Accepted to The first international workshop on methods tools and architectures for enterprise management and resource planning (EMRPS '99), Venice, 25th- 27th of November, 1999.

A Strategy for Organisational Knowledge Evolution

Lars Taxén
Ericsson Utvecklings AB
Department of Computer Science and Information Systems, The University of Linköping
E-mail: lars.taxen@uab.ericsson.se

Abstract

In this paper, a strategy for organisational knowledge evolution is presented. Organisational knowledge is defined as knowledge concerning the management of the artefacts provided by the organisation. The strategy is derived from a theoretical, action-oriented representation of knowledge and uses concrete instruments for iterating between reflection and action. The instruments for reflection are *conceptual models* and *information flow diagrams*. The instrument for action is an *object oriented information management system,* where the models are implemented and tried out in practise. By applying these instruments iteratively, organisational knowledge is generated, both as individual and shared knowledge among the actors, as well as objectified knowledge represented by the models and the implemented information management system. We describe how this strategy has been used at the Ericsson telecommunication company to handle the transition to a new software development model. The transition was complicated by the fact that Ericsson has many designers (more than 10 000) working at local design centres all over the world. Our experience shows that the proposed strategy is a powerful way to quickly acquire, deploy and manifest new organisational knowledge[1].

1 Introduction

The telecommunication market is changing rapidly mainly because of two factors: the entering of many new operators due to the deregulation, which leads to more competition, and the proliferation of new technology, for example mobile communications, intelligent networks, the Internet etc. This has put a demand on the suppliers to be more reactive and flexible to the market needs, which means shorter lead-times and more flexibility in handling late and changing requirements. It is a very challenging task to achieve this considering the size, complexity and in service performance (uptime) requirements of the telecommunication systems.

One example of this situation is shown in figure 1, which is from a software project developing cellular telephone features for the Japanese market. Less than 50% of the original set of requirements remained unchanged during the project. The rest was

1. This work has been sponsored by the Swedish National Board for Industrial and Technical Development, NUTEK, project P10518-1, and The Industry Research School in Applied Information Technology and Software Engineering at Linköping University, funded by the Foundation for Knowledge and Competence Development.

either new, changed or removed requirements, which makes it very hard for the project to fulfil its goals on time. Ericsson is trying to handle this situation by gradually replacing the traditional, phase oriented waterfall development model with an *incremental development* model, which makes a project less sensitive to requirement changes. For example, in this model, late incoming requirements can be allocated to later increments.

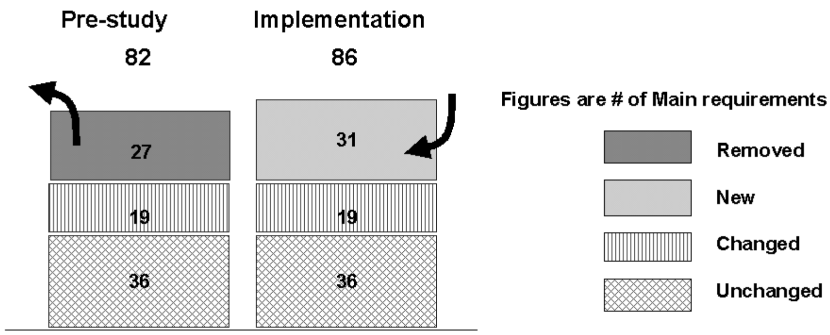


Fig. 1. Typical requirement picture for development projects

To achieve this, an extensive organisational knowledge evolution must take place, since a large number of designers (more than 10 000) are used to the waterfall model. The transition is complicated by the fact the development is usually distributed to local design centres all over the world, where specific local development traditions have evolved. The knowledge evolution means that a new intersubjective understanding must be achieved between the actors, and that the acquired knowledge somehow must be institutionalised within the organisation.

The purpose of this paper is to describe a strategy for organisational knowledge evolution and to discuss the results from applying this strategy in the transition to incremental development. In chapter 2, we give a theoretical, action oriented view on knowledge where *interaction* is regarded as the basic knowledge category. This enables us to structure both individual and shared knowledge as dynamic contexts characterised by a focus, a scope, a pattern of categories and a logical ordering between the categories.

In chapter 3, we propose a definition of organisational knowledge as *knowledge concerning the management of the artefacts provided by the organisation*. We also state some problems regarding management that need to be solved.

Next, we describe the strategy for organisational knowledge evolution, which is based on the theoretical considerations in the previous chapters. The purpose of the strategy is to gradually obtain a context for the management of the artefacts. This is done iteratively by a group of actors, which reflect about the context and try out their reflections in action. In doing so, concrete instruments are used for reflection and action. The instruments for reflection are *conceptual models* and *information flow diagrams.* The instrument for action is an *object oriented information management system,* where the models are implemented and tried out in practical development work. In this manner, organisational knowledge is generated dialectically, both as individual and shared knowledge, as well as objectified knowledge represented by the models and the implemented information management system.

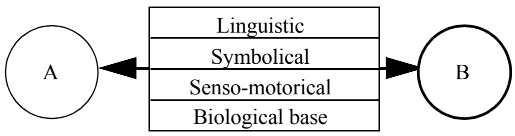
In chapter 5 we will show how this strategy was applied to the transition to incremental development at Ericsson. The results are discussed in chapter 6.

2 The concept of knowledge - a theoretical frame of reference

What is knowledge? In Berger & Luckman (1966), knowledge is defined as "*...the certainty that phenomena are real and that they possess specific characteristics*". Molander (1966) speaks about "*knowledge as a form of attention*", while Rolf (1995, referring to Polyani) looks upon knowledge as "*a process where individual and culture interacts*". Goldkuhl (1993) defines knowledge as "*patterns of categories*". Knowledge is often defined indirectly by some aspect of knowledge like practical, tacit, living, personal etc. As with many similar multi-facet concepts, it is difficult to capture the essence of the concept of knowledge. In the following sections, we will describe a view of knowledge which we have found useful for the purpose of representing organisational knowledge. This view is inspired by, amongst others, Volosinov (1986), Kosik (1978) and the classical work of Jacob von Uexküll (1909).

2.1 Interaction based knowledge

One possible view of knowledge is that knowledge is created when *someone is doing something in relation to something*. The *interaction* is where knowledge is born and sustained, and without interaction, the concept of knowledge has no meaning. This applies to every organism that has the ability to interact with its environment and to build up a persistent preparedness to act from made experiences. The preparedness will guide the organisms' way of reacting in similar and new situations. In this sense, there is no difference between a human, a tree or a virus[2]. All of them will learn from the interaction with their environments in different ways. While a virus only relates to nature, a person also relates to other persons and to society. Furthermore, it is possible to structure the interactions in strata, where an underlying stratum is a prerequisite for the strata above.[3] This kind of stratification could be described with the following interaction model:



All the prerequisites that are needed for an organism's interaction with its environment are found in the biological base. The interaction in this stratum could for example be the cells' responses to different chemical reactions, where the preparedness may be the geometric structures of the proteins. In the senso-motorical stratum, the interaction

2. Even artefacts like a neural network could be regarded in this way.
3. A good analogy is the so-called Open System Interconnect (OSI), which describes a stratified architecture for the communication between nodes in a network.

takes place through the sensory organs and their ability to orient in and comprehend the surrounding environment. In this case, the preparedness is comprised of different sensory memories such as muscle, hearing, touching and seeing memories. In the symbolical stratum, the interaction takes place through symbols, and in the linguistic stratum, through syntactic and semantic terms in the language. The linguistic and symbolical strata can together be characterised as the world of signs carrying meaning to the members of a certain community.

In this paper, we will assume that an organism's knowledge is equal to its preparedness to act in different situations. This action oriented knowledge can in principle be generated in all strata. Knowledge that originates from interaction in the two top strata can, in principle, be articulated and subjected to reflection and reconstruction. Knowledge within the two bottom strata cannot be articulated. The conception *tacit knowledge* (Polyani, 1966) is sometimes referred to as knowledge that cannot be articulated, and would therefore belong to these two bottom strata.

2.2 Patterns in contexts

An organism is learning something when something gets in focus in its interaction with the environment. The interaction links the organism's inner knowledge structure with the perceived structure of phenomena in the outside world. What is in focus in a certain situation, is governed by the internal driving forces of the organism and by events in the environment. The knowledge, which is being perceived in a given situation, may be called *focal knowledge* (Rolf, 1995). Besides the focal knowledge, another kind of knowledge is needed to complement and direct the focal knowledge. This knowledge is called *background knowledge*.

Put together, the focal knowledge and the background knowledge can be described as knowledge in a *context* that is delimited by a *focal horizon*. Knowledge above the focal horizon is active in a given focal situation, while other parts of knowledge are latent or resting in the very same situation, no matter what stratum is concerned. Necessarily, the limiting of a context through the focal horizon is blurred and vague. Furthermore, following Goldkuhl's definition, the knowledge in a context has to be structured in some pattern where categories of knowledge are associated with each other. An example of a context is found in figure2:

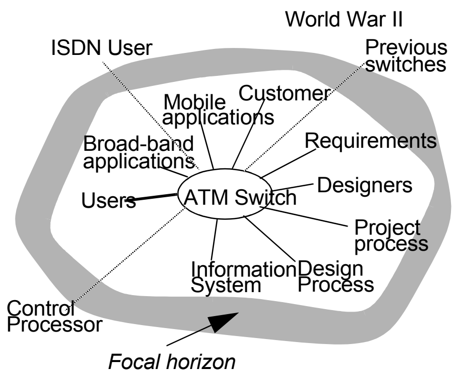


Fig. 2. Example of a context

When a context has been learned by an organism, it can be used in corresponding focal situations. The pattern in the context is used to draw conclusions about the situation and to act in accordance with the information that results from the interaction between the focal situation and the context[4]. Thus, a context has both a static and a dynamic aspect. The static aspect is the pattern, i.e. how knowledge categories are related to each other. The dynamic aspect is the order in which the information is created and used according to the pattern. Thus, knowledge in a context is defined by:

- a focus,
- a scope delimited by the focal horizon,
- categories organised in patterns,
- a logical ordering of categories which shows the order in which the information is generated and used.

2.3 Field of Meaning

When the focus changes, so does the context. Certain categories will disappear below the horizon, while others will remain visible:

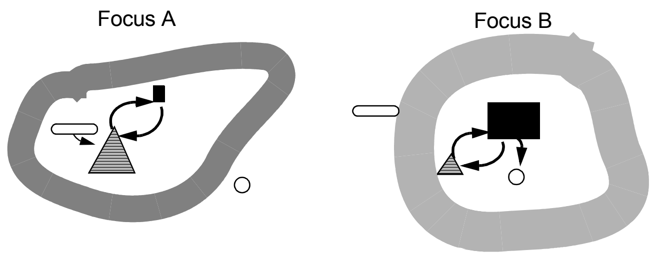


Fig. 3. Focus change

An organism's entire knowledge can be interpreted as its capability to acquire, sustain and activate patterns of categories for focal situations that concern the organism. These patterns will make it possible to recognise the situation and act accordingly. However, if the organism encounters a new focal situation, a learning must be accomplished by acquiring a new pattern corresponding to that situation. This pattern is based on former patterns which may be considered as *pre-understanding* to the new learning.

The entire pattern of categories, which is institutionalised in the organism, can be described as the organism's *field of meaning* (Goldkuhl, 1993). Thus, when the organism acts on a certain focal situation, this is done in accordance to a context where the relevant parts of the field of meaning have been activated. This applies to all the strata in the interaction model, including the senso-motoric. In the two top strata of the interaction model, the field of meaning can be articulated and explicitly described, which is utilised in the strategy described in chapter 4.

---

4. Note that the same focal situation in general brings different kinds of information to different individuals, depending on the fact that their pattens are structured differently.

3 Organisational knowledge - managing artefacts

So far, we have been discussing knowledge from an individual point of view. In order for several organisms to interact, it is necessary that some shared or intersubjective understanding takes place. This means that the interacting organisms have some parts of their field of meaning in common. The common parts are shaped in the dialectical process taken place between the individual's subjective reality and the socially constructed objective reality (Berger & Luckman, 1966). This also applies to the individuals belonging to an organisation. If the organisation is to appear as a unit, a common field of meaning must exist amongst its members.

An organisation needs many specific types of shared knowledge. For example, knowledge about how to develop object oriented software might be necessary for the organisation to fulfil its goals. However, since this domain specific type of knowledge is to a great extent independent of the particular organisation, it cannot be considered *organisational* knowledge. In general, there seem to be no generally accepted meaning of this concept, which can be attributed to the lack of attention to fundamental, precise concepts regarding organisational knowledge (see for example Huysman and van der Blonk, 1998).

We suggest that organisational knowledge must possess at least the following characteristics:

- It is must be tightly related to the purpose of organisation and its goals.
- The knowledge is both individual and shared among *all* the actors in the organisation.
- It must be objectified as descriptions, rules, norms, institutions and the like which are valid for the entire organisation. Otherwise, it is not possible to maintain the knowledge between generations of actors.

Based on these considerations, we define organisational knowledge as knowledge associated with the *management of the organisations' artefacts*, which fulfils the characteristics above. Artefacts may be products developed, services provided etc. These artefacts motivate the existence of organisation, and it is obvious that knowledge about how to manage these must be considered core knowledge in any organisation. In this paper we will call the artefacts *managed items*. Some problems that need to be solved in connection to these items are:

- What items are managed?
- What are the relationships between the items?
- What is the status of a managed item at a particular instant of time?
- In which order is the items managed?
- What properties characterise a managed item?
- How should the responsibilities for managing the items be allocated?
- What guidelines, rules, norms etc. are associated with the management?
- Which is the role of management support systems in relation to other existing infor-

mation management systems in the organisation?

Thus, we will apprehend organisational knowledge as a field of meaning for the managed items of the organisation. In chapter 6 we will discuss to what extent this approach makes it possible to answer the questions above.

4 A strategy for organisational knowledge evolution

In this chapter we describe a strategy for organisational knowledge evolution based on the theoretical considerations in the previous chapters. The purpose of the strategy is to gradually obtain a field of meaning for managed items. As we have seen, the field of meaning can be interpreted as patterns of categories organised in contexts, where each context is structured to handle a particular focal situation concerning the management of the organisations artefacts.

The starting point is that the organisation needs to evolve its organisational knowledge due to for example external events like a changing market. Thus a modified field of meaning must emerge within the organisation. This is achieved in the following way:

- A group of actors is provided with concrete instruments for reflection and action, which they use in a dialectical interplay to achieve shared understanding or intersubjectivity.
- The instruments for reflection are *models* of the static and the dynamic aspects of a context. The static view is modelled by *conceptual models*. The dynamic aspect is modelled by *information flow diagrams*. These models are further described in sections 4.1.
- The instrument for action is an *object oriented information management system*. By action, we mean that instances of the managed items are being created and managed based on the type definitions in the reflection models. The experiences gained from this will, in turn, affect the models for reflection. In order to maintain the dialectical interaction between reflection and action, it must be very simple and straightforward to implement the reflection models in the system. How the implementation is done is described in section 4.2.
- Intersubjectivity is achieved when the actors agree that the models and the implementation in the management system sufficiently describe the new context.
- The new knowledge is institutionalised and manifested by the models and the implemented system, which will be part of the organisation's objective reality. To make the objectification easier within large distributed organisation like Ericsson, the models and support are made adaptable to local circumstances. This is further described in chapter 4.3.

4.1 Instruments for reflection

The instrument for reflection of the static aspect of a context is the Object Modelling Technique (OMT) notation[5]. The most important concepts of OMT are described in figure 4. The categories are described as "boxes" in the OMT, while "lines" between the boxes indicate an association between categories. The complete OMT-model describes the pattern of categories for a particular focal situation.

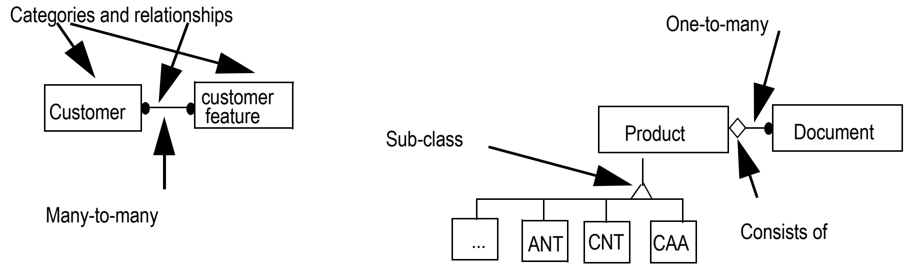


Fig. 4. OMT-notation

The instrument for reflection of the dynamic aspect of the context is information flow diagrams. These belong to a category of process models usually called “Entity Based Process Models” (Humphrey & Kellner, 1989). The focus in these models is put on data status rather than on activity status. The principles of the information flow diagrams are illustrated in figure 5. A downward pointing arrow shows that the category is used in an activity. An upward pointing arrow from an activity changes the status of the category. The categories are the same as in the conceptual model. The information flow diagrams describe in what order the information associated with the categories is processed.

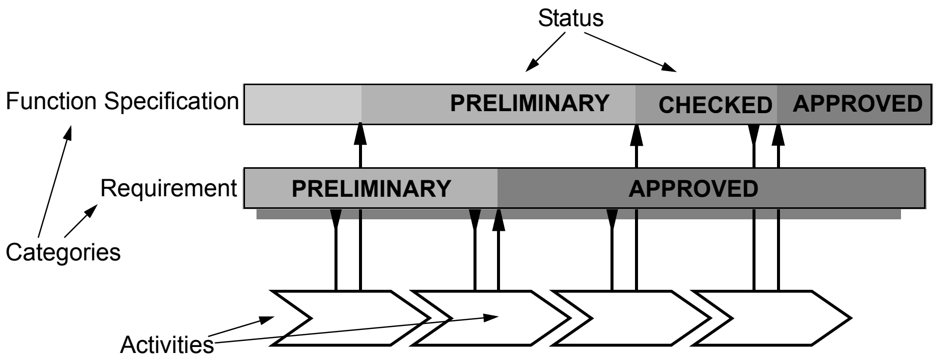


Fig. 5. Information Flow Diagrams

The OMT models and information flow diagrams are being used alternatively during the reflection phase. These models were chosen because they are easily understood by the actors and model the context on an appropriate abstraction level.

5. Other modeling languages may be used, for example the Unified Modeling Language (UML), as long as the language is easily apprehended by the actors.

4.2 Instrument for action

In order to maintain the dialectical interaction between reflection and action, the OMT-models and the information flow diagrams must be directly tried out in the action phase. Thus it is very important that the reflection models can be implemented straight on and that they are easily recognised in the instrument for action. The product data management system Matrix from Matrix-One Inc. is a system with these qualities. The OMT notation is implemented in the business modelling part of Matrix, where object- and relation types are defined. Patterns are formed by stating what types of objects that can be connected to a relation, see the example in figure 6:

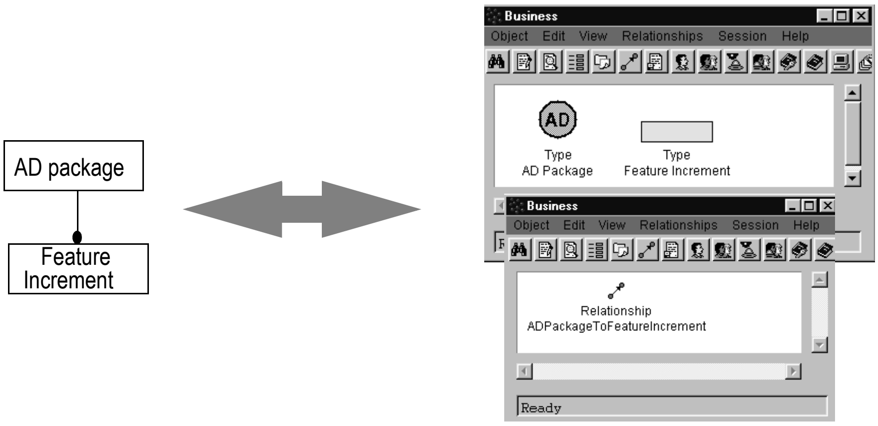


Fig. 6. Definition of object types and relations

Information flow diagrams are implemented by defining the different values of status that an object can assume during its life cycle, see the example in figure 7:

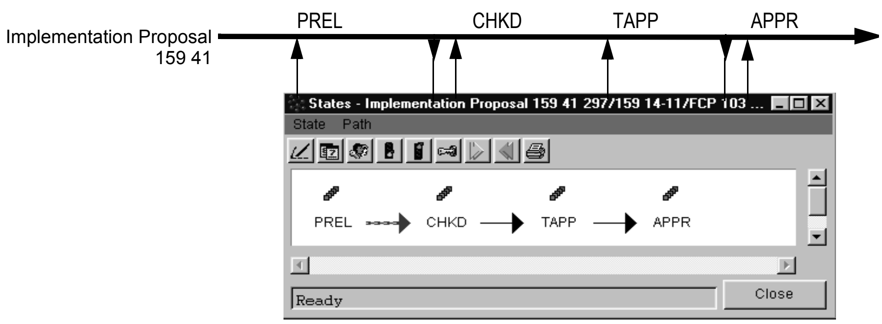


Fig. 7. Definition of information flow

4.3 Adaptation of enterprise strategies to local circumstances

In a large and distributed organisation like Ericsson, there will always be a need to adapt enterprise strategies to specific circumstances at the local design centres around the world. At the same time, there must exist common policies for enterprise tasks, such as for example how to identify products and documents. The balance between strict control and decentralisation is often difficult to maintain, and the organisation tends to oscillate between these extreme points from time to time. In either case, the ability to adapt to changes will be low[6], as illustrated in figure 8:

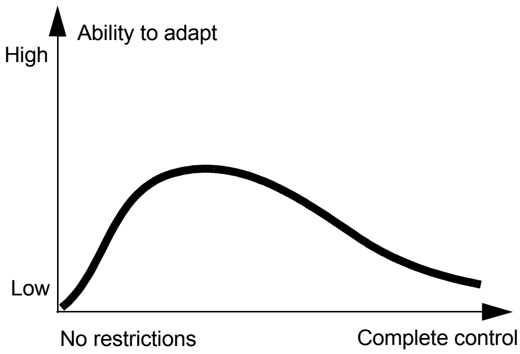


Fig. 8. Ability to adapt versus control

This also applies to the proposed strategy for organisational knowledge evolution, which basically is an enterprise strategy. To avoid the risk of over- or underspecifying the strategy, the provided management support (reflection models and implemented support system) are articulated on the enterprise level as a starting point for adaptations. In essence, this means that the provided enterprise management support is not operational unless finalised at the local design centres. This will accomplish several things:

- The user acceptance of the local management support will be very effective, since the intersubjectivity is achieved in interaction with the actors that are also the users of the support. The difficulty to gain acceptance of new ways of working and of new support systems in an organisation is gravely underestimated. By making the users partly responsible, the support will quickly be established in the local organisation.
- Achieving consensus on the strategy will be much easier, since it only concerns the starting point for adaptations. It is futile to aim for a consensus on an operational enterprise management support other than for very stable or mandatory items.
- The possibility of adaptation makes it easier to find and adjust the optimal balance between enterprise-wide needs and local needs.
- The local knowledge evolution will be quickly built up, since no coordination is needed. In principle, each local organisation will get a tailor made management support according to their needs. It is however desirable that those local adaptations, which are of general interest, can be incorporated in later updates of the enterprise management support.

6. For a discussion on these matters, see for example Nilsson (1976)

5 Transition to an incremental development model - a case study

Within Ericsson, the development of software for the telecom services has for a long time been done according to the waterfall model. This means that the development is done in sequential phases, where a certain result is accomplished and fixed after each phase. For example, the requirements are analysed and frozen very early in the development as is illustrated in figure 9:

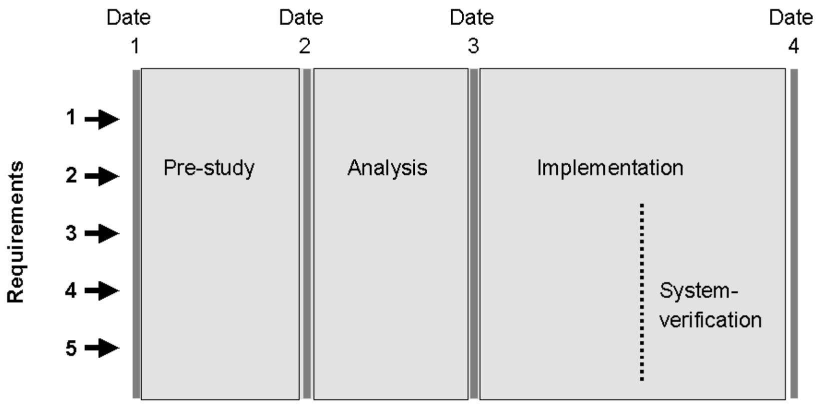


Fig. 9. The waterfall development model

This model is objectified in a method product, which is used by more than 10 000 designers at local design centres all around the world. Some projects are very large (up till 500 000 man-hours) and employ several hundreds of designers at 15 to 20 design centres in several countries. Often, adaptations of the method are made due to local conditions, but the basic foundation remains the same. This means that the method product is a manifestation of intersubjective understanding among Ericsson employees throughout the world.

Recently, is has become obvious that the waterfall model is not suited for the dynamic market situation of today, especially not when it comes to requirement changes during a project. For some time now, different ways of incremental models have been tried at several local design centres. In the beginning of 1996, the time was ripe to consolidate these attempts into a productified method and tool support for incremental development.

In the incremental development model, the whole development assignment is divided up in steps (increments) which can be developed and tested as independent units. This means that there is no need to freeze all the requirements at the exact same time. Late incoming requirements can be directed towards late increments, and removed requirements will only affect single increments. This means greater flexibility with respect to requirement changes, than the waterfall model can offer. Figure 10 shows a principal view of the incremental development model:

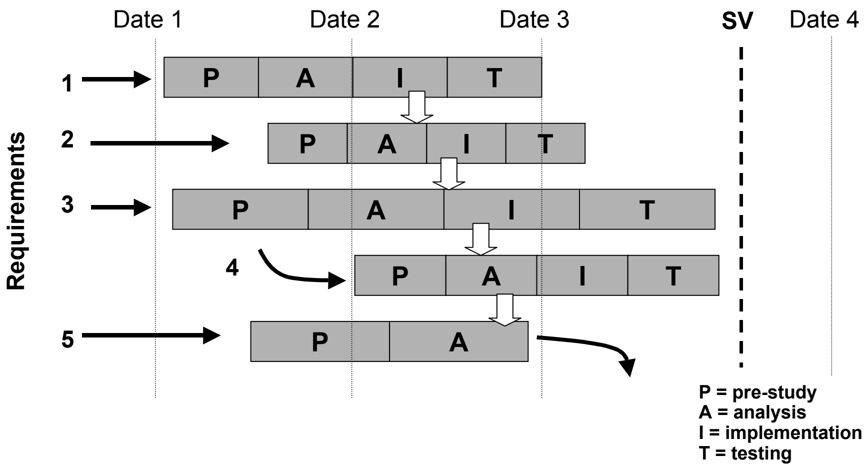


Fig. 10. Incremental development

5.1 Incremental development -phase one

The transition to incremental development was done in two phases. The purpose of the first phase was to develop a method package for incremental development which should support identification, specification, planning and coordination of the increments. The preconditions were:

- No support system was to be specifically developed for the incremental model.
- The management of the information was to be done in a traditional way, i.e. as documents.

To achieve this, a small team (around 10 people) was set up consisting of method developers and a support group for projects denveloping cellular telecom systems for the Japanese market. Since several local development projects already had used some form of incremental development, a form of objectification had taken place at some parts of the organisation, but no intersubjective understanding of incremental development had been achieved within the entire Ericsson. The discussions on how to shape a common model were, in principal, leading nowhere until we started to work with conceptual models according to the OMT notation. After some 30 revisions of the model, there was a limited agreement about the model in figure 11, which may be regarded as the first version of the field of meaning for managing incremental development. The model can also be seen as an articulation of previously tacit knowledge concerning development work at Ericsson.

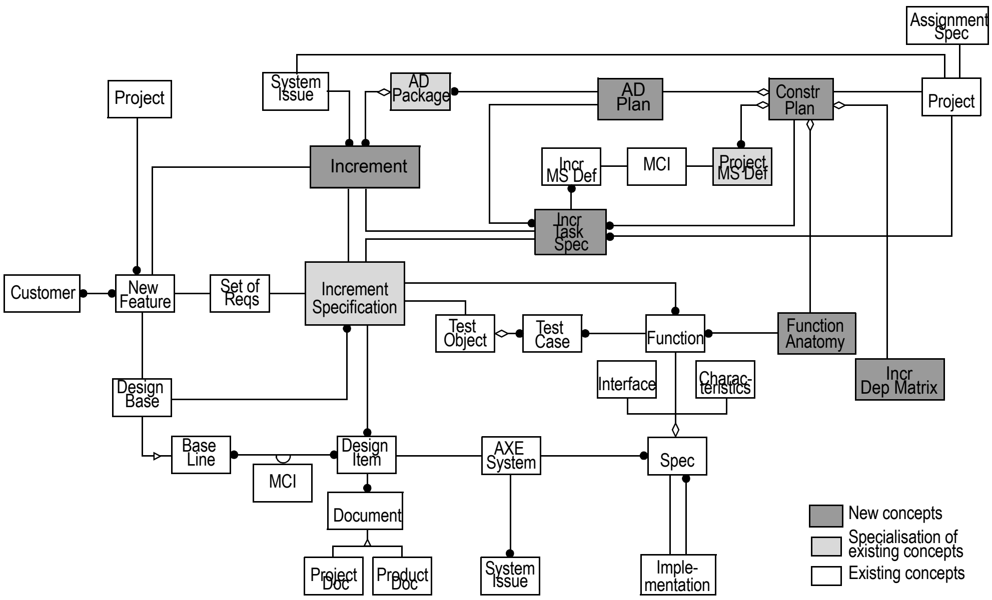


Fig. 11. Conceptual model - phase one

Some interesting observations are:

- The traditional method categories are clearly marked (white boxes). These were not affected. The new categories were either completely new or already existing ones used in a new way. Thus, the model is articulating both traditional knowledge as well as new.
- The managed items are either pure information elements (for example “Function”) or configuration items, which group together a certain set of information elements. Since documents are the traditional form for both types of managed items at Ericsson, the impact of the tradition made it difficult to clearly see this double role of documents. The result was that both document items and information elements are present in the model, which turned out to be confusing.
- Focus (“Increment Specification”), focal horizon and patterns of categories are all represented in the figure. This corresponds to the definition of knowledge in a context given in chapter 2.2.

The method development took about a year to complete, and the first version of the incremental development model was used in a cellular project for the Japanese market. The experiences from this project showed that the model did work, but that a support system was necessary to manage all the information during the project.

5.2 Incremental development - phase two

The purpose of phase two was to develop a support system for managing incremental development. In this phase, the strategy described in chapter 4 was fully utilised. It was anticipated that the development model would be affected when the support system was brought into focus, which is also confirmed by the conceptual model achieved after this phase (see figure 12). The corresponding information flow diagram is shown in figure 13. Compared to the preceding model, it can be noted that:

- The role of documents as configuration items has to a great extent disappeared, and the categories are either information elements or configuration items. This is a consequence of the support system (Matrix), which makes it possible to extract information from different perspectives, and, if necessary, account for it in the traditional way, i.e. as documents which group together a set of information elements.
- The model contains very few categories that are specific to incremental development.
- The model has become less complex, which makes it easier to understand and therefore also to get acceptance of.

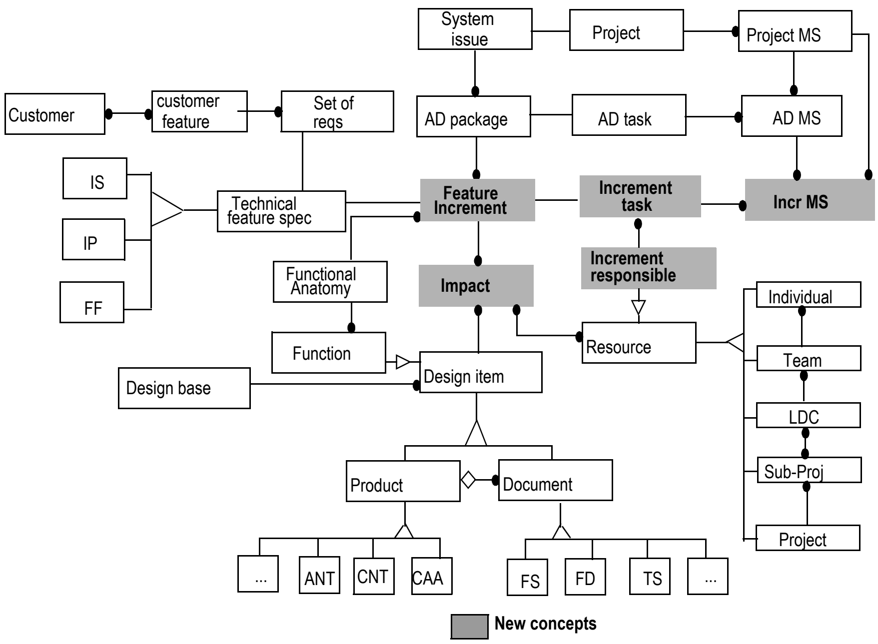


Fig. 12. Conceptual model - phase two

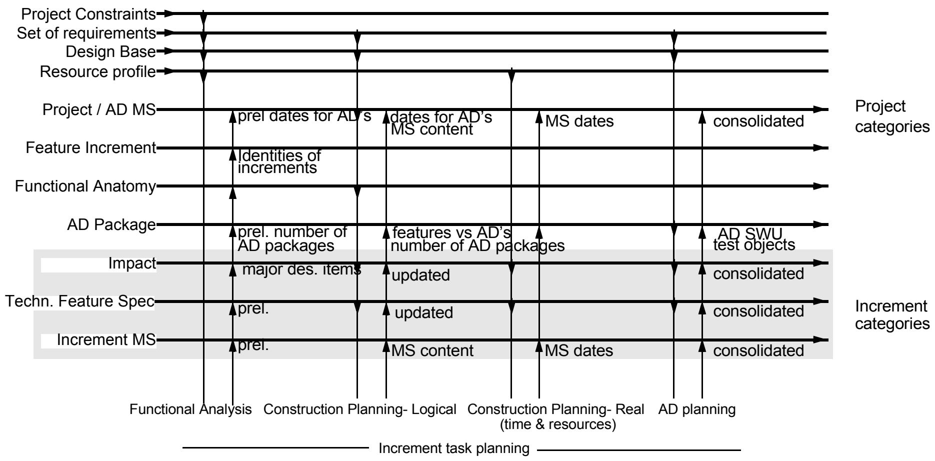


Fig. 13. Information flow diagram - phase two

When the entire field of meaning has been established, specific contexts corresponding to particular focal situations can be formed. For example, a context can be articulated by starting from a focal category and traversing the categories in the context along the relations between categories. One example of this is shown in figure 14, which shows a context with the customer in focus and those managed items which are impacted by a set of requirements from the customer.

Object	Revision	State
Req Issuer PN		New
Input Req MR-1 C	C	AGREED
Detailed Req I-10		AGREED
Detailed Req I-10-01		UNDEFINED
Integration Increment 1 -	-	Identified
Function 01 Start / Restart -	-	READY
CNT 213 1054 R2	R2	PREL
Application Information 155 18 2/155 18-CNT 213 1054	C	PREL
Application Information 155 18 2/155 18-CNT 213 1054	C1	PREL
CAA 107 5256 R2A	R2A	PREL
Data Change Information 109 26 4/109 26-CAA 10	A	PREL
Document Survey 1095 1095-CAA 107 5256 R2A	R2A	PREL
Signal Survey 155 14 155 14-CAA 107 5256 D	D	PREL
Source Parameter List 190 73 190 73-CAA 107 52	C	PREL

Fig. 14. Traversing the context

5.3 Further results

Due to a company restructuring, the work with the incremental method package was terminated in 1998 and the team dispersed. However, the experiences from this work have been used in a similar effort to support the management of integration driven development, which in essence is the same as incremental development. For example, the evolution strategy has been used in the transition from document based to a computer based engineering change order process, and the results are equally positive. In a couple of months, we achieved a fully operational computer based process as well as an intersubjective understanding among configuration managers, project leaders and support staff.

The current conceptual model can be seen in figure 15:

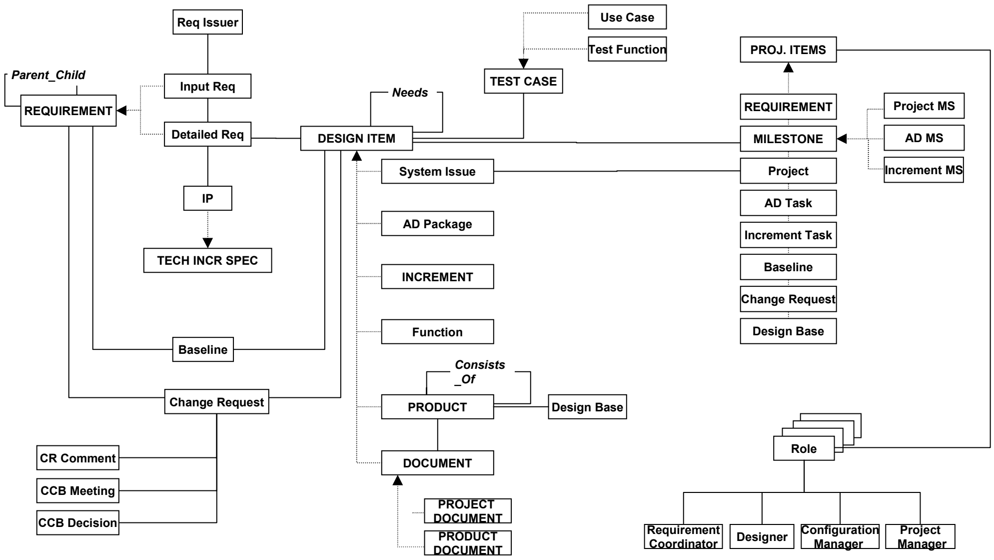


Fig. 15. The current conceptual model

In comparison with the previous models, this model has been further refined. Also, categories concerning requirement- and engineering change management have been added. This support is now used in several, globally distributed projects at Ericsson.

6 Discussion

In this paper we have presented a strategy for organisational knowledge evolution which is based on an action oriented theory of knowledge and which uses concrete instruments for reflection and action. Organisational knowledge is defined as knowledge concerning the management of the artefacts provided by the organisation. We have described how this strategy has been applied in the transition to an incremental way of developing software for telecommunication systems at Ericsson.

The results so far show that the strategy provides a quick and efficient way of generating, deploying and maintaining new organisational knowledge among the participating actors. The total work effort of developing the models, implementing the support system and achieving consensus among the actors accounts to between 2000-3000 man-hours, which in our opinion is surprisingly small.

A lot of effort was spent on discussing the conceptual model, which was constantly revised together with the corresponding support[7]. One reason for this is that the traditional way of working is not very well articulated in terms of relationships between managed items. The development model prescribes a number of documents to be written at certain milestones. This leads to a fragmented way of apprehending the development process, which is quite different from the holistic understanding based on the conceptual model. Once established though, this model has turned out to be a very powerful tool for deploying the new knowledge to persons that are not directly involved in the elaboration of the model. A comment often heard was “that’s obvious, we are already working like that”. Thus, it is clear that the conceptual model has been a good vehicle for articulating tacit knowledge about relationships between managed items.

Another reason for the many revisions is that the support system turns the traditional way of working around. For example, requirements have traditionally been documented in requirement specifications that state the individual requirements. Thus, the document is the managed item. Now, each requirement is a managed item, and the information previously contained in the document is generated from the support system. This means that the traditional role of the requirement specification disappears, which may be difficult to endorse.

Our experiences indicate that it is very hard to specify in advance the conceptual model and the implementation of the support. These have to be developed iteratively, which puts very strict constraints on the support system. It must be possible to implement changes in the model immediately, otherwise the iterative way of working will not work. On the other hand, this approach produces simultaneously individual and shared knowledge as well as objectified knowledge in the form of models and management support.

The strategy and the management support are one possible way to solve most of the problems concerning management stated in chapter 3. For example, a managed item is characterised by at least a type, an identity, a revision, a state and a set of attributes. The reflection models show what items are managed, what the relationships between them are, what status they will have and in what order they are processed. They also provide guidelines and rules for management. We have not discussed the allocation of responsibility, but it is quite straightforward to do this based on the reflection models.

The strategy for adapting enterprise management support to local needs discussed in chapter 4.3 has so far not been tried out in practice. The reason for this is that only one local design uses the management support so far. Also, this strategy needs a supporting unit at the enterprise level, which is not present at Ericsson at the moment.

The reflection models shown in figures 11, 12 and 15, together with the corresponding support, represent a continuous articulation and evolution of the Ericsson enterprise. So far, we have concentrated on the development context, but in principle, it is quite possible to extend the scope to the entire enterprise and include for example production and sales categories. The reflection models and the corresponding support sys-

---

7. We have estimated that the model was revised more that five hundred times from the outset to its present form.

tem are excellent vehicles for discussing and specifying the roles for different information management systems like Product Data Management (PDM) systems, Enterprise Resource Planning (ERP) systems or Local Design Management systems. Whether this can be done in practise remains to be seen. The huge pressure of companies today to react according to changed market needs, stake holder interests and new technologies makes it very hard to sustain new, stable social constructions within large, global organisations like Ericsson.

7 References

Berger, P. and Luckmann, T (1966): *The Social Construction of Reality*, Reprint by Penguin Books, London 1991.

Goldkuhl, G. (1993): *Aktörer i samarbete - aktörsteori som grund för förståelse av förändringsarbete,* (in Swedish), VITS memo-93-02, IDA Linköpings Universitet.

Humphrey, W. and Kellner, M. (1989): *Software Process Modelling: Principles of Entity Process Models*. SEI-89-TR-002.

Hyusman, M. and van der Blonk, H. (1998): *Supporting and Frustrating Organizational Learning: Exploring the Role of Information Systems in Processes of Organizational Learning*. Proc. 31st Annual Hawaii International Conference on System Sciences.

Kosik, K. (1978): *Det Konkretas Dialektik,* (in Swedish), Röda Bokförlaget AB, Göteborg.

Molander, B. (1996): *Kunskap i Handling,* (in Swedish), Daidalos, Göteborg.

Nilsson, G. (1976): *Oordning, ordning - Studier i kärlekens villkor*, (in Swedish), Korpen, Göteborg.

Polyani, M. (1966): *The Tacit Dimension,* Reprint by Peter Smith, Gloucester, Mass. 1983

Rolf, B. (1995): *Profession, tradition och tyst kunskap,* (in Swedish), Nora: Nya Doxa.

Taxén, L (1995) *The Dialectical Approach to System Design*, Proceedings of Integrated Design and Process Technology, Austin, Texas.

Taxen, L. and Karlsson, E. A (1998): *Incremental Development for AXE 10,* Frankfurt, Ericsson Conference on Software Engineering.

von Uexküll, J: *Umvelt und Innenwelt der Tiere*, Berlin 1909.

Volosinov, V.N. (1986): *Marxism and the Language of Philosophy*, Harvard University Press, London.