

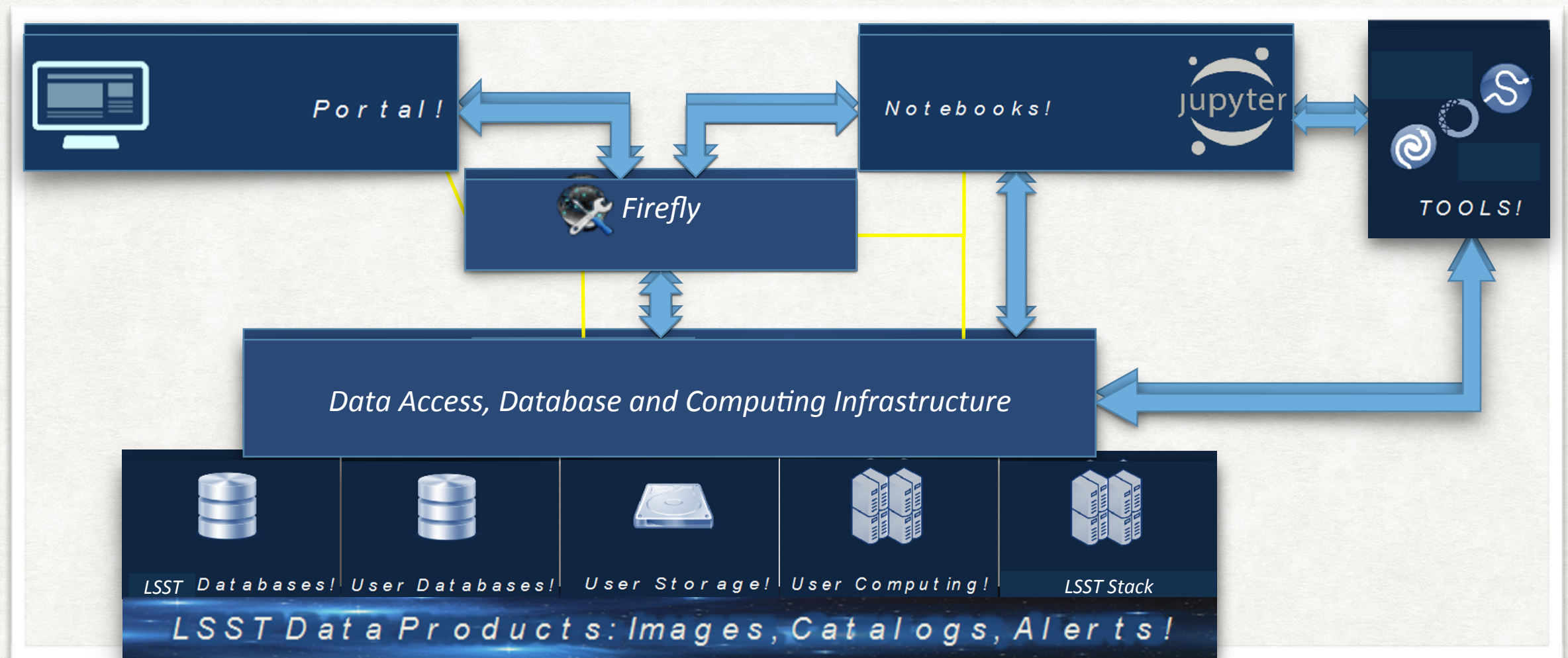
Python widgets for astronomy visualization

David Shupe (Caltech/IPAC)
Scientist, LSST and ZTF

on behalf of the LSST
Science User Interface & Tools team

Context: LSST Science Platform

Astronomy in 2020+



Too much data to download -> Remote workspaces

Python widgets for astronomy visualization

Python widgets for astronomy visualization

- The Firefly visualization system

Python widgets for astronomy visualization

- The Firefly visualization system
- APIs to Firefly

Python widgets for astronomy visualization

- The Firefly visualization system
- APIs to Firefly
- Example: Python interaction with the Firefly app

Python widgets for astronomy visualization

- The Firefly visualization system
- APIs to Firefly
- Example: Python interaction with the Firefly app
- Firefly widgets for Jupyter

Python widgets for astronomy visualization

- The Firefly visualization system
- APIs to Firefly
- Example: Python interaction with the Firefly app
- Firefly widgets for Jupyter
- Resources

Python widgets for astronomy visualization

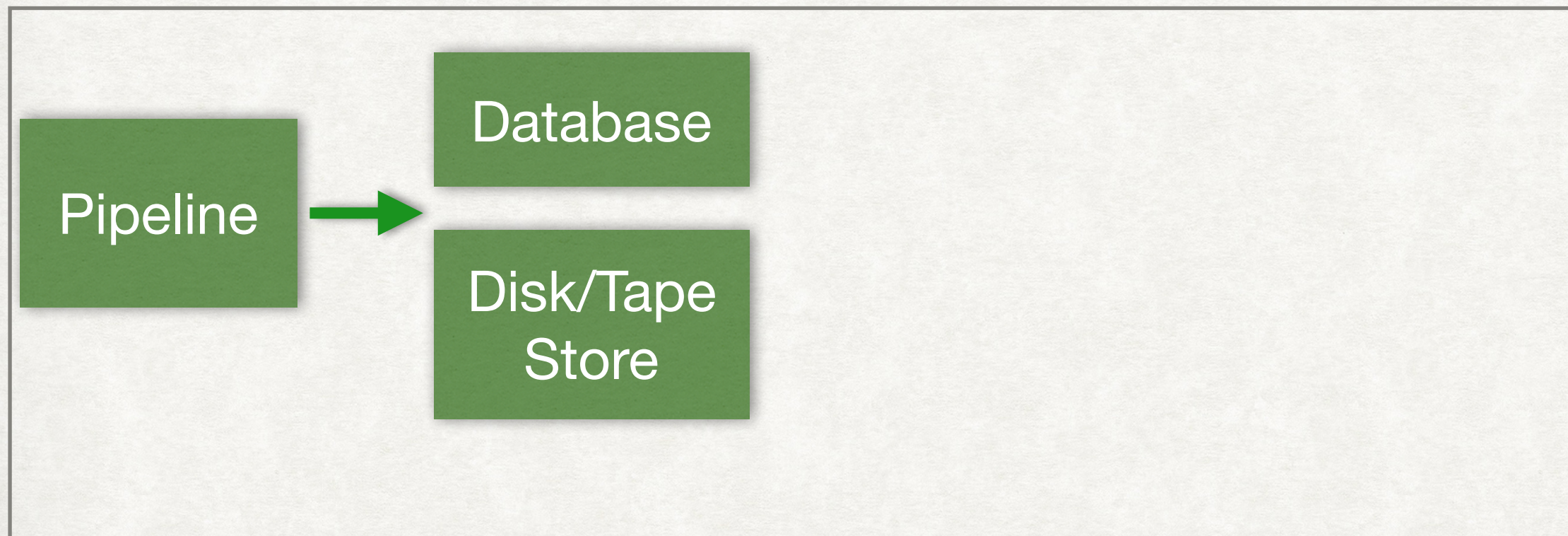
- The Firefly visualization system
- APIs to Firefly
- Example: Python interaction with the Firefly app
- Firefly widgets for Jupyter
- Resources

Firefly

Archive Access System

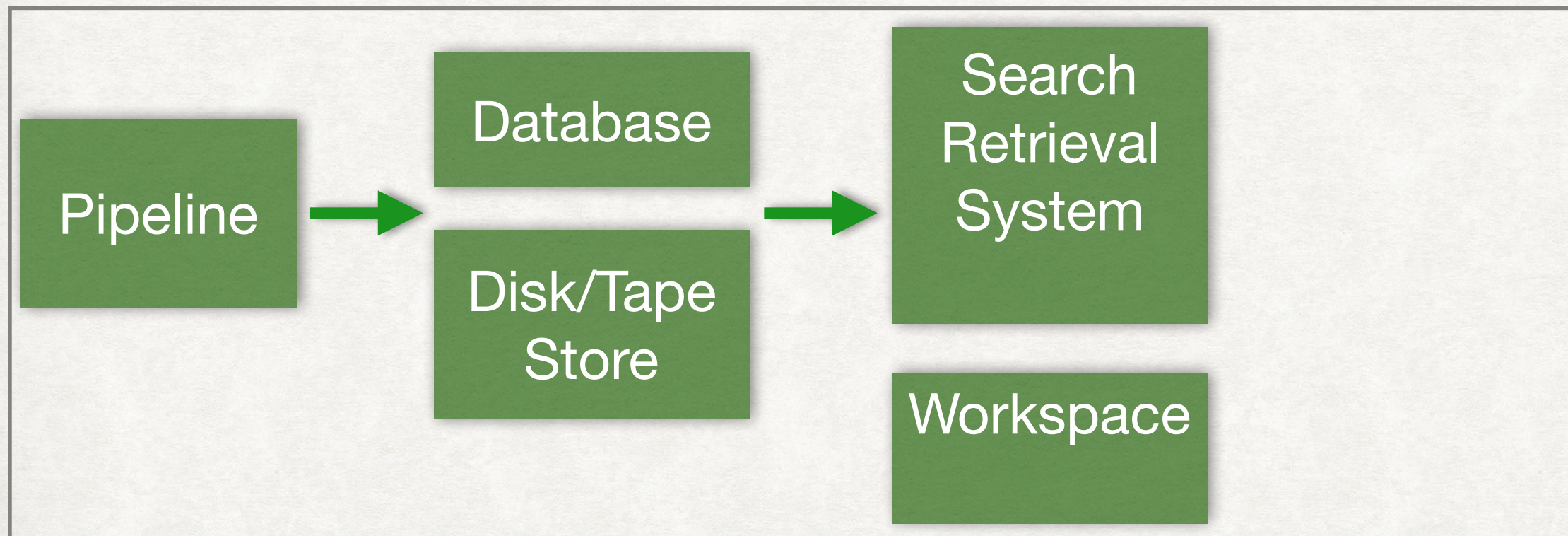
Firefly

Archive Access System



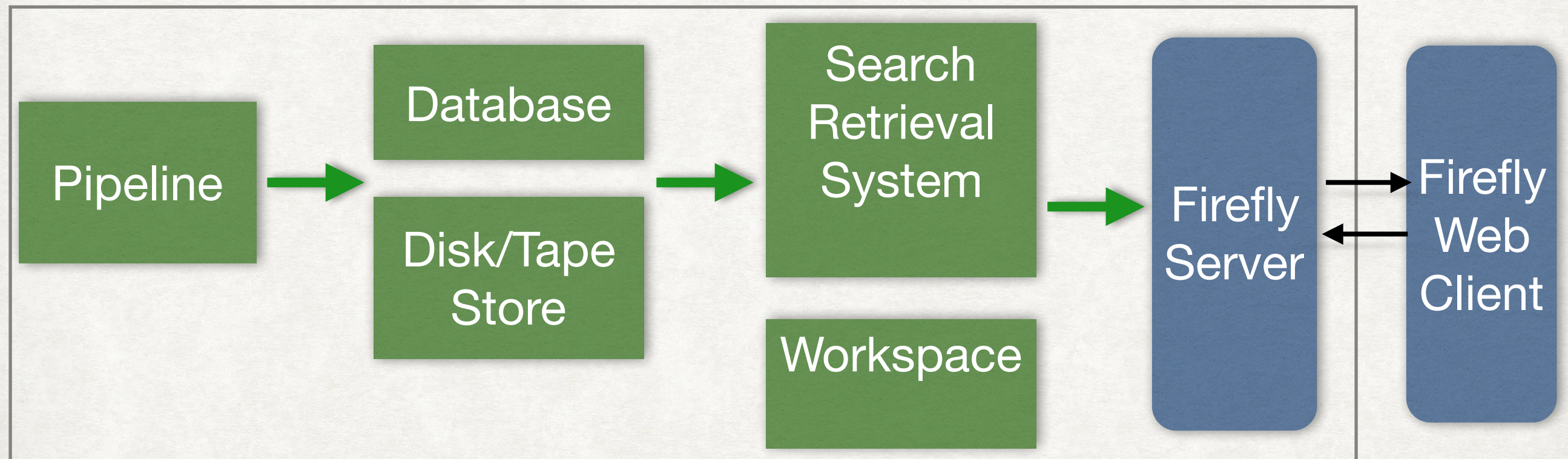
Firefly

Archive Access System



Firefly

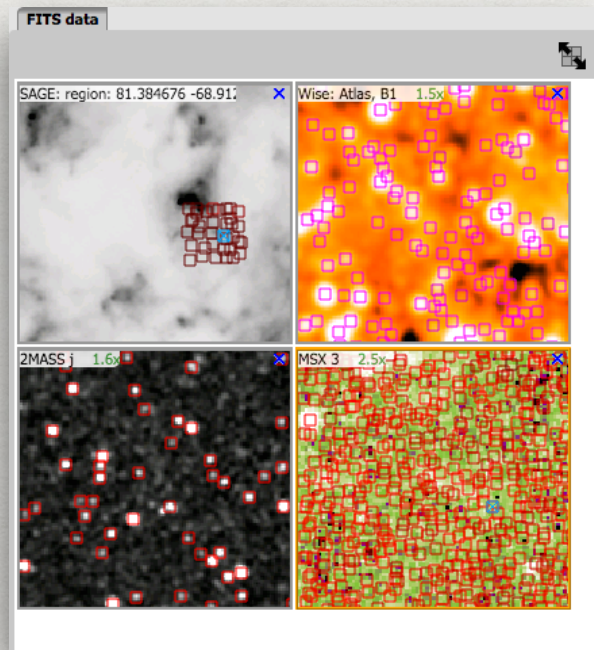
Archive Access System



**FITS Image
Viewer**

Table Display

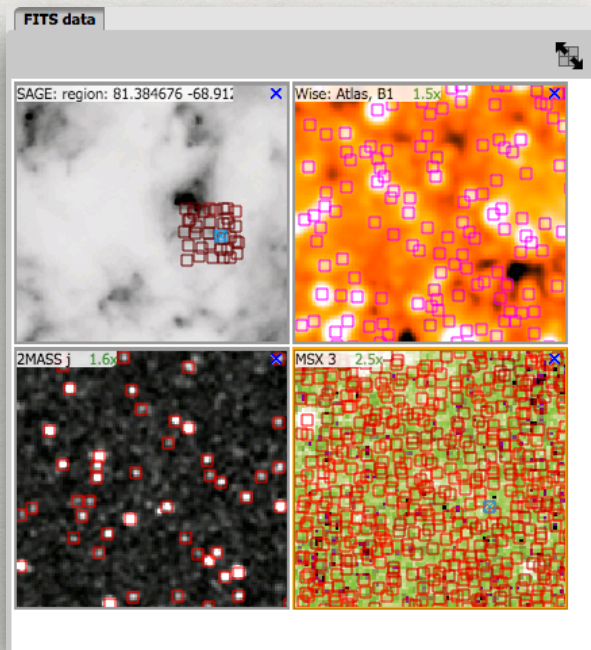
XY Plot / Histogram



- **WCS Readout**
- **Zoom**
- **Flip/ Rotate/ Crop**
- **Color / Stretch**
- **Grid**
- **Region**
- **Magnifier**
- **Distance tools**
- **Markers**
- **Fits Headers**
- **Crop**

Table Display

XY Plot / Histogram



- WCS Readout
- Zoom
- Flip/ Rotate/ Crop
- Color / Stretch
- Grid
- Region
- Magnifier
- Distance tools
- Markers
- Fits Headers
- Crop

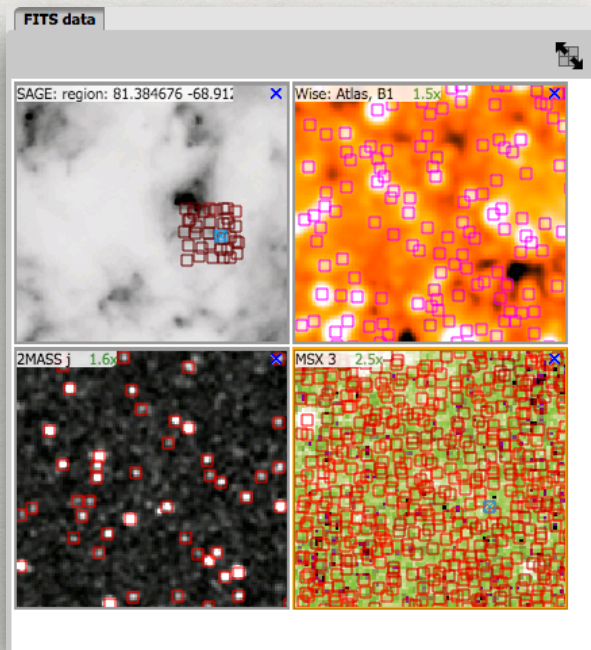
fp_pscBox, X

9 of 21 (401 - 450 of 1015)

	ra (deg)	dec (deg)	clon	clat	err_maj (arcsec)	err_min (arcsec)	err_ang (deg)	designation	i_m (mag)	i_cmsig (mag)	i_msigcon (mag)
<input type="checkbox"/>											
<input type="checkbox"/>	81.279334	-68.919395	05h25m07.04s	-68d55m09.82s	0.14	0.13	2	05250704-6855098	16.027	0.089	0.089
<input type="checkbox"/>	81.370766	-68.836624	05h25m28.98s	-68d50m11.85s	0.18	0.16	13	05252898-6850118	13.504		
<input type="checkbox"/>	81.412929	-68.839584	05h25m39.10s	-68d50m22.50s	0.20	0.16	84	05253910-6850225	16.360	0.137	0.137
<input type="checkbox"/>	81.437965	-68.844025	05h25m45.11s	-68d50m38.49s	0.17	0.17	135	05254511-6850384	16.324	0.133	0.133
<input type="checkbox"/>	81.314225	-68.904945	05h25m15.41s	-68d54m17.80s	0.07	0.07	45	05251541-6854178	15.196	0.072	0.073
<input type="checkbox"/>	81.368899	-68.837242	05h25m28.54s	-68d50m14.07s	0.09	0.08	3	05252853-6850140	14.330	0.053	0.054
<input type="checkbox"/>	81.500049	-68.893616	05h26m00.01s	-68d53m37.02s	0.32	0.27	83	05260001-6853370	16.490	0.140	0.140
<input type="checkbox"/>	81.419247	-68.914131	05h25m40.62s	-68d54m50.87s	0.06	0.06	45	05254061-6854508	15.304	0.062	0.064
<input type="checkbox"/>	81.591179	-68.839294	05h26m21.88s	-68d50m21.46s	0.16	0.14	45	05262188-6850214	16.409	0.126	0.126
<input type="checkbox"/>	81.586821	-68.896202	05h26m20.84s	-68d53m46.33s	0.06	0.06	45	05262083-6853463	14.670	0.039	0.041
<input type="checkbox"/>	81.337872	-68.843903	05h25m21.09s	-68d50m38.05s	0.07	0.07	45	05252108-6850380	15.509	0.052	0.054
<input type="checkbox"/>	81.394806	-68.906075	05h25m34.75s	-68d54m21.87s	0.06	0.06	90	05253475-6854218	14.142	0.034	0.036
<input type="checkbox"/>	81.409027	-68.876686	05h25m38.17s	-68d52m36.07s	0.20	0.18	177	05253816-6852360	16.475	0.163	0.163
<input type="checkbox"/>	81.600449	-68.830826	05h26m24.11s	-68d49m50.97s	0.15	0.14	106	05262410-6849509	16.200	0.124	0.125
<input type="checkbox"/>	81.330078	-68.829193	05h25m19.22s	-68d49m45.09s	0.19	0.17	83	05251921-6849450	16.433	0.129	0.130
<input type="checkbox"/>	81.657667	-68.909805	05h26m37.84s	-68d54m35.30s	0.07	0.07	17	05263784-6854352	15.465	0.084	0.085
<input type="checkbox"/>	81.471096	-68.948822	05h25m53.06s	-68d56m55.76s	0.07	0.07	45	05255306-6856557	15.425	0.055	0.057
<input type="checkbox"/>	81.317534	-68.908012	05h25m16.21s	-68d54m28.84s	0.17	0.15	8	05251620-6854288	16.384	0.147	0.148
<input type="checkbox"/>	81.562626	-68.883179	05h26m15.03s	-68d52m59.44s	0.22	0.20	84	05261503-6852594	16.040	0.104	0.104
<input type="checkbox"/>	81.585997	-68.867485	05h26m20.64s	-68d52m02.95s	0.07	0.06	86	05262063-6852029	15.393	0.057	0.058

- Sort / Filter
- Column Controls
- Supports large tables, 10 Million+ rows
- Very fast response time

XY Plot / Histogram



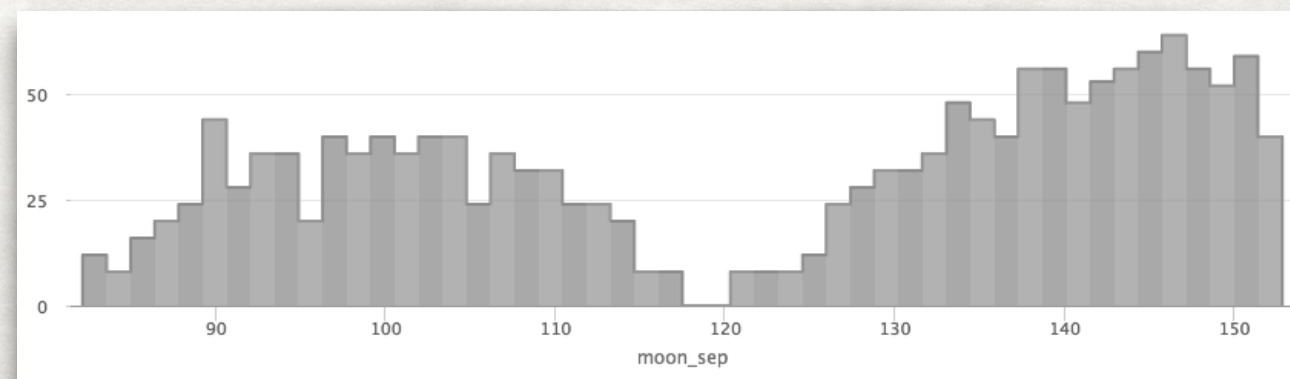
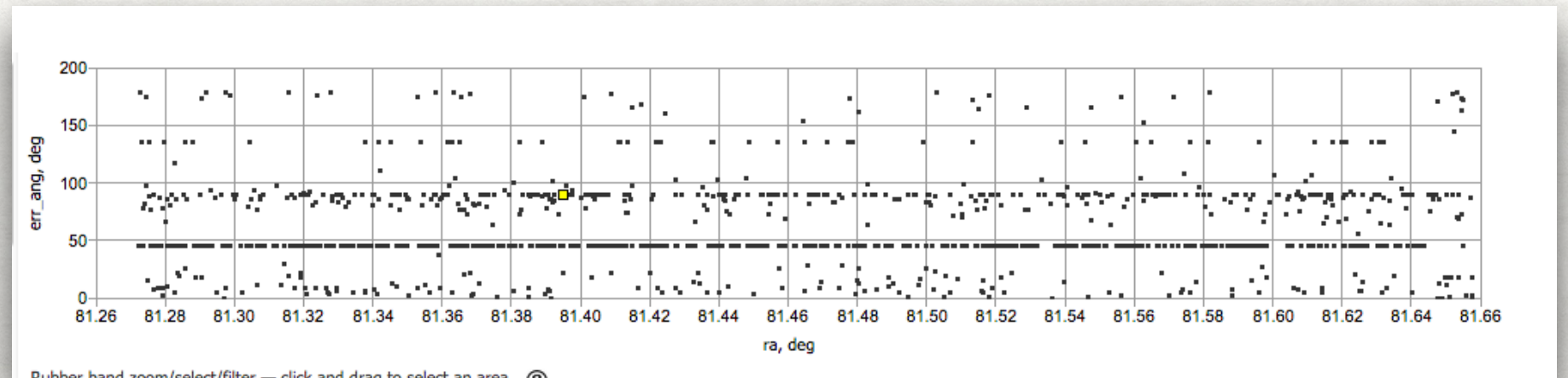
- WCS Readout
- Zoom
- Flip/ Rotate/ Crop
- Color / Stretch
- Grid
- Region
- Magnifier
- Distance tools
- Markers
- Fits Headers
- Crop

fp_pscBox, X

9 of 21 (401 - 450 of 1015)

	ra (deg)	dec (deg)	clon	clat	err_maj (arcsec)	err_min (arcsec)	err_ang (deg)	designation	i_m (mag)	i_cmsig (mag)	i_msigcon (mag)
<input type="checkbox"/>	81.279334	-68.919395	05h25m07.04s	-68d55m09.82s	0.14	0.13	2	05250704-6855098	16.027	0.089	0.089
<input type="checkbox"/>	81.370766	-68.836624	05h25m28.98s	-68d50m11.85s	0.18	0.16	13	05252898-6850118	13.504		
<input type="checkbox"/>	81.412929	-68.839584	05h25m39.10s	-68d50m22.50s	0.20	0.16	84	05253910-6850225	16.360	0.137	0.137
<input type="checkbox"/>	81.437965	-68.844025	05h25m45.11s	-68d50m38.49s	0.17	0.17	135	05254511-6850384	16.324	0.133	0.133
<input type="checkbox"/>	81.314225	-68.904945	05h25m15.41s	-68d54m17.80s	0.07	0.07	45	05251541-6854178	15.196	0.072	0.073
<input type="checkbox"/>	81.368899	-68.837242	05h25m28.54s	-68d50m14.07s	0.09	0.08	3	05252853-6850140	14.330	0.053	0.054
<input type="checkbox"/>	81.500049	-68.893616	05h26m00.01s	-68d53m37.02s	0.32	0.27	83	05260001-6853370	16.490	0.140	0.140
<input type="checkbox"/>	81.419247	-68.914131	05h25m40.62s	-68d54m50.87s	0.06	0.06	45	05254061-6854508	15.304	0.062	0.064
<input type="checkbox"/>	81.591179	-68.839294	05h26m21.88s	-68d50m21.46s	0.16	0.14	45	05262188-6850214	16.409	0.126	0.126
<input type="checkbox"/>	81.586821	-68.896202	05h26m20.84s	-68d53m46.33s	0.06	0.06	45	05262083-6853463	14.670	0.039	0.041
<input type="checkbox"/>	81.337872	-68.843903	05h25m21.09s	-68d50m38.05s	0.07	0.07	45	05252108-6850380	15.509	0.052	0.054
<input type="checkbox"/>	81.394806	-68.906075	05h25m34.75s	-68d54m21.87s	0.06	0.06	90	05253475-6854218	14.142	0.034	0.036
<input type="checkbox"/>	81.409027	-68.876686	05h25m38.17s	-68d52m36.07s	0.20	0.18	177	05253816-6852360	16.475	0.163	0.163
<input type="checkbox"/>	81.600449	-68.830826	05h26m24.11s	-68d49m50.97s	0.15	0.14	106	05262410-6849509	16.200	0.124	0.125
<input type="checkbox"/>	81.330078	-68.829193	05h25m19.22s	-68d49m45.09s	0.19	0.17	83	05251921-6849450	16.433	0.129	0.130
<input type="checkbox"/>	81.657667	-68.909805	05h26m37.84s	-68d54m35.30s	0.07	0.07	17	05263784-6854352	15.465	0.084	0.085
<input type="checkbox"/>	81.471096	-68.948822	05h25m53.06s	-68d56m55.76s	0.07	0.07	45	05255306-6856557	15.425	0.055	0.057
<input type="checkbox"/>	81.317534	-68.908012	05h25m16.21s	-68d54m28.84s	0.17	0.15	8	05251620-6854288	16.384	0.147	0.148
<input type="checkbox"/>	81.562626	-68.883179	05h26m15.03s	-68d52m59.44s	0.22	0.20	84	05261503-6852594	16.040	0.104	0.104
<input type="checkbox"/>	81.585997	-68.867485	05h26m20.64s	-68d52m02.95s	0.07	0.06	86	05262063-6852029	15.393	0.057	0.058

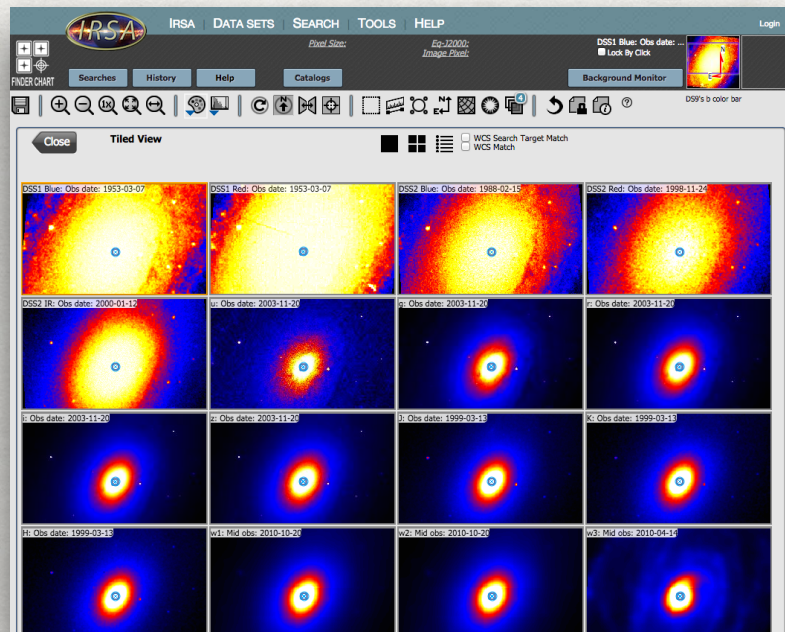
- Sort / Filter
- Column Controls
- Supports large tables, 10 Million+ rows
- Very fast response time



- Interactive
- Column math
- Zoom
- Filter

Many Faces of Firefly

Many Faces of Firefly



DSS .2x

o -- search center x -- retrieved observations

ation ID	Thumbprint	Target	RA (degree)	DEC (degree)	Instrument	Observation Mode
		m81	148.905101	69.079243	PACS	PacsPhoto
		m81	148.888208	69.065278	PACS	PacsPhoto
		m81	148.866706	69.073330	SpIRE	SpirePhotoLargeScan
		m81	148.854483	69.082676	PACS	PacsPhoto
		m81	148.884044	69.083377	PACS	PacsPhoto
		m81	148.885753	69.065112	SpIRE	SpireSpectroPoint
		m81	148.879747	69.032080	PACS	PacsLineSpec



Object/Coordinate

Source

Type

Glra

Glra

Equatorial J2000

MESSIER 81

142.09185

+40.90014

09h 55m 33.17s +40d 03m 55.0s

Cone Search

Constraints: No

25 sources found.

2MASS k 1.6x

ra, deg

dec, deg

ra, deg

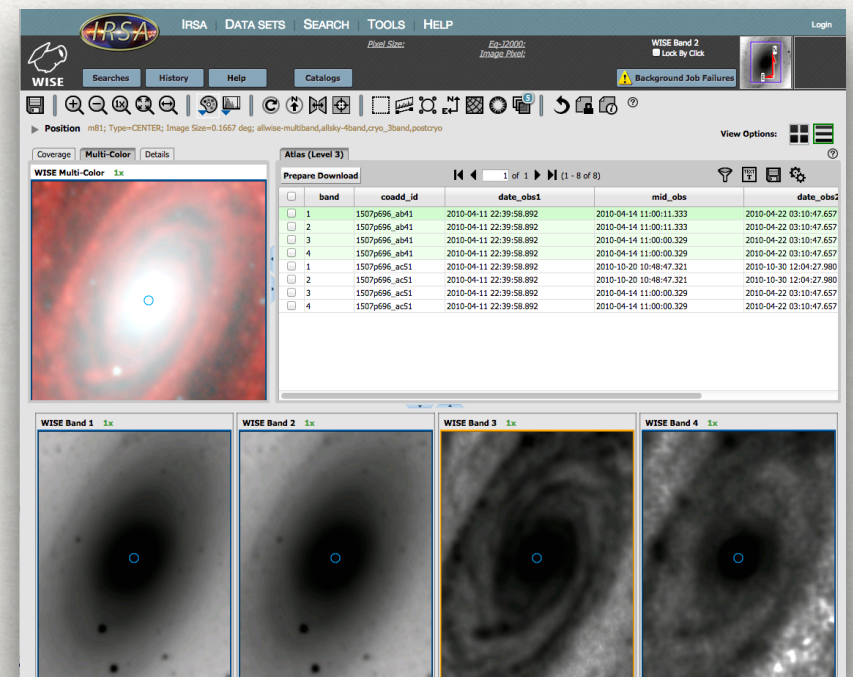
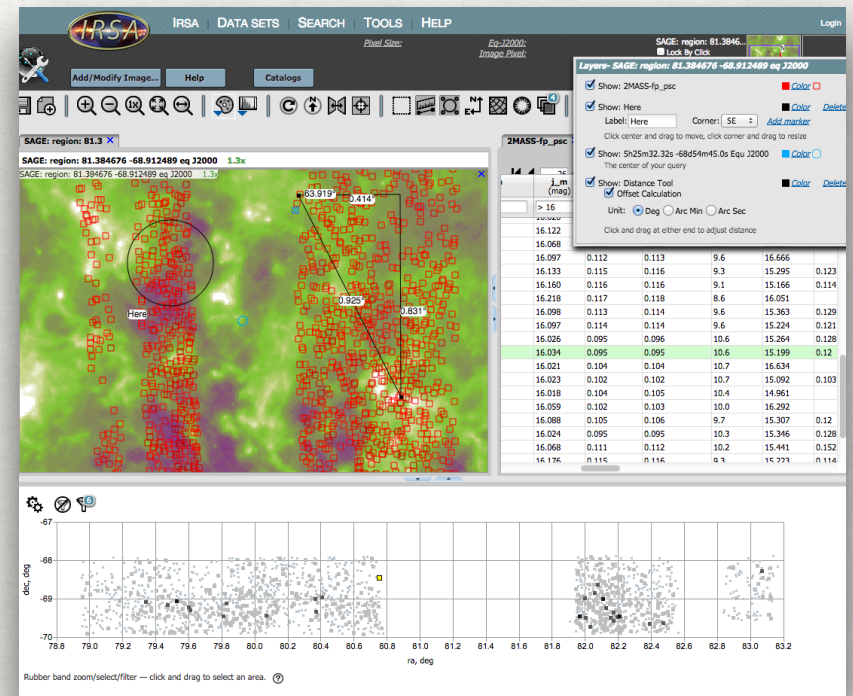
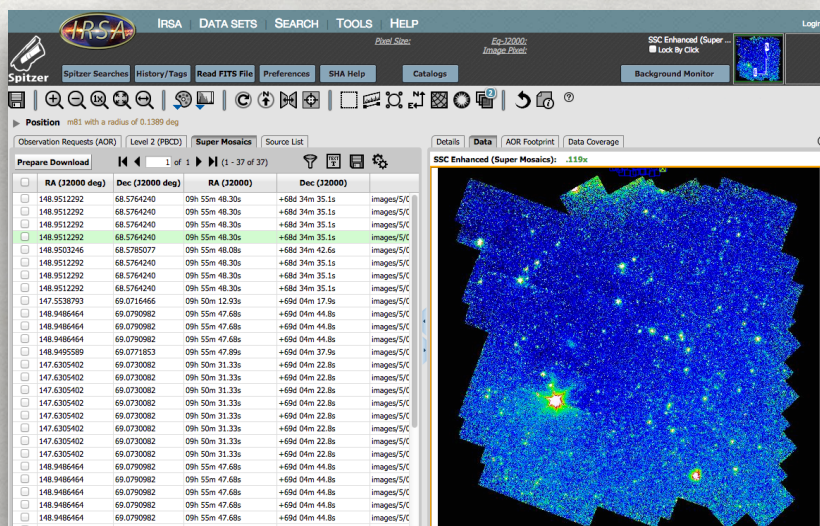
dec, deg

Rubber band zoom/select/filter — click and drag to select an area.

Result IPAC Table

Column Key

ra (deg)	dec (deg)	err_maj (arcsec)	err_min (arcsec)	err_ang (deg)	designation	j_m (mag)	j_cmag (mag)	j_msigcom (mag)	j_snr	j_m (mag)	j_c (mag)
148.857851	69.064140	0.08	0.07	176	09552588+69033509	13.706	0.017	0.021	110.9	13.488	0.015
148.941103	69.050117	0.08	0.07	176	09554386+69030044	15.266	0.055	0.056	26.4	14.481	0.064
148.896449	69.061554	0.14	0.14	108	09553514+69034115	11.271			10.338		
148.895275	69.062691	0.18	0.16	17	09553486+69034556	10.885			11.470	0.198	
148.899725	69.054871	0.15	0.13	5	09553593+69033175	13.208			14.909	0.111	
148.882582	69.068092	0.14	0.13	176	09553181+69040511	12.217	0.201	0.201	437.1	10.112	
148.822464	69.058517	0.08	0.07	0	09551739+69033066	14.624	0.030	0.032	47.6	14.086	0.031
148.893402	69.062233	0.12	0.11	156	09553444+69034040	10.881			10.136		
148.905288	69.057541	0.15	0.15	74	09553726+69032711	12.955			12.216		
148.888387	69.070961	0.12	0.11	24	09553321+69041544	14.539	0.066	0.067	51.5	11.402	
148.889210	69.071922	0.23	0.22	64	09553341+69041899	12.503			13.687	0.266	
148.916544	69.069229	0.50	0.50	0	09553997+69040922	16.773	0.249	0.249	6.6	13.863	0.521
148.879981	69.072540	0.15	0.13	83	09553023+69042111	14.428	0.147	0.148	57.0	11.908	
148.936104	69.059981	0.29	0.26	128	09554466+69033356	16.624	0.184	0.185	7.5	14.085	
148.892998	69.071259	0.32	0.31	42	09553431+69041655	12.468			13.607	0.339	
148.892444	69.063744	0.12	0.11	0	09553418+69034944	12.076	0.017	0.021	497.8	9.452	
148.903639	69.060951	0.12	0.11	13	09553687+69033394	14.509	0.055	0.056	53.0	11.640	
148.880872	69.067589	0.15	0.13	11	09553140+69040333	12.549	0.182	0.183	222.0	11.513	0.333



Using Firefly

Using Firefly

- Servers
 - Application at Infrared Science Archive
<http://irsa.ipac.caltech.edu/irsaviewer>
 - Standalone Firefly
Download a single file, deploy using only Java 8

Using Firefly

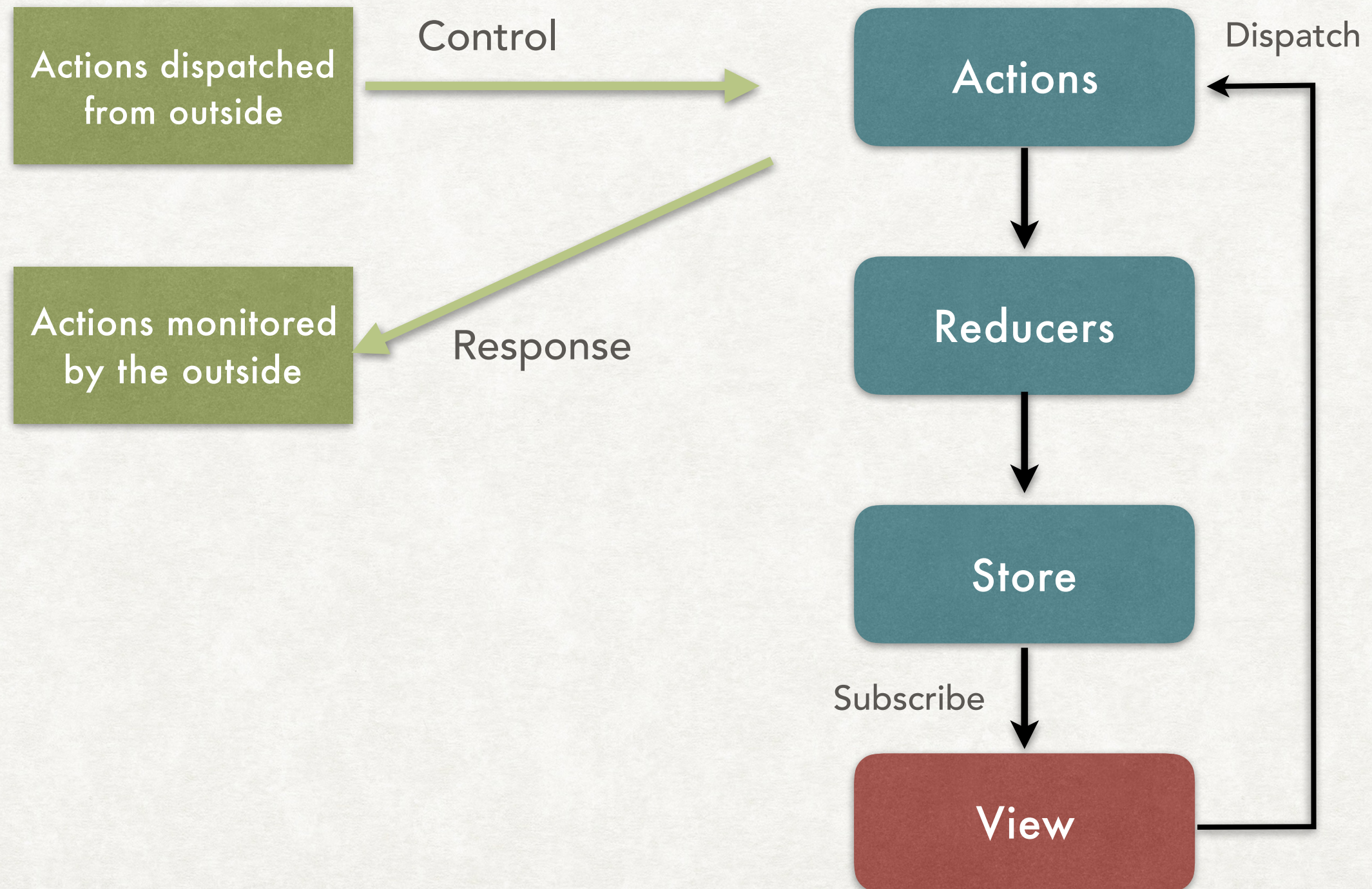
- Servers
 - Application at Infrared Science Archive
<http://irsa.ipac.caltech.edu/irsaviewer>
 - Standalone Firefly
Download a single file, deploy using only Java 8
- Programmatic
 - Javascript
 - Python with Firefly the single-page application
 - Widgets for Jupyter notebook / JupyterLab

Python widgets for astronomy visualization

- The Firefly visualization system
- APIs to Firefly
 - Javascript
 - Python: `firefly_client`
- Example: Python interaction with the Firefly app
- Firefly widgets for Jupyter
- Resources

External and internal actions are enabled

Contemporary JavaScript framework



Javascript API falls out naturally

- Dispatch actions to make a change

```
var action= firefly.action;
```

- To change color:

```
action.dispatchColorChange({plotId: 'myplot',  
    cbarId: 5});
```

- Zooming an image:

```
action.dispatchZoom({plotId: 'myplot',  
    userZoomType: 'UP' });
```

```
action.dispatchZoom({plotId: 'myplot',  
    userZoomType: 'LEVEL', level: zlevel,  
    forceDelay: true });
```


Documentation for Javascript API

JavaScript Firefly Tools API

Firefly tools is an API that can be used from JavaScript. High-level API allows to render the main components of Firefly and make them share the same data model. Low-level API gives direct access to Firefly React components and application state. Firefly tools API also allows to dispatch or add a listener to virtually any action, available in Firefly internally. This makes it possible to extend Firefly by executing custom code on various UI events, like move or region selection.

The main Firefly components are:

- [FITS Visualizer](#)
- [Table](#)
- [Charts](#)

These components can be setup to share the same data model. Therefore you can do the following combinations:

- [Connect FITS viewer coverage image to a table](#)
- [Connect XY plot to a table](#). A Table with any data and a XY Plot showing plots from any two columns of the table.
- Tri-view: Table, FITS coverage, and XY Plot together showing the same data.

Lower level API is built around the following modules:

- `firefly.ui` - UI components
- `firefly.util` - utilities
- `firefly.action` - actions, changing application state

More information about lower level API can be found here:

- [Rendering UI Components](#)
- [Dispatching and Watching Actions](#)
- [Other utilities](#)
- [Utility methods for FITS visualization](#)
- [Adding Context Extensions to FITS viewer](#)
- [Region Support](#)

Starting Firefly Tools in JavaScript

Python API tracks Javascript closely

- `firefly_client` is a thin layer over Javascript API
- Pip-installable
- Dispatch actions, add listeners (callbacks)

- Image display for a local Firefly server:

```
import firefly_client
```

```
fc = firefly_client.FireflyClient('localhost:8080')
```

```
fval = fc.upload_file('myfile.fits')
```

```
fc.show_fits(fval)
```


Python widgets for astronomy visualization

- The Firefly visualization system
- APIs to Firefly
- Example: Python interaction with the Firefly app
- Firefly widgets for Jupyter
- Resources

Forced photometry with LSST stack

Setup with Jupyter notebook

- Define function for forced photometry for (ra, dec, filter)
- Callback listens for row selection and point events
- Extension added to Firefly to enable this button in point selection mode

Forced Photometry

The callback function applies `fetch_forcedphot` when activated in Firefly.

```
def callback_forcedphot(event, verbose=False):
    global src_cat
    selected_filter = None
    if verbose:
        for key in event:
            lsst.log.info(str(key) + " : " + str(event[key]))
        if 'data' in event:
            lsst.log.info('Dumping event data')
            for key in event['data']:
                lsst.log.info(str(key) + " : " + str(event['data'][key]))
    if 'table' in event['data']:
        if 'row' in event['data']['table']:
            if 'filterName' in event['data']['table']['row']:
                selected_filter = event['data']['table']['row']['filterName']
    if 'wpt' in event['data']:
        wpt = event['data']['wpt']
        wdata = wpt.split(';')
        ra = float(wdata[0])
        dec = float(wdata[1])
        if selected_filter is not None:
            outtab = fetch_forcedphot(ra, dec, selected_filter)
        else:
            outtab = fetch_forcedphot(ra, dec, myfilter)
        if outtab is None:
            lsst.log.error('No photometry returned')
            return
        outtab.write('fout.tbl', format='ipac')
        with open('fout.tbl', 'r') as original:
            data = original.read()
        with open('fout.tbl', 'w') as modified:
            modified.write(r"\datasource = img_url" + "\n" +
                           r"\ts_time = mjd" + "\n" +
                           r"\ts_value = base_PsfFlux_flux" + "\n" +
                           r"\positionCoordColumns = ra;dec;EQ_J2000" + "\n" +
                           data)
        tval = fc.upload_file('fout.tbl')
        fc.show_table(tval, tbl_id='Forced Phot')
        fc.show_xyplot(tbl_id='Forced Phot', xCol='mjd', yCol='base_PsfFlux_flux',
                       yError='base_PsfFlux_fluxSigma', yOptions='grid')
    if tv is not None:
        tv.url_or_path = tval
```


Add-on from Python: photometry on click

deepCoadds Coverage

deepCoadds

Image: < > IMAGE 1/3 Options: **Forced Photometry**

id=19202106 0.2x

WCS Match
Target Match

Zoom the

<input type="checkbox"/>	deepCoaddId	tract	patch
<input type="checkbox"/>	19202104	0	293,7
<input type="checkbox"/>	19202105	0	293,7
<input type="checkbox"/>	19202106	0	293,7
<input type="checkbox"/>	19202107	0	293,7
<input type="checkbox"/>	19202108	0	293,7


Add-on from Python: photometry on click

deepCoadds Coverage

deepCoadds

Image: < > IMAGE 1/3 Options: **Forced Photometry**

id=19202106 0.2x



WCS Match
Target Match

deepCoadds x

	deepCoaddId	tract	patch
<input type="checkbox"/>	19202104	0	293,7
<input type="checkbox"/>	19202105	0	293,7
<input type="checkbox"/>	19202106	0	293,7
<input type="checkbox"/>	19202107	0	293,7
<input type="checkbox"/>	19202108	0	293,7

Table of results is returned to the browser

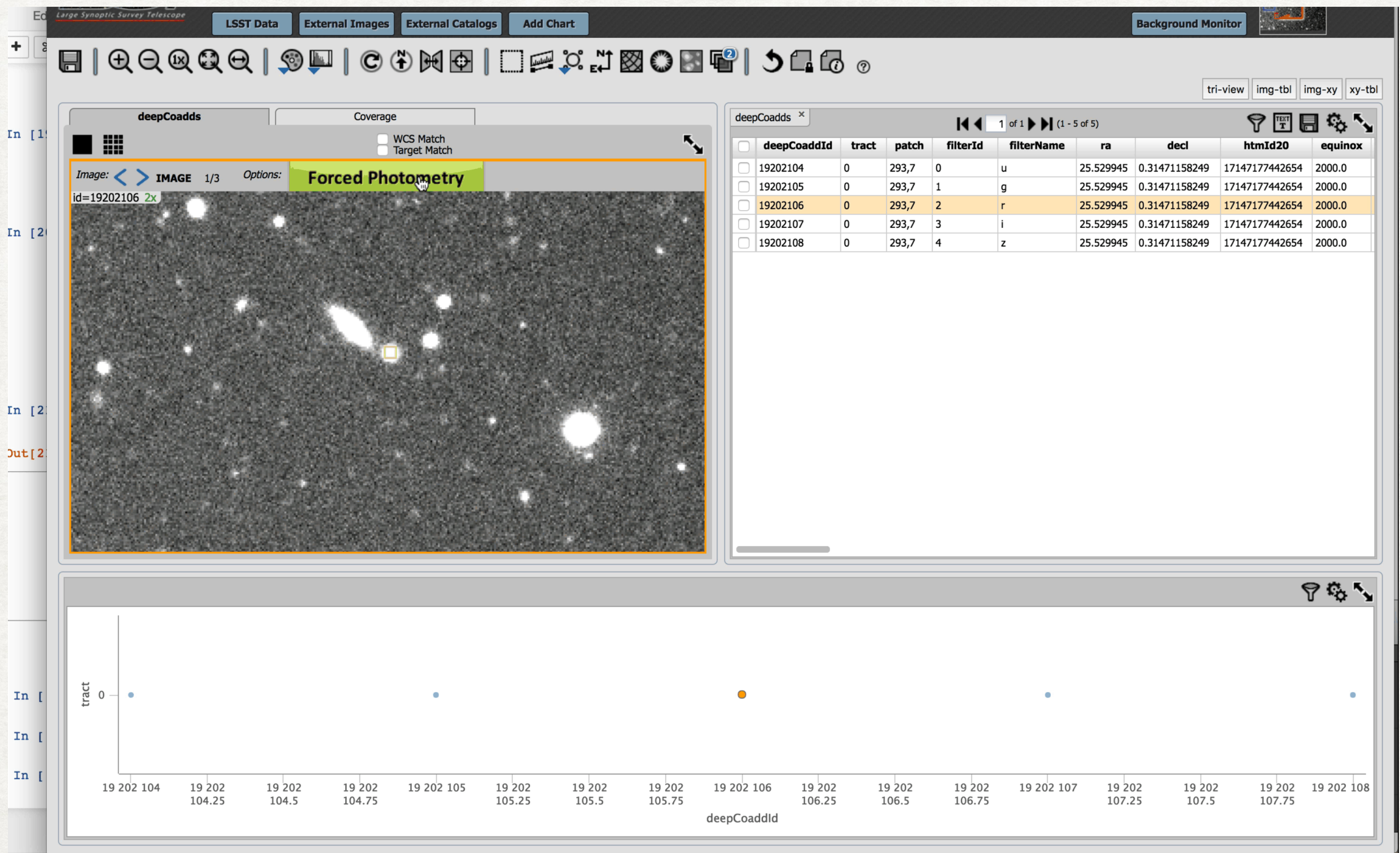
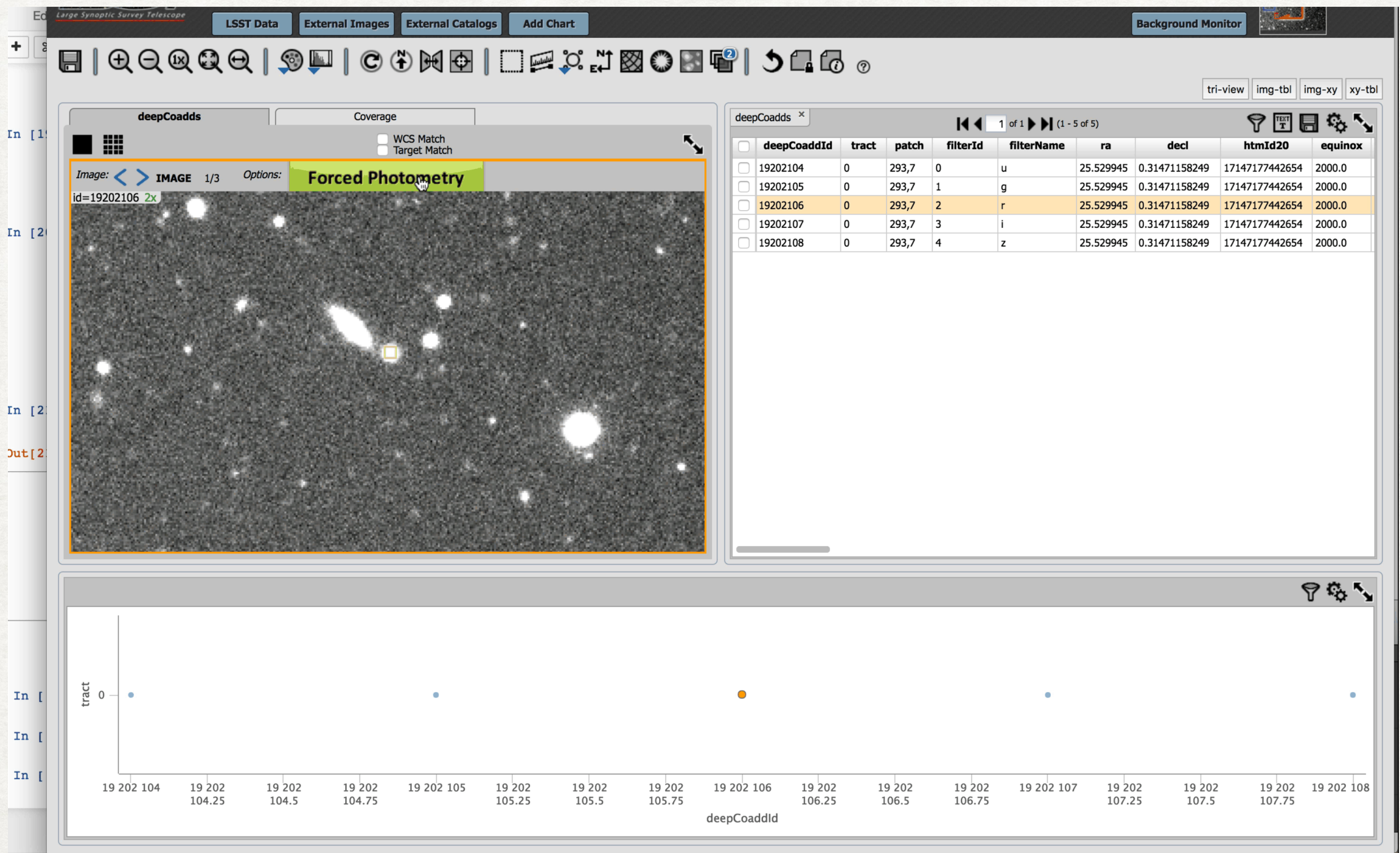


Table of results is returned to the browser



Python widgets for astronomy visualization

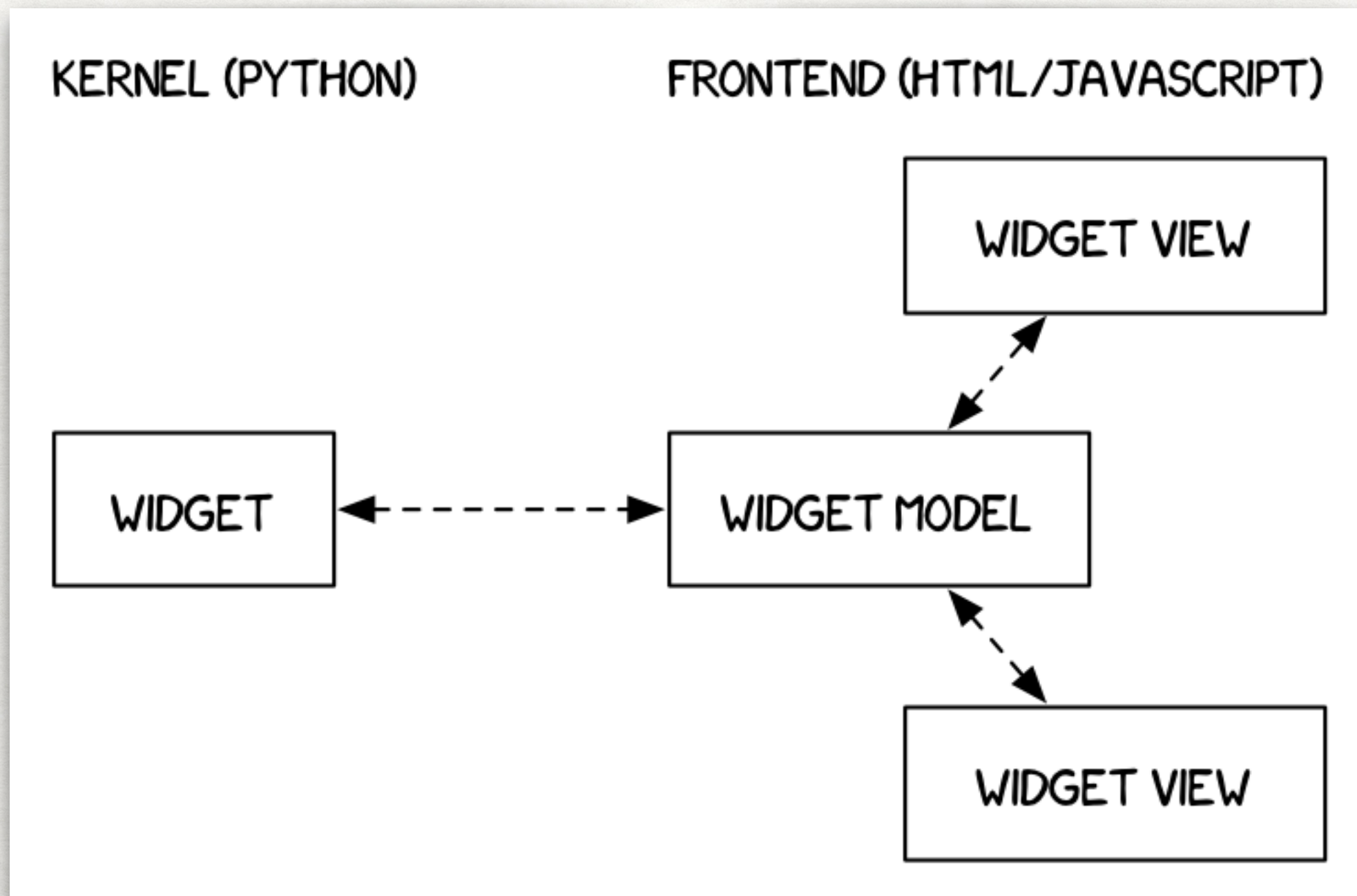
- The Firefly visualization system
- APIs to Firefly
- Example: Python interaction with the Firefly app
- Firefly widgets for Jupyter
- Resources

What are widgets?

Eventful Python objects that have a representation in the browser

What are widgets?

Eventful Python objects that have a representation in the browser

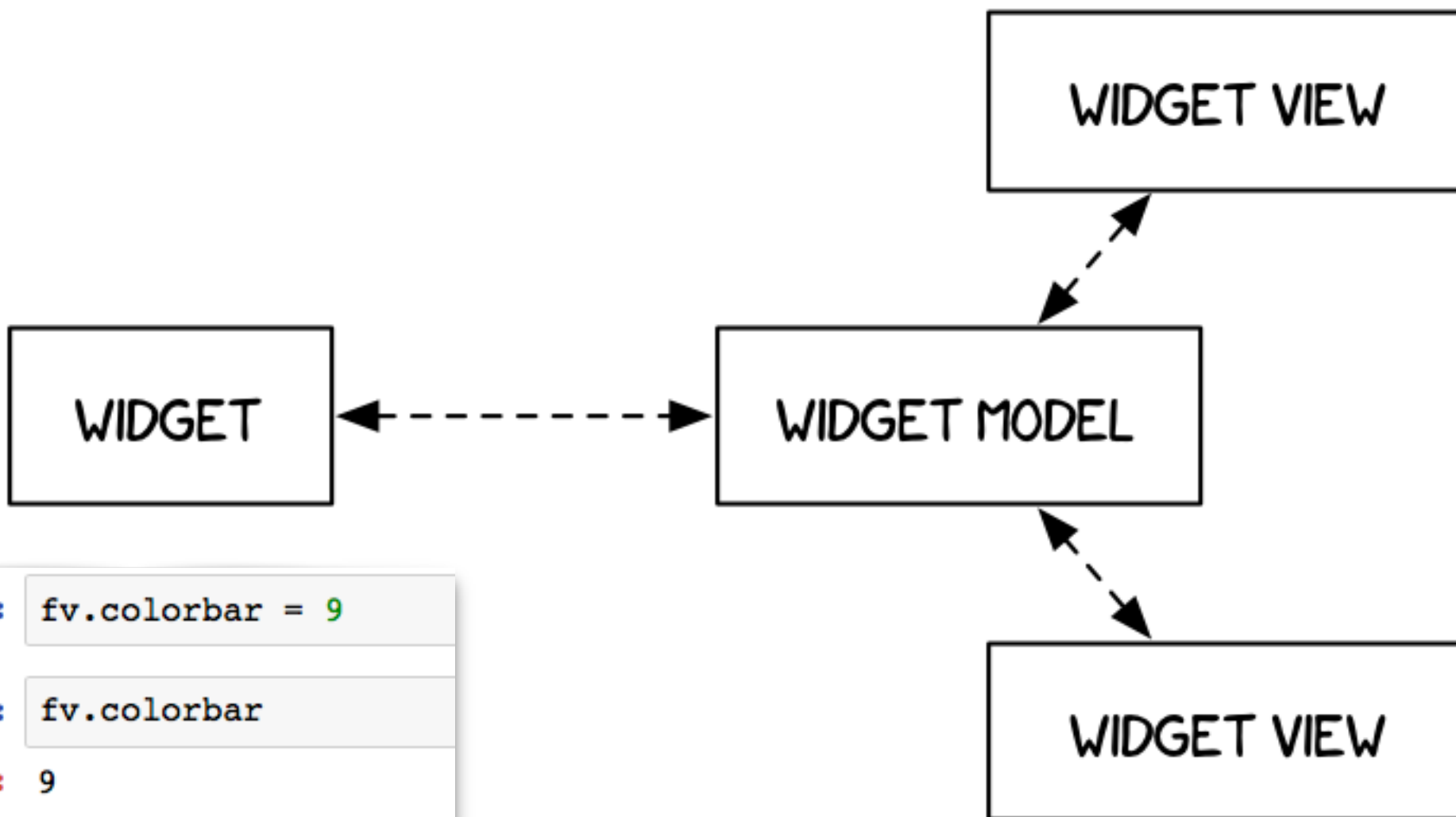


What are widgets?

Eventful Python objects that have a representation in the browser

KERNEL (PYTHON)

FRONTEND (HTML/JAVASCRIPT)



```
In [15]: fv.colorbar = 9
```

```
In [16]: fv.colorbar
```

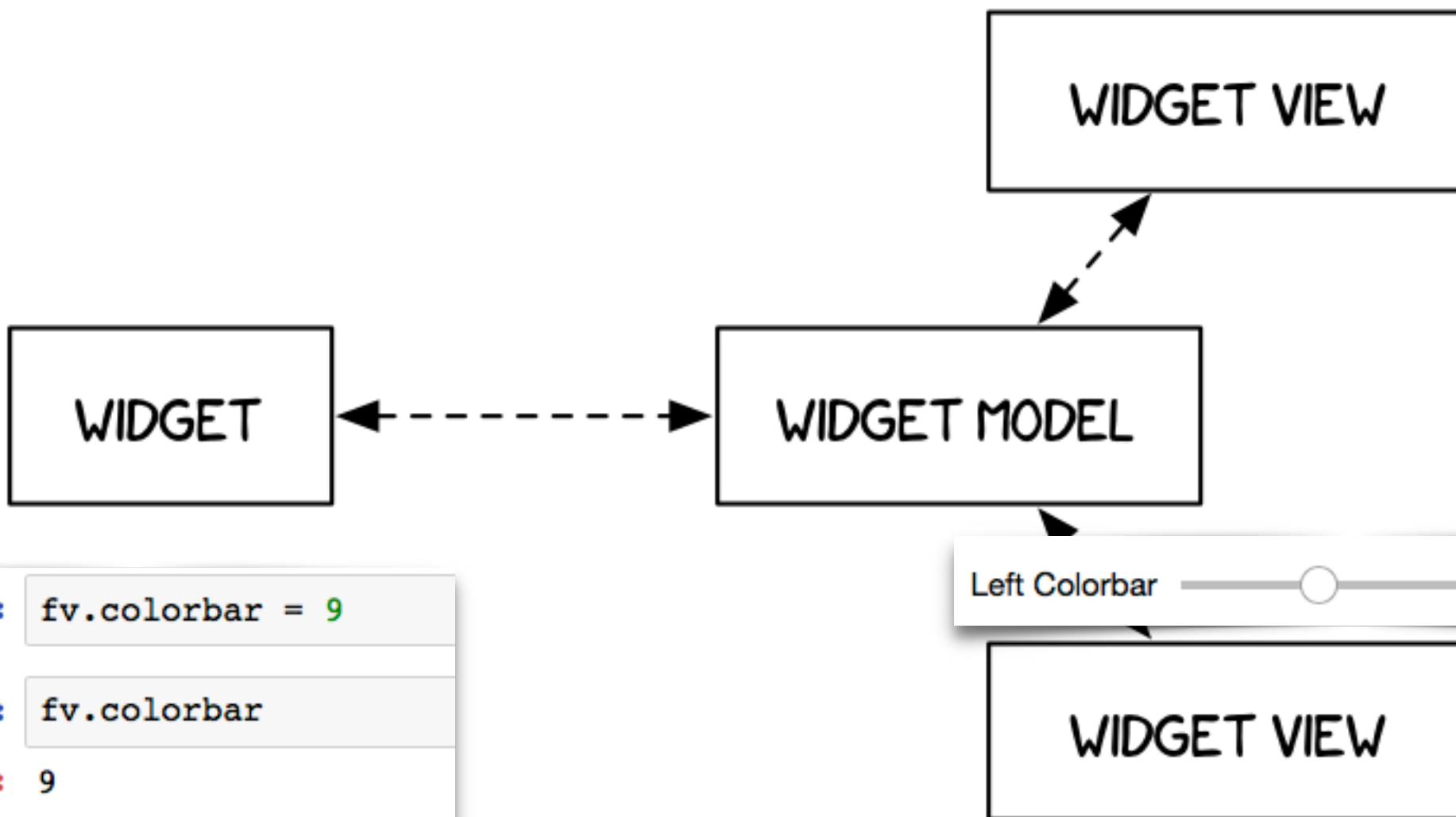
```
Out[16]: 9
```


What are widgets?

Eventful Python objects that have a representation in the browser

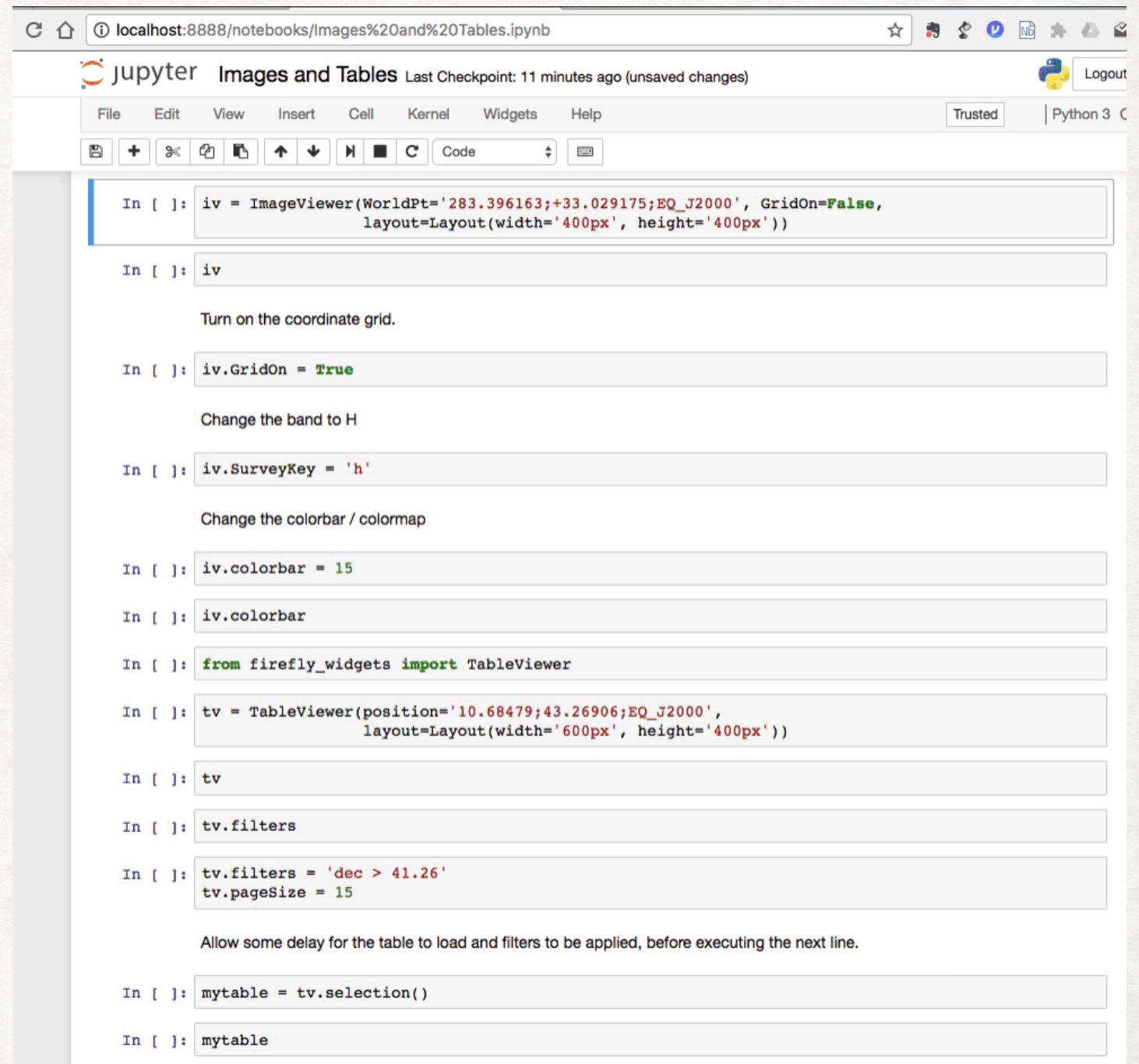
KERNEL (PYTHON)

FRONTEND (HTML/JAVASCRIPT)



Firefly Widgets

- Available as **firefly_widgets**
- Interactive widgets for Images and Tables
- Widgets currently just past experimental state



```
In [ ]: iv = ImageViewer(WorldPt='283.396163;+33.029175;EQ_J2000', GridOn=False,
                        layout=Layout(width='400px', height='400px'))

In [ ]: iv

Turn on the coordinate grid.

In [ ]: iv.GridOn = True

Change the band to H

In [ ]: iv.SurveyKey = 'h'

Change the colorbar / colormap

In [ ]: iv.colorbar = 15

In [ ]: iv.colorbar

In [ ]: from firefly_widgets import TableViewer

In [ ]: tv = TableViewer(position='10.68479;43.26906;EQ_J2000',
                        layout=Layout(width='600px', height='400px'))

In [ ]: tv

In [ ]: tv.filters

In [ ]: tv.filters = 'dec > 41.26'
tv.pageSize = 15

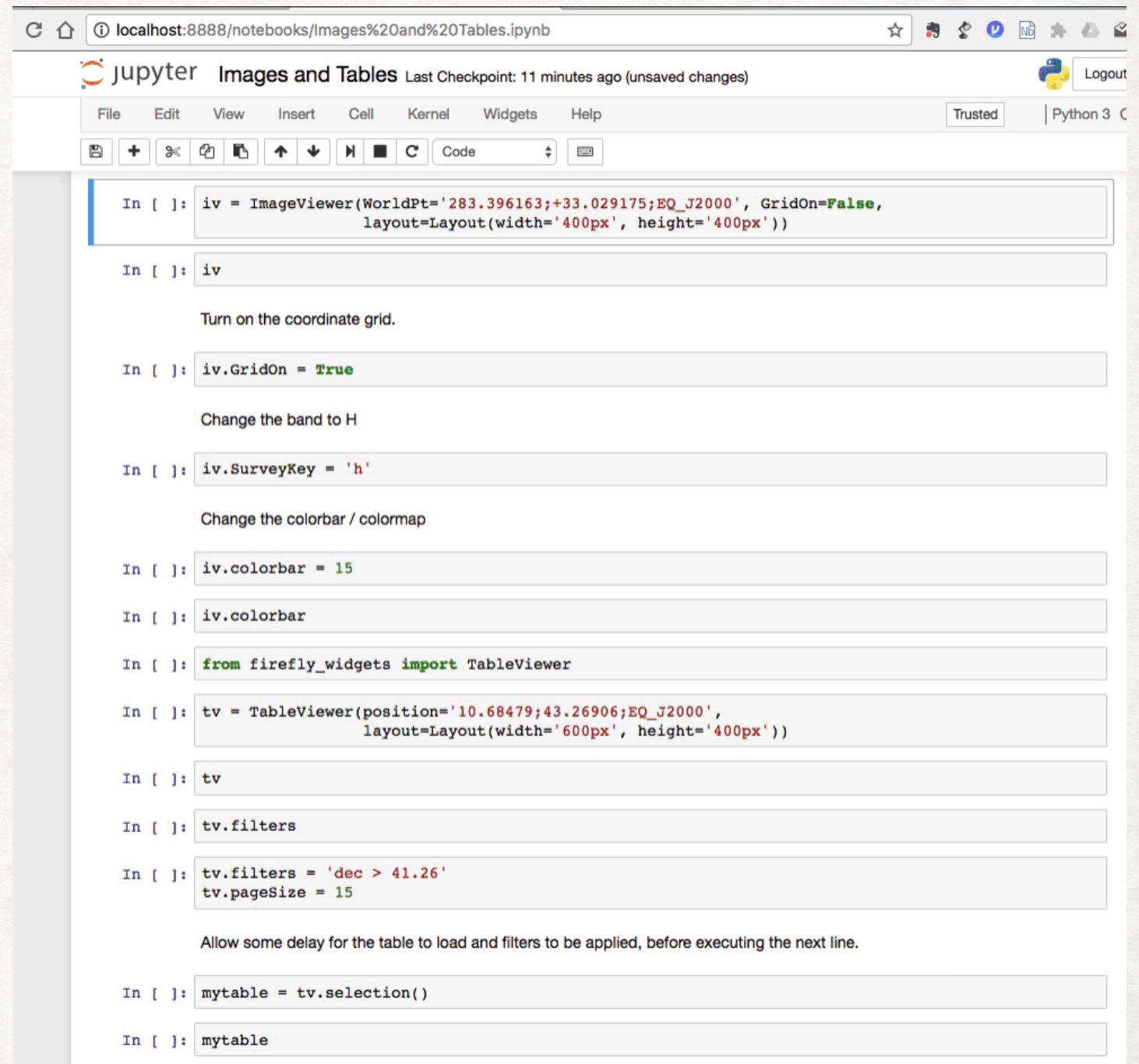
Allow some delay for the table to load and filters to be applied, before executing the next line.

In [ ]: mytable = tv.selection()

In [ ]: mytable
```


Firefly Widgets

- Available as **firefly_widgets**
- Interactive widgets for Images and Tables
- Widgets currently just past experimental state



```
In [ ]: iv = ImageViewer(WorldPt='283.396163;+33.029175;EQ_J2000', GridOn=False,
                        layout=Layout(width='400px', height='400px'))

In [ ]: iv

Turn on the coordinate grid.

In [ ]: iv.GridOn = True

Change the band to H

In [ ]: iv.SurveyKey = 'h'

Change the colorbar / colormap

In [ ]: iv.colorbar = 15

In [ ]: iv.colorbar

In [ ]: from firefly_widgets import TableViewer

In [ ]: tv = TableViewer(position='10.68479;43.26906;EQ_J2000',
                        layout=Layout(width='600px', height='400px'))

In [ ]: tv

In [ ]: tv.filters

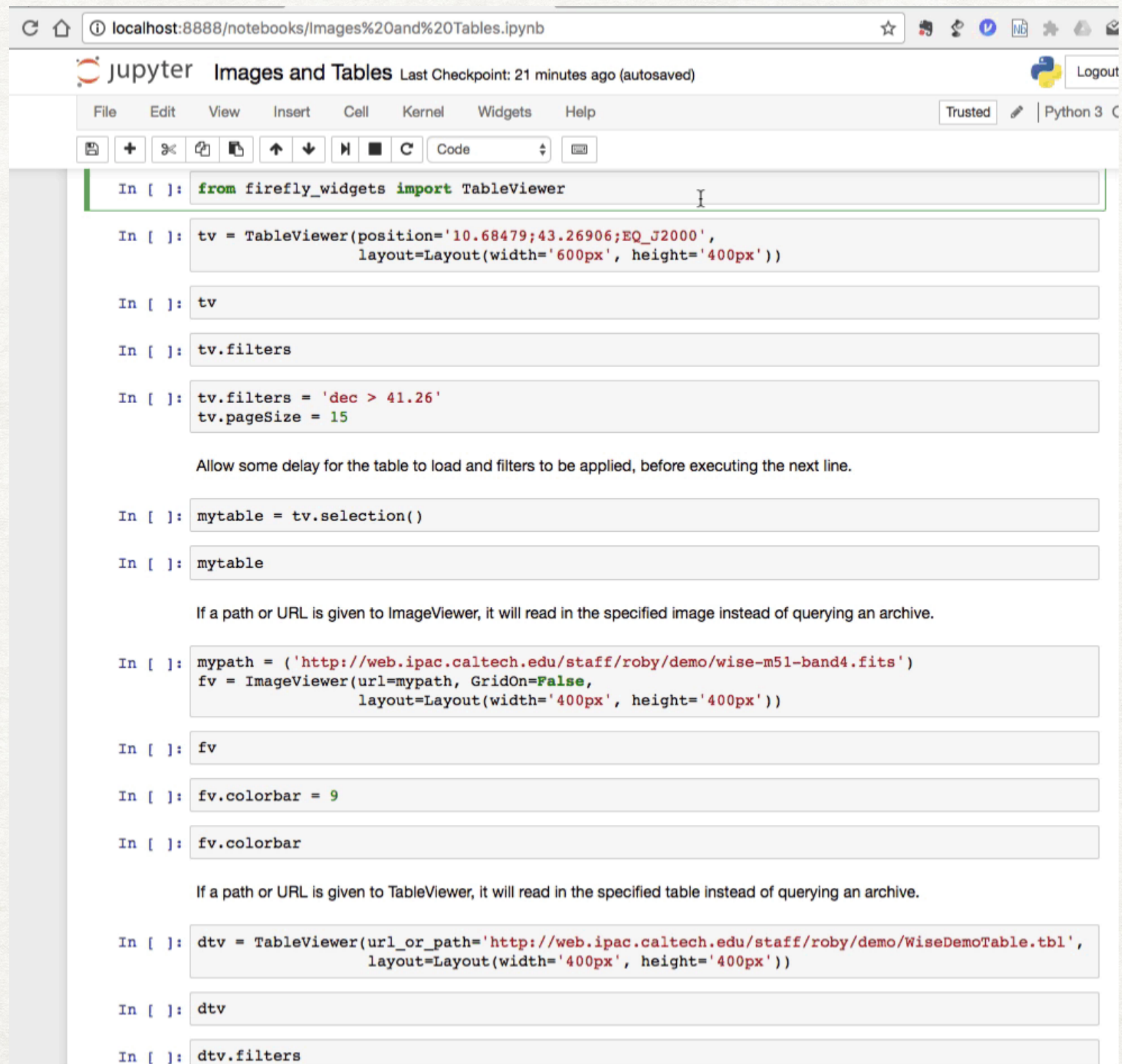
In [ ]: tv.filters = 'dec > 41.26'
tv.pageSize = 15

Allow some delay for the table to load and filters to be applied, before executing the next line.

In [ ]: mytable = tv.selection()

In [ ]: mytable
```


Interactive Table Widget



The screenshot shows a Jupyter Notebook titled "Images and Tables" running on localhost:8888. The notebook contains several code cells demonstrating the use of the `firefly_widgets` library. The code includes importing `TableViewer`, creating a `TableViewer` instance with specific position and layout parameters, setting filters and page size, selecting data, and creating an `ImageViewer` instance with a URL and layout parameters. The notebook also includes explanatory text about the parameters used in the code.

```
In [ ]: from firefly_widgets import TableViewer

In [ ]: tv = TableViewer(position='10.68479;43.26906;EQ_J2000',
                        layout=Layout(width='600px', height='400px'))

In [ ]: tv

In [ ]: tv.filters

In [ ]: tv.filters = 'dec > 41.26'
tv.pageSize = 15

Allow some delay for the table to load and filters to be applied, before executing the next line.

In [ ]: mytable = tv.selection()

In [ ]: mytable

If a path or URL is given to ImageViewer, it will read in the specified image instead of querying an archive.

In [ ]: mypath = ('http://web.ipac.caltech.edu/staff/robby/demo/wise-m51-band4.fits')
fv = ImageViewer(url=mypath, GridOn=False,
                layout=Layout(width='400px', height='400px'))

In [ ]: fv

In [ ]: fv.colorbar = 9

In [ ]: fv.colorbar

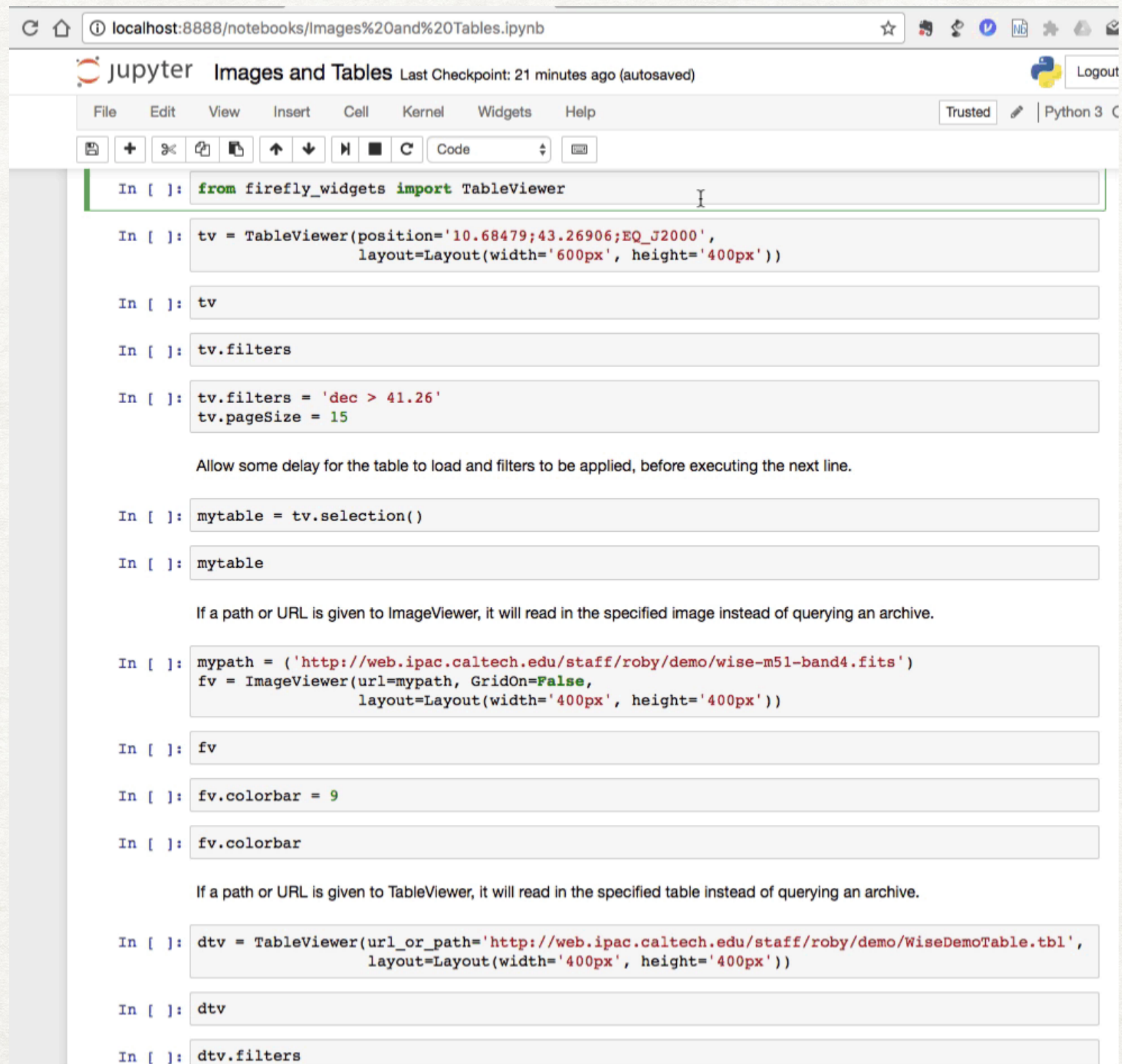
If a path or URL is given to TableViewer, it will read in the specified table instead of querying an archive.

In [ ]: dtv = TableViewer(url_or_path='http://web.ipac.caltech.edu/staff/robby/demo/WiseDemoTable.tbl',
                        layout=Layout(width='400px', height='400px'))

In [ ]: dtv

In [ ]: dtv.filters
```


Interactive Table Widget



The screenshot shows a Jupyter Notebook titled "Images and Tables" running on localhost:8888. The notebook contains several code cells demonstrating the use of the `firefly_widgets` library. The code includes importing `TableViewer`, creating a `TableViewer` instance with specific position and layout parameters, setting filters and page size, selecting data, and creating an `ImageViewer` instance with a URL and layout parameters. The notebook also includes explanatory text about the parameters used in the `TableViewer` and `ImageViewer` constructors.

```
In [ ]: from firefly_widgets import TableViewer

In [ ]: tv = TableViewer(position='10.68479;43.26906;EQ_J2000',
                        layout=Layout(width='600px', height='400px'))

In [ ]: tv

In [ ]: tv.filters

In [ ]: tv.filters = 'dec > 41.26'
tv.pageSize = 15

Allow some delay for the table to load and filters to be applied, before executing the next line.

In [ ]: mytable = tv.selection()

In [ ]: mytable

If a path or URL is given to ImageViewer, it will read in the specified image instead of querying an archive.

In [ ]: mypath = ('http://web.ipac.caltech.edu/staff/robby/demo/wise-m51-band4.fits')
fv = ImageViewer(url=mypath, GridOn=False,
                layout=Layout(width='400px', height='400px'))

In [ ]: fv

In [ ]: fv.colorbar = 9

In [ ]: fv.colorbar

If a path or URL is given to TableViewer, it will read in the specified table instead of querying an archive.

In [ ]: dtv = TableViewer(url_or_path='http://web.ipac.caltech.edu/staff/robby/demo/WiseDemoTable.tbl',
                        layout=Layout(width='400px', height='400px'))

In [ ]: dtv

In [ ]: dtv.filters
```


Synchronize widget properties with other widgets

localhost:8888/notebooks/Image%20Colorbar%20Test.ipynb

jupyter Image Colorbar Test Last Checkpoint: a few seconds ago (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 C

In [9]:

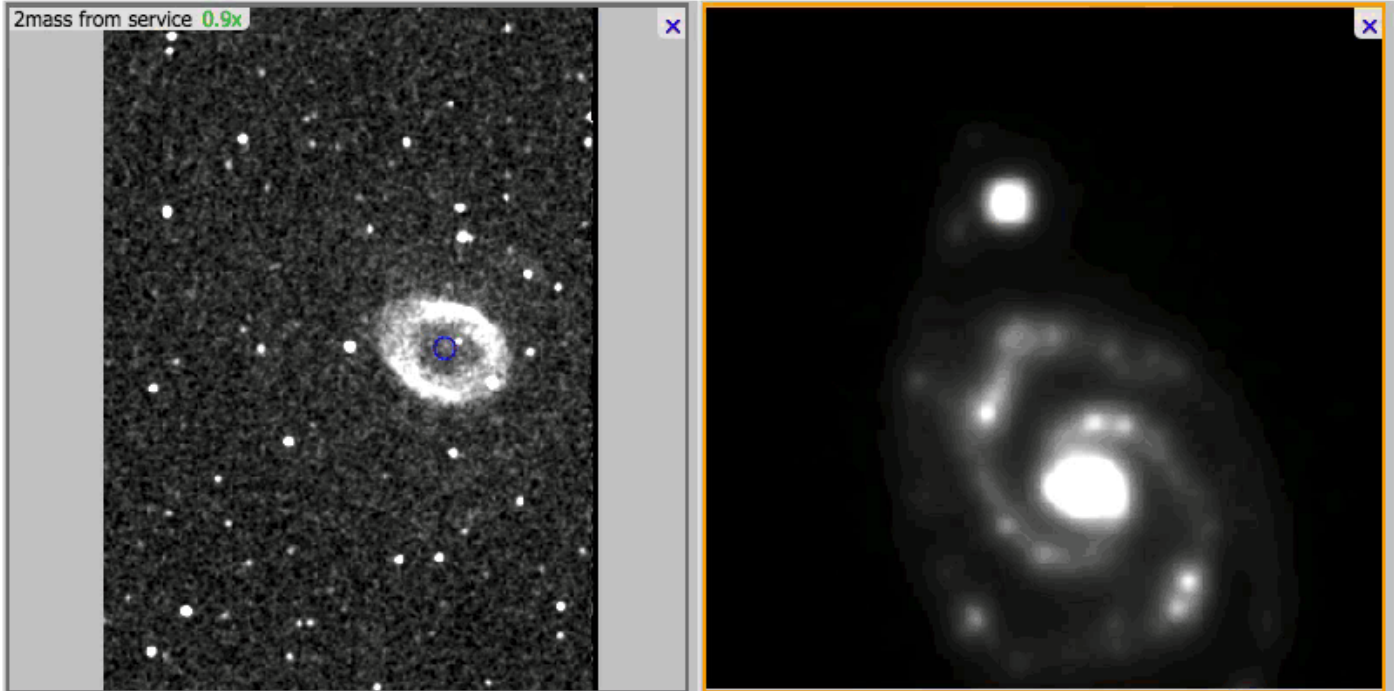
```
left_link = jslink((iw1, 'value'), (iv, 'colorbar'))
right_link = jslink((iw2, 'value'), (fv, 'colorbar'))
both_link = jslink((iw1, 'value'), (iw2, 'value'))
```

In [10]:

```
HBox([VBox([iw1, iv]), VBox([iw2, fv])])
```

Left Colorbar 0 Right Colorbar 0

2mass from service 0.9x



Unlink left column and right column

In []: both_link.unlink()

Re-link left column and right column

In []: both_link = jslink((iw1, 'value'), (iw2, 'value'))

Change the colorbar / colormap

Synchronize widget properties with other widgets

localhost:8888/notebooks/Image%20Colorbar%20Test.ipynb

jupyter Image Colorbar Test Last Checkpoint: a few seconds ago (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 C

In [9]:

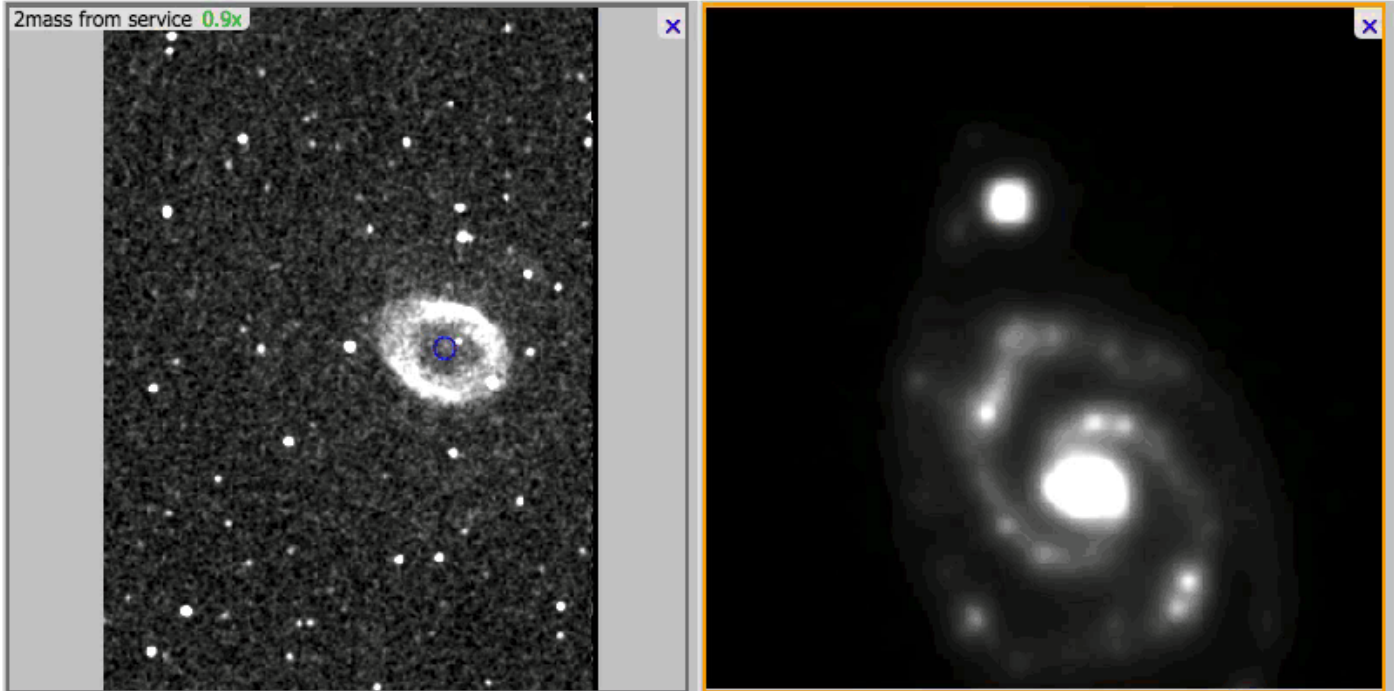
```
left_link = jslink((iw1, 'value'), (iv, 'colorbar'))
right_link = jslink((iw2, 'value'), (fv, 'colorbar'))
both_link = jslink((iw1, 'value'), (iw2, 'value'))
```

In [10]:

```
HBox([VBox([iw1, iv]), VBox([iw2, fv])])
```

Left Colorbar 0 Right Colorbar 0

2mass from service 0.9x



Unlink left column and right column

In []: both_link.unlink()

Re-link left column and right column

In []: both_link = jslink((iw1, 'value'), (iw2, 'value'))

Change the colorbar / colormap

Lessons learned from widget development

Lessons learned from widget development

- Use the cookiecutter template for Jupyter widgets
 - incorporates best practices for packaging
 - makes installation straightforward

Lessons learned from widget development

- Use the cookiecutter template for Jupyter widgets
 - incorporates best practices for packaging
 - makes installation straightforward
- Development is light on Python side, heavy on Javascript

Lessons learned from widget development

- Use the cookiecutter template for Jupyter widgets
 - incorporates best practices for packaging
 - makes installation straightforward
- Development is light on Python side, heavy on Javascript
- Only some actions are good candidates for widget attributes
 - If action is done on the client — can change quickly
 - If action calls the server — can be slow

Python widgets for astronomy visualization

- The Firefly visualization system
- APIs to Firefly
- Example: Python interaction with the Firefly app
- Firefly widgets for Jupyter
- Resources

Resources for Firefly

Resources for Firefly

- Instructional videos at IRSA IPAC channel on YouTube

Resources for Firefly

- Instructional videos at IRSA IPAC channel on YouTube
- Source code repositories with example notebooks
firefly, firefly_client, firefly_widgets

Resources for Firefly

- Instructional videos at IRSA IPAC channel on YouTube
- Source code repositories with example notebooks
firefly, firefly_client, firefly_widgets
- Documentation for Javascript and Python APIs

Resources for Firefly

- Instructional videos at IRSA IPAC channel on YouTube
- Source code repositories with example notebooks
firefly, firefly_client, firefly_widgets
- Documentation for Javascript and Python APIs
- Demo videos made for the Large Synoptic Survey telescope

Resources for Firefly

- Instructional videos at IRSA IPAC channel on YouTube
- Source code repositories with example notebooks
firefly, firefly_client, firefly_widgets
- Documentation for Javascript and Python APIs
- Demo videos made for the Large Synoptic Survey telescope
- See blog post at astropython.wordpress.com