

**GROUP PROJECT REPORT**

**Group 8**

**Topic: Distributed File System (GlusterFS Clone)**

**Group members:**

* ĐàoBình Minh USTHBI6-094
* NguyễnDuy Minh USTHBS1-001
* ĐặngNhật Minh USTHBI7-107
* Võ Minh Đức USTHBI7-042
* NguyễnTrungHiếu USTHBI8-067

1. **Introduction**

The world keeps contributing to the increase in data everyday drastically. For data-intensive tasks such as cloud storage and media streaming, as the amount of data increases, the need to provide efficient, easy to use solutions has become one of the main issues for these filesystem. The best solution to this issue is the use of Distributed File Systems(DFS). Distributed File System are extremely scalable and cheap since it can be entirely built out of commodity Operating System(OS) and utilize common off-the-shelf hardware. It is capable of scaling to several petabytes and handling thousands of clients. It’s based on a stackable user space design and can deliver exceptional performance for diverse workloads.

Nowadays the PCs, are equipped with enormously large hard disks so the distributed file provides the means of utilizing the hard disks of various machinesin place. The project aims at providing a distributed file system that is scalable, transparent and location independent.

Brief Concepts of DFS:

A distributed file system is a client/server-based application that allows clients to access and process data stored on the server as if it were on their own computer. When a user accesses a file on the server, the server sends the user a copy of the file, which is cached on the user's computer while the data is being processed and is then returned to the server.

A distributed file system organizes file and directory services of individual servers into a global directory in such a way that remote data access is not location-specific but is identical from any client. All files are accessible to all users of the global file system and organization is hierarchical and directory-based.

1. **Objectives**

There are a few goals we have to focus on in this project:

* Build a functional DFS application.
* Improve the quality of the application over time.
* Learn from other existing similar applications.
* Able to transfer/ edit files through a connection.
* Have a deeper understanding about Distributed File System.

1. **State of the art**

The state of the art equivalent of this is the: GlusterFS system.

1. **Methods**

* The method we use for connection is through Socket/TCP.
* Each server will connect to the client and the client will connect to multiple servers.
* Client does everything (requests) and servers reply to those requests.
* Clients can Get/set/delete/list/exit from the server.
* Currently servers are not connected to each other like a real GlusterFS system, so there’s no synchronization.
* The system is similar to (maybe) Distributed or Replicated Volume system from the GlusterFS or maybe it’s similar to both of those combined.

1. **Evaluation**

* So as said from above, the system is currently able to connect to multiple servers and servers are not synchronized.
* The respond time is quite quick, though it changes from time to time but most of the time it’s fairly quick.
* It’s unknown how the system will handle high load because it’s only tested on a low number of servers and clients.
* Scalability is good, however it is manually, not automatically… Meaning if you want to scale, you have to add extra lines of codes, however the codes are similar and you can just copy paste most of them.

1. **Conclusion**

* The system is nowhere close to being called a GlusterFS Clone, however it does the basics that a GlusterFS system can do.
* Connection between servers is an issue that we can’t handle at the moment, so no sync between servers.