Electronic Commerce
with Software Agents

Will Clurman
Tim Foley
Rob Guttman
Kristin Kupres

March 14, 1997
Table of Contents

1. INTRODUCTION ............................................................................................................... 2

2. THE DEFINITION OF A SOFTWARE AGENT .............................................................. 2
   2.1 HOW SOFTWARE AGENTS WORK ........................................................................... 2
   2.2 SOFTWARE AGENTS’ DISTINGUISHING FEATURES ........................................... 3

3. A FRAMEWORK FOR UNDERSTANDING AGENTS ............................................... 3
   3.1 PERSONALIZATION ............................................................................................. 4
   3.2 BROKERING ......................................................................................................... 5
   3.3 NEGOTIATION ....................................................................................................... 6
   3.4 SUMMARY OF THE FRAMEWORK ....................................................................... 8

4. NEXT STEPS IN OUR RESEARCH ........................................................................... 8
   4.1 ONTOLOGIES ....................................................................................................... 9
   4.2 REPUTATION/TRUST ........................................................................................ 9

5. CONCLUSION ............................................................................................................. 10
1. Introduction

An amorphous object, called a software agent, is demanding a lot of attention in the Internet world. Can it really be the panacea for the information-overloaded state we Internet-savvy Web users find ourselves in? How are software agents being used today to personalize and expedite Web-based transactions?

This paper examines software agents, their definition, and how they are being applied to Electronic Commerce. In addition to defining a software agent, we describe a framework for evaluating software agents being used in Electronic Commerce applications.

2. The Definition of a Software Agent

2.1 How software agents work

Agents are not clearly defined yet they have been used in the field of computer science for the past ten years. Agents mean “a lot of different things to a lot of intelligent people”\(^1\) and may be more easily described by their capabilities as they relate to particular areas within computer science.

In its simplest form, an agent is a piece of software written in a software language; Java and C++ being some of the more common languages. An agent lives on the client or server machines and can be categorized as an anchored or mobile agents. Anchored agents do most of their work on either the client or the server. Alternatively, mobile agents roam among different servers collecting the information they need at each server. Software agents are being used in Web applications today to personalize information retrieval tasks on the Web; increasing the search efficiency and quality of the resulting data.

The caveat with software agents is that they are only as smart as their developer’s ability to implement a feature set for an agent. The task of searching multiple Web sites is a good example the complexity in developing a software agent. Given that the Web is not standardized, in order for an agent to search multiple sites, it must understand how to parse the information available at each site. As a result, the developer is forced to create

a parsing module for each different Web site the agent needs to poll in real-time. The BargainFinder agent is an example of this type. There do exist more sophisticated agents like ShopBot from NetBots, Inc. which can learn to navigate and parse new sites (and relearn changed sites). The alternative to implementing complicated agents like ShopBot is to build a database using a simple agent with only one parsing module, but this requires that the participating web sites use common ontologies and negotiation protocols.

2.2 Software Agents’ Distinguishing Features

Software agents differ from traditional software because they exhibit three traits not commonly found in the traditional software, which include:

- the ability to personalize,
- the ability to be proactive, and
- the ability to be adaptive.

Personalization is the capturing of the unique goals of the user and acting accordingly on his/her behalf. Personalization is apparent for example in a user’s ability to customize informational searches. The ability to personalize software agents allows the user to construct their individual preferences for searching an on-line database, a list of sites, and even transacting on your behalf.

A software agent is proactive if it has the ability to complete its task independent of the user’s activities. An agent that finds new information for the user, without the user requesting a response is an example of a the ability for the agent to be proactive.

The ability to be adaptive is described as the software agent’s ability to learn from its past actions and from its environment. Learning from past actions is characteristic of the artificial intelligence (AI) technology which preceded much of today’s agent technology.

3. A framework for understanding agents

Intelligent Agents are to the networked world what Artificial Intelligence was to the computing world. The concept is pregnant with promise but just as nebulous as AI. In order to gain a clearer understanding of what agents do and what they are good for in electronic commerce, we chose to focus on three primary dimensions in which agents can add value:
1. *personalization*: information filtering  
2. *brokering*: information gathering and retrieval  
3. *negotiation*: dynamic and flexible execution of transactions

By defining each of these dimensions we hope to develop a framework for assessing what an agent must be good at doing if it is to provide a solution to a specific set of problems. Such a framework would be useful in comparing existing solutions as well as in the design of new agents and perhaps may even yield insights into new approaches to agent-based solutions.

### 3.1 Personalization

The essential feature of an agent is its specificity with respect to a certain function or task. This is the sense in which it is literally an “agent” taking action in the interests of another just as a real estate agent is understood to represent the interests of a buyer or seller. But a software agent is considerably simpler than a human agent as software agents exist solely to perform the tasks for which they are custom-built. To the extent that the outcome of the agent’s primary task is controlled by specific information supplied by the user, the agent is *personalized* for that user.

A search engine such as Yahoo! or Alta Vista is an example of a software program operating on information supplied by a user and illustrates a simple form of personalization. In this case the user-supplied information is not stored by the agent for future filtering tasks, but rather each time a user wishes to obtain results that meet his personal interests, he must supply a term or terms that correspond to these interests. These search criteria act as user profile data on which the agent operates to deliver filtered search results customized to the user’s information needs.

More sophisticated filtering agents store user profile data for multiple, repeated operations such as Mercury Mail’s email publishing services. Mercury Mail delivers time-sensitive information as stock quotes, national weather conditions and sports stories direct to users’ email accounts. AdOne, an aggregator of classified ads from community newspapers across the US offers the AdHound, an agent which stores user profile data for the purpose of notifying the user when there are ads that meet her stated preferences by item type (“computer,” “collectible,” “video game,” or “miscellaneous” (*sic*)). See Exhibit 1 for one of AdHound’s personalization screens.
3.2 Brokering

Agents may also serve to obtain desired information, often from a variety of sources, and supply it back to the user. In the context of electronic commerce this activity brings information from the buyer and seller together and presents it to either the buyer or the seller or to both. Brokering is different from personalization in that brokering is the function of information retrieval and delivery, given personalized instructions. Again, the search engine that delivers query results is a simple example of a broker, as it gathers and delivers results based on personalized input. So-called “meta-indexes” perform this function on more than one data source.

Services such as Andersen Consulting’s BargainFinder are sophisticated brokers designed to aid in online shopping applications by gathering information from websites (nine websites, in the case of BargainFinder) and delivering the price and shipping terms of a certain good requested by the user. See Exhibit 2 for a sample of the information delivered by BargainFinder. In that BargainFinder, through its agency, brings the user closer to a buying transaction that meets the user’s personal criteria, BargainFinder is a useful agent to the user.
The extent of this usefulness is limited in most agents by the detail and accuracy of the user profile data inputs held by the agent and the volume and organization of the information upon which the agent operates. There are also factors extrinsic to the agent itself that increase or decrease an agent’s usefulness such as the availability of the specified good at the time and price at which it is sought. A highly sophisticated brokering agent would likely show a high number of user profile criteria, a high number of online sites at which it shops, and perhaps even a scheme for assigning relative priorities to the different buying criteria supplied by the user. These priorities might be assigned directly by the user but a complex agent might also learn these priorities by capturing actual purchasing decisions, checking them against the user’s previously stated buying preferences and then adjusting the weight of the priorities assigned to the different preferences. Applications in which such a “weighted average” prioritization methodology would be useful are transactions involving several different buying decision criteria and a large number of possible options from which to choose, each of which performs differently in the different decision-making categories. Such a system, if possible, would be useful in complex buying decisions such as car or house purchases.

3.3 Negotiation

The third aspect of value added by agents in electronic commerce is their usefulness in the dynamic process of transacting a deal. An example of this kind of functionality exists in the “program trading” that takes place on Wall Street today, in which software
programs are constantly watching for the occurrence certain conditions or the crossing of pre-assigned thresholds by the changing prices of securities that trade over time in financial markets.

Negotiating agents may be empowered to execute transactions on behalf their users. Although there are no websites at which we observed this kind of empowered agency at work, it is nevertheless easily imaginable. As there is a cost in terms of time to accepting and reviewing information supplied by the activity of an agent, so there may be a threshold value past which the user decides that it is more costly to spend time sorting through choices presented by the agent than it is to let the agent make the buying decision itself. Applications where this kind of agent empowerment seem likely are the purchase of simple goods with relatively few buying criteria or financial products whose values and dimensions are precisely quantifiable by the user.

The Media Lab’s Kasbah project provides an example of a system of buying and selling agents that change the prices they are willing to accept according to given decay functions specified up front by the user. For example, if I am trying to sell a new textbook for a class that I dropped, I might want to ask for a relatively high price until a couple of days before the deadline for returning the textbook to the store, at which time I would be willing to go as low as, or perhaps for the sake of convenience lower than, the refund amount I could get at the store. Kasbah enables a seller to specify a high price and a low price, the period over which the price should drop, and a choice of decay functions for the rate at which the price should drop over the time period. Exhibit 3 shows these options in Kasbah.

Exhibit 3 - Kasbah
Negotiating agents might be useful in the purchase of plane tickets for certain kinds of consumers or in the purchase of grocery items such as sale-priced, economy-size laundry detergents. Imagine an agent that knows when you last purchased laundry detergent and that operates under the rule: “if I have not purchased laundry detergent in the 128 oz. size for at least 4 weeks, then always order one for me, whenever the price is below $2.39.” Such an agent would be useful to the customers and sellers of Streamline and Home Runs.

3.4 Summary of the framework

The dimensions in which we chose to examine the value added of agents in electronic commerce are personalization, brokering, and negotiation. Our purpose was to begin to test the usefulness of these dimensions in clarifying what agents do and how well they perform or might perform given different user needs. While these dimensions overlap somewhat, they nevertheless seem to focus on distinct, and in some cases, quantifiable characteristics with the result that these dimensions offer conceptual handles to understand both what agents do and how they do it.

Whether this framework can also yield insights into the kinds of agents and agent-based solutions we can design is a separate question. There exists a degree of hand-waving and generally confused discourse that surround the topic of intelligent agents in the business and user communities. This is primarily because much of the utility and ramifications of agents in electronic commerce has yet to be analyzed empirically or even qualitatively from an economics perspective. We think that we will be better equipped to do this type of analysis by tying down our observations with specific terminology so that we gain the clarity and common ground necessary to think critically and perhaps creatively about the problems agent-based technologies can solve.

4. Next Steps in Our Research

First we plan on expanding the dimensions in our framework by which agents may be evaluated to include ontologies and reputation/trust. These also happen to be two areas that we identified as being barriers to fully utilizing agents for electronic commerce. Each of these are briefly described below.
4.1 Ontologies

An ontology is a description of the way things are. For example, if you were asked to describe a music CD how many factors would you need and what would they be? BargainFinder is able to capitalize on the rather simple ontology used to describe CDs today: Artist, Album, and Price. Another example would be Sloan’s ontology of careers by which they describe recruiters: banking, consulting, and industry. An ontology delivers a form of understanding that enables more efficient communication. So you can imagine that agents have a much better chance of interacting in a productive manner if they share a common ontology.

We will investigate current ontologies for such things as the product/service description and how ontological problems can be resolved. The Kasbah system, for example, resolved the problem by defining the ontology for both the products and the agents. The environment was entirely controlled within the space of a single server. For such reasons, it may be the case in the future that we see the aggregation of agents and products onto single on-line sites where the ontology and protocols are well defined. To the extent that this starts to look a lot like a standards race, there will probably be an era of many separate sites with different ontologies, yet trading the same products, before they consolidate.

4.2 Reputation and Trust

When your agent brings you the names of five competing sellers of the product you want, and all of their prices are close, how do you choose? Why would you trust one more than the other? Things such as independent rankings could help your agent decide which seller to go with depending on your personalized preferences about price vs. reputation. Your level of trust about a seller can depend on the enforcement policies. What is your recourse if things don’t go well?

In addition to the issue of having agents which can interpret reputation, what about trusting your agent itself? Now that your proactive agent has the ability to transact on your behalf, do you want it to? Do you trust your agent to do the right thing? To help mitigate this problem your agent will likely learn from you. If it has made choices that you like or dislike you can let it know. And maybe someday you can simply ask your agent to
buy a present for your boss and it will do an even better job than you would have on your own without using another second of your time.

Firefly is an example of an “agent” which has been building a reputation and trust. To the extent that Firefly selects new CDs that you like, you will trust it to continue to do so in the future and maybe even trust it to do so with other products like videos.

After expanding our framework, our next step will be to identify which types of agents are best for solving the problems of Asset Specificity vs. Product Description as presented in Professor Malone’s framework.

As the description of the product becomes more complicated even today’s leading agents, like the collaborative filtering mechanism in Firefly, may not translate very well. For example, if you were to use collaborative filtering to rate Web sites and you rated a particular skiing site as being poor, did you rate it poorly because you do not like skiing or because the site itself had a bad design and did not work well. The consequence is that if you like skiing and the site just stunk then the collaborative filter might not recommend any more ski sites for you “thinking” that it was because you did not like skiing.

On this front we will consider how setting priorities for different characteristics of the product could help to extend the boundary of products that can be efficiently traded in the world of electronic commerce.

5. Conclusion

Our conclusion is that agents reduce transaction costs. From the perspective of the user, they can save you time throughout many parts of the commerce process. Most of today’s agents are extremely useful if you know what you are looking for and especially if that thing is not very complicated. By establishing a framework to help get our hands around what an agent is and how it can be useful in this process we hope to both figure out which of today’s agent technologies are best for which part of the commerce process and to discover what agents could do in the future to enable more complicated products and services to be exchanged electronically.