69th International Astronautical Congress (IAC), Bremen, Germany, 1-5 October 2018.

IAC-18,B4,3,4,x46796

ASTERIA Operations Demonstrates the Value of Combining the Mission Assurance and Fault Protection Roles on CubeSats

Amanda J.-N. Donner[a]*, Peter Di Pasquale[a] , Matthew W. Smith[a], Christopher M. Pong[a], Brian Campuzano [a], Mary Knapp[b]

[a] *Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Drive, Pasadena, CA 91109;* amanda.neufeld@jpl.nasa.gov
[b] *Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, MA, 02139,* mknapp@mit.edu

Abstract

On November 20, 2017, ASTERIA (Arcsecond Space Telescope Enabling Research in Astrophysics), a 6U CubeSat performing a technology demonstration of astrophysical measurements, deployed from the ISS. The technology demonstration goals to achieve precision photometry via arcsecond-level line-of-sight pointing error and highly stable focal plane temperature control were met by February 2018. Extended mission operations are ongoing, with the primary focus on observing nearby stars for transiting exoplanets. Throughout development and operations, the roles of mission assurance and fault protection have proven critical to achieving the primary technical goals and to maintaining a healthy spacecraft through multiple extended missions. Given the budget and schedule constraints typical of a CubeSat, innovative tailoring of processes has been critical to success throughout both development and operations of ASTERIA. Mission assurance plays an important role in identifying and evaluating risk and developing cost-effective mitigations. Flexibility in the fault protection design offers a variety of options for implementing risk mitigations as risks have been uncovered both in pre-delivery testing and in mission operations. This paper will discuss the approach taken on ASTERIA to implement mission assurance and fault protection and the resulting benefits to operational efficiency and success. It will briefly address the advantages of this approach during development, in which the combination of the roles provided mission assurance significant insight to system risks, which feeds back into testing methodologies and directly into fault protection design. Operations will be discussed in detail. During this phase, the roles merge to identify in-flight fault protection updates to efficiently respond to anomalies and improve the likelihood of successful technology demonstrations. The paper will also detail the tools that are used to analyse data, identify anomalies, and develop the updates to uplink to the spacecraft. Finally, the general operational approach will be discussed to highlight the usefulness of the ASTERIA processes and their applicability to future CubeSat missions.

Acronyms/Abbreviations

Arcsecond Space Telescope Enabling Research in Astrophysics, Massachusetts Institute of Technology (MIT), Jet Propulsion Laboratory (JPL), International Space Station (ISS), root-mean-square (RMS), Mission Assurance Manager (MAM), Mission Operations Assurance Manager (MOAM), Office of Safety and Mission Success (OSMS), software quality assurance (SQA), hardware quality assurance (HQA), electromagnetic interference (EMI), electrostatic discharge (ESD), single event effects (SEE), project system engineer (PSE), failure modes, effects, and criticality analysis (FMECA), electrical power system (EPS), flight software (FSW), mission scenario tests (MSTs), Morehead State University (MSU), Lightweight Directory Access Protocol (LDAP), two-line element sets (TLEs), North American Aerospace Defense Command (NORAD), virtual private network (VPN), operational readiness tests (ORTs), ground data system (GDS), Wide-field Infrared Survey (WISE), WISE Telemetry Command and Communications Subsysteam (WTCCS), tactical downlink (TDL), Open Mission Control Technologies (OpenMCT), Systems Tool Kit (STK)

1. Introduction

Based on the initial ExoplanetSat concept developed at MIT [1-8], ASTERIA was developed at the JPL in collaboration with MIT beginning in late 2014. The project was funded by the JPL Phaeton Program, a program intended to give early career hires flight project experience from development to operations with the guidance of experienced mentors.

As one of the first CubeSats to be assembled, tested, delivered to launch and operated by JPL, the aggressive nature of the budget and schedule was an adjustment for the JPL team. These challenges fostered a culture among the project of innovation and efficiency, which contributed to the ultimate success of the project. This paper describes the implementation of this approach in the mission assurance and fault protection areas during development and operations. Below is a timeline of the key dates on ASTERIA so far:

Table 1. ASTERIA project timeline.

Date	Event
2014 December 11	Project Kickoff
2015 March 3	Mission Concept Review System Requirements Review
2016 February 24-25	Design Review
2017 June 1	Delivery to NanoRacks
2017 August 14	Launch to International Space Station (ISS) on SpaceX CRS-12
2017 November 20	Deployment from ISS
2017 November 21	Spacecraft Acquisition
2018 February 1	Completed Technology Demonstration Goals
2018 September 30	End of Extended Mission (plan as of this writing)

ASTERIA is a 6U CubeSat, (10.2 kg, 239 mm x 116 mm x 366 mm) designed to achieve the arcsecond-level line-of-sight pointing error and highly stable focal plane temperature control required to perform precision photometry. The spacecraft includes deployable solar arrays, 3-axis attitude control, and S-band telecommunications from vendors specified in Fig. 1. The payload consists of an optical telescope assembly and electronics stack. The payload electronics contain the flight computer and other hardware needed for pointing control and imaging capabilities. These electronics are complemented by the interface board, which houses the thermal control hardware. The mechanical chassis was designed for compatibility with the NanoRacks 6U deployer [9].

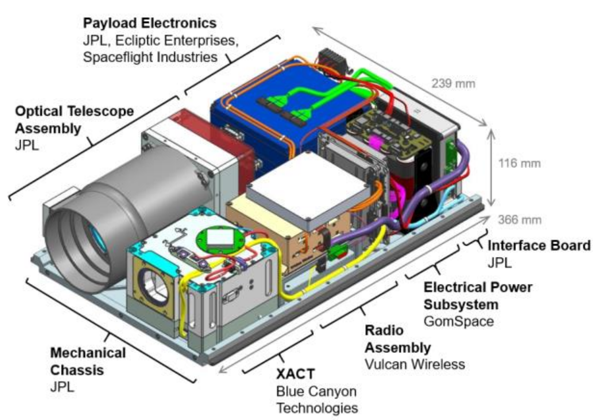

Fig. 1. Internal view of ASTERIA spacecraft (not pictured: solar arrays by MMA Design).

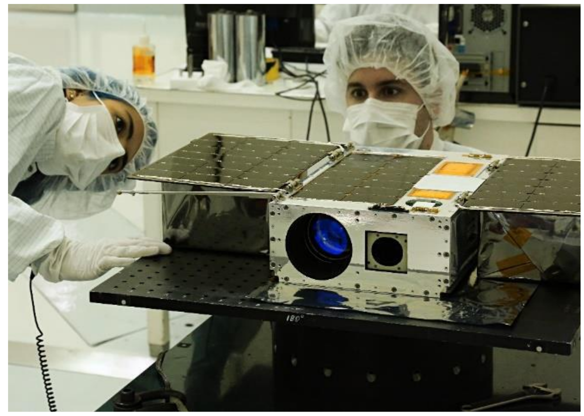

Fig. 2. ASTERIA flight model with solar arrays in the deployed configuration.

At final delivery, the ASTERIA spacecraft was integrated into the NanoRacks deployer by the JPL and NanoRacks teams. The NanoRacks team then packed and delivered the deployer containing ASTERIA for launch to the ISS on SpaceX CRS-12. The prime mission began when ASTERIA was deployed from the NanoRacks deployer in November 2017. By February 2018, ASTERIA demonstrated a temperature stability of ±0.01 K at a single location on the focal plane over a 20-minute observation, and pointing stability of 0.5 arcsecond RMS over 20 minutes and pointing repeatability 1 milliarcsecond RMS from observation to observation [10]. These capabilities have since been used to perform opportunistic science focused on transiting exoplanets, i.e. planets that pass in front of the star as seen from the telescope. During transit, a star's measured brightness will drop by a small amount, equal to the planet-to-star area ratio. ASTERIA has primarily focused on efforts to a) prove the capability to detect known transiting exoplanets by observing the star system 55 Cancri during expected transits of planet 55 Cancri e [11], b) determine whether known planets HD219134 d and f transit the star HD219134 [12], and c) look for signs of transiting exoplanets at the nearby star system, Alpha Centauri [13].

2. Tailoring of large-scale processes for a small-scale spacecraft

The MAM and MOAM are tasked with performing an independent evaluation of the risk facing the project against successful delivery and operations, respectively. This section focuses on the mission assurance approach during the development phase of ASTERIA, and the combination of MAM, quality assurance, and fault protection roles. Subsequent sections discuss how this approach affected operations of ASTERIA.

During the development phase, the MAM holds the responsibility for implementing the safety and mission assurance program on a project. Standard JPL practices are well-defined for larger risk-averse missions, but the thoroughness of these practices is cost prohibitive for CubeSat missions. One of the greatest initial challenges for the MAM during the early development phases of ASTERIA was to determine the best approach to tailoring JPL mission assurance requirements and practices on larger missions to the appropriate scale for this much smaller mission. Subsequent sections discuss the adjustment of the nature of the MAM role for ASTERIA, the tailoring of each of the mission assurance disciplines, and the combination of the fault protection role with the MAM role on ASTERIA.

2.1 Insight vs. independence

At JPL, the mission assurance roles are classically separated from the engineering design roles on a project, in order to ensure safety and mission assurance technical authority exists independent from programmatic authority. In practice, this avoids a scenario in which the same individual is responsible for both design and independent evaluation of said design, especially for critical elements of the mission such as fault protection. This is key to performing true independent evaluation of project risk. In some cases, the lack of independence can either cause "blind spots" in one's own area of design or cause the critical eye to be focused on one's own area of design resulting in "blind spots" elsewhere. Combining multiple roles across mission assurance and design groups is unconventional due to this potential conflict of interest. However, a limited budget on a CubeSat translates to limited funds for mission assurance, particularly the mission assurance manager, making the combination of roles a more attractive option. Combining roles allows the MAM to spend more time on the project not only under the MAM role, but under the additional roles. The insight into the system design that the MAM gains by applying this model fosters a thorough understanding of project risk. This is important in the areas of reporting risk to the institution, evaluating risk in the areas of anomalies, and applying mitigations in a cost effective manner (e.g., adjusting spending to apply more mission assurance focus on critical threats against meeting key requirements that may result in threats to operational efficiency not being mitigated).

The typical break down of the JPL mission assurance support team is shown in Fig. 3. In addition to delivering the mission assurance services to the project manager, the MAM has a separate reporting path to the JPL OSMS that is independent of the project, to encourage independent risk reporting throughout development and operations. This process was maintained on ASTERIA and was useful in ensuring significant risks were not "dropped" along the way. There are several disciplines that report to the MAM, including system safety, software quality assurance (SQA), hardware quality assurance (HQA), environmental assurance, reliability assurance, and electronic parts assurance. The way in which each of these were tailored, and specifically how the combination of MAM and HQA roles worked for ASTERIA, are described in section 2.2. The fault protection role is not shown here as it is a role separate from the safety and mission assurance division that was combined with the MAM role later in the project, as further described in section 2.3.

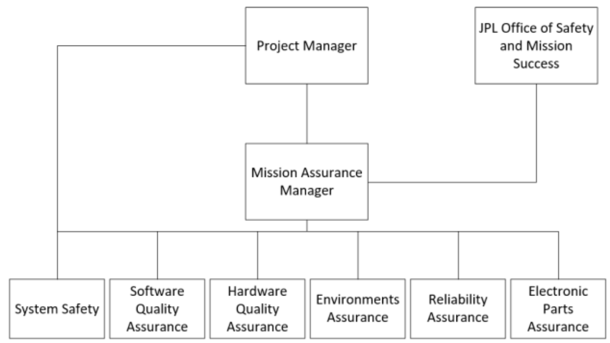

Fig. 3. Traditional project mission assurance organization chart.

2.2 Tailoring of mission assurance approach

Each of the mission assurance disciplines in Fig. 3 tailored the standard large-scale project approach to apply to the ASTERIA CubeSat project. The general approach to tailoring was to apply the standard practices that were capable of mitigating the most risk for the least cost. Each of the subsequent subsections describes the approach to tailoring applied by the mission assurance disciplines.

2.2.1 Quality assurance and system safety

From the beginning of ASTERIA development, the roles of MAM and HQA were combined. While these roles are typically separated on a larger project, in which HQA reports on independent evaluation of hardware risk to the MAM, the combination on a small project allowed the MAM to have additional insight into the hardware design and testing. HQA is responsible for ensuring the quality of flight hardware from the part level to the final assembly level. This involves inspections of individual parts and assemblies, flowing down the appropriate quality requirements to vendors, chairing subsystem quality reviews prior to system level assembly, and oversight during integration and testing of the final flight system. On ASTERIA, inspections were applied starting at the subsystem level, when parts were still primarily visible for inspection. Safe-to-mate procedures, in which resistance values are checked prior to connecting two electrical components, were performed at the board level. This provided another method of verifying the correct design prior to applying power. The combined HQA and MAM role then revisited the status of each subsystem after functional testing was completed in a review before a subsystem was incorporated into the flight system In this review, the status of inspections, rework, functional testing, and any outstanding anomaly reports prior to system integration and testing were addressed. This review also included experienced subject matter experts in the technical area associated with each subsystem. These gate reviews prompted the MAM to identify risk prior to moving forward with assembly and testing of the flight system, and offer mitigations for those risks moving forward. The combined HQA and MAM role continued supporting inspections and during integration and test, allowing for continued monitoring of progress on risk mitigations and of developments on newly identified risks. HQA assisted the small development team in ensuring the build and test processes were appropriately recorded with the focus on allowing for sufficient documentation in case troubleshooting was needed down the line if an issue was identified at a higher level of assembly. Having the MAM act as HQA also offered the MAM additional insight into anomalies that occurred during testing, which ensured the problem reporting was rigorous and timely. Problem reporting is further discussed in section 2.2.3.

System safety worked with HQA to perform surveys of lab areas for personnel and hardware safety hazards. System safety also supported the completion of the safety data package, containing system information required for delivery to NanoRacks for assurance that ASTERIA was safe to deliver to the ISS, which was critical to smooth delivery to and acceptance by NanoRacks in order to meet the launch deadline of August 2017.

SQA support was focused primarily on ensuring the appropriate approach to software development and configuration management at the beginning of project development, and to perform a review of the test reports and documentation accompanying the final flight software version. ASTERIA employed a flight software framework called F-Prime, developed at JPL [14]. SQA was involved in the development of F-Prime, which is intended to be reused across multiple projects. SQA has continued to support in operations for reviews of in-flight software updates.

2.2.2 Environments

Environmental requirements focused on reducing the potential harmful effects on ASTERIA by launch dynamics, EMI, ESD, and radiation environments. These requirements outlined testing, system measurements, and analysis required to verify the flight system is compatible with the expected environments. At a high level, the following approaches were taken for each category of environmental effect:

- Dynamics: Vibration testing was conducted on the final assembled flight system in the launch configuration in the NanoRacks deployer, with sufficient margin against the expected loads as predicted by NanoRacks.
- Thermal: A 7-day thermal vacuum test was conducted on the flight system including cold starts, hot starts, and functional testing at cold and hot extremes with sufficient margin against the predicted environment.
- EMI: A single "plugs-out" test was conducted of the spacecraft to ensure each subsystem operates nominally while undergoing interference from each other. It was identified that the camera and radio adversely interfere with each other, but operational workarounds were identified to mitigate this (see description of flight rules in section 4.3).
- ESD: The key method to mitigate ESD in flight was to define and meet bonding requirements. These requirements specified that bond resistances between various components on ASTERIA must be less than or equal to a certain value. This ensures an equipotential spacecraft to help mitigate potentially hazardous effects including ESD. Resistances between components were measured with a multimeter as the flight system was built up to verify requirements were met.
- Radiation: Total dose was not considered given the short mission life of ASTERIA, and the parts assurance group focused only on single event effects, discussed in section 2.2.3.

At the end of the thermal and vibration tests, the environmental requirements engineer reviewed the results and documented residual risks (if any) that folded into the project risk list.

2.2.3 Reliability and electronic parts assurance

One of the key roles of the reliability engineer is to define the required reliability analyses for flight designs, review the analyses that the designers perform for thoroughness and accuracy, and to offer support to the analyser as necessary. The parts engineer works with reliability to ensure parts will survive the harsh environments of space for the lifetime of the mission via analysis and/or testing. While larger missions approach these analyses to meet the required mission lifetimes of many years, the ASTERIA required mission lifetime was only 90 days. As a result, the reliability and parts assurance efforts were focused on decreasing the chances of a failure within this short time span. On ASTERIA, the reliability engineer aided in JPL designs by reviewing the parts list and schematic and ensuring there was adequate margin between the planned usage of the part (e.g. voltage and current at which it will be used) and the specification of the part. The parts engineer supported this effort by identifying parts that were potentially susceptible to single event effects (SEE) due to radiation based on previous analysis and/or testing of these parts. Specifically, parts that could be destroyed by a charged particle that hits the spacecraft (e.g., due to space weather events or passage through the South Atlantic Anomaly) were flagged for removal or additional circuit protection. The design engineer incorporated changes as necessary with guidance from the reliability and parts engineers.

Reliability assurance also defines the requirements for problem reporting on a project, and is typically a signatory on problem reports. On ASTERIA, formal problem reporting was initiated at flight system integration and test, and the MAM was the only mission assurance signatory. As previously mentioned, the MAM was often working on the floor as HQA when a problem occurred, which provided additional insight into the problem and residual risk after the corrective action was implemented. These residual risks were added to the project risk list, which was periodically reviewed by the ASTERIA management team composed of the MAM, PSE, and project manager.

2.3 Tailoring of fault protection approach

A key issue identified in the ASTERIA design review in February 2016 was the lack of a committed fault protection engineer. The MAM and HQA combined role still did not reach one full time individual, so the fault protection role was added to this position. Fault protection is a key flight software behaviour that monitors various telemetry on the spacecraft, detects off-nominal conditions when this telemetry is outside of normal bounds for some persistence, and executes responses to recover the spacecraft into a healthy state from these-off nominal conditions. The HQA role had given the MAM additional insight into the hardware system, and fault protection role added insight to the software design and associated risks. The MAM is typically a full time role on a single project but on a small project the overlap of roles was critical. The additional insight gained allowed the MAM to be a truly valuable contributor to system risk discussions and design decisions.

Similar to the mission assurance approach, the fault protection design process was also tailored from the standard approach for large missions. Failure modes, effects, and criticality analysis (FMECA) was used to identify failure modes in each subsystem that could be recognized and recovered by a flight software fault monitor and response. While this list became lengthy, the goal was to end up with simple fault protection design in order to keep the flight software integration and testing of the fault protection system manageable, and to minimize the risk that a complex fault monitor and response system could actually create new problems. The full list of over 100 potential failure modes in the FMECA was filtered down to a short list of approximately 10 key faults that could be used as a "catch-all" for a variety of faults. Section 3 further describes the final fault protection at delivery.

3. Fault protection design and testing

3.1 Mode Manager and Fault Responses

The mode manager was designed prior to the design of fault responses, with fault response architecture in mind. The architecture of the mode manager allows for fault responses to manage transitions between modes in the mode manager in order to establish a safe state on the spacecraft.

3.1.1 Mode manager

The design of the mode manager, as incorporated in the delivery version of ASTERIA flight software, is defined in Fig. 4 [9].

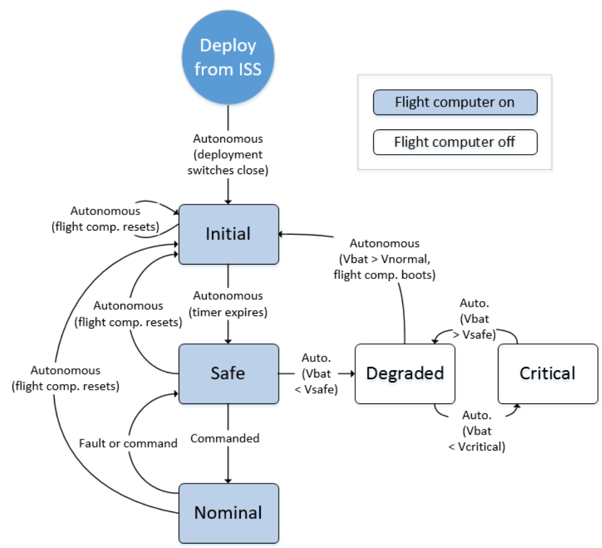

Fig. 4. ASTERIA mode manager design.

"Vbat" is the as-measured battery voltage and "Vnormal", "Vsafe", and "Vcritical" are voltage thresholds (in decreasing value) that govern mode transitions conducted by the EPS (i.e., modes in which the flight computer is off). In Degraded Mode, only the attitude control system (Blue Canyon XACT) remains powered on by the EPS. In Critical Mode, the EPS itself powers down to a mode in which the battery is able to charge. In flight to date, the battery voltage has never dipped below the Vnormal threshold.

Initial mode holds the logic for deployment of the solar array and deployment timer for initial power on of the radio, in order to ensure plenty of time for the spacecraft to detumble and charge the batteries before the radio powered on. Once this timer completed, a flag was set so that the solar array deployment and radio timer delay never executed again when the system passes through initial mode on safe mode entries after deployment. ASTERIA spends a majority of its mission in safe or nominal modes. Entry into nominal mode must be commanded and does not change any power states. Nominal mode does enable additional fault monitors, such as the "sequence failure" fault described in section 3.2.1. For this reason, flight rules govern that sequences may only be run in nominal mode on ASTERIA.

3.1.2 Safe mode response

The intent of the safe mode response is to power off the payload and ensure the spacecraft is operating in a power-positive manner with sufficient opportunities for ground intervention. This response occurs when a fault occurs in nominal mode (which, per flight rules, is the only mode in which sequences can be run and/or the payload can be operated). In the launch version of flight software, the safe mode performs the following basic actions:

- Stops all running sequences
- Powers off the payload hardware (does not power cycle flight computer)
- Asserts the radio into a duty cycling mode to ensure a power-positive state that still ensures the radio be powered on for at least 70% of pass opportunities
- Commands the XACT to a sun-pointed mode

3.1.3 Reset response

The intent of the reset response is to be a "hard hammer" response that can recover the spacecraft from most faults as long as flight software is running. The reset ends with the spacecraft in safe mode, regardless of the initial spacecraft state prior to the reset. The following actions are performed upon a reset:

- Stops all running sequences
- Powers off the payload hardware (does not power cycle flight computer)
- Powers off the radio
- Powers the XACT off, then back on, in order to ensure it is on and maintains a power-positive state in case the next step fails for any reason
- Sends a command to set timers on the EPS to power off the flight computer and then back on 30 seconds later, which will boot the system into Safe Mode

3.2 Fault monitors and watchdogs

The health and safety of the ASTERIA flight system is protected by flight software-controlled fault monitors, and watchdogs external to the flight software, discussed in sections 3.2.1 and 3.2.2, respectively.

3.2.1 Fault monitors

The following fault monitors are examples of ASTERIA monitors that have tripped in flight that act as "catch-all's" for a variety of lower level faults:

- **XACT off-sun:** If the XACT reads a bad sun vector for greater than 43 minutes, a fault protection reset response is called.
- **Low battery voltage:** If the battery voltage reads below a threshold for some persistence, this fault will trip. At launch the response to this fault was an entry to safe mode, but it was updated to a reset in response to an in-flight anomaly as described in section 5.4. This fault has recovered the spacecraft from loss of attitude control and subsystem over-temperature due to excessive current draw.
- **Sequence failure:** If a sequence fails, the system will enter safe mode and stop the sequence. This catches any command errors that result in an execution error.
- **Command loss:** If the FSW module responsible for uplink does not receive a command for a period longer than the time expected to elapse between communication passes, a fault protection reset response is called.

All of the above fault monitors have tripped in flight, and recovered the spacecraft into a safe state.

3.2.2 Watchdogs

There are two key watchdogs that are external to flight software that monitor system health. Both watchdogs reside on the EPS and are defined in Table 2.

Table 2. ASTERIA watchdogs.

Watchdog	Pet method	Trigger case	Trigger response
Flight computer output watchdog	FSW responds to EPS pinging the flight computer channel	Number of failed pings > N seconds (configurable)	EPS power cycles flight computer output
EPS command loss timer	Ground command	Watchdog is not reset by ground command within 7 days	A power cycle of the EPS is performed, effectively power cycling all outputs

The watchdog on the flight computer output has tripped over ten times in flight, safely recovering the spacecraft into a safe state. The EPS command loss timer has not tripped in flight.

3.3 Pre-launch fault protection system testing

All key faults that could be reproduced without hardware manipulation or potential risk of damage were tested on the flight system. Fault monitors that could not be tested on the flight system (i.e., required hardware manipulation) were tested on the flight version of flight software on the ASTERIA testbed. All responses were tested on the flight system. Additional off-nominal testing was conducted via MSTs, that went through scenarios such as an off-nominal deployment (i.e., where degraded mode was entered) to ensure the system recovers gracefully from off-nominal scenarios. One key parameter change was identified in these mission scenario tests. Specifically, a file system check executes upon boot after non-graceful power cycle of the flight computer. In MSTs, it was determined that this file system check exceeded the timeout of the flight computer output watchdog timer, causing the watchdog to power cycle the flight computer prior to completing the file system check. This occurred because flight software did not load until after the file system check completed, and this watchdog was not serviced unless flight software was running. This could lead to an infinite loop where the file system check is interrupted during each boot, never allowing flight software to load. As a result, the parameter for the timeout period for the flight computer output watchdog was increased and has been sufficient to recover the spacecraft in flight to date when this watchdog has tripped.

3.4 Fault protection in-flight update capabilities

The fault protection system design included update capabilities that did not require a flight software update. These capabilities allow changing parameters governing when/if fault monitors trip (e.g., threshold above which a monitor may trip), disabling/enabling fault monitors in each system mode, disabling/enabling responses in each system mode, and changing the responses that each fault monitor calls when it trips. These options were utilized early in the mission to better mitigate unexpected anomalies, as further described in section 5.4.

4. Mission assurance and risk management at delivery and during prime mission operations

4.1 Pre-ship review

At the pre-ship review, or the final review prior to delivery to NanoRacks for integration into the launch vehicle, the MAM, PSE, and project manager presented the flight system status, particularly the top risks from the final project risk list that had been updated by the ASTERIA management team throughout development. Many of the key risks were identified by mission assurance disciplines, such as the susceptibility to single event effects due to the radiation environment. These top risks, along with a detailed collection of data from tests and analyses conducted on ASTERIA throughout development were presented to lab management prior to delivery for launch. The residual risks were accepted by the ASTERIA team and review board, and noted as potential risks to be managed during operations. These risks were actively considered as operations proceeded, as further addressed in section 4.3

4.2 Operations background

As identified in Table 1, true operations of the spacecraft began on November 21, 2017 when ASTERIA was first acquired after deployment from the ISS. Between delivery and NanoRacks and first acquisition, significant work was conducted to ensure the ASTERIA team, ground software, Morehead State University (MSU) ground station, ASTERIA testbed, and science team were prepared for operations. The overview of the ASTERIA operations systems and interactions is provided in Fig. 5.

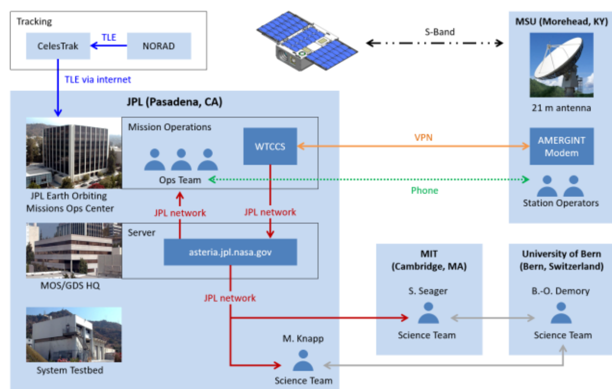

Fig. 5. ASTERIA flight operations overview

The overview shows all of the key components of ASTERIA flight operations. The TLEs from NORAD provide tracking information for the spacecrraft, used by the mission operations team to predict the start and end times of passes over the MSU ground station. A single pass is typically around 10 minutes in length. A VPN connection to the JPL AMERGINT modem at MSU allows live commanding and receipt of data during each pass. Prior to delivery, compatability testing between the flight radio, JPL AMERGINT modem, and MSU ground station was conducted. Residual issues with the ground system (unrelated to the radio) were investigated upon another visit to MSU post-delivery, and workarounds were implemented including leaving the JPL AMERGINT modem at MSU for flight use, to ensure reliable downlink quality during the mission [15][16].

After each pass, the downlinked data is posted to the server which is accessed by the operations team for review. Science data, in the form of images and corresponding payload telemetry (e.g., focal plane temperatures and pointing control data) is also posted on this network, which is accessed by the science team for analysis. In July 2017, the team conducted a week-long series of ORTs that simulated the processes in Fig. 5 with 10 minute "passes" about 90 minutes apart, as expected for flight operations. The AMERGINT modem to JPL network capabilities (i.e., VPN connection) were tested in a brief "refresher" ORT involving MSU in early November 2017. These ORTs also exercised the GDS on the JPL network side in Fig. 6, other than the GitHub repository for uplink products which was added later in operations as an improvement to the product approval process, as discussed in section 5.1.

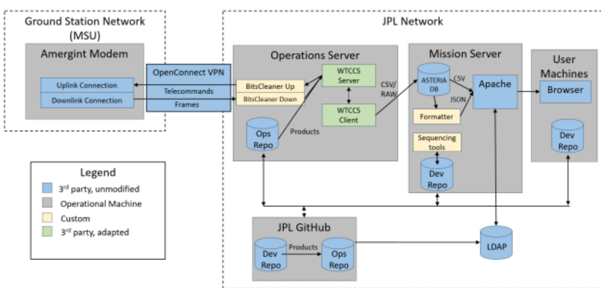

Fig. 6. ASTERIA GDS overview [16]. Note that the ASTERIA testbed is not included in this diagram.

The ASTERIA GDS system is based on the software and interfaces developed for the WISE mission, with additions included to streamline operations for the small ASTERIA team. The WTCCS interface is used for commanding, and all other operations tools are either unmodified software (Apache, GitHub), or custom MATLAB or Python tools designed for interface compatibility or operations streamlining, many of which were written during the prime mission when opportunities for efficiency improvements were identified. Query and database access are controlled through basic authentication provided by the JPL LDAP and Apache [16].

4.3 Initial approach to operations

As the first CubeSat mission to be operated out of JPL, and with a very limited operations budget after delivery, the ASTERIA operations team had to be innovative and efficient. Standards for CubeSat operations at JPL were not clearly defined, so again tailoring the typical approaches used on larger missions was required. One of the most effective methods of mitigating risk during the prime mission was to ensure a majority of the team was composed of the same expertise involved in the development phase. This was also cost-effective in that minimal training was required for the operations team to understand the flight system and its vulnerabilities.

The individual serving as MAM during development became MOAM during operations, and maintained the fault protection role as well. These roles were critical to balancing the risks against meeting the prime mission objectives. The most important aspect of the risk management approach was maintaining a pace of mission operations that made efficient progress toward accomplishing mission objectives while simultaneously protecting the spacecraft against operator errors. One of the key risks identified at delivery was the risk of using commercial parts in space, many of which had not been previously flown or tested for response to single event effects. As a result, there was a certain urgency to achieving the technology demonstration goals as quickly as possible. However, it was also important to take the appropriate precautions in evaluating uplink products prior to execution on the spacecraft to ensure no accidental command errors ended the mission before technology demonstration goals were met.

During both the prime and extended missions, the primary methodologies for ensuring uplink products are safe are testing on the ASTERIA testbed and/or using a text comparer to compare products (a process often referred to as a "diff") to products that have previously successfully executed on the spacecraft. The uplink product is also always evaluated for compliance to flight rules, which are a list of rules that were collected during development and updated after a review for completeness during ORTs. The sequence developer completes an uplink approval form describing the general purpose of the sequence, documenting the approval method (testbed report and/or sequence against which it was compared), and any flight rule violations. At this time, all flight rules were reviewed for violations by the sequence writer. The mission manager documents approval of the sequence on this form and waives any flight rules, with MOAM approval, as necessary. Flight rule waivers were not written separately (as they are on larger mission) in order to reduce document maintenance overhead for a small team. Some examples of flight rules include:

- Only run sequences in nominal mode (to allow for the safe mode response to recover the spacecraft from a command error in a sequence or from a payload anomaly).
- Do not allow the spacecraft to be in a non-sun pointed attitude (including eclipse) for longer than 40 consecutive minutes. This prevents a specific fault monitor from tripping that resets the spacecraft.
- Do not operate the camera and the radio at the same time, not only to maintain a power-

positive state during observations but also to prevent EMI between the two subsystems.

There have still been some command errors in operations (see below), but none that have ended the mission. The operational improvements made to prevent recurrence of these issues are discussed in section 5.2.

As anomalies occurred (both related and unrelated to command errors), the combined MOAM and fault protection role was especially useful. Many issues that occurred required specific risk evaluation by both roles. For instance, early in the mission there was an XACT anomaly that was not threatening to spacecraft health, but was preventing progress towards technology demonstration goals. At that time, the XACT had not been power cycled in flight. The team worked together to evaluate the risk of power cycling the XACT, how fault protection might respond if the power cycle did not complete correctly, and the consequences of worst case scenarios to the success of the mission. The MOAM's insight to the fault protection aided the evaluation of the risk of this activity, which was ultimately completed and recovered the XACT's ability to move forward with technology demonstration activities. This power cycle sequence was stored on board the spacecraft and became useful in when XACT anomalies occurred later in the mission that did threaten spacecraft health.

As anomalies occurred in operations, the need for improved operational efficiencies were identified to help prevent the recurrence of command errors that cause anomalies and to decrease the time between and anomaly occurring and resuming technology demonstration activities. Section 5 details many of these improvements that have been made throughout prime and extended missions.

5. Operations improvement throughout prime and extended missions

In order to preserve a majority of the limited funds for the operation phase, the team did not work on ASTERIA activities between delivery and operations other than for ORTs. As a result, the work necessary for conducting operations was sufficiently complete at deployment, but there were still opportunities remaining for efficiency improvements. The improvements made throughout the prime and extended missions are detailed in subsequent sections.

5.1 Uplink product configuration management

As identified in Fig. 6, GitHub is utilized for development of flight uplink products. GitHub was used on ASTERIA during development for configuration management of flight software, so the team was familiar with the tool, Initially in operations, the team used a more manual process of transferring tested products to the operations machine, however this was not streamlined for long-term use. The use of GitHub was the first applied very early in operations and the efficiency savings were a significant improvement. Two separate repositories were created, one for sequences under development and one for uplink after approval. The development repository is used as a "sandbox" for creating and editing of uplink products. The ASTERIA testbed is connected to this repository so products on this repository can be easily run on the testbed. Uplink approval forms link to the version (identified by a unique series of numbers and letters called a "commit hash") of the product in the "sandbox" that was tested on the testbed and/or approved via a "diff" against an approved product in the operations-approved repository. Only mission managers or deputy mission managers with the appropriate permissions can commit to the operations-approved configuration managed, "cm", repository. Once a product is in the "cm" repository, it cannot be edited per flight rules. The operations machine pulls from the "cm" repository and loads the uplink products into WTCCS for translation and uplink to the spacecraft. This streamlined process has worked well through the remainder of the prime mission and both extended missions. Ultimately additional repositories were added to GitHub for configuration management of MATLAB and Python tools used to generate sequences and downlink commands, respectively.

5.2 Sequence generation

As previously discussed, several processes, including sequence generation, were very manual at the beginning of the prime mission. This resulted in command errors that caused a reset and/or a missed pass opportunity, some examples of which are as follows:

- **Command error 1:** Did not command spacecraft back to sun point after observation in sufficient time, causing the "XACT off-sun" fault to trip and cause a reset.
- **Command error 2:** Absolute time commands entered in sequence as one day in the past, so commands executed immediately when uplinked to the spacecraft and not at the correct time to set up the following pass.
- **Command error 3:** Did not account for the time each compression command takes to complete in addition to the relative delay between commands specified in the sequence. As a result, sufficient time for file compression between passes was not allowed, causing the commands to be delayed and the radio to be off (as it is during compression) for the next pass.

Sequence generation tools were created and edited as a result of these and other command errors. The two primary categories of sequences generated by these tools are pass setup sequences and science observation sequences, the latter discussed further in section 5.6.

Pass setup sequences improve downlink rate by targeting a spacecraft antenna at the ground station during planned pass times. Command errors 2 and 3 both occurred in pass setup sequences. In response to command error 2, the sequence translation tool in Python was updated to return an error any time the dates in the sequence in the past. In response to command error 3, a MATLAB tool that checks and plots the timeline for a sequence was updated to include an additional delay for each compression command, to account for the time a single file compression may take. The tool plots the timeline based on absolute and relative times in the sequence. If relatively timed commands will not finish prior to the next absolute time command, it will return an error and not create a plot. An example of the plot of a sequence that includes an absolute time command followed by hours of relatively timed commands including compression, followed by more absolute time commands (to set up passes) is shown in Fig. 7.

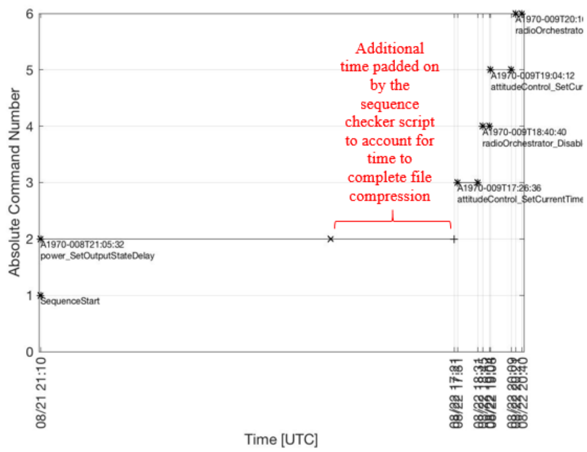

Fig. 7. Timeline of sequence including compression commands.

Note that the X-axis is UTC time corresponding with an absolute time command or the time at which all relative commands after that absolute time command complete. This it typically viewed in MATLAB where the zoom capabilities can be utilized to better view the timeline. This tool has been regularly used in both prime and extended missions, not only to check pass setup sequences, but also to confirm the validity of all sequences that include a mixture of relative and absolute times.

5.3 Spacecraft data analysis and downlink tools

At the beginning of operations, the data decoding and storage process was set up with the intent to provide raw text data formats that could be processed by various local tools that each subsystem lead developed for initial checkout. A single Python script handled the post-processing of all data. In early operations, the key functions of this script included running a flight software decoder tool on files downlinked from the spacecraft file system, processing live data recorded in CSV (comma-separated) format during each pass, and posting the raw and decoded files on the server accessible by the JPL operations team via LDAP authentication. This allowed the team to access the decoded files to plot trends and perform analyses with their various tools. Each subsystem performed a thorough review of data during checkout with similar tools used prior to delivery to verify subsystems were operating as expected.

As checkout was completed and subsystem leads rolled off, more detailed analysis became unnecessary and unrealistic given the shrinking team. Remaining systems engineers identified a need for a simpler method to achieve a higher-level review of data for consistency (nominal behaviour) or obvious anomalous trends. The ASTERIA web architecture shown in Fig. 8 shows the improvements made to address this need. The key improvements included the creation of TDL tools to improve efficiency of strategic downlink of important files and the incorporation of OpenMCT to plot decoded and recorded telemetry for easy viewing of data trends [18].

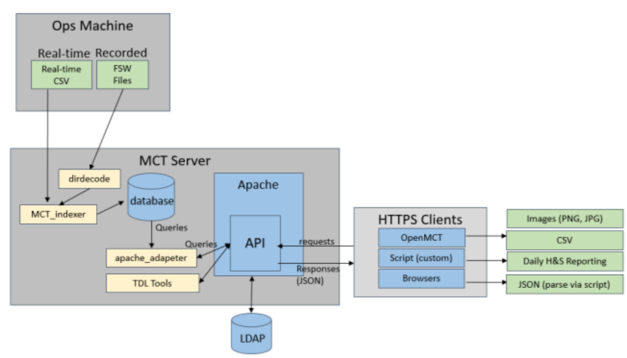

Fig. 8. Web architecture for ASTERIA.

5.3.1 Tactical downlink (TDL) tools

The file system on the ASTERIA flight computer does not prioritize or order files for immediate and automatic downlink upon acquisition of signal lock. Instead, in order to downlink the most recent telemetry and event logs, a listing of files must be generated and downlinked, and then commands to downlink selected files must be generated between passes. In order to allow the latest files to be downlinked in the same pass, scripts were written to compare this downlinked file listing to the list of files that have previously been downlinked. The script generates a series of commands to downlink files that are on the spacecraft but not yet on the server, as well as a series of commands to delete files that are on the spacecraft that have already been downlinked. This allows for quick downlink of data within the same 10 minute pass, which can be useful for troubleshooting anomalies that occurred between or during passes. This also greatly increases operational efficiency in that downlink commands do not have to be manually generated from a file listing.

5.3.2 Data analysis with OpenMCT

Selected critical telemetry for evaluation of spacecraft health is recorded regularly on the spacecraft. This health and safety packet is decoded onto the server, and each telemetry point is plotted in OpenMCT within minutes after the end of a pass. This has been extremely useful in evaluating trends quickly and aiding decision-making in response to anomalous trends. A specific example of this played out about a month into the prime mission. An anomaly occurred off-pass in the XACT, which resulted in the spacecraft tumbling. There were some hints of an anomaly during the pass due to variation in carrier strength, so the TDL tools were used to quickly downlink the most recent health telemetry, which was immediately posted to Open MCT after the pass. Quick evaluation of plotted telemetry showed that the system was not maintaining a sun-pointed attitude and that battery voltage was trending down. On the next pass, a reset was commanded which resolved the issue and recovered the spacecraft into a sun-pointed, safe attitude. A screenshot from OpenMCT in Fig. 9 shows the plot of battery voltage telemetry that was evaluated between passes to decide to command the reset on the following pass. Annotations describe the events that occurred before, during, and after the anomaly and recovery post-reset.

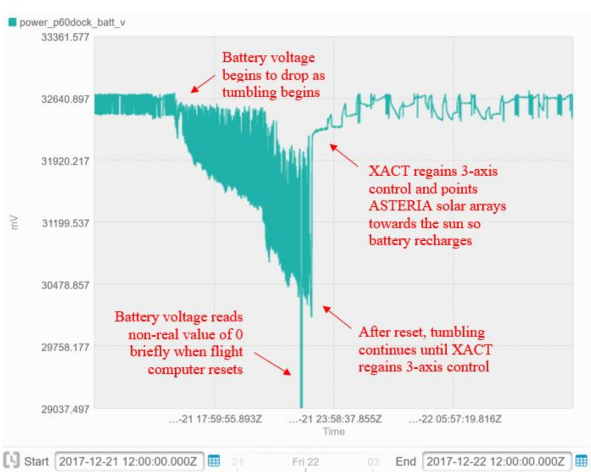

Fig. 9. Annotated view of OpenMCT plot of ASTERIA battery voltage before, during, and after the initial XACT anomaly that caused the spacecraft to tumble.

It was determined that if the reset had not been commanded in the following pass, that fault protection may not have power cycled the XACT to resolve the anomaly. This was because the response to a low battery voltage was to go to safe mode, which would command a sun pointed attitude, but not power cycle the XACT. Further analysis of the anomaly showed that this would not have resolved the issue. Prior to delivery, no analysis indicated that a power cycle of the XACT would be necessary upon safe mode entry. It was shown that commanding the XACT to sun point mode would recover the spacecraft from all scenarios that might lower the battery voltage. However, this specific fault case was not known or observed prior to launch. Fortunately, the fault protection design allowed in-flight modification without a flight software update.

5.4 In-flight fault protection modifications

The ability to update particular aspects of fault protection has proven very useful in ASTERIA operations. After the XACT anomaly described in section 5.3.2 occurred, a new fault table was uplinked that changed the response to a low battery voltage from a safe mode entry to a reset, to ensure the XACT would be power cycled if the same fault were to recur. Weeks later in the mission, a different XACT anomaly occurred that also caused the spacecraft to tumble. This anomaly occurred off-pass, so the operations team could not intervene. As a result, the low battery voltage fault tripped, calling a reset which power cycled the XACT. Data downlinked in the next opportunity for communication revealed that the spacecraft recovered a stable sun-pointed attitude within minutes after the reset.

While the fault protection modifications were sufficient to protect the spacecraft health and safety from the two XACT anomalies that resulted in tumbling, system-level resets were disruptive to operational efficiency. As a result, updates to fault protection that required a flight software update were discussed. Specifically, two fault monitors that could catch each of these anomalies long before battery voltage dropped were designed, and safe mode entry was modified to include an XACT power cycle. These updates were included in a new flight software version that ran successfully on the testbed, so the next step was to uplink it to the spacecraft.

5.5 Flight software update via bspatch

The ASTERIA flight computer runs Linux, and flight software allows low-level access to shell commanding. This has been useful in many applications during ASTERIA's mission, one of which has been updating flight software in an efficient manner. The original flight software update procedure demonstrated prior to delivery was uplink-intensive to the point that it would have been prohibitive operationally. The procedure involved splitting the executable into pieces, compressing the pieces and uplinking each piece to the spacecraft. Once every portion has been uplinked to the spacecraft, each piece had to be decompressed on the flight computer and then concatenates to recreate the new flight software image onboard. Based on laboratory testing of a ~1.5 megabyte flight software image, the update process would have taken a week's worth of passes, which would have significantly affected operational efficiency.

The new flight software approach centers on the bsdiff and bspatch libraries [19]. The usual challenge with creating patch files for flight software updates is the pointer problem. There are numerous references to memory addresses in an executable, and as a result even a small code change can result in massive patch files. The bsdiff tool tackles the pointer problem in by using a suffix sorting algorithm [20]. As a result the ~1.5 megabyte image was reduced to a 100 kilobyte patch file, reducing the uplink data volume requirements by an order of magnitude. This allowed the ASTERIA team to uplink the patch file and run bspatch to recreate the flight software image in a single pass. The process for completing the update was as follows:

- Create bsdiff tool on ground computer
- Create bspatch tool for the flight computer, to uplink later
- Use bsdiff to create a patch file using: bsdiff base_image new_image patch_file
    - base_image - This is the current FSW image that is already on the flight computer.
    - new_image - This is the new FSW image that you want to run on the flight computer.
    - patch_file - This is the output patch file that will be uplinked.
- Uplink bspatch, patch file, and new FSW image hash file to flight computer
- Run bspatch on flight computer via shell commanding to reconstruct new FSW image.
- Copy new FSW image (via shell commanding) to a file location that the ASTERIA boot driver will access on a single reset to volatilely load the new FSW image
- Command a reset to load the new FSW image in a volatile manner
    - If there are any issues with the new version of FSW that prevent it from loading correctly, the flight computer output watchdog on the output watchdog will cause another flight computer reset that will then load the known "good", non-volatile version.
- After at least a week of successful operations on the spacecraft with the new FSW, to copy the new patched FSW version into the location of the "prime" flight software image, so it is loaded and run in a non-volatile manner.

In February 2018, this process was successful in updating flight software to include new fault monitors for two of the XACT anomalies that occurred and include an XACT power cycle on safe mode entry, among other improvements. That flight software version has run as the prime version successfully for 6 months.

5.6 Science observation planning tools

The first versions of the science observation planning tools were developed for the technology demonstration phase. These tools are in MATLAB, with one tool being the primary interface for sequence generation. This tool takes various inputs and outputs a series of commands with predetermined relative timing as defined in the script based on lessons learned from previous observations. One of the primary issues that has disrupted the success of observation sequences is the unreliable initialization of the imager. The fine grain timing of this initialization is not accessible via flight software commanding, but is accessible via shell commanding. Throughout the prime and extended missions, the series of shell commands used to initialize the imager have been adjusted over time to optimize the success rate of observations. These changes are all made in the sequence generation tool, which is also configuration managed in GitHub. GitHub maintains a history of changes that can be referenced to recall adjustments that had been made to observation sequences previously, which has been critical as the team troubleshoots various issues, including but not limited to the camera initialization challenges, encountered during observations.

The science observation generation process is outlined in Fig. 10. Pre-work for identifying stars to observe and ASTERIA camera access to those stars is done in AGI Systems Tool Kit (STK). Inputs to the sequence generation tool are identified in blue boxes. The tool includes prediction of off-sun time, spacecraft momentum buildup, and reaction wheel zero crossings. Prediction of off-sun time was especially important to avoid recurrences of command error 1 described in section 5.2. Off-sun time must be under a limit in order to avoid tripping the XACT off-sun fault which causes a reset. The tool also outputs plots of spacecraft momentum, reaction wheel speeds and zero crossings (indicated by the dark marks that are purple X's in top plot in Fig. 11 in Appendix A), further described below, and a map of the portions of ASTERIA's orbit during which observations will be conducted (example in Fig. 12 in Appendix A).

Momentum management is critical on ASTERIA given that a large residual dipole exists on the spacecraft. This was identified too close to delivery to reduce, and operational workarounds were identified instead, including momentum prediction tools. The sequence generation tool calls these tools, and will alert the user if the spacecraft momentum is predicted to exceed an upper limit, that will trip a fault and send the spacecraft to safe mode. The primary “knob” to turn to prevent exceeding momentum bounds is reaction wheel bias, which is another input to the tool. Frequently, keeping momentum buildup low will cause reaction wheel zero crossings. These crossings do not threaten spacecraft health and safety, but they may introduce a brief transient in the measured photometric timeseries by causing a transient pointing disturbance. The pointing disturbance can affect high precision photometric observations, so reaction wheel zero crossings are minimized by imposing a speed bias on the wheels [10]. Typically zero crossings are preferable to risking exceeding momentum bounds, since an entry to safe mode will stop the observation sequence all together.

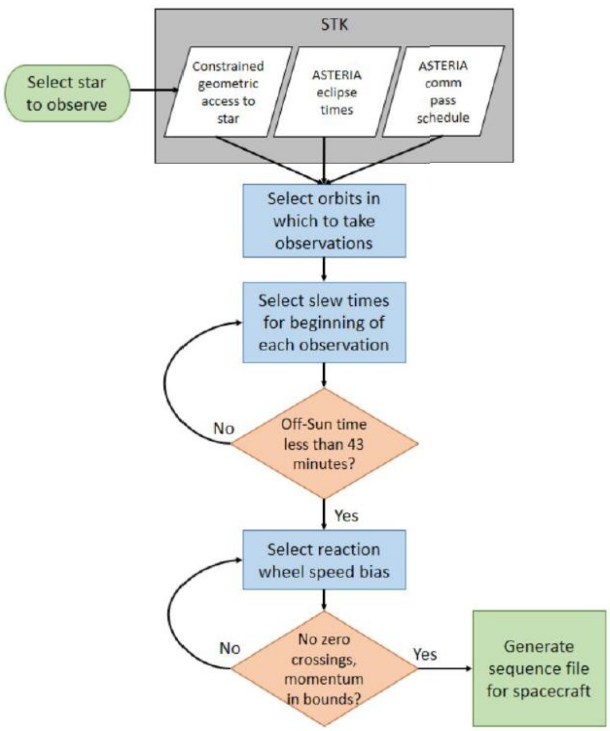

Fig. 10. Science observation planning process [9]

5.7 Future work

As extended missions continue, efforts to further improve the efficiency of operations have also continued. Specifically, automation of passes is in work to reduce the person hours needed to operate ASTERIA. This is primarily conducted by automation capabilities already incorporated into WTCCS, but some modifications are necessary to safely conduct un-manned passes on ASTERIA. For example, a script is in work that monitors key telemetry and will alert a team member if any telemetry appears off-nominal. The ultimate goal is to be able to operate ASTERIA in a “lights out” manner, with no one on console required unless a significant anomaly occurs.

6. Conclusions

On balance, and if done carefully, the benefits of combining the MAM role and an engineering role (e.g., fault protection) on a CubeSat mission outweigh the potential downsides due to conflict of interest. Combining the fault protection and mission assurance roles on ASTERIA did prevent the MAM from performing risk evaluation truly independent of the project. However, the insight gained by the MAM into the system design by adding these additional responsibilities was highly beneficial, and was able to be independently evaluated during monthly reporting by the MAM to JPL OSMS. The development and operations of ASTERIA required innovative tailoring of complex processes typically applied to larger missions, particularly in the areas of mission assurance and fault protection. A consistent approach to tailoring was applied throughout the entire project lifecycle. This approach was to mitigate risk by applying mission assurance at the highest risk areas, and documenting all residual risk in order to inform the institution of accepted risks and to inform the operations team of potential issues that may be encountered due to residual risk.

Some of this residual risk was addressed by fault protection. The simple fault protection design was most effective by identifying “catch-all” faults and applying simple responses that assert a safe state. The ability to modify some elements of the fault protection design is also important, as changing which responses were called by which fault monitors was key in protecting ASTERIA from a potential mission-ending fault identified in flight. The innovative tools developed by the ASTERIA operations team were critical for both MOAM and fault protection roles. Reduction of command errors and some flight rule checking was able to be incorporated in tools to improve mission operations assurance. Fault protection design changes were easily defined and implemented due to access to quick data analysis in OpenMCT.

The challenges and successes encountered in ASTERIA operations have demonstrated the importance of team members “wearing multiple hats” on a small CubeSat team, and having a deep knowledge of the system and tools that would be beneficial to analysing the system. For this reason, combining roles, while often not practical on a large-scale mission, is very useful on a CubeSat mission.

Acknowledgements

This work was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology.

The authors acknowledge the following individuals for their contributions to the ASTERIA mission:

Mary White for project management during the ASTERIA development phase.

Joel Krajewski for mentorship during development and operations of ASTERIA and project management for the checkout phase of ASTERIA operations.

Thomas Ramsey and Lori Risse for mentorship in all mission assurance roles and for continued line management support of ASTERIA through prime and extended missions.

Parviz Danesh and Jessica Clark for mentorship in the MAM and fault protection roles, respectively.

Julia Bell and Bruce Waggoner for mentorship in operations.

Sara Seager (MIT), principal investigator (advisory) and Brice-Olivier Demory (University of Bern) for providing the science case for ASTERIA and supporting the mission.

Maria de Soria Santacruz-Pich for defining environmental requirements and aiding in the verification and validation of these requirements.

Cody Colley for leading the integration and testing of the spacecraft and acting as Deputy Mission Manager for the initial spacecraft checkout in operations.

The core and extended teams that supported this project during development and operations, including Robert Bocchino, Alessandra Babuscia, Jessica Loveland, Jason Luu, Colin Smith, Len Day, Carl Felten, Janan Ferdosi, Kristine Fong, Harrison Herzog, Jim Hofman, David Kessler, Roger Klemm, Tejas Kulkarni, Jules Lee, Jason Munger, Lori Moore, Esha Murty, Chris Shelton, David Sternberg, Rob Sweet, Kerry Wahl, Jacqueline Weiler, Thomas Werne, and Shannon Zareh for the supporting this project during development and operations

The JPL line organization and technical mentors for the expertise they provided throughout the project. Sarah Gavit, who oversaw ASTERIA within the Engineering and Science Directorate at JPL, for her leadership and support during development and operations. Dan Coulter Coulter and Leslie Livesay in the JPL Astronomy and Physics Directorate for their support.

The operators, technical staff, and program management at Morehead State University, including Tobias Gedenk, Chloe Hart, Sarah Wilczewski, Alex Roberts, Bob Kroll, Michael Combs, and Benjamin Malphrus.

Appendix A (Examples of Outputs of Science Observation Generation Tool)

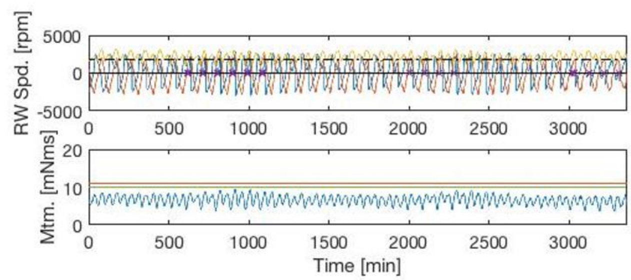

Fig. 11. Reaction wheel speed and momentum predictions output by the science observation generation tool.

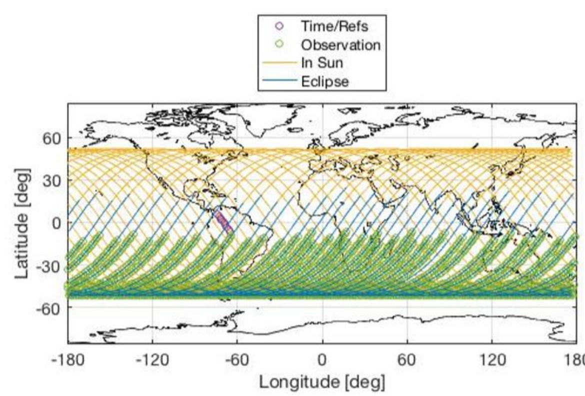

Fig. 12. Output of science observation tool, showing that the observations take place in eclipse (the green circles are along the blue line). This observation was of Alpha Centauri, which is visible during eclipse from the southern hemisphere.

References

[1] M.W. Smith, et al., ExoplanetSat: Detecting transiting exoplanets using a low-cost CubeSat platform, Proceedings of SPIE, 7731 (2010).
[2] C. M. Pong, et al., Achieving high-precision pointing on ExoplanetSat: Initial feasibility analysis, Proceedings of SPIE, 7731 (2010).
[3] C. M. Pong, et al., One-Arcsecond Line-of-Sight Pointing Control on ExoplanetSat, AAS 11-035, Guidance and Control, Vol. 141 Advances in the Astronautical Sciences, Univelt, Inc., Breckenridge, CO, February 2011.
[4] M. W. Smith, et al. The ExoplanetSat Mission to Detect Transiting Exoplanets with a CubeSat Space Telescope, Small Satellite Conference, AIAA/USU, Utah State University Research Foundation, Logan, UT, 8-11 August 2011.
[5] C. M. Pong, et al., High-Precision Pointing and Attitude Determination and Control on ExoplanetSat, AIAA Guidance, Navigation, and Control Conference, Minneapolis, MN, August 2012, pp. 1–24.
[6] M. Knapp, et al., ExoplanetSat: High Precision Photometry for Exoplanet Transit Detections in a 3U CubeSat, 63rd International Astronautical Congress, IAF, Naples, Italy, October 2012, pp. 1–12.
[7] C. M. Pong, et al., Three-Degree-of-Freedom Testing of Attitude Determination and Control Algorithms on ExoplanetSat, Guidance and Control 2014, Vol. 151 of Advances in the Astronautical Sciences, AAS, Univelt, Inc., Breckenridge, CO, February 2014, pp. 285–307.
[8] C. M. Pong, High-Precision Pointing and Attitude Estimation and Control Algorithms for Hardware-Constrained Spacecraft, Doctor of Science thesis, Massachusetts Institute of Technology, Cambridge, MA, May 2014.
[9] M. W. Smith, et al., On-Orbit Results and Lessons Learned from the ASTERIA Space Telescope Mission, SSC18-PI-34, 32nd Annual AIAA/USU Conference on Small Satellites, Logan, UT, 2018, 5-9 August 2018.
[10] C. M. Pong, On-Orbit Performance & Operation of the Attitude & Pointing Control Subsystems on ASTERIA, SSC18-PI-34, 32nd Annual AIAA/USU Conference on Small Satellites, Logan, UT, 2018, 5-9 August 2018.
[11] M. Knapp et al. (2018), *in preparation*.
[12] M. Gillion et al., Two massive rocky planets transiting a K-dwarf 6.5 parsecs away, Nature Astronomy, 1 (2017), 56.
[13] Demory, et al., Hubble Space Telescope search for the transit of the Earth-mass exoplanet α Centauri B b, Monthly Notices of the Royal Astronomical Society, 450 (2015) 2043-2051.
[14] R.L. Bocchino, et al. 2018. F Prime: An Open-Source Framework for Small-Scale Flight Software Systems, SSC-18-XII-04, 32nd Annual AIAA/USU Conference on Small Satellites, Logan, UT, 2018, 5-9 August 2018.
[15] B. Malphrus, M. Combs, J. Kruth, N. Fite, B. Twiggs, R. Schulze, A University-Based Ground Station: The 21 m Antenna at Morehead State University, AIAA SpaceOps Conference, Huntsville, Alabama, 25-30 April 2010.
[16] Alessandra Babuscia, Matthew Smith, Peter Di Pasquale, and Jim Taylor. ASTERIA Telecommunications. JPL Descanso Design and Performance Summary Series, 2018 (to appear).
[17] I. Heinrichsen and E. L. Wright, "The Mission Operations System for Wide-field Infrared Survey Explorer (WISE)", *Proc. SPIE*, Vol. 5270, pp. C-1 to C-9 (2006).
[18] J. Trimble, Reconfigurable Software for Mission Operations, AIAA SpaceOps Conference, Pasadena, California, 5-9 May 2014.
[19] Matthew Endsley and Colin Perceival, 2012, https://github.com/mendsley/bsdiff, (accessed 29.08.18).
[20] Colin Percival, Naive differences of executable code, 2003, http://www.daemonology.net/bsdiff/, (accessed 27.08.18).