

Firefly Support

TAP

Datalink

ObsCore

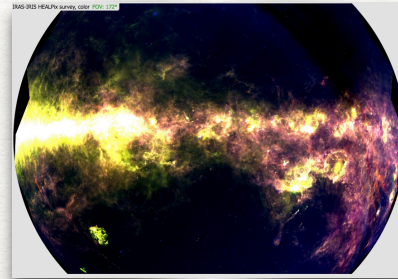
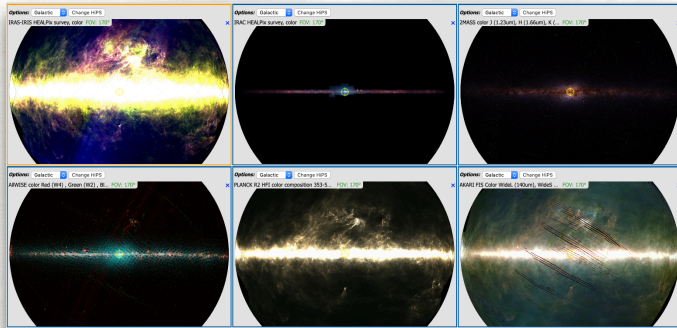
s_region

MOC

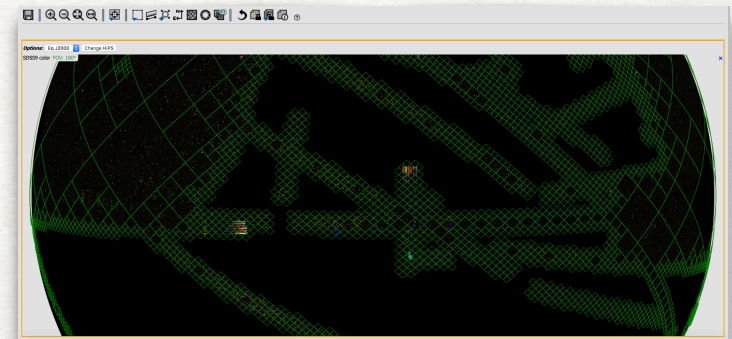


Adding VO Standards

- May 2018: HiPS

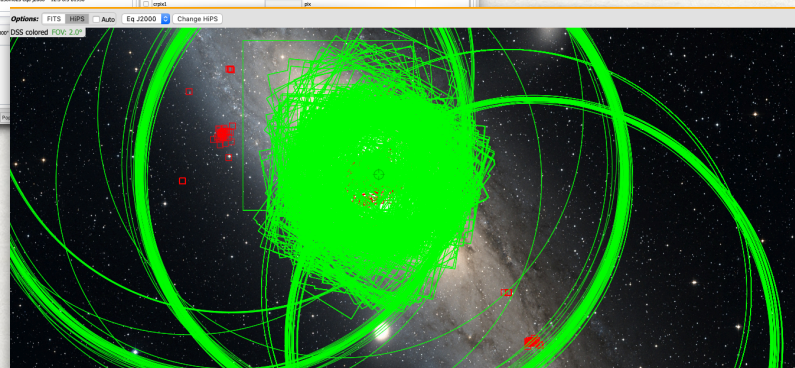
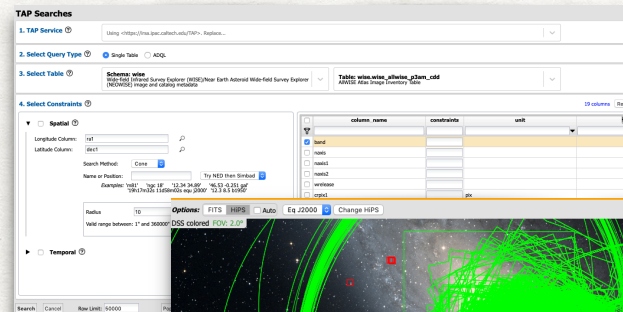


- MOC - Fall 2018



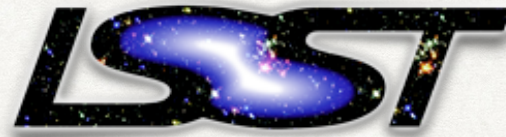
- Spring 2019

- ▶ TAP
- ▶ ObsCore
- ▶ DataLink (partial)
- ▶ s_region visualization



Drivers

- Driven by LSST closeout



- Interest by



- Provided archive flexibility

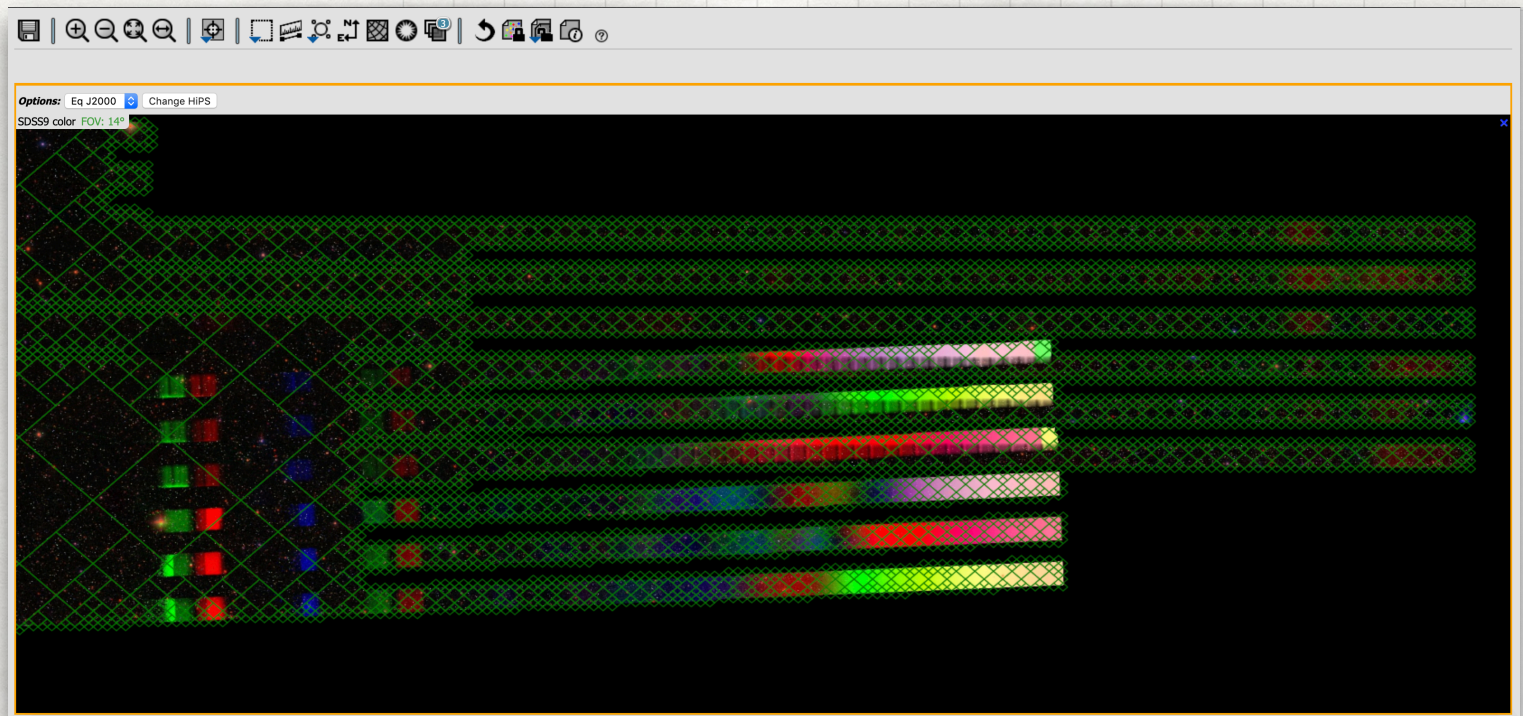
This talk

- MOC
- TAP
- ObsCore
- s_region
- DataLink

for each

- What we did
- Comments or Lessons learned
- Maybe Wishes

MOC



MOC

Turn on MOC

GET MOC

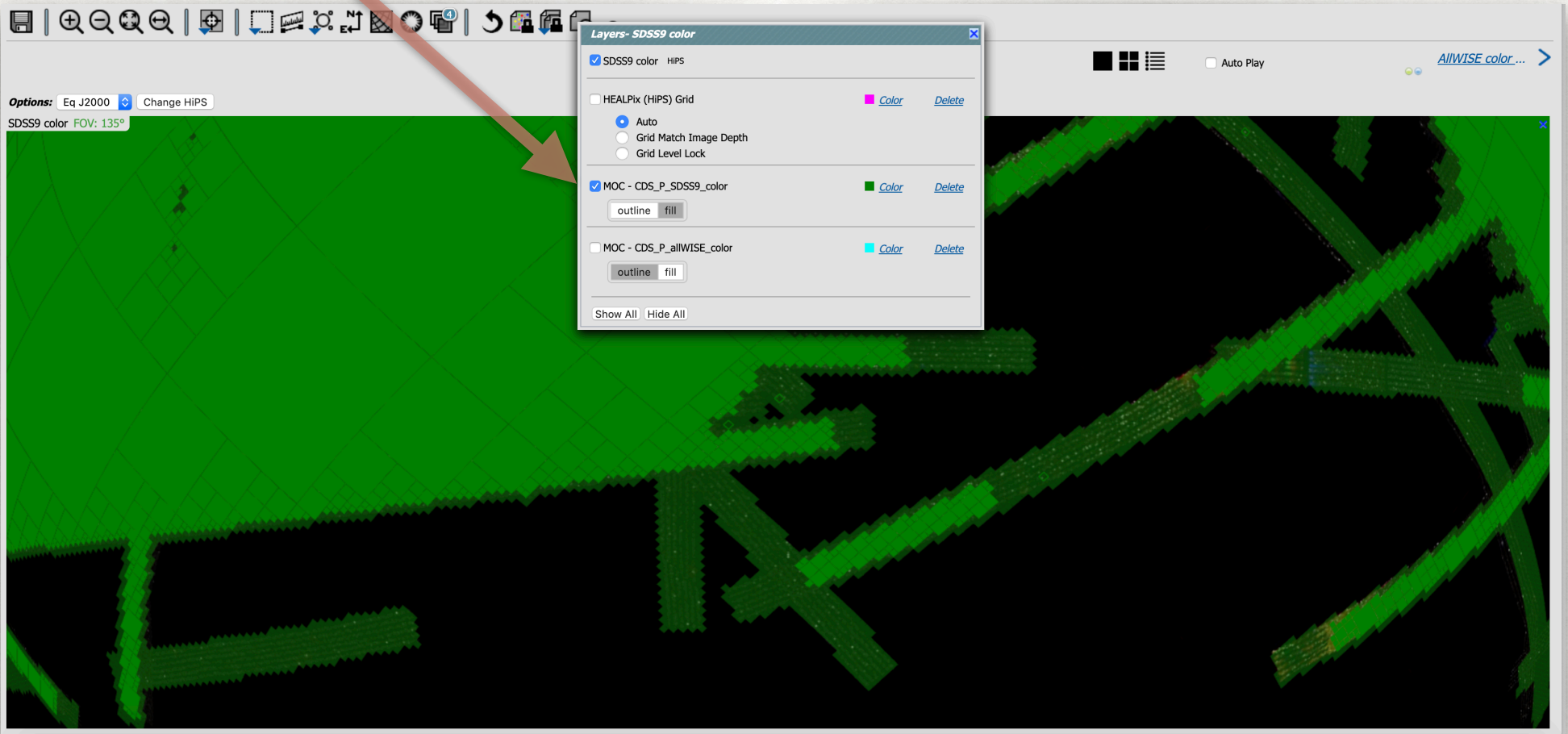
The screenshot displays a web application interface for visualizing a Masked Order Coverage (MOC) grid. The main view shows a star field with a green grid overlay. A settings panel is open, showing the following options:

- SDSS9 color HIPS
 - HEALPix (HIPS) Grid
 - Auto
 - Grid Match Image Depth
 - Grid Level Lock
 - MOC - CDS_P_SDSS9_color
 - outline fill
 - MOC - CDS_P_allWISE_color
 - outline fill

Buttons for [Color](#) and [Delete](#) are visible next to each option. At the bottom of the panel are [Show All](#) and [Hide All](#) buttons. The top of the interface includes a navigation menu, an [Auto Play](#) checkbox, and a [AllWISE color...](#) link.

MOC Fill

FILL



Comments

- All drawing layer controls
- Can read in MOC from other sources
- Firefly can place any MOC over any HiPS (or FITS)
- Small MOC's very straight forward
- Large MOC's like SDSS require significant optimization
- Standard very straight forward
- Probably significant IRSA and NED uses coming



TAP

TAP Searches

1. TAP Service ?

Using <https://irsa.ipac.caltech.edu/TAP>. Replace... ▼

2. Select Query Type ?

Single Table ADQL

3. Select Table ?

Schema: wise
Wide-field Infrared Survey Explorer (WISE)/Near Earth Asteroid Wide-field Survey Explorer (NEOWISE) image and catalog metadata ▼

Table: wise.wise_allwise_p3am_cdd
AllWISE Atlas Image Inventory Table ▼

4. Select Constraints ?

19 columns Reset

Spatial ?

Longitude Column: 🔍

Latitude Column: 🔍

Search Method: ⌵

Name or Position: Try NED then Simbad ⌵

Examples: 'm81' 'ngc 18' '12.34 34.89' '46.53 -0.251 gal'
'19h17m32s 11d58m02s equ j2000' '12.3 8.5 b1950'

Radius: arcseconds ⌵

Valid range between: 1" and 360000"

Temporal ?

<input type="checkbox"/>	column_name	constraints	unit	⚙️
<input checked="" type="checkbox"/>	band			
<input type="checkbox"/>	naxis			
<input type="checkbox"/>	naxis1			
<input type="checkbox"/>	naxis2			
<input type="checkbox"/>	wrelease			
<input type="checkbox"/>	crpix1		pix	
<input type="checkbox"/>	crpix2		pix	
<input checked="" type="checkbox"/>	crval1		deg	
<input checked="" type="checkbox"/>	crval2		deg	
<input type="checkbox"/>	ctype1			
<input type="checkbox"/>	ctype2			
<input type="checkbox"/>	equinox		year	
<input type="checkbox"/>	bunit			
<input type="checkbox"/>	elon		deg	

Search

Cancel

Row Limit:

Populate and edit ADQL

?

TWO MODES Helper

TAP Searches

1. TAP Service

2. Select Query Type Single Table ADQL

3. Select Table Schema: wise_allwise Table: allwise_p3as_psd

4. Select Constraints 45 columns

Spatial

Longitude Column:

Latitude Column:

Search Method:

Name or Position:

m5 resolved by NED
229.64063, 2.08268 Equ J2000 or 15h18m33.75s, +2d04m57.6s Equ J2000

Radius:

Valid range between: 1" and 360000"

Temporal

checkbox	column_name	constraints	unit	ucd
<input checked="" type="checkbox"/>	designation			
<input checked="" type="checkbox"/>	ra		deg	
<input checked="" type="checkbox"/>	dec		deg	
<input checked="" type="checkbox"/>	sigra		arcsec	
<input checked="" type="checkbox"/>	sigdec		arcsec	
<input checked="" type="checkbox"/>	sigradec		arcsec	
<input type="checkbox"/>	glon		deg	
<input type="checkbox"/>	glat		deg	
<input type="checkbox"/>	elon		deg	
<input type="checkbox"/>	elat		deg	
<input type="checkbox"/>	wx		pix	
<input type="checkbox"/>	wy		pix	
<input type="checkbox"/>	cntr			
<input type="checkbox"/>	source_id			
<input type="checkbox"/>	coadd_id			
<input type="checkbox"/>	src			
<input checked="" type="checkbox"/>	w1mpro		mag	
<input checked="" type="checkbox"/>	w1sigmpro		mag	
<input checked="" type="checkbox"/>	w1snr			
<input checked="" type="checkbox"/>	w1rchi2			
<input checked="" type="checkbox"/>	w2mpro		mag	

Row Limit:

CHOOSE SCHEMA & TABLE

- COLUMN SELECTION
- ▶ USING TABLE TOOL
 - ▶ FILTER
 - ▶ SORT
 - ▶ ADD CONSTRAINTS

SEARCH TYPE

TO ADVANCED

TAP UI Attributes

- Requires an Extensive UI
 - For an non-expert user
- Preselect principle columns
- Presort list of schemas, tables, and columns
 - `schema_index`, `table_index`, `column_index`

TWO MODES

Advanced

TAP Searches

1. TAP Service ⓘ

2. Select Query Type ⓘ Single Table ADQL

3. Advanced ADQL ⓘ ADQL edits below will not be reflected in Single Table view

Schema -> Table -> Column

- wise
- neowiser
- wise_allwise
- wise_allsky
- wise_prelim
- twomass
- fp_2mass
- sixx2
- wax
- scal
- ntmass
- tmassr
- seip
- spitzer
- c2d
- astrometry
- public
- swire
- planck
- herschel
- gaia
- cosmos
- ptf
- ztf
- iras
- msx
- akari
- bolocam
- usno
- denis
- correlation
- irts
- blast

ADQL Query:

```
SELECT
designation,ra,dec,sigra,sigdec,sigradec,w1mpro,w1sigmpro,w1snr,w1rchi2,w2mpro,w2sigmpro,w2snr,w2rchi2,w3mpro,w3sigmpro,w3snr,w3rchi2,w4mpro,w4sigmpro,w4snr,w4rchi2,nb,na,w1sat,w2sat,w3sat,w4sat,pmra,sigpmra,pmdec,sigpmdec,
cc_flags,ext_flg,var_flg,ph_qual,moon_lev,w1nm,w1m,w2nm,w2m,w3nm,w3m,w4nm,w4m FROM allwise_p3as_psd WHERE CONTAINS(POINT('ICRS', ra, dec),CIRCLE('ICRS', 229.64063, 2.08268, 0.08333333333333333))=1
```

Schema Browser Hints

Click on a Table node to insert a default SELECT statement of that table into the Query input box.
Click on a Column node to insert the column's name at the Query input box's cursor.

Popular Functions

- TOP n : Limit the results to n number of records
- ORDER BY [ASC/DESC] : Used for sorting
- POINT('coordinate system', right ascension, declination)
- CIRCLE('coordinate system', right ascension center, declination center, radius)
- BOX('coordinate system', right ascension center, declination center, width, height)
- POLYGON('coordinate system', coordinate point 1, coordinate point 2, coordinate point 3...)
- DISTANCE(point1, point2)
- CONTAINS(region1, region2)
- INTERSECTS(region1, region2)

Sample Queries

A 1 degree cone search around M101 would be:
SELECT * FROM fp_psc WHERE CONTAINS(POINT('J2000',ra,dec),CIRCLE('J2000',210.80225,54.34894,1.0))=1

A 1 degree by 1 degree box around M101 would be:
SELECT * FROM fp_psc WHERE CONTAINS(POINT('J2000',ra,dec),BOX('J2000',210.80225,54.34894,1.0,1.0))=1

A triangle search around M101 would be:
SELECT * FROM fp_psc WHERE CONTAINS(POINT('J2000',ra,dec),POLYGON('J2000',209.80225,53.34894,209.80225,55.34894,211.80225,54.34894))=1

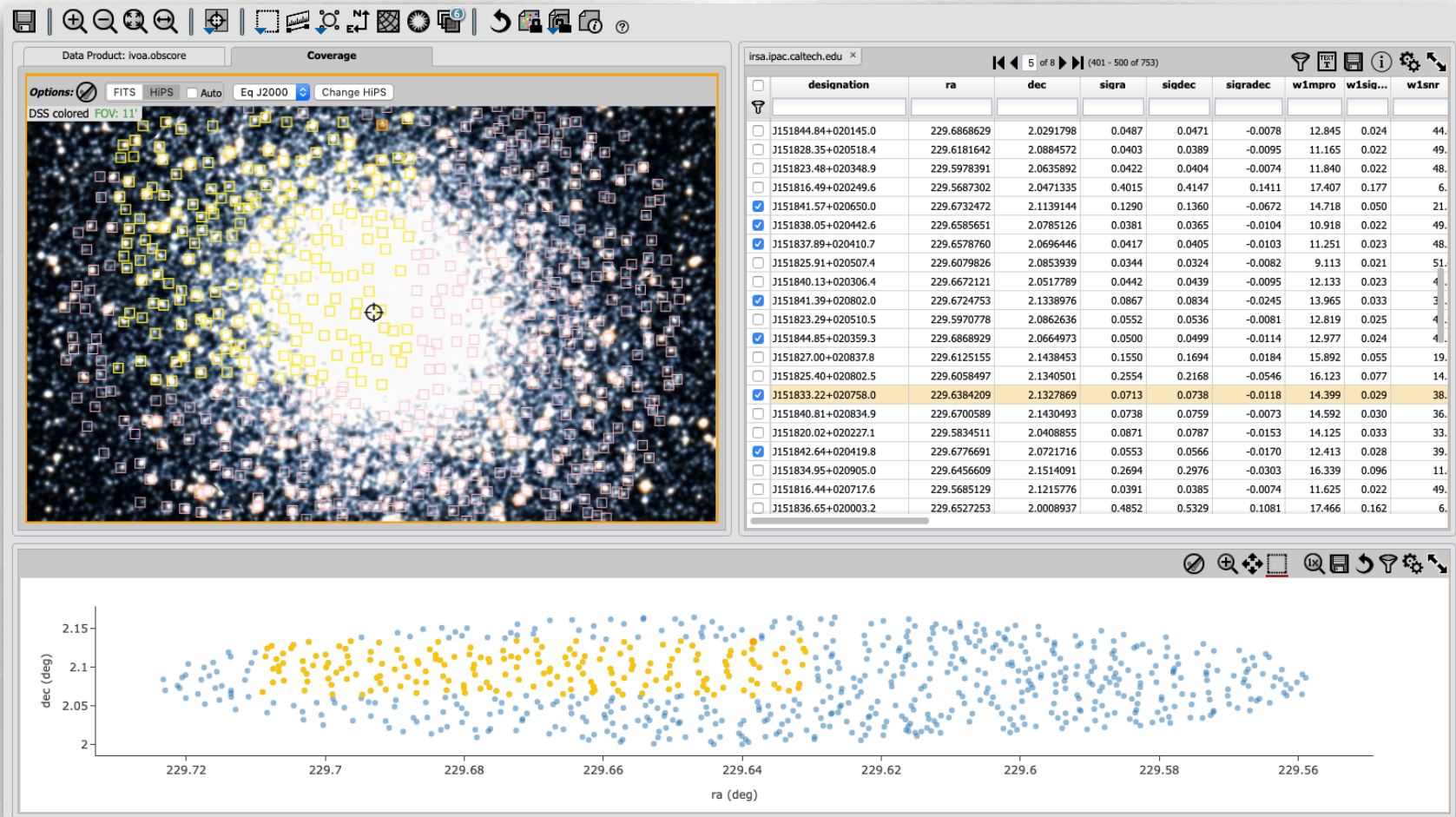
Search Row Limit:

QUERY

TABLE / COLUMN
CHOOSE

HELP

Firefly Tri-View



TAP Next Steps

- Support of Joins In
 - Query Builder
 - Require more UI or lookup
 - TAP_SCHEMA.keys and TAP_SCHEMA.key_columns
- More support helpers
 - Wavelength
 - others?
- Maybe getting TAP services list from registry

TAP Lessons Learned

- Can't go back and forth between Advanced and Helper panels
 - Some sort of JavaScript ADQL parser would be helpful
- UI Hint: We would like to have a `principle` field for table and schemas
- TOP vs MAXREC – was a little hard to understand
 - Feels redundant
 - After time we understood the difference somewhat

TAP: Wishes

UI Hints

- Which tables matter (principle on table and Schema)?
- Common joins, example searches

ObsCore & s_region

TAP Searches

1. TAP Service [?](#) Using <https://ivo.stsci.edu/CAOMTAP/1apService.aspx>. Replace... ▼

2. Select Query Type [?](#) Single Table ADQL

3. Select Table [?](#) Schema: **ivoa**
Definition and support code for the ObsCore data model and table. ▼ Table: **ivoa.obscore**
The IVOA-defined obscore table, version 1.0, containing generic metadata for datasets within this datacenter's CAOM database. ▼

4. Select Constraints [?](#) Reset

Spatial [?](#)

Longitude Column: [?](#)
Latitude Column: [?](#)

Search Method: [?](#)

Name or Position: [?](#) Try NED then Simbad [?](#)
m5 resolved by NED
229.64063, 2.08268 Equ J2000 or 15h18m33.75s, +2d04m57.6s Equ J2000

Radius: [?](#) [?](#)
Valid range between: 1" and 360000"

Temporal [?](#)

<input type="checkbox"/>	column_name	constraints	unit	ucd	description
<input type="checkbox"/>	dataprodukt_type			meta.id	High level scientific classification of the data product, taken from an er...
<input type="checkbox"/>	calib_level			meta.code;obs.calib	Amount of data processing that has been applied to the data
<input type="checkbox"/>	obs_collection			meta.id	Name of a data collection (e.g., project name) this data belongs to
<input type="checkbox"/>	obs_id			meta.id	Unique identifier for an observation
<input type="checkbox"/>	obs_publisher_did			meta.ref.url;meta.curation	Dataset identifier assigned by the publisher.
<input type="checkbox"/>	access_url			meta.ref.url	The URL at which to obtain the data set.
<input type="checkbox"/>	access_format			meta.code.mime	MIME type of the resource at access_url
<input type="checkbox"/>	access_estsize		kbyte	phys.size;meta.file	Estimated size of data product
<input type="checkbox"/>	target_name			meta.id;src	Object a targeted observation targeted
<input type="checkbox"/>	s_ra		deg	pos.eq.ra	RA of (center of) observation, ICRS
<input type="checkbox"/>	s_dec		deg	pos.eq.dec	Dec of (center of) observation, ICRS
<input type="checkbox"/>	s_fov		deg	phys.angSize;inst.fov	Approximate spatial extent for the region covered by the observation
<input type="checkbox"/>	s_region			phys.angArea;obs	Region covered by the observation, as a polygon
<input type="checkbox"/>	s_xel1			meta.number	Number of elements along the first spatial axis
<input type="checkbox"/>	s_xel2			meta.number	Number of elements along the second spatial axis
<input type="checkbox"/>	s_resolution		arcsec	pos.angResolution	Best spatial resolution within the data set
<input type="checkbox"/>	t_min		d	time.start;obs.exposure	Lower bound of times represented in the data set, as MJD
<input type="checkbox"/>	t_max		d	time.end;obs.exposure	Upper bound of times represented in the data set, as MJD
<input type="checkbox"/>	t_exptime		s	time.duration;obs.exposure	Total exposure time
<input type="checkbox"/>	t_resolution		s	time.resolution	Minimal significant time interval along the time axis
<input type="checkbox"/>	t_xel		s	meta.number	dimensions (number of pixels) along the time axis

Search Row Limit: 50000 [?](#)

ObsCore

The screenshot displays the ObsCore software interface. At the top is a toolbar with various icons for file operations, navigation, and data handling. Below the toolbar, the interface is split into two main sections: an image viewer on the left and a data table on the right.

The image viewer, titled "HLA, WFC3/UVIS, hst_13297_79_wfc3_u...", shows a dark field of stars. A red dashed box highlights the toolbar and the image viewer. A red arrow points from the "image" row in the table to the image viewer.

The data table, titled "ivoa.obscore", has columns for "dataproducer type", "calib_level", "obs_collection", and "obs_id". The first few rows are "image" type, and the last few are "spectrum" type. A red arrow points from the "image" row in the table to the image viewer.

A red button labeled "LOAD IMAGE" is located at the bottom center. A red arrow points from the "CHOOSE ROW WITH IMAGE" button to the "image" row in the table.

dataproducer type	calib_level	obs_collection	obs_id
image	2	HLA	hst_13297_79_wfc3_uvis_f336w_02
image	2	HLA	hst_08118_09_wfpc2_f439w_pc_02
image	2	HLA	hst_08118_09_wfpc2_f555w_pc_03
image	2	HLA	hst_06607_03_wfpc2_f336w_pc_03
image	2	HLA	hst_10615_02_acs_wfc_f435w_26
image	2	HLA	hst_10615_02_acs_wfc_f435w_20
image	2	HLA	hst_13297_78_wfc3_uvis_f275w_02
image	2	HLA	hst_06607_03_wfpc2_f336w_pc_02
image	2	HST	j91956boq
image	2	HST	j91956boq
image	2	HST	j91956beq
image	2	HST	j91956beq
image	2	HLA	hst_10120_01_acs_wfc_f625w_03
image	2	HLA	hst_06607_03_wfpc2_f555w_pc_02
image	2	HLA	hst_06607_03_wfpc2_f336w_pc_01
image	2	HLA	hst_13297_78_wfc3_uvis_f336w_01
image	3	HLA	hst_08118_09_wfpc2_f439w_pc
image			
spectrum			
spectrum			
spectrum			
spectrum			
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000
spectrum	2	FUSE	a1080303000

LOAD IMAGE

CHOOSE ROW WITH IMAGE

CHANGE TO COVERAGE
TAB

s_region

CONTROL DRAWING
LAYERS

The screenshot displays a software interface for astronomical data visualization. The main window is titled 'Data Product: ivoa.observ' and shows a 'Coverage' tab. The central area is a dark star field with several overlapping circular and rectangular regions in green and orange. A toolbar at the top contains various icons, with a red box highlighting a subset of them. On the right, a 'Layers- DSS colored' panel is open, listing several layers with checkboxes and color/symbol options. A red box highlights the 'Coverage: ivoa.observ positions' and 'Coverage: ivoa.observ regions' entries in this panel. A 'Layers' table is visible in the background, listing multiple 'image' entries. Three callout boxes with arrows provide instructions: 'CHANGE TO COVERAGE TAB' points to the toolbar, 'CONTROL DRAWING LAYERS' points to the 'Layers' panel, and 'SHOW: ALL, SELECTED, OR HIGHLIGHTED' points to the 'All Selected Highlighted' buttons in the 'Layers' panel.

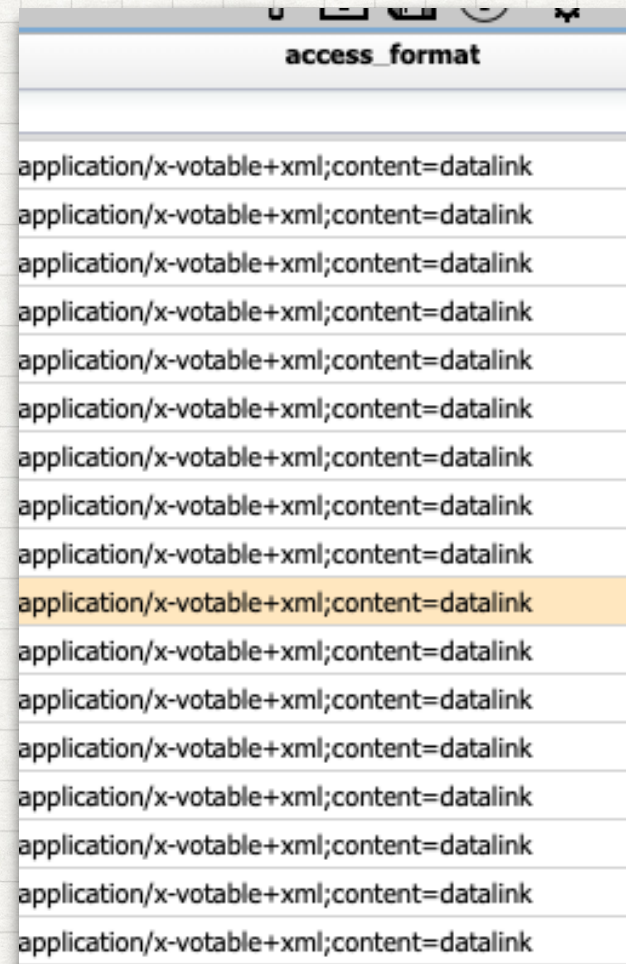
CLICK ON ROW OR OBJECT
AND SEE IT UPDATE

SHOW: ALL, SELECTED,
OR HIGHLIGHTED

ObsCore & s_region Comments

- ObsCore
 - ObsCore is very straight forward, easy to implement
 - How to recognize and ObsCore table?
 - Became more "ObsCore like"
 - `access_url`, `access_format`, `dataprodect_type`
- `s_region` easy to implement because we already support DS9 regions
 - We had all the foundation work
 - Similar parsing but still requires a separate parser
 - only support: `polygon`, `circle`, `box`, `position`
 - not supported: `union`, `intersection`, `not`
 - We draw two layers: the center and the `s_region`

DataLink



DATALINK DATA MIGHT
HAVE MORE OPTIONS

DataLink

The screenshot displays the DataLink interface for the 'ivoa.ObsCore' data product. On the left, a FITS image of a star field is shown, with a red dashed box highlighting a 'More' menu. The menu options are:

- Show FITS Image: Primary product (#this)
- Show PNG image: preview
- Show PNG image: <http://www.opencadc.org/caom2#thumbnail>

On the right, a table lists various data products with their corresponding access formats. The table has two columns: the first column contains the product name (e.g., [mos_4266_acs_wfc_f606w_drz](#)), and the second column contains the access format (e.g., `application/x-votable+xml;content=datalink`).

	access_format
<input type="checkbox"/> mos_4266_acs_wfc_f606w_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> mos_4266_acs_wfc_f814w_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> ?hst_06607_03_wfpc2_f336w_wf_01_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> 06607_03_wfpc2_f336w_wf_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> ?hst_06607_03_wfpc2_f336w_wf_03_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> 06607_03_wfpc2_f255w_wf_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> ?hst_06607_03_wfpc2_f255w_wf_02_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> ?hst_06607_03_wfpc2_f555w_pc_03_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> ?hst_06607_03_wfpc2_f255w_wf_01_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> ?hst_06607_03_wfpc2_f555w_pc_05_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> ?hst_06607_03_wfpc2_f255w_wf_03_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> 06607_03_wfpc2_f255w_pc_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> ?hst_06607_03_wfpc2_f555w_pc_02_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> ?hst_06607_03_wfpc2_f255w_pc_01_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> ?hst_06607_03_wfpc2_f336w_pc_03_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> ?hst_06607_03_wfpc2_f255w_pc_02_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> 06607_03_wfpc2_f555w_pc_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> 06607_03_wfpc2_f336w_pc_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> ?hst_06607_03_wfpc2_f555w_pc_01_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> ?hst_06607_03_wfpc2_f336w_pc_01_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> ?hst_06607_03_wfpc2_f255w_pc_02_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> 06607_03_wfpc2_f555w_pc_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> 06607_03_wfpc2_f336w_pc_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> ?hst_06607_03_wfpc2_f555w_pc_01_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> ?hst_06607_03_wfpc2_f336w_pc_01_drz	application/x-votable+xml;content=datalink
<input type="checkbox"/> ?hst_06607_03_wfpc2_f555w_pc_04_drz	application/x-votable+xml;content=datalink

DataLink Comments

- DataLink is Powerful
 - UI perspective: It might be one of the most useful VO protocols
- Poor Server Performance - makes it hard on a UI tool
- So far, only partial implementation
 - No parameter substitution
 - Firefly does not yet load other resource/meta section of VO table
 - No UI for user input

DataLink Next Steps

- Hope to have full support
- Support percent: 33%
- Still to do:
 - Implement `service_def`
 - Firefly load separate resource section
 - Parameter substitution
 - Generate a UI to do user inputted parameter substitution
- Recognize Data Link beyond obscure
 - beyond `access_format` and `access_url`
 - how
 - e.g. GAIA uses `datalink_url`

Datalink: Wishes

UI Hints

- FITS
 - How many HDUs?
 - break down of HDUs (table, images, cubes)?
- Tables
 - What are the best charting columns?
 - Should this be primarily shown as chart or table?
- Related Images
 - i.e. 4 WISE band images or 3 2MASS band images
- What data really matters?
- Unique name for each piece of data? Just have semantics

Final thoughts

- Services are well defined.
 - Thanks to document writers!
- Would like more UI hints
- Firefly is project driven
 - We don't have a time table next steps
- Seems to be a huge upside to implementing these standards