

Yigal Hoffner

IBM Research, Zurich Research Laboratory
8803 Rüschlikon, SWITZERLAND
e-mail: yho@zurich.ibm.com

This paper addresses the problem of how to make the establishment of business partnerships more dynamic. It examines what is necessary to support the deferral and automation of decision-making, resource allocation as well as certain processes within the business life cycle: the contract establishment, the client-service configuration and the instantiation of the components and their linking.

The paper outlines the restrictions and simplifications that must be applied to the generic life cycle in order to automate parts of it and make it dynamic. It then describes the services needed to support the resulting e-Business on Demand life cycle, and shows what the typical interactions between the parties would look like. Finally, some suggestions are made regarding the relaxation of those initial restrictions to provide more flexibility and variety.

1. e-BUSINESS AND BUSINESS RELATIONSHIPS

e-Business on Demand is concerned with the dynamic and flexible provision and consumption of services as part of the business processes of an organisation, either within or across organisational boundaries. The possibility of a business process crossing an organisational boundary raises the need to view the service provision-consumption as a business relationship with all the complexity such a relationship entails: matchmaking, information exchange, negotiation, contract generation, dynamic creation of configurable service instances, service provision-consumption and service termination. A virtual enterprise is the result of the contractual relationship that allows a business process to cross from one organisation to another and back.

Recent advances in distributed computing infrastructures [ODP 95] [OMG 03], Web services [WS 02] and Internet technology provide an incentive to push the level of automation of organisational processes further than has been attempted so far. The process of setting up business relationships itself is beginning to attract considerable attention and is becoming the focus of automation attempts [Grefen 00] [Hoffner 01]. A central question arises as to whether it is possible to automate the processes of searching, selecting, exchanging information, negotiating, writing and signing contracts without requiring the time-consuming and costly intervention of hoards of domain experts, lawyers and accountants. Would it then be possible to use the contract to instantiate the components necessary for contract enactment?

There is a need to examine the different stages that organisations go through to establish and enact a business relationship successfully. This will help determine how those processes have to be modified if they are to be automated in the context of current technology. Such an investigation forms the basic work needed in order to support e-Business on Demand and virtual enterprises. The following discussion therefore concentrates on a **business-to-business environment**. Although the details in the case of a business-to-consumer environment are slightly different, the same principles apply.

A **business life cycle** is a description of the steps that two organisations have to go through in order to establish, enact, maintain and manage the desired business relationship between them [Milosevic 95] [Dignum 01] [Vetter 01]. While in practice there are many different ways to create and enact a business relationship, we nevertheless regard a “generic” life cycle as consisting of the following phases:

- Find potential partner,
- Define client-service relationship: exchange information, negotiate and agree,
- Establish client-service contract,
- Construct service and client and related infrastructure,
- Deploy service and client,
- Operate service and client (provide and consume) and finally,
- Terminate the relationship.

The most noticeable thing about the generic life cycle is that people bring an inordinate level of flexibility and innovation into the process. On the other hand, where frequent and repetitious relationships have to be set up, enacted and dissolved, the prospect of automation can increase the speed, while reducing the response time and the costs involved. The **central question** is whether it is possible to change the life cycle processes to the point where current technology can enable automation, while making sure that the problems are not trivialised to the extent that they bear no resemblance to the business reality they are supposed to help.

We believe it is possible and suggest a two-stage approach. The first stage consists of an **initial solution**, suggesting restrictions that enable the deferral and automation of processes and decisions but limit the flexibility and choice in the life cycle. This is the subject of the current article. It is followed by an **advanced solution**: a way of relaxing the restrictions imposed in the initial solution and making it more complex. Such an approach increases flexibility and variety while maintaining the advantages of deferral and automation. The advanced solution is beyond the scope of this paper, but we will make a few brief comments on this subject in Section 4.

This paper draws upon the experience gained in the CrossFlow project [CrossFlow] [Hoffner 01] as well as extensive work on advanced forms of matchmaking [Field 02], work on electronic commerce [Cosmos 00] [Alonso 99] [Griffel 98] [Cingil 99] and Agent technology [Dignum 01] [Vetter 01].

1.1 Simplify, Defer and Automate

The primary principles on which we base the changes in the generic life cycle are the simplification, deferral and automation of certain processes in the life cycle. The open-ended nature of the life cycle is simplified by limiting the scope of possible kinds of partnerships, restricting their flexibility and constraining the type of client,

services and contracts dealt with. A central principle of simplifying this complexity is a generalisation of the “**no surprise**” principle behind the type system of CORBA (with IDL [OMG 03]) and Web services (with WSDL [WSDL 01]). The idea is that interactions between the parties are fully defined and choices can be offered during the life cycle as long as they can be dealt with by the other party. The internal implementation of the processes can vary to fit the organisation and its resources, as long as what the other side expects to happen does indeed take place. This approach results in:

- Defining standard client-service contract templates and information models that can be used in the dynamic parts of the life cycle [Law 98].
- Creating closed domains that deal with these standard client-service definitions and contracts and where membership together with an adherence to a prescribed life cycle ensures an acceptable degree of trust between the potential business partners.

There are two central themes in making a process dynamic: **deferral** and **automation**. Within the e-Business on Demand life cycle, decision-making, certain processes and resource allocation can be deferred until the need arises to execute these actions and until more information is available so that the actions can be tailored specifically to the circumstances and the parties at hand. Deferral of allocating resources brings varying degrees of uncertainty. There is a clear trade-off between deferring actions as long as possible to ensure that decisions are as pertinent and up-to-date as they can be, and the risk that delaying a request for allocating resources will result in the resources simply not being available when needed.

There is nothing dynamic about deferring the selection of a business partner as long as possible, only to have to employ a host of lawyers to negotiate the terms of the contract that is needed as the basis of the relationship. In other words, the decision making necessary where choices exist, the processes that have to be performed (selection, exchange of information, negotiations, agreement reaching and contract signing, configuration and instantiation, binding and auxiliary services) and the corresponding allocation of resources must be capable of being executed rapidly. As this requires little if any human intervention, automation is the essential counterpart of deferral if the life cycle is to be made dynamic.

2. THE eBoD LIFE CYCLE

This section presents the e-Business on Demand (eBoD) life cycle (Figure 1) that results from the simplification, deferral and automation approach suggested above. We make a distinction between the **primordial** and the **static** parts, which cannot be automated with current technology, and the **dynamic** part, which can be automated.

The **primordial part of the eBoD life cycle** consists of reaching an agreement within the domain regarding its language and procedures, i.e. defining clients, services and associated contracts as well as the language to talk about them in the different phases of the life cycle. How this part comes about depends on the business and political aspects of each application domain. The **Domain Definition Repository (DDR)** is the placeholder for client-service contract definitions.

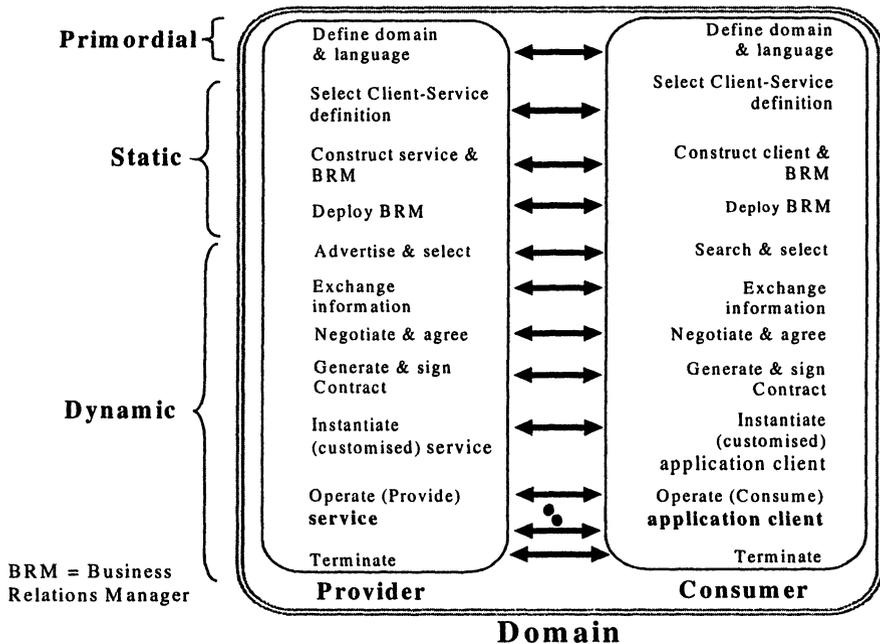


Figure 1 – An e-Business on Demand life cycle and its primordial, static and dynamic parts.

The **static part of the eBoD life cycle** consists of selecting which service and contract definitions are suited to the organisation's needs on the provider and consumer sides. Once one or more types of services have been selected, the components necessary for both parties to manage the rest of the life cycle and create the instances of the service/consumer parts have to be created. This consists of:

- The **service and client application programs**: Each organisation will have to translate the client-service contract into its own world in terms of expertise, components and resources.
- The **business relations manager (BRM)**: A service that can supervise all the activities in the subsequent phases of the life cycle including such things as being able to advertise and search for a compatible business partner, reach an agreement, act as a factory to instantiate the necessary application programs when a contract has been reached, and manage the contract enactment.

The **dynamic part of the life cycle** consists of three major sections and is a candidate for automation with current technology. The first section is the business-driven establishment of a client-service contract. This section covers the search for compatible business partners and the exchange of information that is necessary to reach an agreement and create a contract. The second section of the dynamic part of the eBoD life cycle is the contract-driven client-service instance creation. It entails the dynamic allocation and configuration of organisational resources into the enactment infrastructure and the linking of the infrastructures of the two organisations. The third section is the contract-dependent enactment where the

service and client interact with each other as necessary to provide and consume the service. At the end of the satisfactory provision and consumption of the service, the relationship is terminated.

3. eBoD LIFE CYCLE SERVICES AND INTERACTIONS

This section provides a process view of the dynamic parts of the eBoD life cycle in the context of the on-going business processes of two organisations. It assumes that domain members have agreed upon and carried out the primordial and static parts of the life cycle. Central to this view is the **business relations manager (BRM)**, i.e. the service responsible for carrying out and managing the dynamic phases of the life cycle – the establishment, creation and supervision of the enactment of a business relationship.

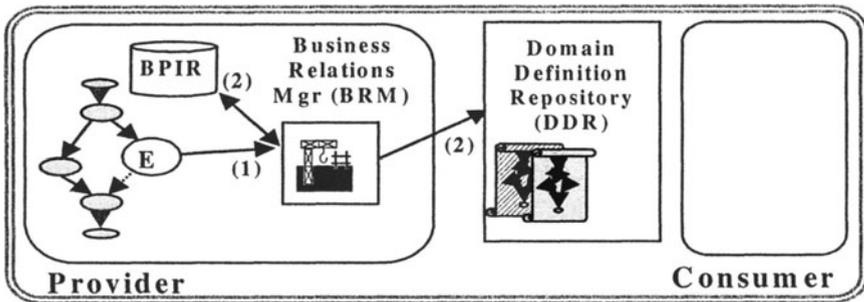


Figure 2 – When a service is ready to be offered externally, the initiating business process uses a business relations manager (BRM). It will acquire the necessary information from the Business Process Information Repository (BPIR) and start the dynamic part of the life cycle of that service.

When a business process activity E on the provider side (Figure 2) determines that it is possible to offer a service defined in the primordial part and selected and constructed in the static part of the life cycle, it either initiates a new BRM or contacts an existing one (1).

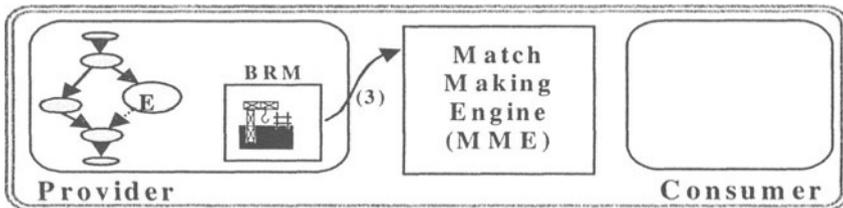


Figure 3 – Having acquired the relevant information about the service to be offered, including the contractual issues, the advertisement can be prepared and sent to the domain matchmaking engine (MME).

The BRM will interact with the organisational management to determine whether the requesting process has the authority to advertise such a service. The

BRM will acquire the necessary information from the business process information repository (BPIR) and the domain definition repository (2) and start the dynamic part of the life cycle of that service.

Part of the client-service definition that the BRM acquires from the domain definition repository (DDR) will contain information about how to represent the service when advertising it to the domain matchmaking engine (MME)¹. Together with information from the BPIR, the BRM in Figure 3 will create the advertisement and submit it to the domain MME (3). The domain MME is the component responsible for matching compatible business partners based on the descriptions and requirements submitted to it.

The process now continues on the consumer side (Figure 4). An ongoing business process reaches an activity C, which recognises the need for an external service. Activity C either initiates a new BRM or contacts an existing one (4). The BRM will interact with the organisational management to determine whether the requesting process has the authority to ask for such a service. The BRM will then acquire the necessary information from the BPIR and the DDR (5). This may also include the address of the matchmaking engine, where the search for a compatible business partner can be conducted. The BRM is then set to start the dynamic part of the life cycle of that consumer.

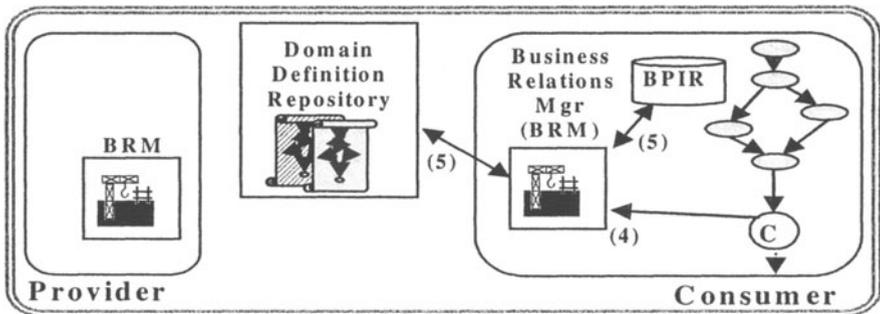


Figure 4 – On the consumer side, when a process needs a service of a specific type, a BRM representing the specific client is engaged. It will acquire the necessary information and start the dynamic part of the life cycle of that client.

Part of the client-service definition the BRM acquires from the DDR will contain information about how to represent the service when querying the domain MME. Together with information from the BPIR, the BRM in Figure 5 will create the query and submit it to the domain MME (6).

The matchmaking process initiated by the query will try to match the request with the available advertisements of the appropriate type. If any of the descriptions of the service contain dynamic information (placeholders for values with information about where to obtain the up-to-date values), and a requirement refers to such a value, it will have to be requested from the provider organisation (7).

¹ The Web Services Matchmaking Engine (WSME) [ALPHAWORKS 02] is an example of an advanced form of matchmaking engine that is suited to the requirements of the eBoD life cycle.

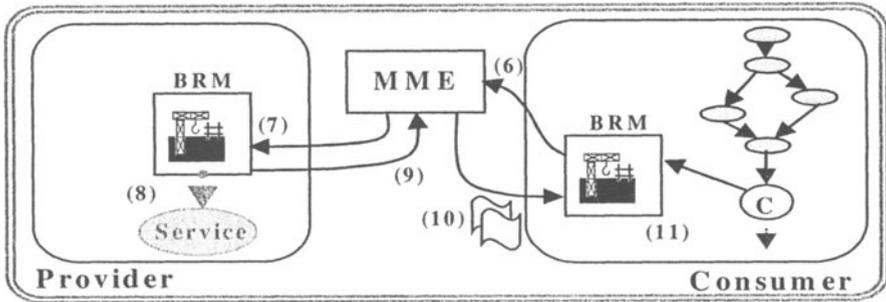


Figure 5 – The BRM sends a query to MME to find matching a service. The matchmaking process may result in a call to the provider organisation to get the up-to-date value of a property describing the service and the provider, thereby enabling the provider to check the availability of required resources. The BRM receives from the MME one or more matching offers.

Such a setup as shown in Figure 5 allows for (8):

- Up-to-date information, which may be dependent on currently available resources;
- The ability to “book” resources needed for the offered service;
- Configuring the service according to the needs of the potential consumer. However, the principle of no surprise must be observed – the offers handed to the consumer cannot be for services that the consumer will not be able to use. So, for example, the interface signature of the service must conform to the type requested. The configuration may be more applicable from the provider’s point of view, i.e. recognising an existing customer, setting up a special deal with reduced prices, etc.

The tailored up-to-date description can be returned to the MME (9), where the matchmaking process can be concluded. Zero, one or more matching offers can then be returned to the consumer (10) for examination (Figure 5). The BRM has to decide which of the compatible offers it wishes to proceed with (11).

Having selected a potential provider, the consumer BRM contacts the provider BRM (Figure 6). An exchange of information and negotiations may ensue (12), depending on how the client-service contract was described in the DDR.

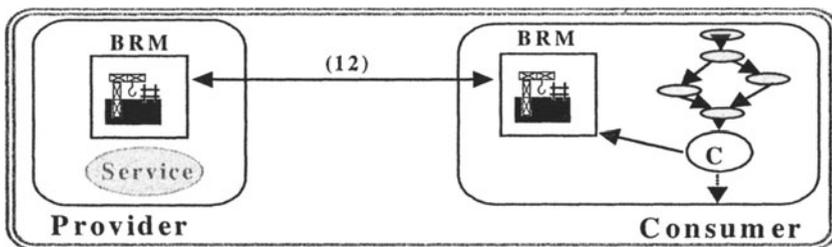


Figure 6 – The consumer BRM contacts the provider BRM and the exchange of information. Detailed negotiations may or may not follow.

The result of the further exchange is an agreement that can be signed in a notary service (13) (Figure 7). The contract should contain the information necessary for performing the subsequent phases of the life cycle.

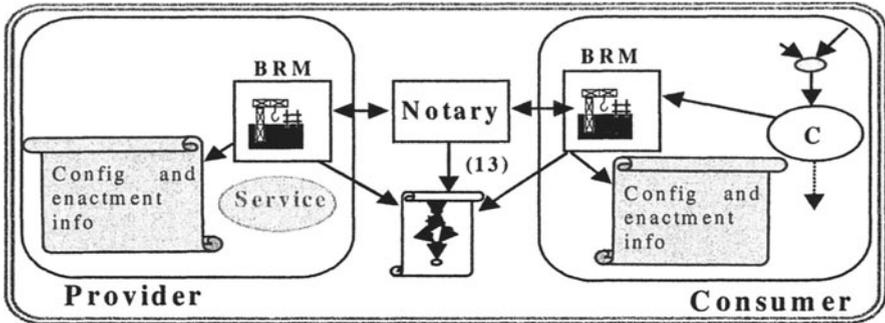


Figure 7 – The consumer BRM contacts the provider BRM; the exchange of information may follow. The result is a contract and some additional information for the configuration and enactment phases.

The full contract is now available to both parties (Figure 8). Further interactions (14) can now result in both parties allocating the necessary resources for providing and consuming the service if this has not happened in any of the previous phases. The provider BRM can then initiate a service instance specific to the consumer (15). Likewise, the consumer can initiate a client proxy (16) and link it to the service (17). This is where the agreed communication protocols of Web services, e.g. SOAP, are likely to play a role. Activity C can then be notified that the external service is ready (18).

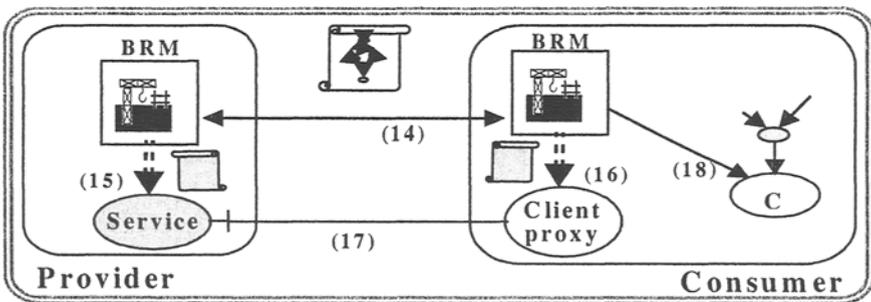


Figure 8 – When an agreement is reached, the provider BRM can create a service instance configured for the consumer, using the information obtained in the previous phases.

Activity C can then interact with the client proxy as if it were the real service (19) (Figure 9). The client-proxy will forward the invocation to the service on the provider side. The client proxy may transform the request, depending on how it was designed and on the difference between Activity C and the service definition in the DDR.

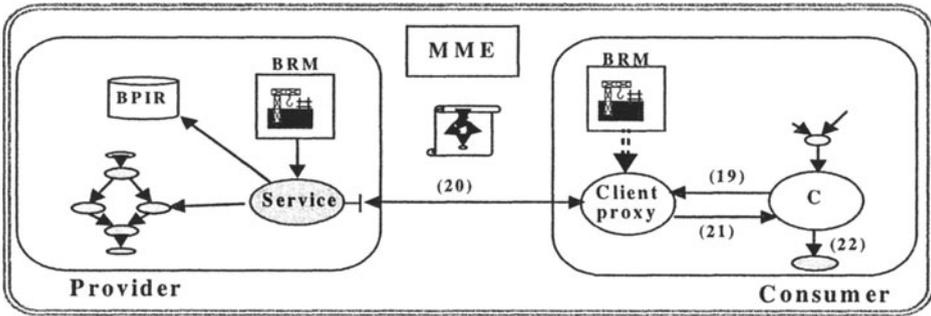


Figure 9 – When everything is in place, the client can request the initiation of the service and interact with it in accordance with the contract.

When the service has been consumed, the link between the two organisations can be withdrawn and the allocated resources reclaimed. The process on the consumer side can then proceed (22).

4. CONCLUSIONS AND FUTURE WORK

We introduced the idea of restricting the life cycle to deal with pre-agreed templates and definitions of client-service relationships. This limits the complexity of the open-ended nature of a business partnership life cycle that involves people in its processes and decision-making points. With this restriction, the parties first have to ensure that they are conversing about the right “type” of relationship. Thereafter, the tasks of matchmaking, information exchange, reaching agreement, creating a contract, configuring and instantiating the appropriate components are simplified by the “no surprise” principle. The restrictive nature of this approach begs the question of how and where is it possible to relax those restrictions to allow more flexibility. One approach is to use building blocks of smaller granularity for the different phases of the life cycle, starting with contract clauses from which it is possible to construct complete contracts. The equivalent modules mirroring the granularity of the contract building blocks must then also be created. Composition rules for creating the necessary entities at each phase of the life cycle must then be specified, and ways of checking the validity of these compositions must be developed. Going from the monolithic approach to the building blocks and composition rules is likely to be difficult but also promises to be highly interesting.

5. REFERENCES

[ALPHAWORKS 02] Alphaworks, Web Services Toolkit, <http://www.alphaworks.ibm.com/tech/webservicestoolkit>, 2002.
 [Alonso 99] Alonso G, Fiedler U, Hagen C, Lazcano A, Schuldt H, Weiler N. “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).

- [Cingil 99] Cingil I, Dogac A, Tatbul N, Arpinar S. "An Adaptable Workflow System Architecture on the Internet for Electronic Commerce Applications". In Proceedings of the International Symposium on Distributed Object Applications. Edinburgh, September 1999.
- [COSMOS 00] Wienberg F, Herz M., The COSMOS Project. <http://www.uib.es/depart/dpr/cosmos.pdf>.
- [CROSSFLOW 00] Cross-Organizational Workflow Support in Virtual Enterprises ESPRIT Project 28635, 2000, <http://www.crossflow.org/>.
- [Dignum 01] Dignum F. "Agents, Markets, Institutions and Protocols." In Agent Mediated Electronic Commerce, The European AgentLink Perspective, Dignum F. & Sierra C. Editors, Lecture Notes in Artificial Intelligence Vol. 1991. Berlin Heidelberg: Springer Verlag, 2001.
- [Field 02] Field S, Hoffner Y. "In Search of the Right Partner." In Collaborative Business Ecosystems and Virtual Enterprises, PRO-VE'02, 3rd IFIP Working Conference on Infrastructures for Virtual Enterprises, Editor: Luis Camarinha-Matos, Dordrecht: Kluwer Academic Publishers, 2002.
- [Grefen 00] Grefen P, Aberer K, Hoffner Y, Ludwig H. CrossFlow: Cross-organizational workflow management in dynamic virtual enterprises. *Int'l Computer Systems Science & Engineering*; invited paper for special issue, 2000; 15(5): 277-90.
- [Griffel 98] Griffel F, Boger M, Weinreich H, Lamersdorf W, Merz M. "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, CA, 1998.
- [Hoffner 01] Hoffner Y, Ludwig H, Grefen P, Aberer K. "CrossFlow: Integrated Workflow Management and Electronic Commerce". ACM: Special Interest Group on Electronic Commerce, Vol. 2.1, Winter, 2001, <http://www.acm.org/sigecom/exchanges>.
- [Hoffner 01] Hoffner Y, Field S, Grefen P, Ludwig H. Contract-driven creation and operation of virtual enterprises. *Computer Networks* 2001; 37: 111-36.
- [Law 98] 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, New Jersey 07101, email: reviser@superlink.net, web site: <http://www.lawrev.state.nj.us>.
- [Milosevic 95] Milosevic Z, Berry A, Bond A, Raymond, K. "Supporting Business Contracts in Open Distributed Systems". In Proceedings of the Second International Workshop on Services in Open Distributed Processing (SDNE95), Whistler, Canada, June 1995.
- [ODP 95] ODP, "Open Distributed Processing Reference Model". ISO/IEC 10476, ITU-T Recommendation X.900, Parts 1–3, 1995.
- [OMG 03] OMG, Object Management Group, "CORBA 3.0: Common Object Request Broker Specification". 2003. http://www.omg.org/technology/documents/formal/corba_2.htm.
- [Vetter 01] Vetter M, Pitsch S. "Towards a Flexible Trading Process over the Internet". In Agent Mediated Electronic Commerce, The European AgentLink Perspective, Dignum F. & Sierra C. Editors, Lecture Notes in Artificial Intelligence Vol. 1991, Berlin Heidelberg: Springer Verlag, 2001.
- [WS 02] Web Services, W3C, <http://www.w3.org/2002/ws/>.
- [WSDL 01] Web Services Description Language (WSDL) 1.1, W3C Note 15 March 2001, <http://www.w3.org/TR/wsdl>.