



UNIVERSITY OF ICELAND  
SCHOOL OF ENGINEERING AND NATURAL SCIENCES  
FACULTY OF INDUSTRIAL ENGINEERING,  
MECHANICAL ENGINEERING AND COMPUTER SCIENCE



# CoE RAISE Use Case Foundations & Lessons Learned from Fact Sheets

Prof. Dr. – Ing. Morris Riedel

School of Engineering & Natural Sciences, University of Iceland, Iceland  
Juelich Supercomputing Centre, Forschungszentrum Juelich, Germany

2021-04-08, RAISE CoE Seminar HPC Systems Engineering in the Interaction Room, Online



@ProfDrMorrisRiedel



@Morris Riedel



@MorrisRiedel



@MorrisRiedel

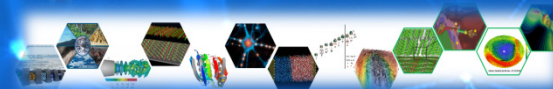


<https://www.youtube.com/channel/UCWC4VKHmL4NZgFfKoHtANKg>

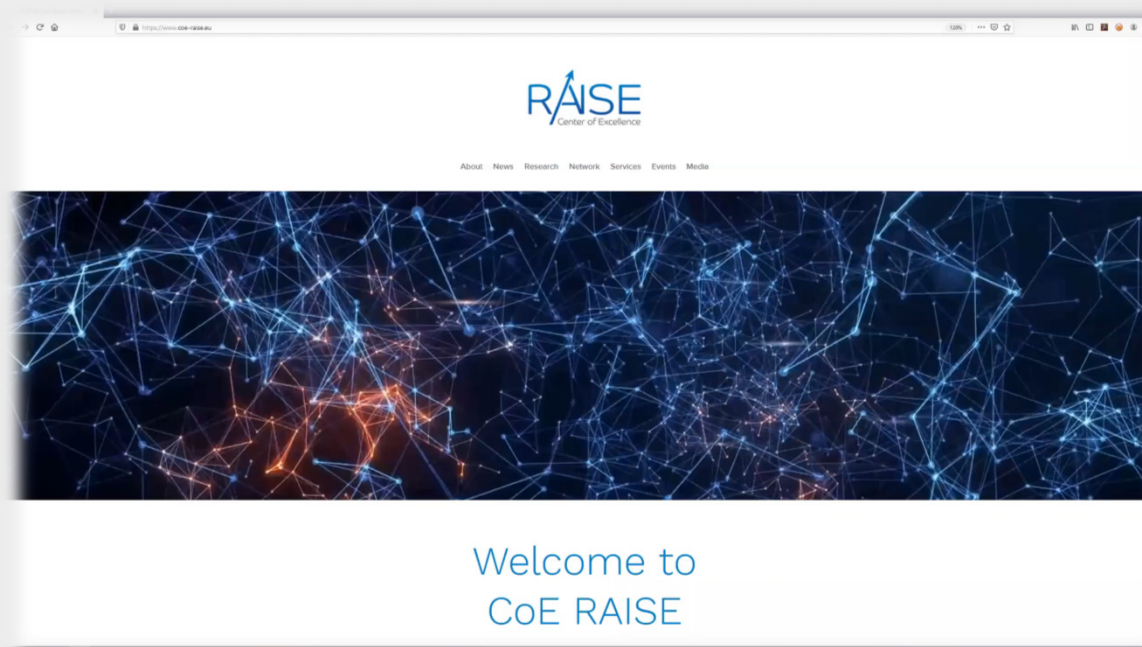
morris@hi.is



IHPC National Competence Center  
(NCC) for HPC & AI in Iceland

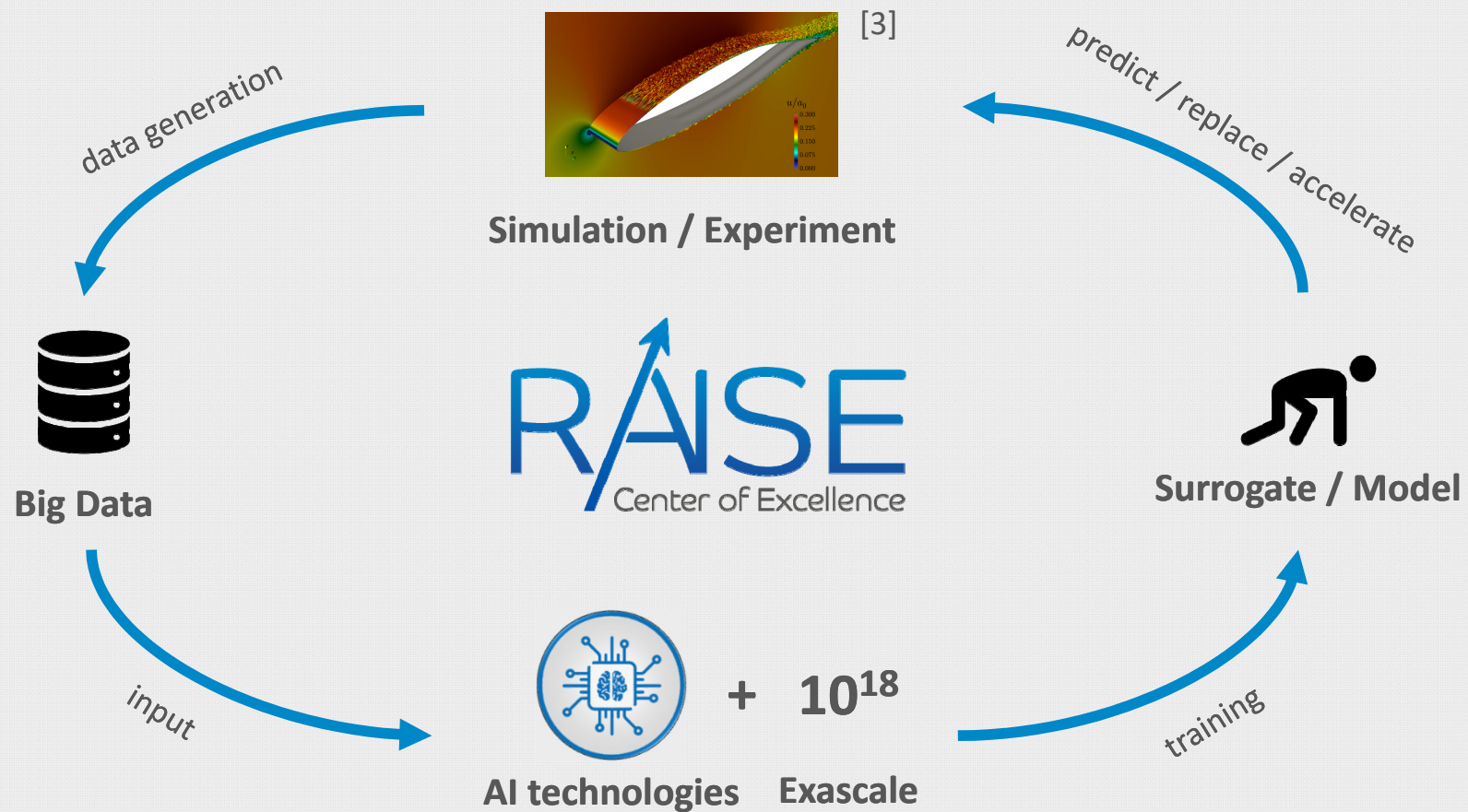


# CoE RAISE Web Page & More Information



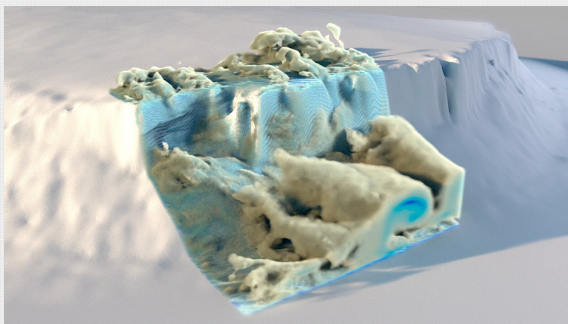
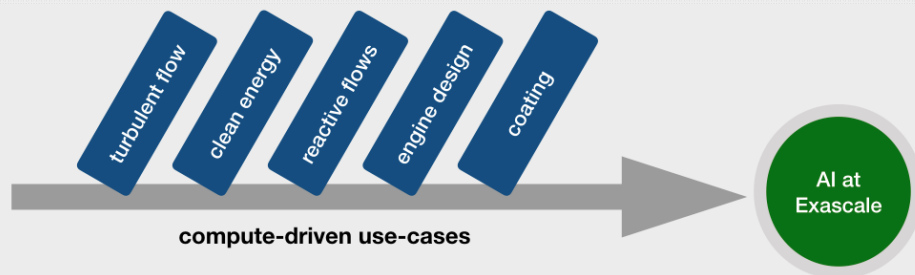
<https://www.coe-raise.eu>

# CoE RAISE – Motivation & Approach

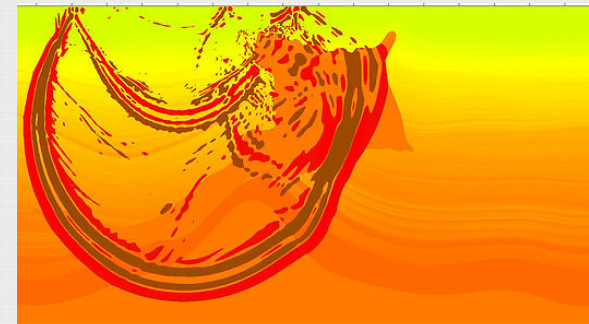
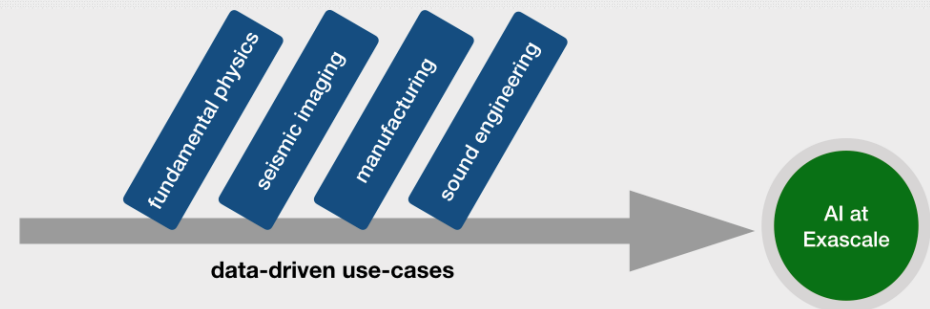


# Use Cases in CoE RAISE

## ➤ Two kinds of use cases:



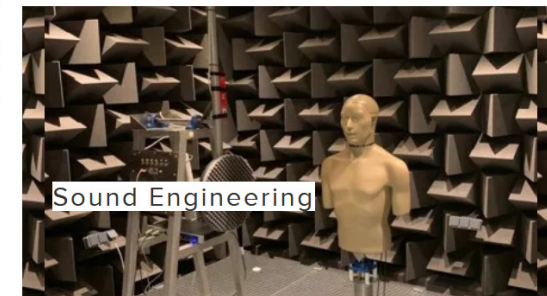
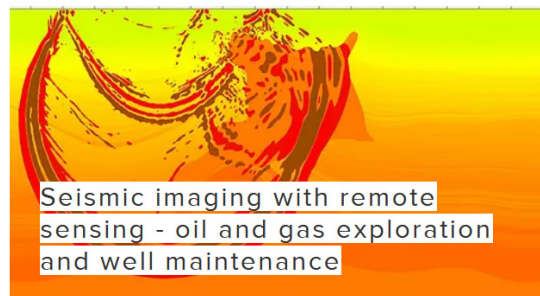
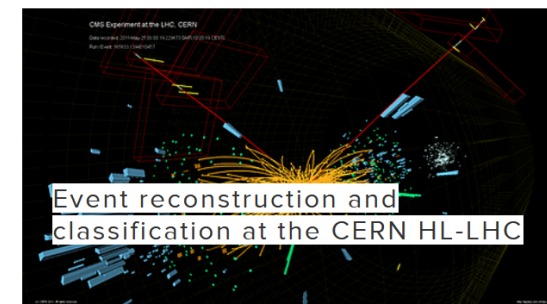
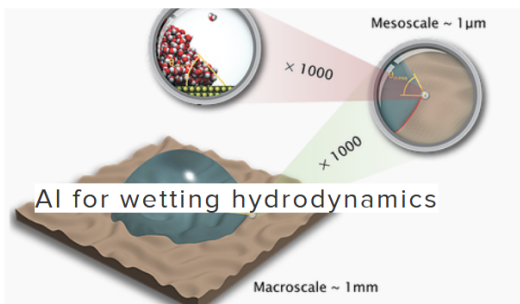
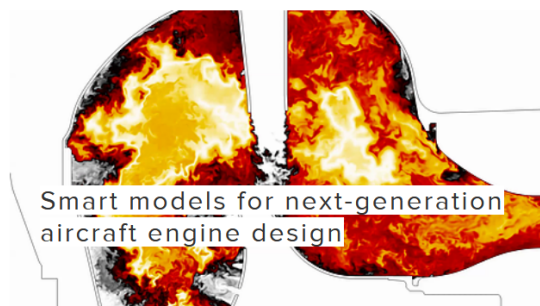
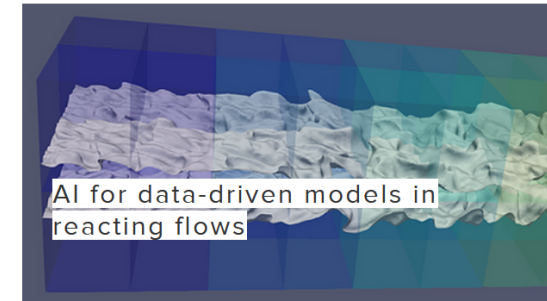
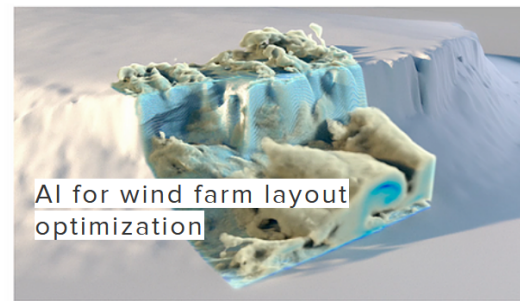
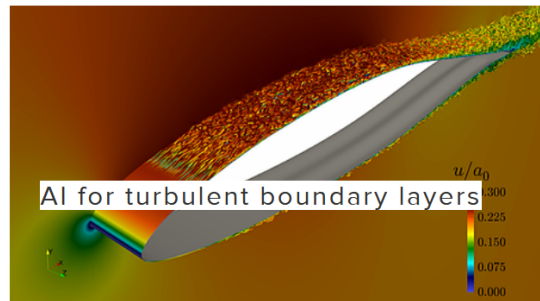
Example from use case "AI for wind farm layout": Turbulence generated by a cliff on Bolund Island, Denmark.



Example from use case "Seismic imaging with remote sensing - oil and gas exploration and well maintenance": Snapshot from a wavefield.



# Compute- and Data-driven Use Cases – Overview



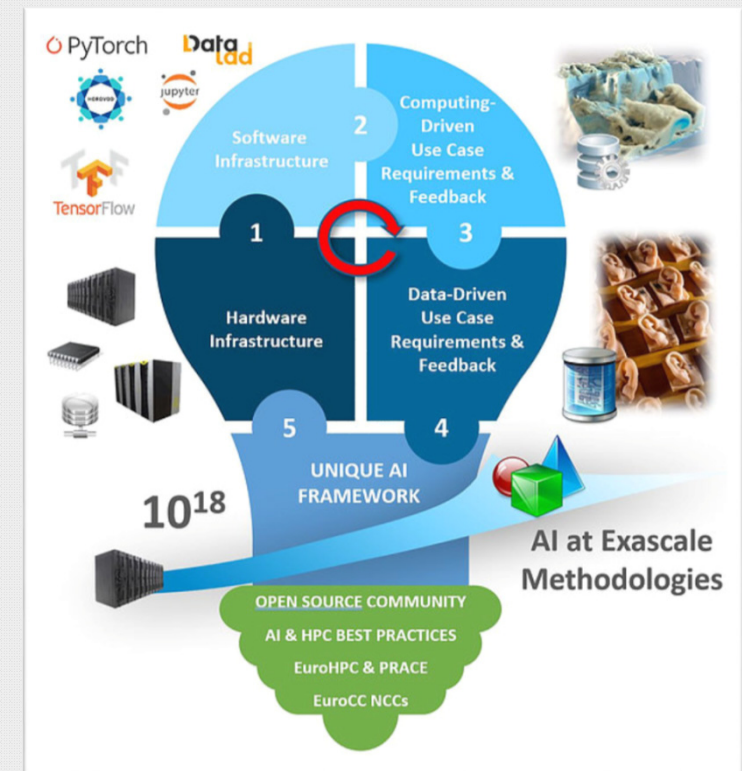
# Partners in CoE RAISE



# CoE RAISE's Objectives (1)

- Development of AI methods towards Exascale along use-cases
- RAISE tightly connects
  - an exceptional hardware infrastructure,
  - an usable and versatile software infrastructure,
  - compute-driven use cases,
  - and data-driven use cases

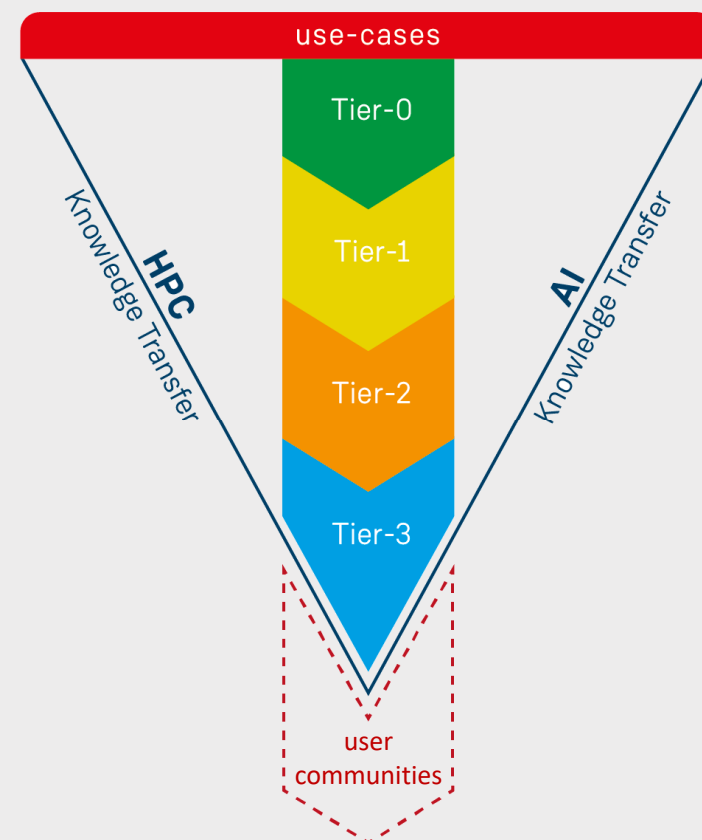
to contribute to a Unique AI framework that will be provided to academic and industrial communities (RAISE AI-Exascale library)





# CoE RAISE's Objectives

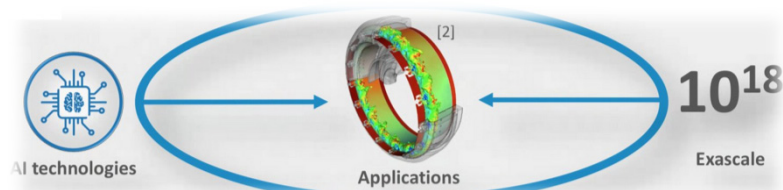
- Knowledge and technology transfer to science and industry
  - Service development and education
- Creation of a European network to support less developed entities
  - Equalize knowledge and technology across borders
- Business development to ensure sustainability
  - Market analysis and finding a market niche for the CoE
- Connect to others
  - Connect to other CoEs, Pre-Exascale projects, and other existing projects



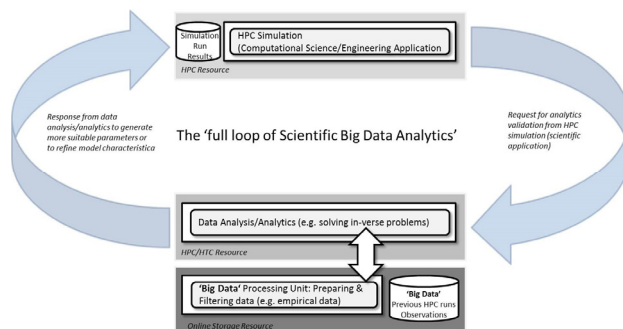


# Vision – Intertwined HPC Simulations & AI – ‘full loop’ ?

## ➤ What means AI & HPC Cross Methods at Exascale?



Today rather high performance data analytics (HPDA)

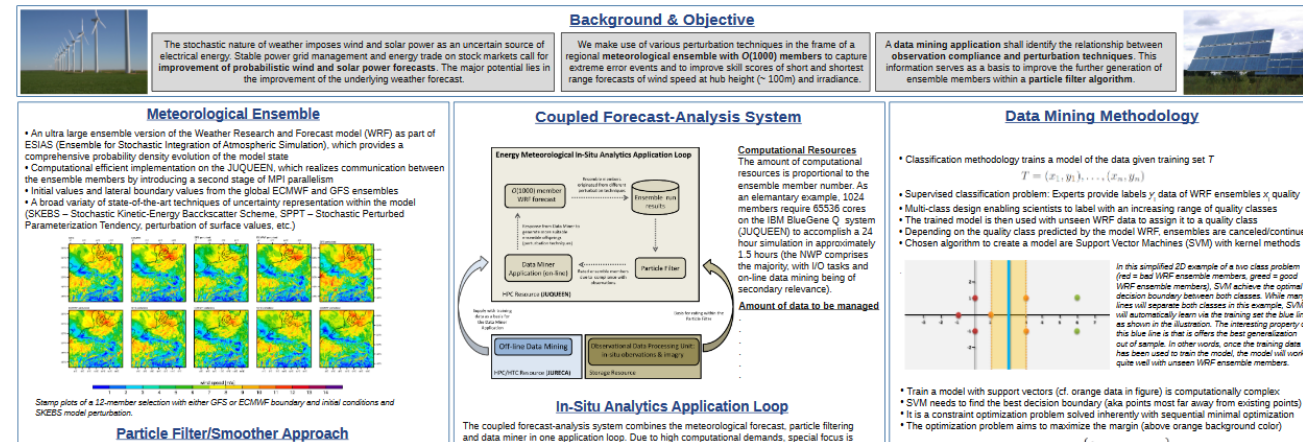


### Energy Meteorological In-Situ Big Data Analytics

Morris Riedel<sup>1,2</sup>, Jonas Berndt<sup>1,3</sup>, Charlotte Hoppe<sup>1,3</sup>, Hendrik Elbern<sup>1,3</sup>

contact: m.riedel@fz-juelich.de  
j.berndt@fz-juelich.de

<sup>1</sup>Institute of Energy and Climate Research (IEK-8), Forschungszentrum Jülich GmbH, Jülich, Germany  
<sup>2</sup>University of Iceland, Reykjavik, Iceland  
<sup>3</sup>Rhenish Institute for Environmental Research at the University of Cologne, Cologne, Germany

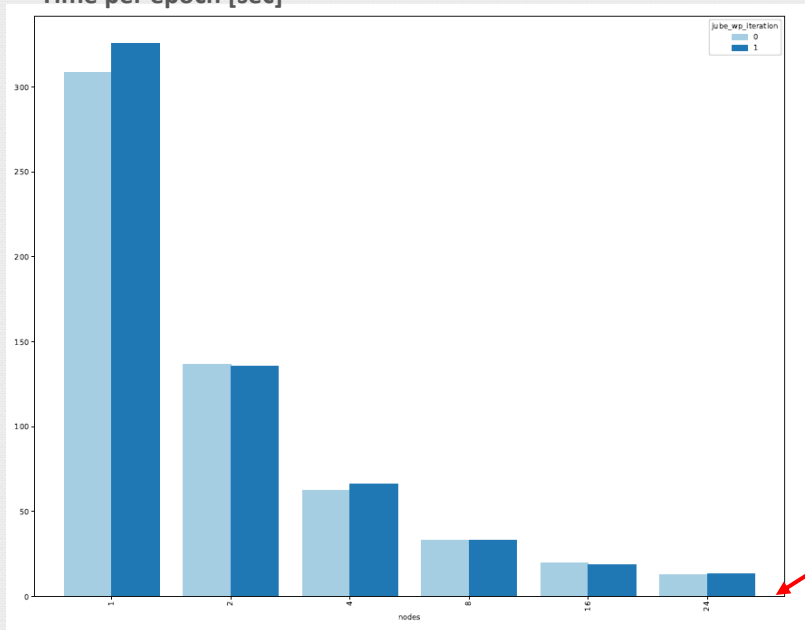


Lippert, T., Mallmann, D., Riedel, M.: **Scientific Big Data Analytics by HPC**, in Symposium proceedings of NIC Symposium 2016 – publication Series of the John von Neumann Institute for Computing (NIC), NIC Series 48 (417), ISBN 978-3-95806-109-5, February 11-12, 2016, Juelich, Germany

Riedel, M., Berndt, J., Hoppe, C., Elbern, H., Energy Meteorological In-Situ Big Data Analytics, Helmholtz Program Meeting, Karlsruhe Institute of Technology (KIT), July 1, 2016, Karlsruhe, Germany, [ [PDF \(~ 4,08 MB\)](#) ]

# Can AI do Exascale & use Disruptive Technologies?

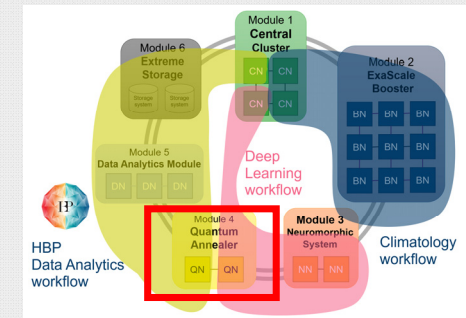
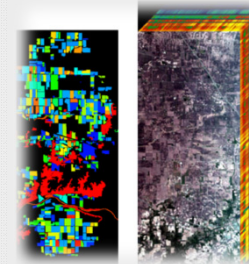
Time per epoch [sec]



Europe #1  
Supercomputer  
(11/2020)

Example from 2019:  
Using partition of the JUWELS  
system has 56 compute nodes,  
each with 4 NVIDIA V100 GPUs  
(equipped with 16 GB of memory)

24 nodes x 4 GPUs = 96 GPUs



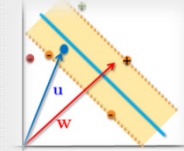
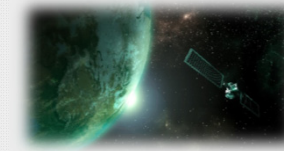
```
In [ ]: from quantum_SVM import *

# Hyperparameters
B=[2,3,5,10]
K=[2,3]
xi=[0.1,0.2]
gamma=[-1,0.125,0.25,0.5,1,2,4,8]
n_experiments=len(B)*len(K)*len(xi)*len(gamma)

hyperparameters=np.zeros([n_experiments,4], dtype=float)

path_data_keys='input_datasets/calibration/*id_dataset*/*'
data_key = 'id_dataset*calibrain*'
path_outs='outputs/calibration/*id_dataset*/*'

trainacc=np.zeros([fold], dtype=float)
trainauc=np.zeros([fold], dtype=float)
trainaup=np.zeros([fold], dtype=float)
```



Sedona, R., Cavallaro, G., Jitsev, J., Strube, A., **Riedel, M.**, Book, M.: **SCALING UP A MULTISPECTRAL RESNET-50 TO 128 GPUS**, in conference proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2020), September 26 – October 2nd, 2020, Virtual Conference, Hawaii, USA

Sedona, R., Cavallaro, G., Jitsev, J., Strube, A., **Riedel, M.**, Benediktsson, J.A.: **Remote Sensing Big Data Classification with High Performance Distributed Deep Learning**, Journal of Remote Sensing, Multidisciplinary Digital Publishing Institute (MDPI), Special Issue on Analysis of Big Data in Remote Sensing, 2019

Cavallaro, G., Willsch, D., Willsch, M., Michielsen, K., **Riedel, M.**: **APPROACHING REMOTE SENSING IMAGE CLASSIFICATION WITH ENSEMBLES OF SUPPORT VECTOR MACHINES ON THE D-WAVE QUANTUM ANNEALER**, in conference proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2020), September 26 – October 2nd, 2020, Virtual Conference, Hawaii, USA

# Towards AI & HPC at Exascale with CoE RAISE Results



## 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

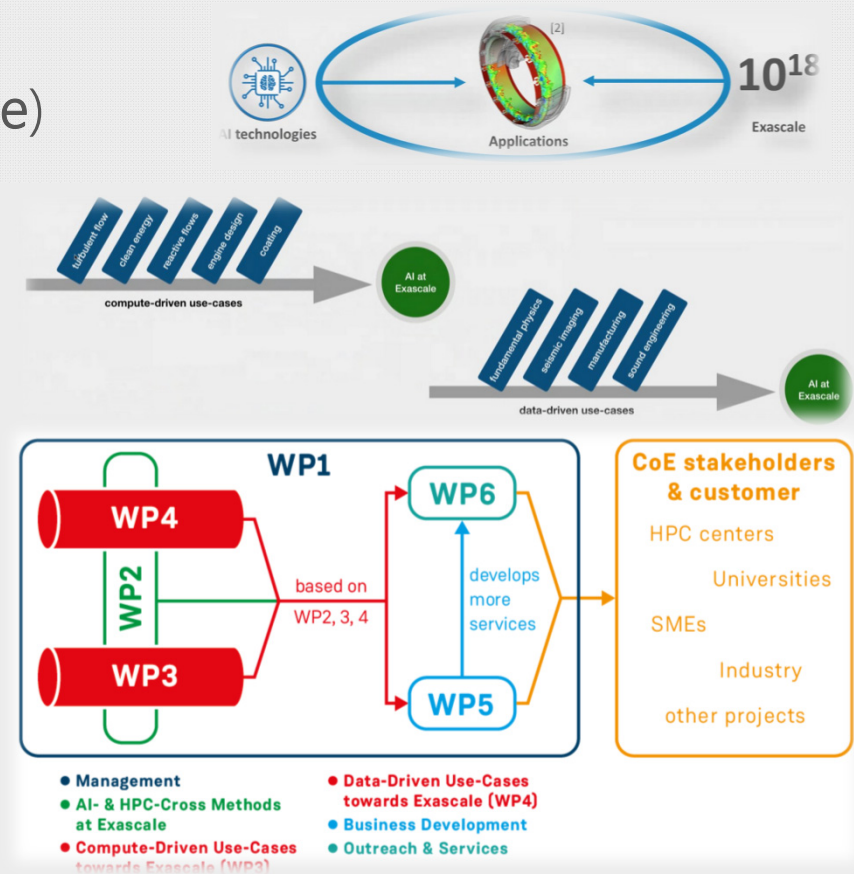
## → UNIQUE AI FRAMEWORK

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



# WP2 – AI- & HPC-Cross Methods at Exascale in a nutshell

- WP3 (Compute-Driven Use-Cases towards Exascale)
- WP4 (Data-Driven Use-Cases towards Exascale)
- Developments in these WPs will be supported by the cross-linking activities of WP2
  - E.g. scaling machine & deep learning codes with frameworks like Horovod/Deepspeed
  - E.g. introduction to new AI methods such as Long-Short Term Memory (Time series)
  - E.g. data augmentation approaches
  - E.g. benchmarking HPC machines and offer also pre-trained AI algorithms (i.e., transfer learning)
  - E.g. offer neural architecture search methods for hyperparameter – tuning in semi-automatic way

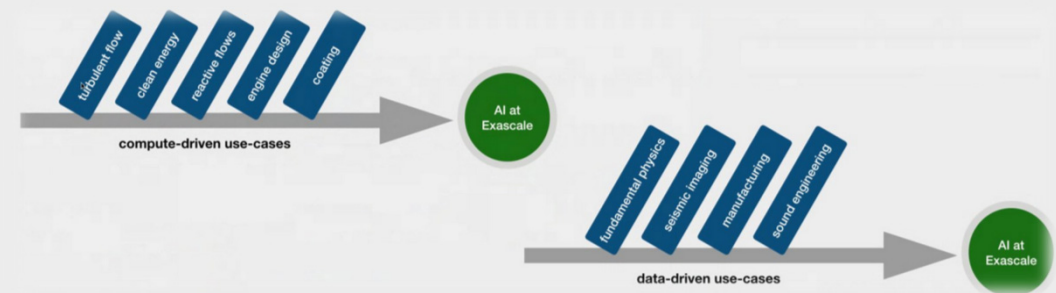




# Selected Techniques to Identify Cross-Methods for HPC & AI

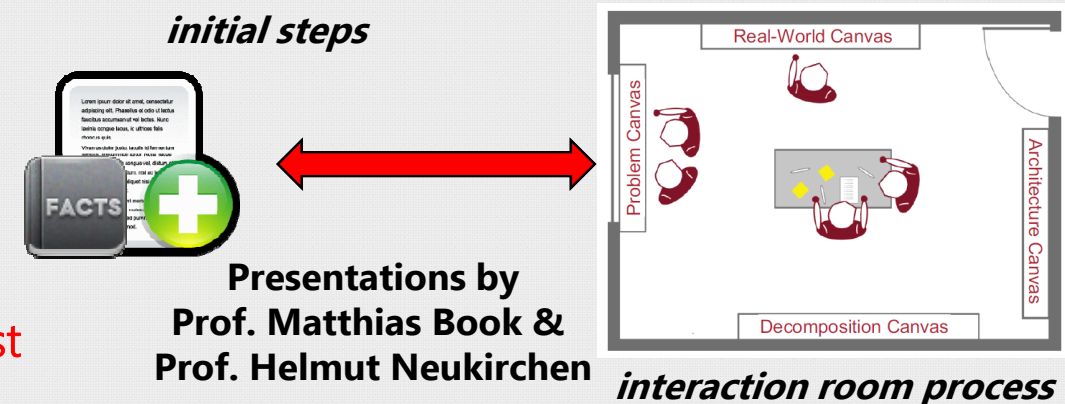
## ➤ Fact Sheets

- Foster initial understanding
- Living document & each Fact Sheet per WP3/WP4 Use Case
- *(Experience from many other EU projects)*

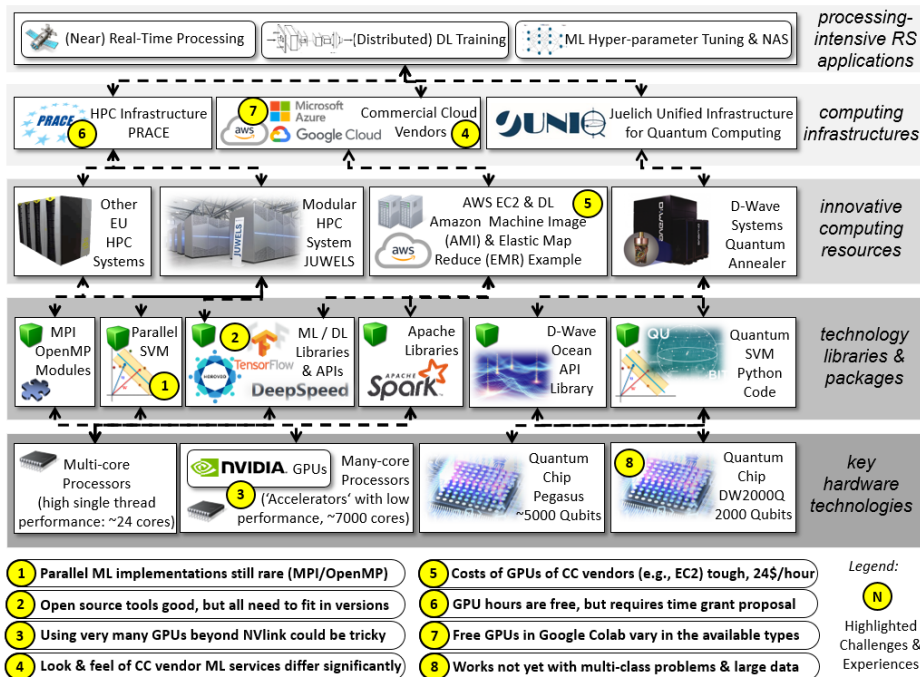


## ➤ Selected Contents

- 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**



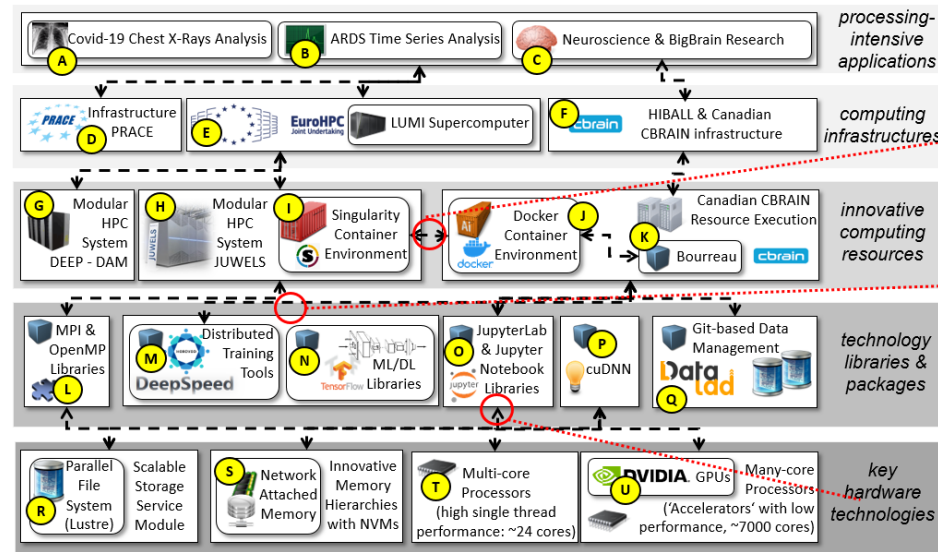
# Fact Sheet Process of CoE RAISE & Early Co-Design Examples



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, Physical and Online event, to appear <https://igarss2021.com/>



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 IDPS Conference, Heterogenous Computing Workshop (HCW), Portland, USA, 2021, Online, to appear <https://www.ipdps.org/>



**Some preparation**

```
$ mkdir winterschool_cache winterschool_tmp
$ cd winterschool_cache
$ export SINGULARITY_CACHE=$PWD -d -p "$(pwd)/winterschool_cache"
$ export SINGULARITY_TMPDIR=$(mktemp -d -p "$(pwd)/winterschool_tmp")
```

**Pull the docker image:**

```
$ cd winterschool
$ singularity pull hus.sif docker://glatond/hus
```

**Step into the container**

```
$ singularity shell --hus.sif
(the prompt changes to ~$singularity)
```

**download a dataset:**

```
$ git config --global user.name "Your name"
$ git config --global user.email "peturheigl@gmail.com"
$ singularity dataset install https://github.com/COMP-PCMD/comp-dataset.git
```

**ARDS Time Series Analysis**

Training and Validation Loss of the GNN model

**Covid-19 Chest X-Ray Analysis**

**Covid-Net**

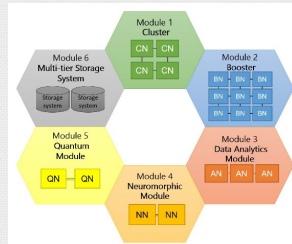
```
#!/bin/bash
# Load required modules
module purge
module use $SHENSTAGES
module load Stages/2020
module load GCCcore/5.3.0
module load Python/3.8.5
module load Tensorflow/1.15.1-Python-3.8.5
module load OpenCV/4.5.0-Python-3.8.5
# Activate python virtual environment
source ~/project/training2020/ingolfsson/jupyter/kernels/ingolfsson_kernels/activate
# Ensure python packages installed in the virtual environment are always preferred
export PYTHONPATH=$(pwd)/project/training2020/ingolfsson/jupyter/kernels/ingolfsson_kernels/lib
python -m jupyterlab
```



# Fact Sheet Process of CoE RAISE & Paper Example

## ➤ 'not a waste of time': Fact Sheets can be re-used for publications, project presentations & dissemination

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, Online, to appear <https://www.idpds.org/>



### Practice and Experience in using Parallel and Scalable Machine Learning with Heterogenous Modular Supercomputing Architectures

Morris Riedel  
Department of Computer Science  
University of Iceland  
Reykjavik, Iceland  
mori@hi.is

Rocco Sedona  
Jülich Supercomputing Centre  
Forschungszentrum Jülich  
Jülich, Germany  
r.sedona@fz-juelich.de

Chadi Barakat  
Jülich Supercomputing Centre  
Forschungszentrum Jülich  
Jülich, Germany  
c.barakat@fz-juelich.de

Petur Einarsson  
Department of Computer Science  
University of Iceland  
Reykjavik, Iceland  
petur@hi.is

Reza Hassanian  
Department of Computer Science  
University of Iceland  
Reykjavik, Iceland  
reh3@hi.is

Gabriele Cavallaro  
Jülich Supercomputing Centre  
Forschungszentrum Jülich  
Jülich, Germany  
g.cavallaro@fz-juelich.de

Mathias Book  
Department of Computer Science  
University of Iceland  
Reykjavik, Iceland  
book@hi.is

Andreas Lintermann  
Jülich Supercomputing Centre  
Forschungszentrum Jülich  
Jülich, Germany  
a.lintermann@fz-juelich.de

**Abstract**—We observe a continuously increased use of Deep Learning (DL) as a specific type of Machine Learning (ML) for data-intensive problems (i.e., 'big data') that require computing resources with equally increasing performance. Consequently, innovative heterogeneous High-Performance Computing (HPC) systems based on multi-core CPUs and many-core GPUs require an architectural design that addresses end user communities' requirements that take advantage of ML and DL. Still the workload of end user communities of the simulation sciences (e.g., using numerical methods based on known physical laws) needs to be equally supported by these architectures. This paper offers insight into the Modular Supercomputing Architecture (MSA) developed in the Dynamic Exascale Entry Platform (DEEP) series of projects to address the requirements of both simulation sciences and data-intensive sciences such as High-Performance Data Analytics (HPDA). It shares insights into implementing the MSA in the Jülich Supercomputing Centre (JSC) hosting Europe No. 1 Supercomputer Jülich Wizard for European Leadership Science (JUWELS). We suggest the technical findings with experience and lessons learned from two application communities case studies (i.e., remote sensing and health sciences) using the MSA with JUWELS and the DEEP systems in practice. Thus, the paper provides details into specific MSA design elements that enable significant performance improvements of ML and DL algorithms. While this paper focuses on MSA-based HPC systems and application experiences,

This work was performed in the Center of Excellence CoE RAISE on AI and Simulation-Based Engineering at Forschungszentrum Jülich, the Euro CC and DEEP-ASST projects receiving funding from EIT, Horizon 2020 Research and Innovation Framework Programme under the grant agreement no. 1010171, no. 951740 and no. 774384 respectively.

we are not losing sight of advances in Cloud Computing (CC) and Quantum Computing (QC) relevant for ML and DL.

**Index Terms**—High performance computing, cloud computing, quantum computing, machine learning, deep learning, parallel and distributed algorithms, remote sensing, health sciences, modular supercomputer architecture

**1. INTRODUCTION**

Today, an academically-driven supercomputing center's (e.g., Jülich Supercomputing Centre<sup>1</sup>, Barcelona Supercomputing Centre<sup>2</sup>, or Finnish IT Center for Science CSC<sup>3</sup>) application portfolio is highly multidisciplinary, raising diverse requirements for a HPC architecture that enables research for a wide variety of end-user communities (U). Examples include but are not limited to astrophysics, computational biology and biophysics, chemistry, earth and environment, plasma physics, computational soft matter, fluid dynamics, elementary particle physics, computer science and numerical mathematics, condensed matter, and materials science. Not only the research approaches in these communities are diverse, but also the way how they employ scalable algorithms, numerical methods, and parallelization strategies. Many of these are 'traditional HPC applications' (i.e., modeling and simulation sciences) that use iterative methods and rely heavily on a small number of

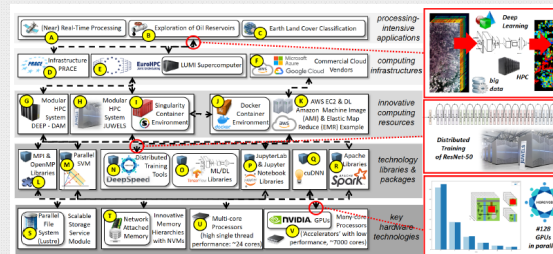


Fig. 2. Remote Sensing applications taking advantage of the MSA ensuring conceptual interoperability with Clouds.

DL is a necessary solution to train DL classifiers in a reasonable amount of time, providing RS researchers with a high-accuracy performance in the application recognition tasks. The same is true for the emerging HPC system landscape currently acquired by the EuroHPC Joint Undertaking as the LUMI supercomputer in Finland (see Fig. 2 E).

Our RS case study mainly takes advantage of the MSA-based JUWELS system (see Fig. 2) at the JSC in Germany, representing the fastest EU supercomputer with 122,768 CPU cores only in its cluster module (cf. Section 2.1 H). While JUWELS and multi-core processors (see Fig. 2 U) offer tremendous performance, the particular challenge to exploit this data analysis performance for ML is that those systems require specific parallel and scalable techniques. In other words, using JUWELS cluster module CPUs with Remote Sensing (RS) data effectively requires parallel algorithm implementations opposed to using plain scikit-learn<sup>4</sup>, R<sup>16</sup>, or different serial algorithms. Parallel ML algorithms are typically programmed using the MPI standard, and OpenMP (see Fig. 2 L) that jointly leverage the power of shared memory and distributed memory via low latency interconnects (e.g., InfiniBand<sup>17</sup>) and parallel filesystems (e.g., Lustre<sup>18</sup>).

Given our experience, the availability of open-source parallel and scalable machine learning implementations for the JUWELS cluster module CPUs that go beyond Artificial Neural Network (ANN) is or more recent DL networks (see Fig. 2 O) is still relatively rare. The reason is the complexity of parallel programming of ML and DL codes and thus using HPC with CPUs only can be a challenge when the amount of data

is relatively moderate (i.e., DL not always successful). One example is using a more robust classifier such as a parallel and scalable Support Vector Machine (SVM) open-source package (see Fig. 2 M) that we developed with MPI for CPUs and used to speed up the classification of RS images [16].

### 3.1. Selected DL Experiences on MSA-based Systems

The many-core processor approach of the highly scalable JUWELS booster (see Section 2.2) with accelerators brings many advancements to both simulation sciences and data sciences, including innovative DL techniques. Using many numerous simpler processors with hundreds to thousands of independent processor cores enabled a high degree of parallel processing that fits very nicely to the demands of DL training whereby lots of matrix-matrix multiplications are performed. Today, hundreds to thousands of accelerators like Nvidia GPUs (see Fig. 2 V) are used in large-scale HPC systems, offering unprecedented processing power for RS data analysis. JUWELS Booster module offers 3744 GPUs of the most recent innovative type of Nvidia A100 tensor core<sup>19</sup> cards. Our experience on MSA-based systems such as DEEP<sup>20</sup> (see Fig. 2 G), JURECA<sup>21</sup>, and JUWELS shows that open-source DL packages such as TensorFlow<sup>22</sup> (now including Keras<sup>23</sup>) or PyTorch<sup>24</sup> are powerful tools for large-scale RS data analysis.

We experienced that it can be quite challenging to have the right versions of python code matching the available DL

<sup>16</sup><https://www.scikit-learn.org/stable/>  
<sup>17</sup><https://www.mellanox.com/products/infiniband-overview>  
<sup>18</sup><https://www.lustre.org/>  
<sup>19</sup><https://www.nvidia.com/en-us/data-center/a100/>  
<sup>20</sup><https://www.fz-juelich.de/ias/jsc/INFOS/press/Supercomputer-DEEP-EST-2020.html>  
<sup>21</sup><https://www.fz-juelich.de/ias/jsc/INFOS/press/Supercomputer-JURECA-AT-RECA-2020.html>  
<sup>22</sup><https://www.tensorflow.org/>  
<sup>23</sup><https://keras.io/>  
<sup>24</sup><https://pytorch.org/>

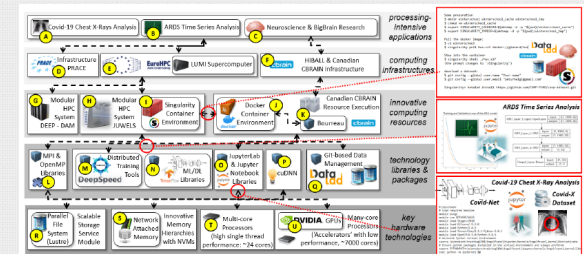


Fig. 3. Health science applications taking advantage of the MSA enabling seamless access for non-technical medical experts.

latest cuDNN support (see Fig. 3 P) the inference and training time of the Covid-Net model is significantly faster as with GPUs of the previous generation given its tensor cores.

We used several publicly available datasets of COVID-19 [25] that is an open access benchmark dataset initially comprising of 13,975 CXR images across 13,870 patient patient cases. But in the last couple of month this dataset was extended numerous times with new datasets made available that in turn we used again with Covid-Net as well. The SSSM of the MSA systems and its parallel file system Lustre (see Fig. 3 R) provides a powerful storage mechanism to store the COVID-19 datasets and its updates.

This module also stores additional data we obtained from a collaborating Pharma company that we in turn used to validate that Covid-Net is able to generalize well to unseen datasets. At the time of writing, the name and dataset of the collaborating pharma company can not be revealed, but will be made available to the workshop organizers during the workshop. Using the MSA-based systems JUWELS and DEEP seamlessly with Jupyter requires the definition of an own Kernel<sup>27</sup> using the module<sup>28</sup> environment of the MSA HPC systems (see Fig. 3 bottom right). Our experience on using own Kernels with Jupyter notebooks is extremely positive while at the same time offering a user interface with notebooks that are user-friendly enough for medical imaging experts.

<sup>27</sup><https://jupyter.org/environments>  
<sup>28</sup><https://github.com/jupyter/notebooks/blob/master/001-JupyterCreate-JupyterKernel-general.ipynb>  
<sup>29</sup><https://github.com/jupyter/notebooks/blob/master/001-JupyterCreate-JupyterKernel-general.ipynb>

### 4.2. Time Series Data Analysis of ARDS Patients

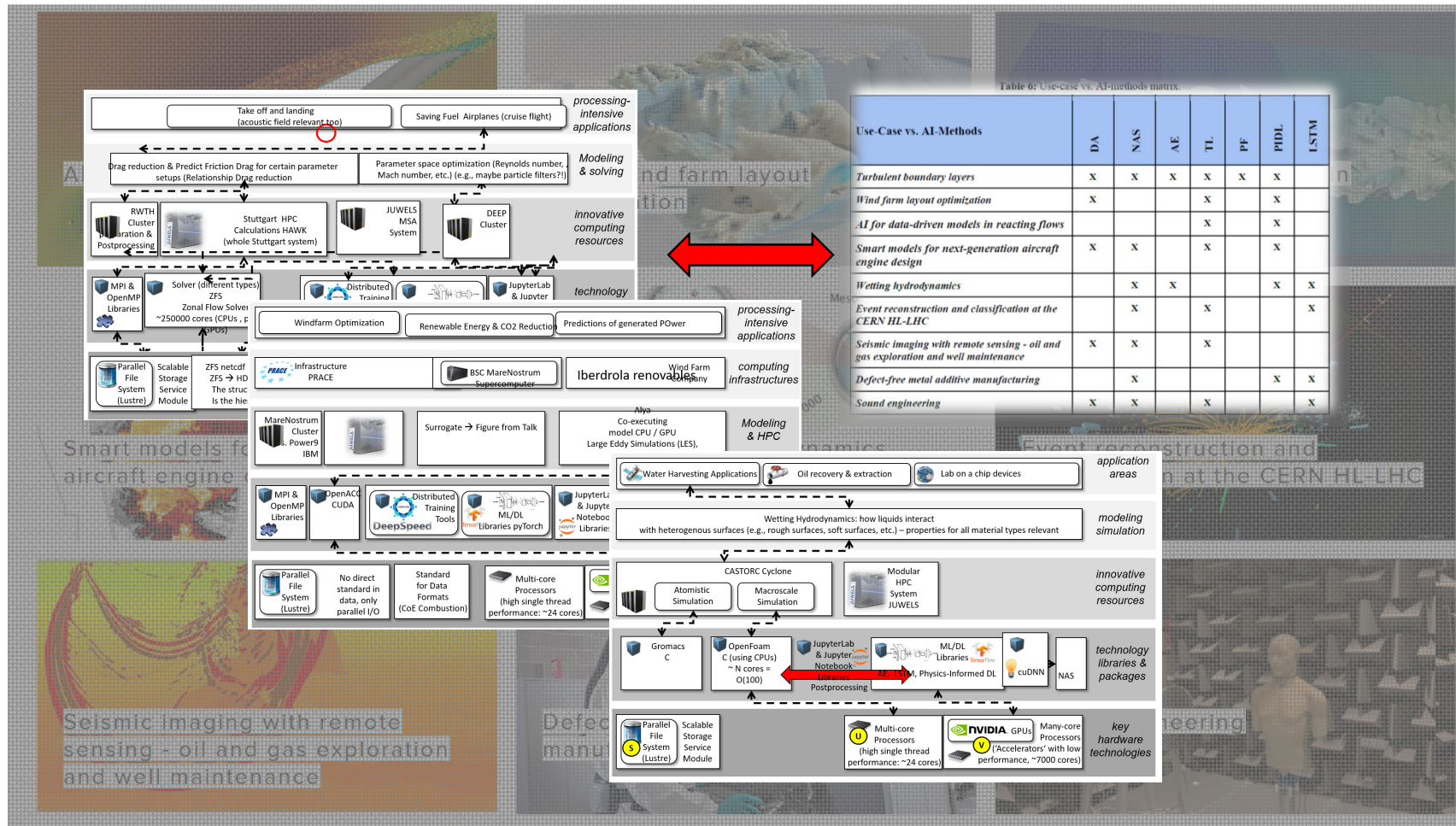
Our application case study 'ARDS Time Series Analysis' (see Fig. 3 A) addresses the medical condition Acute Respiratory Distress Syndrome (ARDS) that affects on average 1-2% of mechanically-ventilated (MV) Intensive Care Unit (ICU) patients and has a 40% mortality rate [26, 27]. At present the leading protocol for diagnosing the condition is the Berlin definition that defines onset of ARDS as a prolonged ratio of arterial oxygen potential to fraction of inspired oxygen (P/F ratio) of less than 300 mmHg, and the lower this value is determined to be, the more severe the diagnosis is [28]. Several papers have determined a correlation between early detection of onset of ARDS and survival of the patient, which highlights the need of early detection and treatment of the condition, before onset of sepsis and subsequently multi-organ failure [27, 29, 30]. Hence, the goal of this case study is to develop an algorithmic approach that provides early warning and informs medical staff of mitigating procedures can be a beneficial tool for ICU personnel.

We take advantage of the freely available ICU patient data provided in the Medical Information Mart for Intensive Care - III (MIMIC-III) database, compiled between 2001 and 2012 from admissions to the Beth Israel Deaconess Medical Center in Boston, MA [31]. The procedure thus, is to build and test our models using patient data from the MIMIC-III database, then verify our results using patient data collected from hospital participating in our German Smart Medical Information Technology for Healthcare (SMITH) project consortium<sup>29</sup> with real hospitals, and finally roll out the developed model

<sup>29</sup><https://www.smith-center.de/>



# Compute- and Data-driven Use Cases Fact Sheets – Drafts(!)



**WORK  
IN  
PROGRESS**



# Lessons Learned from CoE RAISE Draft Factsheets

## ➤ Fact Sheet process

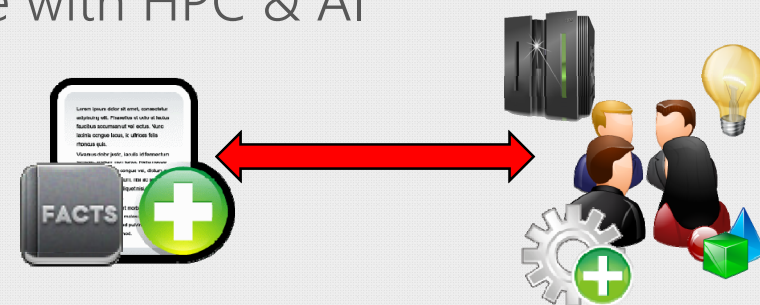
- Participants have been very positive understanding use cases & each other much better
- Enables really to understand where what components of use cases are running & why needed
- Fosters understanding what is confidential, e.g., what could be goals to for industry for a patent

## ➤ Massive complexity observed moving towards Exascale with HPC & AI

- Software engineering expertise required
- AI expertise required
- HPC expertise required
- Application domain-science know-how required

## ➤ Understanding & Communication

- Essential and lots of expertise area-specific terminology and misunderstandings
- Need for a systematic method to succeed in the nine use cases of CoE RAISE & external use cases

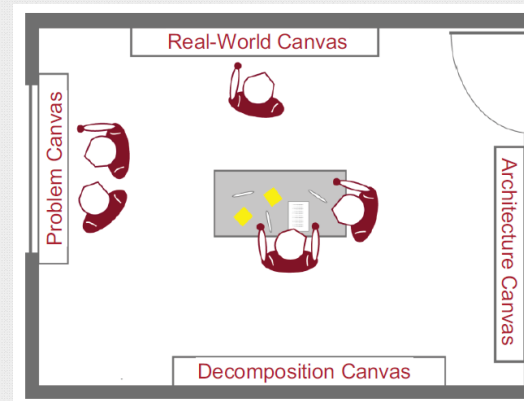


# HPC Systems Engineering in the Interaction Room Seminar

## ➤ 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)

## ➤ Methodology as one CoE RAISE outcome

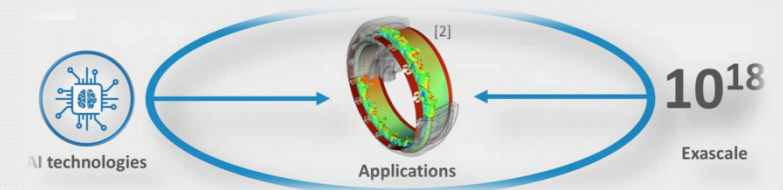


## HPC Systems Engineering in the Interaction Room



Matthias Book

with Morris Riedel, Jülich Supercomputing Centre / UoI and Helmut Neukirchen, University of Iceland



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

drive. enable. innovate.



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