A Social P2P Approach for Personal Knowledge Management in the Cloud*

Rebeca P. Díaz Redondo, Ana Fernández Vilas,
Jose J. Pazos Arias, and Sandra Servia Rodríguez

SSI Group. Department of Telematic Engineering,
University of Vigo. 36301 Vigo, Spain
{rebeca,avilas,jose,sandra}@det.uvigo.es

Abstract. The massive access to the Cloud poses new challenges for the cloud citizens to use services overcoming the risks of a new form of digital divide. What we envision is a socially orchestrated cloud, i.e. an ecosystem where the users put their experience and know-how (their personal knowledge) at the service of the social cloud, where social networks articulate the assistance to users. With his scenario in mind, this paper introduces a peer-to-peer solution to the problem of sharing personal knowledge for social search. We propose a routing mechanism in a peers' network, which uses both information about the social relation among peers and their personal knowledge. Finally, we deploy the proposal over Facebook as a support network.

Keywords: Social P2P, Knowledge Management, collaborative tagging, clustering, folksonomy, tag cloud.

1 Introduction

The Social Web has erupted accompanied by a growing wave of decentralization at all levels of Web value chain: content production, service provision, decision making, tagging, etc. This decentralization trend reaches its highest expression in the main idea underlying cloud computing, the conception of the Web as a cloud of services, resources and infrastructures fully interoperable. Unfortunately, the massive availability of services in the cloud poses challenges and opportunities comparable to those of the massive availability of information in the Internet. In fact, the full realization of the cloud computing vision could worsen the well-known digital divide born with the advent of the Information Society: a citizen with no access to the cloud, or without the ability to take advantage of it, could have lessened opportunities in society. What we

* This work was supported in part by the Ministerio de Educación y Ciencia (Gobierno de España) research project TIN2010-20797 (partly financed with FEDER funds); by the Consellería de Educación e Ordenación Universitaria (Xunta de Galicia) incentives file CN 2011/023 (partly financed with FEDER funds) and by Research supported by the European Regional Development Fund (ERDF) and the Galician Regional Government under project "Consolidation of Research Units: AtlantTIC 2012-2013.

P. Herrero et al. (Eds.): OTM 2012 Workshops, LNCS 7567, pp. 585–594, 2012.

propose for this new sign of digital divide is a socially orchestrated cloudm a Social Cloud. This, taking advantage of the popularity of social networks to assist their subscribers to fulfill needs and offer opportunities in the cloud.

For this ecosystem to be developed, several aspects have to be solved: user modelling, services discovery, service description, modelling of relationships, social data mining, etc. One of the more essential ones is how we can maintain the collective intelligence in a scenario of global dimensions. Apart from the global nature of the data, changes in Web habits linked to cloud computing rule out the possibility of having a centralized architecture in which data is stored about users' consumption, tastes, assessments, etc. Deployment solutions should focus on models where the knowledge about users and services in the cloud (profiles and metadata) are owned by users and no more by providers or intermediaries, at least not only by them. This user-oriented conception (as opposed to product-oriented) comes to the aid of growing concern in the privacy of data. Then, Social Cloud can be aligned with *Personal Knowledge Management* (PKM) which, according to Higgison's definition [1], entails *"managing and supporting personal knowledge and information so that it is accessible, meaningful and valuable to the individual; maintaining networks, contacts and communities; making life easier and more enjoyable, and exploiting personal capital."* Collective intelligence of users takes the form of knowledge disseminated all over the cloud, which does not fit properly with closed solutions to decision assistance. Quite the contrary, it comes to reconciling decision-making in the cloud with P2P-related approaches so that it can take advantage of the knowledge residing in other peers.

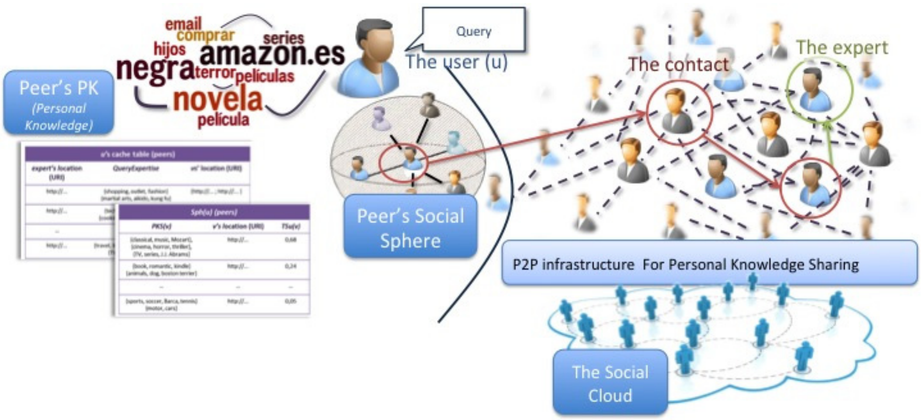

Fig. 1. The scenario por PKM in social P2P network

Taking all this reality in mind, the proposal in this paper is a Social approach to the problem of PKM whose intend is (a) to support the knowledge modeling by mining interactions in social networks; and (b) to articulate the processes in PKM by applying social P2P approaches for knowledge dissemination, retrieving, etc., i.e. P2P approaches which rely on users' relationships (social context) of cloud citizens to establish the peers architecture.

According the scenario we introduce in Fig. 1, the user *u* is a citizen of the cloud who shares his life and experiences in different Social Networks Sites (SNS). The *u's* contacts on these SNS build his social sphere: all of them have some kind of relationship with *u*, varying the strength and the context of these relationships. Besides, *u* uses SNSs as spaces of personal knowledge sharing. Depending on the privacy politics of the SNSs, the information could be confined to *u*'s the social sphere, restricting the possibility for other users outside the social sphere to access to his knowledge. What we propose is that the user *u* takes advantadge of his contacts in order to access to knowledge outside his sphere. We see contacts as social peers and, as other social P2P approaches, we build an overlay network based on the peers' social characteristics. What is a novelty in our contribution is that we propose a solution that respects the privacity and construct the overlay network only with information that is accessible in SNSs throught their publics APIs.

Under these premises, the user *u* firstly tries to find an expert on a specific matter, and so raise the query message to his social peers, which is automatically routed on the overly network until the experts is found. After that, *u* raises the question to the expert. Although it is possible a direct contact between them, we provide a mechanism based on a chain of intermediate peers who participate in propagating the questions to the expert, with the aim of guarantying that all the communications take place between two peers that share, at least, a link in one social network site and so, ensuring the communication is possible.

2 Social Spheres

According to our approach, users have a biased and egocentric vision of the world, and consequently of the social networks structure. This perspective is focused on (a) the colleagues to whom the user interacts (links or ties) and (b) the subjects about they talk about (content). With this information we define the Social Sphere from the perspective of the user *u*, as a set of 3-tuplas: $S_{ph}(u) = \{v, TS_u(v), PKS(v)\}$, where (a) v is a colleague of *u* in the cloud; (b) $TS_u(v)$ represent the tie strength of their relationship (a measure of the closeness and level of activity of the interactions between them) and (c) $PKS(v)$ is the v's personal knowledge summary, i.e. the subjects or themes in which v shows a certain kind of expertise.

The concept of tie strength was firstly introduced by Granovetter [2] as a function of duration, emotional intensity, intimacy and exchange of services and recently these measures of closeness have been used to improve a wide range of applications in social and in computer sciences. In our previous works [3, 4], we have taken advantage of the public APIs that every social network sites offer and we have automatically and dynamically obtained, having the users' consent, information about the tie strength of the users' relationships. However, and from the information about the users' activity in these sites (tie signs), we can obtain not only the tie strength (second component of the social spheres) but also the information about the context of the interaction (ways and content description), used to build the third component of the social spheres.

Tie Strength. Measuring the closeness of the relationship between a user *u* and another colleague *v* (a colleague of user *u* at least in one social site) is the objective of the tie strength index, $TS_u(v)$. Our proposal [3, 4] briefly consists of (a) classifying the tie signs according to their level of intimacy (direct interactions vs. indirect interactions, public interactions vs. private interactions), and (b) normalizing this values with respect to the rhythm of interaction of the user *u*.

$$TS_u(v) = \sum_{c \in \{d,i,pb,pr\}} \alpha_c \cdot f(|Signs_{u|c}(v)|)$$

where α_c denotes the desired weight for the signs in the category direct (*d*), indirect (*i*), public (*pb*) and private (*pr*); *f* is the normalization function and $|Signs_{u|c}(v)|$ is the number of tie signs for the category *c* associated to the interaction between *u* and *v*. Weak ties entail values close to 0, whereas strong ties entail values close to 1. Additionally, this tie strength index is not reflexive and so $TS_v(u)$ may be higher, equal or lower, i. e. since it is *u*-centered only shows the *u*'s perspective.

Personal Knowledge Characterization. We extend the mining process to extract not only the kind and number of interactions in social sites (tie signs), but also their textual description or content (knowledge signs), obtained from the social site APIs. From each knowledge sign, we obtain a set of representative or relevant tags that describe the interaction nature. This extraction is done using a natural language processor; concretely, we use the *Stanford CoreNLP* (http://nlp.stanford.edu) that allows us to generate the word lemmas for the tags and natural language in the matter of the interaction. After that, we remove the common words (*stop words*) that have little value in helping to characterize personal knowledge. Despite *CoreNLP* incorporates some form of stopping; we have utilized a larger list which include specific stop words of the domain of social interaction. The nature of the interactions any user *u* has allows us to model the personal knowledge of *u* (*PK(u)*) as a set of tag-weight pairs (tag cloud) denoted as: $PK(u) = \{t, w_u(t)\}$, where *t* is any representative tag involved in any interaction and $w_u(t)$ is the weight (relevance) of *t* in the *u*'s tag cloud.

This tag cloud is progressively built by adding every relevant tag *t* extracted from the knowledge signs. Being *Tags(u)* the set of tags in *PK(u)*, then we denote (a) its size as *#Tags(u)*, i.e. the number of tags, and denote (b) the multiplicity of tag *t* as *m(t,Tags(u))*, i.e. the number of times tag *t* was extracted as a relevant tag from a interaction description. Consequently, $w_u(t)$ is obtained as follows:

$$w_u(t) = \frac{m(t, Tags(u))}{\#Tags(u)}.$$

Therefore, *PK(u)* represents those matters or topics about the users *u* talks about in the social network sites. Our starting hypothesis is that *PK(u)* constitutes an accurately representative set of terms about user *u* is interested in and so: (a) or user *u* knows about these matters or (b) has in his social sphere other colleagues with knowledge on these topics. We define u as an expert on the topics in PK(u): someone who talks about these matters and, consequently, is interested in them. Therefore, an expert is a key person because he probably knows the answer of questions related to the topics in his knowledge or because he knows someone who can give them to us.

Personal Knowledge Summary. *PK(u)* is an extensive characterization for personal knowledge and therefore, it exhibits a great potential of reasoning. However, when storing information about the peers in order to route a query, we propose that *u* stores only a brief summary of the knowledge of his peers. This *Personal Knowledge Summary (PKS)* is computed from *PK* by applying a clustering algorithm which partitions *PK* into a set of clusters of weighhted tags so that points in the same cluster are close together according to some distance meaure among tags. We have implemented our first trials by using *k*-means algorithm and NGD (*Normalized Google Distance*) [5]. Once we have identified *k* clusters in *PK*, we use the set of cluster centroids as *PKS*. So, in this implementation, *PKS* is a set of k vectors where each vector that contains the most representative tags in the cluster.

3 The Problem to Solve

The *leit motiv* of this paper is providing a P2P architecture based on social relationships that allows any user *u* to find an expert (or experts) on a specific matter or topic, according to the expert definition given in the previous section. The search starts by user *u* asking a specific query in natural language, for instance, *"Which is the dog vaccination schedule?"*, which is processed by using the aforementioned natural language processor *Stanford CoreNLP* to extract a set of relevant tags: a query tag cloud, denoted *QTC*, that constitutes the starting point of the search.

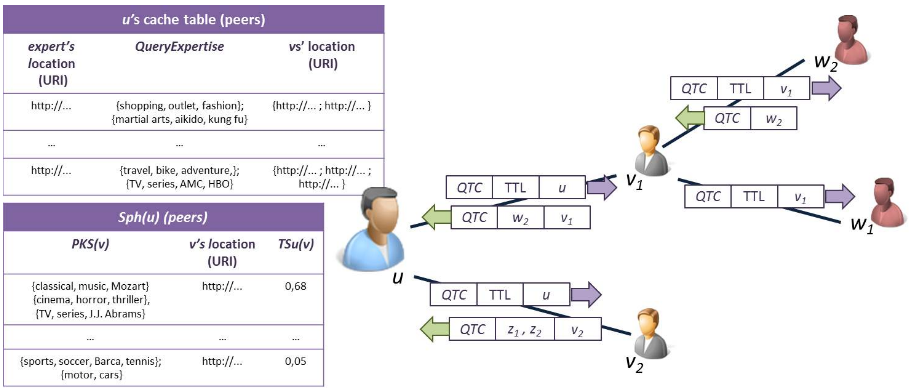

Fig. 2. *u*'s information for the distributed search

Since we have defined a user-centered knowledge model, each user in the cloud maintains his own information coming from two different sources (Fig. 2): (a) his own social sphere, $S_{ph}(u)$; and (b) a cache table keeping the information about experts out of his sphere (second table in Fig. 2), as consequence of the P2P search algorithm.

The social shere, $S_{ph}(u)$ stores the description, location information and tie strength of each user *v* to whom *u* is related in any social network site. The *u*'s table cache stores information about each expert that *u* has found as result of the execution of the

P2P search algorithm: his location information, the tag cloud of the queries on which is supposed to be expert on, and a list of the colleagues in the u's social sphere through whom the expert was located (see Sect. 4).

4 The Intelligent Seach Algorithm

The search algorithm starts trying to find at least one expert among the peers user u has knowledge about (tables in Fig. 2). If this attempt fails, it is time to forward the query to other peers to find this information out of the u's scope. Finally, whenever a user v has a set of experts for the target query, he returns this information to the user u, so he can address the question directly or indirectly to one of them.

Peers Ranking. In order to find those peers in the u's peer set (social sphere and cache table) whose expertise is close to the target matter, the algorithm firstly computes a peers ranking: (a) by comparing the query tag cloud (*QTC*) to the personal knowledge summary (*PKS*) of each colleague in the u's social sphere and (b) by comparing the *QTC* to the *QueryExpertise* of each expert in the cache table.

With this aim, we have proposed a similarity measure between two elements e_i and e_j (which might be any tag cloud previously defined: *PK, PKS, QTC* or *QueryExpertise*). This comparison takes into account not only direct tag matching but also relations among tags in a folksonomy (see our previous work [6]). This folksonomy-based similarity $FolkSim(e_i,e_j)$ takes into account those terms that, although not being included in both tag clouds, are related in the folksonomy:

$$FolkSim(e_i,e_j)=\frac{\sum_{t_k\in TC(e_i)} w(t_k,e_i)\cdot \max\{w(t_l,e_j)\cdot r_{kl} \mid \forall t_l\in TC(e_l)\}}{\sqrt{\sum_{t_k\in TC(e_i)} w^2(t_k,e_i)}\cdot\sqrt{\sum_{t_l\in TC(e_j)} \max^2\{w(t_l,e_j)\cdot r_{kl} \mid \forall t_l\in TC(e_l)\}}} \quad (1)$$

This folksonomy-based similarity does not average the weights of the same tags in both tag clouds, but the more relevant paths in the folksonomy between the tag t_k in the tag cloud of e_i and the tag t_l in the tag cloud of e_j. For that, we select the maximal value of $\{w(t_k, e).\ r_{kl}\}$, been r_{kl} the relationship of t_k and t_l in the folksonomy. We also use NGD [16] to obtain the semantic relatioships among tags, i.e. the relations in a the folksonomy.

Forwarding the Query. Having the peers ranking according to *QTC*, the algorithm stops if there are users whose comparison results are higher than an established threshold Th_{pks}. Otherwise, u has to forward the query (together with the time-to-live, TTL, and the u's URI) to a subset of his peers (see Fig. 2), which is selected according to the following 3 criteria:

a) All those peers having comparison values higher than Th_{approx_pks}. Thus, they are not experts in the query, but their knowledge is close enough to the required one. So, they constitute a target group of peers who talk about related topics to the query.

b) Those peers in the *u*'s social sphere having the highest tie strength indexes; under the premise that the more active a user is, the more possibilities there are of finding an expert among his peers.
c) A set of randomly selected peers in the *u*'s social sphere having low tie strength indexes; with the aim of broadening the search and increase the possibility of finding experts out of the *u*'s social range of action.

So, the whole procedure is as follows. Whenever a peer *v* receives the query message starts checking if himself is an expert by obtaining the similarity between *PK(v)* and the *QTC*, using Eq. (1). If this value is higher than $Th_{pks,}$ *v* is identified as an expert and, consequently, the search ends (like w_2 in Fig.2). Otherwise, *v* checks the degree of expertise of his peers (table cache and social sphere) on the query's subject, if this comparison results on a set of peers having a higher value than the established threshold Th_{pks} then the search ends because a list of experts has been found (like v_2 in Fig.2). If the comparison does not succeed, *v* forwards the query according to the previous criteria replacing in the query message the *u*'s URI by his own identification (URI) until the query TTL expires (like v_1 in Fig.2).

5 After Finding the Expert

Once the routing algorithm has been explained, the process is completed with mechanisms to (1) propagate back experts' location; (2) update information stored by peers; and (3) formulate the question to the selected experts by using some publishing mechanisms in SNS.

Answer Propagation Throughout the Peers Network. Once a peer *v* has a list of potential experts to return (himself or a subset of their peers in his social sphere or his table cache), the answer message is created, including the following information: the *QTC*, the set of the URIs of the potential experts, and the *v*'s URI (like v_2 in Fig.2). This message is propagated throughout the peers network following the same path that the query has done, but in the other way round. So, the answer message is always sent to the colleague who forwarded the query to *v*, who replaces the *v*'s URI by his own URI (like v_1 in Fig.2) and proceeds in the same way starting a chain that ends with the user who forwarded the query in the first place: user *u*. Therefore, *u* has the list of potential experts and now he is ready to make the question.

Updating Both Social Spheres and Table Caches. Information in cache table is updated each time a user *v* receives a list of potential experts from any colleague in his social sphere: he checks one by one if the experts are already in his cache table or no and updates the data, i.e. the query the peer is supposed to be expert on, his location information and the colleague who sent the information. Finally, and with the aim of maintaining a reasonable cache table size, we introduce a forgetting mechanism to remove those entries that have not been used for a long time. Data in social spheres is also updated using the answer propagation mechanism: whenever a user *v* sends a response message to one of his colleagues, he also sends his updated personal knowledge summary, *PKS(v)*.

Formulating the Question to the Experts. The whole procedure ends by user u asking the query, or any other question about a related issue, to one or more of the experts he has located. We have two main issues here: (a) how *u* selects the most appropriate expert to ask the question, and (b) how the question is sent to the expert. Regarding the former, having a list of experts to address the questions, user *u* has the opportunity of selecting a specific subset. Although in the current solution, the question is sent to all of them, we are working in incorporating some mechanism of reputation management that can aid the user in selecting the expert. Regarding formulating the question, there are two different ways to proceed. User *u* might contact directly with the expert by using one of the mechanism supported by a social network site or by using other communication strategies (e-mail, phone, etc.). Althougt this solution is efficient, it may not be the most adequate: (a) because *u* is trying to contact to a total stranger who could be reluctant to give him an answer; and (b) because some social network sites not allow direct communication with users out of our social circle. The other option is propaging the specific queries or questions throughout the overly peers network using the same peers-path used to locate the expert, by in the other way round. This is the reason why we propose to store the information about the colleagues who have forwarded the query message. Using this solution, we avoid the two aforementioned problems of communication in the social network site and/or cooperation from the expert.

6 Deployment

For the deployment of our proposal, we provide a SaaS (Software as a Service) solution for both social mining and knowledge sharing. Regarding the process of social mining, we have developed a crawler service in charge of monitoring and processing Facebook evidences or signs of relationship and knowledge in this social network site. This service, called *mySocialSphere*, acts on behalf of the user and queries the Facebook public API in order to extract evidences and to build up a model of his social sphere and personal knowledge. At the same time, *mySocialSphere* made this information available to other services in the cloud through a REST API (provided that these services have users' permission). In the architectonic scenarios, several socially-enhanced services can be deployed which make use of this information without the need of gathering it.

Despite the statistics analysis of precision and recall of our proposal is part of our future work, we have deployed a Facebook application to validate our proposal. This application improves the functionality of Facebook Questions, which let users get recommendations from his friends and other people. Our application SQ (*Smart Questions*) allows the user to make a question and automatically routes it according to the relevant tags in the question and the personal knowledge and tie strength of social peers according to the routing scheme in this paper. For this purpose, the application asks the user for his permission to use the service *mySocialSphere* on his behalf. SQ posts the question to the wall of the social peers selected by the routing algorithm. We recruit users among (under)graduate students from the University of Vigo to use the application and to give feedback about the perceived utility of the responses and the willingness to rely on a response to take some decision. We compare the students' scores in the Facebook Questions case and in the SQ case. The perceived utility of the responses improves twice in the case of SQ, although the number of responses is approximately 1/3 less in the SQ case. Unfortunaltely, the second parameter (willingness to rely on the response) worsens when the expert is outside the social sphere of the user. As we developed the trials with Facebbok, we can say that students were reluctant to take decisions according to the expert's response when the expert is neither a "friend" nor a "friend of a friend". This observation poses the need to incorporate some form of reputation management in our proposal. Besides, students have expressed a significative disappointment with the fact that some selected experts take some days to give a response in the Facebook wall. In some cases, the response is decreasingly useful when times go. That observation has motivated a modification of the search algoritm as it is described in the discussion.

7 Discussion and Further Work

In this research, we propose a social P2P approach to knowledge management based on the idea that people build social relationships which may help them in finding appropriate information or services more effectively. Social networks and P2P networks have been integrated in several previous proposals: peers in a P2P network can be viewed as subscribers to a social network and, consequently, the edges in the P2P network as the social ties in the social network. Firstly, there are some works that tackle the management of social data in a P2P architecture, i.e. a decentralized social network which does not depend on a Internet connection to access to the social network – PeerSoN [7] Safebook [8] and LifeSocial.KOM [9]. Other works [10, 11, 12] directly use the social relationships among users to construct social overlays in a P2P network that improve content location, preserve anonymity, reduce delays, etc.

More directly related with our work, in [13, 14, 15] peers are organized into social P2P networks based on similarity among users. Using the peers' knowledge and social spheres we define a dynamic overlay networks adapted on the fly to the requested matter. Besides, and because of the random factor in peers selection, deadlocks or endogamy issues are avoid. Unlike other proposals [13,14,15], our approach does not need to build social groups according to users' preferences and similarities. Unlike other proposals in the literature, we are not interested in obtaining a complete interaction network. On the contrary, we provide a social sphere centered in the user, which makes up the accessible network in the social P2P approach to knowledge sharing in the cloud. Apart from that, this social spehere is computed only by taking into account the information available in the cloud through public APIs.

Several open works are going on both in evaluating and extending our proposal in differente ways. On the one hand, after the promising subjective evaluation of undergraduate students in resolving queries, we are preparing a simulated scenario for an objective evaluation of precision and recall. On the other hand, we are defining a new scenario and a modified version of the algorithm to maximize the possibility of obtaining an inmediate response. For this new scenario the routing mechanism has to take into account the availability of the peers or, in other words, the peers that are online in the social network.

References

1. Higgison, S.: Your say: Personal Knowledge Management. Knowledge Management Magazine 7 (2004)
2. Granovetter, M.S.: The strength of weak ties: A network theory revisited. Sociological Theory 1(1), 201–233 (1983)
3. Servia-Rodríguez, S., Díaz-Redondo, R., Fernández-Vilas, A., Pazos-Arias, J.: Using Facebook activity to infer social ties. In: 2nd International Conference on Cloud Computing and Services Science, CLOSER (2012)
4. Servia-Rodríguez, S., Fernández-Vilas, A., Díaz-Redondo, R., Pazos-Arias, J.: Inferring Ties for Social-aware Ambient Intelligence: the Facebook Case. In: 3rd International Symposium on Ambient Intelligence, ISAmI (2012)
5. Cilibrasi, R., Vitanyi, P.: The Google Similarity Distance. IEEE Trans. Knowledge and Data Engineering 19(3), 370–383 (2007)
6. Fernández-Vilas, A., Díaz-Redondo, R., Pazos-Arias, J., Ramos-Cabrer, M., Gil-Solla, A.: A Social Approach to Parental Monitoring Over DVB-IPTV. In: International Conference on Consumer Electronics, ICCE (2102)
7. Buchegger, S., Schiöberg, D., Vu, L., Datta, A.: PeerSoN: P2P social networking: early experiences and insights. In: SNS 2009: Proceedings of the Second ACM EuroSys Workshop on Social Network Systems, pp. 46–52. ACM, New York (2009)
8. Cutillo, L.A., Molva, R., Strufe, T.: Safebook: a Privacy Preserving Online Social Network Leveraging on Real-Life Trust. IEEE Communications Magazine 47(12) (2009); Consumer Communications and Networking Series
9. Graffi, K., Gross, C., Stingl, D., Hartung, D., Kovacevic, A., Steinmetz, R.: LifeSocial.KOM: A Secure and P2P-based Solution for Online Social Networks. In: Proc. of the IEEE Consumer Communications and Networking Conference, IEEE CCNC (2011)
10. Gong, Y., Yang, F., Su, S., Zhang, G.: Improve Peer Cooperation Using Social Peer-to-Peer Networks. In: Proc. 1st IEEE Intl. Conf. on Information Science and Engineering, ICISE (2009)
11. Pouwelse, P., Garbacki, J., Wang, A., Bakker, J., Yang, A., Iosup, D.H.J., Epema, M., Reinders, M., van Steen, Spis, H.: Tribler: A social-based peer-to-peer system. Concurrency and Computation: Practice and Experience 20, 127–138 (2008)
12. Marti, S., Ganesan, P., Garcia-Molina, K.: SPROUT: P2P Routing with Social Networks. In: Lindner, W., Fischer, F., Türker, C., Tzitzikas, Y., Vakali, A.I. (eds.) EDBT 2004. LNCS, vol. 3268, pp. 425–435. Springer, Heidelberg (2004)
13. Yang, S., Zhang, J., Lin, L., Tsai, J.: Improving peer-to-peer search performance through intelligent social search. Expert Syst. Appl. 36(7) (2009)
14. Lin, C.J., Chang, Y.T., Tsai, S.C., Chou, C.F.: Distributed social-based overlay adaptation for unstructured P2P networks. In: IEEE Global Internet Symposium (2007)
15. Wang, F., Sun, Y.: Self-organizing peer-to-peer social networks. Computational Intelligence 24(3) (2008)