Integrated Vehicle Test Bed for IVHM Systems on 2nd Generation RLV[1,2]

Mark L. James, Ed Baroth, Lee Mellinger, Han Park, Tim Stough
Jet Propulsion Laboratory,
California Institute of Technology
4800 Oak Grove Drive
Pasadena, CA 91109
Mark.James@jpl.nasa.gov

Stephen A. Brown
Northrop Grumman Corporation
Integrated Systems
One Hornet Way
El Segundo, CA 90245

Claudia M. Meyer
NASA Glenn Research Center
21000 BrookPark Road
Cleveland, OH 44135

Abstract— The IVHM Virtual Test Bed (IVTB) is a systems integration and test facility being developed by Northrop Grumman Corporation (NGC) and NASA's Jet Propulsion Laboratory (JPL), along with NASA partners Glenn Research Center (GRC) and Ames Research Center (ARC), to support the demonstration and validation of Integrated Vehicle Health Management (IVHM) systems for the NASA's 2nd Generation Reusable Launch Vehicle (RLV) program. The IVTB concept is to validate spacecraft system designs and evaluate new technologies; its focus is to identify and resolve problems at the early stages of development and facilitate new technology transfer. In addition, the IVTB can be used for infusing new technology, verifying the feasibility of new spacecraft designs and concepts; assembling and testing ground and spacecraft elements using the distributed environment capabilities, and supporting post launch operations and mission scenario testing.

For the 2nd Generation RLV program, the IVTB provides a flight-like real-time or non-real-time environment to simulate the integrated operation and evaluation of client subsystems, operating under defined vehicle architecture and a Design Reference Mission. These components can drive a hierarchical health management system comprised of multiple subsystem health managers with each focusing on the health of a particular subsystem. The IVTB enables a system-level IVHM providing on-board and off-board components.

As a proof-of-concept, a simulation of the X-34 RLV vehicle system was chosen as the first experiment at the IVTB. This experiment will focus primarily on two specific subsystems from the X-34: the propulsion feed system (simulation model and real-time diagnostic system developed at GRC and ARC, under the NASA PITEX program) and a liquid oxygen (LO2) tank structure (structural simulation model and structural subsystem health manager developed at NGC). The experiment will use a hierarchical health management system wherein a system-level IVHM will collect and analyze all the subsystem-level diagnostics to demonstrate the diagnostic capabilities not possible with just subsystem level diagnostics.
The IVTB will facilitate system level design trades, fault injection and fault isolation, and concurrent engineering with the goal of supporting development and test of flight and ground systems. This testbed will provide a base of re-useable components and a rich set of state-of-the-art tools to enable rapid prototyping, and an incremental development environment for future space flight systems.

TABLE OF CONTENTS

1. INTRODUCTION

[1] 0-7803-7231-X/01/$10.00/© 2003 IEEE
[2] IEEEAC paper #008, Updated Sept 27, 2001

The TA-5[3] IVHM Virtual Test Bed (IVTB) is a systems integration and test facility being developed by Northrop Grumman and NASA to support the demonstration and validation of Integrated Vehicle Health Management (IVHM). The home facilities of IVTB reside at the Flight System Testbed (FST) at NASA's Jet Propulsion Laboratories (JPL) in Pasadena, CA.

IVTB is similar in concept to the kind of Systems Integration Laboratory (SIL) typically created during any complex aerospace system development program to demonstrate and validate that the vehicle subsystems, and in particular, vehicle system avionics, operate well together. The IVTB is different from typical SILs in that:

1. The IVTB is being made available now, during risk reduction, with the specific intent of integrating a variety of potential subsystem models, which may themselves be fairly immature.

2. The IVTB is focused on maturing vehicle health management, including synergistically leveraging subsystem health management to create an integrated system level health management system. For this reason, the IVTB places a much greater emphasis on modeling and demonstrating the handling of off-nominal behaviors, in addition to nominal behaviors.

From the perspective of subsystem health management (SSHM) developers, IVTB provides facilities and integrated experiments to mature and demonstrate:

- The ability of a given SSHM to communicate with system level health management using standard interface protocols being developed under SLI TA-5.
- How a given SSHM communicates with and provides value to flight and ground operations (TA-9 and TA-4) in the context of an integrated system.
- How a given SSHM can support other subsystems, and the overall IVHM Technology Performance Metrics (TPMs), demonstrating program value beyond the value to its own subsystem.
- How a given SSHM can leverage system level health management to improve its effective TPMs), including fault detection, fault isolation, and false alarm reduction.

In addition, it is expected that the IVTB will be used for integration and validation of any integrated health management experiments that may be flight-tested under the 2GRLV Risk Reduction program.

[3] This work was performed as part of NASA's ongoing Space Launch Initiative (SLI), 2nd Generation Reusable Launch Vehicle Program, Technology Area 5 (TA-5), Integrated Vehicle Health Management. The TA-5 Project is lead by NASA's Ames Research Center for the SLI Program, headquartered at the Marshal Space Flight Center (MSFC) in Huntsville, AL.

IVTB Execution Modes

The IVTB is intended to act as:

- A framework and facility for maturing, demonstrating, and validating subsystem and system health management technologies matured or proposed for SLI.
- A tool for developing, maturing, and demonstrating technologies and procedures required to demonstrate and validate system and subsystem health management technologies for flight and ground use in the 2GRLV timeframe. It is expected that, by Full-Scale Development (FSD), the capabilities pioneered by the IVTB will become an integral part of a larger 2GRLV System Integration Laboratory (SIL) used to validate all vehicle flight and ground software systems.

The IVTB will be required to incorporate and integrate simulation and diagnostic software at widely varying levels of maturity, from a wide variety of suppliers. For this reason, the IVTB architecture is being kept as open as possible, with many of the detailed interfaces and requirements to be negotiated on a case-by-case basis. The flexibility of the IVTB, coupled with JPL's extensive experience in interfacing a wide variety of experimental systems, means that the IVTB can and will be reconfigured to meet a wide range of requirements for each experiment.

For clarity, this document primarily focuses on the simplest mode of operation, consisting of:

1. A non-real-time, all software (virtual), simulation of the client systems, which is used to create a "canned" sequence of sensor values, followed by

2. An integrated set of diagnostic systems, running in real-time on flight avionics class hardware, basing their diagnostics on the previously generated sensor values.

In this "open-loop" mode, the simulated client subsystems are not required to execute in real-time, but instead will be "pre-run" to generate the simulated sensor stream. This supports the common case where detailed client system simulations cannot execute at real-time speeds, as well as allowing proprietary simulation models to be held and run at vendor sites.

This basic form can be extended in various ways, including:

- Replacement or augmentation of some client subsystem sensor streams with data collected during hardware ground or flight tests.
- Real-time execution of simulation models that can support the required data output rates.
- Execution of diagnostic systems at slower than real-time (this is sometimes required to help mature a SSHM and/or its technologies).
- Parallel/coupled execution of client system simulations and diagnostic systems, allowing

closed-loop management of the failure as it occurs. This is the most complex mode of operation envisioned, because it requires that the simulation models and diagnostic systems run at the same frame rate (whether real-time or non-real-time).

IVTB Support for Hardware Testing

Although the baseline IVTB is focused on demonstration and validation of health management systems using simulated client system hardware, the IVTB is expected to support testing of actual client system hardware in a variety of ways:

- As SLI hardware becomes available, the simulated sensor output from appropriate subsystems can be validated and/or replaced by the results of actual hardware tests. Validation of subsystem and sensor simulations remains crucial to acceptance of system validations based on these simulations.
- In some cases, it may be possible and desirable to replace selected subsystems in the IVTB laboratory with actual real-time hardware components.
- IVTB simulations can be used to walk through ("dry run") hardware tests before they are conducted, identifying both expected nominal behavior and various failure modes that may occur during the test. This is especially important if the associated SSHM will be used in active mode during the hardware test, i.e. to automatically shut down tests that threaten to damage the test article or fixtures. This usage extends from component hardware tests all the way through flight test.
- As systems become more complex and expensive, testing specific modes, particularly failure modes, rapidly escalates in cost, schedule, and risk. Generally, hardware tests are required to reduce the uncertainty in specific execution parameters (e.g. the actual vs. expected temperature of an operational heat-exchanger). In some cases the parameters of interest represent nominal behaviors, in others off nominal behaviors (in terms of performance and health management). IVTB-like simulation models and diagnostic systems can be used during the design phase to evaluate the sensitivities of nominal, off nominal, and diagnostic behaviors to specific parameters in question. By comparing the system sensitivities to the engineering uncertainty in specific parameters, hardware tests can be prioritized, justified, or eliminated altogether; e.g. if the engineering uncertainty in the actual heat-exchanger temperature is ±20°, but the (simulated) system operates properly over ±40°, it may not be necessary to perform this hardware test at all.

2. IVHM Virtual Testbed Concept

Figure X shows the functional layout of a typical SSHM in operation. The client subsystem will, of course, be implemented in actual hardware (e.g. an actual engine); and this subsystem will produce an infinite number of time-continuous physical parameters (temperatures, pressures, flow rates, etc). Some finite number of these parameters will be measured by sensors, where each sensor has some schedule for taking readings of its associated parameter, and passing those readings on to a computer responsible for collecting sensor parameters. Some of the parameters measured will be inherently digital (discrete) in nature[4]. The remainder, such as temperature or pressure, is inherently continuous in both time and magnitude.

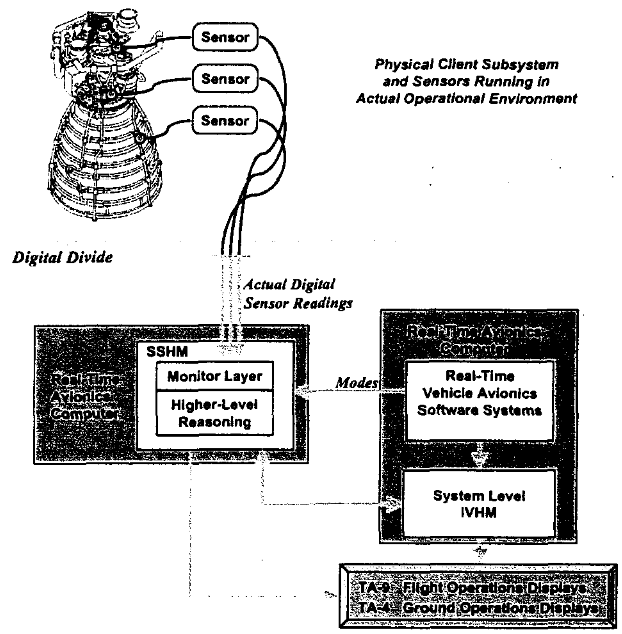

Figure X

However, in all such cases, somewhere between the sensor and the collecting computer, there will be an analog to digital conversion that will cause these continuous parameters to be presented to the SSHM computer as a finite set of discrete values. In the figure this is shown as the "digital divide", and it has some important implications for health management:

1. The subsystem health manager never sees "continuous" values. From the computational perspective, SSHM receives a finite stream of discrete sensor values, and this stream represents its entire view of the client system.
2. In terms of validating operation of a SSHM, the only thing that matters is that the SSHM is

[4] In addition to readings of physical parameters, such as temperature and pressure, SSHMs are also driven by system state information, such as commanded mode changes, state changes annunciated by other systems, etc. These are generically shown in the figure as "modes" information. Without loss of generality, this part of the discussion will be simplified by simply lumping all such sources of mode information in with the discretized sensor readings and calling the whole thing the "sensor stream".

presented with a sensor stream that accurately represents the sensor stream that would be generated by the physical client system in operation. The SSHM neither knows, nor cares, how it is generated on the other side of the digital divide.

This means that an actual SSHM software module can be fully demonstrated and validated by executing it on the actual flight hardware (or functional equivalent), when presented with a representative sensor stream, *regardless of how the sensor stream is actually produced.*

Notes:

1. There are some side issues as to what "executing on actual flight hardware" really means. In particular:

   - Does it matter whether the flight computer is running in an operational environment (vibration, radiation, etc), and
   - If the computer is to be shared, does it matter if the total system load is representative over time?

   The answer to (a) causes some heated debates, but we will claim the answer is generally "no" from the perspective of validating the SSHM software. The answer to (b) is generally "yes", but significant utilization data can also be generated in the absence of other loads, if the flight computer/operating system is instrumented to collect such data.

2. There is often confusion as to whether sensors should be considered part of the client system or the SSHM. In general, IVTB considers sensors part of the client system, for strictly practical reasons:
   - SSHMs typically make use of both *native* and *dedicated* sensors, where:
   -
     - Native sensors are sensors included in the client system design without regard to health management; i.e. they support the basic functional control of the subsystem.
     - Dedicated sensors are sensors that are added to the client system design specifically in order to support health management.

   Although the distinction between native and dedicated sensors is important in terms of justifying the cost of additional sensors, once a sensor is accepted the SSHM does not really distinguish between the two.

   - Sensors (native or dedicated) are themselves important, and fallible, components of the client system, and the SSHM software needs to treat them as such; i.e. the health of sensors should be prognosed/diagnosed along with the health of the rest of the client system. Eliminating false alarms due to faulty sensors is a key goal of the IVHM risk reduction effort.
   - Sensors are physical hardware devices with idiosyncratic physical effects (noise, drift, etc), just like the rest of the client subsystem hardware, and these effects occur upstream of the digital divide.

3. For simplicity, the figures and discussion will generally focus on on-board in-flight SSHM and FV IVHM systems. IVHM in general, however, encompasses both on-board and off-board components, which may run in-flight and/or out-of-flight (while the vehicle is on the ground). The IVTB can be configured to support demonstration of any or all of these components, depending on the requirements of each configuration and experiment.

Validating a SSHM

There are, in general, four separate, but related, questions in validating a subsystem health manager for prognostics and diagnostics:

1. *FMECA*. Are the physical characteristics (signatures) of nominal and failure modes understood sufficiently (correctness and detail) that they should be recognizable and distinguishable by a hypothetical SSHM?
2. *Testability*. Given (1), are the sensor test points selected sufficient to recognize all required nominal and failure modes, and to distinguish between these modes to the extent required to support failure management and the maintenance concept?
3. *Sensor Accuracy and Reliability*. Given (1) and (2), are the physical sensors used sufficiently accurate and reliable, in terms of noise, drift, failure effects, or other operating characteristics, that the implemented SSHM should be able to recognize all required nominal and failure modes, and to distinguish between these modes to the extent required to support failure management and the maintenance concept? This is effectively an extension of the deterministic testability analysis (2), allowing for imperfect sensors.
4. *Software Reliability*. Given (1), (2), and (3), does the SSHM software,
   - Acceptably prognosed/diagnose all required nominal and failure modes, including required levels of fault isolation (correctness)?
   - Over the entire operating range (nominal and failed) of the client subsystem (robustness)?

Virtual (all software) testing, such as that enabled by the IVTB, primarily validates (2) and (4), based on the assumed (simulated) characteristics of the physical hardware. In addition, virtual testing can be used to determine the sensitivity of the SSHM to uncertainties in the actual hardware characteristics.

Verifying the physical characteristics of the client system, (1), as measured by the physical sensors, (3), can only be done through actual hardware tests. However, by comparing SSHM sensitivities (generated from the simulated system) with engineering judgment about the uncertainty in modeled physical parameters, hardware test requirements can typically be minimized and focused on a relatively small set of critical test points. This corresponds to a more general trend in aerospace system development towards using hardware tests to validate the simulation models, and the simulation models to validate system operations.

3. IVTB Overview

The best way to ensure that a sensor stream presented is fully "representative" is to generate it directly from an actual client system and sensor suite, running in the actual operational environment. However, several factors make this approach infeasible in many cases:

- Early in a risk reduction or development program subsystem hardware frequently does not exist at all, and the purpose of the experiment is to co-mature the SSHM and client systems. Even when hardware does exist, it may not be available for data generation, or there may be no practical way to generate responses in the operational environment.
- Frequently the goal is to demonstrate the SSHM's response to potentially catastrophic failure events, where introducing such events into actual hardware might be excessively expensive and/or hazardous.
- Even under the best conditions, hardware tests are generally much more expensive, and have much more schedule impact, than simulation based tests.

Thus, although the IVTB can, and will, make use of sensor streams recorded from actual hardware tests where available, the baseline IVTB uses a combination of real-time and non-real-time simulation models to execute the SSHM without requiring use of the actual client system hardware.

Figure X shows the basic IVTB simulation layout for a single subsystem. On the left side are software modules produced by the subsystem developer, while IVTB provides physical facilities and the software components shown on the right. The SSHM shown in the lower left is the same real-time SSHM as would be used on the operational vehicle, although not necessarily in its final, mature, form.

Above the digital divide, the physical (hardware) client subsystem and sensors are replaced by a software simulation, capable of expressing the sensed parameters of the system in both nominal and failed states. This simulation will be driven by a "non-real-time vehicle level system simulation" (master simulation) developed by the IVTB/FST team that will:

- Synchronize all subsystem simulations with the master vehicle simulation, based on the selected vehicle architecture and reference mission, including supplying externally applied modes and loads as required, and
- Coordinate any subsystem generated modes or loads that need to be transmitted between modeled subsystems.

Thus, among its other duties, the master simulation is specifically responsible for coordinating and time sequencing simulated faults whose signatures (physical effects) spread across multiple subsystems.

The output from the client subsystem simulation is expected to be a file of time stamped sensor readings, as they would appear on the downstream side of the digital divide. These sensor readings can then be fed to the appropriate SSHM, typically in real-time, to demonstrate the ability of the SSHM to correctly diagnose and/or prognose faults within the context of a full vehicle. Execution of the various SSHMs below the digital divide are synchronized by an IVTB supplied real-time vehicle level simulation, which also supplies the appropriate mode annunciations, etc. that would ordinarily be provided by other onboard avionics systems. TA-5 will normally supply the system level integrating IVHM, and IVTB (JPL) supplies the overall system GUIs. In addition, individual SSHMs may elect to supply additional GUIs, or other interfaces, as required to support their specific demonstration requirements.

As long as the activities of the IVHM software (subsystem + system level IVHM) do not impact the failure progression:

- The client simulations above the digital divide can be run independently of the SSHM software below the digital divide, and
- This client system simulation does not need to be executed in real-time.

This "open-loop" mode of operation essentially supports demonstration of diagnostics and/or prognostics, but not the active remediation (failure management) of any failures found.

In order to demonstrate "closed-loop" active remediation, the client system simulation and the IVHM executions must be interleaved and synchronized across the digital divide. Unfortunately, many client system simulations, of sufficient fidelity to be of interest, cannot be executed in real-time on available hardware. In this case, two possible approaches are:

1. Execute the IVHM codes at slower than real-time (typically by inserting blocking states or delays).
2. Iterate the client simulation and IVHM executions, modifying each client simulation to match the response of the IVHM, until the response stabilizes (convergence).

Both of these approaches present significant complications, so the intent, at least initially, is to focus on demonstrating the diagnostics and prognostics that would support active remediation, but not the remediation itself.

Although the IVTB facilities can be used to demonstrate operation of a single stand-alone subsystem, e.g. to prototype a planned hardware test, its primary use within TA-5 is to demonstrate synergistic health management of multiple interacting subsystems.

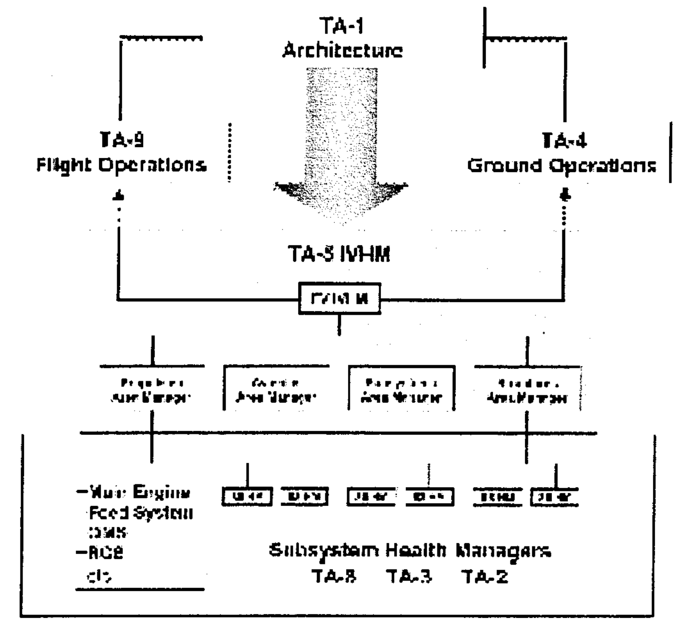

Figure 2

In the Risk Reduction program, each set of demonstration runs in the IVTB is referred to as an "experiment". Each experiment has an associated "configuration", including a reference vehicle architecture and design reference mission (DRM). As shown in Figure 2, the configuration chosen for each experiment is intended to represent a configuration of specific concern to one or more of the TA-1 Architects, both directly and as expressed through their risk reduction agents for flight (TA-9) and ground (TA-4) operations. For reasons of cost, schedule, and relevance, configurations will not implement a complete vehicle architecture (all subsystems), but instead will focus on just a few subsystems that act directly on the functionality to be demonstrated. Similarly, the selected "design reference mission" may cover only a few critical minutes of a much longer overall mission.

Each experiment is managed by an Integrated Product Team (IPT), headed by TA-5, and includes representation from all of the stakeholders shown in Figure 2. In addition to developing the detailed system architecture, DRM, and experiment objectives, the IPT also identifies cross-subsystem failure modes to be demonstrated, details any required Interface Control Documents (ICD), and evaluates the system level results of the experiment. In addition, it is expected that most subsystem suppliers will have additional demonstration objectives particular to their own subsystem and risk reduction goals.

The IVTB planning process to date relies on the assumption that most of the subsystem providers are concurrently developing SSHMs as part of their risk reduction efforts under SLI/2GRLV. This means:

- SSHMs are (should be) designed to interface with other vehicle software systems, including higher-level IVHM managers, through some kind of standard messaging protocol. TA-5 is actively working with the other TAs to develop a baseline SLI IVHM ICD, which will eventually create a standard syntax as well. However, initial IVTB experiments are expected to develop ad hoc ICDs as required for each experiment.
- The existence of client subsystem and sensor simulation models is inferred from the need by each individual subsystem developer to develop, mature, and eventually validate their own SSHM. Integrating sets of such models will, however, be more of a challenge, simply because they may not have been developed with such functionality in mind. The impact of adding such functionality to each subsystem will have to be evaluated on a case-by-case basis as part of the schedule and resource planning for each experiment.

SLI subsystem designs, simulation models, and SSHMs are expected to be relatively immature during the risk reduction program, and interface standards are still under development. To accommodate these realities the IVTB architecture and concept of operations have been kept as open and flexible as possible, and every effort will be made to accommodate the special needs of each supplier. JPL's Flight System Testbed (FST) has extensive experience in integrating a wide variety of hardware and software directly with each subsystem to ensure the best possible results. The FST layout is shown in Figure 2a.

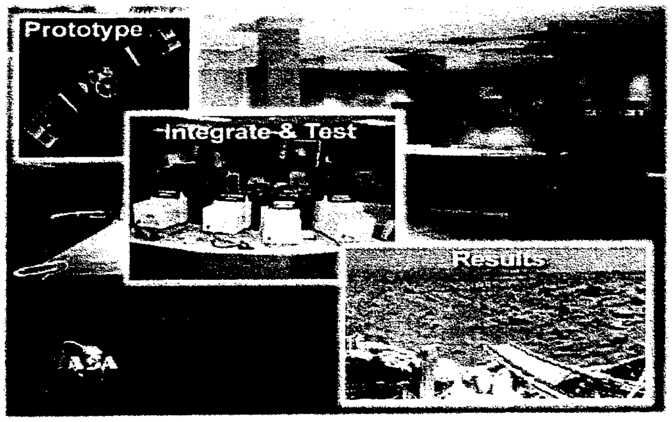

Figure 2a

4. Simulation Architecture

Architecture

The current baseline architecture, shown in Figure 3, runs each subsystem simulation interacts only with the master simulation; i.e. there is no direct cross-communication between the subsystem simulations. The vehicle-level master simulation will:

- Coordinate each of the subsystem simulations, including providing simulated time steps, external loads, vehicle modes and states, etc.
- Resolve interfaces between subsystems; e.g. by transferring physical quantities generated as output by one subsystem simulation to inputs provided to another subsystem on the next time step.

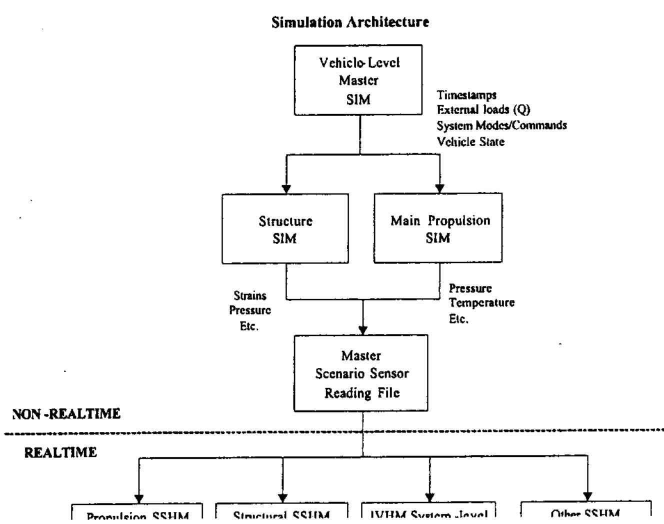

Figure 3

As part of the experiment development process, specific interfaces will be developed between each subsystem simulation and the master simulation (developed by IVTB/FST). Interfaces (boundary conditions) between the modeled subsystems will also be defined in the experiment IPT, but these will actually be implemented by transferring data through the master simulation as shown. This approach minimizes the number of defined interfaces (one per subsystem), and places any data translation requirements in the master simulation controller. As discussed below, by using TRAMEL as the communications interface between the master simulation and the various subsystem simulations, it is possible to interface over the internet or VPN to subsystem simulations at sites other than JPL.

Simulation Models

Each client subsystem simulation model may be generated from real-time or non real-time simulations, or from hardware tests, including direct recording from flight and ground hardware tests. Each subsystem simulation will generate a file with time stamped sensor readings (the sensor stream) consistent with the modes, nominal/failure states, etc. The file output will then be placed in the Master Scenario Simulation File, along with other subsystem simulations as shown in Figure 3. Alternatively, subsystems may arrange to save their own idiosyncratic sensor files.

During Option I a simplified experiment will be constructed in order to validate the IVTB concept, and assist in development of the testbed facilities. This experiment consists of the X-34 main propulsion feed system and LO_2 tank structural models. The client (simulation) and diagnostic models to be used for the propulsion feed system are extensions of the models created previously under the NASA NITEX and PITEX projects. Client (simulation) and diagnostic models of the X-34 segmented liquid oxygen (LO_2) tanks will be developed during Option I by the NGC TA-2 Structural Health Management task. The further details of the simulation can be found in the "IVTB Experiment Test Plan."

During Option II, more ambitious experiments are planned based on the emerging 2GRLV architectures by Boeing, Lockheed, and Northrop Grumman. Subsystem simulation and diagnostic modules (discussed in the next section) will be assembled in order to demonstrate and validate aspects of IVHM focused on the specific needs and concerns of the TA-1 architects.

Implementation

A FST is a controlled environment in which to observe and evaluate experiments in a laboratory environment where hardware and software integration and test activities are supported. It is a computing environment to support software development, instrumentation, configuration, measurement, and analysis. It is also a simulation environment containing an integrated collection of models, simulators, and prototypes for the purpose of observing and evaluating aggregate system behavior. The FST can create a "virtual" spacecraft in the FST by linking, in any combination, real hardware with simulated components and subsystems or just simulated components and subsystems.

There will be the ability to run client models from outside of the FST, but there are some issues such as security (which is currently undergoing changes at JPL) whose impacts are not yet well understood.

5. Subsystem Health Management (SSHM) Environment

Architecture

The baseline real-time IVTB health management demonstration architecture is shown in Figure 4. The Master Scenario Simulation File sends time-stamped sensor data, recorded as discussed in the previous section, to each of the SSHMs. A playback controller will regulate the speed at which the sensor data is sent out. Nominally, the sensor data will be sent out in real-time, but it may be sent slower than real-time for software diagnostics. Although initial IVTB experiments will complete the non real-time simulations (discussed in the previous section) before running any of the real-time software diagnostics, the ability to run the diagnostics slower than real-time will also allow experiments where the client simulations and their diagnostics run in parallel. This, in turn, will allow demonstrating the ability of an active health management system to manage failures in-flight.

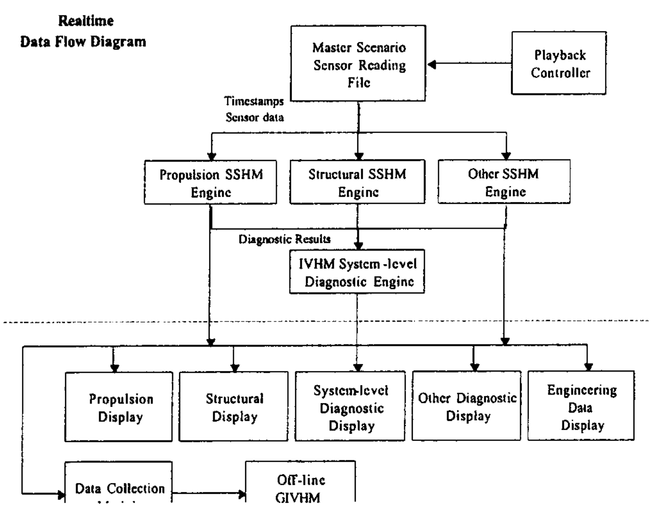

Figure 4

The SSHM software is expected to execute and generate results in real-time. In the simplified Option I experiment, IVTB will host the propulsion feed system SSHM (supplied by PITEX), the LO_2 tank structure SSHM (supplied by NGC), and the system-level IVHM health management software (NGC). Each SSHM will be a complete stand-alone module and will be able to run independently. The system-level health management software will integrate the diagnostic results from each SSHM software module.

The results of the SSHM software will be sent to specific displays as well as a Data Collection Module. The specific displays will display the subsystem and the overall system-level health status. The subsystem-specific displays will be designed and supplied by the SSHM software providers. JPL will assist in the hosting of the displays at the IVTB.

FST/IVTB Environment

The FST/IVTB environment provides a mixture of software and hardware platforms for hosting a variety of simulations and actual hardware solutions. Each subsystem provider will have his or her own specific set of interface and platform requirements. The FST/IVTB personnel have extensive experience in developing, integrating, evaluating, and testing various subsystem components within a heterogeneous distributed test-bed architecture. As part of the FST customer support process, a given subsystem provider will be matched up with the FST/IVTB resources required to accomplish the integration and test activities. An FST engineer will work together with the subsystem provider to design an architecture extension that allows for a seamless incorporation of that providers hardware and software into the FST/IVTB environment.

The FST currently supports Unix, SunOS, SGI Irix, HP-UX, VxWorks, Linux, NT4.0, and Win2k Operating systems. Communication is mostly by TCP/IP over Ethernet, direct links by IEEE 282, 488, 1394, and VME, 1553/1773, Compaq CPI busses. The supported simulation protocol is the FST developed TRAMEL.

6. IVTB Experiments

Although the IVTB facilities can be used to demonstrate operation of a single stand-alone subsystem, e.g. to prototype a planned hardware test, its primary use within TA-5 is to demonstrate synergistic health management of multiple interacting subsystems.

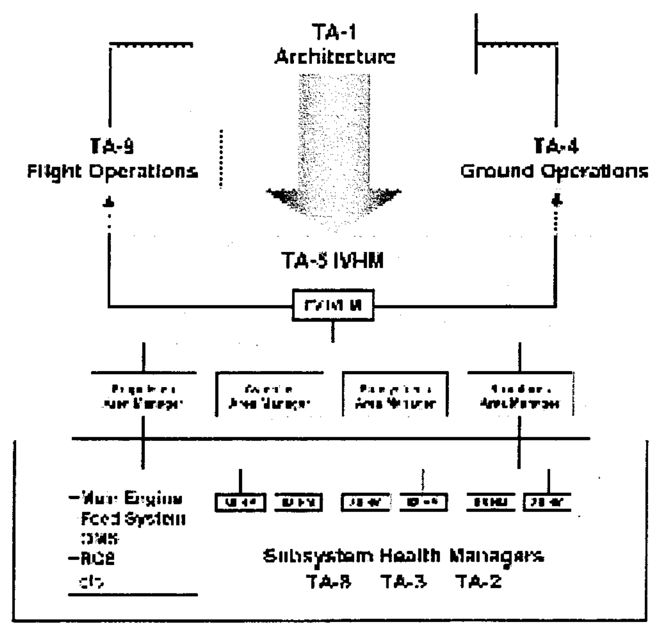

Figure Y

In the Risk Reduction program, each set of demonstration runs in the IVTB is referred to as an "experiment". Each experiment has an associated "configuration", including a reference vehicle architecture and design reference mission (DRM). As shown in Figure Y, the configuration chosen for each experiment is intended to represent a configuration of specific concern to one or more of the TA-1 Architects, both directly and as expressed through their risk reduction agents for flight (TA-9) and ground (TA-4) operations. For reasons of cost, schedule, and relevance, configurations will not implement a complete vehicle architecture (all subsystems), but instead will focus on a few subsystems that act directly on the functionality to be demonstrated. Similarly, the selected "design reference mission" may cover only a few critical minutes of a much longer overall mission[5]. This is already in the IVTB Overview Section.

Each experiment is managed by an Integrated Product Team (IPT), headed by TA-5, and includes representation from all of the stakeholders shown in Figure Y. In addition to developing the detailed system architecture, DRM, and experiment objectives, the IPT also identifies cross-subsystem failure modes to be demonstrated, details any required ICDs, and evaluates the system level results of the experiment. It is expected that most subsystem suppliers will have additional demonstration objectives particular to their own subsystem and risk reduction goals.

The IVTB planning process to date relies on the assumption that most of the subsystem providers are concurrently developing SSHMs as part of their risk reduction efforts under SLI/2GRLV.

- SSHMs are (should be) designed to interface with other vehicle software systems, including higher-level IVHM managers, through some kind of standard messaging protocol. TA-5 is actively working with the other TAs to develop a baseline SLI IVHM ICD, which will eventually create a standard syntax as well. However, initial IVTB experiments are expected to develop ad hoc ICDs as required for each experiment.
- The existence of client subsystem and sensor simulation models is inferred from the need by each individual subsystem developer to develop, mature, and eventually validate their own SSHM. Integrating sets of such models will, however, be more of a challenge, simply because they may not have been developed with such functionality in mind. The impact of adding such functionality to each subsystem will have to be evaluated on a case-by-case basis as part of the schedule and resource planning for each experiment.

SLI subsystem designs, simulation models, and SSHMs are expected to be relatively immature during the risk reduction program, and interface standards are still under development. To accommodate these realities the IVTB architecture and concept of operations have been kept as open and flexible as possible, and every effort will be made to accommodate the special needs of each supplier. JPL's Flight Systems Testbed (FST) has extensive experience in integrating a wide variety of hardware and software, and FST will work directly with each supplier to ensure the best possible results.

Configuration 1: X-34 Propulsion System

A configuration centered on the X-34 main engine feed system has been selected for the first IVTB experiment, to be demonstrated during the first quarter of CY 2003. This experiment configuration relies heavily on an existing feed system simulation model and real-time diagnostic system, developed by the NASA Propulsion IVHM Technology Experiment (PITEX) team[6]. PITEX is a key element of the TA-5 program, and their experience in virtual testing of diagnostic systems helps to direct design of the IVTB described is this paper. The X-34 was selected by the PITEX team because it had direct relevance to proposed 2GRLV architecture concepts and because detailed simulations of key X-34 Main Propulsion System (MPS) components that could be used for failure simulations were readily available. . As a sub-scale RLV demonstrator, the X-34 was specifically designed to incorporate/demonstrate concepts and technologies under consideration for 2GRLV. In addition, the X-34 propulsion concept is analogous to some of the air-launched concepts that have been considered by the TA-1 architects in that it included LO2 and RP-1 as the propellants, relied on in-flight conditioning of the LO2 and featured rocket propulsion coupled with aerodynamic lift.

The Design Reference Mission selected for the first IVTB experiment is the Captive Carry portion of the X-34 mission phase. During this phase of operation, the X-34 is carried to the required launch altitude of 38,000 feet while it is attached to the underside of an L-1011 aircraft. The engine is not running, and most of the other subsystems of the MPS are in a quasi-static state, except for the LOX and RP-1 subsystems.

The X-34 experiment will be developed concurrently with creation of the IVTB core facilities, and will be used to wring out the integrated experiment process. In particular, composition of the IPT for a particular configuration, how and when they interact, and some idea of the costs involved in participation will all be baselined using this configuration.

In addition to the PITEX developed feed system models and diagnostic system, Configuration 1 includes a structural simulation of the X-34's segmented LO2 tank, and a prototype real-time structural diagnostic system. Both the simulation model and the structural subsystem health manager are being developed by NGC under SLI Task Area 2 (TA-2), Airframe Technologies. TA-5 is developing the integrating top-level simulation controllers (JPL) and

[5] One implication of this is that the both the client subsystem simulation and the SSHM must be "initializable" to a nominal state appropriate to the start of the selected DRM. Providing this facility is the responsibility of the subsystem provider.

[6] PITEX is led by NASA's Glenn Research Center, with major participation from Ames Research Center and Kennedy Space Flight Center.

the system level health manager (NGC). In the cross-subsystem failure modes to be demonstrated, the system level health manager will use information from the structural health manager to verify, refute, and/or disambiguate apparent failures (alarms) reported from the feed system health manager. Further details on this particular experiment can be obtained from the authors, or through the NASA SLI TA-5 Project Office.

7. Future

8. Conclusion

References

[1] Mackey, R., James, M., Park, H. G., Zak, M., "BEAM: Technology for Autonomous Self-Analysis," IEEE Aerospace Conference, Big Sky, Montana, March 2001.

[2] Zak, M., Park, H. G., "Gray-box Approach to Fault Diagnosis of Dynamical Systems," IEEE Aerospace Conference, Big Sky, Montana, March 2001.

[3] Preisendorfer, R. W., Mobley, C. D. (editor), Principal Component Analysis in Meteorology and Oceanography, New York, Elsevier, 1988.

Biographies

Stephen Brown is currently serving as Chief Scientist and Deputy Project Manager for the Northrop Grumman TA-5 IVHM Project. He has been with NGC since 1984, developing and maturing technologies for a variety of advanced programs. He received his B.S in Physics, M.S. in Mechanical Engineering, and Ph.D. in Aerospace Engineering all from the University of California at Los Angeles. His research interests span a wide range across aerospace engineering and computer science, including multidisciplinary optimization, aeroelasticity, and intelligent/autonomous systems.

Mark L. James is a senior researcher of the Ultracomputing Technologies Research Group in the Information and Computing Technologies Research Section within NASA's Jet Propulsion Laboratory, Pasadena, CA. At JPL Mark is Principal Investigator and Task Manager in real-time inference and knowledge-based systems. His primary research focus is on high-speed inference systems and their application to planetary and deep spacecraft systems, and, medical applications. His expertise includes core artificial intelligence technology, high-speed real-time inference systems, flight system software architectures and software design and implementation for those systems. Mark has received a number of NASA awards which include: NASA Software of the Year Award Nominee, JPL Team Excellence Award, JPL JET Productivity Award, Major NASA Monetary Award, NASA Certificate of Recognition, NASA Achievement Award, NASA Exceptional Service Medal and NASA Exceptional Service Plaque for 1989. He is author of two inventions, 46 scientific and technical papers and 10 technical manuals.

Lee Mellinger is the Manager of the Flight System Test bed in the Flight System Section at the Jet Propulsion Laboratory, California Institute of Technology. He has been with JPL since 1977. He received his B.S. in Business Administration (Quantitative Methods) and M.S. in Computer Science from the California State University, Northridge. He has worked in systems of the Deep Space Network and the Advanced Multimission Operating System. He was the Data Acquisition and Command Element Engineer for the Mars Observer Project and the Realtime Ground System V&V Engineer for the Cassini Project among other assignments. He has coauthored a paper on the testing of object oriented programs and has authored internal papers on software testing and a study of spacecraft integration and testing His interests are in software test engineering, spacecraft ground and flight systems.

Han Park is a senior member of the technical staff in the Ultracomputing Technologies Research Group in the Information and Computing Technologies Research Section at the Jet Propulsion Laboratory, California Institute of Technology. He joined JPL in 1999 and performs research and development of fault detection/diagnosis algorithms for aircraft and spacecraft systems. He received his B.S. in M.E. at the University of California at Berkeley, S.M. in M.E. at MIT, and Ph.D. in Aeronautics at the California Institute of Technology. His research interests are in the areas of vehicle health monitoring, signal processing, image processing, pattern recognition, color recognition, quantitative visualization, fluid mechanics, and heat transfer.