

# Internet-Based Support for Process-Oriented Instant Virtual Enterprises

The ever-increasing complexity of contemporary products and services demands business supply chains that ultimately involve a large number of autonomous organizations. Competitive markets require these chains to be highly agile, effective, and efficient, which organizations can achieve by forming dynamic virtual enterprises within supplier networks, called instant virtual enterprises (IVEs). The authors present the CrossWork system, which helps these organizations efficiently create and operate IVEs by providing automated, Internet-based support for the composition, setup, and execution of global business processes. This article uses the automotive domain to illustrate the system's optimal use.

In the past, businesses often operated in stand-alone mode or relied on rather static networks of cooperating organizations. Today, competition has become fiercer, fueled by relatively recent developments such as the globalization of business, shortened time-to-market requirements, and increased market transparency. Products and services to be delivered to customers are, on one hand, quickly increasing in complexity and, on the other, subject to increasingly frequent modifications and replacements. To comply with these market settings, organizations must shift their priorities to flexibility and the ability to change if they want to survive. Therefore, dynamically established cooperation among orga-

nizations is becoming much more of a requirement to meet market demands.

Accordingly, we've seen the emergence of new business models focused on new business abilities that rely on intensive collaboration between autonomous business entities based on dynamic partnering. We call this model the *instant virtual enterprise* (IVE) – a temporary business entity executing a dynamically composed, global business process to achieve a specified business goal. Compared to a traditional VE, an IVE is more dynamic and must rely more on its members' existing business processes. It's more dynamic in the sense that its life span typically is on the order of days to months, whereas traditional VEs have a life span on the

**Paul Grefen and Rik Eshuis**  
Eindhoven University of Technology

**Nikolay Mehandjiev**  
University of Manchester

**Giorgos Kouvas**  
Exodus

**Georg Weichhart**  
Johannes Kepler Universität Linz

## Related Work in Interorganizational Process Management and VE Support

The CrossWork system in its instant virtual enterprise (IVE) context is designed to provide an integrated approach that goes significantly further than existing systems for supporting interorganizational business processes and VEs.

An early effort in the interorganizational workflow management field is the Workflow Based Internet Services (WISE) project, which aimed to provide a software platform for process-based, business-to-business electronic commerce.<sup>1</sup> WISE, however, relied on static business consortia, in which business processes are designed manually. The CrossFlow project<sup>2</sup> made a step toward dynamic collaboration by developing concepts and technology for workflow support in dynamic VEs. However, these VEs are limited to dynamic service outsourcing between pairs of organizations with standardized services within a business domain. As such, the approach is too limited for complex IVEs in which arbitrary numbers of organizations collaborate in a peer-to-peer fashion to produce complex products and services. Several recent research efforts have attempted to develop flexible workflow support — for example, ADEPT (Application Development Based on Encapsulated Pre-modeled Process Templates)<sup>3</sup> — but, in general, they focus on the flexible specification and execution of single workflows, whereas IVE support requires the flexible integration of multiple workflows across autonomous organizations.

Researchers have proposed software agents to manage business processes, using autonomy and negotiation capabilities to determine the best course of action under unexpected conditions;<sup>4</sup> they've also developed agent-based approaches in the area of interorganizational business process management.<sup>5</sup> But such approaches often specify local interaction protocols between individual agents instead of focusing on explicit global business processes. In our work, we consider explicitly specified global business processes as the starting point for running

IVEs. In other words, our approach relies on well-defined global behavior across collaborating parties, rather than emerging behavior generated by the local behavior of collaborating parties.

Researchers have also proposed agents to support various e-business scenarios<sup>6</sup> — more specifically, to support the formation and operation of virtual organizations (VOs). The use of agents allows splitting VO construction tasks into subtasks and inviting bids by other agents to take on these subtasks. In one work,<sup>7</sup> an agent receiving such an invitation can propose a ready-made service and a team or decompose the invitation in turn. These decisions are informed by constraint-based reasoning and quality-of-service provisions. In contrast to this service-based approach focusing on VO creation, CrossWork provides a process-based approach covering both creation and operation of VOs.

### References

1. G. Alonso et al., "WISE: Business to Business E-Commerce," *Proc. 9th Int'l Workshop on Research Issues in Data Eng.*, IEEE CS Press, 1999, pp. 132–139.
2. P. Grefen et al., "CrossFlow: Cross-Organizational Workflow Management in Dynamic Virtual Enterprises," *Computer Systems Science & Eng.*, vol. 15, no. 5, 2000, pp. 277–290.
3. K. Göser et al., "Next-Generation Process Management with ADEPT2," *Proc. BPM Demonstration Program at the 5th Int'l Conf. Business Process Management (BPM 07)*, Springer, 2007, pp. 3–6.
4. N.R. Jennings et al.; "Autonomous Agents for Business Process Management," *Applied Artificial Intelligence*, vol. 14, no. 2, 1999, pp. 145–189.
5. M. Huhns and L. Stephens, "Automating Supply Chains," *IEEE Internet Computing*, vol. 5, no. 4, 2001, pp. 90–93.
6. M.P. Papazoglou, "Agent-Oriented Technology in Support of e-Business," *Comm. ACM*, vol. 44, no. 4, 2001, pp. 71–77.
7. T.J. Norman et al., "Agent-Based Formation of Virtual Organizations," *Knowledge-Based Systems*, vol. 17, nos. 2-4, 2004, pp. 103–111.

order of months to years. Moreover, IVEs must rely more on existing — and hence typically heterogeneous — business processes because their short life span doesn't allow adjustment of local business processes (typically neither time- nor cost-wise). Hence, an IVE poses more demands on its infrastructure than a traditional VE.<sup>1</sup>

An explicit process-oriented approach is of paramount importance to make an IVE work effectively.<sup>2</sup> Given its inherent complexity, automated support for this process-oriented approach is required to make it work efficiently. However, an IVE's dynamic and transient character makes traditional business process management technology extremely inadequate, so this approach calls for a more advanced, end-to-end solution.

In this article, we present a system for automated support of IVEs from their construction up to their operation. A consortium of research institutes, software development firms, and industrial end-user organizations has already realized this system in the European CrossWork project.<sup>3</sup> We show here how separation of concerns in the IVE life cycle leads to a high-level architecture; we also discuss the implications for the selection and combination of information technology to support an IVE. Essentially, we've found that a single technology paradigm doesn't provide the ideal basis for IVE support, but a combination of workflow, agent, and service technologies on top of an Internet infrastructure constitutes a good hybrid platform. As such, the CrossWork system uses approaches from several

technology domains; a brief overview of related research efforts appears in the sidebar.

### The Instant Virtual Enterprise Concept

Tight collaboration in a VE implies that collaborating business organizations' local processes must synchronize at a possibly highly detailed level. This means that local business processes must actually be "woven" into a global business process.

Because they operate in frequently changing markets, VEs can't maintain a stable character over time: changing market demands require adding new business competences, rendering existing business competences obsolete, or forcing different selection criteria for quality-of-service parameters. An IVE provides a solution by dynamically selecting collaborating partners and weaving the interorganizational links between their local processes. These local business processes usually remain stable over time because they depend heavily on previous investments (for example, physical production infrastructures).

Figure 1 shows an example IVE in a business market consisting of a set of possible business partners (seven are shown, but in practice, the number is usually much greater). Here, four organizations have organized into an IVE, each with their local business processes shown inside the ellipses. The local business processes of the partners within the IVE are organized into a global process by adding the links between the ellipses in such a way that the global process design takes process dependencies into account and meets overall process-quality requirements (such as throughput times).

We can apply IVEs in many dynamic markets – for example, to produce a specific series of complex products (such as automobiles), to provide complex services in a temporary market situation (such as complex financial services), or to handle dynamically occurring large-scale events (such as the operation of major sports or entertainment events). The trigger to create an IVE is a new business opportunity in a specific market, as observed by an organization operating in that market. As input for the IVE creation process, this organization specifies this opportunity as a concrete, high-level business goal. This process then goes through four phases in a semiautomated fashion:

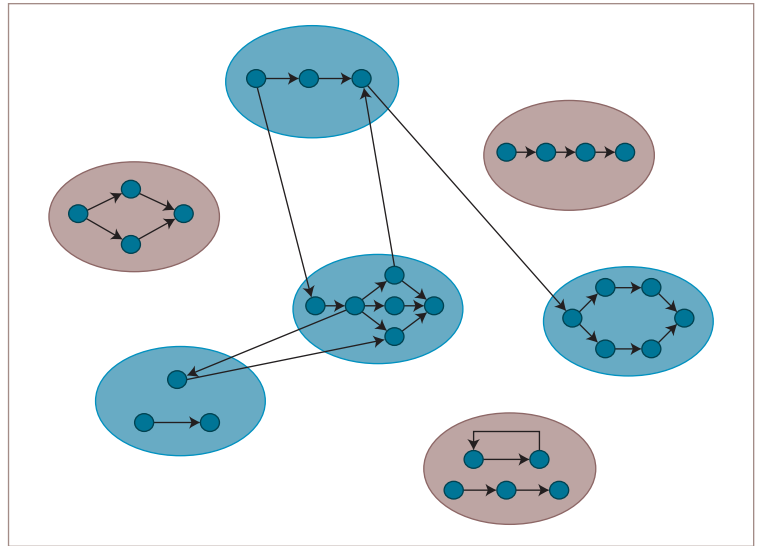


Figure 1. Instant virtual enterprise (IVE). Four organizations have formed into a virtual group; their local processes (shown inside the ellipses) have been linked into a global process.

1. The CrossWork system decomposes the high-level goal into a set of operational business goals with the assistance of a domain expert. The decomposition uses generally accepted knowledge about the domain in which the market operates – for example, in an industrial manufacturing market, this knowledge can be contained in an extended bill of materials.
2. For each identified operational business goal, the system, with the assistance of a market expert, identifies a collaboration partner in the market that can fulfill this goal (possibly the organization that initiated the IVE creation itself). This identification is based on potential partners' capabilities as well as location and quality-of-service attributes.
3. The system retrieves the external specifications of the selected partners' local business processes. These specifications are abstractions from the actual processes in these organizations, so they hide sensitive or irrelevant details.<sup>4</sup> The system then composes the local business processes into a global process by weaving interorganizational control flows between them (as illustrated in Figure 1), assisted by a business process designer.
4. Finally, the system maps the composed global business process onto the IVE's distributed infrastructure and executes it there. One of the IVE partners performs the task of global process coordinator (possibly the IVE initiator); all the other partners contribute by executing

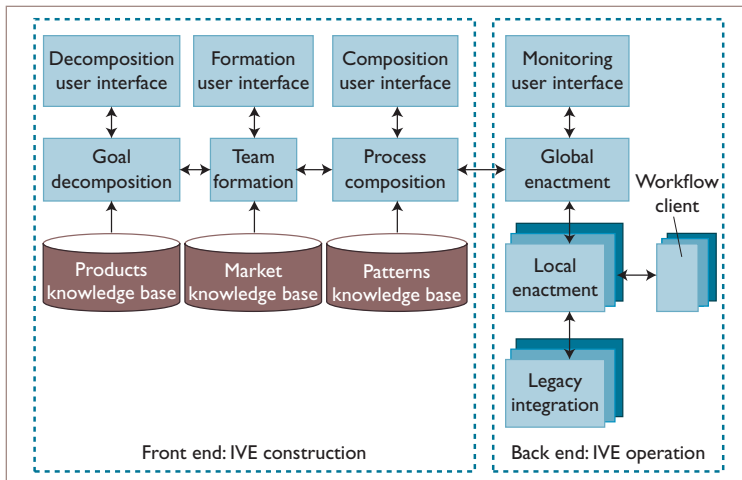


Figure 2. CrossWork architecture. Each vertical column in the architecture coincides with a phase in the instant virtual enterprise (IVE) life cycle.

their respective local business processes and synchronizing with the global process.

This life cycle isn't strictly linear, however – something can go wrong in any phase. For example, it might not be possible to form an acceptable team in the team-formation phase on the basis of the specified goal decomposition. Likewise, it might not be possible to construct an acceptable global business process based on the selected team members' local processes. In such cases, it's necessary to revert to the previous life-cycle phase and redo the work there.

We can best illustrate the IVE concept with an example. In the CrossWork project, we used supply chains in the automotive industry as the application area. VEs are common in the complex supply chains required to produce today's automobiles, and the fact that the automotive market has become increasingly dynamic, as a consequence both of global competition and a shortening life span of product types, underscores the fact that these VEs must in turn become more dynamic as well. This led naturally to our concept of an IVE operating in so-called Networks of Automotive Excellence (NoAEs).<sup>3</sup> In the CrossWork project, we conducted several case studies, but for the sake of brevity, we describe a simple case study here to apply the CrossWork system in a real-world context: a major European truck builder that assembles preconstructed modules to build its trucks.

A simple example of such a module is a water tank that consists of five main parts, each of which is produced by a specialized company:

a tank body, a motor pump, a grommet, a dispenser, and a sealing ring. Apart from this, a production must be prepared and a water pump assembled from its constituents. In this case study, "produce a water tank" is a global goal that the truck builder gives to a supplier operating in a NoAE. Using the bill of materials for the water tank, the supplier uses the CrossWork system to decompose the global goal into operational subgoals, find collaboration partners in the NoAE that can fulfill these subgoals, and weave their local processes into a global, IVE-level process. In the global process, the system can choose parallelism to produce the water tank parts because there are no dependencies between local processes. More complex examples, however, might require synchronizations, leading to complex control flows between local processes. After verification and prototyping, the system executes the newly formed global process on the IVE's distributed infrastructure.

### The CrossWork System

The CrossWork system's architecture is closely related to the IVE life-cycle structure: each of the life cycle's four phases is explicitly supported by modules in the architecture (see Figure 2, where each vertical column coincides with a life-cycle phase).

The first three columns together form the CrossWork architecture's front end, which supports IVE construction. The modules rely on knowledge used for automated reasoning, as depicted by the knowledge bases coupled with the modules. CrossWork works with different organizations, which naturally have different knowledge formats and knowledge bases. To kick-start itself, the system imports core concepts and reference knowledge structures<sup>5</sup> into the CrossWork knowledge bases for the reasoning mechanisms. The modules supporting these mechanisms are designed to query an organization when a module encounters gaps in a knowledge base (using the user interfaces shown). The answers are recorded into the knowledge base, thus gradually filling the gaps and increasing the level of automation possible through CrossWork.

The fourth column forms the architecture's back end, which supports IVE operation – that is, global business process execution. The back end has three main modules. The *global enactment module* is responsible for process management at the IVE level (for interorganizational

process synchronization); it has a user interface for global process monitoring. The *local enactment modules* are responsible for enacting local processes within a single IVE member's boundaries; these modules are workflow servers that connect to several workflow clients at the member. These workflow clients are the interactive interfaces to human workers in the IVE process. Finally, the *legacy integration module* provides interfaces to legacy systems run by individual IVE members.

We chose bidirectional interfaces in the link from goal decomposition to the global enactment modules to support the life-cycle backtracking, as discussed earlier: if a module can't fulfill its task, it calls back the previous module and requests a rework. The bidirectional interfaces in the right-most column of Figure 2 are needed for synchronization during process enactment. Note that the interface between the process-composition and global enactment modules is of a special kind; we discuss this so-called paradigm bridge in more detail later.

Obviously, automated support for an IVE must be highly distributed, given that the IVE itself consists of possibly many distributed, autonomous parties with preexisting local information systems that must be integrated into the global process. The CrossWork system's bottom-level communication infrastructure layer is rooted in the Internet, so the use of standards such as HTTP in this infrastructure provides the communication basis for application functionality in the higher layers.

CrossWork's application level has different requirements for the front- and back-end platforms. On one hand, the front end has requirements in the fields of goal orientation and support for the reasoning mechanisms needed to implement the IVE construction algorithms. For this reason, we chose a multi-agent system (MAS) platform as the basis for the front-end application layer (more specifically, the JADE platform; <http://jade.tilab.com>). On the other hand, the back end has requirements with respect to portability and interoperability to support IVE process enactment in a distributed, heterogeneous environment. To comply with contemporary systems integration practice, we use service-oriented computing (SOC) technology as the basis for the back-end application layer, employing technology from the Web services stack.

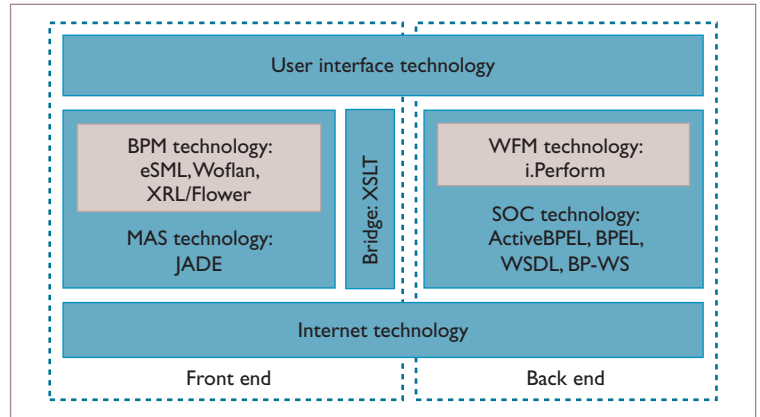


Figure 3. CrossWork technology stack. Because the front- and back-end requirements vary so widely, we use a multi-agent system platform side by side with service-oriented technology.

After choosing these two platforms, the question remained how to combine them. One option would have been to encapsulate agent-oriented modules into service-oriented wrappers. Another would be to encapsulate service-oriented modules into agent-oriented wrappers. However, because the front- and back-end requirements are so distinctly different, we decided to avoid the encapsulation option and use MAS and SOC technology side by side, as in Figure 3. In the MAS environment, business process management technology is embedded for process identification, composition, verification, and prototyping, whereas in the SOC environment, workflow management technology is embedded for local business process execution.

Next, we needed to choose the system's specification languages. The front end required something expressive enough to specify all relevant aspects of IVE configurations. For this purpose, we developed the electronic sourcing markup language (eSML) language.<sup>6</sup> Extendable expressiveness and easy manipulation are its main design criteria; the language is XML-based to fit the Internet environment. eSML also uses the business process specification language XRL<sup>7</sup> as its control flow core. The back end requires interoperability and portability with respect to global process engines, so the Business Process Execution Language (BPEL) was the obvious choice ([www.oasis-open.org/committees/tc\\_home.php?wg\\_abbrev=wsbpel](http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsbpel)).

Given the differences in the platform and specification languages between the front and back ends, we needed an interface to close the gap. Our XSLT-based paradigm bridge accord-

ingly translates eSML specifications into BPEL and communicates them between the MAS and SOC environments. Architecturally, we placed the paradigm bridge in the front end to allow a direct link to standardized BPEL back-end engines.

### The Agent-Oriented Front End

CrossWork's front end supports dynamic IVE creation tasks by using agent technology. As discussed earlier, these tasks include goal decomposition, team formation, and global process composition.

The goal decomposition module supports semiautomatic decomposition of business goals.<sup>8</sup> The goal can be well-defined – for instance, an order specification might contain a full bill of materials, such as in the water tank example. In this case, goal decomposition follows the bill of materials' tree structure: constructed subgoals correspond to the preparation of production (specifically, the production of the water tank parts, and the assembly of those parts into the completed water tank). However, in other cases, the goal can be quite vague, allowing substantial flexibility in how it can be fulfilled.

The team-formation module supports semiautomatic formation of IVE teams by using two complementary reasoning approaches.<sup>8</sup> The module can compose a team by following a centralized top-down approach, which is good for saturated markets and well-specified goals allowing decomposition into detailed services. In the case of sparse markets in which services aren't predefined, we can use a bottom-up composition approach based on a blackboard communication paradigm. Agents representing service providers observe the requested services advertised on this blackboard and identify when they might be able to provide a partial solution for a given service. If they're interested in providing this service, they display their partial solution on the board, inviting complementary partial solutions from other providers. By filling the market knowledge base, this approach supports a gradual transition toward fully automated team formation as well. This blackboard communication paradigm is based on agent-oriented standards in which agents exchange service requests and offers. Note that service-oriented standards such as UDDI are based on publishing existing services and aren't geared toward requesting new ones, hence they aren't usable here.

The process-composition module has the task of building global processes by combining local processes and assuring their quality. Process composition takes potential IVE members' local processes as given, so IVE design obtains flexibility by intelligently interweaving several local processes. This form of process composition distinguishes CrossWork from other approaches that use simpler composition (such as strict client-server composition<sup>9</sup>). CrossWork's approach to process composition consists of three steps.<sup>10</sup> First, it derives dependencies between activities based on types of input and output messages. Next, it types these abstract dependencies with concrete branching types such as AND and XOR to indicate alternative respectively parallel branches. Finally, it uses concrete dependencies to compose activities into a block-structured process model that allows easy mapping to standard process-definition languages. Using a growing set of business process composition patterns and process-composition algorithms, a gradual transition toward fully automated composition is possible. Where automated algorithms<sup>10</sup> can't solve a given problem, the process-composition module calls for the help of a human process engineer through a dedicated user interface.<sup>3</sup> The software agent containing the composition algorithm can compute required global process qualities, such as a global process's maximum throughput time, to assist in achieving optimal process configurations.

Given the dynamic and short-lived character of IVEs, we can't extensively assess composed processes in field tests. This implies that we must place more emphasis on automated support for quality control in the process-composition phase, which is why the process-composition module contains process-verification and prototyping functionality. When a composed global process fails at a verification or prototyping test, the module automatically requests a new process composition or even team formation. The module performs automated process verification (for example, for correct termination) by using the Woflan process-verification tool.<sup>11</sup> It can also use the lightweight prototyping platform XRL/Flower<sup>7</sup> for dynamic process evaluation. The fact that eSML uses XRL as its control flow core enables easy integration of XRL/Flower into the CrossWork system.

### The Service-Oriented Back End

The CrossWork back end supports the execution of dynamically composed global business processes in IVEs using SOC technology. We adopted a two-level approach to process execution in CrossWork: global process orchestration and local process execution. In so doing, we obtain both a separation of concerns between IVE level process synchronization and local business process execution on one hand and an architectural decoupling between global and local process-management technology on the other.

The global enactment module performs the global process orchestration. The basis for this module is a standard BPEL engine – Active BPEL ([www.activebpel.org](http://www.activebpel.org)) in this case. The fact that we use a paradigm bridge allows independent technical choices for dedicated process manipulation in the front end and off-the-shelf process-enactment technology in the back end. Consequently, we can accommodate the deployment of other process engines by simply changing the bridge. The global enactment module is coupled to a user interface for IVE-level process monitoring; this interface displays the global IVE process’s state. The local enactment module provides the local process execution functionality. This module is based on a standard workflow management system – i.Perform ([www.exodus.gr](http://www.exodus.gr)) in this case.

For the design of the bidirectional interface between the global and local enactment modules, we followed the Business Process Web Services (BP-WS) approach,<sup>12</sup> which extends basic Web services with process-oriented elements in the form of business process specifications and business process states that other Web services can access through a set of dedicated interfaces (ports). Specifically, we used four BP-WS classes and their different control-flow interface levels<sup>12</sup> as the basis for the CrossWork system: black box, glass box, half-open box, and open box. In the black-box class, process structures and hence process-instance states remain opaque to other organizations – consequently, the class doesn’t support fine-grained process-based interaction. In the glass-box class, process structures and states are visible to external parties, but those parties can’t influence these states – that is, the class supports synchronization by observation. The half-open box class allows external

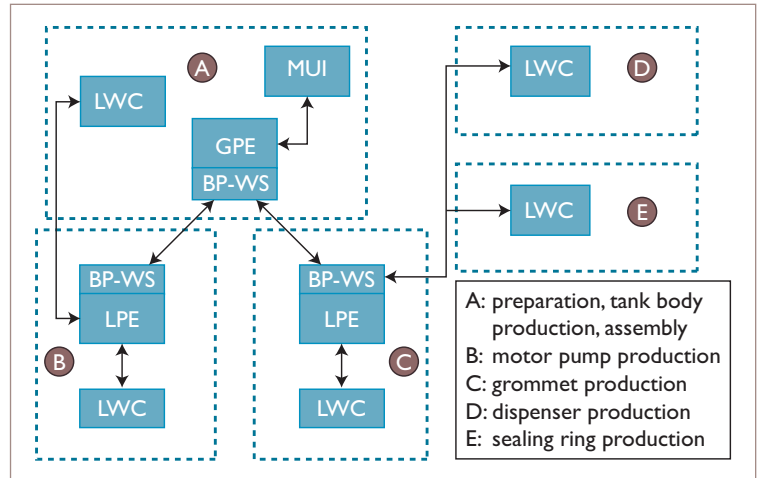


Figure 4. Example enactment topology. The business process of an IVE of five partner organizations is supported by a configuration of one global process engine (GPE) and two local process engines (LPEs). The GPE is coupled to a monitoring user interface (MUI), and the LPEs to local workflow clients (LWCs) in all five organizations.

control over process states in one direction, whereas the open box class allows it in both directions. This means that these classes support synchronization by active interference. This interorganizational process control is supported by offering process-control primitives (such as “pause process” and “resume process”) in the interface to other parties. Using these primitives enables the IVE’s fine-grained, process-based interaction. Each IVE member can decide whether to expose specific details of its services by using the BP-WS classes. Members specify selected internals of their business processes to the outside world using eSML (for process composition in the front end) or BPEL (for monitoring and control in the back end).

The local enactment module supports Web-based remote clients. This remote client technology lets process engines reside at different IVE members (other than the clients they serve), enabling a workflow application service provider topology in which partners with limited automated process-management resources can be included in IVEs with high levels of process automation. This possibility is essential to support free composition of IVEs in application domains in which automated business process support technology isn’t yet generally available at all partners. The automotive domain targeted in the CrossWork project exemplifies this concept; Figure 4 shows an enactment topology that we could use for the water tank example we introduced earlier. In

the figure, we see five autonomous organizations, A through E, which together form a simple IVE. Each organization produces one of the five parts of the water tank (as shown in the legend); organization A also takes care of the production preparation and water tank assembly. The CrossWork system enacts the global process on the global process engine located at organization A, which monitors it through the monitoring user interface. All five partners enact the local processes by using local workflow clients connected to local process engines at organizations B and C.

The last module in the back-end architecture is the legacy integration module, which enables flexible coupling of IVE members' legacy systems, such as enterprise resource planning systems. Tight coupling of these systems to IVE processes is of major importance for an IVE's effective operation: legacy systems are often the core support for IVE partners' primary business processes. The legacy integration module's implementation relies on several technologies. It uses the Java-based J2EE Connector Architecture technology solution to connect application servers and enterprise information systems and Enterprise Java Beans as server-side components to encapsulate applications' business logic. It also uses Apache Axis to provide a Web service interface and to develop client classes for Web service connectors.

**C**rossWork is a new step toward end-to-end support for dynamically created, process-centric IVEs. This approach truly is end-to-end because it includes all tasks, from IVE goals manipulation up to IVE process execution.

So far, we've tested the CrossWork system in several automotive application scenarios, a simple example of which we described here. However, the CrossWork approach and architecture are highly modular, allowing for extension in several directions. Most notably, more advanced algorithms for increasing agent intelligence in the front-end modules are an interesting direction for further research. We're currently applying the CrossWork approach in additional research. One direction is extending the approach's functionality and the other is applying it in domains with specific requirements, such as the financial industry and healthcare. □

### Acknowledgments

The European Commission supported CrossWork in its 6th Information Society Technologies (IST) Framework. We thank all project members for their contribution.

### References

1. L.M. Camarinha-Matos and H. Afsarmanesh, "Elements of a Base VE Infrastructure," *Computers in Industry*, vol. 51, no. 2, 2003, pp. 139–163.
2. P. Grefen, "Towards Dynamic Interorganizational Business Process Management," *Proc. 15th IEEE Int'l Workshops Enabling Technologies: Infrastructures for Collaborative Enterprises*, IEEE Press, 2006, pp. 13–18.
3. P. Grefen et al., "Dynamic Business Network Process Management in Instant Virtual Enterprises," *Beta Working Papers*, vol. 198, Eindhoven Univ. Tech., 2007.
4. P. Grefen, H. Ludwig, and S. Angelov, "A Three-Level Framework for Process and Data Management of Complex E-Services," *Int'l J. Cooperative Information Systems*, vol. 12, no. 4, 2003, pp. 487–531.
5. I. Stalker and N. Mehandjiev, "A Devolved Ontology Model for the Pragmatic Web," *Proc. 1st Int'l Pragmatic Web Conf.*, German Computer Soc., 2006, pp. 38–52.
6. A. Norta and P. Grefen, "A Framework for Specifying Sourcing Collaborations," *Proc. European Conf. Information Systems*, AIS, 2006, CD-ROM.
7. A. Norta, "Web Supported Enactment of Petri-Net Based Workflows with XRL/Flower," *Proc. Int'l Conf. Application and Theory of Petri Nets*, Springer-Verlag, 2004, pp. 494–503.
8. M. Carpenter, N. Mehandjiev, and I.D. Stalker, "Flexible Behaviours for Emergent Process Interoperability," *Proc. 15th IEEE Int'l Workshops Enabling Technologies: Infrastructures for Collaborative Enterprises*, IEEE Press, 2006, pp. 249–254.
9. P. Grefen et al., "CrossFlow: Cross-Organizational Workflow Management in Dynamic Virtual Enterprises," *Computer Systems Science & Eng.*, vol. 15, no. 5, 2000, pp. 277–290.
10. R. Eshuis, P. Grefen, and S. Till, "Structured Service Composition," *Proc. 4th Int'l Conf. Business Process Management*, Springer-Verlag, 2006, pp. 97–112.
11. H.M.W. Verbeek, W.M.P. van der Aalst, and A. Kumar, "XRL/Woflan: Verification and Extensibility of an XML/Petri-Net-Based Language for Interorganizational Workflows," *Information Technology and Management J.*, vol. 5, nos. 1–2, 2004, pp. 65–110.
12. P. Grefen et al., "An Analysis of Web Services Support for Dynamic Business Process Outsourcing," *Information and Software Tech.*, vol. 48, no. 11, 2006, pp. 1115–1134.

Paul Grefen is a full professor in the School of Industrial

Engineering at the Eindhoven University of Technology, where he chairs the information systems sub-department. His research interests include architectures of complex information systems, interorganizational workflow management, and high-level transaction and contract management in electronic business. Grefen has a PhD in computer science from the University of Twente. Contact him at [p.w.p.j.grefen@tue.nl](mailto:p.w.p.j.grefen@tue.nl).

**Rik Eshuis** is an assistant professor at the Eindhoven University of Technology. His research interests include process-oriented information systems and service-oriented computing. Eshuis has a PhD in computer science from the University of Twente. Contact him at [h.eshuis@tue.nl](mailto:h.eshuis@tue.nl).

**Nikolay Mehandjiev** is a senior lecturer in the Manchester Business School at the University of Manchester. His research interests include the provision of flexible and intelligent software using software agents and service-oriented software. Mehandjiev has a PhD in computer

science from the University of Hull. Contact him at [n.mehandjiev@manchester.ac.uk](mailto:n.mehandjiev@manchester.ac.uk).

**Giorgos Kouvas** is a researcher, technical coordinator, and project manager at Exodus. His research interests include workflow management systems, context-aware applications, mobile technologies, sensor networks, and data fusion. Kouvas has an MSc in electrical and computer engineering from the National Technical University of Athens. Contact him at [gkou@exodus.gr](mailto:gkou@exodus.gr).

**Georg Weichhart** is a researcher of communications engineering in the Department of Business Information Systems at Johannes Kepler University Linz. His research interests include business information systems and enterprise network interoperability. Weichhart has an MSC in business information systems from Vienna University. Contact him at [georg.weichhart@jku.at](mailto:georg.weichhart@jku.at).

**cn** Selected CS articles and columns are also available for free at <http://ComputingNow.computer.org>.

## Running in Circles Looking for a Great Computer Job or Hire?



The IEEE Computer Society Career Center is the best niche employment source for computer science and engineering jobs, with hundreds of jobs viewed by thousands of the finest scientists each month - **in Computer magazine and/or online!**

 [careers.computer.org](http://careers.computer.org)  
<http://careers.computer.org>

- > Software Engineer
- > Member of Technical Staff
- > Computer Scientist
- > Dean/Professor/Instructor
- > Postdoctoral Researcher
- > Design Engineer
- > Consultant

The IEEE Computer Society Career Center is part of the *Physics Today* Career Network, a niche job board network for the physical sciences and engineering disciplines. Jobs and resumes are shared with four partner job boards - *Physics Today* Jobs and the American Association of Physics Teachers (AAPT), American Physical Society (APS), and AVS: Science and Technology of Materials, Interfaces, and Processing Career Centers.

 IEEE  
computer  
society