A Practical Approach for Aligning Business and Enterprise Systems

Lars Taxén

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

Abstract. The purpose of this contribution is to outline a novel approach for aligning business objectives with Enterprise Systems (ESs). The approach is based on two constructs called the *system anatomy* and the *activity domain* respectively. The system anatomy is a simple image showing how capabilities in a system depend on each other. The activity domain – a central construct in the *Activity Domain Theory* – frames a socially organized unit working on a particular task that makes sense in an organizational context. With the activity domain at hand, it is possible to conceive of the organization as a confederation of activity domains. Since both ESs and activity domains provide capabilities that the organization needs, it is possible to model both in the same way – as anatomies showing the dependencies (and independencies) between capabilities. Thus, the approach enables a unified view of the organization and ESs, based on how tings depend on each other – a main issue when dealing with complexity.

1. Introduction

Research concerning Enterprise Systems (ESs) is dragging in at least two respects. First, it is mainly analytical in character. Researchers are making inquiries into ESs after these have been implemented in organizations. Rarely are practical guidelines for implementation of ESs suggested. Second, research is more or less exclusively focused on ERP (Enterprise Resource Planning) systems. However, the information managed in ERP-systems, as well as other organizational wide systems like SCM (Supply Chain Management) and CRM (Customer Relationship Management), is highly dependent on another kind of ESs, the so called PLM (Product Lifecycle Management) systems. These are mainly used to manage the product development process, and are qualitatively different from ERP-systems[1] (see Fig. 1):

[1] For example, ERP systems are transactional systems without in-built features for revising information entities, which, on the other hand, is a key feature of PLM systems.

Fig. 1: The relationship between various organizational-wide ESs [8].

> "PLM processes clearly interact with those supported by other enterprise systems, but the creative activity inherent in PLM distinguishes it from ERP and supply chain management (SCM), which support routine processes only. For example, sourcing in the PLM environment involves suppliers in the early stages of product design to speed the development of prototypes and associated early design processes. By contrast, the sourcing function within SCM takes place only after a product is designed and involves little, if any, strategic input from suppliers regarding a product's content."[8]

In this paper, we want to address both these issues. It is by now commonly agreed that fundamental issues related to ESs can be solved only by considering 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. In order to analyze ESs, and propose practical implementation guidelines, it is necessary to depart from some theoretical framework that includes both technological and human aspects. To this end, the purpose of this contribution is to outline a novel approach for aligning business objectives and Enterprise System (ES), based on two constructs called the *system anatomy* and the *activity domain* respectively.

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[2] as a means to coordinate projects developing extremely complex telecom systems [2]. The use of the anatomy has turned out to be a critical success factor for product development at Ericsson, and is now part and parcel of the standard tool box for project management in this company.

The activity domain – a central construct in the *Activity Domain Theory* (ADT; [1]) – frames a socially organized unit working on a particular task that makes sense in an organizational context. Such units may be, for example, teams, organizational functions, business units, entire organizations, or several organizations in collaboration (sometimes called "the extended enterprise"). Each such unit may be represented as an activity domain. This means that an organization can be regarded as a confederation of mutually dependent activity domains.

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

Since both ESs and activity domains provide capabilities that the organization needs, it is possible to model both in the same way – as an *organizational anatomy*; the essence of which is to show dependencies between capabilities, regardless of whether these are provided by activity domains or ESs. Thus, the approach enables a unified view of the organization and ESs based on how things depend on each other – a main issue when dealing with complexity.

The paper is structured as follows. First, the system anatomy is described. Next, a brief account of the ADT is given. This is followed by an outline of how the business and the ES can be modelled as anatomies. Based on this, some practical guidelines for implementation of the ES are given. Finally, some conclusions are drawn.

2. The System Anatomy

When people are acting together towards a common goal, a prerequisite for success is that actions can be coordinated. This in turn requires that the target towards which actions are directed can be "seen" in some way. A common understanding about the target is necessary. In many cases, this is not a problem. For example, in the mammoth hunt described in the next section, the target is obvious – the mammoth. Also, when working on mechanical products like the milling tool in Fig. 2, the target is clearly visible.

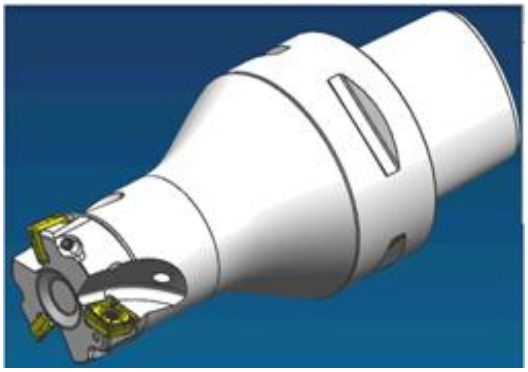

Fig. 2: A milling tool

However, things become more awkward when product complexity increases and functionalities are implemented in software. How do you visualize software in such a way that that a common understanding is reached?

One way of doing so is trough the *system anatomy* [2]. The anatomy is an illustration – preferably on one page – that shows *the dependencies (and independencies) 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. The capabilities can be implemented by hardware, software, or any other means, including humans; this is not important in the anatomy. As an example, an anatomy for a telecom processor is shown in Fig. 3:

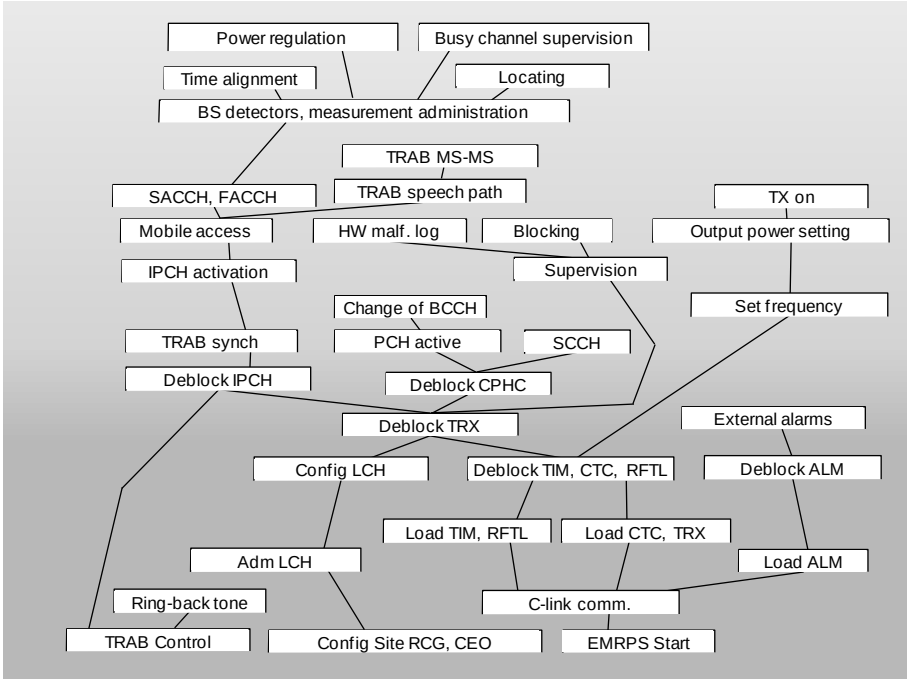

Fig. 3. An illustration of a development object from Ericsson – a telecom processor

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, for example the anatom "EMPRS Start" in Fig. 3. If this capability fails for some reason, the whole system fails. 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.

3. The Activity Domain

The activity domain may be illustrated by the mammoth hunt scenery in Fig. 3.

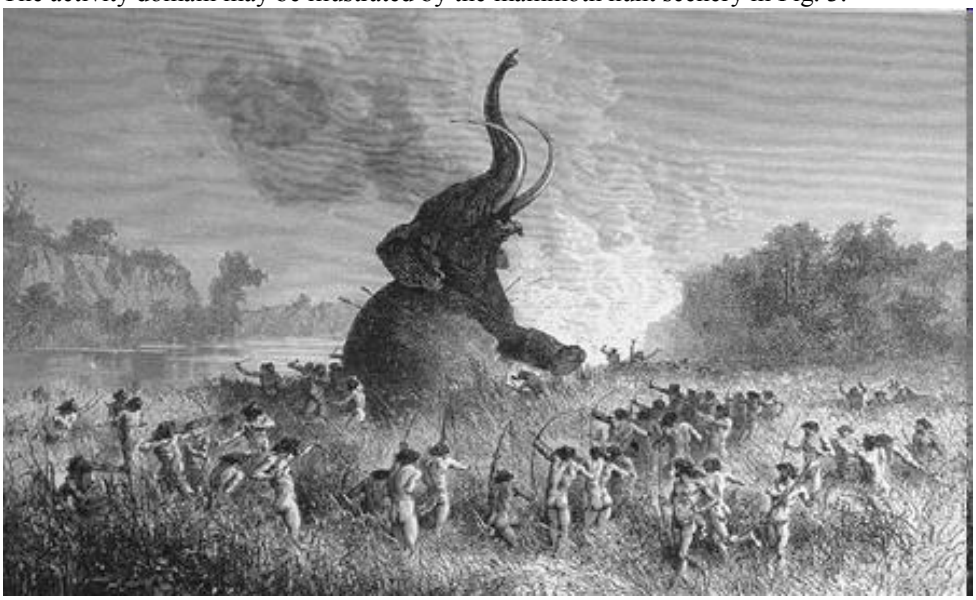

Fig. 4. Illustration of an activity domain ([3], Original wood engraving by E Bayard).

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.

Altogether, 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*.

This means that the activity domain can be considered as a social unit that provides a capability for acting, which may or may not be relevant in a certain social context like an organization.

4. Aligning business and ES capabilities

In principle, any kind system can be visualized as an anatomy. 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. 5, 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.

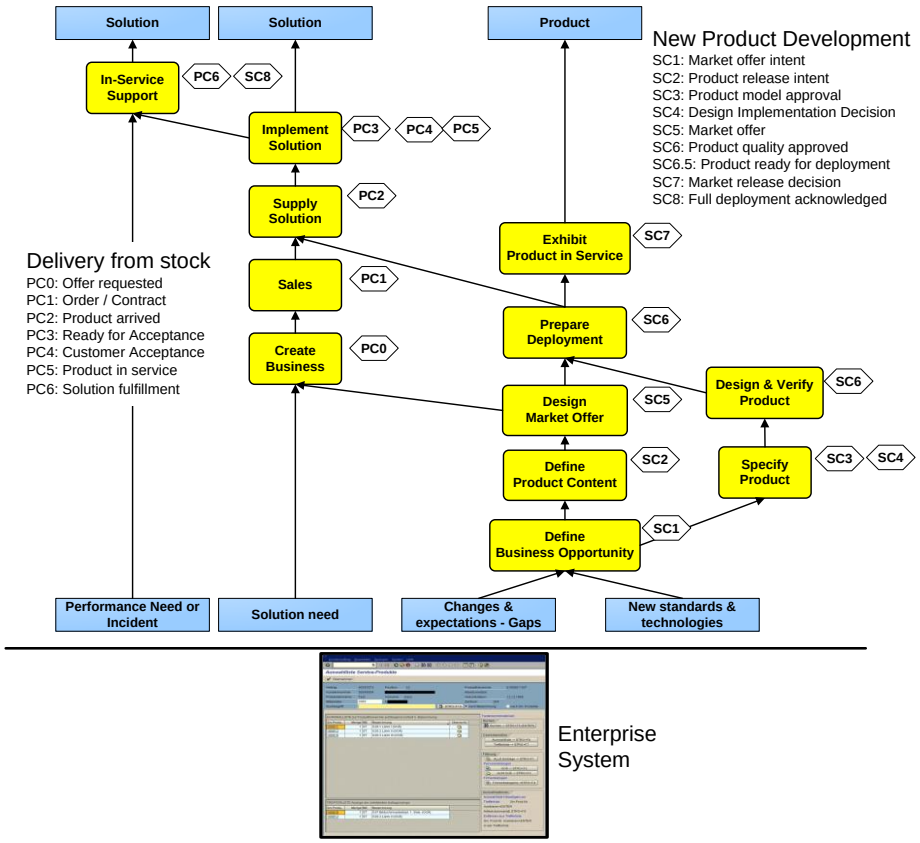

Fig. 5. The organizational anatomy

In addition, the Enterprise Systems (ES) can be seen as a means providing information management capabilities to all activity domains that need such a capability. However, the ES itself can also be modelled 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 in a common way, an approach for aligning business needs to ES capabilities is achieved.

5. Practical implementation guidelines

Based on the view of organizations and ESs presented above, some guidelines for implementing ESs in the organization can be given as follows.

5.1. Define the anatomy of the organization

First, the organizational anatomy needs to be defined in terms of activity domains and their dependencies. Activity domains are determined by the object they work on and the motive for their existence in the organization, meaning that this should be the point of departure for identifying activity domains. Usually, an overall business process model exists in the organization, which may be a good starting point. The result of this activity may be an anatomy of dependent activity domains as in Fig. 5[3].

With the anatomy at hand, several strategic business oriented targets can be managed such as [9]:

- *The constellation of activity domains*: This target concerns the constellation of activity domains involved in producing the outcome of the organization. The main issue is to decide which domains are needed to fulfil the strategic intentions of the organization, and how these domains depend on each other.
- *Domain responsibility*: In most organizations, the responsibilities for the business processes, the core information architecture, the main ESs used, and corporate standards reside in different organizational units. For example, "process owners" are often appointed as responsible for processes. With the activity domain in mind, such a differentiation of responsibilities implies a high risk of neglecting vital interdependencies between these elements. The activity domain approach suggests quite a different responsibility structure: managers should be assigned in line with the constellation of activity domains.
- *Business level coordination*: Another conceivable target for alignment is the coordination of activity domains. The prime strategic issue here is the dependencies between domains at the top-level of the organization. In particular, points of transition between the activities must be considered.
- *Central versus local control*: The dispersed nature of knowledge emerging in different activity domains directly brings another strategic issue to the fore. Each domain enacts a unique worldview; a certain way to conceive reality. However, the coordination of activity domains calls for some common understanding across domains. This implies that the organization has to balance two opposite forces: the drive for excessive commonality and the emergence of detached, incompatible islands of work. Thus, the business strategy should outline how to maintain an optimal balance between central and local control.
- *Core capabilities*: Commonality implies that certain capabilities are valid throughout the organization for all domains. The strategic aspect here is to identify these fundamental capabilities and provide the necessary means to uphold these.

[3] The actual form of the organizational anatomy in Fig. 5 is derived from the overall business process existing at Ericsson around year 2000.

5.2. Do an information analysis

For each domain, information elements (IEs) are defined; indicating what information the organizations wants to manage. This analysis can be advantageously done with the help of so called Information Interaction Models [1]:

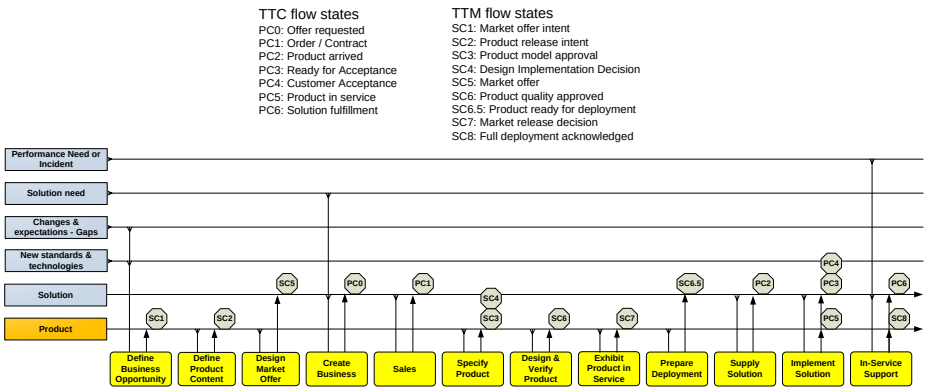

Fig. 6. Focusing on the information elements in the activity domain

This model should be read as follows. Each horizontal line represents an IE. This could be, for example, product information, requirements, orders, customers, and the like. The down-ward directed arrows indicate input to an activity, and the up-ward directed arrows output from the activity. Each output changes the status of one or several IEs. In this way, it is straightforward to follow the progression of each entity managed in the ES.

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

The information interaction models collected in the previous activity is a specification for what capabilities an ES need to have. In order to define this, an anatomy for the ES needs to be defined. In Fig. 7 an example of an such an anatomy is given:

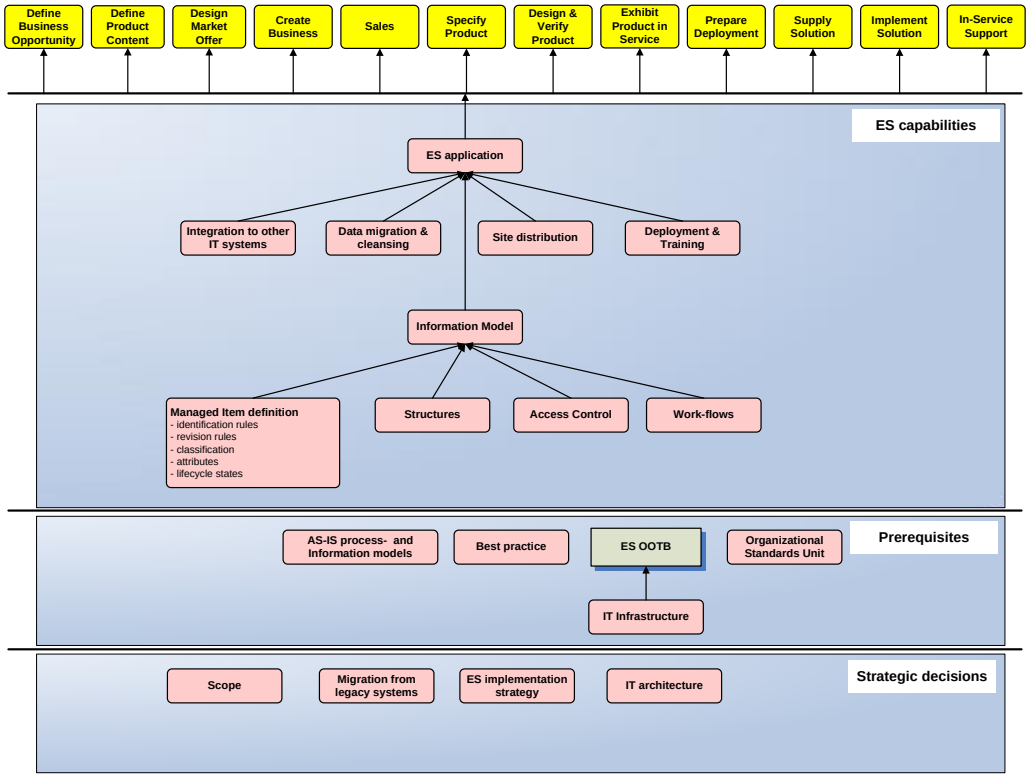

Fig. 7. An anatomy for the Enterprise System

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:

- *Definition of IEs*: The IEs to be managed in the ES must be defined. This definition will include, but are not limited to, identification rules, revision rules, classification of entities, attributes, and entity lifecycle state sets.
- *Structures*: The main types of structures that entities 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 entities in the ES.
- *Work-flows*: Work-flows for routinized tasks like creating a new entities, releasing a product, doing controlled changes, approval of documents, and the like, must be defined.
- *Information Model*: This capability provides a model for what entities 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. interoperability issues.
- *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.

5.4. 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]).

6. Conclusion

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. The approach has been used in practice in some organizations. Hitherto, the results are promising, but as with any work in progress, the approach needs to be further validated and elaborated.

7. 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).
8. MatrixOne: retrieved March 29, 2011, from http://www.scribd.com/doc/48470871/Managing-the-Complete-Product-Lifecycle
9. Taxén, L.: A Practical Approach for Aligning Business and Knowledge Strategies. In M. Russ (Ed.), Knowledge Management Strategies for Business Development (pp. 277-308), Hershey PA: Business Science Reference (IGI Global). ISBN: 978-1-60566-348-7. (2009).