

Farsite and OceanStore

OceanStore

OceanStore was a distributed file system designed in the early 2000s by the University of California Berkeley. OceanStore was designed to be a medium to archive data across multiple machines with two key features; Cryptography and Replication. Replication was done to ensure reliability by pushing copies of the same file to untrusted hosts as well as tiered trusted ones. This has issues on small scale networks however because the margin for error decreases as more machines are added. This was a concern because when the replication happens, copies of the same file may be used when another machine requests the file. If that new copy of the file is corrupted or damaged, then a set of bad files may be spread to others. OceanStore had a way to deal with this, by comparing the checksum to those that existed on the trusted nodes you will know when you're data is fragmented at all.

While other operating systems of the time focused on the software as the utility, OceanStore wanted to make the storage the utility. They even proposed a payment model where they would reimburse you in order to have storage available on your machine and vice versa for using persistent storage.

OceanStore's tiered set of nodes works closely to what we know as Peer-to-Peer file sharing today. A transfer method widely popular by BitTorrent. There were a large access to a group of machines with the same file, distributed and managed by a set of trusted nodes. This meant processing power could be nearly distributed and increases reliability.

OceanStore featured many of the key aspects of a distributed file system that came before it. Like an Access Control List for permission checking and Amoeba style versioning of files to rid itself of the shared state issue. OceanStore was a conceptual model that was eventually put into practice in a system called Pond. Pond was a product of what the OceanStore system would be like but focused on a much smaller scale. Pond also featured small improvements such as using MAC addresses cached in memory to increase performance overall.

Farsite

Farsite was a distributed storage solution with the intended purpose of improving file availability, and creating a failure resistant network. Farsite was originally designed to operate on a large business network, with expectations to serve up to 10^5 user systems, balancing loads between them to ensure files were reliably available, regardless of system downtime or malicious

intervention. By utilizing a byzantine network implementation, Farsite was able to operate without error serving files, so long as $>\frac{2}{3}$ of all networked systems had no errors before a rebalance. By also utilizing basic balancing algorithms, systems with higher reliability and uptime would be utilized as preferred nodes, ensuring higher availability.

These architecture choices are notable for a real distribution of Farsite for a number of reasons. Firstly, the utilization of a balancing system that focuses on system uptime appears to be a poor solution for a truly distributed system. This is due to the end result for the algorithm being a small number of highly reliable systems that would server to store the majority of all network data; effectively reintroducing the server-client architecture type that distributed solutions aim to avoid. As well as the reliance on “trusted nodes” to organize the network, which should not be a component of a truly distributed storage system, as they would again, effectively be servers to be managed. The final issue to take with the Farsite system, is that it utilizes workstation storage to share with other workstations. This is a strange choice for a file system, as even within the Farsite paper, it was noted that most users were not fully utilizing the available storage on their systems. Rather than create redundant data on a poorly utilized network of workstations, it would be more cost effective and reliable to simply give users that need more storage a larger hard drive, and keep all critical data backed up on a remote storage server.

Although the Farsite model seems to be a poor choice for deployment in enterprise environments, it is still interesting to note how it was able to implement a highly reliable file store, utilizing only consumer quality storage mediums. It is disappointing that the system wasn't developed further, perhaps to provide storage redundancy across a number of storage servers, however it does appear to not be ideally suited for this use case, as the reliance on trusted nodes to manage the network enforces a reliance on centralized servers that we wish to avoid within a truly distributed system.