



UNIVERSITY OF TWENTE.

ITC GEOSPATIAL COMPUTING PLATFORM

dr.ing. Serkan Girgin MSc
s.girgin@utwente.nl

Center of Expertise in Big Geodata Science (CRIB) is a *horizontal facility* establish in **March 2020** to **enable** the better use of **big geodata technology** in *education, research, and institutional strengthening* activities at **ITC**

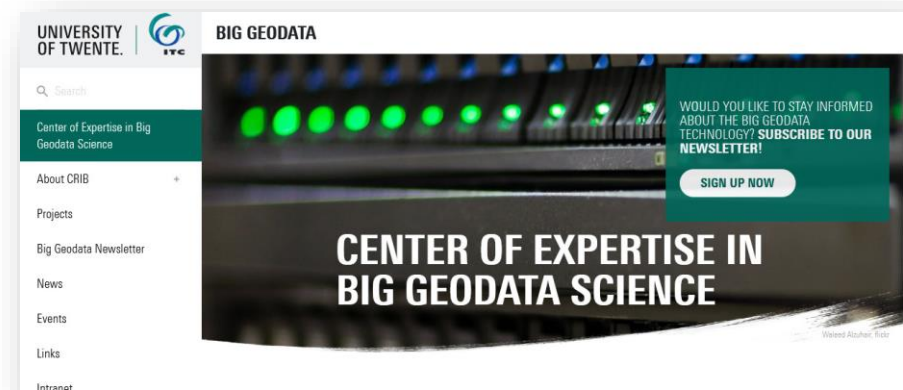
Mission

Collect, develop, and share **operational know-how** on big data technology to solve large-scale geospatial problems

Vision

Position UT/ITC as a *globally renowned* center of excellence in **geospatial big data** science.

<https://itc.nl/big-geodata>



ITC Geospatial Computing Platform

- Designed to serve primary activities identified by the needs assessment:
 - **Self learning**
 - **Exploratory research**
 - **Education**
- Design criteria
 - **Highly available** (24/7, no queue)
 - **Ready to use** (pre-installed scientific software)
 - **User friendly** (web-based UI)
 - **GPU enabled** (GPU for each user)
 - **Distributed-computing enabled** (computing cluster)
 - **Low cost** (feasible investment)
- **Limitations**
 - **Limited performance**
 - **Limited customization**
 - **UI overhead**
 - **Less-powerful GPU**
 - **Resource sharing**

NVIDIA Jetson AGX Xavier

- **8-core CPU**
(NVIDIA Carmel **ARMv8.2**, 2.26GHz, **NVIDIA L4T**)
- **512-core GPU**
(Volta architecture with 64 Tensor Cores)
- **32GB memory**
(256-bit LPDDR4x, 2133MHz, 137GB/s)
- **32GB storage**
(eMMC 5.1)
- **Dual Deep Learning Accelerator**
- **Vision Accelerator**
- **4x 4Kp60 video encoder**
(H.264/H.265)
- **2x 8Kp30 / 6x 4Kp60 video decoder**
(H.265)
- **Gigabit Ethernet**
(RJ45)
- **500 GB / 1 TB M.2 NVMe SSD**
(Samsung EVO 970 Plus, 3GB/s)



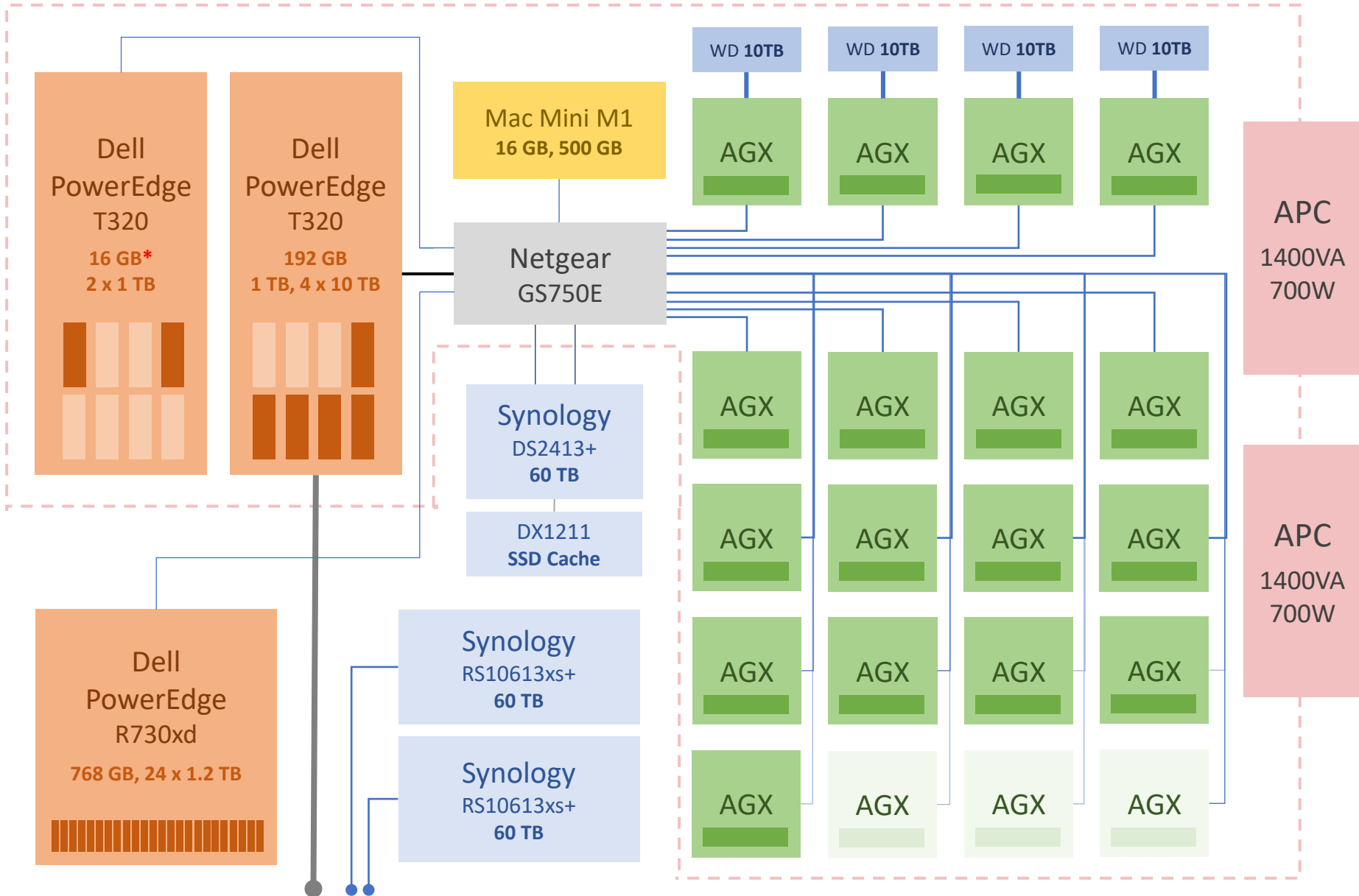
① <https://developer.nvidia.com/embedded/jetson-agx-xavier-developer-kit>

① https://elinux.org/Jetson_AGX_Xavier

Computing Resources

- **17*** **Computing Units** (NVIDIA Jetson AGX Xavier) (136 cores, 544 GB)
 - 8-core CPU (NVIDIA Carmel ARMv8.2, 2.26 GHz)
 - 512-core GPU (Volta architecture with 64 Tensor Cores)
 - 32GB memory (DDR4x, 137 GB/s)
 - 500 GB – 1 TB local storage (NVMe SSD, 3 GB/s)
- **Big Data Computing Unit** (Dell PowerEdge R730xd)
 - 2 x 8-core CPU (Intel Xeon E5-2640, 32 threads, 2.60 GHz)
 - 768 GB memory
 - 24 TB local storage (20 x 1.2 TB 2.5" 10K SAS 12 GB/s HDD, RAID 20+2)
- **2 Servers** (Dell PowerEdge T320)
 - 6-core CPU (Intel Xeon E5-2420 v2, 12 threads, 2.70 GHz)
 - 40 TB local storage (4 x 10 TB 3.5" 7.2K SAS HDD, RAID 2+1 (+1))
 - 192 GB* memory
- 160 TB external storage (0.2 PB total)

Computing Infrastructure



Platform as a Service

- Based on open-source software (Ubuntu, Docker, JupyterHub, ...)
- Accessible through a **web browser** (No software installation is required)
- **No registration** is required (Login with UT credentials)
- Each user has an individual and isolated **working environment**
- Each user has access to all available* **unit resources**, including **GPU**
- Each user has access to all available* **cluster resources**
- **Replicated storage** with minimum two copies (Hardware failure protection)
- **Distributed storage** for big data processing (HDFS)
- Automatically scales and **balances workload** among the units
- Low energy footprint (10-30W per unit)



Key Features

- **Interactive notebook, terminal and remote desktop** access are available
- Multiple interactive languages are supported (Python, R, Julia, Octave, Go, ...)
- **Up-to-date and optimized software packages** are **ready to use** (No setup required)
- Users can install additional packages (e.g., Python, R packages)
- Distributed computing clusters are **ready to use** (Dask, Apache Spark)
- **Public** assets are shared by all users
- **Shared workspaces** allow assets to be shared by selected users
- Access can be granted to **external users**
- **User support** is available*
- Provided and maintained by **CRIB** at no extra cost (i.e., free PaaS)



Current Usage

- Operational since **January 2021**
- **333** registered users
- **5-20** concurrent users at a time
- Provided approximately **20,000** hours of multi-core/GPU computation
- Overall, quite **positive feedback** from a wide-range of use cases
- Several **courses** started to use the platform
- Several **projects** started to use or will use the platform (e.g., IDEAMAPS, EO Africa R&D, TMT+ Bangladesh)
- Several **project proposals** consider to utilize the platform
- Other **UT units** (e.g., DCC, BDSI) are interested in having similar platforms
- **LISA** decided to build a similar platform for UT-wide use

Access

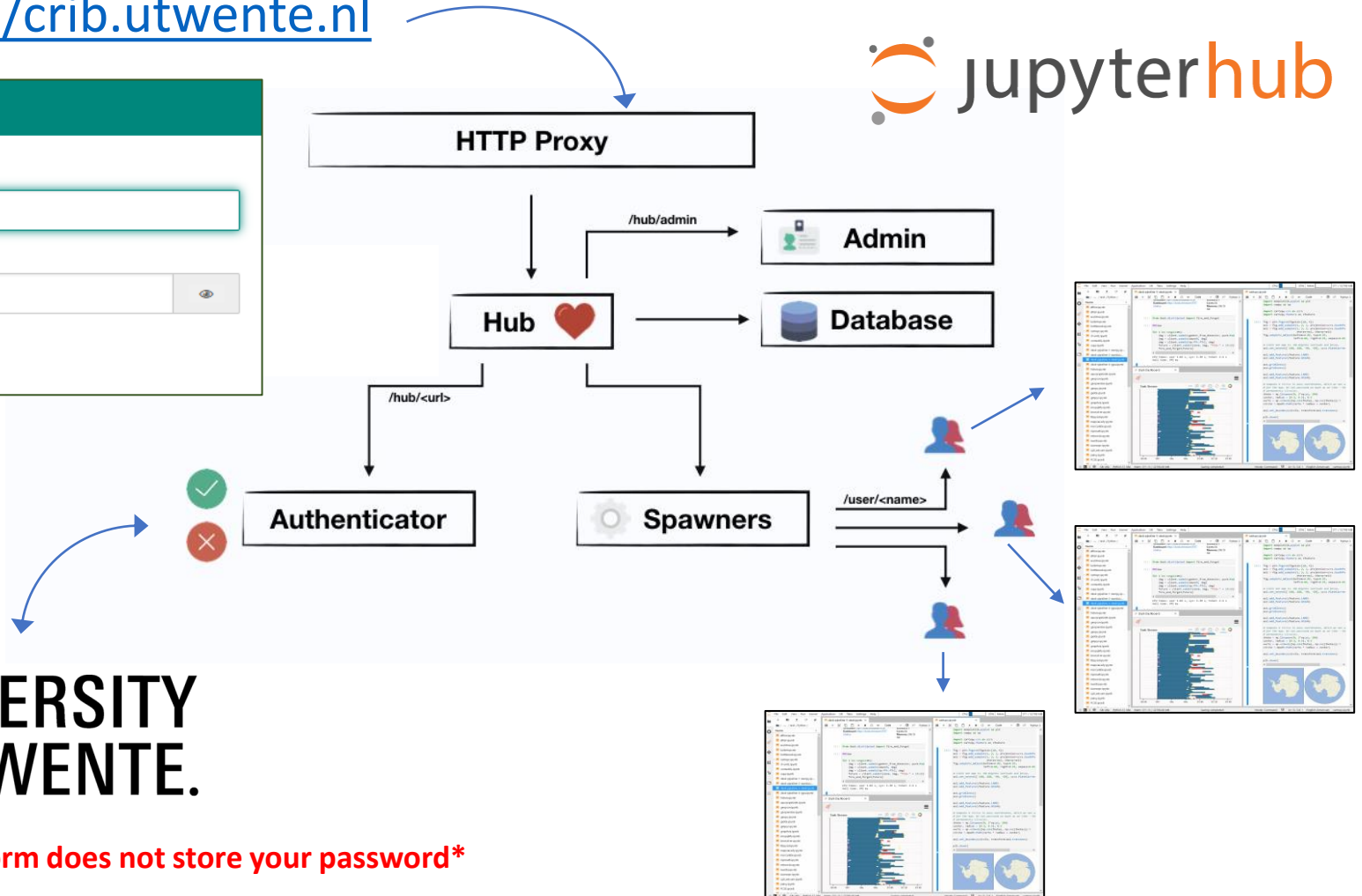
<https://crib.utwente.nl>



Sign In

E-mail Address:

Password:



**UNIVERSITY
OF TWENTE.**

The platform does not store your password*

① JupyterHub: <https://jupyter.org/hub>

Computing Environment = docker

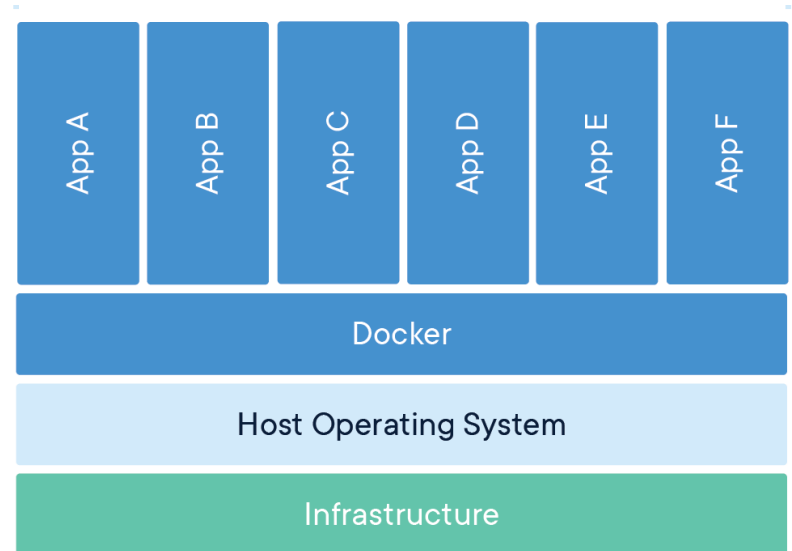
“**Docker** is a set of platform as a service (PaaS) products that use OS-level virtualization to deliver software in packages called **containers**.

Containers are isolated from one another and **bundle their own software**, libraries and configuration files.

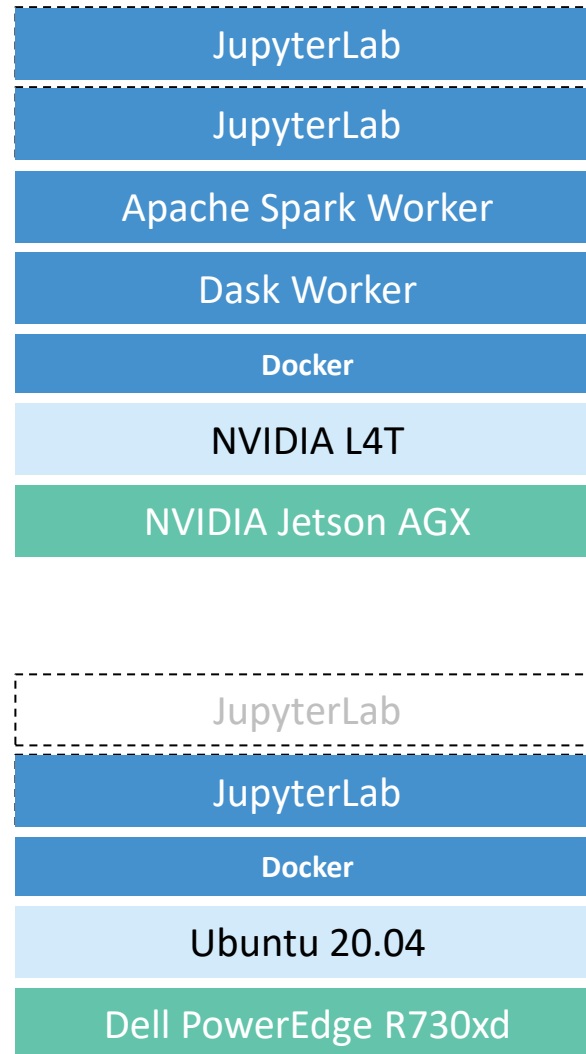
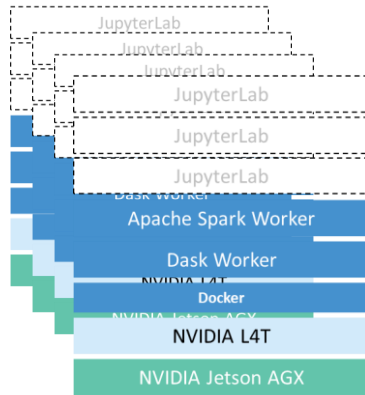
They can communicate with each other through well-defined channels.

Because all of the containers share the services of a single operating system kernel, they use fewer resources than virtual machines.”

[Wikipedia](#)



Container Orchestration



Base image

- Based on Ubuntu 20.04
- Includes all non-UI software
- 18.5 GB

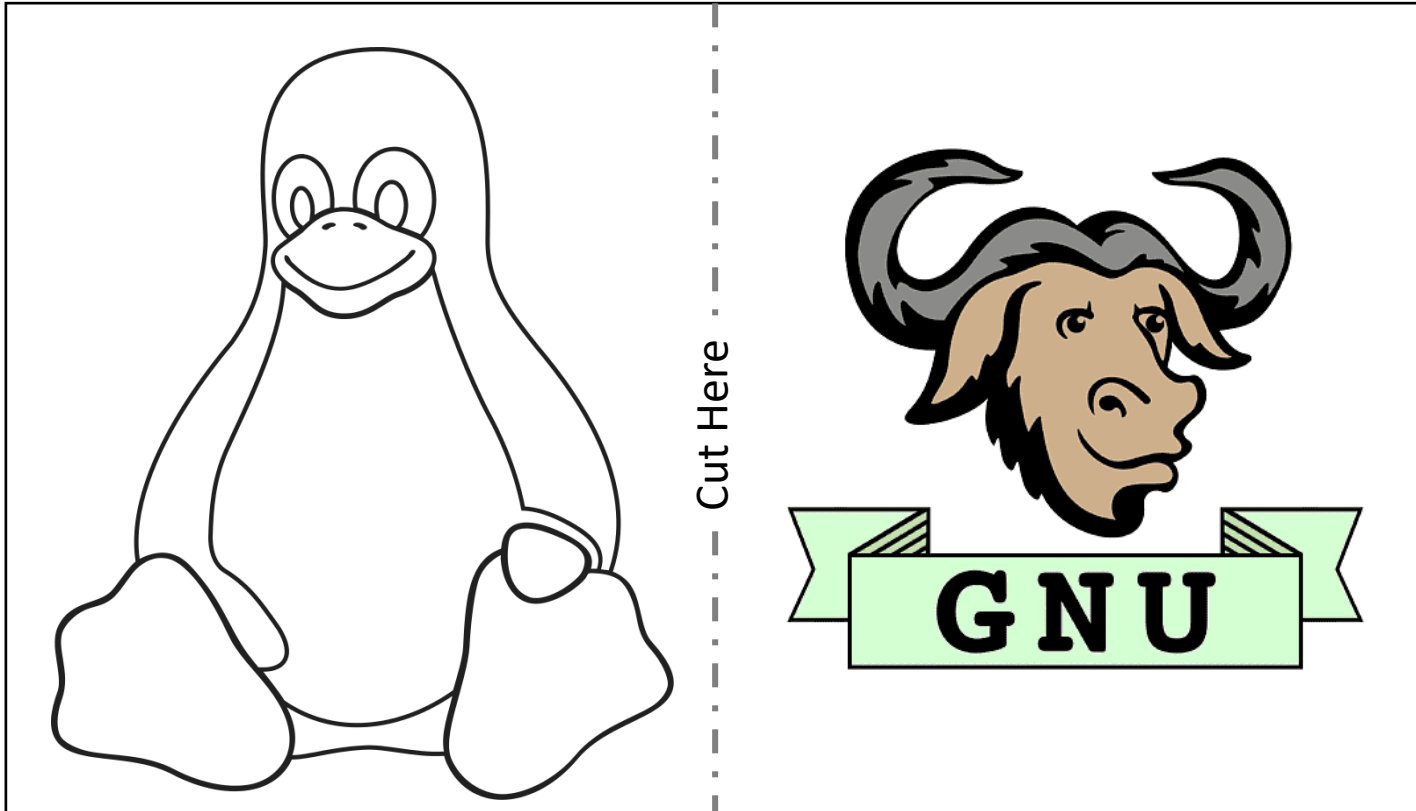
JupyterLab image

- Based on the base image
- Includes additional UI software (e.g. desktop applications)
- 20.5 GB

Computing Environment = docker

Container Image (common, volatile)

Binded Volumes (personal, permanent)



Important Directories (= cut here)

- Home directory (full-access, also used by the system)
/home/jovyan or **~**
- Private directory (full-access)
~/private or **/data/private**
- Public directory (read-only, maintained by CRIB)
~/public or **/data/public**
- Shared directories (read-only or full-access)
~/shared/<directory> or **/data/shared/<directory>**

Network
storage

- Local directory (full-access, PowerEdge R730xd only)
~/shared/<directory> or **/data/shared/<directory>**
- Temporary directory (full-access, not permanent)
/tmp
Useful for intermediate files during computation

Local storage
(fast)

Interactive Access



The screenshot displays a JupyterLab environment. On the left, a file browser shows a directory structure with files like '1-Hello.ipynb', '2-Data-Ana...', '3-Interactiv...', '4-Geospati...', '5-Interactiv...', '6-Geospati...', '7-R.ipynb', and '8-dask.ipynb'. The main area is split into two panes. The top pane shows a code editor with the following Python code:

```
[1]: import rasterio
from rasterio.plot import show

dir = '/data/public/GEODATA/Various-Netherlands/Aerial-Photogr
img = rasterio.open(dir + '253000_470000.tif')
show(img)
```

Below the code, an aerial photograph is displayed with axes ranging from 253000 to 254000 on the x-axis and 470000 to 470500 on the y-axis. The bottom pane shows a histogram titled 'Histogram' with the following code:

```
[2]: from rasterio.plot import show_hist
show_hist(img, bins=50, lw=0.0, stacked=False, alpha=0.3, histty
```

The histogram shows a distribution of values with a frequency axis ranging from 0.0 to 3.5 (scaled by 1e6) and an x-axis from 0 to 250. The histogram is overlaid on the aerial photograph. On the right side of the interface, a 'Living Textbook' is open, displaying the title 'Aerial survey' and the section 'Introduction'. The text in the introduction reads:

Aerial photographs are a major source of digital data; soft-copy workstations are used to digitize features directly from stereo pairs of digital photographs. These systems allow data to be captured in two or three dimensions, with elevations measured directly from a stereo pair using the principles of photogrammetry. Analogue aerial photos are often scanned before being entered into a soft-copy system, but with the advance of high-quality digital cameras this step can now be skipped.

In general, the alignment of roads and railways, lakes and water, and shapes of buildings are easily interpreted on aerial photographs - assuming that the scale of the photographs is not too small. Also, constructions such as dikes, bridges, air fields and the main types of vegetation and cultivation are mostly clearly visible. Nevertheless, numerous attribute data related to terrain features cannot be interpreted on aerial photographs: e.g. the administrative qualification of roads, sea and lake depths, functions of buildings, street names, and administrative boundaries. We will have to collect this information in the field or from existing data sets and maps (e.g. road maps, navigational charts or town plans).

Terminal Access

```
UP -DIR May 13 16:27
./
./astropy 4 Apr 25 11:10
./cache 25 May 11 17:16
./config 29 May 18 14:46
./cupy 3 Feb 7 20:49
./dbus 3 Jan 31 06:08
./glue 4 Apr 25 11:10
./gnupg 8 May 28 11:18
./grass7 5 Apr 5 11:01
./ipynb_checkpoints 11 May 18 14:09
./ipython 6 Feb 16 15:27
./java 4 Mar 14 01:00
./julia 7 Apr 1 16:51
./jupyter 6 May 13 09:00
./jupyter-php 3 Feb 19 03:57
./keras 4 Apr 6 07:48
./local 3 Jan 15 15:27
./netlogo 3 Mar 14 01:00
./npm 3 May 13 14:55
./rstudio-desktop 22 May 13 14:26
./snap 18 May 2 14:04
./ssh 3 Mar 16 14:19
./subversion 6 Apr 5 11:06
./virtual_documents 2 May 20 13:14
./vnc 223 May 20 11:18
./wine32 18 May 19 11:07
./wine64 9 May 19 11:07
./Desktop 2 May 14 23:12
./artifacts 5 Apr 1 13:54
./compiled 3 Jan 17 17:21
./fragstats 2 May 14 23:23
./globetrotter 6 Jan 17 17:42
./grass 3 Mar 13 13:29
./ilwisdata 9 May 11 17:22
./mlruns 4 Feb 9 15:09
./packages 31 Apr 1 13:54
-private 13 Jan 15 15:27
-public 12 Jan 15 15:27
./pysal_data 2 Jan 24 21:58
./qemu 6 May 14 23:04
-shared 12 Jan 15 15:27
./tools 19 May 14 23:11
.ICEauthority 39732 May 20 11:18
.Rhistory 3689 May 13 14:26
.Xauthority 10280 May 20 11:18
UP -DIR
```

- `bash` shell is available through the JupyterLab interface
 - Launcher > Terminal or File > New > Terminal

① bash Tutorial for Beginners: <https://linuxconfig.org/bash-scripting-tutorial-for-beginners>

Remote Desktop Access

The screenshot displays a remote desktop environment with several applications open. In the background, a QGIS window titled 'Untitled Project - QGIS' is visible, showing a map and various toolbars. Overlaid on top is an RStudio window titled 'RStudio' with a project named 'Untitled1'. The RStudio interface includes a menu bar, a toolbar, a script editor with R code, a console, and a plot window. The R code in the script editor is as follows:

```
1 # ggplot2 examples
2 library(ggplot2)
3
4 # create factors with value labels
5 mtcars$gear <- factor(mtcars$gear, levels=c(3,4,5),
6 labels=c("3gears", "4gears", "5gears"))
7 mtcars$sam <- factor(mtcars$sam, levels=c(0,1),
8 labels=c("Automatic", "Manual"))
9 mtcars$cyl <- factor(mtcars$cyl, levels=c(4,6,8),
10 labels=c("4cyl", "6cyl", "8cyl"))
11
12 # Kernel density plots for mpg
13 # grouped by number of gears (indicated by color)
14 qplot(mpg, data=mtcars, geom="density", fill=gear, alpha=
15 I(.5),
16 main="Distribution of Gas Milage", xlab="Miles Per
17 Gallon",
18 ylab="Density")
```

The console window shows the execution of the R code, and the plot window displays a kernel density plot titled 'Distribution of Gas Milage'. The plot shows three overlapping density curves for different gear types: 3gears (red), 4gears (green), and 5gears (blue). The x-axis is labeled 'Miles Per Gallon' and ranges from 10 to 35. The y-axis is labeled 'Density' and ranges from 0.00 to 0.12. A legend on the right side of the plot identifies the colors for each gear type.

- **XFCE Desktop Environment** is available through the JupyterLab interface
 - Launcher > Remote Desktop

① XFCE Desktop Environment: <https://www.xfce.org/>

Available Software







<https://crib.utwente.nl>



and hundreds more...

Additional Services

<https://crib.utwente.nl>

Service	Internal URL	External URL
 GeoServer	geoserver:8080	https://crib.utwente.nl/geoserver/
 PostgreSQL	postgresql:5432	Please contact us
 <small>Version 4</small> pgAdmin	pgadmin:80	https://crib.utwente.nl/pgadmin4/
 MariaDB	mariadb:3306	Please contact us
	gitea:3000	https://crib.utwente.nl/git/
 GeoNode	geonode:8500	https://crib.utwente.nl:8500/

Additional Information




- Up-to-date information on the platform is available in the **public/platform** directory
 - benchmark/ : Performance benchmarks
 - config/ : Configurations of custom-built software packages
 - demo/ : Example interactive notebooks for demonstration purposes
 - test/ : Functional tests of selected packages and libraries
 - languages : List of available programming languages
 - apt-packages : List of installed system packages
 - python-packages : List of installed Python packages
 - r-packages : List of installed R packages
 - custom-built : List of custom-built software packages
 - whats-new : List of changes and platform updates
 - work-in-progress : Features under development
 - faq : Frequently asked questions

Performance

Unit	Single Core Score	Multi Core Score	Good For
NVIDIA Jetson AGX	559	3945	GPU
Dell PowerEdge T320	630	3562	Services
Dell PowerEdge R730xd	909	10346	Multi-core
Mac Mini M1*	1753	7695	Single-core

Geekbench 5 CPU Benchmark

AES-XTS, Text Compression, Image Compression, Navigation HTML5
SQLite, PDF Rendering, Text Rendering, Clang Camera, **N-Body Physics**
Rigid Body Physics, Gaussian Blur, **Face Detection**, Horizon Detection
Image Inpainting, HDR, Ray Tracing, **Structure from Motion**
Speech Recognition, **Machine Learning**

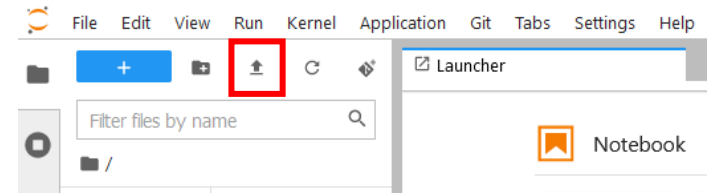
	AMD Ryzen Threadripper 3990X 2.9 GHz (64 cores)	25020
	AMD Ryzen Threadripper 3970X 3.7 GHz (32 cores)	22444
	Intel Xeon W-3175X 3.1 GHz (28 cores)	20286

1. NVIDIA Jetson AGX: <https://browser.geekbench.com/v5/cpu/7930832>
2. Dell PowerEdge T320: <https://browser.geekbench.com/v5/cpu/7930757>
3. Dell PowerEdge R730xd: <https://browser.geekbench.com/v5/cpu/7930707>
4. Mac Mini: <https://browser.geekbench.com/v5/cpu/7930907>

FAQ

- How can I upload files to the platform?

- Use the **Upload** button
- You can upload multiple files
- Files are uploaded to the active folder
- You cannot upload directories
- To **upload directories** with multiple files or sub-directories:
 - Create an archive file of the directory (e.g. zip, tar.gz)
 - Upload archive file
 - Extract archive file
 - On the terminal, `unzip <archive.zip>` or `tar xzvf <archive.tar.gz>`
 - On the remote desktop, Applications > Accessories > Archive Manager
- We will provide better options soon



FAQ

- How can I access files stored on the platform?
 - You can access files only through the platform
 - We will provide better options soon (e.g. [Nextcloud](#), [Owncloud](#))
 - Please [contact us](#) for specific needs

FAQ

- How can I install R, Python, Julia, etc. packages?
 - For **Python**, open a terminal and enter the command:
`pip install <package name> or pip install <package name>==<version>`
 - For **R**, enter the command:
`install.packages('<package name>', repos='https://cloud.r-project.org')`
 - For other languages, please refer to the user documentation
 - Packages are installed to your **home directory** (they are permanent)
 - They are not updated automatically (you should keep them up to date)
 - You may encounter installation errors if the package requires additional system libraries or it is not compatible with the selected architecture
 - Please [contact us](#) if you have difficulties, **we will install it for you**, which will make it also available to other users
 - **Warning: Local package dependencies are not guaranteed for platform updates**
 - **Warning: Local packages are architecture dependent**
 - [Conda](#) is not supported, but you can use [virtual environments](#), if necessary

FAQ

- How can I install additional software applications?
 - You can install portable **Linux** applications to your **home, private, or shared** directories
 - NVIDIA Jetson AGX: **arm64** or **aarch64**
 - Dell PowerEdge T320 and R730xd: **amd64** or **x86_64**
 - They are not updated automatically (you should keep them up to date)
 - You cannot install software that needs to be installed to system directories (e.g. `/usr` or `/usr/local`)
 - You cannot install software by using the default package manager of Ubuntu (`apt`)
 - Please [contact us](#) if you need additional software applications or libraries
 - Some software applications may need a dedicated infrastructure, hence availability will depend on available resources
 - **Warning: Local software might be architecture dependent**

FAQ

- How can I install Windows applications?
 - Windows applications are supported through emulation by [Wine](#)
 - They are not supported on NVIDIA Jetson AGX units (only Intel architecture)
 - They are not guaranteed to work 100%
 - For **Win32** applications, open a terminal and enter the command:
`WINEPREFIX="$HOME/.wine32" wine <setup.exe>`
 - For **Win64** applications, open a terminal and enter the command:
`WINEPREFIX="$HOME/.wine64" wine <setup.exe>`
 - C Drive is located at `$HOME/.wine<XX>/drive_c`
 - Please [contact us](#) if you have difficulties

FAQ

- Can I have multiple instances running at the same time?
 - A dedicated container is spawned for you when you first login
 - When you login again while you are logged in (e.g. from another browser window, or another machine) you connect to the same container
 - Please [contact us](#) if you need more instances (e.g. to finish your thesis), we can create an external account for you which will allow a second instance
 - **Suggestion:** Profile your code to understand resource usage and bottlenecks
 - **Suggestion:** Consider using distributed computing frameworks (e.g. [Dask](#))

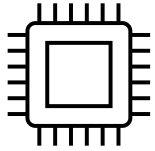
FAQ

- Is it possible to disconnect from the platform for a short period keeping running tasks active?
 - If you log out from the platform (File > Log out), your container is terminated. Hence, all running tasks are also terminated.
 - If you simply close the browser window, the platform keeps your container running for 1 hour.
 - If you re-connect within 1-hour period, you connect to the same container
 - **Warning:** When you close the browser window, running tasks lose output streams. Therefore, although they continue to run, you won't see their output after re-connecting to the same container.
 - **Suggestion:** Use file-based output (i.e. logging) to backup standard output whenever possible

FAQ

- Can I run long-duration computation tasks?
 - Yes, you can. The users, who use the platform for deep learning tasks, usually have tasks that last several days
 - You should be connected to the platform during the computation period, i.e. your web browser should be open
 - **Suggestion:** If you cannot ensure connection from your own computer, connect to a UT computer by remote desktop and use that computer to connect to the platform
 - **Suggestion:** For long-duration tasks do not trust service availability and implement precautionary measures (e.g. checkpoints)
 - We may provide a better option in the future
 - Please [contact us](#) for specific needs

Contact



<https://crib.utwente.nl>



<https://itc.nl/big-geodata>



crib-itc@utwente.nl



[@BigGeodata](https://twitter.com/BigGeodata)



[Big Geodata Newsletter](#)

