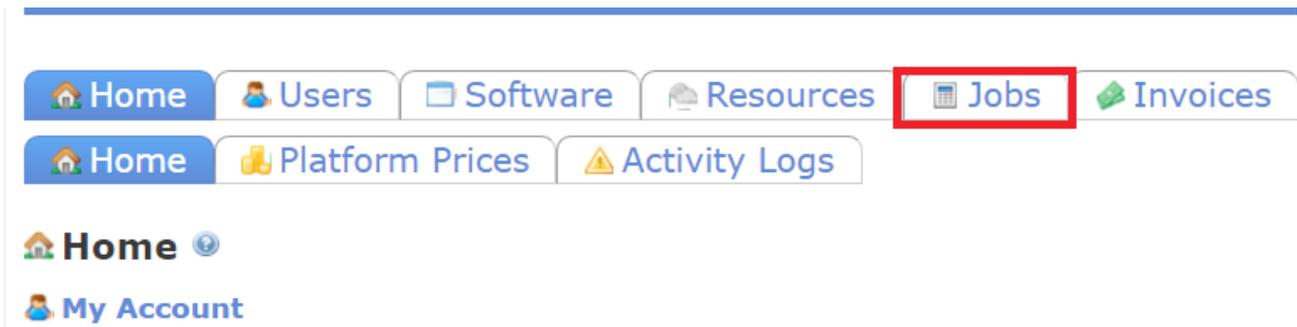# Run jobs in the cloud using the CloudSME-Platform:

In this tutorial, we are going to run a render-example of the Blender-software in the CloudSME Platform.

1. To run a Simulation in the cloud using the CloudSME platform, simply just log in to the platform and navigate to the „jobs" tab *(Fig. 1)*.

(Fig. 1: Location of the „Jobs"-tab)

2. Once your in the „jobs"-tab, click on the green „new" icon *(Fig. 2)* on the left side under the heading. A new wizard for the job-creation will open.
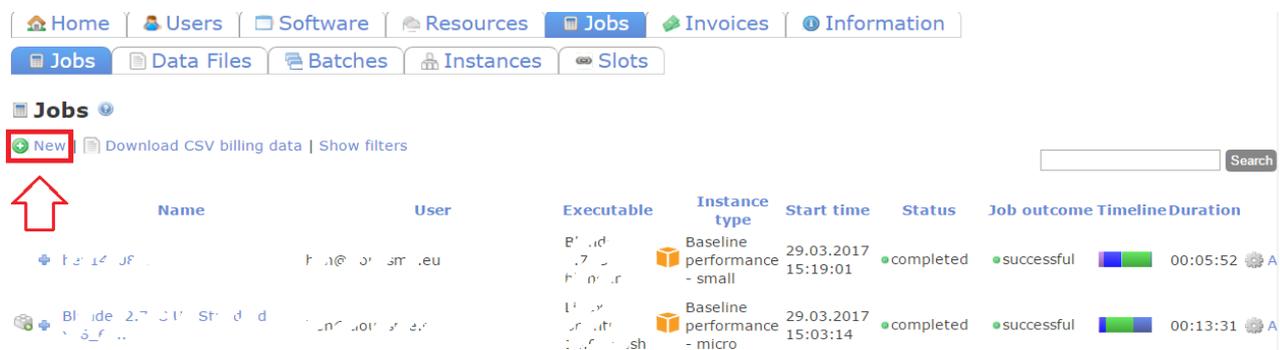
(Fig. 2: „new"-icon)

3. There are some required fields in the job-wizard. They are marked with „*". The field description can be found below.

- **Name (*)**
  - This is the name of your job. It acts as an identifier.
- **Software (*)**
  - The dropdown-menu shows all of the deployed software available on the platform
- **Executable(*)**
  - The executable of the chosen software will be picked automaticly and does not need to be changed.
- **Resource(*)**
  - The resource dropdown-menu shows all infrastructur-resources that your software is able to run on.
- **Region(*)**
  - Every resource can be splitted into several regions you can choose of. Different regions may have a different legal position in the processing and storage of electronic data. Also the speed of the data can vary depending on the region.
- **Instance Type(*)**
  - Instance types describe the selected computing capacity of the machine used to run the simulation. As far as they are deposited, it is possible to choose between different configurations.
- **Nodes**
  - If you want to run a job on several instances in parallel, please give the total number of nodes for it here. This option only makes sense for certain software and executables that are parallelized. Also, often the argument string and/or input file of the job have to reflect the same choice of nodes or the corresponding number of CPU cores.
- **Argument String**
  - An Argument String is needed to specify the exact simulation parameters via an commandline interface on the cloud-instance. To find detailed information about the needed parameters, please read the documentation of the corresponding software you want to use.

**4.** You can find the configuration of our blender test job in the following pictures *(Fig. 3)*.

(Fig.3: The parameters of the Blender test job)

5. If you made all the necessary configurations of the job, you can save it by clicking on the green "save" button *(Fig. 4)* in the lower left edge of the window. Once you did this, you will be redirected to a detail page for your job *(Fig. 5)*. If everything looks correct, please go to the "Jobs"-tab by clicking on the "Jobs"-Button.

(Fig.4: Save button)
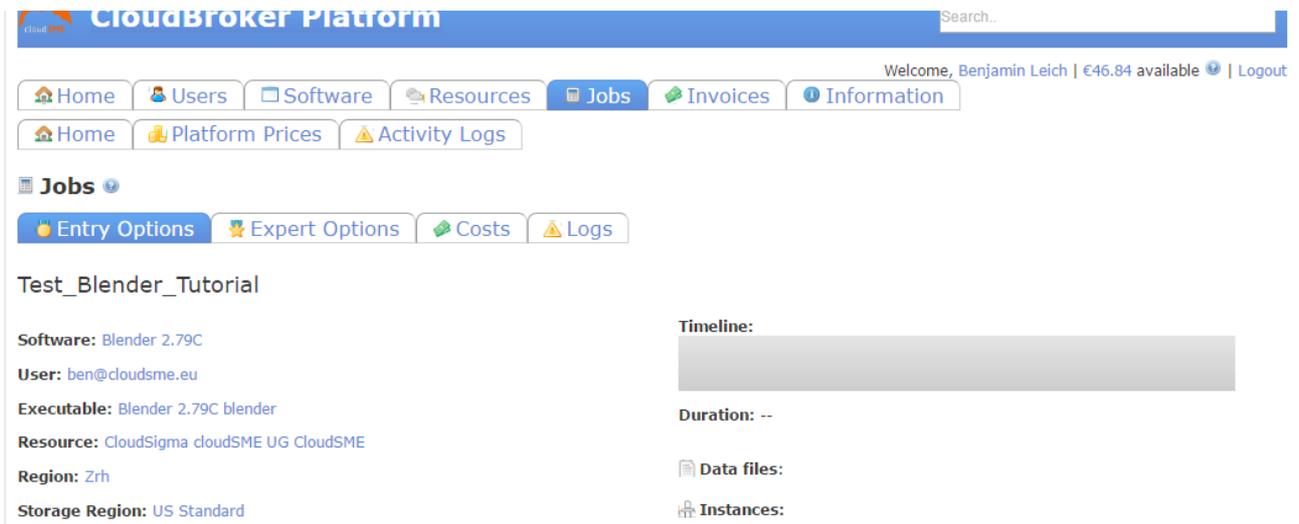
(Fig.5: Job detail page)

**6.** Now you have to provide your actual input data to the job. We downloaded a blender test-file for this tutorial. It can be found via the following link:

Download test input

7. Once you have your input-data, you have to upload it to the platform by hovering over the "actions" field behind your job in the jobs-list *(Fig. 6)*. A dropdown-menu will appear *(Fig. 7)*. You have to choose "data file" and upload your input data in the next window *(Fig. 8)*. After that, you save your configuration and wait until the upload has finished. An alert will appear to show you the costs of the storage you need *(Fig. 9)* for the upload of your input-file and also for the downloading of the output-file. Accept the alert and navigate back to the "jobs"-tab.

(Fig.6: Action field)
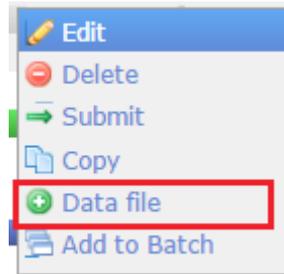
(Fig.7: dropdown menu with file upload)

(Fig.8: Uploading a File)
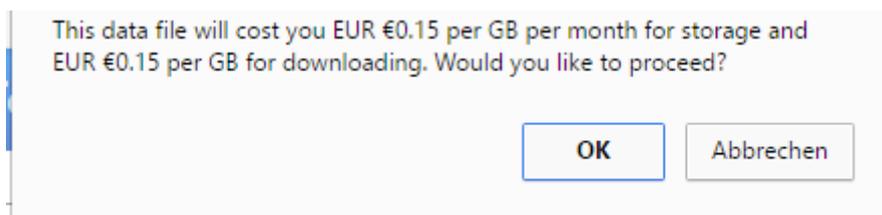
(Fig.9: Alert)

**8.** Now it is time to send the job into the cloud. Open the "actions"-menu on the right side behind your job in the list, then choose the submit button *(Fig. 10)*. The Platform will start the process and generates a new image of the software in the cloud and deploys it on the resource you have choosed. The status of your job will be displayed in the status-section *(Fig. 11)*, and a detailed view of the job status by clicking on the timeline *(Fig. 12)*.
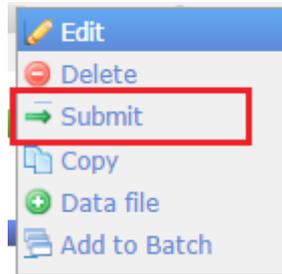
(Fig.10: Submit button)
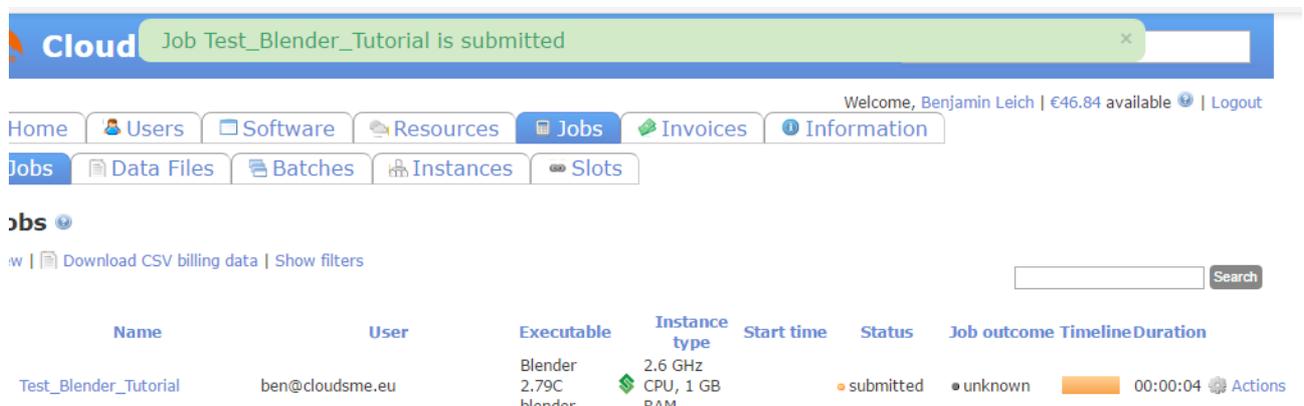
(Fig.11: Job submission prompt and job status bar)
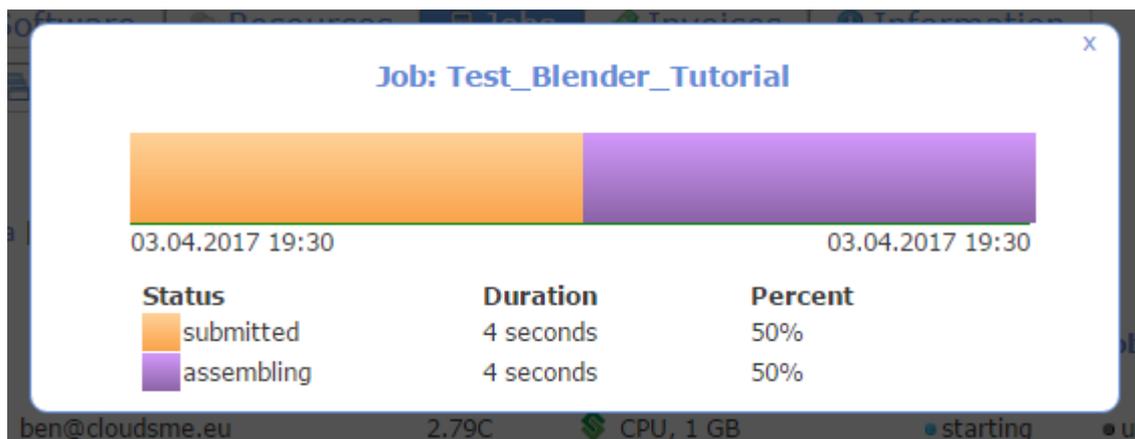
(Fig.12: Click on job status -detailview-)

9. If your job is completed, the status will indicate "completed" and if the calculation was successful, the job outcome will indicate "successful" *(Fig 13)*.

Name	User	Executable	Instance type	Start time	Status	Job outcome	Timeline	Duration	
Test_Blender_Tutorial	ben@cloudsme.eu	Blender 2.79C blender	2.6 GHz CPU, 1 GB RAM		• completed	• successful		00:02:09	Actions

(Fig.13: Job status successful)

10. Once he job is finished, you have to your download your job-output by clicking on the entry of your job in the job-list. Once you selected your job, you will be redirected to a detail-page of your your *(Fig. 14)*. Here you can download your output-files by opening the "Actions"-menu of the wanted data and select "download".

Jobs ⓘ

| 🟡 Entry Options | 🏆 Expert Options | 🟢 Costs | ⚠ Logs | ⬛ Stdout | ⚠ Stderr |

Test_Blender_Tutorial

Software: Blender 2.79C

User: ben@cloudsme.eu

Executable: Blender 2.79C blender

Resource: CloudSigma cloudSME UG CloudSME

Region: Zrh

Storage Region: US Standard

Instance type: CloudSigma cloudSME UG CloudSME 2.6 GHz CPU, 1 GB RAM

Nodes: 1

Argument string: -b 46bda883-f3d2-4d62-8714-f60433d04272.blend -o dolphin_ -a

Status: completed

Job outcome: successful

≡ Index | ⚙ Actions

Timeline:

Duration: 00:02:09

📄 **Data files** (4):
46bda883-f3d2-4d62-8714-f60433d04272.blend   ⚙ Actions
job.out   ⚙ Actions
job.err   ⚙ Actions
dolphin_   ⚙ Actions

⛁ **Instances:**
Zrh ben@cloudsme.eu R1_9   • halted

🏷 **Tags:**
ben@cloudsme.eu 2017 4

(Fig.14: Job details download page)