BCS™

Requirements Quarterly

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

 http://www.resg.org.uk RQ51 (April 2009)

Contents

RE-Soundings

From the Editor

There is always a sense of foreboding as newsletter deadline day approaches. Will there be enough interesting and varied articles to fill the pages, or will I find myself producing a one page leaflet instead? On this occasion we tick the varied, interesting and sufficient boxes in considerable style, with an eclectic mix of events, reviews and original articles.

It must be an indication of progress that there are so many requirements-related events this year. This edition includes details on forthcoming RESG and non-RESG conferences, and reflections on several recent events, including one from New York, a book launch and a successful Brown Field Discussion.

Our RESG Annual Summer Party should certainly be in your diaries by now – it will be real opportunity to take advantage of the varied experiences and knowledge we have in our Specialist Group in a relaxed setting.

And don't forget that you don't have to wait until the next event or RQ deadline day to heap praise on the committee – details of our online comments forum can be found in the Membership section. Do join in!

Simon Hutton, RQ Editor

Chairman's Message

This spring the RESG is bringing you not one but two book launches. Even though I'm one of the authors, I fancy that the nature of the two books says much about the tension inherent in "RE".

Discovering Requirements was launched on the 2nd of April at UCL. We had a lovely party – for which many thanks. The emphasis at the event was very much on people: on food and drink, including Ljerka's wonderful cakes; on meeting, talk, networking. Attendees included RESG members, tool vendors, authors of other books, family, friends, colleagues, students, researchers. The publisher sold some books and publicised others. The needs of different stakeholders, you might say, were met in a balanced way.

Requirements Engineering will be launched in May, also at UCL as it happens. Professor Axel van Lamsweerde will come over from Louvain to present a summary of his life's work in research. His interest is in the "technical" side of requirements, seeking always to make precise and formalisable (and so, provably correct) every part of a system's specification. He admits he is not so interested in the "social" and "psychological" aspects of requirements work. After all, if you are building a train, it matters that the doors close when the train moves off, and that trains don't drive into each other. It is highly desirable to be able to show that these things won't ever happen.

If "Engineering" techniques can help to achieve that sort of certainty, then we need the "E" in RESG. In contrast, *Discovering Requirements* almost entirely avoids the E-word, barring a discussion of the differences between RE and RM, and whether either of them form part of RD. Where *Requirements Engineering* explains how to ensure the safety of a train and its passengers, *Discovering Requirements* explores how to determine whether a train is an appropriate solution to a transport problem, given that at the start we don't know who the stakeholders are, what they want or how far we can satisfy them.

It's not that RD and RE are in conflict, nor even that one comes before the other, though "early" and "late" are not wholly wrong descriptions. Rather, both aspects are real and necessary, and the tension between them is creative. I hope you'll agree that an RESG that can embrace two such different books must be doing something right.

Ian Alexander, RESG Chair

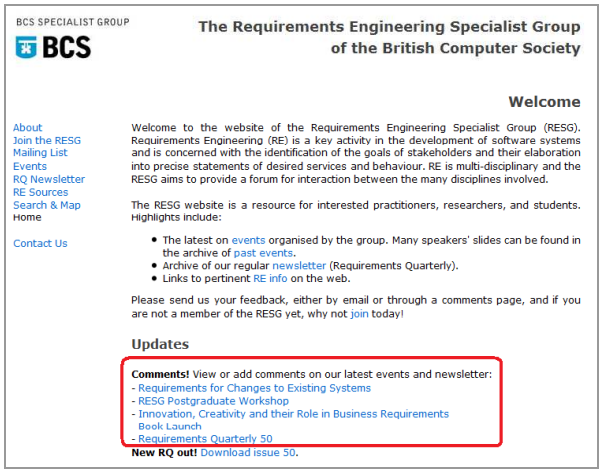
RE-Member

Did You Know You Can Comment On RESG Events and RQ?

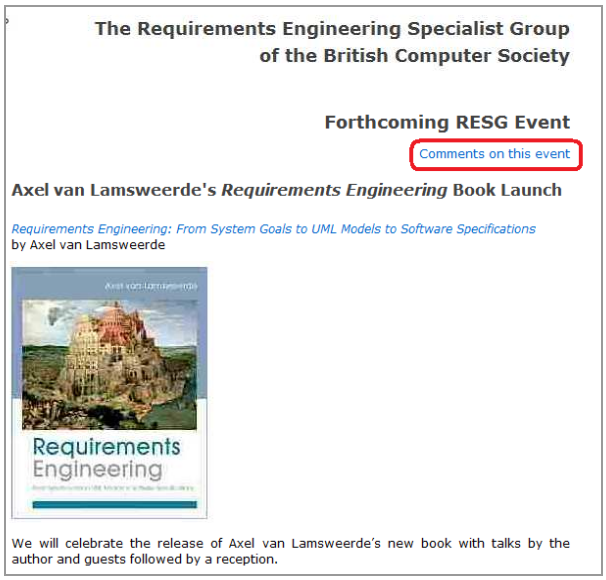
Launched with little fanfare, the new "Have Your Say" feature on the RESG website may so far have skipped your attention. Just as most blogs or news sites let you, the reader, add your invaluable tuppence worth, the RESG site now invites you to leave comments on our recent events and RQ issues. Whether words of affirmation or loud dissent, we want to hear what you have to say about the events we organise and the newsletter we publish. But we hope for more than just this feedback – as important to us as it is. We hope that the comments you leave will become a vital resource to other visitors to the site, possibly sharing your own experiences and insight, or sparking debate in their own right.

Over the next few weeks, we intend to revamp the website to provide more up-to-date content and visitor interaction. The "Have Your Say" feature is our initial step in this direction. If you haven't already found it, follow the steps below!

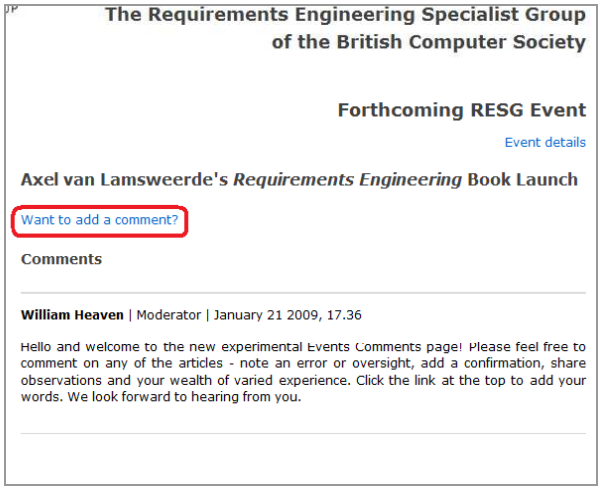
1. Comment pages for recent events and issues of RQ are linked off the home page:



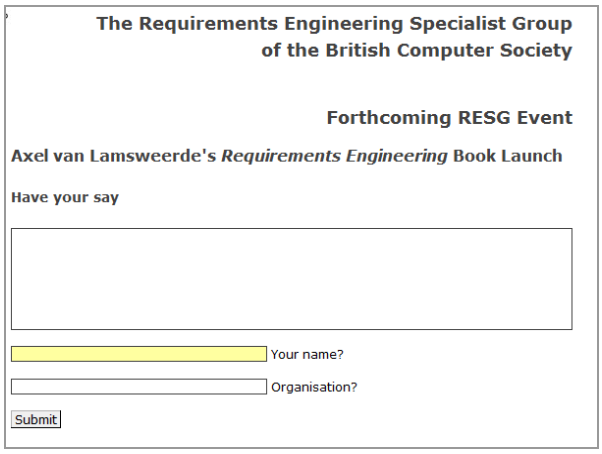
2. Alternatively, each new event and RQ issue also now has a "Comments on this event?" link:



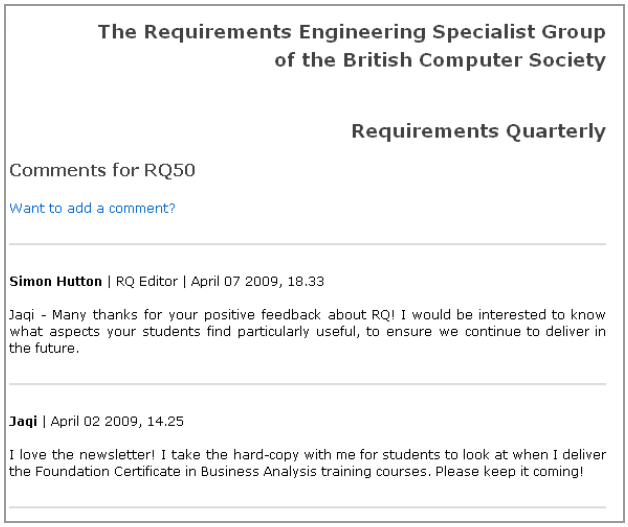
3. This takes you to the comments for that event. If you'd like to leave to add to the comments just click "Want to add a comment?":



4. Leave your comment in the box, with (optionally) your name, and organistion:



5. Your comment will go live as soon as a moderator gets a chance to take a look!



Comments for RQ50 –So Far, So Good!

RQ Distribution

As a reminder, we are now by default distributing RQ by e-mail to reduce costs and time demands on the committee, and to ensure you receive your quarterly newsletter as quickly as possible. If you want to receive a paper copy by opting out of the default e-mail distribution you can by contacting our membership secretary, but bear in mind the costs to our group can be quite high. By reducing the print and postage run we are saving around £300 for every newsletter, and you receive your newsletter a couple of weeks earlier.

Do make sure we have your correct e-mail address and any future updates – contact details are on the back cover of every edition of RQ.

And don't forget that past editions of RQ can be downloaded from our web site at www.resg.org.uk/newsletter.html.

RE-Treats

RESG Event – Emerging Requirements Engineering Practice

6pm to 8pm on 29th April 2009 at The Old Crown Pub, New Oxford Street, London WC1A 1BH

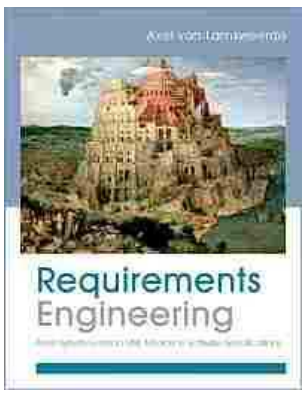
This will be an informal evening get together to talk about requirements. Robert Halligan of Project Performance International will be priming the discussion with an interactive talk. Robert is known internationally for his involvement in the practice and improvement of engineering projects, having spent the past twenty-two years contributing to major systems projects worldwide as a consultant and trainer.

Robert will challenge a number of conventional wisdoms in requirements engineering practice, explain why, and provide alternatives which, he claims, will make the next decade the decade in which requirements engineering practice finally becomes reasonably effective. In doing so, Robert will address topics which include: "requirements creation" versus "requirements capture and validation" elimination of the "then a miracle occurs process" present in most engineering process models confusion of actors defining the whole of "the problem" characteristics of an effective requirements creation approach (aka, design) characteristics of an effective requirements capture and validation approach (aka, requirements analysis) use cases for capture and validation – why they are useful, but nowhere near sufficient Robert will welcome questions and discussion throughout the presentation.

This promises to be a fascinating evening, and details will be provided on the RESG website as they become available – see www.resg.org.uk.

RESG Event – "Requirements Engineering" Book Launch

5:30pm on Friday 8th May 2009 at UCL, London



We will celebrate the release of Axel van Lamsweerde's new book with talks by the author and guests followed by a reception.

The book presents both the current state of the art in requirements engineering and a systematic method for engineering high–quality requirements, broken down into four parts:

- Fundamental concepts and principles including the aim and scope of requirements engineering, the products and processes involved, requirements qualities to aim at and flaws to avoid, and the critical role of requirements engineering in system and software engineering
- System modelling in the specific context of engineering requirements
- Goal–based reasoning techniques to support the various steps of the KAOS method
- Mapping from goal–oriented requirements to software specifications and to software architecture

Talks are from 6pm in the Malet Place Engineering Building, MPEB 1.02
Reception starts at 7pm in The Roberts Foyer, Roberts Engineering Building

Further details are available on the RESG website at http://www.resg.org.uk/event31.html.

Requirements Engineering and Design – A Tribute to Michael Jackson

19th May 2009 - A one-day ICSE'09 Event

This event celebrates Michael Jackson's seminal contributions to software engineering, and recognizes his influence on several generations of researchers. The program features talks by these friends and colleagues, followed by a response from Michael and a reception.

Participants will receive a copy of a new book published in honour of this event, with reprints of Michael's papers and contributions by the speakers.

9:00	Bashar Nuseibeh and Pamela Zave: Welcome
9:15	Tony Hoare: Personal Reminiscences
9:45	Daniel Jackson: A Report on JSP
10:00	John Cameron: Engineering & Business Software
	Worlds and Machines
11:00	Axel van Lamsweerde: From Worlds to Machines
11:30	Anthony Hall: The Quest for Correctness by Construction
12:00	Pamela Zave: Modularity in Distributed Feature Composition
	Current Work
14:00	Cliff Jones: From Problem Frames to HJJ (and its Known Unknowns)
14:30	Bashar Nuseibeh: Abuse Frames: Inferring Security Requirements from Anti-Requirements
15:00	Daniel Jackson: A Notation for Design Rationale
	Response and Discussion
16:00	Michael Jackson Responds
17:00	Discussion
	Reception

Details are at:
www.cs.uoregon.edu/events/icse2009/CoLocatedEvents/

REFSQ'09

The 15th International Working Conference on Requirements Engineering: Foundation for Software Quality

8th – 9th June 2009, Amsterdam

Since 1994, when the first RefsQ took place, Requirements Engineering (RE) never ceased to be a dominant factor influencing the quality of software, systems and services. The RefsQ working conference series has now established itself as one of the leading international forums to discuss RE in its (many) relations to quality.

RefsQ'09 seeks reports of novel ideas and techniques that enhance the quality of RE's products and processes, as well as reflections on current research and industrial RE practices.

RefsQ has a long tradition of being a highly structured and interactive forum. Each session is organised in order to provoke discussion among the presenters of papers and all the other participants.

Ensuring that requirements, and eventually running systems, meet the values of the individuals and organisations that they are meant to serve has always been at the core of RE. Nowadays, continuously changing technology, ubiquitous software, ever-growing system complexity, and unheard of market pressure simultaneously with new business models based, e.g., on crowd sourcing, make the concern for value all the more present and challenging.

The notion of value is inseparably connected to the notion of risk. We are challenged both by product risks that threaten the value we want to achieve with the systems we build, and project risk, i.e. the risk of not achieving the intended value when building a system. Identifying and mitigating risks is a core task of RE.

Further details are at www.refsq.org

CAiSE'09

The 21st International Conference on Advanced Information Systems Engineering

8th to 12th June 2009, Amsterdam

This year's special theme is "Information Systems for Business Innovation". Due to the widespread use of the web, businesses innovate their propositions to customers and come up with new ICT- enabled services. Such innovation requires understanding of the business and of technology in an integrated way. Multi-disciplinary research areas such as Service Science, Networked Enterprises, and Social Networking are paying attention to ICT and business innovation. The special events and invited speakers of CAiSE '09 will shed light on this theme from various perspectives.

Details at http://caise09.thenetworkinstitute.eu

RESG Event – Annual Summer Party

9th July 2009, Imperial College

We should have some live music and good food and drink laid on again. The idea is simply for members to pitch up, eat, drink, and meet like-minded folks beneath the RESG umbrella in a relaxed, social atmosphere. Further details will appear in the next edition of RQ. Professor Bashar Nuseibeh will be whetting our appetites with an opening presentation to prime the event, and most of the Committee will be available to talk about the plans for RESG.

Further details will be available on our web site www.resg.org.

WER'09: 12th Workshop on Requirements Engineering

16th to 17th July 2009, Valparaíso, Chile

WER'09 is the twelfth edition of the Workshop on Requirements Engineering series. The workshop started as a meeting of the Ibero-American requirements engineering community, and now attracts researchers from all continents. Further details are at www.labada.inf.utfsm.cl/wer/

17th IEEE International Requirements Engineering Conference (RE'09)

REQUIREMENTS ENGINEERING:

THE ESSENTIAL BRIDGE

31st August – 4th September 2009, Atlanta, Georgia

The world is becoming ever more dependent on software intensive systems. They are central to our economy, to our society, to the services we depend upon and, increasingly to the very survival of the global ecosystem. Despite many failures, some of them very well publicized, the engineering of such systems has improved consistently over the past few decades. However many challenges remain. Every computer-based system involves relating the myriad, informal facets of the real world to the intricate and formal specifics of a software system. Understanding potentials or details of software systems is not expected of stakeholders, who have their own specialized concerns. Similarly, the eager and technologically capable developers are not expected to understand the nuances of the many domains where software applies.

Requirements Engineering (RE) is the essential capability that can bridge the two perspectives. The RE activity is multi-disciplinary. When defining the requirements of major systems we must bring to bear expertise from a wide range of specialisms such as Human-Computer Interaction, Systems Modelling, and Security. The RE research field builds the effective bridges between these and other sub-disciplines of the Computer Science and Information Systems fields. The many computer-based system needs of business and

society are often contradictory, inadequately defined, and rapidly changing. RE helps stakeholders communicate, helping to reconcile their conflicts, clarify their goals, and reflect their priorities. If our society is to seek a better future we will need all of the models, methods, and tools that RE can provide.

The IEEE International Requirements Engineering Conference provides the premier international forum for researchers, educators, industrial practitioners and students to present and discuss the most recent innovations, trends, experiences and concerns in the field of requirements engineering.

More details will be provided on the conference website, http://www.re09.org.

Business Analysis Conference

28th to 30th September 2009, Radisson SAS Portman Hotel London, UK

The role of the business analyst has evolved significantly over the last couple of years in the UK. The valuable contributions made by the Business Analyst are now becoming recognised at all levels within the organisation to ascertain business problems or identify opportunity areas, to enable successful business solutions and meet the challenging needs of companies today.

This evolution of the Business Analyst role has benefited as organisations undergo significant changes to respond to economic uncertainties. Business Analysts are now increasingly working within areas such as business strategy, organisational change, operational improvements, and new product/service development.

The Business Analysis Conference London 2009 will provide an interactive forum to learn from leading industry experts about the role and future trends of business analysis, what you need to be a high performing and successful business analyst today, success stories from a range of industries, and how to develop a business analyst community within your organisation. The objective of this conference is to increase awareness of the profession and provide professional development opportunities for Business Analysts.

This conference is being held by the UK Chapter of the IIBA, in collaboration with the British Computer Society. Further details are at:

http://www.irmuk.co.uk/ba2009/

RE-Calls

RE'09 - Doctoral Symposium

The RE Doctoral Symposium is an international forum for PhD students working in all areas of Requirements Engineering. The forum brings PhD students together to give them the opportunity to present and discuss their research in a constructively critical atmosphere, and to meet fellow researchers at a similar stage of their career.
To apply as a student participant to the Doctoral Symposium, you should prepare a submission package consisting of two parts, both of which must be submitted via e-mail to Didar Zowghi and Patrick Heymans no later than May 11th, 2009, 23.59 Apia, Samoa time.

Important Dates

May 11th, 2009 Submission deadline
May 31st, 2009 Notification of acceptance
TBD Camera-ready copy
September 1st, 2009 Doctoral Symposium in Atlanta

For more information contact the Symposium Co-Chairs Didar Zowghi at didar@it.uts.edu.au or Patrick Heymans at Patrick.Heymans@fundp.ac.be.

RE'09 - Posters and Research Demonstrations

The 17th IEEE International Requirements Engineering Conference invites submissions of high-quality posters and research demonstrations. The Posters and Research Demonstrations programme complements the main conference programme by offering an opportunity for researchers and practitioners to present late-breaking or as-yet incomplete research results, or significant work-in-progress in Requirements Engineering. We particularly encourage presentations of posters and research demonstrations that are augmented with videos or DVDs.

Important Dates

11 May 2009: Deadline for submission
5 June 2009: Notification of authors
15 June 2009: Camera-ready abstract due

For details contact the Posters And Research Demonstrations Co-Chairs Andrea Zisman at a.zisman@soi.city.ac.uk or Mehrdad Sabetzadeh at m.sabetzadeh@cs.ucl.ac.uk

A Requirements Challenge

Steve Easterbrook is Professor of Computer Science at the University of Toronto. He recently issued a challenge that I include in RQ to ensure it reaches the widest possible audience. There may be an opportunity in the future to share some of the responses through RQ, so please consider participating if you feel you have something to offer.

Steve writes…

Here's a challenge for the requirements modelling experts. I've phrased it as an exam question for my graduate course on requirements engineering (the course is on hiatus, which is lucky, because it would be a long exam…):

Question: The governments of all the nations on a small blue planet want to fix a problem with the way their reliance on fossil fuels is altering the planet's climate. Draw a goal model (using any appropriate goal modelling notation) showing the key stakeholders, their interdependencies, and their goals. Be sure to show how the set of solutions they are considering contribute to satisfying their goals.

The following documents may be useful in answering this question:

(a) A outline of the top level goals (http://www.monbiot.com/archives/2007/12/04/what-is-progress/)

(b) A description of the available solutions, characterized as a set of Stabilization Wedges (http://www.princeton.edu/~cmi/resources/CMI_Resources_new_files/CMI_Wedge_Game_Jan_2007.pdf)

(c) A domain expert's view of the feasibility of the solutions: (http://climateprogress.org/2008/03/31/is-450-ppm-carbon-dioxide-politically-possible-1/)

(d) An initial identification of the actors (http://maps.grida.no/go/graphic/all-actors-towards-a-climate-neutral-society1)

For further details, Steve's blog is at :

www.easterbrook.ca/steve/

2nd International Workshop on Requirements Engineering and Law (RELAW)

The 2nd International Workshop on Requirements Engineering and Law is a multi-disciplinary, one-day workshop that will bring together practitioners and researchers from government, industry and academia to investigate the challenges to ensuring that software systems comply with the law. Recent compliance challenges include balancing privacy and security, patient medical records, corporate governance and accessibility. Authors are invited to submit papers that address one or more of the following topics of interest:

* Identifying, prioritizing and integrating laws and jurisdictions

* Legal requirements acquisition, specification, analysis and validation

* Formal and informal modelling of laws, policies and requirements

* Traceability and alignment between laws, policies and requirements

* Coordinating requirements change and the evolution of law

* Introducing existing products and services into new jurisdictions

* Requirements verification; documenting/auditing evidence of compliance

* Acceptable degrees of compliance assurance and system certification

Important Dates:

10 July 2009	Deadline for Submissions
27 July 2009	Notice of Acceptance
3 August 2009	Camera-ready Due
1 September 2009	Workshop Date

Further details can be found at:
http://www.csc2.ncsu.edu/workshops/relaw/.
Please send any inquiries to: relaw@csc.ncsu.edu

RE-Course

Mastering the Requirements Process

15-17 September 2009, London. Presented by James Robertson, Atlantic Systems Guild

This 3 day seminar & workshop presents a process for eliciting requirements, testing them for correctness and recording them clearly, comprehensibly and unambiguously.

Details at www.irmuk.co.uk/1/

Introduction to Requirements

8-9 September 2009, The IET, London, presented by Ian Alexander, Scenario Plus

This two-day course with exercises gives an overview of the requirements process, with practice in techniques for discovering and managing your requirements.

Course details at http://www.scenarioplus.org.uk

Bookings at http://www.theiet.org/courses

RE-Writings

The Stepmother's Side of the Story

A. Egemen Yilmaz, Havelsan Inc.

A very rich man, who wants to be known as a capable writer even though he has no talent, hires a very famous novelist to write a novel on his behalf. He states his requirements to the novelist as follows:

> "Nowadays, books on mythology, God and religion are very popular; so *the book shall have religious components*. On the other hand, people always like to hear about sexuality; hence, *the book shall have sexual content*. Stories about ordinary people are boring; so *the book shall be about the lives of noble people*. I like mysterious stories and endings; the final of *the book shall lead the reader to curiosity*. And most important of all, people get bored of long novels, and so do I. Hence, *the text of the book shall be as short as possible*."

A couple of days later, the rich man received the following "novel" from the novelist:

> *" 'Oh my God!...' yelled the Duke; 'with whom did the Duchess sleep last night?' "*

Lots of articles in the requirements engineering area include the failure stories of big projects or some negatively impressive metrics about them. In such articles, where the causes of failures are distributed among the project stakeholders, usually the customer / end user gets the biggest share of the blame for not stating (or not being able to state) his own needs clearly. In addition to these, in our daily life we (engineers and developers) usually complain about the customer (especially if the customer is the government or the army) in all our conversations, even during our tea/coffee or lunch breaks, and declare them as the scapegoat of our not-well-going projects. On the other hand, there is an adage in Turkish, which can be translated to English as follows: "Stick the needles to the others, but preserve the biggest one for yourself", which reminds the importance of self-criticism in life. It is time to admit that sometimes we (as developers) are the ones who play tricks, take enormous risks, and cause the eventual failures.

Actually, what inspired and encouraged me to write this article was Richard Veryard's interesting and impressive article appearing in the 29th issue of this newsletter during August 2003. I just wanted to add some more thoughts and create a funny polemic on this nice subject.

In this article, which I have encountered recently while browsing the archives of the newsletter, Mr. Veryard brilliantly approached to the well-known story of the Pied Piper of Hamelin (or *Hameln* in German). Ridding the rats out of the town is humorously treated as a project; of which the citizens and the Piper are the stakeholders. Mr. Veryard identifies the main problem (the Piper being not paid) as a significant shift in the customer preferences, which is a "requirement change"; and points his fingers to the citizens (i.e. the customer) as the faulty stakeholder.

There is a children's book series by Walt Disney, called "My Side of the Story", which rephrases the classical stories from the bad guy's (which is usually the stepmother) point of view. With a similar approach, I would like to claim that the Piper has greater responsibility on the failure. Knowing that he does not have a trustable appearance, and an unusual/questionable technique; before proceeding, he should have done some conceptual proof (i.e. prototyping and demonstration) to the citizens in order to verify that his solution was acceptable. Somehow, he could have picked a small part of the rat population, dropped them to the water, and requested the approval of his approach after showing the citizens all possible consequences of it. Moreover, he could have defined a billing event on this activity, which would have partially solved "his" main problem, namely the cash flow.

Mr. Veryard claims that "… this solution is achieved without the use of noxious chemicals or other environmentally harmful side-effects." I strongly disagree with this statement. Could you imagine what happened when the citizens woke up one day, and saw thousands of rat corpses washed upon the shore? Maybe, that was the main reason for the Piper being not-paid (There might have been violated safety requirements of the customer addressing this issue; which has been hidden from us for years). Going to extremes, this might be the actual reason for the sudden disappearance of the children in town (i.e. deaths due to cholera or any other contagious disease caused by the rat corpses). Hence, the failure of the project might be a complete responsibility of the Piper; the rest of the story (the children of the town being magically kidnapped by the Piper for revenge) might be a setup and mis-advertisement of the Piper for creating an excuse for the children's disappearance in order to preserve his career. Unfortunately, everybody had the tendency to blame the customer and believe the Piper as usual.

Nevertheless, this story should be a lesson for all stakeholders "to be more careful about the coverage of both the functional and non-functional requirements by the design and the implementation". Moreover, it should be noted that "the lack of conceptual proofs of new approaches might yield disasters".

Unfortunately, mis-advertisement or "customer abuse" by the developing stakeholders is not infrequent in the market. In the novel example given at the beginning, what the novelist did was nothing but getting advantage of the fact that there is a distinction between "verification" and "validation", which is not known by his client. He knows that his client cannot say "Okay, this satisfies all my requirements; but this is not a novel at all!" Or coming back to children's stories; after getting bored of the Little Prince's (customer) numerous requirements about a sheep to be drawn, the

narrator (developer) draws only a box (claiming that the sheep is inside and all the requirements are implicitly satisfied), and surprisingly performs the acceptance. This is again getting advantage of the customer's confusion.

The Brave Little Tailor, who kills seven flies at one strike and wears a belt writing "Seven at One Blow" (not clearly specifying what he has killed), gets the job for killing the troublesome ogre. This is nothing but another example for the mis-advertisement of the capabilities (and even unfair competence with the other developers' points of view). Many other examples could be extracted in the children's stories, which are already aimed to be including lessons to all of us.

Let me close the discussion with the following short poem:

The Stepmother is not necessarily a bad person;

She is always right from her own perspective,

And there are cases that she might be right overall.

No one could argue with Cinderella's stepmother's objective (requirement prioritization)

While taking her own daughters to the ball.

References

R. Veryard, "Requirements Engineering as if Stakeholders Mattered", *Requirenautics Quarterly*, Issue 29, pp. 8-10, August 2003.

You can read Richard Veryard's original article by downloading RQ29 from our web site at http://www.resg.org.uk/newsletter.html. And don't forget that you can comment on this article or add to the theme through our comments page accessible through the RESG web site home page.

Simon Hutton, RQ Editor

Seven Traceability Principles

Graham Berrisford, Avancier Ltd

All IT systems are complex systems, composed of many (well-nigh innumerable) and interrelated items. The bottom-level description is composed of detailed process, data and configuration statements that computers use to make and run the operational system. You might include test cases in this ultimate description. Usually, the bottom-level description is too long, too distributed, too incomprehensible and too complex to be usable by users, managers, analysts, architects and the like. That is why we must maintain higher level system descriptions, of course.

We work with many examples of higher and lower level descriptions. We separately document higher-level business goals and lower-level application requirements - higher-level requirement statements and lower-level solution items - higher-level use case descriptions and lower-level test cases - higher-level logical entities and lower-level database tables – higher-level architecture building blocks and lower-level solution building blocks, etc. Each of these descriptions is a configuration in its own right; it should be internally consistent and serve a purpose on its own.

In short, to fully specify even a small system, we describe it in several ways and at several levels of abstraction. Managers and customers expect us to maintain full traceability between these descriptions. Yet a couple of articles in recent BCS Requirements Engineering Quarterly newsletters have drawn attention to the practical difficulties of maintaining such documentation. One article even suggested we questioned why traceability is needed before going overboard with it.

This article sets out practical reasons why complete traceability records will never be completed and maintained at the level of detail naïve managers and customer might expect. However, it also suggests some ways to optimize configuration management and enable enterprise architecture.

Principle 1: Every higher level of description conflicts with SPOD

It is one thing to produce a high level system description *before* lower level descriptions exist. This is a necessary milestone on the way to building and implementing a system. It is quite another thing to maintain a higher level system description *after* lower level descriptions have been completed and the system has been implemented.

An implementable system is a description of items and inter-item relationships at the level computers can interpret. Any more human-friendly and more abstract description is likely to refer to at least some of the items and relationships contained in the bottom-level description.

For example, at the implementation level, a nest of web pages is connected by hyperlinks. To make change impact analysis easier, we might list web page names as rows and columns in a spreadsheet and record the presence of links in the cells of the spreadsheet. This spreadsheet is a higher-level description of the implementation.

Notice that the spreadsheet duplicates information recorded in the implementation-level description. This runs counter to the Single Point of Definition (SPOD) principle, which tells us to record an item or relationship once and once only. Each higher level of description increases our maintenance workload, and increases the risk that our documentation records get out of step.

In practice, most higher-level descriptions (e.g. requirements catalogues, logical data and process models) are left on the shelf after a while. And where they are maintained, the upper and lower levels of description are rarely maintained in perfect correspondence, meaning that traceability between levels cannot be 100% reliable.

A first lesson is this: if the names of items and their relationships are clearly visible in the implemented system, then you may well find higher level documentation of the same information proves redundant. Don't maintain a higher-level description unless it really helps.

A second lesson might be this: if a higher level description is to be maintainable, it should be more abstract by an order of magnitude. Our knowledge of how to maintain abstract descriptions remains weak. The Zachman Framework suggests maintaining descriptions at five levels of abstraction from implemented systems. Has anybody actually maintained that many levels of abstraction in practice? Along with traceability records between all the levels?

Principle 2: Full traceability is impossible

IT system documentation comprises several related but separable configurations. There are distinct structures of goals, requirements, user interface components, software components, database tables, test cases, servers, networks etc. These somehow add up somehow to one humongously large and complex configuration. Change management requires impact analysis. Impact analysis requires all dependencies between bottom level items in different structures to be known, so when one item is changed, we can find all the other items that may have to be changed also. But given the number, size and complexity of the many separate configurations we have to manage, it is in practice impossible to maintain documentation that fully traces all the relationships between all the items at a bottom level.

So, impact analysis will always depend at some level on the knowledge that human beings have about the items within a configuration and their inter-dependencies. Let us be honest: we don't really expect to document complete traceability. We ought to be clearer about what level of traceability documentation can possibly be maintained, how it helps, and how much it costs.

Principle 3: A configuration steward can manage a lot in their head

A configuration steward is the individual who looks after the integrity of a configuration. A configuration steward can hold quite a large configuration in their head. I don't mean they can remember it exactly; I mean they can become so familiar with a structure of related items that they can maintain it with little or no recourse to documentation of relationships between items. If you ask them to make a change, they know which items of the structure need attention. If you ask them to change one item, they know which other items need attention. And where they aren't sure, they know where to look in the structure to find out, very quickly.

I don't claim to be mathematically precise about this. But suppose we say:

> The size of a cohesive configuration is C, the number of items in it.
>
> The complexity of a cohesive configuration is R/C, where R is the number of relationships between items.
>
> The maximum a configuration steward can hold in their head, is L, which measurable as C * R/C.

I suspect L is usually quite large, could be 1,000 or more. But when a configuration grows larger than L, there is a huge step change in the time and cost of change management. The configuration steward becomes unreliable. All inter-item relationships must now be documented in a way that anybody and everybody who changes the configuration can both use and maintain.

Lesson: Given a configuration larger than L, divide it into loosely-coupled substructures, each maintained by one configuration steward whose role is to 'know' that substructure and use this knowledge to do impact analysis on it.

Principle 4: Raise the level of traceability

Naturally, we should strive to keep distinct configurations loosely-coupled, to minimise the dependencies between items in them. Suppose two configurations, each at the limit L, are each managed by one configuration steward. If there are only a few mappings between items in the configurations, then changes will remain manageable. If there are many mappings, then the configuration stewards will be overloaded, and change management will become slow and costly. (And it is hard to see how bringing in a 3rd party to document the mappings would help.)

Given that two configuration stewards do have to maintain two tightly-coupled configurations, then they can minimise the work to maintain traceability records by raising the level of traceability to a higher level of composition, since each configuration steward can readily do impact analysis in their head when a change happens.

They can raise the level of traceability records in either a lop-sided or in a balanced way. They can:

Record an item in configuration A depends upon the whole of configuration B. So they don't have to update the record every time items in B are reorganised.

Record the whole of configuration A depends on an item in configuration B. So they don't have to update the record every time items in A are reorganised.

Reorganise both configurations into substructures - and traceability records between the substructures.

The last of these means (again) grouping closely-related items into substructures that are (in the classic manner) internally cohesive but loosely-coupled one with another.

Principle 5: Configuration management of operational systems is paramount

Though we describe systems in several ways and at several levels of abstraction, the ultimate description is the bottom-level one, composed of statements that a machine uses to make and run an operational system. Let us face it. Business managers and users don't care if operational systems are out of step with higher level descriptions, but they care a great deal if the operational systems stop working. Inevitably, when a change is made, change management of the operational systems (which system users are using) is far more critical than change management of more abstract descriptions (which only system designers use, or might use).

We can argue until we are blue in the face that system descriptions should be updated before system implementations. The fact is, the descriptions will not be kept up to date with live systems unless there is an extremely disciplined and effective process for doing this. And doing it after the fact (by some kind of reverse engineering) may be the best we can hope for.

Principle 6: Mappings between levels of abstraction should be moderate and automated where possible

As soon as a logical model has been turned into a physical solution, further effort to maintain the logical model or the mappings will prove difficult. The closer mappings are to one to one, the less people see value in maintaining the higher level of description.

The further mappings are from one to one, the less people are able to understand and maintain the relationships.

The corollary is that people are only willing and able to maintain traceability relationships between levels of description where the complexity of the mapping is moderate:

- complex enough to make the maintenance of the more abstract model worthwhile but also
- simple enough to make the maintenance of mappings feasible.

Lesson: If the mappings between requirements and solutions are too complex to make maintenance possible, then you might (again) group items into substructures, so you can raise the level of traceability. Better still, reduce the labour by investing in automation of forward and reverse engineering between the two levels of abstraction.

Principle 7: Application Designers must populate the bottom level of the Enterprise Architecture model

Enterprise Architects bring even higher levels of abstraction to the configuration management issues mentioned above. Enterprise architects manage the applications portfolio in which the items are whole applications. Software/application architects manage the configuration of components within an application. In theory, there is a continuous top to bottom decomposition structure. In practice, the two levels of configuration are rarely if ever recorded in the same repository.

At the strategic level, at a high level of abstraction, enterprise architects map business functions to data and manage the applications portfolio. Enterprise architects find it difficult to populate their models to the level of detail that is of interest to the application designers, and to keep their models in step with application design models. The result is that application development teams regard EA as an ivory-tower activity of no relevance to them.

At the project level, software architects manage the design of a given application. They specify application behaviour using Use Cases and/or user interface prototypes. Neither of these techniques maps business functions to data. But since applications are largely about capturing and providing information, this leaves users and other design reviewers with an imperfect picture of the design's intent. It also makes it hard to pick out the key elements to store in the EA model.

We need a new way for application designers to document application behaviour and (at the same time) gather the information to populate the EA model. We need to record screen flows and map user interface functions to data in a way that is independent of user interface appearance and implementation detail. The resulting models will be useful as a description of legacy system behaviour and/or target system behaviour. And in completing these models, application designers will automatically document the bottom-level functions and data types on top of which the EA model should (in fact must) be built.

Concluding remarks

Traceability is useful but difficult. There are many practical reasons why traceability records will *never* be completed and maintained at the level of detail you would like for change impact analysis. However, there are some ways you can use to optimize configuration management and enable enterprise architecture.

Graham Berrisford is the director of Avancier Ltd, which specialises in training enterprise and solution architects. He has been using and teaching systems analysis and design methods for 30 years. His most recent public works are the library of papers published on the web site Avancier.co.uk and his contributions to the new ISEB certificates in Enterprise and Solution Architecture.

RE-Flections

Brown-Field RE: a Discussion Event

12 February 2009, Misys, Paddington

This was an experimental event held at Misys' invitation at their new HQ building in West London. The RESG's usual haunts are university lecture halls, so this was a new departure for us.

Phil Cantor welcomed everyone to Misys and explained why financial software with its complex mass of constraints poses a challenge to business analysts.

Ian Alexander gave a short talk by way of introduction, to set out the problem. Much RE (and Systems, and Software) literature assumes green field; but most projects add to existing brown-field systems, whether by writing some extra use cases, by processing some change requests, or simply by giving the maintenance team a pile of software problem reports and letting them get on with the job.

For example, a consumer electronics firm creating a product like a mobile phone is in a way free ("green field") to develop whatever it likes. But in reality it is constrained by many kinds of constraint including existing system interfaces, and by user expectations conditioned by the user interfaces of the existing product line, so the field is more brown than green

Or a new commuter railway or tram may appear to be a completely green-field choice – any kind of vehicle could be used on any desired route. But there are intense resource conflicts in many urban areas, and many stakeholder groups compete for their points of view to be heard. Further, the existing transport and general urban infrastructures tightly constrain both routes and interchanges. The field is mainly brown.

And as a final example, new tasks for a financial system may freely be specified as new use cases; but these are at once constrained by existing database design, the existing network of terminals and servers, the existing codebase, and not least the wishes of all the different kinds of stakeholder, including clients of different kinds who would each like custom products, and the business which would like a single reusable product. It is definitely a brown field.

Perhaps the question, Alexander suggested, is not what is special about brown-field RE, but whether green-field RE exists at all. He listed some brown-field challenges:

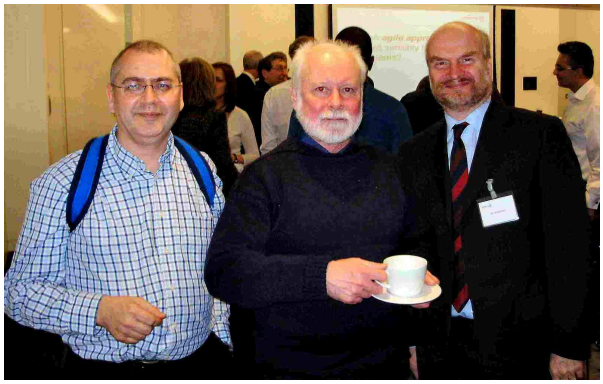
- Legacy features
- Complex requirements with many associated details
- Multiple constraints from existing system interfaces
- Troublesome non-functional requirements
- Pressures from multiple stakeholders
- Loss of organizational memory
- Large numbers of (possibly conflicting) change requests

RE as currently practised has attempted to address these concerns since its inception. For example:

- Legacy features and complex requirements are addressed by traceability, typically with requirements tools, and data models.
- Complex requirements are addressed by scenarios and use cases which attempt to relate many activities together using story, to achieve stated functional goals.
- Multiple constraints and troublesome non-functional requirements are addressed with goal and context modelling.
- Pressures from multiple stakeholders are addressed with stakeholder analysis, trade-offs, and prioritisation.
- Troublesome non-functional requirements and loss of organizational memory can be addressed by agreeing acceptance criteria and quality-of-service measures to define exactly what performance must be delivered and maintained throughout the life of a product or service.
- Loss of organizational memory is addressed by documenting project rationale to capture decisions, and by the project dictionary to capture terms with special meaning within the project.
- Large numbers of (possibly conflicting) change requests are handled by (trade-off) analysis and prioritisation.

In other words, there is a many-to-many mapping between elements of the requirements problem and requirements engineering techniques.

Concluding his talk, Alexander suggested that we could imagine a matrix of requirement elements (such as goals, scenarios, rationale, definitions, measurements, priorities) against discovery contexts (such as individual interviews, group workshops, existing artefacts, trade-off studies). Each product might well have a unique pattern of activity in such a matrix; but it might well be that large, complex brown-field projects had to work more with existing artefacts, interface constraints and trade-offs, and less with individual stakeholders' goals and opinions.



Phil Cantor, Ian Graham, Ian Alexander

Phil Cantor of Misys proposed a list of questions that he was interested in for financial software products:

- To what extent should existing systems be documented in (change) specifications?
- Can agile approaches be used sensibly for these systems?
- What level of detail for intrusive but simple changes?
- What level of granularity for a single requirement?
- Can modern requirements diagram techniques be effective for defining changes?
- How to handle multiple customers (package) versus a single one (bespoke or in-house)?
- Should requirements be captured per venture, per investment, per release or per release component?
- How should non-functional requirements be documented
  (eg platform, performance, reuse, architectural probity)?

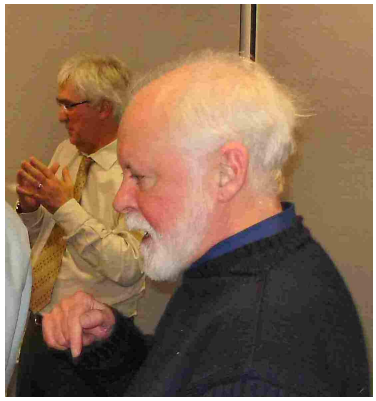
Participants reporting their findings

There was a wide-ranging discussion about several of these, and the pros and cons of different approaches – such as use cases, UML modelling, prototyping – were argued.

The difficulty of applying other worthy techniques for discovering and especially for documenting brown-field requirements – e.g. creating a project dictionary, recording rationale, designing for reuse – was hotly debated. Essentially everybody agreed that these were good and right, but while a project is "under the cosh", anything off the shortest path from the businesses goals to realisation in code is likely to fall by the wayside: causing endless problems later. Perhaps the only cure is educational rather than technical: business leaders need to be made aware of the balance between short-term goals and longer-term business advantage.

The evening was deemed interesting and enjoyable by students, business analysts, experienced RE-ers and greybeards alike.

Among the audience was Ian Graham, author of *Requirements Modelling and Specification for Service Oriented Architecture* (Wiley, 2008). He confessed that he had worked many years ago on a financial system called Midas…



Ian Graham making a point

It was certainly a different audience mix from usual: some familiar faces, but plenty of people from the financial industry. Inevitably, language varies with background: business analysts say "work system" and "technical system" where some of us would say "system" and "product", others perhaps "the work" and "the system". There is ample scope for mutual learning.

The RESG committee members present came away with the definite feeling that we have much to offer to the financial community – a rich literature; many years of research experience; well-informed industrial practice and much experience with requirements tools and techniques. The financial companies still have huge resources, despite the present troubles, and experience of an extremely brown field of software development. We are grateful to Misys for hosting this meeting and introducing us to a different community of practitioners.

The meeting adjourned to the pub in Sheldon Square for much more detailed analysis.

The discussion, with a list of tips that was started during the event, continues on a prototype moderated discussion list at www.resg.org.uk under the 'Brown-Field RE' event. Please join in.

RESG PhD Workshop

6th March 2009, Imperial College London

The RESG PhD Workshop is designed to give new and not-so-new PhD researchers a chance both to present their work to RE peers (in whatever early stages it may be) and to get invaluable constructive feedback from leading RE academics. This year we were very pleased to get Emmanuel Letier (UCL) and Bashar Nuseibeh (Open University) to come along (Anthony Finkelstein, provisionally pencilled in, unfortunately had to pull out, with Emmanuel generously taking his place with 2 days' notice). Many thanks to them!

The RESG has run PhD workshops before but not for the past few years: all the people we had met at these previous events would have moved on, theses submitted, to a rich and rewarding post-doc life. (In fact, if there are any PhDs reading this who came to one of the previous RESG workshops, please drop us a line – we'd love to hear what post-doc adventures an RE PhD might have). So, while we knew there must be new PhD researchers out there, scattered across the universities of the country, we initially wondered how to contact them. We'd tried and failed in the past to reach many of these researchers by broadcasting, so this year we tried something a little different. Luckily, the RESG has always had good academic contacts, so we trawled through the websites of known RE academics looking for the websites of their (as yet) unknown research students, and then emailed these directly, inviting them to take part. (If we missed you, we apologise: again, please get in contact if you'd like to take part in future!)

This simple technique proved very successful and we ended up with an excellent turnout and a fledgling new community to keep in contact with. Thanks to all those who attended, especially those who presented their work. To give a brief idea of the breadth of the new research being done today in RE, we reproduce abstracts of the speakers' presentations below.

IRIS (Integrating Requirements and Information Security)
Shamal Faily (Oxford)

The failure to design for changes to contexts of operation has been widely reported, with consequences ranging from loss of information through to loss of life. As reports of such failures become more prevalent, there is a growing need to explore the relationship between security and its socio-technical contexts. A better understanding of this relationship could lead to important insights into designing and deploying security solutions that are effective at mitigating risks and appropriate to their context of use.

Recent empirical research on cultural contexts within the NeuroGrid e-Science project found that a myopic view of security engineering, coupled with conflicting perceptions of security by different organisational sub-cultures can lead to a significant source of requirements conflict. This research also highlighted the importance of understanding different roles and responsibilities at play within different contexts. This presentation introduces IRIS (Integrating Requirements and Information Security), a framework for supporting integrated requirements and risk management for variable contexts of operation.

We present a meta-model for integrated Requirements and Risk Management, and compare and contrast this with similar models in the security requirements engineering literature.

With the aid of contemporary case study, we then illustrate a number of features of IRIS. These include an approach for security property sensitive quantitative risk assessment, the use of semi-formal requirements grammar to introduce changes to the context of operation, and the visualisation of requirements and risk management artefacts.

Managing Requirements Change

Soo Ling Lim (UCL)

The requirements of a system continue to change throughout the system lifecycle. Requirements change at different rates. Different types of changes have different effects on the system and are dealt with differently. Existing requirements engineering methodologies organise requirements based on their functionality and not on how likely they change. As a result, stable requirements are mixed with or embedded in volatile ones. This causes stable requirements to obstruct changes to volatile requirements. Requirements management becomes inefficient and error prone when frequent changes occur.

I propose that requirements can be classified into layers that evolve in different timescales. This encourages the separation of requirements based on rate and type of change. As a result, changing the requirements in one layer does not affect the other layers. It is also not obstructed by the other layers. The classification can be used to develop a method to organise requirements for managing future changes.

Requirements Traceability for Multi-Agent Systems
Gilberto Cyneiros Filho (City)

Multi-agent systems have emerged as one of the most important areas of research and development in distributed systems in the 1990s. Multi-agents arise as promising new software paradigm capable to provide solution to the need of applications that requires run in an open, complex, dynamic, and distributed environment. Despite its potential, in practice multi-agent systems is rarely used and it has been widely outpaced by the service-oriented architecture. One key reason for the slow adoption of multi-agents by the industry is the lack of tools to support the development of large-scale multi-agents systems.

Traceability has been recognised as fundamental to support the activity of project management and software maintenance and it can be used to understand the relationships that exist between artefacts created during the software development process and has been used as mechanism to support verification, impact analysis and change management and to understand the

evolution of an artefact. The heterogeneity and huge amount of information involved in the development of multi-agent system produce a complex number of relationships that brings up the necessity for automated support for traceability.

We propose a rule-based approach to support automatic generation of traceability relations and completeness checking between the various models generated during the development of multi-agent systems. Our work concentrates on goals and business models represented in i* and design models represented in Prometheus and code in JACK. The rules are represented in an extension of XQuery and the models in XML.

A prototype tool has been developed to assist with evaluation of our work. We also created a traceability model to provide semantics for the different types of relationships in order to support richer analysis about the relationships.

A Novel Architectural Model used to Investigate the Insider Threat

Clive Blackwell (Royal Holloway)

The insider threat poses a significant and increasing problem for organisations. This is shown by the regular stories of data loss in the media such as the 25 million personal records mailed out on 2 CDs by Revenue and Customs in the UK. There is a need to provide a complete and consistent defence from insider attacks because their legitimate access and knowledge of weaknesses means that no single protection mechanism will be sufficient.

We have developed a three-layer architectural model to examine the security of complex systems holistically. Our model covers physical and social level attacks as well as computer and network incidents along with the corresponding defensive mechanisms.

Organisations have positive functional goals that are needed to fulfil their mission and make money that we consider conceptually at the social layer. The security requirements are only necessary to the extent that they efficiently and effectively support the organisational goals in the presence of malicious attack. The security controls are usually lower-layer mechanisms that provide the confidentiality, integrity and availability of system resources. Our model gives visibility and traceability to the organisational requirements at the social layer by mapping them to their implementation and possible abuse at lower levels.

The model is used to determine typical scenarios in the main types of insider attack that cause damage, fraud and theft. We examine the different stages of attack that enables systematic defensive protection by limiting access, constraining the use of the target and limiting the impact of successful attacks. The possible goals of attackers and how to reduce their motivation by persuasion or deterrence are also considered. We show destructive attacks within a systematic tabular classification along with their corresponding defences, which allows the prioritised selection of defensive controls in line with business need.

Our practical architecture appears to have widespread application in other complex systems, as it allows the analysis of systems in their entirety including human and physical factors, not just as technical systems alone. It has already been used to investigate the electricity grid, part of the critical infrastructure, with its widespread scope and weaknesses at all layers to help ensure it meets its vital requirements such as meeting energy demand. It has application to complex financial systems such as banking networks where weak procedural and physical controls are usually exploited rather than the technical controls such as cryptography.

We are formalising the model with a new process calculus called bigraphs that represents both the physical and logical aspects of systems. We take existing structural and behavioural models in UML, or diagrammatic representations in our architectural model and semi-automatically translate them to bigraphs that are then executed to find exploitable weaknesses that breach organisational requirements.

Reputation-based Message Routing: a Flexible Workflow Process

Ben Jennings (UCL)

The integration of human agents within workflows is one fraught with subtle complexity. Such complexity is due to the very nature of human agents, one of unpredictability. When considering flexible workflows, a factor of which may be unanticipated deviations, utilising human agents in such processes has desirable characteristics as more flexible outcomes are possible.

Contemporary solutions in the Service Oriented Architecture (SOA) space have looked to fit human agents into top down generated, hierarchical abstracted generalised web services. In flexible expertise driven domains with evolutionary business processes, SOA abstractions fail to capture the nuance of human expertise and human interaction. This talk will present a bottom up approach, via a Reputation based mechanism, to find the most appropriate human agent, or data object, for the flexible workflow.

Support for Collaborative Elaboration of Requirements Models

Camilo Fitzgerald (UCL)

Collaborative environments are particularly well suited to many software development projects, as Mozilla's Firefox and Ubuntu have shown. Almost no work has been done, however, to make use of these environments to aid those working within them with requirements development and management.

Case studies on the Firefox and OpenOffice projects have unveiled key problems with requirements management in collaborative environments that need to be addressed. One such problem is that of agreeing on a requirements model when multiple actors are involved in discussions. Currently, requirements are developed through free-for-all discussions in Issue Tickets, which are often pathological, unproductive and can take a long time to reach conclusion.

Requirements models are subsequently constructed, which are then frequently subject to same form of discussions.

A solution to this problem would aid those working on requirements in collaborative environments to hold arguments that are more constructive, and lead to preferable solutions in a shorter amount of time. It is therefore proposed that support is given for the collaborative elaboration of requirements by allowing users to annotate goal-models in tandem with their arguments, and notifying them of the subsequent actions available to them that are likely to make discussions progress.

This will provide a significant contribution to an area largely missing from the requirements literature. Almost no work has been done to support the process of developing requirements models in a multi-party collaborative environment. The provision of techniques that guide multiple users' moves in creating and arguing about elements of goal models would fit this niche, and be of great benefit to those working with requirements in all kinds of collaborative software development environments.

RESG Book Launch - Discovering Requirements

2nd April 2009, UCL

This RESG Sponsored event was held at UCL's Engineering Café, opposite Waterstones on Torrington Place. It proved to be the ideal venue – an open but reasonably private space with the bustle of London just outside and clearly visible through the street level windows. The event was organised by RESG to mark the publication of *Discovering Requirements*, authored by Ian Alexander and Ljerka Beus-Dukic.

Around 40 of us and were treated to a combination of chat, food and drink, with thanks to Wiley for their support – in particular the delightful Kate Batchelor who represented the publishers.

Kate Bachelor of Wiley

The event proved to be a wonderful opportunity to hear the story behind the book. This helps the reader to understand the motivation of the authors, and to better apply the ideas. It was also a very good social event – we had excellent home-made cakes, a first class buffet and an almost limitless supply of wine and beer. Couple this with a healthy mix of enthusiastic requirements engineers from industry and academia and you have a recipe for a very enjoyable evening!

Ian with James, Mav and Simon

After some social networking Ian and Ljerka treated us to a relaxed but insightful presentation of the book. The idea to write the book started a few years ago when Ian and Ljerka lectured together. Ljerka's course at Westminster University included a project, during which the students had to discover the requirements. Ian and Ljerka were frustrated that there wasn't a useful text book that they could use. There is plenty of guidance on analysis and tools, but little on discovery. So they set about plugging the gap by producing the book they wished had been available.

Ian emphasised that requirements are a key aspect of a project, and are vital to make sure the money is deployed properly – that we spend what we have on what we want to do. The relationship between the requirements and the design was a recurring theme, and a welcome one. Requirements exist in the real world and are often more useful if they have real world constraints and design built in as early as possible.

Ian also reflected that requirements discovery is as much a social task as well as a technical process. Requirements will include conflicts and assumptions that will need to be worked through with the relevant stakeholders. In essence, Ian asserted that discovering requirements (and all requirements engineering) is actually a relatively simple activity, but emphasised that it is not easy. He used the hand built Indonesian outrigger boat shown on the cover of the book as an analogy to explain.

Indonesian Outrigger

The boat shown is a relatively simple design, with a narrow hull supported by outriggers. It was made by hand, on a beach, entirely from wood using nothing more complex than hatchets and hand-drills. No nails, cranes, welders or design engineers were necessary, How do you make a boat strong and watertight using such simple means? It isn't easy. Simple, but not easy.

Ian then introduced the Discovery Matrix that provides the structure for the book. The matrix shows requirements elements (including Stakeholder Analysis, Goal Modelling, Scenario Analysis, Measurement, Priorities) against requirement discovery contexts such as individuals, groups and things.

Ian explains the Discovery Matrix

Each element of the Matrix was briefly explained – whetting our appetites for more as Kate discretely re-arranged the copies of the book she had brought with her in the background! Ian noted that the requirement elements are common sense, but are often done badly. Priorities are done extraordinarily badly. Simple, but not easy.

I was particularly pleased to be told that 'what you can have' is included. All too often purists separate requirements discovery from design until too late. Ian and Ljerka told us that the requirements must reflect reality and what is possible, from both technology and resource perspectives. We do need to go between requirements and design as early as possible. Requirements discovery is iterative - validating requirements with the stakeholders and with the design is evolutionary.

Richard Veryard questions Ian

Richard Veryard asked Ian about the intended audience of the book – is it aimed at students, requirements managers, systems engineers or G20 leaders? Ian responded that he will be sending a copy of the book to Edward Lee, the chair of the public accounts committee, who often write good reports that are then ignored! The message was clear – the book should have a wide appeal, providing useful guidance to all who need to interact with stakeholders to discover requirements.

It was perhaps appropriate for this event that the book was 'formally' launched by Ian and Ljerka cutting into an excellent cake version of the publication. It was the best book I have tasted for some time!

Ljerka with the Discovering Requirements 'Birthday Cake'

© Simon Hutton, 2009

Mind the Gap – International Workshop

A report on the International Workshop on Requirements Analysis (IWRA) held on 6th and 7th of December 2008 at King's College London.

This was a two day invitation-based event at the CREST centre, King's College London and was supported by ATRIS S.A., the BCS, International Life, Info Success, Piraeus Bank,Tx Plus and the CREST. The CREST group at King's, headed by Prof. Mark Harman, specializes in the testing and evolution of software, with an aim of making these activities better, faster and cheaper. The IWRA event was organized by Manos Nistazakis (Middlesex University) and George Tsaramirsis and Iman Poernomo (the Predictable Assembly Laboratory, King's College London).

The event brought together academics and industry to discuss the problem of the semantic gap that still holds between requirements analysis and system design. The event attracted a large participation from different specialist groups within industry and academia, which facilitated a very lively discussion.

The event's first keynote speaker was Ronald Stamper, one of the founders of the semiotics based approach to requirements engineering, and the originator of the MEASUR methodology. Prof. Stamper provided an enlightening overview of how semiotics solves many of the problems of obtaining requirements, drawing on the philosophical pragmatism of Pierce, the semiotics of Saussure and Austin's speech act theory. Employing concepts from philosophical epistemology, Stamper explained how his techniques can be used to precisely identify the actors, actions, roles and rules of a business process as *shared* ontologies and norms. The second keynote was from Kecheng Liu, director of the Informatics Research Centre at the University of Reading, who gave an encouraging overview of a number of industrial success stories involving semiotics for requirements analysis, and outlined the advances in the approach his group at Reading have put forward.

Several related papers were presented on semiotics for requirements: Yasser Ades (Greenwich) spoke on how semiotics can assist the maintenance of data models in the light of business change; Sani and Kebabchi outlined a practical approach to representing time within requirements documents; Umarov spoke about the relationship between semiotic business rules requirements and formal workflow designs.

An interesting discussion ensued regarding the relationship between essentially human-oriented requirements documents such as those provided by MEASUR and more formal design documents, such as the BPMN or UML statecharts, which come equipped with some form of evaluative or operational semantics. Clearly the former eventually need to be mapped to the latter. Poernomo, Tsaramirsis and Umarov, through their research at King's argued that this gap might be bridged through the use of Model Driven Architecture approaches, where automated transformations across different modelling languages can be utilized. Such transformations demand a form of semantic preservation and this in turn begs the question of how the semantics of a requirements notation can be precisely tackled. A rich notation like MEASUR should be more readily amenable to this. On the other hand, it was also argued that requirements documents, as human-oriented specifications, are by definition informal, and any formalization will defeat their purpose.

An alternative approach to bridging the gap is to develop an end-to-end approach, where requirements and design notations are blurred. This was advocated in process-oriented framework developed at Cambridge and MIT, presented by Behzad Bastani. His talk presented a method specifically designed to support the requirements analysis and design of open systems, and therefore addressed an interesting side issue of mutually exclusive alternation between requirements over time.

Anthony Finkelstein (UCL) gave the last keynote, raising the important point that *governance* is perhaps a more serious unaddressed concern for the requirements analyst.

His theme was in concord with several other presentations, significantly, from industry participants. The problem of adequately addressing human aspects of requirements were discussed in separate talks by Tim Linsdell (TXPlus) and Nicolas Zeni (University of Trento). Requirements management were discussed by Martin Wheatman (Reading), Lindsey Brodie (Middlesex), Tim French (Bedfordshire) and Krikor Marokean (Printee).

How these concerns might relate to the more model-centric work of the semiotics and MDA advocates remains an open question: but by bringing experts from all these areas together, we made some progress toward at least thinking about this possibility.

The proceedings of IWRA 2008 are published by Pearson. They are available from Amazon.co.uk, with an ISBN: 978-1-84776-663-2. Any enquiries relating to this may be addressed to Manos Nistazakis (Middlesex University, e.nistazakis@mdx.ac.uk) and George Tsaramirsis (gtsaramirsis@googlemail.com) and Iman Poernomo (King's College London, iman.poernomo@kcl.ac.uk).

Are Your Lights On? – A Presentation

A presentation given by Don Gause to the New York City Software Improvement Network (NYC SPIN) on April 1st, 2009.

In front of an audience of about fifty software industry professionals, Don Gause combined his unique insight into requirements engineering practice with his passion for story telling and good humour to explore the question – if 'the customer is always right' (and we all know that familiar adage), just who **IS** the customer? Or better, just who **ARE** the customers? For those of you who have not had the great pleasure of attending one of Don's presentations, be warned – it is a rare person who goes unchallenged in the audience and without a role to play. Right from the get-go, Don assigned a member of the audience to keep a record of all his 'profound statements' throughout the evening and to raise his hand every time he thought he recognised one. Although headlined as a highly interactive presentation with a tinge of humour, this was an understatement!

"If I say one thing and I really meant another thing, take the one that's right." – Don Gause.

After threatening to keep the audience until 10:30pm (and trust me, Don would have done so and the audience would have actually stayed if I didn't play timekeeper), Don proceeded to take us on a requirements-related journey that spanned the Stefan Wolpe's Symphonies, Jay Leno, the anticipated life of an eleven-story apartment building, public housing projects, furniture design, doctors on TV, arrival times at oncology clinics… and he somehow even managed to slip 'Kolmogorov Smirnov' into a requirements talk and make it funny. I was too busy laughing most of the time to keep detailed notes I'm afraid, so my record of the presentation comprises the 'profound statements' of Don Gause (with my thanks to Ralph Rizzuto):

- The real requirements never change.
- If the real requirements never change, can they ever be known?
- Understanding what you need to know is a design activity.
- But understand why you need to know, so as to write the five-minute monologue.
- Satisfying the customer is necessary, but not sufficient.
- The client is that party that controls the resources – the person who pays, who hires and fires (a better term than customer).
- The user is an individual that is affected by the system (and this can include the client).
- Users are actually roles that are appropriate at the time.
- Always consider how long does it (the system) have to last? This impacts scope.
- Always experience the system in the way your users would experience it.
- Certain requirements we will never be able to identify until the system is in the marketplace.
- We may have perfect information, but we still have bias.
- Lost opportunity is what the user knows that the designer does not know.
- There is information we cannot obtain – this is information that the designer does not know and the user does not know.
- There are unintended consequences, things we cannot possibly know irrespective of the effort we put in.
- There are surprises … and there is inevitably nature's last laugh.
- If I say one thing and I really meant another thing, take the one that's right.

Don Gause is a Research Professor in the Thomas J. Watson School of Engineering, Binghamton University of SUNY and the Principal of Savile Row LLC. Together with Gerry Weinberg, Don has co-authored two of my all time favourite books in the area of requirements engineering: "Exploring Requirements: Quality Before Design" and "Are Your Lights On? How to Figure Out What the Problem REALLY Is". You can find Don's NYC SPIN presentation slides at: *http://www.nycspin.org/*. You can contact Don at: *dgause@stny.rr.com.*

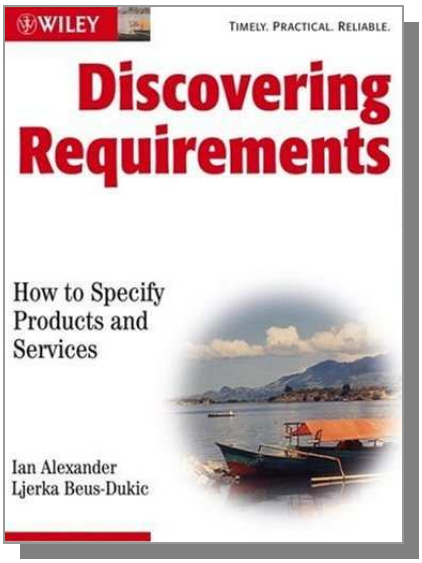
©Olly Gotel, Pace University, (ogotel@pace.edu)

Re-Readings

Discovering Requirements

Ian Alexander and Ljerka Beus-Dukic

John Wiley & Sons, ISBN 978-0-470-71240-5



An extremely readable and useful book brought to you by two RESG committee members. This book is well organised, practical and insightful. The authors propose a matrix of "Elements" and "Contexts" for requirements discovery. Following this model, the book is divided into two parts; Part 1 - Discovering Requirement Elements (with chapters on Stakeholders; Goals; Context, Interfaces, Scope; Scenarios; Qualities and Constraints; Rationale and Assumptions; Definitions; Measurements, and; Priorities), and Part 2 - Discovery Contexts (with chapters on Requirements from Individuals; Requirements from Groups; Requirements from Things; Trade-offs, and; Putting it all Together). In the Introduction, the authors state that requirements specification can be considered to be a network of related elements, "…and indeed, the chapter structure of this book can be seen as a customisable template for organising the requirements on your project."

If pithy quotes are a good measure of the value of a book, then this is a great book. I stopped scribbling down quotes by page 10 when I had amassed the following:

- "Requirements are discovered by the use of appropriate inquiry techniques. They are not sitting about, waiting to be 'captured'."
- "Discovery, however surprising and delightful the actual moment of realisation, comes as a result of a deliberate search."
- "Projects need to pay attention to discovering their requirements, using a battery of complementary techniques…"
- "Perhaps the projects in greatest danger from poor requirements work are those that seem fairly small and simple, but turn out to contain hidden complexities."

In addition to the quotable quotes, the book is also crammed with well-crafted expressions and valid observations. A few that I liked are:

- "goals and stakeholders work together"
- "ill-defined boundaries"
- "interfaces aren't just hardware"
- "requirements archaeology" (gathering requirements from documents)
- "Requirements Chef" (a lovely concept, included in Chapter 15, "Putting it all Together")
- "no two projects are alike"

This book is unusually easy to read and extremely well structured. Each chapter begins with a couple of sentences defining the questions that the chapter will answer. Next is a paragraph summarising the chapter. The reader can therefore establish, in less than a minute, whether a chapter is likely to help them with their immediate problem. Following the main body of each chapter (all of which are also very well structured), there is a "Bare Minimum of…" which, as the heading suggests, defines the least you should do for this element or context. Each chapter also includes a small but valuable set of Exercises and suggested Further Reading.

Many people involved in system development (still) talk about "users" as a homogeneous group. Increasingly, more enlightened people talk about "stakeholders" – but I am not convinced that stakeholder analysis is actually taken seriously in many developments. For this reason I'd argue that chapter 2 of this book ("Stakeholders") is essential reading for every systems engineer/business analyst/project manager. There is a strong emphasis in this chapter (and throughout the book) on the importance of stakeholders and consideration of stakeholder roles. As ever, Ian is quick to remind us that we should always consider negative stakeholders.

The book comfortably straddles what might loosely be called "theory" and "practice". The breadth and depth of the authors' experiences are woven throughout, with

no awkward distinction between concept and application. The final chapter "Putting it all Together" does exactly what it claims. Included in this chapter is an essential table of "possible discovery context/requirement element approaches" and a set of case studies that illustrate how an element/context matrix may be populated for a specific project.

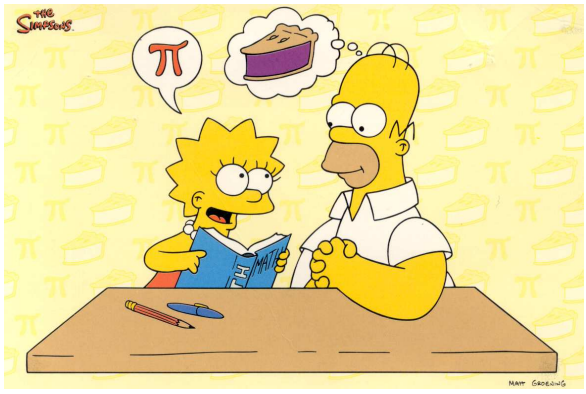
Having read the book and made some notes for this review, I tried an experiment. I randomly opened the book in half a dozen places. On each page I quickly found something interesting and useful. I tried the same experiment with a range of other requirements books, and not one of them was nearly as satisfying. I don't suppose this is how Ian and Ljerka intended the book to be used, but it does give a good indication of its quality and value. Ian recently told me that he and Ljerka had tried (and failed) to write a short book. It may have become a bigger book than they had planned, but I don't think that there is any filler in these 450 pages. I wholeheartedly recommend this book to all readers of RQ.

RE-Verberations

Say What You Mean!

Simon Hutton, Headmark Analysis

A characteristic of a good requirement is that is should be unambiguous – all stakeholders should understand what the requirement means without having to interpret the requirement for themselves. Interpretation often leads to subjectivity and mis-understanding – the very problem we try to avoid with our well written requirements.



Ambiguity leads to false expectations!

Ambiguity can be addressed by writing the requirements in clear, concise language, with all technical terms explicitly defined and the words properly spelt.

Aoccdrnig to rscheearch at an Elingsh uinervtisy, it deosn't mttaer in waht oredr the ltteers in a wrod are, olny taht the frist and lsat ltteres are at the rghit pcleas. The rset can be a toatl mses and you can sitll raed it wouthit a porbelm. Tihs is bcuseae we do not raed ervey lteter by ilstef, but the wrod as a wlohe.

If you read the last paragraph again, you will notice that spelling is not as important as we may think. The researchers at Cambridge University discovered that spelling isn't too important as the human brain can interpret the letters to construct the intended work. The above example is quite extreme, but does explain why we always miss at least one spelling mistake in every document we write. So long as the first and last letters are correct we can often read common words without too much difficulty.

Dr Ken Smith, a lecturer at Buckinghamshire New University, recently told the BBC that common mistakes should be accepted as reasonable variations to the English language. He does lose some credibility by admitting that his motivation is to reduce the time he spends marking papers and essays from his students, but does emphasise he wants to see flexibility in spelling rules rather than allowing text slang to proliferate.

Of course this is a bold move in academic circles, where bad spelling is attributed to a poor education or a lack of attention to detail. Marie Clair of the Plain English Campaign responded to Dr. Smith's proposal with some well chosen (and correctly spelt) words such as 'anarchy' and 'chaos'.

Perhaps we should continue to check that we are spelling our requirements correctly, at least until someone develops a requirements definition language based on texting. We could encourage our stakeholder community to twitter, gradually refining the requirements through social interaction. *It cud B d fucha of needs ngineerin!* (for the confused there is a useful text translator at www.lingo2word.com).

Of course, spelling is one aspect of reducing ambiguity. We also need to ensure the correctly spelt words are understandable. We need to be sure the words we use are appropriate and clearly defined, especially where technical terms are being used. The UK Local Government Association recently issued a banned list of 200 worst uses of jargon to councils. LGA chairman Margaret Eaton said "The public sector must not hide behind impenetrable jargon and phrases". The list includes terms such as 'predicators of beaconicity' and 'coterminous stakeholder engagement'. The LGA list does include alternatives, such as 'measuring' instead of 'benchmarking', or 'idea' instead of 'seedbed'.

It does make sense, and the full list of banned words could make a useful starting point to reviewing your requirements for ambiguity. The list is at http://www.lga.gov.uk/lga/core/page.do?pageId=1716341.

Readability: Realistic Books

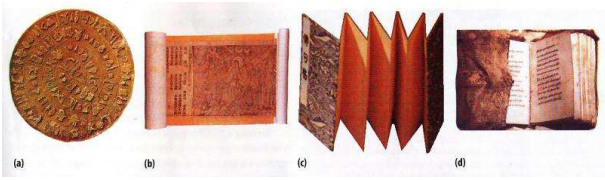
Ian Alexander, Scenario Plus

One of the wonderful insights given to me by a famous consultant (I will spare the name to avoid blushes) is that it is well worth taking the IEEE's main journals, even though most of the articles are ho-hum or not relevant, because every now and again there is something quite wonderful. It is easily worth the subscription for the surprise and delight of coming across a really well-written (and ruthlessly peer-reviewed and edited) article on something new, fascinating, even revolutionary.

Well, in February's Computer [1] there is one such. I fancy it was the most enjoyable article I've seen in that magazine since I started reading it. It was technical, but plainly and simply written. It was exciting, but free of hype. It was novel, but deeply rooted in history. It was about some –ilities (non-functional requirements), yet well-organised. And it was the work of a PhD thesis, yet obviously and directly useful. Pinch yourself, you are reading this.

The article, by Veronica Liesaputra (the PhD) and colleagues in the New Zealand Digital Library (www.nzdl.org) is about Realistic Books. Sounds hubristic, huh? Well, it does what it says.

The article goes straight for the jugular by showing four forms of text display developed over the ages, from Sumerian clay tablets to mediaeval scrolls, concertinas, and codices (that's books to you and me).



(a) Clay Tablet, (b) Scroll, (c) Concertina, (d) Codex

Very nice, but what's that got to do with computing or requirements, you are wondering. Well, these historic media were in their ways revolutionary, making it possible to capture and reuse knowledge, and in stages becoming more portable, easier to access, quicker to use, more robust, and not least more pleasant to read.

Portability, indexability, performance, robustness, usability… now we're talking requirement-speak.

But that's still all ancient history, isn't it? Yes, but as Santayana said in his *Life of Reason*, those who cannot remember the past are condemned to repeat it. As a student, I directly addressed a screen display (pixel by pixel) to draw graphics on a fixed area (I think it was a measly 640 x 480 pixels, but I digress). I was writing on a Clay Tablet.

Text editing became practical on computers with Word Processor programs like Wordstar (on which I typed up my student thesis), and its successors including Word; and HTML for that matter. The text could be divided into paragraphs but basically it was one long stream that you could Scroll up and down. We had reached 150 BC and the Dead Sea Scrolls. All that was missing was the parchment.

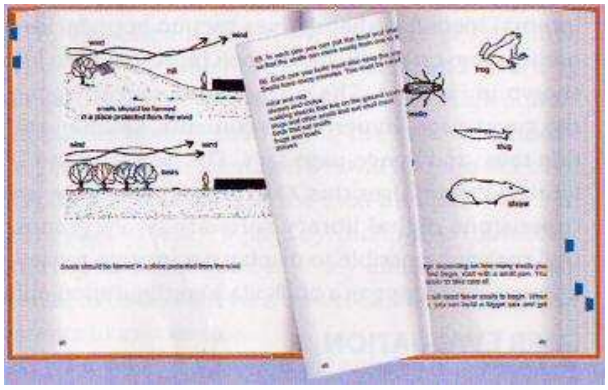
Word represents a considerable advance on the rolled-up scroll; you still scroll up and down, it is true, but page breaks are shown – as dotted lines or as actual page images with black text on a white background, and you can page up or down: it is a Concertina. We had reached 1100 AD! Ya-hey!

Word and other formats, including Adobe PDF, handle a concertina-string of pages well. There are small ancillary indicators to show you where you are: there's a slider, and when you hold it down it says which page (and chapter) you are on; there's also "Page 2 of 27" in the status bar – but most people never notice these subtle cues, and feel quite lost in documents of any length. Various organizations have been trying to get us to read e-books on screen, using flat displays of one or two pages. There is some take-up – we really want the convenience – but plenty of resistance: Scrolls and Concertinas are simply pretty horrible to read for any length of time, as you don't know where you are, and you crave proper context.

The solution to this was invented in, yes, the middle of the 4th century AD in the shape of the Codex. The Codex Sinaiticus (the British Library has generously made it available at

http://www.bl.uk/onlinegallery/sacredtexts/codexsinai.html) is both beautiful and easy to navigate. It's a proper book, and you can turn the pages of it on the British Library website.

But it took a lot of effort to set up that website. What Liesaputra and NZDL wondered was, could a realistic book with turning pages, table of contents, bookmarks and the ability to see where you are and to make notes where you like be created with modest resources and read on an ordinary PC?



A Realistic Book by Veronica Liesaputra

She tried out a range of clever ways of flipping pages, including computationally-intensive particle-and-spring models of paper (with equal thickness all over), but hit upon a simple 2-Dimensional trick: peel the page rather than actually turning it. This only works if you look at the book from directly above, but that's basically what you want to do when reading. The idea is to dog-ear a page, lifting it from any point on one of its free edges and folding it right over into a triangle

(or quadrilateral), then successively advancing and folding the page flat until you have the page fully on the other side:

> "This would be difficult to accomplish physically, and it would make a creased mess of the page, but it is trivial in a computer model."

She came up with a slight refinement: use a spline (a mathematical curve) to soften the edges. The result is pretty realistic, and it uses minimal computation, so it is fast. It's even bearable via the Web all the way from New Zealand, which is a pretty stern test. But the good news is, the tools to create your own are free. You can convert PDF and HTML to what is in fact a simple XHTML structure.

Reader, it works, it really does. How does it feel? The feeling is like that of getting out of the sack in the school sack race, untying one's legs, and stretching out with real, normal movement again. Mediaeval scrolls and concertinas? Bah! Consign them to the museum!

[1] Veronica Liesaputra, Ian H Witten, and David Bainbridge, *Creating and Reading Realistic Electronic Books*, Computer, vol 42, no 2, pp 72-81, February 2009.

RE-Partee

Guess the Requirements Chick or Chico!

Ian Alexander, Scenario Plus

As Requirements become more mainstream, a surprising addition to RE websites is … attractive young women. Or their avatars, anyway. And sometimes blokes. Some are animated (and even talk, if you have speakers on your PC); other just sit and smile.

But how do organisations choose their Req Chicks? Or Chicos? Is there something about each of them that subtly matches the character and style of their organisations? Or is there a universal type of practical intellectual pulchritude that suits all organisations equally?

Your task is to match the Req Chicks or Chicos to the organizations. No cheating now – the answers are at the bottom of Page 24!

The organisations are:

A. Accompa

B. ActiveFocus

C. Case Spec

D. Feature Plan

E. Siemens

And here are the Req Chicks/Chicos:

1. 

2. 

3. 

4. 

5. 

RE-Sources

Books, Papers

Al Davis' bibliography of requirements papers:
http://www.uccs.edu/~adavis/reqbib.htm

Ian Alexander's archive of requirements book reviews:
http://easyweb.easynet.co.uk/~iany/reviews/reviews.htm

Scenario Plus – free tools and templates:
http://www.scenarioplus.org.uk

CREWS web site:
http://sunsite.informatik.rwth-aachen.de/CREWS/

Requirements Engineering, Student Newsletter:
www.cc.gatech.edu/computing/SW_Eng/resnews.html

IFIP Working Group 2.9 (Software RE):
http://www.cis.gsu.edu/~wrobinso/ifip2_9/

Requirements Engineering Journal (REJ):
http://rej.co.umist.ac.uk/

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

Volere template:
http://www.volere.co.uk

DACS Gold Practices:
http://www.goldpractices.com/practices/mr/index.php

Media Electronica

RESG Mailing List
http://www.resg.org.uk/mailing_list.html

RE-online
http://discuss.it.uts.edu.au/mailman/listinfo/re-online

Requirements Networking Group
www.requirementsnetwork.com

RE Yahoo Group
http://groups.yahoo.com/group/Requirements-Engineering/

RE-Actors

The Committee of the RESG

Patron:
Prof. Michael Jackson,
Independent Consultant,
jacksonma @ acm.org

Chair:
Ian Alexander,
Scenario Plus,
iany @ scenarioplus.org .uk

Immediate Past Chair:
Dr. Peter Sawyer,
Lancaster University,
sawyer @ comp.lancs.ac.uk

Treasurer:
Dr. Steve Armstrong,
The Open University,
S.Armstrong @ open.ac.uk

Secretary:
Dr. James Lockerbie
City University,
ac769 @ soi.city.ac.uk

Membership secretary:
Dr. Yijun Yu
The Open University
Y.Yu @ open.ac.uk

Publicity officer:
Dr. William Heaven
Imperial College,
wjh00 @ doc.ic.ac.uk

Newsletter editor:
Simon Hutton
Headmark Analysis Limited
simon.hutton @ headmark-analysis.co.uk

Newsletter reporter:
Dr. Ljerka Beus-Dukic
University of Westminster,
L.Beus-Dukic @ westminster.ac.uk

Student liaison:
Dalal Alrajeh
Imperial College
dalal.alrajeh @ imperial.ac.uk

Member without portfolio:
Dr. Emanuel Letier
University College London,
e.letier @ cs.ucl.ac.uk

Member without portfolio:
Prof. Bashar Nuseibeh
The Open University
B.Nuseibeh @ open.ac.uk

Member without portfolio:
Prof. Neil Maiden
Centre for HCI Design, City University,
N.A.M.Maiden @ city.ac.uk

Member without portfolio:
Alistair Mavin
Aero Engine Controls,
alistair.mavin.jvaec @ rolls-royce.com

Member without portfolio:
Suzanne Robertson,
Atlantic Systems Guild Ltd,
suzanne @ systemsguild.com

Contributing to RQ

To contribute to RQ please send contributions to the Editor, Simon Hutton, at simon.hutton@headmark-analysis.co.uk. Submissions must be in electronic form, preferably as plain ASCII text or rtf.

The deadline for RQ 52 (July 2009) is Friday 10th July 2009.

Joining the BCS RESG

Visit **http://www.resg.org.uk/** for membership details, or email **membership-RESG @ open.ac.uk**

Answers to Chicks & Chiccos: 1.ActiveFocus 2.-Feature Plan 3.-Case Spec 4.-Accompa 5.-Siemens