BDI-Agents for Agile Goal-Oriented Business Processes

Birgit Burmeister[1],M.Arnold[2]
DAIMLER AG
[1]GR/EPF / [2]ITP/DP
D-71059 Sindelfingen , Germany
+49 7031 4389 390
[1]birgit.burmeister@daimler.com
[2]michael.arnold@daimler.com

Felicia Copaciu
Daimler Protics GmbH
628-P305,
D-70546 Stuttgart, Germany
+49 7031 90 45653
felicia.copaciu@daimler.com

Giovanni Rimassa
Whitestein Technologies
Pestalozzistrasse, 24
CH-8032 Zürich, Switzerland
+41 44 256 5019
gri@whitestein.com

ABSTRACT
Business processes are the core assets of enterprises. They turn the business potential into actual competitiveness on the market. To face the challenges posed by today's changing and uncertain business environment, traditional business process management (BPM) approaches are not sufficient anymore. This paper presents an approach to business process management, which leverages Agent Technology, especially BDI-Agent features to obtain agile business process behavior. This paper sketches the problem, describes the solution approach, and presents the experiences gained in a concrete case study in the domain of Engineering Change Management.

Categories and Subject Descriptors
I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence – *Intelligent agents Multiagent system.* H.4.1 [**Information Systems Application**]: Office Automation – *workflow management*

General Terms
Design, Experimentation

Keywords
Autonomous Agents, BDI Agents, Business Process Management, Goal-Oriented Business Process Modeling, Agile Business Processes.

1. INTRODUCTION
The definition of the core business processes is fundamental for any enterprise across all kinds of industries. The effective setup, execution and evolution of business processes have an essential impact on successful business operations. By definition a business processes consist of a set of activities, connected in a structured whole. Business processes describe the modes of operation of an enterprise in given situations.

Cite as: BDI-Agents for Agile Goal-Oriented Business Processes, B.Burmeister, M.Arnold, F.Copaciu, G.Rimassa, Proc. of 7th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2008) – Industry and Applications Track, Berger, Burg, Nishiyama (eds.), May, 12-16., 2008, Estoril, Portugal, pp. 37-44.

Business Process Management (*BPM* for short) subsumes all activities that an organization performs in order to create, maintain, control and evolve its business processes. BPM involves people, organizations and technologies [1], [2]. In addition, BPM can be carried out with varying levels of automation. Nevertheless we think that running business processes should be supported by modern information technology.

The past years in business where characterized by trends towards more flexible ways of working, shorter organizational reaction times and fully embracing market, business unpredictability, along with the increase in distribution and the need to preserve understandability despite more and more complexity. These trends show no signs of abating.

Compared to these challenges, the current status of BPM in most companies is inadequate: Business process models are modeled by graphical modeling tools. Unfortunately these "should-be" processes are usually only used to cover white walls in the offices. This is due to the fact that today's modeling tools support a very simple mind model behind modeling: processes are seen as long and fixed sequences of activities, which is far away from reality and from the challenges. The processes really executed are different from the ones on the wallpapers; "shadow" processes dominate the "official" ones. IT systems are built with the outdated "should-be" processes in mind; the process is mostly hard-coded with no explicit representation of the process. Thus the IT-Systems are outdated as they are rolled out. They are not understood or inflexible and hence misused by many users. Changes of the process and the supporting IT-systems are costly, imply the high risk of code modification, and always lag behind reality.

The main challenge is thus business process agility. Agility means not only to have process flexibility, i.e. the ability of the process to be adapted. An agile process should be able to pro-actively adapt itself quickly to a changing environment. This should be achieved both at modeling level, i.e. a changed process model should be transferred seamlessly into the supporting IT-system; and at execution level, when the executing process adapts itself to the current environment.

This paper presents *agile, goal-oriented business process management* as an effective approach to these challenges. The approach is based on agent technology, which offers methodologies and tools to meet the requirements mentioned above. The approach was successfully applied to a business process for engineering change management in the automotive industry.

This paper is organized as follows. Section 2 describes the major challenges of achieving agility in BPM, with particular reference to the domain of engineering change management. Section 3 presents the idea how Agent Technology is used in conceiving and realizing a solution for agile BPM. Section 4 presents details of the project agile change management (ACM; for short) and describes the experiences we have gained in applying the approach in a real-world business process. Lastly, Section 5 concludes the paper and gives an outlook on future work.

2. REQUIREMENTS FOR AGILE BPM

Compared to typical business processes, e.g. in call centers or financial services, managing engineering processes is even more challenging for several reasons [3]: First, they are long running processes. Designing a car takes many years and the next model of a large airliner is the result of nearly a decade of engineering and production planning. During this time period things change – what has been an up-to-date approach in the beginning may be outdated at the end. Second, engineering processes have to cope with uncertainty because of their mixture of creative tasks, collaborative work and repeating activities. This results in very complex processes with many alternative paths and sections that cannot be planned in advance. Third, some products became so complex that not all engineering tasks can be performed within one enterprise. This results in engineering processes which are partly executed by external partners. Managing such processes means to handle external engineering tasks without knowledge about "how" they internally work to provide their service.

Engineering change management (ECM) is one of the most crucial processes in engineering [4]. The process ensures that any change made to the product is documented, evaluated regarding costs, technical feasibility, compliance to laws and regulations, etc. Based on this detailed evaluation a decision is made, whether the change will be actually performed and implemented. Due to the big number of involved departments change management can be a long running, time-consuming process.

However for a certain class of changes (e.g. software bug fixes or minor changes) the "normal" process is too long and too complicated. Therefore a somewhat different process is needed to support this sort of change requests. Today this problem is handled by so called "light processes". But the demand for light processes tends to increase more and more. In addition any process variants have to be implemented by the supplier of the software and it takes some time to have a new process variant implemented. In the current Daimler ECM system more than a quarter of the changes run such a light process. Moreover often even a light process does not really fit the business needs. Today it is not possible to adopt a fitting process in time. Therefore this way of handling the engineering change process is not feasible any more.

Facing these challenges we think a new approach for modeling and executing business processes is needed. The approach has to

- support designing huge, complex processes
- decrease the effort for changing and maintaining the process model and IT-system by offering a seamless transfer of a process model into an IT-system and
- allow flexibility and agility in process execution through IT-systems.

We have found no commercial product that achieves these requirements in a satisfactory way. E.g. the ARIS tool suite [5] offers comprehensive modeling functionalities, but does not (yet) support the seamless building of an IT-System out of the process models. Approaches towards more flexibility of workflows, as described in [6], add flexibility on a rather low-level. They allow, e.g. to add or skip certain single steps in a workflow. But the overall process stays in principle the same.

We think that agent technology can offer better approaches and methods to meet the requirements stated above. Agent-oriented software technology was first introduced to deal with large-scale, distributed software systems, which are embedded in dynamic environments, and allow for the interaction of different partners.

3. USING BDI-AGENTS FOR BPM

3.1 From Agents to Agile Business Processes

Having the main challenge in mind to make business process agile, agent technology was our first choice to look for support. Multi-Agent Systems have presented the idea of agility in different application areas: flexible production control allows for agile behavior of the whole system facing changes and disturbances [7], system in logistics present an agile transportation planning [8], In the RoboCup [9] robots play football and act very flexibly and agile in their environment. Having this in mind, how can agent technology be used to support agile business processes?

In the BDI (belief-desire-intention) agent architecture an agent is described by its *beliefs*, i.e. the information an agent has about itself, its environment and possibly other agents; its *desires*, i.e. motivations of the agents that drive its course of action; and finally its *intentions*; i.e. the short-term goals that the agent wants to achieve, derived from its desires and external events, to which the agents wants to react. To achieve its goals/intentions an agent has certain plans how the goals can be achieved. Different plans are designed for different situations, which is described by the plan's context condition. A plan consists of certain actions/steps that have to be executed to achieve the corresponding goal.

The BDI architecture was first implemented by [10]. In an implementation the execution is as follows: The agent has to decide which goals it wants to follow next and which plan can be used to achieve the goal. To accomplish this, the agent introspects its goal base and extracts the goals which are not yet fulfilled; next it collects all plans from its plan base which could be used to fulfill these goals; finally it checks the current context (i.e. the current belief base) whether it fits to the context the plan was designed for. Performing the plan means to execute the single steps of the plan. These single steps can be: interacting with the environment, e.g. with the user of the system, performing some kind of computation, manipulating the own data base (belief base), or sending and receiving messages from other agents.

In section 4 we will describe how we have used this goal-oriented and context-aware execution of agent plans to allow for agile, goal-oriented business processes. But before that we will have a look at the relevance goals have in business process management and modeling today.

3.2 Goals in BPM

In day-to-day business operations, it is natural to set goals, decompose a goal into sub-goals, define or reuse plans, and routinely track and check the execution of chosen plans in order to detect problems as they occur (or even better before they do), and to take appropriate actions.

In business organizations there is an upper management level, which coarsely drives the more detailed project planning and tracking. Such a level gives clear direction without unnecessarily limiting the decisional power and the adaptation leeway of the finer-grained management operations.

It is thus natural for upper managers to be more concerned with (and express their views in terms of) what is to be achieved than how to achieve it. Operating at the goal level is a natural approach for such people with the core of the business process captured through goals and sub-goals independently of the actual activities.

When moving to detailed planning in business or project management, there is usually more to the plan than just its tasks and structure. At the very least, the expected objectives of the plan need be stated, and also, in many cases, the initial requirements. Moreover, additional information such as resource and time consumption is also often attached to a plan.

Although the goal-orientation seems quite natural to the area of business process management the modeling of the actual business processes (i.e. plans) is nowadays not linked to the goals. In the widely used business process modeling suite ARIS [5] it is possible to describe business goals, but there is no direct link to the model of the actual processes.

There are some approaches to address this neglect. In [11] the relation of business goals to business activities is described with the aim to build up an object-oriented business process model. A Goal/Means-Hierarchy is defined, where Means are linked to object methods, representing business activities. Input and output of the activities is described to get the execution sequence of the activities. In [12] the authors define a formal framework to relate goals derived from requirements specification to the business processes modeled in BPMN (business process modeling notation), an emerging OMG (object management group) standard for business process modeling [13]. The linking of goals and processes is twofold: A first link between a goal and an activity in the process model, states that the goal has some effect on the activity (this may be an achievement effect, but also an obstructing effect.). A second link states whether the activity satisfies the goal or how it satisfies the goal. Nevertheless, the relation of goals and processes and especially how they are working together in process execution remains unclear in both cases.

We think that this missing link from business goals to executable processes and process steps can be filled by agent technology.

Although the coupling of agent technology and business process management is not new, existing approaches (e.g. [14], [15]) focus on agents' communication and cooperation (and mobility) abilities to support the process execution. Single tasks are modeled as services and agents offer and use these services in executing the process. Moreover different types of agents are used for the implementation of a workflow system [16]. Goals do not play any role in the sense of business goals in these approaches.

Closest to our idea is the work of Georgeff, realized in the Agentis platform [17]. Agentis uses the BDI architecture for business process modeling and management. Nevertheless the main focus is to use agent plans as business process, no explicit modeling of goals is done.

4. GOAL- & CONTEXT-ORIENTED BPM

4.1 Plans and Goals to Express Processes

Inspired by agent technology and the concept of goal orientation and decomposition the main ideas behind our modeling approach are (i) to have a modular process model that describes the single steps of a process (sub-processes, activities) separate from the goals of the process and the different contexts in which the process can be executed; and (ii) to have different modeling levels, for the different parts of the process model. This modular, goal- and context-based process model can then be executed as an agile process, by considering current goal and context when determining the next step in the process, just as realized in the BDI agent architecture. The agent can be seen as an assistant or guide of the user who is responsible for "driving" a task through the process. Since the agent can perform a lot of routine work for its "boss", it can take over the role of a "process driver" on behalf of the user.

An additional effect of using this goal-oriented approach is the separation of the statement of *what* the desired system behavior is, from *how* the behavior is performed.

The process modeling consists of several steps: First the goals that have to be fulfilled by the process are identified and gradually refined by sub-goals. Goals can be operational goals, defining the actual outcomes of the process or more general goals, concerning time, cost and quality that have to be obeyed by the process. These two kinds of business goals result in two types of goals: *achieve goals* are used to model the operational goals of the process; *maintain goals* are used for those goals, the agent will monitor during process execution.

Next the different possible contexts of the process are described by means of *context variables*. These variables describe certain aspects of the process environment that will influence the process execution.

If no more sub-goals can be defined for a goal at the next level the different ways to achieve or maintain a sub-goal, i.e. the plans have to be defined. For each plan a condition has to be defined stating the condition in the process context, when this plan will be used to fulfill its corresponding goal. Maintain goal also have a *maintain condition* that has to be true all the time. The corresponding plans for these goals are called as soon as the condition becomes false. Any conditions refer to the context variables and their possible values.

As a result the final model of the business process will consist of one or more goal hierarchies, a list of context variables (with their possible values) and a set of plans with conditions linked to sub-goals. In the next section a process modelling for an engineering change management process will be explained in some more detail.

4.2 Modeling the ECM process

We have used this modelling approach for the engineering change management process of Daimler AG. We started this project, now called "agile change management" (ACM) in 2005 with a feasibility study. Daimler Group Research has implemented a software demonstrator consisting of a graphical modelling tool for describing the goal hierarchy, defining context variables and modelling plans. The process execution is done with an agent tool, namely JadeX [18]. As described in [19] during this feasibility study we have first led several interviews with experts of the ECM process and system in use today. These interviews were driven by the goal-oriented approach, i.e. first we asked the people about the goals of the process ("What has to be done?") and the influence factors that drive the process, and only later came to the process details ("How is it done?"). This process analysis then resulted in a goal- and context oriented model of the ECM process, which was modelled with the graphical modelling tool. The model was finally executed for different flexibility and agility scenarios in the mentioned software demonstrator. Also part of the feasibility work was the evaluation of different software architectures for a real-world application with the pros and cons. Although an agent tool is very well suited for the approach, also other alternatives exist to implement the goal-oriented approach to agile business processes, for details please refer to [19].

After the successful feasibility study a suitable commercial software tool had to be found to implement a system for ACM with the goal- and context-oriented approach. The software tools used for the research demonstrator are not suited to build up a real-life application with the requirements concerning run-time performance, scalability, integration capabilities into existing infrastructures, maintenance etc.

Therefore in a dedicated selection process several IT suppliers were evaluated. The Whitestein LS/TS platform [20] for multi-agent systems was chosen as a candidate base infrastructure. The domain expertise and innovation-fueled vision of Daimler met with Whitestein agent technology leading offer to jointly tackle the challenging tasks ahead. Therefore in the ACM project Whitestein and Daimler are collaborating aiming at the deployment of a novel, agile BPM system in the domain of ECM.

The agile change management project will rely not only on the generic LS/TS platform, but also on Whitestein's LS/ABPM suite [21] that builds on agent technology to support the modeling, execution and maintenance of agile, goal-oriented business process models. These models in LS/ABPM are expressed using the GO-BPMN modeling language. This is a goal-oriented extension of the OMG standard BPMN; plans are still represented as BPMN workflows, but they are also provided with feasibility preconditions and attached to a goal that they fulfill, which is in turn inserted into a goal hierarchy (see below).

The goal-oriented process model for ACM that was modeled with a prototypical version of LS/ABPM looks as follows (the figures are taken from the modeler tool itself):

The main purpose of the process is to have a change (request) managed and to have the change in the car. This overall goal can be subdivided into 5 sub-goals (see figure 1):

The change request (CR) has to be initiated, it has to be described in detail, the CR has to be evaluated, the decision has to be made, whether the change will be implemented, and finally the change has to be implemented..

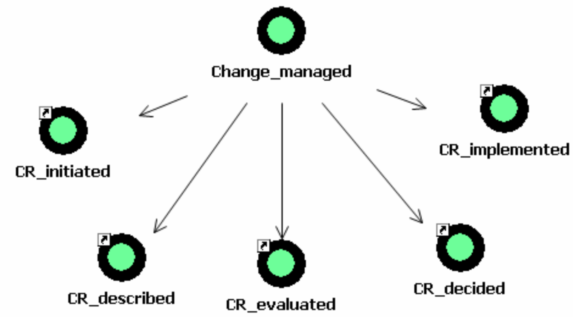

figure 1:goals for ECM process

Each of these sub-goals was further refined into more sub-goals. The number of levels depends on the complexity of the corresponding sub-process. The goal "change_initiated" has one more level of sub-goals; these have one or more plans related. The goals "CR_described" and "CR_evaluated" are more complex because the whole process of collecting and evaluating the necessary information for the decision is modeled here.

The context of the ECM process was described in context variables, ranging for variable for the general context of change management (affected car line, category of change, urgency etc.) to variables holding results of single process steps (change description, list of affected parts, evaluation results, etc.)

These context variables were then used to define the context conditions for plans, as can be seen in figure 2.

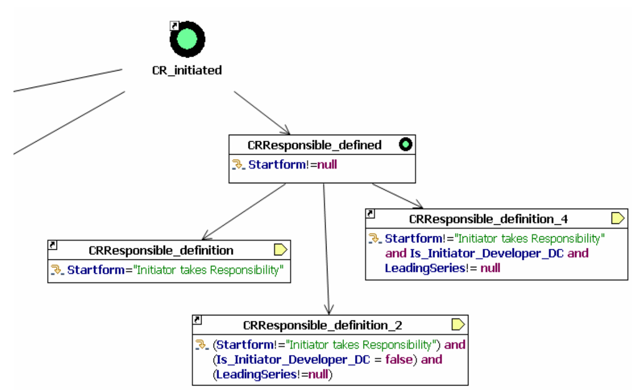

figure 2: plans with context conditions

One sub-goal of "CR_initiated" is that the responsible person for the CR is defined. There are three different context situations, depending on whether the initiating person has already taken this role or not (this fact is hold in the context variable named "Startform". Additionally, the plan to determine the responsible person depends on the organizational the initiating person belongs to and whether a leading car series was already defined. Based on the actual values of the affected context variables the corresponding plan will be chosen during process execution.

The single plans can be modeled as normal sub-processes. Therefore we have used a subset of BPMN. A plan can be more or less complex; the single steps of a plan will be later executed as modeled.

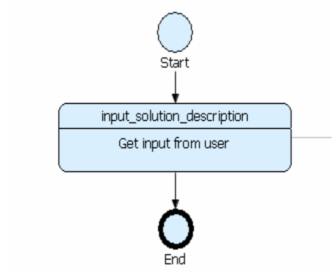

figure 3: simple plan to get user input

The simple plan shown in figure 3 just consists of one task, namely to collect some input (a description of the solution implemented by the change) from the user.

During modeling a plan is composed of a sequence of tasks (branches are also possible, see below). The tasks are chosen from a pre-defined set of task from a task library. The library contains the basic tasks that are needed for the process. For the ECM process these tasks are, e.g. getting input from the user, writing context variables, showing some text to the user, etc.

The plan in figure 4 is more complicated. Besides different tasks it also contains two branches were the control is managed within one plan. The conditions for the branches also contain references to context variables.

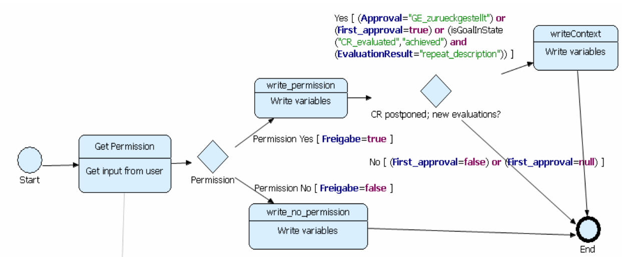

figure 4: complex plan with branches

In addition to the operational goals described above maintain goals were collected into a second goal hierarchy: the time-cost-quality goals. These goals monitor certain conditions of the process that need to be true all the time (figure 5). E.g. one of the maintain goals monitors the *necessary finishing time* of the process, i.e. the time the process has to be finished to meet a certain deadline.

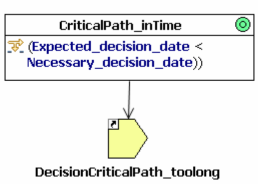

figure 5: maintain goal with condition and plan

Changes in the context lead to regular re-computations of the expected finishing time. If the expected time is greater than the necessary time (i.e. the maintain condition becomes false) a plan is chosen to take some action to let the condition be true again.

These maintain goals and their plans were used to add part of the desired agility to the process. The agent pro-actively monitors the conditions and takes action if a condition becomes violated.

The resulting overall process model consists of 60 achieve goals in up to 7 levels, 10 maintain goals, 85 plans and about 100 context variables.

The model with goals, context variables and plans can be directly compiled into a BDI agent description with goals, beliefs and plans. The model was tested with 10 different more or less agile process scenarios, which reflected different regular process variants and agility scenarios, which could not be handled with today's system.

The following components of LS/ABPM were used in modeling, testing and executing the ACM process:

- Process Modeling and Testing Environment. Graphical tools and execution environment to create, modify, and test business process models.
- Process Execution Engine. A server-based execution environment, allowing executing a large number of business process instances and process participants.
- Process Management and Administration Console. Tool to oversee running business process instances, providing lifecycle management, organizational and user model maintenance, and alarm handling.
- Application Frameworks and Libraries. A diverse set of software components, extension points and libraries that allow leveraging LS/ABPM features to create complex business applications.

The Eclipse-based, graphical modeler tool is used to edit GO-BPMN models at the plan and goal level, before they can be uploaded to the execution engine server. This paper will however not deal with the execution or operation aspects of LS/ABPM.

4.3 Lessons learned and Challenges

This section summarizes the experiences gathered with the goal- and context-oriented modeling approach in general and with the LS/ABPM tool suite during the ACM-prototype phase. During this phase the process model generated in the software demonstrator developed by Daimler Group Research was used as the ACM-reference process model. This model was transferred to the LS/ABPM tool suite. The model was also enhanced to cover more scenarios, chosen to demonstrate normal process flows as well as flexible and agile process runs. The prototype phase lasted about three months. During this time the resulting ACM-process model was developed and tested in several iteration cycles. The LS/ABPM agent-based platform (version 0.2 base) was used to model, execute and test the ACM-process.

In general the goal- and context-oriented process modeling was rather easy to learn. Up to four people, none of them was involved in process modeling during the demonstrator phase, built up parts of the process model. The goal hierarchy was rather quickly built up. Although it is quite large the modeling effort was not that high (see figure 6). Much more effort was necessary (i) to model the plans, which encapsulate smaller or bigger pieces of the process workflow and (ii) to manage the consistency of the multitude of conditions.

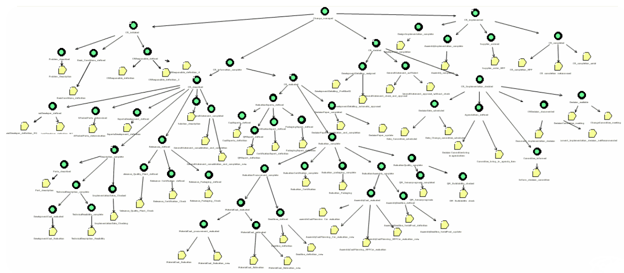

figure 6: goal hierarchy of ACM prototype

An advantage of this modeling approach is that it implicitly offers support for parallel execution of the process parts that do not depend on each other. This can reduce the overall time needed for process execution. Moreover maintain goals are a good means to provide the process with additional agility: the agent monitors conditions that have to be fulfilled throughout the process (e.g. time constraints) and pro-actively initiates activities to avoid problems before they appear.

During development of the prototype the support for rapid prototyping and execution of process models provided by LS/ABPM has proven to be very helpful. The developed models represent "living process models", which can be directly executed and visualized. The part of the web user interface that is coupled with the workflow is generated directly from the process model. The interface is computed directly from the parameters of the corresponding task: context variables, their types and possible values. With this approach changes in the process can be quickly modeled and tested. Thus errors in the models can be discovered and corrected in a short time.

As stated above the starting point for building the ACM-prototype model was the ACM-reference process model developed for the software demonstrator. The underlying agent engine of this demonstrator (JadeX) has a partially different modeling and execution semantics compared to the LS/ABPM tool. There were differences in modeling XOR-gateways, context conditions and maintain goals. These semantic differences lead to deviant behavior in the process model execution and increased the testing effort.

During ACM prototype phase the goal- and context-oriented modeling approach was applied in a bigger setting for the first time. Therefore it was necessary to extend the goal-& context-oriented modeling language. Two sorts of conditions for goals were added to be able to express (i) optional goals, which are not needed in certain situations (modeled by *skip-conditions*), and (ii) sequence-dependencies among goals (*pre-conditions*). E.g. some goals can only be pursued after certain other goals are fulfilled. Although the ACM-prototype represents only a simplified version of the real process, some aspects could not be modeled by employing tasks provided in the standard task library. For example, an interface to an external system required an application specific task, which involved also some implementation effort.

Using an agent tool suite and applying the goal- and context-oriented process modeling approach for the prototype, we also got some general ideas what the main future challenges will be.

The work to model and test the process model was distributed among four persons. The concurrent developing of the process model was one of the challenges of the ACM-prototype. The LS/ABPM modeler provided import functionality to merge different parts of a model. But there was no support in versioning models and managing changes. Therefore it was a major task to keep the merged model consistent.

The model testing effort increased noticeable with the size of the model and the number of defined test scenarios. In ACM-prototype ten scenarios were specified and implemented. All of them were covered by one process model. This leads to more complex context conditions and to an increasing complexity of dependencies between different context conditions. One of the big challenges during the test phase was to keep the model consistent and to define the right context conditions that result in the correct execution for all scenarios. Therefore more support for dependency analysis, automated simulation and testing of the process models is needed.

Based on the experience gathered with the ACM-prototype, it can be concluded that the model complexity is influenced by the size of the model but mainly by the dependencies created through the context variables, context conditions and tasks that can manipulate the goal states (deactivate, re-activate). The size of the model can be expressed by metrics like the number of goals and plans, the number of levels of the goal hierarchies and the number of the global context variables. A good practice to reduce this complexity is to define local variables in plans whenever this is possible. Regarding the dependencies generated by context variables and context conditions, it should be noticed that when a variable is used in a lot of context condition, changing its value in one or more tasks can influence the execution of a big part of the process model. For example, setting a certain Boolean context variable to "true" could lead to the activation of a lot of goals, starting the execution of some plans and to skipping some other goals. The deactivation and re-activation of goals from plans create dependencies that are not visible at the goal-hierarchy level, but "hidden" in tasks' parameters. In order to support a better model complexity management, new concepts for modularization of process models, advanced search functionality for goal names and context variables, as also model analysis tools are developed.

The advantage of generating the web user interface directly from the process model will be enriched to allow more comfortable user interfaces. The prototype version is very simplified: each user input task generates a separate To-Do for a specific role. Data dependencies among different input fields cannot be modeled. In real processes the support of To-Dos that contain several input tasks and also the support to handle data dependencies within To-Dos is needed.

The focus of the ACM-prototype was more on the process control and less on aspects like business data management, user management or an advanced graphical user interface. Therefore these and further aspects, like a sophisticated type system and an organizational model, are needed for a full-fledged goal- and context-oriented business process management suite.

Regarding business data management, the following issues should be considered in the future development:

- additional to process modeling it is necessary to have user-defined data types, allowing the definition of complex, structured and business-specific data types;

- appropriate means for representing the complex data types in the application user interface have to be provided without the need for extensive programming;
- possibilities to distinguish different categories of data, like external administration data, application-specific data and process control variables.

A big challenge is to solve all these issues without significantly increasing the complexity of the modeling tool. The objective of the approach should be kept in mind, namely that the business process modelers with rather little knowledge on implementation level should be able to model and execute realistic, complex business processes from their domain.

The experience gathered with the ACM-prototype was very valuable for demonstrating that goal- and context-oriented process modeling together process execution driven by an agent tool suite provides a suitable technology for modeling and executing agile business processes. It also helped to find out the required extensions to the modeling language to model complex real world business processes. Finally we detected the requirements to manage the resulting model complexity through modularization concepts, modeling support for analyzing dependencies and automated simulation and testing capabilities.

5. CONCLUSION

Business processes are important to the successful operation of an enterprise. While the field of BPM has introduced noteworthy progress in the computer support for handling business processes, more advanced approaches are necessary in order to meet the challenges of business agility.

Based on the requirements for agile processes we think that agent technology can be used to provide a different, i.e. goal-oriented process modelling, and flexible and agile process execution. This paper presented our approach, which uses the BDI agent architecture as a starting point for business process modelling. As a BDI agent is described by its beliefs, goals to be achieved, and plans to fulfil the goals we model a business process with goals, contexts, and plans. This goal- and context-oriented process model can be easily transferred into a BDI agent tool to be executed by an agent.

We have described our experiences in applying the approach to the engineering change management process of Daimler AG. After the feasibility study conducted by Daimler Group Research in a second project phase a prototype was built up with the LS/ABPM suite of Whitestein Technologies. The results gained during this work were quite promising. Nevertheless we also found some challenges that have to be met when applying the agile goal-oriented approach to real world BPM.

The technological leverage of agent technology, together with the combined concepts of goal- and context- orientation, allow the conception and realization of an advanced BPM product, and of an innovative application in the Engineering Change Management domain at Daimler AG, such as the ACM system is going to be. Having a conceptual framework and an infrastructure layer based on multi-agent systems ideas provides some benefit to an agile BPM system, such as:

- Direct execution of a goal-oriented model, through actual BDI agents. A seamless transition from business modelling to actual execution is thus possible.
- Strong runtime agility, that allows adopting and dropping goals at any time, to dynamically modify the state and even the structure of running process instances.
- Autonomic feedback loops, where controlling cycles can be set up and effected at multiple levels (agents can control process models, agents can control other agents, or process models can control other process models).
- High modularity, where the system is composed by sets of strongly encapsulated components (agents, services, process models, capabilities libraries)

The above traits have been already experienced to some extent in the work done so far, but much further progress is expected along the timeline of the overall project.

6. OUTLOOK

The perspective on future development is manifold. On the one hand, further research and conceptual work is planned or even already ongoing, in order to deal with more advanced topics than the core challenges tackled during the ACM prototype work.

In particular, a major topic is going to be the use of cooperative agents to manage a multi-process structure, where different business process instances can interact and coordinate themselves together. This topic is close to the core properties and strengths of the multi-agent system approach, as pointed out by various research works such as [15] and [16]. The scientific objective here will be to combine in an effective and novel way the classical topics of cooperating rational agents, as e.g. agent cooperation protocols [22], [23], with a modelling and tool metaphor that is suitable for the kind of environment and users typically found in the BPM domain.

Moreover, the further evolution of the ACM project will bring new information and new insight about the needs and issues specific to the engineering change management. As noted earlier in this paper, such an application domain presents features such as long running processes that combine creative and repetitive tasks, and can put heavy strain on a BPM solution if agility is not well woven into the system.

Last, but not least, the agent-based software technology is being currently developed and enhanced. The first official releases of LS/ABPM, which will be adopted in the ACM development, try both to learn from the ACM prototype experience and to broaden the scope to cover the space of a full fledged product.

This will result both in a deeper support for the core modelling language features, such as complex data types or multiply instantiated goals, and also in completely new areas such as organizational modelling or quantitative simulation environment.

These and other topics go well beyond the scope of this paper, and are planned to be the subject of future publications, in the form of empirical case study reports or original research features proposing solutions to the engineering problems that we are convinced the future work has in store for the ACM project.

7. ACKNOWLEDGMENTS

We would like to thank all the people participating in the ACM project, especially Jürgen Scharpf, Hans-Peter Steiert of Daimler AG, Stefan Bussmann, Christian Wiech of MentalProof Software

GmbH, Radovan Cervenka and Martin Kernland of Whitestein Technologies for the always excellent and inspiring teamwork.

Thanks to the anonymous reviewers for their valuable comments.

8. REFERENCES

[1] van der Aalst, W.M.P., ter Hofstede, A.H.M. and Weske, M. 2003. Business Process Management: A Survey. In Business Process Management, Proceedings of the First International Conference (Eindhoven, The Netherlands, 2003), LNCS 2678. Springer.

[2] Smith, H., Fingar, P. 2003. Business Process Management: The Third Wave, Meghan-Kiffer Press.

[3] Beuter, T. 2002. Workflow-Management für Produkt-entwicklungsprozesse. Dissertation UniversityUlm. (in german)

[4] Strategic Automotive product data Standards Industry Group (SASIG). 2008. White Paper Engineering Change Management (ECM) Reference Process,Version 1.0

[5] Scheer, A.-W. 1994. Architecture of Integrated Information Systems: Foundations of Enterprise Modelling, Springer.

[6] Reichert, M; Dadam, P. 1998. ADEPTflex - Supporting Dynamic Changes of Workflows Without Losing Control. In Journal of Intelligent Information Systems - Special Issue on Workflow Managament, 10 (2), Kluwer, 93-129

[7] Schild, K., Bussmann, S. 2007. Self-organization in manufacturing operations. In Communications of the ACM, 50 (12), 74-79

[8] Dorer, K. and Calisti, M.,2005.An Adaptive Solution to Dynamic Transport Optimization.In Proceedings of the AAMAS05 industry track (Utrecht, The Netherlands, 2006)

[9] The RoboCup Federation, www.robocup.org.

[10] Rao, A.S., Georgeff, M.P. 1995. BDI Agents: From Theory to Practice. In V. Lesser (ed.) Proceedings of the 1st Int. Conf. on Multi-Agent Systems (San Francisco, USA, 1995). MIT-Press.

[11] Kueng, P., Kawalek, P. 1997. Goal-Based business process models: creation and evaluation. Business Process Management Journal 3 (1), 17-39

[12] Koliadis, G., Ghose, A. 2006. Relating Business Process Models to Goal-Oriented Requirements Modesl in KAOS. In Hoffmann. A. et al. (Eds.). Advances in Knowledge Acquisition and Management (Proceedings PKAW 2006). LNAI 4303, Spinger, 25-29

[13] Object Management Group: Business Process Modeling Notation – Specification V 1.0. 2006. www.bpmn.org

[14] Jennings, N.R., Norman, T.J, Faratin, P., O'Brien, P., Odgers, B. 2000. Autonomous Agents for Business Process Management. In Int. Journal of Applied Artificial Intelligence 14 (2), 145-189

[15] Merz, M., Liberman, B., Müller-Jones, K., Lamersdorf, W. 1996. Inter-organisational workflow management with mobile agents in COSM. In Crabtree, B., Jennings, N.R. (Eds.). Proc. 1st Int.Conf. on The Practical Application of Intelligent Agents and Multi-Agent Technology (London, UK, 1996) The Practical Applications Co. Ltd., 405-420

[16] Joeris G. 2000. Decentralised and Flexible Workflow Enactment Based on Task Coordination Agents. In Proceedings of the 2nd Int.Bi-ConferenceWorkshop on Agent-Oriented Information Systems (Stockholm, Sweden, 2000) 41–62

[17] Agentis Software: Adaptive Enterprise™ Solution Suite. http://www.agentissoftware.com

[18] Braubach, L., Pokahr, A., Lamersdorf, W. 2005.Jadex: A BDI-Agent System Combining Middleware and Reasoning. In Umland, R, Klusch, M., Calisti, M. (Eds.). Software Agent-Based Applications, Platforms, and Development Kits. Whitestein Series in Software Agent Technology. Birkhäuser

[19] Burmeister, B., Steiert, H.-P., Bauer, T., Baumgärtel, H. 2006. Agile Processes through Goal- and Context-oriented Business Process Modeling. In Eder, J. Dustdar, S. et al. (Eds.). BPM 2006 Workshops (Wien, Austria, 2006), LNCS 4103, Springer, 215–226

[20] Rimassa, G., Calisti, M., Kerland, M.E. 2005. Living Systems™ Technology Suite. In Umland, R., Klusch, M., Calisti M. (Eds.). Software Agent-Based Applications, Platforms, and Development Kits. Whitestein Series in Software Agent Technology. Birkhäuser

[21] Greenwood, D., Rimassa, G. 2007.Autonomic Goal-Oriented Business Process Management. In Third International Conference on Autonomic and Autonomous Systems. ICAS'07 (Los Alamitos, CA, USA) IEEE Computer Society

[22] Burmeister, B., Haddadi, A., Sundermeyer, K. 1993. Generic, Configurable, Cooperation Protocols for Multi-Agent Systems. In From Reaction to Cognition: 5th European Workshop on Modelling an Agent in a Multi-Agent World MAAMAW 1993, (Neuchatel, Switzerland, 1993) LNCS 957, Springer

[23] Bauer, B., Müller, J. P., Odell, J. Agent UML: A Formalism for Specifying Multiagent Interaction. In Ciancarini, P., Wooldridge, M. (eds.) Agent-Oriented Software Engineering. Springer, 91-103

9. Author's complete addresses

Birgit Burmeister, Daimler AG Group Research, GR/EPF - HPC G018-BB, D-71059 Sindelfingen, Germany, Phone: +49 7031 4389 390, birgit.burmeister@daimler.com

Michael Arnold, Daimler AG, ITP/DP - HPC H335, D-71059 Sindelfingen, Germany, Phone: +49 7031 90 88754, michael.arnold@daimler.com

Felicia Copaciu, Daimler Protics GmbH, 628-P305, D-70546 Stuttgart, Germany, Phone: +49 7031 90 45653, felicia.copaciu@daimler.com

Giovanni Rimassa, Whitestein Technologies, Pestalozzistrasse 24, CH-8032 Zürich, Switzerland, Phone: +41 44 256 5019, gri@whitestein.com