Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

Distribute File System

by

Deepak Purohit, 17311921

Master of Science in Computer Science
University of Dublin, Trinity College

Introduction:

A distributed system is a collection of independent computers that appears to its users as a single coherent system.

I have managed to implement a simple Distributed File System in Python using Sockets.
Four main components are implemented in DFS

- Directory Service
- Locking
- Caching
- Replication

Directory service.
Directory Server provides a central repository for storing and managing information. Attribute-based naming systems are also known as directory services, whereas systems that support structured naming are generally called naming systems.

Locking:
The lock service is an important user tool. To avoid deadlock a client must have mutually exclusive rights on file before writing it. In the current architecture locks are stored in a list on the server. A thread has the task of checking that all the locks have an owner who is active on the server. This process runs e

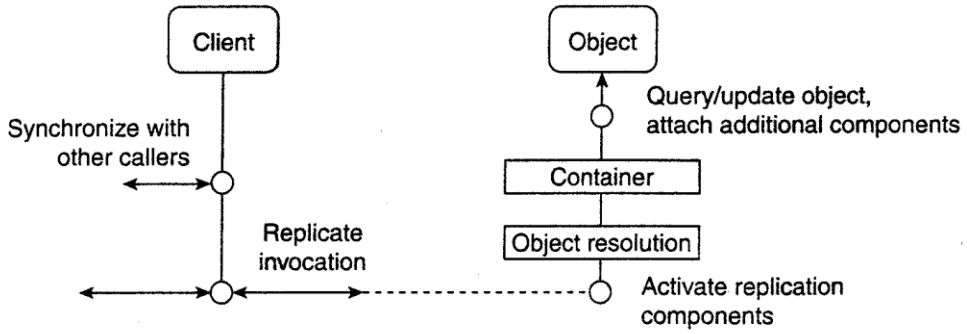
Caching: Caching is a special form of replication, although the distinction between the two is often hard to make or even artificial. As in the case of replication, caching results in making a copy of a resource.
Client -> Directory: Looking or Creating port number for a folder name

Client <- Directory: Reply with a port number if this folder name already exists or picks a server at random to now host till on

Client -> Replication Manager: Query's replication using port from directory. Replication manager holds ports of all the copies of the file. It also interacts with locking server for file locking

Replication:

In the form of performance degradation, it is generally a good idea to actually Replicate components across Distribution File System. Replication not only increases availability, but also helps to balance the load between components leading to better performance.



.

Working.

Compile Bash script that starts up a directory, locking, replication manager and file server on different ports on localhost

bash compile.sh [starting_port] [no_severs] [no_copies]

User will be prompted to enter a start port number, amount of replicant managers and how many replicas each manager will have. Starting port 8000 is recommended as Locking server is put on port 8888.

Test

python client.py [starting_port] starting port must be the same as compile.

References:

- **Distributed Systems: Concepts and Design (5th Edition) 5th Edition by George Coulouris**
- Distributed File Systems: Concepts and Examples by Levy, Eliezer Silberschatz, Abraham

  Code references:

  https://github.com/Alexis-D/DFS

  https://github.com/dramirez2/DFSystem
  https://github.com/PinPinIre/CS4032-Distributed-File-System

  https://github.com/rmccaffr/Distributed_File_System

  https://github.com/Aquila63/DistributedFileSystem
  https://github.com/zachd/distributed-systems-file-server