**Big Dinosaur in v0 version it includes following modules. It will focus on only following modules .It will use the concept of hadoop core as well as provide own algorithm as the part of big dinosaur core .**

**Trash policy will not be implemented in first version**

**Check sum will not be implemented in first version**

**conf**

**fs**

**http**

**io**

**net**

**service**

**util**

To-do & Schedule

**First Priority: Refactor Project Structure**

**Second Priority: Remove other classes that is not part of hadoop core**

**Create Configuration File.Not Done**

**Read From Configuration File :Not Done**

**Show Content of map file in Browser. Show base on block .For Block1, Block2..BlockN: Not Done**

**Show Content of reduce file in Browser Show base on block .For Block1 ,Block2..BlockN: Not Done**

**Show content of data replication for Block on another Block in browser. Show both contents to prove data is replicated :Not Done**

**Show Meta Data of Name Node .Validate name node metadata with data nodes .:Not Done**

**Show filelocation, no of replication ,block id in browser and validate with data nodes : Not Done**

**Create Block size for following pattern :Not Done**

**BLOCK\_SIZE\_DEFAULT**

**CLIENT\_WRITE\_PACKET\_SIZE\_DEFAULT**

**REPLICATION\_DEFAULT**

**STREAM\_BUFFER\_SIZE\_DEFAULT**

**Encrypt and decrypt data input stream and data output stream :Not Done**

**Create a common configuration key for local file system .It must extend common configuration key .It must contain following key**

**BLOCK\_SIZE\_KEY**

**BLOCK\_SIZE\_DEFAULT**

**REPLICATION\_KEY**

**REPLICATION\_DEFAULT**

**STREAM\_BUFFER\_SIZE\_KEY**

**STREAM\_BUFFER\_SIZE\_DEFAULT**

**CLIENT\_WRITE\_PACKET\_SIZE\_KEY**

**CLIENT\_WRITE\_PACKET\_SIZE\_DEFAULT**

**Check Sum file is size after reading and writing: Not Done**

**Create ACL for any combination of read, write and execute: Not Done**

**Perform Permission check before any execution :Not Done**

**Create ACL for a file and directory: Not Done**

Local File System Case:

Download files from one location L1 to another location L2 .The case is local file system case: Done

File Indexing

Creating a file index: Done

Searching against file index: Done

Loading a content from file indexing: Done

Loading content of files on runtime on memory: Done

Searching against search key in memory: Done

Loading data of file content in memory and storing in data structure: Done

Loading all filecontents in Tree.Each Node represents each file. Done

Searching from files on disk for first search: Done

Searching file content from memory stored as tree for second search: Done

Read files from location L1 recursively and crate list of file objects: Done

From the list of file objects in loop in local perform map reduce: Done

Break down Files into small chunks(Blocks ).The size of block is preconfigured: Not Done

Pass Small Chunks into map function: Not Done

Generate zero or more output key/value pairs: Done

Pass Map to reducer: The key/value pairs from map outputs must correspond to the appropriate reducer partition such that the final results are aggregates of the appropriately corresponding data. This process of moving map  
outputs to the reducers is known as shuffling. When the shuffle process is completed and the reducer copies all of the map task outputs, the  
reducers can go into what is known as a merge process. During this part of the reduce phase, all map outputs can be merged together to maintain their sort ordering that is established during the map phase. When the final merge is complete (because this process is done in rounds for performance optimization purposes), the final reduce task of consolidating results  
for every key within the merged output (and the final result set), is written to the disk on the HDFS.

Start Server

Receive Chunks on Server

Process Chunks

Return result

Send Small Chunks to Server to be processed: Not Done

Distributed File System Case:

In directory C:\mnt\pd0\logs\2015\07\02 there are multiple files distribute it on server .In server the case is case the logic that is implemented for local file system but it need to return value

Distribute Files on another server

Product is a collection of following features

Get the link of Your Data.

Provide 360 degree of data .So information about data on UI .Like more risk data, sensitive data

Break Down data into different block according to your preference