



Requireonautics Quarterly

The Newsletter of the Requirements Engineering
Specialist Group of British Computer Society

<http://www soi.city.ac.uk/homes/gespan/rq/rq.html>

© 1999, BCS RESG

Issue 18 (August 1999)

RE-Locations

RE-Soundings	1	RE-Publications	8
Editorial.....	1	Book Review "Practical Software Requirements " by Benjamin Kovitz.....	8
Chairman's Message.....	1	RE-Bites	8
RE-Treats	2	RE-Calls	10
Minutes of the RESG Annual General Meeting.....	2	ICSE 2000.....	10
Next event: Managing Requirements Change.....	3	ICRE 2000.....	10
Other forthcoming events.....	3	ICSR 2000.....	11
RE-News	4	OOPSLA '99 Workshop on Use Case Patterns.....	11
Calendar.....	4	CAiSE 2000.....	12
RE-Readings	4	RE-Sources	13
Requirements for E-commerce.....	4	Web Pages.....	13
RE-Papers	6	Mailing lists.....	13
The Richness of the Requirements Engineering Process.....	6	RE-Actors	13
CORE-Blimey!	8	The committee of RESG.....	13

RE-Soundings

Editorial

Very few things in life never come to an end and editing a newsletter certainly could not be the exception to the rule. This is the last issue of the RQ that I edit. Peter Sawyer (University of Lancaster) is taking over from me as the new Editor-In-Chief of the newsletter. I have no doubt that he will be editing RQ in an effective and creative way. He has my warmest wishes and support.

This is also a good opportunity to thank all the old RESG committee members for the fruitful collaborations we have had over the last two years and wish the members of the newly elected committee success in their future endeavours.

Last but not least the contents of this issue. We've got the minutes of the 5th Annual RESG meeting held on the 14th of July in London. Ian Alexander reports on the event on "requirements for electronic commerce" which was organised by the RESG on the same day. Barbara Paech and Stephanie Lindstaed share with us their experiences with industrial requirements engineering activities and Bashar Nuseibeh reviews a new requirements engineering book by Benjamin Kovitz. Our usual departments complete this issue.

*George Spanoudakis,
Department of Computing, City University*

Chairman's Message

I am writing this just after RESG's Annual General Meeting (AGM), marking the group's fifth anniversary. The AGM saw some changes to the constitution of the RESG executive committee. I am delighted to welcome onto the committee Peter Sawyer from the University of Lancaster, who will take over from George Spanoudakis as Editor-in-Chief of RQ, starting from the next issue. I would like to thank George for his contributions to RQ and the RESG over the last two years, and wish him all the best for the future.

David Shearer from the University of Hertfordshire has also joined the committee, taking over from Sara Jones as Membership Secretary of the group. The co-location of David and Sara at Hertfordshire should mean a smooth transition, with minimal impact on membership renewals when they are due. I would also like to take this opportunity to thank Sara for five years of outstanding service to the RESG in her role as Membership Secretary. I am delighted to say that she has agreed to remain on the RESG

committee, which will benefit immensely from her skills and experience. The minutes of the AGM appear in this newsletter. Please do let me know if you have any comments or questions.

The group's next two meetings will take place outside London. On 31st August 1999 we travel to Keble College, Oxford, for a half-day meeting on "Managing requirements change". The meeting will be co-located with the International Conference on Software Maintenance, and we expect to have a number of participants from outside the UK.

On 23rd November, we travel to York for a meeting on requirements for commercial Off-the-shelf software

(COTS). More details on this will appear in the next RQ and on the RESG web page. Speaking of the web page, our RESG Web officer, Laurence Brooks, is busy applying his considerable HCI expertise to redesign the RESG web page over the summer. Watch the following URL:

<http://www.cs.york.ac.uk/bcs/resg/>

In the meantime, have a great summer.

*Bashar Nuseibeh,
Imperial College, London*

RE-Treats

Minutes of the 5th Annual General Meeting of the Requirements Engineering Specialist Group of the British Computer Society

1. Minutes of the Previous Meeting

The minutes of the previous meeting (RQ, July 1998) were approved.

2. Chairman's Report

2.1 Events

The following events had been held by the RESG in a busy and successful year:

- UML Tutorial (March 1998)
- Scenarios Day (May 1998)
- IEE/RESG Colloquium on: Managing requirements change: a BPR perspective (June 1998)
- Conference on European Industrial Requirements Engineering (October 1998)
- Patterns Day (May 1999)

2.2 Newsletter

4 issues of Requireautics Quarterly were produced during the year.

Pete Sawyer had taken over as editor.

2.3 Industrial Liaison

Dr Barbara Farbey had taken on the role of Industrial Liaison Officer, which she also served on the RENOIR project.

2.4 Other

The RESG had nominated Michael Jackson for the Lovelace Medal, and had solicited the support of the FACS, OOPS, ISM, CASE & Reuse SGs. The RESG was pleased that he had received the medal, raising awareness of Requirements Engineering throughout the profession.

3. Membership Report

Sara Jones reported on a healthy turnover of new members over the past 5 years. There were currently 263 members. Renewal had been extended to 2 years. Membership consisted of a good mix of student, academic, corporate and industrial members.

4. Treasurer's Report

The group's finances remained healthy with a good reserve built up from special 1-day events. Most meetings were designed only to break even so as to provide a service to the Requirements Engineering community.

5. Publicity

Publicity was key to the success of the RESG, and Carol Britton and Vito Veneziano continued to ensure effective mailings, advertisements, hardcopy fliers, and liaison with the website.

6. Election of Executive Committee

Alistair Sutcliffe proposed and Richard Veryard and Mehmet Ozcal seconded the re-election of the committee for a further year. The proposal was carried unanimously. The newly elected committee is:

Chair:	Bashar Nuseibeh
Secretary:	Wolfgang Emmerich
Treasurer:	Neil Maiden
Membership Secretary:	David Shearer
Newsletter Editor:	Peter Sawyer
Associate Editor:	Ian Alexander
Industrial Liaison Officer:	Barbara Farbey
Publicity Officer:	Carol Britton
Associate Publicity Officer:	Vito Veneziano
Web Officer:	Laurence Brooks
Members-at-large:	Steve Easterbrook, Sara Jones

7. Any Other Business

The chairman reported on future plans and the relaunch of the website.

7.1 Planned meetings

The following meetings were already planned for 1999-2000:

- *E-commerce* (held on 14th July 1999, London)
- *Managing requirements change* (31st August 1999, Oxford)
- *COTS* (23rd November 1999, York)
- *Dependable Distributed System Requirements* (February 2000, London)

7.2 Re-launching the RESG website

The group had identified the website as a key priority for spreading the message, informing the community of RESG activities, and attracting new members. It had therefore been decided to relaunch the site with additional information and interaction.

Next event organised by the group.

Managing Requirements Change

Date: 31st August 1999

Location: Keble College, Oxford, co-located with the International Conference on Software Maintenance

This meeting will explore the problems of and solutions for managing changing requirements. In particular, distinguished speakers will explore the relationships and interplay between changing requirements and systems architecture. Four presentations will be followed by an open panel discussion.

Programme:

Chair: Ian Alexander (Independent Consultant)

2:00-2:30 Jane Searles (ICL, UK), "Understanding IT requirements in a changing world – The role of Enterprise Architecture"

2:30-3:00 Dr. Jun Han (Monash University, Australia - visiting University College London), "Managing Air Traffic Control system requirements and their change"

3:00-3:30 Dr. Richard Stevens (QSS, UK), Title: TBA

3:30-4:00 Break

4:00-4:45 Profs. Spencer Rugaber, Gregory Abowd and Prof. Colin Potts (Georgia Institute of Technology, USA) , "Coping with mission-oriented requirements changes"

4:45-5:30 Panel Discussion & Questions

5:30 Close

Participation is free to BCS RESG members, and £5 to all others.

Speakers:

Jane Searle (Process Stream Manager, Initiative for Software Engineering Support Environment Change, ICL, UK)

Richard Stevens (Founder of QSS, UK), Gregory Abowd (Assistant Professor, College of Computing, Georgia Institute of Technology, USA)

Colin Potts (Associate Professor, College of Computing, Georgia Institute of Technology, USA)

Spencer Rugaber (Senior Research Scientist, College of Computing, Georgia Institute of Technology, USA)

Jun Han (department of Computer Science, Monash University, currently on sabbatical at UCL)

Contact: Ian Alexander (iany@easynet.co.uk)

Other forthcoming events organised by the group.

Requirements and COTS

Date: November 23, 1999

Location: York

Contact: Laurence Brooks, University of York (E-Mail: Laurence.Brooks@cs.york.ac.uk)

Dependable Distributed System Requirements

Date: February 2000

Location: London

Contact: Sara Jones, University of Hertfordshire (E-Mail: S.Jones@herts.ac.uk), Wolfgang Emmerich, University College, London (E-Mail: W.Emmerich@cs.ucl.ac.uk)

Calendar

We have received the following notices of forthcoming events in which RQ readers may like to participate.

September 99

REP 99 – 1st International Workshop on the Requirements Engineering Process: Innovative Techniques, Models, Tools to support the RE Process, Florence, Italy, September 3, 1999

<http://www.univ-paris1.fr/CRINFO/REP9>

ESEC/FSE 99, Joint 7th European Software Engineering Conference (ESEC) and 7th ACM SIGSOFT International Symposium on the Foundations of Software Engineering (FSE-7), Toulouse, France, September 6-10, 1999

<http://www.iam.unibe.ch/~esec99/>

CREWS Workshop and Final Review (co-located with the ESEC/FSE '99 conference) on Scenarios for Systems Requirements Engineering, Toulouse, France, September 10, 1999

A two-hour forum organised around the European ESPRIT long-term research project 21903 'CREWS' (Cooperative Requirements Engineering With Scenarios), which is concerned with systematic support for using and managing scenarios in the development of complex systems. The forum will discuss how systems engineering practitioners can improve their processes through the use of scenario-based techniques.

<http://www.cert.fr/anglais/dprs/cadres148.htm>

ISCO4 – Information Systems Concepts: An Integrated Discipline Emerging, Leiden, The Netherlands, September 20-22, 1999

<http://www.wi.leidenuniv.nl/~verrynst/ISCO4.html>

ACRE'99 – The Fourth Australian Conference on Requirements Engineering, Macquarie University, Sydney, Australia, September 29-30, 1999

<http://www.jrcase.mq.edu.au/conference/acre99.html>

RE-Readings

Reviews of recent Requirements Engineering events.

Requirements for E-commerce

Report by Ian F. Alexander

The "Requirements for E-Commerce" meeting was held at Imperial College, London on Wednesday 14th July, together with the RESG's 5th birthday party and AGM. It was organised and co-chaired by Linda Macaulay (UMIST) and Sara Jones (Herts).

Linda Macaulay introduced the meeting, saying that there were two contrasting models of E-Commerce.

Lester Warnegger's Promise Model involved three interlocking areas, namely making, accepting, and keeping promises. These moved from sales and marketing, through electronic service, to actual order fulfillment.

Paul Timmers (in a paper referred to by several speakers, from *Electronic Markets*, Vol 8, no 2, 1998) drew a graph with the x-axis for degree of innovation and the y-axis for the number of functions provided by the E-Commerce site. Sites ranged from simple E-shops (low, low) to 3rd-party markets and integrated value chains (high, high).

E-Commerce was thus a very broad spectrum of different things, and no doubt everyone present had their own definition of it. It brought new understanding of what a 'user' and a 'business' was; new opportunities; accelerating change and globalization; and perhaps new stress on various

quality requirements such as reliability, maintainability, security, and privacy.

Did the short timescales and the pace of change mean there was no time for requirements? She possibly thought so, but the audience possibly did not.

Jim O'Neill (Innovation Management Centre, Stoke on Trent) spoke on SME Requirements for E-Commerce. His centre aimed to help small and medium enterprises in the Stoke area to get going with E-Commerce, by providing training courses, consultancy, drop-in sessions, and conferences. About 200 companies have received training; about 100 companies have received individual kickstart consultancy. However, only 64% of the companies used the web for business, 45% had a website of their own, and none had active E-Commerce sites able to handle transactions. Even E-mail was a major advance for some of these firms; the companies cited cost, lack of skill and lack of time as reasons for not going further. Fears about security were widespread. Basic skills such as use of keyboards and spreadsheets remained weak. An EU-funded (ReChar) project will try to address some of the problems experienced by the companies.

Richard Veryard asked if there was not an implied roadmap, from E-mail to web use to having a website to E-commerce, and O'Neill agreed; E-mail gave a rapid payback. None of the sites built had cost more than £2500, and most were less than £500. At least he could say that

60% of the companies felt their sites were 'a good investment', however unquantifiable that was.

Dr. Kathy Keeling (UMIST) spoke on Requirements for User Acceptance of E-Commerce from her background of research into retailing. She began by explaining how hard it was to put a normal website onto interactive TV, let alone a kiosk in a public place where people did not have their reading glasses, did not know what a scrollbar was, and on average spent just 2 minutes in the kiosk. The market was big and complex, was changing fast, and involved inexperienced users from the C-D social groups. Everett Rogers' classic work showed that there was a yawning chasm between the small percentage of early adopters who enjoyed risking any new gadget, and the majority who were averse to risk and wanted convenience and ease of use. Evidently there were at least two user communities: a third group consisted of laggards and skeptics who would probably never become users.

There was no scope for training kiosk users, so there was exactly one chance to get the service right: otherwise, users would actively avoid the kiosks after a bad experience. The requirements process had to be based closely on value and benefit to (a few) real customers; then later one could do quantitative surveys to decide where to spend the money. The development life-cycle therefore started with some simple goals, observation of task interactions as presently carried out, co-operative requirements capture (a la Linda Macaulay) – done in a day, concept testing using paper and electronic prototypes and beta tests, and building based on user feedback. Finally, quantitative evaluation with a questionnaire took place after installation.

The users' requirements were surprising, including not just information but stimulation: people expected to enjoy their transaction, for example by receiving a coupon immediately. People were afraid of security breaches.

Dr Edward Pearson, Chief Architect, Quidnunc, spoke impressively on a 'Scenario Modelling' Technique for E-Commerce Requirements Capture and Design. He said it was a practical technique which Quidnunc used very successfully. The firm aimed to create new household names for its clients by developing effective E-Commerce websites: it had never built static 'brochureware' sites. The whole lifecycle never lasted more than 6 months and usually less than 1 month; 'heartbeat' projects took 1 week! Requirement / scope creep was absolutely forbidden in such a context. In a memorable case, a legal firm had supplied a 1000 page specification, the product of a million-pound study. Quidnunc modelled their scenarios and produced a 3-page list of requirements, closely prioritized, and built the whole system in 6 months.

With traditional requirements, users felt 'this is my only chance'. Scenario modelling created a list of system behaviours, based on what people wanted to do. Each real-life scenario consisted firstly of a goal – the analysis part; and secondly of an action sequence – the design part. For

instance, 'admin need to register post': the structure was always <named user class> wants to <do something>. The first step was to capture scenarios, and the second step was to prioritize the associated requirements, deleting all that did not trace back to a business objective. The key guideline was to talk to real users, not to believe hearsay. Some useful questions to elicit scenarios were 'what do you do in a typical day?' and 'what is annoying about that?'. Scenarios should not be abstract or vague: all should be 'storyboardable' and have a definite endpoint. They should also not presuppose any specific solution or system feature.

Prioritization consisted simply of scoring each requirement for benefit (impact) and cost (ease/difficulty of implementation). This was a cliché, but it worked, said Pearson. The resulting high-value scenarios were useful for user interaction design, project scoping, testing, and even for package selection.

Benefits included a common language for developers and customers – far better than trying to explain a data model to users!; it was much cheaper; there was an unbroken chain (of traceability) from business objectives to a finished system; and there were spin-off improvements to the user interface, help text, demonstration versions, and user acceptance.

Sara Jones spoke on a conceptual framework for requirements for E-commerce, based on her recent work on the TRUST-EC project at the European Joint Research Centre, using the examples of a virtual hospital and Ruoka.net, a Finnish grocery E-business. It had, remarkably, neither its own warehouses nor its own distribution system; everything was achieved by business partnerships. There were evidently 3 classes of stakeholder: participating, enabling, and supervising; for example, suppliers, IT providers, and legislators. Requirements could perhaps be prepared generically for E-Commerce, under headings such as stakeholders (exclude bogus traders); digital information (confidentiality); infrastructure (services, systems); and the environment (legal, business).

Ian Alexander informally conducted a totally unscientific survey of the E-commerce activities and attitudes of those present (47 people, not all RESG members). 89% had conducted at least one E-commerce transaction such as buying a book from Amazon. 72% still felt concerned about E-commerce's security. 28% had used E-banking, while only 23% had ever used an E-kiosk. This might mean that engineers are, as one might expect, early adopters, but that even this group is far from feeling comfortable with all aspects of E-commerce.

Linda Macaulay finished off the formal proceedings with an interactive session gathering issues for requirements for E-commerce from the audience. These included:

- systems integration
- common E-business terminology (groans from audience)
- standards

- quality requirements
- how to balance novelty against comfort
- E-Commerce benefits
- E-Commerce packages
- E-Commerce providers (turnkey / outsourcing)

Was there anything really new in E-Commerce for requirements engineering? **Edward Pearson** thought that speed of change was the only thing. It would completely

change the business world through its speed of propagation of new ideas. In a final unscientific survey, 83% of those present agreed with him that E-Commerce would bring change as big as the Industrial Revolution.

The group adjourned happily for champagne and birthday cake, though some of us felt that lemonade and jelly might have been more appropriate for a 5-year old's birthday party. Watch this space for the RESG's 10th birthday!

RE-Papers

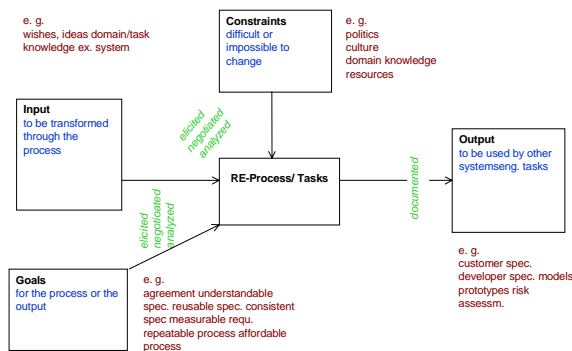
The Richness of the Requirements Engineering Process

By Barbara Paech¹ and Stephanie Lindstaed²
¹FHG IESE, ²DaimlerChrysler

Editor's Note: The following is a summary of the experiences of the participants at the working group discussing the "Requirements Engineering Process", which was part of a recent Dagstuhl Workshop, number 99241, on Requirements Capture, Documentation and Validation - <http://www.dag.uni-sb.de/ENG/>.

The Scope of the RE-Process

The participants of the working group had quite diverse backgrounds and interests. Therefore, we started out by creating a common view on the RE-Process. The result of this effort is captured in the following picture:



The RE-process (and each individual task within that process) is driven by inputs, goals, and constraints and produces some output. The driving forces can be distinguished into forces on the solution space that are transformed through the process (inputs, e.g., customer wishes, existing systems), forces on the solution space that are impossible or difficult to change (constraints, e.g., politics, system type) and forces on the RE-process and its outputs (goals, e.g. agreement, reusable process, consistent specification). The domain knowledge, for example, is partly input (like user tasks knowledge) and partly

constraint (like physical laws). All these driving forces are typically not evident at the beginning of the process, but have to be elicited, negotiated and analyzed within the process.

Depending on the forces and the required output, RE-processes have quite different characteristics. So, a process starting with a vague (product) idea of the customer and aiming at a detailed customer specification will typically be a mutual learning process between customer and developer, dominated by elicitation and negotiation tasks, while a process driven by a quite detailed customer specification (developed by the customer or in a separate project) aiming at a developer specification is typically dominated by analysis. Typically, a contract separates these two types of processes. However, even without a contract there is an important kind of border between the two processes due to the effects of change. Change of requirements is easier in the first part than in the second. Typically the detailed customer requirements reach a state of complexity and settledness that changing one or more of them requires a lot of effort, time, and money. Thus, on that border the willingness of the RE-participants to accept change decreases.

Examples of Industrial RE-Processes

Based on this common understanding we looked at specific instances of industrial RE-processes and their problems. One problem we discussed for several hours was the situation of pre-development and series-production groups at DaimlerChrysler. A pre-development group typically gets a new, vague idea from research (e.g. avoid sliding of the car when braking). The task of pre-development is then to build a first prototypical system that can work in a car. Afterwards the prototype is given to a series-production group. The task of this group is now to re-implement the system cleanly and to consider the additional constraints the system has to meet like specific control units used, etc. The problem is, that much of the design rationale of the prototype is lost when it is given to series-production. So the engineers basically have to re-engineer the prototype. The question is now: how is it possible to capture the important design decisions for reuse in series-production without hindering the creativity and spontaneity of the pre-development engineers who develop the prototypes?

During the discussion 4 ways of dealing with this problem surfaced:

- (1) The series people could elicit the design rationale from the pre-development group after the prototype has been constructed.
- (2) The pre-development people should document each design decision during the prototype development.
- (3) Mix people from pre-development and series-production: send some people from pre-development with the prototype to series-production, and /or have some series people already involved in the pre-development.
- (4) Combine all three approaches: people in pre-development collect some notes on their rationale, reflection workshops are held periodically in which pre-development tries to make their rationale and experiences explicit. On such workshops people from series-production might be present (especially when the prototype becomes more and more mature) and should document the knowledge elicited. The intent is to use the pride of the pre-developers to motivate them to talk about their experiences. The whole process of the reflection workshops should be driven by the informational needs series-production has.

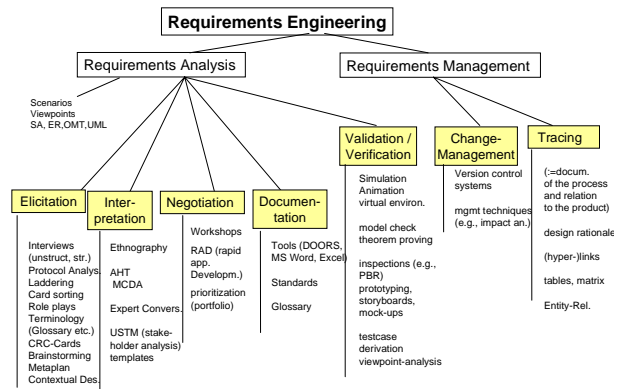
Obviously, options 1 to 3 by themselves are not feasible. However, the combination of them described in option 4 (which could be varied in several ways) tries to combine their merits in a most helpful way.

Another pressing problem DaimlerChrysler faces is the question of how to deal with quality requirements. Already in his presentation on the first day of the seminar Kurt Schneider (DaimlerChrysler) illustrated how quality models can be used. However, this approach is not entirely satisfying and the hope was to get pointers to research work dealing with this problem. Discussing this topic it became apparent that there is no ready to use approach available. In order to break typical quality requirements like usability and reliability down into testable requirements a lot of domain knowledge is required. This knowledge needs to be collected and reused so that it can already be applied in early RE-process phases like in the creation of the contract.

RE-Techniques

Having explored the problem dimension of RE-processes we started on the second day from the solution viewpoint and looked at different techniques applicable during the RE-process. The following picture shows a quite impressive list of techniques for the different RE-tasks. This picture is far from being complete. It only reflects quick brainstorming of

the participants.



We tried somewhat harder to identify techniques for handling non-functional requirements, but besides general techniques for capturing, structuring, and documenting knowledge (in particular quality features) and on resolving conflicts between them nothing much seems to be around. A conjecture was made that the reason for this is the general attitude of treating non-functional requirements as being secondary to the functional ones. The discussion on techniques also made clear that the most fundamental skills of a requirements engineer are communication skills.

The picture also shows that some techniques are more general than others - the most general ones are listed separately. These techniques can be applied to many tasks, but they still have aspects which make them particularly suited for a specific tasks. So, e.g. scenarios are certainly a way to document requirements, but their main benefit is in the elicitation tasks. Also, analysis methods like SA are mainly sold as documentation means. However, there are specific usages of these techniques, for example they may be used for elicitation purposes.

While it is quite easy to give a rough categorization of the techniques according to the RE-tasks and the forces mentioned above, we found it quite hard to describe the situations in detail in which a particular technique is of most benefit. Thus, it is difficult to give general recommendations for techniques to industrial requirements engineers, since the choice depends on many implicit factors.

RE-Process Improvement

Comparing this result with the result of the problem discussions from the first day, it seems that improvement of industrial RE-processes is achieved by first designing an appropriate organizational solution and only then filling out the technical details based on the specific factors of the company.

The working group ended with a characterization of the major achievements of the RE-community in the last years. The participants felt that important points are the focus on elicitation, the shift from requirements modeling to

requirements design and the overall increasing recognition of the importance of RE in industry.

Also, it was emphasized that Requirements Engineering and Software and Systems Engineering are getting reunited in several ways:

- there is an integration of architecture and design issues in the requirements specification,
- a lot of S(W)E-tasks which one would normally not view as RE-tasks (e.g. evaluating project descriptions) contribute to the requirements engineering process and benefit from RE-techniques,

- a lot of outputs from the RE-process can be used for other S(W)E-tasks than only design (e.g. using scenarios for test case generation),
- requirements engineering is a continuous process within the S(W)E-process and does not end with the delivery of the software system created.

Altogether, the group agreed that there is a rich set of issues and techniques associated with the RE-process which become more and more valuable for systems and software engineering as a whole.

***CORE*-Blimey!**

Geoff Mullery will be back with us in the next issue.

***RE*-Publications**

Book Review "Practical Software Requirements" by Benjamin Kovitz, Manning Publication Co., USA, 1999 (ISBN 1-884777-59-7)

Reviewed by Bashar Nuseibeh, Imperial College

I am a great fan of Michael Jackson's seminal book on "Software Requirements and Specifications: a lexicon of practice, principles and prejudices" (Addison Wesley, 1995). However, I have often been asked by practitioners: "how do I apply the Jackson's notion of problem frames in my own work?". Benjamin Kovitz's new book on "Practical Software Requirements" addresses this question, and, in my view, does it very well.

Kovitz focuses very much on the writing of requirements and specification documents, rather than on any of the other wide range of processes and products of requirements engineering, such as elicitation or analysis. His argument is that all the topics of requirements engineering cannot be addressed in a single book, and so he focuses on the process designing requirements documentation. Kovitz's emphasis that the development of requirements documents is a design process is repeated throughout the book, to underline his view that the term "engineering" is quite rightly associated with the requirements development process. It has to be said, however, that the focus of the book is very much on software requirements engineering, and systems engineering issues are left largely untreated.

The book is innovatively divided into five parts: Groundwork, Content, Style, Organisation and Examples.

The first part (six chapters) provides the foundations for Kovitz's approach to writing requirements. It begins with a chapter reviewing the problem solving process, making a particularly strong (and convincing) argument against functional decomposition. The focus then switches to the process of problem definition, which Kovitz regards as moving from what he calls "open-ended problems" to "well-defined problems". An open ended problem is any situation for which some improvement may be possible but no definite criteria for measuring this improvement are possible (this reminds me of "wicked problems" - a phrase coined by Rittel and Webber in 1973, to describe complex problems for which no definite solution can be identified). A well-defined problem on the other hand is a set of criteria according to which a proposed solution either definitely

***RE*-Bites...**

It is necessary for technical reasons that these warheads be stored upside down; that is, with the top at the bottom and the bottom at the top. In order that there may be no doubt as to which is the bottom and which is the top, it will be seen to that the bottom of each warhead immediately be labeled with the word TOP.

(From a British Admiralty Directive 1965)

solves or fails to solve that problem (this also reminds me of Suzanne and James Robertson's notion of "fit criteria", as described in their recent book on "Mastering the Requirements Process", Addison Wesley, 1999). A key message from the discussion in this chapter is that generalised problem-solving methods are insufficient to base a method of requirements writing on them. Indeed, it is on this premise that the remainder of this part of the book is based: requirements writing should be informed by existing problems and design patterns.

Chapter 3 is definitional and is based heavily on Jackson's book. The chapter distinguishes the problem domain (that part of the world where the computer is to produce effects), the machine domain (the computer and its input/output devices), and the software requirements (the effects that the machine is to exert in the problem domain by virtue of its programming). It also defines a software specification as a design of how a machine's input/output devices are to behave (an interface design) and a program as configuration of a machine that determines the machine's behaviour.

Chapter 4 also draws heavily on Jackson's work on problem frames. Framing a problem is about characterising it in a "definite form, with definite parts and definite relations between the parts". Framing a problem is also useful for eliciting requirements ("researching" a problem to use Kovitz's terminology: there is "no way to do systematic research without specific questions to ask"). In this chapter, Kovitz revisits the notion of domain and shared phenomena between domains. He also introduces the frame notation, used to represent different kinds or domains and their relationships, thereby providing a means of representing problem frames.

Chapters 5 and 6 finally give concrete examples of five problem frames (chapter 5) and so-called multi-frame problems (chapter 6). The objective of these chapters is to give practitioners real examples of typical problems "types" that they are likely to encounter, and an approach for articulating these problems. For the record, the five problems (frames) described are: Information, Control, Transformation, Workpiece and Connection.

Moving on to Part 2 of the book (Content), chapter 7 revisits the software development process, and argues for more attention to be focused on the writing of better requirements and specification documents (including user-interface design documents). Chapter 8 then distinguishes and describes the kinds of contents that should go into two kinds of documents: requirement documents and specifications. Chapters 9, 10 and 11 then provide more detailed descriptions of the kinds of information that needs to be documented, and discusses diagrammatic and other notations for representing these kinds of information. So, for example, chapter 9 describes classes and relations (although there is an interesting aside in chapter 12 on why object-orientation does not have much to offer the

requirements writer). Chapter 10 discusses sequences and events, and describes some notations in common use (such as BNF, flow charts and state-transition diagrams). Chapter 11 elaborates on this, discussing specific issues of representation causation and control.

I believe that the reader interested in formal representations and formal reasoning is unlikely to be satisfied with Kovitz's treatment of many of the above topics. The author is quite explicit in saying that the use of formality for the purposes of, say, automated analysis is beyond the scope of this book, and readers are referred to the literature on "formal method" for a more rigorous treatment.

Chapter 12 concludes part 2 of the book by addressing a number of miscellaneous issues ("special topics"). These include elicitation, object-orientation (see above), use-cases and feature interaction to name a few. The chapter makes for an interesting read, although the somewhat brief treatment of these important topics can leave readers unsatisfied and occasionally dissenting with the views expressed. For example, given the book's focus on documentation, I would have liked to have read more on how existing documentation can be used for requirements elicitation. I also found the treatment of use-cases somewhat superficial. Kovitz hints that the use of use-case is akin to "hacking around", to which many of those advocating the systematic use of use-cases would object strongly.

The final two parts of the book are the shortest, but are also the icing on the cake. They are crammed full of heuristics, guidelines and examples for practitioners to use when writing requirements documents (or any kind of document for that matter). Part 3 (Style) contains three chapters that address guidelines for writing reader-friendly documentation (chapter 13), the organisation of the documentation for easy navigation, access and recall (chapter 14), and a wide variety of helpful tips relating to the small details of documentation (chapter 15). So, for example, the reader will learn to avoid Doubletalk and Duckspeak, will think about the implications of "The Magical Number Seven Plus or Minus Two", and will untangle the differences between words like "comprise", "consist" and "compose".

Part 4 of the book (Examples) contains two chapters, each describing a real example of a requirements document taken from projects the author has been involved in as a practitioner. Chapter 16 describes the requirements for a Bug Log program for tracking bugs as people discover and fix them during software development. Chapter 17 describes the user interface specification of the same system. While the examples are relatively small, they are not toy systems and have not been edited for perfection. They serve to illustrate many of the concepts and techniques presented in the book, and as a template for practitioners to learn from and adapt. Both chapters include some commentary by the author to guide the reader.

I found Kovitz's book a pleasure to read. I believe that it is a "must buy" for all software development practitioners with a serious intent on applying both novel and established techniques for requirements writing. It does not try to cover many worthy requirements engineering topics and practices, but instead focuses on presenting a pragmatic approach to solving the very real problem of expressing software requirements. In doing so, it provides a valuable service to

the community and is an excellent addition to the requirements engineering literature.

RE-Calls

Recent Calls for Papers

ICSE 2000 – The New Millennium: 22nd International Conference on Software Engineering, Limerick, Ireland, June 4-11, 2000

<http://www.ul.ie/~icse2000>

Scope:

The revolution in information technology has been the most striking development of the latter half of this century. The astonishing growth of this technology has brought with it major changes in the daily lives of much of the world's population - affecting how we work, learn, entertain, communicate, and interact in social groups at all levels, including within our families. The advent of the information society can be seen as reducing the impact of economic, cultural, social and physical differences. In doing so, it raises hopes of increased equality of opportunity. Appropriate software development can play a vital role in realising these hopes, by facilitating more universal access to information channels.

The focus of ICSE 2000 is on defining the research agenda for the new millennium. Research inevitably concentrates on providing new, advanced functions in a timely fashion. As this new technology is implemented, however, there is a growing need for its software components to be usable, dependable, adaptable and affordable. We must seek to facilitate the transition of technology from research into practical applications, while tightening up the feedback loop from practical experience into research. We also need to ensure continuous professional development for software engineers.

ICSE 2000 marks the starting point of a response to these challenges. In its aim to be the foundational conference on software engineering for the next millennium, ICSE 2000 will bring together researchers and professionals from across the globe in a diversity of forums. These include the main conference, tutorials and workshops. The main conference will feature refereed technical

papers, invited industry presentations, panels and leading keynote speakers.

Important dates:

Deadline for workshop proposals: 1 October 1999
Deadline for papers: 11 November 1999
Notification to authors: 15 July 1999
Final version of accepted papers: 25 August 1999

ICRE 2000 – 4th IEEE International Conference on Requirements Engineering, Schaumburg, Illinois, USA, June 19-23, 2000

<http://www.cse.msu.edu/ICRE2000>

Scope:

ICRE 2000 is the fourth in a biennial series of conferences aimed at bringing together practitioners and researchers to discuss software requirements-engineering-related problems and results. As a technology transfer conference, ICRE2000 is designed to provide (1) practitioners with an evaluation of promising requirements research and practice and (2) researchers with an exposure to real-world requirements problems.

Submissions:

PAPERS: Authors are invited to submit papers addressing theory and/or practice. (The web page contains details and suggested topics.) Full-length papers are limited to 6000 words, typeset with enough room for comments by reviewers. They should include a short (150-word) abstract, a list of descriptive keywords, specification of the paper's category (experience/ research), and complete contact information for the lead author.

PANELS: Proposals that focus on requirements engineering controversies are encouraged, especially those that highlight the gulf between requirements research and practice. Preference will be given to panels that present a diversity of views on the topic chosen. Proposals should include the panel's title, a brief description of issues to be debated, the names of prospective panel members, and a description of their roles.

TUTORIALS: Tutorial proposals should consist of at least the following information: title, instructors' names and affiliations, abstract (to be used for conference program should the tutorial be accepted), intended attendee, in particular industrial or academic or both, assumed background of attendee, what the attendee will learn, outline of topics covered, when and where was this course given last, if it is not new.

Important Dates:

Deadline for Papers: September 1, 1999
Deadline for Panel Proposals: September 1, 1999
Deadline for Tutorial Proposals: October 1, 1999
Author notification: November 30, 1999
Deadline for camera-ready papers: January 30, 1999

ICSR 2000 – The Sixth International Conference on Software Reuse, Vienna, Austria, June 27-29, 2000

<http://icsr6.isys-e.uni-klu.ac.at/icsr6>

Scope:

Software reusability, the use of knowledge or artifacts from existing systems to build new ones, is a key software engineering technology important both to engineers and managers. Reuse research has been very active. Many organizations have reported reuse successes, yet there are still important research issues in systematic reuse. There is a need for reuse solutions that can be applied across domain and organization boundaries. The conference will consist of technical presentations, parallel working groups, plenary sessions, demonstrations, and tutorials on:

- Application of reuse metrics
- Domain analysis and engineering
- Case studies and experiments
- Software architecture
- Copyright and legal issues
- Formal methods
- Current issues in reuse libraries
- Design and validation of components
- Distributed components
- Object-oriented reuse
- Language design for reuse
- Patterns for software design and composition
- Re-engineering for reuse
- Integration frameworks
- Reuse education and training
- Generators and transformers
- Organizational issues

Important Dates:

Submission deadline: October 1, 1999
Acceptance notification: January 23, 2000
Camera-ready copy deadline: March 13, 2000

Submissions:

For completed papers or panel proposals, send six hard copies to:

Bill Frakes
ICSR6 Program Chair:
Computer Science Department, Virginia Tech
7054 Haycock Rd. Falls Church VA 22043
wfrakes@vt.edu <http://frakes.cs.vt.edu>

Organisation:

General Chair: John Favaro, favaro@pisa.intecs.it
Program Chair: William B. Frakes, wfrakes@vt.edu

OOPSLA '99 Workshop on Use Case Patterns, Venice, November 25-26, 1999

Scope:

Use cases are a popular method of requirements modeling, however, people have trouble creating meaningful use cases. Common problems include how to start use case analysis, how to decide how much detail belongs in a use case, and how to organize use cases.

Several processes and templates exist for developing and documenting use cases. While these solutions are helpful and document valuable experience, they may still be difficult to use because each development organization is different. Each targets a different industry and product suite, each uses different processes, and each has its own culture.

Patterns provide a mechanism for capturing people's experiences and documenting their solutions in a manner that is adaptable to different situations. A pattern language is a collection of cooperating patterns that work together to provide a solution to a problem, yet are flexible enough to work in widely disparate development environments. Thus, pattern languages provide one solution to the problem of describing generic methods for capturing use cases.

Participants at a workshop at OOPSLA 98 produced the beginning of a Use Case Pattern Language. Developing use cases takes on many facets, and requires a great deal of knowledge, much more than any one session can capture. The goal of this workshop is to share experiences in use case development and to discuss and refine a Pattern Language for the development, documentation and use of use cases.

We strongly encourage participation from anyone with an interest or experience in requirements modeling and/or use case development. Interested participants should submit the following:

1. Name, email address and affiliations.

2. A brief background including short a description of experience or interest in use cases and/or requirements analysis; and
3. One of the following:
 - a critique of the evolving language
 - suggestions for improvement of the existing patterns and/or
 - additional patterns that fit into the pattern language.

Participants will be evaluated on the basis of these submissions.

Submissions:

Paul Bramble
AG Communication Systems
2500 W Utopia Road
Phoenix, AZ, 85027
Fax: (623)581-4390

Important Dates:

Submission deadline: September 7, 1999
Acceptance notification: September 13, 1999

CAiSE 2000 – 12th Conference on Advanced Information Systems Engineering, Stockholm, Sweden, June 7-9, 2000

<http://www.ul.ie/~icse2000>

Scope:

Vast improvements in communication technology, including the increasing use of Internet and WWW have, over the last ten years, led to better communication and easier information access in general. In addition, it has initiated changes in the way organizations cooperate and trade. Hence, we see the formation of virtual enterprises where smaller companies collaborate to solve tasks they cannot cope with alone. It also leads to improvements in information services to people as consumers and as citizens. However, for such information services in a digitized world to become really useful it is required that they add value to the workings of businesses and to people in their daily lives. This calls for enhancements of methods and tools in order better to understand organizations and the way they interact among themselves and with people. Also, there is a need for enhanced design methods for services that are truly value adding and that function well together with the organizations' legacy systems. In consequence, we are inviting papers and tutorial proposals that address these issues from a wide range of perspectives but with a common goal. The goal is to advance theory and practice of analyzing and designing individual but integrated information services in an organizational context.

In addition to the special theme, topics relevant for submissions to CAiSE*00 include, but are not limited to, the following:

- Business process assessment and improvement
- Business process models and processware
- Component based development
- Data warehousing
- Electronic commerce
- Enterprise Resource Planning
- Information services on the net
- Information systems and services to the citizen
- IT solutions for learning organisations
- Knowledge discovery in databases
- IS strategy and planning
- Managing organisational knowledge assets
- Methodologies and models for IS
- Metrics and assessment
- Novel IS architectures (intelligent, cooperative, distributed, multimedia, open, ...)
- Patterns (organisational, analysis, design, reuse)
- Requirements engineering
- Systems procurement processes
- Virtual enterprising

Submissions:

Papers: The conference organizers solicit papers within, but not restricted to, areas indicated by the topics mentioned above. Papers should not exceed 5,000 words. They must be original, and not submitted to, or accepted by, any conference or journal.

Panels and Tutorials:

Proposals are solicited for tutorials that will be held together with the conference. Tutorials can be half day (3 hours) or full day (6 hours) and should be of interest to participants from the academic and business communities. The conference program will also include a few panels on current topics about which there is controversy and thus a need for public discussion.

Workshops:

We invite prospective workshop organizers to submit proposals stating the topic and goals of the workshop. Workshop topics will be in line with the conference topics. Proposals will be reviewed by the Organising Committee. All participants are expected to attend the main conference.

Posters may also be submitted.

Important Dates:

Workshop proposals: October 30, 1999
Papers, panels and tutorials: November 30, 1999
Posters: March 1, 2000

Workshop acceptance: December 1, 1999

Paper, panel and tutorial accept.: February 1, 2000
Poster acceptance: April 1, 2000

Organisation:

General Chair

Eva Lindencrona, SITI (Swedish Institute for Information Technology), P.O. Box 1243, SE-164 28 Kista, Sweden, E-mail: eva@siti.se, phone: +46 8 752 16 36, fax: +46 8 752 68 00

Program Chair

Benkt Wangler, CAiSE*00, Department of Computer and Systems Sciences, Stockholm University/KTH, Electrum 230, SE-164 40 Kista, Sweden, E-mail: benkt@dsv.su.se, phone: +46 8 16 49 47, fax: +46 8 703 90 25

RE-Sources

For a full listing of books, mailing lists, web pages and tools that have appeared in this section in previous newsletters, see the RQ archive: <http://www.soi.city.ac.uk/homes/gespan/rq/rq.html>

Web Pages

Please note that the web address of the RESG is now: <http://www.cs.york.ac.uk/bcs/resg/>

CREWS web site:

<http://sunsite.informatik.rwth-aachen.de/CREWS/>

An interesting collection of 72 papers (!) and a description of an ESPRIT project on co-operative requirements engineering with scenarios. The CREWS project has developed two prototypical tool suites which can be employed, e.g., as extensions to the Use Case approach in object-oriented systems engineering. One shows traceable multimedia-based current-state analysis and animation of future scenarios, the other provides

guidance for the creation and analysis of text scenarios and for the systematic generation of exception scenarios.

Requirements Engineering, Student Newsletter:

http://www.cc.gatech.edu/computing/SW_Eng/resnews.html

IFIP Working Group 2.9 (Software Requirements Engineering):

http://www.cis.gsu.edu/~wrobinso/ifip2_9/

Mailing lists

The SRE list

The SRE mailing list aims to act as a forum for exchange of ideas among the requirements engineering researchers and practitioners. To subscribe to SRE mailing list, send e-mail to listproc@jrcase.mq.edu.au with the following line as the first and only line in the body of the message:

subscribe SRE *your-first-name your-second-name*.

RE-Actors

The committee of RESG

Chair: Dr. Bashar Nuseibeh, Department of Computing, Imperial College, 180 Queen's Gate, London SW7 2BZ, UK. E-Mail: ban@doc.ic.ac.uk, Tel: 0171-594-8286, Fax: 0171-581-8024

Treasurer: Dr. Neil Maiden, Centre for HCI Design, City University, Northampton Square, London EC1V OHB, UK. E-Mail: N.A.M.Maiden@city.ac.uk, Tel: 0171-477-8412, Fax: 0171-477-8859

Secretary: Dr. Wolfgang Emmerich, University College London, Department of Computer Science, Gower Street, London WC1E 6BT, UK. E-Mail: W.Emmerich@cs.ucl.ac.uk, Tel: +44 171 504 4413, Fax: +44 171 387 1397

Membership Secretary: David Shearer, Department of Computer Science, University of Hertfordshire, Hatfield, AL10 9AB, UK.

Newsletter Editor: Dr Peter Sawyer, Lancaster University, Department of Computing, Lancaster, LA1 4YR, UK. E-Mail: sawyer@comp.lancs.ac.uk, Tel: +44 524 593780, Fax: +44 524 593608.

Associate Newsletter Editor: Ian Alexander, 17A Rothschild Road, Chiswick, London W4 5HS. E-Mail: iany@easynet.co.uk, Tel: 0181-995 3057

Publicity Officer: Carol Britton, Department of Computer Science, University of Hertfordshire, College Lane, Hatfield, UK. AL10 9AB, E-Mail: c.britton@herts.ac.uk, Tel: 01707 284354, Fax: 01707 284303

Associate Publicity Officer: Dr. Vito Veneziano, Department of Computer Science, University of Hertfordshire, College Lane, Hatfield, AL10 9AB, UK. E-Mail: v.veneziano@herts.ac.uk, Tel: 01707 286196.

Web-Master: Dr. Laurence Brooks, Department of Computer Science, University of York, York, YO10 5DD. E-Mail: Laurence.Brooks@cs.york.ac.uk, Tel: 01904 433242.

Industrial Liaison Officer: Dr. Barbara Farbey, Department of Computer Science, University College London, Gower Street, London WC1E 6BT, UK. E-Mail: B.Farbey@cs.ucl.ac.uk, Tel: +44 171 419 3672, Fax: +44 171 387 1397.

Members-at-Large:

Dr. Steve Easterbrook, Department of Computer Science, University of Toronto, 6 King's College Rd, Toronto, Ontario, M5S 3H5, Canada. E-Mail: sme@cs.toronto.edu.

Dr. Sara Jones, Department of Computer Science, University of Hertfordshire, Hatfield, AL10 9AB, UK. E-Mail: S.Jones@herts.ac.uk, Tel: 01707 284370, Fax: 01707284303

***RE*-Creations**

To contribute to RQ please send contributions to Peter Sawyer (sawyer@comp.lancs.ac.uk). Submissions must be electronic form, preferably plain ASCII text. A list of the kinds of contributions we welcome can be found on the web at:

<http://www soi.city.ac.uk/homes/gespan/rq/contributions.html>