A Classification of Requirements Elicitation Problems

Dr Bill Davey
RMIT University, Melbourne Australia
Bill.davey@rmit.edu.au

Dr Kevin R. Parker
Idaho State University
parkkevi@isu.edu

Abstract: This paper proposes a classification of problem types that occur in requirements elicitation. The classification has been derived from a literature analysis. Papers reporting on techniques for improving requirements elicitation practice were examined for the problem the technique was designed to address. This classification allows the requirements engineer to be sensitive to problems as they arise and the educator to structure delivery of requirements elicitation training.

Introduction

An important part of requirements gathering is obtaining requirements from people: requirements elicitation. Failure of information systems is common and effective requirements elicitation (RE) is an important factor in avoiding system failure – the most efficient and well-engineered system must be useful to end users and that is contingent on the right specifications being obtained in the first instance. The purpose of this paper is to present a view of what can go wrong in requirements elicitation based on a study of published research designed to improve RE. First some justification will be made for the contentions that systems failure is common and that good RE is important. Then the results of the literature analysis will be explained.

Failure and abandonment rates

The industry would like to be able to use the term 'engineering' when talking about professional activities. While civil engineering projects are abandoned, and sometimes fail, collapsing bridges or buildings tend to make the international news. This is also the case with aeronautical engineering problems. Using these other engineering areas as a benchmark, information systems figures for failure are terrible.

The Standish Group has conducted large-scale surveys of project managers from 1994 until 2008 and have reported the findings in the 'Chaos reports'. These surveys classified project outcomes as 'success', 'challenged' or 'failed'. A summary by Eveleens and Verhoef of the first five Standish surveys is shown in Table 2.2 and indicates that outright success of information systems projects is rare.

Year	Success	Challenged	Failed
1994	16%	53%	31%
1996	27%	33%	40%
1998	26%	46%	28%
2000	28%	49%	23%
2004	29%	53%	18%

Table 0.1 Success rates of IS projects taken from Eveleens & Verhoef (2010, p2)

The Standish surveys have been replicated by others in the United States with similar outcomes (Emam & Koru, 2008). These empirical results are supported by authors of studies of information systems failure (Anbari, 2003; Basili & Boehm, 2001; Briggs & Gruenbacher, 2002; Dalcher & Drevin, 2003; Eberlein & Leite, 2002; Lamsweerde, 2000; Pan, Shan, and Flynn 2004). Studies have found very similar problems in other countries. For example Sauer looked at the UK. In 565 projects they found 5% abandoned and 55% over budget and 20% of projects delivered with less than 80% of specifications (Sauer & Cuthbertson, 2002). Also in the UK a study in 2000 found that *"Out of 1,027 projects covered only 130, or 12.7%, were successful."* (Taylor, 2000, p3). These examples are consistent with several investigations of failed projects in Australia (Brouwer, 2011; Pearson, 2012), the Netherlands (Tut, 2009) and China (Xiangnan, Hong, & Weijie, 2010).

These gloomy figures for information systems project failure, although consistent, should be read with some note of warning. The most common definition of failure in this context is when a project goes over time, over budget or does not meet its objectives in any way. There is some evidence (Glass, 2002, 2005; Savolainen, Ahonen, & Richardson, 2012) that 'failed' projects are sometimes delivered and go on to meet company needs. While this definition of project failure is seen by some authors to be inappropriate there seems to be no commonly accepted definition that distinguishes between outright failure and poor estimation of cost or time. Some authors make a distinction between the success of the project management and the success of the project, for example Savolainen, Ahonen, & Richardson (2012). Of course this does not apply to the case of abandonment of the project where nothing is delivered.

Cost of failure

The cost of this failure is enormous. Estimates of the cost of failure of systems in the USA have been between $59 billion (Lyytinen & Robey, 1999) and $81 billion (Dalcher & Drevin, 2003). More recent studies showed this had risen to between $84 billion (Reichental, 2006) and $100 billion (Miller & Luse, 2004). Miller and Luse (2004) report that a study by Clark in 2002 of 134 companies in the United States, United Kingdom, Africa, and Australia, found that 56 percent of the companies had cancelled at least one IS project during the last year at an average cost of US $13.6 million.

A very recent case in which a thorough study has been conducted by an independent governmental authority is that of an education system in the Australian State of Victoria. The system, called 'the Ultranet', was investigated by the State Auditor General, D. D. R. Pearson, and so the findings were the result of an unfettered investigation. It was found that the system, originally to have cost $60.5 million Australian dollars but was now expected to cost about $180 million by June 2013. It was also found that the system was used by only 10% of students and that only 27% of teachers had logged in from July 2011 to May 2012 (Pearson, 2012).

In the same jurisdiction a study by the Ombudsman, G. E. Brouwer, of ten ICT enabled projects found that $1.44 Billion Australian dollars had been spent in excess of budget and that several had failed or been abandoned with no outcome for the expenditure (Brouwer, 2011). The detailed study of ten projects included found that none of the projects had met expectations and that the cost of their failure was significant:

"On average, projects will have more than doubled in cost by the time they are finished. Two of the projects will have more than tripled their original budgets in order to reach completion: CRIS, originally budgeted at $22 million, has cost $70 million; and Link, originally budgeted at $59 million, would cost $187 million if it were to be completed. Together, the two largest projects will require almost $600 million more than originally planned: myki, originally costed at $999 million, will require at least an additional $350 million to complete and HealthSMART, originally budgeted at $323 million, will require an additional $243 million to complete" (Brouwer, 2011, p4).

Clearly the failure of information systems projects is a global problem and of significant size.

Persistence of failure

Failure of IS projects is not only at high levels, but has been persistent over many years. In 1997 Sauer summarised ten years of research by concluding:

"IS failure remains an important, unsolved problem. Failures have been (1) persistent, insofar as failure rates have not declined over the last 10 years (Johnson 1995), (2) pervasive, insofar as failures occur even in companies with a track record of successful IS, such as American Airlines (Oz 1994) and the Commonwealth Bank of Australia (Maiden1996), and (3) pernicious, insofar as direct and consequential costs can be severely damaging" (Sauer, Southon, & Dampney, 1997, p349).

Since that time commentary on information systems project failure has continued to indicate that little progress is being made in this costly problem.

The very large survey conducted by the KPMG company in 2005 found that

"IT project success has improved marginally since the last survey" (KPMG, 2005, p1).

Investigations into failure in specific jurisdictions continue to find significant failures like that of the Dutch IT industry

"Even in business, health care and education, it occasionally goes wrong. Computable put 22 remarkable Dutch ICT failures in 2009 among them." (Tut, 2009) (translated from the Dutch original).

Other studies showing the persistence of failure in the UK, Australia, China and the United States include those by Brouwer (2011); Pearson (2012); Shepherd, Patzelt, & Wolfe (2011); Taylor (2000); Xiangnan, Hong, & Weijie (2010).

This persistence is remarkable as significant work has been done in addressing failure and it would be expected that this work would impact professional practice and hence performance. Stoica and Brouse capture this apparent anomaly as an issue in saying

"In the field of Information Technology (IT) there is an observable trend toward project failure. Although multiple actions have attempted to address this failure trend, they have not impacted the extent of the trend" (Stoica & Brouse, 2013, p728).

It seems clear from these studies over a period of time, and including very recent work, that information systems project failure continues to be a significant problem.

The role of RE in Information Systems project failure

The next proposition to be examined is that RE is an important aspect of failure. There is no suggestion here that RE is the only or the critical factor in failure, but that, as an individual factor it is significant when compared with others.

The contention can be broken into two propositions:

- The greatest contribution to system failure comes from poor RE
- The cost of fixing RE problems is higher than other sources of error.

There is a general agreement that poor RE is an important and potentially damaging part of building a system. Fred Brooks puts a view commonly held (and quoted) in discussions of project failure:

"The hardest part of building a software system is deciding precisely what to build. No other part of the conceptual work is as difficult as establishing the detailed technical requirements, including all of the interfaces to people, to machines & to other software systems. No other part of the work so cripples the resulting system if done wrong. No other part is more difficult to rectify later" (Brooks, 1987, p17*)*.

Other commentators (Byrd, Cossick, and Zmud, 1992; G. B. Davis, 1982; Hickey & Davis, 2004; Vessey & Conger, 1993; Wetherbe, 1991) report that requirements elicitation is important using different phrases such as

"poor requirements had a critical impact on software quality" (T. E. Bell & Thayer, 1976, p62).

The Standish reports are used to justify a claim that more than half of overruns and failures occur due to poor RE (Briggs & Gruenbacher, 2002; Eberlein & Leite, 2002; Lamsweerde, 2000). This finding is replicated in other studies in Europe (Lamsweerde, 2000) and in studies by Barry Boehm (2000) Liebowitz (1999) and Moløkken-Østvold & Jørgensen (2003). Over a similar period various attempts have been made to measure the effect of poor requirements elicitation. Variously these studies have resulted in figures ranging from 12 per cent to 71 per cent of failure being attributed to poor requirements elicitation:

"accurately capturing system requirements is the major factor in the failure of 90% of large software projects" (C. J. Davis et al., 2006), "Poor requirements management can be attributed to 71 percent of software projects that fail; greater than bad technology, missed deadlines, and change management issues" (Lindquist, 2005, p54).

There is also a general agreement amongst commentators that fixing the results of poor RE is more expensive than for other mistakes.

"75 per cent of the cost of error removal has its origin in errors in the analysis stage of a project" (Urquhart, 1999, p44).

Again commentators are prepared to estimate from their experience. For example Boehm (1981) estimated that corrections made to requirements later in the development process could cost up to 200 times as much as corrections during the analysis phase.

Significant research into information systems project failure has been conducted using the general framework of project management. In this domain the single most common work is the project management core body of knowledge (PMBoK) (IEEE Computer Society, 2008). This document points to the importance of RE as it happens at the start of the project and the cost of rectifying changes is most affected by this stage.

It seems clear from this general agreement that RE is critical to project success. In the next section evidence is examined as to the relevance of the interview of clients to good RE.

Method

A search for literature was conducted using the databases: ACM: Association for Computing Machinery Digital Library, Expanded Academic ASAP (Gale), IEEE Xplore, ProQuest, CiteSeerX, Science Citation Index Expanded, and ScienceDirect (Elsevier).

Papers (1680) were filtered by removing those with no mention of improvements or identification of problems in RE resulting in 420 final papers. Papers reporting empirical research and meta analyses were given more weight. Papers were then assigned a number of potential key concepts in terms of the problem solved by the treatment proposed or problems identified. These key concepts were then aggregated into the problem types.

A classification of problems in RE

Analysis of the literature generated a very long list of problems identified as leading to poor requirements elicitation. This list comes from both those who have summarised the literature of RE and those writers who use their own judgement as to some underlying causes of poor RE and proceed to investigate some solution to those causes. It is clear from the wide variety of sources and range of reported problems that RE is a complex and difficult task. It is unlikely that a simple "solution" exists to such a complex problem.

Meta analysis of the literature of RE has been conducted by many authors including (Appan & Browne, 2012; Dieste & Juristo, 2011; Hansen et al., 2009),(Zave, 1997), (H. J. Harris, 2006), (Urquhart, 1999), (A. Davis et al., 2006), (A.M. Hickey & Davis, 2004), (Reichental, 2006),(Marakas & Elam, 1998); (Pan, 2005), (Davidson, 1996; Finkelstein, 1994; Majchrzak et al., 2005; Pitts & Browne, 2007; Zowghi & Coulin, 2005) and in a less formal way by (Jain et al., 2003)

Many other authors merely state a type of problem as an assumption underlying their study of a 'cure.'

The identified problems with RE can be summarised in nine categories:

There are human aspects of RE that preclude simple communication between consultant and client:

- Humans have cognitive limitations that preclude complete communication (G.J. Browne & Ramesh, 2002; Jain et al., 2003; Majchrzak et al., 2005; Pitts & Browne, 2007)
- People have different cultures (business or social) and backgrounds and so a common language does not always exist (Alvarez, 2002; Barry Boehm et al., 2001; Davidson, 1996; H. J. Harris, 2006; Jain et al., 2003; Saiedian & Dale, 2000; Urquhart, 1999; Zave, 1997; Zowghi & Coulin, 2005)
- Technical people do not understand business concepts and business people do not understand IT concepts (Barry Boehm et al., 2001; Davidson, 1996; Jain et al., 2003; Saiedian & Dale, 2000; Tsumaki & Tamai, 2006; Zowghi & Coulin, 2005)
- Some plain language statements can mean two different things eg words can be synonyms or homonyms (Tsumaki & Tamai, 2006)
- The amount of information presented can be too large to analyse (Tsumaki & Tamai, 2006)
- The way people express problems can be misleading (Arthur & Groner, 2005)
- Technical people often have poor communication skills (Pitts & Browne, 2004, 2007; Saiedian & Dale, 2000; Urquhart, 1999; Zowghi & Coulin, 2005)
- Business people do not have the communication skills to clearly state their needs (Pitts & Browne, 2007)

- People who must be consulted disagree on what the requirements should be (Alvarez, 2002; Briggs & Gruenbacher, 2002; Davidson, 1996; Tsumaki & Tamai, 2006; Zave, 1997)

The language of humans is not always suitable for technological solution

- Many terms used in the real world eg 'user friendliness' and 'reliability' do not have exact meanings in a technical sense (Zave, 1997)
- Some statements of the problem cannot be used to create a solution because of their form or language (Zowghi & Coulin, 2005)
- Not everything that can be done has equal importance. In a technical solution everything included has the same priority (Zave, 1997)
- Some methodologies have gaps (H. J. Harris, 2006; Jain et al., 2003)
- The problem is interpreted as being bigger than the originally intended problem (Pitts & Browne, 2004; Tsumaki & Tamai, 2006)
- Real world problems are very complex. They are also wicked. A wicked problem is one where the definition of the problem is difficult. (Briggs & Gruenbacher, 2002; G.J. Browne & Ramesh, 2002; Davidson, 1996; Nguyen, Armarego, & Swatman, 2002; Saiedian & Dale, 2000)
- Some consultants find the process difficult and rely upon their knowledge of previous solutions (Zowghi & Coulin, 2005)

Requirements change as the project proceeds:

- Clients learn what is possible during the project (Briggs & Gruenbacher, 2002; Jain et al., 2003; Majchrzak et al., 2005; Robertson, 2001; Saiedian & Dale, 2000; Zowghi & Coulin, 2005)
- Business is essentially dynamic and so requirements change during the lifetime of the project (B. Boehm & Turner, 2004; A. M. Davis, Nurmuliani, Park, & Zowghi, 2008; Larman, 2004; Pitts & Browne, 2007)
- People change their mind about what they want (Tsumaki & Tamai, 2006)

Clients will sometimes ask for requirements that the organization does not need:

- The client asks for something that is not really needed (A. M. Davis et al., 2008; Tsumaki & Tamai, 2006)
- The client asks for something they are not committed to (Tsumaki & Tamai, 2006)

The client cannot say what the business needs:

- Some requirements are tacit. That is, understood by the client, but not stated by them as it forms part of their tacit knowledge. (Eva Hudlicka, 1999; Robertson, 2001; Tsumaki & Tamai, 2006; Zowghi & Coulin, 2005)
- Some clients only know about a single section of the business that needs to be fixed (Arthur & Groner, 2005)

Some clients do not want to help you with the project

- A client representative has interests that conflict with others in the project or with the aims of the project (Alvarez, 2002; Briggs & Gruenbacher, 2002; Davidson, 1996; Giorgini, Massacci, Mylopoulos, & Zannone, 2006; Zowghi & Coulin, 2005)
- Some people will use resistance tactics to avoid a conclusion to RE (Saiedian & Dale, 2000; Tsumaki & Tamai, 2006; Zowghi & Coulin, 2005)
- Clients see the new system as a part of power struggles in the organisation (Alvarez, 2002; Davidson, 1996; Zowghi & Coulin, 2005)

Some sources report that RE failed because it was not done properly (eg user input not allowed)

- no user input was obtained(Hansen et al., 2009)
- theory of RE was not used in practice (Hansen et al., 2009; H. J. Harris, 2006)

Symptoms that are not problems are often reported

Statements are often made using the words “sources of problem” when the description is of a symptom. eg (Hansen et al., 2009) p46 “the three leading sources of project difficulty – i.e., lack of user input, incomplete requirements, and changing specifications”

The most common of these symptoms reported is the outcome of RE being an incomplete or incorrect set of requirements. This can take a few forms:

- Incomplete requirements (S. W. Lee & Rine, 2004) (Arthur & Groner, 2005; G.J. Browne & Ramesh, 2002; Glenn J Browne & Rogich, 2001; H. J. Harris, 2006; S. W. Lee & Rine, 2004; Marakas & Elam, 1998; Tsumaki & Tamai, 2006; Zave, 1997; Zowghi & Coulin, 2005)
- Changing specifications (Hansen et al., 2009)
- Finished system does not include all requirements asked for (H. J. Harris, 2006; Marakas & Elam, 1998)
- Incorrect requirements (Marakas & Elam, 1998)

RE is not deterministic

In his PhD thesis Harris (H. J. Harris, 2006) found an "elliptical misalignment" between the theoretical and empirical worlds of RE. He maintains that this indicates that there are no causal connections between RE activities and eventual requirements. This type of thinking has been reflected in a number of post-modern approaches to studying RE. Examples include the case study of failure by Mitev (Mitev, 2000), the HSO process proposed by (Andreou, 2003),a framing model developed from social cognitive theory in a PhD by Davidson (Davidson, 1996) and the comparison of RE techniques by Coughlan and Macredie (J. Coughlan & Macredie, 2002) using socially oriented methodologies.

Conclusion

All of the papers studied that identified problems with RE were examined and 9 types of problem were identified that incorporate all those sources. The categorization is particularly useful as each problem category is the result of work done to find methods of addressing the problem type. A software engineer finding a problem type occurring can use the category to identify how to approach the job. An educator can use the categorization to structure lessons. This is important as there is some evidence that requirements elicitation is not emphasised in CS and IS courses. In particular there is evidence that graduates are especially weak at requirements gathering.