EVOLVING PERSONAL AGENT ENVIRONMENTS TO REDUCE INTERNET INFORMATION OVERLOAD: INITIAL CONSIDERATIONS

J. Pagonis* and M. C. Sinclair†

Information Overload

In everyday life, the Internet user is faced with the ever increasing problem of information overload, whether this occurs at home, at the workplace, or as may soon happen, everywhere [1]. The overwhelming information feed that computer users face leads to anxiety, strain, inefficiency and finally results in uninformed (or misinformed) and frustrated users [2, 3].

Continuously and increasingly, Internet users are confronted with laborious and difficult tasks of information filtering and/or gathering, which are inherently computer-oriented processes and need not involve human labour. Moreover, the efficiency and speed with which computer systems may retrieve and process massive amounts of data actually exacerbates the problem of ineffectiveness in information filtering without user intervention.

Users of the World Wide Web (WWW), and of other similar networked information services, can experience the phenomenon of information overload by either initiating a *Web search* on a particular subject or by simply—and perhaps more commonly—*habitual surfing* of the Web for subjects or news of interest. There is also the case where the user, taking a starting page (usually termed a *portal* page), begins surfing and becomes lost in the Web by following the hyperlinks embedded in that page (the embedded digression problem [4]). This case we simply regard as part of habitual surfing.

In the first case, in trying to satisfy a specific query, information overload occurs when the user is presented with many (sometimes thousands of) possible answers, which the user then has to process (read, classify and filter). The reasons, such as the vocabulary problem [5], and issues behind this inefficiency are numerous and the researched solutions many. Currently this is a pressing issue that has in the past few years led to several, but still incomplete, commercial solutions.

However, in this paper we will address the other major information overload problem, arising from habitual surfing. We will present our proposed research programme and also outline and support a potential solution. So far this issue has been far less visited and researched, though with the ever increasing use of computers at home and in the workplace, as well as the imminent arrival of information appliances, this may well change [6].

A major goal in tackling the problem is to provide a human-centered solution to information overload. Although needing a panoply of technologies and engineering practices to reach our target, nevertheless we envisage that a successful solution will in the end have to be judged by human users. However, this aspect of our research programme is outside the scope of this paper and will only be touched upon.

Related Work

Current solutions to the problem of information discovery and filtering of Web content do not seem to produce satisfactory results, leading to information overload, and usually require trade-offs from the

*J. Pagonis is with Symbian Ltd., 16 Harcourt Street, London W1H 1DT. Email: `john.pagonis@symbian.com`

†M. C. Sinclair is with the Dept. of Electronic Systems Engineering, University of Essex, Wivenhoe Park, Colchester, Essex CO4 3SQ. Email: `mcs@essex.ac.uk`; URL: `http://esewww.essex.ac.uk/~mcs`

user's perspective. They can be categorised as commercial and research/academic solutions. Interestingly, all of the current commercial solutions we have examined try to solve the first information overload problem, namely that of the user trying to satisfy a spontaneous query (usually a keyword search).

Commercial Approaches

The current commercial services offer the user an interface to massive collections of (partially) indexed Web document databases; hence most of the time they are referred to as *search engines*. The majority of these make the user apply logical operators on keywords to form what is effectively a database query. Others hide this ugliness through basic natural language recognition interfaces.

Alta Vista, for example, (`www.altavista.com`) employs massive clusters of servers over high bandwidth networks in order to service user queries on large full text indices of the Web. These databases have been assembled by indexing results of searches performed by webcrawlers which work using breadth-first search (BFS) algorithms. Nevertheless, Alta Vista, which prides itself in having the most comprehensive Web page index, can only cover a small proportion of the Web space, roughly 16% [7].

Unlike the brute-force approach that Alta Vista favours, other search engines like Lycos (`www.lycos.com`) apply, together with large databases of indexed documents, certain heuristics in order to offer a slightly more intelligent approach. Lycos does not index the full documents, but rather some portions of them and takes advantage of HTML's structure in order to increase the relevance of the retrieved documents. It tries to achieve this by looking at the *metatags*, and other structural constructs of each page, in order to locate important keywords (*e.g.* headings, author). Interestingly though, as a recent study showed [7], only 34% of homepages use such important keyword and description metatags in their structure.

Similar to Lycos' semi-intelligent approach is how Google (`www.google.com`) works. Google uses the "Web's link structure as an organisational tool" [8], where it interprets a link from page A to page B as a vote of confidence towards B. The grading of importance system that Google uses is enhanced by also reviewing the importance of a voter. The results that Google returns are filtered to contain the relevant terms asked. Also, returned pages get a higher relevance score if the requested terms appear at important locations within the retrieved pages and relatively close together (*cf.* Lycos), so that the user would ideally spend less time going through irrelevant material.

Directory-based solutions, like Yahoo! (`www.yahoo.com`), which try to present a structure over the vastness of Web space by building large categorised directories, are also very popular. Yahoo! uses (as most search engines do) webcrawlers for the automated collection of Web pages, although page owners may also register directly with Yahoo!. The way that Yahoo! gives structure to its directory, though, is by having a large staff who manually categorise and filter the Web pages retrieved. Thus, when users need to find information on well documented subjects, they have to manually navigate through the directory and/or apply a search engine query on it. Yahoo! tries to make the separation between the different categories as unambiguous as possible, hence making it easier and faster for the user to discover the area of interest, to which a local search can also be applied. Although popular, and many times effective, this approach suffers from the limited ability of humans to categorise almost 1 billion pages [7], growing exponentially in number, combined with the need for users to effectively perform part of the search themselves.

Another directory-based search engine, which tries to present the user with a more friendly interface, has been implemented by AskJeeves (`www.askjeeves.com`). Again the categorisation of popular subjects is performed by the company's staff. However, AskJeeves takes a rather interesting approach by indexing and categorising popular queries. By doing so, when users ask a question, similar or close to one that has been categorised before, they are presented with an answer which points to the correct category or site. In this way AskJeeves hides the arcane (for many people) operator-based queries from the user and appears to offer a more human-like behaviour. Unfortunately, this human-like interface, which anthropomorphises the service, leads to users having overblown expectations: thus most of the time the expected performance cannot be met [9].

AskJeeves also performs a *meta-search* on other search engines and presents the user with the other engines' possible answers. This strategy of a parasite engine, which uses other engines for the retrieval and initial filtering of information, is used by MetaCrawler (`www.metacrawler.com`) as well, and promises to offer more diversity in the coverage of the Web space. Nevertheless, it does not seem to reduce the time users spend on filtering, nor it can guarantee that the different search engines used cover different parts of Web space.

One of the latest commercial attempts to provide an information filtering agent is that of Apple with the introduction of Sherlock [10]. This is a particularly interesting development due to the combination of features chosen, and also the fact that it is tightly coupled with an operating system. The first obvious difference is that Sherlock lives client-side, on the user's computer desktop. Hence part of the filtering has been disassociated from the search engines. This disassociation also allows Apple to offer one user interface and control center for many search engines. Acting as a meta-search agent, Sherlock unifies many search engine interfaces and databases under one, and employs them as information discovery agents. Sherlock facilitates what is, in effect, activity-based information retrieval, by grouping sets of specialised search engines in what Apple calls *channels*. Every channel is responsible for a single activity, for example, searching for people on the Web, or finding the cheapest book from the online bookshops. Each specialised channel is comprised of several search engine *plugins* which have been tailored for that search engine specifically. These customisable plugins, which can be edited by the users, are effectively filter templates which hide many parameters behind the search engine queries (like the number of hits returned). It seems likely to us that Apple will move Sherlock into the intelligent agent arena in the near future.

Research/Academic Approaches

Certainly the most promising solutions at present are the research/academic ones. However, in this limited review, we will only try to present a selection of the projects that have influenced, fueled, resemble or in some way complement our proposed direction. Having given an overall picture of the commercial solutions in use, and which thus represent the present, in this section we give a glimpse of possible futures.

Letizia [11], a project developed at the MIT Media lab, aims to autonomously and transparently assist the user while browsing the Web. The first task that this agent performs is to quietly monitor the user's actions, in order to infer patterns of behaviour, using various simple heuristics. In parallel to the user's browsing, Letizia explores the same Web space, sourcing from the same collection of hyperlinks that are available to the user. This exploration is performed by a BFS on the hyperlinks of each visited Web page. Then, using the knowledge on the user's inferred interests, Letizia, when asked, makes suggestions of possibly interesting links to follow. Letizia is useful because it effectively performs the browsing that the user would otherwise do, and offers its recommendations only when the user feels stuck or lost in Web space.

Always complementary to the problem of information overload due to habitual surfing is the problem of searching the Web; hence the use of search services. One such search engine research project, similar to the commercial Google search engine, has been developed by IBM and Cornell as the Clever searching project [12]. Clever tries to take advantage of the structure of linked pages on the Web in order to increase the relevance of retrieved documents. It achieves this by investigating the origin of the Web pages that reference the page in question. When a page has been referenced by a site (or page) of great relevance, that is termed an authoritative site, resulting in the referenced page automatically getting a higher ranking on a query's hotlist. Hence by creating a web of hubs and authorities, Clever makes use of the human judgement behind the construction of Web pages which reference others [13].

Encompassing both approaches of surfing and searching the Web for efficient information retrieval, WebMate [14, 15] implements many interesting strategies. WebMate, which comes in the form of a *proxy* and *user interface control applet* pair, constantly monitors the user's actions in order to incrementally establish the user's interests. The system makes automatic *user profile* discovery and incrementally classifies user browsing habits in a number of finite interest domains. The Web-page relevance feedback is performed by the user, while browsing, by giving the system only positive feedback. The accumulated knowledge on the interest domains is then used to make Web page recommendations (*cf.* Letizia [11]) and for compiling a daily newspaper tailored for the user. WebMate also enhances Web searching by applying trigger-pair keyword refinement [15] on parallel searches (meta-search on many search sites) like MetaCrawler.

During ongoing research for the Illinois Digital Library Initiative, genetic algorithm (GA) based spiders were evaluated for global and optimal searches on the Internet [4, 16]. This research compares and contrasts the use of GAs with conventional BFS spider algorithms. The offered user interface allows real-time presentation of progressive searches for user evaluation. The bootstrapping of the system is achieved by the user indicating Web pages of interest. These seed pages are translated to weighted keyword vectors which are then used for the evaluation of each visited Web page. Although the user interface provided in their experiments presented the retrieved, and most relevant, pages in real-time, nevertheless it did not allow for real-time relevance feedback, or indeed any feedback at all. This idiom of the system, although less task intensive for the user, made the bootstrapping of the system very important. Indeed, as was discovered, the choice of seed pages indirectly determined which search method (GAs or BFS) appeared better. For example, the experiments showed [4, 16] that seed pages with a high density of interconnections favoured the use of the BFS algorithms, whereas if the user had initialised the system with homepages containing few links, GAs performed better. Apart from discovering that the two techniques were complementary, the research also demonstrated the exploratory nature of the GA approach, which discovered relevant pages that were not connected in any way with the rest.

An artificial ecosystem of evolving agents that adapt and compete in order to assist the user with information overload was implemented in Amalthaea [17, 18], an MIT Media Lab project. In this ecosystem, software agents are divided into two categories, those responsible for information discovery (termed *ID agents*) and those involved in information filtering (*IF agents*). In this model the latter are responsible for learning and adapting to the user's interests using augmented weighted keyword vectors in their GAs' genotypes. In Amalthaea, the choice of the user domain interests is finite and is set up by the user. For each such domain of interest (or profile) a cluster of agents is assigned and independently evolved. Bootstrapping of the system is very important and in Amalthaea this can be achieved in three ways: by specifying keywords, Web pages and/or passages of text on the subjects of interest. Then, while browsing the Web, Amalthaea (whose engine lives in a server/proxy) monitors the user actions and thus continuously adapts. The user can, at any point, request a (continuously updated) digest from the interest categories that have been specified. The user interface allows for relevance feedback to the system so that it can then dynamically adapt. The results from the Amalthaea experiments are very encouraging and clearly show that adaptation to user interests can happen quite effectively, and that the system can swiftly recover from large changes in a user's profiles [18].

Trade-offs

What some search engines offer is to customise a small part of their Web space according to the user needs, in order to offer the user with a personalised entry point to the Web, *i.e.* a portal. Such customisation, though, is static and limited (in both granularity and interest areas) and most of the time involves disclosure of personal information. These personal preferences and habits, like online buying habits (mediated by the portal) or portfolio monitoring, are tracked and may be sold for marketing purposes. Such trade-offs, which undoubtedly bring revenue to the operators of these sites and keep these services 'free', are not negligible for the privacy-concerned user. As Web users become more aware of their privacy rights and privileges, this formula may well have to change.

Other search engine sites allow, and hence indirectly, make the users navigate through their directories in order to discover the information themselves. Although these services are welcome because they index, and hence in a way centralise (or appear to do so), information, nevertheless they will soon suffer from the rapid growth of Web space. This will lead to a smaller fraction of the Web being indexed and, at the same time, require more time from the user to navigate through them.

Another trade-off, which also leads to yet more information overload, is the increasing use of advertisements. Those which come in the form of banners (within the search engine/portal sites) or, even worse, pop-ups, disrupt the user and consume vital bandwidth. In the case of mobile Web users, who may use portable computers or Wireless Information Devices (WIDs) connected over mobile cellular networks, these advertisements are costly both in terms of operator charges and of the limited bandwidth that is available to them. On top of that, the actual screen space that these advertisements occupy make it practically impossible to use several Web sites.

None of the approaches to information filtering and discovery, at least so far, take into account agent or user mobility, or keep portable/wearable devices in mind, nor do they allow for multiple platforms to be used in conjunction with physical and locational changes across time and/or space.

Evolving Personal Agent Environments

As a means of dealing with the problem of information overload, particularly in the domain of the WWW, this paper outlines a proposal for a personal and adaptive agent-based system. Central to the design of the suggested system are the use of evolutionary algorithms, agent technologies, collaborative filtering, privacy, agent-society trust relationships, location awareness and mobility, as well as continual adaptation.

As a major target, the proposed system aims to constantly and continuously evolve and adapt to the locational and behavioral changes of the user. Such a system will not be so much an advanced search engine, but rather a *personal agent environment*, which aims to transparently assist in the filtering and presentation of information.

This will be based on observation of the user's individual behaviour, enhanced through adaptation and mobility. The goal is to design, and subsequently implement, a mobile personal system that eases the information overload problem and closely follows the individual user's interests. Only a truly personal system will be in position to cope with such demanding tasks.

Such a personal system will be able to 'know' and record sensitive user information, logically and physically (in the form of a portable/wearable device host) follow the user, and thus constantly and continuously evolve. In a world of ever-diminishing privacy, only a truly personal system may hold and negotiate relationships as a user's surrogate [19].

Truly Personal Portal

One major difference between the current approaches to information overload and the proposed system, is the fact that the filtering and retrieval of information should ideally happen prior to a given user request. We propose a system that will not aim to solve the information overload problem resulting from a user making a spontaneous specific query, but rather it will try to solve the habitual-surfing information overload problem. It could thus be regarded as a personal portal to the Web, but unlike the ones currently offered by companies such as Yahoo!.

The difference between the proposed system and these commercial solutions is that the agent environment will evolve as the user's interests evolve. It will take user satisfaction into account, without disclosing personal information, and it will be mobile with the user both in terms of physical location (when possible) and host platforms. Moreover, it should not introduce appreciable delay [4] in the presentation of the information, as the majority of retrieval and filtering will happen prior to the user needing it.

Thus our approach to solving this particular aspect of information overload is to have a truly personal, adaptable and evolving portal to both the Web and other networked information, such as newsgroups, rather than a simple personalised homepage, as may be currently offered.

Mobility and Location Awareness

Location awareness (or context sensitivity [6]) comes as a direct result of the use of portable computers, personal data assistants, WIDs, wearables and other enabling technologies, like mobile IP, cellular and wireless networks. This ability of the mobile device, and hence of the agent environment, to discover its location, will enable new ways of following the user's behaviour, habits and interests, thus adapting and evolving more closely to them.

For example, it could mean that a different portal/digest might be presented to the user, according to different profiles throughout the day, *e.g.* when in the office, when at home, when on the bus. All the user satisfaction feedback scores, behavioral and locational changes could then be recorded. Then, when evolution next occurs, it will include all these different individual profiles. In addition, statistical behaviour could be analysed to anticipate the next locational/calendar profile change, in order to be in a position to present the user with the correct portal.

Possibly, the user should confirm creation of each different profile to allow the agent system to 'legitimately' evolve them. Though the system, based on statistical inferences, could simply make the user realise such profile changes by asking the right questions: "I see that you are on the bus quite often on Monday mornings, do you feel like reading the morning news then?"

An initial survey by the first author (on highly computer-literate and Internet-using subjects, with diverse backgrounds) showed that a large percentage of the subjects (41%) change their Web surfing habits according to location. The same survey showed that more than one third of the subjects look for different kinds of content during different days of the week, though only 16% of them seem to change their surfing interests during different times of the day. This latter behaviour, though, may possibly change with the arrival of portable WIDs and low cost cellular Internet access.

This area is of particular interest, as the additional information overload resulting from such portable, wearable devices and information appliances in general may well exacerbate the problem to such a degree as to impact the users' social behaviour. At the same time though, context sensitivity and mobility will doubtless introduce continual change and development of the users' interests, with the potential for considerable benefits through suitably engineered personal and adaptive individual portals.

User Satisfaction

For the system to be truly personal and adaptive, user satisfaction must be an essential driver for system improvement. By taking this into account, the agent environment can improve and adapt to the user's individual preferences and views. While heuristic approaches, such as indexing and keyword matching, give good objective results, ideally each user would prefer results tailored to their own individual interests and profile.

Through employing an evolutionary-algorithm-based multi-agent environment, where the 'fitness' of each individual is determined by an estimate of user satisfaction, the system should more closely adapt to the user's personality and preferences, thereby yielding continually improving results.

There are several ways to estimate user satisfaction, both direct [17, 18] (like asking for user feedback explicitly) or indirect [11] (such as making inferences from the user's browsing behaviour). Selection of appropriate measure(s) of user satisfaction will form part of the programme of research.

User Interface

Closely coupled to the user satisfaction feedback is the user interface (UI), which will mediate and facilitate such human computer interaction. In fact the mechanisms by which user input will be fed back to the system, and by which information will be conveyed to the user, are critical to the acceptance and adoption of such a system.

The design of an appropriate UI will come in the later stages of our research and will emerge as an outcome of surveys and experiments with various user groups. The information overload problem is human-centered and that is how we wish to treat it. No matter how good or bad the potential technologies driving our proposed system, the human should always be at the center of the delivered solution.

One major challenge to the decision on the most appropriate UI, is the plethora and diversity of devices and form factors that the system will be hosted on. However, as general guidelines, we wish to provide an operating-system independent UI, with minimal (if possible) functionality. These will make it as straightforward as possible for the new user to adopt and use. For example, we anticipate the potential for the user to be distracted (during the presentation of information) will have to be minimised as much as possible. Thus there should be only one possible view of the portal at any given time, directly related to the current profile of the user. The reasoning behind this approach is that although a user may be interested in many things (perhaps too many sometimes), there should be only a limited number of subjects conveyed at any time according to the active profile.

Agent Society Communication and Collaboration

In our approach we intend to exploit and benefit [20] from the inherent distributed nature of agents, which if considered as a whole, appear as an 'intelligent machine' [21]. Collections of agents, if broadly modeled on the real world, can form societies, which may be assembled from communities to which different groups may belong according to their relationships. All these hierarchies may in turn form or belong to different abstractions. In fact, any number of agent sets can exist depending on the one or more relationships amongst them.

If, in this context, we perceive 'agents' as the personal agent environments of each user, then it is easier to see how these relationships can be built depending on the users' real-life relationships. By applying a protocol with which these relationships can be exercised, evolved and built, then communication can take place between the agents on behalf of their users. More importantly, such communication will be in line with the users' preferences and 'politics', while at the same time happening autonomously without engaging the users themselves. This latter quality means that filtering and information discovery can happen collaboratively, hence probably more rapidly and efficiently, and at the same time without breach of trust and any consumption of the users' 'mental bandwidth'. We intend to investigate whether it is possible to achieve transparent collaborative filtering and information discovery within groups of evolving personal agents, as well as determining the effectiveness of this approach.

As an example, illustrated in Fig. 1, an evolving personal agent system may retrieve (possibly) relevant information by crawling the Web (or any repository) or by utilising other agents. In the second case, where collaboration and communication between agents may take place, the system could take advantage of the agent societies in two ways:

- By explicitly communicating with various agents from a known *agent pool*
- By using *brokers*, which act as 'info-mediaries' when the location or identity of other agent systems is not known (for practical, security or privacy reasons)

In the above two cases some form of a reputation or economic model (similar to the ones used be-

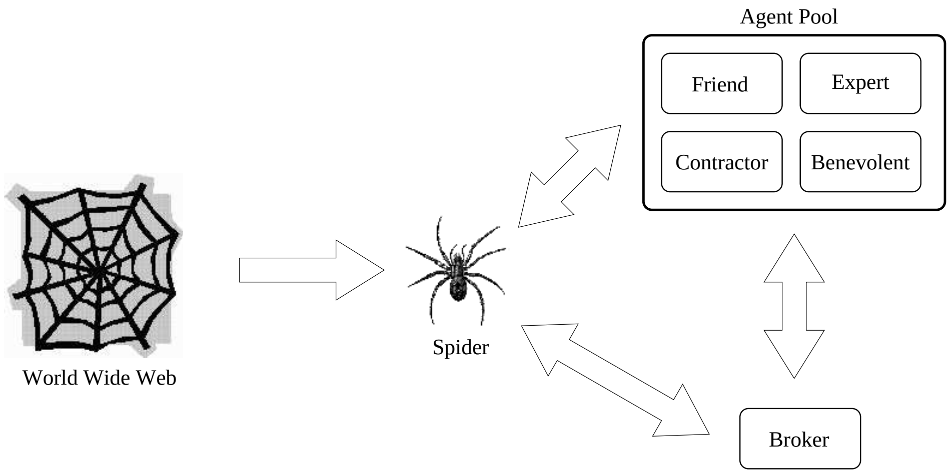

Figure 1: Discovery of Information

fore [17]) adapted for and applied on the inter-agent ecosystem, may possibly prove beneficial. *Benevolent agents* [22], served and contracted by brokers, may well speed up information retrieval as well as allow for better exploration of the search space. At the same time exploitation of *expert agents* may be possible. Moreover, brokers may prove very important in a domain where lots of volunteer (and possibly expert) agents may exist, such as a company or organisation.

The goal here is not only to make use of the network, but also to take a step towards exploiting and enabling the intercommunication of humans and agents as part of one cooperative, self-sufficient and evolving information society.

It should be noted that in such an arrangement, flow of information may come and go in various directions at machine speeds. Hence it is important to have mechanisms by which to delegate trust and enforce communication policies which are in accordance with the users' wishes.

Privacy

For a truly personal agent environment, where data about interests, habits, relationships and other sensitive personal information exist, privacy is a vital requirement. Users will only be frank and confident with their agents if they know that their privacy is secured and guaranteed at all times. Privacy fulfills a major part of the need of the human to be in control of the system [9] and of their own personal space; this to a large extend will decide whether a particular agent system is adopted. Only a personal system should record and hold personal (and private) information. Therefore, in a system where filtering, evolution and adaptation happens separately to information discovery, it should not be possible to track user preferences profiles and habits externally.

For example, consider a setup where the user has a home computer and a WID. In this configuration the (more powerful) home computer could be used for evolving the agent system, whereas the more personal WID is carried at all times by the user and used when away (from the home computer). Whenever the personal portal is used, the WID records the user's Web habits. It will then feed these back to the home system either when the user returns or over a network connection. The latter could be provided via a wireless network and an Internet Service Provider (ISP).

The personal portal would have been readied prior to the user needing it, possibly from the home computer, where lots of information retrieval and filtering could take place. When this is transferred to the WID, the ISP or an eavesdropper could be in a position to monitor the transferred information. At the same time though, due to the system's decentralised nature, they would not necessarily be in a position to know or even infer what information filtering took place, and which user's tastes were actually satisfied. However, taking this a step further, even information exchanged between trusted

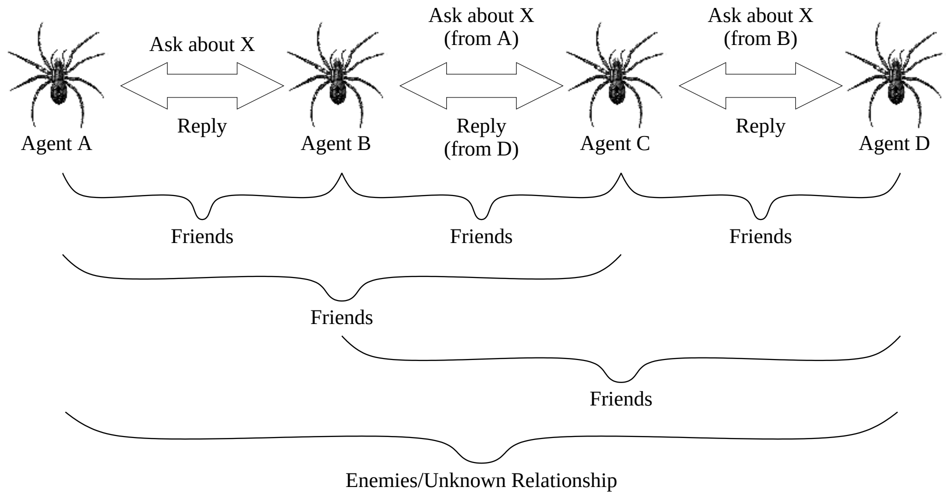

Figure 2: Trust Relationship Propagation Protocol

systems and/or brokers will probably have to be encrypted for enhanced security.

In this minimalistic example, at no point was information filtering preferences, or personal information, willingly handed to anyone outside the user's systems. If there is a need for such sensitive information to be exchanged (for collaborative discovery and or filtering), then we propose the application of *trust relationships*, between the agents, modeled on the users' real-life relationships and at the same time enforcing a *trust propagation protocol* (see Fig. 2).

Conclusions

There are currently two main ways to get lost in the Web, the first is to try to answer a specific spontaneous query by directly searching the Web (using a search engine) and the other is habitual surfing of the Web to discover subjects and information of interest. Both ways lead to information overload and for the latter in particular, we anticipate that truly personal and evolving agent systems will provide a solution.

We will thus try in our research programme to craft a system that adapts according to the user's interests, that is truly personal and mobile and which, more importantly, constantly and continuously adapts to the user's behavioral and locational changes without sacrificing privacy.

References

1. Halla, B.: "How the PC will disappear", *IEEE Computer* v31 n12, 1998, pp. 136–138
2. Ireland, P.: "Management paradigm to reduce information overload", *Computing & Control Engineering Journal* v10 n1, 1999, pp. 29–32
3. Bowman, C.M., Danzig, P.B., Manber, U. & Schwartz, M.F.: "Scalable internet resource discovery: research problems and approaches", *Commun. ACM*, v37 n8, 1994, pp. 98–107
4. Chen, H., Chung, Y.M., Ramsey, M. & Yang, C.C.: "An intelligent personal spider (agent) for dynamic internet/intranet searching", *Decision Support Systems* v23 n1, 1998, pp. 41–58
5. Chen, H.C.: "Collaborative systems: solving the vocabulary problem", *IEEE Computer*, v27 n5, May 1994, pp. 58–66
6. Billinghurst, M. & Starner, T.: "New ways to manage information", *Computer* v32 n1, 1999, pp. 57–64
7. Lawrence, S. & Giles, C.L.: "Accessibility and distribution of information on the web", *Nature* v400 n6740, 1999, pp. 107–109
8. Why Use Google?, `http://www.google.com/why_use.html`

9. Norman, D.A.: "How might people interact with agents?", in Bradshaw, M. (Ed.), *Software Agents*, Ch. 2, MIT Press, 1997
10. Sherlock 2, `http://www.apple.com/sherlock`
11. Lieberman, H.: "Letizia: an agent that assists web browsing", Proc. Intl. Joint Conf. on Artificial Intelligence, Montreal, Canada, August 1995, pp. 924–929
12. IBM Clever Searching, `http://www.almaden.ibm.com/cs/k53/clever.html`
13. Kleinberg, J.: "Authoritative sources in a hyperlinked environment", (to appear in) *Journal of the ACM*, 1999
14. WebMate, http://www.cs.cmu.edu/˜softagents/webmate.html
15. Chen, L. & Sycara, K.: "WebMate: a personal agent for browsing and searching", *Proc. 2nd Intl. Conf. on Autonomous Agents and Multiagent Systems*, Minneapolis, USA, May 1998, pp. 132–139
16. Chen, H., Chung, Y.M., Ramsey, M. & Yang, C.C. "A smart itsy bitsy spider for the web", *Journal of the American Society for Information Science* v49, n7, 1998, pp. 604–618
17. Moukas, A.: "Amalthaea: information discovery and filtering using a multiagent evolving ecosystem", *Applied Artificial Intelligence* v11, 1997, pp. 437-457
18. Moukas, A. & Maes, P.: "Amalthaea: an evolving multi-agent information filtering and discovery system for the WWW", *Autonomous Agents and Multi-Agent Systems* v1, 1998, pp. 59–88
19. Krulwich, B.: "Automating the internet: agents as user surrogates", *IEEE Internet Computing* v1 n4, July-August 1997, pp. 34–38
20. Wooldridge, M.J. & Jennings, N.R.: "Software engineering with agents: pitfalls and pratfalls", *IEEE Internet Computing* v3 n3, May-June 1999, pp. 20–27
21. Maes, P.: "Patie Maes on software agents: humanizing the global computer", *IEEE Internet Computing* v1 n4, July-August 1997 pp. 10–19
22. Huhns, M.N. & Abdulla, M.: "Benevolent agents", *IEEE Internet Computing* v3 n2, March-April 1999, pp. 96–98