SSC18-WKII-05

Defining a New Mission Assurance Philosophy for Small Satellites

Lee Jasper
Space Dynamics Laboratory
1851 Charlene Dr. Kirtland AFB, NM 87117; (435) 713-3400
Lee.jasper@sdl.usu.edu

Lauren Hunt, David Voss, Charlene Jacka
Air Force Research Laboratory
1851 Charlene Dr. Kirtland AFB, NM 87117

ABSTRACT

The concept of mission assurance was developed so that technical, implementation, and management practices could be enabled to increase mission success of otherwise irrecoverable spacecraft. Understanding and implementing the mission assurance trade space for small satellites is important to improve success rates, tackle more challenging missions while managing expectations, scope missions, and minimize oversight burden that inhibits innovation.

Small satellites generally selectively pick and choose, or completely ignore, the majority of the activities defined in Class A-D because constraints are an equally driving force and strongly compete with mission objectives. Further, government funded small satellite missions almost always fall under Class D, but they often create some tailored assurance profile that generally does not meet the intent of Class D, nor does Class D suffice for the realities of most small satellite missions.

This paper organizes assurance profiles into a structure that better represents the current status by accounting for the constraint – mission objective trade space of small satellites. The new infrastructure focuses on studied and implemented practices that produce successful missions. These practices include: a well-defined scope that balances constraints and objectives, significant time dedicated to testing at all levels, and lessons-learned-design principles.

WHY A NEW MISSION ASSURANCE DEFINITION?

Mission Assurance is a mature field for the space enterprise. The satellite community has invested heavily in understanding what constitutes ensuring mission success on-orbit, which has enabled unparalleled capabilities in space. Standards for parts traceability, environmental testing, and other practices and processes have been well researched and incorporated into the aerospace industrial base (e.g. Class A-D systems). This has resulted in operational systems lasting well beyond their required design life in many cases. These have also caused the perception that space systems should always work.

The emerging challenge over the last 10 years has been the tension between needing/wanting more from fielded technology at reduced costs, compounded with the expectation of more rapid technology refresh timelines. Terrestrially, this is a well understood phenomenon enabled by effects such as Moore's Law and manifested in cyclic product releases such as smartphone release schedules. In space, these effects are slowed because of factors such as technical challenges (i.e. radiation), programmatic challenges (i.e. the cost of launch) or cultural challenges (i.e. space systems are scoped around large platforms).

Contradictory expectations have emerged where there is a need for more capability in cheaper and faster timelines. There exists an underlying assumption that has evolved over the last number of decades that these systems cannot fail. While space systems are generally high reliability, failures have happened either due to launch or spacecraft issues (Figure 1). The last several decades of space systems have been in the range of 90% success rate (mission owner accepted outcome without considering it as a failure). With the advent of small satellites, missions have been approached with a much greater dynamic range of assurance practices and definition of success (often with success as simply being communications). It is often assumed that higher assurance practices yield higher mission success rates. Indeed, Figure 1 suggests that one product of accepting a wider variety of risk profiles might also be greater variation in mission success. Still, there is no direct correlation between higher risk and higher failure rate, nor does this explicitly show how mission risk profile (and associated assurance practices) relate to mission

success/failure. **This paper posits that the use, and potential acceptance, of this greater dynamic range allows for systems to increase mission and technology capability at reduced timelines and cost.** Not all small satellites need follow the constraint driven model, but instead follow the requirements-driven model. This paper focuses on the constraint approach.

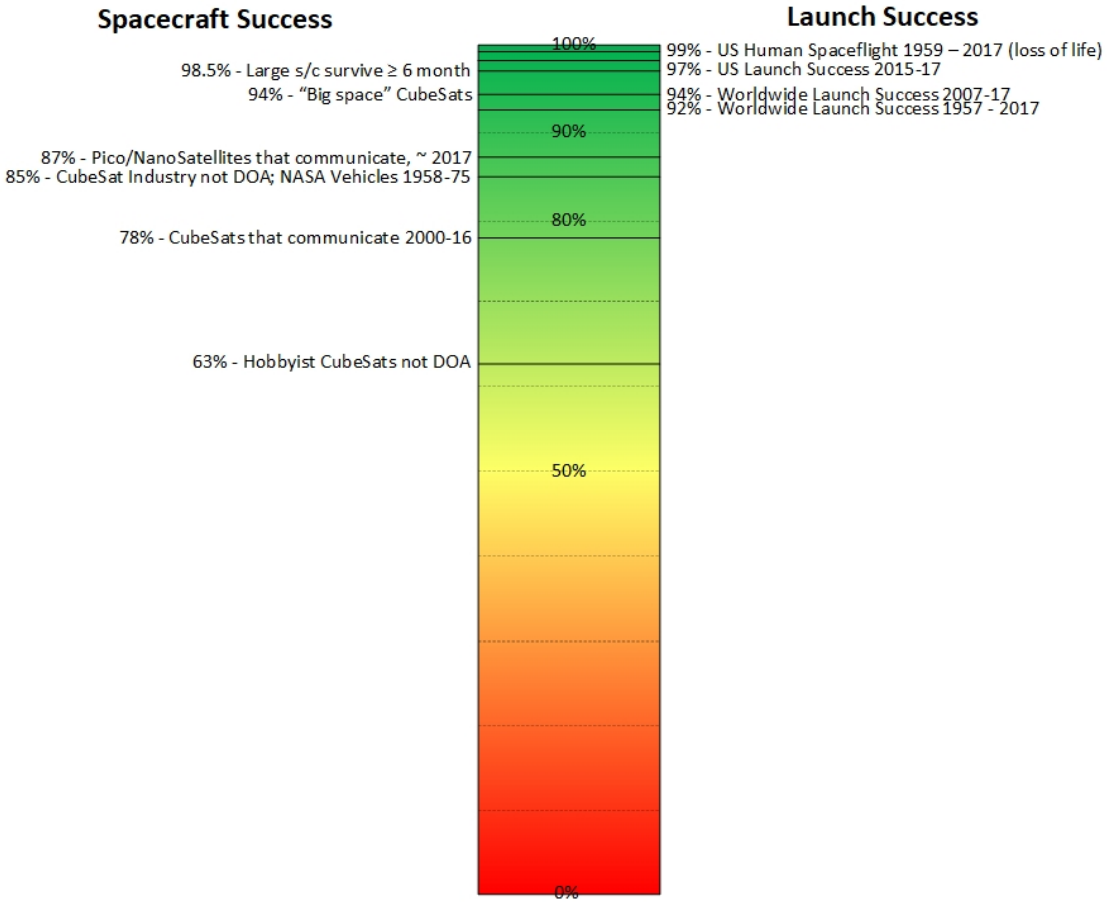

Figure 1: Percentage of Vehicle[1-5] and Launcher[6] Success (DOA = dead on arrival)

Small satellites have pushed major development in the philosophy and implementation of mission assurance. Originally the focus in small satellite assurance was on simply ignoring the standards all together or tailoring from traditional core standards such as MIL-HDBK-343[7] (now cancelled). In order to make small satellites relevant and useful as more than just educational tools, a balance must be struck between doing nothing and traditional standards.

It is becoming clear that programmatic constraints, combined with technical requirements, must be considered together to fully embrace the dynamic assurance/success range available to programs today. Small satellites take advantage of standardized launch, re-usable components (i.e. "commoditized" avionics), and aggressively scoping programs to fit within constraints allowing for better cost and schedule utilization. Intricately tied into this is a relaxing of traditional mission assurance methods.

Understanding how to better define the relaxation of traditional mission assurance is at the core of defining a new mission assurance paradigm. Cost and schedule are often the key driving factors; however, understanding what technical practices and processes should be leveraged in these riskier categories is important, along with implementation of a reduced approval authority structure. Having criteria for the classes enables programs to understand the risk posture for a program clearly and allows program managers to bound proper expectations for leadership and stakeholders. This is critical to helping prevent leadership from wanting a low risk (i.e. higher dollar, longer schedule, etc.) program, with the lower funding profile (and shorter schedule, etc.) of a high risk mission.

While many programs have expressed the desire to take advantage of the increased dynamic range, they lack either the technical knowledge or the programmatic authority, even though driving policy such as U.S. national priorities and the Space Enterprise Vision[8] encourage increasing capabilities at a faster pace. This paper attempts to define the present state of small satellite mission assurance, which does not conform to the old Class A-D paradigm (Figure 2), so that technical and programmatic practices can be more clearly understood.

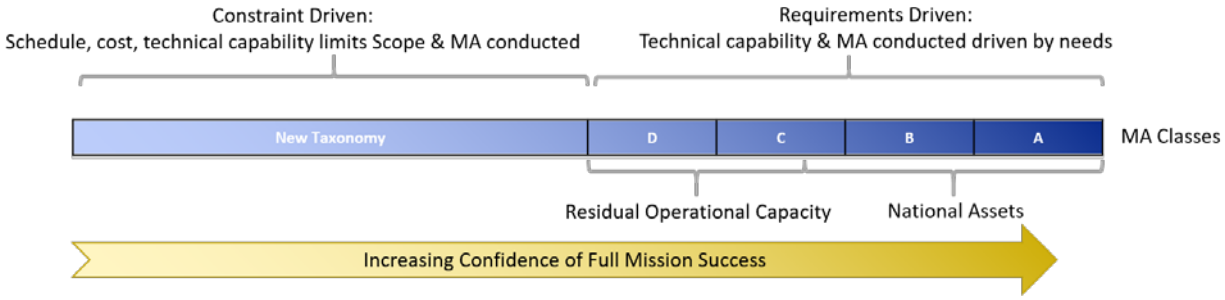

Figure 2: High-level concept for how new classes augment the existing mission assurance architecture

The SmallSat Gap: Non-Class A-D Missions

There is significant definition and literature on mission classification at NASA[9], the DOD[7], and across the aerospace industry[10]. As described in NPR 8705.4, the risk classification level should be defined and agreed upon by various parties. Class D has consistently been the bin into which small satellites have been placed. This has been done because Class D has elements that have technical risks that are medium by design and many credible mission failure mechanisms may exist[11].

It has constantly been shown that multiple requirements of Class D do not match the needs of many small satellites because they are too restrictive and do not allow mission-by-mission variability. Small satellites often ignore the majority of the activities defined in Class A-D because constraints are an equally driving force and strongly compete with mission objectives. With the state-of-the-art in the small satellite industry and funding these missions receive, small satellite producers often create tailored assurance profiles. The tailored profiles do not meet the intent of Class D, nor does Class D suffice for the realities of most small satellite missions.

DEFINITION OF A NEW SMALL SATELLITE MISSION ASSURANCE PHILOSOPHY

There are two key concepts that are essential to the new framework. First, and somewhat independent from mission assurance (MA), proper scoping of small satellite missions is **fundamental** to successful small satellites[3]. This has been expressed by a concept familiar to the small satellite community: constraint versus objective/requirement driven missions. Requirements driven missions (in a puritanical definition) keep their objective/requirements as-is and continue to design/refine technology until those objectives are met. In the constraint driven model, objectives/requirements are more fluid and need to bend much more as the true capability of, and constraints on, the system is understood. In essence, a constraint-driven mission fits within the capability "box", whereas an objective/requirements driven mission has a "box" made for it. A balance must be struck between stakeholders and engineers which recognizes bias towards constraint-driven scoping and holds to that posture once agreed upon. Without this, success is substantially harder and scope creep can drive up resource utilization without significant increase to the return of the mission. (In spacecraft like this, it can take time to understand the constraint space and sometimes re-scoping/reducing mission objectives is necessary to produce a more tangible mission given other constraints; this is a healthy and common outcome to keep within defined limitations.)

Second, good engineering is not replaced by mission assurance. Often, good engineering practice is contained within many MA practices, but here MA is seen as a check-and-balance to the engineering process. Further, **all missions are designed for full mission success; the amount of mission assurance can provide a level of confidence in mission success**. If a mission is not designed for full mission success, either the design is flawed or the scope is poorly formed. Conversely, a mission may have high fidelity design and proceed through well accepted engineering processes (i.e. the system engineering "V"), however the mission assurance profile may still follow low assurance. It should be emphasized that this does not mean the mission will not fully succeed, but that confidence in performance to achieve success is less characterized.

Figure 3 shows the proposed constraint driven branch for mission assurance. There are three major constraints considered in this new architecture and they are discussed in greater detail below.

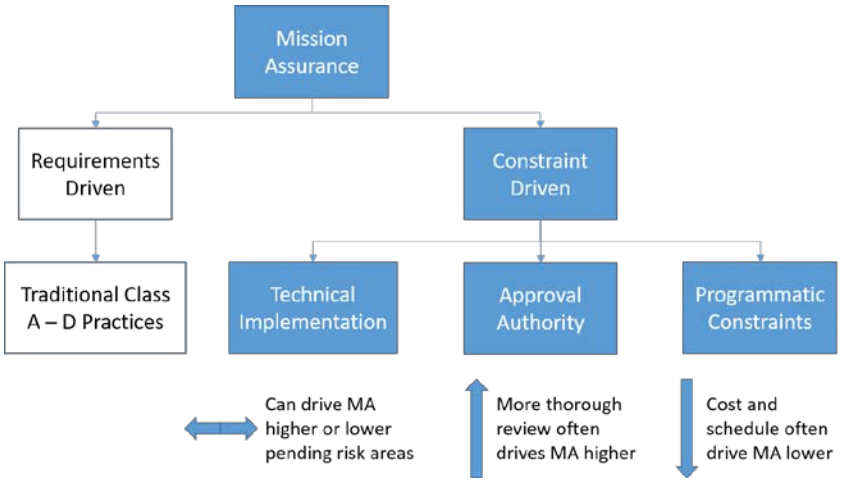

Figure 3: Architecture of constraint driven mission assurance

Approval Authority and Programmatic Implementation

Programmatic constraints (Figure 3) are often the most driving for many small satellite missions because they are expected to be relatively fast and cheap. Independent of expectations, the reality is that lower cost and shorter schedule programs often drive the mission assurance profile lower, i.e. to less characterized system performance. As programmatic constraints relax, more assurance practices can be implemented. A program's approval authority (AA) is considered the convening body that accepts/rejects risks, passes/fails a program at reviews, etc. Conversely to programmatic constraints, as a program's AA moves up an organization's structure, this often drives mission assurance higher. While technical practices might not improve, oversight and the amount of review does increase. Finally, the technical implementation can drive mission assurance both higher and lower. Should there be little new technology, higher state of integration, or lower system complexity, technical assurance may not need to be as rigorous, especially to counter balance any AA or programmatic constraints. However, challenging designs and missions can push some practices to being more rigorous.

Upon program/mission inception, the stakeholders/leadership and designers should have an honest conversation about whether the mission is requirements or constraint driven and understand the trade space the program is to operate within.

The following section focuses on technical implementation but ties some programmatic and approval constraints into a potential taxonomy for small satellites.

Technical Implementation

Currently, small satellite mission assurance practices seem to fall into four major categories as shown in Table 1. These categories consider the demonstrated level of functionality (i.e. mission assurance) increasing in scope from "Do No Harm" up to Full Success. While this is far from a true specification for how mission assurance should be performed, Table 1 seems to represent the present state of the industry and provides a framework for describing small satellites' assurance practices.

The taxonomy presented in Table 1 provides a means to discuss assurance based upon demonstrated capability with engineers and stakeholders. Using the level of demonstrated capability teams can provide a level of confidence, and credibility, for where designs are well vetted, and where they have had less characterization i.e. the demonstrated level-of-integration risk assessment.

Table 1: Small satellites' Program Risk Taxonomy with some example design and test practices

Demonstrated Level of Capability	Implication	Example Practices
Do No Harm	DOA is ok (education and/or fully constrained and not requirement driven)	Vibration testing, bake out, inhibit design review/test, range safety measures demonstrated, no RF transmission within 45 minutes of

Demonstrated Level of Capability	Implication	Example Practices
		deployment/no attitude maneuvers within 15 minutes, 25 year deorbit. Reviews: informal peer, launch readiness. Approval Authority (AA): Program
Survival	Not DOA (power + low-rate comm). May have no higher level functionality	(All of the above), possibly designing power/comm for tumble, long range communications testing with ground station has been completed(1), complete charge/discharge cycle testing completed(2), TVAC. Reviews: informal peer, may have stakeholder. AA: Program
Minimum Functionality	Min. Mission Success. Mission Recoverable in event of fault: Ex: LEOPS/start up Ex: Maintain Formation	(All of the above), full command execution test(3), startup/POR DitL testing(4), Sun-point test(5), other mission specific tests demonstrating survival functionality, mission specific FTA & Self-EMC test, thermal analysis. Reviews: informal-SCR, PDR, CDR, TRR, LRR. AA: Prog. +1 level
Nominal (payload performance driven by **constraints**)	Full Mission Success. Full Functionality	(All of the above), environmental characterization and flow down into requirements (i.e. radiation), full functional and limited performance testing, more detailed FTA & FMEA (flight, ground, GSE), SPF analysis/redundancy, requirement development to at least L2 and V&V. Reviews: formal-SCR, PDR, CDR, TRR, LRR. AA: Prog. +2 level
Nominal (payload performance driven by requirements)	Full Mission Success. Full Functionality	(All of the above), full functional and performance testing, Worst Case Analyses & design. NPR 8705.4, TOR-2011(8591)-21

Because each mission has its own challenges, the practices described in Table 1 are examples (driven by previous experience) but are not ubiquitous. Part of constraint-driven mission assurance has been to allow a building block application of the various practices (defined by References 9 and 10) as is most critical to reduce risk on a given mission. An example of this approach is a high power mission might focus on power and thermal parts derating, worst case analyses, etc., but focus less on vehicle FMECA, FTA, requirements validation, etc. Another example to move faster and incur less expense is to show more detailed systematic capability but not subsystem/component.

In Table 1 the lowest assurance in the taxonomy is "Do No Harm". This is, at its core, not concerned with the direct functionality of the vehicle and therefore dead on arrival (DOA) is possible and may be acceptable **relative to assuring the mission succeeds, and NOT the quality of the engineering behind the design**. Programs here focus on ensuring the small satellite does not harm the primary spacecraft, the other secondary spacecraft on the launch vehicle, and deorbits on time. This includes demonstrating proper power inhibit architecture, as well as bake-out testing and vibration testing. Review and oversight is left to the team or local program control even if funding is provided from outside.

The next level in the taxonomy focuses on gaining confidence that the vehicle is capable of surviving some or all vehicle configurations, including tumble (which all small satellites do), and therefore can communicate with the ground. This assurance posture does not focus on gaining confidence in whether the mission will complete but it does focus more on ensuring the vehicle will make it to orbit alive and is robust enough to survive unexpected and off-nominal cases. Essentially, "if the vehicle is alive, there is hope." Mission assurance practices focus more on ensuring the telecommunications and power systems are robust and that basic functionality of the main computer/software can support this. Review and oversight may include more stakeholder/customer interaction; however, overall go/no-go authority still resides with the program.

The following level transitions to improving confidence that the minimum mission is achieved. Again, the "minimum mission" definition is a scoping exercise, but if done well, very clear criteria can be made. Therefore, this style of mission assurance profile can be properly built. Here, not only does the mission do no harm and survive, but the design is demonstrated to have a recoverable mission even in off-nominal situations. This can require a significant amount of design and mission assurance applied in some cases, e.g. a formation where drift is a driver for the mission. Recovery time may need to be low to recover the formation, therefore assurance practices are leveraged to reduce failure modes and minimize outage. This level rolls in more design and testing practices from the traditional A-D classes as well as best practices (discussed below in the Best Practices section and in

Reference 3). Review and oversight starts to include more traditional review structures, albeit in an informal manner with authority to proceed not necessarily dictated by some/all the traditional reviews. Authority for the mission comes not from the direct program office but the next level higher.

Next, mission assurance practices are followed that provide confidence that the mission may meet full mission success. It should be emphasized that this is still a constraint driven (fitting in the box) architecture, therefore the mission assurance practices are also constraint driven. It is quite possible that this style of mission spends significant time in testing for full functional and performance testing of the payload(s) and vehicle instead of utilizing more analysis, higher reliability, etc. to demonstrate that the mission has a higher probability of full success. Review and oversight includes formal reviews (that can be tailored to the specific mission's needs) as key decision points/authority to proceed and these judgements are provided at the next higher level of the overall organization.

Beyond this point, the standard Class D - A profiles exist and are the transition point to capability driven missions. (For comparison Table 2 shows the Class D definition and how the new taxonomy generally is different.)

Table 2: SMA Related Requirements for NASA Class D[9] Related to New Taxonomy

	Class D	New Taxonomy
Single Point Failure	Critical SPFs (for Level 1 requirements) may be permitted but are mitigated by use of high reliability parts, additional testing, or by other means. Single string and selectively redundant design approaches may be used	Critical SPFs **are** permitted and **may be accepted** or mitigated by use of high reliability parts, additional testing, **selective redundancy, design robustness**, or by other means.
EM, FM, etc.	Limited engineering model and flight spare hardware	**Pending identified risks and risk profile, may utilize prototypes**, engineering models, and limited flight spare hardware
Qual/Test Program	Testing required only for verification of safety compliance and interface compatibility. Acceptance test program for critical performance parameters	Testing required for verification of safety and interfaces **but may expand all the way to full performance**
EEE Parts	Class A, Class B, or Class C requirements, and/or requirements per Center Parts Management Plan	Industrial COTS components through rad-hard parts acceptable pending lifetime-robustness-reliability risk trade
Reviews	Center level reviews with participation of all applicable directorates. May be delegated to Projects. Peer reviews of software requirements and code	As defined by mission profile. PDR, CDR, PSR are common reviews. Peer reviews are encouraged for all elements
Safety	Per all applicable NASA safety directives and standards	As defined by Program's organization (NASA, AFRL, university, etc.)
Materials	Requirements are based on applicable safety standards. Materials should be assessed for application and life limits	As defined by Program's organization and application. Meet Do No Harm
Reliability NPD 8720.1	Analysis requirements based on applicable safety requirements. Analysis of interface	As defined by Program's organization. There may be no specific reliability requirements
Fault Tree	Fault tree analysis required for safety critical functions	Often informal, conducted for ``do no harm" requirements, or only key elements
PRA (NPR 8705.5)	Safety only. Other discretionary applications	Not conducted / customer specific
Maintainability NPD 8720.1	Requirements based on applicable safety standards	As defined by Program's organization. There may be no specific requirements
Quality Assurance NPD 8730.5, 8735.2	Closed-loop problem reporting and corrective action, configuration management, GIDEP failure experience data and NASA Advisory process. Other requirements based on applicable safety standards	As defined by Program's organization. There may be no specific requirements
Software	Formal project software assurance insight	Often utilizes present-day software coding best practices and open-source development
Risk Management NPR 8000.4	Risk Management Program. Risk reporting to GPMC	Conducted for mission as integral part of constraint definition but may not follow NPR 8000.4. Lower level risks tracked informally or by Program's

	Class D	New Taxonomy
		requirements
Telem. Coverage for critical events	During all mission critical events to assure data is available for critical anomaly investigations to prevent future recurrence	Critical events apply infrequently to SmallSats. Best effort data collection

Note that while elements such as radiation/radiation hard parts and COTS parts qualification receive a lot of attention, these elements of mission assurance are often less cost effective than a multi-iteration development cycle that emphasizes testing. This is because a significant portion of small satellite hardware is untested or un-vetted, including payloads. Our experience has shown that a rapid development and test architecture produces more successful results for this class of spacecraft. Small satellites have a relatively high infant mortality risk that can be mitigated by rigorous testing to catch flaws prior to launch. However, if there are technical risks identified for a given design or mission, greater design assurance, reliability engineering, parts planning, etc. may be adopted to mitigate those risks. Robustness, i.e. the removal of single point failures and/or redundancy, is another way to mitigate risks. However this is difficult given mass and volume constraints and the complexities added to the system to handle increased robustness often introduce their own problems. Adding robustness through graceful degradation of systems is generally more successful for small satellites.

Case Studies

USE CASE 1: Transitioning from requirement to constraint driven assurance.

A new small satellite mission is conceived and expected to depart from Class D assurance requirements in several ways. Recognizing this, the team asks the question "why?" They identify numerous constraints such as allowable schedule, cost, available component maturity, launch availability and team experience. These constraints are then, in some cases, prioritized as equal to or more driving to the implementation than the science or technology mission objectives. Having greater clarity on identified constraints, the team allows the development and AI&T process to go forward in a "best effort" mindset with the given resources available.

While one approach could have been to seek waivers for Class D deviations, the team recognizes that given the number of deviations for the mission, it is unclear at what point the spirit and intent of Class D is lost and along with it the confidence that the mission would have succeeded to the same level indicated in Figure 1 or Figure 2. In the constraint driven construct, the small satellite mission team has taken advantage of the ability to identify the minimum level of activity needed to increase confidence and meet expectations in achieving varying degrees of spacecraft functionality below full mission success. In contrast, Classes A-D assume that nothing less than full mission success and functionality are acceptable.

The team and the wider organization see that the mission is a low-cost, in-house, component-demonstration experiment that is most likely to be cheaper to test, fly and re-fly than to design-analyze-test-fly once, assuming launch windows-of-opportunity (i.e. free to the mission program) are used. It would be a relatively low-profile mission within the organization. Therefore they accept a minimum mission success MA profile where verification and validation is mostly achieved through test rather than design and analysis.

USE CASE 2: Competing objectives/constraints influencing the mission assurance posture.

A university team is looking to build their second satellite. They have some practices and processes in place based on success with the previous project, but the team developing this particular satellite is mostly new. Approximately 80% of the team will turnover within two years. The number one priority of the professor overseeing the team is student education of system engineering fundamentals, but in an effort to fund said education she has partnered with Company X to fly their newest never-before-flown product and with Launch Provider Y on their first flight of their rocket. Company X wants to make sure this product has a successful on-orbit demonstration, as it will increase their ability to sell it to paying customers, but does not have the resources to fully finance their own mission and launch. Paying customers are not yet ready to take a chance on this unproven technology for their mission. Launch Provider Y believes in supporting the next generation of students and is happy to help where possible, but will not alter their schedule to accommodate this mission if the university team's schedule slips.

Use Case 2 is an example of the small satellite community's willingness to team, but also highlights the mixture of skill sets (i.e. students to professionals), expectations and constraints a combined team may face. Figure 4 shows how this particular mission settled on its assurance posture.

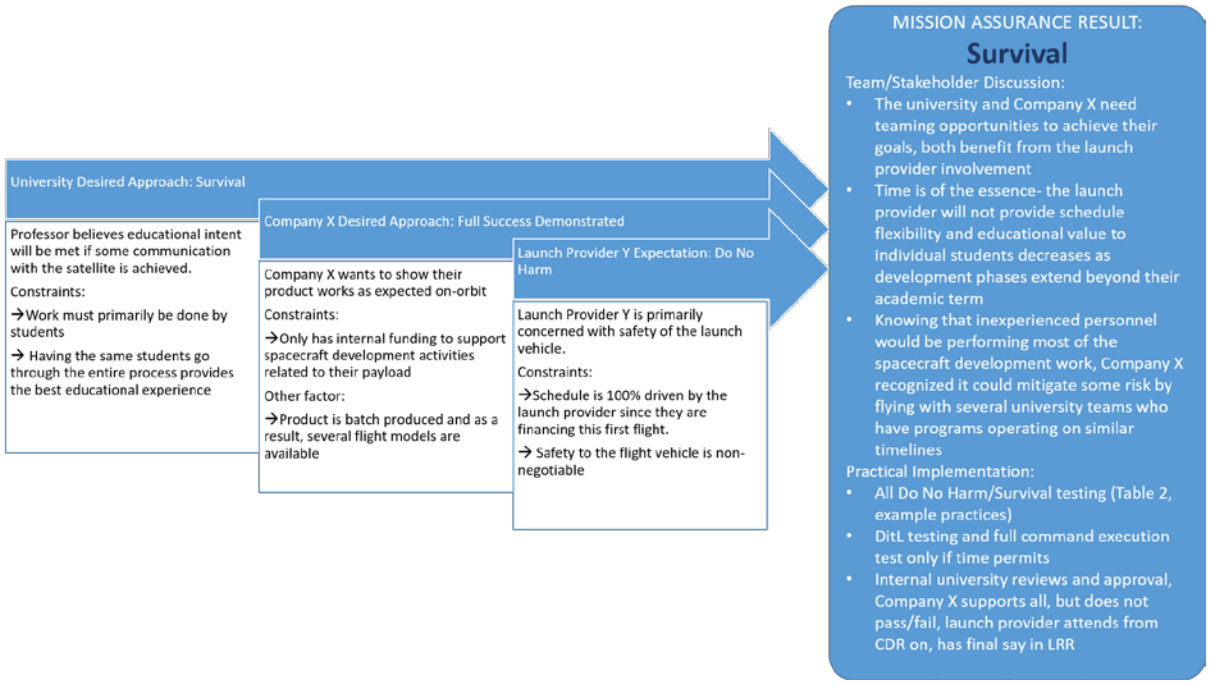

Figure 4: Use Case 2 negotiated mission assurance result

Each mission partner has different expectations of acceptable outcomes and in the discussion of what approach to take, schedule and educational constraints were prioritized over traditional expectations of mission success. With launch costs a non-trivial driver for both the university and company, both were incentivized to find a solution that would meet their constraints even if it meant compromising on ideal outcomes. Had Company X not been able to mitigate risk through additional missions with other universities, they would have had to make the choice of whether to have confidence in the vehicle/assurance up to survival or wait for a future opportunity more in-line with their expectations at the cost of delayed product release. If Company X decided not to proceed, the university professor would have to decide whether educational or technical de-scoping options were available to match the funding available or seek alternate partnerships which may risk the team's ability to meet the Launch Provider's non-negotiable schedule. This use case shows that by having the extended mission assurance definitions, small satellite teams are able to articulate a larger range of acceptable on-orbit performance driven by prioritized constraints.

USE CASE 3: Constellation versus individual satellite mission assurance.

Company X plans to field a constellation of satellites and is working through two funding phases. Phase 1 is a proof-of-concept demonstration for one satellite. The primary goal of the demonstration is to help settle some on-orbit performance considerations. A secondary goal is to help investors assess the prospects of future success. Phase 2 would field the final constellation and include as-needed design, AI&T and mission operations updates based on Phase 1 results. Time is of the essence for Company X: other companies are considering entering this market and investors want to see progress towards a return on their investment. In Phase 1, both investors and mission developers are willing to work with imperfect functionality as long as a path to planned functionality can be developed for Phase 2. In this phase, the time constraint may have a high priority and with future plans to put more satellites on-orbit, Company X might choose a lower mission assurance approach to realize near-term results and maintain momentum. Perhaps achieving survival of the first set of vehicles is sufficient to meet near-term goals. In Phase 2, to save money and time, Company X may look for a mixed mission assurance approach. They may ask themselves questions[1] such as:

- Can we spend a lot of time testing one space vehicle prior to launch to rule out systematic failures (i.e. Full Mission Success), but lessen

[1] The questions for fictional Use Case 3 are inspired/adapted from Ref. 12. The authors of this paper do not intend to imply that the authors of Ref. 12 have knowledge of or agree with this approach.

our testing requirements for the bulk of our satellites (i.e. Survival or Minimum Mission Success assurance profiles)?

- Can we phase our constellation production and launches such that we can improve later designs if issues are discovered on orbit (i.e. Full Mission Success assurance through on-orbit testing/demonstration instead of purely ground assurance practices)?

In these instances, each spacecraft would not necessarily be tested at the same levels to achieve high-certainty with respect to mission assurance, but solutions to achieving company goals within the context of their identified and prioritized constraints may still be possible.

USE CASE 4: Cost and schedule reduction by decreasing external oversight.

The above use cases demonstrate the balance between mission needs as well as technical and programmatic constraints. One other major element of the taxonomy from Table 1 is the ability to keep key decision points closer to the mission implementers.

Program X wants to demonstrate a new satellite data collection method given limited funding resources within just two or three years of starting the program. While good design practices are encouraged and high confidence in mission success is desired, the available funding and personnel availability for Program X is minimal. In the end, the needs of Program X are to fund mission(s) that provide the program the necessary data, not to follow and guide the spacecraft development.

Program X decides to develop the most risky elements (e.g. the detectors) through a more rigorous prototype development prior to building flight units; however, the rest of the hardware is accepted as small satellite COTS and does not undergo similar development. The managers of Program X follow a mission assurance profile of Full Mission Success for the detectors, but only require a survival profile for the vehicle assuming that previous design efforts have demonstrated adequate likelihood of success. Further, the managers of Program X only hold formal critical design reviews for the detectors and vehicles and allow for internal vendor practices on hardware development, thus reducing significant development time and cost.

Best Practices

Mission assurance, for many institutions, incorporates best practices into engineering efforts. While these are not close to exhaustive, these lessons have been taken from lessons learned across multiple small satellite builders, and also include the University Nanosatellite Program's lessons learned.

MAINTAIN SCOPE. Assuring mission success starts with a well-defined scope that pays serious attention to, and is bounded by, the capabilities of the various small satellite form factors that can be utilized. Generally accepting on-market capabilities while only driving 1-2 aspects of the design often greatly simplifies and adds mission assurance; again, constraint driven and not requirement driven.

DESIGN FOR TUMBLE. Every small satellite tumbles at some point in its mission. Designing the power system and communications system (near omni-directional antenna pattern, baud rate scaling) to be operational in the majority of potential tumble orientations greatly increases survivability in off-nominal cases.

CREATE AND VERIFY WELL BEHAVED SAFE MODE, RE-PROGRAMMABILITY. Safe mode should be simple and power positive in the tumble state. This safe mode should be well vetted and verified that it does as intended. Flight software, and subsystem software if possible, should be reprogrammable from the ground since small satellites have categorically had issues completing software validation and verification. Further, re-programmability has saved multiple spacecraft from being lost or not completing their missions.

DESIGN FOR FULL POWER RESETS ON HARDWARE. Since most small satellites accept hardware that has little radiation performance characterization, the simplest fix is to be able to reset hardware, especially the flight computer. Full power cycling must be possible without access to the flight computer (i.e. through watchdog reset, the radio and/or power system). Ideally, full power cycling (switching) of the subsystems is possible from the ground.

TESTING. It is imperative that ALL vehicles go through significant testing, starting early in development, because any incompatibilities and improper assumptions made during design are vetted by means of testing. Workmanship, design flaws, and software mistakes are all found through a rigorous testing campaign; these are the primary problems seen in small satellite systems. It is reasonable to dedicate about half of the overall schedule to testing of hardware. (Even in the most aggressively scheduled missions, about a third of the development time has been in testing.) Outside of common environmental testing and functional testing, SSP and the University Nanosatellite Program require the

following four tests for all their vehicles, with a fifth that is required for systems with attitude control. Note these are referenced in Table 1 with the various test numbers (1) to (5) in the Example Practices column.

1. Long Range Communications Test: verifies that the spacecraft can communicate with the ground station, at far field RF ranges. The radio is assembled in the structure during the test to account for the effects of the structure and other components.

2. Complete Charge-Discharge Cycle: includes draining the battery to its depth of discharge via spacecraft operations and then recharging the battery through the spacecraft's solar panels and regulators. The test demonstrates an autonomous recognition of when depth of discharge of the battery has been reached. This should be followed by an autonomous transition to a safe/non-discharging mode. The test should also demonstrate a charging of the battery, autonomously recognizing when the battery is fully charged, and autonomously ceasing charging demonstrating Peak Power Tracker/charging circuitry.

3. Command Execution Test: executes every command that will be sent to the spacecraft. Ensures that the commands work and do not put the spacecraft into any unknown error states. All commands should be sent to the spacecraft and an effort be made to observe the spacecraft's physical response to the command (meaning not only the successful transmission, but also execution). Depending upon MA thoroughness not every permutation of a command is tested, but every class of command should be verified. Further, all internal commanding of the flight computer to subsystems should be demonstrated (e.g. voltage and current thresholds on the power system for different operations modes). These should be tested in operational use cases if configuration changes from system modes. In some cases off nominal commanding should be executed.

4. Day in the Life: includes spacecraft initialization (i.e. spacecraft separation and turn-on scenario), executing modes and appropriate commands, as well as a turn-off command from the ground. Not every ground command needs to be executed in the DitL, but the DitL should go through every spacecraft mode and scenario. DitL should encompass any nominal commands as expressed in the most current version of the concept of operations document. Test initialization should simulate launch vehicle separation and run through commissioning and checkouts of the spacecraft, then through a full experiment plan for the mission. It should last at least 24 hours.

5. Sun Pointing Demonstration: includes polarity/direction testing of all sensors and actuators to ensure they are correctly assembled and mapped in flight software. A 1-D air bearing test is preferred to demonstrate, but not quantify, that the vehicle correctly tracks a bright light showing functionality of the sun tracking determination and control.

CONCLUSIONS

What this paper describes and begins to propose is a set of mission assurance profiles that expand the current language of mission assurance. This new architecture does not replace the old Class A-D as those are valid, even for small satellites, for some types of missions. Further, this new architecture is still to be defined and the taxonomy presented is not a complete or final representation. The approach is only proposed and has not been approved by the Air Force. However, it is proposed that:

- New mission assurance profiles need to be created that represent constraint driven mission sets.

- These new assurance profiles should heavily weight constraints as being equal-to, or greater-than, science or technology objectives.

- At mission conception, a clear scope and broad understanding of constraints help drive the implemented MA profiles to practices that have the greatest return-on-investment.

- Constraint based MA is driven by the technical, programmatic and approval authority/oversight environment. Generally more constrained missions allow decisions in all areas to be made closer to the project implementers.

As a final remark, while outside the scope of this paper, the small satellite community should consider if there is a minimum bar for the implementation of mission assurance, especially as space policies evolve to encompass the small satellite expansion.

Acknowledgments

The authors would like to thank several people and groups for their feedback in the development of this paper. The Reliability Working Group, part of NASA's S3VI initiative, Paul Oppenheimer at SDL, Mike Swartwout at Saint Louis University, and the SSP team, especially Kyle Kemble, Travis Willett-Gies and Dr. Jeff Ganley.

References

1. Swartwout M., "CubeSat Database," Saint Louis University, 2017. https://sites.google.com/a/slu.edu/swartwout/home/cubesat-database
2. Swartwout, M., "CubeSats and Mission Success" 2017 Update," Proceedings of the Electronics Technology Workshop, NASA/GSFC, June 2017.
3. Tolmasoff, M. and R.S. Delos and C. Venturini, "Improving Mission Success of CubeSats," Proceedings of the U.S. Space Program Mission Assurance Improvement Workshop, The Boeing Company, El Segundo, CA, June 2017.
4. Langer, M. and J. Bouwmeester, "Reliability of CubeSats-Statistical Data, Developers Beliefs and the Way Forward," Proceedings of the 30th Annual AIAA/USU Conference on Small Satellites, SSC16-X-2, Logan, UT, August 2016.
5. Newell, H.E., "Beyond the Atmosphere: Early Years of Space Science," NASA, 2018. https://history.nasa.gov/SP-4211/ch10-5.htm.
6. Kyle, E., "Space Launch Report," 2018, http://www.spacelaunchreport.com/logyear.html.
7. "Design, Construction, and Testing Requirements for one of a kind space equipment," SPVT-2016-005, ORIGINAL ED., DOD-HDBK-343. February 1986.
8. Public Affairs, AFSC, "Hyten announces Space Enterprise Vision", Peterson Air Force Base, CO, April 2016. http://www.af.mil/News/Article-Display/Article/719941/hyten-announces-space-enterprise-vision/
9. O'Connor, B., "Risk Classification for NASA Payloads", NASA Procedural Requirement 8705.4, June 14 2018.
10. Johnson-Roth, G., "Mission Assurance Guidelines for A-D Mission Risk Classes", Aerospace Corporation, TOR-2011(8591)-21, June 2011.
11. Leitner, J., "Risk Classification and Risk-based Safety and Mission Assurance", Goddard Spaceflight Center GSFC-E-DAA-TN19806, December 2014. https://ntrs.nasa.gov/search.jsp?R=20150001352
12. Zimmerman, R. and D. Doan and L. Leung and J. Mason and N. Parsons and K. Shahid, "Commissioning the World's Largest Satellite Constellation", SmallSat Conference, SSC17-X-03, Logan, UT, August 2017.