HELMHOLTZ

MODEL ZOO



Hans Werners, Jennifer Ahrens,

Engin Eren, Philipp Heuser

HELMHOLTZ IMAGING

HAICORE
Helmholtz AI computing resources

Helmholtz Model Zoo

A Model Zoo is a collection of (pre)trained deep learning models, ready to use for inference.

These can include:

- specifically trained and fine-tuned models for defined tasks and data,
- domain and data type agnostic generalised models
- foundation models
- Your models!



HELMHOLTZ MODEL ZOO

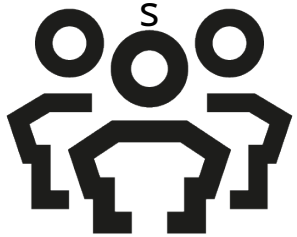
Domain Scientists

?



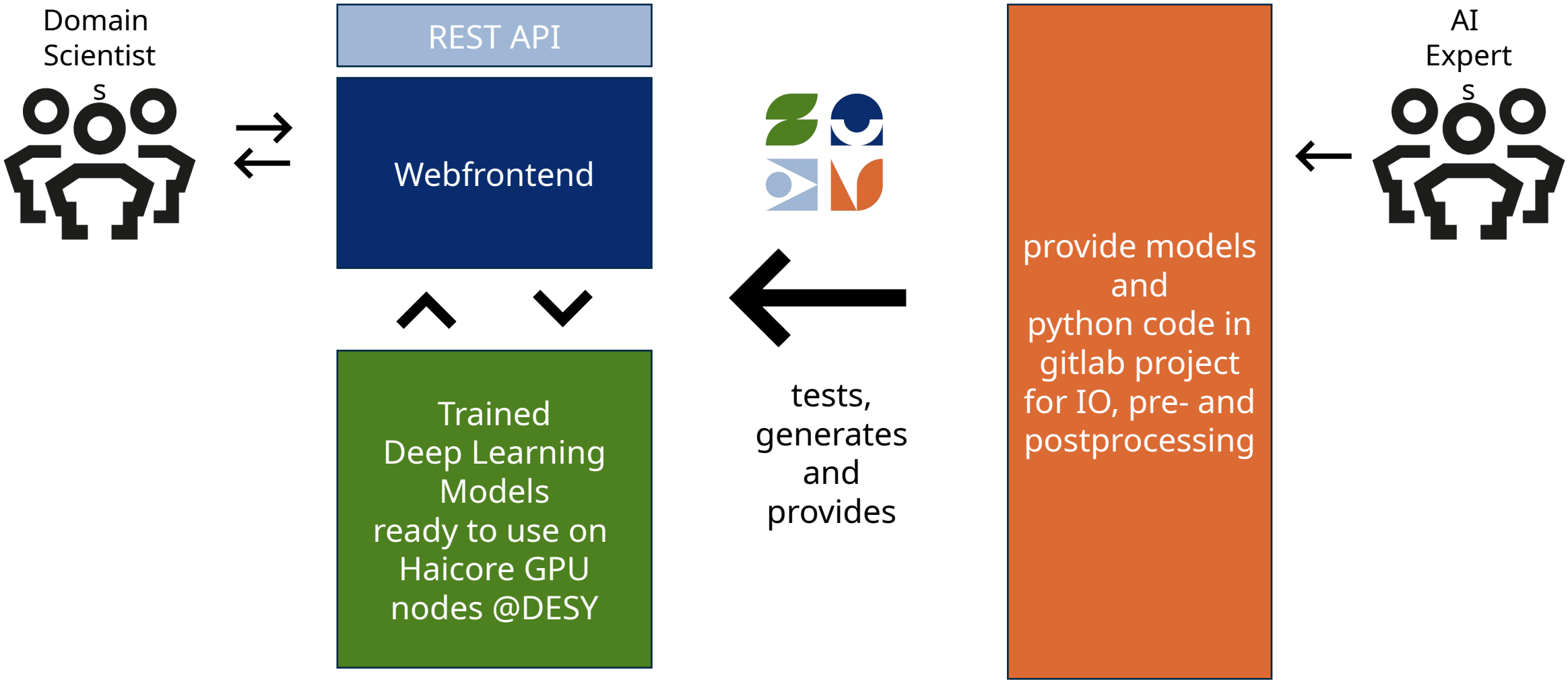
mamba slurm clone dependencies docker permission denied pip **cluster** pull versions install **GPU** deploy

provide models in git

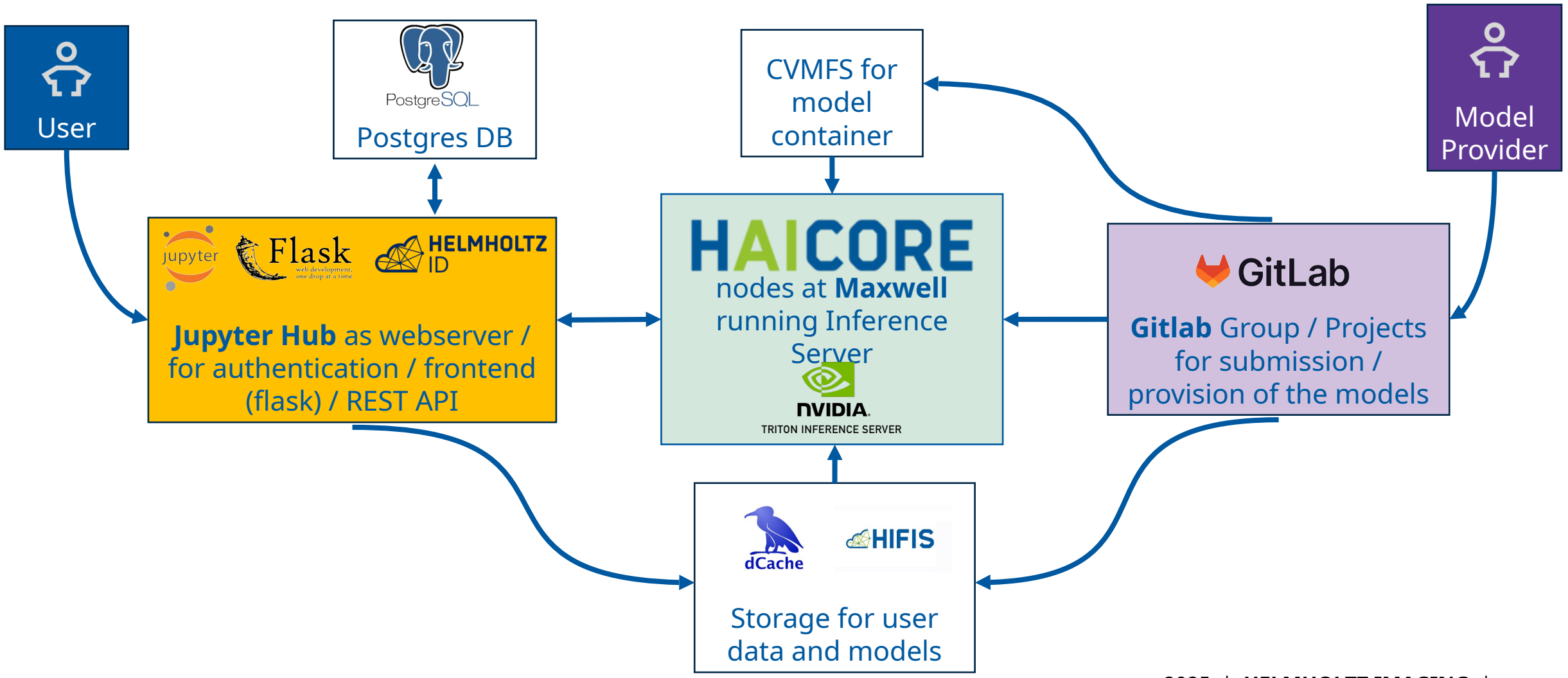


AI Experts

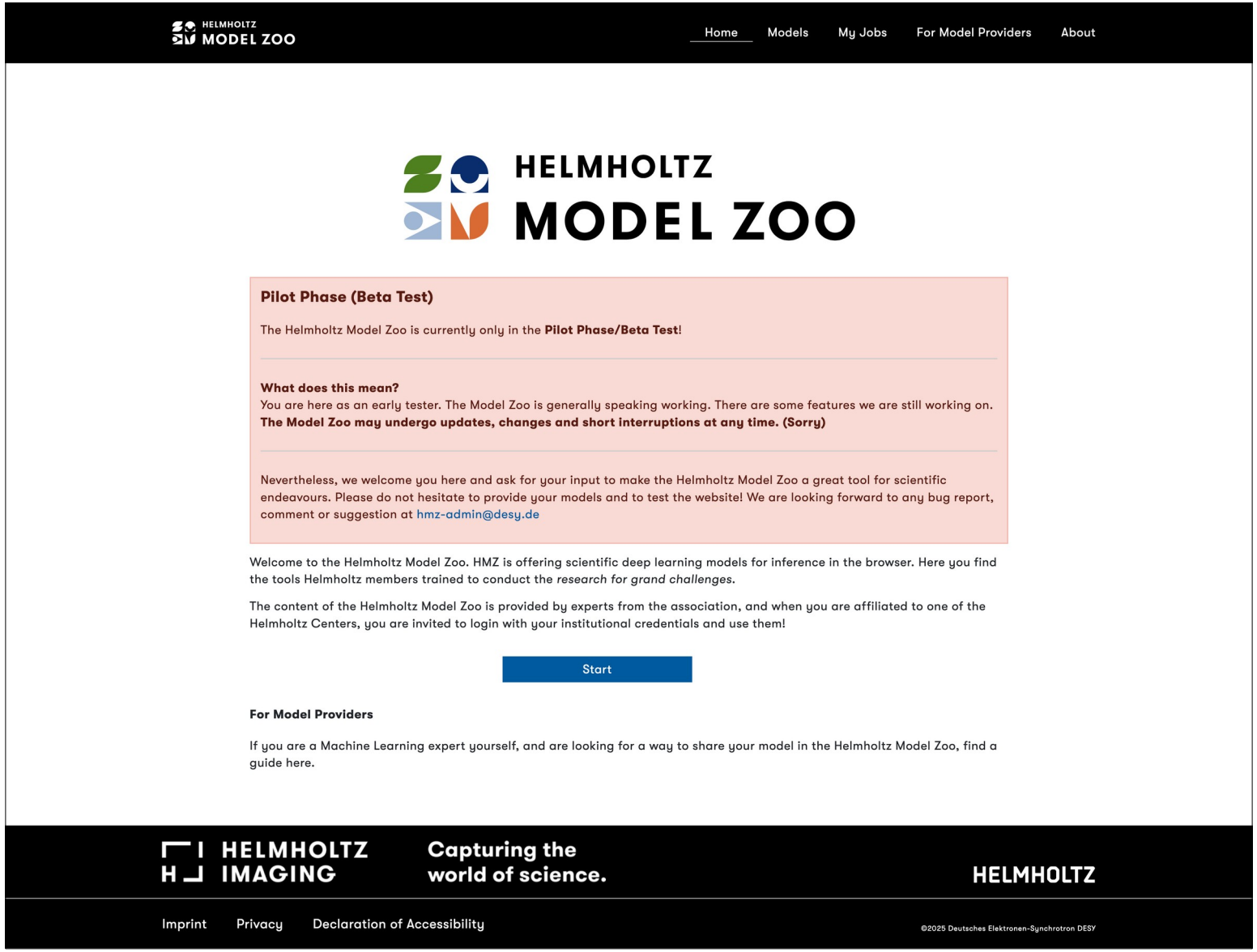
HELMHOLTZ MODEL ZOO



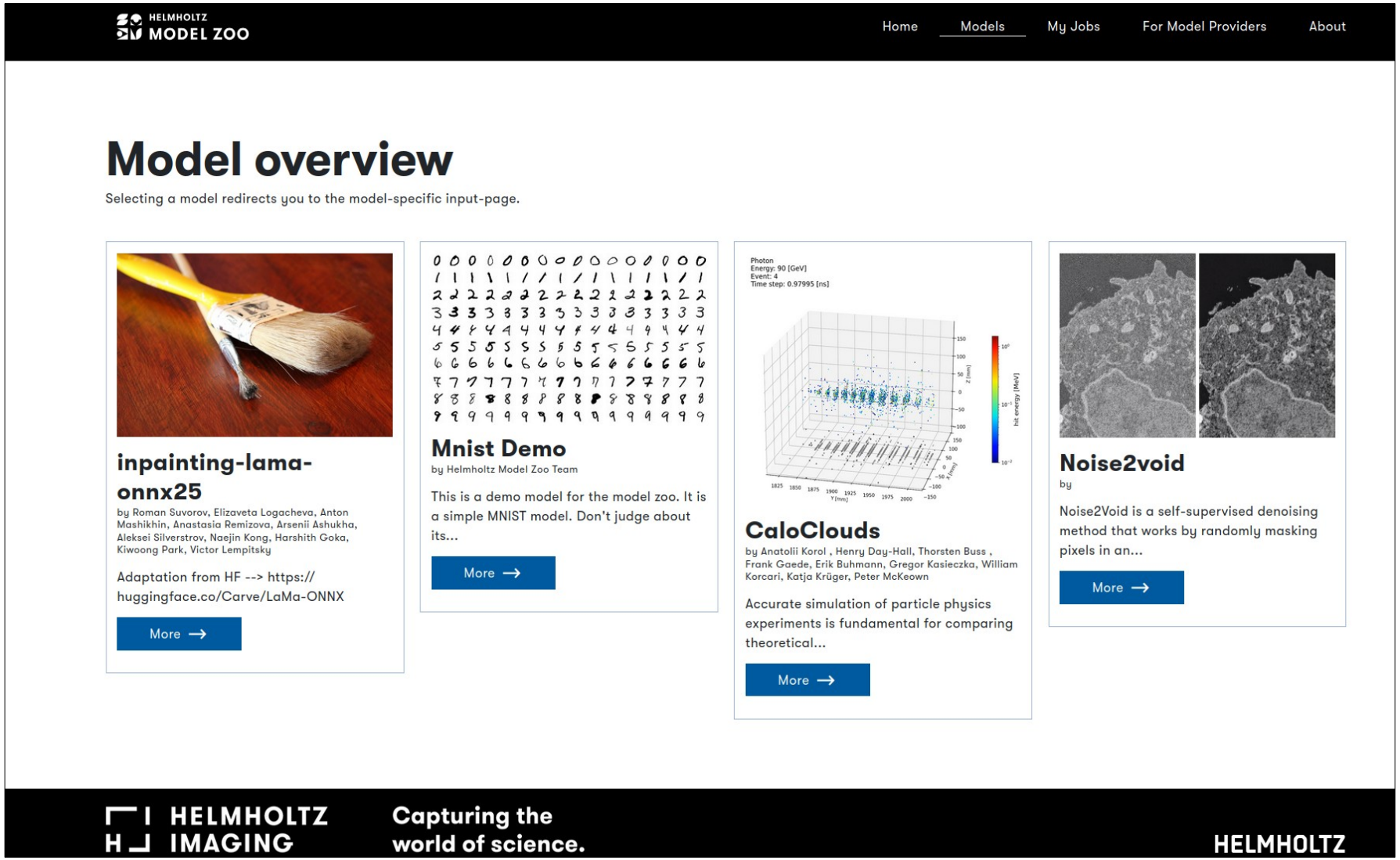
Helmholtz Model Zoo



Model Zoo for the User



Model Zoo for the User



Model Zoo for the User

HELMHOLTZ MODEL ZOO

Home Models My Jobs For Model Providers About

inpainting-lama-onnx25

by Roman Suvorov , Elizaveta Logacheva , Anton Mashikhin , Anastasia Remizova , Arsenii Ashukha , Aleksei Silverstrov , Naejin Kong , Harshith Goka , Kiwoong Park , Victor Lempitsky

Source repository of model
Publication:
Primary contact: hmz_admin@desy.de
License: Apache license 2.0
Tags: HMZ, HMZ Demo, inpainting



How to use inpainting-lama-onnx25 with REST_API?

See here.

Model Description:

Adaptation from HF --> https://huggingface.co/Carve/LaMa-ONNX

This is a Fourier Convolutions, CNN etc.. model, trained on Images.

Data Input

This model expects as input:

Field Name	Help Text	Data Type expected
raw_image	Upload an image to be inpainted. Both needs to have the same name with the mask having '_mask' at the end. E.g. 'image.jpg' and 'image_mask.jpg'	image 2D (jpg, png)
mask	Upload the relative mask to the image to be inpainted. Both needs to have the same name with the mask having '_mask' at the end. E.g. 'image.jpg' and 'image_mask.jpg'	image 2D (jpg, png)

Result

As result you can expect:

Result Folder Name	Result Description	Result Filetype
output	This is inpainted image	image 2D (jpg , png)

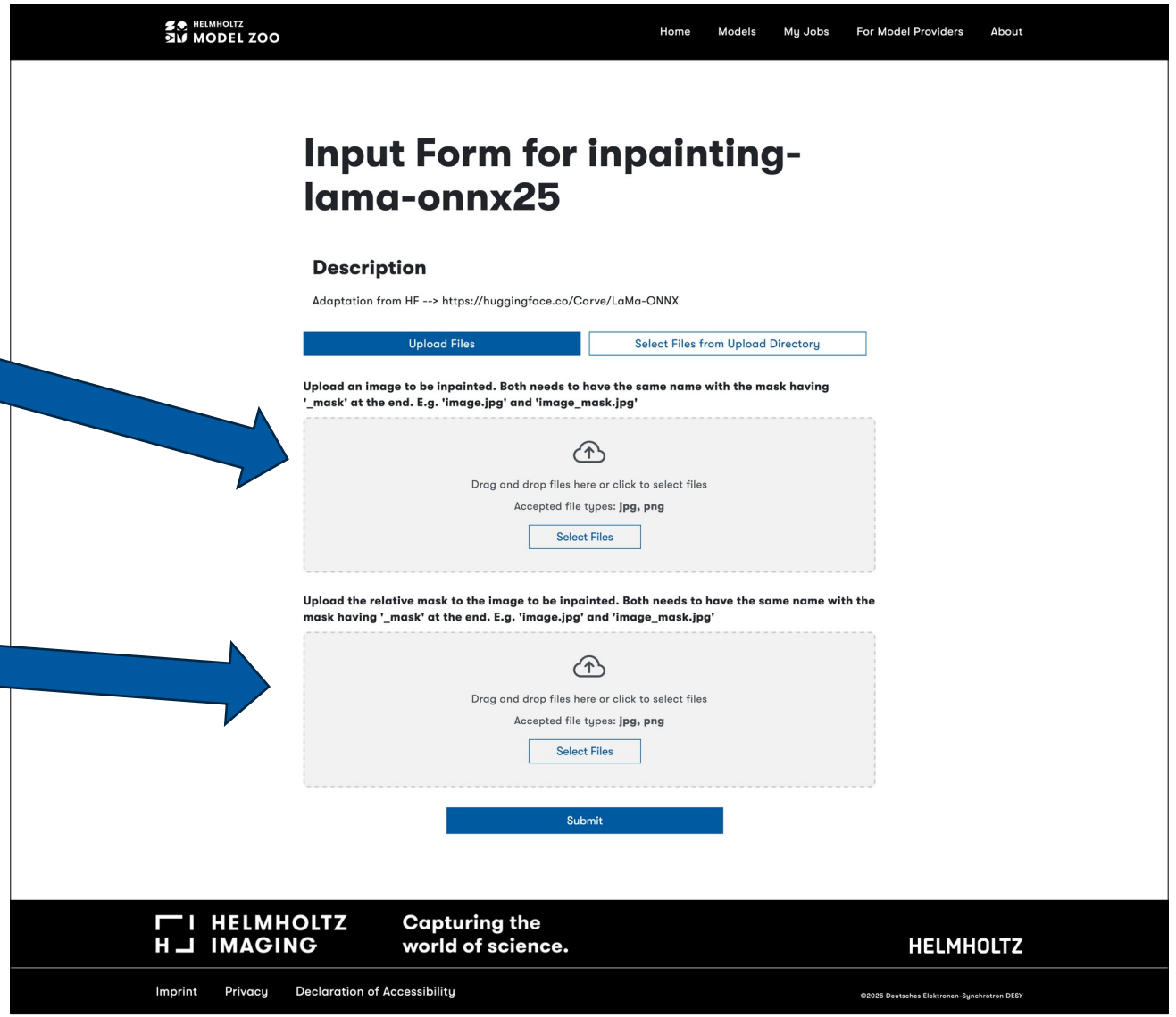
When using this model, please cite:

Use Model →

Model Zoo for the User







Model Zoo for the User

HELMHOLTZ MODEL ZOO

Home Models My Jobs For Model Providers About

Job inpainting-lama-onnx25

Status

Model: inpainting-lama-onnx25
Input: /pnfs/desy.de/hifis-storage/HMZ/user//pheuser/upload/dataset_inpainting-lama-onnx25_2025-06-24_11-58-57

Status: COMPLETED
Start: 2025-06-24 11:58:58
End: 2025-06-24 11:59:04

Results / Download

Results

Download all result files as one zip:

output 1 files

inpainting-lama-onnx25_prediction_dayi.jpg View Download

Logs

Here you can quickly scan some of the log files for information.

status_log slurm_out slurm_error

```
> This is the content of the logfile:
> /pnfs/desy.de/hifis-storage/HMZ/user/pheuser/result/result_inpainting-lama-onnx25_70790f15/logs/status_log.log
-----------
Done
```

Please cite:



HELMHOLTZ IMAGING Capturing the world of science. HELMHOLTZ

Imprint Privacy Declaration of Accessibility

©2025 Deutsches Elektronen-Synchrotron DESY

Model Zoo for the User

Input Form for CaloClouds

Description

Accurate simulation of particle physics experiments is fundamental for comparing theoretical predictions with experimental results. However, with planned high-luminosity upgrades to facilities like the Large Hadron Collider (LHC) and envisioned experiments such as those at the International Linear Collider (ILC), the volume of experimental data is rapidly increasing. Traditional Monte Carlo simulations, like Geant4, struggle to keep pace with these rates due to their computational intensity, posing a significant challenge to projected computing budgets. Specifically, detector simulations, especially for highly granular calorimeters (which can have millions of sensor cells), are computationally expensive. Prior efforts using generative machine learning (ML) models to accelerate these simulations often relied on fixed, regular detector readout geometries, representing showers as 3D images. This approach is inefficient given the sparsity of energy depositions in modern calorimeters. Building upon the original CaloClouds model, which generated calorimeter showers as geometry-independent point clouds, CaloClouds II offers substantial speed improvements and comparable fidelity to traditional Geant4 simulations. Key innovations include the adoption of continuous-time score-based modeling for faster sampling and, most notably, the first application of consistency distillation to calorimeter shower generation, enabling single-step sampling with a 46x speed-up over Geant4 on a single CPU. This work addresses the increasing computational demands of particle physics simulations, paving the way for more efficient and scalable detector simulations.

energy

50.00

Incoming energy of the particle

eta

60.00

the incidence angle of in the incoming particle

phi

0.00

Azimuthal angle

nshowers

100.00

Number of showers

Submit

Model Zoo for the User

HELMHOLTZ MODEL ZOO

Home | Models | My Jobs | For Model Providers | About

Job CaloClouds

Status

Model: caloclouds
Input: /pnfs/desy.de/hifis-storage/HMZ/user//eren/upload/dataset_caloclouds_2025-07-03_09-46-11

Status: COMPLETED
Start: 2025-07-03 09:46:13
End: 2025-07-03 09:49:08

Results / Download

Results

Download all result files as one zip:

output_file 1 files

caloclouds_prediction_input.root View Download

Logs

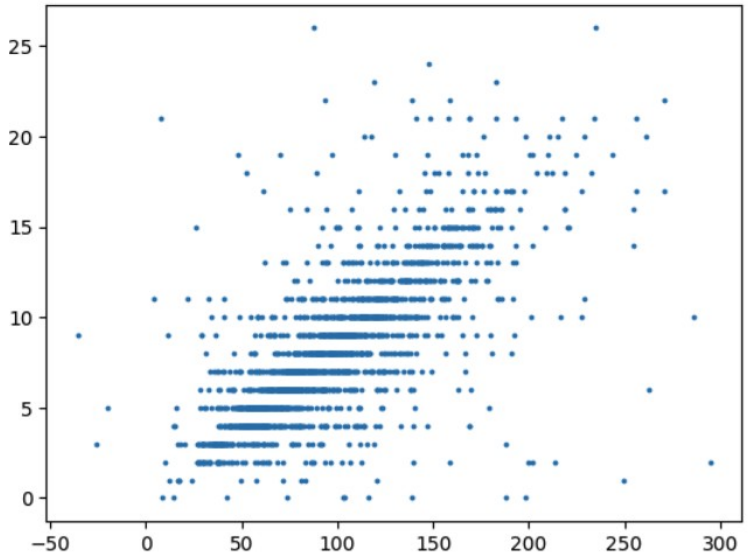
Here you can quickly scan some of the log files for information.

status_log | slurm_out | slurm_error

```
> This is the content of the logfile:
> /pnfs/desy.de/hifis-storage/HMZ/user/eren/result/result_caloclouds_c293fd2a/logs/status_log.log
-----------
Done
```

Please cite:
https://arxiv.org/pdf/2309.05704

```
[1]: import uproot

     file = uproot.open("caloclouds_prediction_input.root")

[2]: print(file.keys())

     ['Events;1']

[22]: import matplotlib.pyplot as plt
      plt.scatter(x,z,s=3)

[22]: <matplotlib.collections.PathCollection at 0x148de86817d0>
```



FAQ user

Who can use the models?

Only colleagues from Helmholtz can use it (and members of a Helmholtz ID VO)

How can I run an inference?

- Use the webinterface.
- Use a REST API

What is about my data (user)?

- You upload the data to a folder owned by you in dcache via the webinterface.
- You own your data, and it does not leave Helmholtz.
- You can also use the documented ways to upload data to dcache via rclone, webdav, etc. (recommended for large datasets)
- You can upload a reasonable amount of data (probably not PB, but a few TB could be fine).
- We will delete all your data after 3 month, or so? (Not entirely defined yet)

For model providers

Who can provide models?

All scientists in Helmholtz can login via the Helmholtz ID and provide their models.

What kind of models? ONNX PyTorch TensorFlow OpenVINO

Which hardware?

Our nodes are provided by HAICORE, run at DESY, each equipped with 4 Nvidia L40S. HAICORE Helmholtz AI computing resources

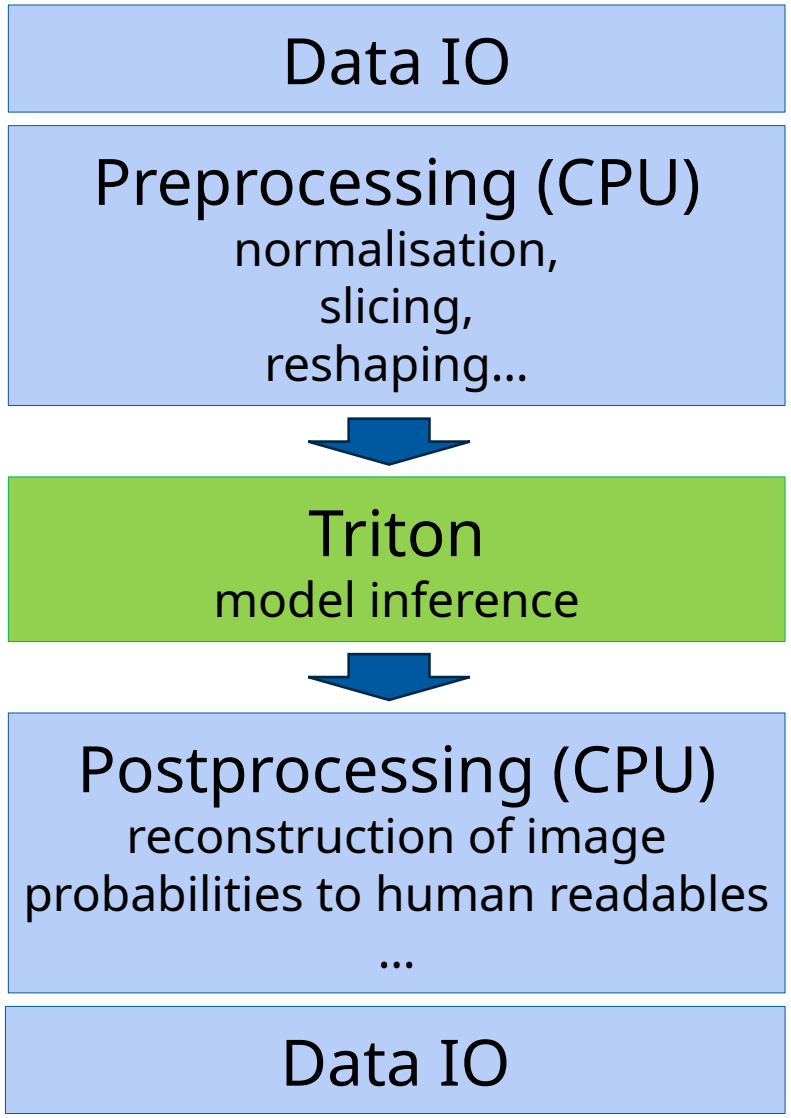
Can I make my model “private”?

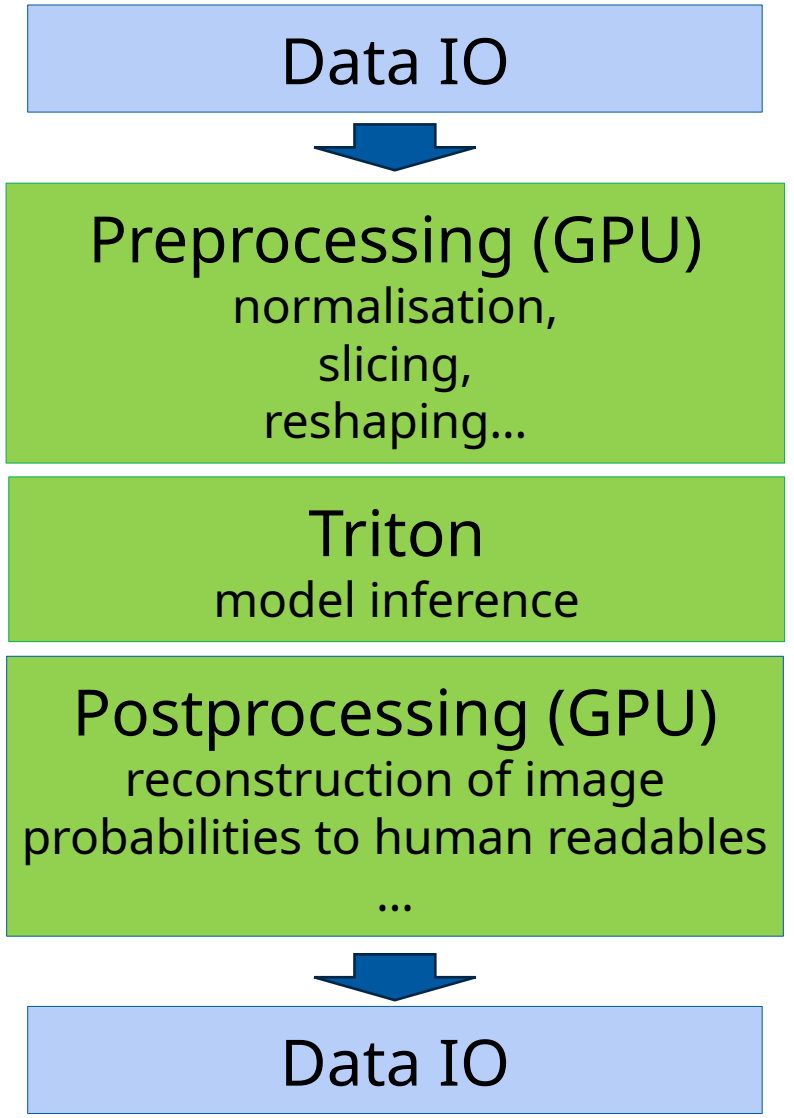
Yes, you will be able to give the usernames of people who are entitled to use your model? (Later via a Helmholtz ID VO)

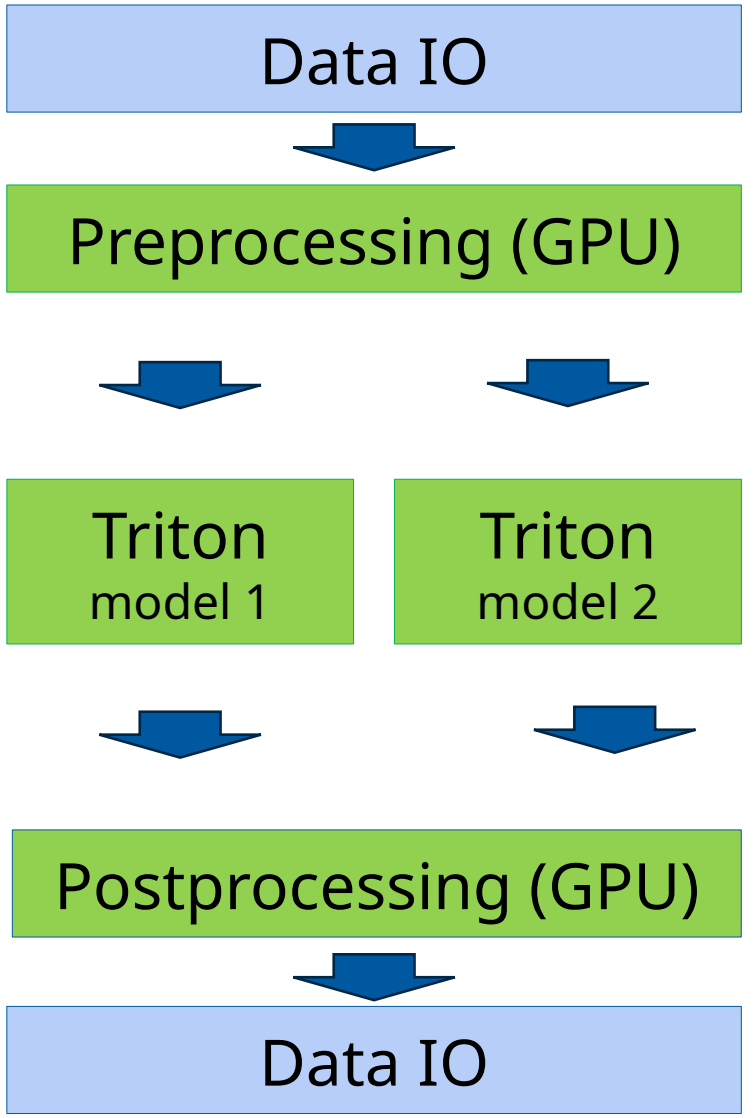
What do I have to do?

- We provide you with a template in the (DESY)-Gitlab.
- We try to get as much as possible of your shoulders, keeping a maximum of flexibility
- You have to add your model and implement the necessary for pre- and postprocessing
- You have to fill in a form for the necessary meta data (some technical details, description, publication)

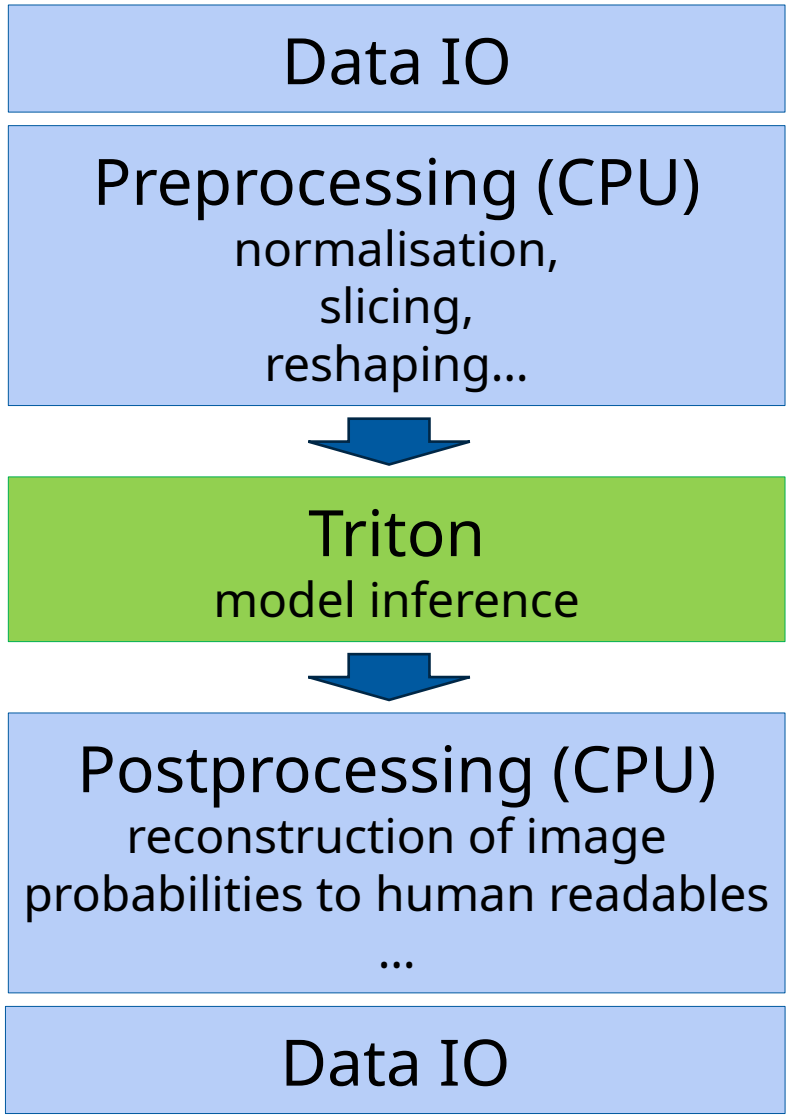
Pre/Post Processing & Triton

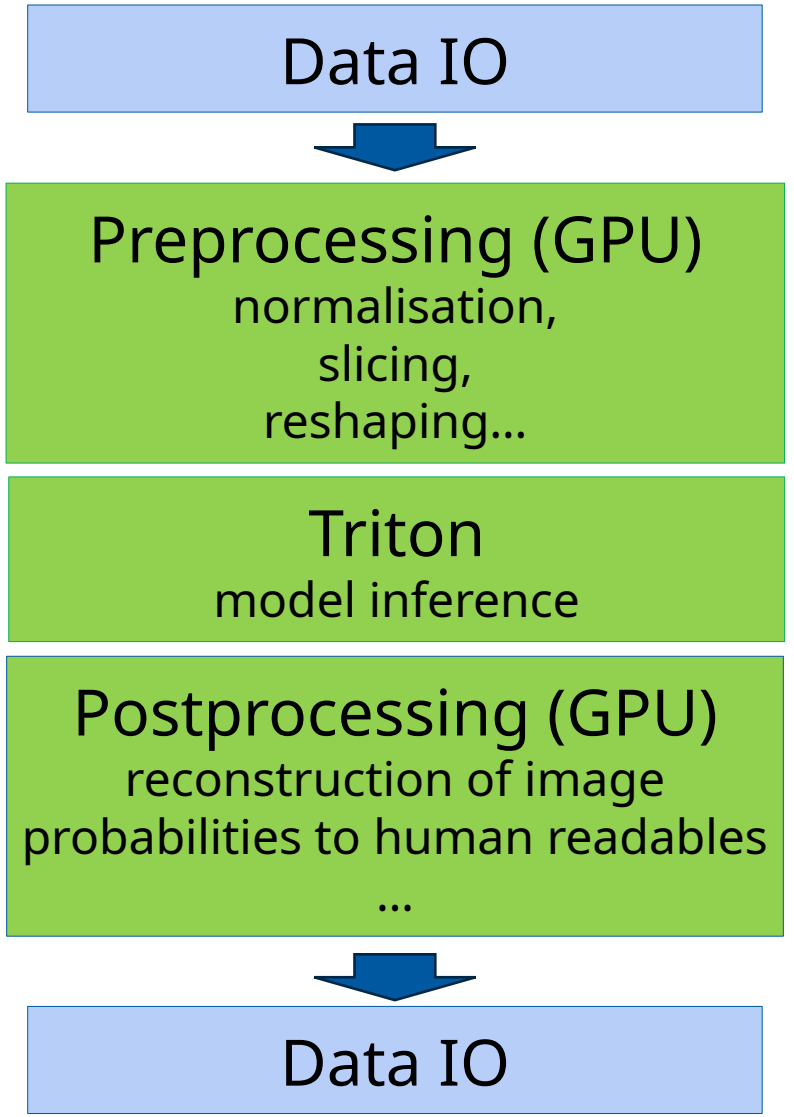


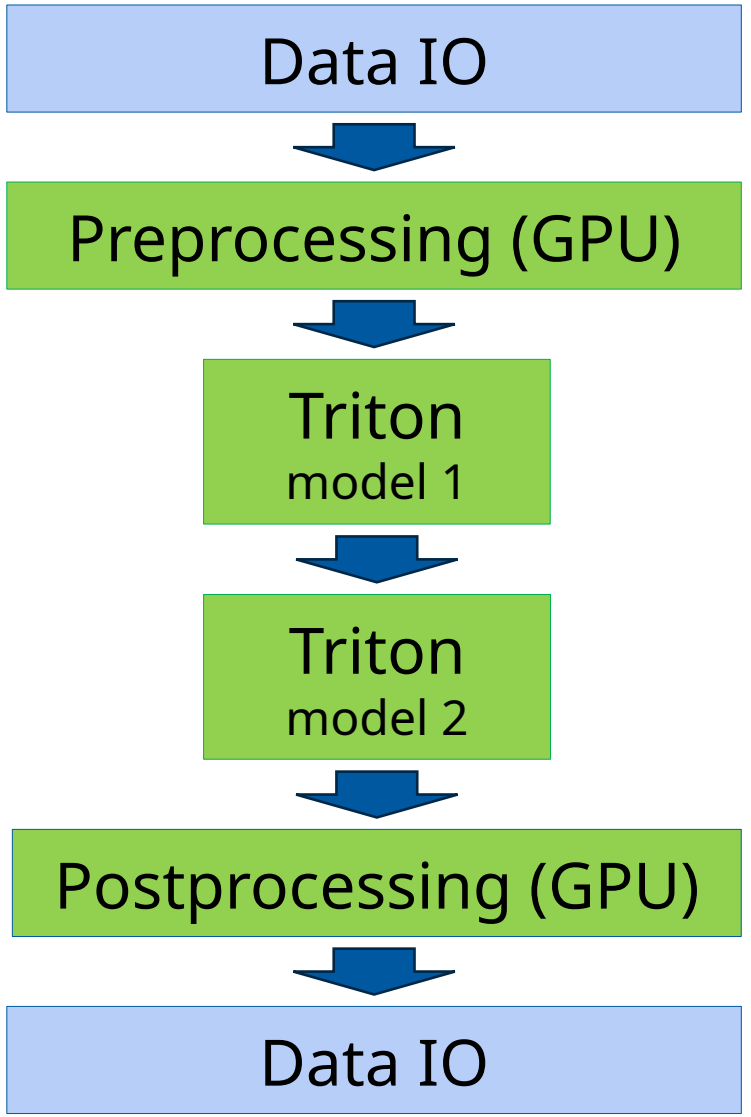




Pre/Post Processing & Triton







Model Files & config.pbtxt

```python
# Convert to ONNX
onnx_path = f"{output_dir}/model.onnx"
example_input = torch.randn(1, 3, 224, 224)

torch.onnx.export(
    model,
    example_input,
    onnx_path,
    export_params=True,
    opset_version=11,
    do_constant_folding=True,
    input_names=['input'],
    output_names=['output']
)
```

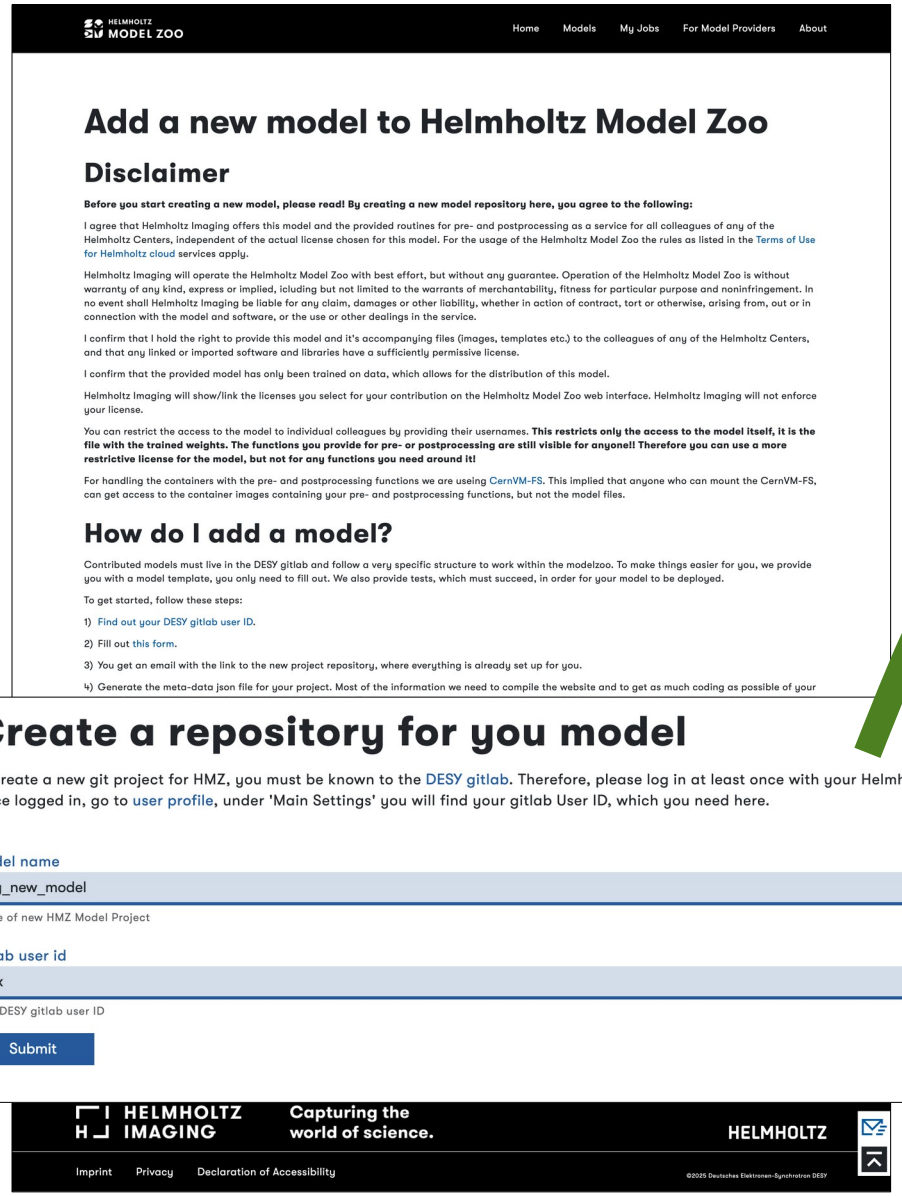
model > particleseg3d > particleseg3d_fold1 > config.pbtxt

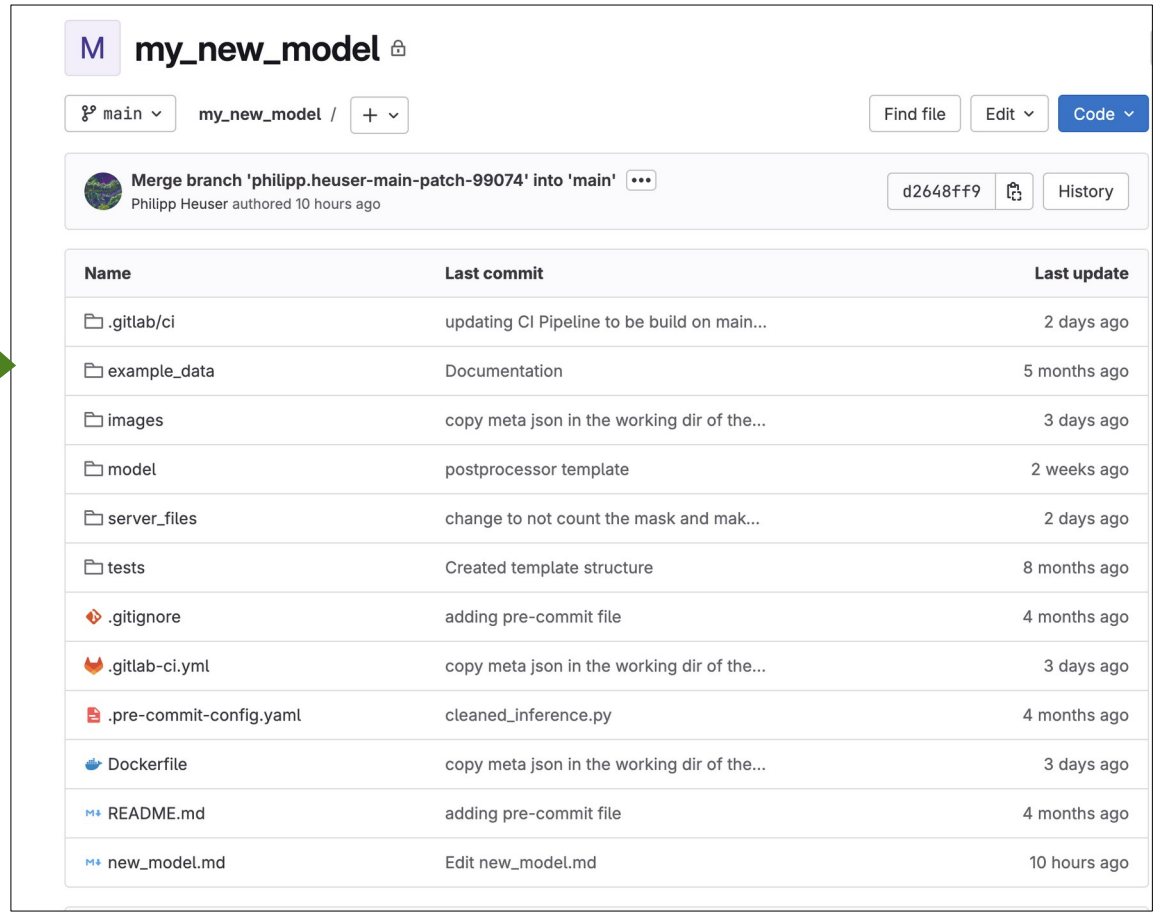
```
name: "particleseg3d_fold1"
backend: "onnxruntime"
max_batch_size: 8

input [
  {
    name: "input"
    data_type: TYPE_FP32
    dims: [ 1, 80, 192, 160 ]
  }
]

output [
  {
    name: "output"
    data_type: TYPE_FP32
    dims: [ 3, 80, 192, 160 ]
  }
]

dynamic_batching {
  preferred_batch_size: [ 1, 2, 4 ]
  max_queue_delay_microseconds: 100
}
```

model > particleseg3d > particleseg3d_ensemble > config.pbtxt

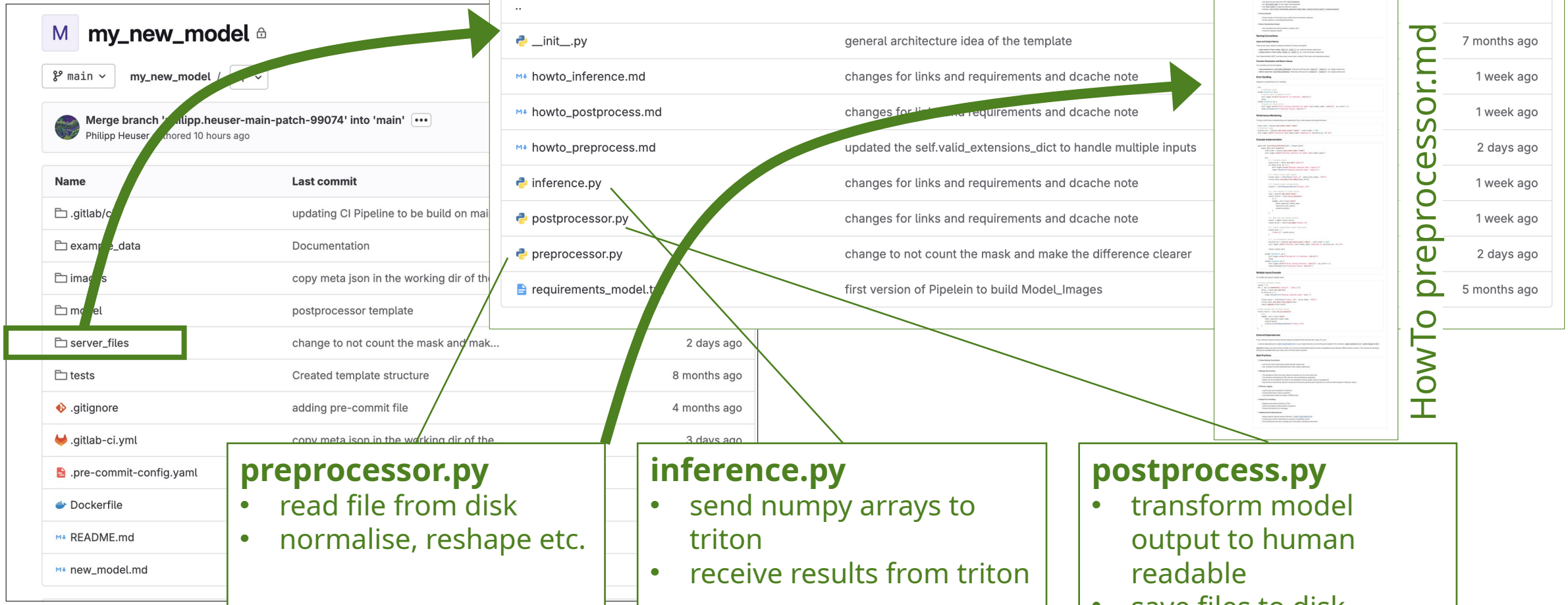
```
name: "particleseg3d_ensemble"
platform: "ensemble"
max_batch_size: 8

input [
  {
    name: "input"
    data_type: TYPE_FP32
    dims: [ 1, 80, 192, 160 ]
  }
]

output [
  {
    name: "output"
    data_type: TYPE_FP32
    dims: [ 3, 80, 192, 160 ]
  }
]

ensemble_scheduling {
  step [
    {
      model_name: "particleseg3d_fold0"
      model_version: -1
      input_map {
        key: "input"
        value: "input"
      }
      output_map {
        key: "output"
        value: "fold0_logits"
      }
    },
    {
      model_name: "particleseg3d_fold1"
      model_version: -1
      input_map {
        key: "input"
        value: "input"
      }
      output_map {
        key: "output"
        value: "fold1_logits"
      }
```

Model Zoo for the Model Provider





Model Zoo Template_0



preprocessor.py

- read file from disk
- normalise, reshape etc.

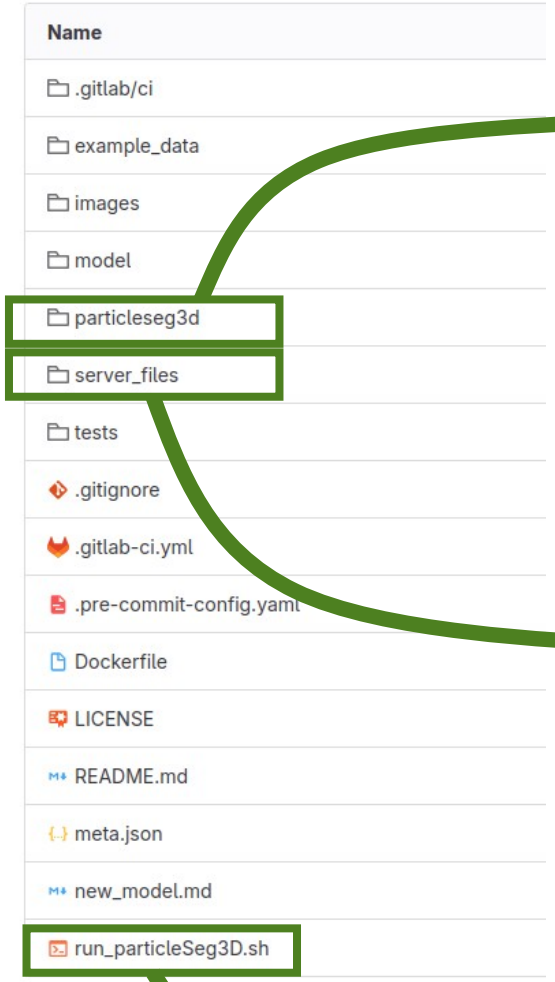
inference.py

- send numpy arrays to triton
- receive results from triton

postprocess.py

- transform model output to human readable
- save files to disk

Model Zoo Template_1(to be deployed)

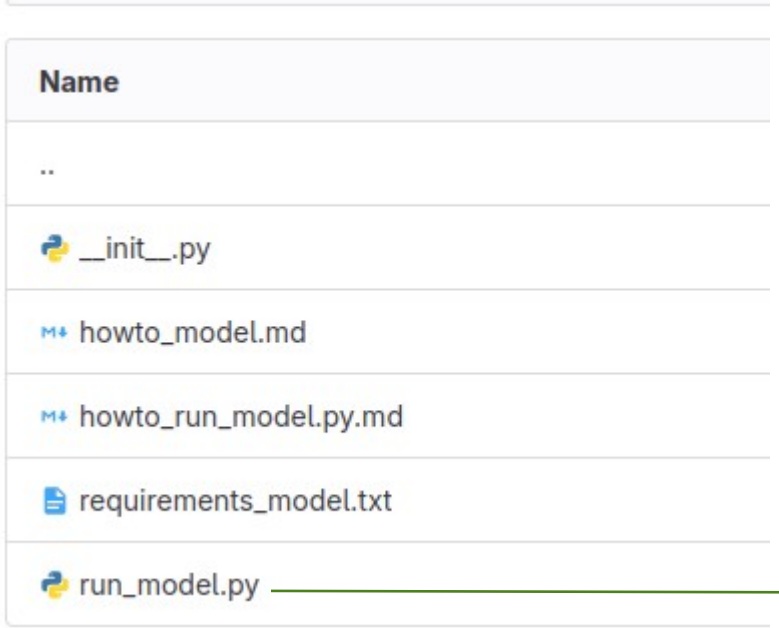


‘old’ inference call

```
model.prediction_setup(aggregator, chunked, zscore)
trainer.predict(model, dataloaders=sampler)
```

Triton inference call

```
# Call Triton inference
response = inference.client.infer(
    model_name=inference.model_name,
    inputs=inputs,
    outputs=outputs
)
```



run_model.py

- Entry point for our backend
- Starts the model entry point using configurable options (shell scripts, etc.)
- Inherits needed attributes

run_particleSeg3D.sh

- Entrypoint to model

HMZ!

Pilot Phase (Beta Test)

The Helmholtz Model Zoo is currently only in the **Pilot Phase/Beta Test**!

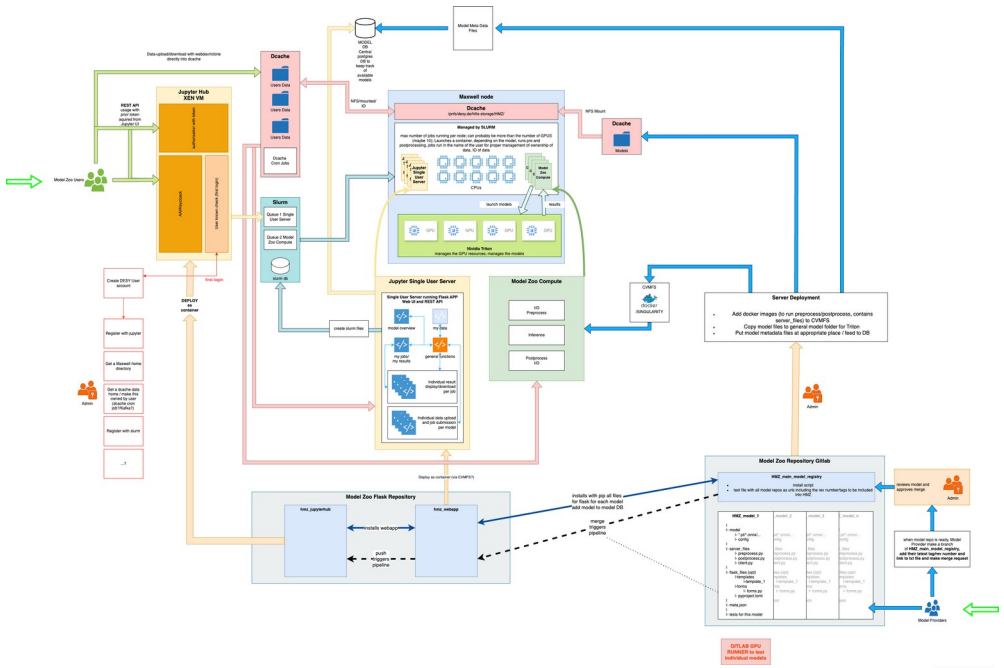
What does this mean?
You are here as an early tester. The Model Zoo is generally speaking working. There are some features we are still working on.
The Model Zoo may undergo updates, changes and short interruptions at any time. (Sorry)

Nevertheless, we welcome you here and ask for your input to make the Helmholtz Model Zoo a great tool for scientific endeavours. Please do not hesitate to provide your models and to test the website! We are looking forward to any bug report, comment or suggestion at hmz-admin@desy.de

Preliminary URL:
https://hmz-hub.desy.de

Get in touch:
support@helmholtz-imaging.de
& talk to us here

Helmholtz Model Zoo



HAICORE HIFIS

HELMHOLTZ IMAGING

Thanks to:

Christian Voss
Elena Gapon
Frank Schlünzen
Johannes Reppin
Kars Ohrenberg
Lusine Yakovleva
Moritz Hamminger
Patrick Fuhrmann
Sven Sternberger
Thomas Hartmann
Tim Wetzel
Uwe Jandt
Yves Kemp
Alexander Trautsch
Stefan Bujack

Talk to us!



HELMHOLTZ MODEL ZOO

support@helmholtz-imaging.de