Improving Solution Architecting Practices

Poort, E.R.

2012

Bibliography

AACE. Risk analysis and contingency determination using range estimating, 2000. AACE International Recommended Practice No. 41R-08. 128

G. Abowd, L. Bass, P. Clements, R. Kazman, L. Northrop, and A. Zaremski. Recommended best industrial practice for software architecture evaluation. Technical Report CMU/SEI-96-TR-025, SEI, 1997. 149

P. Abrahamsson, M. A. Babar, and P. Kruchten. Agility and architecture: Can they coexist? *IEEE Software*, 27:16–22, 2010. 142

Agile Alliance. Manifesto for agile software development, 2001. URL `http://agilemanifesto.org`. 52, 141

M. Ali Babar, T. Dingsøyr, P. Lago, and H. van Vliet, editors. *Software Architecture Knowledge Management: Theory and Practice*. Springer, Aug. 2009. 10, 101, 117

D. Baccarini. The logical framework method for defining project success. *Project Management Journal*, 30:25–32, 1999. 61

S. F. Bacon. *Religious Meditations*. 1597. 107

P. Bannerman. Risk and risk management in software projects: A reassessment. *Journal of Systems and Software*, 81:2118–2133, 2008. 139

M. R. Barbacci, R. Ellison, A. J. Lattanze, J. A. Stafford, C. B. Weinstock, and W. G. Wood. Quality attribute workshops (QAWs), third edition. Technical Report CMY/SEI-2003-TR-016, SEI, 2003. 42, 48, 148

L. Bass and R. Kazman. Architecture based development. Technical Report CMU/SEI-2000-TR-007, SEI, April 1999. 80, 93

L. Bass, P. Clements, and R. Kazman. *Software Architecture in Practice, 2nd. ed.* Addison Wesley, 2003. 5, 16, 26, 35, 38, 42, 43, 55, 58, 93, 118, 124, 139, 149

L. Bass, R. Nord, W. Wood, and D. Zubrow. Risk themes discovered through architecture evaluations. In *6th Working IFIP/IEEE Conference on Software Architecture (WICSA)*, pages 1–10. IEEE Computer Society, 2007. 139

R. Berntsson Svensson. *Managing Quality Requirements in Software Product Development*. PhD thesis, Department of Computer Science, Lund University, 2009. 5, 43, 55, 61, 62

S. Biffl, A. Aybuke, B. Boehm, H. Erdogmus, and P. Gruenbacher, editors. *Value-Based Software Engineering*, chapter Valuation of Software Initiatives Under Uncertainty: Concepts, Issues, and Techniques, pages 39–66. Springer, 2006. 135, 140

B. Boehm. *Software Engineering Economics*. Prentice Hall, 1981. 52, 118, 131

B. Boehm. A spiral model of software development and enhancement. *IEEE Computer*, 21(5):61–72, 1988. 139

B. Boehm and P. Bose. A collaborative spiral software process model based on Theory W. In *Third International Conference on the Software Process, 'Applying the Software Process'*, 1994. 16, 38

B. Boehm and H. In. Identifying quality-requirement conflicts. *IEEE Software*, pages 25–35, March 1996. 5, 16, 22, 35, 51

B. Boehm and R. Turner. *Balancing Agility and Discipline*. Addison Wesley, 2004. 139

B. Boehm, P. Bose, E. Horowitz, and M. J. Lee. Software requirements negotiation and renegotiation aids: a Theory W based spiral approach. In *17th international conference on Software engineering (ICSE)*, pages 243 – 253, 1995. 51

B. W. Boehm. Software risk management: Principles and practices. *IEEE Software*, 8: 32–41, 1991. 127, 139

J. Bosch. *Design and Use of Software Architectures*. Addison Wesley, 2000. 16, 35

J. Bosch. Software architecture: The next step. In *Software Architecture, First European Workshop (EWSA)*, volume 3047 of *LNCS*, pages 194–199. Springer, May 2004. 81, 90

F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, and M. Stal. *Pattern-Oriented Software Architecture, Volume 1: A System of Patterns*. John Wiley and Sons Ltd, 1996. 16

CCPSO. Common criteria for information technology security evaluation, August 1999. URL `http://www.commoncriteriaportal.org/cc`. Produced by the Common Criteria Project Sponsoring Organizations, version 2.1. 22, 23

L. Chung, B. Nixon, E. S. Yu, and J. Mylopoulos. *Non-Functional Requirements in Software Engineering*. Kluwer Academic, 1999. 5, 16, 35, 38, 42, 52, 55, 58

P. Clements. Exploring enterprise, system of systems, and system and software architectures, January 2009. URL http://www.sei.cmu.edu/library/assets/20090122webinar.pdf. 176

P. Clements and L. Northrop. *Software Product Lines*. Addison Wesley, 2002. 150

P. Clements and M. Shaw. The golden age of software architecture: A comprehensive survey. Technical report, Institute for Software Research International School of Computer Science, Carnegie Mellon University, 2006. CMU-ISRI-06-101. 101

P. Clements, R. Kazman, and M. Klein. *Evaluating Software Architectures*. Addison Wesley, 2002. 42, 78, 81

P. Clements, R. Kazman, M. Klein, D. Devesh, S. Reddy, and P. Verma. The duties, skills, and knowledge of software architects. In *6th Working IFIP/IEEE Conference on Software Architecture (WICSA)*. IEEE Computer Society, 2007. 135, 140, 151

V. Clerc. *Architectural Knowledge Management in Global Software Development*. PhD thesis, VU University Amsterdam, 2011. 117

V. Clerc, P. Lago, and H. van Vliet. The architect's mindset. In *Quality of software architectures 3rd international conference on Software architectures, components, and applications*, QoSA'07, pages 231–249, Berlin, Heidelberg, 2007. Springer-Verlag. 166, 176

CMMI Product Team. CMMI for development, version 1.3. Technical report, SEI, 2010. CMU/SEI-2010-TR-033. 6, 22, 79, 83, 85, 91, 97

H. R. Costa, M. de O. Barros, and G. H. Travassos. Evaluating software project portfolio risks. *Journal of Systems and Software*, 80(1):16–31, 2007. 139

L. J. Cronbach. Coefficient alpha and the internal structure of tests. *Psychometrika*, 16 (3):297–334, 1951. 61, 109

D. Dalcher and A. Genus. Avoiding IS/IT implementation failure. *Technology Analysis and Strategic Management*, 15(4):403–407, December 2003. 2

A. M. Davis. *Just Enough Requirements Management*. Dorset House, 2005. 52

R. Davison, M. G. Martinsons, and N. Kock. Principles of canonical action research. *Information Systems Journal*, 14(1):65–86, 2004. 8

R. de Boer, R. Farenhorst, P. Lago, H. van Vliet, V. Clerc, and A. Jansen. Architectural knowledge: Getting to the core. In S. Overhage, C. Szyperski, R. Reussner, and J. Stafford, editors, *Software Architectures, Components, and Applications*, volume 4880 of *Lecture Notes in Computer Science*, pages 197–214. Springer Berlin / Heidelberg, 2007. 125

H. de Bruin and H. van Vliet. Top-down composition of software architectures. In *9th Annual IEEE International Conference on the Engineering of Computer-Based Systems (ECBS)*, April 2002. 16, 22

B. de Winter. Gratis reizen kan. *PC-Active*, (224):16–18, 2011. 1

D. Dvir, T. Raz, and A. J. Shenhar. An empirical analysis of the relationship between project planning and project success. *International Journal of Project Management*, 21:89–95, 2003. 61

K. E. Emam and A. G. Koru. A replicated survey of IT software project failures. *IEEE Software*, September/October 2008:84–89, 2008. 116

ETSI. European digital cellular telecommunications system (phase 1);technical realization of the short message service point-to-point (GSM 03.40), 1995. URL `http://www.etsi.org`. 75

European Commission. Directive 2004/18/EC of the european parliament and of the council of 31 march 2004 on the coordination of procedures for the award of public works contracts, public supply contracts and public service contracts. Official Journal of the European Union, 2004. 39, 42, 49

Expertgroep C2000. Tussenrapportage expertgroep C2000. Ministerie van Binnenlandse Zaken en Koninkrijksrelaties, 2009. 1

G. Fairbanks. Just enough architecture: The risk-driven model. *Crosstalk*, Nov/Dec 2010. 57, 127, 138, 140, 151

R. Farenhorst and R. de Boer. *Architectural Knowledge Management – Supporting Architects and Auditors*. PhD thesis, VU University Amsterdam, 2009. 3, 101, 117, 121

R. Farenhorst and H. van Vliet. Experiences with a wiki to support architectural knowledge sharing. 2008. 150

M. S. Feather, S. L. Cornford, K. A. Hicks, J. D. Kiper, and T. Menzies. A broad, quantitative model for making early requirements decisions. *IEEE Software*, 25: 49–56, March 2008. 140

M. Fowler. Who needs an architect? *IEEE Software*, 20(5):11–13, 2003. 123, 124, 130, 151

J. Frederick P. Brooks. *The Mythical Man-Month: Essays on Software Engineering, 20th Anniversary Edition*. Addison-Wesley, 1995. 110, 116

S. Fricker and M. Glinz. Comparison of requirements hand-off, analysis, and negotiation: Case study. In *2010 18th IEEE International Requirements Engineering Conference*, RE '10, pages 167–176, Washington, DC, USA, 2010. IEEE Computer Society. 52

S. Fricker, T. Gorschek, C. Byman, and A. Schmidle. Handshaking with implementation proposals: Negotiating requirements understanding. *IEEE Software*, 27(2): 72–80, 2010. 51

E. Gamma, R. Helm, R. E. Johnson, and J. Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley, Reading, MA, 1995. 16, 149

D. A. Garvin. What does "product quality" really mean? *MIT Sloan Management Review*, 26:25–43, Fall 1984. 43

D. L. Gibson, D. Goldenson, and K. Kost. Performance results of CMMI-based process improvement. Technical Report CMU/SEI-2006-TR-004, SEI, August 2006. 83

T. Gilb. *Principles of Software Engineering Management*. Addison Wesley, 1988. 5, 21, 139

T. Gilb. *Competitive Engineering: A Handbook For Systems Engineering, Requirements Engineering, and Software Engineering Using Planguage*. Butterworth-Heinemann, Newton, MA, USA, 2005. 5, 21, 24, 42, 53, 82, 141

M. Glinz. On non-functional requirements. In *15th IEEE International Requirements Engineering Conference RE 2007*, pages 21–26. IEEE, 2007. 5

M. Glinz. A risk-based, value-oriented approach to quality requirements. *IEEE Software*, 25:34–41, 2008. 52, 127

R. B. Grady. An economic release decision model: Insights into software project management. In *Applications of Software Measurement Conference, Orange Park, Software Quality Engineering*, pages 227–239, 1999. 62

R. Gram and B. Keulen. Quick scan tunnelprojecten. Ministerie van Verkeer en Waterstaat - Rijkswaterstaat, 2010. Referentie 10 008 R 012. 1, 42

D. Gross and E. Yu. From non-functional requirements to design through patterns. *Requirements Engineering*, 6(1):18–36, 2001. 16

P. Gruenbacher, A. Egyed, and N. Medvidovic. Reconciling software requirements and architecture: the CBSP approach. In *5th IEEE International Symposium on Requirements Engineering*, IEEE CS, August 2001. 16, 35

A. Herrmann and M. Daneva. Requirements prioritization based on benefit and cost prediction: An agenda for future research. In *2008 16th IEEE International Requirements Engineering Conference*, pages 125–134, Washington, DC, USA, 2008. IEEE Computer Society. 142

A. Herrmann and B. Paech. MOQARE: misuse-oriented quality requirements engineering. *Requir. Eng.*, 13:73–86, January 2008. doi: 10.1007/s00766-007-0058-9. 142

A. Herrmann, A. Morali, and S. Etalle. RiskREP: Risk-based security requirements elicitation and prioritization (extended version). Technical Report TR-CTIT-10-28, Centre for Telematics and Information Technology University of Twente, Enschede, August 2010. URL `http://eprints.eemcs.utwente.nl/18342/`. 141

C. Hofmeister, P. Kruchten, R. L. Nord, J. H. Obbink, A. Ran, and P. America. A general model of software architecture design derived from five industrial approaches. *Journal of Systems and Software*, 80(1):106–126, 2007. 80, 94, 132, 133

IEC 61508. Functional safety of electrical/electronic/ programmable electronic safety-related systems, 1999. IEC 61508. 22

ISO 42010. Systems and software engineering - architecture description, 2011. ISO 42010:2011. 4, 96, 123, 124, 125, 149

ISO/IEC 25000. Software engineering – software product quality requirements and evaluation (SQuaRE) – guide to SQuaRE, 2005. ISO 25000:2005. 21, 42

A. Ivanović and P. America. Information needed for architecture decision making. In *2010 ICSE Workshop on Product Line Approaches in Software Engineering*, PLEASE '10, pages 54–57, New York, NY, USA, 2010a. ACM. 122

A. Ivanović and P. America. Strategy-focused architecture decision making. In P. van de Laar and T. Punter, editors, *Views on Evolvability of Embedded Systems*, pages 245–260. Springer, 2010b. 140

M. Jackson. *Problem frames: analyzing and structuring software development problems*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2001. 35

I. Jacobson, P. W. Ng, and I. Spence. Enough process – let's do practices. *Journal of Object Technology*, 6(6):41–66, July-August 2007. 152, 175, 176

A. Jansen and J. Bosch. Software architecture as a set of architectural design decisions. In *5th Working IEEE/IFIP Conference on Software Architecture*, pages 109–120, Washington, DC, USA, 2005. IEEE Computer Society. 3, 121, 122, 123, 125, 149, 151

E. Johansson, A. Wesslén, L. Bratthall, and M. Höst. The importance of quality requirements in software platform development – a survey. In *HICSS '01: 34th Annual Hawaii International Conference on System Sciences*, volume 9, page 9057, Washington, DC, USA, 2001. IEEE Computer Society. 67

C. Jones. *Software Assessments, Benchmarks, and Best Practices*. Addison-Wesley, 2000. 110, 116

D. Kahneman. *Thinking, Fast and Slow*. Farrar, Straus and Giroux, 2011. 141, 144

D. Kahneman and A. Tversky. Prospect theory: An analysis of decision and risk. *Econometrica*, 47:263–291, 1979. 141

D. Karolak. *Software Engineering Risk Management*. IEEE Computer Society Press,U.S., 1995. 52

R. Kazman, J. Asundi, and M. Klein. Making architecture design decisions: An economic approach. Technical Report CMU/SEI-2002-TR-035, SEI, 2002. 28, 42, 52, 78, 122, 140, 141, 143, 149

R. Kazman, L. Bass, and M. Klein. The essential components of software architecture design and analysis. *Journal of Systems and Software*, 79(8):1207–1216, 2006. 151, 152, 175

A. Klusener, R. Lämmel, and C. Verhoef. Architectural modifications to deployed software. *Science of Computer Programming*, 54:143–211, 2005. 130, 143

N. Kock. *Action Research: Its Nature and Relationship to Human-Computer Interaction*. The Interaction-Design.org Foundation, Aarhus, Denmark, 2011. URL `http://www.interaction-design.org/encyclopedia/action_research.html`. 8

P. Kruchten. The 4+1 view model of architecture. *IEEE Software*, 12(6):45–50, 1995. 149

P. Kruchten. *The Rational Unified Process: An Introduction*. Addison-Wesley, Boston, 1998. 3, 24, 51, 121, 123

P. Lago and H. van Vliet. Building up and reasoning about architectural knowledge. In *Second International Conference on the Quality of Software Architectures (QoSA)*, pages 43–58, 2006. 101

A. Lamsweerde. Conceptual modeling: Foundations and applications. chapter Reasoning About Alternative Requirements Options, pages 380–397. Springer-Verlag, Berlin, Heidelberg, 2009. 52

P. A. Laplante and C. J. Neill. Uncertainty: A meta-property of software. In *29th Annual IEEE/NASA Software Engineering Workshop*, pages 228–233, 2005. 52

B. Lawrence, K. Wiegers, and C. Ebert. The top risks of requirements engineering. *IEEE Software*, 18(6):62–63, 2001. 5

H. Lee. Applying fuzzy set theory to evaluate the rate of aggregate risk in software development. *Fuzzy Sets and Systems*, 80(3):261–271, 1996. 139

H. K. N. Leung. Quality metrics for intranet applications. *Information and Management*, 38(3):137–152, 2001. 67

K. Linberg. Software developer perceptions about software project failure: a case study. *The Journal of Systems and Software*, 49:177–92, 1999. 61

D. Mairiza, D. Zowghi, and N. Nurmuliani. An investigation into the notion of non-functional requirements. In *2010 ACM Symposium on Applied Computing (SAC), Sierre, Switzerland, March 22-26, 2010*, pages 311–317, 2010. 5, 58

R. Malan and D. Bredemeyer. Less is more with minimalist architecture. *IT Pro*, pages 46–48, September-October 2002. 138, 151

M. L. Markus. Power, politics, and M.I.S. implementation. *Commun. ACM*, 26(6): 430–444, 1983. 118

T. McCabe. A complexity measure. *IEEE Transactions on Software Engineering*, 2: 308–320, 1976. 68

S. McConnell. *Rapid Development*. Microsoft Press, 1996. 118

S. C. Mcconnell. *Software Project Survival Guide (Pro – Best Practices)*. Microsoft Press, Nov. 1997. 52

Money Magazine. Best jobs in america 2010, top 100. CNN on-line, November 2010. URL `http://money.cnn.com/magazines/moneymag/bestjobs/2010/snapshots/1.html`. Rank 1: Software Architect. 136

J. Mylopoulos. Goal-oriented requirements engineering, part ii. In *RE '06: 14th IEEE International Requirements Engineering Conference*, Washington, DC, USA, 2006. IEEE Computer Society. 5, 38, 55

J. Mylopoulos, L. Chung, and B. Nixon. Representing and using nonfunctional requirements: A process-oriented approach. *IEEE Trans. Softw. Eng.*, 18(6):483–497, 1992. 5, 16

J. Noppen. *Imperfect Information in Software Design Processes*. PhD thesis, University of Twente, 2007. 52

H. Obbink, P. Kruchten, W. Kozaczynski, R. Hilliard, A. Ran, H. Postema, L. Dominick, R. Kazman, W. Tracz, and E. Kahane. Report on software architecture review and assessment (SARA). Technical report, SARA Working group, 2002. URL `http://kruchten.com/philippe/architecture/SARAv1.pdf`. retrieved 11 January 2012. 81, 132

B. Paech and D. Kerkow. Non-functional requirements engineering – quality is essential. In *10th Anniversary International Workshop on Requirements Engineering: Foundation for Software Quality*, 2004. 62

B. Paech, R. Heinrich, G. Zorn-Pauli, A. Jung, and S. Tadjiky. Answering a request for proposal – challenges and proposed solutions. In *Proceedings 18th International Working Conference Requirements Engineering: Foundation for Software Quality (REFSQ2012)*. 51

B. Paech, A. Detroit, D. Kerkow, and A. von Knethen. Functional requirements, non-functional requirements, and architecture should not be separated – a position paper. REFSQ, Essen, Germany, September 2002. 5, 38, 52, 55

D. E. Perry and A. L. Wolf. Foundations for the study of software architecture. *SIGSOFT Softw. Eng. Notes*, 17:40–52, October 1992. 3, 121

I. Pinkster, B. van de Burgt, D. Janssen, and E. van Veenendaal. *Successful Test Management – An Integral Approach*. Springer, 2004. 42

J. Pinto and D. Slevin. Project success: definitions and measurement techniques. *Project Management Journal*, 19:67–72, 1988. 61

G. Pitette. Progressive acquisition and the RUP: Comparing and combining iterative processes for acquisition and software development. 2001. 51

E. R. Poort and P. H. N. de With. Modelling the relationship between quality attributes and architecture of software intensive systems. In *PROGRESS Symposium 2003*, PROGRESS, October 2003. 22

E. R. Poort and H. van Vliet. Architecting as a risk- and cost management discipline. In *Proceedings 9th Working IEEE/IFIP Conference on Software Architecture (WICSA)*, pages 2–11. IEEE Computer Society, 2011. 129

E. R. Poort, H. Postema, A. Key, and P. H. de With. The influence of CMMI on establishing an architecting process. In *Third International Conference on the Quality of Software-Architectures (QoSA)*, 2007. 79, 91

J. D. Procaccino. What do software practitioners really think about project success: an exploratory study. *Journal of Systems and Software*, 78:194–203, 2005. 61

Z. Racheva, M. Daneva, and A. Herrmann. A conceptual model of client-driven agile requirements prioritization: results of a case study. In *2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*, ESEM '10, pages 39:1–39:4, New York, NY, USA, 2010. ACM. 142

RACV Insurance Pty Ltd v. Unisys Australia Ltd. RACV Insurance Pty Ltd v. Unisys Australia Ltd, 2001. VSC 300. 41

B. Regnell, R. Berntsson Svensson, and T. Olsson. Supporting roadmapping of quality requirements. *IEEE Software*, 25:42–47, March 2008. 43, 52, 141

S. Robertson and J. Robertson. *Mastering the Requirements Process (2nd Edition)*. Addison-Wesley Professional, 2006. 35

SHARK. 4th workshop on sharing and reusing architectural knowledge. IEEE Computer Society, 2009. 117, 122

M. Shaw. Toward higher-level abstractions for software systems. *Data & Knowledge Engineering*, 5(2):119 – 128, 1990. 3, 121, 123

H. Simon. *The Sciences of the Artificial*. MIT Press, 1969. 140

O. P. N. Slyngstad, R. Conradi, M. A. Babar, V. Clerc, and H. van Vliet. Risks and risk management in software architecture evolution: An industrial survey. In *15th Asia-Pacific Software Engineering Conference (APSEC)*, pages 101–108, 2008. 139, 143

Standish Group. *Chaos Report*. 1994. 118

S. Supakkul, T. Hill, L. Chung, T. T. Tun, and J. C. S. do Prado Leite. An NFR pattern approach to dealing with NFRs. In *2010 18th IEEE International Requirements Engineering Conference*, RE '10, pages 179–188, Washington, DC, USA, 2010. IEEE Computer Society. 52

A. Sutcliffe. The socio-economics of software architecture. *Automated Software Engineering*, 15:343–363, 2008. 176

A. Tang and J. Han. Architecture rationalization: A methodology for architecture verifiability, traceability and completeness. In *12th Annual IEEE International Conference on the Engineering of Computer-Based Systems (ECBS)*, pages 135–144, 2005. 140

The Open Group. The open group certified architect (Open CA) program – conformance requirements (multi-level). 176

The Open Group. The open group architecture framework (TOGAF), 2009. URL `http://www.togaf.info`. 4

J. Tyree and A. Akerman. Architecture decisions: Demystifying architecture. *IEEE Software*, 22(2):19–27, 2005. 3, 81, 90, 101, 121, 122, 123, 149, 151

US Department of Commerce. Enterprise architecture capability maturity model. online, 2007. URL `http://ocio.os.doc.gov/ITPolicyandPrograms/Enterprise_Architecture/PROD01_004934`. retrieved 8 Jan 2012. 99, 176

US Government. Code of federal regulations, title 48: Federal acquisition regulation. National Archives and Records Administration, 2005. 38, 49

J. S. van der Ven, A. Jansen, P. Avgeriou, and D. K. Hammer. Using architectural decisions. In *Second International Conference on the Quality of Software Architectures (QoSA 2006)*, LNCS, 2006. 81

A. van Lamsweerde. *Requirements Engineering – From System Goals to UML Models to Software Specifications*. Wiley, 2009. 35

J. C. Westland. The cost of errors in software development: evidence from industry. *Journal of Systems and Software*, 62(1):1–9, May 2002. 62

K. E. Wiegers. *Software Requirements, Second Edition (Pro-Best Practices)*. Microsoft Press, 2 sub edition, 2003. 35

O. Zimmermann, T. Gschwind, J. Küster, F. Leymann, and N. Schuster. Reusable architectural decision models for enterprise application development. In *Third International Conference on the Quality of Software Architectures (QoSA)*, number 4880/2008 in LNCS, pages 157–166. Springer, 2007. 122, 150