

# Shopbots: A syntactic present, a semantic future

Maria Fasli

University of Essex, Department of Computer Science,  
Wivenhoe Park, Colchester CO4 3SQ, UK.  
mfasli@essex.ac.uk

**Abstract:** The huge growth of e-commerce has had a profound impact on users who can now choose from an ever increasing number of options online. Inevitably, as the number of choices has increased, so has the need for tools to help users organize, manage and utilize information on these for better decision making. To this end, shopbots can help users decide what to buy and enhance their shopping experience. However, despite the high expectations, the immense potential of shopbots has not been fully realized. In this paper we identify the limitations and drawbacks of current shopbots, in particular with regards to the underlying technology for building such systems. We then discuss how these technical limitations can be overcome by making use of the Semantic Web and Web Services which would enable the realization of the true potential of shopbots.

**Keywords:** shopbots, comparison shopping, web services.

## 1. Introduction

The immense growth of the Internet has had a profound impact on the way organizations and individuals conduct business. In particular, the nature of business to consumer commerce (B2C) has changed dramatically. Companies have realized the potential of the Internet, and as a result, e-commerce sites abound offering a wide range of services and products. Users can now shop from the privacy of their homes for almost anything ranging from flowers to cars - the traditional shopping trip has been transformed into a virtual one.

Although information on different products and services is at one's fingertips, users increasingly require assistance, guidance and support in the vast information space of the Internet. There is simply too much information for an individual to process; information overload is an undisputed fact. Users looking for products online can currently visit sites that they have previously identified, or use a standard search engine and keyword retrieval to identify potential vendors. In each site visited, the user has to search for the product and find out the price and other related information. This simple approach has several shortcomings [9]. Firstly, there may be hundreds of vendors selling the same or very similar products. Unless the user is aware of particular vendors, she is faced with the problem of which of these to visit and when is it that *enough* have been visited to acquire the necessary information. Visiting multiple ven-

dors requires considerable time, but not visiting enough may lead to a suboptimal decision. Also, identifying vendors using classical search engines and keywords may not be the best way to go about it, as the returned set of vendors may be biased in favor of larger sites, which may not necessarily offer the best prices. Secondly, if the user requires several items, there may be no single site that caters for all her shopping needs, which increases the search time for each new product category. Finally, every time that a new vendor is visited the user has to get acquainted with new interfaces which increases the search time and unavoidably hinders impulse shopping.

To help customers stay informed vendors may allow them to sign up to receive alerts, for instance, when the price of a product changes or it becomes available. Such services are not personalized. Users typically have to complete long surveys and may be required to provide personal data. Furthermore to receive email notifications they are also required to reveal part of their identity. Thus, the users' privacy is weakened.

Inevitably, as the number of choices available to users has grown, so has the need for tools to help them organize, manage and utilize this information for better decision making. Agent technology can help users shift through information and find what they require. Unlike traditional software, agents are semi-autonomous, proactive, reactive and continuously running entities that can assist the user in a variety of contexts. The term shopping agent or *shopbot* has been coined to describe systems that can help users decide what to buy and enhance their shopping experience by:

- finding specifications and reviews for products;
- comparing products, vendors and services according to user-defined criteria;
- finding the best value products or services;
- monitoring online shops for product availability, special offers and discounts and sending alerts;
- recommending services and products;
- identifying new products of potential interest.

More specifically, a shopbot is an agent that operates online, receives requests from users for products and services and then can query vendors and providers and retrieve and compare<sup>1</sup> relevant information such as price, delivery options, warranty etc. Delegating such tasks to shopbots who can scour the web on the users' behalf has significant advantages. The user saves time as a shopbot can search a number of vendors much faster than its human counterpart would. As a shopbot can query many more vendors, it may also query vendors that the user has no knowledge of and yet they may be offering better deals. Furthermore, shopbots may be able to uncover special deals that the user would otherwise be unaware of. Another potential benefit is that of psychological burden-shifting [12]. Users are often uncertain about buying a product. By employing a shopbot, they can shift some of the psychological cost of making a decision to the agent. If the decision turns out to be not a very good one, the shopbot can be blamed, thereby minimizing the psychological risk in the purchase decision. In addition, when deploying shopbots users may not even be aware if the recommendation is only suboptimal. Finally, shopbots can lead to more efficient marketplaces. As shopbots provide the means for users to compare the prices and services offered by many vendors, the competition in the marketplace increases. Smaller vendors offering

---

<sup>1</sup> These are *differentiation* and *evaluation* agents according to [19].

competitive prices can be accessible to the user, thereby reducing the larger vendors' monopoly.

Shopbots can alleviate the significant burden of finding products and services and comparing prices and other attributes, ultimately enhancing the users' shopping experience while saving them time and money. However, despite the initial hype [4], the immense potential of shopbots has not been fully realized. This is primarily due to current technical limitations as shopbots cannot 'understand' the information that they retrieve, and moreover, they cannot discover and query vendors dynamically. This paper discusses the current limitations of shopbots as well as how these can be addressed.

## **2. Current technologies and limitations**

Shopbots made their appearance in the mid 1990s and technology analysts at the time were predicting that they would have a huge impact on vendors and the way business would be conducted in the years to come [4]. Some characteristic and indicative examples of shopbots are briefly described here.

The first shopbot for price comparisons was BargainFinder developed by Andersen Consulting [8]. BargainFinder allowed users to compare prices of music CDs from online stores. But many retailers started blocking access to the agent as BargainFinder simply evaluated vendors based on the offered price and ignored all other features that online music retailers had built into their sites. Eventually, BargainFinder ceased operating.

PersonaLogic [5] helped users who described their tastes make decisions on products and services that fitted their needs and preferences. The system created profiles that enabled the identification of products with features important to the users. But vendors had to provide an interface that explicitly disclosed the features of the products in a way that could be matched with the user profiles. PersonaLogic was acquired in 1998 by AOL and the system was withdrawn from the market.

ShopBot [3] was an agent that could learn to submit queries to e-commerce sites and parse the results to extract information on products. ShopBot used an automatic process for building 'wrappers' to parse semi-structured HTML documents and extracting features such as product description and price. The mechanism exploited the regularities present in e-commerce sites. ShopBot was later renamed to Jango and was acquired and commercialized by Excite in 1997. Jango avoided the problem that BargainFinder faced as product information requests originated from the user's browser, not from the agent's server.

A more recent example is IntelliShopper [9]. IntelliShopper observes the user's actions and unobtrusively attempts to learn her preferences and help her with her shopping. It also monitors various vendors for products that may match the user's needs and preferences. IntelliShopper preserves the user's privacy through concealing her IP address by passing requests through one or more anonymizer servers. Moreover, shopping personas are used to hide all information about the user from the IntelliShopper server. Logic about the different vendors that the IntelliShopper deals with is stored in modules which specify how a vendor can be queried and how to interpret the results and extract the information required by the user from the returned HTML pages.

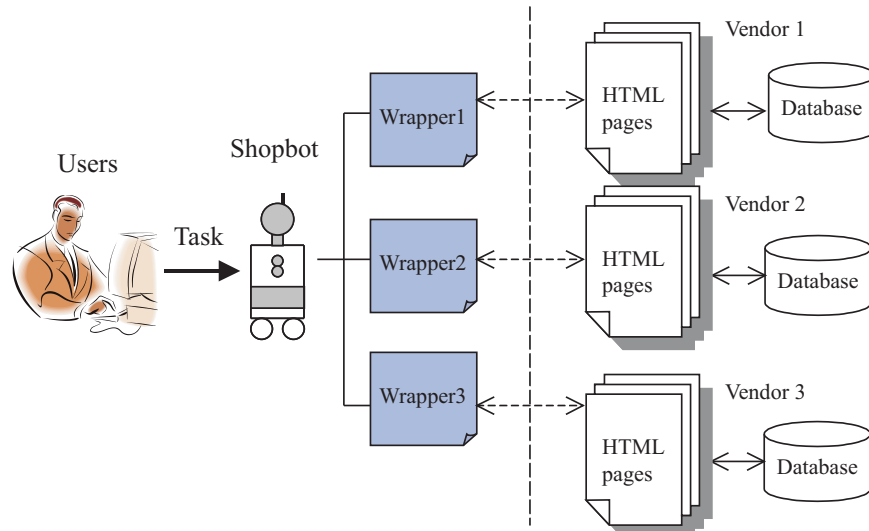


Figure 1. How current shopbots work.

Shopbots operate in a similar way to meta-search engines and retrieve information from different vendors by a form of ‘screen-scraping’ [14]. They typically permit buyers to sort product and vendor information along some desired dimension such as price. Essentially, shopbots interact with a vendor through HTML pages that are designed and generated to be read and understood by humans not programs. They submit queries to vendors and attempt to process the resulting HTML pages by parsing them and searching for the name of the item of interest to the user and then the nearest set of characters that has a currency sign, which presumably is the item’s price. Shopbots rely on regularities in the layout of the vendors’ web pages to retrieve such information [3]:

- navigation regularity: e-commerce sites are designed so that users can easily navigate through them and find what they need;
- uniformity regularity: the web pages of a site are designed so that they have a uniform look;
- vertical separation regularity: white spaces are customarily used to separate products and each new product description starts in a new line.

In fact, the information that the shopbot is after is stored in a machine-processable and well-structured format in the vendor’s database. But the shopbot cannot interact with the database; it only has access to the web pages that are generated from it as illustrated in Figure 1. Consequently, developers have to implement clever heuristics that can extract the original well-structured data from the implicit information on the HTML pages. Such heuristics are inevitably ad-hoc, difficult and time-consuming to develop and prone to errors, while they are only able to retrieve limited information. Inevitably, they must be updated every time the layout of the vendor’s site changes. This makes the development of shopbots cumbersome and the resulting systems inflexible and vendor-specific. New vendors cannot be discovered and queried at run-

time: they first have to be identified and then tailor-made methods for retrieving information from their web pages need to be developed.

As current techniques rely on syntax, shopbots can only retrieve limited information from vendors and ignore attributes such as warranty and shipping options which are harder to retrieve. This is a severe limitation. Although price is often the most important factor in deciding what to buy and where from, other attributes may affect the buying decision such as the delivery and payment options, or the vendor's reputation. Crucially, these value-added services may be what differentiates vendors that otherwise may be offering the same price. Naturally, vendors object to the idea of being compared on price alone as this does not necessarily reflect the full range of services and value for money that they offer. Many vendors block price requests from shopbots thus restricting their ability to make comparisons among multiple vendors and assist the user in finding the best deal around.

This limitation has further consequences on the accuracy of the information retrieved. Often there are discrepancies between the prices reported by shopbots and those listed at the vendor's site. As shopbots do not understand the content of the retrieved web pages, they cannot distinguish between vendors that include shipping charges and taxes in their prices and those that do not. Shopbots therefore appear incapable of providing accurate information on vendors and products.

The aforementioned problems have led to the demise of vendor-independent shopbots as described above and the emergence of comparison shopping sites. Such sites may rely on vendors to provide the necessary information on catalogues and products, or operate as meta-search engines and perform searches on the vendors' sites. Sites like MySimon<sup>2</sup> and DealTime<sup>3</sup> collate catalogues of products which are provided by the vendors themselves who pay to be listed on the site. Comparison sites that operate as meta-search engines such as Kayak<sup>4</sup> and SideStep<sup>5</sup> may have either informal or formal relationships with vendors. Vendors may simply allow such shopping sites to access and retrieve information from their web sites or form explicit partnerships and pay a commission for each hit made to the vendor's site as a result of the listing, or for sales resulting from clickthrough purchases or alternatively they place advertisements. Froogle<sup>6</sup>, Google's search engine for products, indexes products from multiple e-commerce sites in two ways: (i) sellers can submit information on the products that they would like to list in Froogle directly; (ii) Froogle identifies web pages that offer products and services while Google searches and indexes the Web.

Such comparison sites are not actually agents as described earlier. Moreover, they may not necessarily offer impartial advice and although they do allow comparisons among multiple vendors, there may be better deals around. Nevertheless, such comparison shopping sites can only provide information based on price and attributes such as warranty, delivery options etc., cannot be explicitly retrieved and presented to the user in a form that can be used for making comparisons.

---

<sup>2</sup> <http://www.mysimon.com>

<sup>3</sup> <http://www.dealtime.com>

<sup>4</sup> <http://www.kayak.com>

<sup>5</sup> <http://www.sidestep.com>

<sup>6</sup> <http://froogle.google.com>

### 3. Shopbots and the Semantic Web

Despite the high expectations about the impact of shopbots on retail markets, their true potential has not been realized. This is mainly due to the technique used to extract information, i.e. screen-scraping, and its inherent limitations.

A very promising avenue to tackling this problem is offered by Web services. A Web service is a collection of functions that are packaged as a single entity and published on the network to be used by other programs [15]. The programmatic interfaces to Web services can be described in the Web Services Description Language (WSDL) [1] which is an XML based language. The description of a service includes the input and output parameters and their data types, the format of messages that can be exchanged, the service's operations, its network address (Uniform Resource Identifier (URI)), and protocol bindings. The creators of services can describe and register them in directories whose purpose is to facilitate the discovery of services by requesters. The Universal Description, Discovery and Integration protocol (UDDI) standard specification [17] describes a mechanism for registering and locating services. Knowing the location of a service, i.e. its URI, a requester can use the Simple Object Access Protocol (SOAP) [16] to invoke the service with its parameters. The power of Web services lies in that they can interact seamlessly and transparently to achieve combined functionality and produce the required result [15].

Shopbots can use Web services as gateways to vendors' sites since services can provide the means of retrieving information directly from the vendor in machine processable form. For instance, Amazon<sup>7</sup> offers web services that allow programs to perform operations such as retrieving information about products and adding items to a shopping cart. Agents can employ Web services to acquire, aggregate and compare information on products, services and vendors. This will greatly enhance the potential for comparisons not only based on price, but on various other attributes.

Shopbots can potentially access a large number of vendors provided that the latter make available the programmatic interfaces to facilitate the retrieval of the required information. What can hinder this process is lack of understanding when referring to the attributes or other aspects of a product or service. For instance, unless an agent understands that the terms 'warranty' and 'guarantee' mean the same thing to different providers, it will be unable to correctly retrieve information and present it in a uniform way to the user, or perform meaningful comparisons. To facilitate a common understanding, underlying domain ontologies are required. Such ontologies will be used both by creators to describe web services as well as by shopbots to understand and process the retrieved information and present it to the user. Therefore, shopbots need to be endowed with the ability to use and reason about ontologies.

In an increasingly interconnected world there may be hundreds of providers that offer a particular service or product. A shopbot may not know how to find them and invoke their services. Consequently, some means of locating providers is needed. One obvious solution to this problem is to use a UDDI registry to discover providers and their services. Having the description of a service, the shopbot knows its location and how to invoke it. However, the problem with the current XML-based standards for describing Web services such as SOAP and WSDL is that they are designed to provide descriptions of transport formats and mechanisms, address binding, and the inter-

---

<sup>7</sup> <http://www.amazon.com>

face used by each service, but not capability descriptions. Similarly, UDDI does not represent service capabilities and thus cannot facilitate the automatic location of web services based on capability specifications [11,18]. For instance, a search in a UDDI registry for vendors (their services) that can ship outside the US is not possible as UDDI would not be able to represent such a capability. Although, vendors associated with the US in some way would be retrieved in this case, it is extremely difficult to search for services with a specific capability. Another limitation shared by the aforementioned standards is that they lack an explicit semantics: although two XML descriptions of services may look exactly the same, they may mean different things in different contexts.

For shopbots to achieve their full potential, the matching between a request for service and potential providers needs to take into account semantics and capability specifications and also needs to be done in context. Hence, both richer descriptions of Web services and the means to process such descriptions are required. A more suitable framework for describing the capabilities of services is that provided by the DARPA Agent Markup Language (DAML) [2,11] and its most recent reincarnation the Ontology Web Language for Services (OWL-S) [10]. A Web service in OWL-S is described by three interrelated sub-ontologies. The *ServiceProfile* describes the capability specification of a service which is essential in determining whether or not it meets the requester's needs. The *ServiceModel* describes how a requester can use the service, how to ask for it and what happens when it is carried out. The *ServiceGrounding* specifies the communication protocol, message formats and other service-specific details such as port numbers used in contacting the service. The selection of providers using such enhanced descriptions requires a type of registry that enables search based on semantics which is not currently offered by UDDI [18]. Alternatively, middle or broker agents can be used that act as ontology-based search engines and take into account semantics [20]. The role of the middle agent in this context is to match the shopbot's request for service with one or more providers. This process is simply referred to as *matching*. Therefore, middle agents specialize in making connections and they store, maintain and provide connection information. Vendors can publish/register their Web services with middle agents, and shopbots can then enquire middle agents when receiving requests from users to find products.

The elements described above give rise to the setting illustrated in Figure 2. Vendors create web services which allow access to their sites and which they publish/register with middle agents. Users delegate the task of finding and comparing products to shopbots. Shopbots contact middle agents who identify and return a set of providers offering the service along with information on how to contact and interact with the service. This is subsequently used by the shopbot to invoke the service and retrieve information. In such a setting, shopbots would be able to dynamically retrieve information from a variable number of vendors and not depend on static links that may be unreliable. They could also cache the locations of web services for common requests thereby improving efficiency. The end result would be easily digestible information that the user can use to make the final purchasing decision.

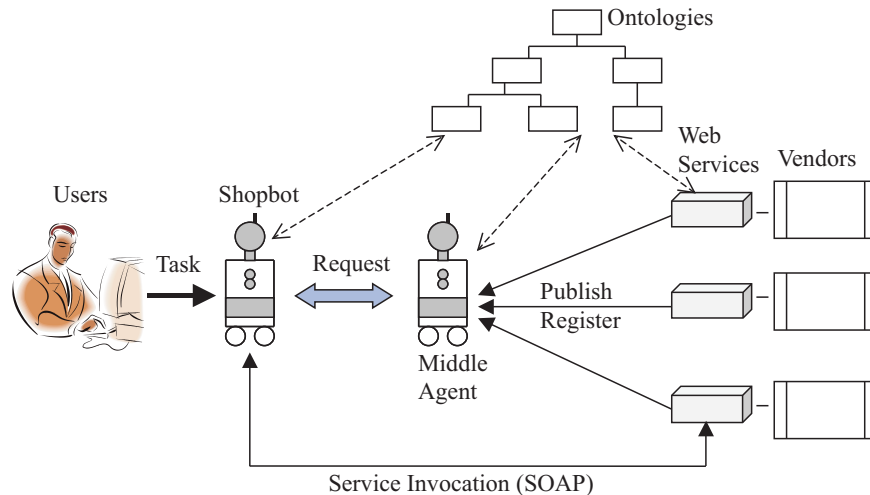


Figure 2. Shopbots: Making use of the Semantic Web and Web Services.

To allow shopbots to retrieve semantic information on products and services from vendors the following main challenges need to be addressed:

- efficient algorithms for matching requests with web services;
- efficient ontology services including ontology consolidation services;
- reasoning mechanisms that enable shopbots to perform true comparison shopping based on the user's preferences, the retrieved information and the underlying ontologies.

Although standardizing ontologies to describe certain parts of the retail sector would be ideal, standardization is a general problem with ontologies. That is why shopbots and middle agents need to be able to reason with and compare multiple ontologies.

Work on shopbots that are able to retrieve information on vendors through web services has been so far rather limited. A recent approach is [7] where a meta-search and comparison system for products is described. The Intelligent Product Information Search (IPIS) system assumes that shopping malls offer web services that describe the products and goods available. On receiving a request from a user the system is able to use the services to retrieve information. Such an approach is along the lines described above, though the concept of a middle agent as an intermediary is not used.

#### 4. Towards a Semantic future

Shopbots have the potential to have a major impact on both brick-and-mortar and purely Internet vendors and ultimately lead to more efficient marketplaces as well as enhance the users' shopping experience. The transition to the Semantic Web and Web Services would alleviate the problems associated with current shopbots and compari-



son shopping sites. In particular, Web services allow for the retrieval of additional attributes apart from price. As shopbots would have an understanding of these attributes, they would be able to offer comprehensive information to users and uncover deals that are differentiated not only by price, but also other attributes. Such information can be presented in a uniform way. Vendors can be discovered and queried dynamically depending on the user's request through middle agents. Fundamentally, shopbots no longer have to be dependent on specific vendors and thus can offer impartial advice. As vendors can be compared on additional attributes and not on price alone, they have an incentive to provide programmatic interfaces to their sites to enable shopbots to locate them thereby increasing their visibility. Although this inevitably increases competition, overall market efficiency improves. Smaller vendors can increase their visibility and benefit from the large volumes of traffic that pass through the shopbot [6]. Larger vendors with a bigger customer base and more loyal customers may have less of an incentive to join or allow access to a shopbot. Nevertheless, some will choose to do so in order to increase their customer base even further.

Shopbots can also facilitate anonymity on the Web as the requests for product information can be processed without revealing the users's identity. Price discrimination schemes would be difficult to apply as vendors cannot know the identity of the user requesting information. Furthermore, shopbots can use reputation systems to filter out unreliable vendors. Reputation systems collate information on vendors by aggregating feedback on past transactions from users (or agents) and then measuring the vendor's trustworthiness via some means [13]. Shopbots may be able to steer users away from dubious vendors and can help tackle market fraud. Finally, users could interact with shopbots indirectly through their personal agents. A user could delegate the task of finding products to her personal agent who in turn contacts a shopbot and requests information. The user's personal agent can filter the information returned by the shopbot even further by considering additional user preferences or other restrictions.

Shopbots can benefit not only users in B2C markets, but also organizations in business to business (B2B) markets. A shopbot can perform searches and comparisons not for consumer goods and products, but for suppliers and components on behalf of businesses and organizations as in a supply chain. Shopbots can locate the cheapest suppliers or those that can satisfy certain conditions, e.g. delivery deadlines. The same infrastructure as described above would enable interactions between parties in B2B markets reducing costs and improving the overall market efficiency.

In a nutshell, shopbots will only realize their true potential if they utilize the power of the Semantic Web and Web services.

#### **Acknowledgements**

The discussion on the problems associated with current shopbots is derived from Fasli, M. *Agent Technology for E-commerce*, 2007, Copyright John Wiley and Sons Limited. Reproduced with permission.

## **7. References**

1. E. Christensen, F. Curbera, G. Meredith, and S. Weerawarana. WSDL. Web Services Description Language. See <http://www.w3.org/TR/wsdl>, 2001.
2. DAML. The DARPA Agent Markup Language. See <http://www.daml.org>, 2006.

3. R.B. Doorenbos, O. Etzioni, and D.S.Weld. A scalable comparison-shopping agent for the World-Wide Web. In Proceedings of the First International Conference on Autonomous Agents (Agents-97), pp.39—48, 1997.
4. J. Edwards. Is that your best offer? Shoppbots search the web for bargains. CIO Magazine, 2000. See <http://www.cio.com/archive/110100/et.html>.
5. A.G. Guttman, R.H. Moukas and P. Maes. Agent-mediated integrative negotiation for retail electronic commerce. In Agent Mediated Electronic Commerce, First International Workshop on Agent Mediated Electronic Trading (AMET'98), LNCS Volume 1571, pp.70—90. Springer, Berlin, 1999.
6. G. Iyer and A. Pazgal. Internet shopping agents: Virtual co-location and competition. Marketing Science, 22(1):85—106, 2003.
7. W. Kim, D. Choi, J. Kim, and J. Jin. Development of a meta product search engine with web services. In Proceedings of the Second Asia Information Retrieval Symposium (AIRS 2005), LNCS Volume 3689, pp.571—576. Springer, Berlin, 2005.
8. B.T. Krulwich. The BargainFinder agent: Comparing price shopping on the Internet. In J. Williams, editor, Bots and other Internet beasts, pages 258—263. SAMS.NET, Macmillan, 1996.
9. F. Menczer, W.N. Street, and A.E. Monge. Adaptive Assistants for Customized E-Shopping. IEEE Intelligent Systems, 17(6):12—19, 2002.
10. OWL-S. Semantic Markup for Web Services. See <http://www.w3.org/Submission/OWL-S/>, 2004.
11. M. Paolucci and K. Sycara. Autonomous semantic web services. IEEE Internet Computing, 7(5):34—41, 2003.
12. V. Rajiv and P. Aggarwal. The impact of shopping agents on small business ecommerce strategy. Journal of Small Business Strategy, 13(1):62—79, 2002.
13. P. Resnick, K. Kuwabara, R. Zeckhauser, and E. Friedman. Reputation systems. Communications of the ACM, 43(12):45—48, 2000.
14. M.P. Singh. The service web. IEEE Internet Computing, 4(4):4—5, 2000.
15. M.P. Singh and M.N. Huhns. Service-oriented computing: Semantics, Processes, Agents. John Wiley and Sons, Chichester, 2005.
16. SOAP. Simple Object Access Protocol. See <http://www.w3.org/TR/soap/>, 2003.
17. UDDI. Universal Description, Discovery and Integration protocol. See <http://www.uddi.org/>, 2006.
18. K. Verma, K. Sivashanmugam, A. Sheth, A. Patil, S. Oundhakar, and J. Miller. METEOR-S WSDI: A scalable P2P infrastructure of registries for semantic publication and discovery of web services. Information Technology and Management, 6(1):17—39, 2005.
19. Y. Wan, S. Menon, and A. Ramaprasad. A classification of product comparison agents. In Proceedings of the Fifth International Conference on Electronic Commerce (ICEC'03), pp.498—504, 2003.
20. H.C. Wong and K. Sycara. A taxonomy of middle-agents for the Internet. In Proceedings of the Fourth International Conference on MultiAgent Systems (ICMAS-00), pp.465—466, 2000.