S3106: PBRA and RBA Process

Version: E

Effective Date: April 26, 2018

Process Owner: IVVO Lead, Joelle Spagnuolo-Loretta

Note: The official version of this document is maintained in IV&V's internal IV&V Management System Website (https://confluence.ivv.nasa.gov:8445/display/IMS). This document is uncontrolled when printed.

Project Based Risk Assessment (PBRA)

Risk Assessment to Support Mission Capability Prioritization for NASA

IV&V Projects and Risk Matrix to Support the IV&V Program Portfolio

And

Risk Based Assessment (RBA)

Risk Based Assessment of Software-level Entities for NASA IV&V Projects

Introduction

Independent Verification and Validation (IV&V), as a part of Software Assurance, plays a role in the overall NASA software risk mitigation strategy applied throughout the lifecycle, to improve the safety and quality of software systems.[1] In order to understand the software risk profile within NASA, NASA IV&V performs assessments of risk on Mission Projects. These assessments are primarily intended to create a mission-specific view of software risk to support planning and scoping of NASA IV&V Project work on each individual IV&V Project that also supports the IV&V Program's portfolio level prioritization determinations. This document contains a two phase process for performing these assessments. Phase One is a Mission level assessment, referred to as the Project Based Risks Assessment (PBRA), that is focused on the development and assessment of Mission level capabilities necessary for the achievement of mission success. Phase Two is a lower level assessment that is performed on the software entities within the system that are necessary for the performance of the defined Mission level capabilities. Phase Two is commonly referred to as the Risk Based Assessment (RBA).

The PBRA process results in a risk score for each mission level capability derived from evaluating each capability against categories of risk. The RBA process results in a risk score for each mission specific system/software entity. An outcome of the PBRA is an initial set of mission level Assurance Objectives that help refine the IV&V Project's area of focus from a system level perspective. The RBA begins to further refine the focus of those Assurance Objectives to the system/software level. The Assurance Objectives will then be used by the IV&V Project team to help determine what analysis activities should be performed with a goal of providing evidence that can turn those Assurance Objectives into positive Assurance Conclusions as IV&V is executed on the Project.

Both of the processes are to be evaluated iteratively during the IV&V Project lifecycle, as additional information about the mission and software becomes available.

Figure 1 below shows the evolutionary process of planning and scoping that an IV&V project takes throughout the project lifecycle. While the project initialization and completion of the lifecycle are time-based, the activities listed in the middle of Figure 1 will continue to be cycled through as the IV&V team executes analysis on the project throughout the entire lifecycle.

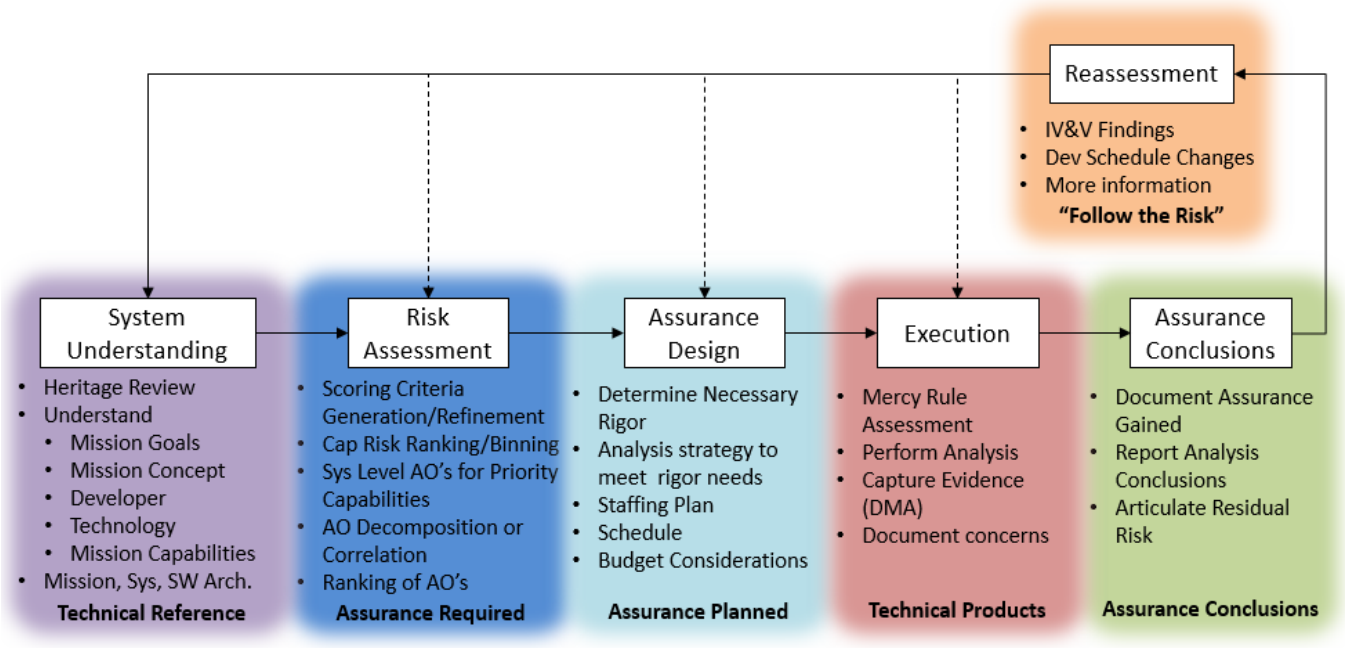


Figure 1: IV&V Project Planning and Scoping Process

Focusing on the first two segments of the Planning and Scoping Process, the System Understanding and Risk Assessment, shown in Figure 2, is where an IV&V project team starts the project initialization process. A precursor to even the PBRA assessment is a Heritage Review. The Heritage Review, represented as the orange highlighted area, is performed to help the IV&V project team evaluate how the new project compares to existing IV&V knowledge and experience. The outcome of the Heritage Review will feed directly into the PBRA process. The green outlined area then represents where the PBRA and RBA processes are performed. During project initialization, the RBA process may not be performed at all given that little may be known about project software characteristics at initialization in most cases. If an RBA is performed at initialization, it will likely be performed from a more high-level, system architecture perspective, and less from a low-level, software perspective.

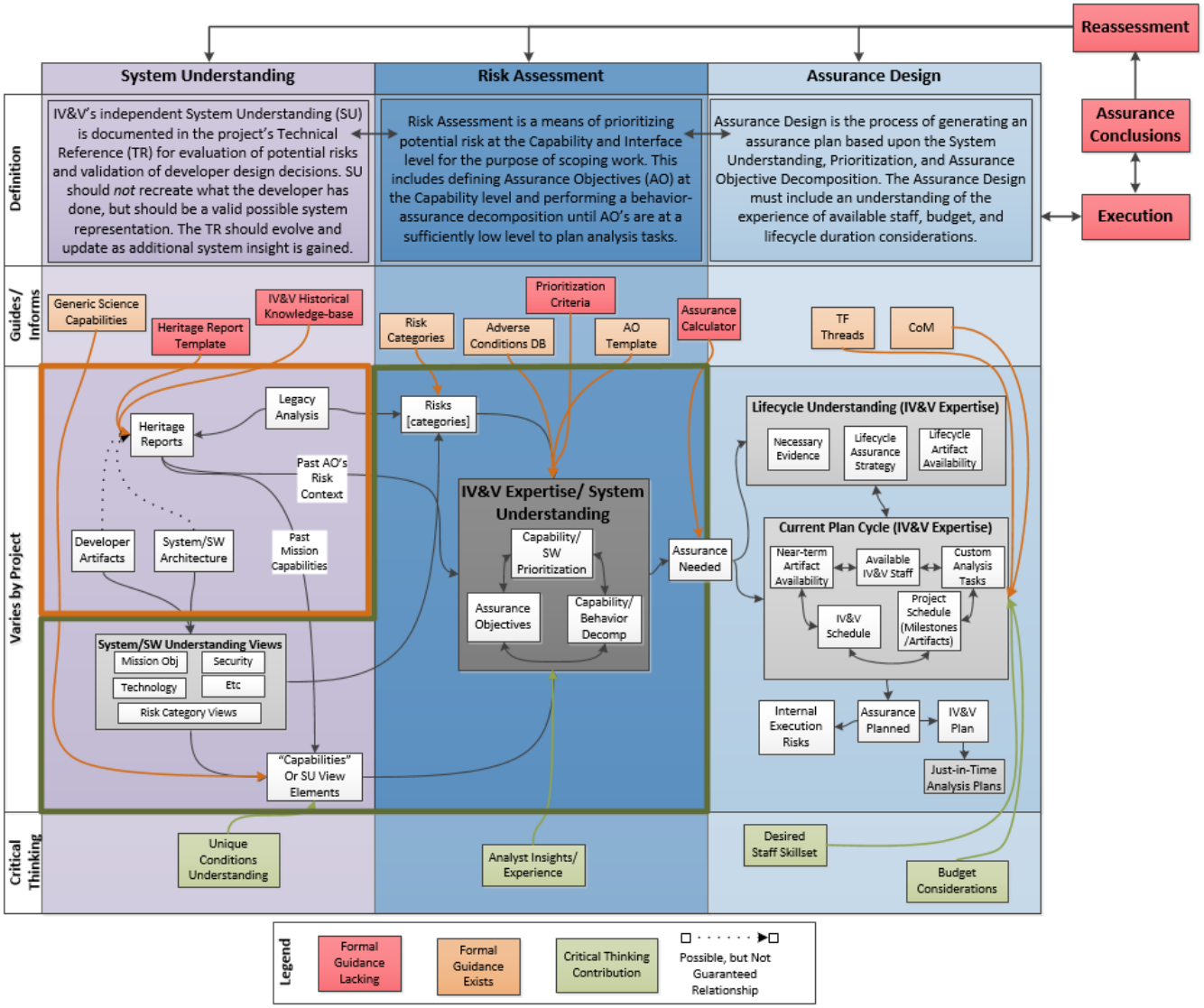


Figure 2: IV&V Planning and Scoping Process - PBRA/RBA Focus

As the project proceeds through its lifecycle, the IV&V team will continually feed any gained understanding and collected analysis evidence back into the PBRA and RBA processes. The feedback of the increased understanding and analytical evidence are key for the IV&V team to have up-to-date understanding of the risk level from both the PBRA and RBA perspectives. That updated risk level understanding then provides the IV&V team with a better basis to update future IV&V focus and analysis planning. While all IV&V teams are encouraged to reassess the PBRA and RBA at least on a semi-annual basis, the teams should reassess and use these tools at any time a new planning effort is scheduled to begin. Ultimately at the end of a project, the IV&V team's goal is to have performed the appropriate type and amount of analysis so that all assessed mission risk at the PBRA capability level is within an acceptable confidence level and can be reported as such to the project at the appropriate milestone reviews.

Definitions

- Assurance Objective – a targeted goal for IV&V assurance that is used to drive IV&V analysis; completed analysis should provide evidence toward the confidence that the target Assurance Objective will be successful and can become an assurance conclusion.
- Capability – the action or reaction of the system desired to satisfy a mission objective; what the system must be capable of doing in order to satisfy mission objectives.
- Limitation – a constraint or condition that can keep a desired action or reaction of the system from occurring, or that can keep a desired action or reaction from occurring in its entirety
    - Results of IV&V provide evidence of limitations in a system's capabilities.

- Relative importance weight – a factor applied to the final risk score *after* the risk assessment. It is derived from the software inventory and is used to differentiate among capabilities that share the same risk score.

Acronyms

AMPL	Agency Mission Directorate Program and Project List
APXS	Alpha Particle X-Ray Spectrometer
CDD	Capability Detailed Description
C&DH	Command and Data Handling
COTS	Commercial Off The Shelf
DAN	Dynamic Albedo of Neutrons
EDL	Entry Descent and Landing
GOTS	Government Off The Shelf
GNC	Guidance Navigation and Control
IBA	IV&V Board of Advisors
IF	Interface
MAHLI	Mars Hand Lens Imager
MARDI	Mars Descent Imager
MBSE	Model Based Systems (or Software) Engineering
OSC	Operational Software Control
P	Performance
PBRA	Project Based Risk Assessment
PCF	Project Category Factor
PS	Personnel Safety
RAD	Radiation Assessment Detector
RBA	Risk Based Assessment
REMS	Rover Environmental Monitoring Station
SAM	Sample Analysis at Mars
SA/SPaH	Sample Acquisition, Processing, and Handling
TS&R	Technical Scope and Rigor

Project Based Risk Assessment (PBRA)

PBRA Process Overview

Figure 3 depicts the PBRA process, which results in a product that supports determination of the IV&V Program portfolio. Subsequent text elaborates the process and expected outputs. Appendix A provides the scoring criteria to support Step 4 of the PBRA process. Appendix C provides candidate examples to support development of the PBRAs.

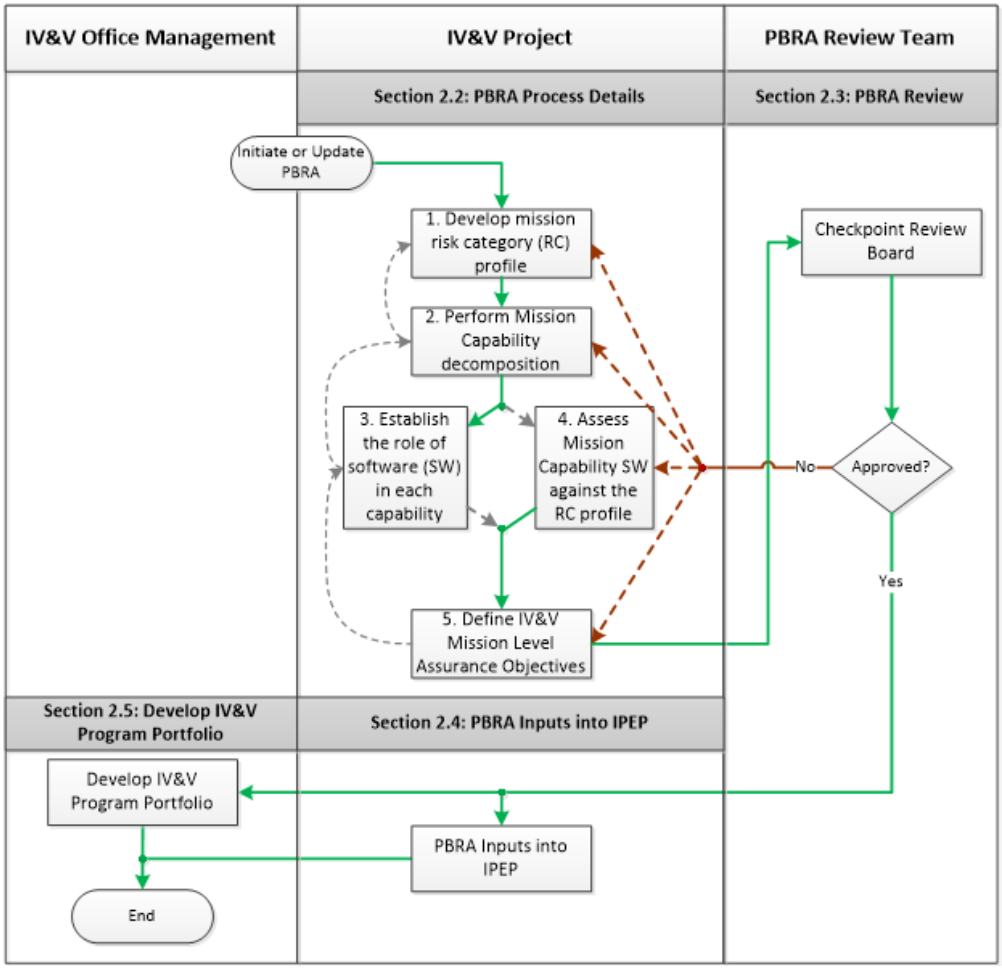


Figure 3: Approach for Supporting IV&V Portfolio Using a Risk-Based Assessment

Notes:

- As demonstrated by the dashed arrows, the process of developing a risk assessment is iterative; information gained in each subsequent step can help to refine previous work as system understanding grows.
- IV&V Office Management is responsible for the entire PBRA process, but may choose to delegate approval authority as appropriate.
- IV&V Projects are responsible for initiating an IV&V community-based PBRA review, though the review itself is performed by non-project personnel, usually through a TQ&E Checkpoint Review
- From a Program standpoint, the PBRA products are reviewed on a biannual basis, as part of the fiscal year and mid-year planning cycles. Additionally, PBRA reviews many be necessary to support IV&V Board of Advisors needs
- From a Project standpoint, the PBRA process is further revisited as needed, e.g. risks, or additional assumptions or questions, identified during IV&V execution, or assumptions and questions understood and addressed as Project matures.

PBRA Development Process

The PBRA process is illustrated in Figure 4. The purpose of this process is to identify the risk in the mission software in its performance of the Mission Objectives. Once risks are identified and assessed, IV&V Assurance Objectives are identified against the most critical items. The process is comprised of five activities. Each activity has defined outputs. The PBRA process is intended as an engineering process, and captures and capitalizes on System Understanding, IV&V Experience and Expertise, and Critical Thinking. This five step PBRA process is iterative in nature, and depending on the mission and assessment team, the steps may be performed in different order or iteratively. As enhanced System Understanding and Risk Position (via Mission development and/or IV&V Analysis), the PBRA is updated and elaborated.

Each of the five activities are defined below, described via a Purpose, Background, Activity Process/Steps, and Outputs. Criteria to support the PBRA prioritization process are provided in Appendix A. Examples products through each step are provided in Appendix C.

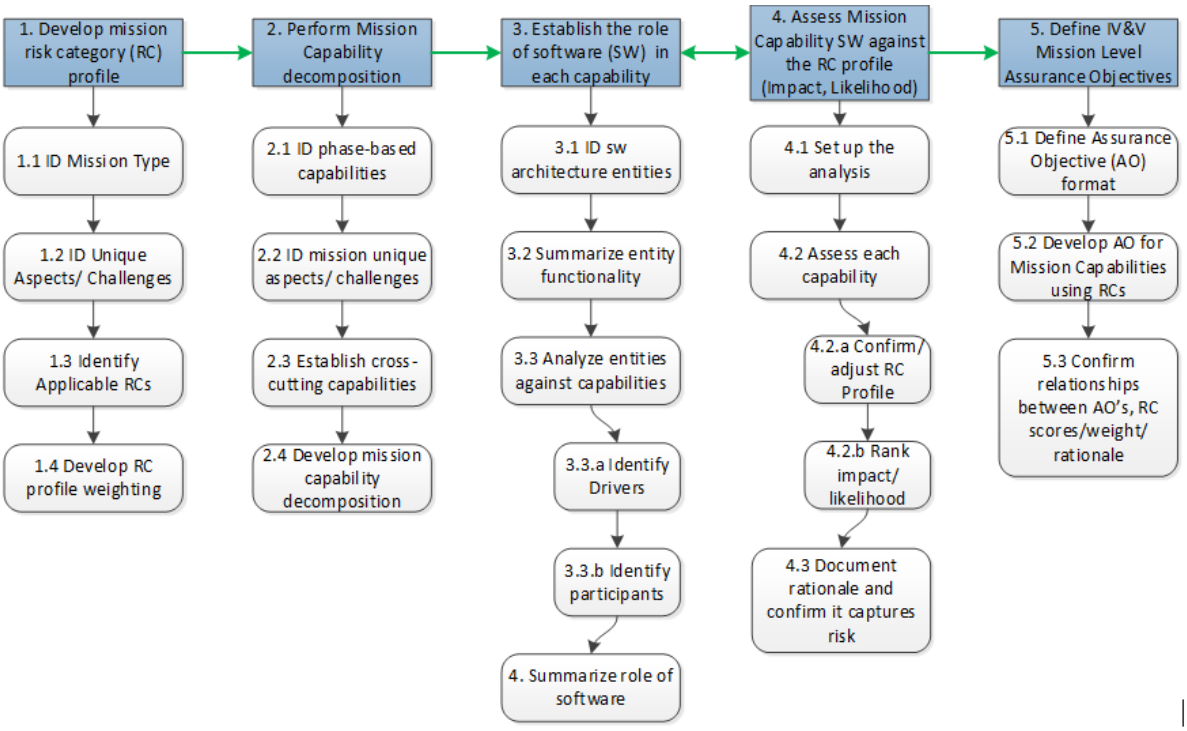


Figure 4: PBRA Process

Step 1: Develop Mission Risk Category Profile and Weighting

Purpose: The "follow the risk" philosophy in efficiently and effectively performing IV&V is based on an understanding of the types of risk that inherently or explicitly apply to the mission. The purpose of this activity is to evaluate the Mission as a black box, and establish the identification and weighting of various risks categories. These risk categories will eventually be assessed against the Mission Capabilities in Step 4 and what will come out of that assessment is the confidence level that IV&V has that each of the Mission Capabilites will be successfully achieved.

Background:

Different types of risks require different types of assurance, and lead to different types of IV&V analysis required to provide assurance. Risks can be based on inherent and explicit factors. For example, candidate inherent and explicit risks drivers are shown in Table 1.

Example: Inherent Risk Drivers	**Example: Explicit Risk Drivers**
• Astrophysics missions have stringent pointing or knowledge requirements, leading to higher performance risk, • Ground systems have more connections external to the system, leading to higher information assurance risk, • Manned missions have humans onboard, leading to higher safety risk, • Deep space missions require more autonomy, leading to higher reliability risk.	• A deep space asteroid mission has a specific Touch-and-Go (TAG) maneuver that is novel, leads to higher functional and performance risk • A key communication link is not encrypted, leading to higher information assurance risk, • High reuse of a complex behavior by the same developer and for the same mission type reduce functional and performance risk • Lack of strong development methodology increases risk

Table 1: Example Implicit and Explicit Risks

To guide the P&S process, the following baseline Risk Categories have been identified. These Risk Categories are identified and defined in Table 2:

Risk Category	**Description**
Complexity	There are a number of complexity-based influences that can affect IV&V's confidence that the Mission Capability can be achieved successfully.

	• From a program integration perspective, the mission might be closely tied to and dependent on other missions or development projects. • From a perspective internal to the project, schedules and processes of possibly multiple developers, including international partners, adversely impact the ability to integrate subsystems/systems adequately. • From an operational perspective, the system may be required to support and/or work in partnership with other operational systems. • From an integration perspective, some Mission Capabilities may be too complex to adequately test in flight-like situations, therefore verification of integration becomes limited to simulation. • Also from an integration perspective, for larger systems, some Mission Capabilities may require a large number of subsystems to work together in order to succeed. This can require the V&V of numerous internal and external interfaces
Robustness	Robustness of the system greatly affects IV&V's confidence that the Mission Capability can be achieved successfully. Having redundant systems that can take over at critical times, systems that have multiple levels of hazard controls, fault tolerance, FDIR, and the possibility that the ground network can intervene as needed increase our confidence. Alternatively, Mission Capabilities that rely on single fault tolerant subsystems, have completely autonomous activity periods, and do not have the option to safe the system will decrease our confidence in success and drive our analysis rigor level to provide more assurance.
Innovation	Innovation in a mission can emerge in many ways and all of them can affect IV&V's confidence that the Mission Capability can be achieved successfully. • The most common is using new technology as part of a mission. An example would be using a new computer processor or new memory storage devices. • Many times new concepts are used to perform Mission Capabilities for a mission. For example, MSL used a new Entry, Descent, and Landing (EDL) concept to successfully land the rover. • Sometimes it is as simple as using an existing development effort in a completely different manner, sometimes even expecting efficiencies. An example here would be reusing a software package for a use that it was not intended for. • Another type of innovation can occur at the development level if a developer chooses to use a different development approach that what they have previously used or even what is used in the software development field as a whole. Examples of this are developers using a new software language, Agile development, and Model Based Systems (or Software) Engineering (MBSE).
Information Assurance	Possibility that confidentiality, integrity, or availability of data can be compromised during mission operation, which can affect IV&V's confidence that the Mission Capability can be achieved successfully. If there is a high threat that a system is vulnerable to attack, our confidence level in that Mission Capability being successful decreases.
Heritage	Past experience with prior similar or predecessor missions can increase or decrease IV&V's confidence that the Mission Capability can be achieved successfully. This includes IV&V heritage on predecessor missions, IV&V experience with the developer, developer experience with similar or predecessor missions, and overall mission experience as a whole, if this is a follow-on mission
Mission Specific	Project defined (as needed)

Table 2: Mission Risk Categories

While these risk categories are identified, these categories are meant to inform the starting point of the risk assessment. In NASA's evolving environment and emergence of new technologies, processes, and operational environments, the risk landscape and risk categories need to also evolve. Many times emerging risks may not be explicitly be identified in the above categories, but with some consideration they may end up implicitly fitting within one of the provided categories. There may be times though that a concern does not fit within the defined categories and should be added as a Mission Specific risk category to assess against. Each project has the responsibility to appropriately define its risk profile. The Program level guidance provided in this standard provides allowances to tailor at the project level. The tailoring is an aspect that will be discussed as part of the review process.

Finally, an important consideration throughout the PBRA process, is the system understanding and considerations of the unique challenges and risks associated with the mission. Unique mission aspects are identified because they frequently are the most risky aspects and thus require the most added assurance. To identify these unique aspects, the team should explore and answer questions such as:

- Is the system performing a familiar function in a novel way?
- Is the system incorporating a new instrument or experiment?
- Is the system going where no spacecraft has gone before?
- Have IV&V Analysis and Findings identified development risk?

Mission Unique Aspects and Challenges help guide the risk assessment at the top level of system understanding, and also through subsequent elaboration and decomposition. The Mission Unique Aspects and Challenges can come from external sources such as project identified challenges, as well as IV&V findings and risks.

The weighting of the risk categories is a step that supports the downstream prioritization of Mission Capabilities as well as prioritization of Assurance Objectives when determining where IV&V analysis will be focused. In Step 4 of the PBRA, the IV&V team will analyze each Mission Capability against the set of risk categories that are defined in this step. From IV&V experience, some of those risk categories tend to drive IV&V focus and rigor more than others. In order to utilize this past experience, applying weighting scores to each risk category, as they are evaluated for each Mission Capability, will provide a clearer picture of where IV&V focus should be placed. The default weighting that was determined for the primary five risk categories is this:

- 30% - Innovation
- 25% - Complexity
- 25% - Robustness
- 10% - Information Assurance
- 10% - Heritage

So as the ultimate goal of the PBRA is to assist the IV&V team in determining risk and confidence levels in the mission success, the risk category weightings informs our scoping decisions by determining which risk categories are likely to have higher impact on our Mission Capability scoring. So using the default scoring above, risks in Innovation will have a higher impact on our Mission Capability score than risks in Information Assurance.

While the default weighting scale is provided to the IV&V team performing the PBRA, it is up to that team to evaluate if they believe that scale is appropriate for the mission being assessed. This scale therefore provides a flexibility that can be applied on a mission by mission basis. In fact, the scaling can even be modified on a Mission Capability by Mission Capability basis. For example, if a Mission Capability has no Information Assurance aspects to it, the scale for Information Assurance can be zeroed out and that 10% can be applied to one or more of the other risk category weights. Additionally, if a new risk category for the mission is created as part of this step, the team will need to allocate part of the overall 100% scaling to that category and adjust the other categories appropriately.

Activity Process/Steps: The Risk Category Profile and Weighting activity uses the concepts described above. The steps to implement the Risk Category Profile and Weighting are as follow:

1. Identify Mission Type
2. Identify Mission Unique Aspects and Challenges (at Mission Black Box level) relating to software
3. Identify any additional Risk Categories
4. Develop weightings of the identified Risk Category Profile

Each step of the process/steps is described in Table 3. Throughout each step, IV&V captures rationale and documents assumptions and questions.

Activity		Description	Typical supporting views (IV&V Tech Reference)
1.	Identify Mission Type	Classify the system into the mission type (e.g. earth observing/orbiting, astrophysics, deep space, lander, orbiter) as well as whether the mission is manned or robotic.	• Mission Type • Whether mission is robotic or non-robotic
2.	Identify Mission Unique Aspects and Challenges (at Mission Black Box level) relating to software	**Capture unique aspects and challenges relating to the mission. Assess potential software contribution to unique aspects and considerations. Capture rationale, questions, and assumptions.**	• Level 1 and 2 requirements • Mission challenges and risks • OpsCon • Heritage information (challenges, risks, as well as successes associated with mission type, developer, and IV&V analysis performed). • IV&V questions, assumptions
3.	Identify additional Risk Categories	Based on steps 1 and 2, identify any risk categories not already defined that apply to the mission. **Capture rationale, questions, and assumptions.**	• Mapping of unique challenges and risks associated with the mission to risk categories • IV&V questions, assumptions
4.	Perform weightings of the identified Risk Category Profile	For the resulting risk category list, perform a weighting of the relative importance of each risk. The sum of the weightings should add to 100%.	• IV&V questions, assumptions

Table 3: Process: Develop Mission Risk Categories Profile and Weighting

Outputs: Table 4 provides the information that results from analysis performed with this step.

	Mission Name	Weighting	Weighting Rationale	IV&V Questions and Assumptions
Risk Categories - General	Complexity	25%	Rationale…mm1	Assumption: aa1… Question: bbb…
	Robustness	25%	Rationale…mm2	Assumption: aa2… Question: bbb…
	Innovation	30%	Rationale…mm3	Assumption: aa3… Question: bbb…
	Information Assurance	10%	Rationale…mm4	Assumption: aa4… Question: bbb…
	Heritage	10%	Rationale…mm5	Assumption: aa5… Question: bbb…
Optional: Risk Categories - Mission Specific	Mission specific A	??%	Rationale…mm7	Assumption: aa7… Question: bbb…
	Missionspecific B	??%	Rationale…mm8	Assumption: aa8… Question: bbb…
	Total	100%		

Table 4: Output: Develop Mission Risk Categories Profile and Weighting

Step 2: Perform Mission Capability Decomposition

Purpose: The Mission Capabilities represent the desired behaviors of the system to satisfy the mission objectives and establish the context for the system software. The mission capabilities are the target of the risk assessment and flow down to IV&V's highest level assurance objectives, which serves as the starting point for all subsequent IV&V analysis intended to add assurance towards mission success. The formulation of solid mission capabilities is a crucial starting point for the IV&V effort. Because mission capabilities exist at a high level, they encompass integrated systems (spacecraft(s), instruments, ground, operator interaction) that may not be wholly accomplished by software.

Ideally, the goal will be that as a program, we will eventually have base sets of Mission Capability decompositions, based on mission types, that can be reused for other IV&V projects. This will create a commanality amongst IV&V projects that will help establish more consistent risk assessments and analysis approaches.

Background: In order to define the necessary capabilities, it is important to understand the following concepts and their role in establishing the Mission Capability decomposition: Phase-based Activities and Cross-Cutting Functionality. These concepts are described in Table 5.

Concepts	Description
Phase Based Activities	**Phase based activities correspond with the distinct timeline of behavioral phases of the mission. These behavioral phases serve as the starting point for defining mission-level capabilities.**
Cross-Cutting Functionality	**Cross-cutting functionality represent system behaviors or services that support many different parts of the mission timeline. Memory management, telemetry collection, downlinking, system fault protection are examples of Cross-cutting functionality. Understanding applicable cross-cutting functionality supports Mission Capability decomposition and enables downstream understanding of the software implications and complexity.**

Table 5: Mission Capability Decomposition Concepts

Mission Capabilities guide the development of risk assessments, the PBRA, and further capability decompositions, which serve as the backbone for Capability Based Assurance efforts. As a result, solid and consistently defined Mission Capabilities are an important starting point and the goal should be to have a set of capabilities that address the entire timeline of the mission. This means that there may be times that you have Mission Capabilities that are occurring at the same time during the mission timeline, but there should never be a period of time within the mission timeline that a Mission Capability does not account for.

Additionally, a consistent use of capabilities provides future projects with a jump start in what the capability decomposition of their project might look like, ultimately helping to ensure that there are no missing Mission Capabilities.

Activity Process/Steps: The Mission Capability decomposition uses the concepts described above. The foundation of the Mission Capability decomposition are the phase based activities. The steps to implement the Mission Capability decomposition are as follow:

1. Identify Mission Phases
2. Identify Mission Unique Aspects and Challenges, for each Phase
3. Develop Mission Capability Decomposition
4. Establish Cross-Cutting Capabilities and roles

In order to fully understand the collective risk within the mission, it is often necessary to perform this elaboration down to the sub-mission phase level and potentially further. Decomposing the mission phases provides more information that will make the risk assessment less subjective and helping to ensure software implementation is fully assessed as it relates to the mission objectives.

Each step of the process/steps is described in Table 6. Throughout each step, IV&V assumptions and questions are captured. IV&V assumptions and questions are about system interfaces, operator interactions, time criticality, autonomy, software implications, etc. Appendix C shows an example of of this process applied to portions of the Deep Space Osiris REx mission.

Activity		**Description**	**Typical supporting views (IV&V Tech Reference)**
1.	Identify Mission Phases (general)	Develop the high level phase based activities for the given mission type. **This set of mission phases is a simple, but inclusive, set from a timeline perspective.**	• Typical mission phases, given the mission type
2.	**Identify Mission Unique Aspects and Challenges, for each Phase**	**Map unique considerations to augment the identified phase-based capabilities. Ensure consistency with the Mission Level (black box) Unique Aspects and Challenges.**	• Level 1 and 2 requirements • Mission challenges and risks • OpsCon • IV&V questions, assumptions
3.	**Develop Mission Capability Decomposition**	**Using the mission phases as a starting point, define the Mission Capability Decomposition. Provide descriptions for each Capability at the top level, including considerations for mission unique aspects and challenges**	• **Table showing Mission Phase to Mission Capability decomposition and description of Capability** • **Model views showing decomposition** • **Within Mission Capability descriptions, unique aspects and challenges/risks, and major cross-cutting capabilities** • **IV&V questions, assumptions mapped to each Capability**
4.	**Establish Cross-Cutting Functionality, and roles**	**Identify the services and supporting behaviors that enable the Phase-based Mission Capabilities to occur. Establish how identified cross cutting functionality support the Phase Based Mission Capabilities**	• **Appropriate elaboration of the cross-cutting functionality** • **Table showing the Mission Capability to Cross-Cutting Functionality decomposition** • **IV&V questions, assumptions**

Table 6: Process: Mission Capability Decomposition

Output: The Mission Capability Decomposition is a table that shows Mission Phases, Mission Capabilities, associated description that incorporates considerations of mission unique aspects/challenges, dominant cross-cutting functionality, and IV&V assumptions and questions (e.g. risks to the assessment).

From a format standpoint, the PBRA activity reflects a progressive and iterative approach, Within this document, the use of green text in the Output charts reflects new information to capture the assessment analysis performed.

Table 7 illustrates the format of the Mission Capability decomposition.

	Mission Phase (augment as desired, e.g. start, end of capability)	**Mission Capability**
1	Mission Phase 1	Mission Capability 1
		Mission Capability 2
		Mission Capability n
2	…	…

Table 7: Output: Mission Capability Decomposition

The Mission Capabilities represent the desired behaviors of the system to satisfy the mission objectives and establish the context for the system's software. Developing a detailed description of each capability is essential as it serves as additional context and rational for the assessment, provides background information for reviewers, captures assumptions made about the capability, and characterizes what it is that we are actually assessing.

A Capability Detailed Description (CDD) can contain a wide variety of information that helps the IV&V team better understand each Capability, and as usual, the more information that can be captured the better. As a minimum, the following information should be gathered to provide at least enough understanding to support Project Initialization efforts, including this risk assessment:

- Description: Provide a parenthetical description of the capability. It is helpful to describe how this capability fits within the mission phases. Describe and elaborate upon any specific scenarios that might exist within the capability.
- Cross-Cutting Functionality: this section documents the list of system functionality that will be utilized to support the successful completion of the Mission Capability as well as a description of how that functionality will be used in the context of the Capability
- Risk Drivers: Describe the concerns or risks which drove the scoring of this capability; this will typically correspond to the highest Risk Category scores. This information serves as a justification for the PBRA score and will help with the generation of the Assurance Objectives.
- IV&V Notes, Assumptions, and Questions: Document any notes, assumptions and questions the IV&V team had while defining the Capability. At the Initialization of a Project, this list could be quite large as there is still much to be learned about the Project. The assumptions and questions will also help determine risk areas that may inform analysis tasks.
- References: List out the this section documents any artifacts that were used for determining the information in the CDD.

A suggested platform for compiling a CDD is Confluence as it allows for easy team collaboration and reference. An example of a CDD, can be found in Step 2: Example - CDD.

The capability decomposition serves two purposes; first, to increase the understanding of each capability; second, to incorporate and understand how cross-cutting functionality is utilized by Mission Capabilities. The depth of the necessary capability decomposition will vary by mission size and complexity. The level of decomposition should be low enough to provide a mission unique level to assess risk against.

Step 3: Establish the role of Software for each Mission Capability

Purpose: The end goal of performing IV&V is to add assurance to the software implementing mission capabilities. An early understanding of how software contributes to each mission capability is necessary to score the capabilities against the risk category profile. This software understanding is documented within the CDD.

Background: Early on in an IV&V Project, especially at the Initialization phase, it is unlikely that a detailed software-level architecture will have been developed yet. As a result of this and for the purpose of the PBRA, it may be necessary to utilize a system level architecture of where software exists in the system, which is the approach that will be discussed in this Step. If a detailed software-level architecture is known at the beginning of the Project, or the IV&V effort has progressed to a point where that architecture is well understood, the approach to performing the more detailed Software Entity evaluation can be found below in Step 2 of the Risk Based Assessment (RBA) section.

The particular software entities contributing to each capability can provide an early indication of the risk inherent in each capability. In order to populate the CDD, an understanding of the mission and developer-provided software architecture is necessary, even if that software architecture is only from a system perspective. In some cases early insight into the software architecture may not be available, and in these instances an IV&V reference architecture may be uses as a placeholder until further information is available. See Table 8 for some considerations regarding software architecture identification.

Architecture Type	Considerations
Heritage	The latest software architecture assessed by IV&V should be used as a starting point. If information is available, the architecture should be "adjusted" for the current mission, including estimations of new software entities driven by unique aspects of the mission. Old instruments should be removed and new instruments should be added.
New or Unknown	If no information is available to estimate a credible software architecture, the generic IV&V reference architecture can be used as a starting point. This reference architecture is not mission specific, so it should be "adjusted" for applicability to the current mission, including any unique aspects.
All	It is important to account for the role of ground software and operator interactions in the overall mission architecture, as mission capabilities will often extend beyond any single development item (spacecraft, etc.). Include COTS and GOTS software as well, if this is even known at the stage of development that exists when performing the initial PBRA.

Table 8: Software Architecture Considerations

Activity Process/Steps: Using the previously defined capabilities and reference or heritage architecture as discussed above, the relevent software entities can be mapped to each capability within the CDDs.

1. Identify the relevant software architecture entities to be assessed.
2. Summarize the functionality of each software entity with a short description.
3. Analyze each capability (or sub-capability) against the total set of identified software entities and describe the role each relevant software entity plays in the accomplishment of the capability, within the CDD.

Each step of the process/steps is described in Table 9. Throughout each step, IV&V assumptions and questions are captured. IV&V assumptions and questions are about software entity breadth, limitations in available information, expected software interactions, boundaries, etc. Appendix C shows an example of a capability and entity mapping using a hypothetical ARM example.

Activity		Description	Typical supporting views (IV&V Tech Reference)
1.	Identify the relevant software architecture entities	"Adjust" a heritage architecture or IV&V reference architecture for the current mission.	• Historical knowledge of similar missions, heritage software • IV&V Reference Architecture • ConOps • IV&V Questions and Assumptions
2.	Describe/ Summarize entity functionality	Understanding of the purpose/functionality of each entity is necessary to help identify questions and assumptions associated with each. This knowledge also assists in mapping the entities to capabilities.	• Software architecture, if available • IV&V developed Reference Architecture • IV&V Questions and Assumptions
3.	Analyze each capability (or sub-capability) against the total set of entities	Denote which software entities contribute to a capability with a short description in the CDD. It is possible for a software entity to have no relationship to a capability. Document the assumptions and unknowns.	• System understanding of similar missions /capabilities • IV&V Questions and Assumptions

Table 9: Process: Establish the role of Software for each Mission Capability

Outputs: There are three products from this activity; each product successively feeds into the subsequent products.

Output #1: Software entities have been identified and summarized. See Table 10.

Entities	Description of Functionality	Assumptions and Questions
Entity 1	Description 1	Assumptions 1
Entity 2	Description 2	Assumptions 2
Entity 3	Description 3	Assumptions 3
…	…	…
Entity N	Description N	Assumptions N

Table 10: Output #1: Description of Software Entities

Output #2: The relationship between capabilities and software entities has been documented within the CDD. Assumptions and questions about the role of each entity within each capability have been documented.

Entity	Role
Software Entity 1 (SW1)	How does SW1 support the completion of the Capability being assessed
Software Entity 2 (SW2)	How does SW2 support the completion of the Capability being assessed

…	…

Table 11: Output #2: Software Role to Mission Capability

For an example of a CDD entry for Software Role to Mission Capability go here: Step 4: Example – Software Role to Mission Capability

Step 4: Analyze each Mission Capability against the risk category profile

Purpose: The purpose of this step is to assess the Risk Categories determined at the mission level, performed in Step 1, against the Mission Capabilities. The assessment is enabled through the activities performed in Steps 2 and 3 where the Capabilities are identified as well as the role of software in those Capabilities. The unique aspects and challenges of the mission have been captured. The ranking of risk against identified Risk Categories at the Mission Capability level provides the basis for the Mission level Assurance Objectives developed in Step 5.

Background: The assessment of the software supporting the Mission Capabilities against the Risk Categories reflects the capability based assurance approach, with a process that follows the risk by adding clarity to the risk that drives an assessment score. Scoring and rationale for mission capabilities (e.g. Launch to Mars, Cruise to Mars, Maintain flight systems, etc.) is a key output of the PBRA process. The output articulates the risk that the Rationale should be unique to the capability and not simply a re-iteration of the provided criteria. Additionally, the rational should clearly specify what it was about the mission capability that resulted in the assigned score. This rationale allows the follow on Assurance Design effort to use analysis Technical Framework items, methods, and threads that specifically address the risk to mission success.

To adequately assess each Mission Capability, the team will produce an Impact score and a Confidence Level score, where the composite Confidence Level score will be determined from the resulting scores assessed for each risk category.

Impact assessment: Impact represents the importance of mission software to the successful achievement of the Mission Capability that is being assessed.

All Mission Capabilities, as defined by Step 2, are critical to performing the mission. If defined correctly, the failure of any mission capability would result in failure of the mission, loss of the asset, and if manned, loss of life. In order to create an Impact score that creates a distinction amongst the Mission Capabilities, to support eventual prioritization, the Impact must be assessed against how critical the mission software is to each Mission Capability.

Criteria for the Impact scoring, on a scale of 1 to 5, can be found in Appendix A.

Confidence Level assessment: Confidence Level represents IV&V's confidence that the Mission Capability being assessed will be developed and implemented completely and correctly and will be successfully achieved operationally.

To determine IV&V's Confidence Level for each Mission Capability defined in Step 2, each Risk Category defined in Step 1 must be first assessed on a scale of 1 to 5 for each Mission Capability. To better support the overall scoring of the Mission Capabilities, the Confidence Level scale is based on the concept that 1 is the highest confidence and 5 is the least confidence.

The criteria for assessing the Confidence Level of each default Risk Category can be found in Appendix A.

Where a mission specific risk is identified, the project team must define criteria for scoring. The project defined criteria typically will address emerging needs for assurance and will be not only used to guide assurance on the project but also provide early understanding of industry and development trends at the program level.

Once each Risk Category has been assessed for the Mission Capability being scored, the IV&V team should evaluate whether the default weighting scale is appropriate for that Mission Capability. If a change to the weighting, as set in Step 1, is determined to be necessary, the weights should be adjusted, making sure their sum is 100%, and a rationale for the scale change should be added to the CDD for that Mission Capability.

After the weighting scale has been assessed, the Confidence Level can then be calculated by using a weighted average approach of the resulting Risk Category scores and the associated weights. The following formula will provide the composite Confidence score:

- ROUND(SUMPRODUCT(riskCategoryScores, riskCategoryWeights))
    - Integer, 1-5

Activity Process/Steps:

The steps to implement the analysis of Mission Capability software against the Risk Categories are as follow:

1. Set up the analysis
2. For each Mission Capability,
    a. Confirm/adjust the risk profile and weighting
    b. Perform ranking for Impact
    c. Perform rankings for Confidence Levels
3. Confirm rationale captures risk across mission

Each step of the process/steps is described in Table 12. Throughout each step, IV&V assumptions and questions are captures. Appendix C shows an example of the output product for a given capability.

Activity		Description	Typical supporting views
1.	Set up analysis	Expand the PBRA progressive analysis sheet (Table 2-12) to incorporate the selected mission level risk profile (Table 2-4). The format of the updated PBRA progressive chart should be in a format as similar to Table 2-14	As needed
2a	**For each Capability, confirm /adjust Risk Profile and Weighting**	**Place the mission level risk profile (Table 2-4) for each Capability. Confirm or adjust risk profiles and weighting to support the Capability. Provide rationale.**	As needed
2b	**For each Capability, perform Impact scoring**	For each capability, score the Impact using criteria provided in Appendix A. Document technical and engineering rationale for each score, clearly explaining scores, as well as questions and assumptions.	As needed
2c	**For each Capability, perform Confidence Level scoring**	For each capability, score the Confidence Level using criteria provided in Appendix A. Document technical and engineering rationale for each score, clearly explaining scores, as well as questions and assumptions.	As needed
3.	**Confirm rationale captures risk across mission**	**Perform an internal peer review of the risk categorization, weightings, scoring, rationale, and IV&V Questions/Assumptions**	As needed

Table 12: Process: Analyze each Mission Capability Software against the risk category profile

Note on Tailoring: Where risk categories were selected that did not have criteria (e.g. Ground Systems, or an emerging risk category relevant to project under consideration), develop scoring criteria consistent with those provided in Appendix A, and score using the generated criteria.

Outputs: Table 13 illustrates the format of the analysis. The items in black are carry-overs from the output of the previous step (**Error! Reference source not found.**). The items in green are updated as a result of the analysis performed in this step.

	Mission Phase (augment as desired, e.g. start, end of capability)	Mission Capability	Scoring Category	Adjusted Weight	Score (1-5)	Rationale (Identified Risk)
1	Mission Phase 1	Mission Capability 1	Impact	-	#	rationale
			Complexity	25%	#	risk rationale a
			Robustness	25%	#	risk rationale b
			Innovation	30%	#	risk rationale c
			Information Assurance	10%	#	risk rationale d
			Heritage	10%	#	risk rationale e
		Mission Capability n	Impact	-	#	rationale
			Complexity	25%	#	risk rationale f
			Robustness	25%	#	risk rationale g
			Innovation	30%	#	risk rationale h
			Information Assurance	10%	#	risk rationale i
			Heritage	10%	#	risk rationale j
2	Mission Phase n	…	…	…	…	…

Table 13: Output: Analyze each Mission Capability Software against the risk category profile

Step 4 also allows the project team to create a ranked list of all the Mission Capabilities, which provides a quick-look at where the priorities of the team may be focused. Since each Capability has a two-dimensional score, with the Impact and composite Confidence scores, the team can use the matrix in Table 14 to determine the resulting Composite Capability Score for each Mission Capability. These Composite Capability Scores can then be used to produce a representation of overall ranking of the full Mission Capability list.

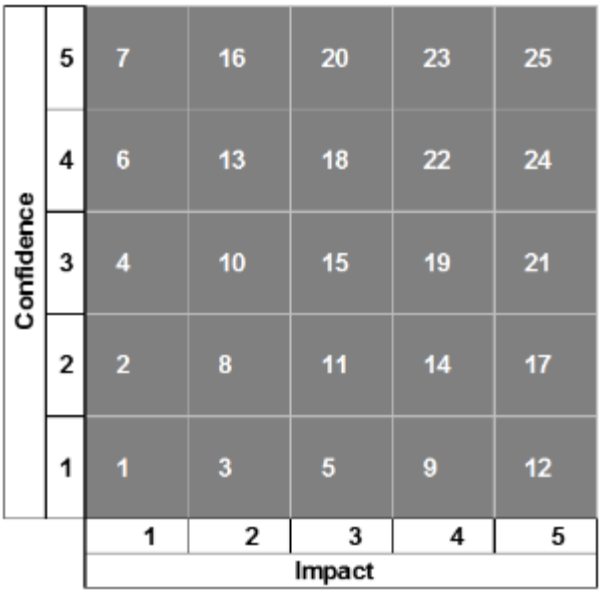


Table 14: Output: Composite Capability Score Using Impact and Confidence Scores

It should be noted here that while it is tempting at this point to use the ranked Mission Capability list to start making focus determinations, Step 5 will start breaking down the Capabilities into Assurance Objectives. That step should be the first step in beginning to make IV&V focus decisions. The ranked Capability list is a high level depiction that can be used to represent to stakeholders where IV&V has the most concern, for instance, it can be used in the IPEP, but it does not provide enough rationale to make the determination that certain Capabilities are ranked low enough to completely scope them out.

Step 5: Define IV&V Mission Level Assurance Objectives

Purpose: IV&V Assurance Objectives provide the basis for subsequent focus and rigor of the IV&V analyses. As a result of the prior steps, the following were developed

- Mission Capability identification, with role of software in performing Mission Capability
- Risk profile, and weighting at mission and Mission Capability levels
- Scoring and rationale for each risk profile for each Mission Capability

These three items provide the basis for risk identification. Assurance Objectives are developed in response to the identified risk in the system software.

Background: Assurance Objectives are the goals that each IV&V project team is striving to achieve with the execution of planned IV&V analysis activities. As the IV&V teams perform analysis and accumulate the evidence towards Assurance Objectives, the Assurance Objectives then can transition to Assurance Conclusions that IV&V can relay as overall completed assurance back to the project. In performing the PBRA process, the goal is to understand where mission risk exists. Assurance Objectives should inherently materialize from those defined mission risks. If IV&V identifies a risk to mission success, there then needs to be one or more Assurance Objectives identified that will help IV&V drive focus and subsequent analysis tasking. For example, if when assessing the Reliability Risk Category for a Capability such as Entry, Descent, and Landing (EDL), a specific risk that might materialize is that there is no possibility for ground or for system fault protection to intervene if something goes wrong during EDL. That risk could then turn into this possible Assurance Objective- “Add assurance that the FSW will perform the EDL activities as planned to mitigate the risk that system will encounter any adverse states that cannot be recovered from.”

Activity Process/Steps:

The steps to implement the analysis of Define IV&V Mission Level Assurance Objectives are as follow:

1. Define Assurance Objective format
2. Develop Assurance Objectives for Mission Capabilities, using Risk Category,
3. Confirm relationships between Assurance Objectives and Risk Category scoring, weighting, rationale

Each step of the process/steps is described in Table 15. Throughout each step, IV&V assumptions and questions are captures. Appendix C shows an example of the output product for a given capability.

Activity	Description	

			Typical supporting views (IV&V Tech Reference)
1.	Define Assurance Objective format	Develop a template for the AOs to be used on the mission's IV&V effort. The AO template should have sufficient detail to provide a basis for Assurance Design, as well as tie back to the risk being assured as part of IV&V analysis	• AO Template and any associated guidance
2	Develop Assurance Objectives	**Develop Assurance Objectives for Mission Capabilities, using AO template. As part of the AO development, consider the following** **- AOs are at the Mission Capability level and address software** **- Magnitude or number of AOs are consistent with weightings and role of software** **- AOs address the identified risk of development or operations**	**As needed**
3	**Peer Review**	**Confirm AO format, and relationships between Mission Capabilities and Software, and Assurance Objectives and Risk Category scoring, weighting, rationale for reasonableness and ensure that IV&V Questions and Assumptions are captured**	As needed

Table 15: Process: Define IV&V Mission Level Assurance Objectives

Outputs: Table 16 illustrates the format of the analysis. The items in black are carry-overs from the output of the previous step (Table 13). The items in green are updated as a result of the analysis performed in this step, specifically the Assurance Objectives as well as IV&V Questions and Assumptions.

	Mission Phase	Mission Capability	Scoring Category	Adjusted Weight	Score (1-5)	Rationale (Identified Risk)	Assurance Objectives that address Identified Risk
1	Mission Phase 1	Mission Capability 1	Impact	-	#	rationale	
			Complexity	25%	#	risk rationale a	AO a1 AO a2 etc
			Robustness	25%	#	risk rationale b	AO b1 etc
			Innovation	30%	#	risk rationale c	AO c1 etc
			Information Assurance	10%	#	risk rationale d	AO d1 etc
			Heritage	10%	#	risk rationale e	AO e1 etc
		Mission Capability n	Impact	-	#	rationale	
			Complexity	25%	#	risk rationale f	AO f1 AO f2 etc
			Robustness	25%	#	risk rationale g	AO g1 etc
			Innovation	30%	#	risk rationale h	AO h1 etc
			Information Assurance	10%	#	risk rationale i	AO i1 etc
			Heritage	10%	#	risk rationale j	AO j1 etc

2	Mission Phase n	…	…	…	…	…	…

Table 16: Output: Define IV&V Mission Level Assurance Objectives

PBRA Review

Once all of the Mission Capabilities have been assessed and the rationale for each has been documented, the results should then be provided to the IV&V Office Lead, IV&V Group Lead, and TQ&E Lead for evaluation and feedback. There is no formal approval or acceptance required for a PBRA, as the expectation is that the PBRA has gone through extensive review internal to the project team during the PBRA development process.

While no formal approval is required for the PBRA, each project will hold a Checkpoint Review (CPR) as it completes project initialization This CPR serves as the setting where the PBRA will be discussed and recorded as well as providing an opportunity for the broader IV&V community to offer feedback on the PBRA product and other initialization products.

Additionally, it is recommended that the IV&V project team meet with their project Point of Contact (POC) to discuss the completed PBRA. This will provide the project the opportunity to provide any clarifications or insight they may have on IV&V's perspective of the project.

Overall, when performing any evaluation or review of a PBRA, Table 17 provides some topics to consider in determining its completeness.

PBRA Process		**Review Considerations**
1.	Develop Mission Risk Category profile	• Explanation of the mission goals and objectives, as well as source document for mission level challenges and risks • Relevant Heritage information and TR documents used • Associated risk profile and weighting with rationale meets Review Board understanding of mission type • Addition or elimination of risk categories is considered and appropriate
2.	Perform Mission Capability Decomposition	• Path which explains how the goals/objectives led to the selected PBRA capabilities • Confirm that Mission Capabilities are at the mission level
3.	**Establish the role of software for each Capability**	• Software role is reasonable, given the Review Board understanding of Developer, Mission Type • Heritage information is brought in appropriately
4.	**Assess Mission Capability Software against the Risk Category profile**	• Tailoring of risk profile and weighting for each Mission Capability is reasonable • Summary of the rationale/scoring for each high level PBRA objective is mission focused, reasonable, and consistent with criteria • Mission risks and challenges to software development/operation is appropriately brought into Mission Capability risk categories/weightings and associated scoring and rationale
5.	**Define Mission Level Assurance Objectives**	• The AO template is articulated and reasonable • Logical ties between the risks associated with PBRA capabilities to the high level assurance objectives • Logical relationship between risk category weightings and AOs, in aggregate

Table 17: PBRA Review Considerations

PBRA Inputs to the IPEP

The PBRA creates two primary inputs into the IV&V Project Execution Plan (IPEP): the mission-level IV&V project risk profile and the Assurance Objectives. The mission-level IV&V project risk profile provides the project with the high level view of the prioritization that IV&V is focusing planned analysis on. The Assurance Objectives subsequently provide the project with a lower level perspective of where IV&V is focusing planned analysis while also providing a view of when they will be in focus by Fiscal Year (FY). The IPEP FY appendices provides the project with IV&V's plan for analysis, review support, and deliverables from a schedule perspective. For the planned analysis by FY, the Assurance Objectives will provide the basis for relaying that focus to the Project.

Develop the IV&V Program portfolio

As the projects that IV&V provides services to vary widely in NASA priority and classification as well as differences in types of projects (Science, Manned, Ground, Earth-orbiting, interplanetary, etc…) it is difficult to solely rely on the PBRA to provide an IV&V Program-level perspective on IV&V resource allocation. The individual project PBRAs do provide the IV&V Program with critical information to support resource decisions. The decision making process must also heavily depend on NASA Standards in determining which projects IV&V performs services on. Beyond that, overall Program fiscal budgets will direct project-level focus and rigor. Additionally, recommendations and directions from the IV&V Board of Advisors (IBA) will affect the overall IV&V direction.

<This marks the end of Phase One (PBRA)>

Risk Based Assessment (RBA)

Figure 5 below depicts the RBA Process.

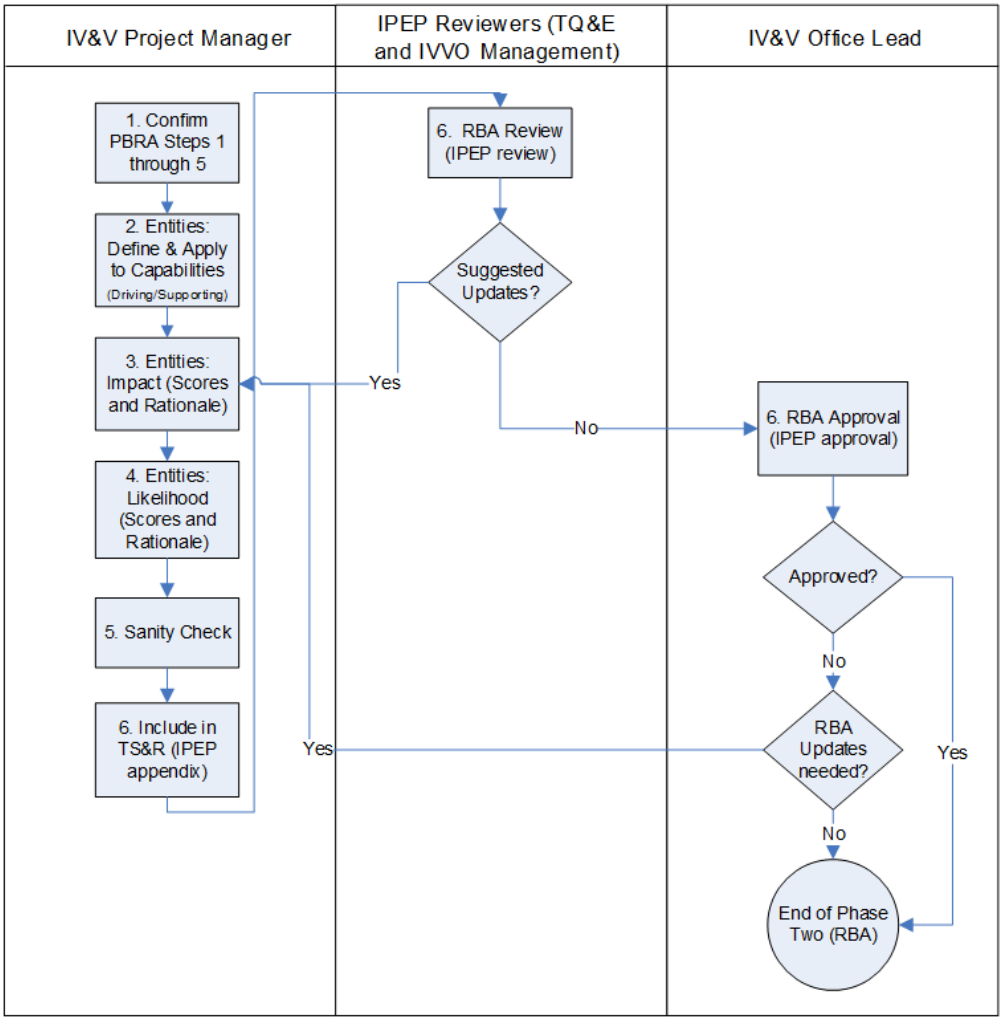


Figure 5: Risk Based Assessment (RBA) Process

Step 1: Confirm the results of steps 1-5 from the (PBRA)

Because new information may have become available to IV&V since these steps were executed, it is important to ensure our results are as accurate as possible. Repeat steps 1-5 from the (PBRA) as necessary.

Step 2: Establish the role of Software for each Mission Capability

Purpose: The end goal of performing IV&V is to add assurance to the software implementing mission capabilities. An early understanding of how software drives and participates in each mission capability is necessary to perform an accurate rating against the risk category profile. The capability and software entity mapping, or C&E, also provides an indication of analysis targets for a capability-based assurance approach.

Background: The particular software entities contributing to each capability, or sub-capability, can provide an early indication of the risk inherent in each capability. In order to perform the C&E mapping, an understanding of the mission and a developer-provided software architecture is necessary. In some cases early insight into the software architecture may not be available, and in these instances it can be a good practice to develop a notional architecture at least at a "software domain" level to establish an initial understanding of the software implications. See Table 18 for some considerations regarding software architecture identification.

Architecture Type	Considerations
Heritage	Heritage architectures may already be understood by IV&V based upon past mission performance. In these cases, the heritage software entities (typically CSCI or CSC level) are understood and a historical perspective on the role, size, limitations, and interactions of each entity is understood. Some thought should be given to how heritage architectures will accomplish new or unique aspects of the mission; it may be helpful to identify explicitly "black box" entities for the unknown unique aspects.
New or Unknown	When past data on the software architecture is unavailable, it can be helpful to "sketch" a notional software architecture at a high level. Notional entities can be defined to encompass broad mission functions at a "domain" level. To increase understanding, it can be helpful to document assumed interfaces and interactions between various entities and mission architecture components. See Appendix C for an example of a notional software architecture.
All	It is important to account for the role of ground software and operator interactions in the overall mission architecture, as mission capabilities will often extend beyond any single development item (spacecraft, etc.). Include COTS and GOTS software as well.

Table 18: Software Architecture Considerations

An example of a representative software decomposition is provided in Appendix C.

Activity Process/Steps: Using the previously defined capabilities and a notional or heritage architecture as discussed above, the software entities can be mapped to each capability (or decomposed capability element) to display the relationship between software and capability.

1. Identify the relevant software architecture entities to be assessed.
2. Summarize the functionality of each software entity with a short description.
3. Analyze each capability (or sub-capability) against the total set of identified software entities; denote drivers and participants.
4. Summarize the role of software in each capability.

Each step of the process/steps is described in Table 19. Throughout each step, IV&V assumptions and questions are captured. IV&V assumptions and questions are about software entity breadth, limitations in available information, expected software interactions, boundaries, etc. Appendix C shows an example of a capability and entity mapping using a hypothetical ARM example.

Activity		Description	Typical supporting views (IV&V Tech Reference)
1.	Identify the relevant software architecture entities	C&E mapping is performed at the mission level, so entities should include ground software, mission operations, COTS, GOTS, etc.	• Historical knowledge of similar missions, heritage software • ConOps • IV&V Questions and Assumptions
2.	Describe/ Summarize entity functionality	Understanding of the purpose/functionality of each entity is necessary to help identify questions and assumptions associated with each. This knowledge also assists in mapping the entities to capabilities.	• Software architecture, if available • IV&V developed Reference Architecture • IV&V Questions and Assumptions

3.	Analyze each capability (or sub-capability) against the total set of entities	Denote which software entities drive a capability and which merely participate in the capability. It is possible for a software entity to have no relationship to a capability.	• System understanding of similar missions /capabilities • IV&V Questions and Assumptions
3. a	Identify Drivers	Driving entities are responsible for the commanding/execution of a capability; these can also be thought of as “active” entities.	
3. b	Identify Participants	Participating entities are responsible for contributing to a capability in some way, but are not primarily responsible for commanding/ execution of a capability. These can also be thought of as “passive” entities.	
4.	Summarize role of software	Use the C&E mapping to determine the role of software in each capability. Some of the following questions can be used for consideration. • What is the extent of automation of this capability? • Is this capability table or sequence driven? • Does this capability require operator interaction? • System understanding of similar missions/capabilities • IV&V Questions and Assumptions	

Table 19: Process: Establish the role of Software for each Mission Capability

Outputs: There are three products from this activity; each product successively feeds into the subsequent products.

Output #1: Software entities have been identified and summarized, which will serve as an input to the software-level risk assessment and inform the C&E mapping. See Table 20.

Entities	Description of Functionality	Assumptions and Questions
Entity 1	Description 1	Assumptions 1
Entity 2	Description 2	Assumptions 2
Entity 3	Description 3	Assumptions 3
…	…	…
Entity N	Description N	Assumptions N

Table 20: Output #1: Description of Software Entities

Output #2: The relationship between capabilities and software entities has been established in table format. See the example below in Table 21. Driving entities have been marked with “X” while participating entities have been marked with “O” in this example. The rationale for the entity’s scoring should explicitly or implicitly refer to this area or areas (e.g. capabilities/behaviors). To help identify driving relationships, ask, “What is the most important thing this entity does?” For example, if Cruise - Power is scored 3-1, and the reason it is scored 3-1 is due to its role in “Establish and maintain power”, then that relationship should be marked with “X”. Similarly, if Rover: C&DH is scored 5-1, and the reason it is scored 5-1 is due to its role in both “Gather engineering and housekeeping data” and its role in “Collect science data”, then both those relationships should be marked with “X”.

Capabilities \ Entities	Entity 1	Entity 2	Entity 3	Entity 4	Entity 5	…	Entity N
Capability 1	O	O	X		X		O
Capability 2	X		X				O
…							
Capability N		O	O	O	O		X

Table 21: Output #2: Mission Capability to Software Entity Relationship

Output #3: The results of Outputs 1 and 2 are summarized in the progressive PBRA development chart in a new column titled “Role of Software” (Table 22), which makes use of the guiding questions defined above.. Additional IV&V assumptions and questions as uncovered by the C&E mapping are also captured.

	Mission Capability	Sub-capabilities	Mission Capability description	Role of Software	IV&V Assumptions and Questions
1.	Phase-based capability	Sub-capability 1 Sub-capability 2 Sub-capability n	Description	Role of software to support Capability (including entity identification)	MC Assumptions and Questions Software Assumptions and Questions
2.	…	…	…	Role of software to support Capability (including entity identification)	MC Assumptions and Questions Software Assumptions and Questions

Table 22: Output #3: Role of Software for each Mission Capability

Step 3: For each entity, assess Impact

Impact represents the relative importance of the capability or entity under evaluation. Impact is a measure of the effect of a limitation or issue within the capability under evaluation (Phase One) or of the result of a failure of the entity under evaluation (Phase Two). Generally, you consider the worst case scenario that is reasonable.

Impact is based on 3 categories, each scored on a scale from 1 to 5. The Impact Score may also be affected by the Project Category identified above in Step #5. The 3 impact categories are as follows:

- Performance
- Personnel Safety
- Operational Software Control

Criteria for these 3 categories can be found in Appendix B, Assessment Criteria. For each entity, score each of the 3 categories. Document technical and engineering rationale for each score, clearly explaining how you reached your conclusions and why a particular value was chosen.

Impact Score algorithm: Impact = (max (PS, (AVG (P, OSC) – PCF)))

PS = Personnel Safety

P = Performance

OSC = Operational Software Control

PCF = Project Category Factor:

Category 1 = 0

Category 2 = 1

Category 3 = 2

Impact Score is calculated as follows:

Take the average score of Performance and Operational Software Control.

If the Project Category is:

Category 1: no change to the result of Step #1.

Category 2: subtract 1 from the result of Step #1.

Category 3: subtract 2 from the result of Step #1.

Take the higher of the result from Step #2” and “Personnel Safety”.

Round to the nearest Integer. The result of this step is the Impact Score.

Step 4: For each Entity, assess Likelihood

Likelihood is assessed to determine the potential for the existence of errors within the Capability (Phase One) or entity (Phase Two) under evaluation.

Likelihood is based on 4 categories:

- Complexity
- Testability
- Degree of Innovation
- Developer Characteristics

Criteria for these 4 categories can be found in Appendix B, Assessment Criteria. For each entity, score each of the 4 categories. Document technical and engineering rationale for each score, clearly explaining how you reached your conclusions and why a particular value was chosen.

Likelihood score algorithm: Likelihood = average (complexity, testability, degree of innovation, development characteristics)

Likelihood score is calculated as follows:

Take the average of the scores from the 4 categories.

Step 5: Perform sanity check

Now that Capabilities and entities have both been scored and relationships have been established and clarified, take the opportunity to evaluate the scoring and rationale to make sure everything seems reasonable.

Include Scoping information in the Technical Scope and Rigor (TS&R) document and IPEP appendix

IPEP review and approval serves as the feedback and approval mechanism for RBA results. IVV 09-4 Project Management is the authority on IPEP review and approval. Current reviewers are the TQ&E Group and IV&V Office Management. Current approver IV&V Office Lead.

Appendix A - PBRA Criteria

Some general notes regarding the assessment criteria found in this appendix:

The intent is not to use the criteria as extremely rigid requirements; instead, the provided criteria are starting points. The intent is promote critical thinking so that each project consistently provides thorough, reasonable, and well-documented scores and scoring rationale.

	Score	Criteria
Impact	1	FSW - At any point during the time period that the Mission Capability is being executed, the FSW does not execute any control of the spacecraft or spacecraft subsystems that could lead to adversely impacting mission success.
	2	FSW - At any point during the time period that the Mission Capability is being executed, the FSW exercises control over the spacecraft or spacecraft subsystems primarily as commanded by the ground system. Additionally, fault management on the spacecraft is primarily limited to monitoring, safing, notifying ground, and waiting for ground direction.
	3	FSW – At any point during the time period that the Mission Capability is being executed, the FSW exercises control over the spacecraft or spacecraft subsystems in such a way that failure of the FSW to operate as expected could cause failure of spacecraft subsystems and ultimately lead to the inability to complete a single mission objective. Additionally, system safing and ground intervention are potential mitigations.
	4	FSW – At any point during the time period that the Mission Capability is being executed, the FSW exercises control over the spacecraft or spacecraft subsystems in such a way that failure of the FSW to operate as expected could cause failure of spacecraft subsystems and ultimately lead to the inability to complete multiple mission objectives. Additionally, system safing and ground intervention are potential mitigations.

	5	FSW – At any point during the time period that the Mission Capability is being executed, the FSW exercises autonomous control over the spacecraft or spacecraft subsystems in such a way that failure of the FSW to operate as expected could cause loss of mission or loss of crew. Additional caveat is that system safing and ground intervention are not available failure mitigations.

	Level of Confidence				
Risk Category	**1**	**2**	**3**	**4**	**5**
Complexity	*Simple and straightforward, nothing distributed and limited interfacing to outside systems* *Simple path to be exercised, input required to stimulate execution path is easily identified and finite, and output easily logged can be automatically compared to success criteria.*	*Relatively basic capability with states and transitions internal to the element. Element capability is realized as a result of combining the behaviors and internal interface information.* *Complex path to be exercised, input required to stimulate execution path is identified but large, and output is compared to success criteria automatically..*	*Relatively basic capability but states and transitions between elements affect the behavior. Capability is realized as a result of combining the behaviors of several objects with the same or similar interfaces.* *One or more paths required to exercise the capability, input required to stimulate execution path may be infinite but easily classified (e.g. equivalence classes), some input dependent on emulators and simulators but not all. Assessing output is fairly straightforward (e.g. some results may require analysis)..*	*Moderately complex, broad engineering community understanding and capability is realized as a result of combining the behaviors across several internal and external interfaces.* *Multiple paths required to exercise the capability, input required to stimulate execution path may be infinite with a few difficult concepts, input is also partially dependent on emulators and simulators. Assessing output is partially dependent on analysis..*	*Very complex, few understand the capability and capability is realized as a result of combining the behaviors of several objects with different interfaces.* *Multiple paths required to exercise the capability, input required to stimulate execution path may be infinite or difficult to conceptualize, input is also entirely dependent on emulators and simulators. Assessing output is entirely dependent on analysis.*
Elaborated Criteria	*Capability is well encapsulated with no significant dependencies and can easily accommodate reasonably expectable interface changes and uncertainties* *Developed more than one like system or current incumbent* *Developer does not use subcontractors and developer staff /management are co-located* *It is feasible to develop a test rig that permits verification of the capability in the lab or on a test range.*	*Capability has minor dependencies on other capabilities and/or changes to interfaces will require minor rework and retest; Operational failure of supporting capability will require recovery or corrective action to prevent mission impact. (Example: External entity requires additional telemetry items or reformat of messages.)* *It is feasible to develop an environment simulation that adequately models the target environment*	*Capability has moderate dependencies on other capabilities and/or changes to one or more external interfaces will require significant rework to allow capability to function; operational failure of supporting capability will impact at least one mission objective (Example: One capability that is described in the Conops is GPS-based attitude determination. Changes to the spacecraft structure can result in changes to the signal structure that will necessitate significant algorithmic revisions)* *It is not feasible to develop an environment simulation to verify the capability, but analysis and modeling tools are available to generate inputs and expected outputs*	*Capability has significant dependencies on other capabilities and/or changes to one or more external interfaces will require redesign of the capability and impose significant schedule and cost consequences; Operational failure of supporting capability will impact multiple mission objectives (Example: Docking system depends on characteristics of host for critical operations)* *Not possible to physically verify capability but similar capability was proven to work correctly. For example, models of a cooperating system are not available, but the proposed capability is based on a currently operational capability and it's believed that works well with existing systems*	*Capability has major dependencies on other capabilities and/or changes to one or more external interfaces will render capability inoperative and require replacement; Operational failure of supporting capability will result in mission loss. (Example: Auto-land capability which requires MLS to function. When MLS is retired, a new landing capability will be needed.* *Example: Changing from IMU with gimbaled gyros to SIGI with strapdown gyros renders the fault management capability inoperative and will require a new fault management capability because the set of external faults it must detect and resolve is different)* *Example is adaptive navigation algorithm for autonomous vehicle for an environment not previously visited*
Robustness	*Failure leaves capability two-fault tolerant*	*Failure of capability reduces fault tolerance and requires intervention of crew or ground personnel to restore fault tolerance*	*Failure of capability results fail-safe status and need for complex recovery procedures to restore fault tolerance*	*Failure of capability leaves the system temporarily zero fault tolerant or in fail-safe mode without available corrective measures*	*Failure of capability leaves system permanently zero fault tolerant*

Elaborated Criteria	*Capability has multiple back-up subsystems or processes that will allow for the completion of the capability even if the prime execution path fails. Ground can intervene, but it would not be immediately necessary for the system to continue to run nominally.*	*Capability has the ability to fail without immediately affecting the overall health of the system. There would be a decrease in fault tolerance due to the failure, but there is a window of opportuinity for crew or ground to respond to keep the system operating nominally.*	*Capability failure will result in fault protection performing a safing action. The system would be temporarily off-nominal, but the failure would not permanently impact the ability to perform the capability in the future. Recovery of the capability would not be autonomously performed by the FSW and would require crew or ground to perform the recovery procedures.*	*Capability failure would result in the system being in an off-nominal state and vulnerable to permanent system or sub-system failure. Due to the activities the system may have been in prior to the failure, the ability to recover from the failure is unknown, but the system would not be in an immediate mission loss type situation. Recovery would require extensive anomaly evaluation and determination of possible recovery activities.*	*Capability failure would result in a situation where the system would likely be lost. FSW would have autonomous control and failure of the FSW to perform the capability would result in mission loss. There would be no possibility of crew or ground intervention.*
Information Assurance	*No effect on Information Assurance*	*The loss of C or I or A could be expected to have a limited adverse effect on mission capabilities, organizational operations, organizational assets, or individuals*	*The loss of C or I or A could be expected to have a serious adverse effect on mission capabilities, organizational operations, organizational assets, or individuals*	*The loss of C or I or A could be expected to have a severe adverse effect on mission capabilities, organizational operations, organizational assets, or individuals*	*The loss of C or I or A could be expected to have a catastrophic adverse effect on mission capabilities, organizational operations, organizational assets, or individuals*
Elaborated Criteria	*Does not cause degradation of mission capability but may cause user inconvenience*	*A limited adverse effect means that, for example, the loss of confidentiality, integrity, or availability might:* *(i) cause a degradation in mission capability to an extent and duration that the organization or mission is able to perform its primary objectives, but the effectiveness of the functions is noticeably reduced;* *(ii) result in minor damage to organizational or mission assets;* *(iii) result in minor financial loss; or* *(iv) result in minor harm to individuals.* *See FIPS-199 for further amplification*	*A serious adverse effect means that, for example, the loss of confidentiality, integrity, or availability might:* *(i) cause a significant degradation in mission capability to an extent and duration that the organization or mission is able to perform its primary objective(s), but the effectiveness is significantly reduced;* *(ii) result in significant damage to organizational or mission assets;* *(iii) result in significant financial loss (Ex: Loss of sensitive data (Intellectual Property), Incurred recovery costs after incident); or* *(iv) result in significant harm to individuals that does not involve loss of life or serious life threatening injuries (Should also be classified under Personnel Safety if the security of it can affect human life).* *See FIPS-199 for further amplification*	*A severe adverse effect means that, for example, the loss of confidentiality, integrity, or availability might:* *(i) cause a severe degradation in mission capability to an extent and duration that the organization or mission is not able to perform one or more of its primary objectives; (Ex: Marginal loss of agency's reputation, Loss of multiple mission objectives)* *(ii) result in major damage to organizational or mission assets;* *(iii) result in severe harm to individuals involving loss of life or serious life threatening injuries (Should also be classified under Personnel Safety if the security of it can affect human life).* *See FIPS-199 for further amplification*	*A catastrophic adverse effect means that, for example, the loss of confidentiality, integrity, or availability might:* *(i) cause a loss of mission capability to an extent and duration that the organization or mission is not able to perform its primary objectives; (Ex: Substantial loss of agency's reputation, Loss of asset, Loss of all primary objectives(s), Loss of sensitive data)* *(ii) result in critical damage to organizational or mission assets;* *(iii) result in major financial loss (Ex: Loss of sensitive data (Intellectual Property), Incurred recovery costs after incident); or* *(iv) result in catastrophic harm to individuals involving loss of life or serious life threatening injuries (Should also be classified under Personnel Safety if the security of it can affect human life).* *See FIPS-199 for further amplification*
Innovation	*Capability has been developed before by this team and has flown on several missions*	*Capability has flown several missions, but has been developed by another team*	*Capability has flown before, fairly mature and well know, but is being modified for mission*	*Capability has flown only one mission, but is modified based on data from that mission*	*Capability is being proven on mission and limited experience in developing like-capability*
Elaborated Criteria	*• Proven on other systems with same application*	*System prototype demonstration in a space environment -or- actual*	*System/subsystem model or prototype demonstration in a relevant environment.*	*Component and/or breadboard validation in a laboratory environment -or- Component*	*Basic principles observed and reported -or- Technology concept*

	• Mature experience *• Well documented testing* *• Solid requirements - little potential for change* *• Little to no integration required* *• No interaction with multiple organizations* *• Actual system "Flight Proven" through successful mission operations.*	*system complete and "Flight Qualified" through test and demonstration (ground or space).*		*and/or breadboard validation in relevant environment.*	*and/or application formulated -or- Analytical & Experimental critical functions and/or characteristics proof-of-concept.*
Heritage	*IV&V has been performed on this Capability as it was developed by this Developer.*	*IV&V has been performed on a similar Capability before and the Developer has developed this Capability before.*	*IV&V has not been performed on a simliar Capability before but the Developer has developed this Capability before.*	*IV&V has been performed on a similar Capability before but the Developer has not developed this Capability before.*	*IV&V has not been performed on a simliar Capability before and the Developer has not developed this Capability before.*
Elaborated Criteria	*IV&V has been performed on a previous version of the mission using this same capability or on a similar mission that used this same capability as it was developed by this developer.*	*The capability is similar to ones that IV&V have assessed before and the developer has familiarity with the capability, even if IV&V has not assessed this specific capability as developed by the current developer.*	*The capability concept is new to IV&V analysis, but the developer has developed this capability before on a project(s) which IV&V was not performed.*	*IV&V has familiarity with the concept behind the capability and has performed analysis on it from a different developer, but the current project developer has never incorporated this capability into any of their prior projects, so it is new to them.*	*The Capability is completely new to both IV&V and the current developer, so there is no basis for confidence at the beginning of the project.*
Project Specific	-	-	-	-	-
Elaborated Criteria					

Appendix B - RBA Criteria

Some general notes regarding the assessment criteria found in this appendix:

- The intent is not to use the criteria as extremely rigid requirements; instead, the criteria are starting points. The intent is to consistently provide thorough, reasonable, and well-documented scores and scoring rationale.
- Two main factors are assessed: Impact and Likelihood
    - Impact criteria are below in the first table
    - Likelihood criteria are below that and spread across three tables
- Several of the categories within Impact and Likelihood have “elaborated criteria”. The basic criteria come almost entirely from the original PBRA process, released in December of 2008, and are often high level. “Elaborated criteria” (along with the RBA processes) were produced by an assessment team in March 2010, and serve as additional content that evaluators may find helpful when assessing lower level entities.

Impact	Very Low (1)	Low (2)	Moderate (3)	High (4)	Very High (5)
Performance	Minimal or No Mission Impact	Minor Impact to Full Mission	Moderate Impact to Full Mission	Major Impact to Full Mission	Loss of Minimum Mission Objectives
Elaborated Criteria	• Failure could cause an inconvenience but no impact to mission success, science value, or cost of operation.	• Reduced system performance that does not result in the loss of a mission objective. • Reductions could include short term loss of science collection, implementation of workarounds with minimal cost, or noticeable but minor impact to science value.	• Loss of a single mission objective (mission success or mission return) or degradation in operational performance. • Minimum mission success criteria are met. • Performance degradation may result in costly recovery options, reduced science value, or long term delays to the accomplishment of science.	• Loss of multiple mission objectives • Some science value is retained • Damage to other spacecraft/assets	• Permanent loss of all mission objectives. Examples: - Loss of spacecraft - Loss of other spacecraft/assets - Loss of ability to collect science data - Loss of primary instrument
Personnel Safety[1]	No Injury	Minor Injury/Illness (ref. 8621.1B Type D)	Lost Time Injury/Illness (ref. 8621.1B Type C)	Permanent Partial Disability (ref. 8621.1B Type B)	Death, Permanent Total Disability (ref. 8621.1B Type A)
Elaborated Criteria	NA	NA	NA	NA	NA
Operational Software Control[2]	Software does not control safety-critical hardware systems, subsystems or components and does not provide safety-critical information.	Software does not control safety-critical hardware systems, subsystems or components and does not provide safety-critical information. However, software resides within a computing device such that failure of the device has the potential for a Level 3 performance impact.	Software item issues commands over potentially hazardous hardware systems, subsystems or components requiring human action to complete the control function. There are several, redundant, independent safety measures for each hazardous event. Software generates information of a safety-critical nature used to make safety-critical decisions. There are several redundant, independent safety measures for each hazardous event. Software does not control safety-critical hardware systems, subsystems or components and does not provide safety-critical information. However, software resides within a computing device such that failure of the device has the potential for a Level 4 or 5 performance impact.	Software exercises control over potentially hazardous hardware systems, subsystems, or components allowing time for intervention by independent safety systems to mitigate the hazard. However, these systems by themselves are not considered adequate. Software item displays information requiring immediate operator action to mitigate a hazard. Software failures will allow, or fail to prevent, the hazard's occurrence.	Software exercises autonomous control over potentially hazardous hardware systems, subsystems or components without the possibility of intervention to preclude the occurrence of a hazard. Failure of the software, or a failure to prevent an event, leads directly to a hazard's occurrence.
Elaborated Criteria	NA	NA	NA	NA	NA

[1] 8621.1B effective date May 23, 2006 Chapter 1, Figure 1

[2] "Operational Software Control" is based almost entirely on the "Software Control Categories" found in NASA Software Safety Guidebook (NASA-GB-8719.13), Table 3-1 MIL STD 882C Software Control Categories. Content was modified to shift from a 4 point scale to a 5 point scale, and to account for software that resides within a computing device such that failure of the device will lead to a certain level performance impact.

Likelihood	Very Low (1)	Low (2)	Moderate (3)	High (4)	Very High (5)
Complexity	Simple and straightforward, nothing distributed and limited interfaces.	Relatively basic capability with states and transitions internal to the element. Element capability is realized as a result of combining the behaviors and internal interface information.	Relatively basic capability but states and transitions between elements effect the behavior. Capability is realized as a result of combining the behaviors of several objects with the same or similar interfaces.	Moderately complex, broad engineering community understanding and capability is realized as a result of combining the behaviors across several internal and external interfaces.	Very complex, few understand the capability and capability is realized as a result of combining the behaviors of several objects with different interfaces.
Elaborated Criteria	Straight-line code with few to no nested structured programming operators: DOs, CASEs, IF THEN ELSEs. Simple module composition via procedure calls or simple scripts. Simple read-write statements with simple formats. Simple COTS-DB queries and updates. Function operates in only one more of system operation. Evaluation of simple expressions.		Simple nesting with some inter-module control including decision tables, message passing and middleware supported distributed processing. Simple I/O processing including status checking and error processing. Multi-file input or single file input with minimal structural changes to the files. Function behaves differently in different modes of system operation. Standard math and statistical routines to include basic vector operations.		Multiple resource scheduling with dynamically changing priorities or distributed real-time control. Performance critical embedded system. Highly coupled dynamic relational and object structures. Object uses different end items (sensors) in different modes (stages) of system operation. Difficult and unstructured numerical analysis: highly accurate analysis of noisy, stochastic data and/or complex parallelization.

Likelihood	Very Low (1)	Low (2)	Moderate (3)	High (4)	Very High (5)
Testability	Simple path to be exercised, input required to stimulate execution path is easily identified and finite, and output easily logged can be automatically compared to success criteria.	Complex path to be exercised, input required to stimulate execution path is identified but large, and output is compared to success criteria automatically.	One or more paths required to exercise the capability, input required to stimulate execution path may be infinite but easily classified (e.g. equivalence classes), some input dependent on emulators and simulators but not all. Assessing output is fairly straightforward (e.g. some results may require analysis).	Multiple paths required to exercise the capability, input required to stimulate execution path may be infinite with a few difficult concepts, input is also partially dependent on emulators and simulators. Assessing output is partially dependent on analysis.	Multiple paths required to exercise the capability, input required to stimulate execution path may be infinite or difficult to conceptualize, input is also entirely dependent on emulators and simulators. Assessing output is entirely dependent on analysis.
Elaborated Criteria	A scriptable interface or test harness is available. Software and hardware states and variables can be controlled directly by the test engineer. Software modules, objects, or functional layers can be tested independently (low level of coupling). Test expectations are fully quantified. Past system states and variables are visible or queriable (e.g., transaction logs). Current system states and variables are visible or queriable during the execution. Distinct output is generated for each input. System states and variables are visible or queriable during execution. All factors affecting the output are visible. Incorrect output is easily identified. Internal errors are automatically detected and reported through self-testing mechanisms. Module can be fully tested via inspection.	Tests are written before coding is performed. Testing is not wholly independent, but only 1 or 2 other interfaces are required. The majority of system states and variables are visible or queriable during execution. Internal errors are automatically detected but requirement manual correction (no self-testing mechanism).	Software and hardware states can be influenced or indirectly controlled by the test engineer. Not all factors affecting the output are visible. Module is not singular in responsibility, i.e., mid-level cohesiveness. Determination of the correctness of the output may require some limited analysis. Test expectations are available, but may not be fully documented. Testing of the module is dependent on a limited number of other modules (mid-level coupling).	Partial visibility of past system states and variables. Partial insight into the current state of the module/system component during testing. Testing through demonstration is acceptable. Some test expectations are non-quantifiable. Testing is reliant on multiple interfaces, many simulated in order to execute the software.	Testing is not considered until coding is complete. Software and hardware states cannot be directly controlled by the test engineer. Software module cannot be independently tested (high level of coupling) without multiple simulated interfaces. Past system states and variables are not visible. Generated output cannot be directly derived from the provided input. Incorrect output is not easily identified - requires manual analysis. Low cohesiveness. Test expectations are unknown or non-quantifiable.

Likelihood	Very Low (1)	Low (2)	Moderate (3)	High (4)	Very High (5)
Degree of innovation	Capability has been developed before by this team and has flown on several missions	Capability has flown several missions, but has been developed by another team	Capability has flown before, fairly mature and well know, but is being modified for mission	Capability has flown only one mission, but is modified based on data from that mission	Capability is being proven on mission and limited experience in developing like-capability
Elaborated Criteria	• Proven on other systems with same application • Mature experience • Well documented testing • Solid requirements - little potential for change • Little to no integration required • No interaction with multiple organizations • Actual system "Flight Proven" through successful mission operations.	System prototype demonstration in a space environment -or- actual system complete and "Flight Qualified" through test and demonstration (ground or space).	System/subsystem model or prototype demonstration in a relevant environment.	Component and/or breadboard validation in a laboratory environment -or- Component and/or breadboard validation in relevant environment.	Basic principles observed and reported -or- Technology concept and/or application formulated -or- Analytical & Experimental critical functions and/or characteristics proof-of-concept.
Development Characteristics	Developer uses a mature engineering approach and makes use of a documented and tried process (industry wide or local)	Developer uses new engineering approaches which are documented and followed	Developer has a mature process planned and evidence suggest that the planned processes are not being followed	Developer has a mature engineering process planned but actual implementation of the process is incomplete and ad hoc engineering is completing them	Developer's engineering approach is ad hoc with minimal documentation as well as planning
Elaborated Criteria	Developed more than one like system or current incumbent Developer does not use subcontractors and developer staff/management are co-located	Developed one like system Developer does use subcontractor(s) and developer staff/management are co-located	Nominal domain or related experience (10+ years) Developer does not use subcontractor and developer staff/management are not co-located	Some domain or related experience (5-10 years) Developer uses one subcontractor and management/staff that are not co-located (i.e., geographically dispersed)	Minimal domain or related experience (less than 5 years) Developer uses multiple subcontractors and management/staff that are not co-located (i.e., geographically dispersed)

[1] NASA Software Safety Standard (NASA-STD-8719.13B)

Appendix C – Examples and Guidance Relating to PBRA Process Steps

Step 2: Capability Decomposition Example

Table 23 provides an example of following each of these steps against a robotic Deep Space mission.

<table>
<tr><th>Step</th><th>Title</th><th>Example</th></tr>
<tr><td>1.</td><td>Identify Phase Based Capabilities (general)</td><td>Start with the “generic” set of Phase-Based capabilities for science missions.
<table><tr><th>Phase-Based Capabilities</th></tr><tr><td>Launch and establish spacecraft</td></tr><tr><td>Travel to science objective</td></tr><tr><td>Arrive at science objective</td></tr><tr><td>Primary mission operation</td></tr><tr><td>Return science</td></tr><tr><td>Decommission spacecraft</td></tr></table>This generic set of capabilities should then be tailored for applicability to the current mission. Consider the JUNO mission, where the primary means of returning science is to downlink data to Earth. In this case, there is nothing distinct or unique about the “Return Science” capability. So this phase-based capability should be removed and the future decomposition of “Primary Mission Operation” should account for the ability to downlink data to Earth.</td></tr>
<tr><td>2.</td><td>Identify Mission Unique Aspects and Challenges</td><td>For example, the OSIRIS-REx mission had two distinct aspects of the “primary mission operation” phase-based capability, as shown below.
<table><tr><th>Phase-based Capability</th><th>Unique Mission Aspects (O-REx example)</th></tr><tr><td rowspan="2">Primary Mission Operation</td><td>Orbital asteroid characterization</td></tr><tr><td>Collect sample from asteroid surface</td></tr></table>The mission objective to acquire a sample from the surface of the asteroid was distinct and unique for the OSIRIS-REx mission and thus worth differentiating from the capability to characterize the surface of the asteroid from a distance using a variety of science instruments.

Note that in some cases these unique aspects may be most appropriately considered as a decomposition of a phase-based capability.</td></tr>
<tr><td>3.</td><td>Establish Cross-Cutting Capabilities</td><td>Cross-cutting capabilities, or services, are essential to the successful operation of the mission and will often arch over many different domains or subsystems. For example, “Perform Fault Management” and “Transmit and Receive commands” are ever-present services among all spacecraft. These services are necessary to accomplish multiple phase-based behaviors and also heavily software relevant, but support rather than define mission capabilities. Applicable cross-cutting capabilities, such as these, must be identified and incorporated into the phase-based capabilities as part of the capability decomposition effort.</td></tr>
<tr><td>4.</td><td>Develop Mission Capability Decomposition</td><td>In the example below, the “Primary Mission Operation” capability for OSIRIS-REx is decomposed, taking into account the unique aspects of sample return, and cross-cutting capabilities of “Perform Fault Management” and “ Transmit and Receive Commands”.

The initial phase-based capability:
<table><tr><td>Primary Mission Operation</td></tr></table>Accounting for the mission’s unique aspects, this capability is split into two:
<table><tr><td>Orbital asteroid characterization</td></tr><tr><td></td></tr></table></td></tr>
</table>

		Collect sample from asteroid surface
		The above identified cross-cutting capabilities are integrated into the sub-capability decomposition:
		Orbital asteroid characterization
		Science sequence uplink and validation
		Microgravity proximity delta-v operations
		Flyby science slew sequences using OCAMS and OTES instruments
		Storage of science data for downlink to earth
		Science data downlink during DSN windows
		Ephemeris processing to determine DSN pointing
		Telecom component selection based on HW State Table
		Fault management to prevent sun on instrument deck during slews
		Collect Sample from asteroid surface
		TAG sequence and uplink and validation
		Articulate sampling arm
		Checkpoint/ Matchpoint calculations and delta-v adjustment
		Surface Contact logic and sample collection
		IMU contact detection
		Arm and execute pyros for N2 sample collection
		Execute surface back away delta-V after delay
		Sample mass verification and stow operations
		Emergency sampling attempt abort
		Note that some sub-capabilities were decomposed to an additional level – the level of decomposition can vary depending upon the level which is necessary to adequately describe the capability.

Table 23: Example: Mission Capability Decomposition Steps/Activities

Generalized Capability Capability Guidance

This guidance was developed as a starting point for interplanetary Science Missions to assist in the generation of mission-level capabilities. This guidance was based heavily on lessons learned from the Mars2020 and OSIRIS-REx IV&V deep-space projects. The capabilities have been tailored to apply to earth science, and manned missions. Thoughts relating to ground systems are also provided.

Mission-level capabilities guide the development of capability risk assessments (PBRA) and further capability decompositions, which serve as the backbone for Capability Based Assurance efforts. As a result, solid and consistently defined capabilities are an important starting point. Additionally, a consistent use of capabilities provides future projects with a jump start in what the capability decomposition of their project might look like, ultimately helping to ensure that there are no missing system-level capabilities.

Step 2: Phase Based Capabilities

Science Missions are typically structured into distinct behavioral phases which coincide fairly well with a pattern. These behavioral phases should serve as the starting point for defining mission-level capabilities. Note that developers often define "Mission Phases" which can be a helpful place to begin. The below "generic" capabilities include some potential sub-capability elaborations, which can vary by science mission type.

Generic Capability	Sub-Capabilities
Launch and Establish Spacecraft	• Launch spacecraft into Orbit (if SC has applicable functionality) • Separate from LV • Deploy Solar Arrays, Antennae, Instruments, etc. • Power Positive/Thermally Stable Attitude, possibly considered Commissioning Spacecraft • Communicate with Ground (**if unique**)
Travel to Science Objective	• Attitude Control (pointing) and main engine/TCM burns • Planetary Gravity Assists • Orbit Changes • Cruise / extended autonomy and maintenance (**if unique**) • Science instrument operations, if necessary

Arrive at Science Objective	• Orbit Insertion
	• Aerobraking
	• Entry-Descent-Landing (EDL)
	  ○ Preparation for Atmospheric Entry
	  ○ Passive Descent Activities
	  ○ Active Descent Activities
	  ○ Landing Activities
	• Science instrument operations, if necessary
	• S/C Health and Maintenance Activities

Primary Mission Operation	• Orbital maneuvers/ novel orbits
	  ○ Microgravity proximity operations
	• Deployment/Commissioning, if necessary
	• Landed motion for Surface Missions
	  ○ Ground-commanded
	  ○ Autonomous
	• Flyby science sequences/slews for Orbiting Missions
	  ○ Active Science Collection
	  ○ Passive Science Collection
	• Sample collection and curation
	• S/C or Rover Health and Maintenance Activities

Return Science	• Return travel to Earth • Data/Telemetry Downlink (**if unique**)
Decommission Spacecraft	• Decommissioning Orbit/Burn • Graceful Shutdown • Science instrument operations, if necessary • Considerations for extended mission operations

Table 24: Science Mission Capability Decomposition Guidance

This list is non-exhaustive, but representative of a typical pattern that science missions follow. Some of these capabilities may not contain any risky behaviors worth adding additional assurance and therefore may fall out of focus when performing the mission-level risk assessment. Discretion should be used in selecting from this generic list.

When defining mission-level capabilities for a project, the goal should be to have a set of capabilities that address the entire timeline of the mission. This means that the capabilities will vary from very specific time periods, like "EDL", to more broad time periods with very little activity occurring, like "S/C Health and Maintenance" during a Cruise phase. Additionally, capabilities may overlap in occurrence as well. For example, "S/C Health and Maintenance" during Cruise is occurring for most of the phase, even if TCMs are occurring as well.

Capabilities at the mission level do not have to exist at the same hierarchical level; for example, "Active Science Collection" is a subset of "Science Instrument Operations". The goal of defining the capabilities at the mission level is to be able to adequately understand the riskiest aspects of the mission. so, in the case of the previous example, the natural decomposition from the Primary Mission Operation general capability is "Science Instrument

Operations", but to adequately understand the risk you might need to decompose one step further. The mission may have science collection from instruments that are fixed to the spacecraft/rover and may not be as risky to mission success as science collection that relies on vision sub-systems and a robotic arm to successfully perform science collection.

Step 2: Unique Behaviors

Consideration should be given to the unique behaviors of each mission, which can deviate from the generic set of capabilities above. Oftentimes, the most unique and novel aspects of a mission are the most risky and thus require the most added assurance. For example, on OSIRIS-REx, the capability to "Acquire Abundant Regolith" from the surface of the asteroid was separate from the ability to collect orbital science (Primary Mission Operation), as the behavior was uniquely risky and warranted additional scrutiny and analysis.

Step 2: Cross Cutting Functionality

A Cross Cutting Functionality tends to support multiple mission-level capabilities. An example of this is uplinking commands from the ground, this functionality will support performing TCMs, S/C maintenance, performing science, mobility, etc… If a functionality is being used to support multiple mission capabilities, it should not be considered a mission capability, it is best included as part of decomposing the mission capabilities into operational scenarios.

Cross-Cutting Functionality Examples:

Maintain Health and Safety of Systems and Instruments

Transmit, Receive, and Process Commands and Telemetry

The OSIRIS-REx mission capabilities, as documented in the project's PBRA, used two "cross-cutting" capabilities (shown above). Below, these capabilities are instead dispersed and integrated as sub-capabilities of the more generic phase-based Capability approach.

Generic Capability	OSIRIS-REx Capability	Sub-Capabilities

Launch and Establish Spacecraft	Establish an Independently Operating Spacecraft	• Separate from LV
		• Power Positive/Thermally Stable Attitude
		  ○ **Achieved by Safe Mode**
		  ○ **Attitude Knowledge Fault Management**
		  ○ **SC Sequence Aborts to accommodate critical activities**
		• **Communicate with Ground**
		  ○ **Comm (uplink/downlink) checkout**

		○ **Commands/sequencing checkout**
Travel to Science Objective	Travel to Asteroid 1999 RQ36	• Attitude Control (pointing) and main engine burns ○ **Attitude Control Fault Management** • Trajectory Correction Maneuvers • Planetary Gravity Assists • Cruise / extended autonomy and maintenance ○ **Autonomous Fault Recovery** ○ **Heartbeat termination and side swap** ○ **SC health and status data telemetering**
Arrive at Science Objective	[Captured in "Travel..."]	• Orbit Insertion delta-V maneuver
Perform Science	Characterize Asteroid and Determine Site Selection for Sample Collection Acquire Abundant Regolith	• Microgravity proximity operations • Orbital maneuvers/ novel orbits • Science instrument operations ○ **Science Instrument sequence generation** ○ **Science sequence validation and execution** ○ **Science data telemetering** ○ **Science data downlink** ○ **Communication outage fault management** • Sample collection and curation ○ **Receive, validate, execute TAG sampling sequence** • **Fault Protection of Sample Attempt, TAG triggers** ○ **TAG abort triggers** ○ **TAG Trigger**
Return Science	Return Sample Collections to Earth	• Return travel to Earth - Main Engine Burn • **Data/Telemetry Downlink** • Sample Return Capsule release ○ **Critical Event Recovery (fault protection)**

Decommission Spacecraft	[Divert maneuver to solar parking orbit captured in "Return..."]	• Decommissioning Orbit/Burn • Considerations for extended mission operations

Table 25: OSIRIS-REx Cross-Cutting Capability Examples

Step 2: Example - CDD

Description

The Flight Operations Team (FOT), Observatory (i.e. Space Segment) and Ground System collaborate to maintain Observatory Health and Safety. The Mission Capability to maintain observatory health and safety is described by the following Operational Scenarios in the L6 Mission OPSCON (Ref. 1)

- Section 6.2, "LGN Contact", of the L9 Mission OPSCON (Ref. 1) describes how the mission system is expected to operate during a contact.
- Section 6.9, " Monitor Health & Safety", of the L9 Mission OPSCON (Ref. 1) describes how the mission system is expected to monitor the health and safety of the observatory
- Section 6.10, "Safehold/Autonomous Failsafe Occurrence and Recovery Mission", of the L9 Mission OPSCON (Ref. 1) describes how mission system is expected to handle an anomaly experienced by the observatory that threatens earth-imaging sensor (I.e. TIRS-2 or OLI-2) or spacecraft health and safety
- Section 6.13, "Backup Flight Operations," of the L9 Mission OPSCON (Ref. 1) describes how the mission system is expected to handle an anomaly experienced by the ground system

In addition, Section 4.2.1, "Sustaining engineering efforts for the Space Segment" of the L9 Mission OPSCON (Ref. 1) identifies maintenance and update of spacecraft and imaging sensor flight software during mission operations.

Risk Drivers

The following is from a list of what Orbital ATK considers as Major Drivers for the design of the Landsat 9 SC Bus (Ref. 2, Section 8, "Spacecraft Systems Engineering," slide 11):

- Storage of SOH telemetry in IEM [SPAM] (LEOStar-3 Upgrade)
- Accommodate increased quantity of TIRS-2 temperature sensors
- Accommodate spare instrument power services

- Electrical Current monitoring of all services (LEOStar-3 Upgrade)

Notes, Assumptions, and Questions

a. Initial, preliminary versions of the Spacecraft FMEA and FTA are not available. These artifacts are not required to be developed until prior to SC PDR

a. It is unknown whether or not there is autonomous time, memory (i.e. memory scrubbing), or processor utilization management on board the observatory

a. It is assumed that an unresponsive RAD750 (possibly due to a software failure) is a type of mission-critical fault that will cause an IEM Side switch

a. It is unknown whether or not the MOC will have the ability to set up and run a contact autonomously

a. According to L9-SRD-103 (Ref. 3), the Spacecraft shall be single-fault tolerant to meeting the mission-level requirements.

a. Look to assure that the LEOStar-3 upgrades since Landsat 8 are appropriate for the Landsat 9 mission and implemented correctly

a. Look to assure that the LEOStar-3 upgrade to store and retrieve the SOH telemetry from the IEM (SPAM) is appropriate for the Landsat 9 mission and implemented correctly (impacts the assurance of **Record Observatory Housekeeping Data** and **Playback Observatory Housekeeping Data**).

a. Look to assure that the LEOStar-3 upgrade to monitor electrical current of all services is appropriate for the Landsat 9 mission and implemented correctly (impacts the assurance of **Maint ain Observatory Power**)

a. Look to assure that the tailoring of the LEOStar-3 to accommodate the Landsat 9 mission requirements is appropriate

a. Look to assure that the modifications to the LEOStar-3 spacecraft software to accommodate the increased quantity of TIRS-2 temperature sensors is implemented correctly and does not introduce any undesired features (impacts the assurance of **Maintain a Thermally Stable Observatory**)

a. Look to assure that the modifications to the LEOStar-3 spacecraft software to accommodate the spare instrument power services is implemented correctly and does not introduce any undesired features (impacts the assurance of **Maintain Observatory Power**)

References

1. Landsat 9 Mission Operations Concept Document (L9-00-00, Revision -, Released on DD/MM/YYYY)
2. Landsat 9 SC SRR
3. Landsat 9 SRD (L9-06-03, Revision -, Released on DD/MM/YYYY)

Applicability to Earth Science Missions

The same set of generic phase-based capabilities is extensible to Earth science missions, with some considerations.

Generic Capability	Considerations
Travel to Science Objective	Earth science missions are typically launched directly into their desired orbits (or close to it), so there may be little use for a behavioral decomposition in this area. Missions which require significant delta-V supplied by the spacecraft (not LV) to achieve the desired orbit, require frequent orbit changes, orbit maintenance, or novel orbits may warrant consideration.
Arrive at Science Objective	This generic capability is probably not applicable to Earth Science, as it typically includes Entry, Descent, Landing or Planetary Orbit insertion.
Primary Mission Operation	Many earth science missions make use of relatively straightforward instrument payloads, which work in concert with spacecraft slews to collect data about the surface of the earth. However, there are instances of instruments on Earth Science missions which displayed more unique/riskier behaviors. For example, on SMAP, the active radar performance relied on deployment and constant spinning of a very large antenna, which has much higher risk

	than the passive radar which only relied on the flight computer turning on and off the instrument.
Return Science	This generic capability is probably less applicable to Earth science, as it could include sample curation, return orbits to Earth, etc. Earth science missions typically return data by downlinking from orbit, so any unique configurations or uses of the telecom system may warrant consideration.

Table 26: Earth Science Mission Capbility Examples

Again, special consideration should be given to any unique behaviors exhibited by the mission. Some Earth science missions, such as Landsat 9 do not exhibit many unique behaviors, as it mostly points and slews to capture data with the two primary science instruments. However, a mission like the Magnetospheric Multiscale Mission exhibited several unique behaviors, including position coordination between multiple spacecraft, formation flying, etc.

Applicability to HEO and Ground Projects

There may be useful, logical generic capability abstractions that exist for HEO and Ground IV&V projects, though no research or assessment was done as part of this CD effort. Here are some thoughts to consider:

Step 4: Example – Software Role to Mission Capability

3.1 Entity	*3.2 Role*
Spacecraft Software	**Record Observatory Housekeeping Data:** The Observatory collects telemetry points regarding health & status information of subsystems, including the imaging sensors. The Observatory will store state of health (i.e., housekeeping) telemetry and make it available for downlink. The data will be stored as files. (p. 13 of the OPSCON) (Table 6-7 of the L9 Mission OPSCON (Ref. 1)) **Telemeter Observatory Housekeeping Data:** During contacts with an LGN station, the real-time housekeeping telemetry data will be downlinked. The LGN station routes this data in real time to the MOC. (Table 6-7 of the L9 Mission OPSCON (Ref. 1)) The observatory uses stored CMDs to turn on the S-Band D/L approximately 30 sec before entering the 0° horizon mask of a station to ensure the D/L is present as an acquisition aid for ground station antenna. (OPSCON 6.2.3)

	Playback Observatory Housekeeping Data: The stored housekeeping telemetry data will be downlinked upon command. (Section 3.1.1.2, "Health and Safety Maintenance" of the L9 Mission OPSCON (Ref. 1)) Downlink of the stored state of health data can occur over the S- or X-Band links. (p. 13 of the OPSCON). **Detect and Respond to mission-critical faults:** The Observatory will monitor the health and status of each subsystem, including the imaging sensors. If any of a predefined list of anomalies occurs, the Observatory will have the ability to detect the anomaly. If the severity of the anomaly warrants it, the Observatory will automatically place itself in a safe and protected state.(Section 3.1.1.2, "Health and Safety Maintenance" of the L9 Mission OPSCON (Ref. 1)) Once the observatory successfully enters safehold mode it should not change modes without ground commanding (OPSCON 6.10.2). **Maintain Observatory Power:** Power management functions will also be performed on board (Section 3.1.1.2, "Health and Safety Maintenance" of the L9 Mission OPSCON (Ref. 1)). **Maintain a Thermally Stable Observatory:** Thermal management functions will also be performed on board (Section 3.1.1.2, "Health and Safety Maintenance" of the L9 Mission OPSCON (Ref. 1)). **Telecommand Observatory**: Commands and spacecraft software updates received via the S-Band communications stream will be processed onboard and forwarded to the respective Observatory subsystems [or imaging sensor] for execution. The Observatory will nominally receive command loads every 24 hours, and each will encompass spacecraft bus activities (i.e. communications management). If an updated command load has not been received at the end of the 72-hour period, the Observatory will be left in a safe operational state. Command links will be established each contact (Section 3.1.1.1, "Communications, Command and Data Handling" of the L9 Mission OPSCON (Ref. 1))
GNE Software	**Telemeter Observatory Housekeeping Data:** Approximately 5-15 min prior to AOS, the LGN station and the MOC will execute the pre-pass sequence to configure the LGN station for L9 and establish the connection between the LGN station and the MOC (OPSCON 6.2.3). The LGN station will acquire the S-Band D/L and program-track the observatory until it exits ground clutter, typically around 2-3°, at which point it start S-Band auto-tracking. The TLM RX and CTPs will start sending the received TLM to the MOC as soon as the station acquires the D/L. (OPSCON 6.2.3) **Telecommand Observatory**: The GNE generates an idle pattern to modulate the U/L when the MOC is not actively commanding. (OPSCON 6.2.3)
MOC Software	**Telemeter Observatory Housekeeping Data:** Approximately 5-15 min prior to AOS, the station and the MOC will execute the pre-pass sequence to configure the station for L9 and establish the connection between the station and the MOC (OPSCON 6.2.3). The MOC/FOT will monitor RT TLM during the course of the contact (OPSCON 6.2.3) **Playback Observatory Housekeeping Data:** The MOC/FOT will download S-Band stored (back-orbit) TLM during the course of the contact. (OPSCON 6.2.3) **Detect and Notify FOT of mission-critical faults:** The MOC Command & Control function automatically monitors all key telemetry points for pre-

	established limit values. Any out-of-limits value is automatically detected and a notification is sent or displayed to the FOT. (Table 6-7 of OPSCON) **Telecommand Observatory**: Once modulation is enabled, the MOC will start sending CMDs to the observatory for RT operations and stored CMD uploads.

HEO

Many HEO projects are designed for reusability – so the same hardware and software will potentially be used in many different situations, operating envelopes, stressing circumstances, etc. This can add a daunting amount of uncertainty to defining mission-level capabilities. For a highly complex human rated mission, there may be a tendency to iteratively add functionality. For instance, the MPCV development is laid out into a series of Flights/Missions which iteratively add functionality and complexity to the software. One potential recommendation could be to limit the focus of the current set of mission capabilities to the circumstances of the upcoming iteration of the mission.

MPCV Mission	Year	New functionality/ circumstances
EFT-1: Exploration Flight Test 1	2014	Launch and establish unmanned craft, Earth EDL
EM-1: Exploration Mission 1	2018	Unmanned with functional life support, Service Module integration, SLS Block-1 interface, Payload deployment, lunar orbit
EM-2: Exploration Mission 2	2023	Manned with four astronauts, SLS Block-1B with Exploration Upper Stage interface, flyby of asteroid in lunar orbit

Table 27: MPCV Mission Decomposition

These milestones in the development in the development help to clarify which capabilities IV&V should focus on at every step of the development. It may be a good practice for capabilities to mirror the new functionality/ mission requirements for the next upcoming milestone.

Ground

There is an increasing recognition at IV&V that ground systems development requires unique approaches to providing assurance. There is a Ground Systems IV&V Initiative in FY16 which may provide some useful guidance on how to generate capabilities for ground systems.

Step 3: Example of a Reference Architecture (where SW architecture was not yet available from Developer)

The development of a reference architecture can be a helpful exercise to establish an early understanding of the necessary software entities when the proper developer artifacts are not yet available. Since this reference will later be replaced by information from the developer it is important to spend just enough time to arrive at a basic understanding. Documents such as the OpsCon, Design Reference Mission, or Mission Definition Review charts can provide the basic information necessary to create a notional understanding of the software architecture. Here are some questions to consider when performing this activity:

- What hardware is necessary for this mission?
- What specialized software must be developed for this mission?
- What software domains does this developer use?
- Are multiple developers contributing partitions/components to the mission?
- Where are the interfaces between contributions from different developers?

Figure 6 provides a reference software architecture below was developed in response to generalized information about an Asteroid Sample and Return mission.

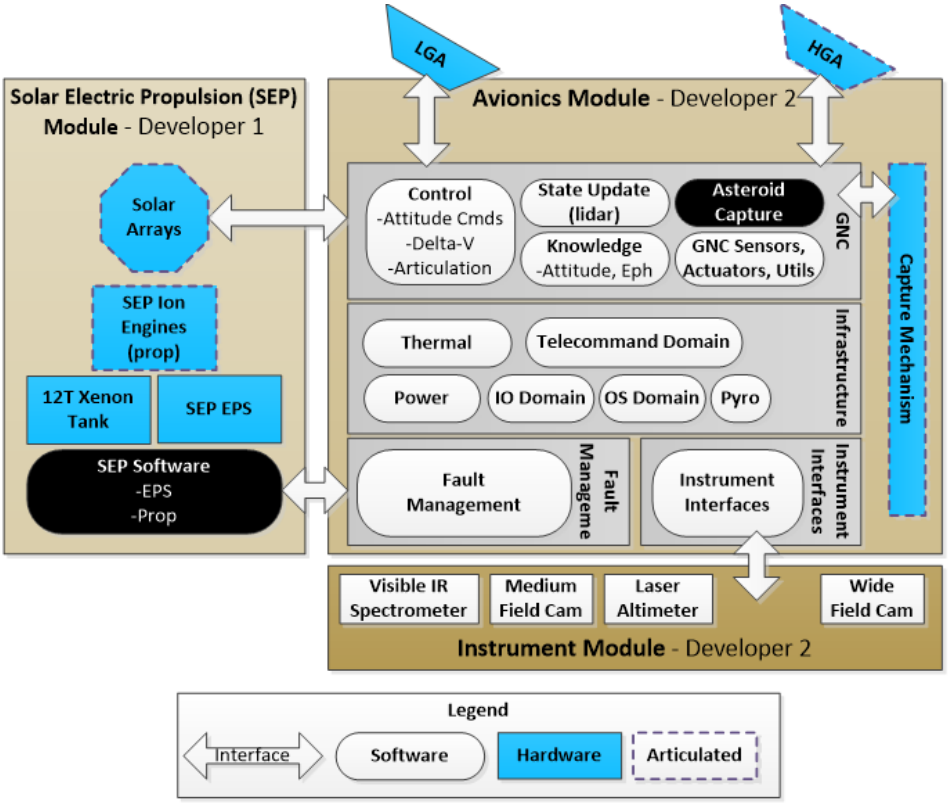


Figure 6: Example of a Reference Software Architecture

As shown in this example, the exact software decomposition was not known at a CSCI level. However, a software architecture could be estimated a as a starting point based upon the mission capabilities and objectives. Note that some notional software entities were understood to correspond to unique mission aspects, so these were represented as black boxes in the architecture diagram. Note also that this architecture includes expected software interfaces, particularly between different development efforts, as these can be drivers for development complications. While it is not necessary to define hardware as part of the notional architecture, it can be helpful in further fleshing out an understanding.

Step 3: Software Decomposition example and means to link Mission Capabilities to Software Elements

Table 29 shows an example Capability and Entity mapping for the capability to “Collect Sample from Asteroid Surface” using the sample set of entities shown in Table 28.

TLCM	GNC		SC	PYLD	INSTR	Ground
VML_2	Nav_lidar_tot	Hw_tagcams	Repm	OCAMs	OCAMS	Operator
Telecom	Nav_lidar_state	Art_macm	Fpe	Payload_		FDS
Upl	Nav_dvu		Sme	common		

Down	Nav_dv		Pyro			
	Hw_thr		eps			

Table 28: Sample set of entities for use in C&E example

The C&E mapping documents the role entities play in achieving a capability. It can be helpful to denote the drivers and participants in this mapping (marked with X and O, respectively). Remember, drivers are active participants in capabilities, while participants provide passive support for capabilities.

Consider the following examples from the table below:

€€€€€ Entity based explanation: Consider the "fpe" entity below. This is the Fault Protection Executive, which is responsible for monitoring spacecraft indications and commanding Safe Mode entry/exit when necessary. During sub-capability 6 (sub6), this entity is mapped as a driver since this capability pertains to a spacecraft abort of a sampling attempt, and aborts are driven by a particular entity (repm) requesting Safe Mode. The fpe entity participates in all other sub-capabilities, as it is monitoring for Safe Mode requests in the background and returning fault protection enable information to requestors, but not active in the accomplishment of these sub-capabilities.

€€€€€ Capability based explanation: Consider sub1 below. This capability involves the ground creating and uplinking a sampling sequence to the spaceship. Starting on the left:

ML_2 (Virtual Machine Language) is a driver because it is responsible for assessing the validity of the uplinked sequence

lecom is a participator because it is monitoring the telecom hardware in use (including fault protection), but is not actively configuring it at the time.

l (uplink) is a driver because it actively receives and directs the uplink from the ground

e (Fault Protection Executive) is a participant.

S (Electrical Power System) is a participant because it is performing its normal monitoring and activity, but is not active in the sub1 behavior.

ttery is a participant because it is performing its normal monitoring and activity, but is not active in the sub1 behavior.

erator is a driver because it is actively transmitting the sampling sequence to the spacecraft.

S (flight dynamics) is a driver because it is the ground team responsible for creating and approving the sampling sequence.

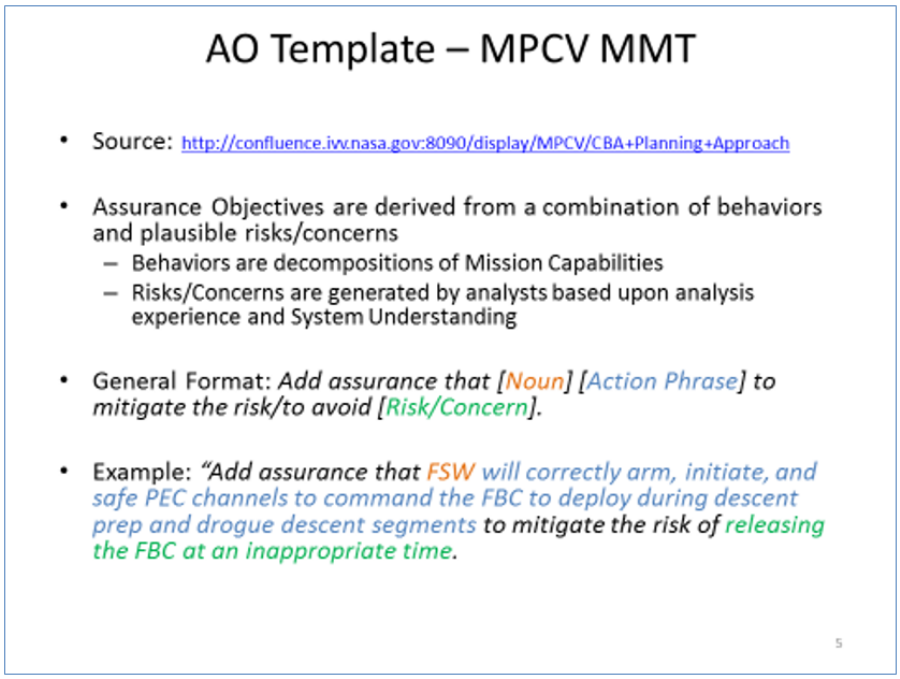
€€€€€ **Interface software explanation: TBD**

Capabilities	TLCM-vml_2	TLCM-telecom	TLCM-upl	TLCM-dwn	GNC-nav_lidar_tot	GNC-nav_lidar_state	GNC-nav_dvu	GNC-nav_dv	GNC-hw_thr	GNC-hw_tagcams	GNC-art_macm	IO-macm	SC-repm	SC-fpe	SC-pyro	SC-sme	SC-eps	SC-battery	PYLD-payload_common	PYLD-ocams	Instr SW-OCAMS	Grnd-Operator	Grnd-FDS
Collect Sample from Asteroid Surface																							
1. TAG sequence uplink and validation	X	O	X											O			O	O				O	X
2. Articulate sampling arm	O	X		O				X	O		X	O	O	O			O	O					
3. Checkpoint/Matchpoint Calculations and DV adjustment	O	O		O	O	X	X	O	O	O			X	O			O	O					
4. Surface contact logic and sample collection	O	O		O	X		O			O			X	O	X		O	O	O	X	X		
5. Sample Mass Verification and stow ops		X	X	X						X	X	O		O	X		O	O	O	X	X	X	
6. Emergency sampling attempt abort	X	X		O				X	O		X	O	X	X		X	X	O					

Table 29: Example Capability to Entity Mapping

Note: the capability to entity mapping should include ALL entities.

Step 5: MPCV Assurance Objective Example



Step 5: Example Capability to Assurance Objective Example

Below is an example of possible Assurance Objectives that may emerge once a Capability has been scored. The Assurance Objectives should result from the uncovered risks discussed in the scoring Rationale

Capability	Risk Category	Score	Rationale	Assurance Objectives
Capability 1 "Perform S/C Health and Maintenance during Cruise"	IA	4	This capability requires extensive communication between the ground and the spacecraft. Given that it is an Earth-orbiting mission, there is a risk that Integrity and Availability of information could be severely affected by agencies intending to harm US assets	-Provide assurance that communications between ground and the spacecraft are secure
	Developer	4	There are multiple developers developing the subsystems that will be implemented into the overall flight system. IV&V has experience with two of the developers and have had poor experiences with the interfacing of their development products to overall systems.	-Provide assurance that products from multiple developers are correct and consistent
	Functional	5	Without the ability to monitor the system state, the Flight System can not ensure that the spacecraft will stay in a safe state. This includes the ability to perform communications with the ground for intervention as well as for keeping the spacecraft in a position that enables proper thermal and power states .	-Provide assurance that internal S/C interfaces provide the necessary telemetry to evaluate the S/C health -Provide assurance that the S/C can completely and correctly inform the Ground system of the current S/C health state -Provide assurance that the Ground system can correctly provide data to the S/C to address S/C health concerns and periodic maintenance
	Performance	2	As the spacecraft will be in a safe orbit during this phase and has the ability to communicate in multiple attitude configurations as well as have a battery that will support an extended period of time without needing charged, there is not a great need for time-sensitive reaction to adverse conditions. If an off-nominal situation is uncovered, the S/C and ground have a large amount of time to react and respond.	-Provide assurance that the S/C can process all required data necessary from all subsystems to be able to assess S/C health

RBA: Software Decomposition example and means to link Mission Capabilities to Software Elements

Table 29 shows an example Capability and Entity mapping for the capability to “Collect Sample from Asteroid Surface” using the sample set of entities shown in Table 28.

TLCM	**GNC**		**SC**	**PYLD**	**INSTR**	**Ground**
VML_2	Nav_lidar_tot	Hw_tagcams	Repm	OCAMs	OCAMS	Operator
Telecom	Nav_lidar_state	Art_macm	Fpe	Payload_ common		FDS
Upl	Nav_dvu		Sme			
Down	Nav_dv		Pyro			
	Hw_thr		eps			

Table 30: Sample set of entities for use in C&E example

The C&E mapping documents the role entities play in achieving a capability. It can be helpful to denote the drivers and participants in this mapping (marked with X and O, respectively). Remember, drivers are active participants in capabilities, while participants provide passive support for capabilities.

Consider the following examples from the table below:

- Entity based explanation: Consider the “fpe” entity below. This is the Fault Protection Executive, which is responsible for monitoring spacecraft indications and commanding Safe Mode entry/exit when necessary. During sub-capability 6 (sub6), this entity is mapped as a driver since this capability pertains to a spacecraft abort of a sampling attempt, and aborts are driven by a particular entity (repm) requesting Safe Mode. The fpe entity participates in all other sub-capabilities, as it is monitoring for Safe Mode requests in the background and returning fault protection enable information to requestors, but not active in the accomplishment of these sub-capabilities.
- Capability based explanation: Consider sub1 below. This capability involves the ground creating and uplinking a sampling sequence to the spaceship. Starting on the left:
  - VML_2 (Virtual Machine Language) is a driver because it is responsible for assessing the validity of the uplinked sequence
  - Telecom is a participator because it is monitoring the telecom hardware in use (including fault protection), but is not actively configuring it at the time.
  - Upl (uplink) is a driver because it actively receives and directs the uplink from the ground
  - Fpe (Fault Protection Executive) is a participant.
  - EPS (Electrical Power System) is a participant because it is performing its normal monitoring and activity, but is not active in the sub1 behavior.
  - Battery is a participant because it is performing its normal monitoring and activity, but is not active in the sub1 behavior.
  - Operator is a driver because it is actively transmitting the sampling sequence to the spacecraft.
  - FDS (flight dynamics) is a driver because it is the ground team responsible for creating and approving the sampling sequence.
  - Operator is a driver because it is actively transmitting the sampling sequence to the spacecraft.

References

REFERENCES	
Document ID/Link	**Title**
IVV QM	NASA IV&V Quality Manual
IVV 09-4	Project Management
NASA-GB-8719.13	NASA Software Safety Guidebook
NASA-STD-8719.13B	NASA Software Safety Standard
NPR 8621.1	NASA Procedural Requirements for Mishap and Close Call Reporting, Investigating, and Recordkeeping

If any procedure, method, or step in this document conflicts with any document in the NASA Online Directives Information System (NODIS), this document shall be superseded by the NODIS document. Any external reference shall be monitored by the Document Owner for current versioning.

Version History

VERSION HISTORY			

Version	Description of Change	Author	Effective Date
Basic	Initial Release	Christina Moats	12/11/2008
A	Made minor revisions for clarity	Christina Moats	1/13/2009
B	Major revision, combining the PBRA process with the RBA process produced by an assessment team in March 2010	Jeff Northey	2/23/2011
C	Remove ambiguous use of IV&V Coverage and coverage categories	Patrick Theeke, et. al.	4/17/2012
D	Updated based on PAR 2012-P-364: Added Safety Criteria: Damage/loss to other spacecraft/assets. Fixed URL. Reworded some Complexity.	Steve Husty	10/4/2012
E	Modified PBRA process to more closely align with assessing project risk relative to the Capability Based Assurance approach to performing IV&V.	Jeremy Fienhold, et. al.	4/26/2018