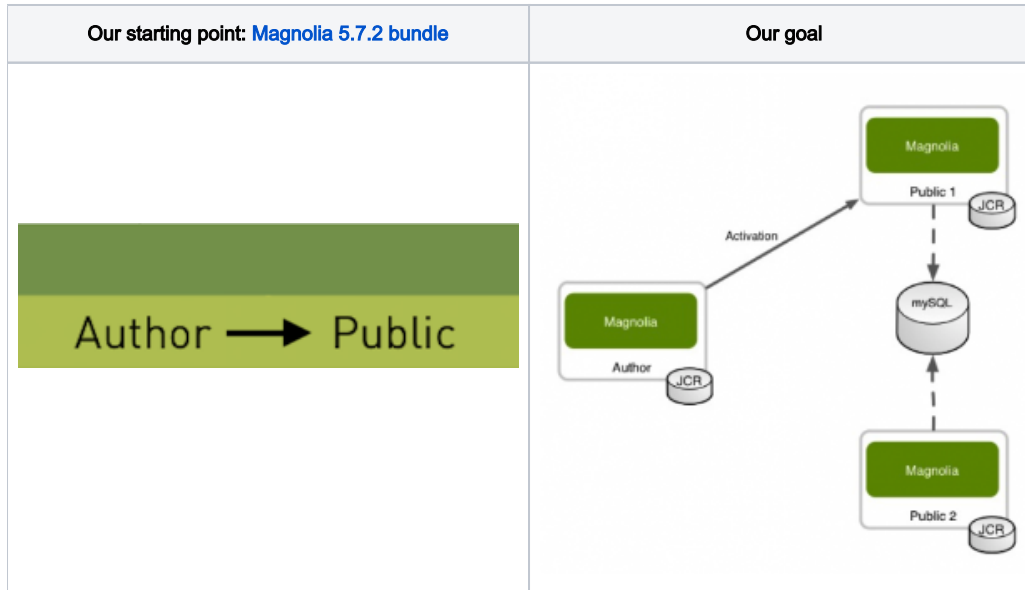


# Clustering two public instances

This is a summary of the entries made by [Philipp Bärffuss \(Setting up a Jackrabbit Clustering\)](#) and [Christian Ringele \(Clustering two publics instances\)](#). This is not intended to be an instructional guide. No further explanation of what *Clustering* is, how it works or pros and cons of this kind of architecture. If you wish to learn more, you can find some breathtaking-stunning entries and docs about clustering at the [annex](#). My goal with this guide is to help you to achieve a five-minute-Clustering set-up for your tests or just for you to get some background on this subject.



## - Guide -

Scenario: I suppose a Magnolia 5.7.2 Bundle placed at `/Users/carlos/bundles/5.7.2` and a MySQL database as central repository for the cluster. Please, notice that the elements marked in **red** are supposed to be replaced by your default local configuration. So here we go 🍌

1. Rename the `magnoliaPublic` instance to `magnoliaPublic1` and make a copy of it for using it later in our second public instance:

- `cd /Users/carlos/Bundles/5.7.2/apache-tomcat-9.0.10/webapps/magnoliaAuthor/WEB-INF/config`
- `mv magnoliaPublic/ magnoliaPublic1`
- `scp -r magnoliaPublic1 magnoliaPublic2`

2. Make a copy of the `magnoliaAuthor` instance and rename it `magnoliaPublic`. Now delete the default folder `magnoliaPublic`:

- `cd /Users/carlos/Bundles/5.7.2/apache-tomcat-9.0.10/webapps/`
- `scp -r magnoliaAuthor magnoliaPublic1`
- `rm -rf magnoliaPublic`


3. Create the MySQL database for the cluster:

```
mysql> create database magnolia_public;
```

Note: For the MySQL connections better use a root user or any other user that has grant \* privileges over the `magnolia_public` schema. For this example I will use `root` user identified by `rootPassword`. If you do not have MySQL downloaded you can get it from [here](#) for free (version 8.0.15) and install it whatever you want. Just remember to add it to your `.bash_profile` path environment var and reload your bash config with the `source ~/.bash_profile` command.

### **.bash\_profile**

```
#set MySQL
export MYSQL_HOME=/usr/local/mysql
export PATH=$PATH:$MYSQL_HOME/bin
```

 For each public instance within the cluster we must configure these elements:

- jackrabbit-bundle-mysql-search.xml
- magnolia.properties
- repositories.xml

4. Create the shared folder (for shared repository) wherever you want:

- mkdir \$HOME/shared\_data (my environment \$HOME var is **/Users/carlos/**)

5. Edit at first place the *jackrabbit-bundle-mysql-search.xml*:

- cd **/Users/carlos/Bundles/5.7.2/apache-tomcat-9.0.10/webapps/magnoliaPublic1/WEB-INF/config/repo-conf**
- nano jackrabbit-bundle-mysql-search.xml

6. Add this cluster snippet editing the elements highlighted in **red** to fit your custom configuration (do not close this file till you do step 7 and step 8):

Note: Please, as long as highlighting text is not allowed inside the { code } snippet, just remember editing all the custom elements → **mysql user, mysql password** and **/Users/carlos** paths.

#### jackrabbit-bundle-mysql-search.xml

```
<Cluster id="cid_pub1" syncDelay="2000">
  <Journal class="org.apache.jackrabbit.core.journal.DatabaseJournal">
    <param name="revision" value="/Users/carlos/shared_data/revision.log" />
    <param name="driver" value="com.mysql.jdbc.Driver" />
    <param name="url" value="jdbc:mysql://localhost:3306/magnolia_public" />
    <param name="user" value="root" />
    <param name="password" value="rootPassword" />
    <param name="schema" value="mysql" />
    <param name="schemaObjectPrefix" value="journal_" />
  </Journal>
</Cluster>
```

7. Search for the datasource tag and modify the elements highlighted in red to fit your custom configuration:

#### jackrabbit-bundle-mysql-search.xml

```
<DataSource name="magnolia">
  <param name="driver" value="com.mysql.jdbc.Driver" />
  <param name="url" value="jdbc:mysql://localhost:3306/magnolia_public" />
  <param name="user" value="root" />
  <param name="password" value="rootPassword" />
  <param name="databaseType" value="mysql" />
  <param name="validationQuery" value="select 1"/>
</DataSource>
```

8. Edit the FileSystem (repository) and DataStore (repository) properties:

#### jackrabbit-bundle-mysql-search.xml

```
<FileSystem class="org.apache.jackrabbit.core.fs.local.LocalFileSystem">
  <param name="path" value="/Users/carlos/shared_data/repository"/>
</FileSystem>

<DataStore class="org.apache.jackrabbit.core.data.FileDataStore">
  <param name="path" value="/Users/carlos/shared_data/repository/datastore"/>
  <param name="minRecordLength" value="1024"/>
</DataStore>
```

9. Edit *magnolia.properties* file:

- cd `/Users/carlos/Bundles/5.7.2/apache-tomcat-9.0.10/webapps/magnoliaPublic1/WEB-INF/config/default`
- nano `magnolia.properties`

10. Edit the properties:

- `magnolia.repositories.jackrabbit.cluster.master=true`

This property indicates which cluster is the master and will be set to true in the instance which will be the receiver of the `magnoliaAuthor`. Rest of public instances (`magnoliaPublic2`, `magnoliaPublic3...`) will set this property to false.

- `magnolia.repositories.cluster=${magnolia.home}/repositories_cluster`
- `magnolia.repositories.jackrabbit.cluster.config=/Users/carlos/Bundles/5.7.2/apache-tomcat-9.0.10/webapps/magnoliaPublic1/WEB-INF/config/repo-conf/jackrabbit-bundle-mysql-search.xml`
- `magnolia.clusterid=cid_publ`

The value of the `clusterId` property must match the value of the property `id` we defined previously at the `jackrabbit-bundle-mysql-search.xml` file (step 6)

#### jackrabbit-bundle-mysql-search.xml

```
<Cluster id="cid_publ"
  ...
  ...
</Cluster>
```

11. Finally, to sum up the configuration of the first cluster, we must edit the `repositories.xml` file:

- cd `/Users/carlos/Bundles/5.7.2/apache-tomcat-9.0.10/webapps/magnoliaPublic1/WEB-INF/config/default`
- nano `repositories.xml`

For this task, we will do the following modifications:

- Comment the block `<RepositoryMapping>` which repositoryName is `Magnolia` → **DO NOT** comment the block `<Repository name="main">`. Once you had that block commented add the following snippet:

#### repositories.xml

```
<!-- Magnolia cluster repository -->
<Repository name="magnoliacluster" provider="info.magnolia.jackrabbit.ProviderImpl" loadOnStartup="true">
  <param name="configFile" value="${magnolia.repositories.jackrabbit.cluster.config}" />
  <param name="repositoryHome" value="${magnolia.repositories.cluster}" />
  <param name="contextFactoryClass" value="org.apache.jackrabbit.core.jndi.provider.
DummyInitialContextFactory" />
  <param name="providerURL" value="localhost" />
  <param name="bindName" value="cluster-${magnolia.webapp}" />
  <workspace name="website" />
  <workspace name="config" />
  <workspace name="users" />
  <workspace name="userroles" />
  <workspace name="usergroups" />
</Repository>
<RepositoryMapping>
  <Map name="website" repositoryName="magnoliacluster" workspaceName="website" />
  <Map name="config" repositoryName="magnoliacluster" workspaceName="config" />
  <Map name="users" repositoryName="magnoliacluster" workspaceName="users" />
  <Map name="userroles" repositoryName="magnoliacluster" workspaceName="userroles" />
  <Map name="usergroups" repositoryName="magnoliacluster" workspaceName="usergroups" />
</RepositoryMapping>
```

12. Copy the full `magnoliaPublic1` configuration to `magnoliaPublic2` instance:

- cd `/Users/carlos/Bundles/5.7.2/apache-tomcat-9.0.10/webapps/`
- `scp -r magnoliaPublic1 magnoliaPublic2`

13. For configuring the second cluster instance, we will re-do the step 6 and the steps from 9 to 11. Please pay attention on the following configurations:



- At the step 6, change the clusterId to an unique value (e.g. cib\_pub2)
- Step 10, the property magnolia.repositories.jackrabbit.cluster.master must be false, property magnolia.clusterid must match the value of the property id we defined previously for this cluster instance and the property magnolia.repositories.jackrabbit.cluster.config must point to the second cluster instance path.

14. Now you are ready to start Magnolia and install Author and publics instances (Only the master public instance will go through the full installation process, the slaves instances will only install those elements which are not shared in the cluster). **WARNING:** do not attempt to make a simultaneous installation of the both public instances, this may cause a repository consistence error.

15. Configure the master cluster instance as receiver (that one with the `cluster.master=true`)

▼	📁 publishing-core	—
▼	📁 config	—
▶	⚡ publicationByPathVoters	—
▶	⚡ publishingLogStorage	—
▼	⚡ receivers	—
▼	⚡ magnoliaPublic8080	—
◆	enabled	true
◆	url	http://localhost:8080/magnoliaPublic1

Note: As you may know/suppose, MySQL, MMSQL, Postgres, Derby... drivers are not included by default in Magnolia. In order the cluster works, please download and install in all the cluster instances the .jar file in the path `/Users/carlos/Bundles/5.7.2/apache-tomcat-9.0.10/webapps/magnoliaPublicN/WEB-INF/lib/`

Now feel free to make some changes in any element that is within the workspaces that are in the cluster (e.g. website). If you change the starting image of the tours carrusel and publish it to public1, if you refresh public2 you will see the publication has spread to this instance too 🍌

## - Annex -

- Check [Architectural Blueprint 3: Magnolia with JCR clustering](#)
- The inspirational sources of this wiki entry: [Setting up a Jackrabbit Clustering](#) and [Clustering two publics instances](#)
- [Official JCR-Clustering](#) info

[Back to top](#)