

Workflows in Computational Chemistry

Prof. Gabor Terstyanszky
 Centre for Parallel Computing
 University of Westminster
 School of Open Science Cloud
 Perugia, Italy
 05 – 09 June 2017



Challenges and Motivation

Challenges

- Learn how to create and use workflows
- Many different workflow systems exist that are not interoperable
- Technological choice (data and computer resources) isolates users and user communities

Benefits of workflow

- Share your own workflow, re-use workflows of others
- Create and execute 'meta-workflows': built from smaller workflows that use different workflow languages/technologies
- Combine workflows and DCIs



Key Players and Challenges

Domain researchers

Researchers of one particular research field, for example Astrophysicists, Computational Chemists, Heliophysicists, Bio Scientists, etc. with basic computing knowledge

10 or 100 thousands or even millions

Challenges: They are not familiar with the technology to run experiments on computing infrastructures and probably they will never learn it.

Workflow developers

They are familiar with both Computer Science and a particular research field

up to a few thousands

Workflow system developers

Computer Scientists with knowledge about data and compute technologies

up to a few hundreds



Workflow Developer's Scenario

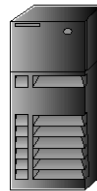
They want

- to develop & publish WFs

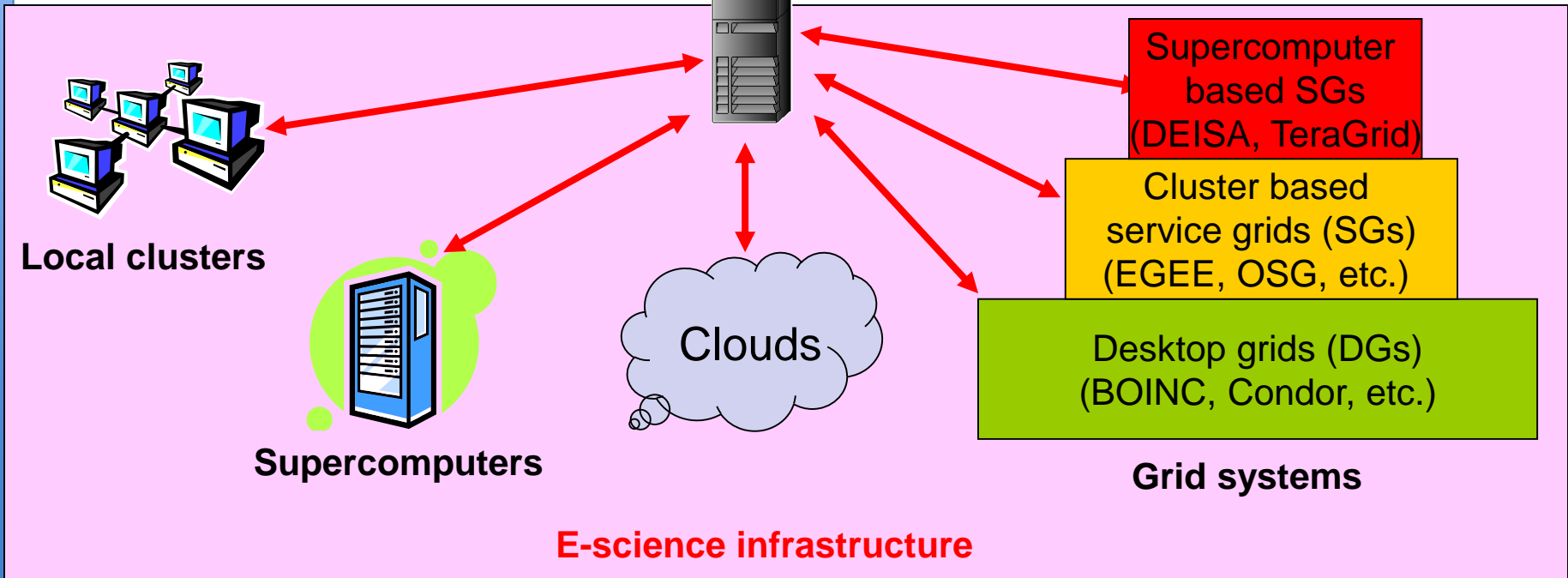
They need

- to develop WFs somewhere
- to publish WFs somewhere
- to manage WFs somewhere
- to execute WFs somewhere

Workflow Repository



Science Gateway





Domain Researcher's Scenario

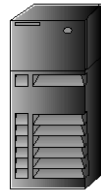
They want

- to run experiments

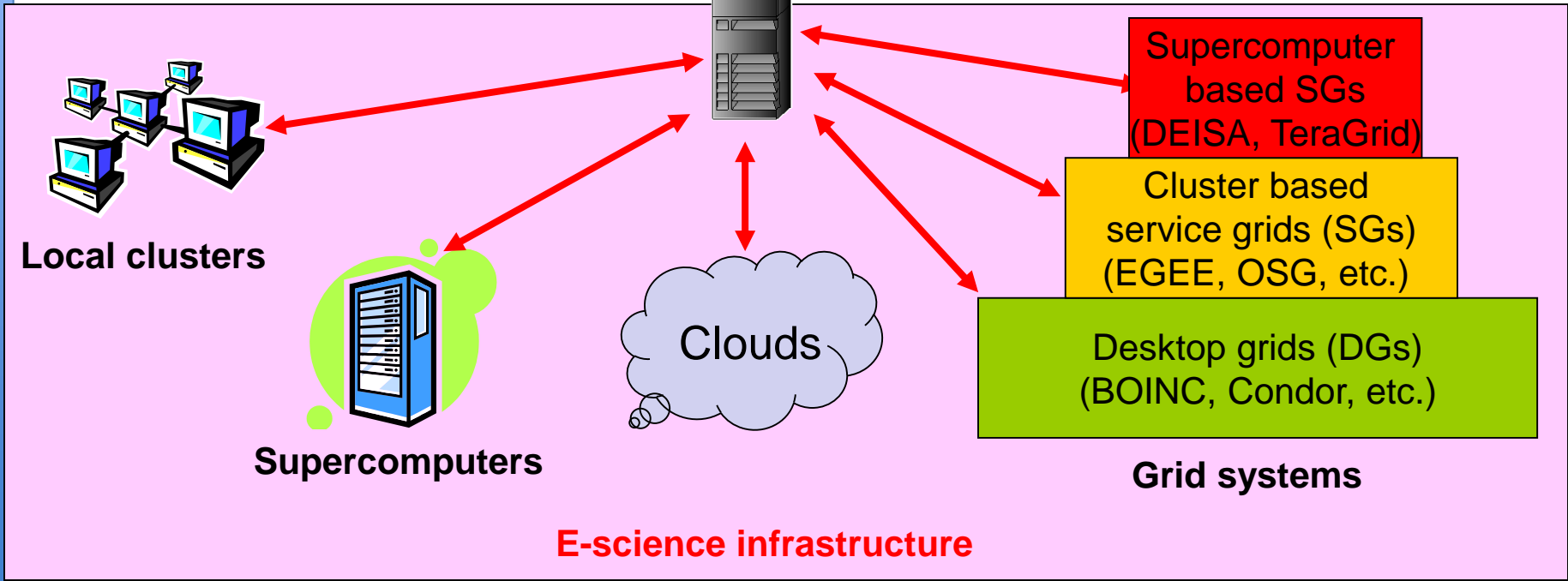
They need

- to run experiments seamlessly
i.e. executing workflows which
access to data and compute
resources hiding all technical
details

**Workflow
Repository**



**Science
Gateway**



E-science infrastructure



Co-operation with Research Communities

Phase 1 – introduction to the workflow technology

Target group: communities without any or basic experience in the workflow technology

Phase 2 – creating and running workflows

Target group: communities those use workflows to run experiments

Phase 3 – combining workflows of different workflow systems

Target group: communities those use workflows to run experiments and are interested in using workflows of other workflow systems

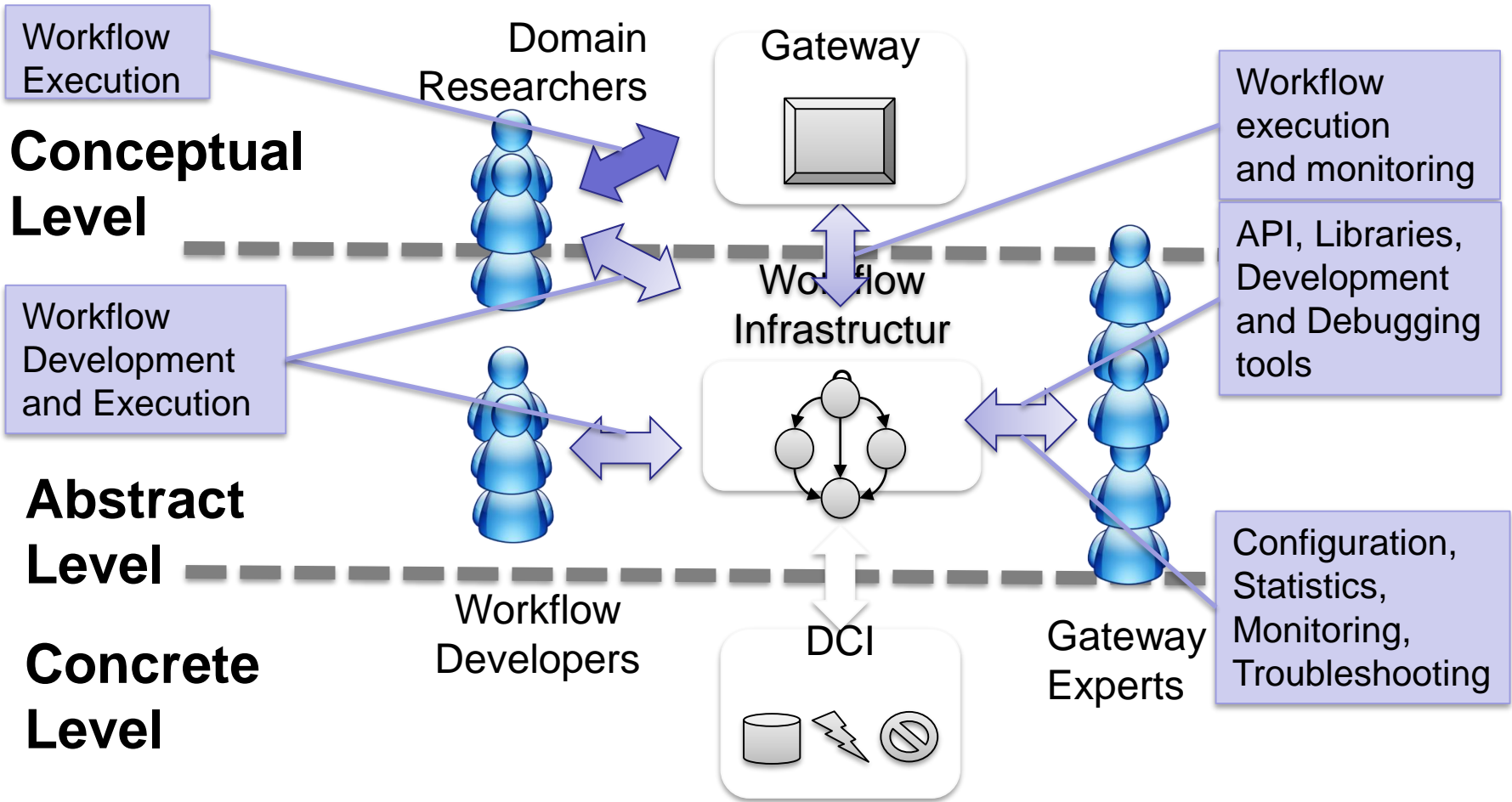
6

6

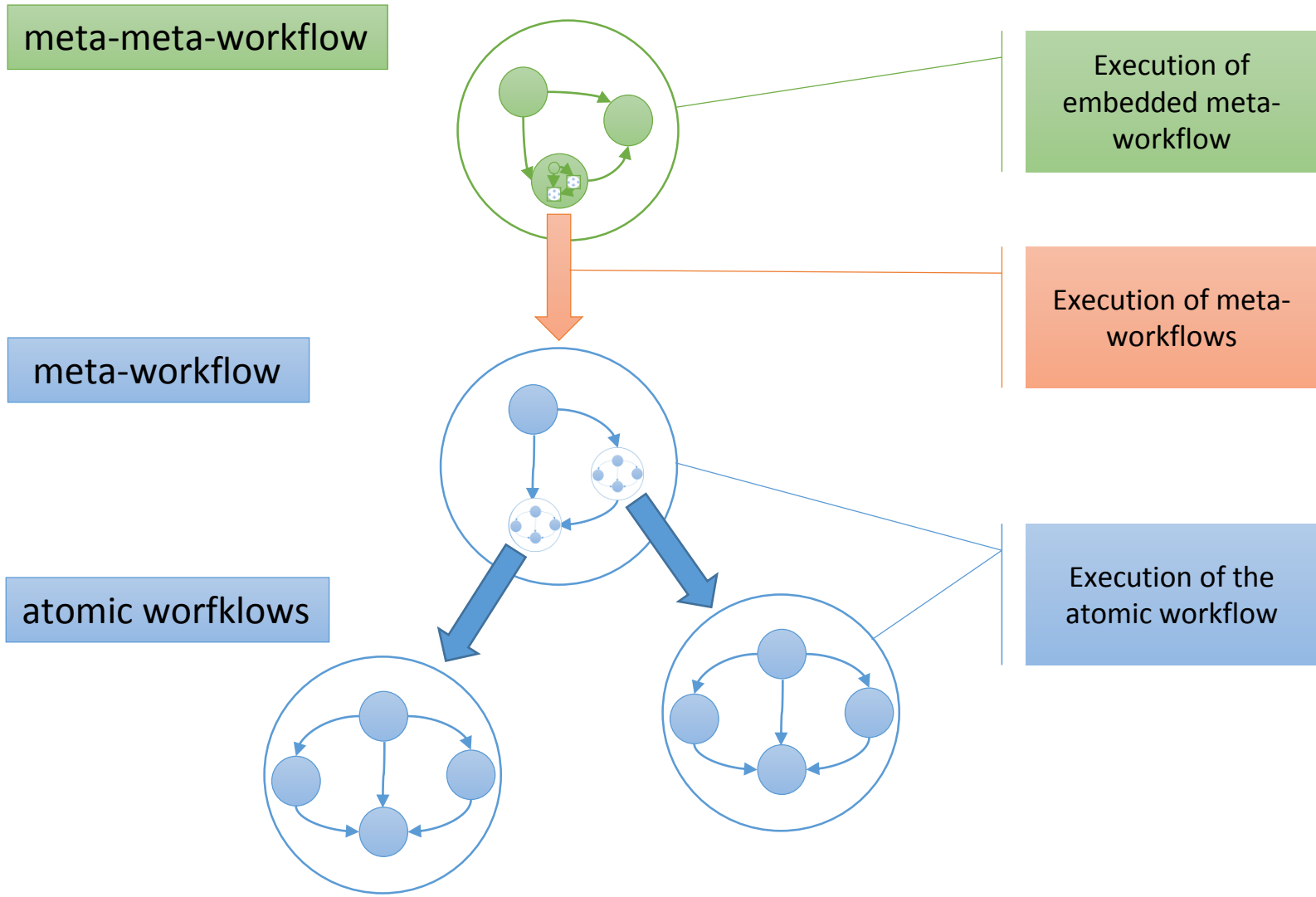




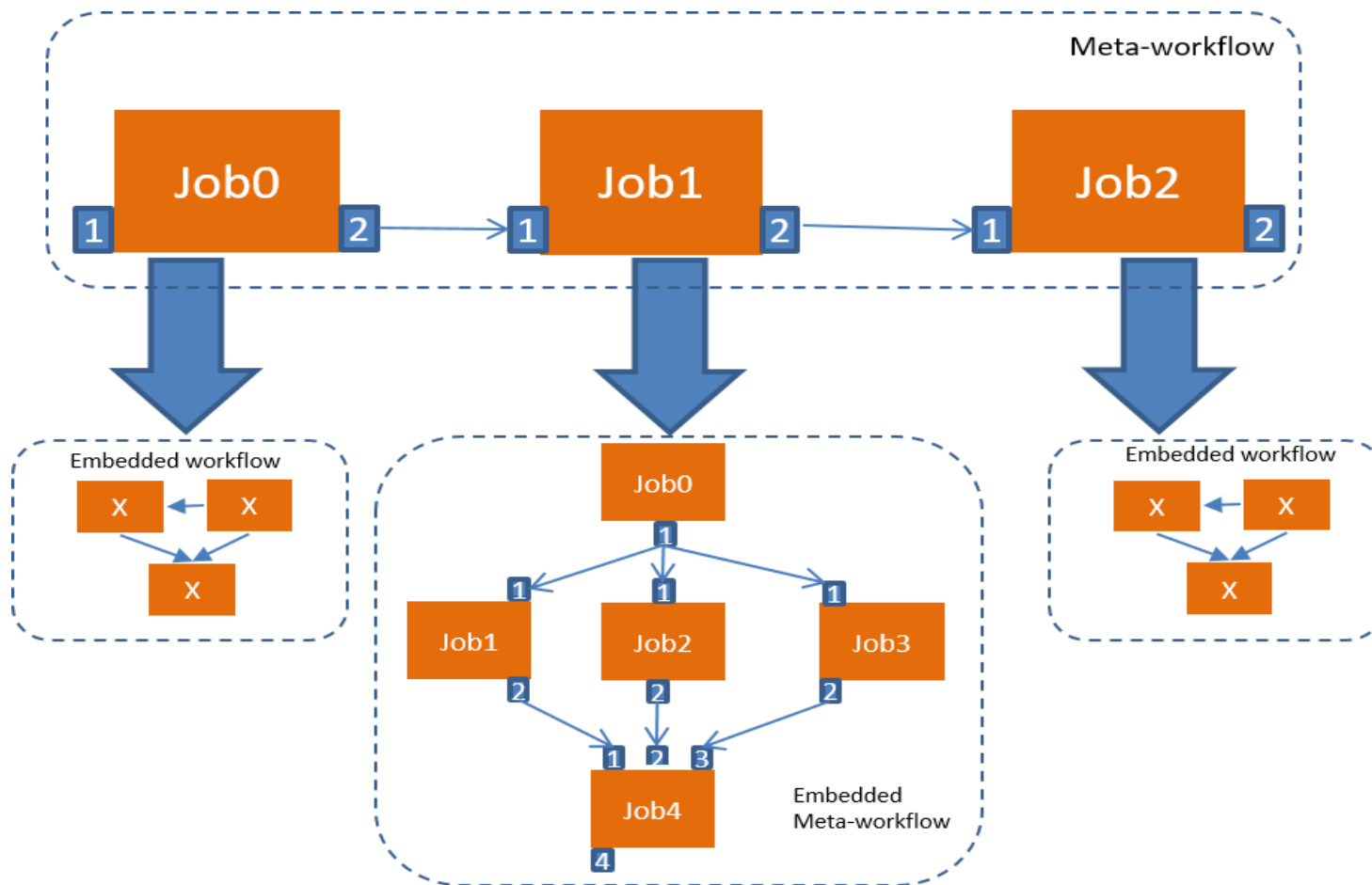
Workflow Levels and Users



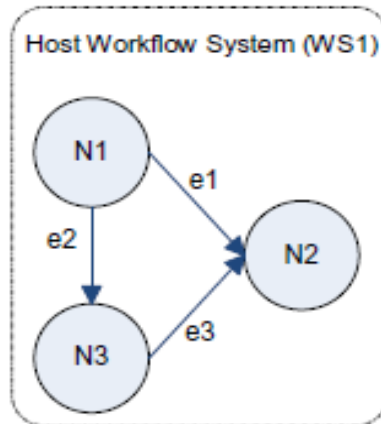
Atomic Workflow Concept



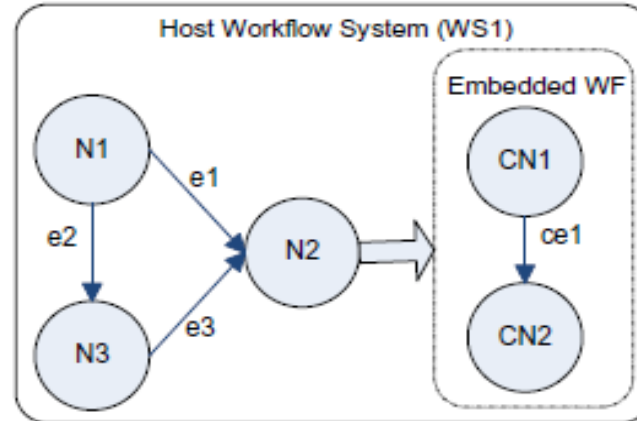
Computational Chemistry: Meta-Workflow Concept (1)



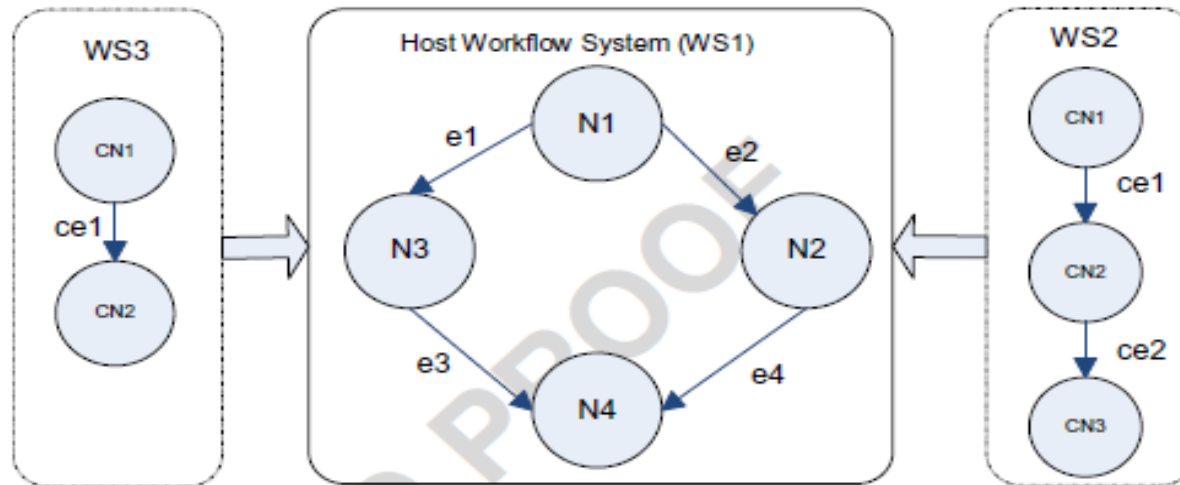
Computational Chemistry: Meta-Workflow Concept (2)



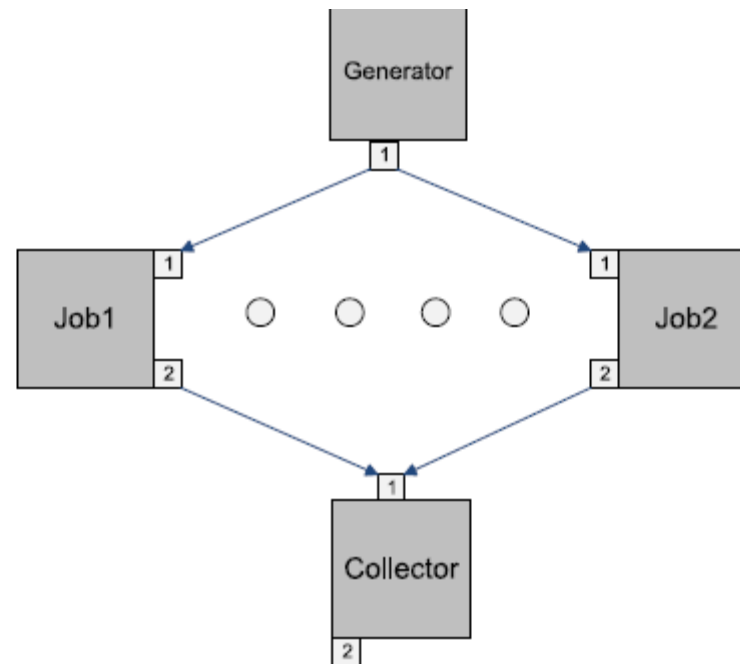
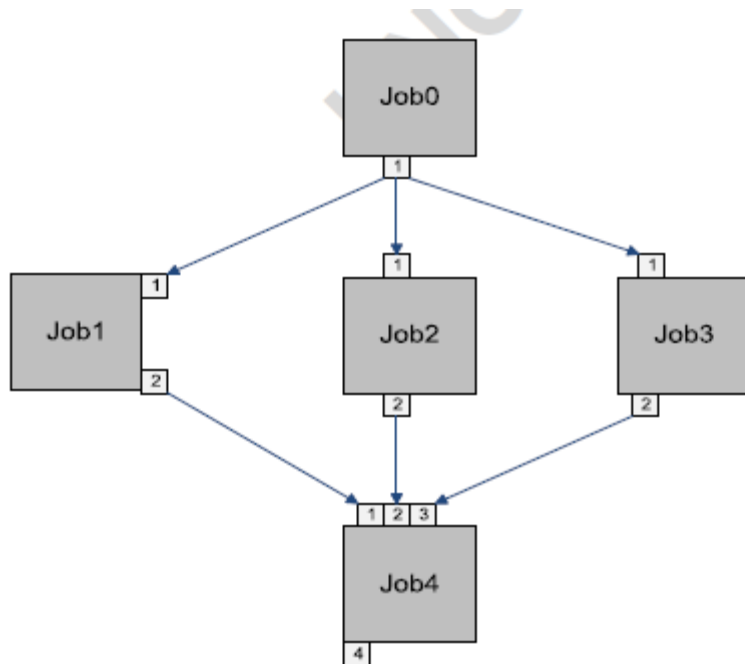
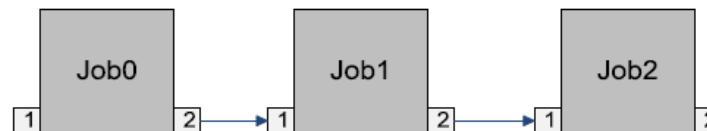
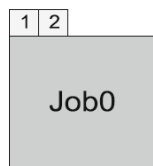
a



b



Computational Chemistry: Meta-Workflow Concept (3)





Workflows versus Scientific Experiments

| Type of Workflow | Technology | Driven by | Usage |
|--------------------------|--|------------------------|--|
| Atomic Workflow | Basic workflows | Use Case | Execution of Atomic Workflows |
| Meta-workflows | Composite workflows of atomic workflows | Science Case | Orchestration of Atomic Workflows |
| Iterative meta-workflows | Parameter study of workflows or meta-workflows | Iterative Science Case | Parameter-Sweep Execution of Science Cases |

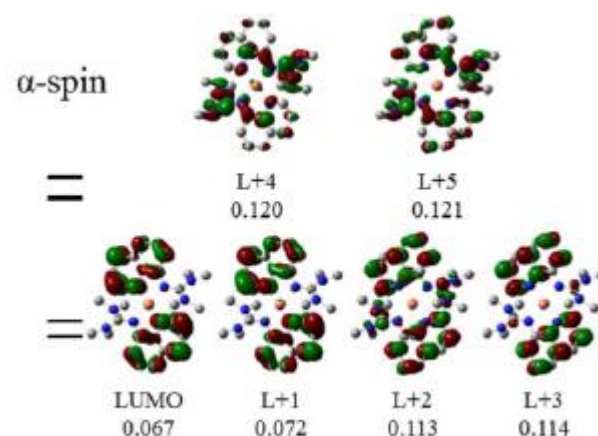
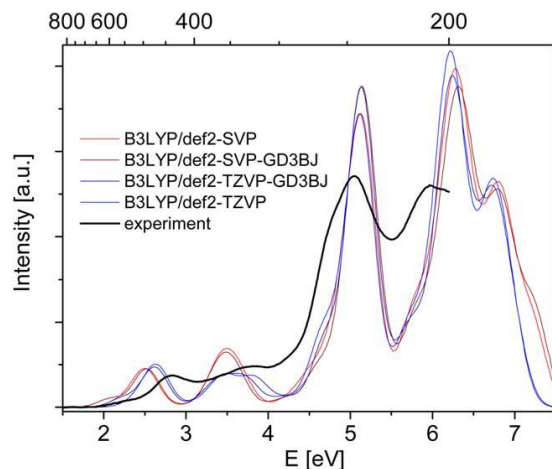
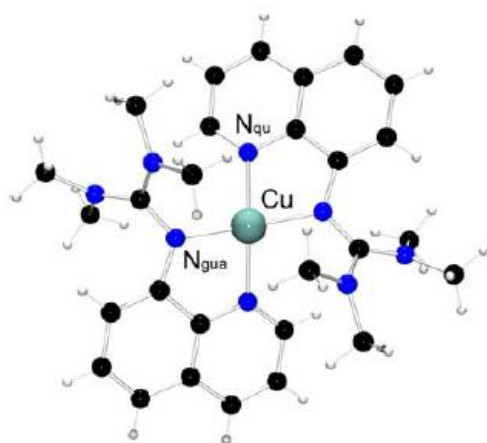


Quantum Chemistry Workflows

| Name | Description | Engine | Middleware | Type | ID |
|-----------------------------------|---|-----------|------------|--------------------|------|
| Spectroscopic analysis | Explore the spectroscopic characteristics of a molecule | WS-PGRADE | UNICORE | meta-workflow | 5739 |
| Spectroscopic benchmarking | Explore the spectroscopic characteristics of a molecule with more functionals/basis sets | WS-PGRADE | UNICORE | meta-meta-workflow | 5745 |
| Parameter sweep | Benchmark a molecule using larger arrays of functionals and basis sets | WS-PGRADE | UNICORE | meta-workflow | -- |
| Transition state analysis | Find a reaction transition state and analyse its frequencies together with the spectroscopic properties | WS-PGRADE | UNICORE | meta-workflow | 5751 |
| TD-UNI | Calculate the optical response of a molecule | UNICORE | UNICORE | Workflow | |
| SpecWSUNI | Explore the spectroscopic characteristics of a molecule | UNICORE | UNICORE | Workflow | |
| PopulationUNI | Apply various population schemes to better electronic understanding of the molecules. | UNICORE | UNICORE | Workflow | |
| Galaxy QM | Optimise a molecule quantum chemically in Galaxy | Galaxy | UNICORE | Workflow | 5754 |
| QM-MD | Optimise the protein scaffold by molecular dynamics and optimise then the metallo-active center by quantum mechanics together with a spectroscopic analysis | WS-PGRADE | UNICORE | meta-meta-workflow | 5752 |

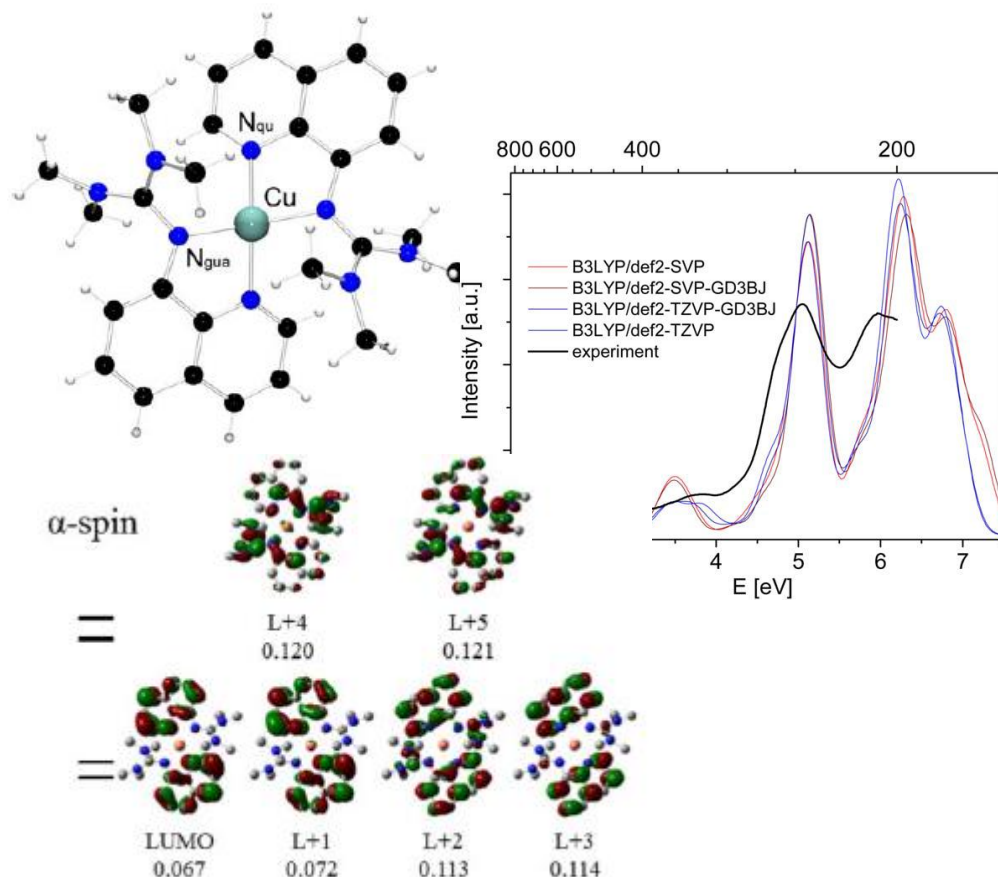
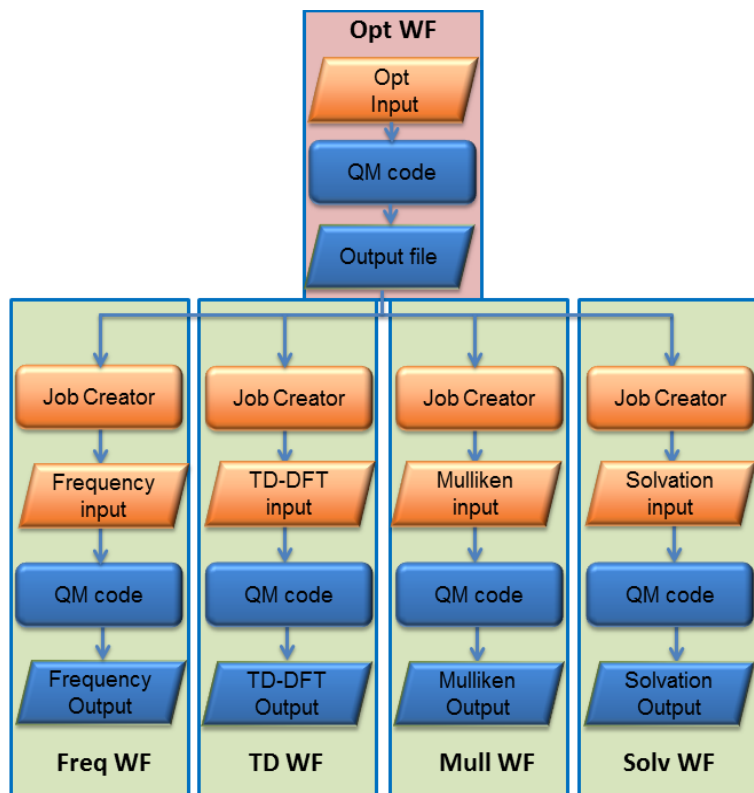
Quantum Chemistry Science Cases

| Name | Description | Workflow Links |
|----------------------------|--|---|
| Spectroscopic Analysis | Explore the spectroscopic characteristics of a molecule | http://www.erflow.eu/spectroscopic-analysis-science-case |
| Spectroscopic Benchmarking | Calculation of optimized geometries, molecular orbitals, population analyses, frequencies, or optical absorptions. | http://www.erflow.eu/spectroscopic-benchmarking-science-case |
| Population UNI | Apply various population schemes to better electronic understanding of the molecules. | http://www.erflow.eu/population-uni-science-case |



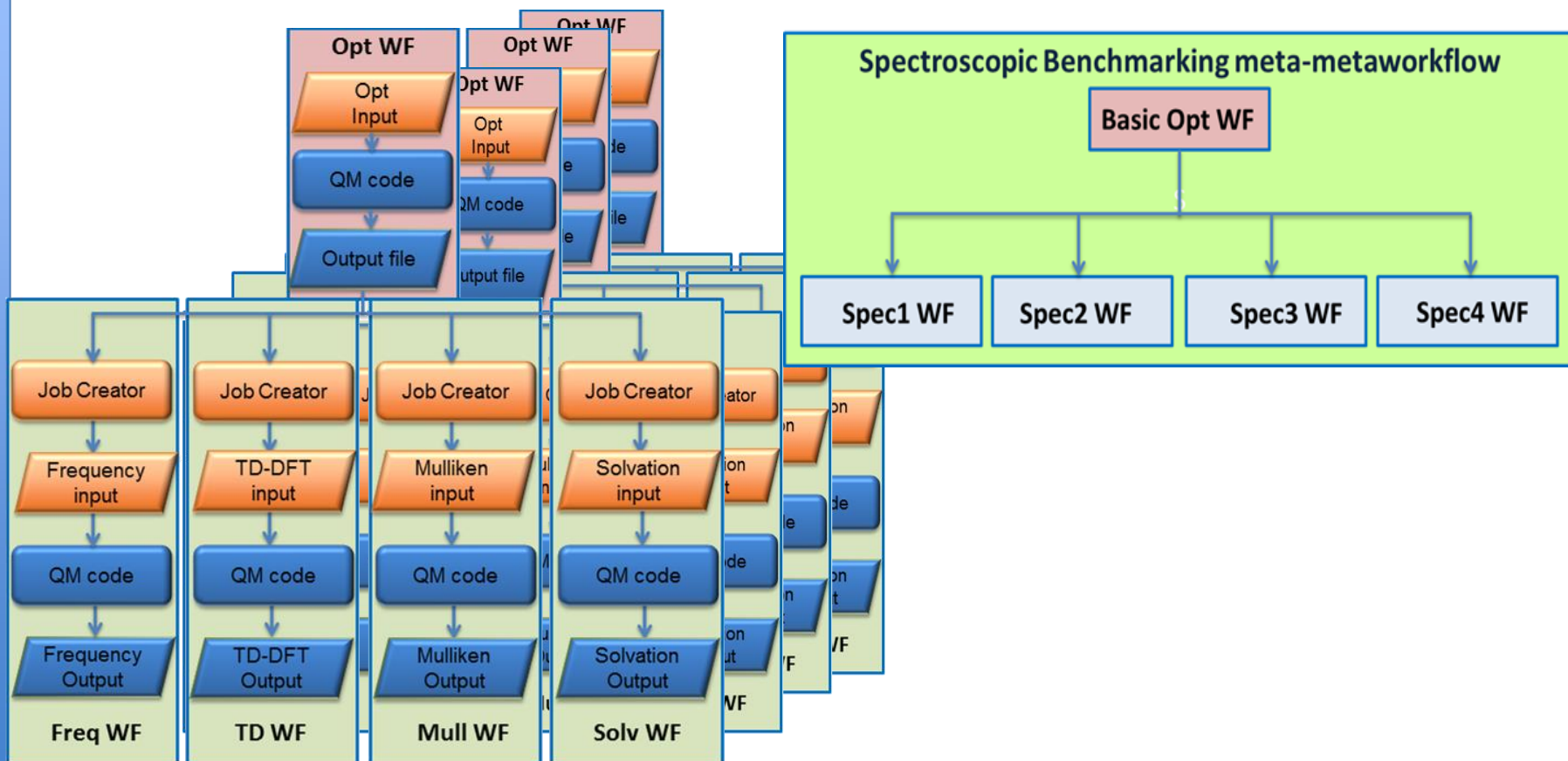
Spectroscopic Analysis Science Cases

- Highly useful for Quantum Chemists in everyday work
- Full simulation of all spectroscopic features of a molecule
- Combination of 5 atomic workflows



Spectroscopic Benchmarking Science Cases

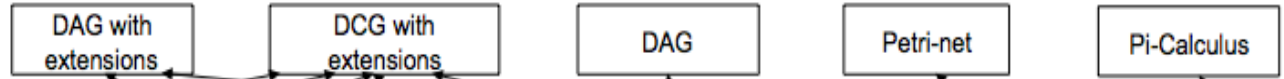
- Highly useful for Quantum Chemists in everyday work
- Combination of 1 atomic workflow and 4 times the Science case 1



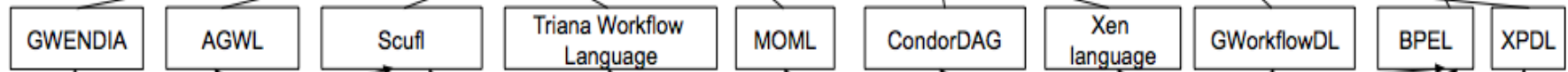


Workflow Interoperability Challenges

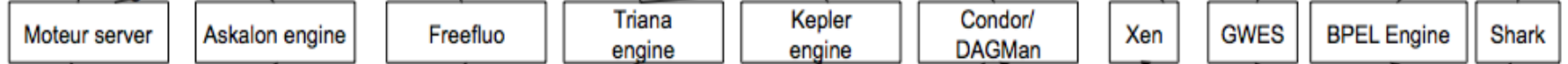
Workflow Formalism



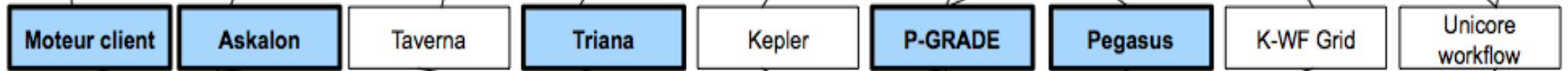
Workflow Language



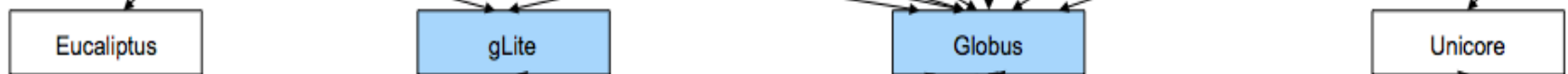
Workflow engine



Workflow system



DCI middleware



DCI infrastructures



Formal Description of WEs and WFs

formal description of workflows

$WF = \{WF_{abs}, WF_{cnr}, WF_{cnf}, WF_{eng}\}$

where

WF_{abs} - abstract workflow

WF_{cnr} - concrete workflow

WF_{cnf} - workflow configuration

WF_{eng} - workflow engine

formal description of workflow engines

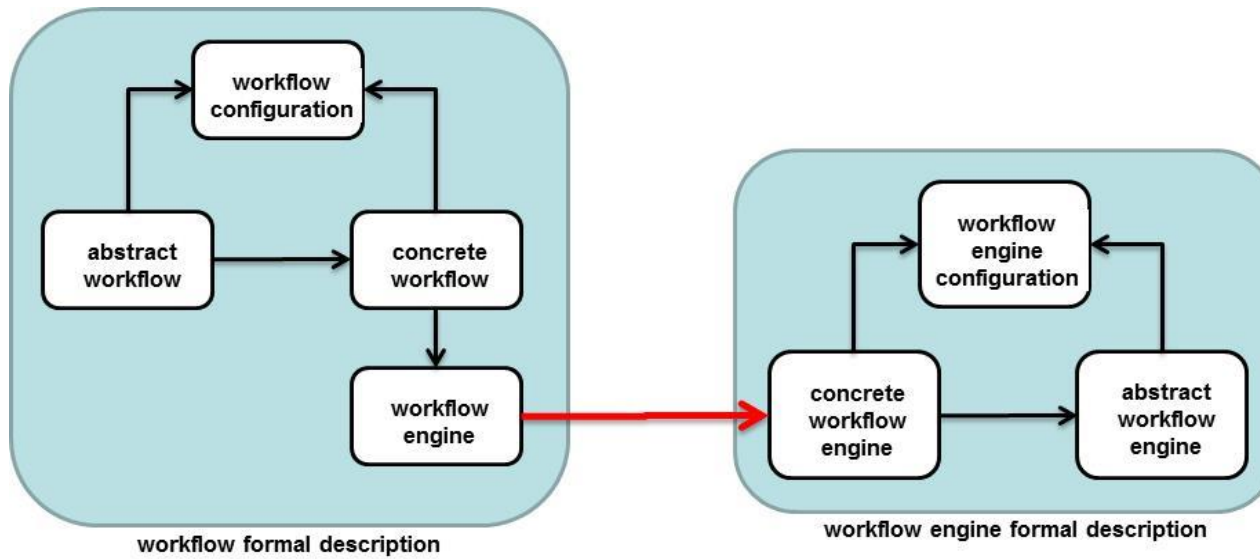
$WE = \{WE_{abs}, WE_{cnc}, WE_{cnf}\}$

where

WE_{abs} - abstract workflow engine

WE_{cnc} - concrete workflow engine

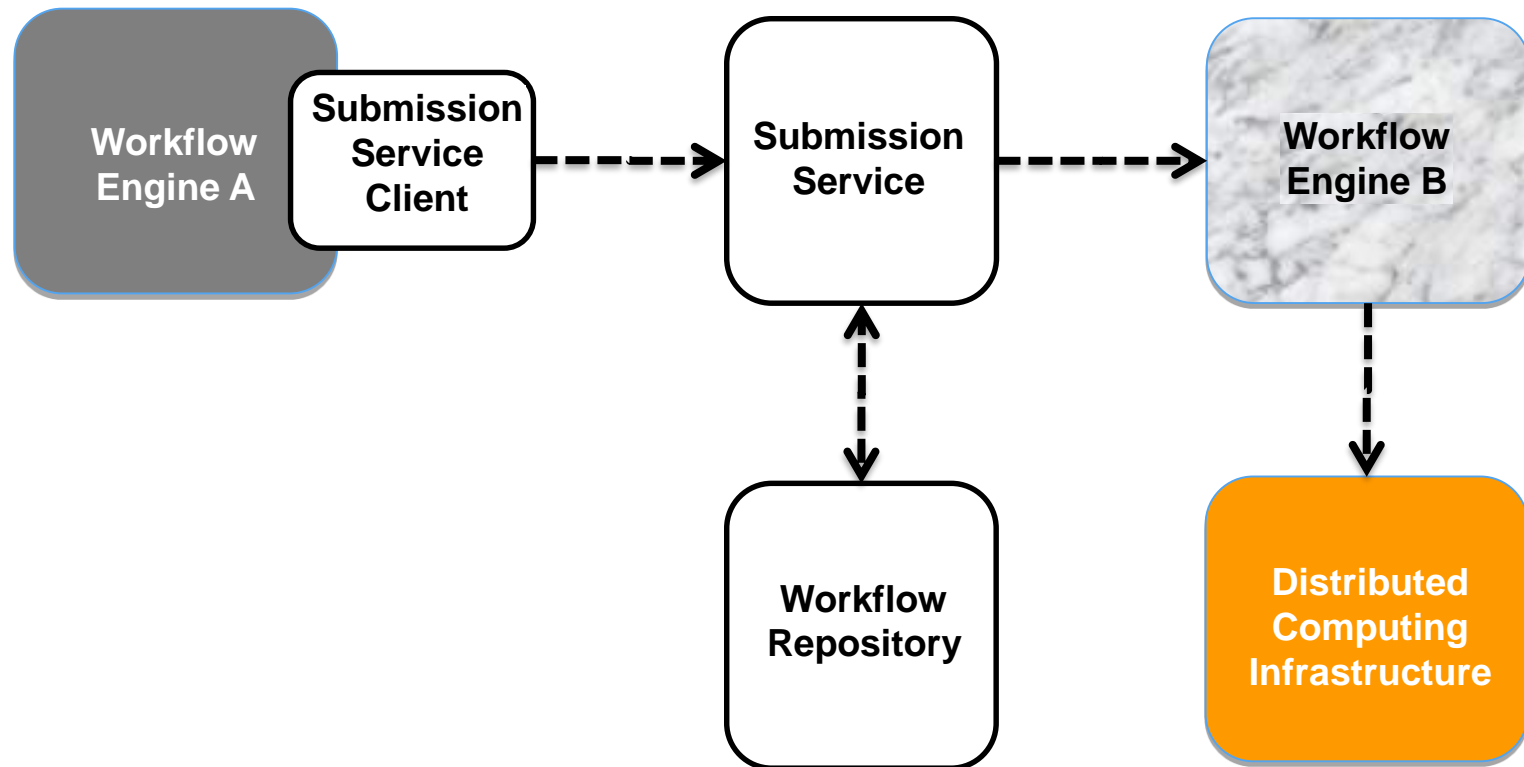
WE_{cnf} - workflow engine configuration





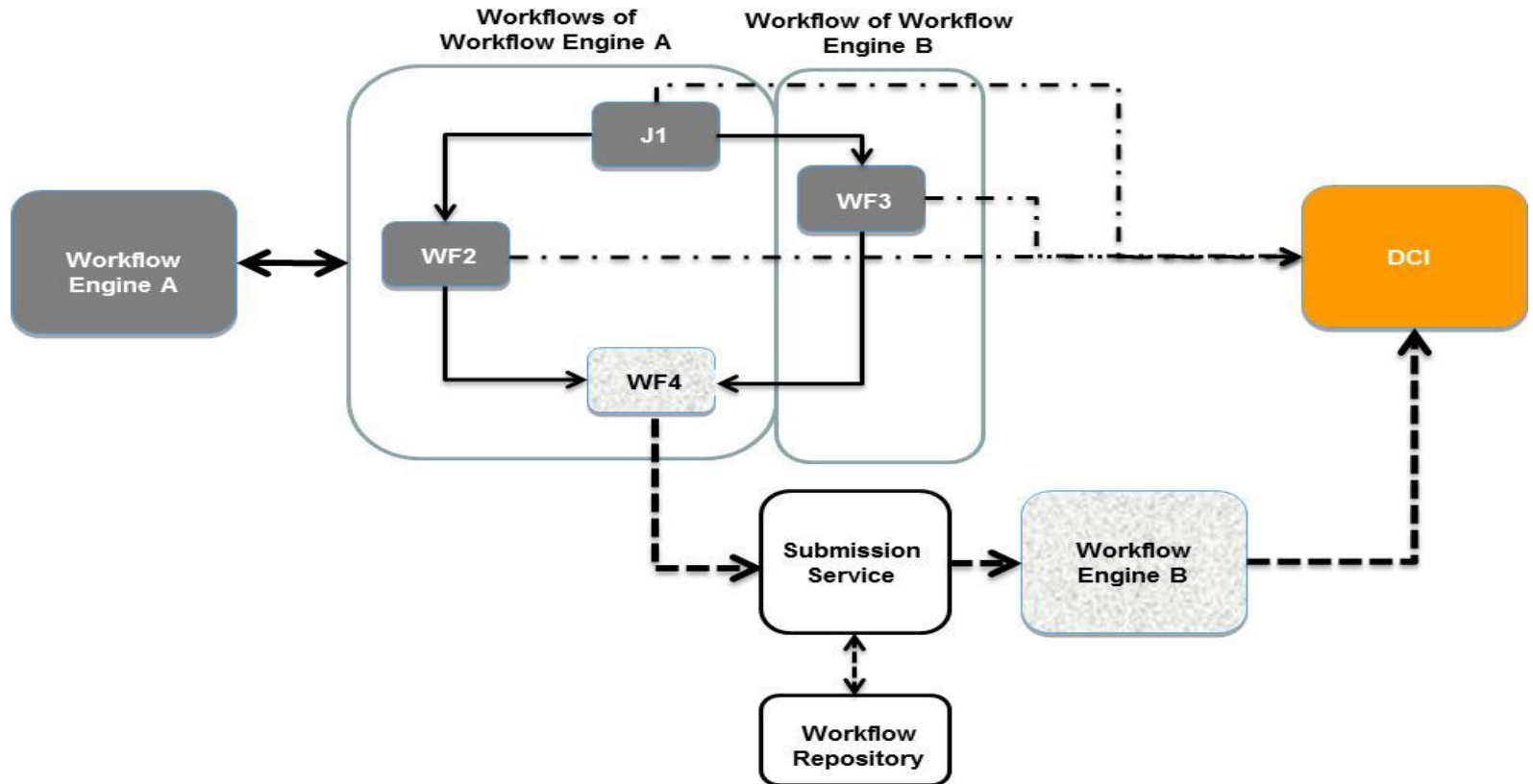
Workflow Interoperability: Coarse-Grained Interoperability (1)

CGI concept = workflow engine integration





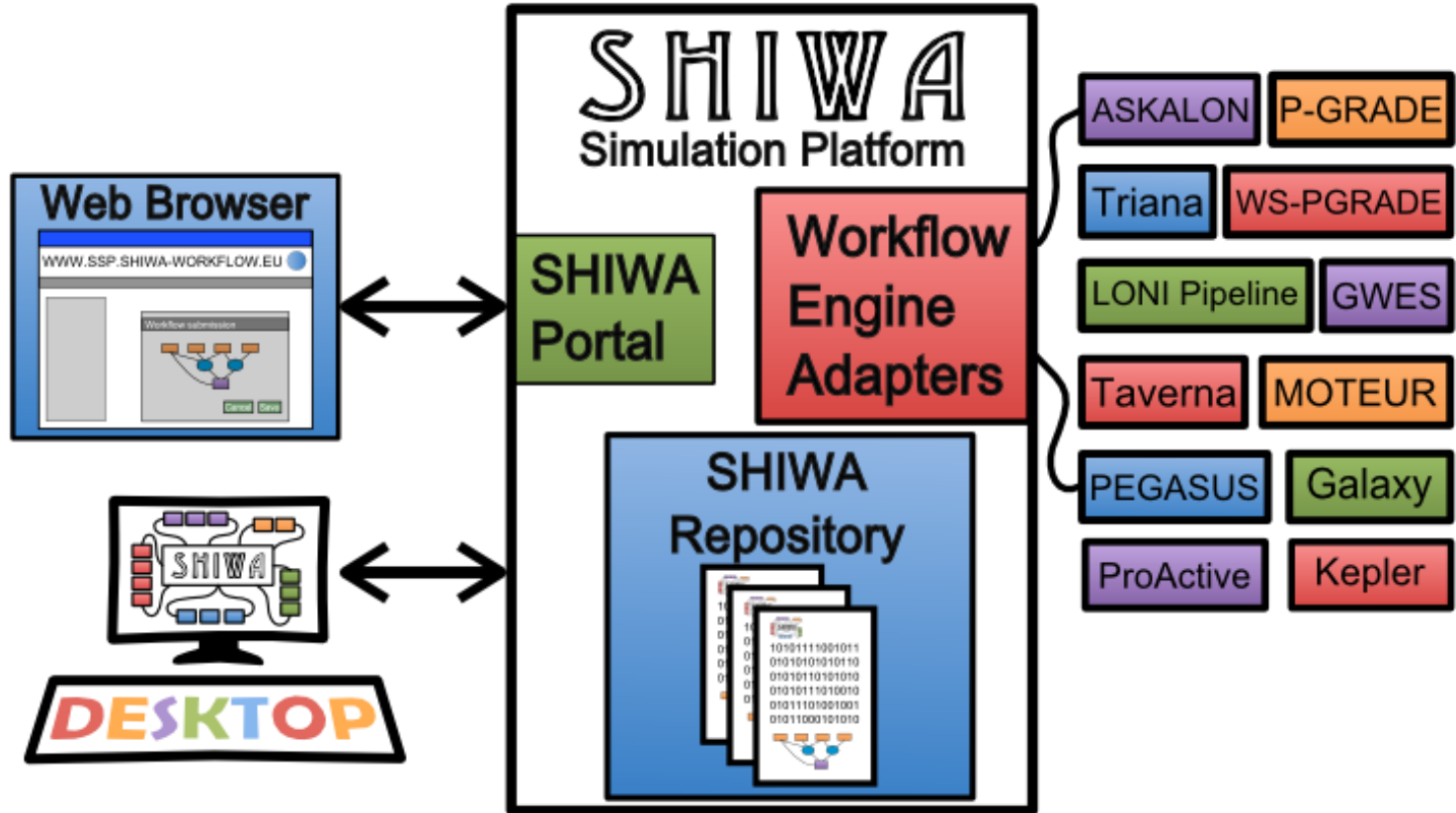
Workflow Interoperability: Coarse-Grained Interoperability (2)



- native workflows: J1, J2, J3
- non-native workflows: WF4
- black boxes which are managed as legacy code applications

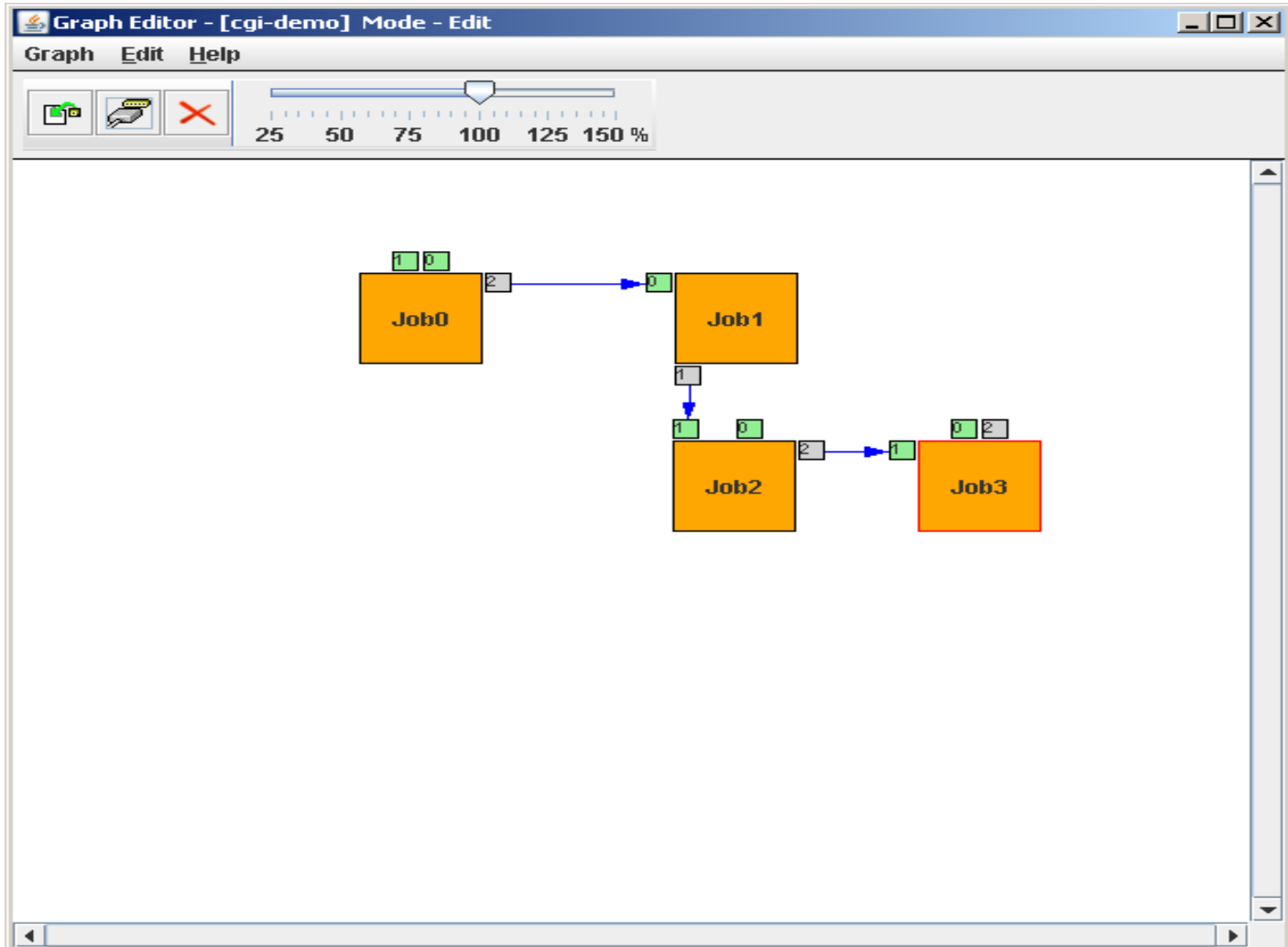


SHIWA Simulation Platform = SSP





SHIWA Portal: Editing Workflow





SHIWA Portal: Configuring Workflow

shiwa-portal.cpc.wmin.ac.uk > Workflow > Concrete

Moteur

Selected Job to be configured

Job's name: Kepler
Optional note: Description of Job

[Job Executable] [Job I/O] [DDL/RSL] [History]

Job execution model:
 Interpretation of the job as a Workflow
 Interpretation of the job as a Library

Type
GEMMLCA Repository:
Service Method:
Resource:
GEMMLCA parameters: 1 input 1 output port(s)

Do not forget to configure the file associations in the Job Inputs and Outputs tab

Save.. Quit

- CIVETPipeline-neugrid: CIVET-LINGA-LonWS-neugrid
- FreeSurfer-LONI-cranium: FreeSurfer-cranium
- GateAdaptors: Gate Adaptors with command line for WE
- Kepler-WF1g: Workflow 1 via GIB
- Kepler-WF2g: Workflow 2 via GIB
- dummy-civet: dummy-civet
- merlin2g: Merlin multiparam kepler via GIB
- testAd: Gate Adaptors with command line for WE
- testbashgib: Simple catter script



SHIWA Portal: Executing Workflow

shiwa-portal.cpc.wmin.ac.uk > Workflow > Concrete

Concrete



Back Refresh

Workflow name: Images-Demo-c1
Note: 2011-8-22
Workflow Graph: ImagesDemo
Workflow Template: --

| Date | Status | Actions |
|-----------------|----------|-----------------|
| 2011-8-23 9:36 | finished | Details Delete |
| 2011-11-24 8:1 | running | Details Suspend |
| 2011-9-9 9:51 | finished | Details Delete |
| 2011-9-9 9:12 | finished | Details Delete |
| 2011-8-22 10:24 | finished | Details Delete |
| 2011-9-12 10:34 | finished | Details Delete |

Selected WF Instance:

2011-11-24 8:1

| Job | Status | Instances | [Actions] |
|---------|---------|-----------|----------------------------------|
| Moteur | init | 1 | View init View all content(s) |
| Taverna | running | 1 | View running View all content(s) |
| Kepler | init | 1 | View init View all content(s) |
| Triana | init | 1 | View init View all content(s) |



SHIWA Workflow Repository

★ Welcome 🏠 Home 📁 Workflows ▾ ⚙️ Implementations ▾ 🛠️ Administration ▾ ? Documentation ▾ ✖️ Log out

Find Workflows

All Domains ▾

subtraction

Search

Show All

Refresh

(1 of 1)



1



10 ▾

Workflow: SimpleWF_IntegerSubtractor

Edit

Details

Workflow Summary

Domain: Demonstration

Application: demonstration

Owner: [Tamas Kukla](#)

Group: shiwaExampleWfs

Keywords: subtraction, integer

Description: This workflow subtracts two integers and outputs the result. The input integers are provided in text files and the result is also a text file containing the difference. This workflow serves demonstration purposes.

Inputs (2) +

Outputs (1) +

Data sets (2) +

Implementation Preview (2)

Kepler Subtract 1.0



Engine: Kepler(1.0)

Version: 1.0

DCIs: SHIWA VO

Keywords: Kepler, local, subtract, integer

Description: This workflow is executed locally to the Kepler engine.

[Edit](#)

Kepler Subtract 1.1



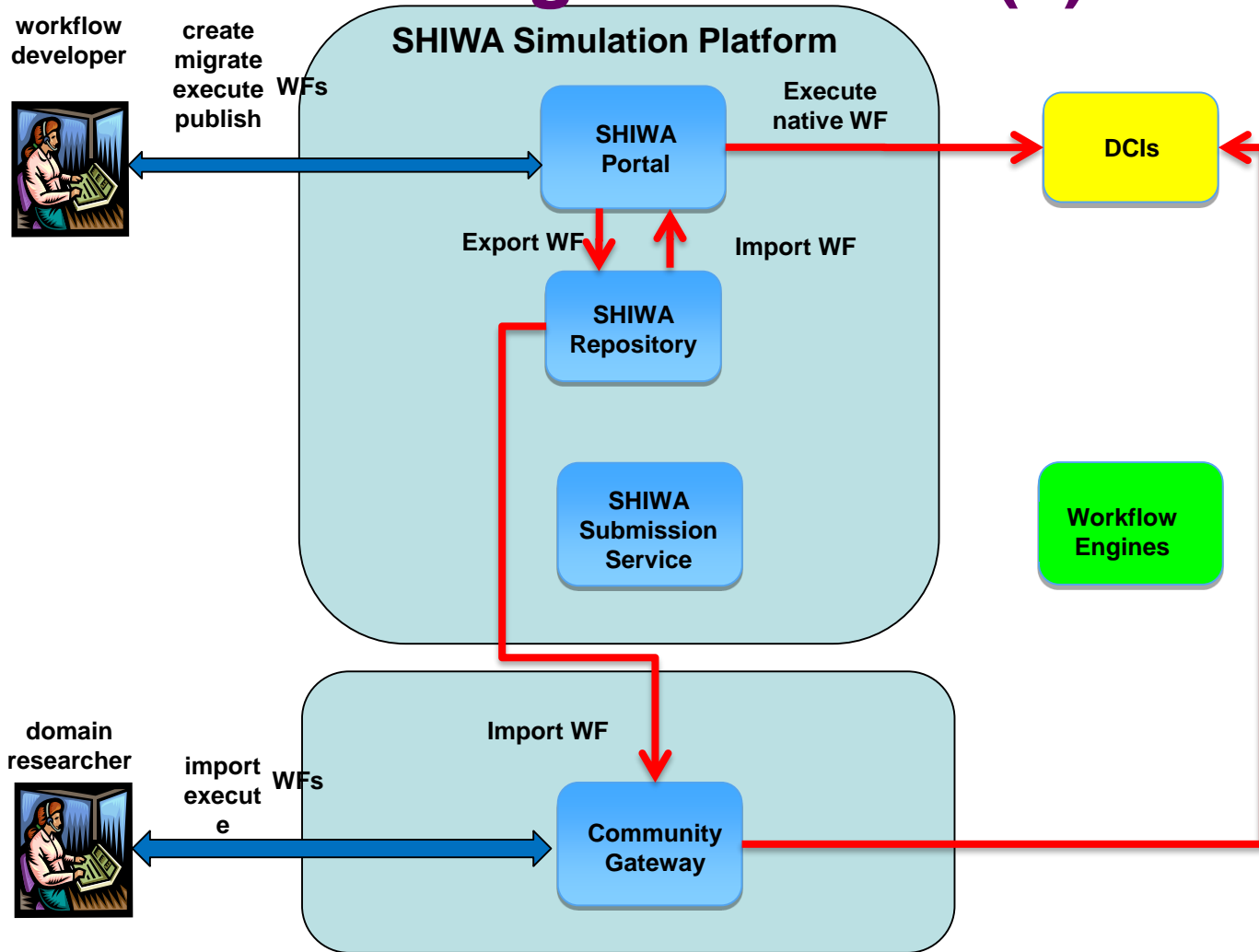
Engine: Kepler(1.0)

Version: 1.1

DCIs: SHIWA VO

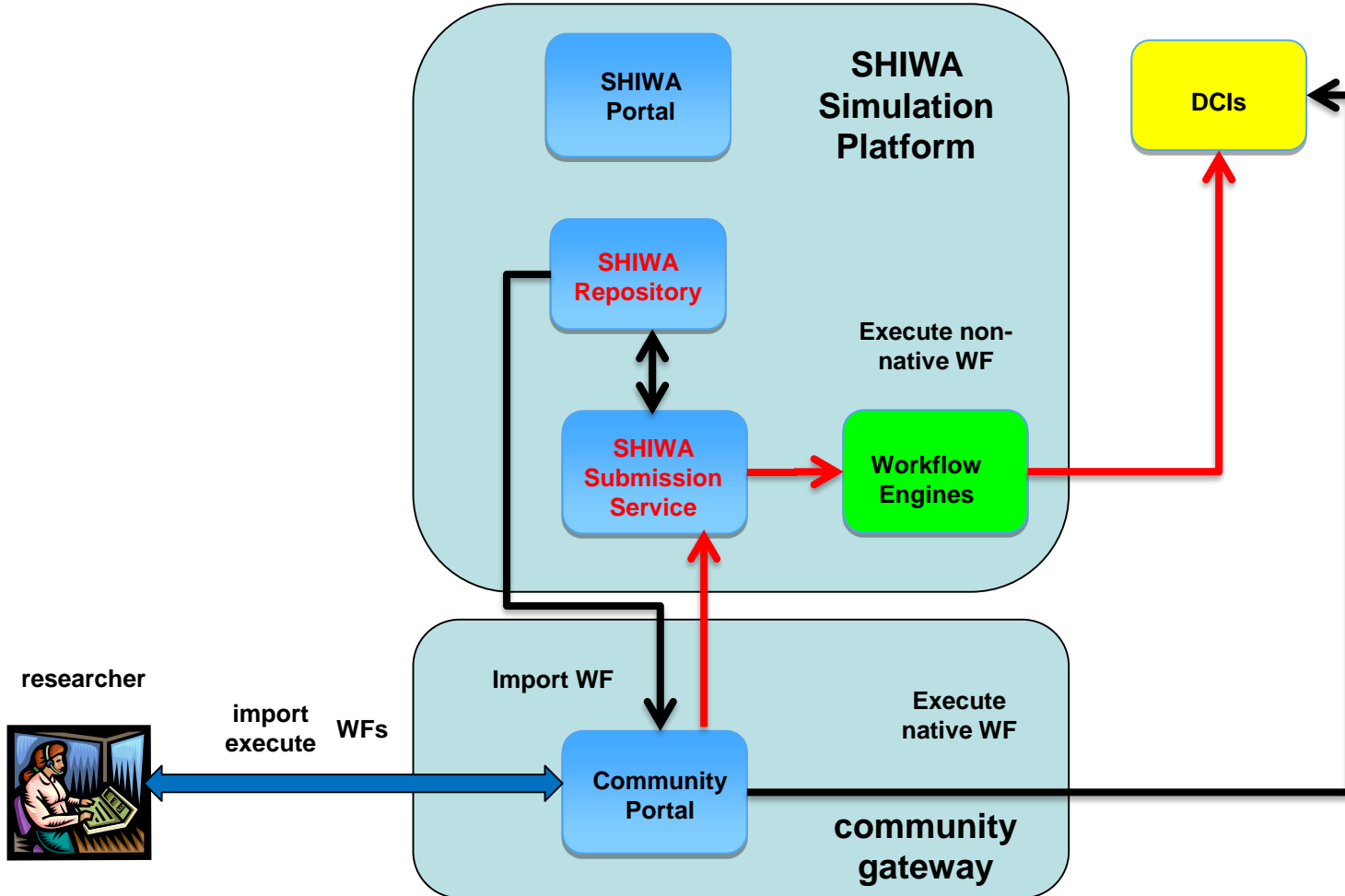


Workflow Interoperability: CGI Usage Scenario (1)



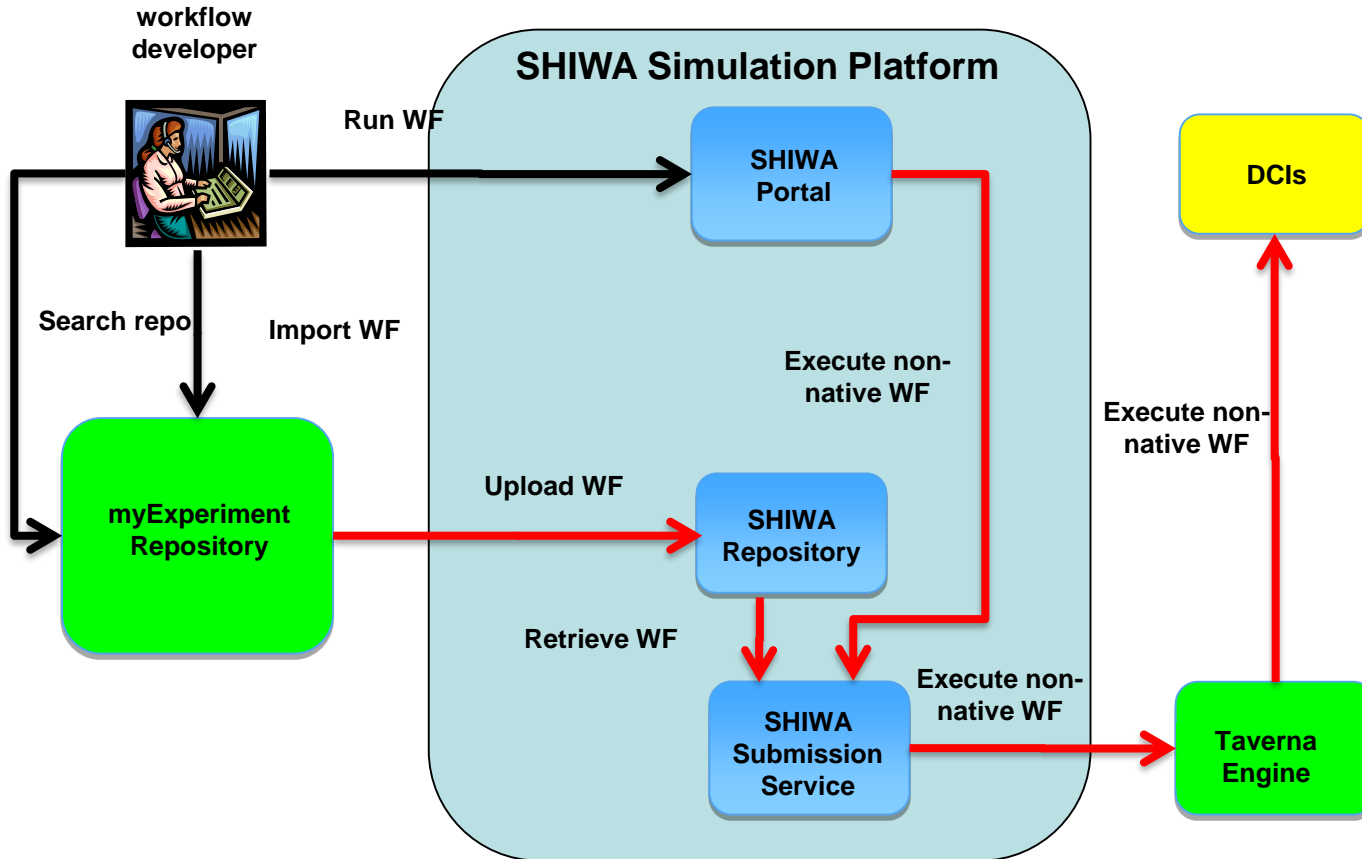


Workflow Interoperability: CGI Usage Scenario (2)





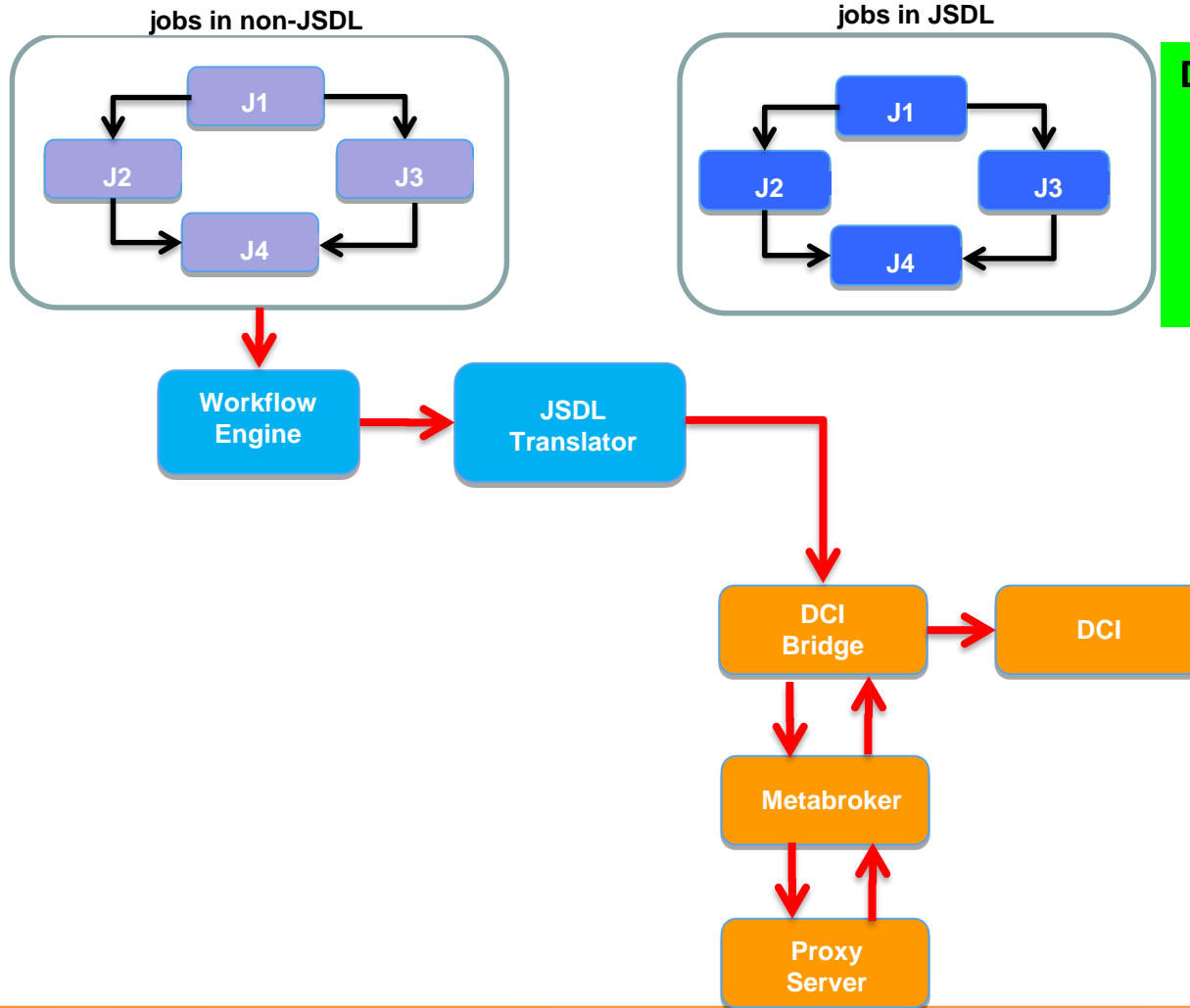
Workflow Interoperability: CGI Usage Scenario + Taverna WF



CGI support:
ASKALON
Dispel4Py
Galaxy (???)
GWES
Kepler
MOTEUR
Pegasus
PGRADE
ProActive
Taverna
Triana



Distributed Computing Infrastructure Interoperability




DCI support:
cloud
cluster
desktop grid
service grid
supercomputer









Usage Scenario: Domain Researcher View

End User



Workflow name: PublicAutoDock423_2012-08-24-100713

Note: 2012-5-22

| | |
|---------------------------------------|---|
| receptor.pdb |  C:/fakepath/receptor.pdb <input type="button" value="Tallózás..."/> |
| docking.gpf |  C:/fakepath/docking.gpf <input type="button" value="Tallózás..."/> |
| docking.dpf |  C:/fakepath/docking.dpf <input type="button" value="Tallózás..."/> |
| ligand.pdb |  C:/fakepath/ligand.pdb <input type="button" value="Tallózás..."/> |
| Number of work units |  1000 |
| Maximum number of best results |  5 |
| Message: | |



Usage Scenario: Domain Researcher View

Navigation: Welcome | Storage | Settings | **End User** | Help | Information | Security | Statistics

ws-pgrade > End User > Configure

End User

Workflow name: UserDemo_calls_embedded_2012-05-17-163833
Note: 2012-5-15

Workflow status: **running**

| Status | Instances |
|-----------------|-----------|
| init | 2 |
| running: | 15 |
| done: | 1 |
| error: | 0 |
| sum: | 18 |


Estimation of accumulated progress: (1 / 18)

Powered By [Liferay](#)



Usage Scenario: Customised View

UNIVERSITY OF WESTMINSTER
University of Westminster
Desktop Grid Portal



Welcome Statistics Help **Docking portlet** Mental Ray

University of Westminster Desktop Grid Portal > Docking portlet

Autodock

Vina AutodockAutogrid < AutodockNo [How to use docking - Help](#)

found results: #5

AutoDock4 - Random blind docking requiring pdb input files

This application requires pdb input files and docks a small ligand molecule on a larger receptor molecule structure in a Monte-Carlo simulation. The workflow uses version 4.2.3 of the AutoDock docking simulation package. For more information on required input parameters please see the window below.
For generic help on AutoDock please see [Autodock usage](#)

| Autodock - specify the inputs | Field description |
|---|---|
| Receptor file (*.pdb) <input type="button" value="Choose File"/> No file chosen | Select and upload the Receptor file for docking. Sample receptor download. |
| Autogrid map file (*.gpf) <input type="button" value="Choose File"/> No file chosen | Select and upload the Autogrid parameter file. Specifies the spatial attributes of the grid maps. Sample map file download. |
| Docking parameter file (*.dpf) <input type="button" value="Choose File"/> No file chosen | Select and upload the Docking parameter file. Specifies the files and parameters for the docking. Sample parameter file download. |
| Ligand molecule file (*.pdb) <input type="button" value="Choose File"/> No file chosen | Select and upload the ligand file. Sample ligands download. |
| Number of docking 10 | Specify the number of experiments you want to carry out. Default value: 10 |
| Best result number 5 | Specify the number of the lowest Autodock score results you want to keep. Default value: 5 |
| Task name Task1 | Specify the name of current task, use only letters or numbers. |
| <input type="button" value="Run task"/> | |

https://dg-portal.wmin.ac.uk/liferay-portal-6.1.0/web/guest/docking_portlet

| Task Status | Actions |
|-------------|---------|
|-------------|---------|



Usage Scenario: Customised View

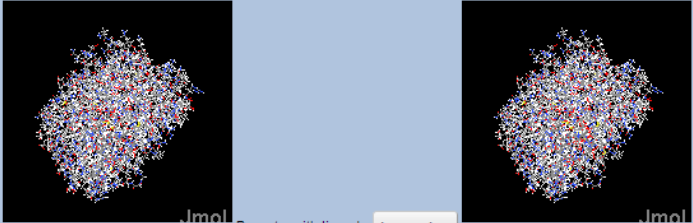
University of Westminster Desktop Grid Portal > Docking portlet

| Task Name | Task Status | Actions |
|----------------------------|--|---------------------|
| Task1 2013-04-23-153020 | FINISHED (0 error, 3 finished, 0 running) | Show results Delete |

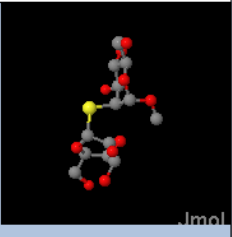
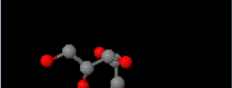
Results for task: Task1 2013-04-23-153020

Download best outputs Download log.dlg zip Re-compute task

Receptor file, with docked ligands: Title:



Receptor: Large view Receptor with ligands: Large view

| Ligand molecule title | Ligand molecule info | Ligand molecule image |
|-----------------------|---|--|
| Title: | Autodock Score: -5.14 [kcal/mol] Download Molecule |  Large view |
| | |  |



Knowledge Transfer and Research Community Support

Academic communities

| | |
|-------------------------|------------------------------------|
| Astrophysics | PGRADE + Taverna |
| Computational Chemistry | Galaxy + PGRADE + UNICORE |
| Heliophysics | PGRADE + Taverna |
| Hydrometeorology | PGRADE |
| Life Sciences | Galaxy + Moteur + PGRADE + Taverna |
| Meteorology | PGRADE |
| Material Sciences | PGRADE |
| Particle Physics | PGRADE |
| Seizmology | Dispel4Py + PGRADE |

Non-academic communities

| | |
|------------------------------------|--------|
| Engineering and manufacturing SMEs | |
| Business Process Simulation | |
| Discrete Event Simulation | |
| Fluid Dynamics Simulation | |
| | PGRADE |



Creating and Executing Workflows

workflows in the repository in 2013

| | |
|--------------|--------------|
| abstract | - 123 |
| concrete | - 119 |
| total | - 242 |

workflows in the repository in 2016

| | |
|--------------|--------------|
| abstract | - 213 |
| concrete | - 385 |
| total | - 598 |

workflow execution number
SHIWA Simulation Platform

- 512 (dev) / 331 (test) / 181 (training)

Community gateways

| | |
|-------------------------|---------------------------------------|
| Astro workflows | - 182 (dev) / 73 (test) 203 (prod) |
| Compchem workflows | - 550 (dyn) / 400 (dock) / 300 (quan) |
| Helio workflows | - 41 (prod) |
| Life Sciences workflows | - 79 (dev) / 325 (prod) |



Research Community Support

Phase 1 – introduction to the workflow technology

Platform: SHIWA Simulation Platform
Support: workflow creation and execution
Training: platform and workflow training

Phase 2 – creating and running workflows

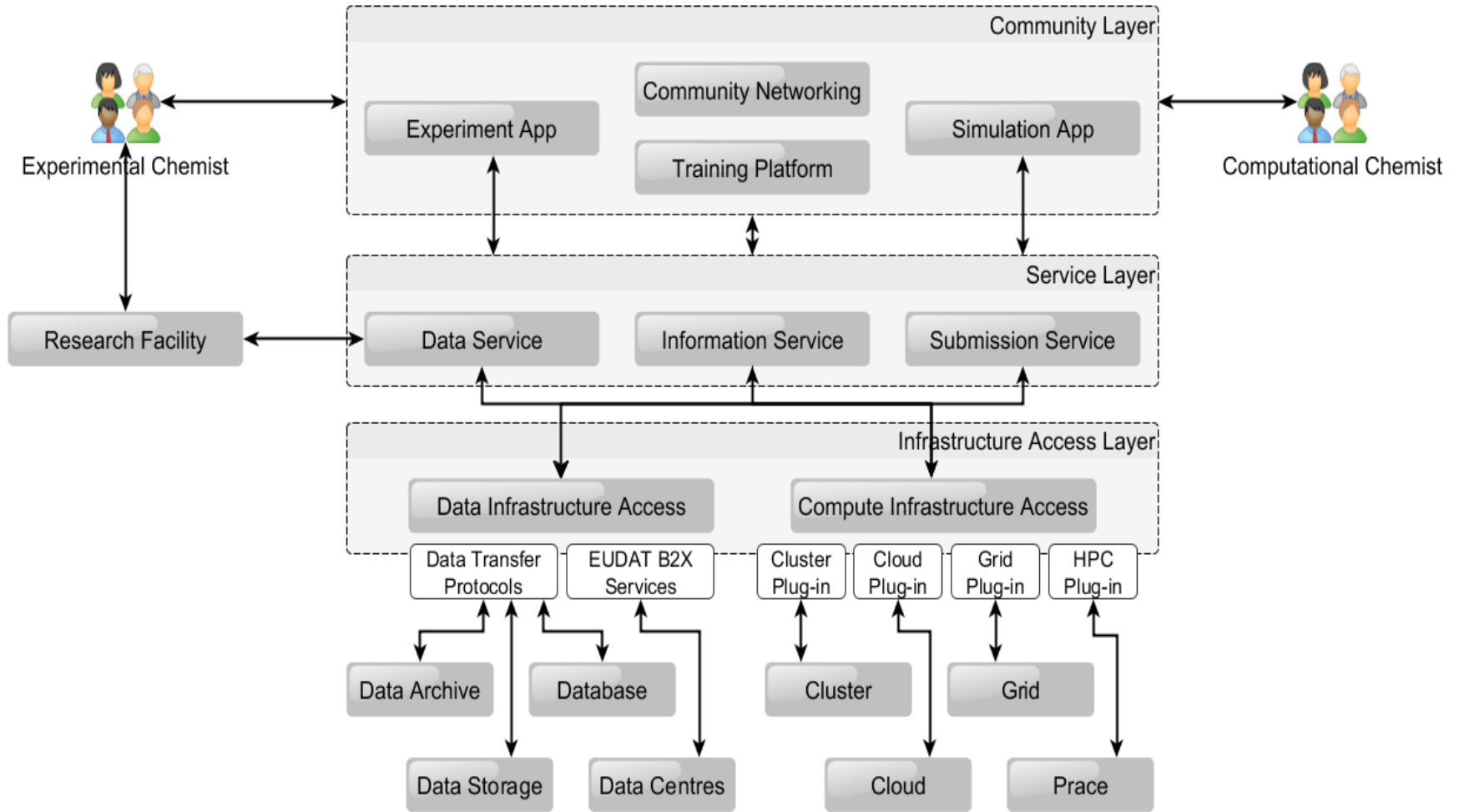
Platform: SHIWA Simulation Platform or community portal
Support: portal deployment + workflow porting
Training: gateway deployment and management

Phase 3 – combining workflows of different workflow systems

Platform: SHIWA Simulation Platform + community portal
Support: access to repository + submission service
Training: CGI training



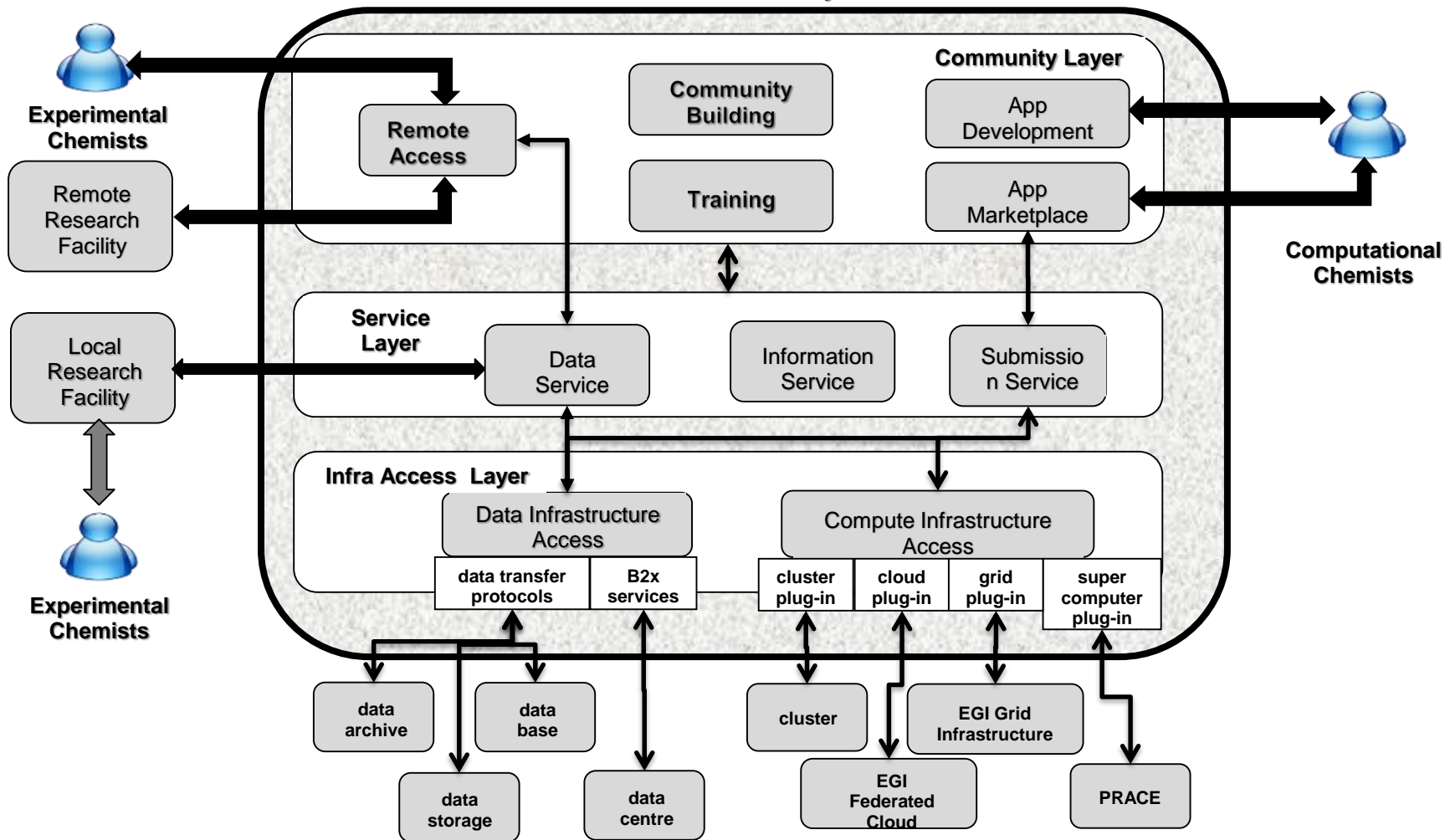
Architecture of the Research Infrastructure





Access to the Research Infrastructure

Science Gateway





Access to the Research Infrastructure

community layer

- will offer social media type services allowing Experimental Chemists to run experiments on remotely available research facilities.
- will provide to access to simulation applications to run simulations for Computational Chemists.
- will also support training activities and community building.

service layer

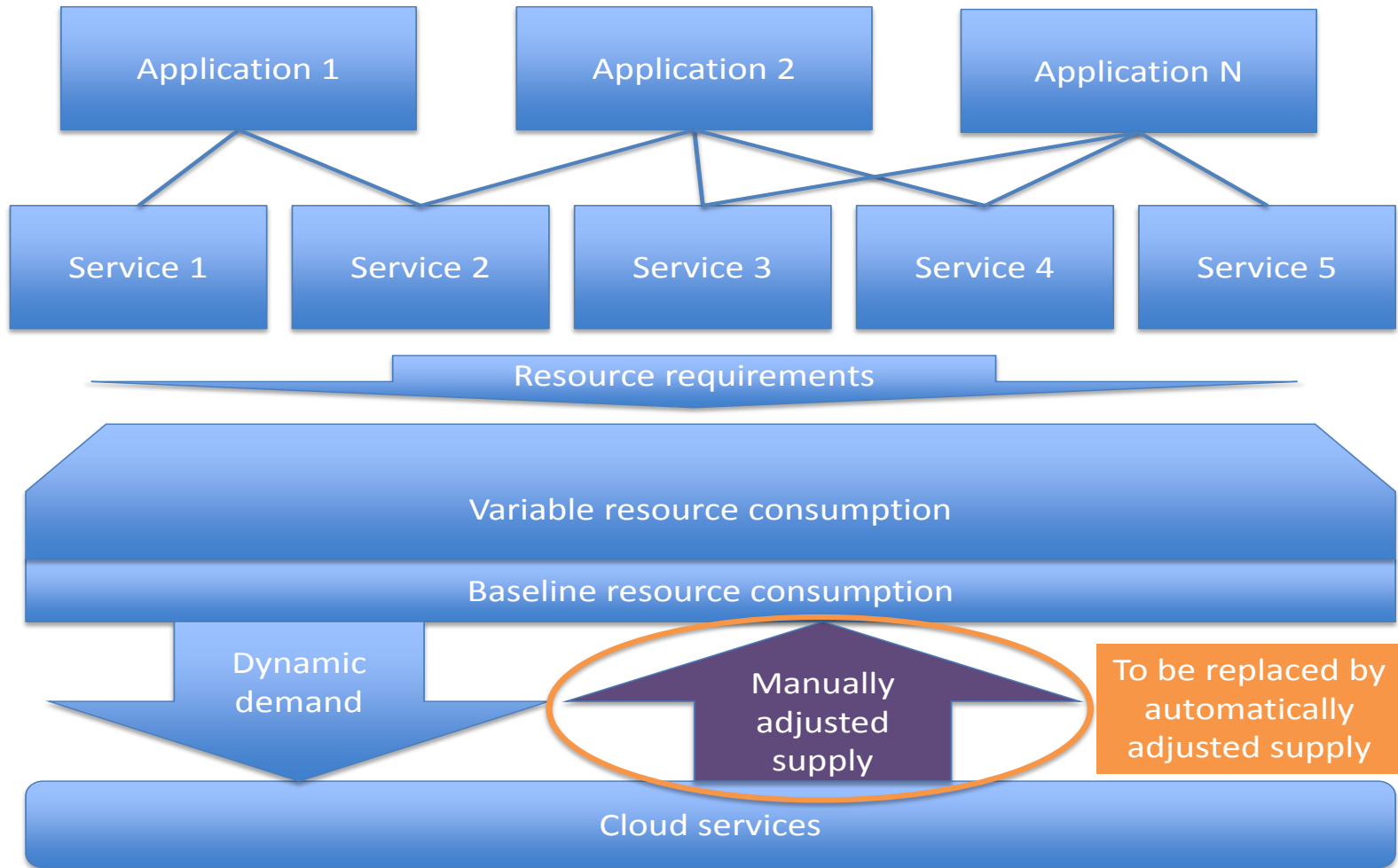
- will connect researchers to the research facilities and e-infrastructure resources using microservices managed by a service orchestrator.
- will provide data service that will connect Experimental and Computational Chemists through scientific data
- Experimental Chemists will use the data service to manage experimental data while Computational Chemists will run simulations through the submission service using the data service.

infrastructure access layer

- computing infrastructure access service - will manage access to major computing resources such as cloud, cluster, grid and supercomputer.
- data infrastructure access service - will manage data using different types of data resources, such as data archives, databases, data collections, data storages using EUDAT B2xx and MASi services, and major data transfer protocols

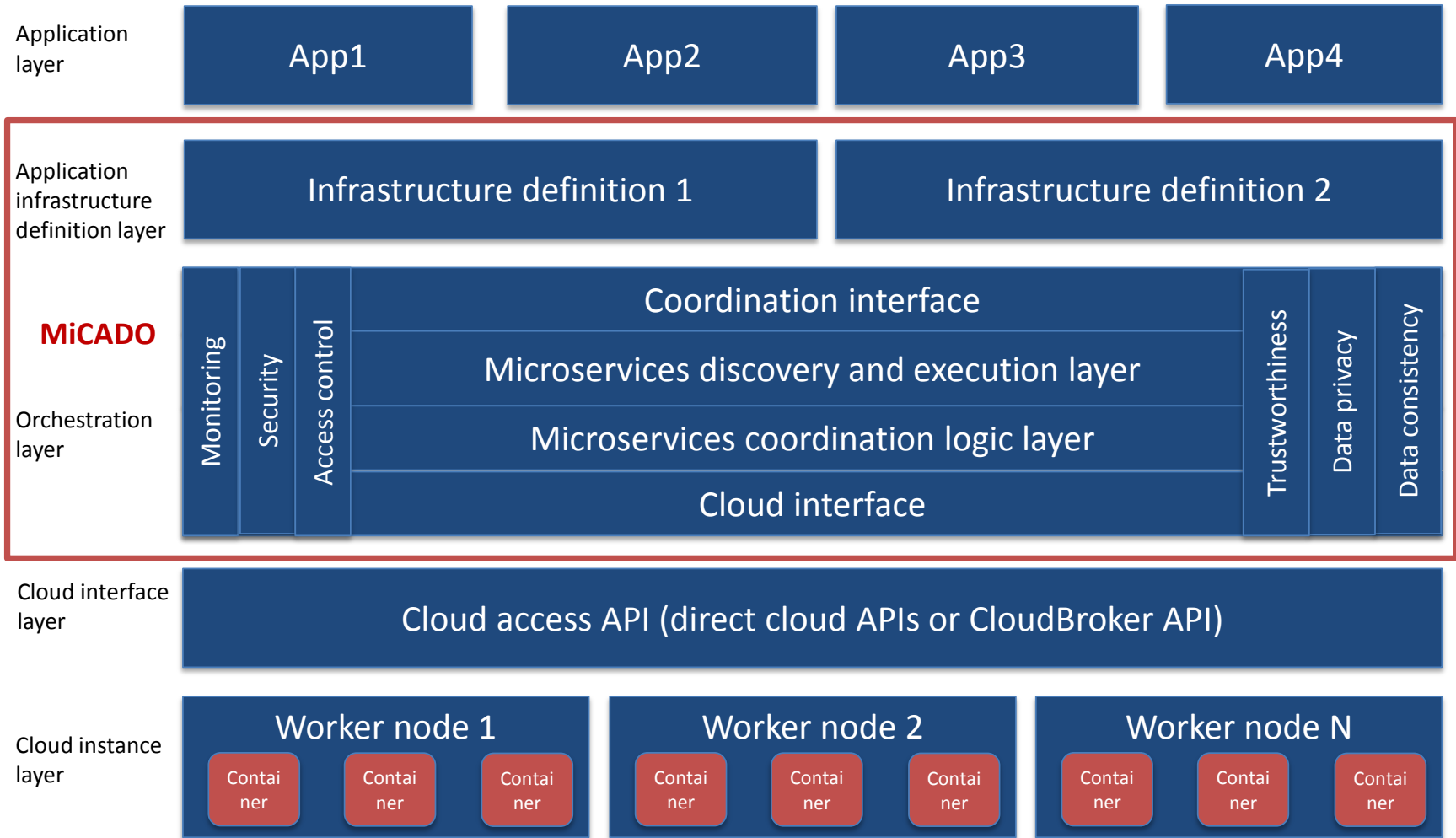


Access to Cloud Resources in the Research Infrastructure (1)



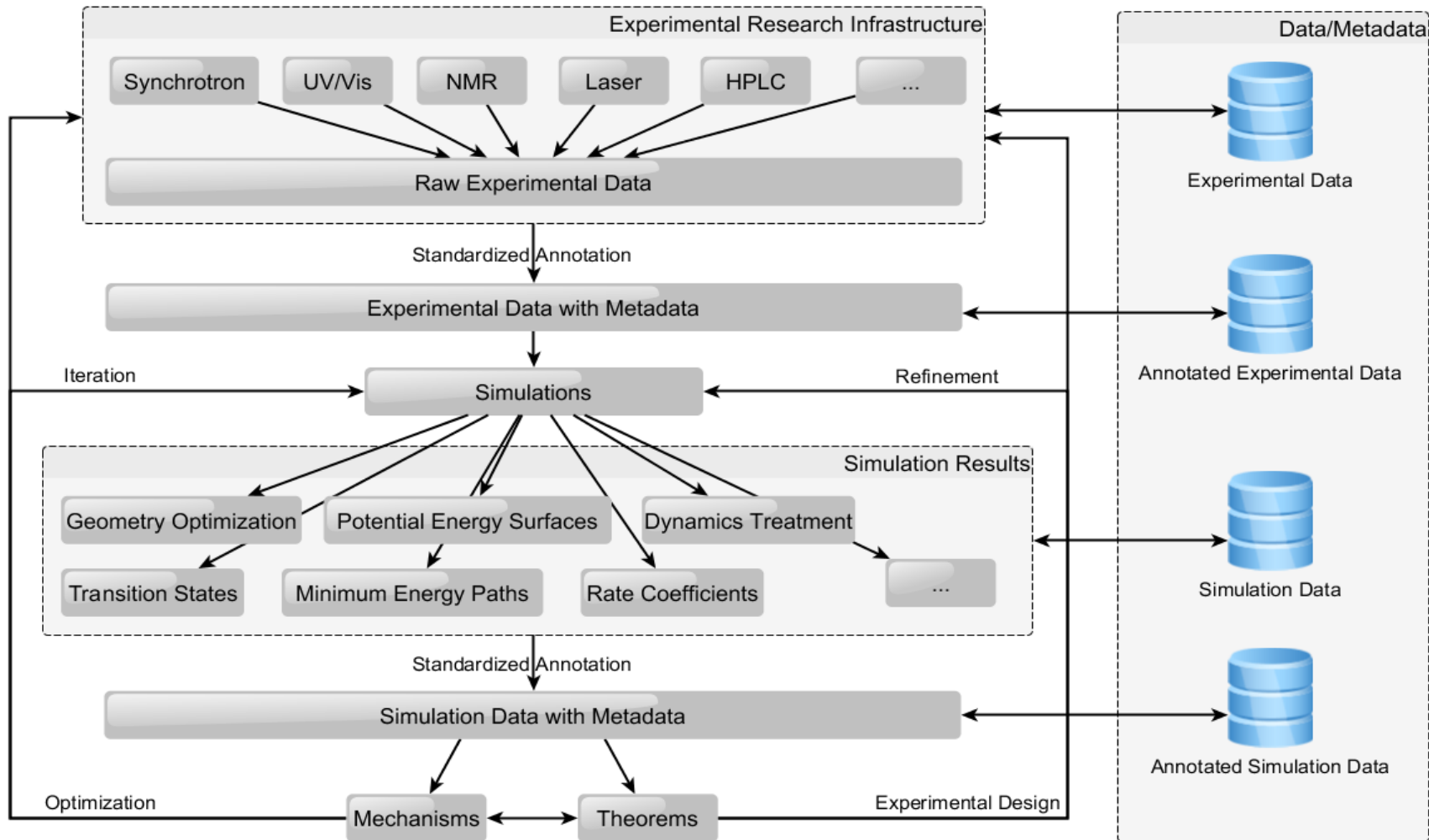


Access to Cloud Resources in the Research Infrastructure (2)





Data Flows in the Research Infrastructure





Acknowledgements

Prof. Sonja Herres-Pawlis

Dr. Jens Kruger



Questions?