A Cognitive Framework for Knowledge-based Process Design

Hans VAN LEIJEN
NOTION, Universiteit Nyenrode
Breukelen, the Netherlands
h.vleijen@nyenrode.nl

and

Walter R.J. BAETS
NOTION, Universiteit Nyenrode
Breukelen, the Netherlands
w.baets@nyenrode.nl

ABSTRACT

We propose a framework for redesigning knowledge-intensive business processes that is inspired by the knowledge-based theory of the firm, and based on ideas from cognitive science. It views a business process as a problem-solving task consisting of five phases: recognition, decomposition, planning, action and evaluation. The coordination of these tasks among multiple agents is viewed as distributed cognition. We give some general principles for identifying process improvements based on manipulating these phases.

Keywords: Business process reengineering, business process improvement, knowledge management, knowledge-based theory of the firm, scripts.

1. INTRODUCTION

Business Process Reengineering [10], [6], is the restructuring of tasks in an organization in such a way that information technology is optimally exploited. In the first half of the nineties, it was proposed as a radical, clean-sheet approach to business transformation. Currently, its methods and tools are applied in an incremental fashion.

We focus on knowledge intensive processes in administrative service organizations. Because of advances in information technology, business processes will shift increasingly from labor intensive to knowledge intensive. Eppler et al. give defining attributes of complex and knowledge-intensive processes [7]. Knowledge intensiveness is characterized by highly contingent outcomes or dependency on chance events, many possibilities an agent has to choose from, necessity of creativity in solving problems, quick obsolescence of knowledge, dependency of performance on agent skill, and long learning time. They distinguish an additional complexity dimension for business processes: the number of steps and the number of agents involved, interdependencies between steps, and the importance of the order in which steps are performed. In this paper, we take knowledge-intensiveness to include process complexity.

Traditionally, BPR has been approached from two angles [6]. The top-down, "strategic" approach from the management literature tries to improve certain performance measures by offering step plans and guidelines (see [20] for an overview). The bottom-up, "technical" approach from the information systems field, on the other hand, offers modeling formalisms to represent and analyze existing and future processes.

We believe there is a need and a means for integration of theories from the management and technological literatures. A growing number of authors have started to develop a middle ground between top-down and bottom-up process reengineering by systematically gathering, integrating and evaluating redesign principles and heuristics. Buzacott [3] formalizes the nine redesign principles from [10] in terms of queueing theory, and evaluates them. Nissen [18], in pursuit of a knowledge-based decision support system for BPR, describes a graph theory-based framework for analyzing processes, that is able to incorporate experiential knowledge of redesign heuristics. Kock and Murphy, in [12], give a set of redesign principles based on manipulating information and knowledge exchange. Lastly, Reijers [21] gives a comprehensive collection of heuristics gathered from the literature. These principles and heuristics give good hints regarding where to look for improvements in business processes, and we believe they can be used to substantiate step plans and direct modeling efforts.

Yet, this work on redesign heuristics is ambivalent about *what it is* that is being redesigned. A business process is not just a structured set of activities that can be expressed exhaustively in a graphical formalism, for if it was, then reengineering would be a purely analytical exercise. For example, while [18] uses a graph formalism to express process transformations with, his system is complemented with the ability of using redesign heuristics to incorporate "experiential knowledge". Also, the bodies of heuristics in [12] and [21] do not even use a formal structure for describing the business process being redesigned. In practice, a business process is a constellation of both humans and machines, both performing strict procedures and doing intelligent improvisation. In order to accommodate reengineering of knowledge-intensive processes, and in order to be more explicit about the object of the reengineering, we use concepts from organizational cognition and learning for the analysis and manipulation of process knowledge.

The solution that we propose is to conceptualize the handling of a case in a business process as solving a problem, and the organization is seen as a goal-directed planner who has to exhibit the right response in reaction to a stimulus. This conception of intelligent behavior was first set out by Newell and Simon in [17]. We describe a business process as being built up of five phases: recognition, decomposition, planning, action, and evaluation. We call this model the *cognitive cycle*. People use *scripts*, patterns of events stored in memory, to be able to execute the cognitive cycle. Each of the five phases can be decomposed recursively into cognitive cycles at lower levels of abstraction.

Our approach resembles Manheim's *Cognitive Informatics* [16]: "the use of knowledge about how people think and act to design information technology (IT) support that enhances the way people think and act." Where that paper uses *patterns*, we use *scripts* to describe the contents of human experiential knowledge. Scripts might become a unifying concept in disciplines related to knowledge and processes, such as BPR, knowledge management and Workflow Management.

We base our model on a knowledge-based theory of the firm [9]. This theory states that the most important role of firms is to integrate specialist knowledge. This is implemented by coordinating specialist competencies using coordination mechanisms. To use these, specialists must have *shared knowledge*: knowledge about someone else's knowledge. The cognitive cycle is therefore matched by models of learning and coordination.

We will then use this model to argue for the view that a process should be seen as an abstraction of coordinated competencies, and therefore, that the process design task is not about designing activities but about designing task competencies, coordination mechanisms to integrate task competencies and shared knowledge to inform the use of those coordination mechanisms. We thereby offer this model as a view on process design that is congruent with knowledge-based theories of the firm and that draws closer ties between process management and knowledge management.

This paper is structured as follows. First, we describe knowledge-based and cognitive theories of the firm. Then we describe the cognitive cycle, consisting of a description of knowledge, learning and coordination. As an illustration of its practical use, we give the outline of a framework for identifying process improvements. We conclude with a discussion of related work, conclusions and future research.

2. KNOWLEDGE IN ORGANISATIONS

The knowledge-based theory of the firm

Grant proposes a theory that explains the structure and behavior of the firm based on its use of knowledge [9]. We use this theory as a point of departure to integrate the top-down and bottom-up views on BPR, based on business economics and information technology, respectively [6].

Grant gives a number of features of knowledge, three of which are relevant for our purposes. Firstly, knowledge is hardly transferable between agents, unless made explicit. Secondly, knowledge elements are hard to combine because of their context specificity: to be able to cooperate, specialists must transfer part of their knowledge to each other. Thirdly, it is necessary to specialize in the acquisition of knowledge because of cognitive limitations: we cannot all know everything.

According to the knowledge-based theory of the firm, integration of knowledge for production is the most important task of the firm. However, knowledge transfer is too expensive as a means of knowledge integration. The key to efficiency is therefore to distribute knowledge and design *coordination mechanisms* in such a way that the necessity to transfer knowledge is minimized.

For these mechanisms to function, *shared knowledge* has to exist: knowledge about other people's knowledge, and shared beliefs. The importance of shared knowledge is that it enables employees to integrate knowledge that *is not* shared.

The cognitive theory of the firm

Nooteboom takes the description of knowledge in organizations one step further in [19]. Based on the work by Gioia and Poole in [8], he explores the notion of scripts to describe the process of knowledge creation in organizations.

A cognitive agent has structures for *perception*, *interpretation* and *evaluation*. Data enters the agent via perception, and is subsequently transformed into *information* by *interpretation* in terms of knowledge structures. *Understanding* transforms information into beliefs and causal insights. *Knowledge* is a set of information that enables the agent to transform understanding into *action*, which delivers *performance*. This performance is the subject of improvement in BPR.

A *script* is a sequence of nodes; each node represents a class of events or a class of possible actions of the agent. Such a sequence can be temporal, logical or causal. A script is a pattern that has been abstracted from a set of perceived event sequences. It is a logical framework for categorizing events: a schema for both understanding and behavior. To understand is to be able to interpret an event as a node in a script, because that makes the near future and behavior expected from the agent somewhat more predictable.

A firm has a cognitive identity: a set of scripts in which people are nodes or contribute actions to nodes. This set has a recursive structure: a script can be a node in another script. The total script memory is distributed across people, documents and computer programs. No one person in the firm knows all the scripts, although it is part of the task of management to keep an overview.

Conclusions

We have made the knowledge concept in the knowledge-based theory of the firm more precise using the semi-formalized *script* concept. This might enable us eventually to replace the *activity* concept in BPR with the script concept, but the latter concept has yet two shortcomings. Firstly, there is an inconsistency between knowledge-intensiveness and scripts: the former signifies dynamics and variability, while the latter seems to refer to a fixed sequence of steps.

Second, the theoretical framework described needs to be translated into methods and tools, in order to enable process engineers to apply the script concept in their work. The script concept as it stands is too broad; we need to elaborate it specifically for the abstraction level of business processes.

In the next section, we view executing a business process as goal-directed planning to restore the dynamics of knowledge and to formalize the script concept one step further.

3. KNOWLEDGE AND LEARNING

We present a simple model of knowledge and learning inspired by work in artificial intelligence. To choose a cognitive model means speaking in terms of mental representations and problem solving behavior: planning a path of interventions in an object world to reach goals. In the service organizations that we focus on, the goal always is an action that serves or protects the interests of a stakeholder, such as a customer, an employee or the organization itself.

Knowledge is conceived of as a representation of behavior of things in the world; those representations enable an agent to find adequate plans. A representation takes the form of Schank and Abelson's script [23]: a contingent pattern of events in the past as perceived by the agent in a set of scenarios. The difference with Gioia's and Nooteboom's script, however, is that according to Schank, a script is not executed directly, but is a causal model on the basis of which plans can be constructed for execution. The particular plan constructed depends on the specific values substituted in a script and the particular goal the agent has in mind.

Knowledge

Each agent in the organization has one or more competencies to transform information in such a way that it eventually leads to appropriate actions, if coordinated successfully with other agents' actions. For example, in a car insurance company, a claims handler knows how to send orders to a towing company to assist customers, but also has to balance the interests of the customer with the interests of the company by asking an inspector whether damage inspection is necessary.

An agent has a memory of *scenarios*: successions of states of the world in the past, or stories. Such a succession is described in terms of *variables* and their values. This story describes behavior of part of the world and the agents' interventions in that part. A *causal model* is a regular pattern perceived by the agent in a set of scenarios. It is contingent and non-deterministic: it describes probabilities of possible courses of events depending on specific values of variables. In the insurance example, a scenario might consist of a notification of a loss, subsequent decisions made by the claims handler, and eventual outcomes in terms of customer satisfaction, damages paid for and apparent fraudulent behavior.

A situation presents a problem for the agent to solve. Being confronted with a problem, the agent will try to remember scenarios that resemble the present one; if those are found the agent will try to find or construct a causal model that predicts the near future in terms of probable paths of events. This is the first phase of our model: *recognition*.

Information about the problem is usually incomplete. The agent will therefore try to construct a more accurate picture of the problem by collecting extra information. This is our second phase: *decomposition*. Recognition and decomposition alternate until the agent is sufficiently confident that the problem matches the model constructed. For example, the claims handler will inquire about the presence of any bodily injuries, and if so, will ask more questions to be able to choose the type of assistance required.

A causal model consists of observation nodes and action nodes: action nodes are the interventions the agent might choose to perform. The next phase is *planning*: trying to find an adequate path of interventions. For this, memory is searched for previous intervention paths associated with similar situations, which are then judged for their consequences in the present situation (plan retrieval). If a suitable plan is not found, the agent can try to find an adequate path by searching the space of all possible intervention paths in the causal model constructed and judging each on the benefits of its consequences. However, this is an expensive combinatorial operation. An example of dynamical search might be when the car has multiple damages and the claims handler has to think about the order in which to undertake repair actions.

While judging proposed plans, hitherto unexpected consequences might crop up in the agent's mind, possibly necessitating a return to *recognition*.

Remaining steps are *action* and *evaluate / register*. During the former, the agent will execute the plan and recursively encounter sub-problems to solve. After executing the plan, the agent might store the new scenario in its memory, and may adapt representations to account for success or failure. In the insurance case, an example is registering the case in an information system for subsequent actuarial analysis.

The cognitive cycle can be recursively refined in all its five phases; that is, executing one phase may result in executing all five on a lower level of abstraction. For example, in a process such as bidding, *recognition* might be implemented by searching a library for all bids made in the last couple of years, or by bringing people together for a meeting (which goes for *evaluation*, too). *Decomposition* is a complex process in itself in, for example, medical diagnosis. *Action* will usually lead to sub-problems requiring a solution. And *register* might entail using complex information systems or taking measurements.

If confronted with failure during plan execution, the agent has to perform backtracking and restore or repair results of executed interventions, and then construct an alternative plan.

An agent will usually not carry out the process sequentially but will interleave phases to increase efficiency. For example, decomposition will often be skipped assuming the problem is "typical". This increases efficiency but also risk of failure. If decomposition is postponed too much, backtracking will be more frequent.

Learning

Learning means improving one's performance in one of three ways: learning new intervention alternatives, learning resultant behaviors of the outside world, and planning actions quickly and accurately. Learning can be done by knowledge transfer or by feedback on the results of interventions. Knowledge transfer is subdivided in interpreting messages and imitating someone else's behavior. This is probably the simplest way of learning but yields shallow representations that are applicable only in a relatively small class of situations.

Transferring knowledge in a message is done by verbalizing causal models. Such a message has three parts: a characterization of a class of situations, a set of intervention alternatives, and a description of resultant behavior of the outside world. The sender of such a message has to determine the size of the class of situations in which this representation is valid. This in turn depends on the amount of context that sender and receiver share. If the sender is unsure about this, he or she has to weaken the message by limiting the class of situations or by weakening the probabilities on the resultant behaviors.

For example, in a car insurance company, John might tell Mary: "drivers under age 24 are reckless; rejecting them protects our loss ratio". Upon second thought, John might limit the class of applicable situations: "*male* drivers under age 24…" Alternatively, John might believe Mary can decide for herself but weaken the message by weakening the probabilities expressed: "*some* drivers under age 24 are reckless; rejecting them *might* protect our loss ratio".

Learning by feedback works by observing consequences of interventions and updating representations accordingly. Based on the model of knowledge described previously, the following learning processes can be discerned:

- Learning statistical relations between variables by collecting statistics.
- Abstraction: decoupling a causal model from the set of scenarios it originated from by introducing abstract variables to summarize concrete ones.
- Compilation: neglecting plan judgment based on observed frequency of chosen plans in relation to problem features. This improves speed but increases the risk of choosing the wrong plan, and decreases flexibility of adapting plans to specific situations.
- Discovering new variables to observe and new interventions to perform. This requires creativity and experimentation and is considered a black box in our framework.

Using a causal model that has been built from a message (knowledge transfer) will be slow until it is sufficiently compiled. For example, the first time Mary encounters a prospective customer under age 24, she has to explicitly consider the effect of accepting the customer on the loss ratio, before she can decide to reject him or her. After repeated encounters however, she will simply recollect a suitable plan (rejection) without judging its consequences. This speeds up her process but makes it less flexible, since she is not anymore able to weigh in circumstances unique for the case at hand.

Learning is generally performed from the outcomes of interventions and can be done at two times: at evaluation time in the cognitive cycle, when the outcome becomes known (proactive learning), or, when a new problem presents itself and triggers recognition (lazy learning). The amount of laziness in learning behavior is a design decision that affects process adaptiveness.

Coordination as distributed cognition

Because of cognitive limitations of man and machine, competencies will have to be distributed over multiple agents. Intervention paths will have to be composed by integrating multiple agents' competencies. Because a customer is also an agent, any action the organization performs needs integrating at least two competencies. We call integrating competencies *knowledge coordination*.

Traditionally, research into coordination focuses on tasks that are largely known beforehand (see for example [15]). By contrast, in knowledge-intensive tasks, coordination is an expensive process of dynamic search.

Seen from the individual, coordination consists of the following steps. Confronted with a problem, a knowledge *seeker* first has to realize it doesn't have the competence to handle it. Then a knowledge *provider* has to be found that might have the right competence. The two have to reach agreement about the adequacy of their mutual competencies, and have to agree on how to integrate the provider's competence into the seeker's intervention path. A crucial criterion here is how much knowledge to transfer back and forth.

This can be likened to the language/action perspective of Winograd and Flores [27]. They discern conversations for action, and for clarification, possibilities, and orientation. The latter three are elaborated here.

Because someone else's knowledge is not easily knowable, it is risky to exploit this knowledge because the consequences of the combined intervention path are uncertain. This is because recognition and decomposition are now distributed: the provider does not know for sure whether the seeker's problem matches the class of problems in which the provider's representation is applicable. This has to be determined in a dialectical process in which provider and seeker decide together how recognition is to be distributed amongst them. They have two possibilities:

1. The provider describes the class of situations in which its competence is applicable, and seeker matches this with the current problem. In this case, the seeker does the recognition. For example: John tells Mary "I specialize in drivers under age 24".
2. The seeker describes the current problem and the provider matches this with the representation. In this case, the provider does the recognition. For example, Mary tells John "I have a driver here under age 24," to which John responds "I know all about that".

After having thus constructed a plan together, there are three possibilities for using the provider's competence in executing the seeker's plan: the first is that the provider executes his intervention triggered by the seeker. For example, John tells Mary to hand him over the insurance application. Second possibility is that the provider advises on a decision to be taken or an atomic action to be executed by the seeker. For example, John advises Mary to reject the customer. The third is that the provider transfers his or her knowledge to the seeker by verbalizing the associated causal model, so that he or she can apply it him- or herself. For example, John tells Mary all about when to reject drivers under age 24.

We can see that knowledge coordination comprises two phases: distributed recognition/decomposition and distributed execution. In both phases, the difficulty is in deciding how much knowledge to transfer from provider to seeker and the other way around. Transferring knowledge has to be done sparingly as empirical evidence shows it is much more expensive than transferring mere information [11].

The result of each process of knowledge coordination is that a certain amount of knowledge has been transferred among participants: the seeker has a better representation of the provider's competencies, while the provider has a better representation of the kinds of situations the seeker often finds himself in. This means less knowledge will have to be transferred on future occasions. This is the way shared knowledge is built up.

Even if a business process is a completely known task, it has probably evolved from a not yet completely known task via this build-up of shared knowledge. Since this build-up process is idiosyncratic, it has likely resulted in the sub-optimal procedures that were the driving force behind the original concept of BPR in [10]. An account of how shared knowledge is built up is therefore relevant to identifying process improvements.

Business process as coordinated competencies

Process knowledge consists of scripts, which are *retrieved plans* on the basis of causal models that can be flexibly applied in a process of goal-directed planning: the cognitive cycle. The old concept of a business process as a structured set of activities is a more or less crude abstraction of a set of coordinated competencies.

Similar to [16], we interpret BPR as the manipulation of scripts to improve the way knowledge is integrated to create value. To transform the theory proposed into an instrument for process improvement, and for the design of IT support for such processes, we have to elaborate a number of aspects. First, how to find improvements; then, how to assess the effects of different alternatives on performance indicators of processes, and lastly, how to implement a design by directly manipulating scripts in people and machines. Here, we focus on finding improvement alternatives.

4. KNOWLEDGE-BASED PROCESS IMPROVEMENT

We define knowledge-based process improvement as the *manipulation of scripts* in *cognitive agents* to serve *stakeholders* within certain *restrictions* and with respect to certain *performance criteria*. Those criteria are generally speed, amount of effort spent, quality, flexibility, and adaptiveness. Quality might refer to the number of errors made. Flexibility is the extent to which behavior can be adapted to the specifics of a case, and adaptiveness is the ease with which behavior can be adapted in response to structural changes in the environment.

There are three fundamental restrictions to process improvement. Processes cannot be improved boundlessly, because there is a fundamental limit to the available knowledge on how to make them better, the speed with which new knowledge can be imported or created (by research or experiment), and the limited speed of transfering knowledge between agents. Second are restrictions set by the environment: the scope of an improvement project. Third are the cognitive limitations of the agents available (man, machine): memory, speed, precision, and flexibility.

In the above, we have looked at knowledge-as-scripts in two ways: the distribution of those scripts among agents as shared and task knowledge, and the internal structure of scripts as the cognitive cycle. We can now define two fundamental ways to improve processes: by making new knowledge available or distributing existing knowledge better (meta-level), or by analyzing and enhancing existing knowledge (object-level). New knowledge can be imported or imitated from competitors, can be created by experimenting, or by combining existing knowledge from multiple agents: collecting *best practices*. Below, we describe process improvement principles at these two levels.

In redistributing knowledge, two considerations play a role. On the one hand, it is advantageous to attune the characteristics of a competence to the cognitive limitations of the agent possessing the competence: if little flexibility is required, for example, a task can be handed off to a machine. On the other hand, fine-grained distribution of knowledge, or specialization, increases coordination costs. Furthermore, redistribution of knowledge requires effort, as it requires explicitation of hitherto tacit knowledge.

Information systems development is in this view seen as an extreme form of knowledge explicitation and is therefore very difficult. The most extreme form of knowledge redistribution is *automation*: the complete handing over of a task to a machine. In case of task knowledge, knowledge-based systems are an example, while automating shared knowledge can be done by employing workflow management systems.

This last case is an example of changing the coordination search space: changing the shared knowledge available to people for performing knowledge coordination. A workflow management system structures and therefore restricts the number of possible collaboration opportunities between agents, but also speeds up the search for combined intervention paths. “Yellow pages” or “expert directories” do the exact opposite: they stimulate people to evaluate a larger number of collaboration opportunities, possibly trading off speed for flexibility and quality.

The second fundamental form of knowledge-based process improvement is enhancing the structure of existing knowledge. The cognitive cycle lends us some insights with which to analyze scripts and alter some balances in them. We give three examples of dimensions that can be manipulated.

We viewed the execution of a business process as the classification of a new problem situation into known classes of situations and selecting an appropriate response by recognition and decomposition. We can now ask ourselves if our number of responses is appropriate with respect to the variety of incoming situations. Having more possible responses costs more effort in recognition and decomposition, but might save effort or increase value down the line. Increasing or decreasing variety in the stimulus-response function that a business process implements is the first dimension to manipulate.

We have said that an agent will in general not execute the cognitive cycle in sequential order: assumptions will be made and actions will be taken before all information is available. Increasing or decreasing interleaving is our second dimension. Increasing interleaving will improve speed at the expense of quality, as atypical situations will be misjudged and more often lead to failure and backtracking. The previous balance is related to this one in that it is often advantageous to have special-purpose processes for simple and for complex cases, and that more caution, or, less interleaving, is often appropriate for complex cases.

Lastly, the proactivity of learning can be adjusted to the adaptiveness requirements of the process. We have defined learning as the adaptation of causal models based on success or failure of problem solving in the *register/evaluate* phase. We can choose to perform this stage after handling a problem, or just before handling the next. In a highly knowledge-intensive task such as project management, learning is often done up front by looking for similar projects in the past and finding success factors, as well as afterwards by having an evaluation meeting. Proactive learning should not be chosen when good information about success or failure becomes available only after long periods of time, such as in insurance or in some branches of medicine.

This short discussion reveals that the model offers many opportunities for searching for possible improvements, while simultaneously balancing efficiency with more knowledge-intensiveness related measures such as flexibility and adaptiveness. A systematic description of the search space of improvements, and the theoretical consequences of an improvement scenario on performance criteria, is a topic of future research.

5. RELATED WORK

The model presented here has not yet been elaborated into a modeling language. We hope to use the concepts of goal-directed planning, task/shared knowledge, coordination and the cognitive cycle in a modeling framework for work transformation. Below, we compare it with existing formalisms for business process and knowledge modeling, to inform such a formalization effort in the future.

Crowston in [4], describes an approach to process reengineering based on choosing appropriate coordination mechanisms to manage dependencies between tasks. This work seems more suited to situations where tasks are completely known beforehand.

DEMO (Dynamic Essential Modeling of Organizations, see for example [22]) is a method based on the Language/Action Perspective (LAP) of Winograd and Flores. Our cognitive model is closest to the essential model of DEMO, and focuses exactly on the behavior of the organization itself, instead of the communication necessary to enable this behavior. In this sense, DEMO is complementary to our work.

CommonKADS [24] is a method for the development of knowledge-based systems and knowledge management by identifying and describing knowledge assets in organizations. CommonKADS is especially suited to describing complex single-agent tasks, and says little the role of knowledge in coordination.

For business process modeling many activity-based formalisms are available, such as Petri nets and IDEF0. Such formalisms are mostly meant for describing non-knowledge-intensive processes or aspects thereof. Some authors seem to aim at extending activity models with knowledge concepts [1], while we take the opposite approach. Also related to this work is the field of *goal-based process modeling*. An early paper in this field is [13]; more recent examples are [14] and [2].

In the knowledge management field a lot has been written about the relationship between knowledge and processes (see for example [1], [25]). In that literature knowledge is usually described as a resource that is used in processes, instead of as a set of competencies from which processes are dynamically woven. Another difference is that knowledge is usually categorized in different ways (for example in *tacit* and *explicit*) but is otherwise left a black box. In our cognitive cycle, the structure of knowledge itself is analyzed. A third difference is that knowledge maintenance is often assigned to a separate process (such as Allweyer's knowledge processes, see [1]), while in our framework it is an integral part of executing a process.

Vidgen, in [26], proposes a synthesis of the Viable Systems Model with BPR. The VSM, developed by Stafford Beer, is an application of cybernetics to organizational design. While this work is not concerned with knowledge per se, the central notion of *variety* in cybernetics seems to bear strong resemblance to complexity and especially knowledge-intensiveness. The allocation of variety might coincide with the allocation of knowledge in organizations. Correspondingly, the notion of *adaptiveness* in cybernetics seem to coincide with the notion of *innovation* in the knowledge management literature.

6. CONCLUSIONS AND FUTURE RESEARCH

We set out to help solve the knowledge problem of BPR by taking a cognitive perspective on business processes, in order to be able to design the proper coordination, maintenance and use of operational knowledge in service organizations. This has resulted in a framework that interprets organizations as goal-directed planners and analyses processes in terms of the five phases recognition, decomposition, planning, action, and evaluation. Knowledge is seen as a causal model that stores experiences regarding regular patterns of behavior of the outside world and the effects of actions therein.

We have sketched the outline of a framework for identifying opportunities for improving knowledge-intensive processes. However, we have not yet systematically analyzed the search space of possible improvements, and the possibly unintended consequences of choosing one particular alternative for different performance criteria.

The crucial difference between our model and existing approaches is that it views a knowledge-intensive and complex process as a result of coordinated competencies rather than the process itself doing the coordination. Related approaches to modeling knowledge and processes described above seem strongly complementary, but do not share this view on the relationship between knowledge, coordination and business processes. Next steps will be refinement of the model through application on more elaborate examples, and validation in case studies.

7. ACKNOWLEDGEMENTS

This research is financially made possible by Achmea, part of Eureko.

8. REFERENCES

[1] Allweyer, T. (1997), "A Framework for Re-designing and Managing Knowledge Processes", IDS Prof. Scheer GmBH, whitepaper.

[2] Bider, I. (2002), "*Goal-Oriented Patterns for Business Processes*", Workshop on Goal-Oriented Process Modeling, London, 2002.

[3] Buzacott, J.A. (1996), "*Commonalities in Reengineered Business Processes: Models and Issues*", Management Science, Vol. 42, No. 5, May.

[4] Crowston, K. (1997), "A Coordination Theory Approach to Organizational Process Design", *Organization Science*, Vol. 8, No. 2, pages 157—175.

[5] Decker, S., Erdmann, M., Studer, R. (1996), "A Unifying View on Business Process Modelling and Knowledge Engineering", *Proceedings of the 10th Knowledge Acquisition for Knowledge-Based Systems Workshop*, Banff, Canada.

[6] Dietz, J.L.G., Mulder, H.B.F. (1996), "Integrating the strategic and technical approach to business process engineering", *Proceedings of the International Symposium on Business Process Modelling*, Springer-Verlag, 1996.

[7] Eppler, M.J., Seifried, P.M. and Röpnack, A. (1999), "Improving Knowledge Intensive Processes through an Enterprise Knowledge Medium", in: Prasad, J. (ed.): *Proceedings of the 1999 ACM Conference on Managing Organizational Knowledge for Strategic Advantage*, New Orleans (USA), April 8th-10th, 1999.

[8] Gioia, D.A. and Poole, P.P. (1984), "Scripts in organizational behaviour", Academy of Management Review, Vol. 9, No. 3, pp. 449—459.

[9] Grant, M. (1996), "Toward a knowledge-based theory of the firm", *Strategic Management Journal*, Vol. 17 (Winter Special Issue), 109–122.

[10] Hammer, M. and Champy, J. (1993), "Reengineering the Corporation: A Manifesto for Business Revolution". Nicholas Breadley Publishing, London.

[11] Kock, N. and McQueen, R.J. (1998), "Knowledge and Information Communication in Organizations: An Analysis of Core, Support and Improvement Processes",

Knowledge and Process Management, Vol. 5, No.1, 29—40.

[12] Kock, N. and Murphy, F. (2001), "Redesigning acquisition processes: a new methodology based on the flow of knowledge and information", Defense Acquisition University Press, Fort Belvoir, VA.

[13] Lee, J. (1993), "*Goal-Based Process Analysis: A Method for Systematic Process Redesign*", In: Proceedings of the Conference on Organizational Computing Systems, Milpitas, CA.

[14] Liu, L., and Yu, E. (2002), "*Designing Web-Based Systems in Social Context: A Goal and Scenario Based Approach*", Proceedings of the 14th International Conference on Advanced Information Systems Engineering (CAiSE'02), Pages 37—51, Toronto.

[15] Malone, T.W., and Crowston, K (1994), "The Interdisciplinary Study of Coordination", *ACM Computing Surveys*, Vol. 26, No. 1, pages 87—119.

[16] Manheim, M.L., and Fritz, M.B. (1996), "*Information Technology Tools to Support Virtual Organization Management: A Cognitive Informatics Approach*", in: Organizational Virtualness – Proceedings of the VoNet-Workshop, April 27-28, Bern.

[17] Newell, A. and Simon, H. (1972), "Human Problem Solving". Englewood Cliffs, NJ: Prentice-Hall.

[18] Nissen, M.E. (1998), "*Redesigning reengineering through measurement-driven inference*", MIS Quarterly, Vol. 22, No. 4; pp. 509—534.

[19] Nooteboom, B. (1996), "Towards a Cognitive Theory of the Firm: Issues and a Logic of Change", EIASM conference on organisational cognition, Stockholm.

[20] Peppard, J., and Rowland, P. (1995), "The Essence of Business Process Re-engineering", Prentice Hall, New York.

[21] Reijers, H.A. (2002), "Design and Control of Workflow Processes: Business Process Management for the Service Industry", Ph.D. dissertation, Technische Universiteit Eindhoven, 2002.

[22] van Reijswoud, V.E., Mulder, H.B.F. and Dietz, J.L.G. (1997), "Speech Act Based Business Process and Information Modelling with DEMO", In: *The Proceedings of the CAiSE'97/IFIP8.1 International Workshop on Evaluation of of Modelling Methods in Systems Analysis and Design*, Barcelona.

[23] Schank, R. and Abelson, R. (1977), "Scripts, Plans, Goals and Understanding", Hillsdale: Lawrence Erlbaum Associates.

[24] Schreiber, G., Akkermans, H., Anjewierden, A., de Hoog, R., Shadbolt, N., Van de Velde, W. and Wielinga, B. (2000), "Knowledge Engineering and Management – the CommonKADS Methodology", MIT Press, 2000.

[25] Staab, S., Schnurr, H.P., Studer, R. and Sure, Y. (2001), "Knowledge Processes and Ontologies", *IEEE Information Systems*, February / January 2001 (Special issue on knowledge management).

[26] Vidgen, R. (1998), "Cybernetics and Business Processes: Using the Viable System Model to Develop an Enterprise Process Architecture", Journal of Knowledge and Process Management, Vol. 5, No. 2, pp. 118—131.

[27] Winograd, T. and Flores, F. (1986), "Understanding Computers and Cognition: a New Foundation for Design", Ablex, Norwood NJ.