****

dAnalytics

Installation Document

Release 0.1

DRAFT

**Version**: 1.0

**Last Modified:** June 22, 2015

Prepared by:

TurningPoint Global Solutions

1355 Piccard Drive, Suite 250

Rockville, MD. 20850

**APPROVALS**

The undersigned acknowledge that they have reviewed the Design Document and agree with the information presented within this document. Changes to this Design Document will be coordinated with, and approved by the undersigned, or their designated representatives.

| Signature: |  | Date: |
| --- | --- | --- |
| Print Name: | David Hong |  |
| Title: | Project Lead |  |
|  |  |  |

**REVISION HISTORY**

| **Version** | **Date** | **Organization/Point of Contact** | **Description of Changes** |
| --- | --- | --- | --- |
| 0.1 | 06/21/2015 | TurningPoint DevelopmentTeam | Initial Draft Version |
|  |  |  |  |

Table of Contents

[APPROVALS i](#_Toc422868082)

[REVISION HISTORY ii](#_Toc422868083)

[1. INTRODUCTION 4](#_Toc422868084)

[1.1. Purpose 4](#_Toc422868085)

[1.2. Intended Audience 4](#_Toc422868086)

[2. Deployment Architecture 5](#_Toc422868087)

[3. Pre Requesties 5](#_Toc422868088)

[3.1. Java (v 1.8) 5](#_Toc422868089)

[3.2. Apache Hadoop 5](#_Toc422868090)

[i. PIG 6](#_Toc422868091)

[ii. Apache Sqoop 6](#_Toc422868092)

[Install/Setup Hadoop Multi-Node 6](#_Toc422868093)

[3.3. Data Storage 6](#_Toc422868094)

[i. MariaDB 6](#_Toc422868095)

[3.4. Presentation Component 6](#_Toc422868096)

[i. Apache HTTP Server 6](#_Toc422868097)

[3.5. Build Tools 6](#_Toc422868098)

[4. Application Installation Steps 6](#_Toc422868099)

[4.1. Database 7](#_Toc422868100)

[4.2. Web Components 7](#_Toc422868101)

[i. Rest API 7](#_Toc422868102)

[ii. Web Applications 7](#_Toc422868103)

[5. Hadoop, PIG and SCOOP Installation 8](#_Toc422868104)

[i. Hadoop Installation: - 8](#_Toc422868105)

[ii. PIG Installation:- 10](#_Toc422868106)

[iii. PIG and HADOOP Integration:- 11](#_Toc422868107)

INTRODUCTION

The OpenFDA is an innovation project that aims at creating easy access to the public data, to create a new level of openness and accountability, to ensure the privacy and security of public FDA data, and ultimately to educate the public and save lives.

The FDA launched its very first openFDA challenge to the developer community to take advantage of the following datasets and explore the range and extent of its impact for 1) research and 2) consumers.

* **Adverse events data.** FDA’s publicly available drug adverse event and medication error reports, and medical device adverse event reports.
* **Recalls data.** Enforcement report data, containing information gathered from public notices about certain recalls of FDA-regulated products.
* **Labeling data.** Structured Product Labeling (SPL) data for FDA-regulated human prescription drug, OTC drug and biological product labeling.

TurnintPoint has taken up the challenge of data analytics and thus the **dAnalytics** was born.

Purpose

The Intallation Document (ID) describes the steps to install and configure dAnalytis for each of the system’s components.

Intended Audience

The intended audience for this document includes Developers, Technical Architects and openFDA Technical Review Board. The document will eventually be made available to the open source community who can make use of the publically available code base to extened it.

1. Deployment Architecture

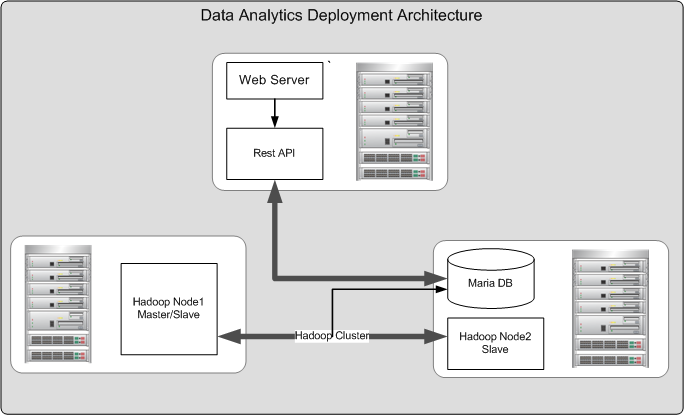


Figure 1.0

1. Pre Requesties

## Java (v 1.8)

Java is an open source, flexible, scalable, platform independent coding language that is widely used by open source developers and users.

If you do not have java installed use following article

<https://docs.oracle.com/javase/8/docs/technotes/guides/install/install_overview.html#CJAGAACB>

## Apache Hadoop

The Apache Hadoop software library is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage. Rather than rely on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, so delivering a highly-available service on top of a cluster of computers, each of which may be prone to failures

### PIG

### Apache Sqoop

Apache Sqoop (TM) is a tool designed for efficiently transferring bulk data between Apache Hadoop and structured data stores such as relational databases.

### [Install/Setup Hadoop Multi-Node](#_Hadoop_Installation:_-)

## Data Storage

### MariaDB

MariaDB is a logical choice for database professionals looking for a robust, scalable, and reliable SQL server. The MariaDB development team works closely and cooperatively with the larger community of users and developers in the true spirit of Free and open source software.

If you do not have MariaDB installed use following article

<https://mariadb.com>

## Presentation Component

### Apache HTTP Server

Apache HTTP server is an open source web container developed by the Apache Software Foundation (ASF).

## Build Tools

1. Application Installation Steps

Installation Order

* Database
* Web Components

## Database

* 1. execute drugSummary.sql
  2. execute populateDrugPeaks.sql
  3. Run PIG scripts
  4. Run Scoop

## Web Components

### Rest API

1. Build the distribution locally using the following gradle command from the cloned repository directory (<https://github.com/TPRockville/jDerive/tree/master/Engg/Src/REST> )
   1. Execute gradlew command with clean build option

Window : gradlew clean build

Unix/Linux: ./gradlew.sh clean build

1. If the previous step succeeds, it places the distribution  under
   1. <repository-folder>\openfda-service\build\distributions\openfda-service.tar and openfda-service.zip
2. Transfer the appropriate distribution to the target server
3. Unzip the contents in openfda-service.tar or zip on the target server
4. Configuring the Rest API
   1. Navigate to conf folder (under openfda-service) to update application.properties
      1. Register the http port under server.port

Example: server.port=8080

* + 1. Register Server IP/Domain name server.address

Example: server.address= localhost

* + 1. Register the JDBC parameters spring.datasource.url=jdbc:mariadb://<db\_server\_name>:<port>/<schema\_name>?user=<username>&password=<password>&useUnicode=true&characterEncoding=UTF8
  1. Navigate to **bin** folder under **openFDA-service** directory, and execute

$nohup openfda-service &

1. Try hitting the healthcheck endpoint as below
   1. http://<server.address>:<server.port>/jderive/health and confirm that the status is UP

### Web Applications

# Prerequisites

1. Install [Node.js](http://nodejs.org)

- on OSX use [homebrew](http://brew.sh) `brew install node`

- on Windows use [chocolatey](https://chocolatey.org/) `choco install nodejs` or download installation file from [Node.js](<http://nodejs.org>)

2. Install Grunt and Bower

Once Node is installed, run the below commands:

* + - npm install -g grunt-cli bower
    - bower install

Grunt Tasks

-------------

Tasks available for development and packaging also testing:

grunt serve #Runs a development server on node and runs livereload.

grunt test #Runn unit tests.

grunt build #Packages app (minified, concatenated, and more) in /dist

1. Hadoop, PIG and SCOOP Installation

### Hadoop Installation: -

* Download Hadoop 1.2.1 tar file
  + wget <http://www.webhostingreviewjam.com/mirror/apache/hadoop/common/hadoop-1.2.1/hadoop-1.2.1-bin.tar.gz>
* Extract to /opt/hadoop location
  + tar -xzf hadoop-1.2.1.tar.gz –C /opt/hadoop
* Set a password-less login to all the machines
  + Generate an RSA key for that user
    - ssh-keygen -t rsa -P ""
  + Copy the generated keys to authorized keys
    - cat $HOME/.ssh/id\_rsa.pub >> $HOME/.ssh/authorized\_keys
  + Give Execute permission to the authorized\_keys
    - chmod 755 $HOME/.ssh/authorized\_keys
  + Check whether it is able to login without password
    - ssh localhost
  + Copy the generated keys to other machines where password-less login is required
    - ssh-copy-id -i $HOME/.ssh/id\_rsa.pub username@ip
* Configure the Hadoop environment
  + vi /opt/hadoop/conf/hadoop-env.sh
    - Add export JAVA\_HOME and set it to JDK home.
    - Add export HADOOP\_OPTS=-Djava.net.preferIPv4Stack=true.
* Configure the core-site.xml file in the conf folder.
  + vi /opt/hadoop/conf/core-site.xml
  + Add the below configuration

<configuration>

<property>

<name>fs.default.name</name>

<value>hdfs://<hostname>:9000</value>

</property>

<property>

<name>dfs.permissions</name>

<value>false</value>

</property>

</configuration>

* Configure the hdfs-site.xml file in the conf folder.
  + vi /opt/hadoop/conf/hdfs-site.xml
  + Add the below configuration

<configuration>

<property>

<name>dfs.data.dir</name>

<value>/opt/hadoop-data/dfs/data</value>

</property>

<property>

<name>dfs.name.dir</name>

<value>/opt/hadoop-data/dfs/name</value>

</property>

<property>

<name>dfs.replication</name>

<value>3</value>

</property>

<property>

<name>dfs.permissions</name>

<value>false</value>

</property>

</configuration>

* Configure the mapred-site.xml file in the conf folder.
  + vi /opt/hadoop/conf/mapred-site.xml
  + Add the below configuration

<configuration>

<property>

<name>mapred.job.tracker</name>

<value><Master HostName>:9001</value>

</property>

<property>

<name>mapred.tasktracker.map.tasks.maximum</name>

<value>4</value>

</property>

</configuration>

* Configure on which machine your namenode/jobtracker has to run on.
  + vi /opt/hadoop/hadoop/conf/masters
    - <hostname>
* Configure on which machine your datanode/tasktracker has to run on.
  + vi /opt/hadoop/hadoop/conf/slaves
    - <hostname>
* Now export HADOOP\_HOME=<install root of hadoop> and PATH=$PATH:$HADOOP\_HOME/bin

And export HADOOP\_COMMON\_HOME=<Install directory>

* Change the ownership of the Hadoop folder, if the username is not root.
  + chown –R <username> /opt/hadoop
* Copy the /opt/hadoop folder to the machines where you want to configure data nodes.
  + scp –r /opt/hadoop username@ip:/opt/Hadoop
* Format Namenode
  + /opt/hadoop/bin/hadoop namenode –format
* Start and Stop Hadoop
  + start-all.sh
  + stop-all.sh

### PIG Installation:-

* Download pig-0.11.1.tar file

wget http://archive.apache.org/dist/pig/pig-0.11.1/pig-0.11.1.tar.gz

* Extract to /opt/pig location
  + tar -xzf pig-0.11.1.tar.gz –C /opt/pig
* Now export PIG\_HOME=<install root of pig> and PATH=$PATH:$PIG\_HOME/bin
* To run Pig
  + Local mode
    - pig –x local
  + MapReduce mode
    - pig or pig –x mapred

Sqoop Installation:-

* Download sqoop-1.4.6 file
  + wget http://mirror.nexcess.net/apache/sqoop/1.4.6/sqoop-1.4.6.bin\_\_hadoop-1.0.0.tar.gz
* Extract to /opt/sqoop location
  + tar -xzf sqoop-1.4.6.bin\_\_hadoop-1.0.0.tar.gz –C /opt/
* Change the user permissions and owner if it’s not the same
  + sudo chmod -R 755 sqoop-1.4.6.bin\_\_hadoop-1.0.0/
  + sudo chown -R <username> sqoop-1.4.6.bin\_\_hadoop-1.0.0/
* Now export SQOOP\_HOME=<install root of sqoop> and PATH=$PATH:$SQOOP\_HOME/bin

### PIG and HADOOP Integration:-

* Add the below in ~/.bash\_rc
  + export HADOOP\_PREFIX=<Root directory of Hadoop>
  + export HADOOP\_CONF\_DIR=<Conf directory of Hadoop>