# IVOA ProvTAP :

## November 2021 status

F.Bonnarel, CDS
on behalf of M.Servillat, M.Louys, M.Nullmeier, M.Sanguillon, L.Michel

CENTRE DE DONNÉES ASTRONOMIQUES DE STRASBOURG

ESCAPE
European Science Cluster of Astronomy & Particle physics ESFRI research Infrastructures

IVOA

# Why Provenance in TAP ?

- Provenance information can be attached to data in various ways :
  - Embedded in the data « header » itself
  - Linked to the data record via DataLink or URL
  - Retrievable via ProvSAP via data id.
- In addition to that , Provenance metadata in a TAP service will allow to discover « data » by constraining Provenance features.
  - It's a « reverse » mechanism.

# « The » issue = complexity

(see « FAIR high level data for Cherenkov astronomy » )

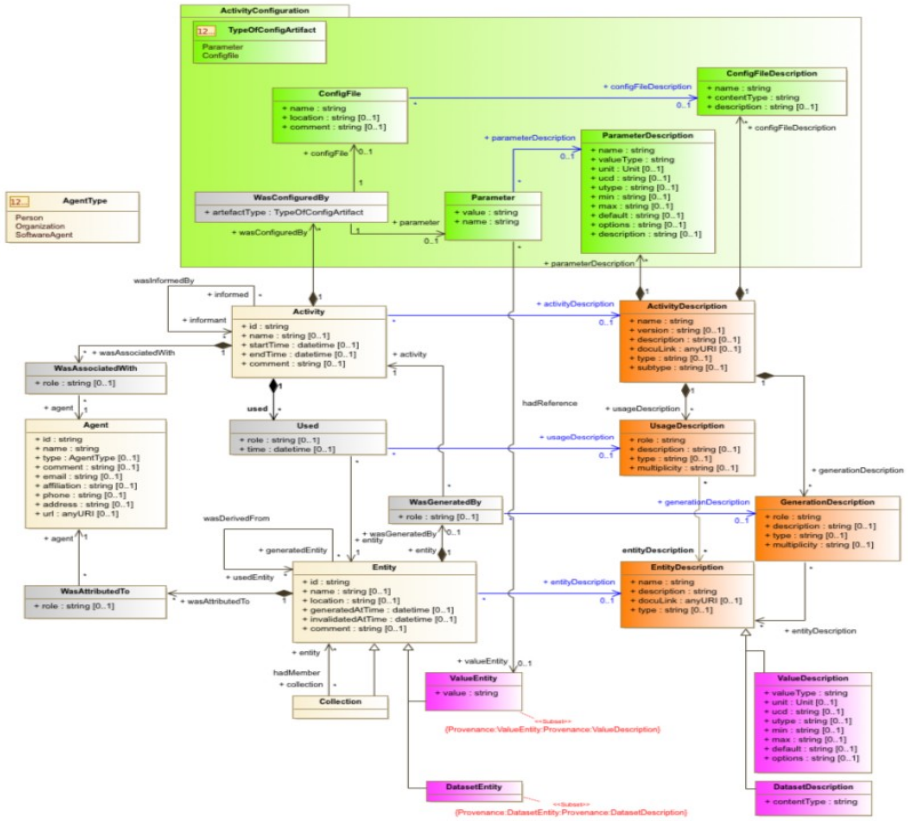



*Figure 8:* Full class diagram of the IVOA Provenance Data Model.

# « The » issue = complexity

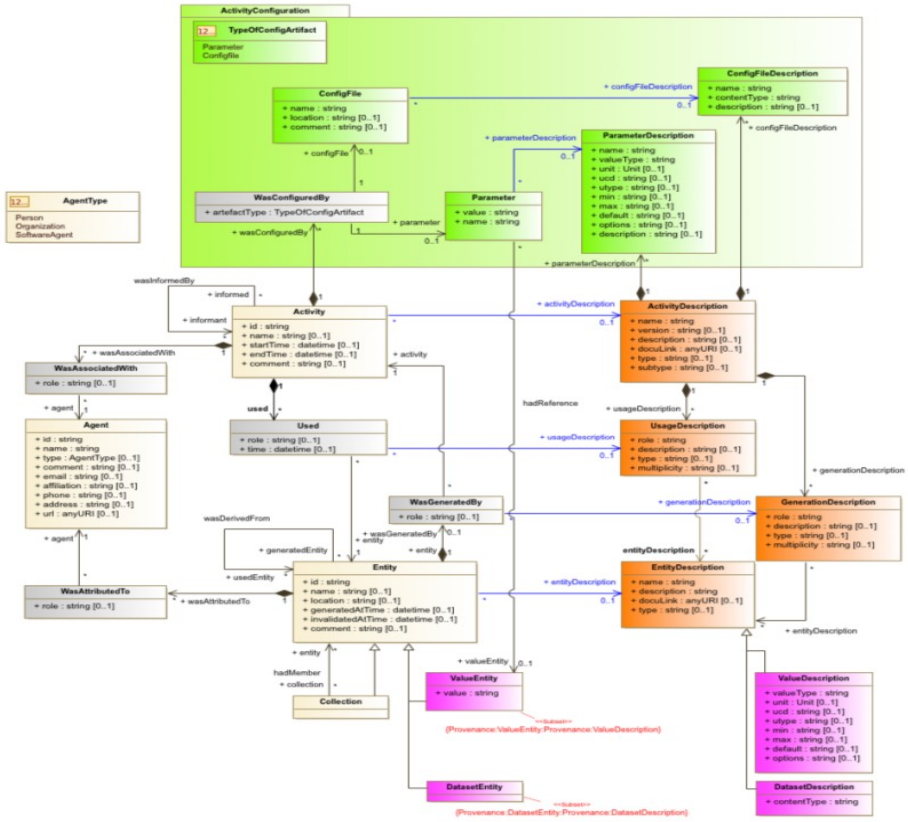



*Figure 8:* Full class diagram of the IVOA Provenance Data Model.

->See M.Servillat

Talk tomorrow (DM)

->1 table per class ?

---→ ProvTAP

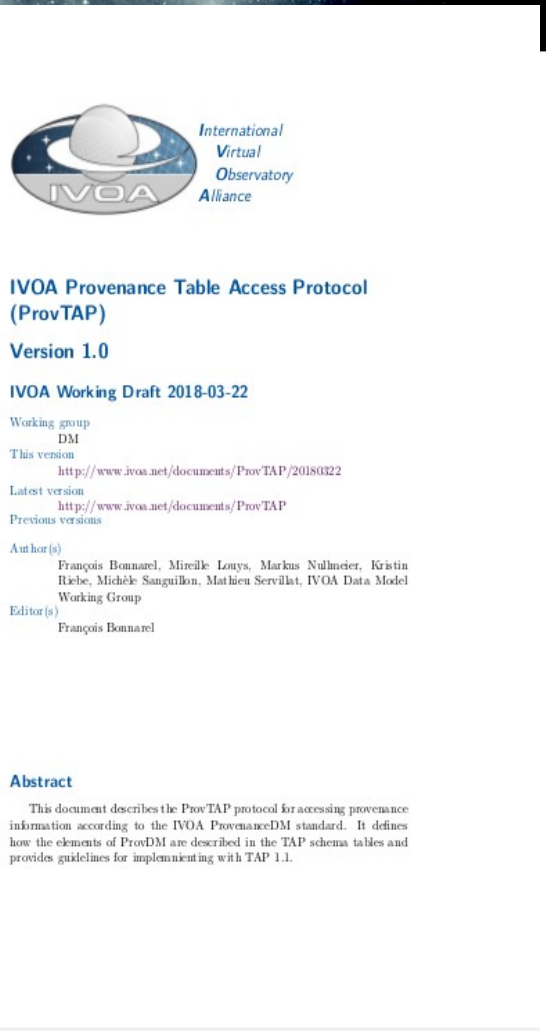
# ProvTAP status

- There is an internal draft
on the IVOA DAL page

- TAP schema mapping classes
as tables

- ProvHiPS (provenance of
HiPS and HiPS tiles) is
an implementation prototype

- From now examples and demos
from ProvHiPS



International Virtual Observatory Alliance

**IVOA Provenance Table Access Protocol (ProvTAP)**

**Version 1.0**

**IVOA Working Draft 2018-03-22**

Working group
DM
This version
http://www.ivoa.net/documents/ProvTAP/20180322
Latest version
http://www.ivoa.net/documents/ProvTAP
Previous versions

Author(s)
François Bonnarel, Mireille Louys, Markus Nullmeier, Kristin Riebe, Michèle Sanguillon, Mathieu Servillat, IVOA Data Model Working Group
Editor(s)
François Bonnarel

**Abstract**

This document describes the ProvTAP protocol for accessing provenance information according to the IVOA ProvenanceDM standard. It defines how the elements of ProvDM are described in the TAP schema tables and provides guidelines for implemnienting with TAP 1.1.

# ProvTAP TAP_SCHEMA: Entity table

| Name | ucd | utype | datatype | status |
|---|---|---|---|---|
| e_id | meta.id | voprov:Entity.id | char | M |
| e_name | meta.title | voprov:Entity.name | char | O |
| e_type | meta.code.class | voprov:Entity.type | char | O |
| e_rights | meta.code.class | voprov:Entity.rights | char | O |
| e_location | meta.ref.url | voprov:Entity.location | char | O |
| e_generated | time.start | voprov:Entity.generatedAtTime | char | O |
| e_invalidated | time.stop | voprov:Entity.invalidatedAtTime | char | O |
| e_comment | meta.description | voprov:Entity.comment | char | O |
| e_classtype | meta.code.class | voprov:Entity.classtype | char OPTION | M |
| e_value | stat.value | voprov:Entity.value | char | O |
| → e_description | meta.id | voprov:Entity.description_id | reference | O |

*Table 2:* Column description for Entity table. The e_classtype column may have the following two values :"dataset" and "value"

# ProvTAP TAP_SCHEMA: parameterDescription table

| Name | ucd | utype | datatype |
|---|---|---|---|
| pd_activitydescription | meta.id | voprov:ParameterDescription.<br>activityDescription_id | char |
| pd_id | meta.id | voprov:ParameterDescription.id | char |
| pd_name | meta.title | voprov:ParameterDescription.name | param dependent |
| pd_description | meta.description | voprov:ParameterDescription.description | char |
| pd_datatype | meta | voprov:ParameterDescription.datatype | char |
| pd_unit | meta.unit | voprov:ParameterDescription.unit | char |
| pd_ucd | meta.ucd | voprov:ParameterDescription.ucd | char |
| pd_utype | meta | voprov:ParameterDescription.utype | char |
| pd_min | stat.min | voprov:ParameterDescription.min | param dependent |
| pd_max | stat.max | voprov:ParameterDescription.max | param dependent |
| pd_options | meta | voprov:ParameterDescription.options | param dependent |

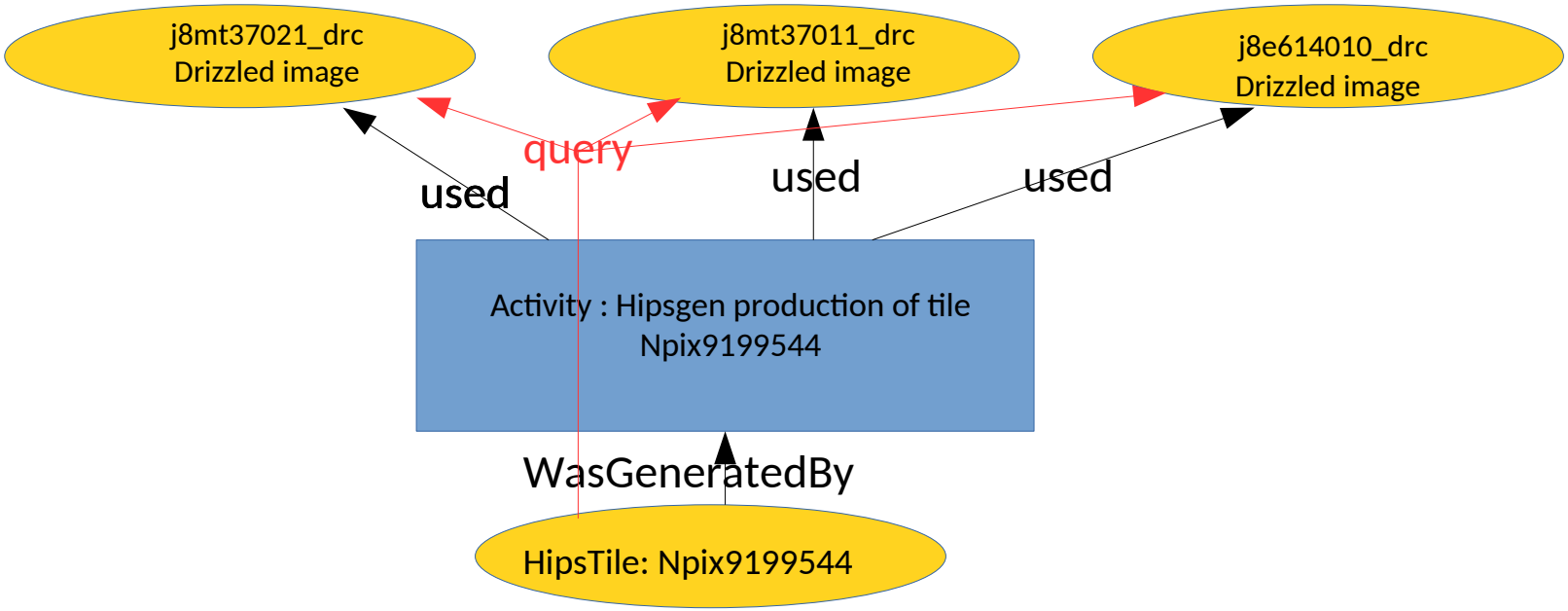
# ProvTAP TAP_SCHEMA: parameter table

| Name | ucd | utype | datatype |
|---|---|---|---|
| p_activity | meta.id | voprov:Parameter.activity_id | char |
| p_id | meta.id | voprov:Parameter.id | char |
| p_value | stat.value | voprov:Parameter.value | param dependent |
| → p_description | meta.id | voprov:Parameter.parameterDescription_id | reference to parameter description |

*Table 8:* Column description for Parameter table

# ProvHiPS ADQL query examples :

## Finding out drizzled images « progenitors » of a specific HiPS tile.



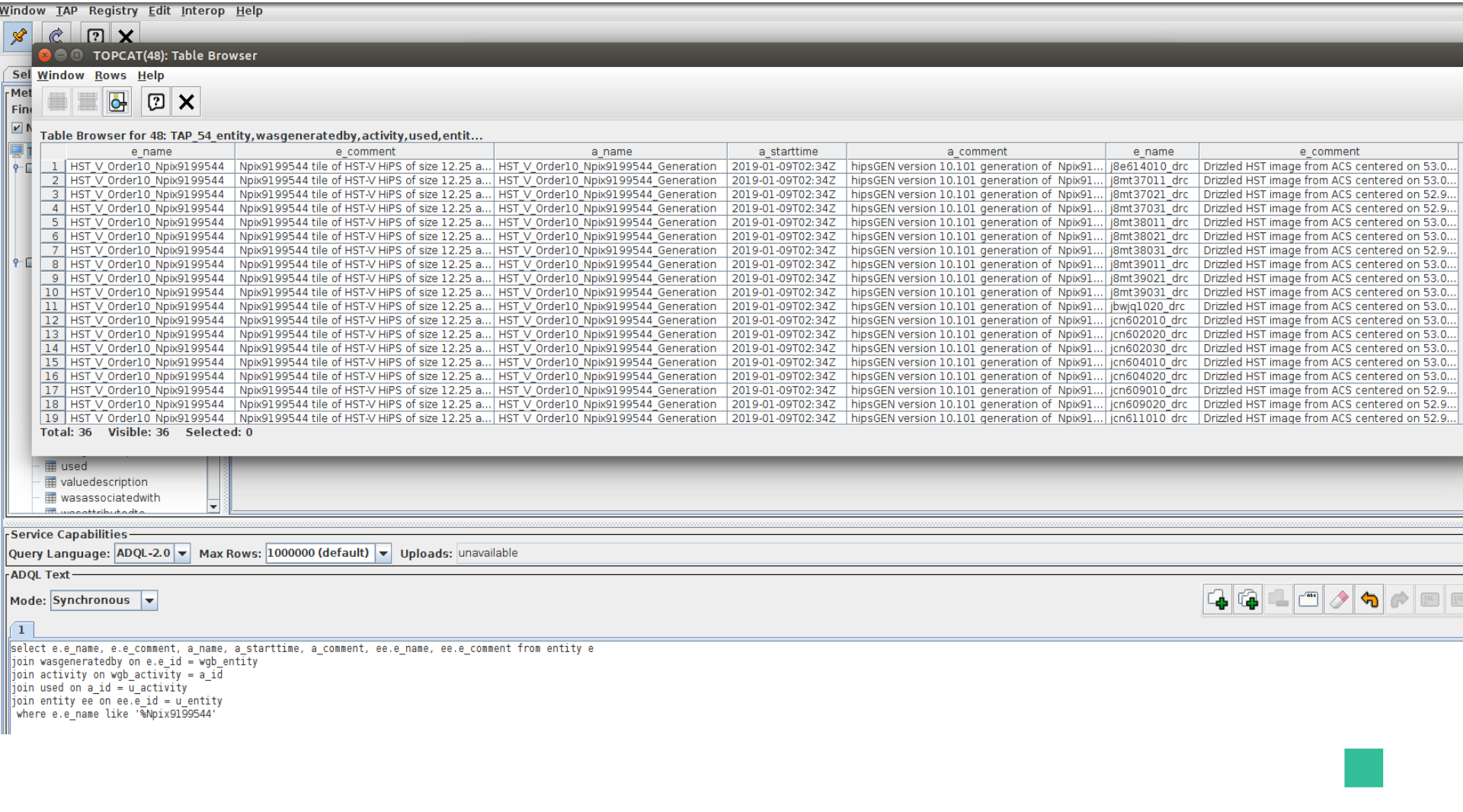
# ProvHiPS ADQL query examples :

## Finding out drizzled images « progenitors » of a specific HiPS tile.

```
select e.e_name, e.e_comment, a_name, a_starttime, a_comment, ee.e_name,
    ee.e_comment from entity e
join wasgeneratedby on e.e_id = wgb_entity
join activity on wgb_activity = a_id
join used on a_id = u_activity
join entity ee on ee.e_id = u_entity
where e.e_name like '%Npix9199544'
```

# ProvHiPS ADQL query examples :

## Finding out drizzled images « progenitors » of a specific HiPS tile.



# Finding agent and activity to whom entity is attributed and generating a given entity

- select ag_name,ag_type,ag_comment, e_name, e_generated,e_location, e_comment, a_name, a_comment

  from agent

  join wasattributedto on ag_id = wat_agent

  join entity on wat_entity = e_id

  join wasgeneratedby on e_id = wgb_entity

  join activity on a_id = wgb_activity

  where e_name = 'jbyq07020_drc'

# Issues

- Table is denormalized : a lot of redundant information
- Loop issue : several occurances of the same triplet (name,utype,ucd) in the same table for different « objects »
- Let’s try minimum or last step provenance by creating a standardized view

# Solutions
## -1 Single step = single table ( = join)

- The join is a permanent view described in the TAP schema
- Columns :

  entity_name, entity_location, entity_comment, …

  generating_activity_name, generating_activity_starttime, ….

  agent_role, agent_name, ….

  used_entity_list
- → Redundancy may be avoided if we group all used entities ids in a single cell
- → possible Recursivity

# Solutions

## Single step ( = join) – 1line per used entity

- View (in postgres)

create view last_step_provenance as select

e.e_name as entity_name, e.e_location as entity_location, e.e_generated as entity_generated, e.e_invalidated as entity_invalidated, e.e_comment as entity_comment,

a_name as generating_activity_name,a_starttime as generating_activity_starttime, a_endtime as generating_activity_endtime, a_comment as generating_activity_comment,

wat_role as agent_role, ag_name as agent_name, ag_type as agent_type ,ag_affiliation as agent_affiliation, ag_email as agent_email, ag_address as agent_address, ag_phone as agent_phone, ag_comment as agent_comment,

ee.e_name as used_entity_name from entity as e

**join wasgeneratedby on wgb_entity = e.e_id**
*join activity on a_id = wgb_activity*
**join used on u_activity = a_id**
*join entity as ee on ee.e_id = u_entity*
**join wasattributedto on wat_entity = e.e_id**
*join agent on ag_id = wat_agent ;*

# Solutions

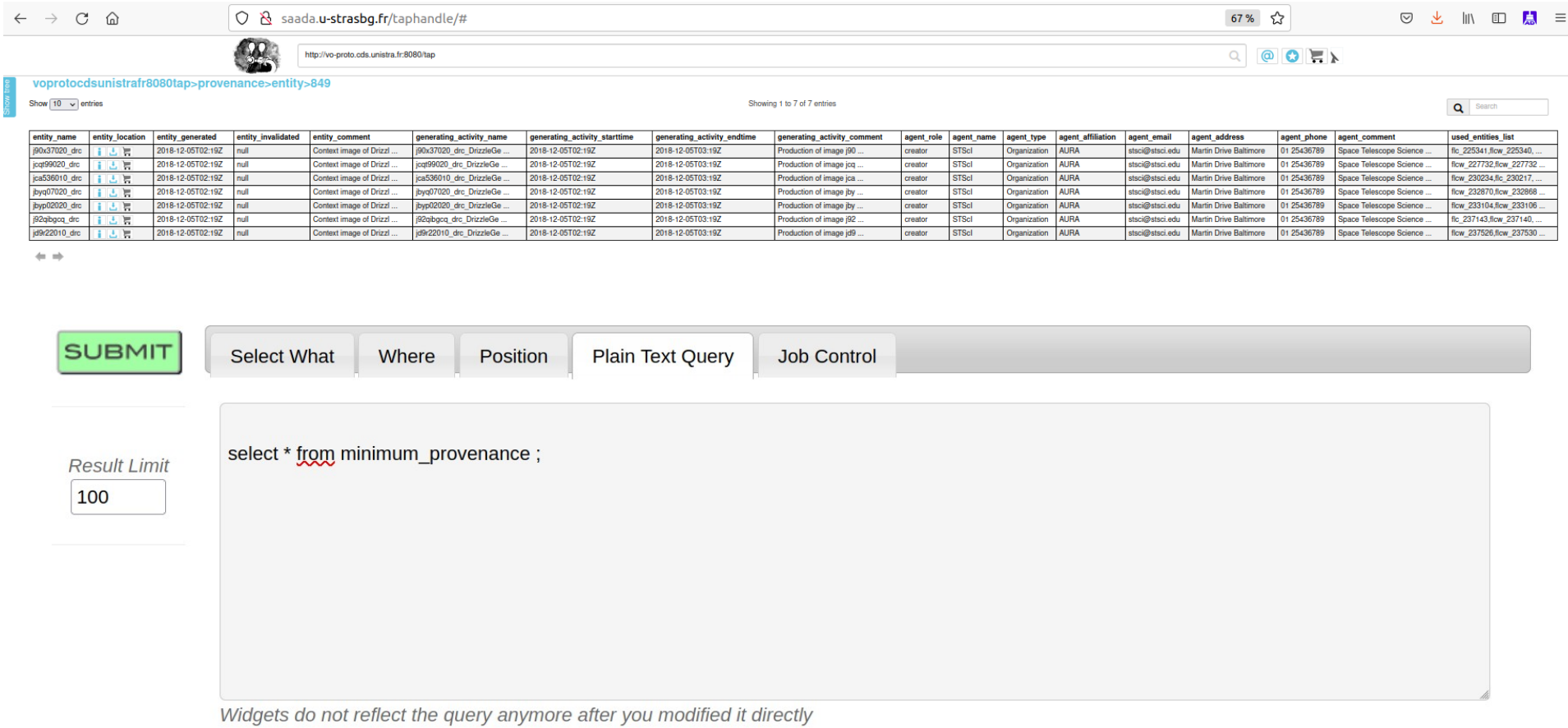
## Single step ( = join) 1 single line per generated entity

- View (in postgres)

```
create view minimum_provenance as select

e.e_id AS entity_id,  e.e_name AS entity_name, e.e_location AS entity_location, e.e_generated AS entity_generated,
    e.e_invalidated AS entity_invalidated, e.e_comment AS entity_comment,

 activity.a_name AS generating_activity_name, activity.a_starttime AS generating_activity_starttime,
activity.a_endtime AS generating_activity_endtime, activity.a_comment AS generating_activity_comment,

 wasattributedto.wat_role AS agent_role, agent.ag_name AS agent_name, agent.ag_type AS agent_type,
    agent.ag_affiliation AS agent_affiliation, agent.ag_email AS agent_email, agent.ag_address AS agent_address,
    agent.ag_phone AS agent_phone, agent.ag_comment AS agent_comment,

string_agg(used.u_entity::text, ','::text) AS used_entities_list
FROM entity e

 JOIN wasgeneratedby ON e.e_id::text = wasgeneratedby.wgb_entity::text
 JOIN activity ON wasgeneratedby.wgb_activity::text = activity.a_id::text
join used on u_activity = a_id
join entity as ee on ee.e_id = u_entity
join wasattributedto on wat_entity = e.e_id
join agent on ag_id = wat_agent ;
```

# Solutions

## Single step = view query execution



# Two queries with these flat views

- select * from last_step_provenance where
  entity_name = 'j90x37020_drc'

  → response in several lines with redundancy
- select * from minimum_provenance where
  entity_name = 'j90x37020_drc'

  → response in one line

# Success and limitations view 1

- Clear column names for distinct objects
- Rather simple recursivity to manage
- But :
  - Redondancy still there
  - No direct retrieval for chains of provenance

# Success and limitations view 2

- Clear column names for distinct objects
- No more redundancy
- But :
  - Complex recursivity to manage
  - No direct retrieval for chains of provenance

# Entity + dataset descriptions

- Finding datasetdescription for entities used by an activity started in november 2018.

```
SELECT  *
FROM datasetdescription
join entity on e_description = dd_id
join used on e_id = u_entity
join activity on a_id = u_activity
where a_starttime like '2018-11%'
```

# Entity + dataset descriptions

Enter a TAP service URL or a keyword

Tap Nodes

- saadaunistrafprovtap SJAU
  - TAP_SCHEMA
  - ivoa
  - provenance
    - entity
    - datasetdescription
    - valuedescription
    - activity
    - activitydescription
    - agent
    - parameter
    - parameterdescription
    - used
    - usagedescription
    - wasgeneratedby
    - generationdescription
    - wasassociatedwith
    - wasattributedto
    - wasinformedby
    - wasderivedfrom
    - collection
    - configfile
    - minimum_provenance
    - last_step_provenance
    - configfiledescription
    - wasconfiguredby
  - Goodies (not used yet)

Hide tree

## saadaunistrafrprovtap>provenance>entity>753

Show 10 entries

Showing 1 to 10 of 509 entries

Search

| dd_name | dd_description | dd_content | dd_type | dd_subtype | dd_doculink | e_id | e_name | e_type | e_rights | e_location | e_generated | e_invalidated | e_com |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| HST_Drizzled_Images_Des ... | Coadded HST images obta ... | application/x-fits-mef | data | coadd_calib | | drz_999945 | j8qj18011_drz | null | public | | null | null | Drizzle |
| HST_Drizzled_Images_Des ... | Coadded HST images obta ... | application/x-fits-mef | data | coadd_calib | | drz_991418 | j8qj18021_drz | null | public | | null | null | Drizzle |
| HST_Drizzled_Images_Des ... | Coadded HST images obta ... | application/x-fits-mef | data | coadd_calib | | drz_1005227 | j8qj19011_drz | null | public | | null | null | Drizzle |
| HST_Drizzled_Images_Des ... | Coadded HST images obta ... | application/x-fits-mef | data | coadd_calib | | drz_991486 | j8qj19021_drz | null | public | | null | null | Drizzle |
| HST_Drizzled_Images_Des ... | Coadded HST images obta ... | application/x-fits-mef | data | coadd_calib | | drz_999407 | j92k53011_drz | null | public | | null | null | Drizzle |
| HST_Drizzled_Images_Des ... | Coadded HST images obta ... | application/x-fits-mef | data | coadd_calib | | drz_991490 | j92k53021_drz | null | public | | null | null | Drizzle |
| HST_Drizzled_Images_Des ... | Coadded HST images obta ... | application/x-fits-mef | data | coadd_calib | | drz_998845 | j92k53071_drz | null | public | | null | null | Drizzle |
| HST_Drizzled_Images_Des ... | Coadded HST images obta ... | application/x-fits-mef | data | coadd_calib | | drz_999407 | j92k53011_drz | null | public | | null | null | Drizzle |
| HST_Drizzled_Images_Des ... | Coadded HST images obta ... | application/x-fits-mef | data | coadd_calib | | drz_991490 | j92k53021_drz | null | public | | null | null | Drizzle |
| HST_Drizzled_Images_Des ... | Coadded HST images obta ... | application/x-fits-mef | data | coadd_calib | | drz_998845 | j92k53071_drz | null | public | | null | null | Drizzle |

Hide query

SUBMIT | Select What | Where | Position | Plain Text Query | Job Control

Result Limit: 100

```
SELECT *
FROM datasetdescription
join entity on e_description = dd_id
join used on e_id = u_entity
join activity on a_id = u_activity
where a_starttime like '2018-11%'
```

*Widgets do not reflect the query anymore after you modified it directly*

# Full entity view

This view gathers entity and datasetdescription columns in one single table

```
create view full_entity as select
e_id,e_name,  e_location, e_generated, e_invalidated,
e_comment
 dd_name, dd_type, dd_subtype, dd_doculink
from entity
join datasetdescription on dd_id = e_description ;
```

# Reduced query using the view

```
SELECT  *
FROM full_entity
join used on e_id = u_entity
join activity on a_id = u_activity
where a_starttime like '2018-11%' ;
```

# Reduced query using the view

Enter a TAP service URL or a keyword

Tap Nodes

- saadaunistrafrprovtap SJAU
  - TAP_SCHEMA
  - ivoa
  - provenance
    - entity
    - datasetdescription
    - valuedescription
    - activity
    - activitydescription
    - agent
    - parameter
    - parameterdescription
    - used
    - usagedescription
    - wasgeneratedby
    - generationdescription
    - wasassociatedwith
    - wasattributedto
    - wasinformedby
    - wasderivedfrom
    - collection
    - configfile
    - minimum_provenance
    - last_step_provenance
    - configfiledescription
    - wasconfiguredby
- Goodies (not used yet)

Hide tree

## saadaunistrafrprovtap>provenance>entity>60

Show 10 entries

Showing 1 to 10 of 509 entries

Search

| e_name | e_location | e_generated | e_invalidated | e_comment | dd_name | dd_type | dd_subtype | dd_doculink | u_entity | u_activity | u_time | u_usagedescription | a_id |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| j8qj18011_drz | | null | null | Drizzled HST image from ... | HST_Drizzled_Images_Des ... | data | coadd_calib | | drz_999945 | a_1026118 | null | drztgen | a_1026 |
| j8qj18021_drz | | null | null | Drizzled HST image from ... | HST_Drizzled_Images_Des ... | data | coadd_calib | | drz_991418 | a_1026118 | null | drztgen | a_1026 |
| j8qj19011_drz | | null | null | Drizzled HST image from ... | HST_Drizzled_Images_Des ... | data | coadd_calib | | drz_1005227 | a_1026118 | null | drztgen | a_1026 |
| j8qj19021_drz | | null | null | Drizzled HST image from ... | HST_Drizzled_Images_Des ... | data | coadd_calib | | drz_991486 | a_1026118 | null | drztgen | a_1026 |
| j92k53011_drz | | null | null | Drizzled HST image from ... | HST_Drizzled_Images_Des ... | data | coadd_calib | | drz_999407 | a_1026476 | null | drztgen | a_1026 |
| j92k53021_drz | | null | null | Drizzled HST image from ... | HST_Drizzled_Images_Des ... | data | coadd_calib | | drz_991490 | a_1026476 | null | drztgen | a_1026 |
| j92k53071_drz | | null | null | Drizzled HST image from ... | HST_Drizzled_Images_Des ... | data | coadd_calib | | drz_998845 | a_1026476 | null | drztgen | a_1026 |
| j92k53011_drz | | null | null | Drizzled HST image from ... | HST_Drizzled_Images_Des ... | data | coadd_calib | | drz_999407 | a_1026478 | null | drztgen | a_1026 |
| j92k53021_drz | | null | null | Drizzled HST image from ... | HST_Drizzled_Images_Des ... | data | coadd_calib | | drz_991490 | a_1026478 | null | drztgen | a_1026 |
| j92k53071_drz | | null | null | Drizzled HST image from ... | HST_Drizzled_Images_Des ... | data | coadd_calib | | drz_998845 | a_1026478 | null | drztgen | a_1026 |

Hide query

SUBMIT | Select What | Where | Position | Plain Text Query | Job Control

Result Limit: 100

```
SELECT *
FROM full_entity
join used on e_id = u_entity
join activity on a_id = u_activity
where a_starttime like '2018-11%'
;
```

*Widgets do not reflect the query anymore after you modified it directly*

# Other views

- TBD :
  - Simplify  the « description part » and point to Workflow description
  - Important for CTA, SKA ??

# Standard

- Add some of these views
  as standard official simplification
- Allow to create services providing only
  the simplified views

# Going Further
## (see TAP and datamodels BOF ) + posters

- TAP annotation of the simple table query
  (see next slides )
- Renormalized response (see BOF)
- Instance query
  (see BOF) → no simulation here

# ProvTAP annotation (on top of first query response)

```
−<VOTABLE version="1.3" xsi:schemaLocation="http://www.ivoa.net/xml/VOTable/v1.3 http://www.ivoa.net/xml/VOTable/v1.3">
 −<VODML>
  −<MODEL_INSTANCE name="ProvDM" syntax="ModelInstanceInVot" uri="https://github.com/ivoa-std/MANGO/blob/master/vo-dml/mango.vo-dml.xml">
   −<TABLE_MAPPING tableref="result_S1635375430691">
    −<COLLECTION dmrole="root">
     −<TABLE_ROW_TEMPLATE>
      −<INSTANCE dmrole="root" dmtype="provdm:Entity">
         <ATTRIBUTE dmrole="provdm:Entity.name" dmtype="ivoa:string" ref="e_name"/>
         <ATTRIBUTE dmrole="provdm:Entity.comment" dmtype="ivoa:string" ref="e_comment"/>
        −<INSTANCE dmrole="provdm:Wasgeneratedby" dmtype="provdm:Wasgeneratedby">
         −<INSTANCE dmrole="provdm:Activity" dmtype="provdm:Activity">
            <ATTRIBUTE dmrole="provdm:Activity.name" dmtype="ivoa:string" ref="a_name"/>
            <ATTRIBUTE dmrole="provdm:Activity.startTime" dmtype="ivoa:string" ref="a_starttime"/>
            <ATTRIBUTE dmrole="provdm:Activity.comment" dmtype="ivoa:string" ref="a_comment"/>
           −<COLLECTION>
            −<INSTANCE dmrole="provdm:Used" dmtype="provdm:Used">
             −<INSTANCE dmrole="provdm:Entity" dmtype="provdm:Entity">
                <ATTRIBUTE dmrole="provdm:Entity.name" dmtype="ivoa:string" ref="ee_name"/>
                <ATTRIBUTE dmrole="provdm:Entity.comment" dmtype="ivoa:string" ref="ee_comment"/>
               </INSTANCE>
              </INSTANCE>
             </COLLECTION>
            </INSTANCE>
           </INSTANCE>
          </INSTANCE>
         </TABLE_ROW_TEMPLATE>
        </COLLECTION>
       </TABLE_MAPPING>
      </MODEL_INSTANCE>
     </VODML>
    −<RESOURCE type="results">
       <INFO name="QUERY_STATUS" value="OK"/>
       <INFO name="PROVIDER" value="CDS"/>
       <INFO name="QUERY" value="select e.e_name, e.e_comment, a_name, a_starttime, a_comment, ee.e_name, ee.e_comment from provenance.entity e join wasgeneratedby on e.e_id = wgb_entity join activity on
       wgb_activity = a_id join used on a_id = u_activity join entity ee on ee.e_id = u_entity where e.e_name like '%Npix9199544'"/>
      −<TABLE name="result_S1635375430691">
         <FIELD ID="e_name" arraysize="*" datatype="char" name="e_name" ucd="meta.title" utype="voprov:Entity.name"/>
         <FIELD ID="e_comment" arraysize="*" datatype="char" name="e_comment" ucd="meta.description" utype="voprov:Entity.comment"/>
         <FIELD ID="a_name" arraysize="*" datatype="char" name="a_name" ucd="meta.title" utype="voprov:Activity.name"/>
         <FIELD ID="a_starttime" arraysize="*" datatype="char" name="a_starttime" ucd="time.start" utype="voprov:Activity.startTime"/>
         <FIELD ID="a_comment" arraysize="*" datatype="char" name="a_comment" ucd="meta.description" utype="voprov:Activity.comment"/>
         <FIELD ID="ee_name" arraysize="*" datatype="char" name="e_name" ucd="meta.title" utype="voprov:Entity.name"/>
         <FIELD ID="ee_comment" arraysize="*" datatype="char" name="e_comment" ucd="meta.description" utype="voprov:Entity.comment"/>
        −<DATA>
```

Mapping allows to reproduce the tree/loop structure of the instances of the model

# ProvTAP annotation

- Nice solution if we have a TAP annoter able to generate the annotation
- Need some client
  - Generic ?
  - ProvDM aware ?

# Related Posters

- → see X4-010 poster on all this
  (Object Oriented Data Model strategy in the context of IVOA Table Access Protocol services )
- And also X3-010 (annotation)