appropriate supplier. These statements are always true for all orders in the system.

Unlike the situation description, the goal shows up as a policy that is currently unsatisfied in the system. In our example, it's not yet true that a bill for Dave for four GS21 tires exists. Now, the unsatisfied policy constitutes an inconsistency in our enterprise physics system's knowledge base. Such a state is undesirable. Therefore, the system seeks to overcome the inconsistency and behaves appropriately. In our example, the system proposes a plan of action that can transform the current situation to a situation that satisfies the goal policy. Thereby, the actions can be manual tasks for human actors to perform or service invocations from John's Tire Center's service-enabled enterprise system. Concretely, the system proposes to procure 100 GS21 tires from Best Tire and Rubber, send the four requested tires to Dave, and handle the billing.

A distinguishing property of our approach is the emergence of a business process based on a user-supplied business goal.

Some technical challenges

Realizing such an approach requires meeting several technical challenges—many not unlike those for successful SOA deployment.

Semantic data integration and interoperability. For our approach to be successful, we must be able to represent and describe involved entities in a consistent form that is expressive enough and yet performable and extendable for use in a real-world business environment.

Researchers have proposed many standards—usually XML-based languages—for all kinds of purposes. We must analyze which standards meet our needs and have a scope that covers descriptions of services, policies, processes, human information exchange, and so on.

In parallel to developing standardized modeling languages such as the Web Services Business Process Execution Language (WSBPEL; www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsbpel), Process Specification Language (PSL; www.mel.nist.gov/psl), or Web Services Description Language (WSDL; www.w3.org/TR/wsdl), we suggest developers work on a unified lingua franca between humans and machines, considering alternative logics as temporal or modal logic. They might also develop such a language for interaction between machines, allowing a better inspection by humans using a controlled natural language.[10]

Dynamic Web service integration. A distinguishing property of our approach is the emergence of a business process based on a user-supplied business goal. This business goal is realized by a business process that emerges from the integration of single services into an appropriate sequence or more complex arrangement. An appropriate planning functionality promotes this emergence. So, we need a planner powerful enough to build business processes from services, including service discovery, consumption, composition, and verification. In real-world applications, it's not enough to describe services solely by their functional properties.[11] Rather, side effects and higher-level information are crucial. We suggest work toward more real-world-appropriate service specifications. In particular, the planner must observe policies imposing business-critical constraints on the composition—that is, the planning task.[12]

Efficiency is an important issue. Not every business goal can be realized only by a brand-new business process. Rather, reuse and adaptation of available business processes are equally important problems of high real-world value.

Enterprise policy research. Research into enterprise policies comes in various forms. Different enterprises and even departments within an enterprise have different policies. Thus, dynamic business policy integration is necessary. The organization must meet its goals while complying with individual policies. Challenges include data and rule inconsistency and incomplete information.

Policy acquisition, formalization, modification, and monitoring are also essential. How can users state correct policies, monitor their behavior, and intervene in conflicts? Can the system derive policies automatically from existing processes?

Although this research has just seriously begun, there are already usable results from computational logics, operational research, even research in legal expert systems and related subjects. Moreover, researchers have already used these results in real-world scenarios.[12]

Human-machine interaction. Especially important is how human users will work with a system based on our approach. What are the benefits, and what are the challenges when humans interact? Although the challenges mentioned so far aren't exclusive to a policy-oriented enterprise management paradigm, the benefits and interaction scenarios are. Prominent issues include the following:

- *Goal identification.* Business process goals are often complex, and it's difficult for users to state them correctly. How can the system help the user determine correct (potentially) achievable goals based on an initial situation and available resources (services, policies, and so on)?
- *Policy selection, acquisition, and monitoring.* A user rarely explicitly formulates policies. How can the system acquire policies? How can the user monitor them for correctness? How can the user intervene when policy application conflicts occur?