Requirements Engineering for E-science: Experiences in Epidemiology

Sarah Thew, Alistair Sutcliffe, Rob Procter, Oscar de Bruijn, John McNaught, and Colin C. Venters, *University of Manchester*

Iain Buchan, *Northwest Institute for BioHealth Informatics*

The Advises project is developing innovative tools that support geographic visualization in epidemiology and public-health decision making by exploring and combining requirements engineering techniques.

Epidemiologists investigate the distribution and determinants of diseases and other health-related states. As with a growing number of disciplines, epidemiology is undergoing a transformation in its research methods. Researchers have already identified many of the "simple" epidemiological associations; now they need methods that can unravel a complex web of factors such as genetics, environment, and lifestyle.[1]

An emergent approach to investigating complex conditions is to use data-driven hypothesis generation methods as a complement to hypothesis-driven experimentation.[2] This approach is part of a wider transformation of research methods known as *e-science* in the UK (or cyberinfrastructure in the US). Basically, e-science is shorthand for a cluster of digitally based methodological innovations driven by the demands of scientific challenges and the data deluge.[3] Despite the critical role of computer-based tools in e-science, requirements analysis and usability engineering have received little attention.[4] (Notable exceptions are research that described a use case and a goals-questions-results requirements analysis method[5] and research that applied ethnographic methods to understand researchers' work practices.[6])

When giving detailed requirements up front is impossible, the requirements analysis process must sustain a continuous dialogue that helps users and developers find common ground. Developers can then gradually refine requirements as users come to understand both their needs and what new technologies can offer. This challenge isn't unique to scientific software development but has been identified as particularly significant for e-science.[1,4]

With the Advises (Adaptive Visualization of E-science) project, we're attempting to develop a deep understanding of researchers' work practices by combining various requirements engineering (RE) techniques in a strongly iterative and agile approach. Specifically, this project is developing tools to support geographic visualization in data-driven epidemiological research. Our experiences have been promising and suggest that the approach can apply generally to projects developing software that lets users exploit the potential of data-driven research.

Toward Increasing Data Exploration

Geography is often a factor in epidemiological investigations, whether they are comparing the distribution of a disease within two populations or considering ease of access to health services across a region. Despite this, epidemiologists haven't fully exploited GIS (geographic information system) tools owing to their complexity.[7]

 0740-7459/09/$25.00

The tools that we're developing in the Advises project will help epidemiologists quickly and easily generate and explore public health maps using statistics that can help them establish the significance of observed patterns. The epidemiologist users are often members of small, colocated groups; they are computer literate and technical experts and are used to handling large data sets and writing statistical-analysis software.

Epidemiology has seen only limited adoption of data-driven research methods. The epidemiologists involved in Advises report a lack of tools to support data exploration. They also recognize that data exploration is unfamiliar to most epidemiology researchers, who tend to develop a detailed hypothesis based on existing knowledge and only then test it on data. Consequently, the RE challenge for Advises is to devise an approach that will facilitate both the understanding of current research methods and the emergence of new ones.

Requirements Approach

The Advises approach to requirements analysis (see Figure 1) is grounded in scenario-based design[8] and a combination of user-centered RE techniques[9] applied within a framework of iterative cycles of requirements elicitation, design exploration, and user feedback.[10] We chose scenario-based design as a way to tackle the potential volatility and complexity of e-science requirements because of its strengths in facilitating user-developer communication.

Our overall RE process was driven by the research questions epidemiologists ask and the language they use. This follows from a key orientation of the project, which sees requirements as research questions and builds on the goals-questions-results method.[5] We aimed to use the epidemiologists' research questions as a source of scenarios and use cases and then imagine a new system to support analysis of these questions. Users' language also contributed to the development of an ontology describing epidemiological research. We use this to support data upload and management and in the query interface design. Consequently, several techniques we used focused on exploring users' language and identifying research questions.

Particular concerns for the RE process were the need to support innovation in scientific work practices and to understand the expertise underlying it, some of which might be tacit and consequently resistant to articulation. The two issues are intertwined: to support new work practices, we needed to understand that current approaches and tacit expertise are, by nature, not always easily observed.

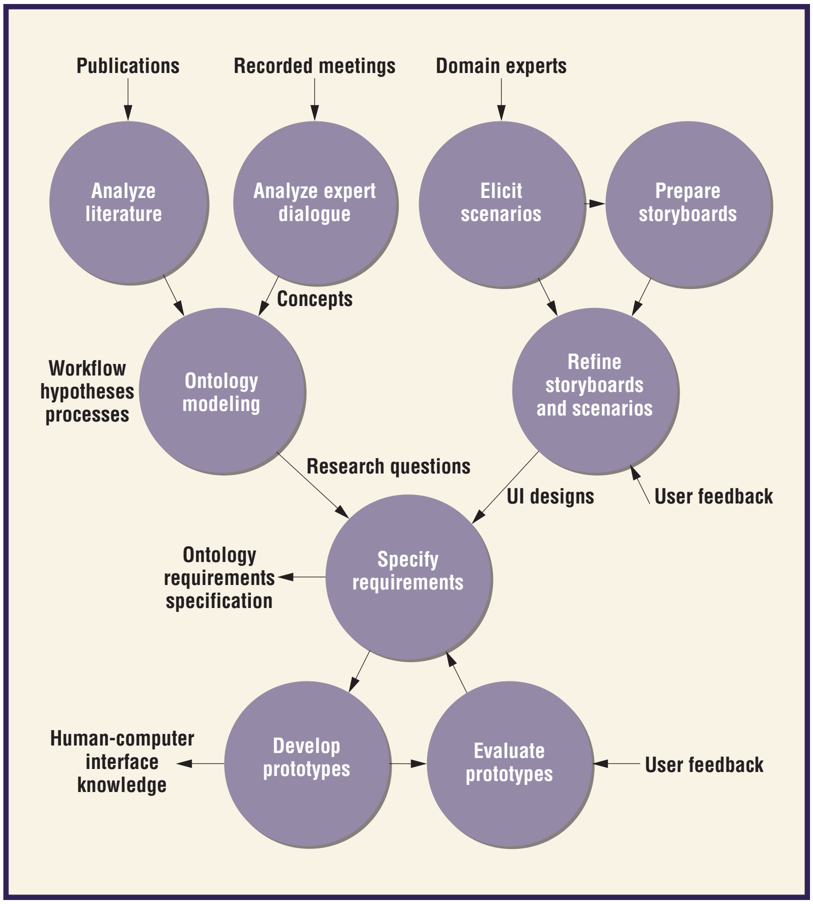

Figure 1. The Advises (Adaptive Visualization of E-science) requirements elicitation strategy. The iterative development of scenarios, storyboards, and prototypes helps users explore the potential of the proposed system and supports communication between users and developers.

We experimented with several RE techniques and evaluated their effectiveness, particularly with regard to tacit knowledge articulation, process, and sublanguage analysis.

Unstructured Interviews

At the project's start, we conducted 22 unstructured interviews (each lasting 1 to 2 hours) to gain background on working practices, user preferences, and domain norms and to discuss designs. Because we conducted the interviews on site, the epidemiologists could show us their preferred software, discuss their data management practices, and show us example data sets.

The interviews gave us the opportunity to explore the epidemiologists' perceptions of data-driven analysis and understand their interest in and concerns about modifying their working practices. In particular, these early interviews provided key insights into epidemiologists' attitudes that influenced the project direction.

Observing users' working practices is a powerful technique for both gaining deeper domain knowledge and understanding expertise.

For example, an early exploratory interview discussed the software tools used in epidemiological research. A question about image- and graph-creation software led to a discussion about the epidemiologists' preference for numbers over images. They feel some images can be ambiguous and open to misinterpretation, so they prefer to see numbers in epidemiological publications.

Also, routine data collection and warehousing are becoming the norm at the UK's National Health Service and Department of Health. During the interviews, we discovered that our users often have access to this data and use it in hypothesis testing but that they're also interested in data-driven hypothesis generation and pattern identification.

Consequently, we decided that interactive querying and visualization at an early stage of analysis was likely an acceptable area in which to include visualization software.

Research has shown that interviews are a poor method for accessing tacit knowledge;[9] our experience confirmed this. They provided a good broad-brush view of epidemiology and its working practices, but they didn't help us effectively explore the subtleties of the analysis process.

User Observation

Observing users' working practices[6] is a powerful technique for both gaining deeper domain knowledge and understanding expertise.[11] We observed the epidemiologists as they carried out their research and in their weekly progress meetings.

The epidemiologists held regular meetings reviewing current and upcoming work; our requirements analyst attended seven such meetings (each lasting 1 to 2 hours) to listen and record discussions but did not contribute. This "fly on the wall" observation technique was an effective approach for gathering background knowledge and understanding epidemiological language. These meetings were also a good source of research questions and an opportunity to listen to the epidemiologists discuss ideas for new research.

In the following exchange, for example, two epidemiologists are discussing ways to model a research question:

E1. I was going to do baseline and change, but a statistician who got interested in this disagreed.

E2. No.

E1. For saying baseline and change?

E2. Because my contention, for example, if you have Body Mass Index [BMI] there, and weight change, it might be relevant for your baseline BMI, but it might depend on whether you're fat and you lose weight, and therefore that's worse. So, biologically, there is an importance to your baseline level.

E1. Just thinking ... If you've got baseline and next as the measurement, rather than baseline and change in the variance structure of the model, isn't then the coefficient associated with the next measurement? Next, independent of baseline?

E2. We don't know it. Because this is an unnatural situation, you don't know whether they're correlated, and you don't know whether the next cholesterol level is dependent on your baseline. They might be totally independent, irrelevant to each other, but it depends what the data is like.

E1. I agree with you, in part, but I don't think we're losing anything by putting change and baseline. It doesn't add anything mathematically, but to me, in terms of interpretation, it adds meaning.

Listening to such exchanges helped us understand the epidemiologists' research questions and models and the data they need to answer these questions. They were also a rich source of epidemiological language.

The epidemiologists' work is essentially desk based—preparing and analyzing data—and is carried out individually rather than collaboratively. To understand how the epidemiologists make decisions that guide the execution of their research, we encouraged them to talk aloud as they worked. For this purpose, during four separate sessions (each lasting 1 to 4 hours), the requirements analyst observed, prompted, and questioned them.

The approach was useful for understanding the intricacies and variety of epidemiological data and appreciating how the epidemiologists' software tools interact. The technique proved less effective for exposing the epidemiologists' thought processes; they found it difficult to articulate this knowledge while carrying out the task at hand. So, although we firmly believe that observing working practices is vital to understand the working environment and task flow, it wasn't entirely effective in this context.

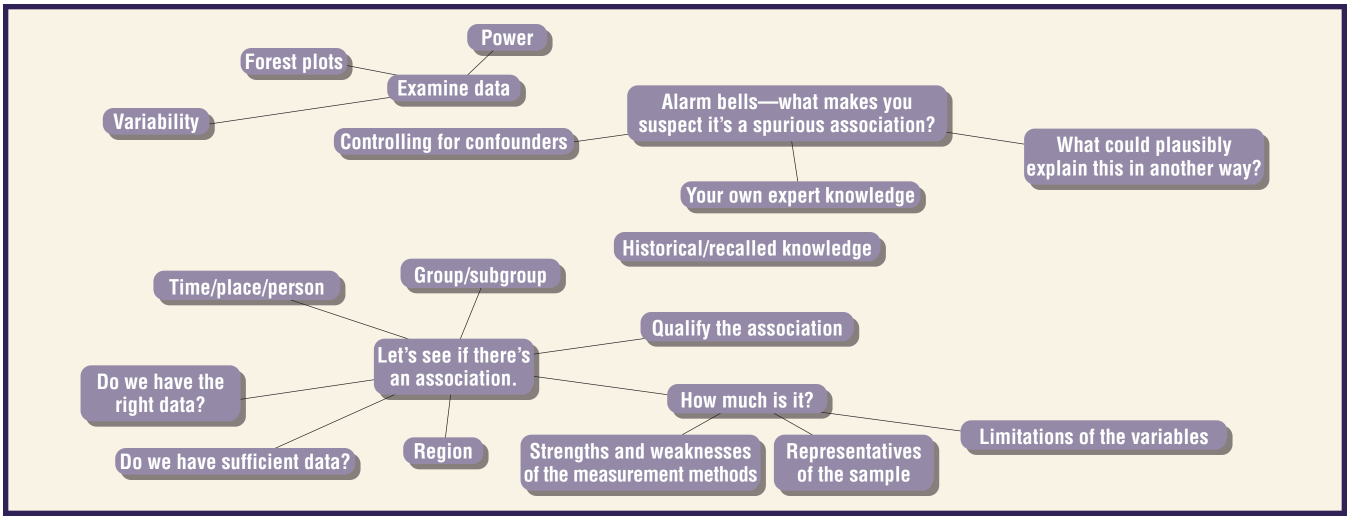

Figure 2. A mind map of the issues epidemiologists consider when deciding whether to investigate a hypothesis. Maps were drawn on large shared sheets of paper during the course of the workshop in order to facilitate and document our discussions.

Domain Knowledge Workshops

The ontology-building process also included four workshops (each lasting 1.5 to 2 hours) during which we asked the epidemiologists to build models of various aspects of the domain, such as developing a research question or assessing evidence for or against a model of causality. They presented models as networks of sticky notes, which they could rearrange and extend. The analyst facilitated these workshops, and we made a video recording of them for subsequent analysis.

The workshops provided a good opportunity for the epidemiologists to articulate tacit dimensions of their expertise. For example, discussions that ensued about the fundamental concepts of epidemiology and work practices were extremely useful in understanding how epidemiologists reason about their data, such as how they assess evidence of causation, or their decisions about a particular data set's reliability. This learning fed into the ontology and was also useful in system design. For example, Figure 2 reproduces part of a mind map considering the evidence used to decide whether an epidemiological phenomenon is real or an artifact of a data set. We subsequently used this when deciding which statistics the system should display alongside a map.

Scenarios

We used three scenario development meetings (each lasting 1 to 2 hours) to explore with the epidemiologists ways the system could support their research questions. The analyst generated scenarios from research questions identified from the workshop data and, with the users, imagined how the system could support data investigation and exploration.

For example, one research question was, "What are the characteristics of the general practitioner (GP) registered population in the UK northwest?" To answer this query, an epidemiologist could explore a map of patients registered to primary care trusts (the organizing bodies for local healthcare in the UK) in the northwest, stratifying the population by location, gender, and ethnicity.

Another question was, "Is there an association between weather conditions, levels of electrostatically charged particles, and the number of people reporting runny noses?" To answer such a complex question, the system would need to support mapping of cases of runny noses, weather conditions, wind direction, and power-line locations over six months.

Scenario creation didn't come naturally to the epidemiologists; they initially constrained themselves to simple research questions and had difficulty thinking of questions that would stretch the system requirements. This is understandable given the difficulties of imagining the potential of software that doesn't exist yet and that might change their work practices. To encourage the users to think more broadly, the requirements analyst suggested straw-man examples based on background domain knowledge, which the epidemiologists were able to respond to and modify.

Although scenario creation was initially challenging, it proved to be a good way to communicate complex problems and requirements back to the development team. Scenarios were also an effective way to explore the processes epidemiologists use to

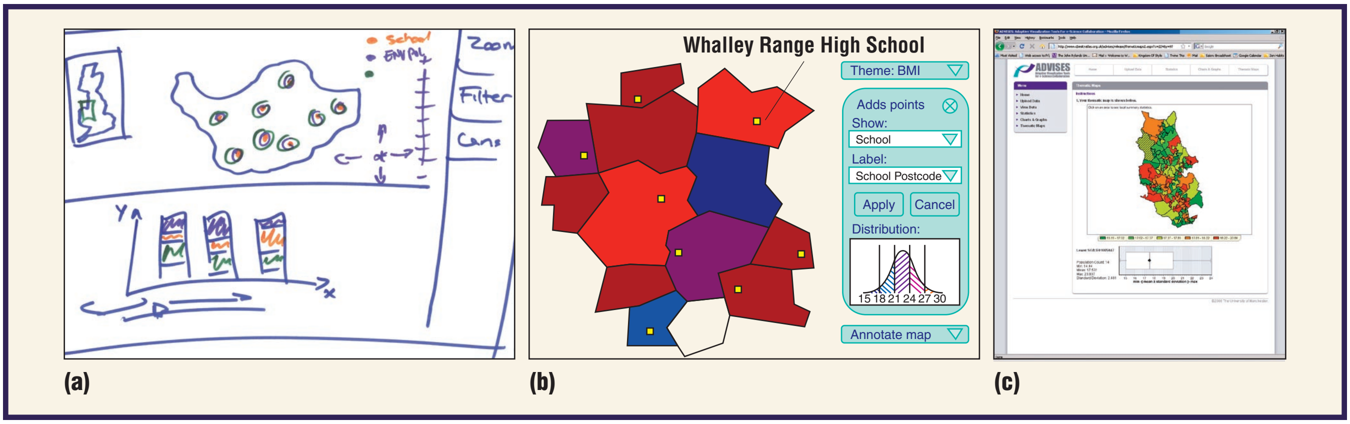

Figure 3. Real-world use of storyboards. The interface design progressed from (a) a rough sketch to (b) a PowerPoint slide to (c) a working prototype.

evaluate a research question. In addition, they provided another method to access tacit expertise—for example, what additional data might be needed alongside a map, and what signs raise suspicions that an observed pattern isn't real? Furthermore, scenarios helped enable innovations, letting the requirements analyst move the discussion away from current work practices to help users imagine new approaches.

Storyboards and Prototypes

We used several pictorial approaches to requirements exploration, from simple, single-cell storyboards exploring initial ideas to paper and Web-based prototypes. These pictorial methods were highly effective for

- communicating ideas from the design team to the users,
- engaging users in the design process, and
- gathering feedback and stimulating debate.

The various prototypes encouraged user involvement in decision making. Users could imagine the steps they would work through and the accompanying statistics they would need to interpret the map. Figure 3 shows a storyboard's progression from a pencil sketch to a PowerPoint slide to a functioning prototype. The sketch shows a distribution graph split into differently colored quintiles; regions of the map are colored according to the quintile the mean value for that region corresponds to. This feature was originally suggested by a user, who then responded to this storyboard by elaborating on the original idea, requesting movable quintile boundaries, with the map colors updating as the boundaries change.

The storyboards supported the requirements development by providing realistic, easily modified illustrations and enabling idea development. By making the new software's potential more concrete, they helped users investigate new working practices. This made it easier for users to understand how the software could empower early-stage research by providing new ways for them to explore data and look for interesting patterns.

Combining RE Techniques

Throughout the requirements process, we looked at the various techniques as a toolkit. Rather than follow a rigid methodology, we devised an agile approach that let us adapt evolving project goals and circumstances.

Table 1 summarizes the appropriateness of each technique we used for identifying the knowledge we needed to drive the RE process forward. This table isn't intended to provide a definitive comparison of techniques; it simply illustrates the different benefits each technique provided in particular contexts.

Early in the project, a combination of interviews, meeting observations, and work observations provided rich domain learning and generated ideas that started to shape the project. However, although interviews and work observations taught us about the more concrete and observable aspects of work, these methods were less effective for articulating tacit expertise about epidemiological workflows and decision making. We found that scenarios and the domain knowledge workshops particularly helped us address this gap and understand how our users consider evidence and make decisions about their data. Once we began early design work, a combination of scenarios and storyboards worked well. Scenarios proved particularly effective for feeding users' requirements to the project team, whereas storyboards and prototypes helped us explore designs with users.

Table 1
The utility of different RE techniques

	Interviews	Meeting observations	Work observations	Domain knowledge workshops	Scenarios	Storyboards
Broad domain knowledge	Good	Good	Good	Some	Some	Minimal
Eliciting expertise	Some	Some	Some	Good	Good	Some
Identification of research questions	Some	Good	Minimal	Some	Good	Some
Ontology development	Good	Good	Minimal	Good	Good	Minimal
Design exploration	Some	Minimal	Minimal	Minimal	Good	Good
Requirements specification	Good	Minimal	Minimal	Minimal	Good	Good
Developing new work practices	Some	Minimal	Minimal	Some	Good	Good

When selecting which requirements-gathering techniques to use, we needed to consider how acceptable each was to users. Open-ended interviews were a good way to initiate the requirements process because users widely understood the concept. Interviews let analysts and users get to know each other in relatively familiar circumstances. The epidemiologists were also comfortable with the presence of an observer/analyst in meetings, so this also proved an effective way to get background knowledge.

In principle, one major advantage of observation-based requirements-gathering techniques is that they don't make significant demands on busy users or disrupt their work. However, our experience was that observation sessions often turned into a dialogue between the requirements analyst and epidemiologist, so this advantage wasn't realized in practice. So, scenarios became our preferred technique for getting to the heart of the analysis process. Although the scenario sessions required the epidemiologists to take time from their research, the epidemiologists could decide when this was convenient.

When we first introduced the use of scenarios, the epidemiologists found the approach abstract and had difficulty contriving situations that weren't grounded in current work. However, as they became more familiar with the method, they became highly creative in thinking about how they would like the system to support investigation of complex research questions.

The domain knowledge workshops placed the epidemiologists in an unusual situation, asking them to discuss aspects of their world they take for granted. To make this task easier, we began by asking them to model some of the more concrete concepts they work with, such as the types of epidemiological studies. As they became more used to the task, we moved to more abstract questions. This approach worked well, and the epidemiologists commented that they found the workshops interesting and engaging. However, as with scenarios, this approach is time consuming; simply getting participants together for two-hour meetings was difficult.

Translating Requirements Analysis into Design Decisions

The requirements analysis raised several issues for the system architecture. One issue was, to what extent should the system interpret the epidemiologists' research questions? We rejected free-format natural-language input to avoid the complexities of natural-language processing; instead, we provided a series of keyword selection lists ordered in question templates. The interviews, scenarios, and domain analysis workshops identified three high-level question types in epidemiology:

- pattern identification (for example, "What's the distribution of diabetes in population *X*?"),
- association-causation ("Is there a link between asthma and obesity?"), and
- comparisons between populations and over time and space.

So, the query interface works by semiautomatic configuration of menu-based lists of terms driven by the variables in the chosen data set. For example, for the query "Is there a link between asthma and obesity?" the user can select **association**, **asthma**, and **obesity**. However, complexity arises in the large space of possible associative combinations between variables, in exposing population structure variation, and in eliminating confounding effects. For instance, answering more detailed questions (such

Providing a fully configurable system would be a wasted effort if users can't find time to tailor it.

as, "Is there a link between smoking and obesity once you eliminate variations in age, exercise, and other illnesses?") requires that epidemiologists invoke complex statistical analyses.

The requirement for processing complex questions came from the domain knowledge workshops; it led us to develop interpreters for high-level-concept keywords in queries. For example, it was clear from scenario development and observation of meetings that the epidemiologists often consider deprivation as a factor in their investigations. In population data sets, deprivation might be indicated by income range or socioeconomic class (in census data) or by housing status and the local environment (in a GIS). To execute a query involving deprivation, the query processor must read the metadata from the selected data sets and then interpret the best fit for "deprivation." If several measures of deprivation are available, the system must ask the user to narrow the options.

Other examples of interpreters are related to terms in spatial queries ("near to," "close by," or "in the region of") and temporal queries ("recent increase in obesity"). The system must resolve spatial queries in terms of proximity by a preset distance or colocation in an area boundary.

To provide a flexible, user-customizable solution for the requirements, we developed configuration editors. These let epidemiologists set their default interpretation of complex query terms, such as "nearby" in spatial queries. They also provided other functions, such as defining terms for annotating results, data validation tolerances, and rules to warn about potentially invalid analysis treatments.

However, providing a fully configurable system would be a wasted effort if users can't find time to tailor it. In interviews about use of existing software, the epidemiologists acknowledged they don't always have time to fully investigate or learn new systems. Because most users don't use the customization facilities in many common software packages,[10] we doubted that extensive configuration facilities would be a good investment. Consequently, we provided a set of analysis, querying, and visualization services that can be integrated in their current applications.

A final issue for the architecture concerned the degree of intelligence and system-versus-user initiative in the visualization and statistics wizard modules. The visualization expert module was easier to design. It must select appropriate ways to code data in maps and graphical displays according to the user's research questions and metadata—for example, by using texture, shading, and color coding so that users can scan a map to detect patterns of association between obesity and social deprivation by area.

The statistics wizard should warn users when their analyses are potentially invalid because the data sets they wish to compare are too sparse. We explored three design options for this wizard:

- a menu of smoothing algorithms that can cure some sparse or irregular geographic-distribution problems,
- a critic module that warns users about potential analysis problems and suggests appropriate algorithms to choose from, or
- a fully automated solution that applies the appropriate algorithm without consultation.

Because the system's users are domain experts and end-user developers, we chose the critic module and gave them the capability to change the rule sets and algorithms.

The Advises project has closely coupled architecture development with requirements analysis in an iterative development cycle. Iteration allows requirements to emerge through a dialogue between the developers and users as designs are critiqued and refined, letting us design an extensible architecture providing for longer-term developments. We're engaged in a third iteration of this requirements-build-test cycle. We've developed a semifunctional Web-based prototype that lets users interact with simple, canned maps and evaluated it using formal usability assessments along with informal user feedback. With each journey around this loop, we've progressively added more functionality.

Key to the Advises requirements approach has been developing a strong working relationship with users and collaborating with them to explore new ways of working. One potential avenue for developing this relationship further is to collocate (or embed) developers in the users' workplace.[12] By co-opting users as full-time members of the development team,[13] embedding provides one way of continuing a tightly coupled, user-driven design and development process.

E-science presents a major challenge for the RE community. For example, Ann Zimmerman and Bonnie Nardi state that

> *We know of no scalable methods of requirements analysis that document the needs of vastly different user populations, continue to document changing needs over time, coordi-*

nate investigation at multiple sites of use, design for large distributed entities, and absorb transformative changes in practice.[14]

Our project hasn't had to deal with all these concerns. Nevertheless, we argue that the essence of our approach—an agile RE process, coupled with strong user engagement, that can adapt to changing circumstances as a project and users' needs unfold—is key to meeting this challenge.

Acknowledgments

The Advises project is funded by the UK Engineering and Physical Sciences Research Council under its E-science Usability program, grant number EP/D049989.

References

1. N. Pearce and F. Merletti, "Complexity, Simplicity, and Epidemiology," *Int'l J. Epidemiology*, vol. 35, no. 3, 2006, pp. 515–519.
2. D. Kell and S. Oliver, "Here Is the Evidence, Now What Is the Hypothesis?" *BioEssays*, vol. 26, no. 1, 2004, pp. 99–105.
3. T. Hey and A. Trefethen, "The Data Deluge: An E-science Perspective," *Grid Computing: Making the Global Infrastructure a Reality*, F. Berman, G. Fox, and A. Hey, eds., John Wiley & Sons, 2003, ch. 36.
4. B. Beckles, "User Requirements for UK E-science Grid Environments," *Proc. UK All-Hands E-science Meeting*, Nat'l E-science Centre, 2004; www.allhands.org.uk/2004/proceedings/papers/251.pdf.
5. V. Perrone et al., "Developing an Integrative Platform for Cancer Research: A Requirements Engineering Perspective," *Proc. UK All Hands E-science Meeting*, Nat'l E-science Centre, 2006; www.allhands.org.uk/2006/proceedings/papers/705.pdf.
6. M. Jirotka et al., "Collaboration and Trust in Healthcare Innovation: The E-diamond Case Study," *Computer Supported Cooperative Work*, vol. 14, no. 4, 2005, pp. 369–398.
7. A.C. Robinson, "A Design Framework for Exploratory Geovisualization in Epidemiology," *Information Visualization*, vol. 6, no. 3, 2007, pp. 197–214.
8. J. Carroll, *Making Use: Scenario-Based Design of Human-Computer Interactions*, MIT Press, 2000.
9. A. Sutcliffe, *User-Centered Requirements Engineering: Theory and Practice*, Springer, 2002.
10. G. Fischer, "User Modeling in Human-Computer Interaction," *User Modeling and User-Adapted Interaction*, vol. 11, nos. 1–2, 2001, pp. 65–86; http://l3d.cs.colorado.edu/~gerhard/papers/umuai2000.pdf.
11. G. Kotonya and I. Sommerville, *Requirements Engineering Processes and Techniques*, John Wiley & Sons, 1998.
12. M. Hartswood et al., "Co-realisation: Towards a Principled Synthesis of Ethnomethodology and Participatory Design," *Resources, Co-Evolution and Artefacts*, M. Ackerman et al., eds., Springer, 2008, pp. 59–94.
13. J. Schopf et al., *Report of the User Requirements and Web Based Access for E-research Workshop*, tech. report, Nat'l E-science Center, May 2006; www.nesc.ac.uk/talks/685/UKeS-2006-07.pdf.
14. A. Zimmerman and B. Nardi, "Whither or Whether HCI: Requirements Analysis for Multi-sited, Multi-user Cyberinfrastructures," *Conf. Human Factors in Computing Systems, CHI '06 Extended Abstracts on Human Factors in Computing Systems*, ACM Press, 2006, pp. 1601–1606.

About the Authors

Sarah Thew is a research associate at the University of Manchester. Her research interests include the sociotechnical aspects of requirements engineering, usability of scientific software, and bioinformatics. Thew received her MSc in computation from the University of Manchester Institute of Science and Technology. She is a member of the British Computer Society and the IEEE. Contact her at sarah.thew@manchester.ac.uk.

Alistair Sutcliffe is a professor of systems engineering in the University of Manchester's Manchester Business School. His research interests include human-computer interaction and software engineering, especially in interaction theory, design methods for multimedia, virtual reality, Web interfaces, usability evaluation methods, and the design of complex sociotechnical systems. Sutcliffe received his PhD in ethology from the University of Wales. Contact him at a.g.sutcliffe@man.ac.uk.

Rob Procter is a professor and the research director at the UK National Centre for e-Social Science at the University of Manchester. His research interests include sociotechnical issues in the design, implementation, evaluation, and use of interactive computer systems, with emphasis on ethnographic studies of work practices, computer-supported cooperative work, and participatory design. Procter received his PhD in electrical engineering from the University of Manchester's Institute of Science and Technology. Contact him at rob.procter@manchester.ac.uk.

Oscar de Bruijn is a lecturer in human-computer interaction and collaborative systems in the University of Manchester's Manchester Business School. His research interests revolve around how people perceive, assimilate, and use information in computer-supported tasks performed either individually or in collaboration with other people. He conducts projects in information visualization, visual data mining, and visual analytics. de Bruijn received his PhD in psychology from the University of St. Andrews. He's a member of the ACM and Interaction (formerly the British Human-Computer Interaction Group). Contact him at o.de-bruijn@mbs.ac.uk.

John McNaught is the associate director of the UK's National Centre for Text Mining and a lecturer in the University of Manchester's School of Computer Science. His research interests include text mining, natural language processing, computational lexicography, and sublanguage analysis. McNaught received his MA in general linguistics from the University of Manchester. Contact him at john.mcnaught@manchester.ac.uk.

Colin C. Venters is a research associate at the University of Manchester. His research interests include information visualization, image retrieval, pattern recognition, and machine vision. Venters received his PhD in computer science from the University of Manchester. Contact him at c.venters@ieee.org.

Iain Buchan is a clinical professor of public-health informatics and the director of the Northwest Institute for BioHealth Informatics at the University of Manchester. His research interests include epidemiological applications of health informatics; health informatics to enable prevention and early intervention; large-scale, naturalistic epidemiology; computational statistics; and obesity and metabolic health. Buchan received his Doctor of Medicine from the University of Liverpool. He's a member of the Royal Statistical Society, British Health Informatics Society, and UK Council of Health Informatics Professionals. Contact him at buchan@manchester.ac.uk.