MX2020 Big Data

GCG Mexico – Architecture MetricsService

June 28, 2019

Versión	Fecha	Descripción del Cambio	Autor/Departamento
1.0	01/07/2019	Creación del documento	[Big Data Architecture]

MX2020 Big Data

Table of Contents

Definición del documento

Purpose

The following document has the purpose of presenting a solution alternative to cover the functionality of measuring and controlling:

- Access to tables by data domain by area
- History Ingest
- Users by domain
- Target Consulting
- Type of query in target

Scope

Define a common structure, projecting a solution as a service.

Service Metrics

Global Architecture - First proposal

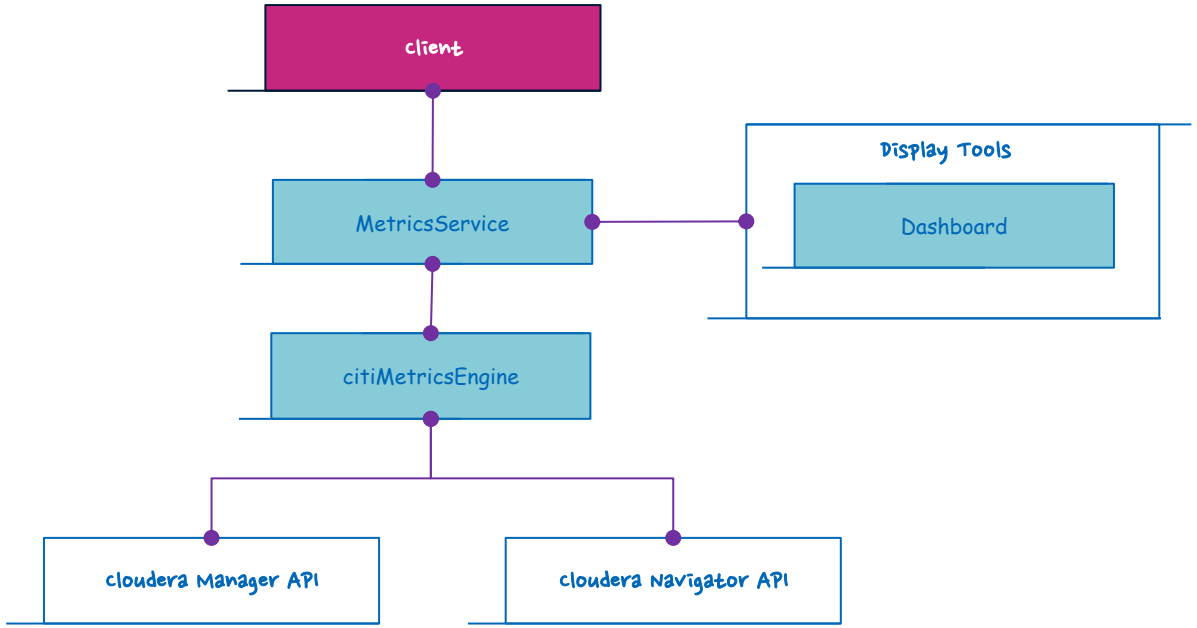
New component
Component to be modified
Component not contemplated

Principios de Arquitectura:

- Separación de Responsabilidades

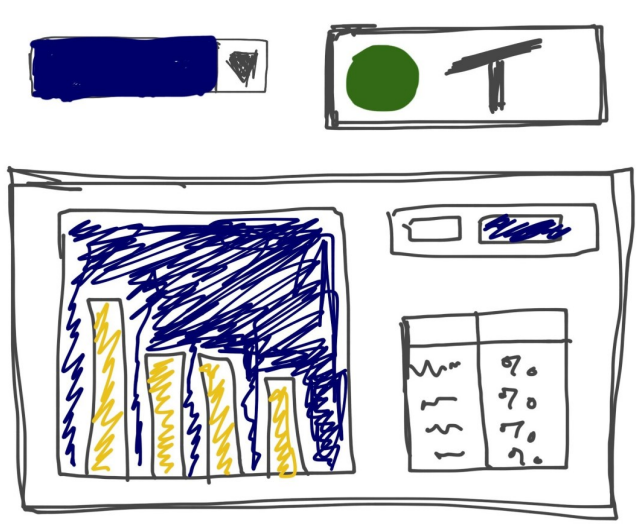
Requerimientos no funcionales:

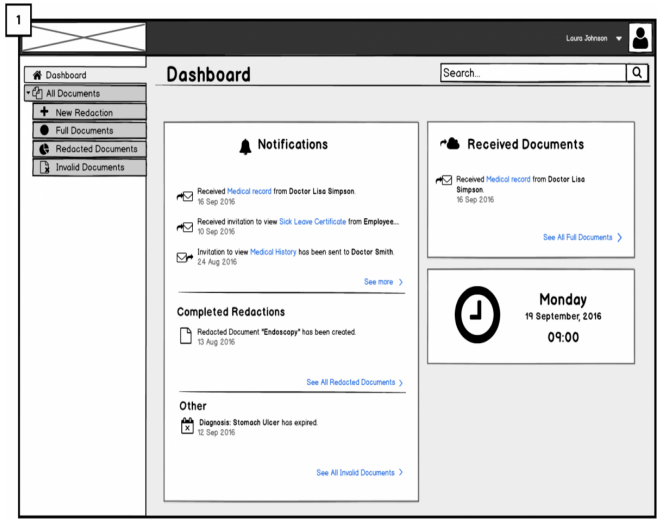
- Usabilidad
- Eficiencia
- Disponibilidad
- Escalabilidad
- Flexibilidad
- Reusabilidad
- Mantenibilidad



Service Metrics

Global Architecture – Customized metrics on a board





1. Image. Example dashboard

Service Metrics

Global Architecture - First proposal

The API also provides access to management functions:

- Obtaining logs and monitoring the system
- Starting and stopping services
- Polling cluster events
- Creating a disaster recovery replication schedule

The API also provides access to management functions:

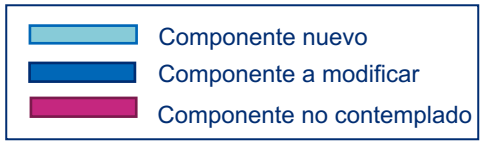
- OEM and hardware partners that deliver Hadoop-in-a-box appliances using the API to set up CDH and Cloudera Manager on bare metal in the factory.
- Automated deployment of new clusters, using a combination of Puppet and the Cloudera Manager API. Puppet does the OS-level provisioning and installs the software. The Cloudera Manager API sets up the Hadoop services and configures the cluster.
- Integrating the API with reporting and alerting infrastructure. An external script can poll the API for health and metrics information, as well as the stream of events and alerts, to feed into a custom dashboard.

REST Resources

This API supports a Representational State Transfer (REST) model for accessing a set of resources through a fixed set of operations. The following resources are accessible through the RESTful model:

Service Metrics

Global Architecture - First proposal



The API also provides access to management functions:

- Obtaining logs and monitoring the system
- Starting and stopping services
- Polling cluster events
- Creating a disaster recovery replication schedule

The API also provides access to management functions:

- OEM and hardware partners that deliver Hadoop-in-a-box appliances using the API to set up CDH and Cloudera Manager on bare metal in the factory.
- Automated deployment of new clusters, using a combination of Puppet and the Cloudera Manager API. Puppet does the OS-level provisioning and installs the software. The Cloudera Manager API sets up the Hadoop services and configures the cluster.
- Integrating the API with reporting and alerting infrastructure. An external script can poll the API for health and metrics information, as well as the stream of events and alerts, to feed into a custom dashboard.

REST Resources

This API supports a Representational State Transfer (REST) model for accessing a set of resources through a fixed set of operations. The following resources are accessible through the RESTful model:

Service Metrics

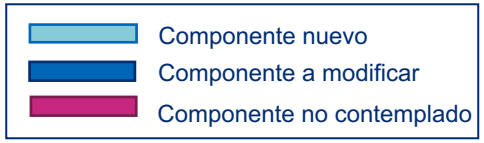
Global Architecture - First proposal

Cloudera Manager API

Functionalities		
Metric Name	Description	Unit
alerts_rate	The number of alerts.	events per second
events_critical_rate	The number of critical events.	events per second
events_important_rate	The number of important events.	events per second
events_informational_rate	The number of informational events.	events per second
health_bad_rate	Percentage of Time with Bad Health	seconds per second
health_concerning_rate	Percentage of Time with Concerning Health	seconds per second
health_disabled_rate	Percentage of Time with Disabled Health	seconds per second
health_good_rate	Percentage of Time with Good Health	seconds per second
health_unknown_rate	Percentage of Time with Unknown Health	seconds per second

Service Metrics

Global Architecture – First proposal - Client Java / Scala / Python



Maven

```
<project>
  <repositories>
    <repository>
      <id>cdh.repo</id>
      <url>https://repository.cloudera.comgroups/cloudera-repos</url>
      <name>Cloudera Repository</name>
    </repository>
    ...
  </repositories>
  <dependencies>
    <dependency>
      <groupId>com.cloudera.api</groupId>
      <artifactId>cloudera-manager-api</artifactId>
      <version>4.6.2</version>      <!-- Set to the version of Cloudera Manager you use -->
    </dependency>
    ...
  </dependencies>
```

Java client

```
public class ListClusters {

  public static void main(String[] args) throws IOException {
    ApiClient cmClient = Configuration.getDefaultApiClient();

    // Configure HTTP basic authorization: basic
    cmClient.setBasePath("https://cm-host:7183/api/v30");
    cmClient.setUsername("username");
    cmClient.setPassword("password");

    // Configure TLS for secure communication
    cmClient.setVerifyingSsl(true);

    Path truststorePath = Paths.get("/path/to/ca_cert_file.pem");
    byte[] truststoreBytes = Files.readAllBytes(truststorePath);
    cmClient.setSslCaCert(new ByteArrayInputStream(truststoreBytes));

    ClustersResourceApi apiInstance = new ClustersResourceApi(cmClient);
    try {
      ApiClusterList clusterList = apiInstance.readClusters("SUMMARY");
      for (ApiCluster cluster : clusterList.getItems()) {
        System.out.printf("Name: %os, Version: %os", cluster.getDisplayName(),
          cluster.getFullVersion());
      }
    } catch (ApiException e) {
      System.err.println("Exception when calling ClustersResourceApi#readClusters");
      e.printStackTrace();
    }
  }
}
```

2. Example client Java

Service Metrics

API Analysis – REST Resources

Resources		
name	path	methods
ActivitiesResource	/clusters/{clusterName}/services/{serviceName}/activities /clusters/{clusterName}/services/{serviceName}/activities/{activityId} /clusters/{clusterName}/services/{serviceName}/activities/{activityId}/children /clusters/{clusterName}/services/{serviceName}/activities/{activityId}/metrics /clusters/{clusterName}/services/{serviceName}/activities/{activityId}/similar	GET GET GET GET GET
AllHostsResource	/audits /audits/stream	GET GET
AuthRoleMetadatasResource	/authRoleMetadatas	GET
AuthRolesResource	/authRoles /authRoles/metadata /authRoles/{uuid}	GET, POST GET DELETE, GET, PUT
RoleConfigGroupsResource	/clusters/{clusterName}/services/{serviceName}/roleConfigGroups /clusters/{clusterName}/services/{serviceName}/roleConfigGroups/roles /clusters/{clusterName}/services/{serviceName}/roleConfigGroups/{roleConfigGroupName} /clusters/{clusterName}/services/{serviceName}/roleConfigGroups/{roleConfigGroupName}/config /clusters/{clusterName}/services/{serviceName}/roleConfigGroups/{roleConfigGroupName}/roles	GET, POST PUT DELETE, GET, PUT GET, PUT GET, PUT
RolesResource	/clusters/{clusterName}/services/{serviceName}/roles /clusters/{clusterName}/services/{serviceName}/roles/bulkDelete /clusters/{clusterName}/services/{serviceName}/roles/{roleName} /clusters/{clusterName}/services/{serviceName}/roles/{roleName}/commands /clusters/{clusterName}/services/{serviceName}/roles/{roleName}/commandsByName /clusters/{clusterName}/services/{serviceName}/roles/{roleName}/config /clusters/{clusterName}/services/{serviceName}/roles/{roleName}/metrics /clusters/{clusterName}/services/{serviceName}/roles/{roleName}/commands/enterMaintenanceMode /clusters/{clusterName}/services/{serviceName}/roles/{roleName}/commands/exitMaintenanceMode /clusters/{clusterName}/services/{serviceName}/roles/{roleName}/commands/impalaDiagnostics /clusters/{clusterName}/services/{serviceName}/roles/{roleName}/logs/full /clusters/{clusterName}/services/{serviceName}/roles/{roleName}/logs/stacks /clusters/{clusterName}/services/{serviceName}/roles/{roleName}/logs/stacksBundle /clusters/{clusterName}/services/{serviceName}/roles/{roleName}/logs/stderr /clusters/{clusterName}/services/{serviceName}/roles/{roleName}/logs/stdout	GET, POST POST DELETE, GET GET GET GET, PUT GET POST POST POST GET GET GET GET GET

Service Metrics

API Analysis – REST Resources

Resources		
name	path	methods
ServicesResource	/clusters/{clusterName}/services	GET, POST
	/clusters/{clusterName}/services/{serviceName}	DELETE, GET, PUT
	/clusters/{clusterName}/services/{serviceName}/clientConfig	GET
	/clusters/{clusterName}/services/{serviceName}/commands	GET
	/clusters/{clusterName}/services/{serviceName}/commandsByName	GET
	/clusters/{clusterName}/services/{serviceName}/config	GET, PUT
	/clusters/{clusterName}/services/{serviceName}/impalaUtilization	GET
	/clusters/{clusterName}/services/{serviceName}/metrics	GET
	/clusters/{clusterName}/services/{serviceName}/roleTypes	GET
	/clusters/{clusterName}/services/{serviceName}/yarnUtilization	GET
	/clusters/{clusterName}/services/{serviceName}/commands/createOozieDb	POST
	/clusters/{clusterName}/services/{serviceName}/commands/createSolrHdfsHomeDir	POST
	/clusters/{clusterName}/services/{serviceName}/commands/createSqoopUserDir	POST
	/clusters/{clusterName}/services/{serviceName}/commands/decommission	POST
	/clusters/{clusterName}/services/{serviceName}/commands/deployClientConfig	POST
	/clusters/{clusterName}/services/{serviceName}/commands/disableJtHa	POST
	/clusters/{clusterName}/services/{serviceName}/commands/disableRmHa	POST
	/clusters/{clusterName}/services/{serviceName}/commands/disableSentryHa	POST
	/clusters/{clusterName}/services/{serviceName}/commands/enableJtHa	POST
	/clusters/{clusterName}/services/{serviceName}/commands/enableRmHa	POST
	/clusters/{clusterName}/services/{serviceName}/commands/enableSentryHa	POST
	/clusters/{clusterName}/services/{serviceName}/commands/enterMaintenanceMode	POST
	/clusters/{clusterName}/services/{serviceName}/commands/exitMaintenanceMode	POST
	/clusters/{clusterName}/services/{serviceName}/commands/firstRun	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hbaseCreateRoot	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hbaseUpgrade	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hdfsCreateTmpDir	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hdfsDisableAutoFailover	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hdfsDisableHa	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hdfsDisableNnHa	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hdfsEnableAutoFailover	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hdfsEnableHa	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hdfsEnableNnHa	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hdfsFailover	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hdfsFinalizeRollingUpgrade	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hdfsRollEdits	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hdfsUpgradeMetadata	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hiveCreateHiveUserDir	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hiveCreateHiveWarehouse	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hiveCreateMetastoreDatabase	POST

Service Metrics

API Analysis – REST Resources

Resources		
name	path	methods
ServicesResource	/clusters/{clusterName}/services/{serviceName}/commands/hiveCreateMetastoreDatabaseTables	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hiveUpdateMetastoreNamenodes	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hiveUpgradeMetastore	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hiveValidateMetastoreSchema	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hueCreateHiveWarehouse	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hueDumpDb	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hueLoadDb	POST
	/clusters/{clusterName}/services/{serviceName}/commands/hueSyncDb	POST
	/clusters/{clusterName}/services/{serviceName}/commands/impalaCreateCatalogDatabase	POST
	/clusters/{clusterName}/services/{serviceName}/commands/impalaCreateCatalogDatabaseTables	POST
	/clusters/{clusterName}/services/{serviceName}/commands/impalaCreateUserDir	POST
	/clusters/{clusterName}/services/{serviceName}/commands/impalaDisableLlamaHa	POST
	/clusters/{clusterName}/services/{serviceName}/commands/impalaDisableLlamaRm	POST
	/clusters/{clusterName}/services/{serviceName}/commands/impalaEnableLlamaHa	POST
	/clusters/{clusterName}/services/{serviceName}/commands/impalaEnableLlamaRm	POST
	/clusters/{clusterName}/services/{serviceName}/commands/importMrConfigsIntoYarn	POST
	/clusters/{clusterName}/services/{serviceName}/commands/initSolr	POST
	/clusters/{clusterName}/services/{serviceName}/commands/installMrFrameworkJars	POST
	/clusters/{clusterName}/services/{serviceName}/commands/installOozieShareLib	POST
	/clusters/{clusterName}/services/{serviceName}/commands/migrateToSentry	POST
	/clusters/{clusterName}/services/{serviceName}/commands/offline	POST
	/clusters/{clusterName}/services/{serviceName}/commands/oozieCreateEmbeddedDatabase	POST
	/clusters/{clusterName}/services/{serviceName}/commands/oozieDisableHa	POST
	/clusters/{clusterName}/services/{serviceName}/commands/oozieDumpDatabase	POST
	/clusters/{clusterName}/services/{serviceName}/commands/oozieEnableHa	POST
	/clusters/{clusterName}/services/{serviceName}/commands/oozieLoadDatabase	POST
	/clusters/{clusterName}/services/{serviceName}/commands/oozieUpgradeDb	POST
	/clusters/{clusterName}/services/{serviceName}/commands/recommission	POST
	/clusters/{clusterName}/services/{serviceName}/commands/recommissionWithStart	POST
	/clusters/{clusterName}/services/{serviceName}/commands/restart	POST
	/clusters/{clusterName}/services/{serviceName}/commands/rollingRestart	POST
	/clusters/{clusterName}/services/{serviceName}/commands/sentryCreateDatabase	POST
	/clusters/{clusterName}/services/{serviceName}/commands/sentryCreateDatabaseTables	POST
	/clusters/{clusterName}/services/{serviceName}/commands/sentryUpgradeDatabaseTables	POST
	/clusters/{clusterName}/services/{serviceName}/commands/solrBootstrapCollections	POST
	/clusters/{clusterName}/services/{serviceName}/commands/solrBootstrapConfig	POST
	/clusters/{clusterName}/services/{serviceName}/commands/solrConfigBackup	POST
	/clusters/{clusterName}/services/{serviceName}/commands/solrMigrateSentryPrivilegesCommand	POST
	/clusters/{clusterName}/services/{serviceName}/commands/solrReinitializeStateForUpgrade	POST

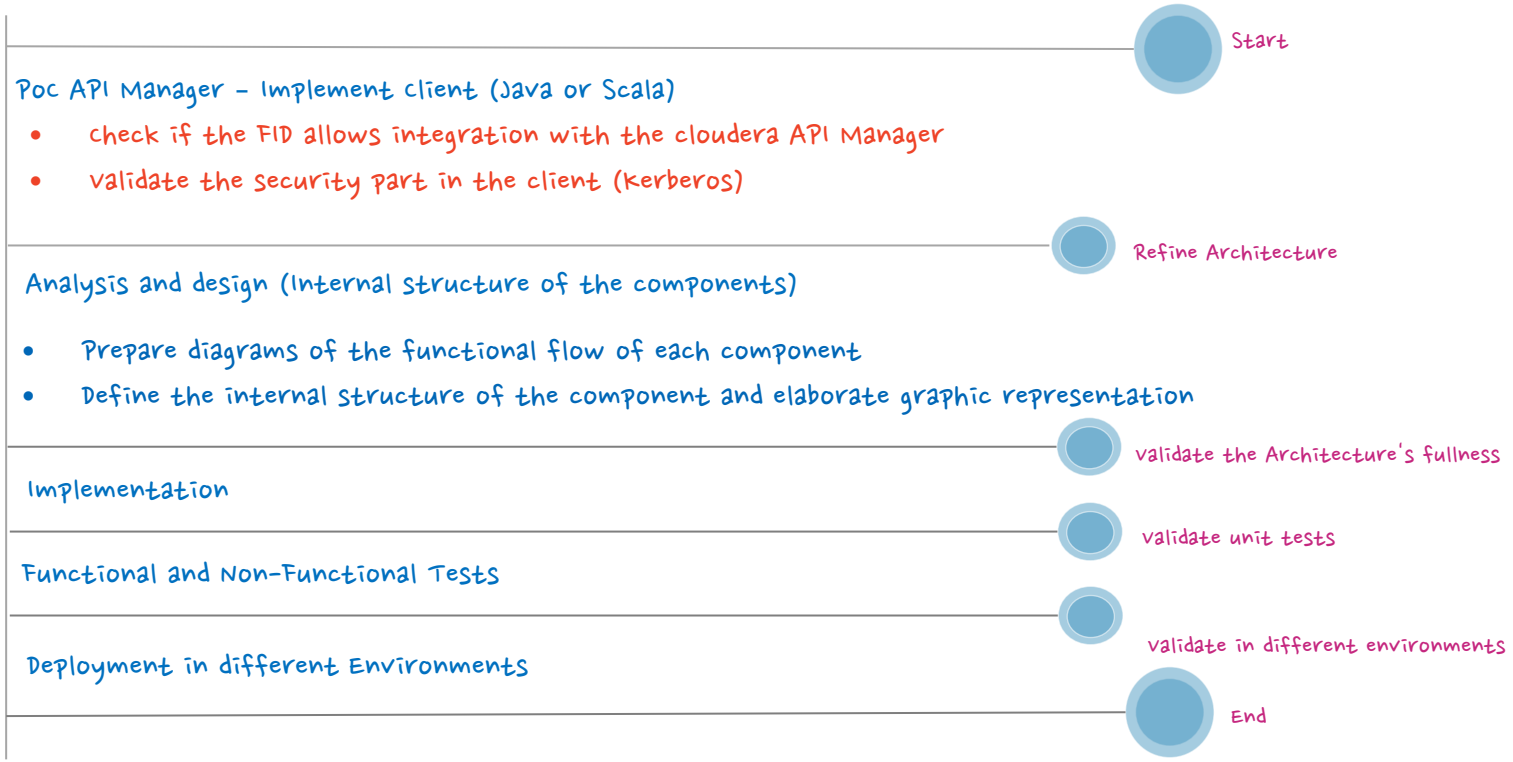
Service Metrics

API Analysis – REST Resources

Resources		
name	path	methods
ServicesResource	/clusters/{clusterName}/services/{serviceName}/commands/solrValidateMetadata	POST
	/clusters/{clusterName}/services/{serviceName}/commands/sqoopCreateDatabaseTables	POST
	/clusters/{clusterName}/services/{serviceName}/commands/sqoopUpgradeDb	POST
	/clusters/{clusterName}/services/{serviceName}/commands/start	POST
	/clusters/{clusterName}/services/{serviceName}/commands/stop	POST
	/clusters/{clusterName}/services/{serviceName}/commands/switchToMr2	POST
	/clusters/{clusterName}/services/{serviceName}/commands/yarnApplicationDiagnosticsCollection	POST
	/clusters/{clusterName}/services/{serviceName}/commands/yarnCreateCmContainerUsageInputDirCommand	POST
	/clusters/{clusterName}/services/{serviceName}/commands/yarnCreateJobHistoryDirCommand	POST
	/clusters/{clusterName}/services/{serviceName}/commands/yarnFormatStateStore	POST
	/clusters/{clusterName}/services/{serviceName}/commands/yarnNodeManagerRemoteAppLogDirCommand	POST
	/clusters/{clusterName}/services/{serviceName}/commands/zooKeeperCleanup	POST
	/clusters/{clusterName}/services/{serviceName}/commands/zooKeeperInit	POST
	/clusters/{clusterName}/services/{serviceName}/commands/{commandName}	POST
	/clusters/{clusterName}/services/{serviceName}/reports/hdfsUsageReport	GET
	/clusters/{clusterName}/services/{serviceName}/reports/mrUsageReport	GET

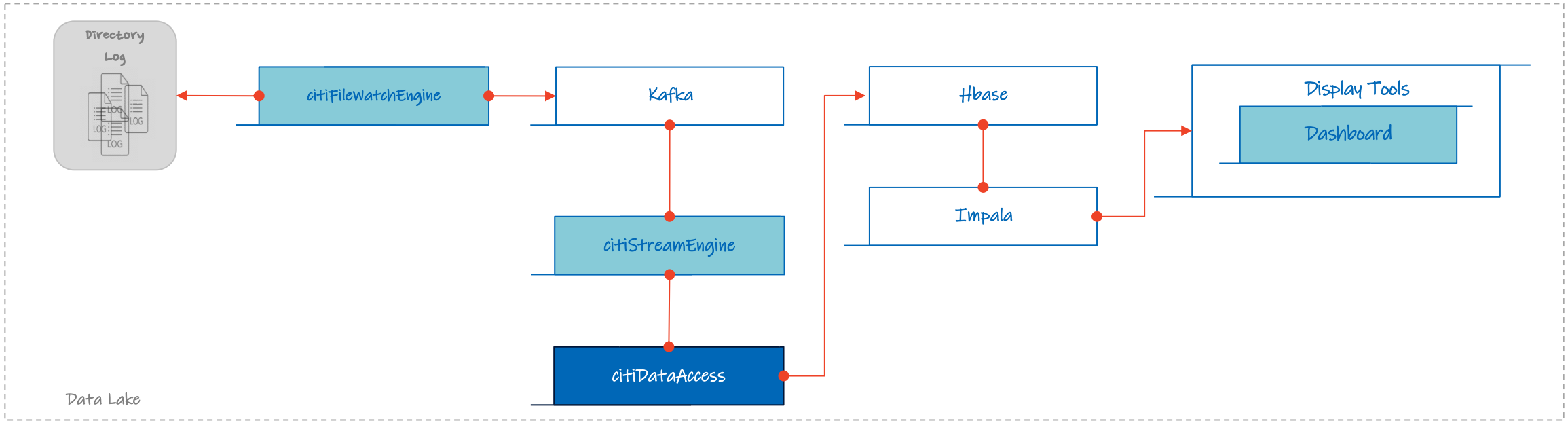
Service Metrics

First proposal - Locks & Activities to do



Service Metrics

Global Architecture - Second proposal



Principles of Architecture:

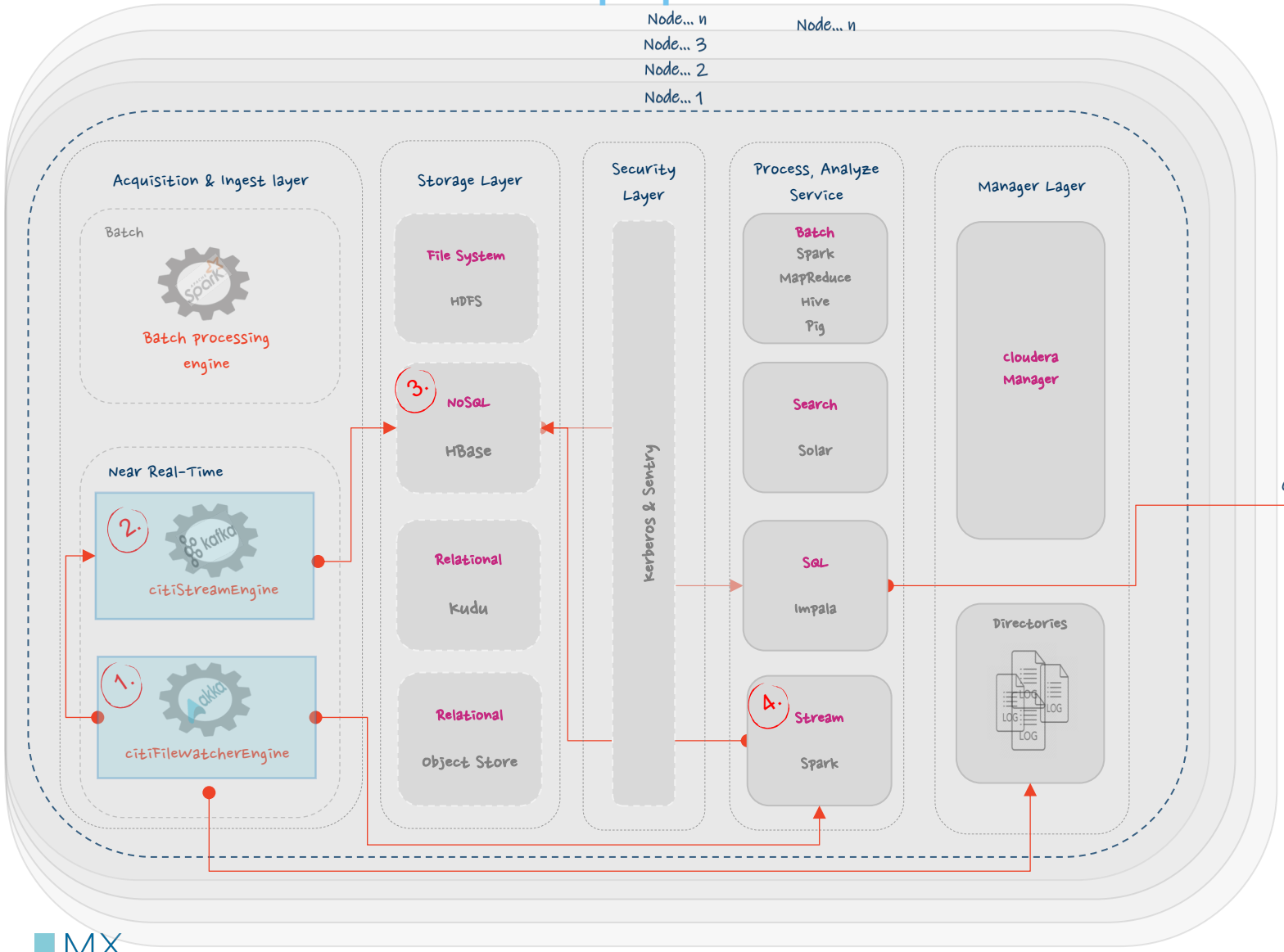
- Separation of Responsibilities

Non-functional requirements:

- Usability
- Efficiency
- Availability
- Scalability
- Flexibility
- Reusability
- Maintainability

Service Metrics

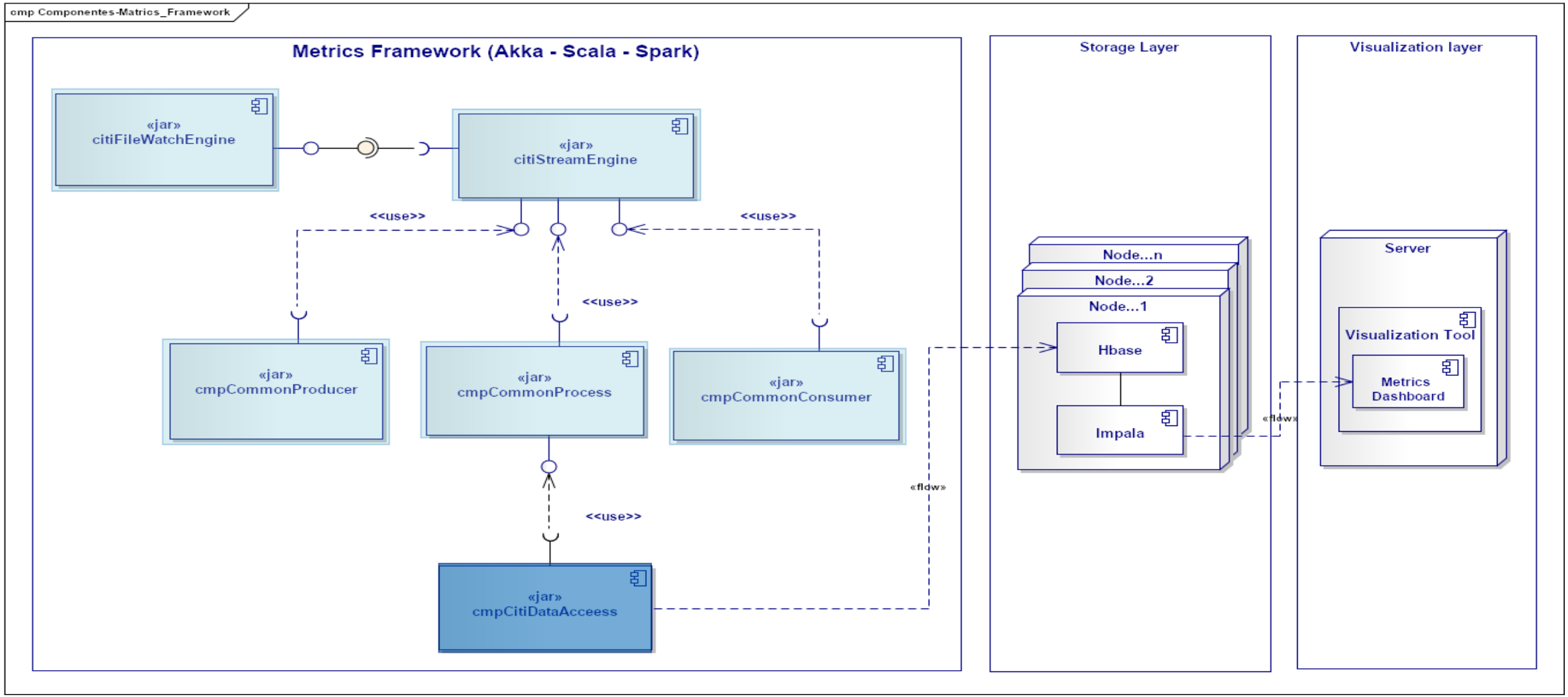
Global Architecture - Second proposal



1. The citiFileWatcherEngine detects events in the log
2. The citiFileWatcherEngine calls the citiStreamEngine to produce an event
3. The citiStreamEngine processes the message
4. Through the citiDataAccess the citiStreamEngine sends a processed event to Hbase
5. The visualization tool is integrated with the Hbase table from Impala

Service Metrics

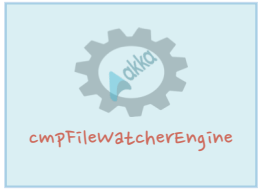
Global Architecture - Second proposal



Service Metrics

Global Architecture – Definition of core components

Componente nuevo
Componente a modificar
Componente no contemplado



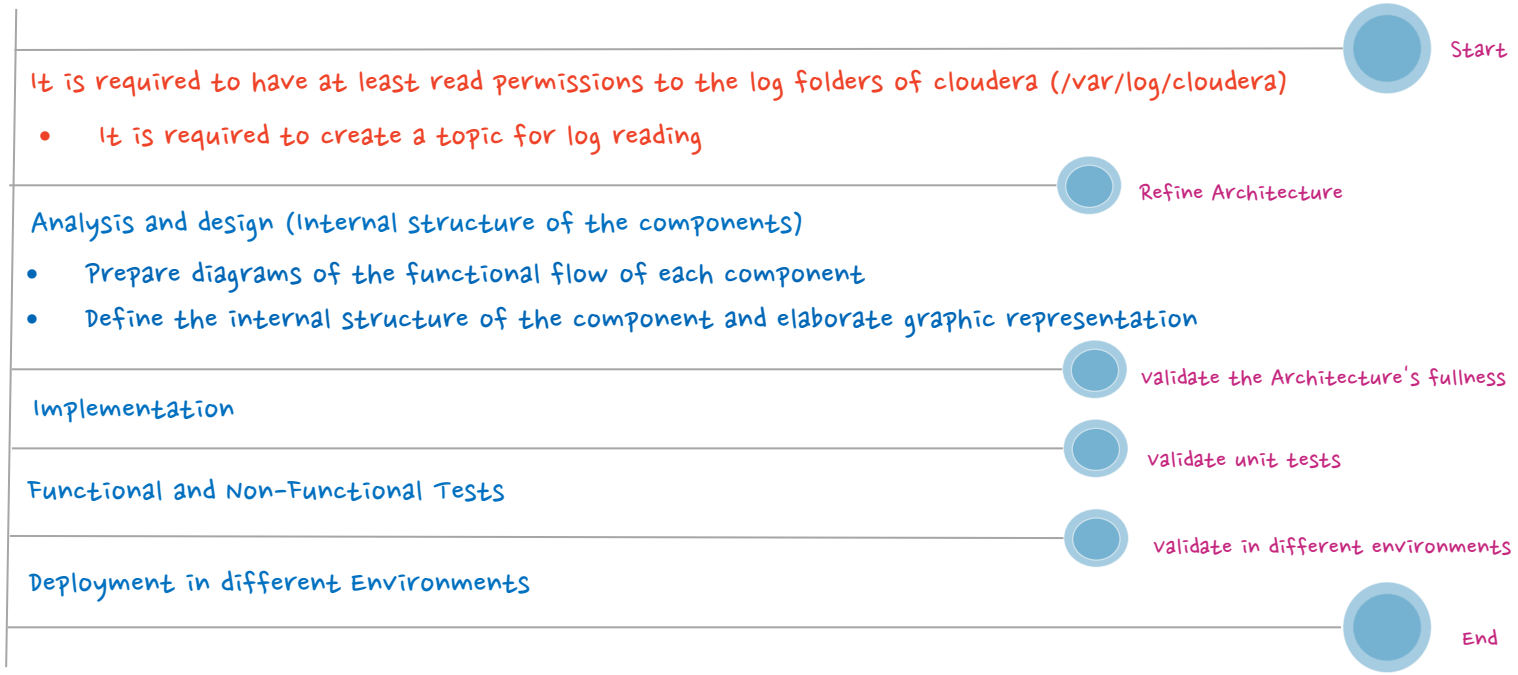
citiFileWatchEngine: It is the component responsible for polling the status of the files under a root directory, verifying the file system and notifying the citiStreamEngenie about events created, modified or deleted.

citiStreamEngine: It is the component that is responsible for producing, consuming and processing events with the information of the logs, it is integrated with the citiDataAccess

citiDataAccess: It is the component that separates integrations at the resource level (Teradata, Oracle, MongoDB, Hbase, Hive, etc.)

Service Metrics

Second proposal – Locks & Activities to do



References and Annex (I)

References

Nombre y version	Fecha	Comentarios	Rol/Departamento
https://github.com/cloudera/navigator-sdk/blob/master/examples/src/main/java/com/cloudera/nav/sdk/examples/extraction/IncrementalExtraction.java	No plica	Descripción y ejemplos para el uso de cloudera API Navegaitor	cloudera
https://www.cloudera.com/documentation/enterprise/5-14-x/topics/cm_intro_api.html	No aplica	Documentación del cloudera API Manager	cloudera