Provenance for MuseWISE

Ole Streicher

`ole@aip.de`

Paris, 2019-05-15

MUSE

- Multi Object Spectrographic Explorer (1st light 2014)
    - 24 Integral Field Units (Spectrographs)
    - Spectral range: 4600-9300 Angstrom, 1.25 A binning, 4000 bins
    - Spacial: 300 x 300 pixel 0.2/0.025 arcsec sampling (WFM/NFM)
    - sampling not uniform
- MuseWISE
    - Based on astroWISE, adopted forMuseWISE
    - Automated standard data reduction framework for MUSE GTO
    - Distributed; Python based, Oracle DB, object oriented
    - stores processing Provenance
    - Integration of
        - instrument specific ESO pipeline
        - post-processing Python scripts

Provenance subjects in MuseWISE

- Data files: generating process, time, input files
- Configuration files and values
- Calibration files and values
- QC/QA results, QuickView graphics
- Log files and debug flags
- ESO programs
- *some associations/attributions*

Provenance for MuseWISE

- Separate database (PostgreSQL)
- Relational model based on the draft (3 months ago)
- Web interface for ADQL/SQL queries
- ActivityDescriptions from MUSE metadata
- **No** ActivityConfiguration branch
- Configuration values and files as Entities, with provenance information
- Work in progress

MuseWISE configuration provenance

- Configuration: files and values
- Flexible (project specific) configuration files, not just "key=value"
- Integrated as `Entity` with specific `Usage` (type *Configuration*)
- Usage in specific roles, like "`astrometry_wcs`"
- Provenance directly attached to configuration:
    - Generation of configuration files
    - Attributions for values and configuration files
        - Values provided by pipeline developers or others as recommendations
        - Special configuration values+files provided by the GTO project
- Re-using configuration files and values in several activities

MuseWISE provenance volume

- covers MUSE deployment and GTO until 2017-01-01,
- complete range from observation to science-ready product
- currently about 150,000 activities,
- 860,000 entities, mostly files (9,000 `ValueEntities`)
- 4,500,000 usages, 820,000 generations

Example query

Find out (scipost) activities without sky subtraction

```
SELECT Activity.name, Activity.startTime
FROM Activity, Entity, Used
WHERE Used.role = 'skymethod'
  AND Used.entity = Entity.id
  AND Entity.content = 'none'
  AND Used.activity = Activity.id;
```

(not interoperable)

Example query - progenitors

Give me the main input of a selected activity

```
SELECT UsageDescription.role, Entity.name
FROM Entity, Used, UsageDescription
WHERE Used.entity = Entity.id
  AND Used.description = UsageDescription.id
  AND UsageDescription.type = 'Main'
  AND Used.activity = '299b1c65bcacdd8063d4feab1373
```

Example query - configuration

Give me the configuration of a selected activity

```
SELECT UsageDescription.role, Entity.content
FROM Entity, Used, UsageDescription
WHERE Used.entity = Entity.id
  AND Used.description = UsageDescription.id
  AND UsageDescription.type = 'Configuration'
  AND Used.activity = '299b1c65bcacdd8063d4feab1373a
```

(not interoperable)

MuseWISE ProvSAP

- W3C provenance compatible
- Middleware, using Python *prov* package
- Relational interface evolving into provTAP compatibility (general usage)
- *ActivityConfiguration* not implemented
- proposing slight modification of the behaviour:
    - by default, restrict to main data path
    - by default, include all used/generated entities
- Fully combined ProvTAP and W3C compatibility questionable

MuseWISE provenance visualization

- Prototype
- Based on ProvSAP/W3C provenance
- Integrated into the ProvSAP server
- HTML output, tabular, some basic javascipt interactivity
- Graphical output possible, but barely useful
- Generic (if the standard is well-defined): MuseWISE, HiPS, CTA

MuseWISE visualization example

Provenance information as activity list

This is the list of related activities, ordered by the start time. To show detailed information about an activity (description, input, output, responsibilities), click on the small triangle ▼ on the left side. Click the triangle again to hide the details.

▼	1	**OBJECT exposure at 2014-10-29T00:29:23**		2014-10-29 00:29:23 - 2014-10-29 00:29:53
▼	2	**OBJECT exposure at 2014-10-29T00:50:43**		2014-10-29 00:50:43 - 2014-10-29 00:51:13
▼	3	**OBJECT exposure at 2014-10-29T01:12:40**		2014-10-29 01:12:40 - 2014-10-29 01:13:10
▼	4	**scibasic_object**	*1.2*	2015-10-07 18:54:28 - 2015-10-07 18:57:47
▼	5	**scibasic_object**	*1.2*	2015-10-07 18:54:28 - 2015-10-07 19:00:24
▼	6	**scibasic_object**	*1.2*	2015-10-07 18:54:28 - 2015-10-07 18:59:12

Description

Version	1.2
Type	recipe
Description	Remove the instrumental signature from the data of each CCD and convert them from an image into a pixel table. ▼

Input

Configuration

nkeep	1 pixel	Number of pixels to keep with minmax
combine	none	Type of combination to use. ▼
skybinsize	0.1 Angstrom	Size of the bins (in Angstrom per pixel) for the intermediate spectrum to do the Gaussian fit to each sky emission line.

Other visualization examples

Click the triangle again to hide the details.

▼ 1	**cta_obs_47802**	*1*	2008-08-27 19:32:35 - 2008-08-27 20:00:43
▼ 2	**cta_obs_47803**	*1*	2008-08-27 20:03:05 - 2008-08-27 20:31:13
▼ 3	**gammapy_maps/4e00f5**	*1*	2019-01-22 11:26:45 - 2019-01-22 11:27:13

Description		
Version	1	
Type	analysis	
Description	Use gammapy to generate a count map from a list of observations	

Responsibility		
owner	opus_user:opus-admin	
author	Mathieu Servillat	

Input		

Configuration

RA	329.7169379 deg	Target Right Ascencion
Dec	-30.2255883 deg	Target Declination
nxpix	400	Number of pixels on the X axis
nypix	400	Number of pixels on the Y axis
binsz	0.02 deg	Size of a pixel in degrees

Main

Click the triangle again to hide the details.

▼ 1	**Generation DSS2 Blue HiPS**	*1*	2015-09-08 12:14:00 - 2015-09-08 12:14:00

Description		
Version	1	
Type	hipsgen	
Description	Genaration of HiPS using CDS hipsgen version 9.615	

Input		

Configuration

hips_order	9 none	HiPS helpix tesselation order
hips_tile_width	512 pix	HiPS tile width: generally 512, not always
hips_tile_format	jpeg fits	HiPS tile format: fits jpg or png
hips_frame	equatorial	HiPS coordinates frame: equatorial, galactic

Main

HiPSprogenitors	DSS2 Blue original	HiPS progenitors

Click the triangle again to hide the details.

▼ 1	**cubeseg_shell_12**	*1.0*	2019-02-26 09:13:00 - 2019-02-26 10:07:35

Description		
Version	1.0	
Type	image processing	
Description	Hyperspectral cube line fitting for velocity maps ▼	

Input		

Configuration

start position	(128, 128)	Start position in pixel number (nbline, nbcol)
linefittingStrategy	shell	Line fitting strategy. ▼

Main

input cube	CubeCalifa-NCG0036	input cube to be segmented

Output		

Main

Halpha amplitude map	Halpha_Amplitude_Map	Location of Halpha line with line amplitude
Halpha velocity map	Halpha_veloc_1	velocity map derived from line doppler

Tests of provenance goals

- **A**: Traceability of products - *partial*
    - No interoperable way to get the entity id for a specific dataset (file)
- **B**: Acknowledgement and contact information - ~~*partial*~~ *yes*
    - No citation information available
    - No standardized structure for citation or acknowledgement requirements
- **C**: Quality and Reliability assessment - *partial*
    - No standarized entity for warnings
- **D**: Identification of error location - *yes*
- **E**: Search in structured provenance data - *yes*

MUSE Provenance experience

- Full provenance works even w/o *ActivityConfiguration*
- One has to deal with
    - Incomplete information
    - Collaboration decisions about the publication of data
    - Combining information from several sources
    - Different interpretations of the (VO) Provenance standard
- ~~Not all goals defined in the standard are reached~~
- ~~Some requirements are really questionable~~
- The *used* table is by far the largest one
- Provenance database can be queried and visualized effectively
- A use case for a *prov:Plan* popped up, which is not solved yet

Provenance for ESO programs

- Each GTO observation is associated with an "ESO Program":
  - Proposal document with goals etc.
  - Target list and allocated time
  - List of collaborators
  - . . .
- This is
  - a *prov:Plan*: "A plan is an entity that represents a set of actions or steps intended by one or more agents to achieve some goals."
  - an organizational *prov:Agent* containing the collaborators workgroup
- We don't have a *prov:Plan* yet
- We don't have a dedicated prov:Role for this kind of agents yet

Provenance of files

- Files, tables, etc.: important entry points into Provenance
- Use case: I have a FITS file from MuseWISE. How do I find out its provenance?
  - **id**? The file does not come with its entity *id*
  - **url**? This is not uniq in MuseWISE (several data servers)
  - **name**? Would work, since we fill the file name here, and this is uniq in MuseWISE (but not f.e. at ESO).
- *Not an interoperable solution*