The following paper was originally published in the
Proceedings of the 3rd USENIX Workshop on Electronic Commerce
Boston, Massachusetts, August 31–September 3, 1998

The Auction Manager: Market Middleware for Large-Scale Electronic Commerce

Tracy Mullen and Michael P. Wellman
University of Michigan

The Auction Manager: Market Middleware for Large-Scale Electronic Commerce

Tracy Mullen and Michael P. Wellman
Artificial Intelligence Laboratory
University of Michigan
1101 Beal Avenue
Ann Arbor, MI 48109-2110 USA
{mullen,wellman}@umich.edu
http://ai.eecs.umich.edu/people/{mullen,wellman}

Abstract

As the number and diversity of electronic commerce participants grows, the complexity of purchasing from a vast and dynamic array of goods and services needs to be hidden from the end user. Putting the complexity into the commerce system instead means providing flexible middleware for enabling commerce within different commercial communities.

In this paper, we present one such commerce middleware component — an Auction Manager designed to simplify and automate both the creation of new markets and the matching of users to existing markets. The Auction Manager determines which markets are appropriate for a given buyer or seller using market-specific inference rules applied to the current market offerings. We also show how these same inference rules can be used by the Auction Manager to automatically compose and decompose market offerings to respond to changing conditions within the marketplace. Finally, we describe how the Auction Manager provides a focal point for expressing policy decisions such as how much to charge for starting and running auctions, as well as who and when to charge.

1 Introduction

At the level of individual computers, operating systems shield applications from system details by providing direct access to general system services. In the same way, middleware supports transparency in remote services provided over networked information systems. In the realm of electronic commerce, the role of middleware is to support the basic operations that might take place within a commercial interaction. Although the varieties of commerce activities are legion, as long as they share some fundamental components, reusable middleware services can offer substantial leverage. For example, the fundamental step of executing an exchange transaction is common to many contexts, and so general infrastructure supporting this operation will have many uses.

A significant burden of middleware, however, is to support a diversity of commercial communities—equity and commodity markets, manufacturing supplier chains, mail-order retail, and publishing, just to name a few. Each such community comes with its own particular vocabularies, institutions, and conventions, evolved for its own purposes and entrenched to varying degrees. This suggests that despite any fundamental commonalities, commerce middleware services need to be flexible and extensible to support (at least) the range of practices we find in the natural commercial world.

An agent wishing to participate in a commercial interaction (an *exchange* of some good or service, which we generically call a *good*) must face each of the following questions on entry to a commerce environment:

1. How can I describe what I want to exchange?
2. Where can I exchange the good and under what terms?
3. Who can I exchange with?
4. How can we execute the transaction?

Addressing each of these questions constitutes a step in a comprehensive commerce process. More specialized commerce environments, such as business-to-consumer commerce, might include additional steps such as promotion. The role of commerce middleware is to serve agents taking these steps. One can conceive of reusable support services targeted at individual steps (or parts of steps), or those spanning multiple steps [9].

Probably the most well-defined of these steps is the fourth—executing the transaction—and indeed, the largest share of middleware services termed "electronic commerce" primarily address this step. Early payment protocols, such as those developed by First Virtual (http://www.firstvirtual.com/) and Digicash (http://www.digicash.com/), as well as proposed generic frameworks [11], clearly fall in this category. As pointed out by MacKie-Mason and White [14], even this relatively circumscribed step presents interesting design choices along many dimensions.

Probably the least well-defined step is the first—describing the good. Ongoing work in developing descriptive metadata for e-commerce ranges from formalizing license terms descriptions [19] to more ad hoc discussions of XML-based content definitions such as CommerceNet's XML exchange (http://www.xmlx.com).

In this paper, we focus on middleware services for step two. However, we build on the growing presence and increasing understanding of the value of on-line auctions [20] for step three—setting the terms of an agreement and matching buyers with sellers.

We describe here an *Auction Manager*, a commerce middleware component designed to simplify and automate both the creation of new markets and the matching of agents to existing markets. Some standalone products aim to provide some of this function. For example, shopping agents, such as Bargain Finder (http://bf.cstar.ac.com/bf) and Jango [6] (http://www.jango.com), provide consumers with compilations of Internet vendors' prices for products such as music CDs or software. In contrast, middleware for this type of task is more infrastructural, serving individual functions as generally as possible, in the context of supporting an overall commerce process. In particular, generic infrastructure for step two could not assume a market model where vendors announce a fixed price for consumers to take or leave. Rather, there might be many modes of negotiation, which market-matching services should take into account in identifying potential matches.

The Auction Manager operates within a dynamic environment, by matching descriptions of goods to existing markets and, when appropriate, creating new markets. While *market* is often used as a non-technical term, we use it here to refer specifically to an auction and its authorized types of participating agents (e.g., stock markets permit only certain authorized broker agent types to participate).

Implicit in the Auction Manager's support for market-matching operations are questions of market policy. For example, when and for what goods should new markets be started? How does the system account for market creation costs? How are community rules, norms, and objectives (if any) expressed—through regulations or incentives?

The Auction Manager was built as part of the University of Michigan Digital Library [2] (UMDL) commerce infrastructure. Whereas UMDL's goal is the provision of library services to library users, the explict realization of that goal is to provide a commerce infrastructure that supports the process of describing, locating, and negotiating for a wide variety of information services.

However, this commerce infrastructure is not restricted or specific to digital libraries. Since UMDL cannot know at design-time what services will be available in the future or what the best negotiation mechanisms are for any given situation, its languages and protocols have been designed for flexibility and extensibility. One of the advantages of this flexibility is that UMDL provides a testbed, allowing us to experiment with different mechanisms in different contexts.

Within the UMDL, library exchanges focus on a particular kind of good—information services. Therefore, in the next sections we often refer to goods when describing abstract exchanges, while referring to particular information services in our more concrete examples.

In the next section, we describe our generic commerce infrastructure. Our main aim is not to describe the existing UMDL architecture[1] in detail, but to give an overview of the context in which the

[1] Ongoing UMDL work can be found at http://www.si.umich.edu/UMDL.

Auction Manager operates. In Sections 3 and 4, we focus on the commerce middleware services of the Auction Manager.

2 Commerce Infrastructure Overview

One part of the commerce infrastructure is the interaction framework: description languages for goods, and negotiation and exchange protocols, which we describe in Sections 2.1 and 2.2. The second part comprises the various infrastructure components, described in Section 2.3, which simplify and automate the use of these languages and protocols.

2.1 Description languages for goods

Description languages, or ontologies [8], permit a large number of potential goods and their attributes to be captured as a taxonomy. Using these structured objects, we can identify and reason about classes of goods [4]—a powerful base for reusable automation.

For example, one common library service is that of answering queries. A query planning service will, based upon the user's query, search the appropriate collections for relevant items, possibly employing a thesaurus or other indexing service. Query planning services may differentiate themselves through the different attributes associated with them, such as target audience (Professional, High School, Middle School), topic (Science, Art, Literature), and recommending organization (ACM, IEEE, National Education Association). Figure 1 shows a simple taxonomy of possible query planning audience and topic attributes.

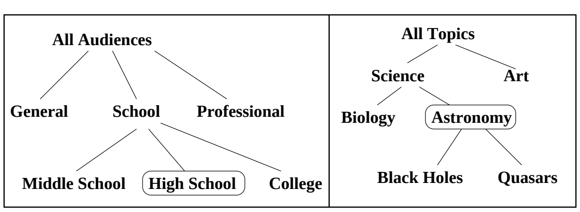

Figure 1: Query Planning: audience and topic attributes

Clearly *High School* audiences are just a particular kind of *School* audiences and *Astronomy* topics are a subclass of *Science* topics. Thus, a service that can respond to queries about High School Science will also be able to respond to queries about High School Astronomy. We discuss the opportunities, and special considerations, involved in using this sort of inferencing to match agents with markets in Section 3.1.

2.2 Negotiation

Given the enormous number of potential goods and the dynamic conditions of the marketplace, there is a concomitant need for a variety of negotiation mechanisms. For example, if a seller has a monopoly good in the current marketplace, the same good would be sold under very different terms than in a more competitive marketplace.

Auctions, which are simply a set of rules for determining a price and/or allocation based on a bidding protocol [16], provide a very flexible negotiation framework—each different auction institution can have a large effect on trading outcome. In effect, auctions are just another service—they provide matching and price setting for buyers and sellers. Auctions also promote automated negotiation through the following characteristics:

- **Mediated**
  Every buyer does not have separately find and contact every seller.
- **Price, not barter**
  Price minimizes and simplifies communication.
- **Formal**
  Standardized, structured offers and auction rules simplify communication as well as processing of offers.

We capture information about the different auction rules and protocols in a compact, reusable manner by using parameterized auction descriptions [18, 25]. For example, an auction's attributes include how often it clears, its price determination rules (e.g., first price, second price), the allowed number of buyers and sellers, as well as what information is publicly available. A Vickrey auction can be described as having a price determination rule of second price, many buyers and one seller, and where the publicly available information does not include the bids (since it is sealed-bid). By changing these kind of auction parameters, we would expect to get a wide variety of behaviors and outcomes. These parameterized auction descriptions provide a description language, or standard vocabulary, for auctions.

2.3 Infrastructure components

Infrastructure components simplify, augment, and automate the use of the underlying goods and services description languages. They do this by encapsulating reusable services common to different steps in the commerce process. We describe our commerce infrastructure components below, as they would be invoked in a typical scenario, diagrammed in Figure 2.

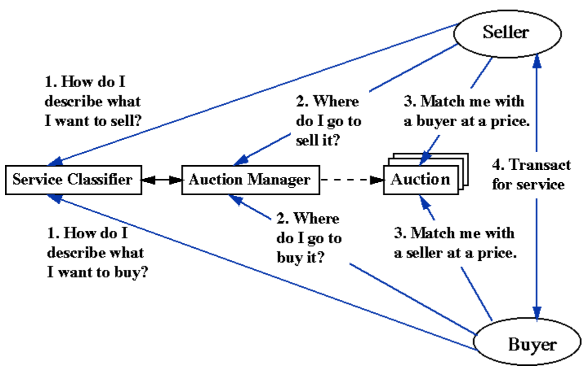

Figure 2: Infrastructure components

Initially, every agent or auction registers with the Registrar (not pictured) and receives an agent-id or auction-id, which uniquely identifies it and is used for communication within the system.

Step one in the information exchange process is for a buyer or seller agent to describe what good it wishes to exchange. For example, agents might describe the particular kind of query planning service they are providing or seeking in terms defined in some standard vocabulary. An agent sends this service description to the Service Classifier [22]. The Service Classifier classifies the service within a comprehensive taxonomy of information services and responds with a service label. The service label acts as a tag identifying a classified service within the system.

In step two, agents want to locate auctions where they can buy or sell the service. An agent sends the service label to the Auction Manager, adding information about particular auction attributes if they wish. The Auction Manager informs an agent of existing auctions for that service or, if necessary, creates a new auction.

In step three, each agent chooses one or more auctions to participate in. Buyer and seller agents are matched and a transaction price is set according to the rules of the auction. In the last step, the two agents transact and exchange the service.

3 Market Management Services

In this section, we focus on the market management services encapsulated by the Auction Manager. In the process of building and using the Auction Manager service within UMDL, we identified three core services: market matching for buyers and sellers, notification of new markets, and data collection and information dissemination. We describe each of these services below. For our examples, we use the query planning services introduced in Section 2.1.

3.1 Market matching

In a large-scale and dynamic commerce environment, the Auction Manager's role of matching agents with the appropriate markets is nontrivial. If the exact market that the agent has requested exists, then all the Auction Manager has to do is return that market. However, given the wide variety of possible service and market descriptions, an exact match may not exist. Even if an exact match exists, the agent may wish to know about all markets in which it could trade—there may be tradeoffs between price and quality that only the agent can make. For example, suppose there are two query planning markets: one in High School Science, and one in High School Astronomy. If the High School Astronomy market is smaller and the quality higher (because it is more specialized), then the price may be higher. An agent wanting High School Astronomy query planning has to make a choice between the higher quality of the Astronomy market versus the lower price of the more general Science market.

Given that we can express these different goods and markets in some description language, how do we reason about them in a market context? In particular, what kinds of market-specific operators and inference rules are needed?

3.1.1 Market-specific operators

In the query planning service example above, we have made some assumptions about what it means in the taxonomy for the topic Science to be more general than Astronomy in a market context. However, there are a number of different ways in which we could interpret this taxonomy.

Consider the difference between a Science query planning service that is composed of *bundled* Astronomy and biology query planners and one that makes *undifferentiated* queries across any and all Science sources. We can describe the *bundled* ($\otimes$) technology as one that can be easily decomposed, or unbundled, into several subparts whereas an *undifferentiated* ($\odot$) technology cannot. Next, assuming that Science can be unbundled into separate Astronomy and Biology query planners, is it the buyer or seller who chooses which service will be used? We describe *buyer's choice* using the operator Ⓑ and *seller's choice* with Ⓢ.

These four logical operators on market descriptions are shown below:

Operator	*Example*	*Meaning*
Ⓑ[class c]	Ⓑ[*Science*]	Buyer's choice of subclasses of Science
Ⓢ[class c]	Ⓢ[*Science*]	Seller's choice of subclasses of Science
$\otimes$[class c]	$\otimes$[*Science*]	Bundled subclasses of Science
$\odot$[class c]	$\odot$[*Science*]	Undifferentiated Science

In the case of query planning services, it seems unlikely that a buyer is going to want to let the seller choose the topic (i.e., seller's choice). Indeed, within the UMDL, we make the default assumption that all query planning services are buyer's choice, considerably simplifying the search for related markets.

However, for other goods seller's choice is a reasonable operator. Consider a newspaper subscription—subscribers don't choose the articles that appear in each day's paper. In a buyer's choice article market, buyers would pick and choose among available articles, essentially creating their own customized paper. Each of these two choice operators composes individual articles into a different, more abstract, bundled article market.

The idea of bundled goods is far from new, and has been generally addressed in economic literature as an alternative technique for price discrimination [17]. By reducing the dispersion in consumer tastes across different bundles a seller can often extract more value from transactions than uniform pricing would. Although historically goods have often been unprofitable to bundle, in the case of information goods and services (with marginal costs close to zero) selling bundled goods can yield higher profits and greater efficiency than selling them separately [3]. One ongoing field trial in this area includes the PEAK project, offering Elsevier journal articles with different experimental bundles and prices [13].

What we introduce here is the ability of the buyer or seller to choose a sub-bundle of the bundled goods, as defined in Table 2. These are not the only possible ways of choosing sub-bundles; the set of market operators could be extended.

Operator	*Definition*
Bundle	$\otimes$[*superclass*] $\equiv$ $\otimes$[$\otimes$[$class_1$], ..., $\otimes$[$class_n$]]
Example	$\otimes$[*Science*] $\equiv$ $\otimes$[$\otimes$[*Astronomy*], $\otimes$[*Biology*]]
Buyer's Choice	Ⓑ[*superclass*] $\equiv$ Buyer chooses one of Ⓑ[$class_1$], ..., Ⓑ[$class_n$]
Example	Ⓑ[*Science*] $\equiv$ Buyer chooses one of Ⓑ[*Astronomy*] or Ⓑ[*Biology*]
Seller's Choice	Ⓢ[*superclass*] $\equiv$ Seller chooses one of Ⓢ[$class_1$], ..., Ⓢ[$class_n$]
Example	Ⓢ[*Science*] $\equiv$ Seller chooses one of Ⓢ[*Astronomy*] or Ⓢ[*Biology*]

Table 2: Market Operator Definitions.

In the next section, we describe how to use this notation for automatic matching of buyers and sellers to markets.

Service Market #	*Trade Context*	*Query Planning Service Attributes*		
		Audience	*Topic*	*Recommending-Org*
1	buy, sell	High School	Ⓑ[Astronomy]	unspecified
2	buy	High School	Ⓑ[Astronomy]	Astronomy Today
3	buy	Ⓑ[School]	Ⓑ[Astronomy]	unspecified
4	sell	High School	Ⓢ[Science]	unspecified
5	sell	Ⓢ[School]	Black Holes	unspecified
6	sell	unspecified	Ⓑ[Astronomy]	unspecified

Table 1: Existing Markets 1-6

3.1.2 Market-specific inferencing

Finding all the potential markets for an agent to trade in requires that the Auction Manager employ not only the taxonomy of good descriptions and market operators; it also needs to know whether the agent wants to buy or to sell a particular good.

To see why this is necessary, consider two agents, one who wishes to buy a query planning service for High School audience for Astronomy topics and one who wishes to sell it. Note that in these descriptions we refer back to the simplified version of the audience and topic attributes taxonomy shown in Figure 1. In table form this service description looks like:

Service	*Service Attributes*	
	Audience	*Topic*
Query Planning	High School	Ⓑ[Astronomy]

We represent the Astronomy topics as buyer's choice because that makes the most sense for a query planning service. Since High School has no subclasses under it, it was not necessary to use an operator to describe how it is bundled. Next, suppose that Markets 1-6 in table 1 already exist. Notice that these markets may include additional attributes, such as *Recommending-Organization*, which were not specified in the agent's service description. Similarly, the markets may also lack attributes which are specified by an agent.

Both buyers and sellers could participate in Market 1, since it exactly matches the service description. However, in Markets 2-6, whether an agent can participate depends on the trading context (i.e., whether the agent wants to buy or sell the good).

In Market 2, the seller cannot participate since it has not been recommended by Astronomy Today. On the other hand, whether a service has been recommended or not is irrelevant to the buyer, so it can participate.

In Market 3, a buyer can always choose High School audience among the different kinds of School audiences offered. However, the seller's service is limited to High School audiences and it cannot participate in a School audiences market because buyers might want to select Middle School audience or Professional audience.

In Market 4, a seller can participate since Astronomy is a Science and the seller has the right to choose which kind Science it provides—in this case the seller would always choose Astronomy. However, if a buyer were to participate, the query planning service the buyer got matched with might very well be for Biology rather than Astronomy.

In Market 5, a seller who can offer any Astronomy subtopics, can simply unbundle the Astronomy subtopics and offer Black Holes separately. On the other hand, a buyer interested in Astronomy topics may not want to confine its queries to Black Holes but may be interested in Quasars also.

Lastly, in Market 6, the seller can participate in a market with unspecified audiences, since buyers haven't specified (don't care) what kind of audience the query planning service is aimed at. However, a buyer who specifically wants a High School audience cannot.

The informal rules expressed in the example above can be captured and used by the Auction Manager to automatically match potential markets.

Rule	*Example*
Generalize Attribute Class	$Operator_1[class] \rightarrow Operator_2[superclass]$
When buying:	Ⓑ[Astronomy] → Ⓑ[Science]
	Ⓢ[Astronomy] → Ⓑ[Science]
	⊗[Astronomy] → Ⓑ[Science]
	⊙[Astronomy] → Ⓑ[Science]
When selling:	Ⓑ[Astronomy] → Ⓢ[Science]
	Ⓢ[Astronomy] → Ⓢ[Science]
	⊗[Astronomy] → Ⓢ[Science]
	⊙[Astronomy] → Ⓢ[Science]
Choice Operators	$Operator_1[class] \rightarrow Operator_2[class]$
When buying:	Ⓢ[Science] → Ⓑ[Science]
When selling:	Ⓑ[Science] → Ⓢ[Science]
Unspecified Attributes	
When buying:	unspecified → anything
When selling:	anything → unspecified

Table 3: Market inference rule examples.

For example, the Generalize Attribute Class rule in Table 3 says that a buyer who wants topics on Astronomy, can be satisfied in a market for buyer's choice Science. Similarly, a seller who sells seller's choice topics in Astronomy, can sell in a seller's choice market for Science topics—for any customer, the seller can always choose Astronomy as the topic. By writing potential trades between markets as transformation rules, we can use forward and backward chaining to ask questions, such as what are all the potential ways that this product could be sold.

These rules model how agents can bundle and unbundle goods and choose from among bundled goods. However, there is no guarantee that these rules will terminate. In practice, both the number of markets and the existing UMDL ontology are sufficiently small that this has not been an issue. In the future, we will need to place restrictions on the rules of inference to be able to guarantee termination. Similar problems for automatic checking of security protocols using formal logics were addressed by an automatic theory-checker generator [12].

3.1.3 Automatic market arbitrage

Our discussion of the inference rules in table 3 above has implicitly made the assumption that selecting the appropriate good from several others is a simple operation. For example, the Generalize Operator rule implies that an agent who wants to buy a seller's choice Science query, Ⓢ[Science], can buy a buyer's choice Science query, Ⓑ[Science], and pick an arbitrary Science topic. There is clearly a potential trade between the buyer and seller, which can occur just by making an arbitrary selection between topics. If the choice is process is well-defined, it could be automated either within an individual agent or by having a specialized Ⓑ-to-Ⓢ *arbitrage agent* created on demand to link the two markets.

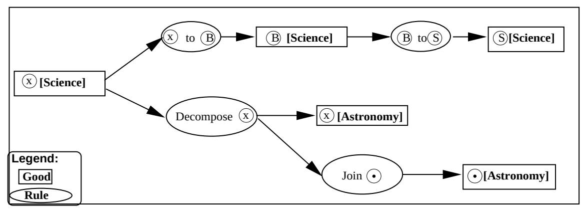

Figure 3: Arbitrage Rules on structured Query Planning Services

Figure 3 shows how these rules, using appropriate control of the rule execution process, can generate a large number of different products, based on a few simple transformations. This suggests that one-time requests for products requiring simple transformations may be more efficiently handled through automatically generated arbitrage agents or by encoding simple transformations within agents, compared to starting up an entirely new market.

3.2 Notification

Once a new market has been created, initially only the Auction Manager and creating agent know about it. For other agents to be aware of the new auction, either they will have to periodically check for new auctions or rely on a notification service to alert them, depending on their individual needs. The Auction Manager serves as a focal point for our notification services since it both creates markets and accepts the original service requests from agents.

Whenever an agent requests a market, it can ask the Auction Manager to notify it if any new matching markets are created. In order to determine if an agent would be interested in a newly created auction, the Auction Manager compares its service label to the service label of the newly created auction using the market matching described in Section 3.1. If the agent's service can be exchanged there, then the Auction Manager sends a notify message to the agent telling it the new auction-id.

3.3 Data collection and information dissemination

Markets are not only means of trading, but also a source of information about the price and other terms under which trades may be made. Of course, revealing this information may itself distort agent behavior, and whether it will turn out to be a good or bad idea to disseminate any market information remains an open question. Two sources of market data are described below.

One source is the data logged by the auctions. This includes consumer and producer bid values, time of bid arrivals, number and time of transactions, as well as clearing prices. From this information, summary data such as bidding frequency, average price, and price variance can be produced.

Another source of market data is the original service request from an agent. Sometimes the Auction Manager can exactly match an agent to a market, otherwise the agent has to participate in a near-match market. However, in the second case, the Auction Manager knows both the original service demanded and that an exact match for that service could not be found. This information may be valuable to sellers, upon being made aware of unmet demands by buyers for services, they could estimate whether it would be worthwhile be willing to invest in developing or specializing their agents to meet that demand. By collecting and providing this kind of data about current market activity and demand for services, the Auction Manager can provide agents with information on which to base their decisions about how to trade their services or when it may be worthwhile to develop new ones.

Such market data can be used offline to evaluate the effect that different market creation and selection policies had on the market configuration and resulting system efficiency and welfare. However, exactly under what circumstances this information should be made available, and to whom, depends on the commerce community and is a matter of market policy.

4 Market Policies

Market policies can serve to support and uphold established business practices for a community as well as account for system externalities such as market creation costs. Policies can be implemented through either rules or incentives. For example, policy issues such as information access management [1], which determine who may access what collections and under what terms and conditions, are more naturally handled as rules. On the other hand, objectives such as increasing some social welfare criteria or providing a base level of library services to all patrons, may be more naturally implemented as incentives, perhaps subsidies or taxes.

Our focus is on policies related to market management costs. Since the number of possible markets is virtually unbounded, while the amount of network resources and agent attention is not, we consider what kinds of policies need to be employed to restrict market creation and select reasonable markets. In Section 4.1, we discuss the sources of market creation costs. In Section 4.2, we discuss the problem of who should decide what new markets can be created.

4.1 Market creation costs

In an ideal system, where running an auction is a cost-free proposition and agent decision costs are not considered, there would be no reason not to allow any and all markets requested. However, this ideal environment does not exist; running an auction is not cost-free.

Infrastructure costs include the computational and network costs of actually running the auction and—even more importantly—the resultant increase in lookup and indexing costs. Clearly these costs must accounted for through a policy which can set the appropriate fees. One simple policy is to have a uniform flat fees charged for infrastructure usage. For example, an auction might either charge a single fee to the auction owner/initiator to run it for a given time period, or charge smaller per-transaction fees to individual auction participants. On the other hand, a library policy might use system-level criteria to promote auctions which the library considers beneficial to the system as a whole, perhaps by charging less for them, or running them for free. A side benefit to having an explicitly stated policy, whether system-oriented or not, is that it could be used to provide guidance when choosing among a number of potential auction specifications for a given product.

Agent decision complexity costs result with the increase in market choices. If an agent can buy a service from several related markets, the agent might handle it in one of two basic ways. The first is to trade simultaneously in multiple markets, facing the risk of ending up with multiple outstanding commitments. The second is to trade in only one market at a time, facing problems of market liquidity where there may be two parties willing to trade, but unable to do so since they are at different markets. Of course, an agent may make sequential offers at different markets, but this results in trading delays and will not necessarily bring all parties together, due to timing difficulties.

The Auction Manager can provide a vehicle to experiment with alternative market-creation policies by evaluating the resultant market configurations.

4.2 Market selection issues

Given the potentially unbounded number of markets that can be created for a partially specified service, how do we decide which market should be created? In the past, market selection has been addressed in work on incomplete markets [10, 15], product differentiation [5], and industrial organization with transaction costs [23]. This work generally involved analyzing the effects for a known and static number of markets.

When neither the exact number nor kind markets can be known ahead of time, methods for comparing and evaluating the benefits of different market configurations [7] will need to be developed. Using such comparisons, we can hope to identify general rules or incentives that promote commerce environments having increased efficiency, profits, surplus or any number of other criteria. We are currently performing small experiments to compare the efficiency and surplus for different market configurations given different buyer and seller value distributions.

In the next sections, we consider the mechanics of two particular market selection processes. The mechanics of the market selection process will have an impact on what kinds of markets can be started as well as the kind of rules and incentives which can be used. In particular, who makes the decisions about what auctions to create, is it the agents (where agents can be humans or software) or the Auction Manager? We argue that in some circumstances it may be more reasonable to have agents decide while in others the Auction Manager, and both should be supported. We also discuss the kinds of information and incentive requirements needed for each situation.

4.2.1 Agents decide

Under certain circumstances, it will make more sense, or at least be more realistic, for agents to determine which markets are created. For example, when an agent represents an outside vendor, especially a brand-name vendor, the vendor may very well want to establish its own market. Also, as buyer agents and/or recommending organizations learn which information providers and kinds of services they prefer, they should be free to establish markets which capture these preferences [21].

However, allowing agents to decide which markets to create means that the externalities involved in creating an auction, such as the costs to the system and other users, must be internalized in the form of auction fees. The fees need to be set so as to provide the right incentives for agents not to abuse the auction creation mechanism. We are in the process of exploring how to set these fees. Some requirements are that the fee structure needs to be kept simple, as complicated schemes may create additional costs that overwhelm the original system costs involved. Reasonable choices include setting a small flat fee per auction transaction, a small percentage fee per transaction or a rental fee per time period that the agent who creates the auction pays.

In deciding whether to establish a new market, an agent may find it useful to consider information about related product auction prices, transaction volumes and frequency, and unmet user demands. This was discussed in Section 3.3

4.2.2 Auction manager recommends

The Auction Manager can provide an auction recommendation service by using auction- and good-specific knowledge to pick reasonable auctions. Thus having the Auction Manager recommend auctions is not necessarily incompatible with having agents deciding which auctions they want to create. Agents can specify which service or auction attributes they care about and the Auction manager can fill in the rest.

Some of these defaults are independent of the particular context. For example, no one would want an auction for immediate query planning service to have a clearing time of once a day. Also, if a market is inactive for a certain amount of time it makes sense to have it deactivate itself, since it can always be restarted if necessary. However, choosing the appropriate auction(s) will generally be affected by user preferences, technology factors, and the market environment. If an agent is a monopolist then its definition of the best kind of auction will differ from one for a library agent who charges its marginal cost.

A key question is how much does the Auction Manager know about participating agents and what are the criteria that it uses to determine the best auctions. In the case of the outside vendors, we may be able to support their choice of auctions in two ways. The first is simply by using defaults to fill in partially specified auctions with reasonable values. The second is by using economic theory to set up auctions depending on what kind of agent requests it. For example, the Auction Manager can choose an auction which is incentive compatible for buyers or for sellers, but not both [24].

In the case of library agents, where the focus is more oriented towards designing simpler agents to allocate library services efficiently, the Auction Manager could have a library policy to determine which are the best auctions to create. Due to the complexity of the auction design space, a realistic library policy would have to be expressed in simple, qualitative terms such as more market efficiency is better, lower transaction costs are better, less price volatility is better, and so forth.

5 Conclusion

The Auction Manager is a market middleware component providing services to support automated negotiation in a large-scale electronic commerce systems. Specifically it supports this by generating and tracking auctions, matching agents to potential markets, and providing a means to notify agents when markets of interest to them are created. We have described how the Auction Manager can use market-specific knowledge to recommend auctions and how it can serve as a focal point for information collection and dissemination.

Future work involves using the Auction Manager to experiment with different auction creation policies and auction pricing policies and, based on the resultant market configurations and market data, evaluate their effectiveness in promoting specified system policies.

Acknowledgments. This work was supported by the NSF/ARPA/NASA Digital Library Initiative. Participants in the UMDL project have contributed to the concepts and work described here, especially William Birmingham, Edmund Durfee, Jeffrey MacKie-Mason, Anisoara Nica, Sunju Park, José Vidal, William Walsh, and Peter Weinstein.

References

[1] William Yeo Arms. Implementing policies for access management. *D-Lib Magazine*, February 1998.

[2] Daniel E. Atkins, William P. Birmingham, Edmund H. Durfee, Eric J. Glover, Tracy Mullen, Elke A. Rundensteiner, Elliot Soloway, José M. Vidal, Raven Wallace, and Michael P. Wellman. Toward inquiry-based education through interacting software agents. *IEEE Computer*, 29(5):69–76, May 1996.

[3] Yannis Bakos and Erik Brynjolfsson. Bundling information goods: Pricing, profits and efficiency. In *Conference on Economics of Digital Information and Intellectual Property*, Harvard University, January 1997.

[4] Alexander Borgida. Description logics in data management. *IEEE Transactions on Knowl-*

edge and Data Engineering, 7(5):671–682, October 1995.

[5] Avinash K. Dixit and Joseph E. Stiglitz. Monopolistic competition and optimum product diversity. *The American Economic Review*, 67(3):297–308, June 1977.

[6] Robert Doorenbos, Oren Etzioni, and Daniel S. Weld. A scalable comparison-shopping agent for the world-wide web. In *First International Conference on Autonomous Agents*, pages 39–48, 1997.

[7] Robert P. Gilles, Dimitrios Diamantaras, and Pieter H. M. Ruys. Efficiency, valuation and cores in economies with costly trade. Technical Report E96-01, Department of Economics, Virginia Tech, 1996.

[8] Thomas R. Gruber. Toward principles for the design of ontologies used for knowledge sharing. *International Journal of Human-Computer Studies*, 43(5/6), 1995.

[9] Robert H. Guttman, Alexandros G. Moukas, and Pattie Maes. Agent-mediated electronic commerce: A survey. In *Knowledge Engineering Review*, volume 13, June 1998.

[10] Frank Hahn, editor. *The Economics of Missing Markets*. Clarendon Press, Oxford, 1989.

[11] Steven P. Ketchpel, Hector Garcia-Molina, Andreas Paepcke, Scott Hassan, and Steve Cousins. U-PAI: A universal payment application interface. In *Second USENIX Workshop on Electronic Commerce*, pages 105–121, November 1996.

[12] Darrell Kindred and Jeannette M. Wing. Fast, automatic checking of security protocols. In *Second USENIX Workshop on Electronic Commerce*, pages 41–52, November 1996.

[13] Jeffrey K. MacKie-Mason and Juan F. Riveros. Economics and electronic access to scholarly information. In B. Kahin, editor, *The Economics of Digital Information*. 1998. (forthcoming). http://www-personal.umich.edu/ jmm/papers.html.

[14] Jeffrey K. MacKie-Mason and Kimberly White. Evaluating and selecting digital payment mechanisms. In G. Rosston and D. Waterman, editors, *Interconnection and the Internet*, pages 113–134. Lawrence Erlbaum, 1997.

[15] Michael Magill and Martine Quinzii. *Theory of Incomplete Markets, Volume 1*. The MIT Press, 1996.

[16] R. Preston McAfee and John McMillan. Auctions and bidding. *Journal of Economic Literature*, 25:699–738, 1987.

[17] R. Preston McAfee, John McMillan, and Michael D. Whinston. Multiproduct monopoly, commodity bundling, and correlation of values. *Quarterly Journal of Economics*, 104(2):371–384, May 1989.

[18] Tracy Mullen and Michael P. Wellman. Market-based negotiation for digital library services. In *Second USENIX Workshop on Electronic Commerce*, pages 259–269, November 1996.

[19] Godfrey Rust. Metadata: The right approach. *D-Lib Magazine*, July/August 1998. http://www.dlib.org/.

[20] Efraim Turban. Auctions and bidding on the internet: An assessment. *International Journal of Electronic Markets*, 7(4), 1997. http://www.electronicmarkets.org.

[21] José M. Vidal and Edmund H. Durfee. Learning nested agent models in an information economy. *Journal of Experimental and Theoretical Artificial Intelligence*, 10(3):291–308, 1998.

[22] Peter Weinstein and William P. Birmingham. Runtime classification of agent services. In *Proceedings of the AAAI-97 Spring Symposium on Ontological Engineering*, Stanford, Palo Alto, CA, March 1997.

[23] Oliver E. Williamson. Markets and hierarchies: Some elementary considerations. *The American Economic Review*, 63(2):316–325, May 1973.

[24] Peter R. Wurman, William E. Walsh, and Michael P. Wellman. Flexible double auctions for electronic commerce: Theory and implementation. *Decision Support Systems*, to appear.

[25] Peter R. Wurman, Michael P. Wellman, and William E. Walsh. The Michigan Internet AuctionBot: A configurable auction server for human and software agents. In *Second International Conference on Autonomous Agents*, pages 301–308, May 1998.