An Architecture for Cross-Organisational Business Processes

Yigal Hoffner, Heiko Ludwig, Ceki Gülcü
IBM Research, Zurich Research Laboratory
8803 Rüschlikon, Switzerland
{yho|hlu|cgu}@zurich.ibm.com

Paul Grefen
University of Twente
P.O. Box 217
7500 AE Enschede, The Netherlands
grefen@cs.utwente.nl

Abstract

Efficient means of electronic interaction are an essential requirement for the integration of different companies' business processes along the value chain. Until recently, this interaction relied on expensive, complex and inflexible solutions, mostly based on EDI or some proprietary means. The high set-up costs and time associated with this type of infrastructure prohibits the dynamic forging of business partnerships, which is of utmost importance to the services industry. The CrossFlow architecture supports the dynamic establishment and enactment of a business relationship between two organisations, based on a contract that specifies this relationship. This is achieved by creating an electronic market where advertising and searching for compatible business partners takes place. This is further enhanced by automating the set-up of the contract enactment and supervision infrastructure, and by connecting them together to allow the business processes of the partners to cross their organisational boundaries.

Keywords: B2B e-commerce, cross-organisational processes, virtual markets, workflow management, electronic contracts, contract match-making

1 Introduction

Business relationships between organisations can be established in a dynamic way by integrating an electronic market with the means to specify a business relationship in a contract. The linking of the business processes of the related parties can be further enhanced by automating the set-up of the infrastructure for the enactment, accompanied by an end-to-end supervision of the contract enactment.

In the past decade, the globalisation of markets and the proliferation of the Internet increased global competition in a way that companies have to face a more dynamic environment, requiring them to focus on their core competencies to stay competitive [1]. This, in turn, puts increasing pressure on companies to connect their business processes to the processes of other organisations to provide services to customers. The manufacturing industry as well as banking and the trading sectors responded to that need by setting up infrastructures and standards based on the *Electronic Data Interchange* (EDI) framework such as UN/EDIFACT [2] and SWIFT. However, such infrastructures have disadvantages in terms of the low level of message-type specifications, time-consuming and expensive case-by-case implementation of gateways for new message types, and expensive software and leased lines. Such an infrastructure is beneficial for large organisations in a relatively stable environment, but is unsuitable for smaller companies and industries in which business relationships are established on a case-by-case basis. Companies in a dynamic environment need functionality for the rapid establishment and management of dynamic business relationships with other organisations, to create the type of a *virtual enterprise* we are interested in. A virtual enterprise is an organisation that provides a complex service to a customer but cannot implement the whole complex service itself within its boundaries. It focuses on its core competency, and buys services from other organisations to actually implement the complex service for its customer. The term *outsourcing* is often used to refer to the concept of one organisation assuming the responsibility for part of a business process of another organisation and carrying it out on its behalf.

The **CrossFlow**[1] architecture described here deals with the dynamic establishment and enactment of business relationships between a service consumer and a provider [3]. Services are provided and consumed as part of the *business processes* of the involved organisations using various resources such as applications, information repositories and people. If a business process spans an organisational boundary, a business relationship has to be specified in terms of what a provider organisation performs as a service for a consumer organisation. The business relationship may also specify that a consumer can have the ability to monitor and control the service while it is being performed on its behalf. All aspects of a business relationship between organisations are specified in a contract.

Companies need the infrastructure that supports all stages of a business relationship (the *business life-cycle*),

[1] CrossFlow is a 4th ESPRIT framework project, funded by the partner organisations, the European Union and the Swiss Federal Department for Education and Science.
See: http://www.crossflow.org

i.e. the advertising and finding services, establishing a contract, as well as connecting and managing the services performed by one organisation on behalf of another. The CrossFlow architecture supports the business life-cycle and is based on a lightweight, cheap infrastructure that is fast to set up once service consumer and provider agree on the business relationship.

Current business processes within organisations are integrated and managed using *Enterprise Resource Planning* (ERP) such as SAP R3 or *Workflow Management Systems* (WfMS) such as MQ Series Workflow. Thus, setting up a business relationship means connecting the process management systems of service consumers and providers, while catering for all the problems that lie along this connection (the *end-to-end* view). Such an *end-to-end* view has to cover the full range of issues concerning the link between two organisations, i.e. technical, conceptual and terminology differences, while protecting the autonomy and integrity of each organisation. The CrossFlow architecture covers the central issues of the end-to-end approach. The support for the business life-cycle, coupled with the end-to-end view, makes the CrossFlow architecture a comprehensive and powerful approach.

In the CrossFlow architecture, the contract between two organisations plays a pivotal role in setting up, enacting and supervising their relationship. It defines among other things the service, the rights and obligations of each partner and how much it will cost. It also defines the way in which a service consumer can influence the service while it is being performed, what the quality guarantees are and how they are measured, and much more. In addition, to achieve end-to-end integration, the partners of the business relationship have to specify in the contract common concepts, language and terminology and technical platforms. Each business partner then has to map the common contract view of the business relationship to its internal concepts, terminology, procedures and systems.

The paper is organised as follows: In §2, we explain the concept of the business life-cycle, in particular in the context of dynamic outsourcing. §3 describes the issues of the end-to-end connection of business processes. The CrossFlow architecture is introduced in §4, and related work in the field discussed in §5. Finally we summarise our approach, compare it to the related work, and provide an outlook on other aspects related to the CrossFlow architecture.

2 A Business Life-Cycle

The **business life-cycle** is a description of the stages two organisations have to go through to establish, enact, maintain and manage the desired business relationship between them. The major phases of the "traditional" business life-cycle are initiation, contracting, enactment and post-enactment [4],[5]. CrossFlow is concerned with the automation of the business life-cycle and therefore focuses more on some parts of the cycle, such as the possible automation of the configuration of organisational resources (IT and people).

In practice, there are many different flavours to a business life-cycle: its exact nature and the timing of the stages will vary from one implementation to another. The description provided below is a generic and abstract view of a business life-cycle.

2.1 Service contract establishment

Contract establishment involves the stages needed to define and agree on a business relationship between two organisations, culminating in a contract.

Making an initial contact consists of any form of advertising and searching in the market place in which the prospective service consumer and provider organisations operate (Fig. 2.1).

Once an initial contact has been established, information is exchanged between the two organisations to establish what is being offered and what is being requested, for

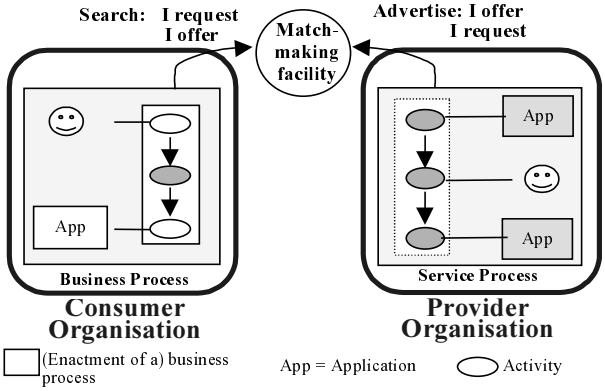

Figure 2.1. Advertising and searching for compatible consumers and providers in a market place.

example, in terms of service, QoS aspects, remuneration demands and promises, and legal guarantees (Fig. 2.2).

If differences exist between what is being offered and what is being requested, and if one or both of the organisations have room to manoeuvre, then negotiations concerning those specific differences may ensue (Fig. 2.3). Negotiations may take place regarding a single attribute,

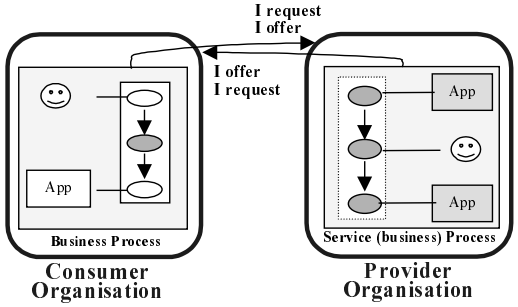

Figure 2.2. Information exchange and negotiations.

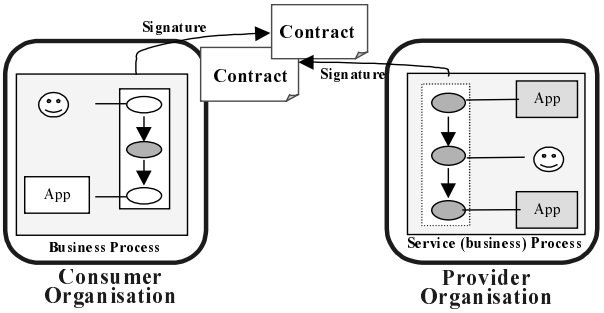

Figure 2.3. Signing and exchanging contracts.

such as price, or several attributes such as price/quantity combination. Negotiations are a highly complex problem and its automation constitutes an open research area.

If successful, these stages are expected to culminate in an agreement and a contract (Fig. 2.3). The agreement specifies what each side promises to deliver and expects to receive within the business relationship. An explicit notification of agreement is likely to be necessary, although in some cases, the consumption of the service contains an implicit agreement. Where necessary, copies of a contract stating what has been agreed on are signed by both parties (Fig. 2.3).

The contract defines the business process that crosses the organisational boundaries, the working relationship between the parties and the common view that both parties have of their business relationship. Thus, together with the organisational policy and resources, it determines the configuration chosen to enact the contract.

2.2 Service contract enactment configuration

Two sources influence the configuration of each organisation's resources for the enactment of the business relationship: the *contract* and the *internal policy*, which define the partnership in terms of:

- What internal resources can be used and how.
- The mapping between the external and internal definition and representation of business processes and their related enactment infrastructure.

The contract enactment set-up is necessary to carry out the service provision-consumption in accordance with the contract and entails the following (Fig. 2.4):

- **Configuring** the resources needed to carry out the enactment: the contract together with the *internal policy* can be used to derive the appropriate configuration. The configuration depends on the exact manner in which the business process is to be divided into management and work processes, and the existing support that can be expected from the environment in terms of available components and enactment infrastructure.
- **Connecting** the configured resources of the two organisations.

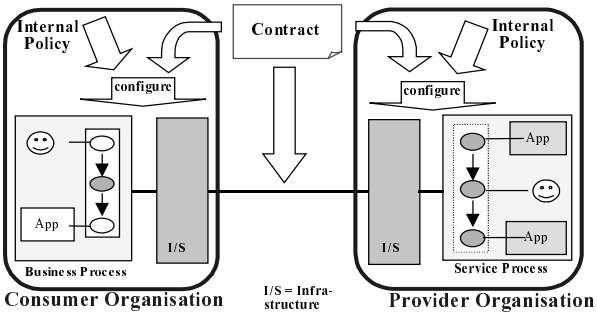

Figure 2.4. The *contract* and the *internal policy* are used to create and configure each organisation's resources.

2.3 Service contract enactment

Contract enactment entails supervising the enactment set-up, ensuring that the necessary administrative components are in place before service is consumed and that the conditions for service initiation/termination are correct. It also entails any actions that have to be taken around the service provision-consumption such as auditing and remuneration (Fig. 2.5). The contract enactment set-up may also have to support multiple service enactment cycles if contracts specify such a possibility.

While the service is being consumed (particularly in the case of long-lived business processes), the consumer

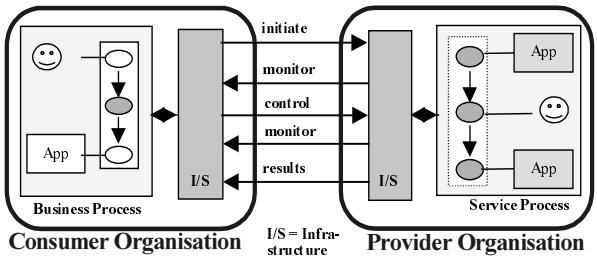

Figure 2.5. Service contract enactment (provision-consumption).

may inquire about the state of the workflow and its progress, modify the workflow or abort it, all in accordance with the contract. Similarly, the provider may notify the consumer of any problems or request for intervention. Monitoring and control may be based on a "push" or "pull" model of interaction.

Results of the service may be fed back from the provider to the consumer and the completion of the service consumption can then be determined, depending on the exact nature of the business process being enacted. This in turn may initiate the service contract termination.

2.4 Service contract termination

Contract termination entails that the *Contract Enactment Manager* fold up the link between the organisations when the service provision-consumption has ended and when all the auxiliary processes needed to complete the business process have taken place, as agreed on in the

contract. It also entails each organisation dismantling the components that have been set up to enact the contract and garbage-collecting the resources freed.

2.5 Service contract analysis

Service contract analysis consists of two main stages:

- **Contract validation**: validating the contract enactment against the undertakings of each party in the contract.
- **Contract evaluation**: evaluation of the results of the enactment against the objectives of the organisation.

Contract analysis can be exploited by both organisations to enhance the definition of the service to better suit the needs of the business parties.

3 An End-to-End View of Cross-Organisational Business Processes

The end-to-end view introduced in this chapter describes the possible physical and logical differences that exist between organisations. It regards the organisational boundary as a placeholder for the variety of boundaries described herein. The approach to the end-to-end problems is described in the next section.

A cross-organisational business process configuration involves the distribution of the process between two organisations, and all the resulting relationships between the autonomous and separated parts of the configuration (shown by the arrows in Fig. 3.1):

1. **Authorities:** The business relationship between the organisations revolves around issues of trust, giving the contractual part of the relationship its importance. Authorities are the entities that are allowed to negotiate, reach agreement and sign a contract. A contract has to cover all the aspects relevant for a harmonious interaction between the two organisations and their workflows. Each organisation is responsible for enacting the contract as it sees fit, depending on the policies and resources at its disposal, and as long as the enactment is in accordance with the contract. It is in the interest of each organisation to protect its autonomy while interacting with its counterpart.

Where the organisations reside in different countries there are likely to be differences in their legal systems. Agreement as to the applicable law within which the contract is to be interpreted will then have to be part of the contract.

2. **Business process:** The manner in which the same business processes are defined and described in each one of the organisations may differ. It will therefore be difficult to describe what each side requires and offers from a potential partner without being able to bridge between the differences. There are different ways of carrying out the activities that represent a **business process**. This manifests itself in different process models, modelling languages, and the operations provided to program the support infrastructure. Subsequently, the manner in which the business model is represented varies from one support

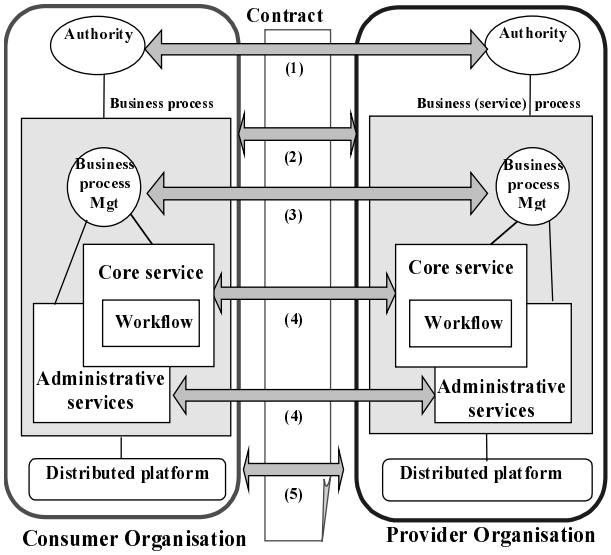

Figure 3.1. Relationships in a cross-organisational business process configuration.

infrastructure to another even for the same business process.

3. **Management of the business process:** Management is responsible for carrying out the contract within the constraints of the organisational policies and available resources. In the case of cross-organisational workflows there are, for example: organisational, departmental, application, Workflow Management Systems (WfMS) and distributed system management levels. Where workflows cross-organisational boundaries, those differences in management procedures will have to be overcome.

4. **Core and administrative services** and their **Support infrastructure and Process definitions**: The administrative processes deal with business-related activities such as remuneration and auditing that surround the core service being provided and consumed.

5. **Distributed application platforms**: The interaction inside and outside the organisations is supported by a distributed platform (e.g. CORBA, Java or MQ Series). In order to communicate, the platforms either have to be compatible or the appropriate translations applied.

The approach to the end-to-end problems is described in the next section.

4 The CrossFlow Architecture

Addressing the problems of the business life-cycle and the end-to-end view results in the CrossFlow approach, which forms the basis for the CrossFlow architecture.

4.1 The CrossFlow approach

The CrossFlow project is based on the following approach:

1. The CrossFlow project uses **workflow technology** to enact the business processes in the consumer and provider

organisations. Workflows provide a way of modelling and animating the sequence of work activities that represent a business process. WfMS provide organisations with the means to initiate, control and monitor workflows that implement business processes. By linking the WfMSs of different organisations, it is possible to connect their business processes and thereby address the requirement set by virtual enterprises.

2. Creation of a "closed" or a "vertical" **electronic market**: A market is where service contract offers and searches are conducted and where potential business partners are brought together. A closed market is a market with a well-defined information model and processes. Indeed, without a considerable amount of prior agreement between the players in a market place concerning a host of issues, such markets are unlikely to succeed. A closed market may also imply that participation is not open unless some registration procedure is adhered to.

The CrossFlow electronic market is based on the following:

- Standard ways of describing services: languages to describe services and products.
- Standard legal forms and processes: for example, *Contract Templates* provide standard legal procedures and contracts that evolved over time in a market place [6].
- Use of standard workflow interfaces and description languages [7], data representation formats (IDL/IIOP), distributed application standards, communication protocols, etc.
- A **Match-Making Facility** (**MMF**) where the advertisements and searches for compatible business partners are sent. The MMF used in CrossFlow, as the market point of advertising and searching, is an enhanced CORBA/ODP trading service [8], [9].

3. A detailed service specification in the form of a **contract** is the basis for **tight co-operation** between service consumer and provider [31].

4. The interaction between the two organisations as defined in the contract is carried out at a level that is **independent** of the specific enactment technology (CrossFlow uses a WfMS as its enactment technology).

5. Using **domains and gateways**: Organisational boundaries can be seen as a placeholder for many different types of domain boundaries such as remuneration, security, management, etc. where each type of domain boundary requires different actions to be taken when activities cross it [10]. When the workflows of two organisations are linked together, it is possible to exploit the explicit communications between them to install *proxy-gateways* that deal with the following aspects:

- **Heterogeneity:** *Translating* the differences between things inside and outside the domain. Examples are discussed in §3.1, and include different business models, definition, naming, implementation and representation of business processes, the manner in which a business process is translated into one or more workflows, communication protocols as well as on-the-wire-formats.
- **Encapsulation:** *Hiding* the internal details from other domains.
- **Security:** *Checking* the interactions to protect the domain's integrity and ensure that they do not compromise the domain's security.
- **Traceability:** *Monitoring* the interactions to provide an audit trail in case of disputes between the domains and to validate and enforce of the relationship between the interacting domains.

6. The service description in the contract includes a definition of the **visibility** of the provider's process and hence provides a definition of the monitoring that must be supported by the provider. In addition, providers may allow the consumer various levels of control of the enacted process.

7. Using the **contract** as the pivot for the interactions between the organisations and exploiting it to **generate the infrastructure and gateways** needed to support the cross-organisational business processes.

8. The functionality offered by the service provider and exploited by the service consumer is enhanced by the advanced co-operation support services (CSS). These services extend the control and monitoring capability of the provider service.

In addition to the above, certain aspects of the business life-cycle presented in §2 are omitted, e.g., negotiations, contract signing, administrative process such as remuneration. The decision to restrict the work to "closed" market places may at first seem overly restrictive. However, most markets for virtual enterprises are likely to restrict their scope to enable them to deal with the complexity of the problem — a generic solution that will apply to all markets, services, contracts and possible business relationships is way too far from being a reality at the moment.

4.2 Service contract establishment

The dynamic quest for a compatible business partner is done through a match-making facility and is based on advertising and searching for matching service contracts [12]. A typical sequence of events that leads to the establishment of a contractual relationship between the provider and consumer organisations is as follows (Fig. 4.1).

When the provider side is ready to launch business process, it notifies the *Contract Establishment Manager* (1), informing it of the service type and some of its given parameters. The relevant information concerning the desired contract to be associated with the service is fetched from the *Service Contract Repository* (2). The service offer advertisement is then sent to the chosen market place *Match-Making Facility* (3). A market is created by having several providers advertise their services to the *Match-Making Facility.*

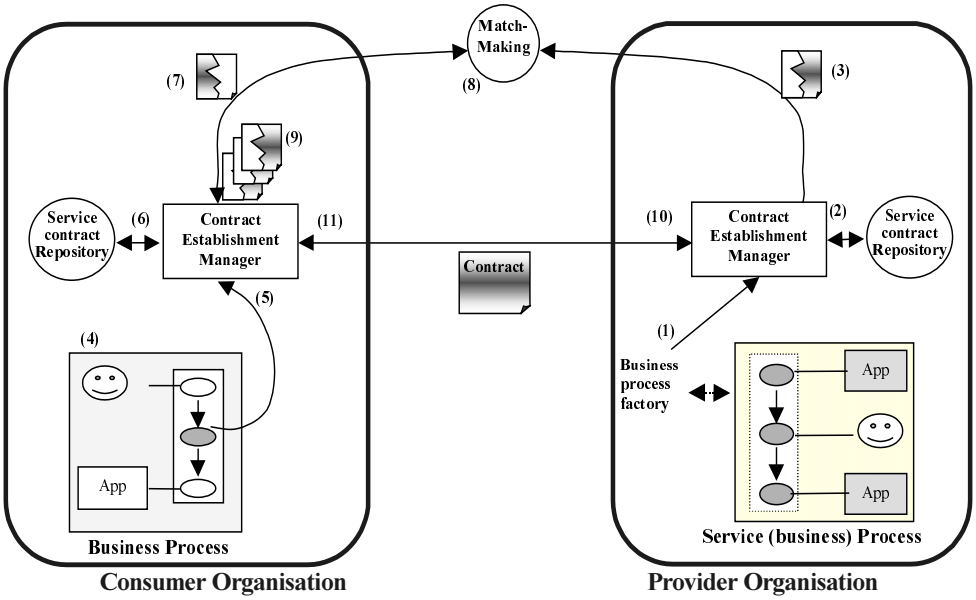

Figure 4.1. The contract defines the relationship between the organisations and influences the configuration of resources needed to enact the contract.

Sometimes later, after a consumer business process has been initiated (4), it may require an external service; it then provides the type and some parameters of the service to the *Contract Establishment Manager* (5). The relevant information concerning the desired service and its contract (affordable pricing, desired QoS, where to search, etc), is fetched from the *Service Contract Repository* (6). The search query is then sent to the chosen market place *Match-Making Facility* (7). The *Match-Making Facility* searches through its repository of advertised service offers, looking for matching offers and searches.

If the match-making process (8) is successful, one or more matching service offers are returned to the consumer organisation (9). The consumer *Contract Establishment Manager* selects one of the service offers. The consumer organisation notifies the selected service provider of its desire to enter a business partnership with it (10). The provider decides whether to accept the offer (i.e., in legal terms, the counter-offer) from the consumer. This can be regarded as an implicit contract. If accepted, the resources to be allocated for the outsourcing are determined. The provider's acceptance of the consumer offer together with any configuration information needed by the consumer to contact the provider is given to the consumer (11). The consumer can now configure the necessary components on its side and link them to their counter-parts on the provider side, see §4.3.

4.3 Service contract enactment configuration

A number of components are necessary for the enactment of workflows across organisational boundaries:

- **Contract Enactment support**: this is responsible for carrying out and overseeing the fulfilment of the contractual obligations as a whole. In addition to core service QoS guarantees and checks which are carried out under the service enactment support, there may be QoS issues which relate to overall contract issues such as timeliness of payment and security. In case multiple enactments of the service are provided for by the contract, this component is also responsible for creating the relevant *Service Enactment Support*.
- **Service Enactment support**: this is primarily responsible for carrying out the service provision-consumption according to its specification in the contract. This includes the co-operative support services mentioned in 4.1.
- **Proxy-Gateways** (**PG**): *PGs* deal with the crossing of domain boundaries (see §4.1), by translating between the internal-external and organisational differences, monitoring and controlling exit–entry to protect the organisation's integrity and security, and helping validate and enforce the contract.

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

Two input sources are required to configure the resources for the enactment of the business relationship between the organisations: the *Contract* and the *Internal Enactment Specification* (*IES*) (Fig. 4.2).

The **IES** is the organisation-specific blue-print that basically specifies how each contract type is to be enacted:

- What internal resources can be used and how.

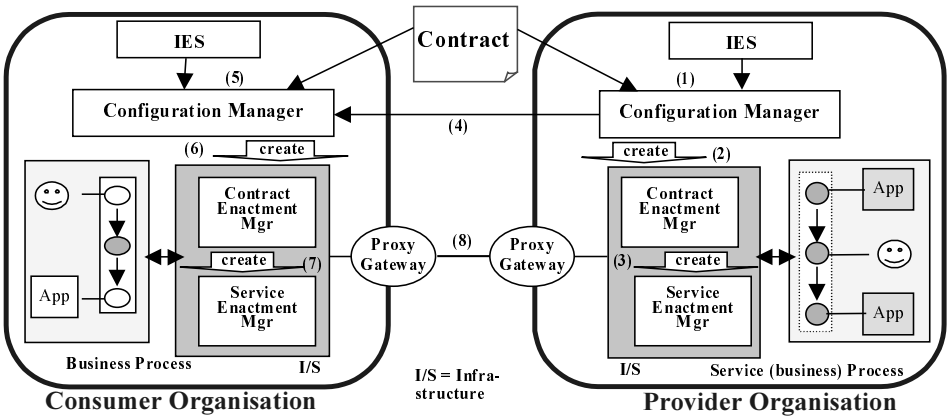

Figure 4.2. The Contract and the Internal Enactment Specification are used to configure the enactment components.

- The mapping between the external and internal details of the business process and its related workflow that is being out-sourced.
- The mapping between the workflow and the description of the service offered in the contract.

Because of this strong dependency, it is possible to derive the configuration of these components fully or partially from the contract and from the IES. This can take different forms ranging from the direct use of some formal representation of a contract to a derived intermediate form such as a script or code. Whichever form of component derivation is chosen, the interpreters or the derived components must be instantiated and/or initialised with the necessary information, and linked to the appropriate workflow and other components.

The necessary configuration of enactment components can be put into place as part of the contract establishment process described in §4.2. After accepting the consumer counter-offer to enter a business relationship, the provider can initiate the configuration of its resources. This is done by first deriving the necessary components (1) and then instantiating them (2).

Where the details of the specific service enactment are already known, the Service Enactment Manager can be created (3). This may however be postponed to a later stage if some specific service parameters are not yet known or if the contract allows for multiple service enactment cycles.

Once the provider configured its resources, it can pass the relevant details to the consumer (4). The consumer can now configure its infrastructure (5) by deriving the necessary components and instantiating them (6 and 7). Finally, the two infrastructures can be linked (8).

4.4 Service contract enactment (service provision-consumption)

When the set-up described in §4.3 is ready, the consumer can initiate the crossing of the business process by contacting the *Contract Enactment Manager* of the provider (1) (Fig. 4.3). It can check the details of the request and either use an existing instance of a *Service Enactment Manager* to deal with the request, provided it was created in the configuration phase (§4.3), or create an instance of a *Service Enactment Manager* to deal with the request (2). In this case, the relevant details are then passed to the consumer who can instantiate its own *Service Enactment Manager*.

Any monitoring information, as agreed in the contract, to be supplied by the provider to the consumer can either be sent as a notification (4) or requested by the consumer. Because of the progress update, the consumer may request that the provider modify the enactment of the business process. This may include a change of parameters or a change in the process direction or structure, depending

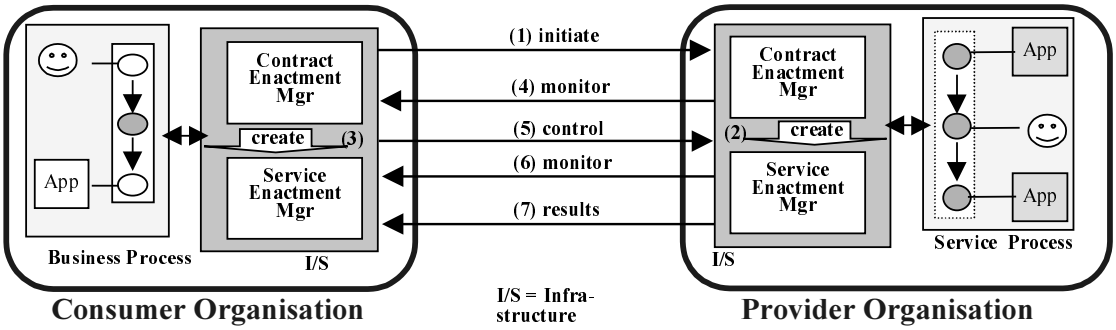

Figure 4.3. Service contract enactment (service provision-consumption).

on the contract (5). Further monitoring information may pass as a result (6) and more changes may be initiated where necessary. Ultimately, the completion of the process and its results will be indicated to the consumer (7).

4.5 Service contract termination

When all the administrative processes have been completed and both sides are satisfied, the infrastructure created earlier can be dismantled. Where multiple service enactment cycles are provided for by the contract, this may entail the dismantling of the *Service Enactment Manager* only. The *Contract Enactment Manager* may remain waiting for the initiation of the next service within the same contract.

The above description of the CrossFlow approach and architecture show the wide and involved range of topics that must be taken into account when dealing with the dynamic establishment and enactment of business relationships between a service consumer and a provider. A survey of the related topics, summarising the work done to-date in these areas is therefore necessary.

5 Related Work

The following is a brief survey of past and current work in a number of areas related to virtual enterprises.

Virtual market systems and service markets: In recent years we have seen the development of a number of systems that enable the advertising and searching in an electronic form of goods and services – marketplace systems [13]. Examples are ViMP [10] and MIT's Kasbah prototype [14]. Schmidt [4] provides an elaborated categorisation of electronic market systems. A number of related standardisation efforts are underway, such as RosettaNet [16] and the eCo Framework [17].

Service contracts and contract signing: Agreements and contracts are used differently in the area of transactions, workflow management and distributed systems, not always for the purpose of specifying mutual obligations between organisations, e.g. the ConTract approach for long-running transactions [19].

Contracts as a definition of a service relationship between organisations are used by a number of recent projects. The Coyote approach [20] provides mechanisms similar to ConTract, while explicitly taking into account that services as parts of transactions can be executed in different organisations. Milosevic [21] points out how contracts can be used to connect information systems of a service provider and a requester in a general way.

Work is also being done on languages for describing contracts. An example is the “courteous logic” approach, which is the basis of several concrete languages (e.g. Business Rules Markup Language – BRML) for expressing contracts between two parties [22]. The CrossFlow approach to the contract language is outlined in [31].

Distribution and interaction of process management systems: The use of distributed process management systems (within the same organisation) has been an issue for years and has been addressed by research in the database area [23]. Solutions have been implemented by major WfMSs such as IBM's MQ-Series Workflow.

An important issue concerns the interoperability between process management systems of different vendors. This is primarily a problem of providing standard interfaces for server-to-server communication, given that the underlying concept of a process is compatible. The Workflow Management Coalition, defined such an interface (called interface 4) [7]. The Simple Workflow Access Protocol [24] addresses the same issue, and is currently being integrated in the WfMC's framework as the XML binding of interface 4. These interfaces help cross WfMS vendor boundaries, but they do not address the issues of organisational boundaries.

A number of projects have addressed the issue of processes crossing organisational boundaries in the past. We can distinguish two approaches:

1. The *buyer-seller approach* where organisations can make process factories externally accessible and allow other organisations to pick their process from a catalogue and integrate it as an atomic step in their process. The Wide Area GroupFlow system [25] and the Virtual Enterprise Co-ordinator (VEC) [26] follow this approach.
2. The *common process approach*, where a number of organisations specify a common public process and assign the individual steps to each organisation. Each organisation then maps the steps onto activities within its internal process management system, which is not accessible by its partners. Examples are the WISE research prototype [27] and Extricity's Alliance product (http://www.extricity.com/).

In both approaches, the boundaries are established by gateways that maintain the integrity of the respective organisations. Ludwig et al [28] provide an overview of additional approaches with special emphasis on particular event-based approaches.

The issue of the organisations’ different ontologies must also be addressed. In the VEC system, the agreement between two organisations defines the common terminology used for a service relationship [26]. This terminology is mapped to the respective internal terminology.

The above-mentioned approaches deal with the interaction of process management systems, independently of a particular business domain. However, much work has been done in the area of domain-specific protocols. Gateways generate messages according to the protocol, triggered by the process management system; received messages can entail action on the process management system. Multiple organisations compete in providing frameworks to define message standards. One effort is Roset-

taNet [16]. Also, traditional EDI gateways from numerous vendors follow this approach.

Integration and end-to-end solutions: Until recently, there have only been a few systems that integrate a service market, the ability to agree on a service contract, the subsequent establishment of a service connection and the automatic control or supervision of the performance of a service. In other words covering the whole life-cycle of a business relationship is still a new area of work.

The *COSMOS project* is developing an architecture that allows organisations to offer and search for services in a catalogue, a negotiation platform, and facilities for contract signing [29]. 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.

The *WISE project* developed an architecture to model a virtual enterprise process by picking steps offered by several organisations from a directory, and to create a new "virtual process" on the WISE workflow engine. The new virtual process in turn contacts the participants WfMSs to trigger the work in the workflow-participating organisations [27]. WISE also proposes means for service providers to advertise process steps in a common directory and for service consumers to monitor the progress of a process in the WISE engine. *MariFlow* follows a similar approach [30].

6 Conclusions and Outlook

The CrossFlow architecture supports the business life-cycle and addresses issues raised by end-to-end integration of business processes. A *Contract Establishment Manager* supervises the advertising of a service contract from the provider's side and the search for a compatible partner from the consumer's side. The *Match-Making Facility* looks for compatible service contract offers and sends the result to the consumer. The *Contract Establishment Manager* selects a business partner and requests it to create the necessary enactment infrastructure. The *Configuration Managers* of each organisation use the *Contract* and its *Internal Enactment Specification* to create the necessary enactment infrastructures and link them together. The service provision-consumption can now be initiated through the enactment infrastructure in accordance with the contract. Monitoring and control of the progress of the enacted service is supported by this infrastructure.

This architecture provides a number of novel aspects that go beyond current approaches:

- The architecture regards the service model as more than a single atomic step, by allowing a flexible degree of service monitoring and control by the consumer.
- The contract approach provides short-term contracts for a single enactment cycle and long term contracts for multiple ones.
- Contract templates facilitate and simplify the match-making process.
- The *internal enactment specification* provides means of mapping the contract to the internal concepts, terminology and infrastructure. This internal mapping can be used to dynamically generate the infrastructure and gateways targeted to the particular requirements of a contract.

Supporting the full *business life-cycle* and catering for all the different *end-to-end* problems needed to connect the business processes of two organisations is a non-trivial and complex problem. The CrossFlow project approach limits the scope of the problem to the point where it can be addressed by extending current technology, while still allowing it to be of value in real life scenarios.

Future work will be necessary to fill the gaps and relax some of the assumptions made in this work. This will be based partly on practice and experience gained in the two application scenarios of the CrossFlow project. One future direction of work will be concerned with the contract framework. The use of fixed monolithic *contract templates* can be generalised to allow contracts to be built from units of smaller granularity. Reuse of contract templates can lead to a hierarchy of contract types coupled to a taxonomy market segments. Usage clauses in contracts have to be elaborated to specify in a flexible manner how contracts covering multiple service cycles can be instantiated into single-cycle contracts. Further work may extend the current one-step offer-request paradigm to negotiation. Finally, further development of the *co-operation support services* that implement the fine-grained monitoring and control of the service provider by the service consumer, are necessary.

The more dynamic and automated the process of establishing, setting-up, enacting, and managing a business relationship between two organisations is to be, the more it will have to adhere to common standards and practices. These common practices and standards will have to cover a wide spectrum of issues ranging from what services are to be provided, how they are to be described, and how they are to be carried out. The maturing of this field will therefore be accompanied by considerable efforts to create the necessary standards. CrossFlow results will be offered as input to the standardisation processes.

Acknowledgements

CrossFlow is a project in the 4th ESPRIT framework, funded by the partner organisations, the European Union and the Swiss Federal Department for Education and Science.

The authors would like to thank the CrossFlow consortium members for helpful discussions: Jacques Saint Blancat (IBM France), Matthias Kloppmann (IBM Germany), Karl Aberer, Justus Klingman and Thomas Tesch (GMD, Germany), Mar-

janca Koestier and Jochem Vonk (University of Twente), Willem Jonker, Wim Nijenhuis, Wijnand Derks, Matthijs Duitshof, Zef Damen (KPN Research, The Netherlands), Sinead Browne and Michael Kellett (Allianz Church & General, Dublin, Ireland), Agustine Gonzalez-Quel (SEMA Group, Madrid, Spain).

References

[1] R. Lee. "Distributed Electronic Trade Scenarios", *Int'l J. Electronic Commerce*, Sept. 1998.

[2] United Nations Economic Commission (UN/EC): Electronic Data Interchange for Administration, Commerce and Transport – Application Level Syntax Rules (ISO 9735), 1991.

[3] P. Grefen and Y. Hoffner. "CrossFlow – Cross-Organisational Workflow Support for Virtual Organisations", *Proc. IEEE 9th Int'l Workshop on Research Issues on Data Engineering – Information Technology for Virtual Enterprises RIDE-VE'99*, Sydney, Australia, 1999, p. 90.

[4] B. Schmidt: "Requirements for Electronic Markets Architecture", *Electronic Commerce*, Vol. 7, No. 1, pp. 3-6, 1997.

[5] R. H. Coase. *The Nature of the Firm*. Economica, New Series. Vol. IV, pp. 386-405, 1937.

[6] Draft Final Report relating to "Standard Form Contracts", NJLRC – New Jersey Law Revision Commission, John M. Cannel, Executive Director, New Jersey Law Revision Commission, 153 Halsey Street, Box 47016, Newark, NJ 07101, email: reviser@superlink.net, website: http://www.lawrev.state.nj.us.

[7] Workflow Management Coalition. *Workflow Standard – Interoperability Abstract Specification*. Document Number WfMC-C-1012, 1996.

[8] Object Management Group and X/Open. *RFP5 Submission: CORBA Trading Object Service*. Document orbos/96-05–6, May 10, 1996.

[9] *Open Distributed Processing Reference Model*, ISO/IEC 10476, ITU-T Recommendation X.900 (1995) Parts 1–3.

[10] Y. Hoffner, C. Facciorusso, S. Field, and A. Schade, "Distribution Issues in the Design and Implementation of a Virtual Market Place", in *Distributed Applications and Interoperable Systems* L. Kutvonen *et al.*, Eds. (Kluwer, Boston, 1999) Vol. 2, pp. 405-421.

[11] Y. Hoffner and B. Crawford, "Using Interception to Create Domains in Distributed Systems", in *Open Distributed Processing and Distributed Platforms*, J. Rolia *et al.*, Eds. (IFIP, 1997) pp. 251 – 263.

[12] Y. Hoffner and A. Schade, "Co-operation, Contracts, Contractual Match-Making and Binding", in *2nd Int'l Enterprise Distributed Object Workshop, "EDOC 98"*, (IEEE, Piscataway, 1998), pp. 75 – 86.

[13] Y. Bakos. "A Strategic Analysis of Electronic Marketplaces", *MIT Quarterly*, September 1991.

[14] A. Chavez and P. Maes. "Kasbah: An Agent Marketplace for Buying and Selling Goods", in *Proc. First Int'l Conf. on the Practical Application of Intelligent Agents and Multi-Agent Technology*, London, UK, April 1996.

[15] Y. Hoffner, "Supporting Contract Match-Making", in *Proc. 9th Int'l Workshop on Research Issues in Data Engineering – Information Technology for Virtual Enterprises "RIDE-VE'99"*, Sydney, Australia, pp. 64-71.

[16] RosettaNet, *RosettaNet Implementation Framework (RNIF) Specification*, Version 1.1. 1999. See: (http://www.rosettanet.org/).

[17] H. Smith, *CommerceNet's eCo Framework in a Nutshell!* Onotology.org, 1999.

[18] J. Shepherdson, S. Thompson and B. Odgers, "Cross Organisational Workflow Co-ordinated by Software Agents", in *Proc. WACC Workshop on Cross-Organisational Workflow Management and Co-ordination.* San Francisco, 1999.

[19] H. Wächter and A. Reuter, "The ConTract Model", in *Database Transaction Models for Advanced Applications.* A. Elmagarmid, Ed. (Morgan Kaufmann, San Mateo, 1991).

[20] A. Dan and F. Parr, "The Coyote Approach for Network Centric Service Applications: Conversational Service Transactions, a Monitor and an Application Style", in *Proc. Workshop on High Performance Transaction Processing (HPTP)*, Asilomar, 1997.

[21] Milosevic, Z., Berry, A., Bond, A. and Raymond, K. "Supporting Business Contracts in Open Distributed Systems", Proceedings of the Second International Workshop on Services in Open Distributed Processing (SDNE95), Whistler, Canada, June 1995.

[22] B. Grosof and Y. Labrou, "An Approach to using XML and a Rule-based Content Language with an Agent Communication Language", in *Proc. IJCAI-99 Workshop on Agent Communication Languages (ACL-99).* 1999.

[23] J. Gray and A. Reuter, *Transaction Processing - Concepts and Techniques* (Morgan Kaufmann, San Francisco, 1993).

[24] K. Swenson, *Simple Workflow Access Protocol (SWAP).* Draft Proposal of May 6, 1998.

[25] G. Riempp and L. Nastanski, "Workflow Management between Distributed Organizations – The Wide Area Groupflow Approach", in *Proc. ESTIEM, IT-Vision Conf.*, F. Lehner and S. Dustar, Eds. (DUV, Gabler Edition Wissenschaft, Wiesbaden, 1997), pp. 265-282.

[26] H. Ludwig and K. Whittingham. "Virtual Enterprise Coordinator – Agreement-Driven Gateways for Cross-Organisational Workflow Management", in *Proc. Int'l Joint Conf. on Work Activities Coordination and Collaboration (WACC '99)*, San Francisco, February 1999, D. Georgakopoulos *et al.*, Eds. & *ACM Software Engineering Notes*, Vol. 24, No. 2, March 1999.

[27] G. Alonso, U. Fiedler, C. Hagen, A. Lazcano, H. Schuldt, and N. Weiler, "WISE: Business to Business e-Commerce", In *Proc. Ninth Int'l Workshop on Research Issues in Data Engineering – Information Technology for Virtual Enterprises (RIDE-VE'99)*, Sydney, Australia, 1999).

[28] H. Ludwig, C. Bussler, M.-C. Shan, and P. Grefen, "Cross-Organisational Workflow Management and Co-ordination WACC '99 - Workshop Report", *SIGGROUP Bulletin*, pp. 59-62, April 1999.

[29] F. Griffel, M. Boger, H. Weinreich, W. Lamersdorf, and M. Merz, "Electronic Contracting with COSMOS – How to Establish, Negotiate and Execute Electronic Contracts on the Internet", in *Proc. Second Int'l Enterprise Distributed Object Computing Workshop (EDOC '98).* La Jolla, 1998.

[30] I. Cingil, A. Dogac, N. Tatbul, and S. Arpinar, "An Adaptable Workflow System Architecture on the Internet for Electronic Commerce Applications", In *Proc. Int'l Symp. on Distributed Object Applications.* Edinburgh, September 1999.

[31] M. Koetsier, P. Grefen, J. Vonk, *Contract for Cross-Organizational Workflow Management*, CTIT Technical Report 99-18, University of Twente, 1999.