

Modelling a Business To Business Intermediation Platform

A. K. Kaltabani, M. A. Lambrou, G. T. Karetsos, M. E. Anagnostou
National Technical University of Athens, Department of Electrical Engineering and Computer Science, 9, Heroon Polytechniou Str. 15780 Zografou Athens, Greece, e-mail:{akalt, marial,karetsos,miltos}@telecom.ntua.gr, Tel: +30-1-7721512, Fax: +30-1-22534

Abstract: Our proposition concerns an architectural model of a broker assisted, business to business e-commerce system. The presented brokerage platform is designed against business and technical requirements, within the transformational B2B landscape, which might push forward either centralised or proliferate, symbiotic brokerage models or even render them all invalid. In our paper, we merely elaborate on the design of a modular, scalable and extensible component-based brokerage system for complex, B2B services transactions. To this end, firstly, a particular conceptualization of a complex services transactions lifecycle is presented. Secondly, the functionality of the components comprising the designed brokerage platform is provided. Thirdly, the interactions among the identified components are given for indicative service procedures. The derived architectural model is proposed as applicable to various business domains in the aid of efficient business to business processes alignment.

1. INTRODUCTION

In this paper, we present a particular brokerage platform, compliant to business to business models and respective systems implementation requirements, that is considered applicable across a number of diverse business domains. The main proposition of our work is a generic, modular, scalable and reusable architectural model of the envisaged business to business brokerage system, enabled by an informed and comprehensive examination of the requirements and determinants shaping the electronic intermediation and business to business arena, nowadays and in the near future. Examining and questioning dominant trends and propositions concerning brokerage and B2B models and systems, enable the presented results.

Moreover, the specific architectural approach is derived by correspondingly exercising the concepts, methodologies and practices of the service engineering domain.

Our paper is structured as follows: In section 2, we overview the evolution of the electronic intermediation phenomenon and as related in particular with emerging business to business models, services and systems. In section 3, a description of the brokerage functionality is given, whereas, in section 4 the detailed architectural aspects and tasks are addressed, for presenting the static and dynamic view of the brokerage platform. Finally, in section 5, conclusions and discussion on the future work are given.

2. THE EVOLUTION OF ELECTRONIC INTERMEDIATION AND BUSINESS TO BUSINESS TRANSACTIONS SYSTEMS

In e-commerce literature, the overall electronic intermediation theme, as well as its relevance and relationship with particular business to business issues, have attracted considerable attention and generated quite comprehensive analyses. Equally important experimentations and commercial exploitation of respective software systems [1, 2] have been, in parallel, achieved.

A number of [3,4,5,6,7] authors report that business transactions, over communication networks and in particular over Web, typically involve all types of intermediaries, not just the traditional wholesalers and retailers, but specialised content providers, search engines, affiliate sites and networks, Internet service providers, the backbone providers, software makers, the advertising networks, and many other entities and players to be generated.

Obviously electronic commerce practices and system capabilities can simultaneously allow suppliers to suppress one or several intermediaries in a distributed network (because of a public shared infrastructure and a direct contact), but it can also provide the opportunity to create new intermediaries in some domains by the provision of added-value in classifying, integrating or managing information and services, among others. Bakos [4,8] hypothesizes that large scale globally distributed intermediaries, formed by industry participants in collaboration with IT companies, will emerge in marketplace. Either by capturing dominant market share in a single industry or by becoming electronic market makers across a number of industries such intermediaries will be capable of sustaining a competitive advantage by securing economies of scale and scope.

First generation electronic brokerage systems, applicable both to business to business and business to customer markets, exploited the value propositions stemming from three formative roles [1,4,9]:

1. *aggregation* of buyer demand and seller offers (products, services), thus enabling economies of scale and scope and normalizing the negotiation power of the involved players, within a market.
2. provision of an enhanced market institution and platform of *trust* for the overall services transaction.
3. *facilitation* to the market and involved players by streamlining the information exchange and the coordination of the respective processes. Consequently, perform an efficient *matching* of the complex specifications of the demand and supply side characteristics.

Organisations and e-commerce systems that employed successfully, most often in an integrated manner, the aforementioned aspects and as conditioned by the specificities of the markets (consumer, industrial) within which those organisations operate, represent the early phase of the e-brokerage models. It is that brokerage model that also critically shaped the evolution of the business to business landscape, in particular. This brokerage model is often quoted and interpreted as the “exchange-like model”, gaining insight, in specific, from the financial markets evolution [5, 10]. Currently, we already observe clear trends towards a fragmentation of the offered brokerage services, beyond the above described model, as provided by distinct, complementary or symbiotic second-generation brokerage, business to business models. Such business models, and following up the analysis of Wise et al [Fehler! Textmarke nicht definiert.], can be viewed as including:

- a) *mega-brokers* that may act as central hubs or intermediation platforms for the execution of most transactions and for buyer – supplier communication and coordination, resembling the classical e-broker.
- b) *specialist originators* that automate, in particular, the buyer decision making process, which is especially vital for complex products and services, and subsequently channel the transactions to the brokerage platform for execution.
- c) *solution providers* can operate separately from open hubs by embedding the product sale into a suite of valuable and thus indispensable services.
- d) *sell side asset brokers* that operate in a rough compliance with a peer to peer paradigm, in which suppliers can trade among themselves, some times after initial transactions with buyers are made on the central hubs or mega brokers.

It is in this very same context, however, that the notion of “napsterization of business to business transactions” appears to considerably challenge the current evolution path, as above described. Following the McAfee argument [9], if companies can complete complex transactions among themselves through peer-to-peer networks and interoperable complex business processes software, the need for centralized brokerage systems decreases dramatically.

Against this background, we subsequently present a particular business to business brokerage platform, and we argue that, although it encapsulates the *centralized-broker* or *mega broker* design principles, it is extensible to a number of the evolutionary, proliferate brokerage models such as the *solution provider* or the

sell side asset broker paradigms. Its potential accommodation to a dominant peer to peer, business to business transaction landscape will be subject to further study.

3. DESCRIPTION OF THE BROKER ASSISTED E-COMMERCE TRANSACTIONS SYSTEM

In this section we present an outline of the proposed brokerage platform, driven by the specific requirements of the involved actors determining its conception, use and operation. This is primarily achieved by envisioning the construction of a generic, reusable and scalable business-to-business (B2B) brokerage framework, and by a conceptualization of the complex service transaction lifecycle and its particular functional decomposition.

In the proposed complex services transactions model we consider three basic domains that is the User Domain, the Service Provider Domain and the Broker Domain, which actually represents the specific brokerage platform domain. The particular interactions that take place among these three domains mirror our understanding and approach to a complex e-commerce service transactions lifecycle and its refined phases. In more detail we design the following phases:

- The request formulation and solutions pre-processing
- The solutions identification and bilateral negotiation
- The solutions evaluation
- The transaction execution and payment

Apart from the functionality necessary for executing the above phases' processes, in the broker domain, the complex knowledge representation is handled by means of Ontology technology [11] to efficiently represent and manipulate the static and dynamic aspects of complex services semantics in different business applications. The presented overall design approach caters for a tripartite and synergistic solution building and negotiation protocol, as is illustrated in the following Figure.

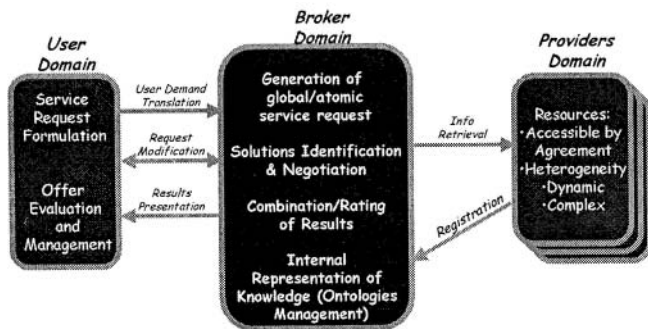


Figure 1: The Brokerage system design approach

The proposed system [12] will be able to handle complex services composed by a set of simple services. A simple or atomic service is defined as to be provided as a whole by an external, affiliated provider, so the system handles to a certain extent the internal complexity (if any) of that service, thus, allowing for a strong provider perspective. The notion of a complex service is also introduced so as to ensure that the specific brokerage platform can manipulate a wide range of bundled services, and thus, at the same time, complying with a customer-centric approach. In the sequel, the functionality of the complex services transaction e-commerce system is described.

After a customer has successfully accessed the system, he/she may construct requests by browsing a hierarchical menu of provided services, in specific business domain(s). The menu presented to a given user is built dynamically and is personalized based on the knowledge of the user's profile and the history of his previous requests. The platform interactively guides the user in the definition of the desired complex service. The knowledge required to dynamically formulate a complex service request that is compliant with the system is provided by means of the Ontology technology capabilities supported by the platform. Upon the user request validation by the system and the confirmation by the user, the system will invoke its internal brokering functionality in order to find an acceptable solution. In the case that several offers are found, the system proceeds with their evaluation, based on the customer preferences indicated in his profile, or other criteria.

When the solution identification phase is completed, the system will present a solution to the customer. The solution may be presented in several ways depending on the customer preferences. The platform is designed so as to support the cooperation with and servicing to three generic classes of providers that are currently met and even anticipated to operate in the e-commerce arena, in the near future. In specific, a brokerage system capabilities as the one described in this paper, can be exploited by service providers that allow for an extensive and online access to the brokered resources (affiliate service providers), and to the ones that allow for a

limited access and consequently limited automation in service delivery (on-line service providers). It may also be used, in a minimalist sense though, by services providers that do not actually run any e-commerce server. The customer of the described brokerage platform, hereafter denoted as user, may redefine the attribute values of simple services for which no satisfactory offers have been found, e.g. increase the maximum acceptable price, modify delivery dates, or other terms of service provision. In this case, the system will validate the updated request and perform a new solution identification process, until a satisfactory solution is found.

Once the proposed solution has been accepted, a corresponding service reservation takes place and the system proceeds with the subsequent phase that is the transaction execution and payment phase. In this later phase, the platform performs the necessary interactions with the service provider domains in order to actually proceed with e-commerce services provision and delivery, which were during the previous phases successfully brokered and reserved. The interactions taking place in this later phase comply with the all-or-nothingness principle that is the complete service reservation will be delivered by the appropriate provider.

3.1 Federation aspects of a brokerage system

One aspect that needs to be addressed when designing a brokerage system is its capability to federate with other such systems specialized in similar, complementary, or ancillary business domains. This is important for the system's scalability whenever a requirement for expansion either in geographical or in business terms emerges. Scalability is also critical when we come to the system's robustness and business value, since users can take advantage of Service Providers (SPs) not directly attached to their home broker and thus the probability to satisfy their demands is increased.

The establishment of a federation among brokers poses a number of technical requirements. First and foremost, a federation agreement should exist between any pair of brokers that are willing to cooperate and seamlessly support their attached users. This is achieved through a mutual registration procedure and knowledge exchange. Second, a query redirection and result handling mechanism should exist that takes care of the information exchanges between the home broker and the remote ones. Third, each broker should update its information regarding its federated brokers either after a predefined interval or whenever a major change has occurred in one of each federated brokers.

The system we describe can relatively easily support federation with enhancements in the Ontology Manager's database to care for the federated brokers and the corresponding remote SPs. However, a detailed analysis of the exact federation mechanisms falls out of the scope of this paper, being an issue of our future work in the area of the design of brokerage services.

Finally, it is essential to note that the introduced federation metaphor and its corresponding technical mechanisms are seen as particularly meaningful in the current transformational phase concerning the operation of robust and advantageous brokerage, business to business models, as discussed in section 2. Such a federation policy might apply to the presented system, in its capacity either as a mega broker cooperating with a broker of a similar scope, in the same or a complementary business domain, or as a proliferate brokerage model in synergy with its complementary ones.

4. ARCHITECTURAL MODEL OF THE BUSINESS TO BUSINESS BROKERAGE SYSTEM

In the following, we present the design of the proposed brokerage system architecture, as being enabled by the exploitation of the service engineering framework [13, 14, 15, 16, 17] concepts, methodologies and practices. More precisely, in the sequel, we elaborate on the tasks and results of the identification of the system architecture components, and the definition of the component interfaces and the component interactions, falling within the service analysis and service design phases of the above framework [13]. To this end, we adopt a two-fold approach. First, in sub-section 4.1, we illustrate the brokerage architecture components, that is a static view of our system, while in sub-section 4.2, we elaborate on these component intercommunications, providing a dynamic view of the system.

4.1 Component based system model

In this section, we focus on the description of the functionality of the main components' that our brokerage system comprises [18]. Each one of these components is further analysed in the next paragraphs. As illustrated in Figure, the components that have been identified are the following:

- The *User Manager* component that is the first access point for the user to the brokerage system. It carries out all the managing user oriented activities and collaborates with the Ontology Manager for formulating the service requests.
- The *Provider Manager* component that participates in the solution building and actually keeps information for the service providers themselves and the services they offer.
- The *Solution Builder* component that is responsible for formulating a service solution corresponding to the service request by collaborating with the User Manager, the Provider Manager and the Ontology Manager.

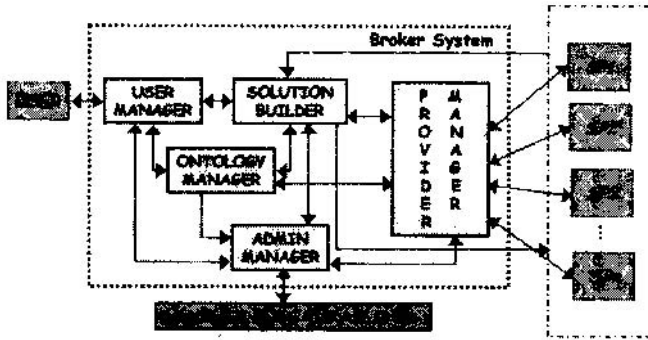


Figure 2: Main view of the system architecture

- The *Ontology Manager* component that handles the interactions of the brokerage system components with the ontologies that model the knowledge of the complex services.
- Finally, the *Administrator Manager* component that is responsible for the control of the overall brokerage system.

Focusing on the User Manager we identify the components that are illustrated in Figure . First of all, an interface abstraction layer that comprises the *User Interface Handler* and the *Graphical Manager* enables the connection of the user to the system and is responsible for handling the information exchanges between the customer and the rest of the components of the system.

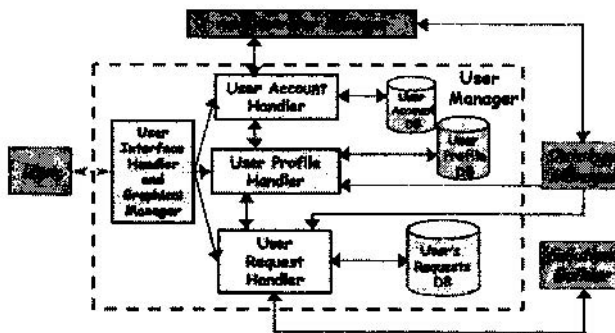


Figure 3: User Manager

The user is able to perform both managing and transaction-related activities through our brokerage system. Regarding the *managing activities*, the customer is able to create, modify or delete his account that consists of a username and a password through the *User Account Handler*. This refers to the users who have a full subscription to the system and take full advantage of it. Other types of users, having various levels of access and use rights to the system, are also foreseen. Moreover,

the user profile that keeps not only personal information about the user but also his own preferences and constraints related to the complex services is managed through the *User Profile Handler*. As it is defined in [Fehler! Textmarke nicht definiert.] , the complex services are described in terms of ontologies, and the User Profile Handler retrieves from them the necessary elements for service provision handling.

On the other hand, the *transaction-related* activities take place when the customer expresses the need for a complex service by means of a suitable request. The aim of the *User Request Handler* is to manage all interactions with the user during and all along the procurement of a complex service. That means to help the user expressing his request as dialoguing with the *Solution Builder* and the *Ontology Manager* as well as modifying the request when no satisfactory solutions are found.

Symmetrically to the User Manager, within the Provider Manager the following components are defined (see Figure). Firstly, as in case of the user side, there is a need for a *Provider Interface Handler* and a *Graphical Manager* that allows each service provider to be connected to the system.

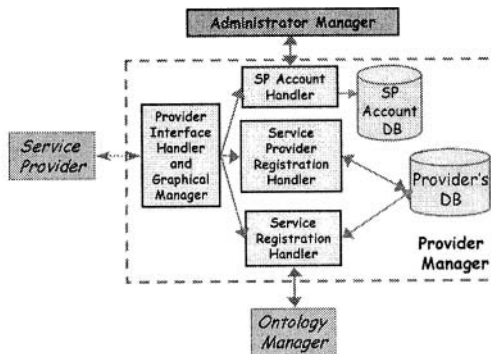


Figure 4: Provider Manager

The *SPAccount Handler* is responsible for creating and handling (e.g., modifying, deleting) the service provider account while all the related elements are kept in the proper database. Moreover, all the mandatory parameters related to SP registration data and required by the system (e.g., company data, business sector, type of services the SP can provide) are managed by the *Service Provider Registration Handler*. Finally, the detailed service content registration is handled by the *Service Registration Handler*. In specific, there is a co-operation between the *Ontology Manager* and the *Service Registration Handler* whenever an alternation to the provider offers' takes place. The main concern is the description of the provider offers in a way coherent with the rest of brokerage platform, i.e. relate the service provider's offer to the right element of the ontology.

The Solution Builder represents the core functionality of the overall system. The components that are identified are depicted in Figure. As already mentioned, the complex service consists of atomic services that may be provided by more than one

service providers. In this sense, the *Solutions Pre-Processor* component is responsible for the identification of the candidate service providers that may offer solutions according to the user service request. This is achieved with the participation of the Provider Manager form where complete information about the SPs' offers is obtained.

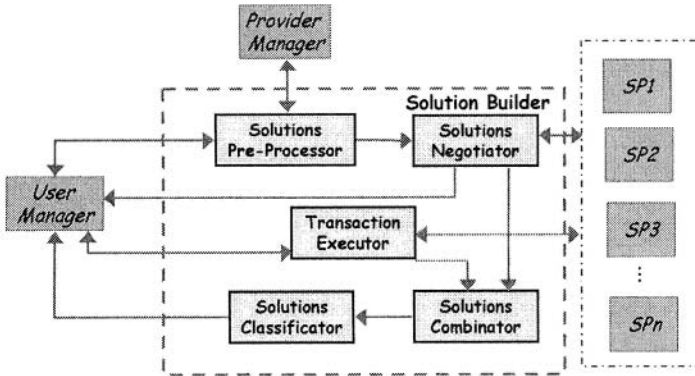


Figure 5: Solution Builder

Based on the constraints set by the user, the solution negotiation takes place by the *Solutions Negotiator* component that is designed in order to identify the actual service solutions with a close collaboration with the specific service providers. The role of the *Solution Combinator* component is to group the various solutions according to the system's business criteria. Moreover, the *Solution Classifier* component is responsible for the evaluation of the service offers according to the user defined criteria (e.g., price, delivery date etc). The customer himself is the one that takes the final decision either to accept or to reject the proposed solutions. The user has also the ability to modify his request if the solutions that are presented to him are not still satisfactory. Then, a new negotiation phase takes place. If the user accepts the offer of his complex request, the transaction execution begins which is handled by the *Transaction Executor* component that encapsulates the logic necessary for the transaction confirmation and payment. Specifically, the user reserves the offer and then, the service providers are contacted in order to confirm that the offers are still valid. If one of the reservations fails, the whole transaction is aborted. In the normal case, where all the reservations are kept for the specific user, the payment procedure takes place and when it is successfully completed, the system informs the user that he will be offered, provided the requested service.

4.2 Brokerage system component interactions

Following, the dynamic model of the proposed brokerage system is provided by highlighting the main procedures that take place during the complex service transactions lifecycle. We emphasize on the solution building that is described in terms of message sequence charts (MSCs) while the rest of the procedures are described in less detail. Firstly, the User and the Service Provider Registration procedures are explained.

In the first one, the User Domain, the User Manager and the Administrator are involved. A user registration request is completed by means of two main processes: (a) *The User Account Management* in which the definition and management of user access rights and system use for all types of users (i.e., registered, identified and anonymous) is carried out and, (b) *the User Profile Management* process, in which the definition and management of general properties, personal data and transaction preferences for all types of users is carried out.

Secondly, in the Service Provider Registration procedure the Service Providers domain, the Provider Manager as well as the Administrator domain are involved. Apart from the *Provider Account Management*, which is designed in a symmetrical and reusable form with respect to the user respective part, there is also the *Provider Registration Management* process. Specifically, this process deals with the definition of the terms of the access agreement with the brokerage system for the affiliate providers, as well as the definition and management of the offered service attributes for all types of providers. The affiliate providers provide all the information (i.e., complete service attributes and values offer) to the system, while the online and offline providers give only a subset of this information in the system's catalogue form.

In the following Figure, the dynamic model of the Service Request Handling, the Solution Negotiation and Evaluation is highlighted. Specifically, after being authenticated into the system, the user is able to formulate an informed complex service request as based on previous existing demands which can either reuse or modify or build a new request from scratch. In the first case, the complex service request being reused is validated with the collaboration of the User Manager, the Ontology Manager and the Solution Builder. In the case of a request modification, the detailed atomic services, subsequently formulated, are validated (updated) with the collaboration of the User Manager, the Ontology Manager and the Solution Builder. In the later case of building a new request from scratch, the user formulates his request based on the current complex service template that the Ontology Manager forwards to the user side. The next step is the retrieval of the candidate service providers identities that offer solutions corresponding to each atomic service request. This is achieved by means of interaction between the Solution Builder and the Provider Manager.

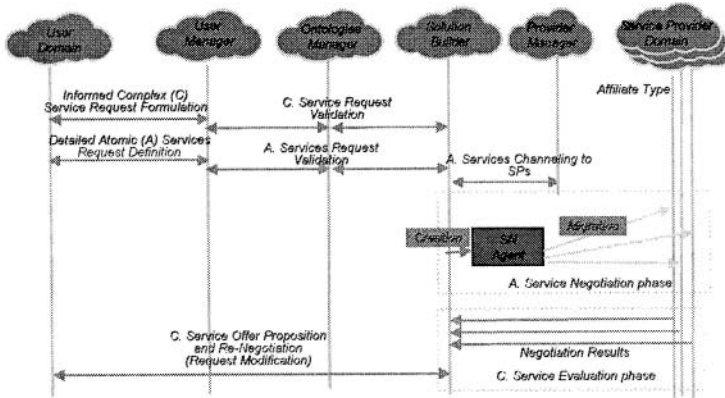


Figure 6: Complex Service Request Handling Procedure

Focusing on the processes that comprise the solution building, we provide the message sequence charts in Figure and Figure, in which the detailed interactions among the identified service components are shown. In this respect, we start with the solution negotiation that begins when the *Service Request Validator* (SRV) which is a part of the *Solution Pre-Processor* component sends the corresponding message to the *Negotiator Controller* (NC) (*Negotiate Service Request()*) who in turn creates and instantiates a *Single Negotiator* (SN) for each atomic service. The SN is created as an agent that migrates to the identified *Service Providers* SP_{ik} machines (where the SP_k provider offers the i type of atomic service, $i=1,..n$ and $k=1,..m$), in order to get the exact service offers for each atomic service. The *GetServiceOffer()* method is invoked and for the affiliate providers a full offer containing the requested static and dynamic elements may be found. Regarding the online or offline providers there is also the need to be contacted offline by the user.

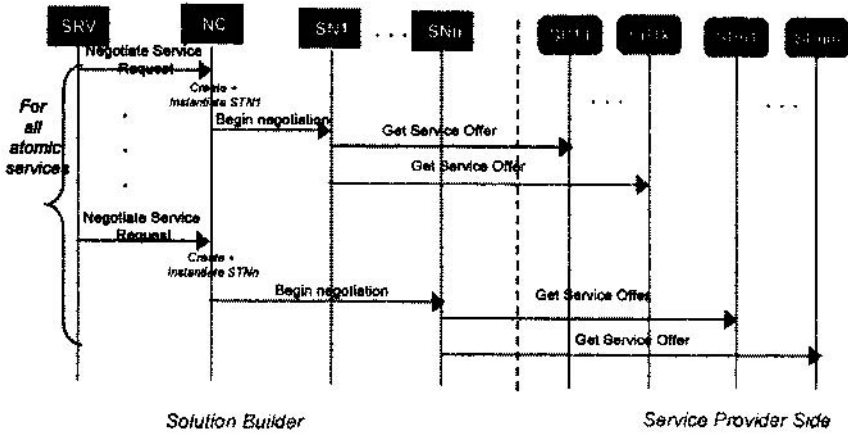


Figure 7: Solution Negotiation Procedure

The reply is returned by the service providers and is forwarding to the user via the NC and URH. In case the user is not satisfied from the offers or he wants to modify his request, then the *ModifyServiceRequest()* method (Figure) of the URH is invoked and a new search and negotiation between the user and the providers takes place until both sides are fully satisfied.

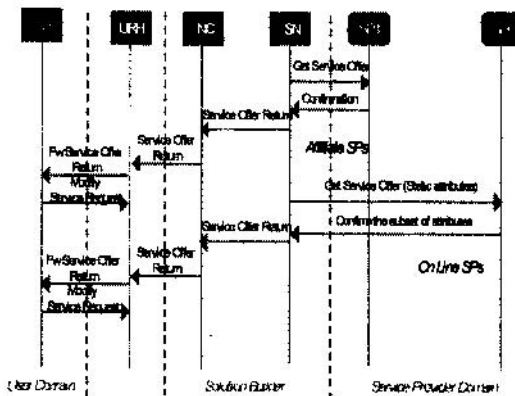


Figure 8: Single Provider Negotiation Procedure

The negotiation results are collected in the Solution Builder and the complex service evaluation phase follows. This process consists of a two level filtering according to the brokerage system and user defined criteria. This is achieved by means of detailed interaction between refined components of the Solution Builder. Specifically, when the refined service offer is created and returned to the NC there is the possibility the user has had specified in his profile that he prefers to see the offers in the form of the complex solution. Moreover, he has the right to request

more than one solutions. In the event of such a user preference the NC forwards the offer (*FwAtomicServiceOffer()*) to the *Solution Combinator Component* (SCC) which in turn creates the complex solution based on the brokerage system business criteria. The offer is forwarded (*FwGlobalSolution()*) to the *Solution Classifier Component* (SRC), where the solution rating takes place based on weighted, user defined criteria. Finally, the complex offer is displayed to the user through the URH (*ShowGlobalOffer()*). In turn, the user is able to either accept one of the proposed solutions (*ChooseOneOffer()*), meaning that the negotiation phase is over, or to re-negotiate the solutions, which is essentially the modification of his service request (*ModifyServiceRequest()*).

Within the complex services transactions execution, a concluding phase is distinguished that comprises transaction confirmation, resources reservation and payment functionality. This will take place after the user has accepted and chosen the offered complex services. The *Transaction Controller* (TC) requests the corresponding *Single Transaction Negotiator* (STN) to make the reservation of the atomic services. Then, the STN contacts the related service provider in order to reserve its offer and makes the last checks. The confirmation is returned to the URH via the TC who in turn invokes to make the payment the *Payment Manager* (PM). The Payment Manager abstracts the functionality necessary to complete the payment process and the result is made known to the URH. The completion of the payment indicates the transaction completion as well.

5. CONCLUSIONS AND DISCUSSION

In this paper, we present a particular business to business brokerage architectural model. Such as model, aims at, primarily, meeting the requirements for comprehensive, reusable and scalable component-based frameworks, in the aid of complex e-commerce transactions services provision. The designed brokerage platform has been constructed upon a refined requirements analysis exercise and an iterative service design process, that have explicitly taken into account the needs and constrains of specific business domains. These domains represent the B2B publishing and the B2B retail sectors, against the conjunctural and structural forces shaping their transformation into the Internet economy. In devising the proposed brokerage platform, we undertake the tasks to identify refined software components and test their intercommunications for certain service scenarios and procedures, which tasks are, in detail, presented in this paper. Moreover, while challenging the proposed brokerage system capabilities and robustness, we anticipate and enforce its relevance and usefulness into the rapidly changing B2B landscape, by discussing a brokerage federation policy, to be considered in detail in our future design work.

6. ACKNOWLEDGEMENTS

This work was supported by the Commission of the European Communities within the IST Programme Smart-EC (IST-1999-10130). The authors would like to warmly thank their Smart-EC colleagues.

7. REFERENCES

- [1] Martin Bichler, Arie Segev, "A Brokerage Framework for Internet Commerce", Distributed and Parallel Databases, Vol. 7, Number 2, pp.133-148, April 1999
<http://haas.berkeley.edu/citm/wp-1031.pdf>
- [2] M.Lambrou, G.Karetzos, E.Protonotarios, "A Service Analysis and Service Design Framework for the Implementation of a generic Brokerage Service", Advances in Information Technologies: The Business Challenge, pp. 226-233, IOS press, 1998.
- [3] Garr N., "Hypermediation: Commerce as Clickstreams", *Harvard Business Review*, January-February 2000.
- [4] Bailey, J. P., Bakos J. Y., "An exploratory study of the emerging role of the electronic intermediaries", *International Journal of Electronic Commerce*, Spring 1997.
- [5] Wise R. et al., "Beyond the Exchange: The Future of B2B", *Harvard Business Review*, November-December 2000.
- [6] Chircu A., Kauffman R., "Reintermediation strategies in business-to-business electronic commerce", *International Journal of Electronic Commerce*, Fall 2000.
- [7] M.B. Sarkar, B. Butler, and C. Steinfield, Intermediaries and cybermediaries: a continuing role for mediating players in the electronic marketplace, *Journal of Computer-Mediated Communication* 1(3) (1996), <http://jcmc.huji.ac.il/vol1/issue3/sarkar.html>.
- [8] Bakos J.Y., "The emerging role of electronic marketplaces on the Internet", *Communications of the ACM*, August 1998.
- [9] McAfee A., "The Napsterization of B2B", *Harvard Business Review*, January-February 2000.
- [10] Ming Fan, Jan Stallaert, Andrew B. Whinston, "The Internet and the Future of Financial Markets", *Communications of the ACM*, November 2000/Vol. 43, No. 11
- [11] M. Fernandez Lopez, "Overview of methodologies for building ontologies", Proceedings of the IJCAI-99 workshop on Ontologies and Problem-Solving Methods (KRR5) (<http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-18>)
- [12] Smart-EC: Support for Mediation And bRokering for Electronic Commerce, IST-1999-10130, RTD Project, Deliverable 3.1, "Service Specifications".
- [13] E. Tzifa, P. Demestichas, M. Lambrou, A. Kaltabani, S. Kotrotsos, M.Anagnostou, "Advanced service creation environment: Practices implementation and application", *Technologies for the Information Society: Development and Oppotunities*, pp. 466-473, IOS press, 1998.
- [14] P.P.Demestichas, N.P.Polydorou, A.K.Kaltabani, N.I.Liossis, S.A.Kotrotsos, E.C.Tzifa, M.E.Anagnostou, "Issues in Service Creation for Future Open Distributed Processing Environments", ICC'99
- [15] J. Rumbaugh, M. Blaha, W. Premerlani, F. Eddy, W. Leversen, Object-oriented modeling and design, Prentice Hall, Englewood Cliffs, NJ, 1991.
- [16] TINA-C Deliverable, Definition of service architecture, Version 5.0, 1997
- [17] TINA-C, TINA business model and reference points 4.0, Baseline Document, 1997

- [18] Smart-EC: Support for Mediation And brokering for Electronic Commerce, IST-1999-10130 RTD Project, Deliverable 4.1, "Smart-EC Architecture V1".