Quality Interaction between Mission Assurance and Project team members

(Practice for Mission Assurance and other Operations Projects)

Helenann Kwong-Fu *[a], Robert K. Wilson [a]

[a] M/S 264-764, NASA Jet Propulsion Laboratory, California Institute of Technology
4800 Oak-Grove Dr., Pasadena, CA 91109 U.S.A.

ABSTRACT

Mission Assurance independent assessments started during the development cycle and continued through post launch operations. In operations, Health and Safety of the Observatory is of utmost importance. Therefore, Mission Assurance must ensure requirements compliance and focus on process improvements required across the operational systems including new/modified products, tools, and procedures. The deployment of the interactive model involves three objectives: *Team member Interaction, Good Root Cause Analysis Practices, and Risk Assessment to avoid re-occurrences*. In applying this model, we use a metric based measurement process and was found to have the most significant effect, which points to the importance of focuses on a combination of root cause analysis and risk approaches allowing the engineers the ability to prioritize and quantify their corrective actions based on a well-defined set of root cause definitions (i.e. closure criteria for problem reports), success criteria and risk rating definitions.

Keywords: Mission Assurance, Quality, Root Cause, Risks, Metric

1. INTRODUCTION

Traditional assessment approaches are focused on actions and operations and how they contribute to a desired goal. To conduct quality assessments, however, one must frequently interact with multiple institutional team members that is time intensive. To be successful, Mission Assurance must establish effective processes to prevent problems from occurring or repeating. It is important and challenging to obtain quantitative and qualitative assessment while promoting good science products without adversely affecting team moral. On SPITZER, we have applied several reproducible metrics that require the deployment of the interactive model to focus on team member interactions, good root cause analysis practices, and risk assessment collectively to avoid re-occurrences.
The Interactive model is based on Team member interactions with involvement of on-going meetings, assurance of team members' understanding on good practices and decisions. They can be contributed from Root Cause Analysis and Risk Assessment.

Defect or problem reports are generated for non-standard operation processes and undesired outcomes. Team members are being trained and practice Root Causes Analysis on every undesirable outcome. Although capturing anomalies have been around for the last 15-20 years, solving problems to prevent recurrence has always been a challenge. Root causes analysis is now being used in operation diagnostics and, in particular, flight projects. Improvement in the problem prevention would mean that the operation team must have some knowledge of history of pre-launch residuals. There are always risk exist in solving problems. For example, problems may have been solved at static level and may not have properly resolved for long term usage and subsequently surfaces during operations. It is generally believed that the main determinant of problem solving on uncertainty arises from human error and problems that cannot be re-produced.

*Helenann.H.Kwong-Fu@.jpl.nasa.gov; phone 1 818 354-2107; fax 1 818 393-6236

The quality of reproducible metrics that emphasize on problem preventions is highly dependent on understanding root cause, residual risks and errors that are more likely to be made by engineers or operators performing routine daily functions. Given the potential for variation from subsystems to systems, interfaces between hardware and software and project dependencies in particular will require consistent high quality metrics. Monitor repeat problems within the same element under the same conditions over time is absolutely vital to the integrity of any root cause study. We have used several key metrics to measure deviations from standardization protocols and processes. This forms the basis for establishing an optimum method for problem prevention and team interactions.

2. TEAM INTERACTION

Mission Assurance (MA) on team interaction is the key element to success. A valid independent assessments can only be achieved via active interactions with team members (i.e. the daily activities) and have good metrics established with results coming from team members' own processes including their interfaces.

Establishing metrics on capturing changes from baseline, root cause mapping, operational risks, and problems that are diagnosed by team members during operations are all essentials team interaction pointers. Individual engineer may try to optimize the outcome, when deviation exists, by adjusting controls with the use of backups and workarounds. Since human generated errors not always detectable, therefore, interactions or checklists between elements must be exercised. It is important that standard settings be established for all procedures and processes. Engineers or operators should be properly trained to comply with the usage of standard protocols. Applicable error controllers through automation tools, interactive processes and shadowing should be emphasized. Periodic refresh training/updates are highly recommended. MA should not be a hindrance but be value added to the team. Figure 1 demonstrates the key MA interactions with project team members.

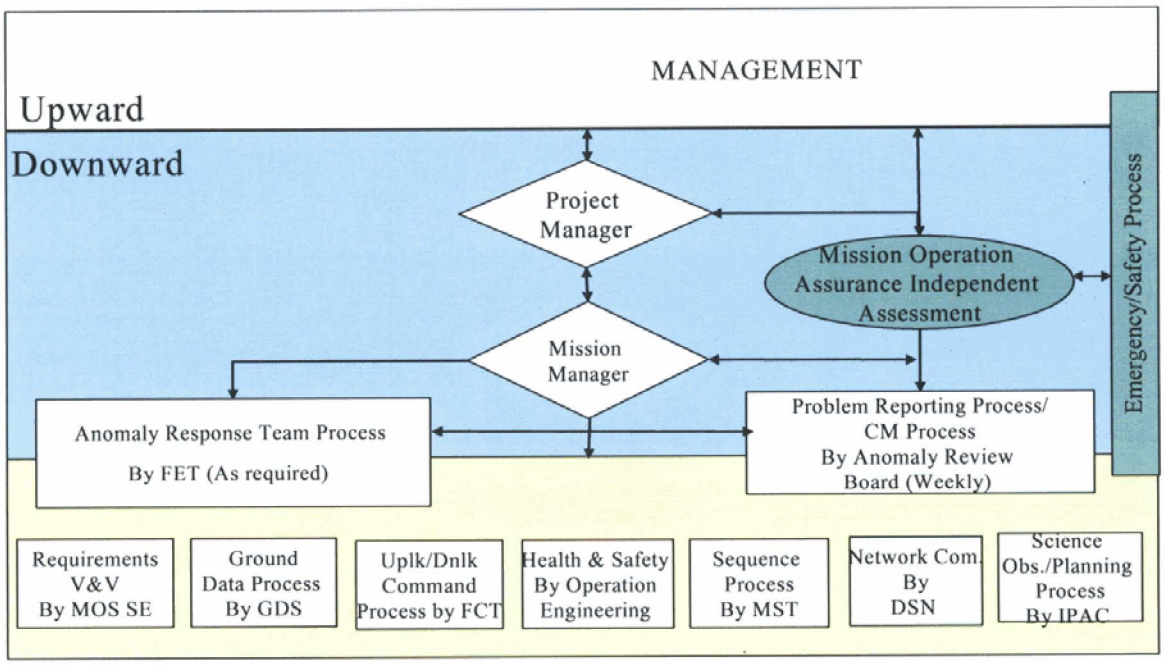

Fig. 1. Operation Team Structure, Process, Decision and Control

2.1 Root Cause Analysis

Problem generating rate can be easily out of control. Aging problems can accumulate rapidly. To overcome these conditions, Mission Assurance uses a proactive approach by implementing Root Cause analysis for all problems generated on SPITZER during operations. However, there are factors which contribute to problem not being generated, such as gray areas exist in team decisions in determining when to generate a problem report. Management decision sometimes may provide a convenient way of having problem reports from being generated

and team members may not be familiar with the processes and baseline. The latter part is more influential with the changes on team members.

Defects or Problem identification is associated with an undesired outcome. Problem description must be clear and easily understood, similar or previous conditions should be noted (including dependencies that are external to the project). Root causes composed of three important areas, identification of its proximate cause of the problem, all potential contributing factors and the actual root cause of the problem. The later is usually supported with the use of fault tree or Fish bone analysis. For Spitzer, team members are required to perform Fish bone analysis or fault tree to support root cause. Heavy team interaction is the key to the success of root cause implementation. Enforcing root cause analysis also enhances the team to share knowledge and have better understanding of the system logic and hence, reduces single point failures staffing problems.

Final corrective action must include the method of verification to the type of testing performed or other mechanism in demonstrating the solution identified was valid. The resolution in verifying whether is a bug in the software, procedure, process, and plan update, sometimes, it may extend to requirement flaws, and all should be captured.

Although problem closures must have valid verification for all correction and resolutions, however, to reproduce the problem is a much more difficult process for projects in operations (post launch). Majority of problems can only rely solely on the test bed (GSE and simulation models), on-board re-verification will subject the observatory to a higher mission risks. Therefore all aspects of impact must have been considered prior making changes on board.

2.2 Risk Assessment

Another important mechanism that must be practice to ensure the Safety and Health of the SPTIZER Observatory is by maintaining an active risk database. Risk management was a practice during development and continued to be enforced in operations. However, risks and risk mitigation are somewhat different for pre and post Launch. JPL Management provides the risk definition for rating, see table 1. Project manager is responsible for the successful guidance of the project in the presence of significant technical and programmatic risk. The Mission Assurance Manager (MAM) ensures/works with each element team member in risk identification, assessing and tracking; entering of risks into the SPITZER Management database, and the development and implementation of mitigation plans, also in compliances to CMMI level 3 (see figure 2)[1].

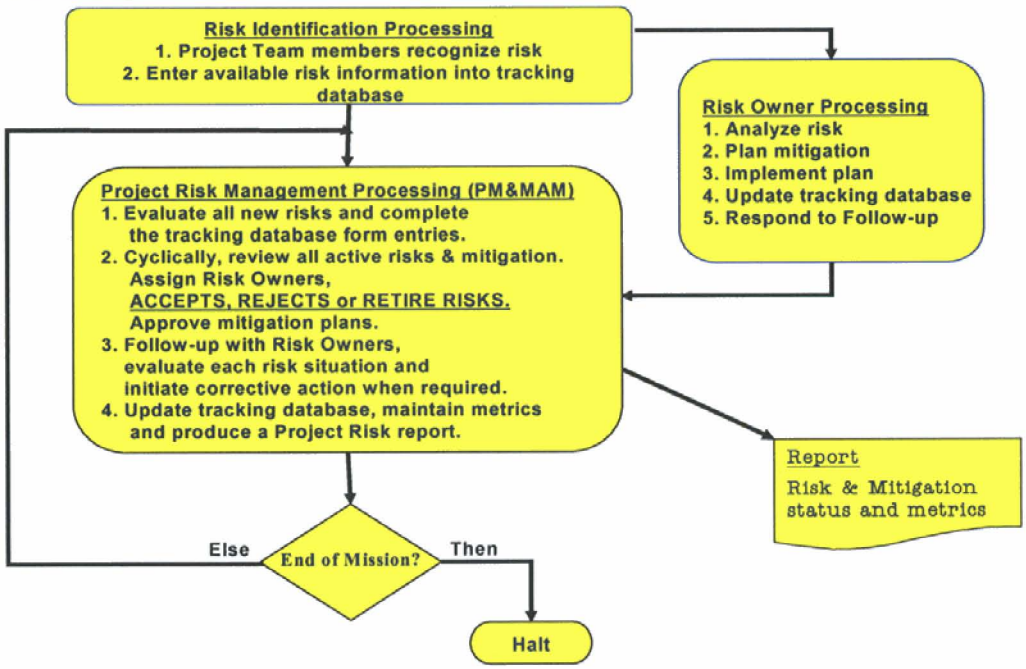

Fig. 2. Risks Process, Decision and Control process

At the SPITZER Project Management Meetings, the MAM communicate program's risk status, upcoming milestones, and highlight risk changes for Project Management Team and Project Manager's review and disposition. At the team level, the MAM holds a risk meeting once a month and discuss current mitigation status and new potential risks. Risk rating of consequences (impact) and likelihood of occurrences are based on a set of risk definitions (JPL definition[2] in compliance to NASA requirement[3]). Table 1 outlines the risk definition[4]; application for both pre- and post launch risks. The final risk posture is a joint decision between project manager and MAM. Having an integrated project risk database provides a structure and process within which the SPITZER project to control, manage and balance risks to achieve mission success under a fixed budget. Figure 3, shows an example of a monthly report demonstrating the monthly highest risks and the risk posture summary in a 5 by 5 matrix. As team members have limited time to carry out mitigation activities besides their everyday functions. Risks mitigation of highest consequence and likelihood usually receives higher priority. Upon completion of mitigation, final evaluation for potential residual risk or retirement based on the mitigation approach and products generated.

Table 1. JPL Risk Rating Definition

	Consequences of Occurrence		Likelihood of Occurrences
	Pre-Launch	**Post Launch**	
5	Overrun budget and contingency, cannot meet launch with current resources	Mission failure	Very High >70%, almost certain
4	Consume all contingency, budget or schedule	Significant reduction in mission return	High >50%, More likely than not
3	Significant reduction in contingency or launch slack	Moderate reduction in mission return	Moderate >30%, Significant likelihood
2	Small reduction in contingency or launch slack	Small reduction in mission return	Low >1%, Unlikely
1	Minimal reduction in contingency or launch slack	Minimal (or no) impact to mission	Very Low <1%, Very unlikely

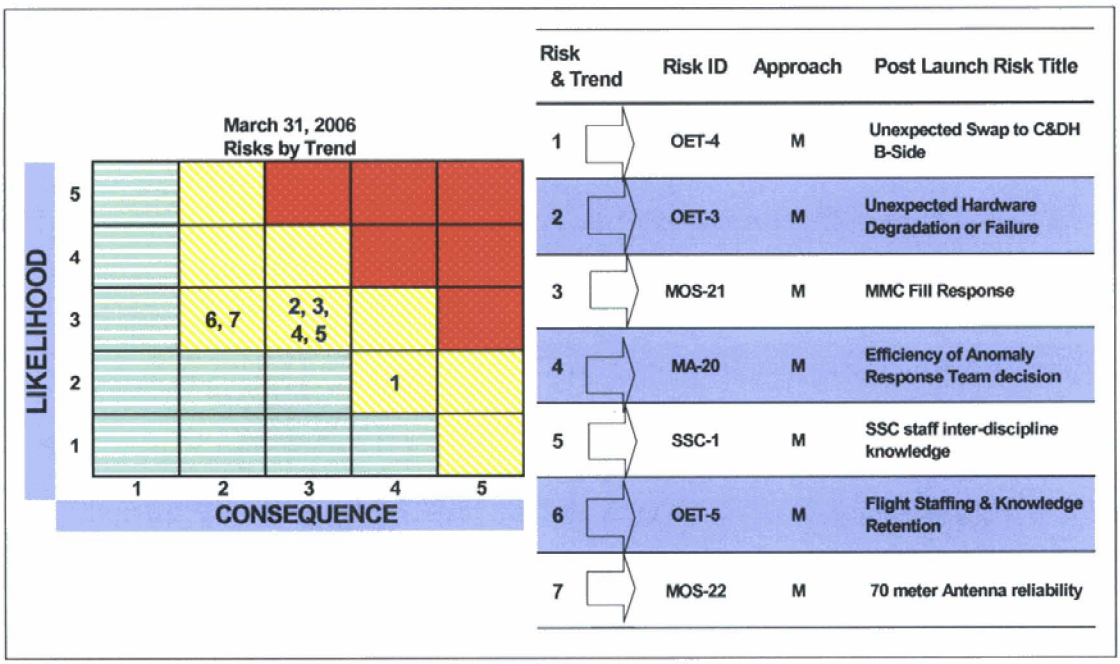

Fig. 3. Risk posture and Significant Risks summary

3. METRICS APPLIED

Metric is used on SPITZER to capture both qualitative and quantitative data of operations through team interactions. Metric begins with Configuration Management. In operations, changes made to baseline are extremely important to the impact of the Safety and health of the observatory. Continuous improvements, problems found during operations, requirement flaws, updating plans and procedures, and products planned for operation deliverables are considered as changes from baseline and are required to submit for approvals from change board. Use of metric to quantify changes allows project to keep good records in compliance to CMMI-level 2 or better[5]. Tracking supporting documentation and analysis of implementation of changes help team members to be more alert of providing proper paper work. Detail on changes to the same parameters can be tracked easily. Figure 4 shows the number of mission change requests (MCR) opened and closed by month and accumulative approved & implemented MCR versus the MCR have been approved but not yet implemented.

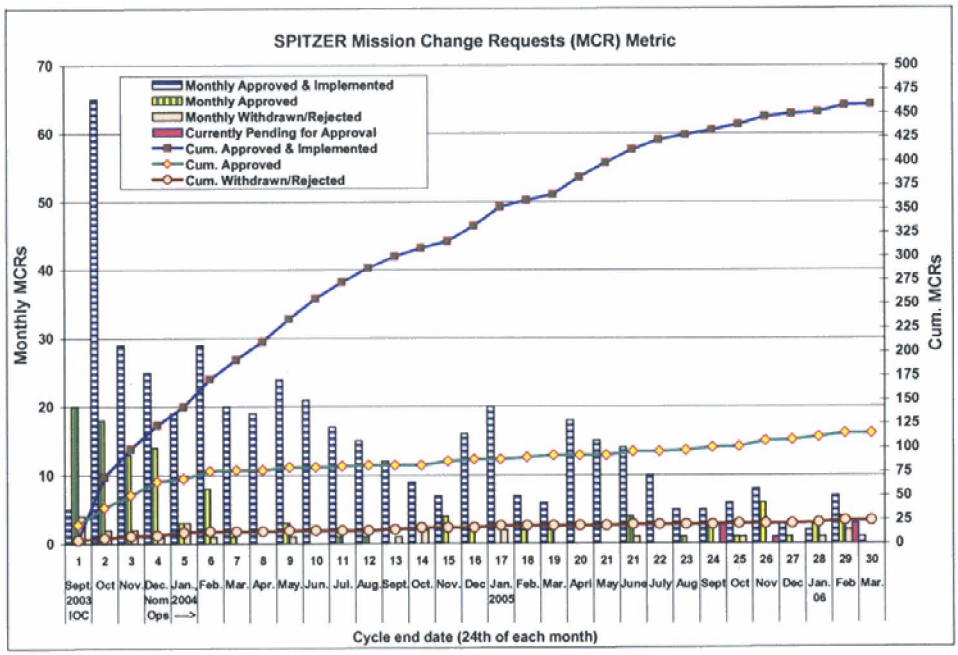

Fig. 4. Mission Change Request by month

Based on experience, the longer the problem remains opened, the higher the difficulty in repeating the problem. There is a need for a metric or tool to help team members and managers to identify aging problems (problems that are open >90 days), along with problem generation rate and trending of higher criticality problems. As shown, figure 5 demonstrates this information clearly. It shows the number of defects or problems opened and closed by month. The closures correlates to the opened problems with in the same month, this way, and one can easily detect where all aging problems occurred (the vertical bars). However, the cumulative opened and closures rates are collected at the end of each cycle month and the closures can be in various months (line curves).

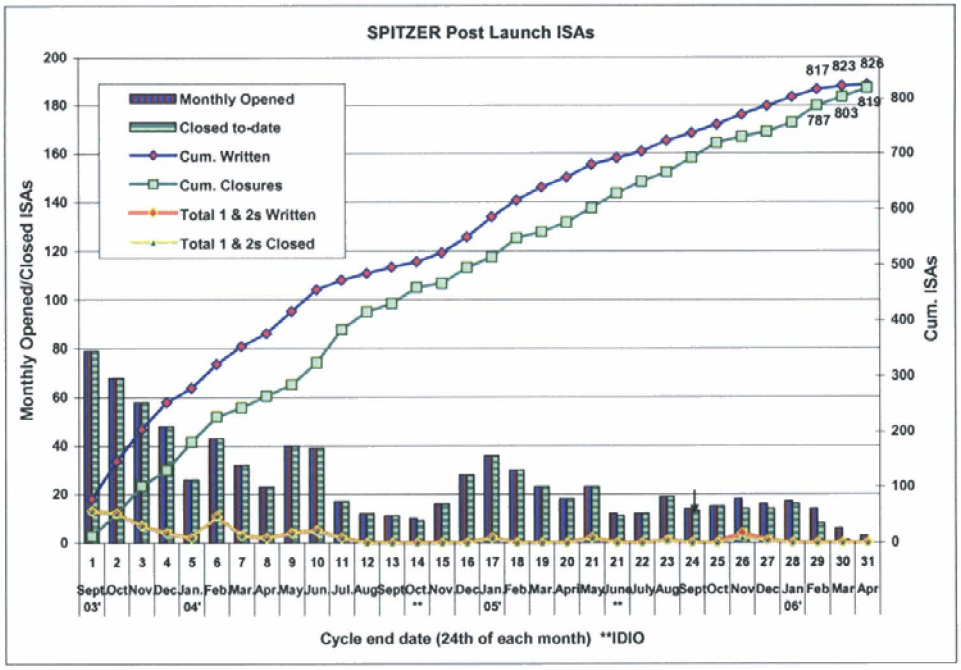

Fig. 5. Mission Problem Reports by month

Another quantitative metric being performed on SPITZER is the ability to manage problems that lies within each subsystem, and the capability that can map out detail root causes and ability to surface the systemic cause. Another useful factor is being able to help team members to recognize their current open problem report status. Table 2 shows a monthly summary problem report by subsystems (detail root cause is not shown here).

Table 2. Problem report summary by element team

Post-Launch ISAs Element	Criticality 1	2	3	4	Still Open	CA-Signed	Closed	Total	(%)
OET		1	1	0	2		45	47	5.69
FET		1	0	0	1		92	93	11.26
GDS		0	4	2	6		229	235	28.45
MST		0	0	0	0		14	14	1.69
MMS-OP		0	8	0	8		271	279	33.78
FCT		0	0	0	0		17	17	2.06
SE/MM		0	2	0	2		10	12	1.45
IRAC		0	0	0	0		8	8	0.97
CE		0	0	0	0		8	8	0.97
IRS/MIPS		0	0	0	0		5	5	0.61
SSC		0	4	1	5		103	108	13.08
Grand total	**0**	**2**	**19**	**3**	**24**	**0**	**802**	**826**	
	0	**0**	**0**	**0**	**0**	**0**	**1236**	**1236**	

4. RESULTS

Significant problems were discovered in the first year of the mission, more specifically the first three months during In-Orbit Check-out. As of today, 59% of the SPITZER problems were generated in the first year (August 25-August 24, 2004), 29% of problems were generated in the second year (August 25, 2004-August 24, 2005), and 12% during

the current year (August 25, 2005-to-date) of the mission. Two element-subsystems holds ~61% of the project problems (Incident Surprise Anomalies), the Ground Data system consist of 28.4 percent and Multi-mission system (DSN support) holds 33.8 percent. Both of these subsystems are external dependencies to the SPITZER project. After 2.5 years of attention put forth in these two areas, their problem generation rate has also decreased.

High correlation between problems found in the first year of operations with mission changes. The metric on mission change requests only had been in placed for 6 months and we found the metric is useful in tracking unfinished paper work such that 115 approved changes that have been implemented still needs final documentation. The change trend showed very high in the first three months but decreased by 5 folds by the end of the second year and remained constant during third year, an indication that majority of the corrections are made in the first year.

The result of implementing Root Cause yields good closures, provide good understanding of the problem, reduces problem report generation rate and mainly prevents problems from re-occurring. Those problems with root cause that can not be ruled out are classified as unverified failures (UVFs). As of today, the UVFs list represents ~2% of the total 826 problems discovered today. Collectively, mapping the potential causes and contributing causes of UVFs could also yield a different perspective in understanding the problems. This work is currently in progress.

Mitigation on Hardware or Software risks identified during development cycle with some risks residuals can become post launch operation risks. Problems with no root cause identified also contribute risks to operations. On-board mitigation is not always feasible; therefore, mitigation for high operation risks usually mainly depend on contingencies procedures. Fault protection parameters tend to be set more conservative prior to launch. After In-orbit checkouts, and along with a few safe-mode and stand-by mode encounters, some of the fault protection parameters have been changed to avoid over sensitive fault protection responses. Although it is ideal to perform risk mitigation and minimize risk in a short duration period, unfortunately, some hardware risks mitigation activities require to be carried out to end of mission such as behavior trending for degradation. Risk mitigation on software requires mostly for patches, but decision on uploading patch adds additional risks to the observatory, therefore patches with new fixes are sometime loaded on board and not use unless is absolutely necessary. In summary, any changes to the observatory should be handled with due diligent.

5. CONCLUSION

Establishing ideal metrics to reveal such conditions is not always simple, and imposing team members to new procedures or new methodologies further complicates this process. The team must be convinced as to why there is a need for such implementation. Thus, the tool must be set up in a form that is easy to carry out. Such a mechanism requires good interaction with team members by providing training and good metric indicators. Additionally, good weekly summary reports enhance the teams' moral.

Although every project has its own unique problems, not every problem is closed with the best resolution. Problems can continue to recur and perhaps rapidly fall out of control. Potential hardware failures in operations can not be avoided; therefore, every precaution must be made to decrease the risks of hardware failures. Such mitigations are on-going by trending the health status of the observatory.

We have applied and used an efficient methodology (metrics) as a basis to gauge the improvement, efficiency, and monitoring of the entire SPITZER operations. The use of metrics as discussed in this paper provided a higher degree of control for improvement management. During development, thousands of software problems and over hundreds of hardware problems or defects were generated. Had similar applications been put in place, we could have managed to solve problems or defects more efficiently.

In summary, there is an increase in reproducibility used for reporting. The CM metric provided better control on documentation of changes. The ISA metric allowed better understanding of root causes, and risk mitigation emphasis used trending to monitor hardware degradation. Documentation on root causes was excellent, and problem report generation rate diminished. In concentrating on the health and safety of the observatory, the mission assurance manager, along with the team members, ensured the efficient operation of the observatory.

REFERENCES

1. B. Vargo, Flight Software Development, Rev. 0, D-71712, Jet Propulsion Laboratory, Pasadena, 2005.
2. T. Gavin, "*Flight Project Practices Compliance*", Rev. 6, D-58032, Jet Propulsion Laboratory, Pasadena, 2006.
3. NASA, "*Requirements deviations impacting formulation*", NPR 7120.5C, NASA, Maryland
4. T. Gavin, "JPL Quarterly Template", Jet Propulsion Laboratory, Pasadena, 2003.
5. D. Nichols, "Software Quality Improvement Project", Rev. 0, D-66672, Jet Propulsion Laboratory, Pasadena, 2004.