



Process Management Tools

Félix García, Aurora Vizcaíno, and Christof Ebert

Developing software is complex, especially with many interacting people and teams. A variety of tools exist to model the development process and thus facilitate communication, automation, and collaboration. This installment looks to support tools for process modeling and their underlying methodologies. It draws from trends discussed at recent IEEE International Conference on Global Software Engineering (ICGSE) conferences. I look forward to hearing from both readers and prospective authors about this column and the technologies you want to know more about. —*Christof Ebert*

AS A SOFTWARE practitioner, you know from your own experiences that the quality of the software created and delivered heavily depends on the quality of the underlying development process. To best introduce, communicate, use, and improve this development process, a variety of tools exist to support processes.¹⁻³ Process management tools support software teams in three ways:

- *Modeling.* Define consistent process models, thereby reducing any ambiguity about how the work should be done because well-defined and documented processes must be defined and consistently followed across all sites.
- *Collaboration.* Share information and knowledge among different sites, thereby avoiding repeated mistakes or inconsistencies between different versions of information.
- *Automation.* Provide tools to automate a process by means of workflows, thereby increasing engineering and development productivity across the life cycle because you're sharing information across tools, orchestrating interfaces, and maintaining work products.

In this column, we outline a brief explanation about process modeling and management.

Process Modeling

Process modeling facilitates the human understanding and communication of processes, their automation, and their improvement. Although we can document simple processes on a sheet of paper, engineering processes often suffer from inadequate support, and thus aren't used or continuously improved. Process modeling helps in training processes, identifying weaknesses, seeing how different roles collaborate, and ultimately federating or integrating with engineering tools. In particular, distributed teams and ecosystems with several companies benefit from such tools.^{2,3}

We can group process models into two main categories. *Descriptive models* are aimed at describing processes and organizational behavior in terms of entities (activities, roles, tools, artifacts, and so on) and the relationships among them; *active models* are intended for building executable systems that support the enactment of processes.

In literature, a wide range of process



TABLE 1

Process management tools.

Tools	Process domain	Process life-cycle support	Process modeling language	Features			License and cost (US dollars)
				Other supported technologies	Development support	Usefulness for engineers	
Eclipse Process Composer (EPFC)	Software	Modeling	UMA (SPEM-based)	n/a	Knowledge, sharing, communication	Publish processes as website; knowledge base	Free
OpenUP	Software	Modeling (restricted)	UMA (SPEM-based)	n/a	Guidance for RUP	n/a	Free
Rational Method Composer (RMC)	Software	Modeling	UMA (SPEM-based)	n/a	Knowledge, sharing, communication	Publish processes as website	\$1,020
Appian Enterprise BPM Suite	Business	Modeling, collaboration, automation	BPMN	n/a	Knowledge, sharing, communication	Enterprise integration of process and content management; collaborative team workspaces and tools	\$35 per month and user
ARIS	Business	Modeling, collaboration, automation	BPMN	EPC, UML, BPEL, and WSDL	Knowledge, sharing	Web-based design; decentralized Web-based management of IT resources	ARIS Express Modeling (free); ARIS Business Designer (from \$3,433)
BizAgi	Business	Modeling, collaboration, automation	BPMN	BPEL, SOA, XPath, Tempo, WS-Human Task Service, J2EE, .NET	Knowledge, sharing, communication	Process publication in website; concurrent process working; interaction with other systems via SOA and SharePoint WebParts	\$157 (single user); \$185 (enterprise)
Intalio	Business	Modeling, collaboration, automation	BPMN	BPEL, J2EE, SOA	Knowledge, sharing, communication	Process publication; collaboration facilities for process management; Intalio Social Portal cloud computing platform	Free (community edition; 80% open source code); enterprise from \$9,500 per year

modeling languages exists. Two main initiatives support process modeling, both under the direction of the Object Management Group (OMG): the Software and Systems Process Engineering Metamodel (SPEM),⁴ which supports software process modeling, and Business Process Modeling Notation,⁵ which supports the modeling of business processes. By using adequate tool support, engineers can execute the process, orchestrate with product life-cycle management tools, and thus facilitate automation and collaboration.

Process Management

Processes on paper tend to be “shelf-

ware,” meaning they aren’t maintained or integrated in teams’ daily activities. Modeling should primarily be a means to an end—namely, to support engineers. Table 1 highlights some support tools and summarizes a few of their features.

Eclipse Process Framework Composer (EPF Composer) is based on the Eclipse development platform. You can download the latest version (for free) from the Eclipse page. This tool, which is part of the Eclipse Process Framework (EPF; www.eclipse.org/epf), uses an evolution of the SPEM 1.1 OMG specification referred to as the Unified Method Architecture (UMA), whose

major parts went into SPEM 2.0. The UMA and SPEM schema support the organization of large amounts of descriptions for development methods and processes. Those content and processes don’t have to be limited to software engineering but can also cover other design and engineering disciplines, such as mechanical engineering, sales cycles, and so on. Moreover, EPF Composer supports different project life-cycle methods such as waterfall, incremental, and iterative development. Because of these features, EPF has become one of the most representative software process modelers (see the sidebar).

The *Open Unified Process* (OpenUP;

<http://epf.eclipse.org/wikis/openup/index.htm>) is open source software within EPF developed to make it easy to adopt the core of both RUP and the Unified Process.⁶ The framework is a basic process suitable for small teams (three to six people) interested in agile and incremental development, so adapting to different processes is difficult with OpenUP. Moreover, it doesn't provide guidance on many topics that projects might deal with, such as large team sizes, compliance, safety- or mission-critical applications, technology-specific guidance, and so on. However, OpenUP is complete in the sense that it can be manifested as an entire process to build a system. In addition, it serves as a base process upon which additional process content can be built, for instance, by using EPF Composer.

Rational Method Composer (RMC; <http://www-01.ibm.com/software/awd-tools/rmc/>) is an IBM commercial version of the EPF Composer tool. The biggest difference between RMC and EPF is that RMC is shipped with RUP, whereas EPF Composer comes with OpenUP/Basic.¹ RUP is more extensive than OpenUP/Basic, which is mainly focused on small project teams. On the other hand, RUP is suitable for small, medium, and large projects. RMC offers more than 100 process guidance tools for solving frequent business, management, or development problems.

The *Appian Enterprise* (www.appian.com/bpm-software.jsp) business process management suite includes a wide set of components necessary to model and manage business processes. It uses BPMN and service-oriented architecture (SOA) methodologies for rapid development and automation along with the Appian analytics component to help managers identify and compensate for process inefficiencies. This tool is a good option for distributed development projects. Moreover, the suite couples process management with asynchronous and synchronous



SOFTWARE PROCESS MODELING WITH EPF COMPOSER

The first important aspect to consider when modeling a process in EPF Composer is that it separates the method content from the processes. We recommend defining first the method content elements (work products, roles, tasks, and guidance). You can then use them to define specific processes. To achieve this, in the library view, you must define a new method plugin and create a content package. For each package content, EPF displays the method element types from which it is possible to define the specific method elements and their properties (name, description, relations with other elements, and so on). You can classify content elements via standard categories (disciplines, domains, work product kinds, role sets, and tools) or custom categories that can contain any kind of method element (see Figure A).

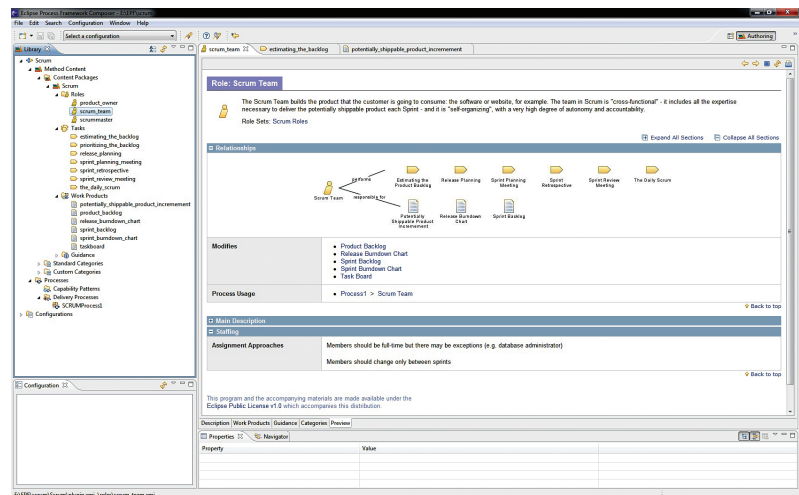


FIGURE A. SCRUM process description with EPF Composer.


The next step is the process definition, which you can do from scratch or by combining method content elements. EPF supports the definition of capability patterns and delivery processes. A capability pattern describes a reusable cluster of doing work in common process areas and can be used to assemble delivery processes or larger capability patterns. A delivery process describes a complete project life cycle, which you can use as a template for planning and running a project. It's defined by populating the

- work breakdown structure, the hierarchical decomposition of the work to do (process patterns, phases, iterations, milestones, tasks);
- work product breakdown structure (work product usage view); and
- team breakdown structure (team allocation view, which is focused on role representation).

EPF provides the facilities required to easily model a software process by promoting the reusability of method elements. Moreover, the framework facilitates process model communication via the publication of method configurations (selection of contents) in HTML and its deployment to Web servers. These facts make EPF today's most popular software process modeler.

INTERNATIONAL CONFERENCE ON GLOBAL SOFTWARE ENGINEERING

Today's IT and software industries demand effective and efficient collaboration of distributed teams. The annual IEEE International Conference on Global Software Engineering (ICGSE) looks to trends and technologies for global development. Previous conferences attracted a broad audience from industry and academia. ICGSE 2011 will be held on 15-18 August 2011 in Helsinki, Finland. For more information and to participate, see the conference website at www.icgse.org.

support relevant standards to manage business processes and that they're focused mainly on application integration and development. Other key features that need more attention are the "human-centric" aspect, which requires communication and interaction facilities, and more tools for adopting the cloud computing paradigm to facilitate distributed work. 

References

collaboration tools that encourage teams to communicate throughout the life of a business process.

ARIS Platform (www.ids-scheer.com/en/ARIS_ARIS_Platform/3730.html) provides integrated software products that help enterprises manage their business processes. These products cover every phase of a business process management project, from strategy definition and process design to transferring the models into IT systems and monitoring process execution. ARIS also lets users identify organizational, structural, and technical problems in workflows and unlock improvement potential. Moreover, process models can be published worldwide for role-based access.

BizAgi (www.bizagi.com) supports the entire business process life cycle via different components that allow—through its graphic and dynamic environment—the building of a process-based solution. The result is the automatic generation of a Web application that is based on and activated by a process diagram without requiring any programming. The BizAgi Process Modeler supports the modeling phase and is a free component that allows business process definitions. For automation, the BizAgi suite includes the Studio component, which interprets the BPMN model and generates a Web application that's synchronized with the model. This server is composed of a collection of components to support business management such as a work portal, business activity monitoring,

business rules, and an integration engine, among others.

Intalio (www.intalio.com) supports any size of business process. The Intalio suite covers the entire life cycle of business processes, supporting the BPMN 2.0 specification and earlier versions, so that the process engine can directly execute BPMN 2 models without having to resort to any code translation. You create business models via the Intalio Designer, which is built on top of the Eclipse platform as a collection of plug-ins. Processes are deployed in the suite's server, which is a native BPEL 2.0 process server based on the J2EE architecture and the Apache ODE open source BPEL engine to support distributed transactions and human workflow. Cloud computing is supported by providing 100 percent Web-based user interfaces, native multi-tenancy, a small memory footprint optimized for virtualization, and support for the most popular cloud deployment options.

In business processes, modeling and automation are closely connected, but software development still separates process modeling from orchestration, tool federation, and thus automation. Different silos exist for the three domains of modeling, collaboration, and automation. Only a few suppliers have started to bridge them, such as IBM, but at the cost of a proprietary framework that can lock in users easily.

From the tool suites we analyzed here, it's interesting to note that they

1. S. Larsen, "Evaluation of Software Modeling Tools for Agile and Industrial Process," Dept. Computer and Information Science, Norwegian Univ. Science and Technology, 2006; www.idi.ntnu.no/grupper/su/fordypningsprosjekt-2006/larsen-fordyp06.pdf.
2. J. Portillo Rodríguez, C. Ebert, and A. Vizcaíno, "Tools and Technologies for Distributed Teams," *IEEE Software*, vol. 27, no. 4, 2010, pp. 10–14.
3. C. Ebert, *Global Software Engineering: Distributed Development, Outsourcing, and Supplier Management*, Wiley/IEEE CS Press, to be published in 2011.
4. "Software & Systems Process Engineering Meta-Model Specification (SPEM), Version 2.0," Object Management Group, 2008; www.omg.org/spec/SPEM/2.0/PDF.
5. "Business Process Model and Notation (BPMN), Version 2.0, Beta 2," Object Management Group, 2010; www.omg.org/cgi-bin/doc?dtc/10-06-04.pdf.
6. V. Nikulsins and O. Nikiforova, "Tool Integration to Support SPEM Transformations in Eclipse," *Scientific J. Riga Tech. Univ.*, vol. 45, 2010, pp. 60–67; <https://ortus.rtu.lv/science/en/publications/8640/fulltext>.

FÉLIX GARCÍA is an associate professor at the University of Castilla-La Mancha (UCLM). Contact him at Felix.Garcia@uclm.es.

AURORA VIZCAÍNO is an associate professor in the Alarcos Research Group at the University of Castilla-La Mancha. Contact her at aurora.vizcaino@uclm.es.

CHRISTOF EBERT is managing director at Vector Consulting Services. He is a senior member of IEEE and the editor of this department for *IEEE Software* magazine. Contact him at christof.ebert@vector.com.



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