The Test Like You Fly and Test What You Fly Approach for the Artemis Human Spaceflight Paradigm

Tracy R. Gill[1], John S. Gurecki[2], Dr. Eirik Holbert[3], Syrus K Jeanes[4], Robert M. Peacock[5], and Joseph M. Schuh[6]
NASA, Kennedy Space Center, FL, 32899

Ben Jimenea[7]
NASA, Headquarters, Washington DC, FL, 20546

James R. Debruin[8]
NASA, Johnson Space Center, FL, 77058

Jody H. Fluhr[9]
Booz Allen Hamilton, Johnson Space Center, FL, 77058

Christy Gattis[10]
NASA, Marshall Space Flight Center, 35812

The principles of Test Like You Fly (TLYF) and Test What You Fly (TWYF) seem self-explanatory—the more closely ground testing can replicate flight conditions and mission operations, the less risk of unexpected issues during execution of the mission. However, for every spaceflight program, a multitude of factors dictate the extent to which that principle can be followed. Therefore, the TLYF process involves a continual assessment to maximize the effectiveness of ground testing, given the limitations, and to characterize the remaining risk to mission success.

An effective TLYF approach relies on test configurations that represent the actual mission configurations with sufficient fidelity, including the vehicle configurations and the operational environment. This paper will discuss the TLYF assessment, exception evaluation, and risk acceptance process as it applies to integrated testing between multiple elements whose size and complexity may preclude the possibility of validating their interfaces on the ground in their final flight configurations.

In the Artemis paradigm, starting with the Artemis III mission, there will be separate government launched elements including Orion and Lunar Gateway elements, and

[1] Systems Engineer, Exploration Ground Systems/Mail Code LX-F, KSC, FL 32899.
[2] Operations Engineer, Exploration Research and Technology/Mail Code UB-C, KSC, FL 32899
[3] Operations Engineer, Exploration Research and Technology/Mail Code UB-C, KSC, FL 32899.
[4] Systems Engineer, Engineering Directorate/Mail Code NE-TE, KSC, FL 32899
[5] Systems Engineer, Engineering Directorate/Mail Code NE-C5, KSC, FL 32899.
[6] Systems Engineer, Exploration Ground Systems/Mail Code LX-F, KSC, FL 32899.
[7] Project Engineer, Exploration Systems and Operations, Mail Code BG01, Washington, DC 20546.
[8] Systems Engineer, Gateway Program, Mail Code JSC-EX, Houston TX, 77058.
[9] Systems Engineer, Moon2Mars Systems Engineering and Integration, Mail Code JSC-XE, Houston TX, 77058.
[10] Supervisor, Aerospace Flight Systems, Mail Code EE04, Huntsville, AL 35812.

commercially launched elements including lunar landers, surface elements such as rovers and habitats, and other Gateway elements. Additionally, there will be spacesuits, logistics, and utilization elements which may be launched on a variety of vehicles. Because of this complex architecture of distributed manufacture, test, and launch, where elements would not otherwise come together pre-flight, opportunities to test hardware together pre-flight must be purposefully driven.

This paper will address the TLYF approach based on case studies of past implementation and outcomes. It will then recommend a standardized process across the Artemis campaign, using a general rubric for scoring exceptions and their importance. To address the importance of TLYF as a process, this paper will conclude with a detailed presentation of a consistent process for assessing both TLYF and TWYF aspects to devise test campaigns that make best effort toward the highest fidelity options as a default. As the Artemis programs implement this process, the expectation is that needs and opportunities for testing are identified in time to be value-added and a better understanding of the residual risk posture is achieved for the campaign.

I. Nomenclature

ARC = Ames Research Center
ARGOS = Active Response Gravity Offload System
CAIDA = Customer Avionics Interface Development and Analysis
CCP = Commercial Crew Program
CoFR = Certification of Flight Readiness
COTS = Commercial off the Shelf
DCR = Design Certification Review
DOD = Department of Defense
EGS = Exploration Ground Systems
EHP = Extravehicular and Human Surface Mobility Program
ET = External Tank
EMC = Electromagnetic Compatibility
EVA = Extra Vehicular Activity
FBC = Faster, Better, Cheaper
FRF = Flight Readiness Firing
GDSS = Gateway Docking System Specification
GN&C = Guidance, Navigation, and Control
HITL = Human or Hardware in the Loop
HLS = Human Landing System
HW = Hardware
ICD = Interface Control Document
ICT = Integrated Compatibility Test
IRD = Interface Requirements Document
ISS = International Space Station
KSC = Kennedy Space Center
LOX = Liquid Oxygen
LRU = Line Replaceable Unit
LTV = Lunar Terrain Vehicle
M2M = Moon 2 Mars Program
MCC = Mission Control Center
MCO = Mars Climate Orbiter
MDM = Multiplexer/De-multiplexer
MEIT = Multi-Element Integrated Test

MPIA	=	Mars Program Independent Assessment
MPTA	=	Main Propulsion Test Article
MRT	=	Mission Readiness Test
MSFC	=	Marshall Space Flight Center
NASA	=	National Aeronautics and Space Administration
NIAT	=	NASA Integrated Action Team
NBL	=	Neutral Buoyancy Laboratory
NRHO	=	Near Rectilinear Halo Orbit
OFT	=	Orbital Flight Test
ORU	=	Orbital Replacement Unit
PCS	=	Portable Computer System
RF	=	Radio Frequency
RPODU	=	Rendezvous, Proximity Operations, Docking and Undocking
RSRM	=	Reusable Solid Rocket Motor
RT	=	Remote Terminal
RWS	=	Robotic Workstation
SARJ	=	Solar Array Rotation Joint
SE&I	=	Systems Engineering and Integration
SIA	=	Shuttle Independent Assessment
SIAT	=	Shuttle Independent Assessment Team
SIL	=	System Integration Laboratory
SLS	=	Space Launch System
SMC	=	Space and Missiles Center
SRM	=	Solid Rocket Motor
SSME	=	Space Shuttle Main Engine
STS	=	Space Transportation System
SW	=	Software
TLYF	=	Test Like You Fly
TOR	=	Technical Operating Report
TWYF	=	Test What You Fly

II. Introduction

A team of National Aeronautics and Space Agency (NASA) and contractor experts spanning the Artemis programs was first assembled in November 2023 to examine possibilities for taking advantage of the opportunity to utilize the Artemis campaign flight systems during integrated test operations prior to flight. Taking advantage of the hardware before it is deployed gives the teams responsible for mission processing the ability to correct problems on the ground where time, access, and resources are more plentiful compared to the situation once the systems are launched; crew time and access for repairs during a mission become more complicated by orders of magnitude. Test operations during the preflight era permits the opportunity to characterize the flight hardware and validate emulators used once the flight systems are deployed. So, in the lingo of a salesman, it's less of a problem asking to be solved than an opportunity that can't be missed. Traditionally, building up spaceflight hardware on the ground allows you the opportunity to physically connect the vehicle structure, the black boxes, the avionics harnesses, and the mechanisms. It allows interface and operability tests to give the system the best chance to operate as expected in flight. A series of system, qualification, environmental, and verification tests build confidence for systems integration into a monolithic flight hardware assembly. Final assembly usually occurs just prior to the shipment to the launch site or may be completed there. For "single stick" launch vehicles with a payload on top that gets integrated, launched, and then deployed in one singular mission, there are past practices and applicable standards [1, 2], but those precedents fall short of the needs for the human space flight paradigm for the International Space Station (ISS) and Artemis, where the payload elements will interface for the first time post-launch in the space environment to become operational.

This paper is intended to provide introductory information for early career professionals or professionals that do not have years of experience testing flight systems. Examples, source material, and overall range of sections within the paper are intended to provide additional information associated with ground testing of space systems as it relates to testing like you fly. Additionally, this paper will describe the process that is being developed to evaluate TLYF for the Artemis missions. This methodical process, for use within and across programs, provides for better understanding of the risks of TLYF exceptions, earlier and more impactful mitigation strategies, and clearer documentation of residual risk.

III. Case Studies of Previous TLYF Outcomes

NASA has been "testing like you fly" throughout many of the human spaceflight programs, for design certification as well as operational and mission readiness scenarios. A variety of challenges drove these pre-flight ground tests, such as new technologies, new types of interfaces, and previous successes or failures. While NASA did not develop a consolidated TLYF lessons learned product, there are papers and mishap reports referenced herein that discuss using TLYF to increase the likelihood of mission success and crew safety. This section provides several detailed examples of NASA historical TLYF activities and the outcomes associated with those specific approaches.

1. *Space Shuttle Orbiter Flight Readiness Firings*

Test Like You Fly testing is important for flight hardware and software/firmware that will fly in support of a mission. This testing is sometimes called "acceptance" or "build verification." One very important test that was performed at least once on each newly built Orbiter was the Flight Readiness Firing (FRF). Flight Readiness Firings (FRFs) were performed on seven occasions during the shuttle program's 30-year history and sought to validate all the vehicle's systems under the closest possible conditions to actual flight, without actually leaving Earth. "On every FRF that we conducted, we learned something new about the vehicle, which made our process and flight hardware better," said Jorge Rivera, deputy chief engineer for Shuttle Processing. [3] As noted initially, cost and schedule implications are often drivers when planning—or not planning—large-scale integrated system tests. However, history has shown us that these tests are very useful in finding off-nominal system level interactions which would not have been discovered, and been able to be mitigated, in lower-level tests.

2. *Space Shuttle Main Engine Design Certification Testing*

The Space Shuttle Main Engine (SSME) Project used ground testing to support both certification and acceptance testing. SSME testing in support of design certification included testing of prototype engine and several development engines, including 150 engine firings with a cumulative run time of 3,500 seconds completed. The tests uncovered significant technical issues, especially with the engine's high-pressure fuel turbopump. Testing continued with development engine firings evolving to full 100 percent rated power for a full flight duration of 520 seconds. By the end of the development testing phase, the SSME test program had accumulated more than 34,000 seconds in 394 firings. [4]

Single engine tests of SSMEs continued through the life of the Space Shuttle Program in support of build verification as well as block upgrades associated with six unique configurations. This ongoing campaign continually enhanced reliability and safety of the SSMEs and reduced overall mission and crew safety risk. The block upgrades were driven by ground test failures, in-flight anomalies, and safety assessments which identified critical failure modes. Until the final SSME test in 2009, 2,307 test firings were completed, representing 228 hours of hot fire time. This campaign represented a significant investment of time and resources, but the outcome was a realized reliability of 1 for the SSMEs during Space Shuttle missions. [4]

3. Main Propulsion Test Article

While single-engine tests remained critical in the SSME testing and certification program, multi-engine testing prior to the first shuttle flight provided a more “like you fly” scenario of the entire propulsion system would behavior during an actual launch. The Main Propulsion Test Article (MPTA) -098 included a flight-weight shuttle aft fuselage accommodating three SSMEs, a simulated orbiter midbody, and an attached flight-like External Tank (ET) to deliver liquid hydrogen and liquid oxygen to the engines. [4]

MPTA testing prior to STS-1 identified at least 4 significant failures. The resolution included redesigning several catastrophic failure modes out of the system design, enhanced inspection of brazed joints associated with the nozzle, increased control/testing of rod weld material, and enhanced inspection of liquid oxygen post concentricity. [5] These flaws, if not identified during a very flight-like test scenario, could have resulted in loss of the initial Shuttle mission, vehicle, and/or crew. In total, in preparation for STS-1, engineers tested 21 engines, including the three that flew on Columbia’s first flight, over the course of 575 single-engine tests and the 18 MPTA tests, accruing more than 87,000 seconds of engine run time. As noted in examples above, these efforts contributed to a very successful mission life for SSME. [4]

4. Shuttle Flight STS-93 Anomalies

During STS-93, there were two serious in-flight anomalies, resulting in NASA leadership chartering the Shuttle Independent Assessment Team (SIAT). The SIAT's charter was to bring to Shuttle maintenance and operations processes a perspective from the best practices of the external aviation community, and report to the Associate Administrator (Office of Space Flight) [6]. Issue #5 in the report was titled “The SSP should adhere to a “fly what you test/test what you fly” methodology.” In the Propulsion section of the report, it states:

“The SIAT considers that a serious lapse in judgment and/or in attention to the engine data base occurred, which allowed two pins to be used in STS-93, without ground test verification firing.” Columbia's right engine had two of these pins jammed into oxygen-injection tubes as plugs to keep liquid oxygen from entering the damaged lines. Handling of the pin insertion and test as a standard repair, which did not necessitate a hot fire retest, precluded management visibility of the frequency of liquid oxygen (LOX) post deactivation. Of 19 pins ejected during ground testing, all but one were ejected during the first engine firing. This process of repair without retest was determined to have significantly contributed to the subsequent pin ejection and the nozzle damage during STS-93 flight. Fortunately, this deviation from TLYF the STS-93 only resulted in an abort to orbit versus a more serious condition [6].

IV. Importance of Proposed TLYF Process

The ISS program went through this same decision-making process during ISS assembly planning and resulted in the program deciding to perform the Multi-Element Integrated Test (MEIT) campaign. The early planning for the ISS elements encompassed a “ship and shoot” approach with no significant test activities at the launch site other than post shipment health check before flight. As the complexity of the interfaces and operations became more obvious, the program direction changed to include more preflight testing at the launch site that included MEIT. That MEIT effort also strove to mitigate risks for elements coming together for the first time in flight by planning to bring the elements together on the ground in test configurations. Those test configurations represented how the ISS elements would come together in low Earth orbit. This campaign was approached as planned work comparable to what was done in previous human spaceflight programs; the Shuttle program with Shuttle payloads was the closest paradigm. While the ISS MEIT work was considered planned work rather than a reaction to risk, by design it was addressing risk mitigation through planned work.

For the Artemis I mission, Exploration Ground Systems (EGS), Space Launch System (SLS), and Orion programs, each were able to follow those practices and employ high-fidelity test facilities in preflight testing. These included the SLS System Integration Laboratory (SIL) at MSFC, the Integrated Test Laboratory (ITL) in Littleton for Orion, and the Customer Avionics Interface Development and Analysis (CAIDA) Lab at Kennedy Space Center (KSC) for EGS. Testing using these high-fidelity assets is not optimal as a distributed system across great distances suffers from the nature of the interfaces and line loss, RF attenuation, latency. To augment TLYF, a variety of other portable, lower fidelity emulators were used in concert with those high-fidelity facilities and were exchanged for use between programs. Despite those efforts, there were many issues only identified when the actual flight and ground systems from the different programs were assembled at KSC for Artemis I. There are countless examples of problems historically found during ground testing, despite efforts to mitigate risks earlier in the flow. Unfortunately, there are also many notable examples of problems found in flight that could have been identified on the ground first but were missed for one reason or another. With elements meeting for the first time in space starting on Artemis III, there is no opportunity to find problems at the launch site, as in the shuttle paradigm or as in the EGS/SLS/Orion paradigm. Elements of the mission come together in lunar orbit unless there is a plan for unique tests to assemble, or at least virtually assemble, the elements on the ground in a preflight test configuration. This configuration must closely resemble what will be performed in flight.

In order to perform similar types of integrated tests with Artemis flight hardware, opportunities would likely only occur in the final few months before flight. Any testing of that scale would need to be scrutinized to ensure that only high priority, high value test operations are performed with low risk to the hardware. It would also need to be accomplished within a limited and well-coordinated schedule window. Every proposed test must follow a technical and programmatic review and approval process, designed and intended to ensure the value being added includes mitigating the risk of flight systems meeting for the first time in flight. This effort requires two parallel paths of forward work areas outlined by the Moon2Mars (M2M) enterprise Systems Engineering and Integration (SE&I) team to come together to make this possible. The first area is focused on enabling test opportunities, including establishing a test configuration along with the means and assets required to execute that test configuration. The second area is focused on identifying the high priority test operations to perform for validation of interfaces, operability, and scenarios that address TLYF, Test Like You Operate, and mission readiness perspectives. These integrated tests with flight-ready hardware would be considered validation and would not replace planned development or verification test objectives using labs, emulators, test assets, or flight systems. While a small number of high priority test objectives may drive the need for a specific test configuration, once the program/enterprise has approved the test configuration, it is worthy to consider what additional test objectives can provide additional value to risk buy down even though they may not have driven the initial rationale for the test to be added to the critical path.

The M2M Verification and Validation Plan reflects an approach that embraces the use of flight systems to the extent possible as part of Test Like You Fly guidance. Because all the Artemis programs have somewhat different approaches to TLYF, M2M SE&I team has been working across the programs to understand those approaches and infuse them with the relevant historical experiences and practices. This paper provides an overview of integrating those efforts into an approach that can be applied across the Artemis campaign paradigm. Additionally, documented risks also exist for individual programs as well as multi-program scenarios for cross program interfaces and integrated mission operations that can and should be addressed with test campaign designed to maximize TLYF aspects for such ambitious early Artemis missions. The program integration leads will continue to work with M2M SE&I and the other Artemis programs to ensure integrated test activities address their relevant risks.

Even when the opportunity exists to test flight hardware on the ground, not all the functional interfaces can be tested in the same operational environment as they will encounter during a mission. Not only are the vehicles not in the space environment, but they may also not be physically connected because the systems cannot be connected in earth's gravity, nor may they even be in the same facility. Therefore, there are some test aspects than can be addressed with ground test methods and some that cannot and must be left for analysis. Testing with the same fidelity of systems that are present in-flight configuration is a key aspect of the Test What You Fly (TWYF) principle. Combined with the earlier discussion about using the systems in representative environments and under scenarios that represent the mission's operations activities, testing with the "what" you fly being represented by the fidelity of the test configuration is part of the overall Test Like You Fly approach. It's not always practical to test with the exact same fidelity due to schedule, logistics, environments, 1-g limitations, and more, but testing with the same fidelity present during mission operations is the ideal for Test Like You Fly approach. In practice, the technical team must assess the best fidelity that can be achieved and inform program discussions on how to balance costs and benefits of the risks in deviating from the ideal TLYF conditions. The following section will provide an overview of the proposed process for making those considerations.

V. Overview of the Evaluation and Documentation Process

A. TLYF Process Flow Described

This section contains an overview of the proposed process for assessing, understanding, and defining program posture and action for TLYF gaps. Based on the findings of this assessment, recommendations for updated testing methods or refinements could be made to close on any TLYF gaps. Residual risks would then be documented, thus providing, in a concise and quantitative fashion, the full TLYF pedigree for the test program. Section VIII describes the process in much greater detail.

For each test that is conducted, a TLYF assessment should be performed. This process applies to internal vehicle testing as well as vehicle-to-vehicle and integrated mission configuration testing. All TLYF exceptions follow the same assessment process at different points of the life cycle.

The general steps of the TLYF assessment process flow, illustrated in Figure 1 below, are:

(1) Perform a technical assessment of each planned test, intra-vehicle or vehicle-to-vehicle.
(2) Create a TLYF exception to document each gap between flight configuration and flight operations verses test configuration and test operations.
(3) Identify the potential impacts to crew safety and mission success from the TLYF exception.
(4) Escalate risk to program for evaluation, disposition, and documentation.
(5) Record assessments and exceptions in a program managed TLYF database.

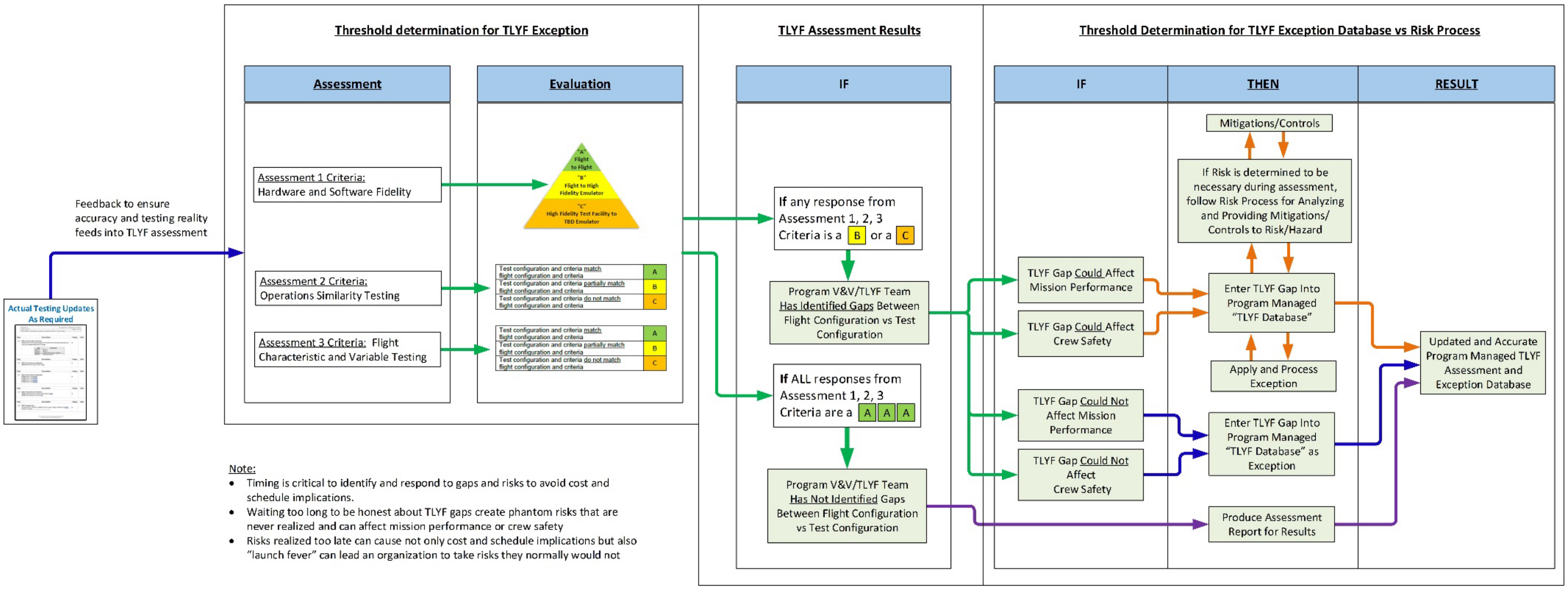


FIG. 1 TLYF EVALUATION PROCESS

The first step of the process is a technical assessment of a given planned test setup. This step defines the three main characteristics of the system under test: hardware and software pedigree, operations similarity, and characteristics and variables of flight systems, for inter-vehicle tests, and interface tests for vehicle-to-vehicle tests. A sample assessment is shown in Tables 1, 2, 3 of Section VIII. A gap assessment is performed, evaluating and scoring the equivalency of the test setup representation of the flight system. A unique feature of this proposed process is the clearly defined rubric for performing this scoring. Representations are characterized as flight to flight, flight to high fidelity emulator, high fidelity emulator to emulator, or commercial off-the-shelf (COTS) to COTS, as depicted by Figure 7/8 in Section V as a first measure of "goodness" of the test. Each gap and associated scoring are documented with a TLYF exception; the terms "gap" and "exception" may be used interchangeably is discussions of this process.

An impact assessment is then performed for each TLYF exception. This technically centered evaluation looks for blind spots to technical insight created by the exception and identifies mitigation and/or control activities to reduce the associated risks. Those recommended activities and a quantification of the remaining risk are documented and brought to the program for assessment, as depicted in Table 4 of Section VIII.

This proposed risk assessment is fed to an existing program risk documentation and evaluation process, where risk managers and other program management perform higher-level evaluation of the controls and accept and/or manage residual risks.

Finally, another unique aspect of the process proposed by this paper, a TLYF database is created. This database, or a purpose-driven superset of an existing program risk database, will document all TLYF exceptions in one searchable and sortable location. This resource allows the full TLYF posture to be clearly captured for milestone reviews and Certification of Flight Readiness (CoFR), giving the program a better understanding of their actual risk posture than by historical documentation methods.

It should be noted that this process should be implemented as tests are being planned, not at the end of a program. If exceptions (TLYF gaps) carry enough residual risk, programs may choose to do more or lower-level testing. Periodic review of exceptions and associated risks should also be conducted, to ensure risk posture is still consistent with program goals and that opportunities for remediation, if needed, are not missed. At CoFR and Flight Readiness Review (FRR), a complete summary of the TLYF exceptions and residual risk will inform the flight readiness assessment for the element or integrated mission configuration.

The process leans into use of existing program test planning and risk management processes. The purposeful, methodical implementation is the unique aspect versus historical implementation. This early and recurring evaluation will provide the program greater opportunities for risk mitigation and better understanding of residual risk.

B. Criteria of the Assessment and Evaluation Process

The three major characteristics of a test to be evaluated are test setup fidelity, operational scenarios represented, and environments to be applied. As noted above, further details are contained in Section V.

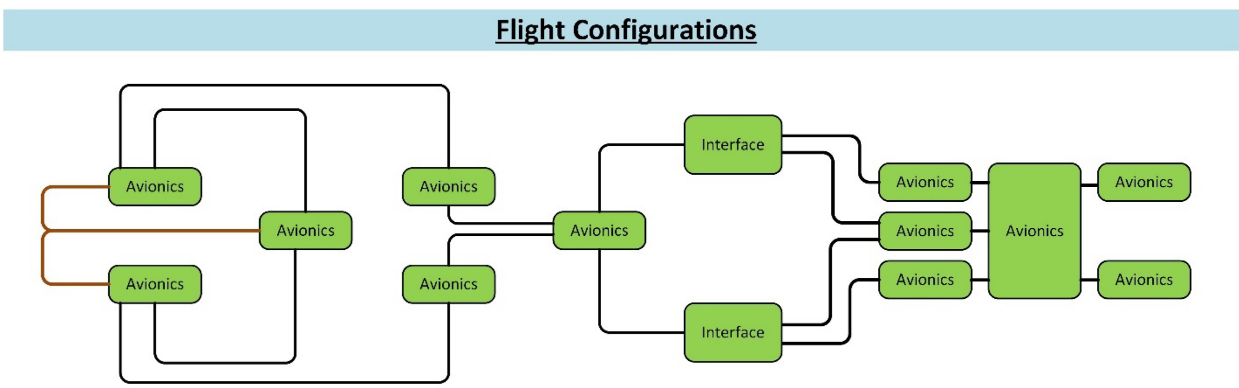
1. Test Setup Fidelity

As test objectives are identified and refined and resulting test configurations derived, it is then important to identify not only the hardware to be used for test, but the specific set of flight and ground software products required, and the necessary functions of each, that are needed to accomplish the goals of the test. During the test planning phase, it is essential that build planning for each necessary software product lash up with overall integrated test schedules to ensure that software versions with the minimum required set of software capabilities of sufficient maturity are available to support specific test milestones.

The Artemis Programs and the providers/contractors performing the design, development, testing, and evaluation of the flight vehicles are using multiple ground assets to support testing throughout the life cycle. In the early stages, testing utilizes lower fidelity COTS emulators. During formal verification and validation of the design to support Design Certification Reviews (DCRs), higher fidelity emulators and test facilities are used. Throughout NASA's history, these ground test capabilities have been used to identify failures early, especially with flight software, so the number of problems identified during flight hardware/software integrated testing at the launch site is more manageable. An evaluation of how "flight equivalent" ground facilities and system emulators are the basis of this step; an example is illustrated by Figure 2.

Another unique feature of flight and ground software is that it is not uncommon (and in fact is usually expected) that it will undergo repeated modifications and updates up until shortly before, and often after, the mission begins. It is therefore important that all software changes incorporated after the conclusion of the final phase of integrated multi-element validation testing be independently assessed for possible impacts to previously captured test results. If any of these pose an unacceptable level of risk that cannot be otherwise mitigated, then regression testing involving the re-execution of one or more test cases may be warranted. The process of TLYF evaluation is ongoing, evolving through test planning and execution and on through any subsequent rework, repair, and/or retest activities.

As discussed above, significant findings from this assessment that could have the potential to negatively impact the accuracy of test results should be documented and dispositioned as part of the TLYF exception process, which will include characterization of any residual risk to the accuracy of test results that might be introduced by a reduction in flight fidelity.

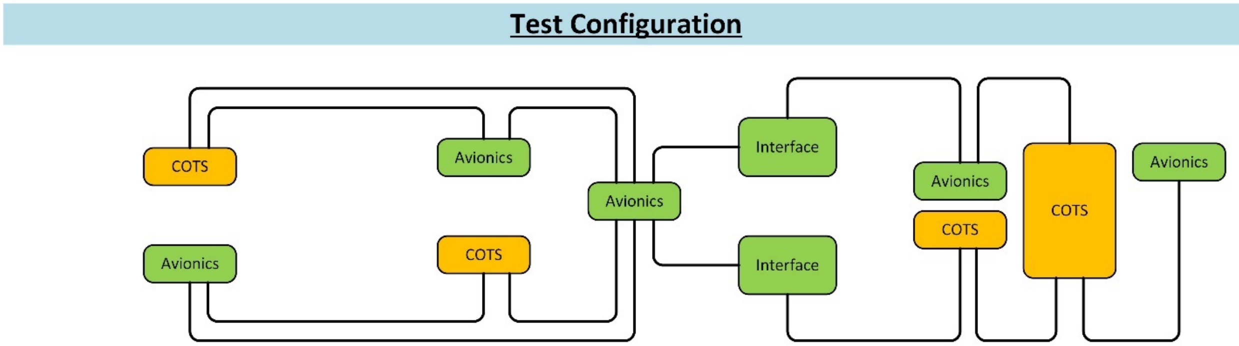


Green = Flight
Orange = Not Flight

Vehicle 1	
Hardware	Flight avionics
Software	Flight software
Cabling	Flight configuration cables
RF	Flight configuration antennas and links
Configuration	Complete flight configuration

Vehicle 2	
Hardware	Flight avionics
Software	Flight software
Cabling	Flight configuration cables
RF	Flight configuration antennas and links
Configuration	Complete flight configuration

FIG. 2 FLIGHT CONFIGURATION ASSESSMENT



Green = Flight

Orange = Not Flight

Test Vehicle 1	
Hardware	Avionics LRUs are flight
Software	Flight software
Cabling	Mixture of flight like and COTS
RF	Flight RF avionics, antennas removed from test
Configuration	Similar flight configuration with test adapters

Test Vehicle 2	
Hardware	Some flight hardware, Backbone hubs not present
Software	Some flight software, no GNC
Cabling	Cables and connections not flight like
RF	Antennas removed from test
Configuration	Not the same configuration as flight

FIG. 3   FLIGHT CONFIGURATION TO TEST CONFIGURATION ASSESSMENT

2. Operational Scenarios

The intent of a comprehensive ground test program is to reduce the likelihood of mission impact and crew safety related failures during the crewed mission. In 2019, NASA’s Commercial Crew Program (CCP) experienced a “high visibility close call,” the inability to dock to the International Space Station, during the first Operational Flight Test (OFT) OFT-1 mission. After the flight, Boeing "acknowledged that the company did not run integrated, end-to-end tests for the whole mission. For example, instead of conducting a software test that encompassed the roughly 48-hour period from launch through docking to the station, Boeing broke the test into chunks [7].” These tests can occur at the component and assembly, single vehicle level, and/or vehicle-to-vehicle level. Performing operational mission scenarios with the actual flight hardware and software and flight procedures helps to gain confidence across the entire system including the people, the paper, the vehicles, the ground and space-based communication assets, etc.

The term “mission readiness test” has been used in both NASA and non-NASA literature with various specific definitions, but the “gold standard” mission readiness test would be one that occurs just prior to the launch of one or more of the vehicles associated with a mission, using flight hardware and software, exercising all planned (and a select set of off-nominal) ops scenarios, and utilizing flight crew procedures. However, as with the aforementioned challenges of testing with the highest fidelity hardware and software, including all ops scenarios in testing can also be difficult and costly to accomplish. Down-selection of the highest value operational scenarios is often required but presents its own set of challenges, since there are many factors in play. The operations team wants to validate their flight procedures and timelines with actual flight crew while also learning more about the system interactions. The technical hardware teams want to confirm complex interactions that may not have been tested previously. The flight software teams may want to confirm changes as a result of non-conformance rework did not adversely impact other parts of the code. Additionally, learning something unexpected about the system, while not a driver for doing these types of tests, is sometimes a valuable unintended consequence, the value of which should be considered in objective trades.

One example of operational scenarios testing is final ground testing of the International Space Station's US Laboratory Module Integrated Compatibility Tests (ICT) which used operationally realistic activities to confirm integrated system performance. During ICT, several payload operations were performed in parallel within certification limits, identifying incompatibilities between planned operations and system performance. This example of strong TLYF informed payload operational plans changes prior to on-orbit utilization, avoiding anomalies and/or impacts to mission success on-orbit. The ICT also performed "day in the life" testing on the more critical systems of the US Lab including but not limited to file transfers to and from ground, Robotic Workstation (RWS) operations, use of multiple Portable Computer Systems (PCS, laptops used by the crew), while the entire vehicle was powered up and transmitting data similarly to on-orbit operations, focusing on maximum loading of the flight computer. During test execution, the primary flight computers, Tier 1 Command and Control Multiplexer Demultiplexers (MDMs), crashed as a result of processor task overrun errors; the root cause was found to be an inaccurate simulation of the input/output of the remote terminal (RT) 1553 bus traffic in the ground flight software lab. As a result of this TLYF finding, the flight software was reworked and retested prior to the launch of the US Lab Module on STS-98.

For Artemis III, there are 2 critical mission phases – lunar orbit and lunar surface. During the Near Rectilinear Halo Orbit (NRHO) phase, the primary operation is rendezvous, proximity operations, docking, and undocking (RPODU); docked operations scenarios also happen during this phase. During the lunar surface phase, the primary operation is Extravehicular Activities (EVA), aka "space walks." Between EVAs, there are turnaround ops to prepare for the next EVA. During the lunar phase of the Artemis III mission, two crew will stay on Orion, and two crew will descend to the lunar surface to perform multiple EVAs. Each EVA will include both crew members in space suits. In addition to nominal operations, there are also "contingency ops" or "off-nominal events" that may occur. An important part of a comprehensive ground test program is testing both nominal and contingency ops. Following are discussions of the Artemis III planned nominal and considered off-nominal operations.

a) Nominal Operations:

Radio frequency (RF) communications between the lander, spacecraft and Mission Control Centers (MCCs) can be organized into two main modes of operation, close proximity, which allows the lander and spacecraft to be in direct connection with each other and out of range, which requires each vehicle to communicate through ground stations to receive data from each vehicle. The definition of these modes of operation is critical to the TLYF process as it provides a fundamental mission driven basis of configuration requirements from which any gaps between flight configuration and testing configurations can be identified and mitigated as necessary.

Within the big picture of nominal orbital and surface operations, there are many smaller operations which must sequence together correctly to allow the larger operations to be successful. Some examples are the transitions between RF and hardline communication interfaces (docking to docked and vice versa) and hardline communication during docked operations which would not require a RF interface.

In addition to orbital and rendezvous operations, the definition of nominal surface operations is critical to the TLYF process as it provides the surface mission configuration requirements which can drive non-surface requirements and testing. There are a multitude of surface assets that will be necessary to work together, each of them presenting opportunities for gaps between flight configuration and testing configurations. Some of these operations will include the following:

- EVA preparation activities utilizing umbilical between suits and lander
- EVA from umbilical disconnect and transit from airlock
- EVA on lunar surface
- Deployment and activation of surface assets and experiments
- Reentry from lunar surface to airlock
- Reconnection of suit to lander umbilical connections
- Post-EVA activities

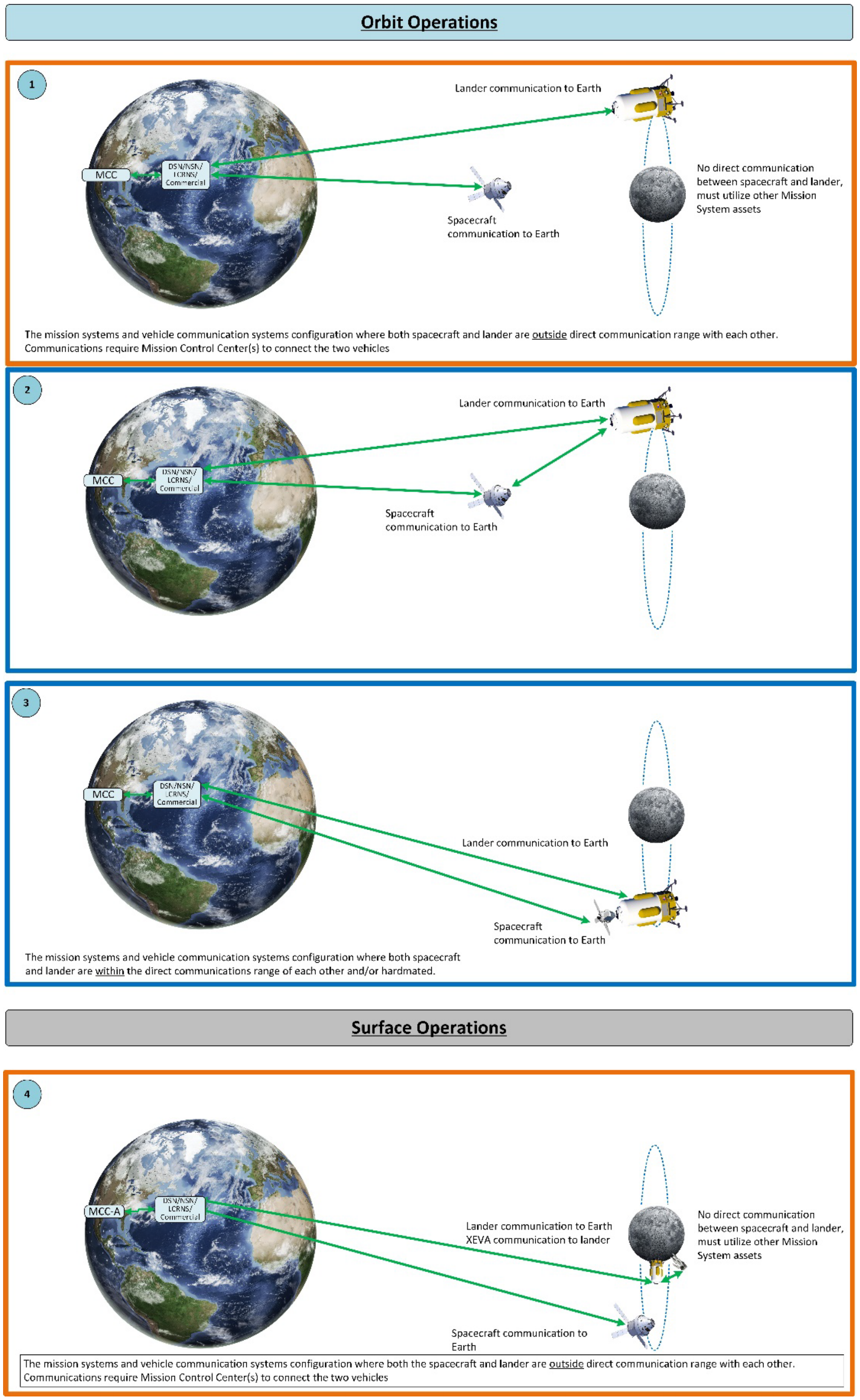


Fig. 4 Communication Concept of Operations

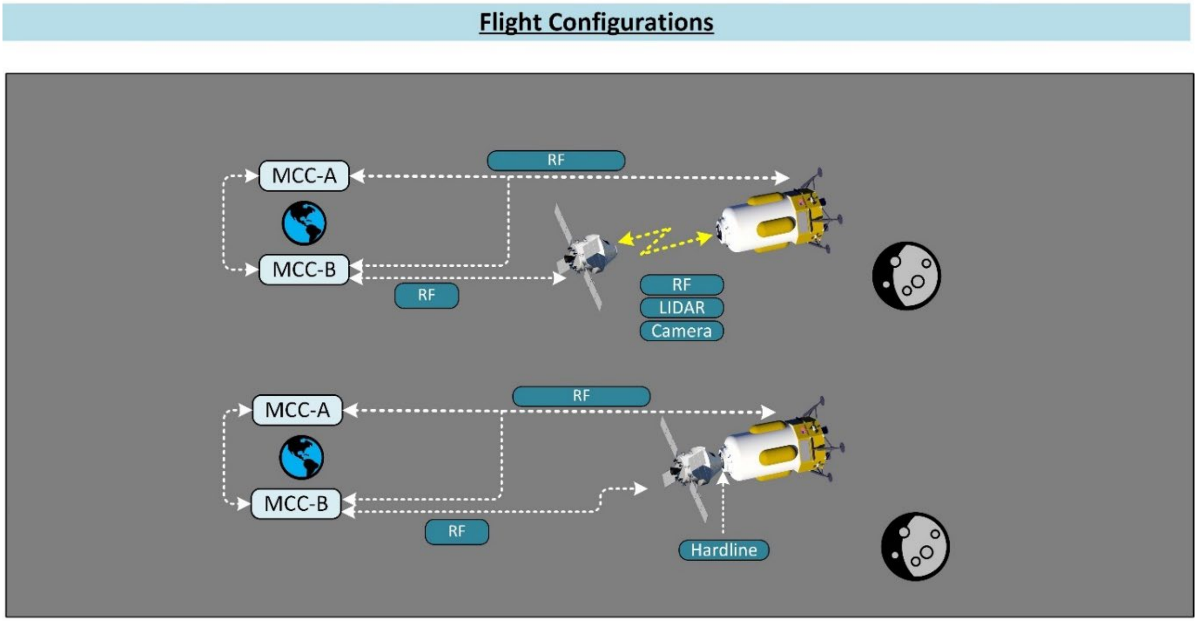


Fig. 5 Radio Frequency (RF) and Hardline Interfaces

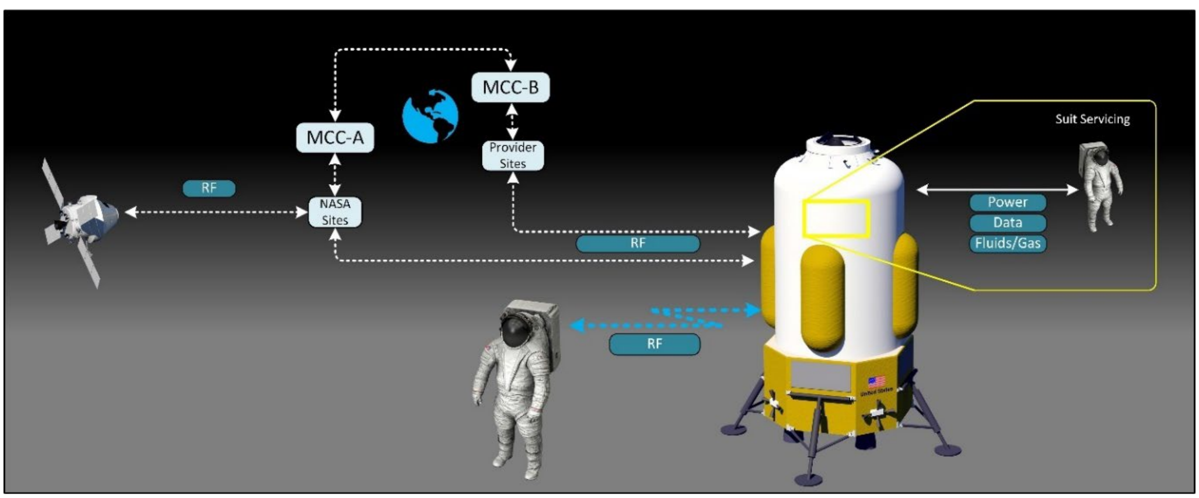


FIG. 6 LUNAR SURFACE INTERFACES

b. Contingencies and Off Nominal Operations:

As part of overall mission planning, contingencies and off-nominal activities must be planned for, utilizing credible failure scenarios. Each of these contingencies and off-nominal events will drive configuration of the vehicles and heavily influence designs. Considerations must be made for fire, rapid depress, slow leak, and toxic atmosphere, Additional scenarios involve incapacitated crew rescue while crew member is EVA, emergency ingress where an even causes EVA crew members to head inside immediately, and decompression sickness treatment. Contingencies and off-nominal events such as these are all part of the vehicle designs and, as such, require dedicated testing to establish flight readiness. Testing involving any of these contingencies and off-nominal events provides valuable opportunity for TLYF assessments to ensure vehicle designs, operations, procedures, and configurations allow for crew safety and mission success.

3. *Environmental Aspects*

a. *Introduction*

In the development of space flight hardware, the effects of many environmental conditions must be considered for every phase of operation. The term "environments" generally is used to encompass both natural, such as humidity, and induced, such as vibratory, environments. The guiding premise of assessing environments is that nothing shall experience an environment for the first time during flight. Additionally, hardware responses to some environments, such as shock, are not well-predicted; therefore, environments are usually best assessed by test rather than analysis. Any differences between how hardware is environmentally tested versus how it experiences these environments during mission ops phases must generate a test like you fly exception for evaluation.

NASA generally follows guidance on test levels, durations, combinations, and order of test from SMC-S-016, Air Force Space Command Space and Missile Systems Center Standard Test Requirements for Launch, Upper-Stage, and Space Vehicles [8] and MIL-STD-810, Department of Defense Test Method Standard for Environmental Engineering Considerations and Laboratory Tests [9]. However, because there is no NASA standard for environmental test or test like you fly, every program sets its own requirements and expectations, and the risks associated with test like you fly gaps are often documented and accepted at the end of a program when there is little time or budget remaining to really address them. As stated in the introduction this paper provides an overview of integrating those efforts into an approach that can be applied across the Artemis campaign paradigm asserts that a rigorous process can ensure a methodical assessment earlier in a program when more options exist for buying down risk.

b. *Test Environments and Durations*

It may seem inherently obvious that the closer the test environment can match the flight environment, the better the assessment of the hardware. However, rarely is "use the correct environment" as simple as it sounds. First, the flight environment must be known, and the only way that can be exactly known is by highly instrumented flight in the exact design configuration and mission profile; rarely do we have that luxury. If analytical techniques are used to extrapolate from other flights, other designs, other locations, etc., assumptions and model limitations are now added to the equation. Enveloping of environments for duration or margin considerations adds another set of analytical uncertainties. Once the flight environments are adequately defined, NASA often uses or tailors SMC-S-016 to define the test environments. SMC-S-016 defines qualification and acceptance test levels relative to maximum predicted environments.

Qualification tests conducted to demonstrate satisfaction of design requirements including margin and product robustness for designs that have no demonstrated history. A full qualification validates the planned acceptance program, in-process stress screens, and retest environmental stresses resulting from failure and rework. The qualification test level for an environment shall provide a specified margin to the acceptance level. Qualification test durations or repetitions demonstrate life remaining for flight after a maximum time or repetitions of acceptance testing at all levels of assembly in support of rework or retest of flight hardware. Qualification testing should not cause unrealistic modes of failure.

To demonstrate workmanship, the acceptance environmental conditions shall stress the hardware to the maximum conditions expected for all flight events, including transportation and handling, but not less than the minimum workmanship levels defined. Acceptance tests are intended to satisfy these goals by subjecting the unit to the maximum environmental exposures expected in service or the environmental stress screen level whichever is more severe. Any gaps in the ability to confidently predict the flight environments and/or the transformation of those to test environments should be addressed in the gap assessment process (outlined later in this paper).

In addition to deriving the correct environments, applying them in a flight-like manner presents another set of challenges. As mentioned above, the test should not induce unrealistic modes of failure. It must also be considered that the constraints needed to apply environments be carefully designed to not change how the environment is imparted to the hardware under test. What else can we say here? Again, any gaps in the ability to load the hardware in a flight-like manner should be documented and assessed.

Although testing to the correct environment seems straightforward, there can sometimes be either technical or programmatic reasons that is not accomplished. NASA's most well-known example of a failure related to a lack of test like you fly with respect to environments is the Space Shuttle Challenger Accident. While that accident is an indelible part of NASA's history, it is also important to acknowledge the testing that occurred prior to the accident, as a direct result of the accident, and for the remainder of the Space Shuttle Program.

A total of seven successful Solid Rocket Motor (SRM) tests, four demonstration and three qualifications, were completed prior to the first Shuttle flight in April 1981 [10]. The environments applied during that body of tests, combined with the lowest launch temperature to date (53 Deg F), caused some concerns with the adequacy of the test like you fly conditions, although not expressed in those terms, on the morning of January 28, 1986. That morning, temperatures were measured at 25 Deg F on the left SRB and 8 Deg F on the right. Although cultural factors directly affected the launch decision that day, the fact that the flight environment was not enveloped by test conditions remains the heart of the issue.

Chapter IV of the Rogers Commission report, "The Presidential Commission on the Challenger Accident, an Accident Rooted in History," concluded that, in addition to a different physical configuration (tested horizontal rather than vertical), tests did not duplicate the full range of operating conditions, such as the record-low temperatures that were experienced on launch day. The Commission recommended that "The certification of the new design should include: Tests which duplicate the actual flight configuration as closely as possible and Tests over the full range of operating conditions, including temperature."

As discussed in earlier in this paper, testing like you fly can also support design certification/recertification, which is a critical element of flight readiness for space systems. During the period between the Challenger Accident and return to flight in 1988, there were a total of six static fire tests performed to support recertification of the SRM design. Static testing continued through the remainder of the Space Shuttle Program.

Correctly quantifying the amount of time, a system is exposed to an environment during mission ops and for the life of the system is as important as using the correct environment. Essentially the environment and duration combine to represent life taken from the hardware as a result of that exposure. Most flight environments are short-lived, as space flight is composed of many different short regimes. However, the in-space environment will normally be encountered for a much longer duration. True "life" testing for environments like in-space thermal, for example, is very time-consuming and is often targeted for reductions such as acceleration or assessment by analysis only. Because of this, as noted previously, assessing test like you fly early in a program can help identify areas of high-risk relative to certain environments and allow time for mitigation by more representative testing.

In late 2007, the large bearings on the main truss of the photovoltaic solar panels on the International Space Station (ISS) photovoltaic solar panels became so hard to turn that they had to be shut down after exceeding drive current safety limits. Design of those bearings, known as the Solar Array Alpha Rotary Joint (SARJ), presented challenges of selecting lubricants and materials because bearings, gears, and mechanisms would operate in the harsh low-Earth-orbit (LEO) space environment for a long life. Additionally, materials would have to allow no chance of contamination of other sensitive ISS systems. During that design process, some research suggested that greases based upon fluorocarbons may be viable inside sealed bearings, and noble soft metals such as thin gold films might be suitable for contacting surfaces exposed to the space environment. These suggestions were adopted in the SARJ design, but during the initial phase of the ISS program, sufficient time was not available nor were there sufficient financial resources for life (30 year) testing. Further, no widely accepted accelerated life test existed then, or even today, to confirm a selected lubrication approach. Once the root cause was determined to be a design vulnerability coupled with lubrication issues and was corroborated through ground tests, the system was repaired on-orbit by replacing worn and damaged components and applying space-compatible grease. This example illustrates not just the importance of assessing to the correct environment, but also the importance of test at the correct level of assembly.

c. Level of Assembly

Not only must the environments experienced be considered during every operational phase, but it must be determined at what level of assembly the environments should be applied. MIL-STD-1540, Department of Defense Standard Practice, Product Verification Requirements for Launch, Upper Stage, and Space Vehicles, [11] superseded by SMC-S-016 in 2016, advised that "in general, the design and manufacturing requirements should be verified at the lowest level of assembly practicable. Requirements affected by integration into higher levels of assembly, such as external system interfaces, should be verified at the highest practical level of assembly." Hardware complexity and ability to apply environments correctly at that level of assembly are some primary drivers.

Predominantly mechanical components or subsystems, such as tubing or wiring, are often tested at higher levels of assembly, where electrical subsystems are often tested at the system level. Conversely, cost (both dollars and schedule) of repair and retest for problems found at higher levels of assembly can drive test to be conducted at lower levels. SMC-S-016 suggests level of assembly for each type of environmental test.

Determining the correct level of assembly for any given test is not straightforward and is generally guided by past experience. It is also important to acknowledge results and non-conformances from higher-level tests to inform whether more testing should be conducted at a lower level to fully understand results or to more realistically apply environments to the system. "Listen to the hardware" is an old adage derived from experience, and it is applicable in this situation. The on-orbit anomaly with the ISS SARJ, mentioned earlier, is a mission ops-realized example of this principle. The TLYF process proposed by this paper addresses lower level of assembly test and how exceptions generated during those tests feed into the overall assessment process as described later in this paper.

d. Order of Application

Ideally, environments would be applied in the order in which they are experienced during mission ops, with functional tests conducted in between different environmental tests. Pretest analysis can also be used to identify the locations of minimum design margins and associated failure modes to determine the priority/criticality and sequence of loading conditions.

Although functional checks in between tests give a good indication that no damage was incurred, there could be small flaws which do not affect functionality at that point but grow during the next environmental cycle. However, disassembly for inspection breaks configuration and risks man-made damage, so is usually avoided; therefore, it is never completely known which environment caused the flaws. The effects of this "catch 22" situation are hard to quantify, and, in reality, only pose potential for increased troubleshooting time; they do not increase the overall risk of not appropriately driving out issues.

MIL-STD-810 acknowledges that in most cases, there is no single defined sequence. Generally, the desire to perceive defects as early in the test sequence as possible and cost of test and/or potential rework are more likely to drive any deviation in test sequence versus operational order. This facet of environmental test is usually the easiest to achieve, so it is often followed.

e. Combination of Environments

Although MIL-HDBK 340A, Department of Defense Handbook, Test Requirements for Launch, Upper-Stage, and Space Vehicles, Volume 1: Baselines was cancelled in 2017, this description captures the historical and current military and NASA approach to order and combination of environmental tests:

"Environmental tests as defined by existing standards are intended to be imposed sequentially, rather than in combination. Nevertheless, features of the hardware design or of the service environments may warrant the imposition of combined environments in some tests. Examples include combined temperature, acceleration, and vibration when testing units employing elastomeric isolators in their design; and combined shock, vibration, and pressure when testing pressurized components. In formulating the test requirements in these situations, a logical combination of environmental factors should be imposed to enhance test perceptiveness and effectiveness."

Combinations of environments are generally harder to apply by test; facilities for combined environments, especially for large systems, are not plentiful. Thermal/temperature and vacuum are often applied simultaneously, but other combinations are generally evaluated by analysis instead. Analytical techniques rooted in 60 years of spaceflight with different vehicles and systems are often chosen over combined tests, due to cost and the aforementioned logistics. However, some environments, like shock, remain difficult to fully describe with analysis and are therefore better to be addressed by test. Fortuitously, applying environments in combination is more stressing, and therefore most effective, at the component level.

A current example of the difficulty of combining environments is the simulation of Artemis lunar surface ops. Crew operations in the lunar gravity environment are currently being simulated in the Active Response Gravity Offload System (ARGOS) gravity-offload system and the Neutral Buoyancy Lab (NBL) pool. Each simulation method has its shortcomings; the ARGOS limits some motions due to the offload mechanism, and the NBL water resistance tends to overdamp motion compared to the lunar environment.

In the aforementioned development of the SSMEs, extensive combined environment testing was conducted at the MPTA. This testing included all three SSMEs in a flight-representative Orbiter aft section and an external tank, enabling simulation of flight pressures, cryogenic effects, and full-duration burns. That campaign remains a strong example of a “test like you fly” philosophy. As new LOX/methane propulsion systems are being developed, particularly by commercial providers, the testing approaches may vary depending on program constraints and objectives. This evolving area of development could benefit from continued attention within the Artemis test like you fly framework to ensure mission-relevant conditions are adequately represented.

Generally, spaceflight systems have encountered few failures due to lack of combined environment testing. This outcome can be attributed to robustness, thorough component-level screening, and low statistical combination over short missions. However, a very recent example of lack of test to combined environments became very public; fortunately, a potentially catastrophic outcome was avoided.

In the Operation Flight Test number 2 (OFT-2) of Boeing’s Starliner spacecraft, a helium leak in the pressure-fed thruster system was ultimately responsible for rendering the vehicle unsafe (per NASA’s decision) to return crew to earth. In Starliner, helium is used to pressurize the propellant tanks, and without the proper pressure, the thrusters failed to produce sufficient thrust. According to Boeing’s official investigation reports and NASA press releases, the leak was traced to a Teflon seal which was expanding under thermal stress, restricting helium flow and triggering system shutdowns. This O-ring was used in another application, but at a much lower pressure, for air not helium, for a very short duration, and in a configuration in which it was destroyed during use, leaving no way of inspection after function. This valve/O-ring configuration was never tested in the combination of environments it encountered during OFT-2 [12].

f. Conclusions

This section has addressed consideration of the effects of environmental conditions for every phase of operation. Environments and durations, level of assembly, order of application, and combination of environments must all be considered in an environmental test campaign. With no standard for environmental test or test like you fly, every program sets its own requirements and expectations, and the risks associated with shortcomings are often accepted at the end of a program when there is little time or budget remaining to really address them. While environmental test is usually costly, retest impact only increases as a system’s readiness for delivery of increases. And ultimately, any test is less costly than a failure realized later. This paper asserts that a rigorous process can ensure a methodical assessment earlier in a program when more options exist for buying down risk.

VI. TLYF Recommendations

The ideal implementation of TLYF principles dictates the use of flight hardware, with flight software and flight operational procedures being tested in the environment the system is intended to operate. To accomplish this best practices effectively, program and project managers must commit, define, approve, and apply this TLYF philosophy as early as possible in the programs or project's life cycle.

This paper addressed consideration of the effects of environmental conditions for every phase of operation and outlines how the durations, level of assembly, order of application, and combination of environments must all be considered in an environmental test campaign. Given the lack of a standard for environmental test for test like you fly, every program owns the responsibility to develop its own requirements, expectations, and risks associated with gaps in environmental testing and must acknowledge shortcomings are often accepted at the end of a program when there is little time or budget remaining to really address them.

Most NASA programs are multiyear efforts greatly influenced by schedule. While programs and projects may initially plan to fully implement a TLYF Best Practices standard, the realities of long-term schedules will influence what can be implemented. Concurrent availability of flight hardware with its associated flight software may not be possible. Concurrent availability of flight hardware with its associated interfaces may also not be possible. Hardware schedule and location disparity affects implementation of TLYF principles. Current paradigms may make it very difficult to co-locate flight hardware from program/project A with its interfacing flight hardware on program/project B. One element may not be available for testing until after the other element has already launched. All this and many other reasons may not allow concurrent testing of flight hardware components.

For cases where flight hardware to flight hardware testing is not possible, the use of high-fidelity emulators is warranted. High fidelity emulator refers to the emulator being manufactured and configured as close as possible to the flight hardware's manufacturing and configuration. Ideally the high-fidelity emulator is a nearly exact replica of the flight hardware. This is necessary to ensure test responses and outcomes are as close as possible to the flight hardware's responses and outcomes.

Programs/projects may decide to conduct testing with emulators that aren't at the highest fidelity possible. While this is not ideal, it may be the only way to facilitate testing of critical interfaces prior to launch. An established TLYF process will assess and document those areas of the test configuration that are not in-flight configuration, allowing the program/project managers to assess if the residual risk due to not testing with flight hardware can be accepted or if further testing with higher fidelity assets is necessary.

Based on lessons learned from past and more recent programs, NASA leadership has placed emphasis on ensuring interfaces that are integrated for the first time during in-flight mission operations are assessed and tested using TLYF principles prior to launch. These preflight ground tests may identify hardware, software, and/or procedure errors or anomalies that must be addressed prior to launch – indeed, repair of an identified hardware anomaly my only be possible before launch. When elements are incapable of being ground tested with each other because of disparate launch schedules, it is imperative that the first element launched be assessed for critical interfaces that must be tested prior to launch so those interfaces will successfully operate in flight when the second element is integrated with the first element. For these cases, testing with high fidelity emulators may be the best and only option to implement TLYF principles.

The programs or project's methodical implementation of TLYF principles and approved processes should lead to a test campaign that yields a lower overall residual risk because the highest fidelity hardware, software, and procedures are used and successful completion of the tests indicates that the flight hardware, software, and procedures should operate as expected on-orbit. Establishing a methodical process to implement TLYF principles allows for detailed assessment of the critical interfaces where flight hardware to flight hardware testing must be implemented or where flight hardware to higher fidelity/lower fidelity emulators may be used. A clearly documented and structured approach allows the program/project to better determine where to apply resources to obtain the best outcome. A methodical TLYF process implemented at the beginning of a program/project should yield lower residual risk for flight because it will allow managers to make informed test decisions throughout the life cycle of the program/project when there is time available to make necessary changes. If the TLYF process is only implemented near the end of the program's/project's life cycle, program managers may have to accept a higher level of residual risk because there is no time or capability to correct errors or implement changes.

The recommended approach for implementing TLYF principles for preflight ground testing of critical interfaces that will integrate for the first time during in-flight mission operations is to ideally conduct tests using flight hardware, flight software, and flight operational procedures. If any flight hardware, flight software, or flight procedure is not available then the highest fidelity equivalent should be used with the approved TLYF processes being implemented methodically to assess and document acceptability of the test configuration. Finally, the use of lower fidelity hardware, software, and operational procedures is possible provided the approved TLYF process is followed to address the cost and benefits of the risks of deviating from ideal test configurations.

VII. Detailed TLYF Evaluation Process

As noted in Section IV, for each test that is conducted, a TLYF assessment should be performed. This process applies to internal vehicle testing as well as vehicle-to-vehicle and integrated mission configuration testing. Part F of this section provides detailed guidance on assessment of the test fidelity and provides a rubric by which to characterize it.

Additionally, a summary of the TLYF assessment process was illustrated in Figure 1 of Section IV. Part G of this section will describe the detailed mechanics of the process: generation of a TLYF exception, technical and programmatic assessment, and characterization and documentation of residual risk.

A. Test Fidelity Assessment

Even when testing flight fidelity systems on the ground, not all functional interfaces can be tested in the same operational environment as they will encounter during a mission. Not only are the vehicles not in the space environment, but they may also not be physically connected because the systems cannot be connected in earth's gravity; in some cases, the vehicles may not even be located in the same facility. Limitations such as these leave some aspects can only be addressed by analysis. Testing with the same fidelity of systems that are present in-flight configuration is a key aspect of the Test What You Fly (TWYF) principle. Combined with the earlier discussion about using the systems in representative environments and under scenarios that represent the mission's operations activities, testing with the "what" you fly being represented by the fidelity of the test configuration is part of the overall Test Like You Fly approach. It's not always practical to test with the exact same fidelity due to schedule, logistics, environments, 1-g limitations, and more, but testing with the same fidelity present during mission operations is the ideal for Test Like You Fly approach. In practice, test conductors must assess the best fidelity they can achieve and balance costs and benefits of the risks in deviating from the ideal TWLF conditions. Additionally, this section will briefly describe the test setup methods that can be deployed to test these vehicle interfaces even in the cases where they cannot be in the same facility much less physically connected.

1. Interfaces

The following section describes general approaches to testing these categories of interfaces. The test setups are described in general terms using best practices commensurate with previous human spaceflight programs.

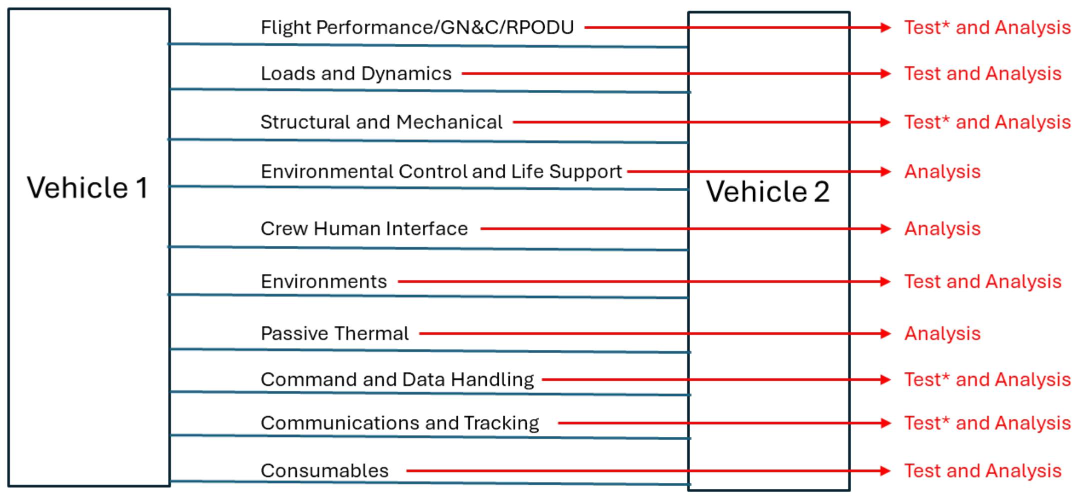


FIG. 7 VEHICLE TO VEHICLE INTERFACE

Figure 4 diagram above is typical of one found in a program-to-program interface requirements document. The black lines represent the functional interfaces, and the red arrows correlate to the methods that are likely to be applied to address verifying those interfaces. For the purposes of performing a validation test using flight vehicles in a test like you fly paradigm, in general, the interfaces labeled Test* can be addressed with the ideal of vehicle-to-vehicle validation testing, though not without some exceptions. Similarly, while some of these interfaces are labeled as analysis, that is a generalization, and there are some test aspects to that analysis approach, but they are unlikely to be addressed with vehicle-to-vehicle testing. For example, two halves of a docking systems may be put in a 6 degree of freedom test rigs to test various loading conditions in that test setup, but the remainder of the verification would be completed by analysis infusing that test setup data into vehicle models used in simulations for mechanical systems behavior.

2. Docking

In general, it is impractical to “dock” fully integrated vehicles to each other in 1-g environments to mate the docking systems, but there are strategies to reduce the risk of those first-time docking system mates in space. Integration activities include mating flight or flight-like docking systems using a docking system test rig, oriented vertically, may be applied. This reduces the effects of Earth’s gravity on the soft capture systems and enables dynamic testing such as 6 degree of freedom vehicle rendezvous and approach testing, as well as physical mate and fit-check testing. However, similar ground tests may also happen horizontally with piece parts of the docking systems with gravity offload mechanisms. In this case, the systems are not installed in final flight configuration and may be engineering or qualification unit systems. But recognize that docking systems in such a standalone test setup may not reflect stresses on the docking system resulting from the installation on the vehicle structure, and the wiring harnesses for docking umbilical connectors are not in a final flight configuration that would give confidence in the wiring compatibility. To remediate these challenges, tolerance measurements and/or the use of a docking system template can be used post-installation to validate that the docking system is within the design guidelines of the Gateway Docking System Specification used for Artemis missions. Also, load, acceleration and strain measurements are taken throughout testing so that they can be correlated to the analysis models of the active and passive docking systems to represent the loads and stresses more accurately on the docking rings and surrounding vehicle structure during docking events. Secondly, for the umbilical’s, pigtail umbilical cables can be connected for test purposes in a configured test setup as a secondary check to the Interface Control Document (ICD) analysis. Typically, once the pigtail cable setup is in place emulators are used to flow data or power as appropriate over the vehicle-to-vehicle interface to validate the wiring compatibility. As was done in the ISS MEIT configurations, this approach can be applied with two flight vehicles next to each other in proximity or with one flight vehicle connected to a configured, high-fidelity emulator. Finally, the team can assess the value of a "Gold Standard Rig" like what has been utilized for commercial partners in the ISS Program. An Artemis Gold Standard would be a Gateway Docking System Specification (GDSS) compliant high fidelity passive docking ring configurable for each variation required of the GDSS configuration.

3. *Radio Frequency (RF)*

When flight vehicles and/or flight emulators are brought in close proximity for system level tests in integration facilities where antenna radiation is not possible or practical, hat couplers can be used to route those data streams to systems where the RF streams can be integrated. The desired ability to use a hat coupler on an RF system can influence the design of the antenna and feed and should be factored into the design process based on test plans. These test points should be identified and coordinated with the hardware providers. RF systems can be connected building to building over RF when in relatively close proximity or may be transmitted via media converters over other means such as fiber optic lines. Analysis of the test setup should be performed to characterize the performance and assess whether any latency or other attributes affected by test setup would impact any of the test requirements.

4. Ethernet and Hardline Data Interfaces

As discussed in the docking system section, for umbilical cables, pigtail cables with flight or flight like connectors can be utilized in a configured test setup as validation check of the ICD analysis. For Ethernet data, such a test flowing data between a combination of vehicle/emulator ensures the buses are correctly routed (buses aren't crisscrossed) and polarity is correct (pins wired correctly). This best practice is recommended for at least the first flight of each new configuration of the GDSS umbilical connector varieties. For any other cables that are to be connected between vehicles internally or externally, similar best practice is recommended using adapter cables between the two vehicles or vehicle to emulator as a validation check of the ICD analysis following through with functional flow of data through that test setup. These pigtail/adapter cables must be designed and procured well ahead of test opportunities as many connectors such as the docking system umbilical connectors are long lead items. The connectors are often long lead items if such pigtail/adapter cables do not yet exist and/or the connectors are not readily in stock with a provider. Analysis of the test setup should be performed to characterize the performance and assess whether any latency or other attributes affected by test setup would impact any of the test requirements. Of critical importance across both the RF and hardline data interfaces are each vehicle's response to the other vehicle's commands and data through all mission use cases for nominal and off nominal scenarios. This evaluation of each vehicle's subsystems behavioral response to the other vehicle's inputs (or lack of inputs) allows for modes of operation to be exercised and validated.

This type of testing opportunity can also exercise the actual flight software on flight hardware which can render different results from that of COTS emulator applications.

5. *Power*

Similar to the discussion for Ethernet and other hardline interfaces, there should be pigtail or adapter cables that connect vehicle to vehicle or vehicle to high fidelity emulator, but an additional step of testing for correct voltage and polarity MUST be performed first, usually through a breakout box setup, before making the connection and flowing power between the end items whether they are flight hardware or ground support power supplies. With 120VDC or 28VDC, significant damage usually results if there is any error in the wiring for power systems. If using emulator systems for one vehicle or another, any sort of power quality assessment will require a high-fidelity power systems emulator representing the impedance and electrical performance of the flight systems.

6. *Fluids*

For Gateway docking configurations utilizing fluids interfaces, servicers can be configured for test purposes as required to serve as an active means of utilizing those interfaces and validating them during ground test operations using additional jumper lines to connect across the test interface. Environmental interfaces such as air flow and pressure control are going to be unlikely candidates for testing with flight vehicles on the ground and are more likely to be candidates for analysis and simulation.

7. Software

Testing of the interoperability between multiple flight elements of any appreciable complexity depends heavily on the use of a tightly coupled set of various of software products. Modern spacecraft architectures are often widely distributed between a collection of specialized platforms dedicated to such functions as guidance and navigation, systems management, mission sequencing, flight crew interface, etc., each of which can contain one or more executable code files and reconfigurable data structures. Additionally, ground-based computer systems are also typically required to support testing of actual mission scenarios. These can include command and control centers, communications network processors, and dynamic/environmental simulations. To ensure the most realistic representation of actual flight configurations and performance of flight elements it is of course necessary to use the version of each of these software products that is as close as possible to that which is expected to be used during the mission. This can present its own set of unique challenges since history has shown that availability of final flight and ground segment software often lags significantly behind the development of their corresponding hardware components. Usually this can be the result of changes in mission or system requirements once design has been finalized, refining or adjusting flight rules and procedures, or fixing issues encountered during systems integration and testing since it is always faster and cheaper to tweak software than to modify hardware devices.

Flight software products can typically be decomposed into a set of lower-level components and functions which are then executed under an almost limitless combination of configurations and conditions both nominal and off-nominal during the course of the mission. It is obviously not feasible to fully exercise the complete set of software capabilities during integrated multi-element validation testing; this effort is performed in (and more suitable to) a software development facility or systems integration laboratory. Because the time available for testing on actual flight hardware is always too brief, any software specific test objectives should be carefully selected to include only the most critical aspects that rely on close coordination between functions residing in platforms on multiple elements or whose proper performance depends upon unique characteristics of hardware devices or components that cannot be easily simulated. The goal is to use the test an opportunity to demonstrate and characterize true end-to-end interoperability of complex functions distributed between flight elements rather than a simple verification of lower-level interfaces. Some examples of these that were exercised in previous programs include automated system/subsystem reconfiguration after commanded change to vehicle state or mode, coordinated changes in overall vehicle attitude or other propulsive maneuver, or cross-element responses to high level faults or emergency conditions (fire, rapid depress, loss-of-comm, etc.).

As test objectives are identified and refined and configurations derived it is then important to identify the specific set of flight and ground software products required and the necessary functions of each that are needed to accomplish the goals of the test. Once these are known then the corresponding software development and test teams can be brought into the test planning effort. Software production life cycles for complex products are traditionally centered around build plans which document the phased incorporation of increasingly more mature features and capabilities into successive versions of each software component. During the test planning phase, it is essential that build planning for each necessary software product lash up with overall integrated test schedules to ensure that software versions with the minimum required set of software capabilities of sufficient maturity are available to support specific test milestones.

As the focus begins to shift to evaluation of readiness to conduct the test it is recommended that a thorough assessment of the status of each required software product be undertaken and the results presented as part of a Test Readiness Review. Items to be considered should include:

- Are all functions and features necessary to fully accomplish the test objectives present?
- How close is flight software to the actual version that will be used for the mission? Are any significant portions missing or stubbed out? Is it native code or does it contain any additional diagnostic features used to evaluate performance or detect bugs that will not be present in the final versions? Was it created using the same compiler and toolchain as the actual flight products?
- Will the software be executed on a flight-like hardware platform rather than a software emulation of a flight processor?
- Has the software completed formal verification and qualification?
- Has a review been completed on all currently documented open work on the software been completed to ensure no adverse impact to performance of the test (problem reports, enhancement requests, etc.)?

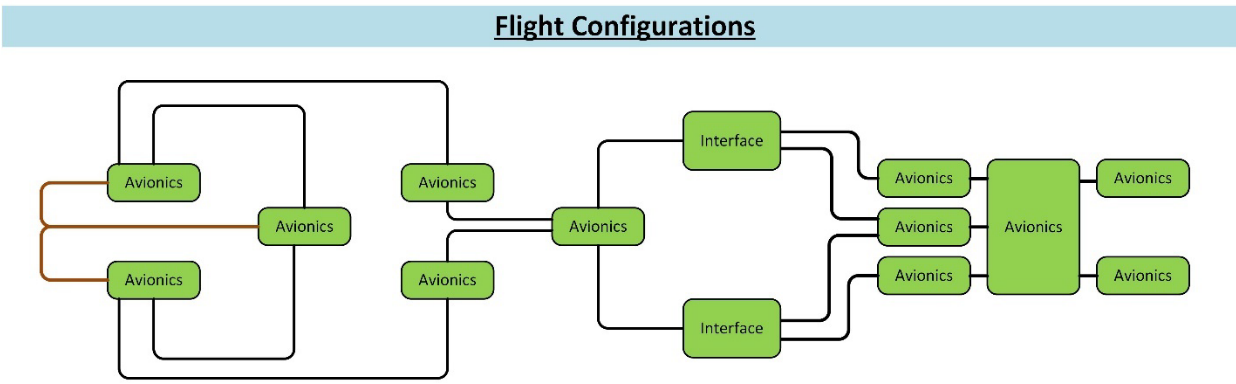
Any significant findings from this assessment that could have the potential to negatively impact the accuracy of test results should be documented and dispositioned as part of the Test-Like-You-Fly exception process, which will include characterization of any residual risk to the accuracy of test results that might be introduced by a reduction in flight fidelity.

Another unique feature of flight and ground software is that it is not uncommon (and in fact is usually expected) that it will undergo repeated modifications and updates up until shortly before (and often after) the mission begins. It is therefore important that all software changes incorporated after the conclusion of the final phase of integrated multi-element validation testing be independently assessed for possible impacts to previously captured test results. If any of these pose an unacceptable level of risk that cannot be otherwise mitigated, then regression testing involving the re-execution of one or more test cases may be warranted.

B. Fidelity

The fidelity of hardware proposed for ground testing is extremely important to evaluate as accepting credit for verification/validation testing results, using hardware that is not an accurate representation of the actual flight interface or in a test setup that is not reflective of the operational configuration, introduces risk. As an example, given in Figure 7, the lander to spacecraft flight configuration is defined along with the tables below outlining the fidelity of the hardware, software, etc. used. Naturally, in this configuration everything must be flight. Defining what the flight configuration then allows for deviations from flight for test scenarios to be outlined and assessed for risk introduction. This is the key concept behind the expression “Test What You Fly.”

Within each flight vehicle a team must define what components define that flight configuration. Shown below is an example of an avionics configuration at the line replaceable unit (LRU) level along with interconnections and interfaces. This next level down within the flight configuration allows detailed comparison to the planned test configuration as shown below the "Flight Configurations" graphic. The key provided for flight verses non-flight demonstrates how a test configuration can deviate from the flight configuration which would be out of alignment with Test Like You Fly principles. These detailed assessments and the configuration deltas they uncover allow for risk mitigation to be implemented.

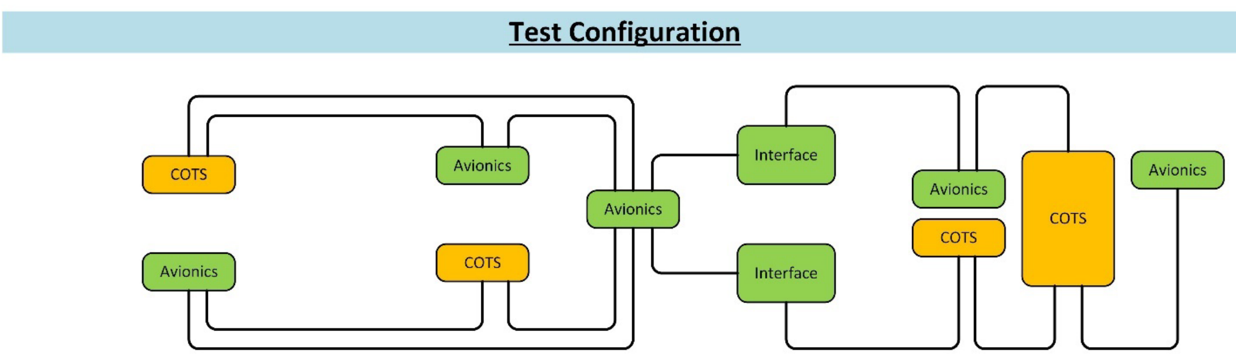


Green = Flight
Orange = Not Flight

Vehicle 1	
Hardware	Flight avionics
Software	Flight software
Cabling	Flight configuration cables
RF	Flight configuration antennas and links
Configuration	Complete flight configuration

Vehicle 2	
Hardware	Flight avionics
Software	Flight software
Cabling	Flight configuration cables
RF	Flight configuration antennas and links
Configuration	Complete flight configuration

FIG. 8 FLIGHT CONFIGURATION ASSESSMENT



Green = Flight
Orange = Not Flight

Test Vehicle 1	
Hardware	Avionics LRUs are flight
Software	Flight software
Cabling	Mixture of flight like and COTS
RF	Flight RF avionics, antennas removed from test
Configuration	Similar flight configuration with test adapters

Test Vehicle 2	
Hardware	Some flight hardware, Backbone hubs not present
Software	Some flight software, no GNC
Cabling	Cables and connections not flight like
RF	Antennas removed from test
Configuration	Not the same configuration as flight

FIG. 9 FLIGHT CONFIGURATION TO TEST CONFIGURATION ASSESSMENT

The flight fidelity of hardware configurations between two elements or within an element under test can be categorized as shown below. These definitions allow detailed definitions of the configurations to has a qualitative assessment approach. It is well accepted within the aerospace community that testing against flight hardware allows for the most thorough assessment of the systems performance and mitigation of risks introduced when not testing with flight or flight equivalent systems.

- Flight Hardware to Flight Hardware Connected and Tested Prior to Flight – "A"
- Flight Hardware to High Fidelity Emulator – "B"
- High Fidelity Flight Like Emulator to COTS Emulator – "C"
- COTS Emulator to COTS Emulator – "D"
- Additionally, as shown in Figure 8, individual elements can be assessed and ranked as combinations (A-A, B-B, C-C).

ASSESSMENT CRITERIA KEY:

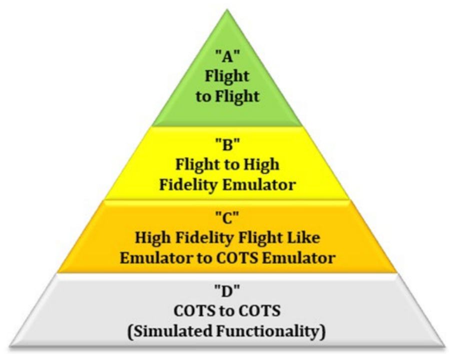


FIG. 10 TLYF FIDELITY ASSESSMENT ELEMENT TO ELEMENT TESTING

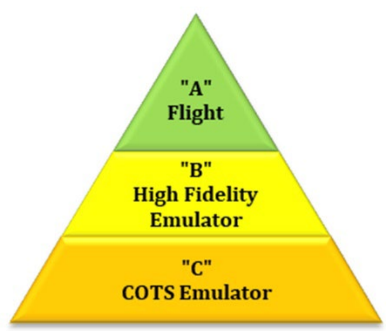


FIG. 11 TLYF FIDELITY ASSESSMENT SINGLE ELEMENT TESTING

C. Fidelity Assessment Process

Requirements which are to be verified/validated by testing and demonstration are typical in scope for the TLYF assessment process. Requirements closed by analysis or inspection are typically not in scope for the TLYF assessment process. As updates occur because of design maturity, controls or mitigations being implemented, the TLYF process will be applied and updated. The TLYF process is designed to engage interfacing programs when applicable and ensures that the proper stakeholders are utilized for final approval of an TLYF exception request. The process also includes pre-coordination opportunities at existing working groups and other test forums.

Definition of the process for assessing, understanding and providing guidance on TLYF gaps is critical and should provide for the following efforts to be captured. The process in Figure 12 provides for the implementation of these items in a systematic process that builds upon each previous definition effort. As summarized previously in Section IV, the major steps of that process are:

(1) Perform a technical assessment of each planned test, intra-vehicle or vehicle-to-vehicle.
(2) Create a TLYF exception to document each gap between flight configuration and operations and test configuration and operations.
(3) Identify the potential impacts to crew safety and mission success from the TLYF exception.
(4) Escalate risk to program for evaluation, disposition, and documentation.
(5) Record assessments and exceptions in a program managed TLYF database.

D. TLYF Fidelity Assessment

The TLYF assessment is a product that captures the technical details of the hardware, software, configuration, testing goals and gaps between the test configurations and the flight configuration. The content of the assessment is captured and controlled within a TLYF database. The TLYF assessment is kicked off when the initial goal to accomplish a test is defined. This allows the TLYF assessment to work in parallel with the test definition and development and avoid time lags which would consume schedule opportunities to reduce risk.

FIG. 12 TLYF EQUIVALENCY FOR TEST VS FLIGHT CONFIGURATION

<u>The TLYF assessment should:</u>

1) Establish the requirement verification evidence and goal of the test
2) Establish the flight configuration and root Concept of Operations activity/event
3) Establish the test configuration
4) Assess hardware similarity
5) Assess software similarity
6) Assess operational similarity
7) Assess characteristics and variables of flight
8) Contain a summary of assessment

The table below provides for positive recording of specific TLYF hardware assessment figures of merit.

Table 1 - TLYF Assessment Criteria #1A– Hardware Similarity

Test Article TLYF Evaluation	Yes/No	Notes/Rationale
1) Is the hardware architecture for Test Article #1 the same as the FLIGHT hardware architecture?		
2) Is the hardware architecture for Test Article #2 the same as the FLIGHT hardware architecture?		
3) Are the interfaces supporting Test Article #1 the same as the FLIGHT hardware interfaces?		
4) Are the interfaces supporting Test Article #2 the same as the FLIGHT hardware interfaces?		
Assessment Criteria 1A Summary: Hardware	**A/B/C/D**	

The table below provides for positive recording of specific TLYF software assessment figures of merit.

Table 2 - TLYF Assessment Criteria #1B Software Similarity

Software Category	Software Item	Version	Native/Ported/ Development/Model	A/B/C/D	Notes/Rationale
Interfacing Software					
Core Software					
Simulation Software					
Test Scripts					
Software Test Procedures					
Assessment Criteria 1B Summary: Software				**A/B/C/D**	**Notes/Rationale**

The table below provides assessment guidelines and figures of merit for evaluation of operational similarity between flight and test scenarios.

Table 3 - TLYF Assessment Criteria #2– Operations Similarity

Test Capability #1:	**A/B/C**	**Notes/Rationale**
Test asset supports all required nominal start to finish mission event and timeline tests.		
Test asset supports all required off-nominal start to finish mission event and timeline tests.		
Test asset supports transitions across operational phases are tested. (e.g. transition from RF to hardline)		
Test Capability #2:		
Test asset will provide fault injection to all required nominal inputs/outputs.		
Test asset will provide fault injection to all required off-nominal inputs/outputs.		
Test assets test each input variable, and the range of each variable is the same as nominal flight conditions.		
Test assets test each input variable, and the range of each variable is the same as off-nominal flight conditions.		
Test Capability #3:		
Test asset uses representative procedure and scripts with TLYF exceptions noted		
Test Capability #4:		
Test specific user input #1		
Test specific user input #2		
Assessment Criteria 2 Summary: Operations	**A/B/C**	**Notes/Rationale**

Assessment Evaluation Key:	
Test asset represents and tests all operations, events and procedures	**A**
Test asset represents and tests some operations, events and procedures	**B**
Test asset represents and tests no operations, events and procedures	**C**
This type of variable is not applicable or in scope for this test. This variable should be evaluated by other tests.	**N/A**

As part of the TLYF assessment for a given system, technical parameter variables are established as criteria that could affect performance of the system. Provided in the table below are examples for a RF system and intended to illustrate the purpose and types of content that would be part of this evaluation but not inclusive of all options. Each assessment would be uniquely tailored to the test elements.

Table 4 - TLYF Assessment Criteria #3– Characteristics and Variables

Test Assets Accurately Represents These Characteristics & Variables of Flight Configuration	**A/B/C**	**Notes/Rationale**
1) Time latencies and constraints are matched with flight conditions		
2) Time responses are matched with flight conditions		
3) Off-nominal transmission characteristic - Power (may have effects on bandwidth)		
4) Off-nominal transmission characteristic - Stochastic Noise (can have effects on bandwidth)		
5) Off-nominal transmission characteristic - EMI/EMC Noise (can have effects on bandwidth)		
6) Off-nominal transmission characteristic - Multipath (can have effects on bandwidth)		
7) RF strings / transmitters represented (including multiple transmitters during joint operations)		
8) Test specific user input #1		
9) Test specific user input #2		

Assessment Evaluation Key:	
Asset represents and tests all characteristics/variables of interface/element	**A**
Asset represents and tests some characteristics/variables of interface/element	**B**
Asset represents and tests no characteristics/variables of interface/element	**C**
This type of variable is not applicable or in scope for this test. This variable should be evaluated by other tests.	**N/A**

E. TLYF Evaluation Process

As referenced on previous pages, the figure below provides for the implementation of TLYF assessment items into a systematic process that builds upon each previous definition effort. A critical part of this process is to ensure that the right questions are being asked early in the process and direct evidence is being given to support each of the answers.

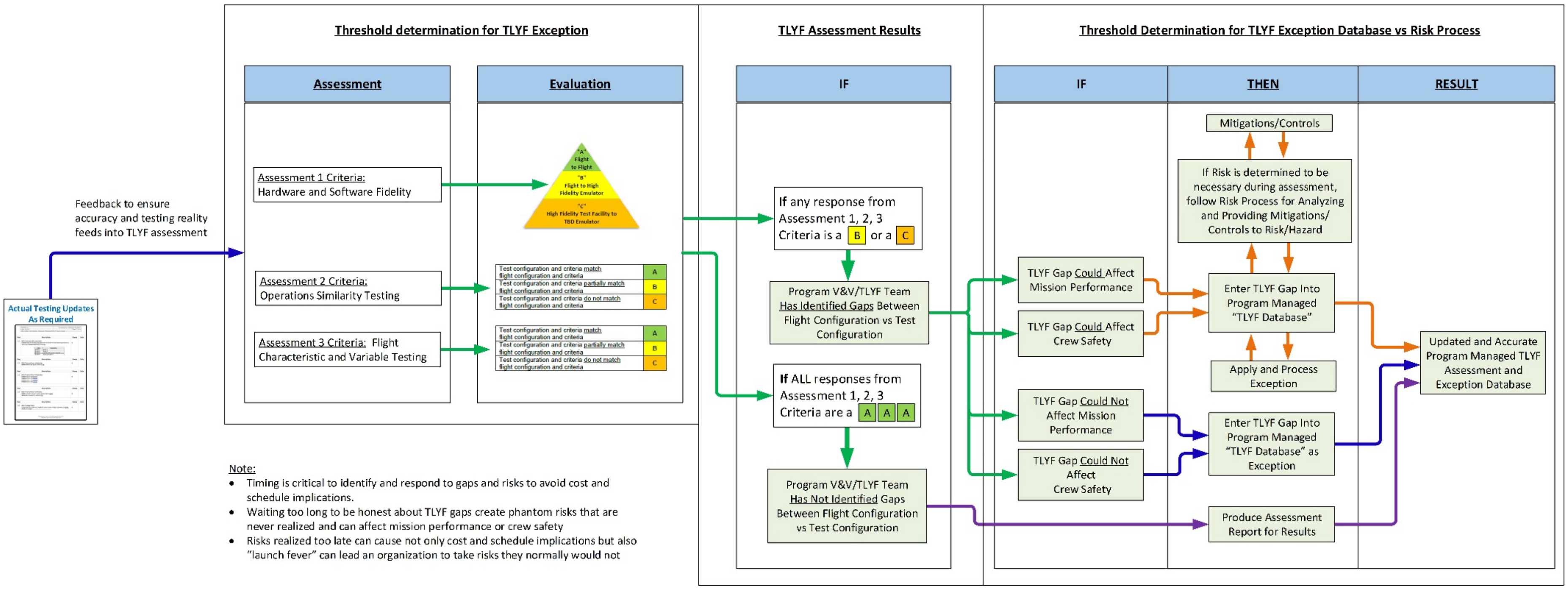


FIG. 13 TLYF EVALUATION PROCESS

The following figure describes the process that allows for the development of the TLYF assessment along with the stakeholder involvement. It is essential to capture the TLYF assessment results for long term technical and programmatic ownership.

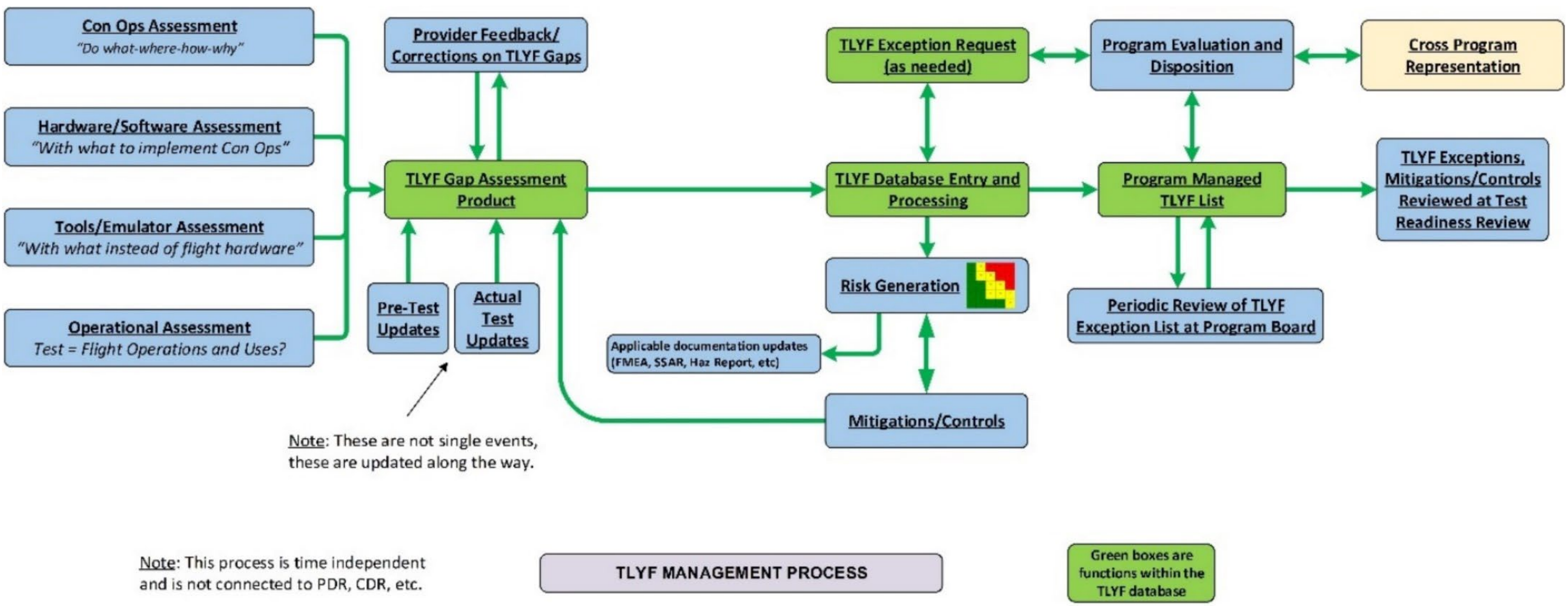


FIG. 14 TLYF MANAGEMENT PROCESS

F. Exception Assessment Process

The Test-Like-You-Fly (TLYF) exception application provides a method for stakeholders to acknowledge and approve known differences between test and flight hardware/software/ configurations. This application utilizes technical data produced during the TLYF assessment. The TLYF exception application is assigned by the program to a TLYF exception application owner along with a respective Technical Point of Contact for processing through the TLYF exception application workflow.

A primary goal of the TLYF exception application is to provide clear technical statements that enable stakeholder understanding. Equally important is avoiding the normalization of deviance—submission of a TLYF exception application does not guarantee approval and it does not make the risk of the TLYF gap any less. A TLYF exception approval is a programs acknowledgment of the TLYF gap, understanding of the risk and approval the TLYF exception is acceptable to the program.

When submitted, the TLYF exception application allows stakeholders to formally determine whether differences between test and flight configurations are acceptable. If an application is rejected, this provides necessary traceability and guidance for implementing changes to the test configuration to reduce risks to crew safety and mission success.

Table 5 - TLYF Exception Application Summary Example

Test ID	Title	Assessment Criteria 1: Hardware & Software Fidelity		Assessment Criteria 2: Operations Similarity			Assessment Criteria 3: Characteristics & Parameter Variables	
		Hardware Pedigree (1A)	Software Pedigree (1B)	Test Capability #1	Test Capability #2	Test Capability #3	RF	Hardline
		A/B/C/D	A/B/C/ n/a	A/B/C	A/B/C	A/B/C	A/B/C	A/B/C
TLYF Gap Could/Could Not Affect Crew Safety?		Yes/No	Yes/No	Yes/No	Yes/No	Yes/No	Yes/No	Yes/No
TLYF Gap Could/Could Not Affect Mission Performance		Yes/No	Yes/No	Yes/No	Yes/No	Yes/No	Yes/No	Yes/No

Have these components flown previously and successfully in this configuration? (flight heritage)	Yes/No	Rationale

Assessment Evaluation Key:

The keys below provide insight into the configuration proposed for this test and the evaluated fidelity compared to flight.

Assessment Criteria 1A - Hardware:	Grade
Flight Hardware to Flight Hardware	A
Flight Hardware to High Fidelity Emulator	B
High Fidelity Emulator to High Fidelity Emulator	C
COTS Emulator to COTS Emulator – "D"	D

Assessment Criteria 1B - Software:	Grade
Native Software	A
Ported Software	B
Development Software	C
Not Applicable	N/A

Assessment Criteria 2 - Operations:	Grade
Asset represents and tests all characteristics/procedures/variables of interface	A
Asset represents and tests some characteristics/ procedures/variables of interface	B
Asset represents and tests no characteristics/ procedures/variables of interface	C

Assessment Criteria 3 - Characteristics:	Grade
Asset represents and tests all characteristics/variables of interface	A
Asset represents and tests some characteristics/variables of interface	B
Asset represents and tests no characteristics/variables of interface	C

VIII. Other Recommended Resources

As noted in the Introduction, this paper is intended to provide introductory information for early career professionals or professionals that do not have years of experience testing flight systems. Examples, source material, and paper content are intended to provide additional information associated with ground testing of space systems as it relates to testing like you fly. In addition to the References, there are two additional resources that can be consulted for further information on Test Like You Fly:

Perhaps the most comprehensive and widely consulted resource on TLYF is a paper produced by The Aerospace Corporation, Aerospace TOR-2014-02537 Rev. A, The Test Like You Fly Process Guide for Space, Launch, and Ground Systems." [11] The TOR addresses both assessment of contract scope and providers compliance or deficiencies that can be remedied early in the development life cycle. It provides a detailed framework, without the fidelity considerations proposed by the process in this paper, to assess and prioritize test objectives associated with mission operational scenarios.

The only NASA center to have its own standard with content specifically dedicated to TLYF is Goddard Space Flight Center. Their standard for Rules of the Design, Development, Verification, and Operation of Flight Systems [12] addresses these topics in sections Requirements 1.08, System End-to-End Testing in final flight configuration and 1.09, Test as You Fly. This standard, as well as NASA's consistent use of the TOR, are important examples of NASA documenting the need to follow these TLYF best practices.

Acknowledgments

Many brilliant engineers, scientists, technicians, analysts, and quality specialists have been performing Test Like You Fly aspects from the beginning of the United State space program. This team builds upon the knowledge of those thousands of folks and the lessons learned across programs and decades of space flight achievements and acknowledges their contributions that we are applying to the Artemis paradigm.

References

[1] *MIL-HDBK 340A, Department of Defense Handbook, Test Requirements for Launch, Upper-Stage, and Space Vehicles, Volume 1: Baselines.* Apr 1, 1999. https://vibrationresearch.com/wp-content/uploads/2021/05/MIL-HDBK-340A.pdf

[2] *MIL-STD-810, Department of Defense Test Method Standard for Environmental Engineering Considerations and Laboratory Tests*. Jan 1, 2000. https://www.atec.army.mil/publications/Mil-Std-810G/Mil-Std-810G.pdf.

[3] Ben Evans. *Firing Up the Shuttle: Looking Back at the Flight Readiness Firings.* AmericaSpace (2015). https://www.americaspace.com/2015/09/20/firing-up-the-shuttle-looking-back-at-the-flight-readiness-firings-part-2/.

[4] John Uri. *45 Years Ago: First Main Propulsion Test Assembly Firing of Space Shuttle Main Engines.* NASA (2023) https://www.nasa.gov/history/45-years-ago-first-main-propulsion-test-assembly-firing-of-space-shuttle-main-engines/.

[5] "Two MPTA Firings Set for December," *Lagniappe*, November 26, 1980: 5. https://ntrs.nasa.gov/api/citations/20050217132/downloads/20050217132.pdf

[6] *Shuttle Independent Assessment (SIA)*, chaired by Dr. Henry McDonald, Director, Ames Research Center (ARC). Mar 7, 2000, https://ntrs.nasa.gov/citations/20000032103.

[7] Berger, Eric, *The Surprise is not that Boeing lost Commercial Crew but that it finished at all*. (2024) https://arstechnica.com/space/2024/05/the-surprise-is-not-that-boeing-lost-commercial-crew-but-that-it-finished-at-all/.

[8] *Space and Missile Systems Center Standard: Test Requirements for Launch, Upper-Stage and Space Vehicles*. SMC Standard SMC-S-016. (2014).

[9] *MIL-STD-810, Department of Defense Test Method Standard for Environmental Engineering Considerations and Laboratory Tests*. Jan 1, 2000. https://www.atec.army.mil/publications/Mil-Std-810G/Mil-Std-810G.pdf.

[10] Dennis R. Moore and Willie J. (Jack) Phelps. *Reusable Solid Rocket Motor—Accomplishments, Lessons, and a Culture of Success*. Marshall Space Flight Center (2011) https://ntrs.nasa.gov/api/citations/20120001536/downloads/20120001536.pdf.

[11] The Aerospace Corporation, Aerospace TOR-2014-02357 Revision A, The Test Like You Fly Process Guide for Space, Launch, and Ground Systems," (2014). https://aerospace.org/sites/default/files/2024-06/TOR-2014-02537-REV%20A%20-20The%20Test%20Like%20You%20Fly%20Process%20Guide%20for%20Space%2C%20Launch%2C%20and%20Ground%20Systems.pdf

[12] *Rules for the Design, Development, Verification, and Operation of Flight Systems*, Goddard Space Flight Center. (2016). https://soma.larc.nasa.gov/lws/pdf_files/3.11%20GSFC-STD-1000Revg_signature.pdf.