# Chapter 31
# A Hierarchical Hadoop Framework to Handle Big Data in Geo-Distributed Computing Environments

**Orazio Tomarchio**
*Department of Electrical, Electronic and Computer Engineering, University of Catania, Catania, Italy*

**Marco Cavallo**
*Department of Electrical, Electronic and Computer Engineering, University of Catania, Catania, Italy*

**Giuseppe Di Modica**
*Department of Electrical, Electronic and Computer Engineering, University of Catania, Catania, Italy*

**Carmelo Polito**
*University of Catania, Catania, Italy*

## ABSTRACT

*Advances in the communication technologies, along with the birth of new communication paradigms leveraging on the power of the social, has fostered the production of huge amounts of data. Old-fashioned computing paradigms are unfit to handle the dimensions of the data daily produced by the countless, worldwide distributed sources of information. So far, the MapReduce has been able to keep the promise of speeding up the computation over Big Data within a cluster. This article focuses on scenarios of worldwide distributed Big Data. While stigmatizing the poor performance of the Hadoop framework when deployed in such scenarios, it proposes the definition of a Hierarchical Hadoop Framework (H2F) to cope with the issues arising when Big Data are scattered over geographically distant data centers. The article highlights the novelty introduced by the H2F with respect to other hierarchical approaches. Tests run on a software prototype are also reported to show the increase of performance that H2F is able to achieve in geographical scenarios over a plain Hadoop approach.*

DOI: 10.4018/978-1-7998-5339-8.ch031

## 1. INTRODUCTION

Technologies for big data analysis have arisen in the last few years as one of the hottest trend in the ICT scenario. Several programming paradigms and distributed computing frameworks (Dean & Ghemawat, 2004) have appeared to address the specific issues of big data systems.

Application parallelization and divide-and-conquer strategies are, indeed, natural computing paradigms for approaching big data problems, addressing scalability and high performance.

Furthermore, the availability of grid and cloud computing technologies, which have lowered the price of on-demand computing power, have spread the usage of parallel paradigms, such as the MapReduce (Dean & Ghemawat, 2004), for big data processing.

However, Hadoop, the most known open-source implementation of the MapReduce paradigm, was mainly designed to work on clusters of homogeneous computing nodes belonging to the same local area network: nowadays, more and more frequently, data are generated and stored in a geographically distributed manner, making existing frameworks such as Hadoop no longer suited to effectively process such data (Heintz, Chandra, Sitaraman, & Weissman, 2014).

The critical choice for every system that has to deal with this scenario is either moving the computation close to the data or, vice versa, moving the data to where the computation has to be done. These choices, of course, represent the two extreme possibilities of many other intermediate choices. Moving the data from different sites to a central one may increase latency introducing delay in processing time; similarly, the cost of transferring huge amount of data may be infeasible as well. On the other hand, moving the computation close to the sites where the data reside is not always possible depending on the characteristics of the processing. Data may happen to be stored in sites with very different computing capacities. Having large data to be locally processed by very low-power computing facilities turns to be a big inefficiency; conversely, using a very powerful data center to elaborate only limited amounts of data is an unacceptable waste.

In this work, we propose a Hierarchical Hadoop Framework (H2F) that overcomes the limits showed by the original Hadoop job scheduling algorithm by taking into account the actual heterogeneity of nodes, network links and data distribution among geographically distant sites (Cavallo, Di Modica, Polito, & Tomarchio, 2016). Our approach follows a hierarchical scheme, where a top-level entity takes care of serving a submitted job. The job is split into a number of bottom-level, independent MapReduce sub-jobs that are efficiently scheduled to run on the sites where the data reside.

We believe a hierarchical computing model may help since it decouples the job/task scheduling from the actual computation: this way, the compelling potentiality of Hadoop is exploited at the bottom level while the job scheduling is delegated to the top level. In our work, we introduce a novel job scheduling algorithm which accounts for the discussed inhomogeneity to optimize the job makespan. Unlike previous works, our job scheduling algorithm aims to exploit fresh information continuously sensed from the distributed computing context to guess each job's optimum execution flow.

Another enhancement we propose with respect to similar works in the literature consists in a novel approach to the study of the job's application profile, which is an important characteristic of the computing context that may strongly affect the job performance.

A prototype of the H2F system has been developed and deployed in a testbed environment: experiments carried out showed that the H2F system outperforms Hadoop in some scenarios where resources (computing capacity, data distribution, network links) are heterogeneous.