

CLEMSite

This is a preliminary version of the software for CLEMSite software, granted under an MIT License. If you decide to use this software, read carefully the document and the linked ones.

CLEMSite is composed of two parts:

- One server engine, called CLEMSiteServer, developed with C#. It is composed by three main elements: a user interface to run and stop the server service, an asynchronous server which accepts connections from the client interface and receives commands, and a logic layer with the Atlas API component, which develops all the functionality associated with the setup and running performed by the Atlas Engine. It is Windows (XP/7/8.1/10) dependent.
- One client engine, called CLEMSite, divided into 2 elements CLEMSite-LM and CLEMSite-EM (containing *Navigator*, *Multisite* and *Run Checker*), developed with python 3.7 and Qt5 in the current version.

DISCLAIMER

The execution of the software was done under the framework of a proof of concept experiments presented in the article <https://www.biorxiv.org/content/10.1101/2021.03.19.436113v1> (end 2019) with an electron microscope Zeiss XB 540. The microscope was equipped with Zeiss SMARTSEM, Zeiss/Fibics server version, ATLAS5 and ATLAS engine, and it was used with specific versions of the software documented in the paper methods. Those versions have been updated since then. In addition, the *Fibics API* requires a special license granted only under Zeiss/Fibics permission.

For this reason, the authors, at the moment of writing this document, are not aware of any Zeiss/Fibics software internal changes in newer versions of the software that could cause stop, alteration or unexpected behaviour of the functionality here stated, and hence causing potential microscope malfunctioning or the current software to work as described. If after reading this you decide to proceed with the installation of the prototype software and, given that you have all the versions and licenses available, start on [Reserved instructions](#).

The software is distributed under the [MIT License](#).

Glossary

Important dependencies

The program depends on the following third party libraries and software:

- SEM Server from Carl Zeiss
- FibicsVE API from Fibics/Zeiss
- Users must have access to the microscope via the SmartSEM user interface.

- Atlas 5 connected to SmartSEM with Atlas 5 including Atlas 3D and with the Atlas API registered to the operative system. Atlas 5 (NPVE plus Atlas 3D without the Atlas 5 viewer) will be referred to as Atlas Engine.

For CLEMSite :

- CLEMSite Server was developed using C# with VisualStudio 2015 from Microsoft. Only one additional library was used for Json from Newtonsoft.

Additionally, we have the following dependencies in terms of software for clients:

- Python 3.7 - <https://docs.python.org/3/license.html>

Associated dependencies :

- *numpy, pandas, scipy, scikit-learn, scikit-image, backports, seaborn, jupyter, opencv2, keras*
- *pyQt5*
- *Tensorflow*

1.1 Python Anaconda installation

1) Download Anaconda with version python 3.7. www.anaconda.com.

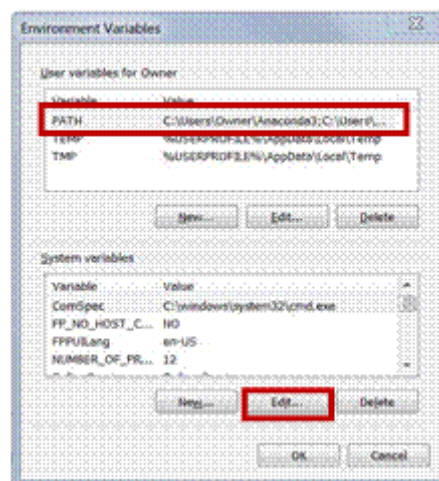
2) In the installing options, make sure that you add Anaconda to your path.

If you have another python installation, it is possible to create a particular environment for the application. Read more instructions about it here:

<https://conda.io/projects/conda/en/latest/user-guide/tasks/manage-environments.html>.

It is still needed to add python with the corresponding libraries to your path.

For Windows 10/8/7:



- Open System Properties (Right click Computer in the start menu, or use the keyboard shortcut Win+Pause)
- Click Advanced system settings in the sidebar.
- Click Environment Variables...
- Select PATH in the System variables section
- Click Edit
- Add Python's path to the end of the list (the paths are separated by semicolons). As an example:

```
path:\to\Anaconda3\;path:\to\Anaconda3\Scripts;path:\to\Anaconda3\Library\mingw-  
w64\bin;path:\to\Anaconda3\Library\usr\bin;path:\to\Anaconda3\Library\bin)
```

Additionally

- Set Anaconda path in *MSite.exe* (server engine) » Options » Set python path... Then indicate your python environment

Reboot your computer and then start anaconda prompt. Type *activate name_of_environment*. Now, move to the directory where you unzipped the CLEMSite application (using *cd* command in the console). Then type:

```
(your_environment) D:\Documents\CLEMSite\dist> pip install -r clemsite_env.txt
```

1.2 Sample setup

In a FIB-SEM, the sample is loaded onto the stage and positioned below the SEM beam. In this case, the sample is a flat, resin disc containing a monolayer of embedded cultured cells that is mounted onto an SEM stub and gold sputter coated. As in any FIB-SEM, there are several steps to align and position the beams and sample before a full acquisition can occur. The sample is moved to the eucentric height (at 1.5 kV SEM) at a safe distance to the objective lens and rotated in a way that the alphanumeric letters and the grid lines are at a 45° angle ensuring best illumination for the pattern detection.

Perform a coincidence point close to your 1st region of interest to ensure imaging and milling at the same position. You can perform a SEM adjustment, by checking the wobbler to ensure a nicely centered SEM beam through the SEM aperture. Stigmators and focus must be set to obtain the best possible crisp image. Before starting the automation in the Atlas Engine the beam shift for the SEM and the FIB beam are set to 0. To compensate for the tilted stage and the angle between the SEM beam and the sample when imaging, the tilt compensation must be set to 54°.

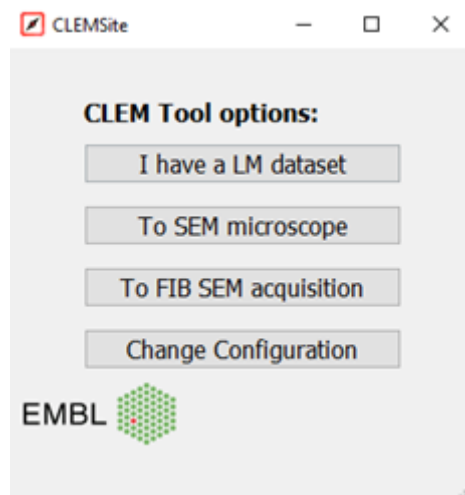
1.3 Running the software

The software is divided in modules. Each module has its own set of scripts and GUI. Modules have the advantage that each can be tested and updated independently.

- **CLEMSite-LM** contains scripts to extract landmarks on images coming from LM and builds a LM map.
- **CLEMSite-Server** connects to ATLAS 5 (via Fibics API) and allows client processes to send commands to the microscope.
 - **Navigator** extracts landmarks from SEM images (sample surface) and builds a SEM landmark-based map, which correlates to a LM landmark based map.
 - **Multisite** executes a FIB-SEM run. For each target cell defined in the SEM map and an automatic acquisition is performed.
 - **Run Checker** keeps a continuous feedback with the microscope to modify parameters – e.g. for tracking or for error and warning messaging.

1) Move to the folder where CLEMSite is located. Open anaconda prompt, activate your environment if needed, and type:

```
python.exe executeMsite2.py
```



2) CLEMSite will show up. During the first installation, click on **Change Configuration**. The different options of the CLEMSite tool are shown here.

```
{
  "preferences": {
    "terrain_detector": [{
      "architecture":
".\\common\\grids\\MatTek\\terrain_detector\\terrain_model.json",
      "weights":
".\\common\\grids\\MatTek\\terrain_detector\\terrain_model.weights.hdf5"
    }],
    "cross_detector": [{
      "architecture":
".\\common\\grids\\MatTek\\cross_detector\\cross_model.json",
      "weights":
".\\common\\grids\\MatTek\\cross_detector\\cross_model.weights.hdf5"
    }],
    "pattern_detector": [{
      "architecture":
".\\common\\grids\\MatTek\\pattern_detector\\patterns_model.json",
      "weights":
".\\common\\grids\\MatTek\\pattern_detector\\patterns_model.weights.hdf5"
    }],
    "last_port": "8098",
    "last_ip_address": "127.0.0.1",
    "last_ip_address_atlas": "127.0.0.1",
    "scale": "micrometer",
    "tolerance_to_closest_neighbor": "50.0",
    "acquisition_parameters": [{
      "black_threshold_active": 1,
      "black_threshold": "90",
      "tracking_stop_point_active": 1,
      "tracking_ASIFT_active": 0,
      "tracking_xcorr": 0,
      "section_thickness": "200",
      "AFAS_period": "1800000"
    }],
  },
}
```

```
"line_detection": [{
  "SEM": [{
    "canny_thresh": "0.05",
    "ignore_calibration": "0",
    "stroke": "16",
    "k": "12",
    "clahe": "1",
    "laplacian": "0",
    "gaussian": "1.2"
  }],
  "LM": [{
    "canny_thresh": "0.06",
    "ignore_calibration": "0",
    "stroke": "20",
    "k": "12",
    "clahe": "1",
    "laplacian": "1",
    "gaussian": "1.2"
  }]
}],
"server_images": [{
  "port_frames": "1382",
  "address_frames": "127.0.0.1",
  "python_folder": "C:\\Users\\Owner\\Anaconda3\\python.exe",
  "use_local": "True",
  "automaton_folder": "default",
  "dir_frames_input": "G:\\Acquisitions\\29Nov19",
  "dir_frames_output": "G:\\Acquisitions\\29Nov19"
}],
"grid": [{
  "template_file": ".\\common\\grids\\MatTek\\template.png",
  "div1": "40",
  "div2": "560",
  "rotation": "0",
  "blocksize": "50",
  "spacing": "10",
  "rows": "26",
  "type": "MatTek",
  "flip": 1,
  "columns": "21",
```

```

        "patterns_list":"'0A', '0B', '0C', '0D', '0E', '0F', '0G', '0H',
'0I', '0J', '0K', '0L', '0M', '0N', '0O', '0P', '0Q', '0R', '0S', '0T', '0U',
'0V', '0W', '0X', '0Y', '0Z', '1A', '1B', '1C', '1D', '1E', '1F', '1G', '1H',
'1I', '1J', '1K', '1L', '1M', '1N', '1O', '1P', '1Q', '1R', '1S', '1T', '1U',
'1V', '1W', '1X', '1Y', '1Z', '2A', '2B', '2C', '2D', '2E', '2F', '2G', '2H',
'2I', '2J', '2K', '2L', '2M', '2N', '2O', '2P', '2Q', '2R', '2S', '2T', '2U',
'2V', '2W', '2X', '2Y', '2Z', '3A', '3B', '3C', '3D', '3E', '3F', '3G', '3H',
'3I', '3J', '3K', '3L', '3M', '3N', '3O', '3P', '3Q', '3R', '3S', '3T', '3U',
'3V', '3W', '3X', '3Y', '3Z', '4A', '4B', '4C', '4D', '4E', '4F', '4G', '4H',
'4I', '4J', '4K', '4L', '4M', '4N', '4O', '4P', '4Q', '4R', '4S', '4T', '4U',
'4V', '4W', '4X', '4Y', '4Z', '5A', '5B', '5C', '5D', '5E', '5F', '5G', '5H',
'5I', '5J', '5K', '5L', '5M', '5N', '5O', '5P', '5Q', '5R', '5S', '5T', '5U',
'5V', '5W', '5X', '5Y', '5Z', '6A', '6B', '6C', '6D', '6E', '6F', '6G', '6H',
'6I', '6J', '6K', '6L', '6M', '6N', '6O', '6P', '6Q', '6R', '6S', '6T', '6U',
'6V', '6W', '6X', '6Y', '6Z', '7A', '7B', '7C', '7D', '7E', '7F', '7G', '7H',
'7I', '7J', '7K', '7L', '7M', '7N', '7O', '7P', '7Q', '7R', '7S', '7T', '7U',
'7V', '7W', '7X', '7Y', '7Z', '8A', '8B', '8C', '8D', '8E', '8F', '8G', '8H',
'8I', '8J', '8K', '8L', '8M', '8N', '8O', '8P', '8Q', '8R', '8S', '8T', '8U',
'8V', '8W', '8X', '8Y', '8Z', '9A', '9B', '9C', '9D', '9E', '9F', '9G', '9H',
'9I', '9J', '9K', '9L', '9M', '9N', '9O', '9P', '9Q', '9R', '9S', '9T', '9U',
'9V', '9W', '9X', '9Y', '9Z', 'AA', 'AB', 'AC', 'AD', 'AE', 'AF', 'AG', 'AH',
'AI', 'AJ', 'AK', 'AL', 'AM', 'AN', 'AO', 'AP', 'AQ', 'AR', 'AS', 'AT', 'AU',
'AV', 'AW', 'AX', 'AY', 'AZ', 'BA', 'BB', 'BC', 'BD', 'BE', 'BF', 'BG', 'BH',
'BI', 'BJ', 'BK', 'BL', 'BM', 'BN', 'BO', 'BP', 'BQ', 'BR', 'BS', 'BT', 'BU',
'BV', 'BW', 'BX', 'BY', 'BZ', 'CA', 'CB', 'CC', 'CD', 'CE', 'CF', 'CG', 'CH',
'CI', 'CJ', 'CK', 'CL', 'CM', 'CN', 'CO', 'CP', 'CQ', 'CR', 'CS', 'CT', 'CU',
'CV', 'CW', 'CX', 'CY', 'CZ', 'DA', 'DB', 'DC', 'DD', 'DE', 'DF', 'DG', 'DH',
'DI', 'DJ', 'DK', 'DL', 'DM', 'DN', 'DO', 'DP', 'DQ', 'DR', 'DS', 'DT', 'DU',
'DV', 'DW', 'DX', 'DY', 'DZ', 'EA', 'EB', 'EC', 'ED', 'EE', 'EF', 'EG', 'EH',
'EI', 'EJ', 'EK', 'EL', 'EM', 'EN', 'EO', 'EP', 'EQ', 'ER', 'ES', 'ET', 'EU',
'EV', 'EW', 'EX', 'EY', 'EZ', 'FA', 'FB', 'FC', 'FD', 'FE', 'FF', 'FG', 'FH',
'FI', 'FJ', 'FK', 'FL', 'FM', 'FN', 'FO', 'FP', 'FQ', 'FR', 'FS', 'FT', 'FU',
'FV', 'FW', 'FX', 'FY', 'FZ', 'GA', 'GB', 'GC', 'GD', 'GE', 'GF', 'GG', 'GH',
'GI', 'GJ', 'GK', 'GL', 'GM', 'GN', 'GO', 'GP', 'GQ', 'GR', 'GS', 'GT', 'GU',
'GV', 'GW', 'GX', 'GY', 'GZ', 'HA', 'HB', 'HC', 'HD', 'HE', 'HF', 'HG', 'HH',
'HI', 'HJ', 'HK', 'HL', 'HM', 'HN', 'HO', 'HP', 'HQ', 'HR', 'HS', 'HT', 'HU',
'HV', 'HW', 'HX', 'HY', 'HZ', 'IA', 'IB', 'IC', 'ID', 'IE', 'IF', 'IG', 'IH',
'II', 'IJ', 'IK', 'IL', 'IM', 'IN', 'IO', 'IP', 'IQ', 'IR', 'IS', 'IT', 'IU',
'IV', 'IW', 'IX', 'IY', 'IZ', 'JA', 'JB', 'JC', 'JD', 'JE', 'JF', 'JG', 'JH',
'JI', 'JJ', 'JK', 'JL', 'JM', 'JN', 'JO', 'JP', 'JQ', 'JR', 'JS', 'JT', 'JU',
'JV', 'JW', 'JX', 'JY', 'JZ', 'KA', 'KB', 'KC', 'KD', 'KE', 'KF', 'KG', 'KH',
'KI', 'KJ', 'KK', 'KL', 'KM', 'KN', 'KO', 'KP', 'KQ', 'KR', 'KS', 'KT', 'KU',
'KV', 'KW', 'KX', 'KY', 'KZ'"
    },
    "email_preferences":
    [{
        "user_address": "clemsitefibsem@gmail.com",
        "recipients": "clemsitefibsem@gmail.com",
        "smtp": "smtp.gmail.com",
        "password": "my_password",
        "username": "clemsitefibsem",
        "port": "587"
    }]
}

```

Update the following fields, for the first time ONLY:

"python_folder": Your current version of python (as explained in the installation step, the address of your python environment). "C:\ProgramData\Anaconda3\python.exe"

Every time:

"server_images": Give a directory where the data is going to be stored for the current project

- **"dir_frames_input"**: "D:\GOLGI\14Nov_NOSPOTS",
- **"dir_frames_output"**: "D:\GOLGI\14Nov_NOSPOTS" It can be the same directory as input.
- **"email_address"**: Provide your email data for getting warnings during the *Multisite* acquisition. You will be exposing the password unencrypted, so for security reasons create an account only for this, do not use a personal account.

NOTE: This is a temporary format for options and follows the JSON convention (<https://www.json.org/>). Be careful with the editing the format: do not provide uncommon characters for names except dashes (_ or -), folders and files have to be given in double bars e.g. ".\name_folder\filename.tif". If a string is quoted, always use double quotes and try to not include spaces or delete commas from the original document.

Save the changes in the text file and close everything. Open CLEMSite again.

2. CLEMSite-LM

LM needs to be adapted to specific fluorescent microscopes and depends on many factors: availability of reflected light, type of feedback microscopy performed, etc...

For the experiment presented in the paper, we developed [CLEMSite-LM](#)

You are free to adapt it to your needs. At the end you will need the two output xml files for landmarks and targets.

3. [Before starting](#)

If you have not yet, please, **read the [DISCLAIMER](#) again.**

Here are some advices that we recommend to follow:

1: In the FIB-SEM (e.g. Zeiss Crossbeam 540):

- both column valves are open,
- EHT is turned on and gun filament heating is complete and stable
- FIB apertures are aligned (always check the alignments before a long term operation).
- Any maintenance issues related to the proper conditions of the FIB-SEM machine must be addressed.
- For the ATLAS 5 recipes : all tables and necessary adjustments in Atlas 5 recipes for Atlas 3D have been properly set up.

2: Users should have a basic knowledge on operating the SmartSEM User interface and Atlas 5 in Zeiss FIB-SEM machines.

3: Target localization in LM can be done independently, so it is advised to be prepared before starting the session in the FIB/SEM microscope.

4: The electron microscopy sample has been processed and designed to be compatible with the application:

- the sample is as flat as possible to avoid and undesired bumping into the objective lens.
- the coordinate system on the sample surface has been preserved and it can be observed in SEM at 54 degrees stage tilt and 45 degrees orientation;
- the resin is suitably hard to perform an uniform FIB milling;
- for biological samples, the chemical processing protocol performed allows significant contrast to detect the structures of interest using the ESB detector.

5: It is recommended to manually operate the microscope to set up a target acquisition manually (like one would do normally), and perform one acquisition fully manually. When satisfied, save the recipe file that worked and check that, before starting *CLEMSite*, the conditions of the microscope are the same as when you started the manual acquisition and all worked to your satisfaction.

6: *CLEMSite* has not been optimized for sections thinner than 20 nm. Thinner sections can make the acquisition fail during the stabilization period. The reason is that usually this process is manually driven by the operator, which observes the acquired images until he/she starts to observe changes in the image cross-section face that is going to be milled. It is quite common that the user uses jumps (advancing or retracting the milling beam) so no time is wasted milling in the “void” before hitting sample, which could generate undesired walls. For automation purposes, it would be possible to use 50-100 nm sections until the sample is hit, then change the section regime to smaller regime, like 10, 8 or even 5 nm. However, we did not implement this automatically, so it is not possible at the moment, and not advisable (you could try it, but there is no guarantee of correct operation).

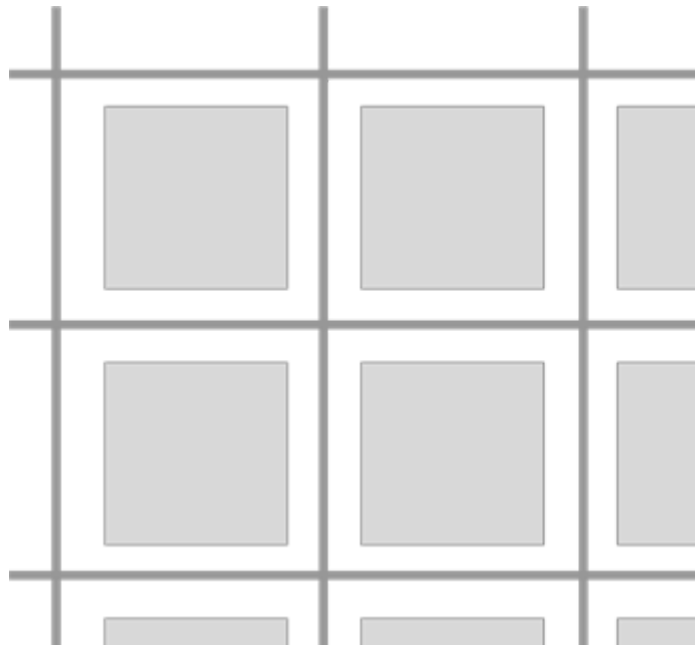
7: In a common FIB-SEM workflow, a deposition is required. A deposition is a layer of platinum on the sample surface covering the region to provide stability for ion beam milling and to protect from continuous radiation exposure when imaging. In a first approach, we instead use a very thick gold sputtered coat of the total sample surface. We observed no compromise to stability under these conditions when acquiring non-isotropic datasets with sections equal or bigger than 20 nm.

Deposition generates a new set of problems in automation developments which led us to the decision to omit it from the workflow at this stage. The two main reasons are the addition of time, as it would add at least one hour per cell in the workflow (using a 35 by 35 μm region) and the complications to find the correct conditions.

The sample has a gold or platinum coating (2-3 minutes at 30 mA current, Quorum (Q150RS); thickness estimated between 25-75 nm). Unproper coating could generate charges and distort the imaging beam or walls in the specimen.

Format expected of images for correct software analysis

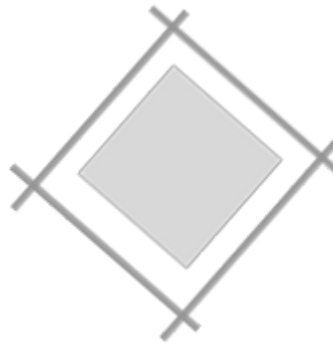
- Reflected light : 4 squares on image. E.g.:



- For SEM, EM crossings, the FOV has to get the full crossing:

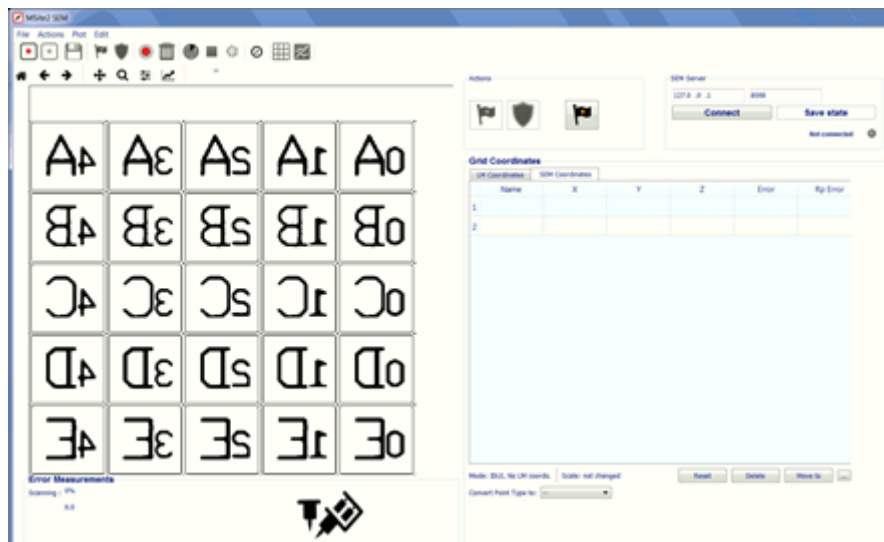


- In the first position, full square image :



4. CLEMSite EM

Go to SEM Microscope and the Navigator will pop up. Here is the usual workflow for sample exploration.



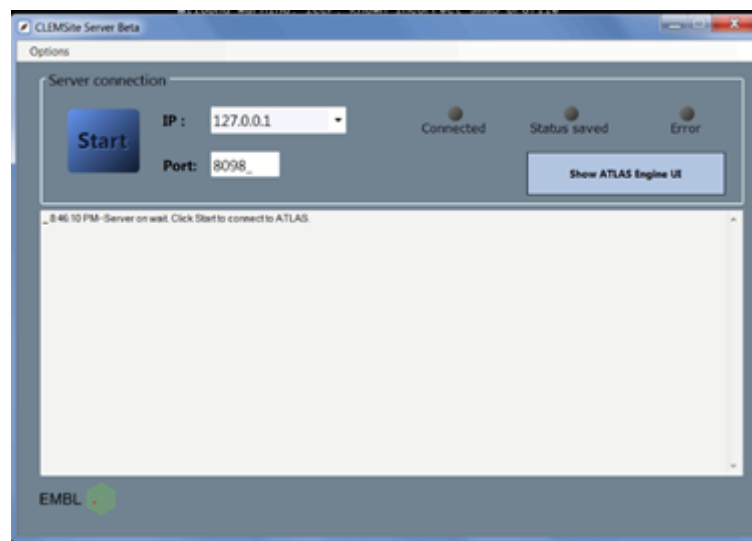
First, we will focus in the right top form, where there is a button called Connect.

3.1.1 Connecting to the FIB-SEM

WARNING: Be sure you have read [Before starting](#) and you fill the conditions stated there.

Any ATLAS software is recommended to be turned off before starting.

1) Start the *CLEMSite Server* (**clemsite-master\CLEMSiteServer\TestApp\bin\x64\Debug\msite.exe**):

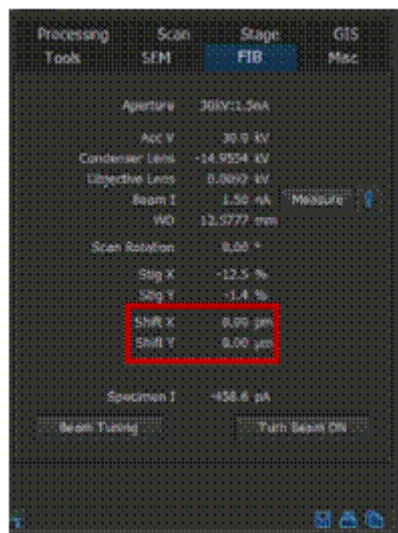
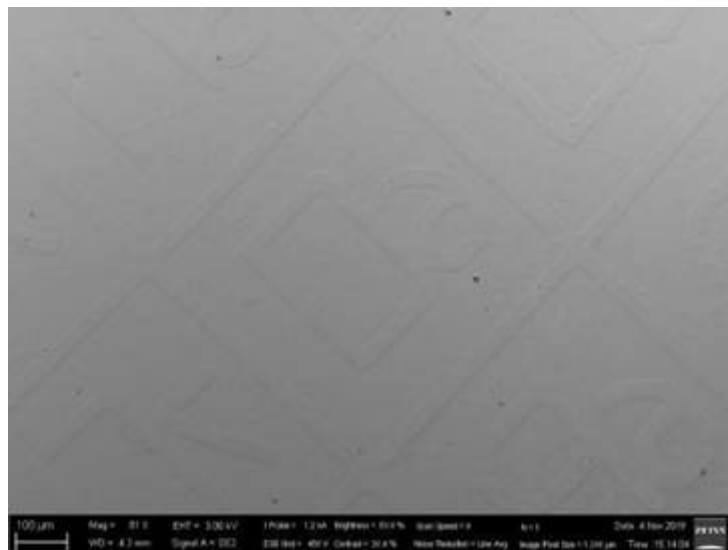


The IP address and port can be selected (IP's can be obtained from the system automatically). The default port is 8098 but could be changed to any other port. If the IP address and port are changed, the server must be stopped and started again. The IP address and port must then be provided to the client application.

2) In the Navigator go to the **Connect** button and click on it. The port and IP are usually 8098 and the localhost. They have to match with the *CLEMSite Server*. If it goes well a yellow icon will lit. This will start the AtlasEngine software (it is important that it was closed before). In the server, click on **Show Atlas UI** button and see that the ATLAS Engine UI pops up.

3) In a FIB-SEM, the sample is loaded onto the stage and positioned below the SEM beam. In this case, the sample is a flat, resin disc containing a monolayer of embedded cultured cells that is mounted onto an SEM stub and gold sputter coated.

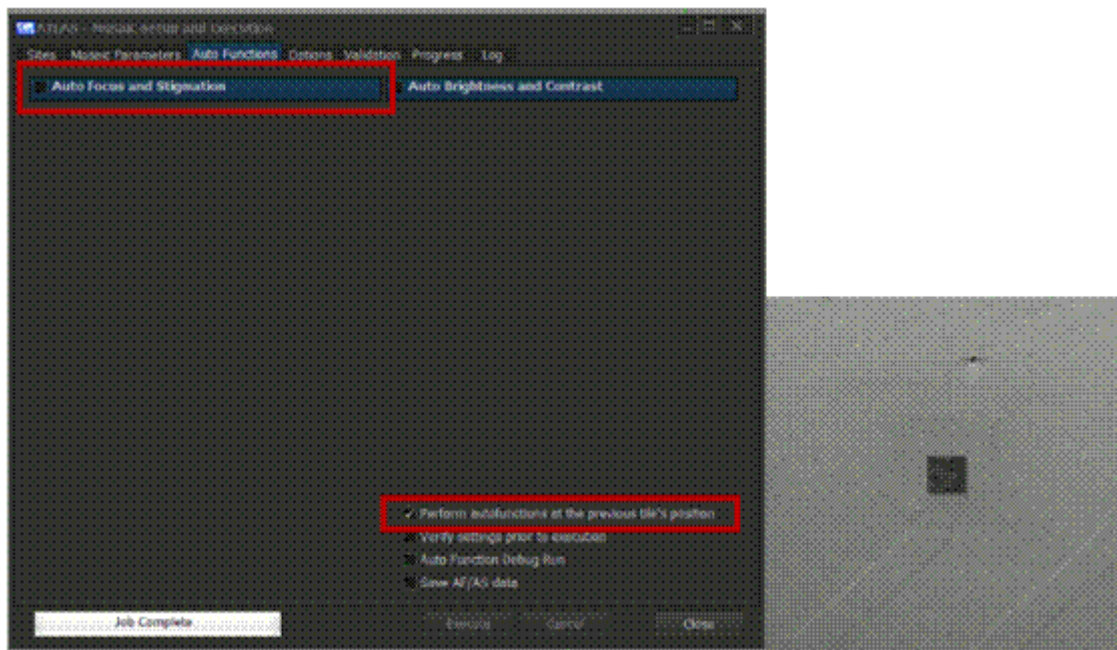
As in any FIB-SEM, there are several steps to align and position the beams and sample before a full acquisition can occur. The sample is moved to the eucentric height (at 1.5 kV SEM) at a safe distance to the objective lens and rotated in a way that the alphanumeric letters and the grid lines are at a 45° angle ensuring best illumination for the pattern detection.



Orientation of sample in the FIB-SEM 45 degrees, and SEM beam shift X,Y to 0.

Perform a coincidence point close to your 1st region of interest to ensure imaging and milling at the same position. You can perform a SEM adjustment, by checking the wobbler to ensure a nicely centered SEM beam through the SEM aperture. Stigmators and focus must be set to obtain the best possible crisp image. Before starting the automation in the Atlas Engine the beam shift for the SEM and the FIB beam are set to 0. To compensate for the tilted stage and the angle between the SEM beam and the sample when imaging, the tilt compensation must be set to 54°.

CLEMSite is using the mosaic function from ATLAS5 to acquire images. Therefore, make sure you have selected no auto functions in the mosaic setup. Otherwise, an autofocus might be performed with the FIB before acquiring SEM images, which could lead to an out of focus image and on problems in the workflow. Before starting, save the 'Mosaic Setup' without any auto functions, with 'Perform auto functions at the previous tile's position' unchecked.



When connecting Msite4 and CLEMSite Server, you should also save the state of the microscope, which includes B&C values for all detectors. Make sure that for the SEM the coordinate system is well visible in the SESI detector (secondary electron detector). For the SEM EsB detector (backscattered electron detector) you should be able to see the surface a bit (when the trench is done, the trench should be visible). For the FIB, set up the SESI detector with the 50 pA aperture (this will be giving you the reference images e.g. for coincidence point), therefore the images need to be in focus and the surface well visible.

5) Click on **Save state**. It takes 2 to 3 minutes for the server to check all beams and save all parameters, so before starting any operation, it is recommended to wait until no activity is visible seen on Atlas (no changes between FIB and SEM).

After saving the state, the LED icon is changing to green. Now it is possible to operate with the microscope using the Navigator.

5. SEM Navigator

Go to [CLEMSite-SEM Navigator](#)

6. FIB/SEM Acquisition

Go to [CLEMSite FIB/SEM Acquisition](#)

7. Some troubleshooting

Some problems that can happen to you. If you have other ones, go the Issues section in Github and report an issue.

```
1) Could not install packages due to an EnvironmentError: [WinError 5] Access is denied: "...PyQt5\\QtCore.pyd"
```

Solution: type pip install pyqt5

2) `ModuleNotFoundError: No module named 'numpy.core._multiarray_umath' ImportError: numpy.core.multiarray failed to import.`

Solution: type pip install numpy == 1.16.* or reinstall tensorflow

3) `Error: Tensorflow corrupted the numpy installation. Reinstall numpy.`

Solution: type pip install numpy == 1.16.*

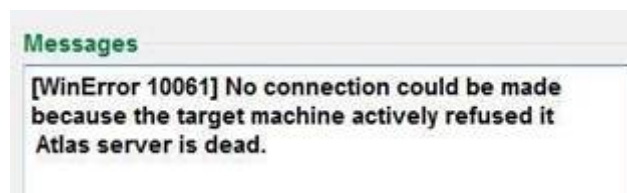
4) `ImportError: cannot import name '_validate_lengths' from 'numpy.lib.arraypad' (C:\Users\username\Anaconda3\lib\site-packages\numpy\lib\arraypad.py)`

Error: sci-kit image below 1.14 has reported this bug in their library. Install 1.14.2 or superior

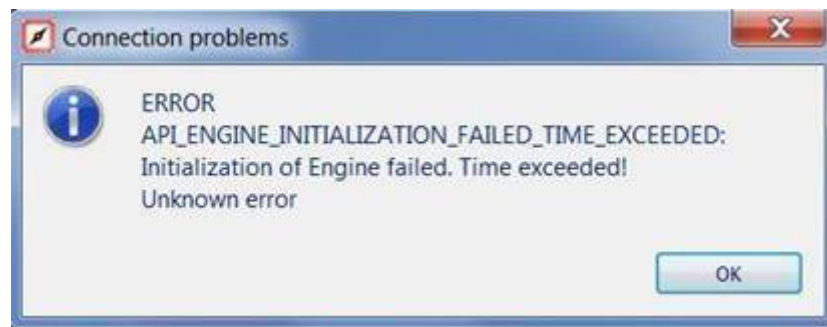
Solution: type pip install scikit-image==0.14.2

5) **Unexpected popping ups in AtlasEngine.** With the automatic *CLEMSite* routine the mosaic window from Atlas is popping up. This can be ignored, the imaging function is used to acquire the images.

6) **Connection lost** Atlas is not running. Open again the *CLEMSite* server and then *CLEMSite* needs to be opened and connection re-established again.



7) **API_ENGINE_INITIALIZATION_FAILED_TIME_EXCEED** If you see this pop up.



Atlas is not running, close AtlasEngine, start the *CLEMSiteServer*, Connect with *CLEMSite* (Navigator or Multisite), then click on Start ATLAS Engine UI.

Authors : Jose Miguel Serra Lleti, Anna Steyer

email : serrajosemi@gmail.com