

# [OLD] Install Cytomine v1.1 on Linux



This page describe **How to easily deploy the *latest* version of Cytomine (for installation procedure of V1.0 (January 2016), see [this page](#))**

Cytomine can be installed on personal computers (laptops, desktops) or on larger servers.

Cytomine uses many libraries and services but this page describes how to have a production instance of Cytomine ready to use with only four steps.

Once Cytomine's server is installed, one can access Cytomine through a modern web browser or through clients using RESTful API.

**Please, before further reading, pay attention to the [requirements](#). This installation documentation is for an Ubuntu 14.04LTS / 16.04LTS / 18.04 LTS.**

Please note we use Docker which is a kind of lightweight virtualization platform, so you do not need to create virtual machines to install Cytomine (it's better not do so for better performances).

**If you experiment any issue, please [contact us](#) by e-mail or describe your problem precisely [in our ticket system on Github](#).**

- 1 [Install Cytomine](#)
  - 1.1 [Step 1 - Retrieve Cytomine-bootstrap](#)
    - 1.1.1 [1.1 Install latest official release from Cytomine cooperative](#)
    - 1.1.2 [1.2 Install latest release with last developments from Cytomine ULiege](#)
  - 1.2 [Step 2 - Install Docker](#)
  - 1.3 [Step 3 - Configure Cytomine installation](#)
    - 1.3.1 [3.1 If you want to deploy on a local host :](#)
    - 1.3.2 [3.2 If you want to make your Cytomine instance accessible from anywhere :](#)
      - 1.3.2.1 [Detailed explanation of configuration options](#)
        - 1.3.2.1.1 [URLS](#)
        - 1.3.2.1.2 [Storage Paths](#)
        - 1.3.2.1.3 [Email](#)
        - 1.3.2.1.4 [IRIS](#)
        - 1.3.2.1.5 [Other advanced configuration](#)
      - 1.3.2.2
  - 1.4 [Step 4 - Deploy Cytomine](#)
  - 1.5 [Step 5 - Install test data \(optional\) - temporary unavailable](#)
  - 1.6 [Step 6 - Contact us and cite us \(optional\)](#)
- 2 [Troubleshooting](#)
- 3 [How to test your installation.](#)
- 4 [How to update your installation.](#)
- 5 [How to uninstall Cytomine.](#)

## Install Cytomine

### Step 1 - Retrieve Cytomine-bootstrap

First of all, we retrieve the Cytomine-bootstrap installation procedure.

#### 1.1 Install latest official release from Cytomine cooperative

If you want to install the last official release, copy/paste these commands:

```
mkdir Cytomine_src/  
cd Cytomine_src/  
wget https://github.com/cytomine/Cytomine-bootstrap/archive/master.zip -O bootstrap.zip  
unzip bootstrap.zip  
mv Cytomine-bootstrap-master Cytomine-bootstrap
```

#### 1.2 Install latest release with last developments from Cytomine ULiege

To install Cytomine with the last features from ULiege research team, copy/paste these commands:

```
sudo apt-get install git
mkdir Cytomine_src/
cd Cytomine_src/
git clone https://github.com/Cytomine-ULiege/Cytomine-bootstrap.git
cd Cytomine-bootstrap
```

## Step 2 - Install Docker

Then, we need to install the Docker engine.

```
sudo apt-get update
sudo apt-get install apt-transport-https ca-certificates curl software-properties-common
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
```

Then, you need to set up the stable repository of Docker but it depends on your kernel architecture. In case you do not know it, run the command `dpkg --print-architecture`

For x86\_64 or amd64 architecture use the following command. In other cases, please refer to the [official Docker documentation](#).

```
sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"
```

Finally, install the Docker package:

```
sudo apt-get update
sudo apt-get install docker-ce
```

(You can check Docker is installed correctly by running `sudo docker run hello-world`)

If needed, you will find a complete documentation on the [official site](#).

## Step 3 - Configure Cytomine installation

Now, it is required to edit the Cytomine file `configuration.sh` in the `Cytomine-bootstrap/` directory to prepare the installation of Cytomine Docker containers, using e.g.:

```
cd Cytomine-bootstrap/
emacs configuration.sh
```

You have two deployment options:

### 3.1 If you want to deploy on a local host :

First of all, the variables "XXX\_URL" will not be visible outside of your local server. So, due to the Docker architecture (isolation of the Docker's containers), we need to apply some changes to authorize network communication between our IMS and CORE containers (see below for further details). To do so, **add these URL values** (CORE\_URL, IMS\_URLS, UPLOAD\_URL, RETRIEVAL\_URL) **to your `/etc/hosts` file** with the following format : `"127.0.0.1 XXX_URL"`.

With the default URL your `etc/hosts` will contain the following lines

```
127.0.0.1      localhost-core
127.0.0.1      localhost-ims
127.0.0.1      localhost-ims2
127.0.0.1      localhost-upload
127.0.0.1      localhost-retrieval
127.0.0.1      rabbitmq
```

In the file `configuration.sh`, we will also need to set the variable `IS_LOCAL` at `true`.

NB : Please be careful that some problems might appear if you use localhost instead of CORE\_URL for a connection to Cytomine. So, keep using CORE\_URL.

### 3.2 If you want to make your Cytomine instance accessible from anywhere :

Contact your institutional system/network administrator before installing Cytomine so that they create DNS entries and make their HTTP port (80) accessible for CORE\_URL, IMS\_URLS and UPLOAD\_URL.

#### Detailed explanation of configuration options

The following tables explain how to configure your installation.

For a basic, simple installation, you need to configure only the mandatory variables.

#### URLS

Please use URLs that are not already present in your */etc/hosts* file to avoid conflicts.

Variable	Importance	Explanation	Example value
CORE_URL	<b>Mandatory</b>	the URL dedicated to the Core of Cytomine	localhost-core
IMS_URLS	<b>Mandatory</b>	the URL(s) dedicated to one or more image server Be careful that their value are space free even between two values	[localhost-ims]
UPLOAD_URL	<b>Mandatory</b>	the URL for image upload	localhost-upload
RETRIEVAL_URL	Advanced	the URL of retrieval server	localhost-retrieval
IIP_CYTO_URL	Advanced	the URL of Cytomine IIP server	localhost-iip-cyto
IIP_JP2_URL	Advanced	the URL of IIP server for JPEG2000	localhost-iip-jp2

#### Storage Paths

You have to choose existing paths on an available filesystem where image files will be stored. Please note you have to specify paths without a "/" at the end (e.g. "/data", not "/data/").

Variable	Importance	Explanation	Example value
IMS_STORAGE	<b>Mandatory</b>	A pre-existing filesystem path to store images (without ending /)	/data (but not /data/)
IMS_BUFFER_PATH	<b>Mandatory</b>	A pre-existing filesystem path to IMS buffer (without ending /)	/data/buffer (but not /data/buffer/)
ALGO_PATH	<b>Mandatory</b>	A pre-existing filesystem path to store Cytomine softwares (without ending /)	/data/algo
RETRIEVAL_PATH	<b>Mandatory</b>	A pre-existing filesystem path to store retrieval data (without ending /)	/data/thumb
BACKUP_PATH	If backup	A pre-existing filesystem path to store auto backup	

#### Email

Variable	Importance	Explanation	Example value
SENDER_EMAIL	Optional to test the app, mandatory otherwise	email params of the sending account	
SENDER_EMAIL_PASS	Optional to test the app, mandatory otherwise	email params of the sending account	
SENDER_EMAIL_SMTP_PORT	Optional to test the app, mandatory otherwise	email params of the sending account	
SENDER_EMAIL_SMTP_HOST	Optional to test the app, mandatory otherwise	email params of the sending account	
RECEIVER_EMAIL	If backup	email adress of the backup reports receiver	

#### IRIS

Variable	Importance	Explanation	Example value
IRIS_ENABLED	Advanced	Enable IRIS (see <a href="#">IRIS project</a> )	

IRIS_URL	Advanced		
IRIS_ID	Advanced	Usefull if we are admin of multiple IRIS version	
IRIS_ADMIN_NAME	Advanced		
IRIS_ADMIN_ORGANIZATION_NAME	Advanced		
IRIS_ADMIN_EMAIL	Advanced	The email adress where user will ask permissions for IRIS projects	

#### Other advanced configuration

Variable	Importance	Explanation	Example value
BACKUP_BOOL	Advanced	To enable an automatic daily backup, set to true	false
IS_LOCAL	Advanced	To deploy on distinct machines, set to false	true
RETRIEVAL_ENGINE	Advanced	memory, redis	redis
RETRIEVAL_PASSWD	Advanced		
RABBITMQ_LOGIN	Advanced		
RABBITMQ_PASSWORD	Advanced		
NB_IIP_PROCESS	Advanced	Number of simultaneous IIP processes	20
MEMCACHED_PASS	Advanced		
BIOFORMAT_ENABLED	Advanced	Enable the bioformat convertor to support more formats (VSI, OME-TIFF,...)	
BIOFORMAT_ALIAS	Advanced		
BIOFORMAT_PORT	Advanced		

## Step 4 - Deploy Cytomine

Then, you can generate your installation file and deploy Cytomine. Run following commands in the Bootstrap directory:

```
sudo bash init.sh
sudo bash start_deploy.sh
```

## Step 5 - Install test data (optional) - temporary unavailable

At the end of the installation we recommend to install test data (which takes roughly 30 additional minutes) in order to follow examples from our [user guide](#).

## Step 6 - Contact us and cite us (optional)

We kindly ask you to cite our ([Marée et al., \*Bionformatics\* 2016](#)) paper and website (<http://www.cytomine.be/>) when using Cytomine in your work.

Please also consider [contacting us](#) to let us know who you are and why you are using Cytomine.

## Troubleshooting

In case of problems, see page [Known problems of v1.0](#) (and specially the entry "Error : postgis\_data not found/not started").

Please note that in some environments problems arise with network configuration (firewall) preventing docker to retrieve required packages from Internet. If docker fails to install the base container with error messages related to [archive.ubuntu.com](http://archive.ubuntu.com), you should first clean docker containers (using the first script above then the following command: `sudo docker rmi $(sudo docker images -q)`); then you have to edit the DNS in the default docker configuration file and restart docker service as explained here: <http://stackoverflow.com/questions/24151129/docker-network-calls-fail-during-image-build-on-corporate-network>

Please note that server components might take one or two minutes to boot the first time (which will generate a "502 Bad Gateway message" or the browser to wait for a reponse).

---

## How to test your installation.

You can log into Cytomine through the [http://CORE\\_URL](http://CORE_URL) (by default: <http://localhost-core/>) in your web browser using the admin session (the password was asked during installation). If you did not install the "test data", you will have an empty instance. You have to create a project, users, ...

Please have look at our [Cytomine User Guide](#) for default usernames/passwords and see examples on toy data (You have to answer "yes" at the end of the installation procedure to inject these toy data).

To add new users, you can log using the admin account (password is asked during installation). You can then open an admin session (top right menu) to have access to User Configuration options.

In addition, you can make basic tests using the Cytomine Java client (see [how to install the java client](#)) or the [Cytomine Python client](#).

---

## How to debug your installation.

We provide a script (in the Cytomine-bootstrap directory) that outputs logs of the different containers and packs them in an reporting.tgz file for further analysis:

```
sudo sh ./reporting.sh
```

If you want to analyze more deeply an issue, depending on your problem (e.g. issue with image viewing), you can enter into the corresponding Docker container (e.g. ims) using the docker enter command and analyze log files.

First, install once the docker enter tool using (or use the docker exec command):

```
docker run -v /usr/local/bin:/target jpetazzo/nsenter
```

Then enter into the corresponding docker (e.g. image server) and analyze the log:

```
sudo docker exec -it ims /bin/bash/  
tail -n 100 -f var/lib/tomcat7/logs/catalina.out
```

Here is a list of log file paths in the different docker containers (sudo docker logs *container*):

Cytomine Component	Docker container	log file
Cytomine-Core	core	var/lib/tomcat7/logs/catalina.out
Image server	ims	var/lib/tomcat7/logs/catalina.out
IIPImage server	iipOff, iipJ2, iipCyto, iipVent	tmp/iip-openslide.out
Database server	db	var/log/postgresql/postgresql-9.3-main.log
MongoDB	mongodb	var/log/mongodb/mongo.log
Cytomine-DataMining	software_router	/software_router/algo/logs/

To debug from the Java client, create a `log4j.properties` file in the directory where you execute your script with the following content (INFO/WARN/ERROR/DEBUG provides different levels of messages):

```
#####
# Root logger option
log4j.rootLogger=DEBUG, stdout

# Direct log messages to stdout
log4j.appender.stdout=org.apache.log4j.ConsoleAppender
log4j.appender.stdout.Target=System.out
log4j.appender.stdout.layout=org.apache.log4j.PatternLayout
log4j.appender.stdout.layout.ConversionPattern=%d{yyyy-MM-dd HH:mm:ss} %-5p %c{1}:%L - %m%n
#####
```

## How to update your installation.

Docker is also used to easily update your Cytomine installation.

For example, if

- We update the current WAR's,
- A newer version of a component is available (e.g. the image server to support newer image formats),
- You customize some Docker images,

all you need to do is to restart the containers to have an up-to-date version of Cytomine with your own customization.



Of course, **restarting all the containers will make you lost all your data**. To avoid this problem, some containers are specifically designed for your data and **must always be running** (if you need to stop them, do a backup first) ! The *"data containers"* will have to be updated only if you made fundamental changes on the DBMS (e.g. update version, change postgres to mysql, ...)

The `restart_keep_data.sh` script (in Cytomine-bootstrap directory) is made to re-deploy an up-to-date app without erasing your data.

So, as much as possible(\*), **only run this script to redeploy your app**.

```
cd Cytomine-bootstrap
sudo sh restart_keep_data.sh
```



NB : If you add a custom container to your installation, you have to update this script.

## How to uninstall Cytomine.

Please [let us know](#) why Cytomine does not meet your expectations.

In order to uninstall it, you have to delete all Docker container images using the following commands.

**!!! WARNING !!!:** This will definitively delete all Cytomine data and all Docker containers.

**!!! WARNING !!!:** If you are using Docker for other projects, you have to delete only Cytomine containers (sudo docker images ; then for each Cytomine container: sudo docker rmi \$id)

```
sudo sh clean_docker.sh
sudo docker rmi $(sudo docker images -q)
```

You should also delete remaining files in your local directories `IMS_STORAGE_PATH`, `IMS_BUFFER_PATH`, and `BACKUP_PATH`.

