

Deriving Information Requirements from Responsibility Models

Ian Sommerville, Russell Lock, Tim Storer, and John Dobson

School of Computer Science, University of St Andrews, St Andrews, Scotland

`ifs@cs.st-andrews.ac.uk`,

WWW home page: <http://www.cs.st-andrews.ac.uk/~ifs>

Abstract. This paper describes research in understanding the requirements for complex information systems that are constructed from one or more generic COTS systems. We argue that, in these cases, behavioural requirements are largely defined by the underlying system and that the goal of the requirements engineering process is to understand the information requirements of system stakeholders. We discuss this notion of information requirements and propose that an understanding of how a socio-technical system is structured in terms of responsibilities is an effective way of discovering this type of requirement. We introduce the idea of responsibility modelling and show, using an example drawn from the domain of emergency planning, how a responsibility model can be used to derive information requirements for a system that coordinates the multiple agencies dealing with an emergency.

1 Introduction

There has been an accelerating trend for information systems development to be based on off-the-shelf products or ERP (Enterprise Resource Planning) systems. Such systems may incorporate or link with existing information systems, operated by different parts of a business or by different agencies. Examples of such systems include medical records systems, so-called ‘enterprise systems’ that integrate the functions of an enterprise and coordination systems that support different organizations who are working together on a shared problem.

The requirements for these information systems are significantly constrained by the system or systems on which they are based. Rather than focus on functionality and behavioural characteristics, requirements engineering for such systems should be concerned with how they can be configured and used to support the work of an organisation.

Our current research focuses on the engineering of these large-scale, organizational information systems and their use in a socio-technical environment. The aspect of this research described here is concerned with discovering and understanding the requirements for information systems that coordinate the work of multiple cooperating agencies. The novel features of the work include:

1. A focus on information requirements as the most significant type of requirements for complex organizational systems.

2. The use of models of the responsibilities in organizations to support the process of information requirements discovery.

We argue that the key requirements for many organizational systems are ‘information requirements’ i.e. requirements that define the information provided to stakeholders to help them do their work, requirements for information sharing and access control and requirements for the information that is generated. These are an essential basis for system configuration and for defining the information in different databases that has to be shared.

In this paper, we discuss what we mean by information requirements and introduce the concept of responsibility as a natural abstraction that supports the discovery of information requirements. We discuss how a model of responsibilities in a system can help understand and analyze these information requirements.

The responsibility models used illustrate emergency planning where several agencies must be coordinated to deal with some emergency such as flooding or a serious accident. We show how information requirements for a coordination system can be discovered from the model. This responsibility-based approach to requirements discovery is compared with viewpoint-oriented and goal-based approaches and we present a qualitative evaluation of the work.

2 Information requirements

Current RE methods that support requirements discovery, by and large, assume that the outcome of the process is a behavioural description of the requirements. These describe the system’s functionality and set out how the system should behave in response to events and user inputs. However, we have found that there are circumstances where a behavioural approach to requirements description is inappropriate:

1. When the the system is constructed using COTS products or ERP systems. Here, the planned system behaviour is determined to a large extent by the underlying generic system.
2. When the requirements are for a system of systems, where there are existing systems that must integrate with a new system. These constrain the behaviour of any new system. This is particularly true where the different systems are owned by different agencies, who may be reluctant to make changes to them.
3. When the requirements have to reflect the social and organizational context of a system rather than its operational use. The social and organizational context establishes the rules, regulations and policies that apply to a system as well as organization-specific requirements that reflect the power structures, culture and ethical values of an organization.

One motivation for introducing COTS and ERP systems is to support new business processes that reduce information processing inefficiencies by sharing of information across business units. For example, information about students

who apply to a university may be captured on application and automatically made available to other functions such as examination, assessment and alumni tracking. Systems may then be configured to ensure that the people involved in the business processes have access to the information that they need.

The key problem for such systems is understanding the information needs of the people in the organization who are responsible for the different functions of the system. These 'information requirements' are used as a basis for defining the system data models, business rules, information flows and generated reports. They should also include knowledge of who needs what information and when it is required to help avoid information overload and to configure the security features of the system.

The term 'information requirements' is not widely used. Here, we refer to those requirements concerning information to be provided to system stakeholders including where and how that information is used; or the information that they are expected to create using the system. These are distinct from both presentation requirements, which reflect how that information is delivered to different stakeholders and processing requirements, which define the automated information processing carried out by the system.

Organizations installing new enterprise information systems usually define new business process models and attempt to derive the information requirements by analyzing these models. However, there have been many anecdotal reports of major problems with such systems; they have often failed to deliver the expected improvements. While there are many different reasons for the problems, we believe that one important factor is that too many organizations rely on a simplistic notion of business processes:

1. *Defined business processes are rarely followed by the people doing the work in an organization.* Almost without exception, ethnographic studies of work have revealed that business process models are not a good description of actual operational processes [1, 2]. Individuals and teams modify and extend their processes to cope with deficiencies in the systems that they use, to adapt these processes to local circumstances and to simplify exception handling.
2. *Business processes are not an appropriate way of representing some kinds of work.* For professionals, much of their work is knowledge-based and reflects their professional discipline, training and culture. Rather than follow a standard process, individuals decide how to organize and record their work, with different people, doing the same job, working in different ways.
3. *Business process descriptions do not (and cannot) define all possible exceptions and failures that may occur.* Exception handling is left to individuals, who use their detailed knowledge and local information to deal with problems.

Our own observations, over more than 15 years [3–5], suggest that local process changes involve maintaining additional, sometimes redundant, information to that provided by a computer-based system or using the affordances of the information representation to support collaboration and communication. These

support resilience, allowing work to continue in the event of system failure and make it easier for people to help others if staff are busy or unavailable.

We argue then that information requirements cannot just be derived from business process models. They also require a detailed understanding of how work is actually done and how information is acquired, used and generated in its undertaking.

Information requirements, as discussed here, are also profoundly influenced by a wider class of requirements on a socio-technical system. These were first identified by the ORDIT project [6] and there called ‘organisational requirements’. We now prefer the term ‘enterprise requirements’ so as to avoid any possible confusion with ‘business requirements’, which reflect the functions and goals of the business.

Enterprise requirements are those requirements on a socio-technical system that are derived from the system being placed in a particular social and organisational context rather than from functions to be performed or tasks to be supported. Examples of sources of such requirements are power structures, responsibilities and obligations, control and autonomy, values and ethics. Requirements of this kind are embedded in organisational structure and policies, often in a way that cannot be directly observed or easily articulated. Therefore, these requirements are not so much captured as debated.

To illustrate what we mean here, consider the situation in hospitals where there is perennial (and probably inevitable) tension between the hospital administrators and the senior doctors. Information that is required to support administration is inevitably different from clinical information and providing that information often requires doctors to do extra work. If doctors are in a strong position within the organization, they may simply refuse to provide that information, thus constraining the information system. On the other hand, if the power struggle favours the hospital managers, then the doctors may comply with the demands to change the way they capture patient information. The information requirements depend on the power relationships in the organization rather than prioritizing stakeholder needs.

We believe that, for a large class of systems, information requirements are the most important requirements because it is often difficult to devise work arounds to cope with omissions and inadequate information. Problems are particularly acute where the information requirements of one stakeholder group are based on information that should be produced by some other group. If there is a mismatch here, then it may be very difficult or impossible to recreate the information required.

For example, in studies of a patient information system, built using a COTS product, it was found that the information requirements of clinicians were quite distinct from the information requirements of hospital managers. Managers required information about why patients were referred for treatment using standard classifications. However, these were quite different from the classifications used by the clinicians working in that domain. Reconciling these turned out to be very difficult indeed (partly because of tensions discussed above) [7].

We argue then, that where there are significant pre-existing limitations on system behaviour, the RE process should focus on understanding the information that needs to be provided to stakeholders to allow them to do their job, the ‘enterprise requirements’ that constrain the way that this information is used and the information that is generated. The focus is not ‘what should the system do’ but ‘what do the stakeholders need and produce’.

3 Responsibility

For some time, our interests have been in the interaction between humans, organisational structures and software systems which form constituent parts of broader socio-technical systems. In particular, we are interested in the influence which such issues have on the dependability of systems [8].

As part of this work, we have investigated how we can use an understanding of the responsibilities of agents in a socio-technical system to discover potential system vulnerabilities that can lead to system failure. In particular, we have been interested in highlighting the potential for ‘responsibility failure’ where an agent cannot properly discharge its assigned responsibilities, or where responsibilities are inappropriately assigned to agents in the system [9]. By a ‘responsibility’, we mean:

A duty, held by some agent, to achieve, maintain or avoid some given state, subject to conformance with organisational, social and cultural norms.

The key points in this definition are that a responsibility is a duty, which implies that the agent holding the responsibility is accountable to some authority for their actions, that responsibilities may be concerned with avoiding undesirable situations and not just with accomplishing some actions and that, in discharging responsibilities, agent behaviour is constrained by laws, regulations and social/cultural conventions and expectations.

In some cases, responsibilities may be procedural, where the agent follows a defined process. However, many responsibilities are knowledge-based and the agent assigned the responsibility decides how to discharge that responsibility, based on their knowledge and experience, local circumstances, etc. Even for procedural responsibilities, agents may have discretion in how to cope with exceptions, busy periods of work, etc.

One important reason for using responsibilities as an abstraction is that they are a natural notion that people can relate to. Work is often defined in terms of responsibilities and, if you ask people what their responsibilities are, they can usually tell you. People can usually separate their responsibilities at work from their responsibilities in other areas (e.g. as a parent).

Responsibilities serve as an effective basis for focusing the requirements debate, because of their ubiquity and naturalness. Different stakeholders may have quite different views about particular responsibilities and by making these responsibilities explicit, we provide a means to stimulate that debate.

A valid question, of course, is how do you discover and understand the responsibilities in a complex work setting? So far, our understanding of responsibilities has been based on three complementary approaches:

1. *Document analysis.* Documents that describe the work of a business such as plans or process descriptions often, explicitly or implicitly, discuss the responsibilities of the agents involved.
2. *Stakeholder interviews.* As discussed above, it is natural for people to talk about their own and other people's responsibilities.
3. *Field observations.* Ethnographic studies of work settings reveal how responsibilities are played out in practice and, in particular, how responsibilities may be delegated, transferred and shared. We have been involved in such studies in four areas - hospital administration [9], emergency planning [10], election systems [11] and, more recently, university administration.

4 Responsibility Modelling

A responsibility model is a succinct description of the responsibilities in a system, the agents who have been assigned these responsibilities and the resources that should be available to these agents to assist them in discharging their responsibilities.

The domain in which we are currently working is that of civil emergency management. This involves the emergency services and local and national government working together to cope with some emergency, such as flooding, terrorist attack, aircraft accident, etc. We have been investigating how we can use a model of responsibilities as a way of revealing planning vulnerabilities and the requirements for systems support by the emergency coordination centres. The case studies that we have used have been based on emergency plans for dealing with extensive flooding.

The role of the emergency coordination centres in the UK is to coordinate the 'work on the ground' of the different agencies involved. This involves prioritizing actions, maintaining an overview of the situation, facilitating information exchange across agencies and communicating with the media. The agencies involved (fire, police, ambulance, etc.) have their own command and control systems and information systems. Communication between agencies is normally managed by liaison officers in the coordination centre.

In this case, the requirements are for an information management system that allows information to be shared across agencies, that facilitates the transmission of information to people dealing with the emergency and that logs decisions taken by coordinators. Logging is critical both as a means of learning from experience after the recovery phase of the emergency has been completed and as an accountable record of actions.

Our responsibility modelling notation identifies responsibilities, agents (organizations or roles) who have been assigned these responsibilities and resources used to discharge responsibilities. Figure 1 is a model of the responsibilities involved in evacuating an area in the face of an imminent or actual threat (from

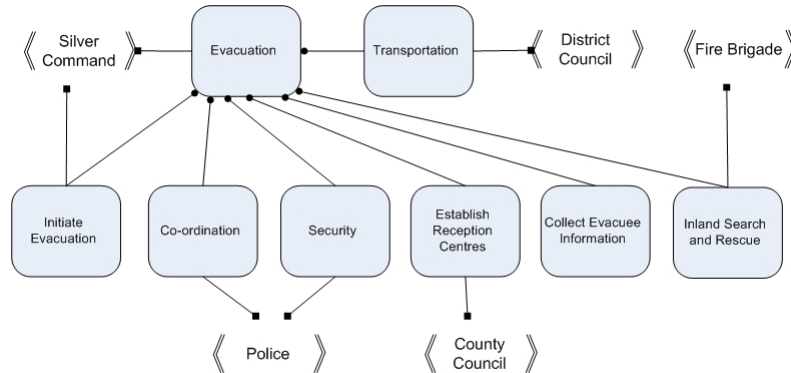


Fig. 1. Responsibility model of evacuation coordination

flooding). Responsibilities are denoted by round-edged rectangles and agents are named in angle brackets. Dependencies between responsibilities (such as decomposition into sub-responsibilities) are indicated by links between the responsibility icons.

Responsibilities have a set of attributes and an associated description that can take a number of forms, depending on the type of responsibility [12]. For responsibilities that are normally discharged in the same way, the description can be a workflow model. For knowledge-based responsibilities, the description is usually textual. Attributes may include the goal or goals associated with the responsibility, the context where it is normally discharged, pre and post-conditions defining assumptions, etc.

To explain Figure 1, Silver Command (the name given to the coordination centre) initiates an evacuation given situation information (flood warnings, weather forecasts, etc.) and a risk analysis. The police coordinate the evacuation process and are responsible for safety and security during the evacuation. If people are trapped, the fire service are responsible for search and rescue but they may not be involved if the evacuation is in advance of a predicted flood. Two levels of local authority are also involved. The district council must provide transport for the evacuees and the county council must provide safe accommodation, food and water, etc. Other agencies, such as the ambulance service, may also be called on if there are ill or infirm people to be evacuated, but we have simplified our model to exclude these.

Notice that the responsibility ‘Collect Evacuee Information’ does not have an agent associated with it. Drawing up the responsibility model revealed this vulnerability in the emergency plan, since it did not define which agency should collect this information.

We can also associate resources with responsibilities as shown in Figure 2. A resource can be information or some physical entity such as a flood barrier, vehicle, etc. In this case, the responsibility ‘Initiate Evacuation’ requires information resources (named between vertical bars) in order to discharge that responsibility.

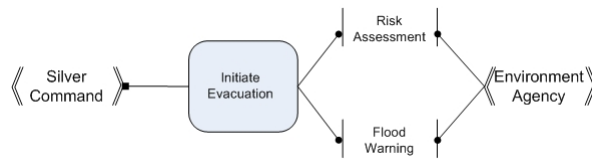


Fig. 2. Resources associated with a responsibility

The Environment Agency must provide a flood warning (which indicates when local rivers may flood in the near future) and a risk assessment, which shows the areas that would be affected by a flood and so should be evacuated.

5 Responsibility and requirements engineering

We are currently exploring how responsibilities can be used to support the early stages of the requirements engineering process, namely the discovery of analysis of requirements from multiple stakeholders. There are three areas where our work suggests they are useful:

- Understanding enterprise requirements
- Information requirements discovery.
- Stakeholder identification

We do not have room here to discuss all of these but, rather, focus on the use of responsibility models in discovering information requirements.

5.1 Information requirements

As we have discussed, we believe that for enterprise information systems, the critical requirements are information requirements. In the case of emergency management, different agencies maintain different types of information. The most important system requirements are to identify the information that has to be shared and how it will be communicated to the people who need that information.

The nature of emergency management is such that the people involved react to the situation on the ground in order to discharge their responsibilities; it is impossible to define an evacuation ‘process’. The information they require normally comes through liaison officers based in a coordination centre.

To help us understand the information requirements, we ask a range of questions, prompted by the responsibility model. These are:

1. What information is required to discharge this responsibility?
2. What channels are used to communicate this information?
3. Where does this information come from?

4. What information is recorded in the discharge of this responsibility and why?
5. What channels are used to communicate this recorded information?
6. What are the consequences if the information required is unavailable, inaccurate, incomplete, late, early?

To see how these questions are used, let us focus on the coordination of evacuation of people, which in the UK is the responsibility of the police, working through a local, mobile coordination centre. When we applied the above questions to the emergency flood plan for Cumbria (in the north of England), we derived the following:

1. *What information is required to discharge this coordination responsibility?*
 - A map of the area that is threatened by the emergency.
 - A list of 'priority premises' to be evacuated. These are premises that are evacuated first because they include people who may not be able to evacuate themselves. hospitals, schools, care homes, etc.
 - A list of assembly points where people should gather. The evacuation involves people being directed (or if necessary helped) to these local assembly points from where they are picked up and transported to a safe location.
 - Information about known, unsafe routes in the area.
 - Information about premises that are known to have been evacuated.
 - Information about the emerging threat situation (e.g. predictions of how long until flooding occurs).
 - Information about the capacity and availability of transport from the assembly points.
 - Information about number of police available to assist with evacuation and the availability of personnel from other emergency services. If the emergency is predicted and has not yet happened, there may be no need for support from other services.
2. *Where does this information come from?*
 - Area map: County council
 - Priority premises: District council
 - Assembly points: District council
 - Known, unsafe routes: Police
 - Evacuated premises: Police, Fire service
 - Threat situation: Environment agency
 - Transport information: District council
 - Personnel available: Police, Other services
3. *What channels are used to communicate this information?*
 - Area map: Radio data link to printers in local mobile command centre
 - Priority premises: Radio data link to printers in local mobile command centre
 - Assembly points: Radio data link to printers in local mobile command centre
 - Known unsafe routes: Radio from Silver command

- Evacuated premises: Radio from Silver command
 - Threat situation: Radio from Silver command
 - Transport information: Radio from Silver command
4. *What information is recorded and why?*
 - Information about premises evacuated, area evacuation times, units responsible for evacuation. (for accountability)
 - Information about unchecked premises. (in case of future emergency calls)
 - Information about unsafe routes (to assist evacuation)
 5. *What channels are used to communicate this information?*
 - Radio from ground units to local control centre. Email to Silver command if available, otherwise radio.
 - Word of mouth local reporting
 6. *What are the consequences if the information required is unavailable, inaccurate, incomplete, late, early?*

To assess this, we need to look at each information item in turn. Take, for example, the list of priority premises to be evacuated.

 - *Unavailable*: Manual premises check required to see if vulnerable people to be evacuated.
 - *Inaccurate*: Manual premises check may be necessary. Possible delay in evacuation of vulnerable people. People may be left behind.
 - *Incomplete*: Possible delay in evacuation.
 - *Late*: Information has to be communicated to units in the field rather than at local coordination centre.
 - *Early*: No consequence.

The information derived from discovering the information required to discharge a responsibility may then be recorded on a responsibility model, using the approach shown in Figure 2. This may then be used as a basis for debate and discussion about the information.

The final stage in the process is to translate the information requirements into system requirements for the system supporting information sharing. We do not have space to discuss this in any depth here but simply provide some examples of requirements (and associated rationale) below:

1. The coordination centre system shall be able to import textual information from the District Council planning system, the Police emergency system and the Fire Service emergency system. (Different types of information needs to be shared and this allows for information transfer between agencies).
2. The coordination centre system shall maintain a list of priority premises to be evacuated for each town in the local area. This shall be updated by the local council when the coordination centre is established. (The premises list is maintained by the local government authority but may not be immediately available outside of normal working hours; While a central list may be out of date, it is better than nothing.)

3. The coordination centre system shall maintain a list of premises evacuated along with the time of evacuation and the units involved in the evacuation. (Allows units involved in the evacuation to be coordinated. Maintains an audit trail of who did what and when.)
4. The coordination centre system shall notify all liaison officers of new information about the threat situation as it becomes available. (Different services may respond differently to changes in the threat situation e.g. local government may withdraw from a situation because they are not equipped to deal with search and rescue.)

6 Related work

Models of responsibility were first proposed by Blyth et al. in the ORDIT methodology [6, 13], a graphical notation for describing the responsibilities that agents hold with respect to one another. Dobson and Strens have also explored the use of responsibilities in requirements engineering [14, 15].

Dewsbury and Dobson have edited a collection of papers that describe much of the research undertaken on responsibility, presenting analyses of inappropriate responsibility allocation in socio-technical systems [16]. In particular, the work includes proposals for graphical responsibility modelling and pattern-based responsibility definition [9, 12].

Responsibilities provide a basis for deriving system requirements from different perspectives and, in that respect, have something in common with viewpoint-based approaches to RE. Viewpoints were probably first proposed by Schoman and Ross in SADT under a different name [17] and the first use of the term 'viewpoint' was in Mullery's paper on CORE [18]. CORE is based on information flow so is useful as a means of analyzing information requirements, once they have been discovered. Other researchers took on the general notion of viewpoints as a means to discover, organize and analyse requirements from different perspectives, using different approaches [19–22].

Responsibility-based approaches are perhaps closest to goal-based modelling approaches, such as i^* [23] and KAOS [24]. These are intended to expose high level dependencies between the goals associated with agents in a given system. Sub-goals may be derived from higher level objectives and assigned to agents for completion. Goals are achieved through the recursive achievement of some or all sub-goals. Relationships between sub-goals express (and, or etc) the possible ways in which the super-goal may be achieved. Analysis of such models can examine, for example, whether a super-goal may fail due to the failure of a single sub-goal (brittleness), or whether a particular agent has been overloaded with too many goals to achieve.

The notion of a responsibility includes the specification of objectives (goals) to be achieved in discharging the responsibility. However, there is also an acknowledgment that in complex socio-technical systems, the achievement of an objective is subject to a range of constraints, that are often implicit. For example, doctors discharge responsibilities subject to ethical constraints, companies

operate subject to the financial regulations of their host country. The notion of responsibility as a duty embodies an embedded assumption that it is how an agent acts and not just what is achieved that is important. This affects the information requirements, particularly those requirements that define the information that must be recorded when a responsibility is discharged. There may also be requirements to record information, in some critical contexts, about options considered and rejected so that the agent holding the responsibility may be held to account.

In a review of research on goal-oriented approaches, Lapouchnian [25] rightly states “Identifying goals is not an easy task”. He states that goals are normally derived from other information that is discovered from stakeholders. Responsibilities are, we believe, a more natural abstraction than goals for requirements discovery for a number of reasons:

1. Job specifications are often expressed in terms of responsibilities. Therefore, responsibilities are a common notion across all stakeholders in an enterprise. While managers and system operators (say) may not necessarily interpret responsibilities in the same way, they can both, at least, talk about them.
2. Business goals are often not effectively communicated to the workforce in an organization. Hence they may find it very difficult to distinguish between professional and personal goals. There is often widespread scepticism in organizations about business goals.
3. Goals are not, in our view, a natural way to express what we call ‘avoiding responsibilities’. They imply positive action to achieve the goal. A sentence such as ‘my goal is to ensure that the pressure does not become dangerously high; a much more likely way to express this is, ‘my responsibility is to ensure that the pressure does not become dangerously high’.

We do not claim that our responsibility-based approach is a substitute for or competes with goal-based approaches. Responsibility modelling operates at a different level of abstraction from goal-based modelling. Responsibility modelling could be used as a starting point for goal identification.

7 Evaluation

Evaluating any proposals for novel approaches to requirements engineering is difficult. There are practical and methodological problems in comparative evaluation, as the baseline knowledge of stakeholders changes as soon as they become involved in a trial. Even when requirements have been discovered, we have no way of telling, until the system has been implemented, if these requirements reflect the real needs of stakeholders. We may have invented an effective new way of discovering requirements but unless these are the right requirements, then our approach is not much use. We cannot therefore clearly demonstrate that focusing on responsibilities leads to better requirements. We can however say that responsibility modelling is a good way of stimulating requirements discussions and hence reduce the possibility that requirements are inappropriate.

We can consider the effectiveness of our responsibility-based approach from several perspectives:

Naturalness: Can stakeholders without experience of requirements engineering relate to the approach? In this respect, our responsibility-based approach scores highly. In emergency management, plans are usually expressed in terms of responsibilities so it is straightforward to understand the concept of a responsibility model. The notion of responsibility and responsible behaviour is widely used in everyday discourse so people can readily discuss their responsibilities in a given situation. The questions used to discover information requirements relate directly to the stakeholder's job and are therefore easy to understand. By contrast, notions such as viewpoints are not particularly easy to explain and, as already discussed, it is not easy to elicit stakeholder goals.

Applicability: How widely applicable is an approach? Because of the pervasive notion of responsibilities, we believe that this responsibility-based approach may be used in most socio-technical systems which involve multiple agencies or several units within the same organisation. So far, we have developed responsibility models in the domains of hospital administration, e-voting systems and emergency planning and are working on models of university administration.

Scaleability: Can the approach be used with real rather than simple example systems? All RE methods suffer from the danger of information explosion where an immense amount of information is generated. In some cases, this makes it practically impossible to scale these up to practical systems. As our approach has been derived from our experience with large-scale socio-technical systems, we have never started with 'toy' examples; we have only ever used this to help us analyse existing, large-scale systems. Of course, the natural structure of responsibilities helps here - it is everyday practice to decompose and delegate responsibilities.

User involvement: Have end-users users been involved in the research? In terms of buy-in from end-users, we are currently working closely with the Scottish Environmental Protection Agency in developing our approach to responsibility models. They are particularly interested in developing simple support systems for use in the field, which reduce the problems of getting the right information to the right person and the right time.

Complementarity: Does the proposed approach complement other methods of requirements engineering? We believe that a responsibility-based approach can be used alongside goal-based and viewpoint-based approaches. Responsibilities can be identified at an early stage in the process and used as a basis for discovering information requirements. By identifying the agents who hold responsibilities, stakeholders can be identified and involved in the RE process for the discovery of more detailed behavioural system requirements.

8 Conclusions

Relatively few complex information systems are now built from scratch but rather are assembled from existing and off-the-shelf systems. This means that behavioural requirements of the system are constrained and that the key requirements are the requirements that relate to the information provided to and generated by system stakeholders.

We have proposed an approach based on the notion of modelling responsibilities in an enterprise that supports the discovery of these information requirements. We argue that responsibilities are a natural abstraction to use in stakeholder communication and that stakeholders are readily able to articulate the information they require to discharge their responsibilities. Models of responsibility in a system provide a basis for stakeholders, potentially from different organizations, to discuss the information that the need to do their job. Hence, these models can support the discovery of information requirements.

We are in the process of using and evaluating these techniques to support emergency management. Here, a coordination system is required to ensure that information is exchanged between the different agencies involved in a timely way. Our initial discussions with end-users suggest that they can easily relate to a responsibility based approach and appreciate responsibility models as a succinct summary of who does what in emergency situations. They have proved to be an effective way of discovering and analysing information requirements.

References

1. Suchman, L.A.: *Plans and Situated Actions: The Problem of Human-Machine Communication*. Cambridge University Press (December 1987)
2. Myers, M.D.: Investigating information systems with ethnographic research. *Communications of the Association for Information Systems* **23**(2) (1999) 1–20
3. Bentley, R., Hughes, J.A., Randall, D., Rodden, T., Sawyer, P., Shapiro, D., Sommerville, I.: Ethnographically-informed systems design for air traffic control. In: *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW'92)*, Toronto, Ontario, Canada, ACM Press (November 1992) 123–129
4. Hughes, D.R.J.A., O'Brien, J., Rodden, T., Rouncefield, M., Sommerville, I., Tolmie, P.: Banking on the old technology: understanding the organisational context of legacy systems. *Communications of the Association for Information Systems* **2**(8) (1999) Article 8.
5. Clarke, K., Hartwood, M., Procter, R., Rouncefield, M., Slack, R.: “minus nine beds”: Some practical problems of integrating and interpreting information technology in a hospital trust. In: *Proceedings of the BCS Conference on Healthcare Computing: Current Perspectives on Healthcare Computing*, Harrogate, UK. (2002) 205–211
6. Blyth, A.J., Chudge, J., Dobson, J.E., Strens, M.R.: ORDIT: A new methodology to assist in the process of eliciting and modelling organisational requirements. In Kaplan, S., ed.: *Proceedings on the Conference on Organisational Computing Systems*, Milpitas, California, USA, ACM Press (1993) 216–227
7. Hardstone, G., d' Adderio, L., Williams, R.: Standardization, trust and dependability. [8] chapter 5 105–122

8. Clarke, K., ed.: *Trust in Technology: A Socio-Technical Perspective*. Springer (March 2006)
9. Sommerville, I.: Models for responsibility assignment. [16] chapter 8 165–186
10. Sommerville, I., Storer, T., Lock, R.: Responsibility modelling for contingency planning. In: *Workshop on Understanding Why Systems Fail, Contingency Planning and Longer Term Perspectives on Learning from Failure in Safety Critical Systems*. (June 2007)
11. Lock, R., Storer, T., Harvey, N., Hughes, C., Sommerville, I.: Observations of the Scottish elections 2007. *Transforming Government: People, Process and Policy* **2**(2) (2008) 104–118
12. Sommerville, I.: Causal responsibility models. [16] chapter 8
13. Dobson, J.E., Blyth, A.J., Chudge, J., Strens, R.: The ORDIT approach to organizational requirements. In Jirotko, M., Goguen, J.A., eds.: *Requirements Engineering, Social and Technical Issues*. Academic Press (June 1994) 87–106
14. Dobson, J.E., Strens, M.R.: Responsibility modelling as a technique for requirements definition. *Intelligent Systems Engineering* **3**(1) (1994) 20–26
15. Dobson, J.E., Strens, M.R.: Organizational requirements definition for information technology systems. In: *Proceedings of the IEEE International Conference on Requirements Engineering (ICRE94)*, Colorado Springs, IEEE Press (April 1994) 158–165
16. Dewsbury, G., Dobson, J., eds.: *Responsibility and Dependable Systems*. Springer-Verlag London Ltd (June 2007)
17. Ross, D.T., Jr., K.E.S.: Structured analysis for requirements definition. *IEEE Transactions on Software Engineering* **3**(1) (1977) 6–15
18. Mullery, G.P.: CORE : A method for controlled requirement expression. In: *Proceedings of the 4th International Conference on Software Engineering*, Munich, Germany, IEEE Computer Society (September 1979) 126–135
19. Finkelstein, A., Kramer, J., Nuseibeh, B., Finkelstein, L., Goedicke, M.: Viewpoints: A framework for integrating multiple perspectives in system development. *International Journal of Software Engineering and Knowledge Engineering* **2**(1) (1992) 31–57
20. Dardenne, A., Fickas, S., van Lamsweerde, A.: Goal-directed concept acquisition in requirements elicitation. In: *Proceedings of the Sixth International Workshop on Software Specification and Design*, IEEE Computer Society Press (October 1991) 14–21
21. Kotonya, G., Sommerville, I.: Requirements engineering with viewpoints. *BCS/IEE Software Engineering Journal* **11**(1) (1996) 5–18
22. Sommerville, I., Sawyer, P.: Viewpoints: Principles, problems and a practical approach to requirements engineering. *Annals of Software Engineering* **3** (1997) 101–130
23. Yu, E.S.: Towards modelling and reasoning support for early-phase requirements engineering. In: *3rd IEEE International Symposium on Requirements Engineering (RE'97)*, IEEE Computer Society (1997) 226–235
24. Dardenne, A., van Lamsweerde, A., Fickas, S.: Goal-directed requirements acquisition. *Science of Computer Programming* **20** (1993) 3–50
25. Lapouchnian, A.: Goal-oriented requirements engineering: An overview of the current research. Depth report, Department of Computer Science, University of Toronto (June 2005)