

# MEASURING REQUIREMENTS COMPLEXITY TO INCREASE THE PROBABILITY OF PROJECT SUCCESS

Holly Parsons-Hann, Kecheng Liu

*Informatics Research Centre, The University of Reading, Reading, Berkshire, RG6 6AY, UK*

Keywords: Project Management, Requirements complexity

Abstract: The widespread adoption of Information Technology has helped reduce market problems caused by geographical separation and allow collaboration between organisations who are physically distributed around the globe. However, despite the successful strategic benefits brought by the evolution of the Internet and other web based services, this has not led to a higher project success rate. The biggest reason for project failure is cited as 'incomplete requirements' which suggests that research must be done into the requirements analysis to solve this reoccurring problem. This paper aims to highlight and analyse the current work done in the software complexity and requirements engineering field and demonstrate how measuring requirements complexity will lead to less project failures.

## 1 INTRODUCTION

Due to changing demands and technological advancement, businesses respond by developing increasingly flexible infrastructures in order to remain competitive in the newly defined market boundaries. However, despite adapting their business practices to exploit newly discovered market niches, organisations are still finding it increasingly difficult to employ successful project management techniques. One of the most famous project software statistics is attributed to the Standish Group's 1995 CHAOS report which states that 31.1% of projects will be cancelled before they ever get completed (Chaos 1995).

Recent investigations have demonstrated that the situation has not improved in the last 10 years despite the release of various project management tools and methods. A survey conducted by Taylor in 2001 purports that out of 1027 software projects only 12.7% were considered successful (Taylor 2001). The reasons for project failure are widely documented and known, however there is no standard list of reasons due to the heterogeneity of the organisations in question. The CHAOS report lists the number one reason for inability to complete a project as 'incomplete requirements', stating that this is true in 13.1% of the organisations interviewed.

The structure of the paper introduces the current work performed in both the software complexity and requirements engineering area followed by the identified factors affecting the requirements complexity. The research is then presented with conclusions and future work at the end of the paper.

## 2 CURRENT WORK IN THE SOFTWARE COMPLEXITY AREA

Software complexity is a well known paradigm within the software engineering domain and one which boasts a rich supply of metrics claiming to be able to define and measure the 'complexity' of software. Within this area, different types of complexity have been identified and defined in an attempt to measure productivity, dependence, testing and maintenance effort.

Inheritance complexity is measured using the depth of inheritance tree, whereas standalone complexity is measured by the weighted methods per class metric (Chidamber 1994). Interface complexity. Cyclomatic complexity (McCabe 1994) is the most widely used and recognised software metric which may be considered a broad measure of the

soundness and confidence for a program. It measures the number of linearly-independent paths through a program module is often used on concert with other software metrics.

One of the problems with measuring complexity and using the metrics described above is the lack of a working definition with intuitive notions and assumptions leading to contradictions thus confusion. In this paper, we are using the Oxford English Dictionary's definition of the word 'complex' which states that

Complex (adj)

1. *Consisting of many different and connected parts*
2. *Not easy to analyse; complicated or intricate*

From this definition we have attempted to define all the different parts e.g. cost drivers which will have an effect on the complexity of requirements.

### 3 REQUIREMENTS ENGINEERING

One of the biggest problems with requirements engineering is that the stakeholders may be numerous and distributed with varying and conflicting goals. Eliciting stakeholders requirements is a challenge in itself (Holtzblatt *et al.* 1995) (Goguen *et al.* 1993) however once collected there is the task of articulating the requirements to be considered. Once collected, the requirements must be expressed into a common language understandable to both the project team and the stakeholders.

By adopting new requirement engineering techniques, it is hoped that stakeholders requirements will be better understood and modelled and the cultural behaviours of stakeholders better understood (Liu 2000). Therefore we propose a complexity measure that will take the natural language requirements as input and associate a value of complexity to each one. The greater the value, the harder project success will be to obtain and therefore more caution must be exercised when implementing such projects.

### 4 FACTORS AFFECTING COMPLEXITY

The concept of complexity is in itself a 'complex' notion. There are many factors which contribute to the level of complexity and make each requirement a more or less complex objective. We have identified the cost drivers we feel are responsible for affecting the complexity of requirements and attempt to define each one below.

#### *Time spent on the project*

Estimating how many hours a task can take an employee is not an easy job. There are many factors to take into account such as the skills and experience of the employee, the resources available and the difficulty of the task in question. If the task is relatively small, fairly accurate estimations can be made, for example 'How long will it take for employee X to write a 3 page report on requirements bearing in mind all the stakeholders are in the same office as that employee' One could estimate that the task would not take more than a day, therefore 7/8 hours and be accurate to within 1 or 2 hours.

On the other hand estimating how long it will take an employee to build a modular component for an Information System when a lot of factors are undetermined is a lot harder. However due to project deadlines an estimation still has to be made. Should the project manager/ project team be experienced in the building of such systems, they may already have some knowledge as to how long such a module usually takes based on past experiences and the types of problems associated with the task. This experience is invaluable as it can sometimes bring very accurate estimations using limited data available.

On the other hand, each project is an individual and therefore what went right for one project will not necessarily go smoothly for the next project, thus basing estimations on past experience is not always the best idea. Therefore an equation needs to be used to estimate the number of hours spent on a particular task as accurately as possible.

#### *Heterogeneity of the Organisation*

Due to the IT sector being a highly fragmented and competitive industry, organisations must be both aggressive and flexible yet still maintain a good communication infrastructure. This can prove challenging when forming partnerships with

geographically dispersed organisations as chances are both business and cultural norms will differ greatly. This problem is magnified if we imagine one company to be in the UK and the other to be in America as due to the difference in time zones the working week is significantly shortened.

Research has been done into a number of 'capability barriers' which prevent effective communication in geographically dispersed groups (Toomey *et al.* 1998). The three identified problems included not sharing a common first language, being separated by sixteen time zones and the difference in typing ability when communicating via a messaging program.

#### *Skill of project members*

According to Wrigley and Dexter (Wrigley *et al.* 1987), expert judgement is still the most dominant method of estimation. This is supported by Vicinanza *et al.* (Vicinanza *et al.* 1991) who concludes that experienced managers can make more accurate estimates than existing uncalibrated algorithmic models, specifically COCOMO or function point analysis. Simon stated that 'Every manager needs to be able to respond to situations rapidly, a skill that requires the cultivation of intuition and judgement over many years of experience and training' (Simon 1987).

This is a widely accepted comment in industry who realise that good managers have both business acumen and business experience which has trained them to react to the changing market demand. However, very few project teams will just consist of experienced high ranking employees. Most projects will have an eclectic range of skilled employees ranging usually from new graduates to the project manager who will have a number of years experience dealing with similar projects.

#### *Number and location of stakeholders*

A stakeholder is any individual or group which can affect or is affected by an organisation's activities, employees are a key group. Individually they can enhance sustainability at the company they work for by bringing their personal skills and experiences to aid change and innovation.

When conducting a project for a Virtual Organisation there is a strong likelihood that some of the stakeholders will not be located in one geographic site. This means trying to elicit the requirements from numerous different stakeholders

from many different locations.

#### *Project Resources*

A resource can be defined as 'something that is used for support or help', therefore we can understand a project resource to be 'a means available to the company to help meet its' objectives'. Money, employees and skills are all examples of project resources as they help the project to be successful and meet all the objectives on time and accurately. However, as stated by the CHAOS report the third biggest reason for project failure was due to lack of resources which 10.6% of all the companies interviewed stated.

By ascertaining and measuring the amount of resources available to the project at the inception stage of the lifecycle we believe it will contribute to the overall requirements complexity value.

## 5 OUR RESEARCH

From the factors identified above, measurements have been created which, when finished will attribute a numerical value to each requirements in terms of complexity, the lower the value, the less complex the requirement is seen to be. When creating a metric to measure the time spent to fulfil the requirement, we assume the following statements are true:

1. The larger the estimation the less accurate experienced guesses will be;
2. The earlier the estimate is made, the greater the estimating errors (Jorgensen 2002).

Logarithms have been chosen due to their ability to be able to analyse exponential processes. As the log function is the inverse of the exponential function it is possible to measure and model many natural processes such as the acidity of a chemical solution or earthquake intensity on the Richter scale. As historical data is currently regarded as the best source of estimating projects (Boehm 2000) the data used must be accurate and complete (Fairley 2002).

Due to this increasing error risk, the metric created has been on a logarithmic scale as previous research has shown range metrics to be more accurate (Jorgensen 2002) (Fairley 2002). Figure 1.

illustrates the metric created with a numeric value associated to each range.

Number of hours					
0	0-8	8- 40	41-160	161-320	321-480
0	1	2	3	4	5

Complexity Number

Figure 1: Hour complexity metric

As most Project Managers prefer short incremental steps in development, 480 hours e.g. 3 months was chosen as the maximum number of hours allowed to be estimated. Should the project team feel that a requirement will take longer than 3 months to fulfil, the value 5 can be attributed to that requirement and should be flagged as a possible ‘high risk’ requirement.

Property	0	1
Time zones the company crosses	0-8 Time zones	8 + Time zones
Native language of all stakeholders	The same	Different
Industry domain both organisations are in	The same	Different
Technology used for communication	The same	Different
Business norms	The same	Different

Figure 2: Stakeholder metric

As the heterogeneity of the organisation can also contribute to the complexity of the requirements due to ongoing liaisons with stakeholders, it is important to acknowledge this fact and use a metric to gauge how complex the requirements are likely to be. Figure 2. gives a clear guide as to which factors are likely to affect the requirements complexity and the number which should be attributed to the requirement after considering each factor. When assessing each factor, the project team should add the score in each row together so that they get a value between the ranges of 0 and 5.

Skill	Value
Highly competent employee, been with company for a long length of time, previous experience in similar projects	1
Competent employee, been with company for length of time, previous experience in similar projects	2
Employee who has been with the company a short length of time, has experience in 1-2 similar projects	3
New employee, experience in industry but not in projects	4
New employee, no experience in industry or projects	5

Figure 3: Employee skill metric.

Skill of the project members has also been identified as a possible factor affecting the complexity of user requirements. Although the skill of members could be measured in a number of very complex ways, only one factor has been highlighted as being important: Has the project member worked on a similar project before, and if so how many similar projects has he/she been part of?

Figure 3. illustrates the skill complexity metric and attributes a value to each project member out of 5. Once all project members have been assessed each value should be added up and divided by the number of project members that are present. For each metric a value out of 5 will be elicited. For each metric used these numbers should be added together and divided by the number of metrics used. For example if the hour and employee skill metrics were the only ones used, the value from both should be added together and the answer divided by two to gain a requirements complexity value.

## 6 CONCLUSIONS & FUTURE WORK

It is clear that requirements complexity contributes to project failure in organisations, what is not apparent is to what degree this statement holds true.

Future work will allow the creation of an effort metric, a metric which takes into account the number and location of stakeholders and the amount of project resources available. Once the metrics are completed they will then be validated by being used in a number of different organisations and the results published. It is hoped that by

measuring the complexity of user requirements it should be possible to have a better understanding of the issues which may cause problems later on in the project lifecycle. By identifying these problems at the first possible opportunity they can be dealt with leaving more project opportunities available than if dealt with at a later date.

expert performance, In *Information Systems Research* Vol 2. No. 4, pp 243-262  
 Wrigley, C., D., Dexter, A., S., 1987. Software development estimation models: A review critique, In *ASAC'87, Administrative Sciences Association of Canada Conference*, pp 125-138

## REFERENCES

- Boehm, Barry W., 2000. *Software Cost Estimation with COCOMO II*. London: Prentice-Hall.
- Chidamber, S, R, Kemerer, C, F, 1994. A Metrics Suite for Object-Oriented Design, In *IEEE Transactions on Software Engineering*, Vol. 20, No. 6, pp. 476-493
- Delany, S., J., Cunningham, P., Wilke, W., 1998. The limits of CBR in Software Project Estimation, German Workshop on Case-Based Reasoning, <http://www.cs.tcd.ie/publications/tech-reports/reports.99/TCD-CS-1999-21.pdf>
- Fairley, Dick, 2002. Making Accurate Estimates. In *IEEE Software*, Vol 19 No. 6 pp 61-63.
- Goguen, J., Linde, C., 1993. Techniques for Requirements Elicitation. *1st IEEE International Symposium on Requirements Engineering* pp. 152-164.
- Holtzblatt, K., Beyer, H. R. 1995. Requirements Gathering: The Human Factor. *Communications of the ACM*, Vol 38, No 5, pp 31-32.
- IT Cortex, 2004. Failure Causes – Statistics, [http://www.it-cortex.com/Stat\\_Failure\\_Cause.htm](http://www.it-cortex.com/Stat_Failure_Cause.htm)
- Jorgensen, M., 2002. A review of Studies on Expert Estimation on Software Development Effort, Simula Research Laboratory, Norway
- Liu, Kecheng., 2000. *Semiotics in Informations Systems Engineering*, Cambridge University Press
- McCabe, Thomas J. & Watson, Arthur H, 1994, Software Complexity. *Crosstalk, Journal of Defense Software Engineering*, Vol 7, No.12
- Niessink, F., van Vliet, H., 1998. Two Case Studies in Measuring Software Maintenance Effort, In *Proceedings of ICSW*, pp 76-85
- Simon, H., A., 1987. Making management decisions: The role of intuition and emotion, In *Acad. Management Exec* Vol 1. pp 57-63
- Standish Group 1995 CHAOS report
- Taylor, A., 2001. IT Projects sink or swim, BCS review, <http://www.bcs.org/review/2001/articles/itservices/projects.htm>
- <http://www.doc.ic.ac.uk/~ban/pubs/sotar.re.pdf>
- Toomey, L., Smoliar, S., Adams, L, 1998. Trans-Pacific Meetings in a Virtual Space, FX Palo Alto Labs Technical Reports
- Vicinanza, S., T., Mukhopadhyay, T., Prietula, M., 1991. Software effort estimation: an exploratory study of