Advanced Business Process Management in Networked E-Business Scenarios

DOI:
10.4018/978-1-5225-9615-8.ch029

Document status and date:
Published: 01/01/2020

Document Version:
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Chapter 29
Advanced Business Process Management in Networked E-Business Scenarios

Paul Grefen
Eindhoven University of Technology, The Netherlands

Oktay Turetken
Eindhoven University of Technology, The Netherlands

ABSTRACT

In the modern economy, we see a shift towards networked business scenarios. In many contemporary situations, the operation of multiple organizations is tightly coupled in collaborative business networks. To allow this tightly coupled collaboration, business process management (BPM) in these collaborative networks is becoming increasingly important. We discuss automated support for this networked BPM: automated means to manage business processes that span multiple autonomous organizations - thereby combining aspects of process management and e-business. We first provide a conceptual background for networked BPM. We describe a number of research approaches in this area, ranging from early developments to contemporary designs in a service-oriented context. This provides an overview of developments in which we observe several major trends. Firstly, we see a development from support for static business processes to support for highly dynamic processes. Secondly, we see how approaches move from addressing simple business collaboration networks to addressing complex networks. Thirdly, we find a move from the use of dedicated information technology to the use of standard technology. Finally, we observe that the BPM research efforts move through time from pushing new BPM technology into application domains to using BPM to realize business-IT alignment in application contexts.

INTRODUCTION

In the past, business process management (BPM) used to be a rather internal issue for most organizations: organizations typically operated their business processes in a stand-alone mode without explicit connection to their business partners. Cooperation scenarios with other organizations obviously existed, but these scenarios were mostly based on the exchange of physical goods and information (e.g., on the

DOI: 10.4018/978-1-5225-9615-8.ch029

basis of electronic data interchange) – not on the execution of integrated business processes by the collaborating partners. Supply chains and service chains were often loosely connected from an operational perspective.

Several important developments have dramatically changed the context in which organizations collaborate, however. In the first place, products and services produced have become far more complex, thus requiring more business capabilities and hence larger networks of collaborating organizations. A clear example is the automotive industry. Here, products have become increasingly complex (Maxton & Wormald, 2004) and consequently, business networks have become larger (von Corswant & Fredriksson, 2002). The fact that global competition forces organizations to concentrate on core business activities only amplifies this development. Secondly, both product specifications and market circumstances have become much more dynamic, thereby requiring business networks and the processes they execute to become more dynamic too (Grefen et al., 2017). This is illustrated for example by Gartner stressing the importance of dynamic BPM for companies to deal with 'increasingly chaotic environments' (Gartner, 2010). Thirdly, business market paradigm changes, like mass customization (Vandaele & Decouttere, 2013) and demand chain orientation (Verdouw, Beulens, Trienekens, & Van Der Vorst, 2011) require much more tightly synchronized business processes across individual organizations in a business chain. Fourthly, time pressure has become much greater in the setup and execution of collaborations between organizations - just-in-time behavior is becoming more important. These four developments are forcing organizations to pay much more attention to how they cooperate, not only to what they exchange. In other words: organizations are forced to co-operate in business processes that span business chains and take part in the design and management of these inter-organizational business processes.

To deal with the complexity of inter-organizational business processes and obtain the required efficiency in setting them up and executing them, automated systems are required for managing business process in business networks. These automated systems should support a number of tasks. They should provide support for the design and configuration of inter-organizational business processes. As we will see in the next sections of this paper, support may be in the form of interactive design tools, but may also go into the direction of fully automatic configuration of inter-organizational business processes, based on predefined sub-processes within participating organizations. These automated systems should support the automated management of the execution of inter-organizational business processes, i.e., that process logic that actually links the internal business processes of multiple autonomous organizations. Then, these systems should support the synchronization of inter-organizational business processes with the internal business processes of the organizations. In the manufacturing domain, for example, the German Industrie 4.0 initiative (Germany Trade & Invest, 2014) illustrates this. These systems have the characteristics of both BPM and electronic business support. Depending on the dynamicity of the targeted application domain, the electronic business aspects become more prominent (this is for example very explicit in the CrossFlow project, which we discuss in a later section).

In the following, we discuss the development of systems for BPM in business networks. The next section provides a background by discussing the differences between intra-organizational and inter-organizational business processes. A three-level framework is explained that shows how to relate these two kinds of processes. Then, the following section discusses early approaches towards inter-organizational BPM. The next four sections present approaches, architectures and technologies of four major projects from the research experience of the authors: CrossFlow, CrossWork, XTC and CoProFind. In doing so, attention is paid to both business process specification and business process enactment, including contractual and transactional aspects. The discussion explicitly shows the development from 'traditional' workflow management via advanced, structured inter-organizational BPM to service-based, highly dynamic business process interaction. We end with a concluding section, which presents an overall analysis of the discussed research projects and identifies a number of important trends in the area of BPM in business networks.

CONCEPTS FOR INTER-ORGANIZATIONAL BPM

In this section, we set the stage for inter-organizational business processes in several steps. In the next subsection, we first discuss the concept of a business process within one organization: an intra-organizational business process. Then, we move to the concept of a business process across multiple organizations: an inter-organizational business process. We will see how control flow interfaces are important here. To explain how intra- and inter-organizational processes are related, we discuss a three-level framework. In the last part of this section, we add the aspect of dynamism to inter-organizational business processes, i.e., the aspect of collaboration networks that change over time. One thing is important to understand here: when we speak of 'organizations', these may be autonomous business entities (like commercial organizations) but also autonomous departments of a single business entity.

The Concept of Intra-Organizational Business Processes

An intra-organizational business process is completely run from the process point of view within the boundaries of a single organization (or autonomous part of an organization). In other words: process flow control is completely within a single organization, as usual with many 'traditional' views on business process automation. An intra-organizational business process may call business services (which may be implemented by business processes (Sewing, Rosemann, & Dumas, 2006)) of other organizations, but does not 'see' the structure or status of these other services explicitly. We can hence define the concept of intra-organizational business process as follows:

An intra-organizational business process is a business process that is enacted by a single organization, but which may call black box business services of other autonomous organizations.

An intra-organizational business process typically has a number of characteristics:

- It has a single point of process control from a conceptual point of view (it may be technically controlled by a distributed system, but this is transparent to the users).
- There are no reasons for explicit hiding of structure or status details of the parts of the process to other parts of the process.
- The process is run in an environment of which the heterogeneity is controlled, both in terms of process specification languages (syntax and semantics) and protocols used, as in terms of the technical infrastructure (like business process management systems – BPMS and middleware). Often, one even finds a completely homogeneous process management environment within a single organization, i.e., one choice has been made for business process support technology.

The Concept of Inter-Organizational Business Processes

After having discussed intra-organizational business processes, we turn to the concept of inter-organizational business processes. We use the following definition of inter-organizational business process:

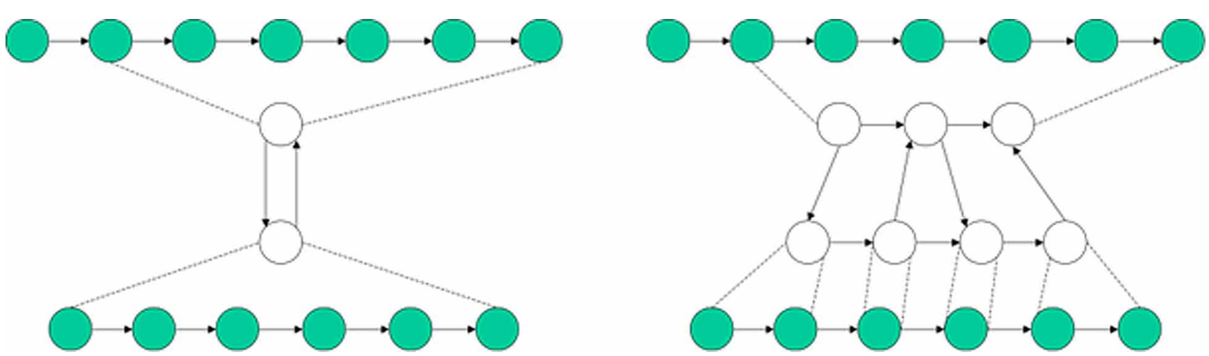
An inter-organizational business process is a business process that is enacted by two or more autonomous organizations, of which at least one organization exposes a non-black box projection of the explicit control flow structure of a process to the other organization(s).

This definition states that in an inter-organizational business process, at least one party must make a non-trivial (consisting of more than one single activity) process structure accessible to its collaborator(s). This process structure is typically a projection of an intra-organizational business process, which is also referred to as a process view (Eshuis & Grefen, 2008). We will see more of this projection relation in the rest of this paper. In the 'more traditional' inter-organizational service invocation (as found in the basic service-oriented computing paradigm (Boukhedouma, Alimazighi, & Oussalah, 2017)), we don't see explicit control flow sharing between organizations: control flow of a service implementation is completely encapsulated by the service specification. Shared control flow can become complex and therefore require careful modeling and analysis (van der Aalst, 2000).

As such, we can distinguish between various classes of process coupling modes (Grefen & Vonk, 2006). Two classes are illustrated in Figure 1, where the open circles in the center levels indicate control flow interfaces, the filled circles in the top and bottom levels indicate local processes that implement what is offered in the interfaces. The process coupling classes range from black box coupling (left hand side of the figure) to explicit two-way control flow sharing, which is called open box coupling (right hand side). The black box class does actually not comply with our definition of inter-organizational workflow, as it is in principle no more than a call-and-return service scenario – the fact that the two services at the interface are implemented by processes is completely invisible to the other party.

In (Grefen & Vonk, 2006), also glass box coupling and half-open box coupling are discussed, which are in between black box and open box coupling. Glass box coupling allows one party to observe the status of the process of the other party, but does not allow interference with it in either direction. Half-open box coupling does allow one-way interference, i.e., one party can exert control over the other

Figure 1. Black box processes (left) and shared control flow (right)



party. We will see in the remainder of this paper that the half-open box coupling mode is relevant for the process-oriented service outsourcing paradigm (in the CrossFlow project, discussed in a later section).

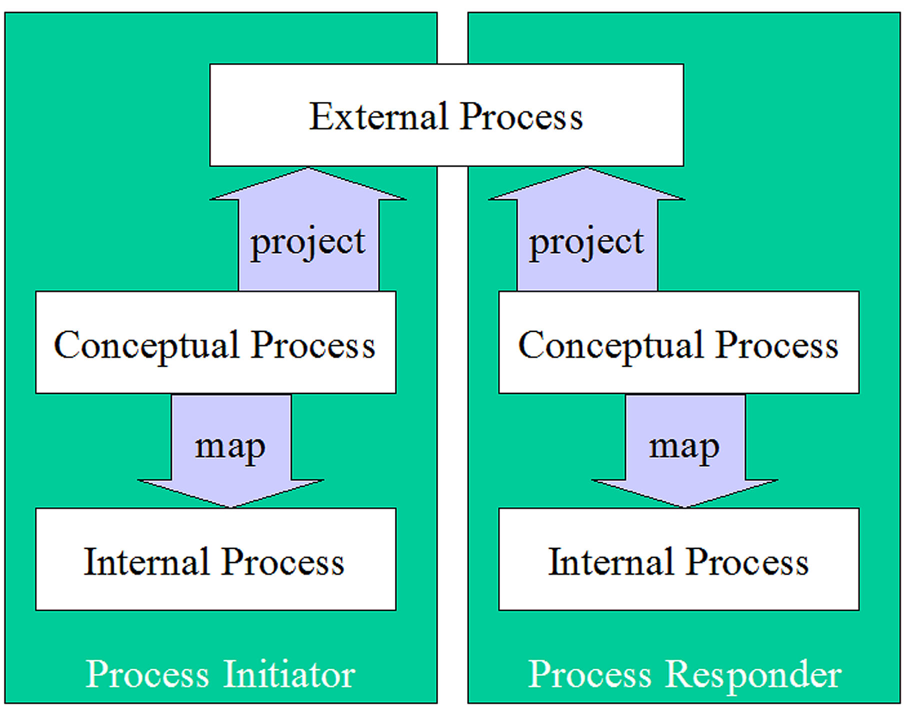
An inter-organizational business process differs in several characteristics from an intra-organizational business process:

- It explicitly has several points of control, as it is run by multiple autonomous organizations.
- There are two explicit reasons for hiding process details: the fact that some details are private to an organization (for reasons of competition) and the fact that some details are irrelevant to other organizations (as they pertain only to internal matters of a single organization).
- The process is often executed in a heterogeneous environment. The fact that multiple autonomous organizations collaborate implies that different local choices have been made with respect to languages, protocols and infrastructures for BPM.

Levels in Inter-Organizational Business Processes

As we have seen above, inter-organizational business processes run across multiple autonomous parties, interconnecting intra-organizational business processes of these parties. To clarify the relation between these two types of processes, the three-level process framework has been proposed (Grefen, Ludwig, & Angelov, 2003). This framework is shown in Figure 2 with two collaborating parties shown as the two large boxes (but can be trivially extended to more parties): one party initiates the process-based collaboration, the other responds by engaging in the collaboration (the two roles are shown in the figure). At both parties, process specifications exist on three levels, as explained below.

Figure 2. Three-level business process framework



The middle level of the framework is the conceptual level for business process models. At this level, business processes are designed, i.e., their intended functionality is specified in abstract terms. The conceptual level is independent from both (internal) infrastructural specifics and (external) collaboration specifics. It does specify the main aspects of intra-organizational processes, taking collaboration in inter-organizational business processes into account in an abstract way.

The bottom level is the internal level, at which process models are directly interpreted by process management systems. Hence, process models at this level are in general technology-specific, e.g., they are described in the specification language of a specific workflow management system (WfMS). The internal level is of an intra-organizational nature. Models at the conceptual level are mapped to the internal level for process enactment (e.g., by a WfMS or other process-aware information systems). For details of this mapping see (Grefen et al., 2003). Note that the mapping is not always trivial, as the functionality of the internal level process management systems may be limited: not all constructs used at the conceptual level may be supported. In that case, it may be possible to map 'missing constructs' to a (sometimes complicated) combination of constructs that are supported.

The top level is the external level, at which process interaction with external parties is modeled for use in inter-organizational business processes. At this level, process models are market-specific, i.e., have to conform to standards and/or technology used in a specific electronic market. Models at the conceptual level are projected to the external level for integration with processes of partner organizations to form inter-organizational business process. Note that projection is used here, since only relevant parts of the conceptual model are of interest at the external level. In the projection, process details are hidden by aggregation/abstraction of process steps.

Note that some authors use the terms 'public process model' and 'private process model' (e.g., (Fdhila, Indiono, Rinderle-Ma, & Reichert, 2015)), where we use the terms 'external process model' and 'internal process model'.

Static vs. Dynamic Inter-Organizational Business Processes

So far in this section, we have looked at the characteristics of inter-organizational BPM. We have, however, not yet looked at the functionality required for the dynamic formation of collaborations. As we have seen in the introduction, this is essential in modern business – thus, this last point is addressed here.

Dynamic formation of collaborations implies that an organization prepares for inter-organizational BPM (the what and the how), without yet knowing which the collaboration parties will be (the who). These parties are selected during the execution of a case or set of cases (business process instances), where selection takes place on the basis of characteristics of the case(s) under execution and current market conditions. To allow so, potential collaborators (process responders in terms of the three-level framework shown in Figure 2) expose their process offerings (at the external level) in market places such that they can be found by process initiators.

Earlier in this section, we have given a definition of an inter-organizational business process. This definition does not yet take the dynamic nature of inter-organizational BPM into account, however. Therefore, we present another definition that adds dynamism to the previous definition:

A dynamic inter-organizational business process is an inter-organizational business process that is formed dynamically by (automatically) integrating two or more external processes provided by the involved organizations. Here dynamically means that collaborator organizations are found at or just *before process run-time by searching business process market places based on the characteristics of (a set of) business process cases and market conditions.*

Note that the above definition is formulated in terms of (near) run-time dynamism, i.e., the formation of an inter-organizational business process instance right before or during the execution of that process instance (or limited set of instances). In this approach, dynamism is not obtained by redesigning the process specification. This is different from design-time dynamism, in which flexibility is achieved by explicitly redesigning the process specification at specific points of time. There is extensive work on various approaches to achieve flexibility in business processes (e.g., (Schonenberg, Mans, Russell, Mulyar, & van der Aalst, 2008; Weber, Reichert, & Rinderle-Ma, 2008) and work on information technology to support flexible processes (e.g., (Wang & Kumar, 2014)).

After the conceptual discussion of inter-organizational BPM in this section, we turn to concrete approaches and systems for this purpose in the next section.

EARLY DEVELOPMENTS IN INTER-ORGANIZATIONAL BPM

In the mid-nineties of the previous century, automated support for intra-organizational BPM (at that time usually labeled as workflow management) entered a mature stage. Attention was paid to advanced aspects like transaction management, exception management, and flexibility issues of business processes (Grefen, Pernici, Sanchez, Vonk, & Boertjes, 1999). At the same time, collaboration between organizations was changing (as explained in the introduction of this paper), also fueled by the rise of e-commerce in that time frame. These two developments together gave way to research interest into automated support for inter-organizational BPM.

In this section, we discuss these early developments. Below, we first give a brief overview of research efforts in the context of inter-organizational BPM. Then, we discuss one project (WISE) in more detail.

Brief Overview of Early Projects

The FlowJet project at Hewlett-Packard was one of the early projects in the domain of inter-organizational BPM. The project aimed at coupling various types of workflow systems in E-business contexts (Shan, 1999). The system was designed with modularity as a starting point to provide feature on demand capabilities (Shan, Davis, Du, & Huang, 1997). Dynamic resource brokering is within the scope of the project, but explicit contracts for detailed service specification are not considered.

The WISE project is comparable to FlowJet as it uses inter-organizational workflow management technology for business-to-business E-commerce scenarios (Alonso et al., 1999; Lazcano, Schuldt, Alonso, & Schek, 2001). WISE is discussed in more detail in the next subsection.

MariFlow (Cingil, Dogac, Tatbul, & Arpinar, 1999) follows a similar approach to the WISE project but is specifically targeted at the marine industry (Dogac et al., 1999). The project aims at providing process management capabilities comparable to the WISE system, enhanced by an advanced marketplace for service contracts.

The COSMOS project developed an architecture that allows organizations to offer and search for services in a catalogue, a negotiation platform, and facilities for contract signing (Griffel, Boger, Weinreich, Lamersdorf, & Merz, 1998). Once the contract is signed, workflow specifications are derived from the contract or encapsulated in the contract constituents of the offering party and a new workflow instance is started.

WISE

The WISE project (Workflow based Internet SErvices) at ETH Zürich aims at providing a software platform for process based business-to-business electronic commerce (Alonso et al., 1999; Lazcano et al., 2001). In doing so, the project focuses on support for networks of small and medium enterprises. The software platform used in WISE is based on the OPERA kernel (Alonso, Hagen, Schek, & Tresch, 1997).

WISE relies on a central workflow engine to control inter-organizational processes (called virtual business processes). As we will see in the discussion of CrossFlow in the next section, a distributed engine approach has been used elsewhere. A virtual business process in the WISE approach consists of a number of black box services linked in a workflow process (Alonso et al., 1999). A service is offered by an involved organization and can be a business process controlled by a WfMS local to that organization – but this is completely orthogonal to the virtual business process.

Specification of virtual business processes in WISE is performed using the Structware/IvyFrame tool (Lienhard, 1998), which is internally based on Petri Nets. This tool and its specification technique are used to construct both the conceptual structure of inter-organizational processes and the specifications of services exchanged between organizations in a virtual enterprise. Hence, it can be placed both at the conceptual and at the external levels of our three-level framework (see Figure 2).

The Structware/IvyFrame tool has, however, also characteristics related to the internal level, as it not only supports process creation, but also configuration management of underlying enactment platforms (Lazcano et al., 2001).

The graphical representation produced by the Structware/IvyFrame process definition tool is compiled into a language called Opera Canonical Representation (OCR) (Hagen, 1999). This language is used internally by WISE to create process templates. As OCR is focused towards process enactment in the context of a specific platform, we can place it at the internal level of our framework. Note, however, that OCR is used for inter-organizational coordination, so has external level characteristics too.

CROSSFLOW: DYNAMIC PROCESS-BASED SERVICE OUTSOURCING

As discussed in the introduction of this paper, many organizations nowadays focus on their core business processes and source processes from partners in the market to perform the additional parts of the process required to reach their business goals. We call this the service outsourcing paradigm. In this paradigm, the outsourcing organization (initiator in terms of the three-level framework) is referred to as service consumer, the service implementing organization (responder) as service provider. The details of service outsourcing are specified in a contract between both parties. The combination of service consumer and service provider can be seen as a virtual enterprise that presents itself to a third party (for example a customer) as a single entity.

Traditionally, these virtual enterprises have a more or less stable character over time, i.e., the combinations of service consumer and provider are fixed over long periods of time (e.g., several years). As discussed before, in dynamic e-business settings, however, players in a market and competitive situations change that fast that a more dynamic approach is required to service outsourcing to create or retain a competitive position. This means that in service outsourcing, service consumers dynamically determine which service providers to use in the enactment of their business processes. We call this business model dynamic service outsourcing, the temporary organization formed by service consumer and service provider a dynamic virtual enterprise. Depending on the business domain and the specific inter-organizational business process, a dynamic virtual enterprise can have a life span ranging from a few minutes to a few months.

The European CrossFlow project has developed information technology for advanced process support in dynamic virtual enterprises (Grefen, Aberer, Hoffner, & Ludwig, 2000). Below, we first discuss the CrossFlow approach to inter-organizational BPM. Then, we pay attention to the architecture of the CrossFlow system. We show that the architecture is of a dynamic kind, following the life cycle of a dynamic virtual enterprise.

The CrossFlow Approach

The CrossFlow approach to inter-organizational BPM is characterized by four main aspects (Grefen et al., 2000):

- Dynamic service outsourcing;
- Contract-based service specification;
- Fine-grained, advanced interaction;
- Contract-dependent generation of enactment infrastructure.

Below, we elaborate these four aspects. Note that the trading-based approach to service outsourcing means that CrossFlow can be considered a project investigating the intersection of workflow management and electronic commerce technology.

Dynamic Service Outsourcing

As indicated above, the CrossFlow approach to inter-organizational workflow management is based on a dynamic service consumer/provider paradigm. This means that an organization that wants a service to be performed on its behalf (the service consumer) outsources this service to an organization that can perform this service (the service provider). This outsourcing is performed dynamically, which means that the decision to outsource is taken during the execution of the process instance (case) requiring the service and that the provider is chosen dynamically.

The dynamic search for compatible business partners is performed through a matchmaking facility, which plays the role of a service marketplace. Service providers advertise their services in this facility. Service consumers query the facility for required services. Matchmaking of services is based on the fact that in many markets standard business practices, standard languages and ways of describing services, and standard legal forms and processes have evolved, resulting in common contract templates.

The interaction between service consumers and providers is based on contracts, as described below. Service providers advertise their services in contract templates, which are completed to individual contracts by service consumers.

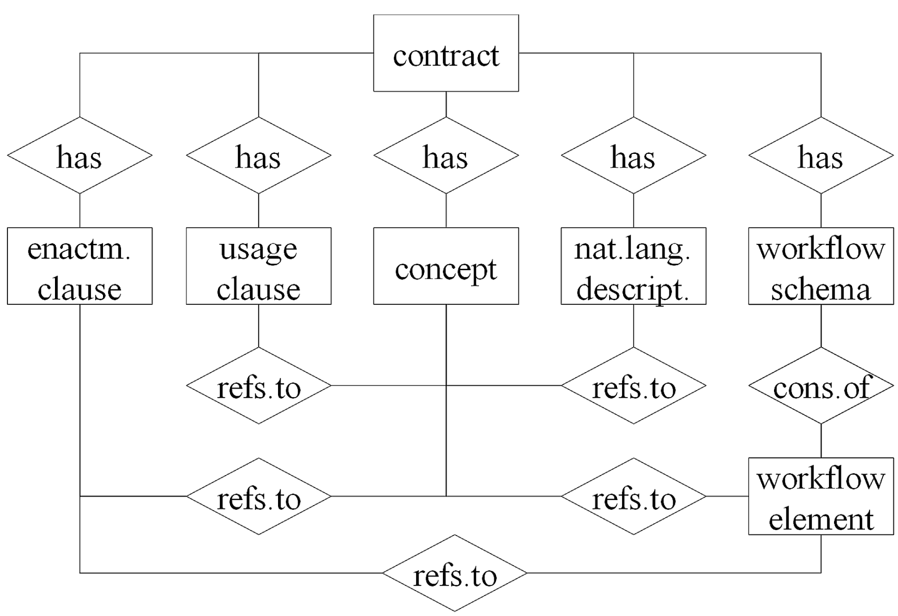
Contract-Based Service Specification

In the CrossFlow approach, the interaction between service consumer and service provider is completely specified in a contract, which formalizes the electronic business relation between the two parties. The contract defines all relevant details of the service provision (Koetsier, Grefen, & Vonk, 2000). Traditionally, this is limited to an identification of the service and all parameters required for the execution of the service. CrossFlow contracts, however, also entail a specification of the process used to execute the service. Specification of this process allows for further integration of consumer and provider processes than a mere black box process would allow. This high level of integration is essential for the close partnerships found in virtual organizations.

In virtual organizations, however, a partner does not require full operational details of other partners. Rather, a well-defined abstraction of their operation should be used to obtain an effective view on both data and processes. As partners in a virtual organization often have different IT platforms, a heterogeneous environment exists. This heterogeneity should be addressed by abstraction of technical details of partners. For both reasons, CrossFlow contracts define the interaction between organizations not in terms of their WfMSs, but on an abstraction level above these systems (i.e., the external level in the three-level framework).

The conceptual model of the CrossFlow contract structure (Koetsier et al., 2000) is shown in Figure 3. In this model, we see that a contract consists of five sections, four of which are meant for automated processing (the natural language description is meant to make the contract understandable to humans). The concept section defines the central concepts the contact is about. The usage clauses and enactment clauses sections describe how to use the contract (e.g., how to establish a new contract instance) and how to execute the contents of the contract - this has a strong connection to electronic business charac-

Figure 3. Conceptual model of CrossFlow contract structure



teristics. The workflow schema section specifies the business process to execute the service governed by the contract, which makes the BPM aspect of the approach explicit.

Fine-Grained, Advanced Interaction

The CrossFlow approach is focused on tightly integrated service consumer and provider processes. For this reason, a common service process specification is included in CrossFlow contracts. To support the tight coupling of processes, additional advanced notions of interaction are required. These notions are operationalized in so-called cooperation support services (CSSs) (Grefen et al., 2000). A broad spectrum of CSSs is relevant for inter-organizational workflow management, like remote process monitoring and control, inter-organizational transaction management, automatic service remuneration, trust and security management, etc. The design of these services should be such, that they can be selected and combined in a modular way, depending on the application context.

In the context of the CrossFlow project, three areas of advanced cooperation support services are addressed. The selection of these three areas is based on the interest and background of the project partners. Quality of Service (QoS) monitoring allows tracking the progress of outsourced services, both online during service execution and offline to provide aggregate information. Level of Control support provides means for high-level, inter-organizational transaction management and consumer-controlled process control over outsourced services. Flexible Change Control allows dynamic changes to execution paths of outsourced processes during their execution.

Contract-Dependent Generation of Enactment Infrastructure

The enactment infrastructure that connects the information systems of service provider and consumer is dynamically set up according to the contract and a specification of the way the contracted service is to be implemented and supervised. To allow this, the cooperation support services are mapped to modular system building blocks and a message-based integration mechanism is used to provide the required level of flexibility. The mechanism uses a subscribe mechanism to cater for flexibility. We will see details of the infrastructure generation in the description of the architecture below.

The CrossFlow System

After discussing the CrossFlow approach above, we now turn to the architecture of the CrossFlow system, which describes support for contract-based inter-organizational workflow management. The CrossFlow architecture includes support for both contract establishment and contract (service) enactment. The architecture is based on commercial WfMS technology, shielded from the CrossFlow technology by an interface layer. In the project, IBM's MQSeries Workflow (formerly known as FlowMark) BPM platform is used.

The lifecycle of service outsourcing in CrossFlow consists of four phases (Hoffner, Field, Grefen, & Ludwig, 2001): contract establishment, dynamic infrastructure configuration, contract enactment, and dynamic infrastructure disposal. We describe each of the four service lifecycle phases below. We conclude this section with a discussion of technical details of the prototype implementation. More information on the architecture can be found in (Hoffner, Ludwig, Gulcu, & Grefen, 2000).

Contract Establishment

The following describes a typical sequence of events that leads to the establishment of a contractual relationship between the provider and consumer organizations – illustrated in Figure 4.

When the provider WFMS is ready to receive requests for enactment of a process on behalf of a consumer organization, it notifies its Contract Manager of its readiness. A Workflow Module (WM) acts as an interface layer to shield the Contract Manager from details of specific WFMSs. It does so by providing a bi-directional activation interface to the Contract Manager. The Contract Manager selects a pre-existing Contract Template that describes the service and its associated QoS guarantees, work schedule, monitoring and control points as provided by the service, etc. Appropriate values for these service guarantees including the cost of the service must then be determined. These will be decided according to the capabilities of the enactment infrastructure, the resources that the provider is willing to assign to the enactment, and the price associated with the resources. In addition, the requirements that the provider places on the consumer within the terms of the Contract Template are also specified. The service description and the demands are translated into the property and constraint language of the matchmaking facility. The result is then advertised into the trader that serves the specific market. In a competitive market, several provider organizations will advertise the same service with the same associated service contract but with different values describing QoS, scheduling and other guarantees, and the price of the service.

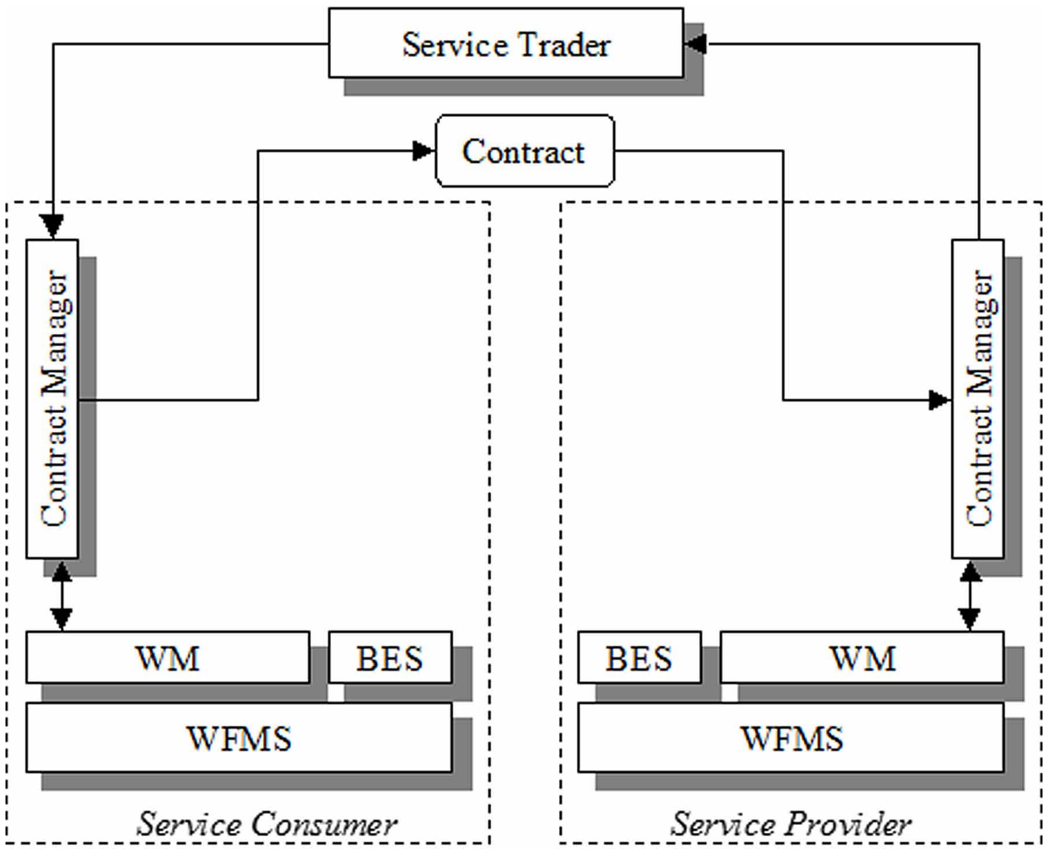
When the consumer WFMS reaches a task that it wishes to have enacted on its behalf externally, it notifies its Contract Manager (again through a Workflow Module). The consumer Contract Manager selects a pre-existing Contract Template that describes the service it is looking for in terms of the QoS guarantees, work schedule, monitoring, and control points it wishes to have associated with the provided service. Unlike the provider who specified those parameters as properties, the consumer can place demands in terms of the speed by which it wishes to have the work completed and the maximum price it is willing to pay for it, for example. The consumer must also describe what it offers in terms of its willingness to pay and the means by which it can pay, for example. The consumer's promises and demands are translated into the property and constraint language of the trader. The result is then sent as a search query into the trader serving the market.

The trader compares the promises and demands made by the consumer against the offers previously posted in it by market providers. The matching offers are then sent back to the consumer. The consumer Contract Manager can then compare the offers and select the one that suits its requirements best. By notifying the selected provider, the consumer in effect makes a counter-offer that the provider can accept or reject. The acceptance of the counter-offer signifies an agreement between the two organizations and an electronic contract is established.

Dynamic Enactment Infrastructure Generation

Once a contract has been made between service consumer and provider, a dynamic contract and service enactment architecture is set up in a symmetrical way for both partners, as illustrated in Figure 5. For this purpose, the Contract Manager activates the Configuration Manager. The configuration of this enactment infrastructure is based on the contract and requires a number of components:

Figure 4. CrossFlow architecture in contracting phase

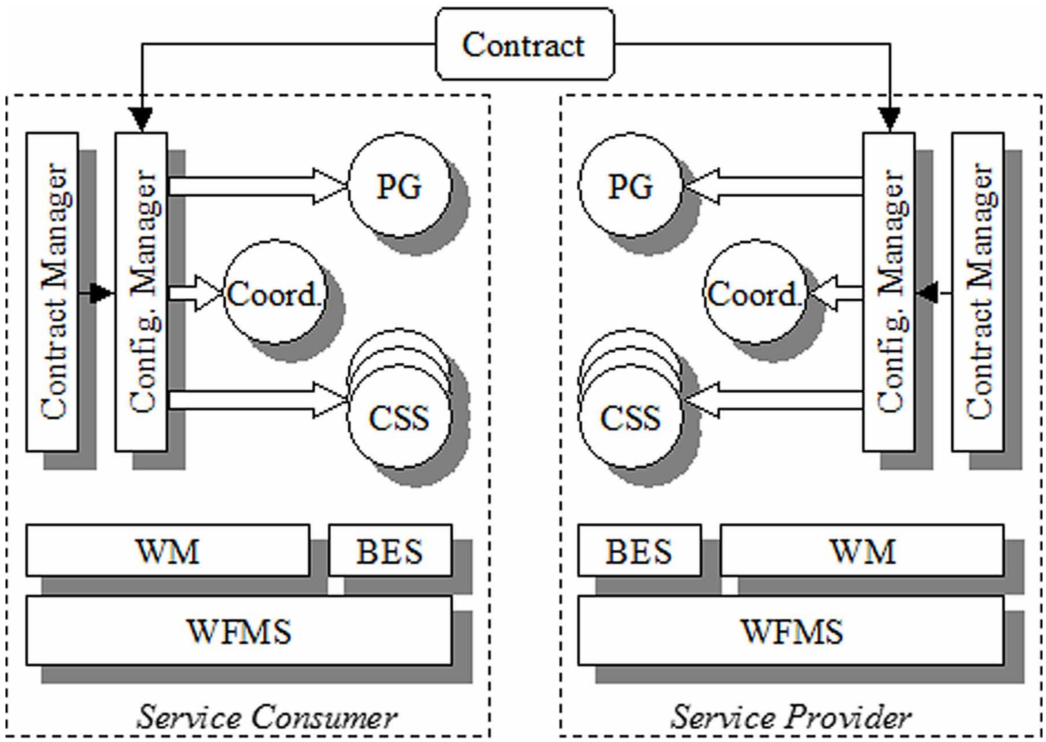


- Cooperation Support Service (CSS) modules implement the advanced cooperation support functionalities. Level of Control, QoS, and Flexible Change Control were chosen in the CrossFlow project, but other CSS modules are possible (as we have seen before).
- Proxy-Gateways (PG) deal with the crossing of domain boundaries by facilitating the interaction between the organizations' systems, by translating between the internal-external and organizational differences on a syntactical level, and by monitoring and controlling exit-entry to protect the organization's integrity and security.
- Coordinators (Coord.) are used at each site to connect the various components such as the CSSs, the PG, and the WFMS through the WM.

The functionality of contract and service enactment components is largely dependent on the contents of the contract and the manner in which each organization sees fit to carry out their part of the enactment.

The Internal Enactment Specification (IES) is the organization-specific blueprint that specifies how the contract is to be enacted. It defines which internal resources can be used in which way. For this purpose, the IES describes which components are needed to enact the service and, in addition, it describes the contract implementation policy for each of the deployed CSS components. It also provides the mapping

Figure 5. CrossFlow architecture in configuration phase



between the workflow process specified in the contract and the workflow process as actually enacted internally by the service provider and similarly the mapping of the data related to the workflow enactment.

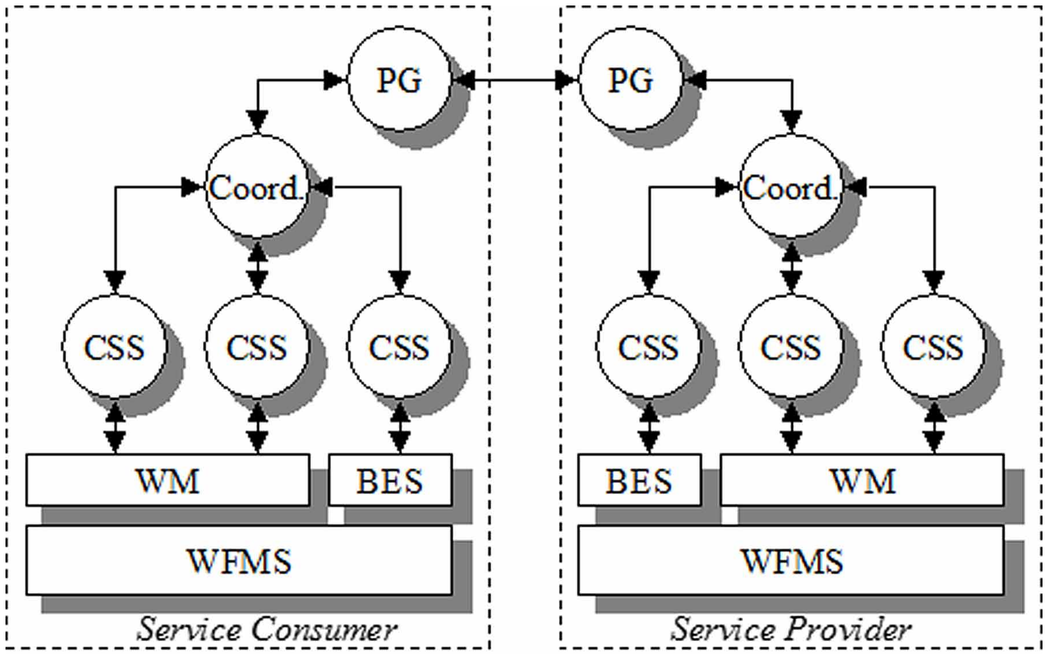
Using the contract and the corresponding IES, the Configuration Manager instantiates and configures a coordinator, a proxy-gateway and a set of CSS components to enact the contract. These components are next linked to each other and to the WM and BES components that provide interfaces to systems at the internal level during contract enactment (shown in Figure 6 and described next).

Contract Enactment

When the set-up described above is ready, the consumer can initiate the actual enactment of the outsourced business process by contacting the provider. The enactment takes place using the dynamically constructed infrastructure as illustrated in Figure 6 (in a simplified way).

Any monitoring information agreed upon in the contract to be provided from the provider to the consumer can either be sent as a notification or requested by the consumer. As a result of the progress update, the consumer may wish to request the provider to modify the enactment of the business process. This may include a change of parameters or a change in the process direction or structure, depending on the contract. Further monitoring information may pass as a result and more changes may be initiated where necessary. Ultimately, an indication of the completion of the process and its results will be passed to the consumer.

Figure 6. CrossFlow architecture in execution phase



Where appropriate, the enactment infrastructure can access the Back-End Systems (BES) interface for specific services. These systems offer CrossFlow services on a permanent basis (not related to the enactment of a single contract) and other more general services.

Dynamic Infrastructure Disposal

When all the administrative processes have been completed and both sides are satisfied with the provision and consumption of the service, the infrastructure created earlier can be dismantled. This means that coordinator, CSS modules, and proxy gateways relating to the service can be deleted.

CrossFlow in Retrospective

The CrossFlow project application scenarios are positioned in the logistics and insurance business domains. For the logistics domain, a highly dynamic scenario was developed for the distribution of mobile phones to customers, in which a telecom company is the service consumer and logistics providers are service providers. For the insurance domain, a scenario was elaborated for damage claim assessment for motor vehicle insurance. Here, the insurance company is the service consumer and assessment expertise firms are the service providers. Both scenarios presented huge steps forward with respect to dynamism in inter-organizational BPM at that time. The CrossFlow approach does have two important limitations though.

Firstly, CrossFlow is limited to one-to-one service outsourcing process topologies. Though complex networks can be built by combining multiple service outsourcing scenarios, direct collaboration of more than two partners in one global business process is not possible in the CrossFlow approach. In the next section, we will see how the CrossWork approach lifts this limitation.

Secondly, CrossFlow is based on dedicated technology. Although CrossFlow uses a commercial WfMS as its basis and a service broker based on CORBA standards, the heart of the system is dedicated technology directly realized in Java. The languages and protocols developed in the CrossFlow project are of a dedicated nature too. An example of a dedicated language is the CrossFlow contract language. Later in this paper, we will see how the XTC project is positioned in a service-oriented architecture (SOA) context from the very start.

CROSSWORK: DYNAMIC BUSINESS PROCESS COMPOSITION

In the previous section, we have seen how CrossFlow supports the bilateral service outsourcing paradigm. Where direct, peer-to-peer interaction between more than two business partners is required, a more general collaboration paradigm is required, however. In the CrossWork project (Grefen, Eshuis, Mehandjiev, Kouvas, & Weichhart, 2009; Grefen, Mehandjiev, Kouvas, Weichhart, & Eshuis, 2009; Mehandjiev & Grefen, 2010), these general inter-organizational business process topologies are addressed, which are called business network processes (BNPs).

Below, we first discuss BPM in these general, peer-to-peer business topologies. We show how these are centered on the concept of instant virtual enterprise, which is a variation of the dynamic virtual enterprise concept of the CrossFlow project. After that, we turn to the CrossWork system and its architecture. We end the section again with placing the approach into retrospective.

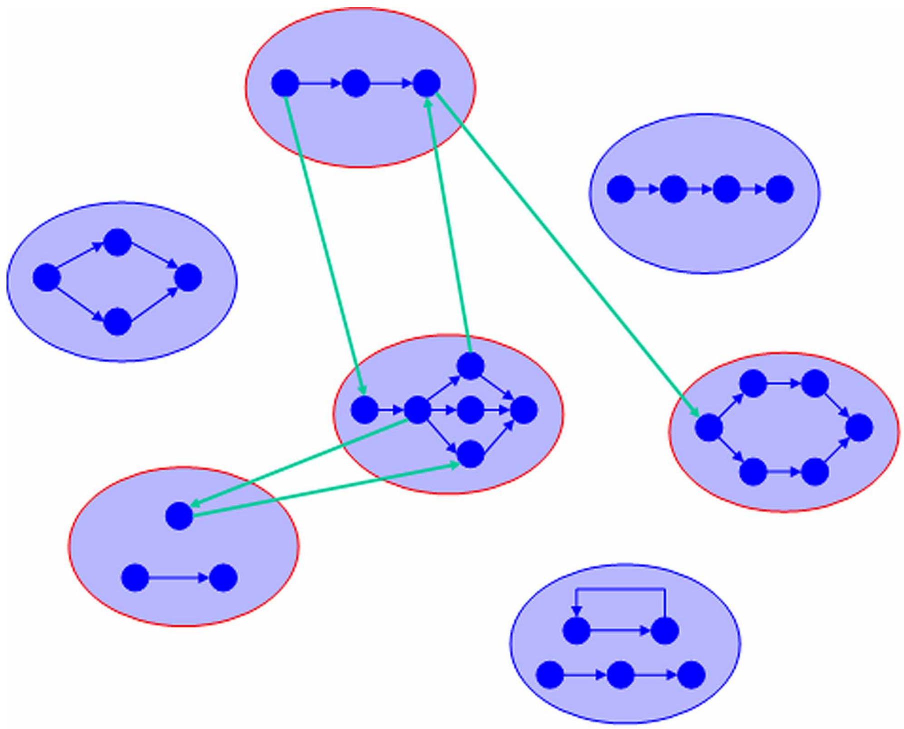
Business Processes in Instant Virtual Enterprises

Many business domains nowadays rely on tight collaboration between (possibly large numbers of) autonomous business organizations, such that each of these organizations can fulfill a sub-goal of an overall business goal. Such a collaboration is commonly called a virtual enterprise (VE). Tight collaboration in a VE implies that the local business processes of the collaborating business organizations need to synchronize at a possibly detailed level. This means that local business processes are actually 'woven' into a global business process (the business network process).

Figure 7 shows an example of a VE in a business market consisting of a set of possible business partners (here, seven are shown, but in practice the number is usually much greater). Four business organizations have organized into a VE. Each organization has its local business process, shown inside the ellipses. The local business processes of the partners in the VE are organized into a global VE business process by adding the business process links between the ellipses in such a way that process dependencies are taken into account and overall process quality requirements (such as throughput times) are met.

Operating in frequently changing markets implies that virtual enterprises cannot have a stable character over time: changing market demands require that new business competences are added, existing business competences may become useless, or different selection criteria are used with respect to QoS parameters. The consequence of this is that VEs must be created in a dynamic way: based on circumstances at a specific point in time, a VE must be set up quickly for a limited amount of operating time. This highly dynamic version of VE is the instant virtual enterprise (IVE). An IVE is mainly determined

Figure 7. Example instant virtual enterprise in a market



by the selection of the partners that collaborate in it and by the inter-organizational process links that are woven between the partners. The local business processes of the individual partners usually remain stable over time, as they heavily depend on investments made by these partners. Required flexibility in IVE process definition is obtained by the flexible composition of the global process.

IVEs have a dynamic character, i.e., they are created and dismantled in the course of time. The trigger for the creation of an IVE is a new business opportunity in a specific market, observed by an organization operating in that market. The business opportunity is, for example, an order coming in from a party to which the market supplies products. The opportunity is translated into a concrete, high-level business goal first. Then, the IVE goes through four phases:

1. The high-level business goal is decomposed into operational business goals. This decomposition is based on generally accepted knowledge about the domain in which the market is operating. In an industrial construction market, for example, this knowledge can be contained in an extended bill-of-materials.
2. For each identified operational business goal, a collaboration partner is identified in the market that can fulfill this operational business goal (possibly the organization itself that initiated the IVE creation). The identification is based on the capabilities of potential partners, but also on location and QoS attributes.
3. The external-level projections of the local business processes of the selected partners are retrieved. These projections are abstractions from the actual processes in these organizations, such that sen-

sitive or irrelevant details are hidden (Grefen et al., 2003). The local business processes are next composed into a global business process by weaving inter-organizational control flows between them (as illustrated in Figure 7).

4. The composed global business process is mapped onto the distributed infrastructure of the IVE and enacted (executed) there. One of the partners in the IVE will perform the task of the global process coordinator. This may be the initiator of the IVE, depending on available local infrastructure. All partners in the IVE contribute by executing their respective local business processes.

The four phases are the elements in the IVE life cycle (as shown in Figure 8). The life cycle is not strictly linear, however, as problems may be encountered in each phase. For example, it may not be possible to form an acceptable team in the team formation phase on the basis of the specified goal decomposition. Likewise, it may not be possible to construct an acceptable global business process based on the local business processes of selected team members. In these cases, it is necessary to revert to the previous phase in the life cycle and redo the work there. This is illustrated by the dashed backward arrows in Figure 8.

The CrossWork System

The architecture of the CrossWork system is closely related to the IVE lifecycle structure discussed above: each of the four phases of the life cycle is explicitly supported by modules in the architecture. The high-level architecture is shown in Figure 9. In this figure, each of the vertical columns coincides with a phase in the life cycle.

The first three columns of the architecture together form the front-end of the CrossWork architecture, aimed at support for IVE construction (one could say that this is the IVE build-time environment). Each of the three modules relies on knowledge used for automated reasoning, as depicted by the three knowledge bases coupled to the modules. As complete automation is not feasible for arbitrarily complex situations, each module is linked to a user interface to communicate with business process engineers.

The fourth and right-most column forms the back-end of the CrossWork architecture, aimed at support for IVE operation, i.e., enactment (execution) of global business processes. The back-end is comprised of three main modules. The Global Enactment module is responsible for process management at the IVE level, i.e., for the inter-organizational process synchronization. It is equipped with a user interface for global process monitoring. The Local Enactment module is responsible for the enactment of local processes within the boundaries of a single IVE member. This means that there are multiple copies of the Local Enactment module in an IVE (as suggested by the figure). The Local Enactment module drives the user interfaces of the human workers in an IVE member (i.e., is a server to workflow clients). Finally,

Figure 8. IVE lifecycle phases

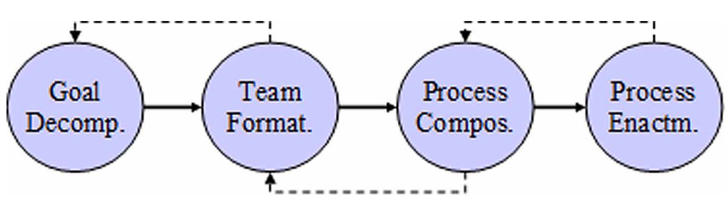
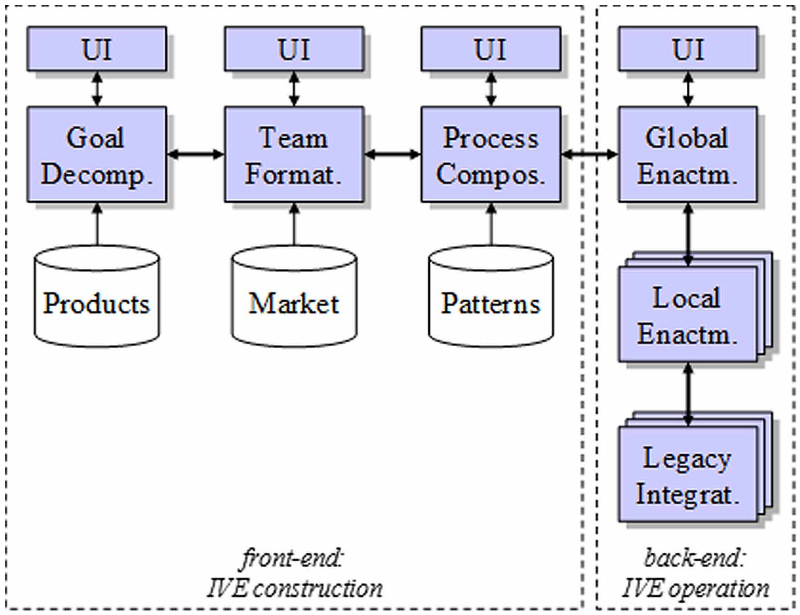


Figure 9. CrossWork architecture



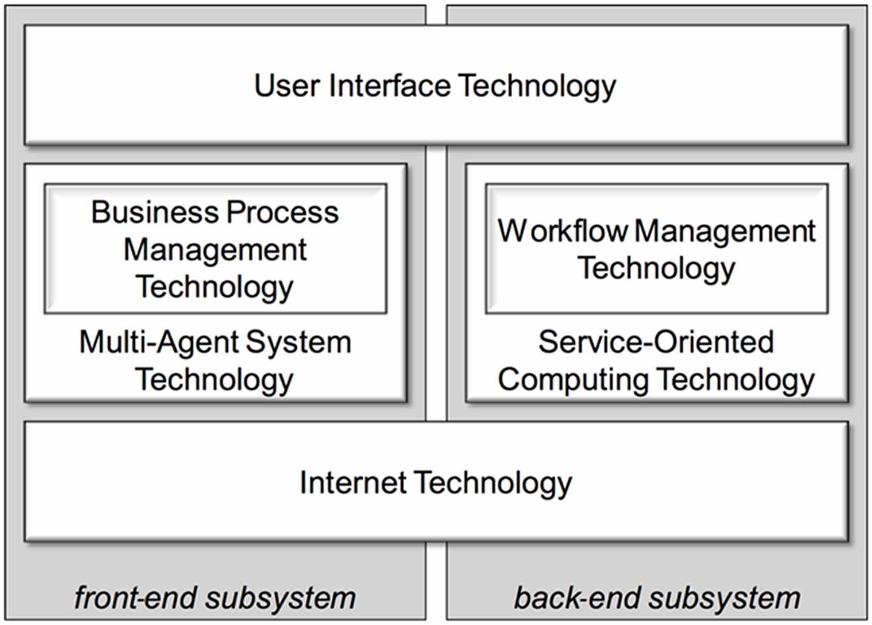
the Legacy Integration module is aimed at providing interfaces to legacy systems run by individual IVE members. Again, this implies that there are multiple copies within an IVE. Typically, one Local Enactment module is linked to one Legacy Integration module.

Note that the interfaces between all architecture components are bi-directional. Bi-directional interfaces in the 'line' from the Goal Decomposition to the Global Enactment modules are chosen to support the life cycle back-tracking as discussed before: if a module cannot fulfill its task, it calls back to the previous module and request a rework. The bi-directional interfaces in the right-most column are needed for synchronization during process enactment.

Obviously, automated support for an IVE must be highly distributed, for the simple reason that the IVE itself consists of possibly many distributed, autonomous parties with local systems that must be integrated into the global process enactment infrastructure. Therefore, the CrossWork system has a distributed nature as well. The bottom-level communication infrastructure layer for the CrossWork system is internet-based. Using internet standards like HTTP as a basic infrastructure allows free choice of higher layers to implement the CrossWork application functionality. This is shown at the bottom of the CrossWork technology stack in Figure 10. Note that this figure distinguishes between the terms business process management technology and workflow management technology. In the context of the CrossWork project, the former term is used for business process design and prototyping functionality, and the latter term for operational process execution functionality. In modern terminology, all is part of business process management technology, making a distinction between design-time support and run-time support.

When we look at the application level of CrossWork, we see different requirements for front-end and back-end platforms. On the one hand, the front-end has main requirements in the fields of goal-orientation and support for reasoning mechanisms needed to implement the IVE construction algorithms. For this reason, a multi-agent system (MAS) platform has been chosen as a basis for the front-end application layer, more specifically the JADE platform (Bellifemine, Caire, & Greenwood, 2007). In the front-end

Figure 10. CrossWork technology stack (adapted from Grefen et al. (2009b))



platform, BPM technology is used for business process design. This is shown in the left of Figure 10. On the other hand, the back-end has main requirements with respect to portability and interoperability to support IVE process enactment in a distributed, heterogeneous environment as dictated by the existing infrastructure at IVE members. To comply with contemporary system integration practice, the back-end application layer is based on the service-oriented computing (SOC) paradigm, employing technology from the Web service stack (Alonso, Casati, Kuno, & Machiraju, 2004), (Daniel & Pernici, 2006). The back-end platform supports SOC-based business process orchestration technology for the enactment of global business processes, as well as 'traditional' workflow management technology for the enactment of local business processes. This is shown in the right hand side of Figure 10. Note that both technologies are referred to as 'workflow management technology' in this figure, conforming to CrossWork terminology as explained before.

Global process orchestration is performed by the Global Enactment module (see Figure 8). The basis for this module is a standard BPEL engine – we use ActiveBPEL (ActiveBPEL, 2007) in our prototype system. We use a paradigm bridge module to 'decouple' the CrossWork front-end and back-end subsystems. This means that we can use dedicated process manipulation technology in the front-end modules and standard, off-the-shelve process enactment technology in the CrossWork back-end modules.

CrossWork in Retrospective

The CrossWork approach and prototype system have been applied in a prototyping scenario in the automotive industry domain (Grefen, Mehandjiev, et al., 2009). The CrossWork technology allows the formation of IVEs in a much shorter time span than the manual approach that is common practice in the domain. The technology also allows the effective handling of more complex business networks and

IVEs – by automating domain knowledge, by semi-automatic generation of possible IVE scenarios, and by support for validation of scenarios.

Although CrossWork is aimed at a more general inter-organizational business process topology than CrossFlow, it is also more limited in two ways. Firstly, there are no explicit contracts included in the approach that underpin the existence of an IVE - in other words, business interests are not managed explicitly in the approach. Secondly, there is no extensible set of cooperation support service modules – CrossWork focuses on 'core' BPM. It would, however, be possible to infuse these additional CrossFlow ingredients into the CrossWork approach.

The CrossWork architecture is designed such that smaller partners in an IVE can make use of the infrastructure of larger partners (e.g., by having web-based clients connect to remote servers). This design was made before the current great popularity of cloud-based solutions. Nowadays, placing BPMS functionality in the cloud would be a good alternative to avoid technology dependency between partners in an IVE.

The CrossWork project is aimed at synchronizing the execution of high-level local manufacturing business processes in global orchestrations. It does not pay attention to the more 'physical', low-level manufacturing processes that are refinements of the high-level local processes. These can, however, be linked to the CrossWork processes, creating lower aggregation levels of processes. These lower aggregation processes are for example addressed in the HORSE project (Grefen, Vanderfeesten, & Boultadakis, 2016), which develops support for detailed manufacturing processes.

Though the CrossWork system uses standard technology from the multi-agent systems (MAS) and service-oriented technology (SOC) technology domains, the use of standard technology was not the main starting point of CrossWork - technologies were selected on the basis of project requirements. We will see in the next section how the XTC project explicitly takes a standard technology paradigm (i.e., SOA) as the basis for inter-organizational business process support.

XTC: TRANSACTIONAL PROCESS SERVICE INTEGRATION

In the previous two sections, we have seen two approaches that are firmly rooted in BPM technology: both CrossFlow and CrossWork rely on underlying business process (workflow) management modules to realize their enactment functionality and use other technology classes to cover the entire spectrum of required functionality. The XTC project takes another approach (T. Wang, 2011): here the service-oriented paradigm (Erl, 2016) is a starting point taken to address the issue of reliable, inter-organizational BPM. Reliability is interpreted in terms of explicit contracting of sub-processes and explicit treatment of transactionality aspects of these sub-processes.

Below, we first discuss the main ingredients of the XTC approach to BPM. Then, we turn to the XTC architecture. As in the previous two sections, we end the section with a retrospective on the approach.

The XTC Approach to BPM

The XTC approach to inter-organizational BPM relies on three main choices: the explicit treatment of process reliability, the combination of a service-oriented point of view with a process-oriented point of view, and a modular, service-oriented approach to process composition. We discuss these three main choices below.

Explicit Treatment of Process Reliability

To allow business organization to rely on the automatic management and execution of their business processes, it is necessary that the execution of these processes is performed in a reliable way. To achieve reliability, the support of business-level transaction management functionality is indispensable. Transaction management ensures reliability and robustness in the execution of business processes, both for intra-organizational processes (Grefen et al., 1999) and inter-organizational processes (Grefen & Vonk, 2006; Vonk & Grefen, 2003).

In a service-oriented context, business services are the point of cooperation between organizations. To obtain reliable business services, transactional support for them must be specified in business terminology that is understood by all involved parties. This means that transaction support is specified at the same level (same business terminology) as the service specification itself. Because of explicit business agreements between the involved parties, the transactional agreements should also be included in the electronic contract established between the parties. The agreed transactional semantics is related to a specific service and is therefore specified in the service level agreement (SLA) that is part of the contract and prescribes the QoS of the aspect under consideration (Transactional Quality of Service in this case, denoted by TxQoS).

The TxQoS specified in an SLA concerns high-level business transaction semantics. In XTC, the following transactional properties have been chosen at this level (T. Wang, Vonk, & Grefen, 2007):

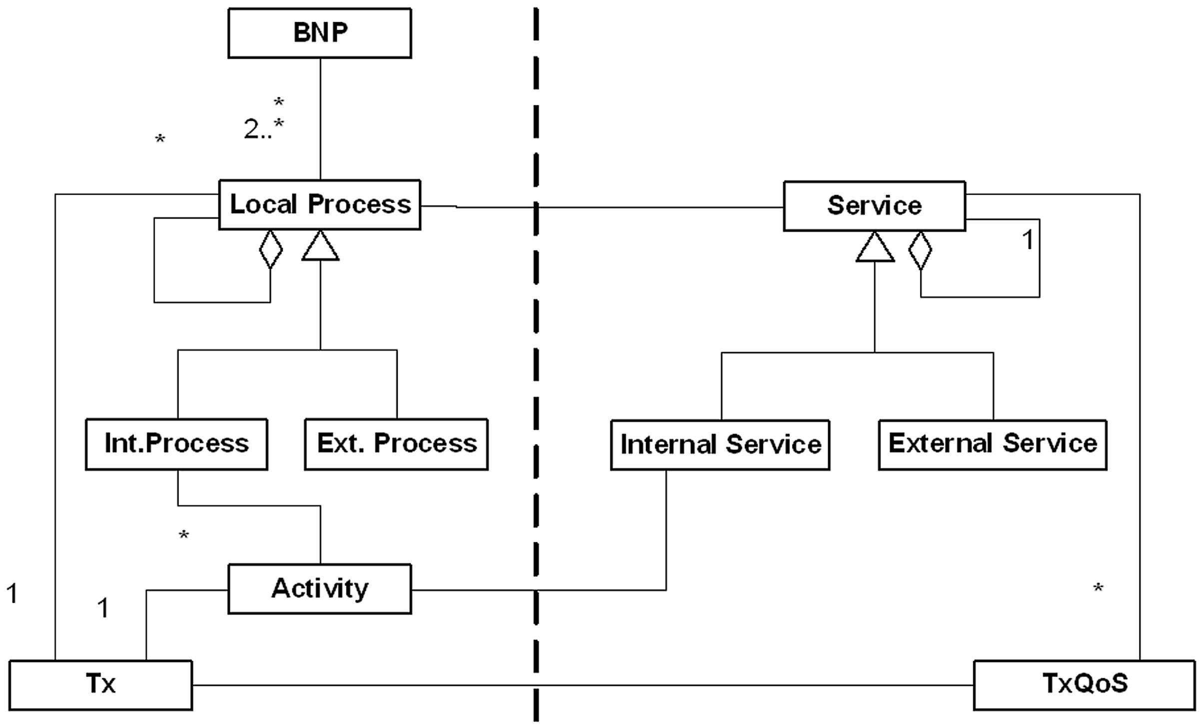
- **Fluency:** The amount of process interruptions that is allowed;
- **Interference:** The possibilities to influence the process execution;
- **Alternation:** The possibilities to use alternative execution paths;
- **Transparency:** The level of visibility of process status details.

These properties have been abbreviated as the FIAT properties. The actual implementation of the transaction support is again an internal matter for an organization. However, a two-way relation exists between the TxQoS offered and the internal underlying systems. First, specification refinement means that the high-level business TxQoS specifications have to be mapped to the low-level technical TxQoS specifications. Second, transaction dependency determines the possible high-level TxQoS, based on the low-level transaction support of the existing systems.

Dual View on Processes and Services

When we relate the process view with the service view, it becomes clear that these views are ‘two sides of the same coin’, i.e., they both model the same real world entity but from a different point of view. So, we have a dual view on the concepts that represent the work (processes and services) that is carried out by organizations (Vonk, Wang, & Grefen, 2007). This dual view is illustrated in Figure 11. Processes specify what has to be done (and in what order), while services are a way to implement (part of) processes. In Figure 11, the thick dashed line illustrates the duality in the model. On the left-hand side is the process view and on the right-hand side the service view. The relation between both is described as follows. Local processes can be implemented as internal or external services, while activities can only

Figure 11. Dual view on processes and services



be implemented as internal services. Because activities are not visible on the external process level, they are inherently internal (Grefen et al., 2003). In the figure, TxQoS corresponds to the high-level Business TxQoS specifications, while Tx corresponds to both the low-level technical TxQoS specifications and (transactional) systems and applications.

Modular, Service-Oriented Business Process Composition

In the XTC approach, processes are composed from sub-processes. Each sub-process is coupled to a service module that describes its transactional behavior. Such a service module is based on an abstract transactional construct (ATC) that specifies an abstract transactional behavior based on an abstracted transaction model (T. Wang, Grefen, & Vonk, 2006). In the XTC project, a library has been developed that contains a taxonomy of ATCs. An ATC is parameterized based on the specifics of a sub-process.

Multiple parameterized ATCs are composed into a composed business transaction (CBT), which describes the transactional behavior of a composed process. Such a process can be a complex sub-process or a complete business process.

The way ATCs can be designed, parameterized and composed into CBTs is governed by the business transaction framework (BTF), a conceptual framework that describes the manipulation of ATCs. A research ingredient of the BTF is a combined algebra and logic (called XTraCalm) that formally specifies the manipulation of ATCs.

The XTC Architecture

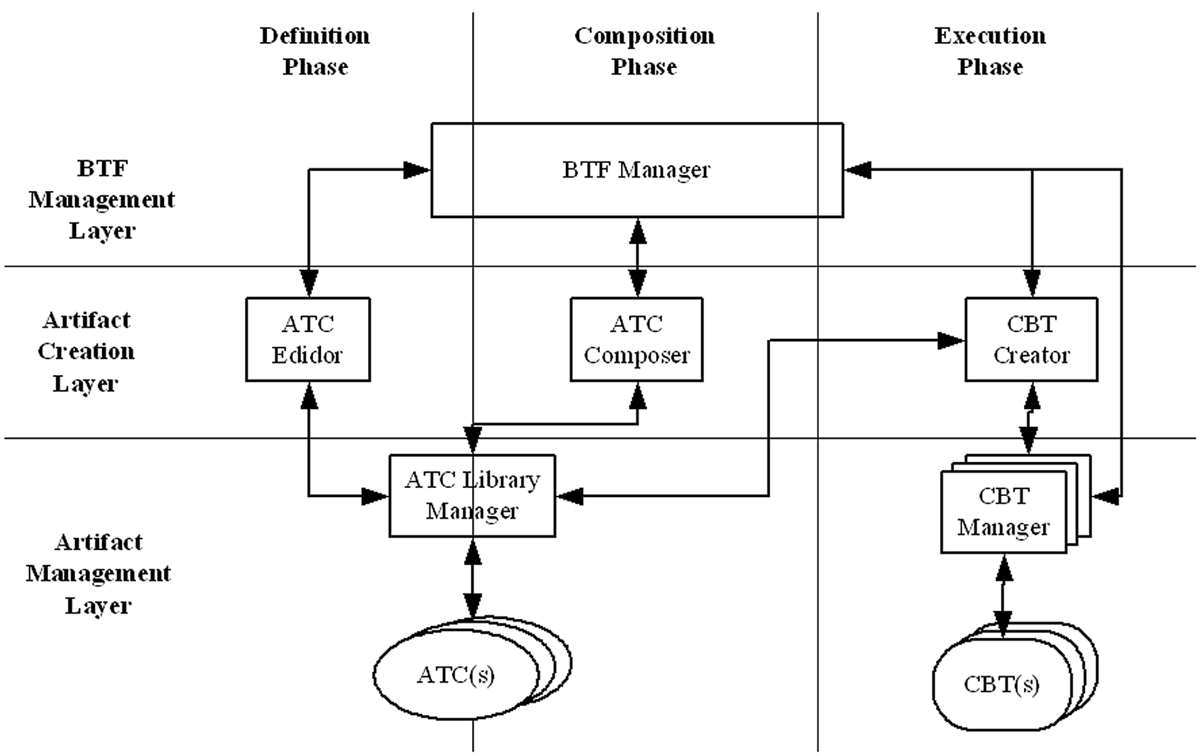
The XTC architecture is based on three phases in the ATC/CBT life cycle (definition, composition an execution) and three levels that distinguish BTF management, ATC/CBT creation, and ATC/CBT management. The XTC architecture can hence be depicted in a 3x3 grid, as shown in Figure 12.

Among all the components of the XTC architecture, the `BTF Manager' is the coordinator which coordinates and controls the activities of other modules. It communicates with the underlying systems, like DBMS, WFMS, etc., through the IT infrastructure such as an Enterprise Service Bus (ESB). Also, it works with other heterogeneous organizations using the open communication standards like SOAP or HTTP. We specify three phases along the BTF life cycle. During the definition phase, the ATC templates are designed based on the classic and widely-adopted transaction models. After the design, one can easily make use of these constructs to build a transaction scheme for a complex process in the composition phase. Also, it is flexible to adjust the transaction scheme to accommodate the changes that often take place in a dynamic business context. Instantiated from the transaction scheme composed in the previous phase, concrete business transactions are executed during the execution phase.

XTC in Retrospective

In the XTC project, healthcare was initially chosen as the application domain for prototyping the approach. The healthcare domain has very complex processes in which many autonomous parties need to

Figure 12. XTC architecture



collaborate. Obviously, process reliability is of major importance in this domain. In an elaborate case study, a complex medical process in a hospital is object of analysis (Vonk, Wang, Aarle, Brugmans, & Grefen, 2008). Based on required reliability characteristics of this process, explicit contractual and transactional elements are infused into the process. In later work, the applicability to the telecom domain is shown (T. Wang, 2011).

Like CrossFlow, XTC pays explicit attention to contracting and transactions. Unlike CrossFlow, XTC places both topics in an open context. CrossFlow does consider extensibility through the use of contract clauses and cooperation support modules, but within a dedicated language and technology context.

XTC focuses heavily on the service-oriented integration of business process support with an emphasis on transactional aspects, as well as a conceptual framework around this support (centered on the BTF discussed before). Consequently, less emphasis has been put on details with respect to interfaces towards specific workflow management technology for the enactment of sub-processes encapsulated in ATCs (where CrossFlow and CrossWork did explicitly include this aspect).

In the next section, we move to the final project in the line-up of research efforts discussed in this paper: the CoProFind project.

COPROFIND: SUPPORT FOR SERVICE-DOMINANT BUSINESS NETWORKS

In the previous sections of this paper, we have discussed a number of research efforts in which the development of technology for BPM has a central role – even when the starting point of a research effort is in business terms (like the concept of instant virtual enterprise in the CrossWork project). In this section, we discuss a research project in which the development of business concepts is central and the existence of appropriate technology components is assumed (such that support can be composed).

CoProFind is a research project that is collaboratively executed by academia and industry in the business-to-business service domain. The project investigates the design and implementation of service-dominant business models (Luftenegger, Comuzzi, & Grefen, 2013; Lusch, Vargo, & O'Brien, 2007) in an agile business context and applies this in innovative business engineering in the financial services industry. The financial services industry is an industry domain with a high potential for business model innovation (Luftenegger, Angelov, Linden, & Grefen, 2010).

The main result of the project is the BASE/X (Business Agility through Service Engineering in an eXtended enterprise) framework (Grefen, 2015a; Grefen, Luftenegger, Linden, & Weisleder, 2013; Luftenegger, 2014) that covers the spectrum of the formulation of service-dominant business strategies to the process-based execution of business services. In the two subsections below, we describe the overall approach including BASE/X and the reference architecture developed in the CoProFind project.

The CoProFind Approach

The overall CoProFind approach (specified in the BASE/X framework) is based on a clear separation of concerns in two design dimensions: the dimension from business strategy to business service (which corresponds to a combination of the abstraction and aggregation dimensions according to (Grefen, 2013)) and the dimension of business concept to information system platform (which corresponds with the realization dimension according to (Grefen, 2013)). We discuss these two dimensions below.

The Strategy to Service Design Dimension

The strategy to service design dimension of the BASE/X framework covers high-level business concepts related to long-term business design to lower-level concepts related to day-to-day business execution. Therefore, it is comparable to the strategy-tactics-operations dimension in traditional business frameworks. In this BASE/X dimension, four layers are explicitly distinguished (Grefen, 2015a; Grefen et al., 2013):

- The business strategy layer defines the service-dominant strategy of a business organization, i.e., it defines the identity of that organization in a business market (Luftenegger, Grefen, & Weisleder, 2012).
- The business model layer defines the business offerings that a business organization makes in a market, i.e., it defines the way the organization makes money.
- The service composition layer defines how a business model is operationalized (implemented) as a composition of business services.
- The business service layer defines the elementary business services of an organization, i.e., it defines the market-oriented business capabilities an organization has to fulfill its strategy.

The BASE/X approach explicitly relates the four above layers by showing the relation between concepts in the layers. It also shows how one can navigate from one layer to the next to perform business engineering in an engineering way. Part of the BASE/X business concept model is shown in Figure 13. In this figure, we can see that one business organization has one business strategy, that one strategy corresponds to multiple business models, and that each business model has a unique service composition. Service compositions and business services have a many-to-many relationship: each service composition uses a number of services and each service can be used by a number of service compositions (hence facilitating re-use of services).

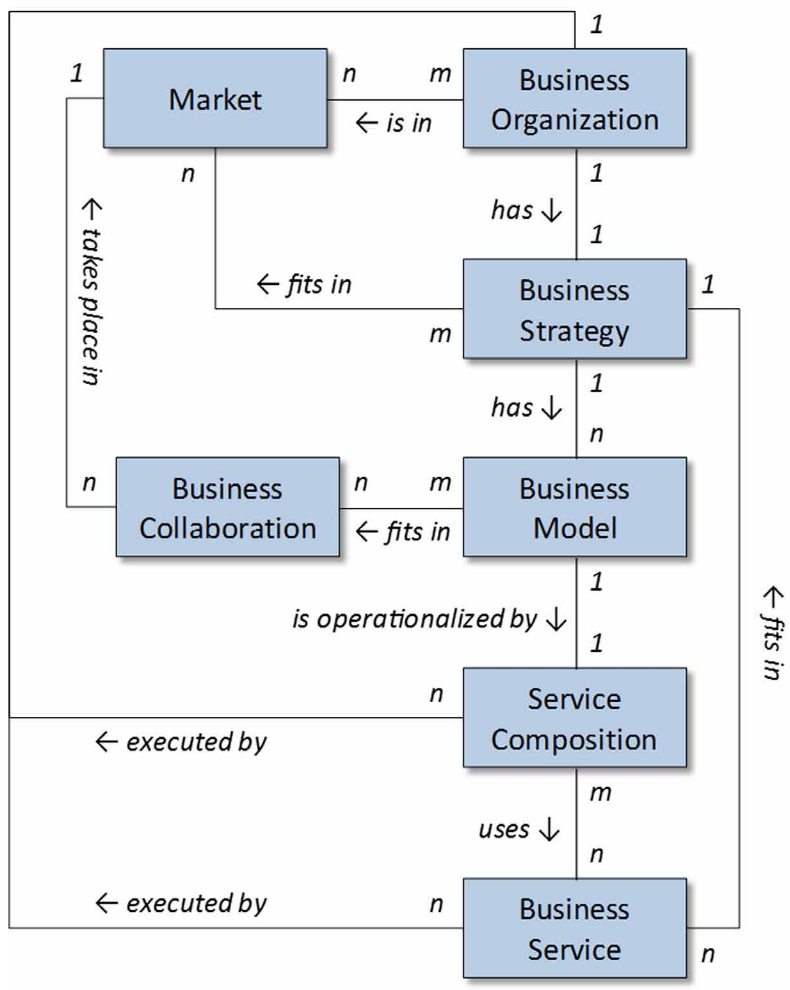
The figure also shows that the CoProFind project explicitly models the business context of service management (and process management, as we will see below): concepts like business organization, market and business collaboration are first order citizens in the BASE/X concept model (see Figure 13).

From the BPM point of view, the service composition layer is the most interesting of the four BASE/X layers. A service composition is the operationalization (the 'how') of a business model (the 'what') in terms of a business process. In the BASE/X approach, two pure forms (which can be mixed in practice) of service composition and hence BPM are distinguished:

- Explicit process management (based on process-oriented service composition in BASE/X terms), in which the responsibility for the process execution is with the business provider.
- Implicit process management (based on mash-up-oriented service composition in BASE/X terms), in which the responsibility for the process execution is with the business consumer.

Both forms of BPM are reflected in the BASE/X reference architecture, which we discuss in the rest of this paper.

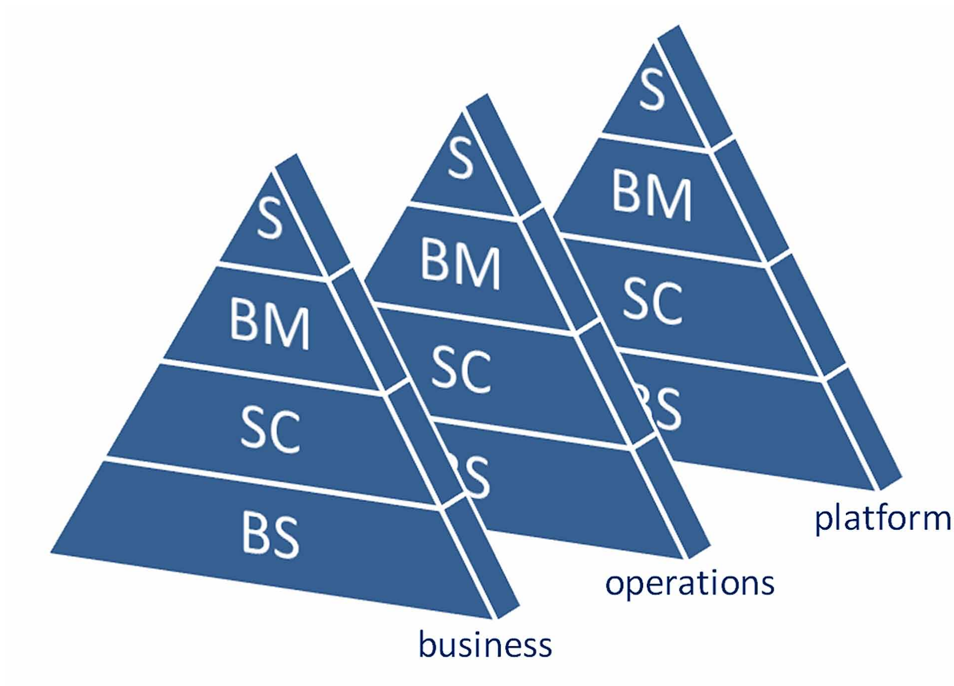
Figure 13. Partial BASE/X concept model



The Concept to Platform Dimension

The second main design dimension in the BASE/X framework (Grefen, 2015a) distinguishes between a business pyramid, an operations pyramid, and a platform pyramid (as shown in Figure 14). All three pyramids have the same four layers of the strategy to service dimension described above. The business pyramid defines service-dominant business concepts purely from a business point of view, i.e., without paying attention to implementation. The operations pyramid defines the implementation of the business concepts in the first pyramid in terms of operational business information system applications (typically automated, but possibly partly manual). The platform pyramid defines the organization of the application-independent technology platforms, such as middleware, database management systems, and BPMSs. These platforms are the basis for the realization of the applications in the second pyramid.

Figure 14. Three pyramids in BASE/X framework



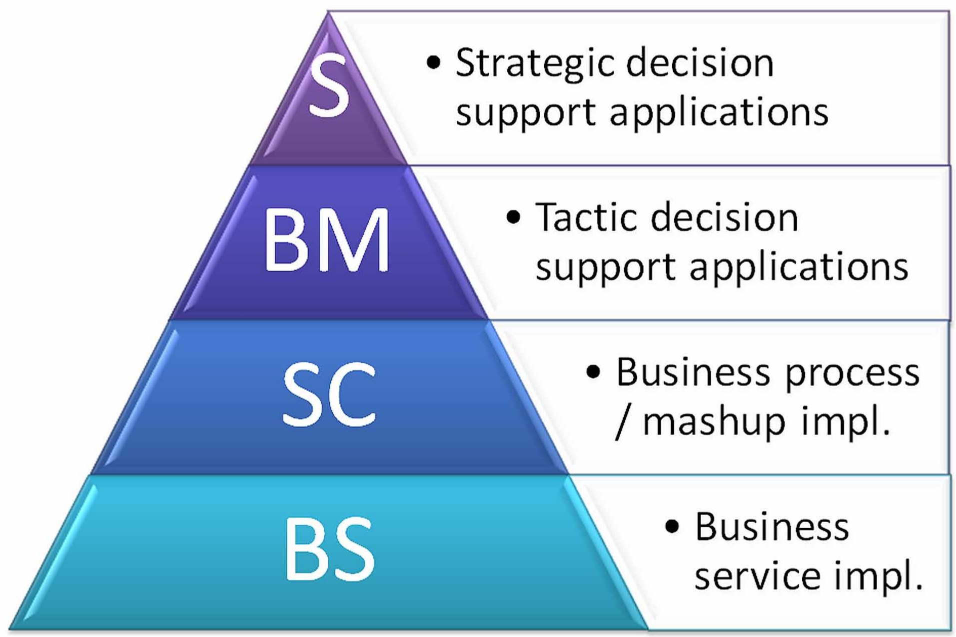
Given this explicit mapping of business structures to IT structures from the strategic to the operational levels (as in the strategy to service dimension discussed before), the CoProFind approach can be considered an engineering approach to business-IT alignment (as discussed by Henderson and Venkatraman (1993)) in a service-dominant context.

An illustration of contents of the four layers of the operations pyramid is shown in Figure 15. In this figure, we see the four layers explained before: business strategy (S), business models (BM), service compositions (SC) and business services (BS).

In the BASE/X operations pyramid, the SC layer is the most relevant of the four BASE/X layers to BPM - as in the business pyramid. This layer contains SC implementations: the specifications that can be executed on the platforms specified in the SC layer of the platform pyramid (see the next subsection).

The SC implementations can have two forms to match the two forms in the business pyramid (as we have discussed before). The first form is that of explicit inter-organizational business process specifications. These specifications are typically expressed in languages like BPEL (Alonso et al., 2004) or BPMN (Freund & Rücker, 2014). In Figure 16, we show an example business process implementing a SC in the travel industry domain (Grefen, 2015b) in informal notation. In this figure, the vertical swim lanes indicate the different organizations involved in the business process. In this explicit form, process instance states are managed by the service provider - the service consumer follows the flow of process. This implies that the provider explicitly manages all process state variables. The second form is that of implicit business process specifications in the form of service mash-up specifications (Carlson, Ngu, Podorozhny, & Zeng, 2008) In this form, process instance states are managed by the service consumer - here, the service provider follows the flow of the process. To support both forms, process management engines are used in the SC layer of the platform pyramid, which we explain below.

Figure 15. Application layers in the BASE/X operations pyramid



The CoProFind Reference Architecture

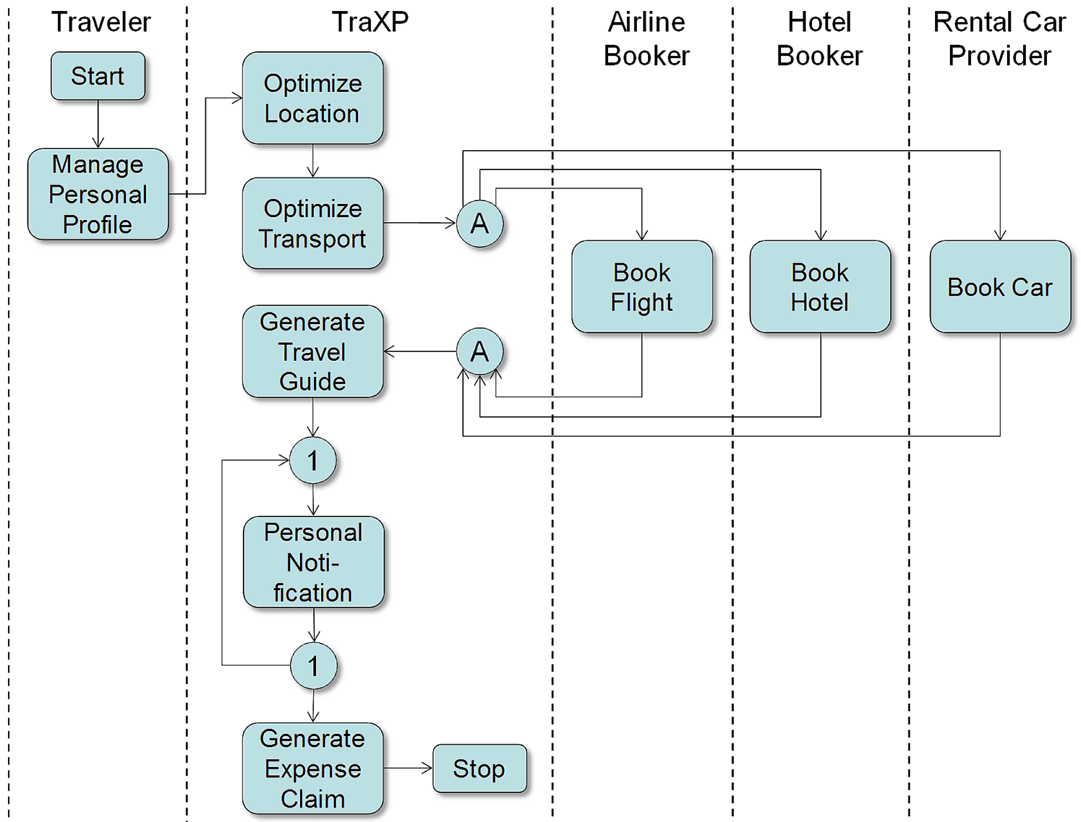
The CoProFind reference architecture is shown in Figure 17. As a reference architecture, it is the basis for the design of concrete architectures in specific application scenarios. This reference architecture is a standard elaboration of the platform pyramid explained before, again using the four layers from business strategy to business service (as shown in the figure by the large horizontal boxes). Apart from the four layers, the architecture includes middleware in the form of enterprise service bus (ESB) components. As the middleware has no business semantics, it is not part of any of the four layers.

In the SC layer of the reference architecture, we see the BPMS and mash-up platform (Mash) components, which are responsible for explicit respectively implicit BPM as discussed before. Both components store their data through a database management system (DBMS) in an operational data store (ODS).

The BPMS and Mash components activate the business services in the BS layer through the ESB, which are organized in a managed service platform (MSP). Services store their process-relevant data in the same ODS as the BPMS and Mash components to ensure easy interoperability. Business services can be internal to a business organization, or they can be external (as shown in the bottom-right-hand corner of the figure).

The strategy and business model layers contain components for the re(design) of business strategies and business models. This re(design) is based on data in strategic respectively tactical data warehouses (S-DW and T-DW), which are filled using data extraction and transformation logic (ETL) modules. For more details on these layers, the reader is referred to Grefen et al. (2013).

Figure 16. Example inter-organizational business process for service composition (Grefen, 2015b)



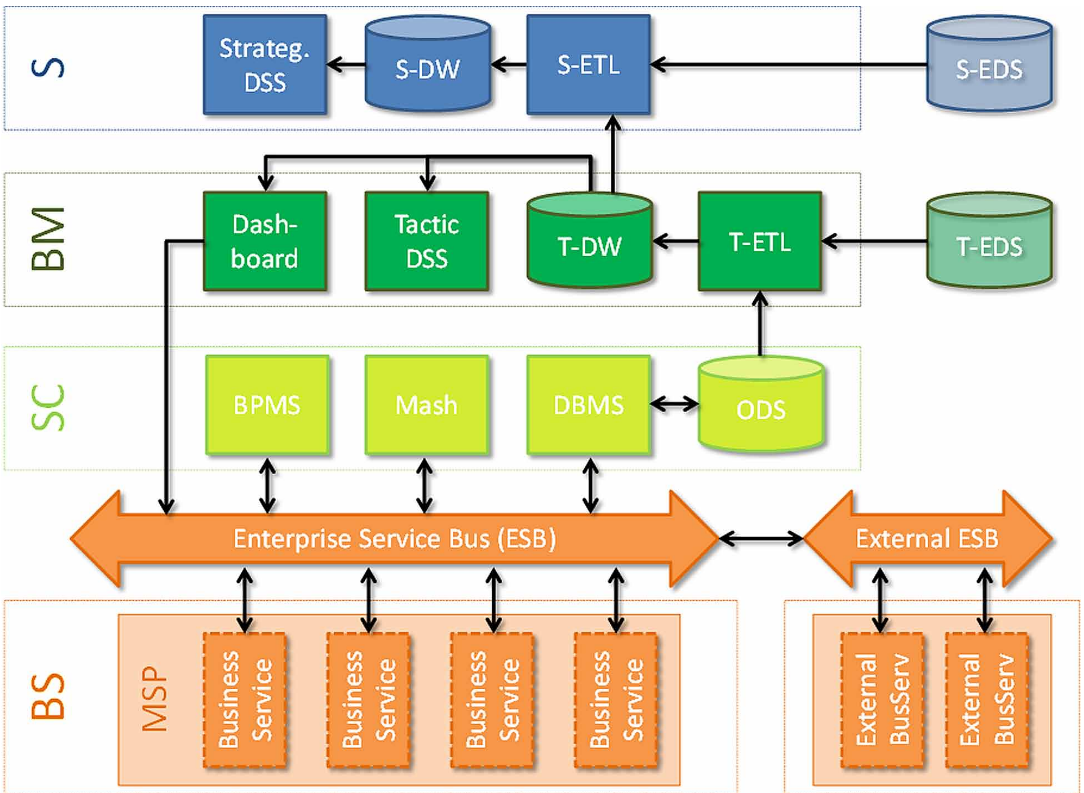
CoProFind in Retrospective

As discussed in the introduction of this section, the CoProFind project takes a highly business-oriented starting point to BPM: business processes are derived from business models. Consequently, the emphasis is much more on concept, framework and architecture development (as described above) than on the development of new BPM technology. Actually, the reference architecture presented above assumes the existence of the necessary platform modules: the innovation is not in new information technology, but in the way it is combined and used to realize advanced BPM for the execution of complex services in agile business environments.

In doing so, the scope of the design also shifts from BPM-centric (which is, for example, certainly the case in the CrossFlow project described in an earlier section of this paper) to enterprise-centric: BPM is treated as an important component in a landscape consisting of a multitude of other important components (as shown in the reference architecture in Figure 17). As such, BPM gets a pivotal role in a broader, service-centric approach facilitating the alignment of business and IT in dynamic scenarios.

To achieve high levels of agility in service-dominant business, proper automated support is essential for transforming business models and business service catalogs into business processes. The development of such support is currently a research theme in the further development of BASE/X.

Figure 17. BASE/X reference architecture



TRENDS, ANALYSIS, AND OUTLOOK

In the previous sections, we have presented the development of BPM in business networks from the viewpoint of a number of projects. In doing so, our aim is not to be complete, but rather to discuss and relate a number of research efforts from the experience of the authors. Consequently, there are many more research projects that could have been discussed in this work.

In this section, we conclude with a discussion of important trends in inter-organizational BPM, a general analysis and an outlook towards new developments.

Trends

From the projects discussed in this paper, we can observe four important trends for the domain of inter-organizational business processes. These trends are summarized in Table 1 and explained in detail below.

The first important trend shown in the paper is a development from static inter-organizational business processes (like in the WISE project) to dynamic inter-organizational processes. All four projects that we have discussed in detail (CrossFlow, CrossWork, XTC and CoProFind) have explicit support for dynamism in the process, i.e., they include facilities to alter collaboration patterns between partners in a market. In all these projects, we see the infusion of electronic business aspects into BPM. The level of dynamism (in terms of the frequency of changes) differs between the projects, however: where CrossFlow

Table 1. Observed trends

Aspect	From	To
Process dynamism	Static	Dynamic
Network topology	Simple	Complex
Information technology	Dedicated	Standard
Development nature	Technology push	Requirements pull

supports dynamism on an operational (client order) basis, the other three projects support dynamism at a more tactical level. In other words, CrossFlow is explicitly geared towards mass-customization of business process instances, whereas the other projects are not. Consequently, the electronic business character is most explicitly visible in CrossFlow (as noted before in this paper).

Secondly, there is a trend in the main discussed projects from simple business network topologies to more complex network topologies. The CrossFlow project focuses on bilateral collaborations between a service consumer and a service provider. XTC allows collaboration between multiple parties, be it in a block-structured context. CrossWork and CoProFind allow inter-organizational processes with arbitrary topologies.

Thirdly, we also have illustrated a move from the use of dedicated information technology (for example in WISE and CrossFlow) via the partial use of standardized technology (like MAS and SOC technology in CrossWork) to the use of standardized technology as a starting point (SOC technology in XTC and CoProFind). Clearly, the use of standardized technology allows the reuse of platforms and enhances the interoperability potential between systems. On the other hand, the use of standards is also dictated to deal with the complexity of advanced inter-organizational BPM: not all can be designed from scratch.

Finally, we have shown how in a period of about roughly two decades of BPM research, the emphasis in the research projects (certainly the ones described in this paper, but probably more generally) has moved from technology push aspects (the development of core BPM technology from a technological starting point) to requirements pull aspects (the development of business-based frameworks for the application of core BPM technology). In other words, the described efforts in the BPM research domain have moved from (stated slightly in a black-and-white fashion) trying to find applications for new BPM technology to using BPM as a way to align business and IT in a flexible way.

ANALYSIS

Apart from the four trends discussed in the previous subsection, we can try and analyze the functional coverage of the various projects. As discussed before, CrossWork is more general than CrossFlow in the process topology dimension and the explicit treatment of domain knowledge, but also misses some important CrossFlow ingredients like explicit contract management and a set of extensible support modules. XTC does rely on a well-accepted notion of SOA and does place process reliability very central on the basis of explicit contracts, but is less strongly rooted in ‘traditional’ BPM technology than CrossFlow and CrossWork. CoProFind takes business strategies and business models as a starting point with a lesser focus on technical details of inter-organizational business process enactment.

From this analysis, we can conclude that none of the discussed projects has a complete coverage of functionality (and to the knowledge of the authors, the same goes for other projects that are not discussed in this paper). Consequently, an 'ideal' platform for inter-organizational BPM requires a 'blend' of ingredients from several approaches. The inherent complexity of this 'blend' is the reason for the fact that an 'ideal' solution has not yet been realized.

OUTLOOK

From what we have presented, the outlook for the research domain of dynamic inter-organizational BPM is clear: there is still ample room for new developments. There is room for development of new theories, concepts and technologies. But there is also much room for integration of existing partial solutions and consolidation of taken efforts: many research efforts (including the ones described in the previous chapters) have resulted in interesting elements but have to be complemented with other elements to arrive at full-blown solutions for practice. As we have stated before: no project that we know covers all aspects required for full deployment in business practice.

The room for new developments is indeed reflected in recent research efforts towards dynamic inter-organizational BPM. An example research effort in the manufacturing domain is the ADVENTURE project (Schulte, Schuller, Steinmetz, & Abels, 2012). ADVENTURE aims at supporting dynamic business processes for manufacturing highly customized products. Dynamic processes are provided by incrementally filling in process templates and by reacting to changes that occur during process execution. An example research effort in the logistics domain is the GET Service project (Baumgrass et al., 2016; Cabanillas, Baumgrass, Mendling, Rogetzer, & Bellovoda, 2014). In GET Service, inter-organizational business processes are dynamically created by composing process snippets, each of which is a template for the process part executed by a specific party. Applying dynamic BPM in logistics business networks can open ways to new forms of collaboration and new business opportunities (Brouns, 2016). A new business domain where inter-organizational BPM is considered is traffic management. Traditionally this is a domain that is asset- or resource-oriented, but service-orientation brings a value-centered view (Grefen et al., 2015; Traganos, Grefen, Hollander, Turetken, & Eshuis, 2015).

To achieve customer values in service-oriented business, service compositions can be operationalized as 'traditional' inter-organizational business processes - as we have seen in this paper so far. An alternative possible way to support this operationalization is subject-oriented BPM (Fleischmann, Schmidt, Stary, Obermeier, & Börger, 2012; Turetken & Demirors, 2013). This approach puts the actors (subjects) with their actions and their communication in the center of attention. In the context of networked organizations, subjects represent autonomous parties in inter-organizational business processes. A process is established by structuring the actions of each subject (internal process) in terms of loosely coupled business services (van den Hurk, Turetken, & van Moll, 2015; Turetken & Demirors, 2011) and the coordination of the required communication among the subjects in terms of business service invocations. The internal process of each subject is orchestrated, while the coordination across the autonomous parties corresponds to the principle of choreography in SOA management.

ACKNOWLEDGMENT

All colleague researchers and practitioners of the CrossFlow, CrossWork, XTC and CoProFind projects are thanked for their work and collaboration, which have provided valuable input for this paper.

REFERENCES

Active, BPEL. (2007). *ActiveBPEL Development Server User's Guide*. Active Endpoints, Inc.

Alonso, G., Casati, F., Kuno, H., & Machiraju, V. (2004). *Web Services: Concepts, Architectures and Applications*. Springer Berlin Heidelberg. doi:10.1007/978-3-662-10876-5

Alonso, G., Fiedler, U., Hagen, C., Lazcano, A., Schuldt, H., & Weiler, N. (1999). WISE: business to business e-commerce. In *Proceedings of the Int. Workshop on Research Issues on Data Engineering: Information Technology for Virtual Enterprises, RIDE-VE'99* (pp. 132–139). IEEE.

Alonso, G., Hagen, C., Schek, H.-J., & Tresch, M. (1997). Distributed Processing over Stand-alone Systems and Applications. In *VLDB '97* (pp. 575–579).

Baumgrass, A., Botezatu, M., Di Ciccio, C., Dijkman, R., Grefen, P., Hewelt, M., & ... . (2016). Towards a Methodology for the Engineering of Event-Driven Process Applications. In Proceedings of the *Int. Workshops of BPM '15* (pp. 501–514).

Bellifemine, F. L., Caire, G., & Greenwood, D. (2007). *Developing multi-agent systems with JADE*. John Wiley. doi:10.1002/9780470058411

Boukhedouma, S., Alimazighi, Z., & Oussalah, M. (2017). Adaptation and Evolution Frameworks for Service Based Inter-Organizational Workflows. *International Journal of E-Business Research*, *13*(2), 28–57. doi:10.4018/IJEBR.2017040103

Brouns, N. (2016). *The Quantitative Feasibility of BASE/X for the Logistics Market* [BSc. Thesis]. Eindhoven University of Technology.

Cabanillas, C., Baumgrass, A., Mendling, J., Rogetzer, P., & Bellovoda, B. (2014). Towards the Enhancement of Business Process Monitoring for Complex Logistics Chains. In *Proceedings of the Int. Workshops of BPM '15* (pp. 305–317).

Carlson, M. P., Ngu, A. H. H., Podorozhny, R., & Zeng, L. (2008). *Automatic Mash Up of Composite Applications. In Proceedings of the Service-Oriented Computing (ICSOC '08)* (pp. 317–330). Springer Berlin Heidelberg. doi:10.1007/978-3-540-89652-4_25

Cingil, R., Dogac, A., Tatbul, N., & Arpinar, S. (1999). An adaptable workflow system architecture on the Internet for electronic commerce applications. In *Proceedings of the Int. Symposium on Distributed Objects and Applications* (pp. 242–251). IEEE. 10.1109/DOA.1999.794038

Daniel, F., & Pernici, B. (2006). Insights into Web Service Orchestration and Choreography. *International Journal of E-Business Research*, *2*(1), 58–77. doi:10.4018/jebr.2006010104

Dogac, A., Beeri, C., Tumer, A., Ezbiderli, M., Tatbul, N., Icdem, C., & ... . (1999). *MariFlow: A Workflow Management System for the Maritime Industry*. MAREXPO Book.

Erl, T. (2016). *Service-Oriented Architecture: Analysis and Design for Services and Microservices* (2nd ed.). The Prentice Hall.

Eshuis, R., & Grefen, P. (2008). Constructing customized process views. *Data & Knowledge Engineering*, *64*(2), 419–438. doi:10.1016/j.datak.2007.07.003

Fdhila, W., Indiono, C., Rinderle-Ma, S., & Reichert, M. (2015, April). Dealing with change in process choreographies: Design and implementation of propagation algorithms. *Information Systems*, *49*, 1–24. doi:10.1016/j.is.2014.10.004 PMID:25892843

Fleischmann, A., Schmidt, W., Stary, C., Obermeier, S., & Börger, E. (2012). *Subject-Oriented Business Process Management*. Springer Berlin Heidelberg. doi:10.1007/978-3-642-32392-8

Freund, J., & Rücker, B. (2014). *Real-Life BPMN: Using BPMN 2.0 to Analyze, Improve, and Automate Processes in Your Company* (2nd ed.). CreateSpace Independent Publishing Platform.

Gartner. (2010). *Gartner Reveals Five Business Process Management Predictions for 2010 and Beyond* (Press Release).

Germany Trade & Invest. (2014). *Industrie 4.0: Smart Manufacturing for the Future*.

Grefen, P. (2013). Networked Business Process Management. *International Journal of IT/Business Alignment and Governance*, *4*(2), 54–82. doi:10.4018/ijitbag.2013070104

Grefen, P. (2015a). *Service-Dominant Business Engineering with BASE/X: Business Modeling Handbook*. Amazon CreateSpace.

Grefen, P. (2015b). *Beyond e-Business: Towards Networked Structures*. Routledge.

Grefen, P., Aberer, K., Hoffner, Y., & Ludwig, H. (2000). CrossFlow: Cross-organizational workflow management in dynamic virtual enterprises. *International Journal of Computer Systems Science & Engineering*, *15*(5), 277–290.

Grefen, P., Eshuis, R., Mehandjiev, N., Kouvas, G., & Weichhart, G. (2009). Internet-Based Support for Process-Oriented Instant Virtual Enterprises. *IEEE Internet Computing*, *13*(6), 65–73. doi:10.1109/MIC.2009.96

Grefen, P., Ludwig, H., & Angelov, S. (2003). A Three-Level Framework for Process and Data Management of Complex E-Services. *International Journal of Cooperative Information Systems*, *12*(4), 487–531. doi:10.1142/S0218843003000838

Grefen, P., Luftenegger, E., Linden, E. d., & Weisleder, C. (2013). *BASE/X: Business Agility through Cross-Organizational Service Engineering - The Business and Service Design Approach developed in the CoProFind Project* (Beta Working Papers 414). Eindhoven University of Technology.

Grefen, P., Mehandjiev, N., Kouvas, G., Weichhart, G., & Eshuis, R. (2009). Dynamic business network process management in instant virtual enterprises. *Computers in Industry*, *60*(2), 86–103. doi:10.1016/j.compind.2008.06.006

Grefen, P., Pernici, B., Sanchez, G., Vonk, J., & Boertjes, E. (1999). *Database Support for Workflow Management: the WIDE Project*. Springer, US. doi:10.1007/978-1-4615-5171-3

Grefen, P., Rinderle-Ma, S., Dustdar, S., Fdhila, W., Mendling, J., & Schulte, S. (2017). Charting Process-Based Collaboration Support in Agile Business Networks. *IEEE Internet Computing*, 1. doi:10.1109/MIC.2017.265102547

Grefen, P., Turetken, O., Traganos, K., Hollander, A. D., Eshuis, R., & den Hollander, A. (2015). Creating agility in traffic management by collaborative service-dominant business engineering. In *IFIP Advances in Information and Communication Technology (PROVE-2015)* (Vol. 463, pp. 100–109).

Grefen, P., Vanderfeesten, I., & Boultadakis, G. (2016). Supporting Hybrid Manufacturing: Bringing Process and Human/Robot Control to the Cloud (Short Paper). In *Proceedings of the IEEE International Conference on Cloud Networking (Cloudnet '15)* (pp. 200–203). IEEE. 10.1109/CloudNet.2016.39

Grefen, P., & Vonk, J. (2006). A Taxonomy of Transactional Workflow Support. *International Journal of Cooperative Information Systems*, *15*(1), 87–118. doi:10.1142/S021884300600130X

Griffel, F., Boger, M., Weinreich, H., Lamersdorf, W., & Merz, M. (1998). Electronic contracting with COSMOS-how to establish, negotiate and execute electronic contracts on the Internet. In Proceedings of the Int. Enterprise Distributed Object Computing (EDOC'98) (pp. 46–55). IEEE.

Hagen, C. (1999). *A Generic Kernel for Reliable Process Support* [Ph.D. Dissertation]. Swiss Federal Institute of Technology Zürich.

Henderson, J. C., & Venkatraman, H. (1993). Strategic alignment: Leveraging information technology for transforming organizations. *IBM Systems Journal*, *32*(1), 472–484. doi:10.1147j.382.0472

Hoffner, Y., Field, S., Grefen, P., & Ludwig, H. (2001). Contract-driven creation and operation of virtual enterprises. *Computer Networks*, *37*(2), 111–136. doi:10.1016/S1389-1286(01)00210-9

Hoffner, Y., Ludwig, H., Gulcu, C., & Grefen, P. (2000). An architecture for cross-organisational business processes. In *Proceedings of the Int. Workshop on Advanced Issues of E-Commerce and Web-Based Information Systems (WECWIS 2000)* (pp. 2–11). IEEE.

Koetsier, M., Grefen, P., & Vonk, J. (2000). Contracts for Cross-Organizational Workflow Management. In *Proceedings of the Int. Conference on Electronic Commerce and Web Technologies (EC-Web 2000)* (pp. 110–121). Springer, Berlin, Heidelberg. 10.1007/3-540-44463-7_10

Lazcano, A., Schuldt, H., Alonso, G., & Schek, H. (2001). WISE: Process Based E-Commerce. *A Quarterly Bulletin of the Computer Society of the IEEE Technical Committee on Data Engineering*, *24*(1), 46–51.

Lienhard, H. (1998). IvyBeans - Bridge to VSH and the project WISE. In Conference of the Swiss Priority Programme Information and Communication Structures, Zürich, Switzerland.

Luftenegger, E. (2014). *Service-Dominant Business Design* [PhD Thesis]. Eindhoven University of Technology.

Luftenegger, E., Angelov, S., van der Linden, E., & Grefen, P. (2010). *The State of the Art of Innovation-Driven Business Models in the Financial Services Industry* (Beta Working Papers; Vol. 310). Eindhoven University of Technology.

Luftenegger, E., Comuzzi, M., & Grefen, P. (2013). *The Service-Dominant Ecosystem: Mapping a Service Dominant Strategy to a Product-Service Ecosystem. IFIP Advances in Information and Communication Technology (PRO-VE 2013)* (Vol. 408). Springer Berlin Heidelberg.

Luftenegger, E., Grefen, P., & Weisleder, C. (2012). *The Service Dominant Strategy Canvas: Towards Networked Business Models. In IFIP Advances in Information and Communication Technology (PRO-VE 2012)* (Vol. 380, pp. 207–215). Springer Berlin Heidelberg.

Lusch, R. F., Vargo, S. L., & OBrien, M. (2007). Competing through service: Insights from service-dominant logic. *Journal of Retailing*, *83*(1), 5–18. doi:10.1016/j.jretai.2006.10.002

Maxton, G., & Wormald, J. (2004). *Time for a Model Change: Re-engineering the Global Automotive Industry*. Cambridge University Press. doi:10.1017/CBO9780511488535

Mehandjiev, N., & Grefen, P. (2010). *Dynamic business process formation for instant virtual enterprises*. Springer. doi:10.1007/978-1-84882-691-5

Schonenberg, H., Mans, R., Russell, N., Mulyar, N., & van der Aalst, W. (2008). Process Flexibility: A Survey of Contemporary Approaches. In *Proceedings of the Int. Workshop CIAO! and Int. Workshop EOMAS* (pp. 16–30). Springer Berlin Heidelberg. 10.1007/978-3-540-68644-6_2

Schulte, S., Schuller, D., Steinmetz, R., & Abels, S. (2012). Plug-and-Play Virtual Factories. *IEEE Internet Computing*, *16*(5), 78–82. doi:10.1109/MIC.2012.114

Sewing, J.-H., Rosemann, M., & Dumas, M. (2006). Process-Oriented Assessment of Web Services. *International Journal of E-Business Research*, *2*(1), 19–44. doi:10.4018/jebr.2006010102

Shan, M. (1999). FlowJet: Internet-Based E-Service Process Management. In *Proceedings of the Int. Process Technology Workshop*.

Shan, M., Davis, J., Du, W., & Huang, Y. (1997). *HP Workflow Research: Past, Present, and Future (Technical Report HPL-97-105). HP Labs.*

Traganos, K., Grefen, P., den Hollander, A., Turetken, O., & Eshuis, R. (2015). *Business Model Prototyping for Intelligent Transport Systems: A Service-Dominant Approach. Beta Publication 469*. Eindhoven University of Technology.

Turetken, O., & Demirors, O. (2011). Plural: A decentralized business process modeling method. *Information & Management*, *48*(6), 235–247. doi:10.1016/j.im.2011.06.001

Turetken, O., & Demirors, O. (2013). Business process modeling Pluralized. In S-BPM ONE 2013, *CCIS* (Vol. 360, pp. 1–9). Springer-Verlag.

van den Hurk, H., Turetken, O., & van Moll, J. (2015). Subject-oriented plural method meets BPMN: A case study. In Proceedings of *S-BPM ONE ‘15* (pp. 1–9).

van der Aalst, W. (2000). Loosely coupled interorganizational workflows: Modeling and analyzing workflows crossing organizational boundaries. *Information & Management*, *37*(2), 67–75. doi:10.1016/S0378-7206(99)00038-5

Vandaele, N. J., & Decouttere, C. J. (2013). The multiple faces of mass customization: Product design, process design and supply chain design. In IFIP Advances in Information and Communication Technology (APMS 2012) (Vol. 397, pp. 270–277). Springer Berlin Heidelberg.

Verdouw, C. N., Beulens, A. J. M., Trienekens, J. H., & Van Der Vorst, J. G. A. J. (2011). A framework for modelling business processes in demand-driven supply chains. *Production Planning and Control*, *22*(4), 365–388. doi:10.1080/09537287.2010.486384

von Corswant, F., & Fredriksson, P. (2002). Sourcing trends in the car industry. *International Journal of Operations & Production Management*, *22*(7), 741–758. doi:10.1108/01443570210433526

Vonk, J., & Grefen, P. (2003). Cross-Organizational Transaction Support for E-Services in Virtual Enterprises. *Distributed and Parallel Databases*, *14*(2), 137–172. doi:10.1023/A:1024884626434

Vonk, J., Wang, T., & Grefen, P. (2007). A Dual View to Facilitate Transactional QoS. In Proceedings of the IEEE Int. Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE 2007) (pp. 381–383). IEEE. doi:10.1109/WETICE.2007.4407193

Vonk, J., Wang, T., van Aarle, J., Brugmans, S., & Grefen, P. (2008). *An Analysis of Contractual and Transactual Aspects of a Cardiothoracic Surgery Process (Beta Technical Report)*. Eindhoven University of Technology.

Wang, M., & Kumar, K. (2014). Developing Flexible Business Process Management Systems Using Modular Computing Technologies. In The Flexible Enterprise (pp. 239–255). Springer India.

Wang, T. (2011). *A TxQoS-aware Business Transaction Framework* [Ph.D. Thesis]. Eindhoven University of Technology.

Wang, T., Grefen, P., & Vonk, J. (2006). Abstract Transaction Construct: Building a Transaction Framework for Contract-Driven, Service-Oriented Business Processes. In *Proceedings of the Int. Conference on Service-Oriented Computing (ICSOC 2006)* (pp. 434–439). Springer, Berlin, Heidelberg. 10.1007/11948148_36

Wang, T., Vonk, J., & Grefen, P. (2007). TxQoS: A Contractual Approach for Transaction Management. In *Proceedings of the IEEE Int. Enterprise Distributed Object Computing Conference (EDOC 2007)* (pp. 327–327). IEEE. 10.1109/EDOC.2007.37

Weber, B., Reichert, M., & Rinderle-Ma, S. (2008). Change patterns and change support features – Enhancing flexibility in process-aware information systems. *Data & Knowledge Engineering*, *66*(3), 438–466. doi:10.1016/j.datak.2008.05.001

This research was previously published in the International Journal of E-Business Research (IJEBR), 13(4); edited by Jeffrey Hsu, pages 70-104, copyright year 2017 by IGI Publishing (an imprint of IGI Global).