

Panel: Reflection on 20 Years of Architecture

Rick Kazman



Copyright 2012 Carnegie Mellon University

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

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.

This material has been approved for public release and unlimited distribution except as restricted below.

The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013 and 252.227-7013 Alternate I.

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.



20 Years Ago...

Google scholar

software architecture

Search

[Advanced Scholar Search](#)

[My Citations](#)

Scholar

Articles excluding patents

1960

- 1992

include citations



[Create email alert](#)

Results 1 - 10 of about 88,700. (0.10 sec)

[Foundations for the study of software architecture](#)

DE Pery... - ACM SIGSOFT **Software Engineering Notes**, 1992 - [dl.acm.org](#)

Abstract The purpose of this paper is to build the foundation for **software architecture**. We first develop an intuition for **software architecture** by appealing to several well-established architectural disciplines. On the basis of this intuition, we present a model of **software ...**

[Cited by 1858](#) - [Related articles](#) - [BL Direct](#) - [All 71 versions](#)

[\[PDF\] from uci.edu](#)

[Carnegie Mellon Fulltext](#)

[\[PDF\] Mass-produced software components](#)

MD McIlroy, JM Buxton, P Naur... - **Software Engineering ...**, 1968 - [st.inf.tu-dresden.de](#)

... That allows for extracting the **architecture** and separating it from „components“ ► **Software architecture** and composition have been successful for ■ Developing in the large ... documents? ■ And unify it with **software architecture** / **software** composition? Page 19. Prof. ...

[Cited by 1055](#) - [Related articles](#) - [View as HTML](#) - [All 11 versions](#)

[\[PDF\] from tu-dresden.de](#)

[Software architecture for hard real-time applications: cyclic executives vs. fixed priority executives](#)

CD Locke - **Real-Time Systems**, 1992 - Springer

We contrast the **software architecture** of a hard real-time application using a fixed priority task structure against the **software architecture** of the same system using a cyclic executive structure to satisfy hard real-time deadlines in response to a set of embedded system ...

[Cited by 278](#) - [Related articles](#) - [Carnegie Mellon Links](#) - [All 7 versions](#)

[\[PDF\] from unr.edu.ar](#)

[\[CITATION\] GreatSPN 1.5 software architecture](#)

G Chiola - **Computer Performance Evaluation**, 1991

[Cited by 167](#) - [Related articles](#)

[Studying software architecture through design spaces and rules](#)

TG Lane, JS Herman, U Signature... - 1990 - Citeseer

Abstract. This report argues that the overall structure of **software** systems (" **software architecture**") is usefully studied by constructing design spaces. A design space identifies the key functional and structural choices made in creating a system design, and it ...

[Cited by 74](#) - [Related articles](#) - [Cached](#) - [All 7 versions](#)

[ASAX: Software architecture and rule-based language for universal audit trail analysis](#)

N Habra, B Charlier, A Mounji... - **Computer Security—ESORICS ...**, 1992 - Springer

After a brief survey of the problems related to audit trail analysis and of some approaches to deal with them, the paper outlines the project ASAX which aims at providing an advanced tool to support such analysis. One key feature of ASAX is its elegant **architecture** build on ...

[Cited by 136](#) - [Related articles](#) - [BL Direct](#) - [All 8 versions](#)



Today

Google scholar

software architecture

Search

[Advanced Scholar Search](#)

[My Citations](#)

Scholar

Articles excluding patents

anytime

include citations



[Create email alert](#)

Results 1 - 10 of about 2,380,000. (0.20 sec)

[\[BOOK\] Software architecture in practice](#)

[L Bass, P Clements...](#) - 2003 - [dl.acm.org](#)

Our goals for the first edition were threefold. First, we wanted to show through authentic case studies actual examples of **software** architectures solving real-world problems. Second, we wanted to establish and show the strong connection between an **architecture** and an ...

[Cited by 4578](#) - [Related articles](#) - [Carnegie Mellon Links](#) - [Library Search](#) - [All 14 versions](#)

[\[BOOK\] Software architecture: perspectives on an emerging discipline](#)

[M Shaw...](#) - 1996 - [mendeley.com](#)

Abstract Good **software** developers often adopt one or several architectural patterns as strategies for system organization. But, although they use these patterns purposefully, they often use them informally and nearly unconsciously. This book organizes this substantial ...

[Cited by 4201](#) - [Related articles](#) - [Cached](#) - [Carnegie Mellon Links](#) - [Library Search](#) - [All 10 versions](#)

[\[BOOK\] Software reuse: architecture, process and organization for business success](#)

[I Jacobson, M Griss...](#) - 1997 - [mendeley.com](#)

Abstract Systematic **software** reuse is the most effective way to significantly improve **software** development. Many organizations adopt object technology expecting significant reuse. Without an explicit reuse process, they will not succeed. Companies succeeding with ...

[Cited by 1266](#) - [Related articles](#) - [Cached](#) - [Carnegie Mellon Links](#) - [Library Search](#) - [All 7 versions](#)

[Software architecture: The next step](#)

[J Bosch](#) - [Software architecture, 2004](#) - [Springer](#)

This position paper makes the following claims that, in our opinion, are worthwhile to discuss at the workshop. 1) The first phase of **software architecture** research, where the key concepts are components and connectors, has matured the technology to a level where industry ...

[Cited by 193](#) - [Related articles](#) - [BL Direct](#) - [All 6 versions](#)

[\[PDF\] from rug.nl](#)

[\[BOOK\] Pattern-oriented software architecture: Patterns for concurrent and networked objects](#)

[D Schmidt, M Stal, H Rohnert...](#) - 2000 - [mendeley.com](#)

Abstract Designing application and middleware **software** to run in concurrent and networked environments is a significant challenge to **software** developers. The patterns catalogued in this second volume of Pattern-Oriented **Software** Architectures (POSA) form the basis of a ...

[Cited by 1309](#) - [Related articles](#) - [Cached](#) - [Carnegie Mellon Links](#) - [Library Search](#) - [All 10 versions](#)

[\[BOOK\] Pattern-oriented software architecture: On patterns and pattern languages](#)

[F Buschmann, K Henney...](#) - 2007 - [books.google.com](#)

Software patterns have revolutionized the way developers think about how **software** is designed, built, and documented, and this unique book offers an in-depth look of what patterns are, what they are not, and how to use them successfully The only book to attempt ...

[Cited by 4256](#) - [Related articles](#) - [Carnegie Mellon Links](#) - [Library Search](#) - [All 14 versions](#)

[\[PDF\] from ispras.ru](#)



Today's Panelists

Linda Northrop, SEI

Doug Schmidt, Vanderbilt & SEI

Ian Gorton, PNNL

Robert Schwanke, Siemens

Jeromy Carriere, X.com

