

# An Sna-Bi Based System for Evaluating Virtual Teams: A Software Development Project Case

Lamia Ben Hiba<sup>1</sup> and Mohammed Abdou Janati Idrissi<sup>2</sup>

<sup>1</sup>Equipe TIME, ENSIAS, Mohammed V Souissi University, Rabat, Morocco

<sup>2</sup> Mohammed V Souissi University, Rabat, Morocco

## **ABSTRACT**

*The dependence of today's collaborative projects on knowledge acquisition and information dissemination emphasizes the importance of minimizing communication breakdowns. However, as organizations are increasingly relying on virtual teams to deliver better and faster results, communication issues come to the forefront of project managers' concerns. This is particularly palpable in software development projects which are increasingly virtual and knowledge-consuming as they require continuous generation and upgrade of shared information and knowledge. In a previous work, we proposed an SNA-BI based system (Covirtsys) that supplements the Analytics modules of the collaborative platform in order to offer a complementary analysis of communication flows through a network perspective. This paper concerns the application of this system on a software development project virtual team and shows how it can bring new insights that could help overcome communication issues among team members.*

## **KEYWORDS**

*Virtual teams, team's communication evaluation, social network analysis, collaborative platforms.*

## **1. INTRODUCTION**

In an attempt to answer the pressing needs for quick, high quality and low cost solutions to the increasingly complex problems brought about by their new ecosystem, organizations are moving from the rigid, hierarchical and traditional functional and matrix team structures into more organic and flexible forms of organizational structure that are team-based, collaborative and less dependent on geography. These new structures, referred to as virtual teams, build on the gains acquired from team-based designs (through delegating decision-making and problem solving responsibilities to line-level employees) and the prevalence of networked information and communication technologies. Virtual teams are thus defined as "groups of geographically, organizationally and/or time dispersed workers brought together by information technologies to accomplish one or more organizational tasks" [1]. They can hence be characterized as teams whose members: (a) interact through interdependent tasks guided by common purposes, (b) use ICT substantially more than face-to-face communication and (c) are geographically dispersed from each other [2].

A particular field that has seen the prevalence of virtual teams is software development. As companies are acknowledging that multi-site software development decreases product development cycle and are increasingly handing off (outsourcing) their software development to firms specializing in this area [3], software development teams moved from co-located to more distributed forms known as project virtual teams. Project virtual teams are defined as virtual teams which conduct longer-term projects with a predefined requirement and expected result (Software product, information system etc.). Software development has been described as "a

collaborative problem-solving activity where success is dependent upon knowledge acquisition, information sharing and integration, and the minimization of communication breakdowns" [4]. While networked technologies ensure communication and information dissemination within software development project virtual teams, they have proven to present many limitations [5]. To cite a few:

- Ineffective communication due to the absence of direct contact between team members and the lack of non-verbal information which is important for building trust [6].
- Lack of a common vision that knits the whole team together as project virtual teams face breakdowns in mutual knowledge that could undermine the success of the project at hand [7].
- Absence of visibility on the load and progress of work [8].
- Difficulties to monitor and manage the performance of the team [9].
- Difficulty of instilling trust among team members due to differences in thought processes across cultures and functional positions [10].
- And complexity of the management due to the differences between time zones, cultures and languages of the team members [11].

Because communication is instrumental to the collaboration of virtual teams, monitoring its flow becomes crucial to their success. The study of communication and dissemination of information relies increasingly on examining the interactions between members of the team. The analysis of these interactions is possible through Social Network Analysis (SNA) which is a field that provides research methods for mapping and measuring relationships and flows between people, groups, organizations and other connected information/knowledge entities. The next section introduces SNA and its underlying principles. It also presents the conceptual foundations of a Covirtsys, a monitoring system we propose to assess the dynamics between virtual team members through the analysis of the structure of their communication network. The third section describes the application of Covirtsys to analyze communication in a software development project case study. The last section concludes the paper and provides hints on future work.

## **2. SNA ON THE TEAM LEVEL**

Social Network Analysis (SNA) is a descriptive, empirical discipline that studies networks as a mathematical representation of complex systems by expressing them in terms of relationships among actors. A Social network is "a set of nodes (e.g. persons, organizations) linked by a set of social relationships (e.g. friendship, transfer of funds, overlapping membership) of a specified type" [12]. A relationship is defined as "the mode or process in which members of a social system are connected or associated interdependently among or between each other; i.e. a partial unification of members which, when considered irrespective of such a relation, would be incapable of being conceived together" [13]. SNA provides graph algorithms that help map, characterize and quantify topological properties of social networks, identify patterns of relations and recognize the roles of sub-groups and nodes within the network. SNA's underlying principles as identified by Wellman [14] can be summed as follows:

- 1) Behavior is interpreted in terms of the group's structural constraints rather than by examining drives, attitudes, or demographic characteristics of the group's individuals.
- 2) The focus of the analysis is on the interconnectedness between units and not on their inherent characteristics
- 3) The analysis assumes an interdependence among the network's actors. For example, a change in the relationships patterns affects network members' behavior.
- 4) The structure is considered as a network of networks. The structural properties of the network are thus seen as more than the sum of dyadic exchanges.

- 5) The analysis regards organizations as overlapping networks with fuzzy boundaries rather than discrete independent units of analysis.

SNA offers two perspectives of social systems: a micro-view and a macro-view. The micro or Ego-centric view focuses on a select actor (ego) and examines its neighbors (nodes that are connected to it), their neighbors and so forth. It studies the features of personal networks. The macro or Socio-centric view, on the other hand, provides a bird's eye perspective of the network and helps examine the structural patterns of the interactions among actors with the aim to explain and potentially generalize an outcome.

Driven by the realization that the behavior of complex systems is shaped by the interactions among their constituent elements [15], SNA is increasingly used to uncover patterns of relations characterizing a group or social system as a whole. Based on the recognition that SNA provides enough theoretical and practical ground to help make sense of communication flow patterns and examine the ties between teams' stakeholders, we proposed in previous works [16][17] a monitoring system for assessing the structure of the communication and information dissemination network within project virtual teams. We refer to this system from this point on as Covirtsys.

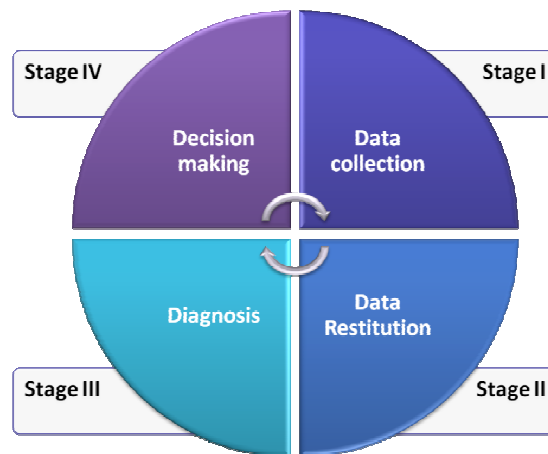


Figure 1. Covirtsys' conceptual model

The four-phases system is built on the collaborative platform in order to offer an SNA-based module that supplements the platform's Analytics module. The first stage consists of alimenting the system with the relevant data that originate from the collaborative platform. In the second stage, the system offers a restitution of the built network and the relevant metrics. The third phase aims to diagnose the network and identify the problems related to communication in the project virtual team and consequently, in the final stage, support taking the appropriate decisions that will enhance the team's structure.

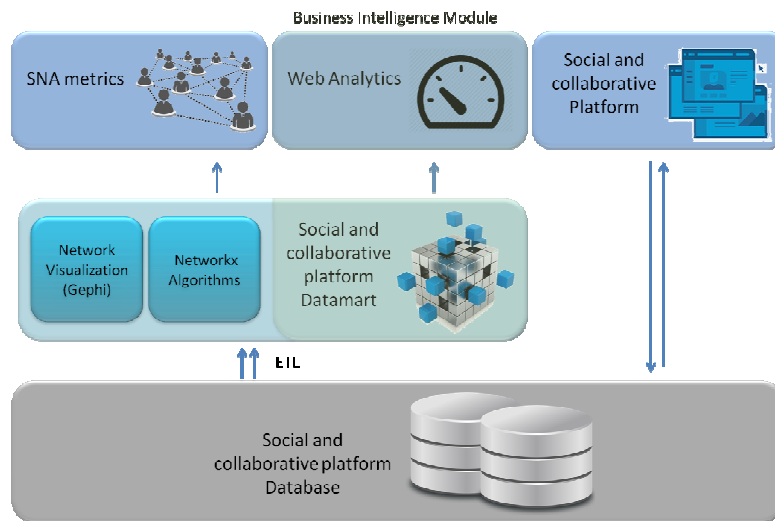


Figure 2. COVIRTSYS' applicative landscape

The upcoming sections of the paper describe the application of the above-mentioned system on a software development project virtual team in order to analyze the team's communication network and identify communication and information dissemination breakdowns.

### 3. ANALYSIS OF COMMUNICATION IN A SOFTWARE DEVELOPMENT VIRTUAL TEAM: A CASE STUDY

#### 3.1. Description of the case study

This case study examines the communication and information dissemination within a distributed software development team based on "Redmine", the used collaborative project management platform. Redmine is an open source web-based project management platform that supports many functions : member roles, Gantt charts, scheduling, calendar, roadmap, versions management, documents management, news delivery, files directory, activity view, and more [18]. It is considered a good choice for spread software development teams [19].

The software development project at the heart of this study, intends to develop an urban district's information system composed of three segments represented by projects on the Redmine platform. This virtual project team provides a good case study because of its virtuality and consequently its reliance on the collaborative platform. The online exchanges are thus considered a good representation of the team's overall interactions. Furthermore, the use of an open source collaboration platforms facilitates the tracking of relational data. The heterogeneous composition of the team (developers, analysts and few key users), makes the latter more susceptible to communication discrepancies, often caused by functional boundaries.

#	Tracker	Status	Priority	Subject	Assigned to	Updated
127	Bug	New	Normal	Ticket with attachments		12/22/2007 12:11 PM
116	Bug	New	Low	Keep playing audio when rw/ff and preserve pitch.	John Smith	12/17/2007 09:56 PM
88	Feature	Assigned	Low	HTTP Challenge-MDS authentication		12/22/2007 04:33 PM
83	Feature	Assigned	Low	Export the parameters of an input	John Smith	12/17/2007 09:56 PM
82	Feature	New	Low	Formatted text rendering support	Dave Loper	12/17/2007 06:58 PM
81	Feature	New	Normal	DVTS support		12/17/2007 06:58 PM
79	Feature	New	Low	QuickTime (re)turning		12/22/2007 04:33 PM
78	Feature	New	Low	Full H323 videoconferencing		12/17/2007 06:58 PM
77	Feature	Assigned	Low	Closed captions / Teletext support		12/17/2007 06:58 PM
74	Feature	New	Low	Progressive download playing		
73	Feature	New	Low	Dshow tuning support		
72	Feature	New	Low	V4L tuning support		
70	Feature	New	Low	Electric Program Guide		
69	Bug	New	Low	SDL vout cleaning		
65	Feature	New	Low	Protocol rollover support		
64	Feature	New	Normal	Improve ZLM functionality		12/22/2007 04:33 PM

Figure 3. Redmine's Issues List

Team members communicate on the Redmine platform around issues which are considered the principal artifact of conversation between the platform's users. Issues are used to track tasks and subtasks regarding bugs, feature requests and support requests and are thus considered a good indicator of the information dissemination processes relative to the collaboration within this particular team. The application of Covirtsys in this virtual team case study amounts to plugging the system on Redmine's database, collecting the pertinent data and computing the SNA-related metrics as a basis for the analysis. The case study is limited to a snapshot of the virtual team at a specific time and doesn't take into account the progress of the communication network that could be historized within the datawarehouse component of the system for team communication's monitoring purposes.

### 3.2. Data Collection

The data collection phase is shaped by the modeling of the virtual team as a social network. This modeling enables the application of the extensive research around network theory and SNA for examining the network's properties and topology. The communication and information dissemination flows that are identified on the collaboration platform are modeled using a network perspective and the structure of the team is examined by studying the underlying relationships that are woven among team members. The project virtual team is thus modeled as a set  $V$  of nodes or vertices, representing team's members, interconnected through a set  $E$  of links, called also edges or ties. A link  $(p,q) \in E$  is created whenever member  $q$  and  $p$  communicate around an issue. The created communication network  $G := (V,E)$  is thus an undirected and unweighted network (where there is no specification of the source and destination of the network's links and all these links are uniform i.e. without weight).

It is worth pointing out that parallel edges (where many ties exist among the same two nodes) are only taken into account for the purpose of visualizing the network. When examining the structure of the network, the parallel edges are reduced into single edges because on one hand, taking account of the existence of the connection is more important than the number of connections in studying characteristics such as clustering, brokerage etc. and on the other hand, the computed SNA metrics for such purposes support only simple graphs.

The analysis of Redmine's database showed that three tables can be used to build the network, namely Projects, Issues and Users. Each team member included in the project is presented by a node in the network. Based on the Issues table, a tie is drawn between two nodes when the author of an issue assigns it to a team member (self-loops aren't taken into account). The number of

occurrences of these pairings, reflecting the frequency of interaction between nodes, is also extracted and stored for later use in the visualization of the network.

The extraction from Redmine's database regarding the project at hand shows that the project is composed of three sub-projects. The team working on the project is comprised of 13 members. As issues consist the main artifact of interaction among team members, they have been identified and tracked while taking into account the author of the issue (who created it), the team member it was assigned to (a different team member or the author himself). There are 330 issues relating to the project, with 96 issues that are self-assigned (62 of them aren't assigned to anyone and thus are implicitly considered handled by their own author). The constructed network is thus comprised of 13 nodes and 267 ties which are later reduced to 28 when the parallel edges are compacted into single edges. For the purpose of this publication, nodes has been anonymized in order to conceal the identity of team members.

### 3.3. Data restitution

The data restitution phase renders the two views of the network through graphic visualizations (macro-view) along with a set of SNA-related metrics (micro-view). The network is built and visualized by Covirtsys through, respectively, a Python-based program and a powerful open source package named Gephi [20] which is often regarded in the network science communities as the “Photoshop for networks” [21].

The visualization of the network is carried out in two different ways. The first graph takes into account parallel edges in order to have an appreciation of the frequency of interactions among team members. However, for better visibility, the parallel edges are compacted into single edges and the frequency of connections is presented as an attribute (label) of the edges and is proportional to the width of the edge in the built graph. The wider the edge, the more interactions occur among the two nodes.

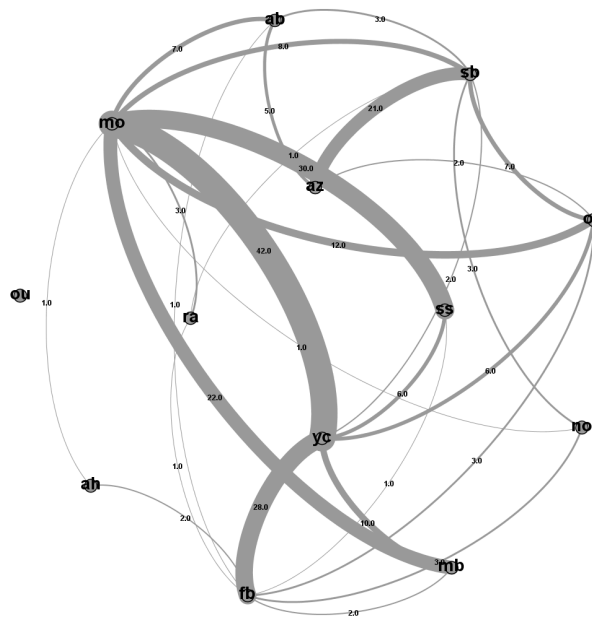


Figure 4 The communication network of the software development project virtual team (with parallel edges)

The second visualization doesn't take into account parallel edges nor self-loops. The built network allows the assessment of the connectivity among team members and dutifully represents the graph used as basis for the computation of the network metrics. For better assessment of the positions of certain nodes, the size of nodes is represented in proportion of their degree (number of neighbors). The bigger the node, the more neighbors it has and the more central it is in the network.

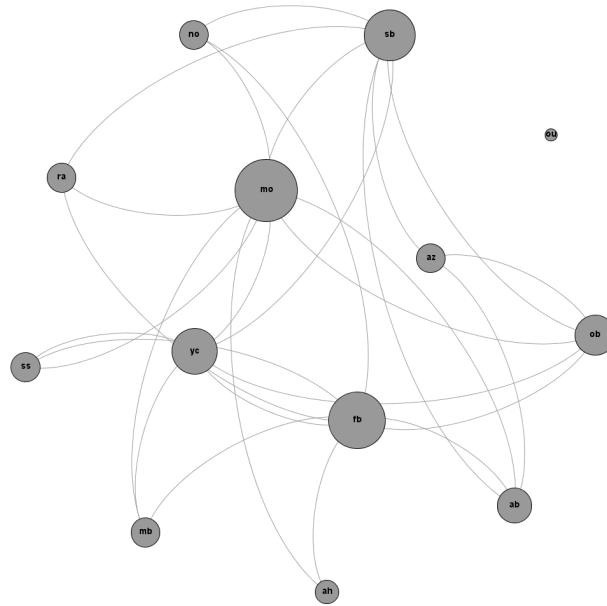


Figure 5. The communication network of the software development project virtual team (without parallel edges)

Visualizations convey a general idea on the structure of the network that can only be confirmed through the network's metrics. Covirtsys identifies the metrics pertaining to team network structure and classifies them in three dimensions: "Density", "centrality" and "cliques and bridges".

Each dimension examines the network through a different angle. The density dimension offers an insight on the level of connectivity among collaborators. The centrality dimension highlights the critical positions of certain actors within the team. The third dimension provides a view on the nature of the relationships among collaborators and their connections with cliques (subgraphs of the network where every two nodes are connected) within the network. The metrics, as specified in Table 1, are calculated using the underlying algorithms of Covirtsys through the SNA toolkit Networkx. "Networkx" is a Python language package for the exploration and analysis of networks. It is widely used within the SNA community as it has the most permissive license which allows integrating it within proprietary software [22].

Table 1. Results the computation of the network's measures

Dimension	Metrics	Results
Density	Density	0.359
	Average Degree	4.308
	Average neighbor degree	5.899
Centrality	Degree centrality	Per node
	Closeness centrality	Per node
	Degree Centralization	0.462
	Closeness Centralization	0.026
Cliques and bridges	Clustering coefficient	0.365
	Brokerage score (Betweenness)	Per node
	Number of maximal cliques	17
	Size of the largest clique	4
	Heterogeneity	0.641

The computation of the structural metrics delivers two types of results: Network-level and node-level metrics. Metrics on the network level are presented in Table 1. However, to get a better appreciation of the metrics that are calculated on the node level, we plotted them in histograms (cf. Figure 6).

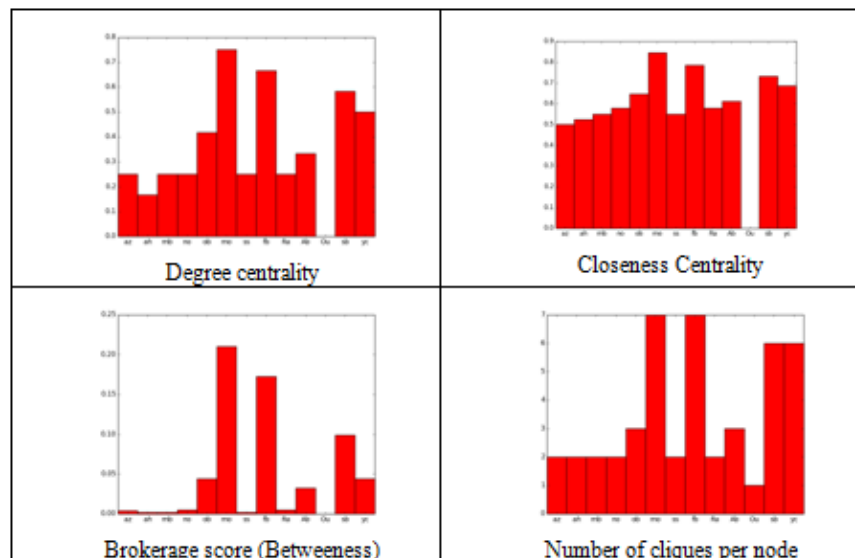


Figure 6. Histograms of SNA metrics on the node level

### 3.4. Diagnosis

The rendered visualizations and metrics represent the basis of the diagnosis phase which aims to identify communication-related issues within the virtual team. Communication issues arise in different forms within the virtual team and can be reflected in a network perspective as [21]:

- Fragmentation: where the network is divided into disconnected subgroups.
- Domination: where communication flows are monopolized by a few central nodes creating bottlenecks that could delay the speed of information spread within the virtual team.



- Insularity: where the nodes at the periphery of the network are disconnected or loosely connected to the rest of the network

The assessment of the studied team's communication network shows that these three issues emerge to different degrees.

Two graphs were constructed based on the studied team's communication network. The first graph shows that most interactions occur among few team members (six) including the most central nodes of the network. It also reveals the existence of an isolated team member ("ou") who hasn't interacted with any other collaborator throughout the duration of the project. The second graph brings to light four central nodes (hubs) who dominate the network("mo", "fb", "yc", "sb"). Apart from the isolated node, only three nodes have connections to non-hubs, while all the others have interacted with at least two of the central nodes.

The rendered metrics seem to confirm the observations made on the basis of both visualizations.

I). The density dimension: The density metric (35%) reflects a weakly dense network with an average degree that borders four and a node's neighborhood that is connected to an average of five nodes. This means that a team member interacts in average with only four (out of twelve) other members while his immediate contacts give him access to an average of five team members (who could overlap with his neighbors).

II). The centrality dimension: The distribution of degree centrality confirms the existence of few hubs that are central in the flow of information within the team. It also shows that most nodes are close to the average and hence, are not very peripheral in the network (except for the isolate). Closeness centrality reflects how fast information spreads from a given node to other reachable nodes in the network [24]. The high closeness centrality of a few nodes shows that they are adjacent to a great portion of the network's node (as it could be seen on the visualization). The existence of central nodes insures that the distance among actors in the network is shortened and thus information can reach nodes in a relatively short time. The centralization measures gauge the variation in the centrality scores among the nodes [25]. In the current case study, a 46% degree centralization shows that the power of actors varies substantially and the centrality is unequally distributed in the network. The weak closeness centralization on the other hand, shows that nodes' closeness centrality scores are at a close range from one another. This is mostly due to the existence of the four central nodes which shorten the paths among team members.

III). The cliques and bridges dimension: The clustering coefficient reflects the tendency of node's neighbors to connect to each other and thus to form cliques (subgraphs of the network where every two nodes are connected). Our network has a clustering coefficient of 0.365 which means that it's loosely clustered. This is confirmed by the high number of maximal cliques in the network (17) and the small size of the largest clique (4). The heterogeneity score is a correlation coefficient between nodes of different degrees. Networks with a positive heterogeneity coefficient (which is the case here) have distributed and consequently vulnerable high-degree hubs [26] as opposed to a negative heterogeneity that reflects a core of interconnected resilient hubs.

### **3.5. Decision Making**

The diagnosis of the network's visualizations and metrics serves to identify communication-related issues and "support" the decision making process that aims to enhance the communication network and by extension the collaboration within the virtual team. Depending on his objectives, and based on information provided by Covirtsys through the three previous phases, the project team manager can take certain measures to steer the existing network towards the desired team

structure. There are three principle issues that are observed in the studied network and that could require immediate attention:

I). The most noticeable issue of the studied communication network is the fragmentation of the network, apparent from the isolation of one node in the network. The integration of the corresponding team member can be performed through an on-boarding program (in case he is a new recruit) or a simple pairing with another more central team member (which will not only connect the isolate node to the network but easily grant him access to other team members). Techniques such as Pair Programming [27], where programmers develop software side by side, could be envisaged in order to fully integrate developers within the overall network.

II). The visualizations and metrics demonstrate the existence of few hubs that dominate the studied communication network as few nodes connect to non-hubs. Bypassing hubs when connecting two team members offers the advantage of lowering the workload of hubs and increasing the network's resilience (the decrease of the fragmentation's risk in cases where hubs decide to leave the team). A redesign of business processes with a concern of circumventing these central nodes is a viable action to make in order to overcome this domination problem.

III). The vulnerability of such network (due to its positive heterogeneity) also hints to a need to drive a knowledge management initiative that captures the knowledge of its few hubs and makes it available for reuse. The redistribution of knowledge will enable team members to address issues without having to go through hubs and thus enhance the overall communication network.

#### **4. CONCLUSIONS**

The fundamental goal of this paper is to present the application of Covirtsys, an SNA-BI based system we have proposed to assess and analyze virtual teams based on their members' interactions within the used collaborative platform, on a software development project virtual team case study. Covirtsys, which complements BI analytics modules, is based on SNA metrics in order to provide network visualizations for communication flows' diagnosis. The principle goal of the system is to continuously support decision makers in overcoming communication issues within virtual teams. In this paper, we have applied Covirtsys to diagnose virtual teams' communication flows at a specific time. The case study shows that Covirtsys is a viable decision support system which would direct the decision maker to take measures that will enhance the existing communication network. An interesting follow-up would be to validate the whole system and provide guidelines in order to support the managerial interpretations and decisions to-be-made based on the restituted results at different stages of the team's lifecycle.

#### **REFERENCES**

- [1] A. Powell, G. Piccoli, B. Ives, "Virtual teams: a review of current literature and directions for future research". *ACM Sigmis Database*, Vol. 35, no. 1, pp 6-36, 2004
- [2] E. Brauner, W. Scholl, "The information processing approach as perspective for group research". *Group Processes and Intergroup Relations*, Vol. 3, pp. 115-122, 2000
- [3] B. Sengupta, S. Chandra, V. Sinha, "A research agenda for distributed software development", in *The 28th international conference on Software engineering ACM*, Shanghai, China, May 20-28, 2006.
- [4] H. P. Andres, "A comparison of face-to-face and virtual software development teams". *Team Performance Management*, Vol. 8 n. 1/2, pp 39-48, 2002.
- [5] B.J. Bergiel, E.B. Bergiel, P.W. Balsmeier, "Nature of virtual teams: A summary of their advantages and disadvantages". *Management Research News*, Vol. 31 n. 2, pp. 99-110, 2008
- [6] J. D. Herbsleb, A. Mockus, "An empirical study of speed and communication in globally distributed software development". *Software Engineering, IEEE Transactions*, Vol. 29 n. 6, pp. 481-494, 2003.

- [7] C. Cramton, "The mutual knowledge problem and its consequences for dispersed collaboration", *Organization Science*, Vol. 12 n. 3, pp 346-371, 2001.
- [8] S. E. Poltrock, G. Engelbeck, "Requirements for a virtual collocation environment". *Information and Software Technology*, Vol. 41 n. 6, pp. 331-339, 1999.
- [9] K. S. Pawar, S. Sharifi, "Physical or virtual team collocation: Does it matter?". *International Journal of Production Economics*, Vol. 52 n.3, pp. 283-290, 1997.
- [10] B. Rosen, S. Furst, R. Blackburn, "Overcoming barriers to knowledge sharing in virtual teams". *Organizational Dynamics*, Vol. 36 n.3, pp. 259-273, 2007.
- [11] P. R Kuruppuarachchi, "Virtual team concepts in projects: A case study". *Project Management Journal*, Vol. 40 n. 2, pp. 19-33, 2009.
- [12] E. O. Laumann, J. Galaskiewicz, P.V. Marsden, "Community structure as interorganizational linkages". *Annual review of sociology*, Vol. 4, pp. 455-484, 1978.
- [13] R. T. Wigand, "Communication network analysis: History and overview", In *Handbook of organizational communication*, Ed. Norwood, NJ: Ablex, 1988, pp 319-359
- [14] B. Wellman, "Structural analysis: From method and metaphor to theory and substance". In *Social structures: A network approach*, Ed. Cambridge, England: Cambridge University Press, 1988, pp 19-61
- [15] E. Bullmore, O. Sporns, "Complex brain networks: graph theoretical analysis of structural and functional systems". *Nature Reviews Neuroscience*, Vol. 10 n. 3, pp 186-198, 2009.
- [16] L. Ben Hiba, M.A.J. Idrissi, "Towards an SNA system for evaluating project virtual teams based on social collaborative platforms". 4th Next Generation Networks and Services (NGNS) IEEE, Olhao, Portugal, December 2-4, 2012.
- [17] L. Ben Hiba, M.A.J. Idrissi, "Towards an SNA-BI based support system for the structural evaluation of project virtual teams". 13th Intelligent Systems: Theories and Applications ~SITA 2013~, IEEE, Rabat, Morocco, May 08-09, 2013.
- [18] A. Lesyuk, *Mastering Redmine*. Packt Publishing Ltd, 2013
- [19] C. Martin, "Comparison On Open Source Collaborative Project Management Software". 5th Knowledge Management: Projects, Systems and Technologies, Bucharest, Romania, November 12-13, 2010.
- [20] M. Bastian, S. Heymann, M. Jacomy, "Gephi: an open source software for exploring and manipulating networks." 3rd Int'l AAAI Conference on Weblogs and Social Media ~ICWSM~, San Jose CA, USA, May 17-20, 2009.
- [21] E. Meeks, "Gephi is like Photoshop for Networks", [Retrieved on Nov. 27,2013], retrieved from Stanford University site <https://dhs.stanford.edu/tools/gephi-is-like-photoshop-for-networks/>, 2011
- [22] D. Combe, C. LARGERON, E. EGYED-ZSIGMOND, M. GÉRY, "A comparative study of social network analysis tools", 2<sup>nd</sup> International workshop on Web Intelligence and virtual enterprises ~WIVE'10~, Saint-Étienne, France, October 11-13, 2010.
- [23] R. L. Cross, R.J. Thomas, *Driving results through social networks: How top organizations leverage networks for performance and growth*, Wiley, 2008
- [24] M.E.J. Newman, "A measure of betweenness centrality based on random walks". *Social networks*, Vol. 27 n. 1, pp 39-54, 2005.
- [25] L. C. Freeman, "Centrality in social networks: Conceptual clarification". *Social Networks*, Vol. 1 n. 3, pp 215-239, 1979.
- [26] M.E.J. Newman, "Assortative mixing in networks". *Physical review letters* Vol. 89 n. 20, pp 2087011-2087014, 2002
- [27] A. Cockburn, L. Williams, *The costs and benefits of pair programming. Extreme programming examined*, Addison-Wesley, 2000

## Authors

**Ben Hiba Lamia**, PhD candidate at ENSIAS, Mohammed V Souissi University.

**Prof Mohammed Abdou Janati-Idrissi**, Professor at ENSIAS, Mohammed V Souissi University.