Advanced ways to find and retrieve data in the LTA

There are some useful ways to find and retrieve your data in the LTA that might not be immediately obvious. This page explains some of the more advanced options you have.

Queries

You can use colons in numeric queries, to select ranges. This will for example give all
observations and pipelines that have a SAS/Observation ID in the range from 432000 to
432190:



In textual entries, wildcards can be used.

You can put a list of SAS/Observation IDs in the query:

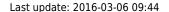


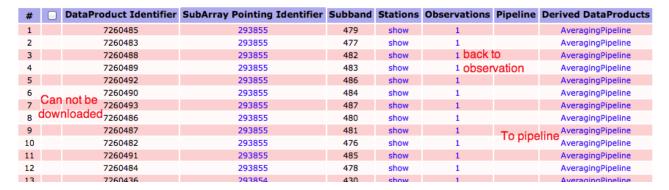
Viewing data

When you are looking at the results of a query you might see something like this:



This means that the observation is known in the LTA, it knows what data was produced, the produced data was not archived, but further processing happened on the raw data and the results of some of those pipelines were archived. If you click on the zero, you will see something like this:





This allows you to navigate from a pipeline back to the original observation, or from the observation to any pipelines that have run on the raw data.

Retrieving data

 You can retrieve data on the Observation and Pipeline level, you don't have to select all files individually.

		Observation	Observing	Antenna Set	Instrun
#		Id	Mode		Filte
	_				
1	⋖	146448	Interferometer	HBA Dual Inner	110-190
2		146447	Interferometer	HBA Dual Inner	110-190
3	⋖	146446	Interferometer	HBA Dual Inner	110-190
4		146445	Interferometer	HBA Dual Inner	110-190
5	⋖	146444	Interferometer	HBA Dual Inner	110-190
6	\checkmark	146443	Interferometer	HBA Dual Inner	110-190
7		146442	Interferometer	HBA Dual Inner	110-190
8	\checkmark	146441	Interferometer	HBA Dual Inner	110-190
9	⋖	146456	Interferometer	HBA Dual Inner	110-190
10	\checkmark	146455	Interferometer	HBA Dual Inner	110-190
11		146454	Interferometer	HBA Dual Inner	110-190
12		146453	Interferometer	HBA Dual Inner	110-190
13		146452	Interferometer	HBA Dual Inner	110-190

• If you have a query with more than 1000 results, you can open the multiple pages each in a separate tab/window.

Observation 1001 to 1100 (showing 100 of total 1156) -

With the small triangle next to a list, you can fold or unfold the list to get a better overview.

Folded entries

Observation 1 to 100 (showing 100 of total 1156)

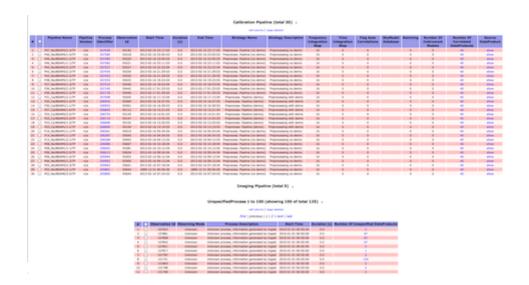
Averaging Pipeline 1 to 100 (showing 100 of total 4060)

Calibration Pipeline (total 30)

Imaging Pipeline (total 0)

UnspecifiedProcess 1 to 100 (showing 100 of total 125)

Unfolded entries



DBView

There is a server that gives the option to run your own queries on the database http://lofar-dbview.target.rug.nl/

A useful query might be this one, that gives you all files for a certain Obs Id (SAS VIC tree ID).

```
SELECT fo.URI, dp."dataProductType", dp."dataProductIdentifier",
    dp."processIdentifier"
FROM AWOPER."DataProduct+" dp,
        AWOPER.FileObject fo,
        AWOPER."Process+" pr
WHERE dp."processIdentifier" = pr."processIdentifier"
    AND pr."observationId" = '123456'
    AND fo.data_object = dp."object_id"
```

```
AND dp."isValid" > 0
```

In this '123456' should be replaced with the Obs Id of an Observation/Pipeline you're looking for.

AstroWise Python Interface

There is also a python interface to the LTA. With this you can also script some advanced queries. This for example gives all observations on a certain patch of the sky:

```
# python code
from pprint import pprint
from common.database.Context import context
from awlofar.main.aweimports import Observation, Pointing, SubArrayPointing
result = {}
for project in sorted(context.get_projects()) :
    print "Project %(project)s" % vars()
    ok = context.set project(project)
    # do your query
    obs ids = set()
    query = (Pointing.rightAscension > 95) & \
            (Pointing.rightAscension < 105) & \
            (Pointing.declination > 20)
            (Pointing.declination
                                    < 30)
    print "Total Pointings %d" % len(query)
    for pointing in query :
        print "Pointing found RA %f DEC %f" % (pointing.rightAscension,
pointing.declination)
        query subarr = SubArrayPointing.pointing == pointing
        for subarr in query subarr:
            query obs = Observation.subArrayPointings.contains(subarr)
            for obs in query obs :
                obs ids.add(obs.observationId)
    result[project] = sorted(list(obs ids))
    print result[project]
pprint(result)
```

From:

https://www.astron.nl/lofarwiki/ - LOFAR Wiki

Permanent link:

https://www.astron.nl/lofarwiki/doku.php?id=public:lta_tricks&rev=145725744

Last update: 2016-03-06 09:44

