**Anto Loyola**

N14732705

09/11/2014

**MapReduce Summary** – looks like a powerful and simple Model

In simple words, the MapReduce programming model is used to reduce the initial input data by finding some similarity and then grouping the input data using that similarity. It is made powerful for our development through the MapReduce library to which we can register our own custom Mapper and Reducer. It is quite a simple Programming Model and reminds me of the Hash Table data structure.

A Real-world use of this would be modeled as :-

* User puts part of program in master
* User places other parts of the program in mapper/reducer workers
* Master assigns worker a mapper or reducer job
* Mapper workers will have access to input data splits (granular splits) and will write to intermediate memory
* Reducer workers will have access to intermediate memory (granted by master) to reduce the Mappers output
* Reducer writes to output file
* User gets access to output file
* Faults are dealt with by repeating the process (restarting the process) – at a few words
* Master tunes the Model to be efficient by knowing when a MapReduce is close to completion

Partitioning Function can be rewritten to help user with specific cases. Also the input and output sources can easily be changed. Counters and other worker-common data can be updated to the master for every workers use.