

# SCRUTINIZING MAPREDUCE MECHANISM WITH ORIENTATION TO REFINE THE PRODUCTIVITY

Rakshitha.N<sup>1</sup>, Prasanna Kumar.M<sup>2</sup>

<sup>1</sup>M.Tech student, Department of computer science, East west Institute of technology, Karnataka, India

<sup>2</sup>Assistant professor, Department of computer science, East west Institute of technology, Karnataka, India

## Abstract

In Today's fast evolving world, number of users of internet is increasing at speed at which light travels. This is directly proportional to requirement of storage. Administering this wide range of data which is commonly called as big data is laborious. Big data is bevy of both structured and unstructured data. Dealing with this big data broach new challenges. Map reduce is the key to tackle these strenuous situations. Map reduce is the most favored evaluating methodology for colossal data processing in disseminated milieu. It is admired and widely accepted because of its outstanding features. Map reduce is hardcore in hadoop. Hadoop is the open source implementation of map reduce framework. It is used for distributed storage and processing of very large data sets on computer clusters built from commodity hardware. Map reduce is combination of map and reduce functions. Mapping and reducing the task based on slot is accomplished to improve momentum. Maps reduce fails to exploit all advantages because of unoptimized resource allocation and utilization. To unravel this we proffer to allow slots to be vigorously awarded to either map or reduce tasks depending on actual need. These designation of slots accomplished break the traditional rules and gave new dimension to overcome the loop holes of slot based methodology. This affects positively and aids to uplift performance of map reduce performance.

**Keywords:** Big data, Map reduce, Resource allocation.

\*\*\*

## 1. INTRODUCTION

Over the decades, progression in miscellaneous discipline is remarkable. In this scenario one thing that comes in to lime light is "Big data". This has become an onerous job to oversee this spontaneous increase in data day by day. Processing the request with respect to these huge amounts of data was becoming next to impossible task. This is exactly the point where hadoop came into existences. Hadoop [1] is an open source software framework for processing data sets in large amount at quick rate.

Map reduce is realization of processing in hadoop. Mapping and reducing of tasks takes place in basic unit called slot. These slots are pre configured. Assumption of slot assignment is considered as constraint.

Assumption is as follows: (1) slots are divided into map slots and reduce slots. Map slots are restricted for use for only map tasks and reduce slots are restricted to only reduce tasks. (2) Map tasks should mandatorily be executed before execution of reduce tasks.

These assumptions inversely affect the hike in performance of system and exploitation of slot utilization. While dealing with this kind of scenario we can overcome this problem by using slot based routine. This point to path adopted to complete task in slots. While considering the granting and sharing of slots as prime factor

## 1.1 Internals of Map Reduce

Map reduce is frame work which is composed of two primitives map and reduce. It consumes data set as its input and cleaves it into small lumps of data. These are processed by map tasks in totally parallel fashion outcomes of map task are furnished into reduce tasks. After execution uncompleted and failed task are again executed till completion of tasks map reduce frame work is eternally associated with key value pair in mapping function. Mapper will do tasks of mapping key value of input to intermediate key value pairs.

Mapping can begin with zero and end up to n number of pairs. In reduce functions there are three ground functions. They are shuffle, sort and reduce. Shuffle and sort occur at same instant. Input from mapping are shuffled and sorted. To consider all possibilities of input and sorting according to the specification of values and final reduction of the tasks take place which result in specific output.

Reducer will reduce all combinations and generate specific ones. This step is affected by initial input and intermediate results. So we can say that each and every parameter considered play vital role in generation of output.

## 2. LITERATURE SURVEY

[A] Improving map reduce performance in a heterogeneous cloud. [2]

Map reduce clusters in hadoop are one of the famous deployments of cluster. But in the darker side, most map

reduce implementations are designed and used for homogenous clusters which furnish low level performance on heterogeneous clusters. This paper identifies the aspects such as system configuration and task scheduling with respect to different scenario in configuration and scheduling to uplift the current features.

Benefit is that every shuffle operation is effective and has positive impact on performance. Dark side is that in early stage of shuffle resource utilization might be complex.

[B] Dynamic split model of resource utilization in map reduce. [3]

Resource utilization problem still exists in some of traditional approaches which are solved by using dynamic split model of resource utilization. Resource is allocated according to the load in the cluster and all jobs status in run time. In this scenario resource usage pipeline is used. It include buffer enlargement in map phase and requirement of slots for map task and reduce tasks. It shows mark able gain in performance.

[C] Locality aware reduce task scheduling for maps reduce. [4]

Scheduling task which consider the specific constraint which definitely improve the overall development of the system. Here scheduling of task is carried out considering the data as the main feature for scheduling. Data location is identified and highly required data for computation are collected along with size and location. This information is precious while scheduling the tasks. This methodology keeps basic level information associated with node. This provides foundation to reach desired performance.

[D] Optimizing schedule of map reduce jobs

Dealing with optimizing and scheduling of map reduce jobs posses major challenges in administering and maintaining the system properties. This work indicates that in order to reach the line in map reduces, time plays major role in order. The Order in which these map reduce tasks are scheduled and amount of time consumed for completion comes into picture which directly affects the performance.

It applies two stages, scheduling of tasks which is carried out in hand by hand fashion. So that problem of over lapping will not occur with inefficiency in utilization of required resource while consistently maintaining and exploring properties of map reduce.

### 3. PROPOSED SYSTEM

Traditional approach which is used in map reduce usage of slots is separated from both map and reduce tasks. This will guide to grievous utilization of slots which is below the average level. Path to oversee this problem is to detach from this assumption and make generalized utilization of slots. Implicit consideration of basic map reduce framework is not accepted and is modified to achieve efficiency. For this to happen we first group the slots into pools which form the base of slot based routine.

Exchange of unused slots is carried out while keeping the fairness constraint which is done not to give place for further compliance. Sharing of unused slots is done at pool levels and outside pools. Sharing at pool level indicates that each pool contains both map and reduce functions and sharing happens map and reduce and then occurs between pools.

Trading outside the pool means different pools are considered for both map and reduce functions. Map has pools ranging from 1 to n and is similar for reduce and tradeoff takes place between pools of map and between pools of reduce and finally between map and reduce functions.

One of the eminent parts of sharing slots is considering history of slots with respect to efficiency. If the slots behavior is consistent with different circumstances and yields good results and provide path to help to achieve betterment in the results.

### 4. CONCLUSION

Identification and dealing with inefficiency of the map reduce is considered as major issue. Slot where map and reduce tasks are performed are used to reach high rate in performance and assignment of tasks to slots are done dynamically. Tasks performed in slots which aim to achieve efficiency is crucial.

Slots are groped in pools and exchange of slots takes place inside pools where map and reduce functions will be a part. Exchange of slots between map and reduce functions where pools will be present inside each map and reduce phase. By doing these kinds of variations to carry out slots exchange it is possible to exploit advantage to maximum extent.

Along with this it is verifies whether slots have reached expectations. If not it will be included into unused slots and steps will be taken to repair this problem. All of these features help to uplift the implementation of map reduce.

### REFERENCES

- [1]. Hadoop. <http://hadoop.apache.org>.
- [2]. Yu S. Tan, Bu-Sung Lee, Bingsheng He and Roy H. Campbell. *A Map-Reduce Based Framework for Heterogeneous Processing Element Cluster Environments*. In Proc. of CCGRID 2012
- [3]. Z.H. Guo, G. Fox, M. Zhou, Y. Ruan. *Improving Resource Utilization in MapReduce*. In IEEE Cluster'12. pp. 402-410, 2012.
- [4]. M. Hammoud and M. F. Sakr. *Locality-Aware Reduce Task Scheduling for MapReduce*. In IEEE CLOUDCOM'11. pp. 570-576, 2011
- [5]. A. Verma, L. Cherkasova, R. Campbell. *Two Sides of a Coin: Optimizing the Schedule of MapReduce Jobs to Minimize Their Makespan and Improve Cluster Performance*. In IEEE MASCOTS, pp. 11-18, 2012.