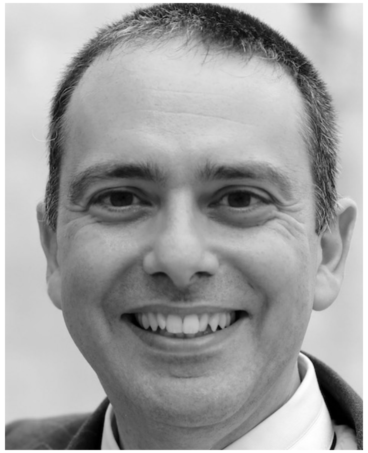
CompBioMed
A Centre of Excellence in Computational Biomedicine

e-Seminar #32
Containers and Cloud-HPC for Virtual Human Twin



Presenter:
Dr Daniele Tartarini
(University of Sheffield)

26 October 2023

The e-Seminar will start
at 2pm CEST / 1pm BST

Moderator:
Tim Weaving
(University College London)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 823712

INSILICO WORLD

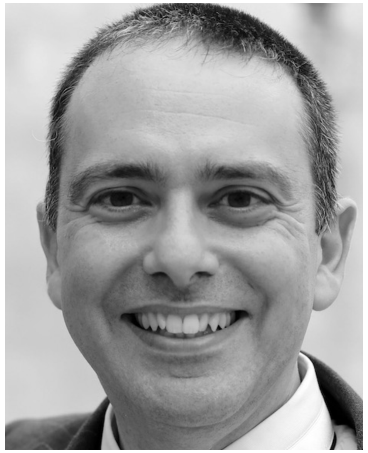
https://insilicoworld.slack.com/archives/C0151M02TA4

The e-Seminar series is run in collaboration with:

VPH Institute
Building the Virtual Physiological Human

e-Seminar #32

Containers and Cloud-HPC for Virtual Human Twin



Presenter:
Dr Daniele Tartarini
(University of Sheffield)

26 October 2023

Welcome!

Moderator:
Tim Weaving
(University College London)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 823712

INSILICO WORLD

https://insilicoworld.slack.com/archives/C0151M02TA4

The e-Seminar series is run in collaboration with:

VPH Institute
Building the Virtual Physiological Human

Outline

- Intro INSIGNEO
- CompBioMed incubation process
- Containers
  - Technology and virtualisation
  - Containers in HPC
  - Parallel Framework for Image Registration
- Virtual Human Twin
  - EDITH-CSA project
  - BoSS: Bank of Sustainable Software
  - Virtual Human Twin platform

Insigneo Institute

Driving innovative research at the interface of healthcare, engineering and science

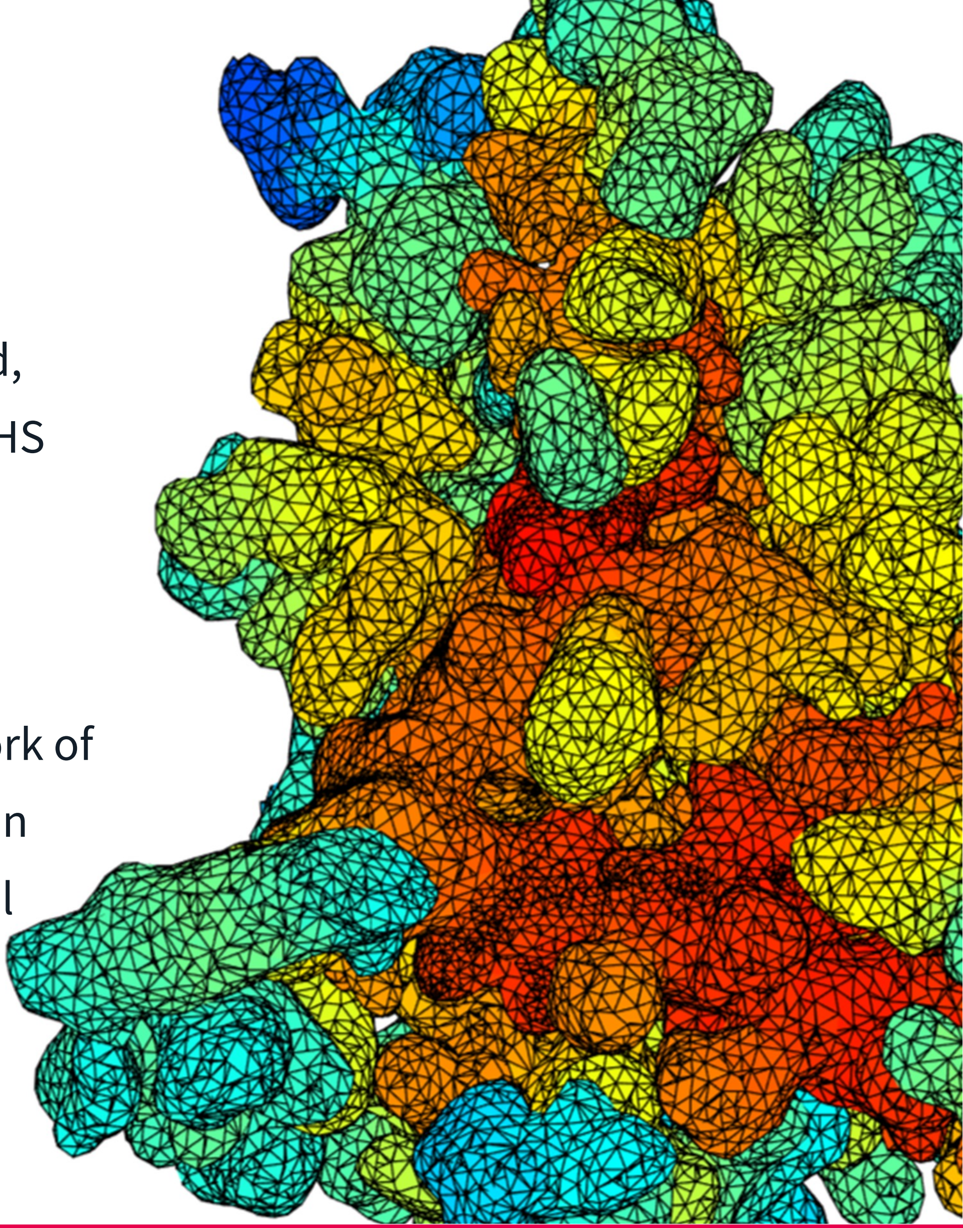
In partnership with:

Doncaster and Bassetlaw Teaching Hospitals NHS Foundation Trust
Sheffield Teaching Hospitals NHS Foundation Trust
Sheffield Children’s NHS Foundation Trust

About us

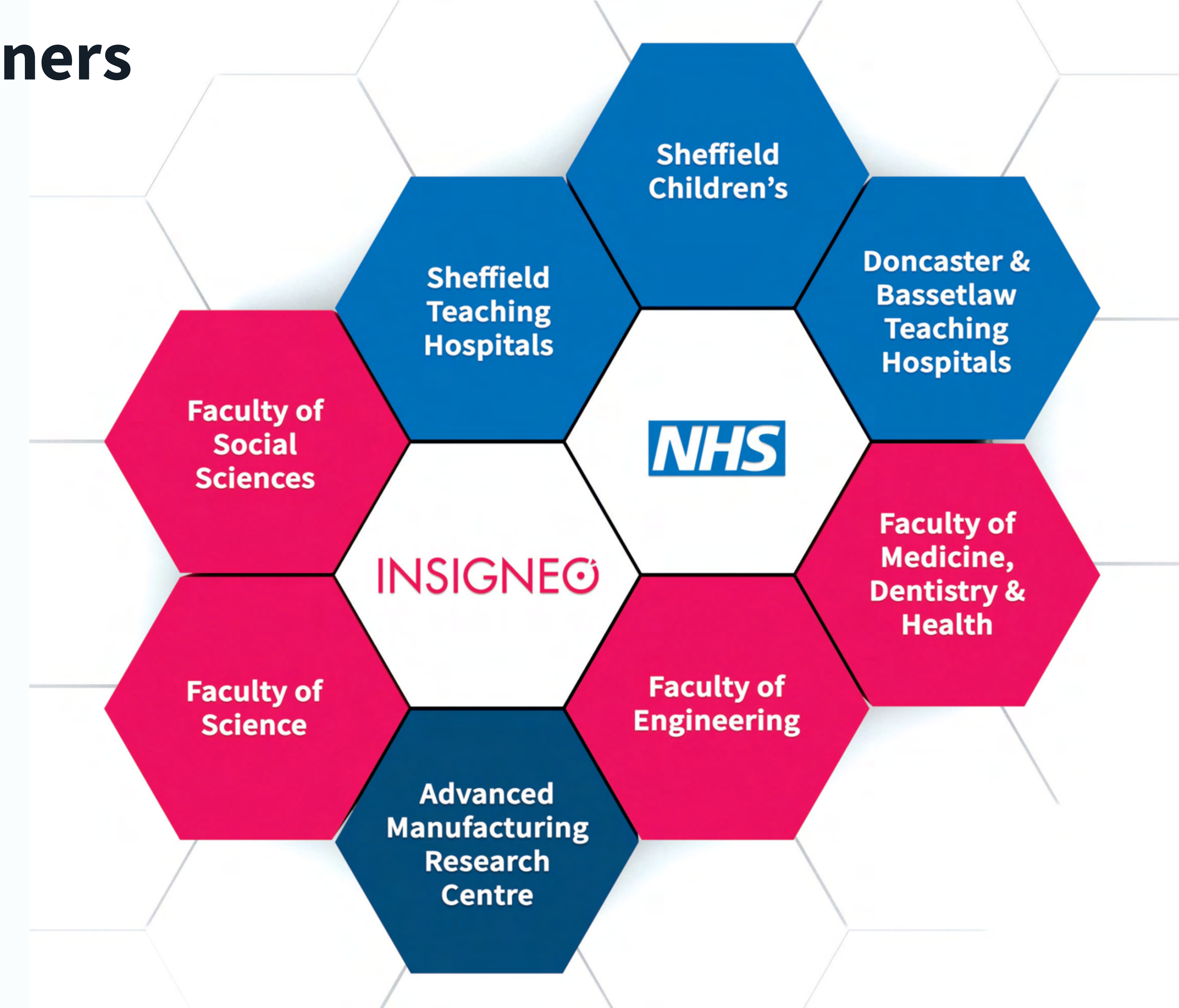
The Insigneo Institute is a collaboration between the University of Sheffield, Sheffield Teaching Hospitals NHS Foundation Trust, Sheffield Children's NHS Foundation Trust and Doncaster and Bassetlaw Teaching Hospitals NHS Foundation Trust.

Established in 2012, the institute has built a strong multidisciplinary network of **290 academics, researchers and clinicians** who bring together expertise in biomedical imaging, healthcare data, computational modelling, and digital healthcare technologies.



https://www.sheffield.ac.uk/insigneo

Insigneo partners


Sheffield Children's
Sheffield Teaching Hospitals
Doncaster & Bassetlaw Teaching Hospitals
NHS
Faculty of Social Sciences
INSIGNEO
Faculty of Medicine, Dentistry & Health
Faculty of Science
Faculty of Engineering
Advanced Manufacturing Research Centre

CompBioMed Incubation Process

Research Development

Optimise algorithm
Validation and feedback from users/stakeholders

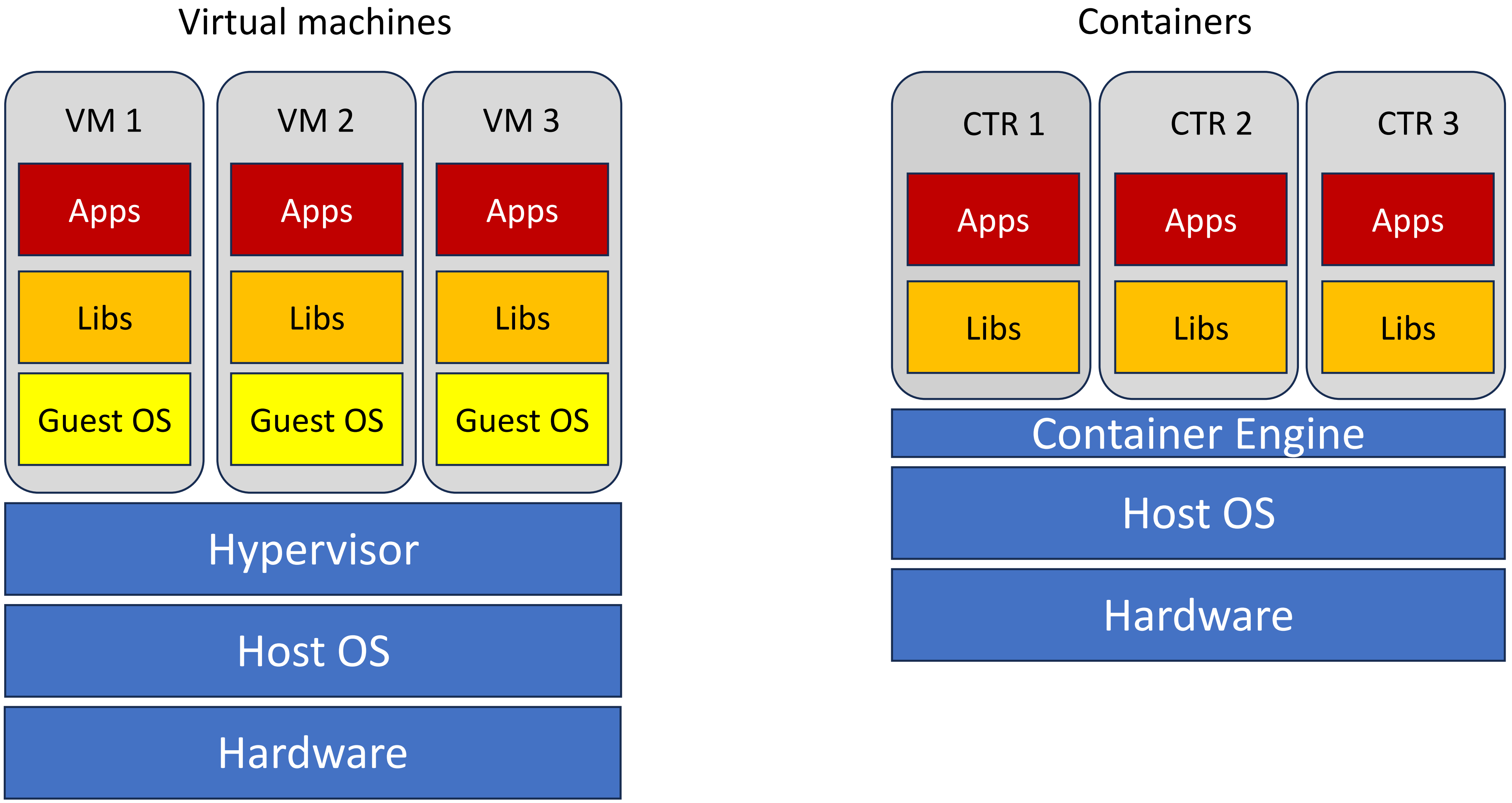
Fast Track

Software Quality Assurance (ISO/IEC 25000, SQuaRE)
FAIR-SW (Findable, Accessible, Interoperable and Reusable)
Reproducibility
Reliability

Deep Track

Optimisation and Parallelisation
Containers, Cloud/HPC Deployment
Preparing for Exascale HPC (10^{18} FLOPS)

Virtualisation Technology



Containerisation Technologies

Docker

- Not designed for HPC/MPI
- Ideal for micro-services
- Initially run in root mode
- Engine/demon model
- Image repository for each instance
- Deployable in Cloud

Podman (RedHat)

- recently support HPC
- Runs Docker containers

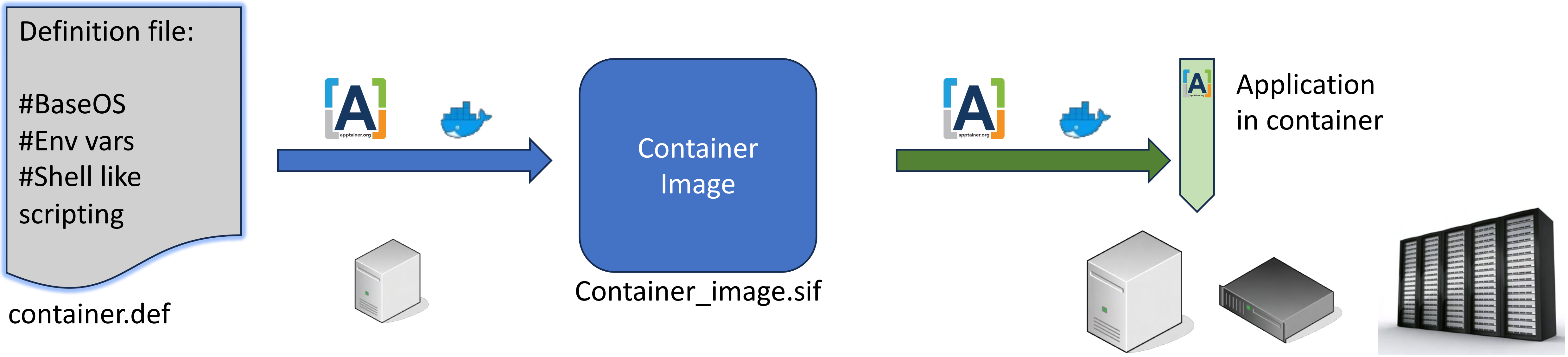
Shifter, Chaliecloud

Singularity/Apptainer (our choice)

- Native HPC/MPI support
- Support for Accelerated hardware (GPUs)
- Support for Docker images and OCI
- Large adoption in HPC centres
- Support for Slurm
- Deployable on cloud systems
- Checkpointing

(Veiga et al 2019 IEEE/ACM)

Containerisation Process



Definition file:

#BaseOS
#Env vars
#Shell like
scripting

container.def

Container
Image

Container_image.sif

Application
in container

Building machine

```
$ apptainer build container_image.sif. container.def
```

Execution machines (also VM, Cloud)

```
$ apptainer exec container_image.sif. App_in_container
```

Compatibility issues between building machine and execution machine

- Container OS needs to be of the same architecture of the execution machine
- Application in container need to be built for the correct target architecture (cross compilation)

FAIR Containers

Findable: DockerHub, Sylab Cloud, Github

Accessibile: CLI interface, Web

Interoperable: Open Container Initiative format

Reusable: inclusion in other containers and run on compatible systems

Compatible linux kernels only

Relies on supported VM engine (Intel CPUs vs Apple)

Relies on Windows Subsystem for Linux (WSL2) VM

Limited portability due to MPI/network

Containers Promise: Build-Once run Everywhere (?)

PRACE-ICEI (Partnership for Advanced Computing in Europe)

PRACE-ICEI for 1 Milion core hours

- Cineca Galalileo100 Scalable computing and Cloud
- Julich Jureca Scalable Computing

JURECA-DC Scalable Computing:

- 768 worker nodes
- Two **AMD** EPYC Rome 7742 64-core CPUs, 512 GB DDR4 memory per node
- 198 of the worker nodes provide 4 NVIDIA A100 GPUs each, 96 worker nodes feature 1TB of memory.
- 100 Gb/**s NVIDIA Mellanox** HDR100 high-speed
- Centos 8.8
- Apptainer version 1.2.4-1.el8

JURECA

(c) Forschungszentrum Jülich

“We acknowledge PRACE for awarding access to the Fenix Infrastructure resources at Cineca and Julich, which are partially funded from the European Union’s Horizon 2020 research and innovation programme through the ICEI project under the grant agreement No. 800858.”

PRACE: Jureca-DC

- Apptainer available but Containers not actively supported
- Build of Apptainer images not allowed on HPC nodes
- Building of images with Julich Build System takes place on a dedicated system that is external to the clusters, now **DEPRECATED**.
- The dedicated building system has different characteristics compared to the HPC machines (different CPU type, no GPUs); images might not be optimized to the fullest extent to the targeted system.
- No container template available

Cineca HPC Galileo100

Scalable Computing:

636 computing nodes

2 x CPU **Intel** CascadeLake 8260, with 24 cores each, 2.4 GHz, 384GB RAM, subdivided in:

- 422 standard nodes ("thin nodes") 480 GB SSD
- 180 data processing nodes ("fat nodes") 2TB SSD, 3TB Intel Optane
- 100Gbs **Infiniband** interconnection.

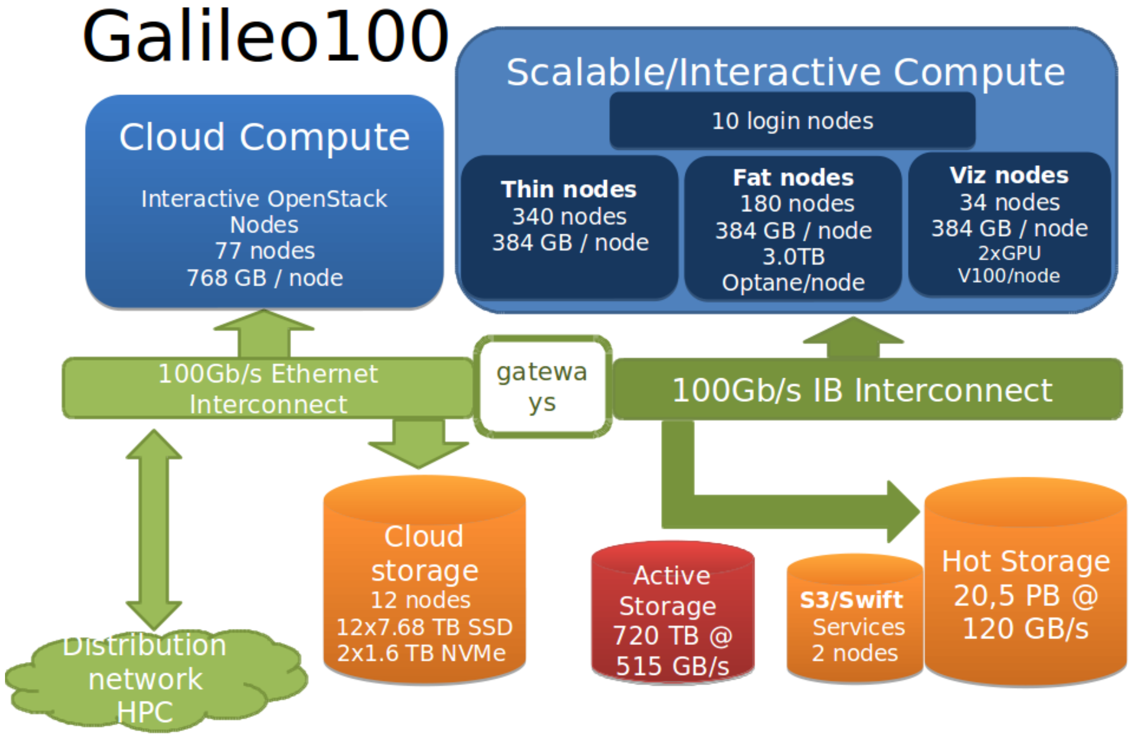
Cloud Computing:

77 computing nodes

- OpenStack for cloud computing
- 100Gbs **Ethernet** interconnection.

- Operating system Centos 8.3
- **Apptainer 1.1.6-1.el8**

Same computing Hardware, different network interconnection



Galileo100

Cloud Compute

Interactive OpenStack Nodes
77 nodes
768 GB / node

Scalable/Interactive Compute

10 login nodes

Thin nodes
340 nodes
384 GB / node

Fat nodes
180 nodes
384 GB / node
3.0TB Optane/node

Viz nodes
34 nodes
384 GB / node
2xGPU V100/node

100Gb/s Ethernet Interconnect

gateways

100Gb/s IB Interconnect

Distribution network HPC

Cloud storage
12 nodes
12x7.68 TB SSD
2x1.6 TB NVMe

Active Storage
720 TB @ 515 GB/s

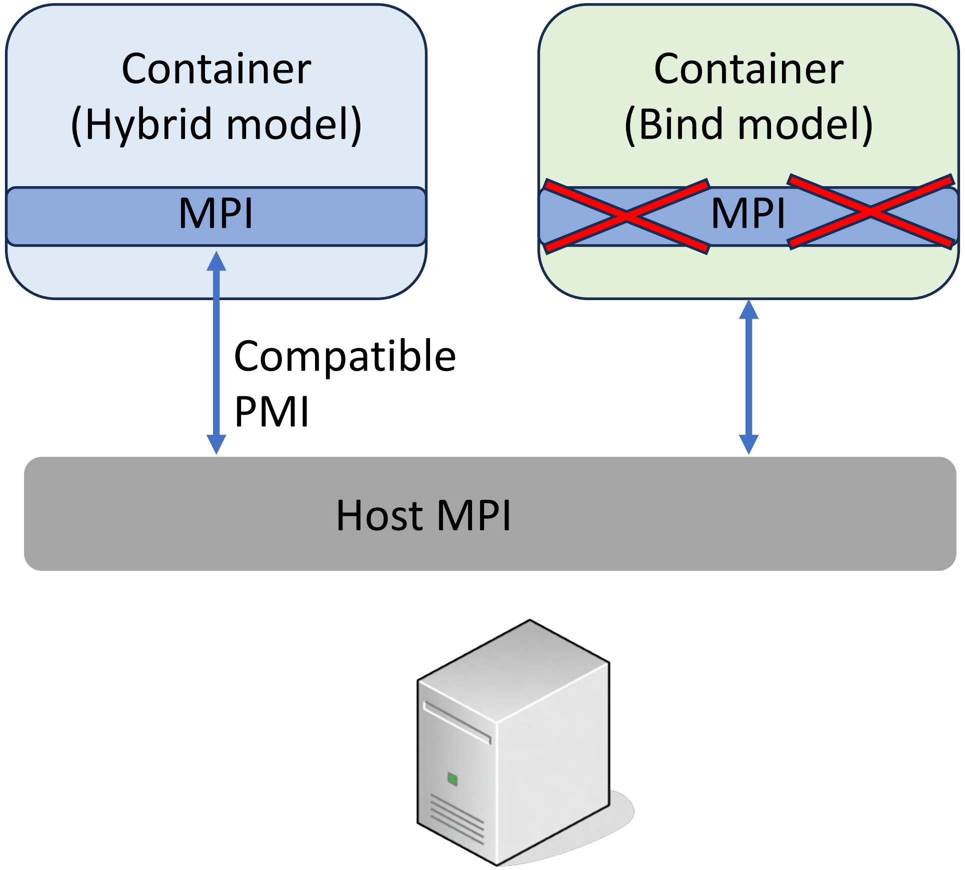
S3/Swift
Services
2 nodes

Hot Storage
20,5 PB @ 120 GB/s

Apptainer and MPI

Supports MPICH and OpenMPI both with GPUs

- **Bind Model**:
    - MPI implementation available on the host and not include any MPI in the container.
- **Hybrid Model (Host MPI):**
    - MPI libraries needs to be installed and configured in container
    - Compatibile Process Management Interface



Apptainer MPI: Bind Model

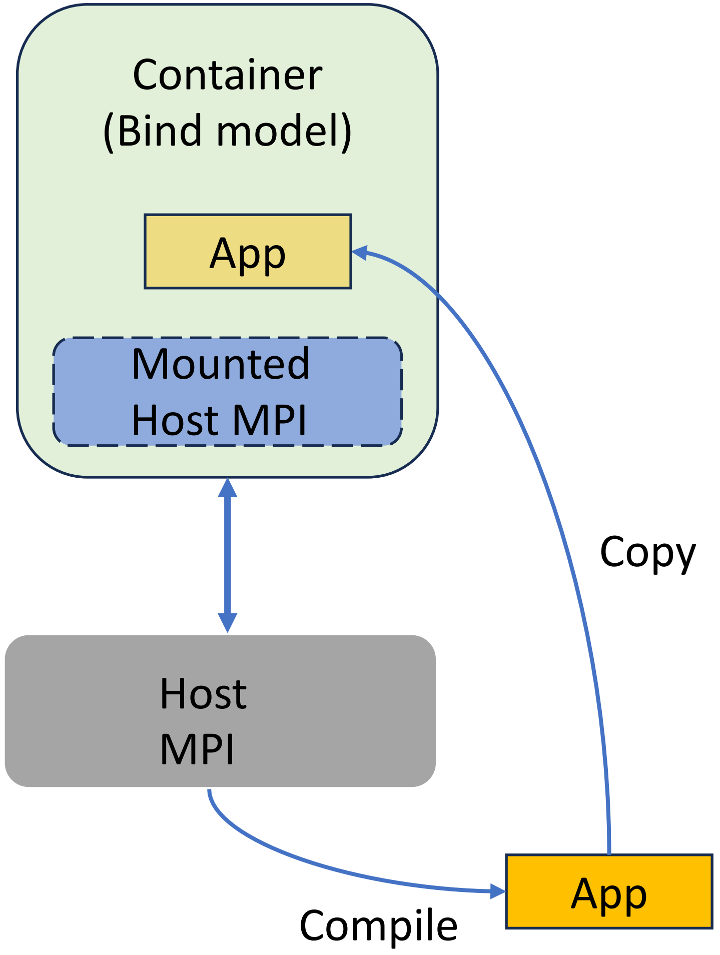
The *bind* model requires binding MPI from host into container

The drawbacks are:

- Knowing MPI location in host
- Binding must be allowed by administrators
- Ensure that host MPI is compatible with the MPI used compile
- Mounting Open MPI and networking libraries into the container raises **glib compatibility** issues (container glib needs to be more recent than host)
- Portability compromised

Pros:

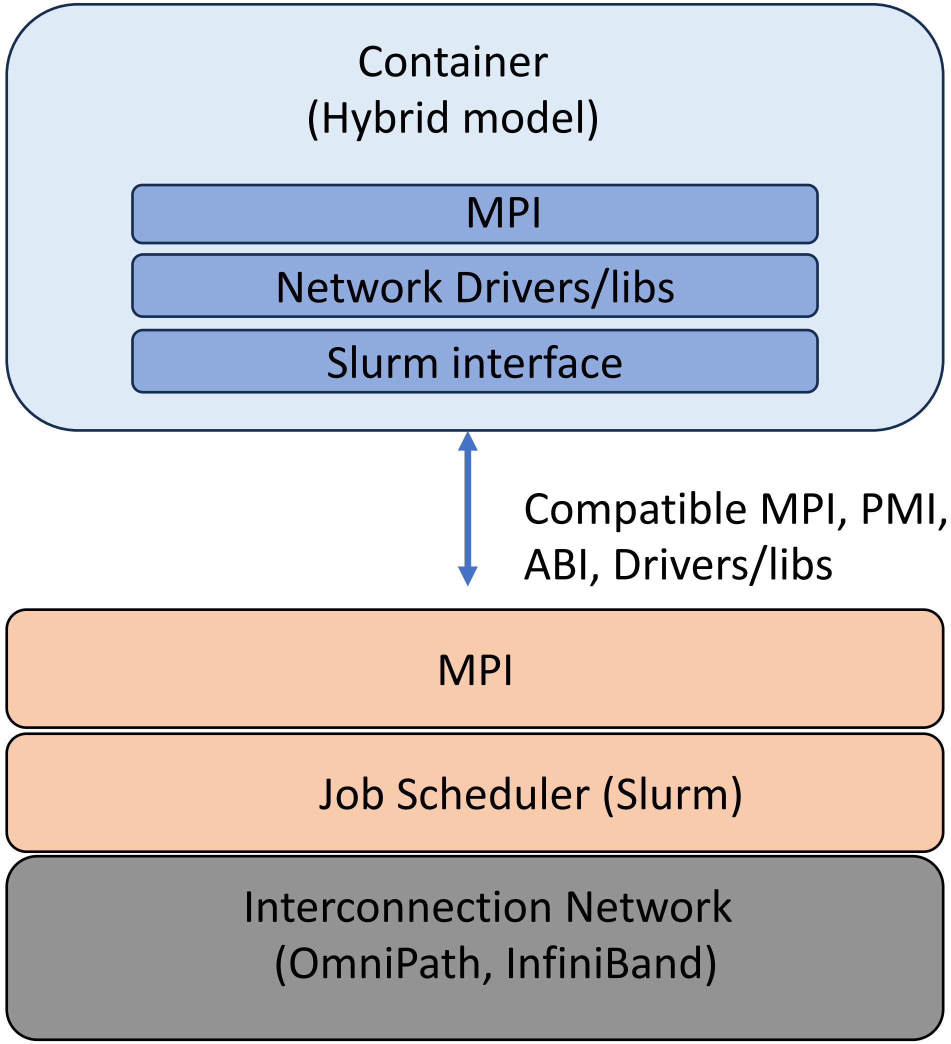
- Integration with resource managers such as Slurm.
- Container images are smaller without MPI libs



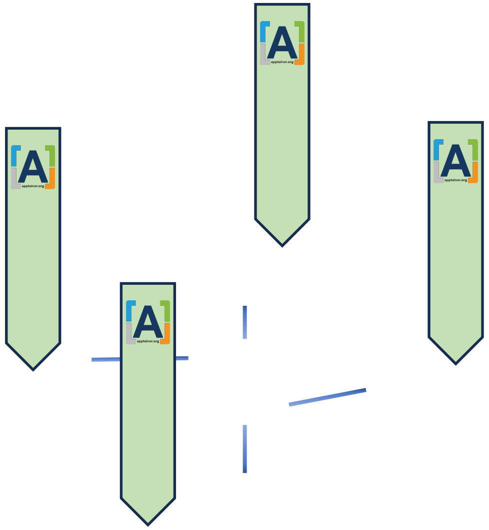
```
$ export MPI_DIR="< … >"
$ mpirun -n <NUMBER_OF_RANKS> singularity exec --bind "$MPI_DIR"   container.sif    App_in_container
```

Apptainer Container MPI Hybrid Mode

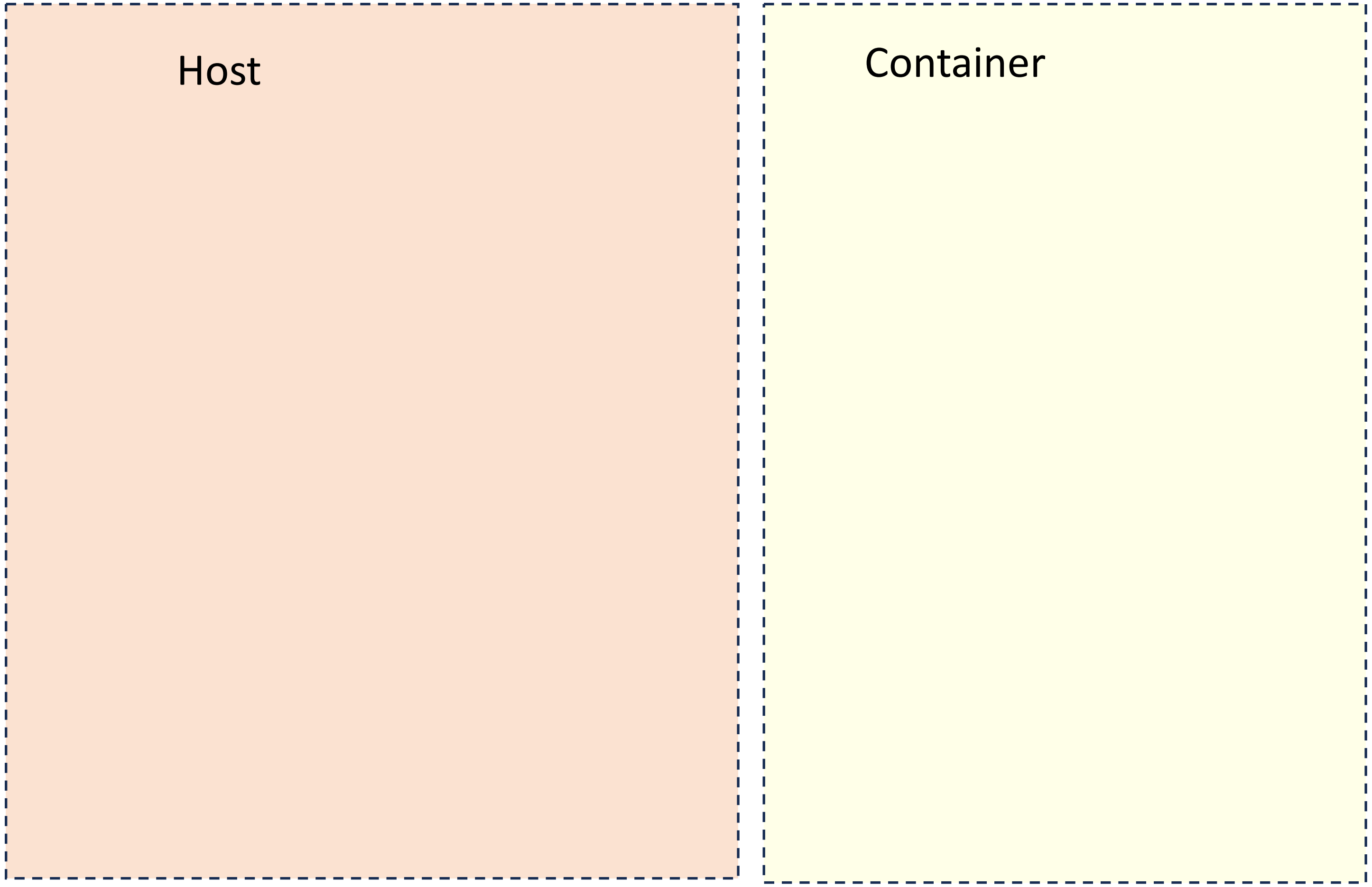
- MPI libraries installed in the container must be compatible with the version on the host
- Network Drivers for High bandwidth/low latency network with Remote Direct Memory Access (RDMA): Intel Omni-Path, Infiniband (OFED, OpenFabrics Enterprise Distribution)
- Process management mechanism and version (e.g. PMI2 / PMIx) should match the one used on the host
- Job Scheduler support configures (Slurm, SGI, PBS, etc)



Apptainer and Hybrid MPI



All non-shared memory communication occurs through the PMI and then to local network interfaces



```
$ mpirun -n #ranks   apptainer  exec   my_image.sif   mpi_application_in_container
```

Lesson Learned: MPI Compatibility/ Portability

ABI compatibility: MPI “is” backward compatible allowing old programs compiled with old versions of a library to run with newer versions without the need to recompile.

Container MPI need to be older than host MPI

		Host OpenMPI					
		2.0.0	2.0.1	2.0.3	2.1.1	3.0.0	4.1.1
Container OpenMPI	2.0.0	🟩	🟥	🟥	🟥	🟩	🟥
	2.0.1	🟥	🟩	🟥	🟥	🟩	
	2.0.3	🟥	🟥	🟩	🟥	🟩	
	2.1.1	🟥	🟥	🟥	🟩	🟩	
	3.0.0	🟩	🟩	🟩	🟩	🟩	🟥
	4.1.1	🟥				🟥	🟩

- Required Container aware of HW/interconnections (e.g. Infiniband, Omni-Path)
  - Drivers
  - Libraries: OFED

Process Management Interface Exascale (PMIx) standard:
Allows different process managers to interact with the MPI library in a standardized way.

(Veiga, et al 2019 IEEE/ACM) (PMI: A Scalable Parallel Process-Management Interface for Extreme-Scale Systems. Balaji, et al)

Apptainer and Hybrid MPI

The advantages of this approach are:

- Possible integration with resource managers such as Slurm
- Simplicity since similar to natively running MPI applications.

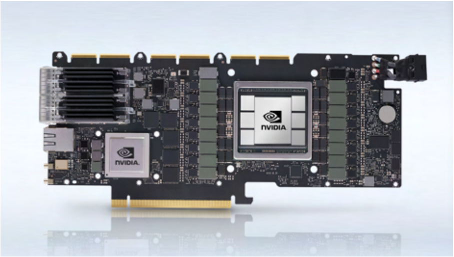
The drawbacks are:

- The MPI in the container must be **compatible** with the version of MPI available on the host.
- The MPI implementation in the container must be **carefully configured** for optimal use of the hardware if performance is critical.

Standard approach is to build the MPI container with the same MPI framework installed on the host from source.

(Apptainer.org)

Containers for GPU Systems



Apptainer natively supports NVIDIA CUDA and ROCm

Host machine needs GPU driver and library installation.

Compatible libc: Running a container with an older libc than the host will work, no guarantee the way round.

Recommended to build on Containers form Manufacturers Hub (Docker):

NVIDIA Container Toolkit
https://docs.nvidia.com/datacenter/cloud-native/container-toolkit/latest/install-guide.html

ROCm Infinity Hub
https://www.amd.com/en/technologies/infinity-hub

Lesson Learned: PRACE

Galileo100@Cineca:

- Singularity build not allowed on HPC.
- Build possible on VM of same hardware but network interconnect/GPUs.
- Challenging to set up correct MPI configuration.
- **Proactive IT Support makes the difference***

Jureca-DC@ Julich:

- Building images requires administrator privileges which regular users do not have on JSC’s clusters
- The Container Build System has different characteristics compared to the HPC machines (DEPRECATED).
- Created images might not be optimized to the fullest extent to the targeted system.

* Thanks to Dr Memmolo, Dr Caravita

Containers in the Cloud experience: SURF

Catalog of pre-configured VMs with single application

You can add your application to the catalog!

SURF Research Cloud is Infrastructure As A Service

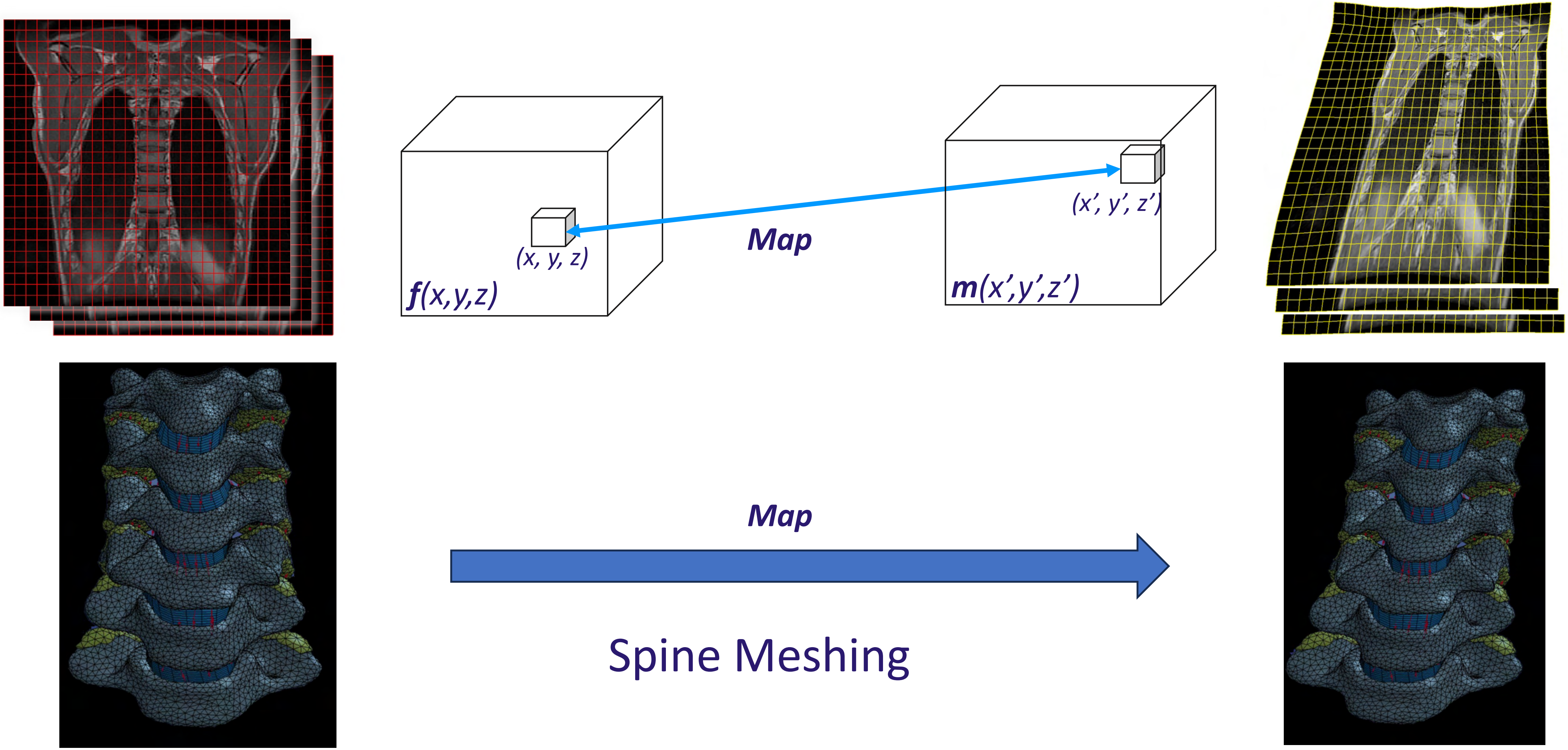
- Customisable VM via user friendly portal: CPUs, RAM, disks, OS, apps
- Available Docker/Singularity
- Root access to VM
- Fast private network for all VMs in project/shared env
- No hardware ownership
- Start/stop of VM generally under 1 minute

Disadvantages:

- You maintain everything in your VM
- Pay for VM uptime, not just compute time
- No automatic backups
- Slow disk I/O

"This work used the Dutch national e-infrastructure with the support of the SURF Cooperative using grant no. EINF-2979."

pFIRE: parallel Framework for Image Registration



f(x,y,z)

(x, y, z)

Map

m(x',y',z')

(x', y', z')

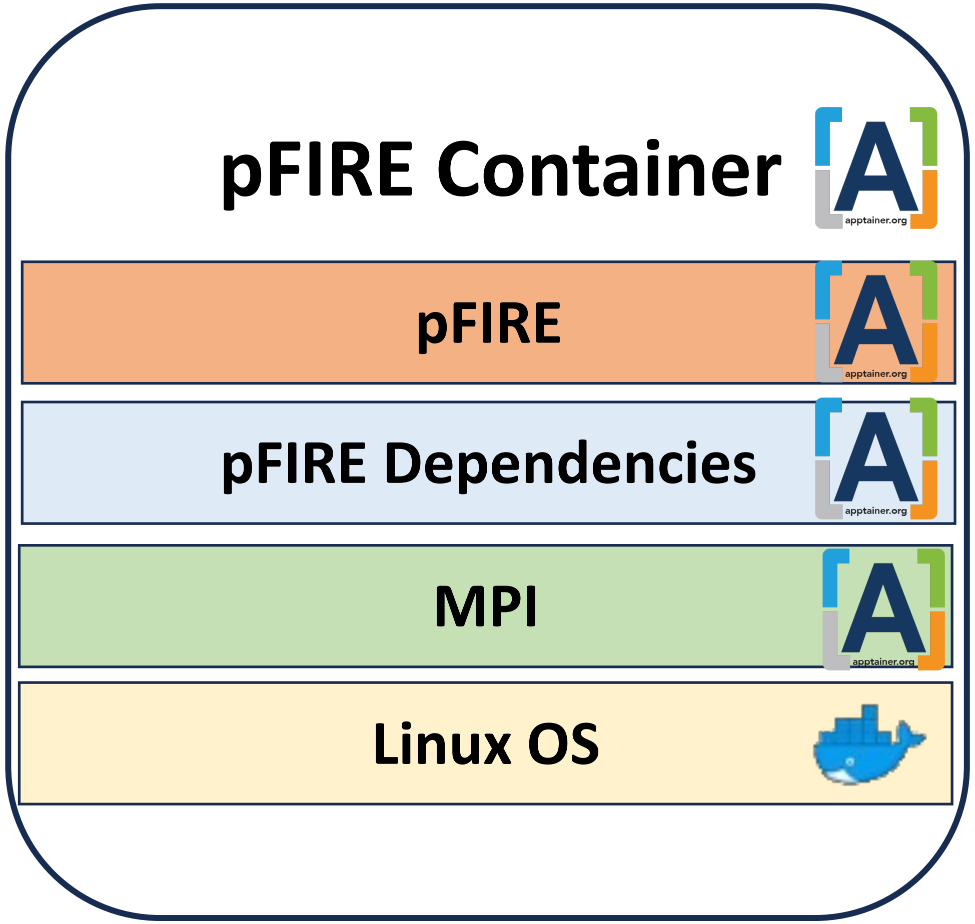
Map

Spine Meshing

https://github.com/INSIGNEO/pFIRE

Image Creative License https://doi.org/10.3389/fbioe.2014.00058

pFIRE Containers Modularisation



pFIRE Container

pFIRE

pFIRE Dependencies

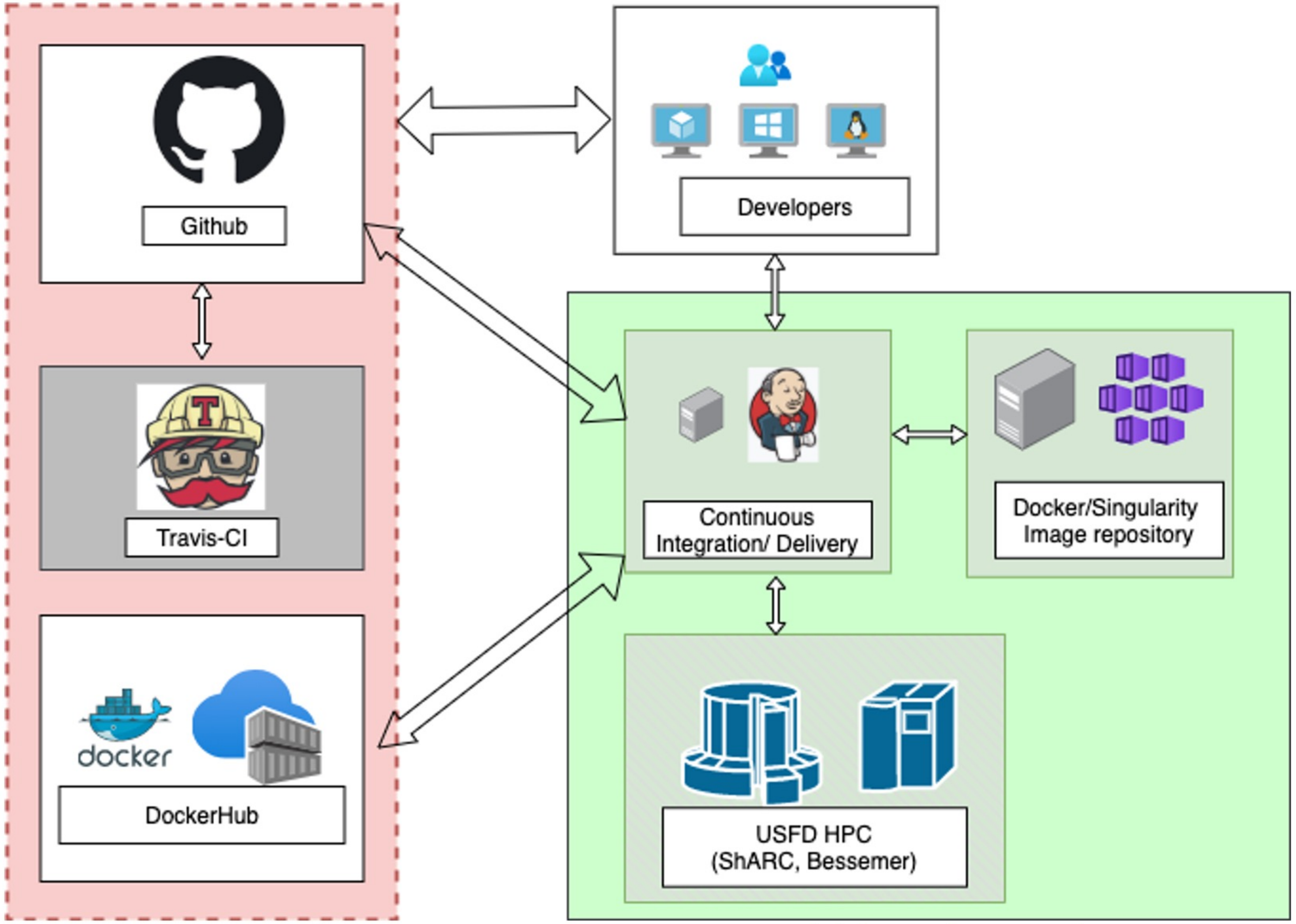
MPI

Linux OS

Matrix of flavours

RC.x	Dev	Maint		
PETSc, HDF5, Boost, Image formats multiple version combinations				
OpenMPI, MPI-CH, Hybrid or Bind mode				
GPU specific	Gravitons	ARM-64	X86-64	Linux distros

Software Engineering: CI/CD

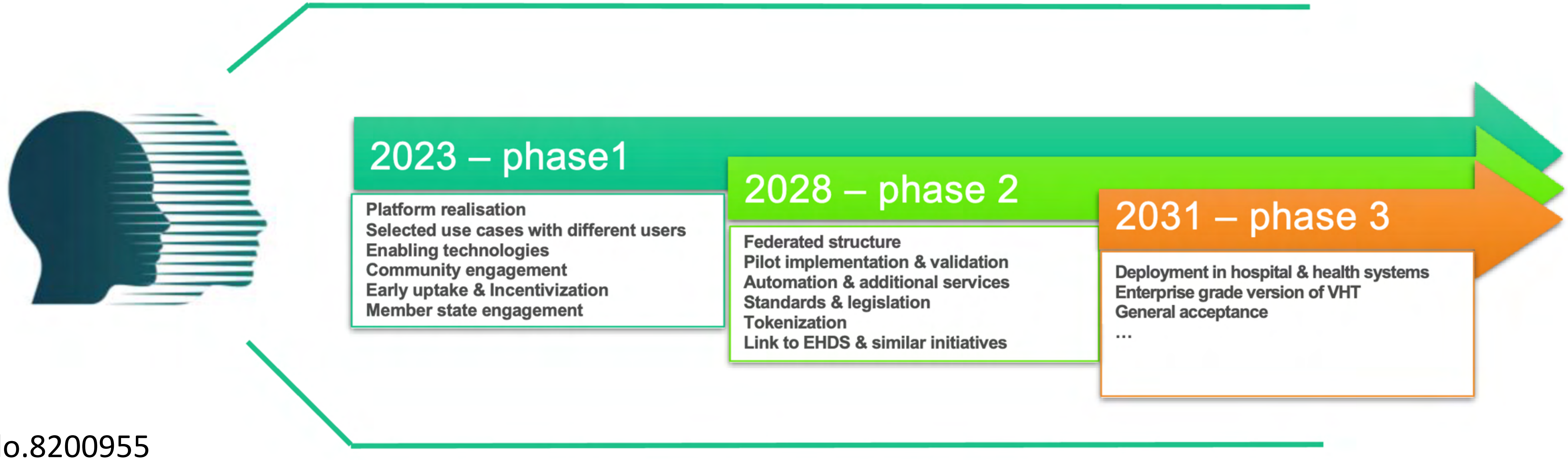

Github
Developers
Travis-CI
Continuous Integration/ Delivery
Docker/Singularity Image repository
docker
DockerHub
USFD HPC (ShARC, Bessemer)

Building the European Virtual Human Twin

EDITH CSA: Virtual Human Twin roadmap (first draft)

Objectives:

- Mapping of the current ecosystem for digital twins in healthcare
- Developing a roadmap toward Human Digital Twin (HDT):
- Implementing a federated cloud-based repository
- Designing a simulation platform



2023 – phase1

Platform realisation
Selected use cases with different users
Enabling technologies
Community engagement
Early uptake & Incentivization
Member state engagement

2028 – phase 2

Federated structure
Pilot implementation & validation
Automation & additional services
Standards & legislation
Tokenization
Link to EHDS & similar initiatives

2031 – phase 3

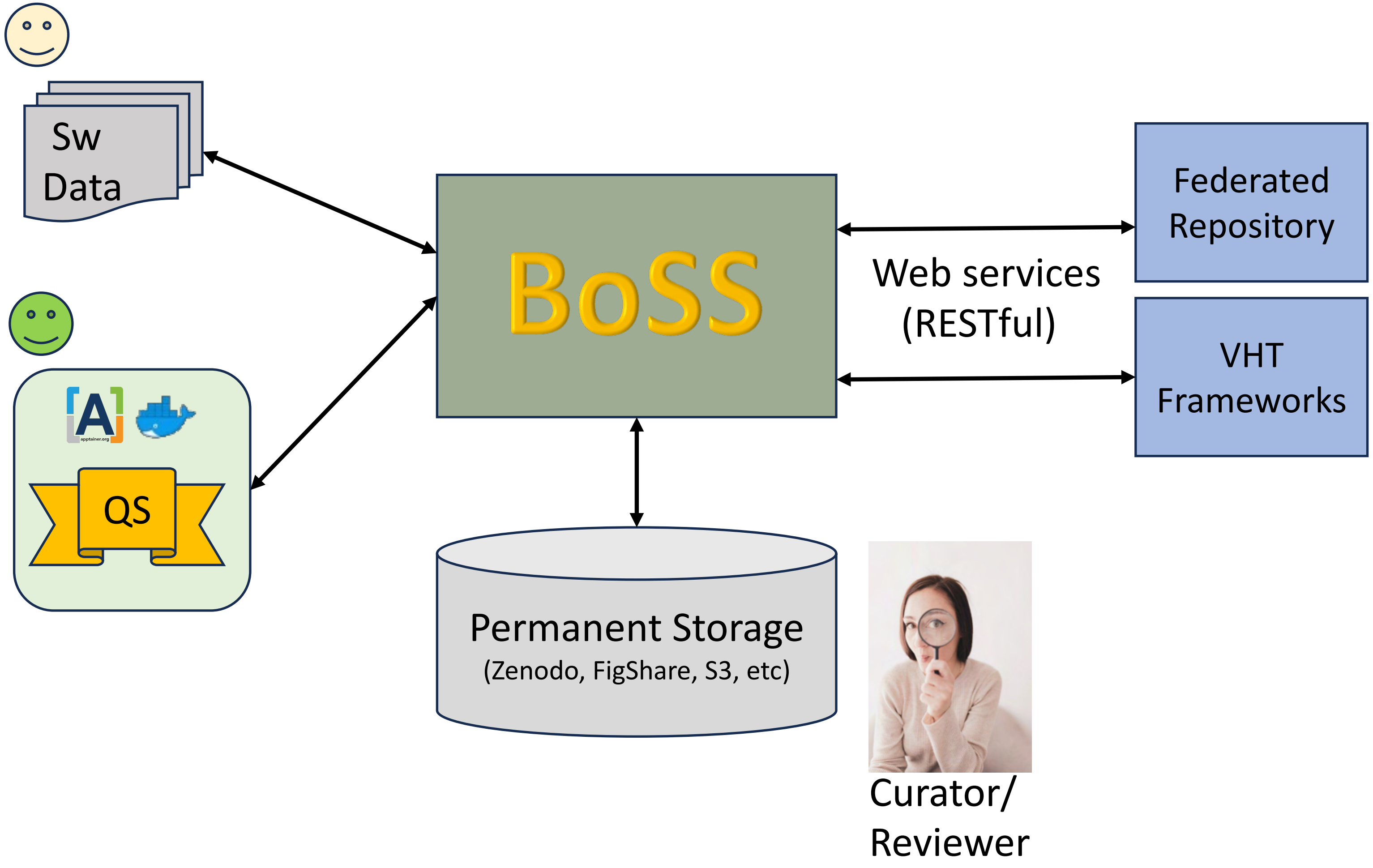
Deployment in hospital & health systems
Enterprise grade version of VHT
General acceptance
...

https://doi.org/10.5281/zenodo.8200955

BoSS: Bank of Sustainable Software

Research England

- FAIR-SW
- Traceability
- Peer Reviewed/Curated
- Sustainable
- Quality Score based on:
    - Testing
    - Verification
    - Validation



Sw Data

QS

BoSS

Web services (RESTful)

Federated Repository

VHT Frameworks

Permanent Storage (Zenodo, FigShare, S3, etc)

Curator/ Reviewer

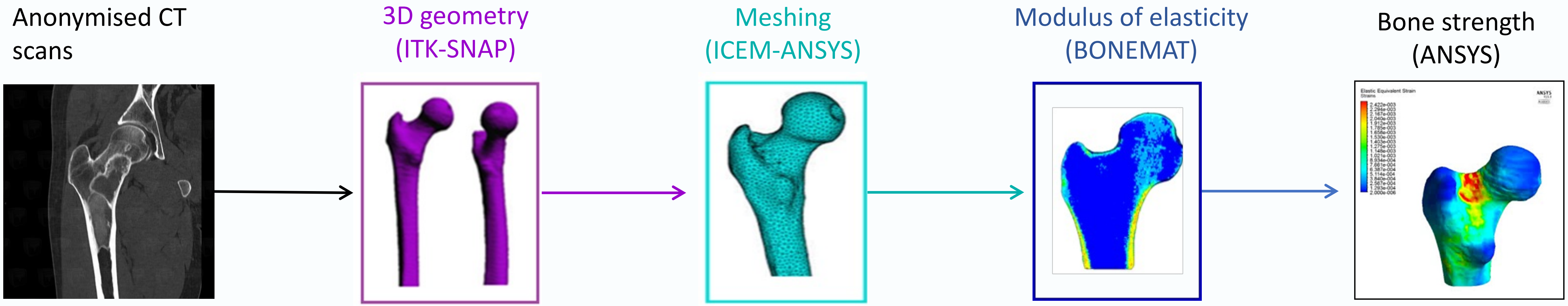
Long Term Sustainability Plans Available

Contact: d.tartarini@sheffield.ac.uk, daniele.tartarini@gmail.com

Research England Open Research Culture Fund

CT2S service: Computed Tomography to Strength

- To created personalized digital twin model of proximal femur to predict the **risk of fracture** using bone strength

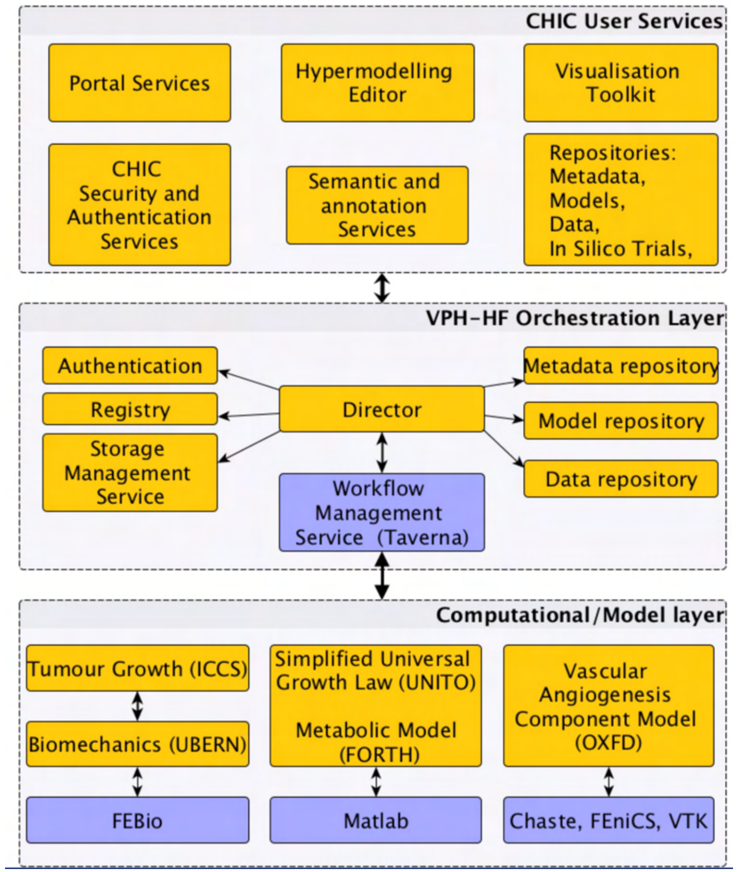


- Currently run as an Insigneo online service: https://ct2s.insigneo.org/ct2s/
- Validated and verified pipeline based on cadaveric experiments
- Requires Ansys license on HPC. ITK-Snap and Bonemat (windows only) are open source software.
- Pipeline previously ported to SHARC (Sheffield HPC) and ARCHER
- **Ansys moving from CentOS to Ubuntu**

Li et al. (2015) J Biomech, 48; Schileo et al. (2008) J Biomech, 41

Virtual Human/Digital Twin platforms

CHIC
Computational Horizons in Cancer



CHIC User Services

Portal Services

Hypermodelling Editor

Visualisation Toolkit

CHIC Security and Authentication Services

Semantic and annotation Services

Repositories:
Metadata,
Models,
Data,
In Silico Trials,

VPH-HF Orchestration Layer

Authentication

Registry

Storage Management Service

Director

Workflow Management Service (Taverna)

Metadata repository

Model repository

Data repository

Computational/Model layer

Tumour Growth (ICCS)

Biomechanics (UBERN)

FEBio

Simplified Universal Growth Law (UNITO)

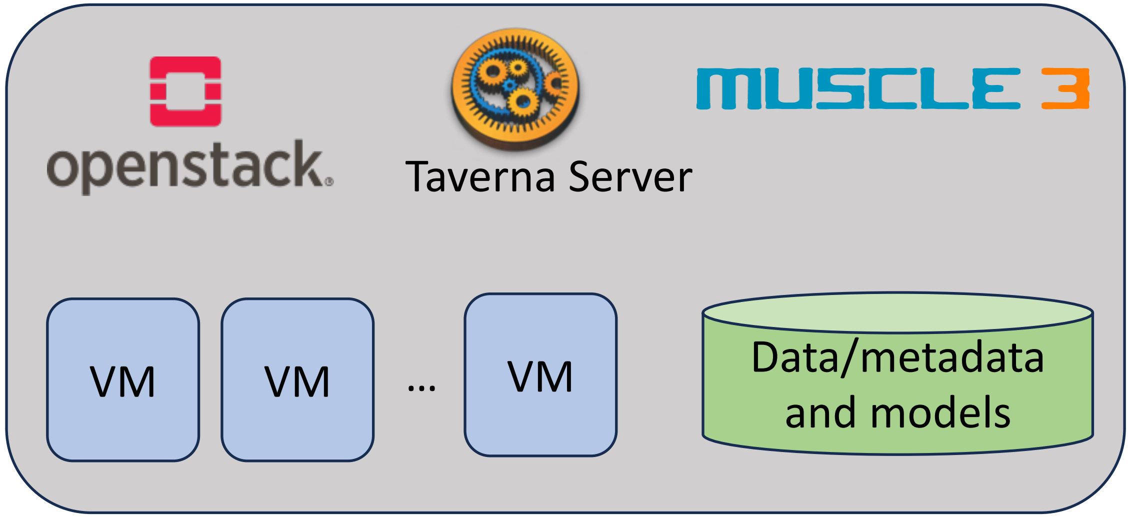
Metabolic Model (FORTH)

Matlab

Vascular Angiogenesis Component Model (OXFD)

Chaste, FEniCS, VTK

Virtual Physiological Human- Hypermodelling Framework: A software framework for the execution of complex subject-specific physiology modelling workflows



openstack.

Taverna Server

MUSCLE 3

VM

VM

…

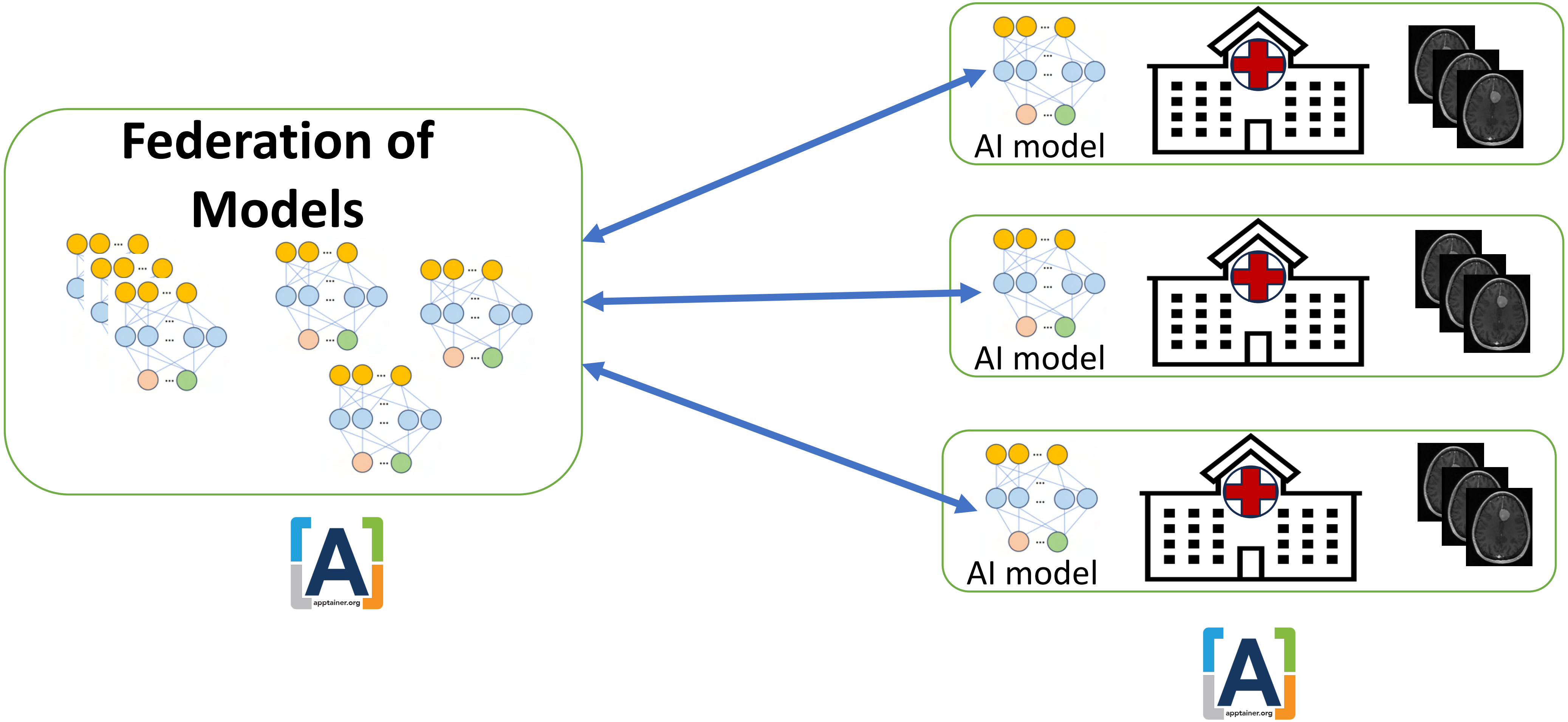
VM

Data/metadata and models

Lightweight workflow components can use Kubernetes

(Viceconti, Sbna', Tartarini, et al. J. Comp Sci 2018)
https://doi.org/10.1016/j.jocs.2018.02.009

Federated Learning



Federation of Models

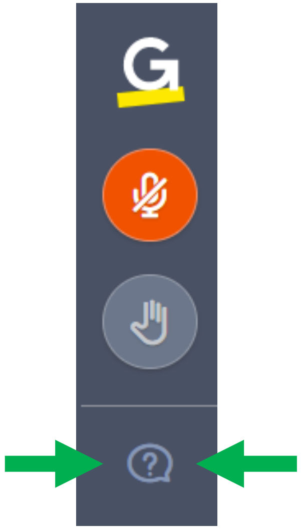
AI model

AI model

AI model

Q&A

To pose a question, please click on the ? symbol and send your question via the 'Ask the staff a question' panel



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 823712

Thank you for participating!

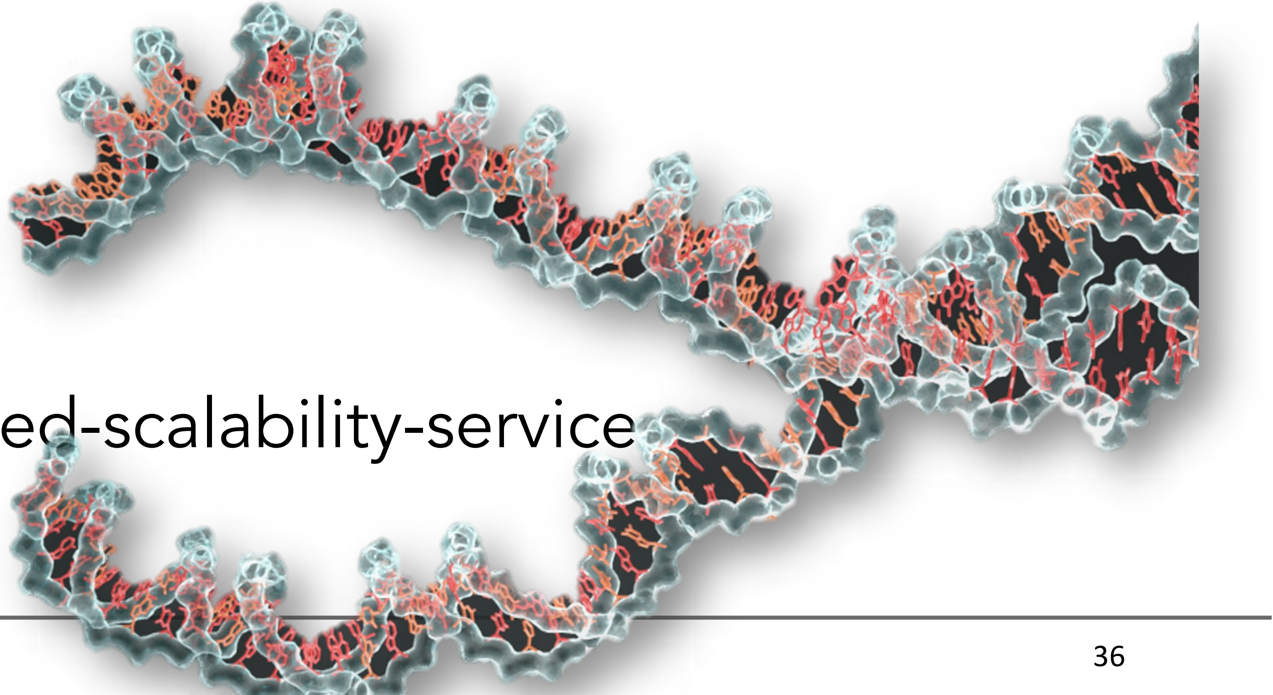
…don't forget to fill in our feedback questionnaire…

Visit the CompBioMed website (www.compbiomed.eu/training)
for a full recording of this and other e-Seminars,
to download the slides
and to keep updated on our upcoming trainings

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 823712

CompBioMed's *Free* Scalability Service

- Improves performance of your biomedicine applications on high performance computers
  - Experts in both biomedical applications and high performance computers
  - Make your biomedicine applications run in parallel
  - Improving the scalability of those already parallelised



- www.compbiomed.eu/compbiomed-scalability-service

www.compbiomed.eu/compbiomed-scalability-service

- Contact for *Free* Service
  - General technical questions
    - Slack: #scalability channel of ***the InSilicoWorld Community of Practice***
    - Email: compbiomed-support@ucl.ac.uk
  - Full service
    - Application Form or light-weight web form
      - Formal collaborative relationship with CompBioMed Centre of Excellence

- Application and Data Security
  - Great care when adapting your applications and managing your data
    - Our Data Policies cover Data Privacy, Data Security and Research Data Management

InSilicoWorld Community of Practice

The first community entirely on *in silico* medicine on Slack

www.insilico.world/community

Expertise

- The community is invitation only: in this way we ensure only interested experts have access

Collaboratio n

- Join teams and collaboratively work on shared goals, projects, concerns, problems or topics

Safe space

- A pre-competitive space where experts from academia, industry, and regulatory agencies can ask for and exchange advices

More than 500 experts have already joined the community and its channels

InSilicoWorld Members

- **Large Biomedical Companies**

  Medtronic, Smith & Nephew, Pfizer, Johnson and Johnson, Innovative Medicine Initiative, CSL Behring, Ambu, RS-Scan, Corwave EN, Zimmer Biomet, Novartis, Bayer, ATOS, Biogen, Agfa, Icon PLC, Amgen, ERT, Exponent, etc.

- **Biomedical SMEs**

  Nova Discovery, Lynkeus, Obsidian Biomedical, Quibim, Mediolanum Cardio Research, Voisin Consulting, CRM-Microport, Mimesis srl, H. M. Pharmacon, MCHCE, etc.

- **Independent Software Vendors**

  Ansys, In Silico Trials Technologies, 3DS, KIT, ASD Advanced Simulation & Design GmbH, Kuano-AI, Aparito, Chemotargets, Digital Orthopaedics, ExactCure, Materialise, Bio-CFD, Matical, FEOPS, 4RealSim, Exploristics, Synopsis, Virtonomy, Cad-Fem Medical, etc.

- **Regulators and Standardisation Bodies**

  FDA, DIN, BSCI China, NICE, Critical Path Institute, ACQUAS, etc.

- **Clinical Research Institutions**

  Istituto Ortopedico Rizzoli, Sloan Kettering Cancer Center, Royal College of Surgeons Ireland, Gratz University Hospital, Charite Berlin, Centre Nacional Invesigaciones Oncologicas, Aspirus Health, Universitätsklinikum des Saarlandes, European Society for Paediatric Oncology, etc.

Questions

- What are the main limitation to adopt Cloud-HPC in the research institutions?
- Limitations in accessing research/commercial clouds
- What type of sustainability plans do you have for BoSS
- What would be your suggested approach to someone willing to containerize their application?

References

- Containers in HPC: a scalability and portability study in production, IEEE 2019
- Balaji, P. *et al.* (2010). PMI: A Scalable Parallel Process-Management Interface for Extreme-Scale Systems. In: Keller, R., Gabriel, E., Resch, M., Dongarra, J. (eds) Recent Advances in the Message Passing Interface. EuroMPI 2010. Lecture Notes in Computer Science, vol 6305. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-15646-5_4. https://www.mcs.anl.gov/papers/P1760.pdf
- A Qualitative and Quantitative Analysis of Container Engines. Baresi et al https://arxiv.org/pdf/2303.04080
- ABI backwards compatibility tracker https://abi-laboratory.pro/index.php?view=timeline&l=openmpi

Virtualisation Technology: Docker

Docker container technology was launched in 2013 as an open source Docker Engine.

It leveraged existing computing concepts around containers and specifically in the Linux world, primitives known as **cgroups and namespaces**. Docker's technology is unique because it focuses on the requirements of developers and systems operators to separate application dependencies from infrastructure.

Success in the Linux world drove a partnership with Microsoft that brought Docker containers and its functionality to Windows Server.

Technology available from Docker and its open source project, Moby has been leveraged by all major data center vendors and cloud providers. Many of these providers are leveraging Docker for their container-native IaaS offerings. Additionally, the leading open source serverless frameworks utilize Docker container technology.

Docker open sourced libcontainer and partnered with a worldwide community of contributors to further its development.

In 2015 Docker donated the container image specification and runtime code now known as runc, to the Open **Container Initiative (OCI)** to help establish standardization as the container ecosystem grows and matures.