Implementing Semantic Interoperability in Electronic Auctions

Juha Puustjärvi [1])

Abstract

Electronic marketplaces can provide several types of business processes depending upon their target audience. A form of such processes is electronic auction. It is a system for accepting bids from bidders and computing a set of trades based on the offers according to a well defined policy. By automating electronic auctions both buyers and sellers can benefit in many ways. In particular they can achieve cost reductions and shorten the duration of the auction processes. The automation of electronic auctions however requires that both buyers' and sellers' auction agent have shared understanding of the auction process as well the semantics of the exchanged messages. How this can be achieved is the main topic of this article. In particular, we illustrate what kind of auction ontologies we are using and how they can be maintained in semantic interoperability. In addition we illustrate how ontology managers communicate through SOAP messages. The communication is coordinated by a workflow engine, which is able to run a variety of auction formats that are specified by BPEL.

1. Introduction

In electronic business buyers and sellers should be able to interact with each others inside an architecture that is easy to use and maintain. Electronic auctions are an interesting approach to achieve this goal by bringing together business in the web.

Technically, an electronic marketplace is a virtual place that resides somewhere in the Internet [1]. They can provide several types of business processes depending upon their target audience. A form of such processes is electronic auction. It is a system for accepting bids from bidders and computing a set of trades based on the offers according to a well defined policy.

Most auction software is targeted only to B2B procurement [2]. At the same time, there is growing interest on advanced auction formats such as on combinatorial auctions and on multi-attribute auctions [3]. In combinatorial auctions bidders are allowed to place offers on sets on items whereas in multi-attribute auction price is not the only negotiable parameter.

We also make the distinction between human oriented electronic auctions and automated electronic auctions. In human oriented auctions the auction system communicates with the humans (buyers and sellers) in carrying out the auction. In automated electronic auctions buyers' and sellers'

[1] Lappenranta University of Technology, POB 20, FIN-53851 Lappenranta , Finland

software modules (auction agents) communicate with the auction system. In particular, auction agents make bids based on the predefined rules. The rules are typically classified into three classes [4]: rules that control the admission of bids, rules that control the information revealed by the auction, and rules that control how the auction computes trades.

By automating electronic auctions both buyers and sellers can benefit in many ways, e.g., they can achieve cost reductions and shorten the duration of the auction processes [5, 6]. The automation of electronic auctions requires that both buyers and sellers have shared understanding of the auction process as well the semantics of the exchanged messages. Shared understanding of the auction processes can be achieved by process models, e.g., we can use for example BPMN (Business Process Modeling Notation) [7, 8], BPEL [9] or UML activity diagram [10].

Shared understanding of the exchanged messages can be achieved by semantic interoperability. That is, after data were transmitted from a sender system to a receiver, all implications made by one party had to hold and be provable by the other. This however does not necessary imply that each participant (buyer, seller or marketplace) has equal knowledge of each auction process. For example, in a sealed auction only the auction system (marketplace) knows all the bids while bidders only know their own bids and the highest bid.

How the parties of an electronic auction can maintain their knowledge of the auction processes is the main topic of this article. Managing this knowledge in a consistent way is crucial because auction agents make their decision based on this knowledge.

The rest of the paper is organized as follows. First, in Section 2, we illustrate the role of auction ontologies in semantic interoperability. In particular, we first illustrate the distinction between auction ontologies and auction instance ontologies as well as the distinction between global auction ontology and view auction omtologies. Then, we explain how these ontologies are maintained according to auction processes. We also give an example of a SOAP message and illustrate how it is used in maintaining auction ontologies. In addition we give a short overview of XML-based ontology languages that we are using in expressing ontologies. In Section 3, we consider the architecture of the auction system that we are developing. In particular we illustrate the role of auction agents, ontology managers, web services and workflow engine as well as their relationships. We also give a simple example how BPEL can be used in specifying an auction process. Finally, Section 4 concludes the paper by discussing the advantages and disadvantages of our approach.

2. Auction ontologies

2.1. Global ontology and view ontologies

An ontology is a general vocabulary of a certain domain [11], and it can be defined as “an explicit specification of a conceptualization” [12]. Essentially the used ontology must be shared and consensual terminology as it is used for information sharing and exchange.

Ontology tries to capture the meaning of a particular subject domain that corresponds to what a human being knows about that domain [13]. It also tries to characterize that meaning in terms of concepts and their relationships. Ontology is typically represented as classes, properties attributes and values. So they also provide a systematic way to standardize the used metadata items.

A salient feature of ontologies is that depending on the generality level of conceptualization, different types of ontologies are needed. Each type of ontology has a specific role in information sharing and exchange.

The purpose of the auction ontology is to describe the concepts of the domain in which auction take place. So, an auction ontology may for example describe the concepts and their relationships related to English auction, combinatorial auction, and multi-attribute auction. To illustrate this, a simple auction ontology is presented in Figure 1. In this graphical representation ellipses represent classes and subclasses, and rectangles represent properties.

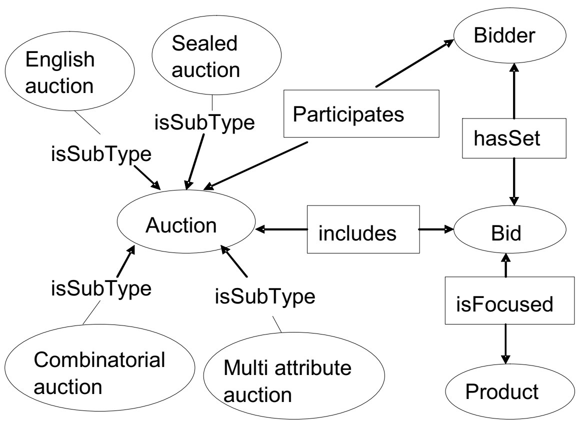

Figure 1: Auction ontology.

The purpose of the auction instance ontology is to describe the concepts of the domain in an auction as well as their instances. To illustrate this, a simple auction instance ontology is presented In Figure 2. It describes a sealed auction (having ID 1234) where Company A has set the bid of $70 and company B has set the bid of $74 of product X531.

Note that as the instance ontology of Figure 2 represents a sealed auction and so Company A cannot see the bids of Company B, and vice versa. Company A's view ontology of the instance ontology of Figure 2 is presented in Figure 3. It includes the value of the highest bid but not any information about its bidder.

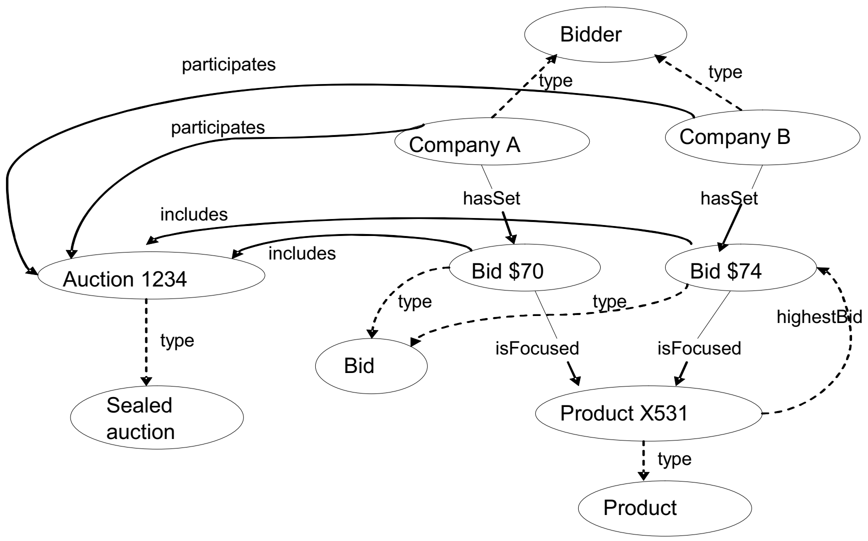

Figure 2: An auction instance ontology.

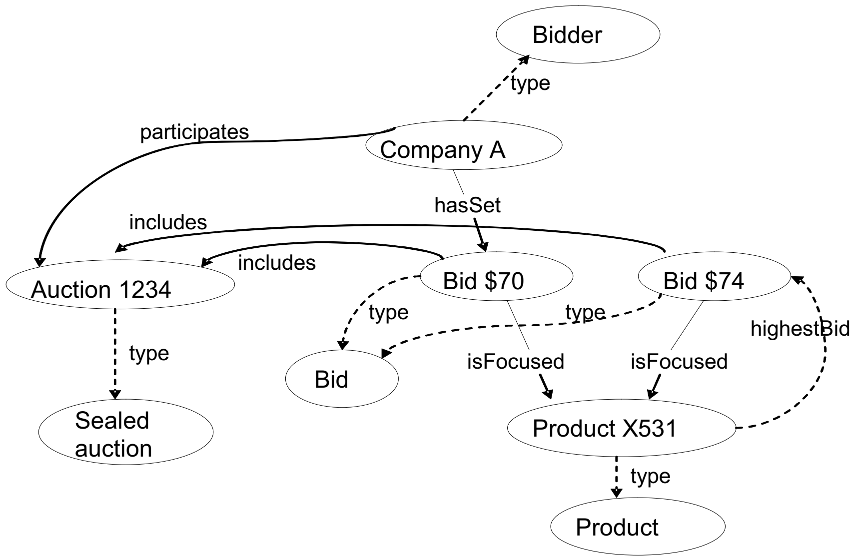

Figure 3: Company A's view ontology.

The view ontology of Company A is constructed and maintained by the ontology manager located at Company A's site (the whole architecture is presented in Section 3). Its functionality is based on its message exchange between the marketplace. We next illustrate how the ontology manager is able to unambiguously interpret the elements of the exchanged messages, and thus maintain the view ontology.

2.2. Identifying auction ontologies

Using the auction ontologies in information exchange requires that the terms in the ontology are globally unique. This requirement is easily achieved by storing the ontology in the Web and identify it by its address, i.e., by its URL (Uniform Resource Locator) [14]. Hence the ontology can be identified for example by the URL: http://www.it.lut.fi/ontology/auction.

Using this URL as the prefix of an XML-element we can give globally unique names for auction models and their elements. For convenience, however, it is useful to specify an abbreviation for the URL, say ao. This can be specified as follows:<xmlns: ao=”http://www.it.lut.fi/ontology/auction” >

Now, for example, the element <ao:sealed_auction> is a globally unique name for the Sealed auction of the ontology presented in Figure 1. Respectively <ao: request_for_bid_message> and <ao:response_for bid_request >are globally unique names for the messages of the Sealed auction model.

This kind of mechanism for specifying globally unique names for elements and the attributes of the markup language is called namespaces. In general, namespaces [15] are important for two reasons: to deconflict the meaning of identical names in different markup languages, and to allow different markup languages to be mixed together without ambiguity.

We next consider how globally unique auction messages can be used in invoking the operations of the Web services, which support auction operations.

2.3. Requesting auction operations

We now illustrate how we connect the elements of the SOAP messages to the auction ontology, i.e., how we can achieve the shared understanding of the exchanged messages. To illustrate this assume that a Web service of the electronic auction supports the operation “response_for_bid_request” which has at least the following parameters: Auction_id, Companyr_id, Product_id, and Bid. The SOAP-envelope of the former operation is presented below in Figure 4 (SOAP [13] is a protocol specification for invoking Web services and defining a uniform way of passing XML-encoded data).

```
<SOAP-ENV: Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  SOAP-ENV:encodingStyle=
     "http://schemas.xmlsoap.org/soap/encoding/">
     <SOAP-ENV:Body>
        <ao:response_for_bid_request
              xmlns:="http://www.it.lut.fi/ontology/auction/">
           <ao:auction_id>1234</ao:auction_id>
           <ao:company_id>CompanyA</ao:company_id>
           <ao:product_id>X531</ao:product_id>
           <ao:bid>70</ao:bid>
        </ ao:response_for_bid_request>
     </SOAP-ENV: Body>
</SOAP-ENV: Envelope>
```

Figure 4: A SOAP message

.

Note that namespace definition "ao" in the beginning of each element gives the semantics for each element, i.e., connects the elements to the auction ontology stored at http://www.it.lut.fi/ontology/auction.

2.4. Expressing ontologies

We now give a short introduction to the XML-based languages that we are using in specifying auction and auction instance ontologies. RDF [15] provides a means for attaching semantics (e.g., metadata values) to objects. The relationship of XML and RDF is that XML provides a way to express RDF-statements. In other words, RDF is an application of XML.

Fundamentally, RDF defines a language for describing relationships among resources in terms of named properties and values. It however, provides no mechanisms for describing these properties, nor does it provide any mechanisms for describing the relationship between these properties and other resources. That is the role of RDF vocabulary description language RDF Schema [15]. It defines classes and properties that may be used to describe classes, properties and other resources. Hence, there is a straight correspondence between RDF schema and object oriented design. For example, using RDF statements and the modeling primitives of RDF Schema we can state that

- "auction is a class",
- "sealed auction is a subclass of auction", and
- "auction 1234 is a type of sealed auction".

OWL Web Ontology Language [13] has more facilities for expressing meaning and semantics than XML, RDF and RDF Schema, and thus OWL goes beyond these languages in its ability to represent machine interpretable content of the ontology. In particular, it adds more semantics for describing properties and classes, for example relations between classes, cardinality of relationships, and equality of classes and instances. For example, with respect to the auction ontology represented in Figure 1, we can state that

- "if the auction is a combinatorial action, then a bid may focus on more than one product" and
- "the classes combinatorial and multi attribute auctions are disjoint".

3. The auction system

3.1. The architecture of the system

The auction system has three types of users: buyers, sellers and the auction system administrator (Figure 5). System administrator is a person (or a role) who maintains the auction system. In reverse auction (i.e., in procurement) a buyer is the auction initiator, and in other forms of auction seller is the auction initiator. Note that even in the figure each site represents either a buyer or a seller, each site may be a buyer in an auction and a seller in another auction. In this sense the architecture is symmetric.

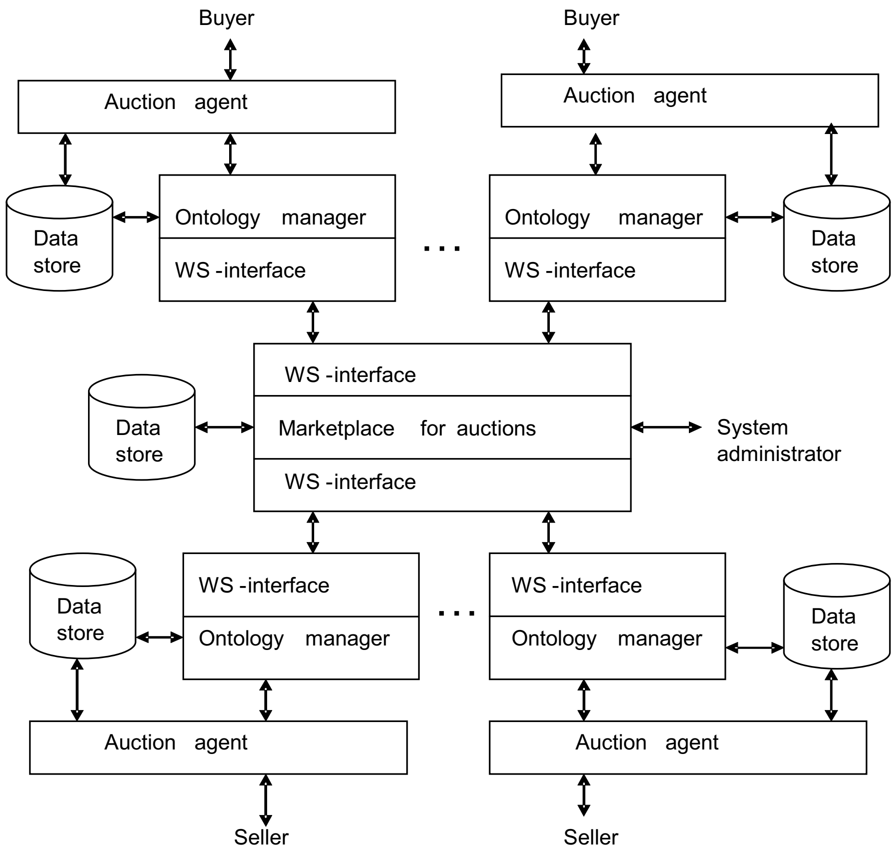

Figure 5: The Architecture of the system

As illustrated in Figure 5 buyers and sellers interact with the auction system through their Auction agents. The Auction agents communicate with the marketplace through their Web-services. Based on this communication the Ontology managers maintain (as illustrated in Section 2.3) their view ontologies that are stored in local Data Stores. For example after the ontology manager has sent the “response_for_bid_request” it inserts to the view ontology the values of the elements Auction_id, Companyr_id, Product_id, and Bid. Correspondingly when the Ontology manager receives information (e.g., highest bid) from the Marketplace it updates the view ontology accordingly (e.g., updates the highest bid). In the same way the marketplace maintains the global auction ontology that is stored at the data store of the marketplace.

Basically, there are two approaches how companies (in the role of buyer or seller) can integrate their system with the Auction agent: A company communicates with the Auction agent through a web service interface, or a company integrates its content management system with the Auction agent. The gain of the first approach is that it has minimal initial costs but has high operational cost as it requires duplication of content management effort. In the second approach the costs are other way around. However this approach is extremely fascinating as it allows (through a web service) the integration of the ERP-system (Enterprise resource Planning system) with the auction agent. In particular, this approach nicely matches with the third wave ERP-systems which are based on the use of Web services.

3.2. Using BPEL in coordinating the auction processes

We now consider how the marketplace coordinates the auction and maintains the global ontology. The coordination is carried out by the workflow engine (a module of the marketplace), which is specified by BPEL (Business Process Execution Language for Web Services) [9]. It represents a convergence of the early workflow languages WSFL (Web Service Flow Language) [16] and XLANG [17]. BPEL can be used within and between enterprises: within enterprises it is used for application integration and between enterprises it enables easy and effective integration of business partners. In particular, BPEL is the key technology for environments where functionality is exposed via Web services. Hence, it is also an appropriate technology for Web service based electronic auction systems.

A BPEL process specifies the order in which participating Web services (Buyers and Sellers) are invoked. With BPEL we can also specify conditional behaviors (e.g., whether the bid of Company A is higher than the Bid of Company B). In addition as electronic auctions are graphs of activities, we can first express it using UML activity diagrams, and then generate the BPEL code from the diagram.

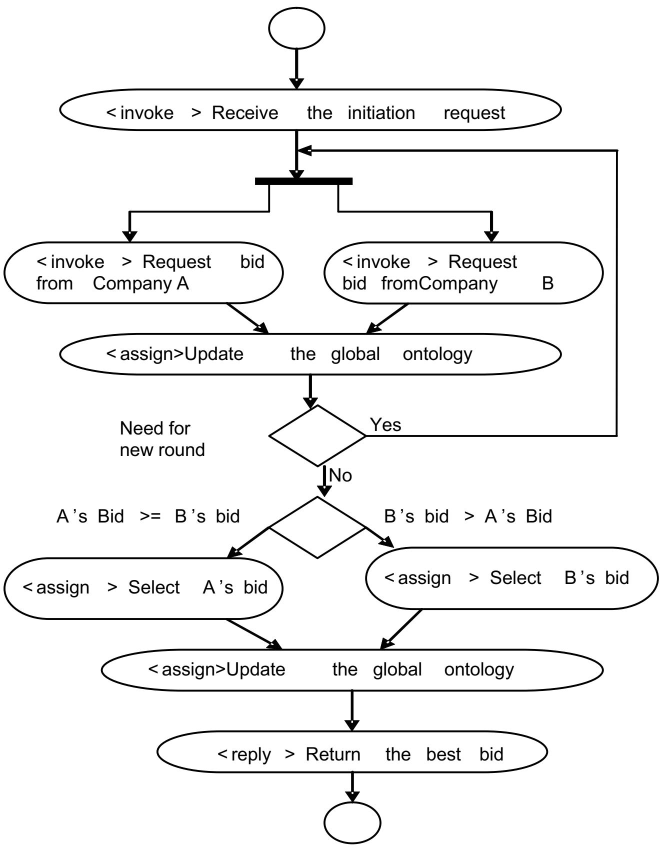

Figure 6: A BPEL process for a simple auction

The reason for using BPEL is that its notation is readily understandable for the buyers and sellers of the system. It is also readily understandable for the business analyst that create the drafts of auction processes as well as for the technical developers responsible for implementing the technology that will perform those processes. In addition, a notable gain of BPEL is that we can also use BPMN (Business Process Modeling Notation) [18] for representing the auction process and then generate an executable BPEL code from it. In order to illustrate how we can use BPEL in specifying electronic auctions an overly simplified auction format is presented in Figure 6. The auction is comprised of time rounds where an offer is required from each seller before the auction can proceed. Such auctions can be conceptualized as having different activities. The types of activities in the Figure 6 are <invoke> (invoking other Web service), <reply> (generates a reply), and <assign> (manipulates data).

4. Conclusions

In electronic business buyers and sellers should be able to interact with each others inside an architecture that is easy to use and maintain. Electronic auctions are an interesting approach to achieve this goal by bringing together business in the web.

In automated electronic auctions buyers and sellers do not directly take part to the auction process but rather their agents participate to the auction process. By such automated electronic auctions both buyers and sellers can benefit in many ways. In particular they can achieve cost reductions and shorten the duration of the auction processes. However, automating electronic auctions requires that both buyers and sellers have shared understanding of the semantics of the exchanged messages. This in turn requires that the participants of the auctions commit to the same ontologies.

References

[1] P. Wurman, Online Auction Site Management. http://www.csc.ncsu.edu/faculty/wurman/Papers/Wurman-article.pdf, 2005

[2] B. Siddiqui, UDDI-based electronic marketplace, in Web Services Business strategies and architectures. Ed. P-Flechter and M. Waterhouse. Expert Press, Birmingham, 2002.

[3] N., Nissan, 2000. Bidding and allocation in combinatioral auction. In proc Second ACM Conference on Electronic Commerce. 2000.

[4] P. Wurman, P, M. Wellman and W Walsh, Specifying Rules for Electronic Auctions. http://www.csc.ncsu.edu/faculty/wurman/Papers/Wurman-article.pdf. 2005

[5] S. Klein, and R. O'Keefe. The Impact of Web on Auctions, International Journal of Electronic Commerce, 3(3).1999.

[6] P. Wurman, P. Wellman, and W. Walsh. A parametrization of the auction design space. Games and Economic Behavior. 35(1-2), 2001.

[7] Business Process Modeling Notation (BPMN), http://www.bpmn.org/

[8] A.White. Introduction to BPMN, IBM Corporation
http://www.bpmn.org/Documents/Introduction%20to%20BPMN.pdf

[9] BPEL4WS – Business Process Language for Web Sevices.
http://www.w.ibm.com/developersworks/webservices/library/ws-bpel/

[10] Introduction to OMG's Unified Modeling Language, http://www.omg.org/gettingstarted/what_is_uml.htm

[11] J. Davies, D. Fensel, and F. Harmelen, Towards the semantic web: ontology driven knowledge management. John Wiley & Sons, 2002.

[12] M. Gruber, Thomas R. 1993. Toward principles for the design of ontologies used for knowledge sharing. Padua workshop on Formal Ontology, March 1993. 23 p.

[13] M. Daconta, L. Obrst, & K. Smith. The semantic web. Indianapolis: John Wiley & Sons. 2003

[14] E. Harold. and W. Scott Means., XML in a Nutshell. O'Reilly & Associates, 2002.

[15] G. Antoniou, & F. Harmelen. A semantic web primer. The MIT Press. 2004.

[16] WSFL – Web Service Flow Language (WSFL) http://xml.coverpages.org/wsfl.html

[17] XLANG–Web Services for Business Process Design. http://www.gotdotnet.com/team/xml_wsspecs/xlang-c/default.htm

[18] Business Process Modeling Notation (BPMN), http://www.bpmn.org/