CrossFlow: Integrating Workflow Management and Electronic Commerce

Y. Hoffner, H. Ludwig , P. Grefen and K. Aberer

IBM Zurich Research Laboratory
Switzerland

{yho,hlu}@zurich.ibm.com

Computer Science Department
University of Twente,
The Netherlands

grefen@cs.utwente.nl

Distributed Information Systems Laboratory
EPFL-DSC, CH-1015 Lausanne
Switzerland

karl.aberer@epfl.ch

Abstract

The CrossFlow[1] architecture provides support for cross-organisational workflow management in dynamically established virtual enterprises. The creation of a business relationship between a service provider organisation performing a service on behalf of a consumer organisation can be made dynamic when augmented by virtual market technology, the dynamic configuration of the contract enactment infrastructures, and the provision of fine grained service monitoring and control.

Standard ways of describing services and contracts can be combined with matchmaking technology to create a virtual market for such service provision and consumption. A provider can then advertise its services in the market and consumers can search for a compatible business partner. This provides choice in selecting a partner and allows the deferment of the decision to a point in time where it can be made on the most up-to-date requirements of the consumer and service offers in the market. The penalty for deferred decision making is the time to set up the infrastructure in each organisation for the dynamically established contract. Thus, a further aspect of CrossFlow was to exploit the contract in the dynamic and automatic configuration of the contract enactment and supervision infrastructures of the respective organisations and in linking them in a dynamic fashion. The electronic contract, which results from the agreement between the newly established business partners, completely specifies the intended collaboration between them. Given the importance of the business process enacted by the provider, this includes fine-grained monitoring and control to allow tight co-operation between the organisations.

1 The CrossFlow approach

1.1 Introduction

Virtual enterprises are based on the ability to dynamically form and dismantle partnerships between organisations, enabling parts of their business processes to be performed for each other as a service[2]. To enable this to be carried out in a dynamic fashion, several essential elements outlined below need to be brought together.

One element is the creation of a “closed” or a “vertical” **electronic market** where service contract offers and searches are conducted and where potential business partners are brought together. To-date, most advertising and searching facilities (matchmaking engines) focus on physical discrete objects like books, materials like oil or wheat, or immaterial objects like seats in an aeroplane. To support the dynamic creation of tightly linked virtual enterprises, however, business processes or business services are to be traded. This requires a considerable degree of agreement on the way to specify services in terms of abstract process structures, process parameters, quality of service guarantees and primitives to monitor and control the enactment of services. Furthermore, this requires a standardisation of services in the context of specific application domains, like the logistics industry or the insurance industry. This helps ensure that the participants in the market use the same language to describe the services that are being offered.

The importance of the business relationship between organisations in a virtual enterprise is such that it must be described in a **contract**, preferably specified in an electronically usable form. Contracts are based on agreed legal forms and procedures that evolve over time in specific markets. A contract includes a detailed service specification as the basis for tight co-operation between service consumer and provider.

---

[1] The CrossFlow architecture is the result of a two-year ESPRIT/IST project that ended in September 2000. The CrossFlow web site: www.crossflow.org.

[2] This is sometimes referred to as outsourcing: the consumer outsources the service to the provider.

An important element in CrossFlow is the level of **abstraction** provided above the particular enactment infrastructure chosen in an organisation. The interaction between the two organisations is defined at a level that is **independent** of the specific enactment technology. In today's businesses, the application of workflow management systems (WfMSs) for automated process support is widespread and it was therefore chosen as the project's enactment technology. Workflows provide a way of modelling and **enacting** the sequence of work activities that represent a **business process**. The use of WfMSs ensures a well-structured and standardised management of processes within organisations. Traditionally, the emphasis of workflow management has been on homogeneous environments within the boundary of a single organisation. Using workflow support in virtual organisations, however, implies extending the functionality of workflow support such that workflow management systems in different organisations can be linked to manage integrated cross-organisational processes. The extended workflow support must be able to deal effectively with heterogeneous workflow environments, well-specified levels of autonomy of partners in a virtual enterprise, and dynamic formation of new and dismantling of existing collaborations. Linked workflow systems should allow one organisation (the service consumer) to start a process (a service) on its behalf in another organisation (the service provider) and receive the results of this process. As black-box processes are too coarse for tightly co-operating organisations, **advanced monitoring and control mechanisms** are required to support fine-grained interaction between these organisations, while preserving their autonomy as much as possible.

To provide a business process outside the organisational boundary usually requires additional functionality to that provided by the core service being offered. There is therefore a need to integrate the core functionality of the service with additional monitoring, control and transaction management functionality where needed, as well as administrative functionality such as accounting and billing. Contracts are used in the **dynamic generation** of the **enactment infrastructure** that supports the provision and consumption of a service outside an organisation.

CrossFlow was a European research project in the 4th ESPRIT Framework (currently IST 5th Framework) that researched and developed cross-organisational workflow support for dynamic virtual enterprises based on an provider/consumer (outsourcing) paradigm. The CrossFlow architecture provides an end-to-end solution, including all functionality from contract establishment to workflow enactment for executing services.

1.2 The CrossFlow life-cycle

A detailed discussion of the CrossFlow approach to service provision/consumption in dynamic virtual enterprises can be found in [Gre00]. Here, we illustrate the CrossFlow approach by means of a simplified view of the CrossFlow architecture. During the service provision/consumption life cycle, the relationship passes different phases [Hof00]:

1. **Contract establishment** to create the specific contract that will define the business relationship.
2. **Dynamic infrastructure creation** for enactment of the provided service and the linking of the components in both organisations together
3. **Enactment** of the provided service including cross-organisational monitoring and control.

In phase 1, the CrossFlow system acts as an electronic commerce platform whose aim is to establish a contractual relationship. In phase 3, the CrossFlow system acts as an advanced cross-organisational workflow management system between the business partners. Phase 2 provides the transition between the contract establishment and enactment phases.

2 Contract establishment

2.1 The contract model and contract templates

The CrossFlow contract model provides the conceptual structure that describes the tight collaboration of service consumer and provider [Koe00]. The design of the model includes concepts for representing the structure of the provided service process described by the contract, high level concepts for monitoring and controlling this process in a cross-organisational context, and concepts for flexible use of contracts. A modular contract structure has been chosen to allow easy adaptation and extension to specific environments. The overall model consists of several parts. The **Concept model** defines the concepts that are used in the contract, creating a concept space in which the other contract issues can be specified. This is similar to the terminology statements in the first section of a regular contract and is related to the electronic market ontology. The **Process model** describes the internal structure (schedule) of the workflow implementing the service at the contract level. The process schedule is composed of process elements,

i.e., the individual activities and transitions. The **Enactment model** provides concepts to represent the advanced co-operative support that is offered during service enactment. Co-operative enactment support can be composed of a number of elementary services, like service execution monitoring, service execution control, remuneration support, and authentication support. The **Usage model** defines the manner in which the contract can be used. It describes the different usage possibilities of the contract and their conditions allowing, for example, short-term contracts for a single enactment cycle and long term contracts for multiple ones. The **Natural language description** is meant for human reading.

Contracts vary enormously in their size, content and complexity. The two reasonably simple contracts specified in the CrossFlow scenarios consisted on hundreds of XML lines. In order to expedite the construction, consideration and processing of contracts, common contract forms often evolve and become a **Standard form contract** [Bou98], also referred to as **Contract template** [Hof99]. The term standard in this context means that it is either agreed among all participants in some restricted application domain, or that it could be a bi-lateral agreement used frequently between two specific organisations. Contract templates are characterised by several things: The pre-agreed content and format, the placeholders or fields for contract instance values, and the 'take it or leave it' basis on which contract templates are often offered.

2.2 Contract matchmaking

In an electronic market for service-based virtual enterprises, three major entities exist. Service providers can enact services on behalf of consumer organisations, and service Matchmaking Engines play a role in creating a market place. An example of a service-based market is a logistics market, in which providers offer logistics services to consumers who do not want to implement their own logistics.

When the provider WfMS is ready to receive requests for enactment of a process on behalf of a consumer organisation, its manager notifies its Contract Manager of its readiness to provide instances of that process. 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 - the offer to support instances of this service for different consumers, is then advertised into the **Matchmaking Engine** (**MME**) that serves the specific market. In a competitive market, several provider organisations 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. 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 MME. The result is then sent as a search query into the MME serving the market.

The MME 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 organisations. Although this is outside the scope of the CrossFlow project, the agreement between the organisations can be digitally signed, making the contract an explicit legal entity.

The architecture supporting this process is depicted in Figure 1. Both consumer and provider organisations use a workflow management system to control their business processes. CrossFlow contract manager modules are used to contact the service MME and to make a contract between two parties. The **Workflow module** (**WM**) shields the contract manager from the workflow management system thereby obtaining portability across workflow management platforms (the CrossFlow project used MQSeries Workflow system [IBM00] but the architecture can deal with other WfMSs). The CrossFlow Matchmaking Engine is based on IBM's e-market technology [Hof99].

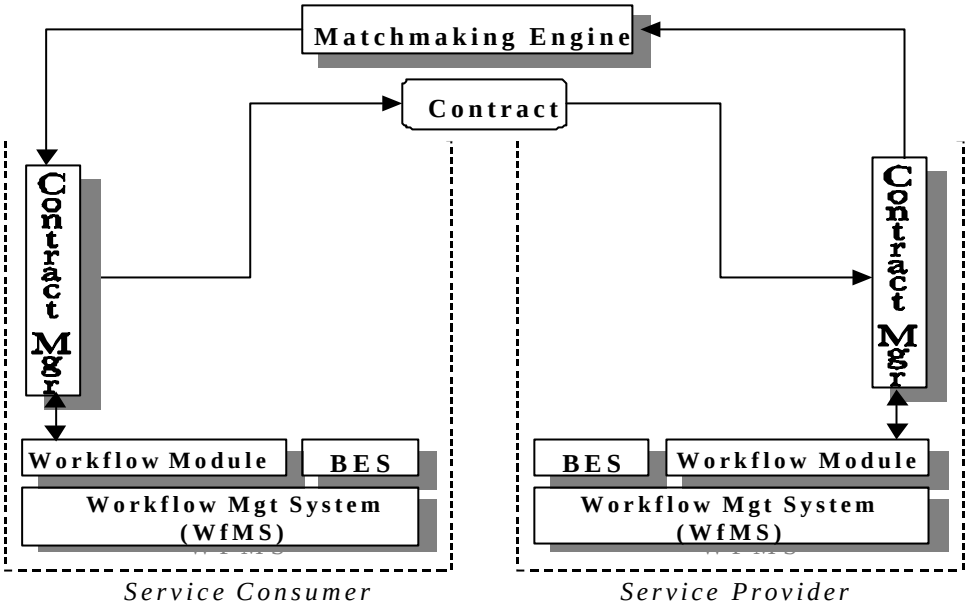

Figure 1: Contract establishment

3 Dynamic infrastructure configuration

3.1 Configuration approach

Once a virtual enterprise has been defined in an electronic contract, its details are used for the dynamic construction of the infrastructure needed for the enactment of the promised service. **Enactment configuration managers** (**ECM**) of the service consumer and service provider build the contract-specific enactment infrastructure separately for the service provider and service consumer [Hof00]. They do this by combining the contract with an internal blueprint, the **Internal enactment specification** (**IES**), that describes the manner by which this specific contract is to be implemented in the respective organisation [Lud01]. The result is called an **Integration facilitator** (**IF**). An IF integrates the core service functionality with any additional functionality needed to offer or consume the service externally, into a coherent service. In CrossFlow, the core functionality is provided by a Workflow management system (WfMS). Additional functionality may include billing, payment handling and accounting, as well as access control and quality of service management One IF corresponds to one service contract between two organisations.

IFs consist of a set of components that either implement core and additional functionality or wrap existing **Back end systems** (**BES**). **CRAFT** is a component framework that provides an approach for building Integration Facilitators corresponds to one service contract between two organisations [Lud01].

3.2 CRAFT

CRAFT defines a set of object types, interaction patterns and the assembly rules for creating contract specific IF instances. The CRAFT objects are:

Proxy-gateways (**PG**): Both parties may require a certain degree of "hiding" in representing and controlling the service and a service may therefore be represented externally with reduced complexity. Where necessary, the IF translates aspects of interactions between organisations such as data formats and naming conventions. By using relevant security mechanisms in the PGs, the IF may limit the access to services, i.e., the usage of services of an organisation from outside as well as the usage of outside services from within. The need to monitor and supervise the execution and performance of services becomes crucial when organisations make contractual promises for which they are liable.

Co-operative support services (**CSS**)**:** When services cross organisational boundaries, the internal model and mechanisms for measuring resource usage, costing, billing and paying have to be modified and sometimes extended. Monitoring, control and transaction management capability may also be necessary in some cases. This additional functionality is provided by CSSs. The CSSs can be used to monitor the interactions between the partner organisations and trigger the billing and payment for usage of service resources, either through BESs or as parts of the IF.

Coordinator: the Coordinator facilitates Interaction between PG and CSSs. This de-couples the various CSSs from each other, providing a modular 'plug and play' like structure where each CSS has its defined role and interacts only with those CSS it needs to.

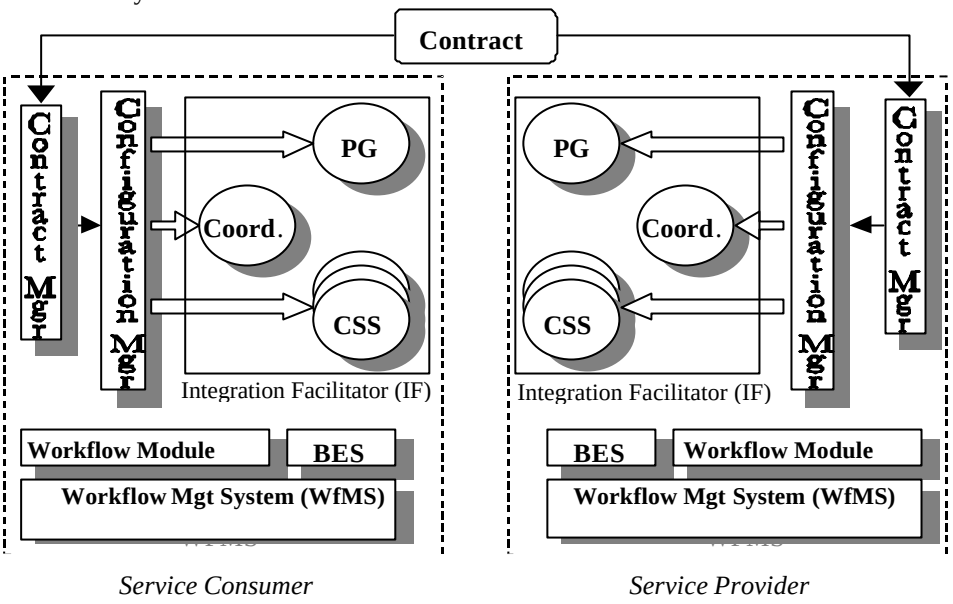

Figure 2: Dynamic infrastructure configuration (BES stands for Back end systems).

Interactions between two organisations originate from CSSs and are passed through the PGs. It transforms an outgoing interaction from its internal format to a contract format and level of abstraction and sends it to its peer PG. The receiving PG transforms the interaction into its internal representation. Once transformed to the internal representation, the PG forwards the interaction to the coordinator. Interactions inside an organisation are controlled by the IF Coordinator, which then forwards them to the appropriate CSSs.

CSSs can assume three different roles:

1. **Supervisor:** A supervisor CSS decides whether a message should be accepted at the current state of the contractual relationship. For example, an accounting CSS could check if the business still has enough credit to perform a control operation, assuming the control operation entails a fee. A message can be associated with multiple Supervisors.
2. **Actor:** An Actor CSS implements the impact on the core service that may be needed by a message, e.g. suspend a workflow. There is exactly one Actor associated with an inbound interaction.
3. **Listener:** A Listener is notified by the Coordinator about outbound and inbound interaction, e.g. for the purpose of measurement of QoS parameters. Multiple Listeners can be associated with one message.

A CSS can assume multiple roles and each role may be implemented by multiple CSSs.

Inbound interaction are dealt with in the following way: A message is received by the PG and forwarded to the Coordinator. Subsequently, the Coordinator asks the relevant Supervisors whether the message is eligible to be processed. If all Supervisors agree, a defined set of Listeners is notified prior to acting according to the message. Then, the associated Actor performs the action that corresponds to the message. After completion, another set of Listeners might be notified of the completion of the message processing. Finally, if the message implies an immediate return of results, a results message is passed back to the business partner through the PG.

Outbound interaction are dealt with in a similar manner. Supervisors are queried for permission before sending an interaction. Listeners can be notified before or after sending the interaction.

The Coordinator maintains the list of which Supervisors, Listeners and Actors to involve in the processing of a given interaction type.

3.3 The Internal Enactment Specification and the Enactment Configuration Managers

For each contract type and for each organisation, an IES defines how the IF is to be configured. The configuration comprises of the set of CSSs (and the roles in the CRAFT framework that they can assume), the PG, and the Coordinator. The definition of each of those components may include a specification of

its parameters. Component parameters may be elements of the contract, e.g. the shared view of the workflow, or other values defined in the IES.

Another important element of the IES is the definition of how inbound and outbound interaction should be dealt with. For each contractual interaction, the IES contains a specification of the Supervisors, the Actor, and (pre- and post-execution) Listeners that are to be involved. This specification is passed to the Coordinator and completes the specification of its behaviour.

To avoid the need to create an IES for each contract, we define and associate an **IES Templates** with every Contract Template defined for the market in which the organisation operates. The Contract Manager takes the contract specific parameters and adds them to the IES Template to create the specific IES which it then hands the Configuration Manager. When the Contract Manager invokes the Configuration Manager with a contract and its IES, the Configuration Manager checks its consistency, instantiates and parameterises all the IF components and configures the Coordinator with the interaction specification. If these steps are completed, contract enactment may begin.

4 Contract enactment

The enactment of the contract through the service provision/consumption, requires a complex cooperation between all CrossFlow modules and the commercial platform below them, as outlined in the previous section. After the enactment infrastructure has been set up, the provider service can be started. For this purpose, the various modules communicate with each other as illustrated in Figure 3 to provide the functionality as well as any monitoring and control capability specified in the contract. Specific CSS modules may need to access dedicated Back End Systems (BES) to perform their tasks. Such BESs may consist of internal logistic services, accounting and other management systems or any legacy systems which provide parts of the core or the administrative functionality needed to support an externally provided service.

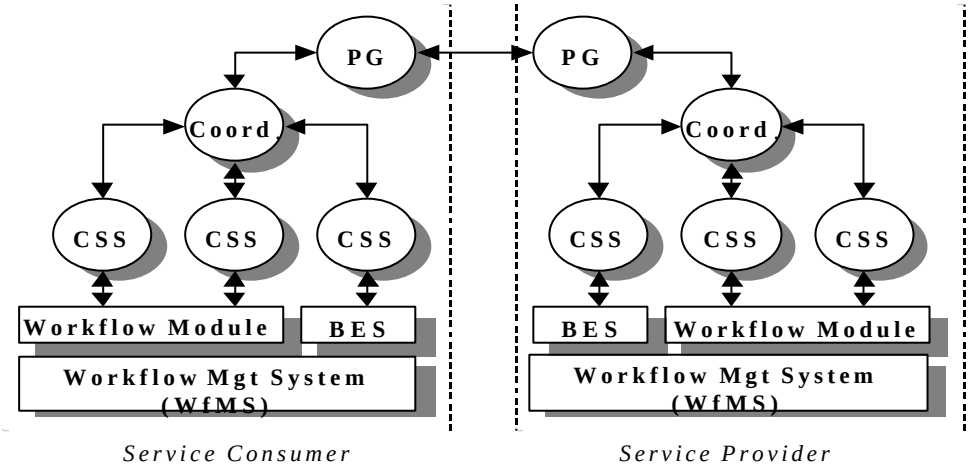

Figure 3: Contract enactment

5 Conclusions and future work

The CrossFlow architecture provides support for cross-organisational workflow management in dynamically established virtual enterprises. It provides a number of novel aspects that go beyond current approaches:

- Contract templates facilitate and simplify the matchmaking process and therefore allow a more dynamic form of establishing partnerships.
- The contract model provides short-term contracts for a single enactment cycle and long term contracts for multiple ones.
- The use an organisational blueprint (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 targeted to the particular requirements of a contract.
- 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. This is a desirable feature in virtual enterprises where business processes cross organisational boundaries.

- The set of monitoring, control and transaction co-operative support services (CSS) provided by the project can easily be augmented by additional services to suit specific circumstances.
- Advanced workflow management technology that allows business processes to cross organisational boundaries is combined with the creation of a virtual market place to create dynamic virtual enterprises.

Further work is required in a number of areas. The creation of virtual markets to support the dynamic creation of virtual enterprises requires a considerable amount of agreement between the market players. More specifically, further work is needed in the area of contract specification languages that can be used in the matchmaking process and also be exploited to generate the contract enactment infrastructure dynamically.

Acknowledgements

All current and former members of the CrossFlow project team are acknowledged for their contributions to the project.

6 References

[Bou98] A. Boundy, *A Concise Business Guide to Contract Law*, Gower Publishing Ltd. UK, 1998.

[Gre99] P. Grefen, B. Pernici, G. Sánchez (eds.); *Database Support for Workflow Management*; Kluwer Academic Publishers, 1999.

[Gre00] P. Grefen, K. Aberer, Y. Hoffner, H. Ludwig; *CrossFlow: Cross-Organizational Workflow Management in Dynamic Virtual Enterprises*; International Journal of Computer Systems Science & Engineering, Vol. 15, No. 5, 2000.

[Hof99] Y. Hoffner; *Supporting Contract Match-Making*; Procs. 9th IEEE International Workshop on Research Issues in Data Engineering; Sydney, Australia, 1999.

[Hof00] Y. Hoffner, H. Ludwig, C. Gülcü, P. Grefen; *Architecture for Cross-Organisational Business Processes*; Procs. 2nd International. Workshop on Advanced Issues of E-Commerce and Web-Based Information Systems, WECWIS 2000; Milpitas, CA, USA, 2000.

[IBM00] MQSeries Workflow Web Site; IBM Corporation, USA; www.software.ibm.com/ts/mqseries/workflow/.

[Kli99] J. Klingemann, J. Wäsch, K. Aberer; *Adaptive Outsourcing in Cross-Organizational Workflows*; Procs. 11th Conference on Advanced Systems Engineering; Heidelberg, Germany, 1999.

[Kli00] J. Klingemann; *Controlled Flexibility in Workflow Management*; Proceedins. 12th Conference on Advanced Systems Engineering; Stockholm, Sweden, 2000.

[Koe00] M. Koetsier, P. Grefen, J. Vonk; *Contracts for Cross-Organizational Workflow Management*; Procs. 1st International Conference on Electronic Commerce and Web Technologies; Greenwich, UK, 2000.

[Lud01] H. Ludwig, Y. Hoffner, *CRAFT: A Framework for Integration Facilitation in Cross-Organisational Distributed Systems*. To appear in: *Proceedings of the 9th EUROMICRO workshop on Parallel and Distributed Processing (PDP'01)*. IEEE Computer Society, Mantova, Italy, February, 2001.

[Von00] J. Vonk, W. Derks, P. Grefen, M. Koetsier; *Cross-Organizational Transaction Support for Virtual Enterprises*; Procs. 5th International Conference on Cooperative Information Systems; Eilat, Israel, 2000.