System
Architecture
Virtual
Integration

SAVI AFE 59 Report

Summary Final Report

Document ID: SAVI-AFE59-09-001

Date: 5/15/2015
Issue: Version 1.0

Release Status: Approved by PMC for Public Release

Author(s): J. J. Chilenski
D. T. Ward

Approved by PMC: 5/31/2015

Aerospace Vehicle Systems Institute
102 Graphic Services Bldg. - MS 3126
College Station, TX 77843-3126
Office: +1-979-845-5568
FAX: +1-979-845-1696
Web: www.avsi.aero

This page intentionally left blank

Document Revisions

REV	DATE	Author(s)	Modifications	Approved
Draft	12/18/10	DTW	Outline for first draft	N/A
Draft	12/24/10	DTW	Collected excerpts from Executive Summaries	N/A
Draft	12/28/10	DTW	Collected excerpts from Executive Summaries	N/A
Draft	12/29/10	DTW	Collected excerpts from other documents	N/A
Draft	12/30/10	DTW	Collected excerpts from other documents	N/A
Draft	12/31/10	DTW	Collected excerpts from other documents	N/A
Draft	1/2/11	DTW	Restructure Sections 2 and 3	N/A
Draft	1/2/11	DTW	Fill out Section 3	N/A
Draft	1/18/11	DTW	Collected excerpts from Use Case Results -009	N/A
Draft	1/21/11	DTW	Collected excerpts from Use Case Results -010	N/A
Draft	1/22/11	DTW	Started Executive Summary; edited content.	N/A
Draft	1/24/11	DTW	Inserted content from GCE 08-001 draft; edited.	N/A
Draft	1/25/11	DTW	Inserted draft paragraphs 3.3 and 3.4	N/A
Draft	1/26/11	DTW	Inserted draft of several paragraphs 6, 8, etc	N/A
Draft	1/27/11	DTW	Normalized prefatory pages, figure numbers, tables	N/A
Draft	1/28/11	DTW	Fixed page, figure, table numbers	N/A
Draft	1/29/11	DTW	Added corrections in Safety Assessment per Lutz	N/A
Draft	2/1/11	DTW	Edits in response to PMC inputs – reduced size	N/A
Draft	2/5/11	DTW	Edits in response to PMC inputs – reduced size	N/A
Draft	2/7/11	DTW	Edits in response to PMC inputs – revised Roadmap	N/A
Draft	2/8/11	DTW	Edits in response to PMC inputs – repaginate	N/A
Draft	2/10/11	KSA	Corrected errors in Tasking and reworded Section 3	N/A
Draft	2/14/11	KPW	Editorial changes	N/A
Draft	3/16/11	GMP	Editorial changes – Fit Use Case primarily	N/A
Draft	3/17/11	DTW	Revised “sunflower” chart – Model Data Exchange Layer	N/A
Draft	5/2/11	DTW	Revised results – John Strasburger (FAA) -> PHF	N/A
Draft	5/10/15	DTW	Miscellaneous revisions	N/A
Draft	5/15/15	MSK	Miscellaneous revisions by DTW, JJC and MSK	N/A
1.0	5/31/15	PMC	Approved for public release	PMC

Outline

Contents

List of Figures

List of Tables

EXECUTIVE SUMMARY

The objectives of this phase of the SAVI development effort are spelled out in ten task statements [2] focused on broadening and structuring the feasibility work completed under AFE 58 [1]. The effort centered on four groups of Use Cases designed to demonstrate new features not addressed during AFE 58's initial conceptual demonstration. Specifically, these four sets of Use Cases were to address (1) integrating spatial, location, thermal, and other physical parameters into the architectural model (often called 'fit' issues); (2) incorporating parameters that allowed assessment of reliability predictions into the architectural model; (3) integrating artifacts useful in a system safety analysis into the upper level model; and (4) demonstrating the ability to feedback analyses (both static and dynamic) that describe the behavior of electromechanical and structural components of a design. These Use Cases are just the beginnings for a large number of such structured demonstrations that SAVI must examine and illustrate capability to handle before the SAVI Virtual Integration Process (VIP) is mature enough to facilitate integration of a system like a commercial or military aircraft. Another of the tasks attempted under this phase of the effort was a broader look at the totality of examples of this type that need to be addressed and in what priority order. This task, aimed at SAVI Version 1.0 and beyond, developed an initial listing of this larger set of Use Cases, but the prioritization planned for this larger set of Use Cases was not finished due to resource constraints.

Results from each of these Use Case exercises are described in this report. All four demonstrations were successfully completed. The conclusion that SAVI is both feasible and desirable (from AFE 58) was reinforced in each Use Case. The "Fit" Use Case showed that geometry and physical descriptions can be passed back and forth with relative ease between physical layout tools (MCAD and ECAD data like that from CATIA, for example) and analysis tools (MATLAB/Simulink) by taking advantage of the extensibility of AADL with straightforward (though tedious) script programs.

Similarly, the Reliability Assessment Use Case showed that use of MOBIUS and ADAPT-M tools allowed and facilitated passing reliability information between those tools and the AADL architectural model. Different deployment configurations of an AFE 58-generated Flight Guidance System (FGS) with dual redundant Autopilot (AP) were exercised to simulate failure probabilities and their effect on system mean time to failure (MTTF). Again, the SAVI VIP was quite capable of handling these kinds of data streams and giving the systems analyst ready insight into effects of changes to system components on reliability. Also, as was expected, there were minor alterations in the AADL Error Annex (a relatively new feature in the current SAVI Architectural Definition Language, AADL). But these modifications were carried out with little interruption in the effort. They were nothing more than the expected insights to be derived from exercising structured Use Cases during the development of the VIP.

The Safety Assessment Use Case used many of the same tools as the Reliability Assessment Use Case but addressed two system safety elements – the Functional Hazard Analysis (FHA) and the Failure Modes and Effects Analysis (FMEA) – and the data sets they typically produce. Once again the AADL Error Annex performed its expected role and the required data flow took place, largely with no problems encountered.

Finally, the Behavior Use Case exercised both a structural element (modeled as a FEM network with a tool called LISA) similar to a wing that simulated use of a movable control surface powered by a generic servoelectrohydraulic valve (modeled in MATLAB/Simulink). Both time histories of the structural response to a generic force excitation and rather simple frequency domain parameters were readily passed between the analysis tools and the AADL architectural model.

Summing up, these four sets of Use Cases revealed nothing detrimental to continuing the development of the SAVI VIP. Each one, though, produced indications of how the VIP is likely to drive further development of both AADL and its Annexes and how translation tools and even some of the analysis tools should evolve in support of the VIP. In every instance, the value of the integrated modeling effort appears to be great. In order to verify the accuracy of this qualitative impression, the SAVI project must continue development in more challenging and realistic environments (like pilot or shadow projects) with a view to quantifying gains that SAVI's VIP brings to the development process. That notion leads naturally to further discussion of the RoI prediction tools explored in AFE 58.

A secondary objective was to broaden and expand the return on investment (RoI) work done under AFE 58. Two primary objectives were identified at the kickoff meeting: (1) improve the usability of the prediction tools that are based on COCOMO II, especially for suppliers; and (2) improve upon how hardware affects the RoI estimate (AFE 58 followed the COCOMO II approach and simply used a constant multiplied by the software RoI to estimate how hardware affects a design). The latter task was evaluated and deemed to be outside the resource and time limits of the Working Group (WG). When the RoI WG reevaluated the shortcomings of the AFE 58 RoI estimation results, they concluded that two other tasks were of higher priority than addressing the hardware effects. First, adding a Monte Carlo algorithm to the COCOMO II analysis was suggested by an expert in RoI estimation and second, as work on AFE 59 progressed, it became clear that SAVI development was more likely to follow an incremental path and be slower in evolving to SAVI Version 3.0 than envisioned during AFE 58. So, it was essential to examine the effects of this kind of evolution and how priorities should be set in choosing a path for growing SAVI. The RoI tool needed to include capability to account for how the topics chosen in this slower development process affected RoI. So these two derived tasks took priority over addressing the hardware effects.

This page intentionally left blank.

INTRODUCTION

1.1 *What is the Systems Architecture Virtual Integration (SAVI) Program?*

The SAVI effort is best described in much the same way it was at the beginning of its first fully structured phase under AVSI – AFE 58 [1]. Quoting from the final report for that phase of the effort:

> “The System Architecture Virtual Integration (SAVI) program is a multi-year, multi-million dollar program intended to implement the capability to virtually integrate complex hardware/software systems before designs are committed to physical form. SAVI is a methodology for managing the exponentially increasing complexity of modern aerospace systems. The program is lead by industry and government participants working together through the Aerospace Vehicle Systems Institute (AVSI), a research cooperative of the Texas Engineering Experiment Station (TEES), itself a member of The Texas A&M University System. Membership in AVSI includes: Airbus; BAE Systems; Boeing; the Department of Defense; the Federal Aviation Administration; General Electric; Goodrich Aerospace; Hamilton Sundstrand; Honeywell International; Lockheed Martin; NASA; and Rockwell Collins.
>
> Increasingly complex hardware and software used in critical aerospace systems pose integration problems nearing the limits of complexity effectively handled by the current system acquisition process. Boeing and Airbus data show rapid increases in the size and complexity of software as indicated by the doubling every four years of the onboard software lines of code (SLOC) for their commercial aircraft systems. Individual companies cannot affordably solve these problems but the industry cannot ignore them. Collaborative, industry-wide, and reusable solutions are sorely needed. The SAVI program is the first cooperative, complete, and cost-effective attempt to provide such solutions. Figure 1 illustrates two of the core elements of SAVI, the Model Data Exchange Layer and the Model Repositories, and how they interact with other elements to allow virtual integration of a system of systems.”

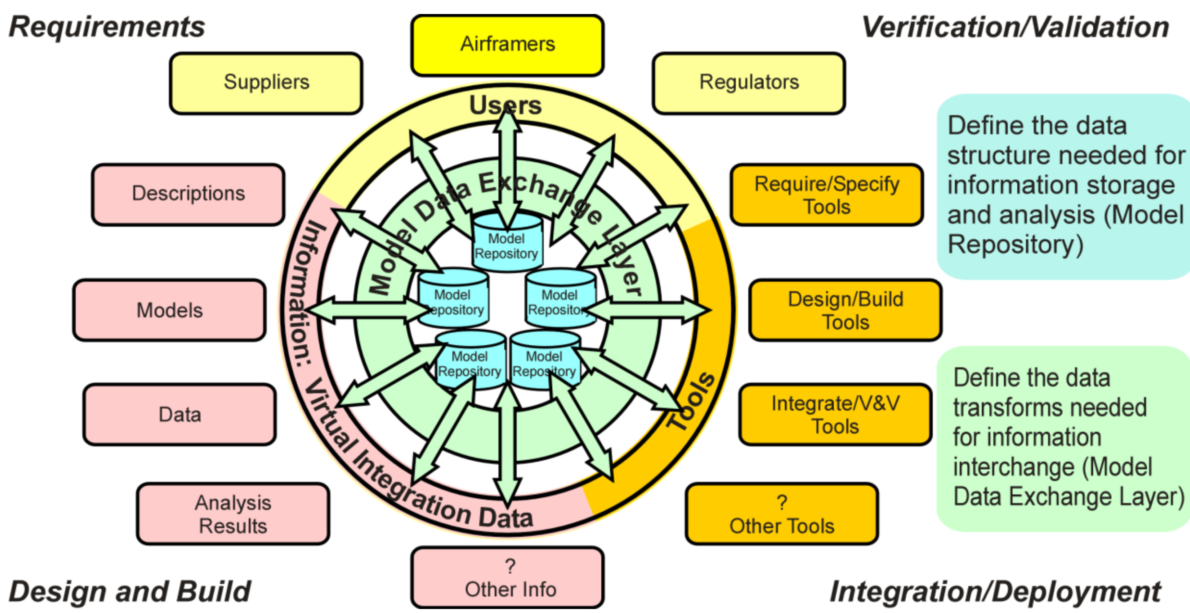

Figure 1 SAVI Core Elements and Interactions

The SAVI Program is developing the definitions for the Model Repository (the data structure needed for information storage and analysis) and the Model Data Exchange Layer (the data transformations needed for information interchange). These components enable multiple business entities to exchange diverse forms of information utilizing multiple tools (both commercial and proprietary). SAVI's model-oriented approach encourages frequent and early system validations (virtual integrations) to help reduce rework, and thereby cycle-time and cost. Tools chains (integration of multiple tools) will enable new and more effective integration checks and analyses.

1.2 SAVI Expanded Proof-of-Concept Demonstration (EPoCD) (AFE #59)

During AFE 58, the SAVI team had largely expected to move directly from that phase into development of SAVI Version 1.0, with a substantial commitment of manpower and money from each participant. However, the resources necessary to step up to that planned larger effort were simply not available during 2009 when this step increase in commitment was considered by the Cooperative leadership. Instead, the PMC was advised by the AVSI Executive Board to structure a smaller effort designed to lend further credibility to SAVI while continuing to make progress toward the final goal of a model-based, architecture-centric paradigm shift in this system engineering approach to system integration. A new AFE 59, called the Expanded Proof-of-Concept Demonstration, became the vehicle for continuing SAVI development. This report summarizes the results of that research effort during 2010.

1.2.1 Shift in Emphasis. The emphasis in AFE 59 was on demonstrating an improved Model Repository and Model Data Exchange Layer in the context of the overall SAVI's Virtual Integration Process (VIP) along with some of the supporting technologies. Use Cases, structured narratives or sequences of actions that describe how elements of the process are to be used or how procedures are to be performed, were taken to be the unifying element bringing these broader demonstrations together. These Use Cases were designed to illustrate how architectural models can be utilized to both pass information between and relate (quantitatively) all levels of the system hierarchy while seamlessly supporting analyses at each level.

Physical system modeling (modeling that concentrates on cyber-physical systems, mechatronic systems, mechanical, electrical, and similar elements of the system) received a larger emphasis during this stage of SAVI development than in AFE 58. The goal was to "convincingly illustrate that the SAVI approach can be used to advantage by airframers, integrators, suppliers, end-users, or regulators…" [2].

1.2.2 EPoCD Objectives. The EPoCD objectives are spelled out in considerable detail in the governing document [2]. They are summarized under ten task statements below.

Task 1: Collect and Prioritize Early Integration Use Cases

This task collects and prioritizes early integration use cases from each member company. These use cases will form the initial requirements set for SAVI 1.0. A subset of these requirements will be used to define and test the extensions to the Model Data Exchange Layer and Model Repository.

Task 2: Identify and Prioritize Gaps in the Model Data Exchange Layer and Model Repository

This task documents and prioritizes the work required in evolving the SAVI environment.

Task 3: Develop and Implement Model Data Exchange Layer and Model Repository Improvements from Task 2

This task develops and implements solutions for the improvements identified in Task 2. Depending on the nature of the improvement being examined, the group working this task may recode portions of supporting software, look for changes in how hardware is modeled, solicit and contract for research to develop new algorithms or modeling tools, alter specifications of previously developed tools or standards, or simply defer seeking a solution to a later time in the SAVI development cycle.

Task 4: Evaluate ADL/IDL and Multi-Language-Model Approaches Best Suited for SAVI Development

This task addresses whether AADL is an appropriate starting point of an ADL/IDL for the long-range development of the SAVI process, and whether a multi-language-model is a better approach for the SAVI repository.

Task 5: Identify and Prioritize Gaps in the AFE 58 RoI Analysis Framework

This task documents and prioritizes steps to improve the RoI analysis framework done for AFE 58.

Task 6: Implement Feasible Upgrades to the RoI Analysis Framework

This task is the action step for improving the RoI analysis framework. Based on the prioritization established in Task 5, the RoI working group will implement improvements that are feasible within the resource and time constraints of AFE 59. Any discussion in the final report of desirable improvements deferred to a more appropriate time in the SAVI evolution will include the group's best estimate of relative importance to the SAVI effort.

Task 7: Identify Potential SAVI Participants within the Aerospace Industry

This task is intended to attract aircraft manufacturers, suppliers to these manufacturers, software tool vendors, regulatory agencies, and expert contractors to the SAVI project. The current active participation in the SAVI project includes three aircraft manufacturers, two (four nominally) suppliers, two observers from governmental agencies, and one federally-funded research organization. This base of support is less than half the size needed to pursue the overall goals and reach the potential returns on investment in SAVI; consequently, the project will focus some of its limited resources on attracting active participation.

Task 8: Forecast and Track Progress in Achieving Broader SAVI Participation

This task sets down goals in terms of increased participation during the course of AFE 59. Specific numerical goals will be spelled out and reported to SAVI participants on a monthly basis. This action builds on the communications list identified as a deliverable under Task 7a but updates this list with the results from these efforts. A tracking section to ascertain whether or not the communications efforts are having the desired effect will be added to the communications data base and maintained by the SAVI PM. These results will be updated at least every two months and will be available for use at PMC meetings.

Task 9: Write AFE 59 Final Report and Supporting Documents

This task includes writing, revising, and approving a summary final report along with supporting documents deemed necessary by the PMC.

Task 10: Manage the AFE 59 Project and Put the SAVI Integrated Program Plan in Place

This task includes all project management tasks deemed necessary by the PMC.

REFERENCES

Note: All SAVI AFE 58 documents (numbered and titled) are found on the SAVI AFE 59 Sharepoint web site at:
System Architecture Virtual Integration > AFE 59 - Expanded PoC Demonstration > AFE 58 Documents
All SAVI AFE 59 documents are found on the SAVI AFE 59 Sharepoint web site at:
System Architecture Virtual Integration > AFE 59 - Expanded PoC Demonstration > AFE 59 Documents

[1]	SAVI-AFE58-00-001	All PMC members, AFE #58 System Architecture Virtual Integration Summary Final Report, October 8, 2009.
[2]	Authority for Expenditure	AFE #59, “Expanded Proof-of-Concept Demonstration,” AVSI, February 16, 2010.
[3]	SAVI-AFE59-08-001	Redman, D. A., “GCE Results,” January 18, 2011.
[4]	SAVI-AFE59-03-001	Pollari, G. M., Appleby, K. S., Lawrence, W. L., and Sharma, S., “Demonstration Plan for Use Case ‘Perform Fit Analyses’,” October 19, 2010.
[5]	SAVI-AFE59-03-008	Lawrence, W. E., Pollari, G. M., and Seah, K. A., “Results for Use Case ‘Perform ‘Fit’ Analyses’,” December 20, 2010.
[6]	SAVI-AFE59-03-005	Feiler, P. H., Lewis, B. W., Woodham, K., Wrage, L., “Demonstration Plan for Use Case ‘Perform Reliability Analyses’,” October 7, 2010.
[7]	SAVI-AFE59-03-009	Feiler, P. H. and Wrage, L., “Results for Use Case ‘Perform Reliability Assessment’,” May 2, 2011.
[8]	SAVI-AFE59-03-006	Feiler, P. H., Lewis, B. W., Woodham, K., Wrage, L., “Demonstration Plan for Use Case ‘Perform Safety Analyses’,” October 7, 2010.
[9]	SAVI-AFE59-03-010	Feiler, P. H. and Wrage, L., “Results for Use Case ‘Perform Safety Assessment’,” January 18, 2011.
[10]	SAVI-AFE59-03-004	Appleby, K. S., Sharma, S., and Ward, D. T., “Demonstration Plan for Use Case ‘Perform Behavior Analyses’,” December 17, 2010.
[11]	SAVI-AFE59-03-007	Appleby, K. S., Sharma, S., and Ward, D. T., “Results of Exercising Use Case ‘Perform Behavior Analyses’,” December 20, 2010.
[12]	SAVI-AFE59-03-013	Appleby, K. S., “FEMplugin-Requirements,” December 21, 2010.
[13]	SAVI-AFE59-03-014	Appleby, K. S., “FEMplugin-UseCaseAnalysis,” December 21, 2010.
[14]	SAVI-AFE59-03-015	Appleby, K. S., “FEMplugin-SoftwareDesign,” December 21, 2010.
[15]	SAVI-AFE59-03-016	Appleby, K. S., “FEMplugin-SoftwareComponents,” December 21, 2010.
[16]	SAVI-AFE59-06-003	Helton, S. B., Pollari, G. M., and Ward, D. T., “User’s Guide for SAVI RoI Estimation Tool (SRET),” December 24, 2010.
[17]		“COCOMO II Model Definition Manual,” downloaded December 23, 2010, from: ftp://ftp.usc.edu/pub/soft_engineering/COCOMO II/cocomo99.0/modelman.pdf
[18]		Lum, K. and Monson, E. “Software Cost Analysis Tool User Document, Version 0,” JPL D-26304, June 11, 2003.

[19] SAVI-AFE59-06-002 Helton, S. B., Pollari, G. M., and Ward, D. T., “SAVI RoI Estimation Tool (SRET),” October 30, 2010.

[20] SAVI-AFE59-10-001 Ward, D. T., “SAVI Integrated Process Plan,” September 30, 2010.

[21] SAVI-AFE59-01-101 Appleby, K. S., “SAVI_UseCases,” Version 0.52, October 1, 2010.

STRUCTURE OF EPoCD DEMONSTRATION EFFORT

The project was organized to begin the period of performance operating with the entire Program Management Committee (PMC) operating as a single group initially. Once the primary Use Cases were defined and semblance of priority established thereby, the PMC broke up into four individual Working Groups (WGs) as suggested in Figure 2. This chart also lays out the working relationships between the Working Groups and identifies which tasks were the responsibilities of each WG. Not explicitly stated, but a major driver of this structure was the priority attached to getting the Use Cases exercised (carried out by Demonstration WG). Because the effort was hampered throughout by lack of manpower to carry out many of the tasks, the PMC was forced to put the DEMO work ahead of the some of the other groups' efforts. The Use Case, Language, Analysis, and Compatibility (UCLAC) WG gave up manpower to these higher priority tasks when the DEMO work fell behind schedule. In late November, the PMC essentially came back together and pooled its expertise to complete the necessary documentation of the work completed during this phase of development. This part of the work is indicated under the Report Preparation and Review (RPR) WG on the right side of Figure 2.

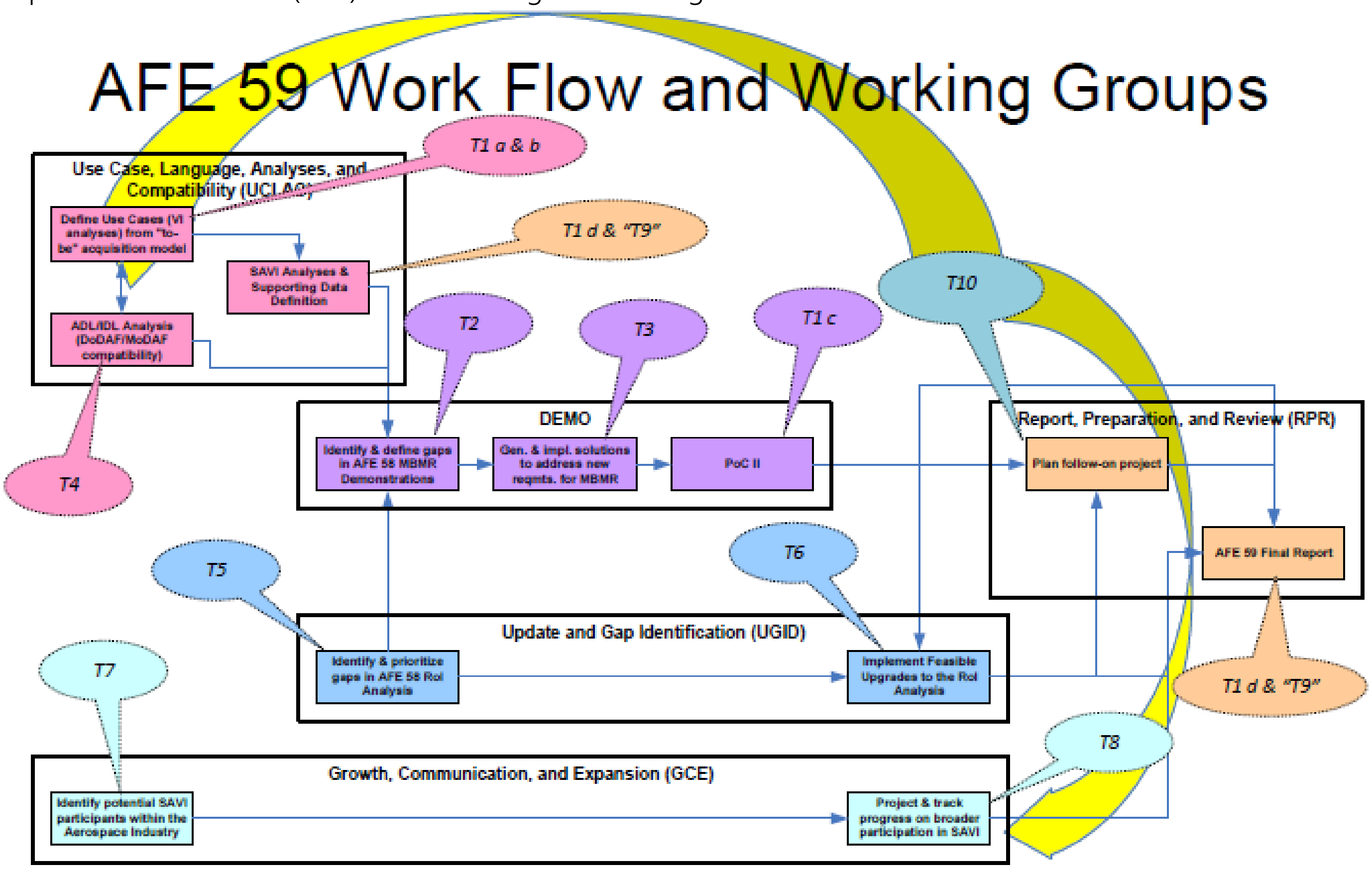

Figure 2 SAVI EPoCD Work Flow and Structure

3.1 Program Management Committee (PMC) Kickoff Meeting

The project was initialized with a three-day meeting hosted by Airbus in Toulouse in March 2010. This meeting set the priorities for this phase of the project and settled on four Use Cases meant to demonstrate the four most meaningful additions to the credibility of AFE 58 Proof-of-Concept effort. These included:

- "Fit" Analyses Use Case – intended to demonstrate a small number of examples that illustrate how the various interfaces (spatial, electrical, hydraulic, and all types of transport elements) can be analyzed and the interactions matched (or mismatches at least discovered) through the use of the SAVI VIP. The "Fit" Analyses subgroup from the DEMO WG addressed a subset of Use Cases "…focused on the use electronic assemblies and their interconnection and packaging…" because the electronics "fit" issues are of primary concern to a number of SAVI participants. These Use Cases were considered one of the higher priority items by a majority of the PMC, though a formal ranking of priorities was not completed.
- Reliability Assessment Use Case – intended to demonstrate how reliability predictions can be integrated into the SAVI VIP. The system (subsystem or component) (model) is exercised and/or analyzed to formally verify it achieves the required reliability criteria.
- Safety Assessment Use Case – intended to illustrate how elements of the system safety assessment can be integrated into the SAVI VIP. Ultimately, the VIP must facilitate a safety analysis/assessment at any level of abstraction (system, subsystem, component, etc.) for the system, a subsystem, or a component.
- Behavior Analyses Use Case – intended to demonstrate how representative analyses of an aircraft system's behavior can be captured and how specific responses affect architectural features of upper levels of the system. Functional behavior analyses are performed on the system (subsystem or component) model(s). The common element to "behavior" analyses is that they are time dependent.

3.2 Growth, Communications, and Expansion (GCE) Working Group

The Growth, Communications, and Expansion Working Group (GCE WG) addressed AFE 59's tasks 7 and 8 (summarized in section 1.2.2). The effort focused on identifying targeted organizations in the classes of participants identified in the OBS and then pursuing opportunities to engage additional participants.

3.2.1. Project Organizational Breakdown Structure. The primary input from AFE 58 [1] to the GCE-WG work under AFE 59 [2] was a matrix of suggested new participants sought to complete the work spelled out in the proposed tasks. The desired types of participants/skill sets sought are summarized in Table 1 [3].

Table 1 Partners/Skill Sets Sought -- Quantitative Objectives [3]

Type	Current	Sought	Total	Comments	Example Organizations
Airframers/Platform Integrators	3	1	4	Small/business jets representation	Airbus, Boeing, Dassault-Aviation, Lockheed-Martin, EMBRAER, Bombardier
Systems Suppliers	3	3	6	More European representation	BAE Systems, GE Aerospace, Honeywell, Rockwell-Collins, Safran, Thales
Engine Suppliers	0	1	1		Rolls-Royce, GE, Pratt & Whitney
Tools Vendors	0	3	3	Safety analysis, functional behavior, requirements management, tools suites	Dassault Systèmes, Esterel Technologies, The Mathworks
IS/IT Integrators	0	1	1		IBM
Labs & academics	1	2	2	European participation	CMU/SEI
Authorities	2	1	3	European participation	DoD, EASA, FAA
Total	9	12	21		

BOLD indicates AFE 58 participant

3.2.2. GCE Efforts during AFE 59. The most important results of the GCE WG effort are intangible; there is an excellent chance that the tangible effect of these efforts will not be visible for some time after the end of this phase of the project. At this writing, two organizations have been added to the SAVI team. Honeywell (a former team member) and EMBRAER are joining for the next phase of the SAVI effort and were participants in the Kickoff meeting for AFE 59 Supplement 1, as were one other former participant and one AVSI member who has never been a part of SAVI. The GCE WG efforts are likely to produce even more positive results during the next phase of the effort, but these efforts must continue and even grow to achieve the levels targeted in Table 1.

3.3 Demonstration (DEMO) Working Group

The Demonstration Working Group (DEMO WG) addressed AFE 59's tasks 2 and 3 (summarized in section 1.2.2). During the Kickoff meeting in Toulouse a set of tasks was discussed and agreed upon by the entire PMC. Those discussions ultimately led to a list of over 250 Use Cases at all levels within the system hierarchy, along with delegation to the DEMO WG and the UCLAC WG authority to select a small, workable number of Use Cases to demonstrate the improvements sought for the SAVI process during AFE 59. The four most important groups of tasks (already described in section 3.1) pointed to Use Cases that: (1) supported integrating physical "fit" (mechanical, volumetric, temperature, etc.) properties of subsystems, (2) facilitated reliability predictions at all levels of the architecture, (3) permitted system safety assessments at all levels of the system, and (4) brought together electrohydraulic, flight control, and structural subsystems analyses. These latter subsystems interfaced through the relatively new AADL Behavior Annex while Use Case groups 2 and 3 brought out the interactions that the AADL Error Annex was meant to address. In short this phase of the effort was structured to show the usefulness of the SAVI VIP to mechanical and physical components along with the software elements that the architectural model had successfully handled during the initial feasibility demonstration under AFE 58.

3.4 Use Case, Language, Analysis, and Compatibility (UCLAC) Working Group

The Use Case, Language, Analysis, and Compatibility Working Group (UCLAC WG) addressed AFE 59's tasks 1a, 1b, and 4 (summarized in section 1.2.2). This WG was concerned with taking the directions suggested by the Use Case listings that were derived from the framework agreed upon during the Kickoff meeting in Toulouse and turning them into a prioritized set of activities. This initial prioritization led to the four sets of Use Cases described in Section 3.3 that were completed during AFE 59. Beyond this initial prioritization, the goal was to set down initial requirements (and a prioritized set of Use Cases supporting them) for SAVI Version 1.0, the prototype process meant to implement the SAVI Virtual Integration Process.

This group was primarily concerned with longer term objectives and strategies for SAVI. Task 1 called for the documentation and prioritization of Use Cases and identifying gaps in the SAVI process. Task 4 called for an assessment of ADLs/IDLs and multi-language/multi-model approaches. Unfortunately, many project members were unable to dedicate the resource hours required to complete (in the case of Task 1) or undertake (in the case of Task 4) the defined tasks. These tasks will have to be addressed, hopefully within the next round of development.

3.5 Update and Gap Identification (UGID) Working Group

The Update and Gap Identification Working Group (UGID WG) addressed AFE 59's tasks 5 and 6 (summarized in section 1.2.2). These tasks zero in on the Return on Investment (RoI) estimation methodology started under AFE

58. These AFE 59 tasks were intended to expand the RoI technique, based on Boehm's COCOMO II spreadsheet, to be more useful to and more usable by all members of the SAVI community. The goals were to expand the capabilities by adding a Monte Carlo prediction algorithm and by adding better prediction of hardware effects. However, the latter proved to be a more time-consuming task than anticipated. A demonstration of usefulness of the prediction tool to suppliers was completed along with a User's Guide to explain the spreadsheet. Finally, a first iteration capability to examine the effect of incremental development on RoI estimates was added.

EPOCD USE CASE DEMONSTRATIONS

The core tasks for AFE 59 are these Use Case demonstrations. They are the activities during which credibility of the SAVI VIP can be captured and documented to provide concrete results. The basic model to which most of the Use Cases refer is shown in Figure 3. It is a modification of a model developed for AFE 58 [1]. This model provides the architectural framework and helps impose system discipline on subsystems integrated through the SAVI VIP with the AADL instantiation of this model.

4.1 "Fit" Use Case

4.1.1. "Fit" Use Case Description and Plan. [4] The "fit" model to be demonstrated is focused on electronic assemblies and their interconnection and packaging. This fit model was chosen to demonstrate improved overall integration of modular electronics in vehicle architecture when SAVI techniques are used. Basic "fit" integration means that signals from individual Printed Wiring Circuit Boards (PWB) to chassis enclosures via Interconnect assembly and wiring harnesses is complete with no mismatches. Logical signal properties, physical signal properties, and mechanical fit properties of all parts must match for a successful integration. The "Fit" Use Case is designed to show how the SAVI VIP carries out, checks, and verifies this matching of all these properties through the use of both the architectural model (in AADL) and analysis tools typically used to design and test such assemblies.

This fit model Demo extends the use of AADL and SAVI to not only help identify the logical errors (that is, signal name matching) but also to help find mechanical mismatches that arise during system development. This modular electronics modeling concept can also be extended to other mechanical systems (hydraulics, for example) by using a similar modeling approach.

The Flight Guidance System physical architecture SY FCP model (Figure 4) has been modified to give more definition below the baseline model for the purposes of the "fit" demonstration. Figure 5 shows architectural detail added to the Flight Guidance System generated during AFE 58. It illustrates this additional level of detail necessary for carrying out the "Fit" Use Case.

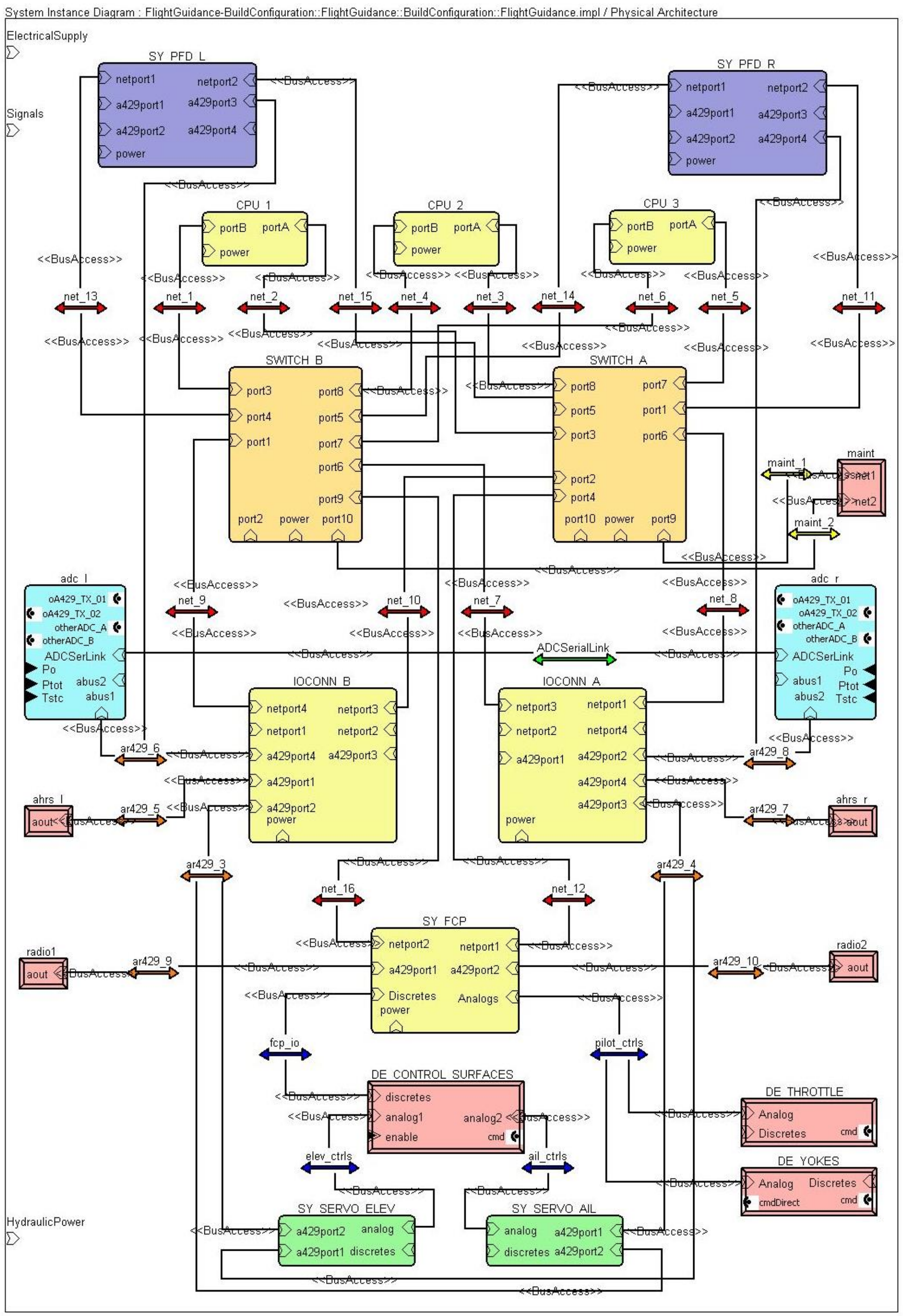

Figure 3. Flight Guidance System - Model Physical Architecture

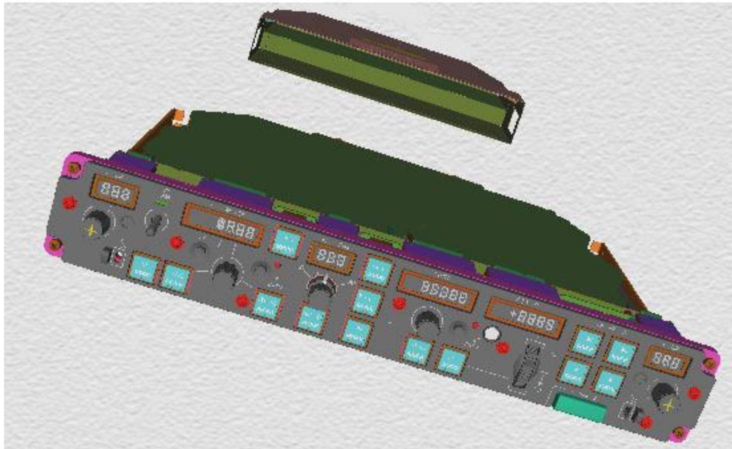

Figure 4 Electronic Assembly Diagram - SY FCP with Detached Housing

4.1.2. Properties Fit Checked for the SY FCP with Detached Housing. [5] The "fit" properties checked included: module heat load/sink compatibility, module volume/chassis volume compatibility, power loading compatibility (modules, chassis), air flow load compatibility (modules to chassis), signal properties compatibility (twisted pair, impedance, gauge, length, signal type, voltage), liquid cooling compatibility (modules to chassis). Due to resource constraints, the analysis for signal names and signal consistency (modules, chassis, wire harness), connector fitment and orientation (module to backplane interconnect) was not completed in this AFE and is deferred to a future AFE.

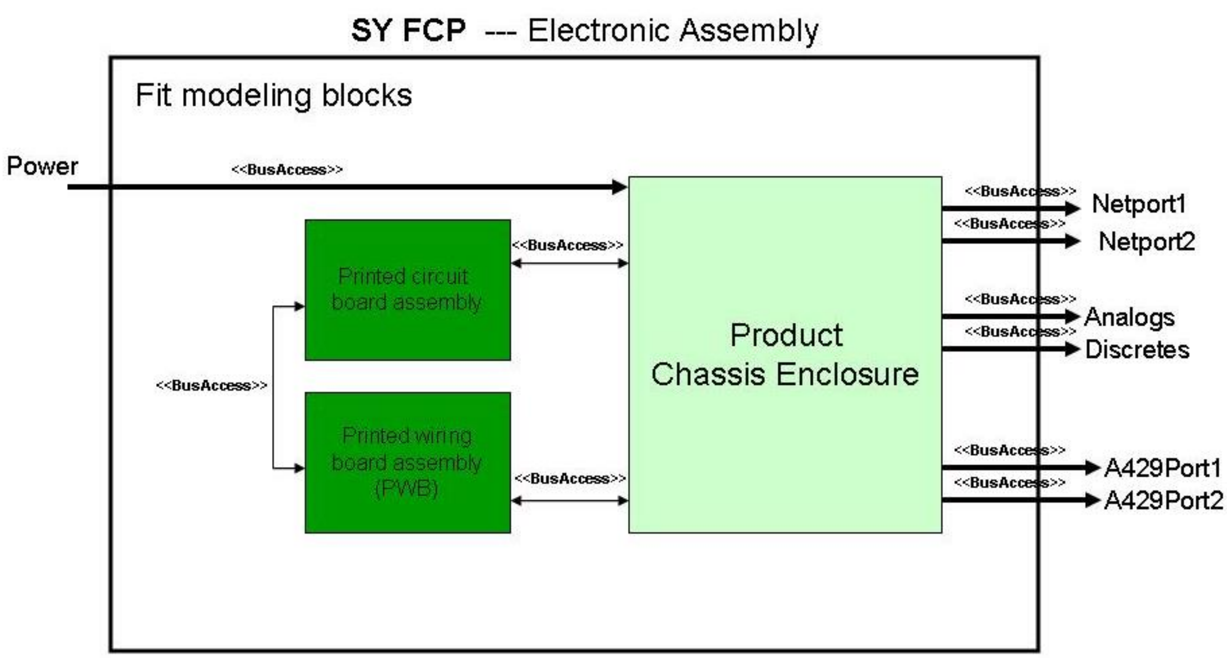

Figure 5. Fit Model Block Diagram

4.1.3. "Fit" Use Case Results. Property extensions were added to the `SAVI.aadl` file containing all the properties defined by SAVI in the AADL property set "SAVI". This file is located in the `SAVICommon` project in the directory `aadl/propertysets`. These new properties were grouped into the broad categories: spatial, location, thermal, air flow, connectors, voltage, impedance, current, conductors, and liquid cooling.

The key questions for this Use Case to answer are given in the following list along with the answers provided by exercising the SAVI model.

1. Can an architecture model of the system and its embedded software be annotated in a standardized fashion with “fit” information to support “single truth” physical and logical integration of the “fit” parameters specified in 4.1.2?
2. Can these data be used effectively to explore interactions between the system of interest (the FGS) and the SY FCP with detached housing and how changes made to facilitate “fit” between such components “ripple through” the rest of the design?
3. What “translators” are needed to make these “fit” data sets convenient and easy to pass between selected design tools and the architectural framework (the “single-truth” model)? Or, do we already have adequate “interfacing” software?
4. What are the “holes” (if any) in the AADL Annexes for fully supporting “fit” integrations at the system and embedded software level through the architectural model? Are there gaps in our ability to exchange data in either direction with reliability assessment tools?
5. Does the current standards addressing “fit” parameters serve to adequately ensure that tools typically used in design produce data formats and data consistency that match what is needed by AADL? Would additional standards or improved standards make this interface more convenient to use?

4.2 Reliability Assessment Use Case

Reliability assessment of physical systems is a well-established practice, while reliability assessment of embedded software is a challenge because physical hardware reliability is primarily driven by wear-and-tear of the physical components, while software failures are design faults. Three aspects of this Use Case scenario were performed under AFE 59. The rationale for only three assessment aspects was driven by the short time frame and by limited resources to carry out the demonstrations. The three aspects addressed in this Use Case were:

- Assessment of the reliability of the Flight Control System (FCS) consisting of the Flight Guidance System (FGS) and the Auto Pilot (AP) in multiple deployment configurations of the computer hardware. This assessment is driven by the reliability of the hardware components, assuming that the software is correct (a common practice). This assessment scenario required confirmation of this assumption by verifying that the redundancy management logic was designed and implemented correctly.
- Assessment of the reliability of the FCS by considering failure rates for embedded software by distinguishing between the flight control software and the redundancy management software. This approach provided insight into the impact of software design faults on FCS reliability.
- Reliability assessment of aircraft flight controls, ensuring the FCS includes its hydromechanical elements.

4.2.1. Reliability Assessment Use Case Description and Plan. [6] This Use Case uses the same FGS shown in Figure 3. Notice that, in the lower part of Figure 3, control surfaces are shown with their electrical and hydraulic power sources indicated. These control surfaces represent an important feature of the architectural model to be demonstrated in this expanded set of demonstrations. The Integrated Modular Avionics (IMA) computer platform consists of three processors (shown in the upper part of Figure 3 with the FGS, AP, and other embedded software subsystems bound to them in several configurations. All three processors are physically connected to both of the switches. These two switches are fully connected to replicates of the Input/Output Connection (IOConn) subsystem, which in turn provide access to the flight control surfaces, throttle, and yokes. The two switches are also connected to the two Primary Flight Displays (PFDs).

Figure 6 illustrates the same system in a "functional architecture" view. In the original demonstration [1], no attempt was made to go beyond showing this functional view of such a mechatronic subsystem. One of the main purposes of the EPoCD expansion was to demonstrate that (1) reliability assessment can be carried out in an analysis tool of choice; (2) design changes to address a shortcoming during subsystem development can be captured in the "single-truth" repository, and (3) the effects of such changes on the system reliability calculated within the original model. Verification that these three "credibility" questions can be answered affirmatively for the VIP is one of the goals of the EPoCD.

Doing a reliability assessment at the system/embedded software level demanded a representation of the system and embedded software architecture in a common notation including fault behavior information. AADL and its Error Annex Standard were leveraged for that purpose. Figure 6 shows the model at the system level with the IMA subsystem expanded and the flight surfaces abstracted. The flight surfaces mechanical aspects were modeled in detail so a reliability assessment in terms of its parts validated the reliability figures used at the system level.

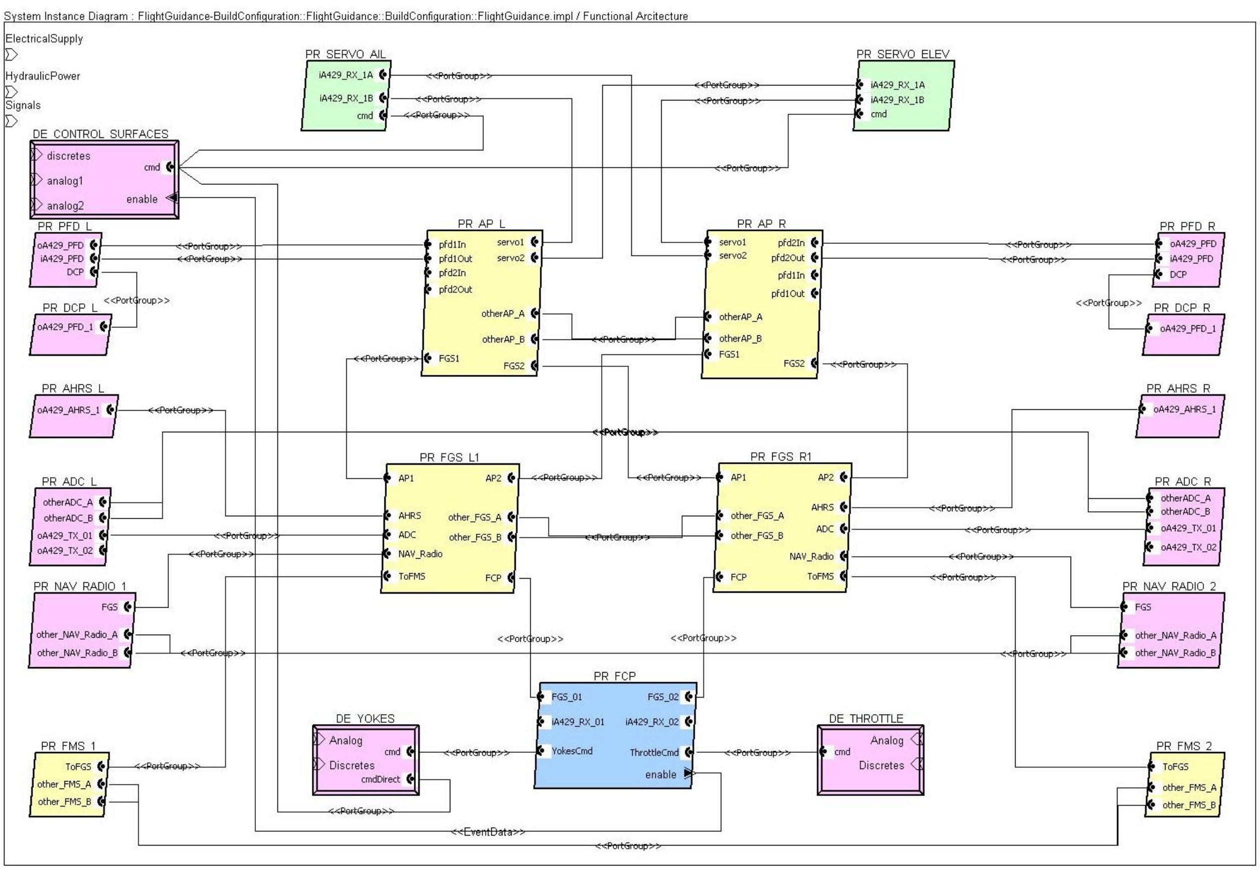

Figure 6. Flight Guidance System - Model Functional Architecture

In addition, embedded software reliability assumptions in the reliability assessment were captured and verified. Even with best assurance practices, such as those specified in DO-178B, we have to assume residual errors.

This demonstration sought answers to the following questions:

1. Can an architecture model of the system and its embedded software be annotated in a standardized fashion with fault behavior information to support "single truth" reliability assessment?
2. Can these data be used effectively to explore effects within the system of interest (the airplane FGS in this case) of changes made to improve the utilization of the computer platform resources by different deployment configurations studied?
3. What "translators" are needed to make these data sets convenient and easy to pass between the reliability assessment tools and the architectural tools (the "single-truth" model)? Or, do we already have adequate "interfacing" software tools (like the ADAPT interface by Rugina (LAAS), and Hecht (Aerospace Corp) or COMPASS/SLIM by Noll (U. Aachen)?
4. What are the "holes" (if any) in the AADL Error Annex for fully supporting reliability assessment at the system and embedded software level through the architectural model? Are there gaps in our ability to exchange data in either direction with reliability assessment tools?

4.2.2. Reliability Assessment Use Case Results. Results consist of both a simulation-based analysis and a stochastic network analysis giving MTBF values for the different FCS subsystem configurations as well as the FCS combined with the flight surfaces. A summary of the results are excerpted below from the WG report [7].

- Hardware-Focused Reliability Use Case Results.

Error events are modeled as error states of normal, fault free, and permanent CPU failure. This latter failure is carried as a probability of failure parameter in the AADL model and is assigned a value in the simulation tool. These error models are associated with the processors and AP and FGS software components in the AADL IMA model. Error annex subclauses were added to the corresponding component declarations. To compare reliability, two different deployment configurations of the dual redundant AP and FGS processes are evaluated. Table 2 shows these two deployment configurations.

Table 2 Deployment Configurations

	3 CPU Deployment			2 CPU Deployment		
Process	CPU 1	CPU 2	CPU 3	CPU 1	CPU 2	CPU 3
PR_AP_L	X			X		
PR_AP_R		X			X	
PR_FGS_L1		X		X		
PR_FGS_R1			X		X	

A simulation of time of 1,000,000 hours and simulation runs were made with different CPU failure probabilities. The results show a linear dependence of system reliability on CPU failure rate, as expected for this simple reliability model. The 2 CPU configuration is more reliable than the 3 CPU configuration. This behavior is explained by the fact that fewer combinations of CPU failures lead to a system failure in the 2 CPU configuration. Tables 3 and 4 summarize the MTTF results for this small set of simulations on these two CPU configurations.

Table 3 MTTF for 3 CPU Deployment – CPU Failures Only

CPU Failure Rate	0	0.00001	0.00002	0.00005	0.0001
One AP Operational	1,000,000	111,969 ±5,565	55,984 ±2,782	22,393 ±1,113	11,196 ±556
Both APs Operational	1,000,000	47,861 ±2,922	23,930 ±1,461	9,772 ±584	4,786 ±292

Table 4 MTTF for 2 CPU Deployment – CPU Failures Only

CPU Failure Rate	0.0	0.00001	0.00002	0.00005	0.0001
One AP Operational	1,000,000	150,992 ±6,873	75,496 ±3,436	30,198 ±1,374	15,099 ±687
Both APs Operational	1,000,000	50,866 ±3,095	25,433 ±1,547	10,173 ±619	5,086 ±309

- Software-Focused Reliability Use Case Results

For results just summarized, software was assumed to be error-free. For the second part of this demonstration, the error model was refined to include a probability of software failure. A CPU failure is no longer the only source of a system failure; software can now fail even though the CPU does not. This slightly extended error model was treated with the same 1,000,000 simulated operating hours and CPU failure rate was fixed at 0.00005 (5 failures per 100,000 hours of operation). Resulting MTTF predictions are shown in Tables 5 and 6.

Table 5 MTTF for 3 CPU Deployment – CPU and SW Failures

Software Failure Rate	0.00001	0.00002	0.00005	0.0001	0.0002
One AP Operational	18,141± 835	14,547 ± 650	10,004 ±438	6,441±270	3,689 ±151
Both APs Operational	8,452 ±488	6,775 ±400	4,531 ±274	2,903 ±167	1,668 ±93

Table 6 MTTF for 2 CPU Deployment – CPU and SW Failures

Software Failure Rate	0.00001	0.00002	0.00005	0.0001	0.0002
One AP Operational	20,949 ±990	16,816 ±773	10,443 ±471	6,573 ±310	3,792 ±166
Both APs Operational	8,323 ±506	6,766 ±404	4,738 ±294	2,972 ±176	1,724 ±103

The next step in considering software failures is to include failures in redundancy management software in the system. So far, perfect redundancy detection and isolation has been assumed. This part of the software can fail either because of a CPU failure or a software defect and is modeled as having a fixed probability for this demonstration. For this new set of simulations the time and the CPU/software failure rate assumptions were the same as before but redundancy failure rate was allowed to vary from 0.01 to 0.2. Tables 7 and 8 spell out the MTTF results from this set of simulation runs.

Table 7 MTTF for 3 CPU Deployment – CPU, SW, and Redundancy Logic Failures

Redundancy Logic Failure Rate	0.0	0.01	0.02	0.05	0.1	0.2
One AP Operational	9,505 ±426	9,433 ±260	9,161 ±408	8,886 ±417	8,323 ±408	7,705 ±406
Both APs Operational	4,450 ±257	4,463 ±260	4,403 ±257	4,395 ±258	4,342 ±259	4,294 ±254

Table 8 MTTF for 2 CPU Deployment – CPU, SW, and Redundancy Logic Failures

Redundancy Logic Failure Rate	0.0	0.01	0.02	0.05	0.1	0.2
One AP Operational	10,294 ±444	10,160 ±435	9,928 ±425	9,449 ±423	9,153 ±444	7,992 ±425
Both APs Operational	4,580 ±262	4,573 ±264	4,535 ±260	4,456 ±251	4,401 ±273	4,203 ±258

❖ Answers to Critical Questions

The critical questions posed on page 22 can now be answered.

1. The reliability assessment conducted in this demonstration required adding information to the PoC model in the form of (a) AADL error model annex standard libraries and subclauses and (b) simulation projects. Error modeling annex information is embedded in the core AADL architecture models, so it became part of the 'single truth' model repository automatically. The reliability simulation consists of a number of files that contain the stochastic activity network, reward, study, and simulation models. From these, the simulation software generates C++ files and compiles them into a binary executable. In this demonstration the stochastic activity network was generated from the AADL error models and others were generated directly in reliability simulation software. Consequently, reward, study, and simulation models must be added to the model repository. Thus additions were easily carried out during this demonstration; adding the needed directories and files to the subversion repository was quite straightforward.
2. This Use Case illustrated the effect of using 2 CPU and 3 CPU hardware configurations and different software deployment configurations on overall system reliability. In addition, it showed the impact of software faults and highlighted a difference between software faults in fault tolerance logic and in application logic on system reliability – an issue often misunderstood and hardly addressed in traditional reliability analyses.
3. In the demonstration the simulation software tools successfully extracted error model information [9] from the AADL model and converted it to a stochastic activity network model, which was analyzed in the reliability simulation software. Use of the prototype tool did have some limitations; for example, transformation of fault propagations along connections was incomplete. However, this limitation was circumvented by adapting the reward model within it.
4. The only limit found in the AADL error annex was the characterization of error free states of the system as a whole. Although the error annex supports derived error models that consolidate error states up the hierarchy, there was no way to mark certain states as error free. Even if ADAPT could handle derived error models, there would still be no automatic mapping of error free states to Mobius variables in the reward model. Eliminating this limitation is already under discussion for the next revision of the AADL error annex.

4.3 Safety Assessment Use Case

Safety assessment of safety-critical systems is a well-established practice with a number of governing documents providing guidance (ARP 4761, ARP 4754, DO-254, DO-178B). Some of these guidance documents are going through revisions to better address advances in software-reliant system development, such as use of formal methods and architecture-centric model-based engineering. The safety assessment activities range from

Functional Hazard Assessment (FHA), Common Cause Analysis (CCA), System Safety Assessment (SSA) to Failure Mode and Effects Analysis (FMEA) and Fault Tree Analysis (FTA). During the development these practices were complemented by formalized specification of safety properties and their validation/verification using techniques such as model checking [8].

4.3.1. Safety Assessment Use Case Description and Plan. Three aspects of this use case scenario were considered under AFE 59 during the summer and fall of 2010, but the short time frame and limited resources to carry out the demonstrations meant only two of the activities above could be addressed:

- Functional Hazard Assessment (FHA) of the flight control aspect of the aircraft with focus on the Flight Control System (FCS). The FCS consists of the Flight Guidance System (FGS) and the Auto Pilot (AP) in several deployment configurations on the computer hardware. By performing FHA at several levels of the system architecture we are able to demonstrate its applicability to software related hazards as well as physical hazards.
- Failure Mode and Effects Analysis (FMEA) of the Flight Control System (FCS) as part of a System Safety Assessment (SSA).

This demonstration answers the following questions:

1. Can an architecture model of the system and its embedded software be annotated in a standardized fashion with fault/hazard information to support "single truth" safety assessments?
2. Can these data be used effectively to support Functional Hazard Assessment (FHA), as well as Failure Mode and Effect Analysis (FMEA) or Fault Tree Analysis (FTA)?
3. What "translators" are needed to make these data sets convenient and easy to pass between the safety assessment tools and the architectural tools (the "single-truth" model)? Or, do we already have adequate "interfacing" software tools?
4. What are the "holes" (if any) in the AADL Error Annex for fully supporting safety assessment at the system and embedded software level through the architectural model? Are there gaps in our ability to exchange data in either direction with safety assessment tools?

Table 9. Functional Hazard Assessment for Flight Guidance Steering Requirement

Reference	Functional Failure (Hazard)	Critical Operational Phase	Aircraft Manifestation	Criticality	Comment
1.1.1	Loss of Guidance Values	Approach	Presence of No Computed Data (NCD) should signal FD and AP disconnect	Minor	Becomes major hazard, equivalent to incorrect guidance, if disconnect fails.
1.1.2	Incorrect Guidance Values	Approach	Gradual departure from references until detected by flight crew during check of primary flight data resulting in manual disconnect and manual flying.	Major	No difference to the AP between loss of guidance and incorrect guidance.

Both safety analyses addressed, the FHA and the SSA, were supported by annotating fault information in the AADL model with additional properties capturing information relevant to this Use Case.

As noted in section 4.3.1, this Use Case considered only two types of safety analyses: (1) a FHA focused on the Flight Control System (FCS). The FCS consists of the Flight Guidance System (FGS) and the Auto Pilot (AP) in several deployment configurations on the computer hardware. Performing an FHA at several levels of the system architecture allowed demonstration of software-related hazards as well as physical hazards; and (2) a FMEA of the FCS as part of a SSA. This Use Case used the Error Model Annex introduced in section 4.2.2, as well as the same reliability prediction software tool. Hence, many of the procedures are similar to those discussed under the Reliability Assessment.

4.3.2. Safety Assessment Use Case Results. Summary results are excerpted in this section from the WG report [9]. The same hazards were present for both copies of the dual redundant FGS component, so the same error model was defined for both, using the model property of the AADL Error Annex. In addition to the hazard property for fault propagations in the Error Annex, a new property, doFHA, was defined in a property set Safety (Figure 7). The component was essential in hazard information export. Only the right FGS process was included in the FHA data export because the left FGS process gave the same results.

Safety.aadl

```
property set Safety is
  -- properties
  doFHA: aadlboolean => false applies to (system, process);
end Safety;
```

Figure 7 Property Set *Safety*

Finally, a new OSATE plugin was created to export hazard data in csv format for all components in a model whose property *Safety::doFHA* is set to *true*. Table 10 shows FHA data exported in a spreadsheet format.

Table 10 Partial FHA Output for the FGS Process

Component	Error	Refe-rence	Functional Failure (Hazard)	Critical Operational Phase	Aircraft Manifestation	Criti-cality	Comment
FlightGuidance_Flight_subsystems_Instance.PR_FGS_L1	Guidance_Loss	1.1.1	Loss of guidance values	approach	Presence of no computed data should signal FD and AP disconnnect	minor	Becomes major hazard, equivalent to incorrect guidance, if disconnect fails.
	Guidance_Incorrect	1.1.2	Incorrect guidance values	approach	Gradual departure from references until detected by flight crew during check of primary flight data resulting in manual disconnect and manual flying	major	No difference to the AP between loss of guidance and incorrect guidance
	Transfer_Control_Loss	4.1.1	Loss of transfer control of flight guidance to AP	all	Flight crew unable to change 'Pilot Flying' side FGS. Manual disconnect and manual flying	minor	
	Inad-vertent_Transfer	4.1.2	Inadvertent transfer of flight guidance data to AP	approach	Possible gradual departure from references until detected by flight crew during check of primary flight data resulting in manual disconnect and manual flying	minor	
	Transfer_State_Loss	4.2.1	Loss of flight guidance transfer state	all	Flight crew unable to determine 'Pilot Flying' side. Manual disconnect and manual flying	minor	
	Transfer_State_Incorrct	4.2.2	Incorrect indication of flight guidance transfer state	all	Incorrect 'Pilot Flying' side indicated. Possible gradual departure references until detected by flight crew during check of primary flight data resulting in manual disconnect and manual flying	major	Departure from references occurs only if pilot flying and pilot not flying have selected navigation sources.

4.3.3. Failure Modes and Effects Analysis (FMEA) Use Case 1.3.4C. The same error models created for reliability analyses [7] were used for the FMEA. To simplify the demonstration, the FMEA error models were different from the error models used in the FHA. For the FMEA, only complete failure of a software component was considered, while the FHA included functional errors, e.g., incorrect guidance values. To conduct the FMEA, initial failure mode information was added. A connecting software tool, which gave the FMEA capability, had an extended error annex that added, among other features, a flag to error events. The FMEA was executed with two different deployment configurations, one with AP and FGS processes deployed on three CPUs and another with these processes deployed on two CPUs (as in Table 2). For both scenarios the model was analyzed with and without failures of the redundancy logic.

The FMEA result was presented in an MS-Excel spreadsheet that listed the components, failure modes, and effects. The results for the scenario without redundancy logic failures were as expected. However, the current version of the conversion tool had a bug when fault propagations along AADL connections were analyzed. This feature propagated redundancy logic failures between software components, so results for this scenario were incomplete. The automated full FMEA had many levels of effects, but without redundancy errors only first and second levels were observed. With redundancy errors, the tool calculated four levels of effects, more than can be handled manually.

Advantages of an automated FMEA included showing higher level effects, analyzing multiple failures, and quickly reassess a large number of effects (Tables 11 and 12) when the model was changed.

Table 11 FMEA Output Example, HW Failures Only (partial) – Three CPU FMEA

Item	Initial State	Initial Failure	1st Level Effect	Transition	2nd Level Effect
CPU_1.cpu	ErrorFree	CPU_Failure	Permanent Effect	out_CPU_Failed	Permanent Error
PR_AP_L	ErrorFree		ErrorFree	CPU_Failed(N)	Permanent Error
CPU_2.cpu	ErrorFree		ErrorFree		ErrorFree
PR_AP_R	ErrorFree		ErrorFree		ErrorFree
PR_FGS_L1	ErrorFree		ErrorFree		ErrorFree
CPU_3.cpu	ErrorFree		ErrorFree		ErrorFree
PR_FGS_R1	ErrorFree		ErrorFree		ErrorFree
CPU_1.cpu	ErrorFree		ErrorFree		ErrorFree
PR_AP_L	ErrorFree		ErrorFree		ErrorFree
CPU_2.cpu	ErrorFree	CPU_Failure	Permanent Error	out_CPU_Failed	Permanent Error
PR_AP_R	ErrorFree		ErrorFree	CPU_Failed(N)	Permanent Error
PR_FGS_L1	ErrorFree		ErrorFree	CPU_Failed(N)	Permanent Error
CPU_3.cpu	ErrorFree		ErrorFree		ErrorFree
PR_FGS_R1	ErrorFree		ErrorFree		ErrorFree

Table 12 Number of Analyzed Effects

	AP and FGS Deployed to 2 CPUs	AP and FGS Deployed to 3 CPUs
No redundancy mechanism failures	185	162
With redundancy mechanism failures	Not analyzed	1512

4.3.4. *Answers to Critical Questions.* The critical questions posed on page 16 can now be answered:

1. Given the implemented extensions to the AADL error annex standard, all information needed for the FHA and FMEA can be represented in error annex libraries and subclauses. As such, it automatically becomes part of the AADL model in the model repository. However, the AADL error annex had to be extended to capture the additional information (see also the answer to question 4 below).
2. This demonstration has shown that the AADL error annex provides a solid basis for FHA and FMEA; needed extensions were easy to implement.
3. For the FHA a new OSATE plugin was implemented to extract error model information into a spreadsheet format. This plugin is based on a plugin created under the original PoC in AFE58 to extract AADL property values. Reuse of existing code was limited by the fact that error model properties in AADL are different from properties in the core language.

In this FHA demonstration the ADAPT-M tool supplied by Aerospace Corp successfully extracted error model information from the AADL model and it was used to conduct an FMEA. As the ADAPT-M tool is a prototype release, there were some limitations. Specifically, transformation of fault propagations along connections was incomplete. However, adapting the reward model in Mobius provided a work around to this limitation. Another limitation results since Rugina's ADAPT tool does not handle derived error models at all. This constraint makes it necessary to define reward models directly in Mobius; they cannot always be generated from the AADL error annex information.

4. Both FHA and FMEA need extensions to the AADL error annex. For the FHA an additional property was needed to store additional information, e.g., criticality, description, etc., about the hazards. In addition, although out error propagations can represent hazards, there is an interaction between error propagations and out guards in the error model that needs to be explored further. For the FMEA the ADAPT-M tool already relies on additional flags in the error model, e.g., to indicate which error events represent initial failure modes and which error states are in fact error-free states.

A revision of the AADL error annex is under way to integrate similar extensions into the standard. Also, AADL version 2 supports the use of the full AADL property mechanism in annexes. Therefore, an extension such as the hazard property for FHA, was defined in a property set with no implementation work for its realization.

4.4 Behavior Analysis Use Case

The AFE 59 PMC placed a high priority on conducting a Behavior Analysis Use Case. However, resource limitations forced the PMC to prioritize the elements [10]. Two assumptions drove the choice of which branch of behavioral analysis to pursue: (1) the demonstration needed to capture at least some of the characteristics of a mechatronic subsystem (like the Flight Control System on an airplane that has hydraulic, electrical, mechanical, and software components), and (2) it must focus on the supplier's needs (design iterations of components, effect of component redesign, and the like), though the effects on the next layer up (the integrated aircraft system) in the system hierarchy should also be captured in the exercise results.

The Behavior Analysis Use Cases were decomposed into a hierarchical tree structure [11] but only the following Use Case subset of this decomposition was demonstrated: (UC.1.4.2.1) Perform Time Step Simulation and (UC.1.4.2.3) Perform Aeroelastic Evaluations.

This set was a workable group of Use Cases for the available time and resources, illustrating the major aspects of the overarching Behavior Analysis Use Case. Other Use Cases were to be addressed and prioritized as part of the Use Case, Language, Analysis, and Compatibility (UCLAC) Working Group effort in Section 6.

4.4.1. Behavior Analyses Use Case Description and Plan. [10] Figures 3 and 6 show notional control surfaces for a FCS. Control surfaces are an important feature of the architectural model and their dynamical features must be captured in the architectural model. Showing that these dynamic interactions within the structural, hydraulic, mechanical, and electrical subsystems, can be captured and passed in both directions between the hierarchical layers of the system design were a crucial demonstration for this expanded set of demonstrations. In the AFE 58 demonstration [1], no attempt was made to go beyond showing this functional view of such a mechatronic subsystem. The purposes of this EPoCD expansion were to answer the following important questions:

1. Can typical analysis tool models provide consistent data from trade studies conducted within the electrohydromechanical analysis domain for the AADL-based Behavior Annex tools?

2. Can these data be used effectively to explore effects within the system of interest (the airplane FGS in this case) of changes made to improve the performance of the control studied? Said another way, does the architectural model permit sensitivity analyses based on alterations to the commonly-exercised design parameters in the specialty domain tools used?
3. What "translators" are needed to make these data sets convenient and easy to pass between the specialty domain tools and the architectural tools (the "single-truth" model)? Or, do we already have adequate "interfacing" software tools for analysis tools used by specialty engineers?
4. What are the interoperability "holes" (if any) in the AADL Behavior Annex for marrying the selected engineering discipline analysis tools to the architectural model? Are there gaps in our ability to exchange data in either direction?
5. Can the impacts of and altered dynamic conditions for failure states in the mechatronics components be readily transported between (both directions) a selected analysis tool and the architectural model?

To answer such credibility questions, verifications had to be carried out on a representation of a mechatronic subsystem in an analysis tool, like MATLAB. Figure 8 illustrates a typical graphical depiction of such a subsystem, the roll control channel for a flight control system. It includes mechanical linkages from the cockpit control inputs, hydraulic components to power the control system, electrical control circuits, and sensors for the feedback loops.

4.4.2. Behavior Analyses Use Case Results. Exercising these Use Cases provided potential users of SAVI credible examples of how data and results can be exchanged between layers in the system hierarchy. Structural and dynamic analysis tools were used to show that the "single-truth" model captures changes as elements were modified, which was the primary goal. The full report [11] covering the results of exercising this Use Case gives additional details; this section is a summary of only the most important results. As was the case with several of the tasks meant to be addressed during AFE 59, the planned tasks had to be modified or omitted because of lack of personnel resources and time to complete them. There simply was no time to complete both the aeroelastic evaluations of the wing structure and a simplified rigid body model of the entire aircraft. With no full airplane model available it was impossible to assess the SAVI AADL architectural model and its ability to capture and traceably follow alterations in the FCS to optimize flying qualities requirements spelled out in regulatory documents. Therefore, the team fell back to a limited assessment of frequency domain characteristics of only the hydromechanical characteristics of FCS components and alterations to them. With this caveat in mind, the results of the three evaluations completed are summarized in the following sections.

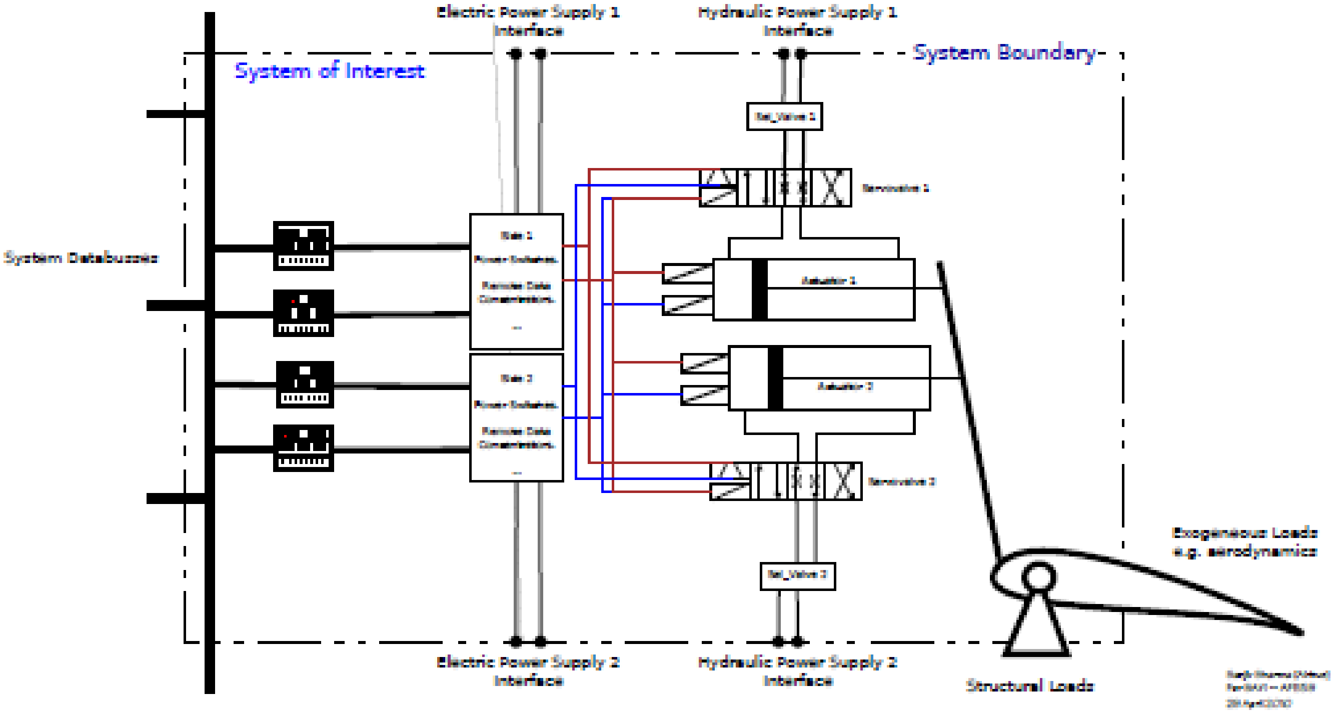

Figure 8. Electrohydromechanical Implementation of a Typical Roll Control Channel

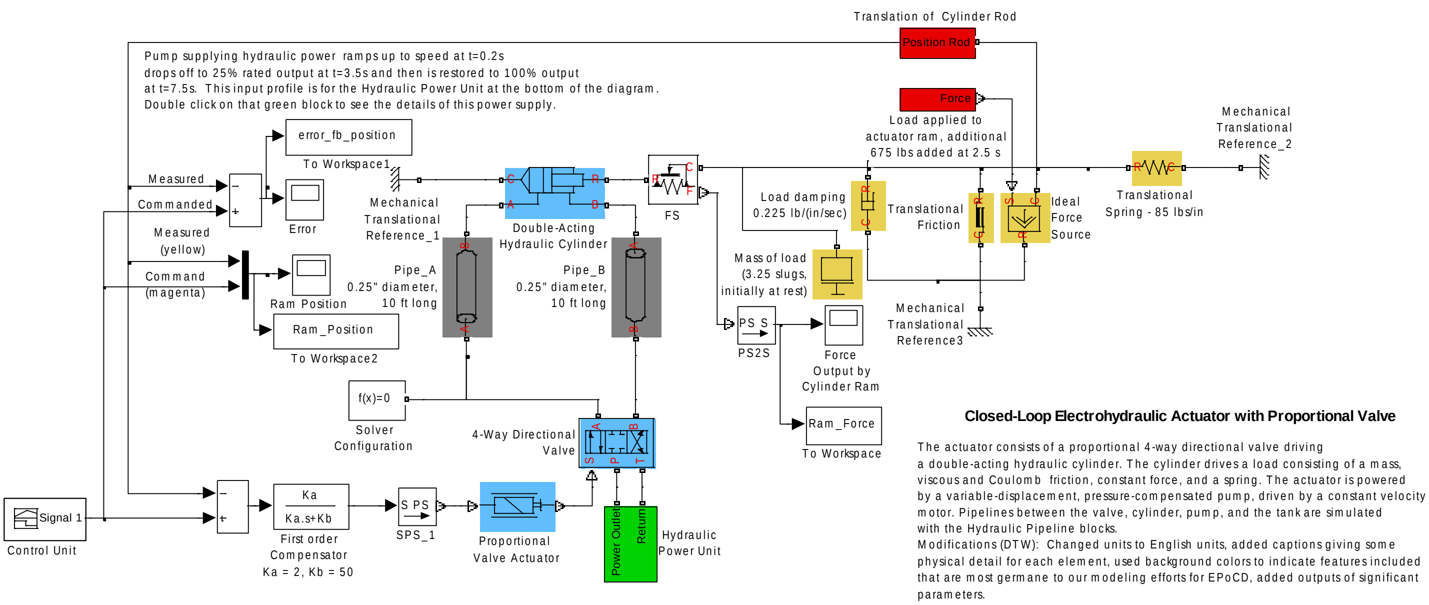

Figure 9. Modified Closed-Loop Electrohydraulic Actuator [xx] with a Proportional Valve for Demonstration

4.4.2.1 Time Step Results. The model of most interest as far as FCS mechatronic elements are concerned is an example (modified for this demonstration) provided by the dynamic analysis tool of a Closed-Loop Electrohydraulic Actuator with a Proportional Valve (Figure 9). This model allowed tailoring of the load on the actuator, using some or all of the usual mechanical components (shown in gold in Figure 9). The load was a step input force of approximately 675 lbf opposing the actuator ram elements (red blocks in Figure 9). The hydraulic power unit (green block) provided a separate input signal to the hydraulic power source for the 4-way directional valve. Finally, piping elements were included both inside this hydraulic power source itself and between the 4-way directional valve and the double-acting hydraulic cylinder. The inputs and outputs of the system can be read into the scope output and saved to the workspace for further analysis, manipulation, and plotting. Figure 10 shows, for example, an original input profile (coming from pilot commands or autopilot commands) to the actuator. These time domain responses allowed information from the hydraulic system to be passed as time histories to the architectural model, making the information available to other subsystems affected by these parameters.

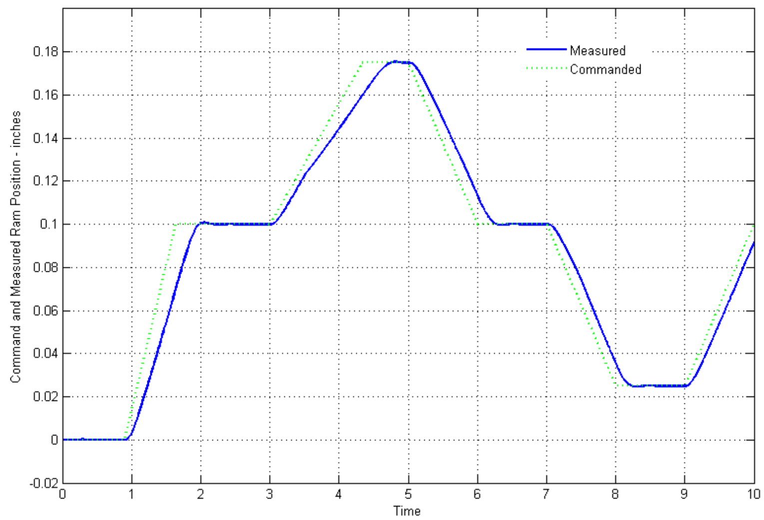

Figure 10. Commanded and Measured Inputs to the Actuator Ram

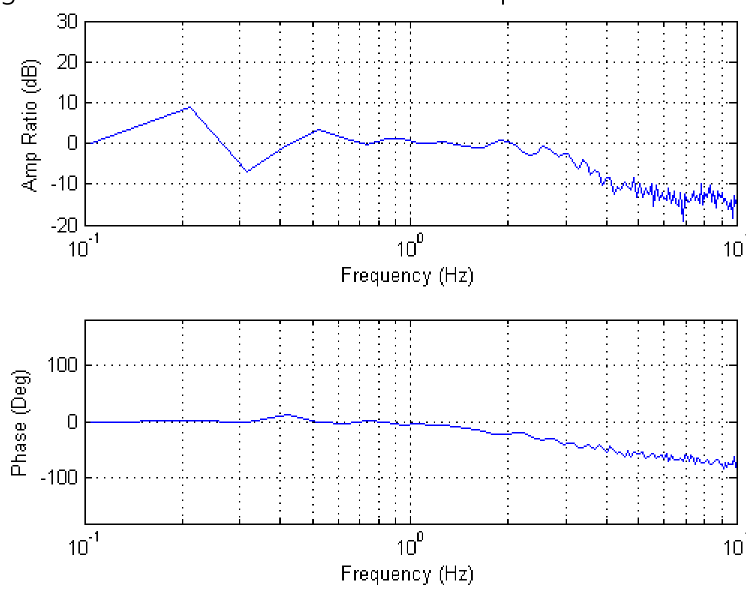

Figure 11. Computed Frequency Domain Actuator Response

4.4.2.2 Frequency Domain Results. Figure 11 shows the actuator response in the frequency domain computed by FFT from the time histories generated by the sinusoidal input [11]. Within the accuracy of the FFT, the response was flat (constant amplitude gain) up to around 2.3Hz, and then fell away up to roughly 7Hz, becoming constant again above that frequency. The amplitude initially drops away, then steadies and remains constant.

4.4.2.3 Aeroelastic Results. Finite element models required to carry out each type of analysis were slightly different for each purpose. The structural FEM was similar to a wing (Figure 12). The structure is rigidly attached to adjacent structure (it can neither rotate nor move) at the four nodes marked with large red circles. All other nodes were free to move in all directions and to rotate freely. Physical properties for each model are provided in the detailed documentation [11]. A corresponding modal response model, composed of triangular shell elements was required to compute the natural vibrational modes of the wing structure.

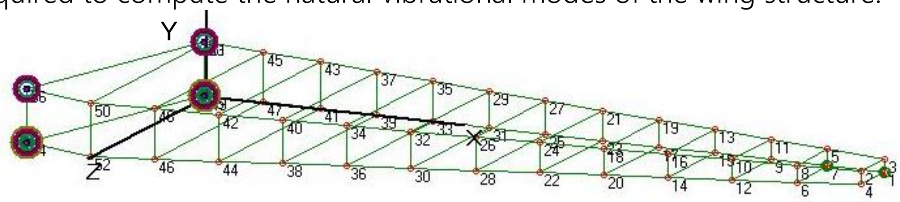

Figure 12. Wing Box Section Finite Element Model (FEM)

Finally, dynamic response to transient structural loads was calculated using a third model, a "two-dimensional" one composed of a number of "lumped-property" elements. Each of these models is described in the results report [11].

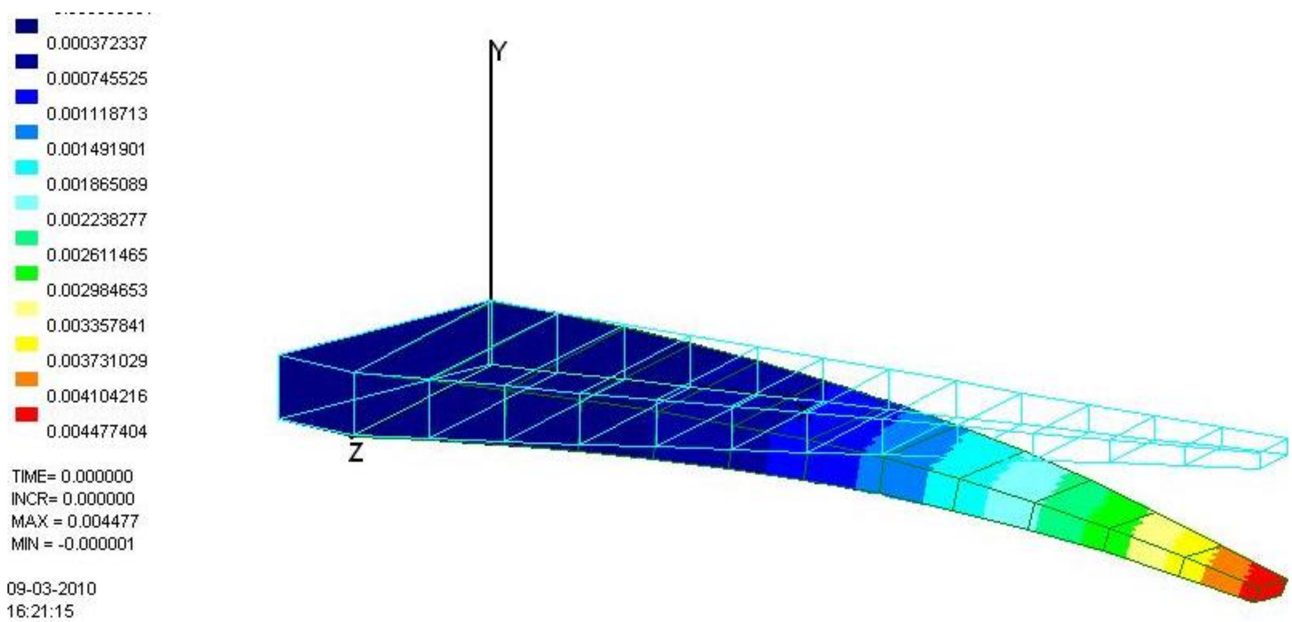

Figure 13. Wing Deflection for a Nominal Load of 980N (220 lbf)

To simply illustrate a static loads calculation, the wing tip was loaded with a point load and wing flexure due to that load was computed, along with the stress at each FE node (Figure 13). The accuracy of these figures was unknown, but irrelevant to the purposes of this Use Case. But, it was obvious that by plotting maximum stress as a function of load (using multiple computations of the model), prediction of critical structural loads, including for a load point at any location on the wing (for example, an engine mount rather than the wing tip) was possible.

Every structure exhibits natural modes of vibration. The FEM and the aeroelastic analysis tool computed frequency and relative magnitude of the natural modes of a structure. These natural modes represent the frequencies at which the structure would "ring" if subjected to an impulse blow. The first mode of vibration of the wing structure is the "flapping" mode – the wing tip moving vertically up and down relative to the wing root. Figure 14 plots the deflection from the wing centerline along the span of the wing.

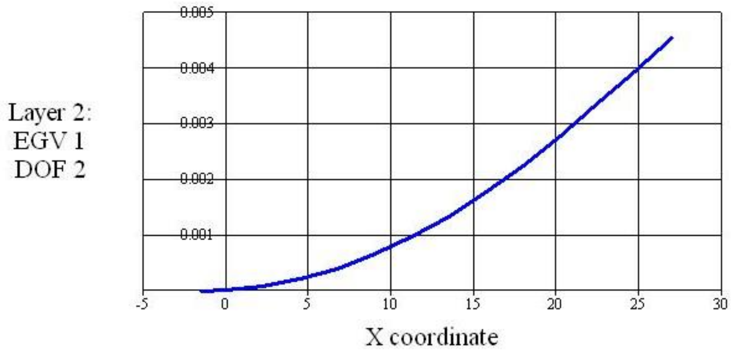

Figure 14. Wing Bending Mode 1

Table 13. Predicted First Ten Natural Modes

Mode	Frequency [rad/s]	Frequency [Hz]
1	2.087546323	0.332243316
2	7.172160632	1.141484817
3	8.214188238	1.307328661
4	16.30812387	2.595518527
5	27.67725126	4.404971349
6	29.77945635	4.73954768
7	47.78713531	7.6055588
8	62.48858614	9.945367369
9	70.38477092	11.20208421
10	97.47836879	15.51416423

Table 13 lists frequencies of the first ten natural modes of the example wing structure. The second natural mode has the wing tip still moving vertically, but inboard of the wing mid-span moves out of phase with the tip, forming a stationary node at about $^{2}/_{3}$ span. Mode 3 is the first torsional mode of the wing, with the leading and trailing edges moving out-of-phase. Anyone familiar with wing vibrations immediately recognizes that a wing does not "ring" as shown in this time history; this behavior resulted because the model was run with little aerodynamic or structural damping. When this damping was included, the expected rapid decay of the oscillation resulted.

The wing response to various load conditions was computed, including response to an impulse (shock) load. This impulse load modeled was a vertical impulse at the wing tip; Figure 15 illustrates the resulting resonant wing behavior with time. The fundamental mode is clearly visible with a period of approximately 0.33Hz (Mode 1 computed for Table 15).

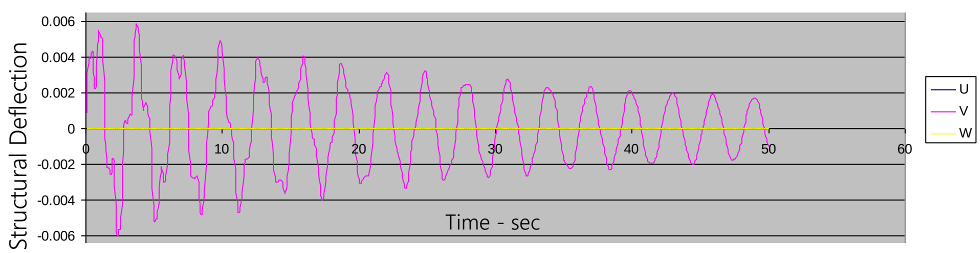

Figure 15. Wing Impulse Response (Undamped)

The FE tool also allowed computation of the structural response to non-impulse loads. For instance, the response to a sinusoidal load was also computed to explore the wing response to aerodynamically induced vibrations (flutter).

Having generated excitation for the wing structure with an arbitrary dynamic load, the FE Transient Response Model was driven with the force output of the dynamic analysis model (Figure 9). That simulation was run and the output actuator force time history was captured and applied as the input loading to the FE model, resulting in Figure 16.

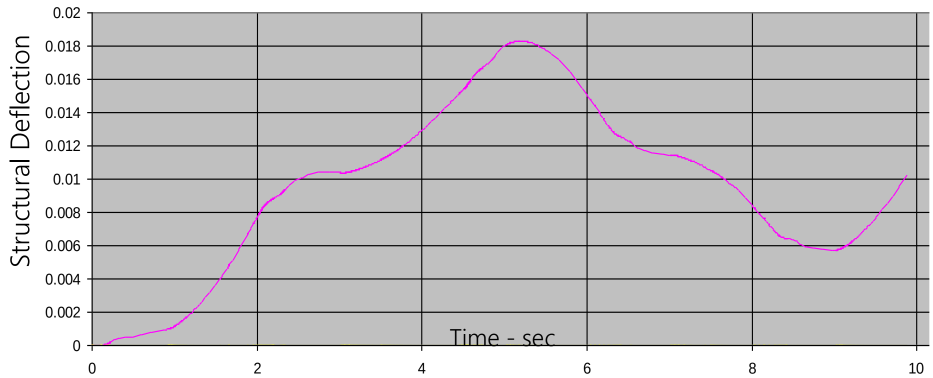

Figure 16. Wing Structural Response to Actuator Excitation

The FEM plug-in acted as a Model Data Exchange Layer component to provide a mechanized means of:

- importing FE models into the SAVI AADL Architecture Framework model;
- exporting FE models from the SAVI AADL Architecture Framework model; and
- executing FE analyses from within the SAVI prototype tool environment (OSATE).

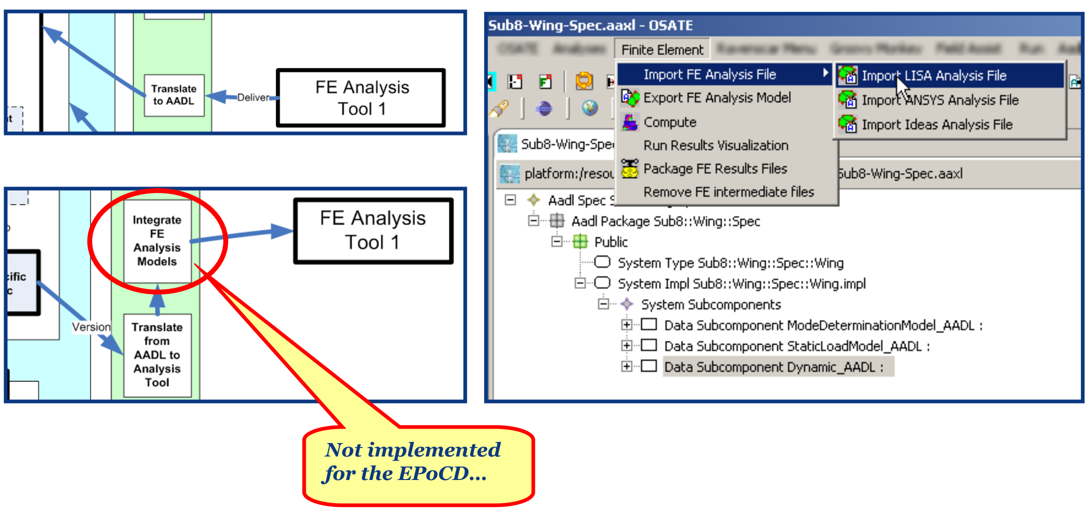

Figure 17. FEM plug-in - Requirements and Implementation

The finite element model definition was translated to a set of AADL property values, contained in AADL 'data' sub-components – represented as an AADL system component in the physical model structure. While AADL provides an 'Annex' mechanism for extending the language, property set definitions provided a better mechanism in this case. An individual AADL component can only declare a single annex clause, fine for describing the behavior or error modes of a component. However, the structural components needed several FE models in this example. The use of AADL data subcomponents was, therefore, a natural choice. Each FE model is represented as a single data subcomponent, and the structural component contains as many data subcomponents as needed.

The functions required of the plug-in (Figure 17) were modeled:

- to capture new functionality required and to integrate functionality provided by the external tool; and
- to create the Java code for the plug-in by code generation.

The FEM plug-in provides a menu interface for importing FE models from the external tool, exporting models back to the tool, and executing the external tool directly from the AADL model environment. For now, automated merging of two or more FEMs was not attempted, along with packaging and clean-up functions. Figure 18 summarizes data flows across the Model Data Exchange Layer and Model Repository that the FEM plug-in enables, and these flows were demonstrated. FEM plug-in development is documented in [12], [13], [14], and [15].

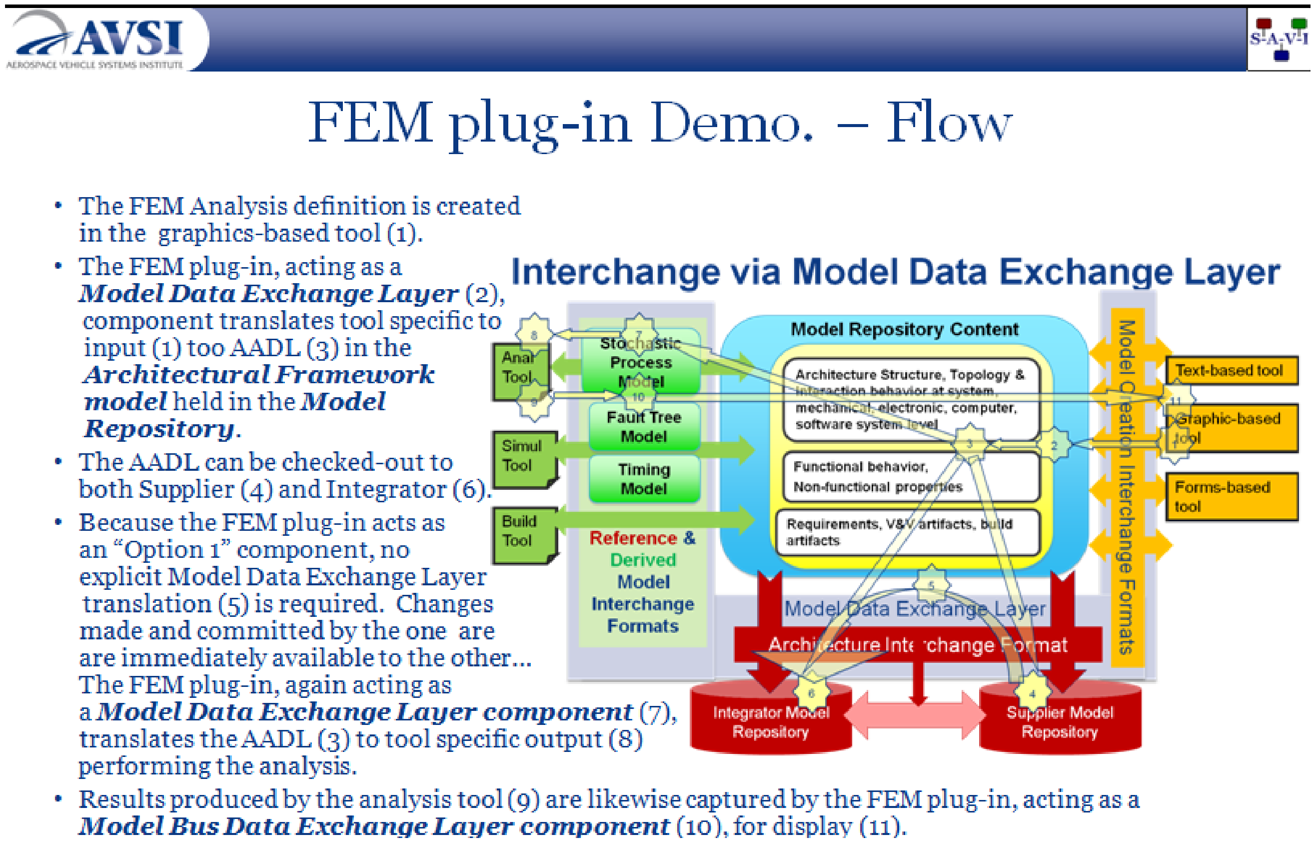

Figure 18. Data Flows across the Model Data Exchange Layer and Model Repository Enabled by the FEM Plug-in

4.5 Evaluation of Model Demonstration Results

The results of the demonstrations just summarized point unequivocally to proceeding with the SAVI development. When the SAVI team started this phase of the exercise, the confidence level was high; the AFE 58 PoC [1] had given the group a glimpse of how powerful the architectural model was in drawing together a complex system. But there were nagging doubts about many of the issues addressed in AFE 59. Could the mechatronic elements (mechanical, hydraulic, and electrical control), as well as the structural, analyses all be captured and viable two-way communications links be established with relative ease? Could we actually reach outside AADL, even with the Error Annex, and bring reliability data and safety data sets into the architectural umbrella? Were the computer automated design tools and their myriad physical details going to be viable to move back and forth between the ECAD and MCAD files and AADL? Since all the answers to these questions came back positive, the SAVI team has every reason for optimism. Even though only a small percentage of the Use Cases known to be needed were completed during AFE 59, there were no negative results so far. There is still a lot of work to make sure all the analyses needed to support a full up airplane design, but the SAVI team, as thin as it resources have been, has not yet found a subsystem that appears to block the concept. All perceived problems center on how best to move ahead, not can we move ahead, with the development of the VIP.

ROI ESTIMATION

Return on Investment (RoI) for the System Architecture Virtual Integration (SAVI) Virtual Integration Process (VIP) has been a key driver of the development efforts undertaken by AVSI since 2007. During AFE 58 a first iteration of estimating tools was generated and exercised. Several shortcomings were also noted and AFE 59 provided an opportunity to address some of those shortcomings.

5.1 Targeted Improvements for AFE 59

The most noticeable improved RoI capabilities made during AFE 59 are: (1) providing capability to wrap a Monte Carlo iteration loop around the COCOMO II estimation methodology and (2) adding capability to estimate RoI for all levels of the system hierarchy (the so-called Tier n supplier RoI). A User's Guide [16] detailing how to use the SAVI RoI Estimation Tool (SRET) has been written and should allow any SAVI participant to adapt the Tool to that organization's needs.

AFE 59 was also laid out with the intent of adding the capability to assess hardware effects on the VIP independent of software effects. The RoI approach initiated during AFE 58 utilized a pure multiplier of the software cost/benefits to account for hardware development. The UGID WG assessed this approach and concluded that quite likely this "constant parameter" gave more weight to the hardware effects and failed to capture the reality of how hardware requirements errors and redesign of hardware by linking it directly to the amount of software (as measured by SLOC) in a system. However, it quickly became obvious that the team lacked the time and manpower resources to produce a new element to estimate the effects of hardware on RoI and still meet the objectives listed above.

Roughly midway through the project's term, factoring an incremental development of SAVI plan into the RoI estimation process became important when economic conditions dictated that a concentrated, large-scale effort, as had been originally planned, simply was not feasible. This work was started and is close to fruition, but has not yet been fully evaluated. Again, constraints on available manpower on the SAVI team prevented completion of this evaluation. The intent is to complete that part of the RoI effort early in the next phase of SAVI development.

5.2 Added Capabilities

The AFE 59 UGID WG built upon the core of AFE 58 work, the COCOMO II RoI estimation tool by Boehm and his collaborators [17]. The following additions to that algorithm led to a modified Excel-based worksheet, the SAVI RoI Estimation Tool (SRET). It has a revamped interface plus a new User Guide to considerably enhance usefulness and efficiency of RoI calculations. But the overall system RoI for using the SAVI VIP on a nine-year commercial aircraft development was largely unchanged from the original COCOMO II estimate generated during AFE 58.

5.2.1 Monte Carlo (SCAT Plugin). Early in work under AFE 59 the UGID WG started looking for a way to apply Monte Carlo techniques to the COCOMO II estimations from AFE 58. The Software Cost Analysis Tool (SCAT) [18] includes a Monte Carlo analysis algorithm captured in an Excel plugin written by George Fox of NASA JPL. Steve Helton from Boeing further specialized this SCAT set of worksheets to fit the express purposes of estimating SAVI's RoI by adding input sheets designed for SAVI purposes, by adding summary tabs tailored to specific features of the VIP, and by adding several basic sheets (covering prioritized systems unique to aircraft developments) to the calculation procedure.

Table 16 shows the result of a ten-run Monte Carlo estimate for such an example development, assuming a triangular distribution in estimates with 30%, 40%, and 50% improved software error discovery/fix with SAVI.

Table 14. Example of Monte Carlo Differences in RoI Estimates [19]

Average RoI for ten Monte Carlo runs			Overall average deviation		
78.09%	98.33%	115.88%	0.81%	1.05%	1.73%

The basic RoI predictions using the SRET are similar to earlier AFE 58 predictions, though and the results should be improved by utilizing a labor rate that takes into account inflation. The AFE 58 calculations erred in using the 2006 labor rate and did not properly account for inflation. The expected RoI for a 9-year commercial aircraft system development pays for the development of SAVI and for its deployment in that first large system development when SAVI's VIP is applied. For such a project the minimum expected arithmetic RoI is over 70%/year [19]. The UGID WG further refined SCAT to include a color-coded interface to assist the user of the tool.

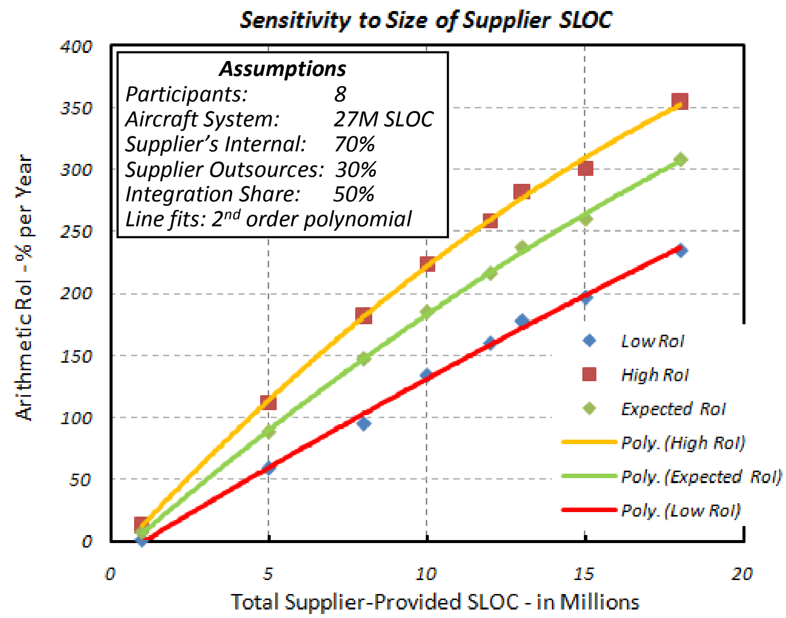

Figure 19. Sensitivity of RoI to Total Responsibility of Tier n Supplier

5.2.2 Tier n Supplier RoI Estimation. The most important job was to provide and demonstrate capability so suppliers can estimate RoI at all levels in the system development hierarchy. The UGID WG spent several weeks refining the procedures for calculating supplier RoI and simplifying the input/output procedures for the SRET. Ultimately the WG produced a straightforward example calculation that any supplier can use. This example assumed a level of commitment (in terms of total SLOC to be produced), a percentage of outsourced versus internally produced code, and a percentage of responsibility shared with the system integrator for integrating the supplier's subsystem into the overall system. This example calculation included looking at sensitivity of the RoI estimate for a supplier to any of these assumptions. Taking as nominal (Tier n), a supplier who contracts to provide a total of 1,000,000 lines of code, who decides to contract out 30% of that effort, and who accepts a 50% share of the responsibility for integrating his subsystem into the aircraft, the SRET estimates that such a supplier's arithmetic RoI is expected to be approximately 7%/year for a commercial aircraft development that takes nine years to complete. Figure 19 illustrates one of the sensitivity calculations of that effort related to the total responsibility of the supplier within such an effort; the lower limit of the green line is at approximately 7% expected RoI. The scatter in the chart is largely due to the Monte Carlo nature of the SRET calculations. A statistical variation of less than ± 1% occurred for each RoI estimate when 10 runs were done. The points on this chart are for one run only. It appears that a second order polynomial fit adequately averages the statistical variation from the SCAT Monte Carlo algorithm. Charts for the sensitivity to the other assumptions listed above are included in the detailed report [19].

5.2.3 Incremental Development Modifications. This growth of the RoI estimation tool was simply outlined during AFE 59. Steve Helton has laid the framework within the spreadsheet for such a next step, but there was not enough time to fully develop this framework. The UGID WG strongly recommends that this effort be continued during the next phase of the development. There are two compelling reasons for this recommendation. (1) SAVI appears to be headed toward incremental (low rate development) at least for one more year and the capacity to quantitatively evaluate the impact of that slower development on RoI is critical to justifying resource commitments over this longer term of development. (2) The Integrated Process Plan [20] depends strongly on choosing the appropriate subsystem(s) to be developed first within an incremental development framework. Multiple revisions are likely as priorities change during this slower SAVI development. Quantitative assessment of how this stretched effort affects RoI is part and parcel of choosing an intelligent sequence of subsystems to be studied as the VIP evolves.

5.2.4 User's Guide. A User's Guide was written to carefully spell out how to install the SRET and how to start using its features [16]. This short document (1) describes detailed installation procedures for SRET and its plugin modules for Monte Carlo analysis; (2) steps through how to use the tool, along with the default assumptions and inputs fields and how to modify them; (3) lays out details of how to make inputs and how to extract the output estimates; and (4) lays out how parameters are set to tailor the estimate to the system or subsystem being considered by the user. A very brief list of definitions, abbreviations, and acronyms is appended to the User Guide.

5.3 Recommendations for Further RoI Improvements

The SRET was further modified and also exercised to produce RoI estimates for suppliers participating in such a commercial aircraft development at varying levels in the system hierarchy. Nominal (using similar conservative assumptions as described in AFE 58) RoIs for such suppliers are expected to be at least ~7%/year for when the supplier is responsible for a total of 1 million lines of code for the system integrator and shares the cost of integration equally with the system integrator. This result also assumes the suppler chooses to outsource 30% of this code, as opposed to producing it all internally.

Sensitivity of the SRET to the assumptions listed above was also addressed during this effort. The percentage of errors discovered and corrected was varied and the prediction is that even if no more than 10% of the software errors are discovered and corrected through the SAVI VIP, the basic return on investment to the overall system development (the aircraft system) would still be ~4%/year for a nominal commercial airplane development. If the discount rate is tripled (to 30%), the RoI for this commercial airplane example system is expected to remain at approximately 20%/year. It is also notable that if the discount rate is less than the assumed 10%, the RoI goes up almost exponentially with any decrease. For a Tier n supplier, the AFE 59 RoI estimator produces the best return for those suppliers who take on large responsibilities and plan to do most of the work internally, as opposed to outsourcing the work. The sensitivity results also show that taking on a large share of the integration costs reduces the Tier n supplier's RoI.

The team also began an incremental development module for an improved SRET that better fits the current resource-limited development environment affecting most SAVI participating companies. The latter capability should allow a more detailed breakdown of costs and benefits of early but incomplete SAVI capabilities.

Unfortunately, neither time nor resources were available to complete this part of the effort or to incorporate a hardware RoI module into the SRET that is independent of software embedded in the system. Both of these capabilities should to be addressed in the next round of SAVI development.

Use Case, Language, Analysis, and Compatibility (UCLAC) Results

As noted earlier much of the prioritization and evaluation tasking for the UCLAC WG was deferred simply because manpower resources were scarce during AFE 59 and the DEMO tasks were deemed to have greater importance. So this section largely points out work that is still to be completed; these are not finished tasks and final results.

6.1 Additional Use Cases

6.1.1 Shadow Projects. The next SAVI development emphasis is on Use Cases that support specific real world projects that are proceeding through a conventional (*not* an architecture-centric model based systems engineering (MBSE) integration process) development. The next step is to parallel one or more systems developments with a SAVI-based integration process. The goal is to glean as much as possible about integration deficiencies *before* they affect the physical integration. SAVI will model as much of the system as possible with the available time and resources by facilitating choices as to which system elements to place in the architectural model and by prioritizing the development of translators and interfaces with analysis tools used by avionics suppliers and airframe integrators. It may be necessary to separate the SAVI parallel effort and to "sanitize" the data used so that SAVI team members can carry out their separate virtual integration on a similar vehicle and similar subsystems with minimal interference with the conventional development. This approach is one way to provide partitioning so that the project principals are never concerned with compromising their data with competitors on the SAVI team. But, it requires additional work to separate data sets and there must be continuing vigilance that sanitized data are representative of the real system artifacts. The approach places the burden of developing models for the VIP architectural model with SAVI. Of course, this architectural model and any findings unearthed by the SAVI team must be communicated to the development program. There must be representatives on the SAVI team who work closely with the developing subcontractors and the integrator; perhaps some of the key players will even have membership on both teams.

Goodrich Aerospace has also proposed a smaller shadow project that focuses on a vibration monitoring subsystem. The current Goodrich preliminary plans follow the same outlined approach as described in the preceding paragraph. However, this effort focuses on a supplier's system (a subsystem to an aircraft manufacturer) and should give direct indication of the impact SAVI can have on a supplier's business model.

Details about additional Use Cases are one of the first orders of business under AFE 59 Supplement 1. As soon as details are available on best candidates for the candidate system, the potential Use Cases generated at the kickoff for AFE 59 [21] must be reprioritized by the SAVI team for which ones best support the planned shadow projects.

6.1.2 SAVI Version 1.0. Beyond the shadow projects, the SAVI roadmap shows output of a Version 1.0 of the VIP. While this version is likely to be limited, a demonstrated capability (at least for the types of systems evaluated in this first set of shadow projects) that allows prospective users to take the SAVI VIP into a few selected production contracts is sought. Tasks for AFE 59 Supplement 1 include effort intended to exercise more Use Cases, including a few more than these shadow projects will require. These additional Use Cases will underpin Version 1.0 of SAVI. A readiness level equivalent to a TRL of 3.5 is sought for this "alpha" level MBSE capability process.

6.2 Multi-Language, Analysis, and Compatibility Considerations

Accepting and using multiple languages, indeed the necessity to do so, has long been on the SAVI team's "to do" list. However, little careful definition of how to handle these multiple languages has been done, other than efforts to show that various analysis tools can be accommodated by and can interact with the architectural language (AADL). Interfaces with SysML and DoDAF, as well as analysis software, must be addressed systematically. Analysis tools that interact with AADL, their translators, and their compatibility with one another should be systematically considered. SAVI seeks to be "tool-neutral" but it may not be prudent "to be all things to all analysis tools". The effort in AFE 59 Supplement 1 will evaluate the language issues in light of SAVI experience to date and lay out multiple language issues, set down priorities, and prioritize the language work necessary to underpin SAVI Versions 1.0 and beyond.

GROWTH, COMMUNICATION, AND EXPANSION (GCE) RESULTS

Table 15. Targeted OBS by Category [1]

Legend: Potential Leader | Contributor | Subcontactrator

	WP0 - Management	WP1 - Acquisition model	WP2 - Analyses	WP3 - Requirements	WP4 - Languages & collaboration	WP5 - Pilot projects	WP6 - Tools & standards	WP7 - Certification
AVSI	L							
Platform Integrators	C	L	L	L	L	L		C
Engine Suppliers	C	L	L	L	L	L		C
Systems Suppliers	C	L	L	L	L	L		C
Tools Vendors	C		C	C	C		C	
IS/IT Integrators	C					S/C	S/C	
Labs & Academics	C				S/C	S/C		
Airworthiness Authorities	C	C	C	C		C		L

7.1 Needed SAVI Organizational Breakdown Structure

Tasks 7 and 8 of the EPoCD were to carry out an "aggressive, targeted effort to attract participants to the SAVI development." This goal was implemented by building toward an Organizational Breakdown Structure (OBS) specified in the SAVI Roadmap of AFE 58 [1]. Table 15 shows the type of participant needed for each of the planned Work Packages. EPoCD Tasks are broken into associated subtasks and deliverables in Table 16.

Table 16. AFE 59 Outreach Efforts

Task 7: Identify Potential SAVI Participants within the Aerospace Industry	Deliverable
a. Identify, Prioritize, and Target Potential Non-AVSI Participants for the SAVI Project	Marketing plan
b. Encourage AVSI Participants to Solidify Support for SAVI Development	Internal communications plan
c. Develop Strategies for Publicizing (Public) Results from the SAVI Effort	External communications plan
Task 8: Forecast and Track Progress in Achieving Broader SAVI Participation	Deliverable
a. Expand the Communications List (Task 7a) to include Target Goals for New Participants	Target goals for each class of participant
b. Track Participation Added to the SAVI Effort	Tracking chart

7.2 GCE Results

7.2.1 Task 7a. As suggested in Table 1, Dassault Aviation, EMBRAER, and Sikorsky Helicopter are manufacturers that have shown interest in SAVI; Thales and Meggitt Industries are two suppliers that are potential SAVI participants; the Mathworks, IBM (especially through their Telelogic division), and Esterel Technologies are examples of software vendors who support aerospace systems design; Pratt and Whitney and NASA JPL have both expressed some interest in SAVI; the European Aviation Safety Administration (EASA) is another regulatory body that would be strongly affected by wide-spread implementation of the SAVI methodology. EMBRAER has joined SAVI for AFE 59 Supplement 1 and Thales is considering joining both AVSI and SAVI. Esterel Technologies attended the AFE 59 Close Out session and is evaluating how to best interface with SAVI. A regularly updated outreach plan resides in the Members Only section of the SAVI web site.

7.2.2 Task 7b. Not all AVSI members have been involved in SAVI. Goodrich Aerospace, one of the few AVSI members concentrating on hardware-oriented subsystems, joined for all of AFE 59. (Notably missing from the AVSI and SAVI membership are some of the other hardware-based systems suppliers like Parker and Moog, who were charter members of AVSI). Hamilton Sundstrand has not been an active participant, but is reevaluating that position. Honeywell did not actively participate in AFE 58, monitored SAVI activities during AFE 59, and has rejoined for AFE 59 Supplement 1. This subtask worked to both attract members like these to support the effort and to provide assistance to the existing participants to continue their support for the development.

An internal communications plan was carried out, both by developing web-based resources and by providing in-person support to AVSI members. Materials (presentations, video demonstrations) suitable for in-house briefings, and self-paced learning about SAVI were generated for the SAVI web site and updated regularly. On-site visits by the SAVI PM are available to SAVI participants/AVSI members. All participants are encouraged to seek out available forums and present results within their domains of interest and many did so during AFE 59.

7.2.3 Task 7c. The SAVI web site now includes dates/contacts/abstract calls and other information to encourage participants to communicate their SAVI experiences with their peers; invited papers will be supported by the SAVI PM and AVSI Director whenever appropriate and as travel budgets and time constraints allow. The SAVI PM and the AVSI Director supported several briefings in Washington, D.C. and gave papers at technical conferences. They also supported numerous teleconferences with participants and associated organizations within their budget and time constraints. These efforts are documented on the GCE tracking section of the SAVI web site. Liaison visits by the SAVI PM and AVSI Director with other groups working in this problem space to build consensus.

7.2.4 Task 8a. As can be inferred from the preceding sections, this communications list (both internal and external to the SAVI project), was maintained on the SAVI web site under the AFE 59 Members Only section.

7.2.5 Task 8b. The results of the growth efforts can also be inferred from the preceding paragraphs: addition of Goodrich Aerospace for AFE 59, reinstatement of Honeywell as a participant for the next phase of the project, the addition of EMBRAER, and the potential for further reinstating/adding more participants or new participants (GE Aviation Systems/Hamilton Sundstrand and Thales) early in AFE 59 Supplement 1 are all positive indicators for these efforts.

INTEGRATED PROGRAM PLAN

8.1 Reassessing SAVI Development Approaches

The SAVI program has changed considerably since its original structure was laid out; indeed, AFE 59 Supplement 1 recognizes the extent of these changes by proposing a modified roadmap and providing more detail in the overall approach to bringing the SAVI VIP to useful capability. Though this Integrated Program Plan (IPP) is likely to change further in the years ahead, it will be a useful guide for strategic planning.

8.1.1 AFE 57, 58 "Initialization". SAVI's 2008 effort (with AFE 58) was preceded by several months of effort, largely carried out by John Chilenski at Boeing, and fostered within AVSI. AFE 32 carried the seeds of MBE to AVSI EB deliberated as early as the 2003 about the need for such work and how best to carry it off.

8.1.2 Pragmatic Constraints. The project has always faced the challenge of garnering enough support from its members to capitalize on the cost of developing such process. Early on it was estimated that such a development would cost at least $50 million dollars (and that estimate quickly doubled as the team laid out the tasks and the challenges to be overcome). This cost led to the conclusion that this new paradigm was too expensive and too daunting a challenge for one organization to tackle on their own. Since the most benefit was expected to accrue if the SAVI paradigm were adopted at all levels and since it appeared to offer promise of a healthier return on investment in that event, the argument was made that it fit naturally into the AVSI framework which is designed to promote cooperation that will be of exceptional benefit to the entire aerospace industry. Thus, almost from its inception, SAVI has been an effort that drew together a large majority of the AVSI membership, including government regulatory agencies.

The earliest SAVI plans envisioned 15-20 participating organizations spread over a range of organizations (as suggested already in Table 1. The expectation that SAVI was a rather complete "game-changer" meant that regulatory agencies needed to be involved almost from the beginning of the effort to help guide the process in a way that could be implemented sensibly from a regulatory point of view. The rule-makers needed to be in lock step with such a fundamental change in how high-cost, very complex systems were to be developed. For similar reasons SAVI planners from the outset recognized that ultimately standards bodies had to know about (perhaps even needed to anticipate) the rather sweeping changes that SAVI might bring. Thus, the need for a broad array of stakeholders was clear from the beginning of the project.

Unforeseen constraints on SAVI growth also appeared rather early in the life of the project. As noted above, 2008 was the time when the project took on the SAVI name. The global economic downturn, which every participant felt, also began about this time. Added to that forcing function, the two biggest commercial aircraft manufacturers and the military manufacturer were all engaged in new product developments at this time and all encountered difficulties that absorbed huge amounts of brainpower. So, though these developments underscored the need for SAVI's projected paradigm shift, SAVI's priority in R&D budgets was well below those projects that offered to reverse the economic currents in which these companies were engulfed. SAVI has tremendous payback potential, but a large investment in SAVI at that time was not palatable to most managers in this environment.

8.2 Avenues of Approaching SAVI Development

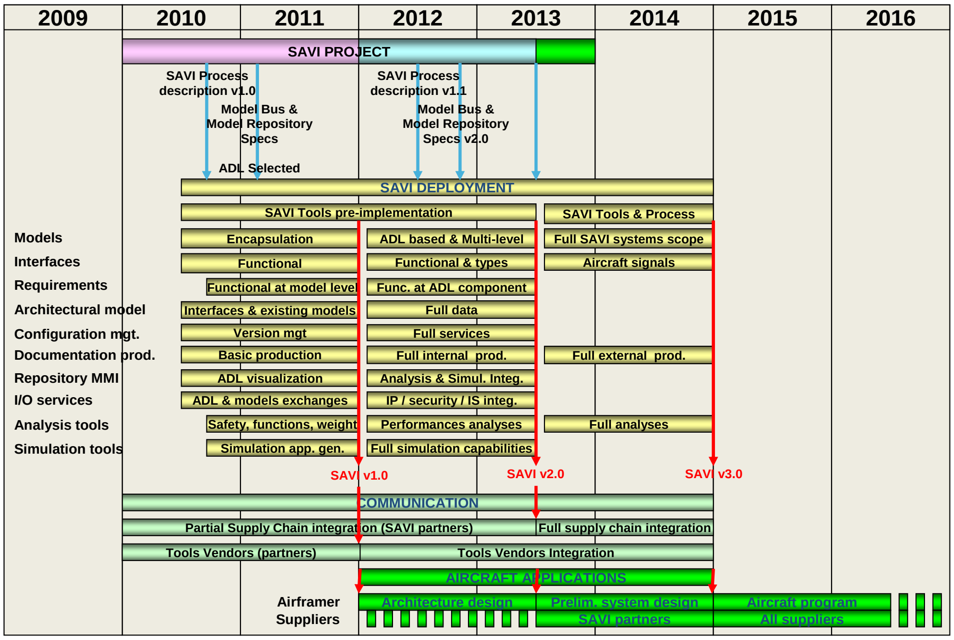

Figure 20. SAVI Optimistic Roadmap

8.2.1 Minimum Time (AFE 58 Roadmap). The first roadmap for SAVI development was developed as one of the primary tasks of AFE 58 [1]. Figure 21 is a summary portrait of what the SAVI team considered the minimum time to get a proven virtual integration capability into the acquisition cycle. In 2008 when this roadmap was set down, it appeared that each organization might be able to support the effort rather vigorously (with at least two person-years of in-kind effort and a cash contribution of approximately $25,000 to $30,000 dollars each year) and have limited SAVI capability available by the end of 2011 and a fully vetted (with at least two full-blown development projects) available by the end of 2014. To accomplish these ambitious goals, the membership was expected to approximately double from about 7 organizations to at least 15 by the end of 2010. To meet this timetable it was also recognized that the effort additionally had to attract the attention of tool vendors willing to invest heavily in new software to support the interactions between analysis tools and SAVI's architectural model.

8.2.2 The Long Road Home. At the opposite end of the possible time scale (in contrast to the AFE 58 roadmap) is continuing to devote approximately 1050 person-hours/participant to development of the SAVI VIP plus a cash contribution of approximately $25,000 annually/participant to the development effort. This resource commitment is roughly 1/3 of the original planned effort. Moreover, the number of participants has not grown as rapidly as originally projected. The likely stretch to the program is an additional 10-15 years before a fully developed SAVI virtual integration scheme could be offered to users with confidence. Some benefit might accrue with the SAVI VIP at TRLs of 6-9 at an earlier date, it is equally likely that such a stretch out in development time would add a significant schedule penalty for SAVI development. More important, the ability to sustain continuing support from the number of participants needed for this effort is highly questionable. Taking 10-15 years to ramp up to a better way of developing systems that are the core business of the participants suggests a very low sense of urgency. Or it may reflect nothing more than economic reality. In any event, compared to similar projects already completed or now being initiated, even ten years is a very long gestation period for the SAVI effort.

8.2.3 The Middle Ground (Incremental Development over a Longer Period of Time). Even as early as late 2009 it was obvious that the AFE 58 roadmap could not be supported at the levels described in the preceding paragraph. AFE 59 was to "bridge the gap" for what everyone hoped would be a short term dip in the economy. The AVSI EB brought that optimistic planning abruptly to a halt with a healthy dose of realism in mid-2010. From that point forward (recalling that AFE 59 carried a task to develop an integrated process plan based for the SAVI project based on the AFE 58 roadmap – Figure 21), the long range planning task took a completely different direction. The SAVI planners began looking toward what is called an incremental development approach.

The new questions to be answered were:

1. What approach for SAVI development is likely to demonstrate the greatest return for the least investment in SAVI development? (Often, this question was couched in terms like: "What are the 'low-hanging' fruit?"
2. How can SAVI make the RoI estimates more credible? (More than one manger simply said: "The RoI estimate is not believable.")
3. What aircraft systems are most amenable to improvements using SAVI's VIP? (Setting priorities for a slower SAVI development meant choosing which systems to tackle first.)

This redirection meant that the tasked Integrated Process Plan under AFE 59 was a first iteration effort and the SAVI team had to first consider the alternatives. If the development of SAVI is allowed to stretch out indeterminately, then very few organizations are likely to be attracted to that kind of effort. So, while SAVI can no longer plan on two person-years of in-kind effort and a cash contribution of approximately $25,000 to $30,000 dollars each year from each participant, there has to be a middle ground.

While it is simply pragmatic that SAVI development cannot expect to follow the AFE58 Roadmap in the current (2010) economic current climate, the goal must be to ramp up the development so it can be completed in less than the 10-15 years estimated if we continue at the current pace (the "Long Road Home"). There are two general ways to accelerate the development pace, both of which are designed to add useful resources: (1) increase the participation levels of each participant and/or (2) add more participants. In fact, the most obvious answer to the conundrum cited in the two previous paragraphs is for SAVI to do both (1) and (2). In fact, the proposed Roadmap revision shown in Figure 22 suggests that SAVI needs to be ramped up by both of these approaches so that at least by the end of 2012, the team members can devote approximately 1500 person-hours per participant and that the membership of the SAVI team will have approximately doubled. These goals are ambitious. This approach will require each participant to contribute roughly 50% more in-kind labor than are now being devoted to the project while the rate of adding new participants would have to at least equal the expansion rate seen so far for each of the next two years (2011 and 2012). Since SAVI increased from 6.5 participants (in AFE 58, counting 0.5 for the one member whose in-kind participant disappeared when that representative left the organization) to 9 participants in AFE 59, SAVI could be at 15 participants by the end of 2012. That rate is about the minimum that would allow the effort to move forward at the desired pace. The SAVI Roadmap report from AFE58 [1] suggested a total of approximately 20 participants (including 6 tool vendors – see Table 1 – where target contacts already made during AFE 59 have been added to the original table) are needed to carry out the work indicated in Figure 21 from 2013-2015. Therefore, this synergistic level of participation may be within reach. However, a continuing well-organized growth and expansion effort must be pursued by all members of SAVI (and that policy must be supported with travel funds and time spent in explaining SAVI to the broadest set of potential members). It is noteworthy that, if one adds only organizations with which SAVI has or has had ongoing discussions during the course of AFE 59, the target list of participants adequately fills out the anticipated number and type of needed organizations. The weakest link appears to be attracting tool vendors, but that apparent weakness may strictly be a function of timing as to when such participants are needed in the effort.

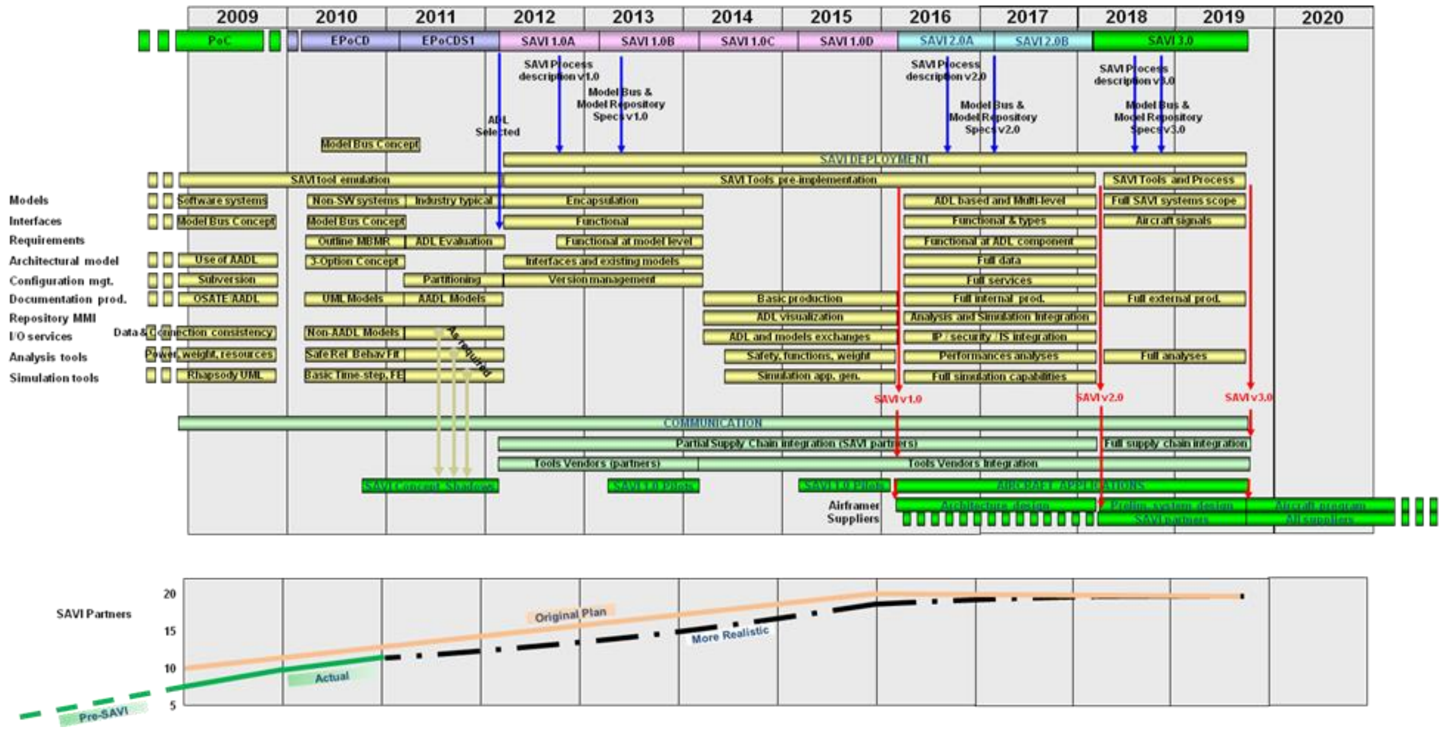

Figure 21. Revised SAVI Roadmap (for the "Middle Ground" Recommended Approach)

8.3 Recommended Way Forward

The path most likely to succeed in the current environment appears to be the "middle ground" (Section 8.2.3).

1. SAVI is developing Supplement 1 for AFE59; it has a good chance of attracting new participants, especially new industrial members with strong modeling skills and willing to increase their commitment up to at least 1500 person hours annually to the SAVI development through 2015. It may be necessary to increase this level of in-kind contributed effort gradually as economic conditions permit.
2. Current SAVI participants should continue to support SAVI development at least at the levels provided during AFE 59 (approximately 1050 person hours/per year for each organization) and make every effort to increase that in-kind participation at the earliest opportunity to the goal stated above.
3. Each participant should plan to contribute approximately $25,000 cash toward management overhead and/or contracted effort for each year through 2015 and to fully support by every means possible expansion of SAVI participation to members with the needed skill sets.
4. The SAVI PMC should come to agreement on AFE 59 Supplement 1 that incorporates the tasking outlined in the current draft that is being circulated to the PMC for comment from every participant as soon as possible. A Supplement is likely the best vehicle for the 2011 effort (and likely repeated as Supplement 2 for the 2012 effort also) because it likely offers hope of slightly increased funding for an external contractual support for one or more critical areas. Reaching agreement on AFE wording and tasking would also allow circulating that document to potential new participants (with a viable

nondisclosure agreement in AVSI possession) for input from them as well. The primary focus of the current draft AFE59S1 is on (1) one or two "shadow" or pilot projects and (2) adding Use Cases demonstrations to support these specific projects, along with a smaller effort to further improve the RoI estimation methodology.

5. A decision regarding whether to escalate the pace of SAVI development at the end of Supplement 1 or continue with a second Supplement at that time must be made by mid-2011. This decision amounts to at least a tentative commitment to full development of the SAVI process at the end of Supplement 1 or a decision to continue with a slower-paced incremental approach through at least 2012. To support this decision gate, management effort during Supplement 1 must address the effect of stretching out SAVI development on RoI estimates and on how to prioritize subjects of the shadow projects described above.

Careful attention must be given to the mix of skills provided in each planned set of tasks, whatever the decisions made regarding escalation of SAVI development. This factor means that the expansion of SAVI participation must be positive and sharply focused on the needed mix of skills for these tasks. At this writing, two organizations, with very strong skill sets in MBE, have indicated that they plan to join SAVI for AFE 59 Supplement 1. That is a positive trend, but the SAVI team must continue to search for more participation from organizations with skill sets appropriate to developing the Virtual Integration Process.