# FLEXIBLE INFORMATION SYSTEMS – AN ACTIVITY-BASED VIEW

*Lars Bækgaard*
Aarhus School of Business, Aarhus University
Fuglesangs Alle 4, DK-8210 Aarhus V, Denmark
e-mail: lab@asb.dk

## 1 INTRODUCTION

In this paper we discuss the flexibility of information systems. It is imperative that such systems can be changed in ways that maintain their business relevance. We discuss flexibility in terms of the business activities that are performed and supported by an information system. We focus on business activities because these are essential for a business' ability to serve its customers.

Many systems development methods are activity-based in the sense that they recommend that information systems development should be based on an understanding of existing and future business activities. Structured Analysis (De Marco 1978) and ISAC (Lundeberg et al. 1978) are early examples of methods that use graphical flow models to represent current and future work activities. Multiview is a method that combines thorough activity analysis by means of Soft Systems Analysis (Checkland 1981) with Structured Analysis (Avison and Wood-Harper 1990).

Various techniques can be used to support activity modeling. Activity-based design is a method that models work actions and their mutual dependencies (Andersen 2006). Action-based modeling is a modeling technique that is based on the language-action perspective (Ågerfalk and Eriksson 2004). A use cases represents a coherent functional unit of a system (usually an IT-system) that offers an information service to actors in its environment (Cockburn 2001). A task description can be viewed as an extension of a use case that focuses on the actor's work activities (Lauesen 2003). An activity cases goes a step further and focuses on a selected activity that may or may not involve IT-systems (Bækgaard 2005; Bækgaard et al. 2007).

When viewed from a flexibility and change perspective all the above-mentioned methods and techniques share a common deficiency. They do not offer a business level representation of the structure of information systems and IT-systems that enables a discussion of change in terms of business concepts. They use work activities in different ways as foundations of the development of information systems and IT-systems but they do not recommend that the structure of the systems should reflect the activities they perform or support.

The paper is based on the observation that an information system must be changed continuously and fast in order to maintain its business relevance. Furthermore, the paper is based on the assumption that this presumes that the systems are structured in terms of relevant business concept. When a business decides to change an information system the decision is made in business terms like actors and activities. If the information system is not structured around such concepts there will be a gap between the business intentions about change and the actual information system changes that has to be done. For example, contemporary development methods consider object-orientation to be state-of-the-art in systems development (Kruchten 2000; Mathiassen et al. 2000). The consequence is that software is structured in terms objects that are not as such based on business concepts. We suggest and argue that information systems should be structured in terms business actions. Briefly, the main reason is that the establishment and maintenance of a business' action repertoire is essential for its economical survival.

## 2 ACTIVITY SYSTEMS

In this section we characterize business systems as activity systems in which human beings and technology performs activities. Also, we characterize IT-systems as activity systems in which human

beings and information technology performs information activities. And we characterize IT-systems as activity systems in which information technology performs information activities.

*BUSINESS SYSTEMS*

Activity systems are systems in which actors perform actions in order to create value for customers and owners (Checkland 1981). The value can be created in terms of products or services (Alter 2006). Activity systems are composed of related actors, things, and information. We use the term actor as a reference to both human beings and IT-systems even though we realize that human beings and IT-systems perform actions in very different manners (Rose et al 2003). Actors perform activities and interact with each other. Actors create and exchange material objects and information objects with or without the support of tools. Actors use information. Actors can participate in events.

Business activities deal with movement, manipulation, and consumption of material objects and information objects and they deal with coordination in terms of requests for, agreements about, control of, and evaluation of work activities (Denning & Medina-Mora 1995).

We view an activity is a set of coherent actions that are directed towards some goal. An action is an atomic unit of activity that is performed by an actor. Atomicity is not an absolute concept. It means that at a given level of abstraction it is not necessary to subdivide the action into even smaller sub actions.

Many actions involve more than one actor because interaction between business parties play important roles (Goldkuhl 1996; Dietz 2006). For example, the ordering of an item in an online store can be viewed as a communicative action that involves a customer and the business. Consequently, actions can be viewed as dynamic relations between elements in a business system. The ordering of an item can be viewed as a dynamic relation between a customer and a business.

Activity systems do not exist per se as hard facts (Checkland 1981). They exists as social constructs that allow us to emphasize certain characteristics. The activity system perspective is particular useful for information systems because such systems contain a combination of people and artifacts. And it makes it possible to use the same vocabulary to describe business systems (including information systems and IT-systems), information systems (including IT-systems), and IT-systems.

*INFORMATION SYSTEMS*

We view an information system as an activity system in which all activities are information activities. They register, store, manipulate, and present information about a domain of interest to actors in support of their activities (Checkland & Holwell 1998). Information systems play important roles in material and coordination activities as well. Material activities may be mediated by digitally controlled machinery and many coordination activities are communication activities in which actors express requests, requirements, contracts, and evaluations. Many information systems operate in ways that are deeply integrated in material business activities. They are part of the business activity system rather than merely systems that support business activity.

An information system can perform important business activities and it can support business activities that are performed by other information systems or by human actors. For example, a data warehouse may deliver information about customer behavior in support of marketing activities. Or, a chat system may support communication between customers and employees.

Information systems are much more than IT-systems. In all businesses vital information activities are performed by human beings. And the information activities that are performed by IT-systems are planned and programmed by human beings.

*IT-SYSTEMS*

An IT-system system is an information system in which all actors are IT-actors. An IT-actor is a coherent unit of software and hardware. It is composed of sensors, actuators, memory, and activities. An IT-actor uses sensors to register information about its environment in terms of selected events, measurement values etc. An IT-actor uses its memory to remember information about selected aspects of its history. An IT-actor can influence its environment by means of actuators that delivers information, turns switches of/on etc. An IT-actor can perform activities in terms of transformation of information, movement of information etc.

Advanced IT-actors are used in many computer games, internet-based systems, and business intelligence systems. Even the simplest IT-functions can be viewed as IT-actors. Therefore, the notion of IT-actors is general enough to cover both the simplest and the most advanced IT-systems. Use cases can be used to specify reactive IT-actors but they cannot be used to specify proactive IT-actors like web site crawlers or business intelligence monitors.

*DISCUSSION*

The notions of activities and actions can be used to discuss the operational roles of IT-systems and information systems in a manner that is relevant from a business perspective. We view information systems (and IT-systems) as activity systems that perform information activities in businesses.

Businesses create value by performing relevant actions. They create products, offer services, facilitate activities etc. A business have a repertoire of actions that it can easily perform. This repertoire is supported by the structure of the business and the capabilities of its actors. For example, a hospital can perform activities like surgery and medication and a car factory can assemble cars.

Our perspective on information systems and their business context is summarized in Figure 1. We view IT-systems, information systems, and business systems as activity systems in which actors carry out activity. Business systems contain information systems in the sense that parts of business may be viewed as an activity system that performs information activities. Information systems may contain IT-systems that carry out information activities.

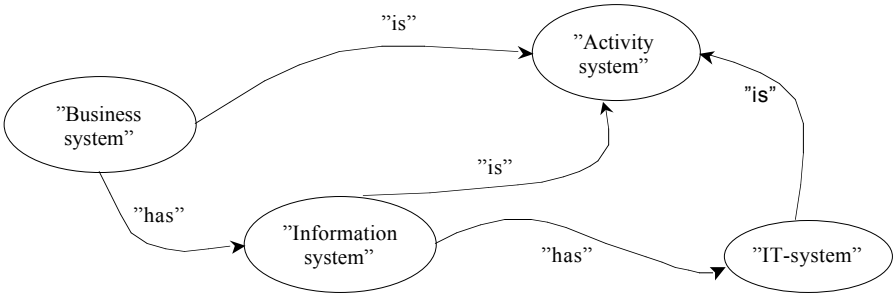


Figure 1 Information systems in context

A business must continuously change in order to maintain the relevance of its action repertoire. The business can use a combination of two strategies to ensure such maintenance. It can adapt to its environment and it can seduce its environment by offering new or improved activities. A significant subset of a business' activities are performed by information systems. These activities must continuously be changed as a part of the change of the business. A significant subset of an information system's activities are performed by IT-systems. These activities must continuously be changed as a part of the change of the business.

## 3 ACTIVITY-BASED DEVELOPMENT

In this section we use the notions of actions and activities to discuss flexible development processes. A continuous change process is necessary in order to maintain the relevance of the business (Bækgaard 1990; Truex et al. 1999). The change process must focus on the IT-systems, the information systems, and work activities in other parts of the business.

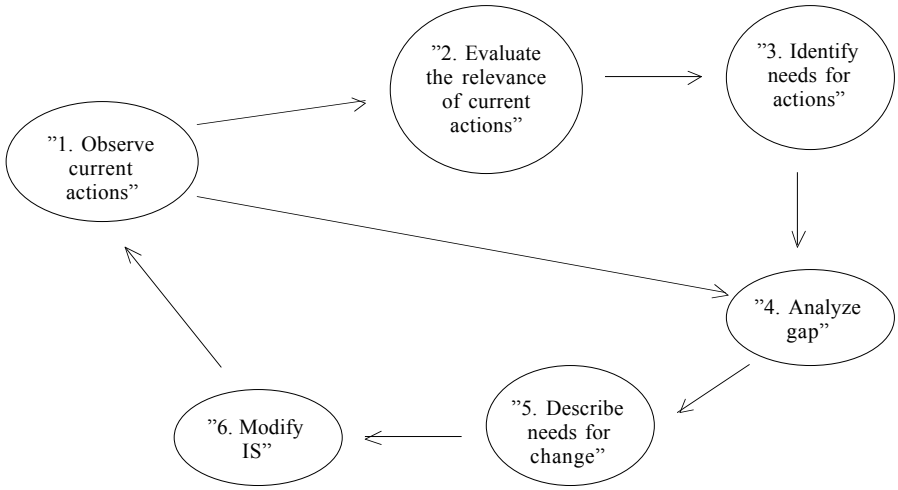


**Figure 2** Activity-based development activities

Figure 2 illustrates the idea of activity-based development in terms of six activities. The current activities in a business are observer and their relevance evaluated on a continuous basis. Based on the evaluation (and creative thought) the needs for business actions are identified. These needs are compared with the current business activities and the gap between the as-is situation and the desired to-be situation is described in terms of changed, canceled and new activities. Finally, the information system at hand is modified according to the described needs for change (often non-information activities have to be changed as well).

Since activity 1-5 are done in terms of business activities it would be desirable if activity could be done in terms of activities as well. An IT-system may be based on standard software or customized software. In both cases the basic question is the same: What is an appropriate unit of change?

We suggest that the notion of an action is an appropriate unit of change. Actions can be composed to activities in many different ways. DEMO is a business ontology in which business activity is viewed as a network of transactions (Dietz 2006). Each transaction represents an interaction between an actor that requests something and an actor that responds to the request. BAT is a business ontology in which business activity is viewed as interaction between business actors (Goldkuhl 1996; Goldkuhl and Lind 2004). Event-based modeling is a technique that uses events (short-duration, shared action) to structure and control business activity (Bækgaard 2001; Bækgaard 2004; Bækgaard 2007).

Service-orientation represents an organizing principle in which everything that is offered by a business is thought of as a service (Papazoglou and Dubray 2004; Barros et al. 2005; Roman et al. 2005). A service can be viewed as the performance of work by one party for some other party (OASIS 2006). A service has a provider that offers the service, an executor that executes the service and a consumer that benefits from the service. The execution of a service can be viewed as series of actions that are performed by the executor and the consumer.

Actions can be viewed as dynamic relations between elements in activity systems (business systems, information systems, IT-systems). An information systems action repertoire determines the types of information activities the information system can perform. If an information systems is explicitly structured as an action repertoire different activities like transactions, business interactions, and

services can be composed from this repertoire. This indicates that in order to obtain flexibility the architecture of an information system should not be based on a composite action ontology like services but on the actions themselves. On top of an action layer composite actions like services.

## REFERENCES


Alter, S. (2006). The Work System Method. Connecting People, Processes and IT for Business Results.

Andersen, P. B. (2006). "Activity-Based Design." European Journal of Information Systems **15**: 9-25.

Avison, D. E. and A. T. Wood-Harper (1990). Multiview, Blackwell Scientific Publications.

Barros, A., M. Dumas, et al. (2005). Service Interaction Patterns: Towards a Reference Framework for Service-Based Business Process Interconnection, Queensland University of Technology.

Barros, A. P. and A. H. M. ter Hofstede (1998). "Towards the Construction of Workflow-Suitable Conceptual Modeling Techniques." Information Systems Journal.

Bækgaard, L. (1990). Designing Adaptable Software - Parametrization of Volatile Properties. Conference on Software Maintenance. San Diego, California, USA.

Bækgaard, L. (2001). Event Modeling. Information Modeling in the New Millennium. M. Rossi, Siau K., Idea Group Publishing.

Bækgaard, L. (2004). Event-Based Activity Modeling. ALOIS'04 - Action in Language, Organisation and Information Systems. Linköping, Sweden.

Bækgaard, L. (2005). From Use Cases to Activity Cases. ALOIS'05. Action in Language, Organisation and Information Systems. Limerick, Ireland.

Bækgaard, L. (2007). Event-Based Information System Models. ICEIS'07 - 9th International Conference on Enterprise Information Systems. Funchal, Madeira, Portugal.

Bækgaard, L., J. B. Jørgensen, et al. (2007). On Industrial Use of Requirements Engineering Techniques. EMISA'07, 2nd International Workshop on Enterprise Modelling and Information Systems Architectures. St. Goar, Germany.

Checkland, P. (1981). Systems Thinking, Systems Practice, Wiley.

Cockburn, A. (2001). Writing Effective Use Cases, Addison-Wesley.

De Marco, T. (1978). Structured Analysis and System Specification. Yourdon., Yourdon.

Dietz, J. L. G. (2006). Enterprise Ontology. Theory and Methodology, Springer-Verlag.

Goldkuhl, G. (1996). Generic Business Frameworks and Action Modeling. First International Workshop on Communication Modeling. Tilburg, The Netherlands.

Goldkuhl, G. and M. Lind (2004). The Generics of Business Interaction - Emphasizing Dynamic Features Through the BAT Model. LAP´04. International Working Conference on the Language-Action Perspective on Communication Modelling New Brunswick, NJ, USA.

Kimble, C. and W. Selby (2000). An Interdisciplinary Study of Information Systems: Christoffer Alexander and IS Failure. 5th UKAIS conference.

Kruchten, P. (2000). The Rational Unified Process - An Introduction, Addison-Wesley.

Lauesen, S. (2003). "Task Descriptions as Functional Requirements." IEEE Software **20**(2): 58-65.

Lundeberg, M., G. Goldkuhl, et al. (1978). Systemeering, Studentlitteratur.

Mathiassen, L., A. Munk-Madsen, et al. (2000). Object-Oriented Analysis and Design, Marko.

OASIS (2006). Reference Model for Service Oriented Architecture 1.0, OASIS.

Papazoglou, M. P. and J.-J. Dubray (2004). A Survey of Web Service Technologies, University of Trento.

Roman, D., U. Keller, et al. (2005). "Web Service Modeling Ontology." Applied Ontology **1**: 77-106.

Sowa, J. F. and J. A. Zachman (1992). "Extending and Formalizing the Framework for Information Systems Architecture." IBM Systems Journal **31**(3590-616).

Truex, D. P., R. Baskerville, et al. (1999). "Growing Systems in Emergent Organizations." Communications of the ACM.

Ågerfalk, P. J. and O. Eriksson (2004). "Action-Oriented Conceptual Modelling." European Journal of Information Systems **13**(1): 80-92.