Future Generation Computer Systems 56 (2016) 773–782

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Improving reliability and performances in large scale distributed applications with erasure codes and replication

Marco Gribaudo [a], Mauro Iacono [b,*], Daniele Manini [c]

[a] DEIB, Politecnico di Milano, via Ponzio 34/5, 20133 Milano, Italy
[b] DSP, Seconda Università degli Studi di Napoli, viale Ellittico 31, 81100 Caserta, Italy
[c] DI, Università degli Studi di Torino, corso Svizzera, 185, 10129 Torino, Italy

HIGHLIGHTS

- We evaluate the performances of mixed erasure coding/replication allocation schemes.
- We model architectures with massively distributed storage.
- We show the effects of the different parameters on the performances of the allocation technique.

ARTICLE INFO

Article history:
Received 3 March 2015
Received in revised form
23 June 2015
Accepted 18 July 2015
Available online 30 July 2015

Keywords:
Performance modeling
Cloud computing and big data infrastructures
Storage systems
Erasure codes

ABSTRACT

Replication of Data Blocks is one of the main technologies on which Storage Systems in Cloud Computing and Big Data Applications are based. With the heterogeneity of nodes, and an always-changing topology, keeping the reliability of the data contained in the common large-scale distributed file system is an important research challenge. Common approaches are based either on replication of data or erasure codes. The former stores each data block several times in different nodes of the considered infrastructures: the drawback is that this can lead to large overhead and non-optimal resources utilization. Erasure coding instead exploits Maximum Distance Separable codes that minimize the information required to restore blocks in case of node failure: this approach can lead to increased complexity and transfer time due to the fact that several blocks, coming from different sources, are required to reconstruct lost information. In this paper we study, by means of discrete event simulation, the performances that can be obtained by combining both techniques, with the goal of minimizing the overhead and increasing the reliability while keeping the performances. The analysis proves that a careful balance between the application of replication and erasure codes significantly improves reliability and performances avoiding large overheads with respect to the isolated use of replication and redundancy.

1. Introduction

The request for services that are based on big computing infrastructures is flourishing, and providers with different capability and specializations compete on the market. The abundance of proposals that are founded on the availability of data center technologies and facilities causes lower margins on the revenues of providers, because big players can afford more investments with longer strategies. Be the service a simple site hosting, a virtual server, or more complex offers like cloud applications or high performance computing, the main problem of a provider is to be able to supply a given level of performances to a given number of customers, at the minimum cost.

A big part of the investments is devoted to sustain quality of services, that is the result of a number of architectural, scheduling and management issues (e.g. see [1–3]). Limiting the scope to the problem of providing a dependable storage subsystem, a big part of the investment is devoted to storage hardware and related connectivity. The quality of service specification over the storage subsystem implies the need for a consistent and safe way to ensure data persistence, that is generally based on replication strategies: data chunks are replicated over different storage units on different nodes, so that if a node or a unit fails (temporarily or permanently) the stored data are not lost and are continuously available to the owner. This solution is effective, even if it has some drawbacks:

* Corresponding author.
E-mail addresses: gribaudo@elet.polimi.it (M. Gribaudo), mauro.iacono@unina2.it (M. Iacono), manini@di.unito.it (D. Manini).

http://dx.doi.org/10.1016/j.future.2015.07.006
0167-739X/

first, the problem of replication management is not trivial, as replicas can be handled with different strategies, to find the most convenient mapping between data chunks and nodes; second, the complete replication of data multiplies the number of storage units that are needed and complicate the interconnection infrastructure, significantly increasing the costs. A smart management of the problem, dealing with the need for space so that a lower expense is sufficient to keep data integrity and availability, would make the difference between being in or out of the market.

A possible solution is to move from replication to erasure codes. Erasure code techniques reduce the need for additional space while keeping the integrity level, by substituting replicas with additional redundant data that take less space. When a redundancy technique is applied, in case of node failures lost data are reconstructed by using the additional data (that have to be properly designed to be sufficient), distributed over other nodes in the most convenient way. The drawback is that in case of failure lost data have to be rebuilt, causing a momentary computing and communication overhead to the system. Choosing a good redundancy technique and designing the best trade-off between additional space and computing needs can lead to significant savings in terms of infrastructural investments, by increasing the overall efficiency of the data storage subsystem. Such a choice is strategic, and must be as much as possible transparent to specific uses of the infrastructure; consequently, the choice should be focused on the lower layers of the system HW/SW stack.

In this paper we deal with optimal storage management. The original contribution of this paper is a simulation technique that supports, in terms of dependability and overall performance evaluation, the design of a strategy that exploits the existing resources by means of a redundancy-based approach that uses erasure codes, to help designers in predicting their best trade off between space and computing resources. We show how it is possible to model the storage subsystem with its management layer by means of an event based simulator that showed to be able to scale up to different architectures. More in detail, in this work we focus on techniques that address data availability and durability in distributed storage systems such as Hadoop Distributed File System (HDFS) [4], Google File System (GFS) [5], and Windows Azure [6]. In these systems, the occurrence of failures requires the storage of more replicas for each data block on more nodes, to guarantee that a useful number of copies are always available. To improve the tolerance of such systems when data are lost due to a node fault, many replication schemes have been introduced. Taking inspiration from the one presented in [7] we propose a new idea in order to achieve the best trade-off among data persistence and overall storage load distribution to avoid non-optimal resources utilization. We implement a simulator able to evaluate this approach and efficiently determine the system parameter settings. We propose a framework that allows tuning the system configuration to achieve the best data availability.

The paper is organized as follows: Section 2 discusses related works; in Section 3 we provide a brief background on replication and erasure coding techniques to improve the fault tolerance of the system and describe the proposed model; in Section 4 we describe the ad hoc simulator. Results are then presented in Section 5, followed by conclusions.

2. Related works

For a general introduction to performance and dependability modeling of big computing infrastructures the reader can refer to Castiglione et al. [8], Barbierato et al. [9], Castiglione et al. [10], Cerotti et al. [11], Xu et al. [12], Yan et al. [13], Barbierato et al. [14], that specifically deal with storage problems; for an introduction to the problem of information dependability in distributed storage systems, we suggest Distefano and Puliafito [15].

More specific references have been very useful for this work. The impact of replica placement in distributed systems is analyzed in [16]. In [7] a highly distributed backup storage system is analyzed, based on nano datacenters to save costs, and the reconstruction process of multiple lost chunks is analytically modeled with its correlations, providing a good reference for a characterization in terms of distributions. The main issue is transferring the lost copies on the remaining nodes according to different policies (random, less load and power of choice), and by tuning the window size that determines which neighbors can be selected to minimize the data loss rate. In our work we aim to improve the reliability of the system by grouping blocks in groups composed of different entities, each stored in a different node. For each group, a number of redundancy blocks are created, the redundant information is stored using network coding techniques.

A first quantitative comparison between erasure coding and replication in self-repairing and fault resilient distributed storage systems can be found in [17]. The solution has been explored in peer to peer systems: in [18] a family of erasure codes with a very low overhead factor is applied to distributed storage to show the appropriateness of this approach; in [19] the problem of lost data blocks reconstruction in distributed storage systems is studied in peer to peer architectures, providing an interesting characterization in terms of empirical distributions obtained by means of event based simulations; a further application is in [20]; an analysis of the combined application of redundancy and erasure coding in DHT is given in [21]; in another analysis [22] the authors evaluate the performances of DHT by an analytical approach based on traces of 3 different applications, and conclude that results are actually confirming the expected advantages, but cost an excess of complexity in the design of the overall system because of the implementation of erasure coding.

More recently, the same solution has been considered for application in Big Data applications: in [23] an application of erasure codes to distributed file system is presented, providing an analytical upper bound of the average service delay due to the network and the needed computations: the paper specially points out the role of the factors that generate latency as a consequence of erasure coding, and their optimal management; in [24] another family of erasure codes is proved to be applicable to Big Data systems.

The work presented in [25] is the most similar proposal to the solution analyzed in this paper. The authors propose a two-phase technique, namely Replicated Erasure Codes, that combines the storage space efficiency of erasure codes and the repair traffic efficiency of replication. The work is supported by an analytical evaluation of its basic characteristics, which are studied in a range of situations by means of a model, based on some simplification assumptions, and a simulation based on a peer to peer node availability trace. With respect to Friedman et al. [25], this paper is more focused to analyze the impact of node failures rather than chunk or file failures, and to provide an optimal selection strategy for the allocation of redundancy and coding data. Consequently, the repair mechanism privileges a reactive approach, considering that bandwidth is a secondary problem in data centers with respect to data integrity and optimal storage management, specially when the target is Big Data. Moreover, our analysis is based on a simulation that focuses on situations that are more likely to arise in real data centers, and also includes transients.

3. The storage model

The simplest replication strategy corresponds to storing several different copies of the same data chunk, allowing a system to be