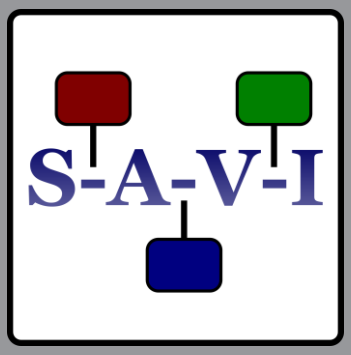

System Architecture Virtual Integration

SAVI AFE 61 Report

Summary Final Report

Release: Public Release
Document ID: SAVI-AFE61-04-001
Related AFE Task: Task 4: Management
Date: 5/9/2015
Issue: Version 1.1

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

Approved:
SAVI PMC
5/31/2015

Aerospace Vehicle Systems Institute
127 TSHB - MS 3126
College Station, TX 77843-3126
Office: +1-979-845-5568
FAX: +1-979-339-4079
Web: www.avsi.aero

This page intentionally left blank.

Document Revisions

REV	DATE	Author(s)	Modifications	Approved
0.1	11/30/2013	DTW	Outline only; moved to new template	
0.2	2/15/2014	DTW	Reformatted to latest template; added information on TVPs, CM, and	
0.3	2/17/2014	DTW	Added material to Section 2	
0.3	2/19/2104	DTW	Completed Section 2	
0.4	2/27/2014	DTW	Added material to Section 3	
0.5	3/7/2014	MSK	Edits	
0.5	3/9/2014	DTW	Added material to Section 3.4 and 3.5	
0.5	3/12/2014	DTW	Added material to Section 4.1	
0.5	3/13/2014	DTW	Moved to folder for PMC Review	
1.0	3/31/2014	PMC	Moved to PMC Approved folder (Cannot find folder on Sharepoint)	PMC
1.1	5/9/2015	DTW	Removed placeholder for Section 4.4; WBS Supplier model not received from Brendan Hall	
1.1	5/31/2015	PMC	Approved removing placeholder	PMC

Contents

List of Figures

List of Tables

Executive Summary

This document summarizes the first year of development of a Virtual Integration Process (VIP) designed as a "game-changing" shift toward model-based systems engineering (MBSE) in the development of complex aerospace systems. The primary focus has been on producing and demonstrating an initial operating capability that uses the SAVI VIP to generate a semi-automated technique to carry out system safety analyses of the type done early in the development process (Preliminary System Safety Analysis – PSSA) during the time when system trade studies are active.

As the basis for this first operational capability for the VIP, the SAVI team has also produced two initial specifications, one for the VIP and one for its two major supporting infrastructure enablers, the SAVI Model Repository and the SAVI Data Exchange Layer (MR/DEL).

Demonstration of safety analysis methodology was based upon an expanded architecture-centric model for an aircraft wheel braking system (WBS), first suggested by the SAE S18 Committee [1] and broadened in scope by the SAVI team. The model set started with assumed aircraft requirements and hazard assessment, originally set down in SysML [2] and then transformed into AADL [3] using a translator that was previously developed for a DARPA requirement [4] but expanded to fit SAVI needs. Additionally, the model set was grown to include a solid geometry model [5] and an inter-model dependency tool (publisher/subscriber model) used to establish and visualize inter-model dependencies.

Applicable Documents

[1] SAE AIR6110, "Contiguous Aircraft/System Development Process Example," SAE International, December 2011.

[2] "OMG Systems Modeling Language (SysML)," Open Management Group (OMG), June 2012.

[3] SAE Aerospace Standard AS 5506A, "Architecture Analysis and Design Language," Society of Automotive Engineers, Revised January 2009.

[4] Cofer, D., "Guided Tour: Complexity-Reducing Design Patterns for Cyber Physical Systems," DARPA FA 8650-10-C-7081, September 30, 2011. http://cps-vo.org/node/2243/browser

[5] Valasek, J., Huber, B., and Husain, R., "Initial Development of Wheel Braking System Solid Geometry Models," SAVI_AFE61-03-004, Version 1.0, February 16, 2014.

[6] Joshi, A., Heimdahl, M. P. E., Miller, S. P., and Whalen, M. W., "Model-Based Safety Analysis," NASA CR-2006-213953, NASA Langley Research Center, February 2006.

[7] Haskins, C. (Editor), "INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities," INCOSE-TP-2003-002-03.2.2, International Council of System Engineering, 2011.

[8] ARP 4754A, "Guidelines for Development of Civil Aircraft and Systems," SAE International, December21, 2010.

[9] ARP 4761, "Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment," SAE International, December 1, 1996.

[10] Ward, D. T., Chilenski, J. J., Lewis, B. A., Mandalapu, S., Kerstetter, M. S., and Manners, R. E., "Terms of Reference," SAVI-AFE61-04-007, AVSI, February 14, 2014.

[11] Pollari, G. M., Chilenski, J. J., and vanHorn, S. B., "Model Repository/Data Exchange Layer Specification," SAVI-AFE61-02-001, AVSI, February 28, 2014.

[12] Ward, D. T., Chilenski, J. J., and vanHorn, S. B., "Rationale for SAVI Process Maturity Assessment," SAVI-AFE61-04-002, AVSI, February 28, 2014.

[13] Sadin, S. T.; Povinelli, F. P.; Rosen, R.; "NASA Technology Push Towards Future Space Mission Systems," Acta Aeronautica, Vol. 20, 1989, pp. 73 – 77.

[14] http://www.crescendo-fp7.eu/modules/download_gallery/dlc.php?file=42&id=1392309948

[15] Ward, D. T., Chilenski, J. J., and Manners, R. E., "Integrated Program Plan for Incremental Development," SAVI-AFE61-04-003, AVSI, April 10, 2014.

[16] Ward, D. T., Chilenski, J. J., Pollari, G. M., and vanHorn, S. B., "Summary of NAR Feedback," SAVI-AFE61-04-009, AVSI, January 13, 2014.

[17] Chilenski, J. J., Hall, B., Oliveira, F., "Model Set Defining AIR 6110 Braking System Example," SAVI-AFE61-03-001, AVSI, March 15, 2014.

[18] Delange, J., Feiler, P. H., and Lewis, B. A., "SEI Report on System Safety Analysis with AADL," SAVI-AFE61-03-005, AVSI, March 15, 2014.

[19] SAE Aerospace Standard AS 5506/3, "AADL Error-Model Annex," Society of Automotive Engineers, in revision as Document AS5506/3 2013.

[20] AC 23.1309-1E, "System Safety Analysis and Assessment for Part 23 Airplanes," Federal Aviation Administration, ACE-100, November 17, 2011.

[21] AC 25.1309-1A, "System Design and Analysis," Federal Aviation Administration, ANM-112, June 27, 1988.

[22] "SAVI AFE 59S1 Report, Summary Final Report", System Architecture Virtual Integration, AFE 59S1 Report SAVI-AFE59S1-08-002, Aerospace Vehicle Systems Institute, 31 August 2012.

1. Introduction

This document summarizes the development activities carried out by the SAVI program during 2013, with Authorization for Expenditure (AFE) 61 serving as the governing project direction.

1.1 Purpose

The purpose of this document is to summarize the results of efforts of the SAVI team during AFE 61, the first phase of operational development for SAVI. Concise task statements for this development include:

- Develop and release a formal specification for the VIP
- Develop and release a formal specification for the MR/DEL
- Demonstrate the SAVI VIP and tool chain for a selected version of the SAE AIR 6110 aircraft braking system safety analyses example
- Manage SAVI Version 1.0A Development

1.2 Assumptions and Constraints

1.2.1 Assumptions

The SAVI project team chose system safety analyses as the focus for the first capabilities to be incorporated into the Virtual Integration Process (VIP). This choice, made in May 2012 by the SAVI Program Management Committee (PMC), is quite logical, given the safety critical character of many elements that make a major aerospace system, like the commercial aircraft on which this group focused.

There were other compelling reasons that made the choice of system safety the PMC's highest priority. The SAE S-18 committee had recently circulated papers describing their proposed subsystem example [1], an aircraft wheel braking system (WBS). These papers included some proposed models for this WBS example [6], which meant that the SAVI team had a reasonable set of model initial conditions for their work on the virtual process. Moreover, the SAE committee was closely allied with several SAVI member companies and the nature of this project held considerable interest for both Original Equipment Manufacturers (OEMs) and suppliers alike. Regulatory agencies were also keenly interested in this kind of work to help answer certification questions raised by industry interest in model-based certification credits.

1.2.2 Constraints

The project team has been constrained throughout the VIP development time (since 2008) with a shortfall in the resources needed to complete needed research objectives.

2. SAVI VIP Version 1.0

Feasibility demonstrations were completed in 2012 and in 2013 the first phase of capability development for the SAVI Version 1.0 Virtual Integration Process (VIP) was completed. Version 1.0 of the VIP is being developed over a four-year period and is to provide capability aimed primarily at integration efforts for system integrators (OEMs) and first level (Tier 1) suppliers. Figure 1 illustrates the phasing for SAVI Version 1.0 development, which is currently set to take place over approximately four years' time (SAVI Versions 1.0A, 1.0B, 1.0C, and 1.0D). During this development period the number of participants is assumed to grow from 10 to 22 participants. Two additional tool vendor participants (TVPs) were added during the last quarter of 2013.

The three core tasks described in Section 1.0 provide the basis for developing Version 1.0 of the SAVI VIP over the period shown. This past year, 2013, the team has taken healthy steps forward, building upon solid results from the previously completed Proof of Concept phase of development.

2.1 Development Roadmap

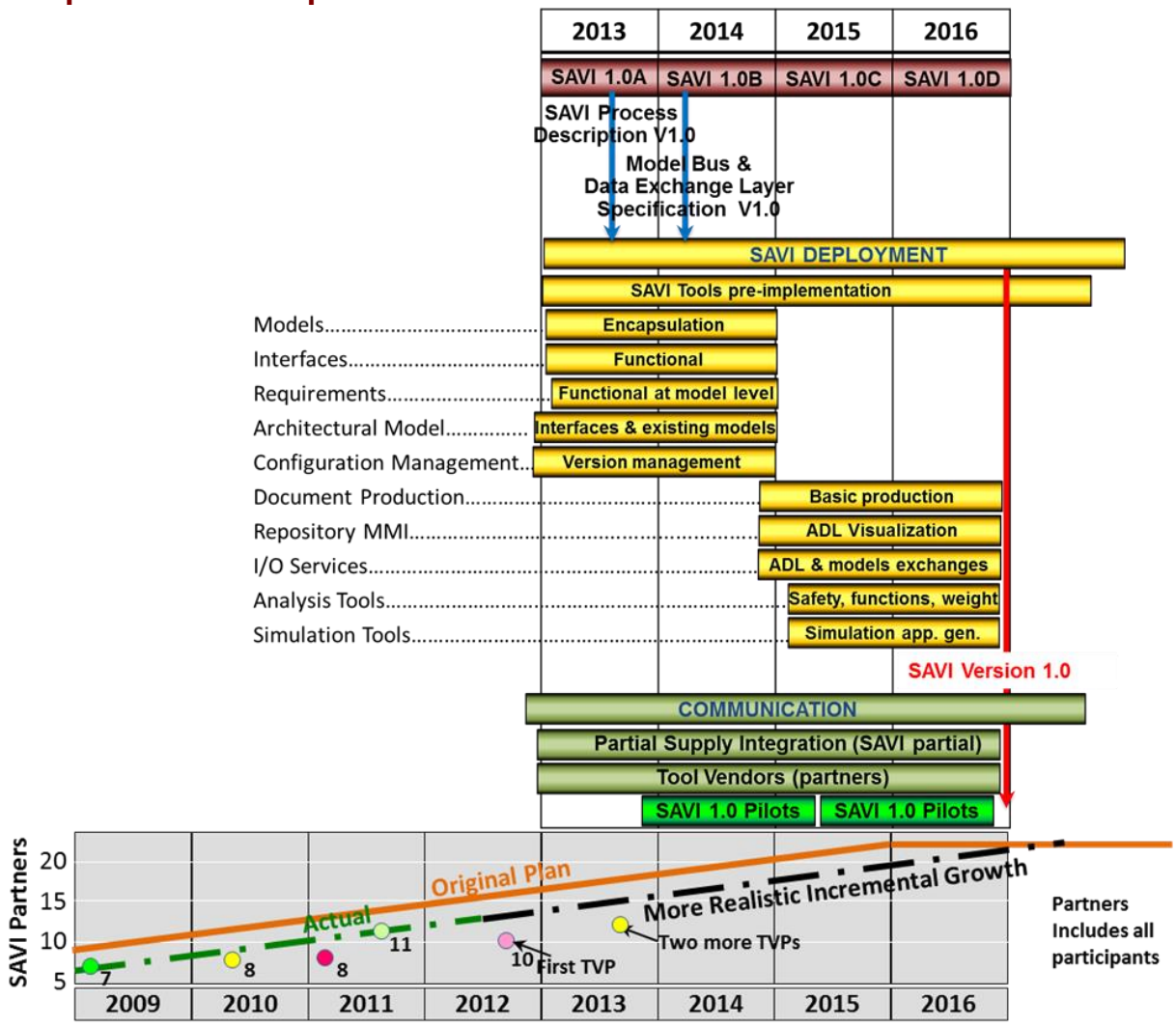

Figure 1. SAVI VIP Version 1.0 incremental development roadmap

2.2 SAVI VIP Version 1.0A Details

Carefully structured Use Cases designed to demonstrate SAVI's capabilities were utilized during both the Proof of Concept phase and during the first year of development of SAVI Version 1.0A. In the latter case these Use Cases were generated in SysML (using Enterprise Architect) to capture VIP requirements and to help generate the first formal SAVI specifications.

To help move the SAVI development in this direction, Figure 2 depicts the capabilities that exist and offers a simple visualization of additional capabilities that must be added. The legend explains the color coding and the graphic captures four classes of capabilities needed to develop an aircraft system along with a qualitative assessment of the current maturity of the VIP. The suggestion from this chart is that at this writing (February 2014) there is no more than about 30% of the total demonstration effort completed. To satisfy a potential SAVI user's concerns about maturity of the VIP, there is lot of work left to be done. Clearly, the Use Case methodology used during AFE 61 strongly influenced this "capability tree". The specification documents developed during this period also indicate that a solid foundation for the initial system safety analysis capability has been laid and that significant progress was made during 2013.

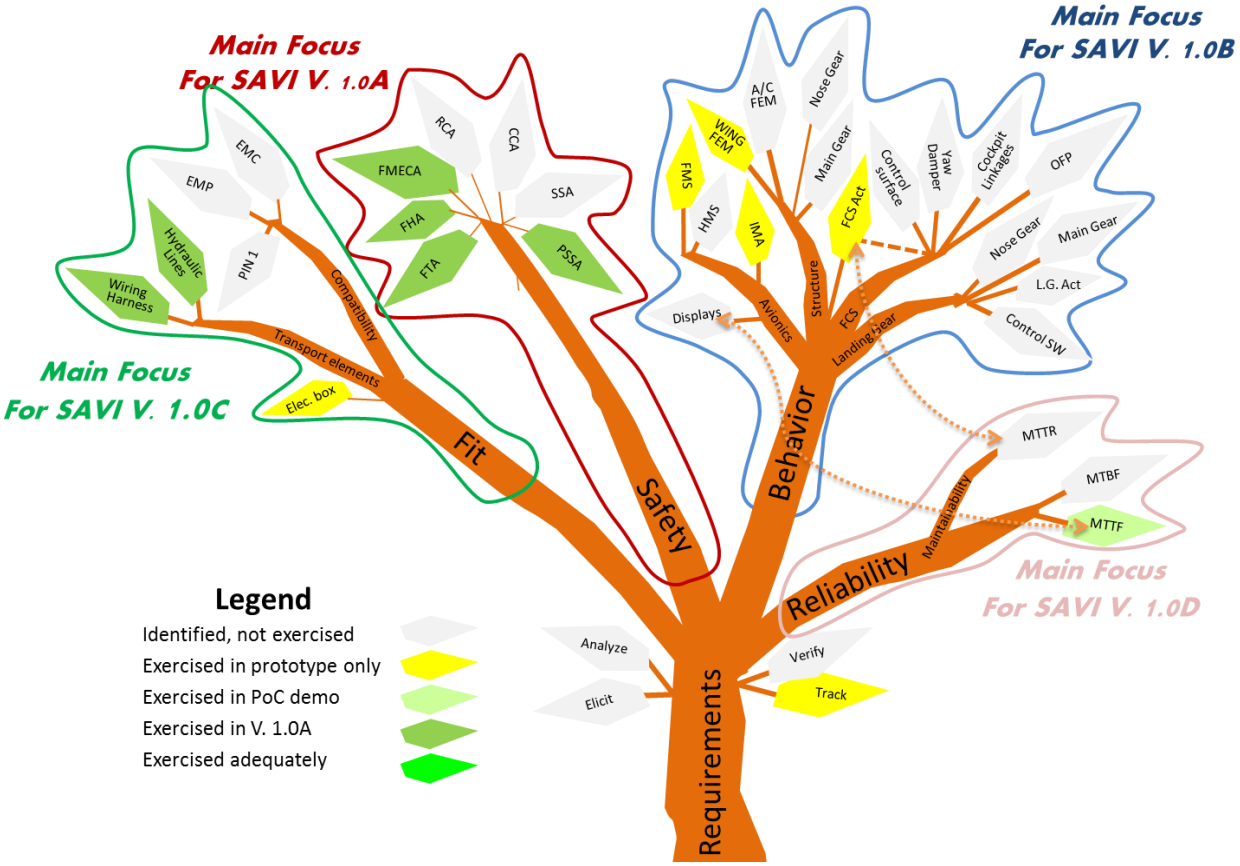

Figure 2. SAVI VIP capability growth tree

2.2.1 Tasks for AFE 61 (2013)

The primary tasks of SAVI Version 1.0A were:

- Set down formal specifications for both the VIP and for the MR/DEL;
- Detail how MR/DEL interfaces will accommodate necessary data transfers between System Integrator (SI) and suppliers, between suppliers, between architectural modelers and analysis domains, and exchanges with certification authorities;
- Implement a "single-truth" model for an aircraft braking system (based upon the SAE AIR 6110 template) and exercise it to carry out the set of safety analyses in that document. The braking system described in AIR 6110 was expanded to better match SAVI goals with components that have available and credible models folded into the example's scenarios.
- Exercise additional Use Cases to give more confidence in SAVI capabilities, carefully choosing projects to facilitate development. These projects may "shadow" real-world projects, but the priority is on involving all participants and in adding capabilities to encourage use of the VIP.
- Support development of a configuration management scheme to manage SAVI Versions;
- Devise and apply a scheme for evaluating the maturity of the SAVI VIP based upon principles used in the DoD Technology Readiness Levels approach to measuring technology maturity [14];

These tasks have been completed, with reports and demonstrations prepared detailing the results. This documentation will include both documents and video demonstrations. The complete list of reports is posted on the SAVI Sharepoint web site at: http://savi.avsi.aero/downloads/download.html. Video demonstrations, which can also be downloaded from that same location, illustrate the use of the MR/DEL, the conduct of consistency checks, and the exercise of SAVI principles in producing a Preliminary System Safety Analysis (PSSA) based on the expanded WBS model set generated by the SAVI team. An overall process description of the VIP as it was generated from the Use Case approach described above is also described in a video clip.

2.2.2 Deliverables

The primary deliverables for this development are:

- A specification document that spells out use cases, requirements governing, and activities completed during execution of the SAVI Virtual Integration Process (VIP).
- A specification document that sets down use cases, requirements for, types of tools to be used, and standards to be adhered to in the processing, storage, and handling of data during execution of the VIP.
- A set of demonstrations that illustrate how the VIP can be executed to produce (in at least a semi-automated fashion) the artifacts essential to the Preliminary Systems Safety Assessment (PSSA) as spelled out in current system development and system safety documents [8, 9].

2.3 VIP Specification

The VIP specification formally sets down what is expected in a SAVI-compliant process. It starts with requirements for a virtual integration that meets the objectives deemed necessary and appropriate to develop an aerospace system using an architecture-centric model set as the basis for system trade studies and for repetitively carrying out consistency checks at each iteration in the architectural design and subsequent model-base system engineering processes.

2.3.1 Domain Model

Delineation of the specification began with laying out a domain model to guide the selection of use cases and, flowing from these use cases, the requirements for the VIP. This domain model was a composite of inputs from all team members, using brainstorming techniques and mind mapping software to create Figure 3. The domain model serves a collection point for what is to be included in the VIP and therefore guides the specification of that process. As is suggested by the yellow comment block in Figure 3, there are still open questions about what should and what should not be included in this guideline. As additional capabilities are added to the SAVI VIP, the domain model is likely to become more complex and harder to interpret. Nonetheless, the thought processes generating this overview of the VIP structure and content are important and are not duplicated elsewhere.

2.3.2 Use Cases

The next step in generating this specification was to select use cases essential to the capability being added to the VIP. Since this was the first operational capability and since it is focused on system safety analysis at the early stages of the development cycle, the use cases chosen were centered on use of the VIP to set up a system architecture that allows interaction between all members of the development team and facilitates trade studies to select appropriate elements. In one sense our objectives were to first be sure that the VIP promoted and made full use of models to systematically analyze the architectural iterations during the Request for Information (RFI) and Request for Proposal (RFP) stages of system development. In terms of processes described and defined in the INCOSE Systems Engineering Handbook [7], these activities would fall under the Stakeholder Requirements Definition Process, the Requirements Analysis Process, and the Architectural Design Process. This portion of the system life cycle is usually recognized as a critical phase when requirements are generated, refined, and captured. It is generally agreed that requirements definition is where most system anomalies or defects are introduced and that a large number of them go undetected until later phases of development. Most important to SAVI's objectives, though, detecting and correcting defects at this early stage rather than later during verification and validation testing is crucial to success or failure. Detection and correction of anomalies with the use of models and architectural analyses can be orders of magnitude less expensive and less time-consuming when the detection is made in the early life of a system.

With this motivation, the SAVI team set down four use cases that drive the VIP toward early detection of requirements errors and one use case that applies system safety analyses to a proposed system architecture.

- Develop Proposed Architecture
- Obtain Proposed Architectures
- Perform Model Change Impact Analysis

- Perform Consistency Checks
- Perform Proposed System PSSA

These five use cases then lead logically to the five requirements spelled for this first version of the VIP. Note that the first four of these are the requirements for the VIP in general and the last one is related to the demonstrations generated to illustrate VIP capabilities under AFE 61. Of course, this fifth use case is also generally necessary for systems development, but it is constrained to some extent to what the team did in 2013.

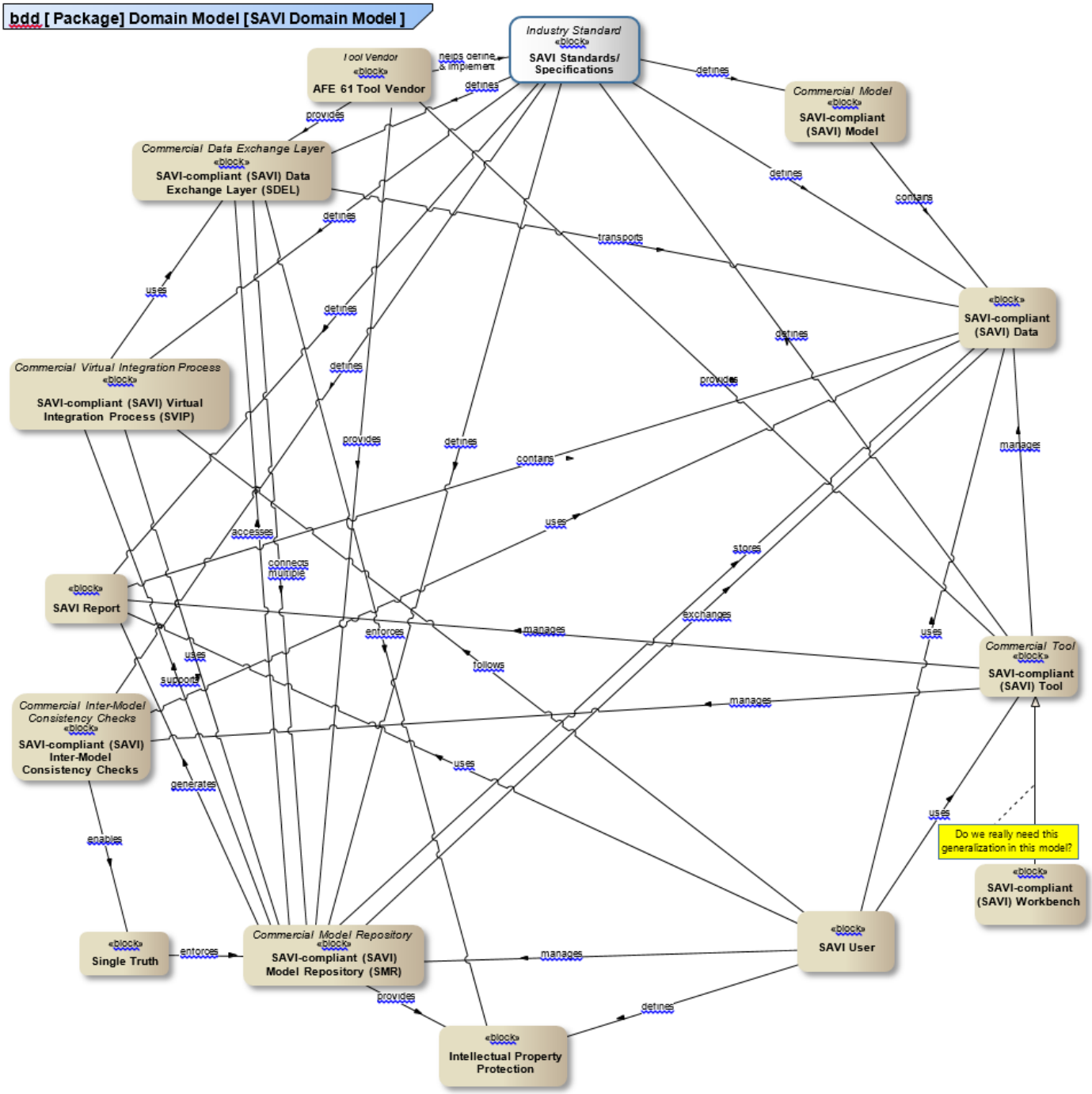

Figure 3. Domain model for SAVI VIP Version 1.0

2.3.3 VIP Requirements

The use cases summarized in the previous section led to five requirements for the SAVI VIP. The specific steps to be followed in meeting these requirements are detailed in the SAVI VIP Specification document in the form of complete use cases with multiple branches where needed.

2.3.3.1 Proposed Architecture Development [VIP_001]

Description	Traces
The VIP shall define a process for using SAVI-compliant Models, Model Repository, Data Exchange Layer, and Tools such that models may be used as part of both the request and the response of a RFP process.	UseCase: Develop Proposed Architecture

2.3.3.2 Model Change Impact Analysis [VIP_002]

Description	Traces
The VIP shall define a process for using SAVI-compliant Models, Model Repository, Data Exchange Layer, and Tools to perform a model change impact analysis.	UseCase: Perform Model Change Impact Analysis

2.3.3.3 Model Consistency Checks [VIP_004]

Description	Traces
The VIP shall define a process for using SAVI-compliant Models, Model Repository, Data Exchange Layer, and Tools to perform model consistency checks.	UseCase: Perform Consistency Checks

2.3.3.4 Obtain Proposed Architectures [VIP_005]

Description	Traces
The VIP shall define a process for using SAVI-compliant Models, Model Repository, Data Exchange Layer, and Tools to support OEMs obtaining proposed architectures from Suppliers.	UseCase: Obtain Proposed Architectures

2.3.3.4 Conduct System-Level PSSA [VIP_003]

Description	Traces
The VIP shall define a process for using SAVI-compliant Models, Model Repository, Data Exchange Layer, and Tools to perform a system-level Preliminary System Safety Assessment (PSSA).	UseCase: Perform Proposed System PSSA

2.4 Model Repository/Data Exchange Layer (MR/DEL) Specification

The SAVI MR/DEL specification spells out high level requirements for the SAVI Model Repository and Data Exchange Layer in executing the SAVI Virtual Integration Process (VIP). The VIP applies to a model set consisting of models from different domains, typically written in different modeling languages, having different data representations, and accessed with tools that may not inherently share data. But the SAVI VIP must ensure model consistency across this model set so that shared properties and dependencies have no contradictions – that is, the VIP must ensure consistency. The underlying information technology framework (MR and DEL) must support this single truth concept across the model set so that it starts integrated and stays integrated.

The high level requirements in 2.4.2 must be further decomposed and expanded to create testable, verifiable detailed requirements for an MR/DEL implementation during a given system development.

2.4.1 Components of SAVI Data Management

The two critical data management components are:

- Model Repository: a data structure needed for information storage and analysis of the reference model. It can also been defined as a container or place in which things (models) can be stored for safety [10].
- Data Exchange Layer (DEL): the set of interfaces that allows data transfer between the elements and components of the SAVI repository structure and the various domain analysis tools [3]. A DEL can consist of data translators, data models, data file specifications, data schema, tools and processes for transporting and linking data and metadata.

The SAVI team has often used the graphical depiction in Figure 4 to describe the functions of the MR/DEL and to distinguish between these components.

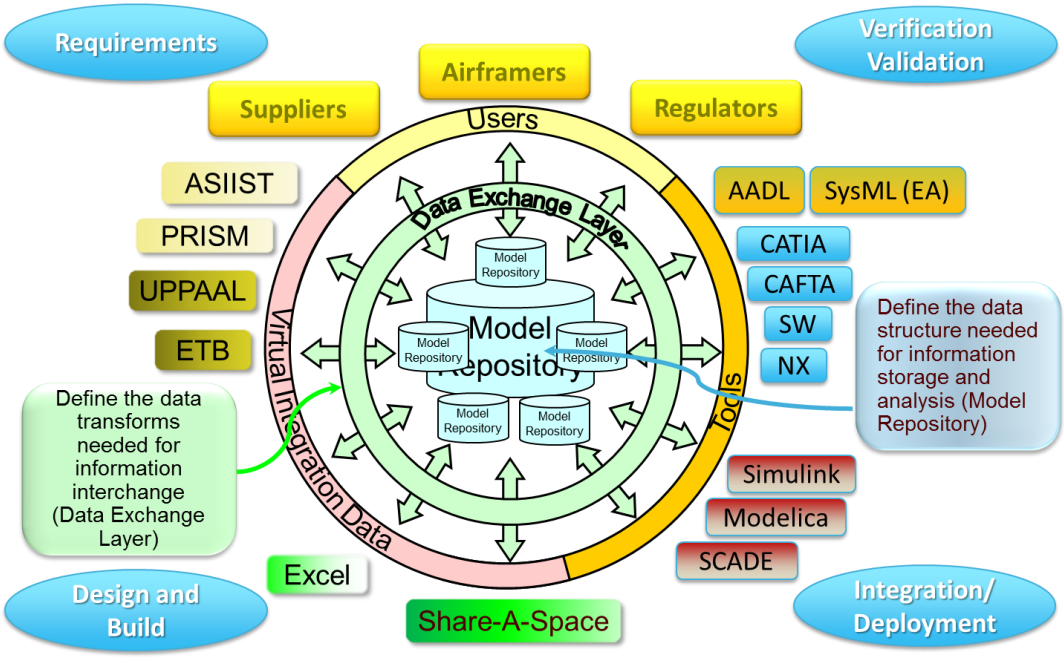

Figure 4. Relationship of MR/DEL to SAVI VIP

2.4.2 MR/DEL Requirements

2.4.2.1 Protect Intellectual Property [MR/DEL_001]

Description
The SAVI MR/DEL shall protect data Intellectual Property rights of the data owners.

2.4.2.2 Control Data Access [MR/DEL_002]

Description
Data owners shall be able to control access rights to all items residing in the SAVI MR/DEL (models, documents, controls, etc.) at the data object level.

2.4.2.3 Maintain Process Neutrality [MR/EL_003]

Description
The SAVI MR/DEL shall not be dependent on product development, configuration management, or other business processes.

2.4.2.4 Maintain Information Technology Independence [MR/DEL_004]

Description
The SAVI MR/DEL shall not be dependent on any one Information Technology (IT) implementation or infrastructure and shall work with existing IT technology and infrastructure.

2.4.2.5 Base on Recognized Standards [MR/EL_005]

Description
The SAVI MR/DEL shall utilize recognized data interoperability standards.

2.4.2.6 Provide Clear Ownership [MR/DEL_006]

Description
The SAVI MR/DEL shall permit data ownership identification at the data object level.

2.4.2.7 Be Auditable [MR/EL_007]

Description
The SAVI MR/DEL shall support auditing for compliance with the SAVI VIP and this specification and to support business processes.

2.4.2.8 Provide Secure Access [MR/DEL_008]

Description
The SAVI MR/DEL shall provide a means to control access at the data object level.

2.4.2.9 Allow for Flexible Content [MR/EL_009]

Description
The SAVI MR/DEL shall accommodate data and metadata associated with a model set defined by the business rules governing the MR/DEL.

2.4.2.10 Leverage Architecture Models [MR/DEL_010]

Description
The SAVI MR/DEL shall provide a means to provide access to all data and metadata for all architecture models used in the SAVI VIP.

2.4.2.11 Accommodate Existing Tools [MR/DEL_011]

Description
The SAVI MR/DEL shall accommodate the data and data formats from existing tools used for product definition and development.

2.4.2.12 Work with Existing Tools [MR/DEL_012]

Description
The SAVI MR/DEL shall work with existing data repositories.

2.4.2.13 Work with Existing Configuration Management Tools and Processes [MR/DEL_013]

Description
The SAVI MR/DEL shall work with existing configuration management tools and processes.

2.4.2.14 Provide Version Control [MR/DEL_014]

Description
The SAVI MR/DEL shall provide version control at the data object level.

2.4.2.15 Provide Export Control [MR/DEL_015]

Description
The SAVI MR/DEL shall provide a means to manage data in accordance with export control laws.

2.4.2.16 Establish Data Ownership [MR/DEL_016]

Description
Data ownership shall be established at the data object level.

2.4.2.17 Exchange Data across a Multi-Tiered Supply Chain [MR/DEL_017]

Description
The SAVI MR/DEL shall support data exchange between supply chain organizations.

2.4.2.18 Be Logically and Physically Distributed [MR/DEL_018]

Description
The SAVI MR/DEL shall support logically and physically separated data repositories.

2.4.2.19 Track the Relationship of Managed Objects [MR/DEL_019]

Description
The SAVI MR/DEL shall provide a means to track the relationship of managed data objects in the SAVI model set.

2.4.2.20 Maintain Synchronization [MR/DEL_020]

Description
The SAVI MR/DEL shall support synchronization (consistency) between data in different repositories.

2.4.2.21 Be Scalable to Large Enterprises and Supply Chains [MR/DEL_021]

Description
The SAVI MR/DEL shall accommodate large enterprises and supply chains of the type associated with air transport class airplane development.

2.4.2.22 Support Users Who are not Always Online [MR/DEL_022]

Description
The SAVI MR/DEL shall support asynchronous usage (users who are not always connected or connected at the same time as other users).

2.4.2.23 Support Asynchronous Model Updates [MR/DEL_023]

Description
The SAVI MR/DEL shall support asynchronous model set updates.

2.4.2.24 Span Lifecycle Support [MR/DEL_024]

Description
The SAVI MR/DEL shall support model set data that spans the lifecycle of an air transport class airplane.

2.4.2.25 Identify User Roles and Grant Privileges [MR/DEL_025]

Description
The SAVI MR/DEL shall provide a means to identify user roles and grant privileges according to those roles.

2.4.2.26 Support SAVI Consistency Checking [MR/DEL_026]

Description
The SAVI MR/DEL shall support consistency checking of the model set, including (detailed in [11]): • Intra-model consistency • Consistency across different models of same system • Architectural consistency • Configuration consistency

2.5 Infrastructure and Enabling Activities

This section of the report summarizes enabling activities essential to the objectives of AFE 61 and describes the five most important infrastructure activities that supported development of SAVI Version 1.0A during 2013.

2.5.1. Tool Vendor Activities

This year was the first year of active involvement of Tool Vendor Partners (TVPs) in SAVI development. The SAVI team refined the tool vendor procedures [10] and set up a basic structure for providing value to both the TVPs and to SAVI. Three organizations signed TVP agreements and actively worked with the SAVI team during this first year. The activities of each TVP during this phase of SAVI development are described, ranging from attending face-to-face team meetings, participating in weekly teleconferences, presenting SAVI Seminars, and providing training in subjects important to the SAVI team. The planned activities next year for each TVP are also being addressed now, but most of those activities are still being discussed by the SAVI Program Management Committee (PMC) and individual TVPs.

The three TVPs joining SAVI during 2013 were Esterel Technologies, Adventium Labs, and Eurostep Group. Esterel Technologies, which became part of ANSYS shortly after they agreed to join the SAVI team, is a company with a strong history of integrating software and bringing together domain specialists. Their SCADE suite of tools has been widely used in Europe as a tool addressing behavioral dynamics of physical systems. Adventium Labs is a small company with very solid credentials in fusing together many different disciplines and in wrapping architectural analysis around different domains. Their work with DARPA (the FUSED approach) fit perfectly with the VIP that the SAVI team is pulling together. Finally, the Eurostep Group has expertise in applying standardized data management to system development; they are particularly strong in both tools (e.g., Share-A-space™) and experience in assuring efficient shared data flow within multi-tiered development environment. They are one of the managers of the STEP AP239 libraries that offer an ISO standard for data handling in the heterogeneous and distributed setting in which most aerospace systems are developed today. All three TVPs are expected to continue in similar roles next year.

2.5.2. Configuration Management Plan

The System Architecture Virtual Integration (SAVI) Configuration Management Plan (CMP) concisely describes a highly tailored approach to maintaining control of four key Configuration Items (CIs) for the SAVI Virtual Integration Process (VIP). A Configuration Control Board (CCB) made up of Program Management Committee (PMC) members and chaired by the Principal Investigator is the controlling body. The Configuration Items (CIs) of interest are:

- Two primary specifications (for the VIP and for the SAVI MR/DEL), which describe what a development project must do to be SAVI-compliant;
- SAVI Data Exchange (DEX) description(s) that spell out data flow within the VIP; and
- Definitions for and application of consistency checking.

This plan is specifically drafted to cover SAVI Version 1.0; but it is a first iteration, currently covering only elements of the SAVI Virtual Integration Process as developed during AFE 61. No formal audits are planned for this rather simple CMP but the PMC will examine and approve all changes to be carried out, since all voting members are also members of the CCB. As SAVI Version 1.0 evolves over the planned three-year development cycle, this plan will be expanded to include additional modeling and a more complete model-based integration process.

2.5.3. Process Maturity Assessment

Reference 12 lays out the rationale for and illustrates the use of a qualitative assessment methodology used to evaluate maturity of a process, specifically the Virtual Integration Process (VIP). The process chosen to assess maturity of the VIP is based upon qualitative assessment tools used by the DoD and by industry with modifications to the wording based upon new overlays tailored for the Virtual Integration Process. The assessment levels, called Process Maturity Levels (PMLs), closely follow the concept of Technology Readiness Levels (TRLs) first proposed by NASA [13] and later modified by Miller [14].

The first two assessments were made by three different SAVI team members, resulting in qualitative scores of an average of 4.0 to 4.5 for the Version 1.0A system safety process overlay set only. Other portions of the SAVI VIP were evaluated as having a process maturity level of about 3.0 to 3.5, based upon Proof of Concept work done earlier in the SAVI development.

The following recommendations followed from the effort to initiate this PML approach to VIP maturity assessment.

(1) Modify the USAF TML calculator with the file containing the overlays presented [12].

(2) Periodically (at least once a year) review the questions for those maturity levels that SAVI is approaching for continued relevance and update them as agreed upon by the SAVI PMC.

(3) Conduct a minimum of two assessments during each phase of development for SAVI Versions 1.0A, 1.0B, 1.0C, and 1.0D with at least 3 members of the SAVI team evaluating maturity levels at each review.

2.5.4. Collaboration Initiatives

A fundamental tenet of the SAVI approach is that there are many other useful concepts applicable to virtual integration that the SAVI will not generate. This assumption led to seeking “open” development where feasible within the cooperative framework. Moreover, this team spent considerable effort remaining abreast of model-based systems engineering (MBSE) efforts worldwide. The following table suggest six such efforts that are contributing to the VIP’s development. In each case, SAVI has consciously sought out representatives from each organization and held seminars, joint sessions, and discussions with the principals.

The last of these organizations is currently providing the DEXs developed for CRESCENDO [14] as a starting point for the SAVI DEX. One of the tasks for 2014 is to modify the DEXs provided from CRESCENDO and utilize that modification as the basic DEX for SAVI Version 1.0. The SAVI team adopted this development approach at the closeout PMC meeting in Seattle in December 2013.

Table 1. Related MBSE efforts

Title	Organization	Remarks
ASSERT	European Space Agency (ESA)	ftp://ftp.cordis.europa.eu/pub/ist/docs/dir_c/ems/assert-fs_en.pdf 7 http://www.dit.upm.es/~str/proyectos/assert/
AVM (META)	DARPA	http://cps-vo.org/group/avm/meta/search
SLIMS	InterCAX	http://www.intercax.com/products/slim/
CRESCENDO	European Framework Program	http://www.crescendo-fp7.eu/
TOICA	European Framework Program	http://www.toica-fp7.eu/
MoSSEC	Aerospace and Defence Industries Association of Europe (ASD-SSG)	http://www.asd-ssg.org/simulation-interoperability http://www.crescendo-fp7.eu/modules/download_gallery/dlc.php?file=42&id=1392309948

2.5.5. Necessity for Growth in SAVI Participation

One of the main recommendations for near term progress in developing the SAVI VIP is that the SAVI team concentrates on attracting more participation in the project [15]. Concern with lack of resources has been emphasized repeatedly in briefings and reports by project management, especially after being redirected to carry out SAVI development in an incremental fashion at the beginning of 2009 (AFE 59). This need was also highlighted again in the report [16] from members of the Non-Advocate Review (NAR) conducted on November 18, 2013. Several of the reviewers raised questions like: "What is the minimal number of organizations/people to build a critical mass to make a difference in the very competitive area?" Comparing the needed growth rate of the SAVI team (Figure 1) and the actual growth rate, suggests that the limitation on resources will likely continue to be a major challenge for the project.

3. Model Set

3.1 Requirements Model

The Requirements Model defines a representative subset of the functional and safety requirements for the WBS. The following types of requirements were extracted from the AIR 6110 document (including traceability information):

- Aircraft Requirements
- Aircraft FHA (Functional Hazard Assessment)
- Aircraft Functional Allocation Requirements
- Safety Derived Requirements from PASA/Aircraft FHA
- Initial Wheel Brake System Requirements
- Braking System Specification Requirements
- WBS Requirements
- BSCU Subsystem Requirements
- Aircraft Level Braking System Safety Derived Requirements

This requirements model traces from aircraft requirements through high level system requirements down to system requirements allocated to system items.

A Microsoft Excel spreadsheet captured these requirements and imported them into Share-A-space. Share-A-space shows requirements traceability between requirements, to reference documents and to design configuration items that satisfy the requirements. Figure 5 is an excerpt from the requirements model.

Requirement ID (Mandatory)	Parent ID (Optional)	Traced From (Optional)	Short name (Mandatory)	Textual definition (Mandatory)	Version (Mandatory)	Type (Optional)	Criticality (Optional)	Source Document (Optional)
S18-ACFT-R-0009			Aircraft shall have a means to decelerate on the ground in accordance with 14CFR 25.735		v001			14 CFR Part 25
S18-ACFT-R-0010					v001			
S18-ACFT-R-0110			Aircraft shall have autobrake function		v001			Business Case Trade Study
S18-ACFT-R-0135			Aircraft shall provide an anti-Skid function		v001			Business Case Trade Study
S18-ACFT-R-0184			Aircraft shall have hydraul-ically-driven brake function		v001			Hydraulic-Electric Braking Trade Study
S18-ACFT-R-0185			The pilot shall be allowed to override the autobrake function		v001			14 CFR 25.73(c)(2)
S18-ACFT-R-0835			Notification of failure (aircraft level)	The S18 aircraft shall provide Flight Crew notifi-cation for failure conditions which could result in a runway excursion (loss of directional control, loss of speed control, loss of directional control, or asymmetric loss of directional control)	v001	Safety		Aircraft FHA Assumption
S18-ACFT-R-0933			DAL Level for aircraft deceleration function	The function responsible for 'decelerate aircraft on the ground' shall be developed using a functional development assurance level of A as described in SAE ARP 4754A/ED-79A	v001	Safety		FHA
S18-ACFT-R-1322			Functional independence between brakes and reverse thrust	There shall be functional independence between brakes and reverse thrust functions	v001	Safety		PASA Independence requirement

Figure 5. Requirements model excerpt

3.2 Publisher/Subscriber Model

Table 2. Sample component identification table

ATA	Name
z24-xx-101	Elec. Pwr. Sys. - L
z24-xx-102	Elec. Pwr. Dist. Unit - Wheel Well - L
z24-xx-201	Elec. Pwr. Sys. - R
z24-xx-202	Elec. Pwr. Dist. Unit - Wheel Well - R
z27-xx-101	Rudder Pedal Assembly - L
z27-xx-104	Rudder Pedal Rudder Position Sensor - L
z27-xx-201	Rudder Pedal Assembly - R
z27-xx-204	Rudder Pedal Rudder Position Sensor - R
z29-xx-101	Hyd. Pwr. Sys. - L
z29-xx-102	HPS - L Isolation Valve - L
z29-xx-103	Selector Valve - L
z29-xx-104	Accumulator - L

The Publisher/Subscriber (Pub/Sub) model defines the major functional components of the WBS and the interconnections between them. This model was implemented with two spreadsheets and a drawing. The first spreadsheet (Table 2) identifies all components in the system of interest. The second column gives the name of the major functional component. The first column gives a unique identifier, based on ATA chapters, for the major functional component (z24-xx-101). This identifier is traditionally used to track equivalence of components across the different documents and models used to produce a system.

Table 3. Sample component interconnections table

Publisher ATA	Publisher Name	Connection	Signal	Subscriber ATA	Subscriber Name	Notes
z24-xx-101	Elec. Pwr. Sys. - L	z24-xx-101_z24-xx-102	Primary Power	z24-xx-102	Elec. Pwr. Dist. Unit - Wheel Well - L	
z24-xx-101	Elec. Pwr. Sys. - L	z24-xx-101_z24-xx-202	Secondary Power	z24-xx-202	Elec. Pwr. Dist. Unit - Wheel Well - R	
z24-xx-101	Elec. Pwr. Sys. - L	z24-xx-101_z27-xx-104	Main Power	z27-xx-104	Rudder Pedal Rudder Position Sensor - L	
z27-xx-101	Rudder Pedal Assembly - L	z27-xx-101_z29-xx-110	Mechanical Power	z29-xx-110	Manual Meter Valve - R Inboard	
z27-xx-101	Rudder Pedal Assembly - L	z27-xx-101_z29-xx-206	Mechanical Power	z29-xx-206	Manual Meter Valve - R Outboard	
z29-xx-101	Hyd. Pwr. Sys. - L	z29-xx-101_z29-xx-102	Hyd. Power (Pressure)	z29-xx-102	HPS - L Isolation Valve - L	
z29-xx-103	Selector Valve - L	z29-xx-103_z29-xx-104	Hyd. Power (Pressure)	z29-xx-104	Accumulator - L	Bi
z29-xx-105	Meter Valve - L Inboard	z29-xx-105_z29-xx-101	Hyd. Power (Return)	z29-xx-101	Hyd. Pwr. Sys. - L	
z32-xx-101	BSCU - L	z32-xx-101_z29-xx-205	Meter Valve – R Outboard Command	z29-xx-205	Meter Valve - R Outboard	
z32-xx-101	BSCU - L	z32-xx-101_z29-xx-209	Meter Valve – L Outboard Command	z29-xx-209	Meter Valve - L Outboard	
z32-xx-109	Weight-On-Wheels Sensor - L	z32-xx-109_z32-xx-101	Weight-On-Weels Sensor – L Reading	z32-xx-101	BSCU - L	
z32-xx-109	Weight-On-Wheels Sensor - L	z32-xx-109_z32-xx-201	Weight-On-Weels Sensor – L Reading	z32-xx-201	BSCU - R	

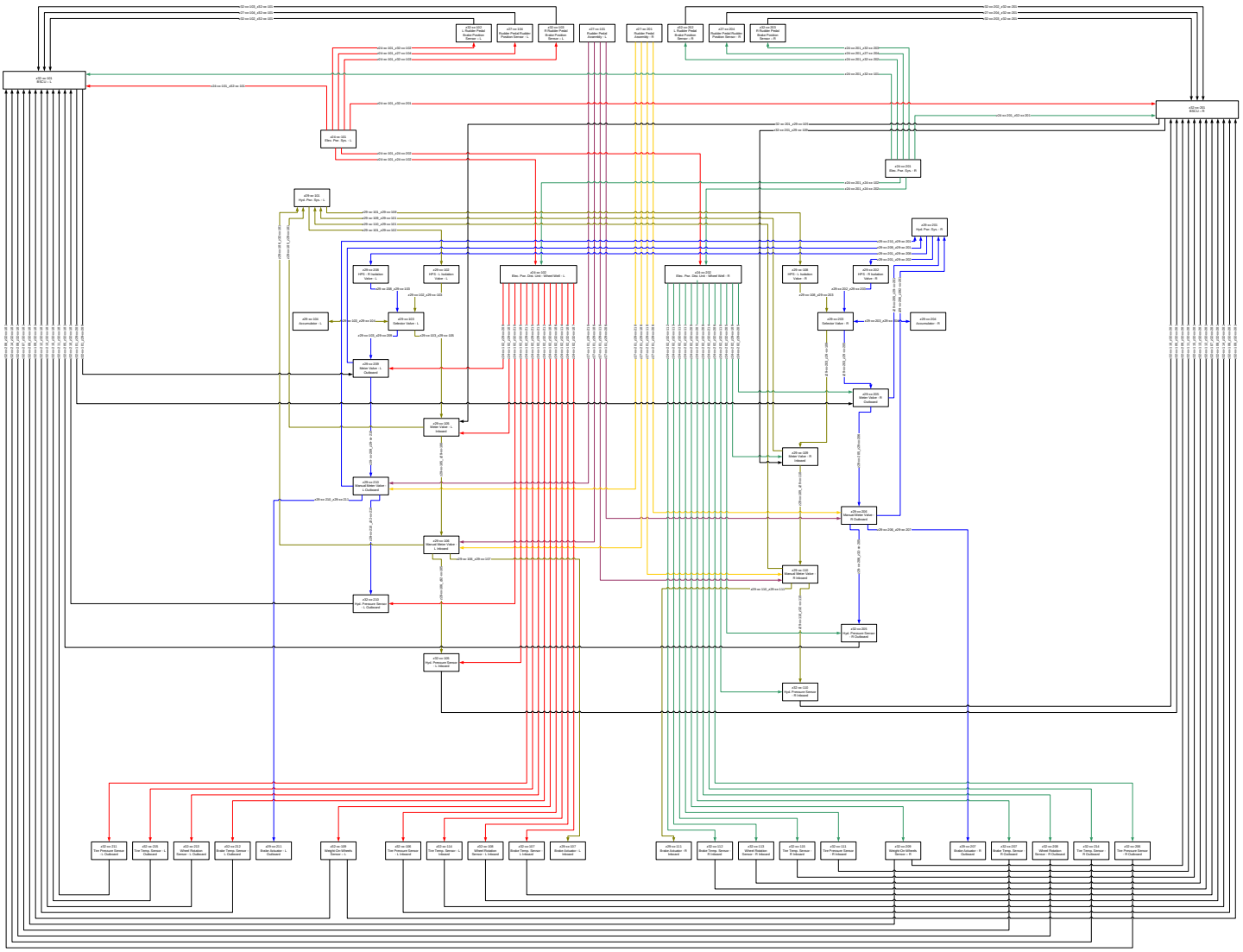

Figure 6. Publisher/Subscriber model drawing for WBS

The interfacing connections between all components are depicted in two ways: with a tabular matrix and with a graphical depiction (Figure 6). A second spreadsheet (Table 3) has seven columns tabulating the interfaces. The first two columns identify the unique identifier and name of a major functional component that is “publishing”, or providing, a “signal”. The fifth and sixth columns identify the unique identifier and name of a major functional component that is “subscribing” to, or using that, “signal”. The seventh column identifies those

special cases where the transfer between the two components is bi-directional or not. The fourth column identifies the "signal" type. The types used in this project include both power (electrical, hydraulic, and mechanical) and signal (input to an actuator or readings from a sensor). The third column provides a unique identifier for the connection. This identifier is formed by concatenating the subscriber's unique identifier to the publisher's unique identifier separated by an underscore. Each row in the table identifies a unique interconnection. If a publisher (for example, Electrical Power System – Left) has seven subscribers for a signal (that is, Primary Power), then there will be seven rows in the table, one for each interconnect

Figure 6 graphically depicts the major functional components and their interconnections, showing the publisher/subscriber model for the WBS. The major functional components are shown as boxes with the unique identifier and component name in the box. Interconnections are shown as color-coded lines with the interconnection identifier on the line. The purple and gold lines running down the center represent mechanical power. The red and green lines represent electrical power. The sea blue and olive green lines represent hydraulic power. Finally, the black lines represent actuator commands and sensor readings.

3.3 Two-Way Translation between SysML and AADL Models

While SysML is a rich and flexible modeling language, it lacks semantic precision which is a prerequisite for applying formal analysis methods (to mathematically prove characteristics of a system). On the other hand, AADL is intentionally built with semantics suitable for application of formal methods. Together, SysML and AADL provide a powerful combination for architecture modeling. The nature of this complementary role is shown in Figure 7 where SysML captures the logical and physical architecture views and where AADL lays the foundation for formal methods analyses. But an initial system architecture model of the Wheel Braking System (WBS) was created in SysML (Figure 8) using Enterprise Architect, version 10 before building the architectural model shown in Figure 7. Notice that both software and hardware are developed in an integrated fashion.

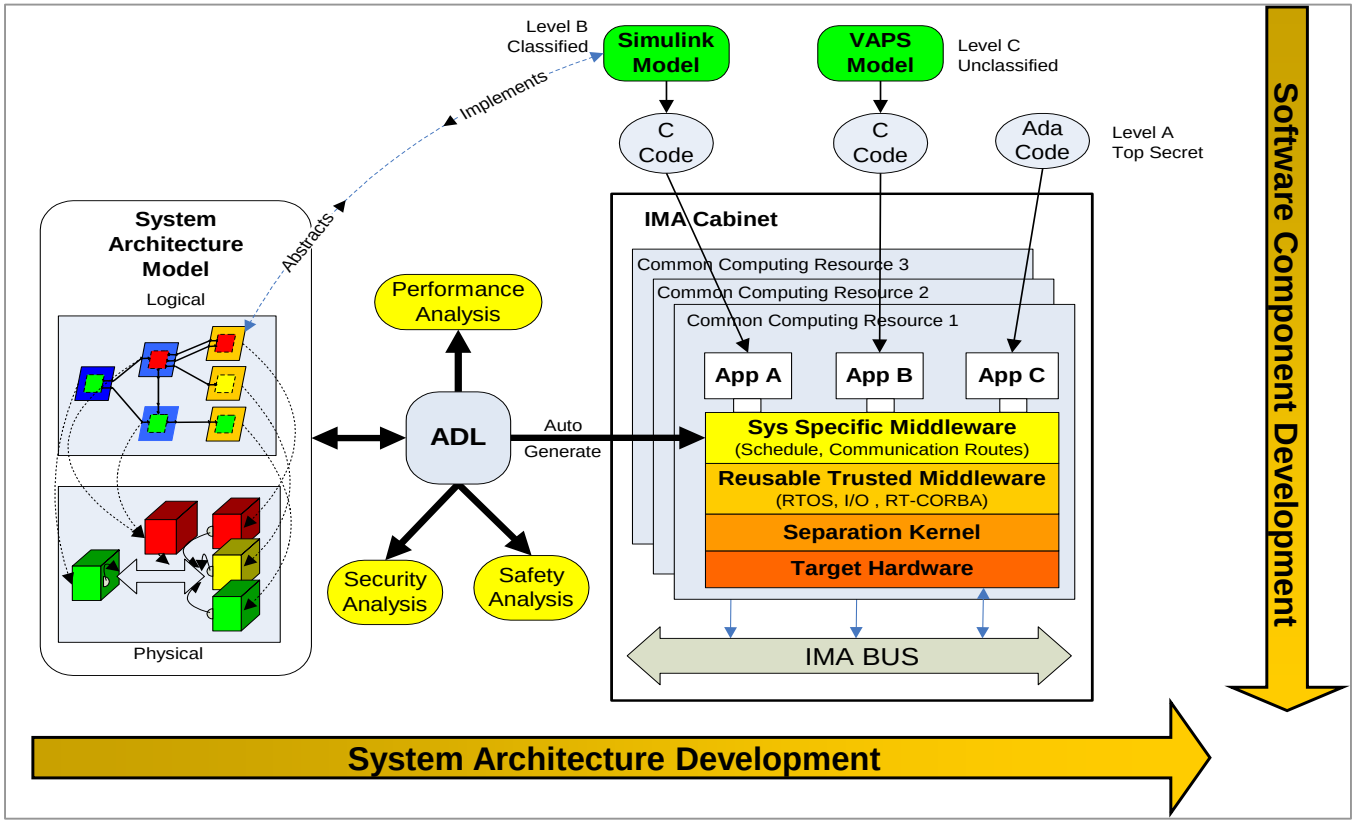

Figure 7. SysML and AADL complementary roles

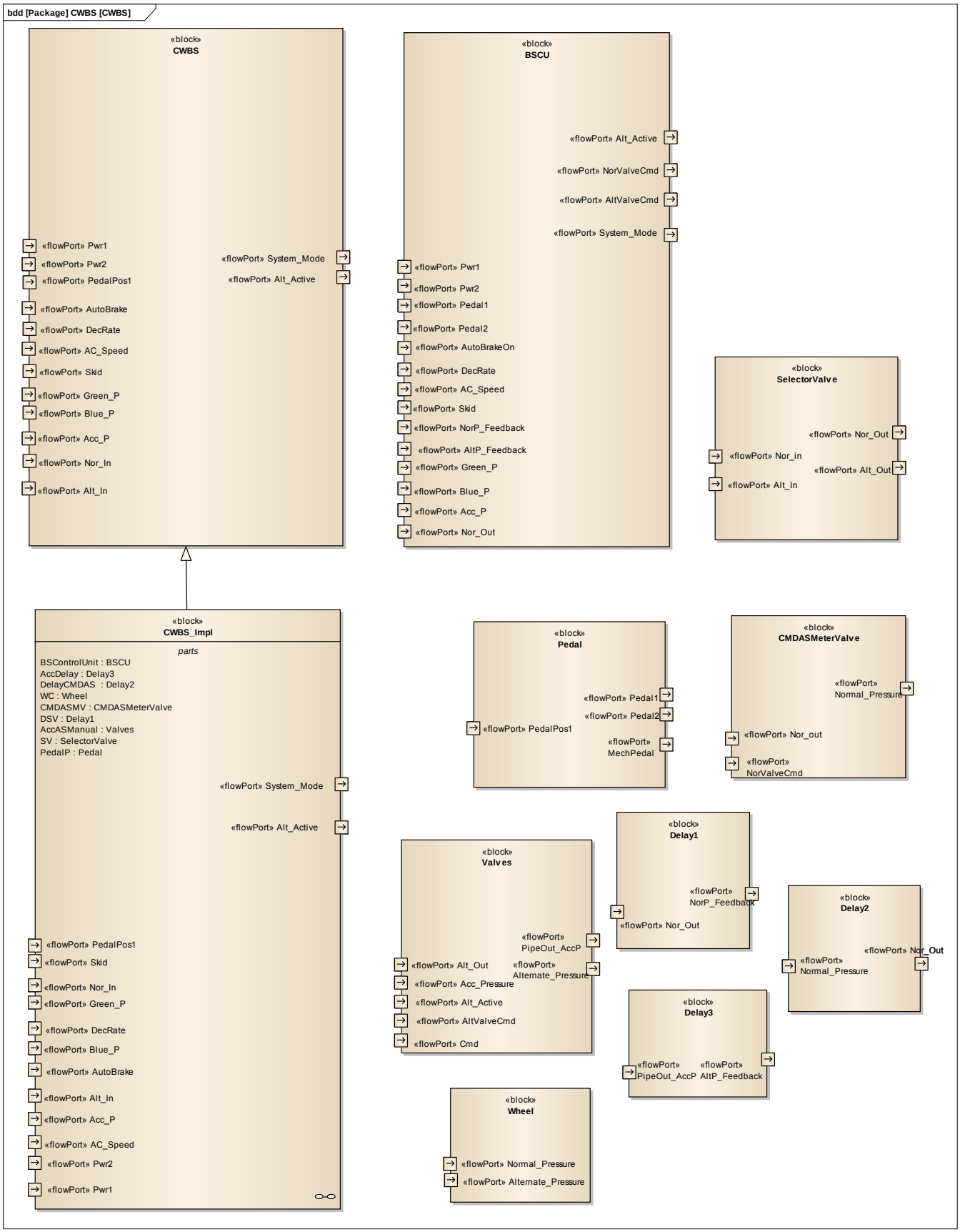

Figure 8. WBS SysML architecture elements

In order to leverage these languages, an AADL profile for SysML was developed during AFE 61. This profile was implemented in Enterprise Architect 10.0 (tool palette shown in Figure 9) so that SysML models could be constructed which would then be translated to AADL using an enhanced version of the DARPA META translator. This enhanced DARPA META SysML-AADL translator was derived from the DARPA META program. It is an open source tool.

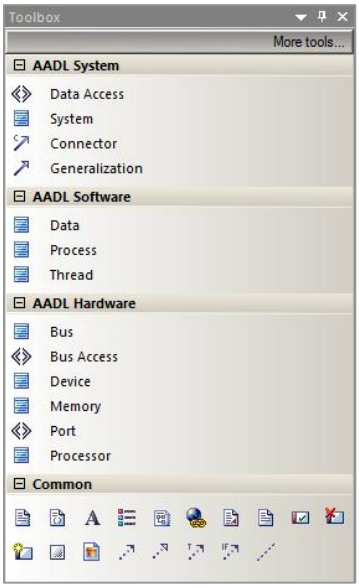

Figure 9. AADL profile tool palette in Enterprise Architect

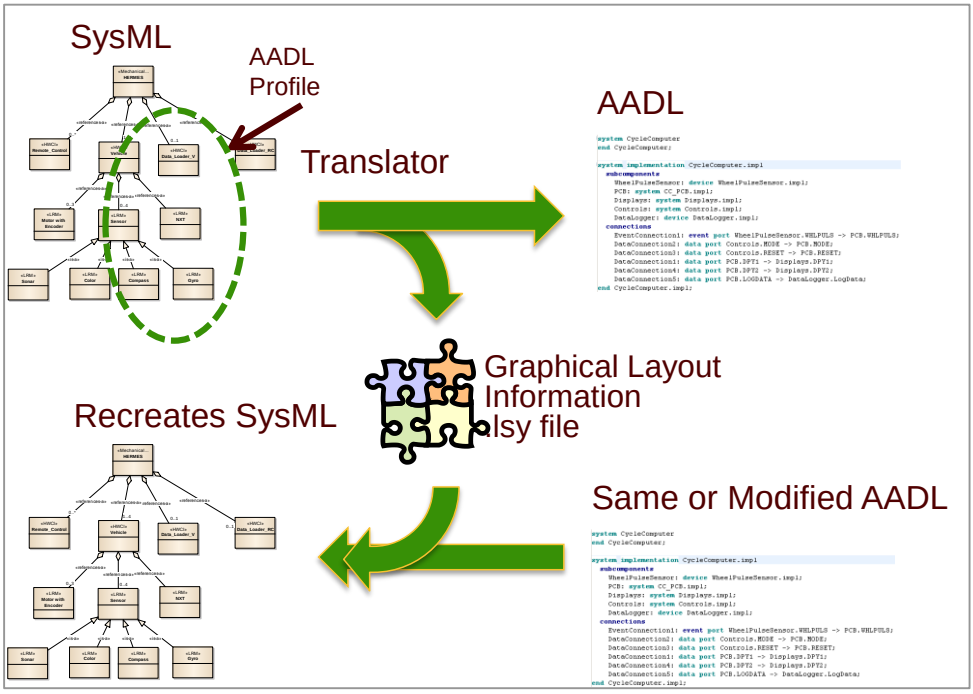

Figure 10. SysML-AADL META Translator workflow

The preferred workflow (Figure 10) starts with a SysML architecture model using the AADL profile in Enterprise Architect 10.0 and translates it into AADL. In the process, a graphical layout file is created on the side, but it is not used in the AADL model. This graphical layout file contains the position of the graphical objects in the SysML diagrams. While the preferred practice is to maintain the architecture in SysML and translate to AADL, situations may arise where changes made in AADL need to be brought back to SysML. Then, the updated AADL model is combined with the previously created graphical layout information file to create a new SysML file containing the new and modified objects. For those "new" graphical objects that did not exist in the original

file, the SysML tool's default location is used in the new diagrams. If there is no preexisting graphical layout information file, then all the graphical objects in the newly created SysML file will be placed according to the tool's default location (usually all objects on top of each other).

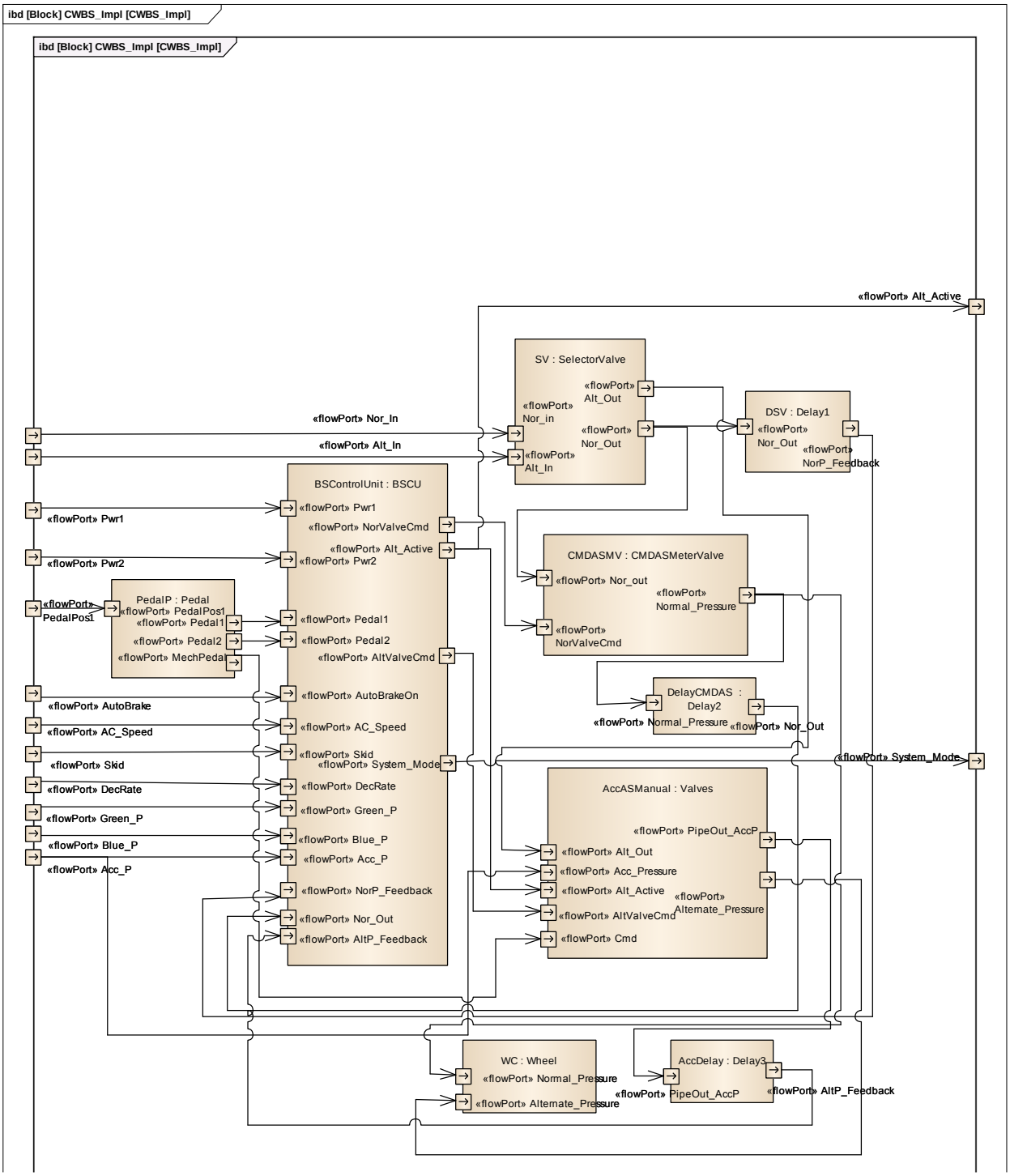

Figure 11. WBS SysML representation using AADL profile

The profile's AADL components are shown in Figure 12 and the profile's AADL features are shown in Figure 13. Not shown in this summary report are details of the translator and examples of its working menus; these details are preserved in [17] for SAVI members.

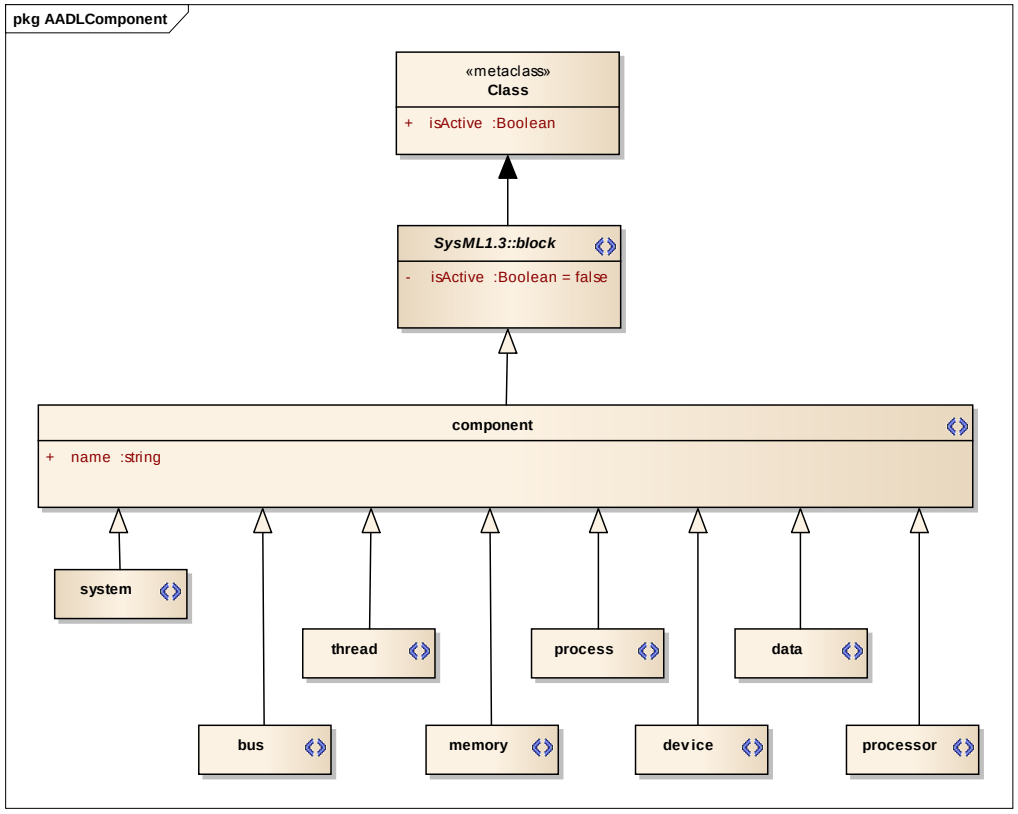

Figure 12. AADL components in SysML profile

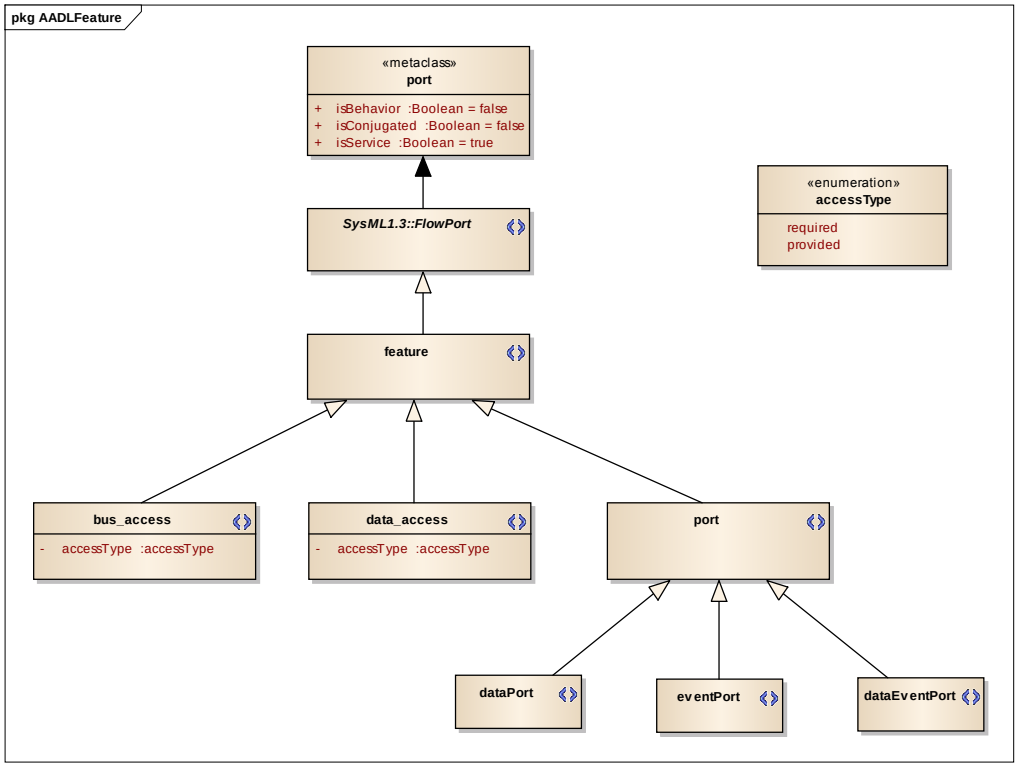

Figure 13. AADL features in SysML profile

3.4 Analyzable AADL Model Set

The focus of AFE 61 was on system safety but included intellectual property (IP) protection for the OEM and suppliers during system development. To tackle these issues, a set of AADL models, both for OEM and suppliers, was created. Due to the focus of this AFE, the models deal with safety features of the system, but an attempt was made to create them at an abstraction level so that extensions for other aspects (such as behavior) can be added easily. The techniques to create such multi-disciplinary models are one of the competencies that SAVI must progressively develop. Finally, AADL was chosen as a primary architectural definition language to provide a strong semantic structure that facilitates analysis of the entire system.

3.4.1 Wheel Braking System (WBS) Functional Description

The system modeled was a wheel-braking system (WBS) largely based on the one described in AIR 6110 [1]. As discussions about the task progressed, the SAVI team changed some specifications of the system for three main reasons: (i) to create a simpler, but representative, system, thus allowing the group to focus on exercising the process, instead of a more detailed system; (ii) to leverage existing knowledge of the partners; and (iii) this initial model set was available and free of intellectual property restrictions. One such simplification was to model a simpler two-wheeled main landing gear rather than the braking system for a bogie type main landing gear configuration. This simpler system is shown in block diagram form in Figure 14. Each strut and axle assembly is supported by two wheels, each wheel having two actuators. Each of these actuators is fed by both hydraulic systems (called Green and Blue). The hydraulic system powering the WBS is selected through an autonomous selector valve. This valve simply allows the hydraulic system with the highest pressure to command brake actuation.

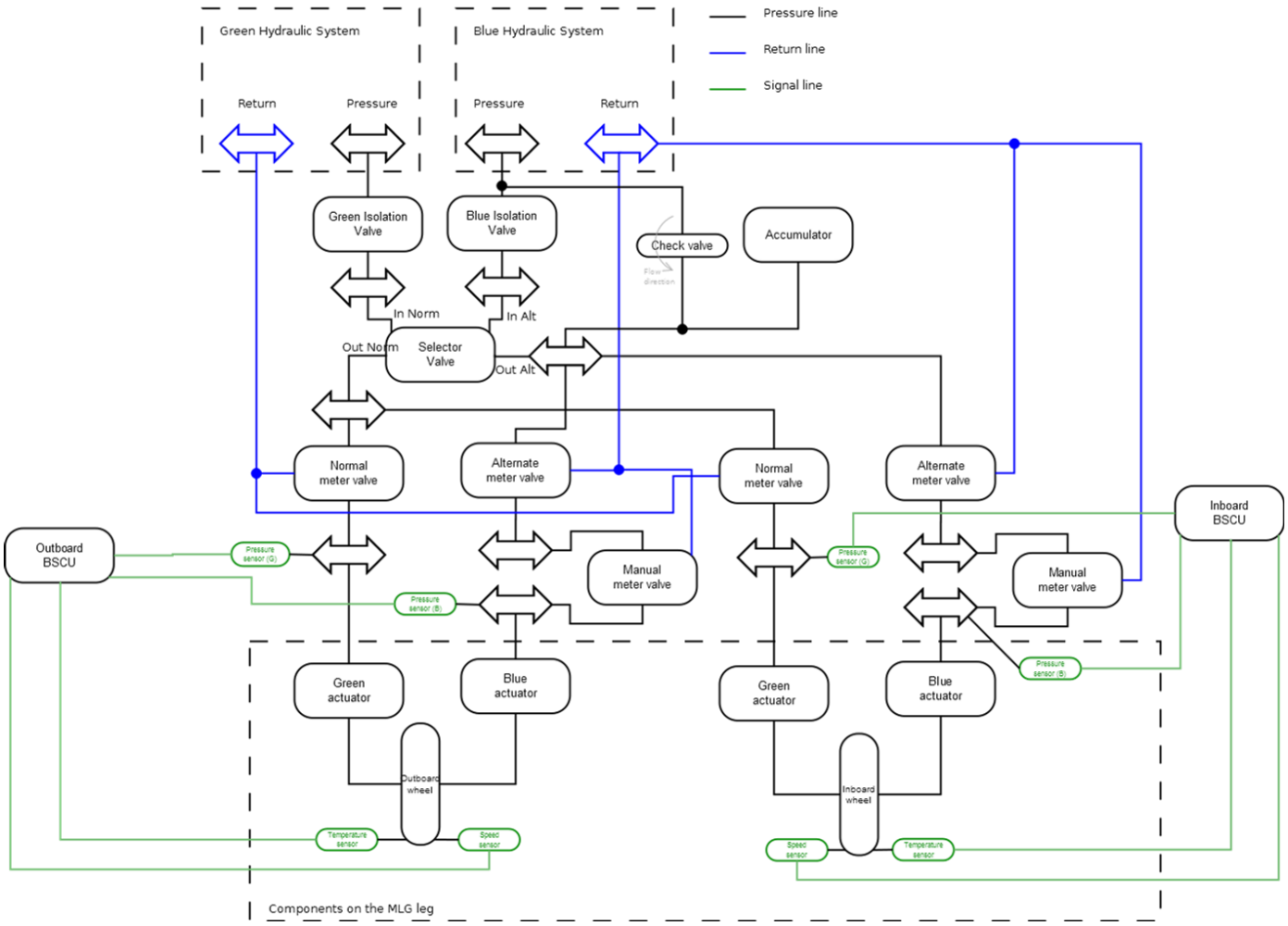

Figure 14. Functional block diagram of one leg of the WBS

3.4.2 System Redundancies Described

Upstream from the selector valve there are isolation valves. These valves avoid damage in case any part of the WBS develops a leak that would drain the hydraulic system. The isolation valves serve as hydraulic fuses that close if the flow is above a given threshold. Downstream from the selector valve there are the electrical metering valves. These valves are commanded by the Brake System Control Units (BSCUs), though this connection is not shown in Figure 14. There are two BSCUs, one that controls the inboard wheels and the other controlling the outboard wheels. This architecture is needed in case of failure of a single BSCU which could lead to asymmetric braking, a condition that can be more hazardous than a symmetrical failure of the brakes (per the FHA in AIR 6110 [1]).

After the automatic metering valves, on the alternate hydraulic line, there is a manual metering valve. This manual metering valve can be actuated by the crew in case of failure of both BSCUs. This manual metering valve can also be fed by an accumulator, providing more redundancy in case both hydraulic systems fail.

Finally, the system uses several sensors (some for the future anti-skid control loop – the wheel speed sensors, for example) as well as in monitoring systems, such as the wheel temperature sensors. For example, the crew should not start a take-off if the brake temperatures are beyond a given threshold, otherwise the brake temperatures can become so high in case of rejected take-off that the brakes would lose efficiency.

3.4.3 AADL Error-Model Annex

The AADL Error-Model Annex [18, 19] augments the AADL through its annex extension mechanism, providing added capability to annotate components with safety-related information so analysts can evaluate how individual errors propagate through the system and ultimately how component changes affect the overall safety metrics for the system. This Annex is a sub-language supported with the Open Source AADL Tool Environment (OSATE) with the following important features:

- The Error-Model Annex specifies several error types designed to distinguish between different kinds of errors. It also contains a comprehensive type library of common error types like TimingError, ValueError, and the like. Each of these types can be further extended to more completely describe an error; TimingError could be extended with EarlyDelivery or LateDelivery to specify the kind of timing error to be considered.
- An error event is an internal event for a system element that is specific to error modeling. An error event is not related to component interfaces but it is part of the internal component specification.
- Error sources, error sinks, and error paths spell out how an error propagates within the system architecture.
- An error behavior state machine is a state machine containing system states, events, and transitions. A transition represents a condition for switching from one state to another. The triggering event may be either an internal error event or an incoming external event propagation. Figure 15 illustrates such an error behavior state machine with four states and three transitions.

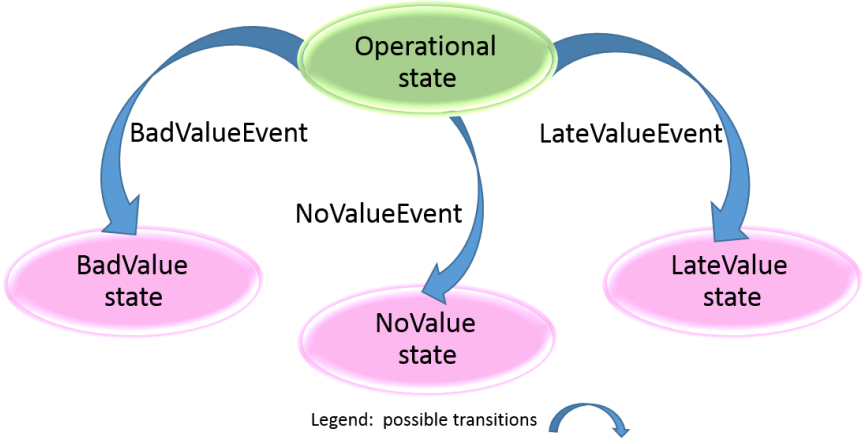

Figure 15. Error behavior state machine [18]

- Composite error behavior state machine explicitly records the relationship between the error states of a system and the error states of the system elements; that is, it defines the actual error state of a system element according to the error state of its sub-element. Figure 16 depicts a composite error model in which the system is failing because subsystem 2 is failing (right side).

Figure 16. Composite error-model [18]

3.5 Solid Geometry Models

The AP214 Solid Geometry Model defines the location of the major system components and the routing of the interconnections between them in three-dimensional space. This model was produced using the Solid Works tool. Since not all member organizations in AFE 61 have this tool, it was exported from Solid Works in STEP AP203 format, read up into NX and exported in STEP AP-214 format. These artifacts are located on the AFE 61 tab on the SAVI SharePoint web site at: https://members.avsi.aero/SAVI/AFE61/PD2/Forms/AllItems.aspx?RootFolder=%2FSAVI%2FAFE61%2FPD2%2FImp%2FGeometry%20Models%2FTAMU%20LG%2DWBS%20Geometry

3.5.1 Layout

Figure 17 shows the layout of the components and interconnections in the fuselage of the aircraft. Figure 18 shows the layout of the components and interconnections in the left wheel well. AFE 61 is the first SAVI project that incorporated a solid geometry model into the model set. The main purpose for adding this solid geometry model is to facilitate inter-model consistency checks that deal with real-world implementation issues rather than only those that can be represented in an abstract functional or logical model.

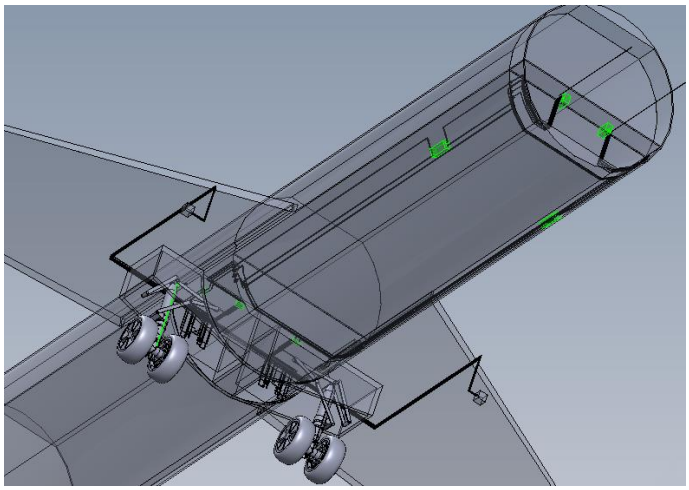

Figure 17. Components and interconnections in the airplane

3.5.2 Limitations of AFE 61 solid models

While these solid models served the purposes of AFE 61 well, they lack details that will be needed for future development of the VIP. The most important of these constraints are:

- Some important subsystems for the WBS are not included in this simplified model. There is neither an antiskid subsystem nor an autobraking subsystem incorporated in this model. These subsystems are essential to evaluate some of the important performance metrics for the WBS.
- The interfacing supply systems (electrical, hydraulic, and pneumatic) are not detailed and the complete set of interfaces with these supply systems is not included in the AFE 61 simplified model. The dynamics of these supply systems must be fully described in models before an adequate evaluation of the performance of the WBS can be considered complete.

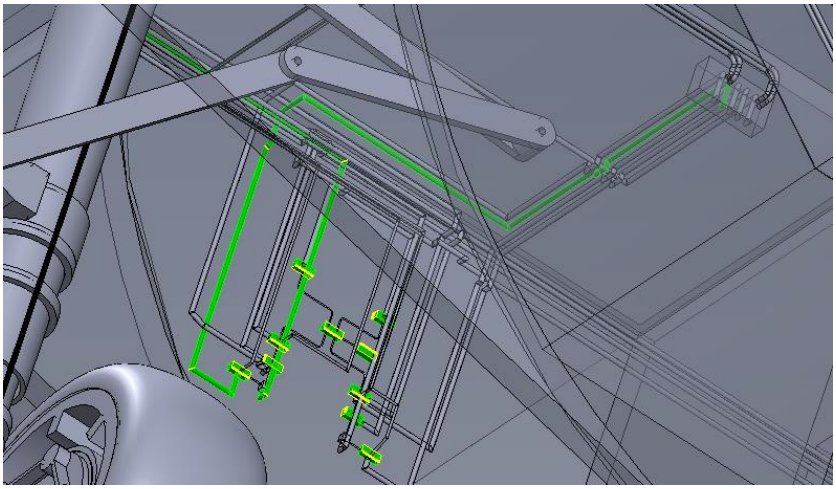

Figure 18. Components and interconnections in the wheel well

4. Demonstrations

4.1 Model-Based System Safety Process

AFE61 concentrated on developing a model-based system safety process underpinning a commercial aircraft development. Specifically, examples were taken from the AIR 6110 wheel braking system (WBS) [1]. The process so far includes generation of the fault tree analysis (FTA) based upon an assumed set of hazards presented as inputs to the process using a spreadsheet-based Functional Hazard Analysis (FHA). A Failure Modes and Effects Analysis (FMEA) has been demonstrated but a full-blown failure modes, effects, and criticality analysis (FMECA) has not been demonstrated. Other safety analyses like the common cause analysis (CCA), the zonal risk assessment (ZRA), and evaluation of a system's development or design assurance level (DAL) have not yet been demonstrated from the SAVI model set.

Aircraft development starts with requirements from marketing, technology research and certification. These requirements are inputs to the AADL aircraft functional model development, starting with a list of high level aircraft functions. As development progresses these functions are decomposed into low level aircraft functions and the model set refined to incorporate more detailed models.

For each function, failure is evaluated in accordance with FAR 25.1309. Failure probabilities shall be verified against values shown in Table 4. A safety probability budget should be allocated for each system at PSSA level and meeting that allocated probability level will be a requirement for the responsible developer (OEM or supplier).

For each supplier of a SAVI- compliant subsystem, the OEM provides information needed through the SAVI MR/DEL, where the supplier and OEM can access information in model form, with AADL the preferred language for system safety evaluations. Once data are available, the supplier develops a subsystem model that meets OEM requirements and provides an FTA to the OEM. Integration of the subsystem model, using provided probability estimates, is then carried out with the other subsystems. Details of the safety process [19] spelling out the model-based process were developed during AFE 61.

Table 4. Hazard classification [21]

Hazard Classification	Maximum Probability/ Flight hour
Catastrophic	10^{-9}
Hazardous	10^{-7}
Major	10^{-5}
Minor	10^{-3}
No Safety Effect	--

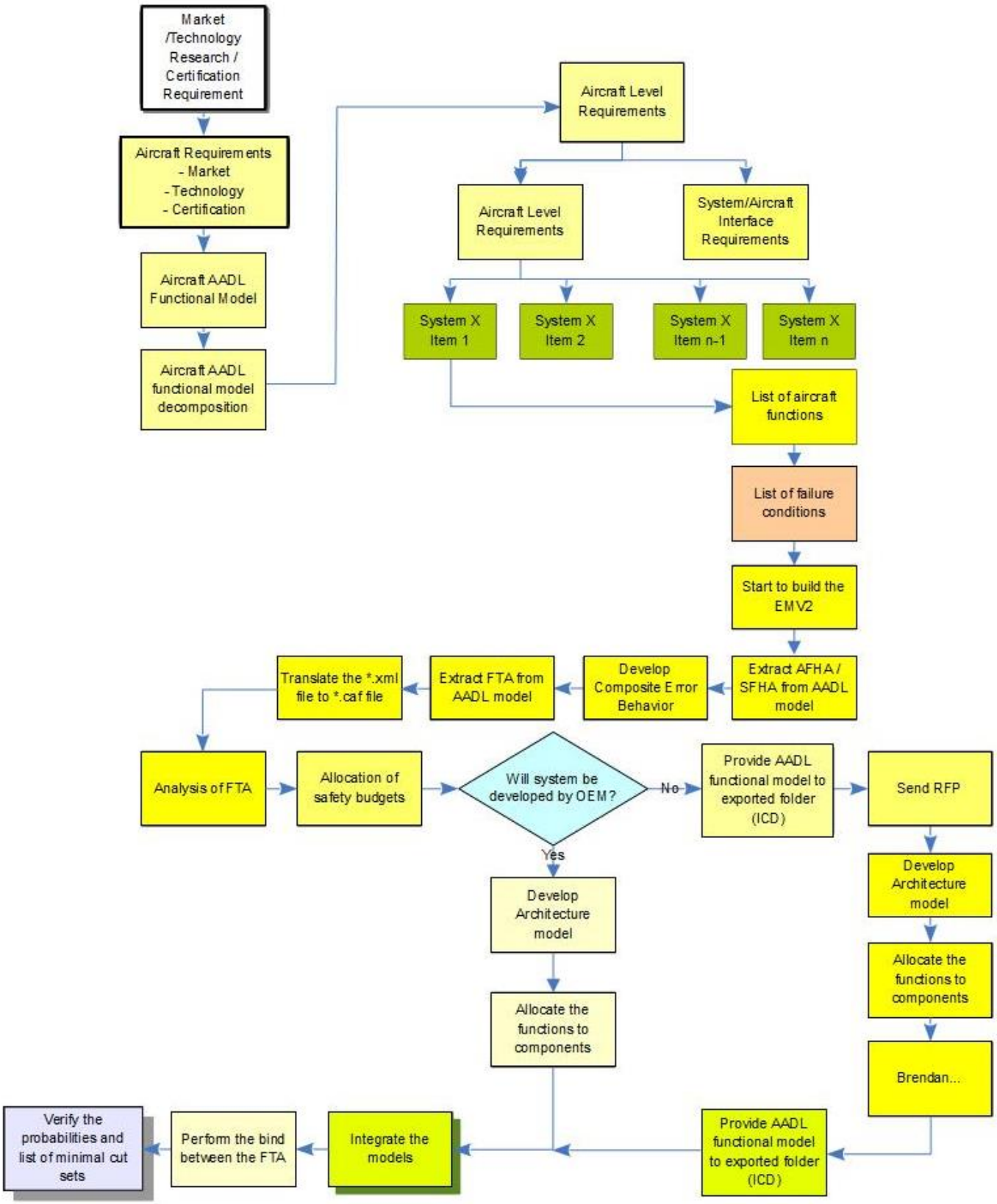

Figure 19. Safety modeling process for AFE 61

4.2 Intellectual Property Protection in AFE 61

The IP protection approach used in AFE 61 is described in a 2012 SAVI report [22]. The basic idea is that each party has separate repositories but is able to interlink the repositories, with each company selecting (manually for now) relevant parts and explicitly exporting them to suppliers as needed. The interlinking mechanism between repositories simplifies updating.

This approach was extended with the use of "Publisher-Subscriber" (Pub-Sub) models with the sole purpose of defining a shared interface, thus facilitating a hierarchical decomposition of models to create assemblies. A Pub-Sub model consists of generic components with their interfaces to inter-connect them with all other components. It formally defines communication mechanisms between components including data types exchanged. Pub-Sub components are exported to a supplier so they can develop an implementation component that complies with the expected interfaces. This compliance eases the integration of all components into a single architecture that has interfaces matched between components that must communicate.

The idea behind assemblies is to model simpler components, with a partially defined interface and then define more complex components in terms of these interfaces. Using this arrangement, variations of a more complex component can be created by changing the references to the basic components. These changes can be easily done in OSATE, the AADL editor used in this exercise.

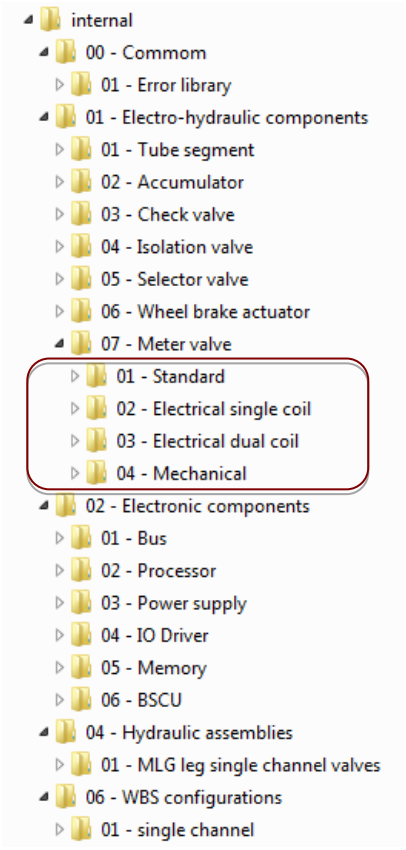

Figure 20. Example of supplier's folder structure

The example above helps in understanding what is meant by a "partially defined" interface. In the single coil/dual coil case, the number of ports in the interface stays the same. The only difference is the definition of one of the feature groups. This arrangement makes it relatively simple to create variations and thus to test several design options in a short time.

4.3 OEM Model

The objective in AFE 61 was to simulate an aircraft system development where models are used as the primary means of communication between OEM and suppliers. Therefore, OEM models focused on defining the basic structure of the aircraft; that is, the high-level systems and how they interface with each other and the functional architecture of the aircraft. The goal was to extract views to simulate aircraft system development utilizing this model set. Then, using the interlinking mechanisms of the repositories, the models could be exchanged with multiple suppliers. Finally, supplier models would be built and tested against the OEM model.

Due to this AFE's focus, the OEM models concentrated on safety-related features. The AADL model developed during the AFE included both the functional model and the error model, with a level of detail compatible with the PSSA phase, according to ARP 4761. These models allowed the automatic creation of part of the data for the FHA and FTA. DAL assignment and CCA were not done during this AFE.

4.3.1. AADL Functional Model

The AADL functional model was created assuming the aircraft development was just beginning. A list of high level aircraft functions (Figure 21) was generated and these functions were sequentially refined as low level aircraft functions to detail the systems. Part of the high level aircraft functions are shown below in a clip of the AADL code used to implement this functional model in OSATE and its graphic view. This partial set of code includes three sequential parts:

1. Provide control on the ground
   - Control speed
   - Control direction
2. Provide operational awareness
   - Awareness of emergency
3. Provide power generation and distribution
   - Provide hydraulic power
   - Provide electrical power generation

```
system AircraftFunctions extends Function
end AircraftFunctions;

system implementation AircraftFunctions.i --> High Level
  subcomponents
    provideControlOnGround             : system ProvideControlOnGround.i;
    provideOperationalAwareness        : system ProvideOperationalAwareness.i;
    providePowerGenerationAndDistribution : system ProvidePowerGenerationAndDistribution.i;

  connections
    controlOnGroundStatus : feature group provideControlOnGround.awarenessMessages ->
                            provideOperationalAwareness.controlOnGroundMessages;
    powerGenerationAndDistributionStatus : feature group
providePowerGenerationAndDistribution.awarenessMessages ->

provideOperationalAwareness.powerGenerationAndDistributionMessages;

    controlOnGroundPower : feature group providePowerGenerationAndDistribution.powerOutput <->
                           provideControlOnGround.powerInput;

end AircraftFunctions.i;
```

Figure 21. High-level functions implemented in OSATE

From the high level functions (Figure 21), the system was refined by creating lower level aircraft functions, during the second iteration of the functional model. Several refinements were evaluated, elaborating the low level aircraft functions to detail the system, resulting the low level aircraft functions (Figure 22).

1. Provide hydraulic power
   - Provide hydraulic power system green
   - Provide hydraulic power system blue
2. Provide electrical power
   - Bus 1
   - Bus 2
3. Control Speed
   - Decelerate aircraft on ground
   - Provide primary stopping force
   - Provide secondary stopping force
   - Decrease lift/create drag
   - Remove forward thrust
   - Transfer stopping force

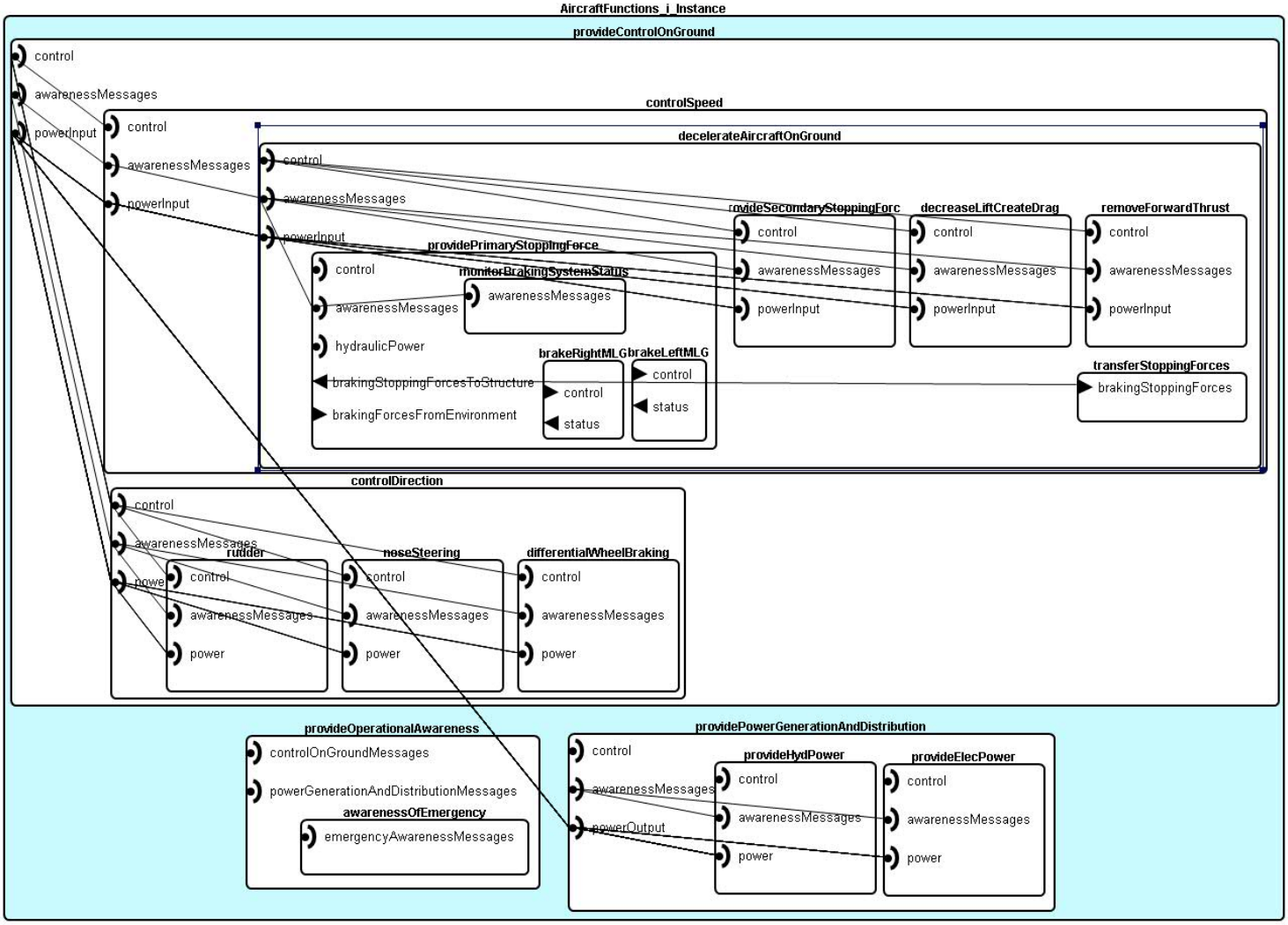

Figure 22. Final aircraft functions refinement / low level aircraft functions

4.3.2. Failure Hazard Assessment

Based on the aircraft functions identified in Figure 19, the next step suggested by ARP 4761 is classification of severity of failure of each aircraft function in accordance with FAR 25.1309. Using the AADL models for each of these functions a list of failure conditions is directly extracted into a Failure Hazard Assessment (FHA) spreadsheet. This spreadsheet contains the failure condition, flight phase, hazard classification for each flight

phase, description of the failure conditions and comments. A part of the code (Figure 23) used to represent FHA information on the models is shown below and the table generated from the same model is shown in Table 4.

```
EMV2::hazards =>
                ([crossreference => "AIR6110 page 36 figure 17";
                  failure => "announciatedPartialSymmetricalLossOfWheelBraking";
                  phases => ("Landing");
                  description => "Annunciated partial symmetrical loss of
                                  wheel braking";
                  comment => "The crew is aware that there is a partial loss of
                              braking before landing. Crew uses wheel braking,
                              spoilers and thrust reversers available to
                              maximum extent to decelerate the aircraft. The
                              temperature on wheels of the loaded brakes
                              increases and could reach point where whee/fire
                              failure occurs. Depending on number of brakes
                              lost result could be an overrun.";
                  severity => ARP4761::Hazardous;
                  likelihood => ARP4761::ExtremelyRemote;
                 ])
                applies to failed.AnnounciatedPartialSymmetricalLossOfWheelBraking;

            EMV2::hazards =>
                ([crossreference => "AIR6110 page 36 figure 17";
                  failure => "asymmetricalLossOfWheelBrakingOnly";
                  phases => ("Landing", "RTO");
                  description => "Asymmetrical loss of wheel braking - brake
                                  system failure only";
                  comment => "Decrease in braking performance. Tendency to veer
                              off the runway. For braking performance and brake
                              temperature the effects are the same as partial
                              brake loss. The crew keeps the aircraft on the
                              runway by using rudder at high speed and nose
                              wheel steering at low speed.";
                  severity => ARP4761::Hazardous;
                  likelihood => ARP4761::ExtremelyRemote;
                 ])
                applies to failed.AsymmetricalLossOfWheelBrakingOnly;
```

Figure 23. FHA code implemented in OSATE

4.3.3 Fault Tree (FT) Analysis

Fault Trees were extracted directly from the models, but it was necessary to first build the composite error behavior in AADL. At this level, the objective of the FT analysis done by the OEM is to provide maximum failure probability numbers for the subsystems; that is, the safety budgets for the supplier to develop their subsystems for the PSSA. After the subsystems are integrated into the overall system a System Safety Analysis (SSA) is completed with the supplier's FTs bound with the OEM's FTs. The failure conditions for these FTs must comply with FAR 25.1309.

Table 5. FHA extracted from AADL model

Component	Error	Hazard Description	Crossreference	Functional Failure	Operational Phases	Environment	Severity	Likelihood	Verification	Comment
Root system	"UnannounciatedTotalLoss OfWheelBraking on	"Total Loss of wheel braking"	"AIR6110 page 35 figure 17"	"unannounciatedTotalLossOfWheelBraking"	"Landing or RTO"		Hazardous	ExtremelyRemote		"Crew detects the failure when the brakes are operated. The crew uses spoilers and thrustreversers to the maximum
Root system	"AnnounciatedTotalLoss OfWheelBraking on	"Annunciated loss of wheel braking"	"AIR6110 page 35 figure 17"	"announciatedTotalLossOfWheelBraking"	"Landing or RTO"		Hazardous	ExtremelyRemote		"Crew selects a more suitable airport
Root system	"UnannounciatedPartialSymmetricalLossOfWheelBraking on failed"	"Annunciated partial symmetrical loss of wheel braking"	"AIR6110 page 35 figure 17"	"unannounciatedPartialSymmetricalLossOfWheelBraking"	"Landing or RTO"		Hazardous	ExtremelyRemote		"The crew detects the failure whe the brakes are used. Crew uses available wheel braking
Root system	"AnnounciatedPartialSymmetrical LossOfWheelBraking on failed"	"Annunciated partial symmetrical loss of wheel braking"	"AIR6110 page 36 figure 17"	"announciatedPartialSymmetricalLossOfWheelBraking"	"Landing"		Hazardous	ExtremelyRemote		"The crew is aware that there is a partial loss of braking before landing. Crew uses wheel braking
Root system	"AsymmetricalLossOfWheelBrakingOnly on failed"	"Asymmetrical loss of wheel braking - brake system failure only"	"AIR6110 page 36 figure 17"	"asymmetricalLossOfWheelBrakingOnly"	"Landing or RTO"		Hazardous	ExtremelyRemote		"Decrease in braking performance. Tendency to veer off the runway. For braking performance and brake temperature the effects are the same as partial brake loss. The crew keeps the aircraft on the runway by using rudder at high speed and nose wheel steering at low speed."
Root system	"InadvertentWheelBrakeApplicationWithoutLocking on failed"	"Inadvertent wheel brake application without wheel locking"	"AIR6110 page 37 figure 17"	"inadvertentWheelBrakeApplicationWithoutLocking"	"Takeoff before V1"		Hazardous	ExtremelyRemote		"The crew stops the aircraft on the runway"
Root system	"InadvertentWheelBrakeApplicationWithAllLocked on failed"	"Inadvertent wheel brake application with all wheels locking"	"AIR6110 page 37 figure 17"	"inadvertentWheelBrakeApplicationWithAllLocked"	"Takeoff before V1"		Hazardous	ExtremelyRemote		"Potential burst of all tires and loss of braking efficiency"
Root system	"InadvertentWheelBrakeApplicationWithAllLocked on failed"	"Inadvertent wheel brake application with all wheels locking"	"AIR6110 page 37 figure 17"	"inadvertentWheelBrakeApplicationWithAllLocked"	"Takeoff after V1"		Catastrophic	ExtremelyImprobable		"Crew cannot takeoff or safely RTO resulting in high speed overrun"
Root system	"UndetectedInadvertentWheel BrakeApplicationWithoutLocking on failed"	"Undetected inadvertent wheel brake on one wheel without locking of the wheel"	"AIR6110 page 37 figure 17"	"undetectedInadvertentWheel BrakeApplicationWithoutLocking"	"Takeoff"		Hazardous	ExtremelyRemote		"Crew cannot detect the failure by the asymmetry which is very small. Brake temperature can reach very high temperature. Crew retract gear resulting in possible wheel fire or tire failure."
Root system	"InadvertentWheelBrakeApplication WithoutLockingPlusHighTemperature on failed"	"Inadvertent application on one wheel without locking of the wheel without locking of the wheel coupled with detected high brake temperature"	"AIR6110 page 37 figure 17"	"inadvertentWheelBrakeApplicationWithoutLockingPlusHighTemperature"	"Takeoff"		Hazardous	ExtremelyRemote		"Crew cannot detect failure by asymmetry which is very small. Brake temperature can reach very high temperature. Crew detects high brake temperature and leaves gear extended to cool brake."

Each function is classified as one of two types of failures in the state machine if an operation fails. Therefore, errors propagated are classified in the same two types: those of omission and those of commission. Omission failures indicate that the function did not successfully complete and commission failures mean the function runs when it should not (Figure 24).

When FTs are generated by OSATE (Figure 25), the data are placed in two files. The file with the extension *.fta is used by the OpenFTA tool while the file with the *.xml extension is used by a translator to generate a *.caf file for the CAFTA tool. OpenFTA does not calculate probability distributions nor does it generate a list of minimum cut sets (a minimal subset of the fault tree events ALL of which must occur to cause the TOP event to happen). Therefore, the FTs were translated from *.xml file to *.caf format so CAFTA can more completely analyze the FT generated from the AADL models.

```
annex EMV2 {**
error types
    FunctionalError: type;

    -- This ensures that Omission and Commission are disjoint
    Omission: type extends FunctionalError; -- The function didn't run when it
should
    Commission: type extends FunctionalError; -- The function run when it shouldn't

end types;

error behavior FunctionalErrorBehavior
    use types ProvidePrimaryStoppingForceErrorLib;

    states
        operational : initial state;
        failed : state{UnannounciatedTotalLossOfWheelBraking,
                       AnnounciatedPartialSymmetricalLossOfWheelBraking,
                       UnannounciatedPartialSymmetricalLossOfWheelBraking,
                       AsymmetricalLossOfWheelBrakingOnly,
                       InadvertentWheelBrakeApplicationWithoutLocking,
                       InadvertentWheelBrakeApplicationWithAllLocked,
                       UndetectedInadvertentWheelBrakeApplicationWithoutLocking,
                  InadvertentWheelBrakeApplicationWithoutLockingPlusHighTemperature};

    properties
        EMV2::StateKind => Working applies to operational;
        EMV2::StateKind => NonWorking applies to failed;
end behavior;
**};
```

Figure 24. Error model implemented in OSATE

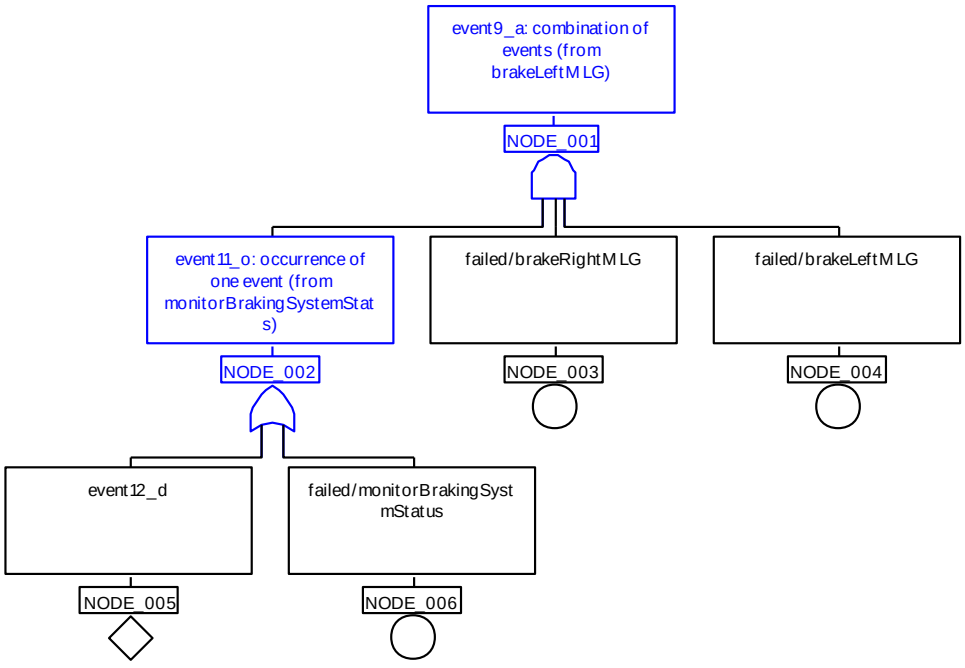

Figure 25. FTA for failure condition unannunciated total loss of wheel braking extracted from AADL model in OSATE

5. Conclusions and Recommendations

5.1 Conclusions

This report illustrates how the SAVI Virtual Integration Process was carried out during its first year of developing an operational set of capabilities..

5.2 Recommendations

5.2.1. NAR-Based

- *Engage more fully with related efforts*. The Non-Advocate Review (NAR) conducted in November 2013 pointed strongly to the necessity of closer cooperation with other projects. While SAVI has attempted to stay in touch with a number of these efforts, there are others that need to be included as well.
- *More please!* The Non-Advocate Review (NAR) conducted in November 2013 pointed strongly to the necessity of closer ties between all groups working on model-based engineering activities. The SAVI team needs to redouble its efforts to collaborate more fully with other MBSE-based development efforts.

5.2.2. Team-Generated

- *Ensure that current work does not conflict with overall MBSE developments*. The success of this long term goal depends on the depth and breadth of MBSE knowledge that the SAVI team maintain; in that context, this recommendation is an extension of the second near term recommendation. But this knowledge is broader and more comprehensive than is needed in the near term. Keeping up with this goal will be difficult to accomplish, given the urgency of the resource limitations SAVI is likely to be under. On the other hand, the long term ramifications of ignoring such depth and breadth of outlook are potentially even more damaging than not completing some of the short term activities. The difficulty part will be balancing the short term urgencies against the long term benefits.
- *Concentrate on attracting more full (paying) participants in SAVI*. As pointed out in [15], the most uncertain assumption in this plan are those made concerning available resources, especially skill sets to carry out the necessary process development. Modelers able to knowledgeably address specific untapped domains are in very short supply and their talents are fully committed for the most part. One way to compensate for this shortage of modelers is to contract with other organizations (industry, government, and academia) that have these skill sets. To meet this essential SAVI need, the team needs the flexibility of cash for contracted efforts. Attracting new members with the needed skill sets is the most pressing immediate need to ensure the health of the SAVI program.