

## ***Intelligent Software Agents In Accounting: an evolving scenario***

**MIKLOS A. VASARHELYI AND RANI HOITASH**

Rutgers University  
Faculty of Management

### ***Executive Summary***

Intelligent agent technology is one of the fastest growing areas of research and Internet related commercial endeavors. It is, however, an ill defined field, with many overstated claims and few specific areas of applications. Both accounting and finance have great potential as fields of application. However, at this stage there are very few, if any, available applications. Most of the applications are still in the primary research stage. For further development of the field, it is necessary to create an operational definition of the field, understand its extant composition, and to postulate a program of research and application development. Such a theoretical work should be of great value as a foundation for an emerging field.

Intelligent Agents today claim some functional "intelligence" where they will perform tasks on the behalf of a user. This paper, explores the spectrum of software agency; from the automated "softbots" that are presently being implemented, to the concepts and projects of the future that are more accurately described as intelligent agents.

The operational definition section should provide some assessment of the current state of intelligent agent technology and who some of the key players are, to-date. The analysis focuses on academic and commercial research. The paper describes the basic mechanics for agency and how agent developers are tackling the challenge of intelligent agents within networked computing environments.

The state-of-the-art section explores the commercial agent landscape. Commercial efforts are just the beginning of the capabilities and potential of intelligent agents. Agents are classified into categories, and examples, from practice, are provided.

Important questions of different nature are raised by this discussion. Some of these questions are technical, others statutory, and some behavioral. For example, does accounting/ finance require special agent technology, what are the laws necessary to allow agents in electronic commerce, and whether consumers will achieve the level of trust for intelligent agents' capabilities to effectively represent them when making important decisions.

### ***Introduction***

The Internet and the World Wide Web continue to grow at a phenomenal pace. As more and more users are exposed to the Internet, the Web is evolving from its early browser-surfing model to an efficient information transfer medium. The phenomenal growth of information sources on the Web has caused many Web users to suffer from an information overload. This information overload has created a new and heightened demand for innovative technologies to alleviate some of the problems. Intelligent agents have been proposed for this task.. There is substantial potential for the development of agents to perform many tasks, from the menial and routine, to the complex and ill-defined. This is driving the agent development from a research environment to a commercial environment. This paper explores first the definition and characteristics of intelligent agents, second the state of the art of its utilization, and third proposes a framework, paradigms, and potential applications of intelligent agents in accounting. Conclusions summarize the paper, discuss some general issues and set a framework for research.

Intelligent agents have come to represent the next level of refinement of the end-user experience with software and the Web. Intelligent software agent theory has evolved from the paradigms of the human agent experience. The goal is to give the intelligent agent a specific mission to achieve on its own, while keeping in mind the desires of the principal.

An **agent** is an entity that performs a task on the behalf of another party, which we here call the **principal**. As the world becomes more closely interconnected, the explosive growth of information available on the Internet, and increased technical competence have made it reasonable to find aspirations for automated (computer) agency. But the promise has been lacking and development has been slow. The aim is to create a virtual world where automated agents can act virtually, making professional and personal lives simpler and potentially richer. It is this vision that calls for intelligent agency maneuvering through the vast (and exponentially growing) sea of information in the Internet and/or corporate Intranets.

The distinction between agents and intelligent agents becomes hazy, particularly when couched in marketing hype. For the purposes of this paper we will attempt to clarify this distinction. In this paper, we will explore what exactly is meant by intelligent agency, how is it (or will it be) accomplished, and who is out there trying to implement it.

## Operational definition

### General Concepts and Definitions

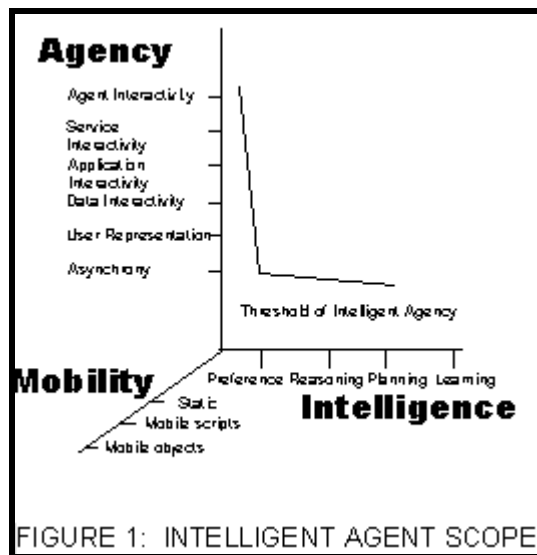
We can define an agent as *anyone or anything that acts as a representative for another party, for the express purpose of performing specific acts* that are seen to be beneficial to the represented party.

Norman<sup>1</sup> defines Intelligent agents as software entities that carry out some set of operations on behalf of a user or another program with some degree of independence or autonomy, and in doing so, employ some knowledge or representation of the user's goals or desires.

A software agent is a software program that performs tasks for its user within a computing environment. Technically speaking, most fourth generation application software could be defined as agents. Every day software performs hundreds of different tasks attempting to achieve some objectives for individuals. These tasks depend upon the agency attributes of the software. Software agents are different from other applications because of their autonomy, mobility and their ability to make independent decisions. When the additional element of agent intelligence is introduced, some form, even if elementary, of AI/ES methodology must be evaluated. Eventually, the agent must be able to understand and process external information given a set of attitudes, beliefs and knowledge of the principal.

### Characteristics of Intelligent agents

Three basic dimensions are used in this paper to describe intelligent agents. They are the levels of agency, intelligence and mobility of the agent. Figure 1 illustrates this concept



### Agency /Autonomy:

Agency is the degree of autonomy and authority vested in the agent, and can be measured by the nature of the interaction between the agent and other entities in the system. At a minimum, an agent must

<sup>1</sup> Norman, "Agents of Change"

run asynchronously. The degree of agency is enhanced if an agent represents a user in some way. A more advanced agent can interact with other entities such as data, applications, or services. A third level of advancement would entail an agent collaborating and negotiating with other agents.

Stein<sup>2</sup> has suggested that agency ( i.e. The property of being an agent) is determined by an observer's intentional stance; what a person views as an agent defines an agent. While this may well be a tenable philosophical position, it is not clear what benefit it provides. Rather, in terms of directing research efforts, particularly with the growing popularity of agents, it might be preferable to narrow the scope of the term to what we discussed.

### Intelligence /Capacity for Reasoning:

Intelligence is the degree of reasoning and learned behavior related to the agent's ability to accept the user's statement of goals and carry out the task delegated to it. At a minimum, there can be some statement of preferences, perhaps in the form of rules, with an inference engine or some other reasoning mechanism to act on these preferences.<sup>3</sup>

The ability to reason as an intelligent being is one of the key aspects of intelligence that distinguishes intelligent agents from other more mechanical agents. Belgrave<sup>4</sup> states that reasoning implies that "an agent can possess the ability to infer and extrapolate based on current knowledge and experiences - in a rational, reproducible way." Roseler and Hawkins<sup>5</sup> describe three types of reasoning scenarios:

- **Rule-based**, where agents use a set of user pre-conditions to evaluate conditions in the external environment,
- **Knowledge-based**, where agents are provided large sets of data about prior scenarios and resulting actions, from which they deduce their future moves, and
- **Artificial evolution-based**, where agents "spawn" newer generations of agents that carry higher and higher reasoning capabilities.

Higher levels of intelligence include a user model or some other form of understanding and reasoning about what a user wants done , and planning the means to achieve this goal. Further out on the intelligence scale are systems that learn and adapt to their environment, both in terms of the user's objectives, and in terms of the resources available to the agent. Such a system might, like a human assistant, discover new relationships, connections, or concepts independently from the human user, and exploit these in anticipating and satisfying user needs.

As shown in Figure 2., if an agent is classified in the domain that falls above and to the right of the threshold of intelligent agency in the Agency / Intelligence plane, then it is an intelligent agent under this definition . For example, expert systems which are not agents may fall below the threshold. Fixed function agents such as traditional systems management agents may fall to the left of the threshold. This graph may be used to approximate qualitative comparisons among intelligent agent- enhanced software offerings.

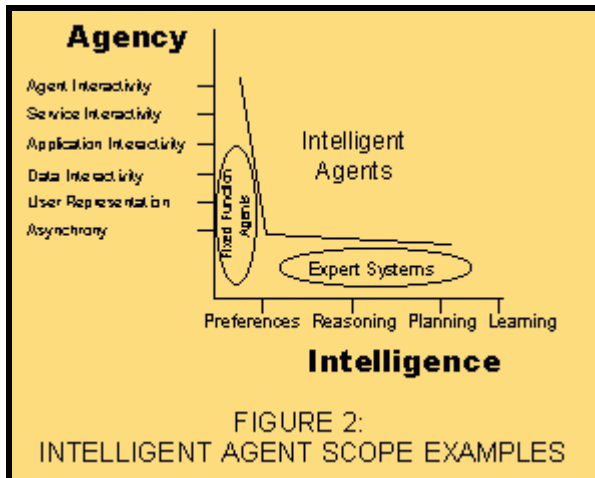
---

<sup>2</sup> Stein, Lynn, 1994 Private Communication, third discussion section. AAAI '94 Spring Symposium.

<sup>3</sup> Intelligent Agents on the Internet: Application, December 1995, Kathie Heilman.

<sup>4</sup> Belgrave, Marc. "The Unified Agent Architecture: A White Paper," [http://www.ee.mcgill.ca/~belmarc/uaa\\_paper.html](http://www.ee.mcgill.ca/~belmarc/uaa_paper.html), 1995.

<sup>5</sup> Roseler, Marina and Hawkins, Donald T. "Gent Agents: Software Servants For An Electronic Information World (and More!)," ONLINE, July, 1994, pp. 19 - 32.



### Mobility:

Networked agent applications add a third dimension to the picture<sup>6</sup>. Mobility is the degree to which agents themselves travel through the network. Some agents may be static, either residing on the client machine (to manage a user interface, for instance) or instantiated at the server. Agents may also be mobile in state, transported from machine to machine in the middle of execution, and carrying accumulated state data with them. Such agents may be viewed as mobile objects, which travel to agencies at which they can present their credentials and obtain access to services and data provided by these agencies.

Both Foner<sup>7</sup> and Wooldridge<sup>8</sup> indicate that intelligent software agents must have "control" over their "internal state and behavior." This would imply that an agent must have the access to a network and have the mobility to travel across it. Mobility brings a host of security, privacy and management challenges<sup>9</sup>. Initially applications will be built around static agents; mobility will appear gradually over time, as the infrastructure for agents matures.

In addition to these basic characteristics, intelligent agents should possess three other desirable characteristics: communication ability, capacity to cooperate and trustworthiness.

### Communication Ability:

Intelligent agents must, in the course of achieving their objectives, access information from third party sources about the current "state" of the external environment. This requires an ability to communicate with the repositories of this information that may be other agents or gatekeepers of information stores. Communication may be in the form of a singular request/survey with a simple and concise set of possible responses or it might be a complex communication with variable responses. On a rudimentary level, Belgrave provides the analogy of a simple passport used to communicate that the agent should be "spoken" with, given some filtering criteria. Communication, however, could be on a higher level involving true dialog, as opposed to the protocol-driven, statement exchanges noted above. Foner describes agent-to-agent communication in terms of a negotiation. Agents reveal intentions and objectives, eventually coming to some agreement or "contract." For Web-based intelligent agents, appropriate communication capability is critical for success.

### Capacity for Cooperation:

A natural extension of the communication attribute is cooperation. Intelligent agents must have a collaborative "spirit" to exist and succeed in what Foner calls "agent-oriented systems." The vision is for

<sup>6</sup> Virdhagriswaran S, Osisek D, and O' Connor P (1995). Standardizing Agent Technology, *ACM Standard View*

<sup>7</sup> Foner, Lenny. "What's an Agent, Anyway? A Sociological Case Study", <http://foner.www.media.mit.edu/people/foner/Julia/Julia.html>, 1994.

<sup>8</sup> Wooldridge, Michael. "Intelligent Agents: Theory and Practice," <http://www.doc.mmu.ac.uk:80/STAFF/mike/ker95/ker95-html.html>, 1995.

<sup>9</sup> Chess D, Grosz B, Harrison C, Levine D, Parris C, and Tsodik G (1995). Itinerant Agents for Mobile Computing, *IEEE Personal Communication Services Magazine*, 5, 34-49.

intelligent agents to work together to perform mutually-beneficial but complex tasks. Roseler and Hawkins note that this is where agency research crosses paths with research of "distributed artificial intelligence." An example would be where a trading agent may find another and exchange information or may merge into a multipurpose agent.

*Trustworthiness:*

Essential to the acceptance of agency is the sense of trust that the agent can accurately represent the user, its client. Intelligent agents must demonstrate what Wooldridge calls "veracity" and "benevolence." In other words, the user must be highly confident that its agents will act and report truthfully, and will act for the user's own good. Foner also notes that users must "balance" comfort with the risk of delegation against the criticality of the task, especially before agents have established a track record.

*Reactivity/Proactiveness<sup>10</sup>:*

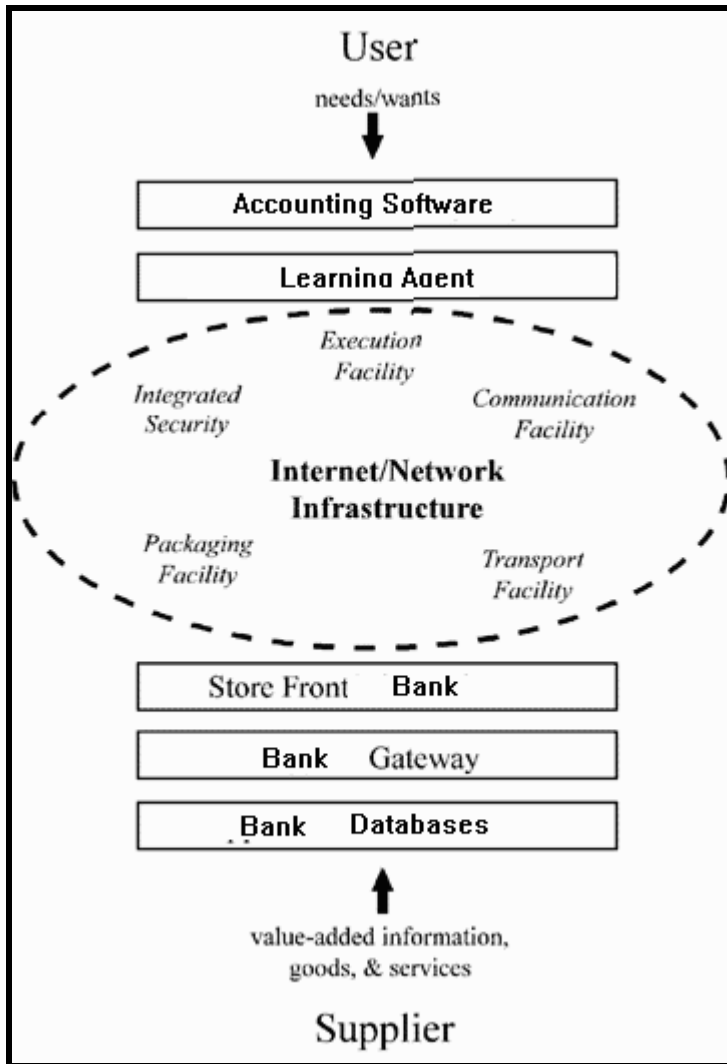
The potential for the agent to perceive its environment and respond in a timely fashion to changes that occur, and the ability of the agent to do more than just simply react to the environment and take the initiative to help the user by making suggestions and/or automating tasks the user normally would have to perform.

**Methodology - How Do Agents Work?**

The basic model of working with an intelligent agent is shown in Figure 3 below. A user with a task to complete uses a Graphical User Interface (GUI) or fills out a typical information profile. This information is passed onto an intelligent agent software application. The intelligent agent software application travels across the Internet or other network infrastructure, at its own discretion, to reach supplier gateways. It could be aided by storefronts or directory services. Finally, the agent is able to access the supplier information source in databases or other storage media, find the information requested by the user, complete any transactions necessary, and return a response back to the user.

---

<sup>10</sup> Woodridge, M. & Jennings, N. (1995), "Intelligent Agents: Theory and Practice," The Knowledge Engineering Review 10 (2), 115-152.



**Figure 3 Agent Methodology: an Accounting Example**

### Remote Programming

The key enabling technology for intelligent agents is remote programming (**RP**). The RP paradigm views computer-to-computer communication as enabling one computer not only to call procedures in another, but also to supply the procedures to be performed. Each message that the network transports consist of a procedure that the receiving computer is to perform and data that are its arguments. The procedure is one whose performance the sending computer began or continued, and that the receiving computer is to continue. This communication paradigm is shown in Figure 4 below.

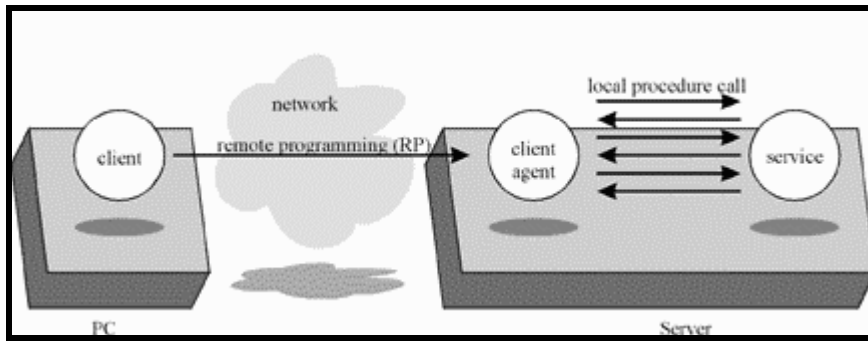


Figure 4 The RP Paradigm

Two computers agree in advance upon the instructions that are allowed in a procedure and the types of data that are allowed in its state. Their agreements constitute a language. The language includes instructions that let the procedure make decisions, examine and modify its state, and call procedures provided by the receiving computer. Such procedure calls are local rather than remote. The procedure and its state are termed a mobile agent to emphasize that they represent the sending computer even while they are in the receiving computer.

A user computer with work for a server to accomplish, sends the server an agent whose procedure makes the required requests of the server on site (for example, "delete") based upon its state (for example, "two months"). In order to delete all files at least two months old from a file server -- no matter how many -- requires just the message that transports the agent between computers. The agent, not the client computer, orchestrates the work, deciding "on-site" which files should be deleted.

The salient characteristic of remote programming is that a client computer and a server can interact without using the network once the network has transported an agent between them. Thus ongoing interaction does not require ongoing communication. The implications of this fact are far reaching for intelligent agents, as this is the underpinning of remote intelligent agent technology.

### Advantages

Remote programming has two important advantages over remote procedure calling. The tactical advantage of remote programming is performance. When a client computer has work for a server to do, rather than directing commands across a network, it sends an agent to the server and thereby directs the work locally rather than remotely. The network is called upon to carry fewer messages. The more work to be done, the more messages remote programming avoids.

The strategic advantage of remote programming is customization. Agents let manufacturers of client software extend the functionality offered by manufacturers of server software. Returning to the filing example, if the file server provides one procedure for listing a client's files and another for deleting a file by name, a client can add to that repertoire a procedure that deletes all files of a specified age. The new procedure, which takes the form of an agent, customizes the server for that client. The remote programming paradigm changes not only the division of labor among software manufacturers, but also the ease of installing the software they produce.

Unlike traditional standalone applications, communicating applications emerging for the Internet have components that must reside in servers. The server components of an RP-based application are dynamically installed by the application itself. Each is an agent. The advantage of remote programming is significant in an enterprise network but profound in a public network whose servers are owned and operated by the multitude of public service providers on the World Wide Web. Remote programming turns a public network into a platform.<sup>11</sup>

### **Mechanics of Agent Technology - Infrastructure Needed**

This section describes network infrastructure, in figure 3, in more detail. The infrastructure applied to intelligent agents includes

<sup>11</sup> White, James E. "Telescript Technology: The Foundation for the Electronic Marketplace," General Magic White Paper, 1994, pp. 3-6.

An Execution Facility  
A Communication Facility  
A Transport Facility  
A Packaging Facility  
Integrated Security

Each of these requirement is explained in detail in the following sections.

#### Execution Facility:

The execution facility, the most visible of the agents' components, is the agent's run-time environment. The execution facility consists of the hardware and software needed to execute agent programs in the agent environment. Hardware can include any kind of PC, client/server, or mainframe computers. Due to the proliferation of agent programming language compilers for multiple hardware platforms, consideration of hardware platform merits little debate other than for points of speed and compatibility with existing hardware and software investments within an organization.

The agent software programming language determines the range of functionality possible in an agent application. Hence, this is an area of intense study and research by academic and commercial institutions alike. Since intelligent agent applications differ greatly, and because the field of technology is still evolving, experts rarely agree on the appropriate programming language for developing intelligent agent applications. Generally, options fall into three categories: general purpose programming & scripting languages, general purpose mobile code languages, and mobile code languages written specifically for intelligent agent applications.

By far the agent programming language which has received the most attention to date is the Telescript language from General Magic.<sup>1</sup> Telescript appears to be the emerging standard for agent-based communication. It is an object-oriented, remote programming language, a platform that enables the creation of active, distributed network applications. Telescript enables programmers to write applications to allow agents to navigate through a variety of services and across diverse computing systems to perform tasks for users. In addition to providing the programming language for the execution environment, the Telescript environment also provides communication and security features. Its technology is generally considered to be superior to its competitors for intelligent agent applications, although the technology is not as mature as many of the general purpose programming languages and mobile code languages. Telescript is particularly well suited to electronic commerce applications.

IBM is in the process of modifying its successful REXX programming language, well-known for its ease of use and rapid-prototyping abilities, to take it "to the object realm -- and beyond -- destination: the emerging technology of agent-based computing." IBM hopes that the new Object-REXX language will become the standard for agent programming. Because Object-REXX has not been released yet, the jury is still out<sup>12</sup>. Some additional languages that facilitate agent applications are Agent-O, Placa, TCL/Tk, Safe-TCL, and Java. Additionally, programmers are still using traditional languages such as C.

#### Communication Facility:

Essentially, the communication facility deals with what the agent is communicating. Communication is the process by which information is exchanged in an agent transaction. Messages communicated have both content and context. Content refers to the data encoded in the message; context is that which effects the change from data to information.

The communication facility must support synchronous and asynchronous communication, and it must be able to communicate simultaneously with multiple dissimilar agents.

The Unified Agent Architecture, proposed by Marc Belgrave at McGill University in Montreal, Canada, defines the communication facility as a standard set of protocols which allows this communication to take place. To date no standard communication facility has emerged.

An agent communication language must be sufficiently expressive to communicate information of widely varying sorts. At the same time, the language must be reasonably compact, it must ensure that communication is possible without becoming excessively specialized. In order to achieve these goals, researchers in the ARPA knowledge sharing effort have defined the components of an agent communication language called ACL. ACL consists of three main parts, (1) its vocabulary, (2) KIF-knowledge interchange format, and (3) KQML- Knowledge Query and Manipulation Language.

<sup>12</sup>

URL: <http://www.w3.org/hypertext/WWW/MobileCode/>



Knowledge Interchange Format (KIF) is a computer-oriented language for the interchange of knowledge among disparate programs. It has declarative semantics, it is logically comprehensive, it provides for the representation of knowledge, it provides for the representation of non-monotonic reasoning rules, and it provides for the definition of objects, functions, and relations<sup>13</sup>.

Knowledge Query and Manipulation Language (KQML) is a language and protocol for exchanging information and knowledge. KQML is both a message format and a message handling protocol to support run time knowledge sharing among agents. KQML can be used as a language for an application program to interact with an intelligent system or for two or more intelligent systems to share knowledge in support of cooperative problem solving<sup>14</sup>.

#### Transport Facility:

Whereas the communication facility deals with what the agent is communicating, the transport facility deals with how the agent is communicating. The transport facility allows the movement of an agent from one execution environment to another in order to complete a task. It also allows for the distribution of un-initialized agents. The transport facility generally supports standard, established data transmission protocols such as HTTP, SMTP, and TCP/IP.

#### Packaging Facility:

The packaging facility provides a standard method of "wrapping" agents along with their associated information. Regardless of internal structure, all agents must encapsulate their state information, authentication information, goal information, agent capabilities, and method/plan information. To date no standard packaging facility has emerged<sup>15</sup>.

#### Integrated Security:

Security is a key issue that is very important if agent technology is to become wide-spread.

The agent infrastructure must provide an inherently secure method for determining an agent's owner and place of origin. In most applications, each agent possess a "passport" which will encode this information in a tamper-proof fashion.

Security methods used in agent infrastructure must also consider resource protection (not to abuse host computers or networks), non-repudiation (claiming a service was not received or not paid for when it actually was), privacy of information, and agent termination rules and procedures. Generally, each intelligent agent application applies its own security protocols to handle the specific requirements of the application.

#### Methods for Building Agents

We have broken up the methods for building agents into the following distinct categories:

- semi-autonomous agents with user programmable rules
- knowledge based agents with domain specific applications and user models
- learning agents that acquire knowledge from the user
- rule-based semi-autonomous agents that consist of user-programmable rules for carrying out user defined tasks

Beyond Mail from Beyond Mail Inc., is an example of a rule based agent where the user can specify rules to manage electronic mail. The main disadvantage of rule-based agents is that the user must program the agent.

Knowledge-based agents contain models both for a specific application domain and for the intended user. These agents assist the user at run-time with specific recommendations. The disadvantages of knowledge-based agents is their lack of customizability after deployment.

#### Key Players in Agent Technology Development

An overwhelming majority of the research in intelligent agent software is still being carried out in the university setting. MIT's Media Lab has been heading up an effort to apply the precepts of artificial

---

<sup>13</sup> URL: <http://www.cs.umbc.edu/kse/kif/>

<sup>14</sup> URL: <http://www.cs.umbc.edu/kqml>

<sup>15</sup> <http://www.ee.mcgill.ca/~belmarc/UUA.references.html>

intelligence (AI) theory to intelligent agents for the Web. The lab's Autonomous Agent Group<sup>16</sup> has become one of the leaders in the development of Web agent technology. Projects like Electronic Profiles and Let's Browse are some of the most promising being tested today.

The University of Washington's AI group<sup>17</sup> has produced an "Internet softbot" that can learn and adapt while accomplishing tasks for its user. The softbot can take requests from its user in a "higher-level" language and use its UNIX shell to carry out its chores.

Other institutions like the Carnegie Mellon, University of Maryland-Baltimore County, University of Michigan, and Stanford have established extensive research activities, either directly within their respective AI groups, or as off-shoots of them. Yoav Shoham from Stanford has been highlighted in particular for his proposed agent-oriented programming paradigm which addresses the more complex attributes of intelligent agency, such as reasoning and agent "mental state [16]" determination.

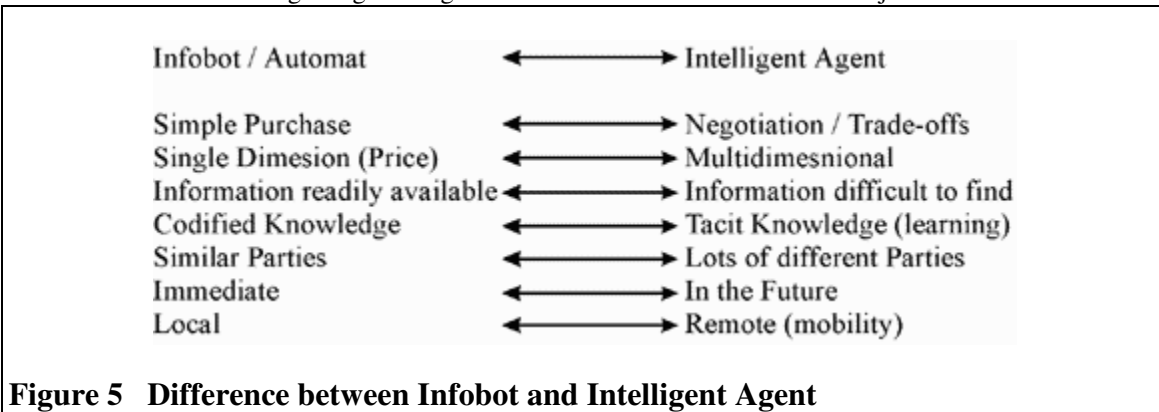
The commercialization of the Web has spurred on some companies to initiate participation in the development of intelligent agent technology. The most notable of these is Andersen Consulting and General Magic with its Telescript agent programming language. Armed with robust technology and backed by powerful industry players, General Magic looks to be the commercial entity to beat in the race to build the technical infrastructure for intelligent agents. Other players like IBM and Microsoft are also looking to enter into this emerging technology.

### The-state-of-the-art

Our state-of-the-art section consists of a framework for assessing the viability of intelligent agents and a well classified survey of existing applications of intelligent agents on the Internet.

### ***Viability of Intelligent Agents***

In order to identify potential for intelligent agents, one should first identify the factors which would make the use of agents attractive. It is important to differentiate between the features of an "Infobot" or "Automate" which does not contain intelligence (but nonetheless operates without human intervention), and those of an true intelligent agent. Figure 5 below illustrates several of the major differences.



### ***Negotiations or Auctions in Transactions***

An intelligent agent is well suited towards electronic commerce because of its functional intelligence. An intelligent agent can search for the best price or the best value for a transaction. An infobot, on the other hand, can only make a simple purchase. It does not have the capacity to bargain or shop-around. Currently, there is enormous potential to develop agents that can incorporate negotiations for electronic commerce.

### ***Problem Dimension***

Humans encounter problems everyday. Some of the problems have a multidimensional features. This means that the individual needs to trade-off between different factors having different weights in his or her utility function. Decision support systems have been designed to help humans deal with such problems. However an agent that can act like a neural network is required. It should encompass in its

<sup>16</sup> <http://agents.www.media.mit.edu/groups/agents/>

<sup>17</sup> <http://robotics.stanford.edu/groups/nobotics/home.html>

domain all relevant input functions and then, based on the human utility curve, evaluate the optimal outcome or search for it externally given the set of external constraints. Clearly, this is a complex task requiring a lot of computational power, and intelligent agent technology is well suited to this application. For example, while searching for a book, an agent should be able to take into account the user's preferences for types of authors, the classification of the book, the size (if relevant), and the time period in reference.

### ***Search Costs***

An important benefit from using an intelligent agent should be the savings in search costs. If information is difficult to find or takes a long time to collate, then the benefit from using an agent is higher. A straight-forward search through a single database cannot be considered intelligent. However, if multiple sources of information are searched using an adaptive search criteria, then this may be considered an intelligent search. For example if a user wanted to find out which American president had the most wives one would first search for American presidents then, based on this search, one would find out how many times each president had married. An intelligent agent would have the intelligence to combine these searches.

### ***Strategy vs. Informational Knowledge***

An intelligent agent can perform a task by using its own intelligence, its own tactics, to arrange the information requested by the user. This is gained through learning and feedback and is not explicitly stated by the principal. The infobot, on the other hand, can only work with information that has been codified.

### ***Communications***

The ability of the agent to communicate with other agents or sub-agents is an important property in its feasibility evaluation. An intelligent agent can interact with other different types of agents and exchange information, even if they do not speak the same language. An infobot can interact only with the same type of infobot using the same language. This advantage of the intelligent agent makes it particularly well suited to electronic commerce. With the wide array of applications and agents being developed to carry out electronic commerce, it is important that an agent be able to exchange information or merge with another agent.

### ***Timing***

The notion of an intelligent agent includes the concept of autonomy. The watcher class of agent described in the next section fulfills this criteria. It will monitor a stream of information until a pre-defined event occurs. This type of operation offers humans significant savings in time since one does not need to be continually monitoring for new events such as news flashes.

### ***Residency***

A characteristic of an intelligent agent is that it need not be resident on the client or server computer. It can in fact physically transport itself to different sites (if allowed through security features) via say "ftp" and then search for the relevant information. It can then come back to the client and report its findings.

## **Current Applications**

This section describes the current use of the term "intelligent agents" on the Internet. We have also classified the agent technology based on the agency relationship with the principal and the type of duties performed by the agent. Numerous applications are also described in this section.

### **Type 1: Learning Agents:**

A learning agent observes user actions in the background, finds repetitive patterns and automates them upon approval. The learning agent paradigm uses the metaphor of a personal assistant. Learning agents are, in particular, applicable when

1. the application domain contains significant repetitive behavior
2. the repetitive behavior differs across the user spectrum

The advantages of learning agents include:

- automatic customization of software to the user, in contrast to the current practice of the user adapting to the software
- continuous adaptation of the software to changing user work patterns
- better user acceptance since the user dictates what the software learns

- lower development costs by achieving customization and adaptability through machine learning.

Learning agents acquire knowledge from the user through the following techniques:

- knowledge acquisition methods that directly query the user to obtain information as needed
- dialog-based learning methods that identify procedures by watching the user perform tasks manually
- memory-based learning where the agent finds the closest match against a memorized set of situation-action pairs
- feedback based learning-users can communicate with the agent by providing positive and negative feedback about relevant topics and through providing examples of desired data
- hybrid neural network and knowledge base system architectures where the neural networks perform knowledge acquisition, and knowledge -based techniques perform inference and knowledge maintenance

Knowledge acquisition methods are used to query the user to supply salient examples and information for use in training the agent. Typically this has been the standard practice when using a production system in building agents.

Firefly was originally called the HOMR music recommendation service and was developed by the MIT Media Lab. Firefly is your own personal software agent capable of communicating with other users and recommending music and movies that it knows you'll enjoy.

Firefly automates the word-of-mouth process, learning about you and your opinions, and leveraging that information to best serve your needs. Firefly uses the tastes, opinions, preferences and idiosyncrasies of those most similar to you (your "nearest neighbors") in order to suggest new music that you might like too. The more you train your agent, the more useful and accurate it gets. The more other people use the system, the smarter the Firefly community becomes.

The system is also similar to the Similarities agent which also recommends music based on the users choice.

### *WebMate*<sup>18</sup>

WebMate is a personal digital assistant that was developed by Carnegie Mellon University to assist users in finding information on the Web. WebMate accompanies users as they browse the web and actively provides recommendations regarding desired documents. WebMate has the following features: it provides URL recommendations, continuously updates user profiles, responds to feedback from the user, and compiles a daily personal newspaper with links to documents of interest to the user.

WebMate is composed of a stand-alone proxy that can monitor user's actions to provide information for learning and search refinement, and an applet controller that interacts with the user. WebMate learns the user profile incrementally and continuously and uses it statistically to determine relevant documents. WebMate utilizes TF-IDF learning algorithms, which represents each document as a vector space so that documents with similar content have similar vectors. The values of the vector elements for a document are calculated as a combination of statistics term frequency (w,d), where w is a word count in a document d.

### *Yenta*<sup>19</sup>

Yenta is an intelligent agent that generates clusters of users who are interested in similar topics. The clusters serve the purpose of introducing users to one another so that they can send messages to a particular user or to all the members of the cluster. Another benefit of Yenta is the ability to find people (within or out of an organization) that work on similar projects, or use similar tools. Yenta can handle textual data such as email messages, news groups articles, and the content of a user's file system. The fundamental assumption behind this agent is that if users have several documents which are similar to each other then it is assumed the users share a common interest. Since Yenta deals with textual information,

---

<sup>18</sup> Chen, Liren and Sycara, Katia, "WebMate: A Personal Agent for Browsing and Searching." Proceedings of the 2<sup>nd</sup> International Conference on Autonomous Agents and Multi-Agent Systems, Minneapolis, MN, May 10-13, 1998.

<sup>19</sup> <http://foner.www.media.mit.edu/people/foner/yenta-brief.html>

similarity is measured using a keyword vector text comparison metric. Agents then group themselves into clusters which reflect their users' interests. These clusters are then used to make introductions or allow users to send messages to others who share their interests.

### *Open Sesame!*

Open Sesame!,<sup>20</sup> which was recently acquired by Allaire, is a learning agent that observes your activities and learns which tasks you repeat again and again. Open Sesame! even automates crucial maintenance tasks that you can easily forget, like rebuilding your desktop. Whether you're a novice or power user, Open Sesame! makes your work simpler and more efficient. The Open Sesame! Technology originated from Charles River Analytics, an early 80's pioneer in neural expert systems and learning agent technology. Open Sesame is based on technology developed as a result of several years of research and development focused on learning agents, desktop personalization and personalized services on the Web. Powerful technology that delivers unmatched profiling, personalization and business intelligence capabilities, Open Sesame! Is a unique profiling and personalization server that offers the following features: learning and recommendation engine technology, implicit and explicit user profiling, individual and group profiling, real-time learning and dynamic profiling, and customer profile data-mart.

A good example of Open Sesame! in action is in the use of documents. For example, each time you open an invoice template inside FileMaker Pro you also open another document which is your database of clients.

Open Sesame! will notice the routine and offer to open the database of clients each time you open the invoice template. The program can actually open or close entire sets of folders, documents and applications with a single click.

### *Let's Browse*<sup>21</sup>

Let's Browse, an experimental agent built by the MIT Media Lab. This agent is geared to assist groups of people in browsing, by suggesting new material likely to be of common interest. It is built as an extension to the web browsing agent Letizia, which acts as an advance scout. The goal was to create an automatic sensor that will trigger the agent as soon as the user approaches the screen. The screen would then continuously show selection of Web pages that the agent determined might be of common interest to the participants, together with explanations of its choices.

### **Type 2: Viewer Agents:**

Viewer agents are constantly on the lookout for information that matches a users profile in a defined area. The user typically fills out a profile of what he or she is looking for. The agent then operates autonomously, makes its own decisions, and notifies the principal when it finds relevant information. Notification can be via e-mail or a news flash.

### *FishWrap*<sup>22</sup>

An example of an intelligent newspaper is MIT's Fishwrap newspaper. It collects stories from a number of different sources and then generates a personalized version of the newspaper automatically, based on the user preferences. As shown in figure 6, the reader starts with his or her table of contents, then focuses on a news topic and then the audio or video article.

The paper allows readers to rate stories and positions these stories in the paper according to their popularity. At this time, the level of customization is rudimentary; users can specify their home town, professional interests, and community affiliations.

The Fishwrap design readily accepts traditional news wire stories and direct contributions from the community. Social and cultural calendars are also collected and integrated into the system. All items coming into the system are analyzed for geographic or topical relevancy. Fishwrap employs an automated news model called "Glue" which is composed of interrelated components that assemble an individual's news: user profile, knowledge representation, news suppliers, authentication, self and community

---

<sup>20</sup> <http://www.opensesame.com/>

<sup>21</sup> <http://lieber.www.media.mit.edu/people/lieber/Lieberary/Lets-Browse/Lets-Browse-Intro.html>

<sup>22</sup> <http://fishwrap-docs.www.media.mit.edu/docs/>

organization, customization, and presentation. It also classifies stories into categories such as "automotive", "fiction", or "artificial intelligence".

The project lacks the ability to automatically modify the structure of the papers presentation to reflect trends in the reading habits of the participants. So currently, the agent does not learn from experience.

Fishwrap is a long way away from Negroponte's<sup>23</sup> vision of an electronic newspaper -- with a dial to change the political bias of the stories it shows. However, Fishwrap and other intelligent newspapers are moving in the right direction.

### *Mercury Center's News Hound*<sup>24</sup>

The best example of a newspaper which notifies its readers when it finds a story of interest is Mercury Center's NewsHound.

News Hound automatically searches for articles from a wide range of newspapers and wire services, as well as classified ads from the San Jose Mercury News. As soon as it finds relevant documents as specified in your profile, it sends them directly to your electronic mailbox.

You create a NewsHound profile by mailing a simple form with words and phrases that could appear in the information you want. In a few minutes, a confirmation message will appear in your e-mail box.

Almost immediately, the NewsHound will begin searching the articles and ads received during the current day and sending any that match your profile. This feedback helps you gauge the effectiveness of the profile. After viewing these articles, you can adjust your profile by adding or eliminating words or phrases, or by changing its "Selectivity."

Once created, a NewsHound profile searches incoming articles and advertisements every hour on the hour throughout the day and sends the most relevant ones to your e-mail box. Users do not have to spend time searching for this information because NewsHound does the work.

This service could be improved by building more intelligence into the search process and including a feedback loop so that the agent could learn what stories are really of interest to the subscriber. Without such a feedback loop, the user gets bombarded with information and eventually ignores the agent altogether.

### *Other Examples*

There are numerous other examples of viewer agents that we discovered. Some of them are: Personal Journal [20] from Dow Jones and Co., PCN from Point Cast Network and Ziff Davis Personal View.

An interesting example is that of Job Center where users post their resume on the Web and receive matching ads for job positions via e-mail. The search engine automatically matches the seekers skills to what the employers are looking for.

### **Type 3: Shopping Agents:**

Shopping agents are capable of doing comparison shopping and finding the best price for an item. However, what is needed is also value shopping wherein an agent can analyze multidimensional features and create a value for each item. Retailers can protect themselves from this sort of competition by giving their products unique names (as is done by Virtual Vineyards [30] or by blocking access from search agents (as is happening to the Bargain Finder agent) or by offering unique items.

### *Bargain Finder*<sup>25</sup>

Currently, the best known shopping agent on the Internet is the Bargain Finder agent from Andersen Consulting. This agent does price comparison shopping for compact disks (CDs). The Bargain Finder exhibits some features of an intelligent agent in that a number of different parties are involved, operation is remote, and information is difficult to find (try for yourself to replicate the action of the Bargain Finder and see how long it takes). The principal factors which weighs against classifying the Bargain Finder as a true intelligent agent is that it only handles a single-dimension problem, namely comparing prices for a unique (standard) product. It does not have to negotiate or make any tradeoffs, so the "intelligence" required is minimal. The clever part is being able to talk to different CD vendors.

---

<sup>23</sup> Negroponte, Nicholas. "Being Digital", Alfred A. Knopf Inc., New York. 1995

<sup>24</sup> <http://www.sjmercury.com/hound.html>

<sup>25</sup> <http://bf.cstar.ac.com/bf/>

### *Pocket Bargain Finder*<sup>26</sup>

Pocket Bargain Finder is an extension of the above agent and is designed to bridge the gap between electronic and traditional commerce. With Pocket Bargain Finder, consumers can show in a physical retail store, find an item of interest, scan its barcode, and search for a lower price among a set of online retailers. The designers placed three main requirements for the Pocket Bargain Finder system: (1) portability of the device, (2) compactness and comfortability, and (3) simplicity of the input device and easy readability of the output. The hardware of the device has three main requirements, a barcode scanner, a wireless communication device, and a small computing device for converting barcodes into a comparable web format. The software includes a web server and a micro-browser that has the capabilities to process HTML messages and Common Gateway Interface (CGI) scripts stored on the web server. The scripts first read the data sent by the browser, translate the data into the appropriate ISBN number (this is a printed material product), submit ISBN to multiple book stores and collect the information including the price, shipping time, and shipping cost. The results are then stored as an HTML page, and the user is notified.

### *KasBah*<sup>27</sup>

KasBah is a Web-based system where users create autonomous agents that buy and sell goods on their behalf. KasBah is a multiagent system, where agents interact with the marketplace on behalf of the user. A selling agent is autonomous in that, by creating the agent, and setting several parameters to guide it as it tries to sell the specified item. The items controlled by the user are, the desired date by which to sell the item, the desired price, and the lowest possible price. The user also has some control over the agent's negotiation strategy; the user has three negotiation strategies to choose from, linear, quadratic, and cubic. The challenge for this type of agent will surface once the marketplace is composed of many similar agents that all have similar negotiation algorithms. One proposed solution by Sam Park of Rutgers University was to create an adaptive strategy selection in order to optimize the negotiation process with other agents.

### *ShopBot*<sup>28</sup>

ShopBot is a fully implemented, domain independent comparison-shopping agent. Given the home pages of several online stores, ShopBot autonomously learns how to shop at these vendors' sites. ShopBot operates in two different stages, the learning phase and the comparison-shopping phase. In the learning phase, an offline learner creates a vendor description for each site. The comparison-shopping phase uses the learned vendor descriptions to shop at each site, finds the best price for a specific product desired by the user, and displays a summary of the results to the user. ShopBot still requires some minimal human intervention, including the identification of relevant merchants (their URL) in a particular domain.

### *AuctionBot*<sup>29</sup>

The agents above facilitate the connection of potential buyers and sellers (Business to Consumers), B to C is usually done in a take it or leave it type of negotiation, where businesses post their prices and consumers can either accept or decline the offer. However, growing trends in today's e-commerce are B to C auctions and C to C auctions. Auctions represent a more general approach to price determination, where involved parties have a price range that is negotiable. AuctionBot users can create new auctions to buy and sell products. AuctionBot supports both human and software agent interactions. The three core auction activities are Receive bid, Supply intermediate information (price quote), and Clear. To support these activities, the following message types are supported: bid, bid withdrawal, bid admittance, bid rejection, price quote, and transaction notification.

### **Type 4: Search Agents:**

Information retrieval agents are capable of searching for information in an intelligent fashion. The most obvious example is an Internet search agent which can conduct complex searches by interpreting the search criteria defined by a user. For example, information retrieval agents are capable of searching for information in an intelligent fashion. This section includes products and technologies which could be linked to the Internet in the future. These examples of information retrieval applications are:

---

<sup>26</sup> <http://www.ac.com/services/cstar/Publications/PocketBargainFinder-HUC99.pdf>

<sup>27</sup> <http://ecommerce.media.mit.edu/Kasbah/>

<sup>28</sup> Doorenbos et al "A Scalable Comparison-Shopping Agent for the WWW"

<sup>29</sup> <http://auction.eecs.umich/index.html>

### *Oracle ConText*

Oracle ConText is a natural language processing technology that can be integrated with any application to create sophisticated systems capable of analyzing and understanding English text. For example, Oracle ConText integrates with Oracle TextServer to provide text filtering for more precise and meaningful text tracking, as well as text reduction for speed-reading and summary viewing.

### *Strouds Consummate Winsock and Peregrine's Page*

Strouds Consummate Winsock or Peregrine's Page in Carlsbad Calif., will help early Windows 95 users with configuration and installation problems. By logging into these sites, users can access the systems management vendor's Information Retrieval Expert system, which contains a knowledge base of Win95 information. The intelligent query agent can search problem -relevant information and solutions quickly.

### *MetaCrawler*<sup>30</sup>

Because of the explosive growth of information on the Web, many search engines have been developed. However, no single search engine indexes all the available information on the Web. Meta search engines deal with the above problem by providing a single user interface to multiple underlying search engines. They accept user's query and execute a parallel search, sending the query to different engines and collecting, ranking, and displaying the search results. MetaCrawler allows users to search for words as a phrase, search for all the words, and to search for any word in the phrase. Additionally, MetaCrawler recommends the user to search according to some more common phrases in order to achieve better results.

### *Amalthea*<sup>31</sup>

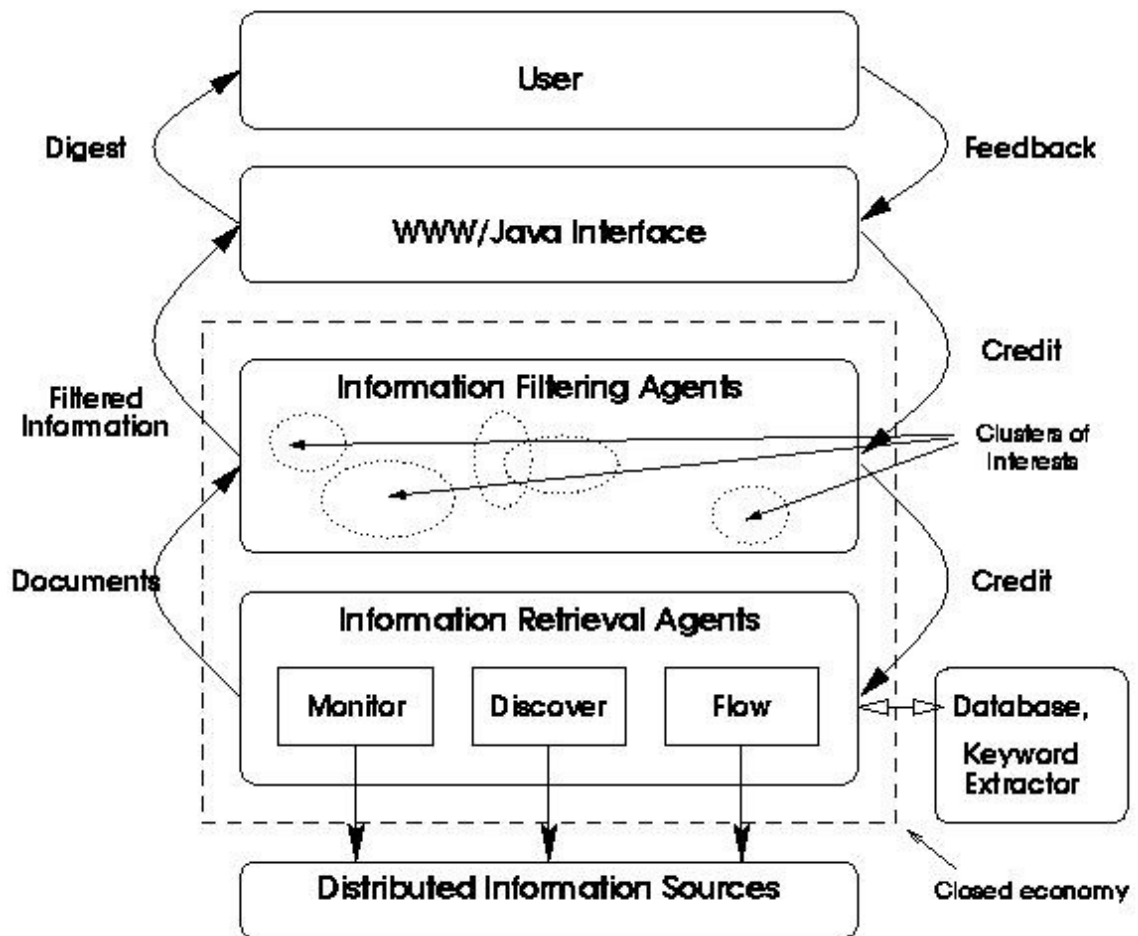
Amalthea is an intelligent agent that applies for both personalized information agents and information discovery agents' domains. Amalthea is a personalized system that proactively tries to discover information from various information sources including: the WWW, information from the MIT media laboratory Newsfeed, and other frequently changing information resources. Amalthea launches multiple agents that utilize existing indexing engines to perform a meta- search in order to discover relevant information for the user. The system then analyzes the retrieved documents using weighted keyword vector techniques in order to select documents that fit best to the user's preference. Amalthea was implemented by creating an artificial eco-system of evolving agents that cooperate and compete in a limited resource environment. Amalthea employs two main species of agents: (1) information discovery agents, and (2) information retrieval agents. The information-filtering agent is responsible for the personalization of the system, and the information discovery agent is responsible for information resources handling.

---

<sup>30</sup> <http://www.metacrawler.com>

<sup>31</sup> <http://amalathaea.www.media.mit.edu/projects/amalathaea/>





### ***Type 5: Remembrance and Helper Agents:***

People have notoriously bad memories. We forget where we left our keys, people's names, and simple directions. Computers, on the other hand, almost never forget the information they store. An agent that can augment human memory is a remembrance agent—that helps us remember.

#### ***Remembrance Agent at MIT<sup>32</sup>***

This agent from the MIT Labs, can log everything a users does, all the information that passes through a user's hands and can help a user remember some information based on the content or context (episodic memory) of the situation. Although it is still far from remembering everything that a human can and organizing it in different ways, this is a step in the right direction.

Remembrance agents are proactive agents that do not require the user to have a query in mind, or even to know what information is relevant to the particular situation. Remembrance agents are designed to provide the user with additional information regarding the nature of the alarm, such as news articles related to some stock price, whose abnormal price fluctuations would have triggered the alarm.

#### ***StreetWise<sup>33</sup>***

This agent is a location-based agent assistant, which is designed to assist users in the physical world. The advent technology has created a new paradigm of communication, the wireless paradigm. Electronic agents can be employed in the physical world using wireless devices such as palm computers and cell phones. This agent provides users personalized information dependent on their geographic location. The designers approach location based computing by structuring the agent communication into

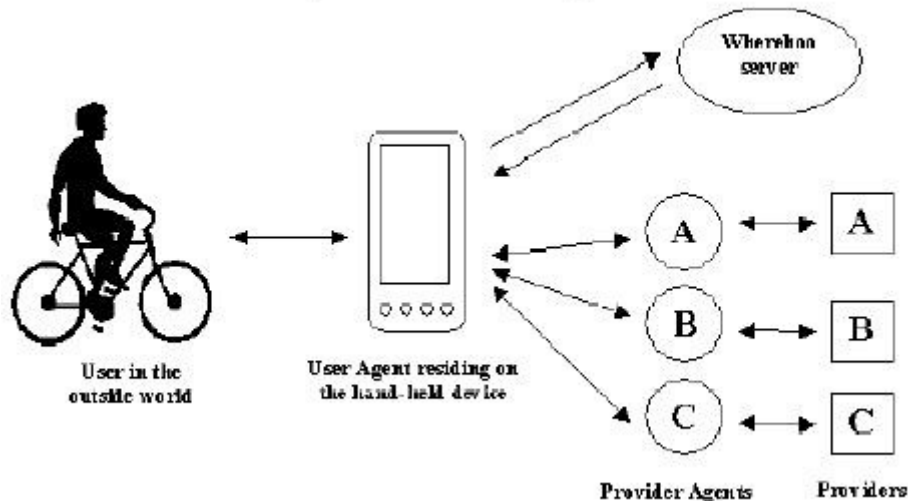
<sup>32</sup> <http://rhodes.www.media.mit.edu/people/rhodes/Remembrance-distribution/>

<sup>33</sup> <http://agents.www.media.mit.edu/groups/agents/projects/impulse/streetwise/>

four components: the User Agent, the Wherehoo Server, Provider Agents, and Providers. (1) The User Agent represents the user's interests and interacts with Providers on the user's behalf. (2) The Wherehoo Server is a search server using location as its primary criterion. (3) Providers are physical resources such as businesses, services, attraction and events, or points of interest to the user's physical search domain. (4) Provider Agents are controlled by Providers and interact with User Agents.

Thus, the Wherehoo server allows a User Agent to ask a question such as "Is there a movie theater next to my location?" So the user can issue a query consisting of location search radius, and keywords to the Wherehoo server (eg. "<GPS location>, 1 km, movie theater") the server returns URLs for physically nearby Providers whose descriptions match the keywords. The User Agent retrieves the documents at those URLs and seek matches with standard XML elements.

#### Agent Interaction Diagram



Based on the premise that people are now expensive, and companies cannot afford to deploy support staff wherever they are needed, we envision an intelligent agent that will continually monitor the health and well-being of every device and application on the network. For example, the agent could figure out that the reason a user was unable to receive mail was because the mailbox was full. The agent could then diagnose and fix the problem automatically.

#### Homework Helper<sup>34</sup>

Homework Helper is the first on-line library and research tool for the entire family. Homework Helper contains more than 100 full-text newspapers, over 400 full-text magazines, several news wires, multiple encyclopedias, hundreds of maps, thousands of photographs, and major works of literature. These resources make Homework Helper a powerful learning resource for children and adults.

Homework Helper has an easy-to-use graphical interface and allows users to ask questions in plain English, and instantly retrieves full-text documents from its wide range of resources. Users can ask the Homework Helper to retrieve only materials appropriate to their age or reading level.

#### Track<sup>35</sup>

FTP's new Intelligent Agent Business Unit is developing an agent technology, Track, to allow network managers to upgrade and distribute software from a single point on the network, as well as facilitate Web searches. Track agents are event-driven, self-modifying software programs that are designed to perform tasks such as network reporting, remote execution, and data gathering.

#### Type 6, Multi-Agent Systems

Lately there is an increased interest in multi-agent systems research and development. Multi-agent systems are able to operate complex tasks that may involve multiple stages, and distributed controls in multiple domains. The term multi-agent system usually refers to all types of systems that are composed of multiple (semi) autonomous components. In multi-agent systems, each agent has incomplete

<sup>34</sup> <http://www.carl.org:80/news/homework.html>

<sup>35</sup> <http://www.ftp.com>.

information, or capabilities for solving the problem, there is no global system control, and the computation is often asynchronous. Some of the challenges in creating and operating multi-agent systems are: decompose and allocate the task, synthesize the results, facilitate communication and interaction among agents, solve conflicts, balance computation, and maintain security.

Many of the recent developments in agent technology involve multi-agent systems. Some examples include the following applications:

*Maxims*<sup>36</sup> - collaborative interface agents for electronic mail filtering which learns, prioritize, delete, forward, sort, and archive mail messages.

*Warren Financial Portfolio Management System* - a multi-agent system that integrates information finding and filtering in the context of supporting users in managing their financial portfolios.

*Straum*<sup>37</sup> - which introduces the idea of an ecology of distributed agents as a paradigm for building distributed software is a system that represents peoples' presence by managing and disseminating data about users.

*Retsina*<sup>38</sup> - multi-agent infrastructure consists of a system of reusable agent types including interface agents, task agents and information agents.

#### *Type 7, Business Side Agents*

Intelligent agents are often used to ease the burden of negotiation, and customer service on business in addition to streamlining business processes. Businesses in the new economy may change their strategy from a uniform take it or leave it pricing method, to a dynamic pricing method, including price negotiation, and demand sensitive pricing. This type of pricing may impose an additional overhead cost for the business, and therefore, is highly dependent on the ability to automate it.

#### *IntelliServe*<sup>39</sup>

Internet commerce is highly competitive; customer service is expected to become one of the main parameters that differentiate one merchant from another. Shoppers expect immediate response to their enquiries in addition to other customer service components. In order to retain and expand the customer base, businesses have to provide timely customer service. IntelliServe system automatically answers most customers' emails without any human intervention. Response to customers is immediate regardless of the number of incoming requests. When the system receives customers' email, it is classified under one of the following message types: (1) Unknown message-forwarded to a customer service representative. (2) Message can be answered with standard response and does not require additional computer processing. (3) Message requires some computer processing, but can be answered. (4) Message requires some computer processing and does not contain all the information needed to process and respond. In this case, the system engages in conversation with the customer to collect the missing data.

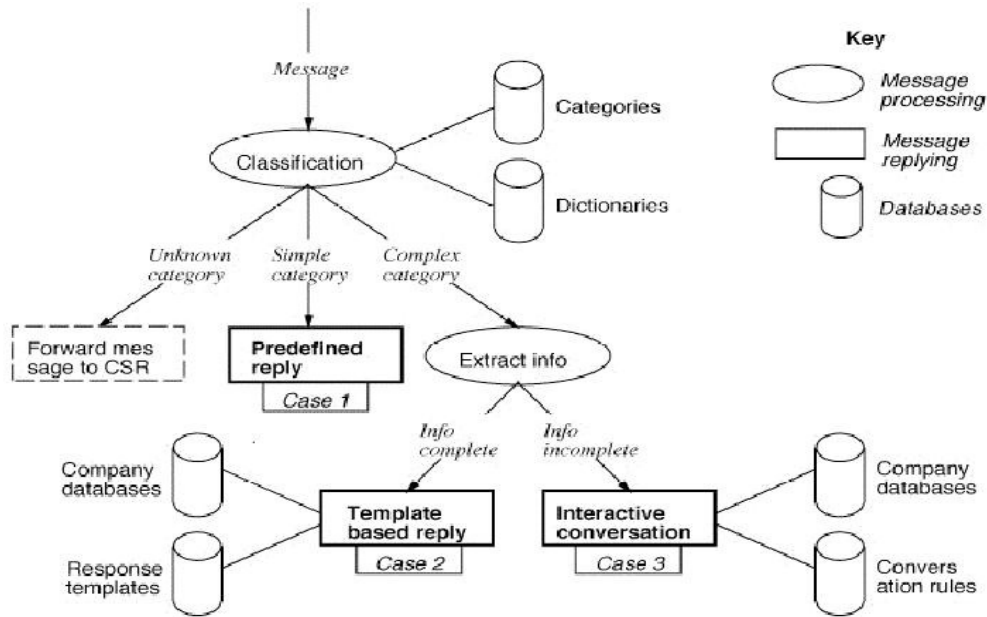
---

<sup>36</sup> <http://agents.www.media.edu>

<sup>37</sup> <http://nelson.www.media.mit.edu/people/nelson/research/masters-thesis/>

<sup>38</sup> <http://www.cs.cmu.edu/~softagents/restina.html>

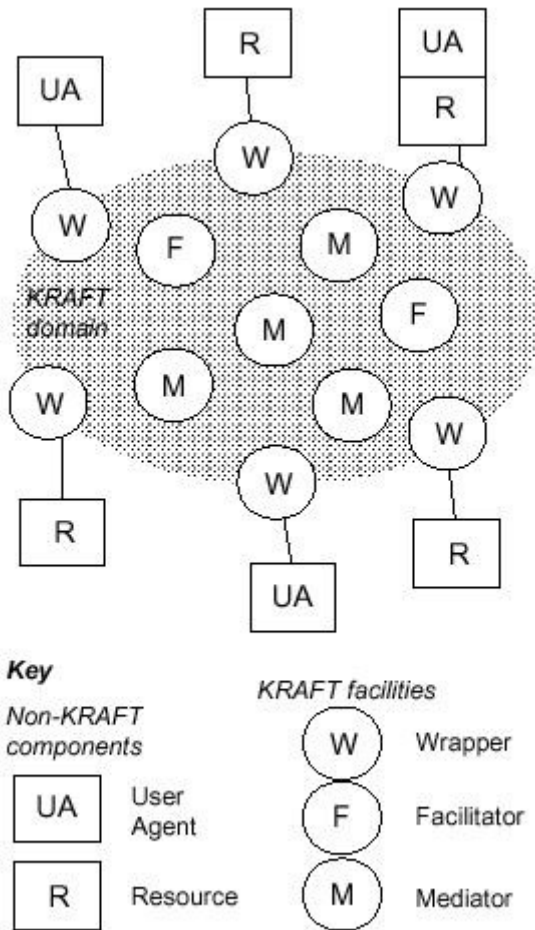
<sup>39</sup> <http://www.cs.umbc.edu/aiec/papers/fox.pdf>



#### *KRAFT*<sup>40</sup>

The Kraft (Knowledge Reuse and Fusion/Transformation) has an architecture that is suitable to support virtual organizations in which members exchange information in the form of constraints expressed against an object data model. KRAFT was created to support configuration design applications among multiple partner organizations with heterogeneous knowledge processing components are realized as software agents. KRAFT has three kinds of user agents: (1) Wrappers-agents that act as proxies for external knowledge sources (such as databases and knowledge systems). (2) Mediators- are internal knowledge processing agents of the system where every mediator adds value to the process. (3) Facilitators- are the matchmaker agents, that allow agents to become acquainted and communicate.

<sup>40</sup> <http://www.csd.abdn.ac.uk/~apreece/publs/aiec99.pdf>



### *Towards Accounting and Finance Agents*

While Finance and Accounting Agents are still not prevalent on the Internet eventually some very valuable tools may appear.

#### ***Personal Financial Software Agents***

It is likely, that several personal financial help agents will evolve from the current set of personal financial software. For example, currently, the purchaser of TurboTax can access the Quicken Financial Network (its holding company) and obtain, a shopping area for Intuit products, get access to the Internet for a price, some low-level tax advice, and stock prices. A natural development was the automatic connectivity of the Quicken software, to some online banking systems, connection to check payment systems (such as CheckFree) and automatic download of checkbook, savings, credit card, and other status information. Now we can expect some more intelligent forms of financial agenting from this type of vendors. for example:

- an agent to find the cheapest bank loan available on the Internet. (a type 2 viewer agent)
- an agent constantly monitoring if there are more economical forms of financing the principal's outstanding debts (for example credit card loans)
- an agent presenting the principal's financial information and applying for loans to a bank. As well as the corresponding bank agent offering the loan and negotiating the terms.

- an agent monitoring the user's relationships with all (or as many as possible) of its suppliers (as mortgages, credit cards, bookstores, etc. ..) for late payment, credit terms, availability of items, special offers.
- an agent balancing the principal's portfolio according to say, a Markowitz model, taking in consideration the principal's utility function, current market conditions, and executing buy and sell orders within a set of allowable parameters.
- a set of resident agents at related entities that feed information to the principal's balance sheet and income statement. For example, the corporate savings plan site, the retirement fund site, banks for balances, employer's for salary information, etc. This set of related agents would monitor individual financial health and provide advice upon request.

Linking the Quicken Financial Service network, to some of the above proposed agents, clearly would raise a series of issues that are more of an obstacle than the actual technology for their implementation which is mostly available. For example, the bank loan agent must be **trustworthy** as Quicken could favor its related entities. It must have **capacity for cooperation** as after the identification of a loan opportunity, the application process may need interaction, and explanation.

The personal financial agent must have resident permission, or interface with another entity's tickler file to allow the controlled disclosure of information, on a regular, or exceptional base. These features require **communication ability** from a simple firing an e-mail message, to extensive secure and protocolled dialogs from both parties.

In many instances, it is noticeable that corresponding agents may be possible (principal's agent looking for a loan, bank's agent looking for prime candidates for loans). This principle of **reciprocity** may assume many levels, from a simple permission to reside, through an adaptable hand-shake to pass information, to a fully corresponding reciprocal agent providing the polar opponent service.

### ***Learning Financial Agents***

Attached to the personal financial tools described above a higher level of intelligence may be prescribed. First we can classify the above tools into a) financial reporting agents, 2) financial action agents, and 3) analytic advisory agents.

An agent could be devised that uses the financial reporting agent to establish the baseline of principal wealth, and over time analyses the change of principal wealth, the result of its financial actions, developing a profile for learning and advising. For example the agent would notice that most expenses are paid at the end-of-the month and there are available balances at the middle of the month. These would be automatically moved to interest bearing accounts.

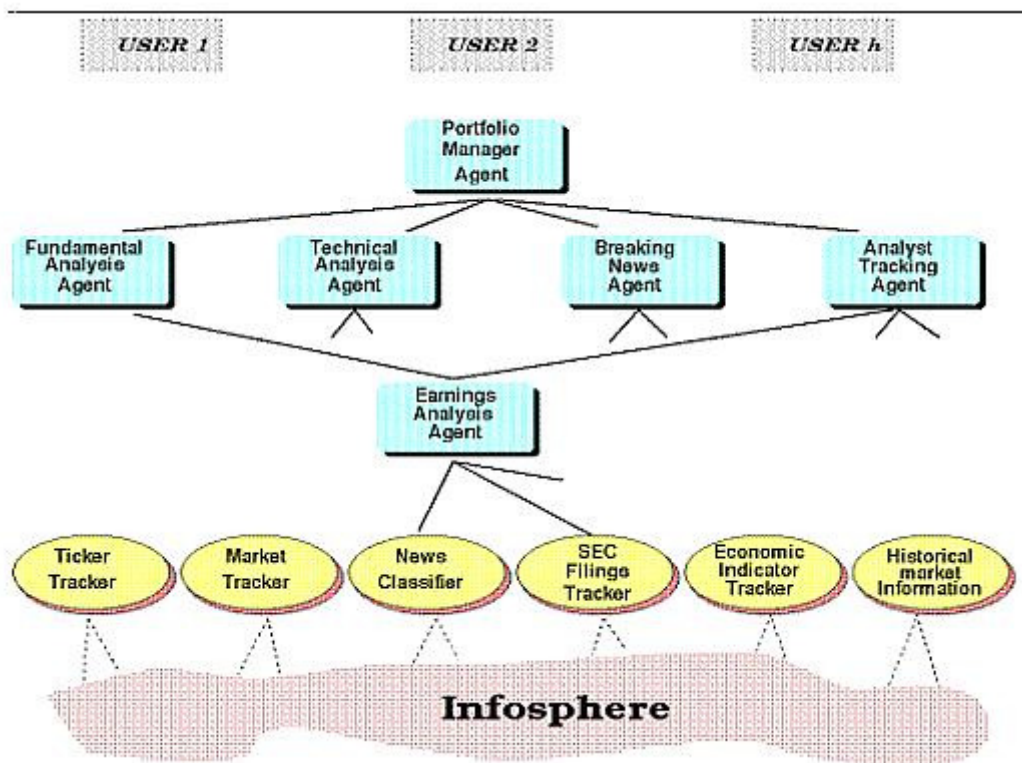
These are also analytic advisory services, but now with impounded learning capabilities.

*Warren*<sup>41</sup>

The Warren system is a multi-agent architecture that meets the challenge of financial portfolio management. Since portfolio management requires coordination of several components across distributed network and information sources a multi-agent approach is the most appropriate for this task. The Warren system builds on current investment practices to deploy a number of different, semi-autonomous software agents. These heterogeneous agents acquire information from and monitor changes to stock reporting databases, interpret stock information, predict the near future of an investment, and track and filter relevant financial news articles. The Warren system is designed to monitor the ongoing portfolio management process, and thus to function under conditions of extreme uncertainty.

---

<sup>41</sup> <http://www.cs.cmu.edu/~softagents/warren.html>



#### *FRAANK<sup>42</sup>*

FRAANK-Financial Reporting and Auditing Agent with Net Knowledge, assists auditors in gathering relevant data for the auditing process. This type of agent is only the beginning of the transition that will occur in automating data retrieval that is value relevant to the auditing process. The integration of XML (Extensive Markup Language) is likely to facilitate accurate retrieval of data, which can be analyzed and used in the auditing process. The FRAANK project is a multi-agent project that includes three sub-agents: the Edgar agent, the ticker agent and a stock quote agent. The Edgar agent is an intelligent agent that searches the SEC Edgar database for a certain desired quantity such as the current cash balance of a user specified corporation. The FRAANK agent is expected to be further developed to include a detailed automated financial statement extraction and analysis, automated acquisition of new synonyms of accounting terms in the financial statement, and interaction with additional sources of information. The value created by the FRAANK can assist accountants in the process of streamlining the auditing process, and facilitate a more timely audit report.

#### *Information Search Agents*

A set of financial information search agents can monitor and gather links or databases of items such as:

- financial statements on the Web
- stock market indices on the Web
- changes on corporate status on the Web
- financial announcements in news services about a company
- foreign exchange data aiming for valuation of assets or arbitrage opportunities

<sup>42</sup> <http://lark.cc.ukans.edu/cgiwrap/srivasta/agentwork.cgi>

- commodity data associated with published information on commodity production as well as market published news.

### ***Fraud Monitoring Agents***

Agents do not need to reside in the Internet, or navigate on the Internet. They may be placed incorporate systems or in the huge emerging Intranets. For example a company may allow for hosted Java scripts in key points of its information flow detecting known fraud patterns, irregular activities, or variations to regular work flow rules.

For example a set of simple tools could be developed to:

- find documents without electronic signatures
- flag transactions from a particular (suspect) account
- identify foreign transfers larger than US \$10,000
- identify transactions of repetitive nature and very small dollar impact (salami technique fraud)
- random sample for manual audit

These tools could be linked to a higher level of intelligence analytic engine that would:

- link master files to transaction files and identify unusual patterns
- flag accounts with market potential
- offer services/products to clients with unusually low turnover related to their regular pattern
- agent for the entity with customer's buying agents

These tools could be linked to an even higher level of intelligence analytic engine that would:

- create a program of "audit by exception"
- negotiate with suppliers and banks for automatic confirmations
- issue system assurance certificates based on indices of financial performance and control measurements a la "continuous control monitoring"
- perform accounting services for third parties at the same system that performs such internally
- perform joint accounting for joint processes of a supplier and its downstream correspondent

These tools could be linked to an even higher level of intelligence analytic engine that would:

- issue audit attestation to systems and clients
- resolve issues with discrepancies in confirmations
- issue automatic consolidated statements

Consequently, intelligence in financial systems can be viewed as a pyramid of increasingly complex services that can be mounted one on the top of another. While a generic model can be built of an hierarchy of agentiation, different industries, cost structures, needs,



states of development of systems, security, and corporate cultures will inevitably develop focuses and heterogeneities.

### ***Financial Reporting Agents***

It is likely that the online world and marketing considerations will considerably change the world of financial reporting. Even at the current basic status of the Internet we can find additional financial information and financial views progressively leaking into corporate published reports. IBM, which two years ago placed a bitmap of its reports on the Web now has a tool that allow the user, on a form interaction basis, to obtain a few charts related to IBM's performance.

While disclosure rules, corporate legal liability and rigidities in tradition limit the speed of development, the user public will greatly benefit from new views at corporate data, saving from electronic distribution, and the potential for substantial increase or user-friendly corporate reporting information.

Among potential developments we may see:

- multilayered reporting: where certain users can drill down to several levels of detail on financial reports
- public management accounting: where certain users can share certain corporate internal reports
- analytic postings: where companies place in their public areas news and press releases as well as articles published about relevant topics, their performance, events in the company, and others.
- electronic distribution of financial statements
- monthly financial statements
- substantial increase of detail per account

### ***The future of financial reporting and data retrieval***

The magnitude of accounting related agents over the auditing and reporting process is likely to increase dramatically in the near future. HTML (Hypertext Markup Language), which was designed as a formatting language for presentation imposed a very complicated task for intelligent retrieval agents. HTML was designed to present information for human users and therefore is not machined friendly. Computer programmers that developed intelligent agents had to extract categorize, and analyzed data that was displayed in an HTML format. This delicate task resulted in a limited functionality of Internet intelligent agents. The transition towards XML (Extensible Markup Language) will facilitate the creation of highly functional agents. XML is a meta-language used to define either domain or industry specific languages. Through supplying DTD (Document Type Definition) each domain can provide rules to define elements and structure for industry specific languages. XBRL (Extensible Business Reporting Language) is an XML type of digital language for the business domain. XBRL is a framework that will provide the financial community with a standards-based method to prepare, publish, exchange and analyze financial reports and the information they contain. XBRL, which will be freely licensed, will also permit the automatic exchange and reliable extraction of financial information among various software applications. Intelligent agents will be able to access financial statements and other financial related sources, gather unambiguous data, and analyze it on the fly. This will create a powerful way to incorporate timely financial and non-financial information in the auditing and financial statement analysis process, which in turn lead to accurate and timely decision making. In conclusion the accounting/finance area offers great potential for the application of intelligent agent technology. It is clear that *a hierarchy of increasing complexity of services* will have to evolve from the current basic features. Among the basic principles that can be inferred from the above discussion:

a baseline of measurement level services already exists in corporate information services

these are eventually to be linked to the world through Inter/Intra-nets.

substantial degree of capacity of trustworthiness will be necessary, in particular for financial systems

on top of these measurement services, some degree of cooperation is necessary to host client's queries, and to serve wandering agents of the own entity

internal analytical services can be developed without many of the security limitations that the above pose

structural rigidities will hamper deeper reporting but market forces and new economics will substantially foster change.

## ***Conclusions***

In a few years, Intelligent agents will change the way people work, but not the industry structure. It will also change financial reporting, but time will pass before legal and statutory entities understand the need for these changes. Intelligent agents should also greatly facilitate electronic commerce on the Internet. Although the development of real intelligent agents is in its primary stage, it has a very promising future.

### ***Summary:***

Industry has made many claims about agents. The current crop of systems that provide some degree of agency is widening but their degree of real intelligence is very limited. The obstacles for this intelligence are more societal than technological, while this field still can benefit from major technological breakthrough, in particular concerning security and cooperative mechanisms.

For example, these societal obstacles that include trust by the public and competent mechanisms may impeach easy progress in the transport industry, where many customers may simply reject the use of intelligent agents and insist on calling travel agents. In the absence of travel agents, airlines would be forced to either increase their points of presence, or set up a booking service that would end up replicating travel agency! By-passing travel agents will not be an easy task. It would not take long for airlines to concede that outsourcing travel arrangement services to travel agency makes the most sense. Clearly, the role of an intermediary is currently important for the industry.

While the travel industry is more mature in the use of automated agents, the financial industry is just incipient. These barriers are probably magnified with the liquidity of many financial transactions, and therefore the societal obstacles larger.

### ***Technology***

Intelligent agent programming languages will continue to be an area of intense research. A Unified Agent Architecture must be developed and accepted by key industry players in order to standardize on intelligent agent infrastructure. General Magic's Telescript technology will be the most likely technology implementation for intelligent agents.

### ***Time Scales for Acceptance of Intelligent Agent Technology***

The use of true intelligent agents will not become widespread until electronic payment systems (one of the baseline services) become accepted in the Internet community. Another basic factor is the speed and method by which the usage of the Internet spreads though the country. The fact that the Internet can form virtual communities, allows for markets within markets to develop without ubiquity of service. The economics, however, of the situation changes substantially with population density

Other baseline events, that must happen in the financial area is the increase of trustworthiness of agents and the critical mass of financial entities or reports that are presented on the Net. This will progressively be enhanced by conversational and interactive methods of relationship prior to the full operationality of full-fledged intelligent agents.

### ***Uses of Intelligent Agents***

Intelligent agents will be used for information retrieval before electronic shopping. The same is true for financial services. Quicken financial services already offers some degree of information retrieval, financial services of substance are still in the drawing board. Financial reports and some degree of graphing is already on the Net, massive information acquisition or trading on this information is still far away. The key is user acceptance, and natural evolution of services. It is unlikely that users will soon entrust their agents with purchasing authority. Instead they will use agents to manage information which facilitates purchasing, paying for the service instead of the purchase.

This paper described several basic issues and definitions of features of intelligent agents, it displayed some current applications classifying these into five types: learning agents, viewer agents, shopping agents, search agents, and remembrance and helper agents. While these taxons are the state-of-

the-art substantial increase in intelligence through cooperation in expected soon. Financial and accounting agents were presented and some basic concepts in their development were discussed. Substantial research is needed for fully developing these concepts and principles.

---

<sup>i</sup> To date, Telescript has only been implemented by AT&T's PersonaLink service, implemented on the Sony Magic Link platform. [10] (Sony's Magic Link also uses General Magic's Magic Cap operating system.) This record is not bad for an emerging market. General Magic has the backing of such industry giants as Apple, Motorola, Matsushita, and Phillips. General Magic and its Telescript language show great promise for bringing intelligent agent applications to reality.