

CEPH GSOC PROPOSAL: SHUBHAM MISHRA

Teuthology Scheduling Improvements

Mentors: Josh Durgin and Neha Ojha

Basic Information

- **Name:** Shubham Mishra
- **City:** Suri, West Bengal
- **Country:** India
- **Time Zone:** Asia/Kolkata (UTC+5.5)
- **Email Address:** smishra99.iitkgp@gmail.com
- **Github:** grapheo12

Project Overview

Project Description

Ceph is a unified, distributed storage system designed for excellent performance, reliability and scalability. Ceph has an extensive testing framework called Teuthology. Teuthology attempts to solve the problem of testing in a highly distributed and scalable setup.

Currently, tests are run by Teuthology in the following way: A Beanstalk priority queue is kept for the jobs. For testing, `teuthology-schedule` command is run with a designated job to run (see the QA suites. This schedules the job in the queue. Multiple worker processes are then run, which consume these jobs from the queue and run the job processes.

These worker nodes compete against each other, to get a lock on the number of nodes required for the jobs they are running. This means that the jobs with lower priority often starve. This creates problems at scale.

The aim of my project is to replace this multiple worker mechanism to a single dispatcher. This dispatcher will walk through the jobs in the queue, fairly allocate them with the required amount of nodes and run them. Currently, workers need to be specified which nodes they must lock in order to carry out their execution. Hence, if those nodes are not available, those jobs are left waiting. The proposed dispatcher will solve this issue by eliminating the competition. The jobs have already have a priority. We can readjust the priority by introducing a micro-parameter based on its node requirements. This will ensure that higher priority jobs run at first. At the same time, amongst jobs with same priority, it will ensure lighter jobs to be executed early, freeing resources as early as possible.

The dispatcher will also stop and bury jobs that have not responded within a predefined threshold time. Currently, low priority jobs can be starved by a

stream of high priority jobs. A stretch goal for the project is to implement an algorithm so that this does not happen. In this regard, the dispatcher can periodically increase the priorities.

Finally, these changes need to be incorporated into the web app for Teuthology, Pulpito.

Communication with mentors

I have communicated to my mentors over a mail thread and IRC over the last month. The mentors had given me a list of tasks which I have duly completed. The tasks were:

1. Cloning the Teuthology repository.
2. Going through the docs to understand the basics.
3. Inspecting the code to find how scheduling and dequeing of jobs are done.
4. Suggesting high level improvements to the code.
5. Reading about Ceph integration testing using Teuthology and the structure of the suites.