



Data Mining with the Product Browser and Data Management in HIPE

Bernhard Schulz
NHSC/IPAC





The Product Browser

- The Product Browser can be found in “Window” as a view under “Data Access”.
- The main panels are “Data Source”, “Search parameters”, and “Query Result Display”
- Data Source shows all the pools in your storage directory and the myHsa pool.
- On the right some search criteria can be chosen.
- The “On-line” button enables querying the HSA through the internet if an observation isn’t present locally in the MyHSA pool.

On-line
Off-line
selector

Data Source (Pools)

Query Result Display

HIPE - saveProduct

File Edit Run Pipelines Scripts Window Tools Help

Editor Product Browser x Navigator Save Products to Pool Export Here

Observations Products Metadata Free Metadata Search Parameters

Data Source

MyHSA

On-line Off-line

Local Pools

DestriperL2Degl

DestriperTest

GP299

Hipe11PhotometryCheck

MarsMaps

Run

Search parameters

Show all versions

Observation Id (obsid) ==

Instrument (instrument) ==

Operational Day (odNumber) ==

+

125 results found Query Result # of Results 20

Pool	obsid	odNumber	tag	object	total size	aot	obsMode
------	-------	----------	-----	--------	------------	-----	---------

History Log Console x

```
HIPE>
HIPE>
HIPE>
HIPE>
HIPE>
```

beschulz Jython Interpreter 100%



Pool Content and Retrieval

- Find and select the newly created pool “myTestPool”.
- Hit the “Run” button.
- The query result list will show one line representing the observation that was just saved.
- Double-clicking the observation will retrieve the observation context.
- Note the more complex script equivalent in the Console View.

The screenshot shows the HIPE software interface. The 'Product Browser' panel on the left lists several pools, with 'myTestPool' selected. An orange arrow points to this entry. Below the browser is a 'Run' button, also indicated by an orange arrow. The 'Query Result' table shows one result for 'myTestPool' with observation ID '1342179050'. An orange arrow points to this row, with the text 'double-click' written below it. The 'Console' view at the bottom shows a complex script for retrieving the observation context. An orange arrow points to the console output. On the right, the 'Observations' panel shows a list of variables, with 'obs' highlighted. A green arrow points to 'obs' with the text 'variable appears in RAM after double-click'.

Pool	obsid	odNumber	tag	object	total size	aot
myTestPool	1342179050	42.obs	1342179050	my l... M74	935942440	Photometer

```

HIPE> QUERY_RESULT =
ProductStorage([PoolManager.getPool('myTestPool')]).select(herschel.ia.p
al.query.MetaQuery(herschel.ia.obs.ObservationContext, "p", "1"))
HIPE> # Added variable: QUERY_RESULT
HIPE> # Added variable: QUERY_RESULT
HIPE> # Added variable: selected
HIPE> obsid_1342179050_1 = QUERY_RESULT[0].product
HIPE>
    
```



MyHSA Access

- To access observations in the **local** myHSA pool, select MyHSA
- Ensure “**Off-line**” is selected.
- Hit the “Run” button.
- Note the presence of the earlier imported observations from SPIRE, HIFI and PACS.

The screenshot shows the HIPE software interface with the following components:

- Product Browser:** Shows a list of Data Sources: MyHSA, DestriperL2Degl, and DestriperTest. The 'MyHSA' source is selected. Below it, the 'Off-line' radio button is selected.
- Run Button:** A button labeled 'Run' is visible below the Product Browser.
- Query Result Table:** A table with 125 results found. The columns are Pool, obsid, odNumber, tag, object, total size, aot, obsMode, and instrument. The table lists observations from pools hsa, including objects like M 101, NGC 1365, Antennae, M74, ngc 6946, and M83. The instrument column shows PACS, HIFI, and SPIRE.
- Console:** A terminal window at the bottom showing the execution of a query: `HIPE> # Added variable: selected`, `HIPE> obsid_1342179050_1 = QUERY_RESULT[0].product`, `HIPE> QUERY_RESULT1 = ProductStorage([PoolManager.getPool('myhsa')]).select(herschel.ia.pal.query.MetaQuery(herschel.ia.obs.ObservationContext, "p", "1"))`, and several `# Added variable: QUERY_RESULT1` messages.



On-line Archive Access Example 1

- To access observations in the HSA, select MyHSA
- Ensure “On-line” is selected.
- Make sure an obsid is given or search criteria that sufficiently limit the number of results (limit in 5000).
- Hit the “Run” button.

The screenshot shows the Product Browser interface with the following components:

- Data Source:** MyHSA is selected. The "On-line" radio button is selected.
- Local Pools:** A list of pools including 1342186799, 1342214364, 1342219041, DestriperL2Degl, DestriperTest, and GP299.
- Search parameters:** Observation Id (obsid) is set to 1342203626.
- Run button:** A button labeled "Run" is visible.
- Query Result Table:** Shows 1 result found. The table has columns: Pool, obsid, odNumber, tag, object, total size, aot, obsMode, inst.
- Console:** Displays log messages from PoolManager.getPool() for various pools.

Pool	obsid	odNumber	tag	object	total size	aot	obsMode	inst
hsa	1342203626	467		SDSSJ0353...	449352857	Photometer	Small Map	SPIRE

```
PoolManager.getPool('rcw120_destriper_demo'),
PoolManager.getPool('spire_cal_9_0'),
PoolManager.getPool('rcw120DestrTst'),
PoolManager.getPool('temp_pal'), PoolManager.getPool('SNR1987A'),
PoolManager.getPool('ngc5134'),
```



On-line Archive Access Example 2

- Let's find all parallel mode PACS observations on observational days 600 and 605.
- Switch to "On-line"
- Enter "PACS" into Instrument.
- Use the + sign to add two more "Observational Day" search criteria and one "Observation Mode".
- Select appropriate logical operators and numbers.
- Enter "Parallel Mode" into Observation Mode
- Hit "Run".
- Select "all" to show the full list of results.

The screenshot shows the HIPE software interface with the following search parameters:

- On-line (selected)
- Instrument: PACS
- Operational Day (odNumber) >= 600
- Operational Day (odNumber) <= 605
- Observation Mode (obsMode) == Parallel Mode

The results table shows 9 results found:

Pool	obsid	odNumber	tag	object	total size	aot	obsMode	instrume
hsa	1342212300	600		DC300-17	5222699...	Parallel M...	Parallel M...	PACS
hsa	1342212302	600		V1	1551557...	Parallel M...	Parallel M...	PACS
hsa	1342212408	605		V4	1607408...	Parallel M...	Parallel M...	PACS
hsa	1342212301	600		V1	1613639...	Parallel M...	Parallel M...	PACS
hsa	1342212407	605		V4	1546483...	Parallel M...	Parallel M...	PACS
hsa	1342212378	604		V3	1549685...	Parallel M...	Parallel M...	PACS
hsa	1342212358	603		V3	1609189...	Parallel M...	Parallel M...	PACS
hsa	1342212379	604		V3	1610537...	Parallel M...	Parallel M...	PACS

The console shows the following output:

```

HIPE> # Added variable: QUERY_RESULT6
HIPE> # Added variable: QUERY_RESULT6
HIPE> # Added variable: QUERY_RESULT6
HIPE> # Added variable: QUERY_RESULT6
HIPE>
    
```



On-line Archive Access Example 3

Here we find all observations between RA = 50..70 deg and DEC = 1..2 deg.

Note that the additions of search criteria have always more logical operators available.

The screenshot shows the Product Browser interface with the following components:

- Search parameters:**
 - Show all versions:
 - Observation Id (obsid) ==
 - Instrument (instrument) ==
 - Operational Day (odNumber) ==
 - dec > 1.0
 - dec < 2.0
 - ra > 50.0
 - ra < 70.0
- Run button:** A button labeled "Run" with a star icon.
- Query Results:**
 - 15 results found
 - Query Result: QUERY_RESULT1
 - # of Results: 20

Pool	obsid	od...	tag	object	total size	aot	obsMode	in
hsa	1342203626	467		SDSSJ0353...	449352857	Photometer	Small Map	SPI
hsa	1342238943	1003		IRASF0345...	1171775...	Line Spec...	Pointed	PA
hsa	1342239848	1022		KPAIRSMAL...	407434472	Photometer	Small Map	SPI
hsa	1342250111	1198		KPAIRSMAL...	902221611	Photometer	Scan map	PA
hsa	1342250112	1198		KPAIRSMAL...	878748227	Photometer	Scan map	PA
hsa	1342250113	1198		KPAIRSMAL...	878681720	Photometer	Scan map	PA
hsa	1342250114	1198		KPAIRSMAL...	878690314	Photometer	Scan map	PA
hsa	1342215978	668		SDSSJ0353...	1821402...	Photometer	Scan map	PA
hsa	1342215979	668		SDSSJ0353...	1573149...	Photometer	Scan map	PA
- Console:**

```
PoolManager.getPool('pipeline-out'),
PoolManager.getPool('h_1342190183'),
PoolManager.getPool('spirephot_ngc6946_nom_rp'),
PoolManager.getPool('OD424_Temple2'),
PoolManager.getPool('Neptune_cal_11_0'),
```



Managing Observations and Pools

- Pools can be created, inspected, renamed, moved, exported, and deleted.
- The **Product Browser** helps to easily visualize and manage the content of pools.
- The myHSA pool is special:
 - MyHsa acts as a **local cache** between the HSA at the Herschel Science Center and your HIPE session on your computer.
 - All pools can be written to, except for the “myHsa” pool.
 - The other pools are intended for saving results that are different from those in the HSA, i.e. products from **your own processing and analysis**.
 - The data transfer between the HSA and the myHSA pool is optimized.
 - Saving observations that were retrieved straight from the HSA into a pool, is less efficient.
- The Product Browser can be used to retrieve observations from the archive by querying myHsa with the On-line option on.



Managing Observations

- Find all observations in all local pools.
- De-select “MyHSA” by clicking on another pool.
- Select “Local Pools”.
- Hit “Run”.
- Right click on an observation.
- The menu allows to:
 1. load an observation context into RAM (create variable),
 2. Delete an observation from a pool (Remove product....),
 3. Export FITS (not useful because it is just a context.).

The screenshot shows the HIPE software interface with the following components:

- Product Browser:** Shows a tree view of data sources. 'MyHSA' is selected, and 'Local Pools' is checked. Other pools include DestriperL2Degl, DestriperTest, GP299, and Hipe11PhotometryChe.
- Search parameters:** Fields for Observation Id (obsid), Instrument (instrument), and Operational Day (odNumber).
- Table:** Displays 286 results found. The table has columns: Pool, obsid, odNumber, tag, object, total size, aot, and obsM. One row is highlighted in green.
- Context Menu:** A right-click menu is open over the highlighted row, showing options: 'Create variable', 'Remove product from storage/pool', and 'Export FITS...'. An orange arrow points to the 'Remove product from storage/pool' option.
- Console:** Shows the command history:


```
HIPE> # Added variable: QUERY_RESULT
HIPE> # Added variable: QUERY_RESULT
HIPE> # Added variable: selected
HIPE> # Added variable: selected
HIPE>
```



Managing Pools

- Select a pool and right-click.
- Pools can be **created, renamed, moved to another storage directory, deleted, and exported.**
- Moving will move the pool to another directory in your file system.
- Export will copy the contents of a pool into a zip file, so it can be sent to collaborators and other pundits.

HIPE – saveProduct

File Edit Run Pipelines Scripts Window Tools Help

Editor Navigator Product Browser x Save Products to Pool Export Here

Observations Products Metadata Free Metadata

Data Source

- Hipe11PhotometryChe
- MarsMaps
- NeptuneCombined
- Neptune_cal_11_0
- OD424Temple2
- OD424_Temple2
- SKglitchTst
- SNR1978A
- SNR1987A
- ThreePublicObs

Search parameters

Show all versions

Observation Id (obsid) ==

Instrument (instrument) ==

Operational Day (odNumber) ==

Run

286 results found Query Result QUERY_RESULT # of Results all

Pool	obsid	odNumber	tag	object	total size	aot	obsM
Neptune_cal_11_0	1342254501	1271	1342254...	Neptune	30891804	Photometer	Small
Neptune_cal_11_0	1342254501	1271	1342254...	Neptune	29474053	Photometer	Small
Neptune_cal_11_0	1342254501	1271	1342254...	Neptune	30958355	Photometer	Small

History Log Console x

```
HIPE>
HIPE>
HIPE>
HIPE>
HIPE>
```

beschutz Jython Interpreter 100%

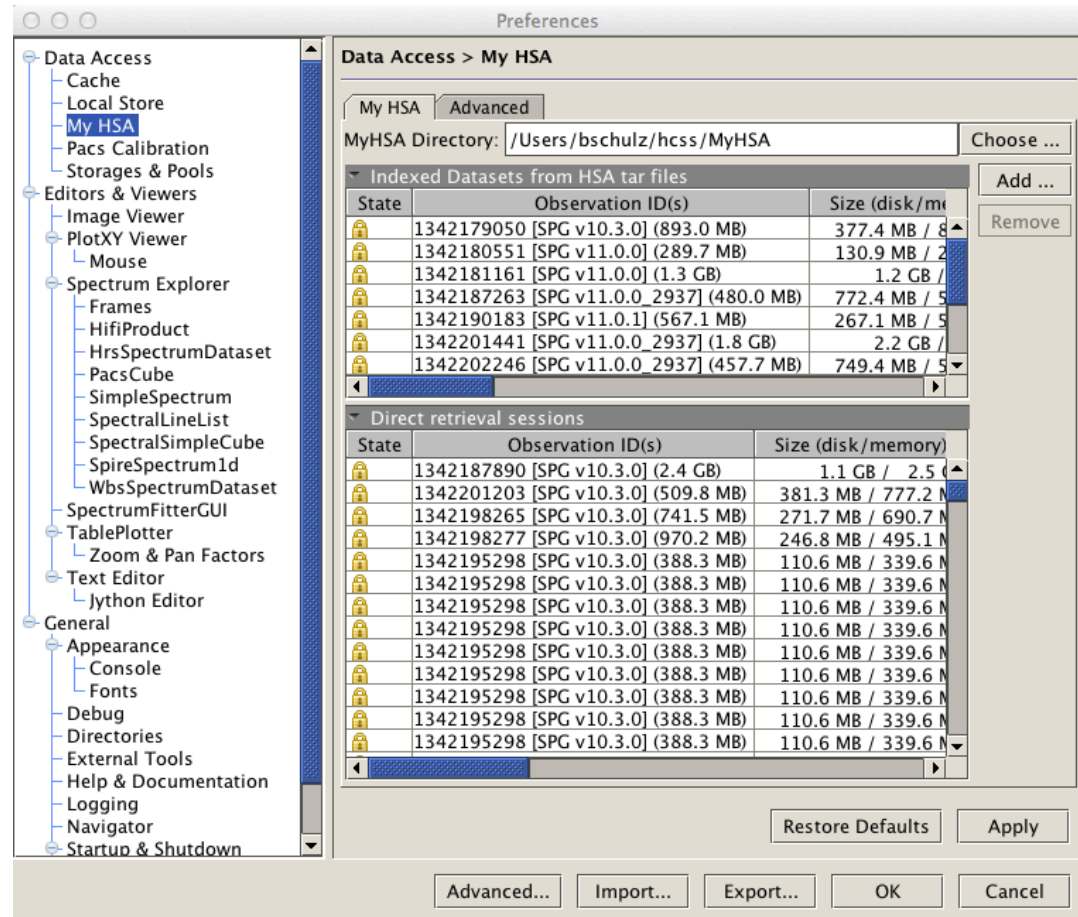


Removing data from myHsa

To control data volume in myHsa data can be deleted in two places depending on their origin:

- 1) Data loaded from unzipped tarballs can be deleted in the upper panel of the myHsa preferences panel.
- 2) Data loaded by online access using lazy-loading (e.g. HUI, getObservation() with save=True, any loading with hcss.ia.pal.pool.hsa.on_demand = True) via the lower session oriented panel.

Select item and click “Remove”
Note that lazy-loaded products can not be removed by observation, only by session.





Getting Data Cheatsheet

Tool used	Effect	Equivalence
obs = getObservation(<obsid>)	Loads observation context into RAM. First looks for observation in local pools, then in myHsa pool.	In Product Browser select local pool, hit return, and double-click on a line in the results table." but there is no control over where the observation comes from. If the observation is found first in a pool it will be loaded from there.
obs = getObservation(path="<path and name of top directory of unpacked tarball data>")	Loads observation into RAM. The observation will also appear in myHsa pool from now on unless it is a reprocessed observation or an exported observation. Then it will appear as a local pool.	In Navigator double-click on xml file of unpacked HSA tarball.
obs = getObservation(<obsid>, useHsa=True)	Loads observation context into RAM. Always loads from online HSA even if it is present in either local pools or myHsa pool. The observation context and all lazy loaded products are stored in myHsa only if hcsc.ia.pal.pool.hsa.on_demand = True.	In Product Browser select myHsa pool, hit return, and double-click on a line in the results table with HSA on-line switch on.
obs = getObservation(<obsid>, useHsa=True, save=True)	Loads observation context into RAM. Always loads from online HSA even if it is present in either local pools or myHsa. The entire observation context is stored in myHsa pool.	No real equivalence: Similar to "In Product Browser select myHsa pool, hit return, and double-click on a line in the results table with HSA on-line switch on", but then the entire observation is only loaded if the observation is saved into a pool.
In Navigator double-click on xml file of unpacked HSA tarball.	Loads observation into RAM. The observation will also appear in myHsa pool from now on unless it is a reprocessed observation or an exported observation. Then it will appear as a local pool.	Equivalent to obs = getObservation(path="<path and name of top directory of unpacked tarball data>")
In Product Browser select local pool, hit return, and double-click on a line in the results table.	Loads the associated observation context into RAM from a local pool.	obs = getObservation(<obsid>, poolName=<name of pool>)
In Product Browser select myHsa pool, hit return, and double-click on a line in the results table.	Loads the associated observation context into RAM. If the HSA on-line switch is on, the product is loaded from the HSA if not present in myHsa. If hcsc.ia.pal.pool.hsa.on_demand = True all lazy loaded products are also stored in the myHsa pool.	Similar to obs = getObservation(<obsid>, useHsa=True) but online HSA access is not forced.
In Product Browser select myHsa pool and all local pools, hit return, and double-click on a line in the results table.	Loads observation context into RAM.	Similar to "obs = getObservation(<obsid>)" except that there is no ambiguity where the observation came from.
obs = getObservation(<obsid>, poolName=<name of pool>)	Loads observation context into RAM. Looks for observation only in specified pool. To look into myHsa the pool name can be any upper-/lower-case version of "myhsa".	"In Product Browser select local pool, hit return, and double-click on a line in the results table." or "In Product Browser select myHsa pool, hit return, and double-click on a line in the results table."