I/O Performance Analysis Framework on Measurement Data from Scientific Clusters

Michelle Koo[1], Wucherl Yoo (Advisor)[2], Alex Sim (Advisor)[2]

[1]University of California Berkeley, [2]Lawrence Berkeley National Laboratory

RESEARCH GOAL

To develop an I/O performance analysis framework to identify performance characteristics in scientific applications by analyzing measurement data collected on a scientific cluster.

BACKGROUND INFO

- This project is motivated by observations that I/O performance analyses can be conducted from monitored performance measurement data from scientific clusters.
- It is challenging to diagnose I/O performance on scientific clusters because of:
  - Unforeseen scales of interaction between hardware components and nodes
  - Increasing measurement data volumes using a large number of machines
- Scientific clusters:
  - Consist of job scheduling engine where job applications are submitted and contain portions of parallel executions divided into tasks, which are assigned to multiple nodes
- Studies of I/O performance behavior were conducted on the Palomar Transient Factory (PTF) application by analyzing measurement data collected on NERSC Edison.

SOFTWARE

- Developed:
  - An I/O analysis framework to identify I/O performance bottlenecks
  - A Darshan log parser for in-depth I/O analysis
- Tools used:
  - iPython notebook and iPython widgets to create the interactive analysis tool
  - Apache Spark to compute and sort measurement data from NERSC Edison into different Resilient Distributed Datasets (RDDs)
  - Python's Matplotlib to create visualization tools of the data stored in these RDDs

APPLICATION: PALOMAR TRANSIENT FACTORY

- The Palomar Transient Factory (PTF) is a fully-automated, wide-field survey of the sky to detect transient objects.
- The PTF analysis pipeline running on NERSC Edison consists of 38 checkpoints, with each checkpoint representing a different step in the analysis pipeline.
- PTF application logs contain the timestamps of the start and end times of each checkpoint.
- These timestamps were analyzed to determine I/O bottlenecks in the PTF analysis pipeline.

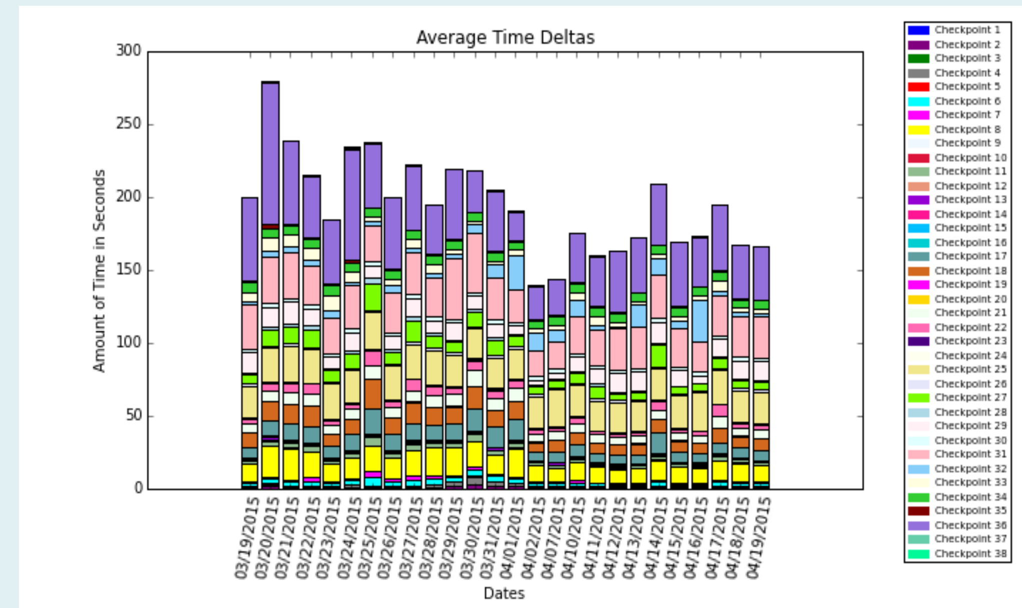

Figure 1: Shows the average amount of time in seconds that each operation takes from March 19, 2015 to April 19, 2015.

RESULTS

- Figure 1 shows the top 3 most significant checkpoints: checkpoints 25, 31, and 36.
- Top 3 checkpoints average daily percentage calculations for 64 days:
  - Checkpoint 25 ~ 11.16%
  - Checkpoint 31 ~ 14.79%
  - **Checkpoint 36 ~ 23.72%**
- Checkpoint 36 seemed to be the largest bottleneck with the largest room for performance improvement and is analyzed further in Figures 2, 3, and 4.
- Checkpoint 36 involves the Transients in the Local Universe query which consists of a table lookup.
- PTF field name 2049 was found to take the longest average amount of execution time in Figures 5 and 6.

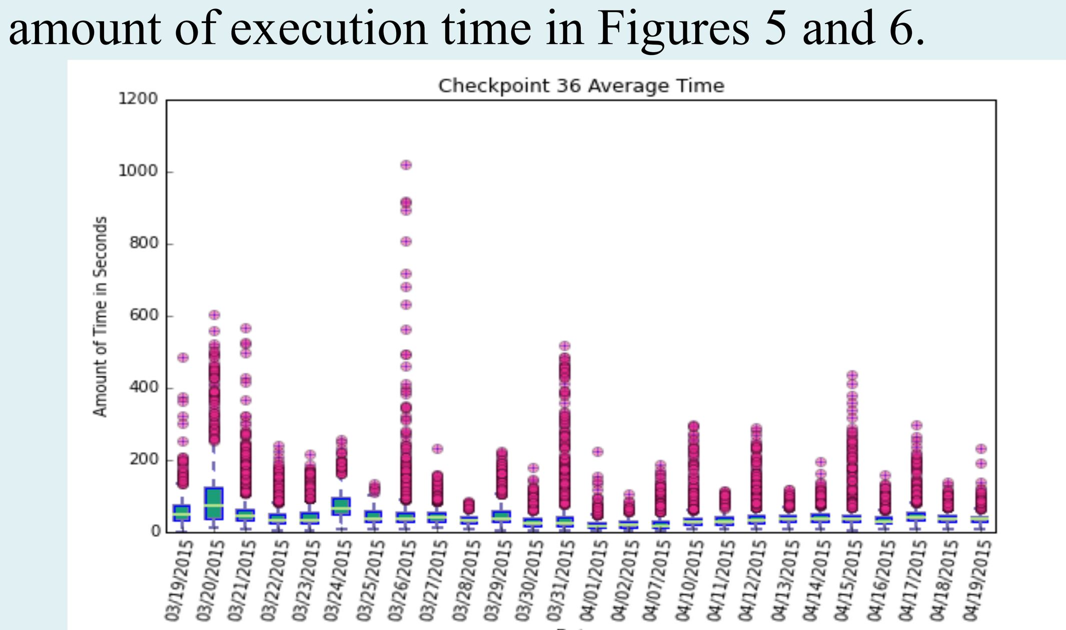

Figure 2: Shows the amount of time in seconds each timestamp of checkpoint 36 takes per day.

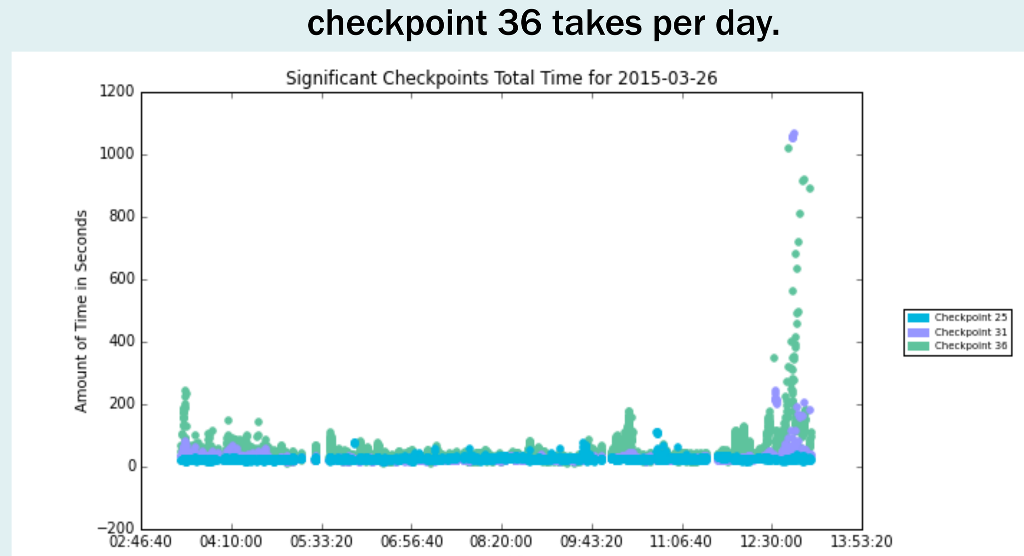

Figure 3: Shows all of the timestamps and the total execution time of Checkpoints 25, 31, and 36 throughout March 26, 2015.

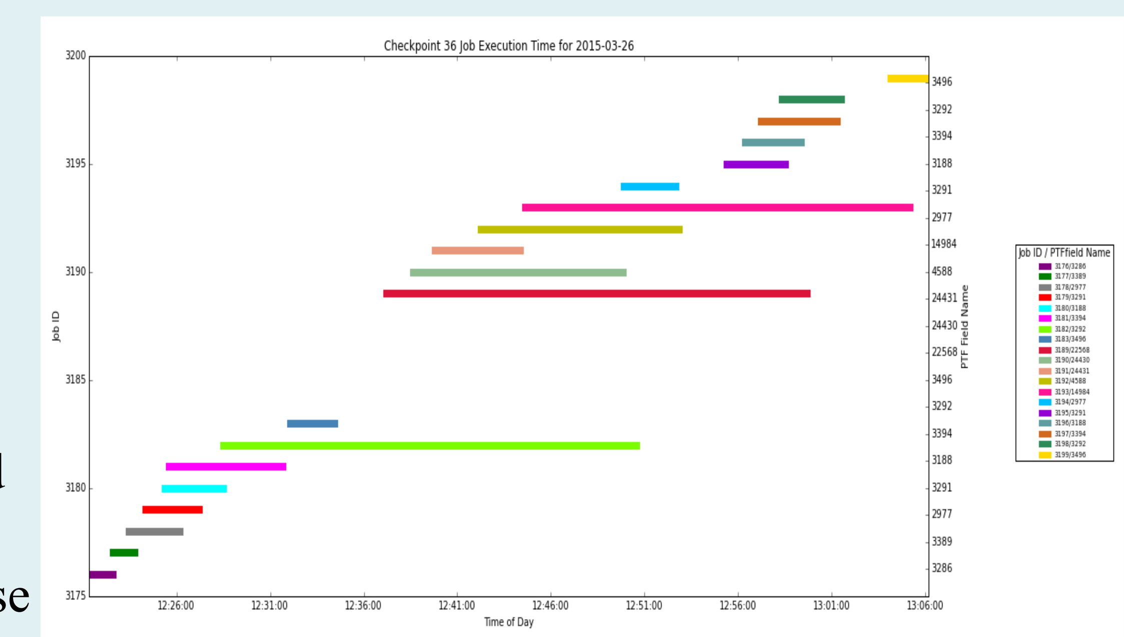

Figure 4: Illustrates the time execution of all jobs and their corresponding PTF field names on March 26, 2015.

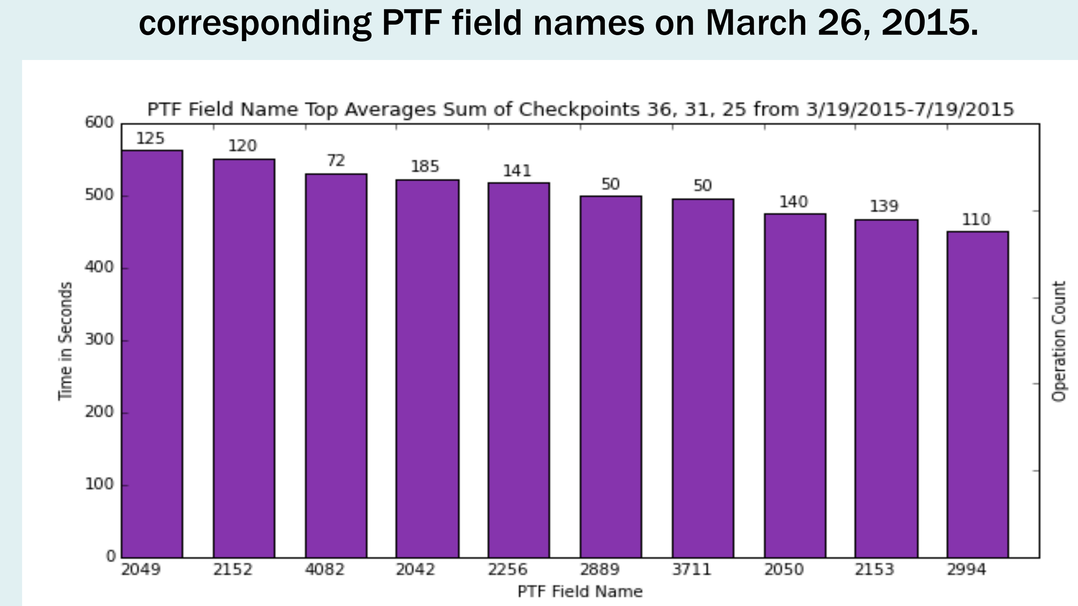

Figure 5: Shows the PTF field names of the average longest time execution of the sum of checkpoints 25, 31, and 36.

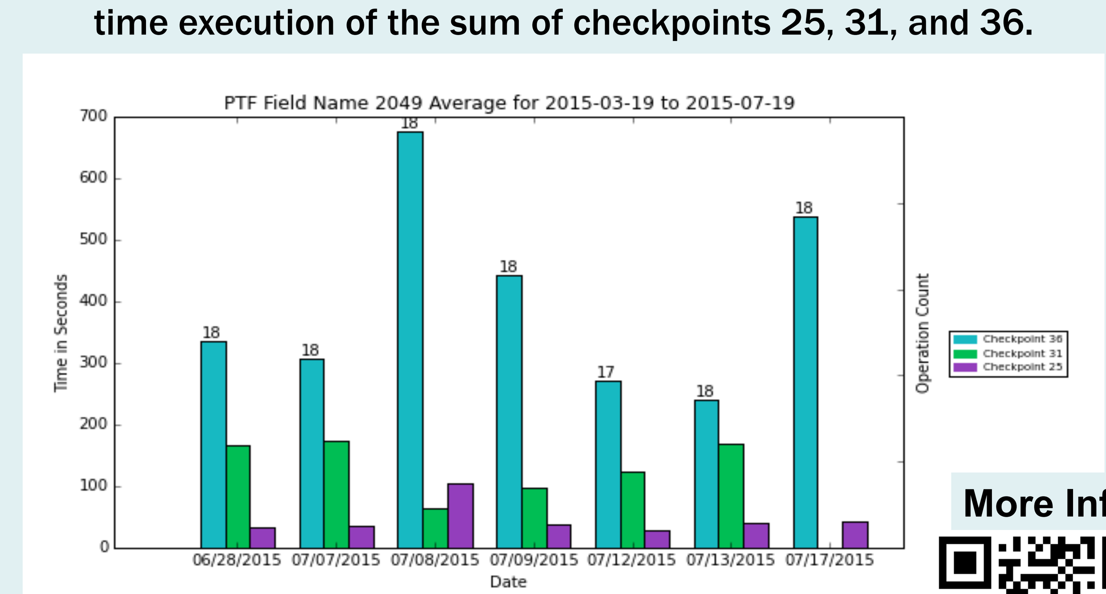

Figure 6: Shows the amount of time on average PTF field name 2049 takes per day for checkpoints 25, 31, and 36.

More Info

FRAMEWORK TOOL

- Developed an interactive visualization tool to determine I/O performance bottlenecks in scientific applications running on scientific clusters.

Figure 7: A screenshot of the I/O Analysis Framework that was developed to identify I/O performance bottlenecks.

CONCLUSION

- I/O analysis framework revealed Checkpoint 36 and PTF field name 2049 as major bottlenecks uncovered in the PTF analysis pipeline
- Scientists suggest TILU query accesses "local" galaxies, which results in a high hit rate, but as the PTF field counts increase, execution time slows, possibly causing this bottleneck
- Important PTF observations:
  - Figure 3 time execution build up shows a possible system anomaly
  - Figures 5 and 6 show PTF field names that need to be optimized
- Developed interactive I/O analysis framework tool:
  - Allows users greater autonomy
  - Helps analyze I/O performance characteristics
  - Helps identify I/O performance bottlenecks
- In the future:
  - Optimize identified I/O bottlenecks in the PTF analysis pipeline
  - Apply I/O analysis framework tool to other scientific applications for I/O behavior analysis

I would like to thank my mentors Alex Sim and Wucherl (William) Yoo, and also John K. Wu, Peter Nugent, and Yi Cao for their support and feedback. This work was supported by the Office of Advanced Scientific Computing Research, Office of Science, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.