**Association for Information Systems**
## AIS Electronic Library (AISeL)

Wirtschaftsinformatik Proceedings 2007 — Wirtschaftsinformatik

February 2007

# Conceiving an environment for managing the lifecycle of collaborative business processes

Dirk Werth

*Institut für Wirtschaftsinformatik im Deutschen Forschungszentrum für Künstliche Intelligenz,* Dirk.Werth@iwi.dfki.de

Philipp Walter

*Institut für Wirtschaftsinformatik im Deutschen Forschungszentrum für Künstliche Intelligenz,* philipp.walter@iwi.dfki.de

Peter Loos

*Institut für Wirtschaftsinformatik im Deutschen Forschungszentrum für Künstliche Intelligenz,* loos@iwi.uni-sb.de

Follow this and additional works at: http://aisel.aisnet.org/wi2007

# Conceiving an environment for managing the lifecycle of collaborative business processes

Dirk Werth, Philipp Walter, Peter Loos

Institut für Wirtschaftsinformatik
im Deutschen Forschungszentrum für Künstliche Intelligenz
66123 Saarbrücken
{dirk.werth,philipp.walter,peter.loos}@iwi.dfki.de

**Abstract**

Business process lifecycle management is established for the continuous improvement of business processes within a single enterprise. However, the concept could also be applied to enhance collaborative business processes spanning over multiple enterprises. In contrast to the intra-organizational case, lifecycle management of cross-organizational collaborative processes imposes several organizational and technological challenges that results from the multiple-independent-actors-environment of collaborations. This paper focuses on these challenges and presents a conceptual solution for the different phases of this lifecycle. This concept has already been integrated into a prototypical software environment, that supports networked enterprises in the lifecycle management of collaborative business processes and that is presented in this paper.

## 1   Introduction

Classical industrial enterprises incorporated a high amount of the value generation within themselves. Due to the increasing complexity in management and manufacturing, enterprises nowadays tend to transform into smaller value units which are strongly specialized in their core competence [PrHa90]. Thus the added value is successively generated in networked structures [ÖsFl00], i.e. the companies intensively interact along the added value chain in order to produce the intended output conjointly. This intensification of exchanges leads to strong collaborative relationships (also called collaborative business, cf. [RöSc01]). So the ground is prepared for enterprise networks and virtual organizations [DaMa92]. Such collaborations are mainly driven

by the intention to generate added value, which is achieved through synchronized execution of associated business tasks. This activity sequence constitutes a collaborative business process. Collaborative business processes [Wert06] are a special kind of conventional (intra-organizational) business processes. Like the conventional ones, they consist of a sequence of business activities. However, they possess special properties that strongly differ from the regular case. First, they are spanning over multiple organizations, because generation of added value is performed through cross-organizational division of labour. Second, each of the individual business activities that compose the process clearly belongs to a unique organization. Thus the collaborative business process can be partitioned into several parts, each of which contains one or more activities distinctly associated with an organization that fully controls this part in the sense that it independently executes, administrates and manages it. In this way those parts of cross-organizational business processes can be characterized as autonomous fragments.

Therefore collaborative business processes strongly differ from intra-organizational ones. Consequently concepts and solutions that are developed for the intra-organizational case are in most cases not suitable for cross-organizational purposes. This article investigates the aptitude of the business process lifecycle concept for such collaborative environments. After showing the gaps within the 'classic' lifecycle concept, we propose a platform which is apt to support the lifecycle for cross-organizational business processes. In the following sections the conceptual and technical basics of this platform are presented. In contrast to other approaches, e.g. [GrAb01], we do not focus bilateral processing of business processes only, but complete end-to-end processes. Therefore we will step through the three phases of the cross-organizational Business Process Lifecycle and show the concepts we developed for every phase. Afterwards we will show how the concept is realized so far and finish with a short outlook

## 2    The Business Process Lifecycle in Collaborations

For a continuous and successful business strategy it is insufficient to cover the design of business processes only, since the design solely results in static models of the considered processes which do not allow for process changes. However, execution of these static models usually yields improvement potential over time, e.g., because the execution context changed or certain execution aspects were not reflected in the model. To realize and quantify these improvement

potentials, it is necessary to measure execution of the models, i.e., perform controlling of them, which allows for identifying weaknesses and changing the models accordingly.

These three steps are integrated in the Business Process Lifecycle shown in Figure 1: business process design, business process implementation and business process controlling [ScJos02]. The basic lifecycle concept can be found in the House of Business Engineering [ScNü95;Sche96]. Business process design refers to modelling of existing as-is or intended to-be processes. This can be accomplished using modelling languages (e.g., EPC [KeNü92], BPML [Arki02]) and the respective modelling tools. Business process implementation summarizes all operative steps that are necessary to execute a process which was modelled before, including IT systems for execution as well as human interaction. Among the technical means for process execution are for example ERP systems and workflow engines. Research effort is currently put into the exploration of mechanisms to minimize the need for human interaction in business process implementation. Business process controlling denotes all actions that aim towards measurement and examination of running and finished processes with the goal of discovering optimization potentials. Once found, such a potential can be realized by changing the process model in the modelling phase of the next cycle pass.
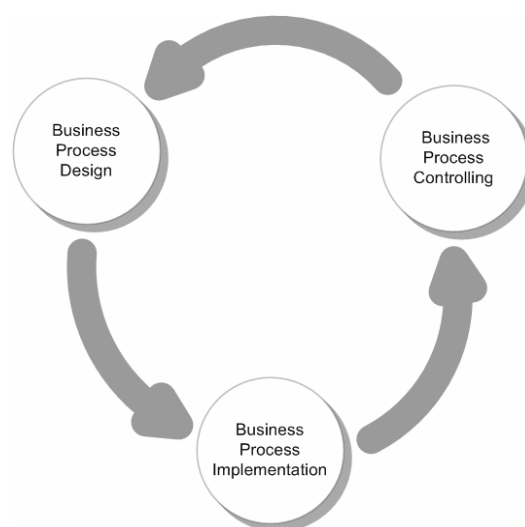


Fig. 1: Business Process Lifecycle

This lifecycle is conceived for a single organization. In the design phase, each process model is changed by a single modeller at a time. During execution, the process is handled by a single execution system within a single organization. Consequentially all controlling information can be gathered "indoor", i.e., within the organization. However, in environments with multiple organizations acting cooperatively, collaborative processes cannot be regarded as monolithic

anymore, since the different parts of them are designed, executed and controlled by different organizations [LuBu99]. Consequently the lifecycle gets very complex and difficult to handle:

The *design* (respectively *modelling*) task comprises multiple autonomous modellers that act independently and follow different goals. This results in self-contained parts of the collaborative business process. Therefore the process design can rather be characterized as an assembly task of autonomous process parts.

The *execution* is distributed over different enterprises. Consequently there is no central processing engine. Instead each autonomous process part has its own independent processing engine, so classic workflow concepts and technologies have to be extended to match the new cross-organizational requirements [Schu02].

*Controlling* means monitoring of running and finished processes and comparing them with set values. However, monitoring in the sense of determining unique process states is impossible for collaborative workflows, because their state is hidden in the autonomous workflow engines. They only disclose virtual state information that clouds the real procedures. Moreover, the controlling comprises the aggregation and calculation of valuation functions. However, these functions contain information on business structures (esp. cost factors). Such information is considered business-critical and inaccessible to third parties, even if they are partners.

Having revealed these gaps, we will step through these three phases and show the concepts for collaborative business processes in the next section.

## 3 Conceiving a Cross-Organizational Business Process Lifecycle

Transferring the concept of lifecycle-based business process management to cross-organizational environments requires a shift from a centralized paradigm to a support for distributed environments, because cross-organizational business processes are characterized by the involvement of multiple actors in the different life-phases. For these actors a collective behaviour cannot be supposed. Thus each phase requires new techniques that are different to those of the classical business process management and that incorporate the split activities. Therefore we do not focus on bilateral processing of business processes, but on end-to-end processes with potentially a huge number of contributors.

## 3.1 Distributed Business Process Modelling

The design of business processes is considered a fundamental management task. In order to document the design, a specification medium is needed. On the conceptual level models have raised as the primary medium for business process specifications (e.g. EPC, BPML, BPEL, etc.). Thus the design task can be summarized as the creation of business process models. With regard to cross-organizational business processes, this actually comprises the model generation for an original that spans over multiple organizations. In principle this can be performed in a centralized and a decentralized way:

Supposing a centralized model creation, a single actor (that may also be incorporated by a group of collectively acting individuals) is responsible for the whole process model. This implies detailed knowledge of and unrestricted access to all aspects of the process. Due to the individual demand of secrecy, real-world organizations usually do not agree to fully expose their knowledge and processes to a third party. So this case can be considered implausible.

Assuming a decentralized model creation, this implies the existence of different modelling individuals, each of which generates only parts of the process. Within this procedure they may follow different modelling paradigms, methods and languages. Therefore this approach requires both a technique for assuring the consistent individual model creation and a technique for the integration of the partial models.

Another dimension is the direction of the model creation procedure. Here we distinguish between creating a model by more and more detailing an abstract description of the model object, and building a model by adding more and more aspects to it and aggregate it afterwards. So we can distinguish a top-down and a bottom-up approach in the model creation procedure:

Many approaches follow a *top-down modus operandi* for modelling cross-organizational business processes [AdCh05]. The foundation for this procedure is a blueprint model (a.k.a. reference model) of the cross-organizational business process which is to be implemented by the different participants. In a second step, each of them must adapt his process parts and refine them according to the blueprint. However, if we postulate independent organizations, i.e. they are legally independent and acting on their own behalf exclusively, this forced adaptation mechanism contradicts with the autonomy property. The presupposition of independent organizations fits most real-world collaboration scenarios, so the modelling procedure must not interfere with the autonomy of the individual organizations.

A *bottom-up approach* follows the reverse direction. It founds on as-is models that reflect the status quo of the participants. These detail models are reduced by removing information that is confidential and not adequate for third party usage. These "alienated" models are composed in a second step and integrated into an overall model. The main advantage of this approach is the elimination of the need to adapt. It satisfies the autonomy postulation, because no participant is obliged to adapt their models and consequently no modifications in the processes have to be performed. In this context, the detail model that constitutes the base of this procedure reflects existing process capabilities, i.e. the ability of a participant to perform a business process. These capabilities are encapsulated in modules from which a new collaborative business process is composed. In order to include flexibility mechanisms in the assembly procedure, the individual actor can incorporate its readiness to adaptation within its module design.

Because these two dimensions in model creation are independent of each other, they can be aligned orthogonally in a 2x2 solution matrix, as shown in Table 1. Reflecting the different characteristics of the two dimensions, neither a centralized modelling nor a top-down approach seams to be appropriate for the special organizational environment of collaborations. In our concept we therefore follow a bottom-up approach using a decentralized modelling procedure (Option 4).

|  | **Centralized Modelling** Single actor in charge of whole process model (overall knowledge and unrestricted access) | **Decentralized Modelling** Multiple modellers generate partial models (for their specific area) |
|---|---|---|
| **Top-Down** From a master plan to the operational realization | 1 | 2 |
| **Bottom-Up** From the capabilities to the overall inter-working plan | 3 | 4 |

Tab. 1: Procedure options for model creation in multi-actor-environments

Although from a theoretical perspective, such an approach has to cope with all potential permutations of modelling techniques, our approach is a homogeneous one based on a single modelling language. Even in this scenario there are sufficient degrees of freedom for the modelling

subject. In our implementation the event-driven process chain (EPC) language is used. Since EPC is one of the most common process model languages (at least in Europe), this seems to be a suitable assumption. More precisely, our design procedure comprises four steps:

1. *Definition of process modules:* In contrast to the top-down approach described above, we start with the assessment of the status quo of the different organizations involved by specifying their capabilities. In our case they have to express their ability to produce output using process models that describe their possible processing sequences. The results are component-like models that can be assembled together and that incorporate process interface descriptions specifying interaction points.

2. *Definition of process intentions:* The composition of process modules has to follow certain business objectives. In order to construct an objective-adequate process model, the intention of the underlying process must be defined. In particular this addresses the output the process has to deliver as well as the organizational constraints (e.g., the whole process has to be performed within the EU).

3. *Process module composition:* The composition itself is performed by analyzing compatibility of process interface pairs. That yields pairs of matching interfaces through which process modules can be connected. Based on those modules which are able to produce the intended outcome, a network of modules is successively constructed. This finally results in a set of modules that generate the final product. Thus the composition is directed by the matching assignments of the process interfaces. The set is filtered by the organizational constraints of step 2 and rated by a common target function. The best rated result is the final one and describes a common cross-organizational business process model for all participants.

4. *Process model consistency analysis:* To avoid contradictions within the overall process model, the composition phase closes with a consistency analysis during which the model is analyzed with respect to flow logic consistency. Such a test is described for example in [SaOr99]. Having passed this test, the cross-organizational business process model can be realized within all involved organizations.

### 3.2 Distributed Business Process Execution

The distributed execution of a business process starts with a common process model that all participants share and that is business oriented, i.e., its content is mainly conceptual and its purpose is organizational management. From this model every participant extracts those parts that he has to execute and augments them with arbitrary information he needs for execution, e.g., refinements of process sub-parts or execution context parameters (cf. Figure 2). Thus the business model is transformed into an IT-oriented workflow model, the main purpose of which is the execution of the contained process. The following section introduces the steps from the common process model to execution of the workflow model:

1. *Splitting Up the Common Process Model:* All activities in the common process model are annotated with the executing organization unit ("Company X"), or with an organization unit role ("Customer") that can be mapped onto a concrete actor within the execution context. So the common model disaggregates in disjoint process model fragments that are executed by exactly one actor each. Because the process modules, which were composed to the common process model during the modelling phase, have interface descriptions, it is possible to define exactly which goods and which information must be transferred from one actor to another.

2. Apart from goods and information, the execution of the whole process devolves from one actor to another at an interface. Therefore it is necessary to define how the control of the process is transferred. At process junctions it may be even possible to split up process control or join multiple execution threads again.

3. *Augmenting the Process Fragments:* Execution of a process fragment usually requires considerable prearrangements on the part of the executing actor. Therefore the process fragment is first transformed from the modelling language into an executable language. Since the business process model is business oriented, it usually does not contain information about execution parameters, e.g., an IP address of an interface or authentication credentials for an ERP system. So it must be augmented with these missing execution parameters during or after transformation to the executable language. After transformation and augmentation, the process fragment is contained in an executable workflow model.

4. Usually the common business process model disaggregates into multiple process fragments, each of which is transformed into a single workflow model. These workflow models are deployed to the respective IT systems then, which are finally configured with the contained information.

5. *Executing the Process:* Figure 2 shows how the whole top-level process is implemented by executing the workflow models of the process fragments which it consists of. After configuration of all involved systems this happens automatically, i.e., without interaction with individual process instances.
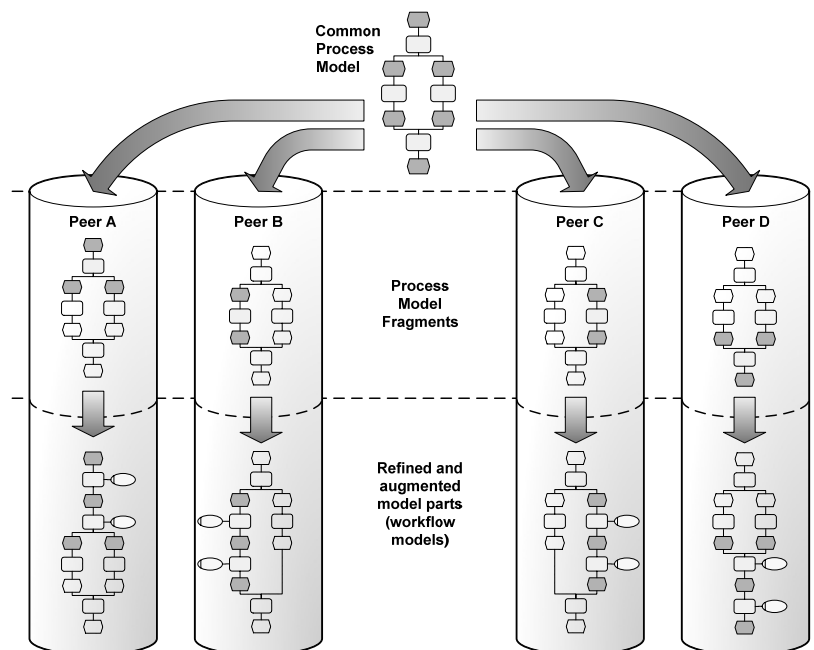


Fig. 2: Distributed business process execution

Since the whole process is executed fragment-wise by multiple separate systems, there must be transition points from one system to another where execution is finished or suspended at the source system and perpetuated at the target system. This transition has two different aspects: data flow and control flow. Data transfer between separate IT systems is widely used already, e.g., between departments within a single organization. However, the transfer of process execution control and context via push and pull mechanisms is not common. Especially in split and join situations, e.g., when a simultaneous execution of multiple process parts on multiple systems begins or finishes, the process context must be duplicated and merged accordingly. During execution, performance data is gathered as a means for the next lifecycle step: the controlling phase.

## 3.3 Distributed Business Process Controlling

From the management perspective, the ability to execute a business process is not sufficient. In order to improve the design and the way of execution it is essential to measure the target object, i.e., to reveal performance indicators of the cross-organizational business process. In the intra-organizational case, this means to extract historical execution information from a single process execution system (mostly a workflow management system) and to calculate the performance indicators from them. In contrast to that, the cross-organizational case is rather complicated. On the one hand there are multiple execution systems, each of which holds only partial information about the execution of a single cross-organizational business process. Thus the challenge is not only to compose performance data from multiple sources, but also to identify linked process chunks and to reconstruct the complete structures of historical cross-organizational business processes under the side condition of heterogeneous keeping of data and system ownership. On the other hand this information on the reconstructed process not necessarily leads to performance indicators for the whole process, because the calculation of these indicators requires the valuation of process execution data. However this valuation (e.g., the cost function) is usually considered a business secret, so an overall indicator processing cannot be performed without exposing individual business knowledge. Therefore we propose to calculate distributed performance indicators in a way equivalent to the execution data processing: each organization transforms the process information gathered from the execution systems into its individual (partial) performance indicators. These figures will then be used to compute the overall indicators. Following this procedure, the organizations are not obliged to publish their calculation scheme and only communicate the resulting values.

## 4  Technical Realization

In this section the realization of the concepts described above will be presented. Within the research project P2E2 (Peer-to-Peer Enterprise Environment[1]), a platform has been developed that prototypically implements the distributed Business Process Lifecycle management principles. The basic idea is to form a network of actors ("peers") which are all equal with respect to rights and what they are able to do [ScFi02]. The network is dynamic, i.e., peers may enter and leave

---

[1] P2E2 is funded under the SE2006 initiative by the German ministry of education and research (BMBF).

the network at any time. The peer-to-peer principle guarantees equal opportunities for all participating parties. Every party distributes models of the processes that it offers to perform. A customer peer can reassemble these process fragments to the model of a complete process and buy the execution of it (or parts of it) from other peers. Thus the P2E2 network structurally corresponds to the organizational network of the collaborating organizations and therefore provides a wide set of advantages as a technological base for enterprise networks [KuWe04].

## 4.1 Distributed Business Process Modelling

First, the processes offered in the network must be modelled, aggregated, assembled and so on. The top-level modelling language used in the P2E2 prototype is the event-driven process chain (EPC). Modelling is performed using the ARIS Toolset by IDS Scheer AG. However, the P2E2 meta-model explicitly supports other modelling languages, too.

In the first step, every peer designs his own processes in any desired detail, thus obtaining a "private" model which can contain arbitrary (even secret) information about the process and therefore is not shared with other peers. Then he generates a "public" view to the model by reducing the contents of the private model to the minimum that is necessary for other peers to comprehend the modelled process and its interfaces.

In the next step, all public models by all actors are distributed among the network. For this purpose we developed the Process Distribution and Discovery Tool (PDDT), a peer-to-peer software which is based on the JXTA peer-to-peer framework and supports distributing, versioning, searching and transferring models (see Figure 3). With the shared information about the available process fragments, any peer can construct a complete process from the fragments. Using the PDDT again, this common process model is shared with all peers that participate in its execution.
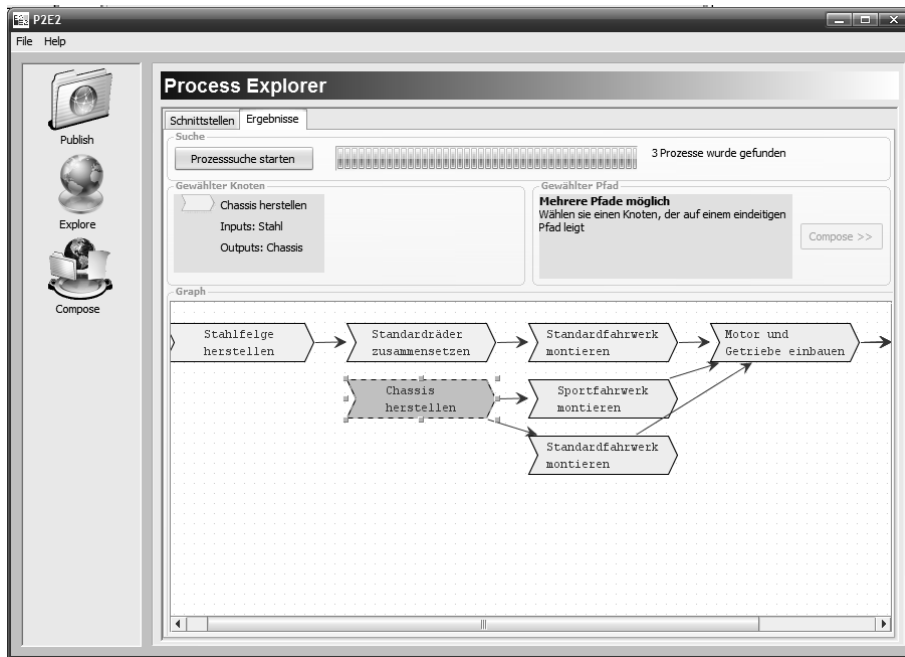
Fig. 3: Screenshot of PDDT

## 4.2    Distributed Business Process Execution

Figure 4 shows the architecture of a P2E2 peer along with the controlling and configuration applications which are not an integral part of the peer itself. This subsection about execution starts with the output of the modelling tool in the lower left corner of the figure.

In P2E2, the execution part of the lifecycle is simplified compared to the scenario outlined in Section 2, because the common process model is composed from several process fragments. So the responsibilities for the execution of the process parts are ex ante established and partitioning the common process can be omitted, because the fragments already exist. The augmentation of the process fragments with execution information also benefits from the fact that the private model with all execution details already exists. So it is sufficient that every peer augments its process fragments once and reuses this information in every execution.

Another part of the augmentation phase is the conversion of all models into a common execution model language, i.e., XPDL in our case: finally, all P2E2 process fragments exist as executable XPDL models. To obtain the final XPDL models, a multi-stage conversion and augmentation is performed. First, the EPC models are automatically converted into XPDL format using the modelling tool. Then the attributes of all XPDL model elements are filled in with data necessary for execution using another tool developed within the project, which is named "augmentation tool" in Figure 4.

Execution in P2E2 is finally performed using workflow engines by Carnot AG and abaXX Technology AG ("WFMS" in Figure 4). Whenever necessary, communication between executing peers is performed by calling BAPI methods using Wf-XML.
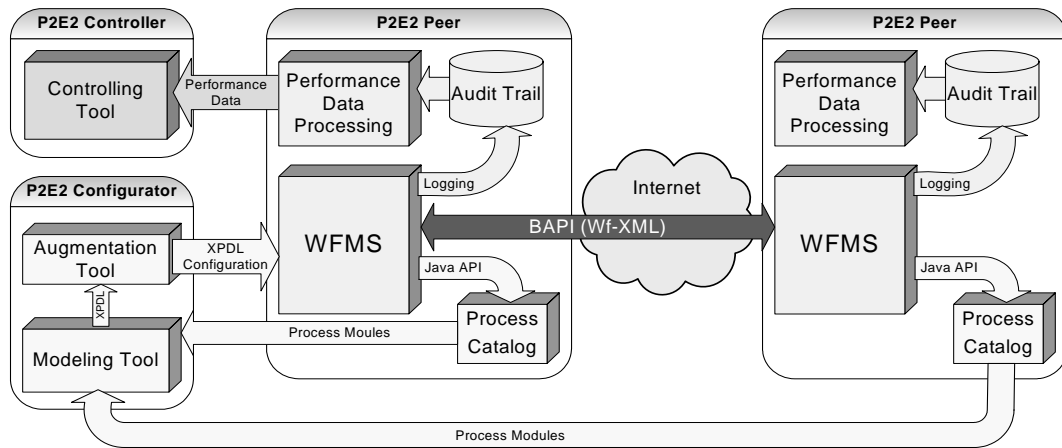


Fig. 4: P2E2 Technical Architecture

## 4.3  Distributed Business Process Controlling

During execution, every engine records performance data and stores it for the third lifecycle phase: controlling. The most basic performance data gathered during execution is stored in the audit trails of the workflow management systems (see Figure 4). However, mainly due to business secrecy, their content is not exposed directly. Instead, every peer processes its performance data to its liking and exposes the results or parts of the results over a specific web service interface exclusively. Of course, this information only refers to the execution of a process fragment, not the process as a whole.

The reassembly from fragments to the whole process is achieved using a specific controlling tool (see Figure 4). It first fetches performance information about process fragments from all participating peers using the web service described above. Then the information how the whole process is composed from process fragments is used to aggregate per-process information from per-fragment data.

# 5 Related Work

The approach presented tries to bring together several research areas that originally are addressed isolated. The concept of distributed business processes has raised ten years ago (e.g. [GrGr96], [AaWi98]). It was mainly driven by distributed system research and tried to archive the cross-system execution of workflows (e.g. [BaDa97], [ScSt94]). Such attempts also resulted in the definition of various standards (e.g. WF-XML) to simplify the interoperability of workflow management systems (cf. [WFMC96]). But they assume the existence of a single, atomic workflow specification model (e.g. [MuWo98]). On the other hand exists various approaches of distributed business process resp. workflow modelling (e.g. [GrGr95]). They describe the creation of singular models by multiple actors. But they mainly miss either the link to the distributed execution or the interconnection to the controlling task. Especially this task is neglected in other management approaches to cross-organizational business processes [DaHs01; LeRo02; PeKl99].

# 6 Conclusion and Outlook

In this paper, we have presented a concept for the cross-organizational business process lifecycle management, including distributed modelling, execution and controlling, that is already implemented in most parts. In particular we addressed and ensured the continuous IT support of all three lifecycle phases, the decision autonomy and secrecy demand of the participating organizations during all three lifecycle phases, and the technical and conceptual feasibility of our approach (which will be finally verified when the entire prototype is completed).

Currently, two business scenarios are evaluated with our concept. One of them is taken from the financial services sector and deals with factoring, the other one deals with supply chain management in international and national product distribution.

This concept was developed at the Competence Centre Business Integration (CCBI), Institute for Information Systems (IWi) at the German Research Centre for Artificial Intelligence (DFKI), Saarbruecken. The work is performed by clustering national and international funded research projects (esp. ArKoS, ATHENA, INTEROP, P2E2), intending the development of solutions for a better interoperability in business networks.

# References

[AaWi98]    van der Aalst, Willibrordus M. P.: Interorganizational Workflows. In: Proceedings of the PROLAMAT 98., 1998, pp. 21-43

[AdCh05]    Adam, O.; Chikova, P.; Hofer, A.; Vanderhaeghen, D.: Customer-Driven Process Management in Value-Added Networks Using an Architecture for Collaborative Business. In: Karduck, A. P. (ed.): Proceedings of CollECTeR (Collaborative Electronic Commerce Technology and Research) Europe 2005 - Collaborative Business, Session 2: Customer-Supported Activity Coordination, 2005, pp. 1-9.

[Arki02]    Arkin, A.: Business Process Modelling Language-Business Process Management Initiative. 2002.

[BaDa97]    Bauer, T.; Dadam, P.: A Distributed Execution Environment for Large Scale-Workflow Management Systems with Subnets and Server Migration. In: Proceedings of the 2nd IFCIS Conference on Cooperative Information Systems., 1997

[Cama02]    Camarinha-Matos, L. M.: Collaborative Business Ecosystems and Virtual Enterprises. Kluwer Academic, Norwell 2002.

[DaHs01]    Dayal, U.; Hsu, M.; Ladin, R.: Business process coordination: State of the art, trends, and open issues. In: Proceedings of 27th International Conference on Very Large Data Bases., 2001

[DaMa92]    Davidow, W. H. & Malone, M. S.: The Virtual Corporation - Structuring and Revitalizing the Corporation for the 21st Century. Harper Collins, New York 1992.

[GrAb01]    Grefen, P., Aberer, K., Hoffner, Y., & Ludwig, H.: CrossFlow: Cross-organizational Workflow Management in Dynamic Virtual Enterprises. In: International Journal of Computer Systems, Science and Engineering 15 (2001) 5, pp. 277-290.

[GrGr95]     Graw, G.; Gruhn, V.: Distributed Modeling and Distributed Enaction of Busi-
             ness Processes. In: Proceedings of the 5th European Software Engineering Con-
             ference. Berlin et al. Springer, 1995, pp. 8-27

[GrGr96]     Graw, G.; Gruhn, V.; Krumm, H.: Support of cooperating and distributed busi-
             ness processes. In: Parallel and Distributed Systems., 1996, pp. 22-31

[KeNü92]     Keller, G., Nüttgens, M., & Scheer, A.-W.: Semantische Prozessmodellierung
             auf der Grundlage "Ereignißgesteuerter Prozeßketten (EPK). In: Veröf-
             fentlichungen des Instituts für Wirtschaftsinformatik, A.-W. Scheer 1992.

[KuWe04]     Kupsch, F.; Werth, D.: Process-Driven Business Integration Using Peer-to-Peer
             Technology. In: Mehdi Khosrow-Pour (ed.): Innovations Through Information
             Technology., 2004, pp. 1294-1300.

[LeRo02]     Leymann, F.; Roller, D.; Schmidt, M.-T.: Web services and business process
             management. In: IBM Systems Journal 41 (2002), No. 2, pp. 198-211

[LuBu99]     Ludwig, H., Bussler, C., Shan, M., & Grefen, P.: Cross-Organizational Work-
             flow Management and Co-ordination - WACC99 Workshop Report. In: ACM
             ASIGGROUP Bulletin 20 (1999) 1.

[MuWo98]     Muth, Peter; Wodtke, Dirk; Weissenfels, Jeanine; Kotz-Dittrich, Angelika; Wei-
             kum, Gerhard: From Centralized Workflow Specification to Distributed Work-
             flow Execution. In: Journal of Intelligent Information Systems 10 (1998), No. 2,
             pp. 159-184

[ÖsFl00]     Österle, H., Fleisch, E., & Alt, R.: Business Networking - Shaping Enterprise
             Relationships on the Internet. Springer, Berlin et al 2000.

[PeKl99]     Pereira Klen, A. A.: Distributed Business Process Management. In: Proceedings
             of the IFIP / PRODNET Working Conference on Infrastructures for Virtual En-
             terprises: Networking Industrial Enterprises. Kluwer, 1999, pp. 241-258

[PrHa90]     Prahalad, C. K. & Hamel, G.: The core competence of the corporation. Springer,
             Berlin et al. In: Harvard Business Review 63 (1990) 3, pp. 79-91.

[RöSc01]     Röhricht, J. & Schlögel, C..: cBusiness. Erfolgreiche Internetstrategien durch Collaborative Business. Addison-Wesley 2001.

[SaOr99]     Sadiq, W.; Orlowska, M.E: Applying Graph Reduction Techniques for Identifying Structural Conflicts in Process Models. In: Proceedings of the 11th International Conference on Advanced Information Systems Engineering (CAiSE '99). Springer, 1999, pp. 195-209.

[Sand99]     Sandler, T.: Collective Action - Theory and Applications. University of Michigan Press 1999.

[Sche96]     Scheer, A.-W.: ARIS-House of Business Engineering: Von der Geschäftsprozeßmodellierung zur Workflow-gesteuerten Anwendung ; vom Business Process Reengineering zum Continuous Process Improvement. In: Scheer, A.-W. (ed.): Veröffentlichungen des Instituts für Wirtschaftsinformatik, No. 133, Saarbrücken : Universität des Saarlandes, 1996.

[ScNü95]     Scheer, A.-W.; Nüttgens, M.; Zimmermann, V.: Rahmenkonzept für ein integriertes Geschäftsprozeßmanagement. In: Wirtschaftsinformatik 37 (1995), No. 5, pp. 426-434.

[ScJo02]     Scheer, A.-W. & Jost, W.: ARIS in der Praxis - Gestaltung, Optimierung und Implementierung von Geschäftsprozessen. Springer, Berlin et al 2002.

[ScFi02]     Schoder, D.; Fischbach, K.: Peer-to-Peer : Anwendungsbereiche und Herausforderungen. In: Schoder, D.; Fischbach, K.; Teichmann, R. (eds.): Peer-to-peer: ökonomische, technische und juristische Perspektiven. Springer, 2002, pp. 3-21.

[Schu02]     Schulz, K.: Modelling and Architecting of Cross-Organizational Workflows, PhD Thesis. University of Queensland 2002.

[ScSt94]     Schuster, Hans; Jablonski, Stefan; Kirsche, Thomas; Bussler, Christoph: A client/server architecture for distributed workflow management systems. In: Proceedings of the 3rd International Conference on Parallel and Distributed Information Systems., 1994, pp. 253-256

[Wert06]    Werth, D.: Kollaborative Geschäftsprozesse - Integrative Methoden zur modell-basierten Deskription und Konstruktion. PhD Thesis, Saarland University 2006.

[WFMC96]    Workflow Management Coalition: Workflow Standard - Interoperability, Abstract Specification, TC-1012, Version 1.0., 1996