

myCAT v1.2
for Ubuntu 18.04 Desktop
(14.04 & 16.04)
Installation Manual

CONTENT

A.	Context	2
B.	Installation Procedure	2
1.	Installing Java 8.....	2
2.	Installing Apache Tomcat 8	3
3.	Removing LibreOffice 3 and Installing OpenOffice 3.4.0	5
4.	Installing myCAT	6
5.	Deploying myCAT through the Tomcat	6
6.	Starting, Stopping and Re-starting myCAT	7
7.	Defining the Location of the Help File and of the Original Documents	8
8.	Changing the Document Collections	9
9.	Adding Languages.....	9

A. Context

myCAT is a Computer-Assisted Translation (CAT) tool which has two essential uses:

- It allows searching for terminology in previous documents and shows how terms were translated in their complete context; and
- It detects whether some parts of a text to be translated were already translated in a previous document, and displays that reference document along with its translation.

Thus myCAT is a tool which facilitates the re-use of previously-translated documents as a source of references for terminology and for whole phrases.

This report describes how myCAT should be installed on a dedicated server running **Ubuntu Desktop 14.04, 16.04 & 18.04** 64-bit version.

B. Installation Procedure

Before starting, you should create an “**olanto**” user with administrator rights, so that myCAT can operate under the **/home/olanto** path.

1. Installing Java 8

The initial step is to remove any possible version of another version of Java called openJDK, which comes by default with some versions of Ubuntu 12.04.

Open a terminal console (CTRL+ALT+T) and type :

```
sudo apt-get purge openJDK*
```

Then execute one by one the following commands:

```
sudo add-apt-repository ppa:webupd8team/java
```

(note: this installs a Java installer)

```
sudo apt-get update
```

(note: this updates the installation utility)

```
sudo apt-get install oracle-java8-installer
```

(type Y to authorize the installation process and hit OK accept the License Terms.)

When the process is completed, type the following command to check the installed java version:

```
java -version
```

The resulting message should look more or less like this:

```
java version "1.8.0_171"
```

```
Java(TM) SE Runtime Environment (build 1.8.0_171-b11)
```

```
Java HotSpot(TM) 64-Bit Server VM (build 25.171-b11, mixed mode)
```

Finally define the JAVA_HOME variable :

Edit the file named “**environment**” which is located in **/etc** (double-clicking the file will open it in the GEDIT text editor. This file may contain only one line when you first edit it.)

```
sudo gedit /etc/environment
```

and add the following line at the end of the file:

```
JAVA_HOME=/usr/lib/jvm/java-8-oracle/
```

then save the file and close it.

Note: To be able to modify the “environment” file with the GEDIT editor you must be logged in as **root** (not as an administrator, but as the real root). Alternatively you can edit the “environment” file in command line with the “sudo” command and a Terminal editor such as VI.

2. Installing Apache Tomcat 8

In the Terminal console type the following commands:

```
sudo apt-get install tomcat8  
(type Y to authorize the installation process)
```

```
sudo apt-get install tomcat8-admin
```

This default installation process uses the following HTTP/1.1 Connector Port: **8080**.

To start and stop the Tomcat use the following commands:

```
sudo /etc/init.d/tomcat8 start (or restart)  
sudo /etc/init.d/tomcat8 stop
```

After the Tomcat is installed a number of parameter modifications must be made. To be able to modify the relevant files with the GEDIT editor you must be logged in as **root** (not as an administrator, but as the real root). Alternatively you can edit those files in command line with the “sudo” command and a Terminal editor such as VI.

- **Replacing Roles and Users**

Find the following file and edit it: **/etc/tomcat8/tomcat-users.xml**

```
sudo gedit /etc/tomcat8/tomcat-users.xml
```

The content of this file is the following:

```
<?xml version='1.0' encoding='cp1252'?>  
<!--  
Licensed to the Apache Software Foundation (ASF) under one or more  
contributor license agreements. See the NOTICE file distributed with
```

this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

```
-->
<tomcat-users>
<!--
  NOTE: By default, no user is included in the "manager-gui" role required
  to operate the "/manager/html" web application. If you wish to use this
  app,
  you must define such a user - the username and password are arbitrary.
-->
<!--
  NOTE: The sample user and role entries below are wrapped in a comment
  and thus are ignored when reading this file. Do not forget to remove
  <!-- ... --> that surrounds them.
-->
<!--
  <role rolename="tomcat"/>
  <role rolename="role1"/>
  <user username="tomcat" password="tomcat" roles="tomcat"/>
  <user username="both" password="tomcat" roles="tomcat,role1"/>
  <user username="role1" password="tomcat" roles="role1"/>
-->
</tomcat-users>
```

Below this piece of code:

```
<tomcat-users>
```

Add the following two lines:

```
<role rolename="manager-gui"/>
<user name="admin" password="*****" roles="manager-gui" />
```

So now the piece of code should look like this:

```
<tomcat-users>
<role rolename="manager-gui"/>
<user name="admin" password="*****" roles="manager-gui" />
```

Save the file and exit.

- **Increasing the maximum size of the WAR files**

Find the following file and edit it: **`/usr/share/tomcat8-admin/manager/WEB-INF/web.xml`**

```
sudo gedit /usr/share/tomcat8-admin/manager/WEB-INF/web.xml
```

Look for the following piece of code:

```
<multipart-config>
  <!-- 50MB max -->
  <max-file-size>52428800</max-file-size>
  <max-request-size>52428800</max-request-size>
  <file-size-threshold>0</file-size-threshold>
</multipart-config>
```

Add "1" to the `<max-file-size>` and the `<max-request-size>` values so the new code looks like this:

```
<multipart-config>
  <!-- 50MB max -->
  <max-file-size>152428800</max-file-size>
  <max-request-size>152428800</max-request-size>
  <file-size-threshold>0</file-size-threshold>
</multipart-config>
```

This operation increases the maximum size allowed when deploying WAR files.

Save the file and exit.

3. Removing LibreOffice 3 and Installing OpenOffice 3.4.0

By default the distribution of Ubuntu Desktop 12.04 comes with a pre-installed version of LibreOffice. We have to uninstall it before installing OpenOffice, which will be used for document conversion.

a. Removing LibreOffice 3

In the Terminal console type this command:

```
sudo apt-get remove libreoffice*.*
```

(type Y to authorize the uninstallation process)

Normally the LibreOffice icons must be automatically removed from the icon bar at the left of the screen.

b. Installing OpenOffice 3.4.0

Download to your `/Home/Download` folder the file called:

Apache_OpenOffice_incubating_3.4.1_Linux_x86-64_install-deb_en-US.tar.gz

which can be found on Olanto's software distribution page.

Type the following command to uncompress it:

```
cd ~/Home/Downloads
sudo tar -x Apache_OpenOffice_incubating_3.4.0_Linux_x86-64_install-deb_en-US.tar.gz
```

Alternatively you can double-click the `.tar.gz` file and the Archive Manager will propose to unpack it.

This will result in extracting a folder called **en-US**. Move this folder to `/Home/Desktop`, then type the following commands to install it:

```
cd /home/olanto/Desktop/en-US/DEBS
sudo dpkg -i *.deb
cd /home/olanto/Desktop/en-US/DEBS/desktop-integration
sudo dpkg -i *.deb
```

By now the OpenOffice 3.4.0 application should be installed on your computer. You can check out by looking for the "OpenOffice" application in the Dash Home environment and execute it.

AT THIS STAGE IT IS USEFUL TO REBOOT YOUR COMPUTER SO AS TO COMPLETE THE TOOL INSTALLATIONS.

4. Installing myCAT

Download the file called **MYCAT.tar.gz** (which can be found on Olanto's software distribution page) and extract it to the **/home/olanto** path with the following command:

```
sudo tar -x MYCAT.tar.gz
```

Alternatively you can double-click the .tar.gz file and the Archive Manager will propose to unpack it. Whichever way you choose, the result should be that a **"MYCAT"** folder (in capital letters) should be extracted to **/home/olanto/**

5. Deploying myCAT through the Tomcat

After myCAT is unpacked it must be deployed through the Tomcat Web Application Manager:

- Start the Tomcat (if it is not already running) with the following command:

```
sudo /etc/init.d/tomcat8 start (or restart)
```

- When the Tomcat is started, open a browser and enter the following URL: <http://localhost:8080/manager/html> . Enter the login and password you defined earlier (see "Replacing Roles and Users" in Section 2 above) and hit ENTER. The following environment is displayed:

Path	Version	Display Name	Running	Sessions	Commands
/	None specified		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/host-manager	None specified	Tomcat Host Manager Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/manager	None specified	Tomcat Manager Application	true	1	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

Scroll down to the bottom of the page, where you will find a "WAR file to deploy" utility:

Deploy

Deploy directory or WAR file located on server

Context Path (required):

XML Configuration file URL:

WAR or Directory URL:

WAR file to deploy

Select WAR file to upload

Diagnostics

Check to see if a web application has caused a memory leak on stop, reload or undeploy

This diagnostic check will trigger a full garbage collection. Use it with extreme caution on production systems.

Server Information

Tomcat Version	JVM Version	JVM Vendor	OS Name	OS Version	OS Architecture	Hostname	IP Address
Apache Tomcat/7.0.26	1.7.0_11-b21	Oracle Corporation	Linux	3.2.0-29-generic	amd64	ubuntu	127.0.1.1

Click on the Browse button and choose the following file:

/home/olanto/MYCAT/dist/TranslationText.war

and click on the Deploy button.

After the TranslationText file is deployed repeat the same operation with the following file:

/home/olanto/MYCAT/dist/mySelfQD.war. As a result, both applications should be displayed in the Tomcat Application Manager:

The screenshot shows the Tomcat Web Application Manager interface. At the top, there is a message box with "Message: OK". Below that, there are navigation links: "List Applications", "HTML Manager Help", "Manager Help", and "Server Status". The main section is titled "Applications" and contains a table with the following data:

Path	Version	Display Name	Running	Sessions	Commands
/	None specified		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/TranslationText	None specified		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/host-manager	None specified	Tomcat Host Manager Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/manager	None specified	Tomcat Manager Application	true	1	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/mySelfQD	None specified		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

Below the table, there is a "Deploy" section with the same form as shown in the first image.

6. Starting, Stopping and Re-starting myCAT

After the Tomcat is launched, the myCAT application can be started or re-started by executing this command:

```
cd ~/home/olanto/MYCAT/shell
sudo bash StartMYCATAgents.sh
```

OR

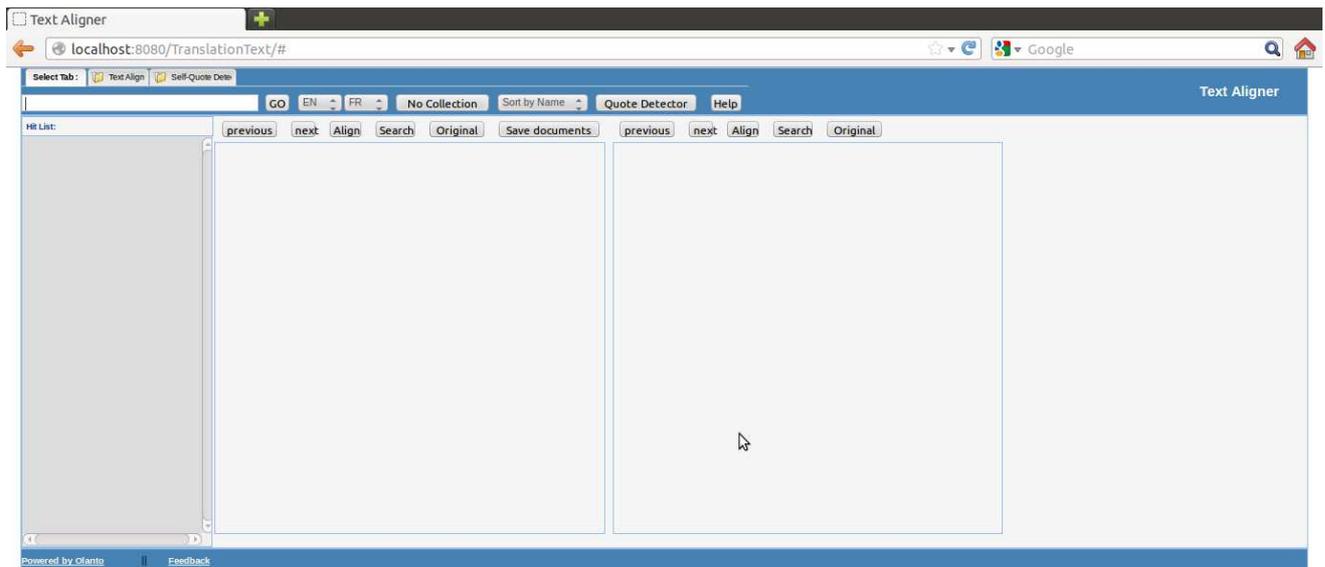
```
sudo /etc/init.d/tomcat8 restart
cd /home/olanto/MYCAT/shell
sudo bash StartMYCATAgents.sh
```

Alternatively you can also double-click this file: **home/olanto/MYCAT/shell/StartMYCATAgents.sh** and ask for it to be executed in a Terminal console. The application can be used when you see the message “**Server is ready**” at the bottom of Terminal console:

```
Terminal
load Master ObjectStore File for: /home/olanto/MYCAT/data/MAP_ARCH/map0/ObjectStore3.def id: ext
load Byte Arrays Vector: /home/olanto/MYCAT/data/MAP_ARCH/map0/ObjectStore3.def_SMALL
--- Byte Array Vector parameters, Current version: ByteArrayVector_InMemory 2.1
pathName: /home/olanto/MYCAT/data/MAP_ARCH/map0
fileName: ObjectStore3.def_SMALL
size: 131072
fixedArraySize: 32
maxUsedLength: 0
load Int Vector: /home/olanto/MYCAT/data/MAP_ARCH/map0/ObjectStore3.def_REALSIZE
load Long Vector: /home/olanto/MYCAT/data/MAP_ARCH/MAP
MAP ARCHIVE STATISTICS global:
Map for mapmax: 131072, Map now: 30, used: 0%
current Map manager is opened
Enregistrement du serveur
name:rmi://localhost/MAP
Server Map manager started
map server is ready ...
initialisation du convertisseur ...
Enregistrement du serveur
name:rmi://localhost/CONVSRV
Server is ready
```

Then you can launch a browser and type the following URL: <http://localhost/TranslationText/#>

The home page of myCAT looks like this:



Congratulations! myCAT is now operational.

Olanto's website contains some tips to perform a quick test on myCAT. For a more detailed use of the application, please refer to the Help file available from the Help button.

myCAT can be stopped at anytime by executing in command line or double-clicking this file:

/home/olanto/MYCAT/shell/StopMYCATAgents.sh

If you add new documents to /home/olanto/MYCAT/corpus/docs and you want them to be indexed immediately, you can execute this file: **/home/olanto/MYCAT/shell/UPDATE and RESTART.sh**

[7. Defining the Location of the Help File and of the Original Documents](#)

A Help file is available from myCAT's GUI (Help button). Additionally, it is possible for users to display reference documents in their original format. The default paths must be defined in the Tomcat by copying the following files from **/home/olanto/MYCAT/config**:

- **original.xml**
- **documentation.xml**

and pasting them into **/etc/tomcat8/Catalina/localhost/**

8. Changing the Document Collections

myCAT's corpus of reference documents is located here: **/home/olanto/MYCAT/corpus/docs**

You can remove it and replace it with your own documents. The supported file formats are MS-Word (doc and docx files), Excel (xls but not xlsx at this stage), PowerPoint (ppt but not pptx at this stage), PDF (indexable, not graphic format), HTML and TXT.

Please comply with the document classification and naming rules which are specified on Olanto's website. You can group your documents into folders, which will automatically become collections in the Collection box.

This distributed version of myCAT supports up to 65'000 documents and 250'000 different words. If you want to use more documents please contact the Olanto Foundation (info@olanto.org) to learn how to change the internal parameters of myCAT.

After you changed completely the content of the "docs" folder, you must execute the following file: **/home/olanto/MYCAT/shell/CLEAR, UPDATE and RESTART.sh**

This action will convert your documents into TXT files, will segment and index them, and will build the alignment maps. It will then automatically re-start the Tomcat and myCAT.

9. Adding Languages

The following languages are configured in this distributed version: **English, French, Spanish, Arabic, Russian, Chinese**. Alignment maps (built with the statistical machine translation tool called Moses) are only available for the following language pairs (in both directions) and all their combinations:

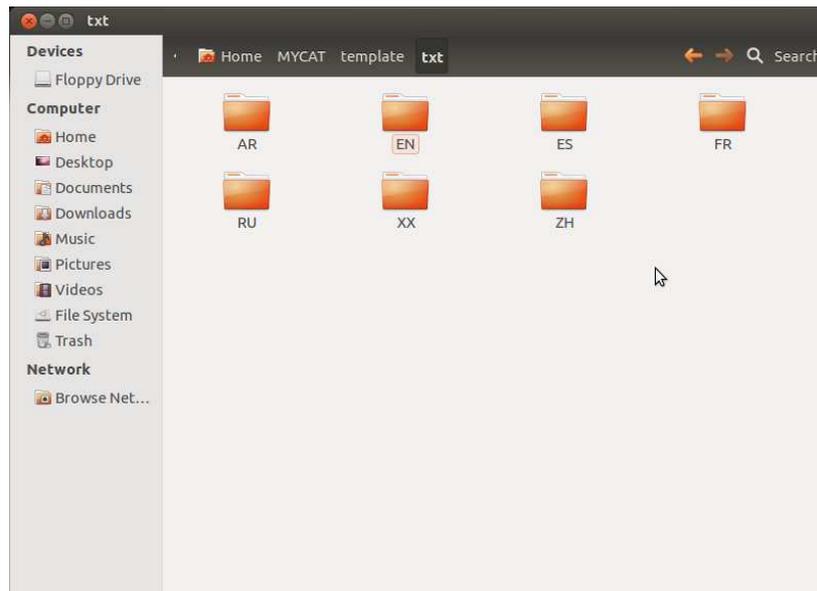
English – French	English – Arabic
English – Spanish	English – Russian
English – Portuguese	

It is possible to add or remove languages to the initial list, but please note that the automatic alignment for the new languages will be performed on the basis of a simple geometric algorithm, not with any alignment map.

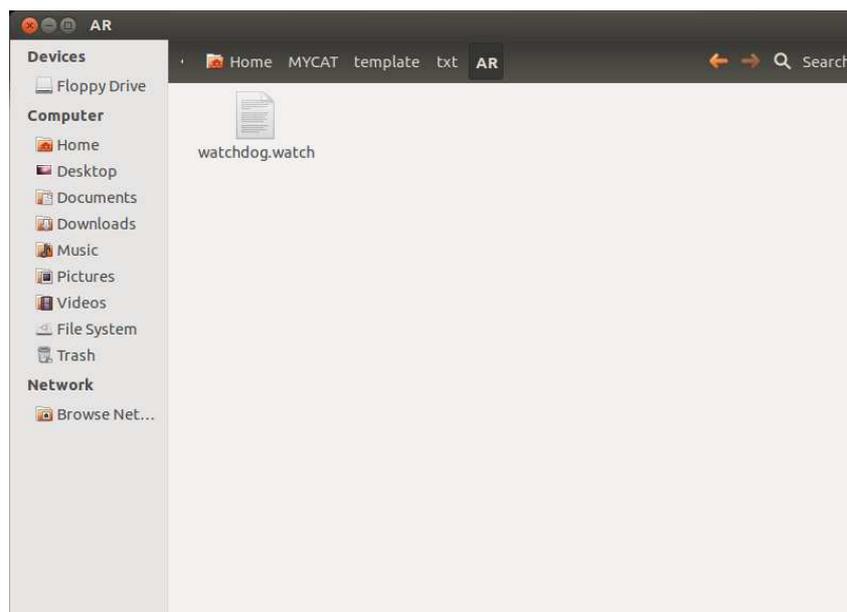
In order to add a new language:

- a. Stop myCAT (**/home/olanto/MYCAT/shell/StopMYCATAgents.bat**) and stop the Tomcat.
- b. In **/home/olanto/MYCAT/corpus/docs/[relevant collection]**, add a few document pairs which include the new language.
- c. In **/home/olanto/MYCAT/template/txt**, create a new folder and name it according to the required ISO 639-1 language code¹ on two letters. The default folder content is the following:

¹ http://www.loc.gov/standards/iso639-2/php/code_list.php



- d. In the newly-created folder, add an empty file so as to allow automatic copies of the folder (which are later performed by various batch files). You can copy such a file from an existing language folder; that file is called "watchdog.watch":



- e. In `/home/olanto/MYCAT/config/`, edit the file called **SEG_fix.xml** and find the following line : `<entry key="LIST_OF_SEG_LANG">EN AR FR ES RU ZH</entry>`
 Add the new 2-letter language code to that list and save the file.
- f. In `/home/olanto/MYCAT/config/`, edit the file called **IDX_fix.xml** and find the following line : `<entry key="ROOT_CORPUS_LANG">EN AR ES FR RU ZH</entry>`
 Add the new 2-letter language code to that list and save the file.
- g. In `/home/olanto/MYCAT/shell/`, execute **CLEAR, UPDATE and RESTART.sh**

The last step will convert, segment, index and map the new document pairs. When that process is completed you can open a new myCAT session in a browser. You should see the new language in the

scroll-down language menus. Choose the relevant language pair and search for a term which you know exists in the new document pair you just added.

Note: If that search does not retrieve the expected new document pair, take a look at the folder in `/home/olanto/MYCAT/corpus/bad/`. This folder contains documents whose conversion into TXT failed; maybe your new document is in there. If so the document was probably corrupt or password-protected; remove it from the "bad" folder and put a new, readable document in the `/home/olanto/MYCAT/corpus/docs/[relevant collection]` folder.

If you want to remove one or several languages, repeat phases a to f above but instead of adding a language folder you delete the relevant one(s). Then repeat phases g to k to re-segment, index and map the corpus.