**CS425 MP4\_Report**

**Taipeng Liu(taipeng2) & Xiang Li (xiang14)**

**1. Design**

**Algorithm:** Our “MapleJuice” system is a MapReduce-like parallel computing framework, which is built on top of reliable Distributed File System and full-membership failure-detection protocol.

In system, we keep namenode in MP3 as global resource manager (RM), which oversees Maple & Juice task assignment and also replica file distribution. All datanodes will act as application manager (AM), finishing tasks and submit results to SDFS for next phase. We use RPC to execute each task from RM to AMs.

**Client-Function Details:**

**(1) WordCount**

**(2) ReverseWeb**

**2. Measurement**

Maple Phase:

**# MP4 Design Docs**

**## Maple-Juice Framework Design**

+ Maple Phase

\* Client input format： maple <maple\_exe> <num\_maples> <sdfs\_intermediate\_filename\_prefix> <sdfs\_src\_directory>

\* Client connects to Resouce Manager(RM), multicasts <maple\_exe> file to each designated maple workers, also sends <sdfs\_src\_directory> to RM.

\* RM will be in charge of task assignments

**## Definition**

\* **\*\*<maple\_exe>\*\*** Executable file (e.g. wordCountMaple/webLinkJuice)

\* **\*\*<num\_maples>\*\*** Client-designated worker number (if available number is less than N, occupy all; otherwise only occupy N nodes)

\* **\*\*<sdfs\_intermediate\_filename\_prefix>\*\*** Intermediate file generated by MapleTask, i.e. File with (key,value) pairs

\* **\*\*<sdfs\_src\_directory>\*\*** Source File Directory (could be up to 100Ms)

**## Code Structure**

**## Todo**

1. How to run specific functions designated by client (e.g. WordCount.go)?

    `Put the exec file into SDFS`<br>

    `Follow up: Should we multicast maple\_exe/juice\_exe to each worker nodes or put them in SDFS?`

2. Should we put sdfs\_src\_directory into SDFS?

    `Pending: No need to do this, client contact RM, which will split the src\_dir to chunks of file then sending to worker nodes`

3. How does RM partition the Mapper task to AM?

    `Removed First, count the total lines of data, get the workload for each AM; Second, use buffer to cache data and send to corresponding AMs`

    `Updated Take 10 lines as a task unit, RM scans through data (maybe a bunch of files), assign each task one by one to available worker nodes(refer to FIFO queue mechanism)`

4. How does each ApplicationMaster (AM) get file?

    `RPC or TCP Connection to get buffer then write to a local File`

5. What's the interface of Mapper & Reducer?

    `Mapper: func Mapper(fd \*FIle) pair<string, string>`

    `Reducer: func Mapper(keyValPair pair<string, string>) pair<string, string>`

6. After Mapper Phase, how does RM collect "key-value" pairs?

    `Iterate through AMs and ask them to send its intermediate files to SDFS`