# **Assumptions**

\*The code developed in a secure network to test it in PROD like environment

\*Spark-scala job is developed to run in cluster mode as it doesn't need a port to be exposed for the job to run the job

\*File ftp://ita.ee.lbl.gov/traces/NASA\_access\_log\_Jul95.gz is downloaded using curl command in start script

\*In case the file cant be downloaded the same is available in /src/main/config folder

\*the input file is placed in hadoop directory and output files are created in output directory so the output can be analyzed in Dermio

\*The validation of both input and output files are done through Dermio for ease of validation in seconds

# **Introduction**

Nasa Web Access Log Analyzer Application

## **Objective**

- Fetch top N visitor

- Fetch top N urls

### **Code walkthrough**

Input Download URL - ftp://ita.ee.lbl.gov/traces/NASA\_access\_log\_Jul95.gz

1. src/main/scala

\* \*com.logAnalyzer.scala.AnalyzerSpark\* - This is main entry class. SparkSession object is build here.

\* \*com.logAnalyzer.scala.AnalyzerTransformer\* - The spark Dataset processing logic is present here.

2. src/main/config

\* \*analyzer.conf\* - all configurations are maintained in a Typesafe config file.

3. src/test/scala

\* \*gov.nasa.test.loganalyzer.NasaWebAccessStatsProcessorTest.scala\* - This is the Unit test class for NasaWebAccessStatsProcessor.

4. \*ScalaSyleGuide.xml\* - This is the formatter configuration file used to format the code.

#**## Configurations**

**Configuration file name- \*analyzer-container.conf\***

Properties set in configuration file-

- \*\*valueOfN\*\*- This property is to set the value of N in the topNVisitors and topNUrls. \*REQUIRED INT value\*

- \*\*writeCurruptLogEnteries\*\*- Some of the log lines are corrupt. If writeCurruptLogEnteries is set to TRUE, then corrupt lines will be writen to filesystem, else not. \*BOOLEAN value\*

- \*\*filterResponseCodes\*\*- When evaluating topNUrls, this property is used to filter the log lines with undesired response code. Eg- when set to value ["304","400"], all log lines with response code 304 and 400 will be filtered. \*LIST value\*

- \*\*downloadFileLoc\*\*- This is the ftp location from where the input gz file will be downloaded. If set to blank, then file-download will not happen, assuming that the file is already on local filesystem. \*STRING value\*

- \*\*downloadFile\*\* -this is set to true or false if we want to totaly get away with the download option within scala code

- \*\*gzFSLocation\*\*- This is the local filesystem path where gz file is downloaded. If the property "downloadFileLoc" is not set, then the application assumes that the gz file is present at this location. \*REQUIRED STRING value\*

Incase required configuration properties are not set, then the application exits with code 0.

**### Assumptions**

1. **Data is structured in the following format-**

\* `<visitor> - - [<date> <timezone>] "<method> <url> <protocol>" <resonseCode> <unknownvariable>`

`E.g.- unicomp6.unicomp.net - - [01/Jul/1995:00:00:14 -0400] "GET /shuttle/countdown/count.gif HTTP/1.0" 200 40310`

String split functions have been used to derive date and other attributes based on the assumption that the data log line will follow this format.

2. When log-lines are tokenized using tokenizer \*\*SPACE\*\* (" "), there are entries with-

\* \*10 tokens\* - log-lines with 10 tokens have protocol value set as HTTP/1.0

\* \*9 tokens\* - log-lines with 9 tokens don't have any protocol value set.

\* \*8 or less tokens\* - these are considered as corrupt lines as it has incomplete information, such as missing method(GET,POST,HEAD etc) and/or missing other information.

**3. To evaluate topNurls, the application gives two options-**

\* filterResponseCodes set with a value - when set with a list of responseCodes, the log-lines with the specified response codes will be filtered out. Eg. a 404 response request can be filtered out when processing top N urls.

\* filterResponseCodes set BLANK - when set to blank, all log-lines will be considered valid to evaluate the top N urls. Only lines missed will be corrupt lines identified as tokens < 9.

4. To evaluate topNvisitors, no special logic has been added to filter log-lines based on HTTP method. The current implementation considers GET, POST, HEAD etc methods as the visitors.

**### Software versions**

- java version 1.11

- Scala version- 2.11.12

- Spark version- 2.4.5

- Intellij 2020.3

- maven

**### Steps to compile**

1. click on maven on the right side of the window screen

2. Run cmd- `maven clean`.

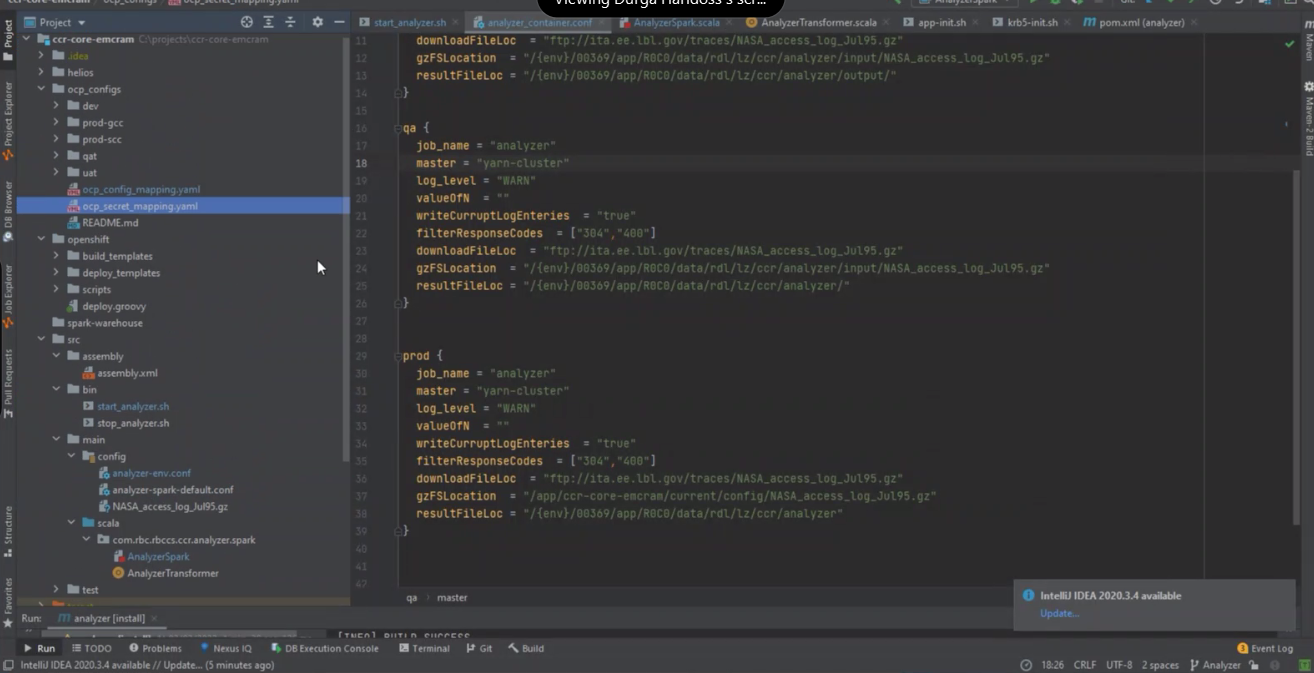
3. Run cmd 'maven install' this will create the tar.gz file based on the details in the assembly

4. The jar is generated in the target directory. Check jar full path in the console-log.

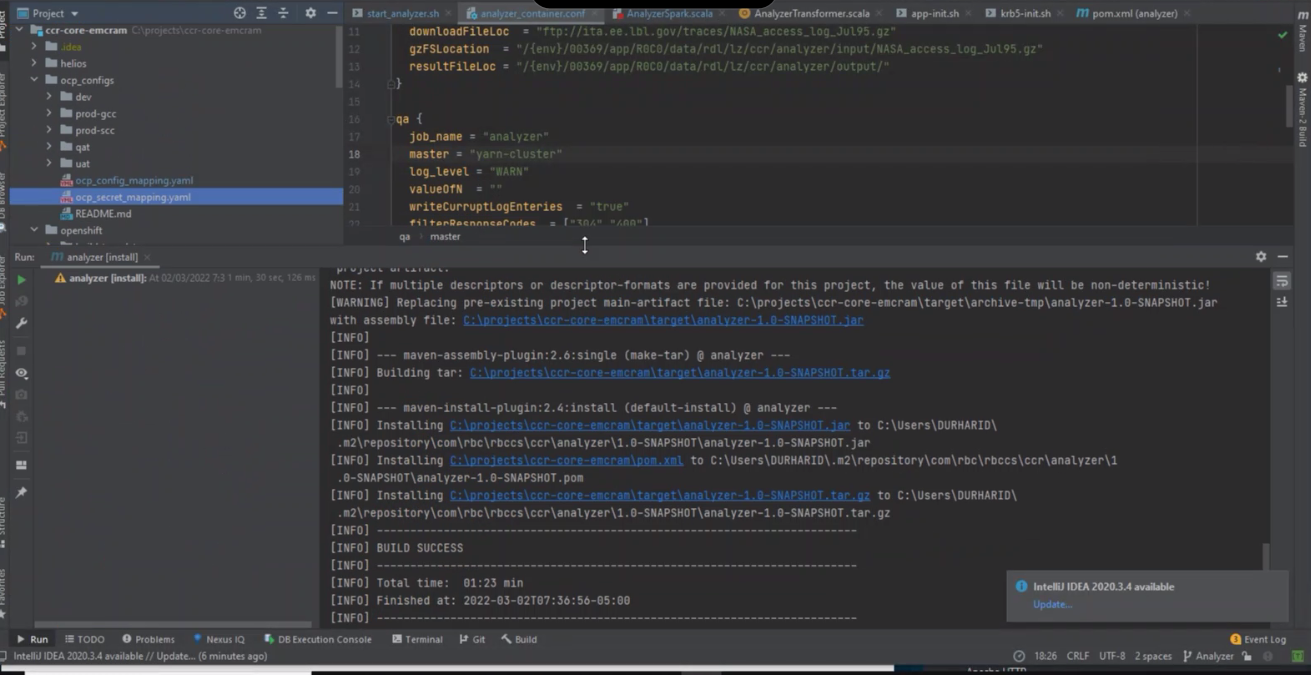
#### Command to execute

\*\*Note\*\*- If configuration property `downloadFileLoc` is set as "", in the config file, then the download of file will not happen. Instead the value of configuration property `gzFSLocation` will be used as the input file location for spark.

**Project directory of Analyzer project:**



**On compilation**:



**Windows Command to build and deploy in openshift:**

1. Oc login <hostname>

Username:

Password

1. Oc project <projectname>

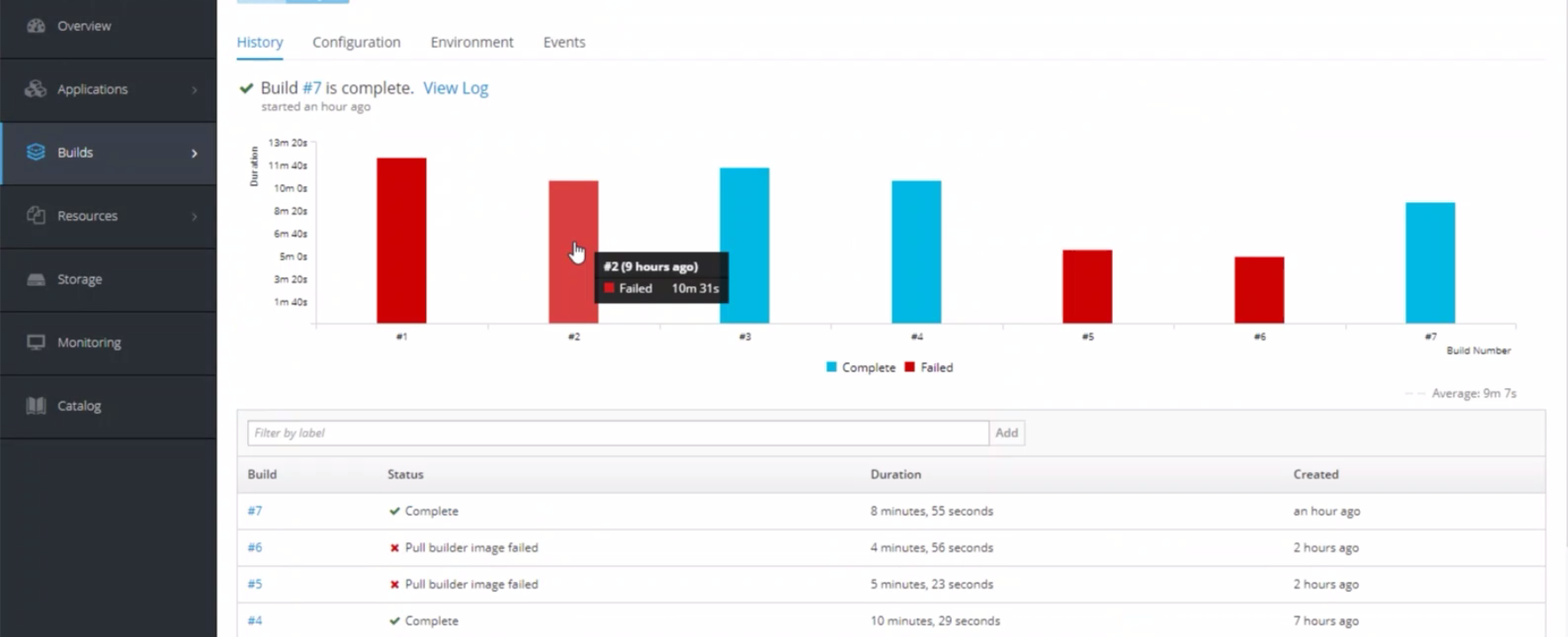
---

3.SET TAG=latest  
SET DEPLOY\_ENV=dev  
:: root directory of the project codebase downloaded  
SET workspace=C:\projects\analyzer  
:: Make sure it is the APP\_NAME defined in env\_config.txt  
SET APP\_NAME=analyzer  
  
cd %workspace%  
  
oc process -f openshift/build\_templates/build\_template.yaml --ignore-unknown-parameters=true -p TAG=%TAG% -p APP\_NAME=%APP\_NAME% --param-file ocp\_configs/%DEPLOY\_ENV%/ocp\_params.txt -o yaml | oc apply -f -  
  
:: Trigger the build from files existing in local working directory.  
oc start-build %APP\_NAME% --from-dir %workspace% --follow  
  
  
  
:: Delete and Create config maps  
FOR /D %G IN (ocp\_configs/%DEPLOY\_ENV%/\*) DO (  
oc delete cm/%APP\_NAME%-%G-%DEPLOY\_ENV%  
oc create configmap %APP\_NAME%-%G-%DEPLOY\_ENV% --from-file=ocp\_configs/%DEPLOY\_ENV%/%G/ )

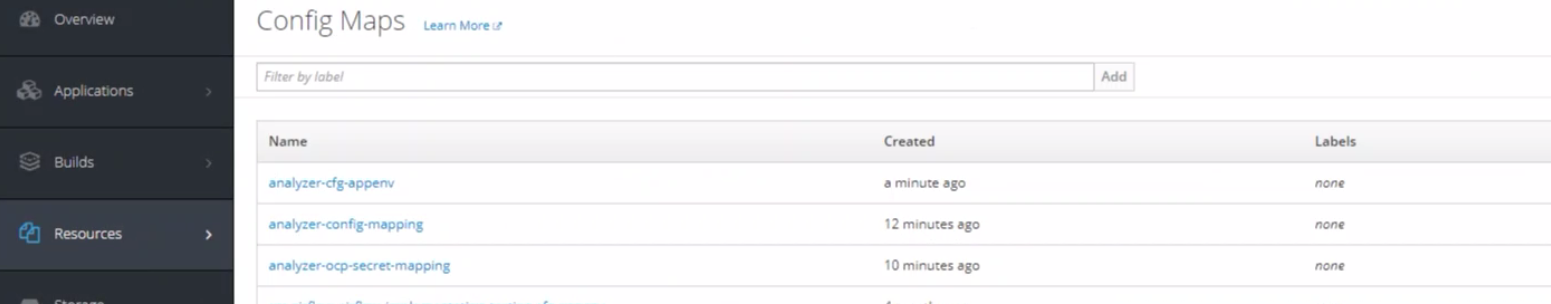
Deploy:  
oc process -f openshift/deploy\_templates/deploymentconfig\_template.yml --ignore-unknown-parameters=true -p APP\_NAME=analyzer  -p TAG=%TAG% -p IMAGE=docker-registry.default.svc:5000/dev-cl1-sif0-ngdp/analyzer --param-file ocp\_configs/%DEPLOY\_ENV%/ocp\_params.txt | oc apply -f -

**Analyzer build:**

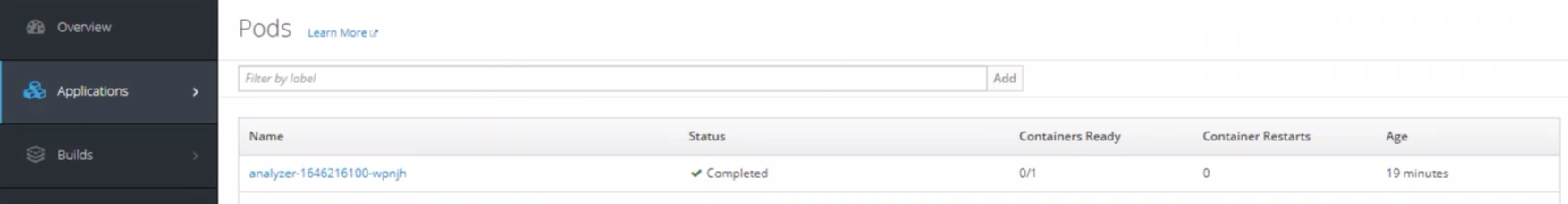


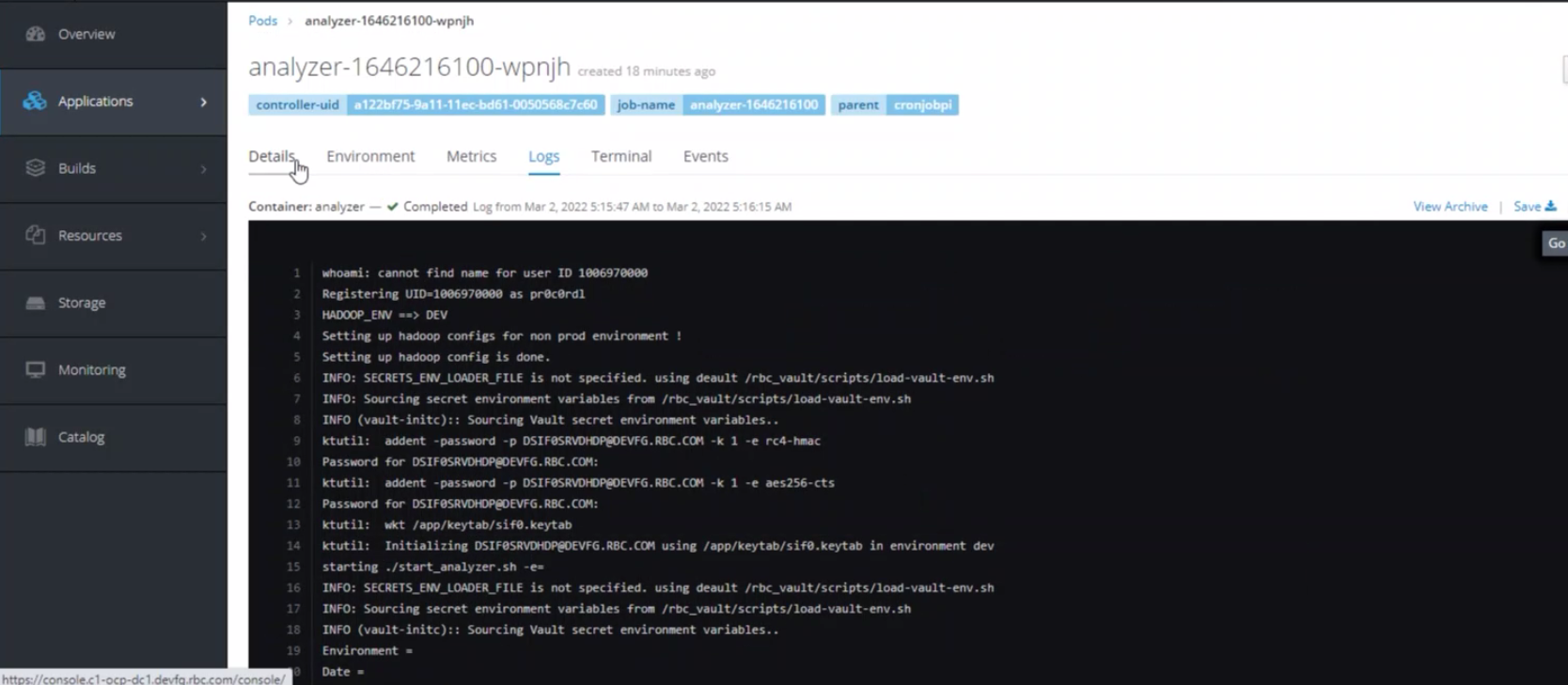


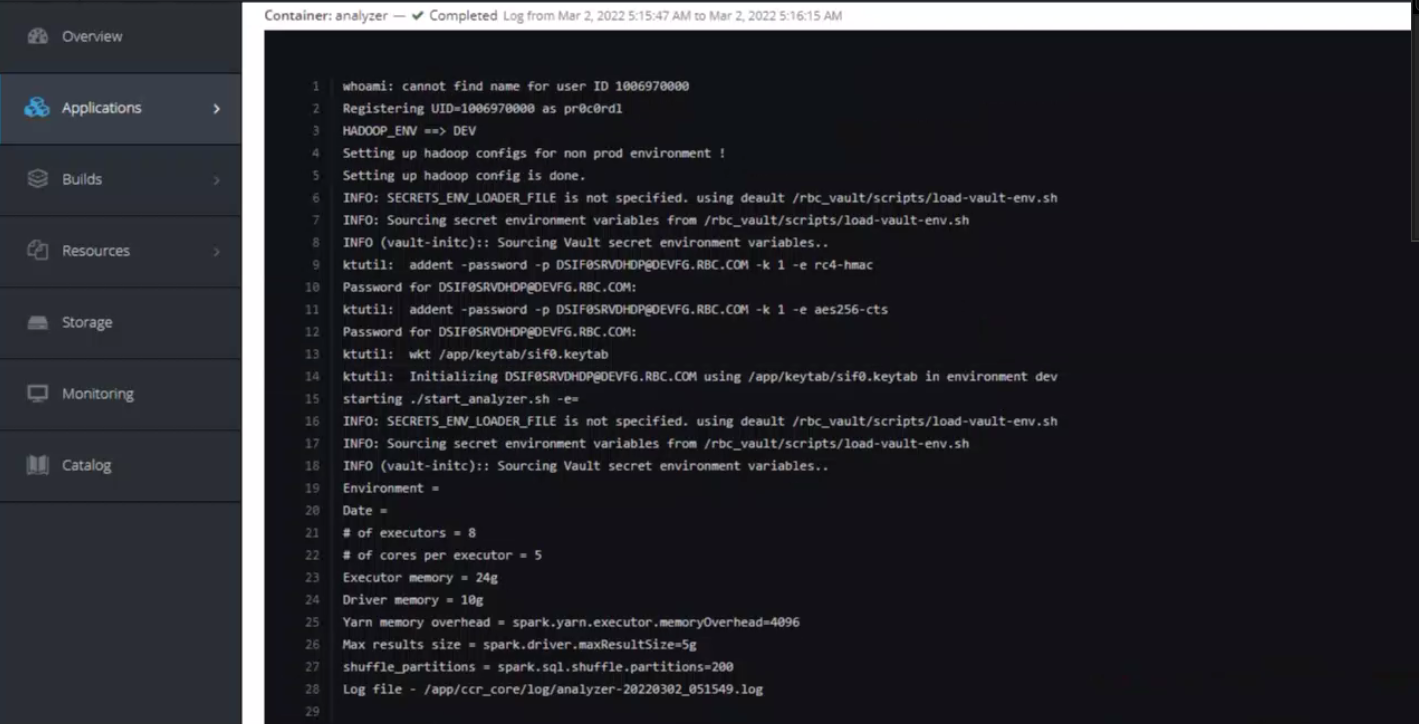
**Analyzer deployment:**



**Analyzer job:**

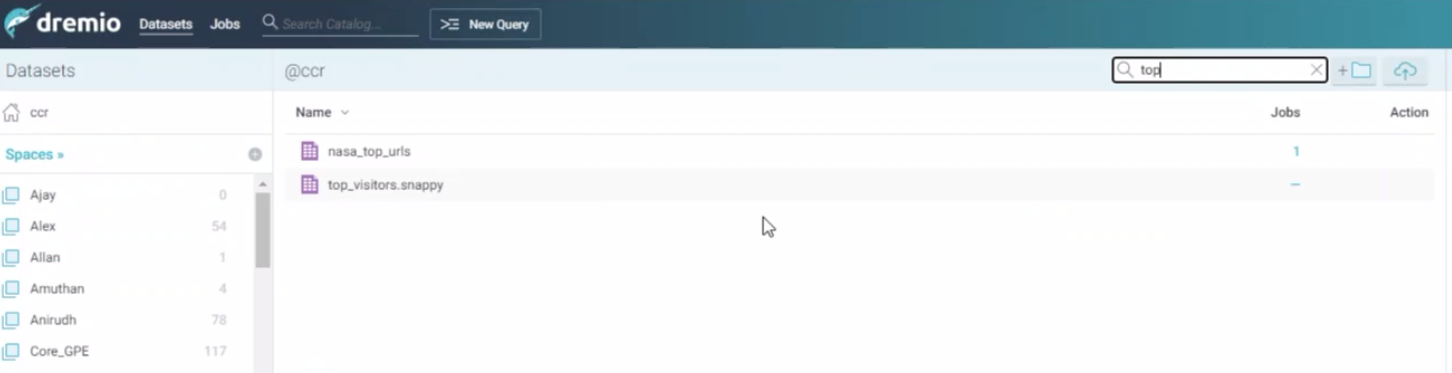




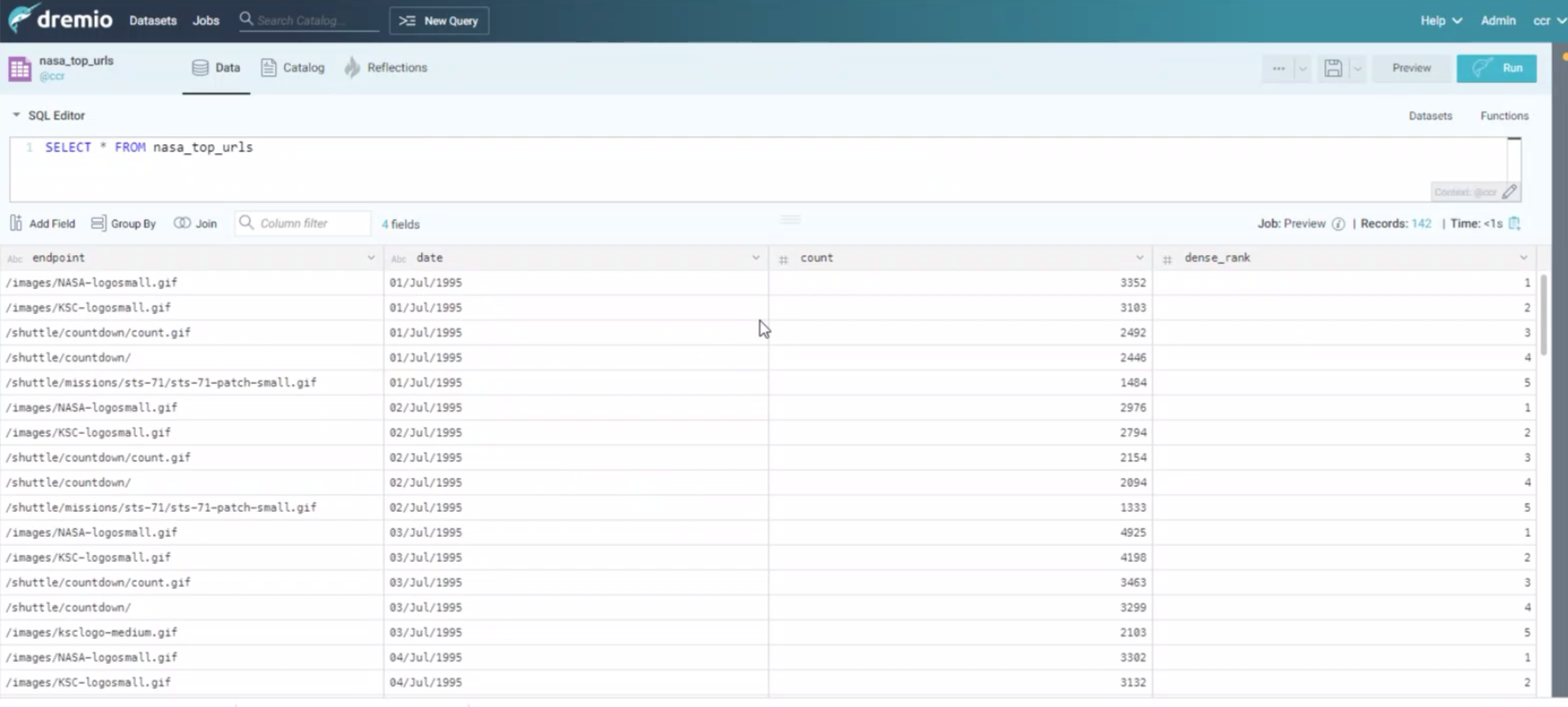




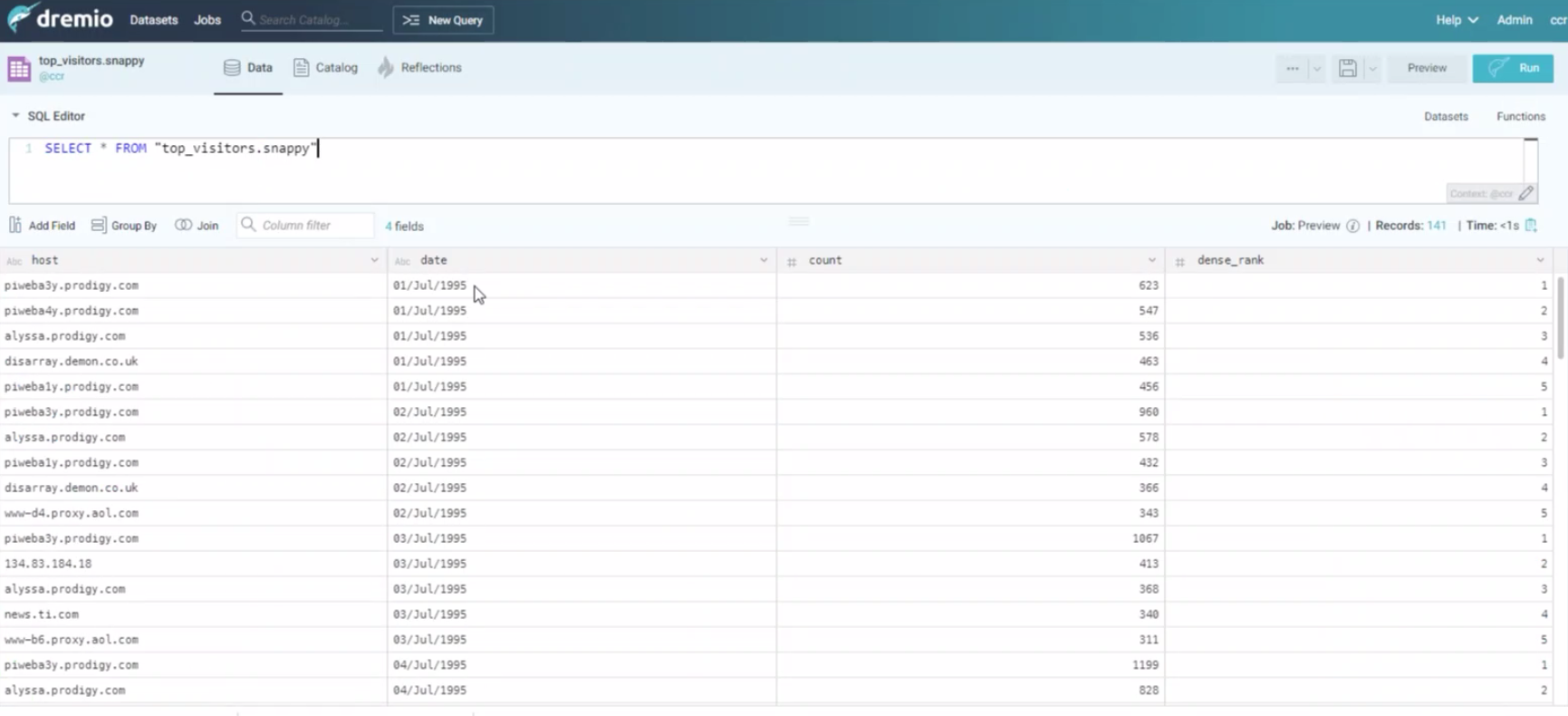
Dermio:



Top URLS by date:



Top visitors by date:



Input file validation:

