Enterprise System Implementation Using the System Anatomy

Lars Taxén

Linköping University, Rundan 91, SE 14645 Tullinge, Sweden
lars.taxen@telia.com

Abstract. The purpose of this contribution is to outline an Enterprise System (ES) implementation approach based on the *system anatomy* and the *Activity Domain Theory*. The system anatomy is a simple image showing how capabilities in a system depend on each other. One central constructs in the Activity Domain Theory is the *activity domain*, which frames a social unit providing some capability that the organization needs. By modelling the organization as an anatomy of dependent activity domains, and the ES as an anatomy of dependent ES capabilities, the approach focuses on the one, single main issue when dealing with complexity – to understand how tings depend on each other.

1. Introduction

It is by now commonly agreed that fundamental issues related to Enterprise Systems (ES) can only be solved by placing the ESs in a wider context that brings individual knowledge, sense-making and technology into a coherent whole. Focussing on the IT-technology only will inevitably yield insufficient results.

ESs are meant to provide relevant capabilities for human actors to achieve something useful in an organization. This means that it is necessary to ground theoretical foundations in human action and coordination of actions. The purpose of this contribution is to outline an ES implementation approach based on the *Activity Domain Theory* (ADT; [1]), and the *system anatomy* construct.

One central constructs in ADT is the *activity domain*, which frames a social unit providing a capability that the organization needs. Another key construct in ADT is the activity modalities - *contextualization*, *spatialization*, *temporalization*, *stabilization*, and *transition* between contexts. These modalities represent innate predispositions for coordinating actions [2].

The system anatomy is a simple image visualizing how various capabilities in a system depend on each other. The anatomy was conceived in the early 1990s at Ericsson[1] as a means to coordinate projects developing extremely complex telecom systems [3].

[1] Ericsson is a well-known leading supplier of telecommunication equipments worldwide: http://www.ericsson.com/

The paper is structured as follows. First, a brief account of the ADT is given. Next, the system anatomy is described. This is followed by an outline of the ES implementation approach. Finally, some conclusions are drawn.

2. The Activity Domain Theory

The activity domain may be illustrated by the mammoth hunt scenery in **Fig. 1**.

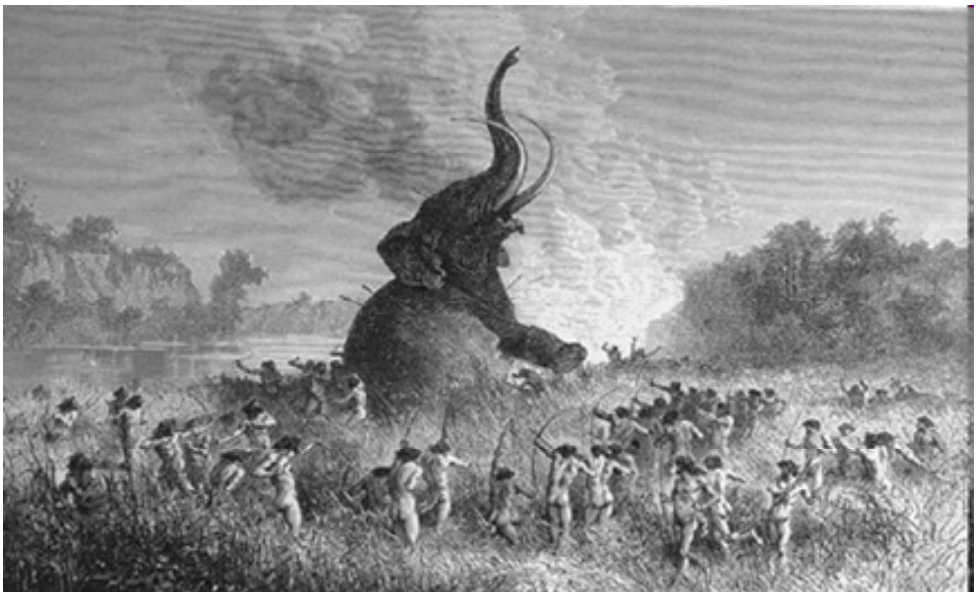

Fig. 1. Illustration of an activity domain ([3], Original wood engraving by E Bayard).

When looking at this scenery some things immediately come to mind. The mammoth is clearly the *object* in focus for actions. According to the Russian theory of Activity, actions are always directed towards some tangible or intangible object [4]. There are also several perceivable *motives* for the hunt: the primary one presumably to get food. Related motives may be to get material for clothing, making arrowheads, and the like. Together, the object and the motive form a centre of gravity around which everything else revolves: hunters, bows, arrows, actions, shouts, gestures, and so on.

In order for hunters to coordinate their actions, certain capabilities are needed. To begin with, there must be a common understanding about the context around the mammoth. This context frames the relevance of individual actions. For example, it can be seen in the background of the illustration that some hunters, the beaters, have started a fire and make noises to scare the quarry away. The mammoth escapes in a direction where other hunters wait to circumvent the quarry and kill it. However, it is only in the light of the activity domain as a whole that the beaters' actions of scaring the quarry away make sense.

Second, a common sense of what things are relevant in the context must be developed. This enables the actors to orient themselves in the same way as a map does. For example, the river is probably relevant since it is hinders the mammoth to escape in that direction. On the other hand, the fishes in the river are certainly irrelevant in this activity domain (they are of course relevant in a fishing activity domain).

Third, individual actions must be carried out in a certain order. For example, the hunters must be in place before the beaters start making noises, the archers may shoot their arrows at a certain command, and so on.

Fourth, the archers cannot shoot their arrows in any way they like. If shooting in a wrong direction, other hunters may be hit rather than the mammoth. Gradually, after many successful (and less successful) mammoth hunts, a common understanding about how to perform appropriate mammoth hunting will evolve. This provides a common sense of the "taking for granted"; rules and norms indicating proper patterns of action that need not be questioned as long as they work.

Fifth, activity domains are not isolated. The brought-down quarry will be cut into pieces and prepared to eat. This is done in a cooking activity, which in turn has its particular motive (to still hunger) and object (which happens to be the same as for the hunting activity: the mammoth). Other related activities might be manufacturing weapons and weapon parts from the bones and the tusks of the mammoth. So, when several activity domains interact, certain issues must be resolved in the transition between activities, such as how to share the quarry among hunters and cooks, or decide how many ready-made arrow heads will be returned for a certain amount of food. Thus, there must be a common understanding about how to coordinate different activity domains.

These five dimensions of coordinating actions are called *activity modalities*, and represent inherent predispositions for acting in the world. In fact, it is possible to conceive these modalities as an extension of Kant's a priori forms of conception (space and time) that exist without any appeal to previous experience. The term "activity modalities", is deliberately coined to connote with *sensory modalities* such as vision, hearing, touch, taste, smell, etc. Thus, the way we experience the world through our senses, is transformed by our brains into an activity modality percept that enables acting as individuals and together with others [5].

An inherent part of activity domain is that actions are always *mediated* by tools or means. The hunters make use of bows and arrows, the beaters use some kind of tools to make a fire, the assault of the mammoth is most certainly coordinated by gestures and shouts, and so on. However, these means need to be *enacted*, which is a process by which capabilities of means and humans together become meaningful resources in the domain [6]. The result is that the activity domain frames an *ideology* - that is, a wide-ranging system of beliefs that prescribes what phenomena are considered real and which actions are regarded as valid.

In summary, the activity domain is characterized by the following aspects:

- The actions in the domain are *motivated* be some need, and directed towards an *object*.
- The object and motive impel the formation of a context in which actions make sense (*contextualization*).
- Actions require a spatial comprehension of the context (*spatialization*).

- Actions are carried out in a certain order (*temporalization*).
- Actions require rules, norms, etc., that signify which actions are valid in the domain (*stabilization*).
- The formation of activity domains according to different motives and objects brings about a need to coordinate domains (*transition*).
- Actions are *mediated* by activity-relevant means.
- Means need to be *enacted*.

3. The System Anatomy

A striking way of representing the development object in new product development is the *system anatomy*; a frequently used means at Ericsson for visualizing complex telecom systems [2]. The anatomy is an illustration – preferably on one page – that shows *the dependencies between capabilities* in the system from start-up to an operational system. Here, "capability" shall be understood as the ability of a certain system element to provide something that other system elements need. An anatomy for a telecom processor is shown in **Fig. 2**:

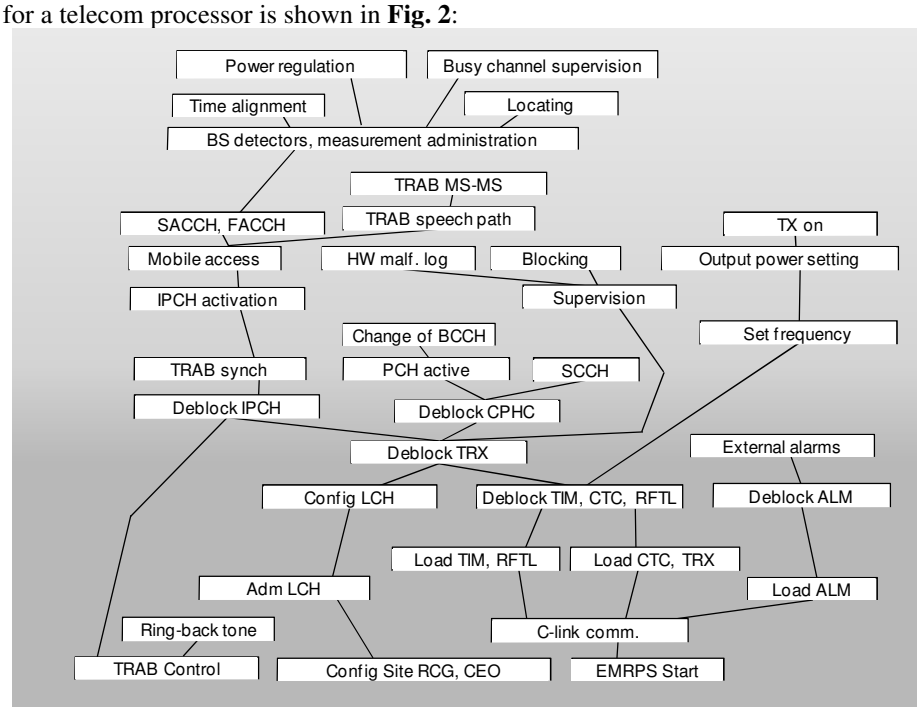

Fig. 2. An illustration of a development object from Ericsson – a telecom processor

A system anatomy is, as the name suggests, a description of a system. Some of the characteristics of the anatomy are as follows:

- *Purpose*: The purpose of the system anatomy is to provide a common understanding among system experts about the system.
- *Motivation*: A common understanding about the system is necessary for coordinating development activities. The system anatomy is simple enough to achieve such an understanding, yet it is powerful enough to show the most important thing when dealing with complex projects – how things depend on each other.
- *System model*: The system anatomy is a model of a finalized system. It describes how we conceive of the system when it has been developed. System should be understood in a wide sense such as products, processes, organizations, organism, or any other arrangement of interest where parts, including humans, interact to form a whole.
- *Visual*: The anatomy is an *image* of related things drawn on one page. Thus, the anatomy is basically *visual* in character, although text can be used to enhance comprehensibility.
- *Capabilities*: The things shown in the system anatomy are *capabilities* in the system. Sometimes these capabilities are referred to as *anatoms* to emphasize the anatomy perspective. The part, module or any other object implementing the capability is not shown in the anatomy
- *Dependencies*: There is an inherent order in the system anatomy signified by the vertical relative positions of the anatoms in the image. The most fundamental capabilities are placed at the bottom of the image. At the top, those capabilities offered to the users of the system (the "money-making" ones) are shown. Thus, the anatomy illustrates dependencies (and independencies) between capabilities.
- *Static*: The system anatomy is at any moment a *static* image; it shows only related things. There is no indication of time in the anatomy; of things changing as time goes along.
- *Social*: The system anatomy is developed by people involved in a development task. This means that the anatomy is a social accomplishment. Thus, given the task of describing a system, two separate groups of people will arrive at different anatomies of the same system (in a particular project, of course, only one anatomy is used). Consequently, the anatomy is not meant to be an exact, formal description of the system. Rather, it is an instrument for achieving common understanding about the essential capabilities in the system and how these depend on each other.

The anatomy can be interpreted as a conceptualization of the object in the activity of developing the telecom processor – a contemporary mammoth hunt. Since most parts of a telecom system is realized by software – which is not physically visible – an easy to apprehend image such as the anatomy is indispensible. The mammoth must come out of the fog, so to speak.

4. An ES Implementation Approach

In this section a brief outline of the ES implementation approach is given. Due to space limitations, the account of the approach can only be rhapsodic.

In principle, the anatomy can be used for visualizing any kind system as dependencies between capabilities. Since the activity domain is conceived as a social unit that provides a certain useful capability, an organization can be modelled as an anatomy where the "anatoms" are activity domains. In **Fig. 3**, one example from Ericsson is shown that illustrates the dependencies between activity domains in the "top-level" Ericsson activity domain. This means that the Ericsson organization is regarded as an activity domain in itself.

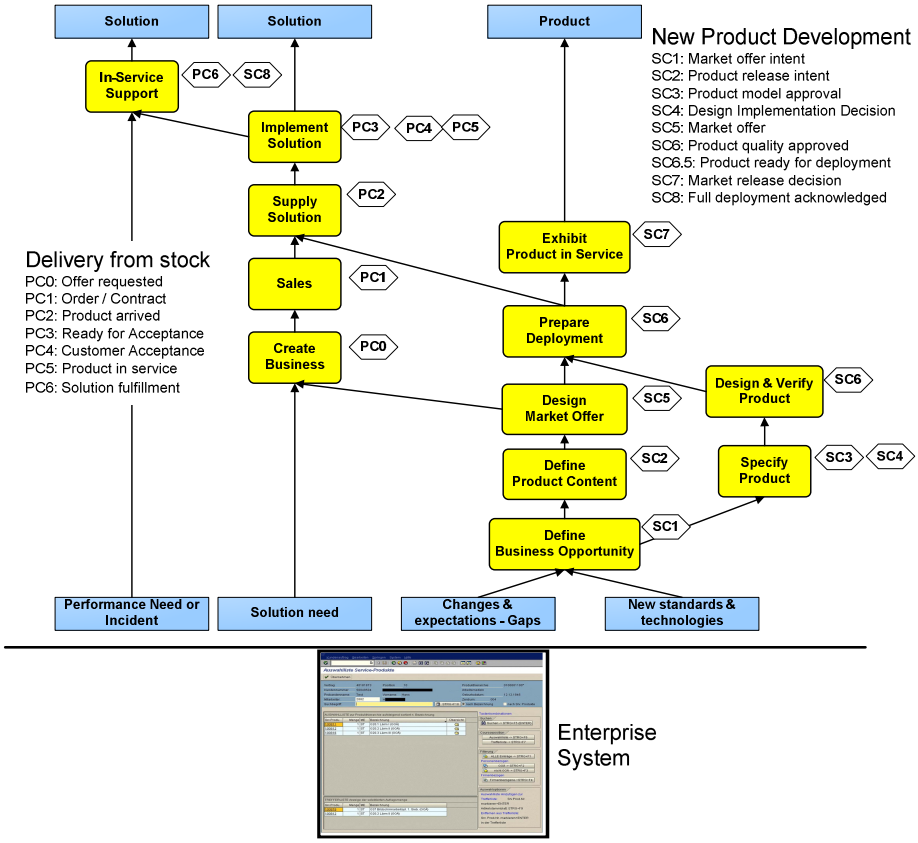

Fig. 3. The organizational anatomy

In addition to the activity domains, the Enterprise Systems (ES) is seen as a means providing information management capabilities to all activity domains that need such a capability. However, the ES itself can also be modeled as an anatomy; that is, we "open up" the ES and make an inquiry into what ES capabilities are needed to implement the ES according to the needs of the activity domains. Thus, by representing both the organization and the ES as anatomies, an approach for aligning business needs to ES capabilities is achieved. From this, the implementation of the ES can be outlined as follows.

4.1. Define the anatomy of the organization

The result is an anatomy of dependent activity domains as in **Fig. 3**.

4.2. Do an information analysis

For each domain, information elements (IEs) are defined, i.e., the information that is worked on and relevant in the domain. This analysis can be advantageously done with the help of so called Information Interaction Models [1]:

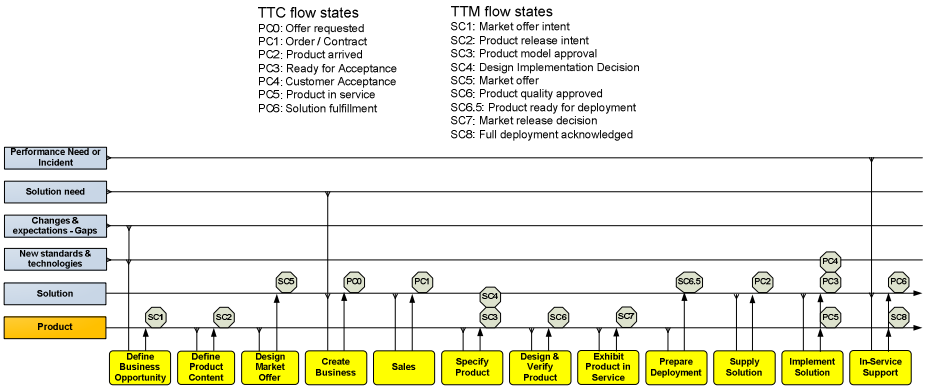

Fig. 4. Focusing on the information elements in the activity domain

4.3. Model the ES as an anatomy of dependent ES capabilities.

In **Fig. 5** an example of an ES anatomy is given:

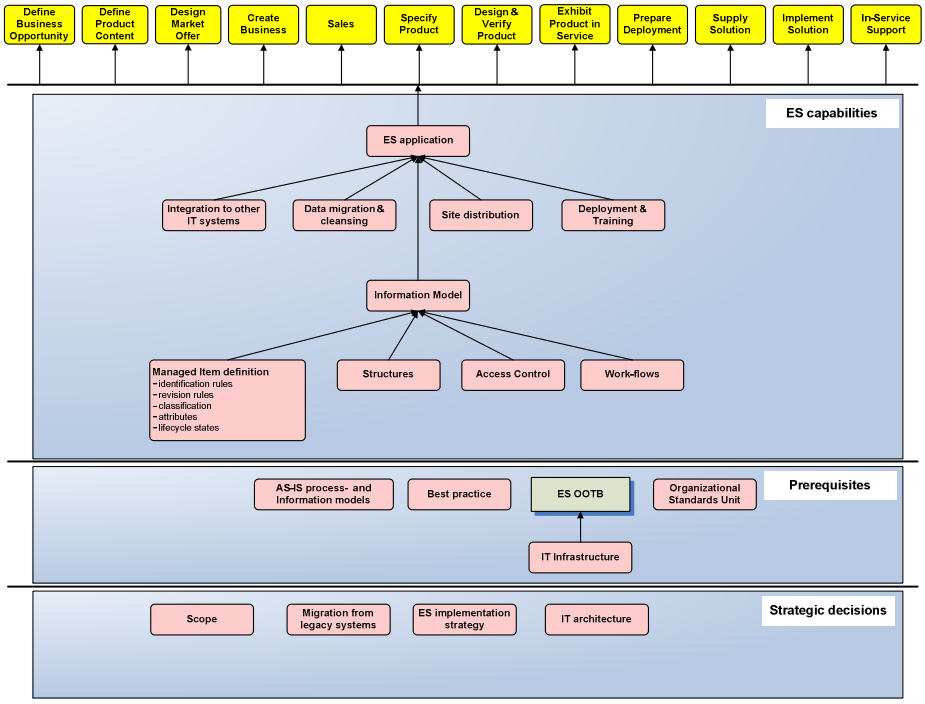

Fig. 5. An anatomy for the Enterprise System

Basically, three groups of capabilities can be identified: strategic decisions, prerequisites and ES capabilities. Strategic capabilities may be the following:

- *Scope*: The scope of the ES needs to be defined in terms of what activity domains should be supported by the ES.
- *Migration from legacy systems*: This concerns how the migration from legacy systems to the ES shall take place.
- *ES implementation strategy*: A decision about the implementation method needs to be taken (agile or traditional)
- *IT architecture*: There is a need to position the ES in the IT landscape; existing or future. This also concerns which legacy systems shall be replaced by the ES.

Some prerequisite capabilities are as follows:

- *AS-IS process- and Information models*: The existing main process- and information models may need to be investigated in order to provide a stepping stone for the ES implementation.
- *Best practice*: There is a need to know what factors alleviate and aggravate the implementation.
- *ES OOTB (Out of the box)*: The organizational-invariant ES platform supplied by the ES vendor.

- *IT Infrastructure*: The computers, network, maintenance, support, etc., needed to run the ES OOTB system efficiently in all activity domains, regardless of where these are physically located.
- *Organizational Standards Unit*: There is a need for some unit (activity domain) that is responsible for the definition and maintenance of mandatory organizational rules, standards, norms, etc. that applies all over the scope of the ES.

At least the following ES capabilities are needed:

- *Managed items' definition*: The items to be managed in the ES must be defined. This definition will include, but are not limited to, item identification rules, item revision rules, classification of items, item attributes, and item lifecycle state sets.
- *Structures*: The main types of structures that managed items can be included in, need to be defined.
- *Access Control*: This capability is necessary for specifying what different actors can to in terms of creating, reading, modifying, and deleting managed items in the ES.
- *Work-flows*: Work-flows for routinized tasks like creating a new item, releasing a product, doing controlled changes, approval of documents, and the like, must be defined.
- *Information Model*: This capability provides a model for what items are relevant in the activity domains, and how these are characterized and related to each other. This model is implemented in the OOTB ES.
- *Integration to other IT systems*: This capability concerns the interaction between the ES and other information systems, i.e. Enterprise Interoperability.
- *Data migration & cleansing*: Before the ESs can be operational, data must be loaded into the ES. In addition, eroded data quality must be restored in the data migration process.
- *Site distribution*: The physical and logical distribution of data must be defined.
- *Deployment & Training*: The ES must be deployed in the organization, and enacted by its users.
- *ES application*: This is the "money-making" capabilities provided to the activity domains – the clients if you like – of the ES.

4.4. Agile implementation of the ES

Since the anatomy shows how the capabilities depend on each other, it is an excellent means for planning and monitoring an ES implementation project. Preferably, the work of implementing the capabilities should be organized in verifiable, small steps in which all impacted stakeholders are involved. Such an agile approach has been demonstrated to be superior to the more traditional "waterfall" approach, which follows a linear path consisting of requirements' specification, analysis, work distribution, module design, integration, and testing (see e.g. [7]).

5. Conclusion

This contribution addresses the following issue in the CFP: “Taking into account the complexity of enterprise environments, which aspects are essential for the support of novel and reusable enterprise practices?”

By combining the framework provided by the Activity Domain Theory, and the System Anatomy construct, an approach has been outlined that singles out the main issue when dealing with complexity – to grasp how things depend on each other. As with any work in progress, the approach needs to be further validated and elaborated in practical settings.

6. References

1. Taxén, L.: Using Activity Domain Theory for Managing Complex Systems. Information Science Reference. Hershey PA: Information Science Reference (IGI Global). ISBN: 978-1-60566-192-6, (2009).
2. Taxén, L (Ed.).: The System Anatomy – Enabling Agile Project Management”. Lund: Studentlitteratur, (n.d.)
3. Bryant, W. C., Gay, S. H. : A Popular History of the United States. Vol. I, New York: Charles Scribner’s Sons, (1883)
4. Kaptelinin, V., Nardi, B.: Acting with Technology - Activity Theory and Interaction Design. Cambridge, MA: The MIT Press, (2006)
5. Taxén, L.: Modeling the Intellect from a Coordination Perspective. In B. Igelnik (Ed.), Computational Modeling and Simulation of Intellect: Current State and Future Perspectives. Hershey PA: IGI Global, (n.d.)
6. Weick, K. E.: Enacted sensemaking in crisis situations. Journal of Management Studies, 25(4), 305-317, (1988)
7. Fredriksson, O., Arola, M.: En fallstudie av ett framgångsrikt affärssystembyte. In J. Hedman, F. Nilsson, & A. Westelius (Eds.), Temperaturen på affärssystem i Sverige, 167-196. Lund: Studentlitteratur (in Swedish), (2009).