GettingStartedYARN

Getting Started with Hama on YARN

Preparations

Current Hama and Hadoop requires JRE 1.7 or higher and ssh to be set up between nodes in the cluster:

- Hadoop-2.x
- Sun Java JDK 1.7 or higher version

For additional information consult our CompatibilityTable.

This tutorial requires Hadoop 2.x already correctly installed. If you haven't done this yet, please follow the official documentation https://hadoop.apache.org/docs/stable/

Configuration

Only two properties which are resource manager address and default filesystem uri is essentially needed for Hama on YARN. The sample configuration is as follows:

```
<!-- Path to your hama-site.xml -->
<configuration>
    <property>
        <name>yarn.resourcemanager.address</name>
        <value>'your resource manager address or hostname':'resource manager port'</value>
    </property>
    <property>
        <name>fs.default.name</name>
        <value>hdfs://'your default file system address or hostname':'default file system port'/</value>
    </property>
</configuration>
```

See also configuration page for advanced configurations of Hama.

Advanced Properties

Property Name	Default	Meaning
hama.appmaster.memory.mb	100 mb	The amount of memory used by the BSPApplicationMaster. The total amount of memory used by the ApplicationMaster is calculated as follows. *memoryInMb = 3 * BSP_TASK_NUM + hama.appmaster.memory.mb*. This is because the application master spawns 1-3 thread per launched task that each should take 1mb, plus a minimum of base memory usage of 100. If you face memory issues, you can set this to a higher value.

Launching Hama on YARN

Launch Hama application which is serialize printing example:

```
$HAMA_HOME/bin/hama jar hama-yarn-0.7.0.jar org.apache.hama.bsp.YarnSerializePrinting
```

You should see "Hello BSP" Messages which each container spawned in HDFS where you defined output path in your terninal. If your application is success, you'll be able to got the following message.

```
INFO bsp.YARNBSPJobClient: Application has completed successfully. Breaking monitoring loop
Hello BSP from 1 of 4: cluster-0:16004
Hello BSP from 2 of 4: cluster-1:16006
Hello BSP from 3 of 4: cluster-0:16008
Hello BSP from 4 of 4: cluster-1:16010
Job Finished in 14.838 seconds
```

How to write a Hama-YARN job

BSP job

The BSPModel hasn't changed, but the way to submit a job has.

Basically you just need the following code to submit a Hama-YARN job.

```
    HamaConfiguration conf = new HamaConfiguration();
    YARNBSPJob job = new YARNBSPJob(conf);

    job.setBspClass(HelloBSP.class);
    job.setJarByClass(HelloBSP.class);
    job.setJobName("Serialize Printing");
    job.setMemoryUsedPerTaskInMb(50);
    job.setNumBspTask(2);
    job.waitForCompletion(false);
```

As you can see, instead of a `BSPJob` you are starting a `YARNBSPJob`.

The `YARNBSPJob` offers an extended API for running on YARN. For example you can set the amount of memory used by a task with

```
    job.setMemoryUsedPerTaskInMb(50);
```

Graph job

Hama Graph jobs also isn't changed but you should change a little code from `GraphJob` to `YARNGraphJob` to run Hama graph job. Let's show the following link, PageRank on YARN.

PageRank on YARN example

Compared to PageRank in existing graph example, this code only is changed from `GraphJob` to `YARNGraphJob`. How to launch graph job on YARN is same as existing graph job.

How to submit a job

General

You have to ways to submit a job, you can either submit it via shell and a packed jar, or you can submit from a java application. In both cases you need the hama-yarn jar in the classpath or inside the jar to run correctly.

Via Shell

```
bin/yarn jar /path_to_jar org.apache.hama.bsp.YarnSerializePrinting
```

In this case the jar in `/path_to_jar` contains the hama-yarn jar or it is already in the classpath of your Hadoop application. You have to replace `org.apache.hama.bsp.YarnSerializePrinting` with the class which contains the main method which runs the Hama Job.

Via Java Application

Just like in the section above, you have to configure the address of the ResourceManager. Then you can run this from a Java Application, just put it into a main-method.

```
HamaConfiguration conf = new HamaConfiguration();
conf.set("yarn.resourcemanager.address", "0.0.0.0:8040");

YARNBSPJob job = new YARNBSPJob(conf);
job.setBspClass(HelloBSP.class);
job.setJarByClass(HelloBSP.class);
job.setJobName("Serialize Printing");
job.setMemoryUsedPerTaskInMb(50);
job.setNumBspTask(2);
job.waitForCompletion(false);
```

How to change existing Hama Jobs to run on YARN

In case you have the following code

```
// BSP job configuration
HamaConfiguration conf = new HamaConfiguration();
BSPJob bsp = new BSPJob(conf);
bsp.waitForCompletion(true);
```

to submit a Hama job. You can just change the `BSPJob` to `YARNBSPJob`. If you want to submit graph job of Hama, only change `BSPJob` object to `YARNGraphJob`.