

The Rational Planning of (Software) Projects

Mark C. Paulk
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213-3890
U.S.A.

Abstract

The software crisis has persisted for decades. Our difficulties in planning and managing software projects may be rooted in fundamental human nature, as suggested by research in rational decision making, more than in the inherent difficulty of building software. The Capability Maturity ModelSM for Software, an application of the concepts of Total Quality Management to software development and maintenance, embodies one approach for improving the software process. The problems addressed by both the CMMSM and TQM seem to lie in the basic ways that human beings think and organize themselves. In many circumstances, normal human decision making can be characterized as “irrational” because of systematic biases and fallacies in the way people make decisions. Mechanisms such as those suggested by the CMM support rational decision making.

1 Introduction

Over the last three decades we have heard many complaints about the “software crisis.” In some software organizations, the typical software project is a year late and double the budget. In a review of one DOD software organization’s 17 major projects, the average 28-month schedule was missed by an average of 20 months. A four year project was not delivered for seven years; no project was on time.

A recent Government Accounting Office (GAO) report /6/ on the major software challenges states, “We have repeatedly reported on cost rising by millions of dollars, schedule delays of not months but years, and multi-billion-dollar systems that don't perform as envisioned.” The report summarizes over 20 GAO case studies involving software or software-related problems in the military. This report concludes, “The understanding of software as a product and of software development as a process is not keeping pace with the growing complexity and software dependence of existing and emerging mission-critical systems.”

This work is sponsored by the U.S. Department of Defense.

CMM and Capability Maturity Model are service marks of Carnegie Mellon University.

This paper was published in the **Proceedings of the First World Congress for Software Quality**, San Francisco, CA, 20-22 June 1995, section 4.

Is it accurate, however, to refer to a situation as a crisis that has persisted for decades? The software crisis is a chronic and severe problem that needs to be addressed, but it does not seem to be acute.

Some software savants have suggested that in other disciplines, such as construction, an estimate off by more than 6% is considered a disaster. Studies indicate, however, that “in capital investment projects, the typical project finishes late, comes in over budget when it is finally completed, and fails to achieve its initial goals.” /10/ For the projects studied, the norm was for actual construction costs to more than double first estimates.

The problems facing the software industry may be more ubiquitous than we sometimes think. Many of the problems contributing to the software crisis derive more from human nature and ability than they do from the inherent complexity of software systems, daunting though that complexity may be. The “software crisis” is not unique to the software industry.

A sizable body of literature on rational decision making and human foibles has grown over the last few years. /1/ The purpose of this paper is to summarize some of those results, how they apply to the planning of software projects, and how the Capability Maturity Model for Software (CMM) /13, 14/ developed by the Software Engineering Institute (SEI) can contribute to making rational decisions, specifically when planning software projects.

2 Systematic Biases in Human Nature

Software engineering is a human-intensive, design-intensive endeavor. The largest single factor in the success of software projects is the competence of the people doing the work. People are fallible, however, and “an accumulating body of recent research on clinical judgment, decision making, and probability estimation has documented a substantial lack of ability across both individuals and situations.” /4/

In this paper we shall concentrate on three general biases that all humans are systematically subject to:

- people tend to be risk averse when there is a potential of loss
- people are unduly optimistic in their plans and forecasts
- people prefer to use intuitive judgment rather than (quantitative) models

Managers are people also, and these human tendencies apply to the managers who make business decisions at the project to the executive level.

2.1 Problem: avoiding risk and uncertainty

In the business world, managers face contingencies they cannot predict and outcomes they will not control, no matter how intelligent, educated, or experienced they are. This variability can be characterized as risk. Managers

agree that risk taking is essential to success in decision making and an essential component of the managerial role. /12/

According to game theory, decisions should be made that maximize expected value. In reality, people are risk averse when dealing with the possibility of loss. For many managers, studies indicate that losses loom about twice as large as gains. /10/ Avoiding risk could be characterized as being prudent, but the axiom “nothing ventured, nothing gained” also applies, especially in an increasingly competitive world.

What are the some of the factors that drive conservatism? First, society¹ values risk taking, but not gambling, and society defines gambling as risk taking that turns out badly. /12/ The results sort decision makers into winners and losers, and society interprets these differences as reflecting differences in judgment and ability, although events outside the individual’s control may have strongly influenced success.

Second, the consequences of acting versus not acting are evaluated differently. This biases decision makers toward passivity, even when passivity may not be in their own best interests. /10/ In general, loss aversion favors conservatism and inertia in decision making. When a manager takes a risk, it is often because of optimistic denial rather than bold acceptance.

It is better, therefore, not to act than to act and be wrong; if you do act, you need to be right. Since acting – taking risks – is an essential part of a manager’s job, the conflict results in the denial of uncontrollable risks. Kahneman and Lovallo state that “for managers, risk is a challenge to be overcome by the exercise of skill, and the role of uncontrollable contingencies is to be denied or minimized.” /10/ While some risks can be allowed for in planning, many other risks are outside the control of an individual manager.

2.2 Problem: undue optimism

While there is a tendency for managers to be conservative, there is a counterbalancing bias that people tend to be overly optimistic about what they can do. Kahneman and Lovallo state that “there are three main forms of pervasive optimistic bias: 1) unrealistically positive self-evaluations, 2) unrealistic optimism about future events and plans, and 3) an illusion of control.” /10/

When there is little effective feedback on how good a planning effort was, judgments are not well calibrated. In one study, Einhorn and Hogarth comment

¹ Society may not be the best term, but it seems the most general. March and Shapira use “society” and “history.” An alternative would be “your company,” but that seems inappropriate. Society as used here can be considered a term for any grouping of people.

that “self-confidence in judgments... was found to increase as a function of the amount of information available... but without any corresponding increase in judgmental accuracy.” /4/ We plan, we act, we replan, we react – and the accuracy of the judgments that contribute to the rework become lost in the noise of the ad hoc, chaotic environment of the typical project.

Managers also tend to ignore possible events that are very unlikely or very remote, regardless of their consequences. /12/ Managers focus on a few possible outcomes – the norm – rather than the whole distribution, and plan with respect to those few points. This aspect of denying risk also leads to unrealistic optimism.

Denial seems to be a common solution to many of the conflicting forces managers have to deal with. As March and Shapira express it, “Managers focus on ways to reduce the danger while retaining the gain. One simple action is to reject the estimates.” /12/ This is hardly unknown in the software world. Proposal managers have been known to cut a schedule by 25%, because the realistic bid would not win the contract. Even after estimates have been developed and revised, most managers believe they can do better than is expected. /12/

Undue optimism is perhaps not a major problem in the software world, though a feeling of futility may be. Few experienced software managers claim great confidence in their planned schedules, because of their experience with:

- volatile requirements. How closely will the system eventually built correspond to the one originally planned?
- rapidly changing technology. Is the underlying technology likely to go through one or more evolutionary advances during development?
- historical performance. How do we define good performance for a software project? Within 50% of estimate?

2.3 Problem: relying on intuitive judgment

Denial of risk and optimism are much easier when there is nothing to rub your nose in the facts. People resist documenting procedures and standards and using quantitative models. Typical reasons include fear of bureaucracy and rigid control of a dynamic process. These reasons are valid, if the process is poorly implemented.

Managers are more comfortable with verbal characterizations of risk than with numerical characterizations. March and Shapira found that “A majority... felt that risk could not be captured by a single number or distribution, that quantification of risks was not an easy task, ... there was no way to translate a multidimensional phenomenon into one number... [but] everything should be expressed in terms of the profit (or loss) at the end of the project.” /12/

Surveys of estimating practices in one organization /8/ indicate that disciplined estimating approaches, such as Delphi techniques or cost models, are rarely used. Variances are so large that there is a 30% probability that any one estimate can be more than 50% off. The vast majority of projects are planned based on intuitive judgment, and success is declared at the level of functionality present when the schedule or money run out. This is a fairly common state of affairs in many software development and maintenance environments.

The result of the intuitive approach to software estimating is that software planning is frequently considered a wasted effort. We thus have the worst of both possibilities mentioned by Einhorn and Hogarth: little judgmental accuracy and little confidence in spite of (or because of) our experience. We cannot address risk aversion and overoptimism biases without also dealing with also addressing the systematic fallacies inherent in global intuitive judgment.

2.4 Summing up the problem

As Hihn and Habib-Agahi point out, “when a person is forced to plan under severe schedule and budget constraints, planning consists mostly of a prioritized list and a lot of optimistic promises.” /8/

Because most software organizations that we have observed (about 75% according to the SEI’s 1994 data) do not use a disciplined approach, we consistently see the results of poor project planning and management evidenced as:

- unrealistic plans, based on optimistic estimates of what can be done (and managerial pessimism regarding the bids that need to be proffered to win a contract),
- ineffective tracking of performance, based on a lack of “measured insight” into how a project is progressing (“rework happens” being the confounding factor),
- volatile requirements that confound estimates and are incorporated into the project in an ad hoc fashion, and
- risks that unexpectedly, but not unpredictably, become catastrophes.

Can we learn from experience? Can we control these natural tendencies in human nature? Einhorn and Hogarth suggest that “the difficulty of learning from experience has been traced back to three main factors: (a) lack of search for and use of disconfirming evidence, (b) lack of awareness of environmental effects on outcomes, and (c) the use of unaided memory for coding, storing, and retrieving outcome information.” /4/

3 Controlling Human Nature

March and Shapira suggest that “it may be more efficacious to try to modify managerial attention patterns and conceits than to try to change beliefs about the

likelihood of events or to try to induce preferences for high variance alternatives.” /12/ An example of the impact of paying attention is that people tend to optimize what is measured, which can confound the effectiveness of measurement programs.

The SEI’s Capability Maturity Model for Software (CMM) recommends institutionalizing measures and procedures based on historical performance and statistical analysis as “the accepted way of doing business,” i.e., part of an organizational culture. Tools and techniques, such as Delphi techniques and estimation models, have been developed to give some degree of objectivity to planning and overcome human frailty.

Specifically, the CMM recommends:

- documenting the way work is performed
- measuring its performance
- using data for controlling the performance
- planning based on historical performance
- training people

The CMM is structured into five maturity levels, which can be briefly described as:

- 1) *Initial* The software process is characterized as ad hoc, and occasionally even chaotic. Few processes are defined, and success depends on individual effort and heroics.
- 2) *Repeatable* Basic project management processes are established to track cost, schedule, and functionality. The necessary process discipline is in place to repeat earlier successes on projects with similar applications.
- 3) *Defined* The software process for both management and engineering activities is documented, standardized, and integrated into a standard software process for the organization. All projects use an approved, tailored version of the organization's standard software process for developing and maintaining software.
- 4) *Managed* Detailed measures of the software process and product quality are collected. Both the software process and products are quantitatively understood and controlled.
- 5) *Optimizing* Continuous process improvement is enabled by quantitative feedback from the process and from piloting innovative ideas and technologies.

These maturity levels, in turn, are decomposed into 18 key process areas. This paper concentrates on one of those areas: Software Project Planning. In the balance of this paper we shall review some of the ways of dealing with the systemic biases that people have and how the CMM can help with these biases.

3.1 Solution: dealing with risk and uncertainty

One of March and Shapira's observations is that risk taking in organizations is sustained more by personal than by organizational incentives. /12/
Organizational support in understanding and prioritizing risks can be a critical component of empowering managers to deal with risk more effectively (note that the CMM is a model for organizational improvement). Activity 13 of Software Project Planning addresses identifying, assessing, and documenting software risks.

Risk management may not be a precise science, but it is a reasonable approach to controlling contingencies that, while they may not be predictable in detail, are predictable in general. The exercise of documenting and tracking (foreseeable) risks is valuable in keeping us from losing them below our attention threshold.

3.2 Solution: being realistic

Mechanisms such as those advocated by the CMM can help a manager be more realistic in planning software projects. These include using documented procedures and standards for estimating and planning and taking advantage of historical performance data.

That does not remove the need to win the contract or to capture the customer, as a necessary first step for doing business. Kahneman and Lovallo point out that winning projects are more likely to be associated with optimistic errors. /10/
Time to market is a critical factor in the profitability of commercial products. Being realistic does not imply not being aggressive.

Realistic here does mean that managers have to decide whether software engineering is a "manageable" process. The CMM takes the position that it is and advocates a particular approach for evolving the process capability of an organization. This is not a unanimous view. Einhorn characterizes two possible classes of error when making judgments. If people decide that a phenomenon is systematic when it is random, the error that results is manifested in myths, magic, superstitions, and illusions of control. If people decide that a phenomenon is random when it is systematic, the error that results is manifested in lost opportunities and illusions of the lack of control. /5/

Managers have to act in the absence of complete information, what the military calls the "fog of war." Documented procedures and standards can help focus

attention on issues that might otherwise be neglected; a logical complement is the use of quantitative models to support the judgments that are made.

In the academic world, we can perform controlled experiments. The concept of a control group is essential to the scientific method, but controlled experiments are typically impractical in the real world. We can, however, capture feedback on judgments, actions, process, environmental variables, and the resulting outcomes in the real world, which is an essential part of improving judgment.

Human beings are fallible. Rational decision making depends on the tools available to the decision maker. The rational, intelligent use of those tools provides a means for organizational learning. There are three conditions for learning: (1) feedback, which is necessary but not sufficient; (2) the ability to rearrange cases so that hypotheses can be verified or discounted; and (3) the ability to tally the accuracy of one's hypotheses. /4/

3.3 Solution: using models

The literature is rife with studies that demonstrate the power of even simple statistical models for combining information over the judgments of human experts. /4/ Dawes, Faust, and Meehl compared the power of the clinical method, where the decision-maker combines or processes information in his or her head, and the actuarial method, where the human judge is eliminated and conclusions rest solely on empirically established relations between data and the condition or event of interest.² Their conclusion was that the actuarial method is almost unanimously equal or superior to the clinical method. In nearly all of the comparative studies in the social sciences, the actuarial method has equaled or surpassed the clinical method. /2/

This is not an intuitively obvious result. Studies /1, 2, 4, 10/ have demonstrated that:

- simple statistical models based on human judges are more often correct than the human judges they are based on³
- even when human judges are provided the outputs of the model and allowed to use their discretion, relying uniformly on the actuarial conclusions provides greater overall accuracy than the human judge⁴

² Dawes, Faust, and Meehl state that “to be truly actuarial, interpretations must be both automatic (that is, prespecified or routinized) and based on empirically established relations.” They then go on to say that “virtually any type of data is amenable to actuarial interpretation.” While the latter quote may be an overstatement, it certainly applies in the domain of software project estimating.

³ One of the advantages of models is that you get consistent answers when given the same inputs.

⁴ When operating freely, human judges apparently identify too many “exceptions.”

- optimal weighting of variables, so long as the sign of the coefficients are correct, is not crucial to performing better than the human judge⁵

Kahneman and Lovallo explain this phenomenon in terms of inside and outside views. /10/ An inside view forecast is generated by focusing on the case at hand, by considering the plan and the obstacles to its completion, by constructing scenarios of future progress, and by extrapolating current trends. An outside view forecast focuses on the statistics of a class of cases chosen to be similar in relevant respects to the present one. The critical question in both cases is whether a to treat a particular problem as unique or as an instance of a class of similar problems.

The inside view is overwhelmingly preferred in intuitive forecasting because it is viewed as a serious attempt to come to grips with the complexities of the unique case at hand. The outside view is rejected for relying on crude analogy from superficially similar instances. Critics of software process management, who emphasize the innovative, creative aspects of software engineering, are basically arguing from the inside view.

Take a simple example. /5/ There is an urn with 60% red balls and 40% green balls. Predict the color of balls as they are drawn from the urn. Knowing the odds, most people use a rule that leads to predicting red 60% of the time and green 40% of the time. That leads to 52% correct predictions ($.060 * 0.60 + 0.40 * 0.40$). The simple rule “always predict the most likely color” leads to 60% correct predictions. Such a strategy deliberately accepts error, but when dealing with a truly random process, it is the best selection rule.

The future of a long and complex undertaking is simply not foreseeable in detail. The outside view is a conservative approach, which will fail to predict extreme and exceptional events, but will do well with common ones; it is much more likely to yield a realistic estimate. Its main advantage is that it avoids the snares of scenario thinking.⁶

⁵ Optimal weights are specific to the population in which they were derived, and any advantage gained in one setting may be lost when the same method is applied in another setting. Equally weighted linear regression models can outpredict models with “optimal” weights on new cases if there is poor data.

⁶ In scenario thinking, a sequence of events may be judged more probable than one of its components. Dawes /1/ gives the example of an alcoholic tennis star who drinks a fifth a day winning a major tournament versus an alcoholic tennis star who drinks a fifth a day, joins Alcoholics Anonymous, quits drinking, and wins a major tournament. The probability of winning is higher than the probability of (joining AA) * (quitting drinking) * (winning), yet the sequence of events in the scenario sounds more plausible. The software planning equivalent could be making an aggressive commitment and bringing the contract in on time and on budget versus making the commitment, hiring highly competent people, providing powerful new tools and methods, and bringing the contract in on time and on budget.

Dawes, Faust, and Meehl state that “although surpassing clinical methods, actuarial procedures are far from infallible, sometimes achieving only modest results.” /2/ This highlights a difficult problem in using such techniques – people would rather remain ignorant of how bad their own judgment is than rely on “mechanical” techniques where they are all too aware of the limitations.

Models are simplifications of the real world by definition. They can never capture the richness and complexity of the real-world phenomenon they describe; therefore there will be errors in prediction. Would we be willing to accept such an obviously oversimplistic rule as “always predict the most likely color” if we were gambling? If we were managing risks in a project that we were responsible for?

Human beings do have a unique capacity to observe and make “atomic judgments,” but this is not the same as a unique capacity to predict on the basis of integration of observations. /2/ Human beings can build models, but when it comes to consistently performing better than the models they have produced, they are almost invariably unsuccessful – unless they “load the dice” by taking action that makes their predictions come true.

March and Shapira argue that “it is entirely sensible for a manager to conclude that the credibility of probability estimates is systematically less than is the credibility of estimates of the value of an outcome, and it is certainly arguable that the relative credibility of estimates should affect the relative attention paid to them.” /12/ However, having a rational reason, rather than a gut feel, for making managerial decisions provides a foundation for learning, both as individuals and organizations.

3.4 Summing up a solution

There has been a consistent theme running through this discussion of controlling human frailty:

- document how decisions are made
- provide guidance and quantifiable criteria where possible
- record decisions and the data used to make them
- analyze results and improve the process where possible
- learn – individually and organizationally

This is process management as advocated in the CMM. It is not a panacea to the difficult problems of management, but it is a powerful approach to attacking those problems.

March and Shapira argue that “We may prefer to have managers imagine (sometimes falsely) that they can control their fates, rather than suffer the consequences of their imagining (sometimes falsely) that they cannot.” /12/ Is ignorance bliss?

As George Box said, “All models are wrong; some models are useful.” If the models we use are successful, even if embedded in a human expert’s head, then there is little demand for changing the current way of doing business. That is not the case in the software world, as we have already shown. A strong need has been expressed for improvement in the way we develop and maintain software.

4 Empirical Data and Experimentation

There are a number of case studies of software process improvement based on the CMM /3, 9, 11, 15/. All indicate that predictability, control, and effectiveness of processes can be significantly improved by improving the software process as suggested by the CMM. The CMM, as argued in this paper, can help organizations control natural human biases related to decision making.

The typical return on investment, based on data from organizations that have done software process improvement for more than 3 years, is about 7:1, with an average gain in productivity of 37% per year, an average 18% increase each year in the proportion of defects found in pre-test, an average 19% reduction in time to market, and an average 45% reduction in field error reports per year.

This is not a statistically rigorous analysis. The SEI began such an effort in 1993, but it will require several years to develop rigorous evidence on the effects of CMM-based improved efforts, although some initial results have been published. /7/ That does not mean, however, that organizations cannot demonstrate the value of these concepts more quickly.

As already mentioned, there is significant resistance in many organizations to the concept of documented processes. They are viewed as being bureaucratic, rigid, and stultifying. Although there are instances where such counter-productive processes have been mandated by organizations, there are also instances where powerful, empowering, effective processes have been developed and deployed across organizations.

To overcome this resistance, it may be necessary to prototype and work our way up in complexity and criticality. Most sizable organizations have a range of software projects. Some are large, complex, and critical to the business. Others are comparatively small and/or of little impact. Software professionals value the concept of prototyping, so it seems reasonable to propose prototyping the use of documented standards, procedures, and models. Champions can then be identified, based on the success (or perhaps failure!) of the prototyping effort.

What would the confounding factors be? First, there might be a Hawthorne effect. Second, we would primarily be using outcome information. For small projects, it may be difficult to incorporate process feedback. Third, people will be acting on the basis of the plans and estimates generated, which may confound

the results. Fourth, the customer is a critical component of the overall system. An unreasonable customer can cripple the process.

5 Conclusion

As Kahneman and Lovallo suggest, facing the facts can be intolerably demoralizing. A realistic view may not provide an acceptable basis for continuing an ongoing project, but no one is willing to abandon the sunk costs and draw the inescapable conclusion that the project should be scrapped – at least until far more money has been expended than was originally anticipated.

Perhaps unfounded optimism is the only effective remedy against paralysis; perhaps there is a genuine dilemma that will not yield to any simple rule. Ignoring the issue puts us in the same position as the lost platoon in Weick's famous story that finds its way in the Alps by consulting a map of the Pyrenees.

The CMM provides a powerful tool for attacking our software management problems. We may be horrified by how "bad" some of the solutions are – yet the really terrifying observation may be that they are better than the current state of affairs. They also provide a foundation for learning and improvement that is currently lacking – and that is the critical driver for beginning the journey of continuous process improvement.

References

1. Robyn M. Dawes, *Rational Choice in an Uncertain World*, Harcourt Brace Jovanovich College Publishers, Orlando, FL, 1988.
2. Robyn M. Dawes, David Faust, and Paul E. Meehl, "Clinical Versus Actuarial Judgment" *Science*, Vol. 243, 31 March 1989, pp. 1668-1674.
3. Raymond Dion, "Process Improvement and the Corporate Balance Sheet," *IEEE Software*, Vol. 10, No. 4, July 1993, pp. 28-35.
4. Hillel J. Einhorn and Robin M. Hogarth, "Confidence in Judgment: Persistence of the Illusion of Validity," *Psychological Review*, Vol. 85, No. 3, 1978, pp. 395-416.
5. Hillel J. Einhorn, "Accepting Error to Make Less Error," *Journal of Personality Assessment*, Vol. 50, No. 3, Fall 1986, pp. 387-395.
6. *Mission-Critical Systems - Defense Attempting to Address Major Software Changes*, General Accounting Office, GAO/IMTEC-93-13, December 1992, pp. 1-29.

7. James Herbsleb, Anita Carleton, et al., "Benefits of CMM-Based Software Process Improvement: Initial Results," Software Engineering Institute, CMU/SEI-94-TR-13, August 1994.
8. J. Hihn and H. Habib-Agahi, "Cost Estimation of Software Intensive Projects: A Survey of Current Practices," *Proceedings of the 13th International Conference on Software Engineering, Austin, TX, 13-17 May 1991*, pp. 276-287.
9. Watts S. Humphrey, T.R. Snyder, and Ronald R. Willis, "Software Process Improvement at Hughes Aircraft," *IEEE Software*, Vol. 8, No. 4, July 1991, pp. 11-23.
10. Daniel Kahneman and Dan Lovallo, "Timid Decisions and Bold Forecasts: A Cognitive Perspective on Risk Taking," *Proceedings of Conference on Fundamental Issues in Strategy, Silverado, 1990*.
11. W.H. Lipke and K.L. Butler, "Software Process Improvement: A Success Story," *Crosstalk: The Journal of Defense Software Engineering*, No. 38, November 1992, pp. 29-31.
12. James G. March and Zur Shapira, "Managerial Perspectives on Risk and Risk Taking," *Management Science*, Vol. 11, No. 11, November 1987, pp. 1404-1418.
13. M.C. Paulk, B. Curtis, M.B. Chrissis, and C.V. Weber, "Capability Maturity Model for Software, Version 1.1," Software Engineering Institute, CMU/SEI-93-TR-24, DTIC Number ADA263403, February 1993.
14. M.C. Paulk, C.V. Weber, S. Garcia, M.B. Chrissis, and M. Bush, "Key Practices of the Capability Maturity Model, Version 1.1," Software Engineering Institute, CMU/SEI-93-TR-25, DTIC Number ADA263432, February 1993.
15. H. Wohlwend and S. Rosenbaum, "Software Improvements in an International Company," *Proceedings of the 15th International Conference of Software Engineering, Washington D.C, May 1993*.