

## 1 Background

The Hadoop Distributed File System (HDFS) consists of processes of different roles running on different nodes. The roles in HDFS involve NameNode, DataNode, JournalNode, Client, and so on. In this paper, we call them components in HDFS. To tolerate failures, a component usually has multiple replications. For example, the HDFS in our tests is composed of 2 NameNode processes, 4 DataNodes, 3 JournalNodes, and 1 Client. We call them the instances of a component.

Because of components are replicated, we can shutdown and reboot the instances of a component once a time and the system is supposed to tolerate these event and run normally. However, if we shutdown an instance and update its configuration file before reboot it, issues may happen in the system. We call the issues that only happen when component instances are reconfigured as reconfiguration errors in the system.

## 2 Definition

Before giving a formal definition, we need to define some other concepts first.

We define two types of reconfiguration methods, cluster-wide reconfiguration and component-wide reconfiguration. Cluster reconfiguration shutdowns and reboot the whole system to perform reconfiguration, while component reconfiguration will shutdown and reboot the instances of a component once a time to sustain system availability.

We define two types of errors in the system, internal errors and external errors. Internal errors are warnings and errors in HDFS logs, and external errors are errors reported by benchmarks running as the clients. We assume that, by repeating tests enough number of times, all the errors can be caught. As external errors indicate explicit and significant system issues, we will check external errors first and do not distinguish external errors.

Given a reconfiguration method  $m$ , a parameter  $p$ , if there exists a value pair  $v1$  and  $v2$  such that  $Test(v1, v2) \not\subseteq Test(v1, v1) \cup Test(v2, v2)$ , then we say that parameter  $p$  is not  $m$ -reconfigurable.