Environments for Collaborative Applications on Distributed e-Infrastructures

Marian Bubak

Department of Computer Science & ACC Cyfronet AGH

AGH University of Science and Technology, Krakow

and

Institute for Informatics, University of Amsterdam

http://dice.cyfronet.pl

www.science.uva.nl/~gvlam/wsvlam

HPC, Cetraro, 25-29 June 2012

Coauthors

Bartosz Balis, Tomasz Bartynski, Eryk Ciepiela,

Grzegorz Dyk, Wlodek Funika, Tomasz Gubala, Daniel Harezlak,

Tomasz Jadczyk, Marek Kasztelnik, Joanna Kocot, Maciej Malawski,

Jan Meizner, Piotr Nowakowski, Katarzyna Rycerz

Adam Belloum,

Mikolaj Baranowski, Reggie Cushing, Spiros Koulouzis,Jakub Moscicki

Outline

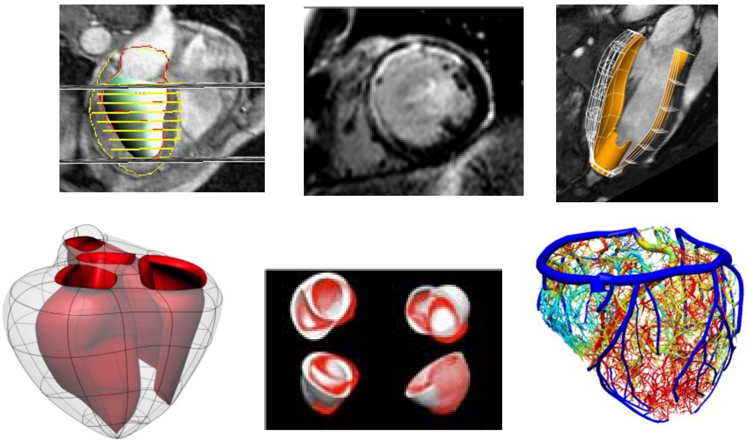
- Motivation
- Overview of applications
- Complexity of computing resources
- Environments for collaborative applications
  - Scripting approach
  - Common Information Space (service orientation)
  - Building and running multiscale applications
  - Federating clouds and HPC
- Executable papers
- Summary

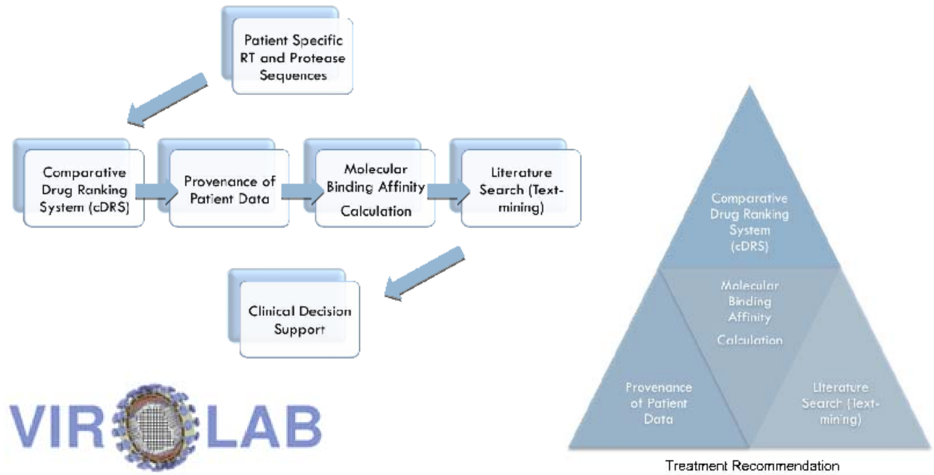
Motivation

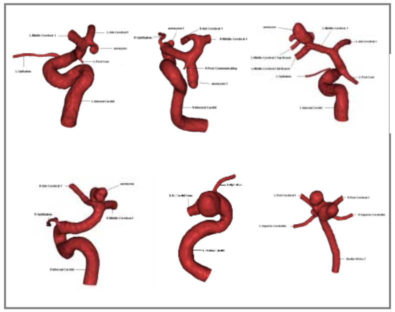
- **Scientific applications**
  - Collaborative (see: 2011 Nobel laureate Brian P. Schmidt)
  - Compute- and data-intensive (4th paradigm)
  - Used in dynamic scenarios – experiments
  - Multiscale, multiphysics
  - Holistic – system level science
  - Various levels of coupling and composition types
  - Legacy codes in many programming languages
  - Science 2.0 – social aspects
  - Linking to publications

- **Computing infrastructures**
  - Distributed and heterogeneous
  - Resources are shared, possibly between different organizations
  - Resources may dynamically change and may be not reliable
  - There is no single middleware
  - Collaborations in virtual organizations can be highly dynamic

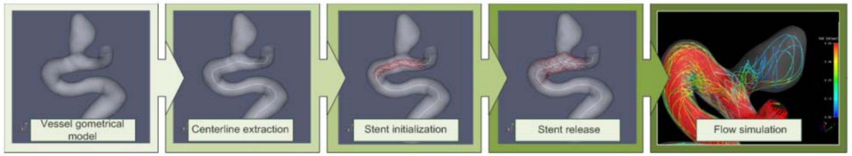
VPH – Towards Patient Avatar

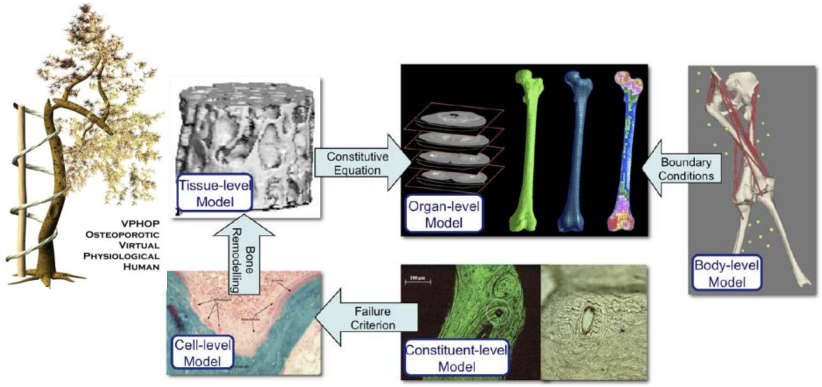




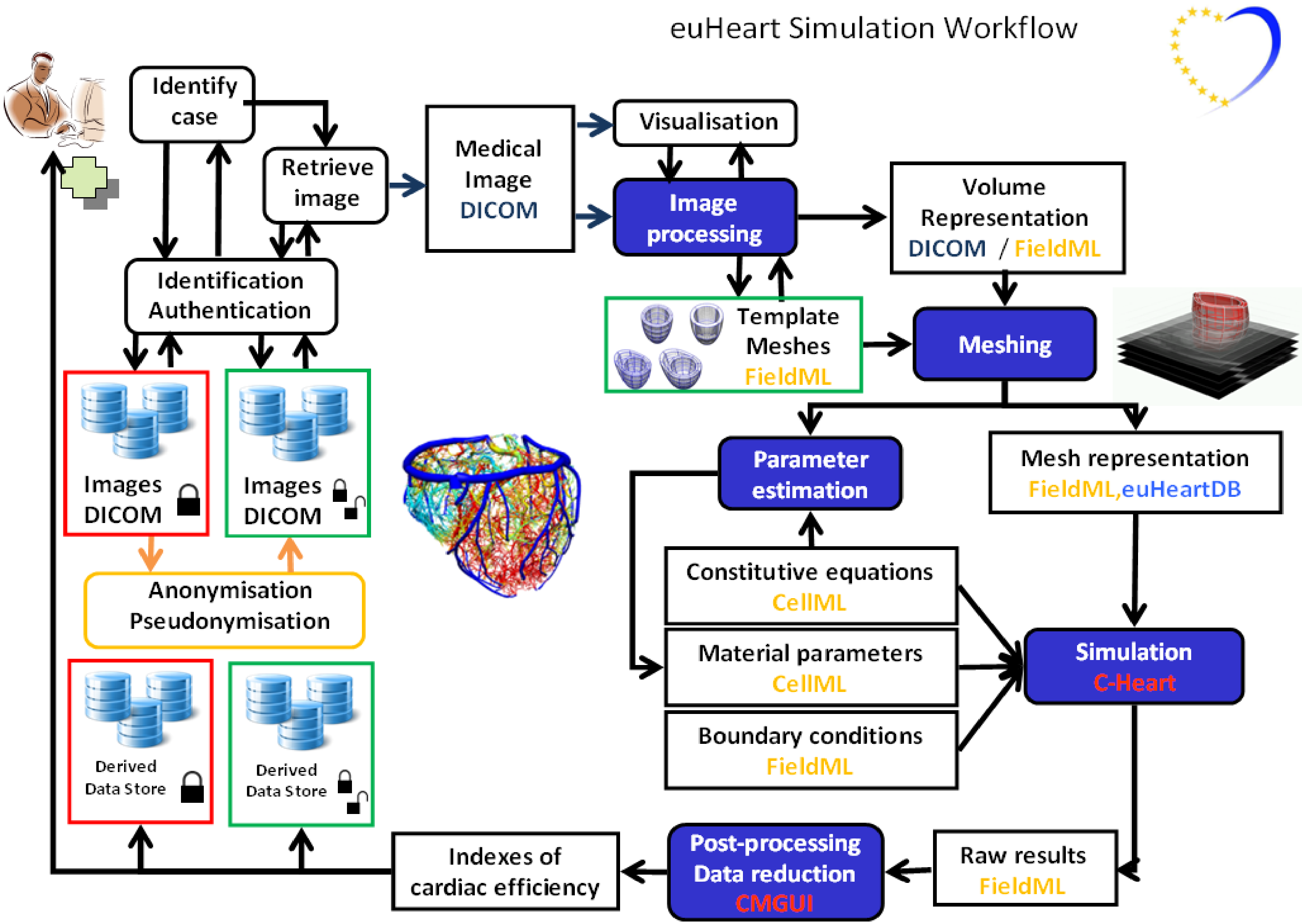


aneurIST
Integrated biomedical informatics for the management of cerebral Aneurysms





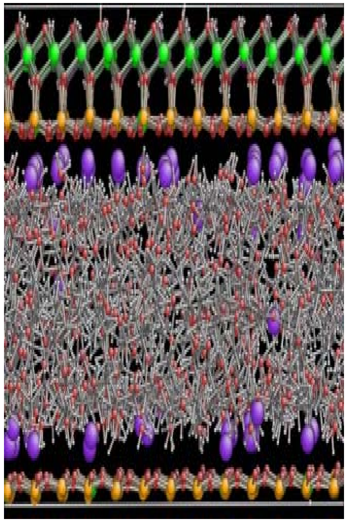
euHeart Simulation Workflow



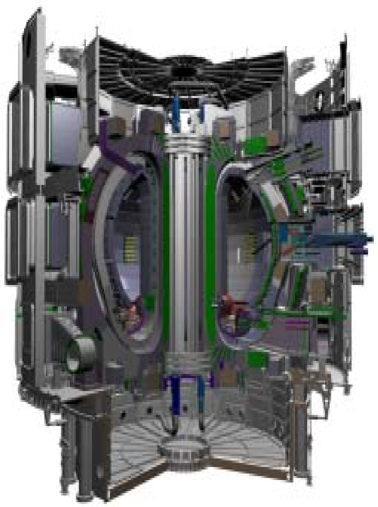
Multiscale Applications

virtual physiological human

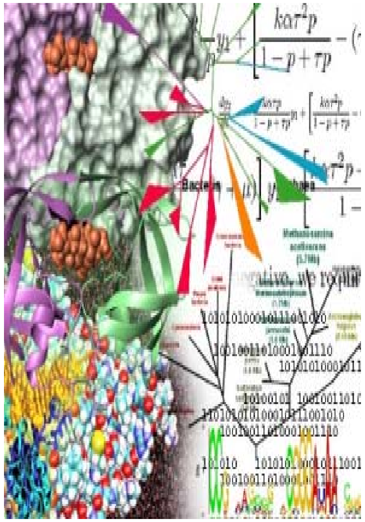
hydrology



nano material science



fusion

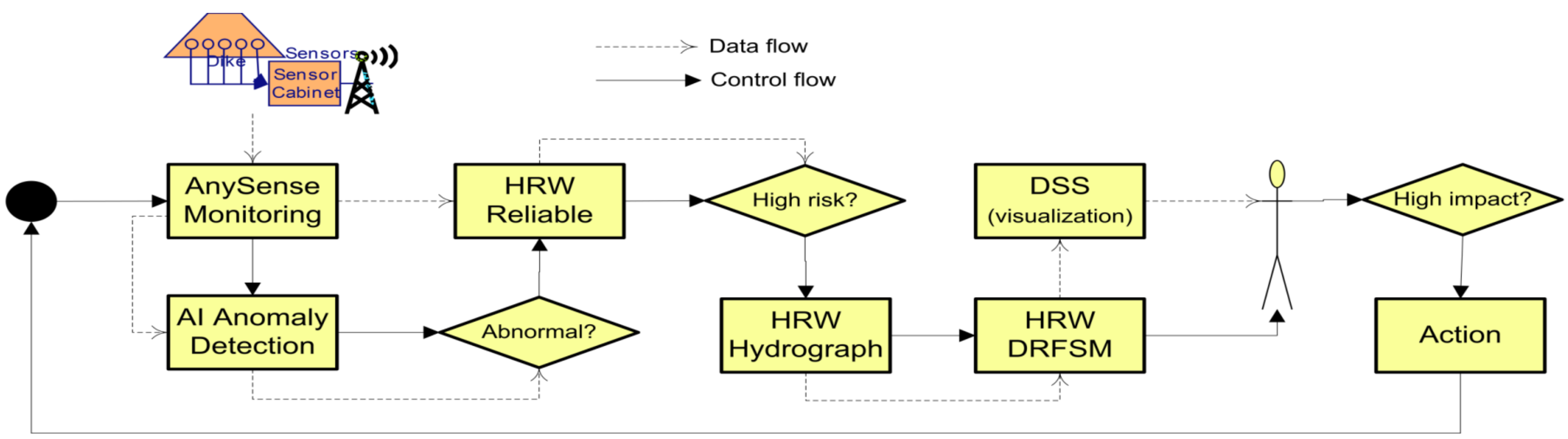


computational biology

- When simulated in three dimensions, they usually require large scale computing capabilities.
- Such large scale hybrid models require a distributed computing ecosystem, where parts of the multiscale model are executed on the most appropriate computing resource.

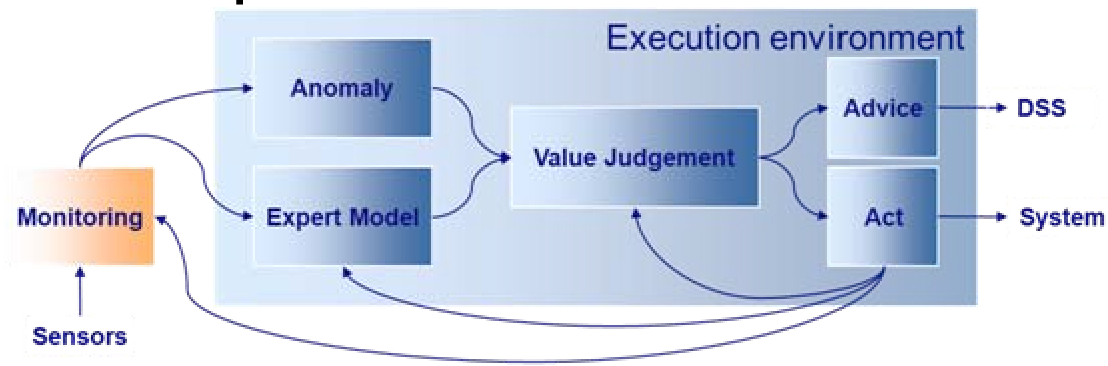
Flood Early Warning System

- Monitoring of dikes using wireless sensors
- AI-based detection of sensor signal anomalies
- Dike failure prediction
- Simulation of inundation due to failure
- Visualization and user interactions on Multi-touch Tables



Early Warning Systems

- Facilitate creation, deployment and robust operation of Early Warning Systems
- **E**arly **W**arning **S**ystem: any system working according to four steps:
  - Monitoring
  - Analysis
  - Judgment
  - Action

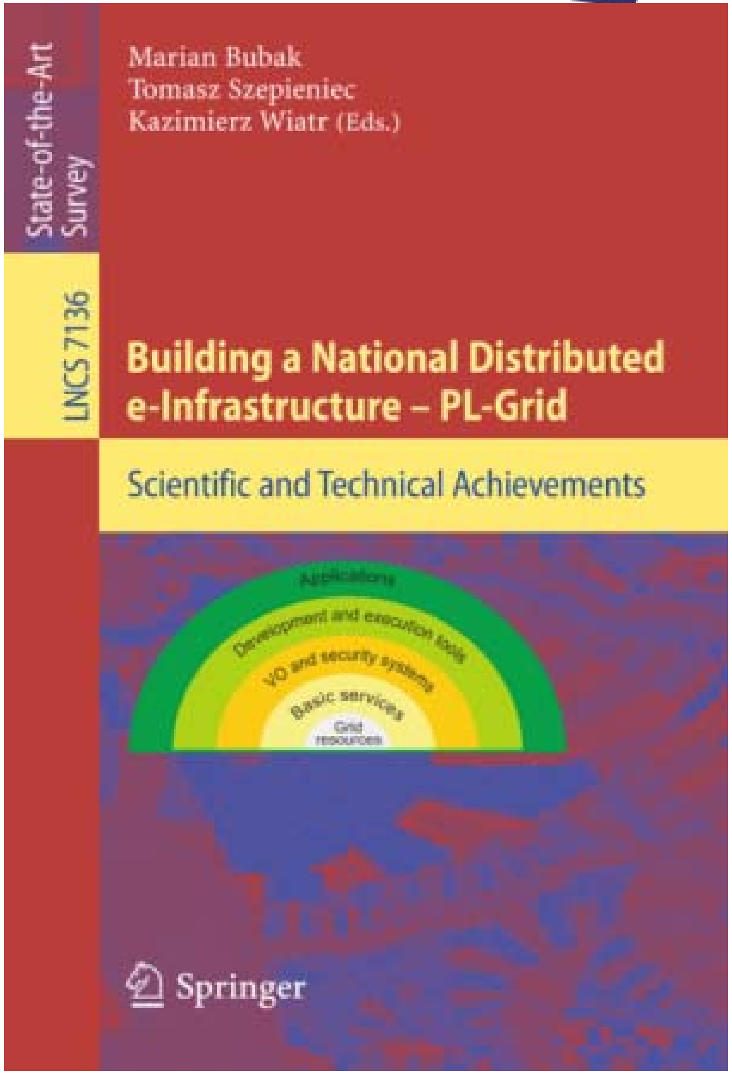


- Example: **environmental monitoring**
- Also: cloud infrastructure monitoring, EWS self-monitoring

B. Balis, M. Kasztelnik, M. Bubak, T. Bartynski, T. Gubala, P. Nowakowski, J. Broekhuijsen: The UrbanFlood Common Information Space for Early Warning Systems, In: Elsevier Procedia Computer Science, vol 4, pp 96-105, ICCS 2011.

Resources: eg PL-Grid, PLGrid+

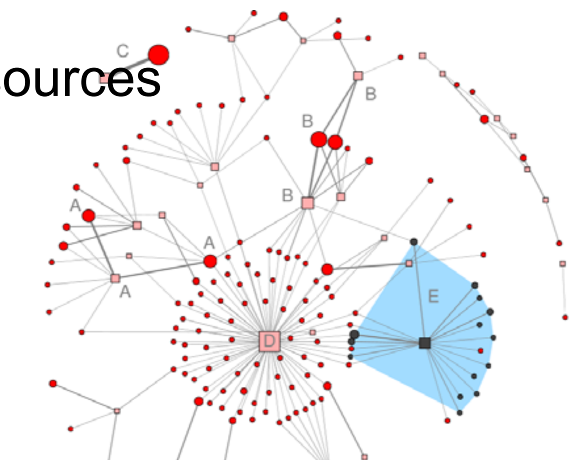
- First working NGI in Europe in EGI.eu (March 2010)
- No of users (March 2012): 900+
- No jobs per month: 750 000 – 1 500 000
- Resources
  - Computing power: 230 TFlops
  - Storage: ca. 3600 TBytes
  - High level of availiability and realibility of the resources
- Services and tools
  - Efficient Resource Allocation
  - Experimental Workbench
  - Grid Middleware
  - Scientific Software Packages
  - User support: helpdesk system
- Various, well-performed dissemination activities



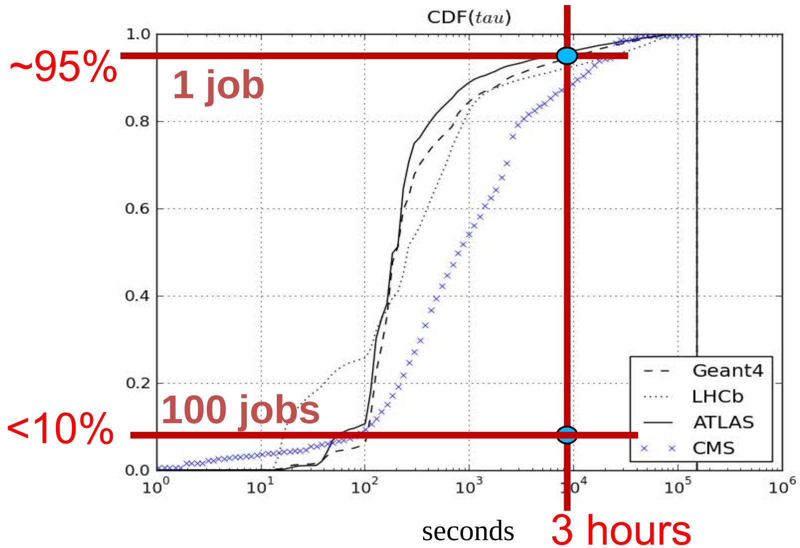
www.plgrid.pl

Spatial and Temportal Dynamics in Grids

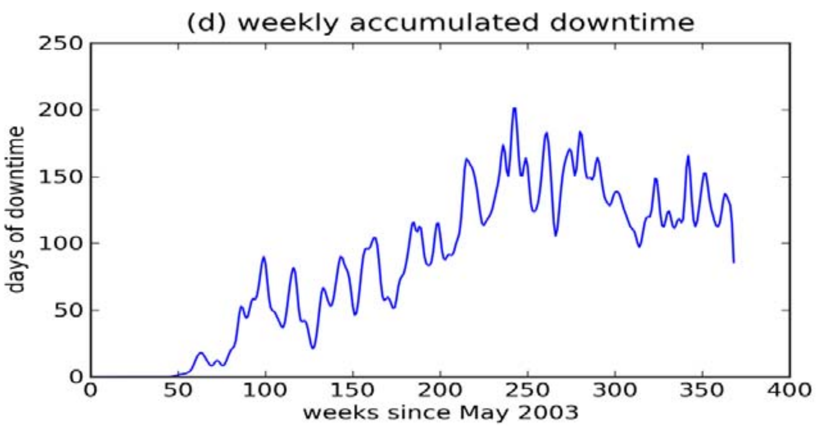
- <u>Grids increase research capabilities for science</u>
- Large-scale federation of computing and storage resources
- 300 sites, 60 countries, 200 Virtual Organizations
- 10^5 CPUs, 20 PB data storage, 10^5 jobs daily



- However operational and runtime dynamics have a negative impact on reliability and efficiency



long and unpredictable job waiting times

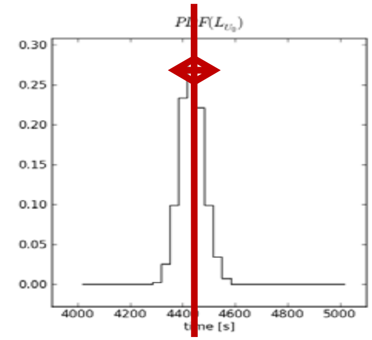


asynchronous and frequent failures and hardware/software upgrades

J.T.Moscicki, *Understanding and mastering dynamics in Computing Grids,* UvA PhD thesis, promoter: M. Bubak, co-promoter: P. Sloot; 12.04.2011

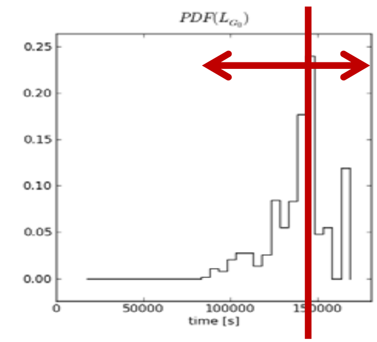
User-level Overlay with Late Binding Scheduling

- Improved job execution characteristics
- HTC-HPC Interoperability
- Heuristic resource selection
- Application aware task scheduling



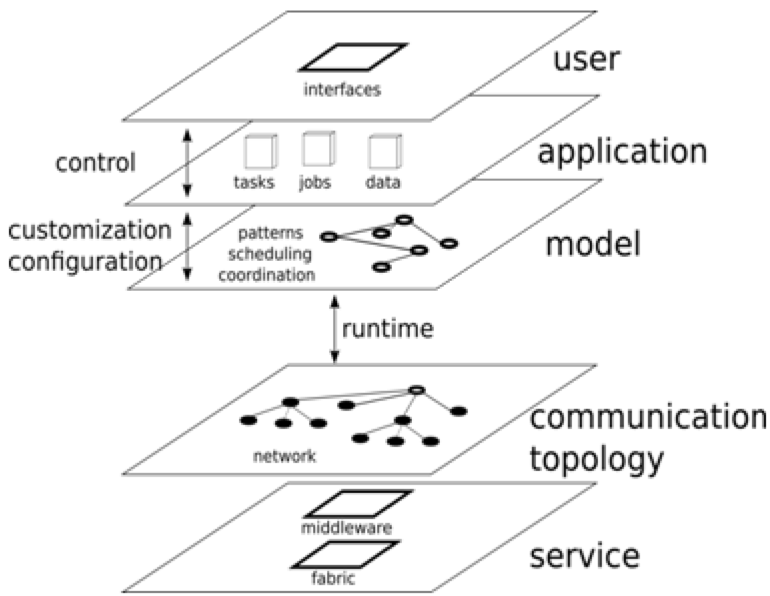
1.5 hours

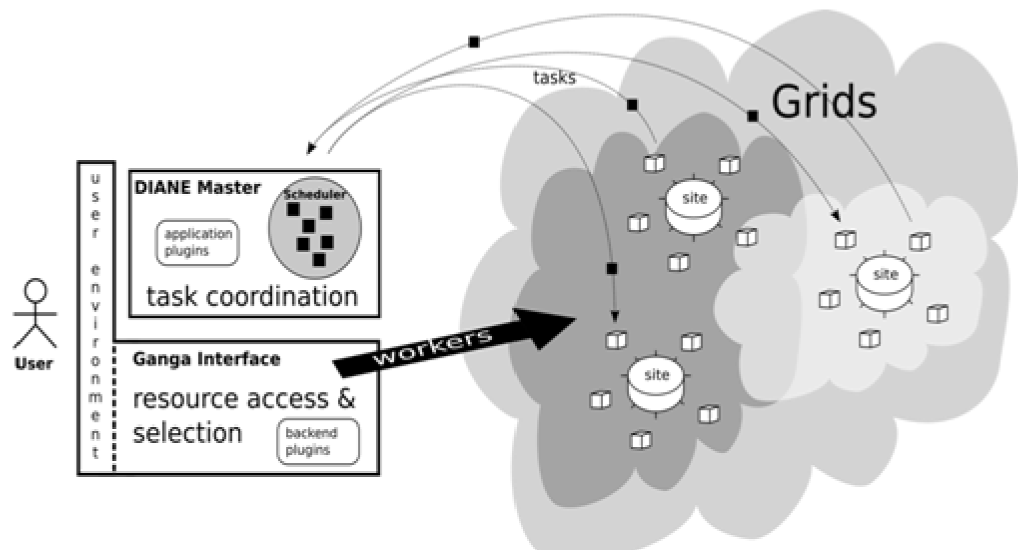
Completion time with late binding.



40 hours

Completion time with early binding.

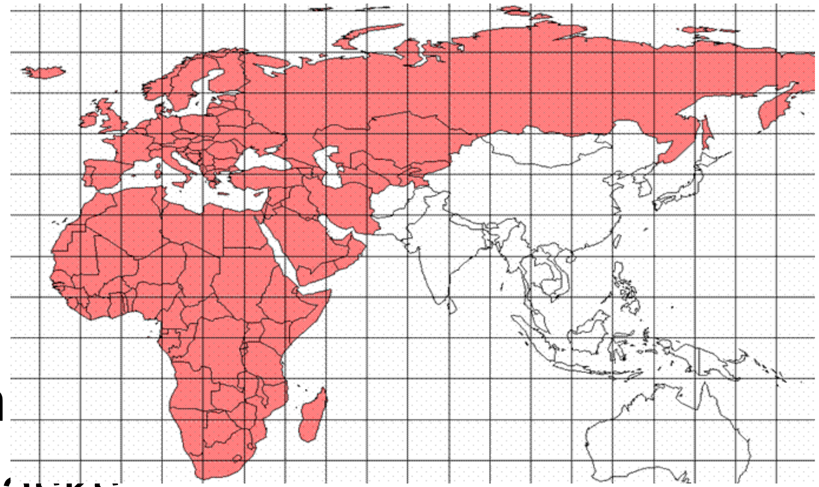




J.T.Moscicki, M.Lamanna, M.Bubak, P.M.A.Sloot: *Processing moldable tasks on the Grid: late job binding with lightweight user-level overlay*, FGCS 27(6) pp 725-736, 2011

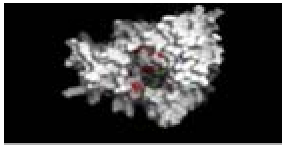
User-level Overlay in action

- Improved Quality of Service
- Scaling out on demand
- Short-deadline computing
- Web portal intergation
- Workflow service integration
- Autonomous processing of tasks



Geographical extent of Grid-enabled planning of radio broadcasting services (ITU RRC06)

ATLAS

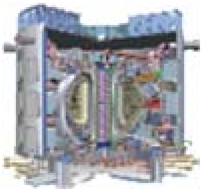


Academia Sinica
Genomics Research Center

LHCb

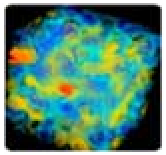
Geant 4

HARP



Fusion

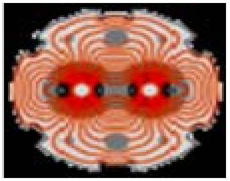
ITU



LQCD

Garfield

med austron



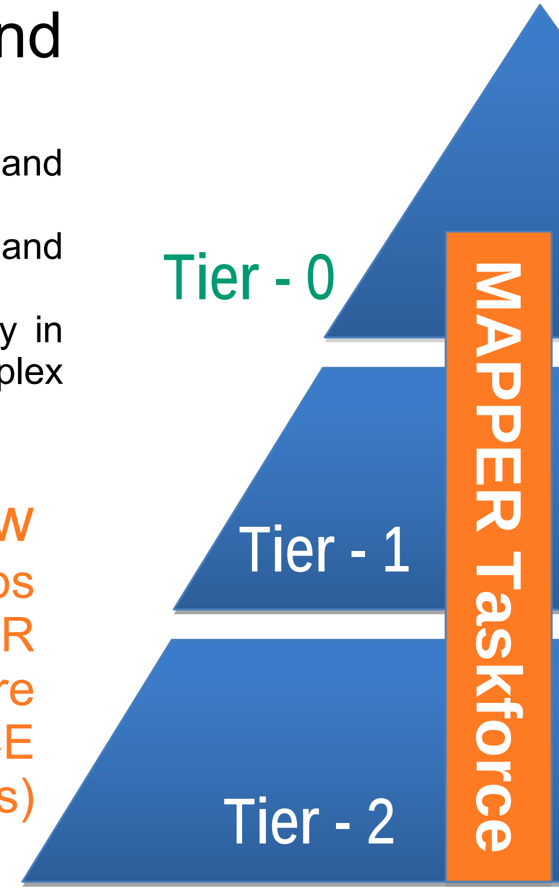
Sixtrack/LHC

COMPASS

J.T.Moscicki, M.Bubak, H.C.Lee, A.Muraru, P.M.A.Sloot: Quality of Service on the Grid with User Level Scheduling, best poster award, CGW, 2007

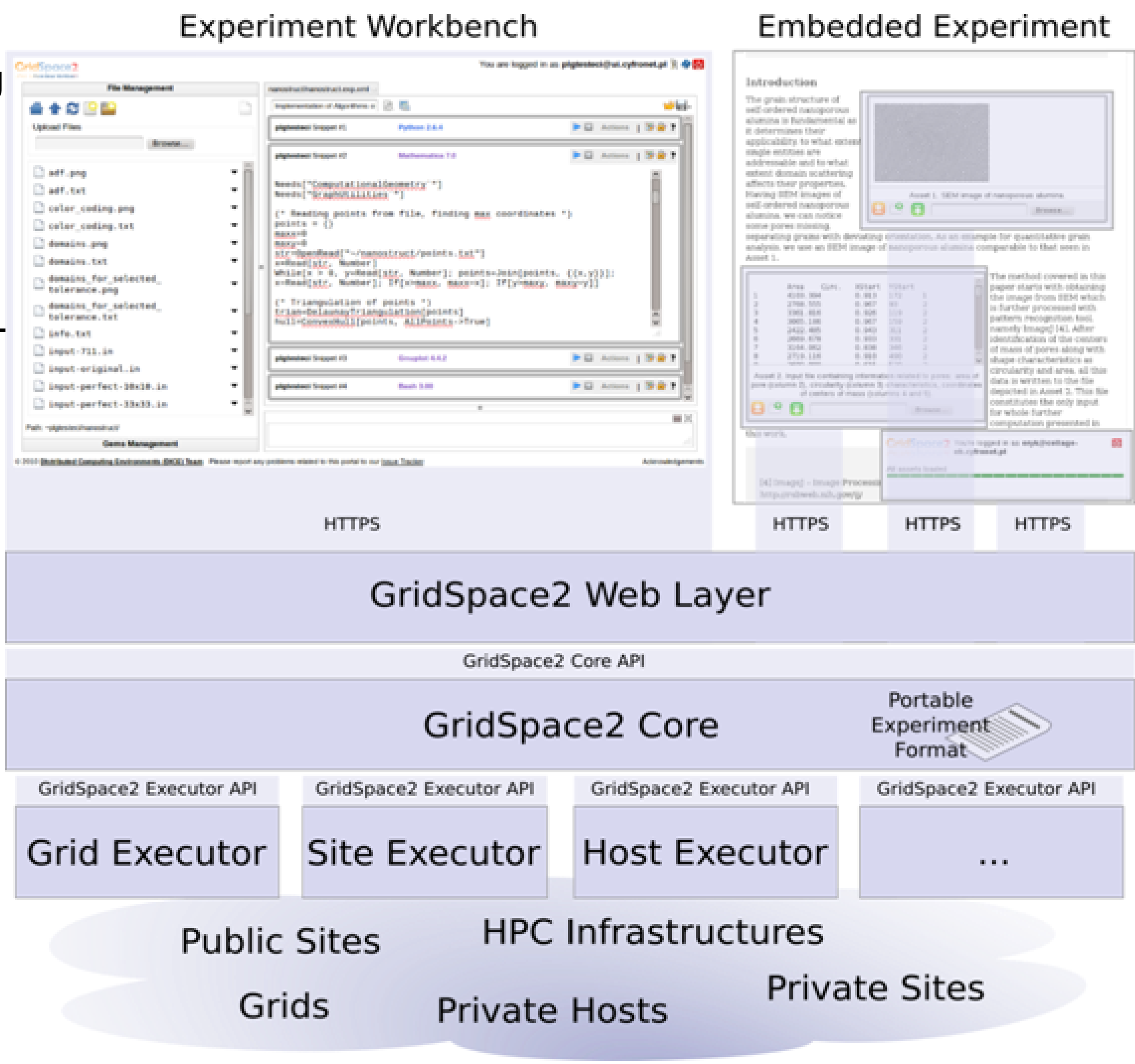
Need of Transparent Access to DCI

- Joined taskforce between MAPPER, EGI and PRACE
  - Collaborate with EGI and PRACE to introduce new capabilities and policies onto e-Infrastructures
  - Deliver new application tools, problem solving environments and services to meet end-users needs
  - Work closely with various end-users communities (involved directly in MAPPER) to perform distributed multiscale simulations and complex experiments



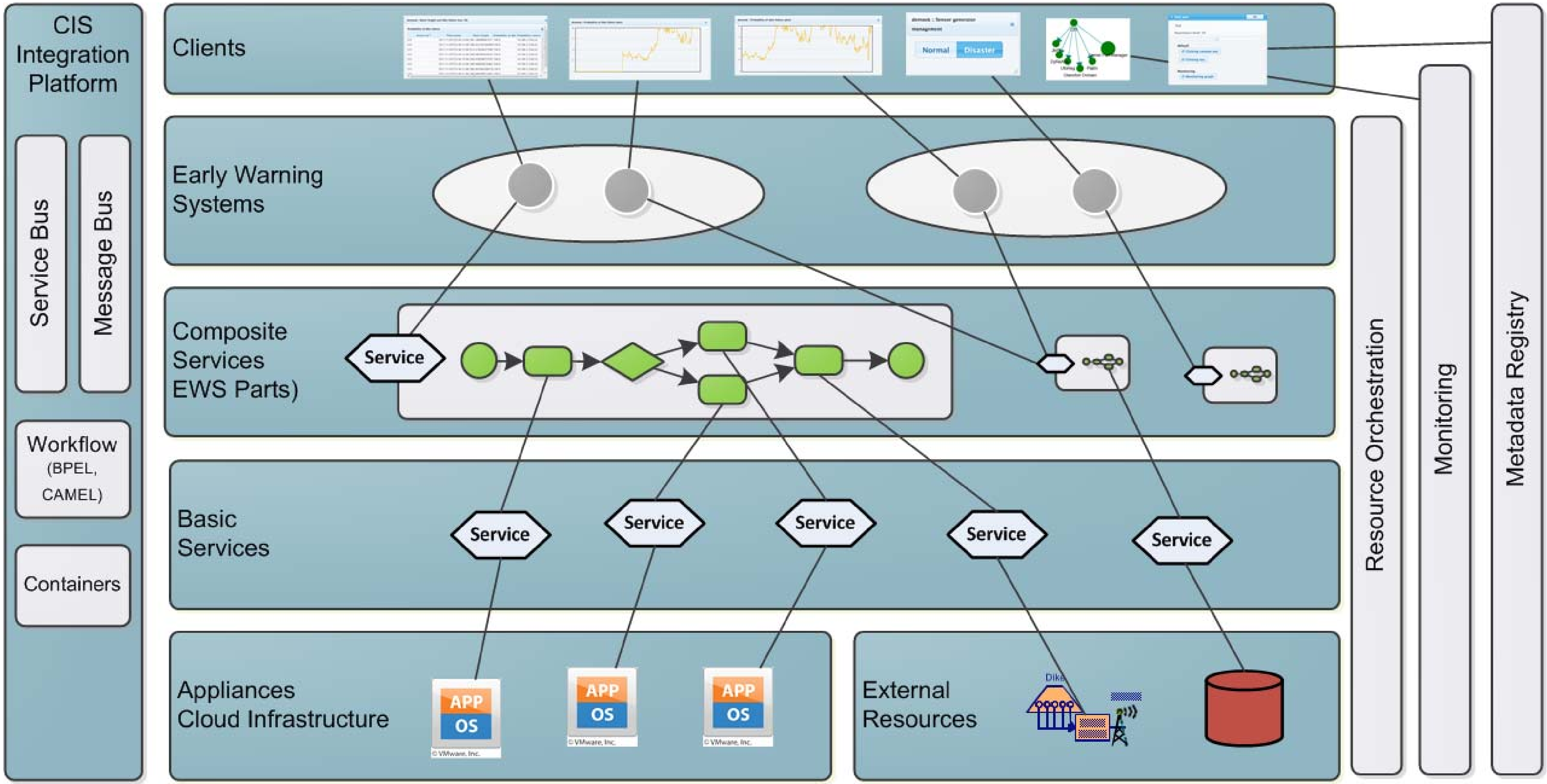
GridSpace - Platform for e-Science Applications

- **Experiment**: an e-science application composed of code fragments (snippets), expressed in either general-purpose scripting programming languages, domain-specific languages or purpose-specific notations. Each snippet is evaluated by a corresponding **interpreter**.
- **Experiment Workbench**: a web application - an entry point to GridSpace. It facilitates exploratory development, execution and management of e-science experiments.
- **Embedded Experiment**: a published experiment embedded in a web site.
- **GridSpace Core**: a Java library providing an API for development, storage, management and execution of experiments. Records all available interpreters and their installations on the underlying computational resources.
- **Computational Resources**: servers, clusters, grids, clouds and e-infrastructures where the experiments are computed.

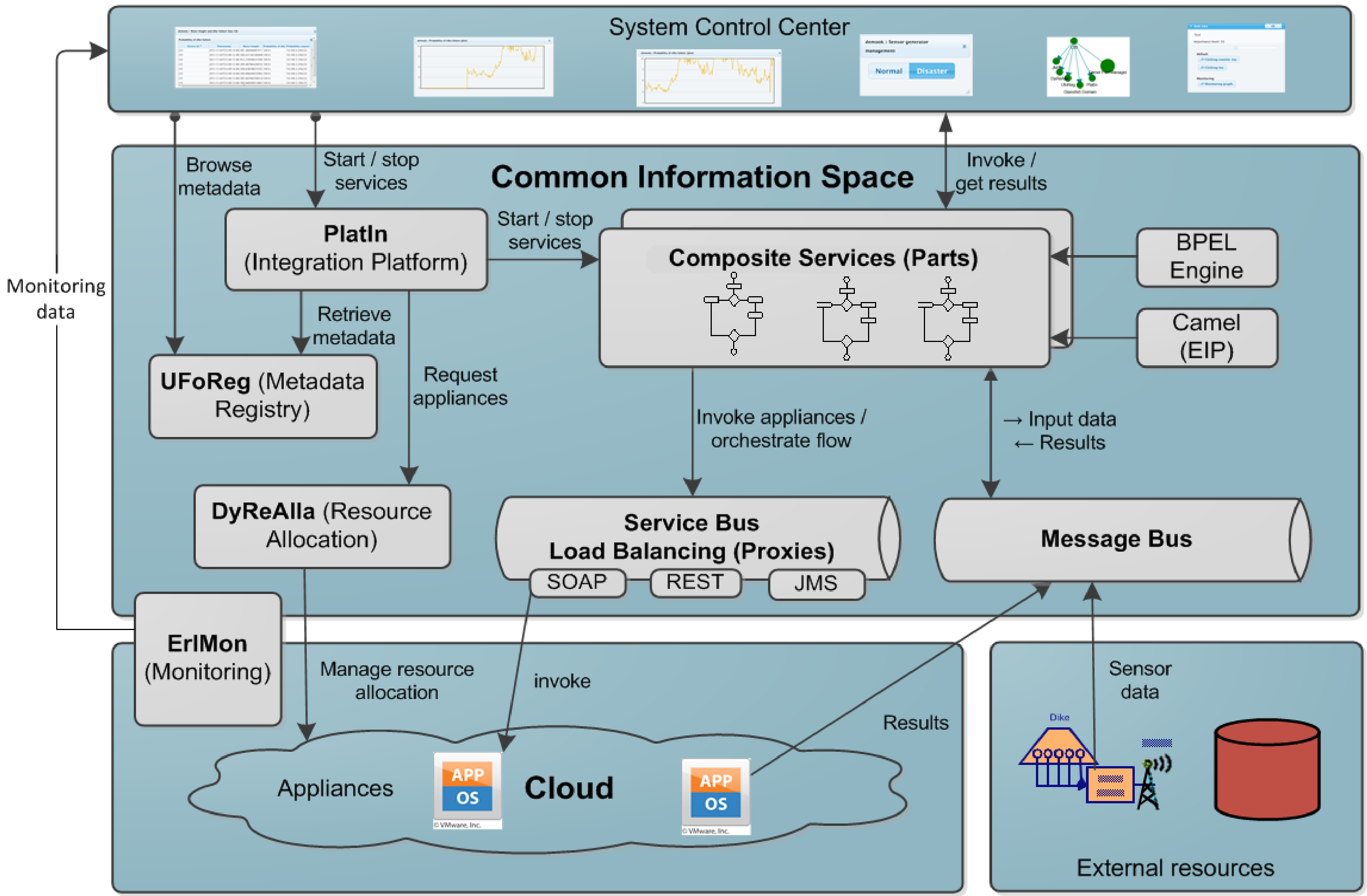


E. Ciepiela, D. Harezlak, J. Kocot, T. Bartynski, M. Kasztelnik, P. Nowakowski, T. Gubała, M. Malawski, M. Bubak: *Exploratory Programming in the Virtual Laboratory*. In: Proceedings of the International Multiconference on Computer Science and Information Technology, pp. 621-628 , October 2010, the best paper award.

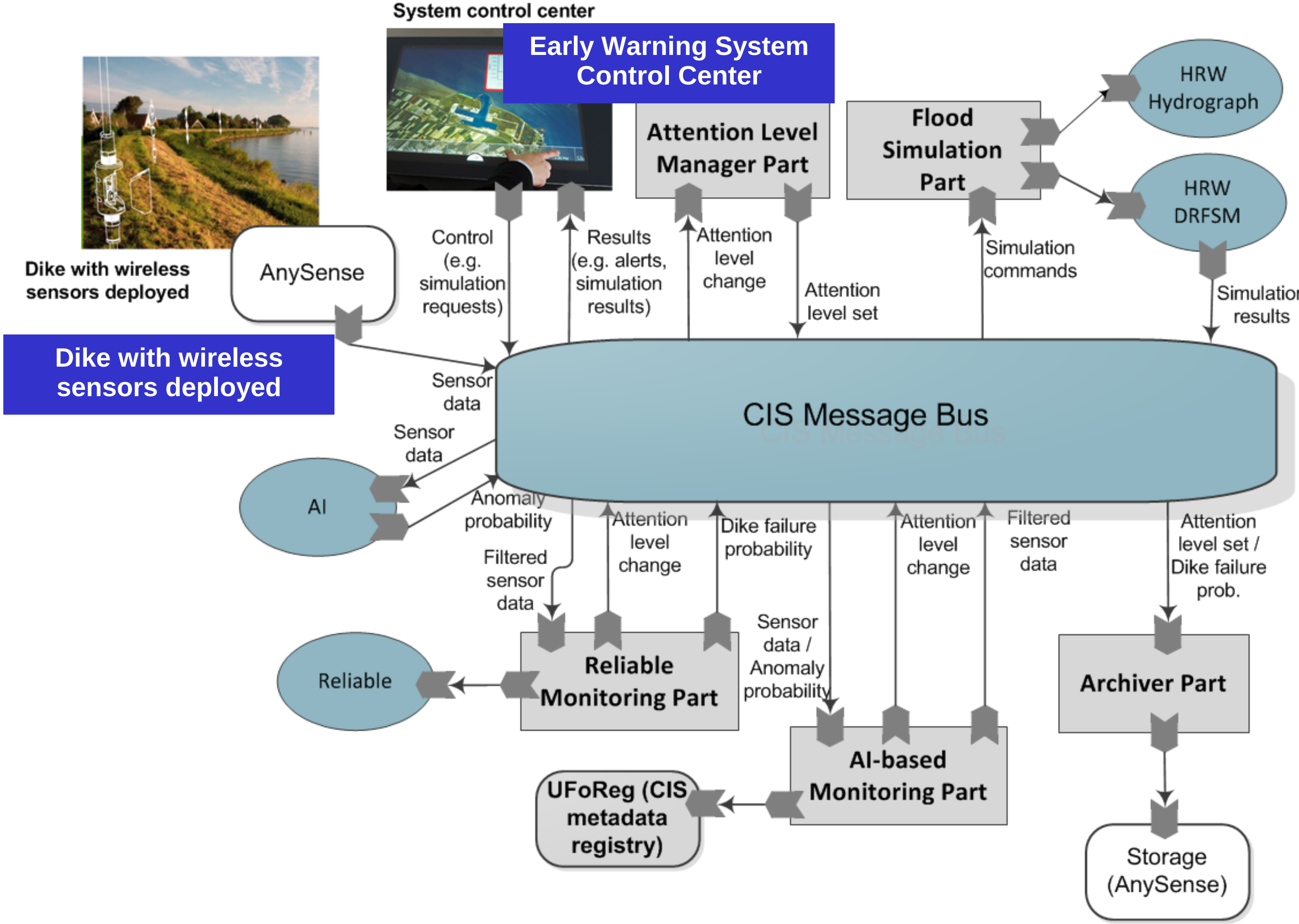
Anatomy of an Early Warning System



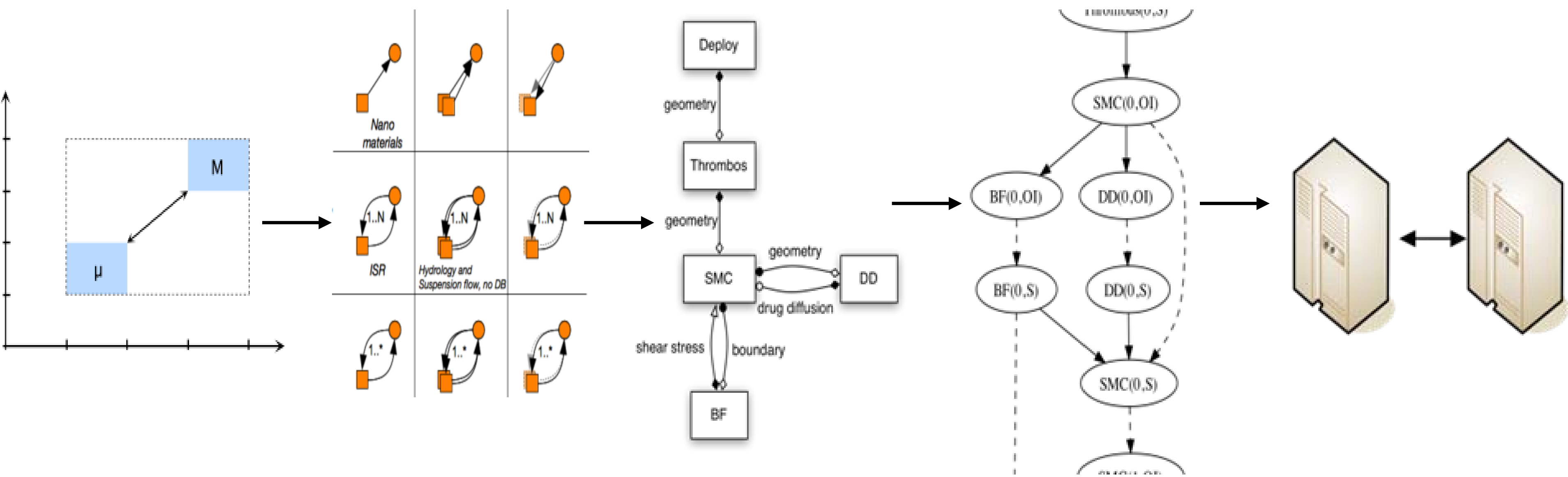
Common Information Space



Flood EWS Implemented with CIS



Generic Multiscale Computing Approach



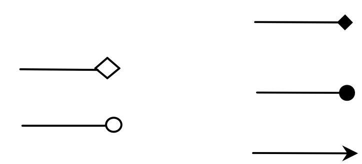
SSM | *Coupling topology* | *(x)MML* | *Task graph* | *Scheduling*

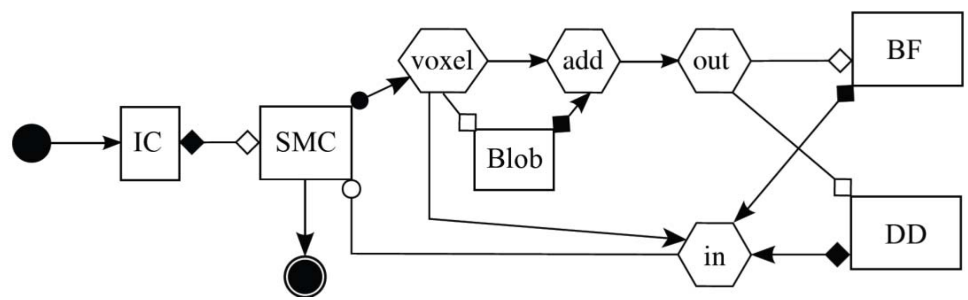
Multiscale Modeling Language

- Uniformly describes multiscale models and their computational implementation on abstract level
- Two representations: graphical (gMML), textual (xMML)
- Includes description of
  - scale submodules
  - scaleless submodules (so called mappers and filters)
  - ports and their operators (for indicating type of connections between modules)
  - coupling topology
  - implementation

Submodel execution loop in pseudocode

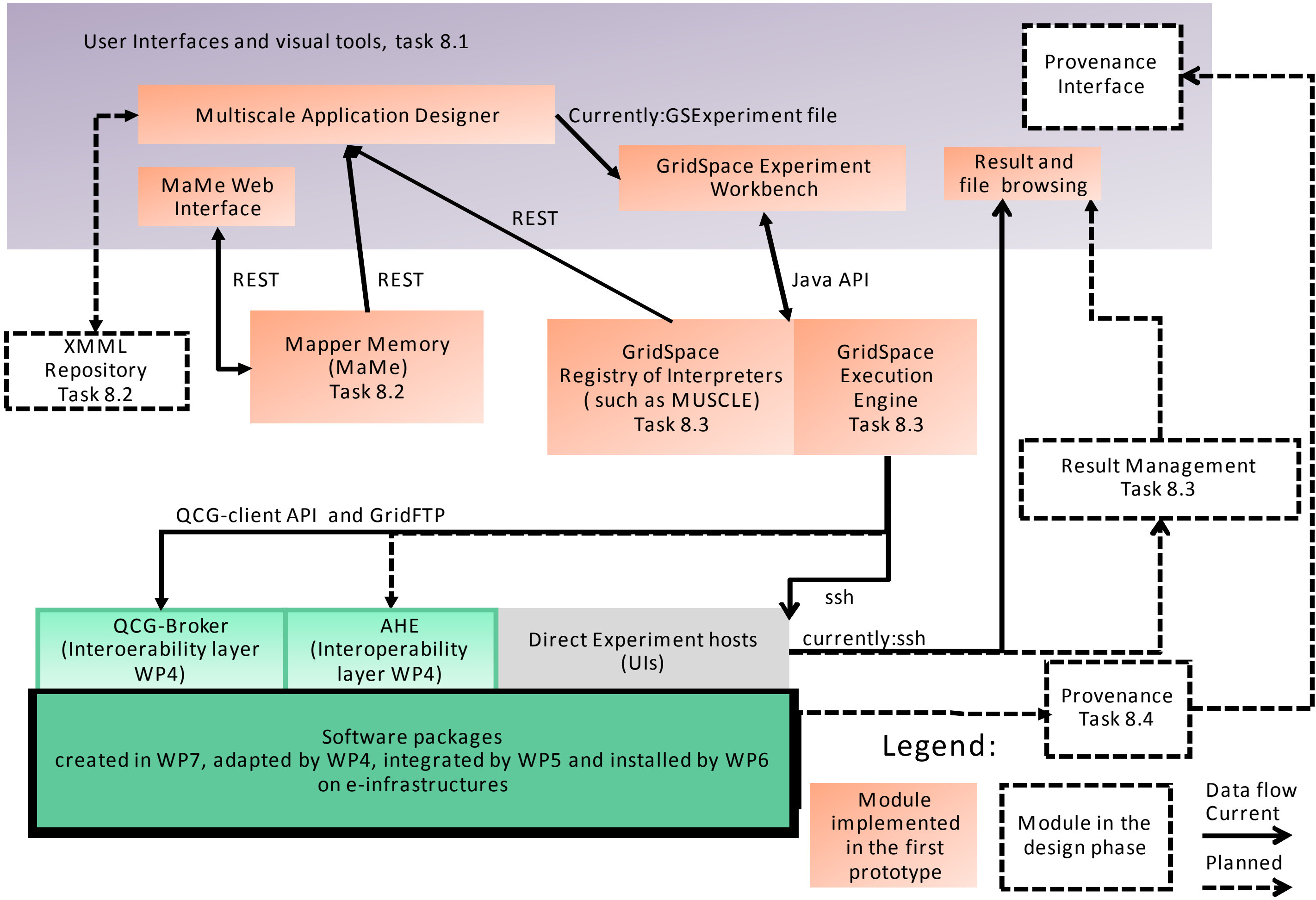
```
f := finit /*initialization*/
 t := 0
while not EC(f, t):
        Oi(f, t) /*intermediate observation*/
        f := S(f, t) /*solving step*/
        t += theta(f)
end
Of(f, t) /*final observation*/
```





Tools for Multiscale Applications

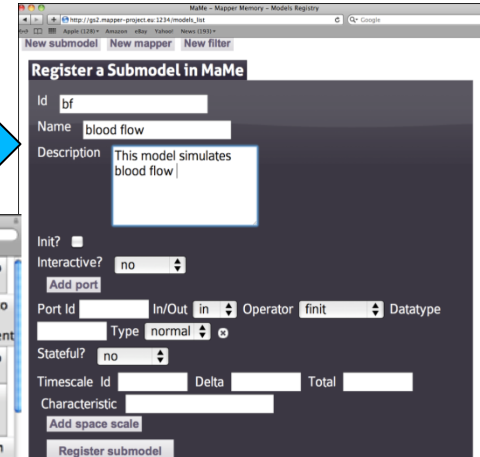
- **MAPPER Memory (MaMe)** a semantics-aware persistence store to record metadata about models and scales
- **Multiscale Application Designer (MAD)** visual composition tool transforming high level MML description into executable experiment
- **GridSpace Experiment Workbench (EW)** execution and result management of experiments on e-infrastructures via interoperability layers (AHE, QCG)



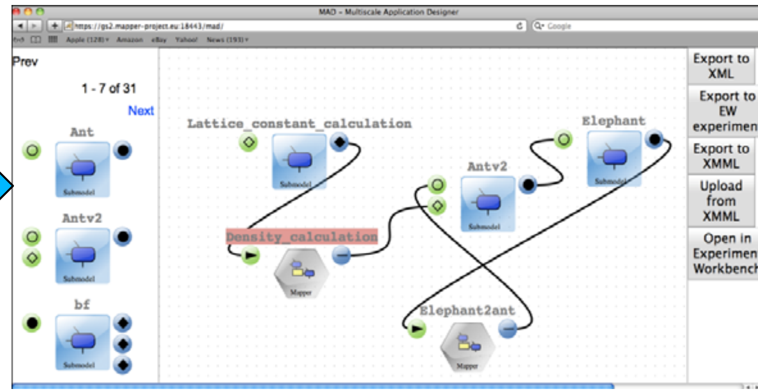
Katarzyna Rycerz and Marian Bubak: Building and Running Collaborative Distributed Multiscale Applications, in: W. Dubitzky, K. Kurowsky, B. Schott (Eds), Chapter 6, Large Scale Computing, J. Wiley and Sons, 2012

Multiscale Application Environment

Registration of MML metadata: submodules and scales

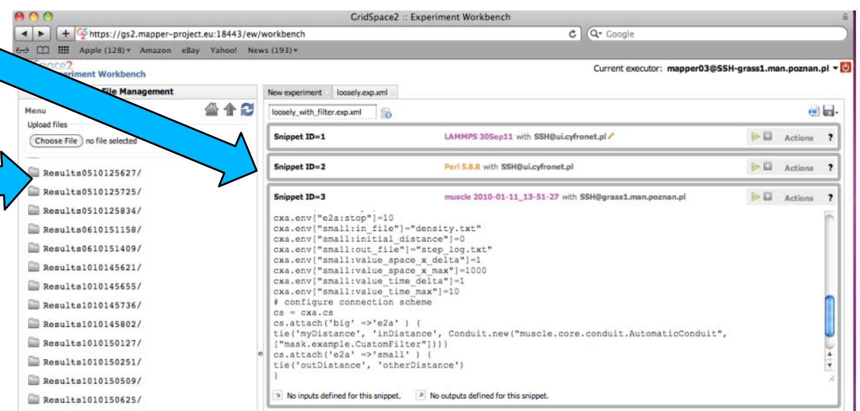


Application composition: from MML to executable experiment



Execution of experiment with an interoperability layer

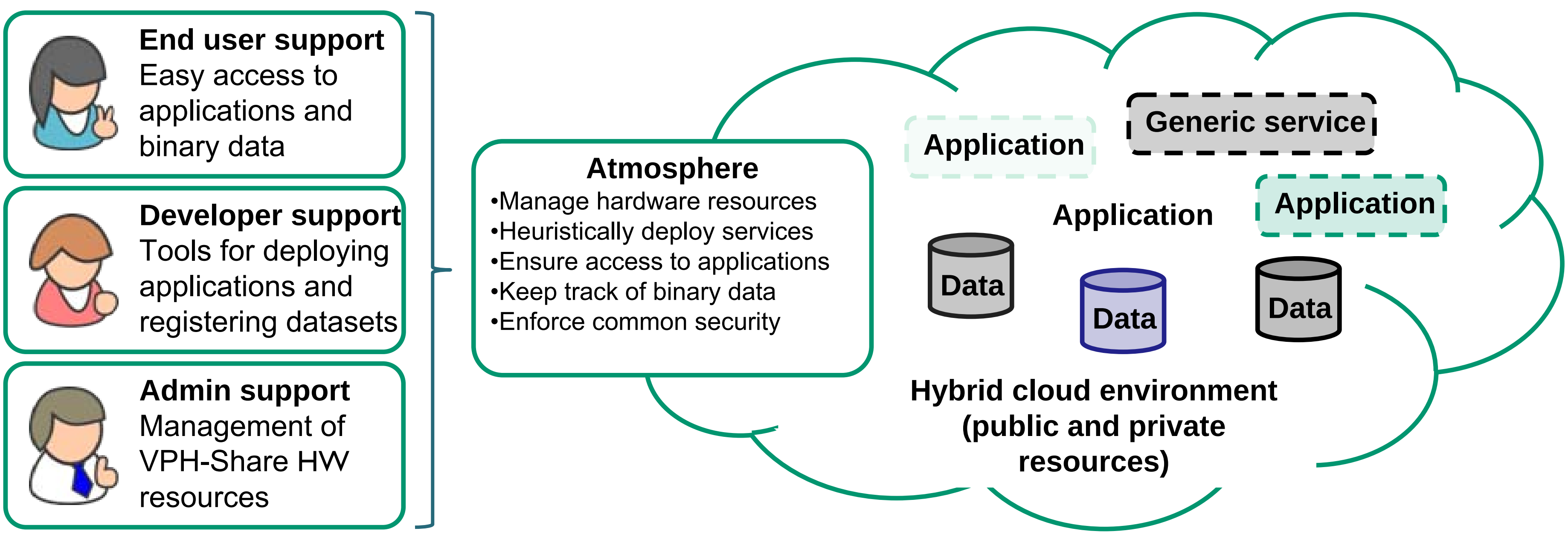
Result Management



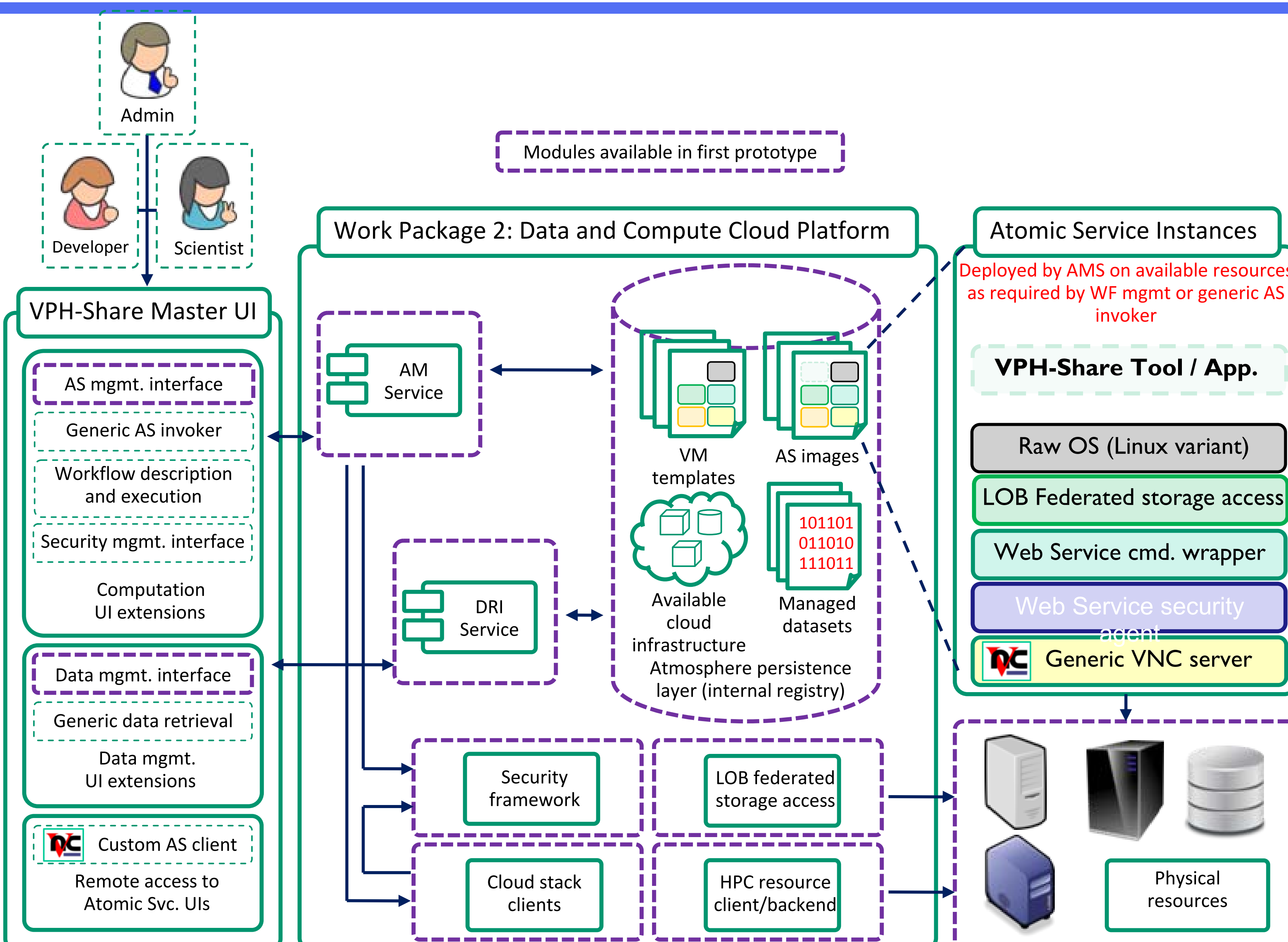
Federating Cloud Resources in the VPH-Share

The goal of the **VPH-Share Cloud Computing Platform** is to manage cloud/HPC resources in support of VPH-Share applications by:

- Providing a mechanism for **application developers** to install their applications/tools/services on the available resources;
- Providing a mechanism for **end users** (domain scientists) to execute workflows and/or standalone applications on the available resources with minimum fuss;
- Providing a mechanism for **end users** (domain scientists) to securely manage their binary data in a hybrid cloud environment;
- Providing **administrative tools** facilitating configuration and monitoring of the platform;

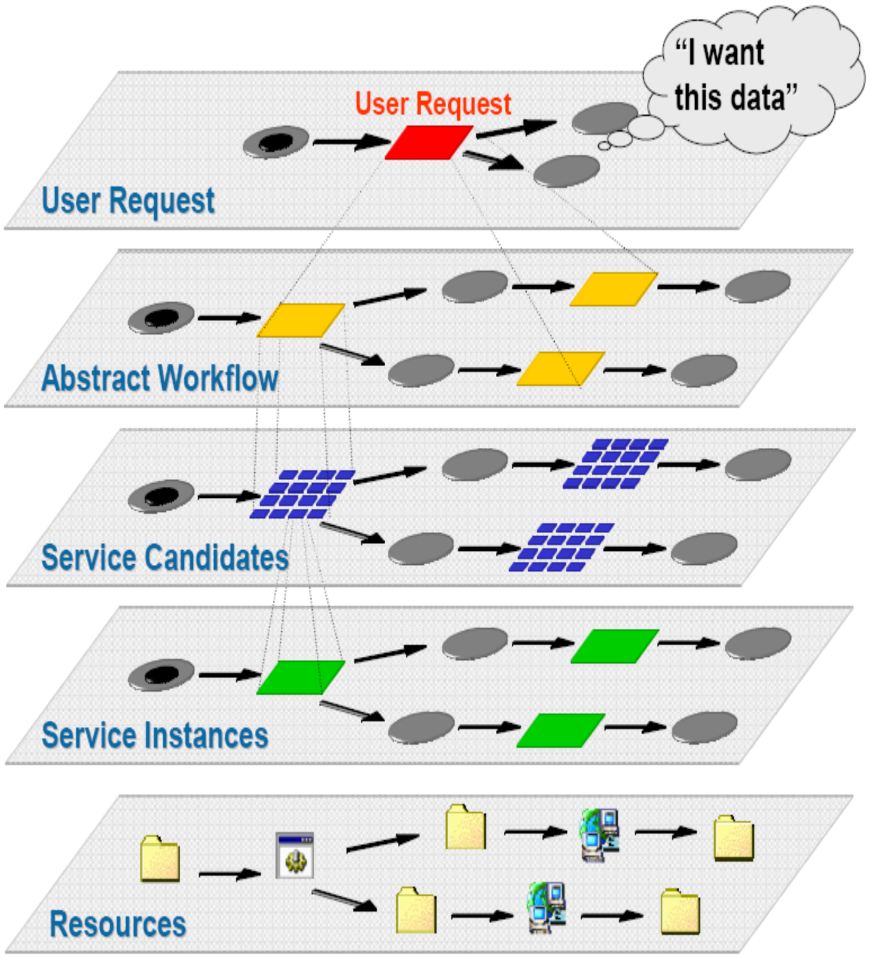


VPH-Share Cloud Platform Architecture



Semantic Workflow Composition

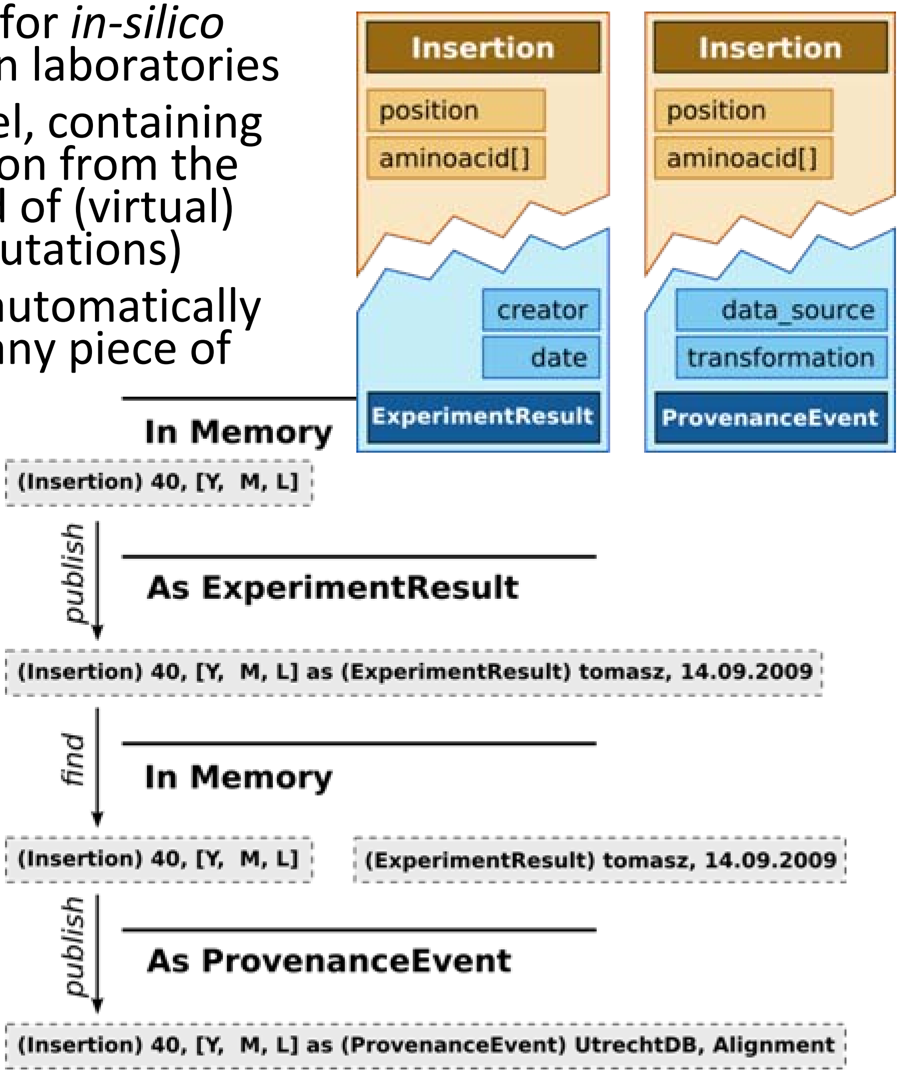
- GworkflowDL language (with A. Hoheisel)
- Dynamic, ad-hoc refinement of workflows based on semantic description in ontologies
- Novelty
  - Abstract, functional blocks translated automatically into computation unit candidates (services)
  - Expansion of a single block into a subworkflow with proper concurrency and parallelism constructs (based on Petri Nets)
  - Runtime refinement: unknown or failed branches are re-constructed with different computation unit candidates



T. Gubala, D. Harezlak, M. Bubak, M. Malawski, "Semantic Composition of Scientific Workflows Based on the Petri Nets Formalism", in "The 2nd IEEE International Conference on e-Science and Grid Computing", IEEE Computer Society Press, http://doi.ieeecomputersociety.org/10.1109/E-SCIENCE.2006.127, 2006

Semantic Integration for Science Domains

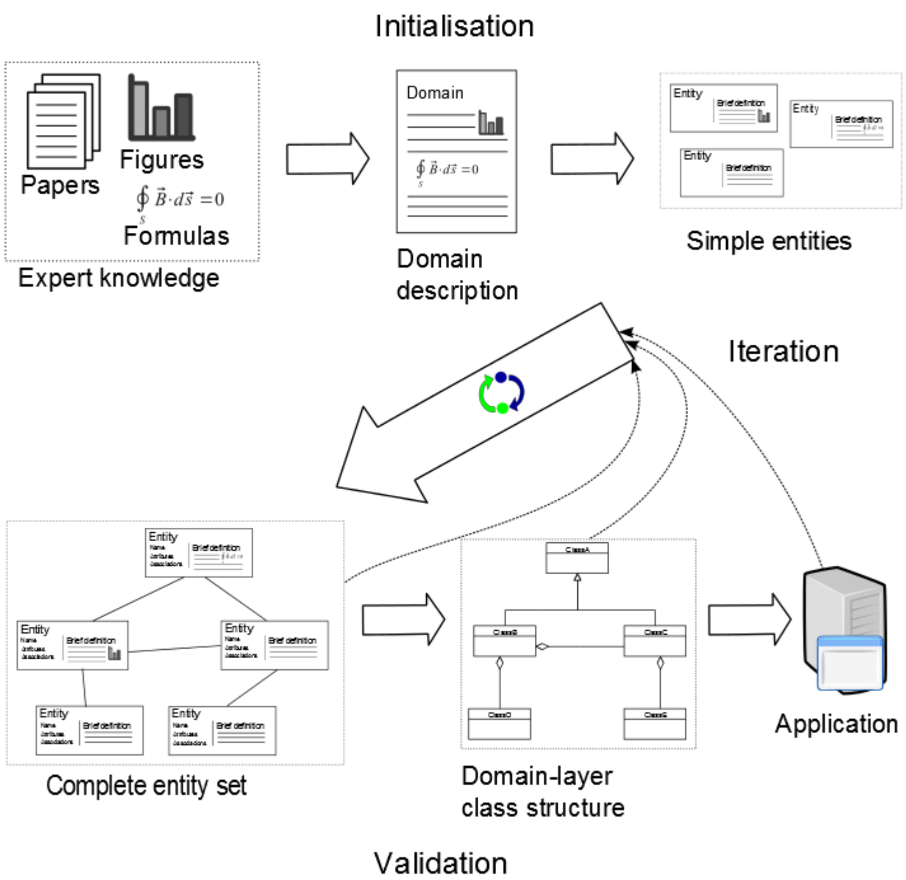
- Concept of describing scientific domains for *in-silico* experimentation and collaboration within laboratories
- Based on separation of the *domain* model, containing concepts of the subject of experimentation from the *integration* model, regarding the method of (virtual) experimentation (tools, processes, computations)
- Facets defined in integration model are automatically mixed-in concepts from domain model: any piece of data may show any desired behavior
- Proposed, designed and deployed the method for 3 domains of science:
  - Computational chemistry inside InSilicoLab chemistry portal
  - Sensor processing for early warning and crisis simulation in UrbanFlood EWS
  - Processing of results of massive bioinformatic computations for protein folding method comparison
  - Composition and execution of multiscale simulations
  - Setup and management of VPH applications



T. Gubala, M. Bubak, P.M.A. Sloot, "Semantic Integration of Collaborative Research Environments", chapter XXVI in "Handbook of Research on Computational Grid Technologies for Life Sciences, Biomedicine and Healthcare", Information Science Reference IGI Global 2009, ISBN: 978-1-60566-374-6, pages 514-530

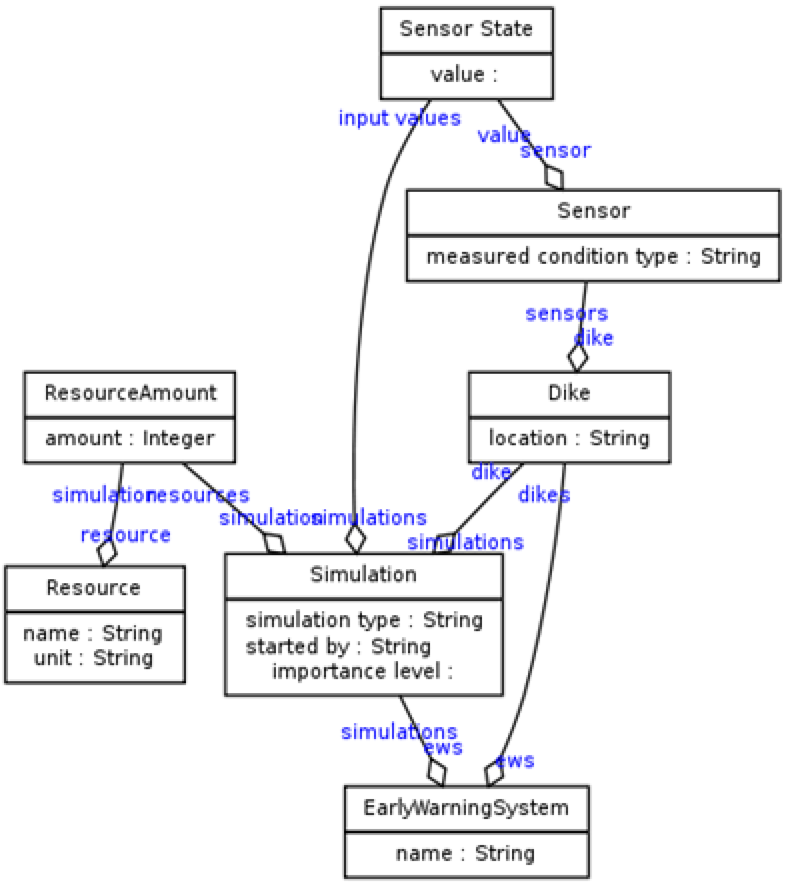
Composition of Semantic Domain Models

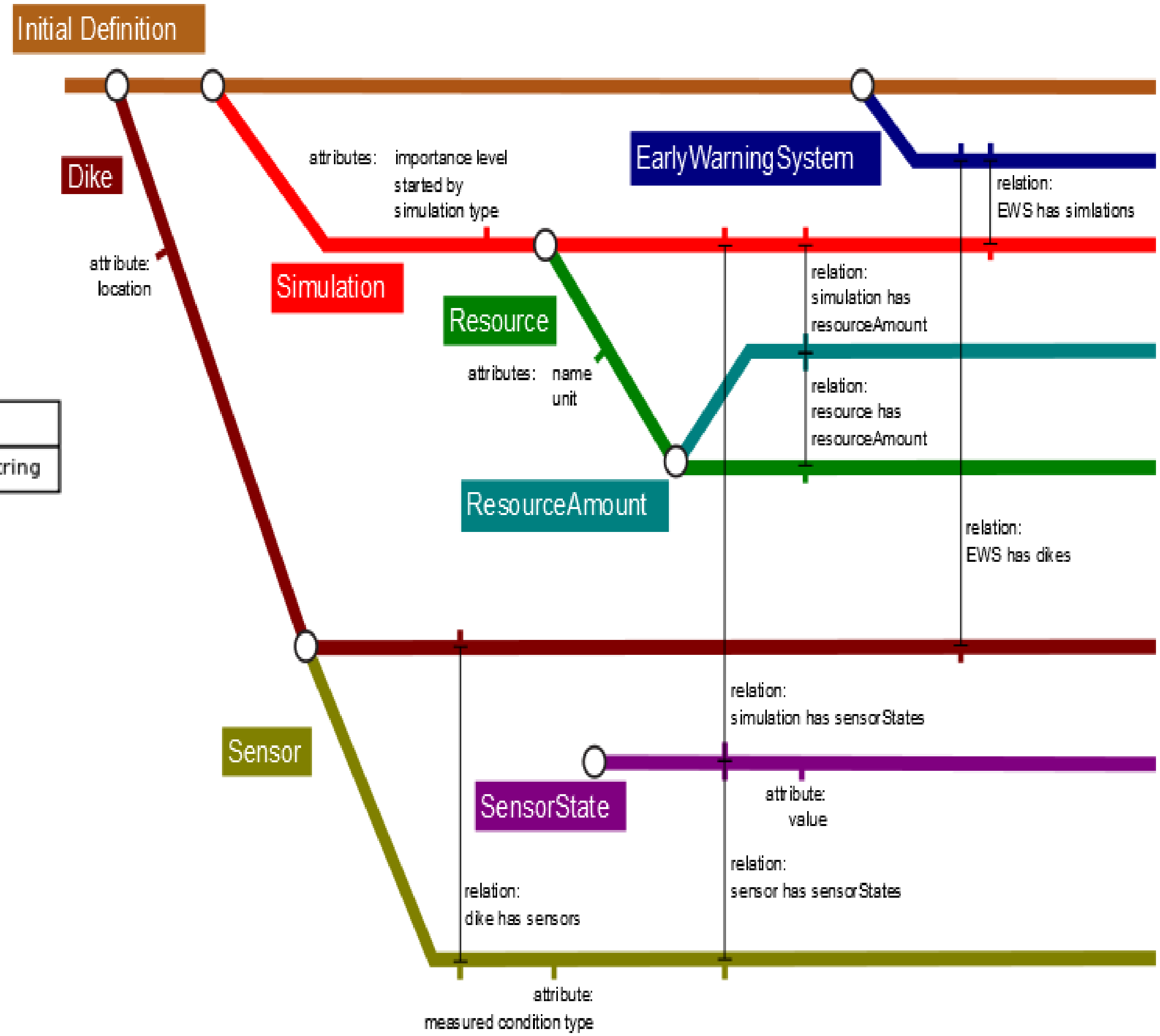
- **Objective:** To facilitate knowledge transmission between domain experts and developers.
- **Solution:**
  - iterative methodology of cooperation
  - tool that supports the methodology



Modeling of EWS

- Modelling with an expert from Urban Flood project
  - domain model
  - Flow of cooperation



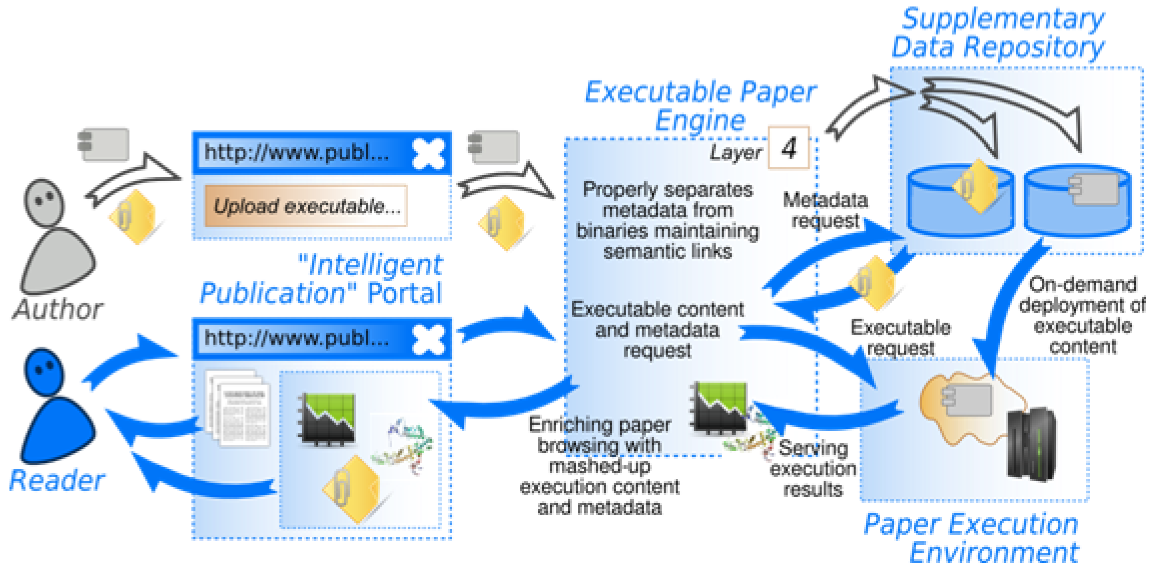


M. Rzasa, T. Gubala, M. Bubak: Methodology and Tool Supporting Cooperative Com[position of Semantic Domain Models for Experts and Developers, Software – Practice and Experience, to be submitted

Collage - Executable e-Science Publications

Goal:

Extending the traditional scientific publishing model with computational access and interactivity mechanisms; enabling readers (including reviewers) to replicate and verify experimentation results and browse large-scale result spaces.



Challenges:

Scientific: A common description schema for primary data (experimental data, algorithms, software, workflows, scripts) as part of publications; deployment mechanisms for on-demand reenactment of experiments in e-Science.

Technological: An integrated architecture for storing, annotating, publishing, referencing and reusing primary data sources.

Organizational: Provisioning of executable paper services to a large community of users representing various branches of computational science; fostering further uptake through involvement of major players in the field of scientific publishing.

P. Nowakowski, E. Ciepiela, D. Harężlak, J. Kocot, M. Kasztelnik, T. Bartyński, J. Meizner, G. Dyk, M. Malawski: *The Collage Authoring Environment*. In: Proceedings of the International Conference on Computational Science, ICCS 2011 (2011),
Winner of the Elseview/ICCS Executable Paper Grand Challenge

Summary

- An overview of different approaches to building collaborative e-Science applications (system-level applications) on distributed e-infrastructures
- Complexity of applications and e-infrastructures
- Need for support of rapid prototyping
- Software engineering methods for scientific applications
- Commercial solutions may be adequate also for scientific applications
- Role of social aspects; developer and domain expert

More at …

http://dice.cyfronet.pl

http://www.science.uva.nl/~gvlam/wsvlam

EU: UrbanFlood, MAPPER, VPH-Share

PL: PL-Grid, PL-Grid++

NL: COMMIT