LIFE CASES
An Approach to Address Pragmatics in the Design of Web Information Systems

Klaus-Dieter Schewe
Massey University, Department of Information Systems & Information Science Research Centre, Palmerston North, New Zealand

Bernhard Thalheim
Christian Albrechts University Kiel, Department of Computer Science, 24098 Kiel, Germany

Keywords: Storyboarding, pragmatics, life case, application domain description.

Abstract: On a high level of abstraction a Web Information System (WIS) can be described by a storyboard, which in an abstract way specifies who will be using the system, in which way and for which goals. While syntax and semantics of storyboarding has been well explored, its pragmatics has not. This paper contributes the first step towards closing this gap. For this we present *life cases*, which capture observations of user behaviour in reality. We discuss the facets of life cases and present a semi-formal way for their documentation. Life cases can be used in a pragmatic way to specify a story space, which is an important component of a storyboard.

1 INTRODUCTION

A Web Information System (WIS) is an information system that can be accessed through the world-wide-web. On a high level of abstraction a WIS can be described by a storyboard, which in an abstract way specifies who will be using the system, in which way and for which goals. In a nutshell, a *storyboard* consists of three parts:

- a *story space*, which itself consists of a hierarchy of labelled directed graphs called *scenarios*, one of which is the main scenario, whereas the others define the details of *scenes*, i.e. nodes in a higher scenario, and a *plot* that is specified by an assignment-free process, in which the basic actions correspond to the labels of edges in the scenarios,
- a set of *actors*, i.e. abstractions of user groups that are defined by *roles*, which determine obligations and rights, and *user profiles*, which determine user preferences,
- and a set of *tasks* that are associated with *goals* the users may have.

In addition, there are many constraints comprising static, dynamic and deontic constraints for pre- and postconditions, triggering and enabling events, rights and obligations of roles, preference rules for user types, and other dependencies on the plot.

While syntax and semantics of storyboarding has been well explored, its pragmatics apart from the use of metaphors (Thalheim and Düsterhöft, 2000) has not. Pragmatics is part of semiotics, which is concerned with the relationship between signs, semantic concepts and things of reality. Main branches of semiotics are *syntactics*, which is concerned with the syntax, i.e. the construction of the language, *semantics*, which is concerned with the interpretation of the words of the language, and *pragmatics*, which is concerned with the current use of utterances by the user and context of words for the user. Pragmatics permits the use of a variety of semantics depending on the user, the application and the technical environment.

This paper contributes the first step towards pragmatics of storyboarding. For this we present life cases, which capture observations of user behaviour in reality. The idea is that these observations give rise to both a concrete scenario for the usage of the WIS including roles and user profiles. We then derive the candidate scenarios by abstraction. Further integration and refinement of these scenarios gives rise to the story space. We deal with the problem of extracting user models in a separate article. The third component of the storyboard, the tasks, has already been subject of strategic modelling. The life cases permit the refinement and further decomposition of these tasks.

Schewe K. and Thalheim B. (2007).
LIFE CASES - An Approach to Address Pragmatics in the Design of Web Information Systems.
In *Proceedings of the Third International Conference on Web Information Systems and Technologies - Web Interfaces and Applications*, pages 5-12
DOI: 10.5220/0001261000050012

2 RELATED WORK

Most methods for web application engineering such as OOHDM (Schwabe and Rossi, 1998), WebML (Ceri et al., 2003), HERA (Houben et al., 2003), WSDM (De Troyer and Leune, 1998) and variants of UML (Conallen, 2003; Lowe et al., 2002) pay little or even no attention to high-level modelling. The rationale underlying our work is that this is far too little to capture strategic and usage issues of WISs. The integration of goals and soft goals into the information systems development process has been proposed by (Mylopoulos et al., 2000; Giorgini et al., 2002). The detailed distinction of intentions as targets, objects, objectives and aims is based on linguistics. The integration of the temporal dimension is necessary because of the information systems context. The extension by the representational dimensions has been made in order to specify aspects of WISs.

Life cases extend and formalise scenarios used in software engineering and user modelling (Courage and Baxter, 2005). Early work like already discovered that human computer interfaces can be developed in techniques of movie writing or play development (Vale, 1982). Life cases have already been envisioned in (Carroll, 1991). Scenarios as defined in software engineering are examples of interaction sessions. This kind of scenario starts with an outline of the interaction, and during requirements elicitation, details are added to create a complete description of interaction. They are specified by a general description of the start and termination states of the system, a description of the normal flow of events, and a description of concurrent events (Sommerville, 2004). Later scenario research has been integrated into agile programming. Customer involvement is one of the requirement of agile methods. Customers should be integrated as early as only possible into the development process (Ambler, 2002). Both approaches are to much oriented towards systems and do not consider life cases as we do. Scenario development has been applied to development of websites in (Rosenfeld and Morville, 1998).

In (Harel and Marelly, 2003) life cases were integrated into the entire software engineering process. Recently, software engineering research developed an engineering approach to simple life case. Business use cases (Robertson and Robertson, 2006) generalise the requirements shell of (Robertson and Robertson, 1999) that is based on verbal description of event/use cases, requirement types, the rationale, originator, fit criterion, customer (dis)satisfaction, supporting materials, and the history. Our life case specification is not as complex as these business use cases. They combine intention, users, actors, constraints/assumptions/scope or context, tasks, migration, risks, costs, documentation, testing, and requirements for functions, data, look and feel, usability, humanity, operating, maintenance, support, security, culture, politics, legal and performance. We concentrate on the requirements the user has as such.

3 FACETS OF INTENTION

The description of intention is based on a clear understanding of aims and targets of the WIS including a specification of long-range and short-term targets, expected visitors, characteristics of this audience, tasks performed by the users, necessary and unnecessary content, and finally an understanding of restrictions of usage.

Utilisation scenarios are developed on the basis of intentions. An intention specifies what one purposes to accomplish or do, i.e. one has in mind to do or bring about. It has four facets that are based on the general characteristics of WISs:

Purpose facet: The *purpose* of stakeholders specifies an anticipated outcome that is intended or that guides the planned actions. It may be based on two additional pieces of information: The *aims* specify what is intended to be attained and what is believed to be attainable. The *objectives* are more general than the aims. They specify something toward which an effort is directed, e.g. goals or ends of an action.

The purpose is already specified at the strategic layer. It depends on the *audience* intended for the WIS, and is influenced by the *mission* that is determined by the WIS provider.

Intent facet: The *intent* suggests a clearer formulation or greater deliberateness. It distinguishes between the following two aspects: The *targets* of stakeholders specify the steps of a plan and its termination or satisfaction conditions. The *object* of stakeholders is related to wishes or needs of users, and specifies an effort or activity of a user in order to satisfy the need.

The intent facet is related to *tasks* the user wants to accomplish. The intent may be ordered into *major intents* specifying the main uses of the system and *minor intents* that may be only partially of interest. Intents may be generalized to *themes* that represent classes of intents.

Time facet: The *time* facet is used for specification of the general time restrictions such as the *design*, which implies a more carefully calculated plan,

and the *end*, which stresses the intended effect of an action, often in distinction or contrast to the action or means as such. Time facets may be very general. In this case, we use *occasions* to represent an entire class of time frames.

Representation facet: Intentions are described or annotated through utterances, words or pictures. The word representation is related to word fields used in the strategic model. The icon representation is based on metaphors. Word fields may be specialised to concept fields discussed later in this chapter. Representation is deeply dependent on the cultural environment and the community that uses the WIS. Intentions can be supported by providing stimuli that rouse or incite activity.

Intentions may be restricted by a *scope*. The scope allows to concentrate on the main aspects. Typical restrictions are the cultural environment, education or other profile properties of potential users, or specific time facets for utilisation of the WIS.

The first two facets of intention have a general form (objective, object) and a more concrete form (aim, target). The purpose facet depends on the mission and the audience, whereas the intent facet depends on the tasks. Therefore, the first facet specifies the 'what', the second the 'how', and the third facet the 'when' of an intention. We may either concentrate on a more general specification or a more concrete one.

The different facets may be considered separately or altogether in a condensed form. The detailed consideration is necessary, whenever a fastidious audience requires sophisticated content. High demands come into consideration due to the content provided, the community that must be satisfied, the sophisticated functionality required for the WIS, or the high attention the WIS gains.

Example 3.1 *The purpose facet is often considered to cover 'soft intentions' or by 'business rules'. In an information service a user may be interested or becoming more interested in some content. The intent facet is different and mainly driven by tasks the user tries to solve.*

Similarly, we may derive a number of intentions a user has in mind whenever s/he visits an edutainment site:

Aim: *An aim of an edutainment user may be to obtain a certificate that proofs the success of learning. Aims of providers of edutainment sites might also be binding the learner to some products such as the software of a supporter of the website. Typical general aims within an edutainment site are to grant equal rights to everybody; nobody should have higher rights than other learners.*

Objectives: *Typical objective of using an edutainment site are greater ease of learning, greater pleasure or satisfaction during leraning, and more fun. Another general objective is security and privacy. Edutainment applications also host confidential information, e.g. information on progress and errors. The learners need to be sure of the security and privacy of their data. They should be informed of the security precautions.*

The audience *of an edutainment site may be rather unspecific or more specific such as students of a college or analysts of a bank.*

The mission *of an edutainment site is to support learning by providing easy-to-grasp knowledge. At the same time, the provider may be interested in increased visitor-to-customer conversion rate, increased number of returning customers, and increased revenue.*

The intent is related to the more concrete tasks a user has in mind while visiting a WIS.

Example 3.2 *Intents come directly from analysing the needs and the demands of users. Some possible* tasks *a user may want to accomplish in an edutainment site are the following:*

- *A user realises that certain knowledge or information is necessary for solving a task. For instance, an analyst of a bank wants to know what the changes are in the customer community that led to a rise of faulty credits.*
- *A student in a college needs to know methods for analysing data. The student is interested in a data mining course that provides material on his/her educational level, permits learning the content interactively, and is comparable to the material the student is currently working with in the college.*

Possible intents in an edutainment site are the following:

Objects: *There are a number of objects such as faster task completion, successful completion of more tasks, or commission of fewer errors.*

Targets: *In the bank analyst case, the analyst needs to know data mining methods, to capture the achievements and the disadvantages and pitfalls of such methods.*

The themes *in the bank analyst case could be the interest in learning association methods, preprocessing of data, and prediction methods.*

The time facet represents general time restrictions such as the time or the interval for visits of the WIS, the time and the interval of repeated visits, or the temporality that may force a user to leave the site. The

representation facet represents the flavour or atmosphere of a site, which is largely determined by the other three facets.

The set of intentions we should consider may be rather large. Moreover, the faceted representation may become too difficult to manage and to satisfy. The order we may use depends on the audience and on the provider. The audience is oriented towards solving tasks. The provider has a 'mission'. We can use these two restrictions to figure out which kind of website supports this profile, to harmonize our understanding with the corporate identity, to order the target realisation (short-term, long-term), and finally to develop an understanding how the site will look like in two years from now on.

Example 3.3 *The intention* prepare for a visit *is based on a number of intents such as addressing visitors beforehand for some purpose, use, or activity. It includes the aim to put the visitor of the WIS in a proper state of mind. A task associated with preparation may be to work out the details of a plan in advance. The task may be extended by putting pieces of information together into a compound form or preparing a report. The time facet may range from 'now' until 'next opportunity'. Typical metaphors supporting this intention are baskets, descriptions, and cultural journals.*

The intention prepare for a cinema visit *extends the intention of* prepare for a visit *by a certain content, a refinement of the time facet to the next possible visit, and a refinement of the purpose facet now targeting at entertainment.*

We can combine the description of intentions in a semi-formal way as follows. The items in the list are optional except the first one.

`Intention space:`	⟨intention name⟩
`Purpose:`	⟨outcome description⟩
`Aims:`	⟨list of aims⟩
`Objectives:`	⟨list of objectives⟩
`Intents:`	⟨outcome description⟩
`Targets:`	⟨list of weighted targets⟩
`Objects:`	⟨list of weighted objects⟩
`Themes:`	⟨class of intents⟩
`Time:`	⟨outcome description⟩
`Design:`	⟨general flow⟩
`End:`	⟨effects, termination conditions⟩
`Occasion:`	⟨list of objectives⟩
`Presentation:`	⟨general style guide⟩
`Atmosphere:`	⟨general description of atmosphere⟩
`Metaphors:`	⟨list of metaphors⟩
`Based On:`	⟨*tasks, audience, mission, goal*⟩

4 LIFE CASES

Life cases allow to overcome the information overload and lost in the hyperspace syndromes typically observed for WISs. For completion of tasks users need the right kind of data at the right moment of time, in the right dose, of the right form, in the complete extent, and within the restrictions agreed upon in advance. Moreover, users are limited in their abilities for verbalisation and digestion of data, and by their habits, practices, and cultural environment.

These limitations may cause intellectual overburdening of users. Most systems that require sophisticated learning courses for their exploration and utilization did not consider these limitations and did not cope with real life situations. The approach we use for avoiding overload is based on observation of real applications before developing the system.

We may extract life cases from observations in reality, which are very useful source, whenever a WIS is going to be developed from scratch or is required to provide a 'natural' behaviour. In the latter case users are not required to learn the behaviour of the WIS. Instead, the user can continue using a 'classical' behavioural pattern.

Example 4.1 *As a motivating example let us consider the life case* relocation *of a person, which consists of*

- *the change of basic relocation data including the possible removal of data on the old location,*
- *the change of official documents such as the passport,*
- *the optional change of relation enhancements such as the registration of pets, relocation of cars,*
- *the change of personal specific data such as family enhancements, or relationships to religious bodies,*
- *the change of data for additional relocation announcements such as tax, insurance changes, and*
- *specific additional tasks such as applications for housing allowances.*

The person acts in the role of an issuer. *We observe that* relocation *is enhanced by the profile of the issuer, by the specific tasks related to the* relocation *of the issuer, by specific laws and regulations, and by advanced functionality required for associating the life case with other life cases.*

The life case relocation *consists of steps such as* change of address data, change of data for associated people, change of registration data *for cars, pets, etc.*, change of specific data, *e.g. data for public authority responsible for aliens,* change of data for social

aid, *etc. These steps are bundled together due to their relationship to one person and to one life case. The associations may be represented by adhesion of different steps, e.g. representing the association of steps by a hypergraph.*

4.1 The Concept of Life Case

Life cases are characterized by:

Observations: We are interested in the collection and assessment of behaviour relating to the specific application. This would typically involve an observation of behaviour of users in real environments, including a background check that relates usage to intentions, goals or tasks.

Processes: This involves arranging all the actions observed in an application into a main logical and coherent pattern. In some case, deviations of the main pattern must be modelled through exceptions. In most cases, we can use parallel execution of independent actions.

Assessment: This involves the reconstruction of the sequence of actions and specific behaviour of users. This will aid in understanding the role each individual has within the story. It assists in developing the user profile.

Individual profiles: A list of background including physical, and behavioural characteristics of individuals is conducted. This list can also be used for deriving the most appropriate interview technique we discuss below.

Interpersonal coherence: A variation in the activity will relate to variations of other life cases.

Significance of time and place: The choices made also depend on mobility, surrounding, and schedules of events of interest.

Characteristics of the life case: Individuals using a service may be grouped by characteristics. Based on this grouping a similar behaviour is observed.

Experience and skills: Individuals may have their own experience with services provided in real life and thus use different behavioural pattern of service employment.

In general, life case studies are designed to produce a picture of service employment that is as accurate as possible. Determining *why, how, where, when* and *why* a service is called using *what* data provides a better picture for utilisation scenario. As life cases are used for quality control, we must carefully record our observations. We either use a natural language specification or a semi-formal one as described later.

Example 4.2 *Let us consider the support of hotel search within an information service. In this case we may observe the behaviour of individuals in travel agencies while seeking for hotels. We observe that in most cases search based on associations is preferred over search by main properties such as name, address or facilities. Associations may be of a large variety, e.g. convenience to reach a hotel, location in certain maps, places of interest, or events that have caused the search. Other search criteria may be bargain or bundled offers. At the same time we observe that hotel search is combined with other intentions of users such as visiting cultural institutions. It may depend on results for search of other individuals.*

Another example of a life case study is the booking of train tickets depending on individuals, offers of railway companies, circumstances of individuals are using trains, etc.

4.2 Life Case Development

Life cases may be developed and interpreted step by step:

1. The first step during life case collection is the survey of possible application cases we might consider. The observations could have more than one meaning and may follow a variety of user-related goals. In this case we consider the most likely meaning.
2. The second step involves a deep assessment of the life cases. We extract the different execution orders, the relationship among actions, and the roles individuals play these actions.
3. The third step extracts those life case characteristics that are distinguishing features and are relevant for time and location. At the same time we search for similarities within these life cases.
4. The final step is concerned with the characterization of potential users, their behavioural patterns, and their roles in various stages of the life case.

Collectively, this information will produce a picture of the life case we are intending to support by a WIS. This may produce further life cases, or may aid in reducing the amount of development. It may result in a prioritisation of life cases under consideration, assist in the linkage of potentially related life cases, assist in assessing the potential of the WIS development, provide the WIS developers with relevant leads and strategies, and keep the WIS development on track and undistracted. These life case are mapped to scenarios in the sequel. The integration of scenarios can also be based on life cases.

Example 4.3 *Let us consider life cases of an information service for a city or a region:*

- *Attracting visitors: The information we are providing is based on some information need visitors may have. The life case follows those that we observe during marketing activities.*
- *Inhabitants information: The life case is based on selected information chunks that may be of interest to individuals, an ordering of the corresponding available information, and a derivable personal newspaper.*
- *Informing tourists: The life case is similar to those observed in city information centres.*
- *Providing official city information to inhabitants: This life case follows the message board metaphor that is used for newspaper information.*

During the development of city information services such as the service `www.cottbus.de` *we have made a life case analysis for city information services and detected around 30 different life cases. We can categorize these life cases by the content and information demand of individuals. We distinguish:*

- *life cases related to tourist content for inhabitants permanently living in the city, for inhabitants temporarily living in the city, for short-term tourists and business people on short term trip, for vacationists planning a trip for more than 5 days, for teenagers with uncertain wishes, for seniors with particular interests, for week-end visitors, for visitors of highlights, festivals, etc.;*
- *life cases related to official content for inhabitants on search through directory of public authority, for inhabitants on search for emergency cases, for inhabitants orienting themselves before applying at public offices, for inhabitants interested in problems of city management, etc.;*
- *life cases related to business content, e.g. for investors considering engagement in the area.*

For the collection and development of life cases it is normally a good idea to interview the personnel currently providing the service.

Life case should be checked against sufficy. As they may have a variety of traces, we add two stages to life case analysis:

Exploration: The actual life case is provided to WIS customers and incorporated into their processes. Life case exploration can be conducted in normal environments of the user such as home or work. Then users can show the actions while they are explaining their aims and targets.

Apprehension: Finally, cross checking of life cases and utilisation scenario is used for correction, extension and affirmation of the developed utilisation scenarios.

During life case elicitation and specification users are confronted with sketches of scenarios. We ask what users think about these conceptualisations. The feedback is used for the refinement of life cases. We may use affinity diagramming, in which case we arrange the individual conceptions we discover on a blackboard, associate them, and discuss their relationship to the general characteristics of WISs. Another technique is based on card sorting similar to the model-view-control cards used in software engineering. In this case cards represent simple life situations. Their associations are then used for organising the conceptions. These sketches can also be used for discussing deviations from the normal case and for search of exceptional life cases. Instead of directing users to specific life cases we can show them two or more alternatives for the problem solution.

Life case analysis goes beyond task analysis. It is simple and easy to carry out, lightweight and straightforward to understand, and yet quite accurate in identifying the real demands users of a WIS might have. While observing real life cases and mapping them to life cases that must be supported by the WIS we carry out a user-centered development. Additionally, we may identify and resolve conflicting or competing intentions. These conflicts can be resolved on the basis of life cases by accommodating both intentions, i.e. fulfilling the stakeholder intentions and supporting the demand of users.

Besides observation of real life situations life case detection can also been based on *interview techniques*. Research on artificial intelligence has also resulted in a number of interview techniques:

- Unstructured interviews can be considered as an informal and explorative conversation on goals a user is following and on tasks a user has to complete. Unstructured interviews observe the rules that are applied to brainstorming. All interruptions should be avoided. The questions asked must be short and simple to answer and should be open-ended questions. Interview partners should share their thoughts and experiences. There is no judgement, confrontation or condescension. Users should not be lead toward a certain scenario. Unstructured interviews should only be interrupted if clarification is required. They need consecutive feedback in order to give the interviewees the impression that the interviewer is listening. While the interviewees are talking, everything that seems to be important is written down and used for later questions.
- Structured interviews are based on query plans.

These plans can be based on the general characteristics of WISs. We start with easy questions on data and main functions and continue with the life cases of their utilization. Context and intentions are more difficult to capture. Structured interviews are also based on features, which can be shown to users for a rating of their importance.

- Life cases can also be detected by observing and critically analysing products of competitors. Elicitation of life cases from existing solutions is based on specifications, results from interviews with business users, excerpts from documents and spreadsheets, analyzed messaging and transactions, knowledge on the solutions that are currently used, meta-data and context information on the utilization within the current framework, and third party information. As reasons for restrictions cannot be captured, the copying of already existing solutions can only be used in exceptional situations.
- Users may also use protocols of "loud thinking", in which case they are provided with a real-life problem of a kind that they deal with during their working life, and asked to solve it. They imagine that they are solving the problem presented to them. As they do so, they are required to describe each step and the reasons for doing what they do. The transcript of their verbal account is the protocol. In this case, they work and explain their current work. These protocols can be the basis for capturing a scenario. This interview technique should be combined with other interview techniques.
- There are other interview techniques that might be useful for life case elicitation such as the laddering or grid techniques. In these cases, a hierarchical structure of the application domain is formed by asking questions designed to move up, down, and across the hierarchy.

4.3 Semi-Formal Representation of Life Cases

Although it is sufficient for life cases to be stated in natural language, we may use a semi-formal representation instead using the following template:

`Life case`:	⟨life case name⟩
`Characterisation`:	⟨outcome description⟩
`Tasks`:	⟨list of user tasks⟩
`Problems`:	⟨list of problems⟩
`Background`:	⟨general characterisation⟩
`Objectives`:	⟨list of objectives⟩
`Life case flow`:	⟨general graphical description⟩
`Milestones`:	⟨graph of milestones⟩
`Content consumed`:	⟨consumed content items⟩
`Content produced`:	⟨produced content items⟩
`Themes`:	⟨class of intents⟩
`Figures`:	⟨actors list⟩
`Expected profile`:	⟨general profile description⟩
`Collaboration`:	⟨general collaboration description⟩
`Context`:	⟨general context description⟩
`Time`:	⟨temporality limitations⟩
`Place`:	⟨assignment of places⟩
`Legacy`:	⟨names of documents⟩
`WIS`:	⟨general WIS context⟩
`Representation`:	⟨general behavior⟩
`Approaches`:	⟨general description of approaches⟩

This template captures the following components of life cases:

Characterisation: Life cases are characterised on the basis of strategic issues, the problem statement, background and objectives for the life case, the methodology that is used for solving the life case, and by describing the basic modules that are used for solving the life case. The characterisation is harmonized with intentions, tasks and goals.

Life case flow: The life case flow combines the observations we made, the processes involved, and the data which are consumed or produced. The life case flow is mapped to a scenario.

Figures: We develop profiles, especially those of individuals, as part of the WIS utilization portfolio, and interpersonal coherence specifications.

Context: Time and location is explicitly described for life cases. The applicability of life cases is restricted by regulations, laws, and orders. These restrictions are seldom given in an explicit form. In addition, the context of life cases is given by the provider, the intended audience, the utilization history, and the availability of data due to the technical environment. We also use this information for the context specification.

Requirements: Life cases are restricted by habits, general approaches, good practices, and boundary conditions for their application. They presuppose experiences and skills of the users involved.

Note that life case we intent to support by a WIS can be completely different in real life. Sometimes we need a complete reorganisation of the business activities. In this case we should not map the real life case to a suite of associated scenarios, but rather envision a better organisation of the tasks and goals and then map these to a new envisioned hypothetical life case.

Let us now outline briefly how life cases can be used in a pragmatic way to specify a story space. More precisely, we just want to derive a prototypical scenario. Other scenarios may result from other life cases, and the scenarios collected this way can be integrated and then be subject to further refinement. Let us concentrate on the life case *relocation* again.

Example 4.4 *In the case of the relocation of a person the steps identified in Example 4.1 give immediately rise to scenes in a scenario, i.e. in addition to an entry scene, say* start, *we obtain scenes for* change of address data, change of data for associated persons, change of registration data, change of specific data, change of data for social aid. *For a start we only have a single actor* citizen.

In principle, the visit of any of these scenes is optional, which gives rise to a classification of actors into citizens with children, citizens with pets, foreign residents, *etc. The life case of a citizen with children gives rise to a scenario for the change of data for associated persons, while the life case of a citizen with pets gives rise to a scenario for the change of specific data, etc.*

5 CONCLUSION

In this paper we introduced the concept of life cases as a contribution to supporting pragmatics of storyboarding, which closes a gap in our development methodology. The general idea is to start from real life observations and to characterize them in a semi-formal way. This gives rise to prototypical scenarios, tasks, roles and user profiles by abstraction. Several of such prototypes can then be integrated and refined to obtain the desired storyboard.

The concept of life cases is new and original. It has already successfully been applied in our web information system projects, e.g. for information services (e.g., `www.cottbus.de`), for edutainment systems (e.g., `DaMiT`), and for community services (e.g., `SeSAM`). Use cases in UML (Conallen, 2003) and business use cases (Robertson and Robertson, 2006) share some similar intentions, but are far too simplistic to capture the same information as the novel life cases. However, life cases still contribute only a part of storyboard pragmatics. They have to be complemented by user models, which should give rise to a deeper understanding of actors, and contexts. Both topics will be addressed next and published in due time.

REFERENCES

Ambler, S. W. (2002). *Agile Modeling: Effective Practices for eXtreme Programming and the Unified Process*. John Wiley & Sons.

Carroll, J. M., editor (1991). *Designing Interaction: Psychology at the Human-Computer Interface*. Cambridge University Press, Cambridge, England.

Ceri, S., Fraternali, P., Bongio, A., Brambilla, M., Comai, S., and Matera, M. (2003). *Designing Data-Intensive Web Applications*. Morgan Kaufmann, San Francisco.

Conallen, J. (2003). *Building Web Applications with UML*. Addison-Wesley, Boston.

Courage, C. and Baxter, K. (2005). *Understanding your users: a practical guide to user requriements - methods, tools & techniques*. Morgan Kaufman, Boston.

De Troyer, O. and Leune, C. (1998). WSDM: A user-centered design method for web sites. In *Computer Networks and ISDN Systems – Proceedings of the 7th International WWW Conference*, pages 85–94. Elsevier.

Giorgini, P., Mylopoulos, J., Nicchiarelli, E., and Sebastiani, R. (2002). Reasoning with goal models. In *ER*, pages 167–181.

Harel, D. and Marelly, R. (2003). *Come, Let's play: Scenario-based programming using LSCs and the play-engine*. Springer, Berlin.

Houben, G.-J., Barna, P., Frasincar, F., and Vdovjak, R. (2003). HERA: Development of semantic web information systems. In *Third International Conference on Web Engineering – ICWE 2003*, volume 2722 of *LNCS*, pages 529–538. Springer-Verlag.

Lowe, D., Henderson-Sellers, B., and Gu, A. (2002). Web extensions to UML: Using the MVC triad. In Spaccapietra, S., March, S. T., and Kambayashi, Y., editors, *Conceptual Modeling – ER 2002*, volume 2503 of *LNCS*, pages 105–119. Springer-Verlag.

Mylopoulos, J., Fuxman, A., and Giorgini, P. (2000). From entities and relationships to social actors and dependencies. In *Conceptual Modeling - ER 2000*, pages 27–36, Berlin. Springer-Verlag.

Robertson, J. and Robertson, S. (1999). *Mastering the Requirements Process*. Addison-Wesley.

Robertson, J. and Robertson, S. (2006). *Requirements-Led Project Process*. Addison-Wesley.

Rosenfeld, L. and Morville, P. (1998). *Information Architecture*. O'Reilly, Cambridge.

Schwabe, D. and Rossi, G. (1998). An object oriented approach to web-based application design. *TAPOS*, 4(4):207–225.

Sommerville, I. (2004). *Software Enginering*. Addison Wesley, San Francisco, seventh edition.

Thalheim, B. and Düsterhöft, A. (2000). The use of metaphorical structures for internet sites. *Data & Knowledge Engineering*, 35:161–180.

Vale, E. (1982). *The technique of screen and television writing*. Simon and Schuster, New York.