#15

Distributed and Dependable Software-Defined Storage Control Plane for HPC

Mariana Miranda
HASLab, INESC TEC & University of Minho
mariana.m.miranda@inesctec.pt

Supervisors: ***João Paulo*** *and* ***José Pereira*** *(HASLab, INESC TEC & University of Minho)*

Motivation

Challenges in HPC storage:

- **High I/O interference**
  Numerous applications compete over HPC's storage resources, causing I/O interference and performance degradation.

- **Complex I/O stack**
  HPC infrastructures' complexity hinders the end-to-end control of I/O flows and the enforcement of global optimizations.

Problem Statement

- Leverage Software-Defined Storage (SDS) to mitigate the storage issues of HPC systems.
- SDS decouples the control layer and data storage into:
  - **Control plane:** logically centralized entity with system-wide visibility that defines the control logic.
  - **Data plane:** applies the control logic defined by the control plane over the I/O flows of applications.
- The **control plane design is often overlooked**, with limited consideration for its scalability and dependability.

Proposed Work

- Produce a **scalable and dependable control plane** suitable for HPC infrastructures.
- Provide **control algorithms** to deliver accurate enforcement strategies at the storage infrastructure, such as I/O prioritization, bandwidth guarantees, latency control, and routing.

Optimize **HPC storage resources** with a **holistic orchestration** and management.

Prototype

- The current prototype follows a hierarchical design with global and local controllers. It is integrated with a state-of-the-art data plane solution[1].
- The **global controller** has system-wide visibility and orchestrates storage services by collecting monitoring metrics and enforcing new policies.
- The **local controllers** are responsible for managing locally deployed stages, by disseminating requests from the global controller to the stages and aggregate results, offloading some of the global controller's work.
- Initial testing showed that it can enforce simple control algorithms (*e.g.,* limit *App1* metadata to *X* IOPS) and manage up to 1,000 compute nodes.

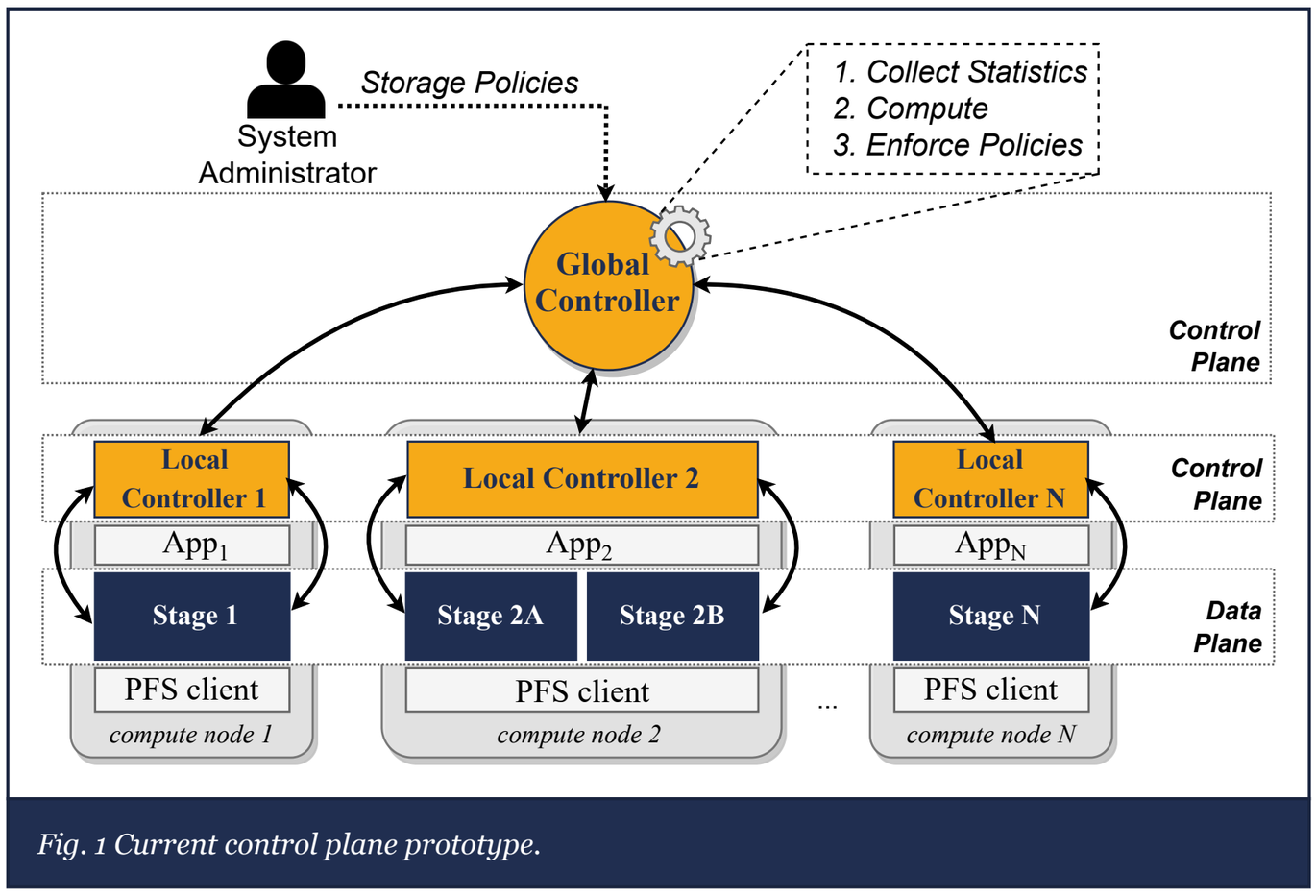

Fig. 1 Current control plane prototype.

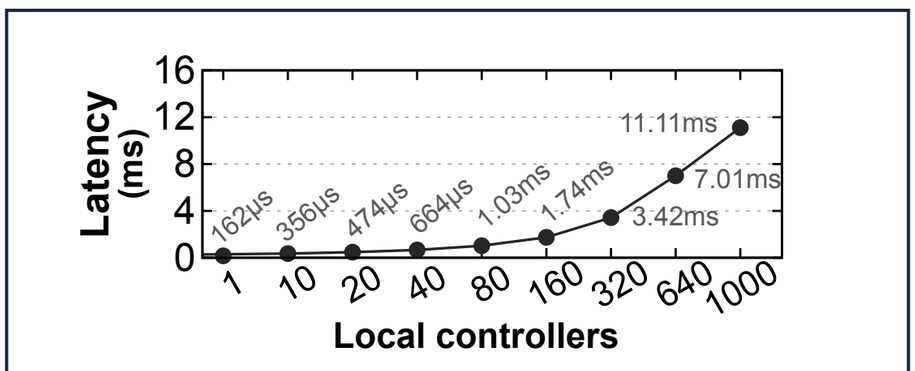

Fig. 2 Average latency of control cycles (in global controller) when the number of local controllers increases.

Moving Forward

- Assess the scalability limits of the current prototype through further testing.
- Research and explore ways to expand the solution to meet the scale requirements of HPC.
- Examine fault-tolerance protocols and existing solutions to enhance the dependability of the control plane design.
- Explore new use cases and control algorithms.

This work was financed by the FCT – Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) through Ph.D. grant PD/BD/151403/2021, and realized within the scope of the project BigHPC – POCI-01-0247-FEDER-045924, funded by the ERDF - European Regional Development Fund, through the Operational Programme for Competitiveness and Internationalization (COMPETE 2020 Programme) and by National Funds through FCT, I.P. within the scope of the UT Austin Portugal Program.

[1] R. Macedo, M. Miranda, Y. Tanimura, J. Haga, A. Ruhela, S. Lien Harrell, R. Todd Evans, J. Pereira, and J. Paulo, "Taming metadata intensive HPC jobs through dynamic, application-agnostic QoS control", in 23rd IEEE International Symposium on Cluster, Cloud and Internet Computing (CCGrid 23).

Scan the QR code to download the **poster and full paper**