

NEGOTIATION PROTOCOL CHARACTERISATION AND MECHANISMS FOR VIRTUAL MARKETS AND ENTERPRISES

Yigal Hoffner, Andreas Schade, Simon Field and Christian Facciorusso

IBM Research, Zurich Research Laboratory

8803 Rüschlikon, SWITZERLAND

{yho,san,sif,cfa}@zurich.ibm.com

This paper demonstrates how an extension of the ODP/CORBA Trading Service can be exploited to provide a mechanism for automatic negotiations. The resulting IBM Dynamic MatchMaking Engine can be used to carry out both single and multi-attribute negotiations. A classification of negotiation protocol characteristics based on a linguistic model of human dialogues is presented. An example of a negotiation protocol is given and the classification is used to characterise the protocol.

1. NEGOTIATIONS IN VIRTUAL MARKETS AND ENTERPRISES

Electronic market places and virtual enterprises must be capable of providing different degrees of automation of the process of negotiations. Electronic negotiation involves the orderly exchange of information between the involved parties either directly or through a third party, allowing the parties to modify their offers and counter-offers. Each modification cycle takes into account the previous offer from the other party. The aim of negotiations is to reach a mutually satisfactory agreement, which is in line with the goals of all parties or the “exchange of offers for the purpose of achieving a compromise” (Kersten, 2000).

There is a considerable body of literature on the topic of negotiations (Raiffa, 1982; Lewicki et al., 1997), automated negotiations (Jennings, 2000), and agent based negotiations (Krovi and Graessner, 1999; Bailey and Bakos, 1997), but there appears little in the way of descriptions of the low-level mechanisms required to facilitate negotiations.

To give a negotiation process any chance of success, it is necessary to **agree on a considerable number of issues** and ensure that they are adhered to. For example, the use of a negotiation intermediary, the attributes that are to be negotiated, the language used in the process, the negotiation procedure details such as initiation and turn taking, termination conditions, as well as many others. This paper presents a

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: [10.1007/978-0-387-35585-6_68](https://doi.org/10.1007/978-0-387-35585-6_68)

classification of negotiation protocol characteristics based on a linguistic model of human dialogues.

Furthermore, the process of negotiation requires the establishment of a well-defined procedure and known points of interaction with agreed interfaces. A mechanism that can animate the negotiation process is therefore needed. This paper describes a **negotiation mechanism** and a **negotiation protocol** implemented on it as an example of the issues involved in implementing negotiations. The classification introduced in the first part of the paper is used to explain the negotiation protocol. The issues of negotiation strategy and decision making are outside the scope of the paper. In fact, one of the principles behind the design and implementation of the negotiation mechanism is to make the mechanism itself policy free, while at the same time allowing it to inject market and player agreed policy where necessary.

1.1 Matchmaking and negotiations

Both virtual enterprises and virtual markets require the ability to find matching customers and providers to enable the provision and consumption of the required service. Thus, service providers advertise their services in some offer space (electronic Yellow Pages for example), and customers conduct a search through such a space. This is usually done with the help of some **MatchMaking Engine (MME)**, such as the ODP/CORBA Trading service (ODP, 1995; OMG, 1996) or the IBM Dynamic Matchmaking Engine (Hoffner et al. 1998; 2000).

The **matchmaking process** provides a good starting point for the process of negotiations and can be exploited in different ways:

- Where a mismatch between the description of one party and the requirements of the other party is detected, the point of mismatch can become the subject of a negotiation process.
- A matchmaking process can constitute a preliminary requirement to the start of negotiations by ensuring the minimal pre-requisite compatibility between the parties. This can be used to ensure that parties negotiating some aspect of the goods (for example, price, quantity, QoS or combinations thereof) are first in agreement concerning the type of goods under consideration. In addition, prior matchmaking activity can be exploited to reach agreement of the negotiation protocol details. Further to the success of the negotiation, additional phases of matchmaking can take place.

The negotiation process can thus be triggered at different phases of the matchmaking process:

- Interspersed with the matchmaking activity itself: Whenever a requirement of one party refers to a property of the other that is defined as negotiable.
- After a matchmaking cycle has finished: If it has finished successfully: the established compatibility is used as basis for the negotiation. Otherwise, the point where the incompatibility was encountered must be resolved, if it is concerned with a negotiable property or properties.

Allowing the parties to adapt their positions as part of the matchmaking process therefore represents an extension to a single-step matchmaking scheme. If a third party carries out the matchmaking process (as in the case of the ODP/CORBA

trading service and the IBM Dynamic Matchmaking Engine), indirect negotiation results. The third party controls the negotiation process by requesting both customer and (partially matching) provider to accept the proposal of the negotiating partner, or to continue negotiating by providing a revised offer. In contrast to direct negotiations where the negotiation process is not mediated through a third party, indirect negotiation schemes require the MME to be trusted by all participants. On the other hand, indirect negotiation schemes provide some potential benefits. The central position of the MME allows supervision of the negotiation process to guarantee the fair behaviour of all participants. Furthermore, optimisations in the case of multiple customer-provider scenarios is also possible.

2. THE ODP/CORBA TRADING SERVICE MODEL

The ODP/CORBA Trading Service (ODP, 1995; OMG, 1996) is used to find a service which matches the customer's requirements in cases where the customer does not know the precise identity of the server object but knows some of the properties describing it (e.g., type of service, QoS parameters, etc.).

In the CORBA trading service, the service provider describes what is offered by the service using properties - typed name-value pairs that are either *static properties* or *dynamic properties*. Static properties are service properties that are fixed at advertisement time, while dynamic properties are to be updated at matchmaking time. In the case of dynamic properties, the advertisement provides the information necessary for updating the property during the matchmaking process. Thus, the value of a dynamic property is computed at matchmaking time, which in ODP/CORBA is initiated by the query from the customer.

Service trading or matchmaking works in several steps as shown in Figure 1:

1. The provider advertising the service it provides to the MME. One or more of its properties are marked as dynamic.
2. The customer sends a query request to the MME.
3. The matchmaking process evaluates the constraint expression of the customer against the service properties.
4. When the constraint expression of the customer refers to the dynamic property of the service, the request to resolve the dynamic property with an up-to-date value is sent to the Dynamic Property Evaluator.
5. The value is sent back to the MME, which can now proceed with the evaluation of the constraint expression.
6. The result of the matchmaking process is sent back to the customer.
7. The customer can now choose one of the offers and use the information provided

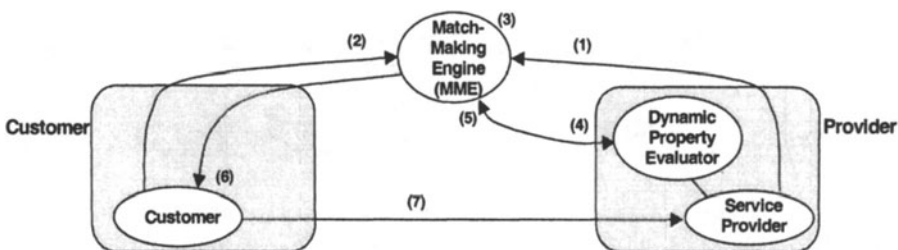


Figure 1 – The ODP/CORBA asymmetric trading process using dynamic properties.

in it to bind to the service provider and use it.

The procedure described above is a unidirectional asymmetric process specified in (OMG, 1996), where offer properties are matched against customer constraints. While the service provider supplies properties describing the service, the customer has no opportunity to describe itself nor is the provider capable of making demands on it by specifying a constraint expression.

3. THE IBM DYNAMIC MATCHMAKING ENGINE

This chapter describes an extension to the traditional ODP/CORBA trading service and shows how the introduction of symmetry to the matchmaking process opens up a number of novel ways in which it can be exploited in different business scenarios.

3.1 Symmetric (Bi-directional) Matchmaking

If customers also submit some descriptive properties and service advertisements also contain constraints on a potential customer, the process can be made symmetric. In the bi-directional matchmaking process, both customers and providers describe what they offer and require of each other. In this case, the service dynamic property update can include information from the prospective customer, to enable the new value to take into account the specific details of the potential customer. Symmetric matchmaking is well adapted to non-commodity markets where providers want to be selective about the customers they do business with (Hoffner et al., 1999 and 2001; Grefen 2000). The IBM Dynamic Matchmaking Engine supports such symmetric behaviour.

The bi-directionality can be further exploited to enable customers to specify some of their properties as dynamic, requiring the matchmaking process to update them at the point at which they are referred to in the service constraint (Fig. 2). As in the provider dynamic properties described above, the dynamic property update can include information from the prospective service provider, to enable the new customer value to take into account the specific details of the potential provider.

Thus, service or product advertising and searching for compatible business partners can be combined with a structured way of modifying what is being offered and what is being requested. This scheme can be further extended to provide a mechanism for negotiations. The ability of the customer and service provider to

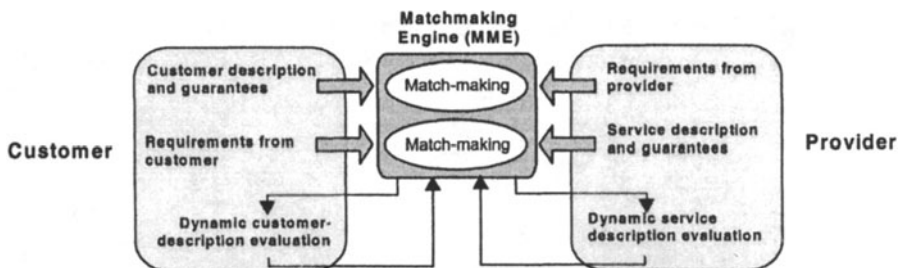


Figure 2 – Symmetric matchmaking process in the IBM Dynamic Matchmaking Engine.

update their values at matchmaking time, taking into account the details of the prospective partner, opens up the possibility of exchanging offers, counter-offers and counter-counter-offers etc., thereby providing a basic and powerful platform for negotiations. A property on the customer and on the provider side which refer, for example, to the price the customer is willing to pay and the price requested by the service provider respectively, can be specified. By extending the dynamic property evaluation further and supporting multiple rounds of updates on each side, the parties have the possibility of updating their demand and supply according to the other party's offer. This can be done on a one-customer-to-one-provider interaction between a single customer and provider or alternatively, on a one-customer-to-many-providers basis after collecting several matching offers and conducting the negotiations in parallel.

By varying different aspects of the mechanism and by modifying the parameters defining the negotiations, the above steps can be combined to provide a mechanism that will support a host of negotiation protocols.

3.2 Using the Matchmaking and Dynamic Properties for Negotiations

The following is a sequence of events, which describes a form of negotiations consisting of multiple cycles of dynamic property updates (Fig. 3), using dynamic customer and provider properties in the matchmaking process. A Dynamic Property Evaluator can be any agent acting on behalf or representing the customer or the

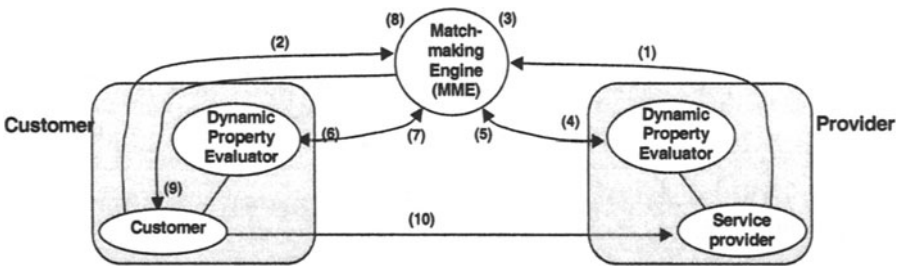


Figure 3 – Using the dual dynamic customer and provider property update facility in the matchmaking process to enable multiple cycles of negotiation.

provider in the negotiation process. It could be a program or a person acting as a manager, for example.

3.3 The Sequence of Events in the Price Negotiation Example

The following is a detailed example of how the proposed IBM Dynamic Matchmaking Engine can be used to implement a specific negotiation protocol. Although the example chosen uses the price as the negotiation attribute, it is possible to use the mechanism to negotiate a QoS parameter or the configuration of some product or service. Furthermore, a combination of attributes can also be used, for example, QoS and Price can be combined and dealt with in conjunction.

The negotiation sequence assumes that the dynamic properties do not provide default values and must be fetched before the negotiation success criterion can be

tested. The previously updated value of the other party is included in the request for dynamic property update:

1. The service provider advertises the offer in the MME by providing it with a description of the provider and its service, and of what it requires from the prospective customer. One of the properties describing the provider is marked as dynamic (*provider.PayRequest*). The property marked as dynamic may contain:
 - Address of the relevant Dynamic Property Evaluator – where up-to-date information about the dynamic property can be obtained.
 - A parameter list of properties from the other party as input: for example, the name of the potential customer so that the required property can be made customer specific.
2. The customer sends the MME a search query with its requirements together with a description of what it offers to a prospective service provider. One of the properties sent is marked as dynamic (*customer.PayOffer*). The parameters needed to evaluate the customer dynamic property may require some of the properties of the provider. A customer or provider requirement can state the relationship between the two dynamic properties that is to form the success criterion for the negotiation. The parameter needed to evaluate what the customer is willing to pay is the provider's name. A negotiation mediator (the MME in this case) requirement can state that the *provider.PayRequest* \leq *customer.PayOffer*
3. The MME checks the requirements of the customer against descriptions of services advertised previously. Similarly, it checks the requirements of each service provider against the description of the customer.
4. When it comes across a service property marked as dynamic, it sends a request for update of the property from the service provider Dynamic Property Evaluator. In the case of the *provider.PayRequest*, the MME will provide the required customer property as the parameter of the dynamic property update request. The service provider Dynamic Property Evaluator can calculate the customer specific value of the dynamic property
5. The service Dynamic Property Evaluator sends the newly evaluated property to the MME.
6. The MME can now resolve the *customer.PayOffer* dynamic value. It does so by calling the customer Dynamic Property Evaluator providing the service name. The customer Dynamic Property Evaluator can calculate the provider specific value of the dynamic property *customer.PayOffer*.
7. The customer provider Dynamic Property Evaluator can then send the new *customer.PayOffer* to the MME.
8. The MME now tests the condition *provider.PayRequest* \leq *customer.PayOffer*, having resolved both dynamic properties.
9. If the success criterion is not met, the negotiations have not reached a satisfactory state; the MME can then decide whether to carry out another cycle according to some pre-agreed termination criterion (imposed by the MME or as requested by the parties).

If a match is not successful, the MME decides to carry out another cycle (the termination criterion was maximum 3 cycles). If it decides to proceed with an additional cycle, it will go back to step (4). If the match is successful after 2

cycles, the MME will add this offer to the collection of matching offers. The returned offer will include a description of the service, which matched the customer’s requirements.

10. When the negotiation cycle(s) come to completion, the customer can now choose

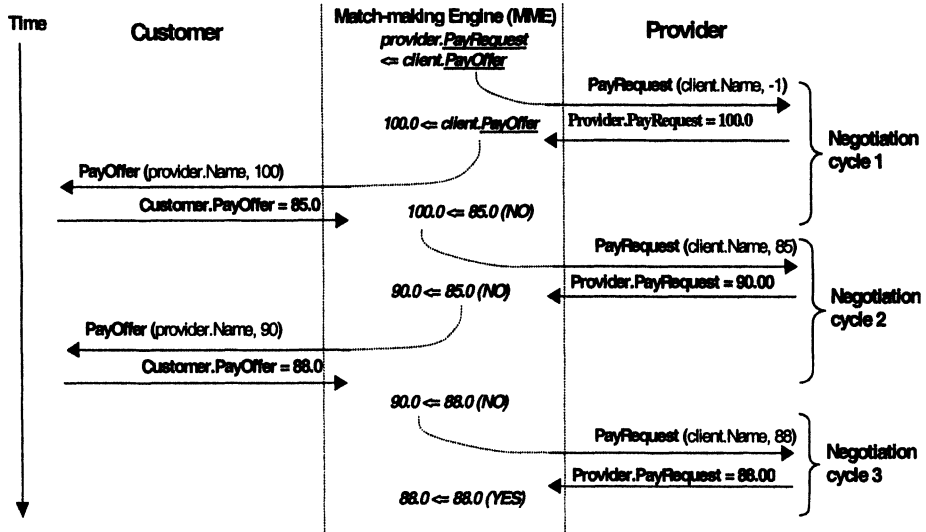


Figure 4 – The sequence of events of the negotiation protocol. The up-to-date property from the other party is sent with the request for update.

one of the offers and use the information provided in it to bind to the service provider and use it.

The time dependencies of the negotiation protocol associated with this sequence are shown in Fig. 4; the numbering of the steps follows the sequence shown in Fig. 3 starting from point (6) and shows an additional negotiation cycle.

4. NEGOTIATION PROTOCOL CHARACTERISATION

Negotiation protocols will vary from each other depending on the mechanism used, on the subject of the negotiations, the circumstances surrounding the negotiations and many other factors. Nevertheless, certain characteristics of negotiations will have to be defined and adhered to regardless of the differences. This chapter presents a high level view of negotiation protocol characteristics.

4.1 Classification of Negotiation Protocol Characteristics

A considerable level of agreement is necessary to allow meaningful negotiations to take place. The classification presented below, based on a linguistic model of human interaction (Greene, 1986), outlines the areas where agreement is necessary. The classification also shows (*in brackets in italic*) how the example presented in the previous section can be mapped to the classification:

1. **Participation:** who participates in the interaction - two-party and multi-party interaction. (*customer, provider and MME as mediator.*)

2. **Universe of discourse, ontology and language:** what is being discussed and what is the language used to discuss it with. Is a single issue being discussed at one dialogue or more? A third party can also act as a translator.

Attribute(s): a specification of what is negotiated, for example, a QoS and/or Price attribute combination. Two options exist: single or multiple attribute negotiations. The attributes must be defined in terms of the (type, permitted range, units, etc.). (*provider.PayRequest (Real) and customer.PayOffer(Real)*)

Information exchange: this includes the negotiation attributes and any other information necessary for the parties to make a decision. In distributed system terminology the signature of the negotiation interface of the two parties defines this:

- Input attribute(s) such as information from the other party or from the mediator.
- Expected attribute(s) to be returned and their associated value type(s). (*Provider: Real PayRequest (String customer.Name); Customer: Real PayOffer (String provider.Name)*)

Point of contact or interface: point of negotiation handling in all the concerned parties. (*customer and provider Dynamic Property Evaluators (rmi interfaces)*)

3. **Purpose:** this relates to where the negotiation starts from, where it is going in terms of its aim, and the direction of interaction and termination condition(s):

Initial conditions: which party should be supplying what information at the start of the process. (*none*)

Success criteria (terminating condition): defines when the goal of the negotiating process is achieved. The customer and provider may specify different criteria but in order to have the possibility of successful conclusion, their criteria must have some overlap. (*provider.PayRequest <=customer.PayOffer*)

Failure criteria (terminating condition): defines when the process should be terminated in spite of not reaching the goal. (*maximum 3 negotiation cycles*)

Convergence criteria: if the protocol allows for multiple negotiation cycles, this defines the direction of change necessary to allow the continuation of the process. This may not be possible to provide in all cases. Care should be taken that the order of dynamic property updates does not create a deadlock situation. (*none*)

4. **Communication medium and mode:** this determines the flavour of the interaction from a communication point of view and includes:

Direct or mediated negotiation: the negotiation can be made face to face or be mediated by a third party. (*MME mediated*)

Mediator intervention level: The level of intervention that the mediator can exercise with regard to matchmaking and negotiation process may vary. (*none*)

Communication mode: exchanges can be conducted in real time or can contain extended pauses. In terms of the implementation this makes a distinction between remote procedure call and queueing style of interaction. A queueing mode allows human intervention in the negotiation. (*synchronous*)

5. **Procedure and feedback:**

Provision of initial default values: defines whether the negotiation starting position values are provided at advertising and searching time or whether there should be at least one dynamic update cycle on both sides. (*none*)

Process triggering: defines **how** the negotiation condition is specified and **when** the process is triggered during the matchmaking process. There are two general approaches:

- As part of the matchmaking constraint expression that specifies the demand one party is placing on the other. For example:
customer.propertyX < provider.propertyY
- As a separate entity from the matchmaking constraint expression. This allows making a distinction between the normal evaluation of a constraint expression and the negotiation part, making the triggering of the negotiations more flexible. Whether this expression comes from one of the parties or from the market itself through the negotiation mediator depends on the type of the market. (*a separate expression from the constraint expression processed after constraint expression has been evaluated*)

Interaction format: Turn taking issue, exchange quantity and closure, gesturing as state indication. (*reply to rmi call carries updated value*)

Structure, order and sequencing: sequential or parallel attribute updates and the order of updating. (*sequential with provider providing offer first, followed by customer*)

6. **Openness and fairness:** the level of exposure to other parties or in the case of third party controlled negotiations, the exposure of offers to the other party can vary. This involves issues of fairness and privacy and the extent to which the third party can observe and audit the exchanges. (*reveal updated value to other party but success and convergence decisions are left to the MME*)

4.2 Agreeing on a Negotiation Protocol

No single negotiation protocol or implementation is likely to be applicable to all cases and therefore many different protocols will exist. In order to enter a negotiation process between two parties, the parties must first agree on the protocol that will form the basis of the negotiation process. This requires agreed ways to specify protocols; this can then be used to agree on a protocol in the first place and thereafter use it to negotiate. The MME can be exploited to achieve this by matching the set of negotiation protocols offered by the two parties. Thereafter, the same MME can be exploited to carry out the selected protocol.

5. SUMMARY AND CONCLUSIONS

This paper demonstrated how an extension of the ODP/CORBA trading service could be exploited to provide a mechanism for automatic negotiations. The ODP/CORBA asymmetric matchmaking process was extended into a bi-directional symmetric matchmaking process where a constraint expression on both sides can refer to the properties of the other party. Thus, dynamic property evaluation can be carried out on both sides, using relevant information from the other party. By extending the dynamic property evaluation further and supporting multiple rounds of

updates on each side, the parties have the possibility of updating their demand and supply according to the other party's offer. Using this scheme, service or product advertising and searching for compatible business partners can be combined with a structured way of modifying what is being offered and what is being requested. The ability of the customer and service provider to update their values at matchmaking time, taking into account the details of the prospective partner, opens up the possibility of exchanging offers, counter-offers and counter-counter-offers, thereby providing a basic and powerful platform for performing negotiations in an automatic manner. This can be done on a one-customer-to-one-provider interaction between a single customer and provider or alternatively, on a one-customer-to-many-providers basis after collecting several matching offers and conducting the negotiations in parallel.

The scheme can be used to carry out single attribute negotiations but can also be exploited for multi-attribute negotiations. This allows trade-off to be made between the different attributes. For example, QoS and price can be trade-off against each other. The complexity of the algorithm for the success and for the convergence criteria rises with the number of attributes, if any trade-off among them is desired. This raises the need for a complex language that can describe the relationships between the attributes.

Care should be taken to avoid deadlock that may occur if a dependency cycle among the negotiation parameters exists. In such a situation, each side may wait for the other side to update its value based on the other party's updated value.

Future work is envisaged in the area of applying different negotiation policies through the mediating Matchmaking Engine.

6. REFERENCES

- Bailey JP., Bakos Y., An exploratory study of the emerging role of electronic intermediaries. *Int'l J Electron Commerce* 1997; 1:7-20.
- Greene J., *Language Understanding: A Cognitive Approach*, Open Guide to Psychology. Open University Press, 1986.
- Hoffner Y., Schade A., "Co-operation, Contracts, Contractual Matchmaking and Binding". In Proc. 2nd Int'l Enterprise Distributed Object Workshop "EDOC '98," pp. 75-86. Piscataway: IEEE, 1998
- Hoffner Y., Facciorusso C., Field S., Schade A., Distribution issues in the design and implementation of a virtual market place. *Computer Networks* 2000; 32:717-30.
- Hoffner Y., Field S., Grefen P. and Ludwig H., "*Contract-Driven Creation and Operation of Virtual Enterprises*", *Computer Networks, The International Journal of Computer and Telecommunications Networking*, North Holland, Volume 37, pp. 111-136, September 2001.
- Kersten GE., Modeling Distributive and Integrative Negotiations. Review and Revised Characterization <http://interneg.org/interneg/research/papers/>, 2000.
- Krovi R., Graesser AC., Agent behaviour in virtual negotiation environments. *IEEE Trans Systems, Man, and Cybernetics (Part C: Applications and Reviews)* 1999; 29: No. 1.
- Lewicki R, Saunders D, Minton J. *Essentials of Negotiation. PLACE: Irwin McGraw-Hill*, 1997.
- ODP – Open Distributed Processing Reference Model, ISO/IEC 10476, ITU-T Recommendation X.900. 1995. Parts 1–3.
- OMG – Object Management Group and X/Open., CORBA Trading Object Service, Document orbos/96-05-06, Object Management Group and X/Open, 1996.
- Grefen P., Aberer K., Hoffner Y and Ludwig H., "*CrossFlow: Cross-Organizational Workflow Management in Dynamic Virtual Enterprises*", *International Journal of Computer Systems Science Engineering*; invited paper for special issue, vol 15, no. 5, Sept 2000.
- Raiffa H., *The Art and Science of Negotiation. PLACE: Harvard University Press*, 1982.