CS554 Project Ideas

ZHT:Graph - Design and implement a graph database on top of ZHT

Overview

A graph database is a database that uses graph structures with nodes, edges, and properties to represent and store data. An example of a graph database is Neo4J. By definition, a graph database is any storage system that provides index-free adjacency. This means that every element contains a direct pointer to its adjacent element and no index lookups are necessary. General graph databases that can store any graph are distinct from specialized graph databases such as triplestores and network databases. 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. In this project you will be work on building a graph database ZHT+, which will leverage the ZHT NoSQL datastore.

In ZHT+, you need to implement DFS, BFS and PageRank algorithms and evaluate the performance with certain datasets and compare your system against Neo4j, Giraph and GraphLab.

Relevant Systems and Reading Material

- [1] ZHT paper: http://datasys.cs.iit.edu/projects/ZHT/ZHT-CRC-PID2666213-Final.pdf
- [2] Project URL: http://datasys.cs.iit.edu/projects/ZHT/index.html
- [3] Graph database on Wikipedia: http://en.wikipedia.org/wiki/Graph_database
- [4] A sample benchmark paper on graph database: http://www.vldb.org/pvldb/vol7/p1047-han.pdf
- [5] Another benchmark paper on graph database http://www.pds.ewi.tudelft.nl/~iosup/perf-eval-graph-proc14ipdps.pdf

Preferred/Required Skills

- Required: Linux, strong C/C++ programming skill (no OOP skill needed)
- Preferred: Shell scripting (for experiments).

Evaluation and Metrics

It is expected that the Graph Database will be evaluated for functionality, latency, throughput, and scalability, and be compared to Neo4J among other systems. The scales you are expected to evaluate these systems are up to 128 VMs on Amazon EC2. You need to use any one of the standard benchmark datasets used in [4] and [5] to evaluate your system and present final results.

Project Mentor

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