



Workflow systems and VO standards

André Schaaff, Observatoire de Strasbourg
schaaff@astro.u-strasbg.fr

Topics

- **Quick introduction to workflows and workflow systems**
- **VO standards and workflows with a focus on the use of the IVOA Characterization standard**

✓ Workflows and workflow systems

Introduction to workflows / workflow systems

■ A workflow ?

■ Different tools executed by hand under shell

- Not very efficient but could be a solution if the tools are interactive

■ Script file

- Same as previous but a first step to the formalization
 - ▶ easily reusable, ...
- Involved tools are mainly black boxes (or converted to) with just I/O
- ...

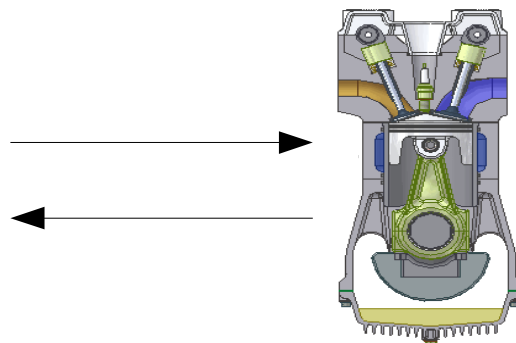
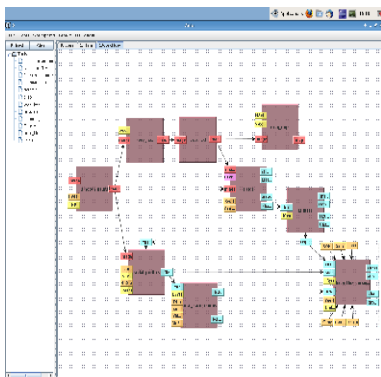
■ Description language

- The workflow is described in a specific language (often in XML format)
 - ▶ not executed like a shell script but interpreted by a workflow engine

Introduction to workflows / workflow systems (2)

■ “Sophisticated” workflow system

- Graphical design tool
- Workflow description (XML, ...) is sent to an engine who executes the workflow by dispatching the tasks
- Execution is often visible step by step
- Possible storage of intermediate data to change some parameters without the re execution of the whole workflow
- Result(s) can be exploited through tools related to the kind of output data (FITS, ...)



Tasks

Local cluster
User desktop
Storage (VOSpace, ...)
Local or remote DB
...

Workflow jungle

■ Workflow languages

- AGWL, BPEL4WS, BPML, DGL, DPML, GJobDL, GSFL, GFDL, GWorkflowDL, MoML, SWFL, WSCL, WSCI, WSFL, XLANG, YAWL, SCUFL/XScufl, WPD, PIF, PSL, OWL-S, xWFL, ...

■ " language formalisms

- Petri net, UML activity diagram, BPMN, DAG, IPO, GPSG, Workflow Patterns, Pi Calculus, Finite-State Machine, Gamma-calculus, ...

■ " engines

- BioPipe, BizTalk, BPWS4J, DAGMan, GridAnt, Grid Job Handler, GRMS, GWFE, GWES, IT Innovation Enactment Engine, JIGSA, JOpera, Kepler, Karajan, Moteur, OSWorkflow, Pegasus (uses DAGMan), Platform Process Manager, ScyFLOW, SDSC Matrix, SHOP2, Taverna, Triana, wftk, YAWL Engine, WebAndFlo, WFEE, ...

■ " composition/designing tools

- ilog's BPMN Modeller, CAT, GWUI, XBay GUI for Workflow Composition, Taverna, Triana, JOpera, Platform Process Manager, ...

■ " mapping from abstract to concrete workflows

- CWG, ACWG, Grid Job Handler, GWES, ...

Focus

■ Taverna, Scufi/XScufi

- Developed in the frame of myGrid which is a UK e-Science project
- See Kevin Benson (Taverna in the VO) and Richard Hook (ESO Reflex : A Graphical Workflow Engine for Data Reduction) talks during the Friday morning tools session

■ <http://taverna.sourceforge.net/>

Why workflows ?

- **Capture and preservation of scientific methodology (formalization of the scientific analysis)**
 - **Tools (different algorithms) to use, data flow, execution details, ...**
- **Management of computation at a large-scale and to mask the complexity**
 - **Iterations of tasks involving an access to clusters, grids, large DBs,**
- **A workflow system can provide a collaborative environment for the analysis/design/execution/validation of new use cases**
- ...

Workflows in the VO

- **An increasing number of services are developed/deployed in the frame of the Virtual Observatory (registries, data services, Web Services, computing and Grid services, ...) : complex use and coordination of the services are possible through workflows**

Workflow use cases in astronomy...

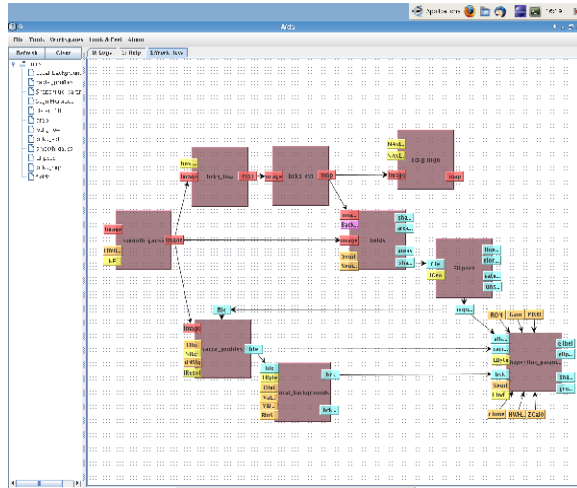
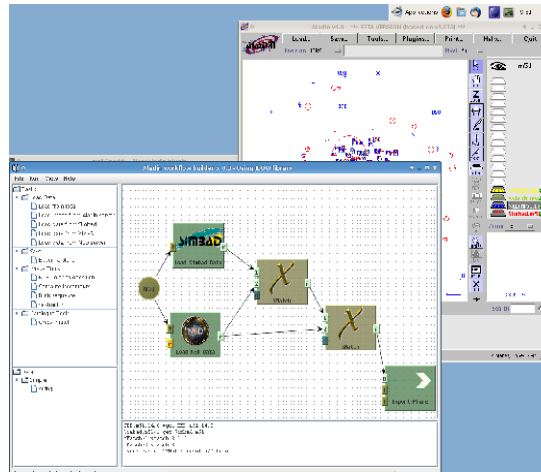
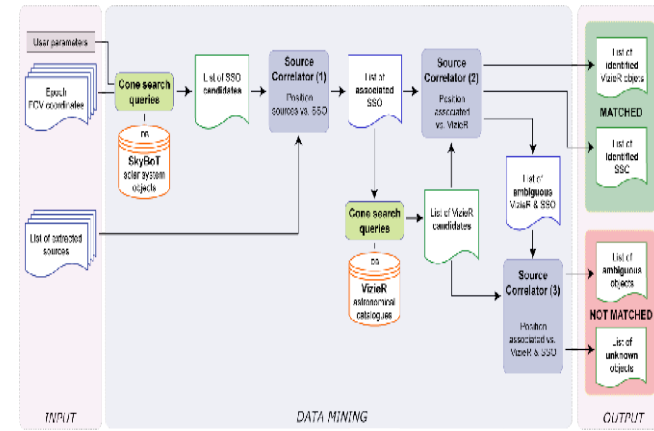


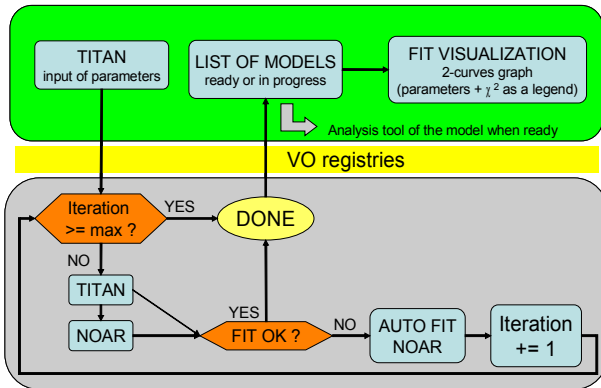
Image processing, E. Slezak.



Aladin scripting, C. Pestel, T. Boch.



Data Mining, J. Berthier et al.



TITAN/NOAR, L. Chevallier.

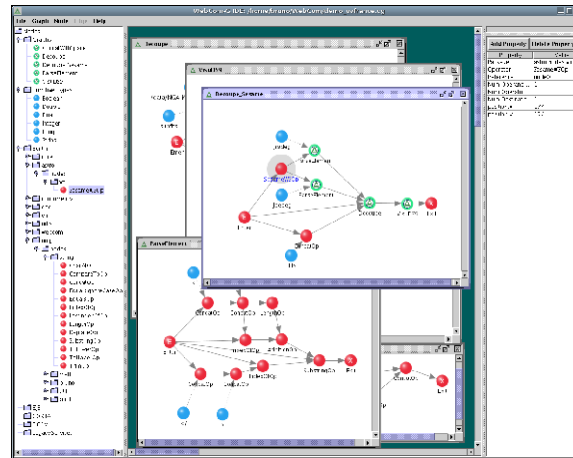
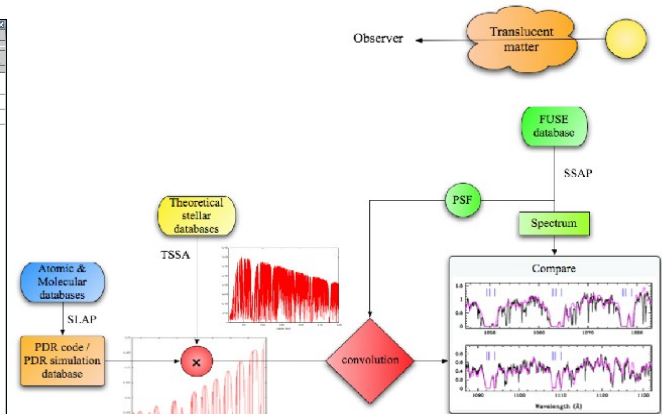


Image extraction from a catalogue, B. Voisin.



Simulation, F. Le Petit et al.

✓ Workflow systems and (some) VO standards

Workflows and VO standards

■ Registry

- Adaptive workflows with a choose of tools depending on parameters like the availability (see VOSI), ...

■ VOspace

- Storage of intermediate (deleted after each execution or temporary conserved to replay partially the workflow, ...) or final data produced during the workflow execution, ...

■ UWS

- Use of asynchronous VO services in a workflow, ...

■ ...

Common problems in workflows

- Applications called in workflows are often developed by different persons, with different languages, on different systems, ...
 - No unified error management, job failure, etc.
- ...
- A workflow can involve computing resources like clusters, grids and access to databases
 - For a 6 steps workflow if the step 3 requires a few hours of computing and the step 4 crashes due to a bad entry value, the workflow will probably end...
 - How to reduce this investment in CPU and user time ?

■ ...

IVOA Characterization

■ From the last reference document

- *This document defines the high level metadata necessary to describe the physical parameter space of observed or simulated astronomical data sets, such as 2D-images, data cubes, X-ray event lists, IFU data, etc...The Characterisation data model is an abstraction which can be used to derive a structured description of any relevant data and thus to facilitate its discovery and scientific interpretation. The model aims at facilitating the manipulation of heterogeneous data in any VO framework or portal.*

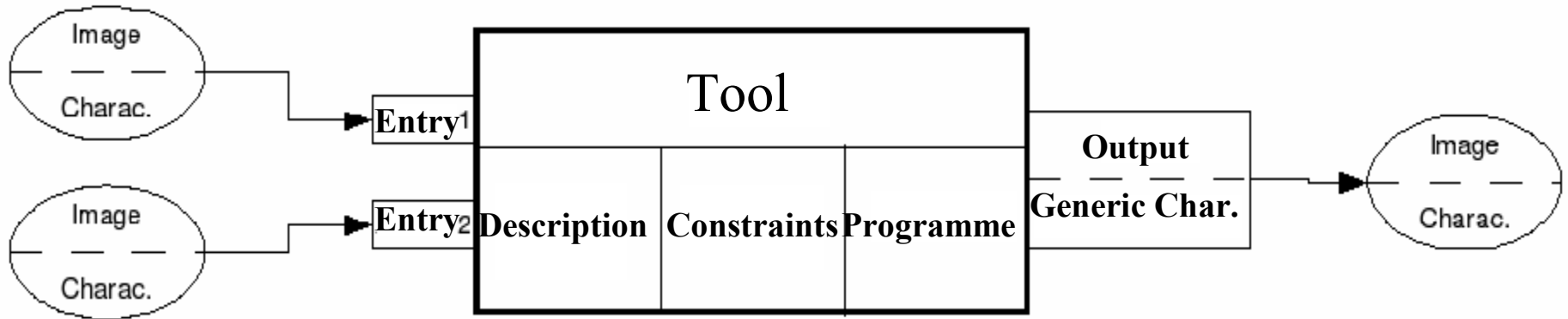
Characterization of the entries ?

- **Goal : validation of a workflow before its execution in the case of FITS images**
 - Try to do it for tools with FITS files as entries

- **Why ?**
 - Validation is done on the client side before the submission to the engine
 - Minimize the use of the external resources if validation fails
 - Optimization of the user time
 - ...

- **It requires**
 - A characterization file for the entries
 - A good knowledge of the tools to define the constraints

At the tool level



■ Before the execution

- Constraints on entries are defined for each tool
- A validation step checks the entries

■ During the execution

- After the step i , a characterization file is generated for the outputs and checked with the step $i+1$ constraints before its execution

Workflow test bed

Astronomical Image processing Distribution Architecture

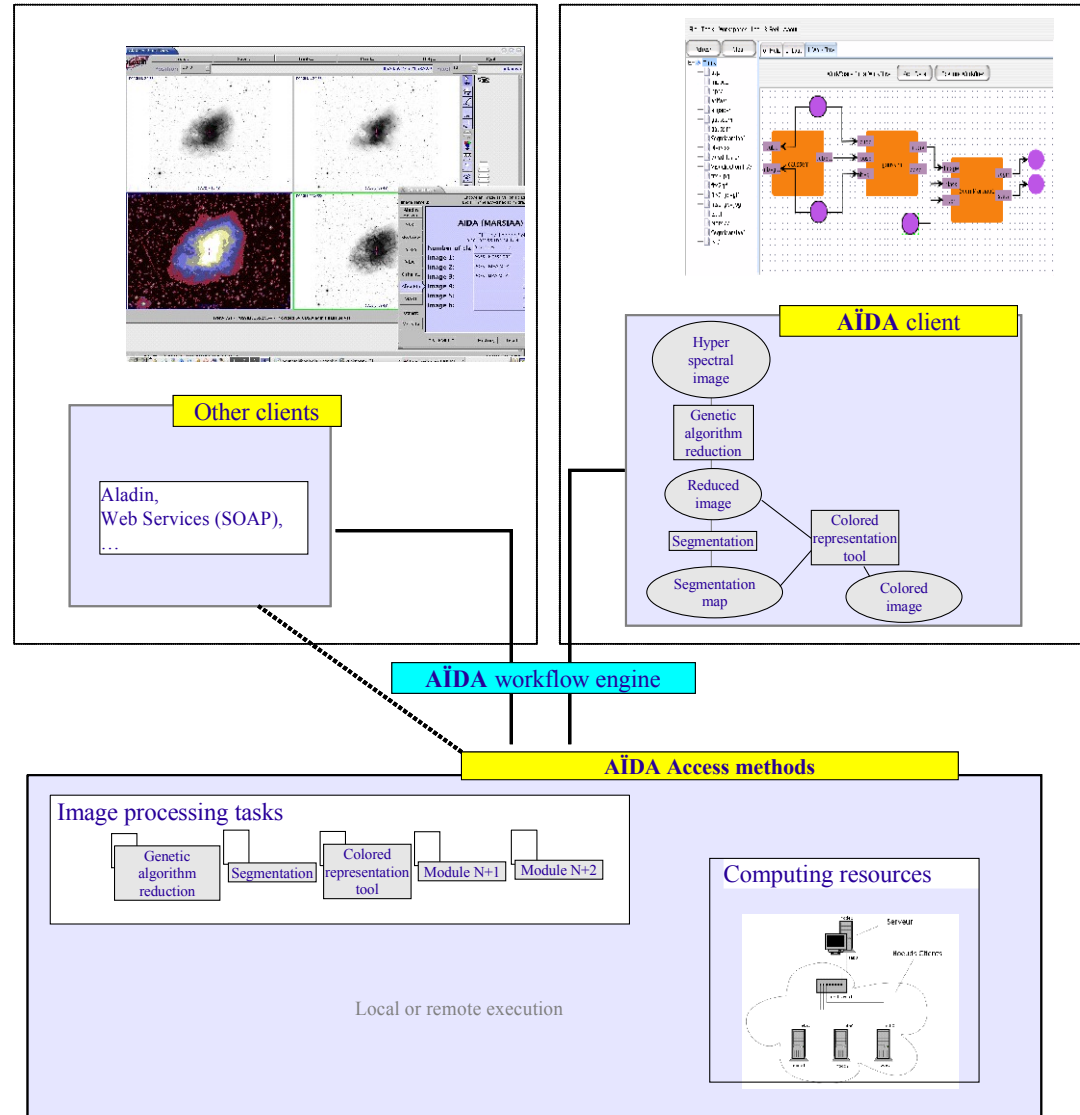
Contributors

- O. Benjelloun, characterization integration
- J. Beugnot*, packaging
- F. Bonnarel, architecture
- J.-J. Claudon*, core development
- B. Gassmann, characterization & Camea
- M. Louys, architecture
- G. Mantelet*, characterization integration
- C. Pestel, JLOW - design capabilities, new developments
- A. Schaaff, architecture

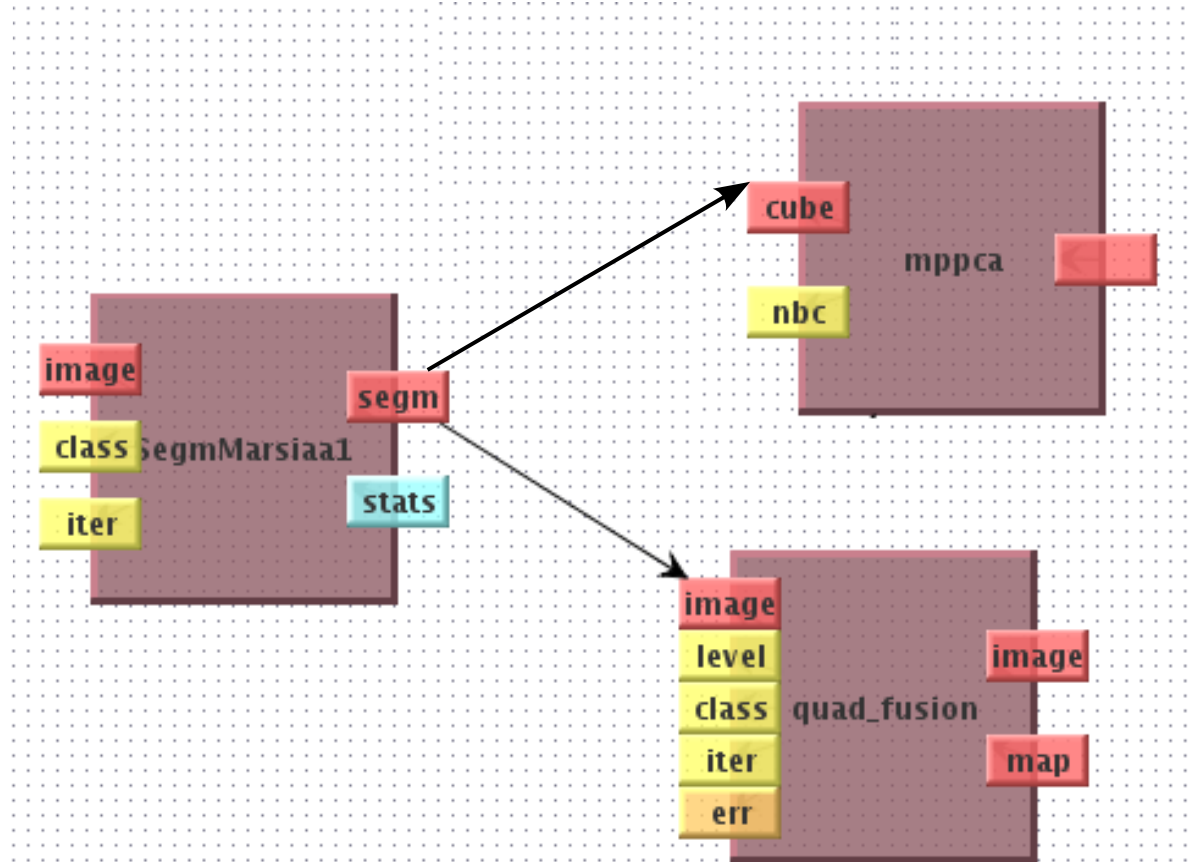
CDS & LSIIT

(* have left)

Work done in the frame of the French « **Massive Data in Astronomy** » project (2003-2006), **OV France** and **VOTECH**

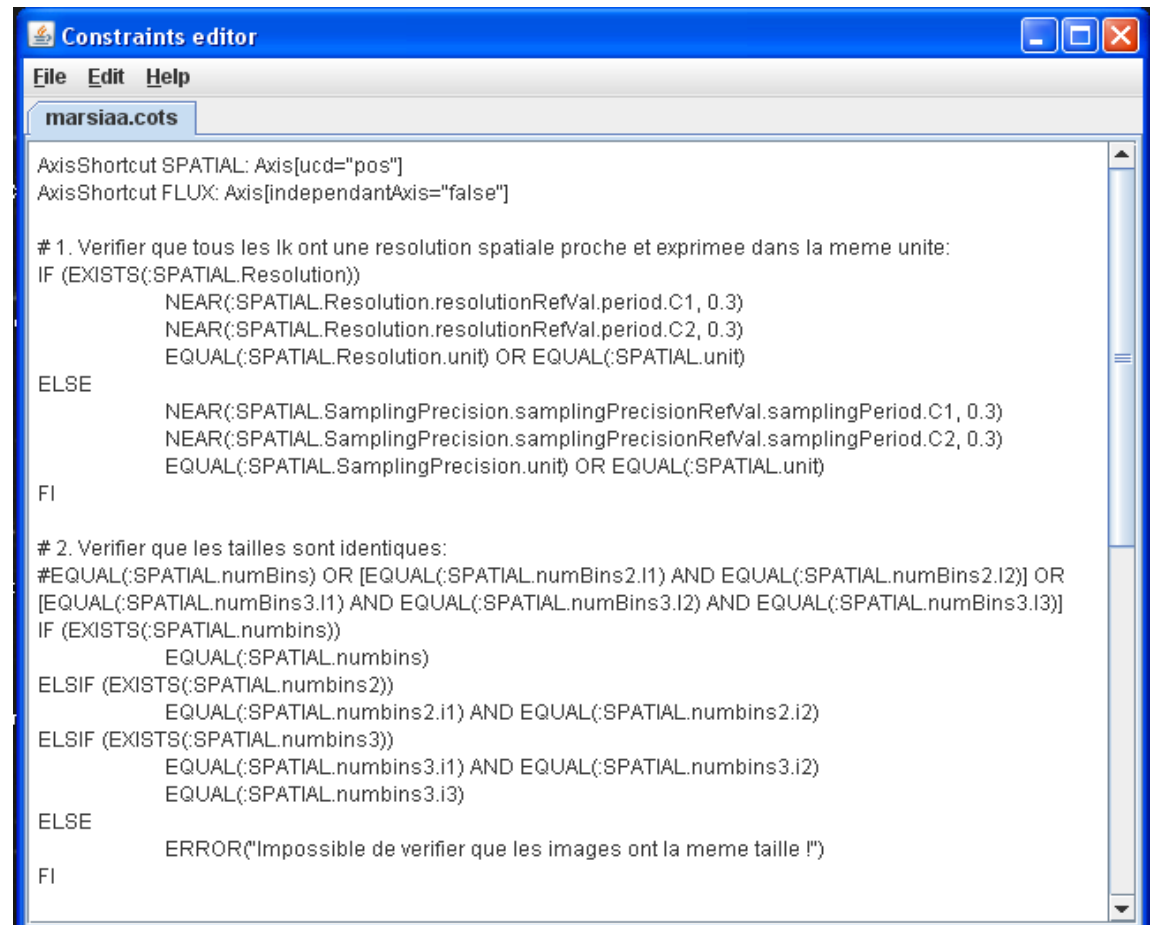


First step : a use case



Second step : write the constraints for each tool

- We have added a simple constraints editor to AIDA



```
Constraints editor
File Edit Help
marsiaa.cots
AxisShortcut SPATIAL: Axis[ucd="pos"]
AxisShortcut FLUX: Axis[independantAxis="false"]

# 1. Verifier que tous les lk ont une resolution spatiale proche et exprimee dans la meme unite:
IF (EXISTS(:SPATIAL.Resolution))
    NEAR(:SPATIAL.Resolution.resolutionRefVal.period.C1, 0.3)
    NEAR(:SPATIAL.Resolution.resolutionRefVal.period.C2, 0.3)
    EQUAL(:SPATIAL.Resolution.unit) OR EQUAL(:SPATIAL.unit)
ELSE
    NEAR(:SPATIAL.SamplingPrecision.samplingPrecisionRefVal.samplingPeriod.C1, 0.3)
    NEAR(:SPATIAL.SamplingPrecision.samplingPrecisionRefVal.samplingPeriod.C2, 0.3)
    EQUAL(:SPATIAL.SamplingPrecision.unit) OR EQUAL(:SPATIAL.unit)
FI

# 2. Verifier que les tailles sont identiques:
#EQUAL(:SPATIAL.numBins) OR [EQUAL(:SPATIAL.numBins2.i1) AND EQUAL(:SPATIAL.numBins2.i2)] OR
[EQUAL(:SPATIAL.numBins3.i1) AND EQUAL(:SPATIAL.numBins3.i2) AND EQUAL(:SPATIAL.numBins3.i3)]
IF (EXISTS(:SPATIAL.numbins))
    EQUAL(:SPATIAL.numbins)
ELSIF (EXISTS(:SPATIAL.numbins2))
    EQUAL(:SPATIAL.numbins2.i1) AND EQUAL(:SPATIAL.numbins2.i2)
ELSIF (EXISTS(:SPATIAL.numbins3))
    EQUAL(:SPATIAL.numbins3.i1) AND EQUAL(:SPATIAL.numbins3.i2)
    EQUAL(:SPATIAL.numbins3.i3)
ELSE
    ERROR("Impossible de verifier que les images ont la meme taille !")
FI
```

Definition of the constraints

- We have defined the grammar to generate the constraints parser
 - Very close to our needs (and to Characterization)
 - It should be replaced by a more standardized one

```
AxisShortcut SPATIAL: Axis[ucd="pos"]  
AxisShortcut FLUX: Axis[independantAxis="false"]
```

```
# 1. Verify that all the Ik have a close spatial resolution and are expressed in the same unit  
IF (EXISTS(:SPATIAL.Resolution))
```

```
  NEAR(:SPATIAL.Resolution.resolutionRefVal.period.C1, 0.3)  
  NEAR(:SPATIAL.Resolution.resolutionRefVal.period.C2, 0.3)  
  EQUAL(:SPATIAL.Resolution.unit) OR EQUAL(:SPATIAL.unit)
```

```
ELSE  
  NEAR(:SPATIAL.SamplingPrecision.samplingPrecisionRefVal.samplingPeriod.C1, 0.3)  
  NEAR(:SPATIAL.SamplingPrecision.samplingPrecisionRefVal.samplingPeriod.C2, 0.3)  
  EQUAL(:SPATIAL.SamplingPrecision.unit) OR EQUAL(:SPATIAL.unit)  
FI
```

```
# 2. Verify if the sizes are identical
```

```
IF (EXISTS(:SPATIAL))  
  EQUAL(:SPATIAL.numbins)  
ELSIF (EXISTS(:SPATIAL.numbins2))  
  EQUAL(:SPATIAL.numbins2.i1) AND EQUAL(:SPATIAL.numbins2.i2)  
ELSIF (EXISTS(:SPATIAL.numbins3))  
  EQUAL(:SPATIAL.numbins3.i1) AND EQUAL(:SPATIAL.numbins3.i2)  
  EQUAL(:SPATIAL.numbins3.i3)
```

```
ELSE  
  ERROR("Impossible de vérifier que les images ont la même taille !")  
FI
```

```
# 3. Vérifier que toutes les images sont superposables
```

```
EQUAL(:SPATIAL.Coverage.location.unit) OR EQUAL(:SPATIAL.Coverage.unit) OR EQUAL(:SPATIAL.unit)  
EQUAL(:SPATIAL.Coverage.location.coord_system_id)
```

```
# 4. Observable : (min-max)>=100 else WARNING
```

```
EQUAL(1[:FLUX.coverage.bounds.unit) OR EQUAL(:SPATIAL.Coverage.unit) OR EQUAL(:SPATIAL.unit)  
IF (1[:FLUX.bounds.limitHi - 1[:FLUX.bounds.limitLo >= 100)  
  WARNING("(Observables: min-max <100) Il faut faire une normalisation en niveau de gris !")  
FI
```

```
# 5. ...
```

```
EQUAL(:FLUX.ucd)  
1[:FLUX.bounds.extent < 100  
.....
```

```
FI
```

Third step : validation report generation

- It reports, for each tool of the workflow, all the errors and warnings detected during the analysis of all the Characterization files

The screenshot displays a workflow editor interface. On the left, a tree view shows various tools and files, including 'T:Tools', 'mppca', 'fpcpa', 'acifast', 'acijader', 'gaussem', 'gausim', 'SegmMarsiaa1', 'Marsiaa', 'cuad fusion', 'Visualisation HSV', 'fts2jpg', 'fts2git', 'fts2rgb-gif', 'fts2rgb-jpg', 'tstb', 'Bools', 'w-analyse2K', 'w-analyse2K-2', 'gaussemim', and 'ragppca'. The main workspace shows a workflow diagram with nodes: 'image', 'class SegmMarsiaa1', 'iter', 'segm', 'state', 'cube', 'mppca', 'nbr', 'image', 'level', 'class quad fusion', and 'image'. A 'Validation report - (Thu Apr 10 13:00:01 CEST 2008)' window is open, showing a table of errors:

Description	Type	Ligne	Co o...
SegmMarsiaa1 (1 messages)			
1[]:Axis[IndependentAxis = 'false'].bounds.extent < 100	ERROR	40	1
1[]:Axis[IndependentAxis = 'false'].bounds.extent	ERROR	40	1
=> Can not validate this property because either one of his field specify into the path do not exist, or the validation (Property:validate(value)) has not been done			

- Last step : execute ? modify the entries with errors ?

AIDA client with validation capabilities

The screenshot displays the AIDA client interface. On the left, a 'Tools' panel lists various processing tools such as 'acp', 'mppca', 'ppca', 'acifast', 'acijader', 'gaussem', 'gauslm', 'SegmMarsiaa1', 'Marsiaa', 'quad_fusion', 'Visualisation HSV', 'fits2jpg', 'fits2gif', 'fits2rgb-gif', 'fits2rgb-jpg', 'tstbl', 'Bools', 'w-analyse2K', 'w-analyse2K-2', 'gaussem1m', and 'regppca'. The main workspace, titled '2:Workflow', contains a diagram with nodes: 'image', 'class segmMarsiaa1', 'iter', 'stats', 'segm', 'cube', 'mppca', 'nbc', and 'image'. A 'Constraints editor' window is open in the foreground, showing a script for validating the workflow. The script includes checks for spatial coordinates, superposability, and flux bounds. An error message is visible in the background: '[ERROR] (line: 40; column: 1) Operand not valid, m'.

```
File Edit Help
/home/cyril/Programmation/aida/aida-mantelet/marsiaa.cots
IF (EXISTS(:SPATIAL.numbins))
  EQUAL(:SPATIAL.numbins)
ELSIF (EXISTS(:SPATIAL.numbins2))
  EQUAL(:SPATIAL.numbins2.i1) AND EQUAL(:SPATIAL.numbins2.i2)
ELSIF (EXISTS(:SPATIAL.numbins3))
  EQUAL(:SPATIAL.numbins3.i1) AND EQUAL(:SPATIAL.numbins3.i2)
  EQUAL(:SPATIAL.numbins3.i3)
ELSE
  ERROR("Impossible de verifier que les images ont la meme taille !")
FI

# 3. Verifier que toutes les images sont superposables:
EQUAL(:SPATIAL.Coverage.location.unit) OR EQUAL(:SPATIAL.Coverage.unit) OR
EQUAL(:SPATIAL.unit)
EQUAL(:SPATIAL.Coverage.location.coord_system_id)

# 4. Observables: (min-max) >= 100 sinon WARNING:
EQUAL(1[:FLUX.coverage.bounds.unit) OR EQUAL(:SPATIAL.Coverage.unit) OR
EQUAL(:SPATIAL.unit)
IF (1[:FLUX.bounds.limitHi - 1[:FLUX.bounds.limitLo >= 100)
  WARNING("Observables: min-max <100) Il faut faire une normalisation en niveau
de gris !")
FI

# 5. ...
EQUAL(:FLUX.ucd)
1[:FLUX.bounds.extent < 100
```

Other example

The screenshot displays a workflow editor interface. On the left is a file explorer with a tree view containing various tool categories like 'Tools', 'Bools', and 'w-analyse2K'. The main workspace shows a workflow diagram with nodes: 'mppca' (top), 'segm' (middle), and 'level' (bottom right). Arrows indicate data flow from 'segm' to 'mppca' and from 'segm' to 'level'. Each node has associated labels: 'cube' and 'nbc' for 'mppca'; 'image', 'class segmMarsiaa1', and 'iter' for 'segm'; and 'image', 'level', and 'image' for 'level'. Below the diagram is a 'Valid' panel showing a tree of messages and error details, including an error message: '[ERROR] (line: 4; column: 1) Operand => Can not validate this prop'. An 'Edit' button is located below the 'Valid' panel. At the bottom of the main window, the status bar shows '813 : 323'. Overlaid on the bottom right is a 'Constraints editor' window with a menu bar (File, Edit, Help) and a text area containing the following code:

```
File Edit Help
/home/cyril/Programmation/aida/aida-mantelet/mppca.cots
/home/cyril/Programmation/aida/aida-mantelet/marsiaa.cots
AxisShortcut SPATIAL : Axis[ucd="pos"]
#2: Axis[ucd="em"].accuracy.statError.flavor = "statistically"
1: Axis[ucd="phot" AND independentaxis="false"].ucd = "phot"
:SPATIAL.numBins2.i1 > = :SPATIAL.numBins2.i1*2/:SPATIAL.numbins2.i1
STOP_AT_UNKNOWN(false)
#1:SPATIAL.numBins2.i2 = 2:SPATIAL.numbins2.i2
#0.0009 > 2: Axis[ucd="em"].accuracy.statError.errorRefval.error
"hour" = 1: Axis[ucd="time"].unit
IF (EXISTS(3))
    ERROR("L'entree 3 n'existe pas !")
ELIF (EXISTS(1,2))
    WARNING("L'entree 1,2 n'existe pas !")
    1,2:SPATIAL.numBins2.i1 = 1,2:SPATIAL.numbins2.i2
ELIF (EXISTS(1,1))
    WARNING("L'entree 1,1 n'existe pas !")
    1,1:SPATIAL.numbins2.i1 = 1,1:SPATIAL.numbins2.i2
ELSE
    WARNING("HELLO !")
    1:SPATIAL.numbins2.i1 = 1:SPATIAL.numbins2.i2
FI
NEAR(:SPATIAL.numbins2.i1, 0.75)
NOT EQUAL(:Axis[ucd="em"].accuracy.staterror.flavor)
[EQUAL(:SPATIAL.numBins2.i3) OR EQUAL(:SPATIAL.numBins2.i1)] AND
EQUAL(:SPATIAL.numBins2.i2)
```

The bottom of the constraints editor window shows the file path: '/home/cyril/Programmation/aida/aida-mantelet/mppca.cots opened' and the status '4, 59 READ_ONLY'.

Summary of this study

■ Done

- Definition of workflow use cases with Characterized image entries
- Definition of a constraint language and integration in AIDA
- Definition of constraint files for the use cases tools
- ...

■ Ongoing work

- Increase the validation scope
 - During the execution : Characterization file generation for the FITS outputs
 - Before the execution : study how to define a virtual Characterization file for an output before the execution...
- Replace the constraint language with a more standardized one

Abstract

- **After a quick introduction to workflow systems, the presentation will focus on the use of Characterization, an IVOA standard, in a workflow test bed architecture. The execution of a workflow may require substantial computing resources and this can take a significant amount of time. Our goal is to introduce a validation step in the workflow process before the "real" execution. This work is done in the frame of the VOTECH project.**