Toward a Theory of Debiasing Software Development

Paul Ralph[1]

[1] Lancaster University, Lancaster, UK
paul@paulralph.name

Abstract. Despite increasingly sophisticated programming languages, software developer training, testing tools, integrated development environments and project management techniques, software project failure, abandonment and overrun rates remain high. One way to address this is to focus on common systematic errors made by software project participants. In many cases, such errors are manifestations of cognitive biases. Consequently this paper proposes a theory of the role of cognitive biases in software development project success. The proposed theory posits that such errors are mutual properties of people and tasks; they may therefore be avoided by modifying the person-task system using specific sociotechnical interventions. The theory is illustrated using the case of planning poker, a task estimation technique designed to overcome anchoring bias.

Keywords: Design Science, Software Engineering, Theory Development, Cognitive Bias, Debiasing, Heuristics, Illusions

1 Introduction

Software development and maintenance constitute substantial economic activity – the 500 largest software companies employed 3,562,407 and accrued revenues of US$491 billion in 2010 [1] and total global information technology spending was US$3.2 trillion in 2009 [2]. The magnitude of this spending makes estimates of failure rates far more alarming. Estimates of completely abandoned software development projects vary between 10% and 44% while between 16% and 52.7% experience "major truncation or simplification … prior to full implementation" [3, p. 17, 19]. More recently, the success rate has been estimated at 32% with 44% "challenged" [4]. A meta-analysis of estimation accuracy studies found an average effort overrun of 30-40% [5]. If civil-engineering projects had similar success rates, abandoned buildings and half-built bridges would litter our cities.

This raises many questions. What causes software project success, abandonment and failure? How should success and failure be measured? How can software project outcomes be improved? Which causes of failure are reparable and by whom? For the purposes of this paper, design (verb) refers to the act of creating a specification of an object intended to accomplish goals in a particular environment using a set of primitive components, subject to constraints where a specification may be a plan or the object itself (adapted from [6]). A software design project then "is a temporal trajectory of a work system toward one or more goals" [6, p. 116] where at least one goal entails creating software.

Van de Ven [7] argues that completely understanding such complex phenomena requires multifarious theoretical perspectives. This paper proposes a cognitive psychology perspective – specifically, using the notion of cognitive biases to understanding *avoidable* errors and how to address them in software design projects. In summary, the purpose of the paper is as follows.

Purpose: to propose a theoretical framework of errors in software development projects and their relationship to cognitive biases.

To this end, the paper is organized as follows. Section Two provides a brief review of existing research on antecedents of project success and failure. This is succeeded by an introduction to the basis for the proposed theory – cognitive biases (§3). The proposed theory is presented in Section Four and evaluated in Section Five.

2 Existing Perspectives on Critical Success Factors in Software Development

Both scientific and popular literature are rife with examples of high profile design failures. Connected Earth's new million-pound website was incompatible with nearly all modern web browsers [8]. The British supermarket Sainsbury's spent US$526 million on a supply chain management system that failed so dramatically that the company had to hire 3000 people to move stock manually [9]. A recent study of a computerized physician order entry (CPOE) system intended to reduce medication errors found that the system caused patients to get the wrong medicine in 22 different ways [10]. When Windows Vista was first released, consumers were so unhappy that "more than one in every three new PCs [was] downgraded from Windows Vista to the older Windows XP, either at the factory or by the buyer" [11, p. 1].

Many studies have listed diverse success, abandonment and risk factors for software projects. Table 1 provides three exemplars. While identifying these factors is useful to predict success and identify high-risk projects, many of the factors are beyond project participants' control. For example, changing requirements and unclear objectives may be unavoidable. The most striking example is perhaps "executive management support". Noting low executive support is important for gauging risk but it is not necessarily actionable. More generally, identifying factors associated with success and failure is not equivalent to demonstrating causal relationships.

Table 1. Success, Abandonment and Risk Factors for Software Projects

#	Risk Factors [9]	Success Factors [4, 51, 52]	Abandonment Factors [3]
1	“Unrealistic or unarticulated project goals”	“User involvement”	“Unrealistic project goals and objectives”
2	“Inaccurate estimates of needed resources”	“Executive management support”	“Inappropriate project-team composition”
3	“Badly defined system requirements”	“Clear business objectives”	“Project management and control problems”
4	“Poor reporting of the project's status”	“Optimizing scope”	“Inadequate technical know-how”
5	“Unmanaged risks”	“Agile process”	“Problematic technology base/infrastructure”
6	“Poor communication among customers, developers, and users”	“Project manager expertise”	“Changing requirements”
7	“Use of immature technology”	“Financial management”	“Lack of executive support and commitment”
8	“Inability to handle the project's complexity”	“Skilled resources”	“Insufficient user commitment and involvement”
9	“Sloppy development practices”	“Formal methodology”	“Cost overruns and schedule delays”
10	“Poor project management”	“Standard tools and infrastructure”	
11	“Stakeholder politics”		
12	“Commercial pressures”		

The non-actionable nature of many success factors and limitations in establishing causality motivate a new theoretical approach. At least five criteria for this approach are evident. First, the theory should clearly identify one or more antecedents of design project success or failure. Second, the theory should be testable. Although this may seem obvious, rationalism has a strong and destructive tradition in design research [12]. Third, the theory should rest on a sound theoretical or empirical basis, derived from existing theory or observation. Fourth, the theory should produce actionable recommendations (e.g., 'model requirements using scenarios,' is more actionable than 'ensure executive support'). Fifth, the theory should possess reasonable face validity. The following section explores a potential theoretical basis – cognitive biases.

3 Cognitive Biases

"Cognitive biases are cognitions or mental behaviors that prejudice decision quality in a significant number of decisions for a significant number of people" [13, p. 59]. For example, *default bias* [14] is the tendency to choose preselected options over superior, unselected options. Although dozens if not hundreds of cognitive biases have been identified, most research considers only one or a few examples [15]. Consequently, some biases seem to overlap. For example, the valence effect (the tendency to underestimate the probability of negative events and the inverse) is very similar to unrealistic optimism bias (the tendency to overestimate the likelihood of success). Given the proliferation of similar and interconnected biases, it may be useful to consider complexes of biases ("biasplexes") rather than individual biases.

To date, I have identified twelve biasplexes with potential ramifications for software development projects (Table 2). In the interest of space, however, this paper focuses on two biasplexes (Table 3) – one well-known and one new. Biasplexes may have both positive and negative consequences. For example, *unrealistic optimism* may encourage people to produce unrealistic project schedules. This is the *planning fallacy* [16], "the tendency to hold a confident belief that one's own project will proceed as planned, even while knowing that the vast majority of similar projects have run late" [17, p. 366]. The planning fallacy may lead to poor decisions and project abandonment [3]. However, *unrealistic optimism* may also help people to cope with stress [18].

Furthermore, biases (and biasplexes) may have at least four causes. First, heuristics "reduce the complex tasks of assessing probabilities and predicting values to simpler judgmental operations" [19, p. 1124]. For example, the anchoring and adjustment heuristic [cf. 20] estimates quantities by increasing or decreasing an existing estimate. This leads to anchoring bias, systematic insufficient adjustment from anchors among decision makers. Second, cognitive illusions – incorrect, unconscious inferences [cf., 21] – are not limited to the optical illusions that populate undergraduate psychology textbooks. The illusion of control [22] for instance, may lead to excessive optimism, i.e., optimism bias. Third, group processes may cause or exacerbate individual biases. For example, groupthink [23] may lead to collective excessive optimism by inhibiting critical thinking. Fourth, other individual psychological phenomena, here called principles, may further create or exacerbate biases. For example, the Pollyanna Principle [24] – the tendency to be overly positive in perception, memory and judgment – may reinforce optimism bias.

Table 2. Twelve Biasplexes

Biasplex	Definition	Example Biases
Affective Forecasting	a collection of tendencies to overestimate the emotional impact of events	impact bias, hot-cold empathy gap
Causality Errors	a collection of tendencies to infer causal relationships without sufficient evidence	actor-observer bias, egocentric bias
Fixation	a collection of tendencies to disproportionately focus on one aspect of an event, object or situation, especially self-imposed or imaginary obstacles	design fixation, negativity bias
Framing Effects	"the tendency to give different responses to problems that have surface dissimilarities but that are really formally identical" [48, p. 88]	loss aversion, irrational escalation
Memory Errors	a collection of tendencies to remember ideas and events inaccurately	hindsight bias, self-serving bias
Miserly Information Processing	the tendency to avoid deep or complex information processing [48]	confirmation bias, belief bias
Overconfidence	the tendency to overestimate one's own skill, accuracy and control over one's self and environment	overconfidence effect, restraint bias
Perception Errors	a collection of tendencies to perceive situations inaccurately	selective perception, halo effect
Probabilistic Reasoning Errors	a collection of tendencies to err in numerical reasoning about chance and uncertainty.	base-rate neglect, subadditivity effect
Inertia	a collection of tendencies that increase disproportional preference for and defense of the status quo	see Table 3
Resistance to Self-criticism	a collection of tendencies to avoid recognizing one's mistakes or downplay their significance	outcome bias, myside bias
Unrealistic Optimism	the tendency to to make overly positive estimates and attributions	see Table 3

Table 3. Unrealistic Optimism and Inertia Biasplexes Expanded

Biasplex	Bias	Definition
Unrealistic Optimism	Wishful Thinking	the tendency to underestimate the likelihood of a negative outcome and vice versa [53]
	Valence Effect	the tendency to give undue weight to the degree to which an outcome is considered as positive or negative when estimating the probability of its occurrence [50]
	Optimism bias	the tendency for predicted outcomes to be more positive than actual outcomes [54]
	Normalcy Bias	systematically "underestimating the probability or extent of expected disruption" during a disaster [62, p. 273]
Inertia	Endowment Effect	"The tendency to demand much more to give up an object than one is willing to pay to acquire it" [55, p. 252]
	Belief Perseverance	"the tendency to maintain a belief even after the information that originally gave rise to it has been refuted or otherwise shown to be inaccurate" [56, p. 112]
	Anchoring Bias	"The tendency, in forming perceptions or making quantitative judgments of some entity under conditions of uncertainty, to give excessive weight to the initial starting value (or anchor), based on the first received information or one's initial judgment, and not to modify this anchor sufficiently in light of later information" [56, p. 51]
	Bandwagon Effect	"the tendency for large numbers of individuals, in social and sometimes political situations, to align themselves or their stated opinions with the majority opinion as they perceive it" (VandenBos 2007, p. 101)
	Semmelweis Reflex	unthinking rejection of new information that contradicts established beliefs or paradigms [61]
	Default Bias	the tendency to chose an pre-selected option regardless of its superiority or inferiority to other options [57]
	Mere Exposure Effect	"the ... increased liking for a stimulus that follows repeated, unreinforced exposure to that stimulus" [60, p. 231]
	Validity Effect	"the validity effect occurs when the mere repetition of information affects the perceived truthfulness of that information … the validity effect occurs similarly for statements of fact that are true and false in origin, as well as political or opinion statements" [58, p. 211]
	System Justification	the tendency to defend and reinforce the existing social order [59]

Debiasing is the process of inhibiting or removing the effects of biases [25, 26]. As many biases are highly robust (e.g., anchoring bias), debiasing may be extremely challenging [27]. Fischoff [25] suggests that where biases are caused by faulty "but perfectible" judges, they may be addressed by one of four increasingly potent interventions – 1) warn participants about biases in general; 2) identify particular biases in play and their magnitude; 3) provide feedback with personalized implications; 4) extended training. Subsequent empirical research has found that the first three of Fischoff's interventions often fail [cf. 28, 29], leaving the often expensive, sometimes impractical option of extended training. Fischoff however further argued that biases may be addressed by restructuring the person-task system. This strategy has been effective in software project management [e.g., 30] and other domains [e.g., 31].

Of course, using cognitive biases to inform information systems and design research is not new. Numerous studies have investigated cognitive biases in software engineering [e.g., 20, 32, 33], information modeling [e.g., 34], engineering design [e.g., 35], software project management [e.g., 36] and requirements determination [37]. In design science research specifically, [13] used cognitive biases to generate recommendations for decision support system development. However, a theory of the role of cognitive biases in software projects remains elusive.

4 Toward a new Theory

This section uses three scenarios to introduce a cognitive, variance theory of design project success. I begin with the scenario that inspired the theory.

Scenario 1. A seven-person software development team meets to estimate the time required to implement a variety of features for a virtual learning environment (VLE). The project manager describes a high-priority feature, a virtual gradebook. Robert, the team lead and most experienced developer, envisions a spreadsheet-like interface and estimates that it will take two days. Robert does not know that 1) the VLE needs to support both numerical and alphabetical grades, 2) different departments use different mappings, and 3) that the mapping criteria are not well documented. James, a junior developer who is aware of these complexities intuits that the two day estimate is insufficient and estimates that the feature may require three or four days, rather than two. The feature eventually takes three weeks to complete and forces a delay in the initial prototype, endangering the overall project.

In this scenario, James used the anchoring and adjustment heuristic to estimate the feature development time. Anchoring bias caused a specific forecasting error, which negatively affected the success of the project. To avoid exactly this type of error, the software project management framework Scrum [38] includes *planning poker*, a practice where developers choose estimates in secret and show them simultaneously. Some empirical evidence indicates that "planning poker improve[s] the team's estimation performance in most cases" [39, p. 23] by preventing participants from anchoring on an initial estimate. This example motivated the proposed theory (Fig. 1, Table 4).

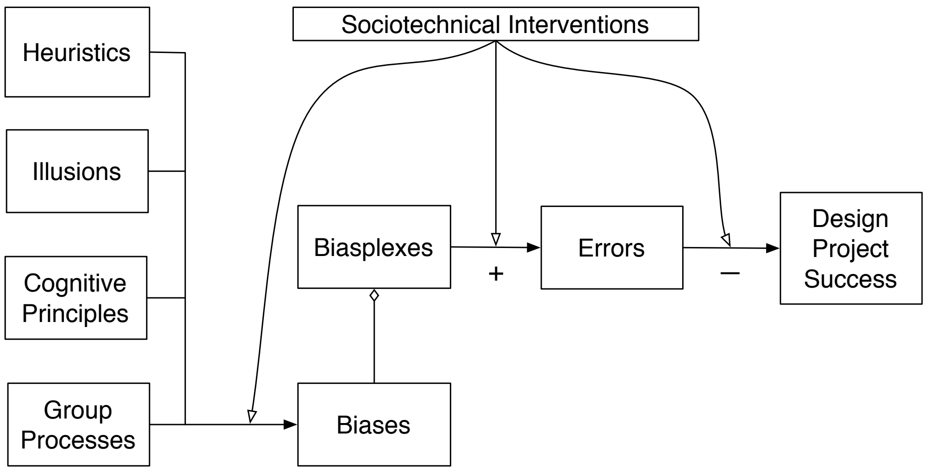

Fig. 1. A Variance Theory of Debiasing Software Project Participants

Note: Filled arrows indicate causation; unfilled arrows indicate moderating effects; plus and minus signs indicate direction of effect; unfilled diamond indicates aggregation.

As described above, bias(plex)es are caused by heuristics, illusions, group processes and cognitive principles. As this relationship is squarely in the cognitive psychology domain, it is included only for completeness and is not the focus of the present research. In contrast, the relationship between biasplexes and errors and between errors and success are primary concerns to understand software development. Moreover, the moderating effect of sociotechnical interventions on these relationships may provide the basis for much design research. Scenarios two and three illustrate the use and usefulness of this framework.

Scenario 2. Following *Scenario 1,* suppose the same team has begun using planning poker to prevent anchoring bias in task estimation. This time, the project manager asks for an estimate for integrating the VLE's gradebook with the university's legacy student information management system. Subconsciously, the developers are inherently optimistic; they assume that they can make minor changes to the legacy system (illusion of control), downplay the bureaucratic complexities of interacting with the legacy system's developers (optimism) and overestimate their own skill and productivity (Pollyanna Principle). These factors mutually reinforce each other, producing *Unrealistic Optimism*. The resulting error, an unrealistically optimistic estimate from each individual during planning poker, produces a cumulatively optimistic estimate, which threatens the project. While planning poker inhibited anchoring and adjustment, participants' shared optimism may still produce unrealistic schedules. While reference class forecasting [40, 41] – an effective technique for avoiding optimistic estimates – is readily available, a review of estimation methods in use was "not [...] able to identify a single study reporting a dominant use of formal estimation methods" [42, p. 228]. This raises the question, how can more effective forecasting methods gain traction in this domain?

Table 4. Theory Constructs Defined

Construct	Definition	Examples
Heuristic	a process or method that simplifies assessing probabilities or predicting values	availability, affect, default
Illusion	an incorrect, unconscious inference prompted by an external stimulus	illusion of control, illusory superiority
Cognitive Principle	a psychological disposition or tendency that interferes with optimal judgment	Pollyanna Principle, cognitive dissonance
Group Process	a shared psychological phenomenon that alters cognition or perception	Groupthink, firehouse effect
Bias	a predictable, systematic *error*	valance effect, default bias
Biasplex	a complex of similar, mutually-reinforcing biases	unrealistic optimism, inertia
Error	a deviation from optimal judgment	planning fallacy, sunk cost fallacy
Sociotechnical Intervention	a method, practice, technique, tool, program or concept that changes the behavior of one or more participants in a software development project	planning poker, checklists
Design Project Success	a multidimensional construct representing the achievement associated with a software design project from various stakeholder perspectives	developer learning, user satisfaction, client profits

Scenario 3. After circumventing some scheduling overruns, the VLE team is asked to deliver a document sharing feature. One of the developers points out that Microsoft

SharePoint may be used for document sharing and that the university already has an unlimited license for it. The project manager asks if SharePoint supports collaborative editing, a desired aspect of document sharing. The developer responds with uncertainty but suggests worrying about advanced features like collaborative editing later while emphasizing how SharePoint is increasingly the default infrastructure choice within the university. The team then builds a SharePoint-based system, which delivers basic functionality in a familiar way but no more. This illustrates the combination of the default heuristic, bandwagon effect and status quo bias (the *Inertia* biasplex) producing a suboptimal decision. Alternatives including Google Apps, which is free and supports collaborative document editing innately, were not considered. This raises the question, what intervention could inhibit Inertia and stimulate better alternative evaluation?

In *Scenario 1*, the development team experienced a threat to success with a causal chain including the anchoring and adjustment heuristic, adjustment bias and a forecasting error. The planning poker intervention inhibits this kind of error and, being part of a popular project management framework, is commonly used in software development. *Scenario 2* is similar in that effective interventions have been identified but different in that model-based forecasting has not caught on. *Scenario 3* is also similar in that a biasplex manifests as a participant error; however, it differs in that no specific intervention has been developed to inhibit Inertia in this context. These scenarios illustrate the following four research streams implied by the proposed theory.

1. What errors are produced by biasplexes in software design projects?
2. How and to what extend do these errors affect Design Project Success?
3. What sociotechnical interventions can ameliorate the effects of biasplexes and resulting errors?
4. How can interventions be propagated through the software development industry?

These questions reveal the interdisciplinarity of the proposed research. Question one is behavioral research, which may be studied using both lab studies and ethnographic approaches. Question two is amenable to ethnography, case studies, surveys and econometric analysis of secondary datasets. Question three appears well suited to the design science approach [43, 44] and possibly action research [45]. Question four is especially challenging and may involve aspects of critical social theory, action research and social network analysis.

5 Conceptual Evaluation

Section Two identified five criteria on which to evaluate the proposed theory – clear antecedents, testability, sound basis, actionable recommendations and face validity. This section evaluates the theory with each criteria.

The proposed theory clearly identifies one class of design project success antecedents – participant errors – and reveals one related causal chain. Isolating errors in this way is motivated by the need to produce actionable recommendations. It does not preclude other causal chains.

The theory is clearly testable. The causes of cognitive biases are already a major subject of empirical study in psychology. Bias-induced errors may be both observed in case studies and demonstrated in lab studies. The effect of errors on specific problems and risks within a project is equally observable, at least through perceptual measures; measuring success however remains challenging.

Cognitive psychology provides ample theoretical basis for the proposed theory. Extensive literature on cognitive biases has developed over the past 40 years,with many quality books and papers on the subject [e.g., 21, 25, 46-48]. As cognitive biases affect everyone, mapping their established role into software development projects should be uncontroversial in principle.

The proposed theory produces two types of recommendations. The more general recommendation is to address common errors in software projects with sociotechnical interventions that modify the person-task system, inhibiting biases so that the errors do not manifest. More specific recommendations comprise the interventions themselves. For example, preliminary analysis suggests that Inertia may be inhibited by designating a Devil's Advocate during decision-making or drawing a high-level design tree showing previous decisions and alternatives. More generally, the theory is useful for stimulating hypotheses about particular errors. For example, one may hypothesize that Unrealistic Optimism will manifest as inflated expectations among clients and intended users, which would increase the chance of disappointment (cf. Expectation Confirmation Theory [49]).

Finally, this paper attempts to demonstrate face validity by using three scenarios (above) to illustrate the causal chain. The idea that software project participants fall prey to cognitive biases resulting in errors seems uncontroversial. As biases are systematic and predictable, it appears credible to hypothesize that the resulting errors will also be credible and predictable. The more difficult questions are the extent of success variance explained by these errors and what interventions may avoid them.

6 Conclusion

This paper's contribution is twofold. First, it introduces the term "biasplex" and identifies 12 such interconnected sets of cognitive biases. Second, it proposes a theory of debiasing software development projects including a chain of antecedents for software design project success and its relationship to sociotechnical interventions for debiasing participants. Conceptual evaluation of the theory appears promising and methods for empirical evaluation are suggested. Future research (in progress) entails hypothesizing common errors caused by each biasplex (Table 2), organized both by participant and by design activity, then identifying evidence of these errors in a series of case studies. This may be followed by developing interventions and testing their effectiveness in reducing the supported errors.

The proposed theory has several limitations. First, it focuses on a single antecedent of Design Project Success (but does not imply that this is the only or primary antecedent). Second, Design Project Success is not well understood, difficult to define and challenging to measure. Third, at this stage no characteristics of successful interventions can be stated other than noting that modest training on biases has been ineffective while extensive, rapid, personalized feedback or changing the person-task system has been effective in previous studies. Finally, as empirical evaluation is still ongoing, the theory is evaluated conceptually. While conceptual evaluation is inherently limited, it is appropriate for this type of greenfield research-in-progress.

References

1. Desmond, J.P.: System Integrators, Outsourcers Gain Again as IT Guards in-House Resources. Software Magazine, King Content Co. (2010), http://www.softwaremag.com/focus-areas/the-software-500-industry/featured-articles/system-integrators-outsourcers-gain-again-as-it-guards-in-house-resources/
2. Gartner: Gartner Says Worldwide IT Spending to Grow 5.3 Percent in 2010. Gartner, Inc. (2010), http://www.gartner.com/it/page.jsp?id=1339013
3. Ewusi-Mensah, K.: Software Development Failures. MIT Press (2003)
4. Standish Group: Chaos Summary 2009. Boston, MA, USA (2009)
5. Moløkken-Østvold, K.: A Review of Surveys on Software Effort Estimation. In Proceedings of International Symposium on Empirical Software Engineering (ISESE 2003), pp. 223–230. IEEE, Rome, Italy (2003)
6. Ralph, P., Wand, Y.: A Proposal for a Formal Definition of the Design Concept. In: Lyytinen, K., Loucopoulos, P., Mylopoulos, J., Robinson, W. (eds.) Design Requirements Engineering: A Ten-Year Perspective. Lecture Notes on Business Information Processing, pp. 103-136. Springer-Verlag (2009)
7. Van de Ven, A.H.: Engaged Scholarship: A Guide for Organizational and Social Research. Oxford University Press, Oxford, UK (2007)
8. Celdman, J.: Designing with Web Standards. New Riders, Berkely, California, USA (2007)
9. Charette, R.N.: Why Software Fails. IEEE Spectrum Online, (2005), http://www.spectrum.ieee.org/sep05/1685

10. Koppel, R., Metlay, J.P., Cohen, A., Abaluck, B., Localio, A.R., Kimmel, S.E., Strom, B.L.: Role of Computerized Physician Order Entry Systems in Facilitating Medication Errors. Journal of the American Medical Association 293, 1197-1203 (2005)
11. Keizer, G.: A Third of New Pcs Being Downgraded to XP, Says Metrics Researcher. Computerworld, (2008)
12. Ralph, P.: Introducing an Empirical Model of Design. In Proceedings of The 6th Mediterranean Conference on Information Systems, Limassol, Cyprus (2011)
13. Arnott, D.: Cognitive Biases and Decision Support Systems Development: A Design Science Approach. Information Systems Journal 16, 55-78 (2006)
14. Samuelson, W., Zeckhauser, R.J.: Status Quo Bias in Decision Making. Journal of Risk and Uncertainty 1, 7-59 (1988)
15. Gheorghiu, V., Molz, G., Pohl, R.: Suggestion and Illusion. In: Pohl, R. (ed.) Cognitive Illusions: A Handbook on Fallacies and Biases in Thinking, Judgement and Memory. pp. 399-421. Psychology Press, Hove, UK (2004)
16. Kahneman, D., Tversky, A.: Prospect Theory: An Analysis of Decision under Risk. Econometrica 47, 263-291 (1979)
17. Buehler, R., Griffin, D., Ross, M.: Exploring the "Planning Fallacy": Why People Underestimate Their Task Completion Times. Journal of Personality and Social Psychology 67, 366-381 (1994)
18. Taylor, S.E., Kemeny, M.E., Aspinwall, L.G., Schneider, S.G., Rodriguez, R., Herbert, M.: Optimism, Coping, Psychological Distress, and High-Risk Sexual Behavior among Men at Risk for Acquired Immunodeficiency Syndrome (Aids). Journal of Personality and Social Psychology 63, 460-473 (1992)
19. Tversky, A., Kahneman, D.: Judgment under Uncertainty: Heuristics and Biases. Science 185, 1124-1131 (1974)
20. Parsons, J., Saunders, C.: Cognitive Heuristics in Software Engineering: Applying and Extending Anchoring and Adjustment to Artifact Reuse. IEEE Transactions on Software Engineering 30, 873-888 (2004)
21. Pohl, R. (ed.) Cognitive Illusions. Psychology Press, East Sussex, UK (2004)
22. Langer, E.: The Illusion of Control. Journal of Personality and Social Psychology 32, 311-328 (1975)
23. Janis, I.L.: Groupthink: Psychological Studies of Policy Decisions and Fiascoes. Houghton Mifflin (1982)
24. Matlin, M.: Pollyanna Principle. In: Pohl, R.F. (ed.) Cognitive Illusions: A Handbook on Fallacies and Biases in Thinking, Judgement and Memory. pp. 255-272. Psychology Press, Hove, UK (2004)
25. Fischoff, B.: Debiasing. In: Kahneman, D., Slovic, P., Tversky, A. (eds.) Judgment under Uncertainty: Heuristics and Biases. Cambridge University Press, Cambridge, USA (1982)
26. Larrick, R.P.: Debiasing. In: Blackwell Handbook of Judgment and Decision Making. pp. 316-338. Blackwell Publishing Ltd (2008)
27. Wilson, T., Centerbar, D., Brekke, N.: Mental Contamination and the Debiasing Problem. In: Gilovich, T., Griffin, D., Kahneman, D. (eds.) Heuristics and Biases: The Psychology of Intuitive Judgment. pp. 185-200. Cambridge University Press, Cambridge (2002)
28. Pronin, E., Lin, D.Y., Ross, L.: The Bias Blind Spot: Perceptions of Bias in Self Versus Others. Personality and Social Psychology Bulletin 28, 369-381 (2002)
29. Pronin, E.: Perception and Misperception of Bias in Human Judgment. Trends in Cognitive Sciences 11, 37-43 (2006)

30. Jørgensen, M.: Identification of More Risks Can Lead to Increased Over-Optimism of and Over-Confidence in Software Development Effort Estimates. Information & Software Technology 52, 506-516 (2009)
31. Thaler, R., Benartzi, S.: Save More Tomorrow: Using Behavioral Economics to Increase Employee Saving. The Journal of Political Economy 112, S164-S187 (2004)
32. Leventhal, L., Teasley, B., Rohlman, D., Instone, K.: Positive Test Bias in Software Testing among Professionals: A Review. In: Bass, L., Gornostaev, J., Unger, C. (eds.) Human-Computer Interaction. Lecture Notes in Computer Science, vol. 753, pp. 210-218. Springer Berlin / Heidelberg (1993)
33. Stacy, W., MacMillan, J.: Cognitive Bias in Software Engineering. Communications of ACM 38, 57-63 (1995)
34. Siau, K., Wand, Y., Benbasat, I.: When Parents Need Not Have Children — Cognitive Biases in Information Modeling. In: Constantopoulos, P., Mylopoulos, J., Vassiliou, Y. (eds.) Advanced Information Systems Engineering. Lecture Notes in Computer Science, vol. 1080, pp. 402-420. Springer Berlin / Heidelberg (1996)
35. Busby, J., Payne, K.: The Situated Nature of Judgement in Engineering Design Planning. Journal of Engineering Design 9, 271-291 (1998)
36. Snow, A.P., Keil, M., Wallace, L.: The Effects of Optimistic and Pessimistic Biasing on Software Project Status Reporting. Information & Management 44, 130-141 (2007)
37. Browne, G.J., Ramesh, V.: Improving Information Requirements Determination: A Cognitive Perspective. Information & Management 39, 625-645 (2002)
38. Schwaber, K.: Agile Project Management with Scrum. Microsoft Press (2004)
39. Nils, C.H.: An Empirical Study of Using Planning Poker for User Story Estimation. In Proceedings of Agile 2006, pp. 23-34. IEEE Computer Society, Visegrád, Hungary (2006)
40. Flyvbjerg, B.: From Nobel Prize to Project Management: Getting Risks Right. Project Management Journal 37, 5 (2006)
41. Lovallo, D., Kahneman, D.: Delusions of Success. Harvard Business Review 81, 56-63 (2003)
42. Molokken, K., Jorgensen, M.: A Review of Software Surveys on Software Effort Estimation. In Proceedings of International Symposium on Empirical Software Engineering, pp. 223-230. ACM-IEEE, Rome, Italy (2003)
43. Hevner, A.R., March, S.T., Park, J., Ram, S.: Design Science in Information Systems Research. MIS Quarterly 28, 75-105 (2004)
44. Hevner, A., Chatterjee, S.: Design Research in Information Systems: Theory and Practice. Springer (2010)
45. Sein, M., Henfridsson, O., Purao, S., Rossi, M., Lindgren, R.: Action Design Research. MIS Quarterly 35, (2011)
46. Gilovich, T., Griffin, D., Kahneman, D. (eds.) Heuristics and Biases: The Psychology of Intuitive Judgment. Cambridge University Press, Cambridge, UK (2002)
47. Newell, A., Simon, H.: Human Problem Solving. Prentice-Hall, Inc. (1972)
48. Stanovich, K.: What Intelligence Tests Miss: The Psychology of Rational Thought. Yale University Press, New Haven, CT, USA (2009)
49. Oliver, R.L.: A Cognitive Model of the Antecedents and Consequences of Satisfaction Decisions. Journal of Marketing Research 17, 460-469 (1980)
50. Taylor, N.: Making Actuaries Less Human: Lessons from Behavioural Finance. The Staple Inn Actuarial Society Meeting, (2000)
51. Standish Group: Chaos Database: Chaos Surveys Conducted from 1994 to Fall 2004. (2006),
52. Hartmann, D.: Interview: Jim Johnson of the Standish Group. InfoQueue, (2006)

53. Poses, R.M., Anthony, M.: Availability, Wishful Thinking, and Physicians' Diagnostic Judgments for Patients with Suspected Bacteremia. Medical Decision Making 11, 159-168 (1991)
54. Armor, D., Taylor, S.: When Predictions Fail: The Dilemma of Unrealistic Optimism. In: Gilovich, T., Griffin, D., Kahneman, D. (eds.) Heuristics and Biases: The Psychology of Intuitive Judgment. Cambridge University Press, Cambridge, UK (2002)
55. Colman, A.: A Dictionary of Psychology. Oxford University Press, Oxford, UK (2009)
56. VandenBos, G. (ed.) APA Dictionary of Psychology. American Psychological Association, Washington, DC, USA (2007)
57. Todd, P.M., Gigerenzer, G.: Environments That Make Us Smart: Ecological Rationality. Current Directions in Psychological Science 16, 167-171 (2007)
58. Renner, C.H.: Validity Effect. In: Pohl, R.F. (ed.) Cognitive Illusions: A Handbook on Fallacies and Biases in Thinking, Judgement and Memory. pp. 201-213. Psychology Press, Hove, UK (2004)
59. Jost, J.T., Banaji, M.R.: The Role of Stereotyping in System-Justification and the Production of False Consciousness. British Journal of Social Psychology 33, 1-27 (1994)
60. Bornstein, R., Carver-Lemley, C.: Mere Exposure Effect. In: Pohl, R.F. (ed.) Cognitive Illusions: A Handbook on Fallacies and Biases in Thinking, Judgement and Memory. pp. 215-234. Psychology Press, Hove, UK (2004)
61. Shore, B.: Systematic Biases and Culture in Project Failures. Project Management Journal 39, 5-16 (2008)
62. Omer, H., Alon, N.: The Continuity Principle: A Unified Approach to Disaster and Trauma. American Journal of Community Psychology 22, 273-287 (1994)