

# Economic Markets as a Means of Open Mobile-Agent Systems

Jonathan Bredin, David Kotz, and Daniela Rus

Department of Computer Science

Dartmouth College

Hanover, NH 03755

{jonathan, dfk, rus}@cs.dartmouth.edu

May 1, 1999

## Abstract

Mobile-agent systems have gained popularity in use because they ease the application design process by giving software engineers greater flexibility. Although the value of any network is dependent on both the number of users and the number of sites participating in the network, there is little motivation for systems to donate resources to arbitrary agents. We propose to remedy the problem by imposing an economic market on mobile-agent systems where agents purchase resources from host sites and sell services to users and other agents. Host sites accumulate revenues, which are distributed to users to be used to launch more agents. We argue for the use of markets to regulate mobile-agent systems and discuss open issues in implementing market-based mobile-agent systems.

## 1 Introduction

One of the more recent items in a network programmer's tool box is code mobility. The technique is becoming more common in applications programming, network management [BPW98], video conferencing [BPR98], software distribution and installation, unreliable networked weather forecasting [Joh98], and client-server networking alternatives [Mul98].

Mobility allows the programmer to easily distribute resource usage throughout the network over time. Resource contention can be mitigated by relocating execution to less utilized machines on the network. Additionally, since mobile agents are autonomous, they may schedule their own computation at a later time at a remote host, avoiding times of congestion.

We examine some restrictions typically present in mobile-agent systems. Specifically, few mobile-agent applications implement communication with agents from other applications or submitted by competing users. This shortfall might be surprising considering that agent technology is often advertised as a unifying design paradigm nurturing both cooperation and competition.

Many of the restrictions faced by usable mobile-agent systems stem from the distributed nature of mobility: because decisions are made throughout the network, coordination becomes more difficult. Human societies have implemented economic markets as a solution for distributed control. We believe that the same solution can be applied to create societies of mobile agents.

We propose that mobile agents buy computational resources from their hosts using a scarce verifiable currency. The agents' priority would be governed by limited endowments. The danger of denial-of-service attacks is bounded by agents' expenditure. Systems can buy outside computational resources to flexibly expand their computing base. Conversely, idle resources may be sold to users from other sites. Finally, a fair price system provides valuable information that allows agents to autonomously and efficiently balance network load.

We conclude by establishing goals necessary to implement computational markets in the context of mobile-agent systems: a low-overhead verifiable currency system, efficient incentive-compatible pricing mechanisms, a set of standards conveying the rules and bases of such pricing mechanisms to allow rational planning by agents, and both demand-based and reservation-based consumption.

## 2 Motivation

As mobility becomes more commonly used in network programming, we observe that, so far, applications operate in a closed environment. The range of sites to which an agent can jump is severely restricted and it is common for all agents in the system to only represent the interest of a single user or cooperating group of users. Additionally, typically agents only communicate with other agents of the same application.

The number of sites to which an agent can jump is limited. Why should a host allow any agents to visit at all? If we ask the analog question in web browsing, in a large number of situations there is a clear advantage for web servers to supply their resource (information) to arbitrary clients. This information dissemination is generally done to boost the reputation of the host site's owners, clients, or products.

Computational resources exported by mobile-agent hosts are much more difficult to control. The host site generally has no assurance that an arbitrary agent's actions will have any beneficial effects. There is little incentive for a host system to provide resources to an arbitrary agent. Not only is there the additional congestion incurred through normal mobile agent use, but there are additional risks from denial-of-service attacks and other irresponsible uses of resources.

A mobile-agent system's value is not only dependent on the number of participating host sites, but also on the the number of participating users and agents. Most non-research mobile-agent applications assume that agents are all issued by cooperating owners. Typically, only one entity issues mobile agents for a task. This entity may be represented by many users working for a university or company, but the interests of the agents are generally complementary and it is in agents' best interest to cooperate. Essentially, agents in such situations can be viewed as having a common owner.

Even if all agents share a common goal, their use distributes decision-making processes throughout the network. To perform efficiently, agents must be able to coordinate and assess

the impact of their actions. Ideally, the medium for this information exchange should be fast and incur minimal overhead.

We see few real-world examples of agents coordinating with agents involved in other applications. In stark contrast, consider the World Wide Web. A user's browser sends requests to thousands of host sites to retrieve information. Much of the information retrieval is more than simply examining an HTML file. Often browsers exchange cookies with servers, negotiate security protocols, retrieve dynamically produced web pages, download applets, and forward retrieved information to other applications to be processed. A typical web browsing-session can involve the use of several dozen application programs. We have the same goal for mobile-agent systems.

### 3 Markets

To overcome the limitations currently experienced by mobile agents, we propose to establish an economic market for computational resources and services. Mobile agents arriving at a host site will purchase the resources necessary to complete their task. These resources could include access to the CPU, network and disk interface, data storage, and databases. Presumably, agents are providing valuable services to their users. It is possible that other agents would also benefit from the service, so agents could sell their services to users and agents. Eventually, currency will accumulate at host sites. Revenues are then distributed to local users who in turn disperse their income to their agents completing the cycle.

The currency used in computational markets does not necessarily have to be tied to legal tender currency. If the currency is exchangeable for real dollars, however, system administrators can essentially export and import their computational resources. Access to underutilized resources may be sold to mobile agents (resource export). If local resource contention is high, then users may launch agents to carry on computation elsewhere (resource import).

Because currency is scarce, budgets are finite, and all resource consumption is tied to expenditure, agents' lifetimes are limited. The extent of a denial-of-service attack, wanton consumption done with the intention of excluding other users from the resource, is limited. Given that resource pricing is fair, hosts will be happy to entertain a denial-of-service attack to maximize revenues. An efficient pricing policy will ensure that demand and price are positively correlated and make denial-of-service attacks extremely expensive operations, deterring offenders.

Price, the same mechanism that discourages wasteful consumption, serves as a simple metric of resource contention and site congestion. Price advertisement provides a simple means of agent coordination as follows. Revenue maximizing hosts will charge what the market will bear. High prices due to congestion give agents incentive to distribute themselves evenly throughout the network or defer execution to a less congested time. Thus a pricing system effectively implements both temporal and spatial load balancing.

The idea of selling computational resources to mobile programs is not new. We discuss a few recent implementations next. POPCORN [RN98] is a system that uses markets to distribute "computelets" through the network to take advantage of idle CPUs. The approach is intended for parallel programs where interaction among threads is limited.

The Geneva Messengers project [Tsc97] applies market ideas to allocate CPU usage and memory to visiting *messengers*, lightweight mobile programs implemented in a Postscript-like language. Host sites heuristically set prices by examining the amount of resources requested by the present messengers.

In Telescript [Whi96] agents carry *permits* to access specific host-site resources. As a permit is used, hosts trust each other to diminish the permit's power. The result is that agents' lifetimes are limited. This use of permits can be viewed as a limited form of a market. Host sites distribute permits for each resource to be controlled, though a permit for one resource is not easily convertible to another. A more general form of this mechanism would be to have a universal permit, currency, that could be exchanged for other permits.

An extreme point of view is taken in MarketNet [YDFH98] where currency-resource exchange is the exclusive form of security. Different levels of security access are sold to users. Sites may discount access to certain populations by setting a lower price in a separate currency. Presumably this new currency is unusable by users outside the group.

## 4 The Challenge

There are many hurdles to cross on the way to implementing an effective market system. Most obviously, there must be some means of exchange such as a secure currency system. The market should be structured in such a way to reward honest behavior, to facilitate planning. Finally, this structure should be well known to all entities participating in the market.

### 4.1 Accounting

All markets rely on a secure means of exchange. Without this, there is no incentive for participants to cooperate. Currency should be scarce to reinforce its value. A prerequisite to this is that currency may only be spent once, i.e. a buyer may not use the same note for more than one purchase. Ensuring this is the crux of any monetary system and can carry with it significant overhead. Electronic currency in the context of mobile-agent systems has one particular caveat: an agent's money is essentially just data, data to which the host potentially has access.

There are several micro-currencies designed to minimize the cost of transaction [GMA<sup>+</sup>96, PHS98]. If the cost of currency exchange is still too large, there are other options to take. Hosts can establish a local account for each agent or its owner. Deposits into the account are periodically made with some secure payment method. Agents then withdraw from the local accounts as they compute, trusting the host site to correctly decrement the account balance. If the deposits are small or the account is known to be used over a long period of time, then there is less incentive for hosts to overdraw the local account since the payoff for cheating once is lower than conducting further honest business.

The other option is to scale the level of resource accounting. Ideally, agents would be charged for every action they take including a precise count of the instructions executed or even the number of processor cycles used; bits sent through the network interfaces; words/milliseconds of storage; etc. Such monitoring is most likely impractical, however, due

to the cost of precise measurement. At the other extreme, agents may pay a fixed fee to execute any set of instructions. Obviously, this is also inefficient in that agents will rarely use the exact level of service for which they pay.

A happy medium between the two extremes must be found. Possibly, profiling existing applications using mobile-agent technology will give some intuition on the appropriate level of control. It is quite possible that the sorts of applications that can take advantage of mobility have similar resource requirements and one can tailor an allocation policy accommodating the majority of applications.

Extending this strategy could allow hosts to offer one of several resource packages and an appropriate billing plan to agents. Each package-accounting pair would have advantages to different groups of applications depending on preferences and resource demands.

## 4.2 Policy Design

There are two dynamics that drive the construction of economic policy. One is the participants' goal to maximize utility, while the other is the system's engineers need to enforce an equitable allocation. These two ideals often conflict under poorly designed policies. A simple example is when the market is allowed to create monopoly. Here a monopolist will act to maximize revenue at the expense of customers. The lesson learned is that either monopolies should be regulated or that conditions allowing the existence of monopolies should not exist.

It is the responsibility of the designer of the economic system to provide an environment in which both buyers and sellers are willing to participate. Frequently, this is facilitated by constructing mechanisms where the parties choose those actions that express their true intentions, i.e, incentive compatible mechanisms. This open honesty mitigates the cost of planning and decision making.

A well used example of an incentive-compatible policy is the sealed-bid second-price auction [Vic61] where potential buyers simultaneously submit a single bid for a good. The auctioneer chooses the winner to be the participant who submitted the highest bid, but the winner pays the highest losing bid. Generally, the optimal strategy is for a buyer to submit a bid equal to the buyer's valuation of the good.

A second issue in policy design: to enforce load balancing, a resource management policy should provide a strong correlation between the contention for a resource and its price. Possibly, this might be accomplished through setting a price that increases as the quantity consumed rises. Alternatively, an auction could be held for the resource and the resource owner could let the buyers compete to set the price.

Negotiation of price is not sufficient for market-based resource allocation. The resource management policy must also take into account resource consumption scheduling. The system designer must decide whether to entertain reservations, consumption on demand, or some combination of the two. Users and agents would likely be willing to pay more for service guaranteed by reservations, but hosts might be able to sell larger quantities of resources on a demand based scheduler. This is an interesting and valuable issue to study as it effects planning on both the host and client sides and will likely deal with computationally complex issues. There is the additional dimension that users will be willing to pay amounts proportional to the quality of service.

### 4.3 Standardization

Finally, any effective policy will require that all participants are aware of the guidelines regulating the system. Sites will have to find some way of publishing their resource pricing and allocation policies for their potential users. For agents to be aware of rules either requires a single standard or some protocol for expressing market parameters.

Again, with this issue, there should be moderation. The point of an “agent” is to shield the user from all the intricacies of computation by providing an abstraction. The user should be aware of the service quality they receive as well as a general level of congestion of the requests to their agents, but it is the agents’ responsibility to efficiently perform the task.

## 5 Conclusion

We believe that markets are the proper tools to provide an open mobile-agent system. They enforce an additional level of security and give incentive for agents to autonomously balance the computational load across the network. Allowing the currency used to buy computational resources to be exchanged for legal tender allows system administrators to temporarily expand their domain by importing resources as well as capitalize on idle resources by exporting them.

Implementation of a mobile-agent computational market will require further research in electronic currency exchange to minimize the overhead of currency validation. Furthermore, careful decisions must be made on the part of market designers and host-site owners to equitably distribute resources among mobile agents and local users while attempting to maximize revenue. Finally, regardless of the resource-allocation policy, for agents (or their programmers) to plan appropriately, they must be able to detect which policy is being used. Policy discovery will likely require either a single standard or a language to describe market protocols.

Establishing markets will achieve distributed decision making in mobile-agent systems. We feel that markets are natural solutions to mobile-agent coordination and resource control and will eventually allow mobile agents to be used in an open multi-application environment, though much work remains to be done to implement a working system.

## Acknowledgments

This work is supported in part by the Navy and Air Force under contracts ONR N00014-95-1-1204 and MURI F49620-97-1-0382, Rome Labs under contract F30602-98-C-0006, and DARPA under contract F30602-98-2-0107.

## References

- [BPR98] Mario Baldi, Gian Peitro Picco, and Fulvio Risso. Designing a videoconference system for active networks. In *In Proceedings of the Second International Workshop, Mobile Agents '98*, pages 273–284, Stuttgart, Germany, September 1998.
- [BPW98] Andrzej Bieszczad, Bernard Pagurek, and Tony White. Mobile agents for network management. *IEEE Communications Surveys*, September 1998.
- [GMA<sup>+</sup>96] Steve Glassman, Mark Manasse, Martín Abadi, Paul Gauthier, and Patrick Sobalvarro. The Millicent protocol for inexpensive electronic commerce. *World Wide Web Journal*, 1(1), Winter 1996. Also in Fourth International World Wide Web Conference, December 1995.
- [Joh98] Dag Johansen. Mobile agent applicability. In *In Proceedings of the Second International Workshop, Mobile Agents '98*, pages 80–98, Stuttgart, Germany, 1998.
- [Mul98] Tomasz Muldner. Mobile computing at acadia university. Dartmouth College Computer Science Colloquia, 1998. slides at <http://evilqueen.acadiau.ca/presentations/mobileagents.ppt>.
- [PHS98] Tomi Poutanene, Heather Hinton, and Michael Stumm. NetCents: A lightweight protocol for secure micropayments. In *USENIX Workshop on Electronic Commerce*, pages 25–36. USENIX Association, September 1998.
- [RN98] Ori Regev and Noam Nisan. The POPCORN market— an online market for computational resources. In *Proceedings of the First International Conference on Information and Computation Economies*, pages 148–157, Charleston, SC, October 1998. ACM Press.
- [Tsc97] Christian F. Tschudin. Open resource allocation for mobile code. In *In Proceedings of The First Workshop on Mobile Agents*, Berlin, April 1997.
- [Vic61] William Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *Journal of Finance*, 16:8–37, 1961.
- [Whi96] James E. White. Telescript technology: Mobile agents. General Magic White Paper, 1996.
- [YDFH98] Y. Yemini, A. Dailianas, D. Florissi, and G. Huberman. MarketNet: Market-based protection of information systems. In *Proceedings of the First International Conference on Information and Computation Economies*, pages 181–190, Charleston, SC, October 1998. ACM Press.