

### xtra tutorial: Working with Oracle Data Visualization Desktop 3.0 and Spark

FINISHED

This tutorial was built for BDCS-CE version 17.3.5-20 and Data Visualization Desktop 3.0 as part of the New Data Lake User Journey: here (<https://github.com/oracle/learning-library/tree/master/workshops/journey2-new-data-lake>). Questions and feedback about the tutorial: [david.bayard@oracle.com](mailto:david.bayard@oracle.com) (<mailto:david.bayard@oracle.com>)

Oracle Data Visualization Desktop ( here (<https://docs.oracle.com/middleware/bidv1221/desktop/index.html>) ) is a lightweight, single-file download tool to easily analyze data. Data Visualization Desktop can connect to a variety of data sources. In this tutorial, we will show you how you can setup the Spark thrift server in BDCS-CE so that DVD can connect to it.

NOTE: As of DVD4.0, there is now a native DVD connection type called "Oracle Big Data Cloud" built for BDCS-CE. If you use that, you do NOT need to follow these instructions. These instructions are for use when you want to use the older "Spark" connection type.

Took 0 sec. Last updated by anonymous at November 16 2017, 11:16:06 AM.

### Configuring the Spark Thrift Server process to use binary transport

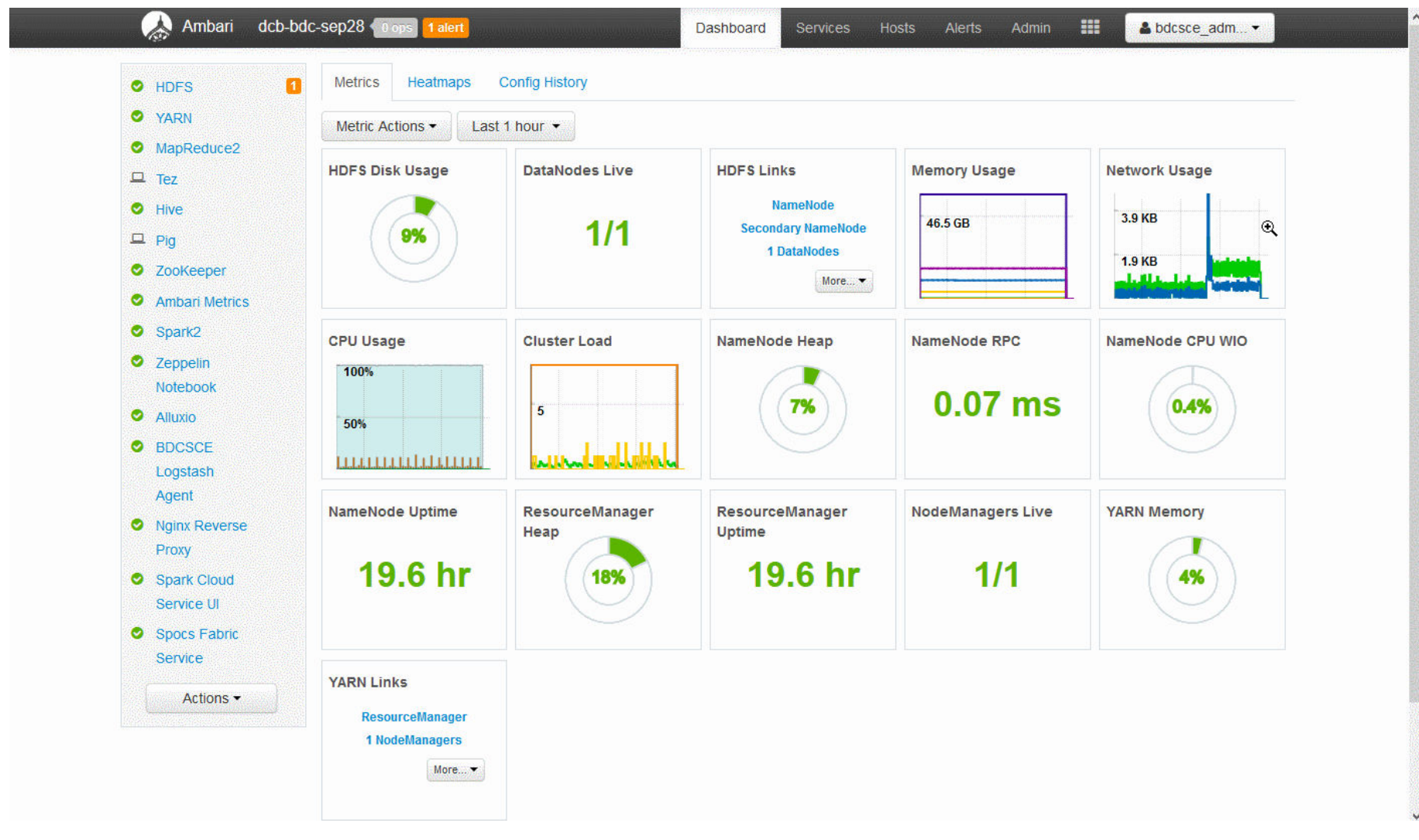
READY

In order to connect to Spark from DVD, we need to configure BDCS-CE's Spark Thrift Server to use the binary transport protocol. By default, BDCS-CE's spark thrift server is configured to use the http transport protocol. These changes are done using the Ambari web console.

Here are the steps:

- 1.Follow the note "xtra Connecting to Ambari" to login to Ambari.
- 2.Once connected to Ambari, click on "Spark2" on the left-hand list of services
- 3.Then click on the "Configs" tab
- 4.In the search box, type "server2"
- 5.In the Advanced spark2-hive-site-override section, change the "hive.server2.transport.mode" to binary.
- 6.In the Custom spark2-hive-site-override section, remove the property "hive.server2.thrift.bind.host" (by clicking on the red - symbol)
- 7.Clear out the search box. Then navigate down to the Custom spark2-thrift-sparkconf section
- 8.In the Custom spark2-thrift-sparkconf section, click on the "Add Property..." link and then add the property "spark.sql.shuffle.partitions=4" and click Add.
- 9.Expand the Advanced spark2-env section and change spark\_daemon\_memory to 2048 MB. **This step is not yet shown in the animation.**
- 10.Also in the Advanced spark2-env section in the "content" field, add the following uncommented line: **This step is not yet shown in the animation.**  
SPARK\_EXECUTOR\_MEMORY="2G"
- 11.Click Save at the top of the screen.
- 12.In the notes field, enter "switch to binary transport"
- 13.Click save again
- 14.If you see a "Configurations" pop-up, click "Proceed Anyway"
- 15.Click OK to acknowledge that changes were made successfully
- 16.Then click Restart, then Restart All Affected

17. Then click Confirm Restart All



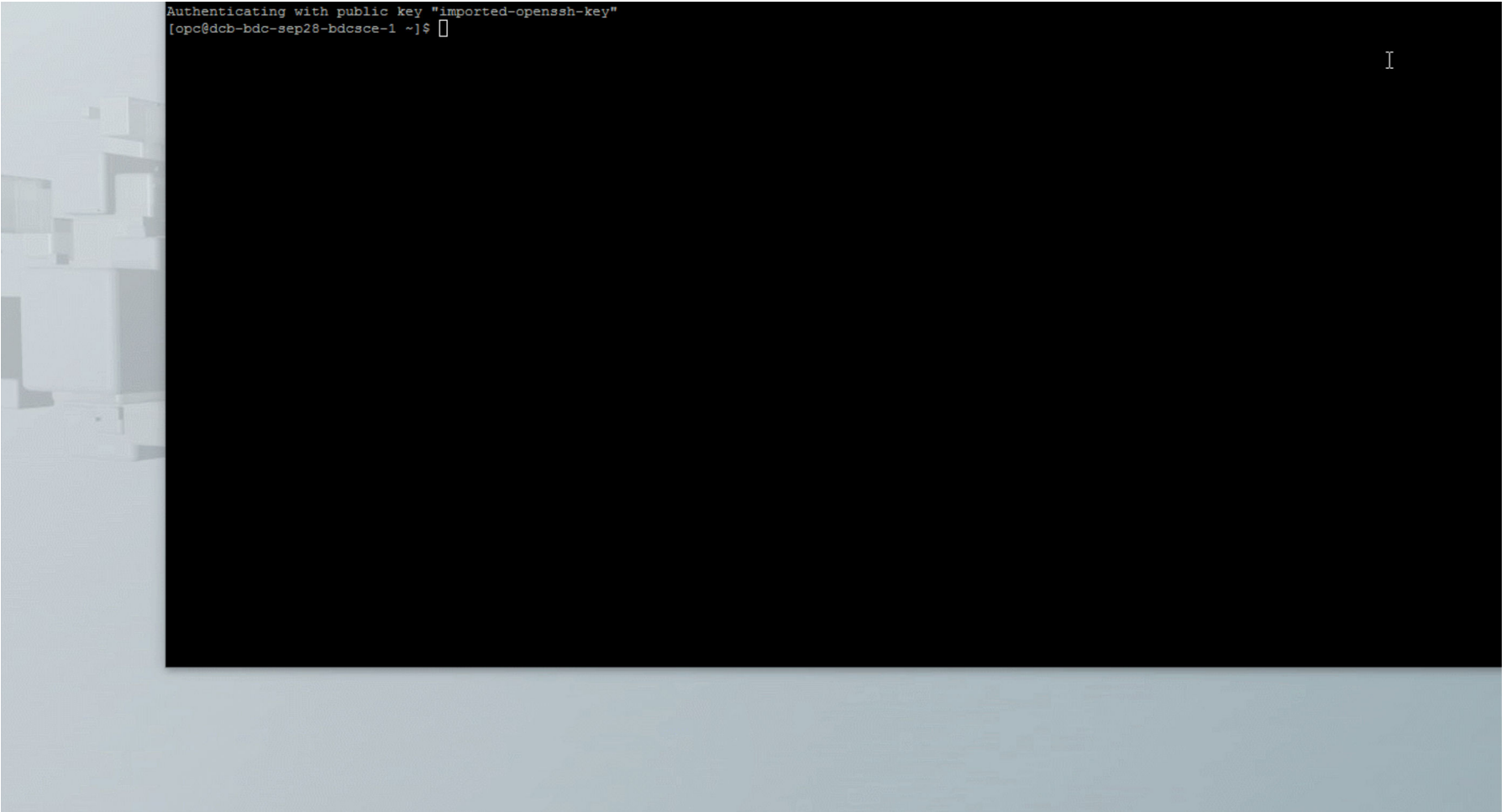
## Connecting to the Spark Thrift Server port (10016)

READY

Now, you need to decide how you want to connect to the Spark Thrift Server port, which is port 10016. You can either choose to use a SSH tunnel (which is very secure) or choose to open port 10016 to the outside world (which can be less secure).

- If you want to use a SSH tunnel, refer to the note "xtra Connecting to Ambari" which has an example of setting up a SSH tunnel (but you would use port 10016 instead of Ambari's 8080).



A terminal window with a dark background and light text. The text shows an SSH authentication process using a public key. The prompt indicates the user is 'opc' on a host named 'dcb-bdc-sep28-bdcsce-1'.

```
Authenticating with public key "imported-openssh-key"
[opc@dcb-bdc-sep28-bdcsce-1 ~]$
```

- If instead of a SSH tunnel, you want to open up port 10016 to the internet, then you will need to create a new access rule for port 10016. Refer to the note “xtra Connecting via SSH” or “OEHCS Tutorial 1” for examples of working with network access rules.

## Define a connection in DV Desktop for the Spark connection

READY

- Open up DV Desktop
- Click on Data Sources
- Click on Connection (Under Create)
- Click on Spark
- Enter the Connection Name
- Enter the Host Name. If you are using SSH tunneling, then enter 127.0.0.1 or localhost. If you have opened up port 10016, then use the IP for your BDCSCE instance.

- Enter the Port. It should now be 10016.
- Enter spark for the username
- Enter x for the password

Create Connection

Spark Database

\*New Connection Name: sparkSep29

\*Host: 127.0.0.1

\*Port: 10016

\*Username: spark

\*Password: x

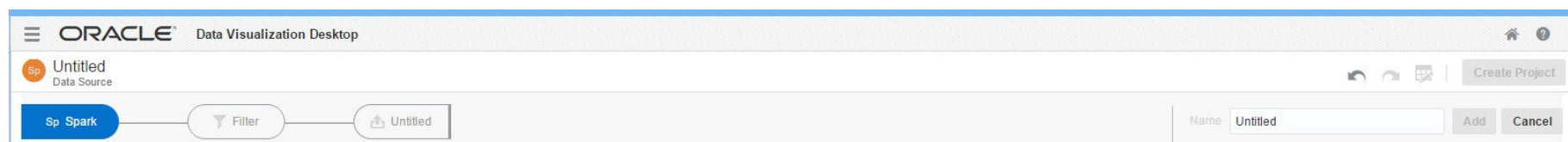
Save Cancel

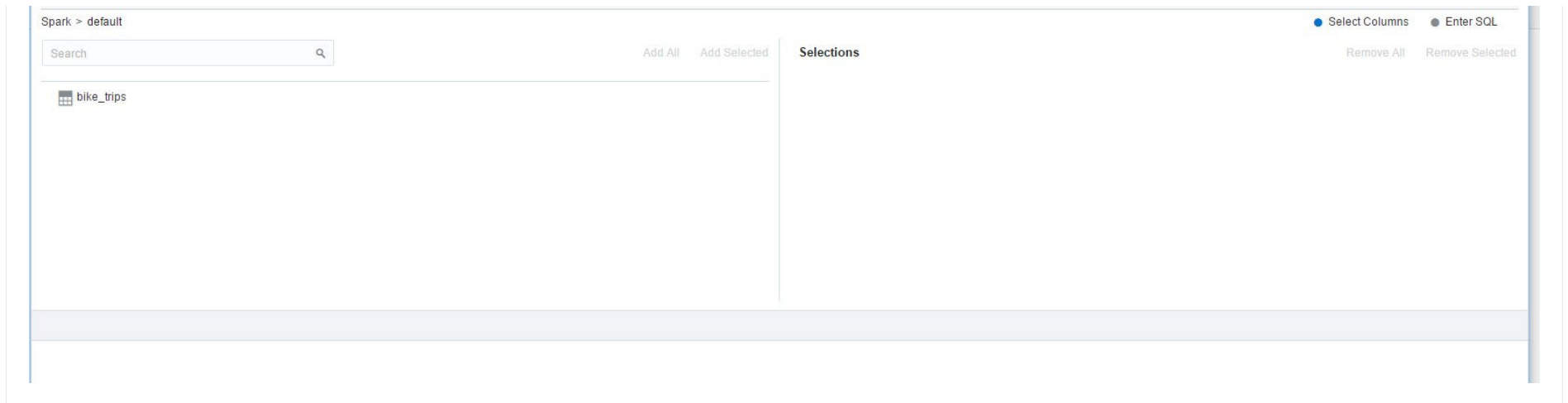
## Create a DV Desktop Data Source for your connection

READY

**NOTE:** With BDCS 17.3.5 and DVD3, you might need to follow the workaround in the following paragraph

- Invoke the pop-up menu on your new Data Source and choose Create Data Source
- Navigate through the database, tables, and columns to choose the elements you want to add.
- Once you have selected your table and columns, click on the rightmost icon in the dataflow pipeline (it will be the icon after the filter icon). Then, click on the Refresh property. Change this to be "Live - Always use the database".
- Name the new data source and Add it





## Tip - DVD 3.0 with Spark 2.1 no databases appear

READY

This may be <https://issues.apache.org/jira/browse/SPARK-9686> (<https://issues.apache.org/jira/browse/SPARK-9686>)

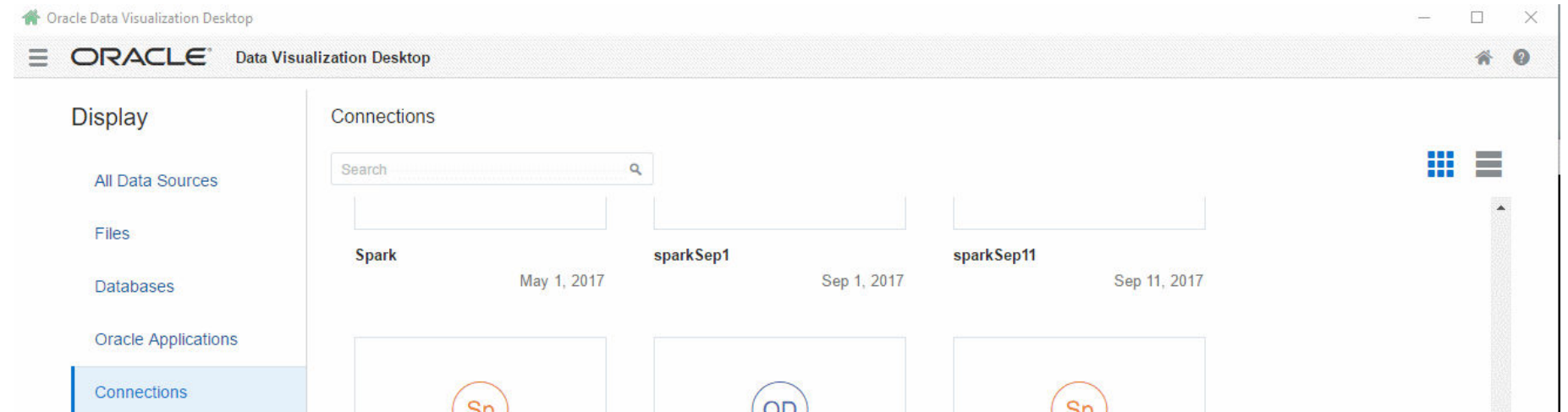
In any case, we have noticed that no databases appear when querying Spark2.1 from DVD 3.0.

There is a workaround...

When you create your data source, use the "Enter SQL" feature to define your sql. It can be as simple as a "select \* from tablename" or more complex.

And once you have entered your sql, be sure to click on the rightmost icon in the dataflow pipeline (it will be the icon after the filter icon). Then, click on the Refresh property. Change this to be "Live - Always use the database".

Here is a video:



Data Flows

Create

Data Source

Connection

Data Flow

Data Source Storage

311.5MB of 100GB used, 0B selected

sparkSep14

Sep 14, 2017

sparkSep1odbc2

Sep 1, 2017

sparkSep29

5 minutes ago

sparktest

Jul 28, 2017

test

Sep 1, 2017

xxx

Sep 1, 2017

yyy

Sep 1, 2017

zzz

Sep 1, 2017

# Tip - Tracking Spark queries

READY

Run the following shell paragraph to peak at queries sent to the Spark thrift server.

## Shell command to peak at queries sent to Spark Thrift Server

READY

```
%sh
egrep '$Running|\x0d|limit' /data/var/log/spark2-thrift/spark-hive-org.apache.spark.sql.hive.thriftserver.HiveThriftServer2-1-*--1.out | tail -400
```

READY

READY

6 of 6

11/16/2017 11:17 AM