Provoking Creativity: Imagine What Your Requirements Could Be Like

Neil Maiden and Alexis Gizikis, *Centre for HCI Design, City University, London*
Suzanne Robertson, *The Atlantic Systems Guild*

Requirements engineering research, with its focus on elicitation, analysis, and management, offers little to support the creation or invention of requirements. Here, innovative techniques encourage creative thinking about requirements for an air traffic control system.

Requirements engineering isn't recognized as a creative process.[1] However, as new systems and products emerge, stakeholders are increasingly creating and inventing ideas that they express as requirements. Requirements engineering, with its focus on elicitation, analysis, and management, has yet to fully grasp this trend.

We applied techniques to encourage creative thinking during the requirements process for a software-based system in a naturally conservative domain—air traffic management (ATM). We applied unusual theories, such as analogical reasoning from cognitive science, to underpin the use of these techniques, and we report basic results and lessons learned. We focus on the creativity techniques we applied (see the sidebar) and demonstrate them with examples from the ATM domain.

Determining requirements for the CORA-2 system

We worked with Eurocontrol, the organization overseeing European air space, to design and implement RESCUE (*R*equirements *E*ngineering with *Sc*enarios for *U*ser-centered *E*ngineering), a process for determining stakeholder requirements. We applied RESCUE to CORA-2 (*Co*nflict *R*esolution *A*ssistant), a system that will provide computer-based assistance to air traffic controllers to help them resolve potential conflicts between aircraft. CORA-2 is a complex sociotechnical system. Air traffic controllers will resolve aircraft conflicts using resolutions and advice from the CORA-2 software. We wanted CORA-2 requirements to

0740-7459/04/$20.00 © 2004 IEEE

Creativity Theories Applied to the Workshops

We applied elements of Osborn's Creative Problem Solving (CPS) model[1] to structure each workshop. The model proposes iterative divergence from, then convergence on, ideas to find objectives, facts, and solutions. This well-established creativity model provided the framework for ordering workshop activities. We designed each half-day period to support divergent activities to generate numerous ideas and convergent activities to reduce this number and concretize ideas. This contrasts with the brainstorming structure and theoretical focus of Rapid Application Development workshops on group dynamics and consensus building[2] rather than on encouraging creative thinking per se.

We applied existing theories of creative processes[3,4] to facilitate divergence and convergence. Four processes are essential to creative thinking: preparation, incubation, illumination, and verification. Incubation is needed to handle complexity—during this relaxing period, people unconsciously and consciously combine ideas with a freedom that denies linear and rational thought.[5] During the subsequent and shorter illumination phase, a creative or innovative idea suddenly emerges, often at the most unlikely time in the most unlikely place. This "eureka" effect has been widely reported in creative problem solving. Creative process theories were adopted because their finer-grained processes let us decompose divergence and convergence in the CPS model into sequential workshop activities. The deliberate use of relaxed incubation periods, during which participants listened to music or looked at pictures, was another characteristic that distinguished our workshops from brainstorming activities that emphasize illumination.

We also designed the three workshops to encourage different types of creative thinking—explorative, combinatorial, and transformational.[5] We designed the first workshop to encourage exploratory creativity, in which people explore the space of possible ideas to create new ones. Although similar to brainstorming, our innovation was to encourage analogical reasoning—common in creative domains—to generate new ideas. Researchers have investigated analogical reasoning to support requirements reuse[6] and have shown that people can exploit analogies if helped to understand them.[7] On the basis of this previous success, we encouraged the participants in the CORA-2 workshops to go further and use knowledge transferred from the non-ATM domains to provoke creative thinking about requirements and high-level designs in the ATM domain.

We designed the last two workshops to encourage combinatorial creativity, which is the creation of new ideas from a combination and synthesis of existing ideas. Margaret Boden states that combinatorial creativity is characterized by the improbability of the combination, or the surprise encountered when such an unusual combination is presented.[5] Most existing techniques decompose requirements to make them more precise and easier to understand. We adopted combinatorial creativity in the two workshops to challenge these traditional trends to decompose and distinguish the workshops from other approaches.

We also designed the last two workshops to encourage transformational creativity. During transformational creativity, people change the solution space in a way that things that were considered impossible are now possible,[5] for example by challenging preconceived constraints and exploring new solutions to existing problems. During the third workshop, participants deliberately attempted to understand and sometimes transform the CORA-2's current constraints by exploring elements of the solution space.

Figure A shows how we combined the different creativity theories in the design of each CORA-2 workshop. We guided stakeholders to diverge from, then converge on ideas during half-day workshop periods divided into idea preparation, incubation, illumination, and verification processes, with different creativity techniques enabling idea incubation and illumination.

We didn't adopt other established creativity techniques such as cultural probes[8] because the CORA-2 workshops didn't provide direct access to the controller's work environment, which undermined the use of the probes. In addition, RESCUE required the workshops to produce an objective set of CORA-2 requirements and design ideas, in contrast to an impressionistic account of beliefs and desires sought using probes.

References

1. G. Isaksen and K. Dorval, "Changing Views of Creative Problem Solving: Over 40 Years of Continuous Improvement," *ICN Newsletter*, vol. 3, no. 1, 1993.
2. C. Floyd et al., "Out of Scandinavia: Alternative Approaches to Software Design and System Development," *Human-Computer Interaction*, vol. 4, no. 4, 1989, pp. 253–350.
3. J. Hadamard, *An Essay on the Psychology of Invention in the Mathematical Field*, Dover, 1954.
4. H. Poincare, *The Foundations of Science: Science and Hypothesis, The Value of Science, Science and Method*, Univ. Press of America, 1982.
5. M.A. Boden, *The Creative Mind*, Abacus, 1990.
6. P. Massonet and A. van Lamsweerde, "Analogical Reuse of Requirements Frameworks," *Proc. IEEE 3rd Int'l Conf. Requirements Eng.*, IEEE CS Press, 1997, pp. 26–37.
7. N.A.M. Maiden and A.G. Sutcliffe, "Exploiting Reusable Specifications through Analogy," *Comm. ACM.*, vol. 34, no. 5, 1992, pp. 55–64.
8. B. Gaver, T. Dunne, and E. Pacenti, "Cultural Probes," *Interactions*, Jan./Feb. 1999, pp. 21–29.

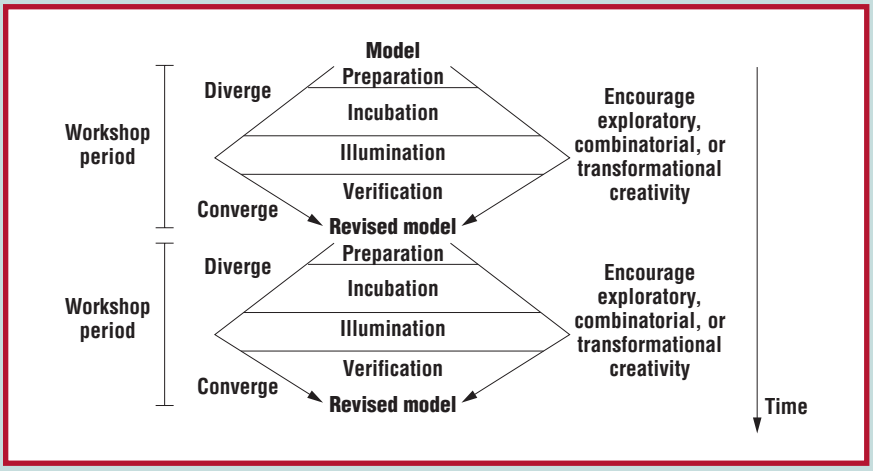

Figure A. The basic structure of creative periods during a CORA-2 workshop.

Table 1

Some analogical mappings between the air traffic management, textile, and music domains

ATM domain	Textile domain	Music domain	Generalization
Air traffic controller	Textile designer	Music composer	A human agent seeking to solve to a problem
Aircraft conflict	Textile-design problem	Composition problem	A problem state the human agent encounters that requires using a tool to resolve the problem
Conflict resolution	Textile pattern	Musical piece	The problem's solution
Resolution strategy	Style or genre of the textile pattern	Musical style	A reusable abstraction that defines the solution type's characteristics and discriminates it from other types
Resolution details	Pattern features such as motif and layout	Musical notes	Elements of the solution
CORA-2 system	—	Musical instrument	The tool the human agent uses to solve the problem

specify how controllers should work and interact with the software system as well as how the software system should function—for example, how to increase automated support for controllers without deskilling them.

The CORA-2 team consisted of one manager, two requirements engineers, two air traffic controllers who acted as domain experts, one human factors expert, and one technical expert. It applied the RESCUE process to establish the stakeholder requirements for the CORA-2 system. Prior to the workshops, the team acquired 50 stakeholder requirements from brainstorming and interview sessions.

Next, we held three one-day creativity workshops over two months to generate requirements and design ideas for CORA-2. Each workshop involved between 16 and 20 team members and stakeholders (managers, air traffic controllers, and technology experts). We designed each workshop's creativity periods on the basis of established creativity theories from cognitive and social psychology and artificial intelligence. Cognitive psychology defines creativity as "the ability to produce work that is both novel (i.e. original, unexpected) and appropriate (i.e. useful, adaptive concerning task and constraints)."[2]

Using theories selected from a wider review of the creativity literature helped us distinguish the CORA-2 workshops from other brainstorming processes (see the sidebar).

The creativity workshops

RESCUE didn't mandate that the team input any models or requirements into the first workshop. After the first workshop, the workshop facilitators encouraged communication between workshop groups via email to discuss requirements and ideas and aimed to create a distributed community.

One notable workshop feature was that we invited experts from domains similar (though not obviously) to aircraft conflict resolution to discuss their domains and encourage different types of creativity.

Exploratory creativity in the first workshop

To explore new ideas for conflict resolution, we invited a textile expert to discuss Indian textile design and a musician to discuss modern music composition. We encouraged participants to find analogies between these domains and ATM, then to generate new ideas about conflict resolution using those analogical elements. We carefully selected these two fields prior to the workshop on the basis of domain analyses that the facilitators undertook and examination of the Operational Concept of Use document that Eurocontrol had already developed for CORA-2. In all three domains—CORA-2, textile design, and music composition—people work with tools to construct and test complex solutions—that is, to construct and test conflict resolutions, textile patterns, and pieces of music. We selected textile design to use textile patterns' different qualities to trigger creative thinking about conflict resolutions. We selected music composition to exploit how a musician composes and tests a music piece in an effort to trigger creative thinking about how an air traffic controller generates and tests conflict resolution strategies.

To encourage incubation, experts presented and answered questions regarding their topics. We wanted to encourage participants to unconsciously and consciously form analogical mappings such as those Table 1 lists. One fa-

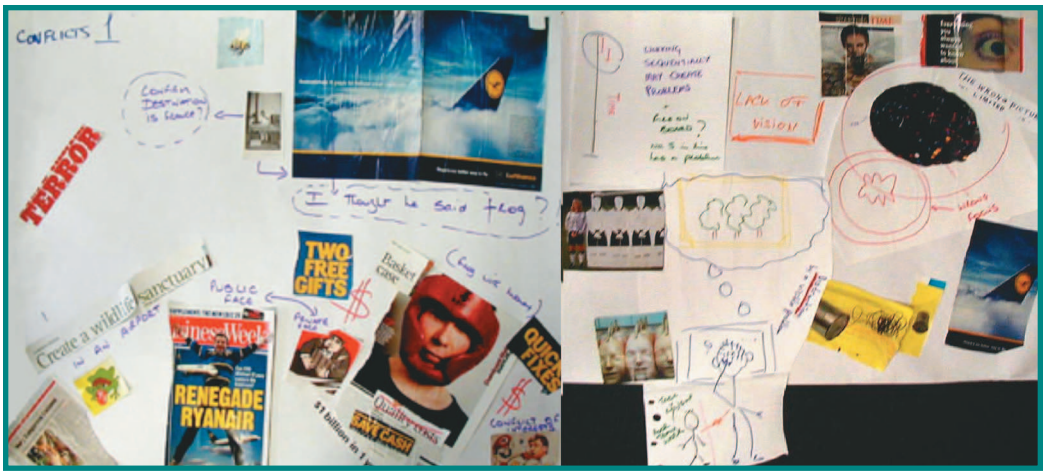

Figure 1. Aircraft conflict storyboards that workshop participants generated.

cilitator then encouraged participants to explore these analogical mappings with the rest of the group. Some analogies were obvious—the musical piece maps to the conflict resolution—whereas others were less so—the style of the piece maps to the resolution strategy.

The facilitators then organized an illumination period. During illumination, the facilitators encouraged participants to consider one mapping at a time, transfer the mapped concept's attributes (for example, a fabric's texture or a musical piece's structure) to the ATM domain, then generate new CORA-2 requirements and ideas from these elements. Consider the following example from the first workshop. Participants reported that one textile design was elegant—in other words, simple, beautiful, and symmetrical. They then transferred these attributes to the ATM domain to generate the requirement for elegant resolution strategies—that is, strategies that are simple (minimum maneuvers) and give pride to the controllers who implement them. Additionally, the participants perceived elegance differently. Again, participants transferred this attribute—subjective perception of elegance by the human agent—to the ATM domain to generate the requirement to accommodate different controller styles during conflict resolution.

Combinatorial creativity in the second and third workshops

We encouraged combinatorial creative thinking once participants had generated many new ideas. In the second workshop, the head of talent at toy manufacturer Lego randomly introduced items into conflict resolution scenarios to provoke creative thinking. We divided participants into groups of three, then asked them to develop worst-case aircraft conflict scenarios. Every 10 minutes, each group randomly received a new object (for example, a toy frog, binoculars, perfume, or a flashlight) to include in the scenario. The results were unusual scenarios that participants represented as collages (see Figure 1) and used during illumination to generate ideas about how CORA-2 would handle these conflicts.

During the third workshop, we encouraged combinatorial creativity by inviting a fusion chef to discuss combining unusual ingredients, share ingredients for tasting, and demonstrate fusion cooking with a lunch that reflected different but complementing ingredients. During illumination, the facilitators encouraged participants to investigate pairs of existing CORA-2 requirements and ideas to create new ones from unforeseen combinations and write them on RESCUE idea cards. For example, participants combined two original ideas from the second workshop—that "air traffic controllers should maintain an accurate mental model of the air space" and that "CORA-2 shall offer new types of situational display to air traffic controllers"—to generate a new requirement. That is, "CORA-2 shall allow air traffic controllers to rewind and fast-forward aircraft movements to develop their mental models of the air space before taking responsibilities for decisions that they will make." In another example, two ideas from the first and second workshops—that "manufacturers equip aircraft with the Airborne Separation Assistance System" and that "pilots in the cockpit have the same notion of conflict as do air traffic controllers in the tower"—led to the idea that

Table 2

Workshop design ideas and constraints

Workshop	Number of ideas	Number of constraints	Valued ideas	Very valued ideas	Very-very valued ideas	Ideas relevant to CORA-2	Ideas relevant to CORA-3	Ideas relevant to CORA+
1st	20	58	N/A	N/A	N/A	N/A	N/A	N/A
2nd	115	0	20	12	5	N/A	N/A	N/A
3rd	18	0	3	6	5	11	3	1
4th	46	0	5	24	7	44	2	0

"air traffic controllers shall use data link to send information about and hence the delegation of a resolution to the pilot."

Transformational creativity in the second and third workshops

The facilitators encouraged participants to change the CORA-2 solution space to make possible ideas that participants once considered impossible. During incubation periods, participants listened to presentations from domain experts with closer associations to ATM. An information visualization expert presented new solutions for displaying complex information at workstations, and a systems engineer explained how to adopt state-of-the-art approaches to complex systems engineering. The participants then worked in small groups with expert-provided candidate solutions to generate new solutions. Examples included controller displays that were blank until the future CORA-2 system detected conflicts and that allowed tactile aircraft manipulation on the display screens. Although we couldn't implement all the ideas in CORA-2, their essence helped participants discover several less radical requirements and designs, such as "transferring knowledge on cockpit design to the design of the controller working position."

Other creative activities

Each workshop started with activities intended to establish an environment more conducive to creative thinking, including

- Explanations of why creativity is difficult, so participants knew the challenges they faced
- Balloons to encourage participants to play and interact
- Shouting sessions to remove inhibitions and promote teamwork
- Relaxing music
- Lunchtime exercises to encourage participants to see creative thinking as continuous

We also applied additional techniques to support different creative processes, including

- Having participants remove constraints from ideas to open up the candidate requirements space
- Using the explorer, artist, judge, and warrior roles to focus the participants on different creative processes[3]
- Playing each other's controller, pilot, and manager roles to generate ideas from unencumbered perspectives
- Making all ideas visible to participants by posting them on cards
- Swapping ideas between groups to encourage combinatorial creativity

In a fourth CORA-2 workshop, participants selected and ranked ideas as relevant to the CORA-2 system, its successor CORA-3 system, or future (CORA+), as yet unspecified systems. We used other categories such as *valued*, *very valued*, and *very-very valued* to promote playfulness when exploring the ideas.

The workshops' effectiveness

The first three workshops generated 201 new ideas for CORA-2. Table 2 shows the number of new ideas generated during each workshop. The first workshop has a low total because the facilitators also spent time discovering important constraints and scoping CORA-2. The second workshop generated most of the ideas. The third workshop produced fewer new ideas, but participants generated 46 more ideas during a follow-up fourth workshop.

The authors of these 201 ideas analyzed them retrospectively and determined that they could only describe 54 of them as stakeholder

requirements—desirable, solution-independent properties of the future system. The remaining 147 encapsulated differing degrees of design and process knowledge. Nonetheless, the 201 requirements and design ideas provided a baseline for writing more precise use cases and generating more precise scenarios that, in turn, enabled more effective requirements acquisition and specification using ART-SCENE, a software environment that automatically generates scenarios from use case specifications, then guides stakeholders to walk through these scenarios to discover requirements.[4]

Key lessons learned

We encountered some problems during the workshops and report seven of the more important ones here, as well as ways to overcome them in future projects.

Provide step-by-step guidance for analogical reasoning

Workshop participants found it difficult to exploit the analogies. They only generated 20 ideas in the first workshop. In hindsight, this shouldn't have surprised us—studies from cognitive science reveal that analogical reasoning with unfamiliar domain classes is difficult without prior learning.[5] What was disappointing was that our facilitation wasn't more successful.

To encourage analogical learning during incubation, you need to explain analogies to participants with simple examples that encourage them to learn the important underlying abstractions—for example, the generic compose-and-test process in music composition and conflict resolution. This takes time, which leads to the next lesson.

Leave time to illuminate analogical ideas

We designed longer time periods for idea incubation and shorter periods for subsequent idea illumination. The problem was that illumination activities rarely lasted more than 30 minutes, which wasn't long enough to illuminate many analogical ideas.

The solution is to treat creative requirements engineering as a learning process, allowing time for learning to occur. Design each workshop to last at least two days, and encourage participants to incubate ideas both inside and outside scheduled workshop times. Adopt more flexible planning to respond to unanticipated threads of creative thinking that take on their own momentum—this was often when the most effective creative thinking occurred. Avoid dynamic short-term exercises because they cut dynamic creative thinking short. Instead, facilitators should be able to apply a range of creativity tasks in response to threads of creative thinking. One option is to select tasks on the basis of a categorization using Roger van Oech's[3] creative roles or Margaret Boden's[6] creative processes.

Interleave creative processes more effectively

Separating exploratory, combinatorial, and transformational creativity processes and techniques in different workshops left some ideas underdeveloped.

To avoid this, structure workshops around ideas rather than processes. Interleave techniques for combinatorial and transformational creative thinking to explore single ideas to their conclusions, marking spin-off ideas that you'll return to later. Encourage participants to explore ideas depth-first until they either reject the idea or develop concrete requirements. Structure pending ideas using visible idea lists and frequent prioritization and reprioritization of established ideas as the set of ideas evolves.

Structuring workshops around ideas requires more flexible use of domain experts to fulfill different creativity roles. Other experts we considered included fashion designers (exploratory), DJs, cocktail bartenders, cartoonists (combinatorial), science writers, and graphic artists (transformational). Build up a network of experts and categorize their contributions according to the domain knowledge they can deliver and the types of creativity they might support. Problem domain categories reported by the NATURE research project[7] provide a faceted classification scheme of domain expertise that will let you quickly identify experts.

Report back the rationale behind ideas

Because idea incubation often defied linear thought and idea generation was in small groups, the rationale behind ideas was sometimes lost to the wider requirements process.

We found it helps to hold frequent report-back sessions across groups to elicit rationale and communicate it to other participants. Have a scribe who is independent of the re-

Structuring workshops around ideas requires more flexible use of domain experts to fulfill different creativity roles.

In the long term, we aim to integrate and embed creativity techniques more effectively into structured requirements beyond workshops.

quirements process record idea rationale using argumentation techniques.[8]

Let people let off steam

Participants aren't always ready to be creative. They can bring problems, political issues, and other baggage that can inhibit creative thinking, as happened in the first workshop.

You need to give participants one or more periods to let off steam before being creative—it's only natural that people need to vent their frustration when faced with change. This happened, somewhat by accident, in our workshops. The first workshop gave people the chance to let off steam, which helped participants produce more creative requirements and ideas during the second one.

Plan, plan, plan the workshops

Scheduling creative activities can be difficult because it's hard to predict where the creative process will go. However, you must plan when experts will be involved to ensure their participation.

A workshop that appears over-planned to participants is less likely to encourage creativity. Instead, workshops should appear flexible, fluid, and responsive. However, facilitators should have plans they can invoke to handle most situations.

Find a champion for the workshops

Setting up the workshop was a challenge because creativity workshops aren't cheap to run, and the benefits of having domain experts present weren't always obvious.

It helps to find people who will champion the workshops. We attribute the CORA-2 success to managers and stakeholders who did just this. The CORA-2 program manager backed our unusual ideas, and the workshops themselves benefited from key stakeholders who embraced and supported the creative opportunities they received.

Integrating creativity workshops into structured processes

Although the workshops delivered outputs that were useful to the CORA-2 project (the resulting CORA-2 specification was reviewed, accepted, and is currently helping provide a CORA-2 system prototype), we identified problems and applied the lessons learned to subsequent workshops. One overriding challenge is to integrate creativity workshops into structured requirements processes that provide inputs into the workshops and use outputs from them. In RESCUE, we now run the creativity workshops early in the requirements process, after we establish the system boundaries but before specifying key design concepts and use cases. Inputs to each workshop are context and use case models and informal use-case précis that we later modify and extend according to workshop outcomes. Establishing first-cut system services within boundaries lets us maintain focus during the workshop and deliver more useful outputs. These outputs—requirements and design ideas—then let the team model system goals and write precise use case specifications that, in turn, they use to acquire more detailed requirements using ART-SCENE scenario walkthroughs.[4]

In the long term, we aim to integrate and embed creativity techniques more effectively into structured requirements beyond workshops. To do this, we'll categorize existing requirements techniques (for example, use case analysis or functional decomposition) according to their support for different creative processes and techniques. However, current creativity theories from cognitive and social psychology and artificial intelligence such as those we've cited are insufficient. We had to interpret and apply them with care to design the three workshops. The software development community needs new and more applied creativity models that will give software engineers the right facilitation skills and guidelines for technique selection.

So how should we develop these applied creativity models? We believe these new models must be informed by requirements artifacts and representations, such as use-case précis in text form and multimedia storyboards, which we use in structured requirements processes as important communication tools. To this end, we're currently developing applied models that will associate different artifact properties (for example, interactive scenarios) with creative processes (exploratory) and techniques that these properties support (for example, random idea generation with what-if questions) based on as-

sumptions about the creative activities and environment. We anticipate that such an applied model will prescribe creative activities within structured requirements processes, using existing cognitive and psychology models[6] to describe and explain the creative processes. At a more fine-grained level, this model will also help creativity workshop facilitators choose the right techniques to support and interleave creativity processes.

We hope that the reported experience and lessons guide readers to make their own requirements activities more creative. We've exploited these lessons learned both to run more creativity workshops[9] and to design tutorials and workshops to help others do the same.

Acknowledgments

We thank Eurocontrol and all workshop participants.

References

1. B. Nuseibeh and S. Easterbrook, "Requirements Engineering: A Roadmap," *The Future of Software Engineering*, companion volume to *Proc. IEEE Int'l Conf. Software Eng.* (ICSE 2000), ACM Press, 2000.
2. R.J. Sternberg and T. Lubart, *Defying the Crowd: Cultivating Creativity in a Culture of Conformity*, Free Press, 1995.
3. R. van Oech, *A Kick in the Seat of the Pants: Using Your Explorer, Artist, Judge and Warrior to be More Creative*, Harper and Row, 1986.
4. A. Mavin and N.A.M. Maiden, "Determining Socio-Technical Systems Requirements: Experiences with Generating and Walking Through Scenarios," *Proc. 11th Int'l Conf. Requirements Eng.*, IEEE CS Press, 2003, pp. 213–222.
5. M.L. Gick and K.J. Holyoak, "Schema Induction and Analogical Transfer," *Cognitive Psychology*, vol. 15, no. 1, 1983, pp. 1–38.
6. M.A. Boden, *The Creative Mind*, Abacus, 1990.
7. A.G. Sutcliffe and N.A.M. Maiden, "The Domain Theory for Requirements Engineering," *IEEE Trans. Software Eng.*, vol. 24, no. 3, 1998, pp. 174–196.
8. T.P. Moran and J.M. Carroll, *Design Rationale: Concepts, Techniques, and Use*, Lawrence Erlbaum Associates, 1996.
9. L. Pennell and N.A.M Maiden, "Creating Requirements—Techniques and Experiences in the Policing Domain," *Proc. 9th Int'l Workshop Requirements Eng.: Foundation for Software Quality* (REFSQ 03), Essener Informatik Beitrage, pp. 27–35.

About the Authors

Neil Maiden is a professor of systems engineering and head of the Centre for Human-Computer Interaction Design at City University, London. His research interests include requirements engineering and scenario-driven approaches to software development. He received his PhD in computer science from City University. He is an associate member of the British Computer Society and cofounder of its Requirements Engineering Specialist Group. Contact him at the Centre for HCI Design, City Univ., Northampton Square, London EC1V OHB, UK; n.a.m.maiden@city.ac.uk.

Suzanne Robertson is a cofounder of the Atlantic Systems Guild, a think-tank for system development techniques, and has more than 30 years' experience in systems specification and building. Her current work includes research and consulting on stakeholders' rights and responsibilities, the specification and reuse of requirements, and techniques for assessing requirement specifications. She is a member of the IEEE and the Australian Computer Society. She is editor of *IEEE Software*'s Requirements column. Contact her at the Atlantic Systems Guild, 11 St. Mary's Terrace, London W2 1SU, UK; suzanne@systemsguild.com.

Alexis Gizikis is a research student at the Centre for HCI Design at City University. His research interests lie at the intersection of systems engineering and creativity. He received his interdisciplinary MEng in mechanical and electrical engineering from the University of Aberdeen. Contact him at the Centre for HCI Design, City Univ., Northampton Square, London EC1V OHB, UK; alexis@soi.city.ac.uk.