Structural Views on System Development Practices
- Towards a conceptualisation based on action, interaction and methods

by

Anders Hjalmarsson

Mikael Lind

Ulf Seigerroth

Research group VITS

University of Borås, Jönköping International Business School,
and Linköping University, Sweden

[Anders.Hjalmarsson@hb.se | Mikael.Lind@hb.se | Ulf.Seigerroth@ihh.hj.se]

Abstract

An obvious quest for system development researchers is to understand, describe and give advices concerning the domain of system development. One part of this domain is the System Development Practice (SDP). SDP means the way that system development projects are planned, structured and carried out in order to produce a product that creates value for its client. Our experiences are that SDP often involves three major dimensions; action pattern, method, and role interaction. In these three structural views we find concepts and aspects that are of importance such as, system life cycle, model, method, results, resources, milestones, project roles, project group constellations, clients etc. These dimensions with their concepts and aspects make the SDP a complex phenomenon to its character. In this paper we will therefore give a character to each of these three dimensions and how these dimensions can constitute the SDP as a whole, i.e. a conceptualisation of the SDP. To accomplish this goal we have based this paper on three major sources for inspiration and generation of knowledge within the problem domain that we have identified. Firstly, we all have experiences from several projects and work where we during recent years in different ways have taken part in projects where development of SDP has been in focus. Secondly, empirical experiences from several projects where development of the SDP has been in focus, and thirdly, theories related to the SDP domain. The approach bears on the foundation of language action theories for information systems, information system development and practices for understanding the domain of SDP. By identifying essential acts performed in SDP's we are able to regard the relations between the three dimensions in order to form a holistic view, which means that we can understand and have constructive discussions about SDP's.

1. Introduction

When business is performed there is often a need for different actors to co-operate in order to achieve results that are of value for its clients. The results, i.e. the products, which are being delivered, should correspond to the client's expectations. These expectations are initially established during offering and contracting. Co-operation between internal and external actors is among other things performed through communication. The information systems associated with an organisation will include both IT-supported and manual communication.

Practices of today generally need to have a continuous development process in order to evaluate and sustain the value that are offered, developed and delivered to the clients. Continuous improvement of the organisations information systems is therefore needed to improve communication and thereby also their organisational ability.

In order to facilitate the need for continuous improvement as well as development of new information systems there exist practices that specialise in this task, so-called system development practices (SDP). Our main premise is that efficient performance through communication also is important for SDP's. By talking about benefits, gains and advantages of system development there is an emphasis on that the result of the system development practice, i.e. delivered information system, should be a support during the value-creation for the clients. The benefit of SDP's is to deliver information systems as communication support to its client in order to facilitate the production of value.

As mentioned above practices in general need to continuously change in order to be able to create products that meet present and future demands. This change in *ad infinitum* also includes SDP´s.

When performing development acts there is a need to have a language, a conceptualisation, to be used in conversations about the practice in question. Such conceptualisation needs to be based on the acts and tasks that in reality are being performed, and also on an understanding of the results that is being produced and delivered by the practice. In a SDP there is a necessity to have a thorough understanding of the notion of information systems as a product and its implication on development and introduction.

We regard practices as actors who perform acts. These acts are structured in action patterns in order to ensure efficient value production based on assignments. These action patterns make up the SDP's structure, which is constituted by project structures since each result from system development is to be regarded as unique. Since there are several actors involved in SDP's, these actors need to be organised in a way that support efficient system development through a supportive structure of role interaction. Another phenomena related to system development are that there exist a great variety of methods for system development, which are supposed to constitute a support for development acts. We therefore argue for that there exist three structural views in SDP:

- action pattern,
- method and
- role interaction

By regarding these three structural views as a whole and how they are inter related with each other we suggest a language to conceptualise the SDP.

In this paper we are analysing concepts used in several case studies, in which have been taking part in, oriented towards development of SDPs. These concepts have been derived from and used in the case studies to be able to talk about practice situations and the problems concerning system development. With these concepts as a foundation we propose a conceptualisation of SDP's, which is based on L/A-oriented theories about information systems and practices as well as experiences from performed case studies. We argue that this conceptualisation can contribute as a tool to ensure that actors within the SDP, as well as to some extend the SDP's clients, in an efficient way can create mutual understanding and understand the benefit of system development. The presented conceptualisation proposes an integrative view on action pattern, method and role interaction. To be able to do this we elaborate with a number of concepts that are the characteristics of SDP's combined with general characteristics of practices in order to conceptualise the SDP's in a holistic manner, i.e. a suggestion for three structural views and their relationships. The paper does not favour a specific instance of any of these three structural views, the paper rather elaborate on meta-concepts that has general implications on the instances.

The paper begins with a theoretical framework where we discuss foundations for practices, structures and system development. The following chapter put forward the three structural views separately by using the notion of social action as a base. In chapter four elaborate on the relationships between the three structural views, first two-by-two as a base to end up in a conceptualisation of the SDP as a whole which includes all three structural views. The paper is then concluded and further research is identified. Before we turn to these described parts of the paper we first want to discuss some research methodological aspects that has permeated our research work.

As stated above we have built our conceptualisation on a base created from findings generated from several case studies at different SDPs. Our ambition has been to identify and evolve different categories and concepts that we have recognised in our many concentrated interactions with actors in these case studies. The driving force of this strategy has been our aim to create an understanding about how SDPs work and to create foundation, a conceptualisation, in order to discuss aspects when developing these practices. The techniques used in this collection of data has varied, ranging from more or less structural interviews to seminars and workshops with the ambition to collect individual and collectively grounded knowledge about notions in the specific SDPs. From this data collection we have been enabled to create categorisations of the concepts used by conceptual modelling techniques advocated by for example Goldkuhl (1992) and Goldkuhl & Röstlinger (1993). As argued by Walsham (1995) a strict inductive way of doing research could be restraining thus it neglect the possibilities that the researchers pre-knowledge and other theories may posses in his or hers effort to interpretative and elaborate with the findings (cf. Glaser & Strauss 1967). Pre-knowledge and theories has therefore been used to create added power in the efforts to analyse and modulate the empirical findings. The pre-knowledge and the theory framework have also created the incentive to encourage and push the research process further and to analytical depth. The character of the process can therefore be described as iterative where theoretical aspects has been used to analyse, evolve and grasp the empirical findings as well as pursue the process forward.

This strategy is by Walsham (1995) described as a iterative research approach and Alvesson & Skjöldberg (1994) labels it as abduction, as a combination of a deductive and a inductive approach. Figure 2 illustrate the configuration of our research methodological strategy:
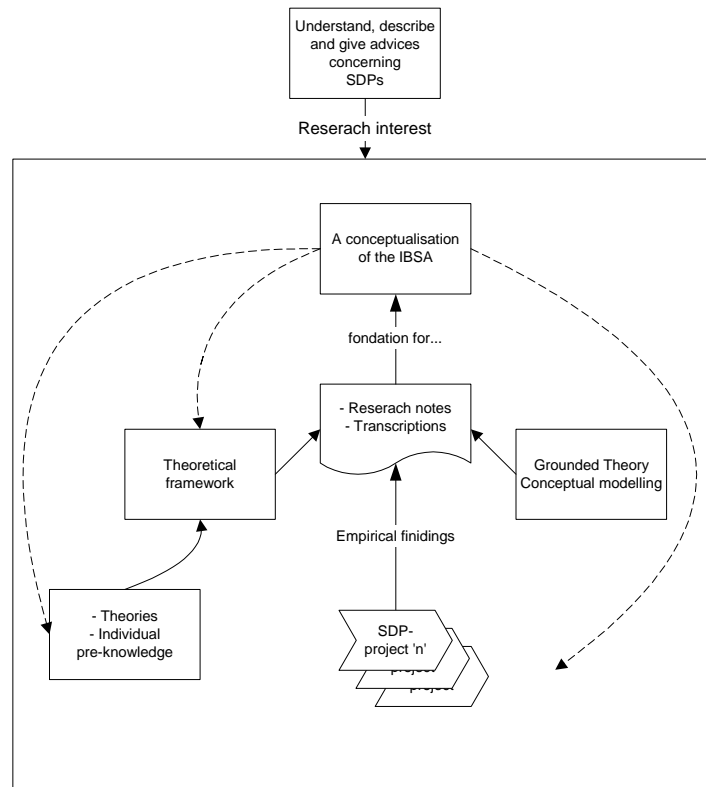

Figure 2: Research strategy

2. Theoretical framework

In this section we will give a number of theoretical underpinnings used to create a base for the coming sessions. Note that the theory presented in this section is not the only theory in the paper. Section 4 is based on additional theories in combination with empirical findings identified in section 3.

2.1 Organisations as practices that agree upon and fulfil assignments

As stated in the introduction organisations can be regarded as a collection of actors that co-operate in order to produce results for its clients. Within in and between organisations there exist actor-relationships. Actors, in the role of the organisation, perform actions directed towards other actors within the organisation itself and actors within other organisations to which the organisation has or establishes actor relationships.

Our view on organisations is to understand the organisation as a human practice (Goldkuhl & Röstlinger, 1999). Such view on organisation uses a number of actor relationship-oriented categories to characterise organisations. The categories used are shown in the ToP-model below.
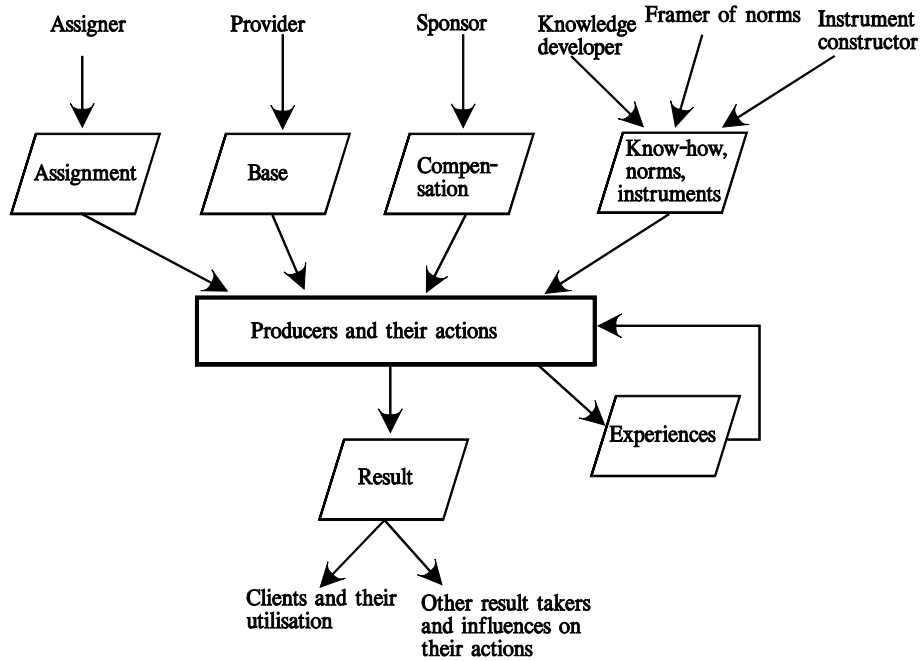
Figure 3: The model of generic practice (ToP model) (Goldkuhl & Röstlinger, 1999)

The unit of analysis used in the model depicted above is business act. According to Lind & Goldkuhl (2001) a business act need to be regarded as multi-functional and be seen as either communicational or material in order to facilitate an understanding of the essentials of the organization. The ToP-model is based on, but also transcends, the language action perspective (Goldkuhl et al 1998, 1999). The language action perspective emphasises the need for focusing actor's communication in order to apprehend human action. Communication is more than just transfer of information. To communicate is to act. Such a communication perspective has consequences on the notion of information systems. Information systems within organizations are systems consisting of actors communicating with each other. IT-supported information systems are systems that enables communication according to a number of predefined rules. Information systems should therefore be regarded as communication systems used for actors in their co-operation with other actors as well as for co-ordination of actions.

Actions performed in organisation are however not only communication based. Organisations do a lot of value-adding transformation-oriented material acts in order to ensure that physical goods delivered to its clients are of high value. Organisation are many times seen as units of transformation that takes one or several basis and by adding value to this/these basis delivering products to its clients. Transformation does however not always have to do with physical products. Transformational acts are also for example performed when we develop information systems. Information systems, or communication systems, are artefacts. The basis in a system development context is the demands from the users and the organisation, and the product is the information system.

In the language action community there is a clear orientation towards regarding human actions in patterns built up by inter-related speech-acts (Winograd & Flores, 1986). Such patterns are used in order to emphasise the commitment made by one or several actors as a basis for the actors' fulfilment of established commitment. The pattern is concluded by the actors' declaration whether the fulfilled commitment met the expectations or not. Such view on organisations can be emphasised as an assignment view on organisation. In methods such as Action Workflow (Medina-Mora et al, 1992) and DEMO (Reijswoud, 1996) such patterns are regarded as the essentials of the organisation.

As the TOP-model above indicates that these two focuses on organisations complements each other. Transformation of input (or basis) to output (or results) needs to be regarded in an assignment perspective. It is important that we find out which expectations the client (or the assignee), has on the products that are supposed to be produced and delivered by the producer, before production and delivery is made. Such a combined assignment and transformation view on organisations shall be used to understand organisations as practices that agree upon and fulfils assignments (Lind, 2001).

2.2 Towards a multi-dimensional character on the concept of organisational structure

Structure in sociology is often related to the term *social* - as for example in social structure: the distinctive arrangement of institutions whereby human beings in a society interact and are able to live together. Social structure is often treated together with the concept of change, which deals with the forces that change society and the social structure itself. Different views on how structure is changed exist. One view claims that the social structures that exists around us shapes the actions that actors are carrying out in an organisation; for example Durkheim (1893/1984) argues that common values and the change of these is the ground for how a change in an society such an organisation can be understood. Other views argues that the individual actor can with his or her actions by her self affect the structures that he or she is a part of; for example Weber (1983) therefore argues that social changes and structures should be analysed in the perspective of the action that individual actors generate (Boudon 1986). A third view from Giddens however argues that we should go beyond this dichotomy and combine this view in a more multi-perspective view in order to fully understand the mutual nature of structures in a social context, "the duality of structure" (Giddens 1984). The basis of Giddens theory impose that structures neither are created by unilateral actions that internal members of the organisation carries out or from the forces outside, but by actors that continuously re-creates the structures of the practice through rational actions. The concept of structure in relation to sociological aspects is by this frequently used in relation to organisations and is therefore intensively discussed in organisational theory.

What characterised the early debate on structure in organisational aspects was that it mainly was focussed on formal structural aspects of the organisation and on power, control, supervision and technology (Scott 1998; Wilson, 1999). Evidence of this is for example the previous mentioned Weber (Weber 1983) Barnard (Barnard 1938) and March & Simon (March & Simon 1958). In Webers analysis of the deviation from the bureaucracy is viewed as organisational dysfunction. Barnard was primary interested in the formal organisation because of his view that it represented a system of co-operation. March and Simon was interested in how the formal organisational structure can influence the actors in the organisation with the intention to improve their ability to accomplish stated business goals. As a reaction against this formal view on structures other views on structure has evolved during the evolution of organisational theory. Critics argued that this focus on formal organisation and on bureaucracy as an organisational form neglected the dynamics in the modern and postmodern(?) form of organisation (Geary, 1995; Clegg 1990; Wilson 1999). This debate is extensive and is a subject for an entirely own analysis.

However with this introduction as a foundation it is clear that the meaning of structure is a bit unclear and divided. As we will see, several different classifications even on a general level are available. For example in Britannica structure is explained as *the aggregate of elements of an entity in their relationships to each other* as well as *something arranged in a definite pattern of organisation*. The Swedish national encyclopaedia explains the term as follows: *a coherent internal construction amongst the parts in a totality*. Piaget (1968) uses a more specified definition from his perspective of structuralism when he discusses the term of structure as *a system of rules for formation and transformation that creates meaning, this is conveyed by signs adherent to a language*. The conclusion that can be drawn from these definitions is that a structure implies some level of order, demarcation and stability and thereby is separated from what can be described as situations where totally randomness and chaos are endeavoured to prevail.

In organisational theory these more general definitions has been explored and developed further, often however with one perspective or dimension in focus neglecting others. This has resulted in a plurality of definitions. In combination with the high level of general vagueness that concerns structure, the many definitions has led to that the term is enclosed with obscurities (Boudon 1986). The clarification of organisational structure therefore must be seen as an important task for everybody who wants to understand the nature of how practices work. It is also our belief that the ambition to clarify must move beyond a discussion concerning if structure should be seen only as formal or informal or a only as a combination between the two. Structure in relation to organisations should instead be viewed as multi-dimensional and as plurality as a replacement for singularity. The organisation contains several different structures, which together make up the whole of the structure in a practice. To understand the practice the organisational structure must be viewed from a holistic perspective, but with the different forms of structural patterns that exists distinguished. Löwstedt (1995) argues that there is a need for categorisation of the types of structures that exists, and proposes the following categorisation:

- *Physical structures.* This includes for example buildings, machines, computer networks, raw material an so on
- *Formal structures.* Which includes business plans, organisational schemas, instructions and rules for how the work should be carried out in the practice.
- *Mind structures.* This structure includes the knowledge that the actors have, as well as ideas, conceptions and perceptions about general and specific phenomena in the organisation.
- *Dialogue structures.* This type of structure refers to the way actors in a organisation discusses and exchange ideas about other structures in the organisation.
- *Action structures.* These structures are the representation of the patterns of action that the actors in the organisation are practising.

This view of structure as multi-dimensional is fertile to explore. We believe that the structure of a practice is complex and therefore multifaceted. Only by viewing the structure of the practice from several different perspectives the whole of the structural patterns could be unfolded in a practice. Our ambition is to create a conceptualisation that grasps these different structural views. We don't believe that our ambition by this paper is finished, as we will argue in the final part of the paper. However we believe that our ambition has created a grounded conceptual platform that could be evolved through further research. Our view on structure as multifaceted does not imply that we see structural aspects of the practice equal in base. In our belief action and action patterns are the main imperatives of the structure of the practice, therefore this type of structure act as an foundation for the other dimensions of structures that Löwstedt describes. One way of conducting the research would be to elaborate Löwstedts categorisation, however our pragmatic ambition on creating a usable conceptualisation imply that we go beyond his theoretical division of structure and instead creates from a theoretical foundation based on a notion on social action new categorisation of structure. This categorisation will be based on different views where different aspects pulled from this social theory as well as aspects of structures in practice are related to each other in the dialectic relationship according to foreground and background depending on the focus in the specific view. Before we present the essential structural views, which are foundation for our conceptualisation, we first will discuss the conditions and characteristics of system development.

2.3 System development – characteristics and conditions

"*The job of software engineering is to deliver high-quality software products at agreed cost and schedule*" (Humphrey, 1997). This over all definition of the purpose with software engineering is also reasonable to apply to system development (SD) since software engineering can be regarded as a part of SD (Jayaratna, 1994). There is however a difference between software engineering and SD where software engineering aims at producing a product that match the quality and needs of a specific software specification, while system development aims at matching the quality and information needs of a particular user (ibid). However, SD does not only mean development of an artefact that matches the information needs of a user that then is introduced in some practice. System development has the characteristics that the practice itself as well as the SD-work, more or less articulated, is the subject for different development efforts. We argue for that system development mean and should mean change and purification of different aspects within the SD-domain. Before we present different aspects that could the be subject for development and purification we will give some important characteristics and conditions for the SD-domain.

Work within the SD-domain is known to be a complex area where the SD-actors often relay on different types of formalised knowledge, such as methods, for guidance through the SD-process (Hoffer et. al, 1999). The characteristics of these methods are that they often are of different type and directed towards different parts of the SD-process. SD-methods also exist on different abstraction levels from, giving an overall structure for the whole SD-process (framework) to precise prescriptives of how to perform a specific task (method components), see section 3.3. The methods that are used in SD-domain are also not just "pure" SD-methods. In many cases these methods are combined with methods that are more directed towards project management and project coordination. Our belief is that work within the SD-domain implies on formalisation of actions and language and where methods are one example of a conceptualisation that supports such formalisation. In section 4 we will elaborate these things further in order to give a suggestion for a conceptualisation of the SD-domain.

Now then, what do we mean by that system development mean and should mean change and purification of different aspects within the SD-domain as we stated above? We, and others, believe that

an important part of the SD-work must be to try to improve the SD-work itself. One important condition for this is to have a reflective process, parallel to or after, the actual SD-process, i.e. learning through abstraction to a meta level and discourse (Goldkuhl et.al, 1997; Habermas, 1984). This can be done in different ways but some kind of general pattern for this could be to go through the following improvement cycle; define the quality goal, measure the product quality, understand the process, adjust the process, use the adjusted process, measure the results and compare the results with the goal. The result from the comparison of results and goals is then input in the process to adjust the process (Humphrey 1997). This should then be an ongoing process in all SD-projects. Aspects in the SD-domain that we have found to be important to improve are therefore:

- Action logic within the practice, the SD-process
- The language used during work in the SD-domain
- Conceptualisations of action logic and language in terms of methods

In order to deal with these aspects of improvement there is a need to not just focus these characteristics and conditions as isolated phenomenon. It is instead important to also take into consideration what relations that exists between them and how they can/will affect each other. This will be further elaborated in section 4.

3. Three essential structural views on SDP

In this paper we have identified three essentials structural views that can be used to characterise and talk about system development practices (SDP's). The essential views form the basis (see figure below) to make abstractions about relevant phenomena within SDP's. These essential structural views are action pattern, method and role interaction. The relationship between these fundamentals and abstractions are shown in the figure below.
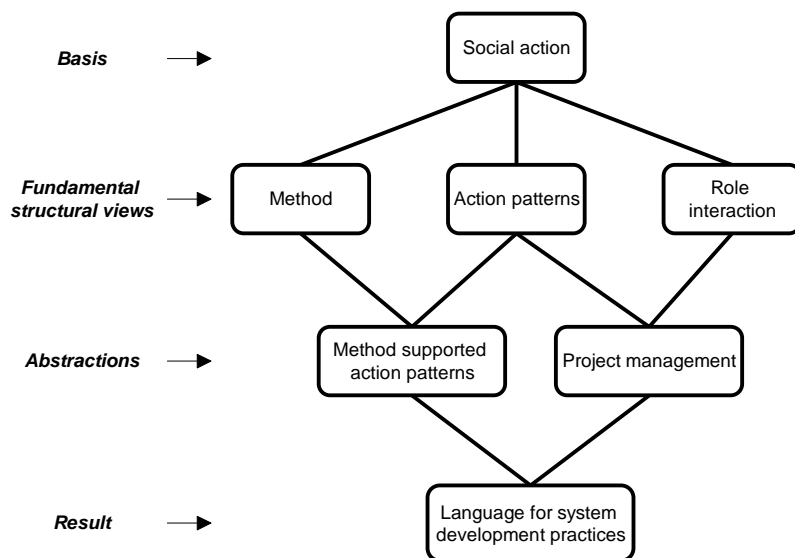

Figure 4: Basis, fundamentals and abstractions for forming the language of SDP's

In the figure above we also identify social action as the basis for the essential structural views. By this we mean that we regard organisations from the perspective of social action. The abstractions and result are discussed in chapter four of this paper. In this section of the paper we elaborate on the three fundamental views as well as the basis for these views.

3.1 Basis: The notion of social action

Organisations consist of humans acting towards each other and together with each other. Humans act in order to achieve ends (Goldkuhl & Ågerfalk, 2000), where human action often aims at making material changes. Humans do however not only act in the material world – they also act in a social world consisting of other humans and their expressions. In the social world humans communicate. Austin (1962) and Searle (1969) mean that to communicate is also to act. Human action is about making a difference, where such difference can have impact in the social as well as the material world.

Goldkuhl & Ågerfalk (2000) have presented a generic model of social action (see figure below), where they include both communicative and material acts, by distinguishing between the following

categories as important phenomenon for social action:

- Action prerequisites (external and internal)
- Actor (interventionist)
- Acting (performance of action)
- Action result (what is done)
- Receiving (interpreting) action result)
- Actor (recipient)
- Effects of action
- Relationships between actors
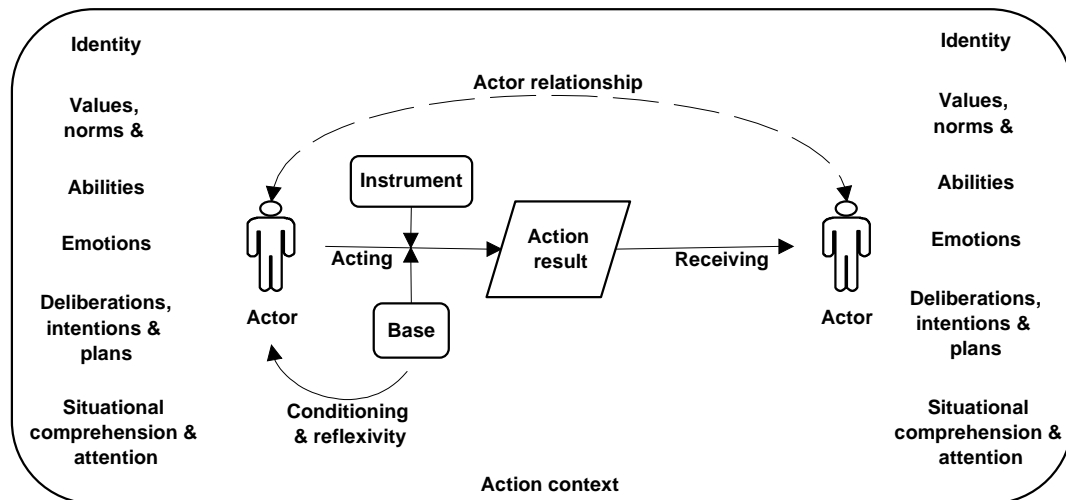- Action context (place and time)

Figure 5: A generic model of social action (Goldkuhl & Ågerfalk, 2000)

As depicted in the figure above there exist within the scope of social action a number of action-related concepts. Concepts used in the generic action model to elaborate on in order to put forward the three essential structural views are the action in itself, instrument used when acting, basis used for action, the actors (both interventionist and recipient) as well as the action results. All these concepts are forming the basis for each structural view, but depending on the structural view some aspects are more emphasised/ focused, i.e. are in the foreground, than others, i.e. are in the background. When we as humans regard a complex phenomenon we have a need to reduce complexity by regarding the same phenomena from different views (Morgan, 1997).

3.2 Structural view 1: Action pattern

The first structural view that we deal with is a part of the project dimension. A lot of work concerning system development is often project-based since the task, i.e. the information system that is going to be developed, often is unique. There are also a lot of different competencies involved in order to manage the production of the resulting information system (Murch, 2001; Yeates & Caddle, 1996). Project is a phenomenon that concerns two aspects; project as action pattern and project as a role interaction (Murch, 2001). This section deals with project as action pattern.

Based on the model of social action the aspects that are emphasised in this structural view are *actions* related to each other in order to produce *result*. Actions need to be related to each other in order to regard patterns used to produce results. Actions are related to each other by regarding the result of one action as the basis for the following action etc. A series of related actions constitute an action pattern.

A project can be characterised as having a clear start and end, where the starting point is some needs from the client that needs to be fulfilled. The end point of the project, which is the ultimate result, is when satisfaction has been reached for the producer and for the client. In order for a project to have a well defined start and end there is a need for the project to have a clear defined goal as well as clear defined expected result. When talking about projects one can characterise the types of goals in the following (Hedberg & Lind, 1999):

- Change goals, which concerns with the issue of expected changes in the organisation that the project is about.

8

- Project goals, which concerns with time and resources that can be consumed within the project.
- Quality goals, which concerns with quality goals at both process and product level within the project.

When formulating goals for a project as well as evaluating the project it is important that one reaches a balance between the quality of the result, the amount of resources that needs to be put in to reach the expected result and the calendar time. The figure below shows these relationships.
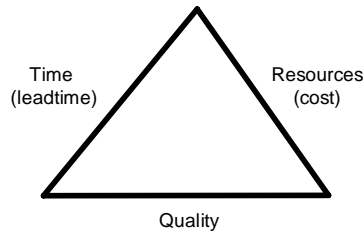
```
        Time                    Resources
     (leadtime)                   (cost)

                    Quality
```

Figure 6: Dependent parameters to be used when optimising projects (Hedberg & Lind, 1999)

The goals mentioned above needs to be agreed upon before the actual execution of the project. To fulfil a goal means making a difference. Actions are performed in order to make differences, i.e. fulfil goals. Within the scope of a project actions are performed in order to arrive at fulfilled goals. Since goals need to be agreed upon and also evaluated, there is a need to arrange actions in patterns that facilitates such needs. Actions can therefore be grouped into phases as well as projects often are divided into a number of phases. A project is divided in the following phases:

- Agreement phase, which is oriented towards coming to an agreement concerning the goals to reach in the project.
- Execution phase, which is oriented towards co-ordination and performance of actions for refinement.
- Conclusion phase, which is oriented towards evaluation of the result in relationship to the formulated agreement.

Within the language/action community (Goldkuhl et al, 1998; Goldkuhl et al, 1999) there are a number of theories and methods that regards the essentials of the organisation to be speech-acts issued by different actors. Studies have been made where work in organisations can be reduced to patterns consisting of inter-related speech-acts. Conversation-for-action (CFA) (Winograd & Flores, 1986) relates such speech-acts as request, promise, report on execution, and declaration. Such patterns of speech-act are very close related to the action patterns used in project-based work. The first two speech-acts are about coming to an agreement (agreement phase), the latter two speech acts is about reporting and evaluating the execution (conclusion phase). In between the agreement and conclusion phase the result need to be realised, i.e. the execution phase, based on the agreement.

In the figure below we have put forward a structure, an action pattern, for project based on the three phases above. The figure also indicates a distinction between the performance level, on which actions that produces result (R) based on prerequisites (P) take place, and the decision level, on which action concerning decisions about the future work in the project take place. The performance level indicates a number of inter-related actions that produces result on the preceding action. Actions are inter-related in order to form action patterns.
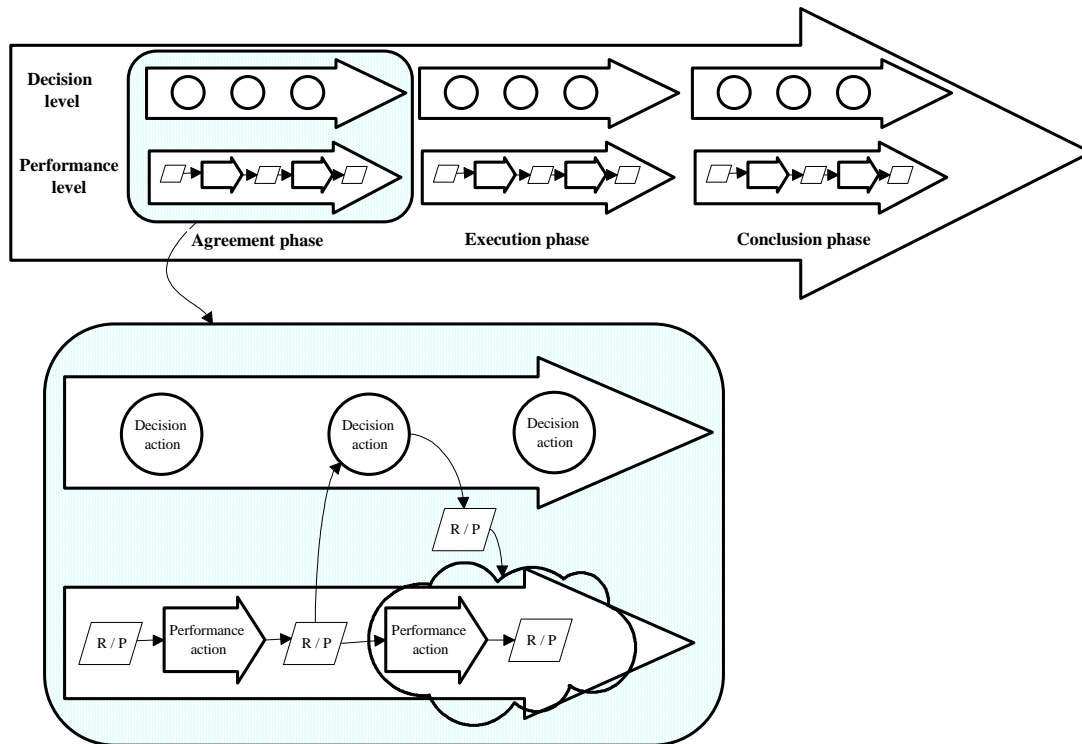
Figure 7: Project as work structure

As the figure above indicates there exist a number of decision points in a project. Some of these are called milestones, which are states where decisions about the altered direction about the project can be made. Milestones and decision points are used to ensure that expected results are reached in the projects. By formulating milestones one will break down different types of goals for the project in different part results. Such part results both determine the action patterns as well as constitute the link between method and action pattern. The backbone of an SDP is the action pattern, determined by the project as work structure, which the other structural views are related to.

3.3 Structural view 2: Method

The second structural view of SDP is method. SDP's often use methods as support to perform different actions through out the system life cycle.

Based on the model of social action the aspects that are emphasised in this structural view are *instruments* as support for related actions in order to produce *result*. Instruments such as methods are often used as support in order to produce certain results.

A method is guidelines for work. Its character is prescriptive (Goldkuhl et. al, 1997). A method tells what to do in different situations in order to arrive at certain goals. In this paper we deal with the method dimension as a process, divided into phases and where different results (R) should be produced based on different prerequisites (P) (see upper part of figure below).
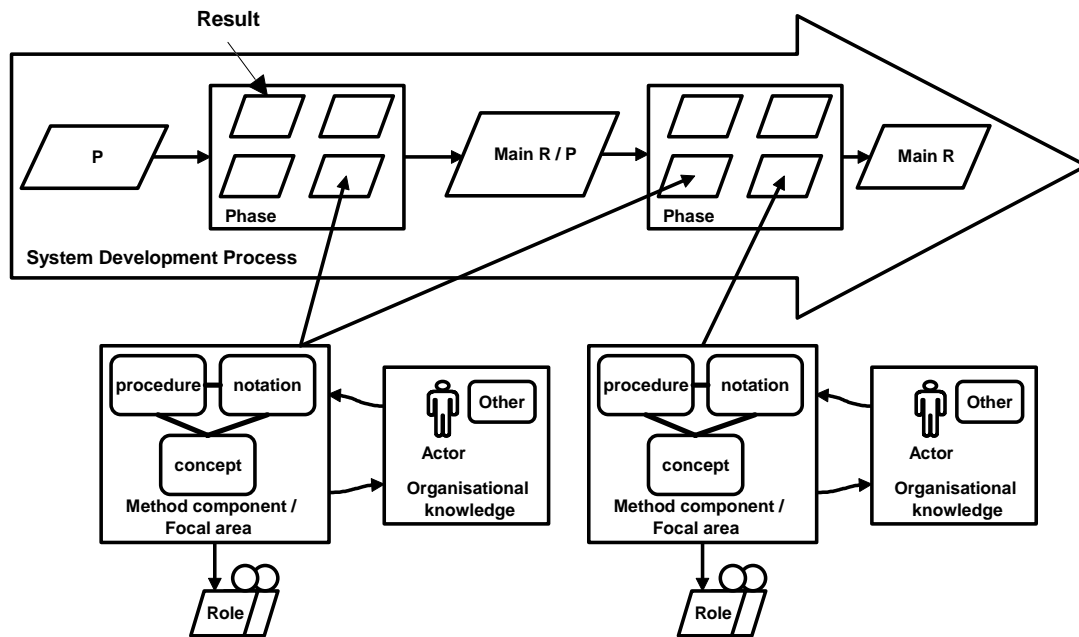
10

Figure 8: SDP and method support

The main goal for the system development process is to produce a result that can satisfy a client. In order to do this there is a need to have control over the development process in different ways. One-way to do this is to structure and divide the system development process into different phases and where prerequisites for a phase, and results from a phase are defined. This gives an over all structure of *what* to do and in *which order* to do things but not exactly *how* to do it. This is often referred to as model or framework (ibid). Within a phase of the system development process we can also accentuate different results that should be produced during SD. As mentioned above we often use methods as support to produce different results. In the figure above this is illustrated with method components / focal areas. A method component should give guidelines on *how* to do things in order to produce a certain result. During SD there is usually a need to document different aspects. Many ISD methods therefore include representational guidelines; what often is called modelling techniques or notations. Such methods also involve procedural guidelines; i.e. how to work and what questions to ask. The notation prescribes how answers to these questions should be documented. Many times the procedure and notation are tightly coupled to each other. The procedure involves some meta concepts as e.g. process, activity, information, object. Such general concepts are used when asking the questions; i.e. they are parts of the prescribed procedure. They are also parts of the semantics of the notation. The concepts are the cement between procedure and notation; the overlapping parts of procedure and notation. When there is a close link between *procedure, notation* and *concepts* we call this a *method component* (Röstlinger & Goldkuhl, 1994); cf the notion of method fragment by Brinkkemper (1995). A method is often a compound of several method components to what is many times called a methodology (Avison & Fitzgerald, 1995). These different method components form together a structure. We call this a *framework*. This includes the phase structure of the method. A method component will also represent a focal area in the system development process. A focal area addresses a certain aspect that is in focus to produce a certain result. Example of focal areas could be goals, problems and objects and where we can have distinct method components for goal analysis, problem analysis and object analysis to address these focal areas. As a consequence of this we can see in the figure above that a certain method component can be a support in more than one phase of the system development process depending on what results that should be produced.

This way of thinking about the system development process is not in conflict with either a more sequential approach such as the system life cycle or a more incremental approach such as Rational Unified Process (RUP). If we use a system life cycle approach we can divide the process in phases with relations between these phases and divide these phases into sub phases with relations between these sub phases and so on. The horizontal relations will be accentuated. If we on the other hand use a more incremental approach we will accentuate what in an iteration that is performed incrementally and how it will other iterations when they are performed more or less at the same time.

The actors who use a certain method component often have a certain competence. A business analyst who has competence regarding performing business analysis could for example use a method

component for business analysis as support during his/her work. In this case the actor has a certain role and the role is associated with a certain type of competence that is needed to produce a defined result in the system development process. This also means that different actors, to different extent, can be involved in different phases of the system development process. In order to produce desired results there is a need to get hold of organisational knowledge concerning the client's organisation. This can be achieved by involving actors from the practice that is in focus for the development effort. There are however, not just organisational actors that can contribute with organisational knowledge. There are other things that can be of value such as different documents, informal meetings etc.

3.4 Structural view 3: Role interaction

In an SDP different actors co-operate together with the purpose to create the results that are of value for the clients. In order to understand structure in SDPs as a whole the need for a holistic view on its exists. Above has *method* and *action pattern* been highlighted as views of structure. In the case of action pattern this view has been related to the phenomena project. This section also deals with the phenomena project, however not as an action pattern but as arena for *role interaction*.

The SDP is organised around particular activities. These activities are arranged in relation to one or several actions, which together results in certain outcomes, action results, and thereby creates effects on the receiving actor how interpret and uses what he or she receives. This outcome, the action results, can be both of materia and/or knowledge/experience. The results are however predominantly aimed to satisfy and create benefit for the receiving actor, for example a client and the experience enables future learning possibilities for the actors in the SDP as well as the client.

As touched on in part 3.2 different acts are conducted on different levels. Two levels, the performance and decision levels, where identified throughout the phases of the project. To be able to understand the complexity that is present in specific individual acts, between different acts and between different acting levels the actor dimension must be decorated in the sense of *role interaction*. Based on the model of social action that was presented in 3.1 the aspects that are emphasised in this structural view are how the *actors* are related to each other in order to conduct *action* and perform *results*. *Actors* are hereby positioned in the foreground and action and results in the background. Actors are related to each other by regarding the roles that they array themselves in a project. Figure 9 below describes this view, by focusing on the relationship between actors based on their roles as the main variable in the specific action:
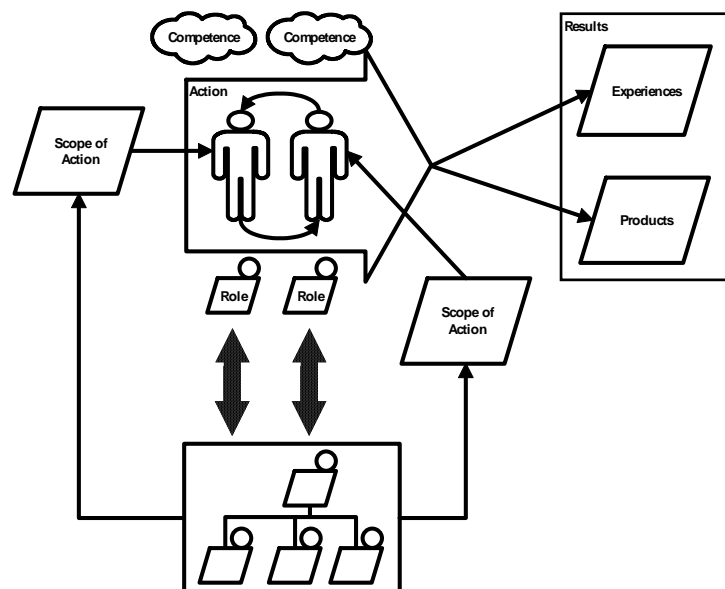

Figure9: Role interaction

Action cannot be existent by itself. Action requires that one or several actors are carrying out the individual acts in the action. In doing this the actors participate in actor relationships with other actors with the common intention to generate results through the use of their aggregated ability. In performing a project the actors therefore put on one or several different roles; for example project sponsor, project manager and project member. These roles inform and position the actor with certain role specific possibilities, obligations and duties. It also defines the actors scope of action. Consequently the role 1)

act as a explanation of what responsibilities that are delegated to the actor when he or she are acting in a specific role as well as 2) function as an agreement form a declaring convention of which obligations the actor is expected to carry out when he or she array a particular role. In relation to the three roles that was used from the project context, the project sponsor, the project manager and the project member three different categories of roles emerge which can be used as a model to analyse and understand which actors that acts on the different acting levels. These categories, meta-roles are *project responsibility role* (including for example project sponsor), *project manager role* (including for example project manager) and *project member role* (including for example GUI-designer etc).

What can be concluded from the empirical studies that has been carried out is that understanding about different roles and their scope of action and therefore *descriptions of roles* are highly important for efficient accomplishment in SDPs. The importance should however not only be reduced to that formal roles and description explicitly exists, but that they also are accepted by the members in the practice, constructed with the practice action patterns in mind and that the specific SDPs roles in for example inter-organisational work conditions are calibrated with the system of roles that the external organisations has. In one SDP studied no formal explicitly made structures of role interaction or descriptions was available made the interaction between different actors in different projects as well as the line organisation difficult to understand, manage and talk about. The lack of this system of roles and role interaction caused problems in project phases; problems such as process stoppages, conflicts and that certain tasks were neglected. The problems were even more frequent when the SDP worked with other businesses in inter-organisational collaborations. In the case of the intra-organisational projects more informal and value-based systems of rules and hierarchy controlled the projects more or less efficiently, but in the projects with inter-organisational character different cultures and experience based ways to perform acts where mixed without no formal structure as base and support. These projects where quite successful if the resulting products are being evaluated, but filled with mistakes, misunderstandings, conflict and pressure if the process of accomplishment is being studied and evaluated.

In the case of the another SDP, one of the IT-departments in this SDP had deliberately calibrated their old and embedded system of role description with the system of roles that their new system development method prescribed. This action of calibration resulted in that the members in this SDP more easily understood could examine and accept new definitions on roles and changes in the arrangement of existing roles. The study of the other IT-department in the multi-national corporation displayed a more chaotic work situation as well as a persisted view on formal role descriptions. The difference between the two departments were that the latter department was rather new as organisational unit, had an unclear and not accepted function in the corporation as a whole, had members that were overall new, and a management function that was not keen in working systematic and saw structures more as obstacles than as support. The consequence of these differences resulted in that this department had more difficult to plan and execute projects and was consequently also them affected with the disastrously problems mentioned above.

What the formal description of roles comprehensively informs the participating actors of is what is expected from him or her in the accomplishment of the action. It could therefore be argued that the role and the description of the role could and should serve as a description of what *minimum* is expected of the actor in certain situation in order to create a structure of systematics and a foundation for creativity – and not something that restrain or damage the ability to be creative. Therefore can it be stated that different roles jointly constitute the formal organisational structure in an SDP and are consequently the formal system of action rules that exists in the SDP. These rules should be obeyed in order to create "hectic" but orderly work situation in the inventive and innovating SDP. This formal system of action rules should also be seen as either a result from an intentional design of how the practice is intended to be carried out or a redesign of more social constructed rules that are engendered through the SDPs evolution.

The rationale and intention with this logic is that it should be viewed as a system of rules for the actions, communicative and material, that occur between actors when they are acting through different roles, e.g. project manager, system analyst or vice president. These three examples is also a illustration how roles is infused with different levels of authority, which impose more or less responsibility to the actor when he or she is acting in the role in question. The different levels of authority in various roles form a hierarchy between different roles, and subsequently actors, in the practice. The hierarchy is important as the foundation for the system of action rules that directs the actors when they are acting

Consequently the rules of action that are appointed to the specific role creates the actors scope of action in the specific activity. This because of the previously described that the role informs the actor what he or she can and should do when performing the action. It is then up to the actor - by using its

competence - to carry out the tasks in co-operation with other actors in order to create the results that the specific action is aimed for. The fundamental purpose by viewing SDPs from this third structural dimension is that a formal system of action rules establish a set of conventions that should be used to support and guide the actors in their interaction with each other and when performing the actions in the SDPs in a systematic way.

4. Towards a conceptualisation of SDP's

All three views of SDP that we have presented above represent different perspectives that can be put on a SDP situation. We therefore argue for that it is important to address these three dimensions in order to have a constructive discussion about the characteristics of SDP's. This section shall therefore be regarded as the result of the paper. The section starts out with relating the fundamental structural views, two-by-two, to each other in order to end up in a conceptualisation where all three structural views are related to each other as a whole.

4.1.1 Method supported action patterns

The major concern with a project is to ensure that expected results are reached within agreed cost and schedule (Humphrey, 1997). We have in different projects experienced that the work structure in a project has a strong relation to SD-methods and vice versa. In reality it is hard to talk about project-based action patterns without taking in to consideration what demands SD-methods will have on the action pattern. One example of this is when an organisation decides that they should use a more incremental oriented SD-model such as RUP (Rational Unified Process) without changing their old project model, as for example PROPS, that is more sequential oriented and in which you have to finish things in a sequential order and go through toll gates where business decisions are made. On the other hand we have also seen that there is also hard to talk about and to deal with SD-methods without taking project characteristics into consideration. One example of this is when a SD-project is on such a tight schedule that you cannot go through every recommended step of the method. In these cases you will have to adjust ordinary methods (stripped within reason) in order to reach project goals. The relation between action pattern and SD-methods becomes obvious through the different results that should be produced through out the project according to Figure 10 below.
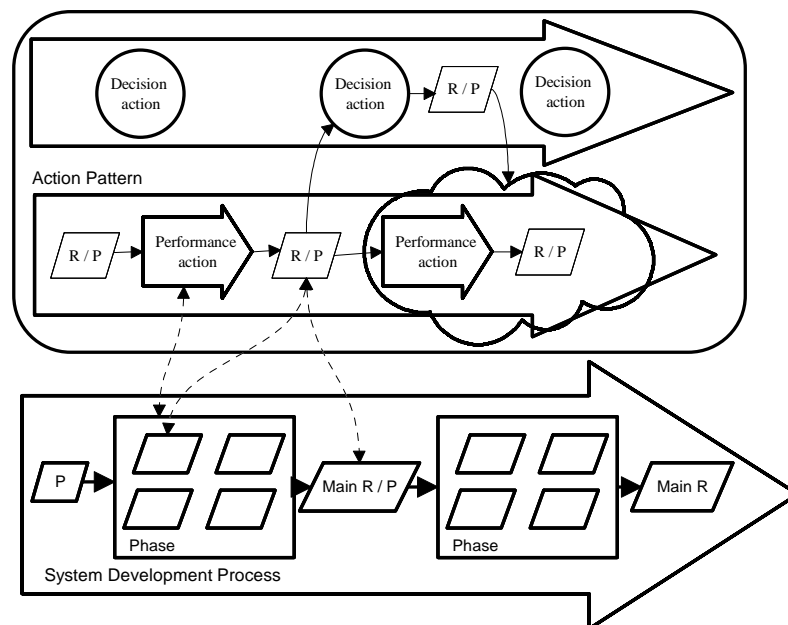

Figure 10: Method supported action patterns

In the figure above we can see that the prerequisites (P) or the results (R) in the performance level of the project-based action pattern correspond to different prerequisites (P) or results (R) that are accentuated in the SD-process. In the same way there is a correspondence between performance action and phases or sub phases in the SD-process. In the SD-project there are a lot of decisions made based on prerequisites/results in terms of what to do in order to fulfil different project goals. One part of this is situation adaptability of methods. Since we believe that situation adaptability is important, we also

believe that it must be possible to, in the SD-process, choose from different method components and to adjust methods depending on the situation (Goldkuhl, et. al, 1997). Depending on different project-based action patterns it should be possible to use the method components in different inquiry situations, where the SD-process gives possible method components to choose. In one inquiry situation one might not need to use all method components that are specified in each phase.

4.1.2 Action patterns and role interaction for project management

As discussed in previous parts of this paper, actions in a project can be viewed as patterns of actions. These patterns are in figure 4 (see section 3.4) divided into two different levels of practice – level of performance and level of decision. Through acts on the performance level different results are being generated. This is a consequence of an assignment. This assignment can for example execute a project. The achieved outcome from one action acts as basis for the continuously evaluation that occur at certain decision points during the different phases of this project. If we integrate the structural view of role interaction into this picture an even broader picture evolves and places in the foreground not only the action and the result but also the actor and the action relationships between the different interactive actors that together is performing the action in question. Figure 11 illustrate how Action patterns and Role interaction is linked in project management.
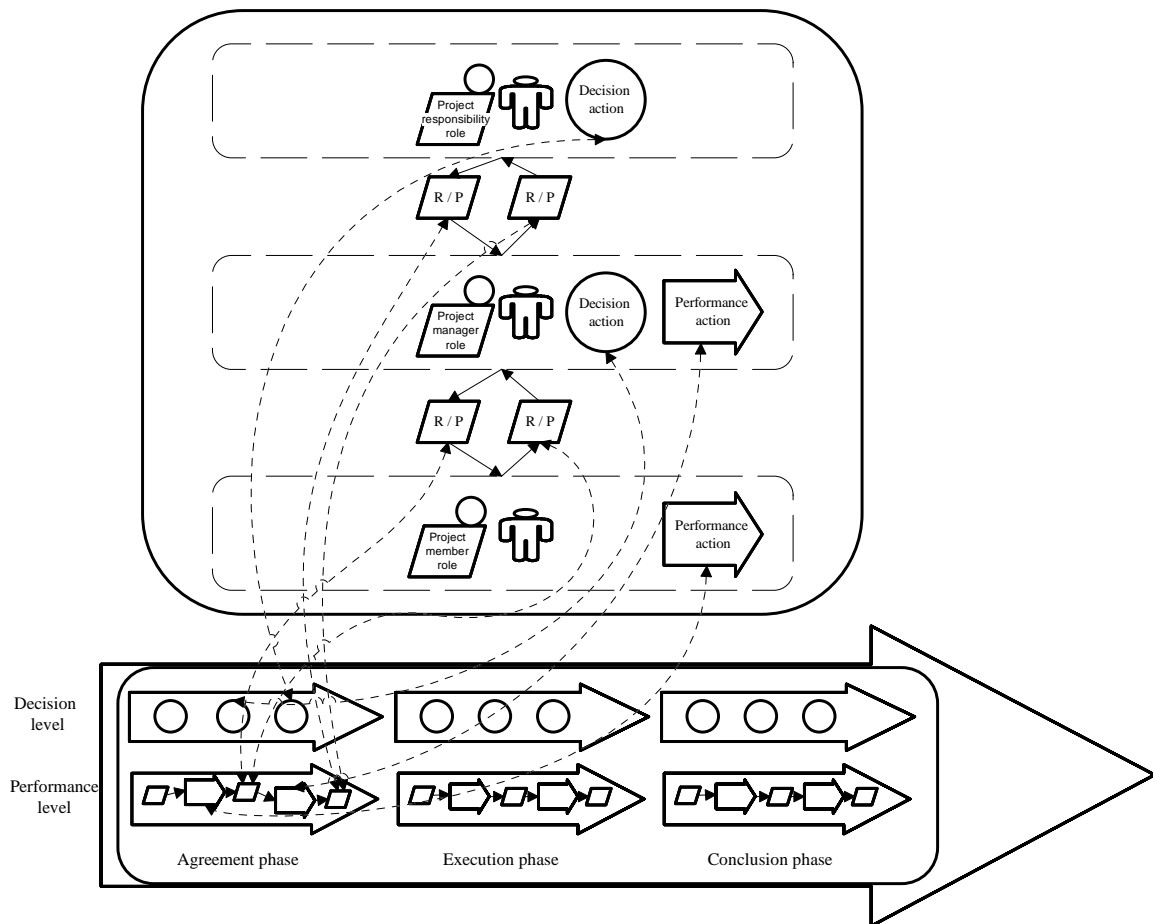

Figure 11: Action patterns and role interaction for project management

We will explain the figure by using a simple example. In the example we use the initial action phase in a project, *the Project Start-Up*, as a way to describe the logic behind the generic character of this integrated structural view. The purpose behind a Project Start-Up is for the project manager to Start-Up the project as rapidly, efficiently and with as many prerequisites established as possible (Archibald 1992). In the agreement phase on the performance level this imply that the actor assign the role *project manager* much seek out and identify prerequisites with the project. This means a lot of communication between different roles on management level - between the developer (SDP) and the client.

Suggestions on roles that are involved are the *project assignor*, the *project providers*, the *project sponsors* and the project manager. This communication are being performed through certain performing activities, with the purpose from the project managers viewpoint to be able to understand what goals, resources, needs and limitation that the project posses. Continuously decisions are being made on the decision level, which enables the work to move forward. In the final decision action point, in the agreement phase, the main result of this phase is established, the assignment to the project manager to Start-Up the project and by doing this achieve this actions main results a planned and organised project ready to execute the project assignment.

The assignment of the action Project Start-Up will then be carried out in the execution phase, which follows the agreement phase. The execution phase includes several performing activities for the project manager; such as identify key potential project members actors, define their responsibilities, capture their view on the forthcoming project, inform them on goals and the content of the project assignment in question, evaluate their ability, assure (by contract?) that they understand the role that they are assigned in the project, establish and secure responsibility for the project amongst the involved actors. During the execution phase the project plan is developed. This is the main result of the execution phase in the action Project Start-Up. And the completion of this artefact as well as the creation of support, accept and responsibility in the project organisation moves the action Project Start-Up over to the Conclusion phase.

In this final phase the result of the planing of the project is presented for the general management in an action of performance, with the base for decision as the action result. The following decision action is carried out by the general management as an evaluation of the base of the presentation (the Project Managers performing act) and the project plan as artefact. The result of this decision act is either a 'go' for the project as planed or a 'no go'; i.e. for example total cancellation or revision. Irrespective of character of decisions another phase divided – agreement-execution – conclusion - sequences is generated after the pronouncement of the decision.

4.1.3 Result: Giving SDP's a language based on structural views

By basing our self on social action we have been able to present three fundamental structural views; method, action pattern and role interaction. These three fundamental structural views has then been used in order to formulate two abstractions; one oriented towards producing agreed results (method supported action patterns) and one oriented towards role interaction based on action patterns (project management).

By combining these two abstractions we are able to present a language that can be used to talk about SDP's. This language is based upon the combined structure showed in the figure below.
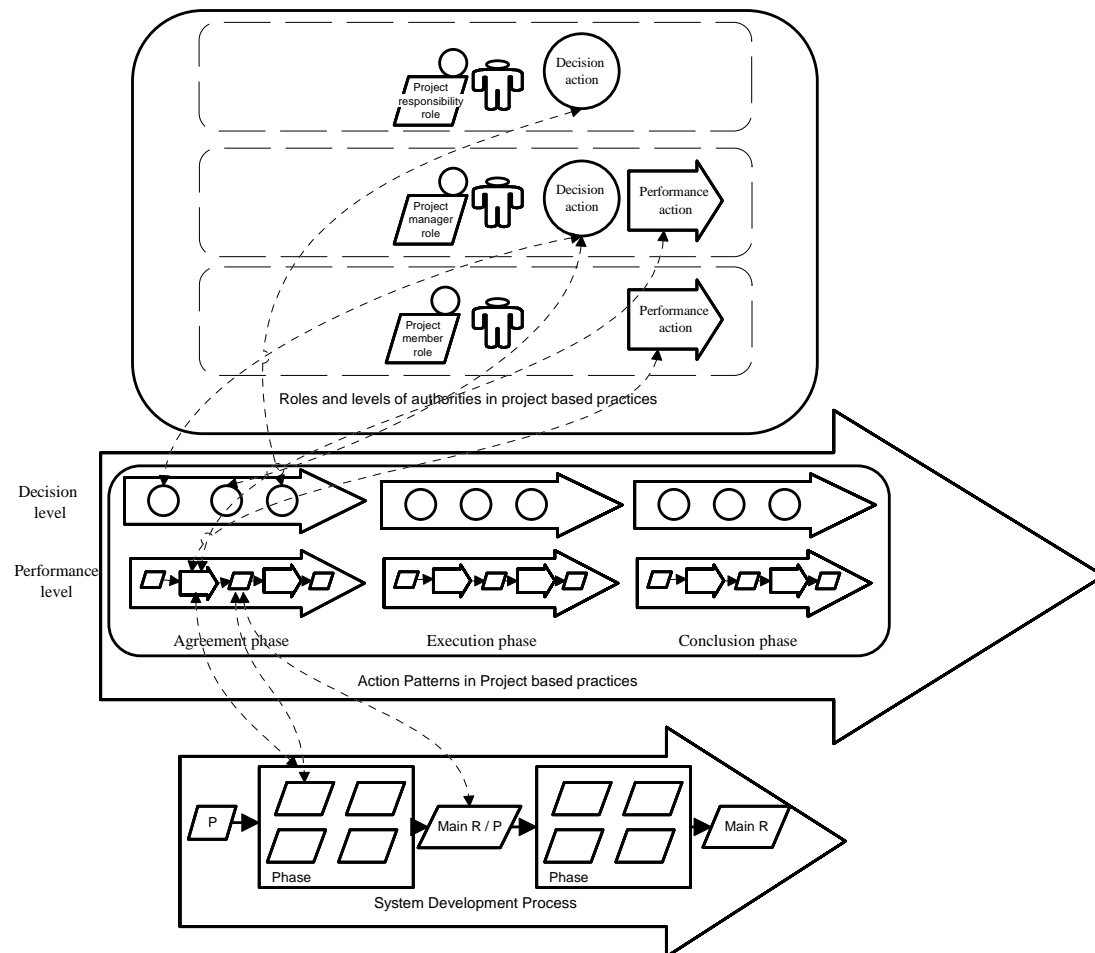

Figure 12: Essential parts in the language of system development practices

As depicted in the figure above there is a production level (bottom part of the figure), which is about how to produce an information system in an efficient way. Such production process is what is emphasised in system development processes. Important results generated from the system development process need however to be related to the action pattern with its results in the project structure (middle part of the figure). The project structure is a condition for generating prerequisites that are based on the expectations from the clients. Lastly, in order to ensure good production as well as good project performance there is a need to be clear about different roles and their interaction (the top part of the figure).

We believe that when talking about system development practices there is often a mixture between the different concepts depicted in the figure above. By presenting a structure that gives SDP's a language we believe that actors involved in the performance or development of SDP's have an easier way to understand each other.

5. Conclusions and further research

In this paper we have elaborated three structural views on SD in order to create a language and conceptualisation for SDP. This language and conceptualisation is based on a social action theory, speech act theory and three essential structural views on SDP: action pattern, method and role interaction.

Our experience from different projects where development of SDP's has been in focus, and as a result of this paper is that it is a necessity to have a language and a conceptualisation for SDP's as a whole when the goal is to develop a SDP of some kind. SDP's have specific characteristics and it is important not to just deal with these characteristics as isolated phenomenon during development

17

efforts. Instead it is important to see how these characteristics are related and how they affect each other. Our conceptualisation of SDP's includes a concept and category apparatus (language) and where we position this language as useful during development of SDP's.

In SDP's there exists an abundance of concepts, models and methods that are used during SD. There are for instance concepts, models and methods related to specific SD-actions, project management, organisational interaction etc. In this context we believe that one important contribution with this paper is the conceptualisation and language for SDP's that can be used for evaluation of what kind of support different concepts, models and methods can give during SD. If we do not make this clear we can some time tend to evaluate things that are unfair, or even impossible, to compare with each other. There are concepts, models and methods that just address things in one of the three structural views at the same time, as there are concepts, models and methods that address thing in more than one of these structural views. An example of a method that direct attendance towards both SD-process and project management is iterative application development (IAD), while an example of a method that just direct attendance toward the SD-process is RUP. IAD is a method invented by CAP Gemini. Therefore it is important to identify how different concepts, models and methods individually and in combination can contribute to the SDP, how they are related and how they affect each other in order to create the foundation for the possibility of understanding SDP as a whole.

We can now also see that there is an additional aspect of SDP's that would be worthwhile to explore. We have touched the aspect in earlier parts of this paper and it concerns business interaction, i.e. commercial aspects of system development. We believe that it would be possible to combine the thinking of the presented three structural views with a thinking of business interaction. Through such combination it would be possible to study which exchange processes that occur between producer and client and how actor relationships develop over time.

Other additional aspects to explore is how inter-organisational collaborations between different actors establishing the SDP affects the possibility to create an understanding about the SDP as a whole. When many more organisational units' collaboration our view is that the complexity regarding actor relationships, communication and co-operation in general becomes more difficult. For a short space of time several actors from different organisation come together in an effort to create value for a common client. Different organisational cultures, languages and norms how to act meets in a complex interplay. A language such as the one that has been presented in this paper probably could act as a common foundation in the effort to create an agreement and an understanding about how to conduct the temporal practice. However this type of inter-organisational form of practice has not been in the foreground in our empirical studies and further research must be conducted. Potential angels for different research purposes could be to try the language as such in an inter-organisational SDP context in order to test it use in this more complex environment. Other purposes could be to focus the director of the inter-organisational and how this actor acts in order to create a common understanding about the temporal SDP with or without a common language and to evaluate the effects that these different approaches imply.

Further research possibilities include the development of a framework for evaluating different methods and models in a present SDP. This framework should for example support the effort to assess how much project management aspects that a certain SD-method possess in relation to what its description state or how much a project management model can handle in supporting development or business creation activities. The framework should be suited for intra as well as inter-organisational SDP contexts and include components that make it possible to estimate different possible support structures in order to create an integrated structure for the specific situation.

References

Alvesson M., Skjöldberg K. (1994). *Tolkning och Reflektion* [*Interpretation and Reflection*, in Swedish], Stundentlitteratur, Lund.

Archibald R. D., (1992) *Managing high-technology programs and projects*, 2 ed New York : Wiley & Sons

Austin J. L. (1962) *How to do things with words*, Oxford University Press

Avison D. E., Fitzgerald G. (1995) *Information Systems Development: Methodologies, Techniques and Tools*, Mc Graw Hill, Berkshire, England

Barnard, C., (1938): *The Functions of the Executive*. Cambridge, MA Harvard University Press.

Boudon, R., (1986): *Theories of Social Change*. Cambridge: Polity Press

Brinkkemper S. (1995) *Method engineering: engineering of information systems development methods and tools*, Information and Software Technology, 1995 37

Clegg S., R., (1990): *Modern Organizations – Organisation Studies in The Postmodern World* Sage Publ. London

Durkheim, E., (1893/1984): *The Division of Labour in Society* London: MacMillan

Geary, J., (1995): „Work Practices: The Structure of Woek" I Edwards (Eds.) (1995): *Industrial Relations: Theory and Practice in Britain*. Oxford: Blackwell.

Giddens, A., (1984): *The Constitution of Society Outline of a Theory of Structuration*. Cambridge Press

Glaser, B., G., Strauss, A., L., (1967) *The Discovery of Grounded Theory* Weidenfeld and Nicholson,

Goldkuhl G. (1992) *Contextual activity modelling of information systems*, in Proceedings of "3rd int working Conference on Dynamic Modelling of information systems", Noordwijkerhout

Goldkuhl G., Lind M., Seigerroth U. (1997) *Method integration as a learning process*, in proc the 5th BCS Conference on 'Training and Evaluation of Methodology Practitioners and Researchers, 27th – 29th August 1997, Preston

Goldkuhl G., Ågerfalk P. J. (2000) *Actability: A way to understand information systems pragmatics*, Presented at the 3rd international workshop on organisational semiotics, 4/7-2000, Staffordshire, UK

Goldkuhl G., Lind M., Seigerroth, U. (Eds., 1998)   *The Language Action Perspective on Communication Modelling*: Proceedings of the Third International Workshop (LAP'98), Jönköping International Business School

Goldkuhl G., Lind M., Seigerroth, U., Ågerfalk P. (Eds., 1999) *The Language Action Perspective on Communication Modelling*: Proceedings of the Fourth International Workshop (LAP'99), Jönköping International Business School

Goldkuhl G., Röstlinger A. (1993) *Joint Elicitation of Problems: An Important Aspect of Change Analysis*, Presented at IFIP wg 8.2 Working Conference on Information Systems Development: Human, Social and Organisational Aspects, 17-19 May 1993, Noordwijkerhout, The Netherlands

Goldkuhl G., Röstlinger A. (1999) *Expanding the scope – from language action to generic practice*, In: Goldkuhl et al (1999)

Habermas J. (1984) *The theory of communicative action 1, Reason and the rationalization of society*, Beacon Press

Hedberg P., Lind M. (1999) *Talking about projects – a project generic model*, In: Enmark R. (Eds.) Reports on Business and Informatics, University of Borås

Hoffer J. A., George J. F., Valacich J. S. (1999) *Modern System Analysis & Design*, Addison Wesley Longman, inc

Humphrey W. S. (1997) *Introduction to the Personal Software Process*, Addison Wesley Longman, inc

Jayaratna N. (1994) *Understanding and Evaluation Methodologies – A Systematic Framework*, McGraw-Hill, London

Lind M. (2001) *Från system till process – kriterier för processbestämning vid verksamhetsanalys* [*From system to process – criteria for process determination at business analysis*, in Swedish], forthcoming PhD-thesis

Lind M., Goldkuhl G. (2001) *Generic patterns for business modelling*, Submitted to Action in Language, Organisations and Information Systems (ALOIS-2001)

Löwstedt, J (1995): *Människan och strukturerna – Organisationsteori för förändring* [*The man and the structures – Theory of organisation in order to change*, in Swedish] Nerenius & Santerus Förlag

March, J,. Simon, H., (1958): *Organizations*. New York: Wiley & Sons

Medina-Mora R., Winograd T., Flores R., Flores F. (1992) *The Action Workflow Approach to Workflow Management Technology*, In Turner J., Kraut R. (Eds.) Proceedings of the Conference on Computer-Supported Cooperative Work, CSCW'92, ACM Press, New York

Morgan G. (1997) *Images of organization*, SAGE Publications Inc., Thousand Oaks

Murch R. (2001) *Project management – best practices for IT professionals*, Prentice Hall, Upper Saddle River

Piaget, J (1968): *Le structuralisme*, 3 éd., mise à jour. Paris

Reijswoud V. E. van (1996) The *Structure of Business Communication: Theory, Model and Application*, Doctoral Dissertation, Delft University of Technology

Röstlinger A., Goldkuhl G. (1994) *På väg mot en komponentbaserad metodsyn* (in Swedish), Presented at conference "VITS Höstseminarium 1994", Linköpings Universitet

Scott, R., W., (1998): *Organisations – rational, natural and open systems* – 4 ed. Prentice Hall, Upper Saddle River

Searle J. R. (1969) Speech acts. An eassy in the philosophy of language, Cambridge University Press, London

Walsham G. (1995) Interpretive case studies in IS research: nature and method. Operational Research Society Ltd.

Weber, M., (1983): *Ekonomi och samhälle : förståendesociologins grunder, 1 : Sociologiska begrepp och definitioner. Ekonomi, samhällsordning och grupper* Lund, Argos.

Wilson, F., (1999): *Organizational behaviour : a critical introduction* New York : Oxford : Oxford University

Winograd T. , Flores F. (1986) Understanding Computers and Cognition: A New Foundation for Design, Ablex, Norwood NJ

Yeates D., Caddle J. (1996) Project Management for Information Systems, 2nd edition, Pearson Professional Limited