Processing Complex Medical Workflows in the EurValve Environment

Marian Bubak[1,2], Tomasz Gubala[1], Rod Hose[3], Marek Kasztelnik[1], Maciej Malawski[1,2], Jan Meizner[1], **Piotr Nowakowski[1]**, Steven Wood[4]

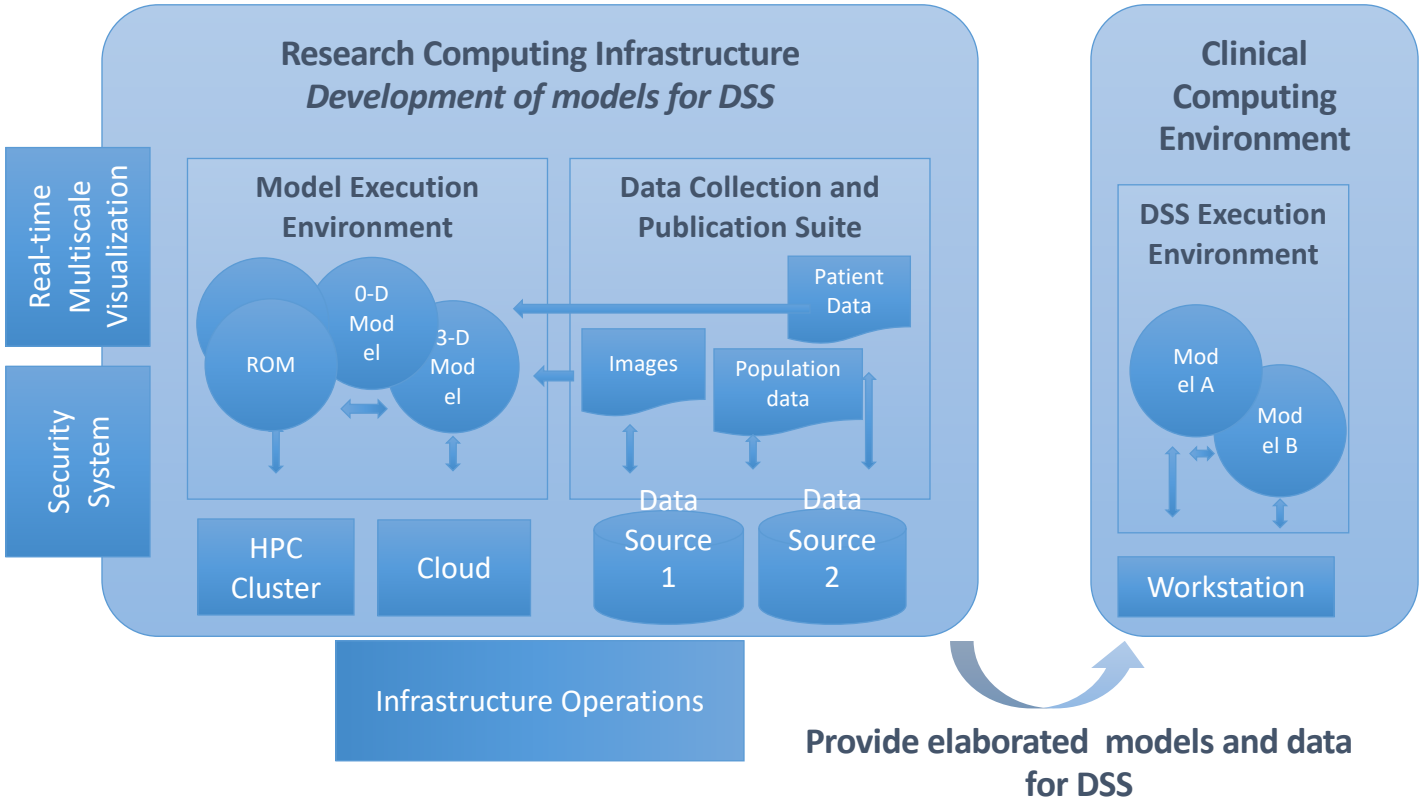
[1]ACC Cyfronet AGH, Krakow, Poland
[2]Department of Computer Science, AGH, Krakow, Poland
http://dice.cyfronet.pl/
[3]Dept Infection, Immunity & Cardiovascular Disease, University of Sheffield, UK
[4]Sheffield Teaching Hospital, NHS Foundation Trust, UK

https://www.primageproject.eu/

Outline

- Motivation: towards a decision support systems (DSS) for valvular deseases
- Requirements for processing patient cases in a research environment
- Architecture of the Model Execution Environment (MEE)
- Implementation of the MEE
- MEE in operation: patient pipeline
- MEE in operation: PRIMAGE applications
- Summary

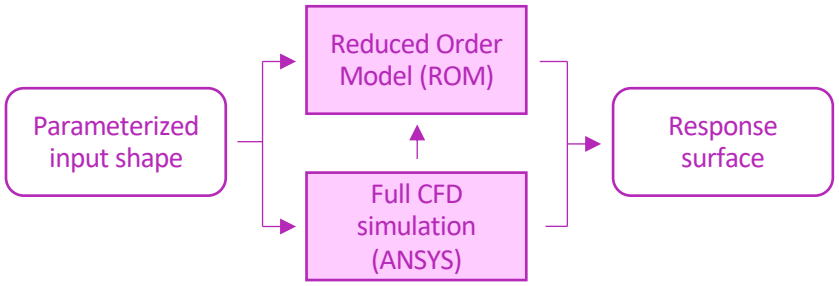
From Research Environment to DSS

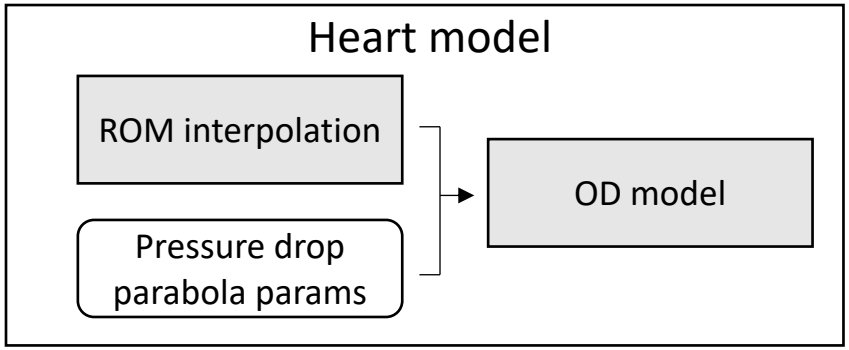


Pipelines for heart models

Data and action flow consists of:

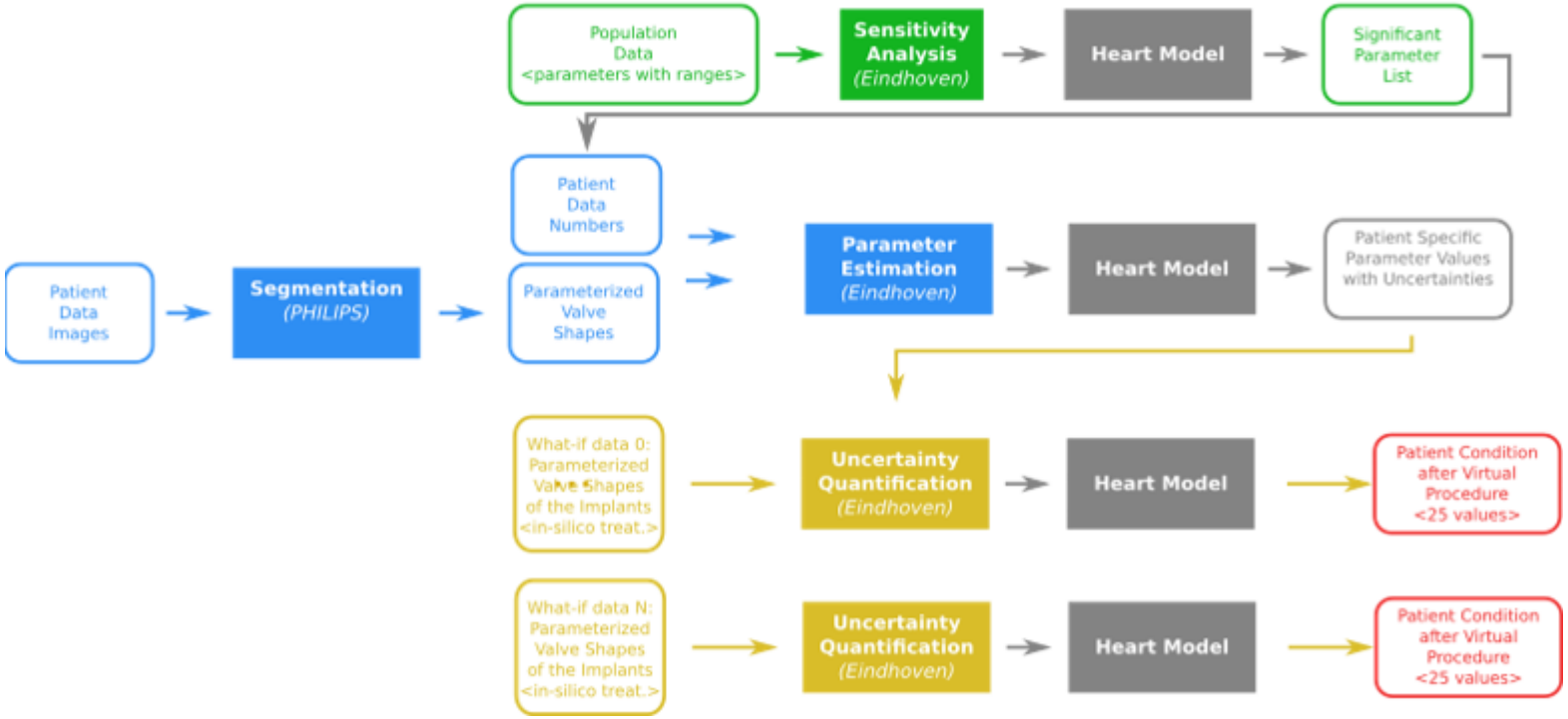
- full CFD simulations based on segmented heart images (computed using an external service)
- Reduced Order Model + interpolation to model the heart without the need for HPC resources in a clinical scenario





Pipelines for sensitivity analysis

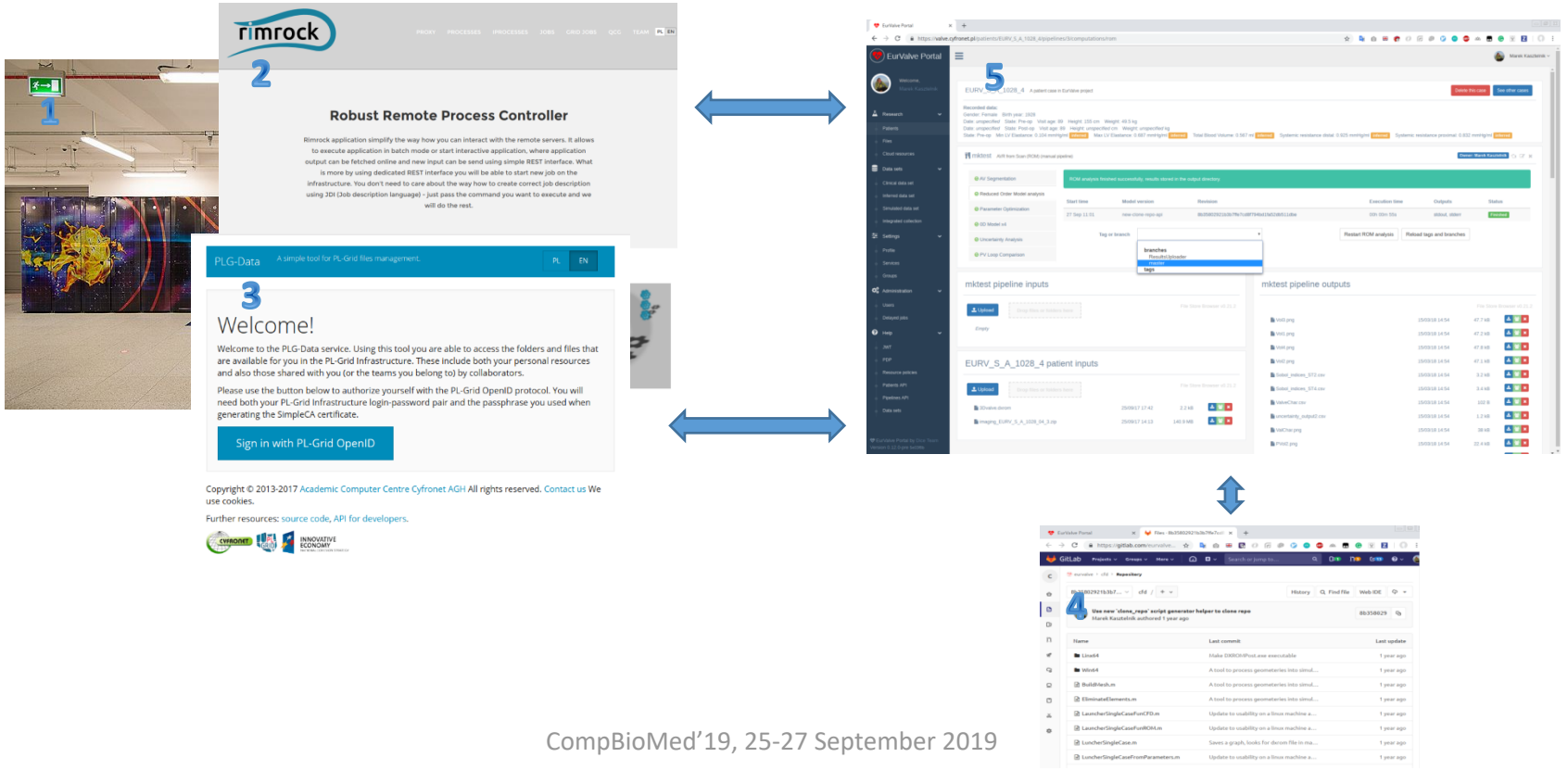
The EurValve Decision Support System is based on identification of parameters to which the simulation is sensitive, and works by running 0D models for individual patients based on precomputed valve shapes and measured values of sensitive parameters (for a set of treatment variants).



Processing patient cases – conceptual framework

Five main elements required when processing patient cases:

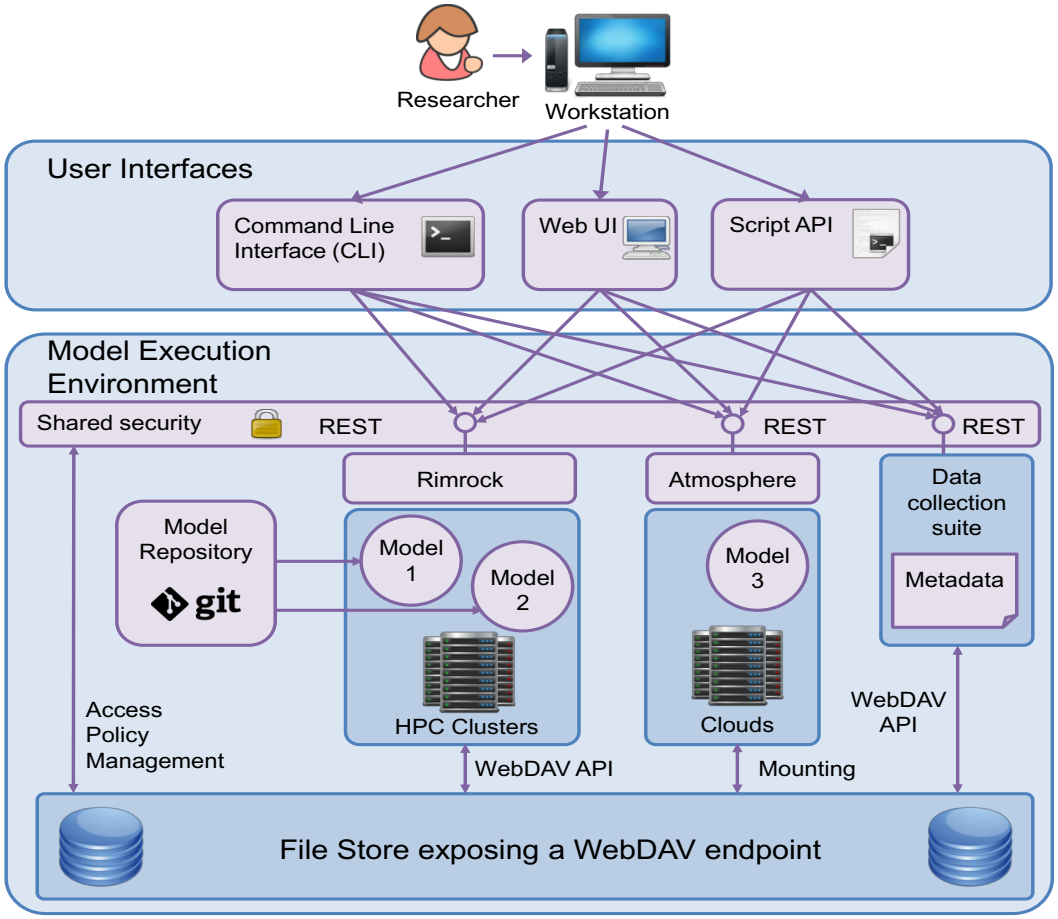
1. HPC infrastructure -> Prometheus cluster
2. Running jobs on Prometheus via REST API -> Rimrock
3. Managing data stored on Prometheus via REST API -> PLGData
4. Model repository and versioning -> GitLab
5. Managing model execution on patient data -> MEE



Motivation: reproducibility

- **Repeatability** – same team, same experimental setup; a researcher can **reliably repeat** own computation
- **Replicability** – different team, same experimental setup; an **independent group** can obtain the **same result using the author's own artifacts**
- **Reproducibility** – different team, different experimental setup; an **independent group** can obtain the **same result using artifacts which they develop completely independently**.

The Model Execution Environment (MEE)

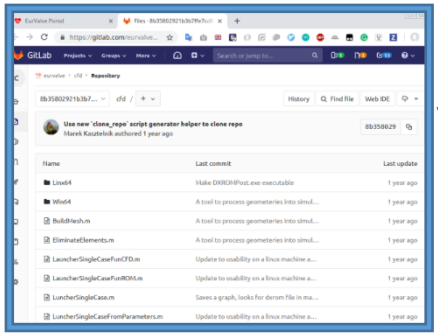


- MEE integrates the capabilities of the research branch of EurValve and is used to compute models for the clinical environment (i.e. for DSS)
- MEE can be accessed from a dedicated GUI (the EurValve Portal), through a RESTful API or through a comman-line interface, depending on the researcher's preferences.
- Computational tasks can be run on HPC resources or in a cloud environment as appropriate.
- A uniform security layer is provided.

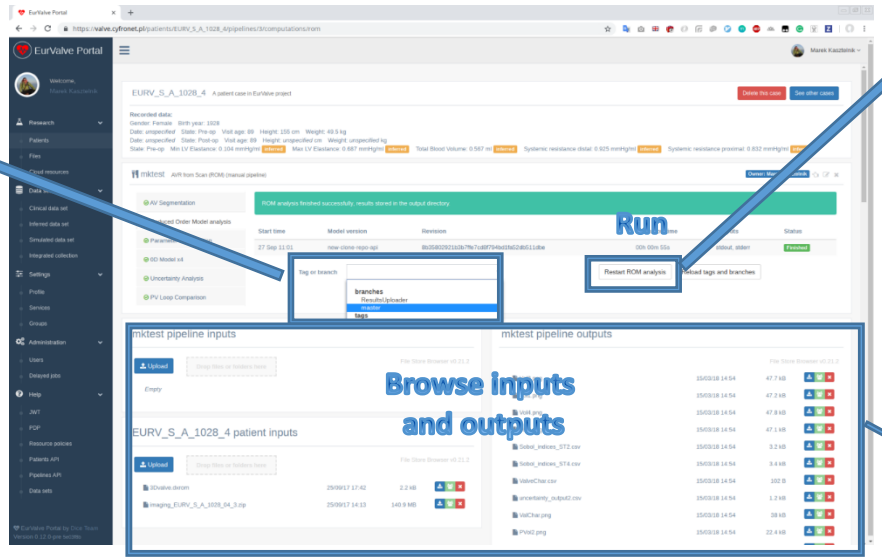
API – Application Programming Interface
REST – Representational state transfer
Rimrock – service used to submit jobs to HPC cluster
Atmosphere – provides access to cloud resources
git – distributed revision control system

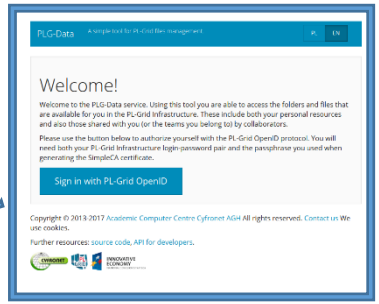
Model Execution Environment – operation

- Integrated with PLGrid infrastructure (automatic proxy generation)
- Can submit job to the Prometheus cluster
- Enables file upload and download to/from Prometheus storage
- Connected with GitLab repositories for model versioning and provenance



Select model version

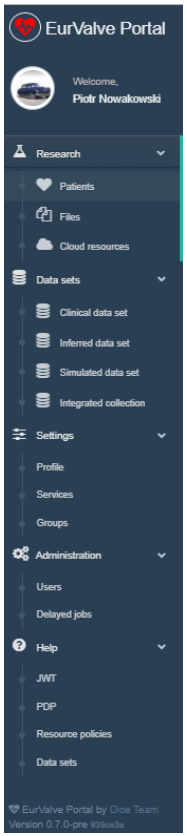




Core features of MEE

- Reproducibility, versioning, pipeline documentation
- Automation of simulation pipelines with a human in the loop for:
  - New models, new versions of models,
  - New users
- Data persistence
- Basic provenance features
- Helpful visualization of simulation flow and results
- Portability

MEE feature set



Model Execution Environment:

- Patient case pipeline integrated with File Store and Prometheus supercomputer
- *File Store for data management*
- Cloud resources based on Atmosphere cloud platform

Data sets

- Structural access to patient databases
- Query interfaces for real, simulated and inferred data

Security configuration

- Service management – for every service dedicated set of policy rules can be defined
- User Groups – can be used to define security constraints

REST API

- Creating a new user session – as a result, new JWT (JSON Web Token) tokens are generated for credential delegation
- PDP – Policy Decision Point: check if user has access to concrete resource
- Resource policies – add/remove/edit service security policies

MEE extensibility

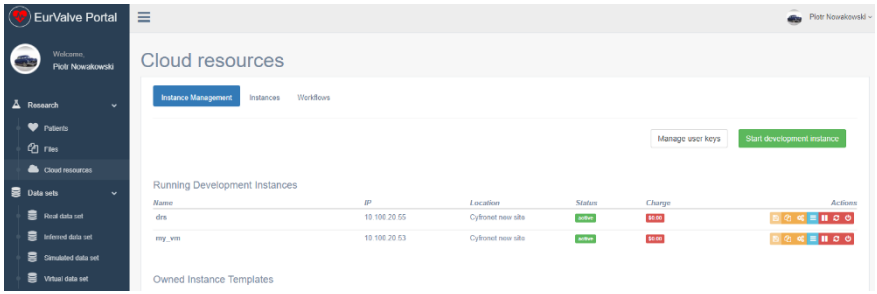
Additional modules can be implemented:

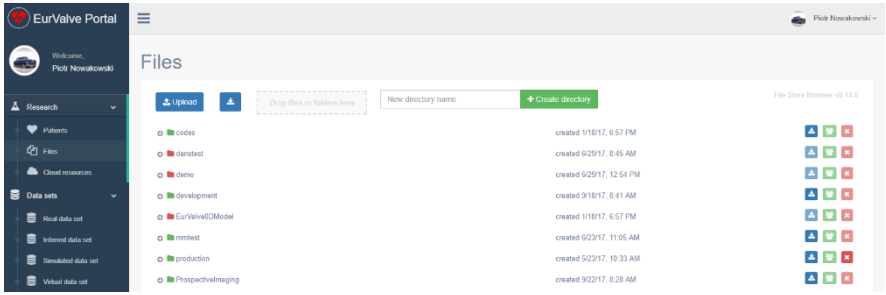
- As applications deployed on the Prometheus supercomputer
- As external services communicating with the platform via its REST interfaces
- As virtual machines deployable directly in the Cyfronet cloud via the Atmosphere extension of the MEE

Encapsulating pipeline steps as HPC tasks:

- Scripts are run on the Prometheus supercomputer via the Rimrock extension
- Files uploaded to the FileStore (e.g. using MEE GUIs) can be accessed on Prometheus nodes via **curl**, leveraging the WebDAV interface provided by FileStore
- Any result files can also be uploaded directly to FileStore from the Prometheus computational nodes
- External tools can be used to monitor job completion status e.g. by periodically scanning FileStore content

Generic MEE tools





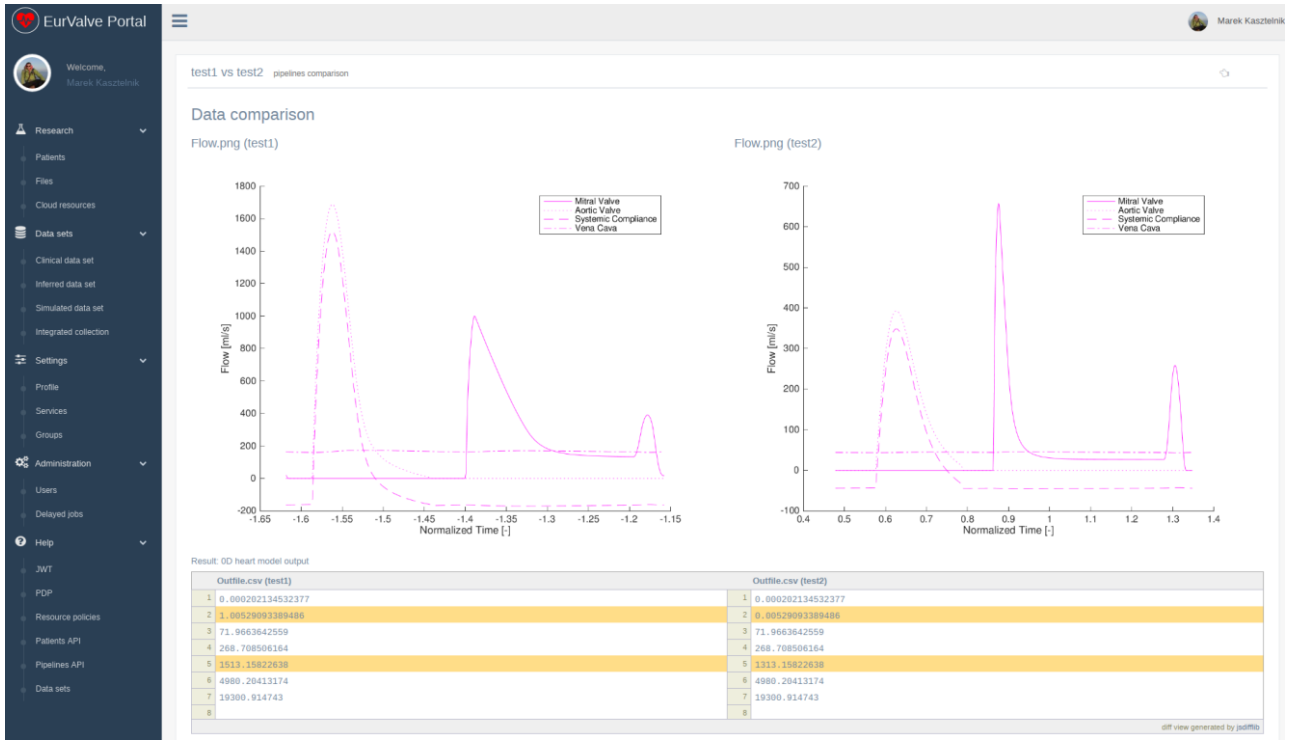
Cloud resources

- Based on Atmosphere cloud platform
- Can start/stop/suspend virtual machine on cloud infrastructure
- Can save running existing machine as template
- (Future) can share templates with other users

File Store

- Basic file storage for the project
- Ability to create new directories and upload/download files
- Can share directories with other users or groups of users
- Can be mounted locally using WebDav clients
- The File Browser GUI can also be embedded in other views

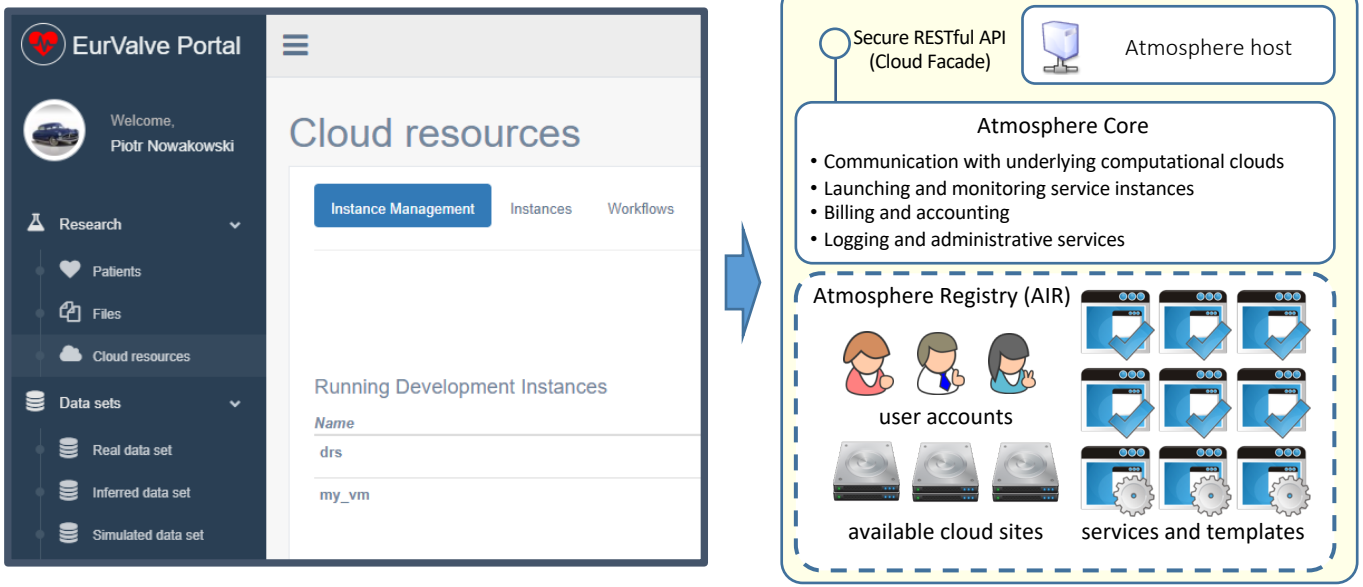
Comparing pipeline results



Result comparison feature

- Run a pipeline multiple times for varying datasets
- Compare the outcomes and identify differences
- Helpful GUI for side-by-side comparisons
- Supports textual and graphical visualization of results

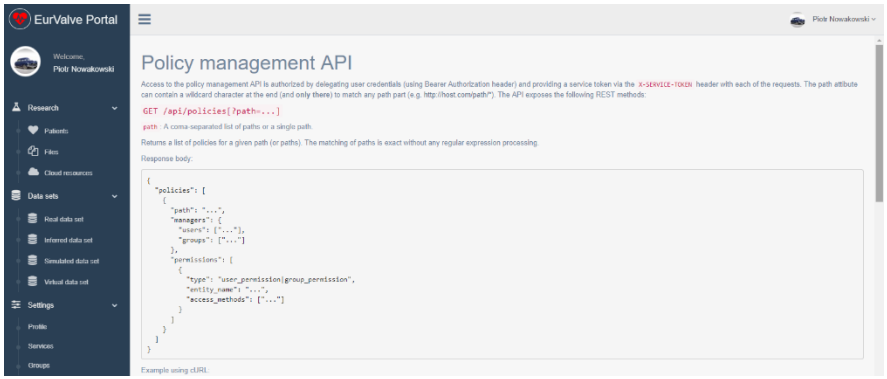
Cloud access via Atmosphere



Access to cloud resources

- The Atmosphere extension provides access to cloud resources in the EurValve MEE
- Applications can be developed as virtual machines, saved as templates and instantiated in the cloud
- The extension is available directly in the MEE GUI and through a dedicated API
- Atmosphere is integrated with EurValve authentication and authorization mechanisms

Accessing MEE via REST API



Generate user JWT Token

- User (or other service) can retrieve new JWT token by passing username and password
- JWT token can be used for user credential delegations by external EurValve services

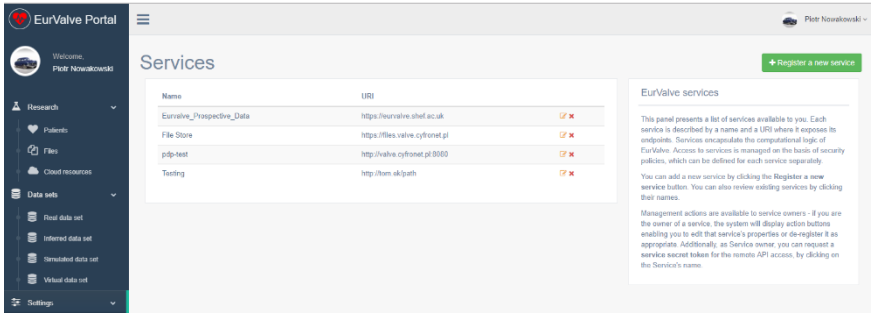
PDP API

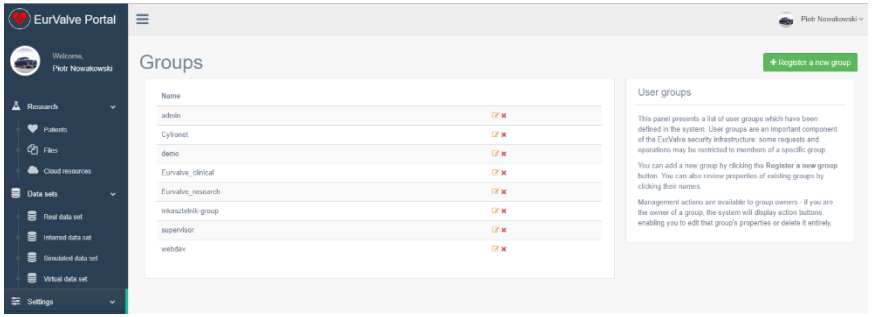
- Check if user has right to access a specific resource

Resource policy management

- Create/edit/delete local policies by external EurValve service on user behalf
- Currently integrated with File Store
- Initial ArQ integration tests underway

MEE security management UIs





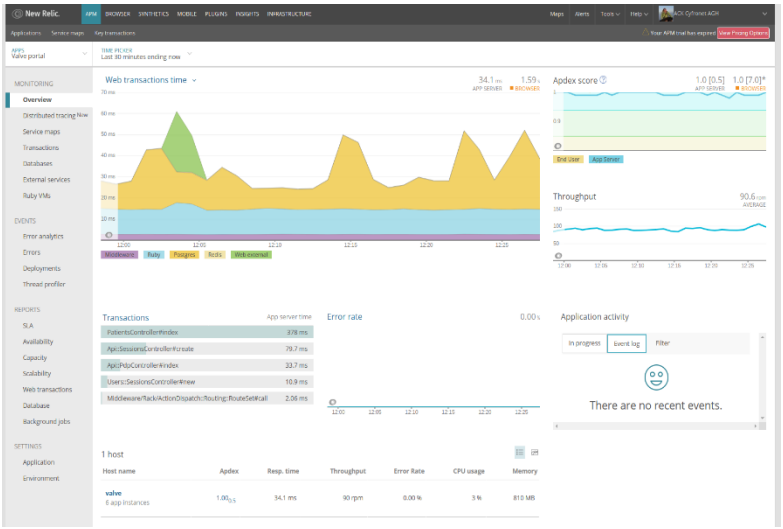
Services

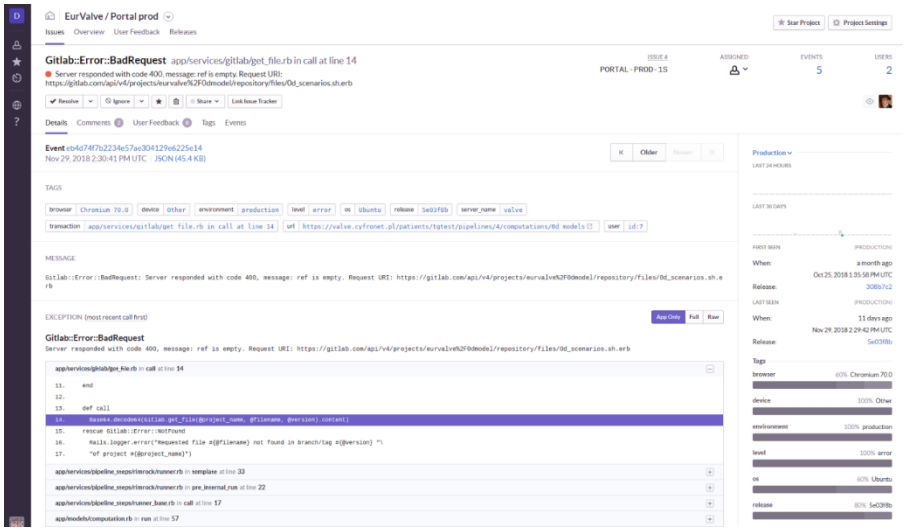
- Basic security unit where dedicated security constraints can be defined
- Two types of security policies:
  - Global – can be defined only by service owner
  - Local – can be created by the service on the user's behalf

Groups

- Group users
- Dedicated portal groups:
  - Admin
  - Supervisor – users who can approve other users in the portal
- Generic groups:
  - Everyone can create a group
  - Groups can be used to define security constraints

Platform availability monitoring

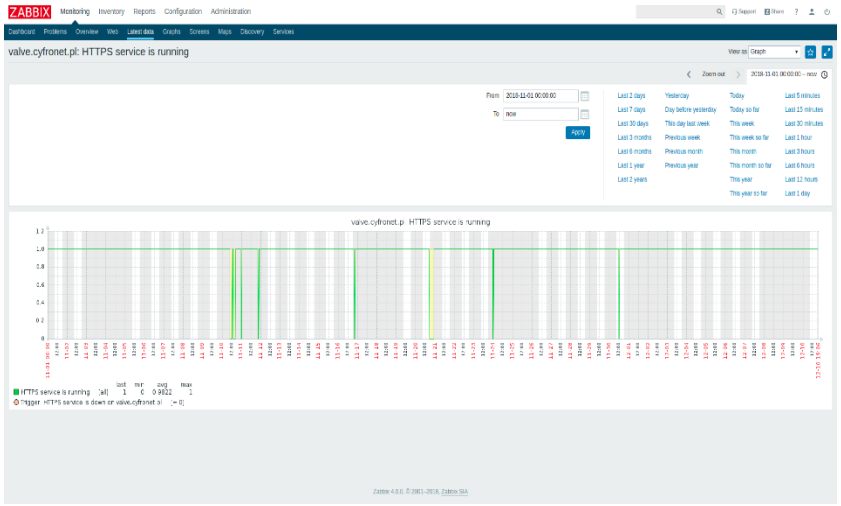


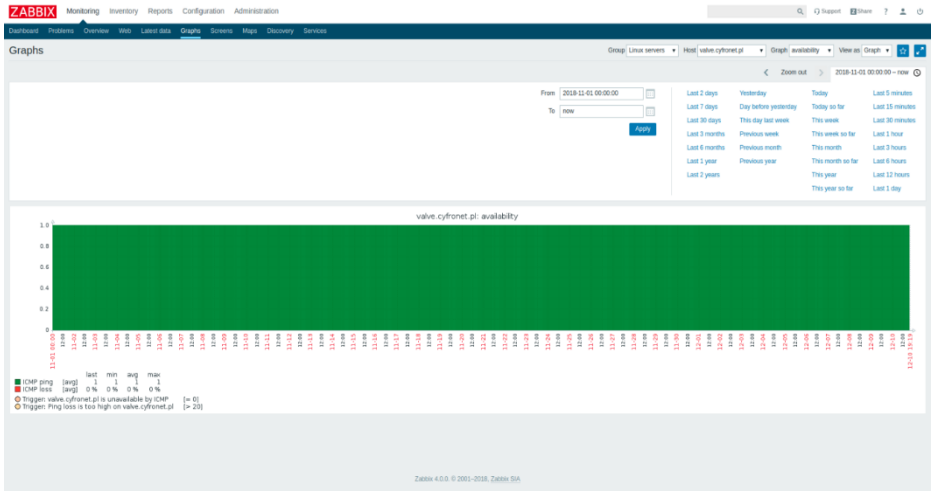


Monitoring services

- NewRelic and Sentry used to monitor platform availability
- Detect application errors as well as infrastructural failures
- Alert administrators of any anomalous conditions

Platform responsiveness monitoring





Monitoring services

- Zabbix is used to monitor responsiveness, service uptime and utilization data
- The monitoring service also evaluates the risk of failures caused by exhaustion of hardware resources
- As with Sentry/NewRelic, alerts can be dispatched to system administrators

The Patient Case Pipeline: an EurValve workflow

1. Segmentation – to start this calculation, a zip archive with a dedicated structure needs to be created and transferred into the File Store input directory. Next, the output directory needs to be monitored for computation output.
2. Reduced Order Model analysis – based on the results of the segmentation step, a ROM simulation is executed and its results uploaded to the File Store.
3. Parameter Optimization – a technical step which prepares suitable parameters for the 0D model sequence
4. 0D model sequence – runs four versions of the 0D model analysis for various input datasets
5. Uncertainty Quantification – Matlab script which can include the 0D Heart Model. It will be executed on the Prometheus supercomputer, where input files will be transferred automatically from the File Store. Results are transferred back from Prometheus to the File Store.
6. Output visualization – produces actionable visualization of the 0D model output data based on File Store contents.

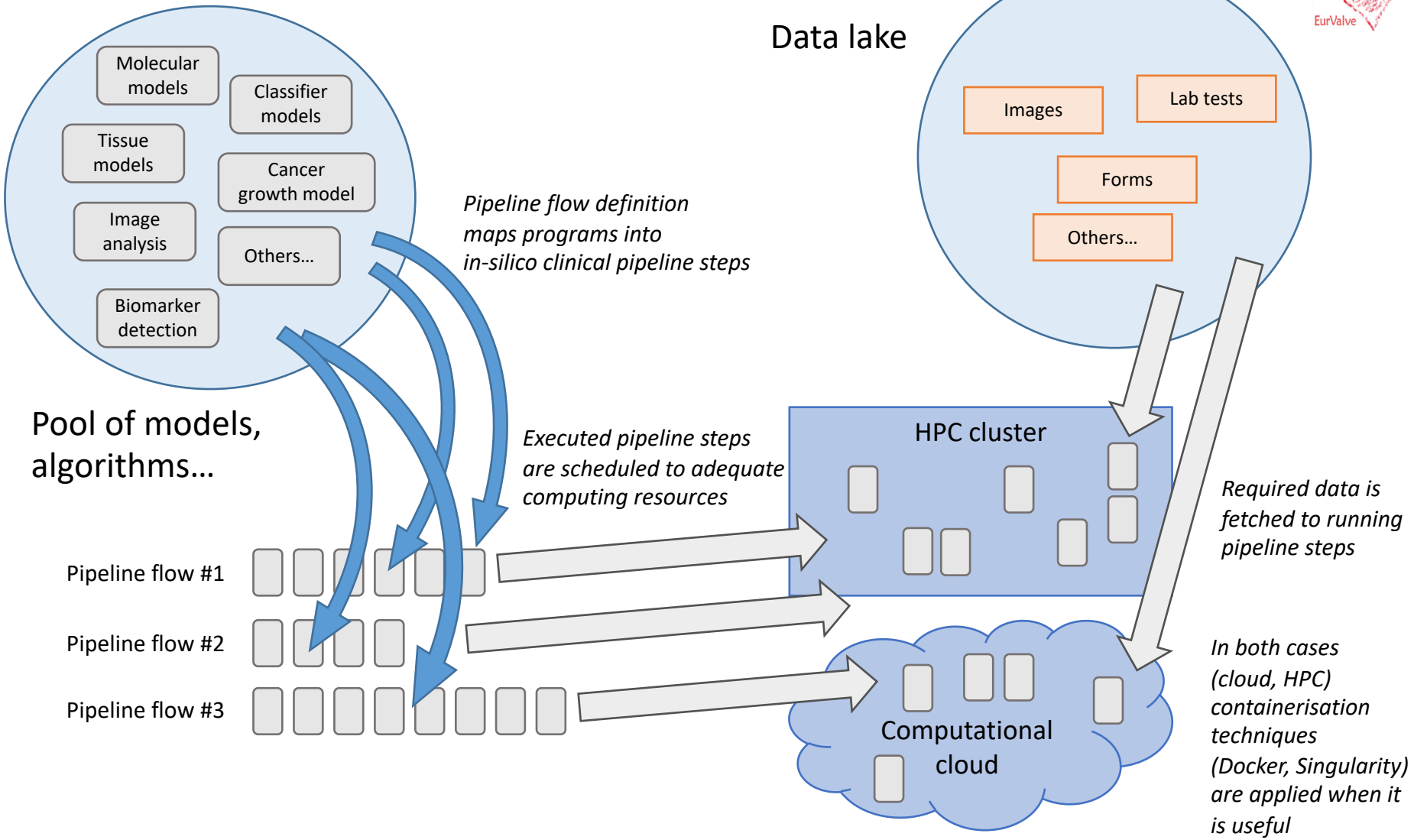
Patient Case Pipeline high-level building blocks:

- Service-driven computation (such as Segmentation) – use case: upload file to remote input directory, monitor remote output directory for results; data is processed by an external service
- Scripts started on Prometheus supercomputer – use case: transfer script and input files from File Store to the cluster, run job, monitor job status, once the job has completed – transfer results from the cluster to File Store (examples: 0D Heart Model, Uncertainty Quantification, CFD simulation)

Further evolution of the environment – PRIMAGE workflows

- **PRedictive In-silico Multiscale Analytics to support cancer personalized diaGnosis and prognosis** (PRIMAGE)
  - to develop an **environment for predictive, personalized medicine, focused on cancer treatment**
  - Data infrastructures, imaging biomarkers and models for in-silico medicine research will be validated in the context of two pediatric cancers: **Neuroblastoma**, the most frequent solid cancer of early childhood, and **Diffuse Intrinsic Pontine Glioma**, the leading cause of brain tumor-related death of children
- PRIMAGE technological platform, to be delivered by several partners, will combine containerisation techniques with deployment of in-silico codes and machine learning models to various computing cloud and HPC supercomputing resources
- Computational workflows will be grouped in pipelines to reflect clinical practice and requirements as closely as possible
- Pipelines will include computational steps, running sequentially or in parallel, using various computational infrastructures, some of which will use containers where beneficial

Containers in PRIMAGE



Computations performed

In the scope of the EurValve project, the Model Execution Environment and the underlying computational infrastructure – including the Prometheus cluster at ACC Cyfronet AGH – was used to develop the Reduced Order Model and perform patient simulations.

- The EurValve research environment was used to process computational workflows for 67 retrospective patient cases. Multiple simulations were carried out for each patient, resulting in a total of over 300 thousand CPU-hours facilitated by the HPC resources at ACC Cyfronet AGH.
- Patient data and simulation results are stored in 72,953 files using 265GB of disk storage.
- Tabular Data Store:
  - Clinical data items: 357 records with 305 data items each
  - Simulated results: 1457 records with 78 data items each

All of the above is managed and accessible through the MEE – see the demonstration at http://www.eurvalve.eu/index.php/eurvalve-video/.

MEE: HPC grant usage

Overall, we applied for 5 computation grants and consumed 305,000 CPU hours:

Grant name	Start date	End date	Requested CPU hours	Consumed CPU hours
EurValve 1	12.01.2016	12.01.2017	25000	22869
EurValve 2	17.02.2016	17.02.2017	500000	38018
EurValve 3	22.02.2017	22.02.2018	150000	94279
EurValve 4	05.03.2018	05.03.2019	150000	150000
EurValve 5	01.02.2019	30.06.2019	150000	

Computational grants

- Required to carry out computations on Cyfronet/PLGrid resources
- Negotiated by Cyfronet on behalf of the EurValve consortium on an annual basis
- Consumed by running pipelines and processing patient data in MEE
- Free of charge („payment” is remitted in the form of publications which acknowledge PLGrid/Cyfronet contributions)

MEE services at Cyfronet

- **EurValve Project Website at Cyfronet AGH**
  - URL: http://dice.cyfronet.pl/projects/details/EurValve
- **EurValve Portal**
  - URL: https://valve.cyfronet.pl
  - Registration at: https://valve.cyfronet.pl/users/sign_up
- **EurValve File Store**
  - URL (docs): https://files.valve.cyfronet.pl
  - WebDAV endpoint (portal account required): https://files.valve.cyfronet.pl/webdav

Recorded demos of MEE

- Logging in to EurValve and PLGrid systems – https://youtu.be/4I907aAOCvU
- File Store Browser – https://youtu.be/-6fXG0am6iE
- Distributed Cloud File Store – https://youtu.be/FTF-QaI5ZZQ
- Services, security, restricted access – https://youtu.be/-6fXG0am6iE
- Cloud Resource Access – https://youtu.be/A4wkxFCRLak
- Patient case
  - https://youtu.be/SIwpxdoQYWw (patient case)
  - https://youtu.be/j0Nu-E-0elE (pipeline diff)
- Integration of computational services – https://youtu.be/SIwpxdoQYWw

In summary...

We are observing an evolution of computational paradigms: **physical computers** -> **cloud VMs** -> **containers** (much lighter than clouds) -> **serverless** -> **serverless containers** (e.g. AWS Fargate, Google Serverless Containers).

The consequence is an increasing number of abstraction layers.

Further development of MEE includes:

1. Pipeline execution management: organizes set of models into a single sequential execution pipeline with files as the main data exchange channel
   - Model development organization through structuralization
   - Retention of execution and development history
2. Computation execution diff: an adequate tool for model developers to compare two different model executions, revealing any changes along with their impact on results
   - Dedicated comparison software for specific types of results
   - Easier problem detection and manual validation
3. Multiorganizational support: capability to switch between workbenches belonging to different organizations/projects which make use of MEE, and to run pipelines independently within the context of each organization.

http://dice.cyfronet.pl

https://www.primageproject.eu/

http://www.process-project.eu

http://www.eurvalve.eu

http://www.vph-share.eu

https://sano.science