IVOA Provenance DM in TAP :

Issues and solutions

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 »)

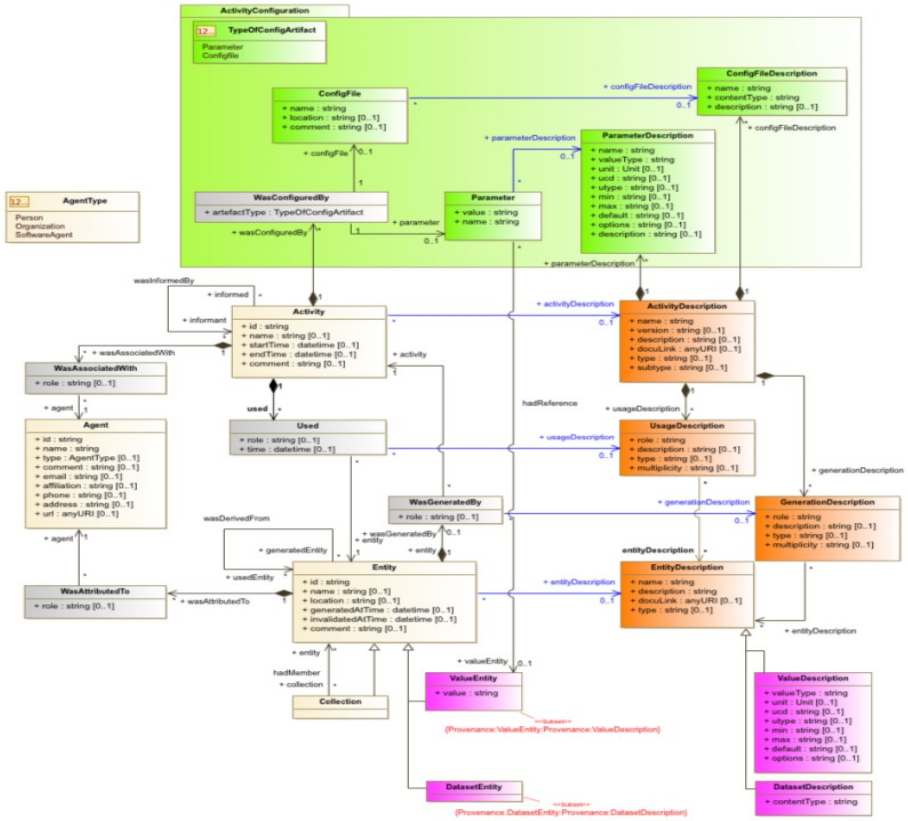


Figure 8: Full class diagram of the IVOA Provenance Data Model.

« The » issue = complexity

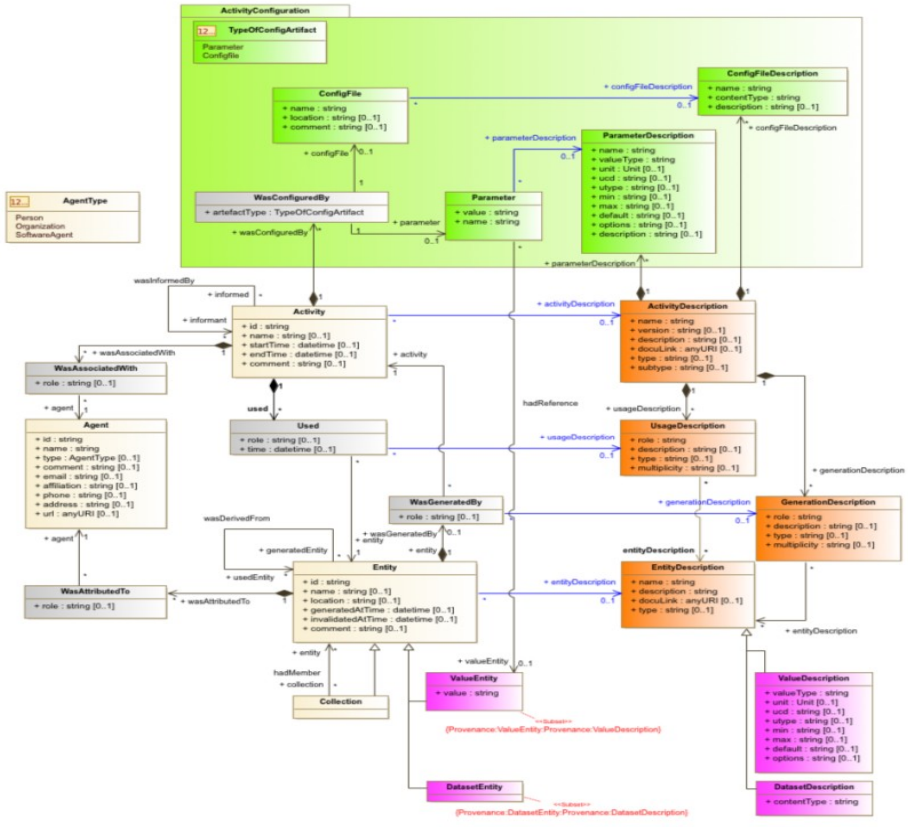


Figure 8: Full class diagram of the IVOA Provenance Data Model.

1 table per class ?

---→ ProvTAP

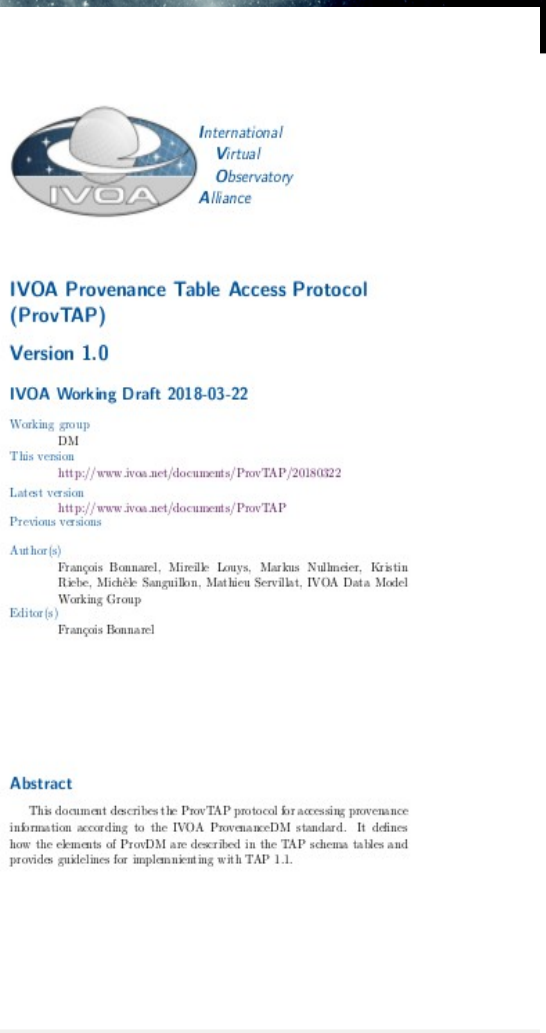
ProvTAP status

- 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. 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

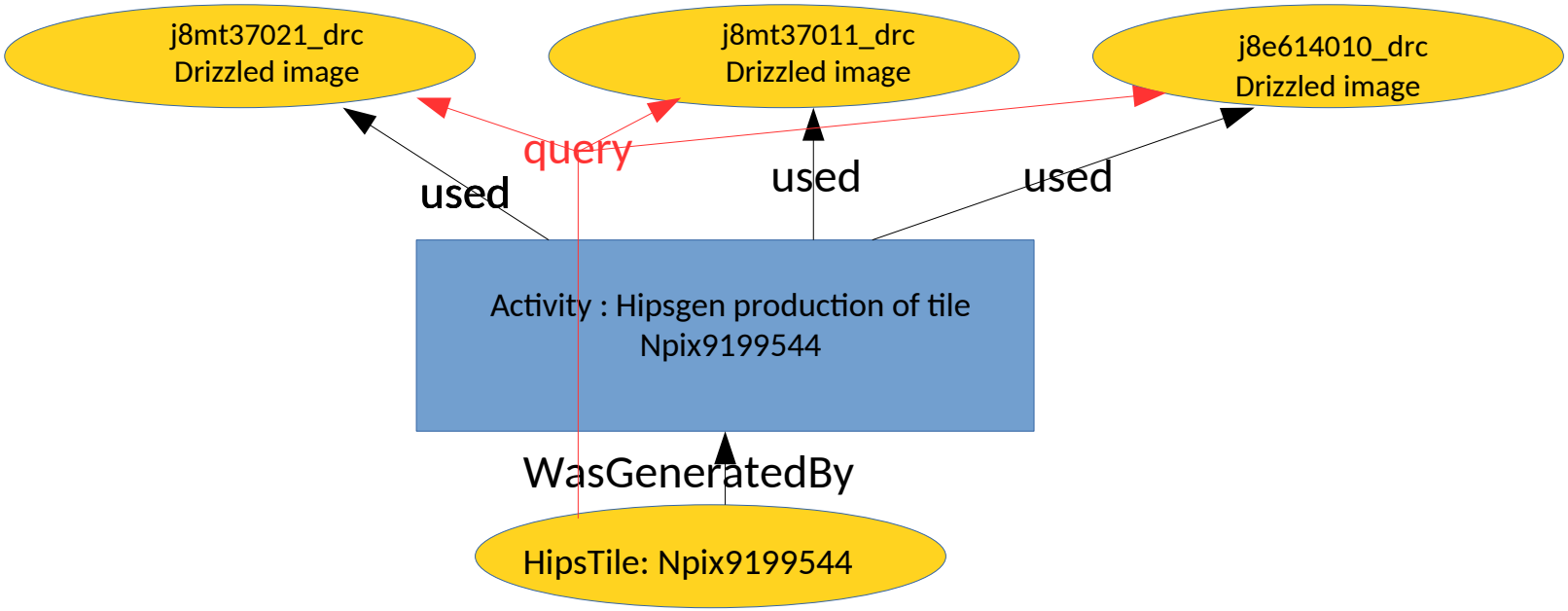
ProvTAP TAP_SCHEMA: parameter table

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.



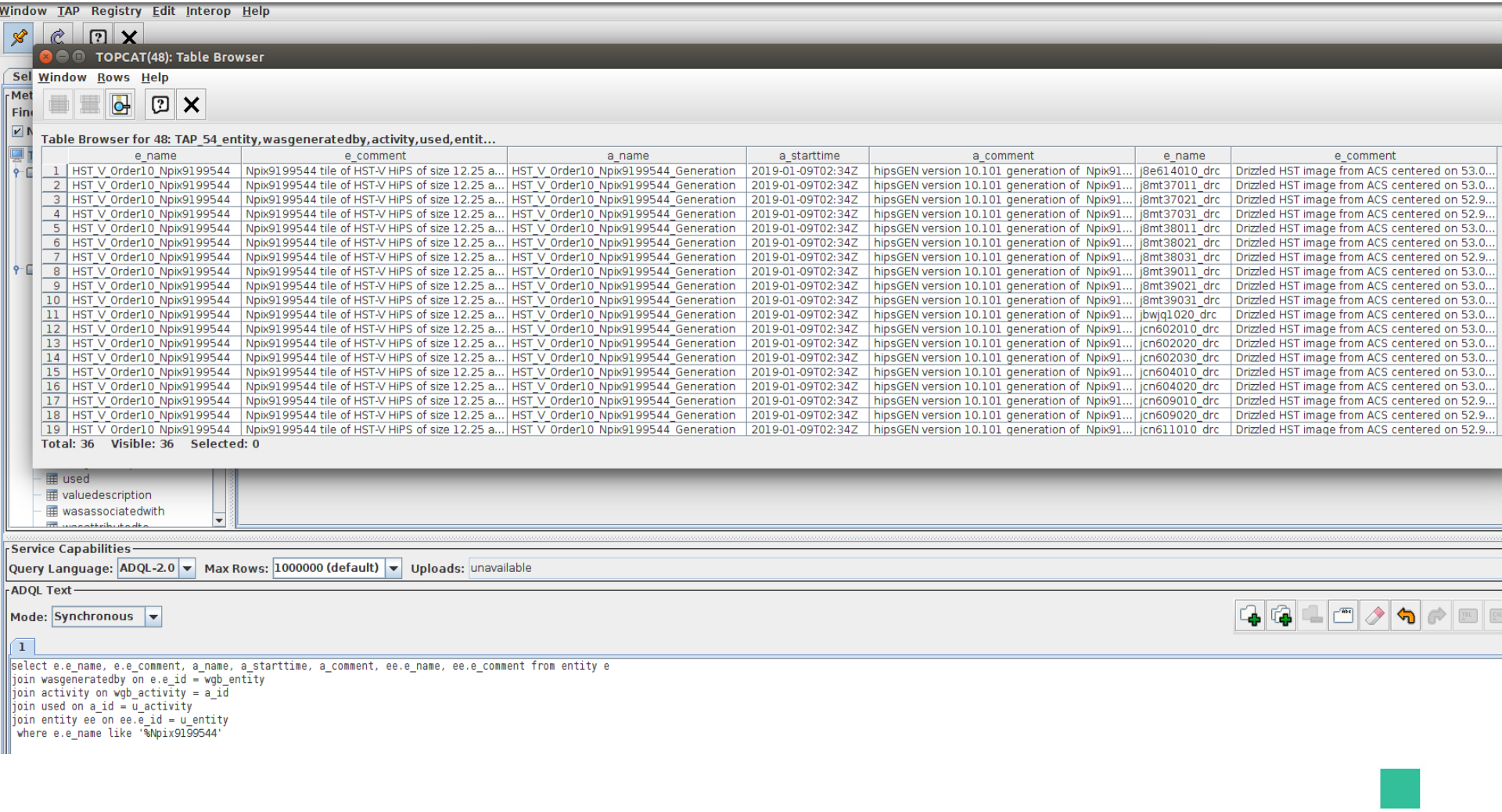
ProvHiPS ADQL query examples :

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.



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

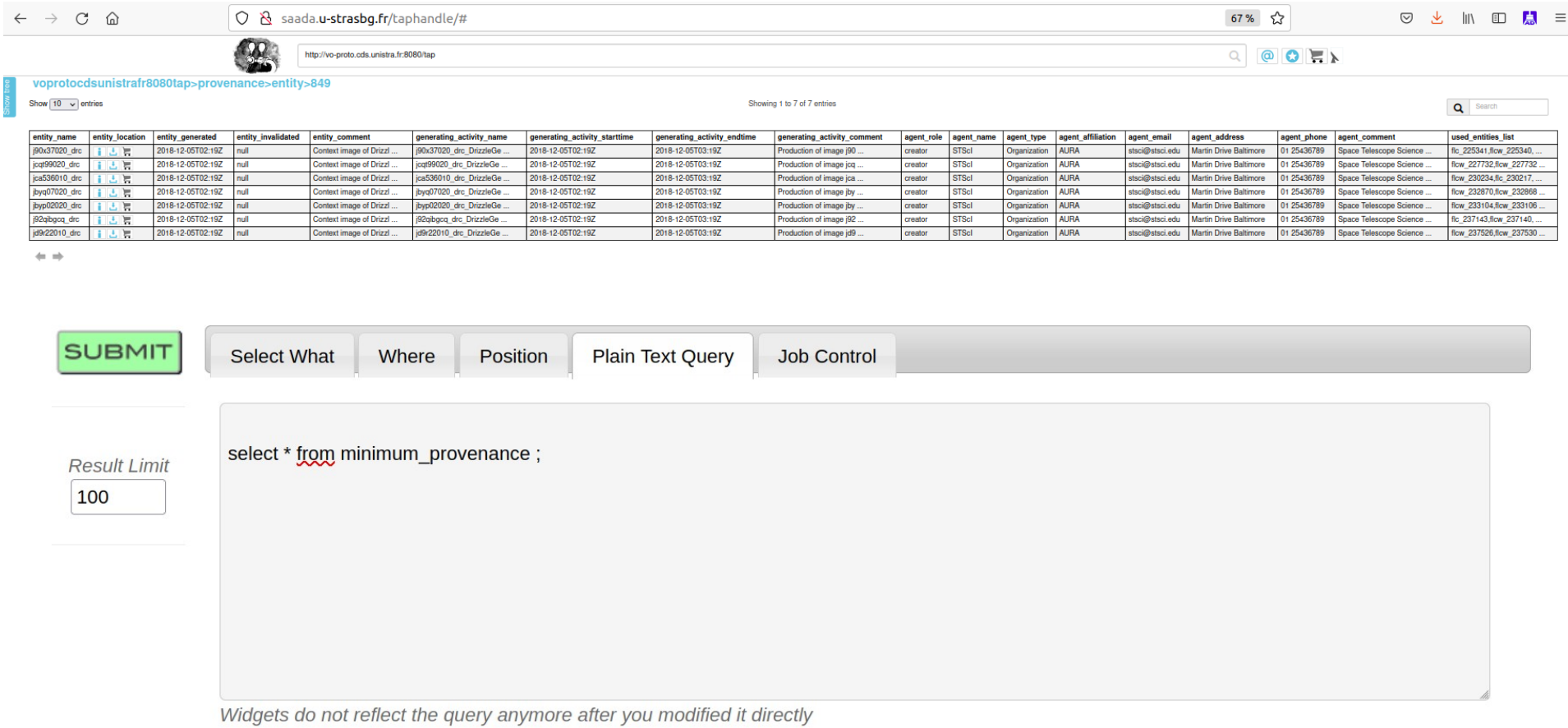
Single step = single table (= join)

- View (in postgres)

```
create view last_step_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



Success and limitations

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

Going Further

- TAP annotation of the simple table query
  (see Mireille's talk)
- Renormalized response (multitable)
- Instance query
  (see DaveMorris/Laurent Michel talk)
  → no simulation here

Renormalized solution


```
<VOTABLE version="1.3" xsi:schemaLocation="http://www.ivoa.net/xml/VOTable/v1.3 http://www.ivoa.net/xml/VOTable/v1.3">
 <RESOURCE type="results">
  <TABLE ID="e" name="entity" utype="voprov:Entity">
   <FIELD ID="eid" arraysize="*" datatype="char" name="e_id" ucd="meta.id" utype="voprov:Entity.id"/>
   <FIELD arraysize="*" datatype="char" name="e_name" ucd="meta.title" utype="voprov:Entity.name"/>
   <FIELD arraysize="*" datatype="char" name="e_comment" ucd="meta.description" utype="voprov:Entity.comment"/>
   <DATA>
    <TABLEDATA>
     <TR>
      <TD>e_1</TD>
      <TD>HST_V_Order10_Npix9199544</TD>
      <TD>
       Npix9199544 tile of HST-V HiPS of size 12.25 and including position 53.0124731662895 and -27.8250294208907
      </TD>
     </TR>
     <TR>
      <TD>e_1000</TD>
      <TD>jcn604010_drc</TD>
      <TD>
       Drizzled HST image from ACS centered on 53.02167776837-27.8496587892353 in spectral band WAVE MIN to WAVEMAX around target ANY
      </TD>
     </TR>
     <TR>
      <TD>e_1001</TD>
      <TD>jcn602030_drc</TD>
      <TD>
       Drizzled HST image from ACS centered on 53.0216928281361-27.8496708083596 in spectral band WAVE MIN to WAVEMAX around target ANY
      </TD>
     </TR>
     <TR>
      <TD>e_1002</TD>
      <TD>jcn602020_drc</TD>
      <TD>
       Drizzled HST image from ACS centered on 53.0216868548142-27.8496543123239 in spectral band WAVE MIN to WAVEMAX around target ANY
      </TD>
     </TR>
     <TR>
      <TD>e_1003</TD>
      <TD>jcn602010_drc</TD>
      <TD>
       Drizzled HST image from ACS centered on 53.0216840785274-27.8496558628425 in spectral band WAVE MIN to WAVEMAX around target ANY
      </TD>
     </TR>
     <TR>
      <TD>e_1004</TD>
      <TD>jbwjq1020_drc</TD>
      <TD>
       Drizzled HST image from ACS centered on 53.0237936758097-27.7961925580348 in spectral band WAVE MIN to WAVEMAX around target Q+A.ACS.CTR.2
      </TD>
     </TR>
     <TR>
      <TD>e_1005</TD>
      <TD>j8mt39031_drc</TD>
      <TD>
       Drizzled HST image from ACS centered on 53.004998675959-27.8234268251943 in spectral band WAVE MIN to WAVEMAX around target ANY
      </TD>
     </TR>
    </TABLEDATA>
   </DATA>
  </TABLE>
  <TABLE ID="a" name="activity" utype="voprov:Activity">
   <FIELD ID="aid" arraysize="*" datatype="char" name="a_id" ucd="meta.id" utype="voprov:Activity.id"/>
   <FIELD arraysize="*" datatype="char" name="a_name" ucd="meta.title" utype="voprov:Activity.name"/>
   <FIELD arraysize="*" datatype="char" name="a_starttime" ucd="time.start" utype="voprov:Activity.startTime"/>
   <FIELD arraysize="*" datatype="char" name="a_comment" ucd="meta.description" utype="voprov:Activity.comment"/>
```


HiPS Tile entity
And used drizzled
HST images entities

Renormalized solution

```
</TABLE>
-<TABLE ID="a" name="activity" utype="voprov:Activity">
  <FIELD ID="aid" arraysize="*" datatype="char" name="a_id" ucd="meta.id" utype="voprov:Activity.id"/>
  <FIELD arraysize="*" datatype="char" name="a_name" ucd="meta.title" utype="voprov:Activity.name"/>
  <FIELD arraysize="*" datatype="char" name="a_starttime" ucd="time.start" utype="voprov:Activity.startTime"/>
  <FIELD arraysize="*" datatype="char" name="a_comment" ucd="meta.description" utype="voprov:Activity.comment"/>
 -<DATA>
  -<TABLEDATA>
   -<TR>
     <TD>a_1</TD>
     <TD>HST_V_Order10_Npix9199544_Generation</TD>
     <TD>2019-01-09T02:34Z</TD>
    -<TD>
       hipsGEN version 10.1o1 generation of Npix9199544tile of HSTV HiPS
     </TD>
    </TR>
  </TABLEDATA>
 </DATA>
</TABLE>
-<TABLE name="used" utype="voprov:Used">
 -<GROUP name="foreinKey" ref="e">
   <FIELDref ref="ue"/>
   <FIELDref ref="eid"/>
  </GROUP>
 -<GROUP name="foreinKey" ref="a">
   <FIELDref ref="ua"/>
   <FIELDref ref="aid"/>
  </GROUP>
  <FIELD ID="ue" arraysize="*" datatype="char" name="u_entity" ucd="meta.id" utype="voprov:Used.entity"/>
  <FIELD ID="ua" arraysize="*" datatype="char" name="u_activity" ucd="meta.id" utype="voprov:Used.activity"/>
 -<DATA>
  -<TABLEDATA>
   -<TR>
     <TD>e_1000</TD>
     <TD>a_1</TD>
    </TR>
   -<TR>
     <TD>e_1001</TD>
     <TD>a_1</TD>
    </TR>
   -<TR>
     <TD>e_1002</TD>
     <TD>a_1</TD>
    </TR>
   -<TR>
     <TD>e_1003</TD>
     <TD>a_1</TD>
    </TR>
   -<TR>
     <TD>e_1004</TD>
     <TD>a_1</TD>
    </TR>
   -<TR>
     <TD>e_1005</TD>
     <TD>a_1</TD>
    </TR>
  </TABLEDATA>
 </DATA>
</TABLE>
-<TABLE name="wasgeneratedby" utype="voprov:WasGeneratedBy">
 -<GROUP name="foreinKey" ref="e">
   <FIELDref ref="wgbe"/>
   <FIELref ref="eid"/>
  </GROUP>
 -<GROUP name="foreinKey" ref="a">
   <FIELDref ref="wgba"/>
   <FIELref ref="aid"/>
  </GROUP>
  <FIELD ID="wgbe" arraysize="*" datatype="char" name="wgb_entity" ucd="meta.id" utype="voprov:WasGeneratedBy.entity"/>
  <FIELD ID="wgba" arraysize="*" datatype="char" name="wgb_activity" ucd="meta.id" utype="voprov:WasGeneratedBy.activity"/>
 -<DATA>
  -<TABLEDATA>
   -<TR>
     <TD>e_1</TD>
     <TD>a_1</TD>
     <TD/>
    </TR>
  </TABLEDATA>
 </DATA>
</TABLE>
</RESOURCE>
```

Activity producing the tile (hipsgen)

VOTable Foreign Keys mechanism Applied to « used »

Used table contains relationships between activities and used entities

VOTable Foreign Keys mechanism Applied to « wasgeneratedby »

Wasgeneratedby table contains relationships between activities and generated entities

Renormalisation

- How to generate queries doing that ?
  - « User Defined Function » for reformatting ?
- Need to have clients making full usage of multitable and foreign key mechanisms

ProvTAP annotation

```
−<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 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)