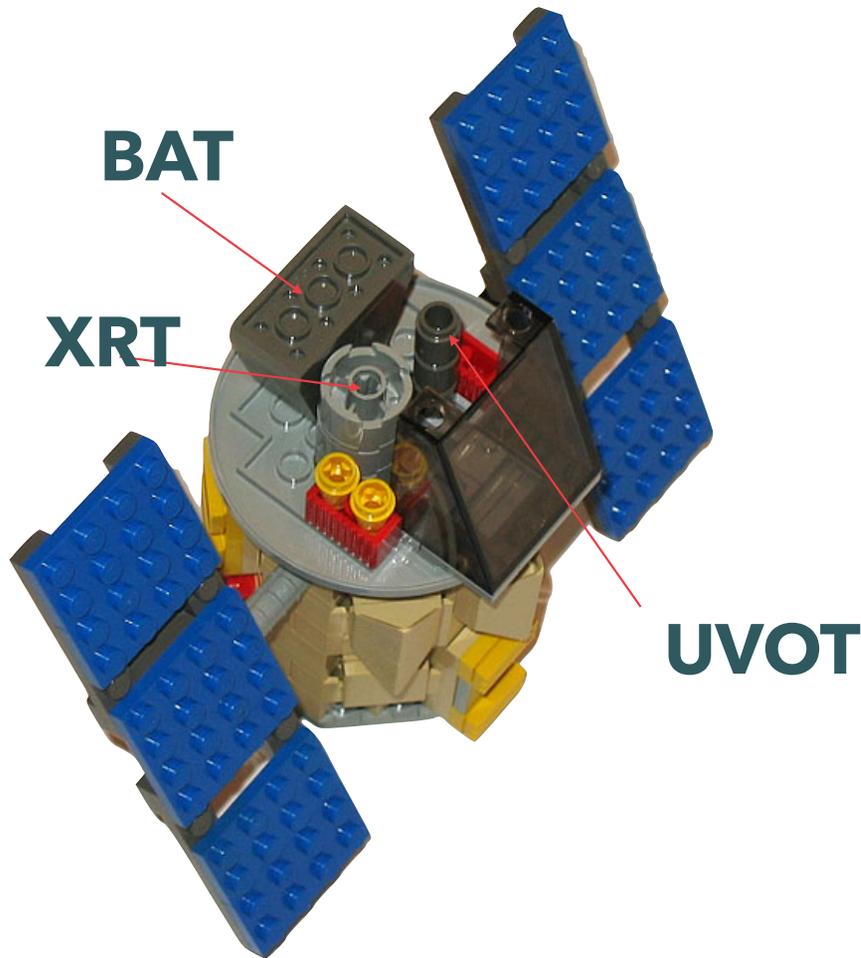




Neil Gehrels Swift Observatory

**Jamie A. Kennea (Penn State)
Science Operations Team Lead, on behalf of the Swift Science
Operations Team**

NASA's Neil Gehrels Swift Observatory



- **Burst Alert Telescope (BAT)**

- "Hard X-ray" 15-150 keV
- 2 sr field of view (1/6th of sky)
- CdZnTe detectors
- Detects ~100 GRBs per year

- **X-Ray Telescope (XRT)**

- "Soft X-ray" 0.3-10 keV
- 23.8 arcminute diameter FOV (~0.12 sq degree)
- few arcsecond (as good as 1.8") positions
- CCD spectroscopy

- **UV/Optical Telescope (UVOT)**

- 170 - 650 nm
- 17 arcminute width square FOV (~0.08 sq degree)
- Sub-arcsecond positions
- Grism spectroscopy
- 6 UV/optical broad-band filters
- 22nd mag sensitivity (filtered)

What Makes Swift Unique?

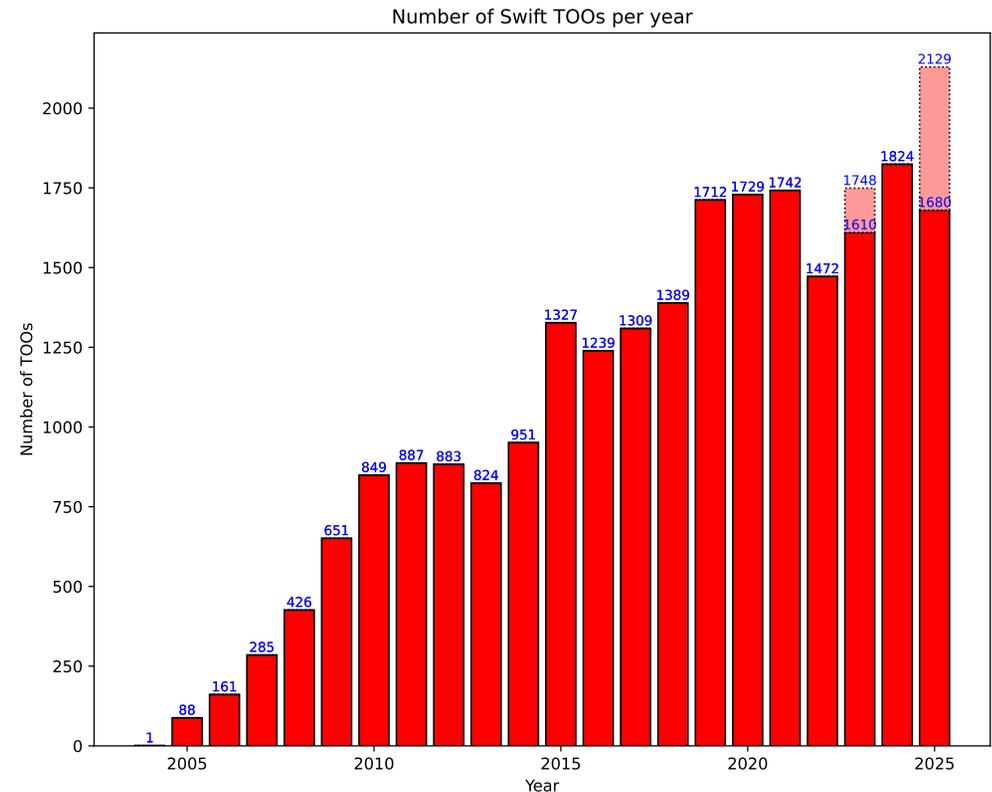
- **Multi-wavelength observations** - Space unique Hard X-ray/ X-ray / UV all in one package, simultaneously.
- **Transient discovery** - BAT is a hard X-ray transient all-sky (in a day) monitor. Triggers on GRBs, SGRs, LMXBs, SFXTs. BAT transient monitor tracks brightness of hundreds of X-ray sources, and discovers new transients.
- **Rapid slewing** - gets you to a GRB fast. Also allows for **very high efficiency** of operations (~72%) Allows **time domain astrophysics** due to ability to perform high cadence high sensitivity monitoring.
- **Open Target of Opportunity (TOO) program with open data program** - Our TOO program is extremely open, with low rejection rates, and our data is made public ASAP.

Swift receives a lot of TOO requests

- **2024: 1824 TOOs in 1 year.**
- **2025: ~2130 predicted** (extrapolation, but EP + SVOM likely driving jump)!

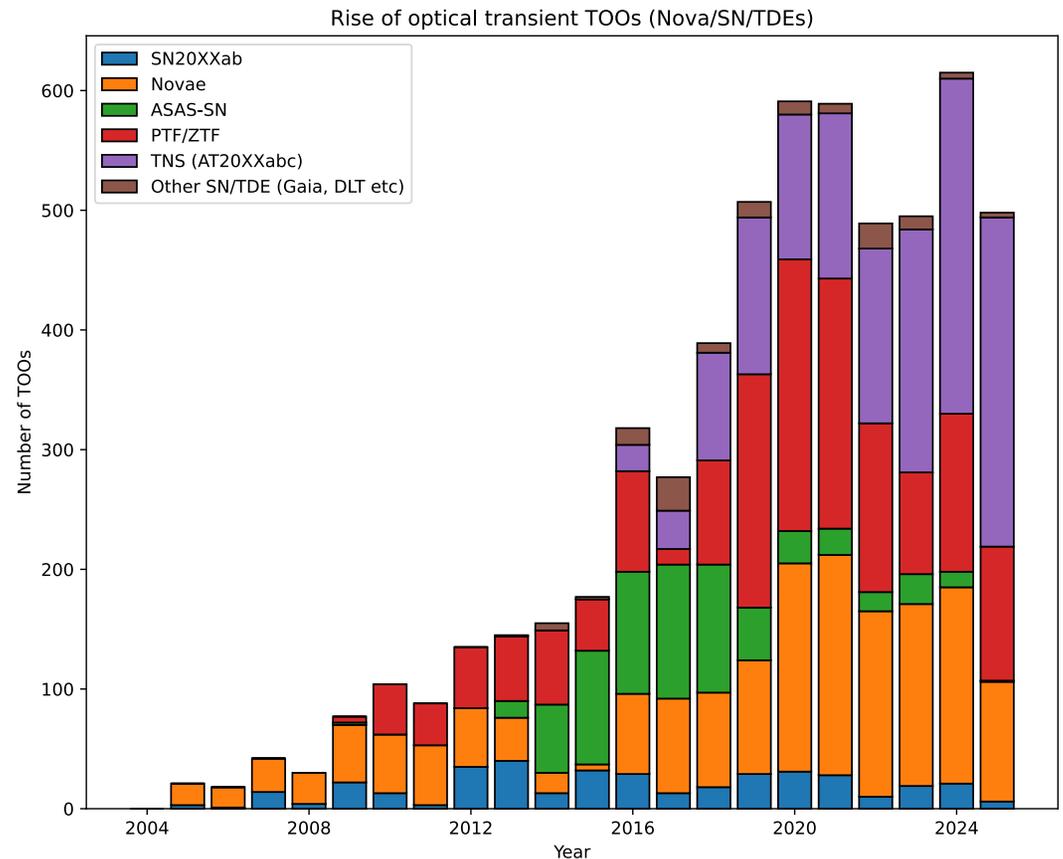
Why so high?

- TOO program open to anyone in the world.
- TOO acceptance rate has been ~99% for entire mission.
- More and more we are getting **automated TOOs through our API.**
- Swift is hugely capable for a broad section of the community.



Optical Surveys account for ~1/3 of all TOO

- Discovery of Supernova, Tidal disruption Events and Novae have driven about ~1/3 of all TOO requests to Swift.
- We are about to enter the era of discovery from the **Vera Rubin Observatory!**
- Swift is ready to perform space-unique X-ray and UV observations of transients discovered by Rubin!



How to submit a Swift TOO

Using the TOO Web Page

The screenshot shows the Penn State Mission Operations Center for Swift website. The page title is "Swift Target-of-Opportunity Request". Below the header, there is a navigation menu with links for Home, Target of Opportunity, Observations, BAT, XRT, and UVOT. The main content area contains instructions for submitting a TOO form, including a note that validated forms will be marked with a green checkmark and missing entries with a red cross. Below the instructions is a form titled "TOO Form" with two sections: "Object Information" and "Type or Classification". The "Object Information" section includes a text input for "Object Name", a "Resolve coordinates" button, and input fields for "Right Ascension (J2000)", "Declination (J2000)", and "Position Error (90% confidence - arcminutes)".

<https://www.swift.psu.edu/toop>
(This is just an API client)

Using the Swift TOO API

The screenshot shows a Jupyter Notebook titled "Swift TOO submission Example". The notebook contains a Python script that demonstrates how to use the Swift TOO API. The script starts by importing the 'Resolve' class from the 'swift_too' module. It then initializes a 'Resolve' object with a username and shared secret. The script sets up a 'Swift_TOO' request with a source name 'SMC X-3' and a debug mode. It then calls the 'Resolve' object's 'resolve' method to get the RA and Dec coordinates. Finally, it prints the RA and Dec coordinates in decimal degrees.

```
swift_too module
Swift_TOO example - Swift Target of Opportunity Submission Example
API version = 1.2, swifttools version = 2.3
Author: Jamie A. Kennea (Penn State)
The Swift_TOO class is used to submit Target of Opportunity Requests for the Swift mission. Before this required manual filling out of a web based form, but given the requirements to trigger Swift using algorithmically determined criteria, this can now be automated through the Swift TOO API, and this class. Here we give a simple example of how to submit a TOO request with this class. Note that we will enable debug mode here, so that the submission will actually complete, but importantly it will not submit an actual Swift TOO.

In [1]: from swift_too import TOO, Resolve

We start by initializing the class and giving our username and shared secret. These can be set up on the Swift TOO website. After you log in, you will find your shared secret, and can modify it as necessary under the Update Account Info link. Note that unlike other TOO API classes, you cannot use anonymous login to submit a TOO request.

In [2]: username = "myuser"
shared_secret = "mysecret"

OK let's set up the Swift_TOO request (in this case we'll use the swifttools 2.3 shorthand, TOO). Also we're setting debug mode here. Note that I'm not passing my shared_secret here, because my computer supports keyring. This records your shared_secret the first time that you use it, so it's not necessary to include it in later requests.

In [3]: too = TOO()
too.username = username
if too.shared_secret = shared_secret
too.debug = True

OK, so what are we going to observe, let's start with a name, oh say, SMC X-3, that's a cool object. However, darn, I can't remember the coordinates, but thankfully we have a class called Swift_Resolve (we'll use Resolve for short) for looking these up.

In [4]: too.source_name = "SMC X-3"
res = Resolve(name=too.source_name)
print("RA/Dec (J2000) = {res.ra:.4f}, {res.dec:.4f}")
print("SkyCoord = {res.skycoord}")

RA/Dec (J2000) = 13.4234, -72.4345
SkyCoord = <SkyCoord (FK5: equinox=J2000.000): (ra, dec) in deg
(13.42343917, -72.43454833)>

Swift_Resolve reports back ra and dec, but it also reports back as a SkyCoord using the skycoord property if you have astropy installed. Swift_TOO can take a SkyCoord directly.

In [5]: too.skycoord = res.skycoord

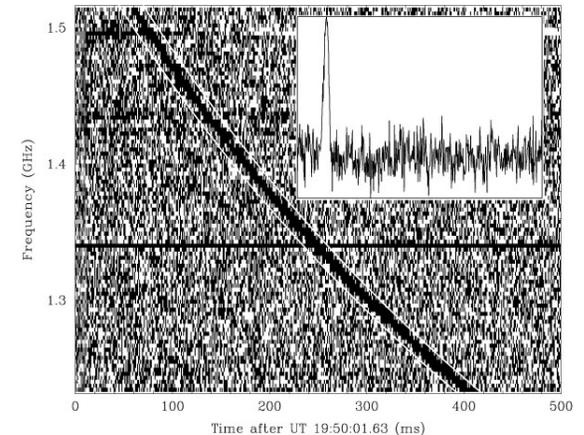
If you use a SkyCoord, it means you can also use more other coordinate systems or formats, rather than just J2000 decimal degrees. Internally all Swift TOOs are stored as decimal RA/Dec in J2000, as this is what Swift uses. Let's check what the values are.

In [6]: print("RA/Dec (J2000) = {too.ra:.4f}, {too.dec:.4f}")
```

pip install swifttools
https://www.swift.psu.edu/too_api

Using Swift search for FRBs

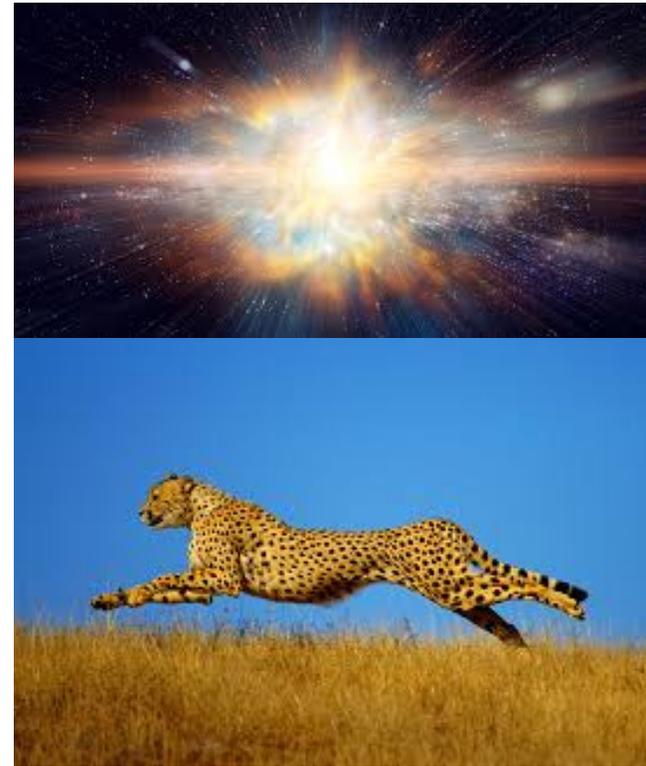
- We trigger off alerts from CHIME and other FRB detectors.
- If error region less than XRT FOV ($\sim 12'$ radius) we try to observe with XRT/UVOT as rapidly as possible using the "AutoTOO system".
- Please read <https://arxiv.org/abs/2006.04550> for a demonstration of such follow-up (no detection sadly). FRB observed with XRT/UVOT at T_0+32 mins.
- Two other FRBs have been followed-up in this way:
 - FRB 20211122A at **T_0+39** minutes (ATEL #15055).
 - FRB 20211211A at **T_0+21** minutes (ATEL #15114).
- In 2023, enabled by advances in Swift operations, we reduced this latency to **$T_0+3.9$ minutes** (ATEL #16233).
- In 2025, thanks to further refinement of our AutoTOO system, we got on target for a CHIME FRB at **T_0+82 seconds (sadly the wrong coords, thanks to a CHIME glitch!)**



Urgency 0 TOOs: Rapid response for all

In 2023 for O4 we enabled “**Continuous Commanding**” reducing latencies for TOOs from hours (typical previous best 14 mins) to seconds, opening up whole new science cases and responsiveness to TDAMM science:

- Possibility of “early warning” slewing to increase odds of coincident BAT detection of NS-NS merger (see paper “*Swiftly Chasing Gravitational Waves across the Sky in Real Time*” by Tohuvavohu et al, 2024 *ApJL* **975** L19).
- Enables us to us to commence GW tiling more quickly, and follow-up counterparts from other telescopes quicker.
- Places triggers from other missions on a level playing field to triggers from BAT.



Urgency 0 GRB Science

Swift now performs ASAP observations of GCN alerts from SVOM ECLAIRs and Einstein Probe WXT.

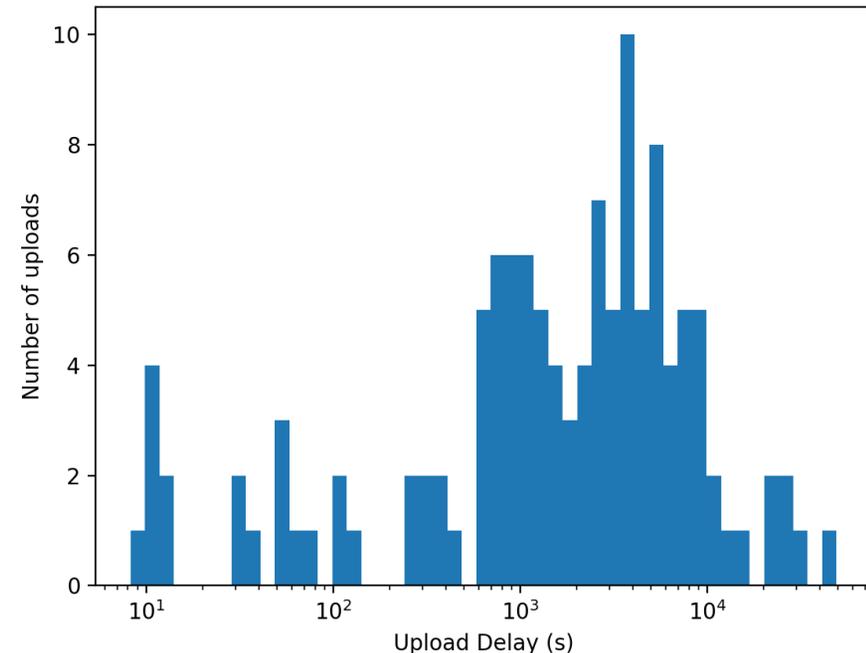
- Swift XRT/UVOT means rapid identification of source class and arc-second resolution observations.
- Results published over GCN and available live at these two websites (thank you Phil Evans):
 - <https://www.swift.ac.uk/EP>
 - <https://www.swift.ac.uk/SVOM>

Swift has proven that it can perform rapid GRB observations not only of triggers from BAT, but also from other missions.



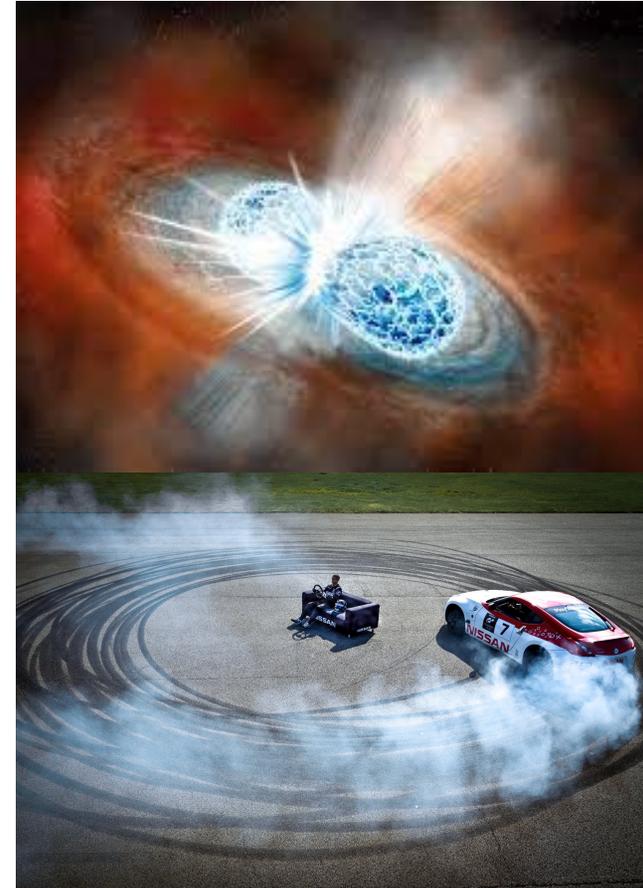
Urgency 0 Statistics

- So far we have performed 155 Urgency 0 automated uploads.
- Fastest response times have been of the order of ~ 10 s from receiving the TOO at the MOC and slewing to it.
- Median response time is ~ 2.2 ks. This reflects that fact that Swift can only see 25 - 42% of the visible sky at any moment (58% blocked out by Earth, the rest by Moon/Sun).



Swiftly Chasing Gravitational Waves across the Sky in Real Time

- Tohuvavohu et al. (2024, ApJL, 975, L19) proposed a novel use of Urgency 0 TOO requests. Trying to catch GW events with BAT as the merger happens, using so called “Early Warning” events from LIGO/Virgo/KAGRA.
- How to do this:
 - Firstly we need to reduce latency as much as possible on the LVK side for Early Warning Alerts.
 - Secondly we need to be able to rapidly respond to these events at the Swift MOC.
 - Thirdly, we need to calculate the optimum way to align BAT with the LVK error region.
 - Fourthly at the time of the merger, we should perform a GUANO dump.



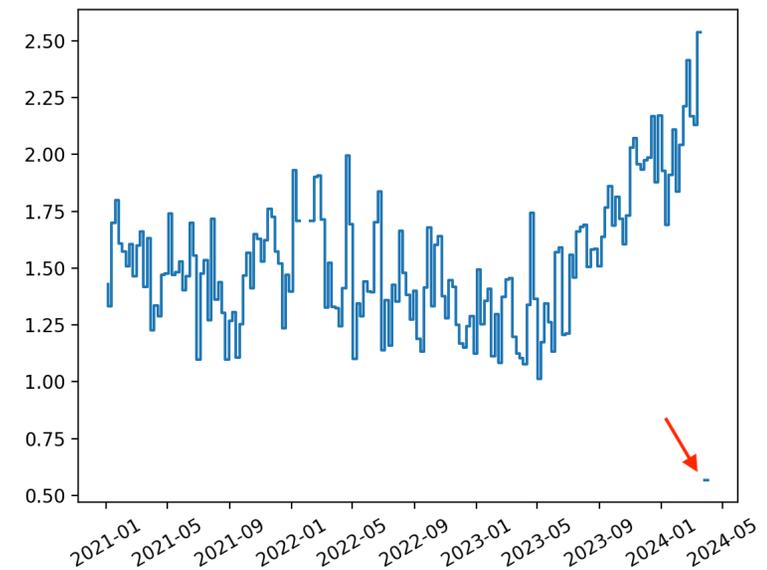
Swiftly Chasing Gravitational Waves ...

- Worked with LVK group to develop and test a low latency pipeline to channel EW GW events through to our system.
- Developed software to process these events and uplink in realtime (processing time **0.2-1.7s**, uplink time~10s).
- Already triggered a few times, here's the timeline of the first:
 - **19:12:39.524**: Alert received
 - **19:12:39.813**: First TOO command written to database (~**289ms** after trigger)
 - **19:12:40.???**: Command sent to MOC computer for uplink.
 - **19:12:49.???**: Spacecraft receives of command (**bit rates makes this ~10s always**).
 - **19:12:53.???**: We get alert thru ITOS from spacecraft for slew (~**13s** after trigger received)
 - **19:13:00.641**: **T0 - Predicted merger time from EW alert.**
- In this case the spacecraft commanded to slew at **T0 minus 7s**. This trigger ended up not being real, **but is a strong demonstration of this process!**
- The next generation of GW detectors will be able to detect NS-NS mergers up to **15 minutes** before merger, so important to demonstrate this capability for the future.

Swift hardware status - gyro issue

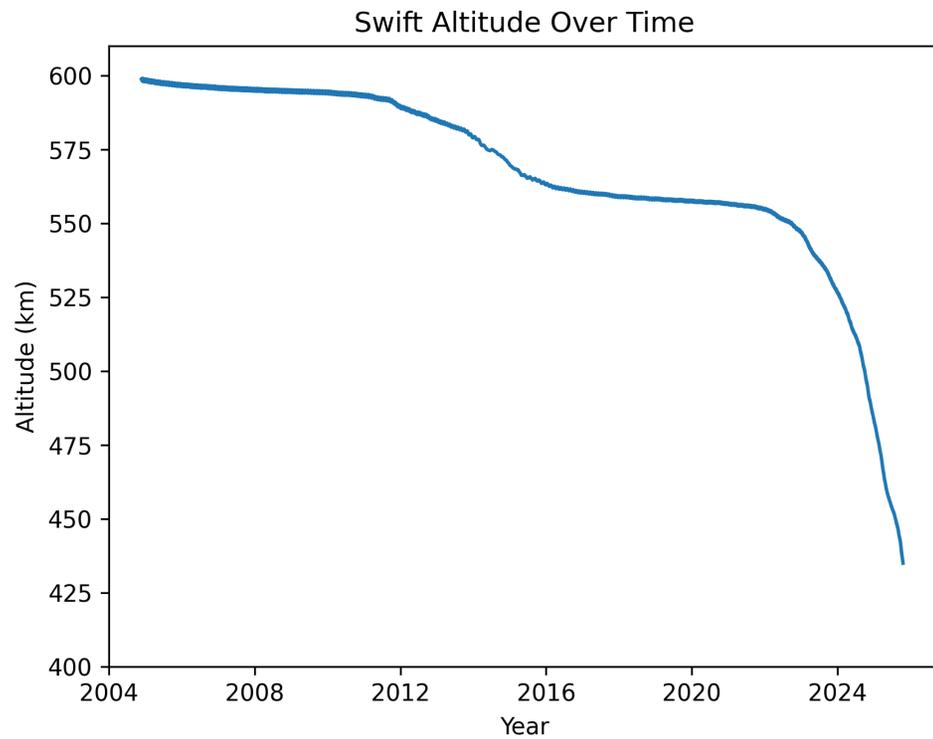
- *Swift* has 3 Two-Axis Rate Assemblies (TARAs, aka gyroscopes) on-board
 - Help provide attitude control (particularly during slews)
- TARA1 began degrading in ~ July 2023, and “failed” in March 2024
 - Unsuccessful slews (false BAT triggers)
 - Trailed UVOT images
- Two gyro operation required flight software patch
- Patch installation successful, returned to science observations ~ April 3
 - **Slew performance better than ever!**

Daily Average Slew Error



Swift won't last forever

- Swift's orbit is decaying, and it will likely re-enter in the next 12 months.



Extending Swift's Life

- There is an effort by NASA reboost Swift to a higher orbit ($\geq 550\text{km}$).
- This is necessary as after 20+ years on-orbit, Swift's orbit has decayed, and the expectation is that Swift will re-enter in the next 2 years.
- If boost mission happens, likely timeframe for launch is June 2026. This is an extraordinarily fast turn around time.
- NASA has selected Katalyst Space to perform the boost.

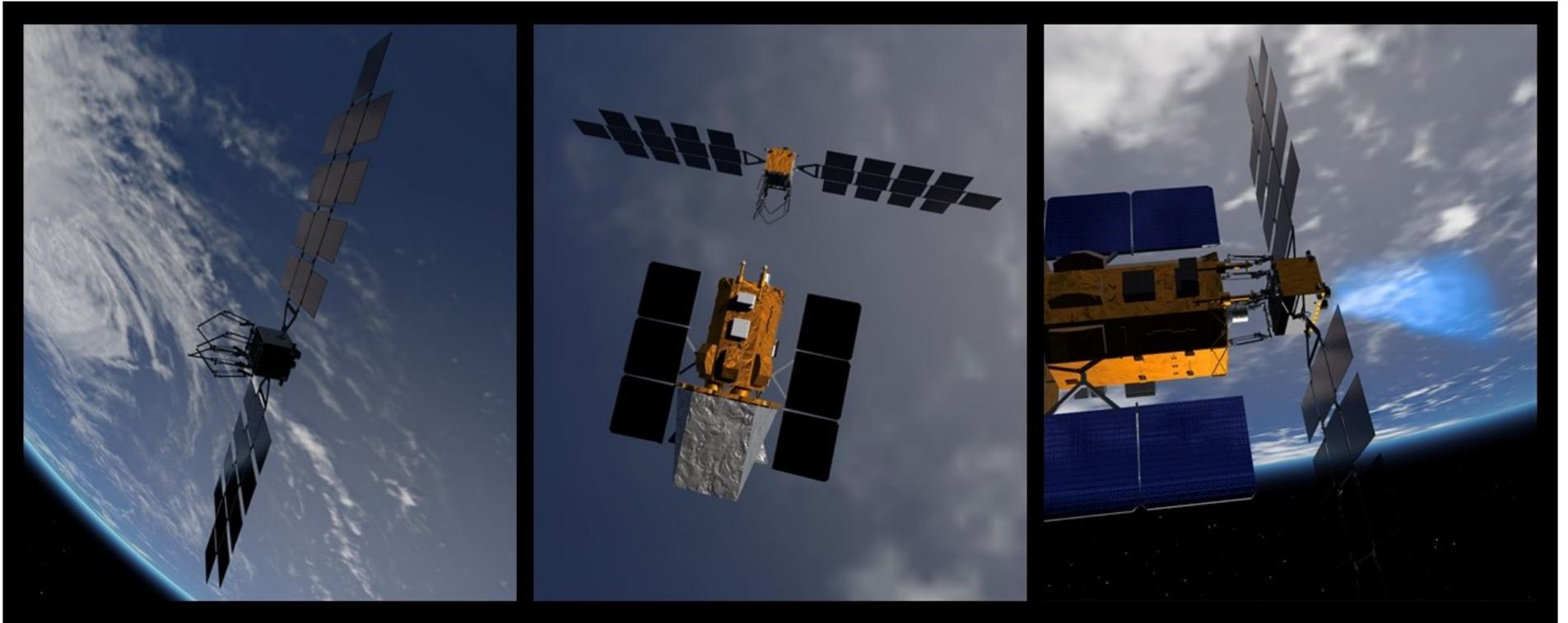
NASA Awards Company to Attempt Swift Spacecraft Orbit Boost



NASA's Neil Gehrels Swift Observatory, shown in this artist's concept, orbits Earth as it studies the ever-changing universe.

<https://www.nasa.gov/news-release/nasa-awards-company-to-attempt-swift-spacecraft-orbit-boost/>

Katalyst Space Boost Mission



Credit: Katalyst Space <https://www.katalystspace.com/post/nasa-telescope-is-about-to-fall-out-of-the-sky>