Roskilde University

Effect specifications as an Alternative to Use Cases

Barlach, Anders; Simonsen, Jesper

Published in:
Proceedings of the 31st Information Systems Research Seminar in Scandinavia, IRIS 31, Public systems in the future – possibilities, challenges and pitfalls, Åre, Sweden, August 10-13, 2008

Publication date:
2008

Document Version
Publisher's PDF, also known as Version of record

Citation for published version (APA):
Barlach, A., & Simonsen, J. (2008). Effect specifications as an Alternative to Use Cases. In A. Asproth, K. Axelsson, S. C. Holmberg, C. Ihlström, & B. Sundgren (Eds.), *Proceedings of the 31st Information Systems Research Seminar in Scandinavia, IRIS 31, Public systems in the future – possibilities, challenges and pitfalls, Åre, Sweden, August 10-13, 2008* Mid Sweden University. http://www.iris31.se/papers/IRIS31-063.pdf

Asproth, A, K. Axelsson, S.C. Holmberg, C. Ihlström, and B. Sundgren (Eds.): *Proceedings of the 31st Information Systems Research Seminar in Scandinavia, IRIS 31, Public systems in the future – possibilities, challenges and pitfalls, Åre, Sweden, August 10-13,* 2008.

Effect specifications as an Alternative to Use Cases

Anders Barlach; Roskilde University; Denmark; Department of Communication, Business and Information Technologies (CBIT); Computer Science/User driven IT-Innovation; barlach@ruc.dk
Jesper Simonsen; Roskilde University; Denmark; Department of Communication, Business and Information Technologies (CBIT); Computer Science/User driven IT-Innovation; simonsen@ruc.dk

Abstract. This article reports from an experiment where use-cases were replaced with effect specifications during the cooperative configuration and implementation of a health care IT system supporting new Health Centres in Copenhagen. While use-cases describe the users interaction with the it-system, effect specifications focus on the user's intended effect from using the IT system – without necessarily specifying this system in any detail. We compare and discuss the differences that we observed when using effect specifications to support communication in a distributed configuration process constituted by three different roles: The user, the configurator, and the developer. We conclude that effect specifications might be instrumental in supporting the articulation of cooperative work between these three roles and that they further support a shift in the translation of overall user needs to system functionality from the user towards the vendor.

Introduction

Today, many system development projects are based on generic systems and flexible technology platforms that can be configured to support specific organizations [1, 2]. Such development environments – or frameworks – form a basis for the development and implementation of IT systems resulting in a configuration and programming exercise, rather than entirely programming. This has resulted in the task of development and implementation being dispersed between three types of roles, users, configurators and developers, which we refer to as a configuration context, see figure 1.

Figure 1 Configuration context with three different roles

The *user* is a representative from the customer and has firsthand experience of the problems and tasks to be considered or supported in the work domain. The *configurator* typically has a background in the application domain of the framework (e.g. a nurse, a doctor or engineer specialized in healthcare). They might not be skilled in advanced programming, but rely on their knowledge from the business domain (in this case, the healthcare sector) and communications skills, combined with specific training in building applications using the framework and its tools for configuration. They use the framework as building blocks creating prototypes and applications from combining relevant features and functionalities offered by the framework to a specific customer. The *developer* has a technical background (computer scientist or engineer) and is responsible for developing and maintaining the generic framework which is used to develop IT systems for many customers spanning various different domains.

The traditional user-developer relation is based on formal documentation required to carry the users context [3] through to the implementation in order ensure the success of projects and outcomes in terms of alignment with the requirements representing the needs of the users. This is also the case for the dispersion of the traditional user–developer into the user–configurator–developer as depicted in figure 1. In addition, the division of labour in a configuration context entails that a large portion of the final product is the result of the user and configurator relation without the developer being directly involved with the user. The configurator and the developer rely on each other for requirements describing and implementing new building block features and functionality.

When designing for quality in software development the user involvement in the specification process becomes central but does not guarantee success by itself. As Gallivan et al. [4] has noted, the nature and quality of the communication between users, and/or configurators and developers, remains an understudied aspect of user participation and there is still much that we do not know about how and why user participation sometimes, but not always, delivers positive benefits. During the negotiation of requirements, detailed knowledge is communicated between the parties in order to gain a shared perception of the work problems and the impact of the intended technical solution. The ability to effectively communicate the requirements is not a one-way transfer between two very different environments; healthcare being non-technical and computer science being non-healthcare. Both are characterized by a high level of skill but with their own terminology and no formal training in utilizing the other party’s representation tools or thinking (e.g. object orientations, Unified Modelling Language, etc.). This may result in a lapse between the need and the solution when utilizing a representation tool native to one party [4].

Use-cases are widely used as the "tool of choice" when it comes to describing and communicating the users requirements in terms of system functionality. Traditionally, use-cases have been used as a means for communicating requirements from the users to the developers of the IT-applications.

In this paper we report from an action research project where we experimented with replacing the traditional use-cases with effect specifications in a configuration context as depicted in figure 1. The experiment was part of a larger research initiative on 'effects-driven IT development' [5]. Our *hypothesis* is that effect specifications may form an alternative to use-cases by providing a better support for the user-configurator communication (by offering a terminology more familiar to the user) while also being instrumental as a requirements specification supporting the configurator-developer communication. To investigate this hypothesis we have conducted an action research experiment, in terms of an IT-project, where we replaced use-cases with effect specifications. The IT-project was conducted by CSC Scandihealth (the IT vendor) and developed a fully functional healthcare IT system supporting new Health Centres in Copenhagen (the customer). The *purpose* of the experiment was to compare effect specifications to use-cases and to explore how effect specifications might change the communication and coordination between the user and the configurator (on the one hand) and between the configurator and the developer (on the other hand). Especially, we wanted to explore how effect specifications – as an alternative to use-cases – can support the configurator and developer with sufficient documentation of requirements for programming and implementing the IT-system. By comparing use-cases usually applied by the vendor in similar projects with effect specifications we intend to illustrate strength and weakness identified during the experiment by addressing the research question:

> *What differences can be identified when use-cases are replaced with effect specifications to support communication and coordination in a configuration context*?

First, we present the new Health Centres in Copenhagen where the project took place. Secondly, the research method and the experiment where we applied effect specifications is introduced. Thirdly, the two different approaches – use-case and effect specifications – are presented. Fourth, we compare the effect and use-case specifications involved in the experiment. We refer to a complete of a effect specification and corresponding use-case attached as appendix to this paper. Finally, we discuss the differences and conclude our results.

Health Centres in Copenhagen

The project took place in the municipality of Copenhagen with the task of delivering clinical pathways in a portal framework within 3 different Health Centres. Health Centres is a new addition to the Danish healthcare system as a consequence of a major national reform in 2006 that reduced the number of municipalities from 275 to 98. The responsibility for local preventive and rehabilitating care was delegated to the reorganized 99 municipalities. The overall political purpose of the new Health Centres is to support preventive care and rehabilitation for all citizens in Denmark. On the operational level this relative new organization is still underway, the Health Centres do yet not know how they will be evaluated and their work procedures and practices are still being formed at each centre.

The case focuses on the “clinical consultation” which is a core activity of delivering care to the patient through a focused dialog. The purpose of the consultation, as defined by the health centers, is to comprehend the complexity of the individual patient situation and support what is expressed by the definition of life skills:

> *“Life skills - ”The personal, social, cognitive and physical skills that enables humans to take control and give their lives direction, and to develop the capacity to live and produce change in their surroundings” [10]*

The Clinical Consultation is typically performed in one of the interview rooms at the centre. At the Cancer centre it has character of a living room where the healthcare provider and patient can sit and talk during the consultation. The healthcare provider usually has a paper with the guide for the consultation, and the guide has a general and specific part. The general part is the same regardless of which problem the patient is referred with, and the specific has been developed with a specific problem in focus.
The common characteristics of conducting the consultation task are the cooperative dialog between healthcare provider and patients focusing on their individual health problems as suggested by the clinical work method adopted in the centre. The most visible variation in the performance of this task is in content and suggested interventions when the tools supporting the specific problem has been tailored or the cultural background of the patient requires an interpreter to attend. These problems vary in origin and are identified by their medical grouping; breast cancer, colon- rectal-cancer, pulmonary cancer, diabetes, Chronic Obstructive respiratory disorder, Cardiac disorder, Metabolic Syndrome and smoke stop counseling.

Research method and experiment

When doing action research one strives to to create knowledge while one at the same time propose and implement change, and improve practice and performance. Action research “involves the formulation of a theory, intervention and action-taking in order to introduce change into the study subject, and analysis of the ensuing change behavior of the study subject” [6], . The study’s subjects or participants in the experiment were introduced to change by one of the authors, who participated in the role of configurator as part of the Industrial PhD-program in Denmark.

The development project hosting the experiment took place in 3 different and newly established Health Centres located in the municipality of Copenhagen. The experiment was initiated in August 2007 (see figure 6). Specification and design was completed in February 2008. The activities involving users, configurators and developer have accumulated 900 hours of participation in workshops, meetings, configuration etc. The vendor implemented a standard configuration delivery, with the exception of using effect specifications instead of use-cases. The project was committed to develop and evaluate the clinical pathways while experimenting with the effect specifications. The final system was delivered in May 2008.

The overall project process was organized with an agile approach relying on the Dynamic Systems Development Method framework [9]. The development processes were designed to take the participants through a series of workshops (see figure 2). A typical workshop had participation of the customer's project manager and selected domain users, along with the vendor's project manager and a configurator. During the workshop the effect specification and prototype is debated and negotiated in order to reach a level of knowledge that allows the configurator to return with the next version of the prototype and the customers to refine their requirements. An initial workshop specified the customers overall effect requirements without having any prototype to support the discussion. Then 1-2 laboratory workshops were conducted supported by early prototypes evaluated in a 'conference environment' where all participants sat around the same table. This was followed by 1-2 "in-situ" workshops were more mature prototypes allowed the users to actually use and evaluate the system closer to their own environment and everyday tasks. Finally, the system and the clinical pathways were completed and put into production.

Each of the three centre selected an initial pathway to implement and a process (as outlined in figure 6). The system was specified by the configurator after interacting with the users at the workshops. For example the designers would ask (at the workshop) what information the users would require to perform the process and feel prepared for a consultation. This would translate into a hypothesis about what dependencies there are regarding the preparation process in terms of the effect of usage the user anticipates. In the experiment the users were physiotherapists, nurses and similar therapists or care providers.

During the workshops involved in the experiment, we encouraged the users to express their needs regarding the work domain in terms of effects.
When we discussed the generalized processes with the users they found it problematic that they should come up with a scenario involving the system as mentioned above. They argued that they did not have the technology experience to discuss the network of imagined functional interactions.

We facilitated the discussions in the workshops so they did not focus on the envisioned IT system and its functionality, but instead focused on a formative specification of the effects from using the IT system.

The process outlined in figure 6 resembles the normal approach for this type of project in the vendor organization. In order to investigate the research question the usual use-case approach was substituted with the effects method. Choosing the consultation as the clinical task to be singled out was done on ground of its common characteristics between the centres involved. The use-case used for comparison has been selected among the vendors documentation for its similarity with the clinical task found in the effects-driven documentation.

The development environment consist of a portal framework, including domain specific functionality regarding organization, user and information, access logic, security etc. The framework currently comprise two modules (a booking module and a clinical pathway module) based on Microsoft's .NET 2.0 platform. These modules operate on a standard database and offers a variety of configuration services to be tailored by the configurator for each customer in by using xml to define layout of forms and views, business rules and functionalities supporting the interaction and utilization of the system.

Use-case specifications

Use-cases define the behaviour of an IT-system towards a user performing a given task. Use-cases has been designed for all the right reasons; to support user-centred techniques, ensure accuracy and quality by capturing requirements from the user's point of view. Therefore they are written in a natural language and utilize an easy to understand notation.
Use-cases are developed as an external view on the IT-system and the primary purpose is to document the users' requirements in terms related to the functions of the IT-system. A function is described as a set of inputs, the behaviour, and outputs of the system [12]. A use-case cannot express concurrency since use-case transactions are inseparable and serial [13]. Use-cases might also lack the ability to convey knowledge of the interactions between use-cases as they cannot communicate between each other [13].

A typical use-case describes the users' requirements in terms of system functionality while the users' interaction with the IT-system for a specific and well delimited task. Higher level use-cases might deal with the business processes interacting with tasks and related system functionality: Such high-level use-cases are often referred to as business process models (BPM). We refer to appendix 1 for an example of a business process model (BPM) and appendix 2 for an example of a use-case.

There are many different templates for use-case [12] but they share most of the characteristics in the Template example Figure 1.

Use-cases map into lower-level User-Interface (UI) descriptions and object models. The use-case description of an IT-system becomes a model describing "what" in contrast to an object-model describing "how" [13]. The primary function of the use-cases then is to describe, to an audience external to the user organisation [13] what are to be developed and implemented, in a more abstract fashion or at the level of "black-box". This means that the object model can be entirely different from the use-case model as they are dealing with different audiences and levels of abstractions (see figure 3).

- Use case name
- Goal
- Summary or Purpose
- Actors
- Preconditions
- Trigger events
- Basic course of events
- Alternative paths
- Postconditions
- Business rules or algorithms'
- Notes
- Author and date

Figure 1: Template example [7]

Level	Level of use-case specifications	Object of analysis – process of the IT system
1	Business Process Model	Focus for the system, motivation for development
2	Business Process Model	Context of the usage of the system
3	Use-case model	Tasks involving the IT-system, transactions, functions
4	User-Interface model	Display/access to information or clusters of information
5	Object-model	data, tables and their relations

Figure 2: Object of analysis – IT-system use-case at descending levels of abstraction.

Throughout the use-case driven design approach the user controls the requirements by defining what the user want to do with the IT-system [13] as opposed to what do the user want to do to the domain and then ask if and how a system can assist. This is illustrated in Figure 2, the IT-system is in focus and the user context presumes an IT-system is involved.

Effect specifications

Effect specifications are descriptions of the effects that the customer and the users would like to obtain by using the IT system.

We have developed a template for effect specifications that include the concepts shown in figure 4. Effects can be specified in terms of interventions, agents, goals, results, and evaluation. These concepts share a relationship within each effect: Results are outcomes of interventions performed by agents (users). Goals are evaluated by means of measurements of the result. The relations indicate that an effect is the anticipated outcome generated by the user when performing a given intervention.

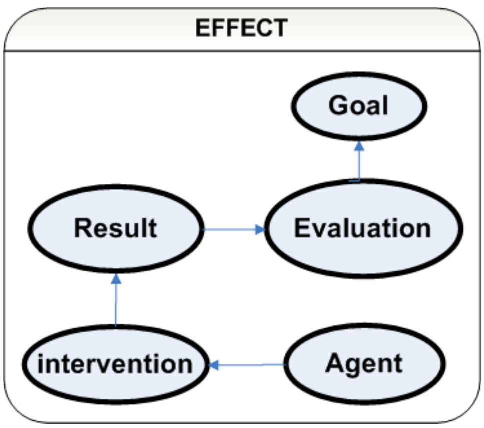


Figure 3: Effect concepts and internal relations

Effects are related to each other, as they can serve as means to achieve other more abstract effect. A hierarchy can be identified in the relationships between effects; they are goals or means depending on their function explaining *how* effects contribute or *why* they contribute. For this purpose we have adopted a 5 level means-ends hierarchy, inspired by cognitive work analysis [7, 8]. Below we describe this hierarchy along with an example of an effect specification from the experiment. We refer to appendix 2 for a complete documentation of the selected effect specification.

The properties represented in the effects means-end hierarchy are purpose and reasons at the top (high level of abstraction) and evolve into functions to be applied as means to these ends (see figure 5). From the bottom of the hierarchy physical properties evolve as outcome or ends at the upper levels. At the less abstract levels (level 4 and 5 in figure 5) IT has a higher degree of influence. A multitude of different organizational properties gain more impact when the level of abstraction increases (to e.g. level 1 and 2 in figure 5). Below we further describe the 5 levels and provide an example (taken from appendix 2) for each level.

1. *Purpose*: This is the highest abstraction level and represents the goals and purposes in relation to the organizational environment and the purpose of the organization pursued through the lower levels. It is typically identified as policies, service goals etc. regarding quality and efficiency at the enterprise level of the customer organization. Example: The health policy of the Copenhagen municipality [14] was integrated into the health care professionals' philosophy for best practise in consultations, and became the delivery of care to prevent and rehabilitate by supporting life skills [10] in an empowerment perspective (appendix 2).

2. *Abstract function*: This level addresses prioritizing and allocating resources to the various generalized processes and activities on levels below. Resources are distributed depending on the overall measured value. Values can be related to money, material, energy or people. Value may be derived from efficiency or the quality of service. Values are applied throughout the hierarchy and can be divided into 3 categories (adopted from the ISO 9241standard on Usability):
    - *Effectiveness*: Accuracy and completeness with which users achieve specified goals [15]. Example: The professionals' ability to comprehend the complexity of the individual patient pathways to improve transfer and reuse of knowledge.
    - *Efficiency*: Resources expended in relation to the accuracy and completeness with which users achieve goals [15]. Example: by adapting the documentation to the work process before and after the consultation the time and effort spend doing these tasks would decrease.
    - *Satisfaction*: Freedom from discomfort, and positive attitudes towards the use of the product [15]. Example: The ability to monitor the patients "wellbeing" in progressing the consultations to follow-up on their own perspective of quality of life.

The categories defined might dictate or influence the measurement methods applied to evaluate the effects. E.g. Effectiveness might be evaluated by means of utilized time and log-analysis; evaluations of satisfaction are often based on questionnaires or interviews and so on.

3. *Generalized processes*: This level represents functions and activities which are general and well-known in the work domain. In the experiment this process was represented by the 'consultation'. It is not a detailed specification of an activity but might be compared to the "black box" metaphor, as it is not important to specify sub-activities at this level. The example throughout this article is the consultation performed by a health care professional and a patient with or without a third person present (next-of-kin or an interpreter if the patient does not speak Danish).
4. *Information processing tasks*: This level represents sub-functions defining the generalized processes, human activities and the use of equipment. Typically they have the form of tasks that proceed or succeed the actual intervention, but contained within the consultation "black box". Example: One of the tasks during preparation involves looking at the overview of the past consultations to determine if there are any topics or events of special interests for the next upcoming consultation.
5. *Physical configuration*: This is the lowest level of abstraction and consist of tools or objects which may form the source of information for a given tasks. At this level we specify detailed descriptions of user interfaces, data requirements or prototypes. It is possible to point out which form and view in the prototype maps into this level. Appendix 2 gives an example where it is the Action Plan (form and view) from which goals and evaluations of the patient is registered and viewed.

The effect specification does not presume technology until it reaches level 4 and 5 (Figure 4). This exact characteristic of the effect specification is utilized in the specification process with the non-technical users. In the experiment we tried to employ the effect specification as a mechanism for "pushing" or translating the interpretation of *how* to obtain a given effect (in terms of using an IT system) towards those participants in the configuration context that have technological skills.

Level	Levels of effect specifications	Object of analysis – function in work domain
1	Purpose	organizational environment, purpose of the entire organization as such
2	Abstract function	prioritizing values, allocating resources
3	Generalized processes	functions and activities, general and well-known terms in the work domain
4	Information processing tasks	human activities, the use of equipment
5	Physical configuration	prototype, paper schemas

Figure 4 Object of analysis – work domain effects at descending levels of abstraction.

When reflecting at the dynamics of the effect specifications throughout the experiment we see that the top levels are relatively stable and the uncertainty is located within the physical levels of the hierarchy. The task of maintaining the documentation was heavier on the configurator-developer side than with the users, which was contributing to the acceptance of the users involved as they were pressed for time to consume large amounts of documentation and could rely on the process testing hypothesis at the general process level supported by the prototype.

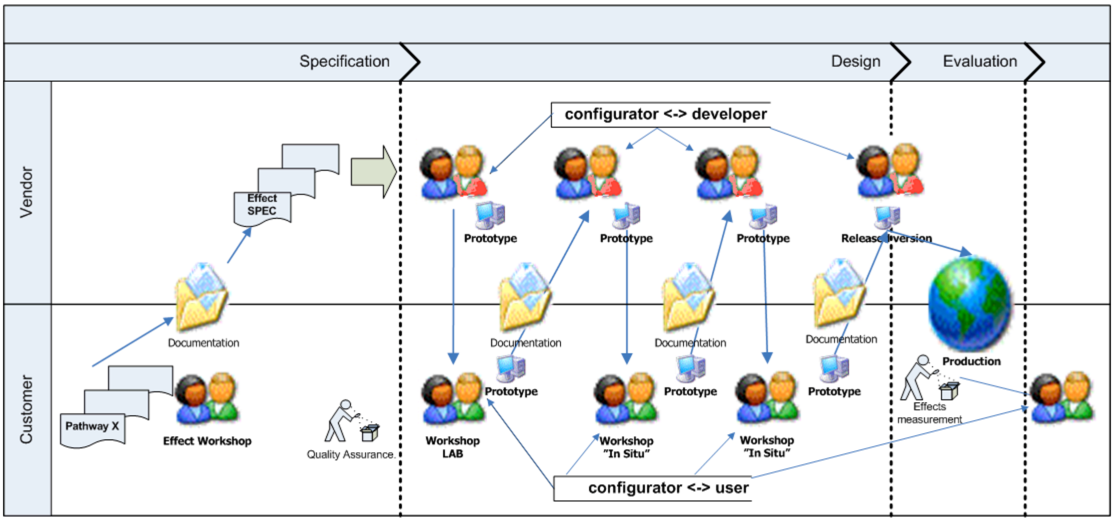


Figure 5: for design and configuration of a clinical pathway

Comparative analysis of an effect and use-case specification

The primary goal for the user participation process and deliverables in term of sufficient documentation is the successful transfer of knowledge from the users with firsthand experience from the domain, to the technical proficient (configurator and developer) tasked with the challenge to develop the system. Documentation using use-cases focus on process, effect specifications focus on function.

The hierarchy for both effects (figure 5) and use-cases (figure 3) is summarized for comparison in Figure 6.

Level	Levels of effect specifications	Level of use-case specifications
1	Purpose	Business Process Model
2	Abstract function	Business Process Model
3	Generalized processes	Use-case model
4	Information processing tasks	User-Interface model
5	Physical configuration	Object-model

Figure 6: The 5 levels of an abstraction hierarchy for effect specifications and – for comparison – the corresponding models from the use-case approach in terms of BPM, use-case, User-Interface and Object model.

In the following analysis we compare use-cases and effect specifications level by level as indicated in figure 6. We refer to appendix 1 for the complete use-case document; appendix 2 for the effect specification document; and appendix 3 for a table summarizing key contributions from effect and use-case specification for comparison.

At the *level 1* the use-case specification is constituted by the Business Process Model (BPM) defining the focus of the system interaction supporting a consultation (see appendix 1: *...conducts the consultation (e.g. admission interview) to obtain and document knowledge about the patient....*).
The effects at level 1 deal with impact on the organizations core and secondary activities. It also addresses impact on stake holders outside the organization; increase in patient motivation or political justification for existence of the centre.

> *"The policy of the municipality was integrated into the health care professionals' philosophy for best practise in consultations, and became the delivery of care to prevent and rehabilitate by supporting life skills [10] in an empowerment perspective" (see appendix 2)*

The work domain is present in both approaches, but the technology is evident already at this level in the BPM. This is seen as it includes start and end conditions which refers to objects in the IT-system (see appendix 1: *an Episode of care exist....*). The effect has no technology references, and is concerned with purpose of the work performed by organization, not the IT-system.

At *level 2* the Business Process Model (BPM) specification is addressing the context of the usage of the system; it includes a workflow performed in the work domain (in our case the consultation) as a sequence of actions some which involves the IT-system, terminology concerning; what makes it a consultation?, what exceptions are can be identified?
The effect specification has the distinction to what is considered important to the users in term of a priority, see appendix 2:
Effectiveness: The professionals' ability to comprehend the complexity of the individual patient pathways to improve transfer and reuse of knowledge.
Efficiency: By adapting the documentation to the work process before and after the consultation the time and effort spend doing these tasks would decrease.
Satisfaction: The ability to monitor the patients "wellbeing" in progressing the consultations to follow-up on their own perspective of quality of life.
The effects carry this user context on in the process when allocating resources while implementing those effects prioritised by the users.

At *level 3* the use-case is constituted by the use-case model which lists sources of information that the user can select when interacting with the system. The corresponding effects specification is focused on the impact on knowledge regarding the patient and the primary intervention of consultation. The object of the exercise on both approaches is to come closer to a requirements specification that will enable the configurator and developer to implement the prototype solution.
The effect is concentrated on honing the effect of the intervention performed in the work domain by looking at functions and activities in general and using well-known terms in the work domain: "*… the therapist to know the status and nature of the problems identified in the patient's pathway in a way that the therapist can account for the focus in the consultation*" (Appendix 2, Effects - Generalized processes).
The use-case focus on the user-system interaction, tasks involving the IT-system, transactions between users or other systems or services and hereby on describing

how the system should provide information: "…*The actor selects a function to view the clinical process.*" (Appendix 1, Use-case).
However as the users have stated their inability to predict how they want to work. For example, often the sequence of clinical tasks cannot be determine or agreed upon by the users, either because they want to act independently or because the clinical context requires it.
The use case specifies the process of the IT-system and allows for this by incorporating alternative sequences. The effect does not specify how the IT-system must be utilized in order to have the desired outcome, but states how the work domain is affected.

At *level 4*, the User-Interface operates on a functional level and is concerned with a functional or programmatic description about how the interface should behave and what data translate onto information, e.g. "*the actor selects... the system displays....*" (Appendix 1, User-Interface)
The effect changes character and becomes more like the use-case; it begins to address human activities, the use of equipment. This is illustrated by the agent changing from the users into the system, and the intervention becomes oriented towards the new agent – the system. For example: "*A view is activated with-in the patient context*" (Appendix 2, Physical process), this is the requirement documentation of hypothesis performed initially between the configurator and user, later it becomes a mechanism for coordination between configurator and developer while working out how the prototype can support the users.

At *level 5* the use-case approach is represented by the Object model and refers to both a physical level regarding software code, how data, tables and their relations are represented. HL7 models of how the objects in the clinical process relate or their characteristics in technical terms.
The effect has a similar physical role, using spreadsheets with data required for each screen in the prototype, paper documents used by the users when performing their work task before the project. The effect defines the physical form of data or functional requirements and is implemented in the prototype and represents the configurator and developers hypothesis regarding utility and anticipated impact on the user's interaction in the work domain.

Discussion

We have experienced limitations and assumptions with use-cases that we find problematic. I our experiment users rarely saw their work task as an interaction with the system alone, but were aware of the surroundings and interactions with their work environment without viewing each task as a series of use-cases. This might entail that the users find themselves involved in an unnecessary artificial situation when communicating with the configurator using use-cases.

The role of the effects is to keep focus on the knowledge originating from the users and allow the experimentation to continue without the confinement of technology to achieve them. Looking at the user-configurator, figure 8, the

absence of technology in the specification determines that IT is not the only way of achieving the effects. Effects could be applied to any other contributing factor such as organization, management, work procedure, etc. which all could form the basis of intervention for the project. For example an efficiency effect could be achieved by means of changing work processes LEAN (Womack and Jones, 1996) or other approaches.
Referring to the work domain as the object of analysis as suggested by the effect specification, the users were allowed to adopt a terminology natural to them without being put into a difficult situation relating to the system and the interaction with it defined as a sequence spanning the start and end condition. This can be seen as it is not until reaching the lower levels the users begin to consider the system as the agent for achieving effects in the work domain.
It appears as if the effect specifications – as a tool facilitating communication – allows the user to avoid or postpone the involvement in describing how the system should be shaped, in other words the system is not a central part in the knowledge exchange between user and configurator, but lies primarily within the configurator-developer dialog (see figure 8).

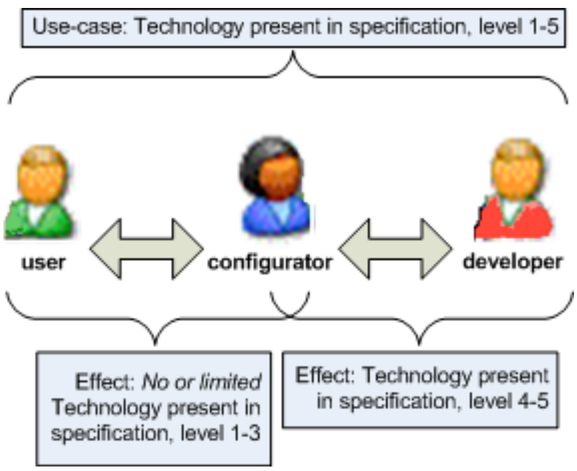


Figure 7: Technology presence in use-case and effect specification

We observed a displacement or shift in the translation of the effects into the prototype as the technology was not present until late in the specification. Although the effect was specified initially with the users, the effect at the lover levels became the focus of interpretation of the configurator and developer.
This allowed for a degree of "artistic freedom" in which the configurator and developer could explore how information was presented in an attempt to test the hypothesis – can the prototype deliver this effect to the work domain of the user? This presents both benefits and draw backs, as the configurator-developer team is free to be innovative, but also put high demand for skill within the team to handle this freedom of expression. This entails that a very close collaboration between all parties within the project becomes a prerequisite for the effects to carry the process.
By using effect specifications the user and configurator could focus on the usage effect in terms of impact on the work domain while the configurator and the developer could focus on specifying the system in terms of impact on functionalities and features. Use-cases, in comparison, require the user to

participate in specifying the functionality of the envisioned IT-system, thus applying terminology and notations that might not be familiar to the user at alle levels (figure 8). We regard this as the main result of the experiment as it opens for a potential organization of a configuration context where users can focus on conceptual proposals defining the problem and on desired outcomes in terms of specified effects, as opposed to a detailed functional specification.

Conclusion

The experiment has led to two general observations: *First* of all we succeeded in replacing use-cases with effect specifications as a tool. It supported the communication and coordination in an iterative configuration process by satisfying the users in expressing their needs and the developer in documentation from which to implement the system.
Second, we observed what differences could be identified?
Use-cases focus on process of the IT-system, where effect specifications focus on function in the work domain.
Use-case involve technology at all levels, effects does not until they reach a level of abstraction describing a physical involvement of the IT-system and then become more like the use-case with IT-system evident in the specification.
This allows effect requirements to be defined by the users without special technical skills until the specifications reach a level where an IT-system can be identified as evident in the user context.
Although the effect specification reaches a level where the technical element is present, it is still undertaken with the users. However the uncertainty of translation of the user context into an IT-system is displaced towards a coordination task undertaken between the configurator and developer. The effect and prototype is then used to bring the user back when confirming this interpretation of effects in the work domain.

Further effect specifications potentially allow us to deal with a broader audience in the user organization as the technical element can be postponed while maintaining a coherent model for specification.

References

[1] E. Balka, I. Wagner, and C. B. Jensen, “Reconfiguring critical computing in an era of configurability,” in Proceedings of the 4th decennial conference on Critical computing: between sense and sensibility, Aarhus, Denmark, 2005.

[2] J. P. Bansler, and E. Havn, "Information systems development with generic systems." pp. 707-718.

[3] K. Bødker, F. Kensing, and J. W. Simonsen, *Professionel IT-forundersøgelse : grundlaget for bæredygtige IT-anvendelser*, Frediksberg: Samfundslitteratur, 2000.

[4] M. J. Gallivan, and M. Keil, “The user-developer communication process - a critical case study ABA,” *Information Systems Journal,* vol. 13, no. 1, pp. 37-68, 2003.

[5] M. Hertzum, J. Simonsen, M. Granlien *et al*. "Effects-driven IT Development," www.effects-drivenit.dk.

[6] R. Baskerville, and J. Pries-Heje, “Grounded action research: a method for understanding IT in practice,” *Accounting, Management and Information Technologies,* vol. 9, no. 1, pp. 1-23, 1999.

[7] A. M. P. Jens Rasmussen, L. P. Goodstein *Cognitive Systems Engineering*: Wiley, 1994.

[8] K. J. Vicente, *Cognitive Work Analysis: Towards Safe, Productive, and Healthy Computer-based Work*, London: Lawrence Erlbaum Associates, 1999.

[9] J. Stapleton, *DSDM*, Repr. ed., Harlow: Addison-Wesley, 1998.

[10] B. J. Smith, K. C. Tang, and D. Nutbeam, “WHO Health Promotion Glossary,” *Health Promotion International,* vol. 21, no. 4, pp. 340-345, December 1, 2006, 2006.

[11] V. Zoffmann, and T. Lauritzen, “Guided self-determination improves life skills with Type 1 diabetes and A1C in randomized controlled trial,” *Patient Education and Counseling,* vol. 64, no. 1-3, pp. 78-86, 2006.

[12] L. Mathiassen, “Reflective Systems Development,” *SJIS,* vol. 10, no. 1+2, pp. 67-118, 1998.

[13] J. Ivar, E. Maria, and J. Agneta, *The object advantage: business process reengineering with object technology*: ACM Press/Addison-Wesley Publishing Co., 1994.

[14] S.-o. O. K. K. Center for Faglig Udvikling, *SUNDE KØBENHAVNERE I ALLE ALDRE Københavns Kommunes Sundhedspolitik 2006–10*, København: Rosendahls Bogtrykkeri, Esbjerg, November 2006.

[15] ISO, "ISO 9241-400," *Ergonomics of human-system - Part 400:Principles and requirements for physical input devices*, ISO, ed., ISO copyright office, 2007.

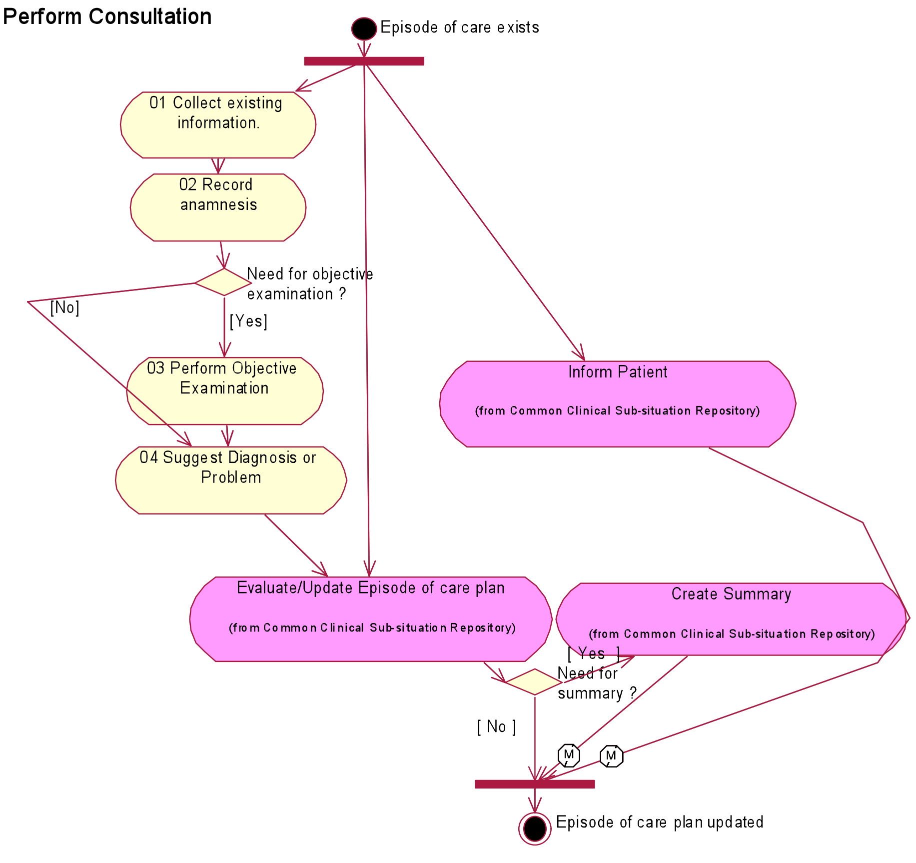
Appendix 1: Sample CSC Clinical Suite

1. Business Process Model

Perform Consultation

Purpose

The actor conducts the consultation (e.g. admission interview) to obtain and document knowledge about the patient and topics relevant to the actual situation. The consultation is conducted as a dialog. The dialog must give the involved parties a feeling of influence, responsibility and confidence.
The difference between a consultation and an intervention is:
- A consultation can in itself be an intervention
- A consultation contains creating/evaluating plans, where the intervention is more an update of plans.
- A consultation does not need a plan in advance unlike the intervention.



Precondition

Episode of care exists

Activities

01 Collect existing information.
The actor collects and peruses all relevant information on the patient, e.g. a list of other episodes of care, episode of care plans, the medical history, results from examinations and tests, various summaries and status notes.
02 Record anamnesis
The actor interviews the patient and/or relatives and documents the gained information.
03 Perform Objective Examination
The actor performs a physical examination of the patient. The examination can be guided by results from previous examinations.
The actor might involve other healthcare personnel for advice choosing either a formal or informal conference. The Staff Calendar helps finding the right person
04 Suggest Diagnosis or Problem
The actor gathers the information from the interview and the physical examination to a brief conclusion as a clinical document, drawn from one or more clinical notes or in some medical specialties as one or more classification codes.
A conclusion provides a general view over long or complicated episode of care and can be read as an independent document.
This leads the actor to set a diagnosis or to define a problem for the patient. The diagnosis or problem can be further specified with a status describing whether it is a suggestion, temporary or final and/or e.g. the severity of the condition.

Post-conditions

Episode of care plan updated

2. Use-case View Clinical Process

Name	**CP View Clinical Process**
Link to BPM	No link available
Purpose	The actor can view the Clinical Process for a given patient both as an overview and as detailed information. In order to give the actor a clinical process overview for a given patient clinical information, i.e. the patient's Clinical Problems and sub-problems, planned interventions, executed interventions and sub-interventions, results, Clinical documents, objectives and assessments, is displayed as hierarchies in relation to each other. In order to provide detailed information for any selected clinical process element.
Frequency	Up to several times a week
Start conditions	The patient is selected in Patient Context.
Special demands	
Final result	The Clinical Process is displayed.
Main approach	
1	The actor selects a function to view the clinical process.

2	The system displays a view of the clinical process.
3	Actor can choose which type of clinical information (i.e. clinical Problems, planned Interventions, executed Interventions or Results/Clinical documents) is to be displayed
4	When the actor selects a clinical process element the system displays detailed information about the element, e.g. a result.
Alternative approaches	

3. Object-model

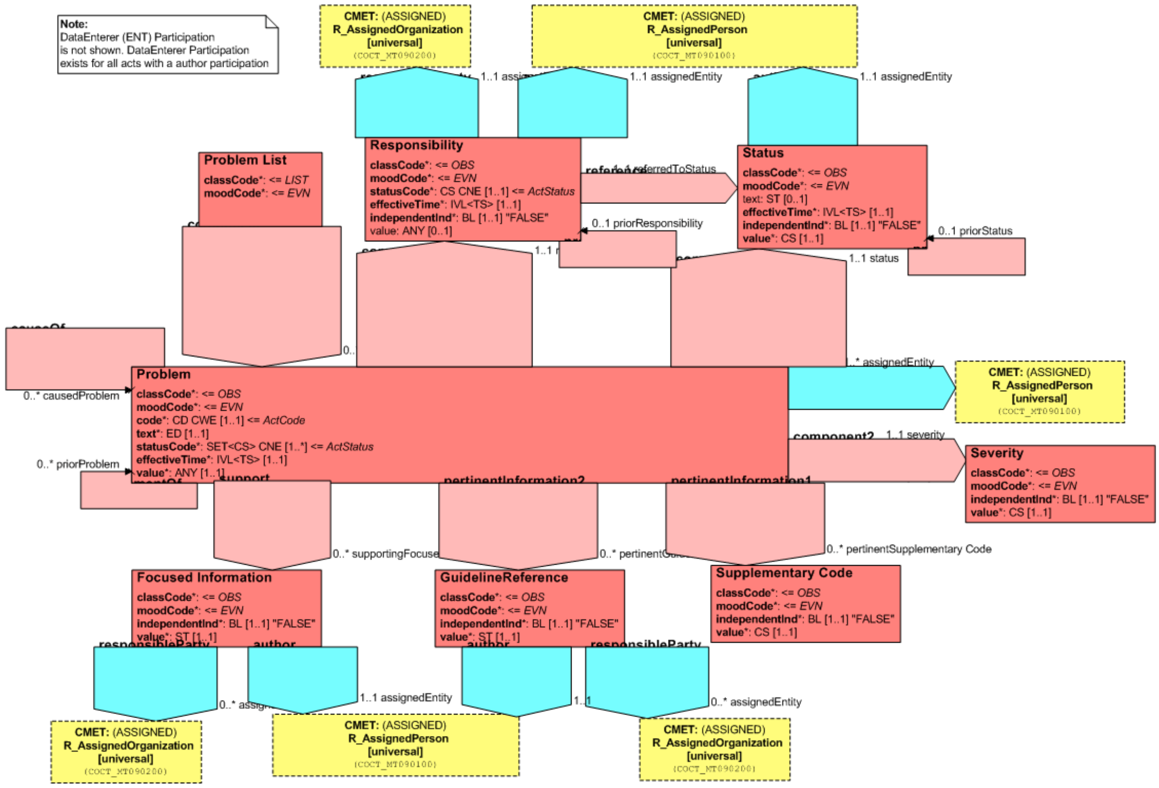


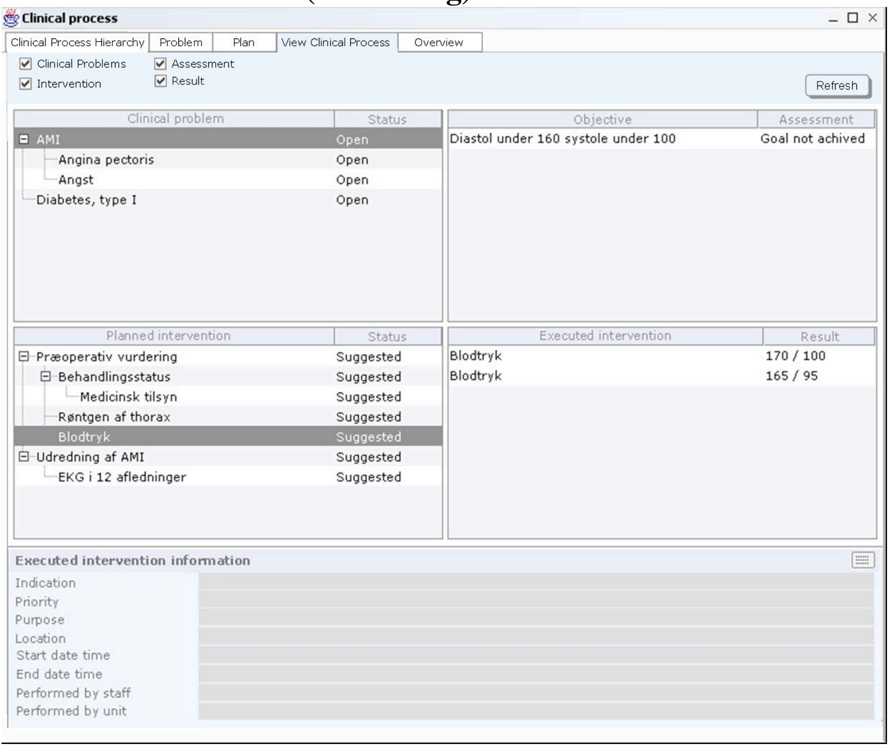
Figure 8 HL7 CSC Clinical Process Model

User Interface Clinical Process

To view the Clinical Process. The view displays all the patient's clinical problems and sub-problems, planned interventions and sub-interventions, executed interventions and sub-interventions, results, objectives and assessment.
When a clinical process element is selected its detailed information is displayed.
This view is intended to illustrate the clinical process circle.

User Interface – sketches

View Clinical Process (Tab dialog)



The checkboxes in this block are used to select or deselect parts of the clinical process. Each part of the clinical process is displayed in its own window. A checkbox is selected using a key or mouse and a mark is set or removed. When one checkbox is marked or unmarked the view of the clinical process is changed accordingly. The view can be refreshed.

……

Appendix 2; an example of the effects specification document

Purpose 1.1: To be able to evaluate the centre core and secondary activities.
Purpose 1.2 To contribute to an increase in patient motivation
Purpose 1.3: To be able to show the justification of the centre in the presence of the politicians (sponsors)
Condensation from WS with Cancer Centre "The policy of the municipality was integrated into the health care professionals' philosophy for best practise in consultations, and became the delivery of care to prevent and rehabilitate by supporting life skills [10] in an empowerment perspective"
Abstract function 2.1 The professionals' ability to comprehend the complexity of the individual patient pathways to improve transfer and reuse of knowledge.
Abstract function 2.2 To have the ability to monitor the patients "wellbeing" in progressing the consultations to follow-up on their own perspective of quality of life.
Abstract function 2.3. To adapt the documentation to the work process before and after the consultation the time and effort spend doing these tasks would be less.
Abstract function 2.4 To be able to handle termination of pathways … without "overtreatment"
Abstract function 2.5 To be able to administer a more uniform evaluation of the patient by the staff, (best practice, good care)
General activities 3.1 Overview before performance of intervention Result: Overview before intervention (measured) Intervention: A consultation with a citizen/patient. Agent: Therapist Goal: The therapist knows the status and nature of the problems identified in the patient's pathway in a way that the therapist can account for the focus in the consultation.
…
3.3 **General activities** Pathway overview – time aspect Result: an overview of the pathway through time Intervention: the tracking and identification of discrepancies or deviations in the planned or anticipated pathway. Agent: Secretary Goal: To verify if the execution of pathway is in accordance with standard (expected) or plan. E.g. to ensure pathway put on hold is resumed.
4.1 **Information processing tasks** Overview before the performance of the intervention Result: A list of key information's necessary for the completion of; the start-up consultation, follow-up or ending-consultation, training exercise etc. Intervention: A view is activated with-in the patient context. Agent: System Goal: To illustrate missing or incomplete information with the purpose of accessing or searching for the relevant supplements.
…
4.5 **Information processing tasks** Pathway overview – time aspect Result: a list of dates representing the interventions in the individual patient's pathway Intervention: A view is activated with-in the patient context. Agent: System Goal: To illustrate the dates for planned and performed interventions in the patients pathway.
...

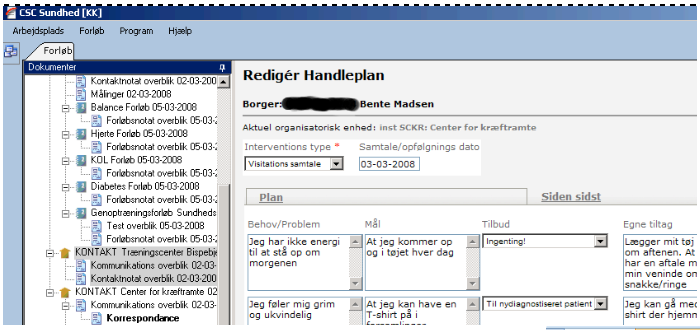


Figure 10 Screen-dump of prototype

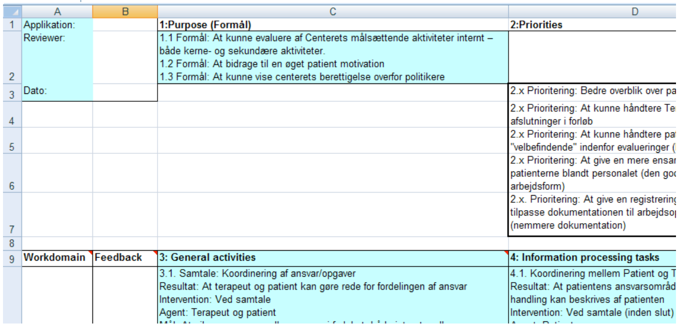


Figure 9 Screen-dump of Spreadsheet documentation tool