

# Motivating Participation in Peer to Peer Communities

Julita Vassileva

Computer Science Department  
University of Saskatchewan  
57 Campus Drive  
Saskatoon, Saskatchewan, S7N 5A9 Canada  
jiv@cs.usask.ca

**Abstract.** One of the most important prerequisites for the success of a peer to peer system is the availability of participants willing to provide resources (files, computational cycles, time and effort to the community). Free riders may not do any harm in file-sharing peer to peer applications, like NAPSTER, because of the nature of electronic data. It can be reproduced at no cost; downloading a copy does not take anything away from the common resources, but instead creates new resource. However, free riders they can be destructive in applications where there are costs associated with the resources shared. The paper argues that providing motivation or some kind of incentives for users to participate is very important. It describes various methods to motivate different kinds of users and describes a design of a peer to peer system called Comutella, which is being developed currently to support file and service (help, advice) sharing in research groups and groups of learners.

## 1. Introduction

Peer to peer systems have become increasingly popular in recent years. Applications like Napster, KaZaA the myriad of Gnutella-based file-sharing software or SETI@home have given publicity of peer to peer (P2P) computing. P2P is considered "invincible" because of the decentralization of control (e.g. Gnutella), "unbeatable" in terms of performance (e.g. Seti@home), and moreover, enthusiastically supported by all the participating users. One of the most important prerequisites for the success of a P2P system is the availability of participants willing to provide resources (files, computational cycles, time and effort) to the community. Studies [2] have shown that P2P systems are prone to be overwhelmed by "free riders", i.e. people who do not contribute resources, but only consume. While this has given rise to some pessimistic expectations about the future of P2P computing, others [14] have argued that free riders are not be harmful in file-sharing P2P applications, like NAPSTER or KaZaA, because of the nature of electronic data. It can be reproduced at no cost; downloading a copy does not take anything away from the common resources, but instead creates a new resource that can be shared. However, the problem remains in P2P applications where there are costs associated with the resources shared, e.g. compute cycles, network bandwidth, or human time and effort. In such systems, free riders can be a menace, and they can bring down the



functionality for other users (and increasingly, for themselves) by their consumption of resources, without contributing anything.

A P2P application, called COMUTELLA (Community Gnutella) is currently being implemented at the Mobile and Ubiquitous Computing Lab at the University of Saskatchewan. The system will enable research or study groups of students to collaborate and share resources, e.g. to exchange both files (e.g. research papers) and services (e.g. help each other). Each user installs a servent (a small application which is both a client and a server) that allows the user to search for resource or services and to offer resources and services to community. The servents communicate with each other using the Gnutella protocol. The nature of resources that are shared requires active involvement of the users (e.g. finding relevant research papers, saving and organizing them, enabling folder sharing, or engaging in conversation with other user(s) to answer questions, provide advice etc.). Therefore, it is important to ensure user participation and a certain level of contribution to the community, so that a reliable service is provided.

This paper does not report on a piece of finished work; it rather describes the design of a system that is being developed right now. The remainder of the paper is organized as follows: section 2 discusses various methods of motivating users to participate actively in a P2P community, section 3 presents the user modelling and adaptation that needs to be done by the servent, section 4 discusses methods for motivating users to participate and section 5 concludes the paper.

## 2. Motivating users to participate

Motivation of users to participate in the community is a crucial factor for the success for a P2P system. Our experience with I-Help, a P2P system for help, that has been used for over 2 years in the University of Saskatchewan showed that if the deployment of the system lacks a "critical mass" of active users, it will never be able to take off [5]. Indeed, if each user logs in, asks for help and after finding (or more likely not finding it) logs out, there would be very few people simultaneously on-line. That means that the likelihood of finding a helper, competent about a given question, would be very low. After trying unsuccessfully to find help a couple of times, people stop trying. In this way the pool of users shrinks quickly in a downwards spiral and the system becomes disfunctional. On the contrary, if users stay logged in, there is a higher probability that when somebody asks a question there would be somebody on-line who can answer. A user who received help is more likely to perceive the system as useful and remain logged in (in case she needs it again), available, when her expertise is required. In this way, the amount of participating users, as well as the value of the system increases in a snowball effect.

It is important that the users perceive the system as useful and are willing to participate. We believe that this is true for file sharing P2P systems as well as for systems allowing sharing services or computing tasks. The size of the community of users defines the level of usefulness, or the value of the system, and the other way around, the value of the system defines the number of users. This "feedback loop" can develop by itself, as happened in file-sharing systems like Napster and KaZaA.



However, in other applications, like our peer-help system I-Help, it doesn't just happen. Our experience showed, that incentives are needed for the users in the beginning. This can be achieved either by providing an ample amount of resources in the system, or some other incentives. Initial investment of resources, for example, hiring a knowledgeable person, teaching assistant or lecturer, to be constantly on-line and to answer immediately any question, is very helpful. It creates the impression in the students that the system is useful and they get used to asking their questions there, instead of phoning somebody. Once they get into the habit of staying on line, the knowledgeable person can withdraw; the students start getting matched with each other and helping each other. An alternative is to provide an incentive by giving some participation marks for using the system. Even though this reward is external to the system, it can draw a significant amount of users and ensure the "critical mass" necessary for the system to function.

Obviously, these two ways of motivating participation in the P2P system depend on the nature of the application and the context of use (in this case formal University setting). Generally, motivation strategies seem to depend on how much participation or effort is desirable from the users and the value that the system provides to them.

There are several levels of user cooperative participation in a P2P system. They are characterized with decreasing degree of active user involvement (activeness):

- *create service*: creating new resources or services and offering them to the community,
- *allow service*: providing to the community disk space to store files for downloads or computing resources to enable a service that has been created by another participant in the community
- *facilitate search*: providing a list of relationships to other users to facilitate their search of files or services. This level of cooperativeness is possible if the servers model the "good" relationships with peers, as we propose in the next section.
- *allow communication*: forwarding ping-pong, query and hit messages, i.e. actively participating in the protocol of the network, thus providing the "glue" that holds the network together [2] thus facilitating the peer infrastructure.
- *uncooperative free-rider*: downloading files or utilizing services when needed, but going off line immediately afterwards.

The "create service" level usually includes "allow service", "facilitate search" and "allow communication", i.e. it describes the most socially cooperative type of user behaviour.

The more typical level is "allow service", describing a user who contributes passively to the community, by providing her resources and relationships, as well as the functionality of her server to enhance the infrastructure of the community, but does not actively bring new resources or services into the system. As shown in [2], in Gnutella only a tiny minority falls into this category - 5% of the users is responsible for sharing over 70% of the files.

According to [2], the majority of users (66%) fall into the category, "allow communication" - they participate in the network infrastructure and therefore can be detected and taken into account. Unfortunately, there is no way to know at any moment how many users are "uncooperative free riders" or "creators of service" due to the lack of history in Gnutella and the anonymity, which does not allow to identify who first introduced a file into the system.



According to [11], more than 80% of the users of Mojo Nation were "1-time, 1-hour" users, and of the remaining users a significant part were "1 time, less than 24 hour" users. We observed a similar behaviour of users in our I-Help system in certain classes, where the instructors failed to motivate a "critical mass" of active users in the beginning, when most of the users log-in just to try the system [5].

All popular file sharing P2P systems, like NAPSTER, Morpheus /KaZaA try to ensure at least the "allow service" and "allow communication" levels of cooperation. Usually "allow service" is ensured by a default setting in the servent, which commands the downloaded files to be saved in the "shared folder", so that they can be found by other servents. The "allow communication" level is achieved by making it harder to quit the servent (even if clicking on the close window button, which normally quits an application, a KaZaA servent will remain active until it is quit once again explicitly by selecting the "Quit" option on the right-click menu). These settings can be changed by the user, e.g. the downloaded files can be saved in a file different from the "Shared Folder", but it requires more knowledge and active involvement from the user, which is a kind of punishment for uncooperative users who wish to be only free riders and not to contribute to the community in any way.

There are four principle ways of motivating users to participate in a community:

- by trying to influence the user's feelings (of guilt, of belonging or owing to the group) to stimulate her altruism towards the community,
- by rewarding the user with visibility / reputation in the group depending on hisher contribution,
- by allowing the user to develop relationships with other users in the community (one would do a favor to a friend, which one might not want to do for anonymous unknown people), and
- by providing an economical model which ensures incentive for user contribution, (e.g. better quality of service, priority in the queues).

It is likely that choosing the appropriate way of motivation depends:

- on the personality of the user, and
- on the nature or the user's interest in the area and the group (community).

Thus, the same user can be altruistic in one group, motivated by reputation in another group and by economic rewards in a third group.

The conclusion is that one needs to know a lot about the user to be able to persuade or motivate her to participate, about the user's interest in the community, about people with similar interests, who might become potential "friends" of the user, and finally about the user herself, in term of how selfish or altruistic she is. User modeling provides means to capture such features of users.

### 3. User Modelling

To create user groups based on interest, the servant needs to understand the character and the interests of the user and facilitate finding and maintaining relationships with users with similar interests. The approach described below is tied to the COMUTELLA application area - exchanging academic papers and on-line help/discussion on various topics. However, it can be adapted for other similar areas,



where users are looking for resources and services that can be classified according to their semantics.

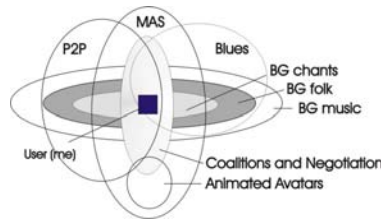
### 3.1. User Model Representation

The user model contains four different parts:

- a model of the user's personality
- a model of the user's interests,
- a model of the user's relationships.

The model of user personality can be simply a number denoting the level of the user's selfishness, varying from -1 (selfish) to +1 (altruistic).

The model of user interests is represented as a list of topics / areas in which the user is interested. An ontology representing typical search terms in a given semantic area allows clustering users into groups sharing similar interests. It is important to note that these user groups can overlap on various levels, e.g. one user can be a member of a group interested in Bulgarian folk music and in a group interested in blues. The same user can be also a member of a group interested in P2P computing and in a group interested in multi-agent systems. However, she may be a member of a sub-group of the multi-agent systems group interested in agent negotiation and coalition formation and not be a member of a group interested in animated avatar agents (see Fig. 1).



**Fig. 1:** User interest groups.

The model of user interests is organized hierarchically as an overlay over the domain ontology. Sub-areas in which the user has shown interest by issuing queries are represented, along with a value that indicates the strength of user interest in the area and a time stamp showing when the user made the last query in this area. The user's strength of interest in an area is calculated based on how many times the user has searched in this area, and how recently she has been searching in this area. The user's interest in areas that are more general (higher in the ontology hierarchy) than the current area of search are also impacted, but in a much weaker way.

It is clear that in order to apply this approach, a necessary condition is that all servants use a compatible representation of a domain ontology or ontology of services. There are various tools to developing such ontologies, e.g. DAML-S [3].

The model of the user's relationships includes the users with whom the user has interacted frequently, i.e. from whom the user has downloaded files frequently and also users, who have downloaded files frequently from the user. These relationships are represented in a list where each relationship contains the unique ID of the other



user, the search area in the context of which the users have interacted and two numbers representing the strength and the balance of the relationship. The strength is a subjective factor reflecting how satisfied the user was with the interaction, i.e. if she kept the downloaded file or deleted it, if she used the file frequently. The balance of the relationship denotes the direction of services / files, i.e. who of the users predominantly uses and who offers resources / services. The next section explains how these three representations are created and updated.

### 3.2. Creating and updating the user model

A simple method - reinforcement learning - is selected as a basis for maintaining all parts of the user model.

#### 3.2.1. Modelling user personality

The level of user's selfishness/altruism is updated based on user actions that have impact on the community, e.g. providing new files to be shared, enabling file-sharing for folders on the user's hard disk, the preferences the user has chosen with respect to file-transfers from other people (see Fig.3). For example, if the user wants to bring a new resource (paper) in the system, she has to spend some effort to annotate the file w.r.t. category, so that it is searchable. If the user is willing to do this, the level of altruism of the user is increased. If the user interrupts a on-going transfer, this is a sign of a selfish behaviour, and the level of altruism is decreased. Other parameters considered in the computation of the personality characteristic of the user are the number of files shared by the user, the relative duration in which the user's servant is active, and the number of user actions that are deemed as uncooperative, such as removing downloaded files from the shared folder, or disallowing sharing of folders.

#### 3.2.2. Modelling user interests

Each servant keeps track of the areas entered by the user for search in the model of user interests. The areas reflect the ontology of the domain. In COMTELLA, these are a subset of the ACM set of subject categories. The strength of user interest  $S^a$  at time  $t$  in each sub-area  $a$  in the ontological heterarchy that is on the path leading to the sub-area related to the words of query is updated according to the reinforcement learning formula:

$$S^a(e_t, t) = i * S^a(e_{t-1}, t-1) + (1 - i) * e_t \quad (1)$$

where the new (at time  $t$ ) evidence of interest  $e_t \in [0, 1]$  is calculated as  $e_t = 1/d$ , where  $d = 1 + \text{the distance}$  between the level of the current search area in the ontology tree and the level of the area  $a$ .

The parameter  $i \in [0.5, 1]$  is an inflation rate used to model the fact that older experiences become less important over time  $t$ , while the most recent experience is the most relevant (since the user's preferences may change over time). It can be fixed at a given value, say 0.5, giving equal weights to old and new evidence. The parameter  $i$  can also be a variable, which depends on the time elapsed since the last



evidence of interest in this area, which allows capturing better the current tendency in user interests. An example is shown in Figure 2.

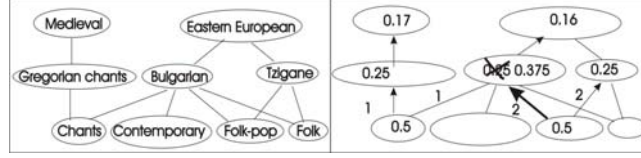


Fig. 2: Updating the model of user interests

### 3.2.3. Modelling user relationships

To model the relationships of the user, the server keeps track of the following:

- which servers respond with relatively many hits to the user's query,
- from which servers the user chooses to download files or to request service,
- the success of each download / service,
- who issues queries that result in hits in the list of resources of the user,
- who downloads files from the user,

The servers who have returned many hits along with those from whom the user has chosen to request service or download a file are entered in the user's list of "relationships" with an indication of the context of the particular area of interest (request) in which the relationship was created. The success of each download or service is used to update the strength of the relationship between the users using a formula similar to (1). Servers who are searching for files / services that are offered by the user and those who choose to download files or use the services offered by the user are added to the list of "relationships" of the user in the context of the particular area of interest depending on the query used for the search.

In addition to the relationship's strength and context, the server keeps track of the balance (reciprocity) of the relationship. The server of user X calculates the balance of its relationship with the server of user Y as:

$$B^{XY} = N^{X \leftarrow Y} - N^{Y \leftarrow X} \quad (2)$$

i.e. the difference between the number of times when the user X has downloaded files from Y and the number of times when user Y has downloaded files from X. If the balance is negative, the user X "owes" user Y.

The sum of the balances of all relationships of a user defines how much she has contributed to the community and how much she has consumed. Keeping a balance of each relationship allows maintaining a model of the user's contribution to individual users, to every interest group in which she participates and to the network as a whole. It is used, along with the model of the user's personality by the server to provide an individualized motivational interface for the user, to persuade her/him to contribute to the community.



#### 3.2.4. Sharing user models among servants

While not involved actively in search, servants can communicate with each other and learn about other users with the similar interests. This learning can take two forms:

- *Direct*: without explicit request from the user, the servant sends queries in a particular area to find out servants that have resources and enters them in the list of relationships of the user. The strength of relationship can be minimal in this case, calculated as in [8] by the percentage of the number of hits by this servant over the total number of hits. In this way the servant explores the network. Also the time of request and reply can be recorded, to capture the compatibility in time patterns of being on-line.
- *Indirect*: by requesting the list of their relationships in a given area from other servants, with whom the servant has strong relationships in this area. In this way the IDs of servants that have provided good files in a given area, have been frequently available and have provided a good service (e.g. did not interrupt the connection during download) are shared among the servants.

Through this collective learning, in a network which is not very dynamic, i.e. most of the servants are active at approximately the same time, ultimately, all communicating agents from one interest group would end up with the same list of relationships, which will lead to an implicit objective measure of quality / ranking of each servant within the group (i.e. reputation). This objective ranking will be contextualized, i.e. it will make sense only in the context of one interest group, since people behave differently in different communities. Of course, this will not prevent servants who have a high rank in one group to have a high rank in another group too.

There are a number of possible strategies concerning the requesting and interpretation of information received from others. For example, we assume that a servant will ask only the servants with the highest rank in its list of relationships about their relationships. However, it is possible to request data from all existing "acquaintances" and use the strength of relationship with each source to compute the strength of relationship in the new "acquaintance". This approach is similar to approaches for trust propagation among agents in multi-agent communities [13].

There are many open questions, for example, how information about a given servant coming along a chain of "acquaintances" should be interpreted. There are various possibilities: by averaging the strength values along the path, or by multiplying them. It is also necessary to define a policy for resolving conflicts between different two different chains of sources. Should two separate representations of the strength of relationship be kept by the servant: a subjective one based on the servant's own experience, and objective based on information received from other servants. How can these two representations be combined and when?

There are arguments against using global reputation measures. For example, if user X requests a service from user Y always at 1 a.m. and never gets anything from him, since Y is never on-line at that time, X's strength of relationship with Y would be 0. However, Y could be a very active member of the community, providing useful files and services at other times. The strength of relationship that X has with Y reflects not just the common interests, but also the compatibility between X's and Y's preferences in the time pattern of usage, and it can not be generalized without attaching a lot of context information. Just averaging the strength of relationship values of many people without considering the contextual information would not be appropriate. More



sophisticated techniques are necessary to retrieve information from appropriate servers, to interpret it in a context and purpose-dependent way. Distributed User Modelling [10] addresses some of these issues, but they are currently out of the scope of this application.

## 4. Motivating Participation in COMUTELLA

By modeling the user, the server can apply persuasion techniques through modifications of the interface or through rewarding the user with a better quality of service to attempt to influence the user's behaviour for the common good. Motivating users to participate is very similar to teaching them how to behave as good citizens. A basic principle of good teaching, leading back to Skinner, is to provide a plenty of positive feedback. It is important to reward users for good behaviour and not to give them the feeling that they are "punished" for bad behaviour (at least not in the beginning), since they may withdraw entirely from the system. Of course, negative feedback should be present too, in carefully selected doses depending on the user's level of participation.

### 4.1. Motivating altruistic users

Altruistically motivated users are devoted to a particular cause (e.g. finding extra-terrestrial intelligence, cancer research or genome sequencing). They are likely to be active participants on higher levels (allow or create service) in an interest group dedicated to the cause, like SETI@home. Influencing people to be altruistic for a given cause is a very difficult task; it requires a very detailed and broad model of user interests and of her acquaintances in the real world (who might be involved in a interest group with a certain altruistic purpose). This seems still beyond the powers of the current user models and existing captology [4] (persuasion) techniques.

A simpler way that could hopefully influence the user is trying to invoke a feeling of guilt for not contributing to a community from which the user has taken a lot of resources. This could be attempted by using subtle cues like running messages in the window frame (should never be obtrusive), or by a face or animal figure that changes its expression with the change in the owing balance of the user to the group (see section 3.2.3). We have developed a simple iconic avatar that represents the server (an eagle), which changes gradually to reflect the level of cooperativeness of the user towards a given community. This level is computed from the sum of balances of the user's relationships with the members of the community, and the user's personality characteristic from the user model. For each avatar there is a set of variants that differ in the level of friendliness of expression. Depending on the user's level of participation in the community, the avatar changes from a friendly sympathetic expression to an unfriendly and even vicious expression (see Fig.3). This is accompanied with a running message on the bottom of the window suggesting what the user can do to participate more actively in the community, depending on the current level of participation of the user. The idea is that, similar to Oscar Wilde's



"The Picture of Dorian Grey" [12], the user will be cued to reflect on her social behaviour and how she can possibly change it for the better.

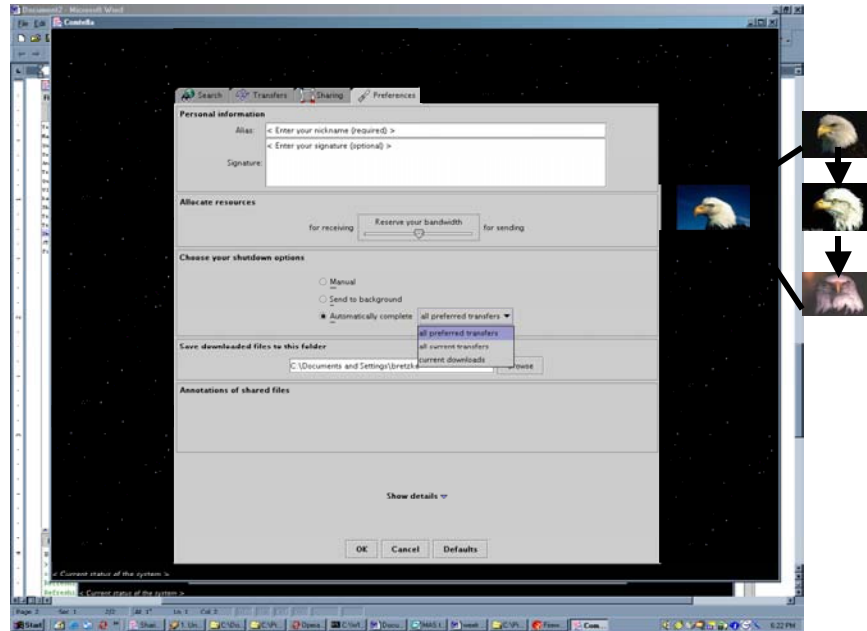


Fig 3: The motivational interface in COMUTELLA.

#### 4.2. Motivating reputation-aware users

Users motivated by social reputation are more likely to be active participants in groups where they already know some participants (even if by alias) and are known themselves. This impact on the user can be achieved through an appropriately designed interface of the server, that creates a global view of the group visualizing in an appropriate way the servers that contribute most. We are currently developing a background image of the server for this purpose. It is inspired by the idea of a night sky where servers are represented as stars varying in size and brightness (see Fig.3). The size denotes the amount of resources or services shared by the server and the brightness denotes the amount of relationships that the server has with other servers. The user can access an annotated version of the image, and by positioning the mouse on a star the user can see the name / alias of the server. The star representing the server of the user is indicated, so the user can compare her



significance in the community by the size and brightness. The image is refreshed periodically to reflect changes in the group.

### 4.3. Creating a feeling of community

Many people are motivated by the feeling that they belong to a community. They want to be known in the community, to find friends and people with similar tastes and values. While not always altruistic, these people tend to be altruistic in groups, which they perceive important for them (e.g. family, work-colleagues, neighbourhood). The forms of the altruistic behaviour and the level of contribution expected varies depending on the community.

More recently, approaches that try to exploit the social interactions between peers have been proposed. Local search strategies introduced in [1] use well-connected servents and have costs that scale sub-linearly with the size of the network. Ramanathan et al. [8] propose to dynamically determine the set of neighbors among those that have most related files (hits), thus indirectly defining a new neighborhood of each servent according to the current search. This reduces the number of query messages sent in the network by both reducing the number of servents to which a query is broadcasted and the TTL of each query). This approach provides a step in the right direction of proving mechanisms for servents in P2P systems to self-organize into groups.

In COMTELLA, friends of the user are treated differently by her servent. The servent ranks the requests coming from other servents depending on the balance and the importance of the relationship. In this way, in a download queue, priority or more bandwidth will be given to request from important servents to the user, or to servents, from who the user has downloaded often resources or whose services were often used. A servent X that "owes" to another servent Y can do it a "favour" by not decrementing the *time to live* (TTL, or the number of hops that the query can make) of a query sent by Y. In this way, the search horizon of a user who has contributed resources to users in the group increases.

Two users can be involved in relationships in several different contexts, indicating that they share interests in different areas. The relationships that the user maintains with other users in a given area of interest are sorted with respect to strength. From these, the top  $n$  ( $2 \leq n \leq 10$ ) servents that are active at a given moment are used as immediate neighbours to send queries when the user starts a search with keywords from this area. In this way, when the user searches for resource in a given area, her servent asks first the "friends" of the user in this area. They are more likely to have resources in the area, since they have been involved in exchange in this area in the past. In this way, a user with many "friends" in different areas is in a better position to find resources.

### 4.4. Rewarding participation

Several P2P systems (most prominently, Mojo-nation [11]) rely on a economic model to stimulate and reward participation. The basic assumption in the design of an



economic model is that the effort and time spent bring new resources or services in the community have inherent costs. To take these costs into account, the resources/services should be made tradable. Thus the payment in a virtual currency (e-cash, duke-dollars, or mojo) may motivate a user to create new resources / services.

It is possible to reward the accumulated currency outside of the P2P system. For example, in the context of our system I-Help [5, 9], participation can be rewarded either in real dollars (for paid teaching assistants) or in marks (for students). The choice of reward depends on the type of the interest group. In Mojo-nation currency could be cashed in gift certificates from real-world vendors, though this wasn't very successful. It seems, that users are generally unwilling to participate in P2P systems with "micro-payments" [14], since the cognitive load to make the decision whether to request a resource/service outweighs the benefit of the service at the tip of your fingers. However, the users do not need to be aware that such micro-payments are happening. They can be arranged between the servents and the only thing visible to the user would be an improved or deteriorating Quality of Service (QoS) in terms of speed for locating and downloading resources.

Introducing micro-payments for resources creates a mechanism to balance the supply and demand of resources/ services and allows taking into account the different quality of resources or services provided [6]. Exchanging the accumulated currency in better quality of service can be a significant motivator and relatively easy to implement, because everything remains within the system.

In COMUTELLA we explore providing better Quality of Service (QoS) by allowing "richer" servents to "buy" themselves a wider the search horizon by negotiating the TTL of each query. Also, depending on the amount of "mojo" it is willing to pay for the service, the servent can jump to the top of the queue for a given service that is in a great demand at the moment. In this way users who have contributed to the community and earned a lot of "mojo" are able to gain a better quality of service, without even knowing that there is economy and currency involved.

If the accumulated currency is "cashed" in better QoS, is important to ensure a gradual improvement or decrement in the QoS depending on the level of user contribution (level of accumulated currency) and always to maintain clear cues in the interface as to what is the reason for the increase / decrease in the QoS and what the user should do in order to improve it. The QoS should not deteriorate completely even for uncooperative free riders, since there is always a hope that they may become cooperative, if they find the right interest group.

An economic model in a P2P system, however, brings an overhead: it requires additional reasoning capabilities on behalf of the servents/ servents, reduces the anonymity in the system [7], and requires centralized components to be introduced to be responsible for the currency transactions (as in Mojonation).

## 5. Conclusions

We propose a variety of methods to motivate users to participate actively in a P2P community. These methods rely on a servent which maintains a user model, which



allows it to know an aspect of the personality of the user, the areas in which the user is interested and the friends or the most important relationships of the user in each area of interest. Such a servant will motivate the user in several ways: by interface cues targeted at provoking reflection in the user, by enforcing the user's feeling of being a part of a community through visualizing the standing of the users in the community, and by providing a better quality of service through a more informed search exploiting the social relationships that emerge between the user and her peers and implicitly creating user groups.

Unlike the approach proposed in [8], where the interest groups of users are highly dynamic and change rapidly to reflect the current search performed by the user, our approach relies on the assumption that users have long-term interests and are likely to search repeatedly in the same area at different times. Therefore, it makes sense to keep track of all "interest groups" of the user, to be able to use them again when a new search happens.

In addition, our approach relies on the assumption that the topology of the network in established active interest groups does not change too rapidly. We believe, that even though one of the biggest strengths of P2P systems is the ability to work in a highly dynamic environment, where servants (e.g. users) can come on-line and leave at any time, there is a pattern of behaviour that can be tracked (locally, by the individual servants) and adapted to, for the benefit of the users. The strength of the relationship between two users reflects not only a certain similarity in tastes and interest, but also a compatible pattern of being on-line. Users, who are related with strong relationships, who have been able to share valuable files / services in a mutually convenient time in the past, are likely to be able to do this again in the future.

The idea of flexibly changing the horizon for search has too been proposed in [8], depending on the how promising is the immediate neighbour to whom the request is sent. We extend this idea with the possibility of negotiating the search horizon between the servants and deploying currency (a measure of the user's cooperativeness, or balance of the relationship with the servant - e.g. if a the servant owes to our servant, it will let the query pass without decrementing the TTL). In contrast with [8], where goal is to reduce network traffic, our goal is to ensure better quality of service to users who contribute to the community.

Our future steps are evaluating the advantages and disadvantages of the proposed design with respect to user satisfaction and performance.

## References

1. Adamic L. (1999) The Small World Web. Proc. 3rd European Conf. Research and Advanced Technology for Digital Libraries, ECDL.  
<http://citeseer.nj.nec.com/adamic99small.html>
2. Adar E., Huberman B. (2000) Free Riding on Gnutella. First Monday, vol. 5, no. 10. Also available on line at: [http://www.firstmonday.dk/issues/issue5\\_10/adar/](http://www.firstmonday.dk/issues/issue5_10/adar/)
3. DAML Services: <http://www.daml.org/services/>
4. Fogg B.J. (1998) Persuasive Computing: Perspectives and Research Directions, Proceedings CHI'88, [http://hci.stanford.edu/captology/Key\\_Concepts/Papers/papers.html](http://hci.stanford.edu/captology/Key_Concepts/Papers/papers.html)



5. Greer J., McCalla G., Vassileva J., Deters R., Bull S., Kettel L. (2001) Lessons Learned in Deploying a Multi-Agent Learning Support System: The I-Help Experience, Proceedings of AI in Education AIED'2001, San Antonio, IOS Press: Amsterdam, 410-421. Available on line at: <http://julita.usask.ca/homepage/public.htm>
6. Golle Ph., Leyton-Brown K., Mironov I. (2001) Incentives for Sharing in Peer-to-Peer Networks. Proceedings EC'01, October 12-17, 2001, Tampa, Florida, ACM press, 264-267.
7. Milojicic D., Kalogeraki V., Lukose R., Nagaraja K., Pruyne J., Richard B., Rollins S., Xu Z. (2002) Peer to Peer Computing. Technical Report HPL-2002-57, HP Laboratories Palo Alto.
8. Ramanathan M. K., Kalogeraki, V. Pruyne J. (2001) Finding Good Peers in Peer-to-Peer Networks. Technical Report HPL-2001-271, HP Laboratories Palo Alto.
9. Vassileva J., J. Greer, G. McCalla, R. Deters, D. Zapata, C. Mudgal, S. Grant (1999) A Multi-Agent Approach to the Design of Peer-Help Environments, in Proceedings of AIED'99, Le Mans, France, July, 1999, 38-45. Available on line at: <http://julita.usask.ca/homepage/public.htm>
10. Vassileva J. (2001) Distributed User Modelling for Universal Information Access, C. Stephanidis (ed.) "Universal Access in Human - Computer Interaction (UAHCI)", Proceedings of the 9th International Conference on Human-Computer Interaction, New Orleans, USA, vol.3, Lawrence Erlbaum: Mahwah, N.J., 122-126. Available on line at: <http://julita.usask.ca/homepage/Agents.htm - MAS>
11. Wilcox-O'Hearn B. (2002) Experiences Deploying a Large-Scale Emergent Network, in Electronic Proceedings for the 1st International Workshop on Peer-to-Peer Systems (IPTPS '02) <http://www.cs.rice.edu/Conferences/IPTPS02/>
12. Wilde O. (1891) The Picture of Dorian Gray, available on line: <http://www.bibliomania.com/0/0/57/103/frameset.html>
13. Yu, B., Singh. M. (2002) Emergence of Agent-Based Referral Networks. Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems: part 3, July 2002, Bologna, July 15-19, 2002.
14. Shirky C. (2000) In Praise of Freeloaders, The O'Reilly Network. Available on line at: [http://www.oreillynet.com/pub/a/p2p/2000/12/01/shirky\\_freeloading.html](http://www.oreillynet.com/pub/a/p2p/2000/12/01/shirky_freeloading.html)