Open social networking for online collaboration

Enrico Franchi, Agostino Poggi and Michele Tomaiuolo
Department of Information Engineering, University of Parma, Italy

ABSTRACT
Features of online social networks are being introduced in various applications and environments, including tools supporting virtual teams and online collaboration in general. In fact, social networking has a large potential for easing collaboration, also across organizational boundaries. However, effective e-collaboration through social networks requires the development of open and interoperable systems, allowing people belonging to different organizations to create ties, reflecting real connections existing in daily activities. Attention has to be paid to identity management, for allowing users to develop their reputation consistently, and to privacy and confidentiality, for creating a more trustful environment and protecting freedom of expression. In this article we propose a novel peer-to-peer system, named Blogracy, leveraging the large BitTorrent network. Its main distinguishing features are: (i) protection of users' identity and privacy, (ii) absence of central control over published content; (iii) interoperability with other systems; and (iv) low costs and scalability of the infrastructure.

Keywords: social networks; virtual teams; micro-blogging; peer-to-peer; file sharing; privacy.

INTRODUCTION
Online Social Networks (OSNs) had an outstanding impact in our society because of their capillary adoption and sheer number of users involved. OSNs not only changed the private sphere of socialization, but also played an important role in evolving the way institutions and organizations interact with their customers. Moreover, many traditional software applications are evolving towards an open model, often based on services, where the social relationships among involved parties are made explicit, taken into account and leveraged. This trend regards existing information systems and also, for some aspects, platforms for Enterprise Resource Planning (ERP). In fact, as inter-organizational collaboration is becoming a central point in the new digital society, focus needs to move from prefigured organograms to the real emerging relationships, in either a local or global network, in which people are actually engaged while performing their activities.

One of the first traditional business applications that is being integrated with “social” ideas is Customer Relationship Management (CRM). The trend towards “Social CRM”, although quite popular, has not always been satisfactory. A study from IBM (IBM Institute for Business Value, 2011) shows that there is quite a large gap between the expectations of brand managers and social media users. In fact, only 23% of users are keen to engage with brands on social media, and only 5% of users declare active participation. Most users, instead, limit their communications and shares with parents and relatives. Among the potentially interested people, many expect tangible benefits, including discounts, services, additional information and reviews about products. The study is in accordance with the difficulties that brands face to engage with users and to launch viral campaigns. Nevertheless, businesses continue to be greatly interested in using social media for rapid distribution of offers and content, reaching new people through trusted introducers, but also for improving customer care and research.

Another area that companies are trying to improve using ideas from OSNs is that of Knowledge Management (KM) systems, where the relationships among the entities involved can be classified using
categories from social networks. In fact, the long-term goal of KM is to let the insights and experiences that already exist implicitly within an organization emerge and become easily accessible for wider internal adoption. Such knowledge can be either possessed by individuals or embedded into common practices. To provide effective access to valuable internal knowledge and expertise, it is essential to recognize and value the particular knowledge possessed by different persons, and then to have means to contact the relevant persons in a timely manner. In many regards, such a scenario requires the addition of some specific features of social networking applications into tools for daily activities.

But also other aspects of traditional ERP systems are integrating features of OSNs. In fact, the centralized and inward approach of early information systems is being challenged also in the core area of production management. An increasingly dynamic production environment, arranged in the form of complex Virtual Organizations and Virtual Enterprises, testifies the drift towards networks of integrated enterprises. In this context, the tasks of supply chain management, project and activity management, data services and access control management require the participation of actors of different organizations and possibly different places and cultures (Madlberger, 2009).

However, while the difficulties faced to manage their business knowledge in an effective way are leading enterprises and organizations to integrate OSNs into their information systems, also those new systems face similar problems, including limited and localized participation. Social networking tools are no silver bullet and they must be modelled according to the peculiarities of each environment where they are deployed.

In the next Section, this article reviews the problems and theories of participation in social networks and other online collaboration systems. In the third Section, it provides a survey of relevant standards, formats and protocols for building open OSNs, adapting to decentralized and dynamic virtual environments. In the fourth Section, it presents a system, that we named Blogracy, which exploits popular peer-to-peer technologies and open standards for realizing a completely distributed platform for social networking, suitable for fluid Virtual Teams and other online communities. Finally, related works and concluding remarks are highlighted.

PARTICIPATION IN VIRTUAL ACTIVITIES

The study of structure of OSNs, expressed in patterns of links among nodes, can exploit models of classical sociology and anthropology, with particular attention to contextual and relational approaches. But these results, obtained in decades of studies of human networks, may require significant efforts for being adapted to the different scopes of Virtual Organizations (VOs), Virtual Teams (VTs), online Networks of Practice (NoPs) and Networks of Interest (NoIs).

Understanding the reasons why users engage and share their valued knowledge in an OSN, requires analysing the nature and structure of the network, and the implication of online activities over both online and offline reputation. A relevant theoretical foundation for analysis is social capital (Burt, 1992), which essentially tries to measure the advantage given to an individual by his social connections. More recently, Wasko & Faraj (2005) analysed a number of factors enabling participation in so-called online Networks of Practice. These include (i) individual motivations, (ii) structural capital, (iii) cognitive capital, (iv) relational capital. Those factors have different importance in different social contexts. The case-study the authors analysed involved a network of legal professionals and the factors they considered were the importance of (i) reputation, (ii) willingness to help, (iii) centrality in the network structure, (iv) self-rated expertise, (v) tenure in field, (vi) commitment, and (vii) reciprocity. In their particular setting, reputation played a pivotal role, since it also affected professional activity. However, other factors were also positively correlated with the number and significance of contributions in the online community.

Contractor & Monge (2003) proposed an even more multifaceted approach, using a multi-theoretical
multi-level (MTML) model, for explaining the various motivations for the existence of social connections in a network. The model takes into account (i) self interest theories, (ii) mutual interest theories, (iii) contagion theories, (iv) cognitive theories, (v) exchange and dependency theories, (vi) homophily and proximity theories, (vii) balance theories, and (viii) coevolutionary theories.

In addition to the lack of perceived social incentives, some online communities may face a problem of scarce participation due to questions regarding adopted mechanisms and policies for preserving privacy, including confidentiality of messages and identity. Facilitating participation in those cases requires to balance the protection of users' online presence with the need to identify and evaluate the parties they communicate with. For protecting the privacy of personal identities, stable pseudonyms could be assigned at registration (Andrews, 2002). Moreover, in online communities and Virtual Teams, acquaintance may occur online, without previous connection in real life. In those cases, a member's reputation is tied to his pseudonym; for creating trust, evaluability of his online activities may be more important than his real world identity. Complete anonymity may also have a value in some activities of Virtual Teams, apart from encouraging participation in general. For example, an anonymous brainstorming activity may help opening a conversation and defining ground rules for future online meetings (Young, 2009).

OPEN SOCIAL NETWORKING

For reaching wider and more effective adoption in open and dynamic online communities, including Virtual Organizations, Virtual Teams and online Networks of Practice, we argue that social networking platforms should embrace an open approach. In fact, many isolated sites could not satisfy the need for an inter-organizational collaborative environment. On the other hand, organizations are not keen to rely on a single centralized site, which may pose risks to privacy and may control published data. Even a simple editorial project, involving cartoons or satirical pieces, for example, could be subject to flagging by other users and thus to some sort of censorship. But more serious threats of this kind can come from service operators as well as other institutions. Instead, online social collaboration would be better based on decentralized and interoperable social networking systems. Possibly, such systems should also provide reliable pseudonymity, confidentiality of messages and resilience to third-party control on published data.

Moreover, openness is important for participation, too. In fact, a closed environment can hardly reach the minimal dimension and variety required for activating the typical dynamics at the basis of the different theories taken into consideration by the MTML model, for explaining participation in OSNs. In the rest of this section, we will describe the basic requirements of OSNs, the interoperability issues and some available solutions and schemes, and finally the problems related to confidentiality of users' identity and content, in terms of access control and encryption.

Basic requirements of OSNs

In OSNs there are at least three distinct functional elements: (i) profile management, (ii) social graph management and (iii) content production and discussion. In fact, by definition, a social network cannot lack social graph management and self-presentation, no matter how minimal. On the other hand, virtually no modern OSN lacks the content generation features. According to these three main functional areas, it is also possible to draw a classification of the OSNs in three main categories: (i) systems where the profile and social graph management is prevalent; (ii) systems where the content has a prominent role with respect to social networking activities and there are frequent interactions with people not closely related; and (iii) systems where the two aspects have roughly the same importance. The archetypal examples of the first category of systems are business-related and professional OSNs, like Linkedin. People pay a great deal of attention in creating their profile. In this type of systems there
are usually various relationships among users, representing the variety of relationships that members may have in real life. Most users do not visit the site daily and do not add content to the system often (Skeels & Grudin, 2008).

The second type include blogging, micro-blogging and media sharing web sites, like Twitter. The “follow” relationships, which are typical for a system of this kind, are usually not symmetric. The focus is in information transmission; often the system does not support a proper profile and sometimes even the contacts may be hidden. Often weak semantic techniques such as Twitter hash-tags are used, in order to read content by subject instead than by author. Through collaborative tagging, the actors of the system may develop a sort of emergent semantics (Mika, 2007), possibly in the form of so-called “folksonomies”. Considering that tags usage is a heavy tailed power-law like distribution, i.e., most people actually uses very few tags, collaborative tagging usually produce a good classification of data (Halpin, Robu & Shepherd, 2007).

The third category includes the personal OSNs, like Facebook. In this type of systems, users have a profile, partly public and partly confidential. Frequently, there is only one kind of relation, “friendship”, which is symmetric and requires approval by both users. These sites have extremely frequent updates: a noticeable percentage of users perform activities on the system at least on a daily basis (Skeels & Grudin, 2008).

**Interoperability of OSNs**

Among the open protocols and data formats for conveying profiles and contacts, **Portable Contacts** (sometimes called “PoCo”, [http://portablecontacts.net/](http://portablecontacts.net/)) shows some benefits, especially from the point of view of interoperability. In fact, it is quite simple and well supported by existing large social networks and mail systems, to manage lists of “friends” and address books, respectively. It also allows to associate tags and relationship types with each user, thus paving the way for semantically annotated social networks. The protocol is not built from the ground up. Instead, it leverages existing technologies and, in fact, it can be described as a combination of OAuth, XRDS-Simple and a wire-format based on vCard. On the other hand, Portable Contacts is at the basis of other protocols, including OpenSocial, W3C Contacts API, OStatus. In order to let users express their profile, Friend of a Friend (FOAF) is another sensible choice (Brickley & Miller, 2005). FOAF is a descriptive vocabulary expressed using the Resource Description Framework (RDF), thus it is a relatively complex format, but providing precise semantic. Among other things, a FOAF profile can contain references to attended schools and workplaces where employed, which are very practical for the automatic discovery of acquaintances. Different FOAF profiles can only be linked by means of the “foaf:knows” relation, which is deliberately generic and should be read as a “follows” relation, when eventually used in a microblogging application.

Content publication and distribution is another important requirement of OSNs; i.e. followers should be notified that one of their followees updated some resource. Along with the diffusion of the web in general, and blogs in particular, **Atom and RSS** emerged as two similar technologies, intended to help readers to receive automatic updates of their favourite websites. Those feed formats, together with newer specifications, can also be applied to social networks. Traditionally, RSS and Atom protocols use a pull strategy, i.e., the observer periodically checks the observed resource for updates. As an alternative, social networking applications could adopt a push strategy, i.e., the update is automatically announced to the observer. The **OStatus** protocol ([http://status.org/](http://status.org/)) is a minimal HTTP-based specification for realizing a publish-subscribe mechanism; it is used for distributing feeds timely and without the typical overhead due to polling. The core of the specification relies on the PubSubHubbub (PuSH) protocol, which is designed around a hub, where both publishers and subscribers need to register. Publishers notify their hub about all updates of their feed. Subscribers are then contacted by
the hub, at the callback URL they provided with their subscription request. The OStatus protocol is relatively low-level and applications have to rely on other protocols for representing and managing contact lists and actual feeds. Usually, information pushing is preferred to polling, for the reason it proves more efficient in term of required resources. For example, Sandler et al. (2005) associate the RSS polling mechanism with issues like superfluity of requests, stickiness of casual readers and 24h traffic. However, information pushing requires that followers are always connected. Otherwise, a disconnected user will not receive updates and, when back online (possibly after a long period), he could miss some previously advertised resource. As a consequence, we advise to combine the two approaches, decreasing the polling frequency to improve performance and providing timely updates using push strategies.

An on-going and well-supported effort to standardize typical users’ activities in social networks is Activity Streams ([http://activitystrea.ms/](http://activitystrea.ms/)). This is an open format specification for the syndication of activities taken in social web applications and services. The activities of a user are represented as a flow, which friends and followers are interested into and want to subscribe to. It defines the semantics of: (i) a significant number of verbs, including Post, Share, Like, Tag, Invite, Join; (ii) a number of object types, including Article, Audio, Video, Image, Collection, File, Event, Group, Person; and (iii) some properties, including Location, Mood, Rating, Source, Tags. These semantic concepts are also suggested for use inside RSS and Atom feeds. Additionally, Activity Streams are increasingly being adopted for project management and tracking, and for information and content management in businesses and organizations. Thus, they allow the integration of diverse data, potentially collected from a number of sources.

The tendency to collect and aggregate content originated from different contexts is directly related to variety of sources. RSS and Atom protocols allow to poll a particular source for new content, while PubSubHubbub adds the possibility to push updates directly to interested readers. The main problem with this approach is that the original source is oblivious to the comments, ratings and annotations that take place in the aggregator. On the other hand, the Salmon protocol ([http://www.salmon-protocol.org/](http://www.salmon-protocol.org/)) is an open, simple, standards-based solution that lets aggregators and sources unify the conversations. When an aggregator reads a feed, it looks for the presence of a Salmon link. Later, when a reader leaves a comment on a feed item at the aggregator, the Salmon URL is used by the aggregator for notifying the comment to the original content source. Sources can then decide to actually publish the salmon, or to filter it according to local moderation rules and policies.

Finally, OpenSocial ([http://opensocial.org/](http://opensocial.org/)) is a set of common APIs, defined in the form of RESTful Web services, that allow developers to access core functions and information at social networks. Those APIs are dedicated to access and manage different aspects of social-oriented applications, namely (i) information about a user's profile, (ii) information about the social graph connecting users, and (iii) activities occurring in the network, including status updates, publishing of new content and media, commenting and tagging. OpenSocial also allows social applications to be developed according to a Service Oriented Architecture, by composing gadgets for collecting and organizing data from different services in a single user interface. The OpenSocial specifications integrate or support numerous other open web technologies, including OAuth, Activity Streams and Portable Contacts. Since its original conception in 2007, OpenSocial has enjoyed support from various OSNs, including Ning, MySpace, Orkut, LinkedIn, Friendster. More recently, it is gaining solid support also from enterprise software vendors, such as Jive, SAP, SocialText, IBM, Nuxeo, Atlassian. They are promoting OpenSocial for operation “behind the firewall”, in particular as an extension mechanism for including social features in web-based business-oriented applications.
Privacy and control
As described in the second Section, reputation is an important motive driving to active participation in online communities. Most theories, in fact, postulate that some persistent identity is associated with a user, thus allowing his contribution to be consistently evaluated. Moreover, a social network cannot be built without any kind of identification of involved users. However, the online identity may often be different from a user's real name, either for opportunity or necessity. Thus, an OSN for inter-organizational collaboration needs to securely bind some kind of identifier to each user. These identifiers can be plain pseudonyms, or they may be associated with users' real identities. If needed, trusted authorities may be required to certify the association with real identities. Otherwise, pseudonyms can be used to conceal real identities when required.

Some issues about anonymity may also be posed at the lower network level. In fact, if posts and actions are related to plain network addresses, these can be easily associated with a particular person or entity. For solving these issues, various anonymizing technologies exist, ranging from simple proxies, to complex networks based on Chaum’s mix-net scheme (Chaum, 1981). The latter include the famous Tor (Dingledine, Mathewson & Syverson, 2004), a network based on the Onion Routing protocol, with centralized management, and I2P (Zzz & Schimmer, 2009), also based on a variation of the mix-net scheme, but completely distributed.

For verifying authorization across different applications, OAuth is often used. It is an open standard which allows users to share their private resources (e.g. photos, videos, contact lists) stored on one site with another site, without having to hand out their original username and password. Instead, a security token is used to grant access to a specific site (e.g., a video editing site) for specific resources (e.g., just videos from a specific album) and for a defined duration (e.g., the next 2 hours).

In order of maintaining confidentiality while transferring sensible messages over public networks, access control is not sufficient and it is necessary to encrypt the messages. Encryption of data in OSNs can be performed through traditional means, based on secret-key or public-key cryptography. Moreover some privacy models, based on the CP-ABE (Cyphertext-Policy Attribute-Based Encryption) protocol, allow attribute authorities to certify the attributes possessed by certain users, so that later a resource owner can protect access to its resources, enforcing its own policies based on those attributes (Bethencourt, Sahai & Waters, 2007).

DEVELOPING AN OPEN SOCIAL NETWORKING PLATFORM
Considering (i) the basic requirements of OSNs, (ii) available protocols and formats for interoperability, and (iii) additional features facilitating adoption and participation in the context of online communities and teams, we therefore present a new system, which we named Blogracy (http://www.blogracy.net/). In the following Sections, we describe its main distinguishing features and how they relate to other existing systems and abstract architectures. Our new system is built on the top of popular services; we chose to leverage existing and widespread file sharing networks, and to provide specific features (i) for micro-blogging applications and (ii) for publishing personal activity streams to those enormous communities.

The architecture of the application is modular and is built around two basic components: (i) an underlying module for basic file sharing and DHT operations, possibly exploiting an existing implementation, and (ii) a module providing the services of the social platform to the local user, who accesses them through a web interface. For better interoperability, the module is implemented as an OpenSocial container. The core system is extended using agents that provide (i) recommendations of both users and content, (ii) personalization of results, (iii) trust negotiation mechanisms. In the following paragraphs we will describe the most distinguishing features realized in Blogracy over this extensible architecture.
For its basic operation, Blogracy uses a peer-to-peer file-sharing mechanism and two logically separated DHTs. Users in Blogracy have a profile and a semantically meaningful activity stream, which contains their actions in the system (e.g., add a post, tag a picture, comment a video). One DHT maps the user’s identifier with his activity stream, which also contains a reference to the user’s profile and references to user-generated content (e.g., posts, comments). These references are keys of the second DHT, which are then resolved to the actual files. The files are delivered using the underlying peer-to-peer file-sharing mechanism.

These basic file sharing techniques are well tested and in widespread use. In particular, it is common practice to associate files (or file parts, or chunks) with their hash, and using the hash itself to identify a file, to share or download. A quite similar technique was found also in Freenet (Clarke et al., 2001), as well as most of the following networks.

In the rest of this Section, we motivate the design and architectural choices we made for Blogracy, with specific attention to social networking features, semantic representation of data and anonymity. First we deal with the problems of identity, anonymity and privacy. Then we focus on the semantic representation and efficient diffusion of users activities and user created content. Afterwards, we are concerned with supporting social networking in an open and interoperable way. In the following, we deal with implementation issues, mainly related to the choice of an adequate DHT, and data availability, i.e., the problem to ensure that content placed on the network is accessible after the publisher disconnects. Eventually, we present some early results of the first tests of the system.

**User identity and privacy**

For all practical purposes, individual users of large networks have to be associated with numerical identifiers or unique strings, since names used in real life are hardly unique. Moreover, true anonymity or, at least, pseudonymity are often a requirement for micro-blogging and other Internet applications. However, even under anonymity or pseudonymity, users’ content need to be verified for authenticity and integrity, properties which can be easily enforced by means of public key cryptography and digital signatures.

Usually, a public key is associated to a person or a legal entity through a certificate issued by a globally acknowledged authority. However, in a key-based identity scheme (Li, 2000) and particularly in micro-blogging applications, this is not required nor always desirable. In these schemes, a user’s public key is used directly to represent the user, so that all content produced by the user can be easily verified against his public key, which is also his own main identifier. Alternatively, a cryptographic hash of the public key can be used (Li, 2000), without loss of security, and this is exactly the scheme adopted in Blogracy. For simplicity, the hash function is the same as the one used by the DHT. This way, the public key works as a pseudonym for its user, and assuring authenticity, thus allowing users to develop their reputation in a reliable way.
It is worth mentioning that, while the system does not rely on Certification Authorities for its core functioning, a traditional public key certificate could still be associated with a user's real identity, if required in a particular online community. Conversely, should users require to hide also their network addresses, which are typically published in DHT entries and associated with shared files, they would need to install additional plugins, available for generic usage on the peer-to-peer platform, for exploiting Tor or I2P protocols.

In previous sections, we argued that many users highly value the protection against unauthorized access to their data, including posts, contacts, communications and activities in general. Since the core sharing system of Blogracy is completely agnostic with respect to published content, data can be easily encrypted with a cryptographic algorithm, either symmetric or asymmetric, including attribute-based encryption. In fact, similarly to Persona (Baden et al., 2009), Blogracy privacy model uses an encryption scheme based on the CP-ABE protocol for creating a sort of very flexible circles of personal contacts, i.e., parametrized roles to be assigned to users for granting a certain set of access rights.

**Semantic activity streams**

Since users are uniquely identified by their ID, i.e., the hash of their public key, we use such key to associate with additional pieces of information, including personal profile and personal activity stream. In Blogracy, personal activities are included into a feed, represented according to the Activity Streams standard and eventually signed to avoid tampering. This feed is shared with friends and followers using the underlying file-sharing platform. Essentially, a hash of its content and a magnet-URI are used for identification and retrieval. Although the term magnet-URI originated in the BitTorrent community, it is of general applicability and we use it to refer to an URI that identifies the resource.

Since we expect feeds to be updated rather often and without a specific pattern, the magnet-URI of a feed file is not sufficient to follow a user’s activities. In fact, adding new activities to the feed would change its content and consequently its hash, resulting in an entirely different entity on the DHT. This is the reason why we have two logical DHTs: (i) the first DHT (DHT1) associates the user’s key with the last version of his activity stream, in the form of a magnet-URI; (ii) the second DHT (DHT2) is just a regular DHT and it is used to resolve the magnet-URI of a resource to some locations where it is actually available. Although the two DHTs are two distinct logical entities, they can be physically implemented on a single DHT.

![Figure 2. New resource sharing process.](image-url)
When a user generates and shares new content, he also updates his own activity feed and shares the updated feed as well. Eventually, he updates the association in DHT1 linking his ID with the new magnet-URI of his feed. The strategy is inevitably more complex than the one adopted by centralized systems, considering that they can simply push notifications. On the other hand, in a decentralized context, these events have to be advertised in a public and open way, i.e., by means of an updated feed file listing a user’s activity. The process is illustrated in Figure 2.

When a user comes online and desires to check the feed of one of his followees, the follower has to search for the followee’s key on DHT1; then, if the magnet-URI is different from the last observed, the file containing the updated feed is retrieved (see Figure 3). Since feeds are signed, it is possible to trust their authenticity and integrity, and since they are marked with a publication date, it is possible to discard older copies.

An important issue regards how followers should be notified that one of their followees updated some resource. Traditionally the strategies are: (i) pull, i.e., the observer periodically checks the observed resource for updates, or (ii) push, i.e., an update is automatically announced to the observer.

Unfortunately, since in a peer-to-peer environment users are not always connected, information pushing alone is not a viable mechanism for dispatching updates about users’ activities. In Blogracy, searching for a user’s feed occurs at start-up and it does not imply the repeated transfer of the whole feed, possibly unchanged. Instead, when requested, only a magnet-URI is retrieved, and the whole feed is downloaded only if the URI changed. The download process itself is fully distributed among available peers that already have a full copy of the feed (called “seeds” in the P2P lingo) and thus it becomes more and more efficient as the number of followers grows. Afterwards, new activities are notified directly among interested nodes. In fact, information pushing is handy in the case of fast interactions between users. In those cases, for example commenting or tagging over recent resources, direct message passing among users interested in a shared resource is used to improve system responsiveness. Consequently Blogracy benefits from peer-to-peer messaging facility, as provided by the file-sharing protocol. In fact, for their basic operation, file-sharing systems usually need to keep track of the peers that are currently seeding or downloading a certain file (sometimes collectively defined as a “swarm”). So, advertising about a new feed is simply a matter of contacting the peers that are sharing the superseded version of the user's feed.
Open social networking
In the previous Section we suggested that an application that does not provide explicit representation for the user’s contacts should not be considered a social networking application. Here we describe the social networking features of Blogracy. Essentially, in Blogracy users define and manage a list of other users, represented by their IDs. At the current stage, for representing profiles and contacts, Blogracy adopts Portable Contacts with OpenSocial extensions. The format has some benefits from the interoperability point of view, because it is quite simple and well supported by existing large social networks and mail systems. It also allows associating tags with each user, thus matching the basic data structure managed by Blogracy.

Besides Blogracy IDs, web URLs can also be used, to provide better interoperability with other web-based platforms. In fact, since Blogracy handles users’ feeds in the form of standard Activity Streams, it can also manage similar feeds obtained in other ways, seamlessly integrating content from web blogs and social networking sites with content from the peer-to-peer network. Moreover, OpenSocial services, and in particular Activity Streams, are increasingly supported in various kinds of software, including project management and tracking systems. One of the best known examples is probably Atlassian JIRA, a system for managing a software development process. Blogracy is designed to import this kind of data into its OpenSocial container, thus providing an integrated interface for managing the various aspects of collaboration in a Virtual Team, for example.

On the other hand, resources distributed through Blogracy can be easily replicated over the web. Actually, the system architecture has a web interface, for user operation. Normally, the integrated web server acts as a gateway for its local user, who can access and publish peer-to-peer content through a browser. By default, the web server runs on the user's machine, together with a module encapsulating basic file sharing services. But a Blogracy instance can also be hosted on a remote node and configured for public access, acting as a gateway for Blogracy public content. Alternatively, it can be configured to serve one or more registered users, who connect from mobile or constrained devices, or have other reasons for not setting up a peer node on their machine.

Blogracy over BitTorrent
The system description provided so far did not refer to a specific platform and it only requires that the underlying platform provide a DHT and file-sharing mechanisms. Moreover, we mentioned that we would build Blogracy over existing and widespread systems. Specifically, we choose BitTorrent (Cohen, 2003) as the file-sharing protocol. BitTorrent is extremely popular and many applications implement the protocol.

The traditional use of BitTorrent relies on a tracker server, which hosts what are commonly called torrent files, referred to as meta-info files. A meta-info file contains information relative to a shared file. It also contains a reference to a tracker server, which lists the locations of the available seeds, from which the file can be finally obtained. Such a centralized indexing has raised some legal issues and has been proven to be an easy target for attacks.

As a consequence, alternatives that do not require a centralized tracker were designed; such strategies are usually referred to as “trackerless”. Trackerless BitTorrent replaces the tracker with a Distributed Hash Table. The popular BitTorrent client Azureus (now called Vuze) was the first program to introduce a working DHT; the Vuze DHT is an implementation of Kademlia algorithms (Maymounkov & Mazières, 2002) and is called Distributed Database (DDB) in source files. Later, the official BitTorrent specification introduced a DHT, called Mainline DHT, also based on Kademlia; nowadays, the BitTorrent application and other popular clients, including Vuze itself, implement it.

Since the DDB, differently from the Mainline DHT, allows to store and retrieve arbitrary keys and values, we decided to use it. Moreover, it has a large user base in the order of one million users
instantly connected. Vuze is implemented as a modular platform meant to be extended with plug-ins that access a well-defined interface. Blogracy communicates with Vuze using a specially crafted plugin. We are gathering the on-going implementation efforts in Blogracy own development website (http://dev.blogracy.net/).

One of the most important issues in such a decentralized system is data availability. In fact, in a personal OSN, popular content will quickly gain lots of seeds. Posts published by peripheral users, with few contacts and sparse online presence, will instead suffer poor availability, to the extent that it is possible that the publisher remains the only seed for his own new posts. However, in a Virtual Team, which exists for completing a given task, it is quite reasonable to require that a minimum of resources are dedicated to the effort. In fact, the design of the system, based on solid file-sharing protocols, makes it resilient to diverse node presence, and it is sufficient that at least one of the nodes of the team is connected at each instant, to assure timely propagation of new content. Otherwise, content is not lost, but its diffusion to interested followers is delayed. As a more generic solution, we foster a replication system based on acquaintances, which gives special responsibilities to introducing users with respect to the users they invite into the system. The protocol will require introducing users to replicate their new friends' content and to suggest other close friends to do the same.

Initial results
After having implemented and tested all the core features of Blogracy, we are confident of the soundness of its architectural design and its realization over solid and widespread technologies. The use of a DHT for both sharing files and advertising new social activities proved to be functional. As expected, the use of a DHT imposes a delay of about two minutes for normal lookup operations. In fact, the results we obtained in our tests are coherent with data provided by some large scale analysis of the Vuze DHT (Falkner et al., 2007). This delay is responsible for the rather long start-up time of a Blogracy node, which is in the order of few minutes. When a node is connected to the network, it is required to find and download the updated streams of activities produced by followees while the local user was offline. Nevertheless, after the start-up phase, the use of adaptive pushing for communicating fresh updates grants a seamless communication among connected users. In fact, apart from the download of larger files, all communications about status updates and new activities occur directly among interested peers. Using the same mechanisms, which leverage the swarm of nodes sharing a user's stream, we also integrated a chat system into Blogracy, for both private instant messaging and group discussion. In both cases, the system operated with the expected responsiveness, not differently from more traditional server-based systems. Additionally, since the users' nodes are connected directly, in a peer-to-peer fashion, Blogracy does not suffer the typical problems of centralized services, for example in the periods when they are overloaded by users, under maintenance work, or outright out of order.

OPPORTUNITIES AND CHALLENGES FOR AN EFFECTIVE DEPLOYMENT
As discussed in the previous Sections, Blogracy has some advantages over a traditional social networking system, mainly related to (i) openness/interoperability and (ii) privacy/confidentiality. In the rest of this Section, we discuss the applicability and the challenges of using Blogracy in a collaborative environment. In particular, we first deal with the management challenges posed by the introduction of an open social networking system in a Virtual Team. Then we present a use-case, related to a network of physicians, that shows the strong points in the confidentiality area.
Management challenges

Some analysts argue that, in general, the adoption of social networking tools by businesses may improve their market presence and also increase revenue (at&t, 2008). In this case, apart from revenue, new indexes have to be analyzed to measure the social performances. The shift could materialize in: (i) expanded reach in new markets and niches; (ii) effectiveness of direct marketing and product launch; (iii) improved customer experience through some form of social CRM; (iv) integration with external information channels. The effects we mentioned are expected to be stimulated by embracing the changes brought by social networking, in communication forms, business vision, internal culture and organization. From the point of view of internal organization, project leaders will probably need to play the role of social networking evangelists, while knowledge management professionals should become social networking architects, in direct contact with other IT professionals (Anklam, 2004; Forgie, 2011; Roy, 2012). However, the adoption of social networking tools is going to occur at different speeds, requiring plans for implementing changes and means for encouraging effective participation. In fact, rather unsurprisingly the impact of social networking is much larger and started sooner on Virtual Teams, start-ups, small and geographically sparse organizations than on large corporations.

In any Virtual Teams, the adoption of an open social networking tool like Blogracy allows each participant to improve his skills and knowledge, not only by accessing raw information, but also receiving mentoring and tutoring hints and guidance from online contacts. On the other hand, a participant in the team will similarly enjoy an increased visibility inside a larger community, making his expertise known and available to his acquaintances. Thus, depending on the dimension of his social network, each participant will be increasingly involved in lateral activities and possibly be solicited to move to other projects. This is potentially disruptive for both large companies and small teams, representing a risk of loss of human and intellectual capital. Accepting this challenge requires teams, as well as large organizations, to motivate participants to continue their engagement in collaborative work. Possible solutions may involve an active role of so-called animators and a specific reward system based on growing reputation and karma inside a community, in addition to individual incentives (Chui, Miller & Roberts, 2009). Career paths could also be designed to acknowledge the important role of animators, business bloggers and other experts, acting in a visible way inside a community and promoting participation.

Especially in the case of software development, product life-cycles are getting shorter and shorter. The materialization of an idea into a mature Web 2.0 application, with millions users, may take just a couple of years. Innovation is often sustained by forms of collective thinking (Dwyer, 2011; Fedorowicz, Laso-Ballesteros & Padilla-Meléndez, 2008; Hayne & Smith, 2005), either through e-Collaboration tools or real meetings, involving team participants and other partners from the value chain. This may be particularly useful for anticipating customer needs. The capacity to react to such a dynamic environment depends on the size of the organization, but also on visions, plans and tools deployed to embrace changes. Conversely, resistance to the introduction of social networking tools may slow innovation down. It is often motivated by the perception of social networking as a cost inefficient activity, in which employees get involved despite of duties with higher priority. This perception may induce a sort of “Big Brother syndrome” among managers. Another important issue is the confidentiality of sensitive information of projects and organizations, which may become harder and harder to maintain while organizational boundaries fade. Encryption mechanisms like those provided by Blogracy are necessary, but not sufficient. Training on the risks of information leakage should not be limited to those in contact with customers. Instead, security awareness should be introjected by all the participants in a Virtual Team or the members of a large organization.
Use case: network of physicians

There's virtually no professional area were the specialists could not greatly improve their skill-set and obtain new insight by discussing with their colleagues and sharing their experience. Most of such communication occurs during conferences and seminars where the delegates can easily socialize and discuss face-to-face. However, new media such as online social networks provide an excellent way to discuss and share experience off-band with respect to regular meetings, even if at the cost of increased cognitive effort and ambiguity, considering the asynchronous nature of the communication and the complete loss of non-verbal signals (DeLuca, Gasson & Kock, 2006; Kock, 2005).

Social networking systems, for example, allow easy sharing of resources and many-to-many discussions on the shared resource. Moreover, social networking systems could support multiple resource formats, e.g., pure text, documents, videos. The comments to the resource itself are not necessarily limited to text, but could be, for example video themselves. However, in many cases, there are legal problems related to confidentiality. The use-case we discuss has such problems: suppose that a network of physicians wants to share information regarding their experience, perhaps involving video-recordings of a novel medical treatment. The data involved can be textual descriptions, spreadsheets, pictures (e.g., radiographies) or videos (e.g., recordings of successful or otherwise interesting medical treatments). In this scenario, attention must be paid to specific rules (usually varying from country to country) regarding privacy and confidentiality of information. Although many different social networking systems exist, they are typically focused on some specific data format (e.g., YouTube for video and Flickr for pictures). Moreover, privacy settings are not always flexible enough to allow fine-grained control and the fact that they are “public” and general-purpose networks negatively influences their perception as work tools. Eventually, most systems require their users to transfer some rights, and in our case the physicians may not be legally allowed to do so.

A possible solution would be creating a professional, ad-hoc and invitation-only OSN. The members would in practice become part of a virtual organization and the solution solves many issues present in the “general-purpose” online social network scenario: (i) the system can be designed to allow the required amount of granularity in accessing data, with full support for the legal requirements; (ii) such a network would clearly be perceived as “professional” and provides a no-frills environment where work-related discussions can take place; (iii) the system could easily support all the file formats of interest, plus any additional feature that the professionals require (e.g., an integrated spreadsheet).

Another problem that is easy to overlook is the actual size of the data that such a network contains and the costs associated therewith. In fact, such a system could not use low-quality videos, because losing information would be too risky. However, not only large videos occupy a large amount of storage, they also create non-trivial bandwidth problems. Video streaming-services typically use multiple facilities geographically distributed in order answer to each request from the most favorable location. Moreover, in peak moments they require huge data throughput, while in most other situations the requirements are rather modest. Creating such infrastructure from scratch requires experience and money.

On the other hand, Blogracy uses BitTorrent as underlying data transfer protocol. BitTorrent is optimized for large file transfer and is especially suitable for that kind of usage pattern: very popular files (peak moments) are extremely fast to access and its peer-to-peer architecture also solves the geographic distribution problem nicely.

Eventually, since all the data in Blogracy is strongly encrypted, it is rather easy to ensure that only the intended and allowed recipients are actually able to access the file. Since CP-ABE allows the creation of extremely flexible rules, it is entirely possible to implement legal requirements that vary according to the nations of people involved. The organizations are not heavily involved and the individual members are in charge of the decisions, effectively constituting a Virtual Team collaborating over the shared resources (Alshehri, Radziszowski & Raj, 2012).
Another issue that is worth mentioning is the requirement of rights transfer. The problem may not be solved easily, as it also has far-fetching implications with the actual ownership of the social networking system. In order for the online social network to operate, some organization need to provide the infrastructure and, unfortunately, would also have legal responsibility of the data. The organization actually managing the network can either be (i) one of the physician's organization or (ii) a specially created consortium. The creation of a consortium would solve most problems related to data ownership and rights, but unfortunately it constitutes a strong organizational and economical burden. On the other hand, if only one of the involved organizations (or a small subset of them) takes a leading role, it is not clear if the other ones would happily allow their employees to upload reserved data on another organization servers.

Since Blogracy is a peer-to-peer system, data is not stored on a dedicated machine (or group of machines) that would typically be under the responsibility or ownership of a given real organization. This property is especially desirable in the context of Virtual Teams, because it would be troublesome to transfer data ownership, partially or entirely, or even rights to the organization that would store the data on its servers.

The deployment of a Blogracy node does not pose particular technical challenges, since the project is entirely developed in Java and is multi-platform. Moreover, a single node can be configured to serve multiple users through a familiar web interface. On the other hand, being a completely distributed system, Blogracy relies only on the nodes themselves for making social content available to users. Since nodes in a file-sharing system are not required to be continuously connected, the files they host may be occasionally unavailable. In Blogracy, in particular, this may occur when a new user is introduced into the network and he has few followers who replicate his content. In this case, the propagation of updates from a new user may be delayed. For this reason, it is recommended that a Virtual Team using the Blogracy infrastructure deploys at least a single node with high uptime. This node may be a user's node, or a node launched specifically for facilitating the distribution of content from new users.

**RELATED WORK**

While many authors argue for the distribution and openness of social networking and micro-blogging services, few usable implementations exist, either in the field of federated networks or as fully distributed solutions. Among the federated social networking systems, one of the best known is **Diaspora** (http://joindiaspora.com/). 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. Another attempt in a similar direction is **StatusNet** (formerly known as Laconica, http://status.net/). It adheres to the OStatus standard protocol for the interconnection of various servers. 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** (Clarke 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, Freenet uses an anonymizing layer based on a mix-net scheme.

Another system for anonymous publishing is **OsirisSPS** (Serverless Portal System, http://www.osiris-sps.org/). It is a framework for creating community-based sites and forums, distributed over a peer-to-peer network and a Kademlia DHT, supporting (i) content replication, (ii)
anonymity, (iii) signature and (iv) encryption of messages. The distributed management of reputation and the possibility to realize so-called Anarchist portals are the most distinguishing aspects of the program.

Specifically in the field of social networking, various systems are being developed on the basis of peer-to-peer communications and DHT indexing. Among these, **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.

**LotusNet** (Aiello & Ruffo, 2010) is a model of a social network to be built over Likir. Likir itself is an implementation of a secured Kademlia DHT, which requires a user to be authenticated before participating in the network. During the registration phase, the user's private key is calculated by a globally trusted Certification Authority, on the sole basis of the user's identity (e.g., OpenId or email address) and a unique private master key. The corresponding public key can be generated by everyone, from the user's identity and the public master key of the system.

**Safebook** (Cutillo, Molva & Strufe, 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.

**LifeSocial** (Graffi et al., 2010) is a prototype developed over FreePastry for DHT indexing and PAST for data replication. It is composed of various mandatory modules, for managing profile, friends, groups and photos. Additional modules are available for chat and whiteboard functionalities.

**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.

**CONCLUSION**

Although the current approach to build Online Social Networks is to create huge centralized systems owned by a single company, such strategy has many drawbacks. For inter-organizational collaboration, the main issues regard: (i) the privacy of users, (ii) the control of published content; (iii) the interoperability with other systems; and (iv) the costs of the infrastructure, especially when it has to support media files. In order to overcome such drawbacks, many different approaches have been proposed, both as federated or peer-to-peer systems.

In this paper we proposed a novel peer-to-peer system for social networking that leverages existing, widespread and stable technologies such as DHTs and BitTorrent. Although the primitives offered by those technologies were created with other goals in mind, nevertheless they could be used with minor modification in our system. In particular, we introduced a key-based identity system and a model of social relations for distributing resources efficiently among interested readers.

Although Blogracy is not yet feature-complete, we created a working prototype, implementing all core functionalities as a layer over a well-tested distributed file sharing system. As described, we plan to extend it further. Areas for study and improvements include: (i) privacy, where recently proposed decentralized cryptographic algorithms based on certified attributes could improve extensibility and flexibility of disclosure policies; (ii) cooperation mechanisms, for allowing smooth introduction of new users into the network; (iii) distributed reputation mechanisms, based on social relations, for
strengthening also the basic functionalities of the underlying peer-to-peer infrastructure; (iv) autonomous direct recommendations; (v) personalization of results; and (vi) trust negotiation mechanisms.

REFERENCES


