**App definition**

1. Pull jmx from all DN every 30 min
2. Aggregate last 48 files and make **daily average** file
3. Aggregate last 7 daily average file and make **weekly average** file
4. Compare **current file** with **daily average** and **weekly average** if any of metric is +/- 30% then send an alert or create **alert report file**
5. Create a log for activities happened for past 7 days.

*Exception handling*

1. What if dn\_hosts.csv is missing

*Performance & Negative impact*

1. Impact and timing performance impact

**Components used**

1. Use sqlite as storage layer
2. Python , shell for processing

**Implementation**

**Architecture**

--> [2 **config file** ]

[5 **curl DNs** ] --> [ 1 **App** (python shell script) ] --> [3 **sqlite** (data & state of App)]

--> [4 **log** (flat file) ]

**System Flow**

1 **App**

* *main.sh* : A driving script file , which could be triggered by cron for every 30 min
  + Creates and updates log file
  + Checks for pre-requisites to start application
* *collect\_stats.sh* 
  + [\*]Collect stats by connecting all DNs, get required metrics from config file to run curl script
    - Check that stats collected from all required DN
  + [\*]Store it in DB at sqlite
* *aggregate.sh* 
  + [\*]Aggregate stats of 48 runs and insert into daily average table - get columns from config file, to generate aggregate query
  + [\*]Aggregate last 7 daily average run data and make weekly average file - get columns from config file, to generate aggregate query
    - Use sqlite as staging , trigger, process data store data in file and drop database
  + [ ]Purge files older than a week
* *health\_check.sh* : Parse through aggregate data with current data - get metrics to be checked from config file.
  + [\*]Compare current file with daily average and weekly average if any of metric is +/- 30% create alert report file
  + [ ]trigger communication [ an email to admin ]
    - Cluster is healthy
    - So and so things has to be observed
  + [ ]Maintain history of alerts generated so far.

2 **Config**

* section DataNode
  + [\*]Metrics to be collected and monitored from jmx list
    - [\*]DDL could dynamically be created from this config file, DDLs should be versioned , every row should have config version.

* [\*]Threshold % to be monitored
* section NodeManager
  + [ ]Metrics to be collected and monitored from jmx list --> DDL could dynamically be created from this config file
  + [ ]Threshold % to be monitored
* Updates in config file should have version and timestamp updated
* []Let the metrics be have a logic how it has to be processed Eg
  + VolumeFailures > 0
  + Container launch time greater than 30% of weekly average

3 **sqlite**

* [\*]schema : Table design, DDL ( To dynamically generate or manually )

4 **log**

* [ ]Log normal operation , what are the steps , are they went well
* [ ]Log abnormal operation , if any generate monitor app health report to admin

**Components**

python2.7 , config - json , sqlite , cron