

# AUTOSCALE v2.0 HOWTO

Roland Kotto Kombi

## Contents

<b>1</b>	<b>Requirements</b>	<b>1</b>
1.1	System . . . . .	1
1.2	Apache Storm . . . . .	1
<b>2</b>	<b>Installation</b>	<b>1</b>
2.1	Database management . . . . .	2
2.2	Configuration . . . . .	2
2.3	Compilation from sources . . . . .	2
<b>3</b>	<b>Run and monitor a topology</b>	<b>2</b>

## 1 Requirements

### 1.1 System

To install and test AUTOSCALE scheduler, the system must fulfill following requirements:

- Java 7 or later
- Python 2.7.0 or later
- Apache Storm 1.0.1 or later
- Alias STORM\_HOME pointing on Apache Storm's installation directory
- Maven 3.3.1 or later (optional, only while compiling from sources)

AUTOSCALE is compatible with Linux and Windows operating systems.

### 1.2 Apache Storm

AUTOSCALE version 2.0 is compatible only with Apache Storm 1.0.1 or later. Versions below 1.0.1 are not compatible anymore because of namespace refactoring.

## 2 Installation

Before installing and running AUTOSCALE, the monitoring database needs to be configured thanks to the sql script **autoscale.sql**. Users must create a database with credentials to perform SELECT and INSERT commands.

## 2.1 Database management

The script `autoscale.sql` must be imported in a MySQL database. Database name and the user credentials must be reported in AUTOSCALE configuration file.

## 2.2 Configuration

To custom AUTOSCALE parameters, a configuration (`conf/autoscale_parameters.xml`) file is provided. Users can modify connection parameters to the Nimbus (host+port), database information (host, name, user credentials) and monitoring parameters (window size, step...). This file must be placed in `./$STORM_HOME/conf/`.

The jar file (`autoscale.jar`) must be copied in `./$STORM_HOME/lib/`. In addition, the following line must be added in (`storm.yaml`):

```
storm.scheduler:  
"storm.autoscale.scheduler.AutoscaleSchedulerImpact"
```

It is also possible to monitor Storm default scheduler and the resource aware scheduler with following statements:

```
storm.scheduler:  
"storm.autoscale.scheduler.MonitoredEvenScheduler"
```

```
storm.scheduler:  
"storm.autoscale.scheduler.MonitoredResourceAwareScheduler"
```

## 2.3 Compilation from sources

Scheduler and topologies can be imported as Maven projects, and can be compiled we commands:

```
mvn clean  
mvn assembly:assembly
```

Versions `***-jar-with-dependencies.jar` must be considered for dependency management.

## 3 Run and monitor a topology

To run test topologies, the following command should executed from the directory including topologies' jar (for example `jarTopologies.jar`):

```
./$STORM_HOME/bin/storm jar jarTopologies.jar  
stormBench.stormBench.MyTopology
```

where MyTopology should be `LinearTopology`, `DiamondTopology`, `StarTopology` or `AdvertisingTopology`. A configuration file `topParameters.xml` allows to configure topology's parameters like resource requirements or state host (basically one of Zookeeper's host).