

# MySQL Cluster Quick Start Guide: Windows, LINUX & Solaris

This guide is intended to help the reader get a well configured MySQL Cluster database up and running on a single or across multiple hosts – whether they be running Windows, LINUX or Solaris. The guide will step through using the MySQL Auto-Installer – a browser-based GUI to perform this. Alternatively, MySQL Cluster Manager is available – a commercial CLI that in addition to creating the cluster can be used for its ongoing management – this [blog explains how to use MCM to configure and deploy MySQL Cluster](#)<sup>1</sup>.

## 1 Get the software

For Generally Available (GA), supported versions of the software, download from <http://www.mysql.com/downloads/cluster/>

Make sure that you select the correct platform (“Linux – Generic” for Linux). If you want to try out a pre-GA version then check <http://dev.mysql.com/downloads/cluster/> . For commercial versions of the MySQL Cluster download from <https://edelivery.oracle.com/>

From MySQL Cluster 7.3, the auto-installer is part of the MySQL Cluster tar ball/zip package. Locate the tar ball or zip file that you’ve downloaded and extract it:

```
$ tar xvf mysql-cluster-gpl-7.3.1-linux2.6-i686.tar.gz # Linux/Solaris. Drag & drop on Windows
```

Repeat this step on every host that will be part of the Cluster (and if different, the machine you’d like to run the installer from).

**Note:** Only use MySQL Server executables (`mysqlds`) that come with the MySQL Cluster installation.

## 2 Launch the installer

A single command launches the installer (for Linux or Solaris):

```
$ bin/ndb_setup # Linux or Solaris
```

or on Windows just double-click `setup.bat` from the top directory.

If you launch the installer from a desktop environment then the first page of the wizard will automatically be opened in your web browser, if not then just browse to the URL that is displayed on the command line.

## 3 Follow the wizard

After the initial splash-screen you’re presented with the page shown in Figure 1 where you specify the list of servers that will form part of the cluster.

You also get to specify what “type” of cluster you want; if you’re experimenting for the first time then it’s probably safest to stick with “Simple testing” but for a production system you’d want to specify the application type and query profile, i.e. write-intensive.

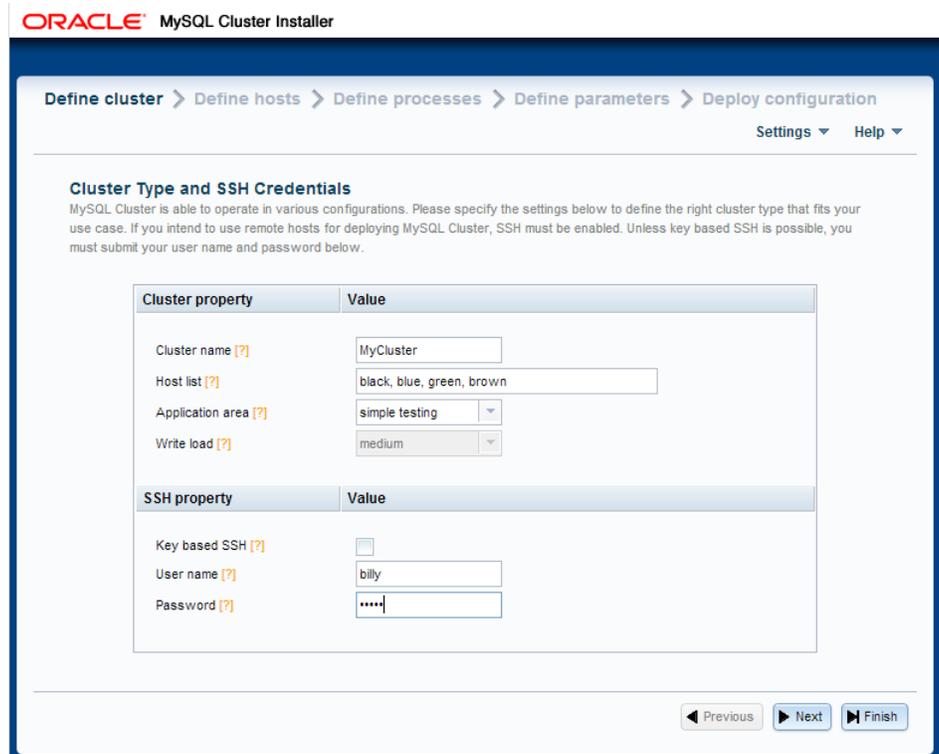


Figure 1 Specify hosts and application type

<sup>1</sup> <http://www.clusterdb.com/mysql-cluster/mysql-cluster-manager-1-1-2-creating-a-cluster-is-now-trivial/>

On the next page (Figure 2), you will see the wizard attempt to auto-detect the resources on your target machines. If this fails (e.g. if you don't have an accessible SSH server on the target hosts<sup>1</sup>) then you can enter the data manually. You can also overwrite the resource-values - for example, if you don't want the cluster to use up a big share of the memory on the target systems then just overwrite the amount of memory.

It's also on this page that you can specify where the MySQL Cluster software is stored on each of the hosts (if the defaults aren't correct) - this should be the path to where you extracted the MySQL Cluster tar-ball file - as well as where the data (and configuration files) should be stored. You can just overwrite the values or select multiple rows and hit the "edit" button.

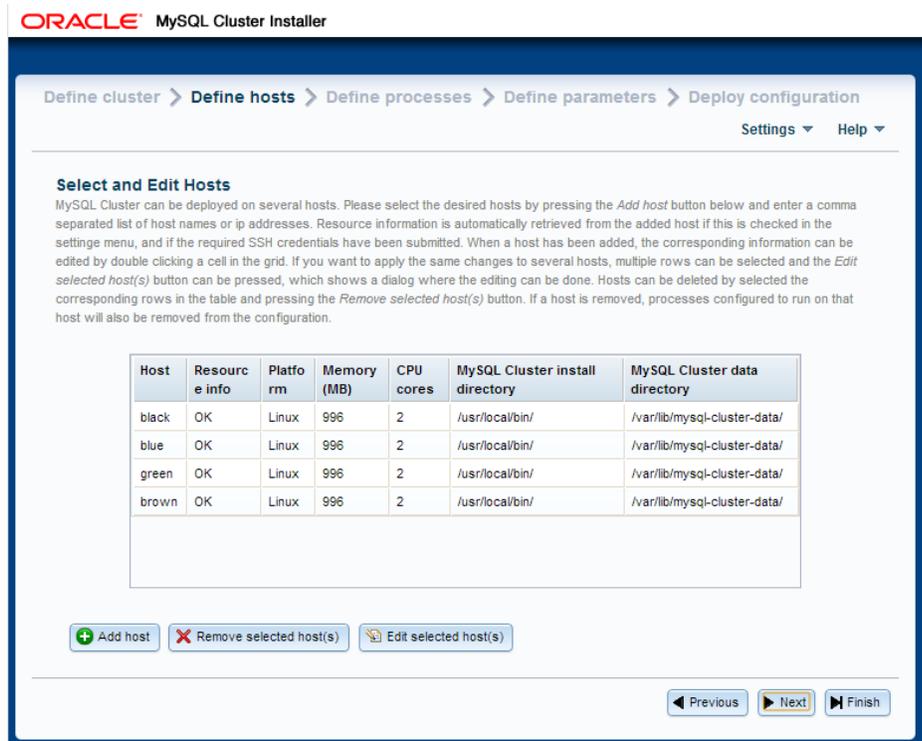


Figure 2 Auto-discovery of host resources

The following page (Figure 3) presents you with a default set of nodes (processes) and how they'll be distributed across all of the target hosts - if you're happy with the proposal then just advance to the next page.

Alternatively, you can add extra nodes, move nodes from one host to another (just drag and drop), delete nodes or change a node from one type to another.

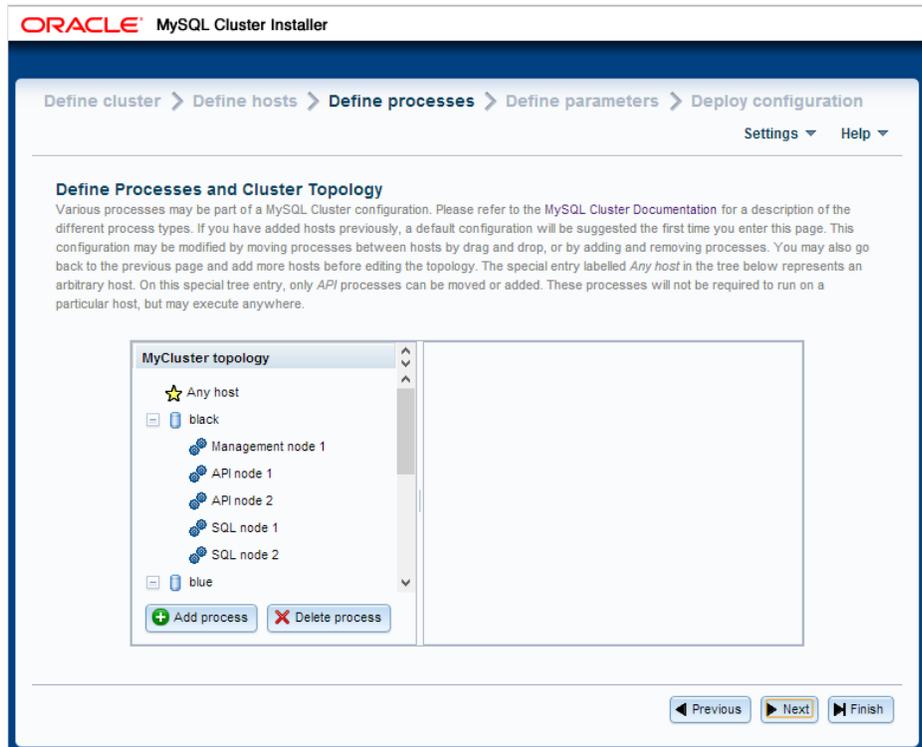


Figure 3 Review Cluster topology

<sup>1</sup> Windows typically doesn't include an SSH server by default and so for auto-discovery and deployment to work, one will need to be installed - for example, freeSSHd.

On the next screen (Figure 4) you're presented with some of the key configuration parameters that have been configured (behind the scenes, the installer sets many more) that you might want to override; if you're happy then just progress to the next screen. If you do want to make any changes then make them here before continuing. Note that you can enable the advanced configuration option here in order to view/modify more parameters.

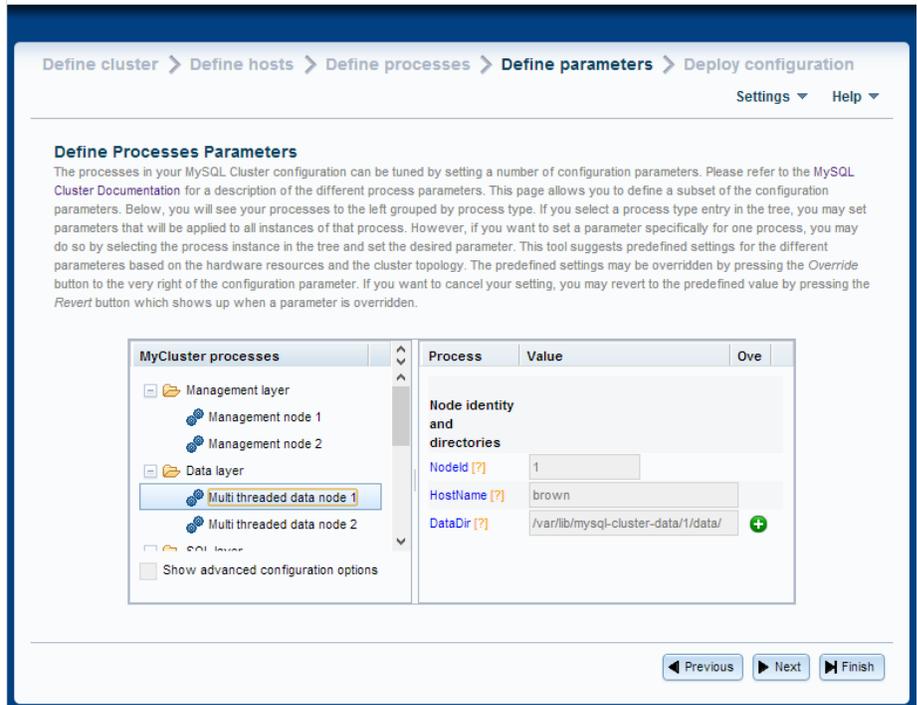


Figure 4 Review configuration parameters

With the click of a button, the final screen (Figure 5) lets you deploy (copy the correct configuration settings to the hosts and create the directory structures) and start the Cluster.

If you prefer or need to start the processes manually, this page also shows you the commands that you'd need to run (as well as the configuration files if you need to create them manually).

A traffic light display shows the various Cluster nodes (processes) coming into service.

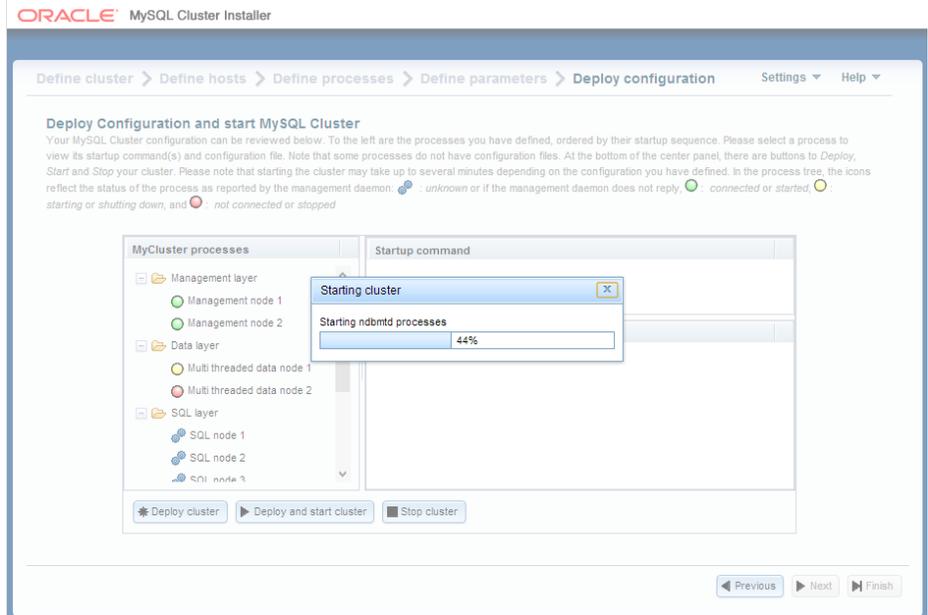


Figure 5 Deploy the Cluster

## 4 For further information

<http://www.clusterdb.com/mysql-cluster/mysql-cluster-7-3-auto-installer/> Demo video: using the MySQL Cluster Auto-Installer.

[MySQL Cluster Evaluation Guide \(http://www.mysql.com/why-mysql/white-papers/mysql\\_cluster\\_eval\\_guide.php\)](http://www.mysql.com/why-mysql/white-papers/mysql_cluster_eval_guide.php) In this whitepaper learn the fundamentals of how to design and select the proper components for a successful MySQL Cluster evaluation.

[MySQL Cluster Performance Optimization Guide \(http://www.mysql.com/why-mysql/white-papers/mysql\\_wp\\_cluster\\_performance.php\)](http://www.mysql.com/why-mysql/white-papers/mysql_wp_cluster_performance.php) In this guide, learn how to tune and optimize the MySQL Cluster database to handle diverse workload requirements.

MySQL Cluster Documentation (<http://dev.mysql.com/doc/index-cluster.html>).