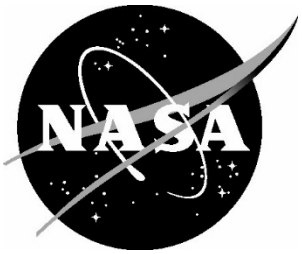
NASA/TP–20260004453



Assumption Management in Complex Spaceflight: The Artemis Campaign

Bharkha Mistry
NASA Intern, NASA Johnson Space Center, Houston, Texas

June 2026

NASA STI Program Report Series

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA scientific and technical information (STI) program plays a key part in helping NASA maintain this important role.

The NASA STI program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI program provides access to the NTRS Registered and its public interface, the NASA Technical Reports Server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA Programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

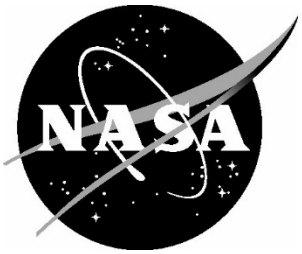
Specialized services also include organizing and publishing research results, distributing specialized research announcements and feeds, providing information desk and personal search support, and enabling data exchange services.

For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at http://www.sti.nasa.gov

.

NASA/TP–20260004453



Assumption Management in Complex Human Spaceflight: The Artemis Campaign

Bharkha Mistry
NASA Intern, NASA Johnson Space Center, Houston, Texas

National Aeronautics and
Space Administration

Johnson Space Center
Houston, Texas

June 2026

Available from:

NASA STI Program / Mail Stop 050
NASA Langley Research Center
Hampton, VA 23681-2199

Assumption Management in Complex Human Spaceflight

The *Artemis* Campaign

Authored By: Bharkha Mistry

NASA JSC Intern

Executive Summary

Human spaceflight does not fail only when hardware breaks. It fails when organizations come to believe that their assumptions are safer, stronger, or more proven than they actually are.

This paper argues that assumption management should be treated as a formal safety and systems discipline in complex human spaceflight programs, particularly within the *Artemis* Exploration Campaign. *Artemis* is not a single vehicle program in the mold of *Apollo* or *Shuttle*. It is a distributed campaign architecture composed of government systems, commercial providers, international partners, evolving infrastructure, shifting policy direction, and mission sequences that depend on capabilities maturing across different organizations and timelines. In that environment, assumptions are not background planning conveniences. They are functional system elements. They shape architecture, verification, operations, safety confidence, and ultimately the conditions under which crews are asked to fly.

The central concern is not that *Artemis* contains assumptions. All complex programs do. The concern is that assumptions can drift, propagate, and harden across the lifecycle until provisional judgments begin to function as inherited truths. A schedule assumption can become an architectural dependency. A partner maturity assumption can become a verification premise. A successful demonstration can be interpreted as operational sufficiency. A prior mission's survival can be mistaken for proof that the system is ready for the next level of risk. When that happens, the campaign may appear increasingly mature on paper while becoming more fragile in practice.

As *Artemis* introduces higher energy, more integrated systems, the validity of verification evidence becomes lifecycle dependent requiring independent interpretation and assumption stewardship to maintain safety at all levels. To examine this problem, the paper develops an assumption lifecycle framework for complex human spaceflight systems. The framework traces how assumptions enter at architecture decision points, acquire meaning through verification and safety interpretation, and reappear in operations such as mission rules, contingency logic, crew survival expectations, and accepted risk. It distinguishes assumptions from requirements and verification evidence, emphasizing that a system can satisfy documented requirements while still relying on unexamined or outdated premises about context, interfaces, infrastructure, organizational capacity, or future readiness.

Historical human spaceflight mishaps show why this matters. *Apollo* 1, *Challenger*, and *Columbia* were not merely failures of hardware or procedure. They were failures of interpretation, escalation, governance, and organizational learning. In each case, warning signs existed before catastrophe, but assumptions about acceptability, prior success, operational normalcy, or program momentum distorted how those warnings were understood. The lesson is not that *Artemis* is destined to repeat those failures. The lesson is that human spaceflight organizations must continuously challenge the assumptions that make risk feel understood.

The paper then applies this logic to modern human spaceflight through *Starliner* and *Artemis*. *Starliner* illustrates how verification evidence can become difficult to interpret when qualification, oversight, communication, and organizational trust are distributed across program boundaries. *Artemis* extends the problem further. Its campaign architecture depends on the maturity of SLS, Orion, commercial Human Landing Systems, xEVAS, Gateway related planning, surface power, mobility, logistics, communications, mission cadence, workforce continuity, budget stability, and governance alignment. Any one of these areas can carry assumptions. Together, they create a campaign wide assumption environment.

A central claim of the paper is that crew survival must be treated as the true measure of *Artemis* maturity. A campaign is not mature because it maintains schedule momentum, publishes an architecture, assigns providers, or completes discrete verification products. It is mature only when the assumptions linking architecture, verification, operations, infrastructure, abort logic, safe haven capability, surface survival, return-to-orbit dependencies, and decision authority are visible, owned, tested, and governed. Momentum is not a survival strategy. Cadence is not proof. Distributed accountability is not the same as integrated accountability.

This paper therefore calls for assumption stewardship at the campaign level. Safety governance must be capable of identifying assumptions that cross program boundaries, tracking how their meaning changes over time, and forcing reassessment when architecture, policy, funding, workforce, partner maturity, or mission context changes. Assumptions that carry architectural or crew survival consequences should be managed with rigor proportional to the consequences of their failure. They should have owners, evidence thresholds, review triggers, retirement criteria, and escalation paths.

The *Artemis* Campaign represents one of the most ambitious human exploration efforts in NASA's history. Its success will depend not only on technical excellence, but on the agency's ability to preserve disciplined judgment across a distributed and changing system. The question is not whether NASA can build pieces of *Artemis*. The question is whether *Artemis* can remain governable as an integrated human spaceflight campaign whose assumptions stay visible before they become hazards.

The standard should be clear: no assumption that carries crew survival risk should be allowed to become invisible simply because the campaign needs it to be true.

Contents

Figures

Tables

List of Acronyms/Abbreviations

AR1	*Artemis* Architecture Revision
AS9100D	Aerospace quality management system standard, Revision D
ASAP	Aerospace Safety Advisory Panel
CAIB	*Columbia* Accident Investigation Board
CCP	Commercial Crew Program
CDR	Critical Design Review
CFR	Certification of Flight Readiness
CFT	Crew Flight Test
CM	Crew Module
CRS	Congressional Research Service
DOE	U.S. Department of Energy
DOS	U.S. Department of State
ECLSS	Environmental Control and Life Support System
EGS	Exploration Ground Systems
EPA	U.S. Environmental Protection Agency
ESA	European Space Agency
ET	External Tank
EVA	Extravehicular Activity
FAA	Federal Aviation Administration
FMEA	Failure Modes and Effects Analysis
FOM	Federation Object Model
FRR	Flight Readiness Review
FSP	Fission Surface Power
FY	Fiscal Year
G1	Gate 1: Architecture Decision
G2	Gate 2: Safety / Verification Interpretation Decision
G3	Gate 3: Operational Decision
GAO	U.S. Government Accountability Office
GDP	Gross Domestic Product
GNC	Guidance, Navigation, and Control
GNP	Gross National Product
GPO	U.S. Government Publishing Office
HALO	Habitation and Logistics Outpost
HLS	Human Landing System
HRO	High Reliability Organization
IAQG	International Aerospace Quality Group
ICD	Interface Control Document
IG	Inspector General
INCOSE	International Council on Systems Engineering
INSRB	Interagency Nuclear Safety Review Board
ISRU	In-Situ Resource Utilization
ISS	International Space Station
JSC	Johnson Space Center
KRUSTY	Kilopower Reactor Using Stirling TechnologY
KSC	Kennedy Space Center
LCC	Launch Commit Criteria

LEO	Low Earth Orbit
LOC	Loss of Crew
LOM	Loss of Mission
LPI	Lunar and Planetary Institute
LROC	Lunar Reconnaissance Orbiter Camera
LRV	Lunar Roving Vehicle
LTV	Lunar Terrain Vehicle
MA	Mission Assurance
MECO	Main Engine Cutoff
MET	Modular Equipment Transporter
NASA	National Aeronautics and Space Administration
NC	JSC Space Transportation Systems Division
NC2	JSC Program SMA Integration Branch
NC3	JSC Space Sys5tems Safety Branch
NEP	Nuclear Electric Propulsion
NEPA	National Environmental Policy Act
NODIS	NASA Online Directives Information System
NPD	NASA Policy Directive
NPR	NASA Procedural Requirements
NRHO	Near-Rectilinear Halo Orbit
NRO	National Reconnaissance Office
NSF	National Science Foundation
NSPM	National Security Presidential Memorandum
NSTM	National Security and Technology Memorandum
NTRS	NASA Technical Reports Server
OCFO	Office of the Chief Financial Officer
OFT	Orbital Flight Test
OFT-1	Orbital Flight Test 1
OFT-2	Orbital Flight Test 2
OIG	Office of Inspector General
OMB	Office of Management and Budget
OMS	Orbital Maneuvering System
OSMA	Office of Safety and Mission Assurance
PIT	Program Investigation Team
PPBE	Planning, Programming, Budgeting, and Execution
PRA	Probabilistic Risk Assessment
PRACA	Problem Reporting and Corrective Action
RCC	Reinforced Carbon-Carbon
RCS	Reaction Control System
RIRM	Risk of Inadequate Mission Rescue
SE&I	Systems Engineering and Integration
SLS	Space Launch System
SM	Service Module
SMA	Safety and Mission Assurance
S&MA	Safety and Mission Assurance
SPD	Space Policy Directive
SRB	Solid Rocket Booster

STEM	Science, Technology, Engineering, and Mathematics
STS	Space Transportation System
TPS	Thermal Protection System
VA	Verification and Analysis
VIO	Verification / Validation and Independent Oversight
VSE	Vehicle Systems Engineer
V&V	Verification and Validation
WFTCA	Working Families Tax Cut Act
xEVA	Exploration Extravehicular Activity
xEVAS	Exploration Extravehicular Activity Services

1. Introduction

Space is hard. Add humans into the mix and it becomes harder. Human spaceflight introduces an element of intricacy that demands robust safety nets to ensure astronauts return home alive and whole. Human spaceflight operates at the intersection of engineering complexity, organizational decision-making, and evolving assumptions about system behavior. As the *Columbia* Accident Investigation Board (CAIB) noted, accidents in complex systems cannot be understood without examining the organizational context in which decisions occur. The National Aeronautics and Space Administration's (NASA) history shows that assumptions where vehicle capability, operational environments and programmatic stability are concerned, can propagate through architecture, verification, and operations[1]. As exploration architectures grow more complex, managing those assumptions becomes increasingly critical to maintaining safety confidence.

Programmatic shift due to political drift is an additional stressor that is not needed to feed the monster within the labyrinth[2]. Policy and programmatic direction shape human spaceflight systems by establishing mission objectives, cadence expectations, funding profiles, and governance structures before technical architectures fully mature. In government space programs, these forces do not act outside the engineering process; they establish many of the assumptions on which that process depends, including assumptions about schedule continuity, workforce stability, infrastructure availability and program governance. As exploration architectures evolve across administrations, budget cycles, and partnership models, those upstream decisions can propagate into design trades, verification strategies, and operational safety margins. Understanding how policy driven assumptions enter and move through complex human spaceflight systems is therefore essential to understanding how risk is created, interpreted, and managed.

The *Artemis* Campaign is the clearest example of this shift. It is not simply a launch vehicle program, a spacecraft program, or a lunar mission series. It is a campaign that depends on the continuing alignment of transportation systems, habitation and surface power systems, launch and ground infrastructure, communications, logistics, power, commercial providers, international partners, and future operational learning that does not yet fully exist. The campaign therefore advances through a distributed architecture in which decisions must regularly be made before the final integrated system is available for direct observation or end-to-end demonstration.

In this environment, assumptions are not peripheral planning conveniences - they function as connective tissue between incomplete elements of the architecture. They allow organizations to proceed despite uncertainty about partner maturity, infrastructure availability, mission cadence, operational constraints, verification closure, and future policy continuity. Assumptions can stabilize decision making and enable progress, but they can also impede by propagating hidden fragilities

[1] *Columbia* Accident Investigation Board, *Report Vol. I* (2003), p. 8 and ch. 7, pp. 195–203; NASA, *Systems Engineering Handbook*, NASA/SP-2016-6105 Rev. 2 (2016), ch. 1, §§1.1–1.2; NASA, *NPR 8705.2C, Human-Rating Requirements for Space Systems* (updated Jan. 19, 2024), Preface P.1–P.2, pp. 4–6, and ch. 2, §§2.1–2.3..

[2] Broniatowski & Weigel, "Political Feasibility and Sustainability in Space Exploration," *Space Policy 24*(3) (2008), pp.148–157.

across organization and technical boundaries when they remain implicit, weakly owned, or insufficiently revisited.

The problem becomes considerable when assumptions begin to migrate from provisional judgements into inherited truths. A planning premise can be continual in schedules, interface documents, verification narratives, operational concepts, and strategic communications until it acquires the appearance of evidence. Once that happens, the architecture may continue to mature around a condition that has not actually been demonstrated or may no longer be valid in its original form. What appears to be a local technical issue may therefore reflect a wider failure of interpretation, stewardship, and accountability across the campaign.

This paper argues that assumption management should be treated as a formal systems-and-safety discipline in complex human spaceflight enterprises. Its concern is not only whether assumptions exist, but how they are formed, inherited, interpreted, challenged and retired across lifecycle gates and program boundaries. The central claim is that assumptions in distributed exploration architectures behave like system elements: they transmit risk, shape design and verification decisions, conditional operational confidence, and influence whether a campaign remains robust as it evolves.

To develop that argument, the paper first frames the rise of campaign based exploration and the increasing structural complexity of modern spaceflight. It then distinguishes assumptions from requirements and verification evidence, proposes a lifecycle based assumption framework, and examines historical patterns visible in *Apollo 1, Challenger,* and *Columbia*. These historical lessons are paired with modern case material, including *Starliner* and recent *Artemis* architecture changes, in order to show how untracked or weakly governed assumptions can accumulate, distort confidence, and shift risk into later phases of a campaign.

To close out the argument, crew survival and the challenges associated with it are highlighted, and safety governance and stewardships are examined before discussions take place.

1.1 Background: Increasing Complexity in Human Spaceflight

As human exploration shifts from discrete missions to campaign-based architectures, assumptions no longer remain confined within a single vehicle or program. Instead, they are distributed across launch systems, crew vehicles, landers, surface infrastructure, commercial partners, and safety oversight organizations, often evolving asynchronously over time[3]. In such environments, safety is influenced not only by whether requirements are verified, but by whether the assumptions underlying those requirements remain valid as architectures, schedules, and program priorities change. This creates a systems engineering challenge in the purest sense: **assumptions must be treated as**

[3] NASA *Systems Engineering Handbook* (NASA/SP-2016-6105 Rev. 2, 2016); 51 U.S.C. §20302(b); NPR 8705.2C, Ch. 1; NASA *Artemis Architecture Definition Document*; NASA *Moon to Mars Architecture Definition Document* (Moon to Mars architecture page).

traceable, reviewable, and safety relevant elements of the program lifecycle rather than as implicit background conditions[4].

The increasing complexity of human spaceflight is more than a story of larger vehicles, complex missions, or increasingly advanced technology. It is also a story of changing system structures[5]. As exploration has moved beyond relatively self-contained program forms and towards campaigns assembled across multiple entities, complexity has become more distributed, more temporal, and more interpretive. The result is not just more parts; it is a different mode of interdependence.

The *Apollo*[6] and *Shuttle* programs were enormous and deeply demanding undertakings, and they should not be reduced or caricatured as simple programs. Both required sophisticated integration, political management, major industrial coordination, and extensive operations. They remain foundational achievements whose lessons continue to shape modern exploration. When compared to the *Artemis* Campaign, they had a narrower scope with a more clearly defined architecture, identity, and alignment between design authority and operations[7]. Their complexity was expansive, but it was less diffused across semi-autonomous programs pursuing asynchronous forms of maturity within a broader campaign construct.

Modern exploration campaigns depend on a different pattern. Vehicles, habitats, landers, surface systems, power systems, logistics chains, mission control functions, communications networks, and launch infrastructure may be developed under different contracts, different leadership structures, different political pressures, and different schedules. Some elements are government managed, some are commercially developed, some are internationally contributed, and some are not yet fully defined. The architecture therefore exists partly as hardware and partly as a negotiated expectation about future readiness[8]. This creates structural exposure in at least three ways:

- First, decisions are often made on the basis of projected maturity[9] rather than demonstrated maturity.
- Second, verification confidence may be assembled from distributed evidence that is locally persuasive but not always globally coherent[10].

[4] INCOSE *Systems Engineering Handbook*, 5th ed. (2023): pp. 3-8, 311–312
[5] NASA, *Moon to Mars Architecture Definition Document*, ESDMD-001, Revision C, NASA/TP-20250010956 (Dec. 12, 2025), Executive Summary p. 9, Architecture Foundations pp. 11–19, and Architecture Components pp. 21–32.
[6] NASA, *Apollo Program Summary Report*, NASA SP-350 (1975), chs. 2–4 and mission summaries.
[7] Dreier, Casey. "3 Charts That Show How *Artemis* Compares to *Apollo*." *The Planetary Society* (2026).
[8] U.S. Government Accountability Office, *Assessments of Major Projects: GAO-25-107591* (2025).
[9] U.S. Government Accountability Office, *Cybersecurity: NASA Needs to Fully Implement Risk Management: GAO-25-108138*. (2025).
[10] NASA *Systems Engineering Handbook* (NASA/SP-2016-6105 Rev. 2, 2016), §§5.2–5.4, 6.2, 6.4, 6.5, 6.8; NASA *Risk Management Handbook* (NASA/SP-2011-3422, 2011), §6.4 Technical Risk Management.

- Third, operational concepts can become dependent on infrastructure or cadence assumptions that only become testable late in development, if they become testable at all before first use[11].

In a campaign architecture, uncertainty is rarely confined to one box on an org chart; it is carried across boundaries through schedules, interface agreements, integration narratives, and inherited planning assumptions.

The lunar return effort illustrates these conditions clearly. *Artemis* is shaped not only by the performance of the Space Launch System (SLS), Orion, the Human Landing System (HLS), Gateway, and Exploration Ground Systems (EGS), but also by the development of next-generation Exploration Extravehicular Activity Services (xEVAS) spacesuits and the Lunar Terrain Vehicle (LTV). It is further influenced by emerging surface capabilities. *Artemis* is additionally shaped by assumptions about how often missions can fly, what level of configuration stability can be achieved, how quickly operations can learn, how infrastructure will mature, and which elements of the campaign will remain politically or programmatically durable over time. These are not secondary observations to questions – they directly affect mission design, crew risk, surface utility, cost exposure, and the credibility of the campaign as an architecture rather than as a series of isolated demonstrations.

For that reason, the background problem motivating this paper is broader than isolated technical failure. It is the growing difficulty of maintaining shared meaning, clear ownership, and disciplined risk interpretation in an exploration system whose components are increasingly distributed across institutional, contractual, and temporal boundaries. As human spaceflight becomes more campaign based, the ability to identify and actively govern assumptions becomes a prerequisite for preserving both safety and architectural coherence.

1.2 From Programs to Campaign Architectures

Author's note to readers, and especially to those who built, flew, and sustained Apollo and Shuttle:

In describing the Apollo and Shuttle Programs as comparatively more tightly bounded program structures, this paper does not intend to diminish their complexity or the extraordinary work of the engineers, operators, managers, and crews who made them possible. These were foundational achievements whose technical, organizational, and operational legacies enabled Artemis. The comparison offered here is analytical — meant only to clarify differences in program structure, not to understate the challenges or historic significance of those earlier efforts.

Apollo and Shuttle were NOT "simple" programs; they were extraordinary technical achievements and indispensable predecessors to Artemis. No disrespect is intended — Artemis stands on the shoulders of success, built on the foundations laid by the giants of Apollo and Shuttle.

[11] U.S. Government Accountability Office, *NASA Artemis Programs: Crewed Moon Landing Faces Multiple Challenges*, GAO-24-106256 (Nov. 30, 2023), Highlights pp. i–ii and pp. 13–25.

For the purposes of this paper, the *Apollo* and *Shuttle* programs are best understood as comparatively more tightly bound programs structures than *Artemis*. This is not because they were simple. They were not. Both programs were technically sophisticated, operationally demanding, and organizationally immense. **The distinction is instead architectural.**

In *Apollo*, the core mission logic was concentrated[12] within a relatively legible set of program and organizational elements: the Saturn launch vehicle, the *Apollo* Command and Service Module, the Lunar Module, and the Mission Operations structure that supported a lunar landing objective. In the *Shuttle* era, the central flight system was likewise concentrated within a comparatively bounded transportation architecture composed of the Orbiter, External Tank, and Solid Rocket Boosters[13]. In both cases, the mission critical stack was more centralized, more stable in structural identity, and more readily understood as a single program framework, even though each program still contained substantial internal complexity. Figure 1 illustrates the shift from earlier, more tightly bounded human spaceflight programs to *Artemis* as a distributed campaign architecture composed on interdependent systems, partners, and mission timelines.

Artemis is not best understood as a single vehicle program in the mold of *Apollo* or even *Shuttle*[14]. Earlier U.S human spaceflight efforts were comparatively more centralized, with mission success tied to a smaller number of tightly coupled program elements developed within a more bounded program structure. *Artemis*, in contrast, is organized as a campaign architecture: a broader exploration system composed of multiple vehicles, infrastructure elements, surface capabilities, logistics pathways, operational concepts, and institutional contributors that **must** function together over time[15]. Its success does not depend on one flight system alone, but on the coordinated performance of many interdependent elements distributed across missions and phases of exploration.

This is a fundamental distinction because campaign architectures behave differently than traditional programs[16]. In a single vehicle program, the principal challenge is often the design, verification, and operation of one dominant system and its immediate interfaces. In *Artemis*, however, the architecture is inherently federated. The SLS and Orion provide launch and crew transportation, HLS enables lunar descent and ascent, and Gateway (though currently postponed) was intended to support staging and orbital operations, and surface systems, logistics, communications, and future infrastructure must mature in parallel. None of these elements are fully meaningful in isolation. Their value emerges from the extent to which they can interoperate, demonstrate punctuality, and support an overall progression of missions. Although each system follows its own development path, they

[12] NASA, *Apollo Program Summary Report* (NASA SP-350), Appendix A. (1975)
[13] NASA, *Space Shuttle Program: STS Reference Manual* (1988).
[14] Author-created synthesis based on NASA, *Apollo Program Summary Report*, NASA SP-350 (1975); NASA, *Space Shuttle Program: STS Reference Manual* (1988); and NASA, *Moon to Mars Architecture Definition Document*, ESDMD-001, Revision C, NASA/TP-20250010956 (Dec. 12, 2025), pp. 9, 11–19, 21–32.
[15] NASA, *Moon to Mars Architecture Definition Document*, ESDMD-001, Revision C, NASA/TP-20250010956 (Dec. 12, 2025), pp. 9, 11–19, 21–32.
[16] INCOSE *Systems Engineering Handbook*, 5th ed. (2023): pp. 3–4, 313

must mature in parallel and reach key milestones of readiness within specific timeframes for the campaign to function as intended[17].

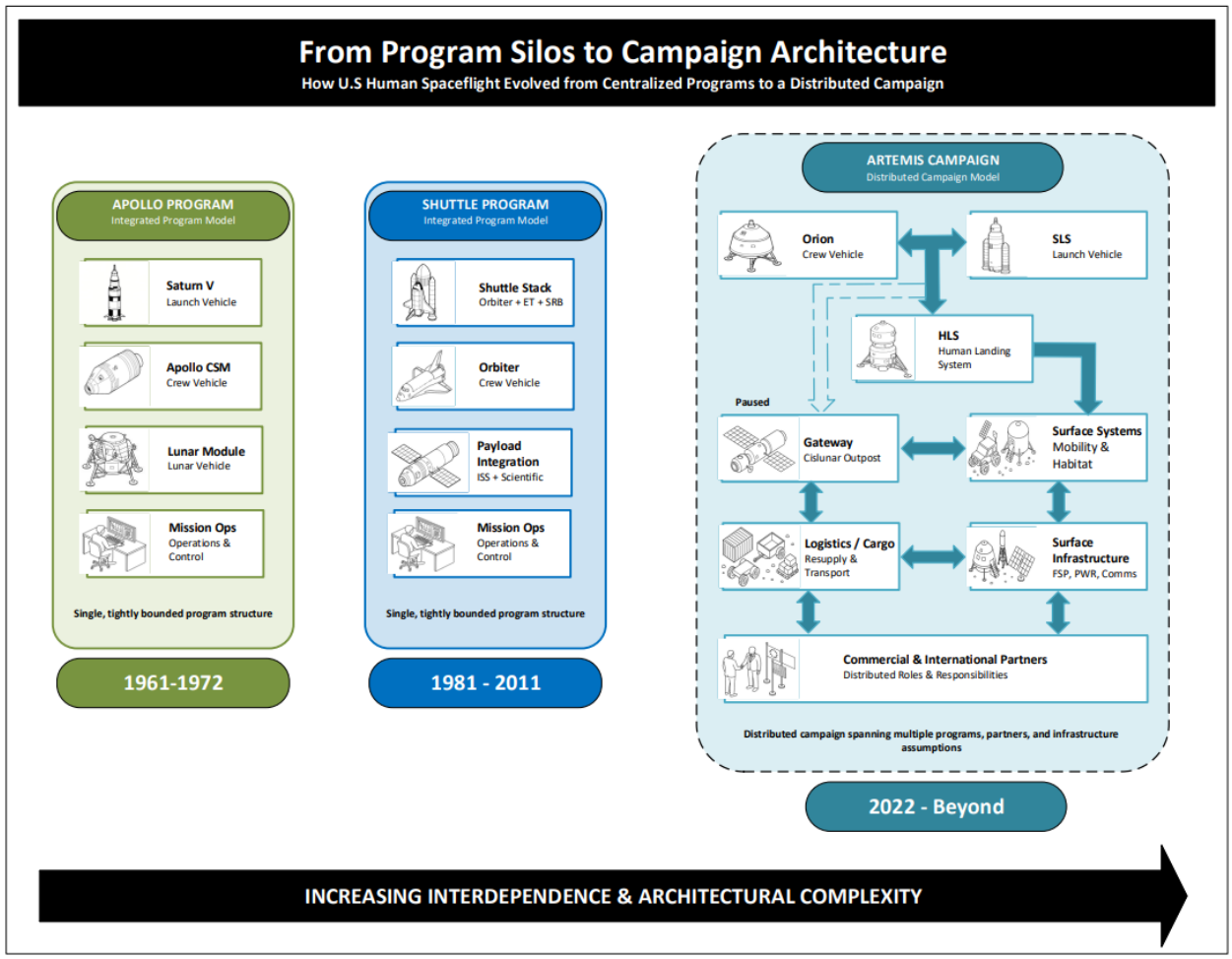


Figure 1 : Architectural Comparison – Apollo, Shuttle, and Artemis

Artemis is therefore better described as a system-of-systems campaign than as a standalone program. Its architecture spans not only hardware but also schedules, organizational boundaries, contractors, international partners, and evolving strategic objectives. This creates a form of interdependence that is broader than technical interface capability alone. Delays in one element can alter assumptions placed on others; changes in architecture can ripple across mission design, logistics, and surface planning; and the campaign's overall credibility depends on whether those moving parts remain aligned. In this sense, *Artemis* represents a shift from human spaceflight programs defined primarily by one integrated vehicle stack to exploration campaigns defined by coordination across distributed capabilities[18].

[17] INCOSE *Systems Engineering Handbook*, 5th ed. (2023): pp. 3–8, 311–313.
[18] NASA *Systems Engineering Handbook* (NASA/SP-2016-6105 Rev. 2, 2016), §§2.4, 5.3–5.4, 6.2, 6.4, 6.5, 6.8.

That shift also increases architectural complexity. A campaign architecture must absorb uncertainty across multiple development paths at once, while still presenting an executable exploration sequence. *Artemis* is not simply building a spacecraft to fly a mission; it is attempting to assemble a sustained exploration framework from interdependent systems that are not always developed on the same schedule, under the same authority, or with the same level of maturity. For that reason, understanding *Artemis* as a campaign architecture is essential to understanding why assumptions, interfaces, and cross program dependencies become central safety and management concerns later in this paper.

1.3 Statutory Commitments as Architectural Assumptions

Artemis did not emerge from a blank slate engineering trade space: It is the product of statutory direction, congressional authorization, industrial-base continuity, national exploration policy, and NASA's evolving approach to human exploration beyond low-Earth orbit (LEO)[19]. This matters because authorization law can stabilize certain architectural choices before every technical, budgetary, or operational dependency has fully matured. Contextually, the NASA Authorization Acts are not exclusively political background but a part of the governance environment in which *Artemis* assumptions are created, preserved, and sometimes constrained.

The NASA Authorization Act of 2010 is especially important because it directed NASA to develop the SLS as a follow-on vehicle to the Space *Shuttle* capable of access to cislunar space and regions beyond LEO[20]. The same Act mandated development of a multipurpose crew vehicle that would advance the human-safety features, designs, and systems of Orion. It also directed NASA, to the extent practicable, to use existing contracts, workforce, industrial base, infrastructure, propulsion capabilities, and *Shuttle*-, Orion-, and Ares-derived assets in developing SLS and the crew vehicle. As a result, SLS and Orion were not only engineering selections; they became statutorily anchored elements of the national exploration architecture. That statutory anchoring provided continuity, but it also created a set of embedded assumptions about launch capability, crew transportation, ground infrastructure, workforce preservation, and the long-term role of government-owned exploration systems.

This distinction is important for assumption management due to the roles played by technical and statutory assumptions. A technical assumption can usually be reopened through design review, test evidence, hazard analysis, or verification results. A statutory-programmatic assumption is harder to reopen because it is reinforced by law, congressional expectation, industrial-base commitments, budget narratives, and institutional continuity. Once those assumptions become part of the architecture, they can shape downstream decisions even when schedule, cost, integration risk, or partner dependencies change. *Artemis* therefore requires governance mechanisms that can distinguish between assumptions that are technically validated and assumptions that are politically or institutionally stabilized.

---

[19] Space Policy Directive-1, sec. 1(a); 82 Fed. Reg. 59501 (The White House, 2017).
[20] NASA *Authorization Act of 2010*, Pub. L. 111-267, §§302–303; NASA *Procedural Requirements NPR 9470.1A* — Budget Execution (NASA, NPR 9470.1A).

The NASA Authorization Act of 2022 further reinforces this point by formalizing the Moon-to-Mars Program and Moon-to-Mars Program Office[21]. NASA describes the office as responsible for hardware development, mission integration, and risk management functions across programs critical to the *Artemis* and Moon-to-Mars exploration approach, including SLS, Orion, EGS, HLS, xEVAS, Gateway, and related deep-space systems. This is directly relevant to *Artemis* because the campaign's risk is not located inside one vehicle or one contractor, it is distributed across interfaces, readiness timelines, partner-provided systems, lunar-orbit operations, surface operations, and return-to-Earth dependencies. The authorization framework therefore recognizes, at least structurally, that *Artemis* requires program-wide integration rather than project-by-project success alone.

The governance problem is then that authorization authority, architecture authority, budget authority, and execution authority do not always move together. Congress may authorize a direction, NASA may define an architecture, appropriations may fund the pieces unevenly, and contractors or partners may mature capabilities on different schedules. A presidential mandate can set national space policy, direct NASA's strategic priorities, and shape the President's Budget Request, but it cannot override funds Congress has already appropriated or legally directed. That creates a governance hole: who owns the architecture if money and strategy point in different directions?[22] **For *Artemis*, this question is not abstract. Annual** cadence, SLS and Orion availability, HLS readiness, Gateway integration, xEVAS maturity, EGS throughput, and surface mission duration all depend on assumptions that cross organizational and funding boundaries. If those assumptions are not actively governed, the campaign can appear coherent on paper while carrying unresolved risk across interfaces.

For this reason, the NASA Authorization Acts should be treated in this paper as part of the *Artemis* governance architecture. They help explain why certain elements are durable, why some design choices are difficult to reopen, and why assumption management must operate above the project level. *Artemis* safety governance is therefore not only about ensuring that individual systems meet requirements. It is about ensuring that statutory direction, funding reality, technical readiness, and crew-survival logic remain aligned as the campaign evolves.

1.4 NASA as a Public Institution: Stakeholders and Democratic Accountability

NASA is a public institution whose mission is officially framed around exploration, innovation and inspiring "for the benefit of all humanity". It is not a private enterprise and not the personal instrument of any single administration[23]. Its missions are funded by taxpayers, authorized through democratic institutions, and justified only insofar as they serve enduring public purposes[24]**. For that reason, the American people are NASA's primary stakeholder.** Presidents, administrators, program managers,

---

[21] NASA *Authorization Act of 2022,* Pub. L. 117-167, § 10802-.

[22] NASA *Governance and Strategic Management Handbook, NPD 1000.0C*, pp. 3, 10, 53–55.

[23] NASA *Act of 1958*, Pub. L. 85-568, 72 Stat. 426, §102(a); NASA Procedural Requirements NPR 9470.1A — Budget Execution (NASA, NPR 9470.1A).

[24] National Aeronautics and Space Act of 1958, Pub. L. No. 85-568, 72 Stat. 426; NASA, *NPD 1000.0C, NASA Governance and Strategic Management Handbook* (updated Jan. 2024), pp. 5, 9–10, 33–43.

centers, contractors, and international partners shape the agency's direction, but none of them are the ultimate beneficiary of the institution. They are temporary stewards of a national capability that exists to serve the public[25].

This distinction is imperative because it clarifies the basis of accountability. NASA does not exist to preserve any single program for its own sake, protect a center from change, satisfy an incumbent administrator's preferences, or perform continuity as an end in itself. It exists to generate public value through exploration, scientific discovery, national capability, technological advancement, responsible stewardship of public funds, and the safe execution of missions whose risks are borne in the public interest[26]. The public therefore remains the enduring stakeholder across political transitions, architectural changes, budget cycles, and leadership turnover[27].

In practice, this means that presidential priorities and administrative directives should be understood as governing inputs rather than as the agency's final purpose. Presidents lawfully set national policies. Administrators are responsible for executing those priorities and managing the institution. Congress appropriates and oversees[28]. **Yet the legitimacy of all these actions depends on whether NASA remains orientated towards the public it serves.** When institutional energy becomes concentrated on internal preservation, bureaucratic self-protection, symbolic milestones detached from durable value, or the maintenance of inherited architectures without reexamination, the agency risks confusing its managers with its mission and its structure with its purpose[29].

This paper treats that confusion as more than a governance problem. It is also an assumption management problem. Complex programs accumulate implicit beliefs about who must be satisfied, what must be protected, which tradeoffs are acceptable, and what counts as success. When those assumptions drift away from the public purpose, decision making can become distorted. Risk may be normalized to preserve schedule credibility. Cost growth may be tolerated to protect institutional agreements. Fragile architectures may persist because they are politically or organizationally difficult to challenge. In such cases, the question moves beyond whether a program is technically executable, but whether it remains accountable to the stakeholder that justifies its existence in the first place[30].

For *Artemis*, this issue is especially important because the campaign is distributed across systems, centers, contractors, political cycles, and competing narratives of value. That complexity makes it easier for stakeholder priorities to fragment and for accountability to diffuse. Reasserting the American public as NASA's primary stakeholder provides an analytical anchor for the rest of the

---

[25] Congressional Research Service, *NASA Appropriations and Authorizations: At a Glance*; NASA Office of the Chief Financial Officer, *appropriations bills and committee reports*; U.S. Constitution, art. I; NASA, *NASA Governance and Strategic Management Handbook* (NPD 1000.0C).

[26] National Aeronautics and Space Act of 1958, Pub. L. No. 85-568, 72 Stat. 426; NASA, *NPD 1000.0C, NASA Governance and Strategic Management Handbook* (updated Jan. 2024), pp. 5, 9–10, 33–43.

[27] Eisenhower Presidential Library, *"Minutes of Cabinet Meeting, October 18 & August 15, 1958"*.

[28] NASA OCFO, "OCFO Appropriations – NASA."

[29] NASA Office of Inspector General, *2025 Report on NASA's Top Management and Performance Challenges* (Jan. 2026), Challenge 3, pp. 8–9, and Challenge 5, pp. 12–13.

[30] Lindbergh, R. (2024). *NASA Appropriations and Authorizations: At a Glance* (CRS Report R43419), Table 1

paper. It creates a basis for evaluating whether assumptions are aligned with public expectations and value, whether governance structures preserve or erode accountability, and whether the campaign architecture is being managed in service of national purpose rather than institutional inertia.

A public facing stakeholder model does not reduce NASA's ambitions: it disciplines them. It requires that bold exploration remains tied to public legitimacy, that technical and organization assumptions remain open to challenge, and that stewardship be measured not by the survival of any particular arrangement, but by whether NASA continues to deliver meaningful value to the people who fund it, and on whose behalf, it flies[31].

1.4.1 Democratic Stewardship and Institutional Continuity

One of the most persistent problems in human spaceflight is the mismatch between political cadence and engineering cadence. Electoral systems reward visible milestones, annual budget control, and near term markers of momentum[32]. By contrast, large exploration systems mature through extended cycles of design, integration, test, verification, rework and operations learning. **These are cumulative processes that cannot be compressed without consequence**[33]. The resulting structural mismatch makes it easier for public ambition to outrun technical maturity and for schedules to become symbolic rather than executable. This is not simply a management inconvenience, but a recurring governance condition that shapes how assumptions are formed and defended. When political systems demand rapid visible progress while engineering systems still require margin, maturation, and iterative learning, assumptions about readiness, risk, and feasibility become especially vulnerable to distortion[34].

A sustainable exploration effort requires durable goals and a pathway resilient enough to survive the normal turnover of political leadership[35]. When each transition attempts to redefine the architecture itself, technical continuity erodes, and program assumptions become contingent on short-lived political signals rather than durable mission logic. Leadership transitions matter not only because they can redirect programs, but because they can interrupt institutional learning before that learning becomes durable. Post-mishap NASA reforms have historically been challenged by leadership instability, signal changes from the top, and the difficulty of refreezing new norms into lasting organizational practices[36]. If leadership turnover repeatedly re-signals priorities, teams can experience a drift in assumptions before verification, accountability, and learning mechanism have

---

[31] U.S. Government, *Budget of the United States Government*, pp. 127–128

[32] NASA *Systems Engineering Handbook* (NASA/SP-2016-6105 Rev. 2, 2016), §§3.0, 5.3; Broniatowski & Weigel, "Political Feasibility and Sustainability in Space Exploration," *Space Policy* 24(3) (2008), pp. 148–157.

[33] Broniatowski," A framework for decision making in complex engineering systems under uncertainty", *Systems Engineering* 15(1) (2012), 48–65.

[34] Broniatowski & Weigel, "Space Policy as a 'Wicked Problem': A Framework for Analyzing the Societal Justification of Space Exploration" *Journal of Aerospace Operations 1(*4) (2012), 325–339.

[35] Broniatowski & Weigel, "Political Feasibility and Sustainability in Space Exploration," *Space Policy* 24(3) (2008), pp. 148–157.

[36] NASA, *Columbia Accident Investigation Board Report, Volume I* (2003), pp. 195–203; Donahue & O'Leary, "Do shocks change organizations? The case of NASA", *Journal of Public Administration Research and Theory,* 22(3). pp. 395–425

time to catch up. Under these conditions, institutional memory and core competencies become fragile and reforms remain provisional.

This is important because assumptions do not live only in requirements documents or trade studies, they are also embedded in *who is empowered to dissent, which concerns are credible, how rigor is enforced, and what forms of uncertainty are tolerated*.

General goodwill towards NASA is real, but it is not the same as sustained public willingness to support expensive long-term exploration architectures[37]. Stakeholders must be able to see progress, understand why it matters, and connect near-term accomplishments to a credible long-term purpose. This is why campaign design is politically relevant, a program that cannot explain its stepping stones, immediate value, or public return, it becomes a lot easier to defer, redesign or reframe programs each time leadership is handed off to someone else. Conversely, a program whose progress is visible, and legible, has a stronger chance of surviving normal political turnover because it does not have to continually justify its existence from the ground up. It would therefore be wise to treat assumptions about stakeholder patience, political support, and acceptable delay as design sensitive assumptions rather than background conditions.

Assumptions in this paper are treated as more than technical propositions about performance, mass, schedule, or operations. In large government agencies like NASA, assumptions also include beliefs about how much political change can be absorbed, which milestones are negotiable, whose confidence must be maintained, and what evidence is sufficient to preserve continuity. Viewed this way, political turnover is not inherently the cause of program failure. The real hazard emerges when routine political change is allowed to reshape architecture, sequencing, budgets, logic, verification expectations, and accountability structures faster than a program can learn and adapt. The problem is not democratic governance; it is the instability created when short-term political shifts unsettle long-range planning. Programs can only gain stability when national goals are durable, resources match execution realities, progress is legible to stakeholders, and technical and safety institutions are strong enough to preserve continuity across administrations[38].

NASA does not need insulation from democratic governance; it needs institutional continuity strong enough to show that it remains a competitive entity and retains its mantle as the powerhouse of space exploration.

[37] National Research Council, *Pathways to Exploration (2014)*, Ch. 1–2; Cameron et al," Goals for space exploration based on stakeholder value network considerations", *Acta Astronautica*, 68(11–12)

[38] Broniatowski & Weigel, "Political Feasibility and Sustainability in Space Exploration," *Space Policy* 24(3) (2008), pp. 148–157; NASA Office of Inspector General, *2025 Report on NASA's Top Management and Performance Challenges* (Jan. 2026), pp. 8–9, 12–13.

1.5 Assumptions as System Elements in Complex Architecture

Assumptions are often treated as background conditions in engineering work whether its planning premises, temporary simplifications, or placeholders that allow development to move forward before all information is available[39]. In complex human spaceflight systems, such a passive treatment of assumptions is too weak to manage risk effectively. Assumptions do not simply surround the architecture, they help constitute it, they shape what designers believe is feasible, what interfaces are expected to hold, what verification approaches are judged sufficient, what operational conditions are anticipated, and what kinds of risk are treated as acceptable. Assumptions should therefore be understood as functional system elements[40]. They may not appear on the hardware bill of materials, but they participate directly in how the system is conceived, integrated, interpreted and flown.

This holds especially true in campaign based exploration architectures, where no single program contains the whole mission logic. In such environments, assumptions become part of the connective structure between elements that are developed on different schedules, by different organizations, and under different constraints[41]. A maturity assumption can migrate into a milestone date. A schedule assumption can harden into a logistics expectation. An operational assumption can shape crew timelines, fault-response planning, communications logic, or staging strategy. These transitions are significant because assumptions rarely remain where they were first introduced. They propagate across technical, organizational, and lifecycle boundaries, often losing visibility even as they gain influence.

Uncertainty is the reason assumptions enter the system in the first place. Human spaceflight programs must make consequential decisions with incomplete knowledge about performance, readiness, environment, funding, partner capability, infrastructure availability, and operational reality. Assumptions make action possible under those conditions, but they do not remove uncertainty. They organize it, defer it, and sometimes obscure it[42]. An assumption is often a decision to proceed as though a condition will hold before that condition is fully demonstrated. That is not inherently irresponsible, it is frequently unavoidable. The problem emerges when uncertainty is hidden inside accepted program logic instead of being kept visible as something contingent, bounded and revisable. When that happens, the architecture begins to rely on conditions it has not yet earned[43].

---

[39] Yang, *Architectural Assumptions and Their Management in Software Development* (PhD diss., University of Groningen, 2018), pp. 132–207; Roeller, Lago, van Vliet, "Recovering Architectural Assumptions," *Journal of Systems and Software* 79, no. 4 (2006), pp. 552–573.

[40] Yang, Liang, and Avgeriou, "A Survey on Software Architectural Assumptions," *Journal of Systems and Software* 113 (2016): 362–380

[41] NASA *Systems Engineering Handbook* (NASA/SP-2016-6105 Rev. 2, 2016), §§5.3, §§6.2, 6.5, 6.8; NASA *Artemis Architecture Definition Document;* NASA *Risk Management Handbook* (NASA/SP-2011-3422, 2011), §6.4

[42] NASA *Systems Engineering Handbook* (NASA/SP-2016-6105 Rev. 2, 2016), §§6.4, 6.8; Broniatowski & Weigel, "Political Feasibility and Sustainability in Space Exploration," *Space Policy* 24(3) (2008), pp. 148–157.

[43] NASA, *Systems Engineering Handbook* Rev. 2 (2016), §6.8.

Assumptions are not peripheral to complex human spaceflight systems. They are a living, breathing part of the architecture's hidden load path.

They carry uncertainty forward, distribute confidence unevenly, and condition whether a campaign remains coherent as it grows in size and complexity[44]. The central danger then is less bounded to the fact that assumptions exist, but more to how they can propagate across the architecture, absorb unresolved uncertainty, shape what is considered verified, and govern operations without remaining visible as governable objects. Treating assumptions as system elements is thus not a rhetorical move, it is a necessary step towards understanding how modern exploration systems hold together, how they become fragile, and where stewardship **must** be strengthened if campaign based human spaceflight is to remain safe and credible over time.

1.6 Research Question and Scope of Paper

Artemis is not simply a return-to-the-moon program in the *Apollo* sense[45]. It is a long duration, politically exposed, commercially distributed, and internationally networked exploration campaign whose architecture, while in the writing of this paper, was still evolving while major systems were being simultaneously designed, certified and integrated. NASA's March 2026[46] architecture update added a new 2027 *Artemis* III LEO demonstration, described a plan to standardize the transportation stack around a common ascent configuration, and stated the agency's intent to move toward one lunar surface mission per year thereafter. At the same time, NASA's *Artemis* III mission page now describes *Artemis* III as a LEO integrated test between Orion and one or both commercial HLS providers, Blue Origin and SpaceX, rather than as a simple continuation of earlier lunar surface mission shape[47]. Taken together, these changes show that *Artemis*'s architecture is not a fixed baseline but a complex moving system whose meaning and outcomes will evolve across time.

The central argument of this paper is that in complex human spaceflight systems, safety risk increasingly emerges not only from component failure but from the mismanagement of assumptions as technical, political, and programmatic contexts evolve across the lifecycle. Verification can show that a system passed a defined test, but it cannot by itself guarantee that the assumptions underlying the test, the mission architecture, and the operational environment have remained stable[48]. The

---

[44] NASA, *Columbia Accident Investigation Board Report, Volume I* (2003), pp. 195–203; NASA *Systems Engineering Handbook* (NASA/SP-2016-6105 Rev. 2, 2016), §§6.4, 6.8, 3.0; NASA *Risk Management Handbook* (NASA/SP-2011-3422, 2011), §3.0.

[45] NASA, *Moon to Mars Architecture Definition Document*, ESDMD-001, Revision C, NASA/TP-20250010956 (Dec. 12, 2025), pp. 9, 11–19, 21–32, 72–82; Broniatowski & Weigel, "Political Feasibility and Sustainability in Space Exploration," *Space Policy* 24(3) (2008), pp. 148–157.

[46] NASA, *"NASA Adds Mission to Artemis Lunar Program, Updates Architecture"* (2026a).

[47] NASA, *Artemis* III mission page (2026a).

[48] NASA Systems Engineering Handbook (NASA/SP-2016-6105 Rev. 2, 2016), NPR 8705.2C Preface P.2 note 1 (NASA, 2017, Preface P.2 note 1, Ch. 1 §§1.1–1.2, Ch. 2 §§2.1–2.3); NASA Risk Management Handbook (NASA/SP-2011-3422, 2011), Ch. 1 §§1.1–1.2 (NASA, 2017, Preface P.2 note 1, Ch. 1 §§1.1–1.2, Ch. 2 §§2.1–2.3); NASA Procedural Requirements 8705.2C — Human-Rating Requirements for Space Systems, Ch. 2 §§2.1–2.3; NPR 8705.2C, Ch. 3 certification/waiver sections (NASA, 2017, Preface P.2 note 1, Ch. 1 §§1.1–1.2, Ch. 2 §§2.1–2.3).

concern is not the verification itself, but whether it *remained valid* after changes in architecture, mission context, and governance.

In campaign based human spaceflight, the central risk is not complexity alone but the accumulation of distributed assumption debt: unverified, inherited, and weakly owned assumptions that propagate across organizational, technical, and lifecycle boundaries until they become safety, integration, and governance hazards.

2. Exploration Campaign Architectures, Policy Drivers, and System Complexity

Large-scale human spaceflight initiatives evolve within a shifting landscape of national policy, technological ambition, geopolitical signaling, and commercial capability. Exploration architectures are therefore never designed in isolation; they are shaped upstream by the strategic objectives that political leadership articulates and downstream by the distributed technical realities of multi-partner implementation[49]. In long-duration programs, such as the *Artemis* exploration campaign, these influences accumulate across years, sometimes decades, resulting in architectures that must continuously reconcile policy intent with engineering feasibility, risk posture, and organizational capacity.

The *Artemis* exploration campaign exemplifies this dynamic. The campaign is not a single program but a federated assembly of missions, systems, and partnerships whose structure reflects both explicit policy commitments and implicit assumptions about how lunar and Mars exploration should unfold[50]. As policy evolves—whether through formal directives, strategic guidance, or international agreements—those changes propagate into architectural decisions, integration patterns, and technical constraints. In turn, the complexity of the campaign reinforces the importance of understanding how early assumptions become embedded, how they shape downstream engineering, and how they must be managed as the system adapts to new political, technical, and institutional realities.

Against this backdrop, Section 2.1 examines how these policy signals manifest within the evolving architecture of the *Artemis* exploration campaign.

---

[49] Broniatowski & Weigel, "Political Feasibility and Sustainability in Space Exploration," *Space Policy* 24(3) (2008), pp. 148–157; NASA Moon to Mars Architecture Definition Document / Strategy and Objectives (NASA, n.d., Strategy and Objectives section).

[50] NASA Office of Inspector General, *NASA's Partnerships with International Space Agencies for the Artemis Campaign*, IG-23-004 (Jan. 17, 2023), pp. 1–5, 9–18; NASA, *Moon to Mars Architecture Definition Document*, ESDMD-001, Revision C, NASA/TP-20250010956 (Dec. 12, 2025), pp. 9, 11–19.

2.1 The *Artemis* Exploration Campaign as an Evolving Architecture

American lunar exploration policy has long tied a human return to the Moon to sustained presence, commercial and international participation, and eventual Mars missions. The 2017 direction associated with Space Policy Directive-1 (SPD-1) reframed U.S. exploration strategy around returning astronauts to the Moon and using the lunar environment as the foundation for later Mars expeditions[51]. SPD-1 also initiated what became the *Artemis* Program, establishing its core architecture, launch cadence, and objectives; identifying SLS, Orion, Gateway, commercial Human Landing Systems, Commercial Lunar Payload Services, and xEVAS as essential building blocks; and shifting NASA toward a federated, Moon-to-Mars campaign.

Subsequent policies further shaped the boundaries of *Artemis*. The 2020 national strategy associated with Space Policy Directive-6 (SPD-6) linked space nuclear power and propulsion to scientific, exploration, national security, and commercial priorities[52]. The *Artemis* Accords introduced a principles-based framework intended to guide civil activities on and around the Moon as more nations and private actors enter the lunar domain.

These policies do not prescribe detailed engineering solutions, but they do define the solution space by embedding assumptions before specific architecture decisions are made. Once policy designates the Moon as the proving ground for Mars, emphasizes commercial and international participation, or prioritizes sustained lunar presence, subsequent engineering decisions begin to inherit and reinforce those assumptions. Individual systems and providers thus become more than standalone program elements—they become carriers of embedded policy logic. Section 2.2 therefore examines how these assumptions move from policy language into the federated program that *Artemis* now depends on.

2.2 Federated Program Structure in *Artemis*

NASA's March 2026 Ignition[53] advisory on implementation of the National Space Policy (SPD-1) is especially important because it provides visibility into what often is only implicit in technical programs**: national policy is actively being translated into architecture and campaign priorities in real time.** Administrator Isaacman stated that the event would explain how the agency is implementing the National Space Policy while accelerating preparations for an American return to the lunar surface by 2028. It also identified priorities such as returning astronauts to the lunar surface by 2028, establishment of the initial elements of a permanent lunar base and advancing nuclear propulsion, with participation from the Administrator, the Associate Administrator, the program executive for the Moon Base and the program executive for Fission Surface Power (FSP).

---

[51] Space Policy Directive-1 — Reinvigorating America's Human Space Exploration Program (The White House, 2017, § 1(a)); 82 Fed. Reg. 59501; NASA, *Moon to Mars Architecture Definition Document*, ESDMD-001, Revision C, NASA/TP-20250010956 (Dec. 12, 2025), pp. 9, 11–19, 21–32.

[52] Space Policy Directive-6 — *National Strategy for Space Nuclear Power and Propulsion* (The White House, 2020, secs. 1–4); 85 Fed. Reg. 82367.

[53] NASA, *Artemis Ignition Architecture Update* (2026), pp. 1–2, 4.

A federated program structure refers to an architecture in which multiple mission elements, directorates, centers, or partner organizations operate with partially independent authorities, budgets, schedules, and verification pathways[54]. Rather than functioning as a single vertically integrated program with unified control, a federated structure distributes ownership of key systems across semi-autonomous units. This creates both flexibility and fragmentation: decisions can be made closer to where the work occurs, but assumptions, constraints, and priorities can diverge unless deliberately synchronized. In a setting like *Artemis*—where landers, suits, launch systems, power systems, logistics chains, and surface infrastructure are owned by different entities.

Federation means that policy shifts at the top can propagate unevenly across the architecture, resurfacing hidden inconsistencies or re-opening design assumptions that were previously treated as stable.

This not only supports but reinforces that policy is not a passive backdrop; it is an upstream source of technical assumptions in the *Artemis* Campaign[55]. The Ignition event makes clear that policy is an active driver capable of reopening or reshaping architectural assumptions. When NASA publicly links lunar return timelines, lunar base development, and nuclear propulsion to the implementation of national space policy, it demonstrates that assumptions about mission sequence, infrastructure timing, campaign cadence, and verification scope are being influenced from the top of the system as much as from bottom-up engineering. This interpretation follows directly from NASA's own description of the Ignition event and its stated agenda. *Artemis* is a live case of top-down assumption redefinition, where NASA is visibly trying to convert a more brittle exploration sequence into a phased repeatable, risk reduced campaign.

2.3 Distributed Development Across Programs and Partners

Artemis is not being developed as a single tightly bounded program in the manner of *Apollo*, nor even as a more operationally centralized system like *Shuttle*. It is being built through a distributed development environment that spans NASA centers, major prime contractors, commercial providers, international partners and mission specific support organizations[56]. This distribution changes how architectural decisions are made, how interfaces are interpreted, and how assumptions propagate across the campaign.

Technical alignment cannot be assumed within this distribution just because each contributing element is individually well managed[57]. Interfaces between programs become sites where responsibility, evidence, and expectations can change. A provider may mature its own system, against local requirements and constraints while still relying on campaign-level assumptions about

[54] Mark W. Maier, "Architecting Principles for System of- -Systems," *Systems Engineering* 1, no. 4 (1998): 267–284.
[55] NASA, *Moon to Mars Architecture Definition Document*, ESDMD-001, Revision C, NASA/TP-20250010956 (Dec. 12, 2025), pp. 9, 11–19, 21–32.
[56] NASA Office of Inspector General, *NASA's Management of the Artemis Supply Chain*, IG-24-003 (Oct. 19, 2023), pp. 1–5, 8–21.
[57] NASA *Systems Engineering Handbook* (NASA/SP-2016-6105 Rev. 2, 2016), §§5.3, 6.4, 6.5; NASA *Risk Management Handbook* (NASA/SP-2011-3422, 2011), §6.4.

timing, integration logic, communications, operations, logistics, or safety that are owned and housed elsewhere. The result is that assumptions do not remain confined within a single design team, they travel across organizational boundaries and can eventually harden into planning logic before they are fully reconciled at the campaign level.

This is where systems engineering becomes more than technical coordination. From the perspective of The International Council of Systems Engineering (INCOSE), *Artemis* must be understood as a system-of-systems in which stakeholder alignment, interface definition, lifecycle coordination, and architectural governance are all necessary to preserve mission coherence. Architecture in this environment functions not only as a technical structure, but also as a cross-organizational constraint. It defines how independently developed systems are expected to fit together, what evidence will count as sufficient at their boundaries, and which assumptions must remain visible if the campaign is to avoid downstream surprises.

For assumption management, the implication is fairly direct: distributed development increases the risk that assumptions become embedded in interfaces without a single authority continuously tracking their meaning across organizations[58]. What appears to be a coordination risk is therefore also an assumption risk. In *Artemis*, contractor structure, interface complexity, and stakeholder alignment are not background noise, they are part of the mechanism through which architectural assumptions propagate.

2.4 The *Artemis* Accords and International Governance Assumptions

Local contractor considerations are not the only entity that impacts *Artemis*. Technical systems are being developed within an international framework shaped by long standing space law and political instruments intended to coordinate behavior around the Moon[59]. The most important of this is that the background legal framework remains the 1967 Outer Space Treaty, which establishes principles including peaceful, non-appropriation, state responsibility for national activities, and avoidance of harmful interference[60]. Within that broader context, sits the *Artemis* Accords – complimenting the Outer Space Treaty as opposed to replacing it. The Accords affirms that NASA intends to reinforce commitments to the Outer Space Treaty, The Registration Convention, and The Rescue and Return Agreement while providing a common set of principles for civil exploration and use of outer space as activity around the Moon increases.

The *Artemis* Accords, here, matter less as a diplomatic headline than as a source of campaign level assumptions. NASA and the U.S. Department of State (DOS) launched the Accords in 2020 with eight initial signatories. Since then, the Accords have been further strengthened, with Paraguay being the 67th Nation to sign on May 7th, 2026. The Accords have been described by the DOS as a non-binding

[58] NASA *Systems Engineering Handbook*, SP-2016-6105 Rev. 2, §6.4; §6.8.
[59] NASA, The *Artemis* Accords: *Principles for Cooperation in the Civil Exploration and Use of the Moon, Mars, Comets, and Asteroids for Peaceful Purposes* (2020).
[60] United Nations Office for Outer Space Affairs, *Treaty on Principles Governing the Activities of States in the Exploration and Use of Outer Space* (Outer Space Treaty), arts. I, III, VI, IX.

set of principles grounded in the Outer Space Treaty and designed to guide civil space exploration and use in the twenty first century - this combination is important[61].

The Accords are politically meaningful and operationally influential, but they are not a substitute for technical maturity, binding program commitments, or guaranteed long term policy continuity.

Several of these assumptions are especially important for this paper:

- *Interoperability*[62] assumes that participating actors will either use existing standards or help develop and follow new ones.
- *Transparency*[63] assumes that national policies and exploration plans will be sufficiently visible to misunderstanding and coordination failures.
- *Deconfliction*[64] assumes that actors will provide information about the location and general nature of their activities and will coordinate to avoid harmful interference, which NASA links to temporary "safety zones" grounded in scientific and engineering principles.

These principles are embedded with value but introduce boundless uncertainty: interoperability may vary in depth, transparency may be constrained by national policy or industrial sensitivity, and deconfliction mechanisms may prove straightforward in principle but difficult in fast moving operational settings.

The broader legal framework reinforces this point: Under the Outer Space Treaty, nations remain internationally responsible for national space activities whether those activities are conducted by governmental agencies or non-government entities[65]. These responsibilities are further shaped by accompanying agreements that specify how states must manage interference, assist personnel, and identify space objects.

- *Article IX* requires states to conduct activities with due regard for the interests of other nations and to engage in consultation when their operations may cause potentially harmful interference, reinforcing the Outer Space Treaty's emphasis on responsible conduct and mutual transparency.
- *The Rescue Agreement* elaborates the obligation to assist astronauts in distress, operationalizing the Outer Space Treaty's principle that astronauts are 'envoys of humankind' and must be given all possible assistance and safely returned to their launching authority.

[61] NASA, The *Artemis* Accords; Broniatowski & Weigel, "Political Feasibility and Sustainability in Space Exploration," *Space Policy* 24(3) (2008), pp. 148–157.
[62] NASA, The *Artemis* Accords — Section 4 (*"Interoperability"*)
[63] NASA, The *Artemis* Accords — Section 3 (*"Transparency"*)
[64] NASA, The *Artemis* Accords — Section 9 *("Deconfliction of Space Activities"*)
[65] United Nations Office for Outer Space Affairs, *Treaty on Principles Governing the Activities of States in the Exploration and Use of Outer Space* (Outer Space Treaty), arts. I, III, VI, IX.

- *The Registration Convention* strengthens the Outer Space Treaty's framework by requiring the registration of space objects, enabling identification, accountability, and coordination among states as space activities expand.

In other words, *Artemis* operates in a governance environment where technical operations, state responsibility, partner coordination, and legal obligations are tightly coupled. Assumptions about partner behavior are not "just" political assumptions, they are system-relevant assumptions with implications for operations, accountability, and safety.

For international and interorganizational assumptions, the *Artemis* Accords should be distinguished from executable implementation agreements. NPR 9090.1C governs partnership-agreement financial requirements and administration, including roles and responsibilities, agreement financial documentation, budget and execution, full cost, and reimbursable agreement pricing (NASA, 2023/2028, NPR 9090.1C, Ch. 1-4). This does not diminish the Accords; rather, it supports the distinction between shared principles and the formal agreements, budgets, interfaces, and evidence required to turn principles into operational capability.

For a campaign like *Artemis*, architectures are vulnerable when political alignment is mistaken for durable operational assurance[66]. A nation may sign the Accords without capacity for contribution, or an international partner may support its principles while still facing domestic concerns ranging from budget instability, industrial constraints, or policy turnover. A shared commitment to shared interoperability or transparency does not guarantee that systems will be ready on the same timeline, that interfaces will mature at the same rate, or that operational coordination will be equally robust across all participants. In assumption management terms, international partnership frameworks *can reduce uncertainty at the level of norms while leaving substantial uncertainty at the level of implementation*. The resulting risk is that diplomatic alignment may be unconsciously treated as a proxy for technical readiness or campaign coherence. That would be a categorical error[67].

The *Artemis* Accords should therefore only be understood in this paper as a source of both strength and exposure. They strengthen *The Artemis Exploration Campaign* by creating a shared political vocabulary for peaceful, transparent, and coordinated exploration and by situating the campaign within a wider coalition rather than a purely national endeavor[68]. Juxtaposed with this, they expose the campaign to a new class of assumptions: assumptions about continuity across sovereign actors, assumptions about consultation and deconfliction in contested operational settings, assumptions about standardization across technically diverse systems, and assumptions about whether political commitments will remain aligned with evolving architecture and mission cadence. Recognizing and addressing these assumptions early is essential, but as an international exploration system whose coherence depends partly on relationships that are outside of NASA's direct control.

---

[66] The *Artemis* Accords (NASA); Broniatowski & Weigel (2008), Space Policy 24(3), pp. 148–157.
[67] McKeown, "*Artemis* Accords: Are Safety Zones Practical for Long-Term Commercial Lunar Resource Utilisation?," *Space Policy* 60 (2022).
[68] The *Artemis* Accords (NASA); Broniatowski & Weigel (2008), Space Policy 24(3), pp. 148–157.

2.5 Infrastructure Based Exploration Systems and Surface Power Assumptions

Artemis differs from earlier human spaceflight efforts due to its increasing dependency on infrastructure based exploration rather than on isolated mission events[69]. In this kind of campaign, infrastructure is not considered secondary to exploration, it is however what makes repeated exploration possible. All the bells and whistles surrounding surface power, communications, mobility support, logistics, staging, and long duration habitation are not being separated from the architecture[70]. They are being integrated into it and enabling the conditions that shape what missions can be attempted, how often they can be attempted, and under what operational margins they can be sustained.

Infrastructure produces a distinct set of assumptions that render this significant. Vehicle assumptions are often visible because they are tied to a specific flight element, certification basis, or mission milestone. Infrastructure assumptions are more diffuse and often appear as background expectations: power will be available when needed, communications will support distributed operations, mobility assets will extend range, and surface systems will be sufficiently mature to turn initial demonstrations into repeatable campaign capability. Yet those assumptions constrain the architecture just as strongly as any launch vehicle or crew system requirement.

Fission Surface Power (FSP) is a great example of this. FSP is not an isolated technological development; it is a cross cutting architectural enabler[71]. Assumptions about available power affect communications design, surface mobility, science operations, habitat planning, thermal management, and the validity of longer duration lunar presence. Once power availability is assumed in campaign logic, downstream systems begin to mature against that assumption even if the infrastructure itself remains immature, rephased, or reinterpreted. The implication is that infrastructure assumptions carry outsized weight because they frequently cascade into numerous dependent systems at the same time.

An INCOSE informed reading sharpens this further; infrastructure should be treated as a lifecycle system element whose maturity, interfaces, and operational role must remain traceable across campaign phases[72]. Early concept choices about enabling infrastructure to create downstream architectural constraints that later verification and operations cannot easily undo. For assumption management, the key lesson is that infrastructure assumptions must remain explicit, because once they become embedded in the campaign baseline they influence not only what *Artemis* intends to do, but what later elements are possible.

---

[69] NASA *Moon to Mars Architecture Definition Document* - Strategy and Objectives.
[70] NASA *Moon to Mars Architecture Definition Document;* NASA FSP Fast Facts (NASA, 2021/updated); Oleson et al; NASA U*nveils Initiatives to Achieve America's National Space Policy - Artemis* Ignition Architecture Update.
[71] Space Policy Directive-6 — National Strategy for Space Nuclear Power and Propulsion (The White House, 2020, secs. 1–4); 85 Fed. Reg. 82367
[72] NASA *Systems Engineering Handbook*, SP-2016-6105 Rev. 2, §3.0; NASA *Artemis Architecture Definition Document;* INCOSE Infrastructure Working Group, *Applying Systems Engineering to Industrial & Infrastructure Projects: Systems Integration*, INCOSE-PI-2015-002-1.0.

3. Assumption Management as a Safety Discipline

Complex human spaceflight and complex engineering systems share a defining vulnerability: poorly managed assumptions have contributed to major engineering failures, including those resulting in loss of life[73]. Assumptions are necessary to enable progress when knowledge is incomplete, but if these are incorrect, outdated, or unexamined, they can create risks ranging from cost growth and redesigns to schedule slips, degraded performance, and catastrophic failure[74]. Safety depends not only on technical performance but on how well project teams understand, control, and revisit the uncertainty embedded in their own reasoning[75].

Assumptions that are accepted as true without complete verification are an unavoidable part of early analysis, design decisions, operational planning, and risk evaluation[76]. While they allow programs to move forward in the absence of complete information, they also introduce hidden vulnerabilities known as Assumption Drift—this is the quiet mismatch that grows when assumed conditions diverge from the system's actual state. To see how this discipline functions within complex engineering systems, three core elements are especially important:

- **Assumption Management** provides the disciplined framework required to prevent these vulnerabilities from maturing into safety hazards[77]. When assumptions are treated as controlled, traceable artifacts—rather than remaining implicit judgments even when informed by experience—engineering teams can ensure that each assumption is visible, justified, monitored, and ultimately validated or retired. This discipline supports safer decision-making by making the boundaries of knowledge explicit and minimizing unverified assumptions from silently shaping critical design or operational decisions.
- **Assumption Drift** is the progressive divergence between an assumption made during system design, planning, or analysis and the real conditions, constraints, or data that emerge over time. Drift occurs when assumptions remain unvalidated, are not periodically reviewed, or naturally become outdated as new information, constraints, or operating conditions develop. Left unchecked, this divergence propagates through requirements, design decisions, safety analyses, verification logic, and operational procedures, creating latent hazards and eroding system safety, reliability, and performance.

---

[73] NASA, *Columbia Accident Investigation Board Report, Volume I* (2003), p. 9; Ch. 7, pp. 177–203; NASA, *Report of the Presidential Commission on the Space Shuttle Challenger Accident, Volume I* (1986); Diane Vaughan, *The Challenger Launch Decision* (1996); Nancy Leveson, *Engineering a Safer World* (2011).
[74] NASA Systems Engineering Handbook, SP-2016-6105 Rev. 2, §6.4; §6.8; NASA Risk Management Handbook, NASA/SP-2011-3422 (2011), §6.8.
[75] NASA, *Systems Engineering Handbook*, NASA/SP-2016-6105 Rev. 2 (2016), §§5.2–5.4, §6.4, §6.8; NASA, *NPR 8705.2C, Human-Rating Requirements for Space Systems* (updated Jan. 19, 2024), Preface P.1–P.2, pp. 4–6; *Columbia* Accident Investigation Board, *Report Volume I* (2003), pp. 195–203.
[76] NASA, *Systems Engineering Handbook*, SP-2016-6105 Rev. 2, §5.3.
[77] Chen, Liang & Avgeriou, "Evaluation of a Process for Architectural Assumption Management in Software Development," *Science of Computer Programming* 168 (2018), 38–70.

- **Traceability** is a foundational element of Assumption Management—the ability to link each assumption back to its parent requirement, analysis, risk rationale, and design intent[78]. Traceability ensures that when an assumption changes, teams can immediately identify downstream impacts, maintain configuration integrity, and update safety assessments accordingly. This level of rigor is enabled through systematic identification, documentation, validation planning, and lifecycle stewardship. When used in this way, Assumption Management strengthens the overall safety posture of complex engineering endeavors and helps ensure that no assumption fails silently.

Assumption Management, Assumption Drift, and Traceability collectively demonstrate why assumptions must be treated with the same rigor as any other safety-relevant element of a complex system. To move from concepts to practice, this discipline requires a structured process that makes assumptions explicit, traceable, and continuously evaluated throughout the system lifecycle. The Assumption Management Process shown below provides this operational framework, detailing the step-by-step activities that ensure assumptions are identified, documented, monitored, validated, updated, and ultimately acted upon in a controlled and repeatable way[79].

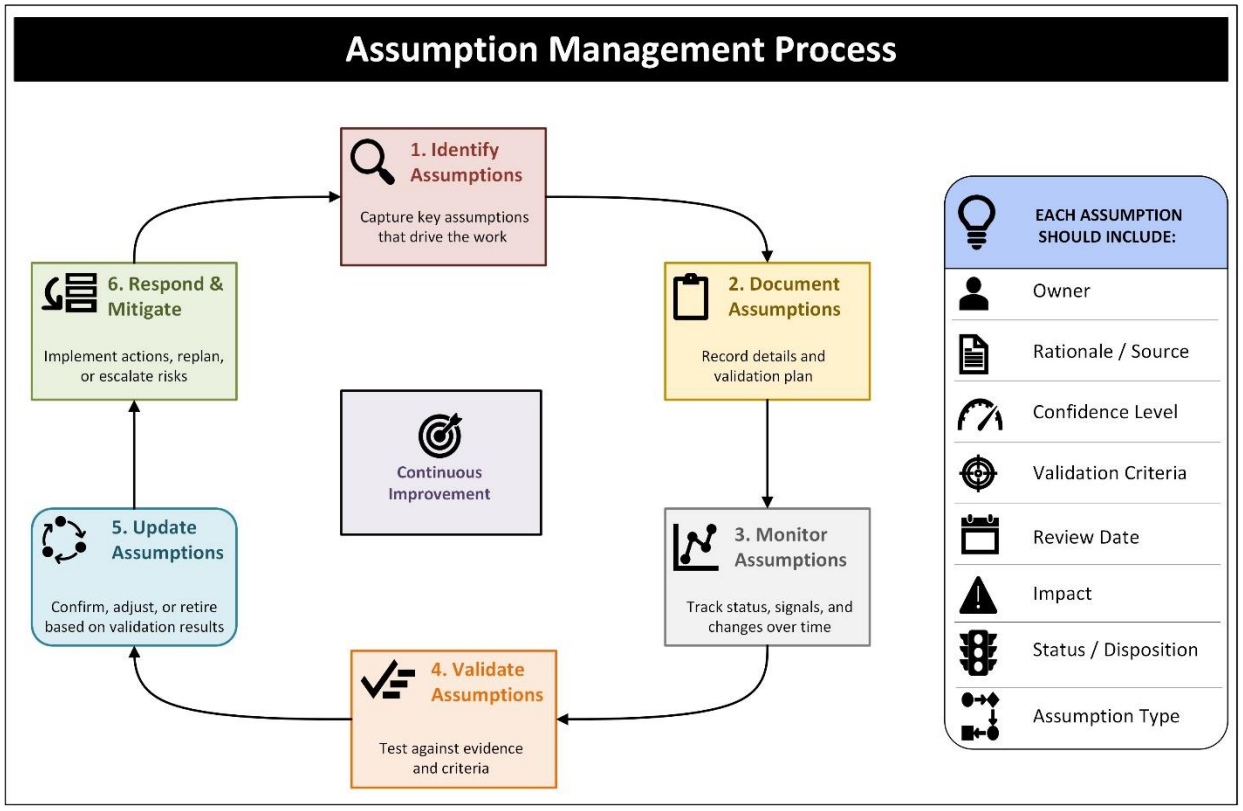


Figure 2 : Assumption Management Process

[78] N *ASA Systems Engineering Handbook*, SP-2016-6105 Rev. 2§6.2; §6.4; §6.5; §6.8; NASA *Risk Management Handbook,* NASA/SP-2011-3422 (2011), §6.2, §6.8.
[79] NASA *Systems Engineering Handbook*, SP-2016-6105 Rev. 2, §6.2; §6.4; §6.5; §6.8; NASA *Risk Management Handbook*, NASA/SP-2011-3422 (2011), §6.2.

The Assumption Management Process provides a structured, repeatable method for identifying, documenting, validating, and updating the unverified beliefs that influence system behavior and engineering decisions. To use this process effectively, it is essential to clarify how assumptions differ from requirements and constraints, why they are safety-relevant, and how they evolve across the lifecycle.

Distinguishing Assumptions, Requirements, and Constraints

- Assumptions – Unverified beliefs about system behavior, operating environments, interfaces, or user actions that fill gaps in knowledge and enable work to proceed.
- Requirements – Statements of needed system functions or performance characteristics that must be satisfied.
- Constraints – Fixed limitations the system must comply with, such as physical boundaries, budgets, standards, or mandated technologies.
- Assumptions differ because they begin uncertain, may later be confirmed or disproven, and influence design decisions even before supporting evidence exists.

Safety-Relevance

- Assumptions introduced early can persist through requirements, design, verification, and operations unless they are explicitly managed.
- Incorrect or outdated assumptions may lead to latent hazards, degraded performance, or invalid verification results.
- The cyclical flow in the figure—Monitor → Validate → Update → Respond—supports continuous evaluation as the system matures and new information becomes available.
- Managing assumptions prevents "silent drivers" of design from remaining unchallenged throughout the lifecycle.

Assumption Types (Mapped to "Assumption Type" in Figure 2)

Assumptions fall into several categories, each requiring different validation strategies:

- Paradigmatic Assumptions – Deep, often implicit beliefs about how the world works (e.g., scientific paradigms, organizational norms).
- Prescriptive Assumptions – Beliefs about how things should be or how actors ought to behave.
- Causal Assumptions – Beliefs that one factor directly causes another, often underlying models and predictions.

Documenting assumption type improves traceability and clarifies how validation should be performed.

The Assumption Engine and Its Alignment with the Process

The Assumption Management Process functions as an assumption engine—a mechanism that systematically transforms implicit beliefs into explicit, testable knowledge:

- Identify (Step 1) – Surfaces hidden or implicit beliefs that drive design and risk decisions.

- Document (Step 2) – Captures rationale, confidence, validation criteria, ownership, review date, impact, status, and type using an assumption log or similar tool.
- Monitor (Step 3) – Tracks changes in evidence, risk exposure, system behavior, and external conditions (assumption drift).
- Validate (Step 4) – Tests assumptions against data, analysis, models, prototypes, or operational results.
- Update (Step 5) – Confirms, adjusts, or retires assumptions as new knowledge emerges.
- Respond & Mitigate (Step 6) – Initiates design updates, risk escalations, or replanning when an assumption proves invalid.

This continuous loop creates the "engine" that turns uncertainty into validated system knowledge. With this foundation, the next section clarifies the distinctions between assumptions, requirements, and verification evidence, and explains how each contributes uniquely to traceability, system integrity, and safety assurance.

3.1 Assumptions vs. Requirements vs. Verification Evidence

This section defines assumptions in engineering systems and distinguishes them from requirements and verification evidence. These elements are often conflated, yet each serves a distinct function in managing uncertainty, shaping design decisions, defining system behavior, ensuring safety, and supporting traceability across the lifecycle.

3.1.1 Assumptions

Assumptions are **statements accepted as true without proof at the time of definition[80].** They address unknowns and uncertainties that must be temporarily stabilized to allow progress in design, analysis and planning.

Key Characteristics

- Represent unverified beliefs or conditions.
- Carry inherent **risk**, as they may later prove incorrect.
- Must be explicitly documented, tracked, and periodically revisited.
- Often exist as the earliest inputs to trade studies, hazard analyses, and architectural decisions.
- May persist across lifecycle phases unless proven false or validated with evidence.

Types of Assumptions

- Paradigmatic (deep worldview; often unconscious)
- Prescriptive (beliefs about how things should operate)
- Causal (beliefs about cause and effect relationships)

[80] NASA, *Systems Engineering Handbook*, SP-2016-6105 Rev. 2, §6.8.

Each assumption should be assigned a confidence rating (e.g., high/medium/low) based on supporting rationale or preliminary evidence. Lower-confidence assumptions indicate greater programmatic, and safety risks. Assumptions are not the same as requirements.

3.1.2 Requirements

Requirements are **mandatory capabilities, conditions, or performance specifications** the system must achieve[81] . A requirement is explicit, allocated, and usually traceable to a verification method. They are authoritative statements derived from stakeholder needs, mission objectives, and regulatory obligations.

Key Characteristics

- Represent verifiable, testable expectations[82].
- Do not carry "confidence levels"; they are commitments.
- Must be met for the system to succeed.
- Drive design, implementation, verification, and validation activities.
- Differ fundamentally from assumptions because requirements must be proven, not taken on trust.

Unlike requirements, assumptions can seep into a system indirectly through architectural decisions, baseline definitions, expectations about partner maturity, production cadence, infrastructure availability, or inherited beliefs about what earlier testing proves. This distinction is important because complex exploration systems may fully meet their documented requirements while still relying on unspoken assumptions that can become outdated, strained, or abandoned as the program evolves. In other words:

A system can be compliant on paper while still becoming unsafe in context if the assumptions surrounding that compliance are not revised.

3.1.3 Verification Evidence

Verification evidence is **objective, reproducible information demonstrating that requirements have been satisfied**. Evidence is produced through inspection, analysis, demonstration, or testing.

Key Characteristics

- Must directly trace back to individual requirements[83].
- Eliminates uncertainty by showing measurable compliance.
- Transforms assumptions into validated knowledge when used to test assumed conditions.
- Required for certification, safety assurance, and regulatory compliance.

---

[81] NASA *Systems Engineering Handbook,* SP-2016-6105 Rev. 2; NASA *Procedural Requirements NPR 7123.1D* - Systems Engineering Processes and Requirements, Ch. 2; app. C.
[82] NASA, *Systems Engineering Handbook*, NASA/SP-2016-6105 Rev. 2 (2016), §5.3; NASA Office of Inspector General, *NASA's Management of the Human Landing System Contracts*, IG-26-004 (Mar. 10, 2026), pp. i–ii, 12–16, 29–35; Aerospace Safety Advisory Panel, *2025 Annual Report* (Jan. 2026), pp. 13, 15, 18–24.
[83] NASA, Systems Engineering Processes and Requirements, NPR 7123.1D, ch. 2; Appendix C.

Role in the Assumption Lifecycle

- Evidence increases or decreases confidence in an assumption.
- Strong evidence may allow an assumption to be retired or converted into a confirmed fact.
- Lack of evidence requires continued tracking and risk mitigation.

In programs as large as *Artemis*, where various directorates and organizations may verify different portions of capability under heterogeneous methods, constraints, and timelines, understanding this distinction becomes imperative. Each body of evidence may be locally valid, while the campaign as a whole still depends on assumptions about how those pieces will interact in reality[84]. That gap between local evidence and architectural confidence is often bridged by assumptions about compatibility, sequencing, fault handling, logistics, mission cadence, or the availability of supporting infrastructure. When those bridging assumptions remain implicit, confidence on paper can grow faster than confidence in practice[85]. The architecture may look increasingly complete while remaining structurally brittle.

3.2 Assumption Drift in Long Duration Programs

Assumption drift occurs when a provisional assumption gradually becomes treated as fixed over time, even as the surrounding technical or organizational context evolves[86]. This mechanism—drift, time, change, and the persistence of outdated beliefs—links the historical cases examined later in this paper. *Challenger* shows how an engineering concern can be normalized into an accepted boundary condition. *Columbia* illustrates how accumulated operational experience can harden into an operational rule. *Starliner* demonstrates how qualification and oversight assumptions can outlast the evidence that initially supported them. In long-duration programs, this drift is amplified by architectural evolution, policy shifts, workforce turnover, and the uneven persistence of institutional memory[87].

For *Artemis*, assumption drift is especially important because architecture, policy, partner maturity, and campaign cadence are all evolving concurrently rather than sequentially, increasing the likelihood that inherited assumptions may outlive the conditions that once justified them.

[84] NASA, *Risk Management Handbook*, NASA/SP-2011-3422 (2011), §6.4; NASA Office of Inspector General, *NASA's Management of the Human Landing System Contracts*, IG-26-004 (Mar. 10, 2026), pp. 12–16, 29–35; Aerospace Safety Advisory Panel, *2025 Annual Report* (Jan. 2026), pp. 13, 15, 18–24.

[85] NASA, *Columbia Accident Investigation Board Report, Volume I* (2003), §5.4; §6.4; §6.8; NASA *Systems Engineering Handbook*, SP-2016-6105 Rev. 2, §5.4; NASA R*isk Management Handbook*, NASA/SP-2011-3422 (2011), §6.4; Nancy Leveson, *Engineering a Safer World* (2011).

[86] NASA, *Columbia Accident Investigation Board Report, Volume I* (2003); NASA, *Report of the Presidential Commission on the Space Shuttle Challenger Accident, Volume I* (1986); NASA *Systems Engineering Handbook,* NASA/SP-2016-6105 Rev. 2 (2016); Diane Vaughan, T*he Challenger Launch Decision* (1996).

[87] Broniatowski & Weigel, "Political Feasibility and Sustainability in Space Exploration," *Space Policy* 24(3) (2008), pp. 148–157; Donahue & O'Leary (2012) Donahue & O'Leary (2012). Do shocks change organizations? The case of NASA. *Journal of Public Administration Research and Theory*, 22(3), pp. 395–425

Verification is one of the places where this dynamic becomes most consequential as it is commonly treated as the means by which requirements are shown to have been satisfied. Verification, however, is never assumption-free[88]. Test conditions are chosen based on expectations about use, loading, environment, and failure significance. Analyses depend on model fidelity, parameter selection, simplifications, and accepted margins. Inspections and demonstrations depend on prior judgements about what must be seen, what counts as representative, and what constitutes as enough evidence. Verification, in other words, does not only assess the system, but it also reflects the assumptions under which the system has been defined and evaluated. A program can therefore accumulate verified items while still **lacking** true system level confidence if the assumptions connecting those items remain weakly examined.

Operations expose the same problem from the other side of the lifecycle[89]. Assumptions shape not only what is built or how it is verified, but also how the system is expected to behave once it is used. They influence what operators are prepared to see, what is treated as normal, what anomalies are considered credible, and which workarounds are assumed to be available. Operational concepts are themselves constructed from assumptions about crew workload, communications continuity, autonomy, ground responsiveness, logistics support, environmental stability, and the reliability of adjacent systems. When those assumptions are weak, outdated, or inherited without revalidation, operations become the place where hidden fragility finally surfaces. Assumptions are more than "design time" conveniences; they are active participants in mission execution.

Assumption propagation refers to the process by which an initial belief, simplification, constraint, or design judgment spreads beyond the context in which it was first made and begins to shape later decisions, interfaces, analyses, and operational expectations. In complex programs, this spread is rarely explicit. An assumption made during early concept development may become embedded in requirements, inherited by downstream models, repeated in verification logic, or treated by adjacent teams as an established position rather than a provisional interpretation. As programs mature, the original rationale can become difficult to recover, even as **the assumption itself gains authority through repeated reuse.** Propagation therefore is relevant, not only because assumptions can be wrong, but because they can accumulate influence without being revalidated as system boundaries, organizational responsibilities, and mission objectives change. In distributed human spaceflight programs, this makes assumption propagation a central systems problem: it may begin as a local judgment but can soon become a program-level dependency long before anyone is formally accountable for its continued validity.

To prevent both assumption drift and assumption propagation, long duration programs require explicit assumption tracking.

---

[88] NASA, *Systems Engineering Handbook*, SP-2016-6105 Rev. 2, §5.3; §5.4; §6.4; §6.8.; NASA *Risk Management Handbook,* NASA/SP-2011-3422 (2011).

[89] Ali, Dalpiaz, Giorgini, and Souza, "Requirements Evolution: From Assumptions to Reality," in *Proceedings of the 16th International Conference on Exploring Modeling Methods for Systems Analysis and Design (EMMSAD 2011)*, *LNBIP* 81 (2011) pp.372–382.

3.3 The Importance of Explicit Assumption Tracking

Explicit assumption tracking is essential in long-duration, multi-partner exploration programs because assumptions shape architecture, verification scope, risk posture, and campaign sequencing[90]. Without formal mechanisms to document, monitor, and update assumptions, the resulting engineering decisions can become disconnected from the conditions that originally justified them. Four elements, visibility, ownership, updates, and traceability, form the backbone of a reliable assumption management process[91].

Visibility

Assumptions must be visible to everyone who depends on them[92]. Making assumptions explicit prevents them from becoming hidden drivers of design decisions or quietly shaping analyses without review. In complex campaigns like *Artemis*, visibility ensures that engineering teams, program management, policy offices, and partner organizations share a common understanding of the conditions under which decisions were made.

Ownership

Each assumption must have a clear owner responsible for its accuracy, relevance, and ongoing validity[93]. Ownership prevents assumptions from becoming "orphaned," where no individual or organization feels accountable for confirming or revising them. When assumptions have identifiable stewards, they can be managed with the same rigor as requirements and constraints.

Updates

Assumptions require regular reevaluation as new information becomes available—test results, partner performance, policy updates, industrial capability changes, or mission planning shifts[94]. Long-duration programs are particularly vulnerable to assumptions that persist unexamined through architecture revisions. Systematically updating assumptions reduces drift, prevents outdated beliefs from becoming embedded, and maintains alignment with the evolving technical and policy environment.

Traceability

Assumptions must be traceable to the decisions, analyses, and requirements they support. Traceability clarifies why an assumption was accepted, what evidence existed at the time, and what

---

[90] NASA, *Risk Management Handbook*, NASA/SP-2011-3422 (2011), §6.4; NASA Office of Inspector General, *NASA's Management of the Human Landing System Contracts*, IG-26-004 (Mar. 10, 2026), pp. 12–16, 29–35; Aerospace Safety Advisory Panel, *2025 Annual Report* (Jan. 2026), pp. 13, 15, 18–24.

[91] John A. Dewar, *Assumption-Based Planning: A Tool for Reducing Avoidable Surprises,* ABP Step 2, pp. 64–90, and ABP Step 3, pp. 91–107,

[92] Ronny Roeller, Patricia Lago, and Hans van Vliet, "Recovering Architectural Assumptions," *Journal of Systems and Software* 79, no. 4 (2006): 552–573.

[93] NASA *Systems Engineering Handbook* (NASA/SP-2016-6105 Rev. 2, 2016), §6.2, §6.4; NASA *Risk Management Handbook* (NASA/SP-2011-3422, 2011), §6.8.

[94] NASA Procedural Requirements 7120.5F — *NASA Space Flight Program and Project Management Requirements* (2021), §§2.2–2.4, App. G.

would be affected if it were modified or overturned[95]. This becomes crucial when re-baselining architectures, reassessing risk, or integrating new partners. In federated program structures, traceability ensures that one organization's assumption does not inadvertently constrain or misinform another's.

Propagation is therefore not a secondary concern; it is one of the defining hazards of complex architecture.

An assumption that begins as provisional can be inherited repeatedly until it is no longer recognized as contingent at all. It may be copied into requirements interpretations, embedded in analysis models, reflected in test planning, or accepted in governance discussions as though it were a demonstrated condition. By the time it reaches operations, it may function as if it were an established fact. This is one of the central dangers in distributed systems:

Assumptions can persist long after the initial evidence that supported them has faded in relevance. They can gain authority without gaining corresponding validation.

For these reasons, assumptions should be treated as lifecycle entities with real governance consequences[96]. They are introduced early, interpreted repeatedly, inherited unevenly, and often tested only indirectly. Some are retired through evidence. Others persist because no milestone forces their explicit reevaluation. Still, others become so normalized that they disappear into routine language, baseline products, or accepted decision logic. In long duration human spaceflight campaigns, the persistence is not benign. Assumptions can survive leadership change, program restructuring, contractor turnover, and architectural evolution while retaining influence over later decisions that are far removed from their original context.

The issue, then, is not whether complex systems can avoid assumptions. Simply put – they cannot. The issue is whether assumptions are treated with rigor proportionate to the work they are performing inside the architecture. This directly parallels criticality. Assumptions that carry architectural load—by shaping interfaces, verification scope, or mission-level dependencies—should be treated with rigor proportional to the consequences of their failure. If an assumption shapes interface logic, verification sufficiency, operational readiness, or cross-program dependency, it should not remain an informal premise buried in local reasoning. It should be visible, bounded, owned, and revisited when conditions change. Assumptions that cross program boundaries or affect mission level

[95] NASA Systems Engineering Handbook (NASA/SP-2016-6105 Rev. 2, 2016), §5.3; NASA Office of Inspector General, *NASA's Management of the Human Landing System Contracts*, IG-26-004 (Mar. 10, 2026), pp. i–ii, 1–4, 12–31.

[96] NASA Office of Inspector General, *NASA's Management of the Human Landing System Contracts*, IG-26-004 (Mar. 10, 2026), p. i, pp. 12–16; U.S. Government Accountability Office, *NASA Artemis Programs: Crewed Moon Landing Faces Multiple Challenges*, GAO-24-106256 (Nov. 30, 2023), pp. 13–25.

confidence should be elevated accordingly, because once they being functioning as structural elements of the campaign, they also become potential sources of structural failure[97].

4. Assumption Lifecycle Framework for Complex Systems

A direct visual representation of how policy created architectural assumptions travel through design, verification, and operations, can be seen in Figure 3[98]. It presents a full lifecycle view of the *Artemis* architecture, showing how assumptions enter, propagate, and harden across policy, system design, verification, and operations. It is divided into several major domains:

- Space Policies & Accords (SPD-1, SPD-6, *Artemis* Accords, NASA Authorization Act)[99]
- *Artemis* Architecture & Orion Integration
- Surface Infrastructure (habitats, rovers, reactors, ISRU)
- Gateway Aggregation & HLS Integration
- Safety -Critical subsystems
- Verification & Assurance Evidence
- Mission Operations
- Emergent Risk & Feedback Loop
- Interpretation Gates:
    - G1: Architecture Decision
    - G2: Safety Decision
    - G3: Operational Decision
- Assumption Stewardship & Lifecycle Tracking

Figure 3 illustrates the central argument of this paper: assumptions must be treated as system elements with direct safety relevance across the entire *Artemis* lifecycle. The figure makes this clear in three ways:

1. *Artemis* depends on interconnected technical, programmatic, and policy assumptions.

The diagram shows how upstream policy directives (SPD-1, SPD-6, *Artemis* Accords, NASA Authorization Act) feed directly into architecture decisions, system design, safety requirements, and verification boundaries[100]. This coupling demonstrates that assumptions do not originate within engineering alone—they are shaped by policy, governance, and mission-level decisions.

---

[97] Soufiane El Fassi et al., "Managing Assumption Driven Design Change via Margin Allocation and -TradeOffs-," *Journal of Engineering Design* (2024).
[98] NASA Office of Inspector General, *NASA's Management of the Human Landing System Contracts*, IG-26-004 (Mar. 10, 2026), pp. 30–33.

[99] Space Policy Directive-1 (2017); Space Policy Directive-6 (2020); *Artemis* Accords (2020); NASA *Artemis* Architecture Definition Document (v1.0).
[100] NASA Office of Inspector General, *NASA's Management of the Human Landing System Contracts*, IG-26-004 (Mar. 10, 2026), p. i, pp. 12–16; U.S. Government Accountability Office, *NASA Artemis Programs: Crewed Moon Landing Faces Multiple Challenges*, GAO-24-106256 (Nov. 30, 2023), pp. 13–25.

2. When assumptions evolve asynchronously, they propagate across design, verification, and operations.

The figure maps how assumptions flow through the three interpretation gates—G1 (architecture), G2 (safety), and G3 (operations)—and identifies where asynchronous evolution leads to ANOMALY / VIO ACTION / POLITICAL DRIFT[101]. These pathways show that mismatches across lifecycle phases can create hidden vulnerabilities as assumptions migrate between architecture, subsystem qualification, and mission operations. In this framework, context matters where Anomaly, VIO Action and Political Drift are concerned, and should be understood to be the following:

Anomaly: An anomaly is the operational signal that an assumption used in design or verification was incorrect, incomplete, or invalidated by real conditions. These can be understood to fall within the realms of:

- A technical behavior that violates expected system performance
- Something discovered during testing, integration, or flight
- Caused by an underlying assumption not matching reality

VIO ACTION (Verification / Validation & Independent Oversight Action): A formal process step where verification meaning is reevaluated because the underlying assumption has drifted, changed, or been proven invalid. These can be understood to fall within the realms of:

- A corrective action triggered when verification evidence no longer aligns with the assumption it was based on
- A point where safety or technical oversight bodies have to step in.
- A governance mechanism—not necessarily a failure, but a required realignment

Political Drift: Political drift occurs when upstream policy, strategic priorities, or governance decisions change, thereby altering the assumptions that originally shaped the system architecture. These can be understood to fall within the realms of:

- A shift in policy directives, budget allocations, or mission priorities that affects architectural intent
- Changes in international partnerships, cadence expectations, or exploration strategy
- Upstream decisions evolving faster than engineering can reinterpret them, resulting in outdated or misaligned assumptions.

They form a **trio of detection pathways** to indicate a defect, fracture or rupture within the assumption chain:

- **Anomaly** → detected in hardware/software or operations
- **VIO Action** → detected in verification or safety oversight
- **Political Drift** → detected in high-level policy/strategy

[101] NASA, *Space Flight Program and Project Management Requirements,* NPR 7120.5F, §§2.2–2.4.

As the diagram shows, assumption misalignment can manifest at any point in the chain, underscoring the need to understand not only individual assumptions but the broader context in which they are embedded[102].

Contextual Assumptions Embedded in the Lifecycle Framework

To fully interpret Figure 3, it is important to recognize the contextual assumptions that underpin its flow and structure. These assumptions shape how the lifecycle is depicted and influence how stakeholders interpret the movement of assumptions across the system:

- Assumed alignment in strategic direction: The model presumes consistent interpretation of SPD-1, SPD-6, and the *Artemis* Accords across all organizations.
- Assumed coherence in governance and decision gates: G1–G3 are shown as synchronized, clearly owned decision points, implying shared criteria and coordination across directorates.
- Assumed stability of subsystem boundaries: Safety-critical subsystems are depicted as discrete and predictable, reflecting an assumption that their interfaces and integration assumptions remain stable.
- Assumed linearity in verification evidence flow: The diagram presents verification as sequential and timely, even though real verification is iterative and asynchronous.
- Assumed ability to detect assumption drift: The "Assumption Lifecycle & Knowledge Validity" layer implies teams have resources and mechanisms to monitor drift continuously.
- Assumed organizational learning: The feedback loop indicates a reliable flow from anomaly → analysis → updates, which is aspirational in practice.
- Assumed predictable mission operations: Mission phases are shown as stable sequences, implying operational conditions do not invalidate earlier architectural assumptions.
- Assumed consistency in meaning across organizations: The Interpretation Gate (CREATED → INFERRED → OBSERVED → VERIFIED) presumes common epistemic standards across centers and partners.

Recognizing these contextual assumptions clarifies how the lifecycle structure itself shapes (and sometimes constrains) the interpretation and management of assumptions.

3. Effective assumption management is essential to safety and mission success.

With these contextual factors made explicit, the rationale for active assumption management becomes clearer. In the bottom-right of the figure, "Assumption Stewardship" and "Tracking Verification Meaning Across Lifecycle" illustrate that maintaining assumption validity is a formal part of safety governance, not an informal engineering task. The diagram reinforces that risk cannot be controlled through verification artifacts alone. Safety ultimately depends on whether the assumptions behind verification, design, and policy remain valid as the *Artemis* campaign evolves.

The G1/G2/G3 structure used in this paper is an author-created analytic model, but it maps onto NASA governance and lifecycle concepts rather than replacing them. NPD 7120.4E states that NASA

[102] NASA *Systems Engineering Handbook*, SP-2016-6105 Rev. 2, §6.8.

programs and projects are managed through phased lifecycles with key decision points, where readiness to proceed is supported by lifecycle reviews and documented principal governing products (NASA, 2017/2026, NPD 7120.4E, §1(2)(a)). NPR 7123.1D likewise organizes systems engineering around common technical processes, lifecycle reviews, and review entrance/success criteria (NASA, 2023/2028, NPR 7123.1D, Ch. 3, 5, App. G)[103]. The gates therefore should be read as a synthesis framework for assumption movement across architecture, verification, and operations, not as official NASA gate names.

[103] NASA *Systems Engineering Handbook* (NASA/SP-2016-6105 Rev. 2, 2016); NASA *Procedural Requirements 7120.5F — NASA Space Flight Program and Project Management Requirements* (2021), §§2.2–2.4; App. G; NASA P*rocedural Requirements 7123.1D — NASA Systems Engineering Processes and Requirements* (2022), Ch. 2.

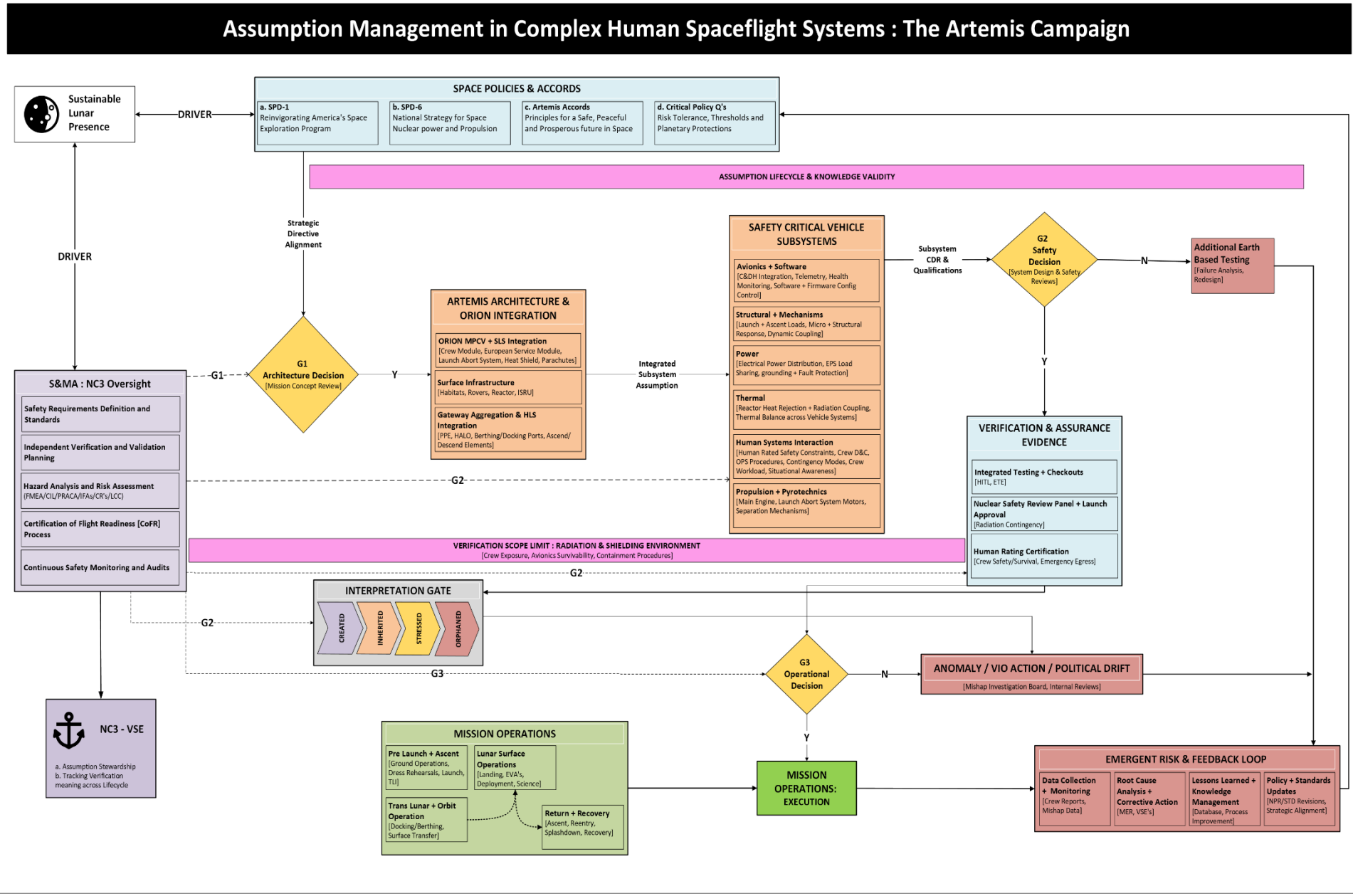


Figure 3 : Assumption Lifecycle Framework for Complex Human Spaceflight Systems

4.1 Assumption Management in Complex Human Spaceflight Systems

Assumption management in the *Artemis* campaign requires treating assumptions as dynamic lifecycle elements that evolve as the architecture matures, verification evidence accumulates, and operational context changes. In a distributed, multiprogram environment, assumptions rarely remain confined to the point where they were first introduced; instead, they propagate through architecture decisions, partner interfaces, certification strategies, and mission operations[104]-. This turns assumption stewardship into a system-level responsibility rather than a task isolated to any single organization or engineering discipline. Effective lifecycle management therefore requires identifying the assumptions embedded at each decision gate, tracking their justification and supporting evidence, and reassessing their validity as the campaign transitions from conceptual design to verification planning and ultimately to crewed operations.

Within *Artemis*, this problem is amplified, which makes this discipline essential for preventing outdated or unverified assumptions from silently hardening into design logic, verification meaning, or operational practice as the campaign evolves. It is why assumption stewardship must be tied to decision management across the lifecycle. At Gate 1, assumptions shape architecture and concept closure; Gate 2 provides insight into what influences count as adequate verification and what evidence is considered sufficient to support safety claims; and at Gate 3 they reappear in operational products, anomaly interpretation, and mission execution. Across all three, technical management, risk management, verification planning, and information continuity must work together to prevent outdated assumptions from hardening into accepted truth.

When approached from an INCOSE perspective, the challenge is not limited to technical correctness, but lifecycle coherence. Assumptions must remain linked to their origin, justification, configuration context, and revision triggers as the campaign matures. Without that continuity, design logic, verification meaning, and operational practice can drift apart while still appearing administratively complete. Assumption management is then not only a supporting activity around the system, but a part of how the system is governed across phases.

The assumption vortex illustrated in Figure 4 shows that assumptions in complex exploration campaigns do not remain fixed at the point where they are first introduced. At Gate 1, they help close the architecture and define what appears feasible. At Gate 2, they influence what counts as sufficient evidence and how safety claims are interpreted. At Gate 3, they reappear in mission execution, nominal interpretation, and operational products. If they are not actively revalidated, they can harden into accepted truths[105].

The vortex also conveys a second point central to this paper: **hidden assumptions do not persist passively.** They feed back into future cycles by shaping the background beliefs that later teams

[104] NASA Office of Inspector General, *NASA's Management of the Human Landing System Contracts* (IG-26-004, 2026), p. i, pp. 12–16; U.S. Government Accountability Office, *Crewed Moon Landing Faces Multiple Challenges* (GAO-24-106256, 2023), pp. 13–34.
[105] *Columbia* Accident Investigation Board Report, Volume I (2003), executive summary p. 9, pp. 177–203; Report of the Presidential Commission on the Space *Shuttle Challenger* Accident, Volume I (1986), Ch. 7, pp. 177–203.

inherit as already reasonable, already proven, or no longer in question. This feedback loop means that:

Assumptions accumulate force not only from their original justification, but also from the institutional memory, documentation, and practices that make them appear stable simply because they have persisted over time.

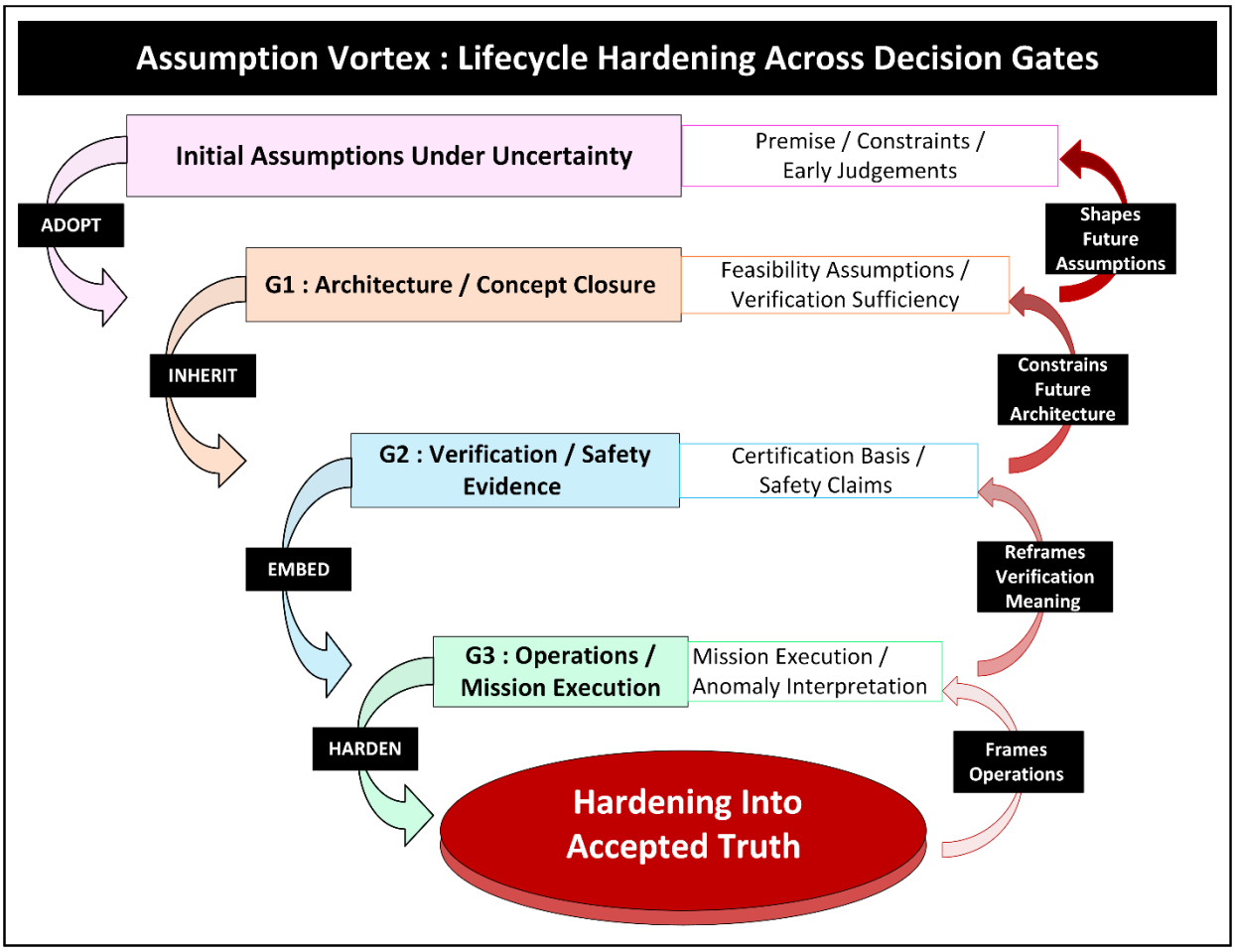


Figure 4 : Assumption Vortex - Lifecycle Hardening Across Decision Gates

The Assumption Vortex shows how early judgments can solidify into architectural expectations as they pass through successive decision gates. Once these expectations become embedded, they begin shaping decisions in ways that may no longer reflect the uncertainty that produced them. This dynamic is not abstract; it appears in concrete areas of the *Artemis* campaign where early assumptions became structural anchors before the supporting evidence fully matured.

4.1.1 Commercial HLS Maturity Assumption

One of the most consequential assumptions in *Artemis* has been the maturity of commercial HLS capability relative to the campaign architecture being built around it[106]. This is not a question of whether either of the two providers can eventually produce a functioning lander. The deeper issue is whether integration, maturity, interface maturity, verification planning, and campaign dependency are evolving at the same rate as architectural reliance on the system.

In this campaign architecture, a landing system does not mature in isolation. Its readiness must be understood in relation to the interfaces and operational conditions that give the mission meaning: launch, integration, rendezvous, communications, mission sequencing, abort logic, crew timeline, surface objectives, and the broader cadence of the campaign operations. If these surrounding dependencies begin to stabilize around the assumed availability or maturity of HLS **before** the relevant evidence base is equally mature, then the program begins to inherit architectural commitments that may be difficult to unwind later.

This creates a classic assumption management problem where early confidence in commercial maturity can silently migrate into verification, planning, schedule, logic and operational expectation. The result is not necessarily immediate failure, but growing tension between the "architecture as planned" and the "architecture as actually supportable". Verification then risks being organized around the needs of the campaign rather than around the actual maturity state of the integrated system. In that environment, the key question is no longer whether HLS can be verified, but **whether the campaign has defined the right thing to verify at the right level of architectural reality.**

This implies that commercial HLS maturity should be handled as a managed campaign-level assumption rather than a static program status attribute. Accordingly, its integration maturity, interface maturity, and verification basis must remain continuously visible, periodically reassessed, and adaptable as the architecture evolves. Otherwise, the campaign risks anchoring downstream commitments to assumptions whose interpreted meaning evolves faster than the evidence required to sustain them.

4.2 Architectural Assumptions – Decision Gate 1

Gate 1 is the point at which broad mission possibility begins to narrow into credible architectural decision. NASA describes Pre-Phase A as the stage in which teams examine a broad spectrum of concepts within technical cost and schedule constraints, and Phase A as the stage in which a proposed mission or system architecture is developed into a credible baseline responsive to expectations, requirements, constraints, and available resources. In practical terms, this is where the program stops asking what could be built and starts deciding *what kind of system it will actually become*.

[106] NASA Office of Inspector General, *NASA's Management of the Human Landing System Contracts* (IG-26-004, 2026), pp. 30–33.

At this gate, assumptions are not yet claims of demonstrated performance, they are architectural premise about feasibility, interface dependencies, and intended use. NASA's Technical Requirements guidance makes this explicit: baselined stakeholder expectations include needs, goals, objectives, assumptions, constraints, and external interfaces, while the requirements process establishes the design boundary by defining constraints, identifying elements already under design control, and identifying external and enabling systems with which the system must interact. Gate 1 assumptions therefore do more than fill gaps in knowledge, they help define what is fixed, what remains open to trade, and what other systems the architecture will presume to exist.

FSP is a useful example as it shows how an early enabling concept can become an architectural premise. NASA states that its FSP effort is intended to provide abundant, continuous power independent of environmental conditions, that the agency plans to use such a system on the Moon first, and that NASA is working with DOE and industry on a 40 kilowatt class lunar system for the early 2030s[107]. NASA also ties that effort to Kilopower heritage - the Kilopower Reactor Using Stirling TechnologY (KRUSTY) demonstration was presented as evidence that a small reactor system could create electricity with fission power and remain stable and safe under nominal and off nominal conditions[108]. That does not mean the later campaign has already proven operational sufficiency, all it shows is that an enabling concept has acquired enough plausibility to begin shaping the architecture.

That shaping effect is the real significance of Gate 1 - NASA's requirements guidance says rationale should document assumptions and design constraints, and its validation guidance says requirements should be checked to ensure they are well formed, using valid assumptions and are consistent with the life cycle success criteria[109]. Once an architectural choice is accepted into the baseline, the cascading effect then requires that data requirements, interfaces, and planning products begin inheriting it.

The consequence is a quiet reduction in the trade space: alternatives that were once peers becomes harder to revisit because the selected concept is now embedded into the design boundary itself.

This is why Gate 1 assumptions deserve special treatment in assumption management. They are upstream, connective and durable. They influence which technologies are treated as essential, which interfaces are planned around, and which operational concepts later gates will be asked to verify and execute. When a capability such as FSP is accepted as part of the campaign's expected future, it begins to structure logistics, mobility, habitat support, communications, and surface operations long before the campaign possesses complete evidence that the capability will arrive on

[107] National Security and Technology Memorandum-3 (2026), §§2–3; NASA, Fission Surface Power (n.d.).
[108] Poston et al. (2020), KRUSTY reactor design sections; Gibson et al. (2018), KRUSTY demonstration sections; NASA, Fission Surface Power (n.d.).
[109] NASA *Systems Engineering Handbook* (NASA/SP-2016-6105 Rev. 2, 2016); NASA *Procedural Requirements 7123.1D — NASA Systems Engineering Processes and Requirements* (2022), Ch. 2 and app. C.

the needed timeline and in the needed form. Gate 1 is where assumption management first becomes a structural issue rather than local engineering judgment.

4.3 Verification Interpretation and Safety Assumptions – Decision Gate 2

Gate 2 is the point at which architectural intent seeks evidentiary closure. Within NASA's Systems Engineering framework, product verification demonstrates compliance with the approved requirement set, while product validation demonstrates that the product fulfills its intended purpose in its intended environment[110]. NASA is explicit that these are different questions: verification asks whether the product meets each requirement statement, while validation asks whether the resulting system actually satisfies stakeholder expectations and use. This distinction is indicative that a program can accumulate a large volume of formally valid evidence without resolving whether that evidence still supports the mission as it is currently configured.

The central issue at Gate 2, therefore, is not limited to evidentiary existence but to its permitted interpretation. NASA's verification guidance emphasizes that procedures are developed against specified requirements, considering enabling products, external interfaces, facilities, personnel, and operational scenarios. NASA's validation guidance likewise stresses that validation must occur against stakeholder expectations, Concept of Operations (ConOps), operators, and the intended operational environment[111]. Read together, these sources show that evidence is never context-free. It is generated against a particular requirement set, interface set, mission profile, and use case. If any of these shift, the evidentiary claim may remain administratively complete while becoming substantively thinner.

Figure 5 illustrates this problem directly. A requirement or safety claim is first defined against an assumed mission profile and operating context. Tests, analyses, inspections, and demonstrations then generate evidence that closes the requirement in the original context. But should the mission or architectural context later change, the meaning of the evidence shifts even if the documentation remains intact. Earlier evidence does not necessarily become false, but its interpretive burden increases: the program must judge whether previously accepted closure still applies to the mission now being prepared. This is the characteristic Gate 2 problem in complex programs—evidence may remain complete on paper while becoming thinner in context.

The KRUSTY demonstration is a useful illustration of the difference between successful evidence generation and full operational meaning. NASA reported that the Kilopower reactor, using Stirling technology tests, successfully demonstrated a Fission Power System that could generate electricity and remain stable under both nominal and off-nominal conditions. While an important technical achievement, it did not by itself certify that a future lunar campaign had answered the broader question of operational sufficiency for FSP within a specific architecture. That broader assessment affects logistics chains, regulatory regimes, and crewed mission design. The Gate 2 issue is therefore

[110] NASA *Systems Engineering Handbook* (NASA/SP-2016-6105 Rev. 2, 2016), §§5.3, 5.4, 6.2.
[111] NASA Systems Engineering Handbook (NASA/SP-2016-6105 Rev. 2, 2016), pp. 115–123.

interpretive as much as technical: how far should a successful demonstration be allowed to travel into safety justification, confidence assessment, mission planning, and architectural commitment?

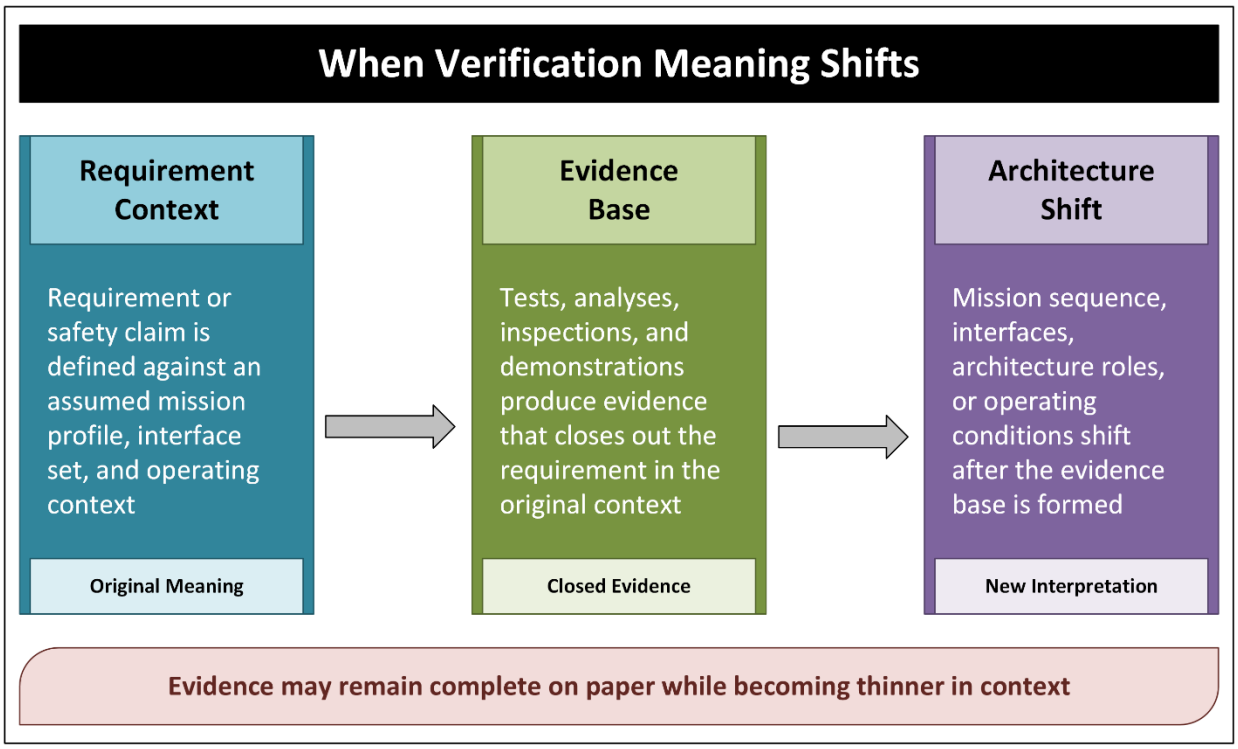


Figure 5 : Verification Context Shift

NASA's technical standards procedures strengthen the distinction between requirement compliance and assumption stewardship. NPR 7120.10B addresses the selection, tailoring, and use of technical standards for NASA programs and projects (NASA, 2022/2027, NPR 7120.10B, Ch. 3). Tailoring is not merely administrative flexibility; it is a point where assumptions about applicability, risk, maturity, and evidence sufficiency can enter the technical baseline and should therefore remain traceable.

NASA's guidance on Phase D[112] and product realization reinforces this interpretation. Verification and validation occur as part of an integrated realization process, and later end-to-end integration activities may still reveal that a system's meaning changes once placed within its external interfaces or in more realistic operational scenarios. This is why Gate 2 should not be treated as a onetime closure event. It is the point at which the program must judge evidence sufficiency, context of use, certification implications, confidence, and the need for revalidation if assumptions have shifted. Gate 2 is where verification becomes interpretation: the program is not simply asking whether it has evidence, but whether it has the *right* evidence for the mission it now intends to fly.

[112] NASA *Systems Engineering Handbook* (NASA/SP-2016-6105 Rev. 2, 2016), §5.0, §3.7 Project Phase D (System Assembly, Integration and Test, Launch).

4.4 Operational and Contextual Assumptions – Decision Gate 3

Gate 3 is where verified products, procedures, and mission rules transition into actual operations, and where assumptions previously judged acceptable are tested against real execution[113]. NASA defines Phase E as the period in which the prime mission is conducted and sustainment is maintained, but its systems engineering guidance is clear: systems engineering does not end at launch. During operations, product-realization activities continue through mission plan updates, responses to changing flight conditions and in-flight anomalies, and revisions of techniques, procedures, and guidelines based on operational experience. Gate 3, therefore, is the start of system use and also the point at which the program must determine whether the claims, constraints, and assumptions inherited from earlier gates still hold under live mission conditions.

Operational context is central to this determination. NASA notes that successful Phase E execution requires established operations capability across four areas: tools, processes, products, and trained personnel. It also states that stakeholder expectations for risk management, planning flexibility, timing of updates, and communication must be reflected in the operations plan, operating constraints, and flight rules. Operational sufficiency is not defined by hardware readiness alone, it depends on whether the mission environment still matches the context in which procedures, timelines, and decision rules were assessed. A strategy that seemed reasonable during development can become fragile when time is short, staffing is limited, environmental effects are sharper than expected, or anomaly response requires faster interpretation than the system can support.

At Gate 3, verification meaning can shift ***again***—now for operational rather than developmental reasons. NASA explains that Phase E verification often consists of unit tests of tools, datasets, and procedures under simulated conditions, while validation occurs in integrated operational scenarios such as simulations, readiness tests, or end-to-end exercises[114]. It also stresses that introducing new operational capabilities requires reviewing prior verification and validation evidence and assessing whether the mission operations system is ready to accept the change. In practice, a capability may remain technically verified, yet its operational meaning can change once inserted into a live mission with active constraints, changing conditions, and limited recovery margins.

NASA's Phase E guidance further shows why assumptions may need to be reopened during operations. It notes that new requirements or changes to existing ones may emerge from flight experience, anomalies, or shifts in mission goals such as extensions. It also warns that change review is harder in Phase E because fewer resources remain for comprehensive assessment. Early and close involvement of safety and mission assurance is therefore critical to determining whether proposed changes remain within allowable risk tolerance. Gate 3 is thus fundamentally interpretive: the issue is not whether the system can continue operating, but if the program still understands the risk and meaning of continued operation under altered conditions.

---

[113] Ramirez, Cheng, Bencomo, and Sawyer, "RELAXing Claims: Coping with Uncertainty While Evaluating Assumptions at Run Time," in *Proceedings of the 15th International Conference on Model Driven Engineering Languages and Systems*, *LNCS* 7590 (2012): 53–69.

[114] NASA Systems Engineering Handbook (NASA/SP-2016-6105 Rev. 2, 2016), §2.4, §3.0, §5.3–5.4, §6.2, §6.4, §6.5, §6.8.

The analytical point is that Gate 3 is where operational continuity can be mistaken for evidentiary sufficiency. A mission may continue running on verified products and improved procedures, but continuity alone does not prove that the original assumptions remain fully valid. Instead, Gate 3 requires disciplined judgment about whether anomalies, context shifts, resource constraints, or mission changes have altered the meaning of earlier closure.

If Gate 1 defines the architecture, and Gate 2 determines what counts as acceptable proof, then Gate 3 determines whether that proof still means what the program thinks it means once the mission is actually being flown.

4.5 Propagation of Assumptions Across Lifecycle Gates

Assumptions propagate across lifecycle gates because the output of one gate becomes the working premise of the next. NASA's systems engineering guidance begins the requirements process with baseline stakeholder expectations that explicitly include assumptions, constraints, and external interfaces[115]. Those expectations are carried forward through requirements verification and validation, transition, and operations. Assumptions are not left behind when a project advances from architecture definition to evidence generation to execution. They travel with the program, initially as concept-shaping judgments, then as requirements premises, and finally as operational expectations about what the system can safely do and under what conditions.

The first propagation mechanism is **transfer** and **inheritance**[116]. Once an assumption is absorbed into stakeholder expectations, ConOps, requirements, or interface definitions, later teams encounter it less as an open hypothesis and more as accepted project structure. NASA's requirements guidance is explicit that reviews should confirm assumptions were formed using valid rationale and are consistent with lifecycle phase success criteria. When requirements containing those assumptions are validated and placed under configuration control, they begin to function as inherited facts. A judgment that originates as a feasibility premise at Gate 1 can reappear at Gate 2 as a verification boundary and at Gate 3 as an operational dependency—without being revisited in proportion to its increasing downstream influence.

The second mechanism is **coupling**. NASA's Phase E systems engineering guidance notes that integration and operations often involve combining multiple existing and modified products into an updated mission operations capability, and that methods must exist to identify and assess the impacts of changes across the system[117]. It emphasizes that integrated scenarios, readiness tests, and end-to-end tests provide the operational context necessary to determine whether a product truly meets mission needs. This illustrates that assumptions do not move linearly through the lifecycle; they become coupled across requirements, integration steps, tools, procedures, operators, and external systems. A change in one domain can therefore alter readiness in one sufficiency or another.

---

[115] NASA Procedural Requirements *7123.1D — NASA Systems Engineering Processes and Requirements* (2022), Ch. 2 & app. C.

[116] Roeller, Lago, & van Vliet, "Recovering Architectural Assumptions," *Journal of Systems and Software* 79, no. 4 (2006): pp.552–573.

[117] NASA Systems Engineering Handbook (NASA/SP-2016-6105 Rev. 2, 2016), §3.0, §6.2, §6.5, §6.8.

The third mechanism is **feedback** and **reinterpretation**. NASA states that significant in-flight anomalies or discoveries during Phase E may require reassessment of stakeholder expectations, and that major hardware or software changes during operations may necessitate reapplying earlier systems engineering processes—effectively treating the modification as a parallel mini-project[118]. In other words, feedback does not sit outside the lifecycle; it acts upon it. Operational experience can force the program to reopen requirements, reinterpret previously verified evidence, or question whether a once-reasonable architectural premise still applies to the mission being flown.

Taken together, **transfer, inheritance, coupling**, and **feedback** explain why assumptions can become both durable and difficult to see[119]. Gate 1 narrows the architecture and reduces the trade space. Gate 2 determines what counts as sufficient evidence to support that architecture. Gate 3 tests whether those claims are still held under actual mission conditions. If these transitions are not managed explicitly, an assumption can remain formally intact while its practical meaning weakens from gate to gate. This is the central lifecycle issue carried into the next section: complex human-spaceflight systems do not simply contain assumptions—they move them, embed them, and sometimes preserve them long after the context that made them *reasonable* has changed.

5. Structural Drivers of Assumption Propagation

Section 4 argued that assumptions move across the lifecycle through transfer, inheritance, coupling, and feedback. Section 5 then asks why that movement becomes especially difficult to manage in a campaign architecture such as *Artemis*. NASA's current Moon-to-Mars framing explicitly describes the architecture as evolvable, segmented, and composed of multiple sub-architectures and elements that build on one another over time[120]. NASA also notes that some elements predate the current architecture effort, while others emerge from capability gaps and are mapped across multiple segments and sub-architectures. This structure is analytically important because assumptions are not limited to propagation through design documents, they also propagate through a campaign whose components are distributed across time, organizations, and capability layers.

The structural issue is about the presence of assumptions, but also the environment in which they must remain valid. In *Artemis*, that environment is shaped by compressed mission cadence, distributed verification, dispersed proof ownership, dependence on enabling infrastructure that may arrive unevenly, and continuing shifts in resources and governance. These are not background management conditions; they are channels through which assumptions can gain momentum, lose traceability, and outlive the context that originally justified them. The subsections that follow examine these drivers in turn.

[118] NASA *Systems Engineering Handbook* (NASA/SP-2016-6105 Rev. 2, 2016, §3.0), §6.5, §6.8.
[119] *Columbia* Accident Investigation Board Report, Volume I (2003).
[120] NASA *Moon to Mars Architecture Definition Document* (NASA, n.d.).

5.1 Campaign Cadence and Learning Cycles

The completion of *Artemis* II gives the cadence argument a formal operational basis. With the world watching, *Artemis* II launched on April 1st, 2026, and returned on April 10th, 2026[121]. Through and through it was a success and executed its mission from launch to splash down beautifully. The mission lasted 9 days, 1 hour, and 32 minutes, and set a new record for crewed distance from Earth during a slingshot fly-back[122].

With *Artemis* II complete, the campaign is no longer reasoning solely from uncrewed flight heritage or planned milestones; it is now incorporating lessons from a full crewed test mission. NASA's post-flight framing reinforces this shift. The agency has pivoted its focus to *Artemis* III, which will test integrated operations with commercially built Moon landers in LEO. This reflects a move away from a simple "demonstration-to-landing" sequence and toward a stepwise campaign in which cadence and operational learning increasingly shape what each mission must demonstrate before the program commits to the next level of risk.

NASA's post-flight assessment also framed *Artemis* II not as an endpoint, but as groundwork for *Artemis* III in 2027 and lunar surface missions beginning in 2028. Around the same time, NASA announced an architectural update adding a new *Artemis* III LEO demonstration and targeting roughly one lunar mission per year thereafter[123]. Together, these developments provide a firmer operational basis for cadence analysis than what was available when *Artemis* existed only as a planned sequence on paper.

The key analytical issue is that learning cycles do not compress at the same rate as schedule ambition[124]. NASA noted that immediately after splashdown, engineers began detailed data analysis on Orion, SLS, and ground systems, including investigation of a urine-vent-line issue and corrective-action planning for *Artemis* III. At the same time, NASA publicly tied *Artemis* II to preparations for *Artemis* III, early surface missions, and the longer-term lunar base. The architecture update similarly emphasized maintaining focus on *Artemis* II lessons while examining capabilities needed to support increased cadence. In a campaign environment, this often means that follow-on architectural commitments harden before the full interpretive value of prior mission evidence has been absorbed.

Cadence therefore shapes how assumptions propagate by narrowing the interval in which lessons can be translated into revised judgments. The risk is not speed alone, but the deeper issue of anomaly data, interface findings, verification caveats, and operational observations that may still be under evaluation while the next mission's design logic is already taking form. When this occurs, the campaign begins building on partially digested experience, allowing assumptions to carry forward as if they were more firmly settled than they actually are. *Artemis* cadence thus becomes a structural

[121] NASA, *Artemis* II Mission Page (2026a).
[122] NASA, "NASA Welcomes Record-Setting *Artemis* II Moonfarers Back to Earth" (2026b).
[123] NASA, "NASA Adds Mission to *Artemis* Lunar Program, Updates Architecture" (2026c).
[124] NASA *Systems Engineering Handbook* (NASA/SP-2016-6105 Rev. 2, 2016), §§6.7–6.8; NASA *Artemis* II & III mission documentation (NASA, 2026a, 2026c).

driver because it dictates the pace at which evidence must be interpreted before new architectural and operational commitments are made.

5.2 Distributed Verification and Program Responsibilities

Assumption management in turn, becomes more complicated in *Artemis* because proof is not generated within a single tightly bounded program. Instead, evidence is produced across NASA Headquarters, integration offices at NASA centers, prime contractors, commercial providers, and international partners—each operating within its own scope, deliverables, and review structures[125]. In that environment, verification may become locally persuasive while campaign meaning remains globally incomplete. An element can show that it has met its own requirements or delivered its assigned artifact, yet the broader exploration case still depends on how that evidence is interpreted in relation to timing, interfaces, mission-sequence logic, and the performance of other systems across the campaign. The question then is whether the resulting proof from the verification that occurred actually closes the campaign-level argument decision makers believe it closes. This is the core idea illustrated by Figure 6, the "Distributed Proof Burden Map."

The diagram depicts stakeholders in a vertical column on the left—NASA Headquarters, NASA centers, prime contractors, commercial providers, and international partners. To the right, it shows corresponding artifact layers: campaign assumptions, integrated requirements, element evidence, provider proof, and partner deliverables. Orange circular nodes represent proof artifacts at each layer, and connecting lines indicate how evidentiary meaning is handed from one organizational layer to the next. The figure conveys that the burden of proof is not concentrated in a single location; it is distributed across organizational levels, each of which produces only part of the total evidentiary picture. The risk highlighted at the bottom states that proof is generated in pieces, but campaign-level closure must ultimately be interpreted somewhere.

This diffusion is central to the assumption problem because assumptions can persist—and sometimes deepen—within these handoffs. A NASA center may validate an integration product against one set of boundary conditions, a contractor may verify hardware against another, and a provider may deliver proof against a narrower service definition, all without any single authority fully owning the combined interpretive burden. In such a system, assumptions do not disappear when evidence is produced; they are displaced into the seams between organizations. Closure at one layer may be accepted as sufficient at the next, even when surrounding campaign conditions have shifted. What appears to be verification completion may therefore be partial closure preserved by organizational fragmentation rather than by strong cross-campaign understanding.

[125] NASA *Systems Engineering Handbook* (NASA/SP-2016-6105 Rev. 2, 2016), §§5.3–5.4; NASA Procedural Requirements *7120.5F — NASA Space Flight Program and Project Management Requirements*, §§2.4–2.5.

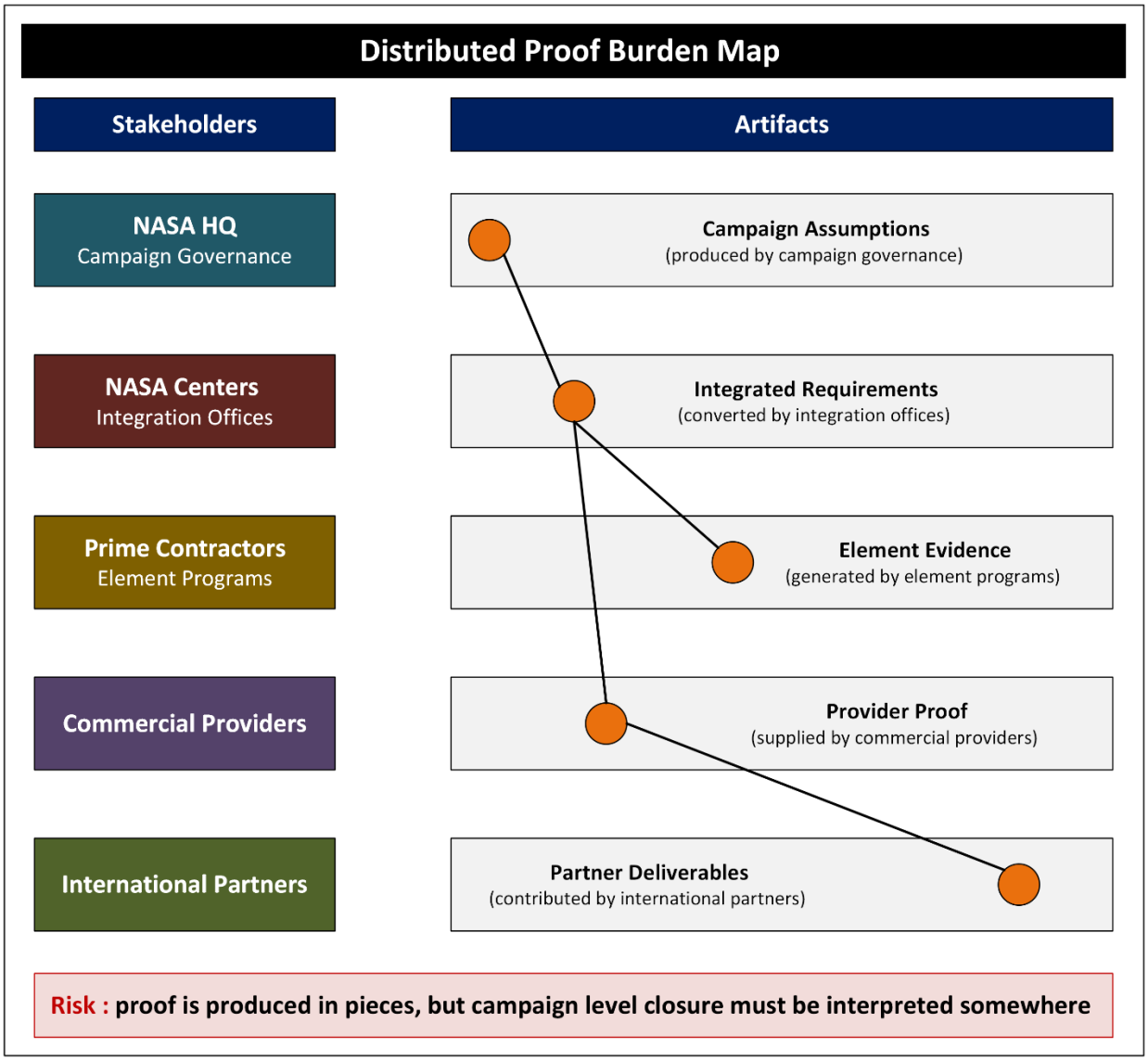


Figure 6 : Distributed Proof Burden Across the Artemis Campaign

For assumption management, the implication is that distributed verification requires distributed accountability—and an integrating authority capable of reopening inherited proof when context changes. If no one is explicitly responsible for checking whether local proof still supports the campaign-level claim, assumptions can harden under the appearance of normal and formally completed work. Figure 6 underscores that the issue is not a lack of evidence, but the challenge of interpreting distributed evidence coherently across the campaign architecture. In *Artemis*, the proof burden is dispersed across many contributors, but campaign-level confidence must still be assembled somewhere—and that assembly point is precisely where assumption visibility becomes critical.

5.3 Infrastructure Maturity and System Integration

A major driver of assumption propagation in exploration campaigns is treating enabling infrastructure as if future availability were already operationally reliable. In autonomous systems, infrastructure is not peripheral; it conditions how other systems are planned, interfaces are managed, sequences are verified, and operations are ultimately trusted. Capabilities like communications, logistics and staging, and surface habitability/mobility are enabling conditions—not isolated "support features"—that shape what the campaign believes it will be able to do. Once such conditions are embedded in architectures, timelines, and interface definitions, they begin influencing downstream decisions before the infrastructure reaches the maturity needed to justify that confidence.

This is why it is essential to distinguish infrastructure maturity from campaign maturity. A technology can show promising performance in development or demonstration and still be programmatically premature as a campaign dependency. KRUSTY's 2017–2018 ground test at the Nevada National Security Site demonstrated a compact fission system's stability and off-nominal tolerance under terrestrial test conditions—an important technical milestone[126]. But a successful demo is not the same as an operational lunar surface power capability integrated into architectures, logistics chains, mission timelines, maintenance concepts, and crew survival planning.

The campaign-level capability NASA seeks under FSP includes explicit requirements such as ≤6 metric-ton system mass, approximately 40 kWe of continuous electrical output, and design goals of autonomous operation for roughly 10 years to bridge lunar nights and shadowed regions[127]. These conditions that must be engineered, baselined, and sustained before downstream plans can responsibly treat the power system as available, reliable, and schedulable. Figure 7, the "Infrastructure Dependency Cascade" should be read precisely this way.

The central assumption branches into enabling infrastructure domains (blue boxes)—surface power, communications, logistics, staging, habitability/mobility—that other parts of the campaign depend on. From there, each domain drives downstream effects (grey boxes)—science operations, mission timelines, verification scope, crew operations. In other words, the blue layer represents the infrastructure capability layer, while the grey layer captures dependent mission consequences. Once infrastructure assumptions are embedded, they do not remain local; they propagate and start shaping multiple unrelated decisions at once.

[126] NASA Kilopower / KRUSTY Demonstration Materials (NASA, 2018).
[127] NASA Fission Surface Power project materials (NASA, 2024/2025).

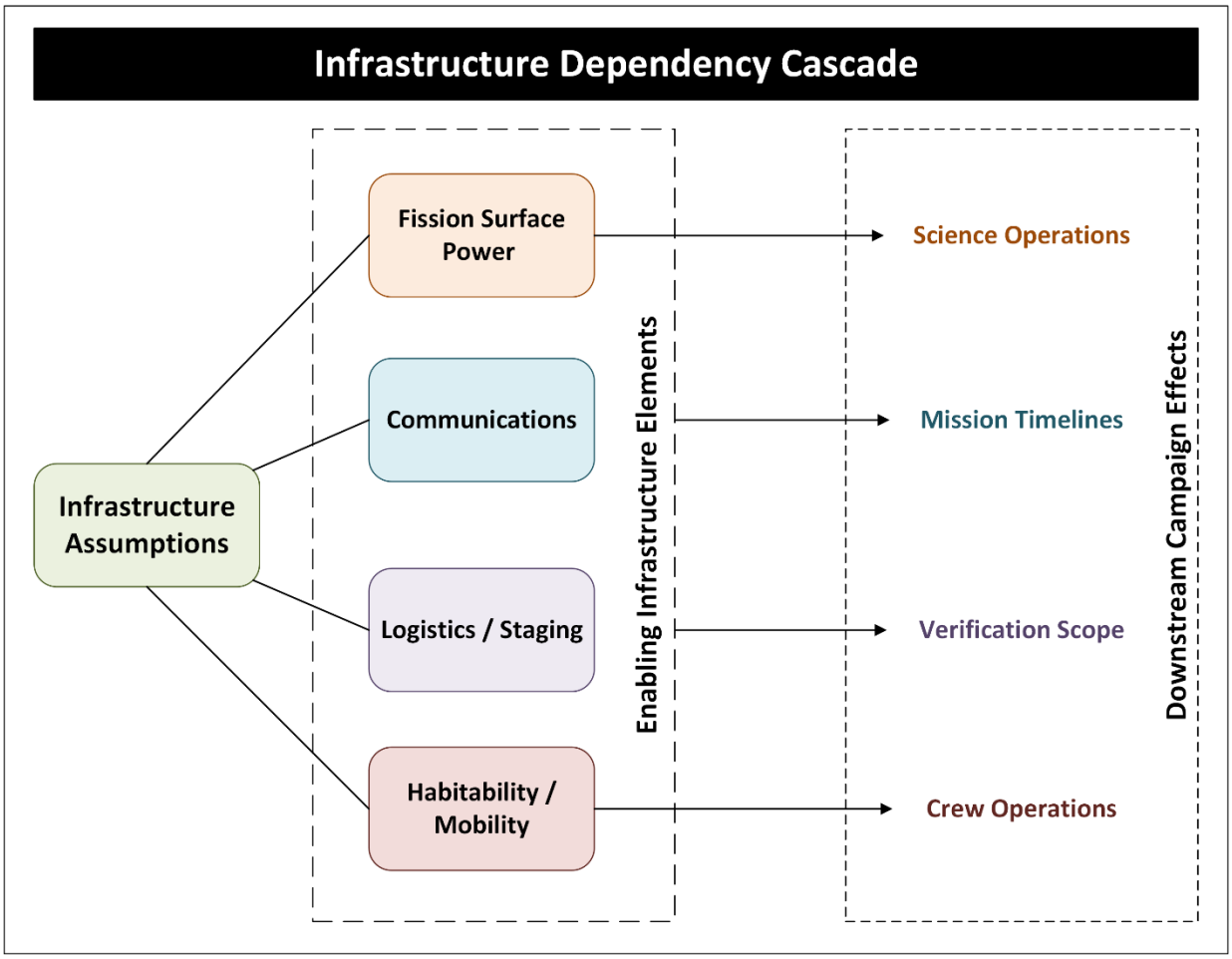


Figure 7 : Infrastructure as a Hidden Driver of Campaign Assumptions

NASA's Moon Base Users Guide reinforces the argument that infrastructure maturity is not a peripheral concern[128]. The guide describes lunar-base development as a phased effort: early systems may be designed for self-sufficiency, but sustained presence requires scalable, shared infrastructure for power, logistics, communications, and navigation. This is precisely the dependency pattern captured in the Infrastructure Dependency Cascade. Once shared infrastructure is assumed, downstream functions begin organizing around its expected availability. With that in mind, the infrastructure supports more than the architecture after it is built, but rath er, it works to define what the architecture believes it can safely and credibly become.

The guide also identifies architecture-driven technology gaps and data gaps as demand signals to industry, academia, and international partners. This confirms that lunar planning is being organized around capabilities that are still maturing rather than capabilities that are fully closed. The campaign can responsibly use these gaps to guide investment and partnership, but assumption management must keep the distinction clear: a named gap, a directed development effort, or a phased

[128] NASA Moon Base Users Guide (NASA, 2026e).

implementation plan is not evidence that the needed infrastructure will be available, integrated, and operational when downstream systems begin depending on it.

For that reason, the Moon Base Users Guide strengthens the claim that infrastructure functions as an assumption carrier[129]. Power, logistics, communications, navigation, mobility, and habitation become more than discrete system categories; they become shared conditions that influence design choices, verification scope, operational planning, mission timelines, and contingency concepts. The key assumption-management challenge is not whether the campaign has identified needed infrastructure, but whether it can prevent those future capabilities from being treated as present, stable, and dependable before the evidence supports that conclusion.

The guide is therefore useful precisely because it does not present sustained lunar presence as a closed capability. It presents it as a phased architecture dependent on the closure of functional technology and data gaps across power, logistics, communications, mobility, habitation, landing, and surface operations.

For that reason, integration must be treated as a lifecycle process, not a late assembly task. INCOSE's Systems Engineering Handbook emphasizes that integration is the sustained management of interfaces, dependencies, and emergent behavior across the life cycle—agreement processes, technical management (risk, configuration), and technical processes (integration, verification, transition, operation, maintenance)—not the physical "bring-together" of subsystems near launch[130]. In NASA practice, NPR 7120.5 requires baselined plans and configuration control of interdependent products by phase; when enabling infrastructure is delayed, descoped, or redefined, affected downstream artifacts (schedules, ICDs, verification plans, ops concepts) must be updated and re-baselined under formal change control to prevent documentation/real-capability mismatches.

From an AS9100D-oriented quality perspective, this issue raises process consistency, disciplined readiness claims, and change control[131]. Rev D embeds risk-based thinking (Clause 6.1) and operational risk management (Clause 8.1.1) to improve predictability; it also mandates Control of Changes (Clause 8.5.6): identify and document proposed changes; review impacts on product quality, safety, and compliance; approve before implementation; maintain traceability; and inform/train relevant personnel. Overstating infrastructure readiness may yield short-term schedule optimism but often reappears downstream as integration friction, rework, cost growth, and lost margin—precisely what AS9100D seeks to avoid.

[129] Fassi, Guenov, and Riaz, "An Assumption Network-Based Approach to Support Margin Allocation and Management," *Proceedings of the Design Society: DESIGN Conference* 1 (2020): 2275–2284.
[130] INCOSE *Systems Engineering Handbook* v5.0 (INCOSE, 2023), Chs. 4–5.
[131] AS9100D (IAQG 9100, 2016), cls. 6.1, 8.1.1, 8.5.6.

Accordingly, enabling infrastructure should be handled as an explicit assumption carrier within assumption management. Each infrastructure dependent claim should identify:

1. What infrastructure is presumed to exist and at what readiness level (e.g., baselined interfaces, qualification unit completed, operations doctrine accepted),
2. Which downstream functions depend on it (science ops, timelines, verification scope, crew ops),
3. What re-evaluation triggers (schedule slips, design changes, failed integration or qualification milestones) will force reopening the assumption and re-baselining linked artifacts.

This discipline prevents infrastructure from becoming a hidden driver of campaign logic and improves credibility by separating technical demonstration from architectural/operational readiness. FSP's 40 kWe/10-year targets illustrate how program goals should be treated as maturing infrastructure, not "already available."

That distinction is indispensable because infrastructure being treated as present before it is truly integrated can quietly condition verification posture, operations concepts, and safety logic without anyone explicitly deciding to trust it. The remedy is then lifecycle integration, baselined configuration control, and risk-based change management tied to concrete infrastructure maturity evidence.

A new White House directive strengthens the infrastructure argument by making lunar surface power less speculative and more programmatically explicit. The National Security and Technology Memorandum (NSTM-3), issued on April 14, 2026, directs NASA to initiate—within 30 days—a program to develop a mid-power space reactor, with a lunar FSP variant ready for launch by 2030, alongside an option for a Nuclear Electric Propulsion (NEP) demonstration[132]. The memorandum specifies design preferences for the reactor, calls for multiple vendors followed by a rapid down-select, targets at least 20 kWe of power for a minimum of five years on the lunar surface, and requires that at least one design be extensible to 100 kWe. Analytically, this shifts surface power from a loosely anticipated enabling capability to a formalized campaign dependency. NSTM-3 does not remove uncertainty; it changes its character. It increases the likelihood that downstream plans will treat reactor availability as a more stable premise even though industrial-base readiness, technology maturation, and integration and deployment risks remain active variables.

The current request can likely support a narrower, reprioritized Moon to Mars program, but it does not appear comfortably sized for all the stated timelines and cadence ambitions without congressional add-backs, scope discipline with commercial substitution, or additional schedule slips. NSTM-3 is especially important in this context because it adds a new schedule-bearing requirement on top of an already compressed exploration agenda. The memorandum directs NASA to:

- Begin the program within 30 days
- Deliver the lunar FSP variant by 2030
- Achieve at least 20 kWe for five years
- Ensure one design can scale to 100 kWe

[132] White House, National Space Technology & Management Directive 3 (NSTM-3, April 2026).

It also directs the Department of Energy (DOE) to assist so industry can produce up to four space reactors within five years. Because NSTM-3 was issued on April 14, 2026—after NASA's FY 2027 budget materials dated April 3–7, 2026—it is reasonable to infer that the directive's full funding and workforce consequences are not yet reflected in the 2027 request.

A major qualifier, and potentially destabilizing factor, is that the presidential request is not the final budget. The agency does not operate solely on the discretionary request, which offers some protection. Even with that support, the core issue remains: **the campaign is trying to sustain *Artemis*, reprioritize Moon-to-Mars, integrate commercial capabilities, build infrastructure, and now execute explicit lunar-reactor direction under a flat discretionary request and a historically lean workforce.**

5.4 Resource, Workforce, and Governance Variability

Resource workforce and governance variability are not secondary management concerns in complex exploration campaigns; they are structural conditions that shape whether assumptions remain valid long enough to support design, verification, and operations. In a distributed program such as *Artemis*, assumptions about architecture, maturity, schedule, and integration are sustained not only by technical evidence, but also by continuity and funding, staffing, authority, and execution. When those conditions shift, the problem is not limited to work becoming harder to accomplish. The deeper problem is that assumptions formed under one institutional environment may continue to govern decisions even after the conditions that once played a part in justifying them have materially changed.

Campaign architectures depend on long chains of interdependent commitments and funding instability can alter what is realistically mature enough to carry forward. Workforce turnover can erode the interpretive memory needed to challenge inherited premises, and governance discontinuity can disrupt the alignment between political intent, program authority, and execution reality. Under those conditions, assumptions are more likely to persist by inertia rather than by active revalidation. Resource, workforce, and governance variability should therefore be understood not as background context, but as one of the principal mechanisms through which assumptions propagate and harden across the life cycle.

5.4.1 Budget Continuity and Execution Alignment

Budget continuity is one of the clearest ways resource variability becomes an assumption management problem. For *Artemis*, funding cannot be abated to being just a financial input; it is part of the condition under which architecture, verification, scope, workforce planning, infrastructure timing, and mission cadence are judged feasible. When the resource picture changes between presidential requests, congressional appropriations, and actual execution, the result is not administrative uncertainty but a shift in the institutional basis on which earlier assumptions were accepted as reasonable. In this environment, assumptions about continuity, maturity, and schedule can persist longer than their supporting fiscal conditions, making budget instability a direct contributor to assumption drift rather than a separate management issue.

This distinction is especially important for an agency such as NASA since budget stability cannot be inferred from top-line political support alone[133]. Presidential budget requests are proposals, not final enacted budgets, and even enacted funding must still be released and executed in a way that supports program continuity. For *Artemis* and similar long-duration campaigns, that means planning assumptions are exposed not only to appropriations outcomes, but also to the broader problem of alignment among proposal, appropriation, execution, and program authority.

Once that alignment weakens, the campaign can continue to operate on inherited premises about staffing, verification effort, infrastructure, sequencing, and schedule credibility that are no longer fully justified by the actual resource environment.

A major misconception about NASA is that it is a huge federal spender with a budget large enough to absorb delays, political churn, and architectural resets without much consequence. This is false. NASA is politically visible and culturally prominent, but fiscally modest. This inconsistency distorts public expectations about what the agency can sustain and how vulnerable long-duration exploration campaigns are to budget and governance instability.

This misconception persists because NASA is unusually visible. Rockets, astronauts, telescopes, lunar announcements, and Mars concepts make the agency feel larger than it is. Political rhetoric reinforces that effect by describing programs in billions without explaining how small those amounts are relative to national totals. Public favorability also matters. NASA is one of the most admired federal agencies, which makes it easy for symbolic scale to be mistaken for fiscal scale. Pew found that 67% of Americans viewed NASA favorably in 2024, and its earliest space-attitudes work found roughly three-quarters of Americans held a favorable view in 2022. That unusually strong public standing helps explain why NASA often feels larger than it is. The agency's involvement with the public, planetary exploration, science, and national prestige shows a symbolic footprint that far exceeds its fiscal one.

Budget comparison sharpens this point. NIH's FY 2025 appropriation was $48.5 billion, roughly double NASA's FY 2025 funding. EPA's FY 2025 budget was about $9.135 billion, and NSF's FY 2025 appropriations were about $8.826 billion. When set against macroeconomic totals, NASA is smaller still: U.S. GDP and GNP were each around $31.1 trillion in 2025. NASA is therefore significant as a civil space agency, but not large as a share of federal spending or national output. For NASA this is epochal, just as it is for the public, because people do not interpret delay or restructuring in a vacuum. When the public assumes NASA is a major spender, it becomes easier to see program instability as simple management failure rather than as a product of constrained discretionary funding, yearly appropriations politics, and shifting executive priorities.

The unfortunate budget reality is that NASA's own FY2026 Agency Fact Sheet lists the agency at $24.9 billion in FY2024 and $24.8 billion in FY2025. The President's FY2026 request drops NASA to $18.8 billion and keeps NASA at that lower level through the outer years, as shown in the fact sheet.

[133] NASA Budget Request Archive (NASA, 2026f); *The Planetary Society* FY 2027 Budget Analysis (Dreier, 2026).

This will impact key science and exploration initiatives while also hamstringing the grand plans that its administrator has started implementing. NASA's previous budget request archives explicitly note that presidential budget requests are proposals rather than final budgets, and that actual funding is determined only after congressional action. This distinction means that even the top-line budget number used in public debate may not be the number the agency ultimately operates under.

That symbolic footprint also sustains a recurring perception gap about NASA's budget. The Planetary Society's budget guide notes persistent survey-based overestimates, including a widely cited 2018 result in which respondents believed NASA received 6.4% of federal spending when the real figure was closer to one-half of 1%. The same guide shows that in 2025, NASA accounted for only about 0.35% of U.S. spending. Put differently, NASA often occupies a five-to-ten-times larger space in public imagination than it does in the federal ledger. Although highly visible, NASA's top-line number is small in federal terms. The Planetary Society highlights that when the federal government spent roughly $6.8 trillion in 2024—equivalent to 272 "NASA's" if NASA is treated as a $25 billion unit—NASA's budget remained firmly within the non-defense discretionary category, a narrow slice of the overall budget that must be re-justified annually. This context makes clear why the common refrain that "NASA gets so much money" is misleading: the agency competes within a constrained, annually negotiated segment of federal spending rather than commanding a dominant share of the total. What does not help is historical comparisons.

Historically, *Apollo* distorts public memory. NASA did briefly command a much larger share of federal spending during the 1960s, but that period was exceptional[134]. The Planetary Society notes that NASA's budget peaked during *Apollo* and that since the 1970s it has averaged about 0.71% of annual U.S. government spending, while since the 2010s it has generally been between 0.4% and 0.3%. In other words, *the period when NASA truly did consume a much larger share of federal spending is the exception, not the norm*.

In the modern era, NASA's budget share has been a fraction of 1% of total U.S. spending. The public often remembers the symbolic scale of *Apollo* while overlooking the much smaller fiscal reality of contemporary NASA. This misconception is analytically important because it encourages unrealistic expectations about what NASA can stabilize or absorb. If the agency is assumed to be lavishly funded, then schedule slips, descopes, and architecture changes look like self-inflicted failures. But if NASA is understood as a small discretionary account operating inside a contested budget environment, then budget continuity, execution timing, and governance alignment become visible as part of the assumption environment rather than as background noise.

Figure 8, below, illustrates that budget and staffing do not rise together, and that reduced workforce depth under constrained fiscal conditions can narrow the institutional margin available for integration, verification, and sustained campaign execution[135]. It also shows that NASA's present

[134] The Planetary Society FY 2027 Budget Analysis (Dreier, 2026).
[135] Aerospace Safety Advisory Panel Annual Report (ASAP, 2025) + NASA historical budget tables / Planetary Society dataset.

exploration push is being pursued from a comparatively lean institutional base relative to earlier periods of major human spaceflight expansion.

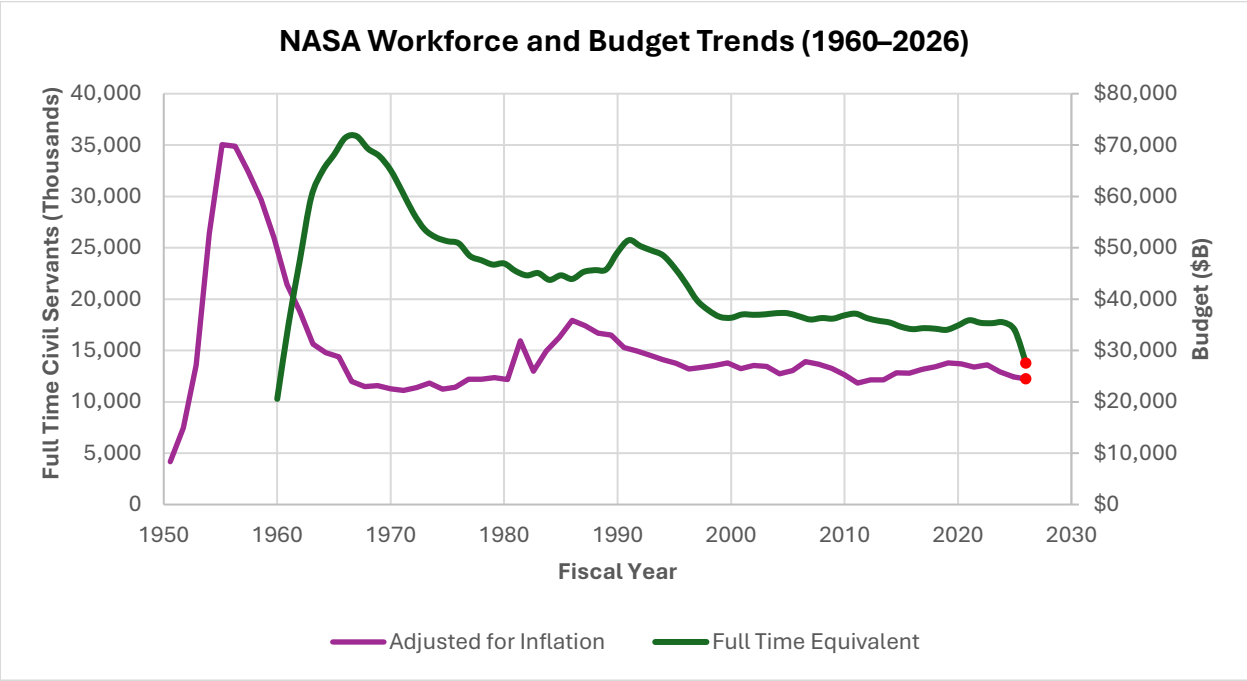


Figure 8 : Inflation Adjusted Budget and Civil Service Workforce Levels 1960-2026

This budget and workforce diagram also reinforces why budget continuity matters as an assumption management issue rather than as a simple accounting concern. As the diagram shows, NASA's workforce and inflation-adjusted budget both peaked far earlier than the present campaign era and have generally trended downward over the long term, even though short-term fluctuations and temporary recoveries appear at several points. The important point is not that budget and workforce move identically; they do not. The important point is that campaign planning occurs within an institutional environment that has less human and fiscal slack than what *Apollo*-era memory often implies. That makes continuity itself a more fragile premise than public discourse often assumes.

Read this way, the budget–workforce relationship helps correct a persistent misconception about NASA as a massively resourced agency. The historical trend instead suggests a more constrained reality: even when budgets remain nominally large, the agency's long-run operating base has narrowed relative to its historical peak. For a campaign architecture such as *Artemis*, that means that assumptions about integration capability, verification throughput, workforce availability, and schedule resistance cannot be treated as background constants.

They are conditioned by a resource environment that has become leaner and less forgiving across time.

A recent Senate Appropriations response illustrates this dynamic clearly. In January 2026, the Senate Commerce Committee described the FY 2026 appropriations bill as a bipartisan rejection of the

administration's proposed cuts, stating that the enacted legislation would provide NASA $24.44 billion, only slightly below the prior year's $24.88 billion, rather than the substantially deeper reductions proposed in the President's request. The same Senate statement specified line-item funding levels of $2.005 billion for the HLS, $7.25 billion for NASA Science, $920.5 million for Space Technology, $935 million for Aeronautics, and $143 million for STEM Engagement, while emphasizing that the administration's earlier proposal would have reduced NASA's budget by more than $6 billion[136]. Senator Maria Cantwell characterized the proposed cuts as jeopardizing economic competitiveness and national security, and the committee framed the enacted package as preserving America's ability to do big things in space. This shows that the campaign can be forced to operate between sharply different fiscal futures: one proposed by the Executive Branch and another stabilized, at least temporarily, by congressional action. Under those conditions, assumptions about staffing, verification effort, infrastructure timing, and mission sequence are not simply technical premises; they are governance-contingent judgments.

This is fundamentally an assumption management issue, because it shows that campaign planning may be forced to operate between sharply different fiscal futures: one proposed by the Executive Branch, another stabilized by Congressional action, and a third ultimately determined by execution. Under such conditions, assumptions about maturity, staffing, verification, scope, and mission sequencing may persist even while the resource base supporting them remains politically contested.

Budget instability affects more than just what gets funded; it also affects who stays in place to interpret, question, and advance the assumptions supporting the campaign. When resources shift in an exploration program, the effects are absorbed through people: who gets hired, who stays, which contractors remain, who provides technical leadership, and what institutional memory is preserved or lost. Workforce continuity is therefore not distinct from budget continuity; it is one of the key ways fiscal instability reshapes the assumptions that guide a campaign.

5.4.2 Workforce Continuity and Institutional Memory

Workforce continuity is essential to assumption management because assumptions are rarely preserved only in formal documentation[137]. They are also preserved in the interpretive memory of the engineers, analysts, safety personnel, managers, and contractors who understand why a given premise was accepted, what evidence originally bounded it, which caveats remain open, and under what conditions it should be revisited. This kind of knowledge is especially important because assumptions often outlive the moment in which they were first introduced. They are inherited into later design, verification, and operations phases, where their original meaning may no longer be obvious unless continuity of interpretation is actively maintained.

This problem is amplified in distributed programs such as *Artemis*, where technical work is spread across multiple NASA centers, contractors, commercial providers, and supporting organizations.

[136] U.S. Senate Committee on Commerce, Science, and Transportation (2026); FY 2026 NASA Appropriations Materials (2026).
[137] Aerospace Safety Advisory Panel Annual Report (ASAP, 2025); INCOSE Systems Engineering Handbook v5.0 (2023), Ch. 5.

Under these conditions, institutional memory is not merely a matter of retaining individual expertise; it is a matter of preserving cross-boundary understanding. A team may inherit a requirement, interface, constraint, verification result, or operational rule without fully inheriting the reasoning that once made it credible. When workforce turnover, restructuring, or contractor transition interrupts that chain of understanding, the assumption may remain in place while the context that once justified it is forgotten. In practice, this makes assumptions significantly harder to challenge, not because it has become more valid, but because the knowledge needed to interrogate it has thinned.

Figure 9 shows the year-to-year changes in NASA civil servant staffing levels and emphasizes that institutional continuity is shaped by headcount and by the total rate at which personnel are added or lost[138]. The pronounced outliers at the beginning and modern end of the series highlight moments of sharp expansion and contraction, underscoring how abrupt workforce shifts can disrupt technical memory, mentorship continuity, and the interpretative capacity needed to challenge inherited assumptions in complex exploration campaigns.

NASA's own workforce and knowledge policies support treating workforce continuity as an assumption-management issue. NPD 3010.1B states that workforce planning should reflect mission goals, work demand, and decisions about workforce size, makeup, and skill set across both the budget horizon and the longer term, and should align with acquisition strategy, facility master planning, and PPBE (NASA, 2022/2027, NPD 3010.1B, §1). NPD 7120.6A adds that NASA should capture, retain, use, and share technical and program/project knowledge and mitigate knowledge loss from attrition, demographic trends, and project closeouts (NASA, 2024, NPD 7120.6A, §1. a). In *Artemis*, workforce depth is therefore not a background management variable; it is part of the institution's capacity to keep assumptions visible, challenged, and revalidated.

Workforce continuity is especially sensitive not only to total staffing levels, but to the rate at which the workforce expands or contracts. The workforce change diagram is useful here because it shows that the most significant discontinuities occur at moments of sharp institutional transition, with pronounced outliers at 1961 and 2026. In one case, the agency was scaling rapidly into a new era of national commitment; in the other, the data suggests an abrupt contraction at the modern edge of the series. These moments indicate that assumption stewardship depends not just on who is present at a single point in time, but on whether expertise is being preserved or displaced quickly enough to disrupt interpretive continuity.

For long-duration exploration campaigns, abrupt workforce change is hazardous because the loss is not limited to headcount. It can also mean loss of rationale, caveat memory, and the human continuity required to recognize when an inherited assumption no longer matches present conditions. A system may retain its documents, requirements, and verification artifacts while still losing the people who understood why those artifacts mattered in the first place. Workforce

[138] Aerospace Safety Advisory Panel Annual Report (ASAP, 2025) + dataset source.

instability then becomes one of the mechanisms by which assumptions remain administratively intact while becoming interpretively thinner across time.

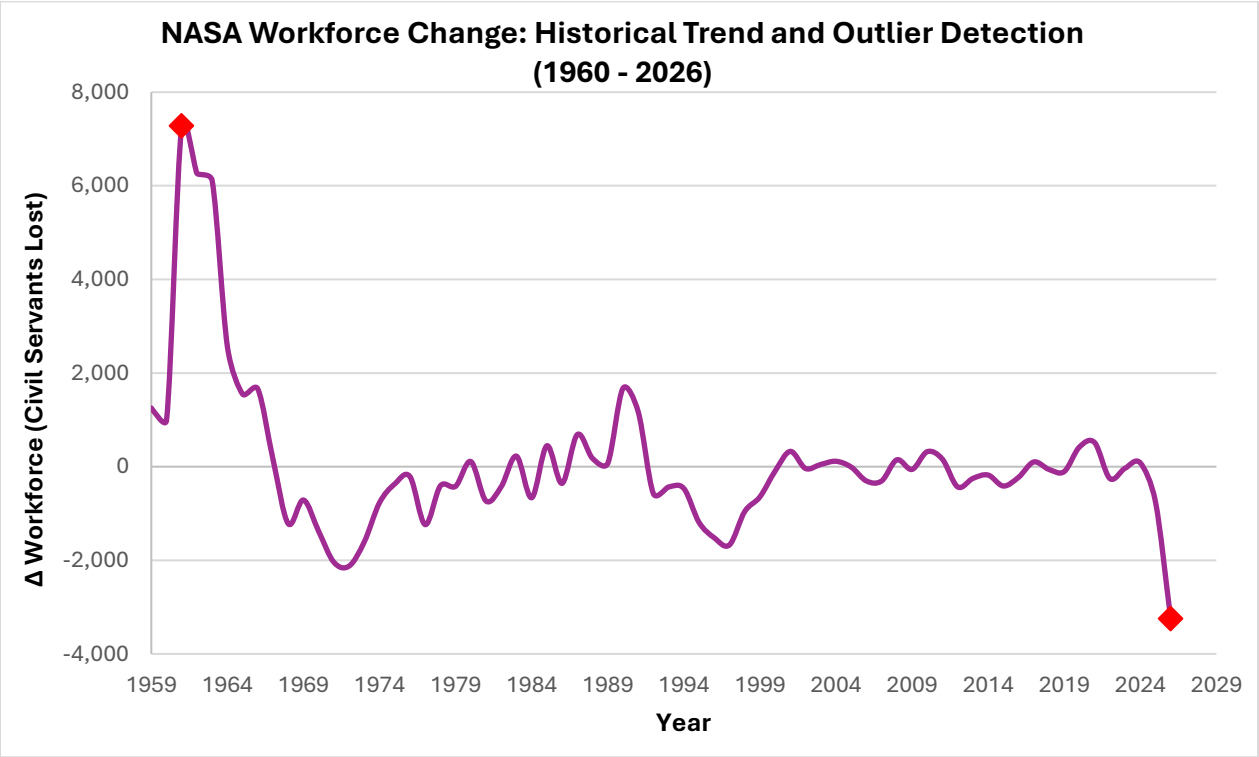


Figure 9 : NASA Workforce Change from 1960-2026

Workforce continuity, however, does not exist in isolation. The ability to preserve institutional memory, sustain specialized expertise, and challenge inherited assumptions depends on a broader governance environment that determines which priorities endure, what funding is protected, and which program elements are reinforced through law, appropriation, and execution. In long-duration campaigns, workforce instability becomes more dangerous when it is coupled with governance instability, because the system can lose the people who remember why the assumptions were accepted, but also the institutional alignment needed to reopen those assumptions when conditions change. For that reason, the problem of continuity in *Artemis* is not only a staffing issue or a budget issue. It is also a governance issue.

5.4.3 Governance Continuity and Assumption Validity

Governance continuity is the broader condition that links budget and workforce instability to assumption validity. Assumptions about mission sequencing, infrastructure availability, workforce growth, verification scope, and long-horizon integration do not remain credible simply because they were once approved in a design or program review. They remain credible only if the institutional conditions that supported them continue to hold. NASA's own budget archive makes clear that presidential budget requests are proposals rather than final budgets, while the U.S. Government Accountability Office's (GAO) Impoundment Control Act guidance states that even enacted funding

can be delayed or withheld through executive action or inaction[139]. In other words, continuity depends not only on what is proposed, but on the continuing alignment among proposal, appropriation, administration, and execution.

That distinction highlights that campaign assumptions are often formed under one governance condition and then carried forward under another. A plan may be developed on the basis of an executive request, partially stabilized through appropriations, and then reinterpreted again during execution. The issue is not purely that funding levels change, it is that the institutional basis on which assumptions were first judged reasonable may no longer match the political and administrative environment in which those assumptions are later used. When that happens, assumptions surrounding continuity, maturity, and feasible sequencing can persist longer than the governance conditions that originally supported them. Governance discontinuity therefore becomes one of the mechanisms by which assumptions drift without being formally reopened.

Legislative direction also shapes campaign assumptions more explicitly by embedding particular architectures, dependencies, and obligations into statute. The Budget Reconciliation Act of 2025 (H.R. 1) enacted 51 U.S.C. § 20306, providing $9.995 billion in special appropriations for Mars and *Artemis* missions, and the Moon-to-Mars program, including $700 million for a Mars telecom orbiter, $2.6 billion for Gateway, $4.1 billion for SLS for *Artemis* IV and V, $1.25 billion for the International Space Station (ISS), and $1 billion for infrastructure improvements at named NASA centers, among other directed purposes[140]. This influences assumption validity because it illustrates how governance continuity operates both in broad funding decisions and in laws that solidify particular parts of the program. Once priorities such as Gateway, ISS operations, Mars communications architecture, and center recapitalization are legislatively anchored in this way, they become easier for downstream teams to treat as fixed premises rather than as revisable assumptions. In other words, governance does not simply fund the campaign; it can also harden parts of the campaign's expected structure into planning reality.

For assumption management, the implication is clear: governance should be treated as part of the technical condition of campaign integrity rather than as external political background. A campaign can appear stable in program language while remaining unstable in the institutional arrangements that support it. If governance continuity is weak, then assumptions about feasible schedule, protected architecture, infrastructure readiness, workforce stability, and verification sufficiency may survive by administrative momentum even after their original basis has materially changed. Conversely, when Congress acts to preserve or legislate specific campaign elements, that action can stabilize some assumptions while hardening others. Assumption stewardship in *Artemis* therefore requires not only technical traceability, but also explicit attention to legal, fiscal, and institutional structures that determine whether the campaign's governing premises are still valid.

[139] NASA Budget Archive (NASA, 2026f); U.S. Government Accountability Office, *Principles of Federal Appropriations Law* ("Red Book") — Impoundment Control Act Guidance (GAO, 2024).
[140] 51 U.S.C. § 20306(a)(1)–(6), Special Appropriations for Mars/*Artemis*/Moon-to-Mars (U.S. Code, 2026).

Collectively - budget instability, workforce discontinuity, and uneven governance do more than complicate execution. They shape the environment in which assumptions are preserved and inherited across the campaign. These conditions will fluctuate, as is expected, but it is the fact that they fluctuate drastically while the campaign itself continues to move forward. Under those circumstances, assumptions can remain embedded in architecture, verification logic, and operational planning even after the institutional conditions that once justified them have changed. The cumulative effect is a form of pressure that encourages preservation over reconsideration.

5.4.4 Structural Misalignment: Demand vs. Capability

The combined effect of budget constraint, workforce instability, and schedule acceleration is not indicative of slower execution – that is not the actual concern. What it does, however, expose is the structural mismatch between the pace being demanded of the *Artemis* campaign and the institutional margin available to support that pace. NASA's FY 2027 budget request holds the agency at about $18.829 billion not only in FY 2027 but flat through FY 2031, while preserving exploration by sharply reducing other portions of the portfolio[141]. Within that same request, Safety, Security, and Mission Services falls from $3 billion in FY 2026 to about $1.999 billion in FY 2027, and Engineering, Safety, and Operations are shown at roughly $462 million. Even prior to any additional pressure being applied, this is not the profile of an agency being given abundant slack for rework, independent review, or expanded assurance steps.

The workforce picture reinforces these concerns. NASA's public facing pages are not consistent: one page reports just under 18,000 civil servants, while the current careers page lists 14,000[142]. The inconsistency is in itself revealing. Whatever the exact number, NASA is publicly presenting a comparatively lean civil service base for the scale and complexity of the Moon-to-Mars campaign. NASA's Aerospace Safety Advisory Panel's (ASAP) 2025 annual report situates current civil service levels in historical context and warns that major safety challenges increasingly arise not from isolated technical problems, but from the interaction of workforce dynamics, acquisition strategy, technical authority, budget pressure, and mission complexity.

Ther civil servant numbers play a large part in the bigger picture, because cadence is not a neutral scheduling preference. In this environment, cadence becomes a demand signal placed on systems engineers, verification pathways, throughput, independent judgment, mentorship, continuity, anomaly interpretation, and the retention of technical memory. The faster missions are stacked, the more the institution must rely on people and processes capable of reopening inherited assumptions rather than simply carrying them forward. ASAP warns that rapid attrition can erode technical authority, weaken mentorship pipelines, and diminish independent verification and validation capability. In parallel, NASA's Office of the Inspector General (OIG) has warned that *Artemis* must become fiscally sustainable and that the agency must sustain mission-critical capabilities—

[141] NASA Fiscal Year 2027 Budget Request (NASA, 2026g).
[142] Aerospace Safety Advisory Panel Annual Report (ASAP, 2025).

including technical workforce and aging infrastructure—even as it transitions and reprioritizes across the portfolio.

The relevance to *Artemis* is direct, consequential and germane. NASA's current public framing describes *Artemis* III as testing rendezvous and docking between Orion and a commercial spacecraft in LEO in preparation for a 2028 lunar landing, while the White House's April 2026 NSTM3 directs NASA to initiate within 30 days a program for a lunar FSP variant ready for launch by 2030. These are not isolated facts. Together they show that the campaign is simultaneously absorbing a redefined mission sequence requiring integration with commercial landers and a newly explicit infrastructure push around nuclear surface power. ASAP has already warned that *Artemis* III and subsequent missions combine too many first-time systems and operations into a single flight and has recommended reexamining objectives and architecture to create a more balanced approach to risk[143].

This is why historical analogy becomes analytically relevant. The *Apollo* 204 review board found that awareness of fire risk had become muted by prior experience, and that the assumption of low fire probability had been proven wrong. The Rogers Commission showed how decision makers during *Challenger* were operating with incomplete communication of critical concerns. *Columbia*'s accident investigation identified the dangerous assumption that prior success could be treated as evidence of continued success. Across all three, the common theme was not only technological failure **but a system losing the ability to challenge inherited premises under pressure.** This is why the present *Artemis* condition needs to be taken seriously: under physical compression, lean staffing, and accelerated tempo, assumptions are more likely to survive because the institution has less time—and less independent margin—with which to reopen them.

Recurrent fiscal instability can induce anticipatory retrenchment before formal reductions are fully absorbed, leading to workforce attrition, continuity loss, weaker integration and oversight, and reduced institutional recoverability. As resilience declines, programs become more vulnerable to further criticism, restructuring, and renewed pressure for budget and staffing cuts, creating a self-reinforcing fragility cycle as seen in Figure 10.

If this pattern continues, the damage will not be limited to schedule slip. It will begin to affect readiness and interpretation, verification claims, judgments, and the degree of deviation the organization becomes willing to normalize in the name of keeping cadence. A campaign may continue to appear to move forward even as its actual capacity for self-correction is thinning. It should be understood as a demand placed on a historically lean institution—one that risks shifting pressure into verification, judgment, integration decisions, and operational interpretation. Under these conditions, assumptions will become structurally preserved instead of drifting.

[143] Aerospace Safety Advisory Panel Annual Report (ASAP, 2025); NASA Fiscal Year 2027 Budget Request (NASA, 2026g); NASA Office of Inspector General, *NASA's Management of Major Programs* (NASA OIG, 2024); U.S. Government Accountability Office, *Annual NASA Major Projects Assessment* (GAO, 2025).

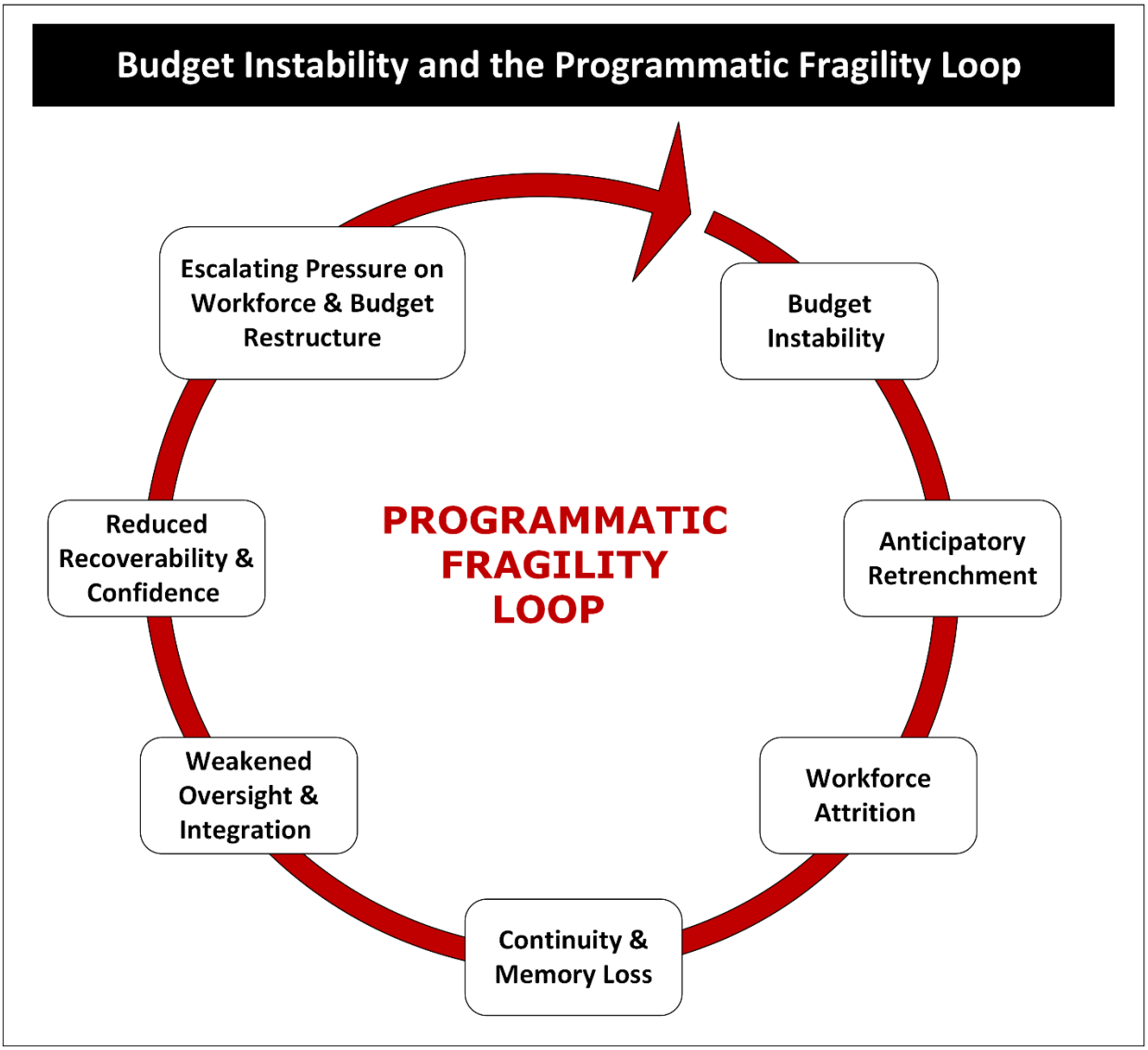


Figure 10 : Recurring Budget Shock and the Programmatic Fragility Loop

There is a limit to how much budgetary and programmatic whiplash NASA can sustain before its consequences will be felt. The programmatic fragility loop conveys that instability is not just an external stressor. Once retrenchment, attrition, and weakened oversight begin to interact, the agency inadvertently incorporates fragility into the very mechanism of its own decline. Given enough time, the instability will degrade workforce continuity, weaken technical judgment, eliminate industry experience, and narrow the margin that is needed to manage complex human spaceflight sagely and safely.

5.4.5 From Variability to Assumption Pressure

Resource, workforce, and governance variability exert pressure by converting uncertainty into implicit assumptions. A constrained or unstable budget limits what can be matured, tested, staffed, or deferred. Workforce turnover erodes the interpretive memory needed to challenge inherited premises. Governance instability alters the institutional basis on which assumptions were originally

judged reasonable. Each factor is consequential on its own, but in a distributed campaign they interact and reinforce one another.

Assumptions rarely fail abruptly; they persist across changing conditions unless they are deliberately reopened. A campaign may speak the language of continuity even as its resource base, staffing profile, and governance environment shift. In this context, assumptions about schedule, credibility, infrastructure readiness, verification sufficiency, partner maturity, and architectural sequence can endure through momentum rather than validation. The risk is not only technical error but the quiet normalization of premises whose foundations have weakened.

For *Artemis*, assumption management cannot be limited to technical interfaces or requirement traceability. It must also consider whether the campaign still retains the institutional conditions that once supported its judgments. Resource variability, workforce churn, and governance shifts are structural drivers of assumption drift and help explain why campaign logic can appear stable on paper while growing fragile in practice.

Assumption stewardship therefore requires periodic reassessment of the non-technical conditions that underpin technical decisions. Budget continuity, workforce continuity, and governance alignment are not background concerns; they define the environment in which assumptions remain valid. In complex exploration systems, preserving campaign integrity requires safeguarding the institutional continuity necessary to keep assumptions visible, challengeable, and appropriate to current conditions. Campaign architecture is shaped by technical coupling and by the resource, workforce, and governance conditions that determine whether outdated premises are revised or allowed to solidify into accepted truths.

Figure 11 exhibits how budget compression, lean staffing, and compressed cadence, and stacked first-time objectives can produce institutional effects that preserve assumptions across architecture, verification, and operations. It is a mechanism that shows how all of these factors convert assumption drift into assumption preservation. In a normally adaptive program, assumptions should remain revisable as new evidence, test results, operational constraints, and integration concerns emerge. However, when budget growth is constrained, staffing is lean, mission cadence is compressed, and multiple first-time objectives are stacked together, the organization has less slack to revisit earlier decisions. The institutional effects are cumulative: technical memory weakens, mentorship continuity thins, verification margin narrows, independent challenge capacity decreases, and schedule-driven decision pressure increases. Under these conditions, assumptions may survive not because they have been repeatedly validated, but because the program lacks the time, personnel, or governance capacity to reopen them. This is the central danger for future programs: **assumptions can become structurally preserved inside architecture, verification, and operations long after the original basis for accepting them has changed.**

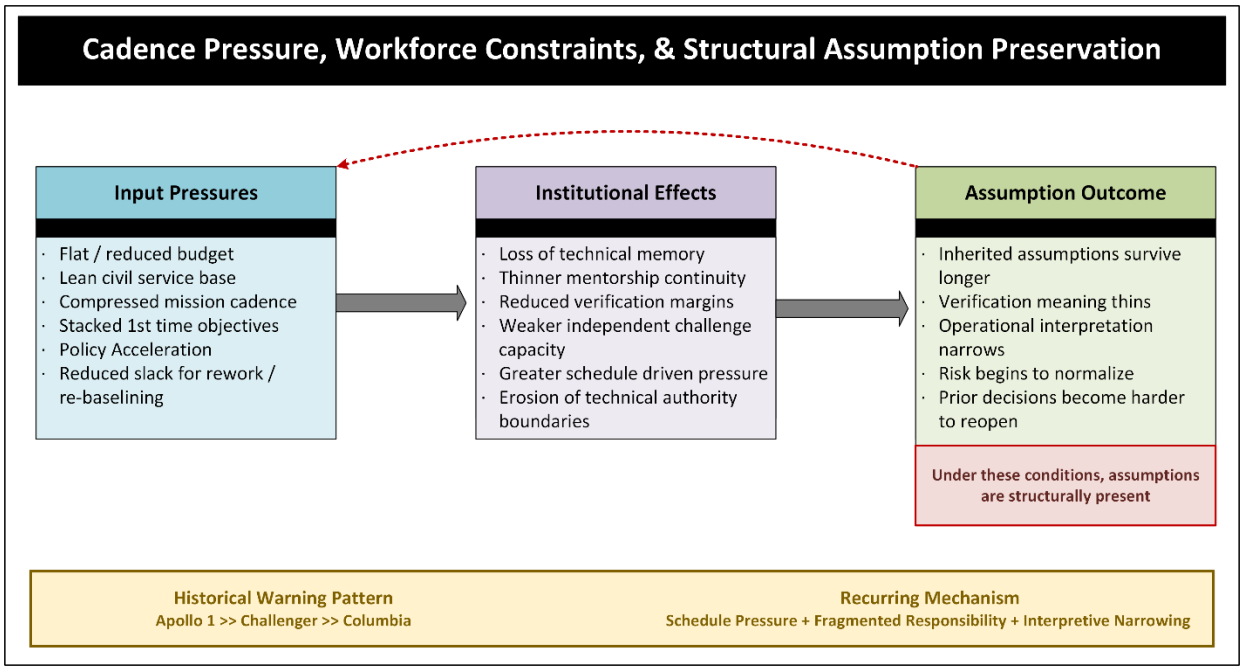


Figure 11 : External Pressures Driving Assumption Entrenchment

The structural drivers outlined in this section convey why assumptions and complex campaigns are difficult to keep provisional once they become embedded in architecture, verification, logic and institutional planning. Resource instability, workforce continuity, governance variability, distributed verification burdens, and infrastructure dependents complicate execution, and create the conditions under which assumptions are more likely to persist beyond the context that originally justified them. These pressures are especially visible in contemporary exploration systems, but they are not new. The historical cases in the following section show that assumption related failures in human spaceflight have long emerged at the intersection of technical conditions, organizational interpretation, and programmatic pressure.

6 Historical Context: Systemic Patterns in Human Spaceflight

Historical human spaceflight mishaps are often remembered through their proximate technical causes, yet their deeper significance lies in the organizational and programmatic patterns that allowed those causes to persist. In complex flight systems, failure rarely emerges from a single defective part alone. Rather, it develops within an environment in which assumptions about safety, performance, verification, and operational acceptability gradually harden over time. These assumptions may begin as provisional judgements, but under schedule pressure, repeated success, and distributed responsibility, they can become treated as settled truth without being formally revalidated.

For this reason, historical context is essential to the present analysis. *Apollo 1*, STS-51-L *Challenger*, and STS-107 *Columbia* are not included here as retrospective accident or anomaly narratives, they are used as reference cases for understanding how verification interpretation, organizational

learning, and assumption persistence behave in real programs[144]. Each case reveals a different manifestation of the same broader problem where technical evidence did not speak for itself, and decision makers operated within inherited interpretive frameworks that shaped what risks were seen, dismissed, normalized or deferred.

Examined together, these historical cases show that assumption related risk is not unique to *Artemis*. What changes in *Artemis* is the scale and distribution of the architecture in which such assumptions must be managed. The lessons of earlier programs, therefore more than historical background, establish the systemic patterns that make assumption management a legitimate safety concern in modern campaign-based exploration systems[145].

The historical meaning of *Apollo 1*, *Challenger* and *Columbia* cannot be separated from the lives that were lost. These mishaps are studied for their technical, organizational and safety lessons, but their names should not be treated as historical footnotes.

The crew of all three missions - Virgil "Gus" Grissom, Edward H. White II, and Roger B. Chaffee of *Apollo 1*; Francis R. "Dick" Scobee, Michael Smith, Judith Resnik, Ellison Onizuka, Ronald McNair, Gregory Jarvis, and Christa McAuliffe of *Challenger*; and Rick D. Husband, William C. McCool, Michael P. Anderson, Kalpana Chawla, David M. Brown, Laurel B. Clark, and Ilan Ramon of *Columbia* — remain central to the story.

Remembering them keeps the analysis grounded in its true stakes.

The importance of historical context in human spaceflight lies not only in understanding how these mishaps occurred, but in recognizing that failures of design, judgement, communication, and governance carried irreversible human cost. That human cost is what gives continued force to the lessons of *Apollo 1*, *Challenger,* and *Columbia*, and why those lessons must remain active in the evaluation of present and future exploration architectures.

6.1 *Apollo 1*

Apollo 1 was supposed to be the first crewed step toward the Moon — a straightforward test flight to prove the *Apollo* command module in Earth orbit. In the weeks before the mission, the three crew members worked tirelessly through long days of training and spacecraft checks, committed to shaping the new vehicle into something ready for flight. To many, *Apollo* felt on the cusp of real momentum.

On January 27th, 1967, the crew entered their spacecraft at Launch Complex 34 for a routine plugs-out test, a full countdown rehearsal with the capsule sealed and running on internal power. Despite communications issues and delays, nothing suggested imminent danger. The vehicle was still on the pad, engines inactive, and the test was considered low-risk. At 6:31 p.m., a spark ignited

[144] Report of the *Apollo* 204 Review Board (1967); Report of the Presidential Commission on the Space *Shuttle Challenger* Accident, Volume I (1986); *Columbia* Accident Investigation Board Report, Volume I (2003)
[145] Charles Perrow, *Normal Accidents: Living with High-Risk Technologies*

the cabin's high-pressure oxygen environment. Fire swept through the spacecraft in second. Ground teams fought to reach the crew as they tried to respond to the emergency, but the inward-opening hatch and rapid pressure buildup made escape impossible, and the crew was lost before the hatch could be opened[146].

This was the first major fatal accident in NASA's human spaceflight program, and it shook the agency to its core. The lessons learned reshaped the program's culture and strengthened every *Apollo* vehicle that followed. When *Apollo 11*[147] reached the Moon two years later, it did so on a foundation built partly from the sacrifice of the *Apollo 1* crew[148].

6.1.1 Designing Danger into Normalcy

If *Apollo 1* revealed anything, it was this: **danger does not arrive as a surprise**[149]. It accumulates in plain sight, layered into procedures, materials, assumptions, and pressures that everyone thinks they understand. The fire was not a sudden rupture in a pristine system. It was the ignition of vulnerabilities that had been tolerated, normalized, and rationalized over time—an oxygen-rich atmosphere, flammable cabin materials[150], wiring concerns[151], and a hatch that resisted escape[152]. None of these conditions were hidden. What hid them was false confidence.

Warnings had been logged. Risks had been raised[153]. But each signal was absorbed into the belief that the boundary between hazard and acceptability was nominal. The absence of prior catastrophe spoke louder than the hazards themselves. In that environment, risk didn't escalate—it blended into routine.

That is how a spacecraft becomes dangerous not through a single oversight, but through a series of unchallenged decisions that reshape the organization's sense of what is "manageable," "understood," or "within limits."

And the pressures were not solely technical. *Apollo* was a national commitment, a geopolitical promise[154], a program with an unstoppable trajectory. That trajectory narrowed the space for hesitation. It made momentum feel mandatory. When schedules begin to anchor perception, the first thing to erode is not equipment but the freedom to stop and question the frame. *Apollo 1* shows how

[146] Report of the *Apollo* 204 Review Board (1967), "Chronology from T–10 Minutes Through Medical Determination of Death," pp. 1–15.
[147] Report of the *Apollo* 204 Review Board (1967), "Findings, Determinations and Recommendations," pp. 18–32.
[148] *Apollo 11 Mission Report* (MSC-00190), Appendix I (1969).
[149] Report of the *Apollo* 204 Review Board (1967), pp. 1–15.
[150] *Report of the Apollo 204 Review Board (1967), "Findings, Determinations and Recommendations," pp. 18–32.*
[151] NASA Manned Spacecraft Center, *Apollo Operations Handbook: Block I Command/Service Module* (1966), sections on wiring and materials.
[152] Report of the *Apollo* 204 Review Board,*"Hatch Design and Crew Egress,"* (1967), pp. 80–95.
[153] NASA, *Apollo Program Safety Review and Hazard Analysis Summary* (1967), warnings logged prior to the plugs out- test.
[154] John M. Logsdon, *The Decision to Go to the Moon: Project Apollo and the National Interest* (1970), chapters on geopolitical and schedule pressure.

a system can converge toward disaster even while every individual believes they are acting responsibly.

The lesson *Apollo 1* drives home is simple and unforgiving: assumptions become dangerous long before they become fatal[155]. They become dangerous the moment they stop being tested. The fire did not erupt because no one recognized risk — it erupted because the organization convinced itself that those risks were contained, compartmentalized, and controlled. *Apollo 1* is not only a technical tragedy. It is the first proof that in human spaceflight, the most combustible material is unexamined certainty.

6.1.2 From Demonstration to Campaign Learning in the *Apollo* Program

The *Apollo* Program is a valuable addition to this discussion not only because it culminated in successful lunar landings, but because it shows how mission assumptions evolved within a program that is often remembered too simply as a single national sprint. *Apollo 11* was primarily a proof mission, its central purpose: accomplish the national goal set by President John F. Kennedy to land humans on the Moon and return them safely to Earth before expanding scientific goals[156]. *Apollo 11* was first and foremost about demonstration, credibility, and safe return. Science was present from the beginning, but it was not yet the dominant organizing logic of the landing campaign. The mission had to prove that a crewed lunar landing was operationally possible before more ambitious scientific and program level objectives could be meaningfully expanded[157].

Once this was successfully proven, *Apollo* did not remain conceptually fixed. The program achieved an additional five successful crewed lunar landings and one mission, *Apollo 13*, that demonstrated loss-of-mission recovery capability:

- ***Apollo 12* - 1969**
    - Moved beyond the first lunar landing and emphasized precision, repeatability, and disciplined surface operations[158].
    - Pinpoint landing near Surveyor-3 and deployment of *Apollo* Lunar Surface Experiments Package (ALSEP)[159] provided more confident field activity and created more sustained scientific returns.
- ***Apollo 14* – 1971**
    - Resumed lunar landings after *Apollo* 13 and deepened the scientific profile of the campaign through field geology, sample return, orbital photography, communications test, and engineering analysis in the Fra Mauro region.
    - Performed extensive geological investigations, including hauling the "Modular Equipment Transporter" (MET) and collected nearly 43 kg of lunar samples[160].

---

[155] NASA History Office, *Apollo 1 Fire: Organizational Causes and Lessons Learned* (1978)
[156] NASA *Apollo* mission reports (*Apollo* 12–17 series, 1969–1972).
[157] *NRO's Critical Role in Apollo 11's Safe Return from the Moon*, National Reconnaissance Office, Aug. 14, 2025
[158] *Apollo 12: The Pinpoint Mission*, *NASA*, July 8, 2009
[159] *First Look: Apollo 12 and Surveyor 3*, Arizona State University / NASA LROC, November 1969 (ALSEP deployment; Surveyor-3 visit).
[160] "*Apollo* 14 Lunar Samples," *LPI*, accessed April 3, 2026

- ***Apollo 15* – 1971**
    - Program entered a new operational phase altogether as the first "J-mission" with expanded surface stay time[161].
    - Landed near Hadley Rille and Apennine Mountains for advanced geological studies and had use of the Lunar Roving Vehicle (LRV) to widen traverse range and increase exploration depths.
- ***Apollo 16* – 1972**
    - Extended the logic from *Apollo* 15 into the Descartes highlands[162].
    - Emphasis turned to investigating a distant geological terrain while continuing orbital experiments and engineering evaluation.
- ***Apollo 17* – 1972**
    - Represented the full development of the landing campaign and was the final lunar landing.
    - A science centered mission in Taurus-Littrow[163], led by an astronaut-scientist and designed to maximize geological return, surface experiments, and orbital observations

The analytical importance of this progression is straightforward: *Apollo's* landing missions evolved from proving basic feasibility to extracting operational, scientific, and geological knowledge at increasing depth[164]. The question: *Can a human landing on the Moon be done?* was answered by *Apollo 11*. The program then expanded its inquiry: *Can it be done repeatedly? Can it be done precisely? Can it be done productively? Can it be done across different terrains while increasing mission duration, complexity, and scientific return?* This evolution shows that assumptions did not disappear after initial success but migrated. A program may solve problems surrounding feasibility only to inherit new assumptions about repeatability, equipment reliability, surface operations, scientific value, and campaign continuity. Even in *Apollo*, a comparatively bounded national program, those shifts were real. In more distributed modern campaigns, where architecture, authority, interfaces, and funding are less centralized, the governance challenge surrounding those evolving assumptions becomes more significant.

In systems engineering terms, this reflects what INCOSE describes as the **evolution of mission lifecycle assumptions[165]**, where early-phase feasibility assumptions give way to new assumptions that emerge through iterative learning, increasing system maturity, and the progressive refinement of lifecycle models.

[161] *Apollo 15: Mission Details*, *NASA*, July 8, 2009
[162] *Apollo 16: Mission Details*, *NASA*, July 8, 2009
[163] *Apollo 17 Landing Site, The Taurus-Littrow Valley – NASA Science*
[164] NASA SP-350, *Apollo* Program Summary Report (1975), chs. 2–4 & mission summaries.
[165] *INCOSE Systems Engineering Handbook (*2023), "System Life Cycle Processes and Models

6.2 *Challenger*

On the morning of January 28th, 1986, *Challenger* stood proudly on Kennedy Space Center's Pad 39B[166] under skies far colder than any previous *Shuttle* launch day. The crew of STS-51-L boarded for what they and NASA considered to be a routine flight, and a mission with historical significance. At Kennedy, families, schoolchildren, NASA personnel, and media gathered to watch, while a national audience followed along as the countdown reached zero and the *Shuttle* lifted off. The first minute was a vision of engineering success - everything was as it was meant to be. No anomalies were called out from Mission Control, the vehicle tracked its planned ascent corridor, and the bright exhaust plumes arced over the Atlantic.

Just over a minute into flight, something on the stack behaved differently. A flicker on the righthand booster, captured on ascent cameras – a brief and an almost unnoticeable disturbance against the backdrop of the exhaust plume and the clear sky. Inside the cockpit, the crew continued working through their ascent procedures, unaware that the External Tank had been compromised due to the hot gases escaping from a booster joint. Seconds later, a celebrated launch turned into one of the darkest moments in U.S. human spaceflight history as the entire vehicle broke apart and seven crew members were lost.

6.2.1 The Normalization of Deviance

If *Challenger* revealed anything, it was this: **the system had begun rewriting its own rules long before the plume appeared beside the right booster.** The Roger's Commission was unambiguous—this was not a cold-weather surprise or an unpredictable snap of rubber[167]. It was the predictable endpoint of a culture that had slowly absorbed repeated warnings into its definition of normal. What should have been a hard boundary between acceptable and unacceptable behavior became porous, then flexible, then meaningless.

O-ring erosion had become "expected[168]." Engineers had watched it appear flight after flight, and instead of escalating concern, leadership downgraded it. Erosion again? Blow-by again? The absence of catastrophe was allowed to outweigh the presence of a clear, recurring hazard. In that environment, uncertainty didn't trigger investigation, it triggered reinterpretation. "Normalization of deviance[169]" was not an academic label applied in hindsight; it was the operating logic of the *Shuttle* Program.

The pressure was not solely technical either. Launch rates climbed. Schedules tightened. Political commitments hardened[170]. Each of these forces narrowed the space in which engineers could raise

---

[166] Report of the Presidential Commission on the Space *Shuttle Challenger* Accident, Vol. I (1986), Ch. II — accident chronology.
[167] Report of the Presidential Commission on the Space *Shuttle Challenger* Accident, Vol. I (1986), Ch. V — "Contributing Causes of the Accident."
[168] Presidential Commission on the Space *Shuttle Challenger* Accident, *Report*, Appendix H: "Flight Readiness Review Treatment of ORing Erosion and -BlowBy," (1986-)
[169] Diane Vaughan, *The Challenger Launch Decision: Risky Technology, Culture, and Deviance at NASA* (1996)
[170] Report of the Presidential Commission on the Space *Shuttle Challenger* Accident, Vol. I (1986), Ch. V, pp. 142–156.

alarms and managers could truly listen. Under that pressure, the first thing to erode is not hardware. The first thing to erode is the willingness to stop and *challenge the assumptions*.

When an organization becomes more fluent at defending its assumptions than interrogating them, risk is no longer assessed. It is inherited

That is the lesson *Challenger* drives home**: assumptions do not simply persist, they metastasize**[171]. They harden. They become insulated from the very engineering scrutiny that once defined NASA's identity. Once technical evidence passes through an institutional narrative before reaching decision makers, it no longer functions as warning. It becomes justification. In such a system, terms like "flight-proven," "acceptable," and "within experience" drift away from engineering meaning and become cultural reflexes.

Challenger makes one fact impossible to ignore: disasters do not erupt from isolated decisions, but from the slow decay of vigilance that transforms yesterday's anomaly into today's expectation. O-ring erosion did not become deadly on a single frigid morning; it became deadly the moment the system began treating damage as "typical" and survival as validation. That is *Challenger's* enduring message: not that the joint failed, but that the culture failed first. When engineering doubt is softened into managerial confidence, risk boundaries don't shift. They dissolve.

Challenger is not only a story about the physics of cold rubber[172]. It is the story of how a high-performing, technically brilliant organization can talk itself into believing that a warning is noise. Once the burden of proof reverses, once the system demands engineers prove danger instead of requiring itself to prove safety, the countdown has already begun[173]. *Challenger* shows that the most hazardous condition in human spaceflight is not a flaw in hardware, but the quiet, accumulating conviction that repeated success means the danger is gone.

6.3 *Columbia*

Columbia lifted off from Kennedy Space Center on January 16th, 2003, for mission STS-107 carrying more than eighty experiments crammed into its middeck and Spacelab module for a long-awaited two-week microgravity and Earth science mission[174]. From the outside, it looked like a smooth and confident return to routine *Shuttle* operations – launching just four days after the seventeenth anniversary of *Challenger*. The dedicated seven-person crew threw themselves into the grueling research schedule which demanded days, nights, and alternating work shifts. Along the causeway and across the country, families, colleagues, and supporters followed the launch with pride.

In orbit, the crew was in their element – running experiments, downlinking results, sending updates to Earth, and completely unaware that during ascent a brief spray of debris from the External Tank

[171] NASA Office of Safety and Mission Assurance, *The Cost of Silence: Normalization of Deviance and Groupthink* (2014)
[172] Diane Vaughan, "*How the Challenger Disaster Became a Case Study of the 'Normalization of Deviance,'" Columbia* Magazine, Winter 2025–26
[173] Traci Purdum and Trish Kerin, *"Challenger Disaster: The Deadly Cost of Reversing Safety Burden,"* podcast, (2026)
[174] *Columbia* Accident Investigation Board Report, Vol. I (2003), Ch. 2b/2c — pages detailing final flight, pp. 39–45.

had struck the Orbiter's left wing, an event visible only in high-speed camera footage reviewed later on the ground. From their perspective, all was well, all systems appeared nominal, there was nothing to suggest that the reinforced carbon-carbon panels on the wing's leading edge had been damaged. Meanwhile, teams on the ground debated whether additional imaging was needed, emails and assessments were traded, and the outcome was: the mission will continue on schedule.

Sixteen days later, on February 1st, 2003, what had begun as a dedicated science flight became one of the most profound losses in U.S. human spaceflight history. As *Columbia* and her crew began their descent home, minutes after crossing the California coastline, rising temperatures inside the left wing led to structural failures too rapid to counter. The Orbiter broke apart high above Texas, ending the mission and claiming the lives of all seven astronauts.

6.3.1 Loss of Critical Viewpoint

If *Challenger* exposed how repeated anomalies become routine, then *Columbia* exposed something worse: **the collapse of NASA's ability to learn from its own history.** The CAIB was blunt—this was not a one-off mistake, not an unlucky hit of foam[175]. It was the inevitable outcome of a system that had forgotten the lessons written in the wreckage of *Challenger*.

Foam shedding had become "normal[176]." Managers had watched it happen flight after flight, and instead of escalating concern, they downgraded it. Prior fixes didn't work? That became irrelevant. The anomaly kept recurring? It became background noise. In that environment, evidence did not correct assumptions—assumptions reinterpreted evidence. The phrase "normalization of deviance" wasn't a metaphor; it was a diagnosis[177].

And the pressures did not push from the outside alone. Schedule commitments, ISS assembly deadlines, political promises[178]—each one tightened the frame through which engineers were allowed to speak, and managers were allowed to listen. When pressure enters the system, the first casualty is not hardware. The first casualty is the willingness to question the frame. Once an organization begins defending its assumptions instead of interrogating them, risk is not assessed. It is inherited.

The lesson *Columbia* underscores is that assumptions do more than persist, they take root[179]. They become resistant to scrutiny, buffered by organizational routines that gradually reinterpret uncertainty as stability. When technical evidence must conform to overarching program narratives before it can influence decisions, its role shifts: it stops acting as a signal and becomes a rationale. In that environment, terms like "tested," "acceptable," or "safe" lose their analytical meaning and instead reflect institutional comfort. Modern exploration systems that are spread across centers,

---

[175] *Columbia* Accident Investigation Board Report, Vol. I (2003), Ch. 7 — "Systems Engineering, Organizational Structures, and Safety Culture," pp. 97–104.
[176] *Columbia* Accident Investigation Board Report, Vol. I (2003), Ch. 7, "Foam Shedding History and Structural Vulnerabilities," pp. 47–60.
[177] *Columbia* Accident Investigation Board Report, Volume I (2003); Report of the Presidential Commission on the Space *Shuttle Challenger* Accident, Volume I (1986); Diane Vaughan, The *Challenger* Launch Decision (1996).
[178] *Columbia* Accident Investigation Board, *CAIB Report*, Vol. 1, "Organizational Causes," 169–198
[179] NASA Office of Safety and Mission Assurance, *The Cost of Silence: Normalization of Deviance and Groupthink* (2014)

contractors, and architectures face this same vulnerability. Risk stops being evaluated strictly through engineering insight and becomes entangled with legacy decisions, cultural expectations, and the inertia of the program itself.

In the end, *Columbia* forces us to confront the truth NASA never wanted to face: **disasters are not triggered by single decisions, but by the quiet accumulation of unchallenged assumptions.** *Shuttle* did not fail because engineers lacked data or managers lacked intelligence. It failed because the system had grown facile and comfortable. Comfortable with risk, comfortable with uncertainty, and comfortable with the idea that mission success meant safety.

Challenger was supposed to shatter that comfort. Instead, its lesson was absorbed, diluted, and ultimately narrowed to the O-ring, and the organizational failures that enabled it went unchallenged and remained embedded. *Columbia's* destruction is a historical event; but it is also a warning. Unless organizations continuously interrogate the anecdotes, they tell themselves about risk, reliability, and success, those stories will calcify. And one day, they will kill again. The next accident will not come from a foam strike or an O-ring. **It will come from the conviction that those failures are behind us.**

6.3.2 *Challenger – Columbia*: When Warning Becomes Noise and Noise Becomes Truth

The pattern.
An anomaly appears.
It recurs[180].
It doesn't kill us—today.
So the boundary between safe and unsafe blurs.
We rename the warning: *typical*.
We mistake survival for understanding.
That is not engineering. That is drift.

What changed.
For *Challenger*, O-ring erosion and blow-by moved from alarm to background hum.
For *Columbia*, foam strike took the same path—recurring, visible, unresolved.
In both cases, concern stayed alive in the trenches, but decision forums learned to live with it.
Flights succeeded; doubt softened; risk was rationalized.
The system recalibrated danger into familiarity.
Survival was misread as validation.

The inversion.
The burden of proof flipped.
Engineers were pressed to prove danger instead of leadership proving safety.

[180] Report of the *Apollo* 204 Review Board (1967), pp. 104–110; *Columbia* Accident Investigation Board Report, Vol. I (2003), Ch. 7, pp. 104–110.

In *Challenger*, the question became "show us it is unsafe.[181]"
In *Columbia*, the question became "show us it matters.[182]"
When organizations stop demanding evidence of safety, countdowns start silently.

The lesson.
Neither *Challenger* nor *Columbia* is only about hardware—cold rubber or falling foam.
They are about a system that slowly taught itself to trust repetition[183] over analysis, comfort over vigilance.
Risk boundaries don't move a little.
They dissolve.
And once the system begins treating a warning as noise and noise as experience, the next accident is already moving toward ignition.

6.4 Distributed Accountability in Historical Programs

A recurring lesson across major human spaceflight mishaps is that responsibility for safety can become fragmented long before a vehicle is lost[184]. Failures in historical programs did not occur because a single engineer, manager, or subsystem made one wrong decision in isolation. Instead, risk accumulated across organizational boundaries, technical disciplines, review chains, and decision forums in ways that diffused ownership of the final hazard. When accountability is distributed but not effectively integrated, critical concerns can remain visible in pieces without ever being acted on as a whole.

This pattern is especially important in the historical record because programs such as *Apollo* and *Shuttle* are often remembered as more centralized than modern exploration architectures. Yet even in these earlier programs, complex interactions among contractors, engineering teams, managers, operators, and review authorities created ambiguity over who was responsible for challenging assumptions, escalating anomalies, or curtailing momentum. Responsibility existed, but whether it remained coherent enough to convert technical concern into protective action was anything but guaranteed. Understanding that dynamic helps explain why major mishaps were not only engineering failures but also failures of accountability across the broader program structure.

6.4.1 Political and Programmatic Drivers of Risk

Political and programmatic constraints act as the first link in the assumption-formation chain. They determine what architectures are *feasible*, which risks are *acceptable*, how fast *development* must proceed, and where authority *resides* when concerns arise. In every major human spaceflight

[181] Report of the Presidential Commission on the Space *Shuttle Challenger* Accident, Volume I (1986), Chapter V — "Contributing Causes of the Accident," pp. 142–156
[182] *Columbia* Accident Investigation Board, *Public Hearing, April–May 2003: Debris Assessment Team Testimony* (Washington, DC: GPO, 2003).
[183] *Starship's Blaze: Can SpaceX's Culture of Iteration Outpace the Risks?*, TrendPulse Finance, June 20, 2025; Report of the Presidential Commission on *Challenger* Vol. I (1986), Ch. V; *Columbia* Accident Investigation Board Report, Vol. I (2003), Ch. 7; Diane Vaughan, *The Challenger Launch Decision* (1996).
[184] Report of the *Apollo* 204 Review Board (1967), pp. 104–110; *Columbia* Accident Investigation Board Report, Volume I (2003), pp. 104–110.

program, these forces have shaped the assumptions that guided design, verification, and operational planning, sometimes creating incentives that made it structurally challenging to revisit those assumptions once they hardened. Identifying these drivers is therefore essential to understanding how distributed accountability took root in historical programs, and why similar pressures now shape the early assumptions embedded within *Artemis*.

Throughout NASA's history, political priorities have consistently influenced the engineering and operational assumptions embedded within human spaceflight programs. These influences are not external 'noise,' but foundational drivers that shaped budgets, schedules, authorities, and risk tolerance. *Apollo 1, Challenger,* and *Columbia* also represent distinct political eras that each produced their own characteristic pressures on risk decision-making.

Understanding these political and programmatic forces is essential for understanding why certain hazards persisted and how specific assumptions solidified within each program.

Apollo is an excellent example that demonstrates that technical success does not guarantee strategic continuity. The United States did not stop returning to the Moon after *Apollo 17* because operations had become technically impossible – it stopped because the political and budgetary conditions that created the *Apollo* program did not survive *Apollo's* success in their original form[185]. The Moon landing objective had been defined within a particular Cold War moment, and a particular political consensus. Once that objective was achieved, the urgency that had justified extraordinary concentration of resources began to weaken. In that respect, the post-*Apollo* period is directly relevant to assumption management: one of the most consequential assumptions in any large exploration effort is that demonstrated capability will naturally produce sustained program continuity. *Apollo* shows that this is not necessarily true[186].

The timing of *Apollo's* contraction reinforces that point, NASA's budget reached its high-water mark in the mid-1960s[187] and began declining before *Apollo 11* actually landed on the Moon. Even when *Apollo* was producing historic achievements, the fiscal environment that had sustained it was already eroding. At the same time, the institutional basis for continued lunar operations was becoming less secure. *Apollo 20*[188] *was* cancelled because its Saturn V was needed for Skylab, and the Saturn V production line had already been shut down in 1968, leaving no open ended production bases for extended lunar campaigning. In September 1970, two more planned lunar missions were cancelled due to budget reductions, while juxtaposed with this, post-*Apollo* proposals for more expansive future human spaceflight were judged too ambitious and costly by the Nixon administration[189]. The result was not a collapse of engineering capability, but a shift in political willingness, fiscal tolerance, and national program priorities.

[185] NASA History Office (2020/2024); John M. Logsdon, *The Decision to Go to the Moon* (1970/2010);
[186] NASA's *After Apollo, What?* (2019) and Factually's *Why Humans Stopped Returning to the Moon* (2026).
[187] "How much did the *Apollo* program cost?" The Planetary Society. April 2026.
[188] "50 Years Ago: NASA Cancels *Apollo* 20 Mission." *NASA*. January 3, 2020.
[189] 55 Years Ago: Space Task Group Proposes Post-*Apollo* Plan to President Nixon. *NASA*. September 16, 2024.

Table 1 below summarizes how political and programmatic forces shaped risk assumptions across *Apollo 1, Challenger, and Columbia*. Each case reflects a different policy environment, yet all of them show how institutional pressures, mission expectations, and external commitments influenced the assumptions that guided engineering judgment and operational decision-making. These drivers formed the context in which hazards were interpreted, normalized, or overlooked, and they reveal how early pressures translated into durable assumption patterns within each program.

Table 1 : Political and Programmatic Pressures Underlying Three NASA Mishaps

Mishap	*Political and Programmatic Drivers*	*Assumption Impact*
Apollo 1	*Apollo* operated under Cold War urgency, a decade-end lunar goal, and intense national visibility. Program momentum carried political meaning, and delay risked not only technical setback, but symbolic failure	Schedule urgency and national commitment helped stabilize assumptions that the Block 1 spacecraft and plugs-out ground test were acceptable enough to proceed without full hazard reclassification.
Challenger	*Shuttle* was politically sold as a reuseable, routine airline-like transportation system that would fly frequently and economically. Civil, military, and commercial expectations reinforced pressure to sustain launch cadence.	Political commitment to routine access to space encouraged assumptions that recurring anomalies could be managed within normal operations rather than treated as evidence of deeper system fragility.
Columbia	By the *Columbia* era, *Shuttle* had become embedded in a policy environment that treated it as a productive national asset despite budget strain, aging hardware, and growing operational compromise.	Program continuity and institutional commitment helped preserve assumptions that known debris behavior was understood and bounded, reducing the likelihood of fundamental reinterpretation.

Implications for Assumption Management:

The common pattern across these cases is that political and programmatic priorities influenced not only schedule and budgets, but the very standards by which uncertainty was judged acceptable. Once assumptions become aligned with national urgency, operational continuity, or institutional commitment, they are less likely to be challenged even when evidence should have forced reinterpretation.

Key Lesson: assumption management must account for the way external pressure and internal program momentum can legitimize weakly supported premises, making them appear stable, necessary, or too costly to revisit.

6.4.2 Physical and Engineering Vulnerabilities in Distributed Systems

Physical and engineering vulnerabilities act as the material layer through which assumptions become dangerous. Political urgency and organizational fragmentation matter, but they become

lethal when they are embedded in systems whose physical design already contains narrow margins, hidden couplings, or weak recovery pathways. In human spaceflight, this means that assumptions about acceptable environments, component behavior, failure containment, and emergency response cannot be treated as separate from the architecture itself. They are often built into the hardware long before they are ever challenged in operations.

The three mishaps discussed here show that engineering vulnerability is rarely a single isolated defect. In *Apollo 1*, the *Apollo* 204 Review Board identified a sealed cabin pressure with 100 percent oxygen at 16.7 psia, extensive combustible materials, vulnerable wiring and plumbing, and inadequate provisions for crew escape, rescue, and medical response[190]. This led the Board to conclude that the test conditions were extremely hazardous. With *Challenger*, the Rogers Commission traced the accident to the fault Solid Rocket Booster (SRB) joint design and showed how repeated O-ring erosion and blow by were carried forward as acceptable despite growing evidence that the field joint posed a catastrophic risk[191]. During the *Columbia* investigations, the CAIB found that the foam impact on the wing leading edge created a breach that was not adequately observable or assessable during flight, turning an ascent debris strike into an unrecoverable reentry loss[192]. In all three cases, the hardware did not simply fail; it embodied vulnerabilities that prior assumptions had not adequately constrained.

Assessing physical and engineering vulnerabilities is essential because it reveals where hazards persist, even when they are only partially visible. The issue is not as straightforward as "a fire could occur", "an O-ring could erode", or "foam could strike the vehicle"; the deeper problem was that the surrounding programs had already accepted system conditions in which those vulnerabilities were easier to rationalize than to redesign. Table 2, below, summarizes how material weaknesses in *Apollo 1, Challenger*, and *Columbia* interacted with program assumptions and helped convert known or unknown hazards into accepted operating conditions.

[190] Report of the *Apollo* 204 Review Board (1967), pp. 18–32.
[191] Report of the Presidential Commission on the Space *Shuttle Challenger* Accident, Volume I (1986), chs. IV–V.
[192] *Columbia* Accident Investigation Board Report, Vol. I (2003), Ch. 7, pp. 47–60.

Table 2 : Physical and Engineering Vulnerabilities Across Three NASA Mishaps

Mishap	*Physical and Engineering Vulnerabilities*	*Assumption Impact*
Apollo 1	Flammable cabin materials, high-pressure pure oxygen atmosphere, inadequate emergency egress design, wiring faults, and poor harness routing created a highly ignition-sensitive environment during ground test.	Assumptions that the Block I spacecraft configuration was sufficiently safe for a plugs-out test led to acceptance of hazards that had not been reclassified or mitigated, permitting a lethal ground environment to persist.
Challenger	O-ring resiliency degradation in low temperatures; joint rotation under pressure; lack of full-scale, low-temperature testing; inability of the SRB joint design to tolerate thermal and pressure variability.	Assumptions that O-ring erosion and blow-by were "acceptable" and bounded by flight history created tolerance for a failure mode that was never fully understood or qualified.
Columbia	Reinforced Carbon-Carbon (RCC) panel vulnerability to foam impact; bipod insulation shedding; lack of validated impact models; insufficient testing of foam-to-RCC strike loads.	Assumptions that foam shedding was a maintenance nuisance rather than a critical hazard normalized repeated anomalies and masked the lethality of an untested impact regime.

Implications for Assumption Management:

Across all three cases, critical engineering vulnerabilities were not just technical defects, they became embedded in decision environments where teams assumed the system was better bounded than it really was. Assumption management therefore has to not only track component risk, but also whether the physical design creates hidden conditions that make escape, recovery or reinterpretation impossible once the system goes off nominal.

Key Lesson: distributed systems can preserve nominal confidence long after the underlying hardware has become interpretation sensitive. If the physical configuration narrows recovery margins faster than the organization recognizes, assumptions will harden around survivable-looking test or flight history rather than actual system resilience.

6.4.3 Organizational and Governance Fragmentation

Engineering failures in major human spaceflight mishaps are also failures of structure. Even when technical problems are visible somewhere in the system, they do not automatically become protective action. Concerns must travel through organizations, review chains, contractors, centers, management processes, and decision authorities capable of recognizing significance and forcing closure. When those pathways are fragmented, concerns can remain technically valid while becoming operationally ineffective. Under such conditions, assumptions do not persist because no one solved the problem, but because responsibility for challenging them is diffused across too many actors.

Apollo 1 demonstrates that even an earlier and more centralized era program could fail to convert responsibility into coherent protective action. The *Apollo* 204 Review Board found that the organizations responsible for the planning, conduct, and safety of the plugs out test failed to identify it as hazardous and had not prepared contingency procedures for crew escape or rescue. It also

found the overall communication system unsatisfactory during operations preceding the accident. *Challenger* shows the same structural problem in a more distributed review environment: the Rogers Commission found that NASA's anomaly tracking, and Flight Readiness Review (FRR) system failed despite a persistent history of Oring erosion and blowby, and that repeated waivers and reductions in the information carried upward weakened effective closure[193]. *Columbia* extended this pattern into a variegated *Shuttle* structure involving multiple contractors, centers, and civilian organizations. CAIB emphasized that communication across these organizations was paramount and recommended an independent technical engineering authority with no connection to schedule or program cost[194]. Across all three cases, fragmentation did not eliminate responsibility, but it weakened the program's ability to translate technical concern into decisive intervention.

Understanding organizational and governance fragmentation is therefore essential for understanding why assumptions harden in distributed programs. The problem is not that many institutions are involved, but that hazard recognition, dissent, waiver logic, and the authority to stop forward motion do not always remain aligned. The table below summarizes how fragmentation in *Apollo* 1, *Challenger*, and *Columbia* affected the way assumptions were escalated, reviewed, and preserved within each program.

Table 3 : Organizational and Governance Fragmentation Across Three NASA Mishaps

Mishap	*Organizational and Governance Fragmentation*	*Assumption Impact*
Apollo 1	Diffuse accountability between NASA centers, contractors, and program offices; unclear authority for hazard reclassification; fragmented reporting lines for safety concerns.	Assumptions that existing governance mechanisms were adequate led to acceptance of incomplete hazard reviews and failure to elevate known risks to decisive action.
Challenger	Fragmented communication between NASA, contractor engineering teams, and management; conflicting interpretations of O-ring data; managerial override of technical concerns; pressure from schedule and program expectations.	Assumptions that launch readiness could proceed despite technical dissent reflected a governance structure where ambiguity in authority allowed optimistic judgments to dominate.
Columbia	Disconnected mission management teams; limited access to imagery and analysis; insufficient challenge function; cultural normalization of foam shedding across organizations.	Assumptions that existing structures provided adequate independent review reduced urgency to challenge accepted explanations or seek external analysis during flight.

Implications for Assumption Management:

In each case, the problem was not that many organizations were involved – it is never as simple as that. It was that the **responsibility** for recognizing, escalating, and closing assumptions was

[193] *Apollo* 204 Review Board (1967), *Challenger* Commission Vol. I (1986), pp. 104–110, 142–156; *Columbia* Accident Investigation Board Report Vol. I (2003), pp. 169–198.
[194] *Columbia* Accident Investigation Board Report, Volume I (2003), pp. 169–198.

diffused. In distributed programs, assumption stewardship must therefore be aligned explicitly, with authority strong enough to override normal organizational compartmentalization.

Key Lesson: authority is distributed across technical disciplines, centers, contractors, commercial providers, safety organizations, mission managers, and political overseers. When no one actor owns the full meaning of the assumption, everyone can end up believing that someone else has already validated it.

6.4.4 Interpretive Drift and Assumption Persistence Across Programs

Interpretive drift occurs when repeated exposure to unresolved anomalies changes how those anomalies are understood. Instead of functioning as warnings, they become familiar. Instead of signaling that a system is poorly bounded, they begin to appear manageable, normal, or already incorporated into acceptable risk. This process is especially dangerous in long-duration programs because assumptions do not remain fixed at the moment they are first made. They are carried forward through repeated decisions, inherited by later teams, and reinterpreted in ways that make them seem more justifiable than the underlying evidence actually supports.

Apollo 1 shows an early version of this dynamic. NASA's summary of the *Apollo* 204 Review Board notes that the long history of successful testing and launching in pure oxygen environments contributed to overconfidence and complacency, making the danger of the plugs out test harder to fully appreciate. *Challenger* provides the clearest formal example: the Rogers Commission found that both NASA and Thiokol management had failed to recognize the joint design as a serious problem, then failed to fix it, and finally treated erosion and blowby as an acceptable flight risk[195]. The Commission explicitly noted that management increased the amount of damage considered acceptable and that the anomaly tracking system still permitted flight despite a persistent warning history. *Columbia* demonstrates the same logic in *Shuttle* form. NASA's CAIB materials emphasize that the *Shuttle* program's complexity demanded strong communication and independent technical authority, while other NASA hosted CAIB documents show that foam shedding had increasingly come to be viewed not as a safety of flight issue but as a maintenance or risk accepted issue. In each case, anomalous evidence did not disappear. Its meaning changed.

An awareness of interpretive drift reveals why certain assumptions remain in place despite the disappearance of the factors that initially validated them. A program may continue to operate with growing confidence even as the underlying anomaly record becomes more troubling, because repetition itself begins to substitute for proof. The table below summarizes how *Apollo* 1, *Challenger*, and *Columbia* each illustrate a pattern in which recurring, previously tolerated conditions gradually lost their warning function and instead reinforced the persistence of assumptions that should have been revisited.

[195] Report of the Presidential Commission on the Space *Shuttle Challenger* Accident, Volume I (1986), ch. V / App. H;

Table 4 : Interpretive Drift and Assumption Persistence Across Three NASA Mishaps

Mishap	*Interpretive Drift and Assumption Persistence*	*Assumption Impact*
Apollo 1	Hazard warnings lost urgency over time; concerns about oxygen atmosphere and flammable materials became background risks rather than active design drivers.	Persistence of outdated assumptions about acceptable ground-test risk allowed the environment to remain unchanged despite accumulating indicators of vulnerability.
Challenger	Gradual normalization of O-ring erosion and blow-by; repeated reframing of anomalies as operationally tolerable; erosion treated as data rather than risk.	Assumptions hardened around the idea that past survivability implied future safety, allowing drift from engineering judgment to historical precedent.
Columbia	Foam impacts repeatedly recast as minor or expected; requests for imagery and analysis were minimized; anomaly patterns treated as routine rather than indicative of systemic hazard.	Assumptions that foam loss was non-critical persisted despite new evidence, enabling continued operations under misinterpreted risk conditions.

Implications for Assumption Management:

The recurring failure mode is not merely a technical error, but the gradual reinterpretation of recurring anomalies until they no longer appear anomalous. Assumption management must therefore treat repetition itself as a warning sign: the longer an unresolved condition persists, the more likely the organization is to mistake familiarity for evidence.

Key Lesson: assumptions do not survive from one event to the next; they accumulate legitimacy through habit, schedule continuity, institutional memory, and previous survival. Persistence can make inherited assumptions seem more mature than they actually are.

6.5 Near Misses as Signals of Unresolved Assumption Drift

The one thing that major mishaps reveal in their most visible and irreversible form is systemic failure, but near misses are often just as equally revealing where the study of assumptions is concerned. Near misses do not end in loss of crew or vehicle, they do however leave weaker institutional memory, less external scrutiny and fewer forcing mechanisms for sustained reform in its wake. Their survival, however, makes them analytically valuable and helps to unearth the conditions under which hazardous assumptions remained active, but fortuitously happened not to culminate in catastrophe.

Near misses are then especially important to this discussion where assumption propagation and interpretive drift are concerned[196]. Close calls occupy the space between anomaly and catastrophe and give life to how hazardous conditions remain embedded within programs after mishaps – even with formal recommendations or visible reforms. In several cases, the system survived because the failure path was interrupted by contingency performance, improvised response, or favorable circumstance, and not because the underlying assumptions were valid. These near misses are

[196] NASA Mission Reports and official NASA History Office documents for *Apollo* 12, Skylab 3, *Apollo*–Soyuz, STS-1, STS-51D, STS-51F, STS-27, Mir-Progress, STS-93, STS-114, STS-117, STS-120.

important because they reveal the persistence of unresolved assumptions in a less final but often more ambiguous form. Unlike *Apollo*1, *Challenger*, or *Columbia*, they did not produce the kind of public rupture that forces immediate institutional reckoning, but they remained absorbed within a narrative of mission success, technical recovery, or operational resilience.

This ambiguity needs to be discussed, because of the value it brings to the table in highlighting how vulnerable a system is, and how its continued functioning may have obscured the significance of the warning. Viewed historically, near misses and close calls help bridge the gap between formal lessons learned, and actual lessons retained and absorbed. They show how hazardous assumptions have survived accident boards, redesign efforts and procedural reforms – either because the next threat emerged adjacent to the previous causal chain or because the organizational procedures were adequate. The cases in this section therefore serve as signals of unresolved assumption drift across otherwise successful programs and are grouped by historical period to emphasize how unresolved assumptions persisted after major mishap investigations.

Post–Apollo 1 to Pre–Challenger (1967–1986)

Apollo 204 Review Board recommendations and near misses in Table 5 show which investigation recommendations existed, whether/when NASA implemented them, and how each event fits against those recommendations.

Table 5 : Post Apollo 1 - Significant Unresolved Events

Incident	Event Description	Assumption Signal	Relevance
Apollo 12 (1969)	*Launched into an overcast stratocumulus in defiance of a rule that no vehicle be launched into a thunderstorm. Was subsequently struck by lightning –* ***twice.***	*Launch environment was treated as sufficiently bounded by existing weather and vehicle certification rules. The event showed that the vehicle itself could trigger a hazardous electrical pathway*	*Shows that Apollo 1 reforms addressed the fire, materials, egress, and test disciplines but did not automatically bound adjacent launch environment hazards.*
Skylab 3 (1973)	*An oxidizer leak forced the isolation of one reaction control quad, and a second leak triggered rescue mission operations.*	*Redundancy was treated too easily as independence. The event showed that backup systems may share common vulnerabilities or produce uncertainty about common cause failure.*	*Strong false redundancy case. It shows that backup capability only supports safety if its independence is understood, verified, and operationally protected.*
Apollo-Soyuz (1975)	*During reentry, unignited reaction-control propellants vented and were ingested through the cabin air intake, exposing the crew to toxic fumes and sending them to the hospital.*	*Configuration discipline, vent-path behavior, and cabin-intake placement were treated as inherently safe during terminal phases, masking a latent pathway between propellant venting and crew exposure.*	*Highlights that nominal reentry phases can still hide life-threatening environmental couplings, reinforcing the need for integrated design-ops verification of airflow, venting, and late-phase procedures.*
STS-1 (1981)	*Sustained significant damage to its thermal protection system from the overpressure wave of the solid rocket boosters and revealed unexpected thermal stresses on an early shuttle flight.*	*Acoustic, structural, and thermal environments were treated as sufficiently captured by analysis and scaled testing and ground purge hazards were treated as administratively controllable rather than dynamically hazardous.*	*Shows how first-flight certification can underestimate real ignition loads, and how organizational safety processes must be validated against full-system behavior, not paperwork or partial testing.*
STS-51D (1985)	*After a deployed satellite failed to activate, the crew attempted an improvised EVA "flyswatter" fix that was unsuccessful, and the orbiter later landed with brake and tire damage that prompted future landings to shift away from KSC.*	*Autonomous payload activation, ad-hoc EVA fixes, and KSC landing conditions were treated as operationally reliable without proof that the supporting margins or mechanical states could withstand off-nominal behavior.*	*Illustrates the limits of improvisation in orbit and the need for conservative ground-risk models; mission "success" obscured vulnerabilities in both payload integration and landing-site policy.*
STS – 51F (1985)	*A main-engine sensor problem triggered the Shuttle's first Abort-To-Orbit, forcing the crew to complete the mission with reduced propulsion margin and a replanned scientific program.*	*Sensor redundancy, failure-detection logic, and abort procedures were viewed as robust enough to handle ascent anomalies, but the event exposed how tightly coupled sensor validity is to real-time abort decision-making.*	*Demonstrates that ascent survivability depends not only on the existence of abort modes but on their operational executability under imperfect data, validating Shuttle abort architecture through real, high-stress activation.*

These post *Apollo* 1 cases show that reform can be real and still remain bound by the causal frame[197]. *Apollo* 1 produced major changes to spacecraft design, materials, and emergency procedures, but later events revealed unresolved assumptions in launch environments, redundancy independence, toxic exposure, reuseable Thermal Protection System (TPS) behavior, and ascent recoverability.

Post–Challenger to Pre–Columbia (1986–2003)

Rogers Commission Board recommendations and near misses in Table 6 show that Rogers-era reforms did not eliminate adjacent pathways of vulnerabilities and contingency performance.

Table 6 : Post Challenger - Significant Unresolved Events

Incident	Event Description	Assumption Signal	Relevance
STS-27 (1988)	*Ascent debris struck Atlantis and caused one of the most severe tile-damage cases in Shuttle history, leaving a missing tile and hundreds damaged. Forced the crew to reenter with uncertain survivability.*	*Debris-environment models and onboard inspection methods were treated as sufficiently bounding the ascent TPS hazard, but the classified-mission imagery constraints exposed a gap between expected inspection fidelity and what was required for risk disposition.*	*Shows how survivability depended on unverified TPS tolerance and inadequate imagery pathways—an early, unheeded signal of systemic debris-risk underestimation that later became central to Shuttle safety reforms.*
Mir-Progress Collision (1996)	*Progress resupply vehicle struck the Spektr module, breached the hull, caused depressurization, and forced the crew to isolate the module.*	*Manual docking using limited visual cues was treated as operationally acceptable, and station wiring/power routing was treated as robust to single-module loss—yet the collision exposed fragile cross-module dependencies and inadequate situational-awareness safeguards.*	*A pivotal demonstration that long-duration platforms require validated autonomy, multi-path redundancy, and hazard-tolerant layout; informed later ISS docking, fault-management, and crew-response doctrine.*
STS-93 (1999)	*A hydrogen leak from a main engine injector failure, combined with electrical shorts that tripped engine controllers, caused premature MECO and left the vehicle in a lower-than-planned orbit requiring OMS recovery burns.*	*Engine controller wiring integrity and injector hardware retention were treated as stable and isolated failure domains, but the simultaneous electrical and mechanical faults revealed hidden coupling between propulsion health and avionics reliability.*	*Highlighted that ascent safety depends on eliminating wiring-chafe and single-point injector vulnerabilities; led to major fleet-wide wiring inspections and reinforced the need for robust fault-tolerant propulsion monitoring.*

The post-*Challenger* cases are especially important as they separate implementation from absorption[198]. Roger's era reform changed the program, but later events show that adjacent hazards, latent maintenance defects, and repeated anomaly patterns were still interpreted too narrowly.

[197] NASA NTRS and NASA History mission reports including *Apollo 12 Mission Report*, *Skylab Mission Reports*, *Apollo–Soyuz Test Project Report*, *STS-1*, *STS-51D*, *STS-51F*.
[198] STS-27 Mission Report (NASA, 1988); ISS/Mir collision reports (NASA, Roscosmos joint reports).

Post–Columbia (2003–present)

CAIB recommendations, near misses, and recovery cases in Table 7 highlight both visible implementation of CAIB lessons and the continued dependence of human spaceflight on recoverability and contingency performance.

Table 7 : Post Columbia - Significant Unresolved Events

Incident	Event Description	Assumption Signal	Relevance
STS-114 (2005)	*Discovery launched with continued foam shedding concerns, and the crew conducted an unprecedented EVA to remove protruding gap fillers from the orbiter's belly to ensure safe reentry.*	*External-tank redesigns and TPS margins were treated as providing acceptable risk, yet the observed foam release and gap-filler protrusion showed that pre-Columbia hazards persisted, and that real-time TPS repair capability was essential, not optional.*	*Established on-orbit inspection and repair as part of Shuttle crew survival doctrine and shaped the post-Columbia ET improvement and imagery architecture.*
STS-117 (2007)	*Atlantis suffered a torn thermal blanket near the OMS pod and a jammed solar array, forcing a contingency EVA blanket repair and repeated manual work to safely stow the array.*	*Thermal-blanket behavior during ascent and automated array mechanisms were treated as sufficiently predictable; the anomalies revealed sensitivity to minor mechanical shifts and inadequate provisions for nominally "routine" deploy/stow cycles.*	*Drove expansion of Shuttle-ISS contingency EVA techniques, highlighting the importance of preparedness for high-workload, unplanned repairs on critical external surfaces.*
STS-120 (2007)	*A solar array ripped during deployment, prompting a high-risk EVA on the end of the Orbiter Boom Sensor System to stabilize the live array using improvised "cufflink" braces.*	*Array deployment loads and mechanical tolerances were treated as bounded by design margins, and no standard tools existed for repairing energized, semi-deployed arrays—revealing a gap between structural assumptions and real operational behavior.*	*Demonstrated the need for flexible, extreme-reach EVA procedures and validated astronaut improvisation as a last-resort safety layer in modular space-station assembly.*

The post *Columbia* cases show both sides of learning: STS-114 and STS-117 demonstrate that CAIB lessons were visibly embedded into inspection and repair practice[199]. While STS-120 shows that complex on-orbit assembly still depended on real-time engineering judgement, crew ground coordination, and improvised recovery when planned operations encountered unexpected system behavior.

Across these post-*Apollo* 1, *Challenger* and *Columbia* events, a consistent pattern emerges: formal learning often occurred, but it did not always become durable learning. NASA and its partners repeatedly improved hardware, procedures, inspection practices, and safety governance after major accidents. Yet close calls continued to reveal adjacent hazards, false redundancy, normalized anomalies, and dependence on real-time recovery. These events show that a system can be safer than before yet still carry forward unresolved assumptions. For future programs, the caution is that

[199] *Columbia* Accident Investigation Board Report, Volume I (2003), CAIB Recommendations; NASA Return-to-Flight documentation for STS-114, STS-117, STS-120.

reforms must be continually tested against new configurations, new operating environments, and new organizational pressures.

6.6 Caution and Implications for Future Programs

The historical record in human spaceflight does not support the comforting view that major failures arise only from isolated technical mistakes or unusually poor decisions made at singular moments. *Apollo* 1, *Challenger*, and *Columbia* reveal a more difficult truth: **catastrophic outcomes can emerge when programs gradually lose the ability to discern which assumptions remain valid, which have become conditional, and which have silently hardened into institutional fact.** The warning for future programs is that lessons and mishaps are not always due to technical fault, but also organizational, interpretive, and temporal. A system can remain operational long after its underlying assumptions have begun to drift.

Vigilance is especially important for modern exploration campaigns such as *Artemis*, which are rarely bounded or self-contained efforts. *Artemis* is a highly distributed environment composed of multiple vehicles, contractors, governing bodies, software ecosystems, mission phases, and evolving political commitments. In such settings, assumptions do not fail all at once. They migrate across organizations, documents, interfaces, and generations of personnel. An assumption introduced as a temporary simplification early in design may later be treated as a requirement, an operational constraint, or a proven fact simply because it has survived repeated use without being revisited. The danger is not only that assumptions may be wrong, but that their status can change without clear recognition. Here, interpretive drift becomes particularly consequential.

The meaning of a risk margin, test result, anomaly, or operational precedent can shift over time even while the language used to describe it remains unchanged.

The cases examined in this section warn against overvaluing visible technical performance while undervaluing degraded interpretive performance. Programs can continue to meet milestones, conduct tests, and generate documentation even as their capacity for critical questioning erodes. Under such conditions, recurring anomalies may become normalized, dissent may be procedurally acknowledged but substantively neutralized, and schedule or budget pressures may quietly redefine what counts as acceptable evidence. This does not necessarily produce open negligence. More often, it yields a system increasingly confident in its own internal logic while becoming less sensitive to contradiction. This pattern has appeared repeatedly, and it is especially dangerous in human spaceflight, where the most consequential failures are often preceded by ambiguous signals that seem manageable until they are reinterpreted too late.

For future programs, the central implication is that assumption management must be treated as an ongoing governance function rather than a one-time analytical task. Assumptions should not remain buried in trade studies, inherited models, inefficient agreements, or informal expert memory. They must be made explicit, periodically revalidated, and tied to the decisions, evidence, and operating conditions that justify them. Just as requirements, hazards, and risks are tracked throughout program

evolution, critical assumptions should be traceable across lifecycle transitions[200]. This is particularly important when programs move from concept development to integration, from testing to operations, or from one institutional configuration to another. Each transition alters context—and a change in context is often what transforms a once-reasonable assumption into a hidden vulnerability.

A second implication is that distributed accountability must not become diluted accountability. Large, multi-organizational architectures lack a single author who can perceive the full consequences of assumption drift, even when each actor behaves reasonably within their local responsibilities. Therefore, future program structures need mechanisms that do more than assign ownership of components; they must preserve ownership of cross-cutting assumptions. This includes assumptions about interfaces, environments, operational use, maintenance, communication pathways, and off-nominal behavior. Responsibility tends to fragment precisely where integrated interpretation is most needed, and it is in this space that such mechanisms play a critical role.

A final warning from the historical record is that lessons learned are not self-executing[201]. Organizations often preserve the language of past lessons more successfully than the discipline required to apply them under new conditions. Invoking *Apollo* 1, *Challenger*, or *Columbia* is not sufficient on its own to prevent future mishaps. The real test begins when programs build structures capable of detecting when familiar assumptions are being stretched beyond the context in which they were originally formed. The enduring lesson is not simply to "listen to warnings," but to create systems in which warnings remain legible, assumptions remain contestable, and evidence can still alter the course of a program before accumulated drift becomes irreversible.

6.7 Assumption Propagation in Campaign Architectures

Historical human spaceflight failures are often remembered through the technical events that immediately preceded loss: a flammable pure-oxygen cabin environment, an eroded seal, or a foam strike. Yet major investigation reports show that catastrophic or near-catastrophic outcomes do not arise from hardware alone; they originate in organizational environments where assumptions about acceptable performance, adequate evidence, operational readiness, and decision authority persist longer than they should. The deeper historical pattern is not merely technical error, but the hardening of assumptions across time, structure, repeated decision cycles, and mindset.

The relevance of these cases becomes sharper when viewed against the structure of *Artemis*. Unlike *Apollo* or the *Shuttle Program*, *Artemis* is a campaign architecture—dependent on multiple vehicles, infrastructure elements, commercial providers, international partners, and evolving operational

[200]NASA, *NASA Systems Modeling Handbook for Systems Engineering*, NASA-HDBK-1009A (2025), pp. 5, 8, 47–49, 73–78, 81–82.
[201] NASA OSMA *The Cost of Silence*.; *Columbia* Accident Investigation Board Report, Volume I (2003), pp. 169–198; Diane Vaughan, *The Challenger Launch Decision* (1996);

sequences—where assumptions can propagate across program boundaries before they are fully stress-tested in integrated operations. This is not theoretical: external oversight has warned that *Artemis* III concentrates too many "firsts," and has documented schedule delays, technical difficulties, integration challenges in the HLS, and the absence of a current rescue capability should crew become stranded in space or on the lunar surface[202]. In such an environment, architectures do not fail solely when a subsystem underperforms; they become vulnerable when early-formed assumptions remain active after the surrounding architecture has changed.

This is why historical mishap literature belongs here. The purpose is not to force a simplistic analogy between old and new programs, but to show that the underlying safety problem has become more structurally demanding, not less. If past systems experienced catastrophic consequences when assumptions drifted inside one major program, then a distributed lunar architecture with multiple "firsts," major contractor dependencies, and evolving operational architectures has even greater need for explicit assumption ownership, lifecycle review, and interpretive governance.

Assumptions mutate as campaigns evolve: **a cadence assumption can become a logistics assumption; a logistics assumption can become a verification boundary; a verification boundary can become an operational risk posture.**

By the time a later mission depends on that chain, the original premise may be difficult to identify—let alone revalidate. From the historical perspective, *Apollo 1, Challenger,* and *Columbia* together show how complex systems become brittle: not through a single mistake, but through the untracked inheritance of meaning across time and structure. *Apollo 1* demonstrates how a misjudged test environment can create a lethal failure pathway before flight ever begins; *Challenger* shows how repeated anomaly exposure can redefine risk as normal; and *Columbia* illustrates how organizations can lose the ability to challenge their interpretive frame.

Section 6 therefore establishes the historical foundation for the rest of the paper. *Apollo 1* highlights how hazardous conditions can be misinterpreted as routine, *Challenger* reveals how anomaly can become normalized, and *Columbia* exposes erosion of critical organizational challenge. *Artemis* inherits all of these patterns within a campaign form that amplifies their interaction. The historical record thus supports the central argument that assumption management is not ancillary to safety in modern human spaceflight; it is increasingly one of its organizing necessities.

The patterns traced through *Apollo* 1, *Challenger*, and *Columbia* become even clearer when examined not as isolated events but as linked transfers of meaning and structure across programs. Figure 12 synthesizes these cases to show how hazardous interpretations, organizational habits, and pathways of escalation were inherited forward—often silently—into later decisions. By mapping how routine-treated hazards, normalized anomalies, and constrained interpretive frames migrated from one program context to the next, the figure highlights the mechanism by which brittle

[202] NASA Office of Inspector General, *NASA's Management of the Human Landing System Contracts* (IG-26-004, 2026);

architectures form: not through any single failure, but through the cumulative, untracked inheritance of assumptions. This conceptual view provides the footing for the argument that follows, illustrating why assumption behavior must be treated as an active system component rather than an incidental byproduct of technical performance.

Taken together, these cases establish the central historical claim of this paper: in complex human spaceflight systems, safety is degraded when assumptions about evidence and risk persist without explicit ownership, periodic reinterpretation, and lifecycle-aware governance. The significance of *Artemis* is therefore not that it introduces a new class of safety problems, but that it combines known historical patterns with a campaign architecture even more dependent on cross-program assumptions, evolving interfaces, distributed responsibility and accountability.

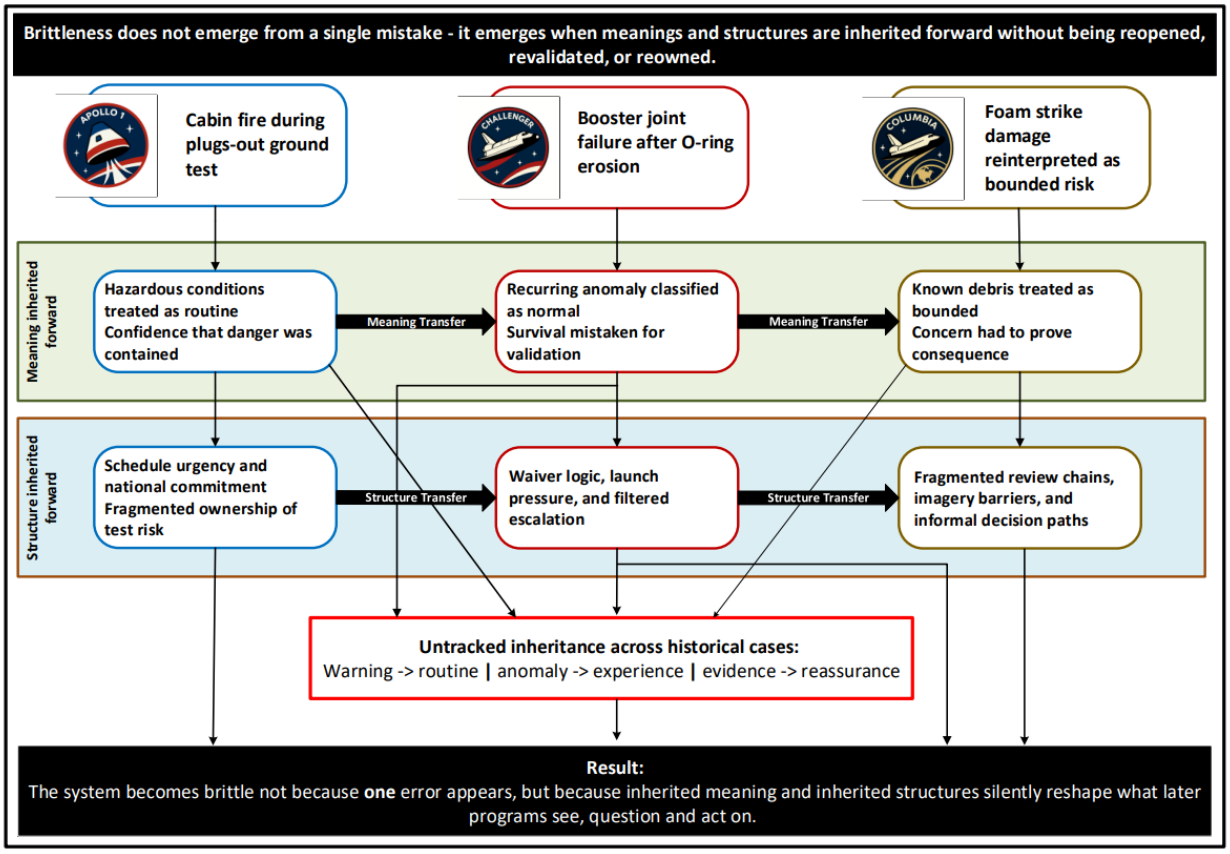


Figure 12 : Untracked Inheritance of Meaning and Structure in Historical Spaceflight

The relevance of these historical patterns is not confined to past programs but extends to modern human-spaceflight efforts in which distributed development, anomaly interpretation, and verification confidence still shape risk understanding. The systemic behaviors traced through *Apollo* 1, *Challenger*, and *Columbia*—assumption hardening, verification drift, fragmentation of accountability, and the difficulty of challenging inherited interpretations—remain active features of today's architectures, even as the technical and organizational landscape has evolved. These patterns have taken on new forms within commercial partnerships, multi-program integration

environments, and verification chains spread across contractors, providers, and mission directorates. The most recent and instructive example is the Boeing CST-100 *Starliner* Crewed Flight Test, where real-time anomaly management and verification interpretation exposed how assumption-driven risk manifests in a modern, distributed system. Section 7 therefore turns from historical precedent to contemporary practice, using *Starliner* as a case study to examine how assumption behavior operates within the architectures that now defines human spaceflight[203].

7 Case Study: *Starliner* and Verification Interpretation in Modern Human Spaceflight

Starliner is a useful modern case study as it shows how a program can accumulate extensive verification activity, formal review structures, and corrective action processes while still struggling to establish integrated confidence in human rated mission readiness. NASA's Commercial Crew Program (CCP) was meant to certify Boeing 's transportation system could safely carry crews to and from the ISS, and *Starliner's* 2024 Crew Flight Test was explicitly framed as a readiness demonstration for long duration rotational service[204]. Yet the road from first uncrewed Orbital Flight Test (OFT) in 2019 to crewed test in 2014 was marked by software anomalies, communications problems[205], a major valve issue that delayed OFT-2 in 2021[206], continuing qualification and certification work, and then propulsion system anomalies during the crewed mission itself[207].

The *Starliner* case produces more than a catalog of specific revelatory defects. It exposes a deeper systems problem: in modern human spaceflight, the presence of verification artifacts does not by itself answer the harder question of whether the total body of evidence supports justified confidence in the spacecraft as an integrated operational system. NASA's 2020 post-OFT review by the Problem Investigation Team (PIT) identified more than coding errors; it also reopened hazard reports, reassessed software verification plans, and revisited subsystem interfaces[208], indicating that the problem expanded beyond isolated bug fixes to the adequacy of the actual verification itself. The same pattern reappeared during the 2024 crewed test, when NASA ultimately decided that the uncertainty surrounding *Starliner's* propulsion performance was too high to accept additional risk for crew return[209].

Starliner, in this paper, is best understood as a case of verification interpretation under distributed technical and organizational conditions. The key issue is not that verification was absent, but that evidence was generated across software teams, hardware qualification streams, contractor

[203] U.S. Government Accountability Office, *Commercial Crew Program: Schedule Uncertainty* (GAO-19-504, 2019); NASA *Starliner* OFT and CFT Investigation Reports (NASA, 2020; 2026h).
[204] U.S. Government Accountability Office, Commercial Crew Program: Schedule Uncertainty (GAO-19-504, 2019); NASA, *Starliner* Crewed Flight Test Investigation Report (2026c); NASA *Starliner* program releases and OFT/CFT mission pages (NASA, 2020–2026).
[205] NASA, "NASA Shares Initial Findings from Boeing *Starliner* Orbital Flight Test," *NASA News*, April 4, 2025
[206] NASA, *Starliner Propulsion System Anomalies during the Crewed Flight Test – Investigation Report*, February 5, 2026
[207] NASA, "NASA Releases Report on *Starliner* Crewed Flight Test Investigation," *NASA News Release*, February 19, 2026
[208] NASA, "NASA Update on Orbital Test Flight Review Team," *NASA News Release*, November 2020
[209] NASA, *FAQ: NASA's Boeing Crew Flight Test Return Status*, NASA

processes, NASA oversight channels[210], and evolving mission contexts, while the meaning of that evidence at whole-system levels remained contested or incomplete. *Starliner* illustrates a central claim of this paper: assumptions about integration, representation, and readiness can persist beneath a program's formal verification record, and those assumptions may only become visible when the system is forced to operate in its real mission environment.

7.1 Overview of the *Starliner* Investigation

The *Starliner* investigation did not begin with the 2024 crewed test. Its modern history begins with the December 2019 uncrewed Orbital Test Flight, during which NASA and Boeing identified a Mission Elapsed Timer Anomaly (ETA), a service module disposal sequence software problem, and an intermittent space-to-ground forward link issue that affected the team's ability to command and control the vehicle[211]. NASA's subsequent update on the independent review team reported 61 corrective and preventative actions for the two software anomalies alone, while also describing NASA's own decision to reassess hazard-report verifications, review software verification plans, audit test scripts, and revisit interface control documentation to ensure hazards were fully defined, controlled and verified[212]. Those actions show that the early *Starliner* problems were never basic "software bugs" in a narrow sense; they immediately raised questions about verification coverage, system integration, and NASA's own depth of insight into contractor processes[213].

The next major disruption came in August 2021, when unexpected valve-position indications in *Starliner's* propulsion system halted the planned launch of OFT-2 and forced NASA and Boeing back into factory troubleshooting and root-cause work. When OFT-2 finally flew in May 2022, NASA described it as an end-to-end test from launch to docking, re-entry, and landing, intended to generate the data needed for certification and to prove that improvements after OFT-1 were sufficient to support a later crewed test. OFT-2 was not just a repeat flight, but a recovery effort meant to rebuild the evidentiary bridge between local fixes and system-level confidence. Then came the 2024 Crewed Test Flight.

On June 5th, 2024, the Boeing CST-100 *Starliner* - dubbed *Calypso* - embarked on its maiden Crewed Flight Test (CFT) atop an Atlas V rocket with veteran astronauts Barry "Butch" Wilmore and Sunita "Suni" Williams on what was planned to be an 8-14 day mission[214]. On June 6th, 2024, during docking operations, five Service Module (SM) RCS thrusters triggered failure detection logic and shut down, temporarily eliminating CFT's full six-degree-of-freedom control (6DOF). Flight controllers successfully restored four jets in-flight, allowing docking to proceed[215]. This, however, is not where

[210] Aerospace Safety Advisory Panel, ASAP Annual Report 2025 (2026), workforce, technical-authority, acquisition, and integrated-risk sections.
[211] NASA, "NASA Shares Initial Findings from Boeing *Starliner* Orbital Flight Test," *NASA News*, April 4, 2025; NASA, *What You Need to Know About NASA's Boeing Orbital Flight Test-2* (NASA Human Spaceflight, May 16, 2022).
[212] NASA, *NASA Update on Orbital Test Flight Review Team* (NASA News Release, Nov. 2020).
[213] NASA Aerospace Safety Advisory Panel, *Aerospace Safety Advisory Panel 2024 Annual Report*, January 25, 2025
[214] NASA, *NASA's Boeing Crew Flight Test*, accessed April 2, 2026
[215] Marcia Smith, "Boeing's *Starliner* CFT Docks with ISS Despite Thruster Issues," *SpacePolicyOnline.com*, June 6, 2024

the issues end as there were additional anomalies ranging from helium system leaks and redundant component failures in-flight[216].

On September 7th, 2024, during reentry maneuvers, one Crew Module (CM) RCS thruster failed[217]. This reduced fault tolerance to zero, compromising redundancy during critical phases. CFT remained in orbit for 93 days, 13 hours, and 9 minutes, from launch on June 5, 2024, until its autonomous landing at White Sands on September 7, 2024[218]. CFT landed safely at White Sands Space Harbor sans astronauts[219]. The extended mission concluded with a successful, albeit crewless, landing. The astronauts spent 286 days in space before eventually returning to Earth aboard SpaceX Crew9 Dragon on March 18, 2025[220].

7.1.1 Verification Without Confidence

The 2024 Crew Flight Test forced the program into its most consequential verification and risk decision yet. According to NASA, some of *CFT's* thrusters "did not perform as expected," several helium leaks were observed, and the agency ultimately decided not to accept more risk than necessary for Butch and Suni's return[221]. ASAP described the mission as one of the most complex and challenging crew-safety decision NASA had faced in years. In its account, *CFT's* service module reaction control system experienced multiple failures associated with helium-manifold leaks and overheated thrusters during approach to station[222]. After weeks of evaluation, the risk of subsequent thruster failures during undock-through-deorbit could not be quantified to a level NASA considered acceptable for crew return.

The formal investigative picture sharpened in 2025 and 2026. NASA announced in February 2026 that, after *CFT's* 93 day mission, the PIT had examined the "technical, organizational, and cultural contributors" to the crewed test issues. They, unsurprisingly, found an interplay of combined hardware failures, qualification gaps, leadership missteps, and cultural breakdowns inconsistent with NASA's human spaceflight safety standards[223]. NASA further classified the flight as a Type A mishap[224] because the loss of maneuverability near station created the potential for a significant mishap despite the absence of injury. Read together, the post-2019 and post-2024 investigations suggest that the programs' central challenge was not just defect discovery, it was the recurring difficulty of determining whether the existing body of verification evidence really justified confidence in *Starliner* as an integrated human spaceflight system.

---

[216] NASA, *Starliner Propulsion System Anomalies during the Crewed Flight Test – Investigation Report*, February 5, 2026.
[217] Marcia Smith, "*Starliner*'s Crew Flight Test Comes to an End," *SpacePolicyOnline.com*, September 7, 2024
[218] NASA, "NASA, Boeing Welcome *Starliner* Spacecraft to Earth, Close Mission," press release 24-113, September 7, 2024
[219] Jessica Taveau, "Welcome Home! NASA's SpaceX Crew9 Back on Earth After Science Mission," *NASA News Release*, March 18, 2025
[220] Sawyer Rosenstein, "Crew9, *Starliner* CFT Astronauts Return to Earth aboard Dragon," *NASA'SpaceFlight.com*, March 18, 2025
[221] NASA, *Starliner Propulsion System Anomalies during the Crewed Flight Test – Investigation Report* (Feb. 5, 2026)
[222] Aerospace Safety Advisory Panel, ASAP Annual Report 2025 (2026), workforce, technical authority, acquisition, and *Artemis* integrated-risk sections.
[223] NASA, NASA-STD-3001, Volume 2: Human Factors, Habitability, and Environmental Health, Rev. D (2025), pp. 49.
[224] Marcia Smith, "NASA Classifies *Starliner* CFT as Type-A Mishap, Leadership Changes Coming," *SpacePolicyOnline.com*, Feb. 19, 2026

CFT exemplifies a problem that is increasingly important in modern human spaceflight: a program can accumulate substantial verification activity without achieving equivalent confidence in integrated mission readiness. The central lesson of the *Starliner* spacecraft cannot be reduced to the anomalies themselves, but to the fact that the presence of reviews, tests, corrective actions, and certification artifacts did not, on their own, resolve the more fundamental question of whether the spacecraft was sufficiently understood to fly crew under real operational conditions. Verification was present but confidence remained conditional[225].

This is a distinction that is informative because formal verification and confidence are not the same thing. Verification, in its procedural form, asks whether requirements have corresponding evidence through analysis, test, inspection or demonstration. Confidence asks something broader and more demanding: whether the total body of evidence is sufficient to support belief that the vehicle will behave safely and predictably in the mission environment for which it is being certified. *CFT* makes this gap visible.

A program may appear strong in verification while remaining fragile in confidence.

The post OFT-1 response is especially revealing because unearths that NASA did not treat the 2019 anomalies as isolated software defects that could be cleanly corrected and closed[226]. Instead, the agency reopened hazard reports where necessary, reassessed software verification plans, audited test scripts, and revisited interface control documentation[227]. That response suggests that the real concern was not the presence of coding errors, but the possibility that the existing verification structure had failed to capture important time-dependent interactions, sequencing logic, and cross-subsystem behaviors in a mission representative way. The issue was therefore epistemic as much as technical, and the anomalies called into question what the existing body of evidence actually justified the program in claiming to know.

The same pattern reappeared with CFT. NASA's decision to return *CFT* uncrewed rested on more than a simple finding that the spacecraft had categorically failed[228]. Rather, it reflected unresolved uncertainty about propulsion system behavior, failure tolerance, and risk return. The key point is that the uncertainty remained significant enough that NASA was unwilling to treat existing analysis and test history as an adequate basis for crew return. That is the essence of verification without confidence[229]. The program possessed evidence, but not enough integrated understanding to justify accepting additional risk to the astronauts.

---

[225] NASA, *NASA-STD-3001, Volume 2: Human Factors, Habitability, and Environmental Health*, Rev. D (2025), pp. 155,236-237,241,261,313,346.
[226] NASA, "NASA Shares Initial Findings from Boeing *Starliner* Orbital Flight Test," April 4, 2025.
[227] NASA, "NASA Update on Orbital Test Flight Review Team," November 2020.
[228] NASA, *Starliner Propulsion System Anomalies during the Crewed Flight Test – Investigation Report*, February 5, 2026; Aerospace Safety Advisory Panel, *2025 Annual Report* (Jan. 2026), pp. 8, 13, 15, 18–24, 47.
[229] Aerospace Safety Advisory Panel (ASAP), *2025 Annual Report*, February 15, 2026.

This makes the *Starliner* spacecraft especially relevant to assumption management. Verification regimes are never assumption free – they depend on judgements about what must be tested, which conditions are representative, how interfaces are bounded, how anomalies should be interpreted, and when evidence is sufficient to support closure. When those judgements remain implicit, verification can become satisfied within a local context, while the assumptions required to translate the local result into whole system confidence remain weakly examined. Under those risk conditions, the program carries not only technical risk, but interpretive risk.

Starliner therefore shows that confidence is not just about the sum of its completed verification products, it is a judgement about the adequacy, representativeness, and meaning of the evidence as a whole. This judgement becomes especially difficult in human spaceflight. Where software timing, propulsion interactions, operational contingencies, and cross-organizational interfaces can determine mission outcome more decisively than any single requirement closure. The lesson is that verification must be treated not only as a documentation process, but as an evidentiary argument that remains open to challenge when live system behavior reveals gaps in understanding.

For this paper, that is the real significance of *Starliner*: **it demonstrates that modern spaceflight programs can become procedurally mature while still lacking integrated confidence in readiness.** The danger is not only that defects may survive into flight, but that programs may confuse completion of verification work with the achievement of justified belief. Once that confusion takes hold, assumptions about readiness, representativeness, and acceptable uncertainty can remain hidden beneath an apparently complete certification record. The PIT findings disclose just how costly that distinction can become[230].

7.2 Integration Assumptions Across Program Boundaries

Starliner demonstrates that verification problems in modern human spaceflight are rarely confined to a single subsystem. They emerge instead at the interfaces between contractor work, NASA oversight, certification logic, mission management, and operational decision making. For that reason, one of the most important lessons from the *CFT* mission is that while the anomalies occurred the program **still** depended on a chain of integration assumptions spanning organizations as well as hardware and software[231]. These assumptions shaped how local evidence was interpreted, how much independent visibility NASA possessed into contractor work, and how confidently the agency could translate subsystem level findings into agency-level claims of readiness.

A second lesson from *Starliner* concerns the way integration assumptions propagate across organizational boundaries. The CCP was structured around contractor-led development with NASA providing insight, oversight, and certification authority. The arrangement can combine innovation with rigorous government standards, but it also creates a verification environment where evidence, responsibilities, and decisions are distributed across Boeing, NASA program offices, review boards, technical authorities, mission managers, and ISS operations. In such a distributed environment,

[230] NASA, "NASA Releases Report on *Starliner* Crewed Flight Test Investigation," February 19, 2026
[231] Aerospace Safety Advisory Panel (ASAP), 2025 Annual Report, February 15, 2026.

integration confidence depends heavily on assumptions about who owns which risk, how local technical concerns are elevated, and whether the evidentiary basis produced inside one organizational boundary is sufficient for confidence at a broader system level[232].

Integration assumptions about failure tolerance, operational margins, waiver justification, and the meaning of prior qualification data become especially significant when the existing evidence base is not strong enough to support them. The subsections below show how cross-boundary verification challenges were not issues of documentation alone, but recurring interpretation problems in which the adequacy of one body of evidence depended on how it connected to evidence generated elsewhere.

7.2.1 Qualification and Test Coverage Assumptions

Starliner's qualification program rested on several unverified assumptions about how hardware would behave under mission-representative conditions[233].

- Qualification testing lacked fully enveloping thermal, dynamic, and duty-cycle environments.
- Integrated propulsion behavior was not assessed under realistic mission loads.
- Several subsystem interactions were never exercised in ground test configurations.
- Gaps in qualification led to reliance on engineering rationale rather than demonstrated performance.

7.2.2 Data, Telemetry, and Visibility Limitations

Limited telemetry and data-collection capabilities constrained NASA's ability to independently evaluate system behavior and verify contractor claims.

- Low sample-rate telemetry obscured thruster dynamics and masked early signatures of failure.
- Lack of onboard high-frequency data storage prevented full reconstruction of anomalies on OFT-1, OFT-2, and CFT.
- Limited data prevented NASA from validating subsystem-level performance assumptions.
- Verification decisions often relied on Boeing's assessments instead of direct measurement.

7.2.3 Acceptance of Resolved Anomalies

Several unexplained anomalies were carried forward across missions without confirmed root-cause resolution, creating implicit assumptions about acceptable risk.

- Unexplained anomalies (UAs) persisted from OFT-1 and OFT-2 into CFT.
- Engineering judgement replaced root-cause determination in multiple cases.
- Systemic issues were allowed to migrate across flights without corrective closure.
- Recurring propulsion issues demonstrated that anomaly acceptance created accumulating risk.

---

[232] NASA Office of Inspector General, *NASA's Management of the Human Landing System Contracts*, IG-26-004 (Mar. 10, 2026), p. i, pp. 12–16; Aerospace Safety Advisory Panel, *2025 Annual Report* (Jan. 2026), pp. 18–24.
[233] NASA Office of Inspector General, *NASA's Management of the Human Landing System Contracts*, IG-26-004 (Mar. 10, 2026), pp. i–ii, 12–16, 29–35.

7.2.4 Oversight and Insight Constraints

NASA's ability to independently verify readiness was limited by data restrictions and subcontractor opacity.

- NASA lacked access to detailed subcontractor-level qualification data.
- Insight into component-level test results and failure modes was incomplete.
- Limited access restricted NASA's ability to perform independent verification.
- Program readiness relied on contractor-provided evidence rather than independent evaluation.

7.2.5 Long Term Hardware and Infrastructure Assumptions

Several architectural assumptions shaped the program but were never explicitly treated as integration risks.

- Limited availability of spares created sustainment and lifecycle risks.
- Long lead components constrained the ability to recover from failures.
- Dependence on the Atlas V launch vehicle introduced architectural exposure as retirement approached.
- Program plans assumed long-term hardware continuity that was not guaranteed.

7.3 Distributed Accountability, Technical Authority, and Risk Ownership

Starliner also demonstrates that verification problems in modern human spaceflight are not only technical, but also organizational. In distributed development environments, evidence is generated across multiple teams, disciplines, review boards and institutions, but accountability for interpreting that evidence may remain diffuse[234]. This creates a structural vulnerability in which everyone contributes to assurance, yet no single actor fully owns the burden of doubt. *Starliner* provides valuable insight into how that condition can weaken confidence even when formal oversight mechanisms exist.

Risk ownership is equally important since they may be identified by one team, modeled by another, dispositioned in a third forum, and operationally borne by a fourth. That fragmentation can be manageable only if responsibility for final interpretation remains explicit. *Starliner* suggests that one of the most consequential forms of assumption debt arises when risk is broadly shared but insufficiently owned. Under those conditions, technical findings can circulate through the system without forcing a decisive answer to a basic question: *who is obligated to preserve doubt until the evidence is truly sufficient?* Without a clearly empowered answer to that question, the architecture may drift toward collective reassurance rather than disciplined uncertainty management.

Artemis and other campaign based architectures will involve even more distributed evidence, interfaces, mixed authorities and organizational handoffs than commercial crew. If *Starliner* revealed that governance assumptions can become mission significant as hardware assumptions, then future

[234] NASA, *NASA-STD-3001, Volume 2: Human Factors, Habitability, and Environmental Health*, Rev. D (2025), pp. 155,236-237,241,261,313,346.

exploration systems must treat role clarity, technical authority, and risk ownership as first-order elements of assumption management[235]. Otherwise, cross-program assurance may appear coherent until a live anomaly reveals that no one fully owned the right to stop, challenge, or reinterpret the path to readiness.

This sets the stage for examining how ambiguity in governance and decision authority can harden unnoticed and compromise system safety.

7.3.1 Governance and Decision-Authority Ambiguity

The program structure distributed authority across multiple organizations, resulting in unclear decision pathways and inconsistent application of technical requirements.

- Roles and responsibilities were not consistently defined across CCP, ISSP, and Boeing.
- Escalation authority for technical issues was ambiguous.
- Governance fragmentation complicated risk disposition and issue closure.

7.3.2 Trust, Transparency, and Inter-Organizational Dynamics

Working relationships across organizations were affected by inconsistent transparency and strained communication channels.

- Selective data sharing reduced trust between NASA and Boeing.
- NASA teams reported exclusion from critical discussions.
- CCP teams experienced information saturation and competing inputs.
- Transparency gaps limited the effectiveness of joint risk assessments.

7.3.3 Leadership Posture and Risk Tolerance

Leadership across NASA and Boeing adopted risk postures that influenced how evidence was interpreted and how concerns were elevated.

- Leadership was perceived as more risk-tolerant than expected for crewed missions.
- Dissenting technical views were often discounted or minimized.
- Readiness assessments were shaped by optimistic interpretations of limited evidence.
- The resulting posture weakened the authority of independent technical review.

7.3.4 Communication and Team Effectiveness

Organizational and communication inefficiencies limited coordination across technical teams.

- Meeting structures were inconsistent and rapidly changing.
- Cross-disciplinary communication paths lacked clarity.
- Team surveys indicated low confidence in organizational structure and decision processes.

---

[235] Aerospace Safety Advisory Panel, *ASAP Annual Report 2025* (2026), governance, technical authority, and organizational-root-cause sections.

- Ineffective coordination contributed to delays in identifying and resolving anomalies.

7.3.5 Cultural Misalignment Between NASA and Boeing

Differences in organizational culture affected expectations, verification rigor, and interpretations of risk.

- NASA's documentation-driven, conservative approach conflicted with Boeing's provider-autonomy model.
- Differing expectations for rigor led to disagreements on anomaly significance.
- Cultural divergence contributed to inconsistent application of standards and processes.

7.3.6 Organizational Root Causes of CFT Anomalies

The PIT identified several organizational contributors that collectively weakened program integrity.

- NASA's hands-off contracting approach reduced visibility into vehicle development.
- Boeing's systems engineering oversight was insufficient, particularly for subcontractors.
- CCP increasingly emphasized program continuity and schedule expectations over technical rigor.
- Combined effects allowed systemic defects to propagate into CFT.

7.3.7 Mishap Classification and Implications for Accountability

The CFT anomaly met multiple criteria for a high-severity mishap, raising questions about how risk was owned and interpreted across the program.

- The loss of 6DOF control met the requirements for a Type A mishap under NPR 8621.1D.
- Some argued the event qualified as a High Visibility Close Call due to successful recovery.
- The PIT recommended retroactive Type A classification to reflect the severity of contributing factors.
- Divergent interpretations highlighted ambiguity in risk ownership and authority.

7.4 Lessons for Complex Exploration Systems

The *Starliner* case points to several lessons that extend well beyond one spacecraft. First, the exploration systems should distinguish far more explicitly between requirement closure and readiness confidence. NASA's responses after OFT-1 and CFT show that completed reviews and accumulated artifacts do not end the problem; the agency still had to revisit hazard verifications, interfaces, test adequacy, and operational risk because the existing record did not fully answer the integrated safety question. Future campaign architectures should therefore treat verification as an evidentiary argument that must remain open to reinterpretation when anomalies reveal that the system was understood less completely than previously believed.

Second, integration assumptions must be surfaced and governed as first order risks. *Starliner*'s history demonstrates that assumptions about sequencing logic, communications behavior, propulsion system failure tolerance, interface ownership, and the translation of local evidence into

whole system certification can become mission significant. In campaign architectures those assumptions will be even more distributed because the system-of-systems boundary is wider, heterogeneous, and the number of organizational handoffs is larger. What *Starliner* shows in one provider relationship can appear in *Artemis*-scale form across vehicles, ground systems, commercial services, logistic chains and mission operations.

Third, technical authority and role clarity are not bureaucratic add-ons, they are part of the safety architecture. ASAP's 2024 and 2025 reports repeatedly return to the importance of role clarity, independent technical authorities, and the need to resist allowing "on paper" progress to stand in for actual risk reduction. The lesson itself is directly applicable to exploration campaigns, where schedule pressure and architecture dependency can create strong incentives to treat unresolved uncertainties as manageable because a milestone must be preserved. *Starliner* suggests that independent technical challenge is most valuable precisely when the program is under pressure to convert ambiguous evidence into reassuring narratives.

Fourth, operational flexibility is a crucial safety resource. NASA's ability to keep the *CFT* crew aboard ISS and return them later on Crew-9 was not incidental, it was a structural buffer. It allowed the agency to choose a lower-risk path when confidence in *CFT's* return performance proved inadequate. The lesson for exploration campaigns is that architectures should not be judged only by its nominal path but if they preserve off-ramps, alternate returns, contingency logistics, and the time needed to think clearly under uncertainty[236].

A system that has no safe option except to trust its original assumptions is a brittle system, even if it looks efficient on paper.

Finally, *Starliner* reinforces the broader thesis of this paper. The most consequential risk in modern human spaceflight may not be the absence of process but the quiet accumulation of weakly owned assumptions beneath apparently mature process. NASA's 2026 investigation tied the crew-test issues to combined hardware failures, qualification gaps, leadership missteps, and cultural breakdowns, while ASAP's 2025 report warned that milestone progress does not necessarily track technical risk. Together, these findings suggest that complex exploration programs must become better at asking not only whether evidence exists, but what that evidence actually warrants the program to believe.

Artemis will depend on far more interfaces, providers, logistics chains, and mixed authorities than commercial crew programs, and *Starliner* is not just a commercial crew case study. It is an early warning for any campaign architecture that confuses the completion of verification work with the achievement of justified confidence.

We may not be so lucky next time.

[236] Aerospace Safety Advisory Panel, ASAP Annual Report 2025 (2026), *Artemis* and Commercial Crew integration-risk sections.

8 The *Artemis* Exploration Campaign

The *Artemis* Exploration Campaign underwent a significant transformations in February and March 2026, when Administrator Isaacman announced a major overhauls from the previous architecture. This transformation introduced some much needed chaos, turbulence, momentum, and challenged assumptions embedded in the earlier mission sequence.

Credit where credit is due to Administrator Isaacman for moving NASA forward, but momentum should not be confused with maturity. A future rewritten, *Artemis's* future, is not necessarily a future that has been fully reasoned through. Especially when mission cadence, infrastructure readiness, workforce capacity, crew survival logic, and most importantly budget must all align simultaneously for this to work.

Early *Artemis* planning centered on a "single landing demonstration sequence" in which *Artemis* III would accomplish the first lunar landing, followed by Gateway assembly, logistics buildup, and eventual progress toward a sustained lunar presence. This design depended on a tightly coupled sequence—SLS, Orion, HLS, Gateway, mobility systems, and surface infrastructure—where progress in one element set the pacing for the entire campaign. Table 8 highlights how the updated architecture departs from that earlier model.

Table 8 : Comparison of Original vs. Updated Artemis Architecture

Feature	***Original Artemis Architecture***	***Updated Artemis Architecture***
Artemis III	First lunar landing	LEO/cislunar systems-integration mission (landing removed)
First Landing	*Artemis* III	*Artemis* IV
Mission Cadence	~1 mission every 2–3 years	Target ~1 mission per year
Gateway Role	Early staging node; central to architecture	Reduced role early; deferred to later campaign phases
Key Testing Approach	Lunar landing validates integrated stack	New orbital demonstration validates interfaces before landing
Program Logic	Demonstration sequence	Stepwise operational campaign
Element Dependencies	Strong sequencing dependence (Gateway → HLS → landing)	More flexible, earlier testing of interfaces in Earth orbit
SLS Configuration	Planned transition to Block 1B early	Block 1 retained longer; ML-2 reconsidered

Artemis III no longer performs the first lunar landing and instead becomes a LEO and cislunar systems-integration mission[237]. A new orbital demonstration precedes the landing, intended to validate critical interfaces such as Orion–HLS docking, xEVAS integration, and cryogenic-propulsion operations. The first landing now shifts to *Artemis* IV, with later missions targeting an annual cadence focused on repeatability and operational learning. Gateway is no longer the early staging node; its

[237] Aerospace Safety Advisory Panel, ASAP Annual Report 2025 (2026); NASA, *Artemis* III Mission Page / Preliminary *Artemis* III Mission Plans (2026).

role moves later in the campaign - maybe, with early operations emphasizing Earth-orbit testing of the most safety-critical interfaces rather than lunar-orbit integration.

This is more than a rearrangement of mission numbers; it is a restructuring of the underlying exploration logic. The program is moving from a sparse demonstration driven sequence to a stepwise operational campaign with standardized elements and a higher tempo, but not without implications. A key architectural implication is that the burden of assumption stability shifts:

Verification boundaries change, sequencing relationships change, and the acceptable order of operations changes.

These changes also propagate across lifecycle decision gates:

- **Gate 1 (architectural):** NASA is reframing capability progression, risk posture, and policy-driven timelines.
- **Gate 2 (verification):** *Artemis* III's reframing forces a reevaluation of what is actually being verified before the first landing and what evidence is still deferred
- **Gate 3 (operational):** The shift from a 2027 landing to a systems-integration mission alters the nature of crew and ground operations planning, exposing how assumptions hardened around the prior landing sequence must now be explicitly reinterpreted.

This point is critical because ***Artemis* is not a single-vehicle program**. NASA's Moon-to-Mars architecture is organized around campaign segments composed of multiple architectures and elements that evolve over time. NASA describes the architecture as spanning human lunar return, foundational exploration, sustained lunar evolution, and humans-to-Mars, with systems such as transportation, communications, habitation, logistics, mobility, power, and infrastructure mapped across multiple uses and mission phases. Some of these elements—including SLS and Orion—predate the current architecture-definition effort, while others emerge from newly identified capability gaps.

This structural layering makes *Artemis* highly sensitive to reinterpretation. When the mission sequence changes, the meaning of the assumptions inherited from earlier phases or architectures changes with it. The architecture does not reset cleanly; earlier assumptions persist unless they are deliberately reevaluated. As a result, any shift in campaign design forces a corresponding reconsideration of what the program believes to be true about system maturity, interfaces, dependencies, and readiness.

The architecture reset is more than just a program update; it is a live case of assumption management[238]. By changing mission sequence, staging logic, cadence expectations, and the role of

[238] NASA, *NPR 8705.2C, Human-Rating Requirements for Space Systems* (updated Jan. 19, 2024), pp. 4–6, 15, 41; NASA Office of Inspector General, *NASA's Management of the Human Landing System Contracts*, IG-26-004 (Mar. 10, 2026), pp. 29–35; NASA, *Moon Base Architecture User's Guide* (2026), p. 11; NASA Human System Risk Board, *Risk of Inadequate Mission Rescue (RIRM): Human Spaceflight Risk Technical Report* (2024).

key systems, the reset altered the conditions under which prior *Artemis* assumptions had been formed. The question then is not **"was the earlier architecture right or wrong in isolation?"**, but **if the assumptions carried forward from that architecture remain valid under the revised campaign.** Section 8.1 examines the *Artemis* Architecture Reset and Ignition as a real-time example of how architectural change can expose, challenge and redistribute assumptions across a complex human spaceflight campaign.

8.1 *Artemis* Architecture Revision and Ignition: A Live Case of Assumption Management

The February 2026 *Artemis* Architecture Revision (AR1) within this section is being treated as a scheduling update but as a **live case of assumption management** across the *Artemis* campaign. The revised mission sequence changed what *Artemis* III is expected to prove, shifted the first lunar landing to *Artemis* IV, altered the near-term role of commercial landers, and restructured assumptions about SLS configuration and annual cadence. These decisions affect architecture, verification, integration operations, and crew survival logic simultaneously.

Ignition, introduced in March 2026, also provides a retrospective analytical lens for understanding how the February architecture revision reshaped the assumption landscape. Although it did not exist at the time AR1 was announced, Ignition offers a structured way to examine how the revised mission sequence exposed or altered underlying assumptions across systems, interfaces, and mission logic. By making these assumptions explicit, traceable, and reviewable, Ignition allows the campaign to evaluate which pre-revision assumptions remain valid, which require modification, and which have been overtaken by the new architectural reality. In this sense, Ignition becomes the framework through which the effects of the February reset can be systematically interpreted and managed.

Against that backdrop, the consequences of AR1 become clearer when examining how it reshaped expectations for *Artemis* III. Prior to AR1, *Artemis* III carried both the symbolic and technical burden of becoming the first *Artemis* lunar landing. After the revision, *Artemis* III became a LEO rendezvous and docking mission, designed to test integrated operations between Orion and one or both HLS providers. This reframes *Artemis* III from a landing milestone into an integrated proof-of-operations step. It also indicates that assumptions about lander readiness, docking behavior, crew transfer, mission sequencing, and provider maturity have been redistributed rather than removed.

These are exactly the kinds of conditions in which assumption drift can emerge.

An assumption formed under one architecture can remain present in documents, schedules, or leadership language even after the architecture has changed around it. The risk is not that the revision occurred; revisions are necessary. The risk is that the campaign continues carrying earlier assumptions without explicitly identifying **which ones remain valid, which require amendment, and which have been overtaken by the new mission sequence**. The revision then becomes a test of whether *Artemis* can rebaseline its assumptions as deliberately as it rebaselines its mission plan.

The successful completion of *Artemis* II adds another layer of analysis to this issue, providing fresh evidence that must now be interpreted within the revised architecture and the assumptions

highlighted through Ignition. Initial post-flight assessments found that Orion's thermal protection system performed as expected, SLS met its test-flight objectives, EGS sustained minimal damage, and NASA began analyzing a urine-vent issue to implement corrective action before *Artemis* III. This is significant because the *Artemis* campaign is now integrating live crewed-flight evidence into a reinterpreted campaign architecture at the same time that it is attempting to accelerate cadence.

The key lesson is that the revision is not a clean break from the previous *Artemis* architecture, but a reinterpretation of it. SLS, Orion, HLS, Gateway, xEVAS, surface power systems, communications, and mission operations all carry historical constraints, inherited assumptions, and embedded logic. The revision changes how these elements are arranged, but it does not erase the assumptions built into them.

Thus, assumption management becomes the discipline of determining what the new architecture is allowed to inherit.

8.2 Cross Program Interface Assumption

Within *Artemis*, several program elements do more than provide isolated capabilities—**they generate architectural expectations** that shape the broader campaign. These are *assumption-engine technologies*: platforms or governance pathways whose anticipated availability drives mission design, timelines, logistics, and risk posture well beyond their individual function.

The campaign depends on interfaces that traverse contractor, center, and partner boundaries. And these interfaces are **technical, organizational, contractual, and temporal** in nature. NASA's Moon-to-Mars architecture explicitly maps elements across multiple sub-architectures and campaign segments, meaning one element may support several functions—and one function may rely on multiple elements. This makes interface assumptions especially consequential, because they are where local progress meets campaign-level confidence.

An **interface assumption** is more than the expectation that two systems can connect. It includes assumptions about timing, configuration, authority, data, crew transfer methods, operational tempo, contingency response, and evidence sufficiency[239]. For example, if Orion is expected to rendezvous with a commercial lander, the assumption does not end at docking hardware compatibility. It includes navigation performance, software maturity, crew procedures, communications support, abort logic, training depth, readiness criteria, and the authority structure that determines whether the combined operation is safe to execute.

The Ignition reset has made these interface assumptions **visible and explicit**. NASA now describes *Artemis* III as a mission that will test rendezvous and docking between Orion and HLS in LEO before sending astronauts to the lunar surface. In this framing, interface proof becomes a campaign

[239] NASA *Systems Engineering Handbook* (NASA/SP-2016-6105 Rev. 2, 2016); NASA *Procedural Requirements 7123.1D — NASA Systems Engineering Processes and Requirements*, Ch. 2 and app. C; NASA *Artemis* III Mission Page / *Preliminary Artemis III Mission Plans*.

milestone rather than a background dependency. The interface is no longer a mechanism supporting a landing, it becomes the purpose of the mission.

This leads directly to the issue of **distributed proof burden**. The evidence supporting a campaign-critical interface is produced by different organizations and interpreted at different levels. A commercial provider generates evidence for its system. NASA provides insight, oversight, mission integration, and crew-safety judgment. Centers and integration offices evaluate cross-program compatibility. Mission operations assess procedural executability. None of these evidence streams alone closes the campaign-level claim. The assumption that the interface "works" must be assembled from **pieces of evidence produced across the architecture.**

This condition is not inherently problematic—distributed development is normal in aerospace. The issue arises when **distributed proof becomes distributed ambiguity**. If each organization can demonstrate acceptability within its own domain, but no entity owns the combined assumption, the campaign can become confident in the subsystems without ever establishing confidence in the integrated whole. Figure 6, the Distributed Proof Burden Map, is a great illustration example depicting that that *Artemis* is not lacking in proof activity—but it faces the harder problem of integrating the **meaning** of that proof across a distributed architecture.

This dynamic must be explicitly understood because **campaign-level assumptions propagate**. They influence habitation, mobility, logistics, communications, mission sequencing, surface operations, and risk posture. These dependencies can outpace demonstrated maturity or executable program reality. The issue is not whether a capability is valuable, but whether the campaign begins behaving **as if it is already stable and integrated** before those conditions exist. Three cases illustrate this:

1. **KRUSTY** as a technology-demonstration-driven assumption engine
2. **Gateway** as an infrastructure-driven assumption engine
3. **Nuclear regulatory lag** as a governance-driven assumption engine

Together, these examples show that assumptions travel not only through hardware and system design, but also through **schedule logic, policy constraints, and institutional processes**.

8.2.1 KRUSTY and the Propagation of Surface Power Assumptions

KRUSTY[240] fits this discussion as an enabling assumption engine because it sits at the boundary between demonstrated feasibility, implied future dependence, and occupies a unique position within the *Artemis* ecosystem:

[240] Gibson, M. A., et al. *Kilopower Reactor Using Stirling Technology (KRUSTY) Nuclear Ground Test Results and Lessons Learned*, NASA Technical Memorandum 2018-219941; NASA Human System Risk Board, *Risk of Inadequate Mission Rescue (RIRM): Human Spaceflight Risk Technical Report* (2024).

It is neither a flight system nor a purely abstract technology study. Instead, KRUSTY functions as an assumption engine — a non-flight demonstration whose outcomes are repeatedly invoked to justify architectural, safety, and operational decisions for future human lunar missions.

KRUSTY is a ground demonstration built around a 1 kWe Kilopower reactor using Stirling technology. NASA has broadly described Kilopower as a small space-fission power effort designed to scale into the one- to ten-kilowatt range. According to the agency, the project developed concepts and technologies that could support long-duration stays on planetary surfaces. KRUSTY represents an important step in the maturation of space fission concepts, but it did not itself constitute part of the additional lunar surface power architecture.

This is a critical distinction because NASA's later FSP effort explicitly builds on Kilopower and projects the concept forward into sustained lunar and Mars operations. The program now describes the system as a 40-kilowatt-class reactor intended to provide continuous power on the lunar surface by the early 2030s, presenting the capability as relevant to habitats, rovers, and future base-camp-style operations.

Once that larger architecture is envisioned — centered around sunlight-independent surface power — the underlying assumptions begin to propagate outward into landing-site selection, mission duration, **site-level power integration concepts**, mobility planning, and the credibility of long-duration surface operations.

Although KRUSTY was tested exclusively in ground-based, non-crew, non-ascent environments, its technical success is often treated as sufficient evidence to reduce the perceived risk associated with the far more complex surface fission systems envisioned for *Artemis*. This creates a subtle but consequential shift in the burden of proof: downstream programs may inherit confidence without inheriting the verification environments or constraints that originally justified that confidence.

Within complex human spaceflight systems, such assumption propagation can be particularly hazardous. Policy directives (SPD-6), safety frameworks (NSPM-20), technical standards (NSTM-3), and programmatic urgency can collectively elevate the results of a demonstrator into quasi-heritage status, even though the original system was never designed to meet human-rating, launch, or operational safety requirements[241].

For these reasons, KRUSTY should be understood not merely as a technological success, but as a driver of architectural expectation. Its significance lies in how a successful ground demonstration has encouraged planning efforts that assume future nuclear surface-power availability long before such a system has been fielded, validated, or demonstrated at mission scale on the Moon. In assumption-management terms, **the hazard does not stem from the technology itself, but from the gap between proof-of-concept and operational reliance** — a gap that becomes consequential

[241] NASA Office of Inspector General, *NASA's Management of the Human Landing System Contracts*, IG-26-004 (Mar. 10, 2026), pp. i–ii, 12–16, 29–35.

when strategy, mission design, and long-range exploration narratives begin treating a laboratory test as if it were an already-executable campaign condition.

8.2.2 Gateway and the Propagation of Staging Assumptions

KRUSTY serves not as a cautionary tale of failure, but as a case study in how successful demonstrations can quietly harden assumptions. Without explicit assumption stewardship, such successes risk narrowing the space for dissent, down-tiering safety scrutiny, and compressing verification pathways in future nuclear-enabled human exploration architectures.

Gateway, by contrast, functions as a different kind of assumption engine because its role is infrastructural rather than strictly technological[242]. Gateway has been described as central to *Artemis*—a multi-purpose lunar-orbit outpost intended to support lunar surface science, orbital operations, and future deep-space exploration. These descriptions, especially those used prior to Gateway's programmatic pause, are significant because they **implicitly** assigned Gateway a far larger supporting role than it holds today. It was framed as a campaign-shaping asset.

Once an architecture is designed around a platform understood in those terms, its expected availability begins to influence mission profiles, interface logic, and the way planners conceptualize sustained presence beyond a single sortie-style mission. Detailed Gateway documentation makes clear that the platform was conceived as a structurally central element of the campaign. The station was designed to provide docking ports for multiple visiting spacecraft, a habitable volume for crew, and the capability to stage and prepare for lunar surface missions. The Habitation and Logistics Outpost (HALO), was one of Gateway's core elements, is described by NASA as providing command and data handling, storage, electrical power distribution, thermal regulation, and communications and tracking. It acted as the structural and operational backbone of the initial Gateway configuration and enabled the platform to support rendezvous, docking, and staging operations for *Artemis* missions.

Reporting on *Artemis* IV shows how Gateway assumptions became operationally binding. Future missions were expected to incorporate Gateway directly into crewed mission flow, with Orion and the HLS interacting through the station and crew transfer occurring via Gateway— rather than through direct Orion-to-HLS docking.

This is what makes Gateway an unparalleled propagator of assumptions: **once the campaign is structured around it, Gateway's schedule, capabilities, interfaces, and continuity become embedded in the logic of the broader exploration architecture.** Delays, scope changes, governance shifts, or budget instability affecting Gateway do not remain localized to the outpost. They ripple outward into lunar mission sequencing, logistics planning, communications timing, and

[242] NASA Office of Inspector General, *NASA's Management of the Human Landing System Contracts*, IG-26-004 (Mar. 10, 2026), results brief pp. i–ii; Aerospace Safety Advisory Panel, *2025 Annual Report* (Jan. 2026), pp. 18–24, 47.

the overall viability of sustaining the *Artemis* concept because Gateway once shaped multiple architectural layers—mission sequence, logistics flow, interface design, and vehicle interoperability.

Gateway is more than a hardware element—it is a concentrator of campaign-level assumptions. And that is precisely why uncertainty around Gateway has consequences that extend far beyond the platform itself.

8.2.3 Regulatory Delays and Program Dependency

Assumption-driven governance is particularly significant in the nuclear domain. FSP does not become an architectural reality due to engineering concepts being promising or successful ground test demonstrations. NASA's own nuclear environmental guidance states that launch approval requirements under NSPM-20 guide the design of space fission systems, and that nuclear-enabled missions are subject to separate launch safety reviews, with authorization authority dependent on system characteristics and potential hazards. Moving from technical feasibility to campaign-level use requires more than engineering maturation; it requires safety analysis, environmental review, interagency coordination, and formal launch authorization.

The current FSP effort makes this point especially clear. NASA and the DOE now describe lunar FSP as a future capability intended to support sustained missions and continuous power on the Moon, with deployments targeted for the late 2020s or early 2030s. Yet the same policy frameworks that enable such missions also impose a deliberate and rigorous approval architecture governing whether, when, and under what conditions such systems can fly.

Federal Aviation Administration (FAA) guidance for launches involving space nuclear systems categorizes missions into tiers based on system characteristics, potential hazards, and national-security considerations. It ties launch, payload, environmental, and policy review to broader public-safety determinations, environmental-hazard assessments under the National Environmental Policy Act (NEPA), and national nuclear-interest evaluations. NASA briefing materials and NSPM-20 further note that high-risk or high-activity nuclear missions require independent assessment by the Interagency Nuclear Safety Review Board (INSRB) and, in the most demanding cases, presidential authorization.

For the *Artemis* campaign, the implication is that regulatory maturity can lag behind technical optimism. A campaign may begin speaking as though dependable lunar FSP is purely a matter of engineering completion, when in reality the capability must still pass through a layered approval process before it becomes executable mission infrastructure. This creates a second gap—distinct from the KRUSTY gap—within assumption management. It creates a gap between demonstration and deployment and between deployable hardware and authorized use.

Nuclear regulatory lag therefore belongs squarely in the architecture discussion, because it shapes whether a capability can truly be counted on, how much schedule confidence is justified, and what

downstream campaign plans are implicitly resting on—a capability that is technically imaginable, but not yet programmatically real or operationally authorized[243].

When KRUSTY, Gateway, and nuclear regulatory delays are all assigned into the larger *Artemis* context, it becomes clear that cross-program assumptions are produced by far more than hardware alone. Cross-program assumptions emerge when technology demonstrations, infrastructure plans, and governance processes begin operating as campaign-shaping commitments.

This is why assumption management in the *Artemis* campaign cannot limit its tracking to technical interfaces, it must also incorporate institutional mechanisms that determine when a capability is demonstrated, when it is integrated, and when it is actually executable.

8.3 Infrastructure Maturity, Surface Systems, and Fission Surface Power

Infrastructure is one of the strongest assumption carriers within the *Artemis* campaign because it quietly shapes expectations long before it is fully available. NASA's Moon-to-Mars architecture identifies **infrastructure support, logistics, mobility, communications and positioning, power, habitation, and transportation** as distinct sub architectures[244]. These are not decorative supplements to a landing sequence; they define **mission feasibility**, **crew endurance**, available **contingencies**, and the campaign's ability to transition from initial return to **sustained lunar presence**.

A key distinction must be maintained between **tactical maturity** and **campaign maturity**. A technology may be promising, partially demonstrated, or under accelerated development, while the campaign that depends on it still lacks a fully integrated operational basis to treat it as available. Once infrastructure is **assumed present within the architecture**, downstream systems begin designing against its **expected presence**, not its demonstrated readiness. This creates a dependency cascade:

- **Power** influences communications and habitation.
- **Communications** shape operations and contingency response.
- **Logistics** shape duration, consumables, and recovery margin.
- **Mobility** sets crew range and science return.
- **Surface support systems** shape what missions can credibly be considered mature.

FSP illustrates this issue clearly. NASA's Moon-to-Mars architecture treats power systems as a core sub-architecture that enables and conditions other systems. Recent national policy now directs NASA to initiate a program to develop a **mid-power fission reactor** effort within 30 days, with a lunar FSP variant ready for launch by 2030. This includes multiple vendors, preliminary design milestones, ground-test requirements, and attention to cost, schedule, and program objectives. This directive

[243] National Security and Technology Memorandum-3 (2026), §§2–3; NASA Fission Surface Power
[244] NASA *Systems Engineering Handbook* (NASA/SP-2016-6105 Rev. 2, 2016), §§2.4, 5.3–5.4, 6.2, 6.4, 6.5, 6.8; NASA Moon Base Users Guide.

accelerates the policy relevance of FSP but simultaneously increases the need to differentiate between **directed development**, **demonstrated capability**, and **campaign-ready integration**.

The **KRUSTY** demonstration is instructive. KRUSTY proved that compact fission power could operate under terrestrial test conditions and helped establish technical credibility for space nuclear power. But a terrestrial demonstration, a White House policy directive, and an operational lunar FSP capability exist at **different maturity levels** and come from **different institutional lineages**. Assumption management requires these levels to remain distinct. The campaign can responsibly build toward FSP without treating it as already integrated into **mission duration**, **crew survival**, **logistics**, or **surface operations**.

The Infrastructure Dependency Cascade, Figure 7, is a great resource to reflect on here. Its value is that it shows how an upstream infrastructure assumption branches into enabling capability areas—surface power, communications, logistics, staging, habitability, and mobility—and then into downstream mission consequences such as science operations, mission timelines, verification scope, and crew operations. The figure demonstrates why infrastructure assumptions are powerful: **they do not remain local; they propagate across multiple unrelated planning streams at once.**

This also means that **infrastructure maturity must be managed as a lifecycle issue**, not a late-integration task. If the assumed infrastructure shifts, the campaign must revisit the mission concepts, verification claims, operational procedures, and risk postures built upon it. Otherwise, infrastructure becomes a **hidden driver of campaign confidence**, allowing planning to advance while quietly deferring the burden of proof into later mission phases.

NASA's Moon Base User's Guide makes the maturity gap explicit. It states that Phase 1 Moon Base missions are intended to collect data and mature technologies needed for later phases, while architecture driven technology gaps represent the difference between available capabilities and desired future capabilities. This framing shows that the Moon Base is not a single deployed system awaiting use, it is a phased architecture on which later claims depend on closing known functional, technology, and data gaps.

This is directly relevant to FSP and other surface support systems. A future power capability may be strategically necessary for sustained operations, but strategic necessity is not the same thing as operational readiness. The guide's emphasis on phased demonstrations, shared infrastructure, and evolving gaps supports the distinction this paper draws between technical maturity and campaign maturity. A technology can be promising, prioritized, or policy-directed while the campaign still lacks the integrated evidence needed to treat it as a stable operational phase.

For this section, it should therefore be understood that the Moon Base Users Guide is used to frame Moon Base planning as a structured dependency network rather than as proof that sustained lunar architecture is already available. Surface power, communications, logistics, habitation, mobility, and cargo delivery are all part of the future capability stack. If those capabilities mature unevenly, the architecture must be able to identify which assumptions require revalidation before they are inherited by mission design, verification planning, and crew survival logic.

8.4 Gateway, Budget and Architectural Reinterpretation

The tension surrounding Gateway is real. NASA's FY 2026 request proposed canceling the Gateway program, but Congress subsequently enacted a significantly larger NASA budget along with a supplemental spending package, the FY 2026 "minibus". It bundled funding for Commerce-Justice-Science (funding NASA), Energy and Water (funding the DOE), and Interior-Environment. In doing so, Congress effectively linked, within a single appropriations vehicle, NASA's *Artemis*/Gateway decisions and the DOE's nuclear energy portfolio. This does not mean Congress passed a unified "Moon reactor bill," but it does mean that appropriations for the space architecture and its key energy sector partner agency **advanced together** within the same legislation[245]. As a result, Gateway resources do not simply disappear; they become questions of **reprogramming, obligation, contract modification, and political intent**.

The proposed pause for Gateway—pending a broader *Artemis* architecture reinterpretation—therefore should be treated as an isolated program decision embedded in the administration's FY 2026 request. NASA stated that the budget would terminate Gateway and repurpose already-produced components for other missions. Congress, however, took a different position: the FY 2026 Joint Explanatory Statement explicitly allocates **$700 million for Gateway** within Exploration and rejects the proposed termination of other core autonomous elements. This mismatch illustrates a central point: *Artemis*' future is shaped not only by technical logic, but by **statutory boundaries and congressional intent**. Any "pauses" of significance would occur inside an architecture whose fiscal and governance parameters Congress has already reinforced.

The Joint Explanatory Statement also clarifies the interface problem. It stipulates that departments and agencies funded under the Commerce, Justice, Science; Energy and Water Development; and Interior and Environment Appropriations Act, 2026 (Pub. L. 119–74) may not relocate resources or reorganize activities except as provided in the Act, and that any program, project, or activity cited in the Explanatory Statement, House Report, or Senate Report may not be reduced or reprogrammed without prior approval from the relevant appropriations committees. Because Gateway is specifically named and funded, a meaningful "pause" is not an internal NASA sequencing choice—it touches the congressional control layer of the *Artemis* campaign architecture itself.

The analytical point is more important than the procedural one: in *Artemis*, **program interfaces are not only technical**; they are also **fiscal, organizational, and governance interfaces**. Even if NASA leadership decides that the campaign architecture should change, that decision must still pass through appropriations lines, committee expectations, and execution controls. This means assumptions about Gateway's future cannot be treated as internal assumptions within Gateway alone. They propagate across the broader campaign because **budget authority, schedule, congressional intent, and architectural direction do not move at the same speed**. For this reason, Gateway has been analyzed as an **architectural assumption carrier**, not merely a hardware element.

[245] NASA *Procedural Requirements NPR 9470.1A* — Budget Execution.

In earlier *Artemis* logic, Gateway functioned as a staging node, aggregation point, logistics enabler, and a symbol of sustained lunar presence. The Ignition reset shifts near-term emphasis by introducing an *Artemis* III LEO-HLS demonstration and moving the first lunar landing to *Artemis* IV, while NASA states that *Artemis* V will begin building the Moon Base. None of this removes Gateway from the campaign; it simply alters **when and how staging assumptions are invoked**.

The important question is not if Gateway is "in or out." The more meaningful question is **which assumptions Gateway was carrying, and where those assumptions now reside within the revised architecture.** If Gateway previously stabilized assumptions about staging, aggregation, logistics, loiter capability, international participation, or future expansion, then a revised mission sequence must identify how these assumptions have been redistributed. Some may shift to Orion, some to HLS, some to surface systems; some may be deferred; and others may become less explicit while still shaping planning in the background.

Budget realities add another critical layer to this reinterpretation. Architecture is often discussed as though it evolves exclusively through technical trades, but a campaign architecture is also shaped by **appropriations, statutory direction, contracting constraints, industrial capacity, and workforce distribution**. A mission sequence may be technically attractive and still become fragile if it assumes funding, staffing, facilities, or production throughput that are not stable. The more *Artemis* accelerates, the more **budget becomes part of the architecture itself**, rather than an external constraint.

This is why Gateway's budget, and its architectural role, must be analyzed together. A change in staging logic is not only a mission-design matter—it affects contract protections, center workloads, integration responsibilities, partner expectations, and architectural functions that are treated as fixed, even as visible mission sequencing changes. Architectural reinterpretation inevitably creates **assumption exposure**, revealing which parts of the campaign were genuine requirements, which were inherited preferences, and which were governance-stabilized expectations.

8.5 Implications for Assumption Management

By this point, the implication for *Artemis* should be clear: **cadence should never be treated as a standalone indicator of progress.** Under current conditions, a higher tempo is being pursued against a lean and internally pressured institutional base. ASAP continues to warn that workforce dynamics, technical authority, budget pressure, acquisition structure, and mission complexity are now interacting systematically, not as isolated management issues[246]. The Office of Inspector General simultaneously warns that *Artemis* must become fiscally sustainable while preserving mission-critical capabilities.

In this environment, assumption management becomes more important—not less. If cadence accelerates without corresponding increases in workforce depth, systems-support capacity, and verification margins, the result is not simply "working harder." Instead, the institution's ability to

[246] Aerospace Safety Advisory Panel Annual Report; ASAP 2025 Annual Report.

challenge inherited premises degrades, allowing assumptions to solidify into architectural baselines, verification logic, and operational planning before they are tested. This pattern mirrors the historical pathway by which earlier human-spaceflight programs normalized weak assumptions until a mishap exposed them.

For Gateway, the practical consequence is that a genuine programmatic pause would almost certainly trigger formal reprogramming controls, not an informal administrative slowdown. FY 2026 explanatory materials specify that agencies may not relocate resources or reorganize activities outside the established process. Section 505 of the enacted minibus appropriations act states that funds may not be reprogrammed to eliminate programs, reorganize activities, or materially shift funding levels without advance notice to the Appropriations Committees. The Senate report tightens this further, requiring 30-day prior notification for reprogramming actions above $500,000 or 5 percent.

These sources support a simple conclusion: any material Gateway pause would require formal committee review, not unfold as a quick internal adjustment.

NASA's own budget-execution rules reinforce this timeline. NPR 9470.1A states that reprogramming actions exceeding congressional limits require OMB approval and subsequent congressional notification before the revised plan can be formally recorded in NASA's financial system. The directive also requires NASA to submit an initial operating plan for 15-day congressional review, and any later changes exceeding the plan's limits must undergo another 15-day review before taking effect. NASA further distinguishes reprogramming (moving funds within an appropriation) from transfers (moving funds between appropriations), the latter requiring explicit congressional authorization and a revised OMB apportionment.

The implication for assumption management is clear: an architecture can change conceptually far faster than it can change executively. A campaign may internally conclude that Gateway should be paused, downsized, or rescoped, but the appropriated program may remain legally active because budget authority, committee review, operating-plan controls, and OMB actions have not aligned. This is directly relevant to assumption propagation: fiscal control mechanisms become part of the architecture, determining how long legacy interfaces remain in force, how quickly new concepts can become real, and whether a gap opens between strategic intent and executable program reality.

Accordingly, assumption management in *Artemis* is not solely a technical and operational discipline; it is also a governance discipline. Budget-execution rules can preserve constrained campaign elements long after architectural preferences or strategic intent have shifted.

The new *Artemis* architecture shows that assumption management must function as a continuous campaign activity. A major architecture update should trigger more than a scheduling revision; it should trigger a systematic review of assumptions across architecture, verification, operations, workforce, budget, and crew survival. NASA's revised sequence changes the proof path for *Artemis* III, the landing burden for *Artemis* IV, the production and cadence expectations for

subsequent missions, and the timing relationships among commercial landers, surface systems, suits, and infrastructure.

The first implication is that assumptions must be tracked by origin and dependency, and assumptions formed under one architecture should not automatically transfer into another without review. If an assumption concerns lander readiness, docking, surface power, Gateway staging, suit compatibility, or annual cadence, it should have:

- an owner
- a rationale
- a validation trigger
- a defined consequence or contingency if it fails

Without these, a revision may change the visible mission plan while leaving earlier premises unexamined.

The second implication is that architectural changes should reopen verification meaning. Evidence generated for an earlier configuration may remain technically valid, but its relevance must be reinterpreted under the new mission sequence. As developed in Section 4, verification closure under one profile does not carry the same meaning when the mission profile, interface set, or operational context changes. Under the *Artemis* reset, this becomes an operational principle, not a theoretical one.

The third implication is that crew survival must become a central test of whether the architecture has actually matured. A revised campaign may appear coherent on paper while introducing new stresses across abort logic, rescue capacity, safe-haven assumptions, life-support readiness, surface-power dependencies, return-to-orbit timing, and workforce comprehension of revised procedures. Section 9 will take up this question directly by examining what the reinterpreted *Artemis* campaign means for crew-survival logic.

9 Crew Survival in the Reinterpreted *Artemis* Campaign

Crew survival can no longer be understood as a property of any single vehicle, subsystem or mission phase. Survival is now distributed transportation, docking, xEVAS capability, life support, communications, mobility, habitation, logistics, and the institutional ability to judge whether those elements still form a coherent human rated system. This is an important distribution because the crew can be endangered even when no individual system has clearly failed. If interfaces are mature but infrastructure is late, contingency paths are weak, or assumptions about timing and compatibility are no longer valid, survival margins can erode well before mishaps make that erosion visible[247].

The recent reinterpretation makes this problem more acute. Once mission sequence, staging, logic and proof boundaries begin to change, the survival cases change with them. The campaign may

[247] NASA Procedural Requirements 8705.2C — Human-Rating Requirements for Space Systems, Preface P.2 note 1; ch. 1 §§1.1–1.2; ch. 2 §§2.1–2.3; NASA Human System Risk Board, *Risk of Inadequate Mission Rescue (RIRM)* Human Spaceflight Risk Technical Report (2024), Ch. 1 §§1.1–1.2.

appear to still be progressing programmatically, while also becoming more fragile from the standpoint of actual crew protection. Crew survival is one of the most demanding measures of *Artemis* maturity for this reason. It forces the analysis away from whether isolated elements are advancing and toward whether the architecture, as it is now configured, can still preserve human life across nominal operations, off-nominal conditions, and contingency scenarios.

It should therefore be understood that this section treats crew survival not as a downstream operational concern, but as an integrative test of the reinterpreted campaign. The subsections that follow examine how survival depends on HLS maturity, suit readiness, abort and rescue logic, life-support and safe-haven assumptions, and broader survival dependencies, and the broader question of whether *Artemis* is maturing as a campaign faster than its survival logic can be credibly proven.

9.1 *Artemis* After Ignition: Crew Survival, Risk and Campaign Maturity

Artemis entered its Ignition period at a moment when the campaign's technical ambition was beginning to outpace its architectural resilience. The March 2026 reset represents more than a shift in sequencing; it marks NASA's acknowledgment that early *Artemis* missions had accumulated numerous interdependent first-use systems to support a reliably repeatable pathway to the lunar surface. Rearranging the order of demonstrations buys down certain risks, but it also clarifies how dependent the program remains on unproven lander systems, complex orbital integration, and survival assumptions that have yet to be tested in deep-space conditions.

Against this backdrop, *Artemis* is evaluated through the lens that ultimately governs every exploration architecture: whether it can keep crews alive when events unfold outside the nominal envelope. Ignition improves the structure of the campaign, but it does not resolve the fundamental challenge that early crews will face thin margins, limited recovery options, and a distributed multi-provider architecture where cascading failures are harder to contain. These dynamics shape the true maturity of *Artemis*, how the new sequencing redistributes risk, and where the remaining pathways to unrecoverable outcomes persist despite the architecture's evolution.

9.1.1 Architectural Reset and Risk Redistribution

NASA's February–March 2026 *Artemis* Architecture Revision (AR1)—announced publicly on March 3, 2026, and subsequently reinforced through the Ignition rollout—should be understood as a corrective response to a campaign that had become overloaded with technical first-use events, schedule compression, and stacked architectural dependencies. AR1 reorganized the sequence so that *Artemis* III becomes a 2027 LEO mission that demonstrates rendezvous and docking with one or both commercial HLS providers, performs integrated vehicle checkout, and validates early xEVAS suit operations before *Artemis* IV attempts a crewed lunar landing. Strategically, this is a more rational buildup than the previous plan because it relocates high-risk crew interfaces closer to Earth and shifts critical demonstrations earlier in the campaign.

This change matters because NASA's own ASAP had concluded that *Artemis* III, as previously baselined, could not be executed with appropriate safety margins, and that NASA had not

demonstrated an integrated risk-management strategy commensurate with the complexity of the campaign[248]. In this sense, AR1—and the Ignition reframing that followed—it is not a schedule update; but an institutional acknowledgment that the former sequencing placed too many first-use capabilities, too many unproven providers, and too many mission-critical handoffs into a single lunar sortie.

AR1 redistributes risk across multiple missions instead of requiring a single crewed landing to validate the entire architecture at once. By doing so, it reframes early *Artemis* not as a sprint to a landing, but as a sequence of risk-buydown demonstrations, each one absorbing a piece of what had previously been compressed into *Artemis* III.

9.1.2 Crew Survival as the True Measure of *Artemis* Maturity

The revised AR1 architecture should not be mistaken for a solved program. Although AR1 and Ignition improve the logic of progression by moving high-risk demonstrations earlier and closer to Earth, they do not remove the hard dependencies that dominate early *Artemis* risks. HLS remains the leading driver of uncertainty because the campaign rests on providers to be able to execute complex orbital integration, lunar South Pole operations, ascent and redocking reliability, and schedule-sensitive testing whose most crew-critical elements have yet to be demonstrated in flight-like conditions[249].

From a crew-survival perspective, a nominal mission can be completed, but can the revised architecture keep astronauts alive when operations become off-nominal? This lens is especially important because NASA's OIG found that existing *Artemis* crew-survival analyses suffer critical limitations: they typically consider only one catastrophic event at a time, assume that systems perform normally after the initiating event has been survived, and do not sufficiently address extended survival after immediate danger has passed. As a result, the official survival logic can understate the probability of cascading failures — precisely the type of hazard most likely to emerge in a distributed, multi-provider lunar campaign.

The starkest implication of the OIG report is that NASA does not currently possess the capability to rescue a stranded *Artemis* crew from lunar orbit, the lunar surface, or deep space. That finding changes the meaning of risk across the entire architecture. It means that prevention, design margin, fault tolerance, abort logic, consumables discipline, and operational rehearsal are carrying a burden that — in a more mature exploration architecture — would be shared by recovery capacity. Under the present posture, astronauts may survive the initial failure yet still be unrecoverable, because there is no safe haven, no rapidly launchable backup vehicle, and no standing rescue architecture capable of reaching them within consumables limits.

This is **why crew survival becomes the ultimate test of *Artemis* maturity** under AR1. The question is not whether the revised mission sequence is technically coherent; it is whether the redistributed architecture can withstand off-nominal conditions without leaving crews in unrecoverable states.

---

[248] Aerospace Safety Advisory Panel, *2025 Annual Report*, pp. 33–34.
[249] NASA Office of Inspector General, *NASA's Management of the Human Landing System Contracts*, IG-26-004 (2026), p. i, pp. 12–16; Aerospace Safety Advisory Panel Annual Report.

AR1 may redistribute risk — but it also exposes where survival logic and architectural resilience have yet to catch up with the revised campaign.

9.1.3 Go/No-Go Flags in the New Architecture

Incorporating crew survival shows that Ignition's strongest feature is not limited to faster cadence but better risk distribution[250]. The new *Artemis* III mission profile offers a chance to close genuine interface debt before a lunar landing by exercising docking, integrated communications, life-support interactions, and suit systems operations in a nearer term test environment. It is a meaningful green flag because survival in deep space exploration is built first through integration discipline rather than through rhetoric about tempo. If *Artemis* III is a true gate rather than a cosmetic schedule waypoint, it can materially improve the survival posture of *Artemis* IV and V.

The architectural red flags are unfortunately glaring. The HLS remains the dominant driver of crew survival risk; manual operation control questions[251] and under-demonstrated crew critical functions remain live concerns while the lunar South Pole environment can convert modest off-nominal conditions into life threatening conditions. A mission can become unrecoverable at launch or Earth return through:

- unresolved Orion or SLS issues, during in-space docking of Orion, the HLS and crew systems do not behave as an integrated stack,
- before crew boarding if the propellant transfer chain does not mature,
- at the South Pole through terrain, slope, tilt or plume effects,
- during surface operations if mechanisms, power, thermal control, communications, or life support degrades, or
- during ascent and rendezvous., where a surviving initiating fault can still end in a stranded crew.

Taken together, these green and red flags show that the architecture contains both meaningful opportunities to buy down risk and persistent structural hazards that could compound across missions. To clearly differentiate where the program is positioned to make progress versus where it remains exposed, it is useful to frame *Artemis* III and IV through a set of recurring crew-survival dimensions. Table 9 organizes these dimensions into Go/No-Go indicators, highlighting how each element of the architecture either strengthens or weakens the mission's overall survival posture.

[250] NASA Office of Inspector General, *NASA's Management of the Human Landing System Contracts*, IG-26-004 (2026), results brief p. i, pp. 29–31, 34–35.

[251] Aerospace Safety Advisory Panel Annual Report; ASAP 2025 Annual Report

Table 9 : Artemis III/IV Risk Dimensions Informing Crew Survival

Dimension	*Go*	*No-Go*	*Crew Survival Driver*
Mission Sequence	*Artemis III in Earth orbit buys down docking, suit, and integrated systems risk before a landing.*	*A partially successful test could still be treated as enough to protect a landing date*	*The quality of what Artemis III closes matters more than whether it flies on time*
Lander Maturity	*Two providers give NASA some strategic redundancy over time*	*Crew Critical functions remain under tested in flight like conditions*	*The lander is the principal loss of crew driver in early lunar missions*
Crew Survival Posture	*NASA formally identifies survival gaps and provides recommendations*	*OIG found the analyses narrow and still not a substitute for rescue capability*	*A crew can survive the first failure and still be lost afterward*
Campaign Tempo	*Cadence could eventually create learning effects and operational competence*	*Aggressive cadence before maturity turns schedule pressure that leads to accepting greater risk*	*Fast repetition helps only after the architecture demonstrates stable performance*

These dimensions illustrate that crew survival is not determined by any single system but by how consistently the architecture delivers maturity, integration discipline, and operational closure at each phase of the campaign. A Go in one dimension cannot compensate for a No-Go in another if the combined effect still allows a single initiating failure to cascade. As *Artemis* transitions from demonstration missions to sustained lunar operations, the program's ability to resolve these No-Go drivers—rather than work around them—will ultimately define whether the architecture can support repeatable, resilient, and survivable human activity at the lunar South Pole.

9.1.4 Key Failure Paths and Mishap Pathways

The aggressive timeline revealed in the architecture changes unfortunately magnifies each of these hazards exponentially. NASA has announced annual lunar missions after *Artemis* IV and, looking beyond *Artemis* V, an eventual goal of landings every six months using more commercially procured and reusable hardware. In principle, cadence can reduce risk through repetition and learning – *but only when that learning is systematically translated into design improvements and operational techniques*. In practice, cadence only becomes a safety asset **after** a system has demonstrated enough maturity to turn anomaly data into disciplined operational experience. If cadence is pursued before architecture has earned it, schedule pressure becomes a risk multiplier: design waivers become easier to rationalize, test objectives become easier to narrow, and partial success becomes easier to mischaracterize as sufficient for closure.

The schedule risk interaction is especially interesting because the 2026 OIG HLS[252] audit explicitly did not assess the cost, schedule, or technical impact of NASA's latest campaign shift. In other words, the architecture has changed faster than the public independent oversight picture has stabilized around it. That gap does not prove the reset is unsound, but it does mean confidence

[252] NASA OIG. *NASA's Management of the Human Landing System Contracts*, IG-26-004 (2026)

claims may overstate the true situation. The strongest analytical position is therefore mixed: Ignition is a more survivable direction than the older single mission approach because it buys down risk incrementally. But *Artemis* remains a program with thin recovery options, unresolved HLS maturity questions, and a timeline that could quickly turn residual risk into mishaps. Especially if the campaign begins treating dates as evidence of readiness. These dynamics set the stage for schedule pressure to become the catalyst that triggers the mishap pathways underlying all other hazards.

Scenario 1: Schedule as the Initiating Event

The first scenario pathway is the schedule becoming the initiating event. Under that pattern, the problem is not one failed vehicle but a compressed sequence of delayed demonstrations, partially closed tests, unresolved interfaces, and rising pressure to preserve public cadence. In that environment, schedule protection begins to function as an unofficial design requirement, and the campaign becomes vulnerable to treating date preservation as a proxy for technical readiness.

Scenario 2: Partial Success Treated as Closure

A second worst case scenario is that *Artemis* III in Earth orbit succeeds only partially, yet it is treated as full closure for the architecture. Because the revised campaign depends on that mission to retire docking, communications, suit and integrated systems risk before a lunar landing, a mixed result could be especially dangerous if it is used primarily as a proof point rather than a true gate. In that case, risk would not be retired so much as transferred forward into *Artemis* IV under the appearance of progress.

This scenario is extremely important—should it ever come to fruition, a consequential risk vector in its own right is that it will drive false confidence in the system and the architecture.

Scenario 3: Manual Control Logic Becomes a Late Waiver Issue

A third scenario concerns manual control logic late in the development cycle. If landing control requirements remain unsettled while the schedule tightens, the temptation grows to preserve an automation first posture rather than redesigning towards a more robust crew override capability. That is a particularly serious concern in a lunar landing architecture, because it turns a disputed design issue into a potential survival issue at exactly the point when real flight heritage is still thin.

Scenario 4: Technically Survivable but Operationally Fatal Lunar Landing

A fourth scenario branch is a lunar landing that is technically survivable but operationally fatal. A crewed lander may touch down without a crash and still become unrecoverable if tilt, access, communications, thermal control, power or ascent preparation degrade after landing. This is analytically important because it sits in the gray zone between nominal success and obvious catastrophe: the vehicle reaches the surface, the crew survives the first event, and yet, the architecture still fails to preserve a path home.

Scenario 5: Surviving the Mishap, Defeated by No Recovery

The most consequential scenario is that the crew survives the first mishap but not the absence of recovery. A fire, propulsion fault, docking failure, or surface system malfunction may leave the astronauts alive but stranded, and under the current posture the mission can shift from anomaly to fatality simply through elapsed time and consumables burndown. This is why the rescue gap matters so much: it means early *Artemis* missions depend more on never entering an unrecoverable state than on possessing a credible means of recovery once one exists - **a distinction that announces that robustness is only real when both conditions are met.**

9.1.5 Bottom Line: What Ignition means for *Artemis*

Ignition is a strategically smarter *Artemis* Architecture due to its ability to redistribute some of the old *Artemis* III "stacking of firsts" into the nearer term LEO test and reorients the campaign towards recurring surface operations. *Artemis* maturity should be judged by the quality of its survival margin rather than by the elegance of its roadmap[253]. A crucial distinction follows:

> **A program built for sustained lunar presence must prove not only that it can land, but that it can absorb failure, preserve crew contingencies, and prevent an off nominal event from becoming an unrecoverable one.**

Ignition moves *Artemis* in a better architectural direction. But the new architecture still rests on a tight schedule, immature capabilities, incomplete crew survival options, lack of congressional buy-in, and - most critically – the absence of a true rescue layer for a stranded crew. **Crew survival is, and always will be, the most visibly important measure of *Artemis*'s maturity: not whether a mission can work nominally, but whether the architecture can absorb failure without becoming unrecoverabl**e.

9.2 Campaign Maturity, Survival Logic, and Assumption Exposure

Campaign maturity **must** be evaluated as a survival question, not a program-management milestone[254]. A campaign that carries crews inherits a different burden than a single mission: it must demonstrate that its assumptions, evidence, interfaces, and operational dependencies hold under evolving architectural conditions. *Artemis* II delivered essential data, anomalies, and validated performance—but campaign maturity concerns what happens when that success is placed under cadence pressure, architectural reinterpretation, commercial integration, and shifting demonstration pathways. The survival logic of a campaign depends on whether the system can

[253] NASA Human System Risk Board, *Risk of Inadequate Mission Rescue (RIRM)*, Human Spaceflight Risk Technical Report, NASA HQ, 2024

[254] NASA *Systems Engineering Handbook* (NASA/SP-2016-6105 Rev. 2, 2016); NASA *Risk Management Handbook* (NASA/SP-2011-3422, 2011); NASA *Procedural Requirements 8705.2C — Human-Rating Requirements for Space Systems* (Preface P.2 note 1; Ch. 1 §§1.1–1.2; Ch. 2 §§2.1–2.3).

protect the crew not only during a single proof flight, but across a sequence of missions that rely on increasingly interdependent elements.

This distinction becomes central under the revised architecture. Success in one mission may confirm specific performance claims, but it does not prove the architecture can sustain docking demonstrations, integrate new commercial elements, support early surface operations, absorb delays, or maintain contingency depth when the sequence changes. Campaign maturity therefore requires more than the ability to launch—it requires the ability to adapt without allowing assumptions to drift or survival margin to erode. The integrated architecture must show that it can update its survival logic as interfaces shift, as cadence tightens, and as distributed proof streams converge into a single, crew-relevant argument.

A mature campaign knows which assumptions it carries, who owns them, what evidence supports them, and what happens when they fail. The sections that follow examine how mission success can mask assumption exposure, how architectural revisions distribute proof across partners, and how cadence pressure can preserve assumptions that should have been reopened. Each of these conditions challenges the core requirement: ***Artemis* must remain survivable not only when it succeeds, but when it is stressed, reinterpreted, delayed, or forced into off-nominal conditions.**

9.2.1 Campaign Maturity as a Survival Question

Artemis II delivered what a crewed test flight is supposed to deliver: hard data, anomalies to learn from, and validated systems at scale. NASA's initial public assessments report that Orion's thermal protection system performed as expected with significantly reduced char loss versus *Artemis* I, and that SLS met its targeted insertion performance; at the same time, teams began root-cause analysis of a urine-vent line anomaly for corrective action before *Artemis* III. That blend — acknowledging success, examining the data closely, and immediately rolling lessons forward — is the discipline of a healthy test campaign and sets an evidence baseline for subsequent missions.

Where the bar now rises is the difference between mission success and campaign sufficiency. NASA's updated plan shifts the first crewed lunar landing to *Artemis* IV in 2028, establishing an expected annual surface cadence that increases campaign pressure. The survival-relevant question becomes whether the integrated architecture — Orion, HLS, xEVAS, operations, logistics, and surface systems — can sustain compressed cadence, interface complexity, and contingency depth without eroding crew margin as the campaign transitions from proof to production. A single successful flight builds confidence; it does not, by itself, prove the campaign can absorb accelerated integration and still protect crews under off-nominal stress.

What must be shown: that the chain of evidence established on *Artemis* II remains valid in new contexts — docking in LEO, integrated operations with commercial landers, and later surface operations — and that success remains cumulative rather than isolated. In practice, this means bounding what *Artemis* II actually proved (TPS performance, SLS insertion, flight ops under a free

return profile) and what remains to be proven (Orion–HLS docking, transfer protocols, suits in operational environments, and surface logistics under duration creep)[255]-.

9.2.2 Assumption Exposure Across Architecture, Verification and Operations

Assumption exposure occurs when the architecture changes faster than the assumptions supporting it are reviewed[256]. *Artemis* is now in that condition. The reset changed mission sequence, proof boundaries, staging logic, cadence expectations, commercial lander roles, and the timing of surface operations. Those changes create exposure across architecture, verification, and operations.

Architecturally, the campaign now depends on a different order of demonstrations. *Artemis* III no longer carries the same meaning as the first lunar landing mission; it is an integrated proof step. Verification-wise, evidence generated for earlier concepts must be reinterpreted against the revised sequence. Operationally, crews and flight controllers must train for a new set of handoffs, timelines, decision points, and contingency cases.

Exploration requires assumptions because not every future condition can be fully proven in advance and there is inherent but understood and accepted risk. The overlooked risk, however, is that assumptions become detached from their original rationale. An assumption about HLS readiness has one meaning when *Artemis* III is a landing mission and another meaning when *Artemis* III is a LEO docking demonstration. A Gateway related assumption has one meaning in a Gateway centered staging concept and another meaning when early missions shift emphasis toward Orion-HLS integration and direct surface operations. A suit readiness assumption may appear acceptable when schedule margin exists but become survival critical when the first surface landing date compresses.

For that reason, assumption exposure should be treated as a safety issue. When the architecture changes, assumptions should not migrate silently. They should be reidentified, re-owned, and revalidated against the new sequence. Otherwise, the campaign may appear aligned while the different parts of the system continue operating against different mental models of what the mission is proving.

Cross-Program Interfaces and the Distributed Proof Burden

The distributed proof burden becomes especially important in crew survival. NASA's HLS effort depends on SpaceX and Blue Origin landers that must allow crew to descend to the lunar surface, temporarily live and work there, and ascend back to lunar orbit. NASA OIG states that NASA is working with both providers, but that lander development challenges will delay planned *Artemis* launch dates. The same OIG summary states that NASA does not currently have the capability to rescue crew if they become stranded in space or the lunar surface.

---

[255] NASA Systems Engineering Handbook (NASA/SP-2016-6105 Rev. 2, 2016), §§5.3–5.4, 6.8.

[256] NASA Procedural Requirements 8705.2C — Human-Rating Requirements for Space Systems (Preface P.2 note 1; ch. 1 §§1.1–1.2; ch. 2 §§2.1–2.3); NASA Office of Inspector General, *NASA's Management of the Human Landing System Contracts*, IG-26-004 (2026); NASA Human System Risk Board, *Risk of Inadequate Mission Rescue (RIRM) Human Spaceflight Risk Technical Report* (2024); NASA Moon Base Users Guide.

That finding alone makes the distributed proof burden unavoidable. The survival case is not produced in one place. A provider may generate evidence for its lander. NASA may provide insight, oversight, certification, safety judgement, and mission approval. Orion teams may own Earth-return capability. Mission operations may own procedure execution and real-time decision making. xEVAS teams may own suit performance and surface exposure limits. Surface systems may own power, mobility, communications, and infrastructure. None of those evidence streams alone closes the crew survival claim.

This does not mean distributed development is inherently unsafe. It means that distributed evidence must be integrated into a single survival argument. The danger is false confidence created by evidence that is valid locally but incomplete systemically. A provider can show progress inside its design boundary. NASA can show insight into risk areas. Orion can show Earth-return performance. A suit provider can show developmental progress. But the crew depends on the combined systems under off-nominal conditions. If no organization owns the combined meaning of the evidence, survival assumptions can become dispersed across the same architecture that the crew depends on.

This is where the distributed proof burden image once again becomes relevant. Its purpose is not to show that *Artemis* lacks proof activity, but that proof is distributed across stakeholders, centers, providers, programs, and partners, while the campaign level survival claim still has to be assembled and defended somewhere. The distributed proof written is not a criticism of commercial partnership, it is a recognition of what distributed architectures require. If SpaceX, Blue Origin, Axiom, Orion, Gateway, mission operations and surface systems each produce evidence, the campaign still needs an integrating authority capable of determining whether those partial proofs close the survival case.

For *Artemis*, that assembly point is critical: If no organization owns the combined meaning of the evidence, then assumptions about survival can become dispersed across the same architecture that the crew depends on.

Structural Assumption Preservation Under Cadence Pressure

Cadence pressure can preserve assumptions by making them harder to reopen. When a campaign is moving towards annual missions, there is less time for anomaly interpretation, requirements for vision design, reconsideration, retesting, retraining and procedural change. NASA's architecture update states that the revised campaign aims for about 1 lunar mission per year after the added 2027 *Artemis* III demonstration and the planned 2028 landing. That ambition increases the value of learning quickly, but it also raises the cost of reopening assumptions that disrupt the next mission.

Under these conditions, conditions once again, assumptions do not drift. They are structurally preserved. They remain in place because the system has become dependent on them. The Cadence plan needs stable mission packaging. Mission packaging needs stable interface assumptions. Interface assumptions need stable evidence claims. Evidence claims rely on interpretation by a workforce that may already be constrained by workload schedule and institutional turnover. The

more the campaign accelerates, the more difficult it becomes to interrupt the chain and ask whether an inherited premise still deserves confidence.

This is one of the most important links between Section 7 and Section 9. Structural pressure does not have to force an explicit bad decision to create risk. It can narrow the system's willingness to revise earlier decisions, and that matters because the crew enters the architecture at the end of a long chain of assumptions. **If that chain has been preserved by schedule necessity rather than renewed confidence, survival margin may be thinner than the official milestone sequence suggests.**

What Campaign Maturity Must Prove

Campaign maturity must prove more than technical progress. It must prove that the architecture can protect crews when assumptions change, when elements mature unevenly, and when off nominal conditions force decisions across organizational boundaries. For *Artemis*, that means proving that Orion, HLS, xEVAS, surface infrastructure, mission operations, abort logic, radiation management, medical response, and return pathways form a coherent survival system.

The most credible test of *Artemis* maturity is not whether each element can report progress inside its own lane, but whether the campaign can maintain a valid survival case as the architecture is reinterpreted. A mature campaign knows which assumptions it carries, who owns them, what evidence supports them, when they must be reopened, and what happens if they fail. This is the standard Section 9 applies to the *Artemis* survival chain.

9.3 Abort, Rescue, and Contingency Assumptions

Abort, rescue, and contingency functions must be treated as core survival logic in *Artemis*, not as optional features of individual vehicles, since no single system carries the full burden of keeping the crew alive across all mission phases. Abort is often described as a propulsion or trajectory function tied to a specific spacecraft, but in *Artemis* it becomes a campaign-level survival assumption[257]. The crew's ability to escape danger depends on which element can still act, which interface remains reachable, and which transfer, or shelter option remains viable under stress. Rescue and contingency capability therefore cannot be inferred from the existence of multiple systems in the architecture; they must be demonstrated as reachable paths the crew can actually use within the time available.

This framing is essential because *Artemis* survival logic is distributed across Orion, HLS, Gateway (when present), surface systems, suits, and mission operations. Each phase—launch, rendezvous, docking, transfer, descent, surface stay, ascent, loiter, and return—requires a different combination of vehicles, interfaces, and operational decisions to protect the crew. Abort and contingency capability must therefore be understood as a chain of survivable options, not a single maneuver. If

[257] NASA Human System Risk Board, *Risk of Inadequate Mission Rescue (RIRM) Human Spaceflight Risk Technical Report* (2024), human-system-risk sections; NASA Moon Base Users Guide.

any link in that chain depends on an element that is delayed, incompatible, unreachable, or uncrewed, the architecture may portray redundancy that does not exist in practice.

NASA's Moon Base Users Guide reinforces this distinction by showing how future infrastructure—power systems, shelters, mobility, logistics hubs—cannot yet be assumed as operationally dependable. Until such elements exist as reachable safe-haven or rescue-enabling assets, abort and contingency logic must rely on what is actually available, not on anticipated future capability. The campaign must therefore identify which abort or rescue paths are real, which are conditional, and which remain aspirational. If the crew cannot reach a refuge in the time provided by consumables, trajectory constraints, or system endurance, that option is not part of the survival architecture.

For Section 9.3, then, the task is clear: abort, rescue, and contingency paths must be evaluated as mission-phase-specific survival capabilities grounded in evidence, readiness, interface compatibility, and timeline. Anything less risks allowing high-level architecture graphics to imply crew protection that does not exist in operational reality.

9.3.1 Abort as a Campaign Level Survival Assumption

Abort is often discussed as a vehicle function, but in *Artemis* it becomes a campaign level survival assumption. Orion provides the central crew transportation and Earth-return capability, but *Artemis* missions increasingly depend on handoffs among Orion, commercial landers, potential staging systems, surface assets, suits, and mission operations. Abort logic must therefore be understood as a chain of reachable options rather than a single escape maneuver.

The question that has been central to this discussion is whether the crew has a credible path to safety at each phase of the mission. During launch and return, that path may be tied primarily to Orion and SLS. During rendezvous, docking, transfer, lunar descent, surface operations, ascent and return to Orion, the path depends on the architecture remaining coherent under stress. Abort becomes a campaign level assumption because the crew's ability to survive depends on which element can still act, which interface remains available, which vehicle can receive the crew, and which operational decision can still be executed.

This is important under the revised architecture because *Artemis* III now exists to test rendezvous and docking in LEO before the first lunar landing. That is a prudent step, but it also confirms that docking and transfer are survival relevant assumptions, not background mechanics. They are the pathways through which the crew moves between safe states.

Rescue Capacity and the Limits of Distributed Architecture

The strongest current source for the rescue problem is NASA OIG's March 2026 HLS audit. The OIG states that NASA is taking steps to mitigate and prevent hazards associated with landers, but the agency does not currently have the capability to rescue crew if they become stranded in space or on the lunar surface. That finding should be treated as central in this section because **it turns crew survival from an abstract principle into a concrete architectural limitation.**

A distributed architecture can increase capability, innovation, and flexibility, but it can also expose the limits of rescue. If a crew became stranded on a lander in lunar orbit or on the surface, rescue is not possible simply because multiple systems exist somewhere in the campaign. Rescue requires the right system to be ready, reachable, compatible, staffed, fueled, launched or positioned, and authorized within the crew's survival window. The existence of alternative providers does not automatically create a rescue capability.

This is where assumption management becomes survival management. If rescue is unavailable, the campaign must be explicit about what that means. It may choose to accept that risk under defined conditions, but it should not treat distributed architecture as if it inherently provides redundancy. Redundancy only exists when alternate paths are operationally usable within the time available to the crew.

Contingency Logic Across Orion, Gateway and HLS

Contingency logic must be mapped across Orion, Gateway, and HLS because no single element owns the entire off-nominal sequence. Orion may provide Earth return. HLS provides lunar descent, surface stay, lunar ascent, and crew transfer. Gateway may provide staging, communications, logistics, or safe-haven capability depending on the mission configuration. The architecture therefore must define which element carries which portion of the contingency burden and under what conditions that burden transfers.

This is where Gateway's role becomes important, even if its near-term usage evolves. If Gateway is part of a contingency or safe-haven concept, then its timing, availability, and configuration directly affect crew survival. If Gateway is not available for a specific mission sequence, the campaign must identify what element absorbs the contingency functions Gateway would otherwise support. Reinterpreting Gateway as less central to early missions does not remove the need for staging and refuge logic—it relocates that burden.

HLS complicates the issue further because it is both a mission vehicle and a temporary crew environment[258]. OIG describes HLS as enabling crew to descend to the lunar surface, temporarily live and work there, and ascend back to lunar orbit. That means HLS is part transportation system, part habitat, part abort platform, and part return-path dependency. Contingency logic must account for all of those roles.

Recovery Windows, Loiter Time, and Decision Margin

Recovery windows are where survival logic becomes time logic. A crew can survive an off-nominal condition only if the architecture provides enough time for diagnosis, decision, action, and recovery. Loiter capability, consumables, thermal margins, battery life, communications availability, and trajectory constraints all determine how much decision margin exists.

---

[258] NASA, NASA-STD-3001, Volume 2: Human Factors, Habitability, and Environmental Health, Rev. D (2025), pp. 49, 155, 236–237, 241, 261, 313, 346.

Decision margin is especially vulnerable in a compressed campaign. If mission designs assume tight timelines, minimal loiter capability, infrequent rendezvous opportunities, or short safe-haven durations, then contingency response becomes brittle. The crew may have options in theory but lack the time to execute them. In that case, the architecture contains backup paths, but those paths do not function as real survival margins.

This is why recovery windows should be treated as assumptions requiring explicit ownership. The campaign should identify which recovery windows exist, which systems sustain the crew during those windows, what conditions close them, and who has authority to adjust mission posture when a window begins to narrow.

When Backup Plans Exist Only on Paper

The most dangerous contingency assumption is the one that appears real in architecture diagrams but has not been demonstrated under operational conditions. A backup path exists only on paper if the hardware is unavailable, the interface is untested, the crew cannot reach it, the timeline is too short, the procedures are immature, or the authority structure cannot make a timely decision. This is why the OIG finding on rescue capability matters so much. *Artemis* has multiple providers and multiple architectural environments, but survival depends on usable paths, not conceptual alternatives.

It should then stand that the campaign **must** distinguish redundancy from contingency and aspiration:

- Redundancy means a viable alternate path exists.
- Contingency means a planning response exists under defined limits.
- Aspiration means the campaign 'hopes' an alternate path will exist later.

Confusing these categories can make the architecture appear more survivable than it is.

The assumption-management requirement is direct: every backup path should be tied to evidence, timeline, authority, crew access, and defined failure conditions. If a path cannot meet those tests, it should not be used to support survival confidence.

9.4 Life Support, Habitability, and Safe Haven Logic

Life support and habitability must be treated as core survival systems in *Artemis*, not as environmental conveniences or comfort features. On the lunar surface, these functions determine how long the crew can survive, how effectively they can work, and how much margin exists when conditions deteriorate. The Moon Base Users Guide reinforces this by defining habitation systems as the volumes and supporting subsystems that ensure astronaut health and performance in controlled lunar environments[259]. It explicitly links habitability to longer surface duration, expanded science operations, increased EVA time, extended medical capability, greater operational independence

[259] NASA Human System Risk Board, *Risk of Inadequate Mission Rescue (RIRM) Human Spaceflight Risk Technical Report* (2024), human-system-risk sections; NASA Moon Base Users Guide.

from Earth, and reliable waste-stream management. These are not secondary benefits—they are direct determinants of survival margin[260].

Habitability becomes part of the survival chain alongside life support, safe-haven access, consumables, medical capability, thermal control, operational workload, and the crew's ability to sustain performance under stress. A habitat or vehicle may satisfy nominal living requirements yet still fail as a safe haven if its power, logistics, medical capacity, thermal stability, or communications are insufficient to protect the crew during an off-nominal scenario. For Section 9.4, the Moon Base Users Guide therefore provides a crucial frame: habitability must be evaluated as a mission-phase-specific survival capability, not inferred from the existence of a habitable volume or a pressurized shell.

An integral distinction because future *Artemis* crew are not passive riders—they are operators whose cognition, judgment, procedures, and physical endurance directly affect survival. Habitability influences time-on-task, error rates, decision quality, and fatigue tolerance. Safe-haven logic likewise depends on whether the refuge is reachable, powered, habitable, and able to sustain the crew long enough for recovery. Life support, habitability, and refuge systems must therefore be treated not as parallel technical domains but as integrated survival mechanisms that define how much time the crew has, how safely they can act within that time, and how they recover when the mission no longer unfolds as planned.

Life Support as a Survival Margin

Life support is not a background spacecraft function – for an astronaut in the lunar environment it is time converted into survivability. In nominal conditions, life support enables planned mission duration. In contingency conditions, it determines how long the crew can wait, troubleshoot, retreat, transfer, or recover. That makes life support a margin system rather than an environmental convenience.

Orion's role is central because it carries the crew through deep-space transit and Earth return. NASA describes Orion as a spacecraft that carries astronauts to deep space, sustains the crew during space travel, provides emergency abort capability, and returns crews safely to Earth. That makes Orion the survival backbone of *Artemis*. But a backbone is not a complete survival architecture. Once the crew transfers to other elements, life-support responsibility becomes distributed across HLS, suits, potential safe-haven assets, and mission operations.

The *Artemis* II urine-vent-line issue is useful because it shows how a non-catastrophic crew-systems anomaly can become a learning point for later missions. NASA's initial assessment states that teams were gathering data to identify root cause and initiate corrective action for *Artemis* III. The issue did not define the mission, but it demonstrates why life-support and crew systems must be evaluated as part of survival logic, not just comfort or habitability.

---

[260] NASA, NASA-STD-3001, *Volume 2: Human Factors, Habitability, and Environmental Health, Rev. D* (2025), pp. 49, 155, 236–237, 241, 261, 313, 346.

Habitability, Human Performance, and Time-on-Task Risk

Habitability affects crew survival because humans are not passive payloads. They must interpret conditions, execute procedures, manage tools, respond to anomalies, and sustain performance under fatigue, confinement, workload, thermal stress, suit constraints, and communication delays. NASA's Moon-to-Mars architecture treats habitation systems as a sub-architecture designed to ensure the health and performance of astronauts in enclosed environments. That framing is important: habitability is directly tied to performance, and performance is directly tied to survival.

Habitability considerations span Orion, HLS, xEVAS, rovers, surface shelters, and potential safe-haven locations[261]. A mission may meet the minimum technical requirements for life support while still imposing human-performance demands that erode safety margins. If crews are rushed, fatigued, poorly sheltered, or required to operate inside awkward interfaces, the probability of error increases. A survival architecture must account for human performance as part of the system, not as an afterthought.

Time-on-task risk is particularly important for surface missions. The longer the crew operates away from a protected environment, the more survival depends on suit performance, communications, navigation, mobility, medical support, and the ability to retreat. Habitability determines how much usable time the crew has before stresses begin to consume operational judgment.

Safe-Haven Logic and Reachable Refuge

A safe haven is meaningful only if the refuge is reachable, habitable, and operational within the time available. In campaign language, it is easy to refer to Orion, HLS, Gateway, or a surface habitat as potential refuges. In survival logic, however, these refuges must be assessed against location access, docking compatibility, power, thermal control, consumables, communications, and duration. A safe haven cannot be assumed because an element exists somewhere in the architecture; it must be tied to a specific mission phase.

During transit, Orion may serve as the refuge. During surface operations, HLS or a habitat may carry that role. During lunar-orbit operations, Gateway could provide value—if it is available and configured for that mission. The assumption that the crew can shelter must therefore be broken into phase-specific claims.

This is important not only for the crew but for mission risk posture. Safe-haven assumptions can quietly reduce perceived risk: if reviewers believe a reachable refuge exists, they may accept narrower operational margins elsewhere. If that refuge is delayed, unavailable, untested, or incompatible with the failure case, then earlier risk acceptance was based on a false support condition.

---

[261] NASA, NASA-STD-3001, *Volume 2: Human Factors, Habitability, and Environmental Health, Rev. D* (2025), pp. 49, 155, 236–237, 241, 261, 313, 346.

Safe-haven logic must be verified as a mission-specific capability and not inferred from campaign ambition.

Consumables, Reliability, and Duration Creep

Consumables are another area where assumptions can become fragile. Oxygen, water, waste handling, carbon-dioxide removal, power, thermal control, suit consumables, and food all determine how much time the crew has before an abnormal condition becomes unrecoverable.

Duration creep occurs when a mission's actual or contingency timeline expands beyond the original planning basis while the consumables logic remains tied to earlier assumptions. Duration creep arises from delayed rendezvous, docking anomalies, surface-timeline slips, suit problems, communication loss, ascent delays, or weather-related recovery constraints on Earth. Each event may be manageable on its own, but the hazard only materializes when multiple small delays accumulate and consume reserves faster than the mission model assumed or accounted for.

Consumables should also be managed as part of assumption traceability and the campaign must be fundamentally aware how long each element is expected to support the crew, what margins exist, what assumptions drive those margins, and what happens when an architecture update changes mission timing. If consumables are treated as static engineering numbers while mission sequencing shifts around them, the crew may inherit less flexibility than the architecture suggests.

When Habitability Assumptions Become Survival Hazards

Habitability assumptions become survival hazards when they are treated as stable even though the mission environment has changed. A cabin-volume limit, suit-endurance limit, waste-management capability, thermal margin, or sleep plan may be adequate for one mission profile and inadequate for another. The danger lies in carrying the original interpretation forward after mission duration, task sequence, crew workload, or contingency expectations have changed.

This is precisely why habitability should be tied directly to assumption triggers. **IF** mission cadence increases, if surface duration changes, if the landing site becomes more operationally demanding, if suits or rovers slip, or if safe-haven logic shifts, habitability assumptions need to be reopened. The crew can endure hardship—human spaceflight has always required endurance, and this is precisely why countless months is required for training, but if the campaign is relying on endurance to compensate for assumptions, then it needs to be revised[262].

9.5 Human Landing System Maturity and Interface Risk

HLS is not a peripheral or isolated element; it is the campaign's lunar surface-access capability. NASA's current HLS posture assigns SpaceX to provide the landing system for *Artemis* III and *Artemis* IV, and Blue Origin to provide the Blue Moon system for *Artemis* V[263]. NASA's own HLS overview makes clear that the long-term intent is not a one-off landing, but a recurring service

[262] NASA, *NASA-STD-3001, Volume 2: Human Factors, Habitability, and Environmental Health, Rev. D* (2025), pp. 49, 155, 236–237, 241, 261, 313, 346.
[263] NASA Office of Inspector General, *NASA's Management of the Human Landing System Contracts*, IG-26-004 (2026).

model—one designed to support increased crew size, more delivered mass, and longer surface stays. HLS is both an enabling system and a pacing system: if it is late, immature, or only partially verified, *Artemis* surface operations will more than just degrade—they stop[264].

There are strong reasons for NASA to use commercial HLS providers. NASA explicitly argues that pursuing multiple lander providers can increase competition, reduce cost to taxpayers, support a regular cadence of lunar landings, and help develop a long-term lunar economy. The agency also states that industry partnerships allow NASA to share knowledge while maintaining oversight of safety as companies develop, test, and mature their designs. These are not trivial advantages. A multi-provider HLS approach gives *Artemis* a pathway away from a single-point architectural bottleneck, broadens the industrial base, and aligns better with a campaign architecture aimed at recurring—not singular—missions.

At the same time, HLS is exactly the kind of element that must be treated with caution within an assumption-management framework. *Artemis* clearly needs a lander, but NASA and its partners are at risk of treating HLS as more mature, more available, or more representative than the current evidence supports. HLS functions as a dense node of architectural assumptions: assumptions about commercial development rates, launch cadence, cryogenic propellant transfer, cross-program interface stability, representativeness of demonstrations, crew survivability, and the degree to which local contractor progress can be translated into campaign-level confidence.

Bottom line: HLS is indispensable because *Artemis* cannot achieve lunar surface access without it. But both HLS's must be approached with circumspection and vigilance since NASA's own oversight shows that both providers still carry significant technical, schedule, verification, and crew-safety uncertainties. Uncertainties that can propagate directly into the campaign architecture.

9.5.1 SpaceX and Blue Origin as Assumption Carriers

The SpaceX and Blue Origin concepts are ambitious, but in different ways. For *Artemis* III and *Artemis* IV, SpaceX's Starship HLS architecture depends on a propellant storage depot in LEO, more than ten tanker launches, repeated rendezvous and transfer events, and a propellant-aggregation campaign that begins more than 200 days prior to crew launch. NASA's OIG reports that SpaceX is targeting approximately a one tanker flight every six days until sufficient propellant has been accumulated, after which the uncrewed lander launches, is refueled, proceeds to near-rectilinear halo orbit (NRHO), and remains there awaiting the crew.

Blue Origin's Blue Moon architecture likewise depends on cryogenic management and on-orbit refueling, but through a different chain: a transporter in LEO acting as a propellant depot, a fleet of refuelers, additional aggregation in a higher-energy staging orbit, and a final propellant transfer in NRHO before descent to the lunar surface. In both cases, the HLS is not just a vehicle, but the visible

[264] NASA Office of Inspector General, *NASA's Acquisition of Next -Generation Spacesuit Services*, IG-26-006 (2026), pp. 2-6, 20.

tip of a broader logistics and aggregation architecture whose success depends on many upstream events occurring on time and as designed.

This assumption burden weighs heavy since both HLS architectures embed maturity assumptions upstream of the landing itself. For Starship, a successful lunar mission presupposes that vehicle-to-vehicle cryogenic transfer, long-duration storage, pad turnaround, and repeated tanker operations can all be performed with enough regularity to support the mission timeline. NASA OIG identifies cryogenic propellant transfer as one of the most significant technical challenges facing SpaceX. It notes that the relevant technologies and processes are entirely new and have never been performed between vehicles, and reports that NASA is already tracking a top risk that the cryogenic technology SpaceX is developing may not be adequately mature ahead of *Artemis* III. That same report states that SpaceX has not yet demonstrated the required 12–24 day launch-pad turnaround, and that failure to meet the planned propellant-aggregation cadence could affect the *Artemis* III mission timeline. These are not narrow provider-level issues; they are campaign assumptions with direct implications for autonomous scheduling, launch sequencing, and mission readiness.

Blue Origin carries a parallel, though not identical, assumption burden. NASA OIG reports that Blue Origin's uncrewed demonstration mission—intended in part to demonstrate cryogenic propellant transfer—faces similar significant technical challenges. This includes the ability to store cryogenic propellants in space, minimize boil-off, and perform transfer operations across multiple orbital regimes. OIG explicitly warns that because Blue Origin's solutions are still maturing, and the aggregation chain may not yet be sufficiently demonstrated. This two-provider strategy designed to improve resilience can itself become an assumption hazard if provider diversity is mistaken for schedule margin before critical capabilities are actually demonstrated.

9.5.2 Cadence as a Risk Multiplier

NASA's public rationale for maintaining multiple HLS providers is tied directly to recurring operations. The agency states that it is pursuing multiple providers to support a regular cadence of astronaut lunar landings and continuous missions. It frames future HLS work around repeatable services such as increased crew capacity, greater delivered mass, and longer surface durations.

The Option B addendum issued in March 2026 reinforces this point, describing NASA's February 2026 campaign update as including an additional *Artemis* mission in 2027 and at least one lunar surface mission every year thereafter. From an assumption-management perspective, cadence is no longer a scheduling target but an architectural claim about the repeatability of the system. A stable launch tempo and infrastructure readiness provide the basis for reliable cross-program interfaces.

The real risk is that cadence can harden an aspiration into an operational assumption before the underlying capability has been earned. The NASA OIG finds that HLS development depends on concurrent development of Orion, Gateway, and next-generation spacesuits. These systems are at different levels of design maturity or operational readiness, changes in one can force additional

analysis, interface updates, and working groups in the others. In other words, HLS cadence is not determined by the lander alone; it is constrained by the entire architectural stack[265].

When the campaign begins to discuss annual landings while HLS interfaces and associated systems are still undergoing major design changes, cadence itself becomes an assumption that propagates into budgets, milestones, public commitments, and risk posture faster than the technical evidence supports.

Verification, Operations, and Crew Survival Risks

This cannot be said or highlighted enough in this paper: the strongest reason for caution is crew survival. The OIG states that the SpaceX and Blue Origin uncrewed demonstration missions will not use configurations fully representative of the planned crewed vehicles, identifying missed opportunities to apply "test like you fly" principles. Specifically, NASA did not require demonstration of critical crew systems such as ECLSS, the vehicle airlock, or the elevator. As a result, the test vehicles will not carry representative mass, and end-to-end propellant aggregation will not be fully flight-tested prior to the first crewed missions.

For SpaceX specifically, the OIG notes that omitting the elevator from the uncrewed mission removes the opportunity to test its performance in the actual lunar environment, where surface tilt or regolith conditions may affect operation. These findings highlight areas where technically valid subsystem verification may still leave a gap between demonstrated performance and the demands of high-rate crewed flight operations.

The operational concerns deepen with issues of crew control and crew survival. The OIG reports a disagreement between NASA and SpaceX over whether the provider's proposed landing approach meets NASA's requirement for manual crew control. It warns that failure to reach a resolution before CDR could effectively lock in automation as the only available landing mode. The OIG emphasizes that manual control is a key element of human-rating certification and an essential crew-survival strategy.

More broadly, the OIG finds that crew-survival analyses are often conducted too late to influence early design decisions, tend to consider only one catastrophic event at a time, and do not account for prolonged survival after the initial hazard. Most critically, the report states that without a rescue capability in the early *Artemis* missions, the crew would be lost if the HLS were to become disabled on the lunar surface or unable to dock with Orion or Gateway in NRHO.

This finding alone demonstrates why HLS cannot be treated as a routine transportation service just because a contract exists or a demonstration flight has occurred.

[265] NASA Office of Inspector General, *NASA's Management of the Human Landing System Contracts*, IG-26-004 (2026), results brief pp. i–ii; Aerospace Safety Advisory Panel Annual Report (2025).

9.5.3 HLS Within an Assumption-Management Framework

Within the framework of this paper, the HLS is best understood as a high-consequence carrier of distributed assumptions:

First, it carries **architectural assumptions**:

- commercial providers will mature critical propulsion, storage, transfer, docking, and surface systems on the timeline the campaign requires;
- repeated launch and aggregation sequences will operate reliably;
- cross-program interfaces will remain sufficiently stable to support recurring lunar operations.

Second, it carries **verification assumptions**:

- uncrewed demonstrations, ground tests, analysis activities, and partial flight-like configurations will be representative enough to support crewed confidence, even when key systems or operational sequences are not exercised end-to-end.

Third, it carries **operational assumptions**:

- crews will retain adequate control authority;
- the survival posture is acceptable in the absence of a rescue capability;
- novel lunar-environment effects—dust, plume interaction, tilt, and ascent hazards—will remain within manageable bounds.

Finally, it carries **governance assumptions:**

- NASA's insight and oversight model will provide sufficient visibility and influence to assure safety, even though the agency does not own the vehicles and has reduced some traditional government reviews and data-submission requirements.

For these reasons, HLS should appear in this paper as more than a simple subsystem as it is a campaign-level assumption driver. Problem statements should reference HLS because, without a lander, there is no lunar surface campaign. However, analyses should treat HLS only within a posture of explicit assumption tracking and periodic revalidation.

SpaceX or Blue Origin will eventually field capable systems – that should not be doubted. But whether autonomous decision-making will maintain clear distinctions between capability that is contracted, capability that is demonstrated, and capability that is fully mature for recurring human operations is key here. If these categories blur, agility can become a mechanism by which ambition outpaces evidence. If they are kept distinct, HLS becomes one of the clearest examples of why assumption management is a core safety and systems-engineering function in modern human spaceflight.

Human-rating policy gives a stronger basis for the distinction between completed verification and justified confidence. NPR 8705.2C treats human-rating certification as a lifecycle process involving system design, verification and validation of system capabilities and performance, flight testing, and certification/operation of the human-rated system (NASA, 2017/2028, NPR 8705.2C, Ch. 1-2). For *Artemis*, this supports the argument that HLS, xEVA, Orion, Gateway, and supporting systems cannot

be evaluated only as isolated products; their evidence must support the integrated conditions under which crew safety actually depends on them.

9.6 xEVAS Readiness and Survival Critical Assumptions

The xEVAS is a survival system, not a peripheral accessory to lunar exploration. It is the only means by which the crew can safely leave a pressurized environment, operate on the lunar surface, and return to a vehicle alive. Pressure containment, life support, mobility, thermal regulation, communications, dust tolerance, and contingency response are not comfort features—they are the boundaries of human survivability on the surface. Suit maturity is therefore inseparable from campaign maturity: the architecture cannot credibly assert lunar-surface capability until xEVAS readiness is established as a survival-critical function.

Suit readiness also determines what the campaign believes it can safely ask a crew to do. The suit mediates every human interaction with the surface environment: how far a crew can travel, how long tasks can last, how quickly anomalies can be addressed, how fatigue accumulates, and how wide the crew's operating envelope can safely extend. Weak suit assumptions propagate directly into weak surface-mission assumptions. If mobility, endurance, dexterity, thermal performance, dust protection, or emergency-return capability are not fully understood, then surface timelines, traverse plans, scientific objectives, and contingency logic become speculative rather than survivable.

The single-provider reality intensifies this dependency now that Collins Aerospace is no longer active under xEVAS, NASA now relies solely on Axiom Space for next-generation suit capability[266]. That consolidation does not inherently undermine success, but it concentrates technical and schedule risk into one development path. A delay is not merely a procurement issue—it is a survival constraint on whether the crew can safely conduct lunar-surface operations at all. Suit readiness is not downstream of the architecture but its survival pillar.

For Section 9.6, the requirement is clear: xEVAS must be treated as a mission-phase-specific survival capability, not a technology program. The suit must be evaluated not only as hardware but as a system integrated with landers, airlocks, rovers, mobility systems, communications, consumables, procedures, and timelines. Suit maturity defines human exposure margin, and exposure margin defines whether surface exploration is survivable when conditions deviate from nominal. A campaign that assumes surface operations before suit readiness is fully demonstrated is not only assuming technology—it is assuming human survivability.

9.6.1 The Spacesuit as a Survival System

The spacesuit is a survival system, not an accessory to surface exploration. It provides the pressure environment, life support, mobility, thermal protection, communications capability, and operational envelope that allow the crew to leave a pressurized vehicle and return alive. That makes suit maturity inseparable from crew survival.

[266] NASA Office of Inspector General, *NASA's Acquisition of Next-Generation Spacesuit Services*, IG-26-006 (2026).

A lander can reach the Moon, and a crew can descend to the surface, but lunar exploration cannot be considered operationally mature if the suit cannot support safe egress, EVA execution, contingency response, and reentry into the vehicle. Suit readiness therefore belongs in the survival section, not only in a technology-development discussion[267].

The suit also mediates between the architecture and human performance. It determines how far the crew can move, how long tasks can last, how quickly a crew member can respond to a contingency, and what kinds of surface operations are credible. If suit assumptions are weak, surface-mission assumptions weaken with them.

Axiom as a Single-Provider Survival Dependency

NASA OIG reports that NASA originally awarded contracts to Axiom Space and Collins Aerospace to compete to provide next-generation spacesuits for *Artemis* lunar missions and ISS operations[268]. However, NASA and Collins agreed in 2024 to remove Collins's task orders, leaving Axiom as the only active provider under xEVAS. OIG also states that NASA has been challenged to ensure spacesuit readiness for the *Artemis* lunar surface mission in 2028 and for ISS needs before the station's planned decommissioning in 2030.

That changes the assumption structure. A two-provider strategy distributes schedule and technical risk. A single-provider strategy concentrates it. This does not mean Axiom cannot succeed, but it does mean the campaign's survival logic depends more heavily on one development path. If that path slips, the consequence is not limited to procurement—it affects whether crews can safely perform lunar surface operations at all.

For this paper, the key point is that xEVAS is part of the architecture's survival logic. So, when a campaign assumes lunar surface operations **before** suit readiness is firmly established it is not only assuming a technology but assuming that human exposure can be managed. And that is hubris in its purest form.

Suit-Vehicle Interoperability and Interface Risk

Suit risk becomes especially serious at interfaces. A suit must work with the lander, airlock or cabin architecture, communications systems, tools, rovers, mobility systems, emergency procedures, and crew timelines. Interoperability is not just a physical-fit issue; it includes pressure compatibility, umbilicals, ingress and egress clearances, donning and doffing procedures, dust mitigation, emergency return, communications, thermal performance, and crew workload.

If the suit and vehicle mature on different timelines or under different assumptions, the interface can become a survival bottleneck. A suit may be technically capable in isolation but still impose unacceptable operational burdens if the lander layout, hatch geometry, egress path, dust-control approach, or emergency procedures are immature. Likewise, a lander may be technically capable

---

[267] U.S. Government Accountability Office, *NASA Artemis Programs: Crewed Moon Landing Faces Multiple Challenges*, GAO-24-106256 (November 30, 2023), pp. 1–2, 21–22.
[268] NASA Office of Inspector General, *NASA's Acquisition of Next-Generation Spacesuit Services*, IG-26-006 (2026)

but not fully survivable if the suit cannot support contingency return, crew rescue, or off-nominal egress.

The assumption-management question then becomes: who owns the combined suit–vehicle survival claim? Axiom can own suit development, HLS providers can own lander design, and NASA can own mission approval. But the crew depends on the integrated result, which needs explicit closure—not confidence inferred from parallel progress.

Surface Operations, Human Performance, and Exposure Margins

Surface operations convert suit assumptions into human exposure. Every EVA has limits: consumables, thermal conditions, mobility, visibility, terrain, fatigue, communications, dust, tools, and timeline. The suit is not just a tool that permits exploration—it defines the boundary between controlled exposure and unacceptable hazard.

Human performance is central here. A suit that meets basic technical requirements may still constrain the crew if mobility, dexterity, visibility, or workload create operational friction. Surface missions near the lunar south pole may involve challenging lighting, terrain, temperatures, communications geometry, and task complexity. Suit readiness should therefore be evaluated in terms of mission execution, not just component performance.

Exposure margin is the amount of time and capability the crew retains after something goes wrong. If a task takes longer than expected, if a crew member tires, if a tool fails, if communications degrade, or if return to the lander is delayed, the suit becomes the immediate survival system. xEVA assumptions must therefore be tied to operational timelines and contingency cases rather than treated as separate certification artifacts.

Assumption Gaps and Crew Survival Consequences

The crew-survival consequence of unresolved suit assumptions is direct: without a capable suit, the surface mission cannot safely execute. Without a compatible suit, the lander cannot fully support crewed surface operations. Without adequate suit margin, surface tasks may be reduced, delayed, or performed under narrowed contingency options. The suit is one of the clearest examples in *Artemis* of a subsystem whose readiness determines whether the architecture's human objectives are credible.

The OIG spacesuit report makes the assumption gap visible: NASA needs next-generation spacesuits, one provider remains active, and readiness for *Artemis* lunar surface operations remains a challenge. These facts do not mean the campaign cannot recover—they mean suit assumptions must be treated as active, survival-critical assumptions until they are closed by evidence.

9.7 Surface Survival and Return-to-Orbit Dependencies

Surface survival is not a single-system problem in *Artemis*—it is an infrastructure-dependent condition shaped by the combined maturity of power, communications, mobility, logistics, habitation, human-systems engineering, and operational support. A crew on the lunar surface relies on an ecosystem, not an element. The strength of that ecosystem determines how long the crew can

remain safe, how far they can operate from a refuge, how many contingencies the architecture can absorb, and how quickly a problem becomes unrecoverable.

This interconnectedness is what makes surface survival uniquely vulnerable to infrastructure mismatch. If the campaign assumes that power, comms, mobility, logistics, or habitation capabilities will be ready—and they arrive late, at reduced performance, or with untested interfaces—then surface design may exceed the actual support environment. The result is a crew that arrives inside an architecture whose documents describe sustained presence while the deployed infrastructure still only supports sortie-level operations. Surface survival therefore becomes the point where optimistic infrastructure assumptions translate directly into human consequence.

The Infrastructure Dependency Cascade introduced earlier in the paper returns here as a survival determinant. Upstream assumptions about power availability, communications coverage, mobility range, and logistics depth do not only shape science goals or mission timelines—they define the conditions under which a crew can remain safe when something goes wrong. A surface mission becomes survivable only if its dependency chain is real, reachable, and mature in the configuration the crew encounters. Anything less turns infrastructure assumptions into exposure pathways.

9.7.1 Surface Survival as an Infrastructure Dependent Condition

Surface survival is an infrastructure-dependent condition. Surface missions do not rely on a single survival system—they depend on an ecosystem of power, communications, mobility, logistics, habitation, human-systems engineering, and operational support. The maturity of that ecosystem determines how far, how long, and how safely crews can operate.

Surface survival is then vulnerable to infrastructure mismatch. If the campaign assumes that power, communications, mobility, logistics, or habitation capability will mature on time—and they arrive late or immature—then surface-mission design may exceed the actual support environment. The crew may arrive in an architecture whose paperwork describes a sustained presence but whose deployed infrastructure still supports only narrow sortie-level operations.

Upstream assumptions that shape science objectives or mission timelines also shape how long crews can remain safe when conditions degrade, especially if infrastructure is treated as a structural driver of assumption propagation.

Power, Communications and Mobility as Survival Enablers

Power, communications, and mobility should also be treated as survival enablers. Power sustains habitats, thermal control, communications, instruments, rover charging, suit support, and emergency response. Communications enable command, telemetry, medical consultation, navigation support, and anomaly resolution. Mobility expands operational range but also determines whether the crew can return, relocate, rescue, or be rescued.

These systems are often described as enablers or productivity multipliers, but for crew survival they are also contingency assets. A rover is not only a science mobility platform, but it may also determine

whether the crew can return to the lander after a suit issue or terrain delay. Communications are not just data pathways—they determine whether the ground can help diagnose anomalies, update procedures, and support time-critical decisions. Power is not only infrastructure, it is the foundation of thermal control, shelter, charging, and emergency response.

Surface power is particularly important because sustained lunar presence requires more than short-duration sortie capability. A campaign may plan to build advanced surface power, including fission systems, but survival analysis must distinguish between planned, demonstrated, deployed, and mission-available capability. A power system should not support survival confidence until it exists in the configuration, location, and operational state required by the mission.

EVA Exposure, Surface Timelines and Recovery Margins

Surface timelines convert infrastructure assumptions into exposure. Every minute outside a protected environment consumes suit resources, human stamina, communications support, thermal margin, and operational attention. The longer or more complex the surface mission, the more survival depends on infrastructure that reduces exposure or expands recovery options.

EVA exposure becomes especially consequential when timelines are compressed by cadence, pressure, or mission ambition. If the campaign expects more frequent missions and expanding surface objectives, then each surface timeline must be evaluated for whether crew tasks, suit endurance, rover availability, communications coverage, and return paths remain credible under off-nominal conditions. A timeline that works in a nominal storyboard may still fail as a survival plan if it provides little margin for delay.

Recovery margin is the difference between a problem and a catastrophe. If a rover fails, if a suit alarm occurs, if terrain slows movement, if communications degrade, or if the crew must return early, the surface architecture must preserve options. The campaign should therefore identify which surface assumptions create recovery margin and which consume it.

Return-to-Orbit as a Conditional Capability

Return-to-orbit is sometimes treated as the natural closing step of a lunar surface mission. In assumption-management terms, it is a conditional capability. The crew can return to lunar orbit only if the ascent system is healthy, the crew can reach it, the lander can support pre-ascent operations, the trajectory and rendezvous geometry remain valid, communications and navigation are adequate, and Orion or another receiving system is available and configured for handoff.

This makes HLS ascent one of the most important survival assumptions in *Artemis*. It is the bridge between surface survival and Earth return. If ascent fails, if rendezvous fails, or if the crew cannot transfer, Orion's Earth-return capability may remain technically intact but become unreachable. This is why the OIG finding that NASA lacks current rescue capability for a stranded crew is so consequential. The return path is not just a mission step—it is the crew's escape from the surface architecture.

Return to orbit should therefore be treated as an assumption cluster: ascent reliability, abort modes, loiter capability, rendezvous geometry, navigation, communications, crew procedures, and receiving-vehicle readiness. The cluster should be reopened whenever the landing profile, lander design, surface duration, crew tasks, or staging architecture changes.

Surface Dependency Chains and Mishap Pathways

Surface mishap pathways are rarely isolated. They tend to propagate through dependency chains: a power issue affects communications; communications degradation affects navigation and decision-making; mobility loss affects return timing; suit-consumables constraints compress response time; delayed ascent leads to narrower rendezvous windows and reduced recovery options. Each link in this chain may be manageable alone, but the chain can defeat the crew if multiple assumptions were optimistic.

This is where *Artemis* differs from a single-vehicle safety case. The surface mission is a temporary ecosystem assembled from systems that mature at different rates and are governed by different organizations. A mishap may begin in one element and become unrecoverable because another element cannot provide the assumed support. This is the practical meaning of distributed assumption risk.

Radiation as a Survival Constraint

Radiation is a survival and mission-duration constraint. Radiation must be treated explicitly because it is one of the major differences between a short demonstration mission and a sustained lunar campaign. *Artemis* crews operating beyond LEO face a radiation environment that cannot be managed solely through rapid evacuation. Galactic cosmic rays, solar energetic particle events, shielding limits, mission duration, warning time, and storm-shelter availability all shape nominal planning and contingency response.

In assumption-management terms, radiation is not only an environmental hazard but also a time-dependent survival constraint. A mission may be technically sound under ordinary conditions but become unsafe if an SPE occurs during a vulnerable phase, if the crew is outside a shielded volume, if suit or rover constraints delay return, or if surface infrastructure does not provide adequate shelter. Radiation assumptions must therefore be phase-specific: transit, lunar orbit, docking, surface EVA, surface shelter, ascent, and Earth return each impose different exposure and response conditions.

The survival question is not only whether the crew remains below an aggregate dose limit, but whether the architecture provides credible means to detect, shelter, replan, and recover when radiation conditions change. If the campaign increases surface duration or mobility before sheltering, communications, and monitoring mature, radiation risk can become an architectural-level assumption rather than a medical afterthought.

Medical Contingency as a Survival Chain

Medical contingency must be made explicit because lunar operations reduce the rescue and evacuation options that exist in LEO or terrestrial environments. NASA's South Pole Safety Challenge

identifies incapacitated-crew rescue as a critical concern for south-pole EVAs, noting that rocks, craters, low gravity, unique lighting, extreme temperatures, and the availability of only one other crew member all complicate rescue efforts.

The medical event transforms itself from a health problem to an integrated-architecture problem. An injured or incapacitated crew member must be stabilized, moved, returned to the lander or refuge, supported by life support, and ultimately brought back through ascent, rendezvous, Earth return, reentry, splashdown, and recovery. Each step requires assumptions about suit access, mobility aids, crew strength, vehicle layout, communications, medical procedures, and decision authority.

Medical contingency should therefore be treated as a survival chain. The campaign should identify what injuries or incapacitation events are credible, what response time is available, what equipment exists, how one suited astronaut can assist another, and what thresholds require EVA termination, surface abort, or mission redesign. Without that chain, medical risk can remain acknowledged but not operationally owned.

9.8 Decision Authority for Survival Assumptions

Decision authority is one of the most important additions because assumption management fails if no one has the authority to act when evidence changes. *Artemis* assumptions span NASA centers, commercial providers, international partners, program offices, safety authorities, mission operations, and leadership. That structure makes it essential to define who can open or reopen an assumption, who can require additional evidence, who can hold a mission milestone, and who can declare a backup path usable.

The survival chain cannot rely on informal escalation alone. If an HLS interface, xEVAS timeline, safe-haven duration, surface-power dependency, radiation-sheltering strategy, or medical-rescue procedure becomes questionable, the campaign needs a defined authority path for reopening the assumption. Otherwise, concern can remain trapped within organizational boundaries while the mission sequence continues to advance.

This is a survival-critical requirement and not a bureaucratic detail. A clear authority structure determines whether evidence that challenges a schedule or architecture can change the mission before the crew is exposed.

This is why Section 9 reframes *Artemis* maturity around the crew rather than the architecture. The reinterpreted campaign may be more coherent, phased, and deliberate than earlier sequences, but coherence on paper does not guarantee survivability in operation. Crew survival depends on whether assumptions about abort, rescue, life support, HLS, xEVAS, radiation, medical contingency, dust, lighting, terrain, surface infrastructure, return to orbit, entry, and recovery remain valid as the campaign evolves[269].

[269] NASA, *Human Rating Requirements for Space Systems, NPR 8705.2C,* current through Dec. 16, 2028.

Sapience then dictates that assumption management is not an administrative exercise but a crew-survival discipline. In *Artemis*, the most important question is not whether the campaign can return humans to the Moon, but if it can preserve enough verified margin, authority, and recovery capability to bring them home when the architecture does not behave as expected.

Artemis must demonstrate that maturity is not proven by architecture diagrams, milestone language, or distributed progress reports.

Maturity will be proven only when the campaign can identify its survival-critical assumptions, defend them with evidence, reopen them when conditions change, and preserve the authority needed to protect the crew over the schedule.

10 Safety Governance as Assumption Control

Section 9 argued that crew survival is the hard test of *Artemis* maturity and that a campaign can appear technically coherent while still leaving the crew exposed. Especially if abort paths, rescue assumptions, life-support margins, suit readiness, HLS maturity, surface infrastructure, and return-to-orbit dependencies are not integrated into a defensible survival case. Section 10 extends that argument by asking: **who protects that survival case as the architecture changes?** Safety governance is not only a compliance function, but also an assumption-control function.

Safety governance matters because assumptions do not manage themselves. They are created in design trades, embedded in requirements, carried into hazard analyses, interpreted through verification evidence, and sometimes preserved by schedule pressure. Once an assumption becomes part of the architecture, it can continue shaping decisions even after the original rationale has changed. The role of safety governance is to prevent those assumptions from becoming invisible. It forces the campaign to ask what is being assumed, who owns the assumption, what evidence supports it, when it must be reopened, and what happens to the crew if it fails.

Assumption dysfunction is especially important here since the campaign is distributed across NASA programs, centers, commercial providers, international partners, contractors, infrastructure efforts, and mission operations. No single vehicle or organization contains the full survival case. Orion, SLS, HLS, xEVAS, Gateway-related assumptions, surface infrastructure, and mission operations each carry part of the risk picture. Safety governance must therefore operate across boundaries. It must connect local evidence to campaign-level consequences.

10.1 Safety and Mission Assurance in Exploration Programs

Traditional program safety can sometimes become centered on hazard reports, risk matrices, verification, closure, and review-board disposition[270]. These tools remain necessary—but they are not sufficient for an architecture like *Artemis*. *Artemis* risk is located inside individual components, sits in the relationships among components, the timing of transfers, the availability and maturity of

[270] NASA *Procedural Requirements 8705.2C — Human-Rating Requirements for Space Systems* (Preface P.2 note 1; ch. 1 §§1.1–1.2; ch. 2 §§2.1–2.3).

commercial landers, the compatibility of suit–lander interfaces, the survivability of surface timelines, and the ability to return from the lunar surface to Orion.

NASA policy already gives this paper an official bridge between safety governance and assumption management. NPD 8700.1F requires safety and mission-success progress to be substantiated across the lifecycle through systematic argumentation, explicit assumptions, and objective evidence; it also requires risks to crew safety and mission success to be identified and either managed to closure or formally accepted within an integrated risk-management framework (NASA, 2022/2027, NPD 8700.1F, §1.a). This supports treating assumption management not as an added analytic preference, but as part of the evidentiary logic NASA already expects in safety and mission-success governance.

For that reason, safety governance must address both hazards and assumptions. A hazard identifies what can go wrong, while an assumption explains what the campaign relies on to believe the hazard is controlled. For example, a hazard might involve crew being stranded on the lunar surface. The assumptions behind the hazard control may include HLS-ascent reliability, sufficient consumables, communications availability, crew access to the ascent vehicle, valid rendezvous geometry, Orion availability, and defined decision authority. But the moment any of those assumptions change, the hazard control will no longer mean what they originally meant.

This is why assumption control must be treated as part of safety governance rather than a separate administrative activity. In a live campaign, assumptions change when mission sequences shift, when provider schedules slip, when verification evidence is reinterpreted, when infrastructure is delayed, when policy direction accelerates, or when budget conditions alter what is realistically achievable. Safety governance must then create the discipline to reopen assumptions before they become normalized into the next milestone.

A second concern follows directly from this: safety governance is not only a function of process—it is a function of capacity. Assumption control, cross-program integration, verification interpretation, and campaign-level safety judgments require experienced personnel who can challenge optimistic interpretations, detect drift, and understand how local evidence translates into crew-survival consequences. The long-term NASA headcount data for Safety and Mission Assurance (S&MA) engineering workforce has remained effectively flat—hovering around 150 engineers for many years—even as *Artemis* complexity, distributed interfaces, and cross-provider integration demands have increased. This flat capacity constitutes a structural constraint.

Figure 13 shows that while NASA's overall civil service workforce has remained relatively stable up until 2025, the S&MA engineering workforce has remained proportionally small and has not scaled with the increasing complexity of *Artemis*[271]. At Johnson Space Center, the S&MA workforce in the NA/NC branch space consistently represents **around one percent or less** of the Center's civil service population. That percentage trend line remains nearly flat, even as *Artemis* introduces more

[271] NASA civil-service and FTE workforce/budget datasets; JSC headcount data; Internal JSC S&MA mail-code/headcount data.

distributed interfaces, more commercial integration, more cadence pressure, and more survival critical assumptions than any prior lunar program. This misalignment becomes transparent when JSC's overall civil-service footprint is used as a comparative baseline- approximately 20% of NASA—to the S&MA footprint supporting it, which remains below one percent of the Agency.

Safety governance must close cross-program assumptions and not only check subsystem hazards—subsequently this staffing decline becomes a structural risk. An architecture whose survival case spans Orion, HLS, xEVAS, Gateway, logistics, surface systems, and commercial integration requires more interpretive capacity, not less. If the number of personnel who steward the survival logic decreases while the complexity of the architecture increases, the gap is not procedural. It is mathematical.

This trend is not an indictment of the workforce—it is a signal about the load being placed upon it. A shrinking S&MA footprint must now carry a growing interpretive burden across multiple providers, evolving mission sequences, and assumptions that change faster than the infrastructure that supports them.

The figure shows NASA's total civil-service workforce as tall blue bars ranging from roughly 18,000 to over 24,000 employees across the years plotted. In front of those bars, a smaller set of orange bars shows JSC's civil-service population, consistently around 3,000 employees, which visually illustrates that JSC represents roughly one-fifth of the Agency's overall workforce.

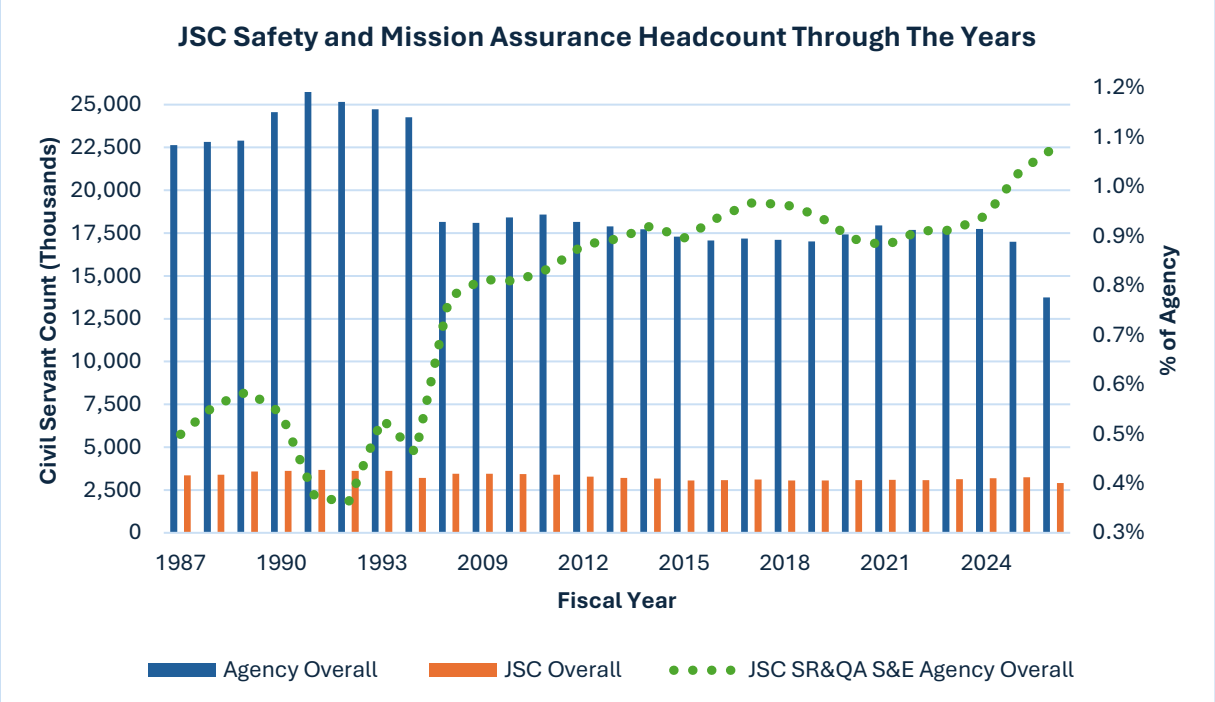


Figure 13 : JSC S&MA Headcount

Superimposed on those bars is a green trend line representing the percentage of the Agency's civil-service workforce made up of JSC's S&MA personnel. The line remains close to the bottom edge of the chart because the values are less than 1.2% across all years. The shape of the line shows slight year-to-year movement, but it never approaches the scale of either the Agency or JSC headcount. It remains a thin, low-lying curve that visually contrasts with the height of the blue and orange bars.

What this layout makes clear is the proportional mismatch: the figure juxtaposes a large and relatively stable Agency workforce, a sizeable JSC workforce that consistently constitutes around 20% of NASA, and an S&MA footprint at JSC that remains below 1% of the agency total. The disparity between the scale of the bars and the scale of the SMA percentage line signals a persistent structural constraint—the workforce responsible for safety governance is small relative to the organizational mass and mission complexity it must oversee.

NASA's financial directives support the claim that budget execution is part of the architecture, not merely its funding background. NPD 9050.3G requires NASA to maintain administrative control of funds under OMB A-11 and the Antideficiency Act, and to restrict obligations and expenditures from each appropriation or fund account according to apportioned or available amounts and NASA budget-execution requirements (NASA, 2024, NPD 9050.3G, §1). NPR 9470.1A further organizes budget execution around operating/execution plans, funds distribution and control, commitments, obligations, expenditures, and reporting (NASA, 2024, NPR 9470.1A, Ch. 2-5). This supports the argument that architecture can change conceptually faster than it can change in execution.

This is why safety governance must include explicit assumption stewardship: without sufficient workforce, assumptions do not get reopened; they get preserved by necessity.

10.2 Cross-Program Safety Integration and Assumption Stewardship

Within Safety and Mission Assurance at Johnson Space Center in Houston, there sits a division with the mail code "NC". NC division is known as the Space Transportation Systems Division that provides system safety, reliability, and risk analysis for Human Space Flight Programs and works with program offices to reduce risk by providing technical assessments, analytical services and guidance on SMA requirements throughout the program and project lifecycle[272]. NC has an additional two branches under its wing:

- **NC2: Program SMA Integration Branch**
  NC2 provides program-level S&MA support by defining requirements, advising program offices, leading key safety reviews, supporting Systems Engineering and Integration (SE&I), test and verification, software safety, flight operations, and crew-survival work, while also performing Probabilistic Risk Assessments (PRA) and reliability analyses using fault trees, event trees, Monte Carlo methods, and cross-industry data to assess Loss of Crew, Loss of Mission, and system performance**.**
- **NC3: Space Systems Safety Branch**

[272] Official JSC S&MA organization chart; internal NC division role description; NASA organizational documentation.

NC3 provides S&MA requirements, assess vehicle designs, approve testing and certifications, and support launch and on-orbit operations across spacecraft systems including structures/TPS, Environmental Control and Life Support Systems (ECLSS), Guidance, Navigation and Control (GNC), power, propulsion, pyrotechnics, avionics, and the crew/service module. Responsibilities include verification and validation, subsystem risk and hazard analyses - including Failure Mode and Effects Analysis (FMEA), Critical Items List (CIL), certification, life-limit reviews, Launch Commit Criteria (LCC)/flight rules/procedure assessments, and evaluation of change requests, waivers, in-flight anomalies, and Problem Reporting and Corrective Action (PRACA) items.

Both teams within the NC branch share a strong passion for safer human spaceflight and bring a remarkable depth of experience from working across NASA centers and partner organizations. They combine technical expertise with operational judgment, enabling them to anticipate risks, challenge assumptions, and strengthen the safety posture of every mission they support. This mission-driven culture and cross-program insight allow them to see risks early, ask the right questions, and help shape safer, more resilient architectures.

The NC3 Vehicle Systems Engineers (VSE) are especially well positioned to support assumption control because their role sits at the intersection of vehicle knowledge, safety judgment, mission assurance, and systems integration[273]. They are not only reviewers of finished products, in the framework of this paper, they function as assumption stewards: engineers who identify where technical claims, safety evidence, operational expectations, and crew-survival consequences intersect.

This is particularly important because assumption drift often occurs between disciplines. A design team may understand the vehicle; a provider may understand its subsystem; mission operations may understand the procedure; program leadership may understand the milestone. Safety and mission assurance must understand what happens when those pieces are combined and the crew depends on the integrated result. NC3 VSEs help translate local technical evidence into campaign-level survival meaning.

Their value is not only in asking whether a requirement was verified, but whether the verification still supports the current mission assumption. A test may be valid for one configuration but less meaningful after a mission sequence changes. A hazard control may be adequate for one surface duration but insufficient for another. A safe-haven concept may appear credible until the crew's actual location, consumable stocking, access, or timeline is examined. VSEs help keep those distinctions visible.

In practice, this means asking questions such as:

- What assumption is this risk carrying?
- Which mission phase depends on it?

[273] NASA, *NASA Safety Culture Handbook*, NASA-HDBK-8709.24 (2021), pp. 7–13.

- What evidence closes it?
- Does the evidence still apply under the revised architecture?
- Who owns the assumption if it crosses program or provider boundaries?
- What is the impact on the crew if the assumption does not hold?

These questions turn safety governance from passive review into active assumption management.

NPR 8621.1D provides a formal NASA anchor for treating near misses and close calls as assumption-management signals. The directive covers mishap and close-call readiness, response, investigation, findings, recommendations, corrective-action plans, monitoring and closeout, lessons learned, and evidence retention (NASA, 2020/2028, NPR 8621.1D, Ch. 1-6). The implication is that anomalous evidence should not be treated as isolated event data only; it should feed back into the assumptions, controls, and lifecycle products that allowed the condition to occur or persist.

10.3 Hazard Analysis and Verification Interpretation

Assumption control should not occur only at major reviews. By the time a risk reaches a final readiness review, many assumptions may already be embedded in requirements, schedules, contracts, interface documents, verification plans, training products, and operational procedures. Effective safety governance must begin earlier and continue across the life cycle[274].

Formulation and architecture definition states that assumption control should identify the major campaign premises shaping the mission—assumptions about cadence, commercial readiness, lander maturity, suit availability, Gateway roles, surface infrastructure, rescue capability, and crew exposure. The goal is not to eliminate all assumptions, but to prevent major assumptions from becoming invisible[275].

Design and development stages warrants that assumption control connect these premises to requirements, interfaces, hazard controls, and verification plans. If a requirement depends on a specific landing capability, suit endurance, docking behavior, communications path, or surface-power source, that dependency must be traceable. If the supporting architecture changes, the requirement must be reviewed.

Verification and validation requires that assumption control should guard against overinterpreting evidence. A successful test does not automatically close every assumption associated with that system. The campaign must distinguish between component verification, integrated verification, operational demonstration, off-nominal demonstration, and crew-survival confidence. This is key for *Artemis* since many of its most important risks emerge at interfaces and during transitions between mission phases.

Operations and readiness reviews needs assumption control to ask whether the survival case remains current. The question is not merely whether each item has a closure record—but whether

[274] Hawkins et al., "A New Approach to Creating Clear Safety Arguments," in *Advances in Systems Safety* (2011), pp. 3–23.
[275] NASA Langley Research Center, Understanding Assurance Cases, Module 1 pp. 3-32 for foundations/definition; combined educational series pp. 1-142.

those closures still support the mission now being flown. If an assumption has changed, the campaign should either revalidate it, restrict the mission, add controls, or formally accept the residual risk with full visibility.

This approach prevents assumptions from becoming static background logic and instead integrates them into the same disciplined lifecycle as requirements, hazards, verification, and configuration management[276].

10.4 Maintaining Confidence in Campaign Architecture

One of the most important responsibilities of safety governance is protecting the right to reopen assumptions. Historical mishaps show that risk grows when organizations become committed to prior interpretations of evidence. Once a program has invested in a schedule, configuration, or public milestone, reopening assumptions can feel disruptive, yet in human spaceflight, the ability to reopen an assumption may be the difference between learning and normalization of deviation.

This issue is especially important because the *Artemis* campaign is being asked to move with urgency. The revised mission sequence, annual-cadence goals, commercial integration, and surface-infrastructure ambitions all increase the need for disciplined learning. They also increase the pressure to preserve assumptions once they are embedded in the plan. Safety governance must counterbalance that pressure.

NC3 VSEs can support this by identifying **assumption triggers**. A trigger is a condition that automatically requires reassessment. Examples include a mission-sequence change, a provider schedule slip, a failed or partial test, a design-configuration change, a new hazard, a changed landing site, a reduced consumables margin, a suit delay, a rescue limitation, or a shift in surface-infrastructure availability. When the trigger occurs, the question cannot be limited to whether reopening the assumption is convenient. The trigger itself should require review.

This is where safety governance becomes more than oversight—**it evolves into a control mechanism against interpretive drift**. If an assumption is allowed to survive every change without review, the campaign is no longer managing risk; it is preserving belief.

10.5 Assumption Control Governance

Assumption control depends fundamentally on decision authority. It is not enough to identify assumptions if no one has the authority to act when they weaken. The campaign needs clear pathways for elevating assumption concerns, holding closures, modifying flight rules, restricting operations, or delaying a mission when the survival case no longer supports the planned activity.

Flight rules are one of the places where assumptions become operational. A flight rule should not only specify what action will be taken but also make the underlying assumptions visible. This should include assumptions about communications paths, vehicle state, crew timelines, consumables

[276] Denney et al. (2015), "Safety Case Patterns: Theory and Applications," NASA Ames Research Center / FAA REDAC presentation, NASA/TM-2015-218492, pp. 1–50.

margins, or available abort options. If those assumptions change, the rule may appear valid even though the condition that made it safe no longer exists.

Dissent is also a critical component of assumption control. Safety governance must preserve a protected path for technical disagreement when evidence does not support the preferred interpretation. Assumption drift is often subtle: it may not present as a clear violation but as discomfort with how evidence is interpreted, how test results are generalized, how provider claims are accepted, or how backup paths are described. NC3 Vehicle Systems Engineers and S&MA personnel must be able to raise these concerns without being treated as schedule obstacles.

A mature campaign does not eliminate disagreement—it uses it to test whether assumptions remain defensible.

For assumption control to be useful, it must produce artifacts that programs can actually use. The goal is not more paperwork; it is clarity. Assumptions must be visible, traceable, challengeable, and updatable. Several products—tracking lists, closure evidence summaries, trigger logs, interface-assumption maps—can strengthen *Artemis*'s assumption management. These products should be integrated into existing systems engineering, risk, hazard, verification, configuration management, and readiness review processes. Assumption management should not become a parallel bureaucracy but should operate as an explicit layer within processes that already exist.

Safety governance is the mechanism that keeps assumption control alive after an architecture is approved. This role is essential in *Artemis* since the campaign is distributed, evolving, and survival-critical. Assumptions about HLS, xEVAS, Gateway, surface infrastructure, rescue capability, cadence, and return-to-orbit cannot be allowed to migrate silently from one mission sequence to another. They must be owned, traced, tested, reopened, and retired when evidence no longer supports them[277].

NC3 VSEs are important in this process because they connect technical detail to crew consequence. Their role is more than verifying that documentation exists or that a requirement is marked closed—they ask whether the closure still means what the campaign believes it means[278]. That question sits at the heart of assumption control.

The central argument of this unified section is that safety governance is an active engineering discipline, not a final compliance checkpoint. For *Artemis*, safety governance is how the campaign protects itself from inherited assumptions, partial verification, distributed ambiguity, and confidence preserved by schedule pressure. If Section 9 shows that crew survival is the hard test of *Artemis* maturity, Section 10 shows that safety governance is the discipline that keeps that test honest.

[277] NASA, *NASA Systems Modeling Handbook for Systems Engineering*, NASA-HDBK-1009A (2025), pp. 5, 8, 47–49, 73–78, 81–82.
[278] NASA, *Human -Rating Requirements for Space Systems*, NPR 8705.2C, current through Dec. 16, 2028.

11 Discussions: Momentum is Not a Survival Strategy

Artemis is not immature because it lacks ambition. *Artemis* is immature because ambition is being misinterpreted as readiness. The campaign is being asked operate at a strategic tempo it is not yet structurally prepared to sustain, while its survival case still rests on distributed systems, volatile budgets, unresolved interfaces, commercial timelines, institutional memory, and assumptions that are not always owned by the same organizations that relied on them[279]. **That is the core danger.**

> ***Artemis* is not just a collection of vehicles- it is a campaign architecture.**
>
> **Campaign architectures fail differently than vehicles do.**

This paper has argued that assumptions are not passive. They are created, inherited, embedded, normalized, defended, and sometimes forgotten. In human spaceflight, assumptions become especially dangerous when they move from analysis into architecture without clear ownership. Once embedded, they begin shaping decisions faster than formal review mechanisms can respond. *Artemis* now stands at this architectural threshold. It is not only moving from development to operation but from promise to consequence.

The essential question is not whether *Artemis* is inspiring — it is. The question is not whether it is technically impressive — it is. The question is not whether NASA, its scientists, its contractors, its international partners, and its commercial providers contain extraordinary talent — they do. **The question is "can the architecture protect a crew when the campaign becomes stressed, delayed, underfunded, politically pressured, technically incomplete, or operationally degraded?"**

This is the discussion *Artemis* requires: not applause for motion, but accountability for survival. The historical cases examined in this paper showed that human spaceflight failures are rarely caused by ignorance alone. *Apollo* 1, *Challenger*, and *Columbia* were not failures of ambition. They were failures of interpretation, integration, authority, and assumption control. Warning signs existed. Technical concerns existed. Organizational signals existed. But these signals did not surface with sufficient force to reshape the architecture before it shaped the outcome before the architecture consumed the crew. That is the pattern *Artemis* must refuse to inherit[280].

Artemis is not under modern dangers of repeating *Apollo* 1, *Challenger*, or *Columbia* exactly — it will not. The vehicles are different, the contractors are different, the political era is different, the acquisition model is different, the architecture is different. The deeper pattern has survived under new names and forms: fragmented evidence, schedule pressure, normalized uncertainty,

[279] NASA, *NASA Safety Culture Handbook*, NASA-HDBK-8709.24 (2021), pp. 7–13.
[280] *Columbia* Accident Investigation Board Report, Volume I (2003); Report of the Presidential Commission on the Space *Shuttle Challenger* Accident, Volume I (1986); NASA *Systems Engineering Handbook* (NASA/SP-2016-6105 Rev. 2, 2016); NASA *Procedural Requirements 7120.5F — NASA Space Flight Program and Project Management Requirements*.

ambiguous ownership, and any survival case that looks stronger in documentation than it is in operation.

The pattern is not the vehicle. The pattern is the drift.

That is why Section 11, and the following discussions, must be more than a summary. It cannot retreat into weak prose, passive voice, or language so carefully softened that it becomes analytically irrelevant. Not by fear of the consequences of telling the truth.

> *Every person that has been a part of Artemis, whether contractor or civil servant, whether their tenure was brief or lifelong, has earned analytical honesty. They earned it through blood, sweat, tears, pride, sacrifice, and service.*
>
> *This section is for them:* ***for those who continue to fight for science, for space, for the agency, for the future, and for the crews whose survival will depend on whether Artemis becomes as accountable as it is ambitious.***

Section 11 therefore must be ambitious in its objective: it must name the pressure points. *Artemis* is more than a technical program. It is a national strategy, a budget construct, a political symbol, a commercial marketplace, a partnership framework, a workforce challenge, a safety case, and a moral obligation. Full Stop.

If these pieces do not mature together, the campaign will not mature at all. It will move forward with unresolved assumptions hidden beneath the language of process and the perception of advancement.

Momentum is not maturity. Momentum is not integration. Momentum is not survival.

11.1 *Artemis* Maturity and the Problem of False Confidence

Artemis risks being judged by the wrong evidence. A program will look mature because its successful missions, funded line items, impressive hardware, international agreements, commercial contracts, visible cadence, and political momentum are all highly visible. None of those signals by themselves prove that the integrated architecture is mature[281]. They prove activity. They prove commitment. They prove progress. They indicate performance, not architectural robustness.

They do not prove survivability.

A successful *Artemis* II is meaningful. A working Orion is meaningful. SLS performance is meaningful. HLS development is meaningful. Suit maturation is meaningful. Gateway and partner commercial contributions are meaningful. But none of this guarantees that the architecture itself is maturing. The

[281] Lempert, "Robust Decision Making (RDM)," in *Decision Making Under Deep Uncertainty: From Theory to Practice*, ed. Vincent A. W. J. Marchau et al. (2019), pp. 23–51.

campaign, however, will not mature simply because its parts are moving in parallel. In this architecture, individual progress can coexist with integrated fragility. That is the danger: each element can defend its own maturity while the system-level survival case remains incomplete. **A program can appear mature simply because it is in motion. That is the trap.**

Motion disguises fragility. Cadence disguises schedule compression. Funding disguises uncertainty. Requirement closure can disguise unresolved operational consequence. Political commitment can disguise technical exposure. Commercial procurement can disguise accountability gaps. *Artemis*, then, needs to be judged not by whether it appears organized around the future, but if it can protect a crew when that future arrives damaged, delayed, underfunded, misaligned, or technically degraded.

False maturity is dangerous because it is comfortable. It enables the program to emphasize milestones instead of margins, plans instead of readiness, and agreements instead of authority. It allows organizations to say that their piece is on track while the integrated architecture remains full of seams. It emboldens success in one phase to be used as emotional proof that later phases are equally mature. But *Artemis* II does not validate *Artemis* III, and *Artemis* III does not validate *Artemis* IV. A lunar flyby does not prove a surface campaign, and a surface campaign does not prove a sustainable lunar architecture.

Progress is not maturity. Progress is only maturity when it reduces the unknowns that can kill the crew.

The significance of this distinction is being addressed because *Artemis* is a campaign, not a single event. A campaign must survive recurrence effects. It needs to be able to survive workforce turnover, budget instability, contractor transitions, configuration drift, delayed interfaces, technology gaps, launch cadence pressure, and the accumulation of small compromises. A one-time success can occur inside an immature system. **Campaign maturity requires that success become repeatable without eroding safety margins each time the schedule advances.**

The false maturity signals must therefore be rejected. Funded hardware is not architectural maturity. Requirement closure is not assumption control. Contractor awards are not evidence of operational readiness. Political urgency is not technical evidence. Cadence is not resilience. A successful mission does not close the next mission's survival case.

The *Artemis* maturity question then moves from "*Can the campaign proceed?"* to "*Can the campaign still protect the crew when proceeding becomes difficult?"*

11.2 Crew Survival is the True Measure of Campaign Maturity

Crew survival is not one metric among many. It is the *only* metric that strips away program theater and organizational self-approval. It is indifferent to an architecture that is politically elegant, commercially innovative, internationally impressive, or budgetarily convenient. It asks whether human beings can live when the plan breaks. If *Artemis* cannot answer that question across launch,

transit, lunar orbit, descent, surface stay, ascent, rendezvous, contingency, loiter, return, reentry, and recovery, then *Artemis* is not mature. It is simply advancing.

If programs balk at crew survival as the measure of *Artemis* maturity, the reaction reveals a deeper governance problem. Human spaceflight exists on the premise that exploration risk can be understood, bounded, communicated and governed with discipline. A campaign that measures success by cadence, budget execution, milestone completion, or architectural expansion while treating crew survival as a secondary or uncomfortable standard has inverted the purpose of the enterprise. Crew survival is not an emotional appeal or an external criticism; it is the hard test of whether the architecture works when the assumptions fail. If *Artemis* can appear successful while unresolved survival dependencies remain distributed, then the problem is not that the survival standard is too severe. It is that the program has accepted that crew survival is optional.

This framing does not require *Artemis* to eliminate risk. No human spaceflight campaign can do that. Rather, it requires the campaign to demonstrate that the risk is being governed through explicit ownership, verified assumptions, integrated contingency planning, and accountable decision authority. Crew survival therefore becomes a measure of architectural integrity and not just crew protection. Survival is not a narrow endpoint; it is the clearest measure of whether *Artemis* has matured from a collection of mission elements into a governed human spaceflight exploration mission.

This is the moral center of the *Artemis* discussion. Human spaceflight is not only a demonstration of enduring engineering capability. It is a decision to place human beings inside a machine, send them into an environment that offers no forgiveness, and forces dependence on architecture, operations, training, logistics, software, hardware, judgment, and governance to bring them home. That decision carries a burden that cannot be satisfied by optimism or schedule confidence. Crew survival is the true measure of *Artemis* maturity because it is the only measure that refuses to flatter the architecture.

Survival asks the questions that program momentum prefers to soften:

- What happens if HLS is late?
- What happens if the suit is less mature than the landing timeline assumes?
- What happens if Gateway is paused, altered, reduced, or bypassed?
- What happens if Orion is healthy, but the surface architecture is not?
- What happens if a crew must wait longer than planned?
- What happens if ascent is delayed?
- What happens if communications degrade? Or if there is partial communications loss?
- What happens if a commercial provider meets the contractual milestone, but the integrated operational consequences remain unresolved?
- What happens if every organization can defend its local decision, but no one can defend the whole survival case?

These are not pessimistic questions. They are human spaceflight questions. They are questions that separate confidence from theater. *Artemis* does not need a safety narrative that only works when the architecture behaves. It needs a survival case that holds when the architecture is damaged, late, constrained, partial, or wrong.

A program that resists crew survival as a maturity measure is telling on itself. It may not be indifferent to crew safety — that is not the point. The point is more subtle and more dangerous: **resistance suggests the program may be more comfortable proving that the mission can proceed than proving that the crew can survive when it cannot.**

If crew survival is treated as too harsh a maturity standard, then the architecture is not yet mature enough to carry crew.

This is not emotional language; it is architectural language. A system that can only demonstrate success under normal conditions is not mature. A system that cannot clearly identify abort paths, rescue assumptions, contingency timelines, authority chains, consumable margins, communications dependencies, medical constraints, and degraded-mode survival options is not mature. A system that depends on multiple organizations but cannot show who owns the integrated consequence is not mature.

The survival case then must be made explicit. It must be owned. It must be tested. It must be reopened when budgets change, schedules move, interfaces shift, program priorities evolve and when mission mode, profile, or risk posture shifts. It cannot be buried under certification language or dispersed across element-level reviews. The crew will not experience *Artemis* as separate budget lines. The crew will experience *Artemis* as one integrated reality.

The crew does not fly in an eight-page PowerPoint architecture. The crew flies in the consequences of every assumption the architecture failed to control.

11.3 Budget Authority, Architecture Authority, and the Governance Hole

The FY 2027 budget posture exposes one of the most dangerous contradictions inside *Artemis*: The United States is demanding a faster, more ambitious lunar campaign while narrowing the institutional base that must make the campaign safe. That is not as simple as a budget issue; it is an architecture issue, a governance issue, and therefore a crew survival issue.

NASA's FY 2027 budget request lists the agency at $18.829 billion, down from $24.438 billion in FY 2026. Concurrently, Exploration rises from $7.783 billion in FY 2026 to $8.514 billion in FY 2027. Science falls from $7.250 billion to $3.894 billion. Space Operations falls from $4.175 billion to $3.047 billion. Space Technology falls from $920.5 million to $624.3 million. Aeronautics falls from $935 million to $609.5 million. OSTEM Engagement is reduced from $143 million to $0 in the request. Safety, Security, and Mission Services falls from $3.000 billion to $1.999 billion. Construction and Environmental Compliance and Restoration falls from $185.3 million to $100.6 million. These numbers are not small variations; they are structural shifts.

This is the contradiction in plain language:

***Artemis* is being protected while much of the agency ecosystem around it is being cut.**

***Artemis* is being fed while the ecosystem that makes it governable is being starved.**

The Working Families Tax Cut Act (WFTCA) does not erase this contradiction. It provides real money, but it is not $10 billion per year. NASA's budget documents identify $9.995 billion in special mandatory appropriations available until September 30, 2032, with planned obligations of $575 million in FY 2025, $4.010 billion in FY 2026, $2.110 billion in FY 2027, $2.025 billion in FY 2028, and $1.275 billion in FY 2029. NASA describes the funding as supporting *Artemis* missions, Mars missions, Moon-to-Mars programs, space operations, and construction/environmental compliance activities.

NASA's own summary states that the WFTCA supplements funding for *Artemis*, ISS continuity, and critical exploration infrastructure. This includes SLS and Orion through *Artemis* V, advanced communications, ISS cargo resupply, the US Deorbit Vehicle, and facility modernization. This means that the WFTCA funding is not imaginary, symbolic, or irrelevant. It is a real supplemental brace.

But it is still a brace. It has not proved that the architecture is structurally sound. A supplemental appropriation can buy time. It cannot buy maturity.

Money can protect selected lines. It can keep hardware moving. It can keep delays from becoming cancellations. It can accelerate a political objective. But money does not preserve institutional memory. Money does not automatically produce accountability. Money does not automatically close the survival case. If the money points one way and the architecture points another, the campaign does not become stronger. It becomes harder to govern.

A presidential mandate can set the national space policy, direct NASA's strategic priorities, and shape the president's budget request, but it cannot by itself override money Congress has already appropriated and legally directed. Once Congress enacts an appropriations law, NASA must execute within the purpose, time, amount, transfer, programming, and notification limits Congress attached to those funds. Which in turn means only one thing: Presidential direction can accelerate, reprioritize or reframe *Artemis* strategy but it does not erase congressional control over appropriated funds**.** Whether Congress has funded Gateway, SLS, Orion, HLS, or other *Artemis* elements through appropriations, NASA must execute within the legal limits of those appropriations unless Congress later changes the law or approves a permitted reprogramming, transfer or rescission. **This is critical: *Artemis*'s architecture authority and budget authority are not always being held by the same set of hands.**

The result is not that presidential policy "beats" congressional funding, or that congressional funding automatically resolves the architecture risk. The result is structural tension: the White House may set the destination and urgency, Congress may protect or fund specific architecture elements, and

NASA must integrate both into a safe and executable campaign. That is the governance hole: **Who owns the architecture when money and strategy point in different directions?**[282]

Budget authority and architecture authority are not the same thing. Budget authority decides what is funded. Architecture authority must decide what is coherent, survivable, integrated, and ready. When those authorities align, a program can move with confidence. When they diverge, the program can move with force, but not necessarily with wisdom.

That sentence cannot be softened. It is the question. If the budget says accelerate *Artemis*, protect Exploration, supplement SLS and Orion, adjust Gateway, shift toward commercial systems, and increase cadence, then someone must also own the integrated consequence of those decisions. Someone must own what happens when a funding choice changes a technical assumption. Someone must own what happens when political cadence changes a safety margin.

Surely, ***someone*** must own what happens when a commercial transition changes rescue logic, interface control, verification, evidence, or operational authority?

If no one owns that integrated consequence, then *Artemis* does not have an architecture. It has a collection of funded intentions. **A campaign cannot be governed by budget momentum alone.**

11.4 Cost, Cadence, and the Survival Chain

The same assumption control logic applies to cost. A campaign cannot claim integrated maturity if its costs are evaluated around individual budget lines while crew survival depends on the performance of the full architecture. SLS, Orion, EGS, HLS, xEVA, Gateway or no Gateway operations, LTV, pressurized rovers, communications, power, logistics, rescue, and return are not separate from the crew once the mission begins. They form the funded survival chain— the architecture the crew actually depends on. Therefore, cost estimation is not only a financial exercise; it is a way to test whether the architecture being promised is the same as the architecture being funded, verified, and protected.

Cost estimates for a new *Artemis* cadence should not be presented as a single fixed number. Cost per launch depends on what the estimate is allowed to include. A narrow estimate can price the SLS/Orion launch stack only. A broader estimate can include the systems required to make that launch meaningful, such as a lunar surface mission. A campaign-level estimate can allocate annual fixed costs across the number of missions NASA expects the architecture to support. The difference between these estimates matters because the paper's core question does not doubt whether *Artemis* can launch. But rather: Can *Artemis* repeatedly protect crews through a changing and distributed architecture?

The most defensible starting point is the NASA OIG SLS/Orion operating-cost estimate. The OIG projected the cost to fly a single SLS/Orion system through at least *Artemis* V at approximately $4.1 billion per launch at roughly one mission per year. That estimate included about $1.0 billion for

[282] NASA, *NASA Governance and Strategic Management Handbook*, NPD 1000.0C, pp. 3, 10, 53–55.

Orion, $300 million for the ESA Service Module, $2.2 billion for SLS, and $568 million for launch-related EGS costs. The OIG also stated that this launch-cost estimate did not include prior development spending or next-generation items such as Exploration Upper Stage, Orion docking systems, or Mobile Launcher-2. That makes $4.100 billion a strong legacy launch-only baseline—not a complete lunar-surface mission estimate.

The proposed new *Artemis* cadence makes this analysis unavoidable. NASA's 2026 architecture update states that the agency is increasing the cadence of *Artemis* missions by standardizing the SLS configuration, adding an additional mission in 2027, and undertaking at least one surface landing every year thereafter. NASA's March 2026 HLS update states that beyond *Artemis* V, the agency intends to incorporate more commercially procured and reusable hardware, with an eventual goal of landings every six months as capabilities mature. These cadence claims must be evaluated against cost, production capacity, launch and processing throughput, HLS readiness, suit availability, surface mobility, logistics, rescue assumptions, and return capability. For this paper, the recommended formula is:

cadence-normalized cost per crewed lunar mission = mission-specific transportation cost
+ mission-specific surface-system costs
+ allocated annual fixed costs
+ allocated architecture-transition costs

Each term is significant:

- **Mission-specific transportation cost**
  The cost of transporting crew and cargo—SLS, Orion, EGS processing, and associated expendable elements. This is the price of ascent, transit, and return, but not of surface survival systems.
- **Mission-specific surface-system costs**
  The costs that make a landing survivable: HLS services, suit readiness, surface mobility, power systems, comms, logistics, and any mission-unique surface capability.
- **Allocated annual fixed costs**
  The ongoing cost of maintaining the architecture—workforce, facilities, ground systems, safety infrastructure, operations—that must be distributed across the mission cadence.
- **Allocated architecture-transition costs**
  Costs associated with evolving the architecture—Gateway configurations, next-generation stages, new surface assets, commercial transitions, and other structural changes that affect both cost and survival.

This formula is important because cadence does not automatically make *Artemis* cheaper. Cadence reduces average cost only if the architecture is designed around reusable assets, stable fixed-cost infrastructure, mature supply chains, repeatable operations, and reliable integration. If major elements remain expendable, late, or frequently re-baselined, a higher cadence can increase pressure faster than it reduces cost. That is the assumption-management issue: the architecture may promise repetition before the cost model proves that repetition is sustainable.

The cost gap is therefore not separate from the governance gap; it is one expression of it. *Artemis* can be described as a campaign architecture, but its costs are still publicly visible as separate budget lines, contract scopes, and fiscal-year decisions. That separation creates an analytical risk. If the question is *"How much does one launch cost?"* the answer may stop at SLS, Orion, and EGS. If the question is *"What does it cost to safely complete and repeat a crewed lunar surface mission?"* the answer must include HLS, suits, surface mobility, communications, power, logistics, rescue capability, return-to-orbit capability, and architecture-transition costs. It is that second question that matters most for crew survival.

This is where cost becomes a maturity test. A mature *Artemis* cadence would show that the funded architecture, the planned architecture, and the survival architecture have converged into one architecture. If they are not the same, cost becomes another path through which assumptions drift. A program can appear affordable by pricing only the launch vehicle and spacecraft while leaving lander readiness, suit availability, surface mobility, Gateway configuration, rescue concepts, or commercial-transition risk outside the launch estimate. That may be acceptable for accounting. It is insufficient for assumption management.

The governing finding is direct: *Artemis* cannot use cadence as evidence of maturity unless the cost model demonstrates that each repeated mission can fund, integrate, verify, and protect the full crew-survival chain. A yearly landing cadence and the later six-month landing target require more than faster scheduling. They require scaled production capability, ground-system throughput, lander availability, suit readiness, surface logistics, system integration, contingency capability, and the cost of repairs. If those elements do not scale together, cadence becomes a pressure source rather than a maturity signal.

This also reinforces the earlier governance question: **who owns the architecture if money and strategy point in different directions?**

The same question applies to cadence: **who owns the cost model if the mission is described as integrated, but the funding, contracts, and safety evidence remain distributed?**

Until the program answers that question clearly, cost-per-launch estimates should be treated not as final numbers but as diagnostic tools for identifying where architecture, budget, and crew survival are no longer aligned. These estimated values should therefore be understood as part of a cost-estimation method, not as official NASA cost-position figures. The purpose is diagnostic, not declarative. In that spirit, the safest framing is that this paper uses public NASA budget lines and

NASA OIG launch-cost estimates to develop a cadence-normalized proxy estimate—an analytical tool for identifying whether architecture, budget, and crew survival remain aligned.

11.5 Cadence is Pressure, Not Proof

NASA's Architectural Revision and the Ignition agenda make the cadence problem explicit. The agency states that automatic updates include standardizing the SLS configuration, changing the mission posture in 2027, and undertaking at least one lunar surface landing every year thereafter. It also describes a post-*Artemis* V shift toward commercially procured reusable hardware, initially targeting crewed lunar surface landings every six months. NASA's public release similarly states that *Artemis* III is scheduled as a 2027 Earth-orbit integrated systems test ahead of an *Artemis* IV lunar landing, with annual landings thereafter and an eventual six-month landing cadence beyond *Artemis* V.

This is not a small adjustment. This is a declaration of campaign tempo. It transforms *Artemis* from a sequence of high-stakes missions into a recurring lunar enterprise. That ambition may be strategically valuable, but ambition is not evidence. A cadence can be announced long before the architecture has earned it.

Cadence is not maturity. Cadence is pressure.

Cadence can be the result of maturity, but it is not evidence of maturity by itself. **A mature architecture earns cadence by demonstrating that each mission strengthens the next one.** An immature architecture uses cadence to outrun unresolved questions. *Artemis* must be careful not to confuse the ability to schedule missions with the ability to absorb them. Annual landings and eventual six-month landings require more than launch vehicles and landers. They require production stability, launch site readiness, trained crews, flight controllers, certification evidence, consumables, logistics, suit availability, service infrastructure, communications, medical support, maintenance capability, rescue options, and governance systems that can keep all of those dependencies visible. They require a system that can recover from anomalies without converting recovery actions into unplanned schedule debt.

They require a campaign that can learn without pretending that learning is free.

A cadence-driven architecture creates a specific kind of assumption pressure. When cadence becomes a declared objective, unresolved assumptions become obstacles. Obstacles in turn become candidates for reinterpretation. Reinterpretation then flows into normalization. And normalization turns into drift. That is how schedule pressure enters the survival case without announcing itself as a safety problem. A launch cadence is not a safety argument. A landing cadence is not a survival argument.

The danger is not cadence itself. Regular operations can produce learning, discipline, standardization, and experience. **The danger is unearned cadence.** The danger is treating recurrence as proof of readiness before the architecture has demonstrated that recurrence can be

sustained without eroding margin. If *Artemis* is forced to move faster than analysis, verification evidence, workforce infrastructure, commercial dependencies, and safety governance can support, cadence will not mature the campaign. It will compress it.

That compression is integral because *Artemis* is not just repeating the same flight. Each mission introduces changing configurations involving commercial roles, surface objectives, logistics dependencies, and potentially altered Gateway assumptions. A campaign with changing architecture cannot claim maturity through repetition alone. It must be governed, explicitly owned, rigorously tested, and shown survivable.

The question is not whether *Artemis* can declare a cadence; it is whether the architecture can survive the cadence it declares.

11.6 Distributed Accountability Is Not Integrated Accountability

Artemis distributes work across NASA centers, contractors, commercial partners, international partners, acquisition models, and budget lines. That distribution may be necessary. It may even be the only plausible way to build a modern lunar campaign. But necessity does not make it safe. A distributed architecture requires a stronger accountability structure, not a weaker one. Distributed accountability is not integrated accountability.

This is one of the most important distinctions in the entire paper. *Artemis* can assign responsibility for SLS, Orion, HLS, Gateway, xEVAS, Ground Systems, Communications, Logistics, Science Services, Mobility, and Commercial Services. But assigning responsibility for elements is not the same as assigning responsibility for the integrated consequence. The crew does not care which organization owned the failed assumption. The crew experiences the failure as one event.

The question is not whether SLS is responsible for SLS, Orion is responsible for Orion, HLS is responsible for HLS, or Gateway is responsible for Gateway. The question is: **Who is responsible for the crew when the boundary between those systems becomes the failure surface?** The boundary between systems is where accountability most often goes to hide.

That is the inherent danger in a distributed architecture. Each organization can be competent. Each element can be professionally managed. Each contractor can satisfy its contract. Each interface can appear traceable. Each review can close. Yet the integrated survival case can still contain holes because the consequence lives *between the parts*, not inside any one part.

This is an important point for *Artemis* because the architecture is changing even as it matures. Gateway decisions, commercial lander development, suit readiness, lunar surface infrastructure, cargo logistics, advanced communications, and post–*Artemis* V commercial transitions are not static background conditions. They are active variables. Each variable carries assumptions. Every assumption requires ownership. Every ownership boundary creates the possibility of drift.

So, it stands to reason that a distributed campaign must have a **visible architecture owner** with authority over the integrated survival case. This is a coordination function, not a review board, and not just an interface-control process. An owner with the authority to say:

- This assumption is not closed.
- This dependency is not mature.
- This cadence is not yet justified.
- This interface is not survivable.
- This budget change reopens the hazard analysis.
- **This mission should not proceed until the crew survival consequence is understood.**

If every element owns its own success and no one owns the integrated consequence, the architecture has a moral blind spot.

The moral blind spot is not created by bad people. It is created by structure. It is created when local success is easier to defend than system-level risk. It is created when contracts reward deliverables more clearly than they reward integrated survivability. It is created when schedule pressure treats unresolved assumptions as irritants rather than warnings. **It is created when no one wants to stop the train because everyone can point to a different part of the track and say, "This section is fine."**

Artemis cannot afford that. The campaign is too complex, too public, too expensive, too politically important, and too morally loaded. A distributed lunar architecture must be **more** accountable than *Apollo* or *Shuttle*, not less, because its risk is more dispersed. The more distributed the work becomes, the more explicit and empowered the survival authority must become.

A distributed campaign without integrated accountability is not modern. It is materially exposed and inherently risky.

11.7 Workforce, Institutional Memory, and the Hidden Safety Infrastructure

Workforce is not overhead. Workforce is survival infrastructure[283]. Experienced engineers, safety personnel, analysts, operators, program integrators, mission assurance teams, and technical authorities carry more than labor capacity. They carry memory. They know which assumptions were inherited, which trades were painful, which waivers were tolerated, which risks were normalized, and which interfaces were never as clean as the chart suggested. When that workforce is reduced, the loss is not limited to headcount. The program loses context, and when context disappears, assumptions become easier to repeat because fewer people remain who remember why they were dangerous to begin with.

[283] Aerospace Safety Advisory Panel Annual Report; ASAP 2025 Annual Report (workforce, technical-authority, acquisition, and *Artemis* integrated-risk sections).

Artemis is so much more than a hardware challenge. It is an institutional memory challenge. A program of this duration inherits decisions across administrators, contractors, centers, and budget cycles. It carries assumptions from Constellation, SLS, Orion, Gateway, commercial cargo, commercial crew, ISS operations, lunar architecture studies, suit development, and human-rating practice. Some of those assumptions are visible, while others remain buried in decisions that were once carefully debated but are now treated as background chatter.

Institutional memory is one of the few defenses against assumption drift.

When institutional support accounts are reduced, the consequence is not always immediate. The rocket may still roll out. The spacecraft may still be processed. The review may still occur. The mission may still launch. This is what makes the loss dangerous. Institutional thinning often appears survivable until the system needs the judgment that was removed.

It is the missing person, the missing analyst, the missing safety voice, the missing test capability, or the missing historical memory that becomes visible only when the program reaches for it and finds an empty space.

NASA's FY 2027 request reduces Safety, Security, and Mission Services from $3.000 billion in FY 2026 to $1.999 billion in FY 2027, while Construction and Environmental Compliance and Restoration falls from $185.3 million to $100.6 million. The same request increases Exploration. Do not mistake this for a spreadsheet contrast. It is a governance signal. The campaign is being asked to do more while the institutional systems that support engineering discipline, technical authority, infrastructure, safety, and operations are being reduced.

A program cannot cut the nervous system and then celebrate the strength of the muscles.

The *Artemis* workforce does more than execute tasks. *They* interpret anomalies. *They* challenge optimistic assumptions. *They* recognize patterns. *They* know when a requirement closure is too clean. *They* remember when a design trade was accepted because the schedule was tight, not because the risk was gone. *They* recognize when a contractor's confidence must be validated by independent evidence. *They* see when a program is beginning to talk itself into believing that a gap is manageable because naming the gap would be inconvenient. *They* are the reason why *Artemis* exists.

That kind of judgment and experience cannot be replaced quickly. It is built through years of exposure, failure, mentorship, documentation, testing, operations, and hard-earned skepticism. They are not interchangeable. *They* are not an administrative luxury. *They* are a part of the survival system. *They* are the people who remember the assumptions baked into the architecture.

If *Artemis* loses too much institutional memory, the program may still look organized. It will have all its charts up, all its plans and reviews and signatures—everything necessary to move forward. But it will become easier for old assumptions to return in new language. It will become easier for unresolved questions to be treated as settled because the people who remember the original uncertainty are gone. It will become easier for governance to become procedural rather than protective. That is how a program becomes fragile while still appearing extraordinary.

11.8 Historical Lessons: The Pattern is Not the Vehicle, it is the Drift

The historical cases in this paper do not warn against exploration; they warn against interpretation failure. *Apollo* 1, *Challenger*, and *Columbia* each show what happens when warning signs are present but not given sufficient architectural authority *to act*[284]. In each case, the failure was not just technical—it was interpretive. The system had information, but it did not convert that information into protective action with sufficient force, urgency, or authority.

That distinction is essential. A technical anomaly becomes catastrophic when the organization misreads it, contains it too narrowly, normalizes it, or assigns it to a category that prevents action. In *Apollo* 1, the danger was more than a flammable cabin environment; it was the failure to treat the integrated ground-test configuration as a lethal system. In *Challenger*, the danger was more than O-ring behavior; it was the normalization of erosion and the inability of an engineering concern to defeat launch pressure. In *Columbia*, the danger was more than foam impact; it was the organizational interpretation that converted a strike into an accepted maintenance issue rather than an urgent survival threat.

The lesson is not that *Artemis* must fear the same failures. The lesson is that *Artemis* must fear the same mechanisms.

Artemis has different hardware, but the mechanisms can return - schedule pressure, budget pressure, fragmented ownership, optimism around recovery, reinterpretation of anomalies, late-stage acceptance of known gaps, and the quiet transformation of "not yet proven unsafe" into "safe enough for now**." The campaign does not have to repeat history literally in order to inherit its dangers.**

The program's cost history reinforces the concern. NASA OIG projected *Artemis* campaign costs reaching $93 billion between FY 2012 and FY 2025 and estimated that the SLS/Orion system plus related ground infrastructure would cost at least $4.1 billion per launch for the first four *Artemis* missions, excluding $42 billion in earlier formulation and development costs[285]. OIG warned that these costs challenge long-term sustainability, especially as Congress urges increased SLS/Orion

[284] *Columbia* Accident Investigation Board Report, Volume I (2003); Report of the Presidential Commission on the Space *Shuttle Challenger* Accident, Volume I (1986); Report of the *Apollo* 204 Review Board (1967); Diane Vaughan, *The Challenger Launch Decision* (1996); Nancy Leveson, *Engineering a Safer World* (2011).
[285] NASA Office of Inspector General, *Artemis* / SLS-Orion cost audit report (sections with the $93 B FY2012–FY2025 estimate and SLS/Orion/ground-system cost findings).

cadence under budget constraints. NASA OIG further estimated that a single SLS Block 1B produced under the Exploration Production and Operations contract would cost at least $2.5 billion, excluding systems engineering and integration—and called NASA's 50% savings goal highly unrealistic[286]-.

Cost growth is more than a financial problem; it is a safety pressure. Cost growth produces schedule pressure. Schedule pressure produces reinterpretation pressure. Reinterpretation pressure produces assumption drift. Assumption drift produces operational risk. **A dollar overrun does not kill a crew. The pressure created by unmanaged overruns can.**

GAO has reported that three *Artemis* projects account for nearly $7 billion in cost overruns—almost half of the overruns collectively experienced by 53 NASA major projects assessed since GAO's final annual review in 2009. That is not a reason to abandon *Artemis*; it is a reason to govern it more honestly. Cost growth at this scale is a warning that architecture, acquisition, requirements, and execution are unrestrained. Strain does not automatically produce disaster. But unmanaged strain is where assumptions begin to mutate.

It should be understood that the danger is not that anyone intends to weaken safety. The danger is that safety can be weakened indirectly by the need to preserve momentum. A program under pressure begins to search for acceptable interpretations—and asks whether delay is really necessary, whether a test can be descoped, whether a waiver is defensible, whether a dependency can be carried forward, whether an interface can be finalized later, whether risk can be closed with partial evidence, whether a concern can be treated as local rather than architectural.

This is how drift happens: drift does not arrive wearing a warning label. It arrives wearing the language of reasonableness.

11.9 The Standard *Artemis* Must Be Forced to Meet

Artemis should be judged by the standard human spaceflight demands and not whether the nation wants it, not whether Congress funds it, not whether contractors can sell it, not whether NASA can brief it, and not whether a single mission succeeds. It must be judged by whether the campaign can name its assumptions, assign ownership to them, test them, reopen them, fund them, stop them, and protect the crew when those assumptions fail**. Anything less is NOT maturity. It is merely being performative.**

This is the standard that should govern *Artemis* from this point forward. Every major architecture decision should be forced through the survival case. Every budget change should reopen the assumptions it affects. Every cadence declaration should identify the verification, workforce,

[286] U.S. Government Accountability Office, GAO-25-107591, *NASA Major Projects* (highlights/summary and *Artemis* project profiles).

infrastructure, and rescue logic required to support it. Every commercial transition should identify who owns the integrated consequence if the provider meets its contract, but the campaign remains exposed. Every Gateway change should identify what happens to staging, contingency, loiter, rescue, communications, logistics, and international commitments. Every suit delay, HLS delay, SLS change, Orion finding, ground-system issue, or surface-architecture shift should be treated not as an isolated program event, but as a possible change to the crew survival case. **The survival case must be a living architecture product, not a ceremonial safety case.**

This is where assumption management becomes central. The program must know what it is assuming. It must know who owns each assumption. It must know what evidence supports each assumption. It must know when the assumption expires. It must know what decisions reopen it. It must know what happens to the crew if the assumption fails. If that chain breaks, the assumption is not controlled; it is simply being carried forward.

Artemis should therefore be required to maintain an integrated assumption-control framework tied directly to crew survival. That framework should include budget assumptions, schedule assumptions, interface assumptions, commercial-provider assumptions, rescue assumptions, logistics assumptions, communications assumptions, surface-duration assumptions, suit-readiness assumptions, HLS-maturity assumptions, Gateway or non-Gateway staging assumptions, and workforce-capacity assumptions. Each should have an owner, an evidence threshold, a review trigger, and a survival-consequence definition.

An assumption without an owner is a risk waiting for bureaucracy to excuse it.

This paper does not argue that *Artemis* must stop. It argues that *Artemis* must grow. A campaign that carries astronauts beyond LEO cannot be governed by enthusiasm, inherited confidence, partial readiness, or the prestige of the goal. It must be governed by disciplined truths. The architecture must be forced to confess its weak points before the environment does. The program must be able to say not only what is planned, but what is survivable. It must be able to explain not only how the mission works, but how the crew lives when the mission does not.

If *Artemis* is to be worthy of the crew it carries, it must become more than ambitious. It must become accountable. **The Moon does not care whether the architecture was politically convenient. Vacuum does not negotiate with budget strategy.**

That is the final test. *Artemis* cannot be allowed to call itself mature because its pieces are funded, its milestones are met, its cadence is desired, or its political value is obvious. It becomes mature only when its assumptions are visible, its accountability is integrated, its survival case is governed, and its crew can be protected across the full architecture under degraded conditions.

Anything less is not maturity; it is momentum wearing a badge.

Momentum is not a survival strategy.

12 Conclusion

This paper began with a simple concern: assumptions in complex human spaceflight systems do not remain harmless because they are invisible. They move. They travel through requirements, schedules, contracts, hazard analyses, verification plans, design reviews, operational concepts, and budget decisions. They are inherited by people who did not create them. They are defended by organizations that may not fully remember why they were accepted. They are normalized by repetition. They are protected by momentum. Therefore, if they are not deliberately managed, they become part of the architecture itself.

That is the central argument of this paper: assumptions are not background conditions in human spaceflight. They are active system elements. They shape what is designed, what is tested, what is waived, what is funded, what is deferred, what is believed, and what is allowed to proceed. In a program as distributed and politically significant as *Artemis*, assumption management is not an administrative exercise. It is a safety function. It is a governance function. It is a survival function.

The historical record makes this impossible to ignore. *Apollo* 1, *Challenger*, and *Columbia* were not simply technical failures—they were failures of interpretation, authority, integration, and institutional response. In each case, the danger existed before disaster. Signals existed before consequence. Warnings existed before loss. The failure was not that the systems lacked information**. The failure was that information lacked the authority, urgency, or organizational force needed to stop the architecture from moving toward catastrophe.**

That pattern highlights that *Artemis* is not protected from history by being new. It is not protected by modern language, commercial acquisition models, international partnerships, digital tools, or inspirational mission statements. These things matter, but they do not repeal the physics of risk or the sociology of drift. Human spaceflight remains unforgiving. Vacuum remains unforgiving. Fire remains unforgiving. Reentry remains unforgiving. Lunar distance remains unforgiving. **The Moon will not care whether a risk was hidden in an interface document, a budget assumption, a contract boundary, a maturity chart, or a schedule commitment.**

The environment does not honor our organizational charts. It only exposes them.

Artemis is ambitious—and it should be. Returning humans to the Moon, building a sustained lunar presence, preparing for Mars, expanding scientific reach, and strengthening national capability are worthy objectives. This paper does not argue against *Artemis* but for it. It argues against a shallow definition of *Artemis* maturity. It argues against the temptation to confuse motion with readiness, funding with integration, cadence with resilience, and political commitment with crew survival.

A campaign does not become mature because it is important. It becomes mature when it can prove that its assumptions are visible, owned, tested, reopened, and controlled. It becomes mature when success on one mission is not used to silence hard questions about the next. It becomes mature when budget changes trigger architectural reassessment instead of quiet optimism. It becomes mature when commercial interfaces are treated as survival boundaries, not procurement conveniences. It becomes mature when workforce, institutional memory, safety authority, and engineering skepticism are understood as part of the architecture—not overhead surrounding it.

Most importantly, *Artemis* becomes mature only when crew survival is treated as the measure that matters most. Crew survival is not a sentimental concern. It is not an emotional add-on to technical readiness. It is the most honest test of whether the campaign has matured as an integrated system. It asks, *"What happens when the mission stops being clean?"*. It asks, "*What happens when a vehicle is delayed, a lander is late, a suit is not ready, a communication path degrades, a surface timeline stretches, a rescue option narrows, a budget decision changes the architecture, or an organization discovers that its local success did not guarantee system-level safety?"*. Ultimately, crew survival is the truth serum of *Artemis* maturity.

The truth may be uncomfortable, but discomfort is not a reason to avoid it. If the survival case is strong, it should withstand scrutiny. If it is incomplete, the crew deserves that truth before they are asked to carry the consequence. A program worthy of human spaceflight does not fear the hard question. It builds itself around answering it.

That is why assumption management must move from the margins of program discipline to the center of safety governance. *Artemis* must know what it is assuming. It must know who owns each assumption. It must know what evidence supports each assumption. It must know when the assumption expires. It must know what decisions reopen it. It must know what happens to the crew if the assumption fails. Without that chain, the assumption is not managed, it is merely tolerated. And tolerated assumptions are dangerous because they often look reasonable until the moment they are not.

The budget and cadence pressures discussed in this paper sharpen the concern. *Artemis* is being asked to accelerate, standardize, transition, commercialize, sustain, and repeat. It is being asked to become a campaign while still proving the maturity of critical dependencies. That creates a governance burden that cannot be met by confidence alone. It requires integrated authority. It requires disciplined truths. It requires a survival case that remains intact even when strategy, funding, schedule, and architecture do not move in perfect alignment.

A campaign cannot be governed by momentum and then judged surprised when momentum becomes pressure.

The question, then, is not whether *Artemis* should continue. It should. The question is what kind of *Artemis* will continue. Will it be an architecture that names its assumptions before they harden into doctrine? Will it be a campaign that protects dissenting technical judgment before schedule pressure turns it into inconvenience? Will it be a program that treats contractors and civil servants as partners in truth rather than instruments of momentum? Will it be an agency effort that understands science, safety, engineering, operations, infrastructure, and workforce as one ecosystem? Will it be a national campaign mature enough to say "not yet" when "not yet" is the answer that protects the crew?

That is the standard. Not perfection. Not fear. Not paralysis. Human spaceflight has never been risk-free, and it never will be. The standard is not the elimination of risk, but the honesty about risk. The standard is the ownership of risk. The standard is the refusal to let assumptions drift beyond accountability. The standard is the discipline to understand the difference between accepting risk and inheriting it blindly. ***Artemis* does not need softer language. It needs clearer obligations.**

It owes truth to the crew. It owes truth to the workforce. It owes truth to the engineers, analysts, operators, safety professionals, technicians, contractors, civil servants, scientists, and leaders who have given pieces of themselves and their lives to this mission. It owes truth to the people who know where the hard problems are and continue to show up anyway. It owes truth to those who fight for science, for the agency, and for the future this campaign claims to serve.

That truth is not disloyalty. It is the highest form of loyalty.

To tell the truth about *Artemis* is not to weaken it—it is to protect it. To question its assumptions is not to undermine the mission. It is to make the mission worthy of the people who will fly it. To demand a crew-survival standard is not to insult the program, but to honor the lives the program will place beyond Earth.

The conclusion of this paper is therefore direct: *Artemis* must not be allowed to mature only on paper. It must mature in architecture, authority, verification, funding, realism, workforce protection, commercial accountability, contingency planning, rescue logic, surface operations, institutional memory, and in the disciplined management of assumptions. Anything less leaves the campaign vulnerable to the oldest danger in human spaceflight: believing that because the mission can proceed, the system is ready. Proceeding is not the same as being ready.

Artemis can be extraordinary. It can be worthy of *Apollo* without being trapped by *Apollo*. It can learn from *Shuttle* without repeating *Shuttle*. It can use commercial capability without surrendering integrated accountability. It can honor political urgency without becoming governed by it. It can build a lunar campaign that expands human presence while preserving the moral seriousness of human spaceflight. The decision to choose a maturity model must be deliberate.

The Moon does not care whether *Artemis* was difficult to fund. Vacuum does not care whether an interface was commercially negotiated. Distance does not care whether a requirement was closed. Hardware does not care whether a schedule was politically important. A failed assumption will not pause because the program has good intentions.

That is why assumption management matters. That is why crew survival matters. That is why Section 11 could not be passive, and why this conclusion cannot be either. The stakes are too high for language that politely steps around the truth. *Artemis* is not only a mission to return humans to the Moon; it is a test of whether NASA and the nation can still tell the truth before consequence does.

Momentum is not a survival strategy.

Per Aspera Ad Astra; Veritas In Tacito Scribendo.

Glossary of Terms

Assumptions and Assumption Governance

Architecture Assumption
An assumption tied to the way systems, vehicles, partners, missions, schedules, or interfaces are expected to work together across a campaign.

Assumption
A condition, belief, dependency, or expectation accepted as true for planning, design, verification, operations, or risk acceptance before complete evidence is available.

Assumption Control
The governance discipline that prevents assumptions from becoming unmanaged drivers of safety, design, schedule, or mission risk.

Assumption Decay
Weakening of an assumption's validity as technical, organizational, budgetary, or operational conditions change.

Assumption Drift
The gradual movement of an assumption away from its original context, evidence, or validity without formal recognition.

Assumption Life Cycle
The progression of an assumption from creation through use, verification, reinterpretation, drift, reopening, and closure.

Assumption Management
The deliberate process of identifying, documenting, owning, testing, updating, retiring, or escalating assumptions as evidence, architecture, schedule, risk, or mission conditions change.

Assumption Ownership
Clear accountability for monitoring, validating, updating, and escalating a specific assumption.

Assumption Propagation
The movement of an assumption into other requirements, interfaces, analyses, schedules, or operational decisions, often without explicit restatement.

Assumption Register
A controlled repository listing assumptions, owners, evidence basis, affected systems, conditions, triggers for reopening, and closure status.

Assumption Reopening
The formal act of reassessing an assumption because new evidence, anomalies, delays, interface shifts, or risk conditions have changed its basis.

Embedded Assumption
An assumption that becomes built into a requirement, interface, schedule, control, verification plan, or procedure—often losing visibility as an assumption.

Invisible Assumption
An assumption that continues to shape decisions but is no longer explicitly tracked, challenged, or owned.

Operational Assumption
An assumption about how a system, crew, ground team, procedure, timeline, or contingency response will perform during mission execution.

Programmatic Assumption
An assumption related to funding, schedule, contract performance, partner delivery, organizational authority, or management structure.

Safety-Critical Assumption
An assumption whose failure could affect crew survival, loss of mission, loss of crew, or abort capability.

Verification Assumption
An assumption about what a test, analysis, inspection, demonstration, or certification result actually proves.

Governance and Authority

Architecture Authority
Authority to define, control, integrate, and change the overall mission or campaign architecture.

Budget Authority
Authority to allocate, constrain, or redirect funding that affects program direction.

Distributed Governance
A condition in which authority and decision-making are spread across organizations, contractors, partners, boards, or programs.

Distributed Accountability
A condition where many organizations influence risk or outcomes, but no single owner controls the full consequence chain.

Governance
The structure defining authority, accountability, escalation paths, and control over program direction, safety, configuration, and risk.

Governance Hole
A gap where authority, funding, responsibility, and integration do not align, leaving no entity fully responsible for architectural consequences.

Misaligned Authority
A condition in which the organization controlling money, strategy, safety, or architecture does not control the consequences of those decisions.

Safety Governance
The structures and processes ensuring safety risk is identified, owned, challenged, and controlled—not subordinated to cost, schedule, or political pressure

---

Architecture and Systems Integration

Campaign Architecture
A multi-mission, multi-system, multi-partner structure requiring coordinated maturity across interdependent elements.

Certification
Formal acceptance that a system or element meets defined criteria for use or flight.

Certification of Flight Readiness (CFR)
Formal determination that a system or mission is ready for flight based on evidence and risk posture.

Change Control
Evaluating, approving, rejecting, or constraining changes to requirements, design, schedule, operations, configuration, or risk posture.

Configuration Management
Formal control of approved design documentation, requirements, interfaces, software, hardware, and changes over time.

Integrated Architecture
An architecture assessed as a whole for compatibility, timing, risk, safety, readiness, and crew survival performance.

Interdependency
A condition in which one system, schedule, or organization depends on another in a way that transfers risk.

Interface
A physical, functional, data, operational, organizational, or contractual connection between systems or organizations.

Interface Control
Defining, documenting, managing, and verifying interfaces so changes do not create unmanaged risk.

Requirement Drift
Gradual change in the meaning, verification, or enforcement of a requirement without formal recognition.

Requirements Traceability
The ability to link requirements to their sources, rationale, verification, implementation, controls, and operational consequences.

Qualification
Evidence-based proof that hardware, software, materials, processes, or systems can perform safely and reliably under expected conditions.

Reverification
Additional verification after a change, anomaly, repair, substitution, reinterpretation, or new risk condition.

Verification
Confirming that a system or component was built correctly according to specified criteria.

Validation
Confirming that the built system satisfies mission needs and operational intent.

V&V (Verification and Validation)
The combined discipline ensuring the system both meets requirements and that the requirements are sufficient.

Hazards, Risks, and Risk Behavior

Dissenting Opinion
A formally or informally expressed technical or safety disagreement.

Hazard
A condition that can lead to injury, loss of crew, loss of mission, damage, or unacceptable consequence.

Hazard Analysis
Structured identification of hazards, causes, controls, verifications, residual risks, and consequences.

Hazard Control
A design, procedure, verification, or mitigation intended to prevent or reduce a hazard.

Lagging Indicator
Evidence revealed after a failure or anomaly.

Latent Condition
A hidden weakness that can combine with other factors to cause failure.

Leading Indicator
A sign that risk may be increasing before failure occurs.

Normalization of Deviance
Treating abnormal conditions as normal after repeated acceptance.

Organizational Silence
Failure to voice or escalate concerns or technical objections.

Residual Risk
A risk that remains after mitigations.

Risk Acceptance
Formal decision to proceed despite residual risk.

Risk Migration
Gradual relocation of risk without explicit recognition.

Risk Posture
Overall state of risk in a system or architecture.

Risk Transfer
Movement of risk across systems, organizations, or mission phases.

Risk Ownership
Defined responsibility for a specific risk.

Technical Authority and Programmatic Pressure

Budget Pressure
Programmatic pressure driven by funding constraints.

Independent Technical Authority
Technical authority insulated from cost and schedule pressure.

Programmatic Pressure
Pressure from schedule, cost, politics, or commitments that influences technical judgment.

Schedule Pressure
Programmatic pressure driven by calendar commitments.

Technical Authority
Responsible for independent technical judgment, standards, requirements, and risk evaluation.

Margins, Survival, and Operational Safety

Abort Capability
Ability to terminate or redirect a mission phase safely.

Abort Path
Defined sequence enabling safe return from a hazardous state.

Crew Survival Case
Integrated argument that a mission protects the crew across all phases.

Contingency Operations
Planned responses to failures or emergencies.

Loss of Crew (LOC)
Catastrophic outcome involving crew fatality.

Loss of Mission (LOM)
Inability to complete primary mission objectives

Margin
Difference between required and actual capability.

Operational Margin
Procedural, time, consumable, or performance flexibility under off-nominal conditions.

Rescue Assumption
Assuming another asset or system can recover the crew.

Safety Margin
Margin protecting crew or critical systems.

Survivability
System and crew ability to withstand and recover from hazards.

Mission Assurance Disciplines

Common Cause Failure
A single cause defeating multiple protections.

Fault Tolerance
Ability to continue operating safely after failures.

Mission Assurance
Ensuring mission objectives can be achieved safely and reliably.

Quality Assurance
Ensuring products and processes meet standards

Redundancy
Extra systems or functions to preserve capability.

Reliability
Probability that a system performs as intended.

Resilience
Ability to absorb disturbance and recover.

Single-Point Failure
One failure causing Loss of Mission/Loss of Crew.

High-Reliability and Organizational Behavior

Deference to Expertise
Letting those with the most knowledge influence decisions.

Escalation
Raising a concern or risk to higher authority.

High Reliability Organization (HRO)
Organization operating in high-risk domains with disciplined decision-making.

Mindful Organizing
Behavior emphasizing weak-signal detection, expertise, and operational sensitivity.

Weak Signal
Early or ambiguous evidence of risk.

Bibliography

1. 51 U.S.C. § 20302, Vision for space exploration, including statutory notes for the Moon to Mars Office and Program added by Pub. L. No. 117–167, div. B, title VII, § 10811 (2022). https://uscode.house.gov/view.xhtml?edition=prelim&req=granuleid%3AUSC-prelim-title51-section20302
2. 51 U.S.C. § 20306. Special appropriations for Mars missions, *Artemis* missions, and Moon to Mars program. https://uscode.house.gov/view.xhtml?edition=prelim&num=0&req=granuleid%3AUSC-prelim-title51-section20306
3. Aerospace Safety Advisory Panel. (2026). 2025 annual report. National Aeronautics and Space Administration. https://oiir.hq.nasa.gov/asap/reports.html
4. Ali, R., Dalpiaz, F., Giorgini, P., & Silva Souza, V. E. (2011). Requirements evolution: From assumptions to reality. In Proceedings of the 16th International Conference on Exploring Modeling Methods for Systems Analysis and Design (EMMSAD 2011), 372–382. https://link.springer.com/chapter/10.1007/978-3-642-21759-3_27
5. Anderson, M., Sargusingh, M., & Perry, J. (2017). Evolution of requirements and assumptions for future exploration missions (ICES-2017-271). https://ttu-ir.tdl.org/items/6b4342a4-948e-40fb-98df-88571d2d2e91
6. *Apollo* 204 Review Board. (1967). Report of the *Apollo* 204 Review Board. National Aeronautics and Space Administration. https://history.nasa.gov/*Apollo*204/
7. Broniatowski, D. A. (2012). A framework for decision making in complex engineering systems under uncertainty. Systems Engineering, 15(1), 48–65. https://broniatowski.weebly.com/publications.html
8. Broniatowski, D. A., & Weigel, A. L. (2008). The political sustainability of space exploration. Space Policy, 24(3), 148–157. https://doi.org/10.1016/j.spacepol.2008.06.004
9. Broniatowski, D. A., & Weigel, A. L. (2012). The role of communication, coordination, and knowledge transfer in high-reliability aerospace organizations. Journal of Aerospace Operations, 1(4), 325–339. http://content.iospress.com/articles/journal-of-aerospace-operations/aop050
10. Brookfield, S. D. (2017). Becoming a critically reflective teacher (2nd ed.). Jossey-Bass. https://www.wiley.com/en-us/Becoming+a+Critically+Reflective+Teacher%2C+2nd+Edition-p-9781119049708
11. Cameron, B. G., Seher, T., & Crawley, E. F. (2011). Goals for space exploration based on stakeholder value network considerations. Acta Astronautica, 68(11–12), 2088–2097. https://doi.org/10.1016/j.actaastro.2010.11.003
12. Crues, E. Z., Whittington, P., & Dodd, K. (2025). Development of the *Artemis* distributed simulation FOMs. NASA Johnson Space Center. https://doi.org/10.1109/AERO63441.2025.11068606
13. Dekker, S. W. A. (2011). Drift into failure: From hunting broken components to understanding complex systems. Ashgate. https://www.routledge.com/Drift-into-Failure-From-Hunting-Broken-Components-to-Understanding-Complex-Systems/Dekker/p/book/9781409422228
14. Denney, E., Pai, G., & Habli, I. (2015). Safety case patterns: Theory and applications. NASA Ames Research Center / FAA REDAC presentation. https://www.faa.gov/about/office_org/headquarters_offices/ang/redac/redac-sas-201503-nasa-safety-case-patterns.pdf
15. Dewar, J. A. (2002). Assumption-based planning: A tool for reducing avoidable surprises. Cambridge University Press. https://www.rand.org/pubs/commercial_books/CB101.html
16. Dewar, J. A., Builder, C. H., Hix, W. M., & Levin, M. H. (1993). Assumption-based planning: A planning tool for very uncertain times. RAND Corporation. https://www.rand.org/pubs/monograph_reports/MR114.html
17. Donahue, A. K., & O'Leary, R. (2003). Disaster and the dynamics of collaboration: Lessons from the World Trade Center response. Public Administration Review, 63(3), 9–20. https://onlinelibrary.wiley.com/journal/15406210
18. Donahue, A. K., & O'Leary, R. (2012). Do shocks change organizations? The case of NASA. Journal of Public Administration Research and Theory, 22(3), 395–425. https://doi.org/10.1093/jopart/mur034
19. Dreier, C. (2026, April 23). The FY 2027 NASA budget request. The Planetary Society. https://www.planetary.org/articles/analyzing-the-fy-2027-nasa-budget-request
20. Dreier, C. (2026, March 25). 3 charts that show how *Artemis* compares to *Apollo*. The Planetary Society. https://www.planetary.org/articles/3-charts-that-show-how-*Artemis*-compares-to-*Apollo*
21. Ehrenfreund, P., Peter, N., & Billings, L. (2010). Building long-term constituencies for space exploration: The challenge of raising public awareness and engagement in the United States

and in Europe. Acta Astronautica, 67(3-4), 502-512. https://doi.org/10.1016/j.actaastro.2010.03.002
22. Eisenhower Presidential Library. (1958). Minutes of Cabinet meetings, August 15 and October 18, 1958. https://www.eisenhowerlibrary.gov/research/online-documents/early-history-and-development-national-aeronautics-and-space
23. El Fassi, S. (2021). Assumption management in model-based systems engineering: An aircraft design perspective [Doctoral dissertation, Cranfield University]. https://www.researchgate.net/publication/364328347_Assumption_Management_in_Model-Based_Systems_Engineering_An_Aircraft_Design_Perspective
24. El Fassi, S., Chen, X., Riaz, A., Guenov, M. D., van Heerden, A. S. J., & Jimeno Altelarrea, S. (2024). Managing assumption-driven design change via margin allocation and trade-offs. Journal of Engineering Design, 35(10), 1258-1291. https://doi.org/10.1080/09544828.2023.2259741
25. El Fassi, S., Guenov, M. D., & Riaz, A. (2020). An assumption network-based approach to support margin allocation and management. Proceedings of the Design Society: DESIGN Conference, 1, 2275–2284. https://doi.org/10.1017/dsd.2020.25. https://www.cambridge.org/core/journals/proceedings-of-the-design-society-design-conference/article/an-assumption-networkbased-approach-to-support-margin-allocation-and-management/224C5249F72E7C146606D2DAA85E8AA4
26. Executive Office of the President. (2019). Launch of spacecraft containing space nuclear systems. https://trumpwhitehouse.archives.gov/presidential-actions/presidential-memorandum-launch-spacecraft-containing-space-nuclear-systems/
27. Gibson, M. A., Poston, D. I., McClure, P., Godfroy, T. J., Briggs, M. H., & Sanzi, J. L. (2018). Kilopower Reactor Using Stirling TechnologY (KRUSTY) nuclear ground test results and lessons learned (NASA/TM-2018-219941). NASA. https://ntrs.nasa.gov/citations/20180007389
28. Grace Eliza Goodwin. (2026). There was a bit of toilet trouble on NASA's *Artemis* 2 mission to the moon. https://www.space.com/space-exploration/*Artemis*/theres-a-bit-of-toilet-trouble-on-NASA's-*Artemis*-2-mission-to-the-moon
29. Graydon, P. J. (2015–2016). Understanding assurance cases: An educational series in five parts. NASA Langley Research Center. https://shemesh.larc.nasa.gov/arg/uac-all5.pdf
30. The Guardian. (2026). Chance of alien life 'goes to heart' of space missions, Nasa chief says. https://www.theguardian.com/us-news/2026/apr/05/alien-life-nasa-jared-isaacman-*Artemis*-ii
31. Hawkins, R. D. (2025). Developing compelling safety cases. arXiv. https://arxiv.org/abs/2502.00911
32. Hawkins, R., Kelly, T., Knight, J., & Graydon, P. (2011). A new approach to creating clear safety arguments. In Advances in Systems Safety (pp. 3–23). Springer. https://link.springer.com/chapter/10.1007/978-0-85729-133-2_1
33. Helton, R. (2022). NASA nuclear flight safety: New protocols. NASA Technical Reports Server. https://ntrs.nasa.gov/api/citations/20220004403/downloads/NASA_NFS_New_Protocols.pdf
34. House Committee on Science, Space, and Technology. (2026, April 22). Administrator Isaacman testimony prepared for NASA FY2027 budget hearing. https://republicans-science.house.gov/_cache/files/b/d/bd4222ca-1bcb-4bf7-81fc-099fbcb043b4/5BE37E3EA231D55FD784FD60FD75613C487846BCE9905CFA2E7C07FD8D5067DD.04.22.26-nasa-budget-hearing---administrator-isaacman-testimony.pdf
35. IAQG 9100 Team. (2019). 9100:2016 overview / major changes presentation. https://iaqg.org
36. INCOSE. (2023). INCOSE systems engineering handbook: A guide for system life cycle processes and activities (5th ed.). Wiley. https://www.incose.org/resource/incose-systems-engineering-handbook-a-guide-for-system-life-cycle-processes-and-activities-5th-edition/
37. International Aerospace Quality Group. (2017). 9100:2016 series of standards: Frequently asked questions. https://iaqg.org/tools/9100-2016-faq/
38. International Council on Systems Engineering. (2020). Systems engineering and system definitions (INCOSE-TP-2020-002-06). INCOSE. https://www.incose.org/about-systems-engineering/system-and-se-definition
39. Jessica Taveau. "Welcome Home! NASA's SpaceX Crew-9 Back on Earth After Science Mission." NASA News Release. March 18, 2025. https://www.nasa.gov/news-release/welcome-home-NASA's-spacex-crew-9-back-on-earth-after-science-mission/
40. Johnson Space Center Safety and Mission Assurance Directorate. (n.d.). NC division / NC2 / NC3 organizational description. https://www.nasa.gov/johnson/
41. Kaldon, M. (2024). Fission surface power project overview. NASA Technical Reports Server. https://ntrs.nasa.gov/citations/20240003580
42. Kinser, J. (2016). Don't make an ass out of you and me—using assumptions effectively. Project Management Institute. https://www.pmi.org/learning/library/assumptions-based-planning-analyze-techniques-6582

43. Lempert, R. J. (2019). Robust decision making (RDM). In V. A. W. J. Marchau, W. E. Walker, P. J. T. M. Bloemen, & S. W. Popper (Eds.), Decision making under deep uncertainty: From theory to practice (pp. 23–51). Springer. https://doi.org/10.1007/978-3-030-05252-2_2. https://www.rand.org/pubs/external_publications/EP67834.html
44. Leveson, N. G. (2011). Engineering a safer world: Systems thinking applied to safety. MIT Press. https://library.oapen.org/bitstream/20.500.12657/26043/1/1004042.pdf
45. Lewis, G. A., Mahatham, T., & Wrage, L. (2004). Assumptions management in software development (CMU/SEI-2004-TN-021). Software Engineering Institute. https://doi.org/10.1184/R1/6572057.v1
46. Lindbergh, R. (2026). NASA appropriations and authorizations: A glance (CRS Report R43419). Congressional Research Service. https://crsreports.congress.gov/product/pdf/R/R43419
47. Maier, Mark W. "Architecting Principles for Systems-of-Systems." Systems Engineering 1, no. 4 (1998): 267–284. https://incose.onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291520-6858%281998%291%3A4%3C267%3A%3AAID-SYS3%3E3.0.CO%3B2-D
48. Marcia Smith. "Boeing's *Starliner* CFT Docks with ISS Despite Thruster Issues." SpacePolicyOnline.com. June 6, 2024. https://spacepolicyonline.com/news/boeings-*Starliner*-cft-docks-with-iss-despite-thruster-issues/
49. Marcia Smith. "NASA Classifies *Starliner* CFT as Type-A Mishap, Leadership Changes Coming." SpacePolicyOnline.com. February 19, 2026. https://spacepolicyonline.com/news/nasa-classifies-*Starliner*-cft-as-type-a-mishap-leadership-changes-coming/
50. Marcia Smith. "*Starliner*'s Crew Flight Test Comes to an End." SpacePolicyOnline.com. September 7, 2024. https://spacepolicyonline.com/news/*Starliner*s-crew-flight-test-comes-to-an-end/
51. McManus, H., & Hastings, D. (2005). A framework for understanding uncertainty and its mitigation and exploitation in complex systems. INCOSE International Symposium, 15(1), 484-503. https://doi.org/10.1002/j.2334-5837.2005.tb00685.x
52. Mezirow, J. (1991). Transformative dimensions of adult learning. Jossey-Bass. https://eric.ed.gov/?id=ED353469
53. NASA Office of Inspector General. (2021). NASA's management of the *Artemis* missions (IG-22-003). https://oig.nasa.gov/office-of-inspector-general-oig/audit-reports/NASA's-management-of-the-*Artemis*-missions/
54. NASA Office of Inspector General. (2023). NASA's partnerships with international space agencies for the *Artemis* campaign (IG-23-004). https://oig.nasa.gov/office-of-inspector-general-oig/audit-reports/NASA's-partnerships-with-international-space-agencies-for-the-*Artemis*-campaign/
55. NASA Office of Inspector General. (2024). 2024 report on NASA's top management and performance challenges. https://oig.nasa.gov/office-of-inspector-general-oig/2024-report-on-NASA's-top-management-and-performance-challenges/
56. NASA Office of Inspector General. (2024). NASA's management of the *Artemis* supply chain (IG-24-003). https://oig.nasa.gov/office-of-inspector-general-oig/audit-reports/NASA's-management-of-the-*Artemis*-supply-chain/
57. NASA Office of Inspector General. (2024). NASA's readiness for the *Artemis* II crewed mission to the Moon (IG-24-011). NASA Office of Inspector General. https://oig.nasa.gov/audits/NASA's-readiness-for-the-*Artemis*-ii-crewed-mission-to-the-moon/
58. NASA Office of Inspector General. (2024). NASA's management of major programs: 2024 annual report. https://oig.nasa.gov/office-of-inspector-general-oig/audit-reports/NASA's-management-of-major-programs-2024-annual-report/
59. NASA Office of Inspector General. (2024). NASA's management of Space Launch System Block 1B development (IG-24-015). https://oig.nasa.gov/audits/NASA's-management-of-space-launch-system-block-1b-development/
60. NASA Office of Inspector General. (2024, January 17). NASA's *Artemis* campaign: Key challenges facing NASA's *Artemis* campaign (CT-24-001). https://oig.nasa.gov/office-of-inspector-general-oig/ct-24-001/
61. NASA Office of Inspector General. (2026). NASA's acquisition of next-generation spacesuit services (IG-26-006). https://oig.nasa.gov/audits/NASA's-acquisition-of-next-generation-spacesuit-services/
62. NASA Office of Inspector General. (2026). NASA's management of the Human Landing System contracts (IG-26-004). https://oig.nasa.gov/office-of-inspector-general-oig/audit-reports/NASA's-management-of-the-human-landing-system-contracts/
63. National Aeronautics and Space Act of 1958, Pub. L. No. 85-568, 72 Stat. 426. https://www.govinfo.gov/content/pkg/STATUTE-72/pdf/STATUTE-72-Pg426.pdf
64. National Aeronautics and Space Administration Authorization Act of 2010, Pub. L. No. 111–267, 124 Stat. 2805 (2010).

https://www.govinfo.gov/content/pkg/PLAW-111publ267/html/PLAW-111publ267.htm
65. National Aeronautics and Space Administration Authorization Act of 2022, Pub. L. No. 117-167, div. B, tit. VII, 136 Stat. 1366. https://www.congress.gov/bill/117th-congress/house-bill/4346
66. National Aeronautics and Space Administration Engineering and Safety Center. (2024, December 24). Characterizing the visual experience of astronauts at the lunar South Pole. https://www.nasa.gov/centers-and-facilities/nesc/characterizing-the-visual-experience-of-astronauts-at-the-lunar-south-pole/
67. National Aeronautics and Space Administration History Office. (2004). *Columbia* Accident Investigation Board synopsis. https://history.nasa.gov/*Columbia*/Troxell/*Columbia*%20Web%20Site/Documents/Congress/House/2004/CHRG-108hhrg96086/CHRG-108hhrg96086.pdf
68. National Aeronautics and Space Administration Human System Risk Board. (2024). Risk of inadequate mission rescue (RIRM): Human spaceflight risk technical report. NASA HQ. https://www.nasa.gov/directorates/esdmd/hhp/human-system-risk-board/
69. National Aeronautics and Space Administration Manned Spacecraft Center. (1969). *Apollo* 11 mission report (MSC-00190). National Aeronautics and Space Administration. https://ntrs.nasa.gov/citations/19700007734
70. National Aeronautics and Space Administration Office of Safety and Mission Assurance. (2014). The cost of silence: Normalization of deviance and groupthink. National Aeronautics and Space Administration. https://sma.nasa.gov/
71. National Aeronautics and Space Administration Office of the Chief Engineer. (2020). NASA systems engineering handbook (NASA/SP-2016-6105 Rev. 2). National Aeronautics and Space Administration. https://www.nasa.gov/reference/nasa-systems-engineering-handbook/
72. National Aeronautics and Space Administration Office of the Chief Financial Officer. (n.d.). OCFO appropriations and authorizations. https://www.nasa.gov/ocfo/appropriations-and-authorizations/
73. National Aeronautics and Space Administration Office of the Chief Financial Officer. (n.d.). OCFO functions / financial management policy. https://www.nasa.gov/ocfo/ocfo-functions/
74. National Aeronautics and Space Administration. "NASA, Boeing Welcome *Starliner* Spacecraft to Earth, Close Mission." Press release. September 7, 2024. https://www.nasa.gov/news-release/nasa-boeing-welcome-*Starliner*-spacecraft-to-earth-close-mission/
75. National Aeronautics and Space Administration. "What You Need to Know about NASA's Boeing Orbital Flight Test-2." NASA Human Spaceflight. May 16, 2022. https://www.nasa.gov/humans-in-space/what-you-need-to-know-about-NASA's-boeing-orbital-flight-test-2/
76. National Aeronautics and Space Administration. (1975). *Apollo* program summary report (NASA SP-350). National Aeronautics and Space Administration. https://ntrs.nasa.gov/citations/19760005190
77. National Aeronautics and Space Administration. (1988). Space *Shuttle* program: Space transportation system reference manual. https://ntrs.nasa.gov/citations/19880008544
78. National Aeronautics and Space Administration. (2008). *Columbia* crew survival investigation report. https://www.nasa.gov
79. National Aeronautics and Space Administration. (2011). NASA risk management handbook (NASA/SP-2011-3422). https://ntrs.nasa.gov/citations/20110008236
80. National Aeronautics and Space Administration. (2018). Kilopower reactor using Stirling technology (KRUSTY) demonstration materials. National Aeronautics and Space Administration. https://www.nasa.gov/directorates/stmd/kilopower/
81. National Aeronautics and Space Administration. (2019/2029). Knowledge Policy for Programs and Projects w/Change 1 (NPD 7120.6A). https://nodis3.gsfc.nasa.gov/displayDir.cfm?c=7120&s=6A&t=NPD
82. National Aeronautics and Space Administration. (2020). NASA update on Orbital Flight Test review team. https://www.nasa.gov/news-release/nasa-boeing-to-provide-update-on-*Starliner*-orbital-flight-test-reviews/
83. National Aeronautics and Space Administration. (2020). NASA's lunar exploration program overview. https://www.nasa.gov/wp-content/uploads/2020/12/*Artemis*_plan-20200921.pdf
84. National Aeronautics and Space Administration. (2020). The *Artemis* Accords: Principles for cooperation in the civil exploration and use of the Moon, Mars, comets, and asteroids for peaceful purposes. https://www.nasa.gov/wp-content/uploads/2022/11/*Artemis*-Accords-signed-13Oct2020.pdf
85. National Aeronautics and Space Administration. (2020, February 7). NASA shares initial findings from Boeing *Starliner* Orbital Flight Test investigation. Commercial Crew Program Blog. https://blogs.nasa.gov/commercialcrew/2020/02/07/nasa-shares-initial-findings-from-boeing-*Starliner*-orbital-flight-test-investigation/

86. National Aeronautics and Space Administration. (2020, January 3). 50 years ago: NASA cancels *Apollo* 20 mission. https://www.nasa.gov/history/50-years-ago-nasa-cancels-*Apollo*-20-mission/
87. National Aeronautics and Space Administration. (2021). NPR 7120.5F: NASA space flight program and project management requirements. https://nodis3.gsfc.nasa.gov/displayDir.cfm?Internal_ID=N_PR_7120_005F_&page_name=main
88. National Aeronautics and Space Administration. (2022, May 16). What you need to know about NASA's Boeing Orbital Flight Test-2. https://www.nasa.gov/
89. National Aeronautics and Space Administration. (2022/2027). Strategic Workforce Planning (NPD 3010.1B). https://nodis3.gsfc.nasa.gov/displayDir.cfm?c=3010&s=1B&t=NPD
90. National Aeronautics and Space Administration. (2022/2027). Technical Standards for NASA Programs and Projects (NPR 7120.10B). https://nodis3.gsfc.nasa.gov/displayDir.cfm?c=7120&s=10B&t=NPR
91. National Aeronautics and Space Administration. (2023). *Artemis* architecture document: Version 1.0. https://www.nasa.gov/wp-content/uploads/2023/04/*Artemis*-architecture-document-v1-0.pdf
92. National Aeronautics and Space Administration. (2023). NASA-STD-3001, Volume 1: Spaceflight human-system standard—Crew health (Revision C). https://standards.nasa.gov/standard/nasa/nasa-std-3001-vol-1
93. National Aeronautics and Space Administration. (2023). National Aeronautics and Space Act. https://www.nasa.gov/history/national-aeronautics-and-space-act-of-1958/
94. National Aeronautics and Space Administration. (2023). NPR 7123.1D: NASA systems engineering processes and requirements. https://nodis3.gsfc.nasa.gov/displayDir.cfm?Internal_ID=N_PR_7123_001D_&page_name=Chapter1
95. National Aeronautics and Space Administration. (2023). Orion reference guide. https://www.nasa.gov/wp-content/uploads/2023/02/orion-reference-guide.pdf
96. National Aeronautics and Space Administration. (2023, April 6). New program office leads NASA's path forward for Moon, Mars. https://www.nasa.gov/people-of-nasa/new-program-office-leads-NASA's-path-forward-for-moon-mars/
97. National Aeronautics and Space Administration. (2023/2028). Partnership Agreements – Financial Requirements and Administration (Updated with Change 2) (NPR 9090.1C). https://nodis3.gsfc.nasa.gov/displayDir.cfm?c=9090&s=1C&t=NPR
98. National Aeronautics and Space Administration. (2024). Human-rating requirements for space systems (NPR 8705.2C). https://nodis3.gsfc.nasa.gov/displayDir.cfm?c=8705&s=2C&t=NPR
99. National Aeronautics and Space Administration. (2024). Mishap and close call reporting, investigating, and recordkeeping (NPR 8621.1D). https://nodis3.gsfc.nasa.gov/displayDir.cfm?Internal_ID=N_PR_8621_001D_&page_name=main
100. National Aeronautics and Space Administration. (2024). NPD 1000.0C: NASA Governance and Strategic Management Handbook. https://nodis3.gsfc.nasa.gov/displayDir.cfm?Internal_ID=N_PD_1000_000C_&page_name=main
101. National Aeronautics and Space Administration. (2024). NPR 9470.1A: Budget execution. https://nodis3.gsfc.nasa.gov
102. National Aeronautics and Space Administration. (2024, July 17). NASA ends VIPER project, continues Moon exploration. https://www.nasa.gov/news-release/nasa-ends-viper-project-continues-moon-exploration/
103. National Aeronautics and Space Administration. (2024, November 14). South Pole safety: Designing the NASA lunar rescue system. https://www.nasa.gov/get-involved/south-pole-safety-designing-the-nasa-lunar-rescue-system/
104. National Aeronautics and Space Administration. (2025). NASA systems modeling handbook for systems engineering (NASA-HDBK-1009A). https://standards.nasa.gov/standard/NASA/NASA-HDBK-1009
105. National Aeronautics and Space Administration. (2026). *Artemis* II: NASA's first crewed lunar flyby in 50 years. National Aeronautics and Space Administration. https://www.nasa.gov/mission/*Artemis*-ii/
106. National Aeronautics and Space Administration. (2026). *Artemis* III. National Aeronautics and Space Administration. https://www.nasa.gov/mission/*Artemis*-iii
107. National Aeronautics and Space Administration. (2026). Budgets, plans and reports. National Aeronautics and Space Administration. https://www.nasa.gov/budgets-plans-and-reports/
108. National Aeronautics and Space Administration. (2026). Fiscal year 2027 budget request. https://www.nasa.gov/fiscal-year-2027-budget-request/
109. National Aeronautics and Space Administration. (2026). Ignition. NASA. https://www.nasa.gov/ignition/

110. National Aeronautics and Space Administration. (2026). Moon Base Users Guide. https://www.nasa.gov/
111. National Aeronautics and Space Administration. (2026). Moon to Mars architecture: Architecture definition documents. https://www.nasa.gov/moontomarsarchitecture-architecturedefinitiondocuments/
112. National Aeronautics and Space Administration. (2026). NASA organization. https://www.nasa.gov/organization/
113. National Aeronautics and Space Administration. (2026). *Starliner* propulsion system anomalies during the crewed flight test—Investigation report. National Aeronautics and Space Administration. https://www.nasa.gov/wp-content/uploads/2026/02/nasa-report-with-redactions-021926.pdf
114. National Aeronautics and Space Administration. (2026, April 10). NASA welcomes record-setting *Artemis* II Moonfarers back to Earth. National Aeronautics and Space Administration. https://www.nasa.gov/news-release/nasa-welcomes-record-setting-*Artemis*-ii-moonfarers-back-to-earth/
115. National Aeronautics and Space Administration. (2026, April 20). NASA on track for future missions with initial *Artemis* II assessments. https://www.nasa.gov/missions/nasa-on-track-for-future-missions-with-initial-*Artemis*-ii-assessments/
116. National Aeronautics and Space Administration. (2026, February 19). NASA releases report on *Starliner* Crewed Flight Test investigation. National Aeronautics and Space Administration. https://www.nasa.gov/news-release/nasa-releases-report-on-*Starliner*-crewed-flight-test-investigation/
117. National Aeronautics and Space Administration. (2026, February 27). NASA adds mission to *Artemis* lunar program, updates architecture. National Aeronautics and Space Administration. https://www.nasa.gov/news-release/nasa-adds-mission-to-*Artemis*-lunar-program-updates-architecture/
118. National Aeronautics and Space Administration. (2026, January 13). NASA, Department of Energy to develop lunar surface reactor by 2030. https://www.nasa.gov/news-release/nasa-department-of-energy-to-develop-lunar-surface-reactor-by-2030/
119. National Aeronautics and Space Administration. (2026, March 26). NASA unveils initiatives to achieve America's national space policy. https://www.jpl.nasa.gov/news/nasa-unveils-initiatives-to-achieve-americas-national-space-policy/
120. National Aeronautics and Space Administration. (2026, March 3). NASA strengthens *Artemis*: Adds mission, refines overall architecture. https://www.nasa.gov/directorates/esdmd/nasa-strengthens-*Artemis*-adds-mission-refines-overall-architecture/
121. National Aeronautics and Space Administration. (n.d.). *Artemis* IV. https://www.nasa.gov/mission/*Artemis*-iv/
122. National Aeronautics and Space Administration. (n.d.). Human Landing System. NASA. https://www.nasa.gov/humans-in-space/human-landing-system/
123. National Aeronautics and Space Administration. (n.d.). Human-rating and NASA-STD-3001. https://www.nasa.gov/general/human-rating-and-nasa-std-3001/
124. National Aeronautics and Space Administration. (n.d.). Lessons learned: Test as you fly / differences between test and flight environments. https://llis.nasa.gov
125. National Aeronautics and Space Administration. (n.d.). Lunar terrain vehicle. https://www.nasa.gov/suits-and-rovers/lunar-terrain-vehicle/
126. National Aeronautics and Space Administration. (n.d.). NPR 8715.3D: NASA general safety program requirements. https://explorers.larc.nasa.gov/2023APPROBE/pdf_files/NASA07153D.pdf
127. National Aeronautics and Space Administration. (n.d.). Nuclear flight safety. https://sma.nasa.gov/sma-disciplines/nuclear-flight-safety
128. National Aeronautics and Space Administration. (n.d.). Office of Safety and Mission Assurance: About. https://sma.nasa.gov/about
129. National Aeronautics and Space Administration. (n.d.). Orion recovery operations. https://www.nasa.gov/humans-in-space/exploration-ground-systems/orion-recovery/
130. National Aeronautics and Space Administration. (n.d.). Technical authority. Office of the Chief Engineer. https://www.nasa.gov/technical-authority/
131. National Aeronautics and Space Administration. (Various years). Space *Shuttle* mission reports and mission summaries. National Aeronautics and Space Administration. https://ntrs.nasa.gov/ | https://history.nasa.gov/sts.html
132. National Aeronautics and Space Administration. NASA-HDBK-8709.24, NASA Safety Culture Handbook. Official handbook / standards page. https://standards.nasa.gov/standard/NASA/NASA-HDBK-870924
133. National Aeronautics and Space Administration. NASA-STD-3001, Volume 2: Human Factors, Habitability, and Environmental Health. Revision D.

2025. https://standards.nasa.gov/standard/NASA/NASA-STD-3001_VOL_2
134. National Research Council. (2014). Pathways to exploration: Rationales and approaches for a U.S. program of human space exploration. National Academies Press. https://doi.org/10.17226/18801
135. Office of Management and Budget. (2024). Budget of the U.S. government, fiscal year 2025. https://www.whitehouse.gov/wp-content/uploads/2024/03/budget_fy2025.pdf
136. Oleson, S., et al. (2022). A deployable 40 kWe lunar fission surface power concept. NASA Technical Reports Server. https://ntrs.nasa.gov/citations/20220004670
137. Perrow, C. (1999). Normal accidents: Living with high-risk technologies. Princeton University Press. https://press.princeton.edu/books/paperback/9780691004129/normal-accidents
138. Pettersen, K. A., & Schulman, P. R. (2019). Drift, adaptation, resilience and reliability: Toward an empirical clarification. Safety Science, 117, 460-468. https://doi.org/10.1016/j.ssci.2016.03.004
139. Pidgeon, N. F., & O'Leary, M. (2000). Man-made disasters: Why technology and organizations (sometimes) fail. Safety Science, 34(1-3), 15-30. https://doi.org/10.1016/S0925-7535(00)00004-7
140. The Planetary Society. (2026). NASA budget. https://www.planetary.org/space-policy/nasa-budget
141. Poston, D. I. (2020). KRUSTY reactor design. NASA Technical Reports Server. https://ntrs.nasa.gov/api/citations/20205009350/downloads/03_Poston_KRUSTY_ReactorDesign.pdf
142. Presidential Commission on the Space *Shuttle Challenger* Accident. (1986). Report of the Presidential Commission on the Space *Shuttle Challenger* Accident (Vol. I). U.S. Government Printing Office. https://history.nasa.gov/rogersrep/genindex.htm
143. Rahimi, M., Xiong, W., Cleland-Huang, J., & Lutz, R. R. (2017). Diagnosing assumption problems in safety-critical products. In 2017 32nd IEEE/ACM International Conference on Automated Software Engineering (ASE) (pp. 473–484). IEEE. https://dl.acm.org/doi/10.5555/3155562.3155623
144. Ramirez, A. J., Cheng, B. H. C., Bencomo, N., & Sawyer, P. (2012). RELAXing claims: Coping with uncertainty while evaluating assumptions at run time. In Proceedings of the 15th International Conference on Model Driven Engineering Languages and Systems (pp. 53–69). Springer. https://link.springer.com/chapter/10.1007/978-3-642-33666-9_5
145. Rev. (2026, April 23). NASA budget hearing transcript. https://www.rev.com/transcripts/nasa-budget-hearing
146. Review of U.S. Human Space Flight Plans Committee. (2009). Seeking a human spaceflight program worthy of a great nation. https://www.nasa.gov/wp-content/uploads/2024/01/augustinecommsummary.pdf
147. Roeller, R., Lago, P., & van Vliet, H. (2006). Recovering architectural assumptions. Journal of Systems and Software, 79(4), 552–573. https://doi.org/10.1016/j.jss.2005.10.017
148. SAE International. (2016). AS9100D: Quality management systems—Requirements for aviation, space, and defense organizations. https://doi.org/10.4271/AS9100D
149. Sawyer Rosenstein. "Crew 9, *Starliner* CFT Astronauts Return to Earth aboard Dragon." NASA'SpaceFlight.com. March 18, 2025. https://www.NASA'spaceflight.com/2025/03/crew-9-return/
150. Turner, B. A., & Pidgeon, N. F. (1997). Man-made disasters (2nd ed.). Butterworth-Heinemann. https://www.sciencedirect.com/book/9780750620874/man-made-disasters
151. U.S. Congress. (2026). Joint explanatory statement for fiscal year 2026 appropriations. https://www.congress.gov/committee-print/119th-congress/house-committee-print/56466
152. U.S. Government Accountability Office. (2009). NASA: Constellation Program cost and schedule will remain uncertain until a sound business case is established (GAO-09-844). https://www.gao.gov/products/gao-09-844
153. U.S. Government Accountability Office. (2019). NASA Commercial Crew Program: Schedule uncertainty persists for start of operational missions to the International Space Station (GAO-19-504). https://www.gao.gov/assets/gao-19-504.pdf
154. U.S. Government Accountability Office. (2022). NASA lunar programs: Moon landing plans are advancing but challenges remain (GAO-22-105323). https://www.gao.gov/products/gao-22-105323
155. U.S. Government Accountability Office. (2023). NASA *Artemis* programs: Crewed Moon landing faces multiple challenges (GAO-24-106256). GAO. https://www.gao.gov/products/gao-24-106256
156. U.S. Government Accountability Office. (2023). Space Launch System: Cost transparency needed to monitor program affordability (GAO-23-105609). https://www.gao.gov/products/gao-23-105609
157. U.S. Government Accountability Office. (2024). NASA *Artemis* missions: Exploration Ground Systems should conduct a schedule risk analysis for *Artemis* IV (GAO-25-106943). https://www.gao.gov/products/gao-25-106943
158. U.S. Government Accountability Office. (2024). NASA lunar programs: Gateway program has made

progress but faces technical and management challenges (GAO-24-106878). https://www.gao.gov/products/gao-24-106878
159. U.S. Government Accountability Office. (2025). Cybersecurity: NASA needs to fully implement risk management (GAO-25-108138). https://www.gao.gov/products/gao-25-108138
160. U.S. Government Accountability Office. (2025). NASA: Assessments of major projects (GAO-25-107591). https://www.gao.gov/products/gao-25-107591
161. U.S. Senate Committee on Commerce, Science, and Transportation. (2026, March 4). Commerce Committee passes landmark NASA Authorization Act. https://www.commerce.senate.gov/press/dem/release/commerce-committee-passes-landmark-nasa-authorization-act/
162. United Nations Office for Outer Space Affairs. (1967). Treaty on principles governing the activities of states in the exploration and use of outer space, including the Moon and other celestial bodies. https://www.unoosa.org/oosa/en/ourwork/spacelaw/treaties/outerspacetreaty.html
163. United Nations Office for Outer Space Affairs. (2017). International space law: United Nations instruments. https://www.unoosa.org/oosa/en/ourwork/spacelaw/treaties.html
164. Vaughan, D. (1996). The *Challenger* launch decision: Risky technology, culture, and deviance at NASA. University of Chicago Press. https://press.uchicago.edu/ucp/books/book/chicago/C/bo22781921.html
165. Weick, K. E. (1993). The collapse of sensemaking in organizations: The Mann Gulch disaster. Administrative Science Quarterly, 38(4), 628-652. https://doi.org/10.2307/2393339
166. White House Office of Science and Technology Policy. (2026, April 14). NSTM-3: Guidance for a National Initiative for American Space Nuclear Power. https://www.whitehouse.gov/wp-content/uploads/2026/04/NSTM-3-2026_04_14-corrected.pdf
167. The White House. (2017). Presidential memorandum on reinvigorating America's human space exploration program (Space Policy Directive-1). Federal Register, 82 Fed. Reg. 59501. https://www.federalregister.gov/documents/2017/12/14/2017-27160/reinvigorating-americas-human-space-exploration-program
168. The White House. (2020). Presidential memorandum on the national strategy for space nuclear power and propulsion (Space Policy Directive-6). Federal Register, 85 Fed. Reg. 82367. https://www.federalregister.gov/documents/2020/12/21/2020-28155/national-strategy-for-space-nuclear-power-and-propulsion
169. Yang, C. (2018). Architectural assumptions and their management in software development [Doctoral dissertation, University of Groningen]. University of Groningen. https://research.rug.nl/en/publications/architectural-assumptions-and-their-management-in-software-develo/
170. Yang, C., Liang, P., & Avgeriou, P. (2018). Assumptions and their management in software development: A systematic mapping study. Information and Software Technology, 94, 82–110. https://doi.org/10.1016/j.infsof.2017.10.003. https://www.sciencedirect.com/science/article/abs/pii/S0950584916304189
171. Yang, C., Liang, P., Avgeriou, P., Eliasson, U., Heldal, R., Pelliccione, P., & Bi, T. (2017). An industrial case study on an architectural assumption documentation framework. Journal of Systems and Software, 134, 190-210. https://doi.org/10.1016/j.jss.2017.09.007
172. Yang, Chen, Peng Liang, and Paris Avgeriou. "A Survey on Software Architectural Assumptions." Journal of Systems and Software 113 (2016): 362–380. https://doi.org/10.1016/j.jss.2015.12.016. https://www.sciencedirect.com/science/article/abs/pii/S0164121215002824
173. Yang, Chen, Peng Liang, and Paris Avgeriou. "Evaluation of a Process for Architectural Assumption Management in Software Development." Science of Computer Programming 168 (2018): 38–70. https://doi.org/10.1016/j.scico.2018.08.002. https://www.sciencedirect.com/science/article/pii/S0167642318303253

Appendix A - Data Sources and Methodological Notes

Purpose. This appendix identifies the data sources, transformations, and interpretive limits associated with the budget, workforce, headcount, and author-created synthesis figures used in this paper. It supports traceability for the quantitative and framework-based claims in Sections 5, 6, 9, 10, and 11.

Scope. The appendix does not introduce a new argument. It documents the basis for figures and tables that support the paper's central claim: in complex human spaceflight campaigns, assumptions must remain visible, owned, and periodically revalidated as architecture, funding, workforce, governance, and operational context change.

Evidence Area	Primary Use in Paper	Source Basis	Interpretive Limit
NASA budget and workforce trends	Supports Sections 5.4.1 and 5.4.2; supports Figure 8 and Figure 9.	The Planetary Society historical NASA budget/workforce dataset; NASA historical budget tables; NASA budget request archives; ASAP workforce discussion where applicable.	Budget and workforce trends are structural indicators of institutional capacity, not direct measures of mission readiness.
Year-over-year workforce change	Supports discussion of institutional memory, workforce continuity, and abrupt expansion/contraction effects.	Derived from annual civil-service headcount values.	Year-over-year change captures the rate of change, not the skill composition or distribution of expertise.
JSC S&MA headcount	Supports Section 10.1 and Figure 13.	JSC S&MA/NC headcount source material; NASA civil-service headcount baseline; JSC civil-service population baseline.	Headcount shows scale and proportionality, but not workload complexity, contractor support, or informal technical capacity.
Historical near-miss tables	Supports Section 6.5 and Appendix E.	NASA mission reports, accident-board reports, NASA History Office material, and official post-mishap documentation.	Near misses are used as assumption signals, not as exhaustive accident investigations.
Author-created frameworks	Supports Figures 2, 3, 4, 5, 6, 7, 10, 11, and 12.	NASA systems engineering policy and handbooks; S&MA policy; INCOSE concepts; historical evidence synthesized by the author.	Frameworks are analytical models. They are not official NASA process diagrams unless explicitly identified as such.

Recommended citation treatment. The main paper should cite the original source at the first use of each figure or claim. This appendix may then be cited as a methodological note explaining how the evidence was organized, transformed, or interpreted.

Appendix B - NASA Budget, Workforce, and Headcount Data

This appendix contains the supporting data structure for Figure 8, Figure 9, and Figure 13. These figures are used to discuss institutional capacity, budget continuity, workforce continuity, and safety-governance load, the supporting data is being made visible in a structured format rather than left as an uncited chart image.

B.1 Era Summary Table

Period	Human Spaceflight Context	Budget / Workforce Pattern	Assumption-Management Relevance
1960s	*Apollo* buildup and lunar landing commitment.	Rapid expansion followed by peak fiscal and workforce intensity.	*Apollo*-era capability was supported by an extraordinary national resource condition that should not be assumed for *Artemis*.
1970s	Post-*Apollo* contraction, Skylab, *Apollo*-Soyuz, and *Shuttle* development.	Budget and workforce decline after *Apollo* peak.	Technical success does not guarantee sustained program continuity.
1980s	Operational *Shuttle* era and *Challenger*.	Reusable-spaceflight cadence expectations under programmatic pressure.	Cadence expectations can reshape risk interpretation.
1990s	*Shuttle* operations, Mir, ISS assembly transition.	Institutional continuity persisted, but with increasing complexity and aging systems.	Long-duration programs accumulate inherited assumptions.
2000s	*Columbia*, Return to Flight, and ISS assembly.	Safety reforms and inspection/repair logic emerged under institutional and operational stress.	Reform can improve controls while new assumptions continue to form.
2010s	Commercial crew, SLS/Orion development, *Artemis* formation.	Distributed development and commercial integration became more central.	Cross-program and provider maturity assumptions became more important.
2020s	*Artemis*, Moon-to-Mars planning, commercial HLS, xEVAS, surface infrastructure, and S&MA capacity concerns.	Campaign complexity grows while workforce and assurance capacity remain constrained relative to mission scope.	Budget, workforce, and governance continuity are part of the assumption environment, not background context.

B.2 Annual NASA Budget and Workforce Dataset

The budget and workforce figures presented in this appendix were compiled using two primary sources:

1. The Planetary Society's Historical NASA Budget Dataset, which provides both nominal (current-year) and inflation-adjusted budget data for NASA across the full historical record. This dataset includes appropriated funding levels, constant-dollar conversions, and long-term budget trends derived from publicly released NASA documentation.
2. NASA Historical Workforce Data, drawn from internal workforce tables and historical records contained within the uploaded NASA datasets. These include annual full-time equivalent (FTE) civil-service counts, year-over-year workforce changes, and related operational metrics.

The combined dataset enables year-to-year comparison of NASA's financial resources and workforce levels from 1960 through 2026, presented in a consistent format for analysis. All inflation-adjusted budget values reflect the Planetary Society's use of NASA's New Start Index conversions, ensuring comparability across decades.

Year	Nominal Budget ($BN)	Inflation-Adjusted Budget ($BN)	Civil-Service Workforce	YoY Workforce Change
1960	523.0	8341.85	10284.0	991.0
1961	964.0	14898.62	17559.0	7275.0
1962	1825.2	27125.0	23824.0	6265.0
1963	3674.1	52752.94	29934.0	6110.0
1964	5100.0	70074.0	32499.0	2565.0
1965	5250.0	69762.0	34049.0	1550.0
1966	5175.0	64873.79	35708.0	1659.0
1967	4968.0	59372.56	35860.0	152.0
1968	4588.9	52028.94	34641.0	-1219.0
1969	3995.2	42857.29	33929.0	-712.0
1970	3749.2	37623.38	32548.0	-1381.0
1971	3312.6	31271.12	30506.0	-2042.0
1972	3310.1	29562.69	28382.0	-2124.0
1973	3407.6	28791.23	26777.0	-1605.0
1974	3039.7	23958.91	26007.0	-770.0
1975	3231.1	22983.13	25638.0	-369.0
1976	3551.8	23179.19	25426.0	-212.0
1977	3819.0	22498.25	24188.0	-1238.0
1978	4063.7	22208.12	23779.0	-409.0
1979	4558.7	22752.81	23360.0	-419.0
1980	5243.3	23637.12	23470.0	110.0
1981	5485.7	22464.30	22736.0	-734.0
1982	6020.0	22863.96	22310.0	-426.0
1983	6837.7	24410.58	22534.0	224.0
1984	7197.5	24377.93	21870.0	-664.0
1985	7552.2	24741.00	22316.0	446.0
1986	7656.0	24346.08	21960.0	-356.0
1987	10434.0	31875.87	22646.0	686.0
1988	8956.451	25982.66	22823.0	177.0
1989	10791.0	29869.48	22893.0	70.0
1990	12296.6	32573.62	24566.0	1673.0
1991	14016.0	35838.91	25741.0	1175.0
1992	14317.0	34804.62	25165.0	-576.0
1993	14310.0	33385.23	24731.0	-434.0
1994	14570.0	32971.91	24265.0	-466.0
1995	13854.0	30548.07	23075.0	-1190.0
1996	13886.0	29868.78	21555.0	-1520.0
1997	13711.0	29094.74	19883.0	-1672.0
1998	13637.0	28228.58	18924.0	-959.0
1999	13627.0	27540.16	18278.0	-646.0
2000	13588.0	26387.89	18183.0	-95.0
2001	14254.0	26711.99	18510.0	327.0
2002	14868.0	27104.37	18471.0	-39.0
2003	15449.0	27561.02	18518.0	47.0
2004	15342.0	26464.95	18633.0	115.0
2005	16187.0	27097.04	18624.0	-9.0
2006	16570.0	26876.54	18318.5	-305.5
2007	16285.0	25437.17	18013.0	-305.5
2008	17309.4	26102.57	18159.0	146.0
2009	18784.4	27800.91	18100.0	-59.0
2010	18719.0	27329.74	18420.0	320.0
2011	18432.0	26486.78	18578.0	158.0
2012	17773.0	25273.21	18145.0	-433.0
2013	16865.0	23627.87	17894.0	-251.0
2014	17646.0	24245.60	17715.0	-179.0
2015	18010.0	24259.47	17302.0	-413.0
2016	19285.0	25668.34	17075.0	-227.0
2017	19653.3	25608.25	17175.0	100.0
2018	20736.0	26355.45	17116.0	-59.0
2019	21500.0	26810.50	17011.0	-105.0
2020	22629.0	27607.38	17430.0	419.0
2021	23271.3	27367.02	17944.0	514.0
2022	24041.3	26782.00	17691.0	-253.0
2023	25383.7	27211.33	17657.0	-34.0
2024	24875.0	25820.25	17738.0	81.0
2025	24875.0	24875.0	16986.0	-752.0
2026	24438.0	24438.34	13738.0	-3248.0

B.3 Calculation Notes

- Inflation-adjusted values should identify the inflation basis used, such as constant-year dollars or the source's published inflation adjustment.
- Year-over-year workforce change should be calculated as current-year workforce minus prior-year workforce.
- If 2026 values are estimated, requested, projected, or derived from budget materials rather than final enacted data, label them explicitly.
- If multiple public NASA workforce values conflict, identify the conflict and state which value was used for the figure.
- Headcount should not be treated as a direct proxy for technical competence. It is used here as an indicator of capacity, continuity, and institutional margin.

B.4 JSC Safety and Mission Assurance Headcount Support

The table below supports the proportional comparison made in Figure 13. The purpose is to show the relationship among NASA-wide civil-service workforce, JSC civil-service workforce, and the JSC S&MA/NC engineering workforce supporting safety-governance interpretation.

Year	NASA Workforce	JSC Workforce	JSC S&MA / NC Headcount	S&MA as % of NASA	S&MA as % of JSC
1987	22646	3347	112.99	0.499	3.376
1988	22823	3392	127.05	0.557	3.746
1989	22893	3571	133.98	0.585	3.752
1990	24566	3609	132.99	0.541	3.685
1991	25741	3674	95.892	0.373	2.61
1992	25165	3620	91.01	0.362	2.514
1993	24731	3609	131.02	0.53	3.63
1994	24265	3205	112.06	0.462	3.497
2008	18159	3443	140.99	0.776	4.095
2009	18100	3449	147.0	0.812	4.262
2010	18420	3438	149.00	0.809	4.334
2011	18578	3399	155.00	0.834	4.56
2012	18145	3279	159.99	0.882	4.879
2013	17894	3206	159.99	0.894	4.991
2014	17715	3163	162.99	0.92	5.153
2015	17302	3065	154.99	0.896	5.057
2016	17075	3074	159.99	0.937	5.205
2017	17175	3110	166.00	0.967	5.338
2018	17116	3060	165.00	0.964	5.392
2019	17011	3050	159.99	0.941	5.246
2020	17430	3084	156.00	0.895	5.058
2021	17944	3101	158.00	0.881	5.095
2022	17691	3079	161.00	0.91	5.229
2023	17657	3130	161.00	0.912	5.144
2024	17738	3195	166.99	0.941	5.227
2025	16986	3249	175.00	1.03	5.386
2026	13738	2916	148.00	1.077	5.076

Appendix C — Assumption Control Register Template

This appendix provides a practical register for the assumption-management approach proposed in the paper. It converts the paper's analytic argument into a governable artifact. Assumptions that affect architecture, verification, operations, or crew survival should be visible, owned, reviewed, and retired or escalated when evidence or context changes.

A-001 — HLS Interface and Maturity

Assumption statement	HLS provider interface maturity will support the revised *Artemis* mission sequence.
Type / program element	Commercial, interface, and schedule assumption affecting HLS, Orion, and Mission Operations.
Evidence basis	Provider test data, NASA integration reviews, docking/rendezvous evidence, and certification artifacts.
Review trigger	Provider delay, interface change, docking-test issue, revised mission sequence, or certification concern.
Crew-survival consequence	Loss of mission sequence, degraded abort posture, or reduced confidence in return-to-orbit assumptions.
Status	Open; campaign-level owner required.

A-002 — xEVAS Readiness

Assumption statement	xEVAS readiness will support planned EVA duration, mobility, contingency return, and surface exposure limits.
Type / program element	Survival, operations, and human-systems assumption affecting xEVAS, HLS, and surface operations.
Evidence basis	Suit qualification data, integrated EVA testing, thermal/vacuum testing, and human-in-the-loop evaluations.
Review trigger	Suit delay, mobility issue, thermal issue, consumables concern, or interface mismatch with HLS.
Crew-survival consequence	Crew exposure, loss of mobility, reduced rescue options, or degraded surface survival.
Status	Open; survival-critical readiness trigger required.

A-003 — Surface Communications

Assumption statement	Surface communications will support anomaly diagnosis, medical consultation, command, telemetry, and time-critical decision support.
Type / program element	Infrastructure and operations assumption affecting communications and mission operations.
Evidence basis	Communications architecture, coverage analysis, latency analysis, and integrated mission simulations.
Review trigger	Coverage gap, latency issue, architecture change, provider-interface change, or surface-route change.
Crew-survival consequence	Reduced anomaly-response capability and degraded ground support during crew contingency.
Status	Open; mission-level owner required.

A-004 — Surface Power Availability

Assumption statement	Surface power will be available in the location, configuration, and maturity state required by the mission.
Type / program element	Infrastructure and survival assumption affecting surface power, habitation, mobility, and communications.
Evidence basis	Power-system test data, deployment plan, operations concept, logistics analysis, and maintenance analysis.
Review trigger	Power-system slip, reduced output, landing-site change, thermal-control issue, or mission-duration increase.
Crew-survival consequence	Loss of safe-haven capability, degraded thermal control, reduced communications, or shortened surface survivability.
Status	Open; infrastructure maturity review required.

A-005 — Safety Governance Capacity

Assumption statement	NASA safety-governance capacity will be sufficient to steward cross-program assumptions across Orion, HLS, xEVAS, surface systems, and commercial providers.
Type / program element	Governance, workforce, and integration assumption affecting S&MA, Technical Authority, and mission decision authority.
Evidence basis	S&MA staffing data, workload assessment, review-board cadence, open-risk inventory, and integration-review outcomes.
Review trigger	Staffing decline, review backlog, increased provider interfaces, architecture reset, or new mission objective.
Crew-survival consequence	Assumptions may remain locally verified but systemically incomplete, increasing crew-survival interpretation risk.
Status	Open; workforce and assurance capacity should be tracked as safety-governance conditions.

Recommended use. Each assumption should be reviewed at major architecture changes, safety reviews, mission readiness reviews, certification milestones, provider schedule changes, and post-flight learning cycles. Assumptions that cross program boundaries or affect crew-survival logic should not remain local artifacts.

Appendix D — *Artemis* Campaign Assumption Exposure Matrix

This matrix summarizes the major assumption carriers identified in the paper and links them to the areas where they can affect architecture, verification, operations, and crew-survival confidence.

Campaign Area	Assumption Being Carried	Why It Matters	Governance Need
HLS	Commercial lander maturity, interface readiness, docking/rendezvous compatibility, ascent reliability, and provider schedule credibility.	HLS is not an isolated lander. It defines lunar descent, surface access, ascent, abort logic, and return-to-Orion assumptions.	Campaign-level assumption owner; integrated evidence review; provider-interface maturity gate.
xEVAS	Suit readiness will support surface operations, contingency movement, thermal protection, consumables, and HLS interface compatibility.	The spacesuit is a survival system, not simply an EVA productivity asset.	Survival-critical readiness trigger; integrated suit-HLS-surface operations validation.
Surface Power	Power will be available, deployed, located, and mature enough to support mission duration and contingency needs.	Power supports communications, thermal control, charging, habitation, instruments, and emergency response.	Infrastructure maturity review; readiness state explicitly separated as planned, demonstrated, deployed, or mission available.
Communications	Surface and mission communications will support command, telemetry, medical consultation, navigation, anomaly response, and decision support.	Communications are part of survival logic because the ground may be needed to diagnose and manage anomalies.	Coverage and latency assumptions owned at mission level; integrated communications simulation required.
Gateway / Staging Logic	Updated architecture preserves staging, logistics, rescue, and contingency meaning even if Gateway is delayed, descoped, or reinterpreted.	Gateway-related assumptions affect abort, aggregation, logistics, and mission-sequence interpretation.	Architecture-authority review whenever staging assumptions change.
Budget Continuity	Funding stability will support planned scope, verification effort, workforce continuity, and infrastructure maturation.	Budget conditions shape whether assumptions remain valid long enough to support the campaign.	Budget-dependent assumptions reopened at proposal, appropriation, execution, and rebaseline points.
Workforce / S&MA Capacity	NASA retains sufficient technical memory and safety-governance capacity to challenge inherited assumptions.	Assumption control requires people who understand context, caveats, interfaces, and prior decisions.	Workforce continuity tracked as a safety-governance condition, not only an HR metric.

Appendix E — Historical Near-Miss Evidence Matrix

This appendix supports the historical analysis in Section 6.5. These cases are not presented as full accident investigations. They are included because they reveal how assumptions can remain active after reforms, survive because of contingency performance, or become absorbed into narratives of mission success.

Historical Period	Representative Events	Assumption Signal	Why It Belongs in the Appendix
Post-*Apollo* 1 to Pre-*Challenger*	*Apollo* 12 lightning strike; Skylab 3 oxidizer leak; *Apollo*-Soyuz toxic exposure; STS-1 TPS and overpressure effects; STS-51D improvised EVA and landing damage; STS-51F abort-to-orbit.	Reforms can be real but remain bounded by the causal frame of the prior mishap. Adjacent hazards can persist outside the dominant lesson learned.	Shows that assumption drift may continue in areas not directly addressed by the original accident-board reform package.
Post-*Challenger* to Pre-*Columbia*	STS-27 tile damage; Mir-Progress collision; STS-93 propulsion and electrical issues.	Rogers-era reforms did not eliminate all adjacent pathways of vulnerability, especially where maintenance, debris, propulsion, and operational-coupling assumptions persisted.	Shows the difference between implementing a reform and absorbing its deeper organizational meaning.
Post-*Columbia* to *Artemis* Era	STS-114 foam shedding and gap-filler repair; STS-117 contingency blanket repair; STS-120 solar-array repair; additional recovery and contingency cases as appropriate.	Inspection and repair capability improved after *Columbia*, but mission safety continued to depend on contingency performance and reinterpretation of anomalies.	Shows that recovery capability can strengthen safety while also masking persistent fragility if treated as proof that the underlying hazard is controlled.

Appendix F — Figure and Framework Methodology

Several figures in this paper are author-created synthesis figures. This appendix distinguishes source-derived figures from analytical frameworks to avoid implying that an author-created model is an official NASA diagram.

Figure	Figure Type	Source Basis	Purpose / Interpretation
Figure 1	Conceptual comparison	NASA *Apollo*, *Shuttle*, and *Artemis* architecture sources; systems-of-systems concepts.	Author-created synthesis showing the shift to campaign architecture.
Figure 2	Process model	NASA systems engineering and risk-management guidance; assumption-management logic.	Author-created model showing the assumption-management cycle.
Figure 3	Lifecycle framework	NASA SE Handbook; NPR 7120.5; NPR 7123.1D; NPR 8705.2C; *Artemis* architecture material.	Author-created framework for assumption movement across architecture, verification, and operations – traceability map in Appendix G
Figure 4	Conceptual risk model	Historical mishap patterns and lifecycle-gate analysis.	Author-created model showing assumption hardening across decision gates.
Figure 5	Verification model	NASA verification/validation guidance and evidence-interpretation logic.	Author-created model showing verification context shift.
Figure 6	Conceptual map	NASA distributed program structure; systems-engineering verification logic; *Artemis* provider structure.	Author-created map showing distributed proof burden.
Figure 7	Dependency model	Moon-to-Mars architecture, lunar base planning, surface-power, and infrastructure sources.	Author-created model showing infrastructure dependency cascade.
Figure 8	Data visualization	Planetary Society / NASA budget and workforce data; supporting NASA budget sources.	Source-derived visualization with author interpretation.
Figure 9	Data visualization	Derived from workforce dataset.	Source-derived visualization showing year-over-year workforce change.
Figure 10	Causal-loop synthesis	Budget, workforce, governance, and program-fragility analysis.	Author-created synthesis showing programmatic fragility loop.
Figure 11	Conceptual synthesis	Section 5 structural drivers.	Author-created synthesis showing external pressure and assumption entrenchment.
Figure 12	Conceptual synthesis	Historical case analysis from *Apollo* 1, *Challenger*, *Columbia*, and near-miss patterns.	Author-created synthesis showing inherited meaning and structure.
Figure 13	Data visualization	JSC S&MA headcount source material and civil-service workforce baseline.	Source-derived visualization with author interpretation.

Appendix G – Master Assumption Traceability Map

This appendix provides a master assumption traceability map for the *Artemis* Exploration Campaign. The map connects major policy blocks, architecture decisions, safety decisions, verification gates, and operational decision points to the assumptions and constraints they carry, the evidence or artifacts that support them, and the principal risks or drift modes that may emerge if those assumptions are not actively governed.

The purpose of this appendix is to make the paper's assumption-management logic visible in one place. It does not introduce new argumentation. Instead, it functions as a traceability artifact showing how policy, architecture, verification, safety assurance, surface infrastructure, mission operations, and crew-survival assumptions interact across the campaign lifecycle.

The map should be read from left to right. "Element Type" identifies whether the item is a policy block, architecture block, decision gate, verification gate, or operational element. "Element Name" identifies the specific campaign element or decision point. "Assumptions / Constraints" states the assumption being carried. "Evidence / Artifacts" identifies the documents, reviews, standards, or program artifacts that currently support interpretation of the assumption. "Key Risks / Drift Modes" identifies how the assumption could degrade, propagate, or become misleading if the evidence base, architecture, policy environment, or operational context changes.

This appendix is intended to support campaign-level assumption stewardship. It helps distinguish between assumptions that are technically validated, assumptions that are administratively embedded, assumptions that are politically or programmatically stabilized, and assumptions that remain vulnerable to drift across lifecycle gates.

Element Type	Element Name	Assumptions / Constraints (short)	Evidence / Artifacts	Key Risks / Drift Modes
Block	SPD-1 / Reinvigorating America's Human Space Exploration Program	Assumes sustained lunar presence + program continuity; schedule pressure is acceptable within managed risk.	Policy directive language; *Artemis* program plans derived from policy intent.	Policy-to-engineering translation creates implicit requirements and hidden assumption debt.
Block	SPD-6 / National Strategy for Space Nuclear Power and Propulsion	Assumes nuclear systems can be developed and used safely; interagency roles and launch approval process apply.	SPD-6 memo and strategy; launch approval guidance; nuclear flight safety processes.	Overreliance on TRL; confidence exceeds completeness when testing is limited.
Block	*Artemis* Accords	Assumes partner interoperability and data sharing sufficient for integrated safety decisions.	Accords principles applied through program agreements and interface governance.	Interface blind spots and inconsistent safety interpretation across partners.
Block	Critical Policy Qs (risk tolerance thresholds)	Assumes risk thresholds are stable; public and program tolerance understood.	Human rating and safety acceptance frameworks; approval processes.	Political drift changes risk acceptance; silent re-scoping of acceptable outcomes.

Block	NASA VSE Oversight & Safety Assurance	Assumes independence is preserved; oversight acts at interpretation layer not component layer.	Standards tailoring; IV&V planning; hazard analysis; Certification of Flight Readiness; monitoring/audits.	Oversight can become procedural; loss of cross-lifecycle meaning checks.
Decision	G1: Architecture Decision	Assumes architecture interfaces sufficiently defined; downstream V&V will catch residual risk.	Mission concept review inputs; architecture trade studies; interface definitions.	Lock-in of flawed assumptions; deferred hazard recognition and evidence gaps.
Block	*Artemis* Architecture & Orion Integration	Assumes composability of subsystems; interfaces stable across phases; distributed ownership manageable.	Architecture docs; Orion references; integration ICDs (program artifacts).	Interface drift across timeline; orphaned assumptions as components evolve asynchronously.
Block	Safety-Critical Vehicle Subsystems	Assumes subsystem verification implies system safety; environmental bounds apply.	Subsystem verification artifacts; FMEA/CIL; hazard analyses; requirements.	Anomaly without single failure; emergent coupling; confidence > completeness.
Decision	G2: Safety Decision (CDR/Qualification)	Assumes verification evidence retains meaning in current integration context; nuclear constraints satisfied.	Qualification test evidence; safety reviews; nuclear safety review steps; human rating logic.	Verification meaning degrades under context shift; untested interface interactions.
Block	Surface Nuclear Power / Fission Surface Power	Assumes shielding/thermal/autonomy constraints remain bounded; integration is manageable.	FSP program docs; KRUSTY evidence as technical lineage; nuclear flight safety protocols.	Shielding environment overlays whole system; long-duration drift; limited test representativeness.
Block	Verification & Assurance Evidence	Assumes evidence is applicable across lifecycle; closure implies operational validity.	Integrated test and checkout; safety reviews; human rating certification packages.	Evidence persists while assumptions drift; interpretation gap at ops decision points.
Gate	Interpretation Gate: Meaning of Verification	Assumes evidence must be re-interpreted under new context; not re-verified.	Evidence provenance; applicability statements; anomaly data and lessons learned.	False confidence when evidence is reused outside original scope; phase-dependent meaning.
Decision	G3: Operational Decision	Assumes operational context matches verified context; procedures cover off-nominal states.	Flight rules; launch commit criteria; crew procedures; readiness reviews.	Operational drift; mismatch between procedure assumptions and physical reality.
Block	Mission Ops (Pre-Launch/Ascent, Translunar/Orbit, Surface, Return/Recovery)	Assumes training and procedures mitigate emergent states; logistics and comm windows adequate.	Crew procedures; operations concepts; mission rules and constraints.	Off-nominal conditions stress inherited assumptions; anomaly without failure.

Block	Mission Ops: Execution	Assumes real-time decision-making aligns with certified rules and risk posture.	Telemetry, monitoring, operational decision logs.	Normalization of deviance; silent workarounds; data without interpretation.
Block	Anomaly / V&V Action / Political Drift (MIB etc.)	Assumes escalation triggers are defined; governance responds appropriately.	Anomaly reports; corrective action; recertification actions.	Assumption-driven anomalies misclassified as isolated events; political drift changes thresholds.
Block	Emergent Risk & Feedback Loop	Assumes lessons learned are captured and reintegrated into requirements, verification planning, and standards.	Data collection; root cause analysis; knowledge management; policy updates.	Failure to institutionalize learning; orphaned assumptions persist across programs.
Context Layer	Assumption Lifecycle & Knowledge Validity (Created→Inherited→Stressed→Orphaned)	Assumes lifecycle transitions change context faster than artifacts update.	Assumption registers; applicability statements; lessons learned.	Orphaned assumptions produce emergent risk without component failure.
Context Layer	Verification Scope Limit: Radiation & Shielding Environment	Assumes radiation environment affects crew, avionics, and verification applicability across phases.	Radiological analyses; shielding constraints; nuclear flight safety requirements.	Underestimated dose pathways; avionics upset; constraints treated as local not system-wide.
Arrow	Policy/Accords → G1 Architecture Decision	Policy intent becomes architecture constraints; implicit assumptions enter requirements.	Policy docs; architecture concept reviews.	Misinterpretation: policy treated as requirement; schedule pressure locks assumptions.
Arrow	VSE Oversight → G1/G2/G3 (dotted)	Independent oversight informs gate decisions.	Hazards, IV&V, readiness certification artifacts.	Oversight bypassed or reduced to compliance checks.
Arrow	Architecture & Integration → Subsystems	Interfaces and allocations drive subsystem requirements and verification scope.	ICDs; requirements allocation; verification planning.	Interface drift; allocation assumptions become orphaned.
Arrow	Subsystems → G2 Safety Decision	Subsystem evidence aggregated for safety qualification.	Qualification evidence; hazard closure packages.	Confidence exceeds completeness; missing cross-coupling tests.
Arrow	G2 → Verification & Assurance Evidence	Safety decision defines evidence thresholds and closure criteria.	Certification packages; review outcomes.	Closure criteria do not capture context drift; evidence meaning assumed stable.
Arrow	Verification Evidence → Interpretation Gate	Evidence package must be re-read in current operational context.	Evidence provenance, applicability, anomaly history.	Evidence reused outside scope without re-interpretation.
Arrow	Interpretation Gate → G3 Operational Decision	Meaning check informs operational go/no-go.	Flight rules, LCC, procedures, waiver logic.	Operational decisions made on stale assumptions.

Arrow	Mission Ops Execution → Emergent Risk & Feedback Loop	Operational data feeds learning and updates.	Telemetry, anomaly reports, lessons learned processes.	Data not converted into updated assumptions; organizational memory loss.
Arrow	Emergent Risk & Feedback → Policy/Standards Update	Feedback influences future requirements and standards.	Standards updates; policy alignment updates.	Lessons remain local; no systemic update.