bcs
The Chartered Institute for IT

Requirements Engineering Specialist Group

Requirements Quarterly

Contents

RE-soundings

From the Editor

A belated New Year's issue has arrived! I apologise for the delay, but hope that this bumper issue more than makes up for it. As well as the usual previews and reviews of RESG events, we have Olly Gotel's high-energy guide to what whitewater rafting can teach us about requirements engineering; Camilo Fitzgerald's thought-provoking case for why RE could be applied to microfinance; and David Gelperin's whistle-stop tour of what a requirements management platform should – and should not – be doing for you.

On top of all that, I would like to thank Ralph Young especially for taking up my challenge and agreeing to revisit the vision for requirements engineering he set out in these pages over 10 years ago. In doing so he has consulted widely among his many expert peers and drawn some extremely interesting conclusions. Was his vision fulfilled? What does he now recommend? Do you agree?

Read on to find out!

William Heaven, RQ Editor

Chairman's Message

This is our first newsletter since the last AGM back in September. A discreet but significant event at the AGM was Ian Alexander's decision to step down as chairman of the group. Ian has played a major role in the group, first as a prolific writer for this newsletter,

then as its editor, then as chairman for the last 3 years. His experience and good spirit will be really missed at the helm. We wish him all the best in his new adventures and hope he'll continue to stimulate the group with his sharp comments and colourful writing. I wanted to take this opportunity to thank him once again for the amazing work he's done for the group.

We have had several events during the last quarter. In November, we held our annual "Careers in RE" event that attracted around 50 undergraduate and post-graduate students eager to hear about the day-to-day lives of those who analyse requirements for a living. In the same month, we've had the first of our new series of "RESG evening" events at the BCS headquarters near Covent Garden. The evening theme was requirements traceability; you can read more about it further down in the newsletter.

Following popular demand, our next RESG evening, on the 22th of March, will be on requirements in agile projects. This follows several successful events organized on the same topic in Telford and Lancaster last year. In March, we will also have the second edition of our workshop for early career researchers aimed at sharing information about career strategies and prospects after a PhD in requirements engineering.

More details about these events can be found in this newsletter and on our website (www.resg.org.uk)

We will have a further RESG evening in London on 14 June 2012. The theme is still to be confirmed. There has been significant interest recently, both in research and industrial practice, about the use of social networks and web-based systems to facilitate requirements gathering, evaluation, and prioritization. This June evening might be a good opportunity to evaluate the situation on this emerging topic. And why not experimenting with such as system for ourselves? Can it help us to collectively define our goals for the RESG and identify the best ways of achieving them?

In the meantime, you can still share with us your goals and implementation ideas (such as activities you'd like us to organise) in the traditional way by contacting any of us on the committee by email (our details are on the last page) or over a drink at one of our events.

I hope you'll enjoy this newsletter!

Emmanuel Letier, University College London
RESG Chair

RE-treats

Agile Development – A curse or a blessing for requirements engineers?

6pm-8:30pm, Thursday 22 March 2012

Venue: BCS Offices, 5 Southampton Street, London WC2E 7HA

Many organisations have turned to, or are considering turning to agile methods of software development. These methods have been described as the revenge of the programmers against heavyweight documentation-centric processes that many associate with traditional requirements engineering. Agile methods aim at reducing the risks of requirements problems that have plagued many projects by developing software in small iterations with intensive customer involvement throughout development. Requirements knowledge is transmitted orally with minimal documentation in the form of user stories and acceptance tests. But what is the reality of agile practices at addressing requirements risks? Can we believe the hype surrounding agile methods? What are the effects of these practices on the day-to-day work of requirements engineers and business analysts?

This RESG event will take a look at agile development from a requirements engineering perspective and will leave plenty of time for discussion to identify common concerns and practical guidelines for requirements engineers.

Post-Docs / Early Researchers Workshop

March

Venue: 180 Queen's Gate, Imperial College London

This year, the RESG is planning a half day workshop with talks from leading experts in Requirements Engineering and a panel discussion where researchers in their early career will have the opportunity to discuss and engage in exciting and innovative trends in RE research. The aim of this event is to give post docs a wider perspective on the problems and solutions being considered in the RE community today.

The event will be hosted at Imperial College London. Further details will be posted on the RESG website. To register your interest in attending, please contact dalal.alrajeh04@imperial.ac.uk.

In the meantime, have a look at the format of last year's successful event:

www.resg.org.uk/index.php/Post_Docs_2011

Social Requirements (TBC)

Thursday 14 June

Venue: BCS Offices, 5 Southampton Street, London WC2E 7HA

The use of social networking to elicit requirements.

Details to be confirmed.

2012 AGM & Annual Party

September

Venue: 180 Queen's Gate, Imperial College London

Details to be confirmed.

Non-RESG Event

Business Rules and Decision Analysis Masterclass

26-27 March – London

Ron Ross

This hands-on workshop gives you the essential tools you need to achieve order-of-magnitude improvements in your company's capacity to manage decisions. The result is simpler, smarter process models and a huge boost in business agility.

Delegates will learn:

* Conduct smarter, more effective business analysis
* Identify and analyse decisions in business processes
* Capture business rules
* Write clear, business-friendly rule statements
* Create robust decision tables
* Identify anomalies in decision logic and correct them early
* Perform concept analysis and develop a structured business vocabulary
* Develop a pragmatic rule management approach

RESG members get a 10% discount.

Visit www.irmuk.co.uk/events/75.cfm for full details or contact IRM UK on +44 (0)20 8866 8366 or email customerservice@irmuk.co.uk.

Non-RESG Event

Mastering Business Analysis

19-20 April / 8-9 November 2012 – London

James Archer

Business analysis provides the foundation for almost every kind of business change. The craft of business analysis is to investigate the business, to find its problem hot spots and recommend ways to improve them. This two-day seminar and workshop in business analysis gives you the skills and tools to discover your client's real business, and to determine and demonstrate the best ways of improving it.

Delegates will learn to:

* Discover real business needs, not just the most talked-about ones.
* Improve the business processes by applying automation or other means.
* Define the most beneficial scope for the analysis project.
* Use models to understand and communicate the business processes, and ensure stakeholders also understand.
* Understand how to employ business events as a way of partitioning the business for easier understanding.
* Be better at interpersonal communication.
* Think systemically, and find truly the best way to improve your client's business.
* Be a better business analyst

RESG members get a 10% discount.

Visit www.irmuk.co.uk/events/90.cfm for full details or contact IRM UK on +44 (0)20 8866 8366 or email customerservice@irmuk.co.uk.

Non-RESG Event

Mastering the Requirements Process

16-18 October 2012 – London

Suzanne Robertson, Atlantic Systems Guild

Requirements are the most misunderstood part of systems development, and yet the most crucial. Requirements must be correct if the rest of the development effort is to succeed. This workshop presents a complete process for eliciting the real requirements, testing them for correctness, and recording them clearly, comprehensibly and unambiguously.

Delegates will learn to:

* Determine your client's needs-exactly
* Discover the real business, and how to improve it
* Write requirements that are complete, traceable, and testable
* Precisely define the scope of the project
* Discover all the stakeholders-and keep them involved
* Get the right requirements quickly, and incrementally
* Learn state of the art requirements techniques

RESG members get a 10% discount.

Visit www.irmuk.co.uk/events/1.cfm for full details or contact IRM UK on +44 (0)20 8866 8366 or email customerservice@irmuk.co.uk.

More?

If you have an idea for an event – especially one based outside London – please get in contact with a member of the committee (see the back page).

RE-member

Requirements Engineering for Systems, Services, and Systems of Systems (RES^4)

30 August 2011 – RE Workshop, Trento, Italy

The RES^4 workshop at RE 2011 was headed by **Luciano Baresi** (Politecnico di Milano), **Jane Cleland-Huang** (DePaul University) and **Neil Maiden** (City University London). It brought together two areas of interest under one umbrella – services and systems of systems. The original programme included 14 presentations including an invited talk from Peter Bahrs of IBM. Alas, the keynote speaker was unable to attend due to Hurricane Irene grounding his flight from the United States!

Services

Work presented in the area of services included service monitoring, adaptation and evolution of services, and new methodologies proposed for the development of service compositions. **Liliana Pasquale** described a monitoring technique to assess temporal requirements that are characterized by uncertainty in the temporal domain. Also in the area of service monitoring, **Lin Liu** described different ways to evolve service specifications and design to satisfy emerging needs, and classify the triggers for evolution. **Axel Hoffmann** looked at exploiting emergent properties of service-based systems for the development of portals, and **Sebastian Adam** discussed the challenges of managing requirements in an emergent environment in which it may not be possible, or feasible, to understand and discover all requirements in advance. **James Lockerbie** presented a model to facilitate the description and the measurement of the qualities of cloud-based IT services, and also returned later to present research on addressing the development of choreographies from their design to their enactment. **Munindar Singh** proposed an approach and a tool for the design of cross-organizational processes, where requirements of multiple stakeholders are harmonized by selecting, refining, and composing protocols.

Systems of systems

Contributions in the area of systems of systems addressed the problems and challenges of managing all aspects of the requirements process within the setting of large and often heterogeneous environments. Topics included identifying specific challenges for discovering, analysing, specifying and managing requirements for systems of systems, and working with stakeholders. **Peter Bahrs**' work provided an industrial perspective on the impact of smarter architectures and engineering environments upon the requirements process. **Fan Yang-Turner** gave two presentations on requirements discovery. The first looked at techniques for creative requirements elicitation and the second presented extended features of use case diagrams. The work addressed the challenges of working with stakeholders in an EU collaborative project. **Leyna Cotran** presented a process for using tags and metadata to identify and organize requirements in an established domain, and **Anitha PC** described a process model for use in the development of systems and systems. Work from **Marina Berkovich** focused on analysing the degree to which the analysis techniques of software engineering are suitable for product service systems. **Cornelius Ncube** presented a summary of the key challenges facing the requirements engineering community in the area of systems of systems.

The contributions showed that requirements engineering for systems, services, and systems of systems represents an area of research in which advances in technology and process are met with new challenges. RES^4 provided participants with a good opportunity to explore and discuss these challenges and to present and identify areas of future research for the community.

James Lockerbie, City University

Report adapted from: Baresi L., Cleland-Huang J., Lockerbie J., Maiden N.A.M, Pasquale L., "An introduction to the workshop on requirements for systems, services, and systems of systems", RES^4, Trento, Italy, August 30, 2011.

2011 AGM

7 September 2011 – Imperial College London

As usual, last year's AGM served as an excellent excuse for a summer's evening meet-up with friends, colleagues and like-minded new faces. A selection of ales and fine wines were laid on to wash down the finger-food and, as the hubbub of greetings and introductions quieted, **Ian Alexander** began his last round-up of the year – his final official duty as chair of the RESG.

The AGM also featured a brief but fascinating research talk by **Sanaz Yeganefard** from the University of Southampton, in which she gave a tour of her work on structuring requirements for control systems to aid formal refinement. Among the ideas she introduced was that of "commanded" phenomena in addition to the usual pairing of "monitored" and "controlled", so that the role of an operator can be more explicitly captured. More details can be found in her AVOCS 2011 paper, "Structuring Functional Requirements of Control Systems to Facilitate Refinement-based Formalisation", co-authored with Michael Butler.

The main event, however, was **Neil Maiden**'s keynote, "Requirements Research and Practice: Stories Past and Future ..." Neil had promised in his abstract that "In this seminar the speaker will deliberately and outrageously overuse the keyword in the seminar title to describe previous uses of scenarios and stories to discover future system requirements, look back and tell stories about the early days of the RESG, and encourage collaborative improv storytelling as the future of requirements engineering". And we were not disappointed. We were regaled with a host of anecdotes spanning Neil's career in RE and the history of the RESG (being co-founder with Bashar Nuseibeh, he has, of course, been around since the beginning) all tied in with his research – and fieldwork – investigating the use of stories as practical and effective tools for requirements capture.

Stories come in many forms in RE – use cases, scenarios, storyboards, and, most recently, user stories – and their staying power is likely due to their naturalness. Ask people to describe a process – existing or desired – and the answer is usually most simply expressed as a story of one kind or another. Stories also have beginnings, middles, and ends and so often help catch details that would otherwise be missed out.

Stories also have characters, which can occasionally make it easier to identify peripheral stakeholders in a system. But, perhaps most importantly, stories can be freeform if need be – get the information first and structure it later if the circumstances – such as interviewing busy airport workers, from pilots to stewards – make a more formal method of capture tricky.

William Heaven, RQ Editor

Careers in RE

16 November 2011 – University of Westminster

Over the years, Careers in RE established itself as an event highlighting the difference between requirements engineering education and industry practice. This year, we had three panellists coming from the private and public sector as well as academic research.

Lucy Hunt (Business Analyst/IT Consultant at UNISON) described herself as "endlessly curious". For a number of years she worked as a software engineer (software development and support) and then, after volunteering as an IT adviser for a VSO in Nepal, changed her career path and became a business analyst. The turning point came after a facilitation course where she realised she enjoyed being engaged with people who need a technical solution to their problem from the start.

Shahid Zaman, University of Westminster graduate, now Lead Software Engineer at Docobo Ltd (UK healthcare solutions provider) talked about his experiences in developing a product that enables clinicians to manage the care of patients with long term conditions in their own homes. There is one technique which he singled out as being successful in getting requirements; prototyping works with internal (employees playing the role of patients) and external operational stakeholders (trained patients).

James Lockerbie (Research Assistant at City University) told us about his work on a multi- partner EU-funded research project. His role, mainly related to large scale collaborative i* goal modelling, includes work with different groups of stakeholders (technical and domain partners, knowledge experts) where it is important to identify and document their goals and achieve common understanding and consensus.

Being our primary audience, the event was mainly attended by the final year and postgraduate students (largely from University of Westminster and some from City and Bedfordshire University). Participants showed a high level of interest in the panellists' jobs and how they are related to requirements engineering. The questions raised ranged from specific ones – "How do you conduct interviews?" (Q. for Lucy), "Why you need to use standards?" (Q. for Shahid), and "How you use i*?" (Q. for James) – to more general ones – "How do you cope with unrealistic requirements and lack of requirements?". The panellists were drawn in a lively discussion with more

questions than the time allowed. Perhaps, in this time of economic crisis, the need for requirements engineering knowledge is greater than ever.

© *Ljerka Beus-Dukic, University of Westminster*

Traceability Workshop

24 November 2011 – BCS Offices, London

This was the first of our hoped-to-be regular events held at the BCS offices on Southampton Street. An early-evening seminar with a couple of speakers and plenty of lively discussion – plus refreshments, of course – seemed to be just the thing to attract an animated post-work crowd. The theme this time round was Traceability – an old RE bugbear, perhaps, but one that's still at the root of a great many problems in systems development. Our invited speakers were **Christian Nentwich**, the founder of Model Two Zero – a software company based in London that produces components for high-complexity enterprise IT architectures – and **Stephen Morris**, an ex-town planner for London local government and an Honorary Senior Visiting Research Fellow in the Department of Computing at City University.

Christian, who has experience working with some of the world's largest financial institutions, talked about the problems faced by organisations that need to integrate large and complex systems – off-the-shelf software, internal (legacy) systems, small bespoke applications – with typically tens of thousands of data items. Combining these scales with inconsistent terminology, aliasing needs, and a lack of document-ation can give even the most seasoned systems engineers and analysts a headache. On top of this, organisations need to be able to maintain such systems even when specs are not updated, knowledge leaves as employees move on, and reverse engineering is unfeasible. One solution offered by Christian is the use of a Natural Rule Language that mitigates some of the complexity by formally tracking – at a metadata level – many of the traceability links between the vast number of different elements across an organisation's systems.

Stephen gave us a version of his talk from RE 2011, reporting on a survey that he carried out with Olly Gotel that looked at what could be learnt from traceability techniques in other disciplines. He opened with a question: Is traceability a new problem peculiar to software engineering or an extremely ancient one, with a solution that is essential to many established disciplines? Stephen set out to convince us that the question is largely rhetorical. Drawing compelling analogies from fields as diverse as animal tracking, metrology, epidemiology, art provenance, luggage handling, and food traceability, Stephen pointed out the problems common to each. The key to tackling the Traceability Problem in all these examples is to distinguish between a *track of signs* and a *trace* formed by following that track. A sign is an identifying mark and a track is a pattern – or line – of signs, but to trace is to *identify* a track by *following* its pattern sign by sign.
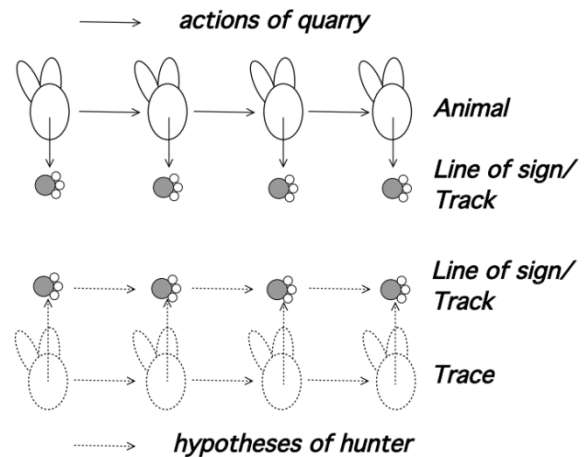
Tracks and traces (Stephen Morris)

Stephen used the breadcrumb trail dropped by Hansel and Gretal to illustrate one of his points, prompting one audience wit to complain that, unfortunately, most developers only dropped pizza boxes. In fact, this comment turned out to be indicative of the frustration that many of the professional members of the audience – requirements engineers and business analysts for the most part – appeared to feel when up against the real-world expectations of clients. Somebody observed that the reason many organisations have the systems they do is because they went to a tradeshow an bought the most expensive one. So what is the use of asking what their original requirements were? Get a client to say what they need, not what they want, was one suggestion. Documentation is not always the answer, responded somebody else – words don't always mean the same thing later. One contractor simply described his experiences as "chaos – you just do enough to stop it blowing up in your face".

Gradually a divide between software deliverers – the camp that most of the evening's participants fell into – and software purchasers – their clients – became apparent. Is there a sense that an event like this is preaching to the converted? How do we then communicate across that divide? A few people called for decent tools, while others confidently asserted the only answer was "really, really good and experienced business analysts". The truth is likely to be that both are essential. Either way, it is clear that Traceability is not a stale topic by a long shot.

William Heaven, RQ Editor

RE-flections

A Whitewater Guide to Requirements Deliverance

Olly Gotel

The Put-In

One of the only times that I get completely lost in the moment is when I am paddling white water. It is both a physically and mentally taxing activity! As the difficulty of the rapids increases, there is simply no room to concentrate on anything else but the water and my moment-to-moment actions to work with it. If you are not a boater, perhaps you sat at the edge of your seat when you first watched Jon Voight and Burt Reynolds canoe down monstrous rapids to the sound of distant banjo music in that classic 1972 movie?

When running white water, it is important to maintain composure in the midst of 'apparent' turbulence, and there are actually many interesting parallels between what makes for a challenging but safe river run and what makes for the successful deliverance of project requirements. Always one to combine work with pleasure, here is my ten-point white water inspired guide to help get you from the requirements put-in to the take-out on your project.

1.   Research Before You Put-In

There is an internationally recognised scale for rating white water difficulty, from class I (easy) to class VI (extreme and exploratory). The class of the river can give you a good indication of whether it is: (a) within your skill level to undertake; and (b) something that you even want to run. The rating system is based upon factors such as the constriction of the river channel, the

Me, in the thick of it in Ottawa River
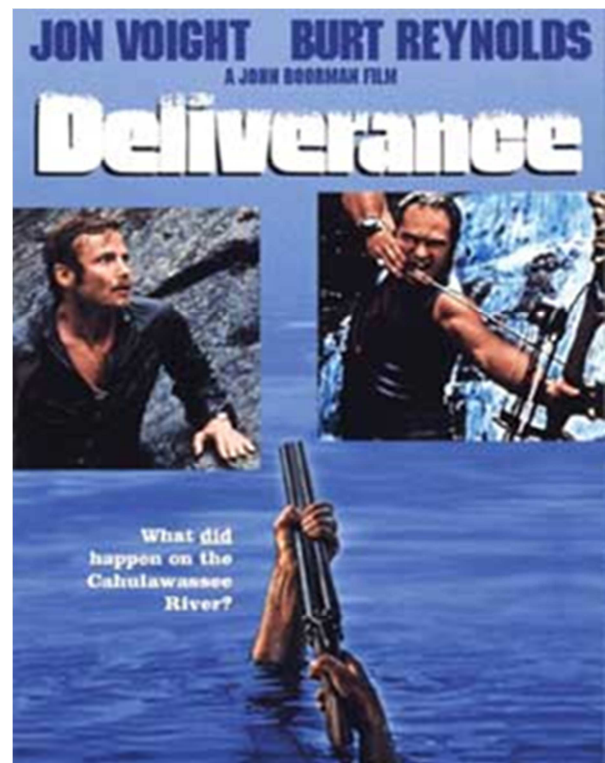
***Deliverance*, the movie**

gradient of the riverbed, the rate at which the water flows, the need to navigate around obstacles, the likely consequences if you mess up, the difficulty of rescue, etc. A class II boater floating carelessly into a class V rapid is certainly in for a surprise!

Before embarking on the requirements gathering for a project, spend some time exploring those factors that suggest the likely complexity of the task ahead, such as the number of distinct stakeholder types, their distribution, the clarity and difficulty of the problem being tackled, the novelty of the projected solution, etc. Have your checklist of criteria and, not only ensure that you gain some idea as to the scope and viability of the upcoming task, also ensure that you have the necessary skills within your team to undertake it. Maybe there would even be value in using a similar rating scale to classify the complexity of requirements engineering projects with some universal consistency?

2.   When in Doubt, Scout

When you are on a raging river and cannot see the full layout of the rapid ahead, it is time to stop and scout. The purpose of scouting is to discern a navigable path down the rapid, reading the water from the bottom of the rapid back up to the top, so as to take a decision as to whether and how to run it. What we are commonly

Scouting the Chattooga River's 'Sock-em-Dog' rapid

looking for when we scout is a way to chain together downstream 'tongues' of green water that indicate smooth deep channels to follow, along with a safe stopping spot to scout once again when part way down the rapid if needed. An obvious case for scouting arises when you see a 'horizon line' ahead, as this usually signals a sudden drop – it could be a two foot ledge or it could be an eighty foot waterfall – so the self-preserving boater gets out of their boat and looks.

There are certainly some equivalent markers on a project that signal that it is time to stop and look more closely at the requirements being engineered. Proceeding with an un-validated requirement is like running a horizon line blind, whereas implementing a requirement without stopping to examine its structure and end point first (i.e., its satisfaction criteria) is like blundering one's way down an unknown rapid and hoping to make it through. While you do not have to see all the way to the end of a project and trace every single requirement's possible path through it ahead of time, you do need to take just enough time to scrutinise and decompose each requirement if it is to be engineered with any sense of control and with the potential for eventual satisfaction.

3. Embrace Moments of Calm

Water flows downstream and an eddy is formed when there is a reversal of this main current. This is due to water passing around an obstacle and creating a displacement behind it into which the water then flows upstream. Eddies are generally good places to stop, catch your breath and scout a rapid from your boat on the river. Building on these rest stops, eddy-hopping is a great way to break a rapid down into manageable chunks. When you eddy-hop, you sit in an eddy, look for the next eddy in a rapid, paddle into it and repeat all the way down the rapid. Some eddies are the size of aircraft carriers and hold multiple boaters quite

comfortably. A micro eddy is the size of a bikini, so things can end badly when more than one boater tries to pile in. Either eddy size, crossing the fierce eddy line that divides the opposing current directions is a common place to flip upside-down, especially if you misjudge or lack control. So, you always combine this localised and detailed view of the 'safe' eddy stops in a rapid with a broader picture of what lies ahead downstream before committing to the run.

Focusing too intently on incrementally decomposing and tackling any one particular requirement can take you to that last-stop pre-occupied micro eddy where you are faced with no other option but to run the scary waterfall ahead, and maybe even upside-down. Therefore, maintain perspective as you engineer requirements and avoid tunnel vision. Zoom into the detail of requirements, but also pan out to the wider context of interrelated requirements and project constraints to gain the whole picture. This situational awareness and context switching is as paramount to a project's survival as it is to a boater's.

4. Use Clear-Cut Signals

River terms are simple and shared, such as upstream and downstream (water flows in the downstream direction, except in eddies), and river right and river left (when looking downstream), for the four basic river directions. Because there is little point relying upon voice communication when stuck in a micro eddy with white water whooshing past you, there is also a small set of signals that boaters use to communicate unambiguously among a group when on a river. These include pointing a paddle in the direction to go in so as to avoid obstacles. Note that boaters never point a paddle at a hazard because that is straight where their buddies will head! Also, a tap on the helmet is the universal signal to both check and confirm that a boater is 'okay'. A whistle is used when boaters loose this visual line of communication.
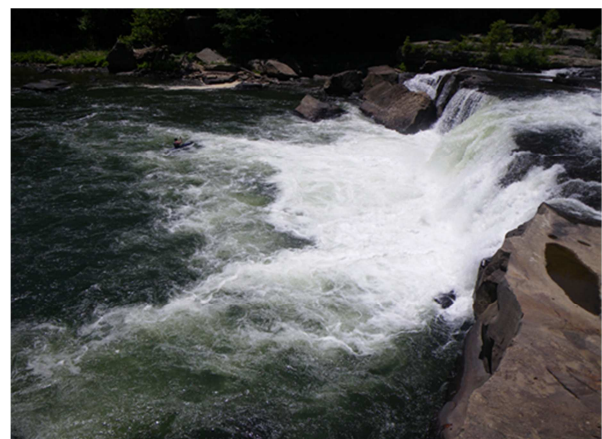
'I didn't see it coming!'
The Ohiopyle Falls on the Youghiogheny River

(Spot the kayaker, near top left, for scale)

A wave formed by water flowing over a broken dam on the Farmington River

Communication is just as fundamental to the smooth running of the requirements engineering process, yet we can overwhelm each other with so much noise in our accompanying documentation that it can be hard to discern the salient requirements and prioritise actions, let alone establish what is and is not timely at any one moment. Agree on the essentials of requirements-related communication and the different ways in which this communication can take place, especially when real-time information is of the essence. Perhaps also check up more regularly with the various project stakeholders along the way: is the customer still happy with the way in which the requirements are being engineered and are the team members in a good place right now?

5. Understand How Features Form and Change

The rock is the foundation for many of the features that we either get to enjoy or to fear on a river, and we learn the moves that we need in our repertoire in order to negotiate them. A 'boof' can get you over a partially submerged rock, 'side-surfing' can help keep you upright in a friendly hole and twirling a paddle on the face of a wave can be the ultimate camera pose. As water levels rise, rocks begin to get covered and form pour-overs, which evolve into hydraulics (holes). These then turn into waves and, eventually, all signs of the rock's presence may disappear. Water levels are always rising and falling and the difficulty of the river is therefore dynamic, so you never paddle exactly the same river twice. Importantly, and often from trial and error, we learn to read the water so that we can distinguish a fun play hole at one water level from a keeper hydraulic[1] that re-circulates water and holds boats (and boaters) at another level.

---

[1] A very dangerous example of a keeper hydraulic is a low-head dam, also referred to as a drowning machine. If you learn one thing from this article, learn this … don't go near low-head dams!

Requirements anchor and shape projects, and the texture they lend to a project is also in transition as the engineering progresses, and as requirements become satisfied by architectural designs and implementation components. Understand the various features that arise and evolve around different types of requirements as the project moves forward in time and as knowledge develops. Understand how any change to these base requirements impacts these features and can change the character of the entire project continuum. Further understand that adding new requirements to an established project can have the same impact as a rock landslide crashing down into a river, dislodging everything you have come to rely upon. You can learn and pre-empt patterns too.

6. Avoid Known Hazards

Water always finds a way to flow, but solid objects are a different matter altogether. Strainers are fallen trees and any boater passing through a strainer is likely to end up like spaghetti. Boulder sieves and undercuts are formed by rocks and can lead to the same untimely end for a boater as the force of the current wraps or pins a boat around the obstacle. Water is extremely powerful and unrelenting, and a river in flood stage is a menace when it bursts its banks because it is typically filled with all these hazards and more. However, water is also predictable, so we learn about these common hazards and the behaviour of water around them, and we strive to avoid them.

There are foreseeable perils and pitfalls on the path to engineering requirements that demand the same care and attention before the project momentum becomes overwhelming, such as the inability to reach consensus on key requirements, unresolved conflicting requirements and progressive requirements scope creep. Catalogue these hazards, learn how to recognise them and do not intentionally set off on your engineering journey if these hazards are overwhelming

The Deerfield River in flood stage during Hurricane Irene

Portage is (usually) always an option

and you have no tactics to deal with them. Further, if you find that your projects burst their banks with some frequency, either reinforce your processes and scoping practices or move on.

7. Safety Net or Portage

We wear helmets and life-vests on a river for our own personal safety, but we carry ropes for our buddies and we learn how to use them. When running a particular rapid is a high risk activity for a boater (i.e., the chance of success is outweighed by the cost of failure), we set up safety for them, which may comprise another boater on land and ready with a throw rope to pendulum any swimmer into an eddy before they head into an undercut rock. When the risk cannot be brought down to an acceptable level, or when we are simply having an 'off' day, we portage the rapid (i.e., we carry our boat and walk around it). Walking a rapid can be the sign of a very savvy boater who is in tune with his or her abilities on the day.

With projects, certainly examine the risks to requirements and put contingency in place, but also get better at understanding that this risk changes with time and perspective, and that sometimes there are requirements that simply have to be re-negotiated or shelved. Unless truly warranted, do not let attempting to satisfy one requirement jeopardise the entire project or exhaust the whole team. A healthy requirements

backlog essentially gives us things to work towards in the next version.

8. Prepare for the Worst

Entrapment, getting a foot or hand stuck between rocks on the riverbed, is one of the leading causes of death on a river. The force of the water flowing downstream will hold a person under water and lead to drowning. You therefore never attempt to stand up in moving water unless you are trained in swiftwater rescue and in its safe wading techniques. Coupled with the many hazards on a river, you are generally safer in your boat than out of it, so one of the first skills that we all learn is how to 'roll' (i.e., how to turn a boat upright, while still in it, after a flip upside-down). But, the saying goes that we are all 'in-between swims', so we also learn how to assume a safe swimming position for when that swim inevitably happens in hairy rapids (i.e., defensive and facing downstream, with your feet in front of you, until you can roll over and swim aggressively for an eddy).

Bad things do happen on projects, often as a result of a series of small miscalculations. Before we know it, we can easily find ourselves out of our depths and in danger of project drowning when trying to engineer unachievable or unwanted requirements. Think about the strategies and skills that will be needed to curtail or to deal with such issues before they happen and threaten the deliverance of requirements on your project. Maybe projects even need the equivalent of swiftwater rescue teams to come in to undertake requirements rescue and recovery operations, lending a more objective standpoint to project chaos? As they would demonstrate, proper professional training in the key skills is far more important than getting the latest and most expensive equipment, and the simplest approach to problem solving is usually the fastest to implement and effective.

9. Progress from Sneak Routes to Hero Routes

Boaters like to name rapids and the hazards within

Dave, practicing a rope-assisted swim in the Farmington River

Dave, taking one of the well-known routes in Fife Brook's Zoar Gap rapid

them (e.g., Oh S*** (OS) Rock in Fife Brook's Zoar Gap rapid). Naming enables boaters to discuss and share advice as to the various lines through rapids. Sneak routes indicate the simplest possible path down a rapid and are great for those boaters running rapids for the first time. Hero routes are generally high-risk moves, often undertaken right in front of intimidating hazards, demanding significant skill, guts or both. The smart boater learns to appreciate his or her own skills and only progresses with more difficult lines over time. In the course of this journey, they will experience failures and even 'out-of-boat' experiences, but with sufficient safety in place when learning, along with the capacity to learn from the instant feedback they get, they will usually progress to become better and / or wiser boaters.

With those individuals first learning to undertake requirements engineering tasks on projects, there are an abundance of things to think about and many skills to acquire to do an excellent job. The trick to skill progression is equally incremental in nature and depends upon finding just the right challenge at the appropriate time so that they can learn and retain enthusiasm for doing the job. While we may not realise it, we often send these new individuals straight into the project's equivalent of the Death Slot (the name of the hero route in the Dryway's Dragon's Tooth rapid). Do not set trainees up for failure after failure; build up a systematic training program for them so that they acquire the competence, confidence and judgement that they need so that they continue in the job.

10. Have Take-Out Rituals

What may at first appear to be a solo pursuit is very much a group activity. Every river-runner needs to be a self-sufficient member of an inter-dependent group, and we have to both value and trust the people we choose to run rivers with. It is therefore traditional to share drinks and snacks with your buddies at the end of

a river run. This gives us the opportunity to dissect any carnage and to re-run the most spectacular move of the day. It is these stories and their elaborations that reinforce lessons and take us forward with plans for making our next river run even better. Over time, it is these collective community experiences that inform the ongoing classification of rivers and the naming of their rapids and routes. It also enables groups to 'gel'.

Project retrospectives have always been the signature of the reflective practitioner, and the trials and tribulations of the requirements engineering process therein demand as much individual scrutiny. When you get to the take-out of your project, do not skip or postpone these very important knowledge-sharing and team-bonding rituals. Which requirements caused the most problems and why? How were they eventually tackled? What requirements techniques proved the most valuable at different stages in the project? Moreover, if the engineering team is to become more cohesive and advance as a unit going forwards, perhaps try to make these retrospectives a little more fun and a lot more social!

The Take-Out

Why don't you think of your next requirements engineering project as a white water river to be run? To run this river, you need to consciously choose and execute a dynamic process that demands knowledge, judgement, skill and a good team. You need to locate those all-important requirements rocks and scope out the project's riverbanks and riverbed. The rapids that these then form as knowledge levels vary may be intimidating at times, but with support and practice you will learn to understand the features, read the project water and find your route through. But, watch those pesky rocks; they have a way of presenting nasty obstacles or providing safe havens in equal measure. Also remember that it is frequently possible to go back upstream using eddies or portages if you break the rapid down and don't get swept along too fast. This means that you still often have the opportunity to re-scout those rocks and re-run the rapids a little differently if needed. A white water river is run one rapid at a time; always give them your respect once you put-in and you will reach that take-out.

If you have made it to the end of my whitewater-inspired guide to requirements deliverance, I think I should add a couple of important disclaimers:

- If you promptly go out and hire a class III / IV boater, don't assume that she / he will deliver your project's requirements. In fact, watch out for those 'sick days' after heavy rains.

- Just because you have delivered the requirements successfully on past projects, don't assume that you

are ready to run the Chattooga River[2]. Be sure to scout Deliverance Rock and listen to a little banjo music from the comfort of your armchair first[3].

Dr. Olly Gotel, New York City
2011
olly@gotel.net

RE: An Alternative Perspective on Microfinance

Camilo Fitzgerald

Recently I had a chat with the CEO of a microfinance institute in Nicaragua. We exchanged thoughts on the problems faced in microfinance and the types of solutions that requirements engineering can provide. I was very happy when the conversation culminated in a job offer in a research position for when I finish my doctoral thesis, which I gladly accepted. So, what exactly can RE do for microfinance?

Microfinance is a movement aimed at providing financial services to those in developing countries who would not otherwise receive them. Management of loans and other financial services has a cost that remains constant regardless of the amount of money involved. There is therefore little incentive for commercial banks to provide these services to poorer sections of society where it becomes difficult to break even. The microfinance movement aims to set up self-sufficient financial service providers that enable small businesses to manage their risk in developing countries – thereby stimulating economic growth.

There are many challenges in setting up and maintaining microfinance projects that remain largely unsolved since the movement's inception in the 70's. Many of these challenges are underpinned by the problem of providing services to the developing world that have been created and evolved in developed countries. From a software engineering perspective a parallel can be drawn with the migration of a legacy software system into a new business environment. The requirements that were valid for commercial banking simply do not fit those of the microfinance world – and

---

[2] The Chattooga River (Georgia, USA) was where a portion of the movie *Deliverance* was filmed. You can find out more about this river and the other rivers mentioned in this article here:
www.americanwhitewater.org/content/River/view/.

[3] If you are interested in learning about how to run white water safely, please contact representatives from either the British Canoe Union (www.bcu.org.uk/) or the American Canoe Association
(www.americancanoe.org/).

Community Based Saving Bank in Cambodia

a lot of redesign needs to be done at all levels of the software lifecycle.

The mismatch includes many examples. In the developed world a business owner can easily calculate their wage costs and report them to a lender, while in a predominantly agricultural economy where family members are employed for free and supported though school, clothing and food this becomes more difficult. Capital assets can be hard to calculate when they consist of a herd of cows. Little recourse is possible against defaulting borrowers.

Solutions have been implemented in a similar fashion to many legacy systems – hack the current architecture until it plays nice. An example that has had some success involves lending to groups of borrowers in a community to reduce management costs and encourage members of the group to protect against their neighbours defaulting. For the majority of developing countries, however, microfinance systems are still in need of redesign if they are to stand a chance of providing meaningful support the vast majority of those who do not have access to financial services.

So, taking an RE perspective, what techniques could be used to help?
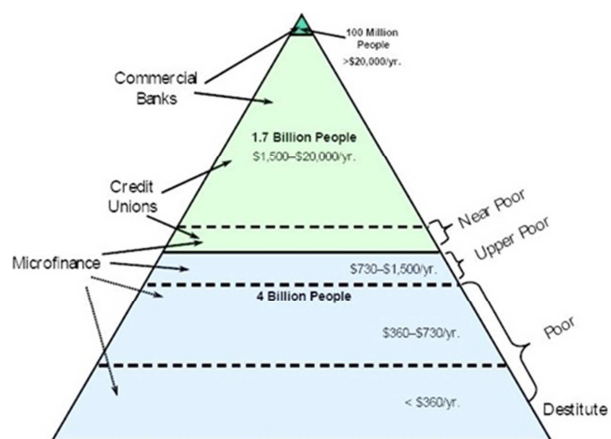
Users and Providers of Financial Services

Goal oriented analysis could provide a start. Constructing goal models for aspects of commercial banking may reveal areas of the model where there are mismatches with the goals for microfinance. Do goals need to be added or removed? Have conflicts or obstacles been introduced into the model as a result? Given a set of goals and requirements aspects of the 'implementation' of microfinancial systems could be traced back to the goals they satisfy. Are all the goals satisfied? Do we have redundancies?

Quantitate goal satisfaction analysis may also be applicable. Data on existing transactions in microfinancial systems could tell us the degree to which certain goals are satisfied, and alternative goals could be constrained or relaxed to order to maximise the satisfaction of higher level goals. For example, decreasing the level to which borrowers need to be accountable for their business when the risk of defaulting is lower. This approach may also introduce a feedback loop into the requirements thereby encouraging microfinancial systems to take on a 'self-adaptive' nature.

The self-adaptive paradigm could also be taken a step further by viewing alternative implementations as 'components' in a microfinancial 'architecture', which can be interchanged to maximise the satisfaction of higher-level goals with respect to a specific operating environment. A simple example of this approach would be to increase the number loans made to groups where borrowers are defaulting.

Advances in social requirements engineering could also be put to use. Feature request management systems, for example, could be used to source requirements directly from borrowers or get microfinance institutions across to the globe to collaborate on defining and evolving microfinance requirements. Social stakeholder discovery using a tool such as Stakesource might reveal previously unknown stakeholders in new microfinance projects.

An agile approach to developing new services could be employed, where lightweight 'throw away prototype' trial projects are run to explore and evolve requirements.

The microfinance context is a prime example of where RE techniques designed for software systems development could be applied to address issues in other complex large-scale systems. Should RE not have a higher profile?

The ideas here are presented at a high level, but I would be interested in discussing any further thoughts or comments on the topic: There is a discussion page I've opened up on the RESG LinkedIn Group.

© Camilo Fitzgerald, University College London

Best Requirements Management Platform?

David Gelperin

"Using Word or Excel for managing requirements is analogous to using roman numerals for arithmetic."

Introduction

When selecting a requirements management platform, most believe there are only two choices. One is to use word processors (Word) or spreadsheets (Excel). The other is to use complex commercial products.

This note identifies a third option that replaces the use of word processors or spreadsheets. It identifies products that are easy to use, inexpensive and functionally superior to Word or Excel for managing requirements.

Platform

We begin by identifying a set of basic requirements for a Requirements Management Platform (RMP). For example, a RMP should provide a core set of functions such as: create, update, delete, find/search, select, print, and export. The basic RMP requirements are:

A RMP must --
1. Enable user to choose types of requirements information to be recorded
2. Enable user to change information choices during a project at little cost
3. Enable user to create and manage outlines
4. Enable user to create and manage tables
5. Enable user to describe and manage states
6. Enable user to manage links between content elements
7. Enable user to manage links from content to external information
8. Enable user to perform Boolean searches of its contents
9. Enable user to print and export search results
10. Enable user to export RTF or Word files [for import into commercial RM products]
11. Enable user to collaborate with geographically distributed groups
12. Be actively supported by a reliable supplier

	Word Processor	Spreadsheet	Outliner / Personal Info Manager	Commercial Requirements Manager
Broader uses	✓	✓	✓	
Easy to learn	✓	✓	✓	
Easy to use	✓	✓	✓	
Supports outlining	✓		✓	✓
Boolean search			✓	✓
Selectable entries		✓	✓	✓
Core RMP functions			✓	✓
Diagram generation				✓
Test generation & management (and other ALM functions)				✓
Change control				✓

Figure 1. Capabilities of Alternative RM Platforms

Outliners

Based on these requirements and our experience, we make the following assertion:

Given current technology, any outliner complying with the basic RMP requirement is the best basic platform for collecting, analysing, and communicating natural language requirements.

Currently, at least two commercial outliners (WhizFolders and TreePad) comply with the basic RMP requirements and are inexpensive.

Although no web-based outliners are compliant, at least one (Checkvist) permits extended outlines (i.e., those with a notes area attached to each outline entry) to be imported and exported via extended OPML files. This provides easy access for stakeholders to review and comment on a requirements spec (extended outline).

Like word processors and spreadsheets, outliners have many other uses. If you do one or more of the following tasks –

- take notes
- write reports/papers
- design websites
- gather and organize thoughts/ideas
- write documentation
- perform root cause analysis
- analyze situations
- write training materials

- perform fault tree analysis
- manage lists
- write books/plays
- plan events
- manage contacts
- create agendas
- manage tasks
- create outlines
- manage issues/bugs
- create presentations
- manage changes

– and you are not using an outliner (or a special-purpose product), you are working too hard.

Word processors and spreadsheets are common, but poor choices for requirements management since neither complies with the basic RMP requirements.

Commercial Systems

Most commercial requirements managers comply with the basic RMP requirements as well as providing additional capabilities (e.g., rigorous change control, diagram generation, test generation and management -- along with other application life-cycle management (ALM) functions). These systems are more complex and much more expensive than compliant outliners, but are useful when their additional capabilities are needed.

Comparing Alternative Platforms

(*See Figure 1*)

Outsourcing

When customers (e.g. government agencies) outsource development, it is common practice to provide requirements specs as documents or word processing files. Either choice provides unmanageable information that developers must recover and import into their requirements manager. The alternative is to provide requirements information in a (compliant) outliner file.

Educational Opportunities

Outliners make it feasible for requirements courses to provide realistic experiences with requirements management activities. Students should already have access to an outliner to help them with activities listed in section 3.

Experience base

This note is based on experience developing a multilevel RM template (LiteRM) for a compliant outliner (WhizFolders). This experience has provided empirical evidence for the assertions above.

David Gelperin

CTO & President of ClearSpecs Enterprises

david@clearspecs.com

RE-writings

A Call to Action: Vision for Requirements Engineering Revisited

Ralph Young

"Engineering can provide a life of genuine satisfaction in many ways, especially through ministering in a practical manner to the needs and welfare of mankind."

Vannevar Bush

"No part of the delivery process is as critical, nor as difficult – because requirements map the human world to the technological world."

Jim Highsmith

Background

William Heaven requested that I write a sequel to my article that was published in the July 2001 issue of *Requirenautics Quarterly* (RQ)[4] that would provide an assessment ten years later regarding the extent to which my vision has been fulfilled.[5] One might argue

---

4 The article is available online at
www.resg.org.uk/images/6/65/RQ24.pdf, pp. 3-6.
5 In this article, I make generalizations. Also, I suggest that people need to act differently; however, it is not my intent to imply that professionals in the requirements engineering field are less capable than those in other fields. In addition, I have not provided suggested solutions in all cases (although the information in the provided references should be helpful). I do not pretend to have all of the answers. My intent is to reflect thoughtfully on what I have learned with the hope that some of the ideas will be helpful for others to explore and apply, resulting in

that great progress has been made. My perspective is that things have improved somewhat – for example, there are new professional organizations that are looking at the issues from a broader perspective; certifications are being offered and earned; research is being done concerning the return on investment of systems and requirements engineering; there have been improvements in the availability of useful and effective automated tools; and there have been other technology advances. These initiatives can improve the understanding of what needs to be done. Professionals including the readers of RQ have been working hard to grow and improve the field. Yet, clearly there are ample opportunities for further progress.

Introduction

My vision for the goals for a contributing requirements engineering field in July 2001 was: 1) satisfied customers, 2) systems and software engineers who are fulfilled by their work, 3) trained and experienced requirements analysts, 4) projects completed on schedule and within budget, and 5) less than 15% rework and less than 10% wasted resources on development projects.

The key points of the article included a set of opportunities for requirements engineers and analysts to improve project success rates; suggestions concerning how requirements engineers can help project managers, customers, and developers; the observations that striving for excellence should be our standard for daily living and that we needed to do things differently; and a challenge to readers to make

---

improvement in the practice of requirements and systems engineering.

ever stronger efforts to address requirements-related problems.

Requirements engineering may be defined as an area within the broader fields of systems and software engineering as well as the business analysis profession that has emerged since 2003 that focuses on the requirements process, a full system life-cycle set of actions concerning the necessary attributes of systems. Systems engineering (SE) concerns how to parcel out a huge task into smaller doable pieces so that when they are integrated, they work effectively and a customer and user group find value as a result.

I believe requirements engineering is critical because the requirements provide the basis for all of the other work that is performed on a project. One could argue that if the requirements work is not performed well, there is little chance that a project will be successful. Moreover, we know that the requirements will change while the system is being developed; therefore, we require a methodology that will accommodate requirements changes and new requirements.

Among my goals for this article, the most important is to motivate you to take additional actions to further strengthen and improve the practice of requirements engineering. My hope is that this article will stimulate some ideas and that you will be motivated to take action on them.[6]

The Data

One of my mentors emphasizes "Manage by fact". In other words, get data and use it to guide decision-making. If we look at current data concerning the components of my vision, they are not very encouraging:

Trained and experienced requirements analysts: Total number of CSEPs[7]: 802 (as of November 2011);[8] 25% of the members of IIBA have been business analysts for more than 10 years; 31% have a Masters Degree; 18% have vendor certifications; 10% CBAPs; 10% PMPs; 80% still using MS desktop tools to manage requirements.[9]

Projects completed on schedule and within budget: A 1998 Standish Group Report indicated that 26% of projects were successful, 46% were challenged, and

28% failed.[10] Today, for information technology (IT), 37% of projects are successful, and we don't measure business benefit/value/project return on investment (ROI).[11]

Less than 15% rework and less than 10% wasted resources on development projects: Worldwide, we are losing over USD 500 billion per month on IT failure and the problem is getting worse.[12]

Note: Data concerning components 1 and 2 of my vision are beyond the scope of this article. However, my view is that things probably haven't changed much.

Perhaps underlying the fact that not much has changed is that we have approached change as if the program or project manager or program management office can achieve the needed change without a systems approach involving the entire organization. Current project environments are large, complex adaptive systems – it is very difficult to make a difference in a very interconnected and complex (even competing) landscape.

PMs (as well as other professionals addressed in this article) are ultimately constrained by obsolete management expectations and politics and are not incentivized or encouraged to embrace and model evolving practices. In fact, I would be so bold as to say they are penalized. Another key factor is that the consumers of the work products are not being trained and expected to evolve their ability to consume. Corporate culture allows adoption resistance to prevail and the other excuse is that "we don't have time to do it right".[13]

Additional Insights

Capers Jones has devoted his professional career to the assessment and improvement of quality. Capers pointed out in 1977 that up to a point, the projects that achieve the lowest defect rates also achieve the shortest schedules, and that most projects can shorten their schedules by focusing on fixing defects earlier.[14]

---

6 A new organization, the Americas Requirements Engineering Association, is dedicated to this goal. See www.a-re-a.org. Tom Love shares his article, "Sorting Out the Terms" there – the article provides useful insights concerning agile, requirements, collaborative, and lessons learned.

7 All acronyms will be defined.

8 INCOSE, November 2011.

9 Forrester/IIBA, 2010.

---

10 Standish Group, 1998.

11 Standish Group, 2010.

12 Roger Sessions.

13 Peer review comment contributed by Anne Hartley, AH Consulting LLC – see www.annehartley.com. Ms. Hartley is leading an effort to form a consortium, the Business Analysis Leadership Consortium (BALC) to elevate critical transformation capability.

14 *Applied Software Measurement: Assuring Productivity and Quality*, 2nd ed., 1977. 80% of all product defects are inserted in the requirements definition activities as documented in *Effective Requirements Practices*, p.79.

His view is that systems engineering needs added discipline.[15]

Professional development in the form of certification programs has increased significantly (Table 1). The data reflect intensive efforts by several international requirements-related professional organizations to work toward improving requirements engineering and business analysis in practice by providing certification programs. The International Council on Systems Engineering (INCOSE) among other professional associations including the Project Management Institute (PMI), International Institute of Business Analysts (IIBA), and The International Requirements Engineering Board (IREB) have initiated and supported certification programs over the past decade (Certified Systems Engineering Professional [CSEP]), Project Management Professional [PMP]), Certified Business Analysis Professional [CBAP] and Certification of Competency in Business Analysis (CCBA)[16], and Certified Professional Requirements Engineer [CPRE]) respectively. This is encouraging. We have witnessed the development and publication of INCOSE's *Systems Engineering Handbook* (SEH), PMI's *Project Management Book of Knowledge* (PMBOK), and IIBA's *Business Analysis Body of Knowledge* (BABOK) and an agile extension to it.[17]

INCOSE痴 Systems Engineering Effectiveness Working Group (SEEWG) under the leadership of Joe Elm is performing insightful and useful research concerning the return on investment (ROI) from applying systems engineering on projects.[18] The results of this research have shown that projects that apply systems engineering best practices effectively are 40% more likely to succeed than projects that do not. Honour has progressed with similar research over a period of 12 years. In prior work, his SE ROI Project reported that effective SE at a level of approximately 16% of a project's cost optimizes cost, schedule, and stakeholder success measures.[19] In current work,

---

15 Personal email to the author, October 23, 2011.
16 The latter is an intermediate-level certification for professionals who wish to expand their career options and obtain recognition for their ongoing investment in their professional development. See www.theiiba.org.
17 IIBA's draft for public review of its *Agile Extension to the BABOK Guide,* November 2011 is available at www.iiba.org.
18 See INCOSE's Systems Engineering Effectiveness Working Group (SEEWG) website, www.incose.org/practice/techactivities/wg/seewg/. Mr. Elm is performing this research in coordination with the National Defence Industrial Association (NDIA), the IEEE Aerospace and Electronic Systems Society, and Carnegie Mellon University (CMU).
19 Eric Honour, "Sizing Systems Engineering Activities to Optimize Return on Investment", *Proceedings of the INCOSE International Symposium*,

Honour identifies the optimum level of SE total and for each of eight SE activities, including mission/purpose definition, requirements engineering, system architecting, system integration, verification / validation, technical analysis, scope management, and technical management/leadership.

Performance Based Earned Value[20] (PBEV) was published in 2007. It was written to advocate a change in industry practice concerning Earned Value Management (EVM), specifically, basing earned value on the successful completion of quality work products rather than on the amount of effort that has been performed. Paul Solomon has continued to advocate strongly for the needed change.[21] Unfortunately, Eleanor Haupt's[22] hope that was expressed in the Foreword of this book has not been fulfilled - that "Solomon and Young propose a new approach (to EVM) that relies on system engineering standards and maturity models that enhances the value of EVM through more objective assessment of true work performance and product quality…they have not only advanced the practice of EVM; they have achieved what many have only dreamed of: proving the worth of EVM to the technical community". As a result, millions of dollars continue to be wasted due to deployment and use of ineffective EVM techniques.[23] What is required to address effectively the obviously needed change for which a proven solution has been provided? More generally, why is it that improved practices, information, and tools are not used? Why aren't more effective techniques embraced by practitioners quickly, easily, and enthusiastically? This suggests that an enterprise-wide transformation initiative is required.

Perhaps we are using the wrong mental model for how systems are built. People often refer to it as the "development" process. Perhaps that is too simplistic, and in reality, most of the time it is a "discovery" process instead. "Development" implies that we basically know how everything works, and how to put it together, and it is primarily a problem of assembling the parts, like the construction of a building. "Discovery" implies that there are things that we will uncover as we move forward on the project, and which

---

Denver, Colorado, USA, 2011. Mr. Honour was awarded the "Founder" award from INCOSE and was the 1997 INCOSE President. See www.honourcode.com.
20 www.amazon.com/Performance-Based-Earned-Value-Practitioners-Solomon/dp/0471721883
21 See pb-ev.com/aboutus.aspx; see also "Improving the Quality of Earned Value Management Information", journal.thedacs.com, July 2011 Vol 14 No 3.
22 Past President, Project Management Institute – College of Performance Management, August 2006.
23 PBEV, p. iv.

we could not plan for at the beginning. In fact, the whole "life-cycle" model is based on the traditional model of building a building, which is a development process, not a discovery process.[24]

A key difference is that software engineering, in our current state, is still primarily different from that of building construction has to do with testing. Software has to be tested, extensively, to ensure it works right. In building construction, there is little testing. In other words, when they are constructing a building, they don't have to take each I-beam out and test it to make sure it is strong enough. All of that proofing that the component will work has been done through a very sophisticated set of standards, rules, regulation, and culture. The structural engineer merely does the analysis to determine what maximum loads could be imposed on the structure; then he goes into a table and looks up the I-beam that meets those loads and specifies it to the people doing the construction. He does not need to take the I-beam out and load it with a bunch of sand-bags to prove that it will work, before he can install it in the building. But that is what we have to do every day with software.[25]

Improvements in Automated Requirements Tools

There has been significant improvement in the availability of automated requirements tools over the past decade. A recent issue of *The Systems Engineering Newsletter* (SyEN) published by Project Performance International (PPI) provides a comprehensive list of requirements tools.[26] Another issue of *SyEN* describes a tool called inteGREAT available from eDev Technologies in Toronto, Canada.[27] This tool is inexpensive and offers enhanced collaboration capabilities to help evolve the real requirements.[28] Ian Alexander provides descriptive information concerning requirements tools on his website.[29] However, in my opinion, leadership, commitment (not just financial), training, and experience in applying automated tools effectively are required to make good use of any tool. There is some evidence that 'heavy' automated tools can make things worse when there isn't a suitable culture in place.[30]

Technology

Among the technology improvements we have witnessed in the past decade are vastly improved collaboration capabilities, capability to generate, store and use data at lowered costs, development of ubiquitous networks and internal access, mobility (laptops, cell phones, tablets), security (remote network access), and consumerisation (personal iPhones that enable integration with work lives).[31] Simply put, technology advances should have helped improve the delivery of successful projects – yet we still have an unacceptable number of failed projects a decade later. We still lack the ability in many situations to deploy highly functioning development teams and to deliver the desired results (implement the real requirements, on time, within budget, and in doing so, add value).

From Agile to Evolutionary Development

The Agile Revolution has taken hold worldwide. It has had an impact on expectations, by giving the impression that we can do anything. Or, the other way round, one could assert that our need for instant success has resulted in the agile movement. A long-term result of the agile approach may be a major impact on human factors. For example, over the long term, there may be a loss of knowledge. It seems there is likely to be a convergence over the next five to seven years in the preferred development approach – we will find that agile development is not a panacea. Considering alternative development approaches, it seems likely that evolutionary development (EVO) as advocated by Tom Gilb[32] offers advantage to both customers and developers because the methodology supports discovery, identification, and implementation of the real requirements for a desired capability. Further, it helps us deal with complexity. Although EVO was first described in print in 1988, it is among other rarely used innovations, including risk management planning, JAD sessions, design for change, incremental integration, inspections, and continuous improvement. Most of these innovations have been around for 25 years or more. Why aren't they being used more extensively? We are confronting a problem of slow diffusion of innovation and best practices to the field (everyday practice). We are on

---

24 Insight provided by peer reviewer John E. Moore.
25 Ibid.
26 www.ppi-int.com/newsletter/SyEN-036.php#tools
27 www.ppi-int.com/newsletter/SyEN-035.php#tools ; see also www.edevtech.com/.
28 See *Effective Requirements Practices*, pp. 58-91, for a detailed discussion concerning how to evolve stated requirements into real requirements.
29 www.easyweb.easynet.co.uk/iany/
30 Requirements Engineering in Automotive Development – Experiences and Challenges." IEEE Joint International Conference on Requirements Engineering, 2002.Available at

ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=1048 546.
31 Observations of Jeff Young, Chief Technology Officer, FactSet Research Systems, November 3, 2011. See *The Lights in the Tunnel: Automation, Accelerating Technology, and the Economy of the Future* by Martin Ford.
32 See www.gilb.com/Blog.

the brink of significant improvement, but we aren't there yet.[33]

Leadership

I and others have identified the need for better leadership both as a root cause of the limited improvement we have experienced and as a critical component of the solution. What is meant by "better leadership"? A leaderly executive will take time to figure out where she or he can weigh in and help. She will identify and apply a set of key principles that enable her to craft and communicate a vision, support staff, delegate, motivate, recognize and acknowledge effort, and hold people accountable for desired results. These principles need to address evolving the real requirements for the project, team building and teamwork, empowerment, process improvement, time to delivery, a development approach that engages customers, a development plan for each member of the staff, and proactive steps to constantly improve communications. The program or project manager (PM) must be given the opportunity and responsibility to manage. She should not be expected to also perform marketing, product development, recruiting, and organizational committee tasks, among many other tasks that PMs are often assigned. We often expect more of organizations; however, organizations are powerless without individual leaders.

Empowerment

During my career, I have had the opportunity to participate in and lead empowered teams. My experience is that an empowered team can accomplish anything it sets out to do. In one instance, it was considered impossible to achieve an objective, but the Team underwent an organizational effectiveness[34] process and found that it was able to achieve the objective through effective leadership, commitment, and strong teamwork. One framework that has worked very well in my experience is "partnering".[35] Through use of this technique, we were able to overcome barriers at an important U.S. Federal Government department and to achieve major successes. When project partnering (or "Enterprise Partnering") is not used, "business as usual" continues. The disappointing thing is that project management curriculums in

universities and at PMI have not made partnering a distinct part of learning. They must be assuming it is merely good project management so there is no need to do so. Newsflash: if it is not part of the standard lexicon, it will never survive except through a few dedicated and enlightened individuals.

Possible Opportunities for Stronger Improvement

For us to achieve significant changes, an enterprise-wide transformation initiative is required that takes into account the following areas:

1. **Increased Expectations** – of customers, providers, managers, and ourselves. Simply put, the bar is not sufficiently high. Expectations have much to do with results.[36] We need to work smarter to create cultures of excellence.[37]

2. **Stronger Leadership, Increased Responsibility and Accountability, and Greater Motivation** – see *Execution: The Discipline of Getting Things Done* for help in establishing a set of key leadership principles.[38] Neal Whitten's summary of leadership advice from those who said it best also provides insights into aspects of this root cause.[39] Whitten's "10 Best Practices to Promote the Advancement of Project Management in Your Organization" provides specific suggestions for how to do better.[40] Specifically concerning requirements engineering, the ten "Opportunities for Requirements Engineers to Improve Project Success Rates" I provided in the article are still relevant. As I noted then, it is management's job to provide a motivating work environment in which people can be effective and grow. I describe practical application of many of these aspects and provide an extensive set of references in *Effective Requirements Practices*.

---

33 Steve McConnell, *After The Gold Rush*, pp. 144-147. See also Everett M. Rogers' *Diffusion of Innovation* and Geoffrey Moore, *Crossing the Chasm*.

34 See swat.gis.ksu.edu/documents/org_effectiveness.pdf for a thorough discussion of this powerful technique and references.

35 See www.facilitationcenter.com/ to learn more about facilitated partnering. Contact Charles Markert (Charlie@FacilitationCenter.com) to discuss using this powerful technique.

36 See Roger Connors and Tom Smith, *How Did That Happen: Holding People Accountable for Results the Positive Principled Way* for powerful insights. The Penguin Group, 2009.

37 Actually, Steve McConnell provided an insightful description of a recommended pathway forward twelve years ago. See *After The Gold Rush: Creating a True Profession of Software Engineering*, Microsoft Press, 1999. The second edition of this book has a different title, *Professional Software Development: Shorter Schedules, Better Projects, Superior Products, Enhanced Careers*; Addison-Wesley, 2004. See also www.stevemcconnell.com/.

38 Larry Bossidy and Ram Charan, Crown Business Publishers, 2002.

39 PM Network, October 2010, www.pmi.org

40 www.nealwhittengroup.com

3. **Stronger and More Effective Teamwork.** I devote a lot of emphasis to this in *How to Save a Failing Project*.[41] As noted previously, if you have experienced the blessing of leading or being a member of an "empowered team" (as contrasted with a dysfunctional team), you know the huge difference this makes. We need to embrace more modern approaches to project team leadership and leveraging entire team competency, for example, allowing the lead requirements/business analyst to serve as a partner of the project manager (and that not be interpreted as the PM being insufficient).

4. **Strengthened commitment of individuals, projects, and organizations to continuous improvement.** We should all aspire and work smarter to become ever better. An example that comes to mind is the general failure to utilize and leverage so-called "lessons learned" – in my experience, lessons are often observed but rarely learned and acted upon.[42]

5. **Stronger senior management commitment**. It has been my experience over 46 years of professional activities that it is difficult to accomplish significant efforts on projects and in organizations without senior management commitment. If one can't gain senior management sponsorship for a significant effort, the probability of successful completion of the effort is limited significantly. Sarah Sheard's article explains what senior management commitment is and also how to achieve it. [43]

6. **Increased Use of Effective Practices, Information, and Tools that are available**. It's not that we don't know or can't find out what to do; it's that we don't do it. It's been my observation over many years that many PMs do not read extensively nor do they take the time and effort to understand, learn, and apply methods, techniques, processes, and tools. In my experience, it is not sufficient for a PM to depend upon staff to provide the leadership to take good advantage of these capabilities. The PM's

understanding, leadership, and commitment are prerequisite to effecting positive change.[44]

7. **Investment of additional time and effort up front to evolve the real requirements**.[45] The responsibility here is with customers and managers who are obsessed with getting on with "the real work" (coding). Contractors are also responsible to provide informed, honest advice. Individuals are responsible to advocate for improvements. Requirements people need to prevail upon PMs to explain to our customers the value of taking the time and effort to evolve the real requirements before initiating other technical work. Ample evidence is available to make this case.

8. **Increased honesty and openness in human interaction, and better communication management.**[46] Steve Gaffney has devoted his professional career to developing and providing practical ways to address this. He provides seminars that make a difference on projects and in organizations.[47]

9. **Reduced rework.** Industry data indicate that the average amount of rework performed on the typical project is 45-50% of total effort and that this metric has not changed over the past decade.[48] This is an area where the PM and other leaders can foster an environment to chip away at this waste. What approaches might you provide to address this on your project or in your organization?[49] A lucrative place to initiate improvement is at project startup.[50]

10. **Training and implementation of a simple defect prevention initiative/process**. Utilize the extensive knowledge of your devoted project staff

41 See Chapter 6.
42 See Norman L. Kerth, *Project Retrospectives: A Handbook for Team Reviews* and Ester Derby et al, *Agile Retrospectives: Making Good Teams Great* for sound advice concerning how to make good advantage of lessons learned.
43 See Sarah Sheard, "What is Senior Management Commitment?" for an insightful description of this critical role. Available at www.ralphyoung.net/articles.html.

44 See *Project Requirements: A Guide to Best Practices* for an extensive array of ideas, suggestions, and recommendations.
45 *Effective Requirements Practices*, Chapter 4.
46 "How many of us have gotten our colons in a wad as a result of a misunderstanding?" JoAnne Owens-Nausler. See *Making Do out of Doo-Doo: Lessons in Life for Hardiness, Health, and Fitness*, Dageforde Publishing, Inc., Lincoln, Nebraska, USA, www.dageforde.com.
47 See Steve's website, www.stevengaffney.com/. Even better, invite Steve to provide a workshop for your project or organization. This will help.
48 *How to Save a Failing Project,* pp. 32-33.
49 See *Effective Requirements Practices* for ideas and suggestions.
50 Thomas J. Neff et al, *You're in Charge, Now What?: The 8 Point Plan*.is a useful resource

by sponsoring a brainstorming session to identify major causes of problems. Multi-vote to determine the groups' sense of their relative importance, impact, and capability to address them. Then brainstorm the group's suggestions for how to address the high priority causes of problems. Your staff has an enormous capability to help you, if only you let them. Undertaking this process can be an effective team building exercise, in addition to enabling removal of defects earlier.[51]

11. **Inculcate requirements engineering and partnering into university and PMI Curricula and make them standalone topics of learning**. Without doing this, these techniques will take a backseat and never become mainstream.

12. **Transition of program and project managers' mind-set from manager to leader**. PMs need to transition into the role of a leader by strengthening skills such as visioning, relationship building, influencing, negotiating, creative thinking, innovating, communications management, empowering, building teamwork, acknowledging effort, teaching defect removal techniques, mentoring project performers, employing systems engineering effectively, and setting the tone (and the requirement for) excellence.[52]

Needed: A Broader Scope for Requirements Engineering

When I attended the Project World Conference in Orlando, Florida USA in 2006, I became acquainted with another large group that is also working toward improving and utilizing requirements: the Business Analyst Community. The mission of the previously mentioned International Institute of Business Analysis (IIBA) is to develop and maintain standards for the practice of business analysis and for the certification of its practitioners. The IIBA vision is to be the leading worldwide professional association for business analysts. Being an invited speaker, I inquired of my audience of approximately 200 how many would consider themselves aligned with the area of requirements engineering – after explaining what requirements engineering is and what requirements analysts, requirements engineers, and requirements managers do, almost all agreed that the two areas seem very aligned.

Another of our colleagues, Neal Whitten, has been an advocate of project management best practices for many years. In 1998, Neal wrote an article, "Meet Minimum Requirements: Anything More is Too Much".[53] Neal is associated with PMI. As we know, the goals of effective project management are very similar to the objectives of requirements engineering: complete the project successfully, on time, within budget, and add value. Neal maintains a sharp focus on the role of leadership in enabling successful accomplishment of significant efforts.[54] See his website for additional insights and resources (www.nealwhittengroup.com).

Kathleen Hass, a member of the IIBA Board of Directors, has a consulting business that specializes in business analysis and complex project management.[55] She and other experts have made the case that it is the gap in business analysis and complexity management capabilities and competencies that is the root cause of the inability to deliver successful business solutions.[56] She provides a set of building blocks that she feels will enable us to proceed down the "daunting road ahead" and a BA Practice Maturity Model that she feels provides a framework to move forward more effectively. She advocates forming Business Analysis Centers of Excellence (BACOE) to guide the needed improvement efforts. Hass feels that the BA / Requirements Engineer role will change dramatically once we have wide adoption of tools like inteGREAT to manage the fidelity of requirements and keep all requirements artefacts in sync and up to date. "I suspect that BAs spend about 50-80% of their time documenting, managing changes to artefacts, and tracing requirements. Once the tool does that for them, their role will be much more of a change agent, creative leader, innovator, team leader, and influencer."[57] I am not as confident as is Kitty concerning this happening anytime soon. Hass also points out that IT projects are complex, and we don't

---

51 A simple defect prevention process is described on pp. 177-180 of *How to Save a Failing Project*.
52 A good resource is *Emotional Intelligence for Project Managers: The People Skills You Need to Achieve Outstanding Results*, by Anthony Mersino, PMP.

53 *PM Network*, September 1998:12(9)19, available at http:nealwhittengroup.com/1998/.
54 See "Words of Wisdom: Leadership Advice from Those Who Said It Best", PM Network, October 2010, www.pmi.org.
55 Ms. Hass has authored *Managing Complex Projects: A New Model* and a series of six books that comprise "The Business Analysis Essential Library". In addition, her most recent book, *The Enterprise Business Analyst: Developing Creative Solutions to Complex Business Problems*, aids the transition from a tactical, project-focused role to a creative, innovative role. See www.ManagementConcepts.com/pubs.
56 See "Planting the Seeds to Grow a Mature Business Analysis Practice," White Paper published by Kathleen Hass and Associates, www.kathleenhass.com, p. 3. A companion White Paper is "Business Analyst Proficiency: How Capable Do I Need to Be?"
57 Personal email to the author, November 8, 2011.

know how to diagnose, manage, and capitalize on complexity. "Complexity breeds creativity if we know how to leverage it." Hass has developed a Project Complexity Model to diagnose the size, complexity, and risk of a particular project.[58]

Leveraging Others' Efforts

Given that we have many disparate professional organizations and groups advocating and working toward improvement in requirements, one of the challenges is the risk of these groups becoming increasingly fragmented. We need to ensure that this doesn't result in more distinct rival blocks – in other words, the organizations and the individuals in them need to "reach out" to others rather than focus inwardly. Related, from my own writing experience, through contacts with many other authors in several countries, I have observed a need for added inclusion and collaboration across geographical areas - we need to be even more collegial in overcoming the islands of the U.S., Europe, Australia, China, Japan, and on, sharing with and learning from one another.

Toward a Profession of Requirements Engineering

In 1999, Steve McConnell, a leader in defining software engineering's best practices, authored a book, After The Gold Rush: Creating a True Profession of Software Engineering (ATGR). My note to myself after reading it in March of 2000 was: "A great book. One of the very best in software engineering." Steve's comment that "the practices needed to create good software are well established and readily available, but there is a chasm between the average practice and the best"[59] applies to requirements engineering as well. The theme of Steve's book is how to change. Close to my heart, Steve notes that the most frequent causes of software project failure have to do with requirements problems, which as he notes were prevalent as early as 1969. Another pearl is that "new tools are useful, but not a substitute for clear thinking." A conclusion is that "despair arises from the fact that some problems have been with us for a quarter century or more and are still common!"

Steve informed me that between publication of ATGR in 1999 and Professional Software Development in 2004, he came to a better understanding of how licensing will be needed - he now believes that only a tiny percentage of software engineers will need to be licensed, and that several of the of the things he suggested in ATGR apply only to those practitioners who need to be licensed.[60]

Below are the things McConnell suggested in ATGR that we can do to achieve a more mature profession of software engineering that apply equally well to both requirements engineering and systems engineering:

- Implement a mandatory code of ethics/standard of professional conduct, and meaningful enforcement authority
- Have a requirement for initial professional education
- Accreditation
- Skills development
- Raise the average level of practice
- Increase expectations, for example, adopt a code of ethics that imposes higher standards than those normally tolerated in the market place
- Emphasize social responsibility and duty to behave as members of a disciplined and honourable profession
- Require a prerequisite of a license prior to admission to practice, e.g., professional engineer
- Ongoing professional development
- Participation in professional societies
- Organizational certification/independent external evaluation of practices
- Individual certification
- Continuing education
- Thoroughly diffusing best practices into industry
- Management commitment to continuous improvement

Adapting Steve's generalization concerning software developers, we can observe that requirements engineers obtain our occupational education from the school of hard knocks; experience can be a good teacher, but it is also slow and expensive. In summary, we are in a position to transform average practice into best practice.

Call to Action

As Neal Whitten so aptly observes, "If you want to make things happen, do not rely on others to lead the way. Leadership is not only about the ability of those around you to lead, it's also about your ability to lead despite that which is happening around you." Organizations need leaders who speak for and insist on improvement.

The concerns and suggestions to deal with them that I discussed in my July 2001 article are as valid today as

---

58 See "Living on the Edge: Managing Project Complexity" available at www.ralphyoung.net/articles.html. See also Howard Eisner, *Managing Complex Systems: Thinking Outside the Box*, John Wiley, 2005.

59 Microsoft Press, 1999, p. ix.

60 Personal email to the author, 11/12/2011.

then in order to continue the progress that has been made. Talented and experienced practitioners, academicians, researchers, contractors, customers, developers, vendors, and authors have offered advice concerning how things can be done better. It seems clear that in order to make progress more quickly, individuals and groups need to address the root causes of limited improvement over the past decade that have been identified in this article. During my career, I have experienced highs and lows in the effectiveness and success of the projects in which I have been involved or led. The highly successful efforts were characterized by strong leadership and the other aspects that I have discussed in this article. If we are to achieve higher rates of improvement, we need to act more deliberately in the areas identified. Also, we need enlightened workforce management practices - there is a lot of talk about empowerment and personal initiative, but in reality, this is very much a function of enterprise maturity.

Conclusion

In this article, I have taken a broader perspective concerning my vision for requirements engineering than I did ten years ago. I have proposed opportunities to achieve stronger improvement. Much more significant improvement can be achieved by raising expectations, stronger and more understanding leaders who are able to lead and facilitate the work of others, stronger and more effective teamwork, establishing a culture of continuous improvement, strengthened senior management sponsorship, increased use of effective practices, information, and tools that are available, investment of time and effort to evolve the real requirements before other technical work is initiated, added honesty and openness, communications management, reduced rework, strengthened commitment, bolder actions, deployment and use of a simple defect prevention process, project managers taking the time and effort to learn and apply practices, information, and tools that are available, inculcating requirements engineering and partnering into university and PMI Curricula and making them standalone topics of learning, and transition of program and project managers' mind-set from manager to leader.

Perhaps our efforts to improve requirements engineering have not paid enough attention to the human elements that cause projects to fail, for example the management support issues noted above. However, it seems that the problem goes deeper and we need to identify more clearly what causes our best intentions to go astray. The field of behavioural economics may provide insights that will help.

John Moore has observed that people deal differently with negative information (mistakes, defects, and errors) than they deal with positive information (successes, accomplishments, and achievements). Since requirements engineering is still primarily a "discovery" process, there will be lots of opportunities for the organization to have to deal with negative information. Perhaps our "development" oriented processes don't take fully into account this human dynamic – we sometimes try to impose a success and goal-oriented model on a process that will, by nature, involve the uncovering of many problems, mistakes, errors, and misunderstandings. If the organization and its processes cannot deal positively with this negative information, then there is great potential for dysfunction and failure. This is an area that requires more exploration.

Are you ready? In other words, are you willing to rededicate yourself to taking added actions toward the identified opportunities to achieve stronger improvement? For it is only if we individuals ("leaders" [all of us] in Whitten's context) take actions that the rate of progress will increase. The references provided in this article offer a wealth of ideas, suggestions, recommendations, and advice concerning actions that will help executives, project managers, requirements engineers, and business analysts. One can strengthen continuous improvement by making good use of any of them.

Acknowledgements

Among those who have contributed to this article and stimulated my thinking are Ian Alexander, Anne Hartley, Kitty Hass, Capers Jones, Charles Markert, Joe Matney, Steve McConnell, John Moore, Kent Schneider, Asif Sharif, and Neal Whitten.

Dr. Ralph Young, CSEP
ralphyoung.net

Contributing to RQ

To contribute to *RQ* please send contributions to William Heaven (william.heaven@gmail.com). Submissions must be in electronic form, preferably as plain text.

Joining the RESG

Visit www.resg.org.uk/index.php/Join for membership details, or email Yijun Yu (Y.Yu@open.ac.uk).

RE-actor

The committee of the RESG

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

Chair:
Emmanuel Letier
University College London,
e.letier@cs.ucl.ac.uk

Deputy Chair:
Ljerka Beus-Dukić
University of Westminster
l.beus-dukic@wmin.ac.uk

Treasurer:
Steve Armstrong,
The Open University,
s.armstrong@open.ac.uk

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

Membership Secretary:
Yijun Yu
The Open University
y.yu@open.ac.uk

Publicity Officer:
Camilo Fitzgerald
University College London,
c.fitzgerald@cs.ucl.ac.ukk

***RQ* Editor**:
William Heaven
Freelance Writer / Consultant,
william.heaven@gmail.com

Post Graduate Officer:
Dalal Alrajeh
Imperial College London
dalal.alrajeh@imperial.ac.uk

Student Officer:
Ben Jennings
University College London
b.jennings@cs.ucl.ac.uk

Regional Officer:
Shehan Gunewardene,
CAP Gemini/Aspire,
shehan.gunawardena@hmrcaspire.com

Academic Member:
Peter Sawyer,
Lancaster University,
sawyer@comp.lancs.ac.uk

Academic Member:
Prof. Bashar Nuseibeh
The Open University
b.nuseibeh@open.ac.uk

Industrial Member:
Alistair Mavin,
Rolls-Royce PLC,
alistair.mavin@rolls-royce.com

Industrial Member:
Vesna Music,
Delphi Diesel Systems,
vesna.music@delphi.com

Industrial Member:
Suzanne Robertson,
Atlantic Systems Guild Ltd,
suzanne@systemsguild.com

Immediate Past Chair
Ian Alexander
Scenario Plus Ltd
iany@easynet.co.uk

Your Name Here!
Please contact any of the
committee if you'd like to
join us.

RE-sources

Books & Papers

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

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

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

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

IFIP Working Group 2.9 (Software RE):
eeat.cis.gsu.edu/ifip2.9/

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

Lists & Online Communities

RESG LinkedIn Group

www.linkedin.com/groups/Requirements-Engineering-Specialist-Group-RESG-2662234

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

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

ReQuirements Networking Group
www.requirementsnetwork.com

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