A SEMANTIC APPROACH TO MONITOR BUSINESS PROCESS PERFORMANCE

By: Manoj Thomas, Richard Redmond, Victoria Yoon, and Rahul Singh

Thomas, M., Redmond, R. T., Yoon, V., and Singh, Rahul "A semantic approach to monitoring business process performance", *Communications of the ACM*, December 2005 Vol. 48, No. 12, p.55-59

Article:

Finding an effective method for managing and evaluating the performance of business processes is a key element for e-business success.

The productivity and profitability of organizations depend on the efficacy of their business processes. Monitoring the performance of these processes in delivering organizational value propositions provides a basis for critical managerial decision-making activities including work scheduling, capacity planning, and process design or refinement [8] . Business processes are automated in whole or in part using workflow management systems (WFMS), where documents, information, and activities flow between participants according to existing process models and rules [9]. Workflows are abstractions of business processes that are typically modeled as deterministic, action-event sequences in WFMS [8] . Organizations employ tactical WFMS with sampling-based schemes or data-driven operational WFMS using firsthand observations for process evaluation. Managing the performance and quality of business processes based on accurate performance measures has direct impact on the success of an organization [8, 10] .

Business process performance is evaluated on many levels including timeliness, stability, cost effectiveness, and utilization [2] . Measurements of business activities on dimensions including—cycle time, delay time, and cost—are essential inputs to evaluate overall process performance [10]. Performance measurement activities are integral to systems that monitor business processes [10]. Process modeling languages, including Business Process Execution Language (BPEL) based on the Web Services Definition Language (WSDL), are statically binding to executing processes at the operational level [11].Process modeling languages, including BPEL, describe the coordination flow of activities, state information, transactional state changes, and exceptions for business process activities. However, they lack the semantic expressiveness needed to describe these activities and their performance criteria in a normative manner to support inferencing on the performance of individual activities and the overall business process [8] . Assessing overall process performance by reasoning with performance metrics of individual activities requires knowledge representation of business activities and their multiple performance metrics, as well as process models that aggregate individual activity performance measures into an overall process-level view [10].

Advances in the Semantic Web and multi-agents systems offer an opportunity to effectively monitor the performance of business processes and workflow systems [2, 8, 10] . The Semantic Web [3] provides a common framework for semantics and data to be shared and reused across application and enterprise boundaries. The Semantic Web architecture defines an environment where software agents carry out sophisticated tasks for users by employing semantic documents. Central features of the semantic e-business vision include ontology, knowledge representation, and agents used in business applications [7] . Here, we show how this approach can be applied to monitor the performance of business processes. Activity performance measurement that supports management goals can be managed by a decoupled monitoring system overlaid on, instead of tightly coupled to, the business process. A loosely coupled semantic architecture overlaid upon a business process, where agents communicate to monitor business process performance from business activities within a workflow is the essence of this article.

SEMANTIC ARCHITECTURE TO MONITOR BUSINESS PROCESSES PERFORMANCE

Agent properties including *situatedness* (receiving inputs from the environment and performing actions to change the environment), *adaptability* (taking independent action without direct human intervention), and *flexibility* (interacting with and responding to other agents, human or software, in a timely fashion) are useful to design a business process performance monitoring system [4]. We show how hierarchical multi-agent architecture, loosely coupled with the WFMS, can support process managers by monitoring the performance goals of executing processes by directly interacting with WFMS process models.

Process models are expressed in standard languages (including BPEL) and business activities. BPEL describes the business process, its constituent business activities, and the transactional state changes therein using a directed graph and flow language approach. Performance criteria for business processes and activities are described using the Web Ontology Language (OWL). Agents monitor the overall business process and the performance of individual business activities based on multiple criteria using these documents. OWL documents provide semantic descriptions of the contextual requirements for agents to monitor business processes performance. With robust theoretical foundation Description Logics (DL) [1], OWL is the W3C standard for semantic knowledge representation. Tools like Protégé (protege.stanford.edu) and Racer (www.racer-systems.com) generate problem-domain ontologies from DL and verify their consistency and conformance to model requirements.

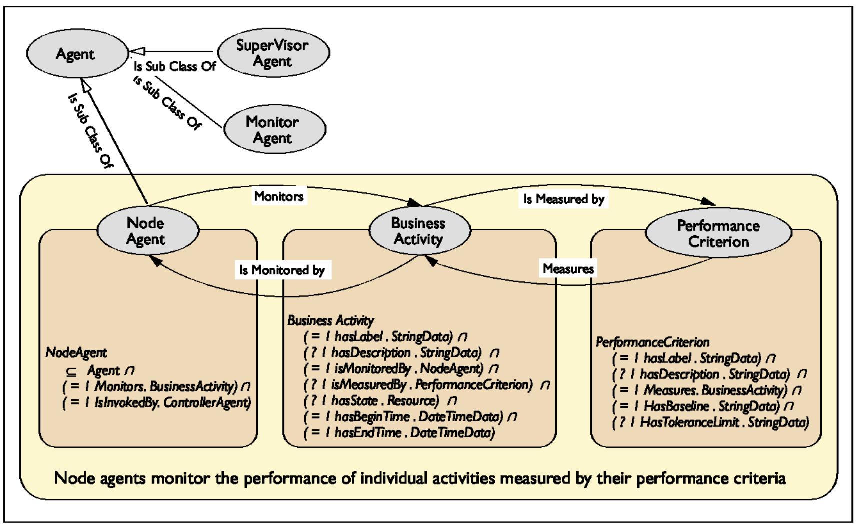


Figure 1. DL model of business activity performance criteria monitored by Node agents.

Figure 1 shows the OWL-DL knowledge representation required for agents to monitor business activities based on their performance criteria. BPEL and OWL provide the structured knowledge representation and maintain abstraction from the underlying business process. We utilize existing business process models that are either available or can be represented using BPEL. Together, BPEL and OWL provide software agents with global process knowledge, including the flow of individual business activities and the aggregate effects of their individual performance criteria on the overall business process performance.

The agent architecture and roles of each tier are described as:

Supervisory agent (SA) is the primary interaction agent for managerial decision initiatives. Inputs to SA are process performance goals defined by process managers through decision interface APIs. An SA operates at the process management level and invokes Monitor agents (MAs) to support process performance objectives. The behavior of an SA is controlled through its specifications in OWL document. An SA has access to process knowledge in BPEL documents. An SA invokes a MA for each business process performance goal and provides the MA information about the business process.

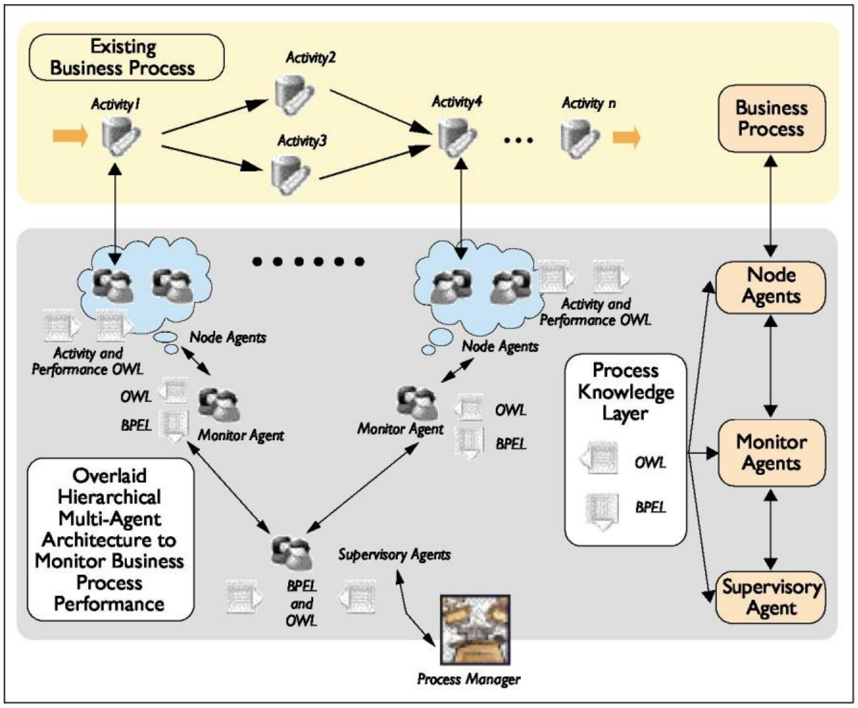


Figure 2. A hierarchical multi-agent architecture to monitor business process performance

Monitoring agents are instantiated and dispatched by SA for each specific goal (for example, quality assurance, performance assurance, node status). MAs support decision models utilizing process performance inputs from node agents (NAs). Process knowledge in the BPEL documents provide an MA with information needed to identify individual business activities in the business process. Activity-performance ontology with descriptions of the individual activities and their performance criteria for each activity in the workflow provides knowledge representation for an MA to reach inferences for a given decision predicate. An MA utilizes process knowledge to invoke NAs for each business activity described in the BPEL process model.

Node agents are data gatherers and transformers. An NA is instantiated by MAs to monitor and assess the performance for each business activity. The NAs transform process performance observations into information to support the decision models of its parent MA. The NA transformation function includes the capability to transform continuous and discrete activity observations including the state of the process with respect to established performance criteria based on established tolerance limits.

In Figure 2, a process manager interacts with the SA to specify a performance goal for the business process. The SA invokes goal-specific monitor agents that use process knowledge to invoke NAs for each business activity in the workflow instance. If new goals are defined by the decision maker for which no MA exits, their measurement criteria is defined as inputs by a human decision maker through the SA API interface. In this case, the SA modifies the SA's OWL to create definitions for new MAs to address the new process performance measurement criteria and creates a new goal-based Service reference to deploy a MA to accommodate this need. NAs are invoked to monitor every business activity, with knowledge of the activity and its performance criteria contained in OWL documents. Business logic described by the BPEL is used by the MA to determine global process performance measures using the monitored performance criteria of individual business activities reported by NAs. Each goal-specific monitor agent collects information from individual NAs using performance criteria specified in *activity -performance ontology* (described in Figure 1), and reports the process performance to the SA. The SA can provide aggregated and goal-specific process performance information to the process manager to support decision making using business process performance measures.

AN ILLUSTRATIVE EXAMPLE

To demonstrate the architecture, consider a managerial decision to monitor a specific process performance goal, such as cycle time for a workflow instance. The process manager invokes the SA and specifies the performance goal, to monitor using the SA interface API. The SA uses process performance criteria definitions, OWL, to identify the variables of interest (such as the time in queue, delay time) and process models, BPEL, to map these variables to the business process. A simplified BPEL document that represents the behavioral properties of nodes and the transitional state of the activities associated with an order fulfillment workflow is shown in Figure 3.

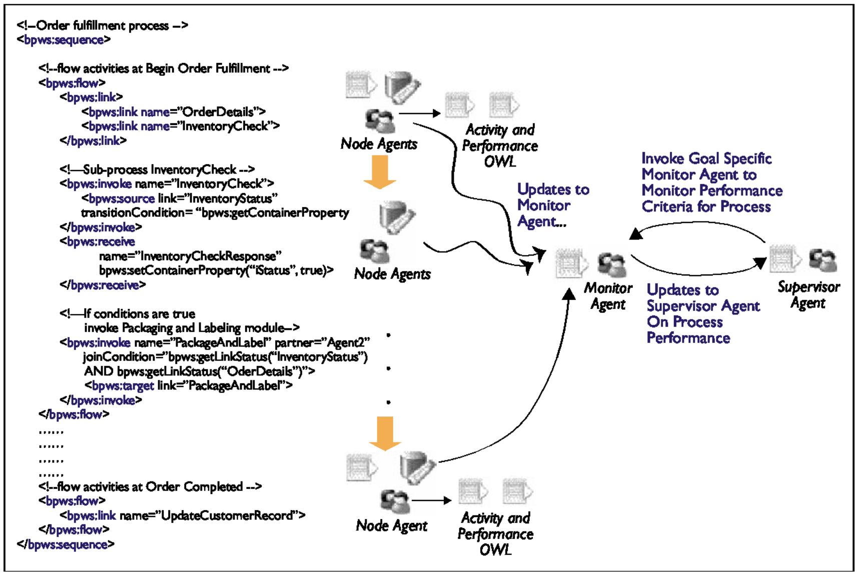


Figure 3. Semantic process performance monitoring for an order-fulfillment process e-criteria monitored by Node agents.

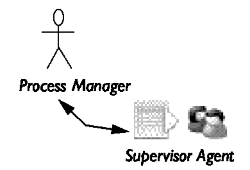
The invoked SA dispatches goal-specific MA to monitor the business process performance. Goal-specific MAs are described in the SA's OWL. The Service reference of SA's OWL forwards necessary details for the MA to work with the NAs. The table shows the communications between the process manager and individual agents at each level of the hierarchical multi-agent architecture to support the process performance monitoring objective.

Service references of the MA's OWL are used to dispatch the necessary NAs to the business activity nodes of a workflow to gather, transform, and communicate the result of the cycle time computation back to the MA. Individual activity definitions and process knowledge is available to MAs as OWL and BPEL documents. At each business activity node, the NAs process the observed quantitative data to generate state information or behavioral pattern changes for each monitored performance criteria variable pertaining to the goal of the MA. NAs perform the operational computations at the nodes defined in the compute element of the Service reference to determine the conformance of a business activity node to pre-established performance criteria, including baselines and tolerance levels, for each activity. The combination of inputs from NAs provides the decision base for the MA to develop an informed response for the state of the overall business process performance with respect to the performance criteria of process cycle time. The MA returns to the SA with an informed aggregate inference of the cycle time, based on responses from each metric that summarizes the outcome of the monitoring goal set by the decision maker.

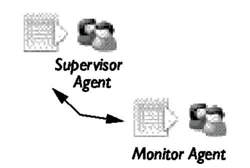
The SA and MA are aware of the overall process knowledge through the BPEL document. Individual MAs can deploy NAs to gather business process performance measurements to inform decision models employed by the MA. A MA responds to its SA by formulating an aggregate, goal-specific conclusion based on NAs responses of activity performance on cycle time at the activity nodes of the business process. Since OWL uses the same encoding style of the WSDL's SOAP binding mechanism [6, 11], each agent's role-specification agent is determined using a combination of URIs, references, and namespace definitions as illustrated in the OWL snippet listings in the accompanying table. The use of RDF/XML and URL-based reference syntax within the OWL documents provide flexibility to extend or customize the responsibility of a MA.

CONCLUSION

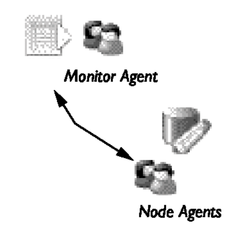
A distinguishing feature of the proposed architecture over traditional workflow management systems comes from the use of knowledge representation languages to create ontology for each business activity that can be used by agents to monitor performance of the business activity. A FIPA-compliant content language like OWL is used to create the ontology documents. One concern is the need for multiple descriptive languages (such as BPEL and OWL) in the proposed model. Semantic architecture must incorporate process models and knowledge representation of the entities and agents as well as relevant process effectiveness measurement.

<table>
<tr><td>
</td><td>A process manager (human agent) provides process performance goals for the entire process.

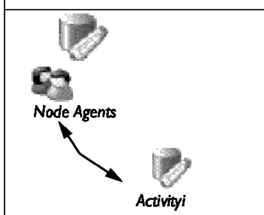
For example:
Cycle time = 10 min
Tolerance = 1 min

SA returns performance goal results back to Process manager.</td></tr>
<tr><td>
</td><td>#GoalDescription:
Cycle time, C = 10 min, Tolerance, t = 1 min
Decision model:
$C \pm t = f(c_i, t_i)$;
where c_i = cycle time for each activity, and
and t_i = tolerance limit for each activity.

<profileHierarchy:MA-Agent-1 rdf:ID="Profile-MA-Goal-Agent-1">
<!-- MA goal: Monitor node Wait time -->
<profile:hasInput rdf:resource="&MAAgentModel.owl#GoalDescription" />
<profile:hasOutput rdf:resource="&MAAgentModel.owl#Output" />
<!-- Goal based MA service -->
<service:presentedBy rdf:resource="&MAAgentModel.owl#Service" />
</profileHierarchy:MA-Agent-1>

Monitor Agent reports results back to the SA</td></tr>
<tr><td>
</td><td>#MeasureDescription:
Captures cycle time (BeginTime - EndTime) from each activity node and returns information to Monitor Agent.

Example of one node agent's OWL description.
<profileHierarchy:NA-Agent-1 rdf:ID="Profile-NA-Agent-1">
<!-- NA objective: determine wait time delta value at the workflow node i, where i is defined by MA -- >
<profile:hasInput rdf:resource="&NAAgentModel.owl#NodeID" />
<profile:hasInput rdf:resource="&NAAgentModel.owl#MeasureDescription"
<profile:hasOutput rdf:resource="&NAAgentModel.owl#Output" />
<!-- Goal based MA service -->
<service:presentedBy rdf:resource="&NAAgentModel.owl#Service" />
</profileHierarchy:NA-Agent-1>

Node Agents return individual cycle times to Monitor Agent.</td></tr>
<tr><td>
</td><td>#Service: Returns the activity cycle time to the Monitor Agent.

<service>
<description> determine the cycle time at the node </description>
<!-- direct measurements at the activity node --!>
<compute> BeginTime – EndTime</compute>
</service></td></tr>
</table>

Sequence of communications and OWL-S profile descriptions for the hierarchical multi-agent architecture.

Process modeling languages, including BPEL, lack the ability to capture semantic representations of business process activities and their performance criteria. Recent advances examine enhancing the expressiveness of Web Services [11], which provides the foundation for greater expressiveness in WSDL-derived languages such as BPEL. Future research examines the value of a single descriptive language with a vocabulary capable of combining features of business logic in BPEL and performance-monitoring logic for agents in OWL for business process performance monitoring.

The descriptive power of semantic languages can be used by agents to provide inputs for process reconfiguration decisions based on process performance measures [8, 10] . The model allows the agent communication language to remain the same while workflow improvements are effected on the system. The model presented here permits informed decision making based on the individual requirements of the specific products moving through the business process. We have utilized static descriptions of the business process, expressed in BPEL, and examined the semantic composition of processes in a dynamic manner [8]. Our future work explores how agents can dynamically adjust the variances and tolerances permitted in a business process, based on product- or customer-specific requirements.

REFERENCES

1. Baader, F., Calvanese, D., McGuinness, D., Nardi, D. and Patel-Schneider, P.F., Eds. *The Description Logic Handbook: Theory, Implementation and Applications.* Cambridge University Press, Cambridge, U.K., 2003.
2. Basu, A. and Kumar, A. Research commentary: Workflow management issues in e-business. *Information Systems Research 13*, 1 (2002), 1–14.
3. Berners-Lee, T., Hendler, J., and Lassila, O. The Semantic Web. *Scientific American 284,* (May 2001)34–43.
4. Jennings, N., Sycara, K. and Wooldridge,
M. A roadmap of agent research and development. *Int. J. of Autonomous Agents and Multi-Agent Systems 1*, 1 (1998), 7–38.
5. Kumar, A., Wainer, J. Meta workflows as a control and coordination mechanism for exception handling in workflow systems. *Decision Support Systems 40* (2005), 89–105.
6. OWL-S and WSDL; www.daml.org/services/owl-s/1.0/owl-s-wsdl.html.
7. Singh, Rahul, Iyer, L.S. and Salam, A.F. Semantic eBusiness. *Intern. J. of Semantic Web and Information Systems, 1*, 1 (Jan.–Mar. 2005), 19–35.
8. Sivashanmugam, K., Miller, J.A., Sheth, A.P. and Verma, K. Framework for Semantic Web process composition. *Intern. J. E-Commerce 9*, 2 (Winter 2004), 71–106.
9. Stohr, E.A. and Zhao, J.L. Workflow automation: Overview and research issues. *Information Systems Frontiers 3*, 3 (Sept 2001), 281–296.
10. Wang, M., Wang, H. From process logic to business logic—A cognitive approach to business process management. *Info & Mgmt*, 2005.
11. Web Service Semantics—WSDL-S; www.alphaworks.ibm.com/g/
g.nsf/img/semanticsdocs/$file/wssemantic_annotation.pdf.