Social Monitoring and Social Analysis in Internet of Things Virtual Networks

Orfefs Voutyras, Panagiotis Bourelos, Spyridon Gogouvitis, Dimosthenis Kyriazis, Theodora Varvarigou
National Technical University of Athens
Athens, Greece
orfeasvoutiras@gmail.com

***Abstract*—The integration of social networking concepts into Internet of Things systems is a burgeoning topic of research that promises to support novel and more powerful applications. In this paper we present the social approach that the COSMOS project introduces in order to achieve enhanced services like discovery, recommendation and sharing between Things enriched with social properties. We investigate how typical notions and modes of interactions of social networking can be extended to the networks of Things, providing a Social Internet of Things platform, and we discuss two main components supporting the socialization of Things; Social Monitoring and Social Analysis. The first one involves all the main tools and techniques needed for the monitoring of the social properties of the Things, whereas Social Analysis is used for the extraction of their complex social characteristics, as well as models and patterns regarding their behavior and relations between them.**

Keywords—Internet of Things; Social Network Analysis; Social Internet of Things

I. INTRODUCTION

The Internet of Things (IoT) will exponentially increase the scale and the complexity of the formed networks. The world of trillions of Things and the different administrative domains on which they operate, require new approaches that will make Things able to cooperate in an open and reliable way. Taking into consideration the rate at which IoT devices are deployed and used in different applications, one of the main challenges refers to the efficient and optimized management of these entities. Existing centralized management approaches are proven inefficient when applied in systems with a huge number of Things or not applicable due to communication problems, something that highlights the need for distributed management approaches. Moreover, approaches are required that will allow management decisions to incorporate situational awareness and propose management actions based on them. Finally, an additional challenge with respect to IoT management relates to the autonomous reasoning of Things on a context-aware basis. Autonomous management will integrate different types of knowledge (e.g. device-specific, situational, application-specific, administration-related etc) and trigger decisions accordingly. Autonomicity depends strongly on how much situational-aware, cognitive, smart and social the Things are [1]. Smart objects able to communicate and to discover their situation are already available, while various proposals aimed at giving social-like capabilities to Things exist [2]. However, the IoT vision can be fully achieved only with actual social networks that allow Things to cooperate in an open and reliable way and guarantee the effectiveness and scalability of the system.

To this direction, in order to overcome the aforementioned inefficiencies, the COSMOS project [3] will provide a framework for the decentralized and autonomous management of Things based on service-, interaction-, location- and reputation-oriented principles, inspired by social media technologies. Following the IoT-A reference model [4], the architecture supports real-virtual world integration by representing Things of the real world via their counterparts in the Cyberworld: Virtual Entities (VEs). VEs acquire perception through accessing sensor readings and can impact their environment or undertake physical actions using actuators via IoT-services. Moreover, they may have their own goals and be equipped with an internal logic in order to achieve them. VEs may form groups, called Groups of Virtual Entities (GVEs), that aggregate a potentially large number of them. GVEs can embed other GVEs too, resulting in communities where social behavior is more than necessary. Like VEs, GVEs have their own properties, based on properties of embedded individual VEs, and their own objectives (often management and optimization functions).

VEs and GVEs may interact for various purposes, such as collaboration (sharing a common goal), cooperation (getting help from other VEs in order to achieve specific objectives), advertising of their properties/attributes, offering actuation services etc. In other words, they have to operate as social actors and have a set of dyadic ties between them. Thus, it is of major importance to enhance the VEs with the key features of a social intelligent entity. A VE that has social characteristics can discover and provide services in its social networks. It can acquire knowledge through various means, such as learning from experience, and can reason with knowledge to make plans, explain observations etc. Ontologies were chosen to describe VEs and GVEs, as they provide a rich vocabulary for the general domain knowledge. Moreover, formalization of ontologies improves retrieval, similarity adaptation and learning [5].

Our approach supports the definition and the establishment of social properties and relations between VEs and provides the functionality required to form a social network following the Social Internet of Things (SIoT) paradigm [6].

It should be mentioned that during our analysis, we keep separate the two levels of people and Things, thus allowing Things to have their own social networks and humans to impose rules to protect their privacy and to access only the result of autonomous Things interactions occurring on the virtual social world. Socially-enriched coordination considers the role and participation scheme of VEs in and across networks. Management decisions and runtime adaptability is based on Things security, trust, location, relationships, information and contextual properties. Extended complex event processing and social media technologies extract only the valuable knowledge from the information flows, while workload-optimized data object stores facilitate efficient storage by also exploring the interplay between storage and analytics on networks of data objects.

The social network perspective provides a set of methods for analyzing the structures of whole social entities and their networks. For the study of these structures, we use Social Network Analysis (SNA) [7] to identify local and global patterns, locate influential entities and examine network dynamics. To this direction, in order to manage social relations and interactions between the VEs, we introduce into the COSMOS control loop [8] two components: Social Monitoring and Social Analysis.

The rest of this paper is organized as follows: Section II describes our proposed architecture. Section III identifies social relations and introduces the Social Monitoring component. Section IV discusses the issue of social links establishment and proposes recommendation mechanisms and criteria. Section V analyzes some further functionalities of the Social Analysis component. Section VI presents some suitable computational models and social network tools. Finally, Section VII concludes the paper.

II. Our Architecture

From our point of view, there are three elements that would justify the characterization of a platform as a SIoT platform:

- it maps the social relations and interactions of the individuals, companies etc. to their VEs,
- it defines, monitors and exploits social relations and interactions between the VEs,
- it uses technologies and exploits services from the domain of social media.

In this paper, we mainly focus on the second element. In this direction, we define social properties and relations between VEs. The COSMOS social ontology [9] is used to enrich the VEs description with social properties like VEs' Domain, Location, Trust, Reputation etc. It contains all the relevant to the VEs data that can be used to infer social connectivity between them and evaluate their performance upon request from any VE. This way, the platform is always informed of changes to the social characteristics of the VEs and dynamic social behavior is achieved. In order to exploit these social properties and behaviors, we develop components and mechanisms that support the socialization of VEs.

The **Profiling and Policy Management (PPM)** component assigns a unique ID to a VE and enables the entry of all the information needed for the description of the physical entity through the domain ontology of the corresponding VE. Moreover, it enables the owner to determine the social "openness" of the VE: the IoT-services that can be used by other VEs, the kind of experience that can be shared, the sets of VEs which can access such information etc. However, the "openness" of VEs is affected by the social selfishness, a basic attribute of human beings. Thus, the concept of Opportunistic loT [10] should be taken under consideration.

The need for effective and decentralized discovery of experience and IoT-services brings us to the most important social concept that has to be implemented: *friendship between VEs*. This relationship will be a guideline, a road map of communication, as each VE will maintain a group of VEs (a friend list) which have been deemed to be in a position to provide or receive help from it. Friendship is considered as a non-mutual relationship, which means that the concept of "Friends" matches this of Twitter "Followers" rather that of Facebook "Friends". The choice of Friends is based on other social properties included in the COSMOS social ontology like VEs' Domain, Location, Trust, Reputation etc.

The **Friends Management (FM)** component is responsible for creating and maintaining the list of friends that a VE has. In other words, it allows VEs to initiate, update and terminate their friendship with other VEs on the basis of the owner's control settings. It provides the owner with the option of setting new friends to his/her VEs, offers friend-recommendation request services and monitors the friend list of a VE regularly or on demand.

In order to achieve self-management and autonomicity we follow the MAPE-K model [11], as we estimate that it is very close to the nature of the IoT management. Each VE maintains its own Knowledge Base (KB), part of which is a Case Base (CB) [12] where experience can be found. Storage of experience in a local or central KB [13] depends on whether the individual's knowledge needs are constant or opportunistic. Such a categorization of needs will be primarily based on the "domain" membership of individual VEs as well as technical limitations that may be present. A KB can be shared between VEs with suitable social characteristics, something that improves the decision making mechanisms. Moreover, VEs representing weak devices that do not have their own KB can take advantage of the KB of their social group. However, by adapting the social view of the Things, we extend the MAPE-K loop by introducing two new components: Social Monitoring (SM) and Social Analysis (SA).

The **Social Monitoring (SM)** component contains all the main tools and techniques that are used for the monitoring of the social properties of the VEs. Its main objective is to collect, aggregate and distribute monitoring data (events) across the decision making components (Planners) of the collaborating groups. The events are generated by interactions in response to - directly or indirectly - user actions (e.g. registering a new VE) or VEs' actions (experience sharing). The Social Monitoring "feeds" the VE Registry and forwards its data to the Social Analysis component.

The **Social Analysis (SA)** component, based on the results of the Social Monitoring component and taking advantage of Social Network Analysis (SNA), is used for the extraction of complex social structural characteristics of the VEs (e.g. centrality), as well as models and patterns regarding the behavior of the VEs and the relations between them. These social properties and relations that will be extracted could be used by the components of the project platform or even offer services directly to individuals.

III. Social Relations & Monitoring

Instead of trying to map various social relations and characteristics of the real world to the IoT, a more concrete approach would be to identify the various interactions that could exist between VEs. Inspired from the social media domain, examples of these relations that are monitored are:

Followees: The friends that are being tracked by a specific VE. The friends list (from now on **Followees List**) defines the receivers of the VE's search requests.

Followers: The VEs that track a specific VE. They are held in the **Followers List** and indicate the credibility and reputation of a VE. Moreover, their number can act as a rough indicator of the frequence of search requests from other VEs.

Groups: To how many GVEs a VE belongs and how many VEs and GVEs (separately and in total) a GVE contains. A VE belonging to many GVEs gives us a measure of its "centrality" in the whole system and the real world, whereas the members of a GVE provide the main data needed to grasp its size and complexity.

The relation between a VE and its Followees is trust-based and non-mutual (low reciprocity [14]). This means that VE1 may use the experience of VE2, but on the other hand, VE2 may not do the same for VE1. A VE (trustor) trusts blindly its Followees (trustees) and requires access to their experience.

On the same spirit, examples of interaction metrics that are used are:

Shares: How many times a VE has shared its knowledge with other VEs. This value is used as an indicator of the reputation of the VE. However, for a more valuable evaluation, the number of followers, the amount of the shareable resources and the number of the received requests should be taken under consideration.

Mentions: How many times an IoT-service of a specific VE is mentioned in the Case Base of other VEs. This is another indicator of the reputation of a VE.

Applauses: How many times the social shares have been regarded as useful from the receivers. This value could be used as an indicator of the trustworthiness of the VE. This is a quite important property but rather difficult to monitor, compared to other elements, as feedback is needed.

The interaction metrics (Shares, Mentions and Applauses) are maintained in the Followees List. In other words, the Friends Management component is responsible to update them. They are calculated in a distributed manner by the VEs on a per-VE basis and are the main input for the Social Analysis that will follow. For each metric identified we develop/choose the corresponding Key Performance Indicators (KPIs) and tools that should be imported into the VE during the phase of registration. Based on the chosen configuration, different levels of reporting granularity would be possible in order to keep the monitoring tasks as light-weight as possible but still to be able to perform in depth analysis whenever needed. The events that are generated at the Social Monitoring level can be evaluated at different platform levels (node level, group level or system wide) against a set of rules. The rules, which can be added, deleted or updated at runtime, may be specified by the consumers of information to set and control the flow of events and the aggregation output.

All the above are a good starting point for social characterization and accelerate the discovery mechanisms as they reduce significantly the target groups. It should be mentioned that a quite complex approach is used, as we can monitor how many times two specific VEs have (succesfully) exchanged experience with each other. Instead of just monitoring that "VE1 shared its experience 1302 times", if needed, we can know that "VE1 shared its experience with VE2 203 times, with VE3 523 times…". It is obvious that in order to choose the best set of relations we need to study various types of representative applications and observe the interactions involved.

The aforementioned metrics feed the social properties of VEs, as they are expressed by the COSMOS social ontology. More specifically:

The **Reputation Index** is a property which indicates the total Shares and Mentions a VE has. It is a cumulative and comparative indicator.

The **Trust Index** of a VE is a property which states how many times a VE has successfully shared its CB and can be calculated as the ratio of the total Applauses from its Followers to the total Shares to its Followers. Coupled with the concept of feedback and through refinement of its calculation, we can use this index as a means to simulate social mobility in the platform, as Trust will be one of the most important components of friendship recommendation.

The **Reliability Index** is an absolute indicator of the performance of the Physical Entity that quantifies its efficiency to complete successfully the experience-sharing mechanism relatively to its ideal or normal operation.

Our main goal is to combine Reputation, Trust and Reliability and express them by only one social measure, the **Dependability Index** of the VE. This measure, which is further discussed later, is a crucial indicator that will lead a VE to take decisions regarding the selection of new experience or new Followees.

The relationship between Followers and Followees can be better understood by the following mechanism: Whenever a VE decides to initiate the decentralized discovery mechanism of experience, it targets its Followees List in its KB. The number of its Followees defines the depth of communication. That is important mainly because of the recursive way the discovery mechanism works, meaning that if a Followee of the original VE does not locate a suitable answer in its own KB, it will check the depth required (mentioned as TTL/time-to-live of the query) and initiate a new version of search, this time directed at its own Followees. Therefore, brokers are dynamically designated taking into account that Followees are willing and able to act as such for their respective Followers. By defining an appropriate upper limit for the ttl, the recursive nature of the discovery mechanism does not cause the social equivalent of an infinite network loop. The corresponding mechanism works by using the number of the specific VE's Followees and a number indicating the maximum population coverage that the request should reach. Following the theory of 'six degrees of separation' [15], the output is a number between one and six and indicates the maximum search depth that this VE can initiate. Therefore, any requests received by this VE will be propagated by suitable downwards modification of the incoming ttl number.

IV. Social Links Establisment

Trust among a VE and its Followees is a direct relationship. However, a VE can trust unknown VEs based on the recommendation of its Followees or the recommendation of the SA.

A. Followee Acquisition

The process of Followee acquisition begins at the phase of registration. The user can manually set the Followees List of the VE (e.g. in order to link his/her VEs with each other). This is the most basic way a VE forms social bonds. Such friends will have a number of benefits during the social monitoring or discovery phases (e.g. greater priority). Recognizing the opportunity of a malicious user trying to create a Trust block and therefore create imbalances in the social network through collusion, the platform takes into account the specific social characteristics of the registering VE and adds random suitable VEs in its Followees List.

Another way of acquiring Followees is through a discovery mechanism, which is always based on recommendation. Discovery through recommendation is more reliable and provides protection from malicious behavior. New Followees can be recommended to a VE by its current Followees or by the SA component.

In the first case, transitivity is used (e.g. a VE1 recommends to VE2 its own Followees as new Followees). After the VE acquires a number of recommended Followees, it asks the SA component for their Dependability Indexes. The SA component calculates the indexes, as it will be discussed later, and forwards the result back to the VE. Finally, the FM component of the VE, based on the thresholds set by the user, decides whether it will accept the new recommendations or not.

In the second case the VE sends a Followee recommendation request to the SA component. Practically, this leads to the renewal of its Followees List. The VE's FM sends the request, passing as parameters weights for calculating the Dependability Index, a minimum acceptable limit of its value and the current Followees List. The SA calculates the Dependability of the Followees, based on the above input. If the new indexes are bellow the limit, the SA purges these VEs from the list, replacing them with more reliable ones, and a new Followees List is returned. Followees that have been set by users are not thrown away, but they are isolated.

B. Recommendation Criteria

The choice of suitable Followees is based on three composite criteria: **Relevance**, **Dependability** and **Structural Power**. Relevance aggregates the concepts of Homophily (Domain and Physical Entity attributes in our social ontology) and Distance Proximity (Location & GeoLocation), Dependability refers to Reputation, Trust and Reliability, while Structural Power is evaluated by several structural network characteristics. Our goal is to combine all these criteria in order to obtain the "**Social Power**" of a VE [16].

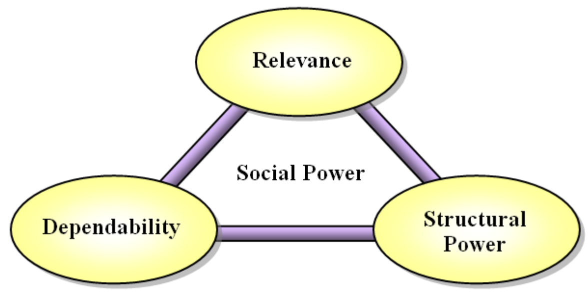

Fig. 1. The Social Power of a VE.

1) Relevance/Similarity

The basic criterion for choosing a new Followee, which will most probably have useful cases in its CB, mainly depends on its similarity to the VE, regarding both its nature and its environment. As a result, two properties have to be studied:

Homophily/Heterophily: the degree to which VEs and GVEs form ties with similar/dissimilar others. Similarity can be based on domain-dependent social characteristics of the VEs as they are described in their social ontology. Homophily leads to the formation of homogeneous groups, where the relationships are easy to build. However, for VEs associated with more domains, heterophily is desirable. These two aspects aim at providing more beneficial friend recommendation services. For example, the homophily value is taken under consideration from friend recommendation services when the sharing requests concern experience. In the case of requests concerning IoT-services sharing, the opposite characteristic, the heterophily value, is extracted. In general, various recommendation algorithms could be developed incorporating both values [17].

Propinquity: The tendency for VEs to have more ties with geographically close others. The mobility of Things should be taken under consideration [18].

2) Dependability

On the subject of the calculation of the Dependability Index, the developed solution is a platform specific service that is initiated by the SA component and entails the querying of Followers of a specific VE. We refer to this VE as "Evaluated VE". The first action of the SA is to acquire the Followers List of the Evaluated VE. The SA extracts the group of Followers of the EvaluatedVE and then randomly decides which ones and how many of them to use as a querying basis (if their number is too great). This element of randomness is essential in the development of the mechanism, as it can prevent collusions which may alter the final result of the evaluation process. After this step, the SA requests the stored Applauses, Mentions and Shares for the EvaluatedVE from the Followees List of each Follower of the VE. After receiving the requested metrics, the SA component calculates both the Trust and the Reputation Indexes which, combined with certain weights defined by the user, result to the Dependability Index. It should be noted that for the calculation of the Dependability, the EvaluatedVE itself does not provide any information at all, but instead, all the information needed is offered by its social environment.

3) Structural/Social Analysis Properties and Metrics

As we already discussed, the analysis and formalization of social relations among Things is critical and the computation of social measures is necessary. They improve the effectiveness of the system as they help in taking decisions at many levels. However, the analysis of social networks is basically related to their structural analysis. Social network analysis is the analysis of social networks viewing social relationships in terms of network theory. These relationships are represented by nodes (individual actors within the network) and ties (relationships between the individuals). The computation of structural power is considered to be a step of high importance.

At a first level, it is important to refer to some SNA properties and discuss the way they can help us identify the key nodes, the relationships strength (ties strength) and the cohesion of our social networks. There is a great variety of metrics that could be used under the functionalities of the Social Analysis, offering more detail and information about the networks being analyzed. The main metrics that have been identified and whose role is evident even at the early steps of the Social Analysis component are:

Centrality: Centrality refers to a group of metrics that aim to quantify the importance or influence (in a variety of senses) of a particular VE or group of VEs within the network. Examples of common centrality metrics include degree centrality, closeness centrality, betweenness centrality, eigenvector centrality, alpha centrality etc [19]. Centrality is one of the main metrics that should be taken under consideration from the recommendation services.

Distance (Shortest Path): The minimum number of ties required to connect two particular VEs, as popularized by Stanley Milgram's small world experiment and the theory of 'six degrees of separation'. This theory introduces the idea that each node in a freely emerged network can be reached by propagating items of information via six hops. Distance is important as it is involved in various other measures, e.g closeness centrality and betweenness centrality. Closeness centrality is distance-based and increases when the distance between nodes decreases, while betweenness centrality is a measure of the number of shortest paths in a network that traverse the node.

Tie Multiplexity: The number of content-forms contained in a tie of a dyad of VEs, in other words how many relationships represents a tie. For example, two VEs that can share knowledge will have a tie with multiplexity of 1, whereas, in case they can share IoT-services too, they will have a tie with multiplexity of 2 and so on. Multiplexity is associated with relationship strength and durability and may be an indicator of network effectiveness. Some other kinds of relationships that can be defined and affect the multiplexity of ties are presented in [6] and are the Parental object relationship, the Ownership object relationship, the Co-location object relationship etc.

Cohesion: The degree to which VEs are connected directly to each other by cohesive bonds. Structural cohesion refers to the minimum number of members who, if removed from a group, would disconnect the group. This characteristic is quite important when reconfiguration of a group of VEs must take place.

Density: The proportion of direct ties in a network relative to the maximum possible number of them. When density is close to 1, the network is dense and can resist to tie failures more easily, otherwise it is sparse. For density 1 the network is called a clique. Density is related to the speed with which information is diffused among the actors and is useful in comparing networks or different regions of a single network.

Centralization: An aggregate metric that characterizes the amount to which the network is centered on one or a few important nodes.

Clustering coefficient: A measure of the likelihood that two randomly selected neighbors of a node are connected to each other. It represents the density of a node's neighborhood. A higher clustering coefficient indicates a greater 'cliquishness' [20]. For an entire network it is computed as the average of all its nodes' clustering coefficients and represents the tendency to form clusters and groups.

Structural holes & Bridges (Mediators): The structural hole concept, developed by sociologist Ronald Burt and sometimes called social capital [21], refers to the lack of ties between two parts of a network. The structural holes theory introduces the concept of bridges or mediators as individual actors that fill a structural hole, thus connecting two previously unconnected (or at least loosely coupled) actors and gain valuable insights in others work. Finding and exploiting a structural hole can offer novel and competitive innovation opportunities [22]. In our approach, mediators could be used to facililate the communication between VEs and GVEs. Mediators can influence partners and build high reputation.

Other network level analysis metrics include average distance (average distance between all pairs of nodes), metrics that integrate attribute data with network data (for example, metrics that measure homophily) etc.

The structural metrics discussed above give us the opportunity to identify the key VEs in the network. The centrality metrics bring out the nodes of great 'importance'. For example, the degree centrality is a measure of a VE's connectedness and represents the number of friends it has in its neighborhood. The beetweness centrality (BC) is a measure of how often a VE is the most direct route between two other VEs and represents its potential to act as mediator. The closeness centrality (CC) represents how fast a VE reach every other in the network, whereas the eigenvector centrality (EC) represents how well a VE is connected to other well-connected VEs.

Of course, the importance of a VE depends on the context and use case. For example, in the home automation domain, high degree centrality may indicate the key room in a building, in the smart city domain, VEs of high betweeness that bridge various communities may be of great importance, while in the transport domain, VEs of high eigenvector that influence the whole network may be of greater significance (Table 1). The various metrics are correlated and they do not necessarily have the same tension. A VE with high eigenvector may not have high closeness and/or high betweeness, meaning that it does not have the greatest local influence and/or it has low bridgering potential.

TABLE I. TYPES OF CENTRALITY AND VES' ROLES

	Metrics Interpretation		
	High BC	***High CC***	***High EC***
VE Role	Mediator	Group Leader	Network Influencer

Reciprocity and multiplexity metrics bring out the tie strength. However, the aspects of heterogeneity, such as diversified bonds and/or dissimilar nodes, add more complexity. SNA could be enhanced by the use of tie weights (weighted network). Ties weighted in relation to frequency of interaction, influence, capacity etc. provide a more real world indication of the dynamics of a particular network. They affect the path that information takes, the speed it travels and may characterize the nodes that use them as more or less key ones. Thus, centrality measure results are affected by tie weightings. Moreover, centrality is affected by ties between dissimilar VEs (multimodal network). A metric such as betweenness centrality applies to uni-modal networks and there is no clear definition in a multimodal one. All these mean that new algorithms are required for the calculation of centrality for weighted and multimodal networks.

At this point, it should be noted that, while some VEs relationships will be defined by the users, a great percentage of them will have to be appointed by the COSMOS platform. As a result, the case we study here is not just of an independent system from which some metrics are extracted for research, but instead a system where many times new relationships are appointed dynamically depending on known desired values of these metrics. In other words, instead of extracting the properties of an existing graph, we create a graph based on the desired values of the properties. Consequently, the creation and maintenance of the VEs' social environment is reduced to determining the proper metrics and their corresponding values.

In general, VEs social networks will be self-organized, emergent and complex [23], such that globally coherent patterns will appear from the local interaction of the elements that make up the system. These patterns will become more apparent and rich as the size of the network increases. However, a global network analysis of all the relationships between millions or billions of VEs is not feasible and is likely to contain so much information as to be uninformative. The nuances of a local system may be lost in a large network analysis, hence the quality of information may be more important than its scale for understanding network properties. Thus, social networks should be analyzed at the proper scale, depending on the application or the needs of a user or a functional component of the platform. Generally, there are three scale-levels into which networks may fall: micro-level, meso-level and macro-level [24]. At the micro-level, social network research typically begins with tracing inter-VE interactions in a small group of a particular domain. Meso-level analysis begins with a population size that falls between the micro- and macro-levels and may also refer to analysis that is specifically designed to reveal connections between micro- and macro-levels. At last, macro level analysis generally traces the outcomes of interactions over a large population. It becomes quite evident that, in case we have to take decisions at system level regarding e.g. the reconfiguration of some entities, this kind of analysis becomes really important.

V. OTHER FUNCTIONALITIES OF THE SOCIAL ANALYSIS COMPONENT

The services and functionalities of the Social Analysis component will be used by both the users (External use) and other functional components (Internal use). From the plethora of the metrics available and the social interactions that can be monitored, it is quite evident that the Social Analysis component can provide a great number of functionalities, depending on the needs of the system. Briefly, the functionalities that have already been presented in the paper are:

Computation of Dependability Index

Recommendation of VEs: By finding the similarities between VEs or identifying the needs of a specific VE, it is possible to provide many recommendation services. One representative example is the friend-recommendation service (Fig. 2).

Extraction of structural characteristics of the networks: There are many properties of the networks that could be analyzed without direct modelling and could be of great use for recommendation services. Questions that could be addressed are whether there is any "leak of knowledge" from one team/cluster to another, if so, how fast that knowledge flows, whether a team has any weak points that can be structurally overcome etc. A representative example is the discovery of structural holes. Networks rich in structural holes are a form of social capital in that they offer information benefits. COSMOS could make recommendations to fill in these structural holes and exploit the social capital.

Some other of the main functionalities that have been studied and we work on are the following:

Extraction of relational-models: Another functionality of the Social Analysis component is the extraction of Relational Models. Such models are the Communal Sharing (behaviors of VEs with collective relevance e.g. a service offered by an entire swarm of VEs), Equality Matching (VEs operate as equals and request/provide information among them in the perspective of providing IoT-services to users while maintaining their individuality), Authority Ranking (established between VEs of different complexity and hierarchical levels) and Market Pricing (VEs working together in the view of achieving mutual benefit and participating in this relationship only when it worths doing so) [25].

Modelling and Visualization of networks: Visual representations of social networks help to understand features of the network that are not easily identifiable and convey the result of the analysis. Collaboration graphs are used to illustrate the quality of relationships between VEs based on characteristics such as the evolution of their Trust and Reputation. Moreover, the network propagation modelling can be included in this functionality.

Extraction of higher-level goals of VEs: A desired feature is the comparison of the same targets/goals of the VEs and the extraction of more abstract goals that will characterize certain groups, based on homophily and propinquity metrics.

User attribute and behavior analysis: COSMOS could support activities such as customer interaction and analysis, by analyzing the results of monitored interactions such as the popularity of the IoT-services or the Case Base of a VE among the users. The prediction of the potential demands of people regarding new services and knowledge is an interesting topic of research [26].

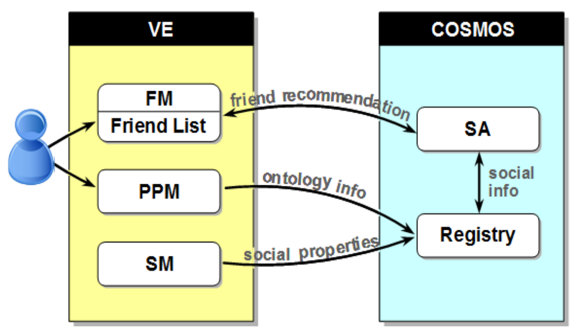

Fig. 2. Management components in the COSMOS architecture.

VI. Computational Models and Social Network Tools

There are many tools that can be used from the Social Analysis component and a need for the development of new tools may exist. Social network and dynamic network metrics, trail metrics, procedures for grouping nodes, identifying local patterns, distance based, algorithmic and statistical procedures for comparing and contrasting networks, groups and individuals from a dynamic meta-network perspective, geo-spatial network metrics, identification of key players, groups and vulnerabilities, are but a few issues that have to be addressed.

Some of the main tools that we take under consideration are:

- **Representation Formats, markup languages and ontologies:** DyNetML and GraphML could be used as a reference model.
- **Dynamic Network Analysis and Social Network Analysis:** Tools for reasoning under varying levels of uncertainty about dynamic networks, their vulnerabilities and their ability to reconfigure themselves, choosing Dynamic Network Analysis (DNA) metrics and then using one or more of the available optimizers to find a design that more closely meets an ideal as well as exploring network graphs. The dynamic network visualization has been a challenging topic due to the complexity introduced by the extra dimension of time [27]. Some tools that have been studied are ORA, IGraph, Networkx and Pajek [28],[29].
- **Network Document Analysis and Data Entry:** Tools that enable the extraction of information from texts using Network Text Analysis methods and other techniques. A typical tool that has been studied is SocIoS [30].

VII. Conclusion

The COSMOS platform can be characterized as a SIoT platform since it defines, monitors and exploits social relations and interactions between the VEs and uses technologies from the domain of the social media. The Social Monitoring and Social Analysis components improve the mechanisms used to establish social links that facilitate cooperation and enable selective sharing of knowledge and IoT-services through recommendation services. To sum up, in this paper we go beyond the state of the art by:

- identifying and establishing social properties and relations between VEs in such a way that the resulting social network is effectively manageable;
- describing a decentralized IoT architecture which supports the functionality required to form a social network following the Social Internet of Things paradigm. We discuss some services and mechanisms, like distributed relations management and decentralized discovery mechanisms.
- studying social analysis metrics and properties and identifying roles for the VEs and desired functionalities for our SA component.

Acknowledgment

The research leading to these results is partially supported by the European Community's Seventh Framework Programme under grant agreement n 609043, in the context of the COSMOS Project.

References

[1] D. Kyriazis and T. Varvarigou, "Smart, Autonomous and Reliable Internet of Things", 4th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN), 2013.

[2] Atzori, L., Iera, A. and Morabito, G .2014. From 'Smart Objects' to 'Social Objects': The Next Evolutionary Step of the Internet of Things" IEEE Communication Magazine, Networks, vol. 52, no. 1, 2014, pp. 97–105.

[3] COSMOS project: http://iot-cosmos.eu/

[4] IoT-A: http://www.iot-a.eu/public

[5] N. Guarino, “Formal ontology and information systems”, Formal Ontology in Information Systems, FOIS 98, pages 3–15, Trento, Italy, June 1998. Ios Press.

[6] L. Atzori, A. Iera, G. Morabito, and M. Nitti, “The Social Internet of Things (SIoT) – When social networks meet the Internet of Things: Concept, architecture and network characterization”, Computer Networks, Volume 56, Issue 16, 14 Nov. 2012, pp. 3594-3608.

[7] S. Wasserman and K. Faust, “Social Network Analysis: Methods and Applicati ons”, Structural Analysis in the Social Sciences, Chambridge Univeristy Press, 1994.

[8] O. Voutyras, S.V. Gogouvitis, A. Marinakis, T. Varvarigou, “Achieving Autonomicity in IoT systems via Situational-Aware, Cognitive and Social Things”, 18th Panhellenic Conference on Informatics, PCI 2014.

[9] O. Voutyras, P. Bourelos, D. Kyriazis, and T. Varvarigou, “An Architecture supporting Knowledge flow in Social Internet of Things systems”,10th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), 2014, pp. 45-50.

[10] A.R.M. Teutle, “Twitter: Network Properties Analysis”, 20th International Conference on Electronics, Communications and Computer (CONIELECOMP), 2010, pp. 180-186.

[11] “An architectural blueprint for autonomic computing.”, IBM, Autonomic Computing White Paper, June 2005, Third Edition

[12] Z. Budimac and V. Kurbalija, “Case Based Reasoning – a short overview”, Second International Conference on Informatics and Information Technology, pp. 222-233.

[13] V. Supyuenyong and N. Islam,. “Knowledge Management Architecture: Building Blocks and Their Relationships”, Technology Management for the Global Future, PICMET 2006 (Volume: 3).

[14] A. Falk and U. Fischbacher, “A theory of reciprocity,” Gamesand Economic Behavior, vol. 54, no. 2, 2006, pp. 293-315.

[15] J. Guare, “Six Degrees of Separation: A Play”, Vintage Books, 1990.

[16] K. S. Cook, R. M. Emerson, M. R. Gillmore, and T. Yamagishi, “The Distribution of Power in Exchange Networks: Theory and Experimental Results”, American Journal of Sociology, Vol. 89, No. 2 (Sep., 1983), pp. 275-305.

[17] H. Al-Qaheri, S. Banerjee, and G. Ghosh, “Evaluating the power of homophily and graph properties in Social Network: Measuring the flow of inspiring influence using evolutionary dynamics”, Science and Information Conference (SAI), 2013, pp. 294-303.

[18] H.Z. Asl, A. Iera, L. Atzori, and G. Morabito, “How often social objects meet each other? Analysis of the properties of a social network of IoT devices based on real data”, Global Communications Conference (GLOBECOM), 2013 IEEE, pp. 2804-2809.

[19] D.A. Bader and K. Madduri, “Parallel Algorithms for Evaluating Centrality Indices in Real-world Networks”, International Conference on Parallel Processing (ICPP), 2006, pp. 539-550.

[20] A.R.M. Teutle, “Twitter: Network Properties Analysis”, 20th International Conference on Electronics, Communications and Computer (CONIELECOMP), 2010, pp. 180-186.

[21] R. S. Burt, “Structural holes and good ideas”, American Journal of Sociology, 110 (2004), pp. 349–399.

[22] E. Zhang, G. Wang, K.Gao, X. Zhao, and Y. Zhang, “Generalized structural holes finding algorithm by bisection in social communities”, Sixth International Conference on Genetic and Evolutionary Computing (ICGEC), 2012, pp. 276-279.

[23] R. Albert, AL Barabasi, “Statistical mechanics of complex networks”, Reviews Of Modern Physics, vol. 74, 2002.

[24] D. Knoke, “Social Network Analysis”, Quantitative Applications in the Social Sciences series, SAGE Publiations, Second Edition, 2008.

[25] M.W. Baldwin, “Interpersonal Cognition”, The Guildford Press, 2005.

[26] K. Yogo, A. Kida, R. Shinkuma, T. Takahashi, H. Kasai, and K. Yamaguchi, “Extraction of Hidden Common Interests between People Using New Social-graph Representation”, 20th International Conference on Computer Communications and Networks (ICCCN), 2011, pp. 1– 6.

[27] L. Shi, C. Wang, and Z. Wen, “Dynamic Network Visualization in 1.5D”, Pacific Visualization Symposium (PacificVis), 2011 IEEE, pp. 179-186.

[28] JM. Huisman and M.A.J.van Duijn, “A reader's guide to SNA software”, The SAGE Handbook of Social Network Analysis, J. Scott and P.J. Carrington Editions, 2011, pp. 578-600.

[29] N. Akhtar, “Social Network Analysis Tools”, Fourth International Conference on Communication Systems and Network Technologies (CSNT), 2014, pp. 388-392.

[30] SocIoS project: http://www.sociosproject.eu/