Requirements elicitation: Towards the unknown unknowns

Conference Paper · July 2013

DOI: 10.1109/RE.2013.6636709

Alistair G. Sutcliffe
The University of Manchester

Requirements Elicitation: Towards the Unknown Unknowns

Alistair Sutcliffe, Pete Sawyer
School of Computing and Communications
University of Lancaster, Lancaster, LA1 4WA, UK
a.g.sutcliffe@lancaster.ac.uk, p.sawyer@lancaster.ac.uk

Abstract—**Requirements elicitation research is reviewed using a framework categorising the relative 'knowness' of requirements specification and Common Ground discourse theory. The main contribution of this survey is to review requirements elicitation from the perspective of this framework and propose a road map of research to tackle outstanding elicitation problems involving tacit knowledge. Elicitation techniques (interviews, scenarios, prototypes, etc.) are investigated, followed by representations, models and support tools. The survey results suggest that elicitation techniques appear to be relatively mature, although new areas of creative requirements are emerging. Representations and models are also well established although there is potential for more sophisticated modelling of domain knowledge. While model-checking tools continue to become more elaborate, more growth is apparent in NL tools such as text mining and IR which help to categorize and disambiguate requirements. Social collaboration support is a relatively new area that facilitates categorisation, prioritisation and matching collections of requirements for product line versions. A road map for future requirements elicitation research is proposed investigating the prospects for techniques, models and tools in green-field domains where few solutions exist, contrasted with brown-field domains where collections of requirements and products already exist. The paper concludes with remarks on the possibility of elicitation tackling the most difficult question of 'unknown unknown' requirements.**

Index Terms—**Requirements elicitation, models, techniques, common ground, tacit knowledge**

I. Introduction

Requirements elicitation is a relatively mature area of RE [1], [2], and the basic techniques (i.e. interviews, observation, scenarios, workshops, focus groups, protocols, prototypes, models, etc.) have been described in several RE books [3]-[7]. However, elicitation still remains problematic; missing or mistaken requirements still delay projects and cause cost over-runs [8]. No firm definition has matured for requirements elicitation in comparison to other areas of RE, although most authors agree that elicitation covers identifying stakeholders, fact gathering, collecting requirements in diverse forms (e.g. problems, goals, features, aims, etc.), prioritising and recording them. Elicitation and requirements analysis share an ill-defined boundary, necessarily so, since to gather information involves understanding it to determine its worth.

A model of elicitation technique selection [1] proposed matching techniques to the RE situations composed of facets of the domain, maturity of requirements, stakeholders and organisations. This model was used as a framework for evaluating CSEM (the Collaborative Software Engineering Methodology) and interviewing RE experts, reporting that they tended to pick and mix techniques flexibly according to the domain, and adapted elicitation techniques as the RE process progressed. However, no comprehensive set of matching rules connecting techniques to contexts was reported. In a meta-review of elicitation papers [9], no advantage was found for other techniques over semi-structured interviews; use of representations (models, prototypes) did not appear to help, although the empirical evidence in the reviewed papers was limited. Couglan and Macredie [10] compared elicitation techniques in Soft Systems Methodology, JAD, and Participatory Design against a framework of user designer roles, communication activities and techniques including interviews, prototyping, cognitive (protocols), contextual (ethnography), group workshops, and model-driven representations, concluding that collaborative, dialogue-based methods which included workshops were more effective. More specialist techniques, with origins in knowledge engineering, such as card sorts, AHP (Analytic Hierarchy Process) and laddering were compared by [11], who argued that techniques could be matched to particular elicitation problems such as card sorts for groups of similar requirements, laddering for goal decomposition, etc. Domain knowledge improves knowledge elicitation, particularly in interviews, although it also has disadvantages as domain expertise may encourage tacit knowledge omission [12]. Knauss [13] reviewed elicitation techniques from the point of view of communication with end users and the use of multimedia, pointing out that new opportunities are emerging through social media and netography (Internet logging and requirements capture).

Apart from the emergence of Internet-based RE, it appears that requirements elicitation is not amenable to much improvement; however, in this paper we set out to explore the field from the perspective made famous from Donald Rumsfeld's quote of the "known knowns, the known unknowns, and unknown unknowns". This perspective poses challenges to requirements elicitation since it probes the boundaries of knowledge and who possesses it, thereby creating a useful stress test for requirements elicitation techniques, methods and tools. In the following sections of this paper, first we elaborate the unknowns definition to create a structural *Elicitation Review Framework* (ERF) for the review.

978-1-4673-5765-4/13 © 2013 IEEE

RE 2013, Rio de Janeiro, Brasil
Research Track

Subsequent sections analyse elicitation techniques and tools using the ERF, leading to a gap analysis where further research may be beneficial. The paper concludes with a road map for future requirements elicitation research using the ERF as well as placing the suggestions in the context of green-field (new applications) or brown-field (evolution of existing applications, product lines) RE.

II. Elicitation Review Framework (ERF)

Gacitua et al. [14] formalised the Rumsfeld taxonomy within a proposed Tacit Knowledge Framework using the properties of *expressible*, i.e. known knowledge; *articulated*, as documented known knowledge; *accessible,* which is known but not in the foreground of the stakeholder's mind and therefore a memory recall problem; and *relevant* to the project and domain. This produced definitions for:

- *Known knowns*: expressible, articulated, and relevant.
- *Known unknowns*: not expressible or articulated, but accessible and potentially relevant.
- *Unknown knowns*: potentially accessible but not articulated.
- *Unknown unknowns*: not expressible, articulated or accessible but still potentially relevant.

The precise meaning of the above is context dependent, but for simplicity here we take the analyst's perspective of an analyst/user-stakeholder dialogue. With this in mind, known knowns are clearly not a problem; while known unknowns pose a process problem, since the analyst is aware of the type of required knowledge and is faced with the problem of eliciting it from a stakeholder who may be unaware of it or have forgotten it.

Unknown knowns are knowledge held by the stakeholder and accessible to them, but not articulated; e.g. it might be suppressed for political, social or emotional reasons. Hence it is a considerable challenge to discover this tacit knowledge and then elicit it. Consider the following example of knowledge elicited from the operator of a steel rolling mill looking at a slab of white-hot steel that needs rolling into a steel plate [15] "… sometimes you can sit here and look at it and think, 'that one's going to be a bastard'". An analyst trying to understand the requirements for a new generation of rolling mills would need to know why a slab may be a 'bastard', and how the operator can tell. Once the operator articulated this glimpse of his knowledge, it became a known unknown to the analyst; clearly there was some important knowledge that the analyst did not already know. The key was to elicit the how and the why.

Unknown unknowns present the most severe test, in which the analyst and stakeholder are unaware of the missing, but relevant, knowledge; it isn't accessible to either actor. This might be caused by a lack of domain knowledge on both sides or inadequate design exploration so the user-stakeholder is simply unaware of the technical solution possibilities. To illustrate: consider a flood warning system implemented as a wireless sensor network (WSN) [16]. In one deployment, the gateway node, implemented as a GSM uplink and responsible for transmitting data off-site, was positioned next to a small building housing a pump. When the river flooded, the pump started up and the WSN lost its connection to the outside world because of the electromagnetic interference emitted by the pump. The phenomenon of EMI was not understood by the system developers or their hydrologist colleagues; it was an unknown unknown due to insufficient domain knowledge. More pernicious examples of unknown unknowns occur when even the best domain knowledge is incomplete, e.g. in companies developing products for markets that are volatile and subject to changing fashion, social expectations, the unintended consequences of new technologies and so on.

This 'over-the-horizon' knowledge, we contend, is the greatest extant challenge to requirements elicitation and motivates the following review. The Tacit Knowledge Framework [14] poses three challenges:

(i) *Identifying* tacit knowledge: the unknown knowns; even when the analyst suspects they exist (known unknowns), making tacit knowledge accessible may not be easy.
(ii) *Knowing* what is relevant and should be articulated from the analyst's perspective: the necessary detail problem.
(iii) *Articulating* the knowledge where it is needed, in the correct context, so it can be understood by all stakeholders.

These challenges also form part of our review framework. To make these challenges meaningful we also need to consider their relevance to unknown unknowns, since these are not a phenomenon of tacit knowledge. For unknown unknowns, neither the analyst nor the user-stakeholder can identify that there *is* missing knowledge, far less identify what the missing knowledge is. A first step to resolution of the problem is recognising that there may be missing knowledge, and being prepared to invest resources in finding out if there is, and if so, where the gap lies. Articulating the knowledge once identified may be easy or hard but the presumption must be that it is possible to be articulated, if found. Once found, an unknown unknown should be capable of evolving to a known known, provided it is articulated, although of course it may have a value that discourages making it known (e.g. to business competitors).

We will review a selection of requirements elicitation/analysis techniques, approaches and tools using the ERF to assess how well techniques address the knowness problems, and how representations and models might augment techniques and support articulation; then how tools give further support either as editors for representations or more active tools for discovering unknown requirements. The ERF is used to elucidate properties of elicitation techniques, representations, etc., rather than to score them for effectiveness or appropriateness since our objective is to investigate how the different aspects of requirements elicitation contribute towards requirements understanding and thereby identify gaps in the state of the art for future research.

A second part of the ERF uses Clark's Theory of Common Ground [18], to evaluate elicitation techniques and representations in terms of human conversation. Common ground (CG) explains how meaning is constructed by conversation and action, which progresses towards a mutually agreed goal; the *Action Ladder* and *Project* in Clark's terms. Meaning in conversations has different layers: the surface layer of explicit expression, then layers of tacit meaning which rely on deeper understanding of metaphors, and linguistic interpretation of puns, irony, jokes and fiction. Conversations take place in a *Setting*, i.e. in a specific location and time, and are associated with knowledge held by the participants of the conversations: the *Arena* of shared knowledge about the culture, norms, history and assumptions which allow dialogue between people to be interpreted in their context. An extension of the theory [19] provides desiderata for the communication channel used, such as co-presence/visual/audio modalities, sequential/concurrent exchanges, persistence of content, and whether it can be revised and edited. Rich media (visual and audio), concurrent conversations are more effective for constructing understanding through dialogue; whereas persistent media, sequential exchanges and revision/editing facilitate reflection and analysis between dialogue sessions. Clark's theory suggests criteria probing the nature of the analyst/user-stakeholder dialogue, the ambit of knowledge covered (Arena/Setting) and the representations used.

The Tacit Knowledge Framework [14] treats Common Ground as knowledge that is Accessible to both the analyst and the user-stakeholder. It makes no presumption about whether the CG is Expressible, Articulated or Relevant, but recognizes that identifying CG knowledge and then knowing whether it needs to be Articulated are problematic. Domain knowledge is an inadequate proxy, because unless acquired in a common Setting and Arena, peoples' knowledge and perception of the domain may differ; a degree in metallurgy will not guarantee that the analyst understands why some steel slabs are 'bastards' to roll.

For our purposes, Common Ground contributes 'tools for thought' which can address the unknowns problems. For example the Arena suggests questions to discover more details of the users' background which may uncover unknown concerns (viz cultural, political issues); alternatively the concept of joint action suggests cooperative exploration of requirements via prototypes, and questions about user-system interaction. Tracks and meta-level discourse could be developed as heuristics for managing elicitation dialogues. Developing heuristics and guidelines for applying CG to the unknowns part of our ERF forms a research topic in the road map in the concluding section of this survey.

How to deal with unknown unknowns is itself a known unknown that draws on different disciplines so we have not attempted to do a systematic literature review. Instead, we surveyed the literature using the ACM digital library, IEEE Xplore and DBLP databases to follow authors' publications, supplemented with other sources for (e.g.) psychology material. Our criteria for including papers were a combination of citations, our own knowledge of the literature and feedback from RE'11's panel on tacit knowledge.

III. Requirements Elicitation Techniques

As noted earlier, the basics of interviews, observation (ethnographic techniques), case studies, prototypes and model-based elicitation have been established for a number of years. Although there appears to be no advantage in any one technique over structured interviews [1], there may be advantages in combining techniques. For instance, the use of scenarios and prototypes is a frequent combination that helps elicit design improvements as well as basic requirements [20]; furthermore, combination of design rationale, scenarios and prototypes produced some evidence of improvement over single-technique elicitation [21].

A. Unknowns Discovery

Observation using ethnography might have an advantage in eliciting tacit knowledge since the combination of contextual long-term observation and interviews produces rich contextual descriptions. Such descriptions may provide memory prompts to improve accessibility; furthermore, many ethnographic accounts report discovery of serendipitous knowledge and hence unknown unknown requirements [22]-[24]. However, the power of ethnography comes at the penalty of resources necessary for long-term observations, and the sampling to detect unknown unknowns is often a matter of luck. Table I summarises the affordances of basic elicitation techniques from the unknowns perspective.

TABLE I. Potential of Elicitation Techniques for Discovering Unknowns

Technique	Known-unknowns	Unknown-unknowns	Articulation
Interviews	Depends on plan	Follow-up questions, sample size	Natural language ambiguity
Observation	Duration and context plan	Duration and context	Ambiguity in interpretation
Workshops	Plan and composition	Number and composition	NL ambiguity
Protocols/ dialogues	Plan and analysis codes	Limited potential	Narrow, detailed analysis
Scenarios	Plan and sample scope	Sample size and diversity	Sample and bias, NL ambiguity
Prototypes	Design variations	Limited potential	Extent of implementation.

For many techniques the ability to detect the known unknowns depends on the analyst's plan and the sampling strategy. *Scenarios* are economic to collect, and hence may have a slight advantage over other techniques, while interviews provide more flexibility to refine plans during sessions. *Protocols/dialogues* and *prototypes* are least effective since they are more resource intensive, so articulation of requirements is limited by the number and variety of prototypes. *Scenarios*, *interviews*, *workshops* and *observation* have some potential for discovering unknown unknowns but all depend on the sample size and diversity as well as duration for *interviews* and *workshops*. Since sampling is blind, by definition with unknowns, it is difficult to scope the resource necessary to explore tacit knowledge. *Interviews*, *workshops* and *scenarios* all depend on communication by natural language, which runs the risk of ambiguous interpretation.

Observation may also suffer from ambiguous interpretation as users motives are hidden and observer viewpoints may differ. While *protocols* and dialogue analysis produces more detailed articulation, it is resource intensive and hence has a narrower scope of application.

Creative elicitation approaches may incorporate prototypes, scenarios and workshop techniques. They have adapted traditional creativity approaches, such as Creative Problem Solving and the KJ method [25], [26], with probes to stimulate ideas, dialogue management for developing, reflecting and collating ideas, as well as tools and environment support to support user generation of new requirements [27], [28]. These techniques directly address unknown unknowns in the sense that the target product is usually only partially known; however, creative approaches may be more appropriate to new or green-field domains where few requirements exist. Creativity techniques' power for eliciting tacit domain knowledge is less sure, although tool support may help to capture tacit knowledge of domain constraints. Other additions to the catalogue of elicitation techniques are the use of role playing in enacted scenarios [29], and variations of ethnography and prototyping in the cultural and technology probe tradition. Here, a designed artefact is placed in its expected context of use, such as a home setting, and then user interaction is observed in order to derive insight into the design and further requirements [30], [31]. Both of these approaches are labour intensive in creating materials and analysis; furthermore, they suffer from the sampling problem and offer little advance on other techniques, although technology probes do address the evolution problem.

B. Establishing Common Ground Through Elicitation

The affordances of elicitation techniques from the CG perspective are given in Table II. Interviews approach natural human conversation so they support the dynamic construction of understanding and exploration of the Arena and Setting of the dialogue by explicit questions and observation. However, interviews are weaker for reflection, where the analyst has to rely on notes and recordings. Workshops are natural multi-party conversations so the complexity of constructing shared understanding is a function of the number of participants and the workshop organisation, e.g. facilitation, moderation, etc. Reflection is via recording and notes. Both interviews and workshops are limited by the sample of participants, number of sessions and duration. Furthermore, eliciting tacit knowledge depends on the analysts' questioning skills. Observations have the virtue of co-presence so non-verbal and contextual cues can be analysed, but the analytic process is primarily sequential and no dialogue is possible since the analyst-observer plays a passive role. Reflection is better as analysis is based on video recordings; however, the scope of contextual information is limited to the visible information. Protocols are 'think aloud' stylised monologues which are subsequently analysed in depth, so this sequential process supports reflection rather than dynamic construction of CG. Scenarios are a sequential technique since there is a gap between creation/capture and analysis. Reflection is effectively supported since the concrete examples stimulate thought, but the downside is the size and diversity of the sample, which can lead to biases and omissions. Finally, prototypes are an interactive visual artefact that is embedded in an interview-style session as a walkthrough, demonstration or hands-on testing. This technique is semi-concurrent, since the prototype has to be prepared beforehand, but it is good for encouraging reflection, since interaction with the artefact is even stronger than dialogue for developing a shared understanding between analyst and stakeholders about the artefact.

TABLE II. POTENTIAL OF ELICITATION TECHNIQUES IN THE CG [18,19] FRAMEWORK

Technique	Conversation media	Concurrent/ sequential	Reflection	Scope-Arena/Setting
Interviews	Co-presence Dynamic dialogue	Concurrent	Only via notes/ recordings	Good but depends on participants & time
Workshops	Co-presence Dynamic dialogue	Concurrent	Only via notes/ recordings	Depends on participants sample & time
Observation	Co-presence/ video	Sequential for analysis	Good for review	Location context only
Protocols/ dialogues	Stylised dialogue	Sequential for analysis	Detailed review	Limited
Scenarios	Text, images	Sequential	Good but depends on sample	Depends on participants sample
Prototypes	Visual media -artefact	Semi concurrent	Good but depends on extent	Limited – ecological evaluation

From our experience and the evidence in previous surveys [1], [9], [10], interviews and workshops are more effective for tacit knowledge elicitation among the basic techniques since they approximate to natural conversations; however, most techniques are used in conjunction with representations and models. Scenarios have the merit of grounding examples in concrete reality to stimulate understanding and questions; similarly, prototypes establish CG in the design or solution space, by stakeholders evaluating the consequences of interaction. Hence a combination of techniques rather than structured interviews *per se* is probably the most effective approach. This concords with our experience [32] and the approach adopted by experts [9].

IV. MODELS AND REPRESENTATIONS

Simple representations accompany most elicitation techniques, i.e. natural language lists of requirements, sketches, scenario texts, as well as artefacts such as mock-ups, storyboards and prototypes. This section focuses on structured representations as models recorded either as diagrams or formal specification languages. As the diversity of requirements modelling languages is vast only a limited review will be conducted to illustrate the potential for model-based augmentation of elicitation techniques. Models may be passive representations designed for inspection-based analysis, or more formal models which are integrated with model-checking and reasoning tools. Active tool support is dealt with in the next section; in this part we investigate the scope of the model's semantics and how it addresses different types of requirements knowledge. Model-based techniques range from simple use cases used in conjunction with scenarios [33], to formal goal-oriented techniques with obstacle/barrier analysis, such as KAOS [34], goal models with skills-preferences trade-offs [35], and the rich agent-relationship dependency semantics of i* [36], which extends domain modelling. User-oriented representations have been proposed based on activity theory and UML models as a means to guide elicitation with a subject-object-activity-outcomes framework [37].

A summary of the semantics of a limited sample of RE models is illustrated in Table III.

TABLE III. REPRESENTATIONS & MODELS FROM THE PERSPECTIVE OF TYPES OF RE KNOWLEDGE

Model	Goals/Reqs.	Req. Spec.	Domain Knowledge	Articulat./ Accessib.
Use cases [33]	Implicit goals	Action scripts	Limited- ext entities	++
Volere [4]	Goals, req. statement	Rationale, ownership	Org. & env. context	+++
KAOS [7]	Goal hierarchies	Object processes	Obstacles, domain assumptions	-
i* [39]	Goals, softgoals	Agents, tasks, resources, relationships	Roles, agent attributes, org. Setting	+
ISRE [40]	Not explicit	Object processes	Env. Setting, spatial location, org. Setting	++

Use cases represent the baseline in requirements representations, with goals being implicit in the context diagram, although they may be explicitly stated with limited action pathways in the text format. Little domain knowledge is recorded apart from external entities, although use cases can be augmented by scenarios to provide contextual detail [33]. The Volere template [4] is a widely adopted means of recording requirements, with goals and requirements statements being represented with rationale for the requirement, stakeholder ownership, prioritisation and contextual detail of the system environment. However, specification is a limited representation since no model is used.

Goal-oriented requirements engineering (GORE) [38], [41] organises goals in hierarchies. GORE representations are typically complemented by a variety of specification models giving details of agents, objects, actions, events and processes, which may be used for tool-based model checking; for example, the skills preferences approach [35] that matches agents' skills and preferences against properties of goals.

KAOS [7] extends GORE approaches with obstacles to the realisation of goals, thus providing a stimulus to refining requirements specification. Specification knowledge is represented in detail by a formal language for model checking and temporal reasoning, which can discover known unknowns. Domain knowledge is represented in the form of obstacles and domain assumptions, both of which serve to help identify environmental constraints on goals achievement and help enrich decomposition. Like KAOS, i* [39] also provides a rich representation, with goals, softgoals, agents, tasks and resources. Inter-agent relationships are modeled as dependencies (to satisfy some goal, or provide some resource), while goal decompositions model how agents achieve the goals for which they are responsible. i* is particularly strong at stimulating the exploration of alternative goal satisfaction strategies using the degree to which alternatives contribute to softgoal (typically used to model qualities) *satisficement*. Model checking can be carried out on matching agents to goals, within the constraints of resources, capabilities, etc., using the Tropos knowledge representation language and reasoning support [42], [43]. Domain knowledge is implicit in the i* strategic dependency model as social aspects of the system are modelled, i.e. organisation, agents, roles, capabilities; however, no explicit details of location or environmental conditions are recorded.

ISRE [40] describes a representation framework for organising and generating scenarios rather then requirements *per se*; however, it provides a pertinent contrast since it focuses on domain knowledge with details of organisation, spatial location, environmental conditions, with a range of normal and exceptional events. Domain knowledge may include descriptions of the spatial environment and ecological context of the system, e.g. weather and environmental events [40]. The boundary between domain and specification knowledge is inevitably blurred as requirements analysis progresses towards design, so the extent to which domain knowledge is routinely collected to inform design is an open question for future research. The ISRE approach is based on domain knowledge taxonomies in safety-critical systems engineering [44], and uses environmental information for simulation and model checking with Bayesian reasoning to assess the probabilities of adverse impact from environmental factors on system performance, goals and operations [45]-[47].

Representations in natural language as lists or in formatted tables, such as scenarios and the Volere template, tend to be easier to understand for all stakeholders, hence they are more accessible. The action-object script of use case pathways is slightly less accessible as are complex diagrammatic notations. Narrative scenarios which constitute the main representation in ISRE with informal diagrams that organise scenarios in conceptual maps may be easy to understand but sacrifice precision since users can experience difficulties in finding appropriate details as the number of scenarios increases. Goal trees use a diagram format that is understood by most end users; however, richer languages such as i* and more formal languages such as KAOS are not accessible to all. Even though KAOS does employ goal trees as a bridging representation there is still the understanding gap between informal diagrams and formal languages which can only be bridged by natural language explanation tools.

From the CG perspective, use cases and GORE focus only on the basic subject matter of the analyst-stakeholder conversation, i.e. goals, requirements and the system. KAOS partially extends the Setting by including domain knowledge in the form of obstacles and assumptions. i* has a richer model of the stakeholder's Setting in the agent-dependency model. As models capture more Arena/Setting domain knowledge they extend Jackson's [50] framework of the relationship between domain knowledge dependencies and assumptions and the requirements specification. An ongoing research challenge is to establish the analytic value of richer domain modelling semantics.

The trade-off for representations lies between parsimonious models which are easier to understand, hence easier for articulation, and more comprehensive representations that cover more aspects of the domain but at a cost of adding too much confusing detail [51]. Models for inspection-based requirements analysis run into a complexity limit, so to realise the worth in richer models semi-automated analysis is necessary.

Representations also include design rationale, which records the results of decisions and arguments supporting requirements analysis in a simple diagram format linking goals to options to realise a goal, and arguments supporting or detracting from goal achievement [52], [53]. Design rationale has been integrated with other representations such as scenarios and prototypes [21]; however, uptake by industry has been limited [54] even though this genre of representation is simple and easy to articulate.

In spite of the plethora of models in RE, evidence of industrial uptake is limited. Simple template representations such as Volere and informal means of recording knowledge (text, lists, sketches, simple diagrams) appear to be the accepted practice [9]. i* has made some limited progress towards acceptance in industrial practice [55], but for most requirements elicitation, representations remain simple and informal. This probably reflects the low cost of creating simple representations that produce sufficient benefit. The power of representations and models for discovering unknowns depends on the scope of their semantics and how comprehensively the problem space is covered; but this articulation comes at the price of accessibility. Larger and more comprehensive models, which represent more of the Arena and Setting in CG terms, are more difficult to inspect. In the CG perspective the role of models depends on how well they are integrated into the analyst-stakeholder dialogue, either by manual processes such as walkthroughs, interaction such as pointing and annotation, or by automated analysis.

We now turn to the prospects for support tools which have the potential to mitigate higher costs of constructing model-based representations by returning additional benefit in automated checking and refining requirements.

V. Tool Support

Tool support for elicitation can be divided into natural language (NL) tools that process requirements text and documents; model-based tools, i.e. checkers and reasoners; and more general support tools including the recent genre of socially oriented collaboration tools. Standard requirements management tools, e.g. DOORs, are not reviewed since these form the baseline of current industrial practice.

A. Natural Language Tools

Two directions have emerged in NL-oriented tools. First are ontologies to support inspection-based elicitation and refinements such as lexicon-based tools [56], [57], and more formal domain ontologies [58] which have been developed to support web services. Ontology tools afford simple model checking of terms for consistency, detecting conflicts, etc., as well as supporting matching of requirements terms to a domain ontology via semantic lexicons such as WordNet. More powerful linguistic tools detect potential ambiguities in requirements statements, helping to refine requirements by eliminating conflicting or vague statements [59]-[61]. The automatic extraction of requirements from NL text is impossible, but there has been some success in the identification of key domain abstractions from NL documents, which need not necessarily be explicit in requirements documents [17], [62]. This class of tool is useful for helping the analyst's understanding of the domain; for example, by triggering human investigation of why the tool infers the importance of a particular abstraction. It may also aid construction of a domain ontology and so act to aid the accessibility of domain knowledge [63].

The second direction has been to use text-mining tools on structured sets of requirements documents to cluster and categorise similar requirements from different sources and then use stakeholder ownership details to prioritise requirements [64], [65]. Text mining and information retrieval technology enable existing sets of requirements to be categorised, while recommender systems support the process of requirements negotiation and prioritisation by grouping requirements according to ownership and similarity.

Apart from lexicon-based approaches, natural language tools do not support elicitation directly; instead they assume a corpus of existing documentation, hence text-mining technology is more appropriate for brown-field evolutionary RE in product-line domains. Where extensive requirements documentation exists in product-line domains, tools can automatically aggregate requirements and produce optimised 'bundles' of requirements for different product releases using evolutionary computing algorithms [66].

In conclusion, language-based tools show considerable potential for further development to address refinement of known unknowns and articulation by removing ambiguities and inconsistencies in requirements texts. Support for the unknown unknowns needs more development; however, inspections of ontologies and lexical lists, and exploratory text mining might stimulate consideration of over-the-horizon requirements.

B. Model-Checking Tools

Most of the models reviewed under representations have support tools which use a formal specification language. The review concentrates on semantic model checking rather than tools oriented to lower-level testing for reachability and consistency, e.g. SPIN. Goal-oriented requirements can be checked for missing requirements and conflicts between obstacles and goals [7], [67], while the GRAIL tool augments KAOS specifications to provide model checking of goals and obstacles, refinement of specifications, and detailed checking of action pathways to goal realisation, temporal conditions, consistency, etc. [68]. Model checking for i* employs variants of the Tropos specification language to reason about consistency and matches between agents' capabilities and goal properties [43], [48]. Other forms of model checking assess pathways to attaining high-level goals from relationships among subordinate goals, allocation of agents to goals by matching requirements to their skills, preferences and abilities [35], as well as trust and security properties of models [69]. The ISRE model [40] is not supported by model checking; instead, probabilistic reasoning is used to evaluate attainment of softgoals such as reliability and safety, for a set of agents, processes and environmental properties. In this case the tool simulates system performance under a variety of environmental conditions and agent/process choices, to discover optimal requirements for a set of conditions [45]. A similar optimisation tool, DPP, reasons over requirements goals, obstacles and mitigations (to remove obstacles) to derive the best set of requirements for a given set of environmental events and constraints [70], [71].

Model-checking tools continue to develop as further specialisms evolve; for instance, self-aware, adaptive systems have become a prominent focus; and models and model checkers have emerged to determine the match between requirements for monitoring and environmental events and states, suggesting how adaptations can be matched to information and its source in the environment [72]-[74].

Most model checkers rely on a semi-formal interface in which the specification language is translated into a more user friendly notation or natural language to help articulation [75]. Another approach to formal model checking integrates execution and assessment of models with animations of system operation so users can inspect operations to validate their requirements as well as receiving feedback on possible problems with the specification [76]-[78].

Model-checking support tools can only address known unknowns since the model has to be directed by a known agenda. However, run-time monitoring [79], [80] which is emerging as a technique to support requirements-aware systems [73] offers the prospect of collecting data that, if mined and appropriately analysed, might reveal unknown unknowns. Articulation is also supported as requirements become more complete, correct and consistent. Furthermore, unknown unknowns can be tackled by creativity support tools, such as i-require [28], a mobile application which allows users to record requirements notes on the move; and the requirements presenter [103] which collates requirements from different team members to stimulate discussion and elaboration of ideas. Other creativity support tools implement processes from traditional brainstorming methods to collect ideas, collate and categorise requirements, and encourage reflection by structuring information exchange dialogues [26].

C. Social Collaboration Support Tools

A recent development has been tool support for the social process of requirements elicitation, which harnesses human collective effort rather than relying on models or existing documents. Social collaboration extends Common Ground to multi-party conversations by social networks (Arena and Setting awareness), and by capturing and integrating the opinions of many over time (action ladder). Social collaboration also stimulates awareness of knowns from both the requirements engineering and user perspective by information sharing. Some of these tools facilitate elicitation of requirements and solutions by crowd sourcing over the Internet [81]-[83], while others are integrated with social networks and assume existing requirements documents. Social networking with information exchange analysis has been used to support collaboration among distributed teams of analysts [84]. The StakeRare tool [85] goes further by integrating stakeholder social networks with collaborative filtering and a recommender system, so when requirements have been documented in a community, the process of aggregation, prioritisation and agreement is supported. Limited support for unknown unknown requirements is provided by data-mining techniques.

In conclusion, social collaboration tools have considerable potential for new approaches to elicitation; although crowd sourcing does depend on terms of reference for small well-defined problems [86], or considerable search, matching and integration problems when requirements are realised for heterogeneous components as in Open Source Software [87].

VI. State of the Art and Research Road Map

In this section we review the prospects for improvement in requirements elicitation.

A. Techniques

The complement of techniques has remained relatively stable over several years, apart from the emergence of creativity-oriented approaches. Elicitation techniques have also remained stable in other domains, such as knowledge acquisition [88], so the prospects for new techniques arising may be limited. Nevertheless, crowd sourcing and collaborative solutions combined with participatory design [82] may have potential for application to RE.

Linguistic techniques deserve further application, such as Clark's Common Ground Theory [18] which emphasises the process of building mutual understanding in discourse, exactly the elicitation problem. Clark's theory could be applied as a metaphor for understanding the dependencies between software machines and their environment. The value in this theory lies in the structure of the Arena, Setting and discourse model which could be applied in personal, contextual models of dialogue as a constructed process, which suggest guidelines for knowledge elicitation. Patterns for requirements elicitation dialogues [89] could benefit from a more theoretical origin. Another possible direction for techniques intersects with representation and tools: exploring the role of simulation and elicitation techniques. Construction of simulations can be resource intensive, so this technique, relatively unexplored in RE, may be better suited to complex, brown-field domains. Simulation can harness computation power to explore known unknowns, but may also be combined with interactive graphical worlds (e.g. SecondLife) to help elicit unknown knowns. Virtual worlds can facilitate anonymous interaction and hence overcome emotional and political fears.

As more applications become global and web oriented, remote elicitation techniques will become more prominent [13]. Capturing feedback by forms, questionnaires and logging interaction with prototypes are standard techniques; however, there is potential for richer media approaches using voice, video and virtual co-presence. Common ground indicates richer media in concurrent dialogues, and coupled with more persistent, shared documents (e.g. wiki requirements elicitation) may be an effective combination. These approaches may be scaled up by collaborative crowd sourcing over the web, although some caution needs to be exercised about how such requirements capture can be structured effectively. The social implications of authority and power [91] need further attention not only in collaborative elicitation but also in resolving tensions between user participation and designer's authority in producing solutions.

B. Representations and Model-Based Tools

Representations are being extended with more socio-technical domain knowledge; for instance, power and authority influences on requirements [91]; the contribution of stakeholder values and motivations towards user requirements and preferences [92]; and investigating emotions to understand users' reaction to requirements and systems acceptance [93]. A focus on people-oriented issues may help to elicit the knowledge category of unknown knowns when people suppress what they know for political reasons or may be unwilling to articulate requirements because of clashes with their values and emotional reactions. Making such knowledge accessible requires the personal communication skills and 'emotional intelligence' much in vogue in the business literature; making these issues explicit can facilitate articulation. Richer socio-economic concepts may stimulate elicitation from inspection of dependencies across the domain system boundary, but inspection-based approaches will always suffer from the increasing complexity of notations [51], so tool support either as hypertext advisors [94] or more active model-based reasoning will be necessary to realise the benefits of more complex representations. Representations can address the articulation and accessibility problem; however, by their nature they cannot deal with evolution of requirements over time, although version-control tools may provide part of the answer.

One direction for model-checking tools is increasing the scope for reasoning about assumptions [95], so dependencies between assumptions made about domain knowledge, requirements and stakeholder preferences are questioned. More generally the prospects for model-based tool development depend on solving the dilemma of the cost of model construction versus the pay-back from improved checking and requirement refinement. Model-based reasoning needs to be integrated with simulation tools, so the model not only supports refinement of known unknowns (i.e. consistency, completeness and validity) of requirements, but also design exploration (towards unknown unknowns). With probabilistic reasoning some known unknowns may be discovered by reasoning over a wider range of domain knowledge, as well as the performance implications. Non-functional requirements of different designs can be explored from the perspective of different environmental conditions. Where the solution space is relatively well understood, evolutionary computing algorithms can be applied to optimise product requirements, as has been demonstrated for product lines [66] and design selection for optimal non-functional requirements [45].

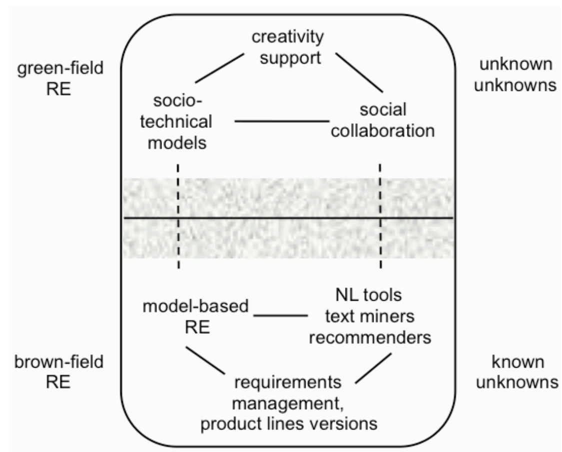

Fig. 1 Road map for future requirements elicitation in the perspective of brown- and green-field domains. The shaded boundary acknowledges that systems are rarely 100% 'green or brown field.

More sophisticated model checking of dependencies in domain models and specifications is being explored in extensions to the i* language, such as social considerations of trust, privacy and agent responsibilities [69]. The focus on personal requirements for applications in persuasive technology, accessibility, or customisation is another area for future development, where matching between characteristics of individual users, their goals and technical solutions [96] needs to be explored. Self-aware and adaptive systems [73], [90] extend this theme since the focus of adaptation is frequently the 'human in the loop'. Reasoning and model checking for such systems need to consider embedding more sophisticated models of the domain phenomena; for instance, rather than reasoning about human properties of attention, understanding and attitudes, the system embeds a model of human cognition which can predict these properties from monitored elements of the environment.

Natural language tools have further potential as more sophisticated text-mining algorithms are applied to elicit requirements semi-automatically, possibly extended by directed learning techniques, so once a domain expert has marked up the relevant parts of requirements documents, machine learning can be applied to discover the unknown unknowns in similar, analogous document sets.

The road map for future research is shown in Fig. 1. In green-field applications, we suggest the way ahead is to improve creativity support tools [55] and make extensive use of the growing number of Internet-based ontologies, so the unknown unknowns may be discovered by augmenting human imagination. These tools need to be placed in a social context to harness the creative power and critiques of many stakeholders. A further extension is to create or harvest libraries of scenarios from analogous domains which could then be text mined to discover the unknowns. Automatic generation of combinations of different scenario facets (e.g. environmental conditions, locations, people, even cultures) with a variety of solutions, which then form the subject matter for goal-directed text mining, is another possible area of future work. The solid lines between creativity tools, socio-technical models social collaboration hint at the value of integrating these tools and approaches, while dashed lines suggested other possible connections.

In brown-field sites there are many opportunities for eliciting known unknowns, since requirements documentation exists in product lines and domains with evolving requirements. Here the quest will be to improve the articulation of requirements and make them more accessible among stakeholder groups. Information retrieval, cluster analysis and text-mining approaches are already being applied to aggregating and matching existing requirements for product releases [66], [85], [87]. Augmenting these approaches with social network analysis and recommender systems facilities prioritisation and negotiations among stakeholder communities. One future prospect is the increased application of machine learning in brown-field domains, where training data exists, to improve text- and data-mining algorithms with expert mark-up to capture requirements criteria that can be learned and applied to any analogous data set. Evolutionary algorithms have only just started to be applied to optimise solutions for different environments, domains and markets. These tools may effectively support the time dimension in brown-field domains by assessing changes in versions over space and time. Further investigation may be necessary to improve the articulation of requirements for new product lines at the strategic level, and single versions at the tactical level [49], to integrate layers within continuous requirements elicitation. The prospects for the unknown unknowns in product lines is less sure, since there is a natural conservativism when the requirements solution space is relatively well known, to be satisfied with variation points rather then radical changes in requirements. The tension between paradigm-shifting applications and traditional product lines will be a considerable unresolved challenge for the future.

Another perspective for future research on the unknowns is illustrated in Fig. 2.

There are four research directions to push the boundaries of the unknowns.

1. *Unknown knowns*. The problem for the analyst is discovering what the stakeholder knows but does not articulate. Sensitivity to political issues, user values and emotions needs to be researched in depth to provide 'emotional intelligence' guidance for analysts so they can anticipate these unknowns and elicit sensitive tacit knowledge. The Common Ground quest is to be more sensitive to the stakeholder's Setting, feelings, norms and culture.

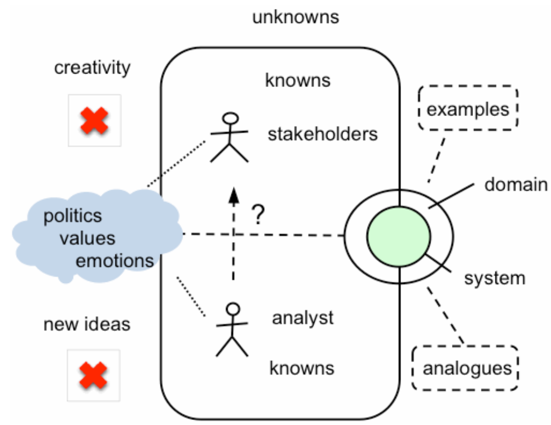

Fig. 2 Road map from the Common Ground-unknowns perspective

2. *Known unknowns*. In this case the analyst has some awareness of the necessary knowledge, so an agenda for elicitation can be set. Most techniques involve exploring the implications of the system-domain boundary, exemplified by Jackson's formulation of the RE problem [50]. Challenging assumptions, reasoning about the implications of obstacles, and relaxing domain constraints need further research towards not only hard influences, but also soft, probabilistic implications. Examples and analogies need to be extended so obstacles relevant to 'this type of domain' can be recognized [100].
3. *Design discovery*. This is a variant of known unknowns where the challenge is to solve the "I'll know what I want when I see it" problem. Much progress has been made in this area with prototypes, storyboards and mock-ups; however, simulations and virtual worlds may have further potential. Clark's theory predicts that conversational goals are realised by action; by analogy, eliciting users' goals will be validated by interaction with designs.
4. *Unknown unknowns*. Two approaches could address elicitation of 'over-the-horizon' knowledge. First, creative RE, which is already established [55], needs to be integrated with social media, so that collaborative creative RE is empowered. This could involve designing socio-technical elicitation systems, as e-communities, communities of practice with global distributions for the increasing number of Internet applications. Secondly, the use of analogies and examples, and also counter examples, can challenge the boundaries of the possible to develop new design ideas.

We are confident that progress can be made and that elicitation of unknowns will become less a matter of chance, as analysts are equipped with the means to identify the types of unknowns with which they must deal, and with the ability to select the blend of techniques needed to turn the unknowns into knowns. In the meantime, given the current state-of-the-art, analysts need to be alert to clues to the potential for unknowns. Thus, unknown unknowns are most likely to exist when the domain is new or the envisaged solution will radically change the way its users work. Here, analysts should invest in prototypes and pilot applications, and use creativity techniques to explore the problem and solution spaces. Clues to the existence of unknown knowns vary, but any moderately complex work process is likely to depend on knowledge that the actors have acquired over a long time and internalized, making it hard for them to articulate it, or easy for them to withhold it. This likelihood can sometimes be confirmed by linguistic clues. For example, a stakeholder who uses the words "obviously" or "of course" when imparting knowledge that is not obvious to the analyst may be withholding more knowledge that they don't realize the value of. Here the analyst may need to develop a detailed and broad range of scenarios, using a range of media (e.g. images and diagrams, as well as spoken or written text) and consider using observation. Known unknowns imply the need for exploration of the problem and solution spaces; (e.g.) using a combination of obstacle analysis and prototyping.

VII. Conclusions

The prospects for advances in requirements elicitation towards tackling the unknown unknowns are considerable, through more sophisticated support tools, extension of natural language approaches with the application of ontologies, social networks and recommenders. Modelling semantics need to become more sophisticated to take account of more domain phenomena; however, this has to be accompanied by advanced model-checking and reasoning tools with more sophisticated probabilistic reasoning as well as logic-based tools. Rapid gains are possible by extending the application of social networking analysis, data and text mining as well as creativity support tools. These technologies exist, so their extension and application to RE should not be difficult.

Simulation and remote requirements elicitation are both growth areas, so in the future there may be seamless gradation of techniques and tools in two dimensions: first, from co-present (same place, same time) RE with traditional interviews, workshops, etc. to remote (different place, different time) RE by Internet requirements elicitation via logs and feedback from interactive prototypes, to video conferencing and stakeholder 'inhabited' worlds for virtual prototyping; secondly, the transition from requirements to prototypes to products may include simulations in complex domains where investment in requirements exploration will pay off. Common ground points towards the benefits of more interactive, rich media, elicitation approaches.

While there is no ultimate solution to the unknown unknowns, beyond harnessing the power of human imagination, it may be worth reflecting on safety-critical systems engineering, where the quest for unknowns is driven by life-critical requirements. Solutions in this area have motivated more sophisticated domain and environmental models with reasoning tools that embed knowledge from other domains such as psychology, sociology and environmental science [44], [97]-[99]. Although progress has been made, many unkowns still remain unknowns even though in retrospect they might have been discovered as knowns, if only domain knowledge and assumptions had been critiqued, as NASA's experience with the Space Shuttle and aircraft accident investigations exemplify [100]-[102]. The motto for future requirements elicitation could well be creative imagination coupled with perspicacity.

References

[1] A.M. Hickey and A.M. Davis, "A unified model of requirements elicitation". *Journal of Management Information Systems,* 20(4), 2004, pp. 65-84.

[2] B. Nuseibeh and S.M. Easterbrook, "Requirements engineering: a roadmap", in *Proceedings ICSE-00*. New York: ACM Press, 2000, pp. 35-46.

[3] D.C. Gause and G.M. Weinberg, *Exploring Requirements: Quality Before Design*. New York: Dorset House, 1989.

[4] S. Robertson and J. Robertson, *Mastering the Requirements Process*. Harlow: Addison Wesley, 1999.

[5] I. Sommerville, and G. Kotonya, *Requirements Engineering: Processes and Techniques*. Chichester: Wiley, 1998.

[6] A.G. Sutcliffe, *User-centred Requirements Engineering*. London: Springer, 2002.

[7] A. Van Lamsweerde, *Requirements Engineering: From System Goals to UML Models to Software Specifications*. Chichester: Wiley, 2009.

[8] *Business Computing World*, http://www.businesscomputing world.co.uk/top-10-software-failures-of-2011/, accessed 1 January 2013].

[9] A. Davis, O. Dieste, A. Hickey, N. Juristo, A.M. Moreno, "Effectiveness of Requirements Elicitation Techniques: Empirical Results Derived from a Systematic Review" in *Proceedings 14th IEEE Int. on Requirements Engineering (RE'06)*, Minneapolis, USA, 2006. pp. 176-185.

[10] J. Couglan and R.D. Macredie, "Effective communication in requirements elicitation: a comparison of methodologies". *Requirements Engineering*, 7(2), 2002, pp. 47-60.

[11] N.A.M. Maiden and G. Rugg, "ACRE: selecting methods for requirements acquisition. *Software Engineering Journal*,. 11(3), 1996, pp. 183-192.

[12] I. Hadar, P. Soffer and K. Kenzi, "The role of domain knowledge in requirements elicitation via interviews: an exploratory study". *Requirements Engineering*, 2012, DOI: 10.1007/s00766-012-0163-2.

[13] A. Knauss, "On the usage of context for requirements elicitation: end-user involvement in IT ecosystems", in *Proceedings 20th IEEE Int. Conference on Requirements Engineering (RE'12)*, Chicago, Il, USA, 2012. pp. 345-348.

[14] Gervasi, V., Gacitua, R., Rouncefield, M., Sawyer, P., Kof, L., Ma, L., Nuseibeh, B., Piwek, P., de Roeck, A., Willis, A., Yang, H.: "Unpacking Tacit Knowledge for Requirements Engineering", in W. Maalej, A. Kumar Thurimella, H. Becker (Eds), Managing Requirements Knowledge, Springer, 2013.

[15] M. Hartswood, R. Procter, M. Rouncefield, R. Slack, J. Soutter, A. Voss. "Repairing the Machine: a Case Study of Evaluating Computer-aided Detection Tools in Breast Screening", in *Proceedings Eighth European Conference on*

Computer-Supported Cooperative Work (ECSCW'03), Helsinki, Finland, 2003. pp 59-75

[16] D. Hughes, P. Greenwood, G. Coulson, G. Blair, "Supporting Flood Prediction using Embedded Hardware and Next Generation Grid Middleware", in *Proceedings 4th IEEE International Workshop on Mobile Distributed Computing*. Niagara Falls/Buffalo, NY, USA, 2006.

[17] R. Gacitua, P. Sawyer and V. Gervasi, "On the effectiveness of abstraction identification in requirements engineering", *in Proceedings 18th IEEE Int. Conference on Requirements Engineering (RE'10)*, Sydney, Australia, 2010. pp. 5-14.

[18] H.H. Clark, *Using Language*. Cambridge: Cambridge University Press, 1996.

[19] H.H. Clark and S.A. Brennan, "Grounding in communication", in L.B. Resnick, J.M. Levine and S.D. Teasley (eds.), *Perspectives on Socially Shared Cognition*. Washington: APA., 1991, pp. 127-149.

[20] J.M. Carroll (ed.) *Scenario-based Design: Envisioning Work and Technology in System Development*. New York: Wiley, 1995.

[21] A.G. Sutcliffe, and M. Ryan, M. "Assessing the usability and efficiency of design rationale", in *Proceedings INTERACT-97*. London: IFIP/Chapman and Hall, 1997, pp. 148-155.

[22] J. Gougen and C. Linde, C. "Techniques for requirements elicitation", in *Proceedings 1st International Symposium on Requirements Engineering*. Los Alamitos CA: IEEE Computer Society Press, 1993, pp. 152-164.

[23] D. Martin and I. Sommerville, I., "Patterns of cooperative interaction: linking ethnomethodology and design". *TOCHI*,. 11(1), 2004, pp. 58-89.

[24] D. Randall, R. Harper and M. Rouncefield, *Fieldwork for Design: Theory and Practice*. Berlin: Springer, 2007.

[25] N.A.M. Maiden, C. Ncube and S. Robertson, "Can requirements be creative? experiences with an enhanced air space management system", in *Proceedings ICSE-07*. New York: ACM Press, 2007, pp. 632-641.

[26] V. Sakhnini, L. Mich and D.M. Berry, "The effectiveness of an optimized EPMcreate as a creativity enhancement technique for web site requirements elicitation". *Requirements Engineering*, 17(3), 2012, pp. 171-186.

[27] C. Mancini, Y. Rogers, A.K. Bandara, A. Coe, L. Jedrzejczyk, A.N. Joinson, B.A. Price, K. Thomas, B. Nuseibeh, "Contravision: Exploring Users' Reactions to Futuristic Technology", in *Proceedings CHI-10*. Atlanta, GA., USA, 2010. pp. 153-162.

[28] N. Seyff, N.A.M. Maiden, K. Karlsen, J. Lockerbie, P. Grünbacher, F. Graf, C. Ncube, "Exploring How to Use Scenarios to Discover Requirements". *Requirements Engineering*, 14(2), 2009, pp. 91-111.

[29] A.F. Newell, A. Carmichael, M. Morgan and A. Dickinson, "The use of theatre in requirements gathering and usability studies". *Interacting with Computers*,18, 2006, pp. 996-1011.

[30] W. Gaver, J. Bowers, T. Kerridge, A. Boucher, N. Jarvis "Anatomy of a Failure: How we Knew When our Design Went Wrong, and What we Learned From it", in *Proceedings CHI-09*. Boston, MA, USA, 2009, pp. 2213-2222.

[31] H. Hutchinson, W. Mackay, B. Westerlund, B. Bederson, A. Druin, C. Plaisant, M. Beaudouin-Lafon, S. Conversy, H. Evans, H. Hansen, N. Roussel, B. Eiderbäck, S. Lindquist, Y. Sundblad, "Technology Probes: Inspiring Design for and with Families", in *Proceedings CHI-03.* Fort Lauderdale, Fl, USA, 2003, pp 17-24.

[32] A.G. Sutcliffe, S. Thew and P. Jarvis, "Experience with user-centred requirements engineering. *Requirements Engineering*, Vol 16(4), 2011, pp. 267-280.

[33] I. Alexander and N.A.M. Maiden, *Scenarios, Stories, Use Cases: Through the Systems Development Life-Cycle*. Chichester: Wiley, 2004.

[34] A. Van Lamsweerde and E. Letier, "Handling obstacles in goal-oriented requirements engineering". *IEEE Transactions on Software Engineering*, 26(10), 2000, pp. 978-1005.

[35] S. Liaskos, S.A. McIlraith,S. Sohrabi and J. Mylopoulos, "Representing and reasoning about preferences in requirements engineering". *Requirements Engineering,* 16(3), 2011, 227-249.

[36] E.S.K. Yu, P. Giorgini, N.A.M. Maiden and J. Mylopoulos, "Social modeling for requirements engineering: an introduction", in E.S.K. Yu, P. Giorgini, N.A.M. Maiden and J. Mylopoulos (eds) *Social Modeling for Requirements Engineering*. Cambridge, MA: MIT Press, 2011, pp. 3-10.

[37] R. Fuentes-Fernández, J.J. Gómez-Sanz and J. Pavón, "Understanding the human context in requirements elicitation". *Requirements Engineering*, 15(3), 2010, pp. 267-283.

[38] A.I. Anton and C. Potts, "The use of goals to surface requirements for evolving systems", in *Proceedings 1998 International Conference on Software Engineering: Forging New Links*. Los Alamitos CA: IEEE Computer Society Press, 1998, pp. 157-166.

[39] E.S.K. Yu, "Social modeling and i*", in T. Borgida, V. Chaudhri, P. Georgini and E.S.K. Yu (Eds), *Conceptual Modeling: Foundations and Applications: Essays in Honor of John Mylopoulos*. Berlin: Springer, 2009, pp. 99-121.

[40] A.G. Sutcliffe, B. Gault, B. and N.A.M. Maiden, N. A. M., "ISRE: Immersive Scenario-based Requirements Engineering with virtual prototypes. *Requirements Engineering*, 10(2), 2005, pp. 95-111.

[41] J. Mylopoulos, L. Chung and E.S.K. Yu, "From object-oriented to goal-oriented requirements analysis. *Communications ACM*, 42(1), 1999, pp. 31-37.

[42] J. Castro, M. Kolp and J. Mylopoulos, "Towards requirements-driven software development methodology: the Tropos project". *Information Systems*, 27(6), 2002, pp. 365-389.

[43] P. Giorgini, J. Mylopoulos and R. Sebastiani, "Goal-oriented requirements analysis and reasoning in the Tropos methodology". *Engineering Applications of AI*, 18(2), 2005, pp. 159-171.

[44] E. Hollnagel, *Cognitive Reliability and Error Analysis Method: CREAM*. Oxford: Elsevier, 1998.

[45] A. Gregoriades and A.G. Sutcliffe, "Scenario-based systems assessment of non-functional requirements". *IEEE Transactions on Software Engineering*, 31(5), 2005, pp. 392-409.

[46] A.G. Sutcliffe, and A. Gregoriades, "Validating functional system requirements with scenarios", in *Proceedings 10th IEEE Int. Conference on Requirements Engineering (RE'02)*, Essen, Germany, 2012. pp. 181-188.

[47] A.G. Sutcliffe, N.A.M. Maiden, S. Minocha and D. Manuel, "Supporting scenario-based requirements engineering", *IEEE Transactions on Software Engineering*, 24(12), 1998, pp. 1072-1088.

[48] J. Horkoff, E.S.K. Yu, "Finding Solutions in Goal Models: An Interactive Backward Reasoning Approach", In Proceedings 29th Int. Conference on Conceptual Modeling (ER 2010), Vancouver, Canada, 2010. pp 59-75

[49] M. Komssi, M. Kauppinen, H. Töhönen, L. Lehtola, and A. Davis, " Integrating analysis of customers' processes into

roadmapping: The value-creation perspective", in *Proceedings 19th IEEE Int. Conference on Requirements Engineering (RE'11)*, Trento, Italy, 2011, pp 57-66

[50] M. Jackson, *Problem frames: analysing and structuring software development problems*. Harlow: Pearson Education, 2001.

[51] D.L. Moody, "The 'physics' of notations: toward a scientific basis for constructing visual notations in software engineering". *IEEE Transactions on Software Engineering*. 35(6), 2009, pp. 756-779.

[52] J. Conklin and M.L. Begeman, "gIBIS: a hypertext tool for exploratory policy discussion". *ACM Transactions on Office Information Systems*, 64, 1988, pp. 303-331.

[53] A. MacLean, R.M. Young, V. Bellotti, and T.P. Moran, "Questions, options and criteria: elements of design space analysis". *Human-Computer Interaction* 6(3/4), 1991, pp. 201-250.

[54] S.B. Shum and N. Hammond, "Argumentation-based design rationale: what use at what cost". *International Journal of Human-Computer Studies*, 40(4), 1994, pp. 603-652.

[55] N.A.M. Maiden, S. Jones, C. Ncube, J. Lockerbie, "Using i* in Requirements Projects: Some Experiences and Lessons". In E.S.K. Yu, P. Giorgini, N.A.M. Maiden and J. Mylopoulos (eds) *Social Modeling for Requirements Engineering*. Cambridge, MA: MIT Press, 2011, pp155-186.

[56] K. Breitman and J.C.S.P. Leite, "Ontology as a Requirements Engineering product", in *Proceedings 11th IEEE Int. Conference on Requirements Engineering (RE'03)*, Monterey, CA, USA, 2003. pp. 309-319.

[57] J.S. Leite, G. Hadad, J. Doorn, G. Kaplan, "A scenario construction process". *Requirements Engineering,* 5(1), 2000, pp. 38-61.

[58] H. Kaiya and M. Saeki, "Using domain ontology as domain knowledge for requirements elicitation", in *Proceedings 14th IEEE Int. Conference on Requirements Engineering (RE'06)*, Minneapolis, USA, 2006. pp. 186-195.

[59] F. Chantree, B. Nuseibeh, A.N. De Roeck and A. Willis, "Identifying nocuous ambiguities in natural language requirements, in *Proceedings 14th IEEE Int. Conference on Requirements Engineering (RE'06)*, Minneapolis, USA, 2006,. pp. 56-65.

[60] N. Kiyavitskaya, N. Zeni, L. Mich and D.M. Berry, "Requirements for tools for ambiguity identification and measurement in natural language requirements specifications". *Requirements Engineering*, 13(3), 2008, pp. 207-239.

[61] H. Yang; A. De Roeck. V. Gervasi, A. Willis, B. Nuseibeh, "Speculative requirements: automatic detection of uncertainty in natural language requirements", in *Proceedings 20th IEEE Int. Conference on Requirements Engineering (RE'12)*, Chicago, Il, USA, 2012. pp. 11-20.

[62] L. Goldin and D.M. Berry, "AbstFinder: a prototype natural language text abstraction finder for use in requirements elicitation". *Automated Software Engineering*, 4(4), 1997, pp. 375-412.

[63] R. Gacitua, P. Sawyer and P. Rayson, "A flexible framework to experiment with ontology learning techniques". *Knowledge-Based Systems*, 21(3), 2007, pp 192-199.

[64] C. Castro-Herrera, C. Duan, J. Cleland-Huang and B. Mobasher, "Using data mining and recommender systems to facilitate large-scale, open, and inclusive requirements elicitation processes", in *Proceedings 16th IEEE Int. Conference on Requirements Engineering (RE'08)*, Barcelona, Spain, 2008. pp. 165-168.

[65] C. Duan, P. Laurent, J. Cleland-Huang and C. Kwiatkowski, "Towards automated requirements prioritization and triage". *Requirements Engineering*, 14(2), 2009, pp. 73-89.

[66] A. Finkelstein, M. Harman, S.A. Mansouri, J. Ren, Y. Zhang "A Search Based Approach to Fairness Analysis in Requirement Assignments to Aid Negotiation, Mediation and Decision Making". *Requirements Engineering*, 14(4), 2009, pp. 231-245.

[67] B. Cheng, P. Sawyer, N. Bencomo and J. Whittle, J., "A goal-Based modeling approach to develop requirements of an adaptive system with environmental uncertainty", in *Proceedings 12th ACM/IEEE International on Model Driven Engineering Languages and Systems (MODELS-09)*. New York: ACM Press, 2009, pp 468-483.

[68] R. Darimont, E. Delor, P. Massonet and A. Van Lamsweerde, "GRAIL/KAOS: an environment for goal-driven requirements engineering", in *Proceedings ICSE-97*. Los Alamitos CA: IEEE Computer Society Press, 1997, pp. 612-613.

[69] G. Elahi, G. and E.S.K. Yu, "Trust trade-off analysis for security requirements engineering" in *Proceedings 15th IEEE Int. Conference on Requirements Engineering (RE'09)*, Atlanta, USA, 2009, pp. 243-248.

[70] M. S. Feather, S. Cornford, K. Hicks, J. Kiper, T.Menzies, T., "A Broad, Quantitative Model for Making Early Requirements Decisions". *IEEE Software*, 25(2), 2008, pp. 49-56.

[71] M.S. Feather and T. Menzies, "Converging on the optimal attainment of requirements", in *Proceedings 10th IEEE Int. Conference on Requirements Engineering (RE'02)*, Essen, Germany, 2002, pp. 263-272.

[72] C. Ghezzi and G. Tamburrelli, "Reasoning on non-functional requirements for integrated services", in *Proceedings 15th IEEE Int. Conference on Requirements Engineering (RE'09)*, Atlanta, USA, 2009, pp. 69-78.

[73] P. Sawyer, N. Bencomo, J. Whittle, E. Letier, A. Finkelstein, "Requirements-Aware Systems A research agenda for RE for self-adaptive systems", *in Proceedings 18th IEEE Int. Conference on Requirements Engineering (RE'10)*, Sydney, Australia, September 2010, pp 95-103.

[74] V.E. Silva Souza, A. Lapouchnian, W.N. Robinson and J. Mylopoulos, "Awareness requirements for adaptive systems", in *Proceedings SEAMS-11*. New York: ACM Press, pp. 60-69.

[75] J. Whittle, P. Sawyer, N. Bencomo, B. Cheng, J-M. Bruel, "RELAX: A language to address uncertainty in self-adaptive systems requirements", *Requirements Engineering*, 15(2), 2010, pp 177-196.

[76] P. Haumer, K. Pohl and K. Weidenhaupt, "Requirements elicitation and validation with real world scenes". *IEEE Transactions on Software Engineering*. 24(12), 1998, pp. 1036-1054.

[77] P. Heymans and E. Dubois, "Scenario-based techniques for supporting the elaboration and the validation of formal requirements". *Requirements Engineering*, (3/4), 1998, pp. 202-218.

[78] T. Van Hung, A. Van Lamsweerde, P. Massonet and C. Ponsard, "Goal-oriented requirements animation", in *Proceedings 12th IEEE Int. Conference on Requirements Engineering (RE'04)*, Kyoto, Japan,, 2004, pp. 218-228.

[79] S. Fickas and M.S. Feather, "Requirements monitoring in dynamic environments", in *Proceedings 2nd IEEE Int. Symposium on Requirements Engineering (RE'95)*, 1995, pp. 140-147.

[80] W. Robinson, "A requirements monitoring framework for enterprise systems". *Requirements Engineering*, 11(1), 2006, pp. 17–41.

[81] E. Chi, S. Munson, G. Fischer, S. Vieweg and C. Parr, "Advancing the design of technology-mediated social participation systems". *IEEE Computer*, November 2010, pp. 29-35.

[82] G. Fischer, "Extending boundaries with meta-design and cultures of participation", in *Proceedings of NordiCHI'2010, Reykjavik*. 2010, pp. 168-177.

[83] P. Greenwood, A. Rashid and J. Walkerdine, "UDesignIt: towards social media for community-driven design", in *Proceedings ICSE 2012*. New York: ACM Press, 2012, pp. 1321-1328.

[84] S. Marczak and D. Damian, "How interaction between roles shapes the communication structure in requirements-driven collaboration", in *Proceedings 19th IEEE Int. Conference on Requirements Engineering (RE'01)*, Trento, Italy, 2009, pp 47-56.

[85] S.L. Lim, and A. Finkelstein, "StakeRare: using social networks and collaborative filtering for large-scale requirements elicitation". *IEEE Transactions on Software Engineering,* 38(3), 2012, pp. 707-735.

[86] A. Kittur, E.H. Chi and B. Suh, "Crowdsourcing user studies with Mechanical Turk", in *Proceedings CHI-08*. New York: ACM Press, 2008, pp 453-456.

[87] C. Castro-Herrera, J. Cleland-Huang and B. Mobasher, "Enhancing stakeholder profiles to improve recommendations in online requirements elicitation", in *Proceedings 15th IEEE Int. Conference on Requirements Engineering (RE'09)*, Atlanta, USA, 2009, pp. 37-46.

[88] B.J. Wielinga, "Reflections on 25+ years of knowledge acquisition. *International Journal of Human-Computer Studies*, 71(2), 2013, pp. 211-215.

[89] L.A. Scheinholtz and I. Wilmont (2011). "Interview patterns for requirements elicitation", In *Proceedings REFSQ-11*. Berlin/Heidelberg: Springer-Verlag, 2011, pp. 72–77.

[90] O. Brill and E. Knauss, "Structured and unobstrusive observation of anonymous users and their context for requirements elicitation", in *Proceedings 19th IEEE Int. Conference on Requirements Engineering (RE'11)*, Trento, Italy, 2011, pp175-184.

[91] A. Milne and N.A.M. Maiden, "Power and politics in requirements engineering: embracing the dark side?" *Requirements Engineering*, 17(2), 2012, pp. 83-98.

[92] S. Thew and A.G. Sutcliffe, "Investigating the role of soft issues in the RE process", in *Proceedings 16th IEEE Int. Conference on Requirements Engineering (RE'08)*, Barcelona, Spain, 2008, pp. 63-66.

[93] I. Ramos and D.M. Berry, " Is emotion relevant to requirements engineering?". *Requirements Engineering*, 10(3), 2005, pp. 238-242.

[94] J.A. Shin, A.G. Sutcliffe and A. Gregoriades, "Scenario advisor tool for requirements engineering". *Requirements Engineering*, 10(2), 2005, pp. 132-145.

[95] I. Jureta, J. Mylopoulos and S. Faulkner, "Revisiting the core ontology and problem in requirements engineering, in *Proceedings 16th IEEE Int. Conference on Requirements Engineering (RE'08)*, Barcelona, Spain, 2008, pp. 71-80.

[96] A.G. Sutcliffe, S. Fickas and M.M. Sohlberg, "PC-RE: a method for personal and contextual requirements engineering with some experience". *Requirements Engineering*, 11, 2006, pp. 157-163.

[97] N. Leveson, *Safeware*. Reading MA: Addison Wesley, 1995.

[98] N. Leveson, "Intent specifications: an approach to building human-centered specifications". *IEEE Transactions on Software Engineering*, 26(1), 2000, pp 15-35.

[99] C.W. Johnson, C.W. "An introduction to human error, interaction and the development of safety-critical systems", in G. Boy (ed.), *A Handbook of Human-Machine Interaction: A Human Centred Design Approach*. Farnham: Ashgate, 2011.

[100] A.G. Sutcliffe, *The Domain Theory: patterns for knowledge and software reuse*. Mahwah NJ: Lawrence Erlbaum Associates, 2002.

[101] Wikipedia, *Air France Flight 447*, http://en.wikipedia.org /wiki/Air_France_Flight_447 [accessed 27 January 2013].

[102] Wikipedia, *Space Shuttle* Columbia *Disaster*, http://en.wikipedia.org/wiki/Space_Shuttle_Columbia_disaster [accessed 27 January 2013].

[103] N. Seyff, F. Graf, P. Grünbacher, P, and N.A.M. Maiden, "The mobile scenario presenter: A tool for in situ requirements discovery with scenarios", in *Proceedings 15th IEEE Int. Requirements Engineering Conference (RE'07*), New Delhi, India, pp 365 - 366.