Problem

- Debugging the massive parallel computations that run in today’s datacenters is time consuming and error-prone.
- The use of cloud computing makes application development feel more like batch jobs and the nature of debugging is therefore post-mortem.
- Developers are notified of runtime failures or incorrect outputs after many hours of wasted computing cycles on the cloud.

Contributions of Our Work

- BigDebug’s simulated breakpoint enables program state inspection without actually pausing the entire computation.
- Its on-demand watchpoints retrieve intermediate data using a dynamic guard predicate.
- BigDebug provides data provenance capability, which can help understand how errors propagate through data processing steps.
- BigDebug enables users to change program logic in response to an error at runtime through a realtime code fix feature and selectively replay the execution from that step.

Motivating Example

Alice intends to compute the average age of all college students in each year (freshman, sophomore, junior, and senior).

- She starts by reading the data into key-value pairs and then groups the records for each category.
- Once she has all related records grouped together, she computes the average and then collects the final results.

Performance Evaluation and Time Savings

Performance Evaluation

- With the maximum instrumentation setting, BigDebug takes 2.5 times longer than the baseline Spark.
- If we disable the most expensive record-level latency profiling, overhead is less than 34%, on average.

Time Saving

- BigDebug saves up to 100% time saving over baseline through runtime crash remediation.

Simulated Breakpoint and On Demand Watchpoint

- When a breakpoint is in place, a program state is regenerated, on-demand, from the last materialization point, while the original process is still running in the background.
- Alice can perform realtime code fix using code editor in BigDebug’s UI.

Crash Culprit Remediation

- BigDebug alerts Alice on the intermediate record responsible for the crash.
- These alerts turn the corresponding transformation node of the DAG to be red.

Forward and Backward Tracing

- Alice can invoke the tracing query using “Trace to Input” button to perform step-by-step backward tracing, showing all intermediate records tracing back to crash-inducing intermediate record.

Fine-Grained Latency Monitoring

- To localize performance anomalies at the record level, BigDebug reports the top “k” straggler records to the debugger UI.