# EDGE AI Academy

## Summer School

**7-8 July 2025, Pisa, Italy**

# Edge AI Academy Summer School

## AI/ML Inference Engine Software for High Reliability Applications on Space Qualifiable Hardware

KLEPSYDRA
TECHNOLOGIES

Dr Pablo Ghiglino, Klepsydra Technologies, Switzerland

7-8 July 2025 Pisa, Italy

# Online connection details to the raspberry pi devices

**Requirements:**
- Laptop with internet connection
- Terminal with SSH command

**Public IP: 193.200.46.78**
Devices internal network: 192.168.100.0/24

User: kpsr
Pass: klepsydra123

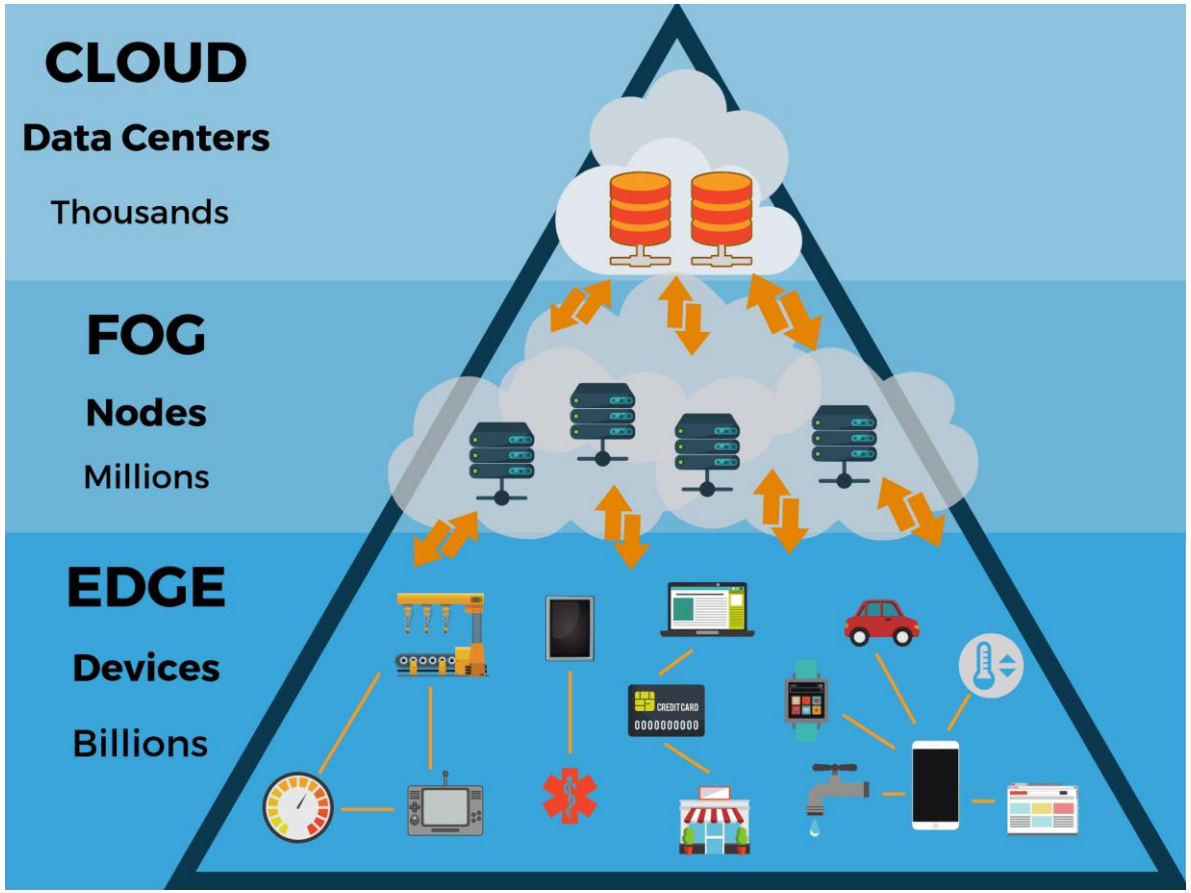
**We will be using the following tools:**
- docker
- linux
- cmake
- make
- vim

| Device | Internal IP | Forwarded port |
|---|---|---|
| rpi3-0 | 192.168.100.30 | 2230 |
| rpi3-1 | 192.168.100.31 | 2231 |
| ~~rpi3-2~~ | ~~192.168.100.32~~ | ~~2232~~ |
| rpi4-0 | 192.168.100.40 | 2240 |
| rpi4-1 | 192.168.100.41 | 2241 |
| ~~rpi4-2~~ | ~~192.168.100.42~~ | ~~2242~~ |
| rpi5-0 | 192.168.100.50 | 2250 |
| rpi5-1 | 192.168.100.51 | 2251 |
| ~~rpi5-2~~ | ~~192.168.100.52~~ | ~~2252~~ |

Example: to connect to rpi4-1: **`ssh -p 2241 kpsr@193.200.46.78`**

Copyright © 2025

# What is edge and what is an AI Inference Engine



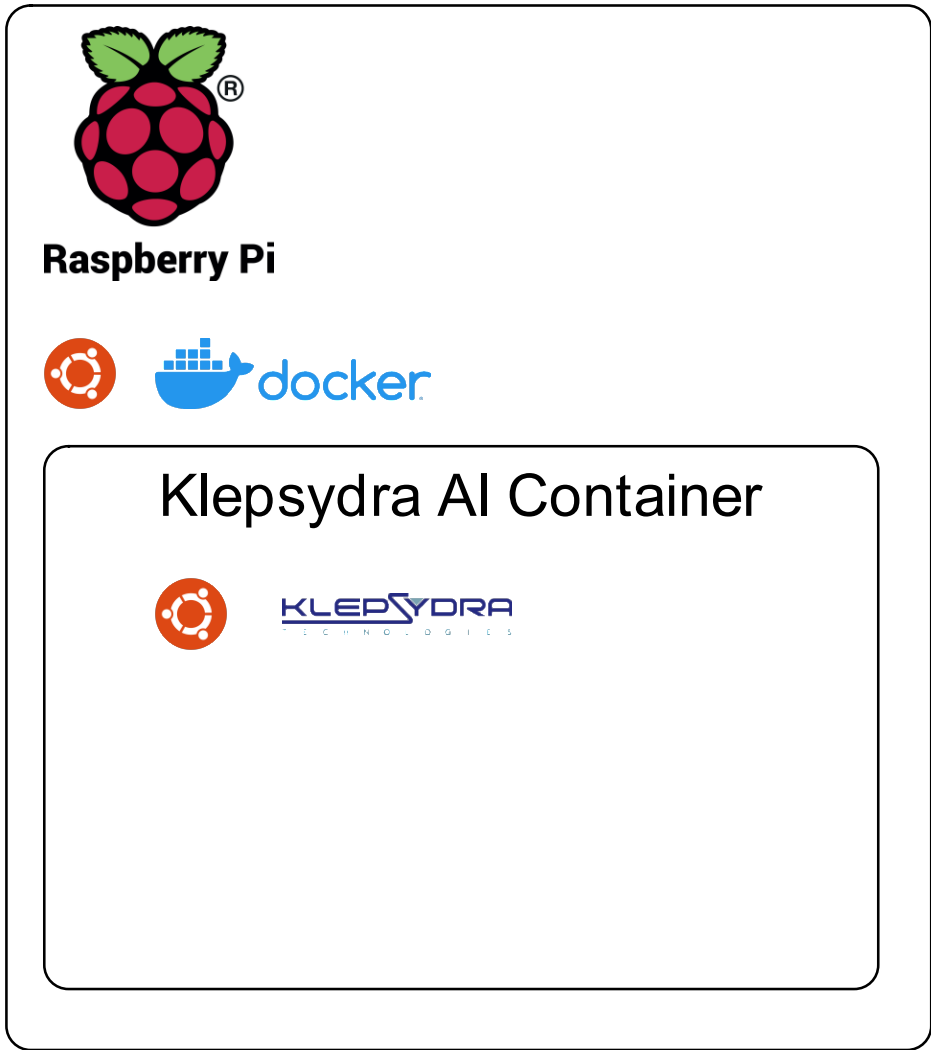
Copyright © 2025

# Contents of the board

## Device software setup

Each raspberry PI contains:

- Ubuntu 20.04 as operating system
- Docker container with Klepsydra AI installed
- License to run Klepsydra AI
- Examples




Copyright © 2025

# 1st step: enter the docker container

highlighted in yellow the commands to execute

kpsr@rpi3-0:~ $ docker images
Expected output:

| REPOSITORY | TAG | IMAGE ID | CREATED | SIZE |
|---|---|---|---|---|
| registry.klepsydra.io/rpi3/kpe-ai-dev | 18.3.0 | **1765c7de288f** | 7 days ago | 1.43GB |

kpsr@rpi3-0:~ $ docker run -itd **1765c7de288f** bash

kpsr@rpi3-0:~ $ docker ps
Expected output:

| CONTAINER ID | IMAGE | COMMAND | CREATED | STATUS | PORTS | NAMES |
|---|---|---|---|---|---|---|
| **8022d097740c** | 1765c7de288f | "bash" | 3 seconds ago | Up 1 second | | adoring_aryabhata |

kpsr@rpi3-0:~ $ docker exec -it **8022d097740c** bash
Expected output:
root@8022d097740c:/#

Copyright © 2025

# 2nd step: check that the binaries are there

highlighted in yellow the commands to execute

root@8022d097740c:/# ls /usr/local/lib/libk*ai*.so.*
Expected output:
/usr/local/lib//libkpsr_ai_cpu_backend.so.18
/usr/local/lib//libkpsr_ai_cpu_backend.so.18.3.0
/usr/local/lib//libkpsr_ai_full_backend.so.18
/usr/local/lib//libkpsr_ai_full_backend.so.18.3.0
/usr/local/lib//libkpsr_ai_interface.so.18
/usr/local/lib//libkpsr_ai_interface.so.18.3.0
**/usr/local/lib//libkpsr_ai_ocv.so.18**
**/usr/local/lib//libkpsr_ai_ocv.so.18.3.0**
**/usr/local/lib//libkpsr_ai_onnx.so.18**
**/usr/local/lib//libkpsr_ai_onnx.so.18.3.0**
/usr/local/lib//libkpsr_ai_serialization.so.18
/usr/local/lib//libkpsr_ai_serialization.so.18.3.0
/usr/local/lib//libkpsr_ai.so.18
/usr/local/lib//libkpsr_ai.so.18.3.0

Not included in the Space product: While these components function similarly to the Space version, they are excluded from this workshop.

Copyright © 2025

# 3rd step: compile the minimal example

highlighted in yellow the commands to execute

```
root@8022d097740c:/# cd /root/
root@8022d097740c:/# cp -r /usr/local/src/kpe-ai-examples .
root@8022d097740c:~# ls
Expected output:
kpe-ai-examples
root@8022d097740c:~# cd kpe-ai-examples/ai-min-example/
root@8022d097740c:~/kpe-ai-examples/ai-min-example# mkdir build
root@8022d097740c:~/kpe-ai-examples/ai-min-example# cd build/
root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# cmake ..
Expected output:
CMake Deprecation Warning at CMakeLists.txt:20 (cmake_minimum_required):
  Compatibility with CMake < 3.5 will be removed from a future version of
  CMake.

  Update the VERSION argument <min> value or use a ...<max> suffix to tell
  CMake that the project does not need compatibility with older versions.

-- The CXX compiler identification is GNU 9.4.0
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Check for working CXX compiler: /usr/bin/c++ - skipped
-- Detecting CXX compile features
-- Detecting CXX compile features - done
-- Found OpenCV: /usr/local (found version "3.4.17")
-- Configuring done (2.4s)
-- Generating done (0.0s)
-- Build files have been written to: /root/kpe-ai-examples/ai-min-example/build

root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# make
Expected output:
[ 33%] Building CXX object CMakeFiles/ai-min-example.dir/src/main.cpp.o
[ 66%] Building CXX object CMakeFiles/ai-min-example.dir/src/input_parser.cpp.o
[100%] Linking CXX executable ai-min-example
[100%] Built target ai-min-example
```

Copyright © 2025

# 4th step: run the minimal example (part I)

highlighted in yellow the commands to execute

root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# ./ai-min-example
Expected output:
Usage: ai-min-example [OPTIONS]
  -f <json configuration file> : default null
  -m <kpsr model file> : default null
  -n <number of predictions> : default 100
  -t <interval in millisecs> : default 1000

oot@8022d097740c:~/kpe-ai-examples/ai-min-example/build# ./ai-min-example -f ../conf/mem_dnn_conf.json -m ../models/alexnet.kpsr
Expected output:
inputData.configurationFile: ../conf/mem_dnn_conf.json
inputData.modelFile: ../models/alexnet.kpsr
[Mon Mar 24 12:37:43 2025] [12:37:43 264848] [klepsydra] [error] [188] License Error: license file not found
[Mon Mar 24 12:37:43 2025] [12:37:43 265240] [klepsydra] [error] [188] Please provide the path to the license file in
[Mon Mar 24 12:37:43 2025] [12:37:43 265306] [klepsydra] [error] [188] the KPSR_LICENSE_FILE environment variable.
[Mon Mar 24 12:37:43 2025] [12:37:43 265398] [klepsydra] [error] [188] If the environment variable is not provided, klepsydra.lic will be used.
[Mon Mar 24 12:37:43 2025] [12:37:43 266934] [klepsydra] [error] [188] ----------------------------------------
[Mon Mar 24 12:37:43 2025] [12:37:43 267035] [klepsydra] [error] [188] **PC Signature:**
[Mon Mar 24 12:37:43 2025] [12:37:43 267070] [klepsydra] [error] [188] ----------------------------------------

Copyright © 2025

# 4th step: run the minimal example (part II)

highlighted in yellow the commands to execute

```
root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# export IDENTIFICATION_STRATEGY=0
root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# ./ai-min-example -f ../conf/mem_dnn_conf.json -m ../models/alexnet.kpsr
Expected output:
inputData.configurationFile: ../conf/mem_dnn_conf.json
inputData.modelFile: ../models/alexnet.kpsr
[Mon Mar 24 12:43:00 2025] [12:43:00 063712] [klepsydra] [error] [206] License Error: license file not found
[Mon Mar 24 12:43:00 2025] [12:43:00 064020] [klepsydra] [error] [206] Please provide the path to the license file in
[Mon Mar 24 12:43:00 2025] [12:43:00 064055] [klepsydra] [error] [206] the KPSR_LICENSE_FILE environment variable.
[Mon Mar 24 12:43:00 2025] [12:43:00 064101] [klepsydra] [error] [206] If the environment variable is not provided, klepsydra.lic will be used.
[Mon Mar 24 12:43:00 2025] [12:43:00 065274] [klepsydra] [error] [206] ----------------------------------------
[Mon Mar 24 12:43:00 2025] [12:43:00 065377] [klepsydra] [error] [206] PC Signature: QAAC-QqwR-AAI=
[Mon Mar 24 12:43:00 2025] [12:43:00 065410] [klepsydra] [error] [206] ----------------------------------------
root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# exit

kpsr@rpi3-0:~ $ cat klepsydra.lic
Expected output:
[KLEPSYDRA]
lic_ver = 200
client-signature = QAAC-QqwR-AAI=
valid-from = 2025-01-01
valid-to = 2025-12-31
sig =
jnWoHofnMfVc5O1Bn6lQYRIerkuC1ZMhbEHykRMTOKWh3CRr2jUlHUGUmmtJ2eFzGIJX0e7or55d5i3aDqjaHI0gSVj9O4G4SmOqSJzlUtk2n/TDIO+kCUfKyti2Hlv0u5fPNk7wVB3JvigivRUpck
YwonMrsPHTCCi7421EsBg=
```

Copyright © 2025

# 4th step: run the minimal example (part III)

highlighted in yellow the commands to execute

```
kpsr@rpi3-0:~ $ docker cp klepsydra.lic 8022d097740c:/root/
kpsr@rpi3-0:~ $ docker exec -it 8022d097740c bash
root@8022d097740c:/# cd /root
root@8022d097740c:~# cd kpe-ai-examples/ai-min-example/build/
root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# export IDENTIFICATION_STRATEGY=0
root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# export KPSR_LICENSE_FILE=/root/klepsydra.lic
root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# ./ai-min-example -f ../conf/mem_dnn_conf.json -m ../models/alexnet.kpsr
Expected output:
inputData.configurationFile: ../conf/mem_dnn_conf.json
inputData.modelFile: ../models/alexnet.kpsr
[Mon Mar 24 12:48:47 2025] [12:48:47 386549] [klepsydra] [info] [223] Layers to be optimised out: StatefulPartitionedCall/functional_1/re_lu/Relu
[Mon Mar 24 12:48:47 2025] [12:48:47 386746] [klepsydra] [info] [223] Removing StatefulPartitionedCall/functional_1/re_lu_4/Relu
.........
It took 187ms to load the model.
Applying AI to dummy image 0 of 100
Image : 0 assigned id : 0
 Use this id in callback to match input to result.
Got result for input with id: 0
Processed 0 of 100 in 106ms.
Applying AI to dummy image 1 of 100
Image : 1 assigned id : 1
 Use this id in callback to match input to result.
Got result for input with id: 1
Processed 1 of 100 in 99ms.
```

Copyright © 2025

# 4th step: run the minimal example (part IV)

KLEPSYDRA TECHNOLOGIES

## Very simple API (it is the same for all supported processors):

```
#include <klepsydra/ai/dnn_importer.h>

int main(int argc, char *argv[])
{
   ...
   std::vector<std::shared_ptr<kpsr::ai::Backend>> extensionBackends(0);
   std::unique_ptr<kpsr::ai::DeepNeuralNetworkFactory> factory =
      kpsr::ai::DNNImporter::createDNNFactory(extensionBackends,
                                              inputData.modelFile,
                                              inputData.configurationFile);
   std::shared_ptr<kpsr::ai::DeepNeuralNetwork> kpsrNetwork = factory->getDNN();
   ...
   auto saveToFileCallback = [&](const unsigned long &id, const std::vector<float> &result) {
      ...
   };
   kpsrNetwork->setCallback(saveToFileCallback);
   ...
   for (int i = 0; i < inputData.numberOfPredictions; i++) {
      ...
      auto id = kpsrNetwork->predict(inputImage);
      ...
   }
   ...
}
```

Copyright © 2025

# 5th step: Statistics collection (part I)

highlighted in yellow the commands to execute

root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# apt update
Expected output:

root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# apt install vim
Expected output:

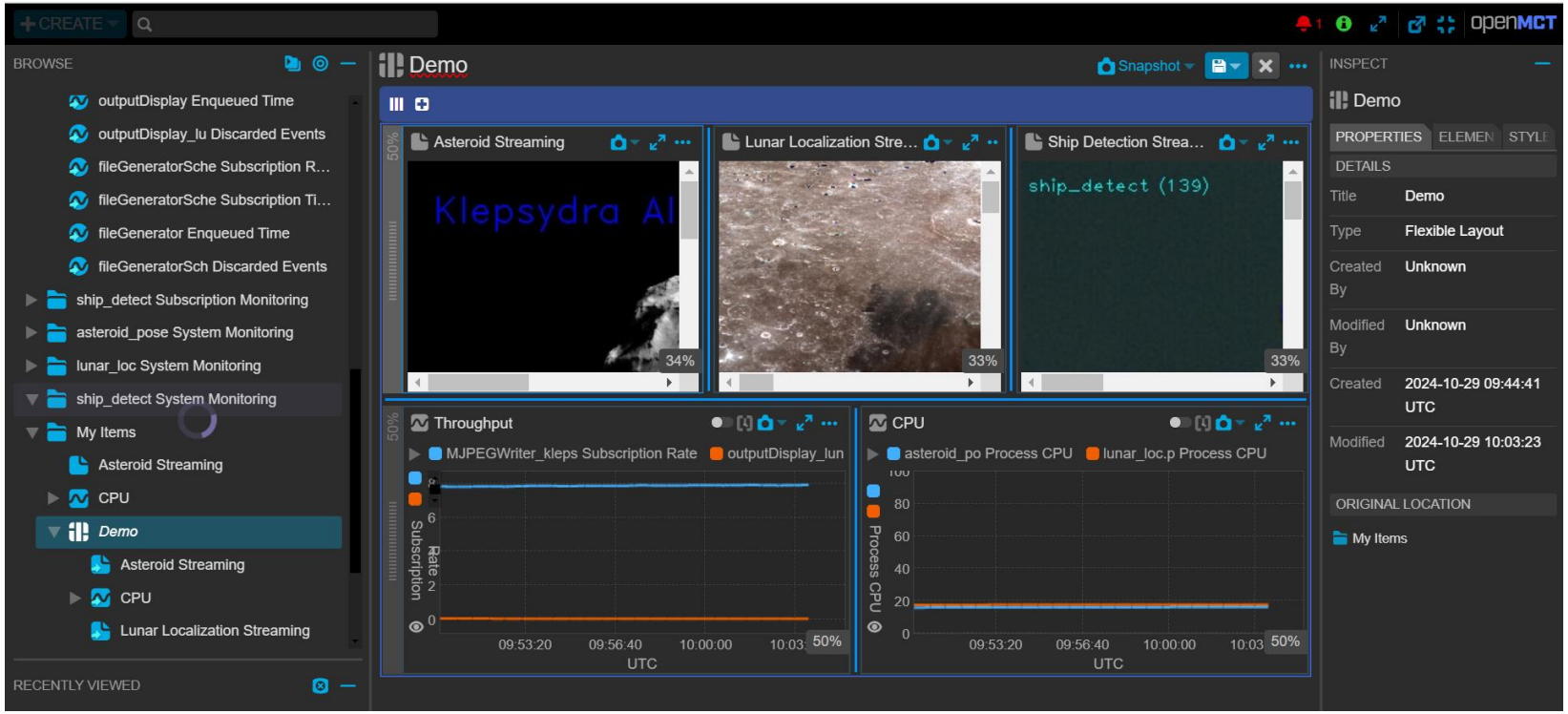
root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# vi ../conf/mem_dnn_conf.json
Expected output:

```
{
   "BoolProperties": {
      "export_layer_connections": false,
      "export_streaming_configuration": false,
      "export_through_tftp": false,
      "log_sdo_enable": false,
      "stat_file_container_enable": false,
      "stat_socket_container_enable": false,
      "test_mode": false,
      "use_default_streaming_factory": true,
      "use_pass_through": false,
      "use_policy_by_ratio": false
   },
   "IntProperties": {
      "critical_thread_pool_size": 4,
      "first_parallel_core": 0,
      "log_level": 2,
      "non_critical_thread_pool_size": 2,
      "number_of_cores": 1,
      "number_of_parallel_threads": 4,
      "pool_size": 1,
      "stat_admin_port": 0,
      "stat_log_interval_ms": 1000,
      "stat_system_port": 0
   },
   "StringProperties": {
      "container_name": "",
      "log_filename": "",
      "log_sdo_filename": "",
      "processor_intensive_layers": "conv",
      "stat_filename": "",
      "streaming_configuration": ""
   }
}
```

Copyright © 2025

# 5th step: Statistics collection (part II)

Socket statistics collection can be use for real-time monitoring in OpenMCT




Copyright © 2025

# 5th step: Statistics collection (part III)

highlighted in yellow the commands to execute

root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# apt update
Expected output:

root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# apt install vim
Expected output:

root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# vi ../conf/mem_dnn_conf.json
Expected output:

```
{
   "BoolProperties": {
      "export_layer_connections": false,
      "export_streaming_configuration": false,
      "export_through_tftp": false,
      "log_sdo_enable": false,
      "stat_file_container_enable": true,
      "stat_socket_container_enable": false,
      "test_mode": false,
      "use_default_streaming_factory": true,
      "use_pass_through": false,
      "use_policy_by_ratio": false
   },
   "IntProperties": {
      "critical_thread_pool_size": 4,
      "first_parallel_core": 0,
      "log_level": 2,
      "non_critical_thread_pool_size": 2,
      "number_of_cores": 1,
      "number_of_parallel_threads": 4,
      "pool_size": 1,
      "stat_admin_port": 0,
      "stat_log_interval_ms": 1000,
      "stat_system_port": 0
   },
   "StringProperties": {
      "container_name": ”alexnet",
      "log_filename": "",
      "log_sdo_filename": "",
      "processor_intensive_layers": "conv",
      "stat_filename": ”./alexnet.csv",
      "streaming_configuration": ""
   }
}
```

Copyright © 2025

# 5th step: Statistics collection (part IV)

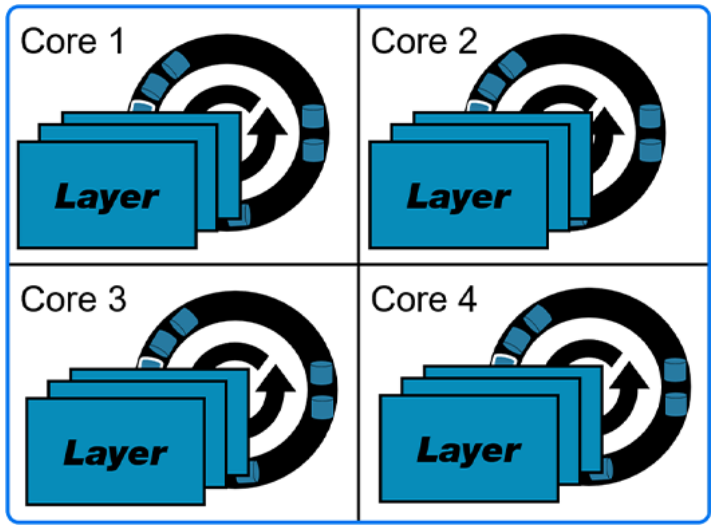
highlighted in yellow the commands to execute

```
root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# more alexnet.csv
1742819362282,cpuData,15,15,ramdata,951332864,784310272,167022592,51421184,managedServices,0,services,0,publishers,7,0_StatefulPartitionedCall/fun
ctional_1/conv2d_1/BiasAdd,892,892,1,11406,EVENT_LOOP,0,1,0_StatefulPartitionedCall/functional_1/
conv2d/BiasAdd,894,894,1,14948,EVENT_LOOP,0,1,0_StatefulPartitionedCall/functional_1/max_pooling2d/MaxPool,893,893,1,13854,EVENT_LOOP,0,1,0_Statef
ulPartitionedCall/functional_1/conv2d_2/BiasAdd,891,891,1,11146,EVENT_LOOP,0,1,0_StatefulPartitio
nedCall/functional_1/conv2d_3/BiasAdd,890,890,1,11094,EVENT_LOOP,0,1,0_StatefulPartitionedCall/functional_1/conv2d_4/BiasAdd,890,890,1,10989,EVENT
_LOOP,0,1,0_first,894,894,1,55885,EVENT_LOOP,0,1,subscribers,14,StatefulPartitionedCall/functiona
l_1/max_pooling2d/MaxPool_external_StatefulPartitionedCall/functional_1/max_pooling2d/MaxPool_external,892,892,1,14941722,UNSAFE_MULTI_EVENT_EMITT
ER,32396,0,StatefulPartitionedCall/functional_1/conv2d_1/BiasAdd-StatefulPartitionedCall/function
al_1/conv2d_2/BiasAdd_StatefulPartitionedCall/functional_1/conv2d_1/BiasAdd,891,891,1,25605154,UNSAFE_MULTI_EVENT_EMITTER,35573,0,first-
StatefulPartitionedCall/functional_1/conv2d/BiasAdd_first,893,893,1,16387332,UNSAFE_MULTI_EVENT_EMITTER,172
706,0,StatefulPartitionedCall/functional_1/conv2d_2/BiasAdd_external_StatefulPartitionedCall/functional_1/conv2d_2/BiasAdd_external,890,890,1,3046
7760,UNSAFE_MULTI_EVENT_EMITTER,31874,0,StatefulPartitionedCall/functional_1/conv2d_3/BiasAdd_ext
ernal_StatefulPartitionedCall/functional_1/conv2d_3/BiasAdd_external,889,889,1,15489998,UNSAFE_MULTI_EVENT_EMITTER,31510,0,first_external_first_ex
ternal,893,893,1,16397019,UNSAFE_MULTI_EVENT_EMITTER,167863,0,StatefulPartitionedCall/functional_
1/conv2d/BiasAdd-
StatefulPartitionedCall/functional_1/max_pooling2d/MaxPool_StatefulPartitionedCall/functional_1/conv2d/BiasAdd,892,892,1,826763,UNSAFE_MULTI_EVENT
_EMITTER,38750,0,callback_StatefulPartitionedCall/functional_1/conv2d_4/BiasAdd,
888,888,1,125884,UNSAFE_MULTI_EVENT_EMITTER,37604,0,StatefulPartitionedCall/functional_1/conv2d_4/BiasAdd_external_StatefulPartitionedCall/functio
nal_1/conv2d_4/BiasAdd_external,890,890,1,130467,UNSAFE_MULTI_EVENT_EMITTER,33854,0,StatefulParti
tionedCall/functional_1/conv2d/BiasAdd_external_StatefulPartitionedCall/functional_1/conv2d/BiasAdd_external,892,892,1,832388,UNSAFE_MULTI_EVENT_E
MITTER,35208,0,StatefulPartitionedCall/functional_1/conv2d_1/BiasAdd_external_StatefulPartitioned
Call/functional_1/conv2d_1/BiasAdd_external,891,891,1,25610415,UNSAFE_MULTI_EVENT_EMITTER,32448,0,StatefulPartitionedCall/functional_1/max_pooling
2d/MaxPool-StatefulPartitionedCall/functional_1/conv2d_1/BiasAdd_StatefulPartitionedCall/function
al_1/max_pooling2d/MaxPool,892,892,1,14936618,UNSAFE_MULTI_EVENT_EMITTER,35469,0,StatefulPartitionedCall/functional_1/conv2d_3/BiasAdd-
StatefulPartitionedCall/functional_1/conv2d_4/BiasAdd_StatefulPartitionedCall/functional_1/conv2d_3/BiasAdd,
889,889,1,15484946,UNSAFE_MULTI_EVENT_EMITTER,34322,0,StatefulPartitionedCall/functional_1/conv2d_2/BiasAdd-
StatefulPartitionedCall/functional_1/conv2d_3/BiasAdd_StatefulPartitionedCall/functional_1/conv2d_2/BiasAdd,890,890,1,30462656,UNSAFE_M
ULTI_EVENT_EMITTER,34843,0,...
```

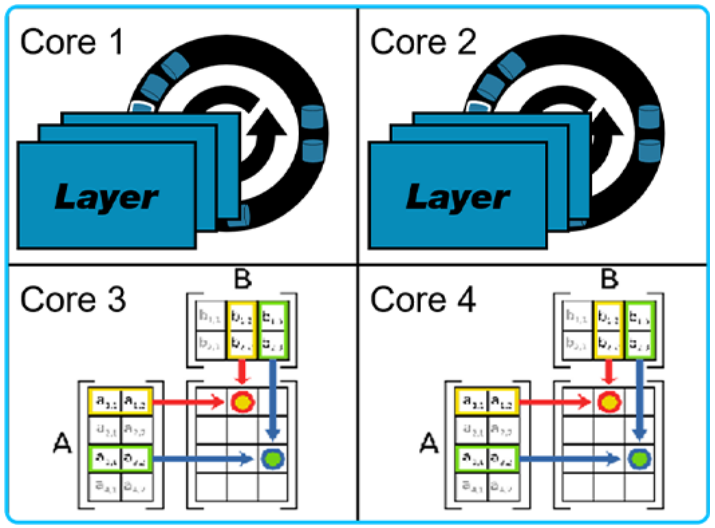
Copyright © 2025

# Example is optimised by default for latency

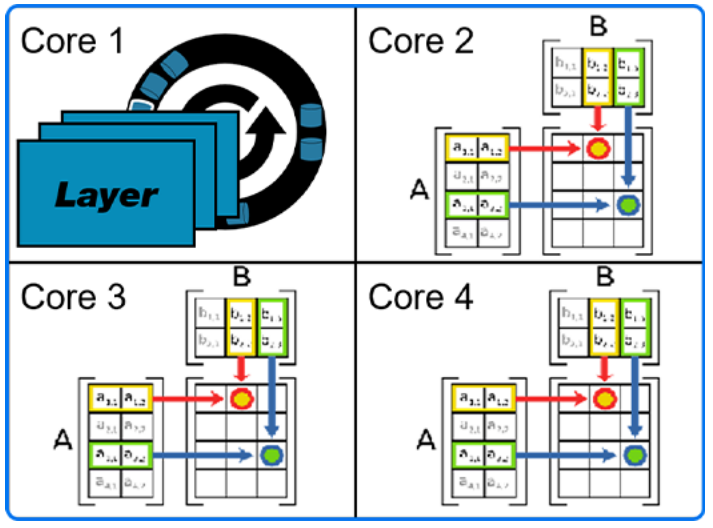
KLEPSYDRA TECHNOLOGIES




- **Low CPU power**
- Mid throughput CPU
- High latency




- Mid CPU
- **High throughput CPU**
- Mid latency




- High CPU
- Mid throughput CPU
- **Low latency**

Copyright © 2025

# 6th step: optimization of the example (part I)

highlighted in yellow the commands to execute

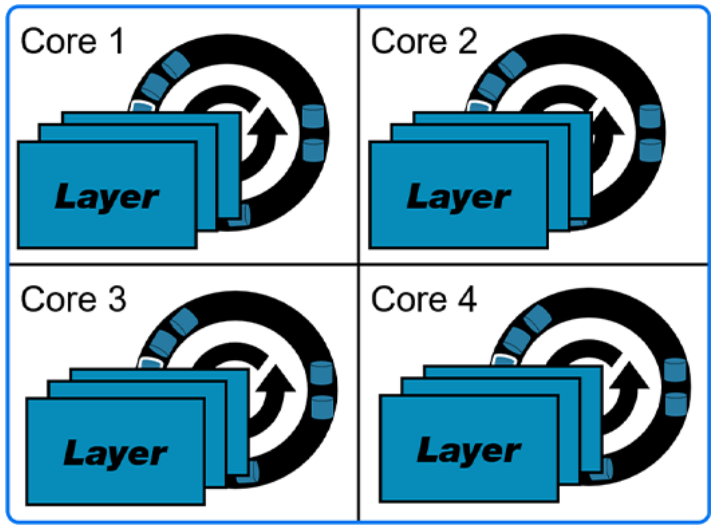
root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# vi ../conf/mem_dnn_conf.json

Expected output:

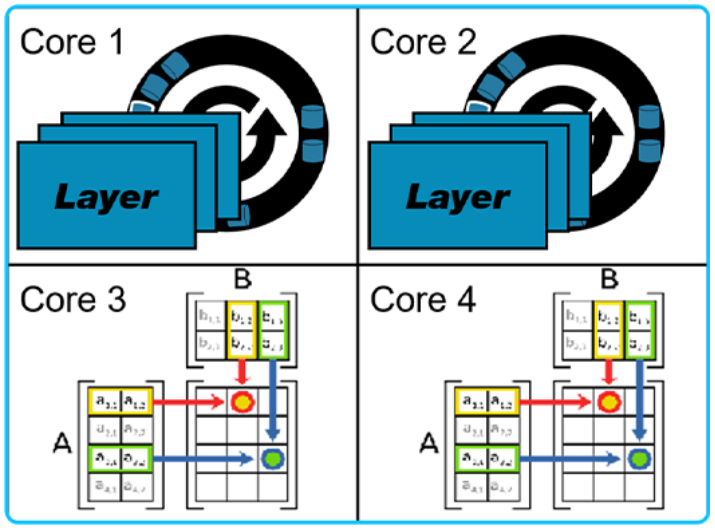
```
{
   "BoolProperties": {
      "export_layer_connections": false,
      "export_streaming_configuration": false,
      "export_through_tftp": false,
      "log_sdo_enable": false,
      "stat_file_container_enable": false,
      "stat_socket_container_enable": false,
      "test_mode": false,
      "use_default_streaming_factory": true,
      "use_pass_through": false,
      "use_policy_by_ratio": false
   },
   "IntProperties": {
      "critical_thread_pool_size": 4,
      "first_parallel_core": 0,
      "log_level": 2,
      "non_critical_thread_pool_size": 2,
      "number_of_cores": 1,
      "number_of_parallel_threads": 4,
      "pool_size": 1,
      "stat_admin_port": 0,
      "stat_log_interval_ms": 1000,
      "stat_system_port": 0
   },
   "StringProperties": {
      "container_name": "",
      "log_filename": "",
      "log_sdo_filename": "",
      "processor_intensive_layers": "conv",
      "stat_filename": "",
      "streaming_configuration": ""
   }
}
```

Copyright © 2025

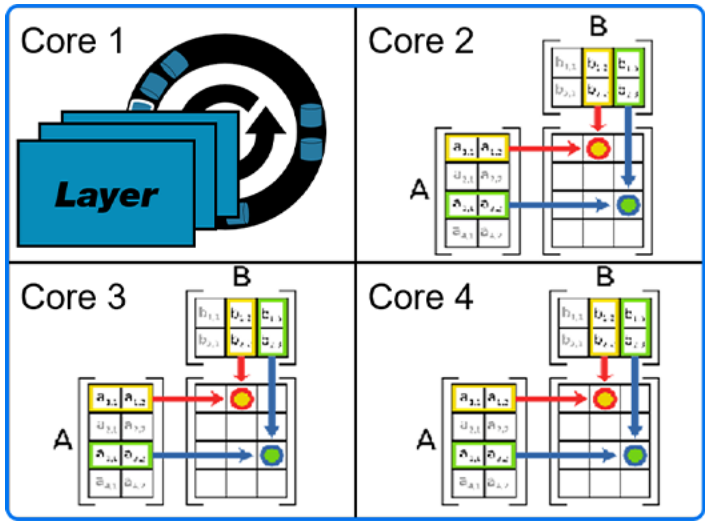
# Trying to optimise for CPU




- **Low CPU power**
- Mid throughput CPU
- High latency




- Mid CPU
- **High throughput CPU**
- Mid latency




- High CPU
- Mid throughput CPU
- **Low latency**

Copyright © 2025

# 6th step: optimization of the example (part II)

highlighted in yellow the configuration variables to change

root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# vi ../conf/mem_dnn_conf.json

Expected output:

```
{
   "BoolProperties": {
      "export_layer_connections": false,
      "export_streaming_configuration": false,
      "export_through_tftp": false,
      "log_sdo_enable": false,
      "stat_file_container_enable": false,
      "stat_socket_container_enable": false,
      "test_mode": false,
      "use_default_streaming_factory": true,
      "use_pass_through": false,
      "use_policy_by_ratio": false
   },
   "IntProperties": {
      "critical_thread_pool_size": 8,
      "first_parallel_core": 0,
      "log_level": 2,
      "non_critical_thread_pool_size": 2,
      "number_of_cores": 4,
      "number_of_parallel_threads": 1,
      "pool_size": 1,
      "stat_admin_port": 0,
      "stat_log_interval_ms": 1000,
      "stat_system_port": 0
   },
   "StringProperties": {
      "container_name": "",
      "log_filename": "",
      "log_sdo_filename": "",
      "processor_intensive_layers": "conv",
      "stat_filename": "",
      "streaming_configuration": ""
   }
}
```

Copyright © 2025

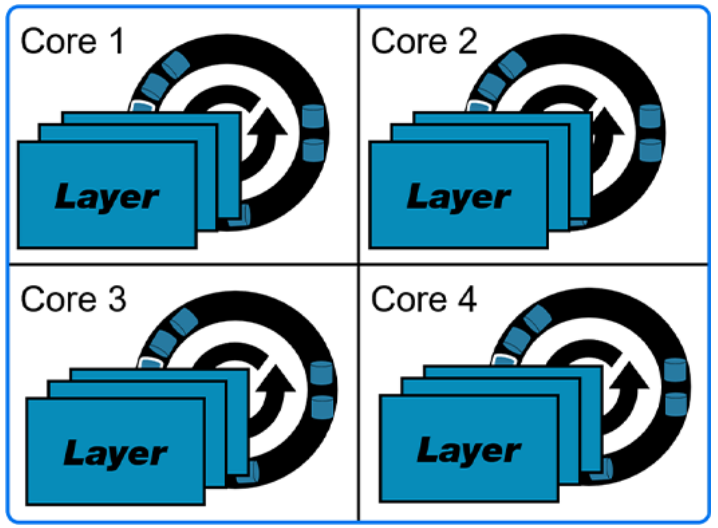
# 6th step: optimization of the example (part III)

highlighted in yellow the commands to execute

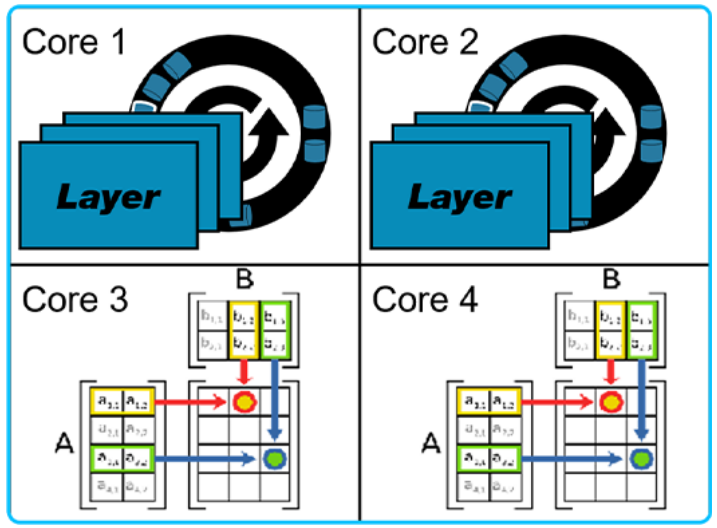
```
root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# ./ai-min-example -f ../conf/mem_dnn_conf.json -m ../models/alexnet.kpsr
Expected output:
inputData.configurationFile: ../conf/mem_dnn_conf.json
inputData.modelFile: ../models/alexnet.kpsr
[Mon Mar 24 12:48:47 2025] [12:48:47 386549] [klepsydra] [info] [223] Layers to be optimised out: StatefulPartitionedCall/functional_1/re_lu/Relu
[Mon Mar 24 12:48:47 2025] [12:48:47 386746] [klepsydra] [info] [223] Removing StatefulPartitionedCall/functional_1/re_lu_4/Relu
.........
It took 642ms to load the model.
Applying AI to dummy image 0 of 100
Image : 0 assigned id : 0
 Use this id in callback to match input to result.
Got result for input with id: 0
Processed 0 of 100 in 334ms.
Applying AI to dummy image 1 of 100
Image : 1 assigned id : 1
 Use this id in callback to match input to result.
Got result for input with id: 1
Processed 1 of 100 in 326ms.
Applying AI to dummy image 2 of 100
```

Copyright © 2025

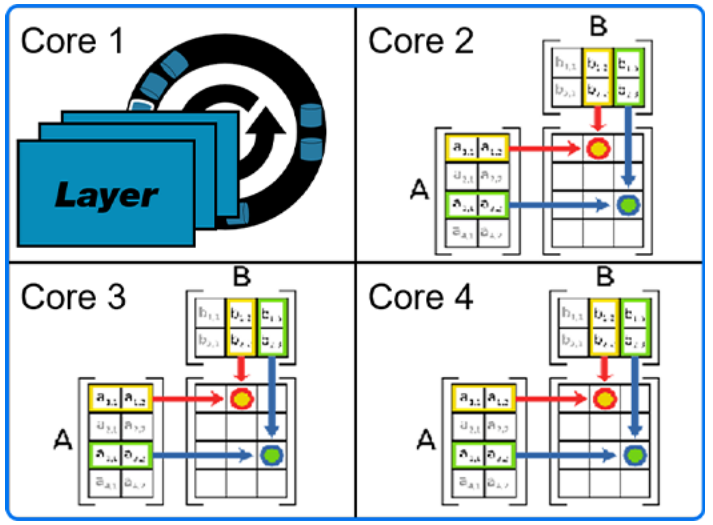
# How to optimise for throughput




- **Low CPU power**
- Mid throughput CPU
- High latency




- Mid CPU
- **High throughput CPU**
- Mid latency




- High CPU
- Mid throughput CPU
- **Low latency**

Copyright © 2025

# 6th step: optimization of the example (part IV)

highlighted in yellow the configuration variables to change

root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# vi ../conf/mem_dnn_conf.json

Expected output:

```
{
   "BoolProperties": {
      "export_layer_connections": false,
      "export_streaming_configuration": true,
      "export_through_tftp": false,
      "log_sdo_enable": false,
      "stat_file_container_enable": false,
      "stat_socket_container_enable": false,
      "test_mode": false,
      "use_default_streaming_factory": true,
      "use_pass_through": false,
      "use_policy_by_ratio": false
   },
   "IntProperties": {
      "critical_thread_pool_size": 8,
      "first_parallel_core": 0,
      "log_level": 2,
      "non_critical_thread_pool_size": 2,
      "number_of_cores": 4,
      "number_of_parallel_threads": 1,
      "pool_size": 1,
      "stat_admin_port": 0,
      "stat_log_interval_ms": 1000,
      "stat_system_port": 0
   },
   "StringProperties": {
      "container_name": "",
      "log_filename": "",
      "log_sdo_filename": "",
      "processor_intensive_layers": "conv",
      "stat_filename": "",
      "streaming_configuration": "./alexnet.json"
   }
}
```

Copyright © 2025

# 6th step: optimization of the example (part V)

highlighted in yellow the configuration variables to change

root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# vi ../conf/mem_dnn_conf.json

Expected output:

```
{
   "BoolProperties": {
      "export_layer_connections": false,
      "export_streaming_configuration": true,
      "export_through_tftp": false,
      "log_sdo_enable": false,
      "stat_file_container_enable": false,
      "stat_socket_container_enable": false,
      "test_mode": false,
      "use_default_streaming_factory": true,
      "use_pass_through": false,
      "use_policy_by_ratio": false
   },
   "IntProperties": {
      "critical_thread_pool_size": 8,
      "first_parallel_core": 0,
      "log_level": 2,
      "non_critical_thread_pool_size": 2,
      "number_of_cores": 4,
      "number_of_parallel_threads": 1,
      "pool_size": 1,
      "stat_admin_port": 0,
      "stat_log_interval_ms": 1000,
      "stat_system_port": 0
   },
   "StringProperties": {
      "container_name": "",
      "log_filename": "",
      "log_sdo_filename": "",
      "processor_intensive_layers": "conv",
      "stat_filename": "",
      "streaming_configuration": "./alexnet.json"
   }
}
```

Copyright © 2025

# 6th step: optimization of the example (part VI)

highlighted in yellow the configuration variables to change

```
root@8022d097740c:~/kpe-ai-examples/ai-min-example/build# more alexnet.json
{
   "version_number": 3,
   "pool_size": 1,
   "number_of_cores": 4,
   "number_of_event_loops": 4,
   "number_of_parallel_threads": 1,
   "first_parallel_core": 0,
   "critical_thread_pool_size": 8,
   "non_critical_thread_pool_size": 2,
   "parallelised_layers": [],
   "event_loop_core_map": {
      "0": [
         0
      ],
      "1": [
         1
      ],
      "2": [
         2
      ],
      "3": [
         3
      ]
   },
   "layer_event_loop_map": {
      "StatefulPartitionedCall/functional_1/conv2d/BiasAdd": {
         "coreId": 0,
         "factoryType": 1
      },
      "StatefulPartitionedCall/functional_1/conv2d_1/BiasAdd": {
         "coreId": 1,
         "factoryType": 1
      },
      "StatefulPartitionedCall/functional_1/conv2d_2/BiasAdd": {
         "coreId": 2,
         "factoryType": 1
      },
      "StatefulPartitionedCall/functional_1/conv2d_3/BiasAdd": {
         "coreId": 3,
         "factoryType": 1
      },
      "StatefulPartitionedCall/functional_1/conv2d_4/BiasAdd": {
         "coreId": 3,
         "factoryType": 1
      },
      "StatefulPartitionedCall/functional_1/max_pooling2d/MaxPool": {
         "coreId": 1,
         "factoryType": 1
      },
      "first": {
         "coreId": 0,
         "factoryType": 1
      }
   }
}
```

Copyright © 2025

# Thank You

@ Name.Name@name.eu

# EAI

https://edge-ai-tech.eu/

The two-day innovative educational initiative is designed for students, young researchers, and professionals eager to expand their understanding and skills in these cutting-edge technologies.

Participants will engage in interactive discussions, hands-on sessions, and lectures covering foundational concepts and innovative applications of edge AI across various domains with leading experts, gain insights into real-world applications, and work on team-based projects to foster collaboration and innovation.

Copyright © 2025

# EDGE AI Academy 2025

## Summer School Objectives

- The EDGE AI Academy Summer School aims to empower participants with advanced knowledge, hands-on experience and an in-depth understanding of edge AI, which combines Artificial Intelligence (AI) with edge computing and the Internet of Things (IoT) to process data closer to its source.
- Edge AI technology is gaining prominence due to its ability to enhance real-time decision-making, reduce latency, improve data privacy, and diminish reliance on continuous internet connectivity.
- The topics will extend to RISC-V architectures as ultra-low-power high-performance RISC-V-based processors and ultra-low-power vector-based accelerators are advancing smart IoT and edge AI computing capabilities.

Copyright © 2025