Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 109C (2017) 18–25
Procedia Computer Science
www.elsevier.com/locate/procedia

The 8th International Conference on Ambient Systems, Networks and Technologies (ANT 2017)

# Data Replica Placement Mechanism for Open Heterogeneous Storage Systems

X. Xu[a,b,*], C. Yang[a], J. Shao[c]

[a] School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China
[b] State Key Laboratory of Information Security, Chinese Academy of Sciences, Beijing 100093, China
[c] Institute of Big Data Research at Yancheng, Nanjing University of Posts and Telecommunications, Yancheng 224005, China

**Abstract**

Many big data-oriented network storage platforms are open and heterogeneous, which means that the data placement mechanism has a great impact on the system performance. The data replica placement mechanism needs to consider the data availability, the load balance and the quality of service for open heterogeneous storage systems. However, current data placement mechanisms are mainly for simple homogeneous storage systems, not entirely suitable for heterogeneous storage systems. We proposed a novel data replica placement mechanism with the adjustable replica deployment strategy (ARDS) for open heterogeneous data storage systems, which considers the data availability, the data access frequency and the storage capacity. The mechanism takes the data availability as premise of the initialization number of replica, then adjusts the number of replica according to the data access frequency, and places replicas based on the node spare capacity, which effectively avoids the condition of load unbalance. We implemented simulation experiments based on OMNet++. The experimental results show that the ARDS-based mechanism improves the system performance including the system load balance and the rate of successful data access.



*Keywords:* Data Replica, Replica Placement, Heterogeneous Storage Systems

## 1. Introduction

Internet of Things, Social Networks and Mobile Computing are becoming more and more popular. The storage platform of Big Data is becoming more and more open and isomerized. The data replication mechanism of the storage system will have a great impact on system performance[1]. The key points involved in the placement of data replicas in heterogeneous storage systems include the determination of the number of data replication and the location of storage[2]. The number of replicas has great impact on the data availability of distributed storage systems. Few replicas could easily lead to overheating of partial replicas and overload of storage nodes. If the number of replicas is large,

* Corresponding author. Tel.: +8613813885172; fax: +862585866433.
*E-mail address:* xuxl@njupt.edu.cn

1877-0509 © 2017 The Authors. Published by Elsevier B.V.
Peer-review under responsibility of the Conference Program Chairs.
10.1016/j.procs.2017.05.290

the storage resources will be wasted. Heterogeneous storage systems choosing those nodes with different performance and the different replicas storage locations affects the quality of service when replicas are accessed.

The current replication placement strategies can be divided into the source request placement strategy, priority placement strategy, path placement strategy, neighbor node placement strategy, and random placement strategy, etc.[3,4,5]. Lightweight adaptive replication is a typical source request placement strategy. The advantage is that the new replication creation mechanism is triggered when the existing nodes that reach the placement threshold, which reduces the creation of redundant replicas, thus reducing the cost of replication creation and maintenance. The disadvantage is that the load of the storage node is easily exceeded, resulting in the unbalance of the overall system load. The advantage of priority placement is that when an access request arrives at the target storage node, the other storage nodes will transfer the same replication to the visited node. The advantage is that the number of storage nodes is reduced. The disadvantage is that the hot spots are easily generated and the overall system load is unbalanced. The path placement strategy is a simple and convenient method to query all nodes on the request path when users access the replication, but it is easy to cause the data redundancy, which increases the waste of storage resources and the maintenance cost of replication consistency. The neighbor node placement is mainly to save the history of the replication access. When a certain node is requested to reach the threshold, it selects the neighbor node as the new storage node, and makes the node access to the neighbor node. The advantage of random placement is load balancing, reducing access latency, the shortcomings are that the number of replicas are too large. Yuan et al. proposed a strategy of data replication placement based on simulated annealing algorithm, which provides a local replication of the remote data that can be quickly accessed and processed by users, and finds the optimal replication placement node by the simulated annealing algorithm[6]. Pang et al. proposed a replication creation policy, which includes the intra-domain replication-derived policy and the inter-domain replication extension strategy[7]. The replicas spread across domains according to the frequency of replication visits. This improves the response speed of user access and reduces bandwidth consumption. Yang et al. proposed a replication optimal placement algorithm based on user's interest, which extracts the user group content interest and gives priority to the replication of the group's interest value[8].

However, the existing data placement mechanism is mainly for the simple homogeneous storage environment, but not for the increasingly complex and dynamic heterogeneous storage environment[9]. The data replication placement mechanism focuses on the replication consistency and reliability for open heterogeneous storage systems. Heterogeneous storage systems need to realize data availability and load balancing, and ensure the quality of service (QoS). In this paper, we propose a new replication placement mechanism with the adjustable replica deployment strategy (ARDS) for heterogeneous data storage systems, which considers the availability of data, replication access frequency and storage node residual capacity. With the premise of the data availability, the mechanism initializes the number of replicas. Then, the number of replicas will be adjusted based on the frequency of replication access. The replica placement mechanism is based on the remaining storage capacities of nodes, and prefers to place the replicas on nodes with higher current capacity, thus effectively avoiding the load unbalance of storage nodes.

## 2. Data replica placement mechanism

### 2.1. Determination of the initial number of data replica

In an open heterogeneous storage system, the online storage node is not all stable and high-performance. So, it is important to consider the availability of nodes to determine the number of initial replicas. The node availability is defined as event $N$, the probability of occurrence is defined as $\overline{N}$, then the unavailability of nodes is defined as $\overline{N}$, the probability of occurrence is $P(\overline{N}) = 1 - P(N)$.

We also need to consider the data availability, which needs to be measured both at the file level and the data block level. The data block availability of event occurrence probability is defined as $P(B)$, while the occurrence probability of data unavailability is $P(\overline{B}) = 1 - P(B)$. The availability of data file is defined as the probability of occurrence of the event, that is $P(F)$, and the probability that the data file unavailability occurs is $P(\overline{F}) = 1 - P(F)$.

It is assumed that the file consists of $j$ pieces of data blocks $\{block_1, block_2, block_3, \ldots, block_j\}$, and each data block has $r$ replicas, described as $\{replica_1, replica_2, replica_3, \ldots, replica_r\}$. When all replicas of $block_j$ are not available, $block_j$ is not available, that is

$$P(\overline{B}) = P(\overline{N_1} \times \overline{N_2} \times \ldots \times \overline{N_R} \times \ldots \times \overline{N_r})$$