



TÜBİTAK
ULAKBİM

Turkish Academic Network and Information Center

Turkish National Science e-Infrastructure (TRUBA)

Sevil Sarıkurt, PhD

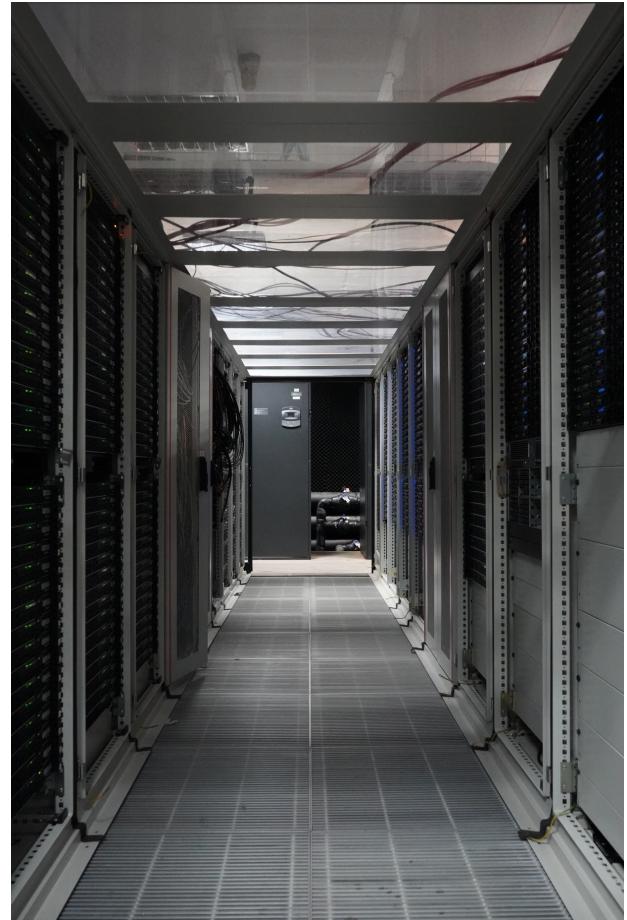
TÜBİTAK ULAKBİM

11 September 2023



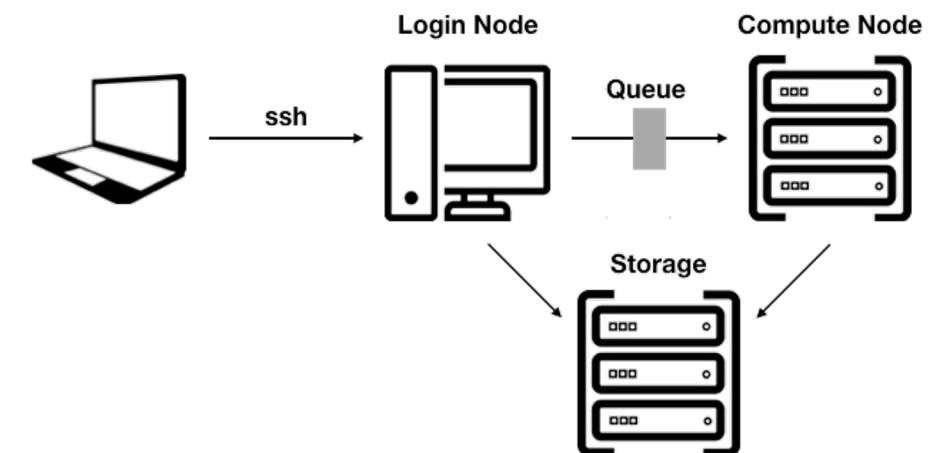
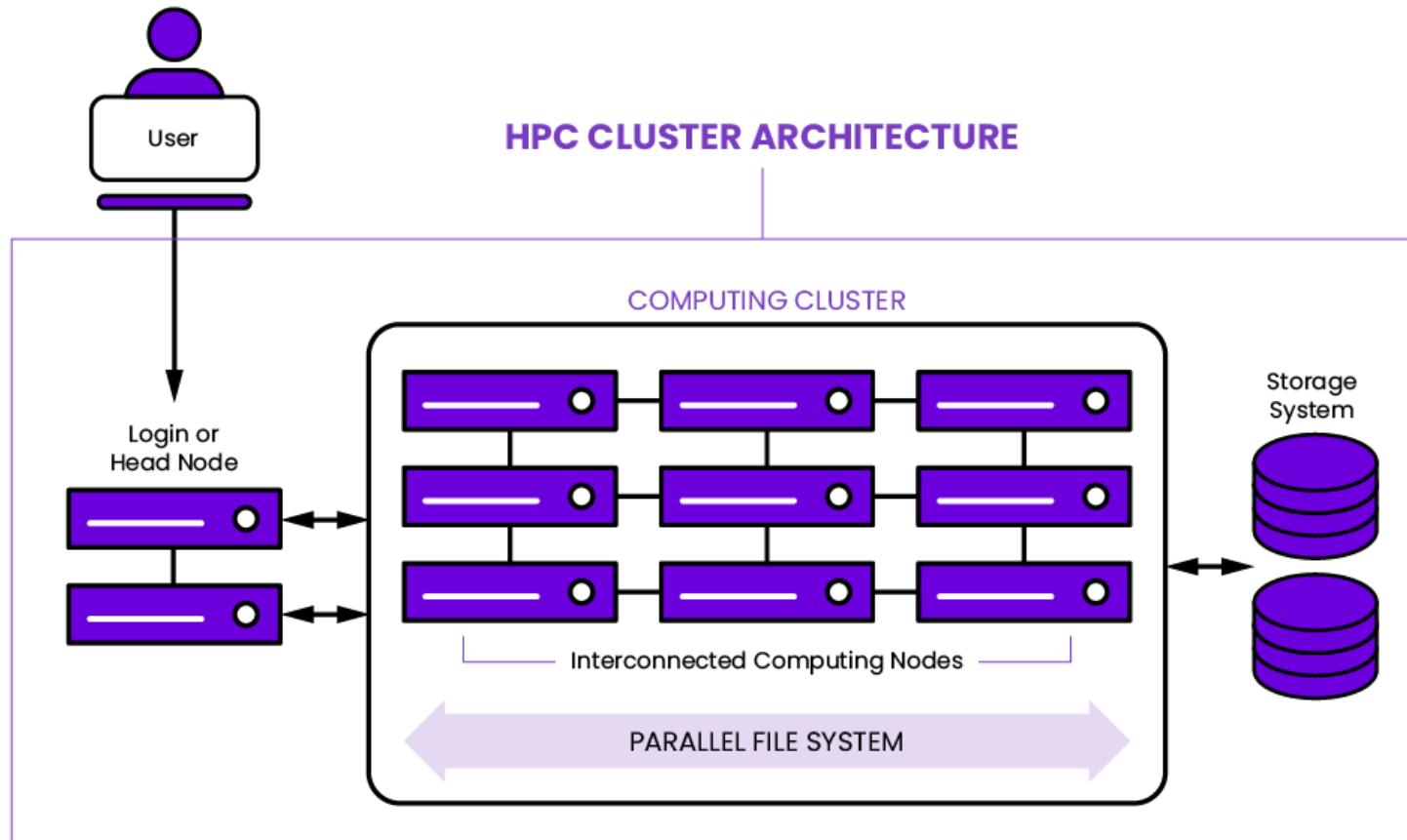
TRUBA
Turkish Science e-Infrastructure

Outline



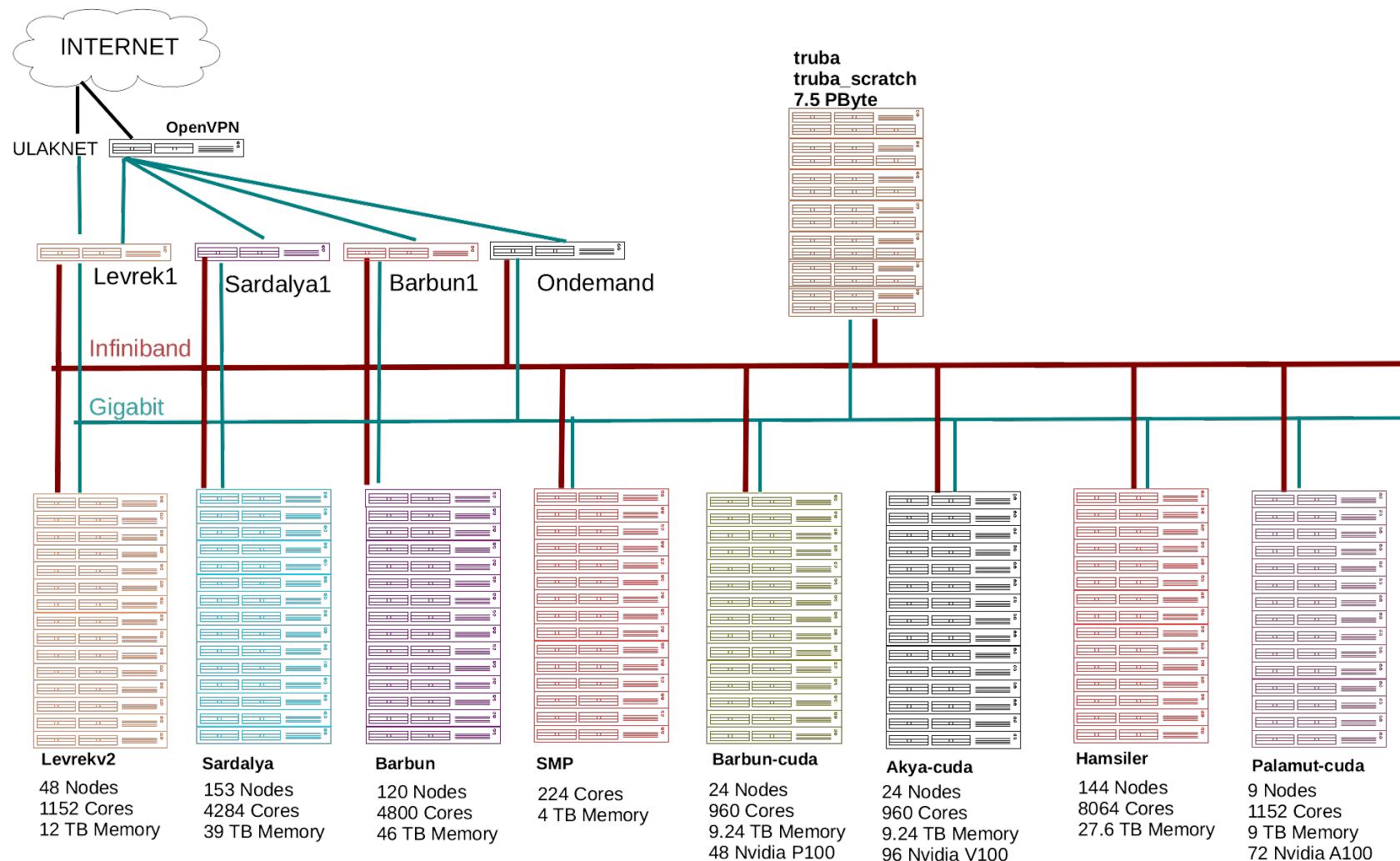
- HPC Architecture
- TRUBA HPC System
- OpenVPN
- SSH connection
- Jupyter Notebook
- SLURM Files

HPC Architecture



TRUBA – Turkish Science e-Infrastructure

- TRUBA: National center providing high performance computing and data storage for all research institutions and researchers in Türkiye.





❑ TRUBA 2023

- ✓ Supporting Organization: Presidency of Türkiye, Presidency of Strategy and Budget
- ✓ TRUBA Turkish Science e-Infrastructure

❑ Resources

- ✓ 25.000 CPUs



More than 60.000 CPUs are loading....

use of new generation GPUs

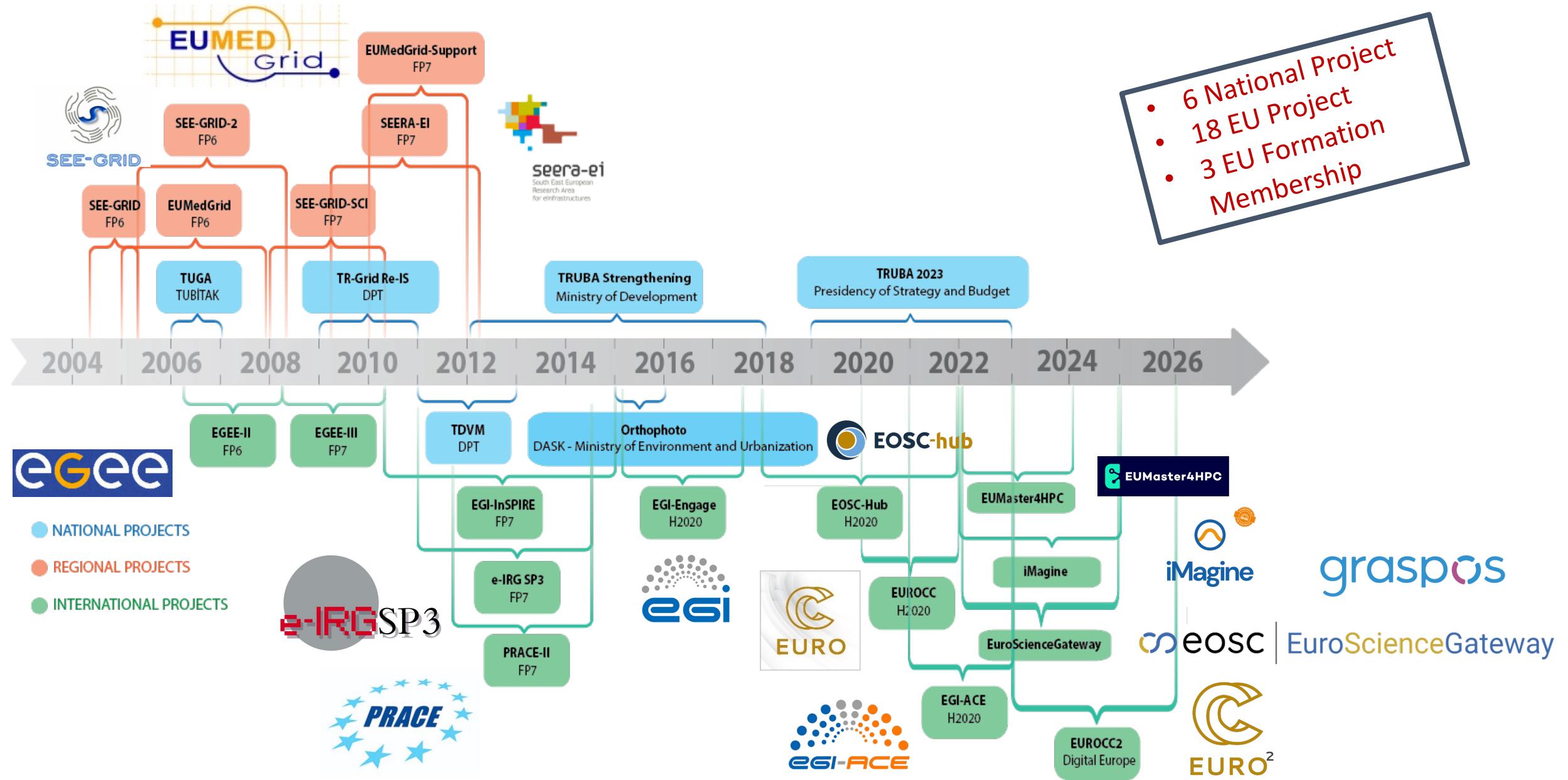
up to 100 Tflops

- ✓ 14 PB Lustre data storage space

❑ Usage

- ✓ 13M CPU hours/month
- ✓ > 5000 registered researcher
- ✓ > 400 researcher using the system at the same time
- ✓ > 165 National Scientific Research Project support
- ✓ National Corporate collaborations : Disaster and Emergency Management Presidency, Ministry of Environment, Urbanisation and Climate Change, Kandilli Observatory, Turkish Aerospace Industries, TÜBİTAK SAGE, TUBITAK National Observatory, TÜBİTAK Marmara Research Center.

TRUBA - Projects



TRUBA-GPU Resources

Calculation Nodes	Year	#Node	Architecture	Processor	Performance	RAM	#GPU
Barbun-cuda	2018	24	20 cores x 2 CPU +2 x Nvidia P100 GPU	Xeon Scalable 6148 2.40GHz	2048Gflops +9400Gflops	384 GB + 2x16 GB HBM2	2
Akya-cuda	2018	24	20 cores x 2 CPU +4 x Nvidia V100 GPU	Xeon Scalable 6148 2.40GHz	2048Gflops + 4x7800Gflops	384 GB + 4x16 GB HBM	4
Palamut-cuda	2021	9	64 cores x 2 CPU + 8 x Nvidia A100 GPU	AMD EPYC 7742 2.24 GHz	4600 Gflops + 8x9600Gflops	192GB + 8x80GB HBM	8

Compute Queues

partitions	nodes	#nodes	max run time	priority	min core
single	levrekv2	8	15-00:00:00	2000	1
hamsi	hamsi	144	03-00:00:00	2800	28
mid2	barbun	189	08-00:00:00	3200	4
long	barbun	189	15-00:00:00	3000	4
smp	orkinos	1	8-00:00:00	2800	4
sardalya	sardalya	100	15-00:00:00	2800	4
barbun	barbun	119	15-00:00:00	2800	4
interactive	levrekv2	14	15-00:00:00	3000	1
barbun-cuda	barbun-cuda	24	15-00:00:00	2800	20
akya-cuda	akya-cuda	24	15-00:00:00	2800	10
palamut-cuda	palamut	9	03-00:00:00	2800	16
debug	barbun, barbun-cuda, akya-cuda, orkinos	238	00-00:15:00	65535	1

OpenVPN

- **ULAKNET (in-campus@Turkiye):** `ssh -l user_name levrek1.ulakbim.gov.tr`
 - Users connecting from universities in Turkiye
- **Off Campus:** `ssh -l user_name 172.16.7.1`
 - *OpenVPN installation in advance*
 - *To use Open OnDemand service*
- Linux: installation through command line
 - `sudo apt-get install openvpn`

```
sevil@pop-os:~/VPN$ sudo openvpn TRUBA-genel.ovpn
2023-09-10 22:18:34 Note: Treating option '--ncp-ciphers' as '--data-ciphers' (renamed in OpenVPN 2.5).
2023-09-10 22:18:34 DEPRECATED OPTION: --cipher set to 'AES-256-CBC' but missing in --data-ciphers (AES-256-GCM:AES-128-GCM
). Future OpenVPN version will ignore --cipher for cipher negotiations. Add 'AES-256-CBC' to --data-ciphers or change --cip
her 'AES-256-CBC' to --data-ciphers-fallback 'AES-256-CBC' to silence this warning.
2023-09-10 22:18:34 OpenVPN 2.5.5 x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO] [LZ4] [EPOLL] [PKCS11] [MH/PKTINFO] [AEAD] buil
t on Jul 14 2022
2023-09-10 22:18:34 library versions: OpenSSL 3.0.2 15 Mar 2022, LZO 2.10
Enter Auth Username: sevil
 Enter Auth Password: *****
```

```
sevil@pop-os:~/VPN$ sudo openvpn TRUBA-genel.ovpn
2023-09-10 22:20:22 Note: Treating option '--ncp-ciphers' as '--data-ciphers' (renamed in OpenVPN 2.5).
2023-09-10 22:20:22 DEPRECATED OPTION: --cipher set to 'AES-256-CBC' but missing in --data-ciphers (AES-256-GCM:AES-128-GCM
). Future OpenVPN version will ignore --cipher for cipher negotiations. Add 'AES-256-CBC' to --data-ciphers or change --cip
her 'AES-256-CBC' to --data-ciphers-fallback 'AES-256-CBC' to silence this warning.
2023-09-10 22:20:22 OpenVPN 2.5.5 x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO] [LZ4] [EPOLL] [PKCS11] [MH/PKTINFO] [AEAD] buil
t on Jul 14 2022
2023-09-10 22:20:22 library versions: OpenSSL 3.0.2 15 Mar 2022, LZO 2.10
Enter Auth Username: sevil
 Enter Auth Password: *****
2023-09-10 22:20:29 TCP/UDP: Preserving recently used remote address: [AF_INET]193.140.99.241:1195
2023-09-10 22:20:29 UDP link local (bound): [AF_INET][undef]:1194
2023-09-10 22:20:29 UDP link remote: [AF_INET]193.140.99.241:1195
2023-09-10 22:20:29 [midye.truba.gov.tr] Peer Connection Initiated with [AF_INET]193.140.99.241:1195
2023-09-10 22:20:40 Options error: Unrecognized option or missing or extra parameter(s) in [PUSH-OPTIONS]:8: register-dns (
2.5.5)
2023-09-10 22:20:40 TUN/TAP device tun0 opened
2023-09-10 22:20:40 net_iface_mtu_set: mtu 1500 for tun0
2023-09-10 22:20:40 net_iface_up: set tun0 up
2023-09-10 22:20:40 net_addr_v4_add: 10.3.9.106/22 dev tun0
2023-09-10 22:20:40 WARNING: this configuration may cache passwords in memory -- use the auth-nocache option to prevent thi
s
2023-09-10 22:20:40 Initialization Sequence Completed

```

OpenVPN config file:

<https://tinyurl.com/TRUBA-openvpn>



```
sevil@pop-os:~$ ssh sevil@172.16.7.1
sevil@172.16.7.1's password: 
```



TRUBA
Turkish Science e-Infrastructure

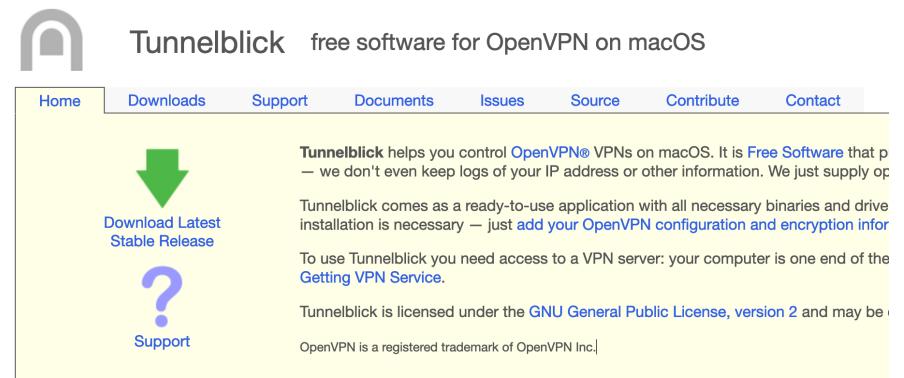
OpenVPN

- MacOS:
 - <https://tunnelblick.net/index.html>
 - <https://openvpn.net/client-connect-vpn-for-mac-os/>
- Windows: <https://openvpn.net/community-downloads/>

Windows 64-bit MSI installer	GnuPG Signature	OpenVPN-2.6.6-I001-amd64.msi
Windows ARM64 MSI installer	GnuPG Signature	OpenVPN-2.6.6-I001-arm64.msi
Windows 32-bit MSI installer	GnuPG Signature	OpenVPN-2.6.6-I001-x86.msi
Source archive file	GnuPG Signature	openvpn-2.6.6.tar.gz

OpenVPN config file:

<https://tinyurl.com/TRUBA-openvpn>



The screenshot shows the Tunnelblick homepage. At the top, there's a navigation bar with links for Home, Downloads, Support, Documents, Issues, Source, Contribute, and Contact. Below the navigation, there's a large green download button labeled "Download Latest Stable Release". To the right of the button, there's a brief description of what Tunnelblick is and how to use it. A support icon (a question mark inside a circle) is also present.



The screenshot shows the OpenVPN Connect for macOS landing page. It features a prominent "OpenVPN Connect for macOS" title. Below the title, there's a brief description of the software and its availability. A "Download OpenVPN Connect v3" button is located at the bottom right.

This is the official OpenVPN Connect client software for macOS developed and maintained by OpenVPN Inc. This is the recommended client program for the OpenVPN Access Server. The latest version of OpenVPN Connect client for macOS is available on our website.

[Download OpenVPN Connect v3](#)
sha256 signature: 4905f309bfedce5e75de48870735b616031aea9eb22d1702289ac98d57e0f85f
For macOS versions titled El Capitan, Sierra, High Sierra, Mojave, Catalina, Big Sur, Monterey, and Ventura.

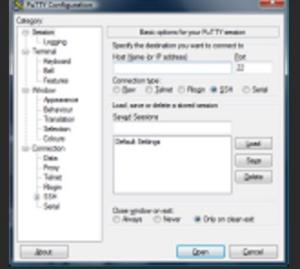


SSH Connection

Windows:

<https://www.putty.org/>

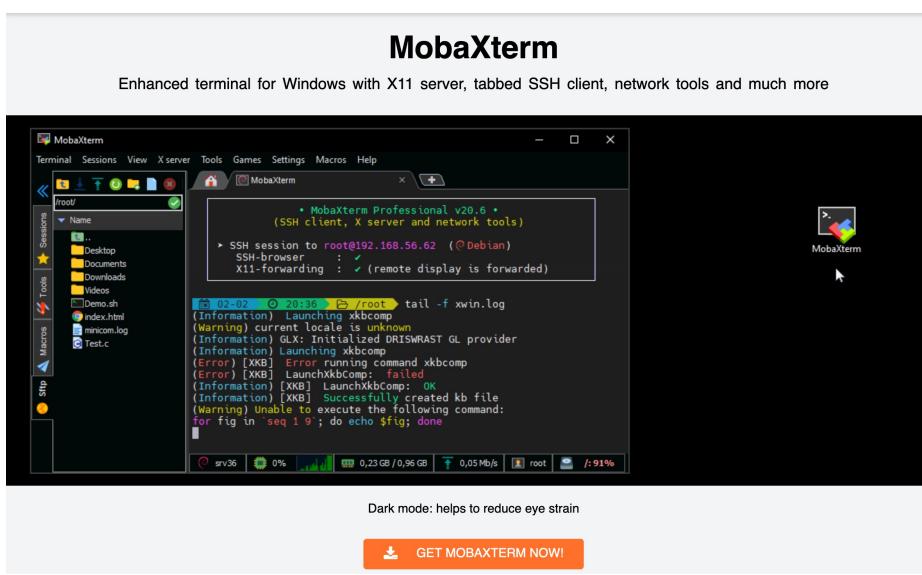
<https://mobaxterm.mobatek.net/>



Download PuTTY

PuTTY is an SSH and telnet client, developed originally by Simon Tatham for the Windows platform. PuTTY is open source software that is available with source code and is developed and supported by a group of volunteers.

[Download PuTTY](#)



Package files

You probably want one of these. They include versions of all the PuTTY utilities (Not sure whether you want the 32-bit or the 64-bit version? Read the [FAQ entry](#).)

We also publish the latest PuTTY installers for all Windows architectures as a free

MSI ('Windows Installer')

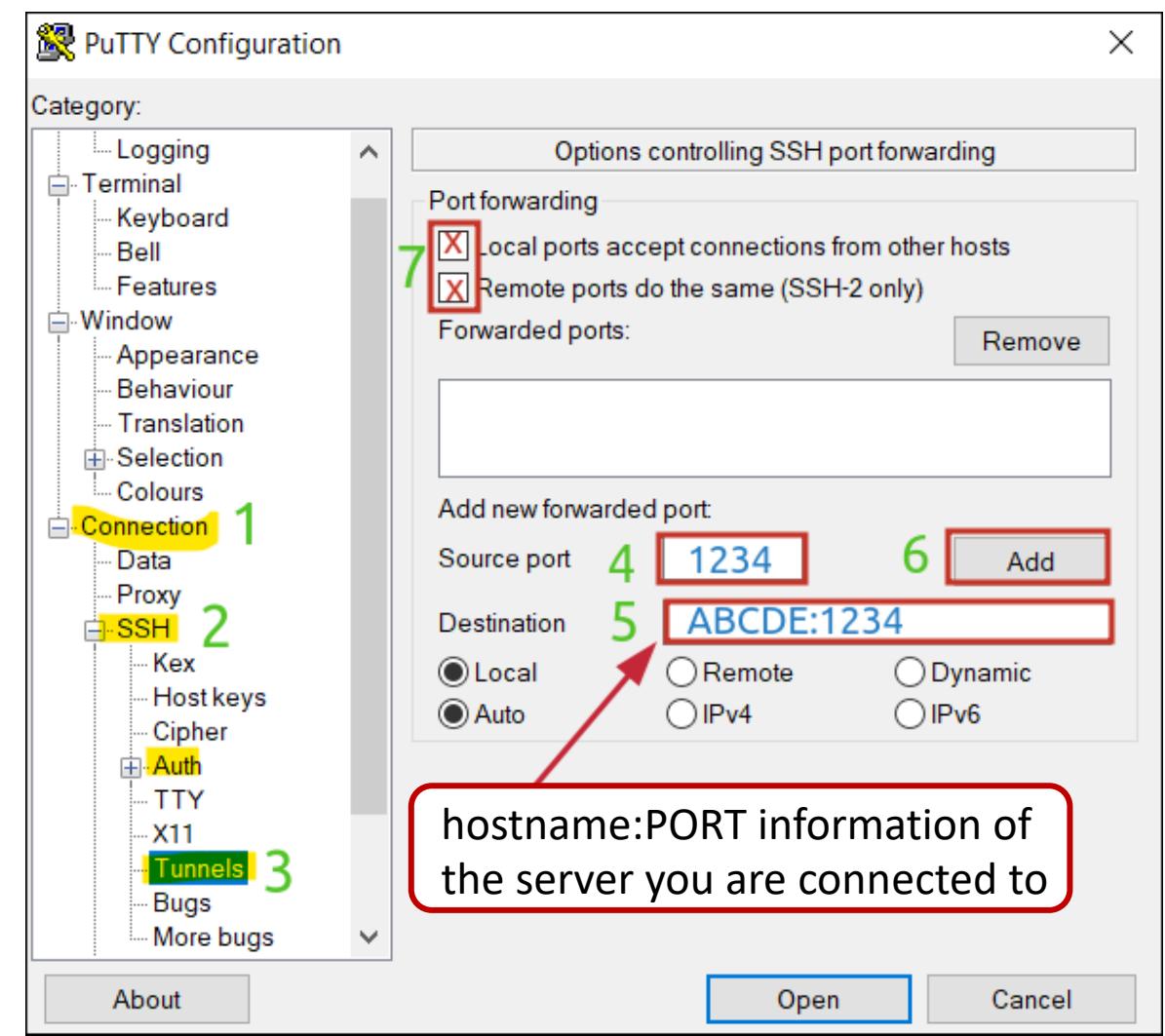
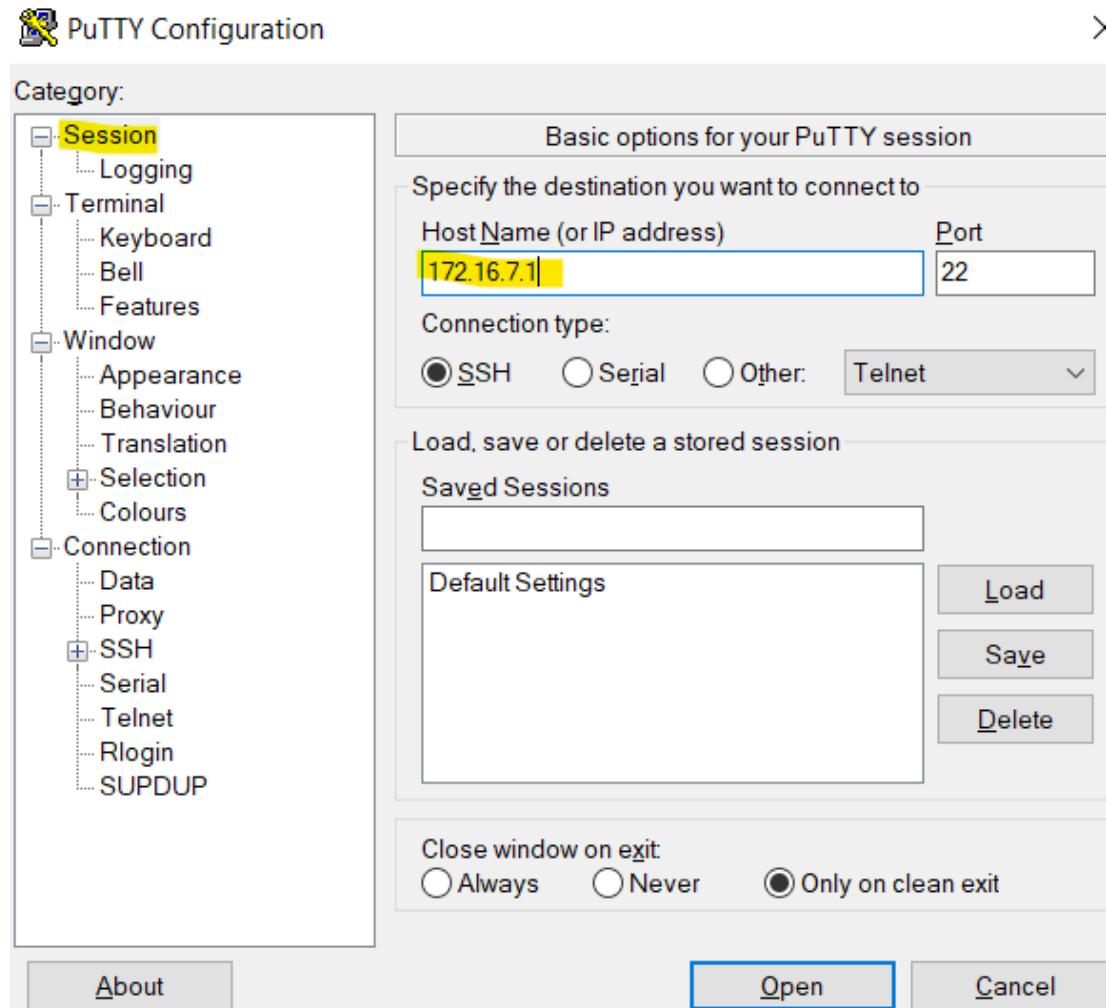
64-bit x86:	putty-64bit-0.79-installer.msi	(signature)
64-bit Arm:	putty-arm64-0.79-installer.msi	(signature)
32-bit x86:	putty-0.79-installer.msi	(signature)

Unix source archive

.tar.gz:	putty-0.79.tar.gz	(signature)
----------	-----------------------------------	-------------------------------

SSH Tunneling using PuTTY

Without closing the PuTTY you are connecting to, open a new PuTTY and enter 172.16.7.1 information instead of Host Name (or IP address).



`ssh -N -L localhost:[PORT]:[HOSTNAME]:PORT username@172.16.7.1`

Jupyter Notebook

- In general, users need to install miniconda/anaconda and create an environment on their home directory (/truba/home/username)

```
$> wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
$> bash Miniconda3-latest-Linux-x86_64.sh
$> eval "$(/truba/home/$USER/miniconda3/bin/conda shell.bash hook)"
$> conda update conda
$> conda create --name new-virtual-env          # to create new virtual env
$> conda activate new-virtual-env              # to activate conda env
$> conda list                                 # to list installed libraries on the env
$> conda install -c conda-forge jupyter
$> conda install -c conda-forge tensorflow
$> ....
```

- For this training, hands-on participants will use the installed miniconda package and customized environment.

- .bash_profile

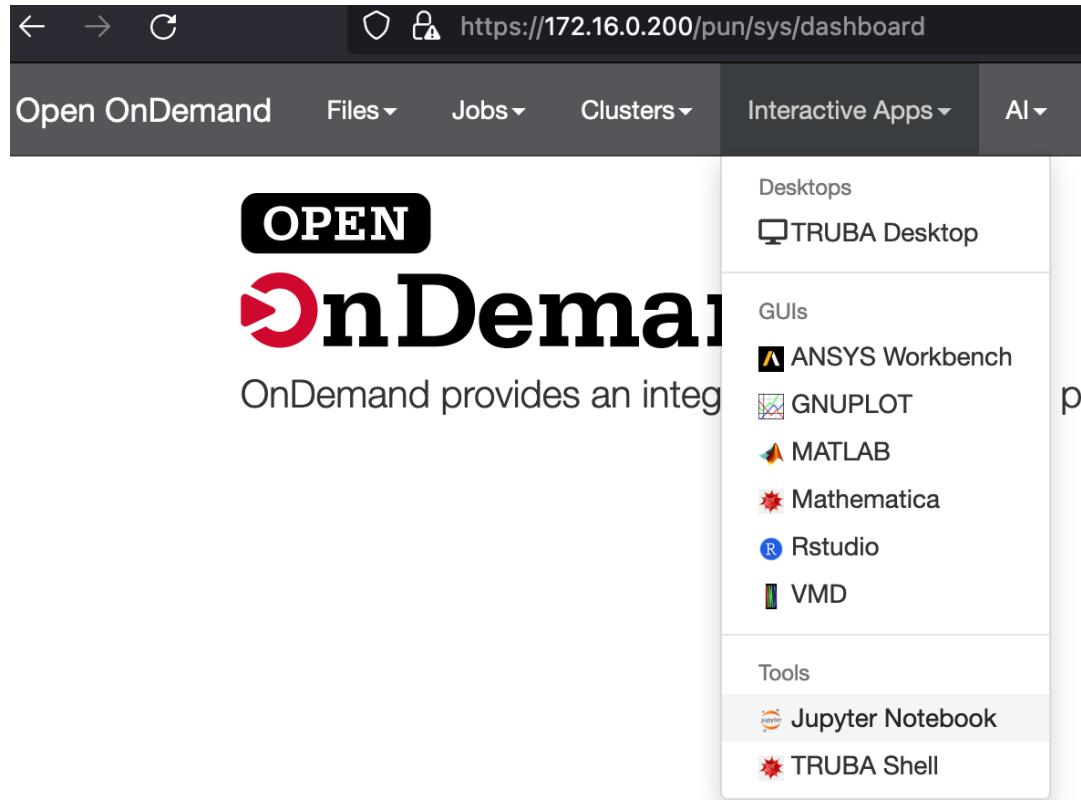
```
eval "$(/truba/home/egitim/miniconda3/bin/conda shell.bash hook)"
conda activate transformer-nlp
```

```
#####
#(transformer-nlp) -bash-4.2$
```

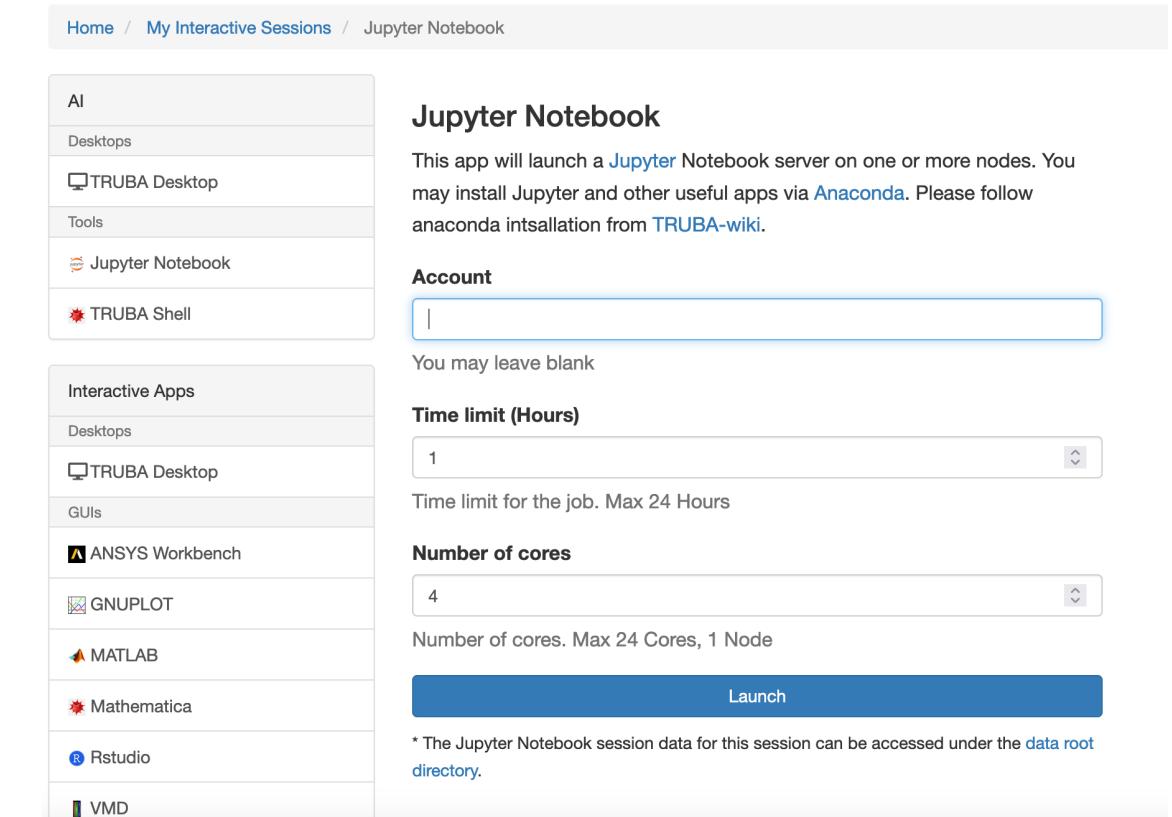


Jupyter Notebook: Open OnDemand

- Open OnDemand Interface
 - can be accessed at <https://172.16.0.200> (This service can only be accessed via OpenVPN).
 - Only CPU partitions can be requested. There is no access to GPU resources via Open OnDemand.



The screenshot shows the Open OnDemand interface with the URL <https://172.16.0.200/pun/sys/dashboard> in the browser address bar. The top navigation bar includes links for Open OnDemand, Files, Jobs, Clusters, Interactive Apps (which is currently selected), and AI. The main content area features a large "OPEN onDemand" logo with a play button icon. Below the logo, the text "OnDemand provides an integ" is partially visible. A sidebar on the right lists various interactive applications categorized into Desktops, GUIs, Tools, and AI. Under the "Interactive Apps" section, the "Jupyter Notebook" option is highlighted.



The screenshot shows the "Jupyter Notebook" configuration page. The URL in the browser is [Home / My Interactive Sessions / Jupyter Notebook](#). The page has sections for "AI", "Account", "Time limit (Hours)", and "Number of cores". The "AI" section lists Desktops, TRUBA Desktop, Tools, Jupyter Notebook, and TRUBA Shell. The "Account" section has a text input field with placeholder "You may leave blank". The "Time limit (Hours)" input field contains the value "1". The "Number of cores" input field contains the value "4". At the bottom is a large blue "Launch" button. A note at the bottom states: "* The Jupyter Notebook session data for this session can be accessed under the [data root](#) directory."

Jupyter Notebook: SLURM

Sample SLURM file to run Jupyter Notebook on akya-cuda:

- The limitations on GPU nodes
 - $10 * [\text{GPU_NUMBER}]$ for akya-cuda cluster
(4xV100s on one node)

```
#!/bin/bash
#SBATCH --account=egitim
#SBATCH --job-name=jupyter
#SBATCH --partition=akya-cuda
#SBATCH --gres=gpu:1
#SBATCH --nodes=1
#SBATCH --ntasks=10
#SBATCH --cpus-per-task=1
#SBATCH --reservation=transformer
#SBATCH --time=01:00:00          # days-hours:minutes:seconds
#SBATCH --output=slurm-%j.out
#SBATCH --error=slurm-%j.err
#SBATCH --mail-user=your_email_address
#SBATCH --mail-type=ALL

### Load modules
module purge
export OMP_NUM_THREADS=$SLURM_CPUS_PER_TASK

eval "$(truba/home/egitim/miniconda3/bin/conda shell.bash hook)"
conda activate transformer

hostname
nvidia-smi
jupyter-lab --no-browser --ip=0.0.0.0 --port=8889
scontrol show job ${SLURM_JOB_ID} >> ${SLURM_JOB_ID}.info

exit
```

Jupyter Notebook: SLURM

Sample SLURM file to run Jupyter Notebook on barbun-cuda:

- The limitations on GPU nodes:
 - $20 * [\text{GPU_NUMBER}]$ for barbun-cuda cluster
(2xP100s on one node)

```
#!/bin/bash
#SBATCH --account=egitim
#SBATCH --job-name=jupyter
#SBATCH --partition=barbun-cuda
#SBATCH --gres=gpu:1
#SBATCH --nodes=1
#SBATCH --ntasks=20
#SBATCH --cpus-per-task=1
#SBATCH --reservation=mastertransf
#SBATCH --time=01:00:00          # days-hours:minutes:seconds
#SBATCH --output=slurm-%j.out
#SBATCH --error=slurm-%j.err
#SBATCH --mail-user=your_email_address
#SBATCH --mail-type=ALL

### Load modules
module purge
export OMP_NUM_THREADS=$SLURM_CPUS_PER_TASK

eval "$(truba/home/egitim/miniconda3/bin/conda shell.bash hook)"
conda activate transformer

hostname
nvidia-smi
jupyter-lab --no-browser --ip=0.0.0.0 --port=8889
scontrol show job ${SLURM_JOB_ID} >> ${SLURM_JOB_ID}.info

exit
```

Jupyter Notebook: SLURM

- Submit the slurm file to the system to reserve a cuda node:

```
$> sbatch jupyter_akyaa-cuda.sh
```

- Check the hostname from the *out file when your work starts to run on cuda nodes
 - akya9.yonetim
- Check port number from the *err file
 - <http://akya9.yonetim:8889>
- On a new terminal type below commands and enter the password
 - ssh -N -L 8889:akya9.yonetim:8889 accountname@172.16.7.1
- Open a browser and copy the url from the *err file

[C 2023-09-10 23:15:25.791 ServerApp]

To access the server, open this file in a browser:

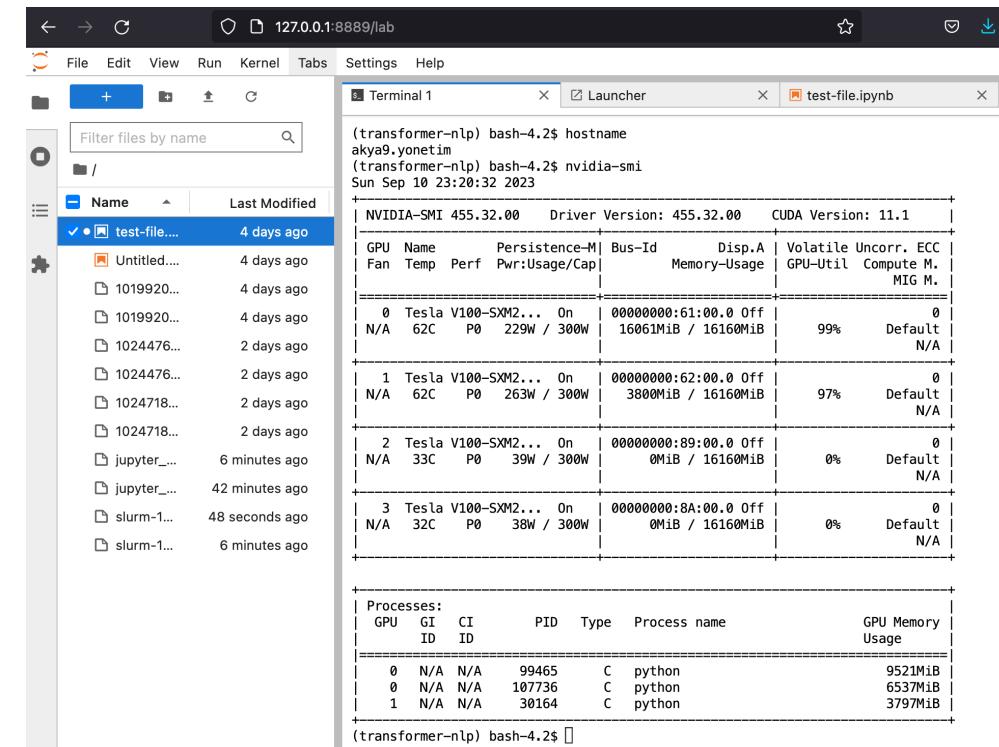
<file:///truba/home/egitim/.local/share/jupyter/runtime/jpserver-131901-open.html>

Or copy and paste one of these URLs:

<http://akya9.yonetim:8889/lab?token=e58d598aa7f1331d9ec7a8b5531b3cf28663475908efce1e>

<http://127.0.0.1:8889/lab?token=e58d598aa7f1331d9ec7a8b5531b3cf28663475908efce1e>

```
(transformer-nlp) -bash-4.2$ sbatch jupyter_akyaa-cuda.sh
Submitted batch job 1030455
(transformer-nlp) -bash-4.2$ squeue
+-----+-----+-----+-----+-----+-----+
| JOBID | PARTITION | NAME | USER | ST | TIME | NODES | NODENAME(REA|
+-----+-----+-----+-----+-----+-----+
| 1030455 | akyaa-cuda | jupyter | egitim | R | 0:01 | 1 | akyaa |
+-----+-----+-----+-----+-----+-----+
```



The screenshot shows a terminal window with the following content:

```
File Edit View Run Kernel Tabs Settings Help
127.0.0.1:8889/lab
Terminal 1
(transformer-nlp) bash-4.2$ hostname
akya9.yonetim
(transformer-nlp) bash-4.2$ nvidia-smi
Sun Sep 10 23:20:32 2023
+-----+-----+-----+-----+-----+-----+
| NVIDIA-SMI 455.32.00 | Driver Version: 455.32.00 | CUDA Version: 11.1 | | | | |
| GPU Name Persistence-M| Bus-Id Disp.A Volatile Uncorr. ECC |
| Fan Temp Perf Pwr:Usage/Cap| Memory-Usage GPU-Util Compute M. |
| | | | | | | MIG M. |
+-----+-----+-----+-----+-----+-----+
| 0 Tesla V100-SXM2... On 00000000:61:00.0 Off 16061MiB / 16160MiB | 99% Default N/A |
| | N/A 62C P0 229W / 300W | | | | |
+-----+-----+-----+-----+-----+-----+
| 1 Tesla V100-SXM2... On 00000000:62:00.0 Off 3800MiB / 16160MiB | 97% Default N/A |
| | N/A 62C P0 263W / 300W | | | | |
+-----+-----+-----+-----+-----+-----+
| 2 Tesla V100-SXM2... On 00000000:89:00.0 Off 0MiB / 16160MiB | 0% Default N/A |
| | N/A 33C P0 39W / 300W | | | | |
+-----+-----+-----+-----+-----+-----+
| 3 Tesla V100-SXM2... On 00000000:8A:00.0 Off 0MiB / 16160MiB | 0% Default N/A |
| | N/A 32C P0 38W / 300W | | | | |
+-----+-----+-----+-----+-----+-----+
Processes:
| GPU GI CI PID Type Process name | GPU Memory Usage |
| ID ID | |
+-----+-----+-----+-----+-----+-----+
| 0 N/A N/A 99465 C python | 9521MiB |
| 0 N/A N/A 107736 C python | 6537MiB |
| 1 N/A N/A 30164 C python | 3797MiB |
+-----+-----+-----+-----+-----+
```

The terminal also shows the command history and the current working directory (127.0.0.1:8889/lab).

Python Jobs Using SLURM

- Converted *ipynb file to *py file:

```
import tensorflow as tf
print(tf.__version__)
print(tf.config.list_physical_devices('GPU'))
```

- Sample SLURM file

```
#!/bin/bash
#SBATCH -p akya-cuda
#SBATCH -A [USERNAME]
#SBATCH -J print_gpu
#SBATCH -o print_gpu.out
#SBATCH --gres=gpu:1
#SBATCH -N 1
#SBATCH -n 1
#SBATCH --cpus-per-task=10
#SBATCH --time=1:00:00

eval "$(truba/home/$USER/miniconda3/bin/conda shell.bash hook)"
conda activate dl-env
python print_gpu.py

exit
```

Reserved Nodes

- 8 nodes from akya-cuda (20 x 2CPU and & 4xNvidia V100 GPU)

```
#SBATCH --reservation=transformer
```

(SLURM file: /truba/home/egitimXX/slurm-scripts/jupyter_akya-cuda.sh)

- 9 nodes from barbun-cuda (20 x 2CPU and & 2xNvidia P100 GPU)

```
#SBATCH --reservation= mastertransf
```

(SLURM file: /truba/home/egitimXX/slurm-scripts/jupyter_barbun-cuda.sh)

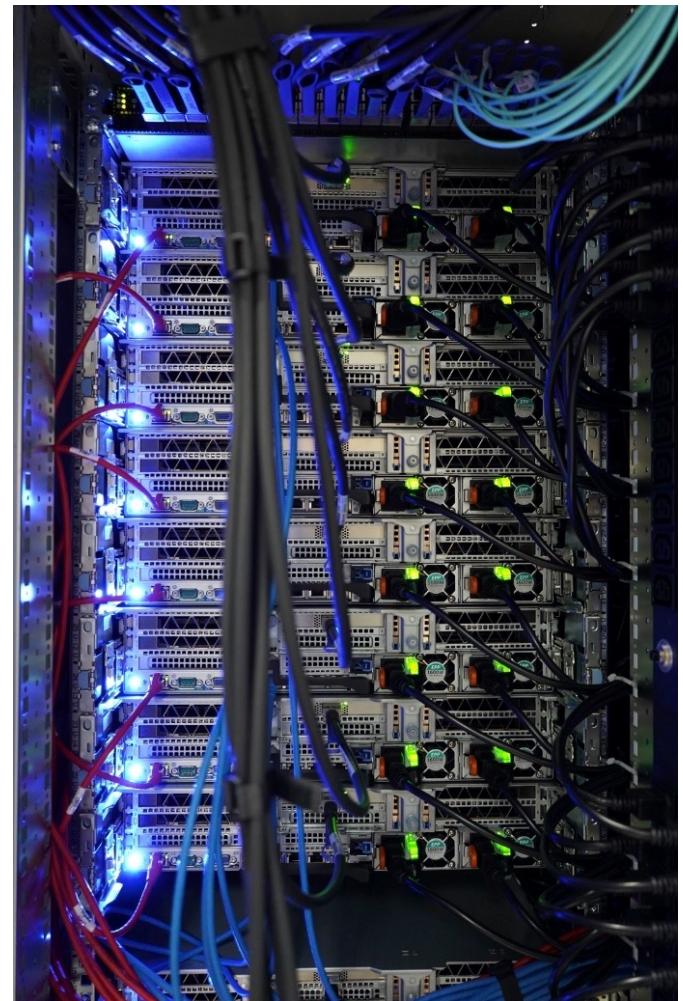


 **Mastering Transformers: From Building Blocks to Real-World Applications**

September 11-13, 2023

Online



User Support

grid-teknik@ulakbim.gov.tr

Thank You!



TRUBA HPC Center

<http://www.truba.gov.tr>

<http://docs.truba.gov.tr>

<http://portal.truba.gov.tr>



<https://twitter.com/TrubaUlakbim>



<https://www.linkedin.com/company/truba>