Workflow Support for Inter-organizational Design Processes

Ri Hai[a], Markus Heller[b], Wolfgang Marquardt[a], Manfred Nagl[b], René Wörzberger[b]

[a]*Process Systems Engineering, RWTH Aachen University, 52056 Aachen, Germany*
[b]*Computer Science 3 (Software Engineering), RWTH Aachen University, 52074 Aachen, Germany*

Abstract

Inter-organizational design processes are inefficiently supported by management tools w.r.t. to their dynamic nature and inherent complexity. In this paper, we present an innovative framework for the definition, analysis, improvement, and management of such design processes. Key features are to bridge the gap between modeling and execution of inter-organizational design processes and the seamless execution support for both dynamic and static parts of the overall process both by appropriate process management systems. A case study is given for the conceptual design process of a plant for polyamide 6 production by hydrolytic polymerization.

Keywords: process design, workflow management, computer aided engineering, inter-organizational design processes

1. Introduction

Design processes in chemical engineering are often carried out by *engineering teams* distributed across different organizations. Each team of *engineers* works on a certain part of the overall design process and is lead by a *process manager* who plans and controls this process part. Management support of inter-organizational design processes for process managers and engineers should not only address the management of the process parts inside one organization but also the coordination of the distributed process parts across organizational boundaries.

A design process part can be identified to be *dynamic* or *static*, depending on its scope and character. When desiging complex, unpredictable processes, many alternative and successive technical products are generated and evaluated. Such a design process (part) is not predictable in every detail and is referred to as “dynamic”, e.g., the overall design process is dynamic and may contain dynamic design process parts. In contrast, some design process parts, e.g. the design of an apparatus, and can be specified in advance on a fine-grained level. Such design processes are called “static”. Obviously, different types of design processes lead to different requirements of management. To coordinate the distributed *design process parts*, design tasks for each organization must be clearly defined to avoid misunderstanding. Especially, the documents to be exchanged and the context of each design step, e.g. its dependence of other design steps must be considered. It is also important to note that one organization may want to protect its sensitive information.

1.1. Requirements for Tool Support of Inter-organizational Design Processes

Management tools must be able to deal with both static and dynamic design processes [1]. For dynamic design process parts, the tools must support process

managers and allow for interleaved planning and execution of the process parts overseen by them. For static process parts, the automatic execution by management tools must be supported. Within each organization, it should be possible to choose the management tools used for executing either dynamic or static process parts individually
Additionally, the coordination of all design process parts across organizations requires adequate tool support. Firstly, the tools must provide a *unified understanding* of the entire design process as well as of the process parts for all involved organizations before the design process begins. Secondly, corresponding management tools must be *coupled* with each other to coordinate the work of all involved engineering teams. Not only the exchange of process status information and resulting documents among the organizations should be supported, but also *protection of the sensitive information* should be handled appropriately.

1.2. State of the Art

Only a few existing systems such as the n-dim system [2] and KBDS [3] address dynamic design processes in chemical engineering directly. Conventional workflow management systems (WFMS) [4] can only be used to execute static (parts of) processes.
Advanced or dynamic WFMS support interleaved planning and process execution. An example of such a system is AHEAD [5]. It has been developed in the context of the long-term research project IMPROVE [6] which is concerned with models and tools for design processes in chemical engineering. AHEAD offers a management environment to process managers to plan, control and monitor dynamic processes as well as a work environment for engineers to access and execute assigned design process activities. In the medium time range, such advanced systems are considered to be of high industrial relevance. A delegation-based approach for the management of dynamic inter-organizational design processes is described in [7], where a client and a contractor organization couple their AHEAD-instances for executing the overall design process. However, we assume that the overall process only includes dynamic processes parts.

2. A New Framework for Workflow Support for Design Processes

Based on the above requirements, we now present a framework developed within the IMPROVE project that offers a suite of modeling languages, methodologies and tools to support inter-organizational process design with both static and dynamic process parts. The framework spans the entire lifecycle of a design process from its early definition to its execution within conventional and advanced dynamic WFMS (Fig.1). Four different phases are identified: (1) process definition and distribution, (2) process formalization, (3) process execution, (4) and process evaluation and improvement. In the following, the phases are described.

2.1 Process Definition and Distribution

In this phase, the focus is put on the understanding of the entire inter-organizational design process. To clarify which organizations have to perform which process steps and deliver what results to whom, an initial draft of the overall process is created, including the planned *public* design processes of all organizations and their dependencies. This overall process is *split up* into process parts for each organization. A process part contains all the public activities to be carried out within an organization and it may depend on other process parts. Therefore, a *process definition* of a design process part should include both the public activities within one organization and the dependencies to other organizations' activities. These process definitions are passed to the corresponding organizations and can be extended inside one organization with *private* process details (not shown in Fig. 1).

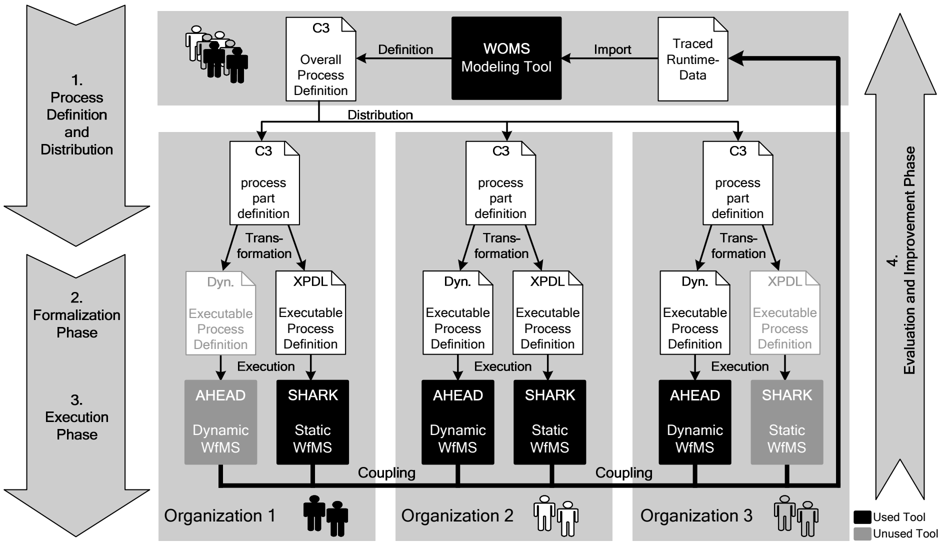

Figure 1. New framework for workflow support of inter-organizational design processes.

In the definition phase, a semi-formal process modeling language is sufficient, since it allows process modelers and analysts of the participating organizations to create clear and simple process models and to gain a common understanding of the overall process. For this purpose, the semi-formal language C3 [8] was developed within IMPROVE and the workflow modeling tool WOMS [9] was developed based on C3 language to define, analyze, and improve design processes on process definition level. When a design process is defined in WOMS, it is possible to analyze the process with different aspects, such as which design tasks are to be undertaken by a certain role, or which document is the input and output for a certain design step.

2.2 Process Formalization

The focus of this phase is on process execution within management tools of each organization. In the process definition phase and the process execution phase, instances of the same design process are on different abstraction levels. Accordingly, the modeling languages used in each phase differ in the modeling capabilities. A semantic *gap* exists between the two types of modeling languages. For example, a semi-formal modeling language targeting process understanding purposes (cf. section 2.1) contains no information regarding the execution state of a process, but an execution-oriented modeling language addresses such information. Further, while the data type definition is an important part of the later modeling language, it is not necessary for the former modeling language. During semi-automatic *transformation* between both modeling languages the missing content can be added to obtain an executable process definition from the semiformal process definition. The transformation also realizes a mapping from a platform-independent notation to a platform-dependent notation. Therefore, the transformation depends on the type of WFMS, which is used in the process parts for execution.

Within our framework, process modeling experts and process managers of each organization can choose different platforms for process execution, i.e. conventional WFMS to automate static process parts, advanced WFMS for dynamic process parts, or both systems in parallel. We use the aforementioned advanced WFMS AHEAD for the management of dynamic processes which models design processes as dynamic task nets with the DYNAMITE modeling language [5]. We use a conventional WFMS, SHARK from ObjectWeb Consortium, because it is able to process workflow definitions in the standardized workflow definition interchange format XPDL [10], so that other WFMS using this standard format can be used without further adaptation of the framework architecture. If there are significant interactions among processes parts, an AHEAD-AHEAD coupling should be used, because specifical modeling elements are introduced in AHEAD for the dynamic design processes, e.g., feedback relationships for iterations [7]. To deal with both WFMS, two different transformers have been realized via simple XSL transformations in our framework: one transformer for transforming the C3-nets into the dynamic task nets for execution within AHEAD, and the second for transforming the C3-nets into the XPDL-nets for execution within the SHARK.

2.3 Execution

In the *execution phase*, either the conventional or the advanced WFMS of the organizations are *coupled* with the WFMS of other organizations. The coupling is bi-directional in each case to avoid the need for a centralized coordination system among all WFMS. Several instances of AHEAD can be coupled for delegation-based cooperation relationships [7], and the coupling of a conventional WFMS (SHARK) with an advanced WFMS (AHEAD) has been recently realized (cf. section 3). Tool integration of AHEAD and SHARK is done in an a posteriori manner, so that no significant modifications of the publicly available source code of SHARK were required. We only need to extended the SHARK system with an event-based coupling component to exchange workflow-related events between the coupled workflow systems.

2.4 Evaluation and Improvement

The process evaluation phase focuses on the re-use of previously executed instances of design processes. Comparison and analysis of the to-be process and the as-is process indicate the deficiency in the process definition. Additionally, the accumulated process instance data can be used to determine the beneficial process details for the optimized and more efficient design processes in the future. In this framework, all the information gained can be recorded with WOMS manually to improve the design process step by step. As described in [11], it is also possible to capture the design process history into process traces automatically as a basis for design process improvement.

Summary

The proposed framework meets all requirements of section 1.1 by providing adequate modeling languages and tool support for process modelers, process managers and the engineers within all phases of the design process lifecycle. Process managers can use different management tools specialized on the execution of dynamic or static processes to support process management. Within the framework architecture, the management tools can be coupled to form a *federation* of process management tools and exchange process information and data with each other.

3. Case Study

In the following, the concepts and tools described so far are implemented through a case study on the design process of a plant for polyamide 6 (PA 6) productions, including the design of the reaction and separation system as well as the extruder configuration. More information about the PA6 design process can be found in [6]. Figure 2 shows a simplified overview on the inter-organizational PA 6 design process. In this scenario, a chemical engineering company and an extruder manufacturer are involved. The roles manager, reaction expert, and separation expert belong to the chemical engineering company and the plastic engineering belongs to the extruder manufacturer. We assume

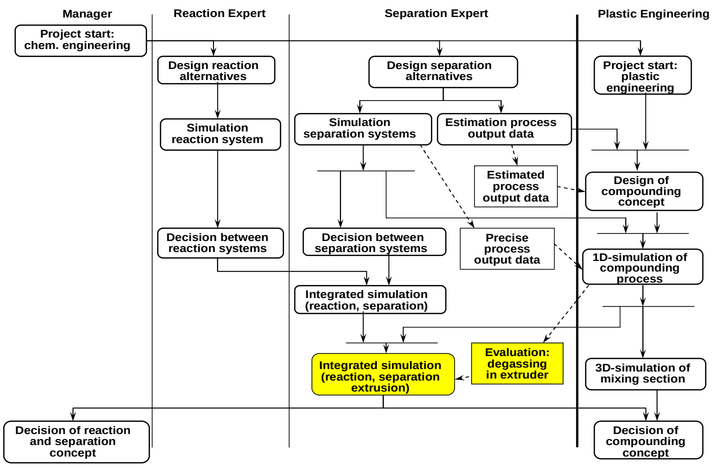

Figure 2. Design process for Polyamide 6 production.

that the chemical engineering company is responsible for the entire process and the extruder manufacturer is a contractor for the design of the polymer processing section. This corresponds to the classical disjunction of the two domains of chemical and plastics engineering. In order to illustrate the cooperation between the two companies without blurring the overview, only the information, which is delivered by one company to another, is shown in Fig. 2.

Now we briefly illustrate the application of the proposed framework by means of this scenario (Fig. 3). Firstly, WOMS is used for the modeling of the entire design process (e.g., by an expert in the chemical engineering company). An initial draft of the design process is created (cf. Fig. 2, without the highlighted blocks). This initial draft is improved by inserting an additional design step (cf. Fig. 2, the highlighted blocks) to account for the fact that the separation system can be reduced considerably in size, when part of the degassing is performed in the extruder. After that, individual design process parts are extracted from the overall process definition and distributed to both companies: the chemical engineering company as the client receives the overall process definition. The plastics engineering company as the contractor receives his polymer part and additionally all inter-organizational control and data flow, which depend on activities of the chemical engineering company.

A similar scenario is given in [7], where coupled instances of AHEAD are used in both organizations. In this paper, we focus on the mixed case: the client uses AHEAD to manage and execute the overall dynamic design process while the plastics engineering company can use SHARK. The process fragment for the plastics engineering company is small, static and well-understood, e.g. it may be repetitively carried out and thus offered as a service to customers like the chemical engineering company. In the formalization phase, the client uses the C3-AHEAD-transformator to create a dynamic task net and imports it into AHEAD. The contractor uses the C3-XPDL-transformator and installs the resulting XPDL workflow description in the SHARK system. The client starts the process within AHEAD and calls for the execution of an instance of the installed workflow within SHARK after providing needed input parameters. Both systems are coupled with a CORBA-based communication server (providing message passing and message queuing) for the exchange of process state information and data transmission.

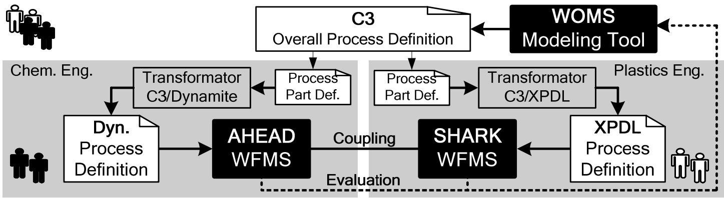

Figure 3. Workflow support for polyamide 6 design process.

4. Conclusion

In this paper, we have presented a framework of models, methodologies and tools from the IMPROVE project to support inter-organizational design processes. It supports process modelers, process managers and the engineers within all phases of the design process lifecycle. Dynamic and static process parts are executed by coupled advanced or conventional workflow management systems. The application of the framework was demonstrated through a case study on the conceptual design process of a plant for polyamide 6 production. In the future, we aim at extending the existing transformers between modeling languages of different phases to address advanced problems, e.g. hierarchical design process structures, and we will work on improved user interfaces.

Acknowledgement

Financial support of Deutsche Forschungsgemeinschaft (DFG) within the Collaborative Research Center 476 IMPROVE is gratefully acknowledged.

Reference

[1] B. Westfechtel. Models and Tools for Managing Development Processes, LNCS 1646, Springer, Germany, 1999.
[2] A. W. Westerberg, E. Subrahmanian, Y. Reich, S. Konda, and the n-dim group. Designing the process design process. Comp. & Chem. Eng., 21(Suppl): S1–S19, 1997.
[3] R. Banares-Alcantara and H. Labadibi. Design support systems for process engineering II KBDS: an experimental prototype. Comp.& Chem. Eng., 19(3):279–301, 1995.
[4] P. Lawrence, editor. Workflow Handbook. John Wiley, Chichester, UK, 1997.
[5] D. Jäger, A. Schleicher, and B. Westfechtel. AHEAD: Graph-Based System for Modeling and Managing Development Processes. In M. Nagl, A. Schürr, and M. Münch, editors, AGTIVE – Applications of Graph Transformations with Industrial Relevance, LNCS 1779, pp. 325-340, Castle Rolduc, The Netherlands, 1999. Springer.
[6] W. Marquardt and M. Nagl. Workflow and information centered support of design processes — the IMPROVE perspective. Comp. & Chem. Eng., 29(1):65-82, 2004.
[7] M. Heller, D. Jäger, M. Schlüter, R. Schneider, and B. Westfechtel. A management system for dynamic and interorganizational design processes in chemical engineering. Comp. & Chem. Eng., 29:93–111, 2004.
[8] C. Foltz, S. Killich, M. Wolf, L. Schmidt, and H. Luczak. Task and information modeling for cooperative work. In Smith and Salvendy. Proc. HCI Intl., vol. 2, pp. 172–176, 2001.
[9] R. Schneider and S. Gerhards. WOMS – A Work Process Modeling Tool. In M. Nagl and B. Westfechtel, editors, Modelle, Werkzeuge und Infrastrukturen zur Unterstützung von Entwicklungsprozessen, pp. 375–376, 2003. Wiley VCH.
[10] Workflow Management Coalition. Workflow Process Definition Interface – XML Process Definition Language. WFMC-TC-1025 Version 1.0, 25 October 1999.
[11] M. Jarke, T. List, and K. Weidenhaupt. A Process-Integrated Conceptual Design Environment for Chemical Engineering, Proc. 18th Intl. Conf. on Conceptual Modeling (ER '99), pp. 520–537, 1999.