

Getting Started

- [Requirements](#)
- [Download](#)
- [Build latest version from source](#)
- [Hadoop Installation](#)
- [Hama Installation](#)
- [Startup script](#)
- [Configuration files](#)
- [Setting up Hama](#)
- [Modes](#)
 - [Local Mode](#)
 - [Pseudo Distributed Mode](#)
 - [Distributed Mode](#)
- [Settings](#)
 - [Starting a Hama cluster](#)
 - [Stopping a Hama cluster](#)
 - [Enabling Fault Tolerance Service](#)
- [Run the BSP Examples](#)
- [Hama Web Interfaces](#)
- [Setup Hama in your Eclipse Workspace](#)

Requirements

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

- hadoop 1.x or 2.x
- Sun Java JDK 1.7.x or higher version

For additional information consult our [Compatibility Table](#).

Download

You can download Hadoop here:

<http://www.apache.org/dyn/closer.cgi/hadoop/core/>

You can download Hama here:

<http://www.apache.org/dyn/closer.cgi/hama>

Build latest version from source

If you're going to use latest (unreleased) version, you can check out TRUNK and build it with maven 3 as following commands:

```
% svn co https://svn.apache.org/repos/asf/hama/trunk hama-trunk
% cd hama-trunk

% mvn clean install -Phadoop1 -Dhadoop.version=1.x.x -U
or
% mvn clean install -Phadoop2 -Dhadoop.version=2.x.x -U
```

See also [HowToContribute](#)

Hadoop Installation

- See <http://hadoop.apache.org/docs/stable/index.html>
- If you use the Cloudera's CDH, you should replace hadoop and its dependencies in \${HAMA_HOME}/lib folder. For example,

```
% rm -rf ./lib/hadoop*.jar
% cp /usr/lib/hadoop/hadoop-test-0.20.2-cdh3u3b.jar ./lib/
% cp /usr/lib/hadoop/hadoop-core-0.20.2-cdh3u3b.jar ./lib/
% cp /usr/lib/hadoop/lib/guava-r09-jarjar.jar ./lib/

% bin/start-bspd.sh
```

Hama Installation

Untar the files to your destination of choice:

```
tar -xzf hama-0.x.0.tar.gz
```

Don't forget to `chown` the directory as the same user you configured Hadoop in the step before.

Startup script

The `$HAMA_HOME/bin` directory contains some script used to start up the Hama daemons.

- `start-bspd.sh` - Starts all Hama daemons, the BSPMaster, GroomServers and Zookeeper.

Note: You have to start Hama with the same user which is configured for Hadoop.

Configuration files

The `$HAMA_HOME/conf` directory contains some configuration files for Hama. These are:

- `hama-env.sh` - This file contains some environment variable settings used by Hama. You can use these to affect some aspects of Hama daemon behavior, such as where log files are stored, the maximum amount of heap used etc. The only variable you should need to change in this file is `JAVA_HOME`, which specifies the path to the Java 1.6.x installation used by Hama.
- `groomservers` - This file lists the hosts, one per line, where the GroomServer daemons will run. By default this contains the single entry `localhost`
- `hama-default.xml` - This file contains generic default settings for Hama daemons. **Do not modify this file.**
- `hama-site.xml` - This file contains site specific settings for all Hama daemons and BSP jobs. This file is empty by default. Settings in this file override those in `hama-default.xml`. This file should contain settings that must be respected by all servers and clients in a Hama installation.

Setting up Hama

This section describes how to get started by setting up a Hama cluster.

Modes

Just like Hadoop, we distinct between three modes:

- Local Mode
- Pseudo Distributed Mode
- Distributed Mode

Local Mode

This mode is the default mode if you download Hama ($\geq 0.3.0$) and install it. When submitting a job it will run a local multithreaded BSP Engine on your server. It can be configured via the `bsp.master.address` property to `local`. You can adjust the number of threads used in this utility by setting the `bsp.local.tasks.maximum` property. See the [Settings](#) step how and where to configure this.

Note: In this mode, nothing must be launched via the [start scripts](#).

Pseudo Distributed Mode

This mode is when you just have a single server and want to launch all the daemon processes (BSPMaster, Groom and Zookeeper). It can be configured when you set the `bsp.master.address` to a host address, e.G. `localhost` and put the same address into the `groomservers` file in the configuration directory. As stated it will run a BSPMaster, a Groom and a Zookeeper on your machine.

Distributed Mode

This mode is just like the "Pseudo Distributed Mode", but you have multiple machines, which are mapped in the `groomservers` file.

Settings

- **BSPMaster and Zookeeper settings** - Figure out where to run your HDFS namenode and BSPMaster. Set the variable `bsp.master.address` to the BSPMaster's intended host:port. Set the variable `fs.defaultFS` to the HDFS Namenode's intended host:port.

An example of a `hama-site.xml` file:

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<configuration>
  <property>
    <name>bsp.master.address</name>
    <value>host1.mydomain.com:40000</value>
    <description>The address of the bsp master server. Either the
      literal string "local" or a host:port for distributed mode
    </description>
  </property>

  <property>
    <name>fs.defaultFS</name>
    <value>hdfs://host1.mydomain.com:9000/</value>
    <description>
      The name of the default file system. Either the literal string
      "local" or a host:port for HDFS.
    </description>
  </property>

  <property>
    <name>hama.zookeeper.quorum</name>
    <value>host1.mydomain.com,host2.mydomain.com</value>
    <description>Comma separated list of servers in the ZooKeeper Quorum.
      For example, "host1.mydomain.com,host2.mydomain.com,host3.mydomain.com".
      By default this is set to localhost for local and pseudo-distributed modes
      of operation. For a fully-distributed setup, this should be set to a full
      list of ZooKeeper quorum servers. If HAMA_MANAGES_ZK is set in hama-env.sh
      this is the list of servers which we will start/stop zookeeper on.
    </description>
  </property>
</configuration>
```

If you are managing your own ZooKeeper, you have to specify the port number as below:

```
<property>
  <name>hama.zookeeper.property.clientPort</name>
  <value>2181</value>
</property>
```

- See also [Configuration Properties](#)

Starting a Hama cluster

Skip this step if you're in [Local Mode](#).

Run the command:

```
% $HAMA_HOME/bin/start-bspd.sh
```

This will startup a BSPMaster, GroomServers and Zookeeper on your machine.

Stopping a Hama cluster

Run the command:

```
% $HAMA_HOME/bin/stop-bspd.sh
```

to stop all the daemons running on your cluster.

Enabling Fault Tolerance Service

By default, FT service is disabled. To enable Fault Tolerance Service, you can set below properties like below:

```

<property>
  <name>bsp.ft.enabled</name>
  <value>true</value>
  <description>Enable Fault Tolerance in BSP Task execution.</description>
</property>
<property>
  <name>bsp.checkpoint.enabled</name>
  <value>true</value>
  <description>Enable Hama to checkpoint the messages transferred among BSP tasks during the BSP
synchronization period.</description>
</property>
<property>
  <name>bsp.checkpoint.interval</name>
  <value>10</value>
  <description>If bsp.checkpoint.enabled is set to true, the checkpointing is initiated on the valueth
synchronization process of BSP tasks.</description>
</property>

```

Run the BSP Examples

Run the command:

```
% $HAMA_HOME/bin/hama jar hama-examples-0.x.0.jar
```

It will then offer you some examples to choose. Refer to our [Examples](#) site if you have additional questions how to use them.

Hama Web Interfaces

The web UI provides information about BSP job statistics of the Hama cluster, running/completed/failed jobs.

By default, it's available at <http://localhost:40013>

Setup Hama in your Eclipse Workspace

Step by step guide to let Hama run in your eclipse workspace with a localrunner:

*First you need a simple Java Project.

*Click on File in the top left corner -> New -> Java Project.

*Give the child a good name, choose at least Java6 and finish. You should see the project in your Package Explorer.

*Add the jars you need for Hama 0.5.0 to your build path, you should get them from a binary release (sometimes called *-dist) lib directory of Apache Hama.

commons-configuration-1.6.jar ; commons-httpclient-3.0.1.jar ; commons-logging-1.0.4.jar; commons-lang-2.6.jar ; hadoop-1.0.0.jar ; hama-.jar ; zookeeper-3.3.2.jar.

*You can also add the configuration XML's to your classpath, by creating a new folder "conf" and add it as a source folder via rightclick->Build Path->Use as Source Folder

*Create a new class to test it and put the following code into it

```

public static void main(String[] args) throws IOException,
    InterruptedException, ClassNotFoundException {
    PiEstimator.main(args);
}

```

*Right-click on the source and choose Run As->Java Application

*Gratulation! You should see the following output

```
12/04/22 11:02:34 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using
builtin-java classes where applicable
12/04/22 11:02:34 WARN bsp.BSPJobClient: No job jar file set. User classes may not be found. See BSPJob#setJar
(String) or check Your jar file.
12/04/22 11:02:34 INFO bsp.BSPJobClient: Running job: job_localrunner_0001
12/04/22 11:02:37 INFO bsp.LocalBSPRunner: Setting up a new barrier for 20 tasks!
12/04/22 11:02:37 INFO bsp.BSPJobClient: Current supersteps number: 0
12/04/22 11:02:37 INFO bsp.BSPJobClient: The total number of supersteps: 0
12/04/22 11:02:37 INFO bsp.BSPJobClient: Counters: 7
12/04/22 11:02:37 INFO bsp.BSPJobClient: org.apache.hama.bsp.JobInProgress$JobCounter
12/04/22 11:02:37 INFO bsp.BSPJobClient: LAUNCHED_TASKS=20
12/04/22 11:02:37 INFO bsp.BSPJobClient: org.apache.hama.bsp.BSPPeerImpl$PeerCounter
12/04/22 11:02:37 INFO bsp.BSPJobClient: SUPERSTEPS=0
12/04/22 11:02:37 INFO bsp.BSPJobClient: SUPERSTEP_SUM=20
12/04/22 11:02:37 INFO bsp.BSPJobClient: TIME_IN_SYNC_MS=178
12/04/22 11:02:37 INFO bsp.BSPJobClient: TOTAL_MESSAGES_SENT=40
12/04/22 11:02:37 INFO bsp.BSPJobClient: TOTAL_MESSAGES_RECEIVED=20
12/04/22 11:02:37 INFO bsp.BSPJobClient: TASK_OUTPUT_RECORDS=1
Estimated value of PI is 3.14764
Job Finished in 3.248 seconds
```