Distributed File Systems

SUPREET SINGH VIRDI

15TH DEC, 2017

INTRODUCTION

The task defined is to build the distributed file system using REST services in client server architecture. Client can access the centralized file system and perform various functions such as read the file, edit in file or delete a file as a basic functionality. Apart from this, there is many more functionality that can be implemented in distributed file systems. I have decided to implement the below mentioned functionalities.

* Basic Functionality (i.e. to open/read/write the file within Distributed file system).
* A Directory Services or name server, which stores the file path and allow the users to access the file.
* Locking Server is used to avoid the ambiguity in case multiple user modify same file at same time.
* Cashing is used on client side, to speed up the file system if client want to access the file he will access it again.

This is the base of the distributed file system, consist of the server, which provides the access to the file, and perform the specific functionality and there will be a client side, which provide the interface to the file system. The code is available on the Github repository (<https://github.com/supreet29/Distributed-File-System>).

OVERALL ARCHITECTURE

FILE SERVER

File server is used to server the file with GET (read the file), PUT(Create or modify the file) and DELETE files. Each file is accompanied with the lock in the file, which is used in the file to lock and be sure that user is authorized to read/make changes (edit)/ delete the file.