

SOFTWARE ARCHITECTURE PUBLICATIONS

September 2018

Contents

Definitions and General Treatment	1
Representation and Documentation	3
Evaluation and Analysis	6
Architecture-Based Design and Development	15
Quality Attribute Workshops (QAWs)	22
Architecture Tradeoff Analysis Method (ATAM)	24
Acquisition	28

Definitions and General Treatment

Architecting Service-Oriented Systems

Philip Bianco, Grace Lewis, Paulo Merson, & Soumya Simanta

Technical Note (CMU/SEI-2011-TN-008), August 2011

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=9829>

This report presents guidelines for architecting service-oriented systems and the effect of architectural principles on system quality attributes.

A Research Agenda for Service-Oriented Architecture (SOA): Maintenance and Evolution of Service-Oriented Systems

Grace Lewis, Dennis B. Smith, & Kostas Kontogiannis

Technical Note (CMU/SEI-2010-TN-003), March 2010

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=9285>

This 2010 report describes the agenda of an SEI-led group that was formed to explore the business, engineering, and operations aspects of service-oriented architecture.

Proceedings of the International Workshop on the Foundations of Service-Oriented Architecture (FSOA 2007)

Grace Lewis & Dennis B. Smith

Special Report (CMU/SEI-2008-SR-011), June 2008

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=8477>

This report presents the results of the Foundations of Software-Oriented Architecture (FSOA) workshop held at the Third International Conference on Interoperability for Enterprise Software and Applications (I-ESA 2007).

Evaluating a Service-Oriented Architecture

Philip Bianco, Rick Kotermanski, & Paulo Merson

Technical Report (CMU/SEI-2007-TR-015), September 2007

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=8443>

This report contains technical information about SOA design considerations and tradeoffs that can help the architecture evaluator to identify and mitigate risks in a timely and effective manner.

A Comparison of Requirements Specification Methods from a Software Architecture Perspective

Len Bass, John K. Bergey, Paul C. Clements, Paulo Merson, Ipek Ozkaya, & Raghvinder Sangwan

Technical Report (CMU/SEI-2006-TR-013), August 2006

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=8121>

In this report, five methods for the elicitation and expression of requirements are evaluated with respect to their ability to capture architecturally significant requirements.

Software Architecture: An Executive Overview

Paul C. Clements & Linda M. Northrop

Technical Report (CMU/SEI-96-TR-003), February 1996

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=12509>

This report summarizes software architecture for an intended audience of mid to senior level management.

The Software Architecture Renaissance

Paul C. Clements & Paul Kogut

Article, November 1994

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=30079>

The increasing importance of software in systems is also driving the software architecture renaissance. This article provides a brief overview of some important architecture related efforts.

Software Architecture in Practice (2nd Edition)

Len Bass, Paul C. Clements, & Rick Kazman

SEI Book (ISBN: 0321154959), April 2003

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=30659>

This book provides an easily accessible overview of software architecture. Anyone requiring a comprehensive overview of the software architecture field will benefit from this book.

The Golden Age of Software Architecture: A Comprehensive Survey

Mary Shaw & Paul Clements

Institute for Software Research Technical Report (CMU-ISRI-06-101), February 2006

<http://reports-archive.adm.cs.cmu.edu/anon/isri2006/abstracts/06-101.html>

This paper updates an invited keynote for *ICSE-23*, "The Coming-of-Age of Software Architecture Research" by Mary Shaw. It is also the basis for "The Golden Age of Software Architecture," published in *IEEE Software*, March/April 2006.

Representation and Documentation

Documenting Software Architectures: Views and Beyond, Second Edition

Felix Bachmann, Len Bass, Paul C. Clements, David Garlan, James Ivers, Reed Little, Paulo

Merson, Robert Nord, & Judith A. Stafford

SEI Book (ISBN: 0321552687), October 2010

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=30386>

This book provides the most complete and current guidance on how to capture a software architecture in a commonly understandable form.

Comparing the SEI's Views and Beyond Approach for Documenting Software Architectures with ANSI-IEEE 1471-2000

Paul C. Clements

Technical Note (CMU/SEI-2005-TN-017), July 2005

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=7427>

This report summarizes the V&B and 1471 approaches to architecture description, and shows how a software architecture document prepared using V&B can be made compliant with 1471.

Creating and Using Software Architecture Documentation Using Web-Based Tool Support

Judith A. Stafford

Technical Note (CMU/SEI-2004-TN-037), September 2004

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=6997>

This report describes a design prototype that demonstrates a web-based approach to creating, communicating, and using software architecture throughout the life of the system.

Documenting Component and Connector Views with UML 2.0

James Ivers, Paul C. Clements, David Garlan, Robert Nord, Bradley Schmerl, & Oviedo Silva
(Carnegie Mellon School of Computer Science)

Technical Report (CMU/SEI-2004-TR-008), April 2004

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=7095>

This 2004 report explores how changes in UML 2.0 affect UML's suitability for documenting component and connector views.

Documenting Software Architecture: Documenting Interfaces

Felix Bachmann, Len Bass, Paul C. Clements, David Garlan, James Ivers, Reed Little, Robert Nord, & Judith A. Stafford

Technical Note (CMU/SEI-2002-TN-015), June 2002

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5939>

This report provides guidance for documenting the interfaces to software elements.

Documenting Software Architecture: Documenting Behavior

Felix Bachmann, Len Bass, Paul C. Clements, David Garlan, James Ivers, Reed Little, Robert Nord, & Judith A. Stafford

Technical Note (CMU/SEI-2002-TN-001), January 2002

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5847>

This report describes ways to document the behavior of systems, subsystems, and components of software architecture.

Documenting Software Architectures: Organization of Documentation Package

Felix Bachmann, Len Bass, Paul C. Clements, David Garlan, James Ivers, Reed Little, Robert Nord, & Judith A. Stafford

Technical Note (CMU/SEI-2001-TN-010), August 2001

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5471>

This comprehensive handbook outlines how to produce high-quality documentation for software architectures.

SEI Workshop on Software Architecture Representation, 16-17 January 2001

Felix Bachmann, Paul C. Clements, David Garlan, James Ivers, Reed Little, Robert Nord, & Judith A. Stafford

Special Report (CMU/SEI-2001-SR-010), May 2001

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5355>

This report summarizes the discussions from the 2001 Architecture Representation Workshop, where five leading software architects and practitioners were invited to discuss aspects of the architecture representation with senior members of the SEI technical staff.

Evaluation and Analysis

Adaptive Flow Control for Enabling Quality of Service in Tactical Ad Hoc Wireless Networks

Jeffrey Hansen, Scott Hissam, B. Craig Meyers, Edwin J. Morris, Daniel Plakosh, Soumya Simanta, & Lutz Wrage

Technical Report (CMU/SEI-2010-TR-030), December 2010

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=9633>

The network infrastructure for users such as emergency responders or warfighters is wireless, ad hoc, mobile, and lacking in sufficient bandwidth. This report documents the results from 18 experiments to investigate Adaptive Quality of Service, an approach to enable applications to fulfill their missions despite tactical network infrastructure limitations.

A Workshop on Analysis and Evaluation of Enterprise Architectures

[John Klein & Michael J. Gagliardi](#)

Technical Note (CMU/SEI-2010-TN-023), November 2010

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=9363>

This report summarizes a workshop on the analysis and evaluation of enterprise architectures that was held at the SEI in April of 2010.

COVERT: A Framework for Finding Buffer Overflows in C Programs via Software Verification

[Sagar Chaki, Arie Gurfinkel](#)

Technical Report (CMU/SEI-2010-TR-029), August 2010

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=9627>

In this report, the authors present COVERT, an automated framework for finding buffer overflows in C programs using software verification tools and techniques.

Case Study: Model-Based Analysis of the Mission Data System Reference Architecture

Peter H. Feiler, David P. Gluch, Kurt Woodham (L-3 Communications-Titan Group)

Technical Report (CMU/SEI-2010-TR-003), May 2010

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=9407>

This report describes how AADL support an instantiation of a reference architecture, address architectural themes, and provide a foundation for the analysis of performance elements and system assurance concerns.

System Architecture Virtual Integration: An Industrial Case Study

Peter H. Feiler, Jörgen Hansson (University of Skovde), Dionisio de Niz, Lutz Wrage

Technical Report (CMU/SEI-2009-TR-017), November 2009

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=9145>

This report introduces key concepts of the SAVI paradigm and discusses the series of development scenarios used in a POC demonstration to illustrate the feasibility of improving the quality of software-intensive aircraft systems.

Evaluating a Service-Oriented Architecture

Philip Bianco, Rick Kotermanski, & Paulo Merson

Technical Report (CMU/SEI-2007-TR-015), September 2007

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=8443>

This report contains technical information about SOA design considerations and tradeoffs that can help the architecture evaluator to identify and mitigate risks in a timely and effective manner.

Quality-Attribute-Based Economic Valuation of Architectural Patterns

Ipek Ozkaya, Rick Kazman, & Mark H. Klein

Technical Report (CMU/SEI-2007-TR-003), May 2007

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=8307>

This report shows how an analysis of the options embodied within architectural patterns allows a software and system architect or manager to make reasoned choices about the future value of design decisions.

Risk Themes Discovered Through Architecture Evaluations

Len Bass, Robert Nord, William G. Wood, David Zubrow

Technical Report (CMU/SEI-2006-TR-012), September 2006

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=8111>

This 2006 report analyzes the output of 18 evaluations conducted using the Architecture Tradeoff Analysis (ATAM). The goal of the analysis was to find patterns in the risk themes identified during those evaluations.

Toward Measures for Software Architectures

Gary Chastek, Robert Ferguson

Technical Note (CMU/SEI-2006-TN-013), March 2006

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=7893>

In this 2006 report, the authors describe the results of a preliminary investigation into measures for software architecture.

Acquiring Evolving Technologies: Web Services Standards

Harry L. Levinson & Liam O'Brien

Technical Note (CMU/SEI-2006-TN-001), February 2006

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=7847>

This technical note discusses some of the challenges of using Web services standards and presents the results generated by an assessment tool used to track the appropriateness of using this technology.

Quality Attributes and Service-Oriented Architectures

Liam O'Brien, Len Bass, & Paulo Merson

Technical Note (CMU/SEI-2005-TN-014), September 2005

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=7405>

This report examines the relationship between service-oriented architectures (SOAs) and quality attributes.

Reasoning Frameworks

Len Bass, James Ivers, Mark H. Klein, & Paulo Merson

Technical Report (CMU/SEI-2005-TR-007), July 2005

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=7637>

This report describes a vehicle for encapsulating the quality attribute knowledge needed to understand a system's quality behavior as a reasoning framework that can be used by nonexperts.

Discovering Architectures from Running Systems: Lessons Learned

Hong Yan, Jonathan Aldrich, David Garlan, Rick Kazman, & Bradley Schmerl

Technical Report (CMU/SEI-2004-TR-016), December 2004

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=7165>

This report describes a technique that uses automatically generated runtime observations of an executing system to construct an architectural view of the system.

Integrating Software-Architecture-Centric Methods into Extreme Programming (XP)

Robert Nord, James E. Tomayko, & Rob Wojcik

Technical Note (CMU/SEI-2004-TN-036), September 2004

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=6989>

The report presents a summary of XP (Extreme Programming) and examines the potential uses of the SEI's architecture-centric methods.

SACAM: The Software Architecture Comparison Analysis Method

Christoph Stoermer, Felix Bachmann, & Chris Verhoef

Technical Report (CMU/SEI-2003-TR-006), December 2003

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=6605>

The report describes SACAM, a method that provides rationale for an architecture selection process by comparing the fitness of architecture candidates for required systems.

Integrating the Architecture Tradeoff Analysis Method (ATAM) with the Cost Benefit Analysis Method (CBAM)

Robert Nord, Mario R. Barbacci, Paul C. Clements, Rick Kazman, Mark H. Klein, Liam O'Brien, & James E. Tomayko

Technical Note (CMU/SEI-2003-TN-038), December 2003

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=6557>

This technical note reports on a proposal to integrate the SEI ATAM (Architecture Tradeoff Analysis Method) and the CBAM (Cost Benefit Analysis Method).

Quality Attribute Workshops (QAWs), Third Edition

Mario R. Barbacci, Robert J. Ellison, Anthony J. Lattanze, Judith A. Stafford, Charles B. Weinstock, & William G. Wood

Technical Report (CMU/SEI-2003-TR-016), October 2003

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=6687>

This report describes the newly revised QAW (Quality Attribute Workshop) and describes potential uses of the refined scenarios generated during it.

Interactions Among Techniques Addressing Quality Attributes

Hernan R. Eguluz & Mario R. Barbacci

Technical Report (CMU/SEI-2003-TR-003), June 2003

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=6587>

This report provides software architects a chart for determining the relationships among techniques that promote different architectural qualities.

SEI Architecture Analysis Techniques and When to Use Them

Mario R. Barbacci

Technical Note (CMU/SEI-2002-TN-005), October 2002

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5883>

When analyzing system and software architectures, the Quality Attribute Workshop (QAW) and the Architecture Tradeoff Analysis Method (ATAM) can be used in combination to obtain early and continuous benefits.

Illuminating the Fundamental Contributors to Software Architecture Quality

Felix Bachmann, Len Bass, & Mark H. Klein

Technical Report (CMU/SEI-2002-TR-025), August 2002

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=6193>

This 2002 report presents the basic concepts of analysis models for two quality attributes—modifiability and performance, identifies a collection of tactics that can be used to control responses within those models, and discusses how to analyze the models in terms of these tactics.

Use of Quality Attribute Workshops (QAWs) in Source Selection for a DoD System Acquisition: A Case Study

John K. Bergey & William G. Wood

Technical Note (CMU/SEI-2002-TN-013), June 2002

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5933>

This case study outlines how a DoD organization used architecture analysis and evaluation in a major system acquisition to reduce program risk.

Using Economic Considerations to Choose Among Architecture Design Alternatives

Jai Asundi, Rick Kazman, & Mark H. Klein

Technical Report (CMU/SEI-2001-TR-035), December 2001

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5785>

The SEI developed the CBAM (Cost Benefit Analysis Method), which incorporates the costs and benefits of architectural design decisions and provides an effective means of making such decisions. This paper reports on the application of this method to a real world case study.

Active Reviews for Intermediate Designs

Paul C. Clements

Technical Note (CMU/SEI-2000-TN-009), August 2000

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5119>

This 2000 technical note describes Active Review for Intermediate Designs (ARID), a piloted software design review technique.

Using Quality Attribute Workshops to Evaluate Architectural Design Approaches in a Major System Acquisition: A Case Study

John K. Bergey, Mario R. Barbacci, & William G. Wood

Technical Note (CMU/SEI-2000-TN-010), July 2000

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5123>

This report describes a series of Quality Attribute Workshops (QAWs) that were conducted on behalf of a government agency during its competitive acquisition of a complex, tactical, integrated command and control system.

Architectural Evaluation of Collaborative Agent-Based Systems

Steve Woods & Mario R. Barbacci

Technical Report (CMU/SEI-99-TR-025), October 1999

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=13525>

This report identifies features in agent-based systems that could be used to classify agent-system architectures and to guide the generation of scenarios applicable to these architectures.

Principles for Evaluating the Quality Attributes of a Software Architecture

Mario R. Barbacci, Mark H. Klein, & Charles B. Weinstock

Technical Report (CMU/SEI-96-TR-036), May 1997

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=12681>

This report describes a few principles for analyzing a software architecture to determine if it exhibits certain quality attributes.

Recommended Best Industrial Practice for Software Architecture Evaluation

Gregory Abowd, Len Bass, Paul C. Clements, Rick Kazman, Linda M. Northrop, & Amy Zaremski

Technical Report (CMU/SEI-96-TR-025), January 1997

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=12653>

This report details the results of two workshops on software architecture evaluation, held at the SEI in 1996.

Quality Attributes

Mario R. Barbacci, Mark H. Klein, Thomas A. Longstaff, & Charles B. Weinstock

Technical Report (CMU/SEI-95-TR-021), December 1995

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=12433>

This report describes efforts to develop a unifying approach for reasoning about multiple software quality attributes.

SAAM: A Method for Analyzing the Properties of Software Architectures

Gregory Abowd, Len Bass, Rick Kazman, & Mike Webb (Texas Instruments)

White Paper, May 2007

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=29288>

This paper describes three perspectives by which we can understand the description of a software architecture and proposes a five-step method for analyzing software architectures called SAAM (Software Architecture Analysis Method).

An Approach to Software Architecture Analysis for Evolution and Reusability

Sonia Bot (Nortel), Kalai Kalaichelvan (Nortel), Rick Kazman, & Chung-Horng Lung (Nortel)

White Paper, November 1997

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=29765>

This paper presents an approach to capturing and assessing software architectures for evolution and reuse.

Scenario-Based Analysis of Software Architecture

Gregory Abowd, Len Bass, Paul C. Clements, & Rick Kazman

White Paper, November 1996

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=29910>

This paper presents an experiential case study illustrating the methodological use of scenarios to gain architecture-level understanding and predictive insight into large, real-world systems in various domains.

Assessing the Quality of Large, Software-Intensive Systems: A Case Study

Alan W. Brown, David J. Carney, Paul C. Clements, B. Craig Meyers, Dennis B. Smith, Nelson W. Weiderman, & William G. Wood

White Paper, September 1995

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=30012>

This paper presents a case study in carrying out an audit of a large, software-intensive system.

An Architectural Analysis Case Study: Internet Information Systems

Gregory Abowd, Len Bass, Paul C. Clements, & Rick Kazman

White Paper, April 1995

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=30057>

This paper presents a method for analyzing systems for nonfunctional qualities from the perspective of their software architecture and applies this method to the field of Internet information systems (IISs).

Evaluating Software Architectures: Methods and Case Studies

Paul C. Clements, Rick Kazman, & Mark H. Klein

SEI Book (ISBN: 0-201-70482-X), October 2001

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=30698>

This book is a comprehensive guide to software architecture evaluation, describing specific methods that can quickly and inexpensively mitigate enormous risk in software projects.

A Case Study in Assessing the Maintainability of Large, Software-Intensive Systems

Alan W. Brown, David J. Carney, & Paul C. Clements

White Paper, March 1995

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=18976>

This paper presents a case study in assessing the maintainability of a large, software intensive system. The techniques used are described, and their strengths and weaknesses discussed.

Making Architecture Reviews Work in the Real World

Rick Kazman & Len Bass

IEEE Software Article (ISSN: 0740-7459), January/February

<https://www.computer.org/csdl/mags/so/2002/01/s1067-abs.html>

This article explores the non-technical aspects of formal architecture reviews—social, psychological, and managerial.

Quality-Attribute-Based Economic Valuation of Architectural Patterns

Ipek Ozkaya, Rick Kazman, & Mark H. Klein

Technical Report (CMU/SEI-2007-TR-003), May 2007

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=8307>

This report shows how an analysis of the options embodied within architectural patterns allows a software and system architect or manager to make reasoned choices about the future value of design decisions.

Using SAAM: An Experience Report

M. DeSimone, Rick Kazman. Proceedings of CASCOM '95, Toronto, ON), November 1995, pp. 251-261.

Architecture-Based Design and Development

Architecting Service-Oriented Systems

Philip Bianco, Grace Lewis, Paulo Merson, & Soumya Simanta

Technical Note (CMU/SEI-2011-TN-008), August 2011

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=9829>

This report presents guidelines for architecting service-oriented systems and the effect of architectural principles on system quality attributes.

Combining Architecture-Centric Engineering with the Team Software Process

Robert Nord, Jim McHale, & Felix Bachmann

Technical Report (CMU/SEI-2010-TR-031), December 2010

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=9649>

ACE methods and the TSP provides an iterative approach for delivering high quality systems on time and within budget. The combined approach helps organizations that must set an architecture/developer team in motion using mature, disciplined engineering practices that produce quality software quickly.

Realizing and Refining Architectural Tactics: Availability

James Scott & Rick Kazman

Technical Report (CMU/SEI-2009-TR-006), August 2009

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=9087>

Tactics are fundamental elements of software architecture that an architect employs to meet a system's quality requirements. This report describes an updated set of tactics that enable the architect to build availability into a system.

Using Aspect-Oriented Programming to Enforce Architecture

Paulo Merson

Technical Note (CMU/SEI-2007-TN-019), September 2007

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=8247>

This report illustrates how to use AOP (aspect-oriented programming) to ensure conformance to architectural design, proper use of design patterns and programming best practices, conformance to coding policies and naming conventions.

Using ArchE in the Classroom: One Experience

John McGregor, Felix Bachmann, Len Bass, Philip Bianco, & Mark H. Klein

Technical Note (CMU/SEI-2007-TN-001), September 2007

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=8191>

The ArchE (Architecture Expert) tool serves as a software architecture design assistant. This report describes the use of a pre-alpha release of ArchE in a graduate-level software architecture class at Clemson University.

Modifiability Tactics

Felix Bachmann, Len Bass, & Robert Nord

Technical Report (CMU/SEI-2007-TR-002), September 2007

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=8299>

This report describes how architectural tactics are based on the parameters of quality attribute models.

A Practical Example of Applying Attribute-Driven Design (ADD), Version 2.0

William G. Wood

Technical Report (CMU/SEI-2007-TR-005), February 2007

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=8319>

This 2007 report describes an example application of the ADD method, an approach to defining a software architecture in which the design process is based on the quality attribute requirements the software must fulfill.

Attribute-Driven Design (ADD), Version 2.0

Rob Wojcik, Felix Bachmann, Len Bass, Paul C. Clements, Paulo Merson, Robert Nord, & William G. Wood

Technical Report (CMU/SEI-2006-TR-023), November 2006

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=8147>

This report revises the steps of the Attribute-Driven Design (ADD) method and offers practical guidelines for carrying out each step.

Elements of a Usability Reasoning Framework

Jinhee Lee, Len Bass

Technical Note (CMU/SEI-2005-TN-030), September 2005

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=7491>

This note describes an ARL implementation of two usability scenarios: displaying progress feedback and allowing cancel.

Discovering Architectures from Running Systems: Lessons Learned

Hong Yan, Jonathan Aldrich, David Garlan, Rick Kazman, & Bradley Schmerl

Technical Report (CMU/SEI-2004-TR-016), December 2004

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=7165>

This report describes a technique that uses automatically generated runtime observations of an executing system to construct an architectural view of the system.

Security and Survivability Reasoning Frameworks and Architectural Design Tactics

Robert J. Ellison, Andrew P. Moore, Len Bass, Mark H. Klein, & Felix Bachmann

Technical Note (CMU/SEI-2004-TN-022), September 2004

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=6949>

In this report, the authors describe an approach to disciplined software architecture design for the related quality attributes of security and survivability.

Integrating Software-Architecture-Centric Methods into the Rational Unified Process

Rick Kazman, Philippe Kruchten, Robert Nord, & James E. Tomayko

Technical Report (CMU/SEI-2004-TR-011), July 2004

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=7133>

This report presents a summary of the RUP (Rational Unified Process) and examines the potential uses of the SEI's architecture-centric methods.

Preliminary Design of ArchE: A Software Architecture Design Assistant

Felix Bachmann, Len Bass, and Mark H. Klein

Technical Report (CMU/SEI-2003-TR-021), September 2003

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=6751>

This 2003 report presents a procedure for moving from a set of quality attribute scenarios to an architecture design that satisfies those scenarios.

Deriving Architectural Tactics: A Step Toward Methodical Architectural Design

Felix Bachmann, Len Bass, & Mark H. Klein

Technical Report (CMU/SEI-2003-TR-004), March 2003

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=6593>

This 2003 technical report provides the status on the work being done by the SEI to understand the relationship between quality requirements and architectural design.

Analyzing Enterprise JavaBeans Systems Using Quality Attribute Design Primitives

Anna Liu, Len Bass, & Mark H. Klein

Technical Note (CMU/SEI-2001-TN-025), October 2001

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5555>

This report introduces the notion of quality attribute design primitives, which are architectural building blocks that target the achievement of one or sometimes several quality attribute requirements.

Achieving Usability Through Software Architecture

Len Bass, Bonnie E. John, & Jesse Kates

Technical Report (CMU/SEI-2001-TR-005), March 2001

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5605>

This paper outlines an approach to improving the usability of software systems by means of software architectural decisions.

Quality Attribute Design Primitives

Len Bass, Mark H. Klein, & Felix Bachmann

Technical Note (CMU/SEI-2000-TN-017), December 2000

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5139>

This report addresses mechanisms that significantly affect quality attribute behavior and have sufficient content for analysis.

An Application of the Architecture-Based Design Method to the Electronic House

Felix Bachmann, Len Bass, & Mark H. Klein

Special Report (CMU/SEI-2000-SR-009), October 2000

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5057>

This report elaborates an example of the application of the ABD (Architecture-Based Design) method to designing software architecture.

The Architecture Based Design Method

Felix Bachmann, Len Bass, Gary Chastek, Patrick Donohoe, & Fabio Peruzzi

Technical Report (CMU/SEI-2000-TR-001), January 2000

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5147>

This paper presents the Architecture Based Design (ABD) method for designing the high-level software architecture for a product line or long-lived system.

Architecture-Based Development

Len Bass & Rick Kazman

Technical Report (CMU/SEI-99-TR-007), April 1999

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=13385>

This report presents a description of architecture-centric system development.

Quality Attribute Design Primitives and the Attribute Driven Design Method

Felix Bachmann, Len Bass, & Mark H. Klein

White Paper, October 2001

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=29604>

This paper discusses the understanding of quality attributes and their application to the design of a software architecture.

Tool Support for Architecture Analysis and Design

Rick Kazman

White Paper, January 1996

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=29989>

This position paper first presents a set of requirements that an ideal tool for architectural design and analysis, and then presents a tool—called SAAMtool—that meets most, but not all, of these requirements. Understanding Architectural Influences and Decisions in Large-System Projects

Paul C. Clements

White Paper, April 1995

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=30043>

This paper discusses the approach taken in a pilot study to uncover the correlation, if any, between architectural influences and architectural decisions in large-scale, software-intensive development projects.

From Domain Models to Architectures

Paul C. Clements

White Paper, January 1994

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=30087>

This white paper was presented at the Workshop on Software Architecture, USC Center for Software Engineering, Los Angeles, 1994, by Paul Clements.

Building Systems from Commercial Components

Scott Hissam, Robert C. Seacord, & Kurt C. Wallnau

SEI Book (ISBN: 0-201-70064-6), July 2001

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=31022>

This book describes specific engineering practices needed to integrate preexisting components with preexisting specifications successfully, illustrating the techniques described with case studies and examples.

The Gadfly: An Approach to Architectural-Level System Comprehension

Paul C. Clements, Edwin J. Morris, & Kurt C. Wallnau

White Paper, March 1996

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=19882>

This paper describes the Gadfly, an approach for developing narrowly-focused, reusable domain models that can be integrated and (re)used to aid in the process of top-down system comprehension.

Quality Attribute Workshops (QAWs)

Impact of Army Architecture Evaluations

Robert Nord, John K. Bergey, Stephen Blanchette, Jr., & Mark H. Klein

Special Report (CMU/SEI-2009-SR-007), April 2009

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=8859>

This 2009 report describes the results of a study of the impact that the ATAM evaluations and QAWs had on Army programs.

Integrating the Quality Attribute Workshop (QAW) and the Attribute-Driven Design (ADD) Method

Robert Nord, William G. Wood, & Paul C. Clements

Technical Note (CMU/SEI-2004-TN-017), July 2004

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=6927>

This technical note reports on a proposal to integrate the SEI Quality Attribute Workshop (QAW) and the SEI Attribute-Driven Design (ADD) method.

Quality Attribute Workshops (QAWs), Third Edition

Mario R. Barbacci, Robert J. Ellison, Anthony J. Lattanze, Judith A. Stafford, Charles B. Weinstock, & William G. Wood

Technical Report (CMU/SEI-2003-TR-016), October 2003

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=6687>

This report describes the newly revised QAW (Quality Attribute Workshop) and describes potential uses of the refined scenarios generated during it.

Use of Quality Attribute Workshops (QAWs) in Source Selection for a DoD System Acquisition: A Case Study

John K. Bergey & William G. Wood

Technical Note (CMU/SEI-2002-TN-013), June 2002

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5933>

This case study outlines how a DoD organization used architecture analysis and evaluation in a major system acquisition to reduce program risk.

Quality Attribute Workshops, 2nd Edition

Mario R. Barbacci, Robert J. Ellison, Anthony J. Lattanze, Judith A. Stafford, Charles B. Weinstock, & William G. Wood

Technical Report (CMU/SEI-2002-TR-019), June 2002

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=6113>

This report clarifies the context in which a QAW (Quality Attribute Workshop) is applicable, provides a rationale for developing the process and describes it in detail, and concludes with a list of lessons learned and a discussion of how these lessons have helped evolve the process to its current state.

Quality Attribute Workshops

Mario R. Barbacci, Robert J. Ellison, Judith A. Stafford, Charles B. Weinstock, & William G. Wood

Technical Report (CMU/SEI-2001-TR-010), May 2001

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5613>

This report describes the QAW (Quality Attribute Workshop) approach, which is a method for evaluating a software-intensive system architecture during the acquisition phase of major programs.

Using Quality Attribute Workshops to Evaluate Architectural Design Approaches in a Major System Acquisition: A Case Study

John K. Bergey, Mario R. Barbacci, & William G. Wood

Technical Note (CMU/SEI-2000-TN-010), July 2000

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5123>

This report describes a series of Quality Attribute Workshops (QAWs) that were conducted on behalf of a government agency during its competitive acquisition of a complex, tactical, integrated command and control system.

Quality Attribute Workshop Participants Handbook

Mario R. Barbacci, Robert J. Ellison, Charles B. Weinstock, & William G. Wood

Special Report (CMU/SEI-2000-SR-001), January 2000

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=4995>

This report describes the 1) process we use to conduct QAW (Quality Attribute Workshop), 2) information required, 3) suggested tools, and 4) expected outcomes of QAWs.

Coming Attractions in Software Architecture

Paul C. Clements

Technical Report (CMU/SEI-96-TR-008), January 1996

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=12539>

This 1996 report identifies a set of promising lines of research related to software architecture and architecture-based system development.

Architecture Tradeoff Analysis Method (ATAM)

Progress Toward an Organic Software Architecture Capability in the U.S. Army

Stephen Blanchette, Jr. & John K. Bergey

Technical Report (CMU/SEI-2007-TR-010), June 2007

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=8403>

This 2007 report describes the Software Architecture Initiative of the Army Strategic Software Improvement Program.

Categorizing Business Goals for Software Architectures

Rick Kazman & Len Bass

Technical Report (CMU/SEI-2005-TR-021), December 2005

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=7751>

This report provides a categorization of possible business goals for software-intensive systems, so that individuals have some guidance in the elicitation, expression, and documentation of business goals.

Using the SEI Architecture Tradeoff Analysis Method to Evaluate WIN-T: A Case Study

Paul C. Clements, John K. Bergey, & Dave Mason

Technical Note (CMU/SEI-2005-TN-027), September 2005

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=7467>

This report describes the application of the SEI ATAM (Architecture Tradeoff Analysis Method) to the U.S. Army's Warfighter Information Network-Tactical (WIN-T) system.

Integrating the Architecture Tradeoff Analysis Method (ATAM) with the Cost Benefit Analysis Method (CBAM)

Robert Nord, Mario R. Barbacci, Paul C. Clements, Rick Kazman, Mark H. Klein, Liam O'Brien, & James E. Tomayko

Technical Note (CMU/SEI-2003-TN-038), December 2003

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=6557>

This technical note reports on a proposal to integrate the SEI ATAM (Architecture Tradeoff Analysis Method) and the CBAM (Cost Benefit Analysis Method)

Using the Architecture Tradeoff Analysis Method (ATAM) to Evaluate the Software Architecture for a Product Line of Avionics Systems: A Case Study

Mario R. Barbacci, Paul C. Clements, Anthony J. Lattanze, Linda M. Northrop, & William G. Wood

Technical Note (CMU/SEI-2003-TN-012), July 2003

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=6447>

This 2003 technical note describes an ATAM evaluation of the software architecture for an avionics system developed for the Technology Applications Program Office (TAPO) of the U.S. Army Special Operations Command Office.

Use of the Architecture Tradeoff Analysis Method (ATAM) in Source Selection of Software-Intensive Systems

John K. Bergey, Matt Fisher, & Lawrence G. Jones

Technical Note (CMU/SEI-2002-TN-010), June 2002

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5907>

This report explains the role of software architecture evaluation in a source selection and describes the contractual elements that are needed to support its use.

Using the Architecture Tradeoff Analysis Method to Evaluate a Wargame Simulation System: A Case Study

Lawrence G. Jones & Anthony J. Lattanze

Technical Note (CMU/SEI-2001-TN-022), December 2001

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5535>

This report describes the application of the ATAM (Architecture Tradeoff Analysis Method) to a major wargaming simulation system.

Applicability of General Scenarios to the Architecture Tradeoff Analysis Method

Len Bass, Mark H. Klein, & Gabriel Moreno

Technical Report (CMU/SEI-2001-TR-014), October 2001

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5637>

In this report, we compare the scenarios elicited from five ATAM (Architecture Tradeoff Analysis Method) evaluations with the scenarios used to characterize the quality attributes.

Use of the ATAM in the Acquisition of Software-Intensive Systems

John K. Bergey & Matt Fisher

Technical Note (CMU/SEI-2001-TN-009), September 2001

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5465>

This report discusses the role of software architecture evaluations in a system acquisition and describes the contractual elements that are needed to accommodate architecture evaluations in an acquisition. The report also provides an example of contractual language that incorporates the ATAM as a software architecture evaluation method in a system acquisition.

An Evaluation Theory Perspective of the Architecture Tradeoff Analysis Method (ATAM)

Marta Lopez

Technical Report (CMU/SEI-2000-TR-012), September 2000

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5237>

This report analyzes and identifies the Architecture Tradeoff Analysis Method (ATAM)'s evaluation process and criteria, as well as its data-gathering and synthesis techniques, and more.

ATAM: Method for Architecture Evaluation

Rick Kazman, Mark H. Klein, & Paul C. Clements

Technical Report (CMU/SEI-2000-TR-004), August 2000

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5177>

This report presents technical and organizational foundations for performing architectural analysis, and presents the SEI's ATAM, a technique for analyzing software architectures.

Using the Architecture Tradeoff Analysis Method to Evaluate a Reference Architecture: A Case Study

Brian P. Gallagher

Technical Note (CMU/SEI-2000-TN-007), June 2000

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5109k>

This report describes the application of the ATAM (Architecture Tradeoff Analysis Method) to evaluate a reference architecture for ground-based command and control systems.

Software Architecture Evaluation with ATAM in the DoD System Acquisition Context

John K. Bergey, Matt Fisher, Lawrence G. Jones, & Rick Kazman

Technical Note (CMU/SEI-99-TN-012), September 1999

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=13301>

This report explains the basics of software architecture and software architecture evaluation in a system acquisition context.

Steps in an Architecture Tradeoff Analysis Method: Quality Attribute Models and Analysis

Mario R. Barbacci, Peter H. Feiler, Mark H. Klein, Howard F. Lipson, Thomas A. Longstaff, Charles B. Weinstock, & Jeromy Carrière

Technical Report (CMU/SEI-97-TR-029), May 1998

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=12927>

This paper presents some of the steps in an emerging architecture tradeoff analysis method (ATAM).

Architecture Tradeoff Analysis Method

Rick Kazman, Mark Klein, Mario Barbacci, Howard Lipson, Thomas Longstaff, & S. Jeremy Carrière. Proceedings of ICECCS. Monterey, CA), August 1998.

Acquisition

A Proactive Means for Incorporating a Software Architecture Evaluation in a DoD System Acquisition

John K. Bergey

Technical Note (CMU/SEI-2009-TN-004), July 2009

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=8935>

This technical note provides guidance on how to contractually incorporate architecture evaluations in an acquisition.

Impact of Army Architecture Evaluations

Robert Nord, John K. Bergey, Stephen Blanchette, Jr., & Mark H. Klein

Special Report (CMU/SEI-2009-SR-007), April 2009

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=8859>

This 2009 report describes the results of a study of the impact that the ATAM evaluations and QAWs had on Army programs.

U.S. Army Workshop on Exploring Enterprise, System of Systems, System, and Software Architectures

John K. Bergey, Stephen Blanchette, Jr., Paul C. Clements, Michael J. Gagliardi, Rob Wojcik, William G. Wood, & John Klein

Technical Report (CMU/SEI-2009-TR-008), March 2009

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=9099>

This report confirms that various architectural genres enjoy more commonalities than differences. Each one has its own important knowledge base, and openness among the various architectural tasks within an organization is growing in importance.

Progress Toward an Organic Software Architecture Capability in the U.S. Army

Stephen Blanchette, Jr. & John K. Bergey

Technical Report (CMU/SEI-2007-TR-010), June 2007

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=8403>

This 2007 report describes the Software Architecture Initiative of the Army Strategic Software Improvement Program.

Software Architecture in DoD Acquisition: An Approach and Language for a Software Development Plan

John K. Bergey & Paul C. Clements

Technical Note (CMU/SEI-2005-TN-019), February 2005

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=7431>

This report discusses the Software Development Plan (SDP), providing an example approach and corresponding SDP language that enable software architecture to play a central role in the technical and organizational management of a software development effort.

Software Architecture in DoD Acquisition: A Reference Standard for a Software Architecture Document

John K. Bergey & Paul C. Clements

Technical Note (CMU/SEI-2005-TN-020), February 2005

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=7437>

This report provides a reference standard for a Software Architecture Document (SAD). Acquisition organizations can use this to acquire documentation needed for communicating the architecture design and conducting software architecture evaluations.

DoD Experience with the C4ISR Architecture Framework

William G. Wood & Sholom G. Cohen

Technical Note (CMU/SEI-2003-TN-027), September 2003

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=6527>

This report discusses the context for using the C4ISR AF, the observations made during the interviews about its use, and the strengths and challenges of using it.

DoD Architecture Framework and Software Architecture Workshop Report

William G. Wood, Lyn Uzzle, John Weiler, Art Krummenoehl, Mario R. Barbacci, Paul C. Clements, Steve Palmquist, Huei-Wan Ang, Loring Bernhardt, Fatma Dandashi, David Emery, & Sarah Sheard

Technical Note (CMU/SEI-2003-TN-006), March 2003

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=6399>

This report summarizes the activities of the Workshop on the Department of the 2003 Defense Architecture Framework and Software Architecture workshop.

Use of Quality Attribute Workshops (QAWs) in Source Selection for a DoD System Acquisition: A Case Study

John K. Bergey & William G. Wood

Technical Note (CMU/SEI-2002-TN-013), June 2002

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=5933>

This case study outlines how a DoD organization used architecture analysis and evaluation in a major system acquisition to reduce program risk.

Software Architecture Evaluation with ATAM in the DoD System Acquisition Context

John K. Bergey, Matt Fisher, Lawrence G. Jones, & Rick Kazman

Technical Note (CMU/SEI-99-TN-012), September 1999

<https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=13301>

This report explains the basics of software architecture and software architecture evaluation in a system acquisition context.

Contact Us

Software Engineering Institute
4500 Fifth Avenue, Pittsburgh, PA 15213-2612
Phone: 412/268.5800 | 888.201.4479
Web: www.sei.cmu.edu | www.cert.org
Email: info@sei.cmu.edu

Copyright 2018 Carnegie Mellon University. All Rights Reserved.

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

The view, opinions, and/or findings contained in this material are those of the author(s) and should not be construed as an official Government position, policy, or decision, unless designated by other documentation.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

[DISTRIBUTION STATEMENT A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

Internal use:* Permission to reproduce this material and to prepare derivative works from this material for internal use is granted, provided the copyright and "No Warranty" statements are included with all reproductions and derivative works.

External use:* This material may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other external and/or commercial use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

* These restrictions do not apply to U.S. government entities.

Architecture Tradeoff Analysis Method®, ATAM® and Carnegie Mellon® are registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

DM18-1071
