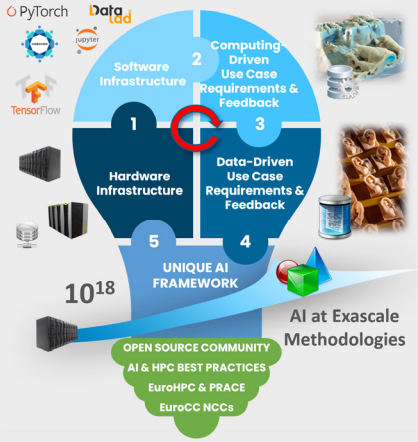
Towards adoptions of a Unique AI Framework for the EuroHPC JU Systems Ecosystem

Prof. Dr. – Ing. Morris Riedel et al.
Full Professor, School of Engineering & Natural Sciences, University of Iceland
Lead of CoE RAISE WP2 – AI- and HPC-Cross Methods at Exascale
2022-12-01, AI & Simulation-based Engineering Workshop, Prague

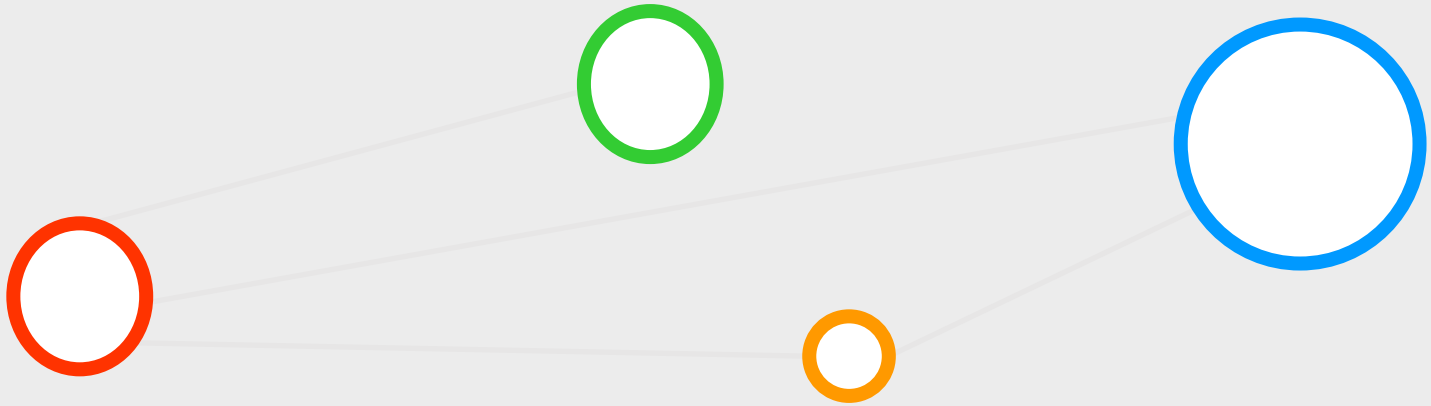
@ProfDrMorrisRiedel @Morris Riedel @MorrisRiedel @MorrisRiedel
https://www.youtube.com/channel/UCWC4VKHmL4NZgFfKoHtANKg morris@hi.is

Outline

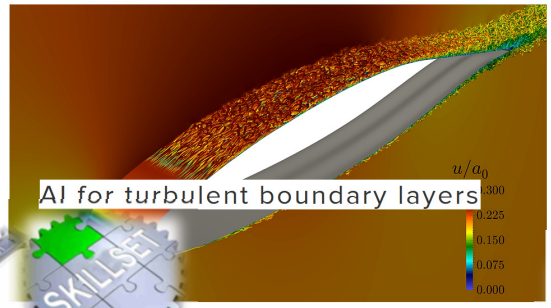
- Challenges in using AI Methods on HPC at Scale
  - Review Toolset & Skillset Challenges
- Unique AI Framework (UAIF) Co-Design Process
  - UAIF Co-Design at A Glance
  - Factsheets & Interaction Rooms
- CoE RAISE UAIF Evolution & Blueprint
  - Evolution of Versions
  - Current Blueprint
- Adoption Roadmap of the Framework
  - Cooperation with NCCs & EuroHPC JU Hosting Sites
- Summary & Q&A
  - Feedback from NCCs
- Selected References



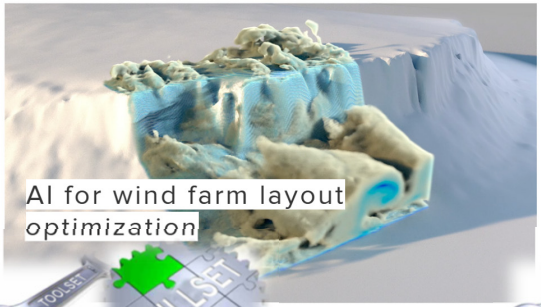
Challenges in using AI Methods on HPC at Scale



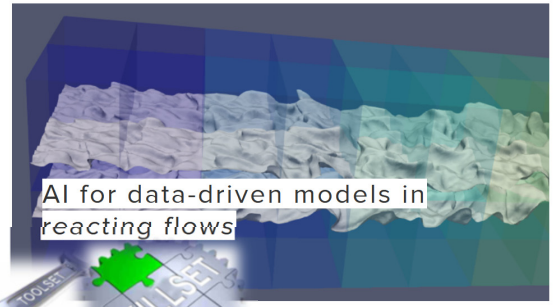
Compute & Data-driven Use Cases – Complex Challenges



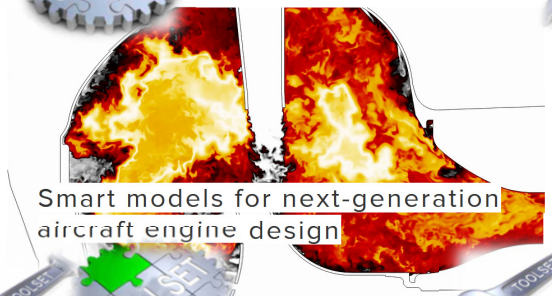
AI for turbulent boundary layers



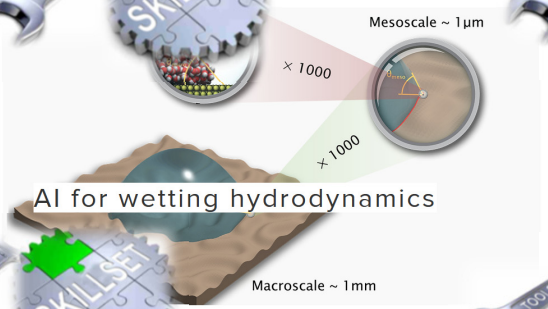
AI for wind farm layout optimization



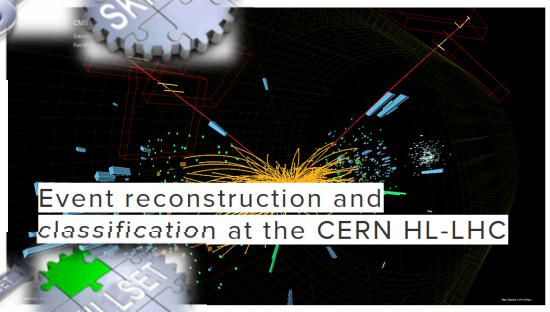
AI for data-driven models in reacting flows



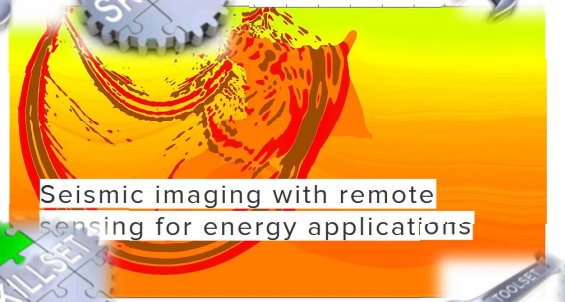
Smart models for next-generation aircraft engine design



AI for wetting hydrodynamics

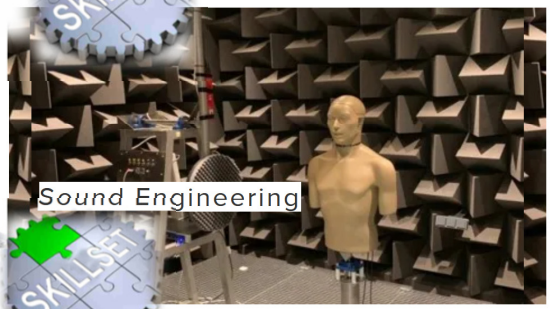


Event reconstruction and classification at the CERN HL-LHC

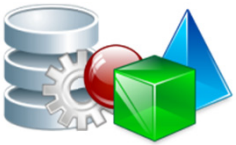


Seismic imaging with remote sensing for energy applications

Defect-free metal additive manufacturing



Sound Engineering



PyTorch

Horovod

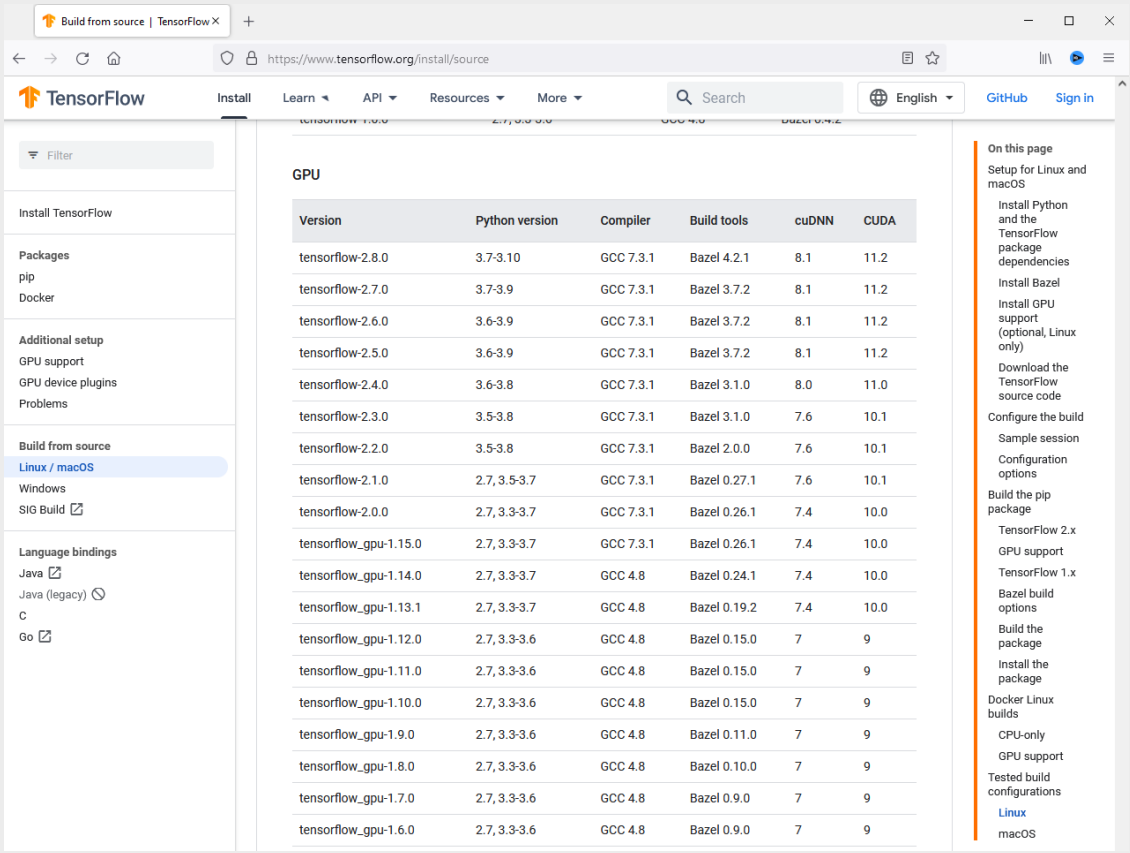
TensorFlow

Data Lad

jupyter

Requirements Gathering Process – Version Challenges

- Example: TensorFlow
  - Can we create an automated module checker for the SW Framework RAISE?
  - Specific versions of TensorFlow require specific versions of underlying HPC modules or other AI frameworks to work in specific versions together
  - Python versions must be correct as well
  - E.g., differences in Python 3.8.x and 3.9.x
  - Support AI developers for many other tools like PyTorch, Horovod, Ray Tune, etc.



GPU

Version	Python version	Compiler	Build tools	cuDNN	CUDA
tensorflow-2.8.0	3.7-3.10	GCC 7.3.1	Bazel 4.2.1	8.1	11.2
tensorflow-2.7.0	3.7-3.9	GCC 7.3.1	Bazel 3.7.2	8.1	11.2
tensorflow-2.6.0	3.6-3.9	GCC 7.3.1	Bazel 3.7.2	8.1	11.2
tensorflow-2.5.0	3.6-3.9	GCC 7.3.1	Bazel 3.7.2	8.1	11.2
tensorflow-2.4.0	3.6-3.8	GCC 7.3.1	Bazel 3.1.0	8.0	11.0
tensorflow-2.3.0	3.5-3.8	GCC 7.3.1	Bazel 3.1.0	7.6	10.1
tensorflow-2.2.0	3.5-3.8	GCC 7.3.1	Bazel 2.0.0	7.6	10.1
tensorflow-2.1.0	2.7, 3.5-3.7	GCC 7.3.1	Bazel 0.27.1	7.6	10.1
tensorflow-2.0.0	2.7, 3.3-3.7	GCC 7.3.1	Bazel 0.26.1	7.4	10.0
tensorflow_gpu-1.15.0	2.7, 3.3-3.7	GCC 7.3.1	Bazel 0.26.1	7.4	10.0
tensorflow_gpu-1.14.0	2.7, 3.3-3.7	GCC 4.8	Bazel 0.24.1	7.4	10.0
tensorflow_gpu-1.13.1	2.7, 3.3-3.7	GCC 4.8	Bazel 0.19.2	7.4	10.0
tensorflow_gpu-1.12.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.15.0	7	9
tensorflow_gpu-1.11.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.15.0	7	9
tensorflow_gpu-1.10.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.15.0	7	9
tensorflow_gpu-1.9.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.11.0	7	9
tensorflow_gpu-1.8.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.10.0	7	9
tensorflow_gpu-1.7.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.9.0	7	9
tensorflow_gpu-1.6.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.9.0	7	9

Requirements Gathering Process – Module Challenges

- Example of Setups
  - Many different versions / combinations
  - E.g. FZJ JSC DEEP-EST HPC System

```
[riedel1@dp-dam01 ~]$ module spider cudnn

---------------------------------------------------------------------------------------------------------------------------------------------------------------
  cuDNN:
---------------------------------------------------------------------------------------------------------------------------------------------------------------
     Description:
       The NVIDIA CUDA Deep Neural Network library (cuDNN) is a GPU-accelerated library of primitives for deep neural networks.

     Versions:
        cuDNN/7.4.1.5-CUDA-9.2.88
        cuDNN/7.5.1.10-CUDA-10.1.105
        cuDNN/7.6.4.38-CUDA-10.1.105
        cuDNN/7.6.5.32-CUDA-10.2.89
        cuDNN/8.0.2.39-CUDA-11.0
        cuDNN/8.2.1.32-CUDA-11.3
        cuDNN/8.3.1.22-CUDA-11.5

---------------------------------------------------------------------------------------------------------------------------------------------------------------
  For detailed information about a specific "cuDNN" module (including how to load the modules) use the module's full name.
  For example:

     $ module spider cuDNN/7.6.5.32-CUDA-10.2.89
---------------------------------------------------------------------------------------------------------------------------------------------------------------
```

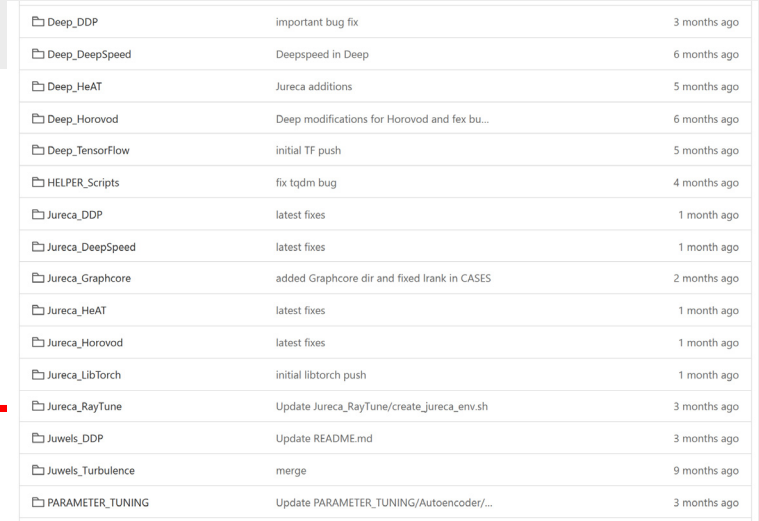
```
[riedel1@dp-dam01 ~]$ module spider nccl

---------------------------------------------------------------------------------------------------------------------------------------------------------------
  NCCL:
---------------------------------------------------------------------------------------------------------------------------------------------------------------
     Description:
       The NVIDIA Collective Communications Library (NCCL) implements multi-GPU and multi-node collective communication primitives that are performance optimized for NVIDIA GPUs.

     Versions:
        NCCL/2.4.2-1-CUDA-9.2.88
        NCCL/2.4.6-1-CUDA-10.1.105
        NCCL/2.4.8-CUDA-10.1.105
        NCCL/2.4.8
        NCCL/2.7.3-1-CUDA-10.2.89
        NCCL/2.8.3-1-CUDA-11.0
        NCCL/2.8.3-1-CUDA-11.3
        NCCL/2.10.3-1-CUDA-11.3
        NCCL/2.11.4-CUDA-11.5

---------------------------------------------------------------------------------------------------------------------------------------------------------------
  For detailed information about a specific "NCCL" module (including how to load the modules) use the module's full name.
  For example:

     $ module spider NCCL/2.7.3-1-CUDA-10.2.89
---------------------------------------------------------------------------------------------------------------------------------------------------------------
```

```
[riedel1@dp-dam01 ~]$ module spider tensorflow

---------------------------------------------------------------------------------------------------------------------------------------------------------------
  TensorFlow:
---------------------------------------------------------------------------------------------------------------------------------------------------------------
     Description:
       An open-source software library for Machine Intelligence

     Versions:
        TensorFlow/1.12.0-GPU-Python-2.7.15
        TensorFlow/1.12.0-GPU-Python-3.6.6
        TensorFlow/1.13.1-GPU-Python-3.6.8
        TensorFlow/2.2.0-GPU-Python-3.6.8-1
        TensorFlow/2.3.1-Python-3.8.5
        TensorFlow/2.5.0-Python-3.8.5
        TensorFlow/2.6.0-CUDA-11.5

---------------------------------------------------------------------------------------------------------------------------------------------------------------
  For detailed information about a specific "TensorFlow" module (including how to load the modules) use the module's full name.
  For example:

     $ module spider TensorFlow/2.2.0-GPU-Python-3.6.8-1
---------------------------------------------------------------------------------------------------------------------------------------------------------------
```

```
[riedel1@dp-dam01 ~]$ module spider cuda

---------------------------------------------------------------------------------------------------------------------------------------------------------------
  CUDA:
---------------------------------------------------------------------------------------------------------------------------------------------------------------
     Description:
       CUDA (formerly Compute Unified Device Architecture) is a parallel computing platform and programming model created by NVIDIA and implemented by the graphics processing units (GPUs) that they produce. CUDA gives developers access
       to the virtual instruction set and memory of the parallel computational elements in CUDA GPUs.

     Versions:
        CUDA/9.2.88
        CUDA/10.1.105
        CUDA/10.2.89
        CUDA/11.0
        CUDA/11.0.207
        CUDA/11.3
        CUDA/11.5

---------------------------------------------------------------------------------------------------------------------------------------------------------------
  For detailed information about a specific "CUDA" module (including how to load the modules) use the module's full name.
  For example:

     $ module spider CUDA/11.0.207
---------------------------------------------------------------------------------------------------------------------------------------------------------------
```

Example: Detailed Knowledge of Modules Necessary

- Modules
  - Vary heavily between different HPC systems
  - 2-3 Days/Months spend by researchers for getting the right environment / HPC system
  - Goal: UAIF simplify setup of components
  - E.g., automated job script generator for right module setup
  - E.g., re-usable scripts



Deep_DDP	important bug fix	3 months ago
Deep_DeepSpeed	Deepspeed in Deep	6 months ago
Deep_HeAT	Jureca additions	5 months ago
Deep_Horovod	Deep modifications for Horovod and fex bu...	6 months ago
Deep_TensorFlow	initial TF push	5 months ago
HELPER_Scripts	fix tqdm bug	4 months ago
Jureca_DDP	latest fixes	1 month ago
Jureca_DeepSpeed	latest fixes	1 month ago
Jureca_Graphcore	added Graphcore dir and fixed lrank in CASES	2 months ago
Jureca_HeAT	latest fixes	1 month ago
Jureca_Horovod	latest fixes	1 month ago
Jureca_LibTorch	initial libtorch push	1 month ago
Jureca_RayTune	Update Jureca_RayTune/create_jureca_env.sh	3 months ago
Juwels_DDP	Update README.md	3 months ago
Juwels_Turbulence	merge	9 months ago
PARAMETER_TUNING	Update PARAMETER_TUNING/Autoencoder/...	3 months ago

```
#!/usr/bin/env bash

# Slurm job configuration
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=4
#SBATCH --cpus-per-gpu=20
#SBATCH --account=hai_so2sat
#SBATCH --output=output.out
#SBATCH --error=error.er
#SBATCH --time=6:00:00
#SBATCH --job-name=BENTF2
#SBATCH --gres=gpu:1 --partition=booster

#load modules
ml Stages/2020  GCC/9.3.0  OpenMPI/4.1.0rc1
ml Horovod/0.20.3-Python-3.8.5
ml TensorFlow/2.3.1-Python-3.8.5
#activate my virtualenv
#source /p/project/joaiml/remote_sensing/rocco_sedona/ben_TF2/scripts/env_tf2_juwels_booster/bin/activate

#export relevant env variables
#export CUDA_VISIBLE_DEVICES="0,1,2,3"

#run Python program
srun --cpu-bind=none python -u train_hvd_keras_aug.py
```

Already available for the community: https://gitlab.jsc.fz-juelich.de/CoE-RAISE/FZJ/ai-for-hpc-oa

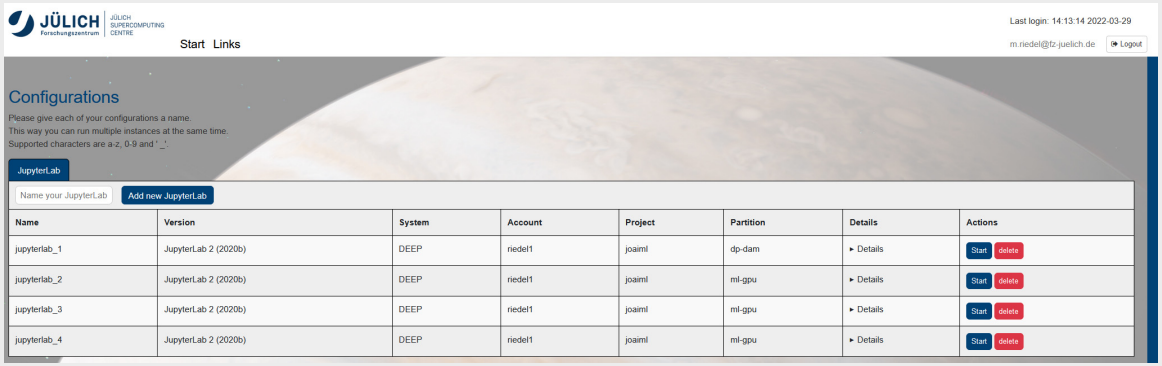
Requirements Gathering Process – Jupyter Challenges

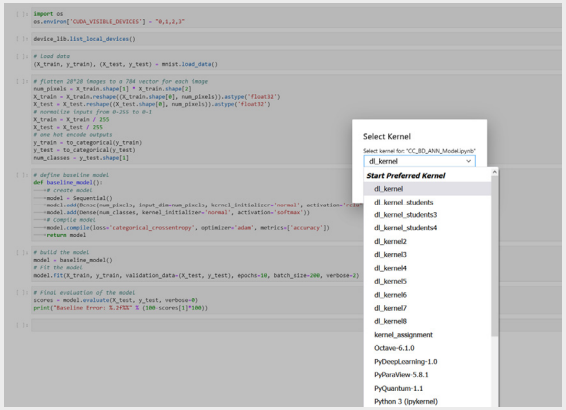
- Initial ideas collected on WP2 RAISE
  - Should be transformed in a proper GIT structure (new WP2 RAISE programmer & RAISE folks → next slide)
  - Selected artefacts of different types: Jupyter notebooks of AI codes, Kernel for Jupyter notebooks, infos links to Nvidia drivers
  - Context: Concrete HPC machines and porting code between them

  - Practice & experience: Shows highly unstable environments for AI configuration and setups (not deterministic behaviours) → room for framework idea
  - Lessons learned: PIs / PHDs invested many hours to identify issues in kernel developments with new stages and new python versions → we need improvements!

Requirements Gathering Process – Time Efforts Challenges

- Example of Setups
  - Tried many varieties of kernels
  - Developers /PIs / PhD Students loose ~3-4 hours average by trying new HPC machine just to get new modules right and/or setup kernels that work with modules
  - Selected debug/solution tools not known always, e.g., nvidia-smi, really scalable components, etc.
  - Note: Jupyter framework itself seems not to be the problem, rather complex hardware/software configurations

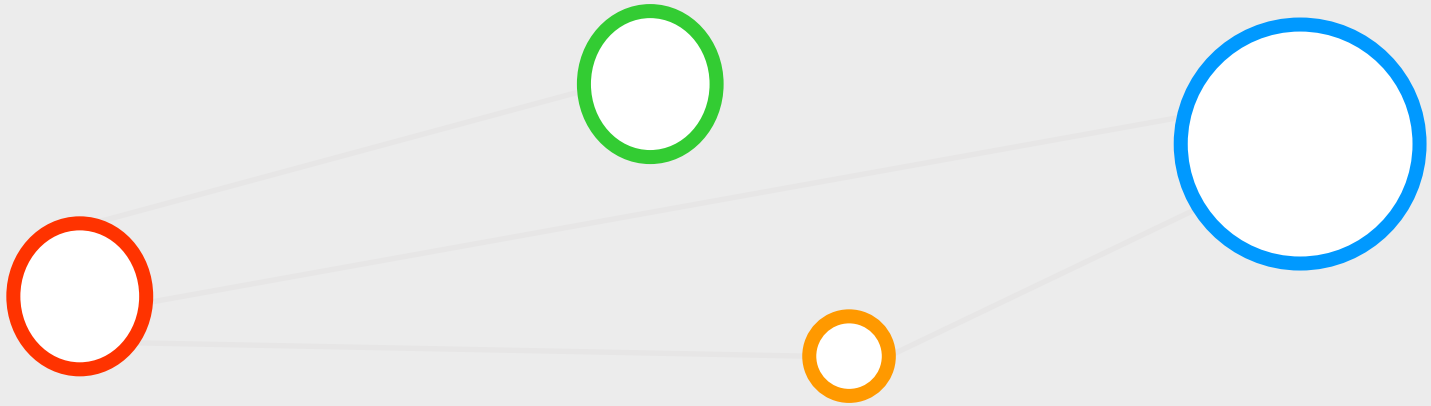




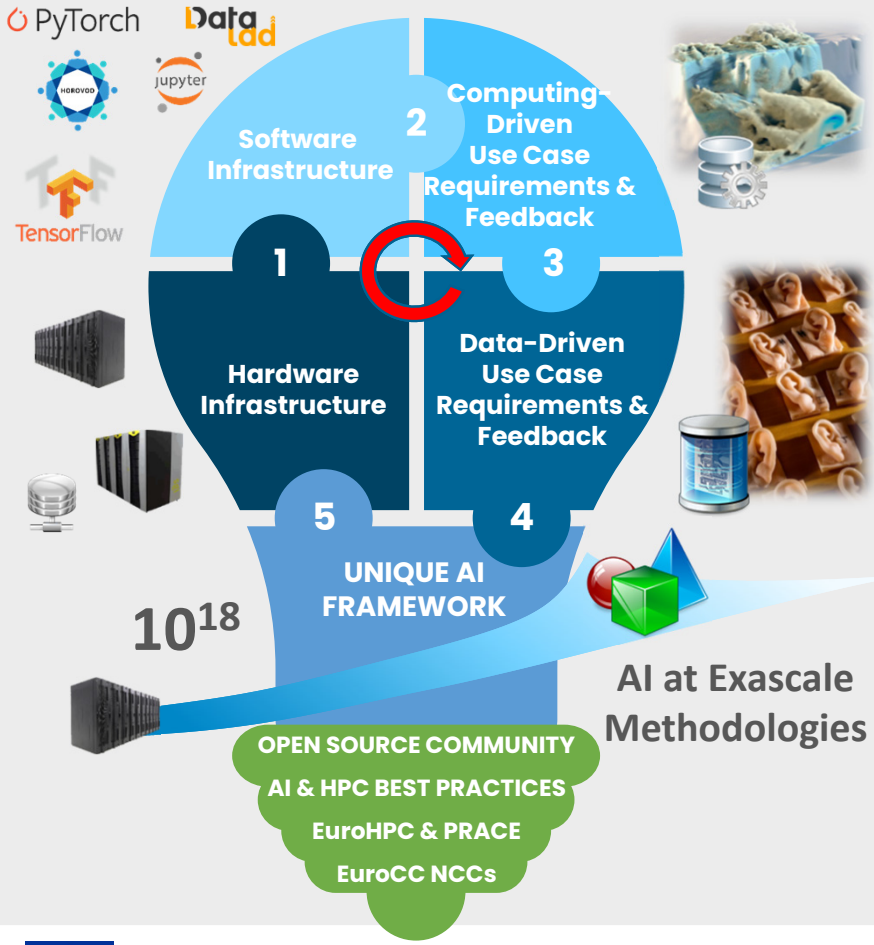
Requirements Gathering Process – Initial Results

#	Description	Level	Affected Technologies
RQ1	Applications need to run on different HPC systems without detailed knowledge of underlying package versions.	Hardware Software	DL libraries, e.g., TensorFlow, Keras, PyTorch, Horovod.
RQ2	Proven scalability of the framework components.	Software	Horovod and PyTorch-DDP should have shown good scaling capabilities, see findings in Section 3.3.
RQ3	Support for ONNX for TL and model sharing.	Software	DL models need to store and re-use ONNX models.
RQ4	High-level access via Jupyter notebook for DL modeling including Kernels aware/choice of HPC modules.	Software	Jupyter notebook and JupyterLab environments should be supported by the framework.
RQ5	Low-level access via batch submission enabling automation and scaling of multi-GPU setups.	Software	SSH protocols to enable low-level access to HPC systems using their batch schedulers.
RQ6	Enable reproducibility by using open-source description of data science tasks and AI models.	Software	MLFlow and ClearML and associated deployments on HPC systems re-using container technologies.
RQ7	Enable support for container technologie to enable portability between different HPC centers with different AI stacks.	Software	Technologies like Singularity and Docker enable the portability between HPC systems and are used in HPC centers, e.g., at JSC and BSC-CNS.
RQ8	The framework should be agnostic with respect to accelerator types to use it with DL applications without knowing the details of different accelerator types and underlying libraries.	Hardware	Accelerators, e.g., NVIDIA GPUs, are instrumental for DL and towards Exascale it is expected to leverage other accelerator types as well, e.g., AMD Instinct MI100.
RQ9	The framework needs to support cutting-edge I/O capabilities for high performance and be able to work with large datasets.	Hardware	Towards Exascale, large quantities of datasets are expected and the I/O capabilities of the framework need to leverage the underlying hardware infrastrastructure.

Unique AI Framework (UAIF) Co-Design Process



Unique AI Framework (UAIF) Co-Design Process at a Glance



Hardware Infrastructure

Prepare & Document available production systems at partners' HPC centers

Examples: JUWELS (JUELICH), LUMI (UoICELAND), DEEP Modular Prototypes, JUNIQ (JUELICH), etc.

Software Infrastructure

Prepare & Document available open source tools & libraries for HPC & AI useful for implementing use cases

Examples: DeepSpeed and/or Horovod for interconnecting N GPUs for a scalable deep learning jobs

Computing-driven Use Cases Requirements & Feedback

Use cases with emphasize on computing bring in co-design information about AI framework & hardware

Examples: Use feedback that TensorFlow does not work nicely, so WP2 works with use cases on pyTorch

Data-driven Use Cases Requirements & Feedback

Use cases with emphasize on data bring in co-design information about AI framework & hardware

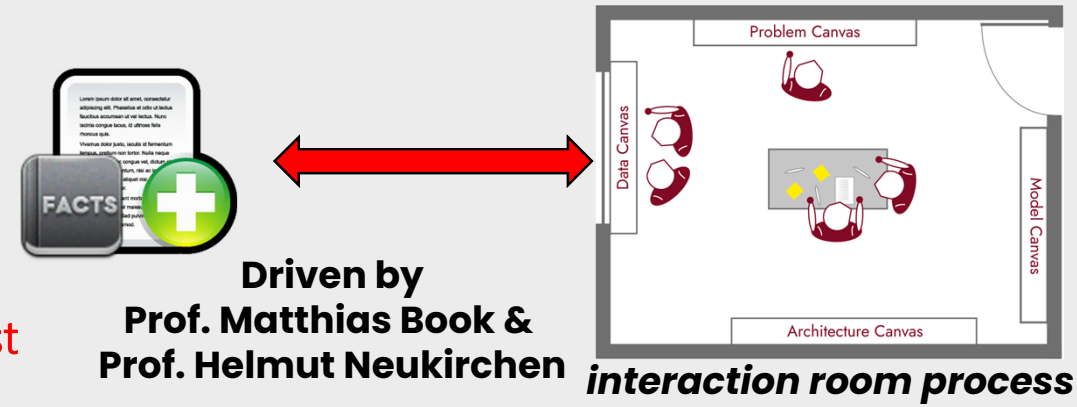
Examples: Deployment blueprint by using AI training on cluster module & inference/testing on booster

→ UNIQUE AI FRAMEWORK (UAIF)

Living design document & software framework blueprint for HPC & AI also with pretrained AI models

Unique AI Framework (UAIF) Co-Design Process Approaches

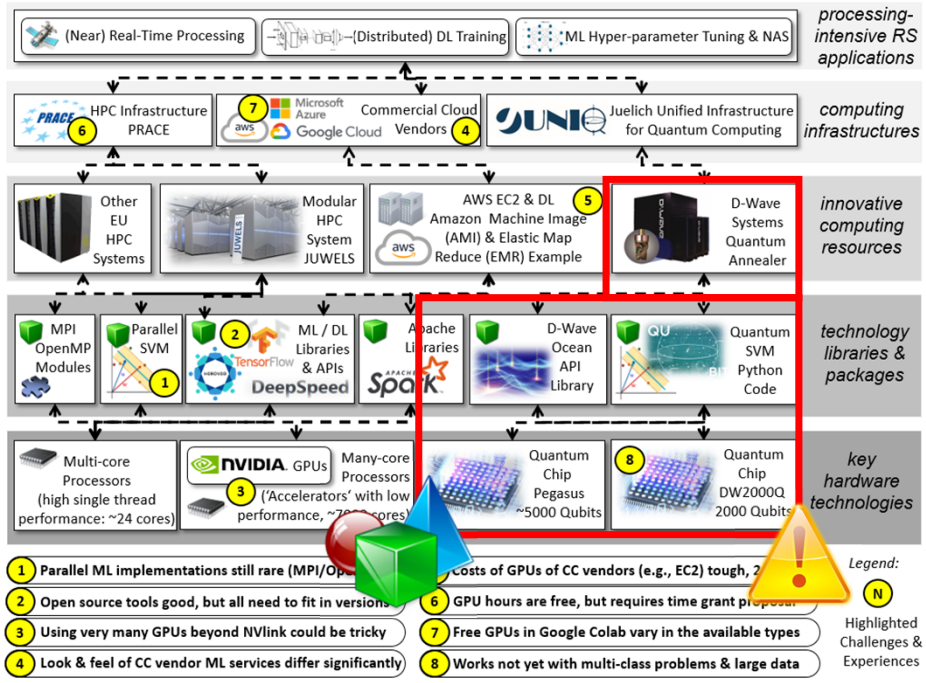
- Fact Sheets
  - Foster initial understanding
  - Living document & each Fact Sheet per WP3/WP4 Use Case
  - *(Experience from many other EU projects)*
- Selected Contents
  - Short Application Introduction
  - Clarify Primary Contacts
  - Codes/Libraries/Executables
  - HPC System Usage Details
  - Specific Platforms & 'where is what data'?
  - Machine/Deep Learning Approaches of Interest



Driven by Prof. Matthias Book & Prof. Helmut Neukirchen

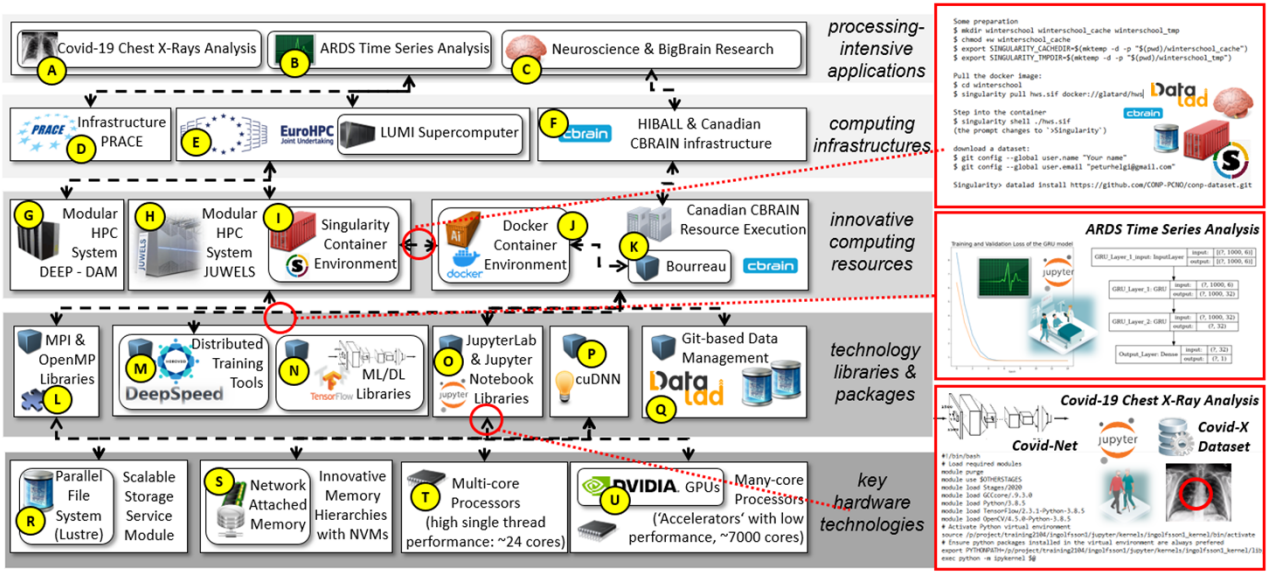
interaction room process

Including Innovative Quantum Computing Hardware

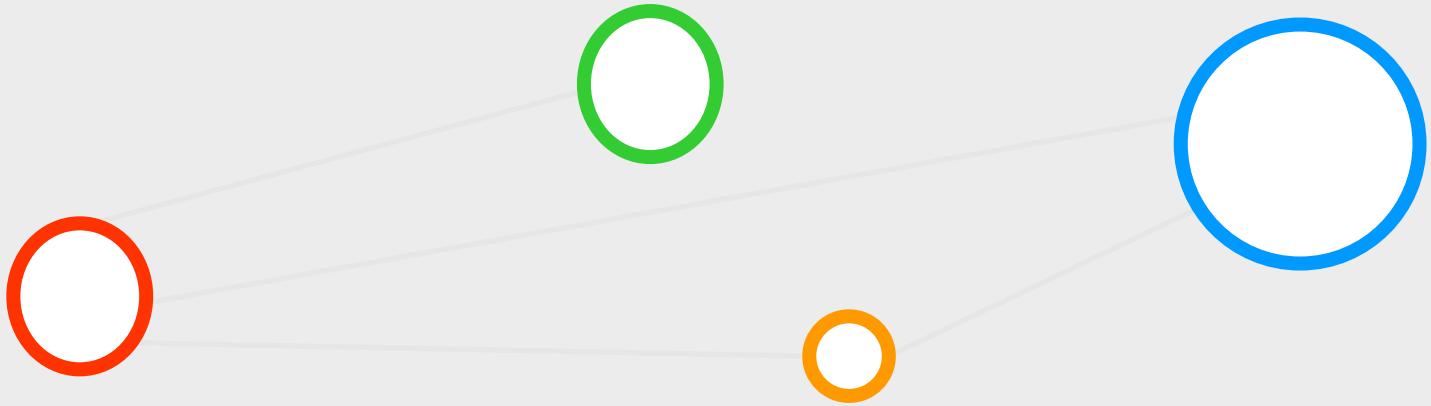


[3] Riedel, M., Cavallaro, G., Benediktsson, J.A.: Practice and Experience in using Parallel and Scalable Machine learning in Remote Sensing from HPC over Cloud to Quantum Computing, in conference proceedings of the IEEE IGARSS Conference, Brussels, Belgium, 2021

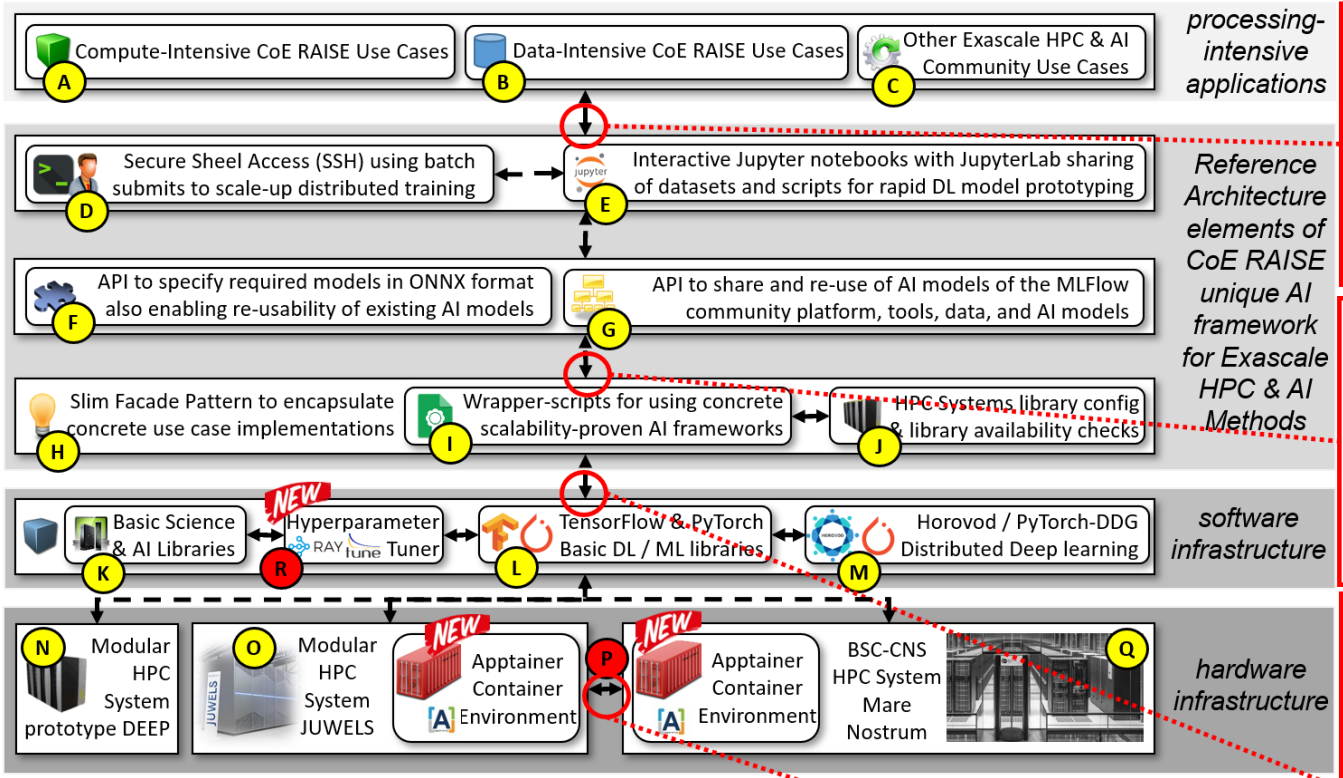
[2] Riedel, M., Sedona, R., Barakat, C., Einarsson, P., Hassanian, R., Cavallaro, G., Book, M., Neukirchen, H., Lintermann, A.: Practice and Experience in using Parallel and Scalable Machine learning with Heterogenous Modular Supercomputing Architectures, in conference proceedings of the IEEE IDPDS Conference, Heterogenous Computing Workshop (HCW), Portland, USA, 2021



CoE RAISE UAIF Evolution & Blueprint



Co-Design Evolution & Towards Realization of UAIF (V2)



✓ RQ1, RQ2, RQ4, RQ5

❖ Parts of the framework layout plan is to provide Kernels for Jupyter notebooks with correct version setups of modules for specific HPC Systems

✓ RQ3, RQ6 NEW

❖ Parts of the framework layout plan is to provide lightweight & abstract Python APIs building on ONNX enabling exchange with MLFlow, OpenML, ClearML, etc.

✓ RQ1, RQ2

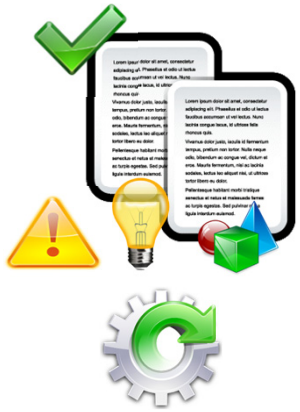
❖ Parts of the framework layout plan is to provide a lightweight Python API that abstracts from low level versioning of AI packages (with proven scalability) and is harmonized with different available HPC system module versions

✓ RQ6, RQ7

❖ Part of the framework layout plan is to provide containers in Apptainer with prepackaged datasets and required software stacks needed for AI models

Legend:

Tangible outputs of RAISE WP2 as part of the unique AI framework layout

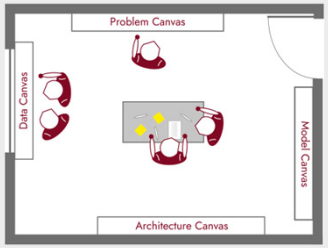


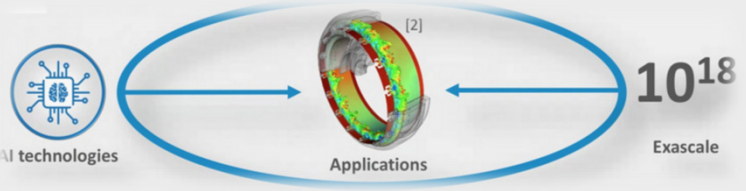
Continously Updating!

[4] M. Riedel, M. Book, H. Neukirchen, G. Cavallaro & A. Lintermann, "Practice and Experience using High Performance Computing and Quantum Computing to Speed-up Data Science Methods in Scientific Applications, MIPRO 2022

HPC Systems Engineering in the Interaction Room Seminar

- CoR RAISE Interaction Room Process as Next Step
  - Supports the proper software engineering design of the unique AI framework blueprint
  - Expecting to work with WP3 & WP4 experts in an open minded way
  - Process will be guided by **Prof. Dr. Matthias Book** (University of Iceland)
  - Supported by Software Engineering & testing expert **Prof. Dr. Helmut Neukirchen** (University of Iceland)
  - CoE RAISE @ YouTube
  - Methology as one CoE RAISE outcome

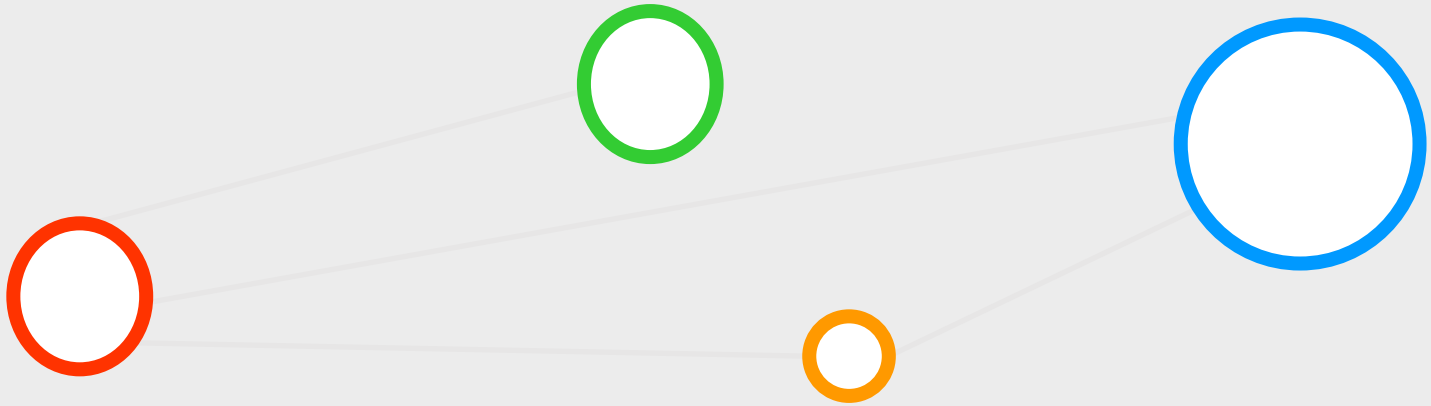




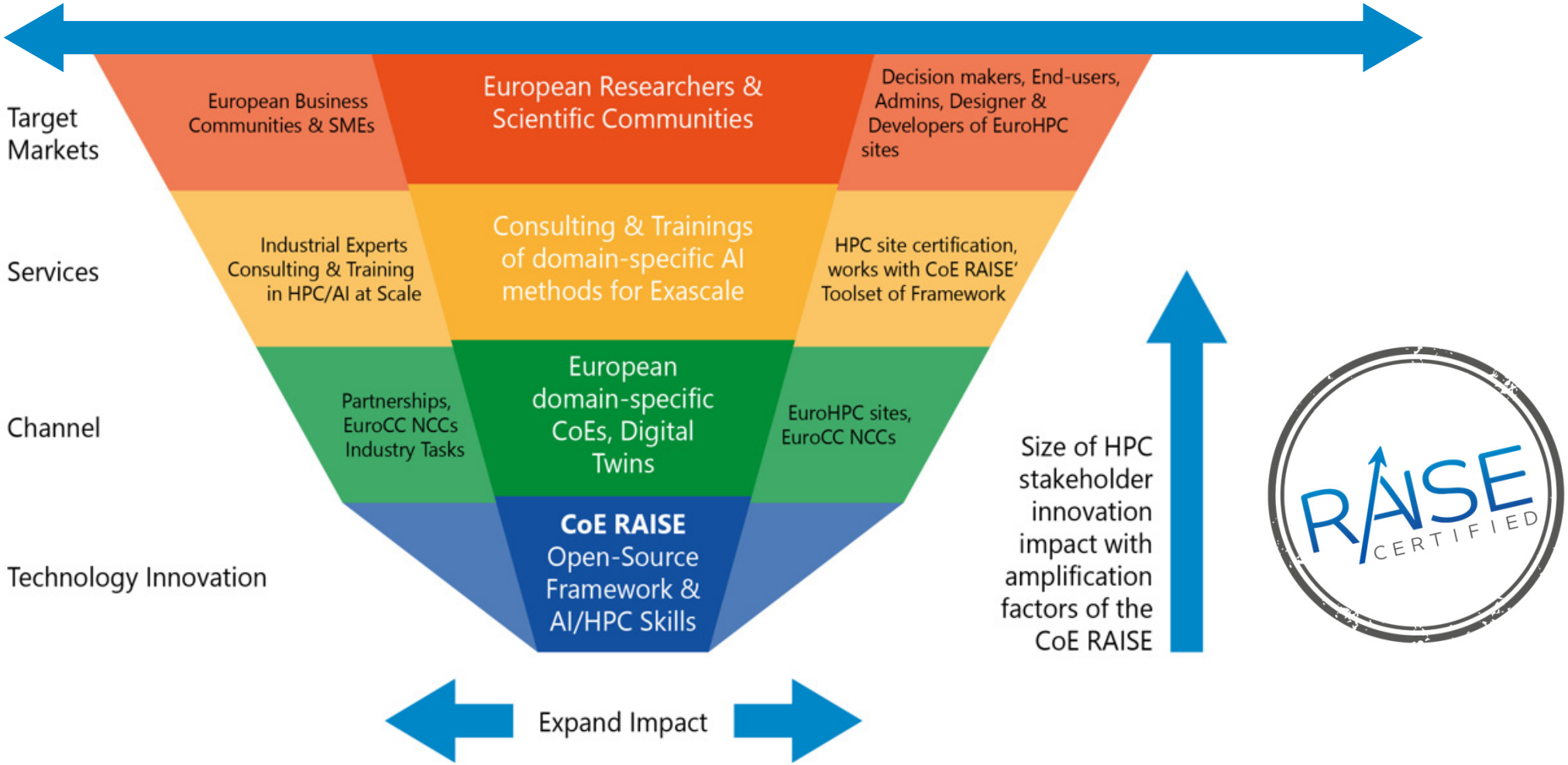
[1] Book, M., Riedel, M., Neukirchen, H., Goetz, M.: Facilitating Collaboration in High-Performance Computing Projects with an Interaction Room, in conference proceedings of the 4th ACM SIGPLAN International Workshop on Software Engineering for Parallel Systems (SEPS 2017), October 22-27, 2017, Vancouver, Canada

[5] Book, Riedel, Neukirchen, Erlingsson: Facilitating Collaboration in Machine Learning and High-Performance Computing Projects with an Interaction Room, International Workshop on Software Engineering for eScience, IEEE 2022

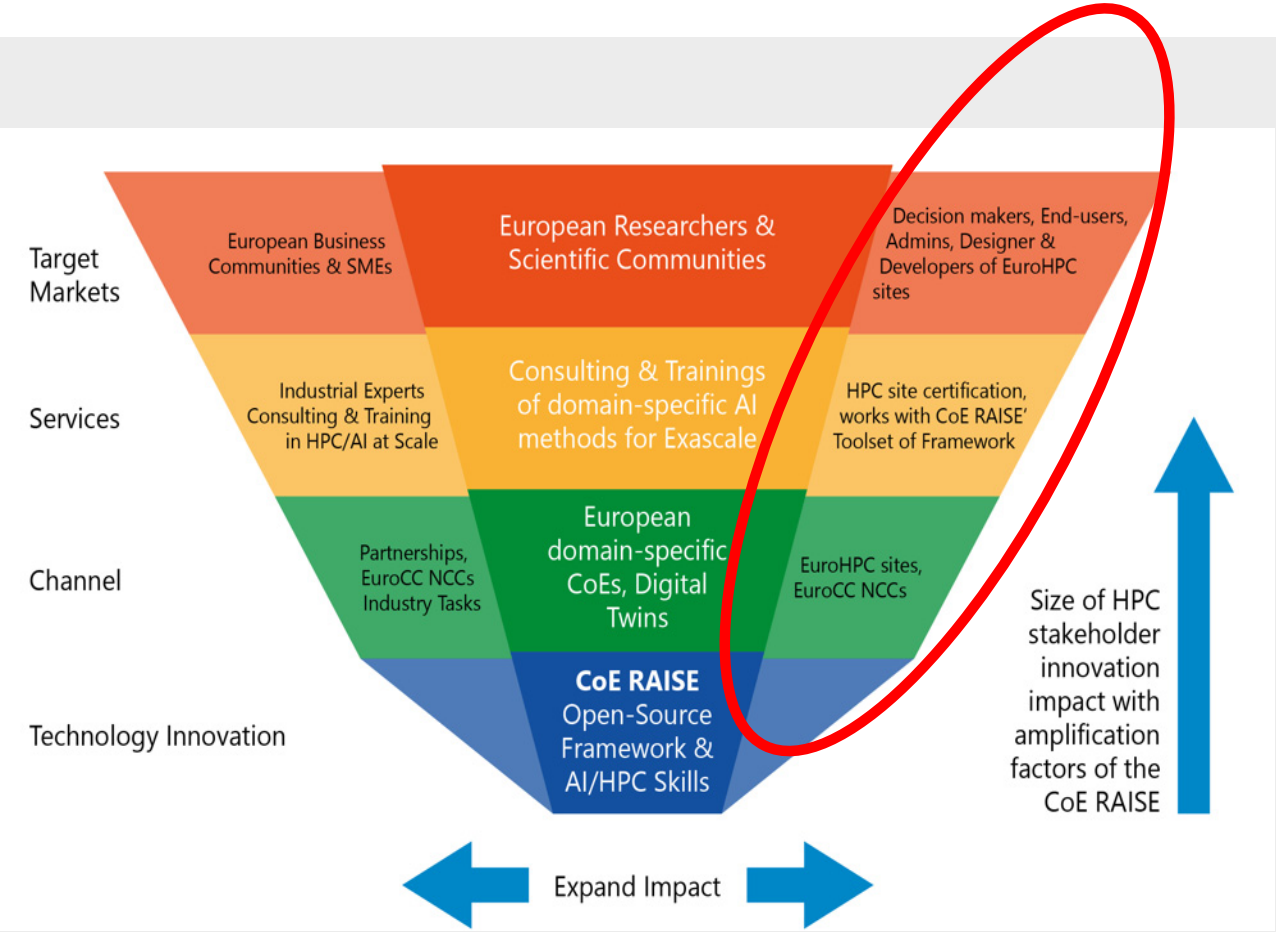
Adoption Roadmap of the Framework

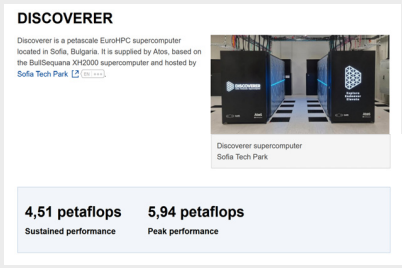


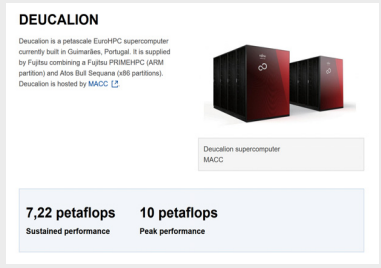
Impact: Enlarging Co-Design & Framework Adoption



Adoption Roadmap of the Framework

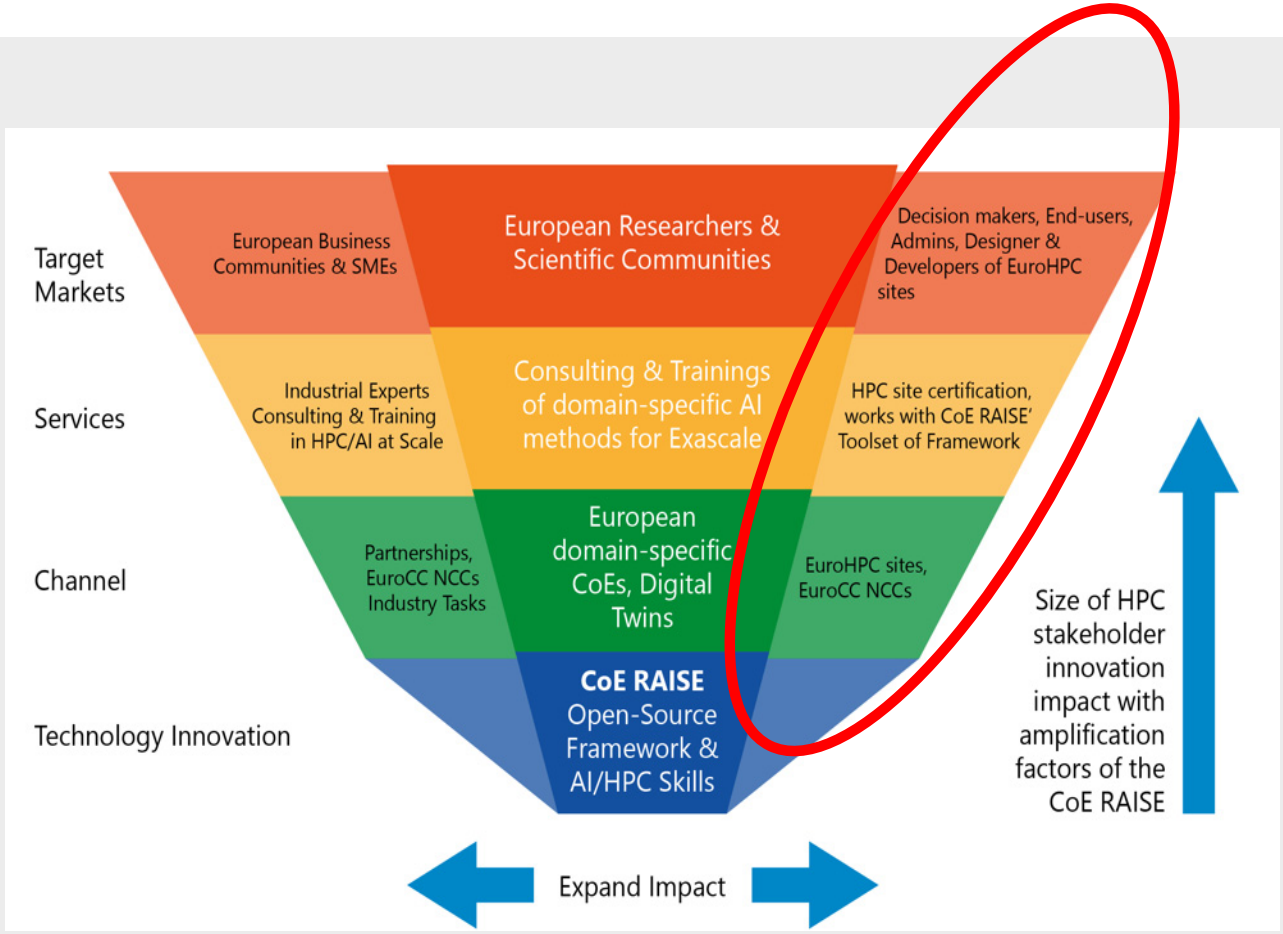


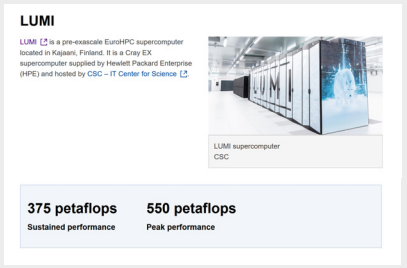


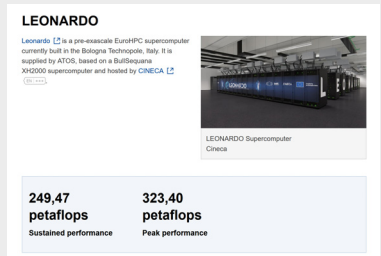


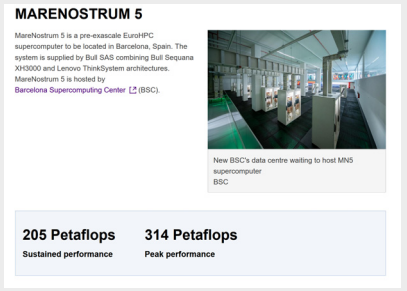
- This workshop goals: Feedback from different NCCs working with EuroHPC JU Hosting sites & AI activities
  - NCC Greece
  - NCC Cyprus
  - NCC Germany
  - NCC Iceland
  - NCC Czech Republic
  - Others?

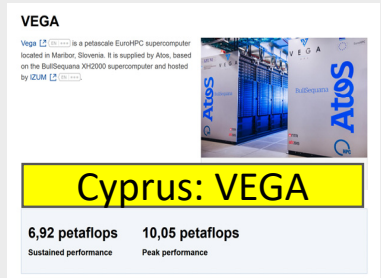
Adoption Roadmap of the Framework

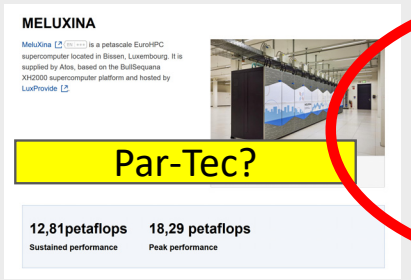




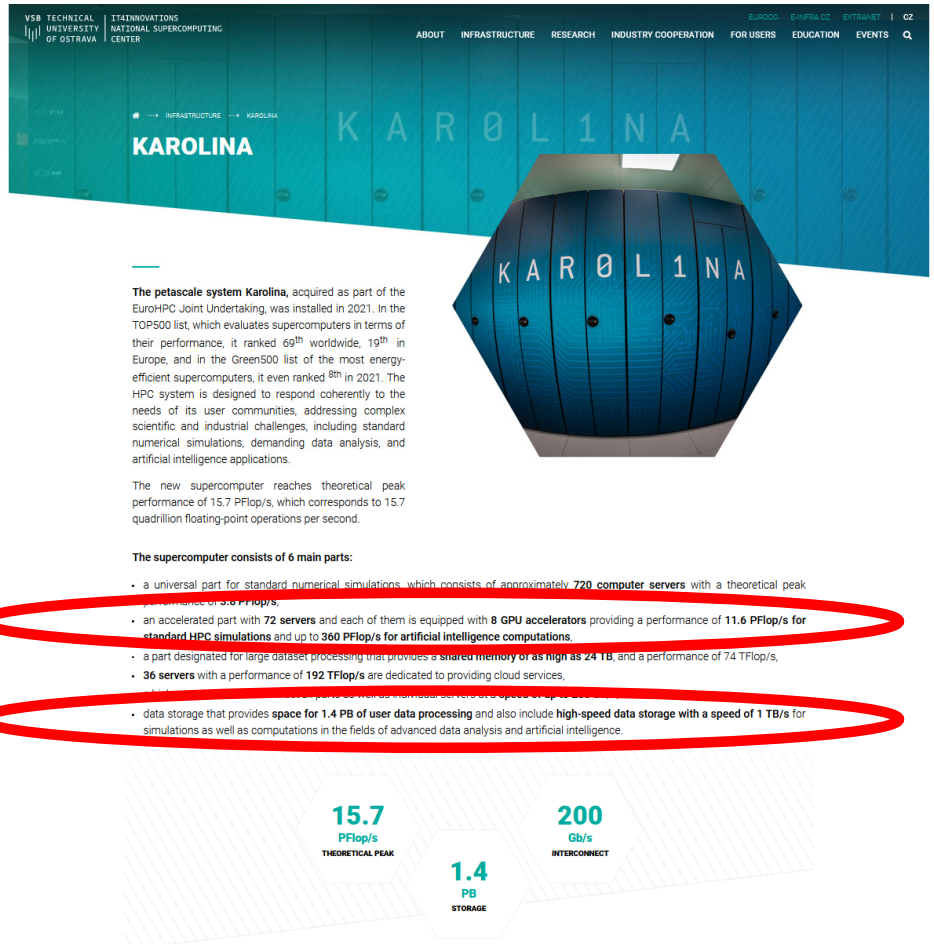


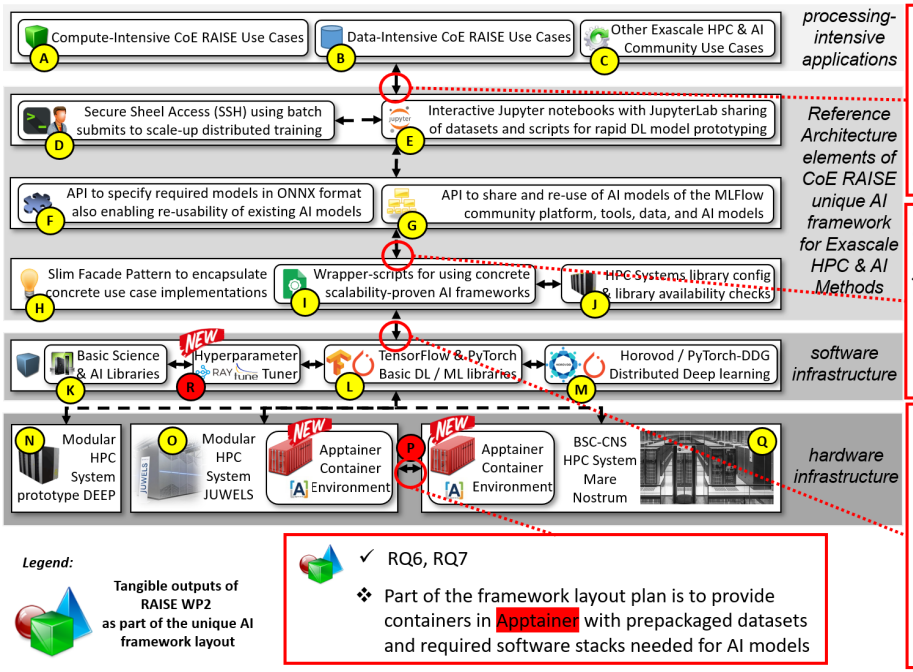






Adoption Roadmap of the Framework: KAROLINA Example





✓ RQ1, RQ2, RQ4, RQ5

❖ Parts of the framework layout plan is to provide Kernels for Jupyter notebooks with correct version setups of modules for specific HPC Systems

✓ RQ3, RQ6 NEW

❖ Parts of the framework layout plan is to provide lightweight & abstract Python APIs building on ONNX enabling exchange with MLFlow, OpenML, ClearML, etc.

✓ RQ1, RQ2

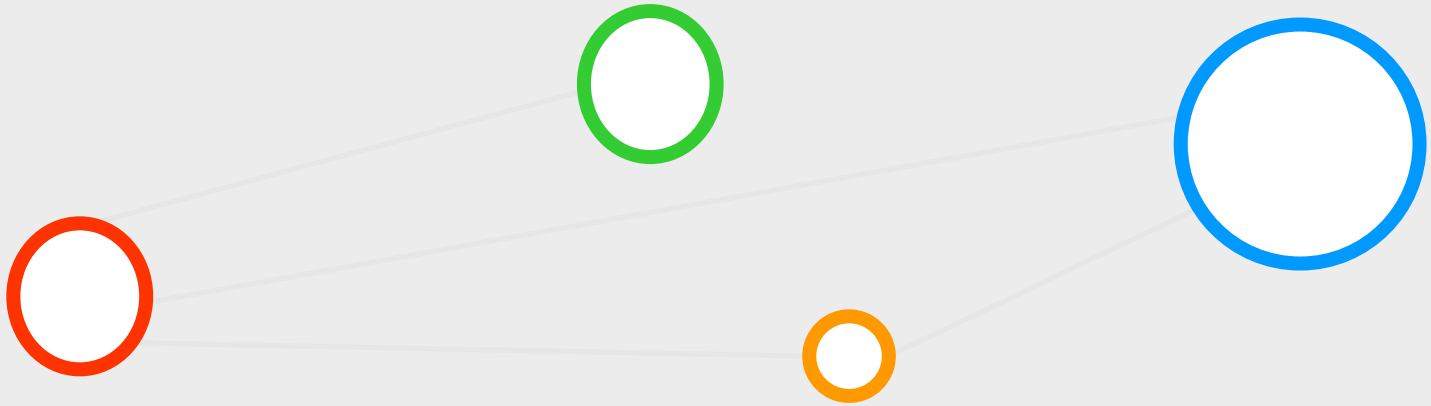
❖ Parts of the framework layout plan is to provide a lightweight Python API that abstracts from low level versioning of AI packages (with proven scalability) and is harmonized with different available HPC system module versions

✓ RQ6, RQ7

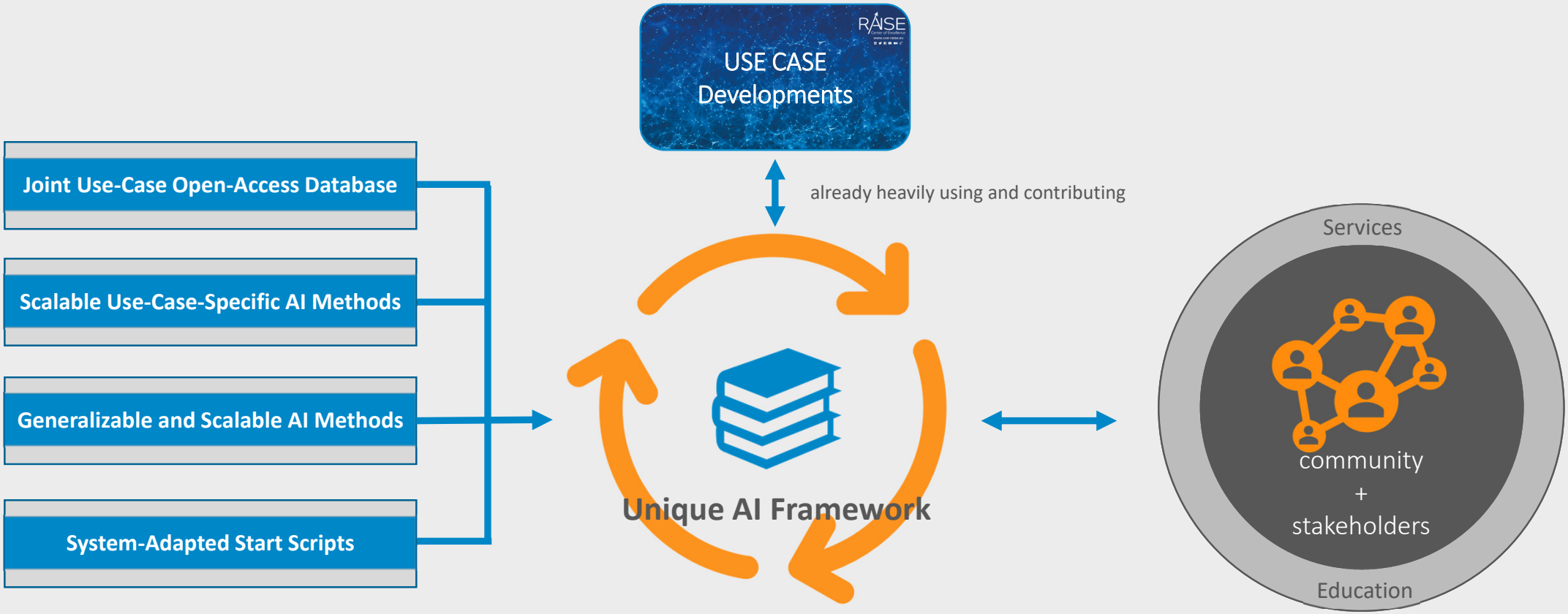
❖ Part of the framework layout plan is to provide containers in Apptainer with prepackaged datasets and required software stacks needed for AI models

- IT4Innovation
  - Vit Vondrak (not in workshop)
  - Tomas (hopefully in workshop)
  - TBD: Check what SW already installed & clarify access for CoE RAISE for scripts, etc.?

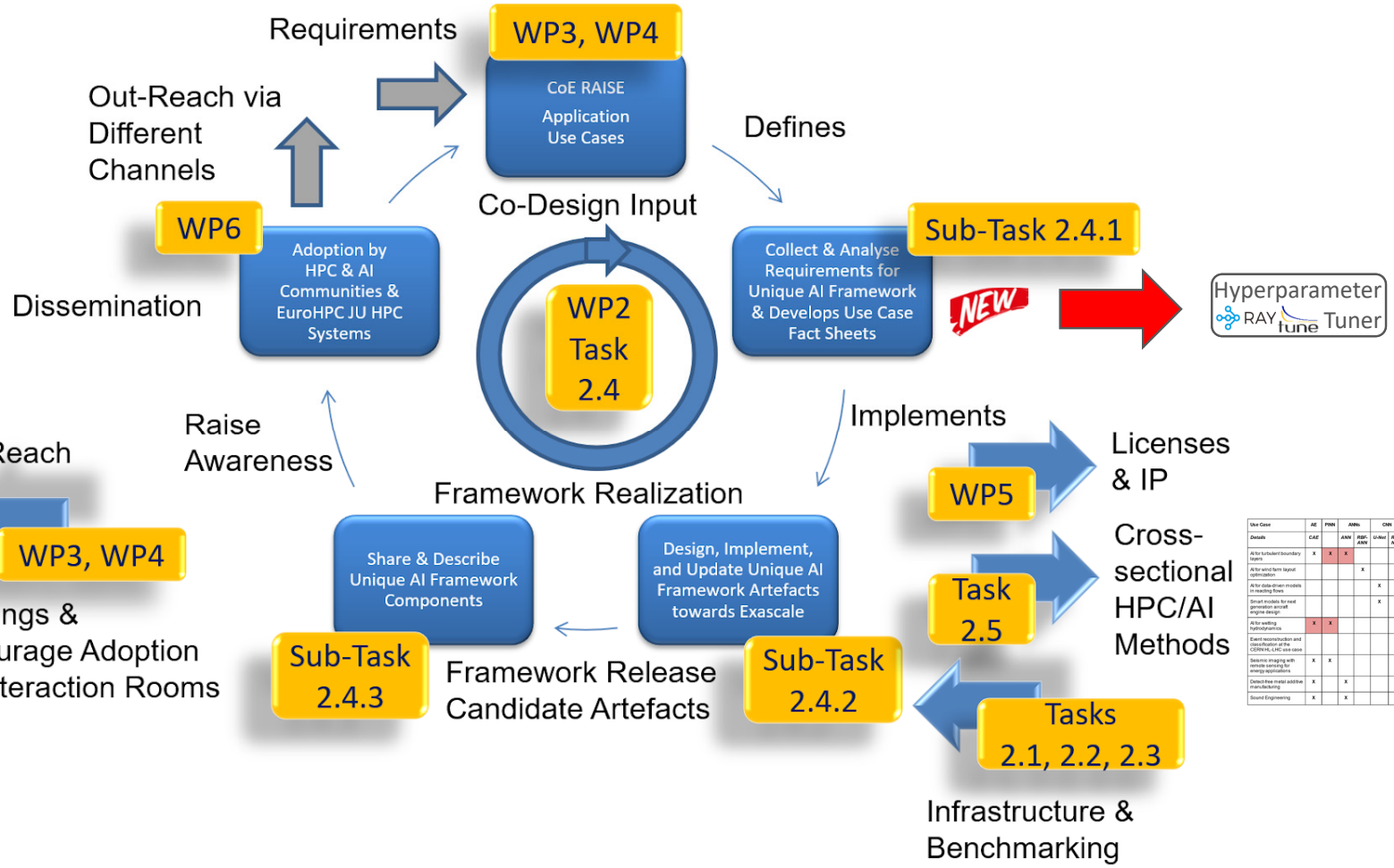
Summary & Q&A



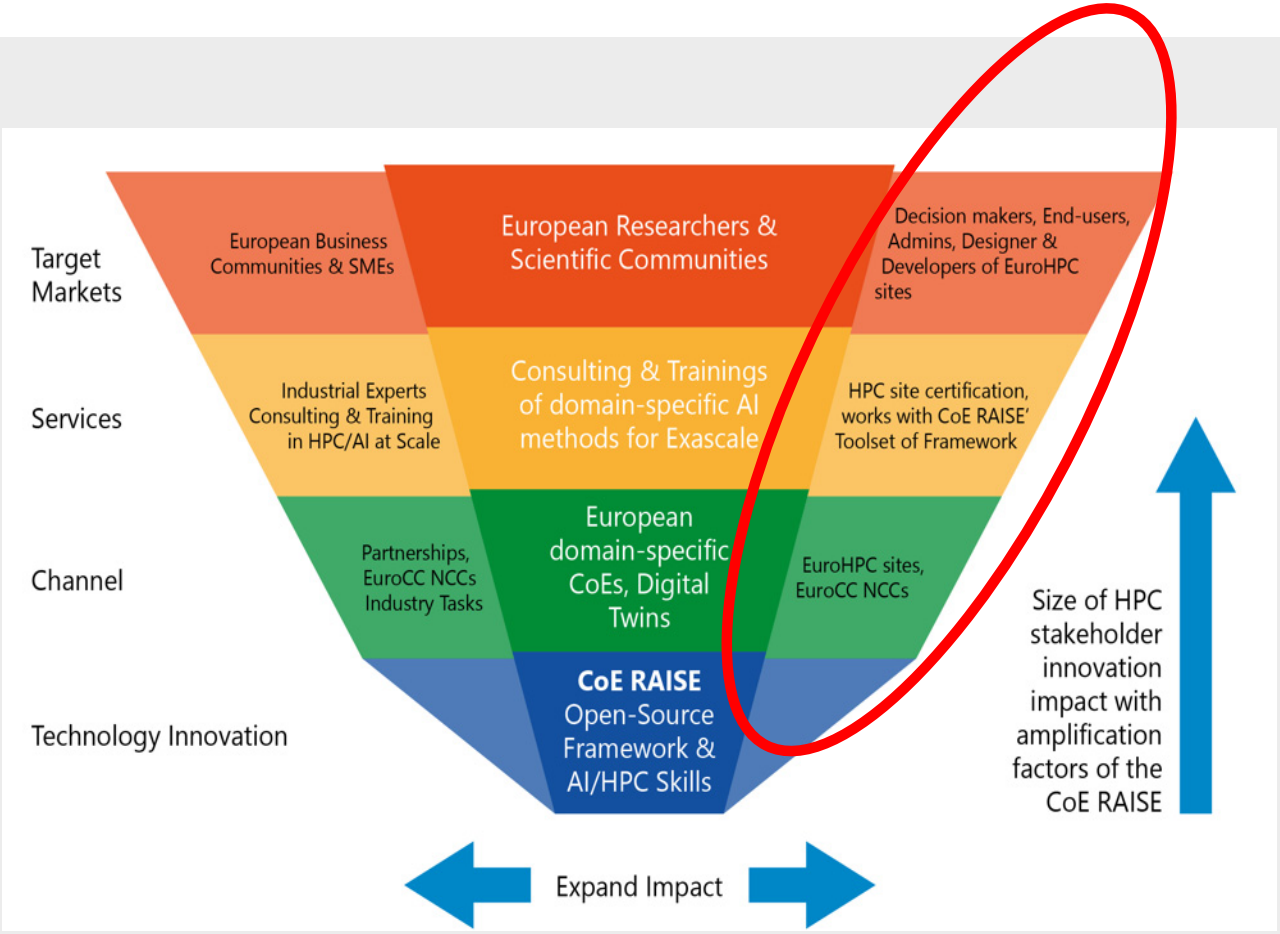
Summary: Unique AI Framework Overview

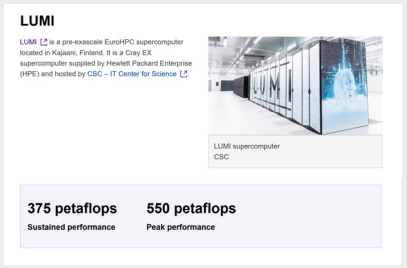


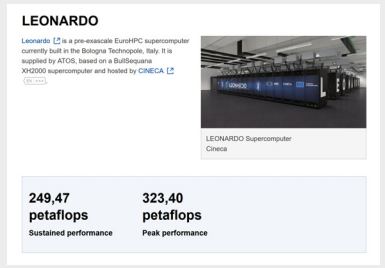
Summary of Framework Realization Process

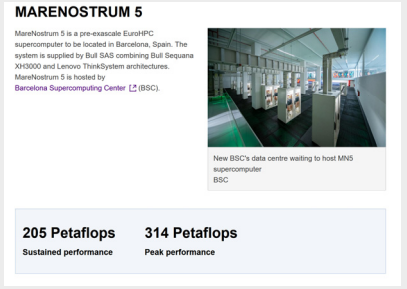


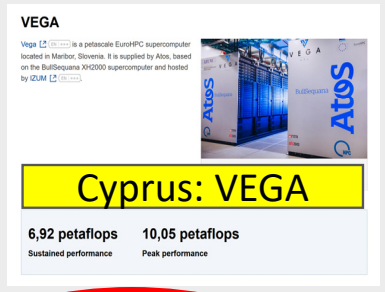
Q&A: NCC Feedback for Adoption?

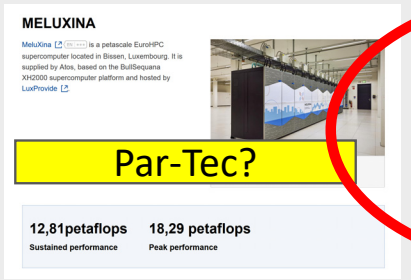


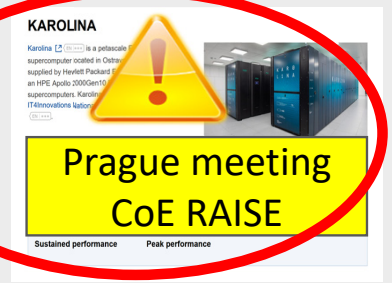




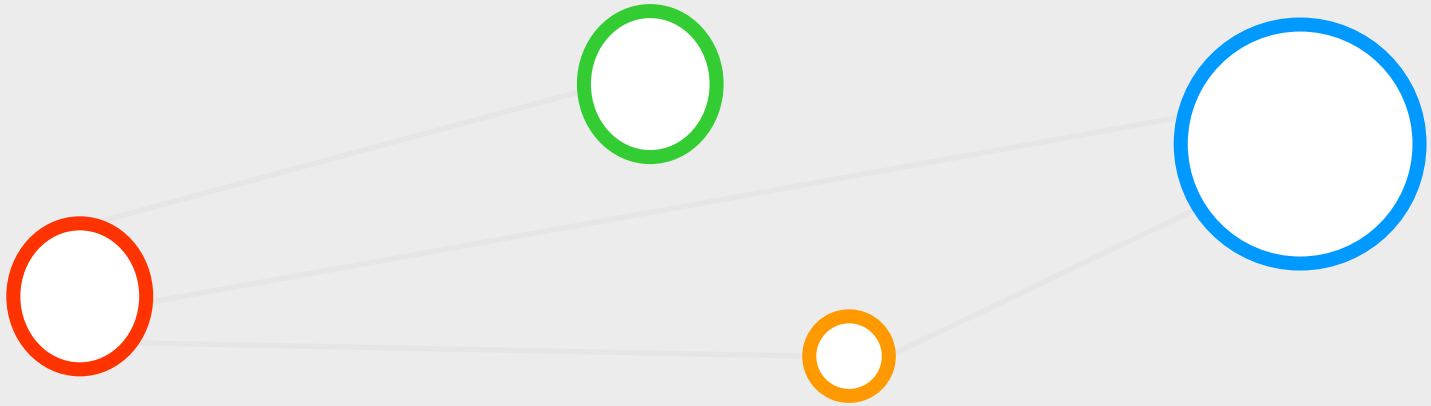








Selected References



Selected References

[1] Book, M., Riedel, M., Neukirchen, H., Goetz, M.: Facilitating Collaboration in High-Performance Computing Projects with an Interaction Room, in conference proceedings of the 4th ACM SIGPLAN International Workshop on Software Engineering for Parallel Systems (SEPS 2017), October 22-27, 2017, Vancouver, Canada,
Online: https://doi.org/10.1145/3141865.3142467

[2] Sedona, R., Barakat, C., Einarsson, P., Hassanian, Cavallaro, G., R., Book, M., Neukirchen, H., Lintermann, A. & Riedel, M. (2021). Practice and Experience in using Parallel and Scalable Machine Learning with Heterogenous Modular Supercomputing Architectures, 2021 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW),
Online: https://doi.org/10.1109/IPDPSW52791.2021.00019

[3] Riedel, M., Cavallaro, G., Benediktsson, J.A.: Practice and Experience in using Parallel and Scalable Machine learning in Remote Sensing from HPC over Cloud to Quantum Computing, in conference proceedings of the IEEE IGARSS Conference, Brussels, Belgium, 2021,
Online: https://doi.org/10.1109/IGARSS47720.2021.9554656

[4] M. Riedel, M. Book, H. Neukirchen, G. Cavallaro and A. Lintermann, "Practice and Experience using High Performance Computing and Quantum Computing to Speed-up Data Science Methods in Scientific Applications," *2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO)*, 2022, pp. 281-286,
Online: https://doi.org/10.23919/MIPRO55190.2022.9803802

[5] Book, Riedel, Neukirchen, Erlingsson: Facilitating Collaboration in Machine Learning and High-Performance Computing Projects with an Interaction Room, International Workshop on Software Engineering for eScience, IEEE 2022,
online: https://iris.rais.is/en/publications/facilitating-collaboration-in-machine-learning-and-high-performan

drive. enable. innovate.

RAISE Center of Excellence

The CoE RAISE project have received funding from
the European Union's Horizon 2020 –
Research and Innovation Framework Programme
H2020-INFRAEDI-2019-1 under grant agreement no. 951733

Follow us: