Creating the Prototype JWST Exposure Time Calculators (ETCs)

Megan Sosey[1], Christopher Hanley[1], Vicki Laidler[2], Kevin Lindsay[1], Mark Sienkiewicz[1], Perry Greenfield[1]

[1] Space Telescope Science Institute, 3700 San Martin Drive, Baltimore MD, 21218 USA

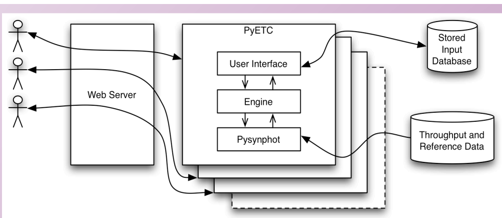
[2] Computer Sciences Corporation at STScI, 3700 San Martin Drive, Baltimore MD, 21218 USA

SPACE TELESCOPE SCIENCE INSTITUTE

Operated for NASA by AURA

ABSTRACT

Exposure time calculators (ETCs) are tools that astronomers use to explore the observing scenarios for their scientific data acquisition. ETCs are heavily used during the periods of observation proposal planning and submission. They are used not only to get estimates for exposure times, but also to check on the health and safety risks those observations might pose to the telescope. The James Webb Space Telescope (JWST) is still in development and does not yet have any orbital data. However, having access to such a tool can greatly aid scientists as they discover the new capabilities that JWST will provide. Recently, the Hubble Space Telescope (HST) ETC was redesigned and we leveraged this experience and the knowledge gained to implement a prototype ETC for a limited set of JWST observing modes. The JWST prototype ETCs were released to support the "Frontier Science Opportunities with JWST" meeting which discussed the high-risk, high-reward, science programs that Webb will achieve. Since HST primarly operates in the visible wavelengths of light and JWST in the infrared, we needed to create new functionality to meet the challenges of this new regime. This included developing new noise models to deal with the IR background environment. In addition, JWST had several extremely complex observation modes requiring a great deal of development and scientific scrutiny to implement correctly. In this poster we discuss the development of the JWST ETC and the tools we were able to employ that greatly simplified both the creation and verification of the final results.



Redesign of the HST ETC

One of the goals of the ETC is to help reduce the work of the user at every place in the proposal process. The Hubble ETC was originally written in a combination of Java, XML and HTML. In the years since it was first deployed, the code increased in complexity until it was no longer feasible to maintain its current state of operations or make updates to its functionality without unreasonably long timescales. Creating a general tool that shares commonality among the different instruments is complicated, it needs to provide accurate calculations, but also be reliable, portable, and maintainable. The HST ETC currently supports all of the active HST instruments and the redesign is flexible enough to support JWST.

James Webb Space Telescope **Exposure Time Calculator** NIRCAM NEAR INFRARED CAMERA

****JWST PROTOTYPE ETC****

ETC Version P1.2

- pyetc 0.96.jwstdev
- pysynphot 0.9.jwstdev
- cdbs P1.2

ETC Help

- User's Guide (PDF)
- User's Guide (HTML)
- Release Notes
- News and Known Issues

NIRCAM ETCs

- Imaging

MIRI ETCs

- Imaging
- MRS Spectroscopy

NIRSPEC ETCs

- MSA

TFI ETCs

- Imaging

Previous results

- Previous calculation results

ETC Request ID: NIRCAM.im.246833

The current version of the ETC does not calculate detector saturation.

Exposure time (seconds) = 900.0000
gives: SNR = 756.4485

Detailed Information	Count rate (e-/s)	Total counts (e-)	Associated noise (e-)
Extraction area (circle with radius 0.08 arcsec)			
Source	641.632	577,469.09	759.91
Background	3.967	3,570.51	59.75
Sky	3.849	3,464.48	58.86
Dark Current	0.118	106.03	10.30
Read out			41.62
Total in selected region	645.600	581,039.60	763.40

Count rate
Source (infinitely large region) 1,144.883
Sky Background (arcsec^-2) 191.455

Extraction region
circle with radius 0.08 arcsec
Area (pixels) 19.63
Fraction of flux 0.56
Effective Wavelength 1.14 μm

Instrument name: NIRCAM
Mode: imaging
Bandpass: [F115W] Wide

Target: [point source]
Spectrum: Flat in units of F_ν
Extinction E(B-V): None
Normalization: 20.0 ABMAG in JOHNSON,K
Redshift: None
Emission Lines: None

Selected background:
Sky Background:
Thermal Background: Average
Zodiacal Light: Average

These results were computed using: ETC version P1.2

If you have any questions regarding this calculator, please contact the STScI Help Desk (help@stsci.edu).
Remember to include your ETC ID as well as the text "JWST ETC" and the instrument in the subject line

An example result page, returned from the NIRCAM imaging ETC

More information on the HST and JWST Exposure Time Calculators can be found on the Space Telescope
The HST ETC can be accessed at this location: http://etc.stsci.edu
The JWST ETC can be accessed at: http://jwstetc.stsci.edu

We would like to thank the numerous people who were involved in making this project a success, including the members of the JWST instrument teams, especially Elizabeth Barker, Christine Chen, Jeff Valenti and Andre Martel, support from members of the Science Software Branch at STScI, and all the testers who helped ensure the quality of the final product.

Please also visit the corollary poster in this session by R.I. Diaz, P033
"The STScI Exposure Time Calculators"

Development of the JWST ETC

Given a user-specified exposure time (in seconds), the prototype JWST ETC calculates the S/N and the count rates due to the combined source and background components for point-sources. Many of the internal calculations depend heavily on the Pysynphot synthetic photometry package, which is part of the stsci-python and is developed and maintained by the Science Software Branch at STScI. The program uses the filter transmission, throughputs of the optics, and the efficiency of the detector, to calculate the expected countrate for the specified configuration. The ETC code then uses this information to find the S/N ratio and integrated counts from the source, the sky background, and the detector background for the specified exposure time and extraction aperture. In order to implement these calculations a new suite of JWST specific files were added to the calibration reference system that pysynphot queries. Also necessary was detailed information on the operating characteristics of the JWST instruments themselves, including quantum efficiency, enclosed energy properties for point spread functions (psfs) and dark current estimates, to name a few. The design of the new python ETC made it easy to collect and store this information for each instrument in human readable text files which the instrument scientists could verify without having any prior knowledge of how the ETC software operated or more than the most basic python constructs. These files can then be stored in a repository which contains a clear chain of history, something the previous version of the ETC lacked.

Since JWST is able to collect data far out into the infrared, we also needed updated and extended stellar and background models which could be used to estimate flux in this regime. The instrument teams put together the most current derived models from the available space and ground based observations. However, because the templates are constructed using a combination of real data and models, artificial discontinuities at certain wavelengths may be noticeable. Since all of the stellar and galactic source templates cover a finite wavelength range, applying high redshifts to the templates can result in pushing the them outside the range of the bluest filter bandpasses, which will produce zero source counts in the results. For every calculation performed with the ETC, an ETC Request ID is provided at the top of the output page. This ID is a key into our database where each ETC calculation is stored. The information can be retrieved later by the user, or by a programmer to debug any problems which might have occurred and has become useful for both testing functionality and creating a suite of regression tests that can be run nightly.

```
{
'date'           : '2011-11-01 16:07:19',
'engine_input'   :
    {
    'detector'          : 'short',
    'extraction'        :
        {
        'detector'      : 'short',
        'gyromode'      : 'normal',
        'instrument'    : 'nircam',
        'science_mode'  : 'imaging',
        'sdiameter'     : 0.0,
        'shape'         : 'circle',
        'size'          : 0.08,
        'stype'         : 'point',
        },
    'instrument'        : 'nircam',
    'obsmode'           : 'nircam,im,f115w',
    'otesunshield_str'  : 'Average',
    'read_mode'         : 'slow',
    'science_mode'      : 'imaging',
    'simmode'           : 'SNR',
    'sky_expr'          : 'spec($PYSYN_CDBS/etc/background/jwst_bkgd_telepss_final.fits) + spec($PYSYN_CDBS/etc/background/zodi_jwst_model_medium_001.fits)',
    'source_expr'       : 'rn(unit(1.,fnu),band(johnson,k),20.000000,abmag)',
    'time'              : 900.0,
    'zodi_str'          : 'Average',
    },
```

An example of the dictionary the ETC engine accepts as input for a full calculation

Challenges of Modeling IR Astronomy

Like most infrared detectors, those on JWST are read-out non-destructively using correlated double sampling. A variety of sampling patterns will be made available to observers so that they can choose the most desirable combination of readout noise, dynamic range and data volumes. However, the full options are not yet available in this prototype ETC, which instead uses a simplified approximation of the read noise which should provide adequate estimates for most purposes. For most modes, exposure times longer than ~1000 seconds are divided into the minimum number of equal-length integrations (known as ramps) which add to the total requested time. The MIRI MRS is an Integral Field spectrograph which obtains simultaneous spectral and spatial data on a compact section of the sky. This instrument has some special requirements and long exposures are split into integrations shorter than 960 seconds (for channels 1 and 2) or 480 seconds (for channels 3 and 4). The division of integrations affects the read noise calculation and was done to help simulate the effects of cosmic rays and dithers. Cosmic ray effects are more of an issue because of the length of the observations and the sensitivity of the detectors. These may be mitigated by making a simplifying assumption for the ETC calculation and then applying statistical corrections.

Total photon counts from the specified type of source is computed by multiplying the pysynphot derived count rate (in photons per second) by the user-specified exposure time. Some terms may only apply to certain instruments (for example the thermal background, which is scattered emission from the telescope structure seen by the instrument detectors in longer IR wavelengths). The following formulation has been developed to approximate the noise realized in reading out a full-frame image with NIRCam, and applied for simplicity to the rest of the JWST instruments with a constant based on exposure time:

$$SNR = \frac{\Phi_* * t}{[1.1 * (\Phi_* + \Phi_{bkg}\Omega) * t + (\sigma^{RON}(t))^2 * \Omega]^{1/2}}$$

Where RON=readout noise, Φ_* is the stellar flux, and Φ_{BKG} is the total background, including sky, thermal, dark current, etc. per pixel. The factor of 1.1 accounts for the loss due to the correlated sampling of the Poisson noise, and Ω is the number of pixels contributing to the noise. In an actual astronomical observation, the background must be estimated from some finite region of pixels. For the ETC, we assume that the number of background pixels is much greater than the number of source pixels and ignore the uncertainty in the estimate. For imaging modes, this is a reasonable assumption. However, for modes such as the NIRSPEC MSA, which does spectroscopy using an array of shutters, it may be the case that only one or two shutters adjacent to the targets shutter are useful for the background measurement. In this case, the combined read noise and background noise could be larger by almost 20%. Emission from the telescope and sunshield are included in the reported sky values, along with the emission from the zodiacal light. The contribution of zodiacal light depends on the position of a target relative to the Sun and the ecliptic and thus has an annual modulation. For most JWST observing modes, this zodiacal light dominates the other background noise sources. There is no allowance for cosmological dimming in this ETC. The wavelengths are multiplied by (1 + z), but the fluxes depend on how the spectrum is normalized, which is applied after the redshift.

While calculating the the achievable signal to noise ratio is not overly complex, the time calculation for JWST is not analytically reversible from the SNR calculation and needs to be built on iteration. The exact method for this has not yet been decided as the exposure time is also directly related to the sampling method used and the dither pattern selected, meaning there are finite and specific observation times which are allowed. In addition, saturation may come into play when science in one band may be limited by saturation in another, something the user also needs to consider when constructing their observations, and should be supported in a future implementation.

How Python and its associated toolsets aided our effort

The redesign of the previous HST ETC into Python has enabled many improvements in how the code is managed and utilized by users. Python is well supported in the scientific community, allows for rapid development and has native support for web development. Django was used as the high-level web framework because it has a useful template and caching system and the object-relational mapper allows the definition of data models entirely in python. Python also makes it easy to translate scientific code, often written by scientists in languages such as IDL, the IRAF CL and Fortran into ETC usable code which can be understood by all parties involved. This makes it easier to confer directly with scientists while debugging, allows for use of the PDB to step through engine calculation and display real time plots, using matplotlib, and current variable values associated with the calculation. Documentation was made easy by using Sphinx to autogenerate HTML, LaTeX and subsequently PDF sources, available in realtime or offline. Finally, the current Python implemented ETC design is intelligible and flexible, the data files are human readable, and updates and improvements take significantly less time to implement.