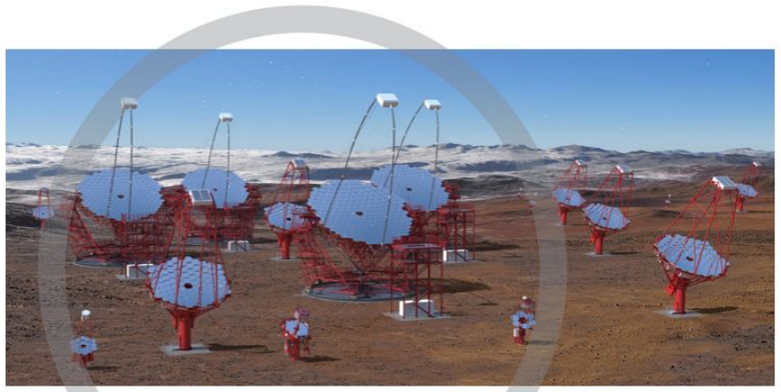
IVOA Provenance Data Model

Mathieu Servillat

Observatoire de Paris - LUTH

Paris Astronomical Data Centre

Objectives and context



Observatory

Consortium

Data acquisition
(triggers, cameras, …)

Data Centers
(data processing, reduction, reconstruction, simulations, …)

Archive, Dissemination

Web portal

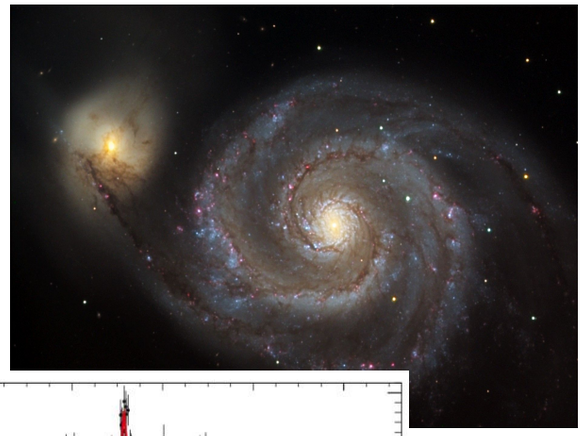
Additional data

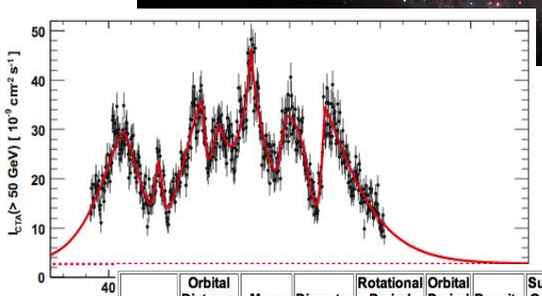
End User

Publications

- Data product generation **obscure** to end user
- **Quality**, **reliability**, **trustworthiness**?
- **Usefulness, pertinence** of the data?

Need **structured** and **detailed** provenance information

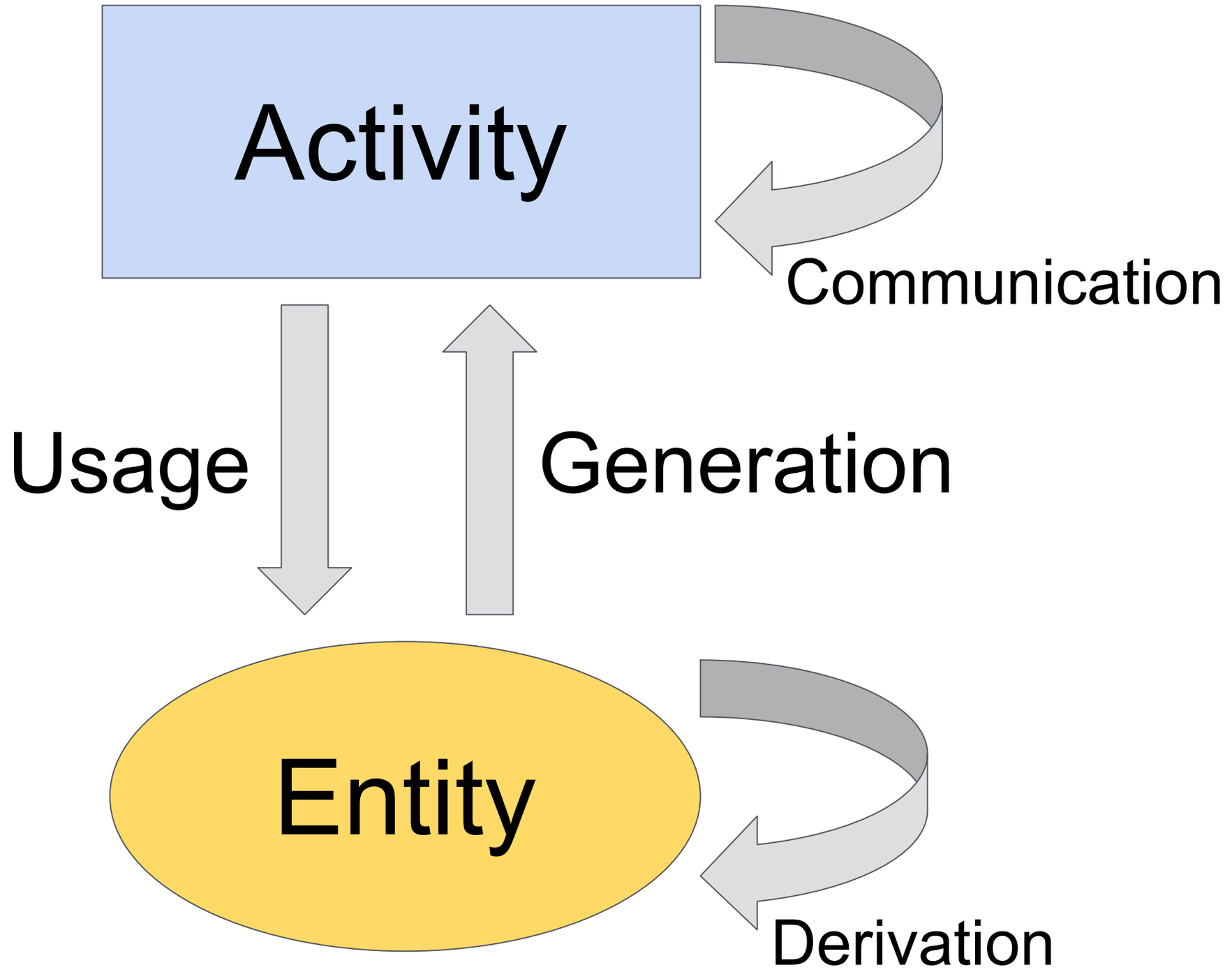


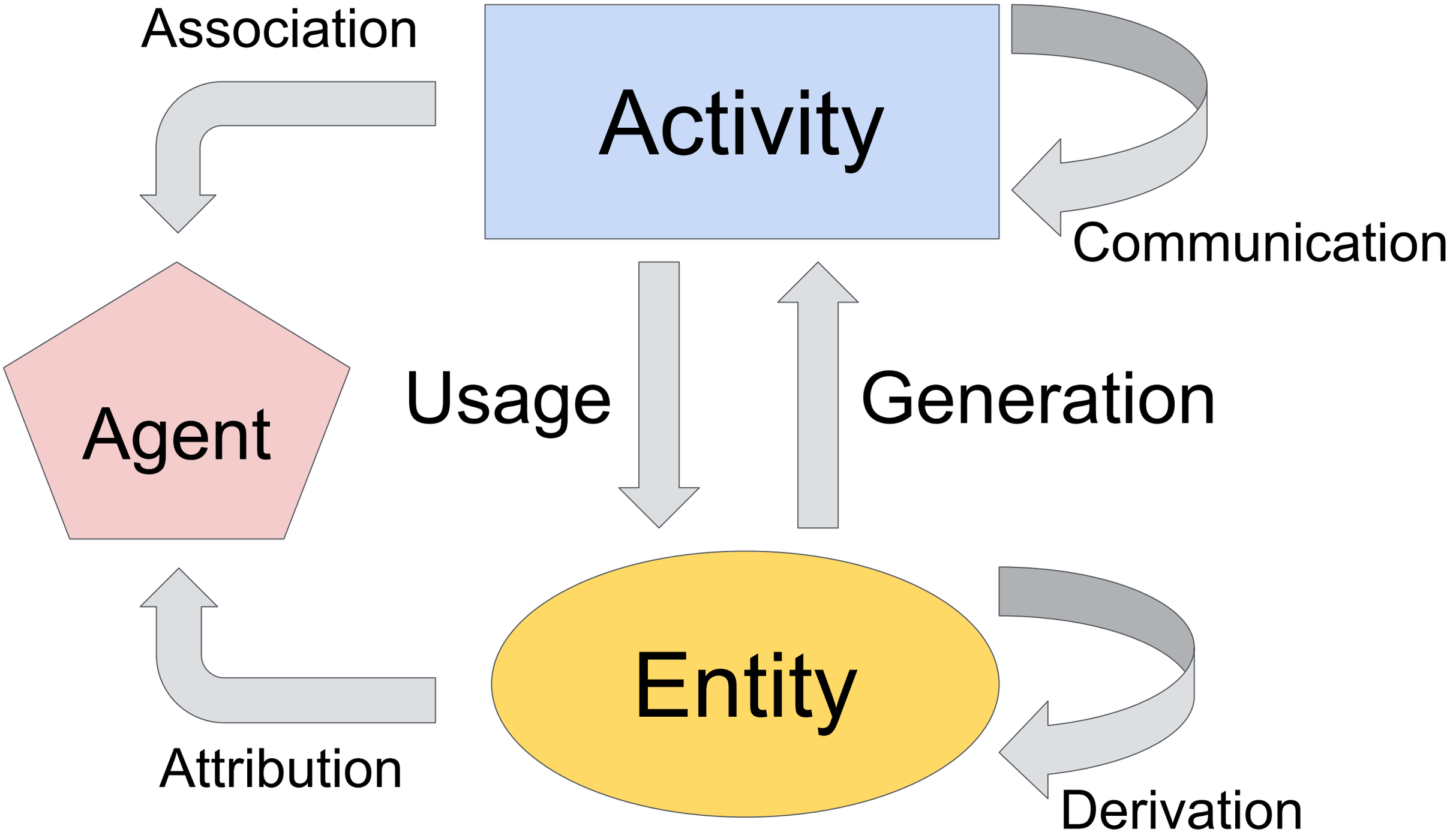


	Orbital Distance (AU)	Mass (earths)	Diameter (earths)	Rotational Period (days)	Orbital Period (years)	Density (earths)	Surface Gravity (earths)	Moons
Sol	0.0	330,000	109.2	25.4	...	1.42	28	...
Mercury	0.4	0.06	0.38	59	0.24	0.98	0.38	0
Venus	0.7	0.81	0.95	243	0.62	0.95	0.90	0
Earth	1.0	1.00	1.00	1.00	1.0	1.00	1.00	1
Mars	1.5	0.11	0.53	1.03	1.9	0.71	0.38	2
(Ceres*)	2.8	0.00015	0.07	0.38	4.6	0.38	0.03	0
Jupiter	5.2	317.8	11.2	0.42	11.9	0.24	2.34	63
Saturn	9.5	95.2	9.4	0.44	29.4	0.12	1.16	60
Uranus	19.2	14.5	4.0	0.72	83.7	0.23	1.15	27
Neptune	30.1	17.2	3.9	0.67	163.7	0.30	1.19	13
(Pluto*)	39.4	0.002	0.18	6.40	248.0	0.37	0.04	3
(Eris*)	67.7	0.002?	0.18	~8	557	?	?	1

Entity

Derivation


Activity
Communication
Usage
Generation
Entity
Derivation


Association
Activity
Communication
Agent
Usage
Generation
Attribution
Entity
Derivation

W3C Provenance definition

W3C PROV (PROV-DM, 2013)

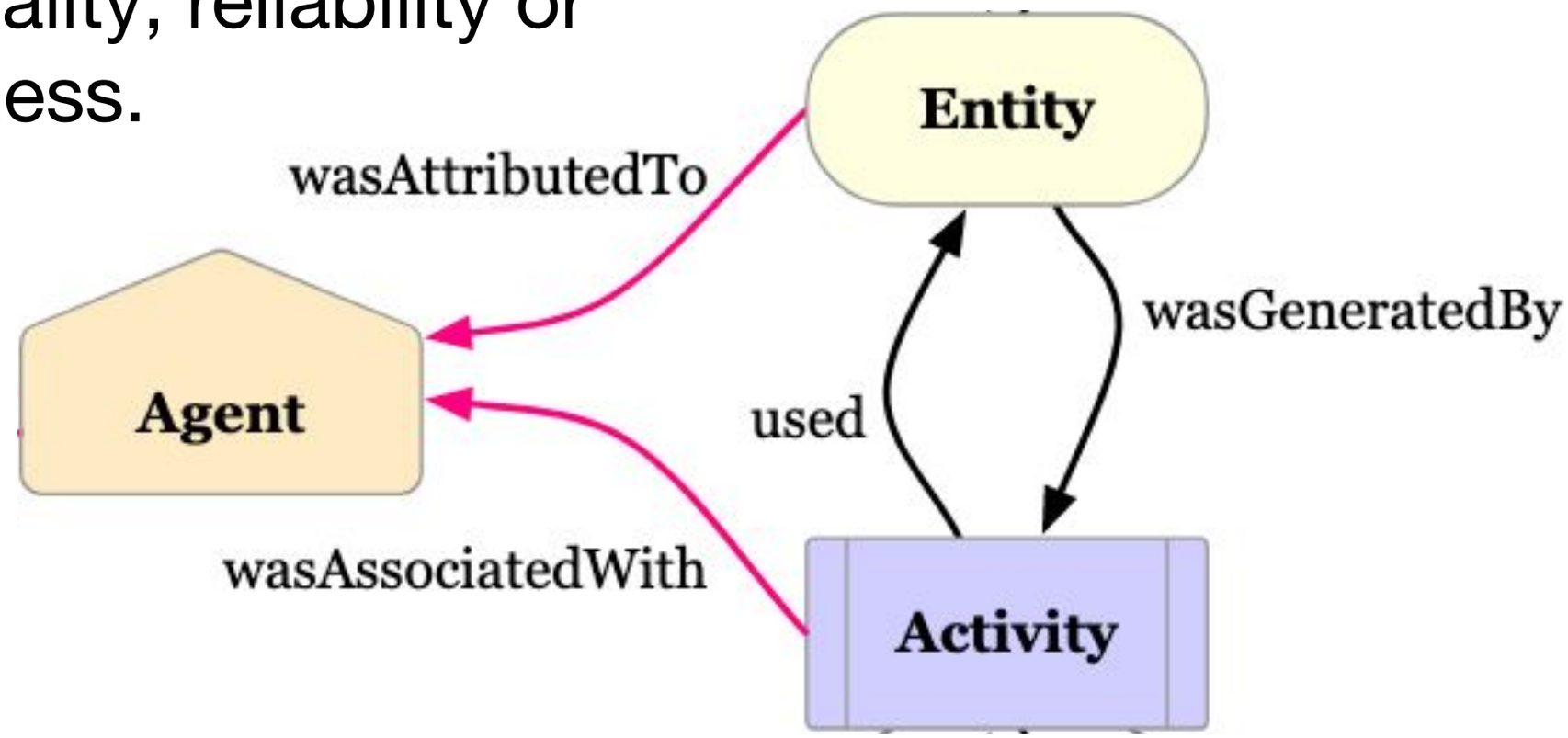
Provenance is information about entities, activities, and people (agents) involved in producing a piece of data or thing, which can be used to form assessments about its quality, reliability or trustworthiness.

http://www.w3.org/TR/prov-overview/

Word Wide Web Consortium

Existing Implementations:

- Java library
- Python package
- ProvStore web service



IVOA Provenance Data Model

Version 1.0

IVOA Proposed Recommendation 2019-07-19

Working group
DM

This version
http://www.ivoa.net/documents/ProvenanceDM/20190719

Latest version
http://www.ivoa.net/documents/ProvenanceDM

Previous versions
WD-ProvenanceDM-1.0-20190614.pdf
PR-ProvenanceDM-1.0-20181015.pdf
WD-ProvenanceDM-1.0-20180530.pdf
WD-ProvenanceDM-1.0-20170921.pdf
WD-ProvenanceDM-1.0-20161121.pdf
ProvDM-0.2-20160428.pdf
ProvDM-0.1-20141008.pdf

Author(s)
Mathieu Servillat, Kristin Riebe, Catherine Boisson, François Bonnarel, Anastasia Galkin, Mireille Louys, Markus Nullmeier, Nicolas Renault-Tinacci, Michèle Sanguillon, Ole Streicher

Editor(s)
Mathieu Servillat

Use cases from projects

- CTA (Cherenkov Telescope Array) data processing and access
- RAVE (Radial Velocity Experiment)
- POLLUX (synthetic stellar spectra service)
- CDS image databases
- SVOM gamma ray burst / transients
- APPLAUSE photographic plates database
- MuseWise pipeline

⇒ Different aspects of Provenance

- How to **collect** the provenance information
- How to **store** this information
- How to **access** and **visualize** the provenance
- Provenance **on-top** or **inside**

Use cases addressed

A: Traceability of products

- Having a dataset, find the progenitors and in particular locate the raw data
- Find out what processing has been already performed for a given dataset

B: Acknowledgment and contact information

- Find out who was on shift for data taking for a given dataset
- Find out which proposals and PIs/CoIs are associated to a given dataset

C: Reliability and Quality assessment

- Get detailed information on the methods/tools/software that were involved
- Check if the processing steps (including data acquisition) went "well"
- Extract the ambient conditions during data acquisition (cloud coverage? wind? temperature?)

Tasks involving provenance

D: Identification of error location

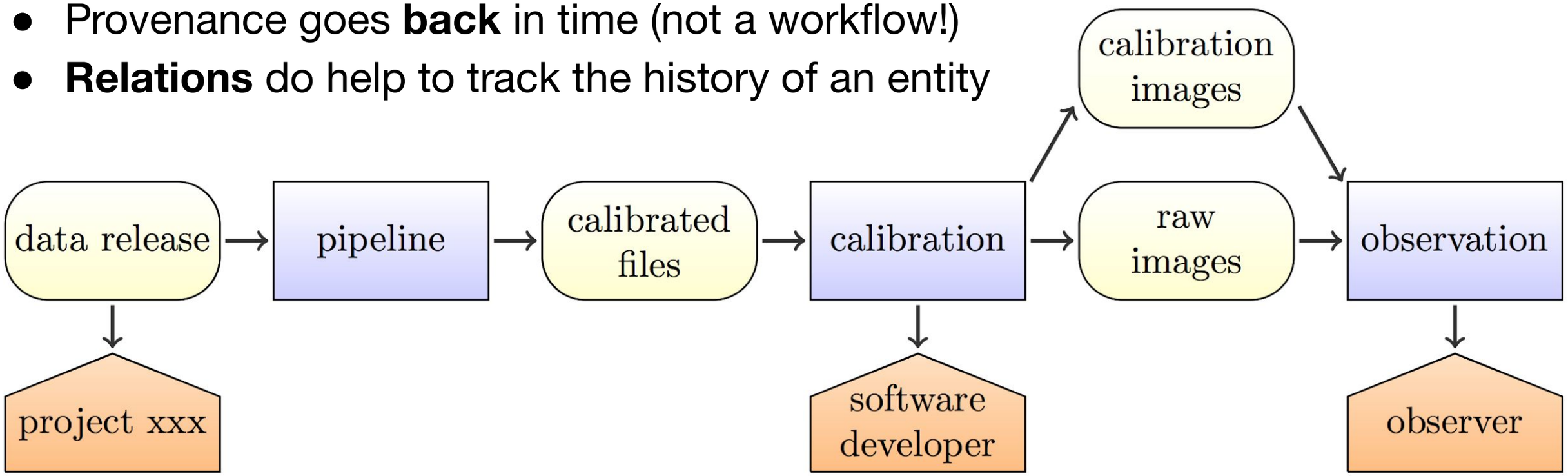
- I found something strange in an image. Was there anything strange noted when the image was taken? a warning during the processing?
- Which pipeline version was used, the old one with a known bug for treating bright objects or a newer version?
- What was the detailed configuration of the pipeline? were the parameters correctly set for the image cleaning step?

E: Search in provenance metadata

- Find more images that were produced using the same version of the CTA pipeline.
- Get an overview of all images reduced with the same calibration dataset.
- Extract all the provenance information of a SVOM light curve or spectrum to reprocess the raw data with refined parameters.

Chain of entities, activities, agents

- Provenance goes **back** in time (not a workflow!)
- **Relations** do help to track the history of an entity



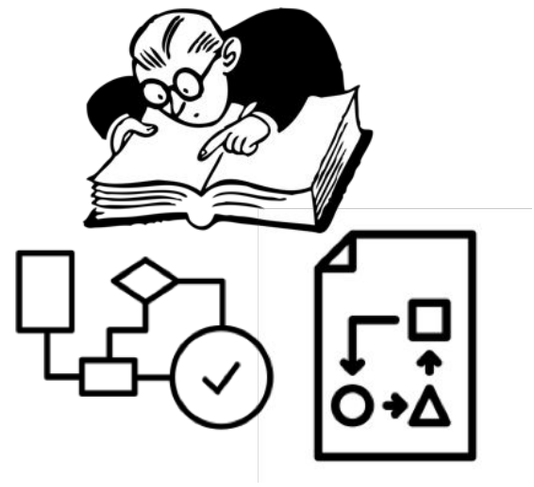
- Does this answers all our goals? A, B: **yes**, but C, D, E: **no**
- Need more **information** that will be relevant to :
  - assess the quality and reliability (C),
  - locate errors (D)
  - enable searches in provenance metadata (E)
- The core model is **too generic** to provide this information

Relevant provenance information?

C: Quality and Reliability*:*

How was the calibration performed, which steps, which algorithms?

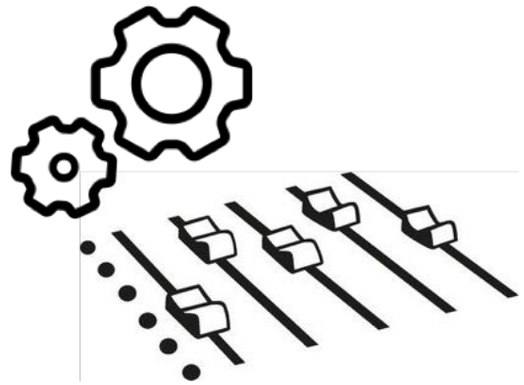
⇒ **Descriptions**: information on the expected working of an activity and on the expected structure of an entity, i.e what is known before any activity or entity instance is created



D: Locate errors

What was the detailed configuration of this pipeline step? Are all parameters adjusted to fit my needs?

⇒ **Configuration information**: parameters given to an activity so that it occurs in the desired conditions.



C + D tasks

Need Context: information on something that influences the development of an activity, but for which there is no or little control at the moment of its execution (e.g. Ambient Conditions, Instrumental Context, Execution Environment).

IVOA Provenance Data Model

Proposed recommendation:

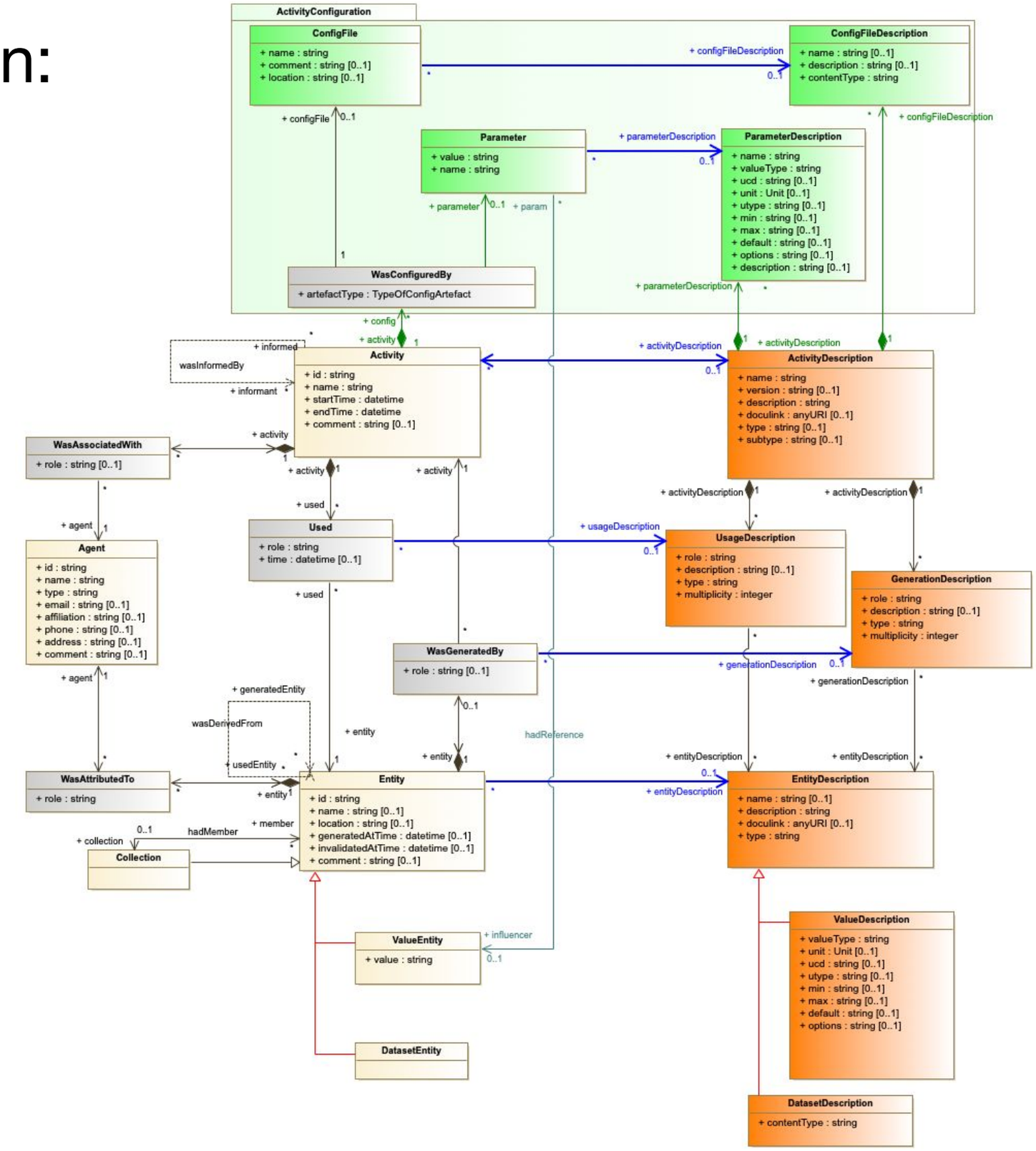
- Core W3C model
- Activity **Descriptions**
- Activity **Configuration**

+ Detailed granularity and execution **context**

⇒ **Reproducibility**

+ Relevant information attached to datasets

⇒ **Pertinence** of dataset for Science



Descriptions and specific entities

Usage/Generation

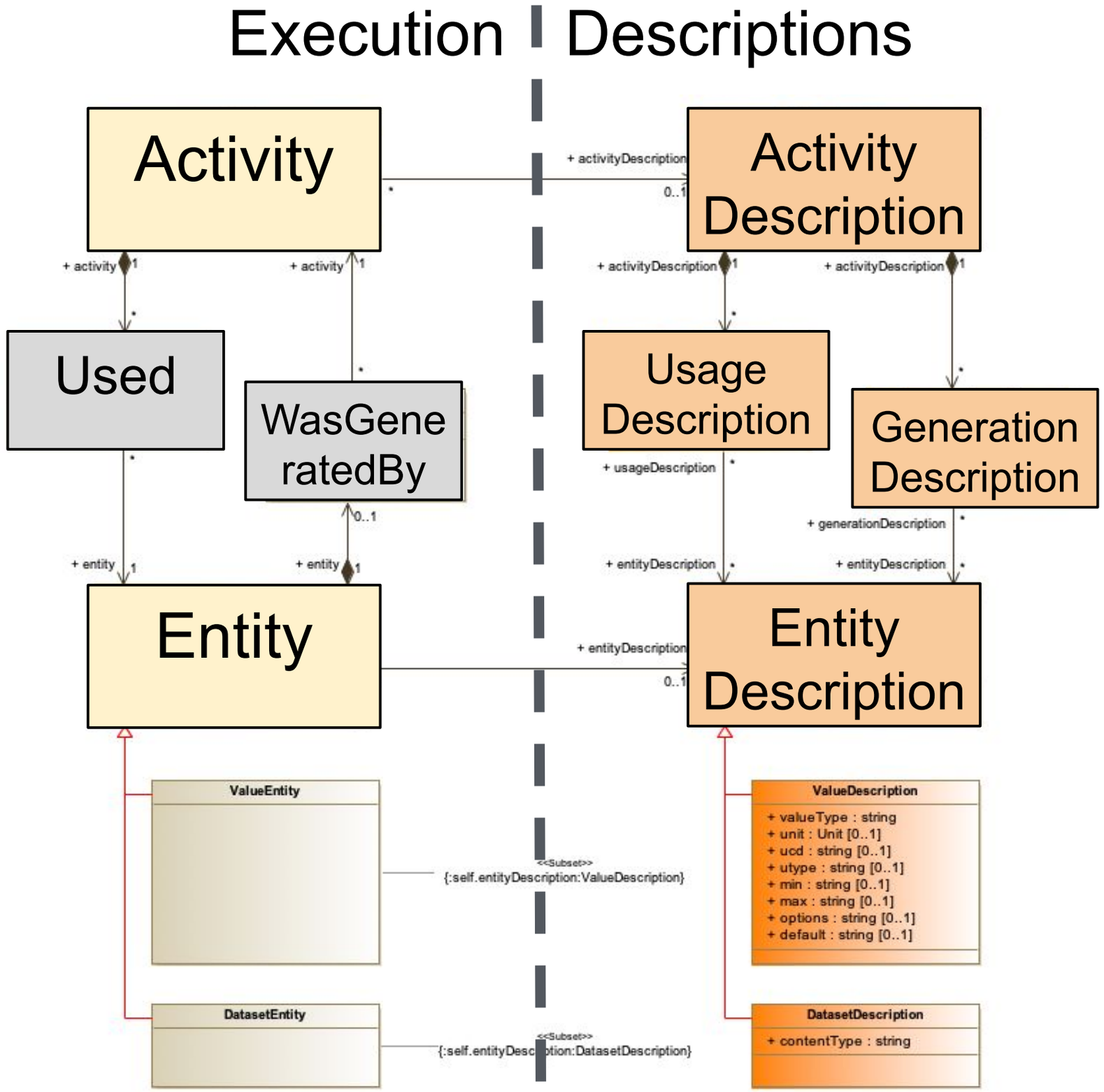
- **role** (master_bias, IRF, eventlist, ...)
- **type** (main, calibration, preview, quality, log, context)

Value

- **valueType**
- unit/ucd/utype
- min/max/options

Dataset

- **contentType** (similar to access_format in ObsCore)



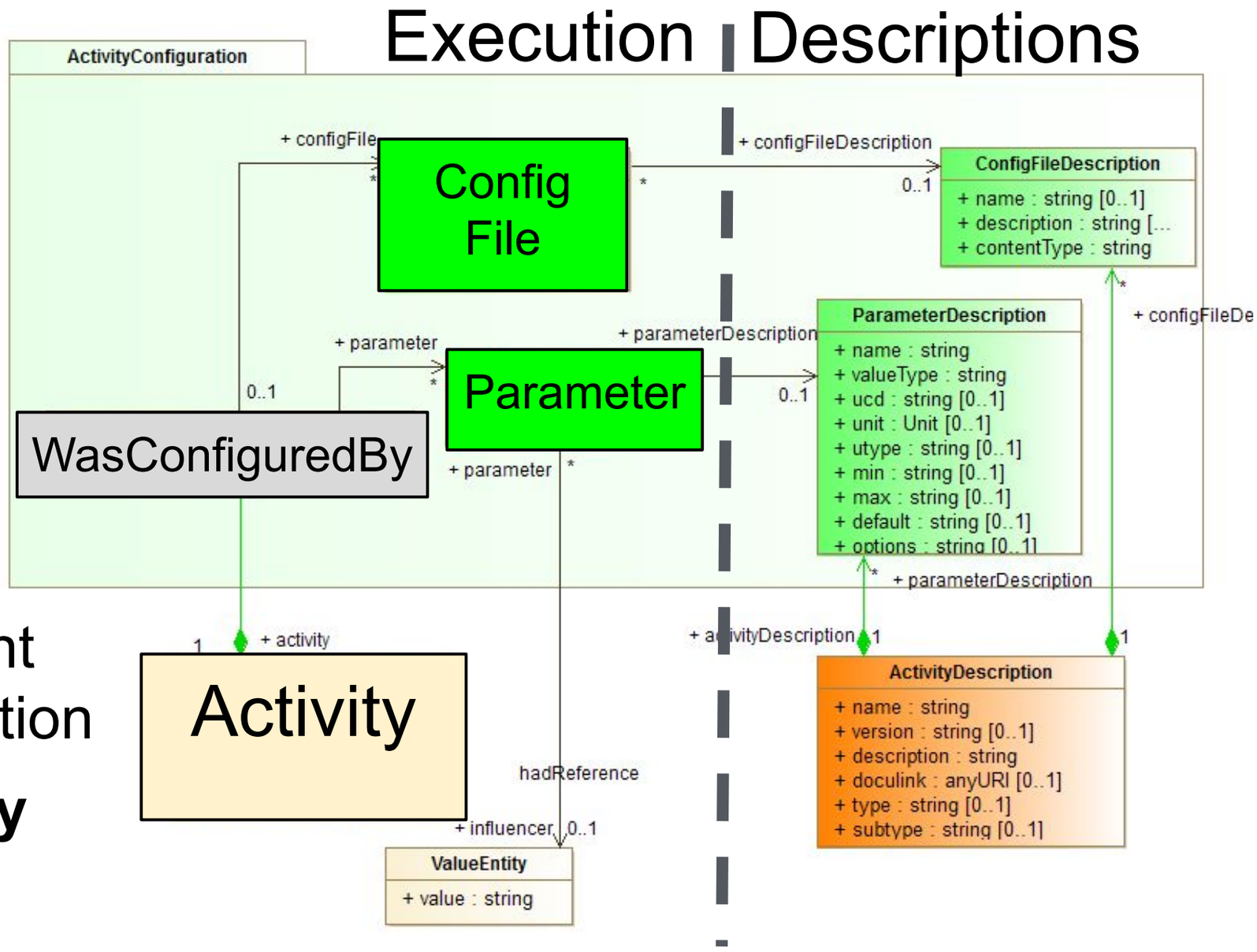
Configuration Package

- Identify **how** a data product was produced ⇒ **Provenance**
- Identify what **detailed options** were used ⇒ **Configuration**
- Different **lifecycle,** different **access** and **usage** → **different classes**

- Parameter
  - name=value
- ConfigFile
  - name & location
- Descriptions

Configuration is dependent on Activity/ActivityDescription

⇒ **fosters reproducibility**



ActivityDescrition examples - ESO

ESO recipe fors_img_screen_flat

ftp://ftp.eso.org/pub/dfs/pipelines/fors/fors-pipeline-manual-5.7.pdf

9.3.1 Input files

SCREEN_FLAT_IMG: required set of raw, unprocessed screen flat field frames.
MASTER_BIAS: required master bias frame. Just one should be given.

9.3.2 Output files

MASTER_SCREEN_FLAT_IMG: Master screen flat field calibration frame. Configuration parameters directly affecting this product are: --stack_method, --xradius, --yradius, -- degree, and --sampling.

9.3.3 Configuration parameters

The following parameters determine how the fors_img_screen_flat recipe will process the input frames.
--stack_method: Frames combination method. Default: average
See explanation in recipe fors_bias configuration parameters (Section 9.1.3, page 76).
--xradius: Median filter x radius (unbinned pixels). Default: 50 pixel
See the --yradius parameter.
--yradius: Median filter y radius (unbinned pixels). Default: 50 pixel
These parameters define the size of the running box used for smoothing the flat field for determining the large scale trend to remove. These parameters are ignored if the --degree parameter is greater than zero.
--degree: Degree of bivariate fitting polynomial. Default: -1
If this parameter is greater than or equal to 0, then a polynomial with the specified degree will be fitted to the illuminated part of the CCD for determining the flat field large scale trend to remove.
...

ActivityDescrition examples - hipsgen

Aladin hipsgen

Usage:

java -jar Aladin.jar -hipsgen in=file|dir [otherParams ... ACTIONs ...]

java -jar Aladin.jar -hipsgen -param=configfile

The config file must contain these following options, or use them directly on the command line :

Required parameter:

in=dir: Source image directory (FITS or JPEG|PNG +hhh or HiPS), unique image or HEALPix map file

Basic optional parameters:

out=dir: HiPS target directory (default ./+"AUTHORITY_internalID")

obs_title=name: Name of the survey (by default, input directory name)

creator_did=id: HiPS identifier (syntax: [ivo://]AUTHORITY/internalID)

hips_creator=name: Name of the person|institute who builds the HiPS

hips_status=xx: HiPS status (private|public clonable|clonableOnce|unclonable - default: public clonableOnce)

hdu=n1,n2-n3,...|all: List of HDU numbers (0 is the primary HDU - default is 0)

blank=nn: Specifical BLANK value

skyval=key|auto|%info|%min %max: Fits key to use for removing a sky background, or auto detection or percents of pixel histogram kept (central ex 99, or min max ex 0.3 99.7)

...

Advanced optional parameters:

hips_order=nn: Specifical HEALPix order - by default, adapted to the original resolution

hips_pixel_bitpix=nn: Specifical target bitpix (-64|-32|8|16|32|64)

...

ActivityDescrition examples - gammapy

GammaPy image bin

Usage: gammapy image bin [OPTIONS] **EVENT_FILE REFERENCE_FILE OUT_FILE**

Bin events into an **image.**

You have to give the event, reference and out FITS filename.

Options:

--overwrite: Overwrite existing files?

-h, **--help**: Show this message and exit.

Submit a registration form

Web form inputs:

Title: Mr./Mrs./Ms./Mx./Dr./Prof.

First Name:

Last Name:

Name Printed on Name Badge:

University/Affiliation:

...

Banquet Dinner: yes/no

...

Abstract:

Output: registration email, records in a database

- Those activity descriptions define **inputs**, **outputs** and **parameters**
- Sometimes parameters are in fact (or point to) **entities** that needs to be traced, e.g. **files**, **devices** or **documents**
- What is to be traced is the decision of the project or user that provides this activity
  → what is **relevant**
  → **granularity**
  → **level of detail**