

# Lessons learned from the Kepler mission and space telescope management

James Fanson \*<sup>a</sup>

<sup>a</sup> Jet Propulsion Laboratory, California Institute of Technology,  
4800 Oak Grove Drive, Pasadena, CA 91109

## ABSTRACT

This paper presents lessons learned over the course of several space telescope mission and instrument developments spanning two decades. These projects involved astronomical telescopes developed by the National Aeronautics and Space Administration (NASA) and were designed to further our understanding of the Universe. It is hoped that the lessons drawn from these experiences may be of use to future mission developers.

**Keywords:** Telescope, space, lessons, Kepler, Spitzer, SIRTf, Hubble, GALEX, Keck, TPF, WFPC2

## 1. INTRODUCTION

Over the period of two decades the author has been involved in a management capacity in the development of several space telescope instruments and missions. The lessons outlined in this paper are drawn from a range of experiences: from projects that remain in early formulation to those that have completed their primary missions; from small missions to large facility class observatories; from competed PI-mode to assigned or directed implementations; from an inauspicious “under the radar” development to one at the center of public scrutiny; and from the “faster-better-cheaper” to the “mission success first” paradigm. Specifically, the missions include: the Wide Field and Planetary Camera-II (WFPC2) instrument, completed in earnest and flown in 1993 to recover the optical performance of the Hubble Space Telescope; the Spitzer Space Telescope (called the Space Infrared Telescope Facility or SIRTf during development), the fourth of NASA’s “Great Observatories,” flown in 2003; the Galaxy Evolution Explorer (GALEX), the first wide area extragalactic ultraviolet survey mission, flown in 2003; the Terrestrial Planet Finder, a large telescope concept intended to image and characterize earth-size planets orbiting the nearest stars; and Kepler, NASA’s first mission capable of detecting earth-size planets orbiting in the habitable zone around stars other than the sun, flown in 2009.

## 2. THE LESSONS

It is necessarily the case that everyone’s experience is unique. There is no single path to success or failure. Lessons are derived from the result of specific encounters with circumstances that are unlikely to be duplicated again in detail. Nevertheless, eventually patterns emerge from the common threads drawn from experience and woven in the loom of time.

### 2.1 Programmatic aspects

*Be prepared for a changing implementation environment.* Scientific space telescopes are, by and large, the domain of the federal government. As a federal agency NASA operates in the turbulent waters of national politics, subject to changing priorities of the Executive branch and Congress, the vicissitudes of yearly budget cycles, and the impact of decisions by other government entities. NASA does its best to bring order and structure to the project implementation environment through the use of decadal committees to formulate long term science priorities, and the promulgation of procedural requirements such as NPG7120.5 to standardize processes. Nevertheless, over the course of the several years typically necessary to develop and operate a space mission, it is likely that substantial changes will occur in any number of areas that can affect the implementation. Examples include:

---

\* James.L.Fanson@jpl.nasa.gov; phone 1 818 354-7014; fax 1 818 354-4950; jpl.nasa.gov

- Budget rescissions due to changing priorities within the agency, or a continuing resolution from Congress in lieu of an approved federal budget.
- Changing enforcement standards for export control regulations or information technology security requirements.
- Substantive changes in NASA process requirements, as often occurs when a new revision of 7120.5 is issued.
- Sea change in risk acceptance following a major mission loss, such as the NASA Integrated Action Team (NIAT) recommendations that ended the faster-better-cheaper implementation paradigm, or the Columbia Accident Investigation Board (CAIB) report that led to the creation of the Independent Engineering Technical Authority.

Such events require revising the forward implementation plan and estimating the impact of the change on budget and schedule—something notoriously difficult to do on the fly. The project may or may not be provided additional resources to compensate for such changes.

*Avoid situations where multiple NASA Centers are vying for control.* Several NASA Centers have a past history of robotic spacecraft development. Each has its own culture and way of doing business. They are all successful. But it must be the case that the Project Management Center has top level management and engineering technical authority for the implementation. It is possible for one Center to deliver instruments or subsystems to another or for one to act as Project Management Center and another as the Program Management Center; but the Project Management Center must have responsibility for defining the detailed implementation processes to be followed, including reviews, systems engineering, mission assurance, and reserve management. The roles and responsibilities must be documented and understood between Centers or substantial inefficiency may result.

*Document high level agreements in writing.* During the course of a mission development it is likely that key personnel will change. It is difficult to enforce oral agreements across a change in personnel; put them in writing.

*Understand the interests of the various stakeholders and endeavor to keep them informed and aligned.* Stakeholders include the science community at large; the NASA HQ Division Director, Program Scientist, and Program Executive; the Program Office Mission Manager and Mission Scientist (if there is one); the NASA Office of the Chief Engineer (custodian of the Engineering Technical Authority independent reporting path); and the NASA Office of Safety and Mission Assurance (custodian of the mission assurance independent reporting path). This is not a complete list. Indeed it is difficult to make a comprehensive list of stakeholders. They do, however, all have one thing in common: they don't like to be surprised. Upward communication is crucial to keeping the stakeholders aware of issues and everyone in alignment. It is easier said than done.

## **2.2 Project management aspects**

*When recruiting and organizing the team, select experienced institutions and individuals for major deliverable responsibility.* It is tempting to believe that institutions that haven't previously delivered similar products or services can perform as well as institutions that have. This is almost always wrong, even if individuals at new institutions have been involved in similar deliveries at other institutions. Keep the scope of work comfortably within the experience base of the institution. Where this is not possible keep the excursion into new territory as small as possible.

*The relationship between the Project Manager and lead scientist (either Project Scientist or Principal Investigator) must be based upon mutual respect and confidence.* The Project Manager must have confidence that the lead scientist is able to make difficult scope decisions to enable the project to be delivered on budget, and the lead scientist must have confidence that the Project Manager is doing everything possible to deliver the best science capability consistent with available resources. This is the key relationship in the implementation team; if it is not healthy the project is headed for grief.

*Organize both the team and the product along simple lines with clear responsibility.* Complex organization charts are a danger sign. It is imperative that every institution and every project team member understands their role and responsibility in order to minimize conflict and prevent things falling through the cracks. This applies to relationships between institutions as well as within institutions.

*Every subsystem and component must have an individual responsible for its delivery.* It's much easier to hold an individual responsible for a delivery than an institution. It's important that deliverable responsibility be clearly identified and that these individuals are held accountable.

*To the extent possible maintain a stable implementation environment.* It is well understood in the world of acquisition that changes to contract scope cost money. What is equally true, but less well appreciated, is that changes to any aspect of the implementation environment—requirements, processes, rigor of policy enforcement, personnel, funding profile, schedule—all have cost (and risk) ramifications. Change is also fertile ground for unintended consequences.

*Don't underestimate complexity or the difficulty of scaling existing technology.* Increasing emphasis is being placed on assessing Technology Readiness Level (TRL). This relates to the degree to which components or assemblies have been demonstrated to meet requirements in relevant environments. TRLs are used to estimate the degree of risk and the resources required to complete development for flight. When making these assessments it is important not to underestimate the scaling challenges of a given technology. For example, while the Kepler focal plane consisted of fairly conventional charge-coupled-devices (CCDs) its large size (one square foot of silicon) and demanding thermal and electronics noise performance requirements resulted in a complex packaging design that became the schedule critical path for several months <sup>[1]</sup>.

*Have effective metrics and management controls.* Effective management requires timely knowledge of the progress of the implementation. Metrics are necessary to produce the dashboard instrument readings that are used to assess project execution, to understand variances from plan, and to act proactively to resolve problems before they become large. A metric needs things to count: days of slack on the schedule critical path; number of tracking milestones accomplished as a function of time; number of full time equivalent labor charging to an account; number of drawings released; number of software modules through unit level testing, etc. When plotted over time useful trend information is available; when plotted against plans variances can be seen. Metrics not only facilitate effective management, they help communicate to the team the state of execution. Metrics also help the stakeholders understand the context of management decisions.

*Be alive to problems, risks, and opportunities.* It is good practice to maintain a list of problems the team is working. It helps focus the team and communicate issues to stakeholders. It's also good practice to have a rigorous risk management process, a forward looking radar to help head off trouble. Typically this involves maintaining a risk database that is visible to everyone on the project and which is used to collect risk items and manage risk mitigation actions. The database should be combed regularly to keep the team aware of and working the major risk items. In general risks should be crisply described in the form of "if ... then... resulting in" statements, ranked for likelihood and consequence, and given clear assignees. Opportunities (the flip side of risks) that might be taken advantage of should also be identified and tracked.

*Keep the review process under control.* This is easier said than done. The number of reviews a flight project will be required to support is considerably larger than what might appear to be necessary. Reviews consume resources. If the number of reviews becomes excessive they can begin to displace necessary planned work and stress the team to the point where they represent a mission risk. It is important to realize that reviews will not find all flaws.

Reviews come in two main flavors: peer reviews and set-piece reviews. Peer reviews provide the greatest technical benefit to the project by subjecting the detailed design to the scrutiny of independent subject matter experts in a manner that can find flaws in time to affect the design. Peer reviews should be scheduled throughout the development, preferably with a fixed cadre of experts who develop familiarity with the design and the risk posture of the project. Detailed schematics, drawings, and analyses are to be preferred over viewgraphs. Advice should be formally collected and acted upon.

Set-piece reviews are large performances in front of a formal standing review board. Such reviews rarely find technical flaws but they are useful by virtue of the preparation needed to stage them. Just as preparation for an exam is more useful to the student than the exam itself, the preparation for a set-piece review can be very useful to the project team. This preparation allows the team to take stock of and document its progress and status. The fact that major reviews are scheduled in advance provides a focal point and forcing function to complete planned work. The performance at such reviews provides an opportunity for the project to demonstrate to itself and to the stakeholders their command of the material and control of the project execution. Such things build confidence. Ideally the composition of standing review boards should remain constant throughout the project life cycle to minimize the inefficiency of educating new people to the long and complex story that evolves over years. Set-piece reviews are an immutable part of the landscape. They

provide decision makers with the information necessary to progress through key decision points or gates, and serve to keep the stakeholders informed and aligned with the project.

*Balance the risk across the project and reduce the overall risk level to the lowest practicable level.* Given that resources are finite efficiency requires that risks be balanced. This is as much an art as a science and where experience counts for much. It is good practice to accept programmatic risk in preference to technical risk as it is very difficult to buy down technical risk after the fact.

*Don't wait too long to invoke descopes.* The value to be gained by invoking a descope decreases rapidly as the project progresses. The “100:10:1” rule maintains that 100 units of benefit derive from invoking a descope in Phase B compared to 10 units the same descope derives in Phase C, and 1 unit in Phase D. Don't let serious problems fester and become gangrenous—as one colleague puts it: “Sacrifice the arm to save the body.” Invoking descopes can also have an important impact on team dynamics and discipline by demonstrating serious intent to deliver within available resources. This will be noticed by other stakeholders as well.

*Risk acceptance among stakeholders decreases monotonically over time.* It is a fact of human nature that as the project approaches launch risk acceptance decreases. Risks thresholds considered acceptable early in the development will not be acceptable toward the end. This results in much more work than might be anticipated near the time of the Flight Readiness Review and Certification for Flight Readiness. Be prepared for this.

*If schedule is slipping because a key individual or team is overloaded, don't hesitate to add qualified help.* When a key deliverable is behind schedule one often hears the argument that it would take too much time to bring additional help up to speed and that one should let the existing team focus and complete the work. This argument is almost always wrong, and results in help being brought in further downstream when in fact earlier intervention would have been better.

*Maintain transparency and open communication within the team.* Effective execution requires confidence and credibility across and within the team. There will be times when sacrifices are called for and at such times it is crucial that everyone believe there are no hidden margins, no hidden reserves, no hidden agendas, and no secret deals. The best way to build confidence is through transparency and open, recrimination free, communication. Finding flaws should be celebrated as only known problems can be fixed. To the extent possible, project books should be open for all on the team to see.

### **2.3 Systems engineering aspects**

*Assure that requirements are flowed down to a low enough level to support robust verification and validation (V&V).*

When resources are tight it is tempting to descope requirements flowdown. Do this at your peril. The level to which requirements are flowed down should be matched to the level of complexity that needs to be verified and validated, and with due regard for the point in the integration and test flow where problems will be found. If problems are found at too high a level of integration, and therefore too far downstream in the integration and test flow, time will not be available to fix them. This problem occurred on Kepler and resulted in the need for costly and complex software development to correct systematic noise introduced in the focal plane readout electronics<sup>[1]</sup>.

*Have the ability to rapidly assess end-to-end performance and relate science inferences to engineering parameters.*

Replanning is an inevitable feature of project development and often requires rapid trading of risk, cost, and performance in support of decision making. The ability to make trades quickly can be very important to maintaining programmatic credibility in times of crisis. The Kepler project developed a “science merit function tool” that was central to the project's ability to make design trades, manage risks, and, ultimately, to decide what capabilities could be sacrificed when cost growth required descopes. It was also used to develop confidence in the scientific integrity of the mission at key programmatic decision points<sup>[1]</sup>.

*Assure lateral as well as vertical systems engineering coverage.* Conventional systems engineering breaks systems down into deliverable elements, flows down requirements and verifies them back up. This is vertical coverage and it is important. Equally important is lateral coverage, by which is meant assuring that performance qualities that depend on multiple aspects across the system are owned by someone in the systems engineering team. For example, delivering data from a focal plane through a readout electronics subsystem, then through a command and data handling subsystem, then through a telecom subsystem, and finally through a ground data pipeline to the science team requires integrated performance of multiple systems. Someone on the systems engineering team should have responsibility for assuring the end-to-end performance of such capability. Similarly, certain chronological periods require simultaneous and integrated performance of multiple systems. The chronology of the mission should be broken into relevant phases and Phase Leads

assigned to assure integrated performance during that phase. Examples of mission phases include launch, critical events such as an aperture cover release, cruise to a destination, or orbit insertion.

*Pay attention to validation as well as verification.* Verification is concerned with assuring that the system is built right; validation is concerned with assuring that the right system is built. It is possible, even likely, that a system can meet all of its requirements and still not deliver the intended performance or functionality. Many institutions are well disciplined in requirements verification, which stresses process rigor. Validation stresses imagination, and is where system level performance and behavior is characterized, where robustness of off-nominal conditions is explored, and where margins are probed.

*Review the data that is collected in tests.* Tests typically produce large volumes of data that in many cases are not critically reviewed. When an anomaly is seen or a fault occurs downstream in a test flow it is almost always the case that the signature of the problem was present in earlier test data but was not recognized at the time. There is no excuse for not reviewing test data. To this end it is good practice to have subsystem cognizant engineers review the data from their subsystems from tests performed at higher levels of assembly, including fault protection tests. It is also good practice to have scientists review the performance data from engineering tests to assure that problems are not missed. The strongest data review teams or those composed of both scientists and engineers.

*Beware the incubus of hubris.* Good systems engineering requires a degree of purposeful paranoia. As one colleague has put it: “Not all paranoia is unwarranted.” To this I would add: “Assumption is the mother of all mistake.” A healthy skepticism is necessary to achieve high reliability. When in doubt, double check. Consider nothing verified until proven otherwise.

*Toward the end of the integration and test period, assure systems engineering coverage is deep as well as broad.* As integration and testing nears completion and the project enters the final gauntlet of flight readiness reviews it is wise to designate a senior member of the systems engineering team to function as the project Chief Engineer. The Project Systems Engineer will be fully occupied keeping the systems engineering process moving forward and will not have time to lead tiger teams or prowl the landscape looking for mud cracks that have opened up. A Chief Engineer will have the freedom of action to perform the deep dives necessary to get to the root cause of late breaking problems and help keep the project on schedule.

*Count on having to upload a new software version at the launch site.* It just seems to happen.

## **2.4 Aspects specific to telescope missions**

*Beware of the DX program occupying the facility you need.* Facilities capable of manufacturing, coating, and testing space flight optics of 1-meter class or larger are limited, and one often finds a DX-rated program occupying them. DX rating trumps the priority of any science mission. Depending on the conflict this can result in large schedule impacts. For this reason it is good practice to plan large schedule margin in the processing of telescope optics and develop alternative facility options.

*The highest technical risk is nearly always located in the focal plane assembly or readout electronics.* The focal plane is usually where the largest technological stretch is being made, whether it's associated with the detector itself, the requirements of the readout electronics, or the scale of the array. It is wise to retire the risk of the focal plane early, in Phase B if possible. Build a full scale engineering unit and test it under realistic conditions. If possible use flight-like components. Expect production yields to be lower than predicted, and for industry to “loose the recipe” for successful flight production runs.

*Multilayer coatings are usually more difficult than expected.* Another empirical observation: test coatings often work better than the flight coatings. Just because something was made to work once in a test, don't count on it working on the flight run.

*There is usually some aspect of end-to-end performance that cannot be adequately tested on the ground.* End-to-end performance testing can be difficult or impracticable for space telescopes. This requires very thoughtful verification and validation architectures to assure that piece-wise verification by a combination of test and analysis is intellectually rigorous. Such architectures should be thoroughly peer reviewed and should involve multiple methods to cross check the various aspects of the approach.

Two examples are illustrative. SIRTf was designed in such a way that the outer skin of the telescope assembly would passively cool to a temperature below 50K. This is colder than liquid nitrogen, the conventional fluid used to cool the

walls of thermal-vacuum test chambers. Moreover, the ability to model the heat transfer characteristics of the flight system under test required an equally accurate model of the test facility and the interaction between it and the flight system. It turned out to be impractical to design a system level test to fully validate the thermal design of SIRTf. Verification and validation were approached in a piece-wise fashion.

Similarly, the performance of Kepler depends on part-per-million photometric precision of the end-to-end system. It proved impractical to attempt such as test at the system level. Instead a piece-wise approach was taken, combining test data from lower level tests with various analyses.

Such piece-wise verification and validation must be approached with the greatest respect, for Mother Nature cannot be fooled.

## **ACKNOWLEDGEMENTS**

The author wishes to acknowledge several individuals who have over the years provided both ideas and source material for the quotations contained in this paper: Gentry Lee, William Irace, Riley Duren, Pete Theisinger, and Brian Cooke. The work described in this paper was performed at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

## **REFERENCES**

- [1] Management and Systems Engineering of the Kepler Mission, Fanson, J., et al., Proceedings of the SPIE, Vol. TBD, pp. TBD, 2010. (accepted but not yet published)