Web Based Tool for Mission Operations Scenarios

Carole A. Boyles* and Dr. Duane L. Bindschadler†
Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California, 91109

A conventional practice for spaceflight projects is to document scenarios in a monolithic Operations Concept document. Such documents can be hundreds of pages long and may require laborious updates. Software development practice utilizes scenarios in the form of smaller, individual use cases, which are often structured and managed using UML. We have developed a process and a web-based scenario tool that utilizes a similar philosophy of smaller, more compact scenarios (but avoids the formality of UML). The need for a scenario process and tool became apparent during the authors' work on a large astrophysics mission. It was noted that every phase of the Mission (e.g., formulation, design, verification and validation, and operations) looked back to scenarios to assess completeness of requirements and design. It was also noted that terminology needed to be clarified and structured to assure communication across all levels of the project. Attempts to manage, communicate, and evolve scenarios at all levels of a project using conventional tools (e.g., Excel) and methods (Scenario Working Group meetings) were not effective given limitations on budget and staffing. The objective of this paper is to document the scenario process and tool created to offer projects a low-cost capability to create, communicate, manage, and evolve scenarios throughout project development. The process and tool have the further benefit of allowing the association of requirements with particular scenarios, establishing and viewing relationships between higher– and lower-level scenarios, and the ability to place all scenarios in a shared context. The resulting structured set of scenarios is widely visible (using a web browser), easily updated, and can be searched according to various criteria including the level (e.g., Project, System, and Team) and Mission Phase. Scenarios are maintained in a web-accessible environment that provides a structured set of scenario fields and allows for maximum visibility across the project. One key aspect is that the tool was built for a scenario process that accounts for stakeholder input, review, comment, and concurrence. By creating well-designed opportunities for stakeholder input and concurrence and by making the scenario content easily accessible to all project personnel, we maximize the opportunities for stakeholders to both understand and agree on the concepts for how their mission is to be carried out.

I. Background

Operations scenarios are widely accepted as a tool to help write good and complete requirements [1,2,3]. In the software development realm, scenarios are put to a wide variety of usages during design, implementation, and testing phases of the project lifecycle [4]. Scenarios are often treated as relatively informal tools; widely divergent methods have been applied to developing and managing them. In particular, Achour [5] advocates developing guidelines for scenario authors, in order to maintain clarity and uniformity of approach to scenarios. A broad survey of scenario development and management practice [6] revealed difficulties in managing consistency in content and level or abstraction of scenarios, and in maintaining control of the content of scenarios as they evolved during project lifecycles. This has lead some to suggest the need for some form of scenario management tool [4].

We contend that scenarios have considerable utility in the development of spaceflight projects throughout formulation and implementation. Operations scenarios identified during the concept and preliminary design phase of a project are useful for communicating the need for operations capabilities and driving out project (level 2) and system (level 3) requirements, identifying design trades or other issues, identifying operations constraints, and identifying interfaces (operational and software) in preparation for Preliminary Design Reviews (PDR's). As the flight and ground systems mature, more-detailed scenarios continue to drive out requirements and raise issues.

* Assistant Manager for Planning and Development, Multi-Mission Ground Systems and Services (MGSS), 4800 Oak Grove Dr., Pasadena, CA 91109/Mail Stop 264-235

† Assistant Manager for Operations, Multi-Mission Ground Systems and Services (MGSS), 4800 Oak Grove Dr., Pasadena, CA 91109/Mail Stop 264-235

During a project's implementation phase, scenarios should be used to identify success criteria for verification and validation of the mission system and as tools for training and certification of operations personnel.

A conventional practice for spaceflight projects is to document scenarios in a monolithic Operations Concept document. Such documents can be hundreds of pages long and may require laborious updates. The Mission Scenarios Tool is a web-based tool that allows easy visibility into varying levels of scenarios. The need for a scenario process and tool became apparent during the authors' work on the Ground Segment for the Space Interferometry Mission, a large astrophysics observatory. It was noted that every phase of the mission (e.g., formulation, design, integration and test, and operations) looked back to scenarios to assess completeness of requirements and design. It was also noted that terminology needed to be clarified and structured to assure communication across all levels of the project. Attempts to manage, communicate, and evolve scenarios at all levels of a project using conventional tools (e.g., Excel) and methods (Scenario Working Group meetings) were not effective given limitations on budget and staffing.

A scenario development process and tool represent a collaborative method for developing evolving, and maintaining mission scenarios. The tool has further benefit of allowing the association of requirements with mission scenarios, and the ability to link operational and software interfaces placing them in a shared context. The result is a structured set of scenarios, which can easily be searched according to the level of detail (e.g., Project, System, Subsystem level) or mission phase.

II. Operations Scenarios Lifecycle Overview

Our objective is to offer a systematic, cost effective approach to managing, maintaining, and communicating operations scenarios across the lifecycle of a spaceflight project. This paper introduces the notion of revising scenarios at key decision points to assure the usefulness of operations scenarios in each lifecycle phase. Each version of the scenario is maintained to provide data to future projects on the level of effort and benefit of producing a complete set of operations scenarios.

The purpose and use of three hierarchical levels of operations scenarios are developed to convey varying amounts of detail. We refer to the highest level as Project scenarios. The purpose of developing Project scenarios is to communicate the higher level activities or 'phases' of a mission. Project scenarios set the ground rules for engineering and science expectations and convey the activities allocated to each phase of the mission. The core of scenario development occurs at the next level or System level. System scenarios generally consist of an activity that occurs in one or more Project-level scenarios and represents a 'slice' through mission operations (or Phase E of the Project lifecycle). The purpose of developing System scenarios is to assist in driving out requirements, assure the intent of the requirements meets the needs of mission operations, and assess the completeness of the mission system requirements set. The lowest level of scenario is called Intra-Team. The purpose of Intra-Team scenarios is to assist the Mission Operations System in verifying processes and procedures during the integration and test phase of a project and also for training and certification of operations personnel.

Creating a Scenarios matrix is an effective means to map out hierarchical scenarios at the Project and System level. The mapping of hierarchical scenarios is complex because multiple instances of a system scenario may be created to support different phases (Project scenarios) in the mission. The system level scenario 'DSN Contact' is a good example (Figure 1). During Launch support, the DSN contact coverage and command strategy is different than during support of Nominal Cruise. The use of the scenario matrix in the requirements process is described in [7].

An Overview of scenario status at Key Decision Points (KDP) in a project lifecycle is shown in Figure 2, using the NASA Project lifecycle as documented in NASA NPR 7120.5D. The detailed description of scenario development and maintenance is described below for each phase in the project lifecycle. Consider the following caution: Systems engineers should avoid the temptation of placing too much detail in scenarios before the flight and ground systems are mature. Stay at the level of detail sufficient to meeting the Key Decision Point objectives.

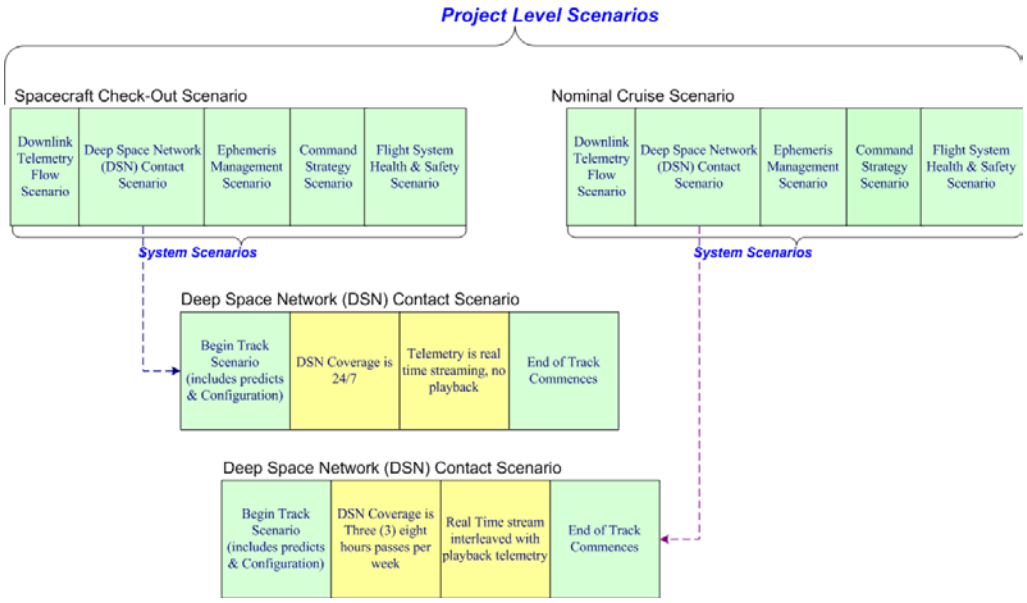

Figure 1. Example of DSN Contact System Scenario

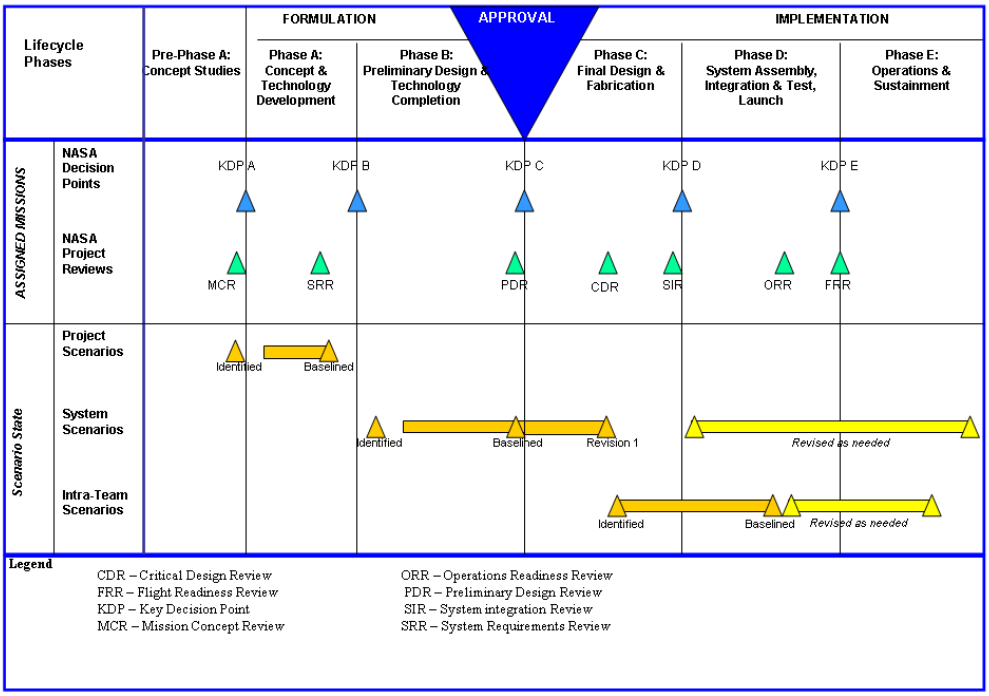

Figure 2. Scenario Status Over Project Lifecycle

A. Concept and Technology Development Phase (Phase A)

Project scenarios are characteristic of the various mission phases. The purpose of Project scenarios in Pre-Phase A and Phase A is to assist in costing the development and operations of the mission and to assist in identifying key technology and interface issues as early in the project lifecycle as possible. The first KDP for Project scenarios is at the Mission Critical Review (MCR). Project scenarios should be sufficiently mature to be baselined in support of MCR. Examples of Project scenarios include Assembly, Launch, and Test Operations (or Integration and Test Operations), Launch, In Orbit Checkout, Instrument Checkout, Flight System Verification, Science Verification, Fly By's, Aero braking, Orbit Insertion, Cruise, Tour, Nominal Operations, and Operations Contingency Scenarios. Participants in Project Scenarios are typically the technical leads for project systems (e.g. Mission System, Flight System, and Science Operations System). Project scenarios require participation and 'buy-in' from project manager, project scientist (or principle investigator), and system level managers. Contingency scenarios may be identified but not typically baselined at MCR due to lack of maturity in the flight and ground systems.

B. Preliminary Design and Technology Completion Phase (Phase B)

Project scenarios tend to be relatively stable through the rest of a project's lifecycle. Development of System scenarios starts with Phase B and represents the core of operations scenario work . System scenarios are used to

- Assess requirements' completeness and assist in identifying "missing" requirements.
- Validate requirement intent against the scenarios.
- Identify assumptions that require verification, or other open issues.
- Identify operational (process) or software interfaces.
- Identify constraints on mission operations and assess operability.

System scenarios represent a slice through mission operations at a given point in the mission timeline. It is important that flight and ground system capabilities, constraints, and open issues be understood as early as possible by both the flight and ground system teams to reduce the cost of missing requirements late in the development lifecycle. (One may conclude that the development of systems scenarios is the beginning of training in support of mission operations readiness.) System scenarios are baselined at the second Project KDP -- Preliminary Design Review (PDR). A more detailed description of how scenarios are used in developing requirements can be found in [7]. Examples of System scenarios include DSN Contact, Ephemeris Management, Data Accountability, FSW Management, Spacecraft Health & Safety, Instrument Health & Safety, Instrument Calibration, Sequence Strategy, Real Time Command Strategy, Reaction Wheel Management, High Gain Antennae Management, SSR Management, and Telemetry Mode Management. System scenarios require participation and 'buy-in' from Project Systems Engineering, System level Managers, development and operations teams (e.g. spacecraft development and operations teams, sequencing team, instrument development, science operations teams, and the data processing team).

C. Final Design and Fabrication (Phase C)

As the flight and ground systems mature the development effort for operations scenarios progresses. Project Contingency scenarios are developed in Phase C and System scenarios are revised in support of project Critical Design Review (CDR). We recommend preserving the scenario set at key decision points because of their significant roles (e.g. requirements assessment at PDR). In addition, preserving scenarios provides notable metrics to future projects on scenario development and level of effort. The purpose for revising scenarios in Phase C is to drive out any remaining requirements, finalize interfaces, and update scenarios based on trade studies or closed issues. Revisions of scenarios should follow the same signature process used to baseline. Intra-Team scenarios are identified and developed in Phase C. The purpose of Intra-Team scenarios is to identify and assure processes, their associated timelines, and procedures are complete, consistent, and ready for mission operations. Some examples of Intra-Team scenarios include generation of Instrument Sequences, Validation of Sequences, Storing, and Querying telemetry. Participants in Intra-Team scenario development are generally team members. Approval and change control of Intra-Team scenarios should be the responsibility of the Team Lead.

D. System Assembly, Integration & Test, Launch (Phase D)

Project and System scenarios are used in Phase D to identify success criteria for verification and validation of the Mission Operations System and as tools for training and certification of operations personnel. Intra-Team scenarios are commonly baselined in support of Operations Readiness Tests (ORTs) where they will be

verified/validated for use in Mission Operations. Project, System, and Intra-Team scenarios are commonly used as a tool for training and certification of operations personnel.

III. Operations Scenarios Tool

The core requirements on the operations scenarios tool were established via a broad survey of scenario development and management practice [6] and through the author's work on the Ground Segment for the Space Interferometry Mission, a large astrophysics observatory. The tool resolves some of the difficulties in managing consistency in content and level or abstraction of scenarios, and in maintaining control of the content of scenarios as they evolve during project lifecycles. The tool has further benefit of allowing the association of requirements with mission scenarios, and the ability to link operational and software interfaces placing them in a shared context. The result is a structured set of scenarios, which can easily be searched according to the level of detail (e.g., Project, System, Subsystem level) or mission phase.

The operations scenario tool was designed to assure scenarios were easily accessible to all customers and stakeholders, scenario content and terminology was consistent across all scenarios, and scenarios could be managed in an environment allowing for ease of collaboration and concurrence. The biggest challenge was to provide the ability to easily compare scenarios to their applicable requirements for validating intent and requirements completeness. The operations scenario tool is built on the Structured Query Language (SQL) and Hypertext Preprocessor (PHP). The tool can run on varying types of servers (e.g. Unix, Linux, PC, MacIntosh) and be accessed from any web based browser. The core capabilities of the operations scenario tool are detailed below.

A. Core Capabilities

1. Scenario Accessibility

Scenarios are maintained in a web accessible environment that provides a structured set of scenario fields assuring consistent content across all scenarios. Scenarios are easily visible regardless of the user's location. Fields in the scenario operations tool are easily searched. Search parameters include the scenario owner, team interfaces, status of each scenario, and requirement(s) applicable to each scenario. Scenarios can be maintained in the project secured server/Local Area Network (LAN) or in a multi-mission environment.

2. Online Collaboration

The scenario tool allows for collaboration through the use of a comments field. Once a scenario is initiated it becomes visible to all customers and stakeholders. Any member of the project has the capability of submitting a comment. The comment is instantly visible on the dashboard of the scenario. Team members have the advantage of responding to the comment or by taking the collaboration into one of the project forums for further discussion.

3. Interleaving Requirements with Scenarios

The scenario tool has the ability to embed requirements into the scenario description allowing for easy comparison of applicable requirements. The scenario tool can be linked to a requirements data base, such that each time a scenario is viewed the requirements are updated assuring the correct version of the requirement is being viewed. Comparison of requirements to the scenario is used to help ensure the completeness of the requirements set. The scenarios process calls for the first version of the scenario to be signed off in preparation for the PDR. After this time the requirements will no longer be updated. This creates a historical record of the scenario with its applicable requirements. As the project enters the Final Design phase, the scenarios will be revised and again the requirements will be pulled into the scenario description and updated upon each viewing of the scenario.

4. Linking to Interfaces

Operational Interface Agreements (OIA's) are created to communicate agreements between teams. With the Advanced Multi-Mission Operations System (AMMOS) at JPL, the OIA's are maintained in a web based tool similar to the scenario tool. The Scenario Tool has the capability of linking applicable OIA's within each scenario. The benefit of having the OIA's linked directly into the scenario become apparent when preparing for Operations Readiness Tests and Review in preparation for launch. Scenarios used to create operations thread tests or ORTs, also show which interfaces are to be verified by a particular test. The capability also exists for the Software Interface Specifications (SIS) created to identify interfaces between software.

5. Maintaining Scenario Mapping

The scenarios tool takes the complex task of maintaining and tracking the interdependencies of scenarios and automates it. The scenario tool has the capability to link scenarios at multiple levels (as shown in Figure yy above). System scenarios can be linked to the Project scenarios they support. Intra-Team scenarios can be linked to the system scenarios they support.

6. Managing the Scenario Process

The scenario tool was designed with the intent to support the process owner in facilitating the operations scenario process as detailed in [7]. The web based scenario tool has been designed with attributes to assist in both communication and collaboration. The attributes specific to managing the Scenario Process are:

User Access – Three user levels are defined, each with varying capabilities. This allows flexibility in collaboration while managing control of the content and status of each scenario as shown in Table 1.

Table 1. Use Access Control Table

Capabilities	User Types		
	Any User	Scenario Owner	Process Owner
Read All Scenarios	X	X	X
Create Scenarios	X	X	X
Update Scenarios		X	X
Add Comment	X	X	X
Manage Signatures/Revisions/Action Items of Scenarios		X	X
Copy a Scenario	X	X	X
Revise Scenarios		X	X

Status – The status attribute controls the ability to update the scenario and requirements when appropriate and consequently freezes the content when the scenario is ready for concurrence. The different terms for status are meant to represent the level of maturity of the scenario. They control the automatic distribution of emails (see below), the automated updating of requirements embedded in the scenario, and privileges of users to update scenario content. While a scenario is in development (Draft or Initiated status), requirements are updated automatically and the scenario owner can freely edit the scenario. Once the scenario is ready for electronic signature (Concurrence Pending), requirements are no longer updated and scenario content is frozen except updates by the scenario process owner. A final state (Approved) indicates that all pending actions associated with the scenario have been closed out. Used with the appropriate process, this capability automates most of the configuration control work associated with a particular scenario.

Automatic email – Automated email is a great asset to the process owner. Email is triggered based on change in status of a scenario. The scenario tool has a 'mailing list' feature that allows individuals to sign up to receive emails. Individuals can sign up for emails based on participating team, participating system or subsystem, or project. In status=draft no emails are sent. Starting with a status change to 'status= initiate' an email is sent to the process owner and individuals who have signed up to receive emails. When the status changes to 'status=concurrence pending' emails are also sent to the signatures as well notifying them that a scenario is ready for their concurrence.

Electronic Signatures - The Scenario Tool has the capability of including electronic signatures. Signatures are used to assure that all teams involved in the scenario agree to the content and their respective roles and responsibilities. It also helps establish a common understanding of the currently outstanding issues in the context of the scenario.

Action Items - Action Items assist the process owner in documenting actions that are currently in process for each scenario. The need to address missing requirements is an example of an expected action item during the scenario process. As new requirements are identified the initiator or process owner selects 'new action item'. A dialogue box opens where the initiator or process owner can type in the action item and select the person to assign it to. Upon 'send' the action item is documented at the bottom of the scenario and an email is automatically sent to the assignee. Once the action is resolved, the initiator or process owner can 'close' the action. One of the key values of having

action items is the ability to know the current status of a scenario with a quick glance, including current actions being taken.

7. Revision Capability

Development of mission scenarios is focused in Preliminary Design (or Formulation) Phases. The approval gate for proceeding into the Final Design and Fabrication Phase is the Preliminary Design Review (PDR). Scenarios are used at PDR to demonstrate completeness of requirements, design maturity, and identify any outstanding issues or trades still needing to be performed. During Final Design and Fabrication the scenarios will mature. It is recommended that each scenario be revised, preserving the state of the original scenario at PDR.

Scenarios can only be revised after they have been approved and revision can only be initiated by the scenario owner (or initiator) or the scenario process owner. When a revision of a scenario is created, it reverts to a draft state and the process used for the original scenario is repeated. All description and information fields in the original scenario are retained by the revision, as well as linkages to interfaces or to other scenarios. Action items, signatures, comments, and attachments are cleared in the revision. Requirements are updated again each time the revised scenario is viewed.

8. Copy Capability

Copy capability is one of the most useful aspects of the operations scenario tool. It was originally intended to give systems engineers an opportunity to quickly create unique cases of scenarios. Any user who has access to the scenario database can copy a scenario. Unlike a revision, a copy does not preserve links to interfaces or other scenarios.

9. Attachments

The scenario tool has the capability to attach numerous files to the scenario. One attachment needed is the scenario flow diagram. Other attachments can be added as needed to assist in better defining the scenario and its context.

10. Search Engine

The scenario tool has a search capability. The search capability was created to offer a quick look at the current status of all scenarios, which the scenario owners are, teams are participating in each scenario, and requirements associated with each scenario. This quick search engine

B. Multi-Mission Use

The web based scenario tool resides in the Advance Multi-Mission Operations System (AMMOS) at JPL. Scenarios are visible to all missions who have access to the AMMOS. A significant advantage of having scenarios in a database versus a document is the ability to leverage off of the work one or more missions have already done on scenarios. A new mission coming on has the ability to review and copy applicable scenarios from multiple missions in their own scenario database giving them a significant jump on the development of their mission scenarios set.

IV. Conclusions

Mission Operations Scenarios have considerable utility throughout the lifecycle of a project. Managing, maintaining, and communicating operations scenarios can be accomplished using a structured process and a web based tool. The key core capabilities of the Operations Scenario Tool resolves some of the difficulties in managing consistency in content and level or abstraction of scenarios, and in maintaining control of the content of scenarios as they evolve during project lifecycles. The Operations Scenarios Tool has further benefit of allowing the association of requirements with mission scenarios, and the ability to link operational and software interfaces placing them in a shared context. Two key capabilities that assist in managing scenarios are the copy and revision capability. Copy is useful for quickly creating unique cases of scenarios while the revision capability allows for the preservation of scenarios at Key Decision Points.

Acknowledgments

This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

References

[1]Leite, J.C.S.P. Rossi, G.; Balaguer, F.; Maiorana, V.; Kaplan, G.; Hadad, G.; Oliveros, A., "Enhancing a requirements baseline with scenarios," *Proceedings of the Third IEEE International Symposium on Requirements Engineering*, Annapolis, Maryland, USA, 1997, pp. 44 – 53.

[2]Hooks, I.F, and Farry, K.A., *Customer Centered Products: Creating Successful Products Through Smart Requirements Management*, 1st ed., AMACOM/American Management Association, New York, 2001.

[3]Fairley, R.E., and Thayer, R.H., "The Concept of Operations: the bridge from operational requirements to technical specifications," *Annals of Software Engineering,* Vol. 3, 1997, pp. 417-432.

[4]Alspaugh, T.A., Anton, A.I., Barnes, T., Mott, B.W., "An integrated scenario management strategy," *Proceedings, IEEE International Symposium on Requirements Engineering*, Limerick, Ireland, 1999, p.142.

[5]Achour, C. B., "Writing and correcting textual scenarios for system design," *Proceedings, Ninth International Workshop on Database and Expert Systems Applications*, Vienna, Austria, 1998, pp. 166-170.

[6]Weidenhaupt, K., Pohl, K., Jarke, M., Haumer, P., "Scenarios in system development: current practice," *IEEE Software*, Vol.15, No, 2, pp. 34-35.

[7]Bindschadler, D.L., and Boyles, C.A., "A scenario-Based Process for Requirements Development: Application to Mission Operations Systems," *Proceedings of the SpaceOps 2008 Conference*, Heidelberg, Germany, 2008, (in press).

Appendix A. Scenario Matrix Example

System Scenarios ↓	Project Scenarios				Scenario Scope	Status	Cognizant Engineer
	Launch	IOC	Cruise	Tour			
Data Accountability	A	A	A	A	Describes the MS function for Data Accountability.	Draft	GDSE
Downlink Telemetry	A	A	A	A	Describes Tlm Flow from DSN receipt to Instrument teams.	Draft	MOSE
DSN Contact	A	A	B	B	Describes the scheduling process for DSS coverage through the timeline	Draft	MOSE
Ephemeris Management	A	A	A	A	Describes the ephemeris generation and update strategy and timeline	Final	Nav.
Flight Software Management	A	A	A	A	Describe flight system FSW strategy for updates.	Final	S/C Team
Spacecraft Health & Safety	A	A	A	A	Describes the functions performed by the MS to process all relevant spacecraft telemetry to monitor the performance and long-term health	Final	S/C Team
Sequencing Strategy	A	A	A	A	Describes the Sequence Generation Process from receipt of Short Term Schedule from SOS to verification of successful receipt by the Flt Sys (includes Seq U/L Strategy)	Final	Seq.Team
Real Time Command Strategy	A	A	A	A	Describes the Real Time Command (RTC) strategy from RTC request to verification of successful receipt by the Flight System	Final	Seq. Tem
Reaction Wheel Management	A	A	A	A	Describes the strategy for ground management of Flight System stored momentum	Draft	S/C Team
Solid State Recorder Management	A	A	A	A	Describes the strategy for managing all partitions in the SSR.	Draft	S/C Team
Anomaly Response	A	A	A	A	Describes the Anomaly Response Process	Draft	MOSE