Goal-Oriented Autonomic Business Process Modeling and Execution: Engineering Change Management Demonstration

Dominic Greenwood (dgr@whitestein.com)

Whitestein Technologies AG, Zürich, Switzerland

This demonstration paper describes the Living Systems Autonomic BPM (LS/ABPM) suite from Whitestein Technologies AG. This unique product consists of several integrated components for modelling, executing and administering business processes using a ground-breaking *goal-oriented* approach to BPM. A real and current customer case in the domain of Engineering Change Management (ECM), from Daimler AG, is used to explore the approach and features of the suite in the demonstration. Key improvements over conventional BPM techniques and technologies include business-goal oriented process modeling and extending process agility beyond the design stage by offering autonomic, self-optimizing process orchestration and execution.

The LS/ABPM suite includes a Process Modeler for goal-oriented process modeling, a Process Navigation Engine for process execution, a Management Console for process deployment and administration, and Process Task Libraries and Application Frameworks to easily build solutions for different vertical markets. The suite tools are all based on, and compliant with, standard technologies (i.e., BPMN and J2EE) allowing seamless enterprise integration.

LS/ABPM is the first BPMS to enable true goal-oriented process modeling [3]. The Goal-Oriented BPMN (GO-BPMN)[1] notation is an extension of standard BPMN, with all model elements given a precise operational semantics allowing unambiguous model execution. Process designers use the LS/ABPM Process Modeler to capture a process' business-level purpose in terms of its goals (what/why) and the plans that are capable of achieving them (how). Owing to this loose coupling of ends and means, goal-oriented process models flexibly and concisely express the available paths across business targets. This is inherently different from conventional approaches that need to model explicit process variants, thus creating more rigidity and complexity. Both business analysts and IT specialists easily grasp and use GO-BPMN models, which are directly executable for smooth testing and deployment.

The LS/ABPM Process Navigation Engine uses the Living Systems Technology Suite (LS/TS) autonomic software technology middleware to directly map process model goals onto process instance goals using a form of Belief-Desire-Intention (BDI) execution logic [2]. The engine interprets a loaded model and dynamically selects the available plans best suited to attain goal-specified business objectives in real-time. It immediately considers changes in goals, plans and environment conditions, and autonomically adapts the executing process,

constantly tracking its goals. The result is real-time process agility and responsiveness to changes in goals, plans and context-conditions as they happen during execution. This yields the greatest benefit to processes not following a strict predefined execution sequence, which is often the case for human-centric collaboration activities and increasingly also in service-oriented architectures.

The development of the LS/ABPM suite was motivated by two factors, first an observation that many enterprises have an extant desire to elevate business process modeling and management from purely the IT domain, toward providing intuitive access and use to business people who often prefer to think in terms of objectives, rather than solely the actions taken toward achieving objectives. Secondly, contemporary flexible methods of working, just-in-time organizational reaction times, distributed intra/inter-organization collaboration and constantly changing markets are creating business landscapes requiring a strong degree of real-time process agility, without sacrificing reliability or robustness. We therefore suggest that some enterprises are now discovering that a common limitation with conventional approaches to BPM is their inability to create business process models that are both meaningful to business people and capable of offering the real-time process flexibility and rapid process adaptation required to cope effectively with the pressing dynamic business conditions typifying many modern enterprises. As evidenced by our work with customers in the manufacturing domain, there is very often a need to alter executing process structures, sometimes in real-time, without perturbing the process as a whole which can continue to run as normal, accounting for updates on-the-fly. If a BPMS is not built to innately manage change in this manner the result can be reductions in both dependability and visibility, especially from a management perspective.

LS/ABPM Process Modeler

The LS/ABPM Process Modeler component of the suite provides business and IT users with a comprehensive set of tools and methodologies for goal-oriented process model design, testing, and validation. Goals and plans to intuitively express business processes LS/ABPM leverages the concepts of every-day goals and plans for a more intuitive BPM experience: first define the goals a process must accomplish and then specify the possible plans that are capable of achieving these goals.

Graphical modeling language: LS/ABPM's business process specification relies on GO-BPMN to focus on business goals and business organization. GO-BPMN extends the widely used OMG-standard BPMN with semantics for modeling goals, plans and their relationships in addition to standard BPMN elements.

Separation of goals and plans: GO-BPMN models cleanly separate the (business) goals to be achieved and the plans to achieve them. Changes to any goal or plan in a GO-BPMN process model are made independently and don't have a ripple effect of consequences as they would in a sequential process model. Hence, changes can be made at any time, even during execution. The resulting adaptability and resilience to changing business conditions save time and reduce the costs associated with business process maintenance.

Separation of process and organizational model: The Process Modeler distinguishes between the goal/plan aspect of a process and the associated organization structures and constraints: what needs to be done is not mixed with who can or should do it. Clean separation between the execution of process logic and the organization of human participants helps adapt to organizational restructuring and isolates a process' business value from human resourc deployment issues.

Accessible process models for business domain experts: Thanks to the primary focus on business goals in lieu of procedures, the resulting process models are of highly descriptive nature. This intuitive method not only supports easier changes, but also enables domain experts to directly do the modeling. LS/ABPM substantially narrows the gap between business and technology.

Modular design for cooperation: Process models are modular allowing collaboration on large models, domain-specific modules, or libraries. Different people can work on reusable modules, later consolidating results into a whole model.

LS/ABPM Process Navigation Engine

The LS/ABPM Process Navigation Engine directly executes GO-BPMN process models. It pursues the defined business goals by creating a path that takes into account model changes and plans alternatives in real-time.

Direct execution of process models: LS/ABPM's GO-BPMN process models are directly executable, and the whole user interface can be automatically generated from the process model. Domain experts can test their models on their own computer for rapid process development and consistent process lifecycle management. Round-trip engineering is intrinsic in LS/ABPM, as the suite never needs to translate between a modeling notation (such as BPMN) and an execution language (such as BPEL).

Autonomic goal-oriented process performance: Each business goal connects to one or more plans, each defining a distinctive way to achieve the goal. The Process Navigation Engine selects and orchestrates the appropriate plans in real-time based on business rules and other domain-relevant context conditions. Sanity conditions can be defined and the system ensures them through continuous monitoring and prompt corrective action.

Agile process navigation and responsiveness: Agility in LS/ABPM is based on the autonomic, real-time composition and navigation of a goal-plan-context model, not on the rigid execution of explicit, situation-specific process model variants. This offers unprecedented adaptivity to dynamic business conditions.

Active coordination and cooperation between process models: The Engine performs active coordination and cooperation between multiple process models through message-driven synchronization between process controllers. Competing goals and plans do not lead to obstruction, but are autonomically resolved.

LS/ABPM Management Console

The LS/ABPM Management Console offers powerful tools for the deployment and steering of processes and other system administration tasks.

Continuous visibility of process execution and events: LS/ABPM provides detailed monitoring of running process instances. At any point, the achieved, running, and waiting goals of a process can be inspected, as well as the corresponding pending tasks.

Systematic control of executing business processes: Supervisors can control all elements (goals, plans, etc.) to fine-tune a process during execution. Other aspects under their control include visualization, data persistency, user management, role-based assignments of personnel, and security.

Operational Overview

Once a GO-BPMN models has been automatically validated by the Process Modeler it can be loaded into the Autonomic Process Navigation Engine for execution. In this respect the model itself is directly executable with no requirement to translate it via an intermediary representation such as BPEL. The engine is composed of two layers, the LS/ABPM process navigation engine and the J2EE compliant LS/TS autonomic middleware platform. In effect, the LS/ABPM process navigation engine is an LS/TS application consisting of a collection of goal-oriented agents with BDI logic engines acting as process instance controllers. An agent controller is assigned to each process instance, responsible for coordinating the process algebra and task structuring within goal-plan combinations, taking into account goal and plan preconditions. When a process model is created using the Process Modeler it is directly loaded into a new process controller agent, wherein process goals are mapped onto logical goals within the goal-oriented execution engine provided by LS/TS. The controller then executes the process instance by initiating the entire goal hierarchy and waiting for appropriate triggers to be sensed within the system environment to activate goals and move forward with process execution. Each process controller is at the heart of an autonomic feedback control loop that uses observations made of the *system* being affected by the process instance to effect decisions within the corresponding process instance relating to, for example, which goals should be activated and which plans selected to meet goal requirements. Such autonomic control allows process instances to be self-configured and self-optimized bringing about both process flexibility and resilience.

References

1. Greenwood, D., and Rimassa, G. Autonomic goal-oriented business process management. In *Proceedings of the Third International Conference on Autonomic and Autonomous Systems (ICAS'07)*, 2007.
2. Rao, A. S. and Georgeff, M. P. BDI-agents: from theory to practice. In *Proceedings of the First Intl. Conference on Multiagent Systems*, San Francisco, 1995.
3. Tibco. Goal-driven business process management: Creating agile business processes for an unpredictable environment. *Tibco Whitepaper*, 2006.