



**Carnegie Mellon
Software Engineering Institute**

Pittsburgh, PA 15213-3890

Architecture Analysis & Design Language (SAE AADL)



Peter H. Feiler
Performance Critical Systems-AADL Initiative

November 2005

Sponsored by the U.S. Department of Defense
© 2005 by Carnegie Mellon University

1



Embedded Systems Problems

Incomplete capture of specification and design

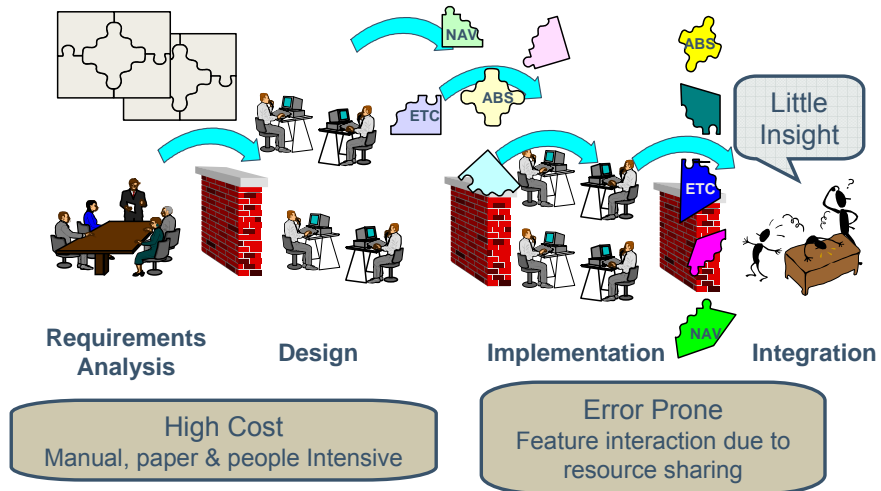
Little insight into non-functional system properties until system integration and test

- Performance (e.g., Throughput, Quality of Service)
- Safety
- Time Criticality
- Schedulability
- Reliability
- Security
- Fault Tolerance

Result:

- system integration – high risk
- evolvability – very expensive
- life cycle support – very expensive
- leads to rapidly outdated components

Typical Software Development Process



© 2005 by Carnegie Mellon University

3

This defines how things are done today.

- Sequential (very little iteration);
- Distinct phases with idiosyncratic methods, techniques, and notations;
- System architecture and designs documented on paper with little formalism;
- Manual code development;
- Large scale at system integration time with little insight on potential problems.



Model-Based Engineering (MBE)

Ensure embedded, real-time system performance and reliability *prior* to system integration, test, or upgrade

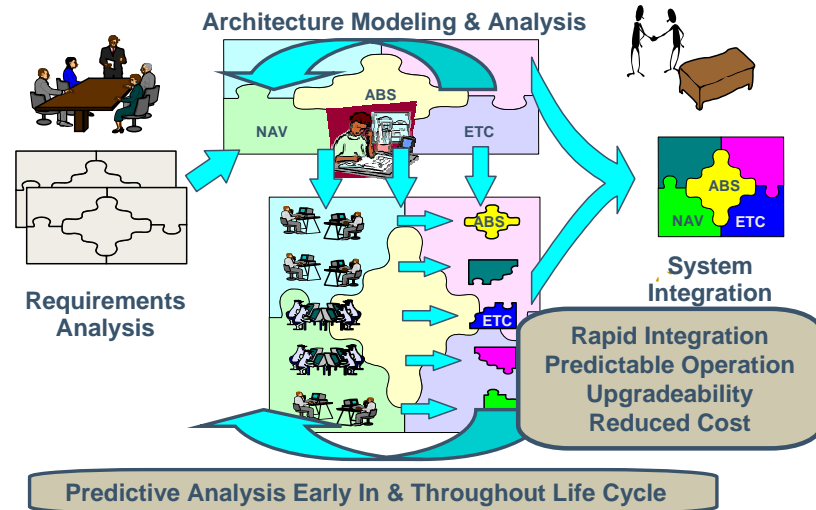
Prediction through quantitative analysis & simulation of system operation based on architecture models

System validation through model verification and implementation compliance checking





Model-Based System Engineering



Model-based system engineering focuses on modeling and analysis of system architectures.

The development and evolution of a system is driven by the use of architecture models throughout the development life cycle.

Predictive analysis of such architecture models early and throughout the life cycle leads to early detection of potential problem, in particular with respect to non-functional system properties such as performance, reliability, security, and safety-critical issues.

Application components are developed in the context of this architecture specification. The development process

This results in reduced integration time and cost, more predictable system operation, and improved ability to upgrade systems.

The Model-Driven Architecture (MDA) concept by OMG is an instantiation of Model-Based System Engineering.



Carnegie Mellon
Software Engineering Institute

The Role of AADL

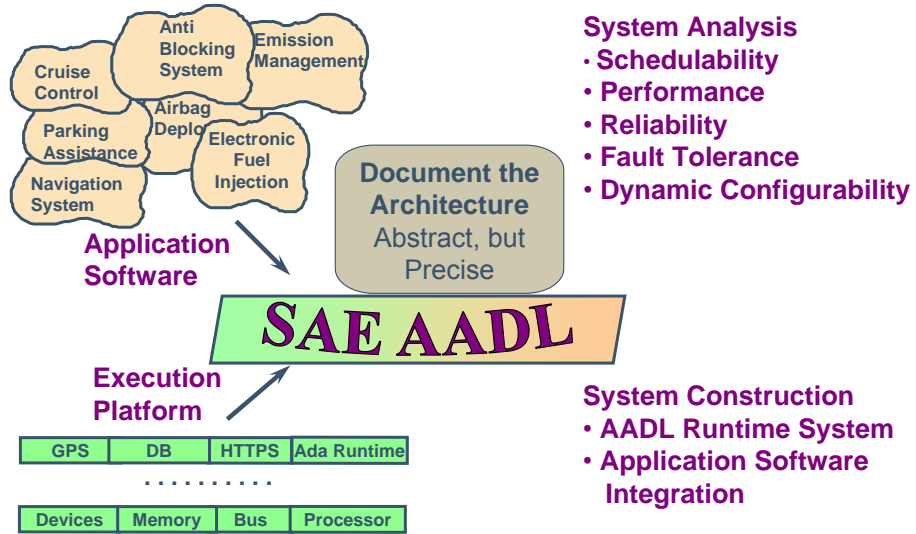
Use the **SAE Architecture Analysis and Design Language** (AADL) standard* and associated tools to specify, analyze, and generate embedded real-time systems

- an SAE standard developed by and for the avionics, aerospace, automotive, and robotics communities
- tool-based analysis in Eclipse framework
- an application of model-based engineering ideas

SAEInternational

* SAE document AS-5506

Model-Based System Engineering





AADL: The Language

Precise execution semantics for components & interactions

- Thread, process, data, subprogram, system, processor, memory, bus, device

Continuous control & event response processing

- Data and event flow, synchronous call/return, shared access
- End-to-End flow specifications

Operational modes & fault tolerant configurations

- Modes & mode transition

Modeling of large-scale systems

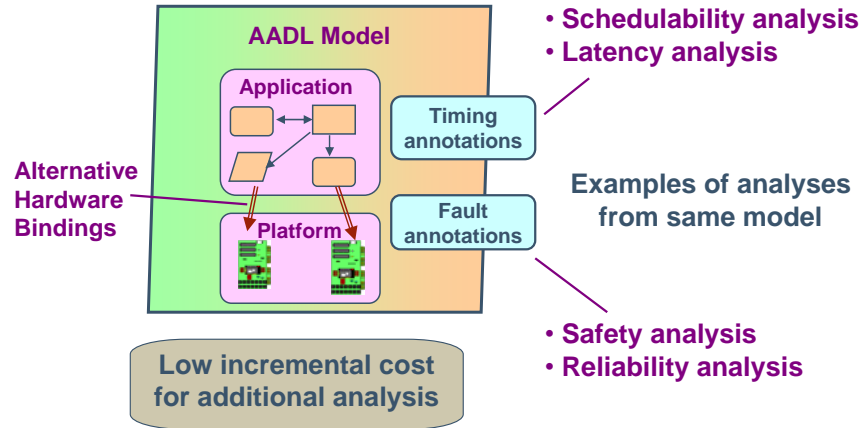
- Component variants, packaging of AADL models

Accommodation of diverse analysis needs

- Extension mechanism, standardized extensions



Common Architecture Model





Key Elements of SAE AADL Standard

Core AADL language standard

- Textual & graphical, precise semantics, extensible

AADL Meta model & XMI/XML standard

- Model interchange & tool interoperability

UML 2.0 profile for AADL

- Transition path for UML practitioner community

Error Model Annex as standardized extension

- Fault/reliability modeling, hazard analysis

The SAE AADL standard has strategically been defined to consist of the following key elements:

Textual and graphical notation with precise semantics for embedded & real-time system modeling permits quantitative analysis of a range of operational system properties ranging from timing to reliability.

As an industry standard architecture modeling notation for applications & execution platforms it provides a common embedded & real-time systems modeling notation across projects and across organizations.

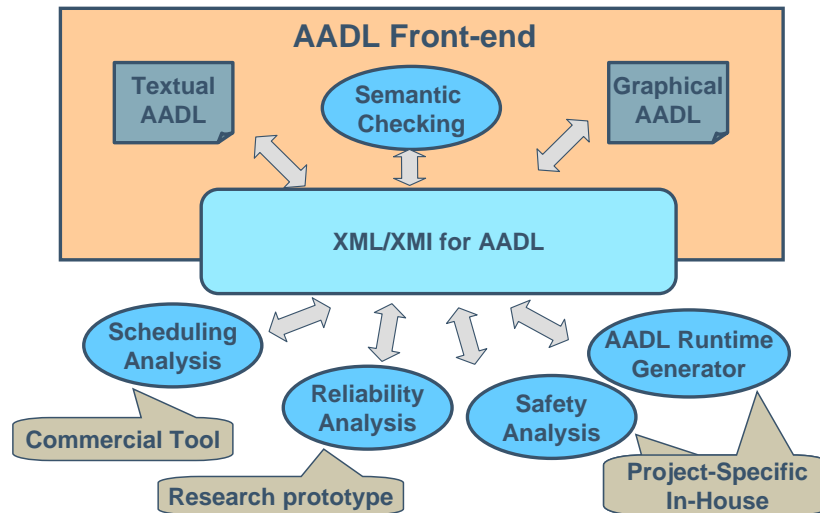
The AADL Meta model & XMI/XML model interchange format standard annex facilitates interchange of AADL models between subcontractors & integrators as well as contractors and government agencies. The model interchange standard also facilitates interoperability of modeling, analysis, and generation tools in support of AADL from different sources. This permits organizations to utilize their existing tool base and evolve and integrated embedded systems engineering environment over time.

The UML profile for AADL standard annex presents the AADL as a specialized modeling notation within the UML framework. This allows the existing UML practitioner community to embrace AADL without giving up their existing modeling environment and process.

The AADL standard has been defined to be extensible to allow for introduction of properties for special modeling & analysis needs as well as the inclusion of analysis-specific sublanguages in a standard-compliant manner.



XML-Based Tool Integration Strategy



© 2005 by Carnegie Mellon University

11

The tool strategy has two aspects.

First, the standard includes an XMI specification and XML schema definition of AADL models.

The XML provides a common interchange format for AADL models between front-end tools and various backend analysis and generation tools. This provides a tool integration and tool interoperability strategy for companies to utilize existing tools and combine them with commercially provided tools and research prototypes into a model-based embedded systems engineering tool environment.

Second, the Software Engineering Institute is making available an open source AADL Tool Environment (OSATE) that includes an AADL Front-end and architecture analysis capabilities as plug-ins to the open source Eclipse environment. Airbus is providing an open source graphical editor that is integrated with OSATE under the TOPCASED initiative.



Two-Tier Tool Strategy

Open Source Tool Solution

- Low entry cost solution based on Eclipse & Eclipse Modeling Framework
- SEI Open Source AADL Tool Environment (OSATE) integrated with open source Airbus TOPCASED tools
- Vehicle for in-house prototyping and for architecture research

Commercial Tool Support

- UML tool environment extension based on UML profile (ILogix, Rational, Artisan)
- Addition of AADL to existing commercial environment (TNI Europe)
- Interface with existing analysis tools (Airbus, Rockwell, Honeywell)

By making an open source version of an AADL tool environment available, the SEI is providing a low entrypoint to adopting SAE AADL as an architecture modeling and analysis approach. Basing the tool set on Eclipse leverages the investment of a large industry base into an open source integrated development environment (IDE). The specification of the AADL meta model in the Eclipse Modeling Framework uses the same MOF technology as used for UML2.0.

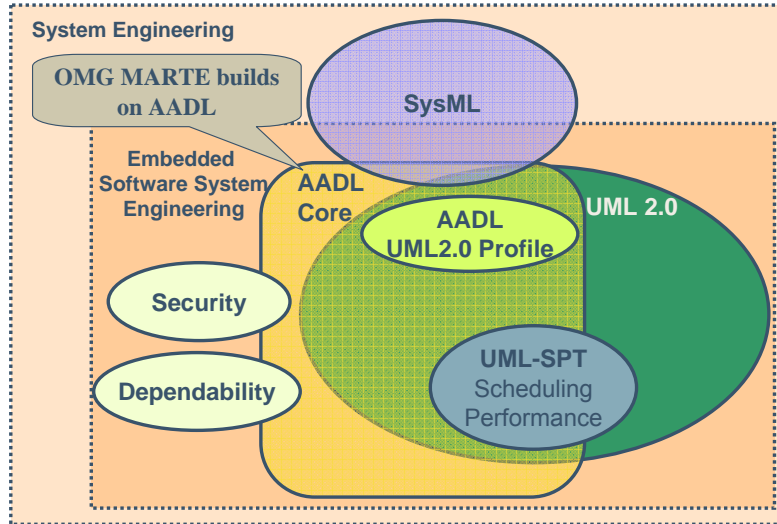
The standardized XML/XMI interchange format supports interoperability of modeling and analysis tools from different sources to operate on the same AADL model. This permits companies to evolve their embedded systems engineering tool environments that incorporates existing tools already in use.

The no cost open source toolset allows universities, research centers, and research departments in industry to prototype new architecture analysis and generation techniques in the context of an industry standard notation, thus, accelerating their research into practice.

The combination of the standardized XML/XMI model interchange format and the standardized UML2.0 profile of the AADL promote the development and adaptation of commercial tools in support of the AADL.



AADL/UML Relationship



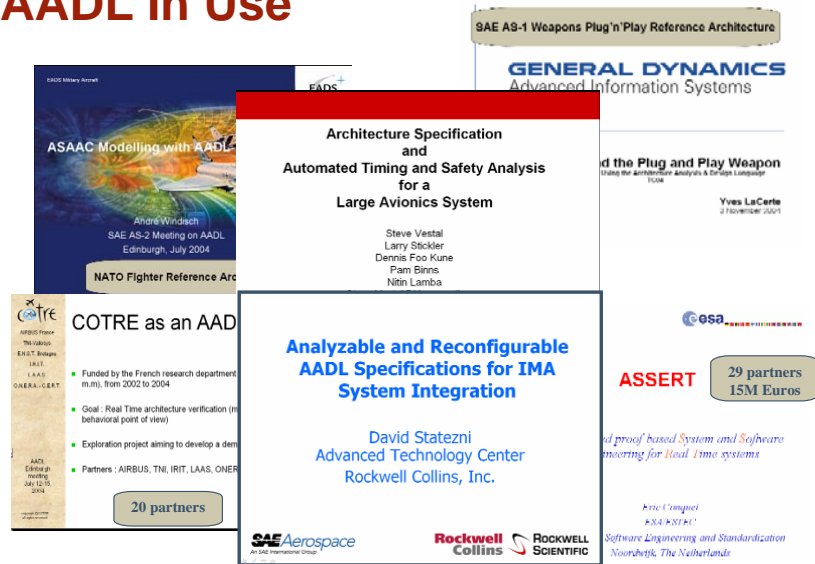
During the development process the AADL has been strategically aligned with UML in order to be of interest to the practitioner community that has embraced UML as their design notation and methodology.

This is achieved through the development of a UML profile of the AADL.

The chart illustrates the overlap in capabilities.

The SAE AADL Subcommittee has recently agreed to collaborate with the OMG MARTE team in the development of a UML profile for Modeling and Analysis of Real-Time Embedded systems (MARTE) by leveraging the AADL standard and its UML2.0 profile.

AADL In Use



SAE AS-1 Weapons Plug'n Play Reference Architecture

GENERAL DYNAMICS
Advanced Information Systems

Architecture Specification and Automated Timing and Safety Analysis for a Large Avionics System

Steve Vestal
Larry Stockler
Dennis Foo-Kune
Pam Binns
Nitin Lamba

ASAAC Modelling with AADL
André Windisch
SAE AS-2 Meeting on AADL
Edinburgh, July 2004

NATO Fighter Reference Architecture

COTRE as an AADL

Funded by the French research department (MRT), from 2002 to 2004

- Goal: Real Time architecture verification (in behavioral point of view)
- Exploration project aiming to develop a demo
- Partners: AIRBUS, TNI, IRIT, LAAS, ONER

20 partners

Analyzable and Reconfigurable AADL Specifications for IMA System Integration

David Statezni
Advanced Technology Center
Rockwell Collins, Inc.

SAE Aerospace
Rockwell Collins
ROCKWELL SCIENTIFIC

ASSERT

29 partners
15M Euros

proof based System and Software engineering for Real Time systems

Eric Compeul
KNA-ROTH**
Software Engineering and Standardization
Noordwijk, The Netherlands

A number of organizations are actively using the AADL in their modeling and analysis efforts.

The pilot applications of AADL range from

Reference architecture modeling and early architecture trade studies to validated system families and performance tuning.

Further details about these projects can be found on the next slide, backup slides, and AADL User presentations on www.aadl.info.



Adopters

Airbus Industries

- Open source tool suite
- Validation of concurrency control in source code

European Space Agency

- Proof-based engineering of satellite system family

Rockwell-Collins

- Performance tuning of avionics system architecture

US Army Common Missile

- Predictive analysis & runtime system generation
- Reduced porting cost

EADS

- Modeling of Eurofighter reference architecture

Honeywell

- Prebid study of architecture alternatives
- Performance studies of avionics architectures

Note that the standard was developed in the Avionics Systems Division of the SAE with active participation of the avionics and aerospace industry with voting members from 22 organizations (see backup slides).

In 2004/05 we have started to expose the automobile industry to AADL. Ford and Toyota have joined the AADL standard subcommittee now.



Wins Of SAE AADL

Static & dynamic architecture in single model

- Improved software process

Validation based on precise semantics

- Validated system architectures

Common architecture notation

- Sub-contractor management

Standardized interchange format

- Tool integration & interoperability

Alignment with UML2.0

- UML profile, MARTE

AIRBUS

esa

axlog

EADS
DEFENCE
& SECURITY

Rockwell
Collins

Tni Europe

Honeywell

THALES

OMG
OBJECT MANAGEMENT GROUP



Carnegie Mellon
Software Engineering Institute

SEI AADL Application & Education

Open source AADL Tool environment (OSATE)

Embedded Engineering With AADL Tutorial

Public two-day course offering by SEI

- *Model-based Engineering with SAE AADL*

Pilot projects with customers

Case studies of Model-based Engineering

AADL User Guide & Embedded Systems Engineering
Handbook



Summary

Model-based system engineering benefits

Predictable runtime characteristics addressed early and throughout life cycle greatly reduces integration and maintenance effort

Benefits of AADL as SAE standard

AADL as standard provides confidence in language stability, broad adoption, and strong tool support

In summary:

Model-based, architecture driven software system engineering is critical to predictably developing and maintaining large-scale systems. Architecture analysis early and throughout the life cycle improves predictability of non-functional properties of mission-critical systems.

The SAE AADL – as an industry standard – provides a stable common framework for contractors to cooperatively evolve large-scale systems and for tool vendors to provide tools for a common architecture representation.



Carnegie Mellon
Software Engineering Institute

Contact

E-mail: info@aadl.info

AADL Web site – www.aadl.info

SEI Web site – www.sei.cmu.edu

SAE AADL Subcommittee chair:

Bruce Lewis (US Army AMRDEC)

bruce.a.lewis@us.army.mil