On the Integration of Project Planning, System Anatomy, and System Architecture

Lars Taxén and Peter Olow

Linköping University Post Print

N.B.: When citing this work, cite the original article.

Original Publication:

Postprint available at: Linköping University Electronic Press
http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-111337
On the Integration of Project Planning, System Anatomy, and System Architecture

Abstract
In this paper, we propose to conceptualize projects as an integration of three views: the planning, the system anatomy, and the architectural ones. The system anatomy is the core in the Integration Driven Development approach, which has been successfully employed in a number of complex projects in the telecom industry. An illustration showing the integrated view is proposed as an instrument for managing the project. We discuss what effects on planning and risk management can be expected from applying this method, and analyze our results from a theoretical perspective called the Activity Domain Theory. Our findings are two: an amelioration of the Integration Driven Development approach with architectural aspects, and a positioning of the Activity Domain Theory with respect to the relevance versus rigor discourse in the project management discipline.

INTRODUCTION
The project management (PM) discipline has accumulated a wealth of practical and theoretical knowledge during at least 60 years (Turner et al., 2013). No doubt, this has resulted in many projects successfully delivered on time and within budget limits. On the other hand, a number of projects have turned into spectacular failures (e.g., Dalcher & Tully, 2002; Dalcher, 2003; Nelson, 2007). Nelson amply summarizes the state of play in PM thus:

If failure teaches more than success, and if we are to believe the frequently quoted statistic that two out of three IT projects fail, then the IT profession must be developing an army of brilliant project managers. Yet, although some project managers are undoubtedly learning from experience, the failure rate does not seem to be decreasing (Nelson, 2007, p. 67).

When contemplating over this unfortunate situation, it may be worthwhile to recollect the works of Alan Turing. One of his findings is that for a certain type of problems, we need to first find a solution in order to know if there is a solution. A similar situation can be found in PM. As we cannot control all parameters in planning, we can never guarantee the success, or even the feasibility of the project. If we want to be sure of the cost and risks of a project, the only way is to “do it once before we actually do it”. Since this is not an option, we need to allow for some insecurity and learn how to cope with it.

Experience proves again and again that one of the hardest parts in PM is to know when we have reached the maximum level of precision in planning and estimation. The typical solution is to carry on the quest for more and more details and precision. Often this requires the project planning to involve people that, according to the planning process, should not be working with planning and estimations. Sometimes we even try to build parts of the system in order to have enough information for plans and estimates; a process which is both costly and time consuming.
In situations like these, there is a need to find out the nature of the project early, and include more and more precise information as time moves on. One possibility is to make use of the Cynefin framework (e.g. Kurtz & Snowden, 2003). In this framework, we find a classification of different types of domains/systems, and suggestions on how we may address problems in these domains (see Figure 1):

![Figure 1: The Cynefin classification framework](image)

**Simple (or known)** domains are such that their behaviour can be predicted and described. Cause-and-effect relationships are evident to everyone. Established and best practice methods can be advantageously applied to this class of domains.

**Complicated** domains are characterized by cause-and-effect relationships that can be discovered but are not immediately apparent to everyone, for example by the sheer size of the system such as aircraft consisting of thousands of components; each however definable and knowable. More than one solution to requirement specifications is possible and expert participation and diagnosis is required.

**Complex** domains have many interacting agents, possibly humans, and several different solutions may emerge during the evolution of the domain. There is a need for creative and innovative approaches, as well as and discovering and stabilizing desirable patterns on the way. Best and even good practices are of little help in managing the development in these kinds of domains.

**Chaotic** domains are highly turbulent, and there is no clear cause-and-effect. There is no point in looking for “right” answers, and there is little time for reflections. Massive change is imminent, and taken-for-granted “truths” need to be questioned and reconsidered.

If we believe that a certain project resides in the complicated domain, an intention to use “good practice” solutions may be wise. However, in modern systems development we have an increasingly larger part of software in our products. There are ample indications that any large, embedded software system belongs to the complex domain. This limits our possibilities to create order based on information that we collect in advance. Instead, we need to rely on patterns that we have seen in similar situations before.

With this as a background, the scope of this paper is to elaborate on a practice-based pattern for managing system development projects in the telecommunication industry. Telecom systems usually consist of a networked infrastructure and applications developed on that infrastructure. A typical system is made up from hundreds of components in a wide range of technologies: hardware, software, radio, optics, etc. Requirements on real-time responses and up-time are very strict, and new systems must
play together with legacy systems developed many years ago. In addition, the market for applications is turbulent with many competing operators that often change their requirements in the midst of the project. Clearly, such projects are paradigmatic examples of the complex Cynefin domain.

In the early 1990s, Ericsson, a well-known supplier of telecom systems worldwide, was involved in a technology shift from the 1st to the 2nd generation of mobile communication systems. The first projects faced new technologies, challenging requirements, very demanding customers; a virtual travel into then unthreaded landscape. It soon became painstakingly clear that extant PM principles could not cope with the extent of complexity faced in these projects. In an attempt to break with established thinking, some ingenious persons realized that the most important thing when dealing with complex systems is to manage dependencies. If we lose control of dependencies, we are lost. The expression of this insight was a simple illustration of the dependencies between capabilities in the system; an illustration which later became known as the system anatomy. Based on the anatomy, an entire approach towards managing complex development tasks was elaborated. This approach, called integration driven development (IDD), is, after some 20 years, an established project management technique at Ericsson, and now gradually spreading to other companies (Jönsson, 2006; Berggren, Järkvik, & Söderlund, 2008; Taxén & Lilliesköld, 2008; Taxén & Petterson, 2010, Taxén, 2011).

Rationale
Although IDD has proven its merits in many projects, it needs to be elaborated in certain aspects. Since the anatomy is an image of dependencies between capabilities, it is mainly a functional description. The system parts that implement the capabilities are subdued for reasons of simplicity and ease of communication. This means that the role of the system architecture, i.e. how functions are implemented in system parts, has to a certain extent been neglected in the early phases of IDD. While the system anatomy focuses on functional capabilities (“What can I do with this product?”), the system architecture is needed to capture non-functional characteristics such as performance, weight, power consumption, cost, etc. The system architecture also captures the inherent complexity of the selected solutions, leading to a better understanding of the required effort and skills for the development. Neglecting the system architecture may result in unacceptable uncertainties in project estimates, caused by a lack of understanding the design and its implementation.

Aim and research question
Thus, we propose that management of complex system development tasks requires control of three different views of the project:

- The traditional project planning view, the rationale of which is to warrant the delivery of the desired system on time and within budget.
- The system anatomy view, the rationale of which is to warrant the requested system functionality by managing the dependencies between system capabilities.
- The system architecture view, the rationale of which is to warrant the compliance with non-functional requirements and other constraints.

Effective PM requires the integration of these three views; something which is particularly important in the early phases when we know next to nothing. Thus, the research question of our contribution can be formulated as follows: “How can the system
architecture view be included in IDD, and what effects can be expected from integrating the planning, anatomy and architectural views?”

Structure of the paper
The paper begins with an outline of the Activity Domain Theory (ADT: Taxén, 2009), the interpretative framework by which will analyze our results. One reason for this is that we want to be explicit about the ontological and epistemological base of the paper. We maintain that a pragmatic or praxis based approach is needed, where the dialectics between rigour and relevance need to be primarily grounded in practical experiences (Cicmil et al., 2006; Winter et al., 2006; Smyth & Morris, 2007; Blomquist et al., 2010; Lalonde et al., 2010).

Next, we sketch the research approach. Basically, our contribution is a conceptual one, and we rely on previous contributions for the empirical grounding of our results. We proceed with outlining the main planning steps in working with the integrated view. In the discussion section, we consider how this view may build a solid foundation for managing complex projects, and what effects can be expected from this. Further, we analyze our results using ADT as an interpretative framework, and position this interpretation with respect to the ongoing discourse about relevance versus rigour. Limitations and suggestions for further research are given. Our main conclusion is that adding an architectural view further ameliorates the IDD approach for managing complex projects.

THEORETICAL FRAMING – THE ACTIVITY DOMAIN THEORY
The theoretical framework used to guide the interpretation of the results is the Activity Domain Theory (ADT; Taxén, 2009). The paradigmatic base from which ADT emerged, is praxis; defined as nexus where humankind creates him or herself by changing nature and by producing the world of cultural objects; ideas that originally emanated from the early Marx and subsequently became elaborated into what has become known as Activity Theory (e.g. Kaptelinin & Nardi, 2006). ADT is an elaboration of these ideas. In order to explain the main features of ADT, the hunting scene in Figure 2 may be employed:
When looking at this scenery some salient features can be noticed. First, the mammoth is clearly the object in focus for the activity. The activity is about something. This is the most fundamental characteristic of any activity. The activity is also motivated by some need; the primary one presumably to get food. Related motives may be to acquire material for clothing, making arrowheads, and the like.

Second, the object and the motive form a centre of gravity around which a context of relevance emerges: hunters, bows, arrows, actions, shouts, gestures, and so on. 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 prey away. The mammoth escapes in a direction where other hunters wait to circumvent the prey and kill it. It is only in the light of the activity as a whole that the beaters’ actions of scaring the prey away make sense.

Third, a 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 no doubt relevant since it obstructs the mammoth to escape in that direction. On the other hand, the fishes in the river are irrelevant in this activity (but they are certainly relevant in a fishing activity).

Fourth, actions must be carried out in a certain order. For example, shooting an arrow involves the steps of grasping the arrow, placing it on the bow, stretching the bow, aiming at the target, and releasing the arrow.

Fifth, the archers cannot shoot arrows at random. If shooting in a wrong direction, other hunters may be hit rather than the mammoth. An understanding of how to perform appropriate mammoth hunting will be acquired after many successful (and, presumably, some less successful) mammoth hunts. This provides a sense of the “taking for granted”; norms, routines, traditions, and rules indicating proper patterns of action that need not be
questioned as long as they work: “This is the way we do things around here!”

Sixth, an activity is not an isolated island; typically, activities are dependent on each other. For example, the prey will most likely 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. However, in this activity, other aspects of the mammoth become relevant (as, for example what parts of the mammoth are edible), which means that the context will determine how the object is conceptualized. Other related activities might be manufacturing weapons and weapon parts from the bones and the tusks of the mammoth. So, when several activities interact, certain issues must be resolved in the transition between them, such as how to share the prey among hunters and cooks, or decide how many ready-made arrow heads will be returned for a certain amount of food.

The six dimensions outlined above – objectivation, contextualization, spatialization, temporalization, stabilization, and transition between contexts – are denoted activity modalities in ADT. As stated, they represent inherent, biological predispositions for acting in the world, which humans (and possibly all organisms equipped with a neural system) have developed during their phylogenetic evolution. The modalities are all part of an integrated whole, meaning that they are interdependent and mutually constitute each other. If one of these modalities fails for some reason, we cannot act. So, for example, a person with a progressive deterioration in the superior colliculus and/or surrounding areas of the brain show a deficit in the ability to shift attention, which means that the transition modality is inhibited (Posner & Petersen, 1990).

Activities conceptualized as above are denoted activity domains in Taxén (2009). An inherent part of activity domains is that they are always mediated by 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. In order to be useful, these means need to be enacted; a process in which humans and means together make up meaningful resources in the domain; they “become one” so to say. The enactment process is framed by the activity modalities. So, for example, learning how to shoot an arrow in a hunting context requires that you can recognize the prey (objectivation), form a context around it (contextualization), validate what things are relevant (spatialization), and shoot the arrow in an efficient way (temporalization and stabilization).

In summary, the epistemological base of ADT, i.e., how we know what we know, may be characterized as pragmatic, focusing on useful knowledge gained in action. The end result of enaction is a domain-specific ideology in the sense that certain beliefs develop about what phenomena are taken for “real”, and which actions are regarded as valid. In this way, the activity domain can be regarded as the nexus of human activity, which means that activity cannot meaningfully be further decomposed into yet more fundamental elements.

**RESEARCH APPROACH**

The empirical basis of this paper consists of multitude of projects at the Ericsson and other companies using some version of the IDD approach (e.g. Adler, 1999; Taxén, 2003; Berggren, Järkvik, & Söderlund, 2008; Taxén, 2009; Taxén, 2011). Details about research approaches can be found in these contributions. Usually, some variant of qualitative methods has been employed. For example, a thorough analysis of the ADT and the IDD approach using a grounded theory method (Strauss & Corbin, 1998) can be found in Taxén (2003).
Our contribution can be seen as a continuation of this line of research, adding yet another facet to the interplay between IDD and ADT. No particular research methodology has been employed. Rather, our contribution should be seen as a conceptual one, aiming at further elaborating the IDD approach and possibly corroborating the constructs of ADT. Since one of the authors of this paper, Olow, has a long professional experience in coaching IDD projects, and the other, Taxén, has been active in both the telecom industry and at academia, the paper can be seen as an example of coproduction of knowledge from both practice and theory (Berggren et al., 2008). Hence, our contribution is an attempt to maintain and elaborate a particular strand of research in project management, which has tried to balance rigour and research for many years.

AN INTEGRATED APPROACH
Our point of departure is that integrating information from different views gives a better precision and visibility in the planning of the project. From experience we know that there are a number of factors that increases the level of complexity; size is one and integration of large software systems is another. To illustrate this, we use a fictitious example from development of a distributed control system, where project increments/work packages mostly consists of integrated software parts.

The anatomy view
The system anatomy shows the functional growth of the system, and how this functionality can be integrated from basic functions to the delivered end user features. The main elements are “anatoms” and “dependencies”. An anatom is the representation of a functionality that is meaningful to the system and can be integrated and tested in a reasonable time. The dependencies are single-directional links between anatoms, showing in which order they shall be developed and integrated. An example of an anatomy is shown in Figure 3:
Typically, an anatomy does not show the allocation on components and how these evolve when adding functionality. The anatomy is usually created at the very beginning of the project, in a workshop – “the anatomy day” – with representatives from all stakeholders and developer disciplines.

The architectural view

The architectural view visualizes the system as system elements and how these interact. This can be expressed in a variety of ways, both functional and component oriented. In most modern SW design methods the architecture can be illustrated as a generic (class) view and an instantiated (object) view. For instance, a layered design is shown as a pattern in the generic view to be implemented later on in the object view.

In many systems, the architecture is a mix of logical components and components derived for an organizational purpose, such as subsystems. The ideal components are those that support efficiency in all aspects of the development project. In our example (see Figure 4) we show a layered design, where the layers represent both the way from generic functions to an end user specific feature, as well as a pattern for how the functionality is deployed over a distributed control system. An important characteristic is the lifecycle of the different components, where the lower layers represent assets that can be used in many applications, and the top layers cover more customer specific features and add-ons.
Gathering information from the architectural view can be made in a similar way as for the anatomy. The basic information is derived from architectural epics given by the strategic decisions in the product planning. They are introduced into the planning by running one or a few workshops in the fashion of the “anatomy day”.

**The project view**
The classic planning views typically contain timing and resource management information. In case there is a plan for incremental deliveries, this information is crucial for the planning of the increments developing components and functionality. The content of those increments is most likely shown in the system anatomy. For a more traditional planning approach, there will be only one or a few planned deliveries, and mostly, the increments have to be made up from information in the other two views. By using the information from the other domains, the need for a delivery is weighed against the capability of the project to deliver. This will eventually be shown in the milestone and delivery plan of the project, for example, as a Gantt chart.

**The integrated view**
The integrated view is based on the anatomy and extended with the available architectural and planning information. This is done by allocating the anatomy in a grid spanned by an incremental project delivery axis (the planning view) and a layered architectural axis (the architectural view) (see Figure 5). Thus, we can illustrate how the system is built in increments, each one giving an additional set of customer features, while still showing how a single feature is distributed over layers in the architectural view.
The layers shall be chosen to, in the best way possible, represent the assets in the evolution of the architecture. In doing so, we use our knowledge of what is required to deliver a specific feature, in terms of increments implemented by components in specific architectural layers.

Typically, the integrated view will be developed gradually. The most significant contribution is the one from the anatomy. As we move on, adding aspects from the other two views, the interaction becomes more important and we start to build the different views on each other. From experience, a plausible way to proceed is as follows:

- Start with the feature deliveries from the cycle plan / project milestones.
- Define the architectural building blocks from our design strategy.
  - Define what good building blocks from a technical perspective are.
  - Perform an analysis of the strategic evolution of the system and make adjustments.
- Build the anatomy in the ordinary way, showing the functional order of integration.
- Map the anatomy to the grid of increments and architecture components.
  - Start to divide the anatoms to fit both the feature and the component evolution.
  - Where the capability and the feature delivery do not match, define the gap.
  - Work on the discrepancy to find the needed information to fill the gap. This can either be to add one or more anatoms, or to change the scope of the existing ones.
  - When drawing the anatoms, derive information of allocation of the functionality to the components. This can, if needed, be shown in an architectural view.
Use the information that we get from the more precise anatoms in the estimation of the effort to develop them.

DISCUSSION
In the previous section we showed how the system architecture view can be included in the IDD approach. In this section, we discuss the remaining part of our research question: “What effects can be expected from integrating the planning, anatomy and architectural views?” We begin by examining strengths and weaknesses of each view in turn, before considering the effects of the integrated view.

How to work with multiple views depends to a large degree of the available information at the start of the planning process. Each view has its own merits and drawbacks, and it is crucial to understand those in order to plan the project in the most beneficial way.

The planning view
The strength of the planning view is derived from the long experience we have with working with this type of information. We have excellent ways to manage the time-line of the project through different types of information; scheduled tasks, allocated resources, effort estimations, tasks distributed to various sub-units in the project, progress monitoring, and the like. Often, this view is illustrated in the form of Gantt-diagrams and work break down structures.

One weakness of large Gantt-charts and other activity-based diagrams such as CPM/PERT, is that they are hard to establish a consensus about. It is difficult for stakeholders to envision what is going on in the project and what the consequences of a delayed or postponed/cancelled activity are. The reason is that the foremost purpose of these charts is planning and monitoring the project, rather than ease of achieving a common understanding of the design and the complexity of the system. Since the architectural and anatomy views are subdued, this might lead to uncertain planning and risk management. To further aggravate matters, the system to be developed – the target of the project – is visible only indirectly, for example, by examining the text descriptions of activities.

The information in the planning view has a high degree of precision, and there is a large diversity of tools to help sort out the details. However, in order to reduce the uncertainty in planning, we need to include information from the other views before detailing the planning. In short, we use the information about milestones and expected customer deliveries to define a good building order and the most suited increments. This information is gathered from the anatomy and architectural views. Thus, we may iterate between the three views in order to improve the planning of the project.

The anatomy view
The anatomy provides stakeholders with an easily comprehensible image that express a common understanding of the project target. It is designed to support communication and coordination, which means that it is quite different in nature as compared with planning diagrams and architectural descriptions. A benefit of the anatomy view is that it highlights focus points for taking decisions, and helps finding integration points.

An additional benefit is that the anatomy can be produced early in the project, and with a limited effort. Because of the simplicity of this view, it is easy to change the picture and to evaluate the consequences of different options. When we get more information on the estimates of single anatoms, it is easy to update the integrated view.
with the necessary changes to keep the project in control.

A weakness of this view is the limitation in complexity given by the requirements on size and visual quality. There is a limit on how much information that can be crammed into this view using only a basic drawing method.

**The architectural view**

The architectural shows how the system is implemented. From a management point of view, the architecture is needed to estimate development effort, assess risks related to module implementation, and validating non-functional requirements. Moreover, many systems – in particular real-time ones – are based on a long-term architecture that is employed over and over again in developing new applications. This means that the architecture constrains – at least in parts - the space of possible solutions, for example, by specifying certain interface protocols that must be adhered to.

The major benefit of the architecture view is that it displays different aspects of the technical solution. This makes it possible to express the inherent complexity without getting lost in details. It is easy to focus on information you need for planning before the start of or early on in the project. The architecture can also be used to cover strategic aspects of the product development, which needs to be considered during the development.

Architectural descriptions are almost never good enough as basis for planning the project; additional views are needed. This is due to its mediocre communicative qualities. It is difficult for other than the most knowledgeable experts to see what the system is intended for, what it really does, and what kind of end user functionality the system supplies. Depending on the scope of the architecture, the precision of the descriptions may vary for different system elements. Sometimes important details and complexities can even be totally omitted, which may have dire consequences for the project.

**The integrated view**

The integrated view will impact coherence – achieving a consistent view of the project - , planning/estimation, and risk management as follows.

**Coherence**

The integrated view of functionality (anatomys), project deliveries (increments) and the evolution of the strategic building blocks (architecture) provides a coherent view of the project that can be communicated to various stakeholders. By integrating the strengths of the three views in a single view (see Figure 5), the weaknesses of each view are counterbalanced. Thus, we can keep the focus on the whole, and take early decisions without proceeding too far in any of the individual views; something which is easy to do unless you see how these are related to each other.

A most important benefit from the integrated view is that stakeholders from various disciplines get a common picture to discuss around. For example, in organisations built around a long-time, strategic product architecture, system architects may have a hard time to understand the necessities of project managers (the planning view) and product owners (the anatomy view). Thus, the integrated view will make it easier to reconcile different “world-views” in the organization; something that otherwise may result in detrimental conflicts or unsolved issues.

**Planning, estimation**

A drawback with the classic methods for estimation is the focus on only one or two characteristics within the development process. With the integrated view we have an
instrument for controlling how both features and architectural components are gradually developed. In the cross section between the increment and architecture views, we can express what functionality is suitable to deliver from both a user capability and from design perspective.

This is a starting point for more precise estimations. We may define a first release of a project backlog within a limited time at the start of a project; thus avoiding the need for a prolonged and complex planning phase based on detailing a single type of information. The possibility to express the anatomies and thereby the work packages, both in terms of functionality and components, gives a better understanding of their relative complexity and the size and required effort. It also gives an opening for prioritization, leading to better decision in the management of the project; simply, it will be easier to decide what is worth doing and when it is time to stop.

**Risk management**

In the integrated view we can see the consequences of running into problems with implementation of an increment. If we have a problem with too complex components in a lower layer in the architecture, this will stop the delivery of end user functionality in early increments. By highlighting the architecture, we get a better view of how to make incremental deliveries of functionality within a layer. As runs contrary to agile methods, the integrated view also gives us a possibility to spot such problems early.

To some extent, project risks can be identified in the anatomy view, for example, by identifying single anatomies carrying too many dependencies, or long chains of dependencies. From the order of integration, in either the functional or architectural dimensions, we get the means to evaluate where we have reached a good enough state, and when we enter areas with high risk. This helps us to direct effort to critical areas, and to add appropriate level of risk to our estimations.

**Theoretical interpretation of the results**

When applying ADT as an interpretative frame for inquiring into projects, the following aspects are illuminated. The constructs of the ADT – the activity modalities and the activity domain – represent generalized patterns, the particular manifestation of which will be determined by the contextual conditions of each individual project. *Objectivation* implies that the target of the project – what the project is all about – needs to be brought to the fore and be made clearly visible to all stakeholders; something that is easily forgotten among myriads of documents and plans. In this sense, the anatomy plays the same role as the mammoth played for ancient hunters; as a centre of gravity around which the project/hunt is organized and coordinated. Since the architecture is part of the description of the target, the architectural view can be interpreted as an elaborated illustration of the objectivation modality; with aspects that are not present in the anatomy.

In large projects, the setting is usually a main project, which coordinates a number of sub-projects. *Contextualization* means that every distinguishable work unit in the project will be regarded as an activity domain of its own, with its own target, and internally structured by the activity modalities. In particular, the increments in IDD approach are interpreted as activity domains. *Transition* comes into play when the domains interact. Whatever is transferred between domains has to be worked out between the domains. The local, inner workings of each domain need in part to be made into a commonly understood conceptualization that both domains can interpret in a meaningful way. In this sense, every transition point in the project can be interpreted as a boundary object; i.e. something that tie together actors in multiple social worlds, while being capable of
assuming different meanings in each world (e.g. Star & Griesemer, 1989). So, for example, in Figure 5, boundary objects can be identified from elements found on both ends of a dependency line crossing increment boundaries. These objects will include both technical elements and elements of mutual understanding among the actors in the interacting increments.

Relevance versus rigour
The theoretical problem addressed in this contribution is that of balancing relevance and rigour. Recently, the need for a “practice turn” in project management research has gained momentum (e.g. Winter et al., 2006; Blomquist et al., 2010). This is a reaction against the hitherto predominant rational approach for at least two reasons: the inability to cope with complex management situations (e.g. Ghosal, 2005), and a growing discontent with the meagre results in guiding project practitioners:

Project management research should not consist of merely creating new frameworks and proffering new theoretical models of project management. It should also target the ways in which this new knowledge can clarify and enrich the professional practice of actors involved in project conduct (Lalonde et al., 2010, p. 29)

Thus, there is a need to scrutinize how theory and practice are interrelated. Lalonde et al. (2010) identifies four theoretical types of project management practice with different views on the dialectics between theory and practice: practice as heuristics, practice supported by prescriptive models, practice supported by descriptive models, and reflected practice and situated theorization.

The first type, practice as heuristics, is characterized as a “form of doing” without any scientific theoretical backup: “project management is what project managers do, period!” (ibid., p. 23). The second type, practice supported by prescriptive models, makes use of prescriptive theoretical models to inform practice, and is considered as a positivistic, quantitative approach. The third type, practice supported by descriptive models, addresses human factors using various theories from social sciences, psychology, anthropology, etc.; usually researched by qualitative methods. These three types are denoted “weak” theories of practice because their theoretical models are borrowed from other disciplines outside project management.

In contrast, theories of the fourth type, reflected practice and situated theorization, are called “strong” theories of project management because they are built from within project management itself; what is actually going on within projects – project actuality (Cimic et al., 2006). This type has the ambition to embrace both rigorous-scientific and practice-relevant research, and questions “both scientific research that is out of touch with practice and practice that has no theoretical basis… at the end of the day, the key is to build knowledge that is relevant for the practitioner (and not just for the project management research community)” (Lalonde et al., p. 29).

The main challenge of type four theories is to build a strong dialectic between the poles of theory and practice in such a way that both rigour and relevance are maintained. This requires that theorization efforts start in project actuality:

[Reflective] practice becomes the main empirical ground on which to sketch theories of project management practice (Lalonde et al., 2010, p. 30)

Rigor is achieved by well-grounded theoretical development and relevance by demonstrated practical results, grounded empirically from a standpoint from within the disciplines itself. This in turn requires a theoretical framework of practical philosophy as a point of departure (Lalonde et al., 2010, p. 30). The essence of theoretical development work is “… to construct, using induction and interpretation, theoretical models that must
be continuously and reflectively tested throughout the fieldwork” (ibid., p. 32). In general, qualitative and interpretative research methods need to be employed since quantitative hypothesis testing is not applicable for validating type four theories. In short, this means that knowledge needs to be “pragmatized” in order to generate university research that is relevant, useful, and fruitful (ibid., p. 32).

The research reported in our contribution is a noticeable example of the development of a type four kind of theory. The theoretical development of the ADT emerged from the very same practice at Ericsson in the early 1990s, when one of the authors, Taxén, started to reflect on the practice he was professionally involved in (Taxén, 2009). The constructs in ADT, the activity domain and the activity modalities, have been reflected upon, continuously refined, and reported in a number of publications. This has been done in close connection with empirical undertakings, the experiences of which have been reflected back into continued theoretical development. Thus, a balance has been upheld between relevance and rigour over an extended period of time of more than 20 years.

A particular aspect of the rigour versus relevance problem is the seemingly unsolvable contradiction between the particular and the general:

We argue that the pursuit of explanations that rely upon identifying general patterns based upon cause and effect marginalises the particular, while a focus upon the particular frustrates the emergence of common patterns, shared meanings and normative recommendations. (Smyth & Morris, 2007, p. 423)

In ADT, this contradiction is resolved as follows. The activity modalities represent inherited, biological predispositions for acting in the world, which means that these modalities are general patterns that underlie every form of human activity, socially organized or individual. These general patterns are manifested in the particular circumstances that the individual encounters during her lifespan; some of which may be to participate in projects. Thus, it should be possible to elaborate further ADT into a well substantiated theory, which encompasses both the general and the particular.

From the discussion above, it should be clear that ADT is not limited to a theoretical framework for projects only. It is equally applicable to any form of organized endeavour that mankind have dwelled into since the dawn of mankind. Thus, in management studies, an organization would be apprehended as a constellation of mutually interdependent activity domains; each of which is structured according to the activity modalities. This makes it possible to address a number of issues in organizational science, such as the thorny problem of levels in the organizational discourse (see e.g. Gherardi, 2001; Abell, Felin, & Foss, 2008; Ashforth et al., 2011), the nature of coordination, knowledge integration, and sense-making in organizations, to mention but a few. This is in line with the thinking of Blomquist et al.:

Research will build an understanding of the management problem-based organization seen as a bundle of communities and intertwined practice, rather than as organizational units, levels, and layers (Blomquist et al., 2010, p. 13)

Limitations and further research
There are several limitations to the study presented in this contribution. The empirical data is historically collected from one organization only: the Ericsson telecom company. However, the IDD approach is now gaining momentum in other organizations as well (e.g. Taxén, 2011). This will open up for further corroboration of the results gained so far.

Another limitation is the theoretical framework of ADT; the state of which may be characterized as “work in progress” still after some 20 years of evolution. The reason is
that ADT represent a far-reaching break with mainstream theorizing in PM and management studies in general. To the best of our knowledge, no other theoretical approach tries to ground these disciplines in our biological and neurological predispositions for coordinating and integrating our actions. Thus, ADT is a bold attempt to thread new paths, and as such of course subject to many, still under-researched aspects. For example, it is currently not known how sensory impressions are integrated in our brains to an actionable precept represented by the activity modalities, or even if the present conceptualization of the modalities is the most appropriate one.

At the relevance pole, practical guidelines for an architectural-view enhanced IDD need to be developed and scrutinized in practical settings. Our contribution may be seen as a start-up for such an elaboration. This will encompass new ways of visualizing and modelling the activity modalities, method guidelines, supporting tools, and much more.

**CONCLUSION**

It appears that various PM schools (e.g. Turner et al., 2013) have not been able to device generally accepted, efficient methods and tools for managing projects that fall into the complex Cynefin domain. In this contribution, we have discussed the IDD approach, augmented by the inclusion of architectural views as a possible alternative to the ‘classic’ methods. We have shown how the three views of planning, anatomy, and architecture, can be integrated into a visually powerful and comprehensive illustration, providing a compact way of conceptualizing and communicating the essentials of the project. Thus, we suggest that the integrated IDD approach provides an opening for more precise estimates and enhanced risk management from the outset of the project.

Since the IDD approach has proven its merits in a substantial number of complex system development projects, there is a need to understand why this is so. An analysis of this approach from the perspective of the Activity Domain theory indicates that the main reason is the ability of the anatomy to provide a comprehensive and easy to understand image of the system to be developed by the project. This indicates that manifestations of the objectivation modality in the Activity Domain Theory must be placed at the forefront of project management approaches; an aspect that is usually neglected in extant PM practices. The grounding of the Activity Domain Theory in reflected practice and situated theorization indicates that this theoretical framework is capable of both providing rigorous research that is in touch with practice, and knowledge that is relevant for practitioners. Thus, we have also contributed to the discourse of relevance versus rigor in project management.

**References**


