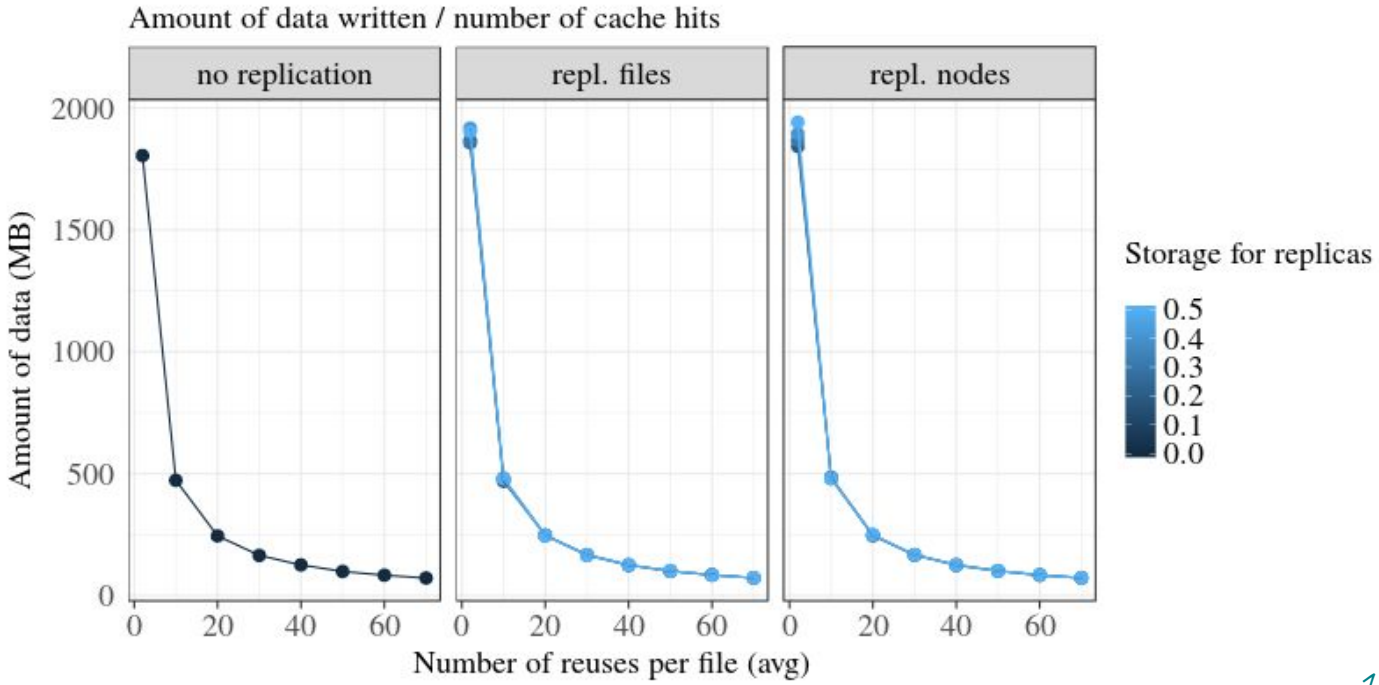
Simulations with static decisions

- Storage capacity is twice the total amount of data
- Accessing the local device is 10 times faster than accessing the centralized storage



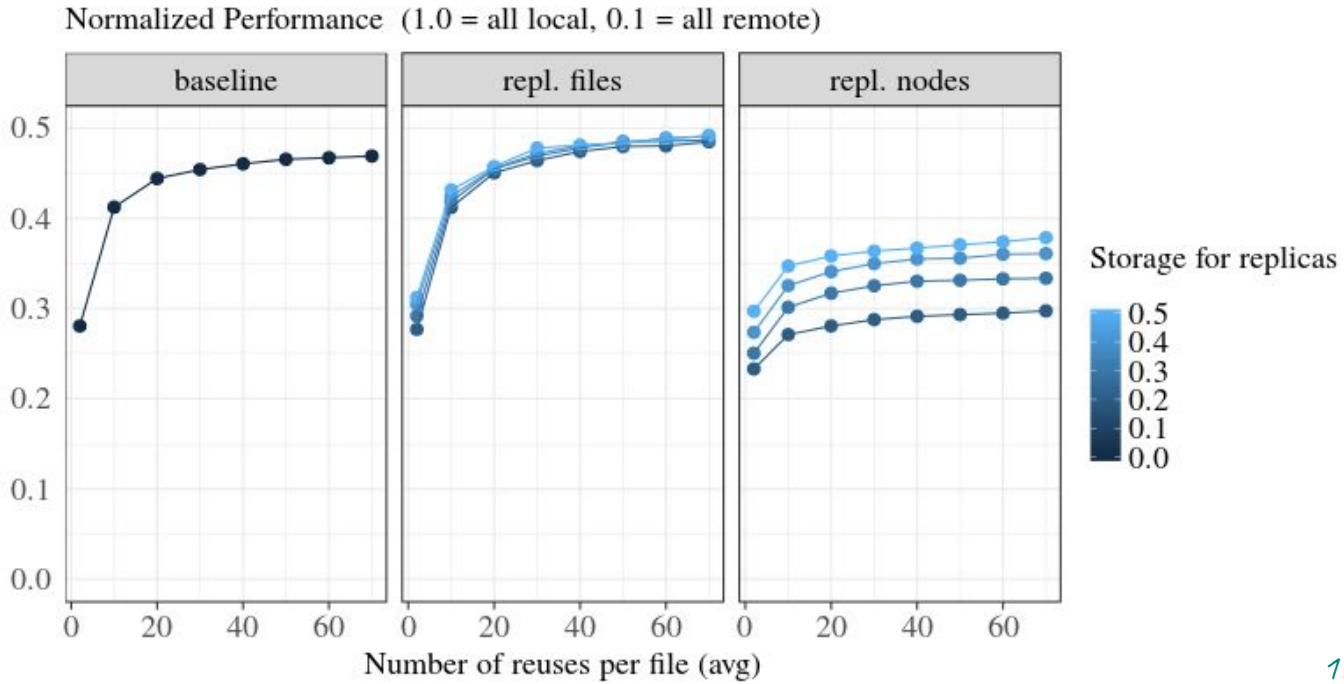
Simulations with static decisions

- **Cost** = amount of written data / number of tasks with local access



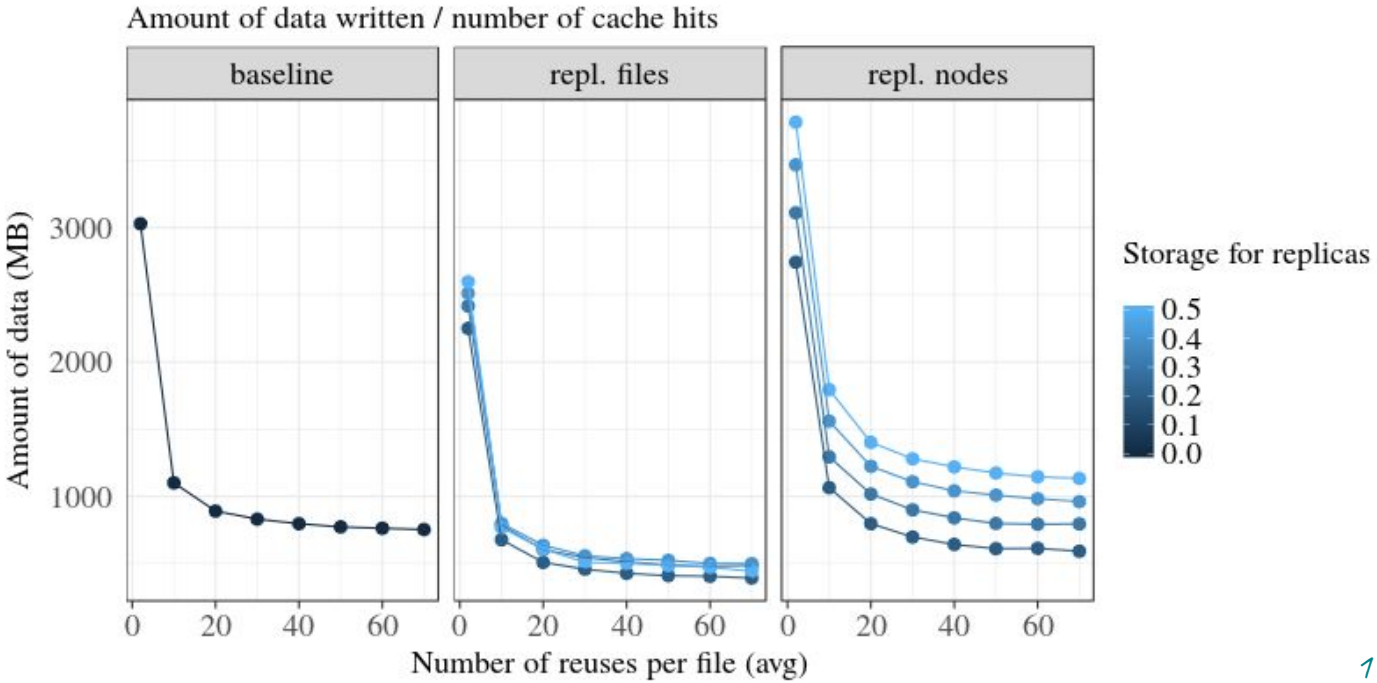
Simulations with dynamic decisions

- **File score** = #accesses / #replicas
- Evict the lowest scores (from each node), replicate the highest scores (global)
- Eventually reset the number of accesses
- Baseline: keep file in local cache after remote access



Simulations with dynamic decisions

- Baseline with 70 reuses: ~430GB written per node, file replication with 70 reuses (20%): ~230GB
  - The dataset has ~100GB, 10 nodes, each with 20GB local cache
- **Cost is ~48% lower with file replication (20%)**



What comes next?

- To refine our replication policies
- Can we use some information about the workload? **Looking for real usage data**
- To use data provenance to avoid data transfers (re-generate it instead)
- Developed the RepliSim simulator https://gitlab.inria.fr/frzanonb/replisim
  - Move on to Wrench? https://wrench-project.org/

References

[Bez et al. 2019a] J. Bez, F.Z. Boito et al., "Adaptive Request Scheduling for the I/O Forwarding Layer", under review. https://hal.inria.fr/hal-01994677

[Boito et al. 2019a] F.Z. Boito et al., "On server-side file access pattern matching", HPCS 2019, best paper award. https://hal.inria.fr/hal-02079899/

[Bez et al. 2019b] J. Bez, A. Carneiro, P. Pavan, V. Girelli, F.Z. Boito et al., "I/O Performance of the Santos Dumont Supercomputer", IJHPCA 2019. https://hal.inria.fr/hal-02270908

[Pavan et al. 2019] P. Pavan, J. Bez, M. Serpa, F.Z. Boito et al., "An Unsupervised Learning Approach for I/O Behavior Characterization", SBAC-PAD 2019, accepted for publication.

[Bez et al. 2019c] J. Bez, F.Z. Boito et al., "Detecting I/O Access Patterns of HPC Workloads at Runtime", SBAC-PAD 2019, accepted for publication.

[Boito et al. 2019b] F.Z. Boito et al., "Instrumental Data Management and Scientific Workflow Execution: the CEA case study", MPP 2019 (IPDPS workshop). https://hal.inria.fr/hal-02076963

Data Management to Promote Near-Data Processing

Francieli Zanon Boito

Marie Sklodowska-Curie Fellow, Inria Grenoble

DAMA project - "extreme scale data management" (since November 2018)

August 2019