

CS554 Project Ideas

ZHT:Const - Eventual consistency support for ZHT

Overview

ZHT is a zero-hop distributed hash table, which has been tuned for the requirements of high-end computing systems. ZHT aims to be a building block for future distributed systems, such as parallel and distributed file systems, distributed job management systems, and parallel programming systems. The goals of ZHT are delivering high availability, good fault tolerance, high throughput, and low latencies, at extreme scales of millions of nodes. At this moment ZHT have achieved most of these goals. But it still has limitation. Currently ZHT only support strong and weak consistency among replicas. Strong consistency brings high latency, weak consistency is not reliable. Eventual consistency is a relatively better choice. Your work will be design and implement eventual consistency module for ZHT and refactor the code so to abstract consistency models into a separated module.

Relevant Systems and Reading Material

ZHT paper: <http://datasys.cs.iit.edu/projects/ZHT/ZHT-CRC-PID2666213-Final.pdf>

Project URL: <http://datasys.cs.iit.edu/projects/ZHT/index.html>

Eventual consistency: http://en.wikipedia.org/wiki/Eventual_consistency

Eventual consistency in Riak: <https://speakerdeck.com/seancribbs/data-structures-in-riak-nosql-matters-cologne-2013>

Preferred/Required Skills

Required: Linux, C/C++ (no OOP skill needed)

Preferred: Shell scripting (for experiments). Preferably finished CS550.

Evaluation and Metrics

Evaluate performance difference between different consistency model and replica numbers. Metrics are latency, throughput, scalability.

Project Mentor

Tonglin Li, tli13@hawk.iit.edu, <https://sites.google.com/site/tonglinlihome/>

Ke Wang, <http://datasys.cs.iit.edu/~kewang/>