On Mobility and Agents

Anupam Joshi

Abstract. The ability to access networked resources, both computational and information, from mobile clients will become increasingly importance in the emerging pervasive computing scenario. There is significant ongoing work in the research community in this area. The objective of this paper is to argue that an agent oriented approach can be used to create systems to enable the goal of ubiquitous access. We show examples from existing work, including our own, where an agent abstraction has been used. We also outline our ongoing work to use a multiagent approach to enable the next generation, ad-hoc network based mobile applications.

1. Introduction

The trend of computing today is clearly towards an environment where computers are pervasive. This is enabled by two trends which are poised to accelerate in the near future. One is the shrinking form factor and cost, and the increasing MIPS of computers. This has allowed for affordable palmtop computers and desktop computers that have the power of supercomputers from less than a decade ago. It will lead to both high end petaflops machines as well as embedded computers in most engineered artifacts. These artifacts span the scale from roads and bridges to cars to even items of clothing. The second is the advent of high speed optical networking and RF wireless networking. The former permits extremely low cost – extremely high bandwidth network backbones and residential connections, the latter provides for universal connectivity. Related to these two are possible developments of micro sensors and actuators (MEMS).

Concurrently, with the advent of dynamic content (e.g. server parsed HTML, CGI) and executable content (e.g. Java), integration of security mechanisms, and emerging metadata standards, it is clear that the Web is transforming into the basis of a globally distributed computing and information access system. For example "Road Warriors" (business users with palmtop/laptop type devices) constitute a large and growing segment of users. Their work typically involves accessing and modifying corporate information repositories (with multimedia data) over low bandwidth networks (wireless, phonelines). More often than not, the applications they use are web enabled and use the browser as a "thin" client. Given the emergence of the Web as the "kernel" for distributed information and computation access, a

This work was supported in part by an IBM faculty development award and NSF award IIS 9801711.

system to permit mobile web access can allow a variety of networked resources to be accessed from mobile clients.

This paper aims to discuss how agents can be used to solve the challenging problems in middleware and data management for mobile access. In particular, we will describe how agents have been used to support web access in the past, and outline our ongoing effort to create a framework that would support applications for next generation ad-hoc networks.

It is well understood in the mobile research community that the web (more specifically, the HTTP protocol) is designed to work in wired, high bandwidth environments, and does not operate particularly well when the access point is a mobile host[**JWM+96, JWH97, Kat94**]. The reasons behind this are quite well known (e.g. low/variable bandwidth, disconnections, etc.[**JWH97, FZ94**]). Further, the mobile host is typically resource poor. Even though the very high end laptop machines can now deliver performance comparable to low-end to moderate desktops, most "thin and light" subnotebooks and PDA type devices are constrained in terms of CPU, power, memory, disk, display capabilities etc.

An argument can be made that these bandwidth and resource related problems are transient, and eventually wireless network speeds as well as the resources available on mobile platforms will increase. That may well be the case, yet the speeds on wired networks and the resources (memory, CPU speed, etc.) on static hosts will also increase. As has been amply demonstrated over the last decade, software catches up and uses all available hardware resources – Existing (and novel) applications will evolve and become dependent on these enhanced capabilities. Thus while the absolute performance measures of mobile systems will undoubtedly improve, what we call the *bandwidth gap* and *resource gap* will remain.

2. Agents to Enable Mobile Web Access

The problems in web browsing caused by the low/variable bandwidth of wireless networks, and the mobile client's limited resources are compounded by the current model of web information access, where the user has to navigate the information space to find information that s/he needs. Bandwidth, a precious resource in the mobile scenario, is thus wasted by transmitting a lot of useless data across the wireless channel in the browsing process. This problem relates to *personalization* of the web space, *i.e.* locating the information needed by a user and possibly filtering it. Given the synchronous nature of the HTTP protocol, a continuous connection is also required throughout the information retrieval process. Disconnections typically require that the information be fetched again. More generally, the problem is a lack of *asynchronous operation* – *i.e.* not allowing the mobile user to disconnect in the process of information retrieval. Further, web servers have no knowledge about the client's resources and assume that it can handle the data sent by them. For example, a user may follow a hyper-link to a VRML world on a machine that does not have the capability of handling a VRML plugin. The client will retrieve the data from the server, and then not be able to use it. This results in consumption of available bandwidth and wastage of time while transferring data (and potentially even the plugin) across the wireless network – a *capability mismatch* in terms of handling content. Moreover, the mobile client expends battery resources in receiving this data, which it cannot use. With the ever increasing use of different types of multimedia content on the web, this phenomenon represents a growing problem.

2.1. Proxy Agents. There exists a large body of work, including our own, which handles the problem of *capability mismatch* for multimedia content in wireless web access. The solution, typically, has been to use a client-proxy-server model[**BJA98, FB96, BMMM96, NSN+, JdT+95**]. The proxy transcodes multimedia formats according to some predefined rules, usually in a manner that trades quality for bandwidth. This proxy has been viewed as an intelligent agent in our prior work[**JWH97**], as well as in the work of Brewer et. al.[**BKC+98**]. The basic idea is that the proxy, given certain (often implicit) rules, monitors (or is otherwise provided) network QoS and knows about client side resources. It then autonomously transforms the data stream being sent to the client so as to best match the network QoS and client capability. Thus the proxy represents an agent embodying the weak sense of agency.

However, as PDA or thin-light notebooks become more popular, the proxy's functionality (and hence the computational power it needs) will increase. For example, some proxies now seek to deal with videos as well as images. This includes work done at Berkeley, as well as our own recent work[**BJA98**]. Our work has also examined other questions that relate especially to PDA type mobile clients, such as what to do with active content and HTML fonts / styles which the PDAs typically cannot handle. This adds functionality to the proxy and thus increases the computational load it places on the hardware. There are also proxy approaches that even render the HTML in a format appropriate for the PDA and then transmit that[**FGG+98**], clearly adding to the computational burden at the proxy machine.

We feel that a purely proxy based solution will become increasingly non scalable, especially with the number of users connecting wirelessly expected to grow. The role of proxies has been recently questioned elsewhere as well – there is some debate as to whether proxy based solutions are really needed to provide networking services to mobile clients (cf. a panel on this theme at MobiCom98). Further, in ad-hoc networks which will be used in conjunction with Bluetooth/HomeRF enabled devices, it is not clear whether the client-proxy-server architecture is even applicable.

2.2. Agents to Negotiate End-End Performance. The alternative to proxy based approaches are end-end systems, where the server adapts the data stream it sent back based on some "negotiation" with the client. Such approaches are well known in networking and systems literature.

In the web context, dual versions (graphics heavy vs. text only) of web pages, with the user explicitly choosing between them, represent a simple form of the end-end approach . To the best of our knowledge, it was Seshan *et al.*[**SSK97**] who first presented the notion that the Web clients could use network performance parameters to download documents from a server at different "fidelities", and explicitly mentioned that this was something beyond text only pages. Implicit in their paper was the idea that the server would indeed have different fidelities to present to the client. Such a scenario goes against the basic assumption underlying proxy based transcoding, namely that the servers will have a document at a fixed "resolution" of a given medium.

Recent developments in multimedia coding also support an end to end approach. For example, new formats (e.g. wavelet based codes) that allow for efficient progressive transmittal of images and videos, and/or reconstruction of data at arbitrary resolutions[**VYZ+99**] from the same bitstream are being developed.

Moreover, with disk storage increasingly inexpensive, it is not altogether unrealistic to expect that content providers will react to the market and make available multimedia content in multiple (fixed) resolutions. A recent example is the mapblast driving directions site (http://www.mapblast.com/), which provides driving directions for the in special formats for palmpilot and WinCE devices, in addition to the regular HTMLized format.

With an end-end approach, the server can also carry out more complex transformations (such as summarizing text, converting text to audio etc.) based on the client device needs. Of course, these may also be computed offline and stored. The end-end approach is predicated upon the client being able to "identify" itself to the server. There has been a very recent initiative (CC/PP from W3C, http://www.w3.org/TR/NOTE-CCPPexchange) to use RDF based HTTP extensions as a standard mechanism for this. The WAP forum (http://www.wapform.org) Wireless Application Environment also provides for mechanisms where state information allows a server to send content optimized for narrowband delivery to a client. Alternatively, the client must have a way to express its preferences for particular resolutions to the server. This is similar in concept to expressing preferences for languages or formats as envisaged in HTTP/1.1 . In our recent work[**BJA98, Jos99**], we have shown how HTTP/1.1's content negotiation mechanism can be leveraged to negotiate multimedia fidelity between the web browser and the server.

In more general terms, the end–end approach represents an instance where the client (agent) and the server (agent) negotiate to get the right content to the user. Unlike the one-shot approach of our existing work, we can envision a more complex negotiation process. The server could present the client with choices for the resource requested in terms of access cost, accuracy or fidelity of information, alternative resources in case the server does not have the requested resource etc. For example, consider the case of a laptop client interacting with a VOD server (clearly, this hypothetical scenario must wait for the bandwidths promised in 3G or 4G wireless networks). Suppose the user requests *www.blockbuster.com/Shakespeare_in_love*. The server could respond with a choice to either provide MPEG2 content (broadcast quality) at a certain price, or QCIF MPEG1 content at a lower price. If this film was not presently available (due to too many requests already being serviced), the server could suggest the movie *Elizabeth* as an alternative. The client could make a counter offer (willingness to pay a premium to get the movie now, requesting a reservation for a future time etc. etc.). In other words, agent based end-end approaches can facilitate a more comprehensive interaction between the mobile client and the appropriate servers. Given the disconnection related problems, we envision that the negotiation on behalf of the mobile client would be done by a proxy on the wired side.

2.3. Agent Based Information Personalization. The functionality of information personalization has, to the best of our knowledge, not been much explored in the context of supporting mobile web access. We have shown[**JWH97, KKW+96**] how this can be used to minimize information flow on the wireless network and implemented it in a proxy agent called W^3IQ *(Web Intelligent Query)*.

Broadly, its operation can be described as follows: The user connects to the proxy agent, makes a request for information, and disconnects. The proxy then tries to assemble a set of URLs that would satisfy the request, and returns them to the user. Should the user be disconnected, the results of his prior request(s) are made available at reconnection. These results are URLs that the system feels will contain the information the user requested. Users then prioritize the URLs provided by the system by rating them, depending on their perception of the relevance of the URL. The user feedback allows the system to infer a "semantic" match between the user's query terms and the information in the document. This helps the system create a user's profile which is used to control the information s/he would receive in response to future queries. The ranking information, the URLs, and the corresponding query are all stored by the W^3IQ proxy as metadata. Over time, then, the proxy builds a repository of ranked URLs which can be used to answer future queries by this user or other users. The system allows a proxy to pass the query to other W^3IQ proxies in the network, to garner any information related to the query in their local metadata. This allows all proxies to share information (specifically, ratings in regard to particular query terms) about URLs obtained by one proxy. When no W^3IQ agent has the needed information, the query is passed on to Lycos.

However, this functionality can also be taken away from proxies and passed on to servers. The idea that web sites (more generally, information sources) should "adapt" based on the user has been discussed in the last few years. Besides our own work on web mining[**JK98, NKJ99**], there is also Etzioni's description of adaptive web sites[**PE98**], and the Webminer project of Han *et al.*[**ZXH98**]. For instance, we have created a system that mines a server's access logs for patterns of traversal. Specifically, we determine browsing "sessions" (URLs from the site accessed by the same host within a defined timespan). We define a similarity metric between two sessions based on how much overlap there is between the URLs visited. When comparing URLs, this metric takes into account the document hierarchy within the server. Given the sessions and distance between them, we use relational fuzzy algorithms to cluster them. This provides associations between URLs visited – in other words URLs belong to the same cluster when they are visited in the same session. Given such information, the server can then use cookie based mechanisms to associate a user with an access pattern, and tailor the information sent back. Once a user is identified with particular information access patterns, the server can tailor the information returned in response to their requests. The same result can be achieved by involving the user directly in the process (for example, by providing rankings or other feedback), although clearly this is more burdensome for the user. An even more primitive form of this idea can be seen in most "personalized" search engines, where users are asked to explicitly provide keywords that match their interest.

3. Agents to create next generation mobile applications

In this section we outline our vision of how agents will be used to create new type of applications that will take advantage of the next generation wireless systems. There is an emerging paradigm shift in the way we thing about designing and building complex systems in general, and the way we approach the design of mobile aware applications in particular. We are moving away from the traditional discipline of developing a single, overarching design for a complex system in which all of the parts are carefully engineered to fit together and toward a new approach in which the individual atomic components are designed to be autonomous ("active"), self-describing ("articulate"), highly interactive ("social"), and adaptive ("intelligent").

These components can be viewed as agents, and a multiagent systems approach can provide the basis for creating a robust environment in which independent devices(from computers, PDAs, routers, printers to even home appliances and cars etc.) existing in a location can and will discover, interoperate, and cooperate with the other devices in their vicinity in an as-needed and as-desired basis. The underlying communication technology is presently being developed by consortia such as Bluetooth (http://www.bluetooth.com/) and HomeRF (http://www.homerf.org/). Note that while we propose these ideas in the context of wireless ad-hoc networks, they will also be applicable to wired networks.

An important enabling component of this vision is the work in distributed object-oriented systems in general, and Java/Jini in particular, which our agent oriented abstraction will build upon. Specifically, systems such as Jini which enable the formation of a community of devices at the syntactic level make it convenient to add "intelligence" to the system and create communities of "agents" which can then user richer communication languages (ACLs) to communicate information at a semantic level – information about abilities, requirements, commitments, intentions, beliefs, goals, etc. State of the art systems such as Jini provide for networked entities to advertise their capabilities. They are sufficient, for example, for a bluetooth enabled PDA to discover that there is a bluetooth enabled printer in its vicinity. They are probably not sufficient for the printer and the PDA to "negotiate" an agreement about whether and when the printer would print the material given by the PDA. They are certainly not enough for one PDA in the vicinity of another to realize that both their users are trying to find the same conference room in an unfamiliar building, and agree to cooperate in the process. Clearly, the enabling of such ad-hoc collaboration leading to "teams" of devices cooperating to solve common and individual problems will allow for a new class of mobile applications to evolve.

Mechanisms to form ad hoc teams: A key to the realization of our vision is the ability for processes and systems at the application level to be able to easily cooperate and form ad hoc teams. We are building on our earlier work on the theory and practice of agent based systems and focus new work on supporting agent teams with rich negotiation languages and strategies. The DARPA KSE has given rise to a model for developing intelligent cooperating agents based on ontologies, content languages and agent communication languages. Although this approach and its associated components are being adopted as a standard for agent communication by both the research community (KQML, KIF) and by industry (FIPA), it requires significant extensions to support the requirements of the pervasive computing scenario outlined above. Some of the issues to be addressed include:

- Developing a language for defining higher-level protocols for peer-to-peer communication. This new language will build on existing network protocol formalisms to define protocols and associated agent behavior, providing a powerful yet intuitive and pragmatic language with sound theoretical foundation for formal analysis. The unit of communication will be the conversation, which involves not only the ordering and interdependency among exchanged messages, but also proper agent behavior such as actions, constraints and conditions associated with actions, as well as alternative actions and tradeoffs.

- Developing a rich set of negotiation primitives able to support various types of negotiations involving different agents (cooperative, competitive, or a mix of both) and in different fashion (peer-to-peer or multi- party). These primitives will be based on the syntax and semantics of the FIPA Agent Communication Language, ensuring their consistency with emerging software standards.
- Developing prototype communication facilitators and brokers that learn the capabilities and interests of other agents through several active strategies. These include monitoring the conversations, both successful and unsuccessful between agents, performing generalization and specialization reasoning to predict future behavior, capturing uncertainty of agent capabilities by neural networks, fuzzy logic, and probabilistic learning, and initiating experiments to refine its meta data about the interests and capabilities of key agents. In our earlier work, we have shown how a network of such broker type agents can form a recommender system to select appropriate software for scientific computing type tasks[**JRH98**]. In other words, the system accepts a high level definition of the task from the user, and chooses appropriate software components needed to solve the task.

Component oriented support for computational tasks: As mentioned earlier, the individual components (software as well as hardware) that make up the networked computational system can be viewed as agents. We are developing a runtime environment based on a MultiAgent Systems approach that permits mobile computers to access not just information, but also computational resources, across the wireless network.

Unlike some recently proposed software component based systems, we will incorporate both software and hardware components. Software components will be mobile (in the sense of code migration) as well as static, and the hardware components will be adaptive (in the sense of the DARPA ACS program), mobile (in the sense of physical movement) as well as fixed. The runtime environment will accept from the end user a high level description of the task as well as some performance constraints and use a recommender system to automatically and "on-the-fly" map this to the appropriate set of available software components needed for the task and hardware components best suited to run them. The synthesis process will be a result of a sequence of active negotiations between the components and the runtime environment to compose the needed software. Computational tasks will be partitioned across a heterogeneous set of computational resources. This partitioning explicitly incorporates the asymmetry of the system which will have low resource systems such as handheld devices as well as high performance wired machines.

In order to achieve this, we will need to build a fundamental descriptive framework which can capture (i) the semantics of the services that each component agent in our system offers, and (ii) its own requirements in order to perform its declared tasks. We are developing an XML based scheme to encode these features in order to leverage the work on this developing standard, as well as an ontology suitable for our proposed application domains.

Initial prototypes: In our preliminary work, we have modified the Aglets mobile agent system so that each (mobile) agent posses an XML based description of the resources it needs (CPU type, software etc) to execute. Similarly, each aglets server (Tahiti) has an XML based description of the resources it has. When an agent is asked to migrate to a remote site (a dispatch, in Aglet terms), it instead sends a lightweight agent with its requirements to the remote site. This agent compares the resources existing at the remote platform with what the agent to be migrated would need, and then returns back. It then either tells the agent to migrate (if there are adequate resources at the remote end), or stops the migration.

We have also shown how a proxy agent based approach can be used by a mobile client to access remote computational resources[**Tri99**]. Our preliminary system deals with the task of landmine detection and remediation. In particular, we envisage that soldiers seeking to detect landmines will be equipped with Sensors (chemical, Ground Penetrating Radar etc.) connected to laptop type wearable computers which will be wirelessly hooked to one another and to nearby truck mounted high end machines. The novel algorithms being developed to detect landmines typically need more computational power than can be expected from the computers that the soldiers carry. Our system allows the mobile client to specify a task (such as detect mines) to the proxy agent and send it associated data. The proxy then locates a remote server providing the appropriate functionality and sends it the data. The server returns the results to the proxy, which then sends it back to the mobile client. Note that the proxy agent functioned as a broker here – the mobile client did not need to know where to find the service. Of course, the proxy also provided services such as compressing the data stream on the wireless side etc.

4. Discussion

As we approach the turn of the century, mobile devices and wireless data networks are becoming more powerful and affordable. This has lead to the growing importance of mobile access and data management. Mobile technology also seeks to expand the scope of networked computing infrastructure and allow mobile clients to participate in networked applications.

However, existing information systems and distributed applications often make inherent assumptions about the connectivity and topology of the underlying networks, as well as the capability of client devices, that are not valid in mobile environments. These assumptions, which are well understood by the research community, present significant difficulty in meeting nomadic users' expectations of having applications operate seamlessly across wireless and wired networks. While there has been work done to alleviate these problems, it remains nascent. Moreover, developments in ad-hoc networks tend to undermine the proxy oriented model which underlies much of this work.

With rapid advances in wireless communication and networking, and the anticipated deployment of 3G/4G wireless systems, it is our opinion that mobile access and data management will become an increasingly important research area. In this paper, we have discussed how an agent oriented approach can help design solutions. We have illustrated this using existing proxy based approaches to mobile web access, and shown how agents also provide the framework for end–end approaches. We have also outlined our ongoing work in creating a framework and runtime environment, using multiagent approaches, to support mobile information and computation access in next generation/ad-hoc wireless environments.

References

[BJA98] Harini Bharadvaj, A. Joshi, and Sansanee Auephanwiriyakyl, *An active transcoding proxy to support mobile web access*, Proc. IEEE Sumposium on Reliable Distributed Systems, October 1998.

[BKC+98] E.A. Brewer, R.H. Katz, Y. Chawathe, A. Fox, S.D. Gribble, T. Hodes, G. Nguyen, T. Henderson, E. Amir, H. Balakrishnan, A. Fox, V. Padmanabhan, and S. Seshan, *A network architecture for heterogeneous mobile computing*, IEEE Personal Communications Magazine **5** (1998), no. 5, 8–24.

[BMMM96] C. Brooks, M. S. Mazer, S. Meeks, and J. Miller, *Application-specific proxy servers as http stream transducers*, Proc. WWW-4, Boston, http://www.w3.org/pub/Conferences/WWW4/Papers/56Application-Specific, May 1996.

[FB96] A. Fox and E. A. Brewer, *Reducing www latency and bandwidth requirements by real-time distillations*, Proc. Fifth International World Wide Web Conference, May 1996.

[FGG+98] A. Fox, I. Goldberg, S.D. Gribble, D.C. Lee, A. Polito, and E.A. Brewer, *Experience with top gun wingman: A proxy-based graphical web browser for the usr palmpilot*, Proc. IFIP International Conference on Distributed Systems Platforms and Open Distributed Processing (Middleware '98), 1998.

[FZ94] G. Forman and J. Zahorjan, *The challenges of mobile computing*, IEEE Computer **27** (1994), 38–47.

[JdT+95] A. D. Joseph, A. F. deLespinasse, J. A. Tauber, D. K. Gifford, and M. F. Kaashoek, *Rover: A Toolkit for Mobile Information Access*, Proc. 15th Symposium on Operating Systems Principles, ACM, December 1995.

[JK98] Anupam Joshi and R. Krishnapuram, *Robust fuzzy clustering methods to support web mining*, Proc. ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery (S. Chaudhuri and U. Dayal, eds.), June 1998.

[Jos99] Anupam Joshi, *On proxy agents, mobility, and web access*, Tech. report, CSEE Department, UMBC, 1999.

[JRH98] A Joshi, N. Ramakrishnan, and E. N. Houstis, *MultiAgent Support of Networked Scientific Computing*, IEEE Internet Computing **2** (1998), no. 3, 69–83.

[JWH97] A. Joshi, S. Weerawarana, and E. N. Houstis, *Disconnected Browsing of Distributed Information*, Proc. Seventh IEEE Intl. Workshop on Research Issues in Data Engineering, IEEE, April 1997, pp. 101–108.

[JWM+96] A. Joshi, R. Weerasinghe, S. P. McDermott, B. K. Tan, G. Benhardt, and S.Weerawarna, *Mowser: Mobile platforms and web browsers*, Bulletin of the IEEE Technical Committee on Operating Systems and Application Environments **8** (1996), no. 1.

[Kat94] R. Katz, *Adaptation and Mobility in Wireless Information Systems*, IEEE Personal Communications **1** (1994), no. 1, 6–17.

[KKW+96] R. Kavasseri, T. Keating, M. Wittman, A. Joshi, and S. Weerawarana, *Web Intelligent Query - Disconnected Web Browsing using Cooperative Techniques*, Proc. 1st. IFCIS Intl. Conf. on Cooperative Information Systems, IEEE, IEEE Press, 1996, pp. 167–174.

[NKJ99] O. Nasraoui, R. Krishnapuram, and A. Joshi, *Mining web access logs using a fuzzy relational clustering algorithm based on a robust estimator*, to appear in Proc. WWW8, August 1999.

[NSN+] B. D. Noble, M. Satyanarayanan, D. Narayanan, J. E. Tilton, J. Flinn, and K. R.Walker, *Agile application-aware adaptation for mobility*, Proceedings of the 16th ACM Symposium on Operating System Principles.

[PE98] M. Perkowitz and O. Etzioni, *Adaptive web sites: Automatically synthesizing web pages*, Proc. National Conference of the American Association of Artificial Intelligence (AAAI), 1998.

[SSK97] S. Seshan, M. Stemm, and R. Katz, *Spand: Shared passive network performance discovery*, Proc. 1st Usenix Symposium on Internet Technologies and Systems (USITS '97, 1997.

[Tri99] Sunil Tripathy, *Distributed mobile system for landmine detection*, Masters Project Report, University of Missouri-Columbia, 1999.

[VYZ+99] J. Vass, J. Yao, S. Zhuang, A. Joshi, and X. Zhuang, *Multimedia information access in wireless environments*, Tech. Report TR-9901, CECS Department, University of Missouri, 1999.

[ZXH98] O.R. Zaine, M. Xin, and J. Han, *Discovering web access patterns and trends by applying olap and data mining technology on web logs*, Proc. Advances in Digital Libraries Conf. (ADL'98), April 1998, pp. 19–29.

Department of Computer Science and Electrical Engineering, University of Maryland Baltimore County, Baltimore, MD 21250

E-mail address: joshi@cs.umbc.edu