INTRODUCTION

In the last ten years, the pervasive adoption of social networking sites has deeply changed the web. Social web sites have attracted users with very weak interest in technology, including people that before the social networking revolution were not even regular users of other popular Internet services and computers in general (Stroud, 2008). The phenomenon is so widespread that many people started using social networking systems to ask questions directly to people instead of querying search engines (Morris et al., 2010) and in place of regular email. Moreover, some of the largest social networking sites constitute a separate and closed network (Sabbag, 2011). After the huge success of the early social networking systems, many other players came in the social networking market and nowadays hundreds of different social networking systems exist. Even if the social networking systems are greatly dissimilar in their user base and functionality, they are almost always centralized systems. The centralized nature allows a simple browser-based user experience and, moreover, many algorithms, e.g., friend suggestion, are far easier and more efficient to implement in this setting. However, it also presents many drawbacks, e.g., lack of privacy, lack of anonymity, risks of censorship and operating costs. The goal of this chapter is to briefly introduce social networks, to show their relationships with peer-to-peer and multi-agent systems, and to discuss about the use of peer-to-peer and multi-agent systems in development of social network systems.

BACKGROUND

A social network is traditionally defined as a structure consisting of a finite set of actors and the relation or relations defined on them, where an actor is simply a discrete individual or a social unit (Wasserman & Faust, 1994). A social networking system is a web-site allowing users to have a profile and managing their online social network, i.e., it allows them to: (i) construct a profile which represents them in the system; (ii) create a list of users with whom they share a connection and (iii) navigate their list of connections and that of their friends (Ellison, 2007).

Although we agree that self-presentation and social network management are extremely important and necessary components of a social networking system, we believe that the social networking revolution is far more related to the paradigm shift that transformed most people from mere consumers of information to full-fledged information producers. Most people create information which is essentially personal and, then, it is mainly of interest for friends and acquaintances. Before the social web revolution such information used to be essentially lost in the web, while, nowadays social networking systems are able to deliver the information to the “right” people.

After the huge success of the early social networking systems, many other players came in the social networking market and nowadays hundreds of different social networking systems exist. Even if the social networking systems are greatly dissimilar in their user base and functionality, they are almost always centralized systems because of the access and implementation advantages.

A minor drawback is that scaling centralized systems to tens or hundreds of million of users is not an easy task. At any rate, we consider this drawback as a minor one, since the problem can be solved providing enough resources. However, the huge operative costs of supporting the infrastructure necessary to provide the service to millions of users can only be justified with robust business plans. While some social networking services have extremely differentiated business models (McGrath 2010), for most of them the primary source of income is advertisement and consequently they have a strong motive for: (i) using user provided data to increase performance for that purpose and (ii) even giving access to authorized commercial third parties to the raw data. This behavior poses serious threats to privacy and data protection...
issues, especially considering that there is no clear legislation on what uses of the user data are legitimate, and regarding the conditions for disclosing the data to third parties, especially when the subjects involved are from different countries.

Another problem is that social networking systems have terms of service that their users give to the system operators a non-exclusive, transferable, sub-licensable, royalty-free, worldwide license to the submitted content (Facebook, 2013; Twitter, 2013). Such terms are needed for legal reasons: in order to serve webpages containing the users' data (e.g., their profile page) the service provider needs some rights over that data; it is nonetheless true that users are essentially allowing the service providers to do with the data whatever they want for free. Moreover, most of the times, the users themselves have not easy and streamlined ways to obtain all the data they inserted in the system in a semantically meaningful or at least in a structured way; this (i) is a serious lock-in problem in its own right and (ii) hinders users' trust in the platform. Most social networking platforms do not provide their users with easy and standard ways to export user submitted contents in a structured way. According to Fitzpatrick and Lueck (2010) this issue (i) is similar to a serious lock-in problem in its own right, and (ii) it hinders users' trust in the platform.

The last problem with centralized social networking system is that service providers are in the position to effectively perform a-priori or a-posteriori censorship, or to disclose all the information they have, no matter how private, to other entities. They can perform such actions either motivated by selfish interests or forced under legal terms and other forms of pressure.

**PEER-TO-PEER SYSTEMS AND SOCIAL NETWORKS**

Peer-to-Peer (P2P) define an open and decentralized overlay network on top of the Internet that users can use for directly communicating to find and share resources, often music and movie files (Schollmeier, 2001). Such networks are one of the few largest distributed computing systems ever, and more surprisingly, they can run with great stability and resilient performance in face of possibly the most ferocious dynamics (Qiu & Srikan, 2004).

Thus, the use of P2P technologies for the development of social network is not only viable, but also highly desirable (Wang et al, 2006). First of all, P2P systems essentially achieve automatic resource scalability, in the sense that the availability of resources is proportional to the number of users. This property is especially desirable for media sharing social networking systems, considering the exceptionally high amount of resources needed. Secondly, the popularity over time of most content on such systems exhibits either a power-law or an exponential behavior and is consequently well suited for P2P distribution (Zink et al., 2009), possibly with fallback strategies for less popular content.

Regarding censorship issues, a P2P system essentially solves them by design. Without a central entity, nobody is in the position of censoring data systematically nor may be held legally responsible for the diffusion of censurable data: the sole owners and responsible of the data are the users themselves. Unfortunately, P2P systems, and especially Distributed Hash Table (DHT) based ones, may be still liable to attacks meant to disrupt the system functionality (Urdaneta et al., 2011), often based on the introduction of a large number of Sybil nodes and the diffusion of bogus information. However, the most popular DHT systems are significantly robust because of the high redundancy they achieve by using data replication and a redundant routing mechanism. Usually, the countermeasures are based on some notion of “trust”, based on either certification authorities or some reputation mechanism (Aiello & Ruffo, 2012). Common consensus algorithms, including Byzantine agreement, have also been proposed and applied (Balfe et al., 2005; Anceaume et al, 2008).

Although peer-to-peer systems overcome the weakness of a single point of failure, there are some well-known and important vulnerabilities, including (i) Sybil attacks, or node insertion attacks, where multiple
nodes are created in the network, each of them representing fictitious identities but all belonging to a single user; and (ii) publish attacks, based on index poisoning, where essentially some bogus content is deliberately spread to the index nodes responsible for other files or keywords. The presence of Sybil nodes allows other attacks, e.g. routing attacks, eclipse attacks, storage attacks. Also publish attacks may exploit the presence of Sybil nodes, if available. Many of the proposed countermeasures to securing peer-to-peer networks are based on some notion of “trust” among peers. Depending on the approach they use to evaluate and manage trust relationships among peers, those countermeasures can be divided in two main groups: (i) credential and policy based, (ii) reputation based. Urdaneta et al. (2011) provide a detailed analysis of threats to DHTs, together with some proposed countermeasures. While underlining the existing vulnerability to Sybil attacks, authors conclude that “Current DHT deployments are not specifically designed to tolerate the presence of malicious nodes. However, most of them are based on Kademlia, which provides relative security by using data replication and a redundant routing mechanism similar to wide paths”.

Eventually, the lack of a central entity which has, or believes to have, interests in hindering interoperability (Shankland, 2011), creates the opportunity to design the system so that heterogeneous units can interoperate, typically providing a semantic common setting for the data.

Various solutions are being proposed to overcome the centralized architecture of the most widespread social networking platforms. Many of these proposals follow a federated approach, allowing users registered on a certain server to create relationships with users of other servers. Others are full-fledged peer-to-peer systems, usually based on a DHT.

Among the federated social networking systems, two of the best known are Diaspora (Diaspora, 2013) and StatusNet (StatusNet, 2013). Diaspora servers communicate by means of an ad-hoc federation protocol and the standard Salmon protocol for comments. Users can (i) participate in the network by setting up their own server, which is named a “pod”, or (ii) exploit already existing pods. Using a number of existing protocols, StatusNet shows quite strong interoperability with other networks. With regards to the completely distributed solutions, their origin can be traced back to Freenet (Charke et al., 2001), which is meant as a distributed, cooperative, uncensored and secure file system. It uses a “best-effort” unreliable routing algorithm to find content and namespaces, over both “OpenNet” and “DarkNet” connections. Various quite popular uncensored forums are built on Freenet, but they usually suffer a large amount of spam coming from anonymous sources. Similarly to other more recent systems.
Specifically in the field of social networking, various systems are being developed on the basis of peer-to-peer communications and DHT indexing. Among them, Maze (Chen et al., 2004), TRIBLER (Pouwelse et al., 2008), PeerSoN (Buchegger et al., 2009), Persona (Baden et al., 2009) and Safebook (Cutillo et al., 2009) are the most interesting.

Maze (Chen et al., 2004) supports a peer-to-peer social network through the use of some centralized services. It uses a ticketing server which issues tickets to all peers to identify them. This ticket is then served as a form of legitimate communication/transaction between peers. The ticket is valid for a single communication. For further communication, all peers need to contact the ticketing server for a ticket. Maze also uses another centralized server which, apart from holding a directory of peers, also checks online status of each of them.

TRIBLER (Pouwelse et al., 2008) is a P2P social-based file sharing network which is built on top of the Bittorrent protocol (Cohen, 2003). Tribler is based on the generation and maintenance of social networks in order to improve content discovery, searching and download performance. In particular, it proposes a decentralized recommendation mechanism based on standard collaborative filtering techniques and that takes advantage of the concepts of friends, friends-of-friends and tastes communities.

PeerSoN (Buchegger et al., 2009) is a system designed to provide encryption, decentralization and direct data exchange in the field of social networks. The first prototype of PeerSoN is designed around a PKI, though some studies are being conducted for weakening this assumption. Each user has a unique ID, possibly computed as a hash of the user’s email. The DHT is used to trace the user's network presence. An index file, containing a list of new content generated by the user, is also registered in the DHT.

Persona (Baden et al., 2009) is designed as a set of social networking services. It uses an interesting attribute-based encryption protocol for protecting access to users' content. It allows each user to create various groups of “friends”, by assigning proper attribute credentials. Content can then be associated with a publication policy and made available only to a restricted audience.

Safebook (Cutillo et al., 2009) is based on a DHT and a network of socially close peers, defined Matryoshka. Peers in a user's Matryoshka are trusted and support the user by anonymizing communications and replicating content and profile information. Safebook exploits a more traditional certification authority. In fact, a user's public key cannot be calculated from his identity, and all public/private key pairs are generated locally by the peers.

MULTI-AGENT SYSTEMS AND SOCIAL NETWORKS

Agents and multi-agent systems are one of the most interesting areas in software research and they have been importantly contributing to the development of the theory and the practice of complex distributed systems (Jennings et al., 1995). In particular, a multi-agent system offers the right abstraction for representing a peer-to-peer system and may provide an appropriate framework for developing peer-to-peer applications (Koubarakis, 2003). Moreover, multi-agent systems share with social networks both the structure and the scope, since they are composed of individuals connected with some kinds of relationship and they are realized for accomplishing individual and/or common goals (Franchi & Poggi, 2012). Therefore, it is natural to think about synergies between social network and multi-agent system research and application and about the use of multi-agent coordination algorithms for the development of typical social network services.
Several interesting works demonstrate how multi-agent systems are a suitable means for implementing social networks and their services. ReferralWeb (Kautz et al., 1997), Yenta (Foner, 1997), Community Organizer (Hattori et al., 1999), Shine (Yoshida et al., 2003), MARS (Yu & Singh, 2003) and Blogracy (Franchi & Tomaiuolo, 2012) are the most interesting.

ReferralWeb is an agent based interactive system for reconstructing, visualizing, and searching social networks on the World-Wide Web, whose main focus is selecting an expert of a given field in one’s (extended) social network (Kautz et al., 1997). In ReferralWeb a social network is modeled by a graph, where the nodes represent individuals, and an edge between nodes indicates that a direct relationship between the individuals has been discovered. For ReferralWeb a direct relationship is implied when the names are in close proximity in any documents publicly available on the Web, e.g., home pages, co-authorship in published papers or organization charts in institutional websites. The constructed network is then used to guide the search for people or documents in response to user queries; a person can: i) ask to find the chain between himself/herself and a named individual; ii) search for an expert in a given topic and providing a maximum social radius (the number of “links” in the chain connecting the person performing the query with the expert); iii) request a list of documents written by people “close” to a given expert.

Yenta is a matchmaking system that helps people with similar interests to get in touch (Foner, 1997). Yenta agents do not query the web; instead, they scan user’s emails, Usenet posts and (possibly) documents in order to discover their users’ interests and hobbies. The idea is that many potentially interesting people do not publicly write and are consequently invisible to tools relying on public data. Collected data are then used to introduce users’ to each other. Considering that in the nineties web communities were built around the idea of common interests rather than personal acquaintance, the system was a truly distributed social networking system for the time.

Community Organizer is a system where agents help the users by gathering and exchanging information, visualizing contexts, and recommending or assisting their users in making a choice (Hattori et al., 1999). Each user has a personal agent and a set of additional community agents have the function of providing shared information, knowledge, or contexts within the community and act as mediators for informal communications between people. In particular, each personal agent acquires the user profile and visualizes potential communities around the user. The community agent collects the user profiles and maintains information on potential communities. Upon a request from a personal agent, the community agent first computes potential communities around the owner of the personal agent, and then sends the necessary data (users in the potential communities and their relevance) to the personal agent.

Shine is a fully peer-to-peer framework for network community support (Yoshida et al., 2003). The framework also provides design guidelines and enables different applications to share program components and cooperate and features a peer-to-peer architecture through which personal agents can flexibly form communities where users can exchange information with peer agents. Essentially Shine is a middleware for collaborative workspaces especially tailored to implement various collaborative workspaces. Agents in Shine are goal-driven through plans: a plan is description of agent action rules. Multiple plans are executed concurrently in the plan execution module of each agent. Some plans are prepared to perform services of applications while other plans are provided by Shine to do fundamental or common tasks. A plan acts in response to external events, such as receiving a message from another agent, user input or notification about a change in the community.

MARS is a multi-agent referral system that finds experts on the basis of personal agents able to learn the user’s preferences and interests and able to build an expertise model of the other users on the basis of their responses [Yu4]. The expertise model is captured through a classical vector space model [Sal1] and each personal agent maintains the models of its neighbors. In particular, the model is updates by a personal agent on the basis of the responses of its neighbors. The response can be an answer of its user or a referral:
if the agent is reasonably confident about the expertise of its user matches the query, it directly answers; otherwise, it suggests a referral to another personal agent. Since the number of neighbors is bounded, some of them will be discarded to make place for new ones.

Blogracy is a peer-to-peer, anonymous and uncensorable social networking platform (Franchi & Tomaiuolo, 2012). The architecture of the platform is modular and is built around a module for basic file sharing and DHT operations, possibly exploiting an existing implementation, and another module providing a set of social services to the local user through a Web interface. Moreover, the platform provides two additional agent based modules respectively providing a set of pervasive services and a set of information retrieval and pushing services. In particular, the current prototype of Blogracy takes advantage of; i) Vuze (Vuze, 2013), a popular BitTorrent client implemented in Java and available as open source software, for implementing the file sharing and DHT operations, ii) Open Social (OpenSocial and Gadgets Specification Group, 2013), a set of APIs supporting the sharing of social data, for implementing the social services, and iii) JADE (Bellifemine et al., 2008), probably the most known agent development environment enabling the integration of agents and both knowledge and Internet-oriented technologies, for implementing the agent-based services.

FURTHER RESEARCH DIRECTIONS

In the previous sections we showed how a peer-to-peer architecture can help in overcoming some of the problems of the most widespread social networking platforms and how multi-agent systems can be used for providing advanced social services on such an architecture. In particular, multi-agent systems might become one of the most important means for the realization of intelligent services for social networks. However, a lot of work is necessary for updating and experiment the coordination, knowledge management, and learning capabilities provided by multi-agent systems for the development of distributed services for helping the members in their activities and for, when will be necessary, using agents as delegate of some members for exchanging information and performing tasks.

CONCLUSIONS

Social networking sites have deeply changed the face of the web in the last years. However, such systems utilize a traditional client-server architecture. This means that all the information are stored and administered on central servers. Although this approach supports highly mobile user access since users can log-in from any web browser, it also presents many drawbacks, e.g., lack of privacy, lack of anonymity, risks of censorship and operating costs. This chapter showed how peer-to-peer technologies may be used for developing social networks that do not present the previous drawbacks. Moreover, it showed how such a distributed implementation makes multi-agent systems the right solution for the evolution of the services offered by social networks.

REFERENCES


ADDITIONAL READINGS


TERMS AND DEFINITIONS

Anonymity: state of being unknown or unacknowledged to the others.

Censorship: modification or suppression of speech or other public communication which may be considered objectionable, harmful, sensitive, or inconvenient as determined by a controlling body.

Multi-agent system: a loosely coupled network of software agents that interact to solve problems that are beyond the individual capacities or knowledge of each software agent.

Peer-to-peer system: a network based system in which each node can act as both client and server for the other ones of the system.

Privacy: the right to be secluded from the presence or view of others.

Social networking system: a network based system facilitating the building of social networks.

Software agent: a computer program that is situated in some environment and capable of autonomous action in order to meet its design objectives.