SEOUL NATIONAL UNIVERSITY
College of Engineering
Technology Management, Economics, and Policy Program

Technology Management, Economics, and Policy Discussion Paper Series

SNA-Based Innovation Trend Analysis in Software Service Networks

Kibae Kim, Wool-rim Lee, Jörn Altmann

TEMEP Discussion Paper No. 2014:115

서 울 대 학 교
공과대학, 기술경영경제정책대학원과정
151-742 서울시 관악구 관악로 599

Technology Management, Economics, and Policy Program
College of Engineering, Seoul National University
599 Gwanak-Ro, Gwanak-Gu, Seoul 151-742, South-Korea
Phone: ++82-2-880-9140, Fax: ++82-2-880-8389

The *Technology Management, Economics, and Policy Program (TEMEP)*is a graduate program at Seoul National University. It includes both M.Sc. and Ph.D. programs, which are integrated into the graduate programs at the College of Engineering.

The *TEMEP Discussion Paper Series* is intended to serve as an outlet for publishing research about theoretical, methodological, and applied aspects of industrial economics, especially those related to the institute's areas of specialization, namely management of technology, information and communication technology, telecommunication, services, health industries, energy industries, as well as infrastructures for development. In particular, paper submissions are welcome, which analyze current technology and industry related issues, discuss their implications, and suggest possible alternative policies. The objective is to gain insights into important policy issues and to acquire a balanced viewpoint of policymaking, technology management, and economics. It will enable us to identify the problems in industries accurately and to come up with optimal and effective guidelines. In addition, another important aim of this series is to facilitate communication with external research institutes, individual researchers, and policy makers worldwide.

Research results disseminated by TEMEP may include views on policies for information and communication technologies, technology management, telecommunication, energy, health, and development economics, but the institute itself takes no institutional policy position. If appropriate, the policy views of the institute are disseminated in separate policy briefs. Thus, any opinion expressed in the TEMEP Discussion Paper Series is those of the author(s) and not necessarily the opinion of the institute.

Finally, the current editor of this series would like to thank Prof. Dr. Almas Heshmati for establishing this working paper series in January 2009 and for guiding its development during the course of the first 55 issues. In particular, Prof. Heshmati provided invaluable contributions for setting up the local IT infrastructure, registering the series with RePEc, disseminating the working papers, and setting the quality requirements.

Jörn Altmann, Editor

Office: 37-305
Technology Management, Economics, and Policy Program
College of Engineering
Seoul National University
599 Gwanak-Ro, Gwanak-Gu
Seoul 151-742
South-Korea

Phone: +82-70-7678-6676
Fax: +1-501-641-5384
E-mail: jorn.altmann@acm.org

SNA-Based Innovation Trend Analysis in Software Service Networks

Kibae Kim[1], Wool-rim Lee[1], Jörn Altmann[2]

[1]Department of Industrial Engineering
College of Engineering
Seoul National University
Seoul, South-Korea
kibaejjang@gmail.com, sara8616@snu.ac.kr

[2]Department of Industrial Engineering
College of Engineering
Seoul National University
Seoul, South-Korea
jorn.altmann@acm.org

August 2014

Abstract: Service networks can be considered to be open innovation systems. It has led to research on the structure of these networks, concentrating on the static network topology and its effect on innovation. However, the research misses the changes of network positions over time. In this paper, we examine the changes of nodes' positions in a software service network. The software service network has been built from empirical data. In this network, a node represents a Software-as-a-Service (SaaS) service and a link denotes a re-use of existing software services through a new service. Our results suggest that: first, software services undergo life cycles in their network positions; second, some software services achieve to hub position in their life cycle while others a core position; and third, an innovation trend appears at service category level not just by a single service. These results imply that innovation studies should not only consider static network positions and topologies but also trends of changing positions within the network.

Keywords: Open Innovation, Network Centralities, Software-as-a-Service, Composite Services, Service Network, Innovation Trend.

JEL Classification Numbers: D85, L86, O33.

1. Introduction

The advancement of IT technologies enabled the provisioning of software services on the Internet. The demand for new business models that motivate users to innovate is behind the forces that allow developers to enhance and combine services at no charge. In this sense, the network of composite software services represents an open innovation environment (Chesbrough, 2003). Examples of open innovation systems are the database of academic journals, which allow researchers to contribute their research articles, read research articles of others, and refer to them (Wagner and Leydesdorff, 2005), the community for open source developers, which stimulates the exchange of information on software development projects (Valverde and Solé, 2007), and the network of software services (SaaS) with open application programming interfaces (API), which allows accessing data and functions of services (Kim et al., 2011).

As open innovation on service networks promotes the reuse of existing innovation resources, the innovation studies get interested in the structure of the entire innovation system (Chesbrough, 2011; Maglio et al., 2006). One of the main efforts of these studies is to apply network analysis to the network of nodes (i.e., innovation agents and resources). It follows the research on social networks (Freeman, 1979), statistical physics (Albert et al., 1999), and co-authorship networks (Newman, 2001; Wagner and Leydesdorff, 2005). The innovation research tested the relationship between a node's social network position in a network structure and its innovation performance (Granovetter, 1973; Krackhardt and Stern, 1980). Similar analyses have also been performed for innovation communities (Grewal et al., 2006; Valverde and Solé, 2007) and innovation resources (Kim et al., 2011; Kim et al., 2010). However, prior research regards the network properties as static and their influence on innovation as invariant. Therefore, it misses the complex and dynamic behaviour of each node in evolving networks.

Complementing prior research (Kim and Altmann, 2013; Kim et al., 2013), this paper explores the dynamic behaviour of an innovation system. We analyze the trends of positions of software services in a software service network, which evolves with maintaining its scale-free topology and openness. The software service network is defined by nodes, which correspond to software services, and links, which represent the joint use of the linked nodes for the creation of new software services. For the empirical analysis of the software service network, we use data that has been collected from a public Web site for listing software services (www.programmableweb.com). Software services with open APIs can be used for creating new, composite software services. From 7427 software services, we selected the 4 most frequently used software services in order to examine their position over time by applying two basic and popular measures (i.e., node degree centrality and betweenness centrality). This allows examining whether the innovation leader shifts from one service provider to another. Furthermore, we use the same measure for the 4 most frequently used software services of the social networking service category. It helps answering whether the shift of an innovation leader's position is related to the innovation of a set of services (i.e., category).

Our analysis exhibits three results related to innovation trends of software services within a stable network structure (Kim and Altmann, 2013). First, each software service shows a life cycle (i.e., its ascent, its maturity, and its decline of its network position within the network) similar to the innovation adoption life cycle (Rogers, 2003). In detail, all software services that we monitored obtained a more central position of a software service network after entering the network. After being at this same level for a while, the service does not get adopted that much anymore. Actually, it gradually loses its central position. Second, the trend analysis shows that some software services achieve hub positions. A node in a hub position interconnects nodes of the entire network. Other software services achieve core positions within a cluster of software services (i.e., a set of nodes denser connected than other nodes (Scott, 1991)). The leading photo service (i.e., Flickr) gets dominated by losing its hub position to a social networking service (i.e., Twitter). The mapping service of Google (i.e., Google Maps) maintains its hub position. Third, the innovation adoption life cycle of a software service also depends on the innovation trend of its service category. The reason can be found in the fact that some of the software services of a software service category act as complements to the most frequently used services. These three findings are expected to help understanding the changing positions within a network with a stable network structure.

The remainder of this paper is organized as follows. The next section gives a brief summary about the conceptual background on the SaaS innovation system and on the network perspectives in innovation research. In section 3, we define our software service network, describe the data collected, explain how we choose representative software services, and define the network centrality measures. Section 4 exhibits the results of the analysis, i.e., the trend analysis of the position of the representative software services within the software service network. Section 5 concludes the paper with a discussion on the academic and entrepreneurial implications of our research.

2. Conceptual Background

2.1. Open Innovation in a Service Network

As IT technology advanced and new business models emerged that motivate consumers to participate in an innovation process (e.g., Web 2.0 service creation), software vendors started to offer their software as a service (SaaS). In detail, SaaS is a paradigm, which allows users to run software that is installed remotely via the Internet. The SaaS paradigm emerged with commercial computing in the 1980s but has been in the downturn with the rise of personal computers. Now, with the emergence of cloud computing in recent years, it moved into the limelight again. SaaS exists in a variety of areas ranging from office software (e.g., Google docs) to computing resources (e.g., Amazon S3) (Campbell-Kelly, 2009).

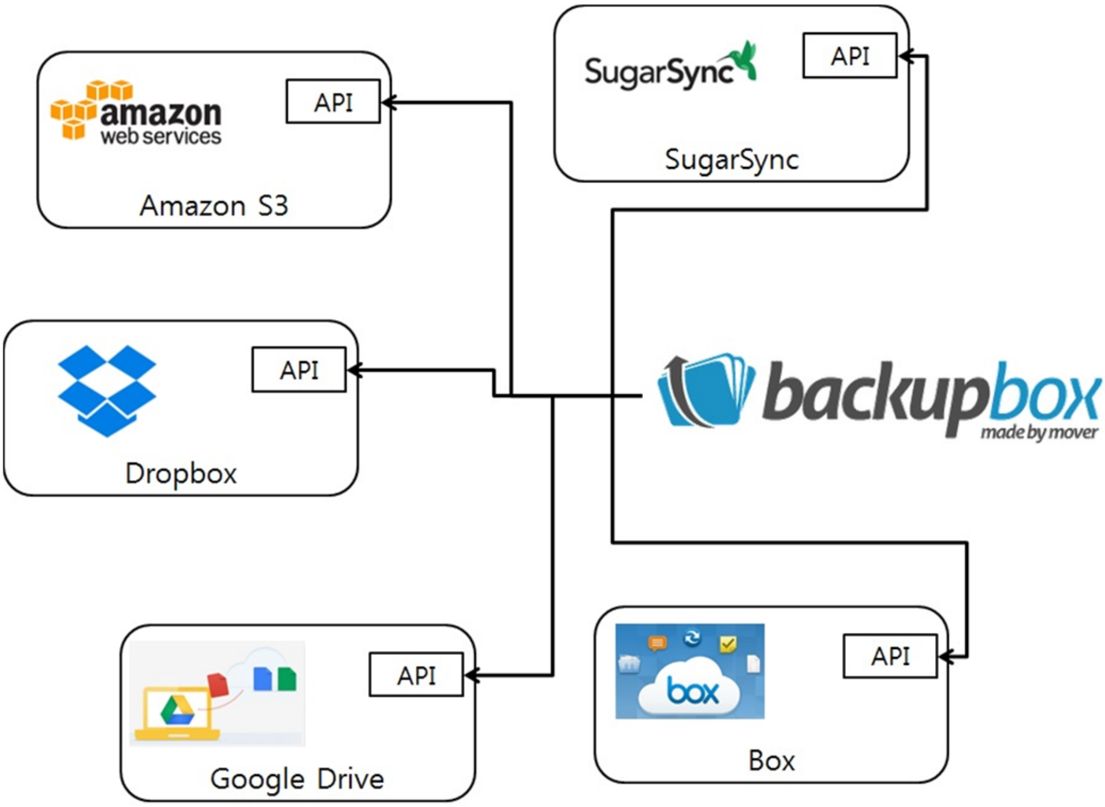

Figure 1. Example of of a composite service, Backupbox.

One type of SaaS implementation is based on Web services under the Service Oriented Architecture (SOA) concept. SOA defines how users can compose and reuse services through open interfaces, called open Application Programming Interface (API). For example, Google Maps offers APIs for other services to access it (Papazoglou and Georgakopoulos, 2003; Haines and Rothenberger, 2010). A composite service is created by adding a unique value to an existing SaaS service or by combining several existing SaaS services. This composite service is also called "mashup" (Ogrinz, 2009). Figure 1 shows an example of a composite service, which is called Backupbox. It supports users moving their files from one cloud storage to another cloud storage. Backupbox accesses the existing cloud storage services (e.g., Amazon S3, Dropbox, Google Drive, SugarSync, and Box) through their open APIs. According to the definition of our software service network, the Backupbox service links all five software services (Figure 1) with each other, resulting in a fully connected graph.

In this architecture, software vendors achieve innovation by utilizing their service users (O'Reilly, 2007). This describes an open innovation environment, in which innovation occurs through the free sharing of software services with their users and even with their competitors (Chesbrough, 2003). Any stakeholder of the SaaS environment can participate in the innovation by simply composing existing services with open APIs or by opening up the APIs of its software service. This allows users to reuse the data and functions of existing services for service development. This system is governed through an indirect network effect. That is, the more users exist to use these services, the more services are developed, benefiting the users building services. The more services exist, the higher the utility of the users. To benefit from these network effects over a longer time period, the development of services with open APIs is advantageous (Shy, 2001).

2.2. Network Position and Innovation

Social network analysis reveals the structural properties of innovation systems. Therefore, it provides qualitative properties. Classical diffusion theories, however, just provide aggregated information on the basis of the technology adoption rate (Bass, 1969; Peres et al., 2010). In social network research, a variety of measures have been developed to analyze the position of nodes in a network including centralities, clustering coefficients, and average path length (Freeman, 1979; Watts and Strogatz, 1998). The position of a node in a network is important, since the position represents the role of the node in the society (Scott, 1991). Two popular measures are the node degree centrality and the betweenness centrality (Freeman, 1979). The node degree centrality measures the number of adjacent nodes, which indicates how deeply a node is embedded among its neighbors. The betweenness centrality of a node considers the number of shortest paths (geodesics) through the node, compared to all possible shortest paths. It is used to determine the extent to which the node interconnects nodes. If a node has a high betweenness centrality but low node degree centrality, it can be derived that the node might bridge several separated communities within the network (Everard and Henry, 2002).

Empirical research about social networks found that the node degree distribution of some real-world social networks shows the scale-free property (Albert et al., 1999; Albert and Barabási, 2002; Valverde and Solé, 2007; Wagner and Leydesdorff, 2005). The scale-free property means that the frequency of node degree decays by a power function. That means, the node distribution is inhomogeneous, or highly skewed, compared to the exponential distribution of random networks or the distribution of regular networks. The empirical research that found many real world social networks to be scale-free has been conducted around the late 1990s. Until this time, research on social and technological networks had assumed that these networks are random. The empirical research of the late 1990s found that the scale-free structure of a network allows any individual to reach any other with a low number of hops (Albert et al., 1999). In a scale-free network, few nodes connect a majority of nodes while the node degree of the majority of nodes is low. The few nodes with high node degree are called “hubs”. The majority of nodes are connected in short distance with each other through these hubs.

The scale-free property is also important for the studies of innovation networks. The probability that a node obtains information correlates with the number of neighbors it has. Moreover, the information flow through a network depends on the network structure. For example, information disperses fast through hubs in scale-free networks (Kuandykov and Sokolov, 2010). In these networks, hubs have an advantage with respect to innovation, as they can gather information through their many neighbors. Nodes with a low node degree can efficiently get access to information it needs efficiently through few hubs.

Furthermore, a node with a low node degree and high betweenness could also be a key to innovation (Burt, 1992; Granovetter, 1973; Grewal et al., 2006). While innovation, in general, is a process of recombining fragmented existing knowledge (Hargadon, 2002), knowledge advances in the context of a group. A node (e.g., a firm or a researcher) of a group creates new knowledge by recombining knowledge of the group. If a node bridges its group with another, even though the connection to the other group is weak and infrequently used, the node enables innovation. The node can access a variety of knowledge of the other group and forward it to its own. This way, new ideas can emerge from the whole system and not only from a clustered group (Granovetter, 1973; Burt, 1992).

Prior research on the analysis of innovation networks misses out on dynamic aspects of network evolution. Innovation research investigating the effect of network position on innovation assumes this stability of network topology in evolving networks. Granovetter (1973) and Burt (1992) emphasized the importance of nodes connecting separated, distant clusters for innovation under a fixed network structure. Other research only focuses on the static properties and statistical relationship between nodes. For example, in the empirical network analysis and the evolutionary model introduced by Albert et al. (1999), the characteristics of scale-free networks are invariant and the network is generated by an invariant preferential attachment rule that determines the link between a new node and existing nodes. According to this model, a node locates at the central position forever, if it has been chosen as a hub at the early stage. Another empirical study found that an evolving network maintains its scale-free topology and openness structure (Kim and Altmann, 2013).

The network position affecting innovation has been investigated with respect to a variety of conditions, specifying the relationship between network position and innovation performance in detail. For example, the more central a node in a network is, the higher the innovation performance of the node is under the condition that the node has a good absorptive capacity (Tsai, 2001), that the node is in a central group (Sasidharan et al., 2011), and that the knowledge exchanged is simple (Hansen,1999). The analyses given in these publications were performed with data that shows a snapshot of network structure and innovation performance. These analyses of Granovetter (1973), Burt (1992), Tsai (2001), Sasidharan et al. (2011), and Hansen (1999) did not consider the time factor in their discussion of the relationship between network position and innovation.

Innovation systems are more dynamic than network studies assumed so far. On the one hand, innovation systems at a variety of levels (e.g., a single technology or entire industry) show a life cycle behavior from their emergence through maturity to their decline. This is explained by models that describe the driving force behind the rise and fall of an innovation system. The rise and fall happens due to the dispersion of technology (Bass, 1969; Rogers, 2003), the opportunities opened up through a new technology and the competition led by the opportunity (Jovanovic and MacDonald, 1994), and the opportunities of strategic alliances with competitors (Lemmens, 2004). On the other hand, especially in case of innovation through collective intelligence, the trend of innovation varies as the interest of the crowd shifts (Jin et al., 2011). The changing interest of the crowd is often revealed in service networks including Web sites, blogs, and forums. The analysis of interest change of the crowd is used to predict the innovation trend (Gloor et al., 2009). As a consequence of these two perspectives, the question rises whether the structure of an innovation network remains stable while the internal status of a node of the network changes.

3. Methodology

3.1. Service Network

For our analysis, data has been gathered from the web site www.programmableweb.com, which lists information about SaaS services. If a SaaS service is composed, this software service is called mashup in the terminology of the Web site. The web site provides information about the name of the software, the openness of the service's API, the services that are reused by the service, and about the launch date of the service. We collected this information for all software service since the first composite service was added on September 14th, 2005. The last entry in our data set is from September 30th, 2012.

There are several ways of defining a software service network (Dojchinowski et al., 2012; Huang et al., 2012; Hwang et al., 2009; Kim and Altmann, 2013; Kim et al., 2011; Weiss and Gangadharan, 2010). In this research, a software service network is defined as a set of nodes, which represent software services that opened up their APIs, and a set of links between these nodes (Hwang et al., 2009; Kim and Altmann, 2013; Kim et al., 2011). Each node represents a software service. We assume that a link appears between a pair of nodes if the two nodes are used together to develop a composite service. In other words, the creation of a composite service yields a complete graph of those software services that have been used for the development of the composite service. For example, because 5 software services with open APIs (i.e., Amazon S3, Dropbox, Google Drive, Box, SugarSync) are used to create Backupbox, as shown in Figure 1, all of the 5 nodes representing these software services are linked with each other in the corresponding software service network. That is, creating Backupbox generates 10 links among all the five nodes (i.e., a link between Amazon S3 and Dropbox, Amazon S3 and Google Drive, Amazon S3 and Box, Amazon S3 and SugarSync, Dropbox and Google Drive, Dropbox and Box, Dropbox and SugarSync, Google Drive and Box, Google Drive and SugarSync, and between Box and SugarSync). The links of the software service network are non-directional and weighted. That is, a link does not show the information about the source and the destination of a relationship but the usage frequency of the link. For example, assuming another composite service is created using Dropbox and Amazon S3 in addition to the software service Backupbox, the weight of the link between Dropbox and Amazon S3 were 2, while the other links still had a link weight of 1.

3.2. Measures

While the software service network is a weighted graph, the centrality measures have usually been defined for binary graphs (Everard and Henry, 2002; Freeman, 1979). A binary graph shows only the information of the existence of a connection between two nodes. It cannot express the heterogeneity among links. Therefore, we adapt the centralities, as defined for binary graph, to weighted graphs (Opsahl et al., 2010; Kim and Altmann, 2011). For this, we introduce the following terminology. Let w_{ij} be the weight of the link between nodes i and j belonging to network G whose size is g. The weight w_{ij} of a link between nodes i and j in a software service network means the number of occurrences of joint use of the two software services i and j in the software service network. It is an integer. If w_{ij} is zero, there is no link between the two nodes.

The degree centrality $k(i)$ of node i for a binary graph is the number of links of the node, i.e., $\sum_{j\neq i} a(i,j)$, where $a(i,j) = 1$, if nodes i and j are connected. If the nodes are not connected, $a(i,j) = 0$. Likewise, the degree centrality of node i in a weighted graph is the sum of weights w_{ij} of the links that node i has with the other nodes j. The degree centrality of a node in a weighted graph is also called the strength of the node (Opsahl et al., 2010). The degree centrality is likely to increase as the network size increases even in networks with identical density. In order to remove the effect of network size g on the degree centrality, the degree centrality in a binary graph is normalized by the maximum possible number $(g\text{-}1)$ of links that a node can have, i.e. $k(i)$ / $(g\text{-}1)$. The normalized degree centrality in a binary graph varies between 0 and 1, and goes to 1, if each node is connected to all the other nodes, and to 0, if no node is connected.

As the software service network is a weighted graph, the normalized degree centrality of node i, $CD'(i)$, in a weighted graph needs to be defined as:

$$CD'(i) = \sum_{j\neq i} w_{ij} \,/\, (g\text{-}1) \tag{1}$$

The normalized degree centrality for nodes of a weighted graph comes with the disadvantage that it can become larger than 1 if more than one composite services are developed from a pair of nodes in our software service network. It can vary to the number of composite services referring to the pair of nodes. The shortcoming is that its maximum is unbounded. Nevertheless, for our investigation, the definition of normalized degree centrality for nodes of weighted graphs is still good for comparing the centrality of the nodes in different networks of different sizes, as normalization still diminishes the effect of network size on the degree centralities.

Since the betweenness centrality is defined on the basis of the shortest paths (geodesics), the shortest path length in a weighted graph needs to be re-defined as well. The shortest path between two nodes is the path that passes the smallest number of links between them. The shortest path length $d(i,j)$ between nodes i and j is the number of links of the shortest path between the two nodes. As a pair of nodes is said to be more complementary as the weight of their link gets larger in the software service network, it is reasonable to assume that the distance between any adjacent nodes i and j is inversely proportional to the weight, i.e. $1/w_{ij}$. Therefore, the shortest path length $d^w(i,j)$ between two nodes i and j in a weighted graph G is defined as the smallest one among the path lengths p_{ij} between the two nodes, where p_{ij} is the sum of inversed weights of the links on the path between nodes i and j (Opsahl et al., 2010):

$$d^w(i,j) = \min \{|p_{ij}| \textit{ for all } p_{ij} \textit{ where } |p_{ij}| = 1/w_{ih} + \cdots + 1/w_{kj}\} \tag{2}$$

As a pair of nodes could be connected with more than one shortest paths according to the topology of graph (Freeman, 1979), the number of shortest paths σ_{ij}^w between nodes i and j in a weighted graph G, whose size is g, needs to be considered. $\sigma_{ij}^w(v)$ is defined as the number of shortest paths between nodes i and j passing through node v. Based on this definition, the betweenness centrality of node v in a weighted graph G is defined as $\Sigma_{i,j}\ \sigma_{ij}^w(v)$ / σ_{ij}^w for all nodes i and j of g, equivalent to the betweenness centrality in a binary graph (Freeman, 1979). In order to be able to compare betweenness centralities of networks despite their different network sizes, the betweenness centrality needs to be normalized by the maximum possible number of pairs of any two nodes (except for the node v) in a weighted graph G with size g, which is $(g\text{-}1)\cdot(g\text{-}2)/2$. Therefore, the normalized betweenness centrality of node i, $CB'(i)$, for a weighted graph is defined as:

$$CB'(i) = \sum_{i,j,j\neq i} \sigma_{ij}^{w}(v) \,/\, \sigma_{ij}^{w} \,/\, ((g\text{-}1)\cdot(g\text{-}2) \,/\, 2) \tag{3}$$

The normalized betweenness centralities for a binary graph and for a weighted graph vary between 0 and 1. The normalized betweenness centrality of a node goes to 0, if no shortest path passes through the node, and to 1, if all the shortest paths pass through the node.

Using the normalized degree centrality and the normalized betweenness centrality, we can classify the network position into four categories (Baek, 2013). First, a node with both high normalized degree centrality and high normalized betweenness centrality is a hub, according to the definition of (Barabási, 2009). Second, a node with high normalized betweenness centrality but low normalized degree centrality is a bridge, following the definition of Everard and Henry (2002). Third, we call the node with high normalized degree centrality but low normalized betweenness centrality a core. If a network contains several clusters (i.e., a set of nodes denser connected than other nodes (Scott, 1991)), cores locate at the centre of a cluster but do cannot reach nodes outside the cluster directly like hubs. Finally, if the normalized degree and betweenness centralities of a node are both low, we call the node a periphery.

4. Analysis Results

The data set, which was surveyed from www.programmableweb.com, includes 7427 software services, which have been registered from September 14th, 2005 to September 30th, 2012. Among these software services, 6780 services are composite services. They were created by utilizing 1153 software services that offered their functionalities through open APIs. With the surveyed information of software services, we defined a service network consisting of 1153 nodes and 23573 links. The service network of our study is undirected and weighted, and the weight of a link between a pair of nodes represents the number of composed software services on the basis of these two nodes.

For our analysis, we selected 8 services. Among them, 4 services (i.e., Google Maps, Twitter, YouTube, and Flickr) are the most frequently used services (Table 1). These are provided by the companies leading the Web service industry. Two of them are owned by Google, and the remaining two by Twitter and Yahoo. These software services were launched and registered early (i.e., between September 2005 and December 2006). The remaining 4 services (i.e., Facebook, Foursquare, LinkedIn, and Facebook Graph) together with Twitter are the most frequently used services in the social networking service category (Table 1). Two of them belong to Facebook, and the remaining two to Foursquare and LinkedIn. Not all of them are new services; LinkedIn and Facebook launched in 2003 and 2004. But, they only registered in the software service network a few years later. Except for Facebook, which was registered in August 2006, the other services have only been registered in February 2008, December 2009, and January 2010, respectively. The social networking service category was chosen since it is a category that became popular later than the other categories (i.e., mapping service category, video service category, photo service category)). The software services in this category might show a different trend in their position compared to the 4 most frequently used software services.

The remainder of this section shows the results of the network position trend analysis of the selected 8 software services. The network position is determined with respect to the normalized degree centrality and the normalized betweenness centrality for each month.

Table 1. Description of the selected software services.

Service Name	Provider	Service Category	Registration Date	Publication Date [a)]	Number of Uses
Google Maps	Google	Mapping	September 2005	February 2005	2263
Twitter	Twitter	Social Networking	December 2006	July 2006	644
YouTube	Google	Video	April 2006	February 2005	598
Flickr	Yahoo	Photo	September 2005	February 2004	590
Facebook	Facebook	Social Networking	August 2006	February 2004	352
Foursquare	Foursquare	Social Networking	January 2010	March 2009	82
LinkedIn	LinkedIn	Social Networking	February 2008	May 2003	46
Facebook Graph	Facebook	Social Networking	December 2009	May 2007	37

a) The publication dates of the listed software services were surveyed from http://www.wikipedia.com.

4.1. Network Position Trends of Most Frequently Used Services

The normalized degree centrality was measured for each of the four most frequently used software services (Google Maps, Twitter, Flickr, and YouTube), in order to investigate how deeply each of the selected software services is embedded among its neighbouring nodes in the software service network and how the embeddedness changes over time. Figure 2 illustrates the trend of the normalized degree centralities during the study period and shows that these services stand out compared to the average normalized degree centrality of all other services in the software service network.

The normalized degree centralities, CD', of Google Maps, Flickr, and YouTube show a similar trend. At the early periods, they increase fast and decline after staying at a certain level for some time. In particular, the normalized degree centrality of Google Maps roughly increases from 0.65 in September 2005 to 2.37 in December 2007. Then, the increase rate is lighter than before. After that, from December 2010 onwards, the normalized degree centrality slightly decreases from 2.70 to 2.28 at the end of the study period. Likewise, the normalized degree centralities of Flickr and YouTube soar from 0.10 in September 2005 to 2.01 in December 2007 and from 0.14 in January 2006 to 1.85 in September 2008, respectively. After a short period of light increase, the normalized degree centralities of Flickr and YouTube decline from 2.18 in May 2010 to 1.68 at the end of study period and from 2.02 in January 2011 to 1.75 at the end of study period, respectively. Twitter was introduced later than the other three software services. The normalized degree centrality of Twitter remains stable at about 0.30 until

January 2008, and increases fast to 1.75 in May 2011. Then, it stays stable again with a slight decrease until the end of study period.

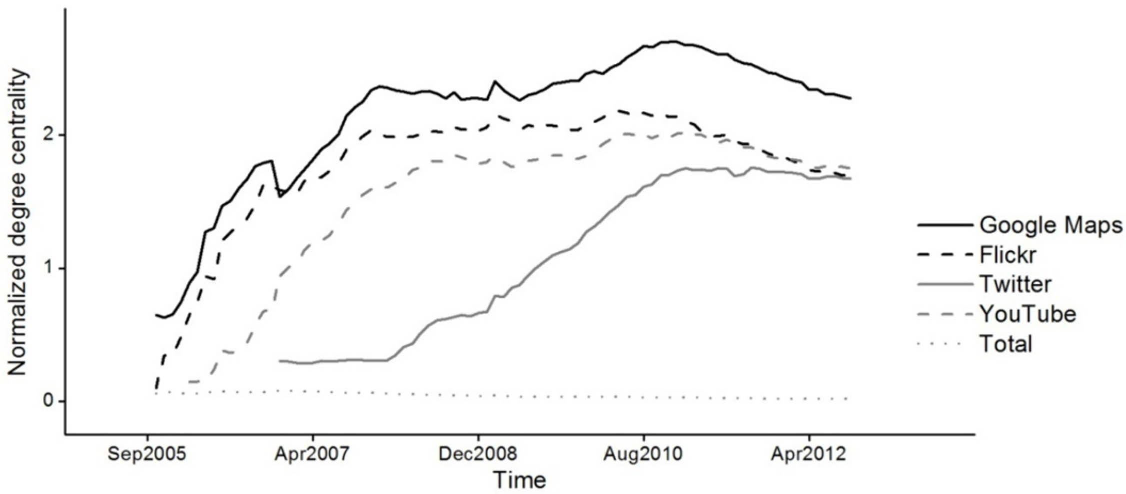

Figure 2. Trends of the normalized degree centrality for the four most frequently used software services.

The trends of normalized degree centralities of the selected software services look like an innovation adoption life cycle of technology. Prior research on diffusion of technology or on users' adoption of technology suggests that, after an inactive early period, a new technology is adopted exponentially but then slows down, so that the cumulated adoption rate of the technology shows an S-like curve (Rogers, 2003). For the software service network, it means that a software service reaches the central position, if it is frequently reused for composite service development, but loses its position (i.e., it is pushed out to the periphery of the network), if it is not reused anymore that frequently. From this perspective, Google Maps, Flickr, and YouTube have grown to their maturity phase during the study period, and now face their saturation period. Twitter is in the maturity phase at the end of study period as it emerged late.

Furthermore, Google Maps, Flickr, and YouTube do not show the early inactive periods. This might be related to the fact that the first three services entered the network at the beginning of the evolution of service network. These services were already known to users. Users already knew how to utilize these services. Therefore, these services did not have to wait for being diffused by "imitators" (Bass, 1969). However, Twitter shows the early inactive period after it entered the service network. Users had to learn the value of Twitter. At that point, Google Maps, Flickr, and YouTube have already been the most popular services.

Next, we calculate the normalized betweenness centralities, *CB'*, to diagnose whether the selected software services are bridges for other software services in the software service network, and to analyze the trend of the betweenness centrality of the selected software services. The analysis results, as described in Figure 3, show that the trend is idiosyncratic for each software service. The normalized betweenness centrality of Google Maps is considerably higher than those of the other three services for the entire study period. Although it shows a slight decrease over time, it remains at about 0.22 in average. The normalized betweenness centrality of Flickr declines steadily after the peak in the initial period. In detail, it rises sharply from 0.00 in November 2005 to 0.18 in June 2006. Then, it declines over time to 0.07 with some fluctuations. The normalized betweenness centrality of YouTube increases gradually during the first half of the study period and stayed at the same level during the second half. In detail, it increased from 0.00 in April 2006 to 0.09 in October 2009, and remained at this level after that. The normalized betweenness centrality of Twitter shows another interesting trend. It stayed at 0.00 between December 2006 and January 2008, and then jumped from 0.01 in February 2008 to 0.03 in July 2008. It jumped again from 0.03 in March 2009 to 0.08 in June 2009, and gradually increases to 0.12 by the end of study period.

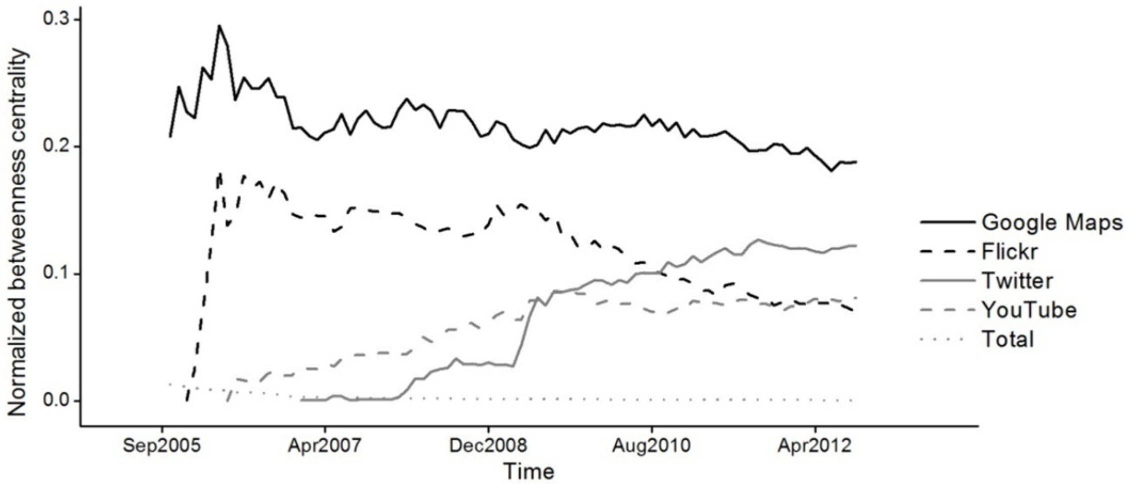

Figure 3. Trends of the normalized betweenness centrality of the four most frequently used software services.

Considering that a node with high betweenness centrality and low degree centrality implies that the node bridges multiple separated and distant clusters, while a node with high betweenness centrality and high degree centralities plays the role of a hub (Everard and Henry, 2002), we can make the following observations. In our software service network, Google Maps and Flickr are the hubs in the software service network initially. That is, they are linked to a lot of nodes in the network and connect many clusters. While Google Maps keeps its level, Flickr's normalized betweenness centrality declines over time. Twitter shows growth in both degree centrality and betweenness centrality. Moreover, Twitter becomes an even more important hub than Flickr from August 2010 onwards. Flickr loses its hub position and is at the same level as YouTube in September 2012. Therefore, we can state that the trend of normalized betweenness centralities of the selected software services depicts that the pivot of innovation shifts from photos to social networking around August 2012. Twitter, as a social networking service, combines software services, which were rarely combined to develop composite services before August 2012. Twitter took over role of Flicker.

It is also notable that the normalized betweenness centrality of YouTube grows slowly while its normalized degree centrality is high in the early periods already. This means that YouTube is linked with many nodes but does not connect many clusters. From a graph-theoretical perspective, such a pattern occurs, if the node is not a hub in the global network but a core node in a cluster.

The lead of Google Maps and Flickr during the study period is due to the market context at the time of the birth of composite software services. Google Maps and Flickr have been among the first software services, which promoted third party users to reuse the function of their software services through an open Application Programming Interface (API). The API of Google Maps was published in May 2005 (Rousch, 2005). Flickr provided an open API (though not fully comparable with today's version) from its beginning in 2004. Furthermore, the innovation trend at this time shifted from keyword search to geographic search and to picture search (Rouse et al., 2007). On the one hand, users demanded images of the site searched. On the other hand, letting users produce content has been an inexpensive solution for the service provider. Consequently, new services connected to Google Maps to tag site-specific information to maps. Flickr has been the source of user-generated content. The rise of Twitter since 2007 (Figure 2, Figure 3) is also related to the situation in the market. At that time, not only social networking services became popular but also mobile communication devices provided the platform for users' social activity and for innovation of new services (Basole, 2011). Twitter tabbed into this environment by providing an open API to its infrastructure.

Considering the results of all centrality measures, we can state that Google Maps is expected to maintain its structural position for a substantial period of time as it is a hub. Innovations will happen through the use of Google Maps together with other major services (e.g., Flickr, YouTube, and Twitter). However, it is also expected that not all services can keep their position. As the normalized betweenness centrality of Flickr indicates, Flickr will be unlikely to keep its role of a major player in the near future. Flickr lost its attraction in the software service network. The place that Flickr left is likely to be occupied by Twitter, whose normalized degree centrality and betweenness centrality grew steadily until the end of the study period. It attracts the creation of new composite services. In conclusion, we can state that the innovation leader changes from a photo service provider (i.e., Flickr) to a social networking service provider (i.e., Twitter). Nevertheless, the change of attraction from Flickr to Twitter in the software service network does not mean that Flickr declines in the market, nor that Flicker was replaced with Twitter. Our result does not show the rise and decline of services with respect to market demand. Our results show a change of innovation trend in the software service network.

4.2. Network Position Trend of Social Networking Services

The normalized degree centrality and the normalized betweenness centrality were also calculated for the five most frequently used software services in the social networking services category, i.e., Twitter, Facebook, Foursquare, LinkedIn, and Facebook Graph. These social networking services have been chosen, in order to analyze whether the centrality position trends of software services are due to the overall trend of a single software services category or whether some social networking services provide unique capabilities.

Figure 4 illustrates the trend of the normalized degree centrality, CD', for those five software services. In detail, the normalized degree centralities of Twitter, Facebook, Foursquare, LinkedIn, and Facebook Graph grow during the study period. In detail, the normalized degree centrality for Twitter increases fast from 0.03 in January 2008 to

1.75 in May 2011, after staying at about 0.03 for 13 months since its entrance in December 2006, and decreases slightly until the end of study period. The normalized degree centrality for Facebook shows almost the same pattern as that of Twitter (i.e., a gradual increase of the value until it reaches the maturity state at the end of the study period). It continuously increases from 0.01 in the entrance period (August 2006) to 1.24 in October 2011, and then remains stable with a slight decrease. The only difference is that Twitter starts to grow with a delay after entering the network while Facebook starts immediately. The other three software services in this category increase as well (see small box in the upper right corner of Figure 4). We re-draw the normalized degree centralities for Foursquare, LinkedIn, and Facebook Graph without Twitter and Facebook in order to see the trend clearly. The normalized degree centralities of Foursquare and Facebook Graph gradually increase to 0.33 and 0.12 until the end of the study period, respectively. The normalized degree centrality for LinkedIn increases overall with two jumps from zero to 0.13 in June 2008 and from 0.13 to 0.26 in January 2009.

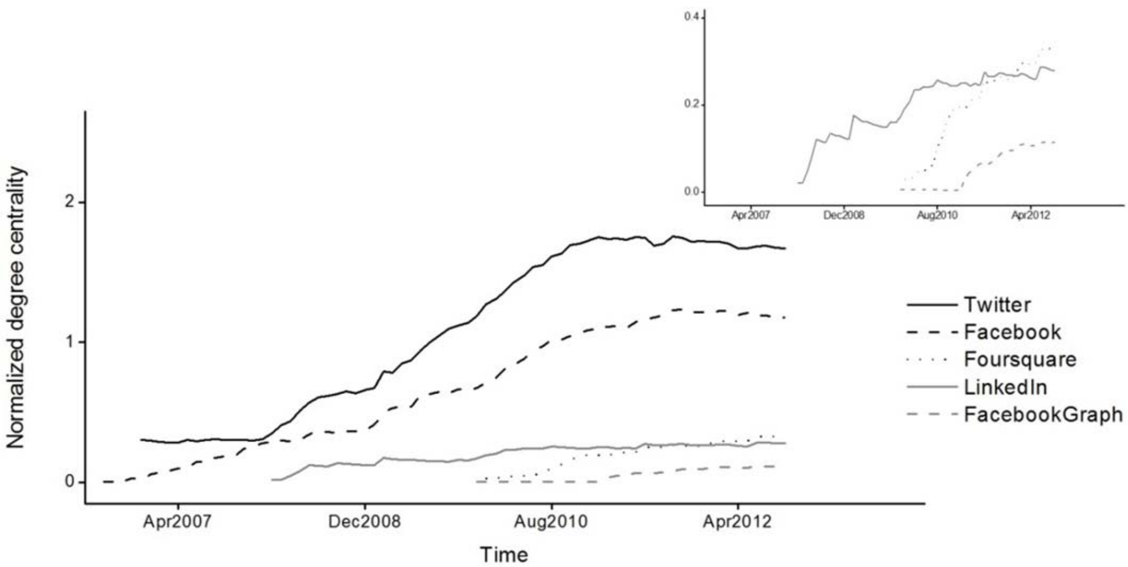

Figure 4. Trends of the normalized degree centrality for the five most frequently used social networking services.

The trend of the normalized degree centralities of the five software services in social networking category suggests that the innovation adoption life cycle of some services is synchronous. Twitter and Facebook emerged in the service network late at almost the same time, and are in their maturity phase at the end of study period, from the perspective of a three phase innovation adoption life cycle model: ascent, maturity, and decline (Rogers, 2003). Their life cycle is different from those of Google Maps, Flickr, and YouTube, which already reached saturation at the end of the study period after reaching their maturity in the middle of the study period. However, it has to be noted that the trends of the centralities might show differences later in their life cycles.

An interesting result is that Twitter shows a sharp increase in its degree centrality in the early periods. This means that Twitter occupied quickly a central position from the beginning. We conjecture that this is caused by the huge demand of service developers for the innovation provided by Twitter. Twitter filled the space for a certain type of service. Service developers utilized Twitter for developing composite services from the beginning of its entrance.

To determine whether the four software services (Facebook, Foursquare, LinkedIn, and Facebook Graph) locate at bridge positions similar to Twitter, we calculate the normalized betweenness centralities for the entire study period (Figure 5). The results show two distinctive trends of normalized betweenness centralities. On the one hand, the normalized betweenness centrality of Twitter and Facebook increase considerably. The normalized betweenness centrality for Twitter even shows a jump from 0.03 in March 2009 to 0.08 in June 2009. Afterwards, it grows to 0.12 until the end of study period. The normalized betweenness centrality for Facebook gradually increases from 0.00 at its entrance to 0.05 in October 2011, and then stays at about 0.04 with a slight decrease until the end of study period. On the other hand, the remaining three software services in the social networking service category have insignificant normalized betweenness centralities during the study period. The normalized betweenness centralities for Foursquare, LinkedIn, and Facebook Graph are all 0.00 except for some fluctuation of the value of Foursquare at the end of the study period, which does not surpass 0.005 (upper right graph in Figure 5).

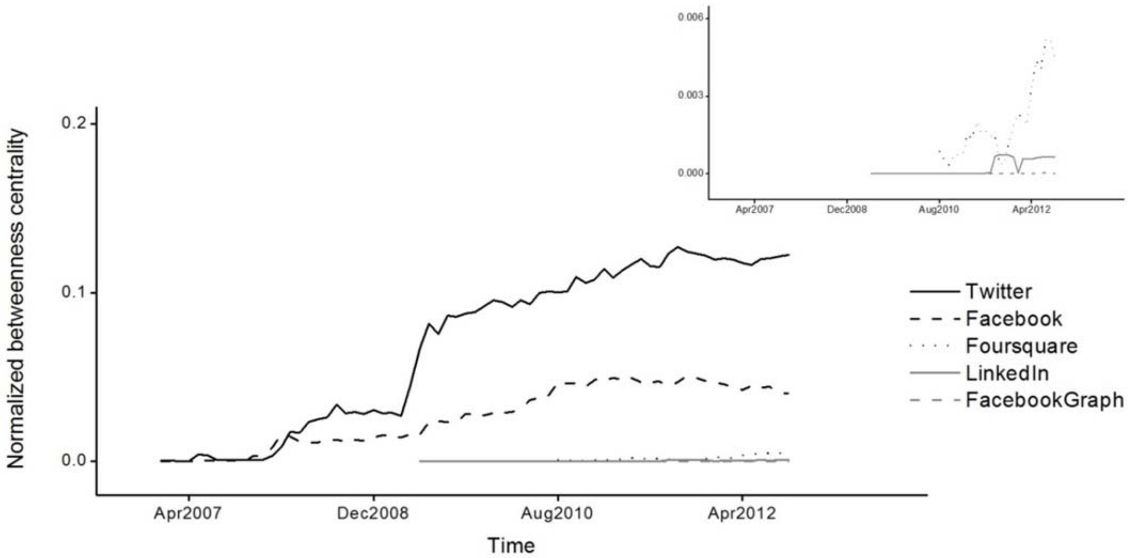

Figure 5. Trends of the normalized betweenness centrality for the five most frequently used social networking services.

Similar to the trends of the normalized degree centralities of the five software services in the social networking service category, the trends of the normalized betweenness centralities, *CB′*, also show similarities, resulting in synchronous innovation adoption life cycles. They can be grouped into two distinctive trends. Twitter and Facebook’s degree centralities and betweenness centralities grow to the peak at the end of the study period (Figures 4 and Figure 5), though being still in their ascent phases. For the remaining social networking services, the betweenness centralities are negligible. It is also noticeable that the betweenness centrality for Twitter soars considerably near April 2009 in Figure 5 while its degree centrality gradually increases in the same periods in Figure 4. This means that Twitter connected distant clusters of software services in the software service network, and new areas of composite services are created in recombining these clusters, which is common innovation process as discussed by Burt (1992) and Hargadon (2002).

Based on Figure 5, we can state that Twitter and Facebook are bridges for other software services, though not as strong as Google Maps. This is supported by the results for Twitter shown in Figure 3. Foursquare, LinkedIn, and Facebook Graph, however, enable innovation within clusters rather than innovation across clusters. They approach to the core of a cluster rather than bridging clusters.

Looking at all results of section 4, we suggest that the change in the innovation trend from photo services to social networking services as indicated in figures 2 and 3 is not only due to the excellence of Twitter but also depends on the innovation trend initiated by the entire social networking service category. The top five social networking services show a strong grows in the number of connections (Figure 4). Considering Figure 5, we can also conjecture that Foursquare, LinkedIn, and Facebook Graph play the role of complementing services to the two leading social networking services (i.e., Facebook and Twitter). They have a considerable amount of joint connections with Facebook and Twitter. Foursquare, LinkedIn, and Facebook Graph co-developed 43, 29, and 20 mashups with Facebook or Twitter among all 92, 47, and 35 mashups that they were part of, respectively.

5. Discussion and Conclusion

Within this paper, we analysed the trend of positions of eight representative software services, including Google Maps, Flickr, YouTube, Twitter, and Facebook, in a software service network with respect to their normalized degree centralities and normalized betweenness centralities. The results of the analysis comprise three aspects: First, the eight software services show innovation adoption life cycle behaviours with respect to their position within the service network (Figure 2). Software services rise to a maximum, remain at this level for some time, and, then, show a slight decline. In particular, all software services in the social networking service category approach to a central position with respect to their degree centrality during the study period while the most frequently used photo service (i.e., Flickr) loses its central position.

Second, the trend analysis also shows that some software services achieve positions that interconnect nodes of the entire network (i.e., they become hubs), and that some software services achieve core positions within a cluster of software services (Figure 3 and Figure 5). Moreover, the analysis of our data set showed that the innovation trend changed from a central photo service (i.e., Flickr) to services that belong to the social networking services category. In particular, Twitter replaced Flickr as a central service within the software service network.

Third, the innovation change from a photo service to social networking services is not just due to a single software service rather due to the emergence of an entire new category of services within the software services network (Figure 4). The reason for this could be based on the fact that some of the social networking software services act as complements to the most frequently used social networking services (i.e., Twitter and Facebook). Foursquare, LinkedIn, and Facebook Graph co-developed 47%, 62%, and 57% of all mashups with Twitter or Facebook, respectively.

Our findings cast important implications for academia and industry. First, academic research on innovation (e.g., Grewal et al., 2006; Kim et al., 2011) should also consider the dynamic network characteristics, in particular the changing positions of nodes in those evolving networks. The static network characteristics that help describing innovation performance are not the only factors, if the position of a node in the network can change as we demonstrated with our research. For example, if a correlation between network position of a software service and its innovation performance is conducted, the research may be misguided if the correlation is performed only at one point in time. As we showed, the performance and position of software services change over time.

Second, from a managerial perspective, our analysis results show that software services also follow a life cycle, coherent with prior research on innovation adoption life cycle for technology and on innovation diffusion (Rogers, 2003; Bass, 1969). These changes over time have been missed in prior research on networks related to information systems. Prior research on network analysis described only the network structure and the evolution rule of a network (Hwang et al., 2009; Valverde and Solé, 2007). Therefore, we suggest applying our trend analysis on the basis of normalized degree centralities and normalized betweenness centralities not only to the innovation within service networks but also to innovation research in general.

However, our methodology has limitations due to the database and the analysis technique. As some SaaS services (e.g., Google Maps) were published earlier than they were registered on www.programmableweb.com, the database does not cover precisely the whole lifespan of each service. That is, the lack of precise data in the early lifespan of these services may distort our life cycle analysis. In addition to this, our analysis does not cover all the software services in a service category. Further studies need to analyse the change of an entire service category to support our initial conjecture on the change of an innovation trend. Besides, our analysis does not consider substitutive relations between services. Currently, links between services are assumed to represent their complementary relations only. The emergence of a new service affects the life cycle of substitutable services according to diffusion theory (Peres et al., 2010). Our current trend analysis, however, does not show any causality between two life cycles of substitutable services.

Overall, our analysis of the trend of positions within service networks opens up the possibility for further research. First, to support our finding that the network position is dynamic, further analyses of network positions in software service networks are needed. For example, the growth rate of degree centralities and betweenness centralities could be investigated. Second, studies could investigate whether the innovation performance depends more on the dynamic properties of a network than on its static properties. Third, in order to generalize our findings, our findings need to be validated with statistical tests for all nodes in the software service network. Finally, the integration of our method into a decision support tool would help service owners to understand the performance of their services

Acknowledgements

This research was partly supported within the framework of the Basic Research Program of the National Research Foundation of Korea (NRF), which is funded by the

Ministry of Education (grant number: 2013R1A1A2058665), and through the ITEA 2 Project 10014 EASICLOUDS, which has been funded by the Korea Institute for Advancement of Technology (KIAT).

References

[1] Albert, R., Jeong, H. & Barabási, A.-L. (1999). Diameter of the World-Wide Web. Nature 401 (6749), 130-131.

[2] Albert, R. & Barabási, A.-L. (2002). Statistical mechanics of complex networks. Review of Modern Physics 74 (1), 47-97.

[3] Baek, S., Kim, K. & Altmann, J. (2014). Role of Platform Providers in Service Networks: The Case of Salesforce.com AppExchange. Proceedings of the 16th IEEE Conference on Business Informatics (CBI 2014).

[4] Barabási, A.-L. (2009). Scale-free networks: A decade and beyond. Science 325 (5939), 412-413.

[5] Basole, R.C. & Karla, J. (2011). On the evolution of mobile platform ecosystem structure and strategy. Business & Information Systems Engineering 3 (5), 313-322.

[6] Bass, F.M. (1969). A new product growth for model consumer durables. Management Science 15 (5), 215-227.

[7] Burt, R.S. (1992). Structural Holes: The Social Structure on Competition. Harvard Univ. Press, MA.

[8] Campbell-Kelly, M. (2009). Historical reflections: The rise, fall, and resurrection of software as a service. Communications of the ACM 52 (5), 28-30.

[9] Chesbrough, H.W. (2003). Open innovation: The new imperative for creating and profiting from technology. Harvard Business School Press, MA.

[10] Chesbrough, H.W. (2011). The case for open services innovation: The commodity trap. California Management Review 53 (3), 5-20.

[11] Dojchinovski, M., Kuchar, J., Vitvar, T. & Zaremba, M. (2012) Personalized graph-based selection of Web APIs. Lecture Notes in Computer Science 7649, 34-48.

[12] Everard, A. & Henry, R. (2002). A social network analysis of interlocked directorates in electronic commerce firms. Electronic Commerce Research and Applications 1 (2), 225-234.

[13] Freeman, L.C. (1979). Centrality in social networks: Conceptual clarification. Social Networks 1, 215-239.

[14] Granovetter, M.S. (1973). The strength of weak ties. American Journal of Sociology 78 (6), 1360-1380.

[15] Grewal, R., Lilien, G.L. & Mallapragada, G. (2006). Location, location, location: How network embeddedness affects project success in open source systems. Management Science 52 (7), 1043-1056.

[16] Gloor, P.A., Krauss, J., Nann, S., Fischbach, K. & Schoder, D. (2009). Web science 2.0: Identifying trends through semantic social network analysis. In Proceedings of the 12th CSE, Vancouver, Canada.

[17] Haines, M.N. & Rothenberger, M.A. (2010). How a service-oriented architecture may change the software development process. Communications of the ACM 53 (8), 135-140.

[18] Hansen, M.T. (1999). The search-transfer problem: The role of weak ties in sharing knowledge across organization subunits. Administration Science Quarterly 44 (1), 82-111.

[19] Hargadon, A.B. (2002). Brokering knowledge: Linking, learning and innovation. Research in Organizational Behavior 24, 41-85.

[20] Huang, K., Fan, Y. & Tan, W. (2012). An empirical study of ProgrammableWeb: A network analysis on a service mashup system. In Proceedings of the 19th ICWS, Honolulu, HI.

[21] Hwang, J., Altmann, J. & Kim, K. (2009). The structural evolution of the Web2.0 service network. Online Information Review 33 (6), 1040-1067.

[22] Jin, J.H., Park, S.C. & Pyon, C.U. (2011). Finding research trend of convergence technology based on Korean R&D network. Expert Systems with Applications 38 (12), 15159-15171.

[23] Jovanovic, B. & MacDonald, G.M. (1994). The life cycle of a competitive industry. Journal of Political Economy 102 (2), 322-347.

[24] Kim, K. & Altmann, J. (2013) Evolution of the software-as-a-Service innovation system through collective intelligence. International Journal of Cooperative Information Systems 22, 1340006.

[25] Kim, K., Hwang, J. & Altmann, J. (2011). Does Web2.0 foster innovation? An analysis of the openness of the Web2.0 service network. In Proceedings of the 44th HICSS, Kauai, HI.

[26] Kim, K. Altmann, J. & Hwang, J. (2010). The Impact of the Subgroup Structure on the Evolution of Networks: An Economic Model of Network Evolution. Proceedings of the IEEE International Workshop on Network Science for Communication Networks (NetSciCom2010), in conjunction with IEEE Infocom 2010, San Diego, USA.

[27] Kim K. & Altmann, J. (2011). A Complex Network Analysis of the Weighted Graph of the Web2.0 Service Network," Proceedings of Collective Intelligence 2011 (COLLIN 2011).

[28] Kim, K., Lee, W.-R. & Altmann, J. (2013). Patterns of Innovation in SaaS Networks: Trend Analysis of Node Centralities. Proceedings of the European Conference of Information Systems (ECIS2013).

[29] Krackhardt, D. & Stern, R.N. (1980). Informal networks and organizational crises: An experimental simulation. Social Psychology Quarterly 51 (2), 123-140.

[30] Kuandykov, L. & Sokolov, M. (2010). Impact of social neighbourhood on diffusion of innovation S-curve. Decision Support Systems 48 (4), 531-535.

[31] Maglio, P.P., Srinivasan, S., Kreulen, J.T. & Spohrer, J. (2006) Service systems, service scientists, SSME, and Innovation. Communications of the ACM 49 (7), 81-85.

[32] Lemmens, C.E.A.V. (2004). Innovation in Technology Alliance Networks. Edward Elgar.

[33] Newman, M.E.J. (2001). Clustering and preferential attachment in growing networks. Physical Review E 64 (2), 025102.

[34] Ogrinz, M. (2009). Mashup patterns: designs and examples for the modern enterprise. Addison-Wesley.

[35] O'Reilly, T. (2007). What is Web2.0: design patterns and business models for the next generation of software. Communications and Strategies 65, 17-37.

[36] Opsahl, T., Agneessens, F. & Skvoretz, J. (2010). Node centrality in weighted networks: Generalizing degree and shortest paths. Social Networks 32 (3), 245-251.

[37] Papazoglou, M.P. & Georgakopoulos, D. (2003). Introduction: Service-oriented computing. Communications of the ACM 46 (10), 24-28.

[38] Peres, R., Muller, E. & Mahajan, V. (2010). Innovation diffusion and new product growth models: A critical review and research directions. International Journal of Research in Marketing 27 (2), 91-106.

[39] Rogers, E. (2003). Diffusion of innovation. 5th ed. Free Press, NY.

[40] Rouse, L.J., Bergeron, S.J. & Harris, T.M. (2007). Participating in the geospatial web: Collaborative mapping, social networks and participatory GIS. In The Geospatial Web: How Geobrowsers, Social Software and the Web2.0 are Shaping the Network Society (Scharl, A. & Tochtermann, K. Ed.), Springer, 153-158.

[41] Rousch, W. (2005). Killer maps. MIT Technology Review, http://www.technologyreview.com/featuredstory/ 404705/killer-maps/ (Accessed on October 1, 2005).

[42] Sasidharan, S., Santhanam, R., Brass, D.J. & Sambamurthy, V. (2011). The effects of social network structure on enterprise systems success: A longitudinal multilevel analysis. Information Systems Research 23 (3-1), 658-678.

[43] Scott, J. (1991). Social network analysis: A Handbook. Sage Publication, UK.

[44] Shy, O. (2001). The Economics of Network Industries. Cambridge University Press.

[45] Tsai, W. (2001). Knowledge transfer in intra-organizational networks: Effects of network position and absorptive capacity on business unit innovation and performance. Academy of Management Journal 44 (5), 996-1004.

[46] Valverde, S. & Solé, R.V. (2007). Self-organization versus hierarchy in open source social networks. Physical Review E 76 (4), 046118.

[47] Wagner, C.S., & Leydesdorff, L. (2005). Network structure, self-organization, and the growth of international collaboration in science. Research Policy 34 (10), 1608-1618.

[48] Watts, D.J. & Strogatz, S.H. (1998). Collective dynamics of ‘Small-World’ networks. Nature 393 (6684), 440-442.

[49] Weiss, M. & Gangadharan, G.R. (2010). Modeling the mashup ecosystem: Structure and growth, R&D Management 40 (1), 40-49.

TEMEP Discussion Papers

2010-65: Radoslaw R. Okulski and Almas Heshmati, "Time Series Analysis of Global Airline Passengers Transportation Industry"

2010-66: Alireza Abbasi and Jörn Altmann, "On the Correlation between Research Performance and Social Network Analysis Measures Applied to Research Collaboration Networks"

2010-67: Kibae Kim, Jörn Altmann and Junseok Hwang, "An Analysis of the Openness of the Web2.0 Service Network Based on Two Sets of Indices for Measuring the Impact of Service Ownership"

2010-68: Bory Seng, "The Driving Forces Underlying the Growth of Total Factor Productivity in Cambodia, Quantitative and Qualitative Studies"

2010-69: Michael Maurer, Vincent C. Emeakaroha, Ivona Brandicand Jörn Altmann, "Cost and Benefit of the SLA Mapping Approach for Defining Standardized Goods in Cloud Computing Markets"

2010-70: Khin Swe Latt and Jörn Altmann, "A Cost-Benefit-Based Analytical Model for Finding the Optimal Offering of Software Services"

2010-71: Jung Eun Lee, Younghoon Kim, Yeonbae Kim and Donghyuk Choi, "The Impact of Technology Licensing Payment Mechanisms on Firms' Innovative Performance"

2010-72: Dongook Choi and Yeonbae Kim, "Effects of Piracy and Digital Rights Management on the Online Music Market in Korea"

2011-73: Tai-Yoo Kim, Jihyoun Park, Eungdo Kim and Junseok Hwang, "The Faster-Accelerating Digital Economy"

2011-74: Ivan Breskovic, Michael Maurer, Vincent C. Emeakaroha, Ivona Brandic and Jörn Altmann, "Towards Autonomic Market Management in Cloud Computing Infrastructures"

2011-75: Kiran Rupakhetee and Almas Heshmati, "Rhetorics vs. Realities in Implementation of e-Government Master Plan in Nepal"

2011-76: Alireza Abbasi, Jörn Altmann and Liaquat Hossain, "Identifying the Effects of Co-Authorship Networks on the Performance of Scholars: A Correlation and Regression Analysis of Performance Measures and Social Network Analysis Measures"

2011-77: Ivona Brandic, Michael Maurer, Vincent C. Emeakaroha and Jörn Altmann, "Cost-Benefit Analysis of the SLA Mapping Approach for Defining Standardized Cloud Computing Goods"

2011-78: Kibae Kim and Jörn Altmann, "A Complex Network Analysis of the Weighted Graph of the Web2.0 Service Network"

2011-79: Jörn Altmann, Matthias Hovestadt and Odej Kao, "Business Support Service Platform for Providers in Open Cloud Computing Markets"

2011-80: Ivan Breskovic, Michael Maurer, Vincent C. Emeakaroha, Ivona Brandic and Jörn Altmann, “Achieving Market Liquidity through Autonomic Cloud Market Management”

2011-81: Tai-Yoo Kim, Mi-Ae Jung, Eungdo Kim and Eunnyeong Heo, “The Faster-Accelerating Growth of the Knowledge-Based Society”

2011-82: Mohammad Mahdi Kashef and Jörn Altmann, “A Cost Model for Hybrid Clouds”

2011-83: Daeho Lee, Jungwoo Shin and Junseok Hwang, “Application-Based Quality Assessment of Internet Access Service”

2011-84: Daeho Lee and Junseok Hwang, “The Effect of Network Neutrality on the Incentive to Discriminate, Invest and Innovate: A Literature Review”

2011-85: Romain Lestage and David Flacher, “Access Regulation and Welfare”

2012-86: Juthasit Rohitratana and Jörn Altmann, “Impact of Pricing Schemes on a Market for Software-as-a-Service and Perpetual Software”

2012-87: Bory Seng, “The Introduction of ICT for Sustainable Development of the Tourism Industry in Cambodia”

2012-88: Jörn Altmann, Martina Meschke and Ashraf Bany Mohammed, “A Classification Scheme for Characterizing Service Networks”

2012-89: Nabaz T. Khayyat and Almas Heshmati, “Determinants of Mobile Telecommunication Adoption in the Kurdistan Region of Iraq”

2012-90: Nabaz T. Khayyat and Almas Heshmati,“Determinants of Mobile Phone Customer Satisfaction in the Kurdistan Region of Iraq”

2012-91: Nabaz T. Khayyat and Jeong-Dong Lee, “A New Index Measure of Technological Capabilities for Developing Countries”

2012-92: Juseuk Kim, Jörn Altmann and Lynn Ilon, “Using Smartphone Apps for Learning in a Major Korean University”

2012-93: Lynn Ilon and Jörn Altmann, “Using Collective Adaptive Networks to Solve Education Problems in Poor Countries”

2012-94: Ahmad Mohammad Hassan, Jörn Altmann and Victor Lopez, “ControlPlane Framework Emergence and its Deployment Cost Estimation”

2012-95: Selam Abrham Gebregiorgis and Jörn Altmann, “IT Service Platforms:Their Value Creation Model and the Impact of their Level of Openness on their Adoption”

2012-96: Ivan Breskovic, Jörn Altmann and Ivona Brandic, “Creating Standardized Products for Electronic Markets”

2012-97: Netsanet Haile and Jörn Altmann, “Value Creation in IT Service Platforms through Two-Sided Network Effects”

2012-98: Soyoung Kim, Junseok Hwang and Jörn Altmann, “Dynamic Scenarios of Trust Establishment in the Public Cloud Service Market”

2012-99: Lynn Ilon, “How Collective Intelligence Redefines Education”

2013-100: Ivan Breskovic, Ivona Brandic and Jörn Altmann, “Maximizing Liquidity in Cloud Markets through Standardization of Computational Resources”

2013-101: Juseuk Kim, Lynn Ilon and Jörn Altmann, “Adapting Smartphones as Learning Technology in a Korean University”

2013-102: Baseem Al-Athwari, Jörn Altmann and Almas Heshmati, “A Conceptual Model and Methodology for Evaluating E-Infrastructure Deployment and Its Application to OECD and MENA Countries”

2013-103: Mohammad Mahdi Kashef,Azamat Uzbekov, Jörn Altmann and Matthias Hovestadt, “Comparison of Two Yield Management Strategies for Cloud Service Providers”

2013-104: Kibae Kim, Wool-Rim Lee and Jörn Altmann, “Patterns of Innovation in SaaS Networks: Trend Analysis of Node Centralities”

2013-105: Netsanet Haile and Jörn Altmann, “Estimating the Value Obtained from Using a Software Service Platform”

2013-106: A. Mohammad Hassan and Jörn Altmann, “Business Impact Analysis of a Mediator between the Network Management Systems of the IP/MPLS Network and the Transport Network”

2013-107: A. Mohammad Hassan and Jörn Altmann, “Disjoint Paths Pair Computation Procedure for SDH/SONET Networks”

2013-108: Kibae Kim and Jörn Altmann, “Evolution of the Software-as-a-Service Innovation System through Collective Intelligence”

2014-109: Shahrouz Abolhosseini, Almas Heshmati and Jörn Altmann, “The Effect of Renewable Energy Development on Carbon Emission Reduction: An Empirical Analysis for the EU-15 Countries”

2014-110: Somayeh Koohborfardhaghighi and Jörn Altmann, “How Placing Limitations on the Size of Personal Networks Changes the Structural Properties of Complex Networks”

2014-111: Somayeh Koohborfardhaghighi and Jörn Altmann, “How Structural Changes in Complex Networks Impact Organizatioanl Learning Performance”

2014-112: Sodam Baek, Kibae Kim and Jörn Altmann, “Role of Platform Providers in Service Networks: The Case of Salesforce.com App Exchange”

2014-113: Somayeh Koohborfardhaghighi and Jörn Altmann, “A Network Formation Model for Social Object Networks”

2014-114: Somayeh Koohborfardhaghighi and Jörn Altmann, “How Variability in Individual Patterns of Behavior Changes the Structural Properties of Networks”

2014-115: Kibae Kim, Wool-Rim Lee and Jörn Altmann, “SNA-Based Innovation Trend Analysis in Software Service Networks”