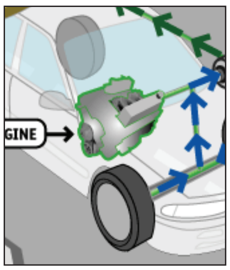
Software Engineering Institute | Carnegie Mellon

More Accurate Prediction of Resource Consumption

Through Architecture Analysis using Model-Based Engineering Tools

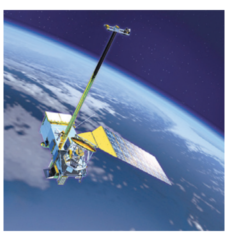
Resource Use Concerns

System designers are concerned about resource contention and consumption.

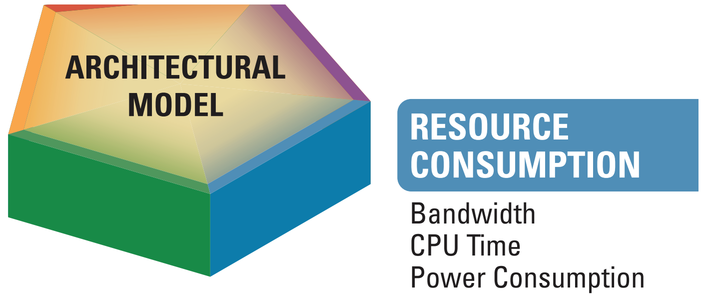


Embedded and real-time systems, such as the braking and ignition systems in an automobile, must share limited resources (e.g., memory, power, and bandwidth).

Shared limited resources result in strong requirements for controlled resource utilization to ensure predictable order, timing, and completion of tasks. These interaction dynamics must often be stringently assured across multiple resources and types of resources to ensure that the system will meet its mission to a specified level of reliability.



More than proper system behavior is of concern. For instance, devices that use less power contribute to lower cost, lower weight, longer battery life, and the like. Power consumption is critical for hand-held, battery-powered mobile computing devices, satellite systems, and aircraft—but important anywhere power supplies or fuses limit available power.



The SEI uses model-based engineering tools, methods, and techniques to more accurately predict system resource consumption.

To deal with problems resulting from contention and consumption issues, system developers typically might have to re-write code to fit limits for memory or CPU cycles, or consider a slower signal processing rate for a controller to reduce power, or add processors to complete the workload on time.

SEI model-based engineering (MBE) tools, methods, and techniques give system designers insight into resource budget and use.

Predicting and Evaluating Resource Consumption

The SEI's MBE tools permit resource budgeting and allocation analysis at multiple levels of architecture model refinement, early and throughout the development life cycle. An initial model may consist of top-level software and the hardware model. The MBE tools roll up budget totals and compare them with available system capacity. If application components are bound to the hardware, a hardware-component-specific capacity comparison can be performed.

As subsystems are elaborated, the budget can be revisited. When expanded to the thread level, thread execution rates and worst-case execution times are compared against the budget, while also providing the basis for scheduling analysis.

This modeling also permits analysis of bandwidth budget and capacity comparison for network traffic and power consumption of hardware components.

Using this information, system designers can, in a more automated, integrated, well defined, and traceable way, verify the adequacy of system resources, evaluate options, and make choices.

Resource Analysis Concern	SEI MBE	Answer
Resource allocation	✔	Account for allocation decisions of application components to processors and to memory and any specification of actual resource usage
Network bandwidth analysis	✔	Provide analysis based on inferred and explicit binding of connections to buses
Power consumption	✔	Check connections for mismatches in supplied and required power characteristics and modeling of the power capacity of a physical system entity
Scheduling system partitions as virtual processors	✔	Support management of end-to-end latency

Read more

ASSIP Study of Real-Time Safety-Critical Embedded Software-Intensive System Engineering Practice (CMU/SEI-2008-SR-001)

Modeling System Architectures Using the Architecture Analysis and Design Language (AADL)

For Course Registration
www.sei.cmu.edu/training/p72.cfm

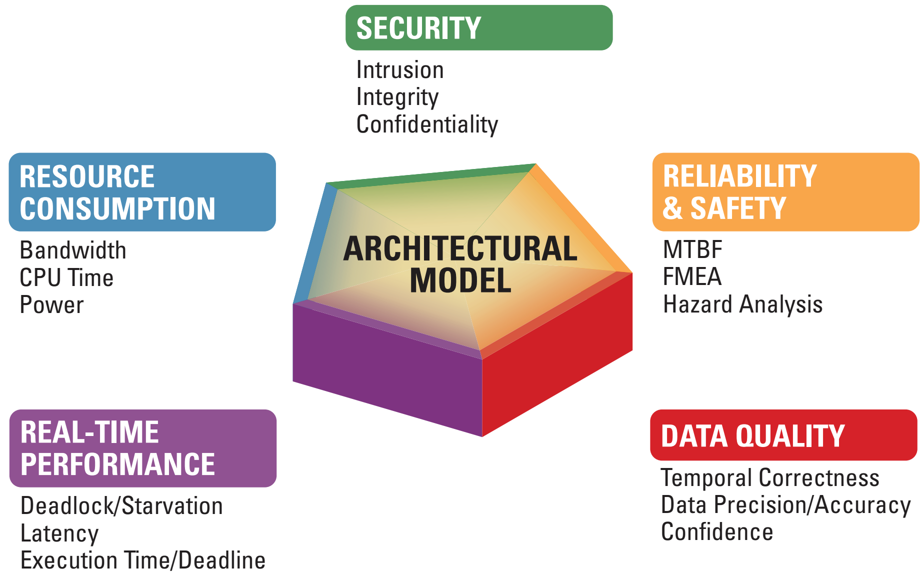
This course may also be offered by arrangement at customer sites. Email course-info@sei.cmu.edu or call +1 412-268-7622 for details.

For More Information
Customer Relations
Phone: +1 412-268-5800
FAX: +1 412-268-6257
customer-relations@sei.cmu.edu

Software Engineering Institute
4500 Fifth Avenue
Pittsburgh, PA 15313-2612
www.sei.cmu.edu

More Accurate Prediction of Resource Consumption

Through Architecture Analysis using Model-Based Engineering Tools



The SEI MBE Toolkit

The SEI uses the ***Architecture Analysis & Design Language (AADL)*** to document a system architecture and provide a platform for multiple analyses.

The AADL, an international industry standard, supports multiple analyses from a single architectural model, enables modeling and analysis throughout the life cycle, and provides analysis of runtime behavior (what) rather than functional behavior (how).

Through its ***XML/XMI interchange format,*** the AADL supports model interchange and tool chaining. And, the SEI offers the ***Open Source AADL Tool Environment (OSATE)*** set of analysis plug-ins. The OSATE Resource Budget and Allocations Analysis plug-in allows users to perform resource budgeting for processors, memory, and network bandwidth and analyze whether the capacity is exceeded by the budgets. The purpose of this plug-in is to support resource-related analysis early and throughout the development life cycle.

The SEI has developed OSATE as a set of plug-ins for processing AADL models that includes:

- a syntax-sensitive text editor, with integrated error reporting
- a parser and semantic checker for textual AADL with conversion into AADL XML
- an unparser for AADL XML to textual AADL conversion
- support for multi-enterprise development through a version control system interface

The AADL also can be used with

- UML state and process charts through its UML profile
- the SEI Architecture Tradeoff Analysis Method®, to drill into root causes and develop quantitative analysis
- assurance cases, to support claims made about the safety, security, or reliability of a system

Prevent System Integration Problems and Simplify Life-Cycle Support

Modeling of system quality attributes is often done—when it is done—with low-fidelity software models and disjointed architectural specifications by various engineers using their own specialized notations.

These models are typically not maintained or documented throughout the life cycle, making it difficult to predict the impact of change on attributes that cut across system functionality. The unanticipated effects of design approaches or changes are discovered only late in the life cycle, when they are much more expensive to resolve.

Analysis of a *system architecture model* offers a better way to predict the behavior of quality attributes. The SEI approach to model-based engineering (MBE) allows analysis

- using a single architecture model
- early and often in the development life cycle or on an existing system architecture
- at different architecture refinement levels
- along diverse architectural aspects such as behavior and throughput

Integration is a major cost and risk in complex systems. System understanding is a major cost driver during system maintenance. Proper use of MBE tools can prevent system integration problems and simplify life-cycle support.

System Architecture Modeling and Analysis

The Carnegie Mellon® Software Engineering Institute (SEI) provides technical assistance and guidance to transform the architectural design process from one based on human evaluation to one based on automated analysis.1

This analysis includes

- validating system quality attributes early in the design phase
- facilitating system integration
- conducting impact and tradeoff analysis using architecture models

For predicting and validating specific nonfunctional properties using model-based engineering, the SEI can help you to

- perform analysis that gives greater assurance that deployment will succeed
- evaluate fault tolerance of architectures
- adopt analytical resource models to validate performance behavior, power consumption, and network bandwidth usage
- model security aspects of architecture
- conduct analysis to guide localized architectural change
- validate data quality requirements such as temporal correctness, accuracy/precision, and confidence

Put MBE to work on your projects quickly!
Register for training by the Software Engineering Institute.
Go to www.sei.cmu.edu/training/p72.cfm.

1 One large defense contractor, for instance, blames human interpretation of the complexity involved with embedded systems for decreasing productivity to 6 or fewer lines of code per day.

The Software Engineering Institute (SEI) is a federally funded research and development center sponsored by the U.S. Department of Defense and operated by Carnegie Mellon University.

0003-2010-03