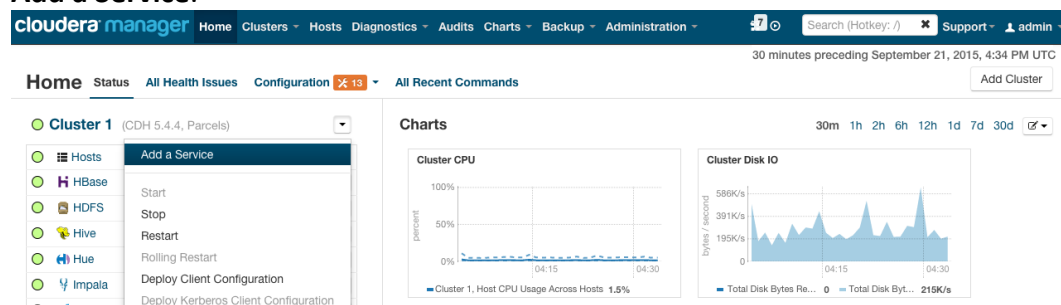# Solr Installation

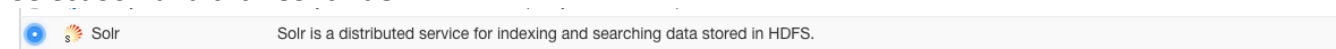## September 2015

HDFS and Zookeeper services are 2 pre-requisites for Solr. Go to Cloudera Manager main page and click **Add a Service**.
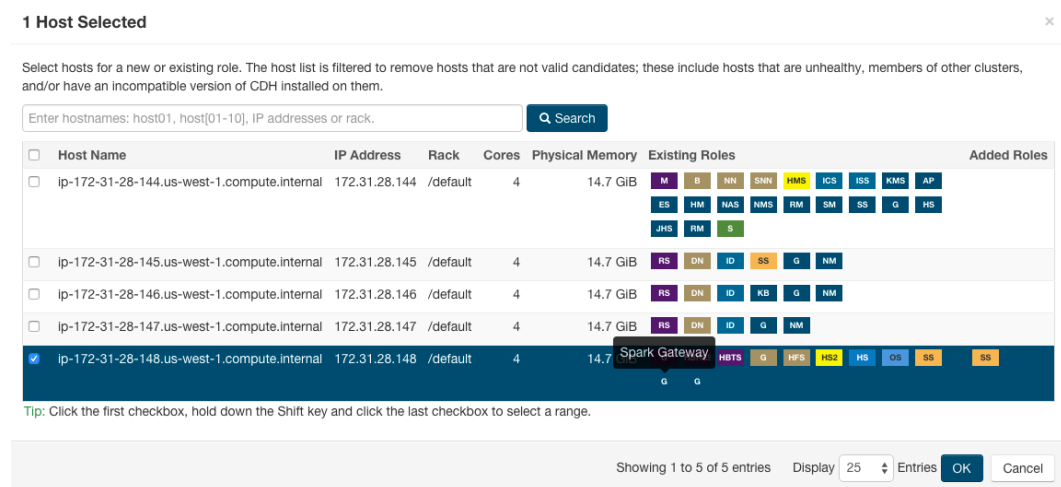

Select Solr and click **Continue**.

Select HDFS and Zookeeper, and select one or more machines to host Solr server.

Cloudera Manager starts Solr servers.
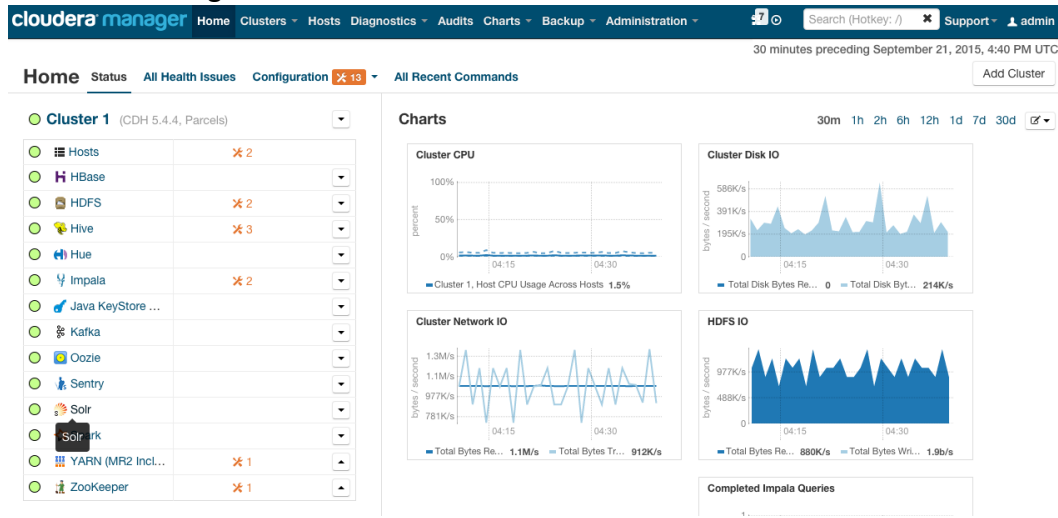
Post verification

Test Solr functionality with Solr utilities and curl tool (Example).

./create.sh

```sh
#!/bin/sh

ZK="172.31.28.144"
COLLECTION="test"

echo "create solr collection"
rm -rf tmp/*
solrctl --zk $ZK:2181/solr instancedir --generate tmp/${COLLECTION}_configs
cp template/schema.xml tmp/${COLLECTION}_configs/conf/
solrctl --zk $ZK:2181/solr instancedir --create $COLLECTION tmp/${COLLECTION}_configs
solrctl --zk $ZK:2181/solr collection --create $COLLECTION -s1 -r1
solrctl --zk $ZK:2181/solr collection –list
```

./delete.sh

```sh
#!/bin/sh

ZK="172.31.28.144"
COLLECTION="test"

echo "delete solr collection"
solrctl --zk $ZK:2181/solr collection --delete $COLLECTION
solrctl --zk $ZK:2181/solr instancedir --delete $COLLECTION
rm -rf tmp/*
```

template/schema.xml

```xml
<?xml version="1.0" encoding="UTF-8" ?>

<schema name="example" version="1.5">

<fields>
<field name="id" type="string" indexed="true" stored="true" required="true" multiValued="false"/>
<field name="doc_name" type="string" indexed="true" stored="true"/>
<field name="doc_type" type="string" indexed="true" stored="true"/>
<field name="doc_text" type="text_ws" indexed="true" stored="true"/>
<field name="_version_" type="long" indexed="true" stored="true"/>
</fields>

<uniqueKey>id</uniqueKey>
<types>
<fieldType name="string" class="solr.StrField" sortMissingLast="true"/>
<fieldType name="int" class="solr.TrieIntField" precisionStep="0" positionIncrementGap="0"/>
<fieldType name="float" class="solr.TrieFloatField" precisionStep="0" positionIncrementGap="0"/>
<fieldType name="long" class="solr.TrieLongField" precisionStep="0" positionIncrementGap="0"/>
<fieldType name="double" class="solr.TrieDoubleField" precisionStep="0" positionIncrementGap="0"/>
<fieldType name="text_ws" class="solr.TextField" positionIncrementGap="100">
<analyzer>
<tokenizer class="solr.WhitespaceTokenizerFactory"/>
</analyzer>
</fieldType>
</types>
</schema>
```

sample/data.json

```json
[{"id":"doc1", "doc_name":"spark", "doc_type":"word", "doc_text":"alex spark"}, {"id":"doc2", "doc_name":"impala", "doc_type":"pdf", "doc_text":"aleximpala"}]
```

./update.sh

```
curl -i 'http://172.31.28.148:8983/solr/test/update/json?commit=true' --data-binary @sample/data.json -H 'Content-type:application/json'
```

./query.sh

```
curl -i 'http://172.31.28.148:8983/solr/test/select?q=*%3A*&wt=json&indent=true'
```

Enable Kerberos in Solr.

Modified version of curl commands to work with Kerberos'edSolr (kinit first before running commands below).

./update.sh
curl -i --negotiate -u : 'http://172.31.28.148:8983/solr/test/update/json?commit=true' --data-binary @sample/data.json -H 'Content-type:application/json'
./query.sh
curl -i --negotiate -u : 'http://172.31.28.148:8983/solr/test/select?q=*%3A*&wt=json&indent=true'