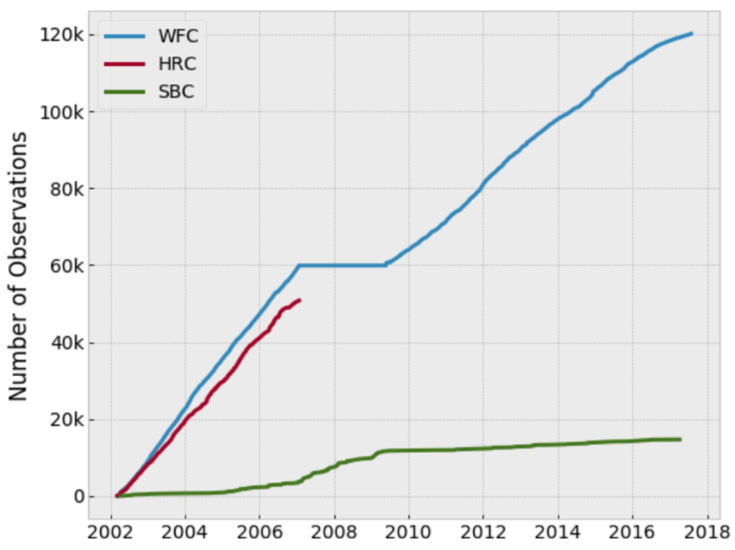
The Hubble Space Telescope Advanced Camera for Surveys Quicklook Application

Matthew Bourque[1], Sara Ogaz[1], Alex Viana[2], Meredith Durbin[3], Norman Grogin[1]

The Hubble Space Telescope (HST) Advanced Camera for Surveys (ACS) instrument has been acquiring thousands of images each year since its installation in 2002 (via Servicing Mission 3B) and subsequent restoration in 2009 (via Servicing Mission 4). The ACS Quicklook project (acsql) provides a means for users to view all on-orbit ACS image and header data through a browser interface, an API to build custom datasets through relational queries, and a platform on which to build automated instrument calibration and monitoring routines. This is accomplished through the various system components: (1) a ~40 TB **filesystem**, which stores all publicly-available ACS data on disk, (2) a MySQL **database**, which stores all publicly-available FITS header data for each ACS filetype and header extension, (3) a Python/Flask-based **web application**, which allows users to navigate and display any public ACS archival data, and (4) a Python **software package**, which serves as the filesystem, database, and website API. ACS Quicklook is intended to be extended to support the forthcoming James Webb Space Telescope mission.

Filesystem

The acsql filesystem contains all publicly-available on-orbit ACS data from each of its three detectors: The Wide Field Camera (WFC), the High Resolution Channel (HRC) and the Solar Blind Channel (SBC). The filesystem is organized based on the 9-character observation rootname. It also contains "Quicklook" JPEGs for each image, organized by the 5-digit Proposal ID.

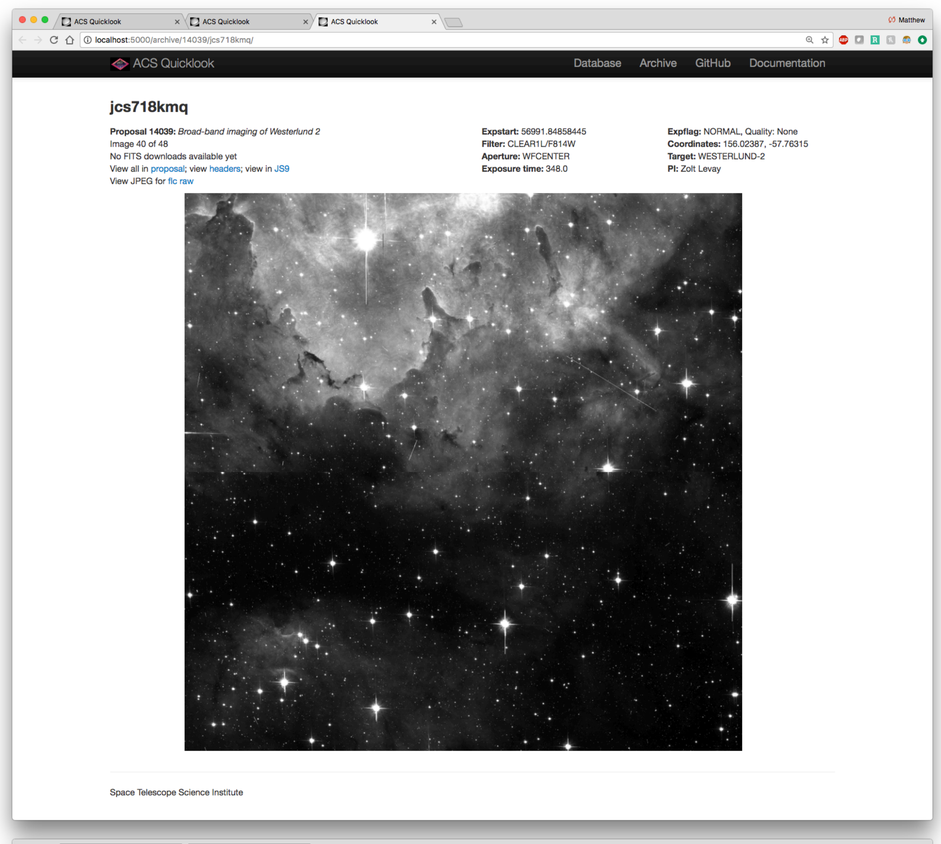



```
filesystem/
    jcp0/
        jcp001kwq/
            jcp001kwq_flc.fits
            jcp001kwq_flt.fits
            jcp001kwq_raw.fits
            jcp001kwq_spt.fits
            jcp001kwq_jif.fits
            jcp001kwq_jit.fits
        jcp001010/
            jcp001010_asn.fits
            jcp001010_drc.fits
            jcp001010_drz.fits
            jcp001010_jif.fits
            jcp001010_jit.fits
        jcp014tyq/
            ...
        jcp001kt1/
            ...
        ...
    jcp3/
        ...
    jcp7/
        ...
    ...
```

```
jpegs/
    12780/
        jbw901jtq_flc.jpg
        jbw901jtq_flt.jpg
        jbw901jtq_raw.jpg
        jbw901jxq_flc.jpg
        jbw901jxq_flt.jpg
        jbw901jxq_raw.jpg
        ...
    12781/
        jbx101f5q_flc.jpg
        jbx101f5q_flt.jpg
        jbx101f5q_raw.jpg
        jbx101f7q_flc.jpg
        jbx101f7q_flt.jpg
        jbx101f7q_raw.jpg
        ...
    12783/
        ...
    12787/
        ...
    ...
```

Web Application

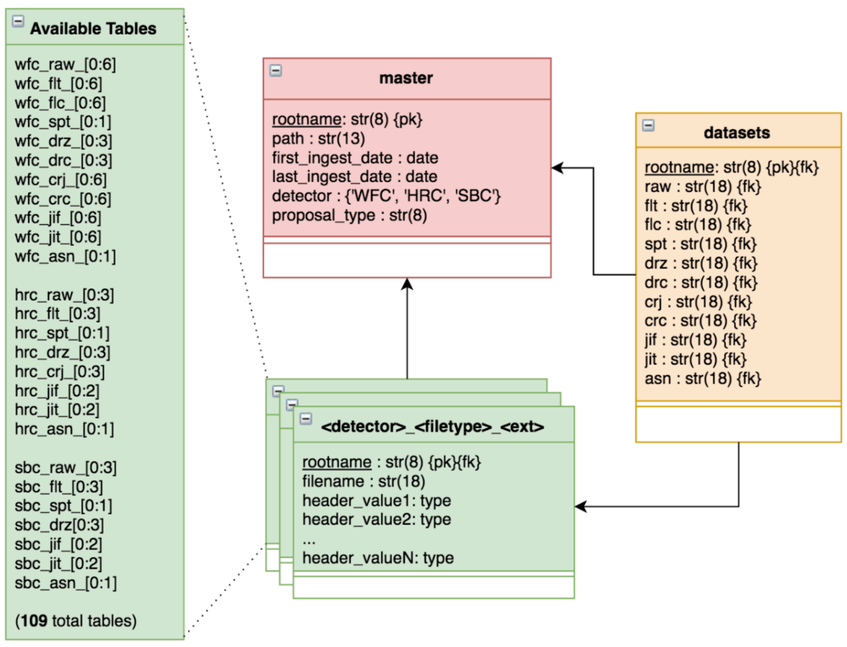
The acsql web application (right) allows users to interact with the filesystem and database through a browser interface. Users may view images for any archival ACS observations along with informative metadata, as well as perform queries on the acsql database.

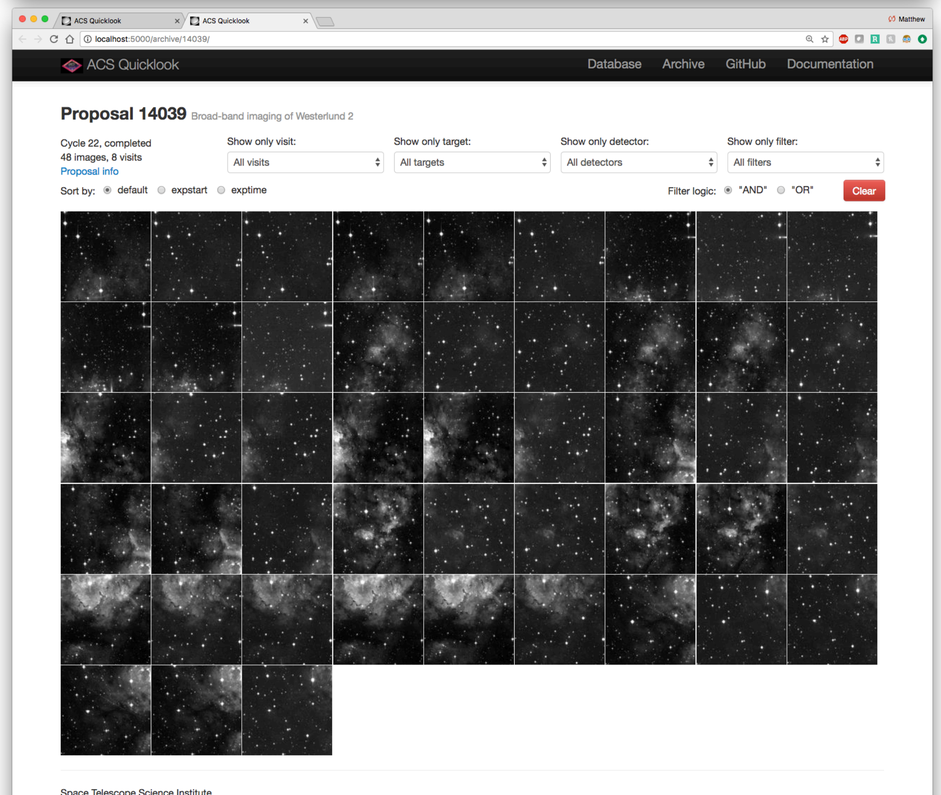


Database

The acsql database is a MySQL database constructed using Python, SQLAlchemy, and the relational schema above. The database stores observational metadata taken from the headers of each ACS FITS filetype and FITS extension for all publicly-available ACS observations. The database_interface Python module provides a programmatic means to perform relational queries on the database.

```
In [1]: from acsql.database.database_interface import session

In [2]: from acsql.database.database_interface import Master

In [3]: from acsql.database.database_interface import WFC_raw_0

In [4]: from acsql.database.database_interface import WFC_spt_1

In [5]: from acsql.database.database_interface import WFC_flt_1

In [6]: results = session.query(
   ...:     Master.rootname, WFC_raw_0.date_obs, WFC_raw_0.time_obs,
   ...:     WFC_spt_1.jwdetmp1, WFC_flt_1.meandark)\
   ...:         .join(WFC_raw_0, WFC_spt_1, WFC_flt_1)\
   ...:         .filter(Master.rootname.like('j6mf12%'))\
   ...:         .all()

In [7]: results
Out[7]:
[('j6mf12ev', '2003-01-26', '16:10:36', -76.6214, 1.1907731),
 ('j6mf12f0', '2003-01-26', '16:19:34', -77.3857, 1.1907731),
 ('j6mf12f4', '2003-01-26', '16:28:32', -76.6214, 1.1907731),
 ('j6mf12fa', '2003-01-26', '16:37:30', -76.6214, 1.1907731),
 ('j6mf12fg', '2003-01-26', '16:46:28', -75.8571, 1.1907731),
 ('j6mf12fs', '2003-01-26', '17:44:00', -76.6214, 1.278055),
 ('j6mf12fy', '2003-01-26', '17:53:26', -76.6214, 1.278055),
 ('j6mf12g2', '2003-01-26', '18:02:52', -75.0929, 1.278055),
 ('j6mf12g8', '2003-01-26', '18:12:18', -76.6214, 1.278055),
 ('j6mf12gd', '2003-01-26', '18:21:44', -76.6214, 1.278055),
 ('j6mf12gg', '2003-01-26', '19:19:03', -76.6214, 1.278055),
 ('j6mf12gj', '2003-01-26', '19:28:29', -76.6214, 1.278055),
 ('j6mf12gm', '2003-01-26', '19:37:55', -76.6214, 1.278055),
 ('j6mf12go', '2003-01-26', '19:47:21', -77.3857, 1.278055),
 ('j6mf12gr', '2003-01-26', '19:56:47', -76.6214, 1.278055)]
```



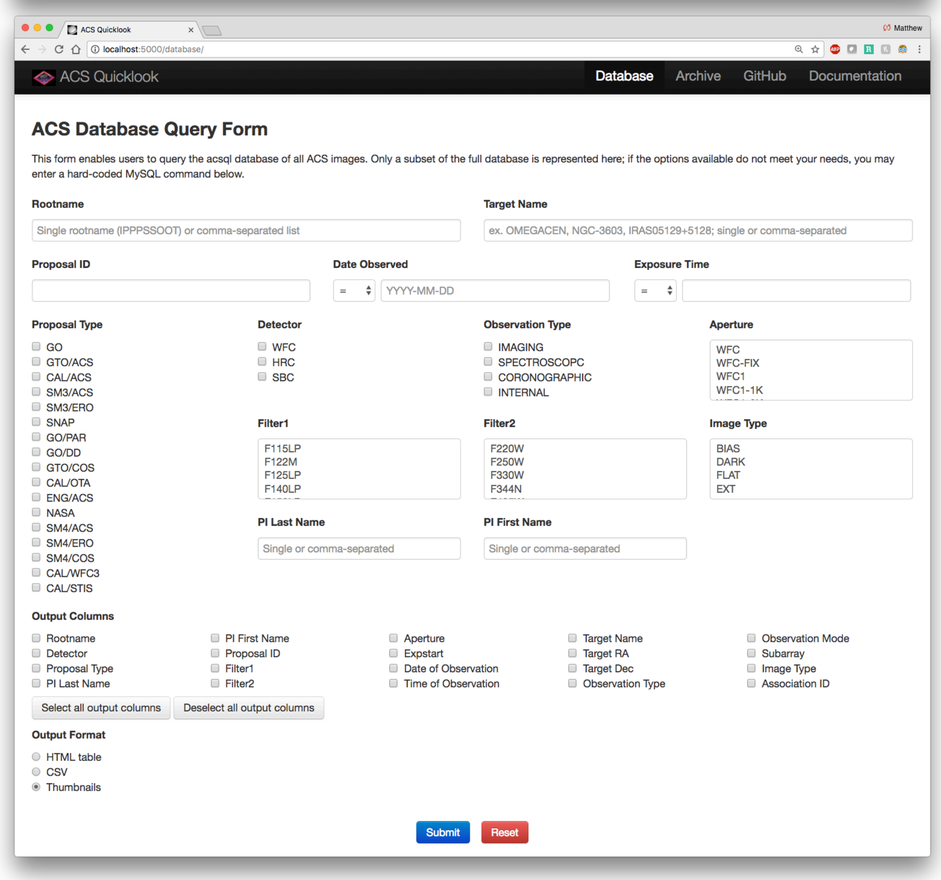


Software Package

All source code and supporting materials for the acsql application are open-source and available on GitHub. API documentation and installation instructions are available on ReadTheDocs.

github.com/spacetelescope/acsql

acsql.readthedocs.io



[1] Space Telescope Science Institute, Baltimore, Maryland, USA
[2] Terbium Labs, Baltimore, Maryland, USA
[3] University of Washington, Seattle, Washington, USA