Requirements Eng (2011) 16:323–351
DOI 10.1007/s00766-011-0127-y

ORIGINAL ARTICLE

On eliciting requirements from end-users in the ICT4D domain

Kristina Pitula · Thiruvengadam Radhakrishnan

Received: 3 January 2010 / Accepted: 30 June 2011 / Published online: 14 July 2011

Abstract Currently, there is much interest in harnessing the potential of new and affordable Information and Communication Technologies (ICT) such as mobile phones, to assist in reducing disparities in socioeconomic conditions throughout the world. Such efforts have come to be known as *ICT for Development* or ICT4D. Although this field of research holds much promise, few projects have managed to achieve long-term sustained success. Among the many reasons for this, from a software engineering perspective, in many cases, it can be attributed to inadequacies in gathering and defining software requirements. Incomplete software requirements and the consequent failures in creating sustainable systems arise because of inadequate consideration of the high-level social development goals, neglect of environmental constraints and/or a lack of adequate input from end-users regarding their specific needs and socio-cultural context. We propose enhancements to the requirements elicitation methodology specifically adapted to address these shortcomings. Our approach incorporates the novel technique of *Structured Digital Storytelling* to elicit input from end-users who have limited literacy and applies a conceptual model derived from *Communications Theory* to analyse the constraints that arise from their socio-cultural context. The needs, goals and constraints thus identified are integrated using a goal-based analysis to produce a more informed understanding of the potential areas of technology intervention and the needed software requirements. We illustrate our approach and validate its effectiveness with a field study.

Keywords Requirements engineering · Needs elicitation · ICT4D · Culture · Storytelling · Requirements elicitation methodology

1 Introduction

In many developing countries, a growing effort is underway to provide disadvantaged people in rural areas with access to digital content and services using Information and Communication Technologies (ICT). Such efforts are referred to by the term *ICT for Development* or ICT4D. Although numerous pilot projects have been attempted over the past decades, few have managed to bring long-term sustained benefits to the people that they target. A too great emphasis on technical success with inadequate concern for the end-users' needs and the social development aspect of the projects are among the factors that have contributed to this lack of success [1]. In many projects, the multi-dimensional sets of goals and constraints that characterise ICT4D projects for rural communities are inadequately addressed. In particular, many of the social, cultural and economic factors that affect the sustainable use of technology in a rural context are often overlooked. Frequently, existing technologies are introduced in a top-down, non-inclusive manner, without sufficient adaptation or reinvention with regards to the users' needs and socio-cultural context [2]. This deficiency can be overcome by

K. Pitula (✉)
Department of Computer Science and Software Engineering, Faculty of Engineering and Computer Science, Concordia University, 1455 de Maisonneuve Blvd. West, EV 8.401, Montreal, QC H3G 1M8, Canada
e-mail: pitul_87@encs.concordia.ca

T. Radhakrishnan
Department of Computer Science and Software Engineering, Faculty of Engineering and Computer Science, Concordia University, 1455 de Maisonneuve Blvd. West, EV 3.411, Montreal, QC H3G 1M8, Canada
e-mail: krishnan@cs.concordia.ca

involving the targeted beneficiaries (or end-users) more fully in elaborating project requirements. However, because of their socioeconomic situation, end-users generally lack the literacy skills to be able to articulate their problems and needs in a manner amenable to conventional requirements elicitation approaches. Cross-cultural differences contribute further to misconceptions about their needs and how they can be effectively addressed.

From a software engineering perspective, many of these shortcomings can be reformulated as inadequacies in the gathering and defining of software requirements and it is this that we address in our doctoral research. Our overall area of research is ICT4D for rural communities with a focus on methodologies for specifying ICT4D software requirements that lead to systems that make a real difference to their intended beneficiaries. We contend, based on well-established engineering principles, that the early involvement of end-users in elaborating project requirements will lead to systems that satisfy their needs more fully and that satisfying these needs within the constraints imposed by the users' socio-cultural context will result in more successful systems.

A key characteristic that distinguishes the end-users' socio-cultural context is their literacy. Here, by literacy what we refer to is not the basic ability to read and write, but rather the associated analytical skills that allow people to analyse and express their problems and needs in abstract terms. Asking the 'right questions', understanding the 'questions right' and giving the 'right answers' are all learnt skills which, because of their socioeconomic situation, people in rural communities may not have had the opportunity to develop. Nor do practitioners necessarily have the experience and skills to know what questions to ask, how to ask them and how to interpret the answers within a particular rural context. Differences in language and social position act as further barriers to effective communication between users and practitioners. Consequently, rural people are likely to have problems articulating and communicating their information needs through conventional interviews or questionnaire media.

To overcome these problems, we propose a methodology for eliciting input that uses *Structured Digital Storytelling* to allow users with limited literacy to express their problems and needs in the form of stories. These stories are analysed and abstracted into sets of needs, goals and constraints to serve as primary input for constructing a domain model and driving requirements from the bottom up. In order to analyse the socio-cultural context, we apply a conceptual model drawn from *Communications Theory* that lays out the key socio-cultural factors that will impact technology acceptance and use in a rural context. Using a goal-based analysis framework, we incorporate the social development goals and analyse this abstracted information to identify potential areas for technology intervention and the positive or negative impact of the various socio-cultural factors therein. Our approach places user needs foremost in driving requirements at the same time that it identifies contextual information that might otherwise be overlooked. By integrating a bottom-up analysis with a conventional top-down approach, we obtain a more complete understanding of the problem and constraints that a potential solution must satisfy. To show that our approach is effective, we present a field study in which we applied this methodology to determine the needs and constraints of several different communities in rural India.

In the following sections, we first provide background information on requirements engineering. We then characterise the ICT4D domain. We describe the specific shortcomings with applying conventional requirements engineering approaches to ICT4D projects and how we propose addressing them. We provide an overview of the theories and applications related to our work, covering Participatory Rural Appraisal (PRA), computer-based interviewing, storytelling, modelling and the dominant views on culture with respect to information technology, including the theories underlying the cultural model we apply. This is followed by a detailed description of our methodology. We then present the experiment whereby we demonstrate our approach and validate its effectiveness before concluding.

2 Requirements engineering

Requirements engineering (RE) is a crucial step in the development of any software system. It is the process whereby the intended purpose of a system is discovered and documented so that it can be analysed, communicated and eventually culminate in a software implementation that meets that purpose. How well that purpose is met is the primary measure of a system's success. Thus, RE is essential in determining what a system will do and how this will be measured. The process is inherently iterative and consists of three major activities: elicitation of needs, requirements specification and requirements validation. The process starts with some ill-defined 'ideas' of what the system should do. These are elicited, analysed and systematically transformed into a technical requirements specification that defines the software system to be built completely and unequivocally. Once sufficient domain knowledge has been elicited, analysts generally proceed in a top-down manner to identify the areas of technology intervention and the ensuing software requirements. Modelling plays a key role in representing, analysing, elaborating and communicating requirements among the stakeholders and developers, while the ability to trace the link between the stated needs and the technical requirements ensures that needs are met without superfluous features. The RE discipline offers a wide range of established methods and techniques for accomplishing the various activities, appropriate for different problem domains and development styles.

RE is recognised as being one of the most difficult engineering tasks [3]. In the industrialised world, the mismanagement of requirements is among the leading reasons that software projects are problematic or fail [4], with a lack of user input, incomplete and changing requirements and unclear objectives among the chief contributing factors [5]. Incomplete and incorrect requirements inevitably propagate into the later stages of software development, leading to implementations that do not meet their users' needs. Such failures can be avoided by involving end-users and other stakeholders early in the process. This is recognised by the engineering community, and the involvement of end-users is a well-established principle of software engineering, with standard methods (such as interviews, workshops, focus groups and ethnographic studies) to facilitate the elicitation and communication of software requirements between stakeholders and analysts.

RE is by nature difficult because it starts in a largely unconstrained problem space and much of the effort revolves around defining system boundaries, identifying relevant environmental conditions and prioritising and selecting which requirements to pursue among the many possible options [6]. A goal-based analysis is an established technique for addressing such complexity [7]. Goals are objectives for the system to achieve, describing some intended capability or property, while constraints are properties and conditions that must be satisfied when achieving those goals. Starting with high-level goals that identify the purpose of the system, these are successively decomposed until a set of functional requirements by which these goals can be met is attained. This is both a top-down and bottom-up process in the course of which goals are refined and abstracted, interdependencies and constraints identified, alternatives considered and conflicts resolved. A goal-refinement tree provides a structure for representing the relationships between the various goals, subgoals and constraints and for linking high-level goals at the top of the structure to operational requirements in the leaves. Chung et al.'s *Non-Functional Requirements Framework* [8] is one such structure that uses labelled arcs to indicate AND/OR relationships, conflicts and positive or negative support between the goals and with respect to the constraints.

RE is a decisive and indispensible stage in the engineering of any software system, including those intended for ICT4D. In the case of ICT4D projects, the characteristics of the targeted user populations make it difficult to apply standard elicitation techniques. To date, although there is a growing body of research related to software development and deployment across national boundaries, little work has been done in the area of cross-cultural requirements gathering with stakeholders from disadvantaged socioeconomic backgrounds. In this section, we have briefly outlined the role of RE in software development and how a goal-based analysis can assist in elaborating requirements for complex systems. In the following section, we describe the specific challenges that ICT4D projects present, rendering conventional requirements elicitation techniques inappropriate.

3 Challenges of the ICT4D domain

The term ICT4D is used to describe a wide range of endeavours that have the common goal of promoting the socioeconomic development of disadvantaged communities through the direct or indirect use of ICT. These projects are driven by high-level social and economic development goals that most often are initiated from outside the targeted communities. Many involve multiple stakeholders from the public, private and non-profit sectors such as social workers, agronomists, representatives of government, business and international funding agencies as well as NGOs and local community initiatives, working together to be most effective [9]. The intended beneficiaries typically have limited schooling, low literacy and income levels and only speak local languages. Many live in dire poverty with no obvious way of extricating themselves. The developing countries and regions where the projects take place are characterised by inadequate infrastructures, intermittent power and connectivity, underdeveloped economic markets, distribution and support networks and a lack of trained personnel. Extreme climatic conditions such as heat, cold, dust or humidity introduce additional operating constraints. All these factors contribute to creating a novel context, far removed from that of conventional ICT applications.

There are three main thrusts to ICT4D initiatives: (1) developing infrastructure to provide power, connectivity and devices appropriate for the prevailing conditions; (2) building ICT capacity corresponding to the skills and competencies necessary to maintain and use the technology; and (3) providing digital content and services. All three are essential for a project's success, and due to the prevailing conditions, many projects must address all three in unison. Converging on a combination that satisfies all the novel conditions and constraints that characterise these projects is complex. And even if a project is technically successful, there is no guarantee that its high-level social development goals will be attained. For this reason, despite best intentions, many ICT4D projects have failed to bring long-term sustainable benefits to the communities in which they are deployed. The following are among the reasons cited in the ICT4D literature [10, 11]:

- Multiple stakeholders have vague and non-converging objectives, with little or no input from the ultimate beneficiaries
- When project objectives are vague, there are no clear metrics for evaluating success, making claims of success largely dependent on which stakeholder defines it
- Deployment and sustained operation constraints are inadequately addressed, with the result that many projects do not survive beyond the prototype stage once external support is withdrawn
- Usability requirements and evaluations are inadequately reported making it difficult to assess how usable a project is by its target population, let alone apply lessons learnt to new projects
- Requirements pertaining to economic sustainability are not considered, limiting a project's potential adoption and dissemination

While these reasons for failure are not restricted to ICT4D projects, they are especially prevalent there due to the ICT4D domain's characteristics. With many stakeholders from different sectors, their diverse backgrounds and areas of expertise will often result in a set of disparate high-level goals. In particular, the concepts and practices related to social development are often not well understood by laymen. If these goals are incomplete or vaguely stated, they can easily be misinterpreted or overlooked by analysts. With regards to the environment, the technical, economic and cultural conditions which characterise the context of use introduce novel constraints that will compromise a project's success if not addressed. Devices must be appropriate for the prevailing operating conditions (climate, infrastructure, available support) and provide relevant services that are affordable and accessible to their intended users. Here, economic and social factors come into play. For example, to keep costs low, shared facilities may be used. However, this may introduce confidentiality and privacy concerns and other social considerations such as access restrictions due to age, gender or social standing (e.g. women may not be able to visit sites frequented by men or people from lower castes may be denied access [12]). And even if the services offered are relevant and accessible, other factors such as personal obligations, public opinion or local customs may prevent users from being able to fully benefit from them.

Determining what is relevant, accessible and applicable requires the input of end-users; yet, all too often they are not consulted when project goals are set. These are frequently established by external experts to comply with the vision of national and international funding agencies and elaborated in a top-down manner without full understanding of local conditions. Even if the villagers' input is solicited, their social status, limited literacy and lack of exposure to ICT act as barriers to their full participation using conventional elicitation approaches. When end-users are disconnected from a project's goals, they are likely to be unmotivated, distrustful or simply unable to make use of a technology.

Although technology is a core component of any ICT4D effort, experience has shown that the technical success of a project is not sufficient for a successful outcome. Here, by the term technology, we refer to the hardware devices, software applications and physical infrastructure to access information and data services electronically. Also, essential is the ability of people to use a technology in order to engage in meaningful and gainful social activities in a sustainable manner [13]. Diffusion theory has established that for a technology to be accepted by its intended users, it must be perceived as beneficial, easy to use and socially endorsed, with an adequate infrastructure in place to support its use [14].[1] To meet these objectives, a technology must be relevant to the community's needs, expand on existing knowledge and skills and be affordable and sustainable. To be part of a sustainable cycle, the benefits from the technology's use must balance the costs. Finally, for a project to be economically sustainable, it must produce a measurable outcome in a cost-effective manner, be scalable as the user population grows and be maintainable after deployment [15]. These factors give rise to the following key challenges specific to ICT4D projects:

1. Success is to be measured by achieving sustained communal benefits that evolve over the long term as opposed to short term. Metrics to measure the resulting benefits are difficult but necessary in order to show a compelling value proposition that justifies the funding needed to sustain a project beyond the prototype stage.
2. Deployment and sustained operation constraints cannot be resolved from a purely technological perspective, but are dynamically interrelated to a community's broader socioeconomic context. For the technologies to be sustainable in communities where widespread poverty is the norm, innovative business models are needed and their requirements must be incorporated into the projects from the beginning.
3. There are major social, cultural, economic and political differences between 'technologically developed' and 'technologically underdeveloped' societies that impact the effective and sustained use of ICT in realising lasting changes; these differences reside in

[1] The *Unified Theory of Acceptance and Use of Technology* (*UTAUT*) model.

the social dynamics as well as structural characteristics of these societies.

3.1 Reformulating ICT4D challenges as a requirements engineering problem

We consider the problem of developing successful ICT4D systems from a requirements engineering (RE) perspective. We contend that by addressing the shortcomings with current approaches for collecting requirements in an ICT4D context, we can make progress in addressing the key challenges identified above. We focus our attention on the elicitation and analysis of user needs where we identify the following issues:

1. Understanding and incorporating the diverse goals of the different stakeholders to converge on a single, agreed upon set of achievable software project goals
2. Identifying all the environmental constraints that will impact project goals
3. Getting input from the targeted end-users with respect to project goals

When multiple stakeholders from different areas of expertise are involved in a project, their goals will reflect diverse concerns and motivations. Stakeholders from different socioeconomic backgrounds will bring different perspectives and express themselves in different ways. These needs and concerns, along with their underlying assumptions, must be elaborated in detail so that they can be mapped into operational project goals and constraints and any conflicts identified and resolved. This is essential in order to converge on a single set of operational goals that drive the software requirements and according to which the project's success will be measured.

The novel context in which ICT4D projects take place introduces new environmental constraints that may render established software solutions ineffective. Understanding what these constraints are and their impact on a potential solution is critical. It is also very difficult, as many of these constraints arise from socio-cultural conditions and practices specific to these rural communities and foreign to the ICT4D analysts whose knowledge of the rural context and the concepts of poverty, illiteracy and powerlessness are largely theoretical. Such socio-cultural factors fall outside the scope of conventional requirements gathering and thus are frequently overlooked.

With regards to getting input from end-users, along with overcoming the barriers of language, social class and literacy, it is also necessary to consider the cross-cultural differences between the targeted society and that of the ICT4D practitioners. As people lack exposure to ICT, they are unaware of the potential benefits and limitations of such technologies and how these might be made relevant to their needs. Consequently, end-users are unlikely to be able to speak of their needs in terms of technological interventions. Similarly, practitioners from outside the community, although aware of the potential benefits of technology, are unlikely to be familiar with that society's precise needs, making it difficult to probe their ramifications. Somehow, it is necessary to reconcile these two views so that practitioners can develop an informed view of the problem to solve and the potential solution addresses the needs of the majority of users within the constraints imposed by their socio-cultural context.

To address these issues, we propose a methodology for eliciting and analysing user needs specifically adapted for ICT4D contexts. To overcome the barriers to eliciting input from end-users with limited literacy, we propose the novel technique of *Structured Digital Storytelling* (SDS). With SDS, the users' needs are elicited in the form of stories that are recorded using an automated tool. Once the high-level development goals are established, each participant is asked to speak on a set of themes related to the problem under investigation. These narrations are subsequently translated, analysed and abstracted in order to identify the goals, needs, constraints and other relevant concepts within the problem domain. The analysis is enhanced with a conceptual model that identifies key socio-cultural factors that may not be obvious from simple observation. Using a goal-based analysis, the needs and constraints thus identified are integrated and elaborated in order to converge upon a single set of project requirements that address the needs of all the stakeholders including end-users.

4 Foundational concepts from related literature

The concepts we apply in our research are drawn from multiple disciplines. This section provides a brief overview of the theories and applications related to our work. From the social development field, we look at *Participatory Rural Appraisal* as a means of engaging participants in locally sustainable action. In the field of software development, we mention the use of computers to conduct structured interviews with end-users. We discuss storytelling from a sociological perspective and describe its application in a range of domains—to develop a collaborative bottom-up analysis that serves as catalysis to action, for information sharing and to collect contextual information. We then consider modelling techniques for representing and reasoning about problems and briefly cover Domain Models, Influence Diagrams and Causal Loop Diagrams to represent and reason about a social system's static structure, causal relationships and dynamic behaviour and feedback mechanisms. Finally, we consider culture with respect to software, where we present some of the prevalent views before expanding on the theories underlying our model.

4.1 Participatory rural appraisal

In social and economic development circles, the need to involve intended beneficiaries in development projects has long been recognised. The term *Participatory Rural Appraisal* (PRA) based on the work of Chambers [16] is used to describe a variety of approaches that have evolved to facilitate the engagement of local people in development efforts. Here, emphasis is placed on involving local people in analysing their situation with the goal of empowering them to plan and act on their own behalf in producing sustainable local action and institutions. The approaches described by Chambers rely on interviews, focus groups and community meetings to mobilise people and collect information using techniques such as participatory mapping and modelling, transect walks, matrix scoring, well-being grouping and ranking, institutional diagramming, seasonal calendars, trend and change analysis and analytical diagramming—all undertaken by local people, often for eventual submission to the funding agencies. PRA approaches are primarily geared towards identifying and introducing social and economic interventions that local people can undertake on their own. Consequently, while they may play an important complementary role, they are not directly applicable to determining software requirements for suitable ICT tools.

4.2 Computer-based interviews

In [17], the authors present a computer-based interviewing tool developed to facilitate the elicitation of information from end-users when gathering requirements and conducting user tests. In this work, the interview tool is primarily a means for automating conventional structured interviews. Analysts prepare a set of pre-defined and focused questions, which the tool then presents in a prescribed order. In this context, the computer acts as a 'non-threatening, non-judgemental interviewer with limitless patience', overcoming issues such as user inhibitions or a lack of personnel with appropriate interviewing skills. Although the authors initially envisaged collecting computer interviews to prepare for actual face-to-face meetings, they also tested the tool as an alternative to face-to-face interviews. Their experimental results showed that the tool was convenient and effective in eliciting useful information from users and participants preferred the tool over paper questionnaires or phone interviews. While this work is similar in spirit to our approach, we differ in how information is elicited. Rather than using structured interviews with focused questions that ICT4D users are likely to have difficulty answering, we apply open-ended storytelling on a set of predetermined themes.

4.3 Storytelling

Storytelling as a technique is applied in a wide range of domains. Stories constitute an art form, a form of entertainment and a fundamental mode of communication in use for millennia. In the social sciences, oral histories are used to provide alternative views on historical events based on first-hand experience, to capture cultural information, and to explore social issues. In business, storytelling is viewed as an integral part of organisational knowledge management [18], while in software engineering, stories in the form of scenarios are used in the design process to communicate among stakeholders and developers. More recently, storytelling has been put forward as a means of eliciting requirements in domains such as health care, where access to end-users and the actual context of use is restricted [19]. Below, we briefly describe this work.

From a sociological perspective, storytelling is an interactive, communicative activity that takes place between the narrator and an audience in the immediate present through the spoken word enriched by intonation and gesture. Storytelling varies from highly spontaneous and interpersonal accounts, such as those narrated around a dinner table, in the course of which everyday events are jointly constructed, deconstructed and reconstructed, to highly stylised presentations where revered stories are repeated almost word-for-word. In all cases, stories are a means of imbuing order and meaning out of the daily experiences of life. As such, they convey important insights into how participants attribute meaning to their daily experience and identify themselves within their social world, and at the same time, they reveal the culturally embedded normative influences under which they live [20].

In sociology, storytelling has proven itself as an effective means for expressing community information, issues and frustrations as the basis for developing a collaborative analysis from the bottom up. In a recent study by Kerr [21], it was applied to identify the problems of homelessness as perceived by the homeless themselves. The resulting analysis revealed a number of significant issues that do not emerge from conventional top-down analyses where input is solicited from such people as social service providers, public officials and academic experts. There was a comparable divergence in the nature of potential solutions and associated issues as viewed from the top-down versus bottom up. Moreover, by moving from stories centred on life histories to stories concerning what could be done about the present situation and broadcasting these stories to a wider audience, Kerr's research process of 'telling and listening' to stories served as a catalyst to the homeless to become active in changing their situation.

In the ICT4D context, we propose digital storytelling as a means for sharing information among semi-literate people in rural villages. A study by Frohlich et al. [2] has shown the viability of storytelling as a means of communication in rural India. The study showed that villagers were enthusiastic about creating and listening to stories. The study also revealed a certain tension between those interested in creating and disseminating serious 'development' content, and others more interested in creating personal and cultural content for entertainment purposes.

More recently, storytelling has been proposed as a means for eliciting requirements in domains where access to end-users and the actual context of use is restricted. In [19], the author describes a field study in the health care domain comparing the requirements elicited using focus groups and interviews to those elicited using focus groups and stories. The objective of this study was to determine if there exists any difference in the number, breadth and depth of themes addressed and the amount of time required by participants. The study concluded that there was no significant difference with respect to the number and breadth of themes addressed. Moreover, storytelling was more effective than interviews in eliciting more diverse context of use and social information and required less time.

4.4 Modelling

Modelling is both a tool and technique for representing and reasoning about problems in many disciplines, from the Arts and Social Sciences to Software Engineering. Models provide an abstract representation of the entities, relationships and behaviours that characterise some phenomenon. The modelling activity itself is a creative process that serves to develop an understanding of the phenomenon under study, to identify and represent the relevant concepts, to predict how the system behaves under different conditions and/or interventions and to communicate these ideas to others. There is a vast array of informal, semi-formal and formal modelling techniques (and supporting tools) for creating descriptive, predictive and even executable models. Different techniques provide different views of the system being modelled and thus support different ways of reasoning about it. We are interested in graphical models that assist in conceptualising complex, real-world situations and are comprehensible to a general audience. Domain Models, Influence Diagrams and Causal Loop Diagrams are all established techniques that provide different ways of looking at the static and dynamic structure of a system as well as its feedback mechanisms, and in their semi-formal form, provide useful descriptions readily understood by people untrained in their use. Below, we briefly describe these different modelling techniques and what each contributes.

Domain models are widely used to represent the important concepts within a domain, how the concepts are related and their attributes. They are useful in identifying and organising the various concepts a particular domain encompasses and for establishing a common terminology for describing it. However, while the overall view they provide is valuable for understanding the static structure of a domain, it does not express the causal relationships or dynamics of the system. Domain models are commonly represented using ER (Entity Relationship) diagrams or the more specialised UML (Unified Modelling Language) class diagrams favoured by software developers. The basic ER notation consists of rectangles to represent the entities, connected by labelled arcs representing the relations, as illustrated in Fig. 1. This diagram can be interpreted from the bottom up as follows: given an environment and set of goals, these characterise the problem to solve, requiring a decision regarding the action to take that produces the result.

Influence diagrams [22], originating in the field of decision analysis, provide concise graphical representations for reasoning about the flow of information in decision situations. They provide an intuitive way to identify and display the essential components of a problem, namely the objectives, the uncertainties, the decisions, their outcomes and how they influence each other. The basic notation consists of: hexagons to represent the variables to optimise, rectangles for decisions, ovals to represent uncertain variables and double ovals for functionally determined variables. Arrows entering a decision node

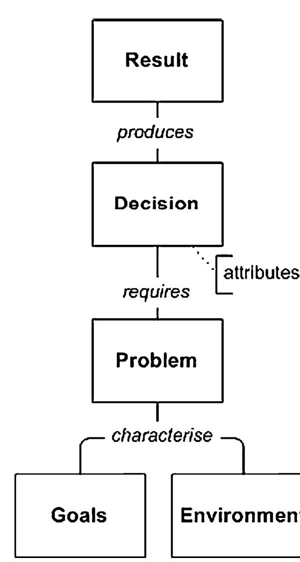


Fig. 1 Domain model

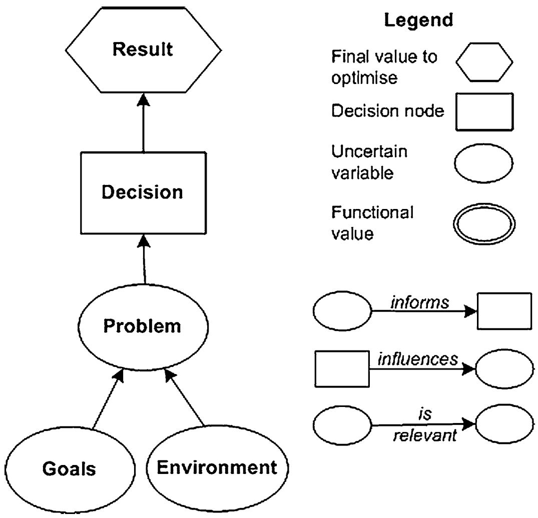


Fig. 2 Influence diagram

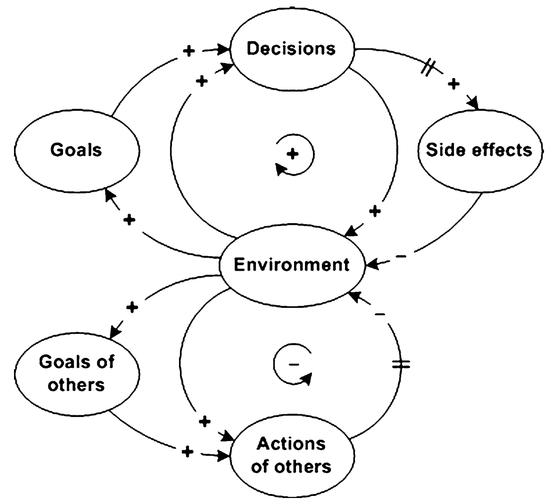


Fig. 3 Causal loop diagram

indicate the information available for making that decision while exiting arrows indicate the decision's influence. Arrows between uncertain nodes indicate relevance, i.e., the information from one node informs the other. An example is provided in Fig. 2. Here, the environment and set of goals 'inform' (i.e. define) the problem, which constitutes the information available to the decision, which in turn influences the result to optimise. While influence diagrams are highly useful for analysing the structure of a decision problem in terms of interdependencies among its components, by definition, they cannot contain cycles and therefore are unsuitable for expressing feedback mechanisms within a system.

Causal loop diagrams (CLD) [23] from the field of Systems Dynamics are used to model complex, non-linear systems with feedback loops. Among their many uses, they are applied in the field of Systems Thinking [24] to model social problems and their underlying causes in order to reason about the consequences of potential interventions to effectuate social change [25]. With a systems perspective, problems are viewed in terms of feedback processes that give rise to problematic behaviour patterns. Here, the system structure is described in terms of its constituent elements, interrelated by circular rather than linear cause-effect chains. These exert a positive or negative influence that, respectively, reinforces or undermines some desired situation. Such positive or negative feedback loops comprise higher conceptual units for describing a system's dynamic behaviour. The CLD notation consists of *elements* linked by arrows (called *causal links*) labelled with '+' or '−' to indicate that they produce a change in the same or opposite direction, with '||'indicating a delay before the effect is perceived, as illustrated in Fig. 3 (adapted from [23]). This diagram can be interpreted as follows: the environment motivates an individual to define goals that lead to decisions regarding a course of action that will change the environment to some desired state. However, these decisions may also trigger unanticipated side effects that will, after some delay, exert a negative influence on the changed environment. Seeing the changed environment, other agents with their own goals will react to restore the situation, to which the individual will react by redefining his goals and so forth in an ongoing cycle.

4.5 Culture and software

The concept of culture is recognised and formalised by people for use in a multi-facetted manner. It is difficult to define precisely and has a wide range of interpretations from different perspectives, each with its own terminology, purposes and traditions. The anthropologist Clifford Geertz defines it as '*an historically transmitted pattern of meaning embodied in symbols, a system of inherited conceptions expressed in symbolic forms by means of which* [*individuals*] *communicate, perpetuate, and develop their knowledge about and attitudes towards life*' [26]. In the Informatics and Management literature, the most commonly cited definition is that of Hofstede [27] who summarises it as the '*collective programming of the mind which distinguishes the members of one group or category of people from another*'. Franklin [28] describes culture as the set of socially accepted practices and values shared by a group of people, with practices '*the way things are done*'. Practices are the observable manifestations of a culture expressed through symbols, artefacts and procedures from forms of discourse, dress and art to societal structures, methods, laws and rituals. Values, in contrast, are largely unobservable, consisting of the set of knowledge, beliefs, norms of behaviour and ways of thinking that underlie the practices and give them meaning [29]. Below, we briefly describe some of the dominant views on culture and technology, before presenting the theories underlying our model.

While culture has been studied from a sociological perspective for a long time, it is only recently that globalisation has brought it to the forefront in the field of Information Technology. Currently, there is much interest in the impact of culture on software as it is recognised that differences in national cultures and values can have a significant impact on software product and process adoption rates [5]. Given that the majority of computer devices, software applications and technology practices have been developed for use in an Anglo-American culture, there is growing awareness of the need for cross-cultural localisation to make them suitable for other cultural contexts, with a corresponding body of research that examines cultural factors with respect to *Software Localisation*. Its focus is on taking existing software products and adapting them to make them suitable for other countries. The related field of *Software Internalisation* is concerned with designing software applications that can be readily localised without requiring engineering changes. A limitation of this research is its focus on the 'external manifestations' of culture (such as language, currency, symbols, presentation formats, conventions, standards, laws and infrastructure) with inadequate consideration given to the deeper aspects of culture [29, 30].

This neglect of 'deep culture' is rooted in the underlying assumption prevalent within the software engineering community that cultural factors only affect the user interface and that core functionality and logic are culturally neutral. This assumption leads to the oversimplifying (and reductionist) view that '*all cultural aspects are encapsulated in the external layer of software'* (emphasis original) and can be localised by simply changing the user interface [29]. The fallacy of applying this view indiscriminately is most evident in the areas of ERP (Enterprise Resource Planning), GSS (Group Support Systems) and Collaborative Software, where corporate mergers and expansion have led to the deployment of such systems across organisational and international boundaries. The need to integrate the business processes and practices from different cultures has drawn attention to the implicit assumptions embodied in the technologies and put these fields at the forefront of research on culture and IT adoption and use [26].

While the literature offers a variety of models for studying culture, these mainly consider culture from a national, ethnic or organisational perspective. Hofstede's Cultural Dimensions [27] (masculinity, power distance, individualism, long-term orientation and uncertainty avoidance) is among the most frequently cited, although it has recognised limitations. Foremost, among these recognised limitations are as follows: the use of the nation–state as unit of analysis; its disregard of cultural differences that occur within or transcend national boundaries; its disregard of multicultural influences; and its view that culture is static over time, contrary to the now dominant view in anthropology that considers culture as emergent and dynamic [31]. Additionally, there is no clear mapping between the cultural dimensions and operational requirements for a software system. Although other models exist, a common theme they share is examining culture in corporate or business environments with sophisticated, urban populations very different to the rural populations targeted by ICT4D projects.

The underdeveloped countries and regions in which ICT4D projects typically take place make culture a key factor. As the field of *Human–Computer Interaction* (HCI) deals with the human issues with respect to computer technology, by convention culture is generally examined from an HCI perspective, with emphasis on the input to and output from the computational elements and how these interactions fit into the broader context of use. The HCI community has a rich repertoire of proven methodologies and techniques for doing this. To understand the users and tasks they perform, ethnographic studies are generally used. Such studies consist of going onsite to talk with users and observe their activities and behaviours related to the proposed system's functionality. This includes observing what supporting artefacts they use as well as the environment in which the activities take place. These studies can be quite intensive in terms of both the extent of the interviews conducted and the amount of observations made. However, when these activities take place within the framework of an actual project, practitioners rarely have the time or resources to conduct comprehensive, in-depth analyses. Traditional ethnographic studies give place to 'rapid ethnography' [32], with the risk that these lapse into 'scenic fieldwork' [33], summarised as 'I went there and this is what I saw'. Such studies are likely to reveal only the surface manifestations of culture, as the larger social fabric in which they are embedded goes largely unobserved and the deeper cultural meanings cannot be readily deduced. While HCI offers a variety of theoretical frameworks for addressing this (cognitive theory, activity theory, situated action, etc.), the considerable skill, time and effort required to understand and apply such frameworks make their practice problematic. Additionally, making such analyses relevant to the software design is not obvious [34]. This has given rise to the nascent field of HCI4D concerned with adapting HCI practice to ICT4D contexts [35].

For our purposes, we draw on *Communications Theory* and Ong's theories in particular [36] to look at the social phenomena surrounding communications at the individual and community level. Information, seen as messages that have meaning within a given cultural context, circulates among members of a society and the means of communication together with the messages in circulation constitute the mindset and shared system of meaning within that society, referred to by Innis as the 'cultural ecology' [37]. According to Ong, culture is a dynamic process, positioned along a continuum between 'orality and literacy', with the mode of communication conditioning how people accumulate, preserve and share knowledge and ultimately how they think and structure society [36–38]. Given that literacy (or its lack) is a major distinguishing characteristic of the targeted user populations, these theories are particularly suitable for examining cultural differences with respect to communications as a means of effecting change in a society. They also highlight the cultural differences that exist between the ICT4D designers, end-users and technology itself. Moreover, the characteristics that emerge can more readily be mapped into operational project constraints and requirements.

Ong contends that an oral culture is by nature traditional, conservative and situational. Traditional knowledge must be carefully conserved as otherwise, once forgotten or distorted, it is permanently lost. In the absence of written records, knowledge is embedded in the stories and practices shared by a community. These are preserved in communal memory, which is continuously refreshed by constant re-enactment. Oral knowledge can only be transmitted through direct contact among community members. People must experience these stories and practices at first hand on a recurring basis if they wish to learn and recall them. In this way, knowledge manifests itself as concrete experience embedded within the social fabric of daily life. By necessity, such a culture is conservative, favouring continuity over experiment and radical change. Here, the collective has precedence over the individual, as it is the collective that embodies the shared experience that constitutes the pool of knowledge available to the community. At the same time, this pool of knowledge evolves adaptively, as what is no longer relevant gradually passes from usage and is eventually forgotten.

In contrast, in a literate culture, knowledge can be permanently recorded. Society is free to experiment and innovate as the original information can always be retrieved if the experiments fail. As noted by Havelock [39], such societies by nature engender individualism, speculation, innovation and change. When knowledge is recorded, direct contact is no longer essential as information can be perused in asynchronous privacy. Reading and writing are in themselves solitary activities that engender introspection. This introduces an objective distance between the author and audience, allowing readers to form their own opinions uninfluenced by live contact. In the absence of a shared environment, the context must be described with analytical precision and abstract concepts are used to synthesise the knowledge embedded in concrete, day-to-day life experience. An analytic viewpoint is more conducive to reflection and speculation, opening the doorway for experimentation, which generates change when it is successful. These processes underlie the scientific method whereby abstract knowledge is separated from experience and then reapplied to new situations. At the same time, because recorded knowledge is relatively static, when change occurs, it is often disruptive.

Ong's original terms of 'oral' and 'literate' to distinguish these two worldviews can be somewhat misleading as what they refer to is not the basic ability to read or write, but rather the extent to which a society has interiorised writing in its thought processes and the value it places on written as opposed to interpersonal sources of information. To avoid confusion in this regard, we henceforth refer to them, respectively, as '*experiential'* (i.e. grounded in a community's world experience) and '*analytic'* (i.e. derived from analysis and theorising). These two should not be viewed as a dichotomy as in fact they manifest along a continuum and are in constant flux, with diverse influences affecting different aspects of an individual's life.

5 A methodology for needs elicitation and requirements analysis

The methodology we propose draws on the concept of storytelling and the theories of experiential and analytic culture to expand the scope of requirements elicitation in an ICT4D context. We augment the standard requirements engineering process by applying *Structured Digital Storytelling* (SDS) to elicit needs directly from the end-users and apply a conceptual model of experiential culture to interpret these needs and additional constraints arising from the broader social context. In this section, we first describe our conceptual model and the SDS concept before explaining how they are integrated within the standard requirements engineering process.

5.1 Conceptual model of experiential culture

The conceptual model we apply for analysing the socio-cultural factors with respect to technology is based on the cultural differences that arise between an *experiential* and *analytic* society. These two worldviews are associated with very different and even dissonant characteristics in the societies that embrace them. Table 1 below derived from [36, 38] summarises some of the characteristic attitudes, traits and tendencies.

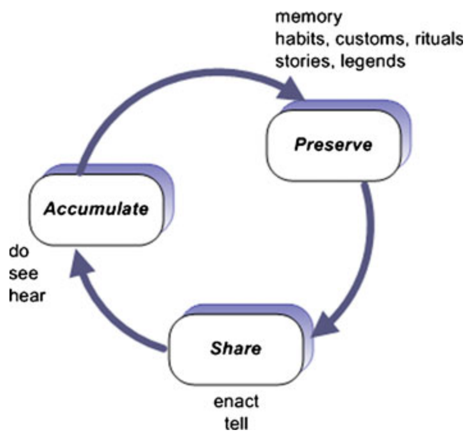


Fig. 4 Knowledge management in an experiential culture

A society's worldview has profound implications on how people manage knowledge. In an experiential culture, knowledge is accumulated through direct experience, by doing, observing and listening. Knowledge is preserved in memory of both the individual and group, with this memory reinforced through constant repetition. Finally, knowledge is shared through its enactment or by telling. This is depicted in Fig. 4. All these activities involve high degrees of interpersonal contact, suffusing knowledge with emotion, empathy and participation in a shared identity. Consequently, in an experiential society, the value of some 'knowledge' is interlinked with the quality of the human relationship as opposed to being based solely on its validity and interpersonal communication is favoured over the impersonal, detached, logical content preferred by analytic societies.

The characteristics of an experiential society introduce certain constraints to bear in mind when considering potential technological interventions and their ability to effect change in a society. The following are among the constraints identified:

1. Averse to disruptive change

Given that traditional knowledge is embedded within their practices, people are averse to disruptive changes that threaten the continuity of this knowledge. To offset this, any proposed change must be gradual and build on existing practices.

2. Reluctant to experiment

Related to an aversion to change is a reluctance to experiment, as this goes against the preservation of traditional knowledge through its constant re-enactment. This reluctance can also be offset by evolving existing practices gradually.

3. Knowledge conveyed through concrete experience

Knowledge that is embedded in practices manifests itself through human action. Concepts are drawn from concrete experience and situated in operational frames of reference. Learning is a situated activity in which the novice learns by observing and emulating the expert without reference to underlying principles. Conveying knowledge within such a context is best achieved by a situated, hands-on learning experience that builds on existing knowledge in familiar situations rather than disconnected, theoretical presentations and by providing observable results as opposed to rational explanations.

4. Not predisposed to formulating abstract plans

Situational thinking is not conducive to abstract analysis and planning with concepts expressed in spatially organised and analytically sparse structures. To counteract this, any required plans must be presented in terms of concrete experience, situated within an individual's operational frames of reference.

5. Reluctant to act individually

Living in a small, tight-knit community bound by a shared, collective experience, people will be reluctant to take a path that sets them apart. This can be offset by promoting group participation in any initiatives.

6. High-context, personal communication

The collective experience and shared environment are conducive to high-context communication. By nature, such societies are participatory and rich in interpersonal relationships and emotion. Unaccustomed to low-context communication, people are likely to be overwhelmed by highly detailed, analytical information and have difficulty relating it to their personal context, which remains largely unanalysed. They are apt to be put off by impersonal communication as they will have difficulty relating to it and consequently distrust it. To offset this, information should be communicated in high context and presented by someone with whom they can relate.

To determine the dominant cultural tendencies in a society, we propose collecting and analysing stories. The styles and forms of speech as well as the informational content of the stories all contribute to establishing how people relate to the events that they are describing. To characterise the discourse of these two worldviews, we draw on the distinction made by Taylor [40] between 'showing' and 'telling'. With showing, the narrator effaces himself from the narrative, giving an impersonal, objective account that conveys a factual nature to the events described. This is the dominant form for expressing scientific knowledge and typical of an analytic society. In contrast, in the telling mode, the narrator participates in the events described, imbuing the account with personal emotion and opinion. Drawing on this distinction and the characteristics described in Table 1,

Table 1 Traits and tendencies of experiential versus analytic cultures

Experiential culture	Analytic culture
Traditional	Experimental, seek change
Conservative	Innovative, seek novelty
Knowledge expressed through human action	Knowledge expressed abstractly
Situational thinking with concepts drawn from concrete experience in operational frames of reference	Analytic thinking with abstract concepts organised in logical categories and lists
Shared, collective experience	Individual, subjective experience
Participatory, emotional	Detached, objective
High-context communication (context construed from shared environment)	Low-context communication (context explicitly stated)
Situated learning	Theoretical learning
Thoughts expressed non-linearly in additive grammatical structures using formulaic expressions, copious	Thoughts expressed linearly as 'spatially' organised arguments, using subordinative structures, analytically sparse and precise
Live in the immediate present with time fluid and flexible (Hall's 'polychronic' perception of time)	Live in computed time managed linearly (Hall's 'monochronic' perception of time)
Collective has precedence	Individual has precedence
Social norms enforced by shame with respect to the collective	Social norms enforced by an individual's guilt with respect to laws

experiential narratives can be expected to display the following salient features:

- speak in concrete terms based on lived experience in operational frames of reference
- directly implicated in the narrative in which they play a central role; speak in the active first person (as opposed to detached third person); convey personal feelings and emotions
- narratives are high context with unstated facts construed from the location or activity
- do not speak of unfamiliar concepts that are outside known frames of reference
- do not speculate about alternatives, tradeoffs or underlying causes
- view interpersonal interactions in terms of personal relationships (trust, honesty, kindness, fairness, etc.) rather than economic or bureaucratic transactions (rules, principles, legal obligations, etc.)

Given that ICT4D end-users and designers generally come from opposite ends of the spectrum, it is useful to note certain practical considerations arising from cultural differences between the two worldviews [36]:

- People from experiential societies are likely to have difficulty construing the meaning of logically organised artefacts such as lists, tables, charts and diagrams that represent abstract concepts spatially. As these are preferred media for people from analytic societies, it is important for practitioners to reformulate such artefacts into situational frames.
- People from experiential societies more readily accept inconsistencies and contradictions when these make sense in the context in which they occur, in contrast to people from analytic societies who strive to find underlying principles.
- A linear plot line is a construct of an analytic society. In an experiential culture, narrative does not necessarily have a chronological ordering but rather consists of a collection of episodes organised thematically and ordered according to the demands of the situation.

5.2 Structured digital storytelling

Our notion of *Structured Digital Storytelling* (SDS) builds on the concept of digital storytelling to which we add a goal-oriented specialisation. Storytelling at its simplest consists of someone telling their personal story on some topic and the narration being recorded. An interviewer may assist to ensure that items of interest are clarified and expanded. Recent digital technologies support the authoring of sophisticated multimedia stories that can be made accessible to a broad audience. *Interactive Voice Response* (IVR) systems offer an alternative approach for collecting user input. IVR applications use structured dialogues to ask a sequence of questions when eliciting information. Our approach combines the two. By adding a multimedia, structured dialogue interface onto digital storytelling technology, we can assist people in expressing their information needs through stories that can then be shared in the community. Instead of asking focused questions about their information needs, the villagers' needs are elicited through a series of open-ended questions, short stories, 'what if scenarios' or by hearing their neighbours' views regarding the issues and frustrations they face. The structured dialogue ensures that relevant themes are covered, while hearing stories told by their neighbours will inspire people to tell their own stories.

In an ICT4D context, SDS has a number of advantages over commonly used requirements elicitation techniques such as interviews, focus groups and ethnographic studies [41]. The limitations of ethnographic studies for eliciting requirements in an ICT4D context have already been discussed. Below, we briefly cover the advantages and disadvantages of interviews and focus groups.

With interviews, practitioners meet with end-users individually to ask them questions regarding their problems and needs. The interviews may be structured, semi-structured or unstructured. In a structured interview, the questions are pre-defined and all users are asked the same questions, whereas with unstructured interviews, the interviewer asks questions on the fly. A semi-structured interview combines the two, with the interviewer asking questions from a pre-defined list and follow-up questions as needed. Interviews have the advantages that both verbal and non-verbal responses can be observed, and users can be probed in depth with follow-up questions. Among the known disadvantages, tacit knowledge is difficult to elicit and information on the context of use is not readily observable. Given the novel nature of the ICT4D context, there is no way of ensuring that all relevant aspects are covered by the questions. Furthermore, because of the socioeconomic differences, end-users may have difficulty answering direct questions or be intimidated by the interviewer. Additionally, conducting individual interviews is time consuming and requires someone who speaks the local language.

Focus groups are similar to interviews, with the difference that users participate in a group rather than individually. In a focus group, a facilitator presents the group a series of pre-defined questions or topics that the participants then discuss. Along with the advantages and disadvantages of interviews, focus groups have the advantage that the discussion among participants may reveal more requirements and they are less time consuming than individual interviews. The disadvantages are that participants may feel uncomfortable stating opinions that differ from those of the group, leading to 'groupthink'. Also, certain participants may dominate the discussion, leaving other valid viewpoints unexplored.

A major difference between SDS and the other elicitation techniques is that the narrator is largely left on their own to tell their story in their local language. Among the advantages, we foresee with SDS are that storytelling capitalises on the villagers' primary mode of communication. Although the questions provide some general guidance, an interviewer is not present to influence the narration. Thus, it is possible to identify problems and needs not initially envisaged and contextual factors that might otherwise be overlooked. Moreover, collecting stories involves fewer resources in terms of facilitators, preparation and elapsed time and it does not require facilitators who speak the local language. A disadvantage with respect to the other elicitation techniques is that there is no one present to provide clarifications, guidance or immediate follow-up on items of interest. Additionally, participants may focus their story on one aspect, leaving other equally relevant aspects unmentioned. Among the challenges to address, storytelling is by nature a social exchange that is enriched by the presence of an audience. As with any technique, consideration must also be given to confidentiality and self-censure, particularly when dealing with sensitive subjects.

5.3 Incorporating SDS into the requirements gathering process

Our methodology, specifically designed for ICT4D projects, applies established software engineering principles such as user-driven requirements, goal-oriented analysis and requirements validation based on traceability to user needs. It augments the standard requirements engineering process by applying *Structured Digital Storytelling* to elicit requirements and contextual information directly from end-users and a model of experiential culture to identify cultural factors that are not directly observable. Using a goal-based analysis, the outputs of this process are incorporated into the standard RE process to provide a bottom-up view of the potential areas of technology intervention, while the cultural model is applied to identify additional constraints. An overview of the augmented RE process is provided in Fig. 5 followed by a detailed description of the SDS process integrated within a conventional RE process, with a focus on the elicitation phase and the requirements specification and validation activities as they relate to the SDS outputs and models produced during elicitation.

5.3.1 Needs elicitation

Elicitation in the standard RE process encompasses the activities required to understand the problem the proposed system will address, delimit the system boundaries and identify requirements. Modelling plays a key role, with domain, task and goal models common techniques for representing the problem space. The SDS process fits within these elicitation activities to assist in understanding the problem and constructing the models based on which the software requirements will be derived.

ICT4D projects typically take place under the auspices of development organisations, funding agencies and technology-related businesses, to deliver information or

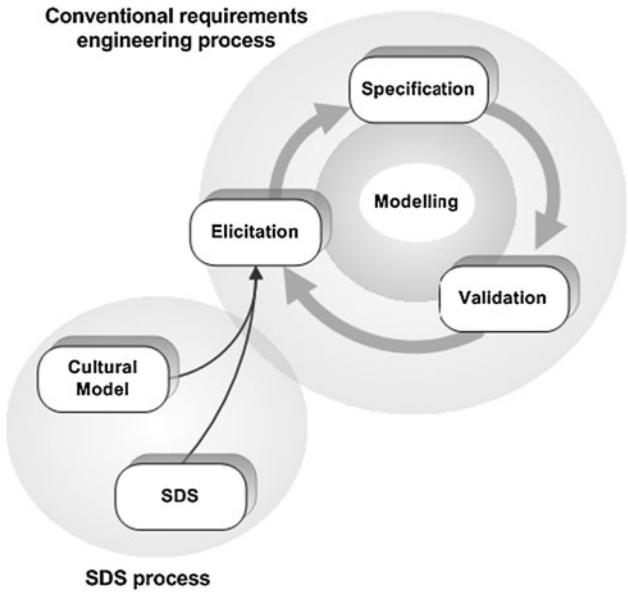


Fig. 5 Conventional requirements engineering process augmented by SDS process

services of value concerning some perceived need in the population. Given this high-level goal, requirements elicitation generally starts with determining what categories of information or services pertaining to that need are critical to the targeted population in prioritised order. These categories correspond to potential areas of intervention for the project, and at an operational level, relate to a community's economic activities (farming, fishing, etc.) or the well-being of its families (health, education, governance, etc.). A promising area is then selected, and its software requirements are elaborated in detail. This corresponds to a classic top-down approach to RE.

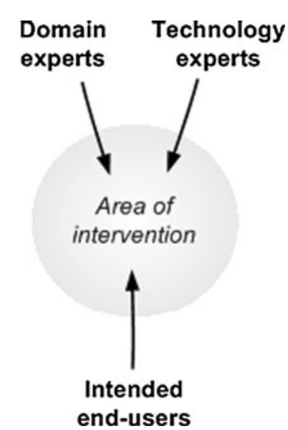
Ideally, the requirements analyst works together with the domain and technology experts as well as the intended end-users to elicit the necessary information. However, while the external stakeholders' needs can be elicited directly using conventional techniques, the rural users are likely to have difficulty articulating their needs for the reasons already discussed. Lacking other sources, the analyst must rely on information obtained indirectly from the domain experts regarding the users' local context and needs. While this second-hand information can be useful, it is also likely to reflect the experts' analytic viewpoint and be inaccurate and/or incomplete, particularly as regards to the local context with respect to technology. Without sufficient input from end-users, a top-down development model is followed, with project requirements driven by external experts rather than the actual needs of the people. It is here that we propose using SDS techniques supported by a suitably designed interactive multimedia software tool, to elicit users' needs through stories, thus providing a bottom-up perspective to complement the experts' top-down view, as illustrated in Fig. 6.

The SDS process itself consists of collecting oral narratives expressed from an experiential perspective and transforming them into an analytic representation suitable for identifying requirements. The process involves 4 major steps: (a) identifying themes of interest; (b) collecting stories; (c) processing the stories to extract information; and (d) modelling the extracted information, as illustrated in Fig. 7. Throughout this process, the analyst works together with the domain and technology experts, to set the themes, collect and analyse the stories and specify and validate requirements. SDS can be applied at any stage of requirements elicitation, to assist in identifying areas of intervention, to identify and validate high-level goals and constraints or to elaborate and validate operational goals and conditions. The use of an SDS approach does not exclude the use of other elicitation techniques, and it can effectively be very complementary, assisting stakeholders in validating that they are focusing on the right problem and that the problem is thoroughly understood in the context in which it occurs.

5.3.2 Identifying themes

Given a topic, the SDS process starts with selecting the focus of elicitation. Depending on how well understood the high-level need is, emphasis may be placed on eliciting general contextual information (e.g. farming in general) or alternatively on eliciting detailed information about a particular activity or event (e.g. planting or selling produce). Working with domain experts, themes of interest are identified and prioritised in order to arrive at an optimal number to produce stories of acceptable length. Assuming 1–5 min of narration per theme, 5 themes plus or minus 2, should be reasonable. A theme may be allocated to eliciting demographic information, or alternatively, a short sequence of focused questions may be asked. We refer to the set of themes associated with some topic as a 'story'.

Fig. 6 Bottom-up view obtained from end-users to complement the top-down view provided by experts



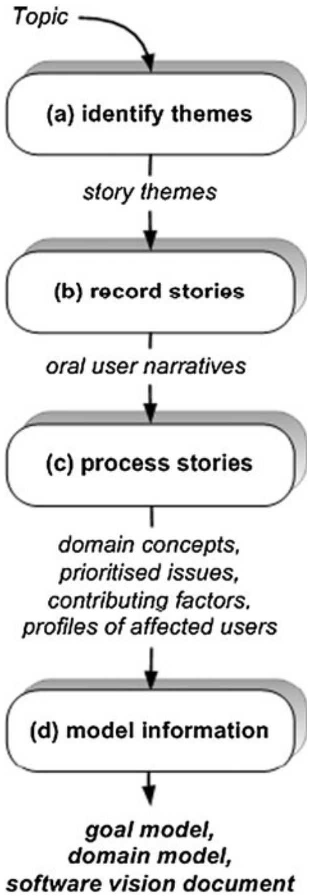


Fig. 7 SDS process

When selecting themes, it is important to consider the experiential nature of the targeted rural societies. While readily able to talk about things they do, people they encounter and events they experience in the context of concrete, familiar situations, participants may have more difficulty responding to themes relating to abstract categories. For example, 'borrowing money to buy seed' is an event well situated in a 'planting' context, whereas it becomes a detached concept if associated with the abstract category 'financing farming activities'. When identifying themes, coarse-grained domain and task models can be used to relate the information sought to the concrete situations in which it occurs. The local vernacular should also be considered when formulating, translating and recording questions. With regards to wording, we emphasise that questions should be open-ended to elicit broad coverage without going into specific techniques for encouraging talkativeness.

5.3.3 Recording stories

Once the themes have been identified, recording stories is relatively straightforward. The set-up will vary depending on available amenities. An application (such as the E-Tool presented in the next section) that plays the questions and records responses can be made available on a suitable device (e.g. laptop or mobile phone) in a location such as a community centre, and villagers invited to use it. As with any such undertaking, a local champion and the acquiescence of local leaders will favour participation. Enlisting respected members of the community to record their own stories as examples for the story library will provide further encouragement, and let participants 'hear' how to respond to the themes. Time wise, the technique takes the time required to tell a story, allowing a reasonable number of stories to be recorded in a few days. Because resources such as interviewers and facilitators are not involved, the application can readily be deployed in a number of villages, increasing the number of the stories recorded and coverage of issues.

5.3.4 Processing stories

Once the stories have been collected, they are first transcribed and translated before being analysed to determine what concerns and problems participants mention, their importance and other noteworthy elements. While transcription and translation are relatively straightforward, the analysis requires more skill to identify and classify the issues, establish what factors are relevant and how they are related and to determine the cultural tendencies that participants manifest. Domain experts are expected to participate in this analysis. The various steps involved in processing and analysing the stories are depicted in Fig. 8 and described below with examples drawn from the case study.

Step 1—translation and transcription Since the stories are narrated in the local language, they are first transcribed and translated into English (or other working language). This is a manual activity that is somewhat labour intensive, but does not require specialised skills beyond knowledge of the local language, English and basic literacy. 'Experiential' narratives will refer to concrete concepts and events expressed in simple grammatical structures, making it easy to produce a literal translation. This can be carried out by people from the region with appropriate qualifications, assisted by local experts familiar with the terms in use. The output of this step is the transcribed stories.

Step 2—concept identification and classification The narratives are examined to extract demographic information and identify significant domain concepts, including problems, issues and concerns as well as additional needs and desirable features. Elements such as activities, actions, agents, objects, events, locations, opinions and attributes are identified and classified, while issues and concerns are flagged and their significance characterised. These are abstracted into concepts for inclusion in the domain model;

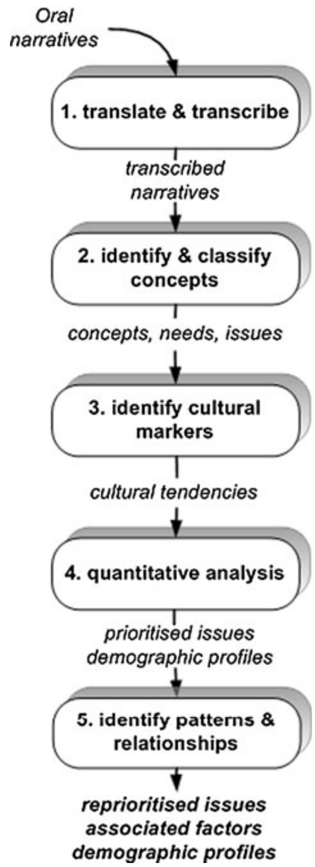


Fig. 8 Steps involved in processing stories

for example, farmers may speak of lacking a well, the well going dry, the rains being late, crops requiring too much water, etc. All of these narrations are classified under the concept 'irrigation' with the problems flagged as 'irrigation issues'. Relationships are of particular interest, as they are often central to problems. Thus, for example, when farmers speak of 'being compelled' or 'having no recourse' in the context of selling their produce, these point to an unequal power relationship, while attributes such as 'dishonest', 'unreliable', indicate trust issues. All the concepts that emerge from the stories are noted, even if mentioned only once, as they might provide a critical insight into some unknown or poorly understood phenomenon, especially when narrations are high context. This phase produces the set of concepts to include in the domain model, additional needs and features for the goal model and a summary view of the individual stories, highlighting the subset of concepts contained in each.

Due to its qualitative nature, certain aspects of this analysis, such as identifying and classifying concepts (commonly known as 'coding') or extrapolating goals, require skilled attention. For this, we draw on established techniques and tools from the field of qualitative data analysis. While a qualitative analysis is interpretative by nature, here again the experiential nature of the narratives—with their focus on tangible experience—is conducive to identifying the entities, agents and actions involved in some event with minimal interpretation. Here also, the domain experts are consulted to validate and clarify the various concepts. A tagging tool can be developed to assist the manual process of mapping information in the narratives to concepts and developing the classification scheme. When dealing with large amounts of data, semi-automated or automated natural language processing can also be applied. Corpus linguistics has been used to process documents for requirements engineering in a number of problem domains. A general overview of the application of natural language processing to requirements engineering is provided in [42].

Step 3—cultural tendencies In this step, the stories are analysed to identify the cultural tendencies manifested. The goal here is to distinguish whether the targeted populations are predominantly 'experiential' or 'analytic' with regards to the planned intervention. As we are dealing with translations, the analysis focuses on what participants speak of and the voice used rather than the syntax and vocabulary of the narratives. This analysis draws on the markers from our cultural model to characterise narratives as grounded/theoretical, implicated/detached, high/low context and emotional/rational. These characteristics can be associated with an entire story or with a particular theme or concept, thus allowing for tracking different cultural tendencies within the same story. This step establishes the cultural characteristics present in the stories. Currently, we view this as a manual process that can be conducted in parallel or following step 2, with the assistance of similar tagging tools.

Step 4—quantitative analysis of stories Assuming a sufficient number of stories have been collected; this step uses the summarised stories produced in step 2 to provide a quantitative overall view of the information. It presents demographic information on the participants and the incidence of the various problems and concerns mentioned, allowing these to be prioritised. Of interest, also are the problems that receive few mentions but are critical in nature, as there may be underlying reasons for their omission from the narratives. The output of this step, which ideally is fully automated, is a list of prioritised issues, concerns and omissions that serves to focus the next step.

Step 5—profiles, patterns and relationships Given a problem, this analysis probes to understand the underlying factors that contribute to it, or alternatively, prevent it from occurring. As this is the domain experts' area of expertise, we expect them to fully participate. Here, we look at the demographic profile of the people who mention a problem, other problems they mention and make comparisons with those who do not, to seek patterns in the data that will permit us to establish relationships between a problem or set of problems and the set of conditions characterising those it affects. For example, while all the farmers mention problems with pests, less complain about the cost of pesticides and even fewer complain about its availability. A profile analysis reveals that those complaining about cost and availability are the larger, better off landholders who can afford to apply pesticides and thus are affected by its cost and availability. Probing the differences between these two groups further, we discover that while larger landholders are effusive in their complaints concerning selling their produce (e.g. commissions, cheating, unfair pricing), smaller landholders are largely silent. Seeking a reason for this omission, we discover that farmers who borrow money are compelled to sell to the lender, thus bypassing the open market system for selling produce. Such omitted issues, that one would expect to be significant, get particular attention as they may conceal underlying problems warranting further investigation such as subjects that are taboo, a fear of repercussions or some socially undesirable behaviour (e.g. borrowing money). Because the data are derived from narratives rather than structured surveys, this analysis is considered exploratory, revealing possible underlying factors and relationships without establishing their statistical prevalence or significance. While the interpretation of such patterns is a skilled manual activity, automated data mining techniques can be applied to detect patterns within large bodies of data. We refer readers to the related literature. The output of this step is a reprioritised list of issues along with contributing factors and the demographic profiles of affected users, including their cultural tendencies.

5.3.5 Modelling the SDS outputs

Modelling is central to requirements engineering as it provides an abstract representation of the problem space based on which system goals and boundaries are established and requirements defined. The SDS process contributes by providing inputs for representing and analysing the problem from the users' perspective. Here, we draw on Domain models, Influence Diagrams [22], Causal Loop Diagrams [23] and Goal Models [7] to represent and transform the SDS outputs into a representation that serves as a basis for elaborating software requirements. These models are used in a descriptive fashion to assist in developing and communicating the static and dynamic nature of the problems described in the stories, to analyse the potential effects of possible interventions and to identify specific needs and constraints. The SDS modelling process is depicted in Fig. 9 and described below.

Step 1—Domain model SDS modelling starts with constructing a problem domain model with the concepts, needs and issues extracted from the stories (see Fig. 11 in the following section). This model serves to identify and organise the concepts, establish their relationships and classify the associated needs, concerns and issues, all of which are linked back to the specific stories and themes in which they are mentioned. This model serves as a base reference for the other models.

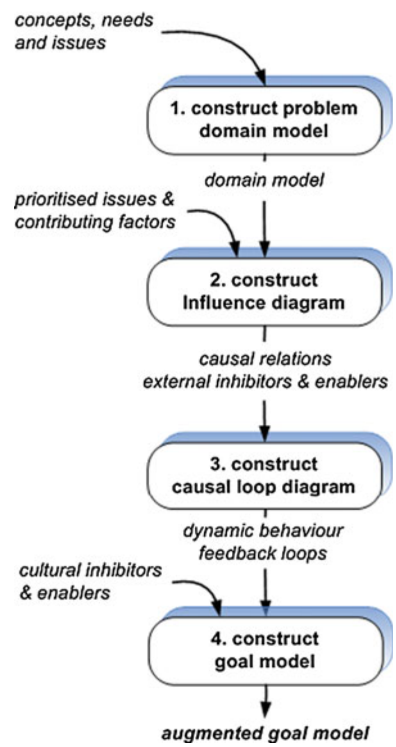


Fig. 9 SDS modelling process

Step 2—Influence diagrams Given a specific issue and set of contributing factors (from the list of prioritised issues and associated factors identified in step 5 of processing stories), the elements that comprise that issue are modelled as a decision situation. Using influence diagram notation, the problem is modelled in terms of its objectives (i.e. the issue being studied), variables, decisions and outcomes in order to explicitly represent and reason about the causal relationships that exist among the elements and their influence on each other (see Fig. 12). While the term 'decision' implies some choice, in an ICT4D context, this choice is often dictated by an individual's circumstances that may impose hard constraints on the options available (e.g. farmers who lack funds must borrow money; farmers without storage facilities must sell their produce immediately). These constraints can be tangible things such as the availability of time, money or tools, to more insubstantial factors such as government policies, access to funding or other incentives that influence a person's behaviour and exert a positive or negative influence on an individual's ability to act. We refer to these real-world obstacles or aids as *external* constraints and introduce the terms *inhibitor* and *enabler* to refer to those that, respectively, impede or facilitate some course of action.

Modelling the problem as a decision situation brings out the underlying cause-effect structure in terms of the inhibitors and enablers that influence decisions and their consequences on outcomes, with different circumstances giving rise to different paths. The dynamics of these paths can then be modelled as causal loop diagrams, with the influence diagram serving as the base reference. Influence diagrams are also useful for deliberating about goals as they can readily be mapped into high-level goal models, with the objective to optimise mapped to the high-level goal and the contributing factors mapped to subgoals, as illustrated in Fig. 14. Influence diagrams can be created to model different issues at different levels of abstraction, depending on the complexity of the problem being studied. The link to the stories is retained by relating the elements in the influence diagram to the concepts and relations in the domain model.

Step 3—Causal loop diagrams (CLD) are particularly appropriate for modelling the different paths through the influence diagram and for explicitly representing the various inhibitors and enablers and their positive or negative influence on outcomes. These paths can be modelled as linear cause-effect chains (or open loops). However, as the problems being studied are dynamic, real-world situations, they often contain feedback loops at the core of the issues mentioned. Modelling the problem using CLD permits us to reason about the system in terms of its dynamic behaviour and to identify problematic feedback loops that trigger or exacerbate the situation. For example, lacking funds to buy supplies, farmers are obliged to borrow with the condition that they sell their produce to the lender at below market price. Consequently, they have less income, which together with the interest charges reduces their available funds, obliging them to borrow again (see Fig. 13). Such negative feedback loops constitute standalone issues in themselves, while positive loops can have a reinforcing influence, leading to the identification of inhibitors and enablers at the systems level. Considering possible interventions with respect to such loops allow us to evaluate the potential impact of these interventions as well as their side effects. While CLD models can be created directly, deriving them from the related influence diagram ensures that they are grounded in the experience described in the stories. Here again, CLD models can focus on different issues at different levels of abstraction, while the link to the stories is retained by maintaining links to the domain model.

Step 4—Goal models The detailed goal model is the representation that synthesises the information derived from the SDS analysis in a form suitable for defining software requirements. This model is developed incrementally by starting with a high-level goal (from the prioritised list of concerns) and then successively decomposing and refining it until a set of operational requirements by which it can be met is attained. To develop the detailed goal model, we draw on the other models. Given an issue, a quick examination of the related influence diagram shows its cause-effect structure. By mapping the underlying causes to subgoals, the primary issue can be addressed, producing a preliminary goal model for that issue. To decompose it further, we now consider the issues associated with the concepts related to each subgoal (e.g. to improve seed supply options, we examine the concerns associated with seed) and reformulate these as more detailed subgoals. The resulting model provides an overview of possible interventions with respect to the primary issue (see Fig. 14).

Whereas some of the concerns will be addressed, others will not and still others are innate (e.g. a lack of funds or distrust of suppliers). We classify the latter as *external inhibitors* and add them to the list of inhibitors identified in the course of our analysis, including any system level inhibitors corresponding to negative feedback loops on the related causal loop model. Before proceeding, we introduce the notion of *cultural* constraints. Cultural attitudes can exert a positive or negative influence on behaviour, which we, respectively, call *cultural inhibitors* and *enablers*. Inhibitors can be addressed by introducing appropriate enablers (or subgoals) to cancel them. Inhibitors and enablers associated with some goal can be captured using the notation presented in [8]. With this notation, interdependencies between goals are indicated by arcs labelled ‘+’ or ‘−’ to show the positive or negative influence they proffer as well as a qualitative assessment of their weight. These goals are termed ‘soft goals’ as there is no clear way of measuring if they are satisfied, instead they are considered ‘satisficed’ if they can be realised within acceptable limits. Elaborating the goal model further, we introduce the external and cultural inhibitors and enablers (see Fig. 15). Once the enablers are laid out, we next consider alternative ways in which they can be realised. For example, a possible way for achieving ‘observable results’ with regards to seed suppliers is to show videos of crops. This would be captured as an operational goal and correspond to a high-level system feature (see Fig. 16). The set of system features supporting some high-level goal may vary significantly depending on the particular subset of goals selected.

Working together with domain experts and other stakeholders, the problem is analysed and potential solutions are evaluated for their viability and potential social impact. The list of prioritised concerns, as perceived by the intended users, serves to focus and potentially redefine high-level project goals, while the contributing factors assist in identifying additional needs and constraints. A profile of affected users helps establish who might benefit from a potential solution and who would not, with the possibility of introducing additional goals to address the reasons for their exclusion. Knowledge of the system’s dynamic behaviour helps identify inhibitors and enablers at the systems level.

The users' cultural attitudes introduce yet more constraints regarding what constitutes an acceptable solution. Goals are elaborated, inhibitors identified and the corresponding enablers found, or alternatively, unattainable goals are abandoned. Additional goals and constraints related to the system's operating environment, business model and HCI dimension are systematically incorporated. This modelling activity can be facilitated by suitable automated tools, like those developed to support diverse semi-formal modelling approaches in the goal-oriented requirements engineering literature [43]. The primary artefacts resulting from the elicitation phase are a detailed goal model, supplemented by a domain model, influence diagrams and causal loop diagrams and a detailed *software vision* document.

5.3.6 Requirements specification

During *specification*, the goals, needs, features and concepts are transformed into a complete set of software requirements. Here, the augmented goal model of the previous phase serves as primary input for a traditional requirements engineering process. Using a conventional goal-based analysis that starts with the high-level goals identifying the purpose of the system, these are successively decomposed and refined until a set of technical requirements by which these goals can be met is attained. Proceeding in a top-down manner and integrating statements from the various sources (user goal model, other stakeholders, HCI analysis, operating conditions, business model, software quality properties, etc.), goals are elaborated and analysed, negative and positive interdependencies identified and tradeoffs negotiated. Here, analysts are expected to work closely with domain experts to reconcile differing goals and clarify unstated assumptions. Throughout this process, the information gathered during the elicitation phase serves to prioritise goals, expose obstacles and identify additional goals and constraints to satisfy. The goals resulting from this analysis are refined until a single set of well-formed technical requirements is attained, documented in the final *software requirements specification* (SRS). This specification unambiguously depicts the set of functional and non-functional requirements for the envisioned software product against which its success will be measured.

5.3.7 Requirements validation

Requirements validation consists of ensuring that the SRS accurately reflects the stakeholders' needs and is consistent and complete. One facet of accuracy involves ensuring that the specified requirements address the stated user needs (quality of conformance) while another considers whether the specification fulfils the users' actual needs and expectations (quality of design). Conformance is established by ensuring that each stated user need is addressed by at least one requirements statement and that superfluous system features unrelated to any need are excluded. Establishing a specification's consistency consists of ensuring it contains no unexpected interactions or conflicts between requirements. Formally expressed requirements can be checked using formal verification techniques such as static analysis or model checking. However, these cannot be used to evaluate the quality of design and formal specification languages are arduous, limiting their use. With semi-formal or informal specifications, validation is accomplished using informal techniques such as walkthroughs, reviews and checklists. In all cases, quality of design can only be assured by having stakeholders review and approve the specification.

When reviewing non-formal specifications, a goal-based analysis offers many advantages [7]. The conformance and completeness of a specification with respect to a set of goals can be established by ensuring that all goals within that set can be achieved and that all requirements are related to at least one goal within the set. Consistency is ascertained by ensuring that all interdependent goals are 'satisficed' with respect to each other. When it comes to validating the quality of design, while some stakeholders might be able to review an SRS, many are overwhelmed by the technical detail and end-users are customarily excluded. The different levels of abstraction present in a goal model permit stakeholders to view a specification at a level of detail that they can comprehend. These views are also useful for exploring alternatives, validating choices and detecting and resolving conflicts. Additionally, by maintaining traceability links between the stories, goals, and requirements, any requirement can be traced back to the specific stories in which the need was expressed, thus validating the design with respect to the needs of the intended users. These links also provide the justification and rationale for including any requirement, making it easier to manage requirements as they evolve over the lifetime of a project. Here, the relative stability of goals, as compared to the wide variability in system features by which these goals can be met, makes a goal model particularly useful.

6 Case study applying the methodology

In order to test the viability of our SDS process for requirements elicitation, we conducted three field studies in two rural areas of India using a prototype elicitation tool called the E-Tool. Our objective was to determine whether the SDS process can effectively be used for requirements elicitation within an ICT4D context by establishing:

1. SDS is acceptable to end-users
2. The needs and constraints that emerge from the stories are non-trivial and non-obvious
3. The process is repeatable and adaptable

Towards this end, we elicited stories on two different topics in two different rural regions with distinct regional languages. Below, we first provide a brief description of the E-Tool prototype and the rural Indian context before describing our experiment and results. To show that this process reveals non-trivial information and demonstrate the application of the SDS process, we present an analysis of the farming stories and demonstrate how the stories are transformed into an augmented goal model suitable for elaborating software requirements.

6.1 The PC-based E-tool

The E-Tool is designed to be a self-contained application for collecting narrations from villagers with minimal intervention from outside support staff. The application runs on a portable laptop computer equipped with a microphone and provides (1) an introductory video explaining the reason for collecting the stories as well as an overview of the application, (2) a story library where the stories are stored and villagers can listen to them and (3) an interview feature which guides users through a series of questions, letting them tell their own story. The application is designed to be easy to use by a non-literate population. Drawing on the experience described in [44, 45], instead of text, navigation aids are provided using graphical icons, buttons with distinct colours and audio prompts to guide users through the various options and a video is provided to motivate users. The E-Tool user interface for the functionalities ‘listen to stories’ and ‘tell my own story’ is illustrated in Fig. 10. A recording session elicits a sequence of recordings, where each recording focuses on a simple theme relevant to the end-user, for example ‘*Tell us about the crops you cultivate*’. A ‘story’ consists of the sequence of recordings made by a given user on those themes. A small sequence of such thematic recordings is pre-planned for a session based on a predefined course-grained domain or task model developed by an analyst.

The PC-based E-Tool presented here is a proof-of-concept with minimal administrative functionality. While useful for validating the SDS concept, a laptop deployment presents certain logistic drawbacks. We have since redirected our work to port the E-Tool to a mobile phone platform. Below, we briefly describe the story library and interview feature of the PC version. A more detailed description including the design rationale is available in [46].

The story library allows users to listen to stories told by other members of the community. On accessing the library page, users are presented the available stories and an audio prompt tells them what they can do. Stories are labelled with an automatically generated number and an optional photo. Sample stories are presented at the top. These are recorded by selected individuals such as local leaders and preloaded in the library to serve as concrete examples of the stories users are expected to tell. As new stories are recorded, they are added to the end of the list. By moving the cursor over a story icon, users can hear that story’s ‘title’. Clicking on the icon causes the complete story to play. Users can pause and resume playback or scroll forward and backward through the narration. Playback stops when the user clicks on another story or exits the page.

The interview feature guides users through the process of recording their own story. On accessing the feature, an introductory prompt is automatically played followed by the first question whose answer will serve as that story’s ‘title’. Users are then presented a set of primary themes, optionally followed by a second set of themes preceded by another introductory prompt. The feature guides users through the sequence of questions or users can skip through the questions as they wish, with appropriate audio and visual feedback provided throughout. On hearing a question, users can replay the question, record their answer,

Fig. 10 E-Tool interface for listen to stories (*left*) and tell your own story (*right*)

stop recording, continue recording, listen to their answer, erase an answer, move on to the next question or save their story to the library. The tool supports a total of 13 questions. The maximum recording length per answer can be specified and is currently set to 5 min.

6.2 The rural Indian context

In India, there are wide disparities in socioeconomic conditions. Whereas the new economy centred on urban areas is giving rise to a comfortable middle class (and aspirations to those below), the traditional economy of the rural areas, largely based on farming, has left much of the rural population in extreme poverty with few options out [47, 48]. Over 70% of the Indian population still lives in rural areas, many barely surviving on subsistence farming, seasonal work and occupations dictated by caste with incomes below the international poverty line. Small plots, land depletion, poor yields, usurious lending practices and corrupt buying agents all conspire to put many farming households into a downward spiral of debt. A crop failure under such conditions has disastrous effects, pushing many farmers into despair and in certain states suicide among farmers has reached crisis proportions. Although the government tries to regulate many aspects of farming, unregulated and corrupt practices persist. Villagers view higher education as the only way for their children to aspire to a better future. However, while primary and secondary education is government funded and locally accessible, higher education is not and there are many private institutions of questionable merit. Word-of-mouth is the only source of information for many parents and children regarding career choices and available education programs. We focus our study on these two problem areas, namely

1. The problems faced by farmers and
2. The problems faced by parents and children when considering higher education.

6.3 The experiment

A total of three studies were conducted—two near Chennai in Tamil Nadu state and one near Bangalore in Karnataka state. All three took place in farming villages typical of the rural Indian context. In Tamil Nadu, stories on farming and higher education were elicited, while in Karnataka, the topic was farming only. Altogether 30 stories were collected, 17 on farming and 13 on higher education. These were told by both male and female participants representing a broad age range, from children to the elderly and a cross section of financial situations, from the very poor to those considered well off by local standards. In Tamil Nadu, the farming stories were collected on the main street of the village and in local homes over a 2-day period. In Karnataka, they were collected in the office of a local NGO involved with farmers over the course of an afternoon. The stories pertaining to education were collected at two local schools over 2 days.

The farmers who participated were primarily male, with only 3 female participants. In Tamil Nadu, they were recruited informally off the street, whereas in Karnataka the NGO invited farmers with whom they regularly work. Participants were 30 years old and over, including two over 60. While one participant had never attended school, the majority had completed some schooling, but only 4 had a higher education (mainly agronomy) and spoke English. The size of the farmers' plots varied from 1 to 10 acres, and farming was the primary activity of all but two. Their financial situation varied from very poor to comparatively well off. Only two did not need to borrow money to finance their farming activities.

In the case of higher education, both the parent and child participated in telling a story. The children consisted of 5 girls and 8 boys aged 15–17, all in the last 2 years of high school and among the top in their class. This, along with availability of a parent, was the basis for recruitment. A higher number of mothers participated, likely due to the sessions taking place at local schools in the middle of the day. None of the parents had a higher education themselves, all were employed in typical traditional activities and some were from among the poorest in the village.

In all three studies, local people assisted in setting up the study, recruiting participants and identifying relevant themes. In the case of farming, the focus was on collecting general information (family, land, and water, crops cultivated, seed, use of fertiliser and pesticides, manual and mechanised labour, and financial aspects). Regarding education, parents were asked to provide background information about themselves, their child, the subject they wanted their child to pursue, financing higher studies and what information they might need. Children were asked about what they liked doing, their school, what career they envisaged and what studies this would require. The themes were formulated into open-ended questions, translated and recorded in the local language using the laptop computer. Respected individuals from the community were then asked to record their story to serve as an example.

The actual sessions were conducted informally, and group participation was encouraged. On arriving, participants were either shown the video or given a brief explanation of the purpose of the study in the local language. They were then shown how to operate the application and invited to record their own story. Each participant recorded his or her story in turn while the others listened. Late arrivals quickly picked up what was going on by observing the others. On completing their story, participants were photographed and offered a small gift (monetary or a box of candy). This photograph would serve to identify their story within the story library.

6.4 Outcome of experiment

The acceptability of the SDS approach exceeded our expectations. All the participants were able to tell their story and were enthusiastic about doing so. Villagers participated readily and quickly picked up the operation of the application. Once they began talking, they became engaged in telling their story and were not distracted by the mechanics of recording. While in almost all cases they participated in groups, their stories were highly personal and did not show any signs of 'groupthink'. At the same time, the group provided an audience for the teller, making the narration a natural communicative exchange. Interestingly, when we first described our study to the personnel of the Karnataka NGO, they were convinced that an informal, group approach could not provide the information we sought. They advised us, based on their experience, that we must interview each farmer individually in depth, as otherwise '*you won't get the answers you want*'. Nonetheless, we proceeded with our experiment, following which the NGO personnel expressed their astonishment at the richness of the stories collected, contrary to their expectations.

Our analysis of the stories indicates that they are highly useful in identifying the participants' concerns and reveal an abundance of contextual information. While a full discussion is out of scope, to support this position and demonstrate how the SDS process is applied, we present examples from our analysis of the farming stories. As illustrated in Fig. 11, the information extracted from the transcribed stories is fully sufficient to construct a meaningful domain model highlighting the major concepts, concerns and relations. Based on the cultural analysis indicating strong experiential tendencies, we established the related cultural inhibitors (i.e. averse to disruptive change, reluctant to experiment, rely on knowledge conveyed through concrete experience, not predisposed to speculation or formulating abstract plans, reluctant to act individually, favour high-context personal communication).

From the prioritised list of issues, profitability emerges as the primary concern. Modelling profitability as a

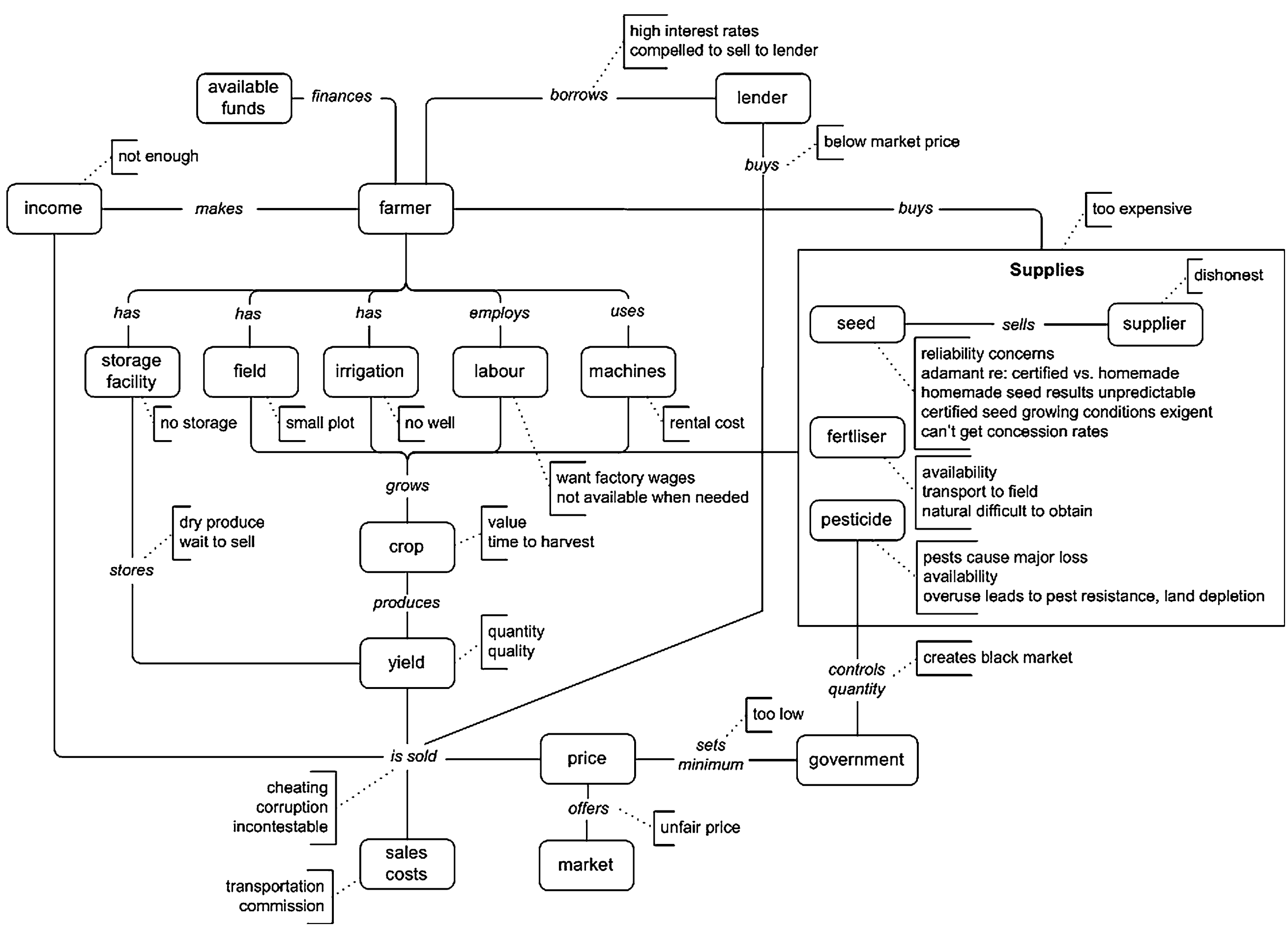


Fig. 11 Problem domain model describing farming

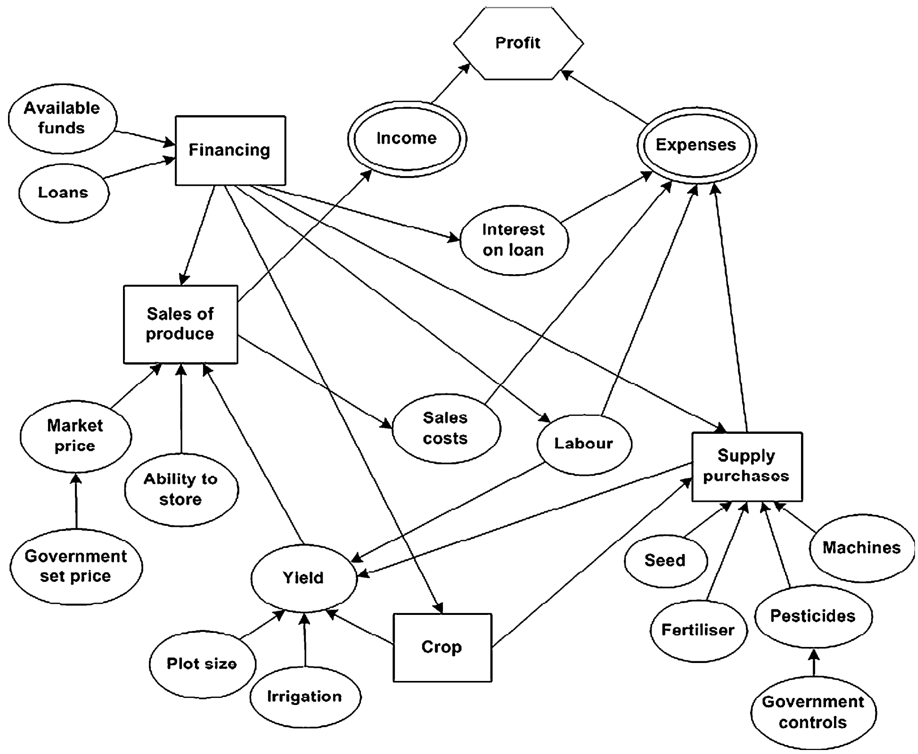


Fig. 12 Influence diagram displaying the causal relationships between the factors contributing to a farmer's profitability

decision situation with profit the variable to optimise produces the influence diagram depicted in Fig. 12. Briefly interpreted, profit is income less expenses, with expenses functionally determined by decisions concerning supply purchases, which in turn influence the yield (i.e. what seed, fertiliser and pesticides are applied influence the quantity and quality obtained). Supply decisions are influenced by decisions regarding crop (i.e. different crops have different requirements), and both crop and supply decisions are influenced by financing decisions, which in turn are based on available funds and/or loans. Examining the income path, income is functionally determined by the sales decision, informed by the market price (informed by the government set price), the ability to store produce (thus dry the produce and wait for better prices) and the yield, with the sales decision influenced by the financing decision (as farmers with loans are obliged to sell to the lender).

A cursory analysis of the influence diagram reveals significant external constraints on the farmers' choices exacerbated by serious negative feedback loops. The profile analysis related to profitability establishes a difference between small versus large landholders (less than 5 and 5–10 acres, respectively). Small landholders typically lack funds, have small plots, no wells and no storage facilities. Mapping the consequences of these external inhibitors produces the causal loop diagram depicted in Fig. 13, with four major feedback loops. The positive *profit cycle* consists of farmers spending available funds to purchase supplies to grow crops, producing a yield whose sale provides an income, increasing available funds. Concurrently, the purchase of supplies incurs expenses that are deducted from available funds, resulting in a negative *expenses loop* that balances the profit cycle and maintains equilibrium. However, while there is a delay between growing a crop and obtaining a yield, the effect of expenses on available funds is immediate. Effectively, farmers must invest in their crop upfront and only obtain an income after the yield is sold. Farmers with insufficient funds will reduce their supply purchases (cheaper seed, less fertiliser, less pesticide, etc.) to the detriment of their crop and have a higher financial risk, thus discouraging them from experimenting with new methods, to the overall detriment of income. Over time, this *insufficient funds* loop has a negative reinforcing effect on available funds through lower yields, land depletion, resulting from inadequate investment in the land. While certain farmers may persevere in this declining state, others are obliged to borrow, putting them into a negative *debt spiral* from which few recover. Borrowing money incurs high interest charges, increases financial risk, impels farmers to seek short-term returns (to pay off the loans) and compels them to sell to the lender at prices below the market rate, all of which have a deleterious effect on income. Lacking funds to begin with, the reduced income they make is insufficient, obliging them to borrow in an ever downward spiral of debt. All of these constitute hard constraints that influence the farmers' choices and impede their ability to act.

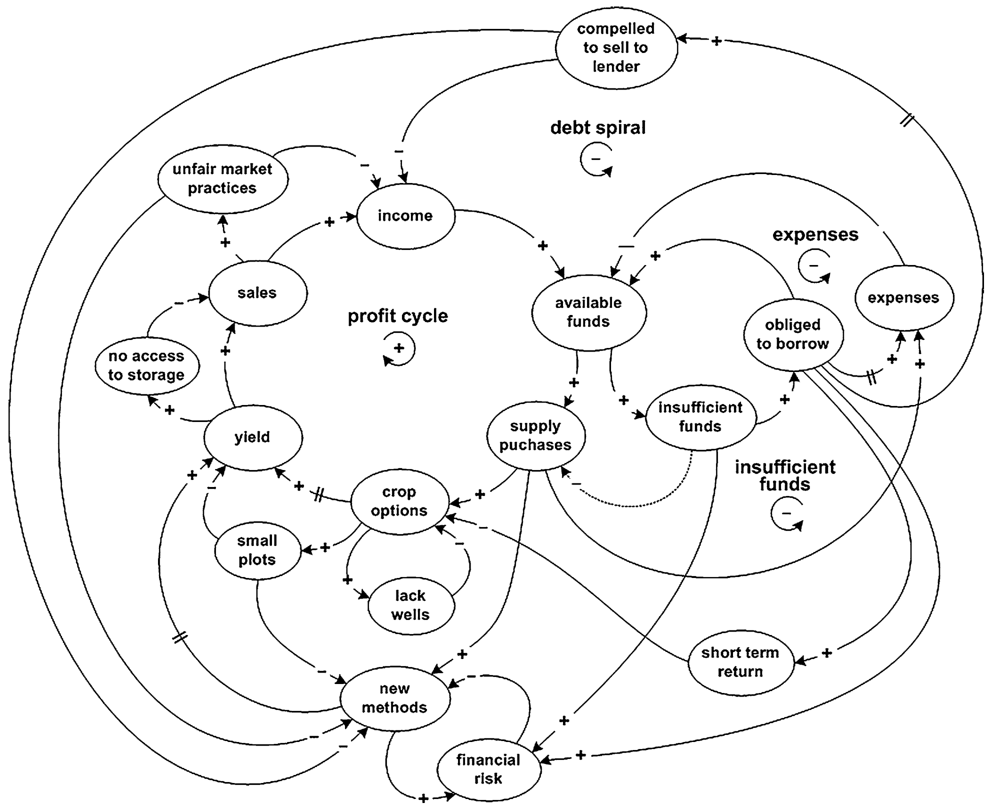


Fig. 13 High-level causal loop diagram of the farming context

We now seek ways of improving the farmers' profitability. From the related influence diagram (Fig. 12), we derive the high-level goal model depicted in Fig. 14. This indicates that profit can be improved by increasing income or reducing expenses, with an increase in income achievable by improving sales options (better prices, less cheating, etc.), yield (new methods, higher quality produce, etc.) and crop (better varieties, more valuable crops). However, sales are influenced by financing choices and lacking funds; many farmers are obliged to borrow and thus compelled to sell to the lender. Consequently, many will be unable to benefit from improved sales conditions. Attempts to improve the yield or crop will encounter the same barrier, especially if these involve additional costs. Effectively, farmers who are in—or susceptible to falling into—a debt spiral lack funds and cannot assume financial risk. This deters them from improving their yield and impels them to seek short-term returns, limiting their choice of crop. Thus, any solution that seeks to increase income must counter the obstacles emanating from insufficient funds and the debt spiral.

Analysing the goal model in this way, potential interventions are assessed and viable subgoals elaborated. For example, providing information on 'honest' seed suppliers (assuming 'honest' suppliers exist) can reduce supply costs (through better rates) and potentially increase yields (with better quality seed and germination rates) without introducing additional expenses. The prioritised issues concerning seed indicate that reliability is the major concern, with many farmers distrusting suppliers because certain suppliers are dishonest. Expanding the supplier branch, distrust is likely reinforced by the cultural inhibitor of low-context communication between suppliers and farmers. The farmers' inability to assume financial risk acts as a further inhibitor. To offset these inhibitors, we introduce the enablers 'no financial risk' and 'trusted suppliers'. Both are reinforced by 'positive concrete experience', which itself is positively reinforced by 'observable results'. Trust in suppliers can also be reinforced by 'collective participation' (i.e. I myself have no opinion about the supplier, but others that I know have trust) or by 'high-context personal communication' with the supplier (i.e. he or she shares my context and is someone that I can relate to within my operational frame of reference). Thus, the goal of providing information on seed suppliers can be achieved by achieving the subgoal of 'trusted suppliers', reinforced by 'collective participation', 'high-context communication' and 'concrete experience', with the latter reinforced by 'observable results'. This is depicted in the detailed goal model presented in Fig. 15.

Once the enablers are laid out, different ways for realising them are considered. For example, observable results can be obtained by cultivating a demonstration plot in the village or by handing out seed samples to individual farmers to grow. However, such approaches are not readily scalable, and in

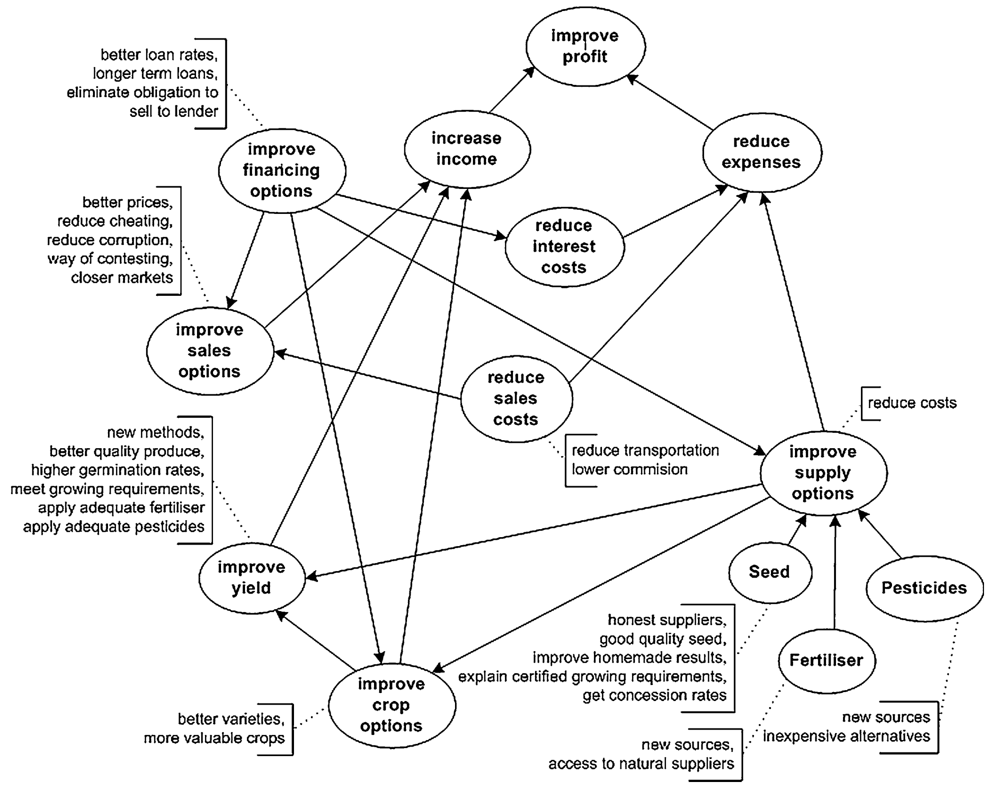


Fig. 14 High-level goal model derived from the influence diagram presented in Fig. 11, annotated with potential ways of addressing the concerns mentioned in the stories

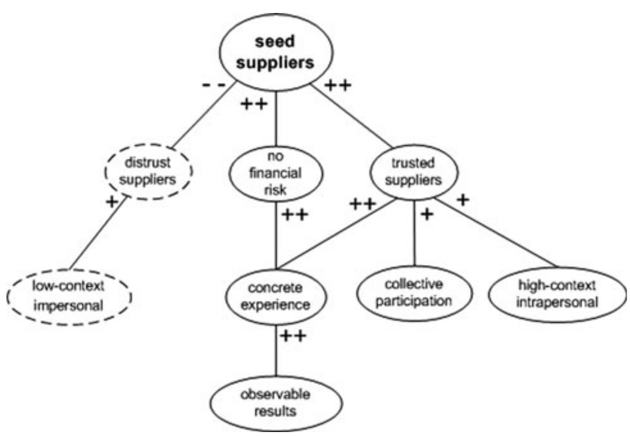


Fig. 15 Detailed goal model pertaining to providing information on seed suppliers with inhibitors in dashed ovals and enablers in solid ovals

the case of samples, the farmers' reluctance to experiment and act individually may limit their effectiveness. Videos showing demonstration plots might be a possible solution. We can also look at ways of increasing collective participation. One possibility is to establish communication with farmers who plant a particular seed variety, effectively creating a 'user group'. Another is to provide testimonials from satisfied farmers, presented in a situated, high-context, intrapersonal way, thus providing a form of indirect experience. The more recognisable the context and the farmers appearing in them, the more trustworthy, with personally known people the most trusted of all. Such an approach, which draws on the villagers themselves to appear in the content, has proven to be successful for building trust in agricultural extension work, with the possibility of appearing acting as a strong incentive to participate [49]. Trust can be further reinforced by high-context, intrapersonal communication with suppliers, through, e.g., village meetings, telephone or video conferencing. Thus, each enabler can be associated with multiple alternative sets of features, giving rise to the detailed goal model illustrated in Fig. 16.

With this example, we have demonstrated how the SDS methodology is applied to develop a detailed goal model for deriving software requirements. We also mention some practical findings regarding applying SDS in rural areas. Many of the narrations were high context, omitting contextual information that participants share in common and assume known. As the narrator's effusiveness appears to be correlated to his or her socioeconomic situation, we recommend recruiting participants from different socioeconomic strata to ensure sufficient background information is collected and to provide broad coverage for comparative purposes. We also recommend that local experts be included in the analysis. In addition, we found that participants do not restrict their answers to a theme when it is brought up; instead, information is spread across the entire narrative. Therefore, the analysis cannot be partitioned by theme as information relevant to a particular concept may be spread across the entire story.

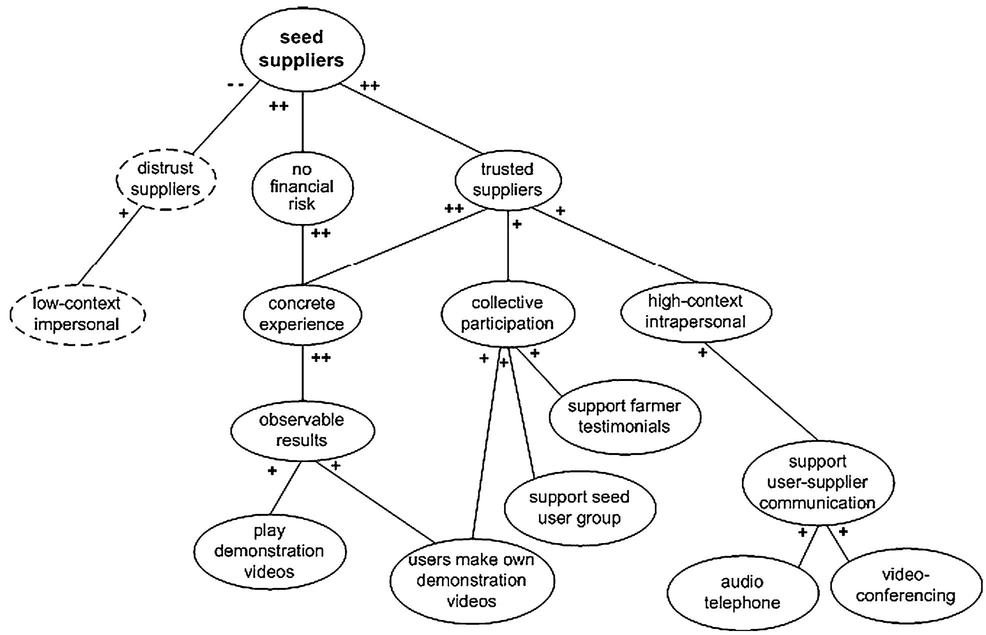


Fig. 16 Elaborated goal model pertaining to providing information on honest seed suppliers

This study clearly demonstrates that storytelling is an effective and acceptable way for people to express their concerns and needs with regards to some problem regardless of their literacy level. While in principle such a study would have been undertaken jointly with a development organisation, for logistic reasons this was not the case. Consequently, development experts were not consulted when selecting themes or analysing stories, nor was the analysis conducted with respect to any well-defined development goals. Instead, for our example, we selected a subgoal unconstrained by external inhibitors requiring interventions at the policy level (e.g. changes to lending practices, the market structure or access to funds). Nevertheless, despite our limited knowledge of local languages, customs and the domain, we were able to develop an informed understanding of the problem that goes well beyond what could be discovered during an equivalent period of 'scenic fieldwork'. Additionally, by conducting this experiment on different topics in different regions and languages, we demonstrate that this approach is readily adaptable to new contexts in terms of population group, language and topic.

Did the SDS methodology provide additional insight to the conventional RE process and how valid was the perspective obtained from the stories? Although the correctness and importance of the features derived in our study cannot be accurately assessed without a deployed system, they appear eminently appropriate based on the strengths and deficiencies documented in the ICT4D project literature. Consequently, the perspective on which the features are based appears to be sound. While we cannot guarantee that the data collected are complete and accurate or that its analysis produces a complete and accurate picture of the problem under study, we can take measures to ensure its validity. As with all data collection, the representativeness of the sample and the themes presented will affect the quality of the data. In the case of SDS, the participants' freedom to respond as they wish, although appearing to have no negative effects, requires further study. As to the validity of the interpretation, the skills of the analyst and the domain expertise available play a significant role. In addition to basic RE skills, applying SDS requires familiarity with qualitative research methods as well as the various modelling techniques used. Though not traditional to software engineering, these are well established in other disciplines, with known techniques for producing credible and trustworthy results. We contend, based on the premise that some information is better than no information that with SDS, we obtain a more informed view of the overall problem than would otherwise be available had the users' stories not been considered.

7 Summary and conclusion

The emergence of mobile phone communication and affordable ICT provide new opportunities for addressing critical social problems in the developing world. Realising this potential requires developing appropriate software applications to deliver relevant information in a manner whereby the intended audience can benefit from it. The nature of the audience targeted by these applications and the types of problems they face make this a challenging software design problem for which conventional requirements elicitation techniques are inadequate.

The SDS methodology presented in this paper is specifically designed to address the shortcomings conventional techniques present when eliciting requirements in an ICT4D context. In particular, it tackles the issues of inadequate attention paid to a project's high-level social development goals, neglect of environmental constraints and a lack of input from end-users. The lack of participation by end-users in elicitation can be attributed to the difficulty that people with limited literacy have articulating their problems and needs through conventional interview media. Without adequate input from end-users regarding their specific needs and socio-cultural context, it is difficult to ascertain the socioeconomic factors affecting a project's sustainability and ultimate ability to attain its stated social development goals. The methodology we propose draws on established theories and techniques from a range of disciplines, such as storytelling, communications theory, modelling in decision analysis and systems thinking, goal-oriented requirements engineering and requirements traceability. These techniques are combined in an original way to produce an approach that addresses the specific challenges of requirements engineering in the ICT4D domain.

Using the novel SDS technique whereby users express their needs through stories, we overcome the barriers of language, social class and literacy to elicit input from end-users from disadvantaged socioeconomic backgrounds. A storytelling approach has the added advantage of leveraging the customary mode of communication prevalent within experiential rural societies. We thus obtain a more complete understanding of the problem and local conditions from the bottom up, based on which one can prioritise the users' concerns and establish what obstacles any particular intervention must overcome for its target audience to be able to fully benefit. For example, in our farming study, insufficient funds and debt emerged as hard barriers preventing farmers from being able to benefit from interventions that incur additional expenses, increase financial risk or involve a long-term investment, with debt imposing an additional barrier on the ability to benefit from improved marketing conditions. Expanding the subgoal of providing information on honest seed suppliers, we identified financial risk and distrust of suppliers as the primary obstacles. We then drew on our cultural model to find appropriate ways of conveying information in order to surmount these obstacles.

Our cultural model, derived from Communications Theory, provides a more nuanced view of the deep aspects of culture that condition how people assimilate and use information in their daily lives. Applying this model, we can identify cultural constraints that might otherwise go unobserved and that can readily be mapped onto operational constraints and requirements. In the case of information on seed suppliers, we determined that the obstacles could be overcome by making the information more acceptable and trustworthy to its intended users. To accomplish this, we identified the additional requirements of providing positive concrete experience through observable results obtained by other farmers, supporting collective participation by the farmers and high-context intrapersonal communication with suppliers. The operationalisation of these requirements led to the identification of system features pertaining to playing and creating demonstration videos, supporting seed user groups and farmer testimonials and providing video and/or audio-conferencing with suppliers in addition to delivering information on suppliers.

The SDS process we have outlined provides a systematic method for extracting, relating and interpreting the diverse concerns that emerge from the stories into a holistic view of the overall problem in terms of both its static structure and dynamic behaviour. Through an incremental process of qualitative analysis and abstraction, the oral narratives expressed in an experiential mode are transformed into an analytical representation suitable for software analysts. By applying the SDS methodology the analyst can prioritise the needs of end-users and thus ensure that project requirements are driven by the needs of the very people that they are intended to benefit. Moreover, the process of telling and listening to stories has the potential side benefit of raising the local population's awareness about the social problems they face and engaging them in the development effort.

Our case study clearly demonstrates that the SDS approach is both feasible and effective in eliciting non-trivial information with regards to software requirements. For example, in the case of our seed example, we found that if we neglect to address the farmers' aversion to financial risk and their distrust of suppliers, this would seriously affect the farmers' ability to benefit from any ICT4D project that aims to provide information on seed suppliers. The approach we have proposed is highly adaptable. Different topics can be addressed by changing the themes, and different linguistic regions are easily supported by simply translating and rerecording the application prompts. By conducting our experiment on different topics in different regions, we have demonstrated that the process is repeatable. As an elicitation technique, SDS demands comparatively few resources: preparatory work is minimal, skilled facilitators are not required, a tool is available and data collection is relatively fast and accomplished at the participants' convenience. The data analysis step requires more specialised skills in qualitative research methods and modelling and is somewhat complex. However, it follows established techniques and thus is relatively straightforward to apply once the techniques are known.

With enhancements, such as porting the elicitation application to a mobile phone platform and providing upload and download capabilities through the Internet, the collection process can be made more efficient and more widely accessible to participants, domain experts and software analysts. We are also developing a Dialogue Editor that will allow non-technical people such as domain experts, to deploy the E-Tool directly (i.e. specify, record and upload questions onto a mobile phone and download the elicited stories). Scaling up the translation and analysis to handle large collections of oral stories poses a different problem for which innovative approaches combining low cost man-power and automated machine capabilities are required. We are investigating this in our further research.

In this study, we have shown how qualitative data collected directly from the grass roots can be integrated into a conventional top-down software development approach to provide a more comprehensive view of the problem being addressed. While the features derived from this expanded view align with the experience reported in the ICT4D literature, without a deployed system, we cannot assess how well the resulting system would actually meet the needs of real users. Nonetheless, based on the positive results of our experiment, we maintain that the SDS methodology shows promise in identifying requirements that will help improve ICT4D project outcomes. The early involvement of end-users is a well-known software engineering principle, acknowledged as helping reduce software project problems and failures. We believe that the SDS methodology allows us to obtain such early input from users and thus can contribute to developing more successful systems. To test this claim, in the next phase of our research, we intend to apply the SDS methodology in the context of actual ICT4D projects and evaluate the resulting systems in the field with real users.

References

1. Heeks R (2008) ICT4D 2.0: The next phase of applying ICT for international development. IEEE Comput 41(6):26–33
2. Frohlich DM, Rachovides D, Riga K, Bhat R, Frank M, Edirisinghe E, Wickramanayaka D, Jones M, Harwood W (2009) StoryBank: mobile digital storytelling in a development context. In: Proceedings of the international conference on human factors in computing systems, pp 1761–1770
3. Pressman RS (2005) Software engineering: a practitioner's approach. McGraw-Hill, Boston
4. Taylor A (2000) IT projects: sink or swim? Comput Bull 41(1): 24–26
5. Boehm B (2006) A view of 20th and 21st century software engineering. In: Proceedings of the 28th international conference on software engineering, pp 12–29
6. Cheng BHC, Atlee JM (2007) Research directions in requirements engineering. In: Proceedings of workshop on the future of software engineering (ICSE'07), pp 285–303
7. Van Lamsweerde A (2001) Goal-oriented requirements engineering: a guided tour. In: Proceedings of the IEEE international symposium on requirements engineering, pp 149–163
8. Chung L, Nixon BA, Yu E, Mylopoulos J (2000) Non-functional requirements in software engineering. Kluwer Academic Publishers, Dordrecht
9. Ramirez R (2001) A model for rural and remote information and communication technologies: a Canadian exploration. Telecommun Policy 25(5):315–330
10. Tongia R, Subrahmanian E (2006) Information and communications technology for development (ICT4D)—a design challenge? In: Proceedings of the international conference on information and communication technologies and development, pp 243–255
11. Unwin T (2009) ICT4D: information and communication technology for development. Cambridge University Press, London
12. Garai A, Shadrach B (2006) Taking ICT to every Indian village: opportunities and challenges. One World South Asia, New Delhi. http://ssrn.com/abstract=897910
13. Warschauer M (2003) Technology and social inclusion: rethinking the digital divide. The MIT Press, Cambridge
14. Venkatesh V, Morris MG, Davis GB, Davis FD (2003) User acceptance of information technology: toward a unified view. MIS Q 27(3):425–478
15. Koch JL, Caradonna TM (2006) Technologies and business models that work in developing countries. In: Proceedings of the international conference on information and communication technologies and development, pp 193–202
16. Chambers R (1994) Participatory rural appraisal (PRA): challenges, potentials and paradigm. World Dev 22(10):1437–1454
17. Hands K, Peiris DR, Gregor P (2004) Development of a computer-based interviewing tool to enhance the requirements gathering process. Requir Eng 9(3):204–216
18. Snowden D (1999) Story telling: an old skill in a new context. Bus Inf Rev 16(1):30–37
19. Gausepohl KA (2008) Investigation of storytelling as a requirements elicitation method for medical devices. Masters Thesis in industrial and systems engineering, Virginia Polytechnic Institute
20. Orbuch T (1997) People's accounts count: the sociology of accounts. Annu Rev Sociol 23(1):455–478
21. Kerr D (2003) "We know what the problem is": using oral history to develop a collaborative analysis of homelessness from the bottom up. Oral Hist Rev 30(1):27–45
22. Howard RA, Matheson JE (2005) Influence diagrams. Decis Anal 2(3):127–143
23. Sterman JD (2001) System dynamics modeling: tools for learning in a complex world. Calif Manag Rev 43(4):8–25
24. Forrester JW (1994) System dynamics, systems thinking and soft OR. Syst Dyn Rev 10(2–3):245–256
25. Hirsch GB, Levine R, Miller RL (2007) Using system dynamics modeling to understand the impact of social change initiatives. Am J Community Psychol 39(3–4):239–253
26. Gullivan M, Srite M (2005) Information technology and culture: identifying fragmentary and holistic perspectives of culture. Inf Organ 15(4):295–338
27. Hofstede G (2005) Cultures and organizations: software of the mind. McGraw-Hill, New York
28. Franklin U (1990) The real world of technology. Anansi Press, Concord
29. Kersten GE, Kersten MA, Rakowski WM (2002) Software and culture: beyond the internationalization of the interface. J Glob Inf Manag 10(4):86–101
30. Esselink B (2003) The evolution of localization. Guide to localization. http://www.isg.urv.es/library/papers/Esselink_Evolution.pdf. Accessed 7 Nov 2007

31. Myers MD, Tan FB (2003) Beyond models of national culture in information systems research. Adv Topics Glob Inf Manag, IGI Publishing, pp 14–29
32. Millen DR (2000) Rapid ethnography: time deepening strategies for HCI field research. In: Proceedings of the designing interactive systems: processes, practices, methods, and techniques, pp 280–286
33. Dourish P (2006) Implications for design. In: Proceedings of the SIGCHI conference on human factors in computing systems, pp 541–550
34. Rogers Y (2004) New theoretical approaches for human-computer interaction. Annu Rev Inf Sci Technol 38:87–143
35. Beale R, Dearden A, Dray S, Jackson A, Jones M, Kam M, Nocera J, Winters D (2009) Human-computer interaction for development (HCI4D): who, what, where, when and how. In: Proceedings of the conference on computer human interaction (CHI09)
36. Ong WJ (2002) Orality and literacy: the technologizing of the word. Routledge, New York
37. Babe R (2000) The communication thought of Harold Innis (1894–1952). In: Babe R (ed) Canadian communication thought: ten foundational writers. University of Toronto Press, Toronto
38. Couch C (1996) Information technologies and social orders. Aldine de Gruyer, New Brunswick
39. Havelock EA (1963) Preface to plato. Belknap/Harvard University Press, Cambridge
40. Taylor C (2008) Trafficking in facts: writing practices in social work. Qual Soc Work 7(1):25–42
41. Leffingwell D, Widrig D (2003) Managing software requirements: a use case approach. Addison-Wesley, Boston
42. Rolland C, Proix C (1992) A natural language approach for requirements engineering. In: Proceedings of CAiSE 1992 on advanced information systems engineering, Manchester, UK
43. Matulevičius R, Heymans P, Guttorm S (2006) Comparing goal-modelling tools with the RE-tool evaluation approach. Inf Technol Control 35(3A): 276–286
44. Medhi I, Sagar A, Toyama K (2007) Text-free user interfaces for illiterate and semi-literate users. Inf Technol Int Dev 4(1):37–50
45. Medhi I, Toyama K (2007) Full-context videos for first-time, non-literate pc users. In: International conference on information and communication technologies and development, Bangalore, India
46. Pitula K (2010). On requirements elicitation for software projects in ICT for development. Doctoral thesis in computer science and software engineering, Concordia University. http://www.gradworks.umi.com/NR/71/NR71106.html
47. Robinson/Sunna S (2007). Seeds of despair. Time.com, May 17. http://www.time.com/time/magazine/article/0,9171,1622180,00.html. Accessed 9 Dec 2009
48. Rezwan (2009) India: a wave of suicides among farmers. Global Voices Online, August 25. http://www.globalvoicesonline.org/2009/08/25/india-a-wave-of-suicides-among-farmers. Accessed 12 Sept 2009
49. Gandhi R, Veeraraghavan R, Toyama K, Ramprasad V (2009) Digital green: participatory video for agricultural extension. Inf Technol Int Dev 5(1):1–15