



Enabling Grids for E-scienceE

NA4/Biomed Demonstration

Medical Data Management and processing

EGEE 3rd review rehearsal, May 4th, 2006

Johan Montagnat

Tristan Glatard

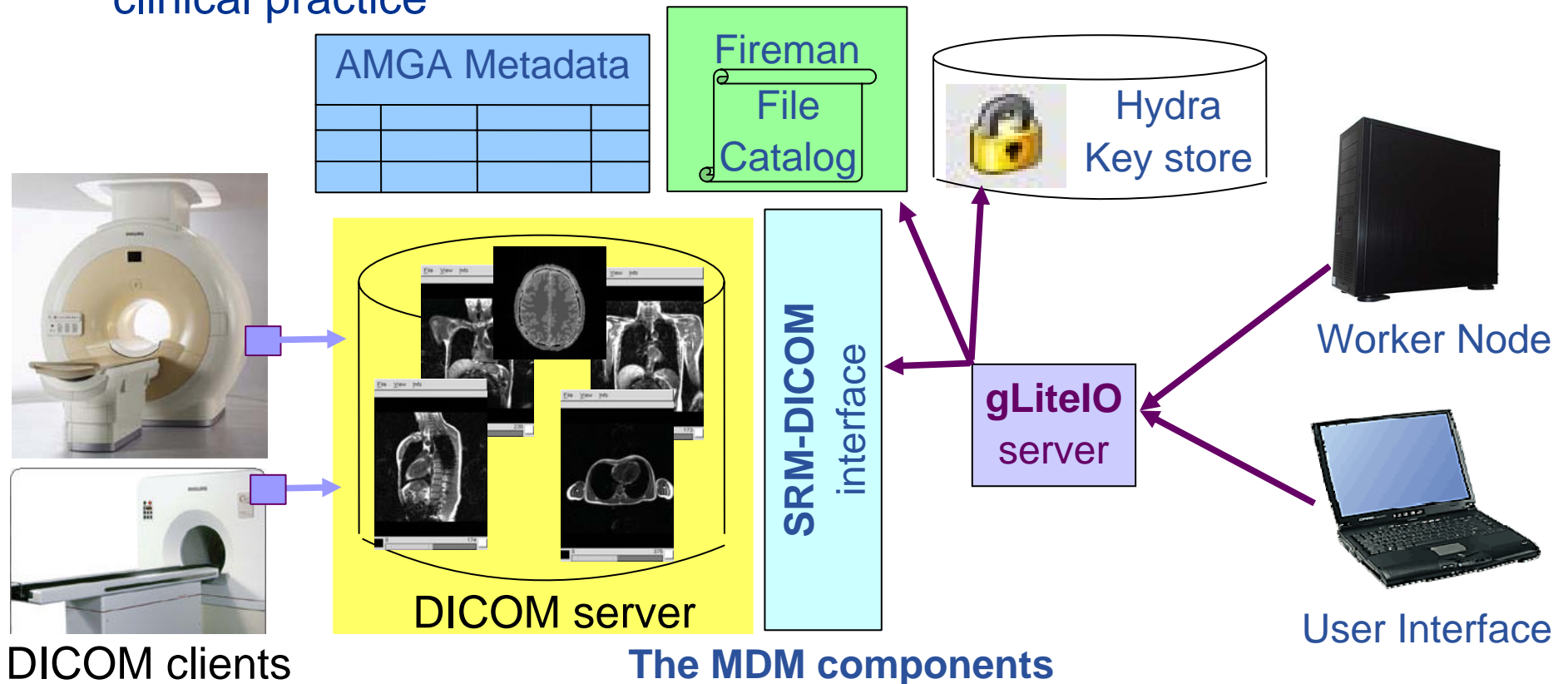
www.eu-egee.org



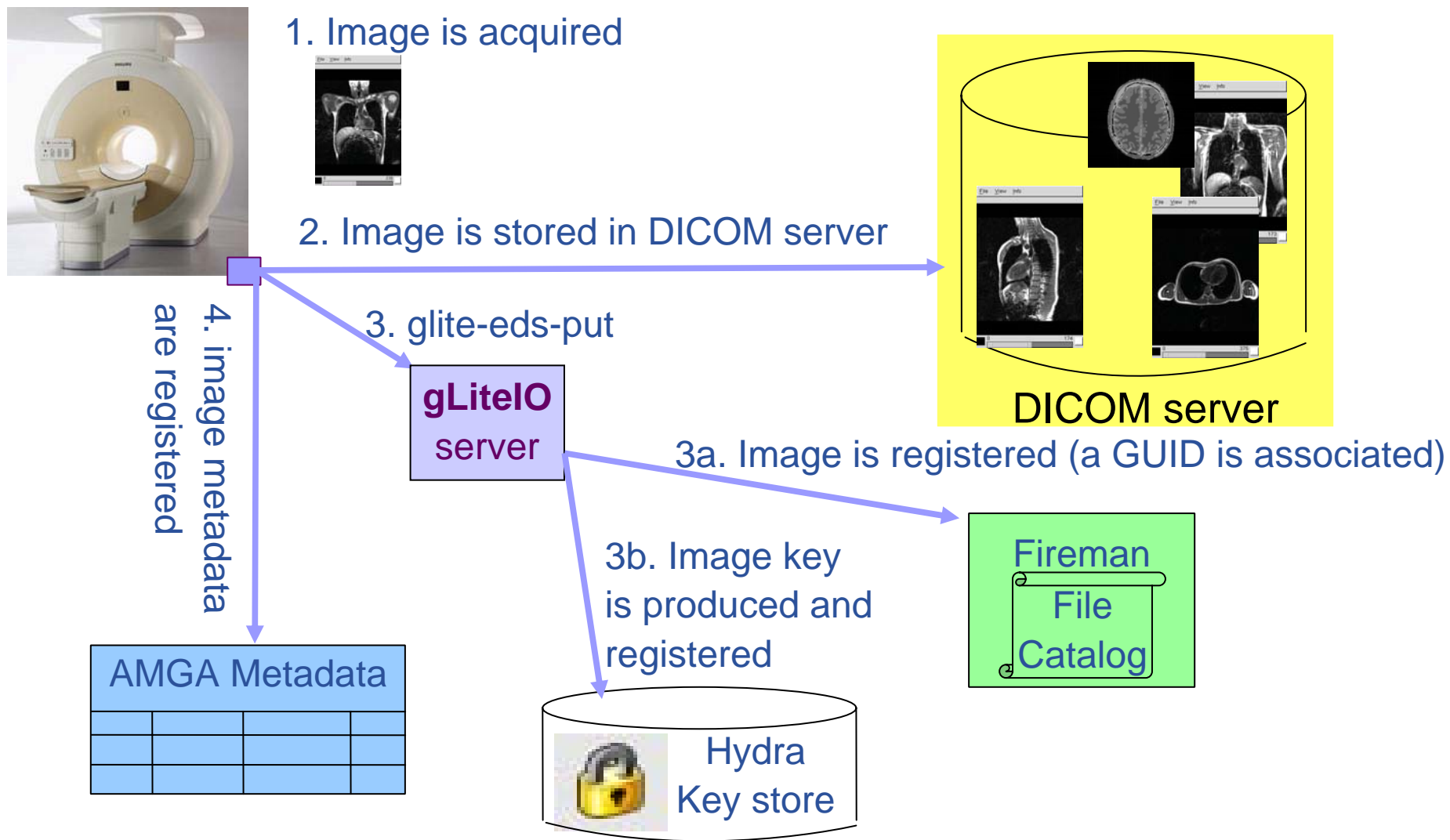
- **Medical Data Manager**
 - Interface to clinical data storage (DICOM)
 - Integrated to gLite 1.5 middleware
 - Tackling data security and privacy needs
 - Result of MDM TCG Working Group
- **Application to medical images registration assessment**
 - Data intensive workflow-based application
 - Scientific results in the medical image processing area with consequences for clinical use
- **Immediate scheduling of jobs submitted for the demonstration**
 - Torque+MAUI configuration for efficient handling of short jobs
 - Result of the SDJ TCG Working Group

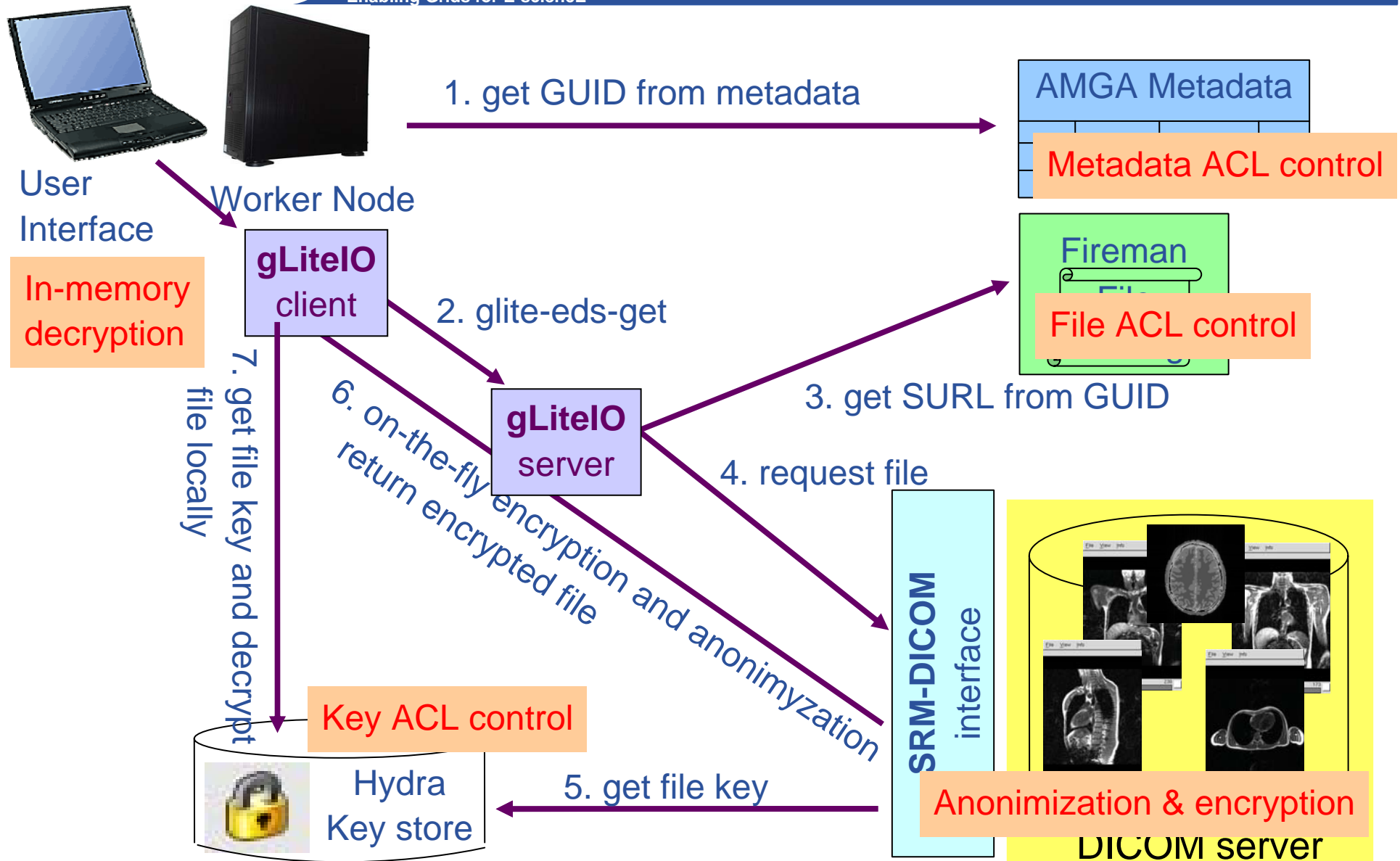
- Objectives

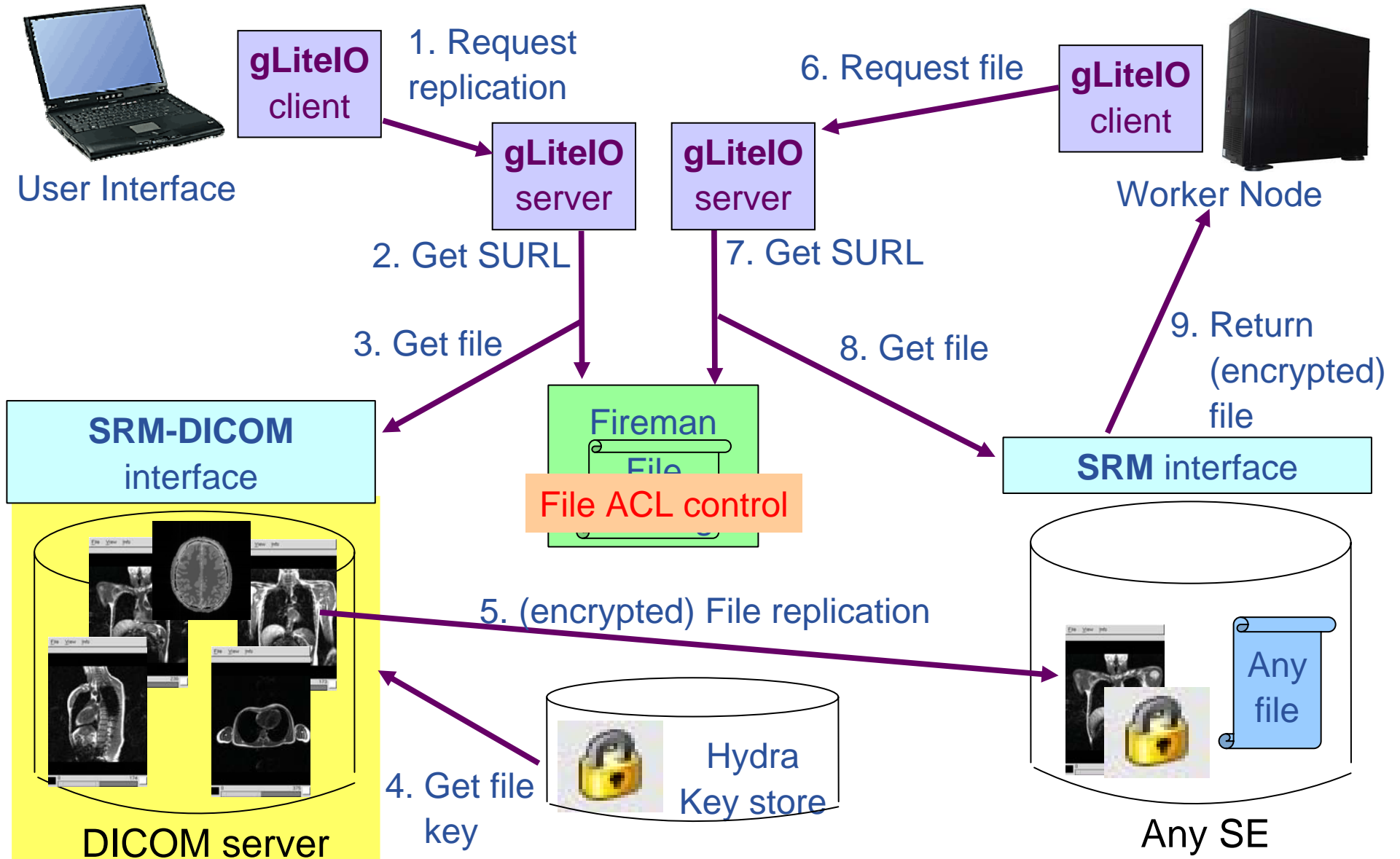
- Expose an standard grid interface (SRM) for medical image servers (DICOM)
- Fulfill application security requirements without interfering with clinical practice



- **Computing interface to medical DICOM storage**
 - Data are acquired from the hospital imagers in native DICOM format
 - Standard SRM interface exposed to the grid
 - DICOM slices are assembled in 3D images
- **Privacy**
 - Fireman provides **gLite 1.5** **ACLs**
 - **gLiteIO** provides **service** **content access control**
 - **AMGA** provides **secured communication** and **ACLs**
 - **SRM-DICOM** provides **on-the-fly data anonymization**
 - It is based on **ARDA service** **cache implementation (SRM v1.1)**
- **Data protection**
 - **Hydra** provides **gLite 1.5 service** **encryption/decryption transparently**







- **Medical image registration**
 - Registration algorithms
 - Real clinical impact
- ~~Interfaced to the medical infrastructure~~
 - ~~To retrieve suitable input images~~
- **Compute intensive**
 - Medical image registration algorithms: minutes to hours of computations on PCs
- **Data intensive**
 - Hundreds to thousands of image pairs
- **Workflow-based**
 - Using the **MOTEUR** service-based workflow manager
 - Developed in the French ACI “Masse de données” AGIR project

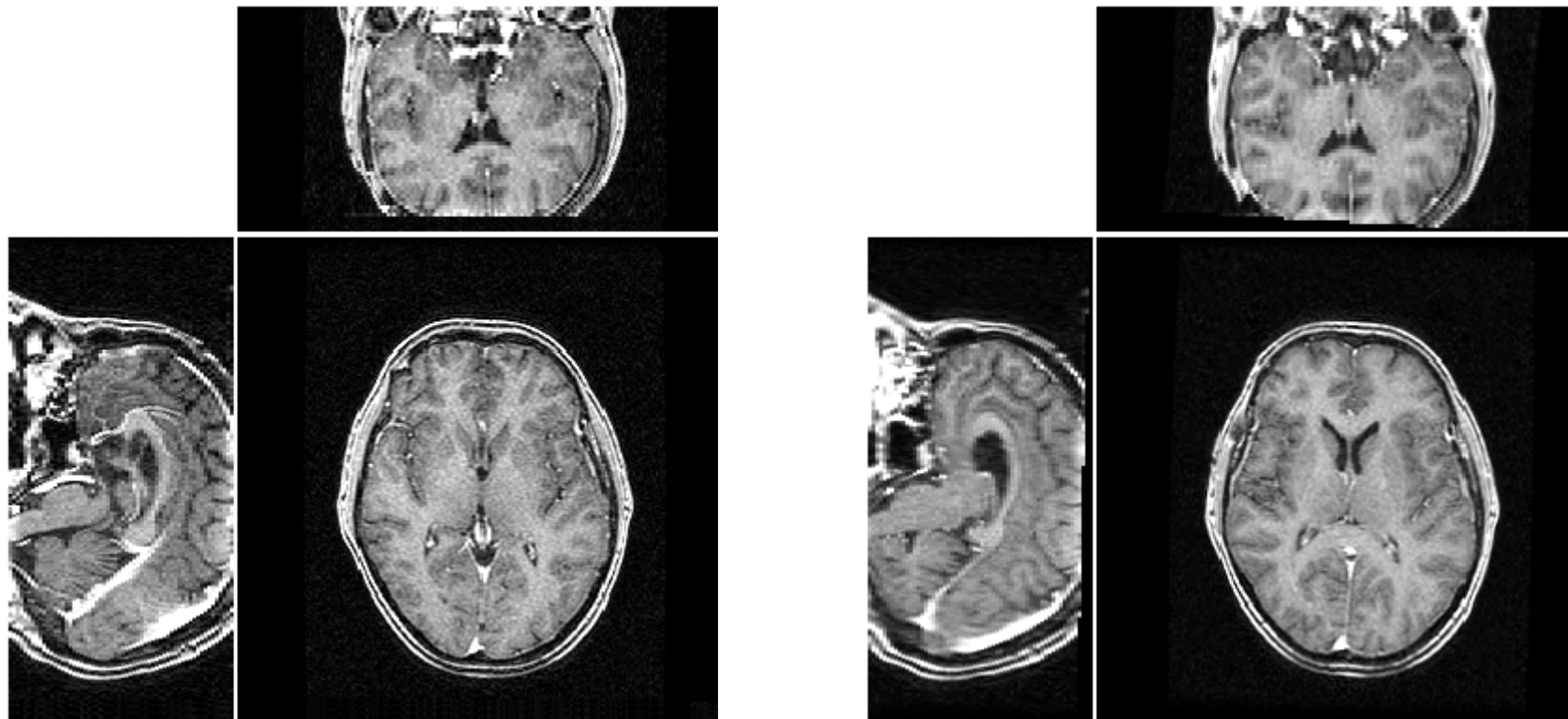
May 4, 2:30pm update

B-plan should work:

- pre-install glite 1.5 DMS on prod.
- Use production infrastructure for the demo

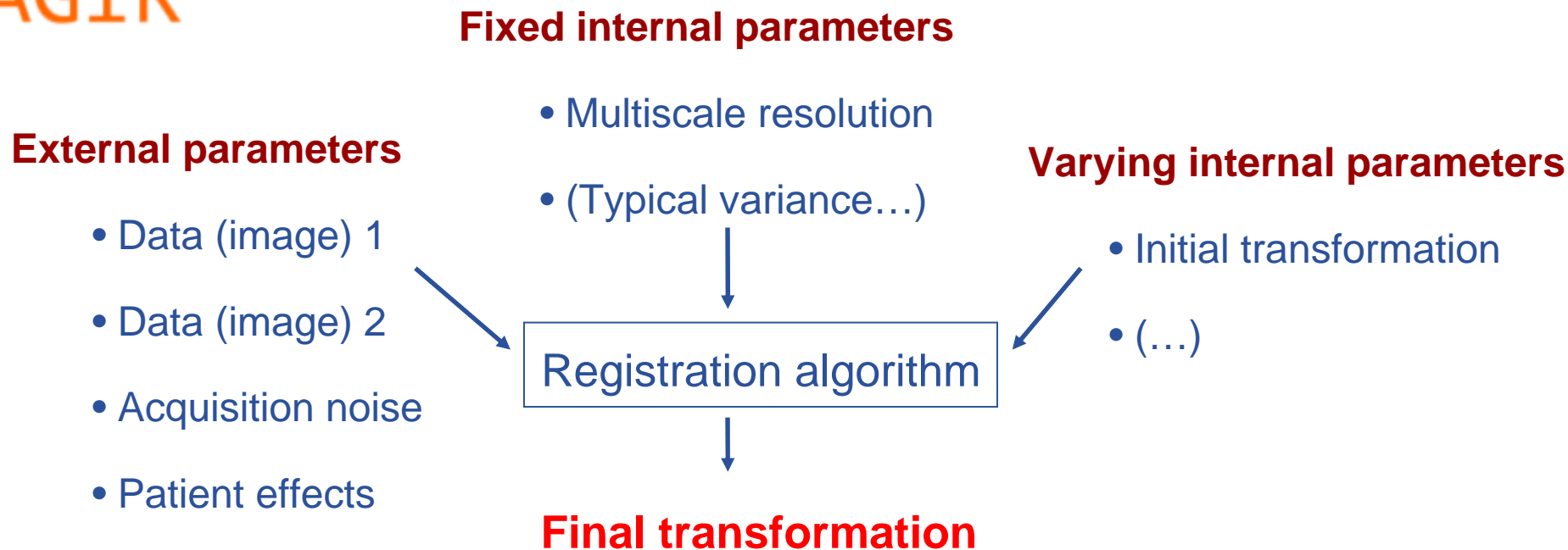
glite 1.5 phased out

AGIR



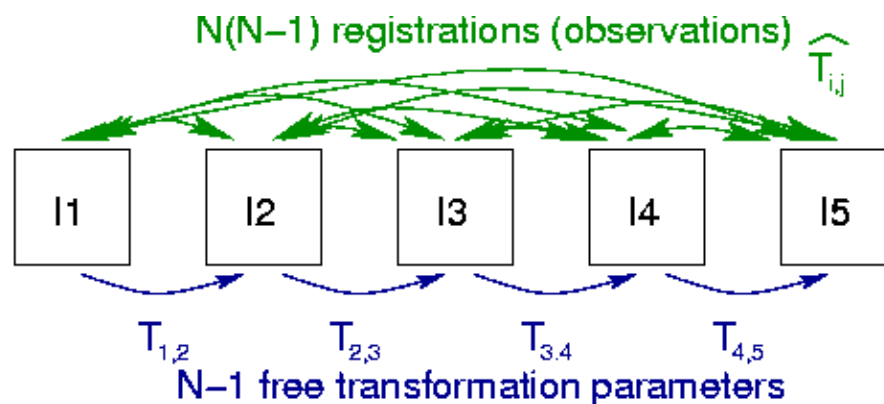
Before registration

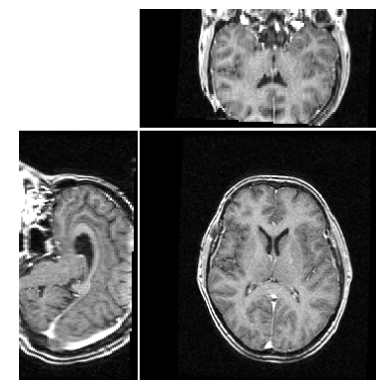
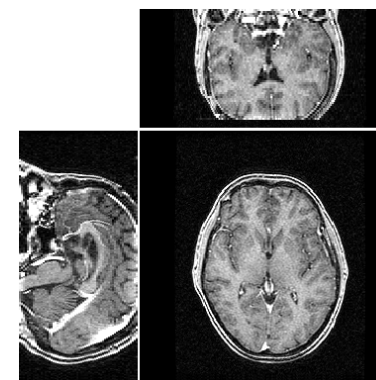
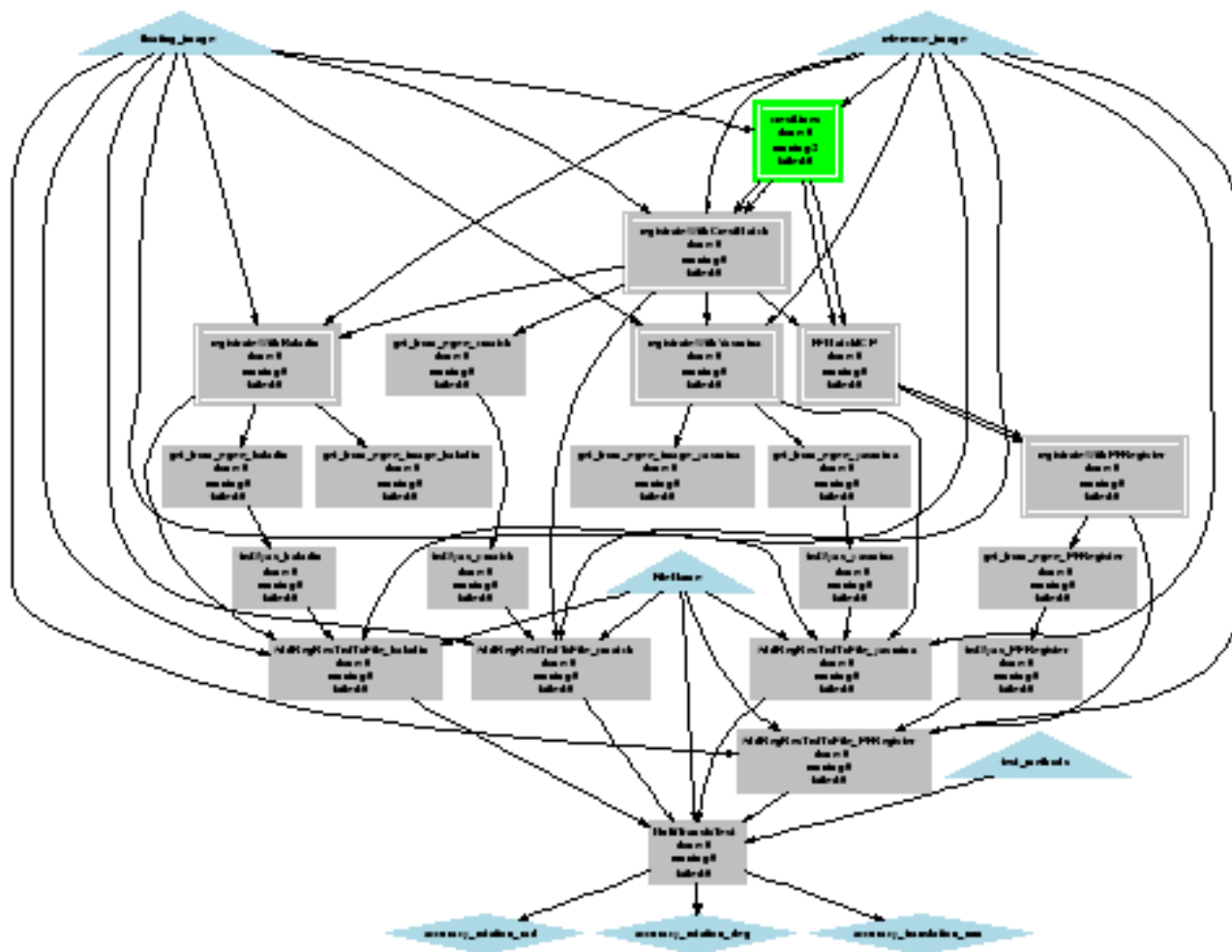
After registration



- **Robustness:** ability to find the right transformation (success/failure)
- **Repeatability:** w.r.t. some parameters (e.g. initialization)
- **Accuracy:** Variability w.r.t. the ground truth for typical data

- **Bronze standard:** The exact result is an unknown variable
- **Unbiased estimation: use redundant information**
 - use **many** different registration algorithms (average biases, so that precision \sim accuracy)
 - Use **many** different data (redundant information to ensure precision)
 - Average transformations (maximal consistency)

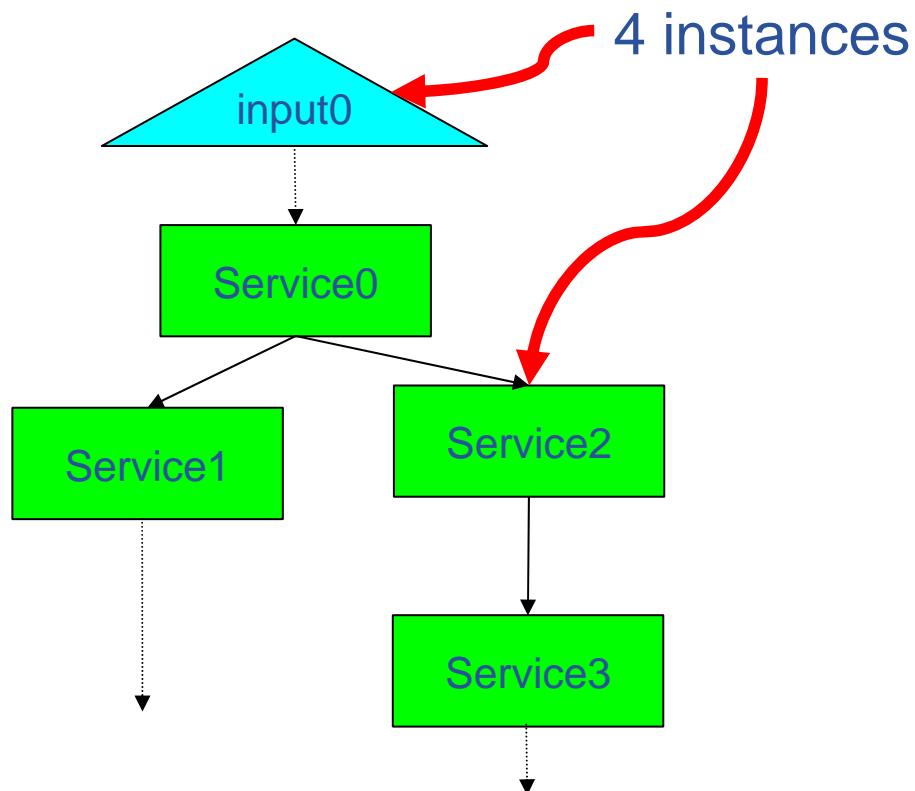




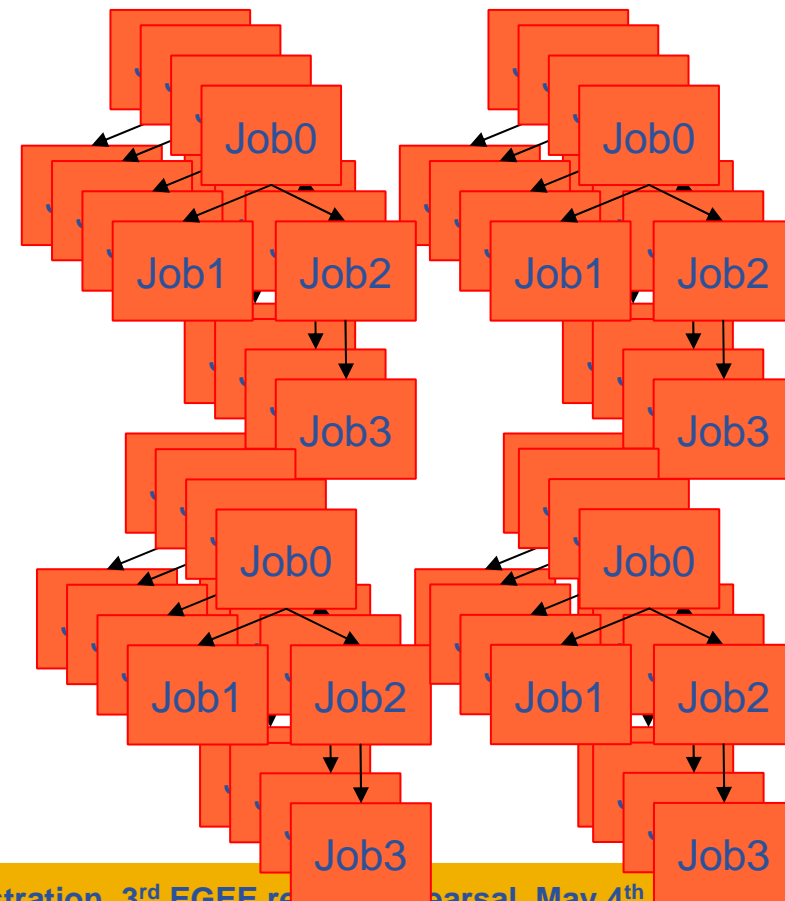
$$T_{i,j}, \sigma_{rot}, \sigma_{trans}$$

- Service-based approach versus task-based approach

Graph of services

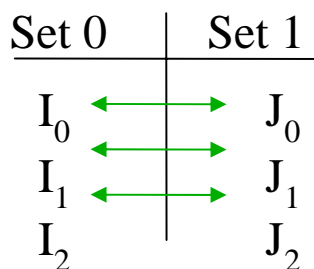


DAG of tasks

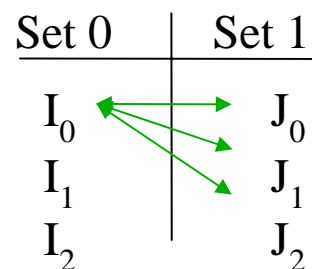


- **Data composition patterns : data intensive applications**

- One-to-one

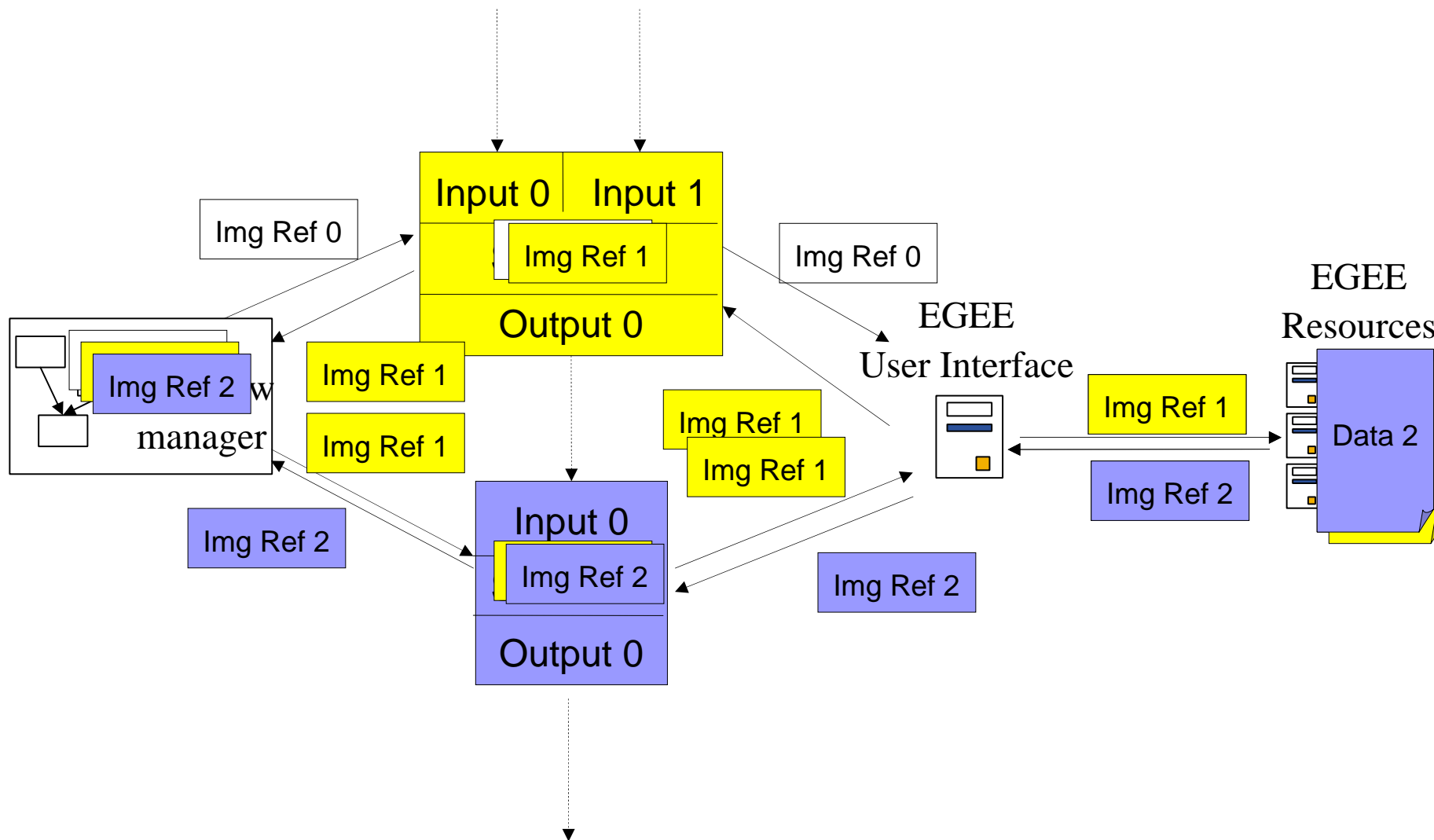


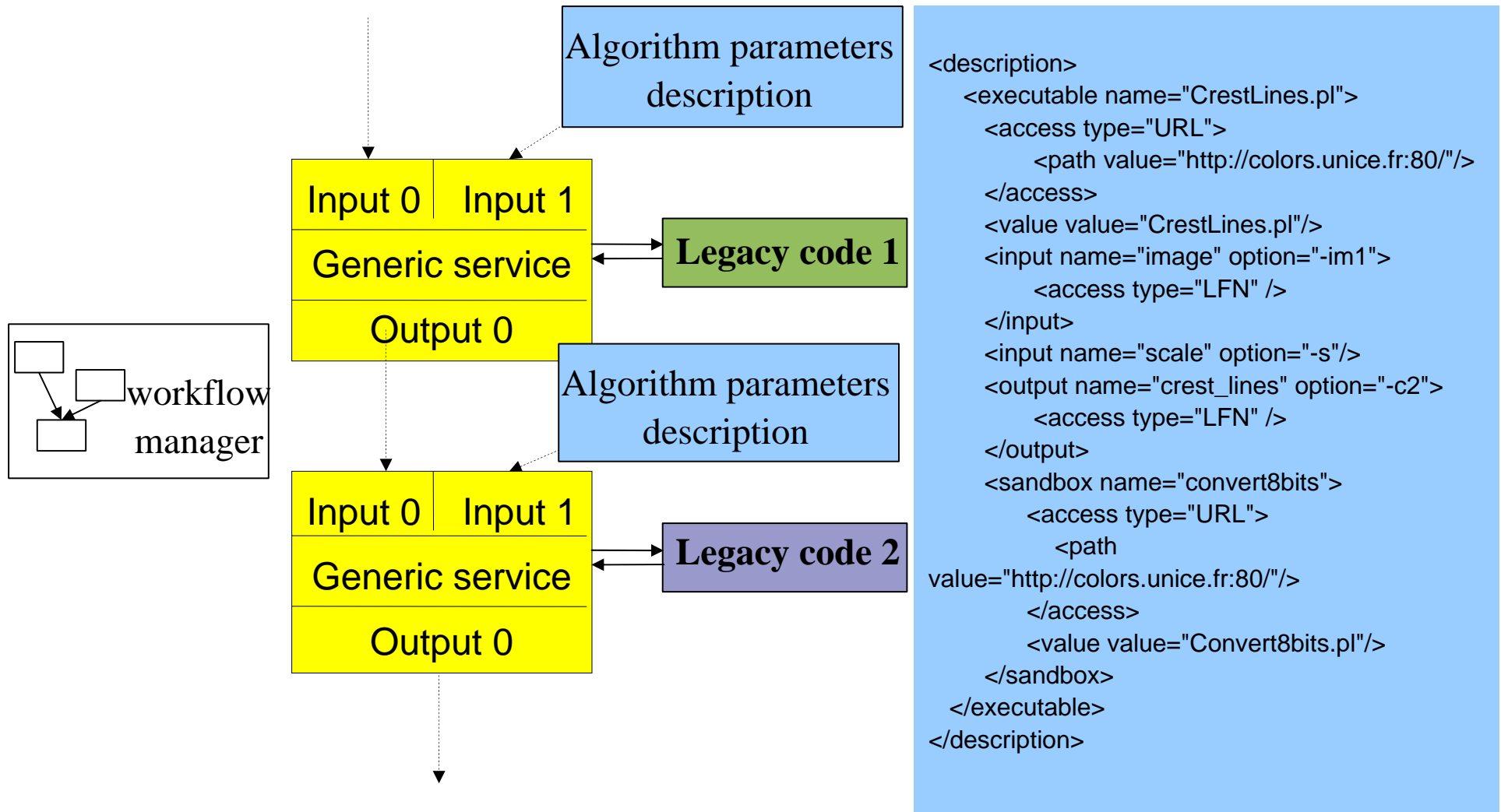
- All-to-all



- In our case: register all images of

- the same patient
- the same modality
- A different exam date

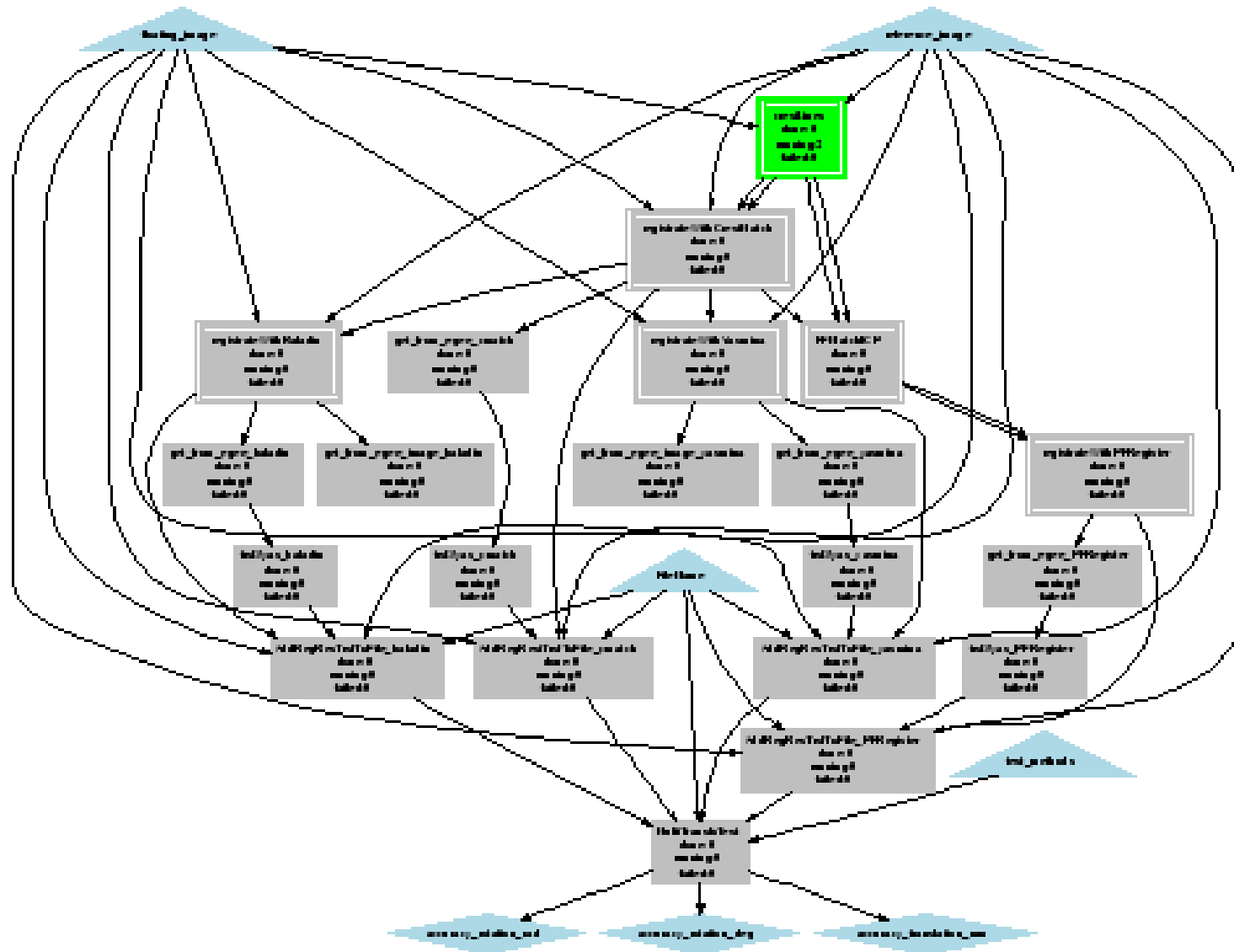


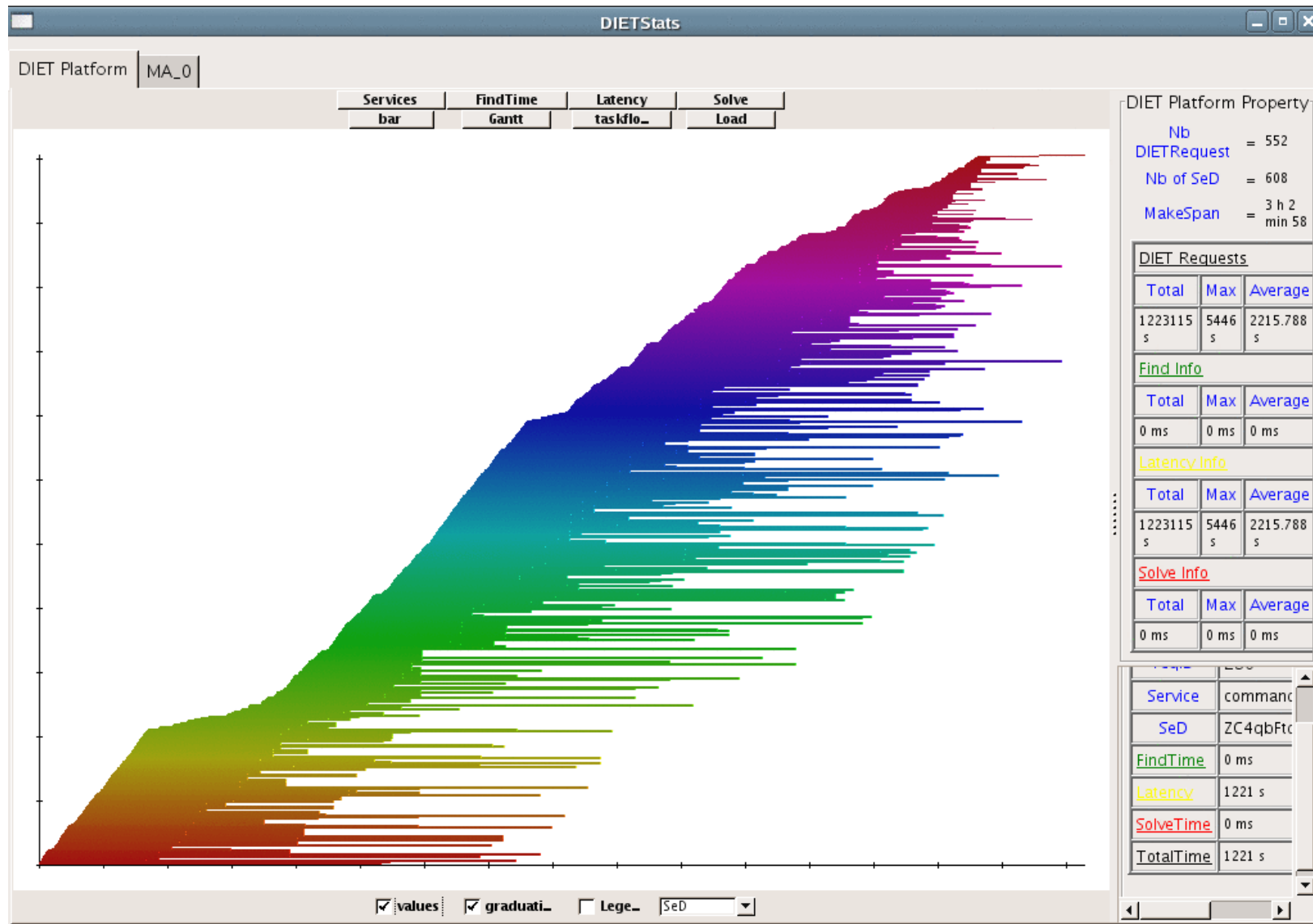


```

<description>
  <executable name="CrestLines.pl">
    <access type="URL">
      <path value="http://colors.unice.fr:80/">
    </access>
    <value value="CrestLines.pl"/>
    <input name="image" option="-im1">
      <access type="LFN" />
    </input>
    <input name="scale" option="-s"/>
    <output name="crest_lines" option="-c2">
      <access type="LFN" />
    </output>
    <sandbox name="convert8bits">
      <access type="URL">
        <path
value="http://colors.unice.fr:80/">
      </access>
      <value value="Convert8bits.pl"/>
    </sandbox>
    </executable>
  </description>
  
```


- **Torque + MAUI specific configuration**
 - Virtual processors allocation
 - Does not interfere with normal batch scheduling (shared processor time)
 - Enables efficient processing of short tasks on the production infrastructure
 - Ersatz for lack of jobs prioritization
- **Special submission queues**
 - Three SDJ queues deployed on biomed-compliant sites
 - Time-limited queues
- **Submit-or-reject paradigm**
 - Jobs are immediately executed or rejected if a too high number of short jobs are already executing.





- 4 rigid-registration algorithms precision estimated on brain image database

Algorithm	σ_{reg} (deg)	σ_{trans} (mm)
<i>CrestMatch</i>	0.150	0.424
<i>PFRegister</i>	0.180	0.416
<i>Baladin</i>	0.139	0.395
<i>Yasmina</i>	0.137	0.445

- To be published in [HealthGrid'06]

- **From days to hours**
 - 10s to 100s of algorithms
 - To adapt to many clinical cases
 - Virtually illimited parameterization
 - Virtually illimited number of image databases
 - Different modalities, different body regions
- **Complex computation procedure**
 - Difficult experimental set up
 - Future plan: application portal
- **Data federation**
 - Obtain data sources needed for validation
- **Algorithms sharing**
 - Use registration services developed in different research groups
 - Reproducible results

- **Medical data management**
 - Advanced Data management functionalities
 - Application area-level layer on top of foundation middleware
 - **Dependent on the deployment of gLite 1.5 services**
- **Bronze Standard application**
 - Complex, workflow-based application
 - Data intensive
 - Non-trivial parallel computations
 - Data federation using grid data management services
 - Production of scientific results
- **Short deadline jobs**
 - Immediate scheduling of short tasks
 - Submit-or-reject paradigm