

# Cover Letter

We thank the reviewers for their hard work and genuinely helpful suggestions. In addition to this cover letter, we have posted a PDF online explicitly showing additions in **red** and deletions in **blue**. Three issues were common across reviewer feedback:

## I. NO EXPERIMENTAL SETUP (REVIEWERS 1, 2, AND 3)

We apologize for the omission of experimental setup and source code details. For the experimental setup, we have added details about the cluster we ran on in Table III as well as a description in the text (addresses Reviewers 1 and 3). For the source code, we have added code snippets to the Evaluation to concretize what was actually done (addresses Reviewer 1).

Additionally, we have added paper artifacts online. For the experimental setup, we have made the infrastructure code available and added links after each figure to show exactly how experiments are run. This infrastructure code contains scripts to deploy the system, run experiments, and gather results. This process follows the Popper Convention<sup>1</sup> [1], which aims to make research reproducible. We also make the source code available and linked in the paper.

## II. STRUCTURE AND LAYOUT OF EVALUATION (ALL REVIEWERS)

One pervasive issue across reviews is that the results and contributions we cite in the introduction are not validated, explained, or even mentioned in the Evaluation. To fix this, we separate out future work to make the contributions of this paper more explicit (addresses Reviewer 1) and we connect all numbers we cite in the introduction to the evaluation (addresses all reviewers). To further clarify the Evaluation, we:

- added Table IV which includes raw baseline, Cudele, and hardware limit numbers. This adds raw numbers and helps us make comparisons to hardware speeds. Ultimately, we hope this adds insight into the results (addresses Reviewers 2 and 3).
- re-organized the section to make experiments more independent and removed cross-references (addresses Reviewer 2). Experiments are now self-contained so the reader can see the effects of different API configurations (addresses Reviewer 2) and we do a better job of explaining how results are derived (addresses Reviewer 3).
- removed the “major takeaways”. We deleted the headings but left the conclusions we make as transitions (Reviewer 2). We also add insights into the results by analyzing the raw numbers we observe.

## III. SYNTHETIC AND/OR IRRELEVANT USE CASES (REVIEWERS 2 AND 4)

We add a section to the introduction that scopes the importance of storage for large scale runtimes and workflows. Parallel and distributed jobs (whether that be MPI-based or more structured runtimes like Hadoop) need parallel and distributed storage systems that keep up with the computation. Furthermore, we highlight the parallel and distributed computing themes (addresses Reviewer 4) and cast light on them from the context of our prototype. We also add the Spark discussion proposed by Reviewer 2 to the Evaluation. The goal is to show how a system such as Cudele would benefit more general runtimes. We also move the discussion about the potential for HDFS and CephFS subtrees in the same namespace and the cost of dynamically changing consistency and durability semantics on a subtree.

## COSMETIC FIXES

## REFERENCES

- [1] I. Jimenez, M. Sevilla, N. Watkins, C. Maltzahn, J. Lofstead, K. Mohror, R. Arpaci-Dusseau, and A. Arpaci-Dusseau, “Popper: Making Reproducible Systems Performance Evaluation Practical,” UC Santa Cruz, Technical Report UCSC-SOE-16-10, May 2016.

<sup>1</sup><http://falsifiable.us/>