

The Hubble Space Telescope Wide Field Camera 3 Quicklook Project

Matthew Bourque,¹ Varun Bajaj,¹ Ariel Bowers,¹ Michael Dulude,¹
Meredith Durbin,² Catherine Gosmeyer,¹ Heather Gunning,¹ Harish Khandrika,¹
Catherine Martlin,¹ Ben Sunnquist,¹ and Alex Viana³

¹*Space Telescope Science Institute, Baltimore, Maryland, USA*

²*University of Washington, Seattle, Washington, USA*

³*Terbium Labs, Baltimore, Maryland, USA*

Abstract. The Hubble Space Telescope’s Wide Field Camera 3 (WFC3) instrument has been acquiring ~50-100 images daily since its installation in 2009. The WFC3 Quicklook project provides a means for WFC3 instrument analysts to store, calibrate, monitor, and interact with these data through the various Quicklook systems: (1) a ~175 TB filesystem, which stores the entire WFC3 archive on disk, (2) a MySQL database, which stores image header data, (3) a Python-based automation platform, which currently executes 22 unique calibration and monitoring scripts, (4) a Python-based code library, which provides system functionality such as logging, downloading tools, database connection objects, and filesystem management, and (5) a Python/Flask-based web interface to the Quicklook system. The Quicklook project has enabled large-scale WFC3 analyses and calibrations, such as the monitoring of the health and stability of the WFC3 instrument, the measurement of ~20 million WFC3/UVIS PSFs, the creation of WFC3/IR persistence calibration products, and many others. The Quicklook system may be extended to support the forthcoming James Webb Space Telescope mission.

1. Introduction

The Hubble Space Telescope (HST) Wide Field Camera 3 (WFC3) ‘Quicklook’ project was implemented early in the WFC3 mission as a means for WFC3 team members to visually inspect incoming data on a daily basis, checking for both expected and unexpected detector features and characteristics. Since then, the Quicklook project has steadily grown to become a robust system that allows for WFC3 team members to analyze, calibrate, and monitor instrument data in a streamlined fashion. This is accomplished by the implementation of a filesystem, database, automation platform, a code library, and a website. These subsystems all work together in a ‘workflow’ that brings WFC3 data from the Mikulski Archive for Space Telescopes (MAST) to the fingertips of WFC3 instrument analysts and scientists via the Quicklook website (see Figure 1). In this sense, the Quicklook website is a ‘one stop shop’ for all WFC3 data, including image metadata, instrument health and stability monitors, and the images themselves.

2. Filesystem and Database

The Quicklook filesystem stores both calibrated and uncalibrated files for all on-orbit WFC3 data (2009-present), as well as providing storage of files related to WFC3 instrument calibration work (i.e. reference file creation, instrument monitoring, regression testing, etc.). The files are

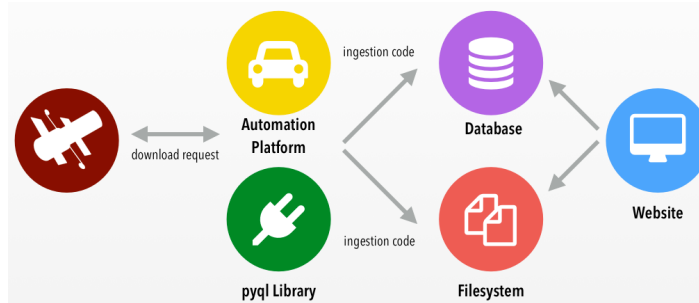


Figure 1. Overview of the Quicklook system.

stored in a ~ 175 TB filesystem spanning over several centrally-accessible hard disks, allowing team members to quickly access the data without needing to wait for MAST archival downloads. Figure 2 shows the disk usage of the filesystem over time since tracking began in early 2013. The data are stored in an organized directory structure (see Figure 2) to allow for easy iteration over numerous files and across various datasets.

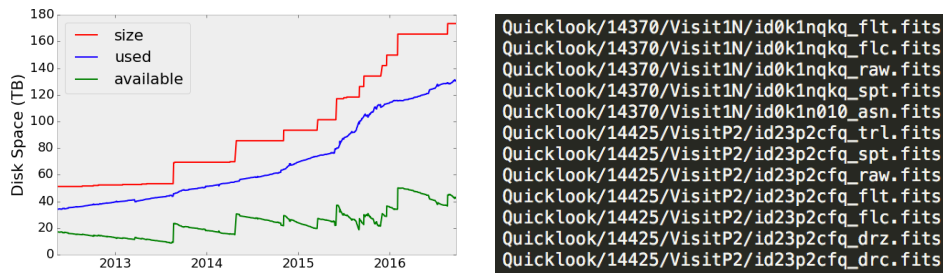


Figure 2. *Left:* The Quicklook filesystem disk usage over time. *Right:* The Quicklook filesystem directory structure, organized by proposal number and visit, showing that it is easy to iterate over various datasets.

The Quicklook database is a MySQL database which stores (1) all on-orbit WFC3 *_flt.fits image header data (which describes the particular observing mode, e.g. exposure time, target information, proposal information, filter(s) used, etc.), (2) all on-orbit WFC3 *_spt.fits header data (which describes the HST telemetry at observing time, e.g. pointing coordinates, overhead times, telescope position, etc.), and (3) Quicklook filesystem metadata (rootnames, ingestion date, filesystem location, disk usage statistics, etc.). With this information, it is convenient to use the Quicklook database to build custom datasets, as shown in Figure 3.

3. Automation Platform and pyql Library

In addition to storing image and header data, the Quicklook system employs an automation platform and a Python library (called ‘pyql’) to automatically execute Quicklook system code as well as WFC3 instrument calibration and monitoring scripts on a daily or weekly basis. Currently there are 22 different automated scripts (as seen in Figure 4) across 85 Python modules and 25,000+ lines of code. The pyql library contains several modules to help support the automation platform, most notably modules that contain archive download tools, filesystem operations, logging, and database connection objects (see Figure 4). These two subsystems work in unison to execute code from a virtual machine via a crontab, and often use third-party libraries such as astropy (Astropy Collaboration et al. (2013)), matplotlib (Hunter (2007)), numpy (Ste-

```

SELECT master.rootname,
       uvis_flt_0.targname,
       uvis_flt_0.exptime,
       uvis_flt_0.ra_targ,
       uvis_flt_0.dec_targ,
       uvis_flt_0.filter
FROM master
JOIN uvis_flt_0
WHERE uvis_flt_0.targname LIKE 'NGC%'
AND uvis_flt_0.date_obs > '2015-01-01'
AND uvis_flt_0.date_obs < '2015-01-30';

```

rootname	targname	exptime	expstart	filter
icnk27m2q	NGC4485	354	57039.07031250	F547M
icnk27m4q	NGC4485	200	57039.07421875	F547M
icnk27l1q	NGC4485	515	57039.04687500	F657N
icnk27l1q	NGC4485	515	57039.05468750	F657N
icnk27l2q	NGC4485	515	57039.06250000	F657N
icnk20awq	NGC5474	354	57041.05859375	F547M
icnk20ayq	NGC5474	200	57041.06250000	F547M
icnk20a8q	NGC5474	515	57041.03515625	F657N
icnk20adq	NGC5474	515	57041.04296875	F657N
icnk20afq	NGC5474	515	57041.05078125	F657N
icnk22urq	NGC7793-2	515	57030.99218750	F657N
icnk22usq	NGC7793-2	515	57031.00000000	F657N
icnk22vbq	NGC7793-2	515	57031.05468750	F657N
icnk22vfq	NGC7793-2	350	57031.06250000	F547M
icnk22v1q	NGC7793-2	200	57031.06640625	F547M

Figure 3. *Left:* An example query of the Quicklook database. *Right:* The results of the query shown on left.

fan van der Walt & Varoquaux (2011)), scipy (Jones et al. (01)), and sqlalchemy (Bayer (06)).

```

cal_ir_make_bad_actor.py
cal_ir_make_bias_plots.py
cal_ir_make_blob_plots.py
cal_ir_make_persistence.py
cal_ir_make_snowballs.py
cal_uvis_make_bowtie.py
cal_uvis_make_ctecorr_darks.py
cal_uvis_make_darks.py
cal_uvis_make_aper_cte.py
cal_uvis_make_gain_plots.py
cal_uvis_setup_contam.py
ql_wfc3_check_duplicate_proposal_folders.py
ql_wfc3_check_filesystem.py
ql_wfc3_check_fsyzqlldb.py
ql_wfc3_check_mast2qlldb.py
ql_wfc3_check_qldb2fsys.py
ql_wfc3_get_missing_files.py
ql_wfc3_get_new_datasets.py
ql_wfc3_get_proposal_status.py
ql_wfc3_make_cron_summary.py
ql_wfc3_make_log_plots.py
ql_wfc3_run_ingest.py

```

```

from pyql.download import download_tools
from pyql.file_system import set_permissions
from pyql.logging.logging_functions import configure_logging
from pyql.logging.logging_functions import log_fail
from pyql.logging.logging_functions import log_info
from pyql.database.ql_database_interface import session
from pyql.database.ql_database_interface import Master
from pyql.database.ql_database_interface import UVIS_flt_0

```

Figure 4. *Left:* A list of the current automated scripts as part of the Quicklook automation platform. *Right:* Example imports from the pyql Library, which provides functionality such as downloading tools, filesystem operations, logging, and database connections.

4. Website

The Quicklook website is a Python/Flask-based website, used as a user interface to the Quicklook system. It allows users to interact with the Quicklook filesystem and database, as well as to easily view and analyze WFC3 images and their metadata, as shown in Figure 5. Users can also view log files of the automated script executions, flag detector anomalies and features, and view instrument monitoring plots. Due to proprietary HST data, the Quicklook website is currently only available to WFC3 team members.

Acknowledgments. We would like to thank the Wide Field Camera 3 team at STScI for their continued support of this project, particularly John MacKenty, Elena Sabbi, and Sylvia Baggett. We would also like to thank the members of the Operations and Engineering Division (OED) and the Information Technology Services Division (ITSD) at STScI for their assistance on archival and computer support issues.

References

- Astropy Collaboration, et al. 2013, A&A, 558, A33. 1307. 6212
 Bayer, M. 2006–, SQLAlchemy: The Database Toolkit for Python. [Online; accessed 2016-10-12], URL <http://www.sqlalchemy.org/>
 Hunter, J. D. 2007, Computing in Science and Engineering, 9, 90. URL <http://scitation.aip.org/content/aip/journal/cise/9/3/10.1109/MCSE.2007.55>

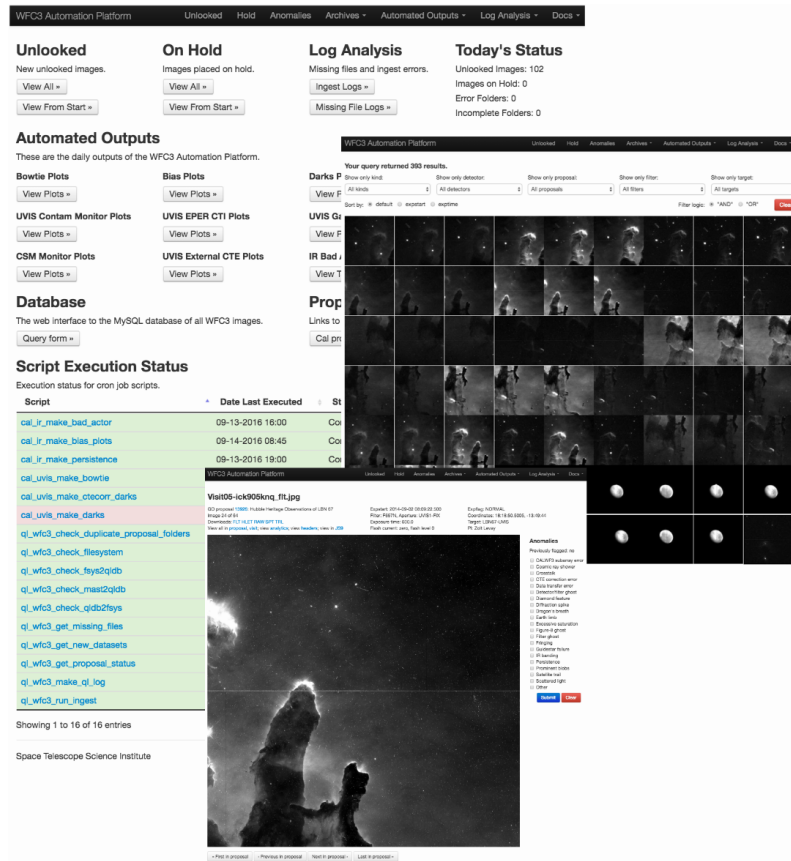


Figure 5. The Quicklook homepage (*top*), a Quicklook webpage showing results of a database query (*middle*), and a Quicklook webpage for viewing a single image (*bottom*).

- Jones, E., Oliphant, T., Peterson, P., et al. 2001–, SciPy: Open source scientific tools for Python. [Online; accessed 2016-10-12], URL <http://www.scipy.org/>
- Stefan van der Walt, S. C. C., & Varoquaux, G. 2011, Computing in Science and Engineering, 13, 22. URL <http://scitation.aip.org/content/aip/journal/cise/13/2/10.1109/MCSE.2011.37>