



Defining a Requirements Process Improvement Model

SARAH BEECHAM
TRACY HALL
AUSTEN RAINER

s.beecham@herts.ac.uk
t.hall@herts.ac.uk
a.w.rainer@herts.ac.uk

Department of Computer Science, University of Hertfordshire, College Lane, Hatfield AL10 9AB, UK

Abstract. Both software organisations and the academic community are aware that the requirements phase of software development is in need of further support. We address this problem by creating a specialised Requirements Capability Maturity Model (R-CMM¹). The model focuses on the requirements engineering process as defined within the established Software Engineering Institute's (SEI's) software process improvement framework. Our empirical work with software practitioners is a primary motivation for creating this requirements engineering process improvement model. Although all organisations in our study were involved in software process improvement (SPI), they all showed a lack of control over many requirement engineering activities.

This paper describes how the requirements engineering (RE) process is decomposed and prioritised in accordance with maturity goals set by the SEI's Software Capability Maturity Model (SW CMM). Our R-CMM builds on the SEI's framework by identifying and defining recommended RE sub-processes that meet maturity goals. This new focus will help practitioners to define their RE process with a view to setting realistic goals for improvement.

Keywords: requirements engineering (RE), requirements engineering process, software process improvement, RE process improvement, Software Capability Maturity Model (SW CMM), goal-question-metric

1. Introduction

According to the software engineering literature requirements engineering (RE) is a very important part of software development, yet getting requirements right continues to be a universal problem. Requirements errors can be costly in terms of lost time, lost revenue, loss of reputation and even survival. Solutions are proposed to help practitioners with their technical and organisational RE problems. Various models have been developed to guide organisations towards optimising their software processes and instruments are designed to measure process strengths. The software industry accepts they need help and the research community is endeavouring to support them. With this plethora of information, however, it is difficult for the practitioner to know where to start to look for help.

In this paper we explain the main stages involved in developing a model that guides practitioners to understand their own RE process. We aim to support the practitioner by providing guidelines for RE process improvement within a familiar framework. Our Requirements Capability Maturity Model (R-CMM) builds on the established SEI Software Capability Maturity Model (SW CMM) (Paulk et al., 1995) improvement framework. Relating the RE process to the SW CMM will enable practitioners to use our model in conjunction with on-going Software Process Improvement (SPI) activities. Our R-CMM can be used with the SW CMM or used independently to assess RE process capability.

Terms such as 'requirements', 'specification', 'requirements engineering' and 'RE' are often used in the literature to embrace the whole of the requirements 'process' (Lindland et al., 1994). The term 'requirements engineering process' or 'RE process' as used in this study, refers to activities performed in the requirements phase that culminate in producing a document containing the software requirements specification (Jalote, 1997). More specifically, the RE process is the set of activities required to gather, specify, validate and engineer a set of requirements (Thayer and Dorfman, 1990); whereas 'a requirement' is defined as "a feature or behaviour of the system that is desired by one or more stakeholders" (Britton, 2000). This study focuses on the 'RE process' and not the individual feature or behaviour of the system. A glossary of terms used in this paper is given in Appendix A.

1.1. Rationale for building a model of RE based on the SW CMM

Our empirical work with twelve software development companies involved in SPI provided the initial motivation for developing the R-CMM (Hall et al., 2002a and 2002b; Beecham et al., 2003c). Forty-five focus groups were held to explore general software development problems experienced by groups of developers, project managers and senior managers within a SPI environment. All groups reported a general dissatisfaction with the way their RE process was managed where problems ranged from poor communication with stakeholders, to a lack of resource. Analysis of the RE problems voiced by our practitioners together with an analysis of the literature led us to explore the possibility of bringing together existing strategies for managing and controlling the RE process (Beecham et al., 2003a).

Basing the R-CMM on a known software improvement framework offers the user many advantages. The framework pulls together disparate work in the field of RE and presents solutions in a way that is accessible to both practitioners and researchers. The R-CMM includes an assessment method that guides the user to understand their current RE process with a view to prioritizing process implementation against maturity goals. The rationale for basing our model on the SW CMM is that it:

- Contains guidelines for many RE-related activities
- Is based on best practice derived from many years of empirical study
- Has a limited set of activities
- Is a known standard
- Has a proven record of achievement
- Is designed to be tailored to focus on specific process areas
- Continues to be supported by the SEI
- Has a maturity structure to help with process prioritisation
- Is goal focussed
- Integrates RE processes with software development

The R-CMM taps into the strengths of the SW CMM to form a specialised best practice model that is familiar, integrates with related software processes, and has a tried and tested methodology. The R-CMM therefore helps practitioners to identify where their priorities lie and guides them towards solving their RE problems from within a software development context.

Our work differentiates itself from other RE models such as Sommerville and Sawyer's good practice guide (1997) as it aligns itself with the SW CMM rather than developing a new maturity structure. Our R-CMM builds on prior work by bringing together proven and familiar methodologies. Humphrey emphasises the need for the research community to follow this strategy by exploiting existing solutions:

“Every time software people have faced a new problem, instead of building on prior work, we have invented some new language, tool, or method. This forces us to start over, and it also forces our users to start over.”
(Humphrey, 2002)

Adapting the SW CMM to focus on a different area in need of improvement is not a new concept. For example its maturity improvement framework has been used in the field of strategic planning, in the IT Service industry and in testing software (Burnstein et al., 1996; Christie, 1999; Neissink et al., 2002). Also, the SEI continues to develop and supplement the CMM, its most recent release being the Capability Maturity Model Integration (CMMI, 2001).

While these related models prove the versatility of the SW CMM, the accompanying literature does not detail the rationale for adapting the CMM concept to new environments. We therefore extend the work on model adaptation by explaining how and why the SW CMM is used to create a firm foundation on which to place the RE process and recommended best practices. The R-CMM is a suite of models that takes practitioners from a high level view of the process, through to a detailed guideline and finally to a process assessment method to guide companies towards satisfying particular company goals.

This paper is organised as follows: In Section two we explain our motivation for building the R-CMM. Section three gives a background to how we use existing techniques to build a framework and provides a rationale for using the SEI's SW CMM to mould the R-CMM. In section four we introduce the R-CMM and outline how the RE process is decomposed into 5 maturity levels. Section five explains how the R-CMM retains a goal focus through an adaptation of the Goal Question Metric paradigm. Section six presents the Level 2 RE process improvement model at differing levels of abstraction. We summarise and conclude in Section seven.

2. Empirical study of SPI problems in software development

The primary motivation for building the R-CMM emanates from our previous empirical research with 12 software development companies (Hall et al., 2002b; Beecham et al., 2003c, 2004). Our research highlighted problem areas in software development that led to a detailed study of the problems practitioners were experiencing in RE (Hall et al., 2002a). The study examined the first four SW CMM levels. One of our most significant findings agrees with Paulk et al. (1995) that, “although software engineers and managers often know their problems in great detail, they may disagree on which improvements are most important”. A primary aim of the RE model, therefore, is to help organisations agree on a strategy for improvement and achieve a consensus on how to implement RE related improvement activities.

Although the companies in our empirical study varied in size and application area, they were all using the SW CMM to guide them in their software process improvement activities. As discussed fully in Hall et al. (2002a, 2002b); and Beecham et al. (2003c), the companies in our study included nine multi-national companies, as well as some small and medium sized software development companies. We interviewed three different staff groups (developers, project managers and senior managers) in 45 focus group sessions and discovered a general enthusiasm for the CMM. A comment from a senior manager shows the wider benefits of implementing a CMM improvement method, "it should help people have a stronger sense of being professional and working for a first class company and should help towards retaining staff and reducing costs". While a project manager takes a more pragmatic view stating that "[the CMM] helps you to control your destiny".

When asked about general problems these practitioners were having with their software development a common theme throughout all focus groups related to RE. For example a project manager states, "I don't believe that we spend enough time up front of the project doing all the work, understanding exactly what we need to do and consequently we learn as we go through and have to keep changing the requirements". Another quote given by a developer clearly shows a frustration with the lack of control over inevitable changes in requirements, stating: "We get changes in requirements during development which add extra resource factors onto our jobs but that is not taken into account. It is not factored into our time scales. It is the biggest problem for me at the moment". These RE problems were common throughout the 4 levels of SW CMM maturity represented in the focus groups.

Taking a closer look at the type of RE problems practitioners were experiencing revealed two distinct categories: Organisational and Technical RE problems as shown in Table 1.

Dividing the number of RE problems reported into these two categories reveals dichotomous behaviour as companies mature through from Level 1 to Level 4. Figure 1 is a normalised representation of 364 organisational and technical RE problems reported across the 4 maturity groups. Here, the companies in our study *appear* to increase their control over RE problems as their CMM capability matures. This is shown by the gradual drop in total number of reported problems in the scale from maturity level 1 to 4 (shown by the striped bar). Indeed, the technical process seems to reflect this with the drop in reported problems in defined capability Levels 2–4. However, organisational problems reported remain fairly constant over the three defined levels (Levels 2–4) as shown by the black bar in Figure 1. This suggests a lack of ability to manage organisational RE problems despite increased process capability. This weakness would be masked in an analysis of all RE problems.

Table 1. RE process problems.

Organisational	Technical
Poor stakeholder communication	Complexity of application
Staff retention	Poor user understanding
Culture resistant to change	Requirements growth
Poor resource allocation	Requirements traceability
A lack of training	Undefined RE process
Lack of skills	Vague requirements

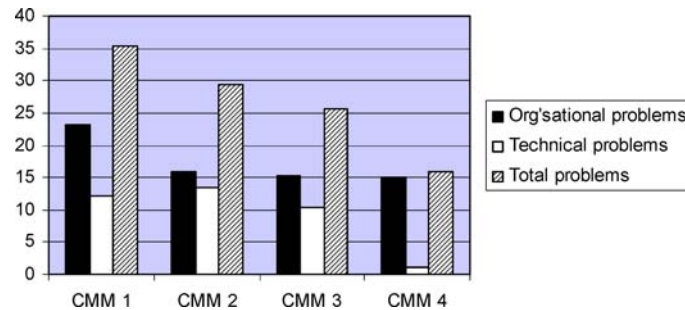


Figure 1. RE problems by SW CMM level (using normalised data).

This analysis suggests a need to help practitioners manage organisational activities along with technical aspects of the RE process. Although the SW CMM is rich in references to organisational best practices, the messages do not appear to be reaching the practitioners in our study. It could be that the SW CMM presents a confused message for process improvement, for example Ngwenyama and Neilsen (2003) contend that the SW CMM makes contradictory assumptions about organisational culture that will cause difficulties to implementation teams. The importance of supporting and controlling organisational sub-processes along with technical sub-processes is highlighted in SPI research (Lubars et al., 1993; Perry et al., 1994). Furthermore, organisational aspects are often overshadowed by the many research efforts aimed at developing technology support (Rossi, 1999). The R-CMM redresses this imbalance by including both organisational and technical processes within its improvement paradigm. The practitioner can thereby gain a more holistic picture of the key activities involved in RE prior to setting goals for improvement.

Our empirical research led us to conclude that the SW CMM in its current form is not helping practitioners to:

- (a) identify and define both technical and organisational aspects of the RE process
- (b) recognise RE process problems
- (c) assess and agree RE improvement priorities
- (d) relate RE process problems to RE improvement goals
- (e) relate RE improvement goals to the general software CMM guidelines and activities

In order to strengthen these aspects of the SW CMM, our specialised process improvement model isolates the RE process and assists practitioners to identify and prioritise their problems. It takes the advice given by Paulk et al. (1995) and guides practitioners to focus on “a limited set of activities” and “work aggressively to achieve their goals”. A primary motivation for building the R-CMM therefore, is to ensure that RE process needs are identified and included in company goals.

3. Augmenting the SW CMM—Learning from previous work

In this section we explain how SW CMM characteristics are used to create a framework for defining the RE process. We take the advice of Wiegers (1998b) and Humphrey (2002)

and guide practitioners to apply techniques defined by existing models and frameworks in a routine and effective way. Wiegers adds that once the practical limits of known approaches have been reached, we can turn to improved models that provide guidance for working in better ways. Therefore, as the current SW CMM approach to improvement seems to be “necessary but not sufficient . . . and does not address many crucial processes or areas of activity” (Rogoway, 1998), we create an augmented, specialised CMM to fill this gap.

3.1. *Reasons for using the SW CMM*

The SW CMM has been criticised as being too descriptive, as missing many important activities and setting incorrect priorities, e.g. (Brodman and Johnson, 1994; Hayes and Zubrow, 1995; Sommerville and Sawyer, 1997; Lauesen and Vinter, 2001; Hall et al., 2002a). We appreciate that the SW CMM is not a perfect model of software process assessment and improvement, as Paulk et al. (1995) admit, “all models are wrong; some are useful. Models are simplifications of the real world they represent and the CMM is not an exhaustive description of the software process”. Despite this failing, there are many compelling reasons in favour of using SW CMM concepts as a basis for creating our specialised R-CMM.

Firstly, a pragmatic reason for using the SW CMM is that it is the most used model for judging the maturity of software processes (El Emam and Madhavji, 1995b) and (Ngwenyama and Nielsen, 2003). According to Rogoway (1998) it has become a *de facto* standard for assessing and improving processes. Thousands of users have made significant improvements in product quality, productivity and cycle time as a result of using the SW CMM together with the SW CMM assessment. In building the specialised R-CMM we found that many of the processes required to meet users’ RE needs are indeed embedded in the SW CMM.

Secondly, the SW CMM is designed to be tailored and adapted to meet specific needs as it is a normative model (Paulk et al., 1995). We know that augmentation of the SW CMM is possible, as its framework has been adapted both inside and outside the field of Software Engineering. There are reportedly 34 CMMs developed by different groups (Reifer, 2000). Examples of model adaptation inside software engineering include Hackos (1997), who adapted the CMM to create a tool for strategic planning and Christie (1999), who created a CMM simulation to support process measurement. An example of CMM adaptation outside software engineering is in the IT service industry where CMM framework is used to help improve processes (Neissink et al., 2002).

Thirdly, the SW CMM is a ‘living model’ and is actively supported by the SEI. It is continually undergoing changes as the SEI recognises the complex needs of the software industry. For example, the SW CMM has been supplemented with paradigms such as the IDEAL improvement model (McFeeley, 1996), the People Capability Maturity Model (Curtis et al., 1995), Personal Software Process (Humphrey, 1997) and Team Software Process (Humphrey, 2000). The latest development is the Capability Maturity Model Integration (CMMI, 2001) that attempts to bring the different SEI process assessment and guideline models together under one meta-model. The concept is to allow users to generate combinations of CMMs of interest to them (Reifer, 2000). This development suggests that the model itself evolves to reflect the current needs of users.

Finally, an empirical reason for using the SW CMM is that our previous studies base their findings on the experiences of practitioners who used the SW CMM as a standard for assessing and improving software processes (Hall et al., 2002b; Beecham et al., 2003c). It is of direct relevance therefore to use this empirical data to create a model that addresses the problems raised by practitioners in their improvement activities. Also, as practitioners continue to use the SW CMM they should be given the opportunity to implement proven methods within a familiar framework (Humphrey, 2002) and (Wieggers, 1998b).

3.2. *Defining the R-CMM*

Defining processes is recognized as a critical element in software process improvement (Christie, 1999) yet to be useful a model must be a clear simplification of the complex world it is modelling (David, 2000). To keep the presentation clear and useable, the R-CMM links the RE process to maturity levels, but is not an exhaustive representation of the RE process. The RE process is broken down into key sub-processes. Each sub-process is defined and assessed. To ensure that all users of the model interpret the terms used in the same way, separate documentation is available that includes a glossary of terms and detailed definitions of each RE sub-process (Beecham et al., 2003b).

The R-CMM is designed to help practitioners strengthen their RE process by implementing sub-processes (or best practices) in a logical order. Figure 2 introduces the R-CMM and places it in context with the SW CMM. This high level view of the model shows how the RE process matures from an ad-hoc undefined level to a continuously improving level. The model also shows how each R-CMM level has a pre-defined goal to help companies focus on their improvement activities. Figure 2 characterises the capability of the RE process at five levels of maturity, where for example a disciplined and structured RE process has a level 2 capability.

Basing the R-CMM on a SW CMM framework also means that any company already using the SW CMM should find it easy to apply and use. However, the R-CMM can be also used independently of the SW CMM to assess the strength of the RE process. Therefore, the R-CMM has a broad application.

4. **Transposing SW CMM characteristics into a R-CMM**

This section presents an overview of how each of the 5 levels of maturity introduced in Figure 2 characterises different RE process capability. The sub-sections detailing maturity levels 2–5 show how SW CMM maturity characteristics act as initial RE process improvement goals. (As in the CMM, level 1 companies do not have any defined goals). The subsequent sections also indicate how our empirical findings from the focus groups influence our model design (Hall et al., 2002b; Beecham et al., 2003c, 2004). This section does not include guidelines for improvement; it focuses on the process capability and problems associated with each maturity level.

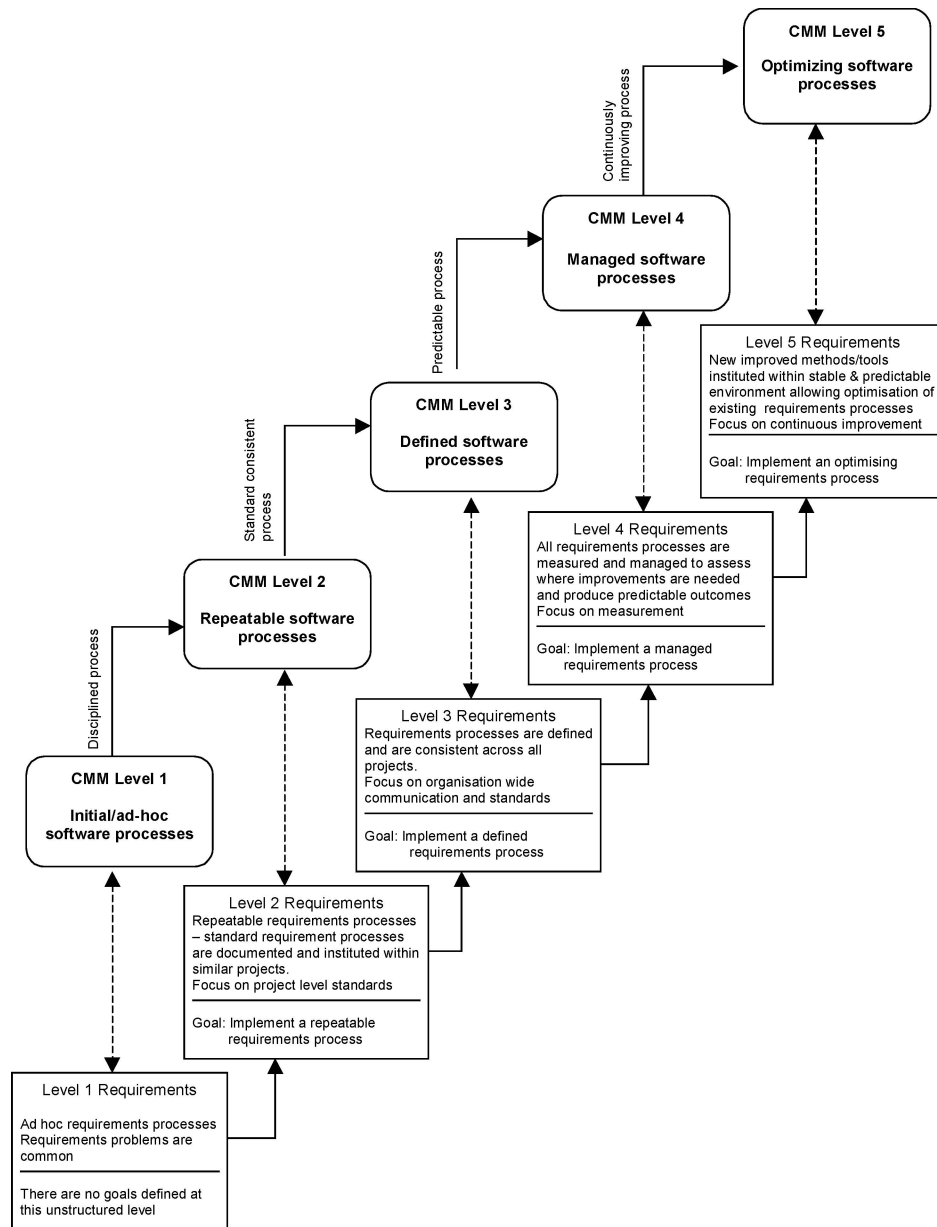
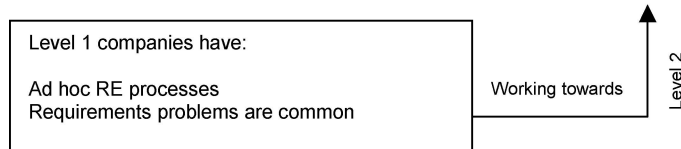


Figure 2. The R-CMM 5 level framework.

4.1. R-CMM Level 1

There are no process improvement goals defined at this unstructured level.



(Figure 2 Segment)

Our empirical study consisted mainly of Level 1 companies, reflecting the state of software development in general where an estimated 70% of companies remain at this undisciplined level of capability (Paulk and Chrissis, 2000, 2002). Within our study, RE problems were found to be common, with Level 1 companies reporting more problems in this area than any other maturity group. Their main problems relate to vague requirements, lack of traceability, undefined RE process, insufficient RE resource, lack of training and poor levels of skill.

Examples of comments recorded from Level 1 practitioners within these categories include:

- “Requirements are very vague . . . Interpretation of requirements is very difficult. There are gaps in them. Not detailed enough . . .”(vague requirements)
- “how do we control the changes that go in? . . .”(undefined RE process)
- “keeping track of requirements over that period of time . . . is a major issue” and “We should be able to trace any requirements via the register, but critical success factors are not good” (lack of traceability)
- “something that if you had the resources to hand would have taken a day takes 3 days as you have to wait until you are both free to speak.” “ . . . we haven’t necessarily put the resources in to setting some of those processes and systems into place.” (insufficient RE resource)
- “[We need to] get people on training re: requirements . . . but very difficult to get training implemented.”(lack of training)
- “Not enough skilled business analysts . . . impacting requirements” and “a lot of our experts are disappearing and a lot of history is going with the projects”. (poor levels of skill)

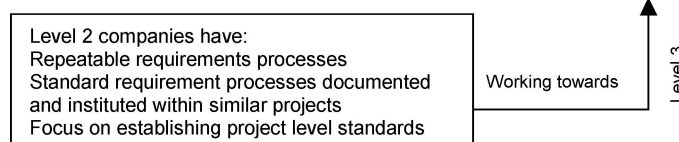
A characteristic of Level 1 companies is that they have not defined their RE process. These companies operate in their own unique way and depend on people rather than process. Paulk et al. (1995) describe success at this level as depending on “the competence and heroics of the people in the organization and cannot be repeated unless the same individuals are assigned to the next project”. However, a general ‘improvement’ goal for a company with ad hoc processes is to mature to Level 2 where their processes become repeatable.

Recommendations: Level 1 organisations need to work towards developing a disciplined process and need to raise their awareness of their RE problems. Examining the R-CMM

sub-processes at Level 2 should help managers gain an insight into their RE process and encourage them to buy into the idea of software process improvement (Beecham et al., 2005). It is likely that managers at this level of maturity will need to prioritise their RE problems. By definition, this ‘ad-hoc’ level has no associated ‘best practices’. It is at Level 2 that the R-CMM addresses the needs of the Level 1 companies. To progress to Level 2 requires that organisations examine their RE process in detail.

4.2. R-CMM Level 2

Level 2 Goal: To implement a repeatable RE process



(Figure 2 Segment)

Our empirical research showed that organisations with a Level 2 capability experience fewer technical problems with their RE process than their Level 1 counterparts. While this suggests that the SW CMM strategies they are following are working to an extent, our analysis revealed that their organisational problems did not ease off. Problems with communication remain a major problem along with staff retention. For example, our study showed that technical difficulty for Level 2 companies centred on complex requirements, requirements growth and an undefined RE process. Definitions of all these problem groups are given in the glossary in Appendix A. When building our model we include sub-processes to address these areas of concern at this level of maturity.

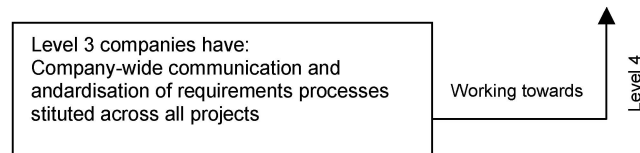
Companies at this ‘repeatable’ Level 2 process capability have established basic project management sub-processes to track cost, schedule, and functionality. The necessary process discipline is in place to repeat earlier successes on projects with similar applications (Paulk et al., 1995).

The R-CMM at Level 2 maturity aims to help managers to identify and document the RE process by learning from previous project successes and failures. This is achieved by an assessment of current processes (assessment, or ‘metrics’ is a part of the R-CMM and is explained in Section 5). Examining their current sub-processes and learning where weaknesses lie will help managers gain a general overview and allow them to address RE issues associated with individual projects.

RE management is a Level 2 key process area (KPA) in the SW CMM. The R-CMM reflects this by creating a baseline model of the RE process that is built on as a company matures. Therefore, ‘baseline model’ refers to the first defined model built that provides a foundation for future process improvement. In the spirit of continuous improvement, a Level 2 compliant organisation should be working towards creating a standard and consistent organisation-wide RE process at Level 3 maturity. We include a model of the Level 2 R-CMM including all its twenty sub-processes in section 6.

4.3. R-CMM Level 3

Level 3 Goal: To implement a defined RE process



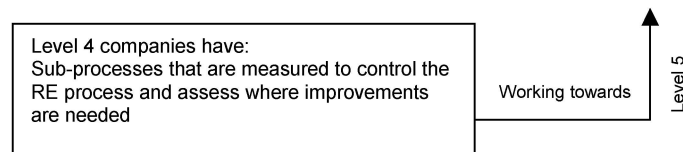
(Figure 2 Segment)

Similar to Level 2 companies, our empirical research showed that Level 3 companies had increased control over their technical RE problems, but saw little improvement in managing their organisational processes. They continued to report problems relating to the support of the RE process. Level 3 companies are most concerned with user understanding of requirements, internal communication and external communication. Examples of these issues as extracted from the focus group data are given in the glossary in Appendix A. When building the R-CMM, the particular issues raised by practitioners in our focus groups, are addressed along with guidelines at Level 3 process capability.

Level 3 R-CMM co-ordinates the RE sub-processes that were established at Level 2. The focus shifts from project based sub-processes towards creating company-wide, organisational standards and visibility. All projects now use a documented and approved version of the organisation's [RE] process (Paulk et al., 1995, p. 193). With these sub-processes in place, management has an increased ability to see and control RE activities which may previously have been viewed as black boxes.

4.4. R-CMM Level 4

Level 4 Goal: To implement a managed RE process



(Figure 2 Segment)

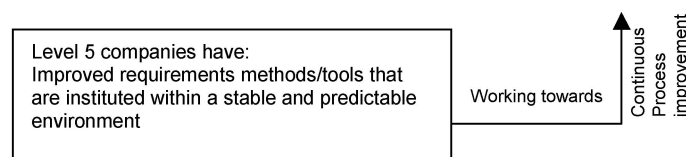
Our analysis of Level 4 RE needs is drawn from a small sample of three focus groups within one company. We therefore use the results to suggest RE sub-processes that may need support. For example, the trend to manage technical requirement problems with increased maturity continues as no technical RE problems were reported at this level. However, organisational problems remain a problem, despite the general increase in process capability. (In the glossary in Appendix A, we give examples of organisational problems).

Companies at this ‘managed’ Level 4 maturity are in a position to collect detailed measures of the RE process and product quality. Both the RE process and products (in terms of defects for example) are quantitatively understood and controlled using detailed measurements (Paulk et al., 1995).

At this level of maturity, the R-CMM is guided primarily by the SW CMM. We do not have sufficient data from our empirical study to support introducing new processes and therefore rely on the SW CMM to specify activities that focus on RE. The R-CMM reflects the SW CMM focus on measurement at Level 4 RE process maturity by introducing quantitative RE quality goals. Managers are also more able to assess the risks of modifying their processes or integrating new process elements (Christie, 1999). Examples of measurement data include: effectiveness of RE training; and number and severity of defects found in the software requirements (Paulk et al., 1995).

4.5. R-CMM Level 5

Level 5 Goal: To implement an optimising RE process



(Figure 2 Segment)

Our empirical study did not include an organisation with a Level 5 software process capability, which is not surprising as there are relatively few companies in the world that have reached this level of maturity (SEI, 2002). We therefore look to the SW CMM for maturity characteristics and refer to the RE literature for complimentary best practices. The R-CMM at this high level maturity is therefore a distillation of RE features from the SW CMM and the literature.

Companies at this ‘optimizing’ level continually improve their processes through quantitative feedback from the process and from testing innovative ideas and technologies (Paulk et al., 1995). Companies moving up from Level 4 to Level 5 should have a wealth of metric data to manage the course of a process (Christie, 1999). This creates an environment where sub-processes can be confidently modified. New methods to improve the RE process can be continually tried in a controlled manner.

5. The R-CMM goal question process metric focus

In this section we explain how the R-CMM adapts the Goal Question Metric (GQM) paradigm (Basili and Rombach 1988) to include a ‘process’ element. Figure 3 shows our goal centred approach to RE process improvement. We have added a ‘process’ element to the GQM as defining sub-processes within a maturity framework is the essence of the R-CMM. The process element directly addresses the needs of a business as defined in

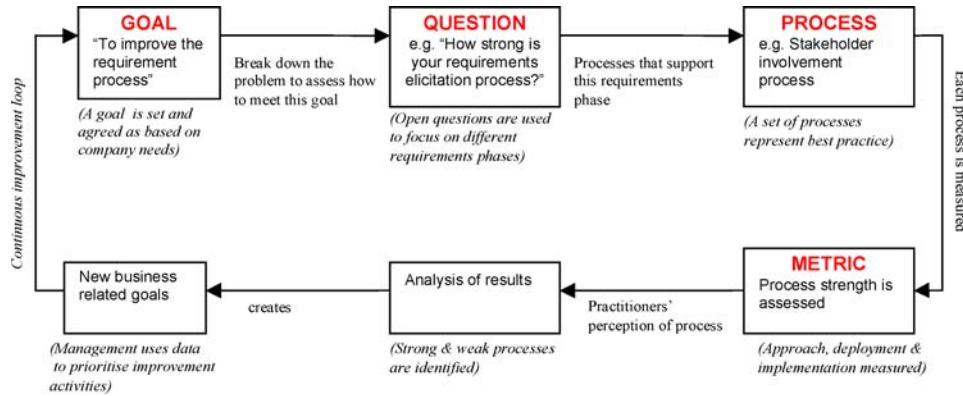


Figure 3. Example of continuous process improvement through a Goal Question Process Metric paradigm.

the initial ‘goal’ element. For example, the continuous improvement approach to process improvement shown in Figure 3 gives visibility into what is required to improve RE quality (e.g. improved traceability) and can combine these technical sub-processes with those that relate to organisational sub-processes such as resourcing, time-scales and cost that are equally important to business (Solingen and Berghout, 1999).

Figure 3 gives an example of how the organisation sets an improvement goal and how this goal is decomposed through a series of questions that relate to given processes. Figure 3 also shows how the R-CMM supports continuous improvement as advocated by Deming (1982) and Humphrey (1989).

The next four sections discuss each of the main elements in Figure 3.

5.1. R-CMM goals

Organisations need to define their improvement goals otherwise the improvement activities will turn out to be as chaotic as the development process itself (Solingen and Berghout, 1999). The R-CMM differentiates between the high level goals in Figure 2 (e.g. “implement a repeatable RE process”) that provide a focus for each maturity level, and the individual ‘tailored’ goals featured in Figure 3. This distinction is made to ensure that the improvement effort is driven primarily by *individual* goals and not maturity level goals (Wiegers, 1998a). Research also shows that if a SPI program is focussed on current business needs and is understood and agreed by management it is more likely to survive in the long term (McFeeley, 1996). This goal focus is further advocated by Rifkin (2001) and Potter and Sakry (2001) in their work on software process improvement methods. A further benefit of this approach is that goals help companies to remember that their main purpose is “to develop software rather than processes” (Fayad, 1997). To meet these aims, the R-CMM guides practitioners towards identifying new business-related improvement goals based on an assessment of their current capabilities as shown in Figure 3.

5.2. *R-CMM questions*

Questions are purposefully designed to be 'open' to encourage users to investigate how to provide an answer to whether the goal is being met. Providing answers to these set questions will help managers to gauge whether progress is being made towards meeting the goal (Pfleeger, 1995). The questions relate to a defined process; a level of quality (e.g. repeatable, managed) and the answers to these questions should relate to this quality perspective (Basili and Rombach, 1988). To meet these aims each question in the R-CMM should be

- clearly defined
- directly related to the goal
- a guide to solutions provided by the processes

Output from answering these questions will be assessed and fed back into the model to define new goals. For example the question, "How repeatable is your elicitation process?" focuses on the Level 2 goal of creating a repeatable RE process and points to quantifiable processes. Questions also create a direct link to different phases in RE and begin to decompose the RE process. The RE phases modelled in the R-CMM questions mirror those found in (Dorfman and Thayer, 1997) and (Pressman, 2001). These phases are: management of the RE process, elicitation of requirements, analysis and negotiation of requirements, documentation of requirements and verification of requirements. The purpose of this phased view of RE is to help practitioners relate the complex RE process to best practices.

5.3. *Processes: The substance of the R-CMM*

The R-CMM 'process' dimension places the RE process in context with goals and questions at different levels of process maturity. The process capability of a company relates to a level of RE process maturity. Each sub-process represents best practice and addresses problems highlighted in our empirical research (examples of sub-processes that represent a Level 2 capability are given in section 6). If a company decides not to include a recommended sub-process they should first run an objective assessment in order to determine its importance. This is because breaking down the RE process into these sub-processes may help to uncover hidden weaknesses.

Modelling processes separately therefore allows companies to examine and prioritise their RE process improvement activities. Also, viewing processes independently eases the transition from a descriptive process (that addresses 'what' should be done to improve processes) to a more applicable prescriptive process (that shows 'how' the process can be implemented). A criticism of the SW CMM is that it is too descriptive and does not provide sufficient examples and specific guidelines to help companies with their process improvement activities, e.g. (Lauesen and Vinter, 2001; Potter and Sakry, 2001). By taking key RE sub-processes and extending them into detailed guidelines the R-CMM features specific activities that in turn can be measured.

5.4. *R-CMM metric focus*

It is only through interrogating processes and assessing their strength that a company can determine how well their goals have been met. Measuring the strength of a process will also lead to a better understanding of current practices that in turn will help companies to set realistic project goals (Basili, 1995) and (Madhavji, 1991). Setting realistic goals means recognising and prioritising which processes need strengthening. Although assessment of processes and improvement of processes would appear to be two sides of the same coin, a survey covering 17 European countries with 3,401 responses from software organisations observed that:

“when companies reported that they are improving their software processes, less than 15% of the total are applying SW assessment methods.” (Ibanez and Rempp, 1996)

Organisations therefore need to be guided towards using assessment methods as an integral part of process improvement. The R-CMM models measurement through assessing individual process strengths, as shown in the ‘Metric’ dimension in Figure 3, whilst retaining a goal focus. The assessment method used in the R-CMM is a tried and tested technique where the ‘approach’, the ‘deployment’ and the ‘results’ of each sub-process are measured. The process measurement method is adapted from a model developed by Motorola (Daskalantonakis, 1994). Section 6.5 shows how the metrics form part of the overall model, and Appendix B explains how a sub-process is measured to confirm its capability.

6. **R-CMM Level 2: Model detail**

This section gives examples of the Level 2 R-CMM and builds on the work given in section 4.2.

6.1. *Level 2 RE goal*

The Level 2 goal “Implement a repeatable RE process” in Figure 4 is a refinement of the SW CMM Level 2 focus where repeatable software processes are established. Companies who have few controls over their RE process need to work towards instituting the baseline activities introduced at this level.

6.2. *Level 2 RE questions*

As detailed in Figure 4, the R-CMM for Level 2 presents users with 5 questions that query whether their RE process is complying with the Level 2 maturity goal. The Level 2 goal is decomposed into questions that link to recognised RE phases. To answer questions (Q1-Q5), the user must determine whether processes are successfully instituted at a repeatable level.

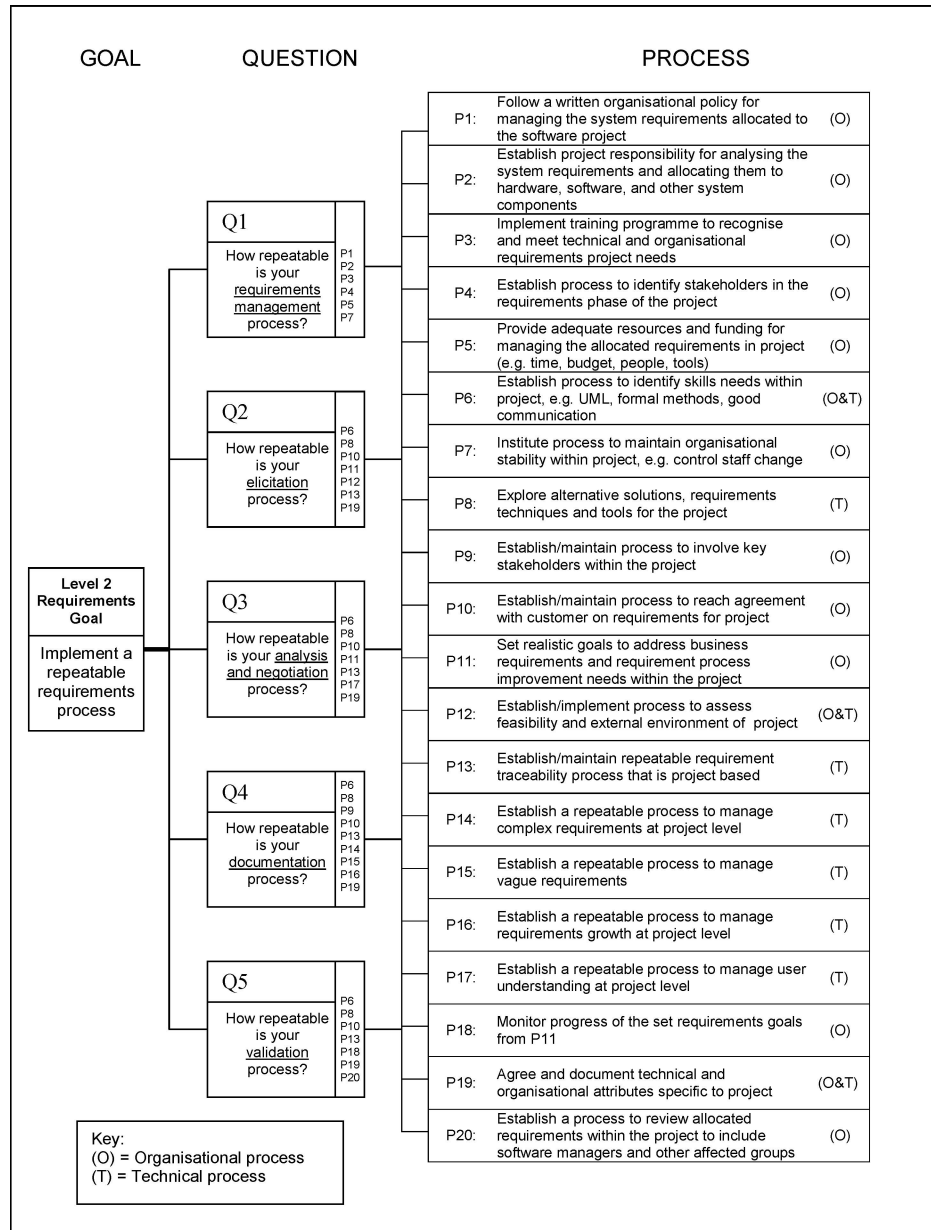


Figure 4. Level 2 R-CMM.

Sub-processes can apply to more than one RE phase, and this relationship is modelled in the R-CMM. For example the sub-process P13 “Establish/maintain a repeatable traceability process that is project based” relates to the elicitation, analysis, documentation and validation phase of the R-CMM. Whereas the sub-process P4 “Establish process to identify stakeholders in the requirements phase of the project” is set up in the RE management phase and therefore relates to this phase. Processes in the R-CMM can therefore span various RE phases and bridge the gap between a traditional, structured ‘lifecycle’ view of RE and the more fluid process view as “requirements are developed iteratively” (Peters and Pedrycz, 2000).

6.3. *The Level 2 RE process*

This section introduces 20 sub-processes that were selected as key to baseline RE needs at a project level as shown in Figure 4. The primary motivation for including each sub-process is shown in Table 2 and is explained in Sections 6.3.1–6.3.3. Process descriptions are kept simple at this level of abstraction; in Section 6.3.4 we provide pointers to more detailed definitions.

6.3.1. *The SW CMM* The SW CMM motivated category (category 1 in Table 2) covers general ‘organisational’ activities within the RE process. From our empirical study we conclude that organisational issues are causing practitioners more problems than technical issues, this is also a finding in Lubars et al. (1993). The SW CMM is therefore placing the right emphasis on ‘managing’ the RE process, yet requires enhancement in the areas presented in our empirical research and the literature (categories 2 and 3 in Table 2).

All SW CMM Key Process Areas (KPA)s start with ‘goals’ to emphasise the importance of this focus. KPA)s include the need to assign responsibilities and resources to each activity. We have adapted these activities to relate specifically to the RE process rather than general software development.

6.3.2. *Empirical research* Problems raised in our empirical research are viewed in two categories: organisational RE problems and technical RE problems (as shown in Table 1). Processes included in the R-CMM directly address the problems raised in both these categories. For detailed analyses of our study refer to Hall et al. (2002b) and Beecham et al. (2003c).

6.3.3. *Literature* Although all SW CMM Key Process Areas (KPA)s start with ‘goals’ there is nothing specific about how companies should identify their own goals based on their own personal RE weaknesses. Therefore we look to the literature for guidance, e.g. (Davis 1988; Sawyer et al., 1997; IEEE, 1998; Robertson and Robertson, 1999) who suggest companies set realistic improvement goals when planning for RE process improvement.

The literature is rich in suggestions for RE process improvement. However, these recommendations can conflict with each other despite being founded on empirical studies. Bach

Table 2. Motivation for including RE processes in the R-CMM at level 2 maturity.

Source	Level 2 RE Processes
1. SW CMM	<p>P1: Follow a written organisational policy for managing the system requirements allocated to the software project</p> <p>P2: Establish project responsibility for analysing the system requirements and allocating them to hardware, software, and other system components</p> <p>P5: Provide adequate resources and funding for managing the allocated requirements in the project</p> <p>P20: Establish a process to review allocated requirements within the project to include software managers and other affected groups</p>
2. Empirical Research	<p>P3: Implement training programme to recognise and meet technical and organisational RE needs within the project</p> <p>P4: Establish process to identify stakeholders within the project</p> <p>P6: Establish process to identify skills needs within project, e.g. UML, Formal methods</p> <p>P7: Institute process to maintain organisational stability within project, e.g. control staff change</p> <p>P10: Establish/maintain process to involve key stakeholders in requirements phase of project</p> <p>P13: Establish/maintain repeatable requirement traceability process that is project-based</p> <p>P14: Establish a repeatable process to manage complex requirements at project level</p> <p>P15: Establish a repeatable process to manage vague requirements at project level</p> <p>P16: Establish a repeatable process to manage requirements growth at project level</p> <p>P17: Establish a repeatable process to manage user understanding at project level</p> <p>P19: Agree and document technical and organisational attributes specific to project</p>
3. Literature	<p>P8: Explore alternative solutions, RE techniques and tools for the project</p> <p>P9: Establish/maintain process to reach agreement with customer on requirements for project</p> <p>P11: Set realistic improvement goals to address problems in the RE process</p> <p>P12: Establish/implement process to assess feasibility & external environment of project</p> <p>P18: Monitor progress of the set requirements goals</p>

(1999) advises not to rely on the literature alone to guide practitioners. Bach states that work of software best practice books can be based on ‘the passions of people who write the textbooks or run huge stuffy companies’. Bach (1999) asks that this knowledge be ‘opened up’ and we have done so by using the literature and best practices in the SW CMM to support our own findings.

6.3.4. Defining the RE process Defining processes needs to be built into software development as it is not a natural activity. As Thomas and McGarry (1994) point out, 4 out of 5 software-development groups have nothing that can be recognised as a software process. Each organisation should tailor the descriptive and necessarily generic definitions given in the R-CMM in order to have a shared and meaningful understanding of the process. Figure 5 is an example of how we define the process through the SW CMM guidelines and the

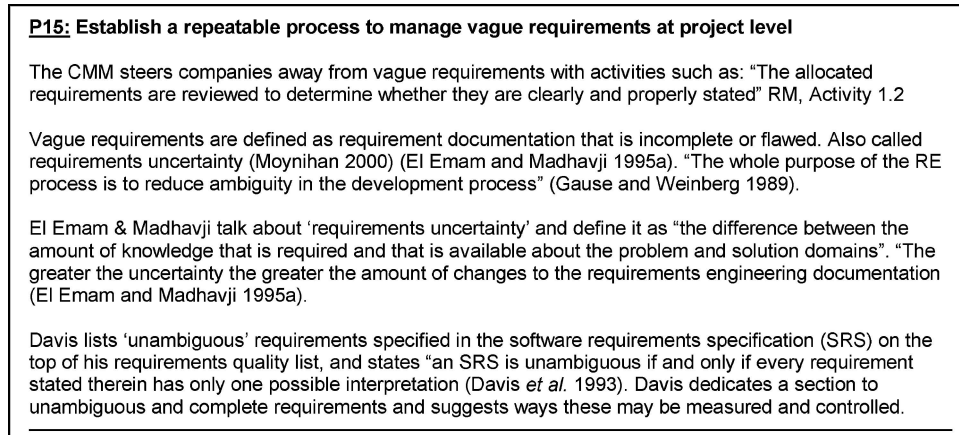


Figure 5. Guiding users to define processes through reference to related work—an example.

literature. These descriptions are not intended to advise users 'how' to solve their problems. At this stage, the model is encouraging users to understand their problems through clear definitions. Detailed definitions and references for all Level 2 processes are included in Beecham et al. (2003b).

6.4. RE guidelines

In the example of a Level 2 process guideline (P4) given in Figure 6, we can trace how the goal to "establish a process to identify stakeholders within the project" comes from the higher level model in Figure 4. This more detailed model retains the Goal Question Process Metric framework of the higher level model. Guideline sub-processes move away from the descriptive process definitions given in Figure 4 as they prescribe lower level processes that are needed to achieve improvement goals. They are based on an analysis of the RE literature. The example given in Figure 6 is based on the work of Sommerville and Sawyer (1997) and is supported by our findings in Hall et al. (2002b). The work of Boehm (2001), StandishGroup (1995), Thayer and Dorfman (1990), El Emam et al. (1996) and Hofmann and Lehner (2001) also highlight the importance of this sub-goal.

Each high level process modelled in Figure 4 has an associated guideline similar to the example given in Figure 6.

6.5. Level 2 RE metrics—The assessment

The assessment forms the final part of the Goal/Question/Process/Metric paradigm. Metrics are used to quantify how well a sub-process has been approached, deployed and what the results of implementing the process yields. This is a flexible assessment method as it is also possible to combine measurements of all the sub-processes to gain a high level view of the RE process strength. We adapt an assessment procedure that has been tried and tested

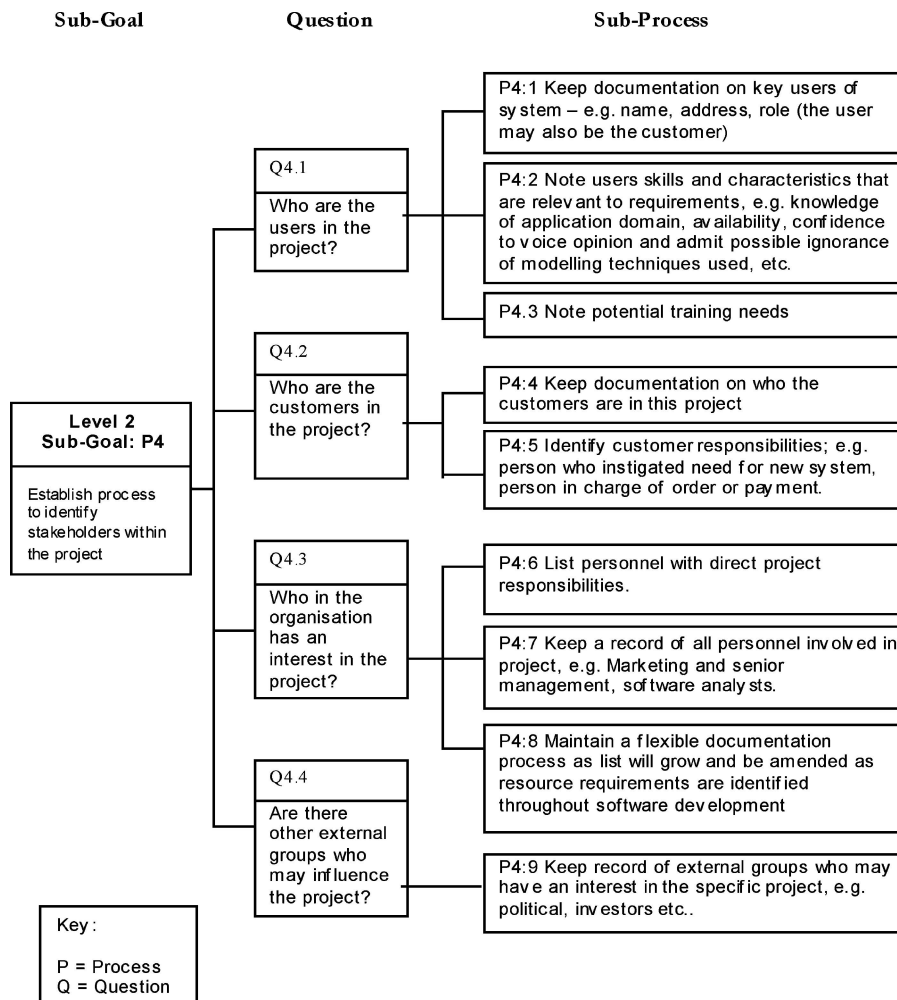


Figure 6. Guideline example of a Level 2 RE process.

in a high level maturity company (Daskalantonakis, 1994) to measure these operations. Assessment is viewed as an essential element of process improvement as any process improvement effort should begin with some kind of assessment, to establish a baseline understanding of current processes and problem areas (Wieggers, 1998a). It is only through an assessment that companies can gain a balanced picture of where their current processes need improving. Further, it is only with this knowledge that companies can set realistic RE process improvement goals (Davis 1988; Sawyer et al., 1997; IEEE, 1998). Practitioners need to identify their own specific reasons for wanting to improve their performance and the assessment will lead companies to look at their current process capabilities and set realistic goals when planning for further RE process improvements.

Sub-processes are assessed one at a time. Appendix B gives an example of how a sub-process is measured and can be combined with other sub-processes to give an indication of the strength of an RE phase such as 'elicitation'.

7. Conclusion

The RE process remains a major source of problems in software development despite the increase in organisations implementing software process improvement programs. In this paper we describe why and how we have adapted the SW CMM to focus on the RE process. We create a specialised process improvement model—the R-CMM—in order to assess the capability of the many sub-processes that combine to make up the RE process. We demonstrate an improvement methodology through a series of models that isolate and focus on the RE phase of software development. The Level 2 R-CMM and process assessment examples in the paper show how practitioners are guided towards recognising their own baseline RE process strengths and weaknesses.

The R-CMM retains the characteristics of the 'parent' SW CMM in order to build on proven methods and promote a smooth transition for practitioners familiar with CMM techniques. Our work is primarily driven by the needs of the software industry, and the model content is based on an analysis of practitioner needs identified in one of our previous empirical studies. Combining the features of the SW CMM and an independent assessment scheme should make the R-CMM applicable to a wide range of software companies.

The processes defined in the R-CMM work together to produce a baseline structure for companies to consider within their software development activities. A first step towards improving and managing RE is to define and tailor processes to meet the specific needs of the organisation. The Goal/Question/Process/Metric continuous improvement approach of the R-CMM guides users towards creating specific goals based on their business needs.

This paper shows how the R-CMM guides and prompts practitioners through the many stages involved in RE process improvement. It is intended that description of the stages involved in developing the model, as outlined in this paper, will provide a foundation for future development in the area of RE process improvement.

8. Future work

A validation study of our R-CMM has recently been carried out with a group of experts in both research and industry (Beecham et al., 2005). Although this is a preliminary study, results suggest the R-CMM has some potential to be a useful tool for both practitioners and researchers in the field of process improvement and RE. Future work includes creating a more flexible, weighted assessment scheme (dependent on a company's experience), and re-defining some R-CMM sub-processes. Verification and evaluation is still required, and future work includes testing the model in an industrial setting.

Appendix A: Glossary of terms

Term	Related term and acronym	Definition of term that applies to this paper
Best Practice		A [proven] tactic or method chosen to perform a particular task and/or to meet a particular objective (Dooley et al., 2001).
Culture	Organisational	“that’s the way we do things around here”(Paulk, 1997, page 10)
Customer		The person, or persons who pay for the product and usually (but not necessarily)decide the requirements. In the context of this and the IEEE (1998) recommended practice the customer and the supplier may be members of the same organisation. The individual, group, organisation that commissions the development of the system (Loucopoulos and Karakostas, 1995).
Engineering		“Engineering is the use of principles to find designs that will meet multiple competing objectives, within limited resources and other constraints, under conditions of uncertainty” (Gilb, 1996).
Engineering Requirements	(see also requirements engineering)	Deals with activities which attempt to understand the exact needs of the users of the software intensive system and to translate such needs into precise and unambiguous statements which will subsequently be used in the development of the system (established as a separate field of investigation and practice in mid 1970s) (Loucopoulos and Karakostas, 1995)
Framework		An essential supporting or underlying structure (Concise Oxford Dictionary, 2001)
Goal Question Metric	GQM	A paradigm proposed by Basili and Rombach (1988) that is used to help decide what measurements should be taken and how they should be used (Sommerville, 2001).
Life cycle	Software	The period of time that begins when a software product is conceived and ends when the software is no longer available for use. The software life cycle typically includes a concept phase, requirements phase, design phase, implementation phase, test phase, installation and checkout phase, operation and maintenance phase, and sometimes, retirement phase. These phases may overlap or be performed iteratively (IEEE, 1999)
Life cycle	System	The period of time that begins when a system is no longer available for use (IEEE, 1999)
Model		A simplified description. . . of a system or process . . . to assist predictions (Concise Oxford Dictionary, 2001)
Normative		Relating to or deriving from a standard or norm (Oxford Dictionary, 2001)
Organisation	Methods of	According to Coad and Yourdon (1990) three methods pervade people’s thinking: (1) the differentiation of experiences into particular objects and their attributes; (2) the distinction between whole objects and the component parts; and (3) the formation of and the distinction between different classes of objects. Taken from “Classification Theory”, Encyclopedia Britannica.

(Continued on next page.)

(Continued).

Term	Related term and acronym	Definition of term that applies to this paper
Organisational RE Process Problems	In PPP project Project management problems	A class of factors internal to the development organisation that indirectly influence the production of the RE Specification that is often the responsibility of management, to include: company culture; developer communication; resources; skills; staff retention; training; user communication.
Paradigm	Worldview	A basic set of beliefs or assumptions that guide qualitative researchers inquiries (Cresswell, 1998)
PPP Project		'Managing Practitioner impact on Processes and Products' (PPP) project. A project funded by the UK's Engineering and Physical Science Research Council under grant number EPSRC GRL91962.
Practitioner		People actively involved in producing software, to include developers, project managers and senior managers.
Practitioner Communication		Communication between staff groups within the Company. E.g. Marketing discussing customer needs and agreements with Software Group, or Requirements Engineers communicating feasibility of design with Software group.
Process		A collection of activities with entity flows among them (Yu and Mylopoulos, 1997) or "particular method of doing something, generally involving a number of steps or operations." Webster's Dictionary in Fayad, 1997.
Process	Maturity	The degree, to which a process is defined, managed, measured, and continuously improved (Dooley et al., 2001).
Process	Tailoring or Customising	"for any process model to be effective in the specific project in hand, there is a need to customise the model according to the project goals. This may be achieved by characterising various aspects of the project (e.g. resource constraints); setting up project goals; assessing how these goals are supported by the adopted process model, tailoring the process model to suit project goals; using the tailored process model in the project; assessing and fine-tuning the model on an on-going basis. The customisation process would be simplified considerably if process models were organised hierarchically, leading from generic models at the top of the hierarchy to specific models at the bottom." (Madhavji, 1991) (the cmm does this to an extent).
Requirement	Or, set of Requirements	A feature or behaviour of the system that is desired by one or more stakeholders (Britton, 2000). A condition or capability needed by a user to solve a problem or achieve an objective. A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed documents. A documented representation of a condition or capability as in (1) and (2) (IEEE, 1999)
Requirements	Allocated (as in CMM)	The agreement with the customer of the requirements for the software project (Davis et al., 1993)
Requirements	Analysis	The process of studying user needs to arrive at a definition of system, hardware, or software requirements. The process of studying and refining system, hardware, or software requirements (IEEE, 1999)

(Continued on next page.)

(Continued).

Term	Related term and acronym	Definition of term that applies to this paper
Requirements	Errors	2 classes according to (Davis et al., 1993) Knowledge errors: caused by not knowing what the true requirements are 2. Specification errors: caused by not knowing how to adequately specify requirements
Requirements	Defects	2 classes according to (Lauesen and Vinter, 2001) 1. Requirements Defects: if the product works as intended by the programmers, but doesn't match the surroundings. One example is that users and customers are not satisfied with it . . . Another example is that the program doesn't cooperate properly with existing, surrounding software. Requirement defects can relate to functional as well as non-functional requirements. 2. Implementation Defects (the development activities that produce a workable program. Implementation is mainly carried out by programmers. We have an implementation defect if the product doesn't work as intended by the programmers. Typically, implementation defects show up as program crashes or obviously wrong results.
Requirements	Functional	A requirement that specifies a function that a system or system component must be able to perform (IEEE, 1999). What the system should do (Sommerville and Sawyer, 1997) e.g. it should generate membership numbers for each person joining club etc.
Requirements	Growth/change	functional and non-functional requirements not documented in original specification that result in changes over time, incorporates changeability decay (Arisholm and Sjoberg, 2000)
Requirements	Non-functional	Systems quantities or quality attributes. E.g. safety, security, reliability, usability, maintainability, cost and development time (Gross and Yu, 2001). High level non-functional requirements often decompose into functional requirements (Sommerville and Sawyer, 1997) they are not specifically concerned with the functionality of a system, placing restrictions of the product being developed and the development process.
Requirements	Phase	The period of time in the software life cycle during which the requirements for a software product are defined and documented
Requirements	Poor user understanding	User understanding of their own needs is often confused and undetected until too late—a customer will often ask for functions that are not needed and prove difficult to implement.
Requirements	Process	Processes/activities within the requirements phase of software development that include elicitation, analysis, documentation and validation as well as links to resources, traceability and general requirements management and engineering.
Requirements	Technical Issues (In PPP Project)	Requirements Growth/change; Vague/ambiguous requirements; Requirements Process definition; Poor user understanding; Requirements traceability. (See also 'Technical RE process problems)

(Continued on next page.)

(Continued).

Term	Related term and acronym	Definition of term that applies to this paper
Requirements	Qualities	Quality attributes in the requirements process (from (Davis et al., 1993)): Achievable; Annotated by Relative Stability; Annotated by Version; Annotated by Relative Importance; At Right Level of Detail; Complete; Concise; Correct; Cross-Referenced; Design Independent; Electronically Stored; Externally Consistent; Executable/Interpretable; Internally consistent; Modifiable; Not Redundant; Organised; Precise; Reusable; Traceable; Traced; Unambiguous; Understandable; Verifiable
Requirements	Specification	A document that specifies the requirements for a system or component. Typically included are functional requirements, performance requirements, interface requirements, design requirements, and development standards (IEEE, 1999).
Requirements	Tacit	Unstated requirements
Requirements	Traceability	A link or definable relationship between entities (Watkins and Neal, 1994) that relates primarily to the requirements stage of software development.
Requirements	Vague/ambiguous	Requirement documentation is incomplete and flawed. Also called requirements uncertainty (Moynihan, 2000).
Requirements Engineering	(See also Engineering-Requirements)	A separate field of investigation and practice established in mid 1970s (Loucopoulos and Karakostas, 1995). The science and discipline concerned with analysing and documenting requirements, including needs analysis, requirements analysis, and requirements specification. It also provides the appropriate mechanisms to facilitate the analysis, documentation, and verification activities. Requirements engineering can also be defined as a combination of requirements analysis and the documentation of the requirements into a form called requirements specifications. Chapter 1 (Thayer and Dorfman, 1990).
Requirements Engineering	Organisational Issues (In PPP Project)	Practitioner communication; Resources; Skills; Staff retention; Training; User communication. (See also Organisational RE Process problems)
Requirements Engineering Process	RE Process	Activities which attempt to understand the exact needs of the users of the software intensive system and to translate such needs into precise and unambiguous statements which will subsequently be used in the development of the system. (Loucopoulos and Karakostas). Activities performed in the requirements phase that culminate in producing a document containing the software requirements specification (Jalote, 1997). The set of activities required to gather, specify, validate and engineer a set of requirements (Britton, 2000) and (IEEE Software—Thayer and Dorfman, 1990, page 1).
Resources	In PPP project	This relates to time, costs, investment in tools and people. Timescales and estimates given at beginning of project to be managed with allocation of adequate resources (staff time/training/costs of new tools) to include long-term software improvement activities.

(Continued on next page.)

(Continued).

Term	Related term and acronym	Definition of term that applies to this paper
Skills	In PPP project	Level of spread and appropriate expertise available to prevent over-dependence on few experienced staff. Sharing of best practice
Specification	Requirements	A document that specifies, in a complete, precise, verifiable manner, the requirements, design, behaviour, or other characteristics of a system or component, and often, the procedures for determining whether these provisions have been satisfied (IEEE, 1999).
Staff Retention	In PPP project	Incorporates recruitment and workforce stability. Recruiting staff of the right level and retaining experienced staff.
Stakeholders		All practitioners and customers, and users—all people affected by the system with direct or indirect influence on the system requirements (Sommerville and Sawyer, 1997)
System		A collection of components organised to accomplish a specific function or set of functions (IEEE, 1999).
System Re-requirements Engineering		Is the science and discipline concerned with analysing and documenting system requirements. It involves transforming an operational need into a system description, system performance parameters, and a system configuration through the use of an iterative process of analysis, design, trade-off studies and prototyping Chapter 1: (Thayer and Dorfman, 1990).
Technical RE process problems	In PPP study	A class of problem that directly influences the production of the RE Specification, more usually the responsibility of Developers and Engineers, to include: Complexity of application; Requirements growth/change; vague requirements; requirements process definition; poor user understanding; requirements traceability.
Traceability		See 'Requirements' traceability
Training		Training needs both in technical and organisational areas
User		The individual, group or organisation that will work with the system itself (Loucopoulos and Karakostas, 1995).
User Commu- nication		Supplier communication with users (e.g. how company structure dictates who discusses customer requirement needs with the customer and user).

APPENDIX B: Example of R-CMM process measurement

This example shows how the R-CMM measures the capability of the elicitation phase of RE. The elicitation phase is just one of the 5 phases represented in the R-CMM. The sub-processes listed in Table 3 define the RE elicitation phase:

1. *Measuring processes*

The first stage involved in measuring the capability of the RE process assesses the strength of one sub-process at a time. This method can be used to assess the strength of any defined process within the R-CMM. Three elements of the sub-process are measured: the *approach* to the process, the *deployment* of the process and the *results* of process application. An average of all three elements gives the score for each sub-process. The six steps below explain how each sub-process is measured to assess its capability. This is adapted from a

Table 3. Level 2 R-CMM Elicitation sub-process.

The R-CMM Level 2 Requirements Elicitation Phase	
Process Ref.	Sub-Process Description
P6	Establish process to identify skills needs within <i>elicitation</i> phase of the project, e.g. UML, Formal methods
P8	Explore alternative solutions , requirements techniques and tools for the <i>elicitation</i> phase of project
P10	Establish and maintain process to involve key stakeholders in requirements <i>elicitation</i> phase of project
P11	Set realistic goals to address business requirements and RE process improvements needs within project
P12	Establish and implement process to assess feasibility & external environment of project
P13	Establish and maintain repeatable requirement traceability process that is specific to the project
P19	Agree and document technical and organisational attributes specific to the <i>elicitation</i> process in the project

method used by Motorola to internally assess their software processes (Daskalantonakis, 1994).

Step One. A clear understanding of the process is confirmed

A detailed definition is included with each question. The participant only continues with the assessment if the definition is clearly understood. Prior to participating in the questionnaire assessment, participants are told “Please note: you do not have to personally be involved in performing the process—it’s enough that you know who performs it to answer the following” (SEI, 1996).

Step Two: Measuring the ‘Approach’ to the sub-process

The first of the 3 measurement elements is based on the participant’s understanding of the company’s ‘*approach*’ to the sub-process. This encompasses the SW CMM characteristics of demonstrating a commitment to perform and ability to perform the process. Table 4 gives an example of how a participant might measure a sub-process. A specific example might be that participants are asked to tick the most appropriate statement (in Table 4) relating to how their company approaches the process P6 - “established a process to identify skills needs within *elicitation phase* of the project, e.g. UML, Formal methods”:

‘Approach’ score for sub-process P6: The process “*Establish process to identify skills needs within elicitation phase of the project*” scores a ‘6’, i.e. $((6 + 8 + 4)/3 = 6)$, so the opinion of this stakeholder is that the company’s ‘approach’ to this process is marginally qualified. The stakeholder then assesses P6 against the other two elements: *deployment* and *results*.

Step Three: Measuring the Deployment of the Sub-Process

The next stage is to assess how a process is deployed in practice. The statements used to measure this incorporate SW CMM characteristics where each process is analysed,

Table 4. Generic matrix measuring an organisation's approach to any defined process.

Approach	Score
<i>Management Approach</i>	(Tick one of the options)
No management recognition of need	<i>Poor</i> (0)
Management has begun to recognise the need	<i>Weak</i> (2)
Wide but not complete commitment by management	<i>Fair</i> (4)
Some management commitment/some are proactive	✓ <i>Marginally qualified</i> (6)
Total management commitment; majority are proactive	<i>Qualified</i> (8)
Management provides zealous leadership & commitment	<i>Outstanding</i> (10)
Management interest not known	<i>N/a</i>
Management interest not believed relevant	<i>N/a</i>
<i>Organisational Approach</i>	(Tick one of the options)
No organisational ability/ No organisational commitment	<i>Poor</i> (0)
The practice is implemented in one or two projects	<i>Weak</i> (2)
Road map for practice implementation defined	<i>Fair</i> (4)
Practice implementation under way in parts of the organisation	<i>Marginally qualified</i> (6)
Practice established as an integral part of the requirements phase	✓ <i>Qualified</i> (8)
Organisational excellence in practice recognised even outside org	<i>Outstanding</i> (10)
Organisational approach not known	<i>N/a</i>
Organisational approach not believed relevant	<i>N/a</i>
<i>Support for Practice</i>	(Tick one of the options)
Practice not evident	<i>Poor</i> (0)
Support items for the practice start to be created	<i>Weak</i> (2)
Several supporting items for the practice in place	✓ <i>Fair</i> (4)
Supporting items in place	<i>Marginally qualified</i> (6)
Supporting items encourage and facilitate use of practice	<i>Qualified</i> (8)
All support items in place continue to be improved	<i>Outstanding</i> (10)
Support for practice not known	<i>N/a</i>
Support for practice not believed relevant	<i>N/a</i>

monitored and verified. It follows similar layout to Table 4 where responses range from poor to outstanding. For the purposes of this example the Stakeholder scores a total of 3 for deployment of process P6.

Step Four: Measuring the Results of the application of Sub-Process

This final dimension measures whether the process goals are appropriate and looks at the effectiveness of the activities performed. These measurements are also characteristics of the SW CMM. The measurement matrix follows a similar pattern of assessment to Table 4 where responses range from poor to outstanding. For the purposes of this example the Stakeholder scores a total of 3 for results of implementing P6.

Step Five: Combining 3 evaluation scores to assess the strength of each Sub-Process

All three evaluation dimensions and their scoring guidelines are examined simultaneously and all dimensions are equally weighted. Averaging the score of process assessment

Organisation: ORG_NAME

CMM Level 2 Processes

Date:

KRPA: Requirements Elicitation Phase

Average Score: 6

(4 + 5 + 7 + 6 + 6 + 7 + 7 = 42 / No. of processes (7) = 6)

	List of key processes	1	2	3	4	5	6	7	8	9	10
P6	Establish process to identify skills needs within <i>elicitation phase</i> of the project, e.g. UML, Formal methods				✓						
P8	Explore alternative solutions , requirements techniques and tools for <i>the elicitation phase</i> of project					✓					
P10	Establish and maintain process to involve key stakeholders in requirements <i>elicitation phase</i> of project							✓			
P11	Set realistic goals to address business requirements and requirements process improvements needs within the project						✓				
P12	Establish and Implement process to assess feasibility & external environment of project						✓				
P13	Establish and maintain repeatable requirement traceability process that is specific to the project							✓			
P19	Agree and document the technical and organisational attributes specific to the <i>elicitation process</i> in the project							✓			

Figure 7. Requirements elicitation phase assessment sheet.

indicates a level of capability. For example P6 is 'fair' if it receives a score of 4 when averaging scores for its approach, deployment and results, for example using scores in steps 2–4 above: $(6 + 3 + 3)/3 = 4$.

Step Six : Combining sub-process scores to assess strength of each RE phase

When all the sub-processes have been assessed, then a capability for each RE phase can be obtained. Figure 7 gives an example of a RE Phase Assessment sheet. It shows how each measured process is combined to give a score that relates to—in this case—the capability of the elicitation phase of requirements. All the 5 RE phases are assessed in a similar way.

This assessment can generate the following results:

- A score for each sub-process (seven given in the example relating to requirements elicitation)
- A score for each of the 5 RE phases (management, elicitation, analysis and negotiation, documentation and validation)
- A score for the RE process (all 5 RE phases)

Validation of the R-CMM highlighted that giving each of the above dimensions the same weighting may not suit some companies (Beecham et al., 2005). For example, the 'application' section may be considered more important than the 'approach', i.e. if the process proves to be very useful and is being used successfully, management support may not be so important. In this case, a company may decide to place a weighting on the application dimension. This is a subject in need of further investigation and forms part of our future work.

Appendix B has shown how a sub-process is defined and assessed to establish its strength within the RE process.

Acknowledgments

We are sincerely grateful to all the companies and practitioners who participated in the ‘Managing Practitioner impact on Processes and Products’ (PPP) project. The PPP project is funded by the UK’s Engineering and Physical Science Research Council, under grant number EPSRC GRL91962. We are also grateful to Pete Sawyer for his support in the project.

Note

1. [®]CMM is registered in U.S. Patent and Trademark Office. Accuracy and interpretation of this document are the responsibility of the University of Hertfordshire, Centre for Empirical Software Process Research. Carnegie Mellon University has not participated in this publication.

References

- Arisholm, E. and Sjoberg, D.I.K. 2000. Towards a framework for empirical assessment of changeability decay, *Journal of Systems and Software* (53): 3–14.
- Basili, V.R. 1995. Measurement frameworks, *Software Quality Assurance and Measurement: A Worldwide Perspective*, eds. N. Fenton, R. Whitty and Y. Iizuka. London, International Thomson Computer Press.
- Basili, V.R. and Rombach, H.D. 1988. The tame project: Towards improvement-oriented software environments, *IEEE Transactions on Software Engineering* 14(6): 758–773.
- Beecham, S., Hall, T., Britton, C., Cottee, M., and Rainer, A. 2005. Using an expert panel to validate a requirements software process improvement model, *Journal of Systems and Software* (Accepted and In press).
- Beecham, S., Hall, T., and Rainer, A., 2003a. *Building a Requirements Process Improvement Model*, Hatfield, University of Hertfordshire.
- Beecham, S., Hall, T., and Rainer, A. 2003b. *Defining a Requirements Process Improvement Model*, Hatfield, University of Hertfordshire.
- Beecham, S., Hall, T., and Rainer, A. 2003c. Software process improvement problems in 12 software companies: An empirical analysis, *Empirical Software Engineering* 8(1): 7–42.
- Beecham, S., Hall, T., and Rainer, A. 2004. *Developing a RE Process Improvement Model*, EuroSPI 2004, Trondheim, Norway.
- Boehm, B. 2001. *Using Win-Win for Requirements Negotiation: A Mini Tutorial*, Imperial College, London.
- Britton, C. 2000. *Object-Oriented System Development : A Gentle Introduction*, London, McGraw-Hill, New York.
- Brodman, J.G. and Johnson, D.L. 1994. *What Small Businesses and Small Organisations Say About the CMM*. 16th International Conference on Software Engineering, Sorrento, Italy, IEEE.
- Burnstein, I., Suwannasart, T., and Carlson, C. 1996. Developing a testing maturity model, Part II. *CrossTalk The Journal of Defense Software Engineering*.
- Christie, A.M. 1999. Simulation in support of CMM-based process improvement, *Journal of Systems and Software* 46(2–3): 107–112.
- CMMI 2001. Capability Maturity Model [®]. Integration (CMMI SM), Version 1.1, Software Engineering Institute.
- Cresswell, J. 1998. *Quality Inquiry and Research Design: Choosing Among Five Traditions*. Sage Publications, Thousand Oaks.
- Curtis, B., Hefley, W.E., Miller, S., and Konrad, M. 1995. People capability maturity model. Pittsburgh, PA, Software Engineering Institute, Carnegie Mellon University.
- Daskalantonakis, M.K. 1994. Achieving higher sei levels, *IEEE Software* 11(4).

- David, A. 2000. Models implementation: A state of the art, *European Journal of operational Research* 134: 459–480.
- Davis, A. 1988. A taxonomy for the Early stages of the software development life cycle, *Journal of Systems and Software* 297–311.
- Davis, A., Overmyer, S., Jordan, K., Caruso, J., Dandashi, F., Dinh, A., Kincaid, G., Ledebner, G., Reynolds, P., Sitaram, P., Ta, A., and Theofanos, M. 1993. Identifying and measuring quality in a software requirements specification, In *Proceedings of the First International Software Metrics Symposium*.
- Deming, W.E. 1982. Quality, Productivity, and Competitive Position. Cambridge, MA, Massachusetts Institute of Technology Center for Advanced Engineering Study.
- Dooley, K., Subra A., and Anderson, J. 2001. Maturity and its impact on new product development project performance, *Research in Engineering Design* 13(1): 23–29.
- Dorfman, M. and Thayer, R.H. 1997. *Software Engineering*, IEEE Computer, Los Alamitos, CA. Society Press.
- El Emam, K. and Madhavji, N.H. 1995a. *A Field Study of Requirements Engineering Practices in Information Systems Development*. Second IEEE International Symposium on Requirements Engineering, IEEE Computer Society Press.
- El Emam, K. and Madhavji, N.H. 1995b. *The Reliability of Measuring Organizational Maturity*, John Wiley & Sons.
- El Emam, K., Quintin S., and Madhavji, N.H. 1996. User participation in the requirements engineering process: An empirical study, *Requirements Engineering Journal* 1(1): 4–26.
- Fayad, M.E. 1997. Software development process: A necessary evil, *Communications of the ACM* 40(9): 101–103.
- Gause, D.C. and Weinberg, G.M. 1989. *Exploring Requirements: Quality Before Design*, New York, NY, Dorset House Publishing.
- Gilb, T. 1996. Level 6: Why we can't get there from here, *IEEE Software* 13(1): 97–98, 103.
- Gross, D. and Yu, E. 2001. From non-functional requirements to design through patterns, *Requirements Engineering* (6): 18–36.
- Hackos, J.T. 1997. From theory to practice: Using the information process maturity model as a tool for strategic planning, *Technical Communication* (44): 369–381.
- Hall, T., Beecham, S. and Rainer, A. 2002a. *Requirements Problems in Twelve Software Companies: An Empirical Analysis*. EASE 2002, 6th International Conference on Empirical Assessment and Evaluation in Software Engineering, Keele University, Staffordshire, UK.
- Hall, T., Beecham, S., and Rainer, A. 2002b. *Requirements Problems in Twelve Software Companies: An Empirical Analysis*. IEE Proceedings for Software.
- Hayes, W. and Zubrow, D. 1995. *Moving on Up: Data and Experience Doing CMM-Based Process Improvement*, Pittsburgh, Carnegie Mellon University.
- Hofmann, H.F. and Lehner, F. 2001. Requirements engineering as a success factor in software projects, *IEEE Software*: 58–66.
- Humphrey, W.S. 1989. *Managing the Software Process*. Addison-Wesley Publishing Company, Inc, Reading, Massachusetts, USA.
- Humphrey, W.S. 1997. *Introduction to the Personal Software Process*, Addison-Wesley, Reading, M.A.
- Humphrey, W.S. 2000. *Introduction to the Team Software Process*, Addison-Wesley, Reading, M.A.
- Humphrey, W.S. 2002. Three process perspectives: organizations, teams, and people, *Annals of Software Engineering* 14: 39–72.
- Ibanez, M. and Rempp, H. 1996. European User Survey Analysis, ESPITI Project Report. 2003.
- IEEE 1998. IEEE guide to software requirements specification, *IEEE Standard*. Piscataway, N.J., IEEE Press. pp. 830–1998.
- IEEE 1999. IEEE Standards Software Engineering, 1999 Edition. Volume One: Customer and Terminology Standards. New York, USA, The Institute of Electrical and Electronics Engineers, Inc.
- Jalote, P. 1997. *An Integrated Approach to Software Engineering: Second edition*, New York, Springer-Verlag.
- Lauesen, S. and O. Vinter 2001. Preventing requirement defects: An experiment in process improvement, *Requirements Engineering Journal* 6(1): 37–50.

- Lindland, O.V., Sindre, G., and Solvberg, A. 1994. Understanding quality in conceptual modeling, *IEEE Software* 11(2): 42–49.
- Loucopoulos, P. and V. Karakostas 1995. *System Requirements Engineering*, London, UK, McGraw-Hill Book Company Europe.
- Lubars, M., Potts, C., and Richter, C. 1993. A review of the state of the practice in requirements modeling, In *Proceedings of the IEEE International Symposium on Requirements Engineering*, San Diego, CA.
- Madhavji, N.H. 1991. The process cycle, *Software Engineering Journal* 6(5): 234–242.
- McFeeley, B. 1996. *IDEAL: A User's Guide for Software Process Improvement*. Pittsburgh, PE, USA, Software Engineering Institute, Carnegie Mellon University.
- Moynihan, T. 2000. Coping with requirements-uncertainty: The theories-of-action of experienced IS/software project managers, *Journal of Systems and Software* (53): 99–109.
- Neissink, F., Clerc, V., and Vliet, H. 2002. The IT Service Capability Maturity Model. Utrecht, Software Engineering Research Centre: 127.
- Ngwenyama, O. and Neilsen, P. 2003. Competing values in software process improvement: an assumption analysis of cmm from an organizational culture perspective, *IEEE Transactions on Engineering Management* 50(1): 100–112.
- Paulk, M.C. and Chrissis, M.B. 2000. The November 1999 High Maturity Workshop. *Carnegie Mellon Uni*, Software Engineering Institute.
- Paulk, M.C., Weber, C.V., Curtis, B., and Chrissis, M.B. 1995. *The Capability Maturity Model: Guidelines for Improving the Software Process*. Reading, Massachusetts, Addison Wesley Longman, Inc.
- Perry, D.E., Staudenmayer, N.A., and Votta, L.G. 1994. People, organizations and process improvement, *IEEE Software* 11(4): 36–45.
- Peters, J. and Pedrycz, W. 2000. *Software Engineering : An Engineering Approach*, John Wiley & Sons, New York.
- Pfleeger, S.L. 1995. Maturity, models, and goals: How to build a metrics plan, *Journal of Systems and Software* 31(2): 143–155.
- Potter, N. and Sakry, M. 2001. Practical CMM, Software Development. 2002.
- Pressman, R. 2001. *Software Engineering 5th Edition*, McGraw Hill.
- Reifer, D.J. 2000. The CMMI: it's formidable, *Journal of Systems and Software* 50: 97–98.
- Rifkin, S. 2001. Why Software Process Innovations Are Not Adopted. *IEEE Software*.
- Robertson, S. and Robertson, J. 1999. *Mastering the Requirements Process*, Addison-Wesley.
- Rogoway, P. 1998. How to Reap the Business Benefit from SPI: Adding SPICE while preserving the CMM (Motorola), SPI NEWSPAPER. European Software Process Improvement. SPI and Assessments. 2003.
- Rossi, S. 1999. *Moving Towards Modelling Oriented Software Process Engineering: A Shift from Descriptive to Prescriptive Process Modelling*. International Conference on Product Focused Software Process Improvement, Oulu, Finland, Technical Research Centre of Finland ESPOO 1999.
- Sawyer, P., Sommerville, I., and Viller, S. 1997. Requirements process improvement through the phased introduction of good practice, *Software Process Improvement and Practice* 3(1).
- SEI 1996. A Description of the Systems Engineering Capability Maturity Model Appraisal Method Version 1.1. Pittsburgh, USA, Software Engineering Institute, Carnegie Mellon University.
- SEI 2002. Process maturity profile of the software community, *Software Engineering Institute*. Pittsburgh, PA, Carnegie-Mellon Univ.
- Solingen, R.V. and Berghout, E. 1999. *The Goal/Question/Metric Method: A Practical Guide for Quality Improvement of Software Development*, Maidenhead, UK, McGraw-Hill Publishing Company.
- Sommerville, I. 2001. *Software Engineering*, Wokingham, Addison-Wesley Publishing Company.
- Sommerville, I. and Sawyer, P. 1997. *Requirements Engineering A Good Practice Guide*, Chichester, John Wiley & Sons Ltd.
- StandishGroup 1995. Chaos Report.
- Thayer, R.H. and Dorfman, M. 1990. *System and Software Requirements Engineering*, Los Alamitos, CA, IEEE Computer Society Press.
- Watkins, R. and Neal, M. 1994. Why and how of requirements tracing, *IEEE Software(July)*: 104–106.
- Wieggers, K. 1998a. Molding the CMM to your organization, *Software Development* 6(5).
- Wieggers, K. 1998b. Read my lips: No new models!, *IEEE Software* 15(5).

Yu, E.S.K. and Mylopoulos, J. 1997. *Modelling Organizational Issues for Enterprise Integration*, International Conference on Enterprise Integration and Modelling Technology, Turin, Italy.



Sarah Beecham is a research fellow in the Department of Maths and Computing in The Open University in the UK. She is currently working on the EPSRC funded CRESTES project (<http://mcs.open.ac.uk/crestes>) looking into modelling resource estimation for long-lived software. She has recently completed her PhD for a program of work entitled "A Requirements-based Software Process Maturity Model". Current research interests are in estimation for software evolution and maintenance and in the general areas of software process improvement. Her particular research interests are in empirical methods in software engineering and requirements engineering.



Tracy Hall leads the Systems & Software Research Group in the Department of Computer Science at the University of Hertfordshire. She specialises in the empirical investigation of technical and non-technical issues within software engineering. During the past ten years Tracy has successfully collaborated with many companies on a variety of research projects. She is very active in the Empirical Software Engineering community and is regularly invited to talk about empirical methods both in the UK and abroad. Tracy is an accomplished researcher having published over twenty high quality journal papers.



Austen Rainer Austen Rainer is a senior lecturer at the University of Hertfordshire. He studied for his PhD at Bournemouth University, in conjunction with IBM Hursley Park. His current research interests include open source software development, longitudinal case study research, and the credibility of empirical evidence for researchers and software practitioners.