Impala QA Checker

Ken Farmer
William Farmer

August 13, 2015

Abstract

This project attempts to act as a health report for your hadoop cluster. At its core, this tool acts as a Hadoop Testing Framework. Tests are either in the form of rules or checks, and either flag the table for review if a rule fails, or throws a warning if a check fails. Tests can be scheduled, and the incremental output can be viewed through the local frontend. This allows for the user to view the health and status of their database over time.

1 Introduction

This tool attempts to perform health checks on your Hadoop cluster through automated scripts that are run. The results of these scripts can be viewed over time to determine the status of the cluster, and how its status compares to the last few days.

2 Tests

Every test is simply a script that lives in a type-specific folder. Rules and Checks exist separately.

Each script returns a JSON-encoded object to `stdout` that meets the following specification.

```
1 {
2     "name":"Name of test",
3     "violations":9000,
4     "Output":"Test Specific output. Either JSON or String"
5 }
```

2.1 Rules

Rules are strict rules about the cluster. They should never be ignored, or disobeyed, and any violation of a set rule results in a violation. Ideally, a "healthy" cluster should have no rule violations.

An example rule would be that a specific column only contains integers.

2.2 Checks

Checks are suggestions about specific tables, or databases. These are a lot more fluid, and a warning from a check does not necessarily indicate an issue with the database, but rather that something new may have happened.

For example, a check that examines the average number of a column may throw a warning if an entry is too far from the mean. This isn't an indication that the data is bad, just that it *may* be bad.