June 30, 2023

Lessons Learned with Risk Management: a Systems Engineer's perspective

Charles Baker
Landsat 9 Mission Systems Engineer
Code 599
NASA Goddard Space Flight Center
8800 Greenbelt Road
Greenbelt, MD 20771
Charles.baker@nasa.gov

David F. Everett
Instrument/Payload Systems Engineering Branch Head
Code 592
NASA Goddard Space Flight Center
8800 Greenbelt Road
Greenbelt, MD 20771
david.f.everett@nasa.gov

Kristina Pevear
DrAMS Verification and Validation Lead
Code 431
NASA Goddard Space Flight Center
8800 Greenbelt Road
Greenbelt, MD 20771
kristina.n.pevear@nasa.gov

Abstract:

Risk management is a communications device that enables systems engineering to effectively balance risk across the project. Developing and baselining risks is an essential continuous task to ensure top project concerns both from bottom up and top down are being mitigated. Risk management provides the opportunity to avoid the consequence of the risk when mitigation steps start early enough. Just discussing risk with all the project flight elements during development, even if no risks are open, provides an excellent communication opportunity between systems engineering and those elements, ensuring concerns and worries have a platform for discussion. A well-managed risk identification process will identify concerns that are serious but not being clearly communicated, and it will enable mitigation of those potential problems before they cause a failure.

Effective risk management requires considerable time and effort, but that effort will save time and money across the development. Risk management must be frequent enough to be useful and in depth enough to bring out emerging issues. It also requires a trusting relationship between the lead systems engineer and element and/or subsystem leads. The discussions need to be with the right number of individuals (typically a handful) and the right duration in time (typically an hour a month). Outside of these risk working groups, there is a formal management process to input, status, and disposition risks, and a monthly Risk Management Board meeting where key project stakeholders are informed.

This paper provides good guidance on effective risk management from a systems engineering perspective and provides project lessons learned from the NASA spaceflight missions NICER, Landsat 9, LRO, and OSIRIS-REx to demonstrate the effectiveness of risk management.

Introduction

Risk management is a phrase that fills many engineers with dread. Many engineers feel there is little value in doing risk management and believe it is simply a management white washing exercise of the real concerns. This feeling is usually based on experiences with poor risk management implementation in the past, often performed by those who don't understand the value or purpose of the process. When implemented poorly, risk management may be used only as an outward reporting tool, or as a method to apportion blame between teams within a project. Often, risk

contributors think the risk they have contributed is somehow better if it is more highly ranked. In these cases, risk management is may be perceived by the team as unnecessarily time-consuming and providing minimal benefit. When implemented well, the risk management process performs as a critical communications tool that supports all levels of the project and provides a way that to ensure risks don't fester. True project wide risk management addresses risks by distributing resources such as mass, power, funding or schedule for additional testing, hardware, or alternative procurement of critical path component or access to senior management to break an organizational log jam for a product lead.

Risk management, at its heart, is a process of communication. Not all concerns or worries are risks; it requires systems engineering expertise to help make the distinction. Often, the most valuable part of risk management are the conversations regarding concerns and worries, ensuring all teams have their needs heard and met.

Risk Management Nuts and Bolts

At the Goddard Space Flight Center, the risk management process is implemented at the lead Systems Engineering level (GSFC calls this the Mission Systems Engineer [MSE]; JPL may refer to this as Project Systems Engineering [PSE]). The risk management process is typically captured in a project Risk Management Plan, which defines risk terminology, roles and responsibilities, and process flow. The risks themselves must be presented to the Project Risk Board (typically chaired by the Project Manager) to open and actively disposition or mitigate. Success of risk management is highly dependent on the commitment of the MSE and Project Management to use this communication device to mitigate risk. No project proceeds without risk. Well-run projects focus on discovering risks and balancing them with other risks and available resources, thereby avoiding significant consequences. Even the process of writing a risk can often free a flight element to decide to proceed in a certain direction. Generally, few technical direction decisions can be made without incurring some risk. Documenting dissenting opinions in risk form can be a way of enabling the team to move forward and building in a monthly check on whether the technical decision was the right one.

The product of the risk management process is written concerns that can be presented, assessed for likelihood and consequence, with either a trigger, watch date or mitigation steps. The quantified risk can then be addressed or accepted by the project. Generally, risks are generated by product development leads with support from the MSE team, but it can be easily done the other way around with the leads reviewing the statement. It is a useful tool for systems engineers to compose a risk that might have a potential to become a risk and leave it unopened for a period to see if the potential risk's likelihood has indeed risen to a high enough level to open it (often as a result of a definitive piece of evidence that the concern is deviating from typical mission concerns called baseline risk).These draft risks will then be presented as part of the risk management formal meetings, where they may be officially opened. Open risks fall into the project management arena to actively report and press for mitigation.

One key to Risk Management is using a risk language formula that allows a broad group to understand what the risk is and what it might be in the future. At GSFC we use the "Given that/There is a possibility that/Resulting in" formula for risk statements. "Given that" is followed by a statement which captures the definitive piece of evidence that caused the concern or worry. "There is a possibility that" identifies potential issues, such as component failure, based on your "given that" statement. The last portion, "Resulting in", is used to state a specific overall impact, such as a defined cost, schedule delay, or missed critical requirement that relates the risk to a criteria by which the consequence can be judged objectively. Most risks have a multitude of likelihoods and consequences that could be attributed to them. There is generally a subjective sweet spot of likelihood and consequence that bounds every risk. We use the word subjective, as it is often the bigger discussion with project management and the risk owner that yields a mutually acceptable assessment. All risks need to be above "baseline" of similar missions at the same class. A good example of this is to consider whether a failure of the launch vehicle is a baseline risk or an elevated risk worthy of adding. A reliable launch vehicle that has launched similar type missions in the past would not be above baseline risk and therefore should not be a risk in the database, even though launch vehicle failure rates tend to be a bit above the minimum likelihood threshold of spacecraft risks. While a new, experimental launch vehicle or a launch vehicle with a recent launch failure has an increased likelihood of failure, and falls above baseline risk.

At GSFC, emphasis is placed on the differentiation between a risk, a problem, a concern, and a worry. Project personnel, including the flight element leads, should recognize that a risk is a *potential* issue (as opposed to something that has already gone wrong [a problem/issue]) that has a fact-based condition and consequence, and with

a likelihood that can be assigned. Often, initially, the likelihood starts at a lower level and can be elevated as more information becomes available. A project concern is typically vague and the likelihood cannot be assigned with confidence. A worry is based on indications that a project element may have an emerging problem but there is no clear indication yet that project resources will be required to address it. Only credible risks (sufficiently high likelihood with a measurable consequence), with the potential to require additional resources or impact schedule or mission performance or be a risk to safety will be opened and moved into active risks in the Project Risk Management Database (RMD) and be formally managed by the Project Risk Manager and Project Risk Management Board (RMB). Although active risks are captured in the RMD and dispositioned by the RMB, the risk owner (often the risk author, or a systems engineer on the team) is typically responsible for the mitigation plan or tracking changes against a trigger date if it is a risk with a watch designation. Further, these active risks have visibility to Project Management, and risk owners are easily able to communicate needs and resource requests to support mitigation activities. Risks remaining above "baseline risk" after all mitigations have been completed will be "accepted" by the project manager provided that he/she agrees that it is an acceptable risk from the project's risk profile. If an "accepted" risk has technical implications (impacts to critical mission requirements), these are retained and tracked as "residual" risks, which require review and approval of GSFC and Program management in addition to the Project Manager.

Without filtering, any risk management system has the potential to be overloaded if *every* potential issue were entered into the database. The system engineering team must filter out risks that pertain to the routine development effort and handled within currently allocated resources (i.e., within normal engineering practices). This empowers the flight element leads to manage risks that do not have the potential to require Project resources beyond those already allocated. This means on any particular project, there could be nested levels of risk management. Where each flight element has its own risk process, manager, and database. This paper is addressing only a risk process at the project level, but many of the principles could be applied at any level as long as it is effective and value added.

If the documentation implementor is unfamiliar with the tools or solicits verbose and unnecessarily complex statuses, engineers can feel the implementation cost is too excessive for the benefit. These implementation pitfalls are important to avoid as they devalue the risk process and distract from the legitimate risks in the project.

Another facet of the Risk Management (RM) philosophy is continuous forward-looking assessment. Due to the Risk process generally taking up to a month from Risk authoring to evaluation, risks that will be resolved/realized in a few weeks are not typically formally opened. In fact, once an adverse consequence of a risk has been realized (i.e. a deadline was missed, or a part failed during qualification), the risk becomes an issue (the likelihood is 100%). When considering potential issues (e.g., budget cuts, schedule compression, staffing shortages, unclear or changing requirements, product quality issues, qualification performance unknowns), Project personnel are encouraged to expect the unexpected, i.e., anticipate what could go wrong. When the Risk Manager and Lead Systems engineer conducts individual meetings with key team members each month, the Project members are encouraged to think in terms of:

- *What* could go wrong?
- *How will we know* something has gone wrong?
- *When will we know* that something has gone wrong?
- *What could* we do about it?
- *What will* we do about it?

These questions can help open up the conversation around risk the first few times the Risk Manager and Lead Systems Engineer meet with key team members. After a few months of periodic small group risk meetings, the conversation and focus will become clear to all involved. Dedicated meetings with risk discussion as the only agenda item openly solicit the individuals input and create a safe environment to openly discuss concerns.

The Lead System engineering should be mindful of risks that remain open for long periods of time. This is often a risk that month after month has unexecuted mitigation plans. This is often symptomatic of a project that is in implementation trouble (and is either poorly managed or has insufficient staffing) or a risk whose likelihood or

consequence has fallen to close to baseline risk. The key to identifying this is to read the monthly statuses and deduce whether the project is actively managing that particular risk, or simply neglecting it.

The intent of the risk process is for systems engineering and the project to expand their awareness of potential problems and anticipate what could go wrong. This philosophy pervades all aspects of the RM process, with constant reminders from the Risk Manager, Systems Engineering team and Project Management.

Difference between a baseline risk and a Risk that should be formally documented

Every project will have a "baseline risk" level, below which risks are not actively tracked. "Baseline risk" encompasses concerns that are common between all similar missions (i.e., launch vehicle failure), and concerns that are mitigated by default through the normal spaceflight development process (i.e. qualification of hardware mitigates the risks of operating in the space environment and over a mission lifetime). Without a specific cause to drive these concerns above the baseline level, the time and effort needed to formalize and track them would devalue the risk management process as it isn't possible to mitigate an unexpected failure mechanism during qualification.

A good example of a baseline risk is "Given that observatory mechanical qualification causes high loading on the flight elements, there is a possibility that, a portion of the flight element may experience a mechanical failure, resulting in schedule delay." This is a real risk, but one in common with almost all flown elements, so therefore it is a baseline risk that wouldn't need to be documented. If the structural analysis were to show negative Margins of Safety with launch loads on a specific component, this baseline risk would then become a specific, documentable risk. This is because: 1) the likelihood increased above baseline because the structural analysis showed negative Margins Of Safety; 2) because a specific component is implicated, the schedule and /or cost to recover can be roughly assessed and it is an above baseline consequence.

Documenting risk is certainly something that can easily be overdone-where risks are written about things that are common across similar missions or are about things that are simply work to go, or underdone-where risk conversations are too infrequent or too cursory to really get to the heart of concerns. This really gets to the heart of the "baseline risk" versus mission unique risk assessment.

Types of Risks

A mission unique risk assessment philosophy also recognizes that differing types of risk with varying potential consequences may lead to different handling strategies. These strategies should be prioritized and implemented in a balanced way that supports all Project objectives. The RM philosophy addresses three distinct aspects of risks: safety, mission performance, and project execution risks. Safety risk include the potential for personnel injury; this type of risk addresses hazards and other safety related risks. A Mission Performance risk (Technical Risk) is a risk with the potential to impact Flight/Ground segments during operations (i.e., preventing "end products" from performing their desired functions in their operational environments). This type of risk addresses inability to meet mission requirements, degraded science, loss of technical margin or system redundancy, or total loss of mission. A Project Execution risk (Programmatic Risk) is a risk with the potential to impact development activities or the ability to deliver the required product within the allocated budget, schedule and technical resources.

How to implement a risk management process at the Systems Engineering Level

Risk management execution at the systems engineering level starts a commitment to dedicate time and energy to strategically mitigate future project issues and allow development teams to take risk to move forward. Waiting to make every decision with zero risk will result in a project grinding down to a standstill. As stated earlier, the conversation between the programmatic team, technical team, and element lead is the most valuable part of the entire risk management process. At a mission systems engineering level, the risk process is a great way to level risk across the project. A monthly, dedicated small group meeting with each flight element enables good communication, and discussion of a worry list supports forward strategic planning. Developing these worry lists using the formulaic risk language allows qualitative evaluation with a quantitative review of potential impact. In addition to using standardized risk language, providing a clearly defined, mission-specific scale for consequences and likelihoods can help risk owners understand how to measure and articulate their concerns. Many risks start as unopened risks in the database as the likelihoods and consequences just aren't yet know with enough certainty to

establish them as above baseline risk. Having key element discussion monthly is a great way to keep track of progress and to assess whether any of the unopened risks have met the threshold of an active risk. These meetings also provide insight to concerns that don't rise to the level of a risk for an individual or system/subsystem, but should be considered for tracking at a project level when encountered across multiple systems (i.e. procurement delays). These conversations link back and support the goal of having Systems Engineering available and engaged in the project strategic thinking. Without these conversations at an informal level, schedule and therefore cost can be actively lost without systems engineers and project management being able to reallocate resources to mitigate that loss. The examples included in this paper show how valuable a properly executed risk management process can be.

Neutron Interior Composition ExploreR (NICER) Risk examples

NICER is an International Space Station (ISS) X-Ray telescope with tight timing performance and a large X-Ray collecting area. NICER is a Goddard Space Flight Center (GSFC) Class D mission launched in 2017. It has exceeded its 1.5 year lifetime and observed X-Rays emitted from Comets' tails, the earth's magnetopause, and black hole event horizons, in addition to its baseline mission of measuring neutron star masses and radii. NICER has had multiple science papers appear on the cover of the journal Nature and is the largest generator of scientific papers on the ISS.

Risk management was a key systems engineering tool used to maintain cost and schedule while building this Class D observatory. NICER was built within cost and available to launch per the schedule written in the proposal (though ISS transport vehicle wasn't available until a year later) despite being a fairly modest explorer mission of opportunity.

1st Example Risk from 2015: Given that Coupled Loads Analysis (CLA) is not required to be performed until 10 months before launch; There is a possibility that: analysis may reveal that some loads are higher than the Design Limit Loads (DLLs) to which NICER is designed, requiring redesign/rebuild and Resulting in: an increase in cost and schedule.

How we used the written risk to mitigate the concern: This is an extremely common risk with respect to launch vehicles and ISS payloads, partly due to the length of time required to get the launch vehicle provider under contract. In cases where the development life-cycle is short, a mismatch between the Design Limit Loads and loads determined by the CLA can drive re-design/re-build (if the loads are assessed prior to testing), or cause under-qualification/over-testing (in instances with hardware already past qualification testing). In this case, a CLA provided at 10 months prior to launch would have been delivered to NICER 6 months after CDR. By identifying the specific cost and schedule impacts of the risk, NICER was able to pressure the ISS Program to pay for an early CLA with the launch vehicle provider so that during our Payload Mechanical Qualification we had sufficiently high mechanical loads. Without this well-defined, quantitatively evaluated risk, it would have been more difficult to get the ISS to change their process. This also benefited NICER when the originally planned companion payload was de-manifested, and an additional representative CLA was needed to define accurate load cases. Communication of this risk supported successful coordination and information exchange with the launch vehicle provider to quantify reduced load sets outside of the standard CLA development process, defining random vibe loads compatible with the NICER X-Ray Timing Instrument (XTI). Without the culture of risk management as a communications tool, NICER would have needed to continue on a path of testing to inaccurate and overly stringent loads, with the potential for hardware damage and an inability to meet mission requirements.

2nd Example Risk from 2015: Given that NICER Detector System or Star Tracker Flight Software could fail to be updated on-orbit and corrupted; There is a possibility that: The NICER Payload functionality would be inhibited Resulting in: Science degradation.

How we used the written risk to mitigate the concern: This risk came about as we considered how we might update Flight Software after launch. We wanted to ensure that if the Flight Software (FSW) in the Detector System and Star Tracker were corrupted due to a memory or radiation event, we could upload a new version to fix the errors. This ultimately led to a boot up code review for each of these components. During Payload level testing, we also demonstrated the ability to update FSW through our ground system prior to launch.

1st Example Residual Risk post-NICER launch: Given that: NICER will be in its stowed orientation on the ground for almost a year; There is a possibility that: the required torque margins could be lower on-orbit initially as the actuator lubricant creeps in a 1-g gravity field, Resulting in NICER failing to deploy on-orbit.

Why this is a residual risk: As was mentioned in the introduction section for NICER, NICER had to wait about a year to be launched after delivery to Kennedy Space Center. We were unable to move the complicated three axis pointing system without disassembling the Payload post qualification. We had a risk-risk trade in this case: there is a risk without movement for a year that the lubricant in the stepper motor actuators would pool and make the first movements of the actuators more difficult or seize entirely versus doing significant structural mechanical disassembly after mechanical qualification. We determined the workmanship risk for performing a disassembly post mechanical qualification and the risk of trying to move the articulated arm with Mechanical Ground Support Equipment was much more likely to cause damage to NICER versus the possibilities the actuators would be seized by pooled lubricant. Hence, we had this residual risk.

2nd Example Residual Risk post NICER launch: Given that: The Soyuz vehicles contain a gamma-ray source that generates direct and indirect radiation detectable by NICER; There is a possibility that: The resulting background count rate, indistinguishable from neutron star X-rays, may significantly impact NICER's sensitivity, Resulting in extended mission life to meet science requirements.

Why this is a residual risk: we were unable to get any information about the intensity of the Soyuz vehicle gamma ray source level prior to NICER's launch. As an instrument sensitive to background radiation, we mitigated this risk as much as feasible by adding radiation shielding, but still felt that there was some residual small likelihood of a risk that the gamma ray source might impact NICER's science. Hence we maintained this as a residual risk.

<u>Landsat 9 Risk examples</u>

Landsat 9 was launched on September 27, 2021. Landsat 9 is a class B mission. The Landsat program is the longest running orbital earth observational program. It images the Earth in spectral bands of scientific and agricultural interest. The history of the observable natural and manmade world can be seen by looking through Landsat's archives. On the day of launch BBC declared Landsat 9: "What is arguably the world's most important satellite. No other remote-sensing system has kept a longer, continuous record of the changing state of our planet." The successful development and commissioning of Landsat 9 was one of GSFC top priorities in 2021 through 2022. Maintaining a healthy risk process was key to that success.

1st Example Risk from 2020: Given that: Landsat 9 is going to conduct their observatory-level thermal vacuum test in the midst of the Global COVID 19 pandemic; There is a possibility that: key personnel will not be able to support the test onsite and may miss a performance or functional anomaly that could result in damage to Landsat 9, Resulting in: schedule loss to recover.

How we used the written risk to mitigate the concern: extensive conversations were conducted within the team including the prime contractor and instrument providers about how best to mitigate this significantly likely risk. Multiple thermal vacuum planning sessions that focused on the unique remote support were conducted team wide. We developed unique Webex enabled remote terminals that allowed remote support to view key telemetry pages and trending graphs. We tasked on-site support to communicate clearly with remote support what was happening within the control room. We developed protocols and console spaces that protected the on-site workers. We paused after each test phase was completed and ensured all the technical stakeholders had been able to review the telemetry and performance data to ensure no anomalous behavior was observed in the previous phase. The Observatory T-Vac test was successfully completed and Landsat was enabled to ship to the launch site.

2nd Example Risk from 2020: Given that the common JPSS/Landsat9 Solar Array Drive Assembly design Life Test Unit failed during Life Test and the FRB concluded the failure was due to inadequate harmonic drive lubrication, there is a possibility that the Landsat 9 unit will fail prematurely or if replaced fail due inadequate post-repair qualification testing, Resulting in shorter mission life of Landsat 9.

How we used the written risk to mitigate the concern: the life test failure itself occurred before Observatory level testing of Landsat 9. The life test failure occurred after the nominal 2 times lifetime for Landsat 9 had been

completed (JPSS has a longer baselined mission lifetime). But as the investigation continued, it became apparent the under lubrication of the actuator would induce an unknown limited life of the Landsat 9 Solar Array Drive Assembly (SADA). Once we decided that we needed to refurbish our SADA, we noticed that a refurbished unit would arrive after our scheduled Observatory T-Vac test. We performed a risk-risk study, looking at the risk to an inadequate Observatory T-Vac test without the flight SADA versus the risk of delaying the test to wait for the SADA. We determined that the SADA wasn't in flight or thermal configuration anyway during the Observatory testing and therefore the risk was low to do the test without the flight SADA. We completed Observatory T-Vac and then installed the flight SADA after all regression testing was completed on the SADA.

Lunar Reconnaissance Orbiter Risk examples

The Lunar Reconnaissance Orbiter (LRO) was originally conceived in 2004, as the first in a series of missions to the moon as part of a new Exploration Initiative at NASA. LRO is a Class C mission. The instruments were selected through an Announcement of Opportunity, and the mission was assigned to GSFC for project management and for an in-house build of the spacecraft. The Exploration organization at HQ wanted to launch the mission by the end of 2008, and the in-house build provided a way to meet that schedule. Funding started in early 2005, the mission was confirmed in May of 2006, and the team maintained schedule until the summer of 2008, when national launch priorities delayed the launch into 2009. Further issues with launches ahead of LRO delayed the launch until June 2009. Prior to the launch delays, schedule demands dominated the development. As problems arose during the design and build of the mission, it was critical that the MSE quickly assess those problems for system-level impact and resolution. As you will note below, at the time risk statements lacked the Given That/There is a possibility That/Resulting in form and simply had an If/Then format.

1st Example Risk from August 2007: If the flight power converters for the C&DH do not arrive on time, then the flight C&DH will not be delivered on time, delaying the launch. Context: The flight power converters were very late and the vendor was not providing a delivery date. The C&DH was needed for system-level testing.

How we used the written risk to mitigate the concern: When the C&DH lead mentioned this concern during a regular one-on-one risk meeting, he thought the project was aware of his concern, since he had mentioned it to the project's parts expert. Needless to say, the project manager wasn't closely tracking every part delivery on the project, but it was clear to the systems engineer at this meeting that this part could end up being a big problem. The project manager quickly elevated the risk associated with the part delivery to GSFC center management, asking for help. The center director contacted the company to ask about the part status and apply some urgency to the delivery. In parallel, the project started the process of up-screening some engineering parts which were already in hand. The flight spare power-supply board was built up using one of these lower-quality engineering parts, with the intent to use this board in the C&DH to get testing started. If the flight-quality parts didn't show up, we would consider flying the lower-quality equivalent after a careful risk assessment. Building up the spare card bought time—the C&DH team integrated the box and worked through the various issues that pop up during integration. The flight parts did arrive, but a few months late. The project used an engineering C&DH to start system-level integration, with the flight C&DH showing up shortly after the system-level testing started. The mitigations put in place lessened the impact of a risk that turned into a problem. The system-level integration went forward with very little impact to the mission schedule. This example highlights not only the strategies for and benefits of risk mitigation but also the advantage of regular one-on-one discussions with team members in order to identify risks.

2nd Example Risk from 2006: "If: A failure or error occurs during the critical Lunar Orbit Insertion (LOI) phase of the mission, then loss of the mission may occur."

How we used the written risk to mitigate the concern: This risk was identified early in the development of the mission. The LOI was a critical, one-time event that needed to occur on time, with the burn in the correct direction for the correct duration. LRO utilized a direct trajectory to the moon, so the LOI would occur just 5 days after launch. If the burn was missed, getting back to the moon was somewhere between unlikely and impossible. The project assigned a dedicated phase lead, who was also the Deputy Mission Systems Engineer, to track all aspects of LOI. In addition to working with the flight hardware and software engineers, the LOI lead worked with the reliability and operations teams, assessing the various failures that could prevent successful completion of LOI. The trajectory analysis and burn planning would happen on the ground during the trip to the moon, so the ground system was intimately tied to this risk. The system was designed with four large thrusters, only two of which were needed

to capture into orbit around the moon (in addition to the attitude-control thrusters); but to best take advantage of this, the planners would need to know in advance if all thrusters were working. So the team added an engineering burn a day before LOI to confirm performance of the system. The LOI lead challenged the Flight Dynamics team to identify trajectory options to further reduce the risk, including the investigation of a strategy to accommodate a missed burn, returning to the moon at a later time. The final LOI burn design only required that the team execute half of the planned change in velocity (delta V). This allowed for failure of half the thrusters or an early end to the burn while still successfully capturing into lunar orbit. The trajectory also included a swing-by option that would allow return to the moon. All of this advanced planning established clear decision points for the team during the burn.

The other big risk mitigation was careful planning of the operations, including ground-in-the-loop contingencies. Because of the short signal delay between the moon and Earth, real-time commanding in contingency situations is feasible at the moon. The LRO team decided to be prepared to handle unexpected issues by thoroughly planning and training for contingencies that included unexpected situations that might require real-time troubleshooting and commanding. As stated above, only half the LOI burn needed to execute to achieve lunar orbit, and once in orbit, the team would have time to work through problems to get to the final mission orbit. So the planning for LOI focused on identifying the point where sufficient delta V had been achieved, planning for real-time restarts, thruster failures, and other anomalies, and knowing when to stop after achieving orbit. An important note here is that LRO is largely a single-string spacecraft, so the set of survivable anomalies was relatively small. Nominally, an on-board sequence would execute and the ground would just monitor. The engineering team was fully staffed for LOI, with a couple dozen people monitoring the system performance, and this team practiced LOI and contingencies many times prior to launch. After all of the ground simulations were complete, the team accepted the remaining risk of an unsuccessful LOI. It turned out that LOI executed as planned, and the risk was retired. Even though this risk did not manifest as a problem, all of the preparation gave the team confidence that they were ready for any survivable anomaly.

<u>OSIRIS-REx Risk examples</u>

The OSIRIS-REx spacecraft launched in September 8, 2016. OSIRIS-REx is a Class B mission. It entered the asteroid Bennu's vicinity in 2018. Bennu is a carbonaceous asteroid whose regolith may record the earliest history of our solar system. It successfully retrieved a sample of Bennu on October 20, 2020. Bennu may contain the molecular precursors to the origin of life and the Earth's oceans. Bennu is also one of the most potentially hazardous asteroids, as it has a relatively high probability of impacting the Earth late in the 22nd century. OSIRIS-REx will determine Bennu's physical and chemical properties, which will be critical to know in the event of an impact mitigation mission. Finally, asteroids like Bennu contain natural resources such as water, organics, and precious metals. In the future, these asteroids may one day fuel the exploration of the solar system by robotic and crewed spacecraft. Note this project also pre-dated the later Given That/There is the Possibility That/Resulting in language. You can see how the language evolved from LRO to OSIRIS-REx.

For context on the first Risk, the Touch And Go operation was the mission critical operation to acquire the sample from the surface of the asteroid and also was an operation that put the flight element spacecraft at risk.

1st Example Risk from February 2012: "If: Ground performance and life testing of GNC Lidar is insufficient to uncover latent defects in design or manufacturing, Then: There may be technical impacts related to reliability on-orbit affecting [the mission critical] Touch And Go (TAG) success." Context: A small company with limited experience was selected to build the lidar. The design was based on a lidar that only needed to operate in low-Earth orbit for a short period of time. There was question about the maturity of the technology the company planned to fly. Related risk: "If the GNC lidar delivers late, the launch date could slip." Because of potential technology development issues, the lidar delivery could stretch out.

How we used the written risk to mitigate the concern:

The team identified the first risk early in the development, right after the lidar vendor was selected. At the time, there were few choices for lidars which met the performance and cost requirements of the mission. The closest competitor to the vendor chosen was significantly more expensive. So management decided that the team could

compensate for some immaturity through additional risk mitigation. As the vendor worked through development issues and delivery stretched out, the schedule risk was identified. Both of these risks were coupled to the technical development of the lidar, so initial mitigations focused on retirement of the biggest risks associated with the lidar development. A GSFC lidar expert was brought on board to work with the vendor, helping their team identify technology development items and creating a technology development plan, which included early risk-reduction testing. NASA's process for new technology involves identifying the Technology Readiness Level, and working through risk-reduction testing to raise that level appropriately, to TRL-6 by Preliminary Design Review (PDR). *There is a good reference that can be used for a definition of NASA's TRL levels called "Final Report of the NASA Technology Readiness Assessment (TRA) Study Team from March 2016.* Several components of the lidar were identified as less than TRL-6, and project management worked with the vendor to establish a plan to build and test these key components before mission PDR. During this testing, several failures occurred, and the vendor updated the design appropriately. The risk-reduction testing did its job, identifying problems with the design early enough to avoid a large schedule impact. But the schedule kept eroding, and the project was still concerned that the entire mission success relied on this unproven lidar.

The Project Systems Engineer, working in concert with the Project Manager and Principal Investigator, had created a strong risk management culture on OSIRIS-REx. This culture included the early identification of risks through regular one-on-one meetings with key element leads, appropriate funding of risk mitigation strategies, and careful risk assessment when making project-level decisions. By the time the team approached the Critical Design Review (CDR), risk-reduction spending had led to an increase in project reserves, but the lidar risks were still looming large. Engineering had identified an option to use cameras instead of the lidar to navigate to the surface of the asteroid for sample collection, but this capability would require significant software development. It also required additional cameras which weren't part of the baseline. Project management decided to add the cameras just ahead of CDR, but hold off developing the software. The thought was that the project could implement the software in flight, should both lidars fail (OSIRIS-REx flew two copies of the same lidar). The decision to implement software pre-launch could be delayed until after CDR, but the risk of that development would grow the longer that decision was delayed. The lidar team continued to encounter issues after mission CDR, so the mission schedule risk started to grow more likely. It was at this point that the project manager decided to implement the Natural Feature Tracking (NFT) flight software, which could completely replace the need for the lidar. For the next year and a half, the project developed the lidars and the NFT software in parallel. Project management committed to a full flight qualification of NFT so that the project could move forward without the lidars if necessary, but the lidar development continued in case NFT ran into some issue. At the time of the Pre-Ship Review (PSR), the flight lidars were not yet on the spacecraft because of all of the development issues. But because of the fully-implemented and qualified status of NFT, the project was able to hold schedule and ship to the launch site, where both lidars were integrated just 3 months before launch. The $10 million spent to mitigate the lidar risks had saved a launch slip that would have cost on the order of $17 million. Upon arrival at Bennu, the team discovered the asteroid to be much rockier than expected. The lidar navigation, which was still baseline, did not have the accuracy needed to dodge the rocks and get to a safe place to sample, so the project ended up using NFT to navigate. The sample was collected from a location that was within 30 cm of the targeted location! The risk-management culture of OSIRIS-REx enabled the project to reach launch having spent $43 million LESS than planned, and operation at the asteroid was MORE robust than the original requirements.

Summary

Risk Management is an important tool that helps to identify, mitigate, and sometimes accept unique risks during development of missions. Effective Risk Management has helped to enable key NASA missions of various classes to be successfully launched and meet their mission objectives, while remaining within cost and schedule constraints. At the mission systems level, it is one of the key tools to help identify previously unknown concerns, quantify their likelihood and consequence, and then develop a mitigation plan. Without this tool, projects will experience an increased number of problems, an erosion of schedule, and an increase in cost during development and additional problems post launch that can jeopardize their mission. Understanding how to optimize the value of risk management and minimize the formal process effort is what every project strives to do. Examples from NICER, Landsat 9, LRO, and OSIRIS-REx will hopefully aid in that endeavor.