Maeda Hanafi

Summary: <http://static.usenix.org/event/osdi04/tech/full_papers/dean/dean.pdf>

The paper discusses the programming model behind MapReduce. MapReduce is based off two main operators, mapping and reducing, both of which was popularly implemented in processing text before Hadoop. Mapping organizes values to keys, and reducing them would reduce the values per key by some operation. In Google's implementation of Hip-hop involves a cluster, hundreds of workstation-like machines (2-4 GB memory and a hard disk), called workers, are connected via Ethernet and serve as computation for map and reduce functions. The machines are interfaced with a scheduler responsible with delegating mapping tasks or reducing jobs to the available machines on the cluster. The scheduler, or master, receives the data to process as input and splits them into partitions. The partitions are forwarded to the mappers and the output of the mappers are forwarded to the reducers. Once the tasks are complete, the results are stored by the master. In cases where failure happens in the master, a checkpoint will indicate at which point to restart at. Failures in the mappers force restarts on their tasks, whereas failures on reducers do need to be restarted since their results are stored on a global disk, different from local.