

Experience with user-centred requirements engineering

Alistair Sutcliffe · Sarah Thew · Paul Jarvis

Received: 3 March 2010 / Accepted: 15 April 2011 / Published online: 8 May 2011
© Springer-Verlag London Limited 2011

Abstract This paper describes the application of human–computer interaction (HCI) principles and methods to requirements engineering in a case study development of a visualisation tool, ADVISES, to support epidemiological research. The development approach consisted of scenario-based design and analysis of the users’ tasks and mental model of the domain. Prototyping and storyboarding techniques were used to explore design options with users as well as specifying functionality for two versions of the software to meet the needs of novice and expert users. Application of HCI functional allocation heuristics to guide system requirements decisions is explained. An evaluation of the prototype was carried out to assess the extent to which the expert model would support public health professionals in their analysis activities. The results of the design exploration requirements analysis study are reported. The implications of scenario-based design exploration, functional allocation and software architecture are discussed.

Keywords Requirements elicitation · Scenarios · User-centred design · HCI · RE methods

1 Introduction

Human–computer interaction (HCI) and requirements engineering (RE) share many perspectives. However, there

have been few studies applying HCI or human factors knowledge to RE, although convergence between HCI and software engineering approaches has been explored in the study by Seffah et al. [43] and several ICSE workshops [23]. RE modelling languages such as *i** [61] do specify users’ goals, tasks and resources, and *i** has been extended to describe social relationships such as responsibility and trust [51]; furthermore, goal-oriented RE has been adapted to account for the influence arising from users’ characteristics, preferences and skills [18, 52], so human issues in requirements analysis have been partially addressed. Furthermore, agile development approaches encourage adoption of human values and user participation in small development teams [7]. Convergence between user-centred design from the HCI tradition [19] and RE has also been explored by Sutcliffe [50] and Lauesen [26], so it appears that the two disciplines should be moving towards closer integration. However, in spite of the possible convergence reported in the research literature, few reports of actual practice of HCI-inspired RE have emerged.

User interfaces (UI) and HCI are direct concerns of requirements engineering rather than a veneer of interactive components that adorn the software system. For example, many requirements for decision-support systems can only be considered in terms of user interaction, while functionalities of the user interface are first-order requirements that serve user goals, e.g., functional requirements to interactively explore information spaces, virtual worlds and social networks. This paper reports experience of exploring HCI influences on the RE process and design goals for specifying requirements and interactive functionality, with implications for the user interface and software architecture. A case study experience that applied HCI principles and methods to the ADVISES (ADaptive VISualisation for E-Science) project is described, which developed

A. Sutcliffe (✉)
Manchester Business School, University of Manchester,
Booth Street East, Manchester M15 6PB, UK
e-mail: ags@manchester.ac.uk

S. Thew · P. Jarvis
Northwest Institute for Bio-Health Informatics (NIBHI),
School of Medicine, University of Manchester, Manchester, UK

visualisation tools to support epidemiological research and public health decision-making.

To encourage epidemiologists to make more use of visualisation tools, the project focused on understanding the mental models epidemiologists use to make decisions about maps [54], while exploring the statistical properties underlying the graphical representations [55]. This can lead to problems as follows: on the one hand, ensuring that visual patterns correspond to meaningful structures in the data; while, on the other hand, being able to explain what those patterns mean. These gaps were referred to by Amar and Stasko [2] as the ‘rationale gap’ and the ‘world view gap’, respectively. These concepts can be considered as ‘HCI requirements’ or goals held by developer stakeholders, akin to non-functional requirements which could be expressed as ‘comprehensible displays’ and ‘transparent mapping of visualisations to models’. Such design goals are related to architectural requirements, for instance, distributed solutions and security which are inevitably intertwined with user requirements [35]. Patterns as reusable knowledge or generic designs have been proposed in both HCI [12, 57] and RE for a variety of areas ranging from privacy to enterprise models [3, 14, 40]. A further motivation for this paper was to apply HCI design patterns to address UI design requirements, i.e., the ‘gaps’ problems, to reflect on how UI requirements might be formulated so that pattern-like solutions could be adopted.

ADVISES had to satisfy the needs of two different user communities. Academic epidemiologists are domain experts and are often advanced computer users, able to develop their own applications for statistical analysis. However, public health professionals, who are rarely computer experts, need to process spatially coded health records for investigation and planning purposes. Hence, the project had to address both expert and novice user communities. In HCI, this problem is familiar, and either automatic adaptation or configuration of user interfaces to meet different user profiles is the accepted response [15, 27]. In RE, this problem might be construed in terms of stakeholder viewpoints and their reconciliation [24, 46]. Requirements analysis in conventional development practices, e.g., RUP/UML, assumes a use case-led approach [21], which tends to focus attention on user system interaction without analysing user activity in detail. Furthermore, use cases do not lend themselves to the exploration of design ideas in a form which users can easily relate to. In the e-science programme that sponsored ADVISES [17], the design goal is to introduce new technology with the intention to change users’ working practices, so a thorough understanding of users’ requirements and their reaction to potential designs is necessary. Consequently, we followed the inspiration of agile approaches to development [8] and adapted scenario-based RE techniques to investigate the

users’ work flows [55, 56] and explore how new visualisation tools might be used by academic health care researchers as well as by public health professionals.

To summarise the study of RE practice applied to ADVISES, the project had three research objectives to augment RE perspectives: (1) developing user-centred RE techniques and processes, to specify functionality for multiple-user communities, with a design exploration process for specification in transformative applications, i.e., where no a priori vision of the desired application exists; (2) investigating the concept of HCI requirements and applying HCI patterns as solutions; and (3) exploring the applicability of HCI techniques in RE processes to inform specification of interactive decision-support tools.

The paper is structured in seven subsequent sections. First, related work on the HCI–RE boundary is reviewed. This is followed by the description of the requirements analysis process and its outcomes, leading to an explanation of functional allocation heuristics and their application. Then, the software architecture of the prototype is discussed in the light of the functional specification for different user communities, with an evaluation. The paper concludes with reflections on the lessons learnt and a discussion of the prospects for integrating HCI techniques and knowledge into RE.

2 Related work

Scenario- and goal-oriented approaches to requirements specification have focused on users goals [37, 38]; however, these do not explicitly model users’ tasks or decisions. The goals–skills–preferences technique extends goal modelling taking users’ characteristics and preferences into account [18], while the *i** models record users’ capabilities and responsibilities which can facilitate reasoning or inspection-based approaches to allocating goals and tasks to human or automated agents [16]. Methods for scenario-based requirements engineering have an extensive history [37, 41, 49]; furthermore, these methods share many concepts and techniques with scenario-based design [13], which has HCI origins. Scenario-based RE has also been successfully applied to an iterative user-centred approach in an air traffic control domain [30], so scenarios appear to form a promising bridge between the disciplines.

Another user-centred influence on software specification has been adapting ethnographic analyses to inform software design [32, 46]; however, specific processes for translating knowledge of users into software features have not been articulated apart from a few patterns [31]. Although ethnography has demonstrated the importance of social and user-centred issues in many studies [22, 28], it is resource intensive and does not suggest design solutions or

even the means to specify precise requirements for the problems discovered during ethnographic investigations.

Health and bio-informatics applications have a poor track record of applying sound RE techniques [9]. Few requirements investigations relevant to the ADVISES domain have been reported in the health informatics literature, apart from one investigation into geospatial analysis in health care that demonstrated the need for geographically based analysis, supported by map-based representations integrated with statistical analysis tools [42]. One of the few tools developed for geospatial analysis in health informatics integrated maps with graphs and statistical analysis functions for epidemiological investigations of cancer [11]. This application provided an interesting baseline of ‘requirements by example’ for the ADVISES project.

3 Requirements analysis approach in ADVISES

We adapted scenario-based design [13] and user-centred requirements engineering [50], both of which advocate the use of scenarios, storyboards and prototypes in iterative cycles of requirements elicitation, design exploration and user feedback to create the process. Scenario-based design (SBD) was chosen due to the often volatile and complex requirements of e-science applications. As research practices often change as an investigation evolves, requirements can become a moving target, which is particularly true in the rapidly developing field of bio-health informatics. SBD is well suited to such circumstances because of its iterative approach, which facilitates collaborative design exploration between users and developers. In contrast to use cases, where scenarios are treated as threads or pathways through a semi-structured set of actions [30, 53], scenarios in SBD are concrete stories of user experience, more closely related to stories in agile methods [7, 8]. We also employed use cases in diagram and structured lists of action formats.

The process is summarised in Fig. 1. Unstructured interviews were conducted at the beginning of the project to gain background knowledge on working practices, user preferences and domain norms. Interviews were conducted on-site, allowing the epidemiologists to show us existing software they prefer to use, discuss their data management practices and view example data sets. Workshops provided a good opportunity for users to articulate their processes and abstract concepts and provided data for both the ontology development and understanding of tacit workflow processes, such as how the researchers make decisions about the reliability of a particular data set. Scenarios facilitated exploring possible system designs as well as producing information on the users’ tasks and workflow. Several design representations were used, ranging from

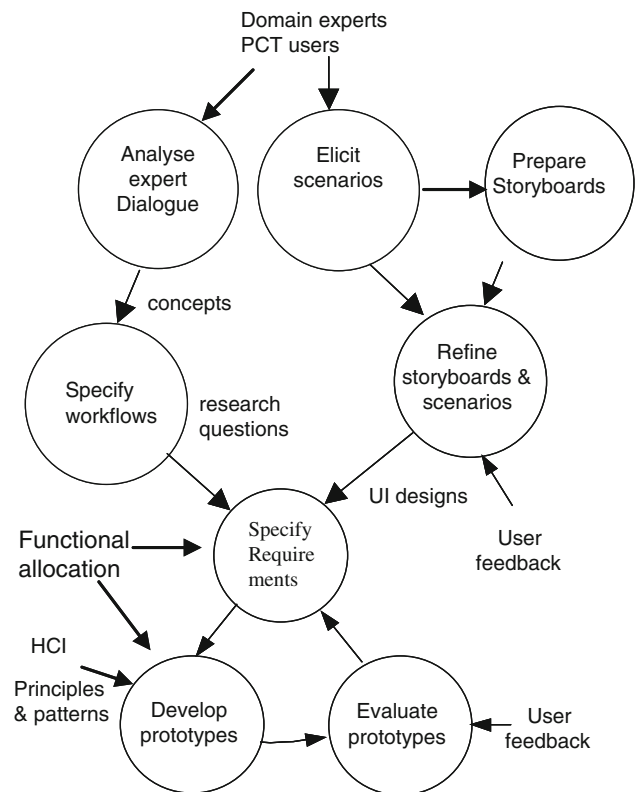


Fig. 1 Scenario-based design process supplemented with HCI techniques and knowledge

simple storyboards or paper prototypes, scripted concept demonstrators to functional prototypes. The various prototypes were used in combination with scenarios in task walkthroughs to explore how the software system might support and even transform the users’ work.

A key orientation to explore requirements as research questions was motivated by the goals–questions–results method [36]. Research questions elicited from domain experts were used to create scenarios and use cases that envisioned a new system to support analysis, e.g., ‘What are the characteristics of the GP-Registered Population in the North West?’ This scenario described how a user could explore a map of patients registered to Primary Care Trusts (PCTs) in the North West, stratifying the population by location, gender and ethnicity. Goal-oriented RE [4] was therefore an important part of the project’s approach. Use cases were used within the design team to document requirements, as well as conventional templates and lists [39]; however, use cases were not shared with users after preliminary meetings indicated users reporting that either the context diagram format contained too little detail or conversely that the action-object structured text was too detailed.

Scenarios were supplemented by analysis of the users’ language in interviews and meetings to develop an

ontology describing the process of epidemiological research. The ontology supported analysis and management of data, as well as informing design of the query interface.

3.1 Experience with scenario-based RE

Preliminary analysis with expert epidemiology researchers elicited high-level goals for a system that supported cleaning systematic errors from poor quality data, querying data sets, statistical analyses of differences between populations and trends over time, and producing displays of the retrieved results on maps and graphs with summaries of the statistical analyses. Since the application had to serve two user communities, we investigated how the initial expert-oriented system might be used by PCT health data analysts in the National Health Service (NHS) who had some appreciation of statistics but were not experts.

The preliminary design used in paper prototyping is illustrated in Fig. 2. This prototype was used in a scenario walkthrough session observing the users' behaviour while they followed scenarios to answer relatively complex, realistic questions. The requirements storyboard walkthrough used several scenarios to assess PCT analyst users' reaction to the prototype and, *inter alia*, the researcher's mode of operation.

Selection of the scenarios was motivated by the world view and rationale gaps problems, to encourage the users to explore functional requirements as well as investigating their domain-specific practices and workflows. For example, one scenario contained data that were too sparse to produce a statistically sound map so the data had to be aggregated into larger units. Another scenario asked users

to interpret population densities in map regions according to the colour coding, to test awareness of the danger of drawing inferences from small samples. Sometimes areas with high levels of diabetes had very small populations, making it difficult to confirm whether they were genuine hotspots or not.

3.2 Requirements specification

The users approved of the basic design concepts: multi-panel displays, query sliders coupled to dynamically updated displays, and the high-level research questions interface rather than SQL-style queries. New requirements emerged for comparison between areas using two maps as well as complex association questions between two or more variables, e.g., 'What is the link between asthma and obesity?' The PCT analysts used local geographic knowledge when interpreting maps and requested support for understanding the implications of local geography; for example, adding overlays of the street network or adding point locations of schools or hospitals. However, the users' actions did show potential errors in walkthroughs with expertise-probing scenarios. For example, the majority of users did not notice the data density problem associated with the colour-coded areas on maps.

Requirements were summarised as goals and an informal process map for researcher epidemiologists and PCT analysts in two workflow diagrams to reflect their practices (see Fig. 3). Researchers progressed through checking and validation tasks to satisfy themselves that the patterns on map displays and accompanying statistical analysis would support valid conclusions, rather than being misled by hotspots in small areas or by inappropriate and sparse distributions. In contrast, PCT analysts did not appear to be concerned with such validation steps; instead, they were more interested in exploring the implications of visible patterns on the map display. The researchers' workflow was more complex, reflecting their approach to analysis with a cycle of database queries, checking the distributions of the retrieved data, and investigating the spatial patterns of epidemiological data in maps before progressing to statistical tests. In contrast, the PCT analysts had a simpler cycle of querying data sets and then visually inspecting distribution patterns on the map. They were more inspection based, while the researchers were more systematic and noted that misleading conclusions could be drawn from inspecting hotspots on maps, which may not be statistically significant at a population level.

In summary, the results of the first phase of requirements analysis pointed to three main conclusions:

1. PCT analysts adopted different workflows from the expert epidemiologists. This reflects different research



Fig. 2 The paper prototype illustrating a map of a fictitious city including an apparent hotspot indicated by shading the distribution

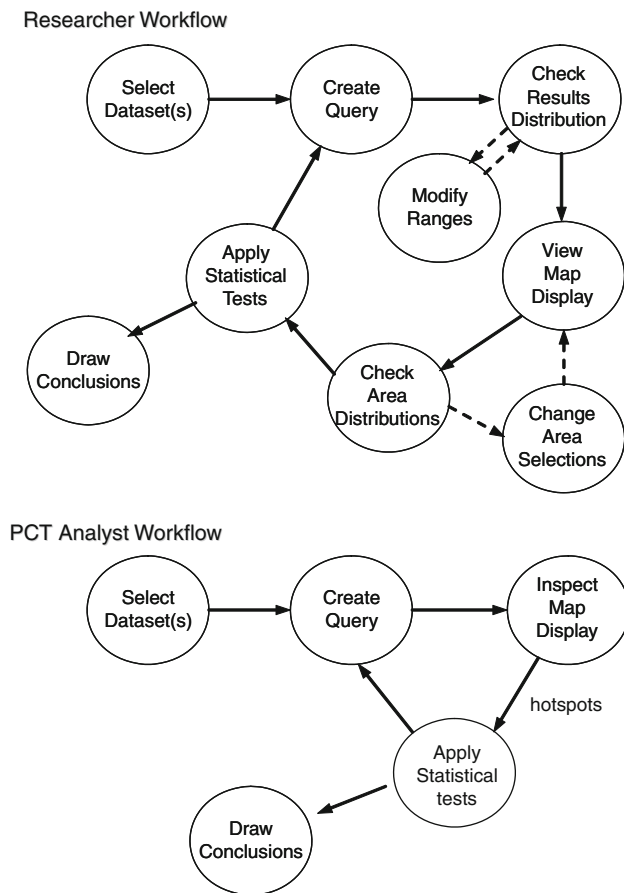


Fig. 3 PCT analyst and researcher workflows

questions; for example, academic epidemiologists are interested in finding general trends and causal influences between several variables, whereas public health professionals requested simpler, location-based questions reflecting their concerns with local health issues.

2. Use of the statistics was often incomplete and sometimes even incorrect, depending on the level of statistical expertise. In particular, some users exhibited a ‘confirmation bias’, employing statistics that confirmed rather than contradicted their hypotheses. Some participants appeared incapable or unwilling to engage in data analysis, assuming that the system would ‘know best’. As a result, they could misinterpret data and draw incorrect conclusions (rationale gap).
3. Local geographical detail is needed so PCT analysts can exploit their detailed local knowledge to interpret patterns apparent on the maps. For example, PCT analysts were interested in plotting the locations of particular services or amenities to see whether these related to the occurrence or outcome of diseases.

The user goals were expanded from the preliminary list to include:

- Data displays as maps and graphs.
- Data distributions shown as discrete categories.
- Functions to segment continuous distributions into discrete categories.
- Display of detailed data as well as map and graph overviews.
- Need to compare trends over time and different areas on maps.
- Research questions ranging from simple queries to complex associations between variables.

4 Integrating HCI and RE in ADVISES

At this stage, HCI design patterns were used to elaborate initial storyboard design and HCI techniques were applied to elaborating requirements for interactive functions.

4.1 HCI requirements and patterns

Two solutions to the HCI requirements were formulated to solve the ‘gaps’ problems. These were proposed in designs for early prototypes and storyboards using HCI principles and design patterns. The user interface had to provide affordances [34] or intuitive functions that help users understand representations of displayed data in the perspective of appropriate domain models of the data and process.

The first pattern, ‘dynamically coupled queries and displays’, recommended that displays be dynamically updated using sliders to express value range queries in an iterative query–view–explore cycle [1, 6]. Sliders allow users to change values in queries leading to dynamic display updates, which facilitates sensitivity analysis by ‘micro querying’. In closely coupled queries, users see changes in the world view corresponding to their queries, and this promotes analysis of emergent visual patterns and their meanings. Research questions were closely coupled with the displays in an iterative querying-visual feedback cycle.

The second pattern, ‘multipanel displays’, recommended tiled windows containing separate views on data pertinent to the user’s task [20]. Users could view concurrent juxtaposed visualisations of maps, graphs and summary statistics, thereby encouraging comprehension of the underlying data models. The third pattern task, ‘appropriate information displays’, advises that information displays should support users’ tasks and decision-making and that only appropriate information should be given to avoid clutter [48]. If the data quantity is large, then an overview-drill down-details on demand control should be provided [10]. Thus, the research questions interface

needed to be linked to information displays appropriate for the user's task. Displays should support users' decisions with representations based on the users' view of the answer (or model), i.e., maps for the location of disease, combined with graphs to show the distribution in the population, zoom and filter controls for details.

4.2 Refining requirements by functional allocation

The requirements produced during the initial exploration phase were refined by functional allocation techniques which originate from the HCI-human factors literature [50, 60]. Functional allocation principles advise on the division of responsibilities between people and computers and categorise functional requirements into three sets: requirements to be fully automated, those to be implemented as manual procedures, and requirements which will be realised by human–computer collaboration. The third category contains requirements for decision support, which need further elaboration to design interactive software to support human cognitive processes. Categorisation of initial requirements is guided by functional allocation heuristics which have their origins in safety–critical systems engineering [45, 60]:

1. Automate repetitive processing, high-volume data processing, and monitoring functions, including deterministic procedures where algorithms can be defined.
2. Complex cognitive tasks become user processes, e.g., interpreting complex patterns, judgement with complex and uncertain data, and general-purpose problem solving.
3. Communication and less deterministic processes, e.g., evaluation, negotiation and judgement, are suitable for people.
4. Control systems with unpredictable events need people to be in command, although the monitoring/alerting may be automated.

5. Intelligent software functions should be considered when the necessary knowledge can be formalised to support novice users and reduce cognitive effort.
6. Decision-support functions for (2–5) include providing information processing by filtering, ranking, and sorting options, with models and simulations to support sensitivity analysis and explore options.

Requirements for human–computer cooperation are refined to specify the information that users need to take a decision; for example, the computer provides facilities to sort and rank suppliers by different criteria (price, reliability and location) to help the user make a purchasing decision. Decision-support requirements are specified as the information that users needs to take a decision, how much background information is required, what options should be provided, etc.

Functional allocation heuristics were applied to each task in the user workflows to elaborate these baseline requirements. Many requirements fell into an intermediate category of collaborative functions in which both user and software system play a role. Information display requirements were also refined using the heuristics and guidelines in the task-information analysis method [48] which provide a walkthrough approach to specification of decision-support functions with questions that focus on the user's information needs during each task step. The outcome of this analysis indicated more detailed requirements for information displays, maps, graphs and basic descriptive statistics which could be mapped to users' tasks and interactive controls. For example, comparison tasks implied two or more sets of data for different areas (area questions), graph overlays (comparing variables), etc. The high-level functional and UI requirements are summarised in Table 1.

Functional allocation analysis also indicated that two expert advisors would be necessary to encourage PCT analysts to successfully draw sound inferences from the

Table 1 Workflow tasks (column 1) with support functions (column 2) following functional allocation analysis

Workflow task	Function	Allocation	UI feature
Select data sets	Database access	Automated	Select DB menu
Select data sets	Data cleaning	Interactive	Data file editor
Create query	Research questions	Interactive	Question menu
Create query	Data retrieval	Automated	
Apply statistical test	Statistical analysis	Decision support	Choice menu
Check results/area distributions	Graphs and statistical displays	Decision support	Interactive map areas
Change/modify ranges or areas	Graphs and statistical displays	Decision support	Sliders for sensitivity analysis
Draw conclusions	Map displays	Decision support	Dynamic link to query-retrieval
Draw conclusions	Annotation of results	Interactive	Terminology menu/editor

data. The first (statistics) advisor would assist them to query, evaluate and explore data sets, while the second (visualisation) expert would automatically encode value ranges on map and graph displays to optimise pattern analysis and hence bridge the world view gap. These advisors were motivated by the requirement to caution against unsafe inferences being drawn from sparse or awkwardly distributed data in map displays and to save the users effort in choosing visual display coding. A third conclusion was to yoke the research questions and workflows to preset configurations of displays to bridge the rationale gap. This conclusion was the consequence of applying the HCI design patterns (see Sect. 4.1).

Analysis of research questions and information requirements suggested that complex multivariate queries needed to be supported, so researcher users could explore the intersection of two or more influences on a problem, for example the effect of socio-economic background and location on obesity or a collocation of high levels of obesity and asthma. These requirements implied expertise in visualisation which the users were unlikely to possess; hence, automated expert assistance was appropriate. The design was also motivated by display combination, so users could view concurrent juxtaposed visualisations of maps and graphs, to encourage comprehension of the underlying data models. In the following section, the interactive functions included in the final prototype are described, with the rationale for software architecture.

4.3 Specification of interactive functions

The workflows from the two user communities posed problems in how to allocate somewhat different sets of the requirements to each user community. Producing two versions of ADVISES would lead to maintenance concerns and incur the additional expense of duplicating software processes. The solution adopted was to develop a layered architecture with a core functionality targeted at the PCT users, with an outer layer of functionality for the domain expert users who require additional statistical analysis. Exposure of the functions was controlled by a role options menu for workflow configuration.

Functional allocation analysis identified several requirements for user interface components. Some of these could be directly mapped to user interface implementation classes present in development environments (e.g. JAVA Swing or the Microsoft.NET Framework); for example, display panels, windows, sliders and menu checklists for queries. However, functions for dialogue management and intelligent advisors needed further elaboration.

4.3.1 Dialogue manager

The dialogue manager links research questions to a set of appropriate display configurations to support workflow tasks. The query phase aims to build appropriate data models by providing high-level research questions users want to ask, followed by an unfolding series of menu picking lists containing clusters of related variables, e.g., person demographics, disease, lifestyle attributes. Analysis of the users' language and working practices reported in this paper and in previous studies [54, 55] suggested that a limited set of research questions could satisfy our users' needs as follows:

- Comparison (between areas, genders, cohorts, etc.).
- Association/co-variation (between genders, cohorts, treatments).
- Difference (from a set threshold for cohorts, areas).
- Trends (over time, gradients across areas).
- Location (where, proximity to).

Users form queries by first picking a high-level question type. Then, queries are elaborated by selecting one or more subject populations from the available data sets with variables such as age, gender, socio-demographics, lifestyle, medical history, followed by the desired measures, which were usually BMI (body mass index) and other anthropometrics. Queries are organised into menu-picking lists (see Fig. 4), configured with constraint rules so that only appropriate choices are offered as the query develops, e.g., trend questions prompt for the time period and intervals; location questions request areas or proximity to displays; and comparison questions prompt for between-populations or variables (e.g. gender) within a population.

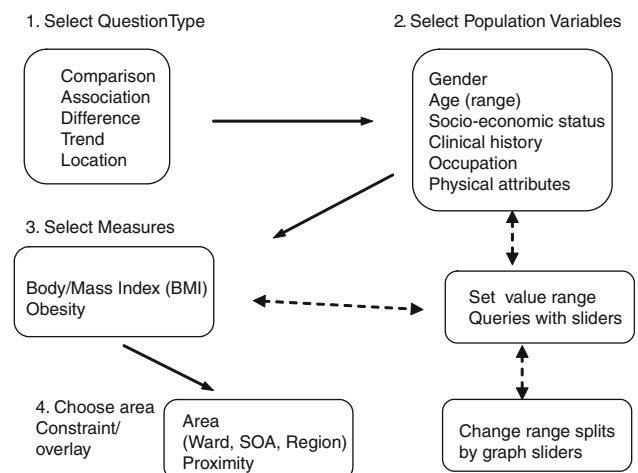


Fig. 4 Query interface operational sequence

The location questions are elaborated with specialisations to create display overlays that support the PCT analysts' desires to investigate local implications of hotspots, such as proximity to doctors' surgeries, hospitals, schools, sports facilities. All queries can be constrained by map areas and overlays selected for additional spatial data, e.g., point location of health centres, sports facilities. Query range sliders become active once the population and measures variables are selected, e.g., for age or BMI range, and the distribution, graphs and maps are displayed. The system automatically selects the appropriate representation of results on maps and graphical displays according to the questions type, as shown in Table 2.

Basic descriptive statistics are displayed in a side panel next to the query area, and the main display area contains a mix of graphs and maps according to the rules derived from

Table 2. The display for the 'check area density' task in the current prototype is illustrated in Fig. 5. The multipanel display affords rapid data inspection and exploration of epidemiology data sets, while colour and patterns in the charts indicate sparse and non-normal distributions when statistical analyses and other inferences may be invalid. Incremental analysis is supported by sliders for value-range queries, so analysts can carry out sensitivity analysis by changing range values, e.g., inspect obesity by area by age.

Range category histograms and descriptive statistics support the 'check the data distribution' task. Users can inspect the shape of the distribution and use skew and kurtosis metrics to check symmetry and normality. They can segment a continuous distribution into discrete categories (e.g. extreme, high, medium, low BMI) using sliders to subdivide the range. This enables sensitivity analysis of

Table 2 Presentation display templates linked to question types

Question type/requirement	Window type	Analysis functions supported
Compare areas	2–4 maps	Comparisons for hotspots, spatial distributions
Difference	Map + quintile graph, box-and-whisker	Threshold setting, colour code for difference
Association	2 maps + quintile graph	2 or more distributions on graphs, inspect co-variance
Trend	Maps + quintile graph	Graph of measure over time, animate map for area change over time
Location	One or more maps	Overlays on maps for point locations
Check distribution	Map + quintile graph, box-and-whisker	Distribution segment inspection, data density
Check area density	Map + forest plot	Validate analyses, area segment inspection, data density

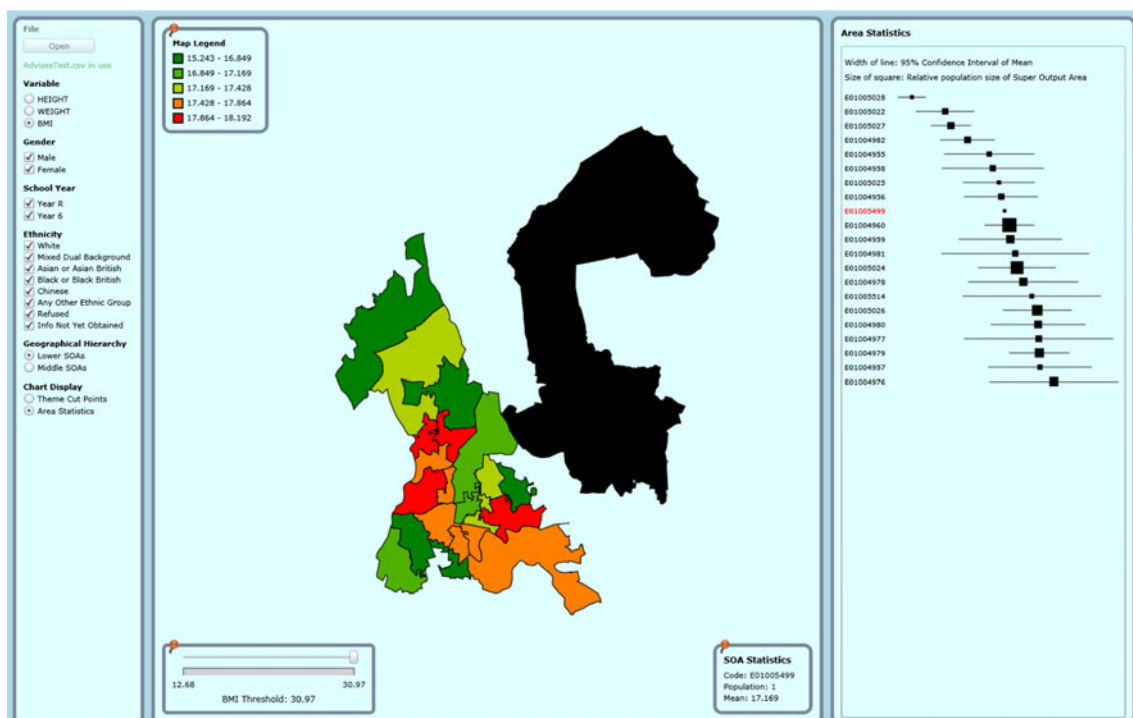


Fig. 5 User interface of the current prototype showing the map-chart display combination for the 'check area density' task

range–category subdivisions to ensure, for instance, that distribution tails have sufficient data points for valid statistical analysis. Forest plots (horizontal histograms) are coupled to the map displays so the boxes represent distributions (means, confidence intervals) within map subareas. These plots support the ‘check area distributions’ task; for example, long thin or short fat boxes indicate sparse distributions with high standard deviations (long thin) or high kurtosis (short fat).

The graphs and maps are coordinated and queryable surfaces, so users can point and click on subareas of the map to extract more detailed information such as the name, population or deprivation score for that area. The multi-panel display affords rapid data inspection and exploration of epidemiology data sets, while patterns in the charts indicate sparse and non-normal distributions when statistical analyses and other inferences may be invalid.

4.3.2 Statistics advisor

Functional allocation following requirements analysis identified the need to support less expert PCT users who might make mistakes in statistical analysis. The aim of the statistical advisor is to warn users about sparse distributions where false inferences may be drawn from low numbers. However, there are occasions when looking at low numbers is unavoidable, for example when investigating a rare disease, so the advisors are configurable and can be turned off under user control. Some advice is given passively by highlighting areas in the presentations that warrant attention, with pop-ups to explain the warnings.

A monitor alert function compares map area populations and densities (populations/area) and distribution statistics (SD, skew and kurtosis) to alert the user when any of these values exceeds a preset threshold. A pop-up containing the threshold value appears when the user’s mouse is placed over the highlighted figure/area. The alert reminds the user about the properties of the underlying data distribution and thus contributes to closing the rationale gap. Since the validity of distribution depends on the nature of the data set, the alert function is configurable so the rules can be edited to deal with general health (normal data sets) or rare events (disease epidemiology—non-normal data sets).

4.3.3 Visualisation advisor

Information analysis indicated that more than one variable might need to be displayed in the results of research queries. Complex research questions may involve 2–3 variables, e.g., ‘What is the distribution of type II diabetes and obesity for different levels of socio-economic deprivation in different areas of the North West health region?’. This association–location question implies visualisation of the

average density of diabetes patients and overweight people in each health district. Design alternatives were to display data from different variables, e.g., ‘What is the association between obesity and asthma in PCT areas?’, in two separate maps. However, this makes comparison difficult since visually analysing areas in a second display while remembering patterns from the first is error prone. HCI visualisation design guidelines [47] advise that displays are overlaid, and interactive controls are provided to facilitate comparisons. Design of the visualisation advisor was therefore motivated by the requirement to display more than one variable on a map. Visual coding requires psychological knowledge; however, the knowledge can be formalised so design of an expert advisor was suggested from functional allocation analysis. The module automatically codes the range categories on the maps and graphs using rules derived from the HCI visualisation literature [47, 58, 59].

Advice on colour coding favoured a single colour saturation scale rather than a rainbow spectrum [59]. Guidance on texture coding was not so specific, so we decided to use single texture density gradients (e.g. dot stipples, bar density) rather than several different textures, to avoid imposing a learning burden on users [44]. Two variables could be represented on one area: one by colour and the other by texture (see Fig. 6).

The visualisation expert inspects metadata associated with the data set to determine whether the variable has a continuous distribution, is discrete or is an enumerated set. This indicates the number of categories for each variable, so in the case of continuous distributions, a default quintile range split is assumed.

The visualisation expert automatically selects the codings, favouring colour if only one variable is displayed. When small map areas are present, a warning is given that discrimination of categories in small areas may not be reliable, since the texture gradients will not be easy to discriminate. This problem was overcome by implementing a zoom and pan tool.

5 Implementation

The functionally complete prototype was implemented as an MS Silverlight application using Visual C# as the base language. MS Silverlight was chosen for its integration of multimedia, graphics, animations and interactivity into a single runtime environment. It was particularly useful when animating map displays for trend questions so that successive displays gradually morph into each other to enable users to see the trend change over time within different map areas. A distributed architecture was adopted, and components were developed as web services where

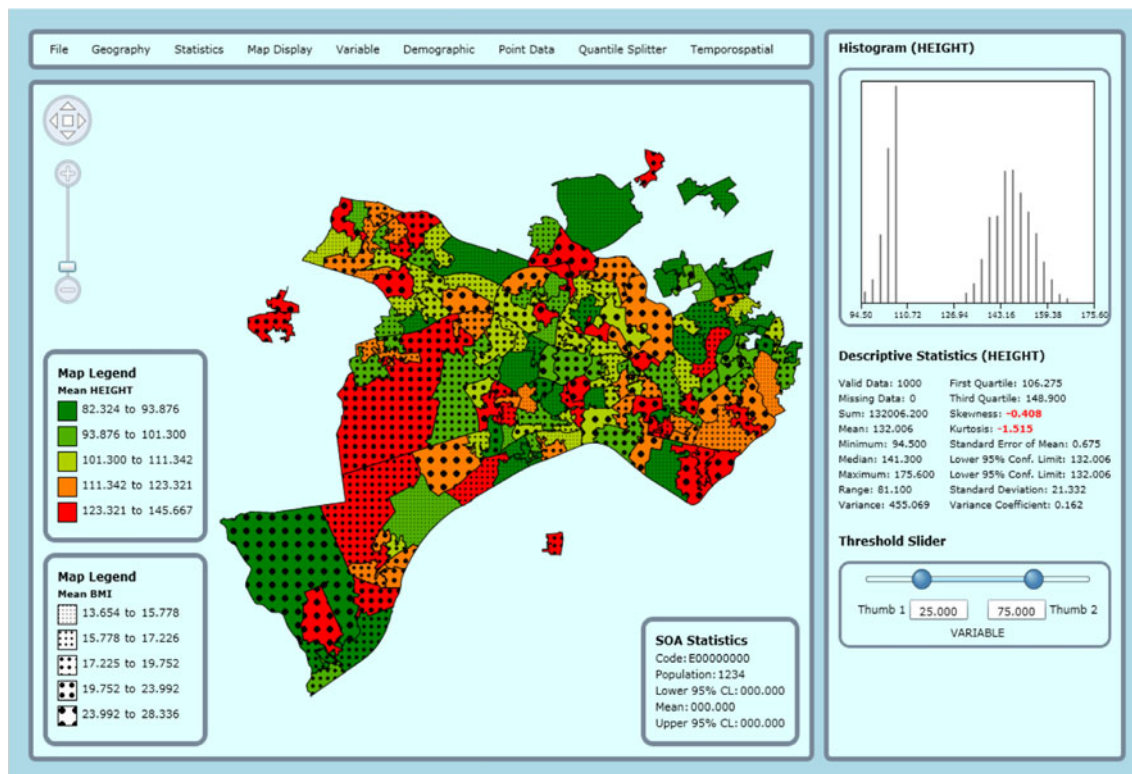


Fig. 6 Visual encoding using red–green colour saturation and texture gradients (colour figure online)

server and/or database access was required. The application was implemented with major class packages in the following functional areas:

- *Data set load, access and cleaning:* loads data sets from remote servers and carries out initial validation of data (cleaning for missing fields, etc.).
- *Map displays:* loads shape files from the UK Land Registry server and displays maps using MS Charting libraries. Map displays can be overlaid so point data (e.g. location of health clinics, sports facilities, etc.) can be displayed at appropriate locations.
- *Charts and statistics displays:* runs basic statistical analysis scripts (R-script calls) and then displays range split histograms, box-and-whisker plots, etc., using MS Charting.
- *Dialogue management:* handles the query interface, interactive query-by-pointing and sliders, as well as linking question types to appropriate window display templates.
- *Expert advisors:* classes which implement the statistics and visualisation experts, with data set monitors to trigger advice.
- *Annotation tagger:* provides picking lists of terms from a controlled vocabulary, which can be associated with map and chart output, based on an ontology of spatial epidemiology from the requirements analysis. The

output can be saved with tags and free-format text comments.

Map shape files, databases, query handling and statistical analysis components were remote services; other components were client resident.

It is worth noting that several functions did not get implemented, in particular a set of configuration editors that would have made the ADVISES system into a portable, flexible toolset which could be configured for different domains to support other scientific data-driven research requiring visualisation, e.g., population dynamics researchers.

6 Evaluation

The prototype was subject to two cycles of evaluation after the requirements exploration–design phase, illustrated in Table 3. Round one was formative, for usability debugging and design improvement, while the second round was more summative in nature and captured users' attitudes and satisfaction ratings for the prototype. In both rounds, users completed a representative set of tasks which enabled assessment of system performance.

During each round of the evaluation, all participants quickly and confidently created their first map and, without

Table 3 Summary of participants' responses. Interviews were transcribed and the participants' answers coded as positive, negative or neutral

	Round 1 responses		Round 2 responses	
	Positive	Negative	Positive	Negative
First impressions of the system?	5	3	6	2
System easy to understand?	7	1	7	1
Enjoy using the system?	8	0	8	0
Would use the system in your job?	7	1	5	1
			2 possibly	

being asked to do so, went on to explore the map, looking at trends, subdividing data into smaller categories, e.g., men and women, switching between geographic boundaries and then reviewing the associated statistics to help them understand the significance (or otherwise) of observed patterns. Participants found the combination of geographic visualisation and descriptive statistics powerful and easy to explore:

I love stuff like this; it's nice having the descriptive stats; when you put data into [commercial GIS package] it can be misleading.

It's really easy to figure out; it's at your fingertips.

After working through the set of tasks, users were asked about their experiences with the system. The majority felt that their experiences were positive, but some users felt that, although they had successfully created a map, the system was not welcoming:

It's very blank and a bit unfriendly looking. Once the data is in it looks much better.

It's not clear where to start; there should be a big 'start here' sign.

These comments led to a redesign of the initial map-creation process; this redesign was evaluated, and further mixed reaction has led to additional design changes to be tested in the final round of evaluation currently in progress. Thus, each round of the evaluation directly influenced the next iteration of design and development.

7 Reflections and lessons learnt

Of the requirements techniques we employed, the combination of storyboards, scenarios and prototypes integrated in a user-centred design cycle was the key to user engagement. Visualisation of realistic designs enabled the users to critique and contribute ideas in their own terms without understanding software engineering notations. Our experience has been that even simple notations such as use cases present a barrier to understanding; furthermore, abstract models are less meaningful for users. The second reflection is the importance of conversation and dialogue,

especially when it is anchored in the user's domain and language. Talking through and demonstrating working practices were important motivators for end users. In workshops, conversations have the added advantage that users outnumber software professionals and hence own the dialogue and can direct it towards their own goals.

The mix of designer-led initiative using HCI patterns and user-centred design that responded to user requirements worked well. However, ADVISES was essentially an action research project where HCI knowledge was transferred from the first author to the other authors who undertook the analysis, design and implementation. Within this limitation, the approach shows promise. The basic design paradigm of multiple displays and dynamically coupled queries and displays introduced research-inspired design into health informatics tools. These design concepts stimulated interest and hence engagement among the users. The expert advisor modules, which were a designer initiative in response to problems discovered during the requirements analysis, were not seen as an imposition by the users, as they might have been; for instance, the statistics advisor might be viewed as criticising users' judgement. We attribute user acceptance of these ideas to the user-centred design process where the problems and proposed solutions were discussed openly with the users and illustrated in storyboards and prototypes so that the design implications were explicit. On the user-led requirements side, several aspects of the design arose directly from users' suggestions; for example, the two-map comparative displays, changes to the forest plots, and functions for subdividing continuous distributions into range categories. Iterative user-centred design made the changes in response to users' requests visible in a short time period, which was a positive motivation for engagement. Transformed working practices emerged throughout the process as users responded to presentations of the tools, so they accepted new workflow by 'osmosis' as tools and tasks co-evolved during the project.

Models were notable for their absence in the RE process. Although use cases and class diagrams, etc., were used within the development team, these representations were not shared with the users, following feedback from preliminary exposure to use cases. Analyst user

communication and design exploration of requirements was driven by storyboards, prototypes and scenarios. Even the workflow diagrams, which were based on a data flow diagram format, had only a minor influence on the process. Other RE notations, for example goal models and i^* diagrams, were considered but not used since users expressed a strong preference for concrete presentations. On reflection, diagram representations might have been employed more forcefully to resolve process sequence issues; for example, the goal or process step to ‘divide continuous distributions into discrete categories’ remained ambiguous in the early phases of analysis, until the requirements became clear, when the design of graphical displays was critiqued.

High-level (business style) requirements were never articulated in ADVISES since the project’s terms of reference were essentially set by the grant proposal objectives. This stated the high-level goal: to develop interactive graphically based tools to support epidemiological researchers, essentially ‘identify causes of disease including spatial factors’. This objective drove the research role analysis; however, a second tacit objective was to spread research best practice to local health analysts, hence the secondary focus on PCT analysts. However, PCT analysts’ top-level requirements only emerged during the analysis, as ‘identify and manage outbreaks of disease within my area’. The lesson, not usual, in requirements practice is to beware of tacit political agendas; in this case, the researchers’ tacit desire to improve work practices of PCT analysts. Fortunately, we were able to resolve these potential conflicts with a mix of tools so PCT analysts would achieve their goals of local analysis and management of disease, while the expert advisor modules provided a resource to improve their analysis practice.

8 Discussion

HCI influenced the requirements specification process as well as the consequent requirements for the ADVISES project in three ways. First, the scenario-based process facilitated exploration of users’ requirements and, more importantly, their design realisation. This enabled users to contribute to developing a software specification which would change their work practices. Experience with a similar user-centred development approach was reported by Maiden and Robertson [30]. The second influence was application of functional allocation as a means of refining the functional requirements and user interface architecture. The functional allocation heuristics which proved to be useful in ADVISES could be applied in RE approaches when different system implementations are being considered; for example, when strategic rationale models are

created to explore alternative implementation boundaries in strategic dependency i^* models [33]. The third influence was application of HCI knowledge in the form of principles and patterns, in particular as solutions to visualisation problems. HCI knowledge was supplied by the first author, supplemented by HCI design patterns literature (e.g. [12]). HCI requirements based on the ‘gaps’ problems [2] stimulated the visualisation design as well as pointing towards the patterns solution, e.g., multiple displays enable different users to scan the maps and graphs according to their needs. Linking research questions to display templates supports the users’ workflow more directly, by providing the necessary information related to the users’ tasks. Although concurrent multipanel displays may appear to increase complexity, none of our users complained about the displays being too complex.

Previous methods and approaches to integrating HCI into requirements engineering have focused on reorienting the development process to emphasise user goals, iterative development, scenarios and prototyping [4, 7, 37]. Our approach shares goal orientation and use of scenarios with mainstream RE; however, the new contributions are, first, integration of functional allocation into the RE process and, second, application of HCI design patterns. Interactive functions such as the statistics advisor and dialogue manager evolved in response to user requests; however, the flexibility of the dialogue manager concept enabled new display configurations to be added without major redesign. The allocation of function requirements to user or system roles that we developed builds on the task-functional analysis proposed by Lauesen [25], by adding heuristics to guide the allocation process.

However, the ADVISES experience also shows the constraints that may arise from a user-centred approach. The ADVISES project was part of the UK e-science programme which aimed to produce distributed, service-oriented solutions for collaborative support in scientific domains. Hence, there were architectural requirements for portability, and the software needed to be customisable for other research domains beyond epidemiology. We originally intended to produce configuration editors to enable end-user adaptation of the system to different databases, research questions–query interfaces and displays within the limitations of maps, and a small range of chart types.

Customisation proved to be difficult. Each layer of adaptation required editors to configure functionality, databases and data displays, as well as more sophisticated end-user programming facilities for workflow organisation and statistical advisors. The increasing complexity exposed the penalty of degrading trust with our primary stakeholders in epidemiology; furthermore, it was difficult to access and engage users in other domains. Users are inevitably engaged with their own specific domain and respond

positively to specific solutions; consequently, general design concerns for customisation were seen as a distraction and a barrier to effective use. Hence, the requirements imposed by the e-science programme for more general customisable solutions tended to militate against successful user engagement in the domain of epidemiology and public health. A possible way forward might be to adopt a phased approach to user-centred development in the small (and within a specific domain), followed by generalisation of the architecture for more customisable solutions. Of course, this assumes that architectural decisions are made in the user-specific phase with generalisation in mind. We did anticipate these architectural requirements with a service-oriented approach, maximising modularity and minimising coupling, although we have no evidence to assess our success in this endeavour. While requirements for diverse solutions are an accepted overhead in product lines, trade-offs for customisation in general RE processes have received little attention, apart from COTS procurement processes [29].

This evidence will arise in the future, when we will extend the architecture and scope of ADVISES with configuration editors so the software can be customised to different research domains and analysis tasks, anticipating architectural evolution as the number of stakeholders increases [5]. The current prototype is being developed into a product version for both PCT and research users. Since this development is being partially sponsored by the NHS PCT users themselves, we take this to be a partial validation of our user-centred RE approach. Further contextual evaluations in the users' workplace are planned to assess how successful the research questions and multiview displays are in bridging the world view and rationale gaps, as well as exploring the fit of the ADVISES architecture with the working practices of diverse stakeholders to deliver effective support for health informatics.

Acknowledgments This research was funded by the EPSRC e-Science Usability Programme grant ADVISES: ADaptive VISualisation tools for E-Science collaboration. The authors would like to thank Iain Buchan and PCT users for their help in the requirements analysis.

References

- Ahlberg C, Shneiderman B (1994) Visual information seeking: tight coupling of dynamic query filters with starfield displays. In: Human factors in computing systems of CHI 94 conference proceedings. ACM Press, New York
- Amar R, Stasko J (2004) A knowledge task-based framework for design and evaluation of information visualisations. In: Proceedings of IEEE symposium on information visualisation 2004, InfoVis04. IEEE Computer Society Press, Los Alamitos, CA
- Anton AI, Earp JB (2004) A requirements taxonomy for reducing website privacy vulnerabilities. *Requir Eng* 9(3):169–185
- Anton AI, Potts C (1998) The use of goals to surface requirements for evolving systems. In: Proceedings of 1998 international conference on software engineering: forging new links. IEEE Computer Society Press, Los Alamitos, CA
- Anton AI, Potts C (2001) Functional palaeontology: system evolution as the user sees it. In: Proceedings of international conference on software engineering, ICSE-01. IEEE Computer Society Press, Los Alamitos, CA
- Aris A, Shneiderman B, Plaisant C, Shmueli G, Jank W (2005) Representing unevenly-spaced time series data for visualization and interactive exploration. In: Proceedings of INTERACT 2005. Springer, Berlin
- Beck K (1999) Extreme programming explained: embracing change. Addison-Wesley, New York
- Beck K (2003) Test driven development by example. Addison-Wesley, New York
- Beckles B (2005) User requirements for UK e-science grid environments. In: Proceedings of UK e-science all-hands meeting
- Bederson B, Shneiderman B (2003) The craft of information visualization: readings and reflections. Morgan Kaufmann, San Francisco
- Bhowmick T, Robinson AC, Gruver A, MacEachren AM, Lengerisch EJ (2008) Distributed usability evaluation of the *Pennsylvania Cancer Atlas*. *Int J Health Geogr* 7(36)
- Borchers J (2001) A pattern approach to interaction design. Wiley, Chichester
- Carroll JM (2000) Making use: scenario-based design of human-computer interactions. MIT Press, Cambridge, MA
- Daneva M, Wieringa RJ (2006) A requirements engineering framework for cross-organizational ERP systems. *Requir Eng* 11(3):194–204
- Fischer G (2001) User modeling in human-computer interaction. *User Model User-Adapt Interact* 11(1/2):65–86
- Giorgini P, Massacci F, Mylopoulos J, Zannone N (2005) Modeling security requirements through ownership, permission, and delegation. In: Proceedings: 13th IEEE international conference on requirements engineering, Paris, 29 August–2 September 2005. IEEE Computer Society Press, Los Alamitos, CA, pp 167–176
- Hey T, Trefethen A (2003) The data deluge: an e-science perspective. In: Berman F, Fox G, Hey A (eds) Grid computing: making the global infrastructure a reality. Wiley, Chichester
- Hui B, Laiskos S, Mylopoulos J (2003) Requirements analysis for customisable software: a goals skills preferences framework. In: Proceedings of IEEE joint international conference on requirements engineering. IEEE Computer Society Press, Los Alamitos, CA
- ISO (1995) ISO 13407: Human-centred design processes for interactive systems. International Standards Organisation
- ISO (2000) ISO 14915-3: Software ergonomics for multimedia user interfaces. Part 3: Media selection and combination. Draft international standard. International Standards Organisation
- Jacobson I, Booch G, Rumbaugh J (2005) The Unified Modelling Language user guide. Addison Wesley, Boston, MA
- Jirotko M, Procter R, Hartwood M, Slack R, Coopmans C, Hinds C, Voss A (2005) Collaboration and trust in healthcare innovation: the eDiaMoND case study. *J Comput-Support Co-op Work* 14(4):369–398
- Kazman R (2003) Report on the ICSE workshops: bridge the gaps between software engineering and human computer interaction. *SIGSOFT Eng Notes* 28(6)
- Kotonya G, Sommerville I (1996) Requirements engineering with viewpoints. *Softw Eng J* 11(1):5–18
- Lauesen S (2002) Software requirements: styles and techniques. Addison-Wesley, Harlow

26. Lauesen S (2003) Task descriptions as functional requirements. *IEEE Softw* (March/April), pp 58–65
27. Lieberman H (ed) (2001) *Your wish is my command: programming by example*. Morgan Kaufmann, San Francisco
28. Luff P, Jirotko M, Heath C, Greatbatch D (1993) Tasks and social interaction: the relevance of naturalistic analyses of conduct for requirements engineering. In: *Proceedings of the 1st international symposium on requirements engineering—RE'93*. IEEE Computer Society Press, Los Alamitos, CA
29. Maiden NAM, Ncube C (1998) Acquiring requirements for Commercial Off-The-Shelf package selection. *IEEE Softw* 15(2):46–56
30. Maiden NAM, Robertson S (2005) Developing use cases and scenarios in the requirements process. In: *Proceedings of international conference on software engineering ICSE 05*. IEEE Computer Society Press, Los Alamitos, CA
31. Martin D, Rouncefield M, Rodden T, Sommerville I, Viller S (2001) Finding patterns in the fieldwork. In: *Proceedings of seventh European conference on CSCW, ECSCW'01*. Kluwer, Norwell, MA
32. Martin D, Rooksby J, Rouncefield M, Sommerville I (2007) 'Good' organisational reasons for 'bad' software testing: an ethnographic study of testing in a small software company. In: *Proceedings of the 29th international conference on software engineering (ICSE)*. IEEE Computer Society Press, Los Alamitos, CA
33. Mylopoulos J, Chung L, Yu E (1999) From object-oriented to goal-oriented requirements analysis. *Commun ACM* 42(1):31–37
34. Norman DA (1999) *The invisible computer: why good products can fail, the personal computer is so complex, and information appliances are the solution*. MIT Press, Cambridge, MA
35. Nuseibeh B (2001) Weaving together requirements and architecture. *IEEE Comput* 34(3):115–117
36. Perrone V, Finkelstein A, Goldin L, Krammer J, Parkinson H, Reddington F (2006) Developing an integrative platform for cancer research: a requirements engineering perspective. In: *Proceedings of UK all hands e-science meeting*
37. Potts C (1999) ScenIC: a strategy for inquiry-driven requirements determination. In: *Proceedings of 4th IEEE international symposium on requirements engineering*. IEEE Computer Society Press, Los Alamitos, CA
38. Potts C, Anton AI (1998) A representational framework for scenarios of system use. *Requir Eng* 3:219–241
39. Robertson S, Robertson J (1999) *Mastering the requirements process*. Addison Wesley, Harlow
40. Robinson WN (2006) Requirements monitoring for enterprise systems. *Requir Eng* 11(1):17–41
41. Rolland C, Achour CB, Cauvet C, Ralyte J, Sutcliffe AG, Maiden NAM et al (1998) A proposal for a scenario classification framework. *Requir Eng* 3(1):23–47
42. Scotch M, Parmanto B, Gadd CS, Sharma RK (2006) Exploring the role of GIS during community health assessment problem solving: experiences of public health professionals. *Int J Health Geogr* 39(5):1–10
43. Seffah A, Gulliksen J, Desmarais MCE (2005) *Human-centered software engineering: integrating usability in the software development lifecycle*. Springer, Berlin
44. Shenat HH, Interrante V (2005) Utilizing texture for multi-variate visualization. *Graphite*, pp 443–446
45. Sheridan TB (2000) Function allocation: Algorithm, alchemy or apostasy? *Int J Hum-Comput Stud* 52(2):203–216
46. Sommerville I, Sawyer P (1997) *Requirements engineering: a good practice guide*. Wiley, Chichester
47. Spence R (2007) *Information visualisation*, 2nd edn. Pearson Education, Harlow
48. Sutcliffe AG (1997) Task-related information analysis. *Int J Hum-Comput Stud* 47(2):223–257
49. Sutcliffe AG (1998) Scenario-based requirements analysis. *Requir Eng* 3(1):48–65
50. Sutcliffe AG (2002) *User-centred requirements engineering*. Springer, London
51. Sutcliffe AG (in press). Analysing the effectiveness of human activity systems with i*. In: Giorgini P, Yu ES (eds), *Social modelling for requirements engineering*. MIT Press, Cambridge, MA
52. Sutcliffe AG, Fickas S, Sohlberg MM (2006) PC-RE: a method for personal and contextual requirements engineering with some experience. *Requir Eng* 11:157–163
53. Sutcliffe AG, Maiden NAM, Minocha S, Manuel D (1998) Supporting scenario-based requirements engineering. *IEEE Trans Softw Eng* 24(12):1072–1088
54. Sutcliffe AG, Thew S, Venters C, De Bruijn O, McNaught J, Procter R, Buchan I (2007) ADVISES Project: scenario-based requirements analysis for e-science applications. In: *Proceedings of UK e-science all hands conference 2007*. <http://www.allhands.org.uk/2007/>
55. Thew S, Sutcliffe AG, De Bruijn O, McNaught J, Procter R, Venters C, Buchan I (2008) Experience in e-science requirements engineering. In: *Proceedings of 16th IEEE international requirements engineering conference RE 2008*. IEEE Computer Society Press, Los Alamitos, CA
56. Thew S, Sutcliffe AG, Procter R, De Bruijn O, McNaught J, Venters C, Buchan I (2009) Requirements engineering for e-science: experiences in epidemiology. *IEEE Softw* 26(1):80–87
57. Tidwell J (2006) *Designing interfaces: patterns for effective interaction design*. O'Reilly Media, Sebastopol, CA
58. Tufte ER (1997) *Visual explanations: images and quantities, evidence and narrative*. Graphics Press, Cheshire, CN
59. Ware C (2000) *Information visualization: perception for design*. Morgan Kaufmann, San Francisco
60. Wright P, Dearden A, Fields B (2000) Function allocation: a perspective from studies of work practice. *Int J Hum-Comput* 52(2):335–355
61. Yu E (2009) Social modeling and i*. In: Borgida AT, Chaudhri V, Giorgini P, Yu ES (eds) *Conceptual modeling: foundations and applications—essays in honor of John Mylopoulos (LNCS volume 5600)*. Springer, Berlin