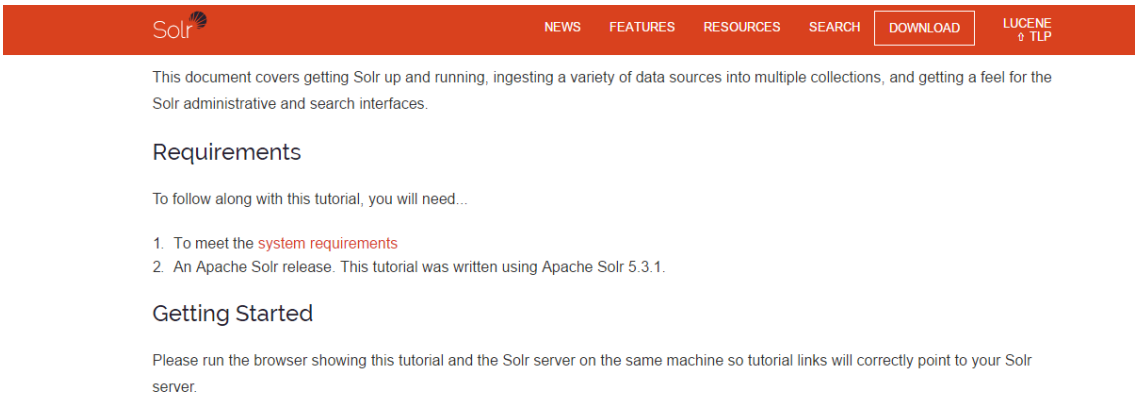
Installation Instructions for Solr

Solr (pronounced "solar") is an open source enterprise search platform, written in Java, from the Apache Lucene project. It provides distributed indexing, replication and load-balanced querying, automated failover and recovery, centralized configuration and more. Its home page is located at http://lucene.apache.org/solr . The core technology underlying Apache Solr is Lucene. Lucene was developed and open sourced by Doug Cutting in 2000 and has evolved and matured since then with a strong online community. In order to use Lucene directly, one writes code to store and query an index stored on a disk. Solr is considered as the server modification of Lucene.

If you are working on a Windows machine begin by starting your virtual Ubuntu machine using Oracle Virtual Box. On a Mac, Solr can be installed directly.

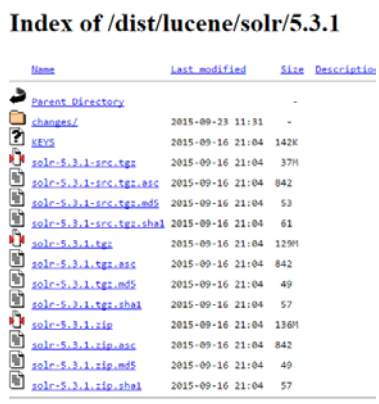
Please make sure that you have JAVA SE 7 or higher in your system, verify using java –version command.

1. Go to http://lucene.apache.org/solr/quickstart.html and click on Download on the top right corner



2. You will be redirected to the following site, click on the suggested mirror site

3. You will see the following directory structure, download the solr-5.3.1.zip folder.



4. Once the folder has been downloaded. Unzip the folder using the command **unzip –q solr-5.3.1.zip** on Windows or else on a Mac it should unzip automatically. You should probably place the solr-5.3.1 folder in your root directory, e.g. ~yourlogin/solr-5.3.1.

Directory Layout of Solr

All examples are contained in the example/exampledocs folders. Solr's home directory includes the following folders of special interest:

- *bin*: these are the binary files
- *dist:* these are some helper jar files
- *server/solr/configsets/basic_configs/conf*: Contains files which help set the Solr configurations, these include:

  - *server/solr/configsets/basic_configs/conf/schema.xml*: This is the schema for the index including field type definitions for given dataset.

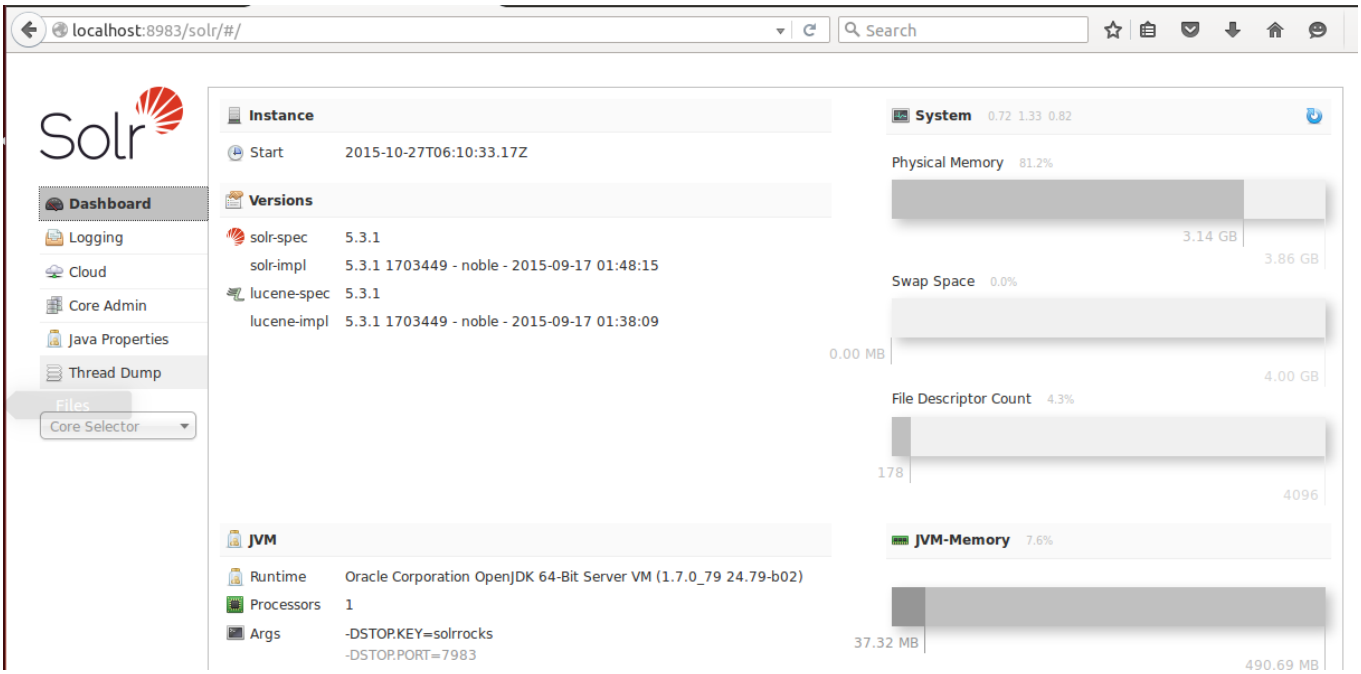
  - *server/solr/configsets/basic_configs/conf/solrconfig.xml*: This is the primary Solr configuration file, which includes port configurations, timeout configurations etc.

- *server/solr/<core_name>:* Solr folder contains the different cores you have created. The folder corresponding to <core_name> includes:
  - conf: Contains the schema, config files, by default these are copied from the basic_configs folder mentioned above. Any schema changes that you want, modifications have to be made to the config files in the folder specific to the core followed by a core reload.
  - core.properties: this file contains the core properties. Basically core name, the configuration directory if not defaulted, the data directory name etc
  - data: Contains the indexed data and logs related to the core

5. Now cd into the solr-5.3.1 folder that has been unzipped and run the command **bin/solr start -e cloud –noprompt**. This will launch the solr server on port 8983. You should see the following message if the launch was successful.

```
SolrCloud example running, please visit: http://localhost:8983/solr
```

6. Go to http://localhost:8983/solr and you should see the solr UI.



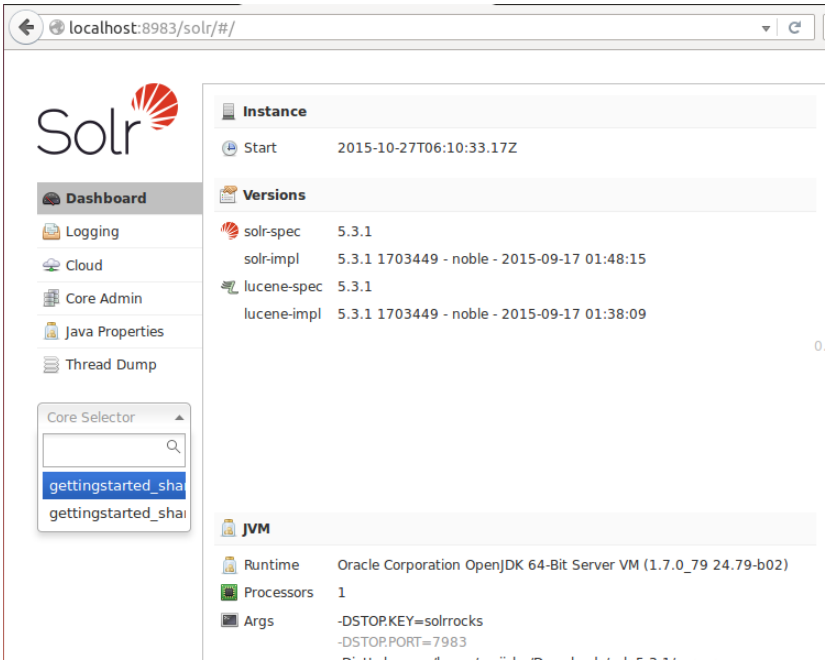
7. Let's index some files. Enter the command **bin/post –c gettingstarted example/exampledocs.** This command basically indexes all the files in the folder example/exampledocs into the collection gettingstarted. gettingstarted is the default collection that gets created when you start the solr server.

   *You can create your own core and define the number of shards/nodes to replicate the data by using the following command **bin/solr create –c <name_of_new_core>** when the solr server is running or by entering the following url in your browser*
   http://localhost:7574/solr/admin/collections?action=CREATE&name=<name_of_new_core>&numShards=1&replicationFactor=1&maxShardsPerNode=1&collection.configName=<name_of_new_core>
   e.g. Replace <name_of_new_core> with "myexample" -> bin/solr create –c myexample
   Posting to this new core will be similar to the example above - bin/post –c myexample example/exampledocs

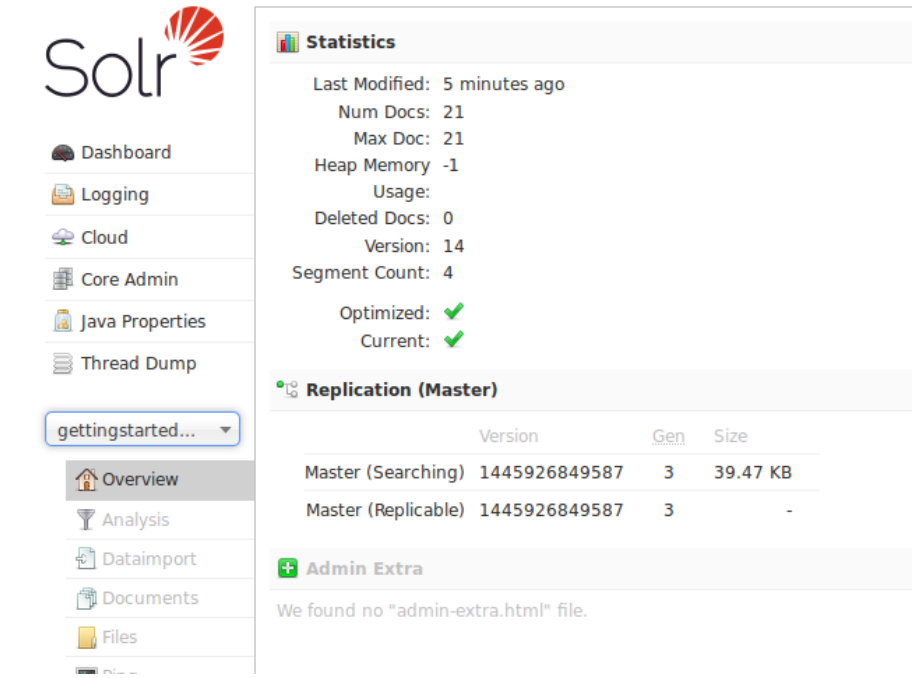
   After completion you should be able to see a similar screen.

```
SimplePostTool version 5.0.0
Posting files to [base] url http://localhost:8983/solr/gettingstarted/update...
Entering auto mode. File endings considered are xml,json,csv,pdf,doc,docx,ppt,pptx,xls,xlsx,odt,odp,ods,ott,otp,ots,rtf,htm,html,txt,log
Entering recursive mode, max depth=999, delay=0s
Indexing directory example/exampledocs (18 files, depth=0)
POSTing file solr.xml (application/xml) to [base]
POSTing file sample.html (text/html) to [base]/extract
POSTing file mem.xml (application/xml) to [base]
POSTing file ipod_video.xml (application/xml) to [base]
POSTing file books.csv (text/csv) to [base]
POSTing file monitor.xml (application/xml) to [base]
POSTing file solr-word.pdf (application/pdf) to [base]/extract
POSTing file utf8-example.xml (application/xml) to [base]
POSTing file money.xml (application/xml) to [base]
POSTing file vidcard.xml (application/xml) to [base]
POSTing file gb18030-example.xml (application/xml) to [base]
POSTing file ipod_other.xml (application/xml) to [base]
POSTing file manufacturers.xml (application/xml) to [base]
POSTing file monitor2.xml (application/xml) to [base]
POSTing file sd500.xml (application/xml) to [base]
POSTing file mp500.xml (application/xml) to [base]
POSTing file books.json (application/json) to [base]
POSTing file hd.xml (application/xml) to [base]
18 files indexed.
COMMITting Solr index changes to http://localhost:8983/solr/gettingstarted/update...
Time spent: 0:00:52.974
```

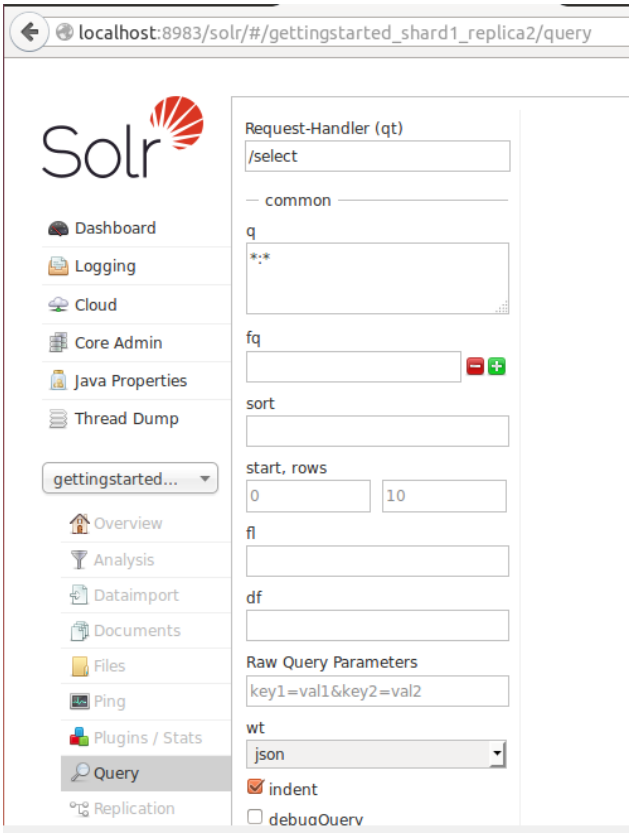
8. Now go to the Solr UI at http://localhost:8983/solr and in the Core Selector dropdown box, select the first entry-gettingstarted_shard1_replica2. These collections were created automatically by solr during launch.



9. After the selection, you will see the statistics below as follows, which confirms that the files that we tried to index have been successfully indexed.



10. Now select the query option from the list.

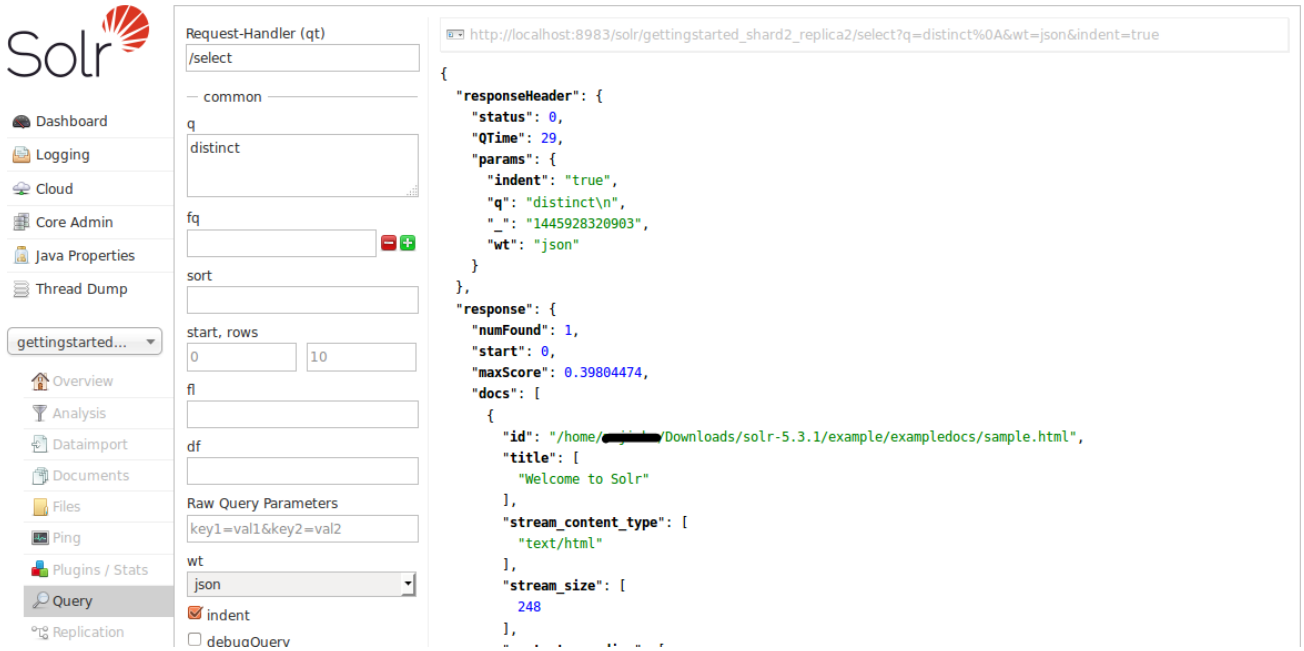


11. Let us try to perform a simple query. First let's check the content of one of the html files in the exampledocs folder – sample.html

sample.html

```
<html>
<head>
  <title>Welcome to Solr</title>
</head>
<body>
<p>
  Here is some text
</p>
<p>distinct<br/>words</p>
<div>Here is some text in a div</div>
<div>This has a <a href="http://www.apache.org">link</a>.</div>
</body>
</html>
```

12. Now let us try querying our index for the word distinct and see if this document comes up in the result. In the input box labeled q, enter the search term – "distinct", and then click on "Execute Query"



The response is in json format by default. On the right side of the screen, you will see the result of your query. In our example, this returned a single document, the sample.html file.

13. You can stop the server using the following command : **bin/solr stop -all**

```
Sending stop command to Solr running on port 7574 ... waiting 5 seconds to allow Jetty process 2621 to stop gracefully.
Sending stop command to Solr running on port 8983 ... waiting 5 seconds to allow Jetty process 2411 to stop gracefully.
```

Note: if you want to start over and remove all previously indexed data, just cd into solr home directory and enter the following command

bin/solr delete –c <core_name>

This will delete all the logs, cores, and indexed data. Therefore be careful, only if you wish to start over use this command.