

Distributed Systems Project Ideas

1 Summary

This page lists some possible ideas for the final group project.

As mentioned in class, the project should be substantial enough to warrant a conference submission.

You are ofcourse welcome, and encouraged, to propose projects of your own choosing. If nothing comes to mind, or if you want to see some possible ideas to get a sense of what kind of projects are possible, then this page lists some project ideas.

Broadly, there are three main types of projects that you can choose from:

- Study a distributed system paper carefully, and implement and evaluate it. This is similar, but much harder than the first MapReduce assignment.
- Design a distributed system to tackle some new problem
- Implement and evaluate improvements to existing distributed system tools or libraries
- Compare different systems in a thorough manner

Here are a few examples from each category:

2 Study, implement, and possibly improve a state of the art system

You can pick any interesting systems paper presented at conferences such as SOSP, OSDI, NSDI, ATC, PODC, etc.

Here are a few:

2.1 TAPIR (SOSP 15)

Building consistent transactions with inconsistent replication

2.2 Distributed Shared Memory

Grappa - ATC 2015 FT is **not** implemented, so do that

2.3 Naiaid

Vector clocks, dataflow, and a whole lot more . Maybe implement it in Apache arrow?

2.4 Hybrid Logical Clocks

2.5 Balanced Consistent Hashing

2.6 Red/Blue consistency OSDI

3 New Distributed Systems

3.1 Decentralized Reddit

3.2 Something With BlockChain

Lol

3.3 Implement a Peer to Peer CDN

4 Extending Existing Systems

4.1 Totally Ordered Multicast with zeroMQ

Along with some application, key-val store etc.

4.2 Advanced caching policies for Memcached

Goal is to study and improve the memcached cache eviction implementation. Vanilla memcached uses the LRU (least recently used) policy for evicting objects. While LRU is simple to understand and usually performs admirably, it is worthwhile to look at other, more specialized algorithms for cache eviction.

One such cache policy is what is known as "Greedy dual size", which you can implement as an alternative to LRU. One of the goals of this project is to conduct a performance analysis of memcached with these different caching policies.

4.3 Implement different consistency schemes in tensorflow

4.4 Implement a custom framework in Apache Arrow

Apache Arrow is a project that allows Look at it's design, and implement a new framework on top of it to leverage the shared infrastructure. In particular, dataflow frameworks such as Naiad.

4.5 Implement a custom framework in Apache Arrow

Apache Arrow is a project that allows multiple distributed processing engines to run. Look at it's design, and implement a new framework on top of it to leverage the shared infrastructure. In particular, dataflow frameworks such as Naiad.

4.6 Implement rollback recovery for key-value stores

5 Comparing Systems

You can compare different systems for key-value storage, consensus, distributed logging, etc.

Author: Prateek Sharma

Created: 2019-02-14 Thu 18:21

[Validate](#)