



Requirenautics Quarterly

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

<http://www.resg.org.uk>

©2003, BCS RESG

Issue 29 (August 2003)

RE-Locations

<i>Editorial</i>	1	<i>RE-Papers</i>	8
<i>Chairman's message</i>	2	<i>Requirements Engineering as if Stakeholders</i>	
<i>RE-Treats</i>	2	<i>Mattered</i>	8
<i>Tom Gilb presents 'Planguage' a planning language</i>		<i>Library of Patterns for Interactive Applications</i>	10
<i>for requirements and more</i>	2	<i>RE-Publications</i>	11
<i>Creativity, the Path to Innovative Requirements</i>	2	<i>Designing Information Spaces: The Social</i>	
<i>Requirements training workshop</i>	3	<i>Navigation Approach Approach</i>	11
<i>Mastering the Requirements Process</i>	3	<i>Visualizing Argumentation Software Tools for</i>	
<i>Extending Requirements: A Practical Workshop</i>	3	<i>Collaborative & Educational Sense-Making</i>	11
<i>RE-Calls</i>	4	<i>User-Centered Requirements The Scenario-Based</i>	
<i>11th IEEE International Requirements Engineering</i>		<i>Engineering Process</i>	14
<i>Conference (RE'03)</i>	4	<i>RE-Sources</i>	15
<i>3rd International Conference on Aspect-Oriented</i>		<i>Mailing lists</i>	15
<i>Software Development</i>	4	<i>RE-Actors</i>	15
<i>RE-Readings</i>	5	<i>The committee of RESG</i>	15
<i>COTS Integration: Why You Need Requirements</i>		<i>RE-Funds</i>	16
<i>Workshop on Requirements Engineering for</i>		<i>Minutes of the 9th Annual General Meeting of the</i>	
<i>Business Process Support (REBPS'03)</i>	6	<i>Requirements Engineering Specialist Group of the</i>	
<i>Stakeholders Without Tears: an exploration of</i>		<i>BCS</i>	16
<i>project sociology</i>	8		

RE-Soundings

Editorial

This issue includes two excellent articles by RE practitioners that illustrate real problems that must be confronted in RE and practical measures that can be taken to ease RE problems.

Jonathan Beebe's article describes how lightweight patterns of interaction are used at Fujitsu Consulting to ease the requirements engineer's task. Jonathan describes how they help ensure that experience across problem/product types can be reused. They illustrate how an organisation can adopt practical and pragmatic mechanisms for maturing it's RE processes.

Richard Veryard writes of the relationship between stakeholder roles and requirements. In particular, he shows that, as projects progress, roles change and impact the requirements. This challenges many of the assumptions that appear, often tacitly, in naïve views of the RE process. The identification of roles/actors/viewpoints/etc. is of course still crucial. What we mustn't do is assume that these provide a

perfectly firm foundation from which to start teasing out requirements and making trade-offs. Even supposing we can do this early in the project, as the project progresses the orthodoxy says that we need to be able to trace the requirements back to their sources. Again, the tacit assumption is often that while the requirements may be volatile, their sources are somehow more concrete; they'll still be there, unchanged, if only we can recover them. In fact organisations and markets may be as volatile as their requirements. It's what makes our lives so interesting and it's why RE is going to be around for quite some time!

RE'03 is about to kick off in Monterey, California. Veterans of RE will be able to enjoy comparing how the discipline has advanced during the decade since the seminal RE'93. This issue also contains a call for submissions to Aspect-oriented software development (AOSD'04). People are beginning to look at if and how aspects help model widely-scoped requirements. AOSD'04 is likely to offer an opportunity to see just where the RE and AOP communities intersect. Readers

wishing to find out more about 'early aspects' might like to visit <http://www.early-aspects.net>.

Pete Sawyer
Computing Department, Lancaster University.

Chairman's message

The last 12 months have seen RESG events attended by record numbers of participants and pre-registrations have had to close before the events take place. The hot summer has not had any perceptible effect with the Stakeholders workshop and AGM presented to a full house at University College London in July.

Attendance at RESG continues to be free to all RESG members, however we are re-introducing a £5 entry fee for non-members to give members priority access to events and to emphasise the benefits of membership.

The Annual General Meeting (AGM) saw an expanded RESG executive committee elected to run the Specialist Group for another year. Thank you for your support!

I would like to extend a warm welcome to new committee members Gordon Woods (Airbus) and Carina Alves (UCL). Gordon joins the Industrial

Liaison team and brings with him his considerable experience in large systems development projects. Carina, a PhD student at UCL, joins as Student Liaison Officer, a new role we have introduced to the committee to focus on promoting and supporting research student needs.

After a tremendously successful year publicising the RESG, Juan Ramil has moved to Membership to apply his magic touch there. Wolfgang Emmerich, after many years as the group's trusted Secretary, has decided to move on, but will continue to contribute to as a member of the Industrial Liaison team. David Bush takes over as the RESG's new Secretary.

I am writing this introduction a week before RE'03 in Monterey, and I expect that in the next newsletter we will be reporting on a packed conference week. There will be two outstanding RE events closer to home to report on too. Tom Gilb's seminar is on the 17th September, and Neil Maiden and Suzanne Robertson's creativity workshop is on the 9th October. Hope to see you there!

Bashar Nuseibeh
The Open University

RE-Treats

*For further details of all events, see www.resg.org.uk
Next event organised by the group.*

Tom Gilb presents 'Planguage' a planning language for requirements and more

Date: 17th September 2003, 2.00 pm
Location: Room 418, Huxley Building, Imperial College, London
Contact: Sebastian Uchitel
(s.uchitel@imperial.ac.uk)

Requirement(s) Engineering is arguably the most critical single discipline in systems and software engineering. Bad requirements are cited as a main cause of large systems engineering project failures (P. Morris, *The Management of Projects*, Telford, London 1994). Too little investment in requirements is proven as a cause for software projects and systems engineering projects (NASA, IT survey). I would argue that the theory and teaching of requirements today is in bad shape. There are several signs of this. Most so-called requirements are actually design for unstated requirements! Most requirements are a nice sounding set of words with no testable or quantified structure and very little information about the requirements and their relation to all other requirements, designs, and plans. This tutorial will open up a radically new approach to requirements that will rectify many of these problems. It is based on the ideas in Planguage a specifically invented and evolved language for requirements specification. Fundamentals are reexamined (what is a

function requirement?). All qualitative requirements are quantified richly. Ten to 30 parameters are used to describe a single reusable requirement fully and appropriately for its task. Requirements are related directly and individually to stakeholders. Requirements are looked at in the wider systems engineering context of design, testing, project management, and engineering-work quality-control.

Creativity, the Path to Innovative Requirements

Date: 9th October 2003, 2.00 pm
Location: Imperial College, London
Contact: Rosie Bower (rosieb@soi.city.ac.uk)

Where do requirements come from? The optimistic view is that the customers or users of a system tell the requirements engineer what they want. For a variety of reasons, this is rarely the case. People don't know exactly what they want, they do not know what is possible, or they cannot express their real needs, or because they think in terms of current solutions or they cannot envisage the future. Requirements engineering is increasingly a creative process in which stakeholders and designers work together to create ideas for new systems that are eventually expressed as requirements.

This half-day tutorial illustrates how we use innovative techniques to imagine, create, discover and formalise requirements. We also illustrate how creativity can be used in requirements engineering and provide a guide for running creative design workshops. This is an

interactive hands-on tutorial based on experience in running creative workshops for air traffic management systems for Eurocontrol and NATS.

Audience

Academics who want to understand more about the changing nature of RE. Practitioners who want to learn how to use creative techniques in the RE process. Project managers who want to understand more about the requirements process. Due to the hands-on nature of this tutorial the number of participants is limited to 24.

Instructors:

Suzanne Robertson: The Atlantic Systems Guild, London.

Neil Maiden: The Centre for HCI Design, City University, London.

Requirements training workshop

Date: 3rd December 2003, 1.00 pm

Location: Room 418, Huxley Building, Imperial College, London

Contact: Elena Pérez-Miñana
(elena.perez-minana@philips.com)

Currently there is an increasing interest, both in industry and academia, to tackle and solve the problems associated with the elicitation, specification and management of requirements. Notwithstanding, there is an evident lack of adequate training courses for people in industry.

This half-day event, organized by the RESG, includes presentations of leading experts in the area of Requirements Engineering (RE). Each of them will provide their point of view to the following three questions:

- Who needs RE training?
- How can RE be taught?
- What do RE practitioners need to learn?

It will present two different approaches to training in requirements; the first one comprises the conventional short courses that are available through organizations such as Learning Tree. The second one labelled "informal/ad-hoc learning" advocates learning through a communal effort following communities of practice.

The event will end with a wide-ranging discussion on RE training. It will be relevant to anyone interested in the role of requirements in their organization, in finding out the type of training their engineers will need to complete in order to do a good job in the requirements area. The event will also be of relevance to those who might already be involved with the requirements area and who would like to put forward their opinion on how it should be carried out.

The last session will be a panel discussion, involving the audience and the speakers. This will give every attendee the opportunity to raise anything of importance in the area.

Audience

This workshop is appropriate for anybody who is involved with the Requirements area, either because they work in the area, or hold a managerial role in their organization and are considering setting up a requirements group. Academics, interested in understanding the training needs for the people working on requirements in industry, will receive useful input, and will have plenty of opportunities of putting their ideas across.

Workshop outline

12.30 Registration
13.00 Introduction (Elena Pérez-Miñana)
13.10 Ian Bray
13.40 Pete Sawyer/Ian Sommerville
14.10 Panel Discussion Part I
14.45 Coffee break
15.00 Stefanie Lindstaedt
15.30 Ken Jackson
16.10 Panel Discussion Part II
17.00 Closing remarks

Other events likely to be of interest to RESG members.

Mastering the Requirements Process

Date: 6th-8th October 2003

Location: London

Contact: Jeanette Hall (Jeanette@irmuk.co.uk)

This seminar gives you a process, techniques and templates to discover exactly what your customers need and want for their systems. It shows you how to understand your users' work, and then devise the best product to help with that work – no matter what kind of work that is.

The seminar gives you the Volere requirements process and template. Then shows you how you can best use these in your own organisation to improve communication with stakeholders, and exactly discover their requirements.

Presented by Suzanne Robertson. RESG members are entitled to a 10 percent discount.

Extending Requirements: A Practical Workshop

Date: 17th-18th November 2003

Location: London

Contact: Jeanette Hall (Jeanette@irmuk.co.uk)

Taking your requirements to the next level. This workshop shows you how to build on your existing expertise and integrate requirements into project management for the maximum effect.

The last five years have seen considerable improvements in the techniques and processes we use

to gather and communicate requirements. Our Volere requirements template has been used by a wide variety of organisations all over the world. The successful application of many things like connecting analysis modelling to requirements, requirements templates, fit criteria for individual requirements, quality gateway testing techniques and trawling techniques, have highlighted other areas where we need to improve our requirements skills.

This workshop extends what you know about requirements and gives you ways of taking your projects to a higher level of requirements excellence.

In this workshop we show you ways of choosing the best set of requirements to give you a competitive edge and still get your product to market on time, this includes techniques for quantifying the business value of investing in requirements. We also address the need

to anticipate market opportunities by creating and inventing visionary requirements. And we address the question of dealing with requirements for existing systems along with techniques for managing meta projects large projects made up of a number of smaller ones.

A good requirements engineer has the skill of helping a wide variety of people communicate with each other. The project sociology structure in this workshop helps you to discover the correct stakeholders for your project. Then we give you proven techniques for helping them to be involved in the appropriate parts of the project, and even more importantly to stay involved throughout.

Presented by Suzanne Robertson. RESG members are entitled to a 10 percent discount.

RE-Calls

Recent Calls for Papers and Participation

11th IEEE International Requirements Engineering Conference (RE'03)

8th - 12th September 2003, Monterey Bay, California USA
<http://www.re03.org>

Requirements Engineering (RE) lies at the heart of software development. It is the branch of systems engineering concerned with the real-world goals for, functions of, and constraints on software-intensive systems. It is concerned with how these factors are taken into account during the implementation and maintenance of the system, from software specifications and architectures up to final test cases. RE requires a variety and richness of skills, processes, methods, techniques and tools. In addition, diversity arises from different application domains ranging from business information systems to real-time process control systems, from traditional to web-based systems as well as from the perspective being system families or not.

The conference will include a technical and an industrial paper track, with refereed papers describing novel research, industrial problem statements, experience reports and surveys. The program also includes keynote speakers, state-of-the-art and practice tutorials, and an exhibition, which includes companies with RE tools and services, book publishers, and other related exhibitors. We also will have several associated workshops, including a doctoral workshop for PhD students, as well as research demos and tools. The RE'03 conference will provide an opportunity for practitioners and researchers to share ideas and experiences, while enjoying the beautiful Monterey

with the Monterey Bay Aquarium, Fisherman's Wharf and Cannery Row. Enjoy the spectacular sights of Monterey Peninsula at Big Sur, explore Point Lobos in Carmel-by-the-Sea and visit Carmel Mission.

For general RE'03 information please contact: Carl K. Chang (c.chang@computer.org) For technical program information please contact: Roel Wieringa (roelw@cs.utwente.nl)

3rd International Conference on Aspect-Oriented Software Development

22nd - 26th March, 2004, Lancaster University, UK.
<http://aosd.net/conference>

The construction and evolution of a software system involves many different aspects including features, qualities, perspectives, use cases, collaborations, requirements, processes, and implementation structures. Aspect-oriented approaches provide new means for modularizing such crosscutting aspects of a system.

AOSD 2004 will bring together researchers and practicing software developers to discuss new ideas in aspect-oriented technology.

The conference program will feature technical papers, practitioner reports, workshops, tutorials, demonstrations, a code fest, and a student event. Keynote speakers include Dr. Daniel Sabbah (IBM Vice President, Development and SWG Technology, Application Integration & Middleware Division, Software Group) and Prof. Bashar Nuseibeh (The Open University). A panel, moderated by Dr. Brian Barry (Bedarra Corporation), will address AOP and enterprise development.

RE-Readings

Reviews of recent Requirements Engineering events.

COTS Integration: Why You Need Requirements

RESG Event, 9th April 2003, University of Central England, Birmingham.

Ian Alexander (Chairman) welcomed approximately 45 participants to another RESG experimental event. In this case, it took the form of a workshop aiming to discuss the issues associated with COTS integration, and the role played by the system's requirements therein.

The workshop included four presentations and a question-answer session at the end to wrap up the main ideas put forward during the afternoon. In the first presentation, the speaker described the experience of developing a global messaging network system, which is built through the integration of heterogeneous COTS components. This was followed by the description of two approaches, which have been designed to ameliorate the problems that appear when handling the various artifacts that are produced when building systems using COTS technology. The presenter of each development method included at the end, a brief description on how its method could be applied to build the system described in the case study. The final presentation gave the workshop participants useful input on the matters that should be solved when trying to build systems using COTS technology that satisfy the level of quality that is expected according to the relevant standards.

The introductory presentation was particularly interesting, because it was designed following a virtual format. **David Bush**, who at the last minute discovered he couldn't make it on the day, had to make use of the latest technology available to stand in his absence. It constituted a good alternative under the circumstances. The description of the case study was clearly put, although there was a sniff of frustration emanating from the audience in view of the impossibility to query the presenter at the end. The aim of the Aeronautical Fixed Telecommunications Network (AFTN) system, the case study described, is to support the exchange of messages to improve the safety, regularity and efficiency of international air navigation. The complexity of such a system demands a design and implementation showing a high degree of reusability. The existing running system includes a number of elements that must be integrated into the new design. Furthermore, safety and reliability play very important roles given the type of system under development.

The methods described in the following two presentations provided appropriate means to deal with the kind of problems David Bush and his team

encountered during the development of the AFTN system. The first presenter, **Carina Alves** described a process that enabled the system stakeholders, to deal with the problem of analysing their system requirements and discovering the components, that once integrated would provide an effective COTS system solution. It advocates a more formal, structured approach to the requirements specification process, including two strategies for managing conflicts, a composition rule-based strategy, and a modifiability strategy. These all offer good alternatives to ensure the stakeholders cover all the variables present in today's COTS solutions. The only handicap I see with the approach is that many of the systems under development cannot boast the level of clarity demanded of the requirements, a problem that is becoming more and more common in today's rapidly changing environment. The requirements outlined for the AFTN system were sufficiently clear, and this allowed Carina to apply her requirements analysis process, although it was a bit difficult to fully relate the requirements to the components available due to the lack of information found in the suppliers' specifications.

The second method, the SCARLET process, was presented by **Neil Maiden** and **Vincenzo Croce**. It is a use case based approach that provides two forms of process guidance to develop COTS-systems. The first one is goal driven and is useful in those cases in which the stakeholders have a clear idea of the requirements, techniques, and components available to build the COTS solution they are after. The second process is situation-based and is useful in those cases in which the requirements and techniques that are necessary to satisfy them cannot be pre-defined. For the AFTN system they managed to identify 6 candidate software components to satisfy the requirements specification. Nevertheless, the component information was insufficient to carry out an adequate component compliance analysis. The SCARLET method also offers its user the means to structure the phases and artefacts generated in the development of a COTS-system but, as with Carina's process, it requires a level of clarity in the system's requirements that is difficult to achieve, even though the situation-based approach might enable the users to handle the level of uncertainty associated to unclear requirements.

The final speaker, **Ljerka Beus-Dukic**, gave the workshop participants a very useful outline of the existing standards for software product quality. As Ljerka very aptly put it, there are a number of standards that software developers can use to ensure the quality of their product, independently of whether they are software components or a fully blown COTS-system. The means to evaluate product quality are out there; the difficulty lies in the fact that not many organisations have, as yet, fully integrated them into their processes.

The panel discussion organised for the final part of the workshop was further proof of the importance of putting in place adequate means to link system requirements to the components used to build the final solution. If this is properly done, it is much easier to demonstrate coverage, with the design of an adequate testing suite, thereby guaranteeing that requirements properties such as the safety constraints are fully satisfied by the solution. The type of system presented as a case study, and the final presentation on standards and software product quality, showed that there are the means to help the developers produce better systems. It is really up to the companies to ensure they apply the methods that are available, and put to good use the existing tools.

Although a lot of work has been done, I feel there is a need to work further in the task of linking system requirements to the specification of the components integrated to satisfy them. In all the discussions that took place, it was never completely clear to me how it was possible to link requirements which can only be satisfied through the integration of a number of components and not by a particular component. Because this means it is not possible to manipulate a component's specification, it is necessary to work around an "integrated" component specification.

© *Elena Pérez-Miñana*

Workshop on Requirements Engineering for Business Process Support (REBPS'03)

At CaiSE 2003, 16th - 18th June 2003, Velden, Austria.

Ilia Bider (IbisSoft, Stockholm) welcomed the participants to this year's workshop in the series on Business Processes (BP). Each year had a special focus and this year's was Requirements Engineering (RE) for Business Process Support (BPS).

Gil Regev (Swiss Federal Institute of Technology) asked Why BPS? It could help to avoid over-specialisation. Processes regulate businesses, but many of them are unstable: being rigidly prescriptive often fails as a strategy. Maybe a BPS system could enable generalist staff to do a range of jobs, helped by only a few specialists. **Ian Alexander** wondered whether that didn't smack too much of control, not only enforcing the procedures for human work, but actually doing the work too? Gil Regev replied that he had in mind instead a kind of Knowledge Management System (KMS) which pointed generalist users to the right specialist to consult on each specific problem. **Peri Loucopoulos** suggested that BPs were governed by the business situation; e.g. the Hewlett-Packard/Compaq merger created a tension between the corporate cultures, neither side wishing to accept the other's processes. **Claudia Eckert** asked what was meant by BP? She meant something that emerged from the way a business worked, whereas Gil Regev seemed to mean something imposed. Peri Loucopoulos asked

how to move towards this Utopia, given a century of increasing specialisation? Gil Regev said it was less utopian than the BP Re-engineering enterprise.

Steward Green (University of the West of England) spoke about eliciting stakeholders's knowledge of goals and processes. He tried out 4 ways – questionnaire, interview (both for explicit knowledge), contextual design (a la Beyer&Holtzblatt, for implicit knowledge), and self-observation & measurement lasting for a week (for new knowledge). The domain was a university helpdesk and the question was whether people working in the domain could articulate what they did in their work. The study found that despite the issue of tacit knowledge raised by Goguen (and deriving ultimately from Polanyi), helpdesk staff were able to articulate their knowledge quite easily. He speculated that this could be because they were well-educated. Ian Alexander suggested it could be because the domain was atypically explicit - helpdesk people after all talk through their knowledge with their clients. Gil Regev said that the 'goals' that people can state are not the same as their 'real goals' beneath the surface. Claudia Eckert asked what if goals were contradictory or incommensurable? For instance, a helpdesk might wish to be efficient, but also to make users happy.

Niko Kleiner (University of Ulm) spoke about Requirements Engineering in workflow application development – specifically in the automobile industry. The project first followed a systematic method (ARIS) religiously. The users rejected the system. The team had expected to foresee optimal BPs, and that work practice would be like the implemented workflow (!). They thought that the implementation technology (not the BP / RE approach) was the cause of failure, so they tried another technology. Another bad system resulted, and requests for service piled up somewhere inside it. They then tried to refine the system; users interpreted the workflow states differently, and requests continued to pile up. Finally, they investigated what was really happening inside the system, and found that there was a misconception about the synchronisation between two processes, so requests were bound to pile up – the system could never have worked. The mistake was resolved and the system was at last looking promising.

Why had this happened? A workflow application influences but does not determine the BP: that emerges from operational use. The two must converge, so more BP modelling was not the answer. Change was inevitable during development, so iteration was vital. Ilia Bider said the trouble lay in talking to managers not to engineers. Claudia Eckert asked why workflow had been chosen in this case, implying it was too prescriptive.

Peri Loucopoulos (UMIST) spoke about the S³ (Strategy-Service-Support) framework for BP modelling. It was commonly agreed that BP had products and customers; goals; activities; and was collaborative. But (see Ian Alexander's paper) was the RE process for BPS distinctive? Could a focus on BPS help? Was modelling any use? He said at once that 20

years' experience said that modelling was vital. He compared RE for a process like electricity industry deregulation with that for the Athens 2004 Olympic games. Unlike information systems, there often wasn't a pre-existing system to replace; there were many alternative realisations; soft issues were very important; there was much more stakeholder involvement; and processes changed rapidly over the years. The overriding purpose of RE for business processes was organisational change (hopefully improvement).

So, was qualitative modelling enough? He argued you needed quantitative modelling as well. People then saw the results and could debate their priorities in an informed way. You modelled the scenarios (a la StateMate) and defined queues and servers, and then tried it out to see what you'd get. For instance, given the initial requirements for the Olympics, it turned out you'd queue for 2 hours for a hot dog! This led to quite a Positivist (hypothetico-deductive) sort of view of the world, relating observed facts to modelled theories to reasoned deductions – making engineering like science, in fact. The limits of this approach, he said (rightly backing away from Positivism) were the bounded rationality and psychology of humans, resistance to policy change, and people's continued reliance on informal approaches. In any case, solving one problem often only meant transferring it to another place. But S³ (with simulation) did have good results with domain experts. You needed a philosophy that combined systems and operational thinking with abstract, non-linear thinking.

Claudia Eckert (University of Cambridge) spoke about planning BPs in product development organisations. She illustrated her talk with projects like developing the EH101 helicopter and jet engines. Tens of thousands of tasks involving thousands of engineers and millions of pounds were needed in such work. Supply chains were complex but few products were delivered per year (just a few hundred engines). The market was extremely political, with project decisions influenced by the size of national contributions. Designs were ridiculously complex, "almost decomposable but never quite" (Herbert Simon), and neither the designs nor the processes were ever exactly repeated.

The domain was thus chaotic, with inseparable activities, powerful social factors, multiple linked processes and resources, uncertainty and unrepeatability. Planning was by cost and lead time, etc, under heavy pressure to succeed despite technical and resource problems.

She argued that you therefore needed to visualise process models and explore their parameters. Simple state-based models weren't very helpful. Automating the entire process of design was sometimes possible, as for wing struts, but it took a lot of effort to construct such a huge expert system. Task-based models didn't work either, as most of the tasks took, errrm, 100% of the project lifetime! So you needed to combine all of

these in a 'signposting' model in which you dynamically remodelled according to the current level of refinement of the model. You used a Markov net to get a probability that the design would be improved, with strong and weak dependency links (not a crude you do A then when 100% complete you always do B, etc). This enabled you to do a critical path analysis and discover the bottlenecks. Peri Loucopoulos was delighted she'd raised the issue of processes across the supply chain, not just in one organisation. But BP was at least as complex as engineering design, and it couldn't be tested unlike a design!

Ian Alexander led a lively and wide-ranging discussion on the scope and value of BPS after the delicious lunch in the warm sleepy afternoon (no-one dozed off). He asked what BPS consisted of, what you got for your money, and whether it was worth it. It seemed to consist of BP as-is Modelling, BP as-should-be Re-engineering, BP Improvement projects, and finally BP Support using the new mechanisms. There were several bandwagons like BPM and BPR. BPS's claim was the Blairite 'joined-up thinking' which if not empty propaganda had to mean breaching the 'silo' walls between departments in place of local optimisation. This was hard because people feared changes to their jobs, because of rivalries and power politics, and so on. The cost included both one-off and recurring costs; the one-off costs were large and risky because if any link in the chain broke the whole thing failed, and because change is continuous, so bigger, slower BPM projects were inherently more likely to become outdated and fail. Perhaps smaller, faster, cheaper and less ambitious projects would be better, making for a more incremental approach to BPS. Optimal solutions were not guaranteed.

Ilia Bider led a discussion in which he argued for the value and distinctiveness of BPS. "Success depends on a majority accepting the system" – he agreed there was no need for it to be optimal. You needed to think big but start small, and evolve, much as Ian Alexander had suggested. You couldn't make a new BP-oriented system just by integrating old lumps of software: what you could do was to reuse software once you had understood and designed a new process.

Elke Hochmüller asked a lot of questions about non-functional requirements (NFRs or simply '-ilities') in BPS systems. Key NFRs included the ability to cope with new situations (flexibility, reusability, simulability, traceability); to have a trustworthy operating base (scalability, reliability, security, etc), and crucially usability. Peri Loucopoulos added designability or explorability. **Pnina Soffer** wondered if the same NFR list would apply to all systems? A discussion explored issues around the user interface like commonality and tailorability, and more general issues like affordability (initial, annual, and modifiability) and the openness (of code, etc).

Peri Loucopoulos said that BPS was here to stay, and that we should ask why, i.e. we should document the

requirements for it. Ian Alexander facilitated a brainstorming session to start the requirements off.

© Ian Alexander, July 2003

Stakeholders Without Tears: an exploration of project sociology

RESG Event, 16th July 2003, University College London, London.

Ian Alexander and Suzanne Robertson welcomed a room packed with enthusiastic attendees expectantly awaiting the RESG 2003 summer event that followed the RESG AGM. This year's episode took the form of a tutorial entitled "Stakeholders without Tears: an exploration of project sociology". It was run by the welcoming committee, who managed to keep the audience's attention all afternoon, notwithstanding the sweltering heat. The aim of the tutorial was to run through the sociological aspects that surface on any project. After listing them out with the help of the audience, Suzanne and Ian provided us with a number of useful techniques to handle them.

The tutorial was managed in a very intelligent manner, something that was quite important considering its very interactive nature. The session was initiated by asking the audience what concerns they had with regards to handling the stakeholders of any project they might be involved with. This was a straightforward and very useful way of initiating the session because it included every participant from the start. Furthermore, throughout the rest of the afternoon, both presenters made sure that every member of the audience was involved in a more interesting role than that of the standard "passive" attendant. They were certainly practicing what they preached by making sure that "all the stakeholders in the tutorial had their say". This was possible because we were presented with an exercise that grew in complexity as the afternoon progressed and which demanded teamwork. The tutors partitioned the session in stages. Each one included an introduction in which one of the tutors briefly summarised what had been achieved to that point, and described the part of the method we would be working on. After that the participants were immediately put to

work by thinking about the best ways to apply what had been learned on the problem at hand.

The framework and tools described and applied during the tutorial certainly do provide the means to manage in a structured manner the different variables that need to be juggled around when developing any project, particularly in the first fuzzy stages of the project's requirements phase. The onion model constitutes an elegant mechanism to organize the various stakeholders as they appear. It also allows those involved in the project development to express in a simple way the influence each of them has in the form and content of the final solution.

The knowledge model is the other exploration tool that was presented. It merges the symbols available to build a UML class diagram with those used to specify an entity-relationship diagram. A knowledge model can be built to describe all the elements that must be handled by a "system" solution, together with the stakeholders. It includes mechanisms to specify the associations that exist between stakeholders and system elements making it straightforward to explain the existence of every component that is incorporated in the final solution, opening a way to implement traceability of a project's artifacts once they have been produced.

The interactive nature of the tutorial made it extremely easy for the audience to remain attentive throughout the whole session. It also helped us to hear the concerns of the other participants and their approaches to solve the exercise we were working on, as it complemented very well the rest of the material discussed during the session. Probably the only weak point I detected was with regards to the lack of opportunity there was to clarify aspects of the requirements process that were not directly related with the exercise that we were trying to solve. For example, even though the list of concerns generated at the start of the session was quite extensive, it was never referred to again. I feel it would have been useful to point out how the techniques and models introduced in the tutorial tackled those concerns, maybe not all of them, but an insight on a couple would have put a great finishing touch to a successful afternoon.

© Elena Pérez-Miñana

RE-Papers

Requirements Engineering as if Stakeholders Mattered

Richard Veryard
Veryard Projects

Introduction

In a recent paper, Ian Alexander mentions the old adage: "He Who Pays The Piper Calls The Tune". The story of the Pied Piper of Hamelyn provides a memorable example of the reverse proposition – what happens

when the piper's fee remains unpaid – and this story has some important lessons for requirements engineering.

Requirements Change

At the start of the story, the citizens of Hamelyn are plagued with rats. Whereas an experienced requirements engineer might see this as merely a symptom of a much deeper problem of public health and hygiene, the citizens are determined to treat the symptom directly, and define their requirements accordingly – Who will rid our town of rats? A piper

is engaged, who delivers an unusual yet effective solution to this requirement – he plays his pipe, and the rats run out of town and into the sea. Furthermore, this solution is achieved without the use of noxious chemicals or other environmentally harmful side-effects.

But once the citizens of Hamelyn are no longer plagued with rats, their requirements change. They no longer perceive the value of the piper’s solution, and seek to renegotiate his fee.

A significant shift in client preferences may be regarded as a requirements change.

The citizens still have a theoretical desire to be rat-free, but the urgency of this desire has disappeared, and this affects their willingness to pay. We often see a similar phenomenon in IT projects; if the users’ most urgent needs are addressed in Phase 1, the users may lose interest in Phase 2.

Engineer as Stakeholder

The Pied Piper is an important stakeholder in this story, and he now has a problem. He hasn’t been paid. But nobody else in Hamelyn sees this as a problem. Why should they?

One of the most important steps in problem-solving is to ask: Whose problem is this? [See Gause and Weinberg] The Pied Piper needs to find a way of turning a problem that is his alone, into a problem for the whole town.

Reframing a requirement can make it forcibly relevant to a new set of stakeholders.

Solution Generalization and Reuse

Fortunately for the Pied Piper, his environmentally-friendly rat-elimination solution turns out to satisfy a number of other requirements. Whether by chance or design, the solution supports the elimination of a much larger class of creatures, including children (Figure 1). He plays his pipe again, and this time it is the children who run out of town after him.

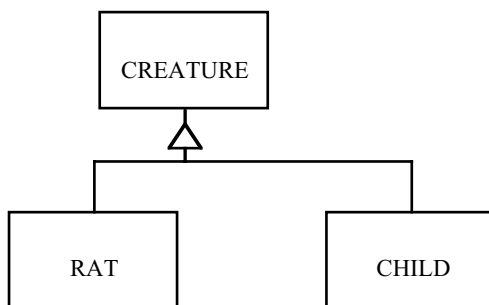


Figure 1

(The symbolic and practical implications of lumping rats and children into a common class may be uncomfortable and unwelcome for at least some stakeholders.)

Stakeholder Role Change

When the Pied Piper reuses his solution to take away the children, this brings into the foreground the citizens’ role as parents. The role of parent overlaps with the role of citizen – not all citizens have children, not all the parents who happen to be in the town with their children at the time are permanent citizens of Hamelyn.

A significant event can cause stakeholders to switch roles – and this typically changes their requirements.

Furthermore, there is now a potential conflict of interest between the parent role and the citizen role. The parent may see a higher value in persuading the Piper to return the children; the citizen may still be eager to keep rat elimination expenditure within a tight budget.

Stakeholder Solidarity

Imagine a group of anxious Hamelyn parents – clearly they are now stakeholders in this story. A key question for stakeholder management is whether they are able and willing to act in concert.

Some parents may think they can get a better deal for the safe return of their children by negotiating individually with the Piper; and a commercially astute Piper might be able to exploit this belief by encouraging individual negotiations. Meanwhile, the leaders of the parents’ group would be trying to encourage parents to act as a group and not break ranks.

Although the parents are all interested in the return of the children, they are each concerned about different instances of the class CHILD. This means that the commonality of interest between this group of stakeholders, and their willingness to collaborate, is dependent on the requirements being framed at the class level rather than at the instance level.

The Revenge of the Good Fairy

Mary Catherine Bateson (daughter of Gregory Bateson and Margaret Mead) once wrote a short but brilliant essay for the Whole Earth Review, showing how “good” technology often has “bad” side effects.

Like the characters in many fairy tales, the citizens of Hamelyn have three wishes – which represent a progression of requirements. Note that each requirement only makes sense in terms of the failure (or incomplete success) of the previous one.

- To rid the town of rats.
- To rid the town of expensive ratcatchers.
- To get their children back.

In some of the most elegant fairy tales, the third wish takes the stakeholder back to the starting position. These are requirements that seemed like a good idea at the time, requirements that have led us into dangerous territory, requirements that we may now happily abjure.

References

Ian F Alexander: "Stakeholders – Who is Your System For?", Computing & Control Engineering, 14(1), pp 22-26. April 2003.

Ian F. Alexander & Suzanne Robertson: "Stakeholders Without Tears", RESG Tutorial, University College London, July 16th 2003.

Mary Catherine Bateson: "The Revenge of the Good Fairy", Whole Earth Review, 1987.

Donald Gause & Gerald Weinberg: Are Your Lights On? How to Figure Out What the Problem Really Is., Dorset House Publishing, 1990.

Library of Patterns for Interactive Applications

*Jonathan Beebe
Fujitsu Consulting*

Working in industry as practitioners, whilst not allowing for rigorous research into Requirements Modelling, does however provide a sharp focus on delivery along with very practical examples. A great deal of our recent work has centred on the specification and development of applications that support and enable interaction of human operators, both self-service (eg kiosk, Internet, etc) and via an agent (eg call-centre, shop, etc), with the processes of a business (eg paying a bill, making a complaint, booking a ticket, etc).

The initial challenge of any project involving the specification of such systems by a team of analysts is to ensure the completeness and consistency of requirements that are gathered from a wide variety of sources, often prior to our involvement in the project. Subsequently, the quality of the implementation and our ability to demonstrate that it meets its requirements depends on traceability of the system components and their tests back to those original requirements. The apparently simple task of reviewing the initial requirements of a project is extremely expensive in terms of the number of people involved so it makes obvious sense to try to streamline this part of the project lifecycle and hopefully, extend these benefits throughout the rest of the project.

With this goal in mind we set about identifying a (small) number of patterns that occur commonly within the systems we are specifying. These patterns are neither specific to the business domain nor are they technology domain specific. They fall somewhere between, which makes them accessible to the widest audience.

We have documented the patterns both as:

- a Pattern Definition according to the "Gang-of-Four" style [Gamma, 1995].
- a UML model comprising a Use Case and Activity Diagrams for each pattern

and are publishing them as a library linked to Fujitsu Consulting's MacroScope[®] suite of business and IT methodologies [MacroScope, 2003].

The **Pattern Definition** style documentation is most useful in identifying which patterns are relevant and useful. The **UML** is useful as a guide when using the pattern as it lists aspects of the pattern that should be included and advice on how they should be documented.

The description text attached to the use case and each activity of a pattern guides an analyst to seek out and document (and/or diagram) the significant features of each item. The patterns themselves can be included, for reference, as UML stereotypes within the requirements model of a project.

The choice of patterns as a mechanism (rather than the more exact form of, for example, a class library) is intended to suggest flexibility in how they should be applied. By this, I mean that it may very well be that in any usage of a pattern complete parts of the pattern may be omitted because they are not relevant, but the remaining parts of the pattern are still of value and should be applied. Similarly, a particular instance of use may require features of a number of patterns to be combined, possibly with one dominating pattern (which would then be chosen as the use case stereotype) and the other patterns in supporting roles.

The names given to the patterns in the library indicate that they originate from a functional view of the problem domain. However, they should include directives on the capture and cross-referencing of quality requirements (performance, security, availability, usability and "potential for change").

So what is a typical pattern? Following are two examples:

- Simple Service Pattern
- Menu Pattern

Simple Service Pattern – An interactive application is likely to be composed of a number of use cases involving direct interaction with the user where the user makes a simple request of the system which responds and reports the outcome. These are the "atoms" of the application.

Menu Pattern – is an example of a "compound" pattern (ie it is made up of a number of, possibly different, atomic and compound patterns). This pattern is not about the visual presentation of a menu, more about the options offered and what happens when they are selected and invoked.

The number of patterns for interaction in the library can be expected slowly to grow and the detail of some patterns is less developed than others – so there is more work to do. I believe the small number of patterns in the library is a measure of its quality and value, since the smaller the number of patterns, the greater is their level of re-use and the easier it is to locate the appropriate patterns. However, this does require careful

management of the library and emphasises the importance of a librarian role.

I am not suggesting that this work presents a revolutionary approach or methodology in the field of Requirements Engineering. The real value of this work lies in the library of patterns itself and the degree to which it can be re-used. Additionally, I expect that this approach could be applied to a wide variety of problem domains and could therefore be of great benefit to practitioners in general.

Whilst an investment of time and effort is required in developing a suitable pattern library in the first place, that investment should be re-paid from improved

effectiveness of communication within and across projects. It is after all, the job of practitioners to bring to their customers the benefits of experience and I believe that this library provides an effective mechanism for sharing and propagating this experience. It has certainly proved to be so for our own customers.

References:

[Gamma, 1995] “Design Patterns” (CD); Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides; Addison Wesley; ISBN 0-201-63361-2; 1995

[Macroscopic, 2003]

<http://uk.consulting.fujitsu.com/thought/methodologies/macroscopic/>

RE-Publications

All reviews by Ian Alexander

Designing Information Spaces: The Social Navigation Approach

*Kristina Höök, David Benyon & Alan J Munro (Eds)
Springer Verlag 2003
ISBN 1-85233-661-1*

“The system shall be easy to use” has become a stock joke among requirements people. Easy for whom? Easy to do what? And how would you know? But Usability requirements do not go away.

Computer-Supported Cooperative Work (CSCW) is the great research movement that straddles the boundary between human-machine interaction, software engineering, and sociology.

The specific CSCW problem that this book addresses is how to help people to find their way in the complex virtual spaces that are needed to allow people to work together without having to be in the same room – or even working at the same time. This isn’t easy, as the ‘bandwidth’ – speaking in human terms – even of powerful networks and videoconferences is very small compared to the mass of cues we get from people around us in a real meeting.

So the requirement is to enable people to have enough of an experience of sharing ideas and intentions to be able to work effectively – there is an essential element of challenge in there!

The solutions proposed in this book are many and various, but they focus on mainly visual means of indicating things like who else is sitting in a ‘lecture room’ with you, how far back you are in a queue and crucially how long you’ll likely have to wait, or who you can talk to. Some of the interfaces seem quite fanciful – things you can look at are represented as odd-shaped skyscrapers in a green landscape, or as islands in a deep blue sea. Others are much more modest, colour-coding hotel bedrooms so you can see if you have to

book right now or can hesitate a bit longer, secure in the knowledge there are quite a few rooms left.

The challenges are enormous. One example discussed is the case of a London Underground control room, in which the Divisional Information Assistant overhears the Line Controller mumbling something to a train driver about a problem, immediately understands it’s urgent, and starts to announce details of the resulting disruption to passengers *before* the Controller tells him about the nature of the problem! We all know examples of really good teamwork where people hardly need to speak. I remember a live-firing night-time demonstration by a tank regiment in which 95% of the radio communication was by a pre-arranged sequence of audible clicks generated by microphone buttons. Would you design a computer system for a mission- and safety-critical application involving no visual communication and almost no voice channel either?

It is hard to judge from a book such as this how successful the proposed approaches are or will be. The topic is obviously important as we rush headlong into a virtual world – we are getting less and less time, and less and less information (like face-to-face contact), on which to base crucial decisions. The book is by its nature researchy – the authors are academics and research workers in industry – but if the enterprise is at all successful it will influence the whole environment in which we work, shop, and play.

Visualizing Argumentation Software Tools for Collaborative & Educational Sense-Making

*PA Kirschner, SJ Buckingham Shum and CS Carr (Eds)
Springer Verlag 2003
ISBN 1-85233-664-1*

“The aim of the whole exercise .. was to promote rational consensus on the main issue. The next stage .. was to review the arguments

as presented on the visualization and to see what this implied... By this time, however, something remarkable had already happened. As the negative case was being visualized, one argument emerged as conclusively establishing that the proposition was false...

The remarkable part was that when this objection was laid out clearly in the context of all other relevant considerations, its overriding force was fully appreciated in a way it had never been when the arguments were rehearsed in standard ways." [Tim van Gelder, p112]

The basic assertion of Computer-Supported Cooperative Work (CSCW) is that there is a synergy when the right technology combines with skilled human effort. The special claim of this book is that CSCW tools that illustrate arguments and their supporting evidence contribute significantly to that synergy. Tim van Gelder in a way summarizes the case: at least sometimes, it is virtually enough just to elicit and diagram an argument. Once everybody can see the battle-lines drawn out, the need for debate often melts away. Presumably the psychological reason for this is shared experience. Haven't you felt, on hearing on the radio that 70% of the public agree with a claim, that you yourself are 70% persuaded and 30% not persuaded? Everyone knows the arguments, but not everyone visualizes them clearly. If so, consensus is just a visualization away.

Part I, Foundations, consists of just two chapters – on the roots of Computer-Supported Argument Visualization (let's call it CSAV for short) by Buckingham Shum, and a cognitive framework for cooperative problem solving with CSAV, by Jan van Bruggen and others. Both discuss the IBIS method (which is supported by several tools).

One wonderful example: John Henry Wigmore's *Chart Method* for analysing legal evidence was devised in 1913. It was certainly an AV method though the computer was just pen and paper. Hypertext, too, was from the start about argumentation, as Vannevar Bush made clear in his famous article *As We May Think* in 1945. Another major root is the philosophy of language of Stephen Toulmin, whose *Uses of Argument* (1958) counters 2000 years of excessive Aristotelian logic. Toulmin devised an argument structure involving Datum, Claim, Warrant, Backing, and Rebuttal, all to be arranged graphically – again, clearly an AV format. Buckingham Shum gently pokes fun at 'tame' problems, contrasted with Horst Rittel's 'wicked' problems that CSAV tries to address. He observes that

"Tame problems may even be amenable to automated analysis, such as computer configuration design or medical diagnosis by expert system." (p11)

This is a pointed reference to the most famous examples of Artificial Intelligence such as Mycin. CSAV's answer is that machines are far too stupid to

solve wicked problems, but that computers and people together may do so.

van Bruggen makes the point that CSAV

"is primarily used for the solution of ill-structured problems" (p27)

the term goes back to Herbert Simon, and is certainly in the same area as wicked problems. Such problems require complex reasoning, the coordination of multiple agents, and CSAV environments that meet stringent cognitive and communicative requirements.

Part II, Applications, consists of seven chapters illustrating different CSAV approaches. Gellof Kanselaar and others write on designing tools for collaborative learning; Carr writes on teaching legal argumentation; van Gelder on deliberation (i.e. critical thinking); Jeff Conklin on dialog mapping using IBIS; Albert Selvin on helping groups use CSAV; Robert Horn on large, long-lived interdisciplinary arguments; and Buckingham Shum and others on internetworked (i.e. hypertext) argumentation.

This part of the book inevitably repeats the basic message in various ways, arguing the merits of different approaches toolkits. Kanselaar illustrates the TC3 tool with several diagrams full of Dutch text, structured much like Toulmin; the article argues that education is a process of argumentation, but their analysis showed that people actually used their tool for visual representation, not to trigger discussion or ideas.

Carr is on stronger ground talking about legal arguments; these are admittedly complex and adversarial, and concepts like claims and evidence are well accepted. He uses Toulmin's scheme, and shows in a careful study that – contrary to expectation – students did not get better exam scores after using the QuestMap CSAV tool, though they may have ended up with better ability to construct arguments (rather than the ability to write them, as tested by the exam).

Van Gelder starts slowly, recapitulating the basic comparison of argument in text versus diagrams, but arrives at 'Deliberation is usually done quite poorly' (p104) and quotes Deanna Kuhn's *The Skills of Argument* (1991), where she shows that over half the population cannot provide any evidence to support their opinions. In a suggestive photograph, he shows three students literally manipulating arguments by pointing with their fingers at text boxes displayed by the Reason!Able CSAV tool on a large touch-sensitive whiteboard. Again contrary to expectation, students did not do better with CSAV practice! Instead, the use of the tool may have provided an immediate benefit – argument visualisation itself – which at once enabled the students to work at a higher level of reasoning. This isn't proof that AV enhances deliberation, but it is certainly suggestive.

In contrast, Conklin says that IBIS needs a 2-day course and then some practice "with the emphasis on practice". Evidently Dialog Mapping in real time is harder than sitting down quietly to do solo analysis

with Reason!Able. Dialog Mapping basically consists of IBIS use with a skilled facilitator, who is supposed to deal with both the people-facing and the system-facing issues at once (it isn't clear why a 'scribe' to help with the actual writing and drawing isn't part of the method – this is a well-known role in systems work at least – but 'or two people' is mentioned in a footnote). The case study is interesting but startling: the QuestMap CSAV tool (implementing IBIS) was applied not just to support face-to-face meetings with Dialog Mapping, but also for the wholly different job of virtual meetings, which could be asynchronous. Not surprisingly, some people "reacted strongly" to this "as just another passing fad of management". Indeed there are powerful arguments against reliance on technological mediation in place of actual meetings. The case study company, SCE, used the tool for a decade because of the 'commitment of Dr Hertel, the Director'. This is quite a low-tech reason for the success of a hi-tech tool, but probably typical in organisational change – just arguing about gadgets is missing the point.

Selvin talks about face-to-face VA: the method (Compendium) and tools (Mifflin, QuestMap) support facilitation of meetings, helping to impose an appropriate structure. Obviously the risk is that the structure is wrong, or resisted by participants. Selvin claims that 'I .. believe that I could add value to almost any group in a problem situation' through years of experience, but that few (or not enough) similarly skilled facilitators exist, and that it is an open question whether the required skills can be taught.

Horn also revisits the basic story of CSAV, offering guidelines for good practice when presenting big philosophical arguments. The example is the perennial Can Machines Think?, and arguments are given in their earliest or strongest forms, such as those of Heidegger, Dreyfus, and Searle. The documented argument ends up in seven large maps – each a 3x4 foot poster! – presenting about 800 steps or 'moves' and 60 photographs linked by 'is supported by' and 'is disputed by' arrows. He claims that this work 'has opened up the development of a new field for the understanding of intellectual history'. The future holds a hypertext version of the approach:

"we envision the ability over time to witness the creation of vast webs of argumentation maps on the web that cover many fields and that show us how humanity working together has asked, debated, and sometimes even answered the Great Questions." (p183)

This bold claim seems to be partly contradicted by the statement

"we are not interested in developing a chatroom. The content review needs to be done by specialists."

Well, if so, and only the few can create the maps, how is a vast web ever going to be created?

Finally, Buckingham Shum takes up the challenge of hypertext, more or less as thrown down by Horn. Will scholarly knowledge still be published solely in prose, or as some kind of semantic web? Will that enable more effective dissemination and analysis of ideas? Sven Birkerts would I think say that it might assist dissemination but that it threatens the really deep integration and human 'sensibility' that goes beyond mere 'analysis'. This question seems to strike at the heart of CSCW (and CSAV for that matter). If McLuhan is right and the medium shapes the message, then mediating between people with tools changes the nature of the communication, and of what people become able to understand. The brave pioneering authors of this book are to some extent obliged (in a quiet sort of way) to believe that the tools, rightly and intelligently applied, will help to bring on the bright new dawn of technologically-mediated splendour. The dour sceptics wonder whether the new media will not instead reduce people's attention span and ability to reason. At the least, the case for CSAV is today not proven.

Doug Engelbart – who has been writing on CSAV since before it existed, back in 1962, concludes with an Afterword. He ranges grandly from his Bootstrap Institute on increasing humankind's collective problem-solving capabilities. It is almost impossible not to believe that computers could do this today; it was very different five decades ago. Engelbart hopes that CSAV tools and methods will themselves improve the pursuit of new tools and processes – a bootstrapping approach. Perhaps today's CSAV tools, and their rather limited diffusion in industry, don't quite seem to be this powerful; but perhaps they aren't far from reaching the critical point where they really start to help.

One omission that seems glaring to a systems engineer's eyes is any mention of safety case argumentation. The situation in which developers are most likely to see or document arguments – if they see any – is in safety engineering. Well-known notations include Goal-Structuring Notation, Adelaar's Claims Arguments Evidence, and Weighted Factor Analysis. These are definitely CSAV tools and methods, but they seem to fall foul of the 'not invented here' syndrome: the CSCW school doesn't read books or papers about safety, and vice versa. On the other hand, there is a good discussion of the problems of capturing Design Rationale. It seems too much to hope that different schools of thought – CSCW, HCI, Software, RE, Systems – should read each others' literature, when the evidence shows that they hardly read their own. The irony, if that is what it is, is that as Sven Birkerts so passionately argues in *The Gutenberg Elegies*, the rising tide of e-data that washes through our networks actually shortens our attention span so much that we can hardly concentrate on the in-depth argument and reflection presented by a book any more. CSAV may, perhaps, help with argument understanding; but equally, it looks like part of the larger problem.

This is one of those few multi-author books that fits together rather well, and has a clear and important message for everyone involved in complex business and system problems. The question of whether computers can assist human thought is a deep and challenging one, and deserves everyone's attention.

Note: RQ readers may like to look at the book's website at <http://www.VisualizingArgumentation.info>

User-Centered Requirements The Scenario-Based Engineering Process

*Karen L. McGraw & Karan Harbison
Lawrence Erlbaum Associates 1997
ISBN 0-80582-064-7*

Have you ever wondered whether there was more to requirements than just lining them up neatly in a good requirements tool, prioritising them, and tracing them to your design and tests? Does anyone know how to get hold of them in the first place, to make sure they are right, that they are what the users want, and reasonably complete? Should you interview department heads or hold a workshop of ordinary users? How much detail should you capture? What should you do if interviews go on too long?

The good news is that there are well-worked-out answers to these practical questions, and to many others. This book very sensibly combines the expertise of an academic (Harbison) and an industrialist (McGraw) to explain how to make your requirements tell the story of what results your users really want to have. The book therefore has some industrial features, like big, simple pictures based on clip-art, clearly illustrated examples, and step-by-step procedures, as for instance an 8-step account of how to document a requirements elicitation session. It also shows the careful hand of an academic, giving details of many techniques that might be used for each purpose rather than just recommending one that's tried and trusted.

As a result, this book is probably most useful as a reference when you are, say, preparing a requirements workshop and are wondering if you should try out a different technique or two, outside your normal repertoire. The alternatives – or at least a good many of them – are plainly set out, and once you have chosen one (perhaps that is the difficult step) you will find a straightforward recipe for doing it in the appropriate section.

For example, to analyse existing or 'as-is' scenarios, the authors suggest making event traces (we'd say swimlanes or sequence diagrams), graphics such as storyboards, matrices (tables) showing events, activities, information, performers, and key issues for each scenario – what we might call an Actors & Activities table – or building an object model showing the human or system roles, the information each maintains, and the activities that each carries out.

A different set of techniques is suggested for desired or 'to-be' scenarios. The book suggests using a

topological or geographical map to show where events occur in relation to each other; a flowchart to depict the major scenes or events, and then running facilitated brainstorming and consensus decision-making sessions to expand each scenario into more detail. These techniques are more exploratory and able to deal with ideas and speculation about the best way to do things, whereas the 'as-is' techniques suggested are more definitive and analytic, as you need when documenting the way things are actually done.

Part I introduces 'the Scenario-based Engineering Process' and the activities and 'artifacts' of knowledge elicitation and analysis. The Scenario-Based Engineering Process has been in industrial use in several companies since at least 1997 and continues to be used in specialised areas where complex knowledge involving human decision-making is important.

Part II presents a thorough and wide-ranging set of processes and techniques: planning requirements activities, selecting techniques, elicitation, interactive observation, interviewing, identifying work processes and task analysis, domain concepts, using process tracing to analyze the problem-solving process (i.e. decision-making, expertise), group sessions, evaluating and refining requirements.

Group techniques are perhaps especially well covered. Techniques include walkthroughs, decision process tracing combined with protocol (transcript) analysis, goal-oriented brainstorming by traditional 'shout and post', brainstorming by round robin – everyone gets a turn, making a problem tree or herringbone diagram, consensus decision-making, the nominal group (no arguments, comments are written down and passed to the facilitator who lists them and organizes a vote), focus groups, computer-aided brainstorming (as with Barry Boehm's Win-Win), or video conferencing. In other words, far from there being no obvious ways to gather requirements, the real problem is choosing the most appropriate ways for a given situation.

The authors have a background in Knowledge Acquisition and Participative Design, rather than Requirements Engineering as such. They are also very familiar with techniques from psychology, such as Kelly's repertory grid, and man-machine interaction. This means that the authors say 'performers' where we might expect 'actors' or 'roles'; 'shareholders' where we might say 'stakeholders', 'defining work processes' rather than 'business process modelling', 'task analysis' instead of 'scenario (or use case) analysis', 'group sessions' not 'workshops'. These little differences shouldn't matter to any reasonably well-informed reader, but they do indicate that people come to the requirements field from many different directions.

There are a few other good books on gathering requirements, notably Ellen Gottesdiener's *Requirements by Collaboration*; there are some that provide practical advice on scenario-based requirements, such as Alexander & Stevens' *Writing Better Requirements* or Alistair Cockburn's *Writing Effective*

Use Cases, but perhaps none that cover so many techniques so clearly.

If you gather requirements and would like to have a practical and well-read pair of experts at your elbow, you could do a lot worse than to search out a copy of this book. It is detailed, accurate, careful, informative, and well-organized.

Students may also find it helpful; there are no exercises or glossary, but the book introduces its topics separately and the references and index are good. For the same reasons it would make a solid basis for a lecture-course.

User-Centered Requirements is full of good sense, so let us give the authors the last word:

"Common problems that lead to defective requirements include:

- Developer assumptions that the requirements are 'obvious'.
- Failure to do enough requirements elicitation and analysis.
- Failure to include appropriate personnel in requirements elicitation and analysis."

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 at the RESG website: <http://www.resg.org.uk>

Ian Alexander's archive of book reviews is available from: <http://easyweb.easynet.co.uk/~iany/reviews/reviews.htm>

The requirement management place <http://www.rmplace.org>

A good general resource for RE issues. Includes Alan Davis' Requirements Bibliography.

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.

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/

Requirements Engineering Journal (REJ):

<http://rej.co.umist.ac.uk/>

Reduced rates are available to all RESG members when subscribing to the REJ.

RE resource centre at UTS (Australia): <http://research.it.uts.edu.au/re/>

Volere: <http://www.volere.co.uk>

Mailing lists

RE-online (formerly SRE): <http://www.staff.it.uts.edu.au/~didar/RE-online.html>

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

subscribe RE-online <your email address>

LINKAlert: <http://link.springer.de/alert>

A free mailing service for the table of contents of the *International Journal on Software Tools for Technology Transfer*.

RE-Actors

The committee of RESG

Patron: Prof. Michael Jackson, Independent Consultant. E-Mail: jackson@acm.org.

Chair: Prof. Bashar Nuseibeh, Computing Department, Faculty of Maths and Computing, The Open University, Walton Hall, Milton Keynes, MK7 6AA, UK. E-Mail: B.A.Nuseibeh@open.ac.uk.

Vice-Chair: Dr Alessandra Russo, Department of Computing, Imperial College, 180 Queen's Gate, London SW7 2BZ, UK. E-Mail: ar3@doc.ic.ac.uk

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.

Secretary: David Bush, National Air Traffic Services, UK. E-Mail: David.Bush@nats.co.uk.

Membership secretaries:

Steve Armstrong, Computing Department, Faculty of Maths and Computing, The Open University, Walton Hall, Milton Keynes, MK7 6AA, UK. E-Mail: S.Armstrong@open.ac.uk.

Juan Ramil, Computing Department, Faculty of Maths and Computing, The Open University, Walton Hall, Milton Keynes, MK7 6AA, UK. E-Mail: J.F.Ramil@open.ac.uk.

Newsletter editor: Dr Peter Sawyer, Lancaster University, Computing Department, Lancaster, LA1 4YR, UK. E-Mail: sawyer@comp.lancs.ac.uk.

Newsletter reporter: Ian Alexander, 17A Rothschild Road, Chiswick, London W4 5HS. E-Mail: iany@easynet.co.uk.

Publicity officer: Sebastian Uchitel, Department of Computing, Imperial College, 180 Queen's Gate, London SW7 2BZ, UK. E-Mail: su2@doc.ic.ac.uk

Regional officer & Chair for the North of England: Kathy Maitland, University of Central England, Perry Bar Campus, Birmingham, B42 2SU. E-Mail: Kathleen.Maitland@uce.ac.uk.

Student Liaison Officer: Carina Alves, University College London, Department of Computer Science, Gower Street, London WC1E 6BT, UK. E-Mail: c.alves@cs.ucl.ac.uk

Industrial liaison:

Dr. Wolfgang Emmerich, University College London, Department of Computer Science, Gower Street, London WC1E 6BT, UK. E-Mail: W.Emmerich@cs.ucl.ac.uk.

Dr Sofia Guerra, Adelard, Drysdale Building, 10 Northampton Square, London, EC1V 0HB, UK. E-Mail: aslg@adelard.com.

Elena Pérez Miñana, Philips Digital Systems Lab. E-Mail: elena.perez-minana@philips.com.

Dr Efi Raili, Praxis Critical Systems, 20 Manvers Street, Bath BA1 1PX. E-Mail: efi@praxis-cs.co.uk.

Suzanne Robertson, Atlantic Systems Guild Ltd. 11 St. Mary's Terrace, London W2 1SU, E-Mail: suzanne@systemsguild.com

Gordon Woods, Independent Consultant, E-Mail: Gordon@cigitech.demon.co.uk

RE-Creations

To contribute to RQ please send contributions to Pete Sawyer (sawyer@comp.lancs.ac.uk). Submissions must be in electronic form, preferably as plain ASCII text or rtf.

Deadline for next issue: 31st October 2003.

RE-Funds

Minutes of the 9th Annual General Meeting of the Requirements Engineering Specialist Group of the BCS

July 16th, 2003
University College London

Chair: Bashar Nuseibeh (BN)
Secretary: Wolfgang Emmerich (WE)

1. Minutes of Previous Meeting (RQ 27)

BN noted that the minutes of the 8th AGM of the RESG had been published in RQ 27 and asked the meeting whether these could be approved as a correct record of the proceedings. There being no corrections, the minutes were approved.

2. Matters arising

There were no matters arising from the minutes of the previous meeting.

3. Chair's Report

3.1 Events

BN noted that the RESG had been able to organize quite a few events during the last year and that most of these events were well attended. The events were:

- Scenarios Work!10th July 2002, UCL, London (following 2002 AGM)
- RE:Fresh, 8th October 2002, UCL, London (evening)
- Using Formal Models to Understand Requirements Better, 6th Nov 2002, Imperial College, London (full day)
- A Masterclass in Security Requirements Engineering, 29th January2003, Imperial College, London (half day)
- COTS Integration: why you need requirements, 9th April 2003, University of Central England, Birmingham (half day)

Furthermore, RESG co-sponsored a number of further events. Co-sponsorship means that RESG informs its membership about these events and that the members generally obtain admission at reduced rates. The co-sponsored events were:

- Requirements, Risk and Value in Systems Engineering, organised by the EPSRC funded

SIMP project, 18 July 02, City University London

- Building Distributed Software Architectures (BDSA), vacation school, 2-6 Sep 02, UCL, London
- RE'02 - 9-13 September 2002, Essen, Germany
- Mastering the Requirements Process, 3-day seminar/workshop, 6-8 Oct 03, London
- Extending Requirements: A Practical Workshop, 2-day workshop, 28-29 April 03, London
- Introduction to Requirements, One day course at IEE, 1st May 2003 London

A number of events are being planned for the remainder of 2003 and 2004. These are:

- 17th September 2003, Imperial College London, An audience with Tom Gilb
- 9th October 2003 (tbc), London, Creativity Tutorial, by Neil Maiden and Suzanne Robertson
- 3rd December 2003, Imperial College London, A workshop on RE Training, featuring Pete Sawyer/Ian Sommerville, Ian Bray, Ken Jackson, Stephanie Linsteadt
- 30th March 2004, London – RE-Day 2!!!
- End of May 2004 – in Manchester!
- 7th July 2004, London, AGM & Debate

3.2 Newsletter

Requirenautics Quarterly (the RESG newsletter), appeared not quite quarterly last year. There were three issues edited by Pete Sawyer (Lancs) with help from Ian Alexander.

BN encouraged the RESG membership, particularly industrial members, to contribute articles. As an incentive the Executive Committee agreed to reward each author of a published article with a copy of a requirements engineering book (till we run out of stock).

3.3 Industrial Liaison

BN noted that the RESG had been quite successful in increasing the industrial participation in its activities. More than half the participants at most events were from industry. BN thanked the industrial liaison team on the Executive Committee for their efforts to make this happen. The team last year was comprised of:

- David Bush (NATS)
- Sofia Guerra (Adelard)
- Elena Perez-Manana (Philips)
- Efi Raili (Praxis Critical Systems)
- Suzanne Roberston (Roving Ambassador!)
- Michael Jackson (Patron)

3.4 Regions

The Executive Committee notes that there is a tendency of organizing events in London, which is partly due to the fact that it is easier to travel to London from most regions than to travel cross country. However, the Executive Committee undertakes to locate at least one event per year outside Greater London. Last year's event was on COTS integration in

Birmingham. The Executive Committee welcomes suggestions for locations and events from the Membership.

3.5 Publicity Report

The RESG Web site (<http://www.resg.org.uk>) is now hosted by Imperial College London.

The RESG Email distribution list, which is being used sparingly and by moderation only to advertise RE-related events. The list now contains 850 addresses (up from 650 last year) of which approximately 500 are non-members.

For suggestions on publicity contact Sebastian Uchitel (s.uchitel@imperial.ac.uk).

4 Membership Secretary's Report

Steve Armstrong reported about the membership. The RESG currently has 348 members, which is slightly down from 2001/02 membership of 483. However previously membership was free.

During 2002/03 membership data storage was improved reduced turnaround time.

The aims for 2003/04 are to increase the membership, particularly of industrial corporate members, to simplify registration and to further reducing the turnaround of mailings.

5 Treasurer's Report

On behalf of the Treasurer, BN presented the financial situation of the RESG, which appears to be in good shape. BN noted that the funds of 22,682.77 were necessary to be able to underwrite events, such as the second RE-Day.

The summary of the accounts for 2002/03 is as follows:

Balance on 1st May 2002:	22,682.77
Receipts	
Subscriptions	1710.00
Interest	4.45
Savings interest	817.20
Event income	2027.50
Expenditure	
Printing	100.00
Event expenditure	2507.28
Other expenditure	2571.77
Net movement	-619.90

Event income and expenditure is all events from slow income for RE product families event in April 2002 to RE COTS Integration event in April 2003. Overall, although year-end figures do not suggest it, we are not making a loss on events.

Other expenditure was:

RESG web-site	90.16
RESG advertising item	301.98
RE02 sponsorship	1528.03
VAT payments	651.60

6 Election of Executive Committee Members

The Executive Committee Members in 2002/2003 were:

Patron:	Michael Jackson	(Independent)
Chair:	Bashar Nuseibeh	(The Open University)
Vice Chair:	Alessandra Russo	(Imperial College)
Secretary:	David Bush	(NATS)
Treasurer:	Neil Maiden	(City University)
Membership Sec:	Stephen Armstrong	(The Open University)
	Juan Ramil	(The Open University)
Publicity Officer:	Sebastian Uchitel	(Imperial College)
Newsletter Editor:	Peter Sawyer	(Lancaster University)
Reporter:	Ian Alexander	(Independent)
Regional Officer:	Kathy Maitland	(UCE at Birmingham)
Student Officer:	Carina Alves	(UCL)
Industrial Liaison Team:	Wolfgang Emmerich	(UCL & Zuhlke)
	Sofia Guerra	(Adelard)
	Elena Perez-Minana	(Philips)
	Efi Raili	(Praxis)
	Suzanne Robertson	(Atlantic Systems Guild)
	Gordon Woods	(Airbus UK)

Richard Veriyard proposed the committee for re-election. Lyn Antill seconded the proposal.

The committee was then re-elected with no votes against.

7 Any Other Business

There being no other business, the Chairman closed the meeting.