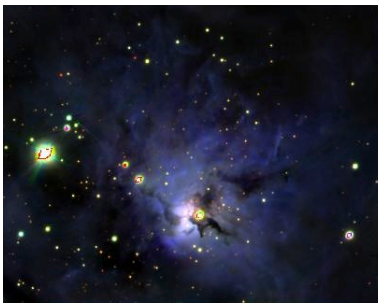
PS1 Image Cutout Service

A simple user interface provides access to the processed Pan-STARRS images. Stack (combined) images are available for the grizy filters in the 3PI survey; beware that the values in these images are related non-linearly to flux as discussed below. The stack images are also combined to create color images. Images may be extracted using an RA+Dec position or using an object name. The resulting cutout images (postage stamps) are available as JPEG images within the web browser or as FITS images. The interface also provides access to download skycell images covering larger sky regions, along with an interactive display for browsing skycell images. Finally, there are web services that can be used for access to the images via scripts.

Single-epoch "warp" images are also available as of the DR2 data release (2019 January 28).



Color 6.9x5.3 arcmin image of M8 using the PS1 riz filters created using the image cutout interface.

PS1 Image Cutouts	
Cutouts	http://ps1images.stsci.edu/cgi-bin/ps1cutouts
Info	http://ps1images.stsci.edu/cgi-bin/ps1filenames.py

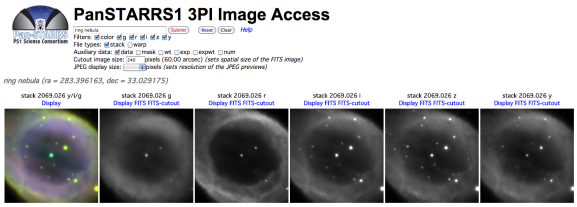
Contents

- The web interface
  - Changing the parameters
  - Warp (single epoch) images
  - Access to FITS images
  - Important FITS image format, WCS, and flux-scaling notes
  - Interactive image display
- Scripted image downloads and image cutout extractions
  - Image List Service
  - Download a FITS File
  - Image Cutouts
  - Python Example Script
  - Bulk Image Download Python Script
- Plans for future additions & improvements

The starting point for the PS1 data archive is at Pan-STARRS1 data archive home page.

The web interface

The PS1 images are accessible through http://ps1images.stsci.edu/cgi-bin/ps1cutouts. The interface has a text box to enter a position or object name plus a few other options. Type an object name or position into the box and click **Submit**:



The search string can be a name that is recognized by NED or SIMBAD (e.g., "ring nebula" as above) or a position in various formats (RA and Dec in degrees, hh mm ss +dd mm ss, h:m:s d:m:s, etc.). A PanSTARRS image identifier (e.g., skycell.2069.026) is also accepted and is translated to the central position of that image.

The labels above each image identify the image type (always "stack" for the current DR1 release), the image identifier (these come from skycell.2069.026), and the filter.

Changing the parameters

The default parameters for the search return 240x240 pixel cutout images (=1x1 arcmin for the PS1 scale of 0.25 arcsec/pixel), including a color image and all five filters, grizy. The filters used for the color image are the reddest (y), the bluest (g), and the middle image (i).

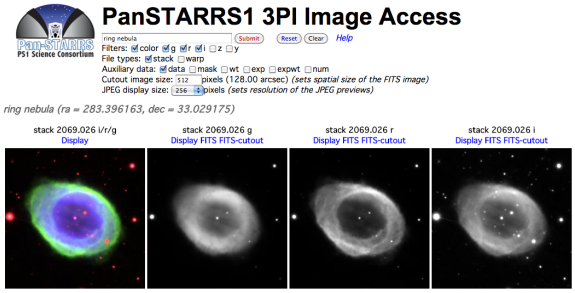
The parameters allow changing the defaults:

Parameter	Description
Filters	Select a subset of the filters using check boxes. If the color box is checked and two or more of the filters are checked, a color image is also shown using the selected filters.
File types	Include the warp (single epoch) images in addition to or instead of the stack images.

Auxiliary data	Include ancillary images such as masks, weights, and exposure times. These will show up as separate rows in the results.
Cutout image size	Specify the extracted image size in pixels. The conversion to arcsec is shown. This sets the spatial size of the region that is extracted from the PS1 image. If this size is large, the cutout image may reach the edge of the PS1 image; in that case the pixels off the edge are filled with blanks. The skycell images are approximately 6000x6000 pixels, so sizes larger than that will always reach the edges of the images. The size is currently limited to 6000 pixels.
JPEG display size	Select the size for the displayed JPEG images. The cutout is scaled to the selected size. This can be used either to shrink the images when the cutout size is large (to fit larger fields on the web page) or to expand the image for small cutouts (to see finer details).

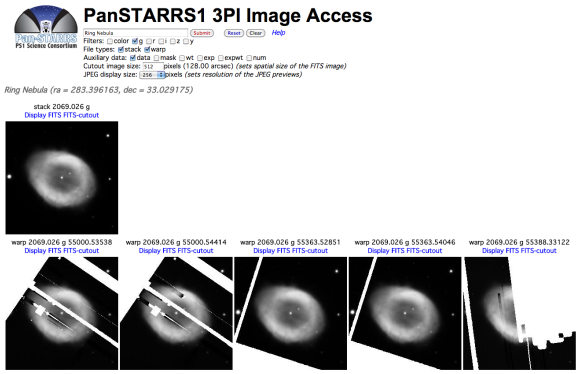
The **Submit** button loads a new web page for the requested sky position and other parameters. Note that the URL of the page encodes all the parameters, so it is possible to copy it and save it for future access to the same image cutouts. The **Reset** button restores all parameters to their values when the current page was submitted. The **Clear** button restores all parameters to their default values from the original web page.

As an example, here are images for the Ring Nebula with the *cutout image size* increased to 512 pixels (128 arcsec), the *JPEG display size* set to 256 pixels (so the JPEG scale is 0.5 arcsec/pixel), and the gri filters selected (which changes the color image as well):



Warp (single epoch) images

If the *warp* box is checked, the results include all single-epoch PS1 images that overlap the requested sky position. Here is an example for the Ring Nebula g-band images:



The result page shows the stack image(s) followed by all the warp images. The warps are grouped by filter (grizy) and are sorted by epoch of observation. Note that the individual warps often show many regions of missing pixels due to gaps between detectors, bad pixel regions, etc. These bad pixels are rejected during stacking, and the warp pointing centers are dithered so that the final quality of the stack images is generally very good. Obviously users will need to take care in using the single-epoch images!

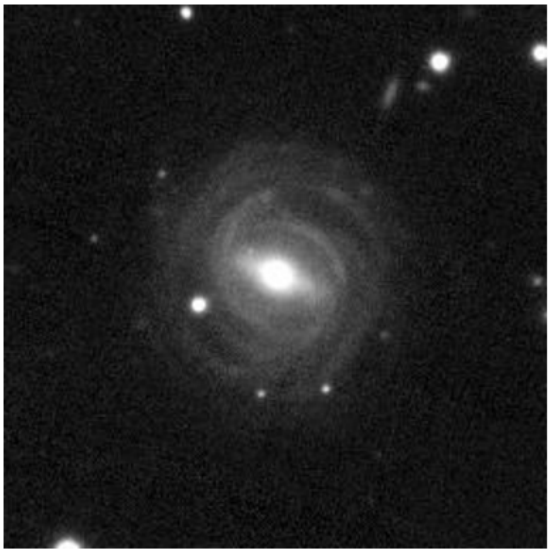
There are usually a large number (~20) of warp images for each filter. Some of the warp images are completely blank because the actual sky coverage for the exposure does overlap the large skycell image but does not necessarily cover the cutout image region. Currently our PS1 image database does not have a detailed map of the precise sky coverage for individual exposures, so the warp images that lack sky coverage in a region are not known before the cutout is extracted. That may be improved in future versions of the interface.

Access to FITS images

In addition to the titles that identify the cutouts, there are links above the images. Here is the result for NGC 7222:

stack 1405.053 g

Display FITS FITS-cutout



The ***FITS*** link downloads the full skycell image for this field. In this example that is a 6240x6243 pixel image (~0.43 degrees across) of the g filter stack for skycell 1405.053. Note that these images are compressed using the FITS tile-compressed image convention. The format can be read by most FITS libraries or can be converted back to a standard uncompressed FITS image using the funpack utility.

The ***FITS-cutout*** link downloads a FITS image of only the cutout region. These images are **not** resampled to the JPEG display size – they have the full resolution of the original images. So e.g. if the *cu tout image size* is 900 pixels, the FITS cutout will be 900x900 pixels. FITS cutout images are not compressed.

The FITS links are not available for color images. Since the various filter images for a skycell are matched pixel-by-pixel, the individual ***FITS*** and ***FITS-cutout*** filter images can be downloaded separately and assembled into multi-band FITS images if desired.

Important FITS image format, WCS, and flux-scaling notes

The PS1 FITS images have a number of unusual characteristics (**Note: this applies only to the full fits images, not the cutouts!**):

- As mentioned above, **the images are compressed using the FITS tile-compressed image convention**. Some older FITS software may not be able to read them, although current FITS libraries (including cfitsio, astropy.io.fits, and the IDL astronomy library) can handle them. The format can be converted to a standard uncompressed FITS image using the funpack utility.

- Another quirk of these images is that **they use the obsolete WCS keywords** PC001001, PC001002, PC002001, PC002002 instead of the FITS standard keywords CD1_1, CD1_2, etc. Many software packages automatically handle the old PC keywords, but some require special processing (e.g., in IDL you should call the fits_cd_fix procedure to modify the header).

- Another issue with the image astrometry is that **the full skycell FITS images do not have a RADESYS keyword**. That leads some software to incorrectly interpret the coordinates as equinox 1950 rather than equinox 2000. At the moment the only known software with this issue is DS9 v8, but it could happen with other software as well. The fix is to insert the keyword RADESYS = 'FK5' in the header. **FITS cutout images have a correct RADESYS keyword** (as of 2019 March 13), but full skycell FITS images do not.

- Yet another header issue with the FITS keywords for the warp images is that the **observation times are given as international atomic time (TAI) rather than UTC time**. Those times differ by the addition of leap seconds, which leads to header times that differ by 34 or 35 seconds from the UTC times. (See Rots et al. 2015 for more details.) The fix for this is to insert the keyword TIMESYS = 'TAI' in the header. **FITS cutout images have a correct TIMESYS keyword** (as of 2022 January 20), but **full skycell FITS images do not have a TIMESYS keyword**. This is important if you care about timing at an accuracy of 30 seconds. Thanks to Peter Van Wylen for discovering this issue and identifying the fix.

- **The image flux scaling is also non-standard for the stack images.** The images have been non-linearly scaled using an arc hyperbolic sine (asinh) transformation that converts them to a pseudo-magnitude scale related to the asinh magnitudes (aka "luptitudes") that are used in the Sloan Digital Sky Survey. The scaling is determined by the BSOFTEN/BOFFSET keywords in the FITS header. Here are the relevant lines from the header of the skycell 1725.051 g-band image:

```
BZERO   =   3.283630371094E+00 / Scaling: TRUE = BZERO + BSCALE * DISK
BSCALE  =   2.008622776974E-04 / Scaling: TRUE = BZERO + BSCALE * DISK
BSOFTEN =   8.739592975627E+01 / Scaling: LINEAR = 2 * BSOFTEN * sinh
(TRUE/a)
BOFFSET =   2.654963016510E+00 / Scaling: UNCOMP = BOFFSET + LINEAR
```

  The comments describe the transformation to convert the pixel values to fluxes, but they are not very clear. BZERO and BSCALE are the standard FITS keywords for converting an integer image to a floating point value. The constant *a* is defined below. "UNCOMP" is the linear flux value. If j is the original integer pixel value, these equations convert to the float pixel value v and then to a standard linear flux:

```
v = BZERO + BSCALE * j
a = 2.5/ln(10) = 1.0857362
x = v/a
flux = boffset + bsoften * 2 * sinh(x)
     = boffset + bsoften * (exp(x) - exp(-x))
     = boffset + bsoften * (10**(0.4*v) - 10**(-0.4*v))
```

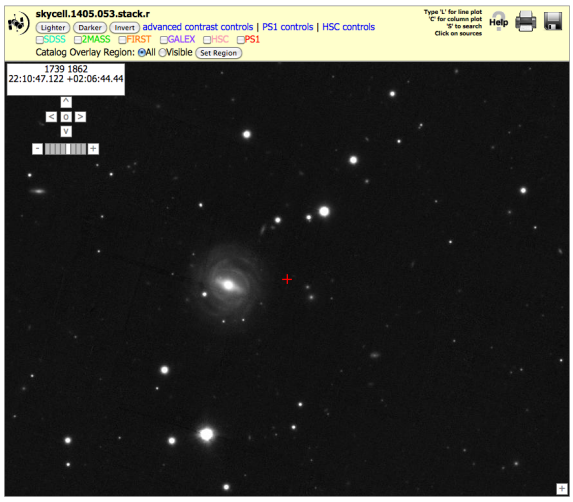
  The 3 equations for the flux are equivalent. The third equation shows the connection of the values to asinh magnitudes.

  Note that the asinh transformation is applied only to the stacked images. **The single-epoch warp images do not use asinh scaling.** Check for the presence of the BSOFTEN and BOFFSET keywords in the header to determine whether or not the sinh correction is required.

All four of the above issues apply to the full skycell images. **Image cutouts (retrieved using the *FITS-cutout* link) are not compressed, include a RADESYS keyword, and have already been converted to a standard linear flux scale by applying the BSOFTEN/BOFFSET equation given above**. Cutout images do still have the obsolete WCS keywords (although that may change in the future).

Interactive image display

The ***Display*** link opens an interactive FITS image viewer that allows browsing the skycell image. The interactive viewer (which was originally developed for the Hubble Legacy Archive) allows panning, zooming and changing the image contrast. It is also possible to overlay external catalogs such as SDSS and 2MASS on the image. There is a Help page that describes the standard functionality of this tool. Here is a screenshot of the display in the vicinity of NGC 7222:



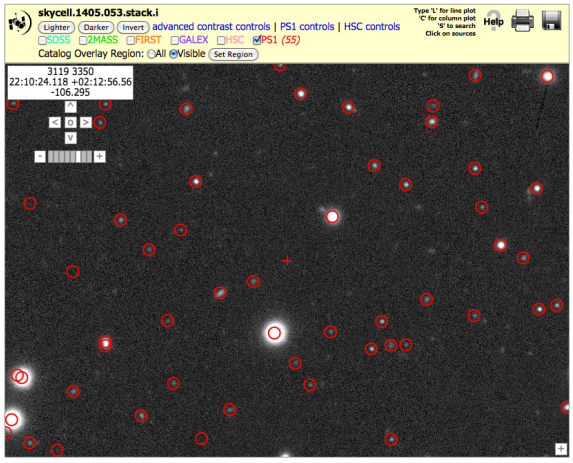
The ***Display*** link is also available for color images. Note that the compressed storage format of the FITS image makes the interactive display somewhat sluggish, especially for the color images (which are reading FITS images for all 3 filters being used). Options are being explored to improve the performance (which is currently tolerable if not ideal).

There is one custom catalog available for the PS1 image display: the **PS1** checkbox overlays objects from the PanSTARRS catalog.

When catalogs are overlaid on the image, you can click on individual sources to see magnitudes and other properties.

A useful option to speed up catalog loading, especially in crowded regions, is to change the ***Catalog Overlay Region***. The default is to load all the sources that overlap the skycell. Click the ***Visible*** button and then the ***Set Region*** button to get only the catalog sources that are in the currently visible region of the image. That is much faster when only a small portion of the image is being viewed. The selected region remains fixed as you zoom and pan the image until you click the ***Set Region*** button again to change it to match the current viewport. When the region is changed, any catalogs that have been overlaid on the image are automatically reloaded.

Additional options for the PS1 catalog are available by clicking on ***PS1 controls***. Currently the only option is to filter the catalog using the number of filters & epochs at which objects were detected. That is useful for filtering out single-epoch detections, which are often spurious. The default is nDetections>2, which produces a clean catalog but omits some faint objects.



Scripted image downloads and image cutout extractions

This section describes the web services that can be used to extract images using a script. See below for an example Python script that uses these services to find and download FITS and JPEG cutout images. Please ask archive@stsci.edu if you need more details.

There are three basic services for accessing the PS1 images:

Service	URL	Description
Get list of images	http://ps1images.stsci.edu/cgi-bin/ps1filenames.py	Returns a table with a list of available images at a given RA, Dec position or for a given skycell.
Download a file	http://ps1images.stsci.edu/*<filename>*	Directly access a single file identified using the above query.
Get image cutout	http://ps1images.stsci.edu/cgi-bin/fitscut.cgi	Extract a cutout image with a portion of a particular image. The image can be returned in FITS or JPEG format. For JPEG images, the image can be shrunk or expanded to a target output size.

Image List Service

The image list service, http://ps1images.stsci.edu/cgi-bin/ps1filenames.py, has a few parameters to determine which images are returned:

Parameters	Example	Values	Description
skycell	skycell=1405.053	Any PS1 mmmm.nnn skycell	Return images associated with a known PS1 skycell

ra, dec	ra=334.0&dec=2.0	J2000 RA & Dec in degrees	Alternative to skycell: Return the image that covers the specified RA /Dec position.
filters	filters=griz	Any subset of grizy	Return only images from the specified filters. Default is all 5 PS1 filters (grizy).
type	type=stack, stack.wt	Comma-separated list of many choices	Include various auxiliary images. Default is stack; some common choices include warp (single-epoch images), stack.wt (weight image), stack.mask, stack.exp (exposure time), warp.wt, and warp.mask. The other available options are stack.num, stack.expwt, stack.psf, stack.mdc, stack.cmf, warp.cmf, and warp.mdc. Currently we do not have documentation for the different file types.
sep	sep=,	tab, comma, space	Separator used between columns in the output table. Default is space.

The Image List Service returns a table of values with a row of column headings followed by information on all images that match the requested criteria. If no images match, an empty table is returned. Here is a sample result from the query http://ps1images.stsci.edu/cgi-bin/ps1filenames.py?skycell=1405.053:

projcell	subcell	ra	dec	filter	mjd	type	filename	shortname
1405	53	332.600451657	2.19980905757	g	0.0	stack	/rings.v3.skycell/1405/053/rings.v3.skycell.1405.053.stk.g.unconv.fits	rings.v3.skycell.1405.053.stk.g.unconv.fits
1405	53	332.600451657	2.19980905757	i	0.0	stack	/rings.v3.skycell/1405/053/rings.v3.skycell.1405.053.stk.g.unconv.fits	rings.v3.skycell.1405.053.stk.i.unconv.fits
1405	53	332.600451657	2.19980905757	r	0.0	stack	/rings.v3.skycell/1405/053/rings.v3.skycell.1405.053.stk.g.unconv.fits	rings.v3.skycell.1405.053.stk.r.unconv.fits
1405	53	332.600451657	2.19980905757	y	0.0	stack	/rings.v3.skycell/1405/053/rings.v3.skycell.1405.053.stk.g.unconv.fits	rings.v3.skycell.1405.053.stk.y.unconv.fits
1405	53	332.600451657	2.19980905757	z	0.0	stack	/rings.v3.skycell/1405/053/rings.v3.skycell.1405.053.stk.g.unconv.fits	rings.v3.skycell.1405.053.stk.z.unconv.fits

The *projcell* and *subcell* columns give the skycell information, and the *RA* and *Dec* are the position. If the query specified a skycell as input, the *RA* and *Dec* are the central position of the image. The next three columns give the *filter*, *Modified Julian Date*, and the *type* of this image (stack, stack.wt, etc.) The *MJD* is zero for stack images (which are a combination of images taken at many different epochs). The 8th column is the *filename* for the image (which typically is very long with lots of gibberish). The last column, *shortname*, is a short filename that we are adopting: it includes the skycell, information on the type of image, and the filter.

Currently if RA & Dec are specified, only the best skycell image (where the position is farthest from the edge) is included. A possible future enhancement would be to include other nearby images. Note that the naming and positions of the images follows a regular tessellation pattern so that it is fairly simple to determine the names of neighboring images.

Download a FITS File

The filename returned by the above service can be used to retrieve the FITS image directly from the web server by prepending the server name, "http://ps1images.stsci.edu", before the "/data". So the URL to retrieve the first file in the table above is

http://ps1images.stsci.edu/rings.v3.skycell/1405/053/rings.v3.skycell.1405.053.stk.g.unconv.fits

Image Cutouts

The http://ps1images.stsci.edu/cgi-bin/fitscut.cgi interface can be used to retrieve cutout images in FITS or JPEG format. To get a single band image, use the filename from the above table as the ***red*** parameter of fitscut.cgi. Additional files using other filters can be specified for the ***green*** and ***blue*** parameters to get color JPEG images. (To get color FITS images, download the files directly.) The ***format*** parameter specifies the output format (***fits*** or ***jpeg***). This script has many additional parameters; see the HLA fitscut documentation for more details, including control of the position, cutout size, contrast, etc.

The image cutout interface works on any type of image (including masks, weights, etc.)

Python Example Script

There is a simple Jupyter notebook script that shows how to access the ps1filenames.py and fitscut.cgi scripts from Python. It is straightforward to retrieve both FITS image cutouts and JPEG /PNG cutouts. The script also shows an example of retrieving a color JPEG image. The script includes some simple helper functions for retrieving both FITS and JPEG/PNG images and shows how to display the images.

Sample JPEG images

This gets single-band grayscale and color JPEG images at the position of the Crab Nebula. The extracted region size is 1280 pixels = 320 arcsec.

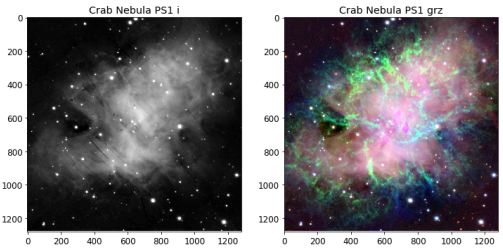
```
# Crab Nebula position
ra = 83.633210
dec = 22.014460
size = 1280

# grayscale image
gim = getgrayim(ra,dec,size=size,filter="i")
# color image
cim = getcolorim(ra,dec,size=size,filters="grz")

pylab.rcParams.update({'font.size':12})
pylab.figure(1,(12,6))
pylab.subplot(121)
pylab.imshow(gim,origin="upper")
pylab.title('Crab Nebula PS1 i')
pylab.subplot(122)
pylab.title('Crab Nebula PS1 grz')
pylab.imshow(cim,origin="upper")
```

```
Downloading https://ps1images.stsci.edu/cgi-bin/ps1filenames.py?ra=83.63321&dec=22.01446&size
=1280&format=fits&filters=i [Done]
Downloading https://ps1images.stsci.edu/cgi-bin/ps1filenames.py?ra=83.63321&dec=22.01446&size
=1280&format=fits&filters=grz [Done]

<matplotlib.image.AxesImage at 0x118c7e828>
```



Bulk Image Download Python Script

It is fairly simply to download PS1 cutout images in bulk. Here is a Python script that queries the interfaces described above using a list of RA, Dec positions and extracts 1 arcmin (240 pixel) FITS cutouts for each position and filter. This script can extract more than 4 cutouts per second if you have a sufficiently fast internet connection. It can be easily modified for special requirements (e.g., downloading different kinds of images, or download JPEG images instead of FITS images).

NOTE: If you modify this script to download images in multiple threads, please do not use more than 10 simultaneous threads for the download. The ps1images service is a shared resource, and too many requests from a single user can cause the system to be unresponsive for all users. If you attempt to download images at an excessive rate, eventually you will find your downloads blocked by the server.

ps1bulk.py

```
"""ps1bulk.py: Get PS1 stack FITS cutout images at a list of positions

NOTE: If you modify this script to download images in multiple threads,
please do not use more than 10 simultaneous threads for the download.
The ps1images service is a shared resource, and too many requests from
a single user can cause the system to be unresponsive for all users.
If you attempt to download images at an excessive rate, eventually you
will find your downloads blocked by the server.
"""

import numpy as np
from astropy.table import Table
import requests
import time
from io import StringIO

ps1filename = "https://ps1images.stsci.edu/cgi-bin/ps1filenames.py"
fitscut = "https://ps1images.stsci.edu/cgi-bin/fitscut.cgi"

def getimages(tra, tdec, size=240, filters="grizy", format="fits",
imagetypes="stack"):

    """Query ps1filenames.py service for multiple positions to get a list
of images
    This adds a url column to the table to retrieve the cutout.

    tra, tdec = list of positions in degrees
    size = image size in pixels (0.25 arcsec/pixel)
    filters = string with filters to include
    format = data format (options are "fits", "jpg", or "png")
    imagetypes = list of any of the acceptable image types.  Default is
stack;
        other common choices include warp (single-epoch images), stack.wt
(weight image),
```

```
        stack.mask, stack.exp (exposure time), stack.num (number of
exposures),
        warp.wt, and warp.mask.  This parameter can be a list of strings
or a
        comma-separated string.

    Returns an astropy table with the results
    """

    if format not in ("jpg","png","fits"):
        raise ValueError("format must be one of jpg, png, fits")
    # if imagetypes is a list, convert to a comma-separated string
    if not isinstance(imagetypes,str):
        imagetypes = ",".join(imagetypes)
    # put the positions in an in-memory file object
    cbuf = StringIO()
    cbuf.write('\n'.join(["{} {}".format(ra, dec) for (ra, dec) in zip(tra,
tdec)]))
    cbuf.seek(0)
    # use requests.post to pass in positions as a file
    r = requests.post(ps1filename, data=dict(filters=filters,
type=imagetypes),
        files=dict(file=cbuf))
    r.raise_for_status()
    tab = Table.read(r.text, format="ascii")

    urlbase = "{}?size={}&format={}".format(fitscut,size,format)
    tab["url"] = ["{}&ra={}&dec={}&red={}".format(urlbase,ra,dec,filename)
            for (filename,ra,dec) in zip(tab["filename"],tab["ra"],tab
["dec"])]
    return tab


if __name__ == "__main__":
    t0 = time.time()

    # create a test set of image positions
    tdec = np.append(np.arange(31)*3.95 - 29.1, 88.0)
    tra = np.append(np.arange(31)*12., 0.0)

    # get the PS1 info for those positions
    table = getimages(tra,tdec,filters="ri")
    print("{:.1f} s: got list of {} images for {} positions".format(time.
time()-t0,len(table),len(tra)))

    # if you are extracting images that are close together on the sky,
    # sorting by skycell and filter will improve the performance because
it takes
    # advantage of file system caching on the server
    table.sort(['projcell','subcell','filter'])

    # extract cutout for each position/filter combination
    for row in table:
        ra = row['ra']
        dec = row['dec']
        projcell = row['projcell']
        subcell = row['subcell']
        filter = row['filter']

        # create a name for the image -- could also include the projection
cell or other info
        fname = "t{:08.4f}{:+07.4f}.{}.fits".format(ra,dec,filter)

        url = row["url"]
        print("%11.6f %10.6f skycell.%4.4d.%3.3d %s" % (ra, dec, projcell,
subcell, fname))
        r = requests.get(url)
        open(fname,"wb").write(r.content)
    print("{:.1f} s: retrieved {} FITS files for {} positions".format(time.
time()-t0,len(table),len(tra)))
```

Plans for future additions & improvements

We may make a number of improvements and additions to this interface in the future. Suggestions for other additions are also welcome (send to archive@stsci.edu).

- Enable bulk downloads of files from the web interface (e.g., download all FITS cutouts on the web page as a tar file).
- Fill in blank pixels from neighboring skycells so that larger image cutouts are useful.
- Give the user the ability to select which filters are used to create the color image.
- Give the user the ability to adjust the contrast in the JPEG cutout images (a capability that already exists in the underlying fitscut service).
- Include more accurate footprints for the warp sky coverage in the database so that blank warp cutouts are rare.
- Speed up the interactive display and access to image cutouts either by recompressing the FITS skycell images to use blocks or by caching uncompressed versions of recently accessed images. Currently the images are compressed row by row, which is inefficient for access to randomly located blocks of pixels.
- Add documentation for all the different file types (stack.wt, etc.)
- Add a service that returns a model PSF at the search position.
- Migrate all images to the cloud to allow more and faster image access.