An Empirical Study of ProgrammableWeb: A Network Analysis on a Service-Mashup System

Keman Huang  
Department of Automation  
Tsinghua University  
Beijing 100084, China  
hkm09@mails.tsinghua.edu.cn

Yushun Fan  
Department of Automation  
Tsinghua University  
Beijing 100084, China  
fanyus@tsinghua.edu.cn

Wei Tan  
IBM T. J. Watson Research Center  
Hawthorne, NY 10532, USA  
wtan@us.ibm.com

Abstract—A service ecosystem consists of services and their compositions (i.e., mashups) and evolves as a complex network system. It is driven by continuously emerged new services and the mashups of old services and new ones. Complex network analysis can be a powerful tool to study the static structure as well as the evolution of a service ecosystem. This paper presents a methodology to study such a system and an empirical study of ProgrammableWeb. To the best of our knowledge, ProgrammableWeb is the largest and most active Web APIs and mashups collection and consists of 4337 services and 6092 service compositions by Nov-2011. We conduct a comprehensive network analysis to quantitatively characterize the static structure and dynamic evolution of the ecosystem. The findings of this paper not only can help understand the current usage pattern and the evolution trace of the ecosystem, but also are applicable to other Web service systems.

Keywords-service ecosystem, complex network analysis, static structure, dynamic evolution, Web API, mashup

I. INTRODUCTION

With the wide adoption of Service-oriented Architecture (SOA), we see an increasing number of Web accessible services and their compositions. This enables service vendors and consumers to collaborate with each other cross organizations boundary. In this paradigm, atomic services are composed in an unforeseen way and are with composite services to form a service ecosystem [1]. Service ecosystem can be considered as a collection of services, their relations and associated entities such as service vendors. The rich set of the correlations [2, 3] result in a complex community structure in the ecosystem. Furthermore, the emerging and perishing of services, as well as their dynamic collaborations drive the service ecosystem to evolve over time [4]. Therefore, how to quantify the current usage pattern and the evolution trace becomes very important to understand the service ecosystem and sheds light on how to manage it. This paper tries to use a network analysis approach to conduct an empirical study of a service ecosystem, i.e. the service-composition collection from ProgrammableWeb [5]. We address the following issues:

- The static structure that reveals the current usage pattern of the ecosystem;
- The dynamic metrics that demonstrates the evolution trace of it.

To the best of our knowledge, ProgrammableWeb is by far the largest online repository of Web APIs (i.e., services) and their mashups (i.e., compositions). Web APIs are Web accessible endpoints for users to invoke. Mashup is a way to compose the Web APIs to create new applications [6]. In this paper, we do not distinguish between API and service, or between mashup and composition. With the growth of this ecosystem over years, the relations among APIs and mashups have been complex and evolving, much like those in a complex network. Thus the complex network analysis can be a powerful tool to study the static structure and the evolution of this service ecosystem. There are many studies on the evolution of different complex networks, such as the scientists’ correlation network and mobile phone calling network [7], the Internet [8], the email network [4] and the metabolic network [9]. Based on our previous work in analyzing the scientific service-workflow network [10], we take a step further to study both static and dynamic features of the ProgrammableWeb service-mashup system.

The contribution of this paper is both the methodology to quantify a service-mashup ecosystem and the empirical study on ProgrammableWeb. We introduce the motivation of the service ecosystem and then several networks are constructed in a formalism way. Furthermore several network-based approaches are applied to quantify the static structure and the dynamic metrics of the ecosystem. The discovered hidden knowledge accrued along with the evolution of the ecosystem can help understand the current usage pattern and the evolution trace of the ecosystem. Meanwhile, the findings of this empirical study offer suggestions on how to better maintain and utilize this system.

The rest of the paper is organized as follows. Section 2 shows the motivation of the study and the empirical data from ProgrammableWeb. The derived networks are proposed to model the ecosystem in Section 3. Section 4 presents the network analysis to quantitatively characterize the ecosystem, the observations and findings. Section 5 presents the related work and Section 6 concludes this paper.

II. MOTIVATION AND DATA ACQUISITION

There are two types of players, i.e., service providers and service consumers in ProgrammableWeb [11]. If there are some entities not only provide services but also develop the service compositions, we can artificially separate them into service providers and service consumers simultaneously. As shown in Figure 1, in the service ecosystem, service providers release services into the ecosystem, and services are classified into different categories (such as email, weather and map) based on their functionality. Service consumers choose one or more services and combine them into a composition (aka, mashup). The composition of services drives the evolution of the relations among services, providers and categories. To quantify the evolution of the...
ecosystem, we should track the growth of services and compositions, and their relations.

We obtain the ProgrammableWeb data regarding services and compositions from June 2005 to November 2011. In this data set, each service contains the information such as name, provider, category and the publication date; each composition contains the information such as name, creation date and the list of services in it. By removing compositions that contains no service, we get a collection of 6092 compositions, 4337 services, and 3416 providers, illustrated in Table I.

<table>
<thead>
<tr>
<th>TABLE I. OVERVIEW OF THE PROGRAMMABLEWEB DATASET</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of services (short as #services)</td>
</tr>
<tr>
<td>Number of providers (short as #providers)</td>
</tr>
<tr>
<td>Number of compositions (short as #compositions)</td>
</tr>
<tr>
<td>Average number of services per composition</td>
</tr>
<tr>
<td>Average number of compositions per service*</td>
</tr>
<tr>
<td>Average number of services per provider</td>
</tr>
<tr>
<td>Average number of services per category</td>
</tr>
<tr>
<td>Services used in at least one composition</td>
</tr>
</tbody>
</table>

* Only the services with at least one composition are counted.

III. DERIVED NETWORKS

In order to describe the relations between services, compositions, providers and the creation time of them, we firstly sort services and compositions chronologically based on their creation times; for each composition, we build a relation between it and the service invoked in it; for each service, we create a relation between it and its provider.

Then we can construct a composition-service network (c-s network), i.e., an undirected network in which a node represents either a service or a composition and edges represent the inclusive relations between them. We can also extract a service-provider network (s-p network) in which nodes represent services or providers and edges represent the ownership.

Similar to [10], we formalize the composition-service network as a $m \times n$ matrix, where $m$ is the number of compositions and $n$ is the number of services.

$$CS = [cs_{ij}], 0 \leq i \leq m, 0 \leq j \leq n$$

where $cs_{ij} = 1$ if composition $i$ invokes service $j$.

We derive two additional networks: a service-service network (s-s network) in which two services are connected if they appear in the same composition together; and a composition-composition network (c-c network) in which two compositions are connected if they invoke the same services. Thus we can formalize the two networks as follows:

$$SS = [ss_{ij}] = CS^T \cdot CS, 0 \leq i, j \leq n$$

where $ss_{ij}$ = the number of compositions where service $i$ and service $j$ are invoked, $ss_{ij}$ = the number of compositions where service $i$ is invoked.

Furthermore, we can get the service-provider network (s-p network) as a $n \times p$ matrix, where $n$ is the number of services and $p$ is the number of providers.

$$SP = [sp_{ij}], 0 \leq i \leq n, 0 \leq j \leq p$$

where $sp_{ij} = 1$ if service $i$ is offered by provider $j$.

Then we derive two additional networks: a composition-provider network (c-p network) in which the composition and provider are connected if the service invoked by the composition is offered by the provider; and a provider-provider network (p-p network) in which two providers are connected if the services offered by two providers participating in the same composition.

$$CP = [cp_{ij}], 0 \leq i \leq m, 0 \leq j \leq p$$

where $cp_{ij} = 1$ if provider $j$ participate in composition $i$.

$$PP = [pp_{ij}] = CP^T \cdot CP, 0 \leq i, j \leq p$$

where $pp_{ij}$ = the number of compositions which provider $i$ and provider $j$ both participate in, $ss_{ij}$ = the number of compositions which provider $i$ participates in.

IV. NETWORK ANALYSIS

Applying network analysis techniques, we study the static structure and the dynamic metrics of the networks introduced in Section III. Due to the space limitation, we only show analysis on the composition-service network and the service-service one in this section.

A. Composition-Service network (c-s network)

1) Static structure

a) Method and Observation
is the new services introduced in the month. indicates that services.

if k compositions each of which contains i,j compositions. Figure 3 shows the network using Pajek \[12\], a software widely used for network analysis. Yellow squares represent compositions, green diamonds represent services, and an edge which connects a composition with a service indicates that the service is invoked in the composition. Only the services involved in at least one composition are included in the figure and they only account for 22.2% of all the published services.

Sorting compositions by the number of services contained in them, we can get the quantity-frequency (QF) distribution \(f_i\) for them, where \(f_i(j) = j\) means that there are \(j\) compositions each of which contains \(i\) services. Similarly, sorting the services by the number of compositions in which the services participate, we can get the QF distribution \(f_s\) for services, where \(f_s(i) = j\) means there are \(j\) services each of which participates in \(i\) compositions. Figure 3 shows the distribution of \(f_s\) and \(f_i\). We can see that 95.45% compositions contain no more than 5 services and 92.89% services participate in no more than 5 compositions.

Furthermore, we calculate the cumulative distributions (CF) of services and composition as follows:

\[
F_s(i) = \sum_{k \leq i} f_s(k) = \text{the total number of services which invoke no less than } i \text{ services.}
\]

\[
F_c(i) = \sum_{k \leq i} f_c(k) = \text{the total number of compositions which participate in no less than } i \text{ compositions.}
\]

Figure 4 shows the cumulative distributions which are both plotted on the log-log axes. From this figure we get the linear relation characteristic which indicates that both of these two distributions meet the power-law distributions.

\[
\log(Y) = -2.3601 \log(X) + 4.1769
\]

\[
\log(Y) = -0.93239 \log(X) + 3.1341
\]

\[
R^2 = 0.98375
\]

\[
R^2 = 0.99403
\]

\[
C_{In i j}^C = \frac{\text{the number of compositions which participate in no less than } i \text{ compositions.}}{\text{the total number of compositions}}
\]

\[
C_{In i j}^S = \frac{\text{the number of services which participate in no less than } i \text{ compositions.}}{\text{the total number of services}}
\]

\[
\log(Y) = -0.87953 \log(X) + 4.1769
\]

\[
\log(Y) = -0.93239 \log(X) + 3.1341
\]

\[
R^2 = 0.98375
\]

\[
R^2 = 0.99403
\]

\[
\text{Figure 2. Overview of composition-service network. Yellow squares represent compositions, green diamonds represent services, and an edge indicates that the service is invoked in the composition.}
\]

\[
\text{Figure 3. Histogram of the quantity-frequency distribution. (a) the number of services (#Services) per composition. (b) the number of compositions (#Compositions) per service}
\]

\[
\text{Figure 4. The CF distribution of compositions and services. (a) number of services per composition. (b) number of compositions per service}
\]

\[
\text{b) Analysis}
\]

From the observation we can see that a minor percentage of services (22.2%) are ever used in compositions and most services are invoked by a few compositions (37.5% of the actually used services involved in only one composition). All these metrics indicate that the reuse of services in this ecosystem is low. Also the significant power-law distribution of compositions number per service shows that most services are used by few compositions while a few services are very popular. The power-law distribution indicates that there can be a preferential attachment [13] for the service selection, that is, consumers intend to select the popular services to form a new composition. This is because consumers usually believe that they can learn from the historical usage of the services and build a new composition with potentially better quality by reusing popular ones. Meanwhile, most compositions invoke small number of services which means that the complexity of the compositions in the ecosystem is still low.

2) Dynamic metrics

a) Arrival of services and compositions

- Method and Observation

In order to quantify the causality between the arrival of services and compositions, we sort the services and compositions chronologically and then count the new services and compositions in each month to form the increase series (IS) for services and compositions.

\[
S_{ln} \text{ is the IS for services and } S_{ln}(i) = j \text{ indicates that there are } j \text{ new services introduced in the } i^{th} \text{ month.}
\]

\[
C_{ln} \text{ is the IS for compositions and } C_{ln}(i) = j \text{ indicates that there are } j \text{ new compositions created in the } i^{th} \text{ month.}
\]

There are 78 months in the time span of our data set (from June 2005 to November 2011), and therefore the length of these two series are 78.

Granger causality is a statistical concept of causality which was introduced by Granger in 1960s and has been widely used in the economics [14]. In order to employ the Granger Causality test, the Augmented Dickey-Fuller test (ADF test) [15] is used for the unit root test to check whether these two series are stationary. As shown in Table II, we find

\[
\text{Figure 4. The CF distribution of compositions and services. (a) number of services per composition. (b) number of compositions per service}
\]

\[
\text{b) Analysis}
\]

From the observation we can see that a minor percentage of services (22.2%) are ever used in compositions and most services are invoked by a few compositions (37.5% of the actually used services involved in only one composition). All these metrics indicate that the reuse of services in this ecosystem is low. Also the significant power-law distribution of compositions number per service shows that most services are used by few compositions while a few services are very popular. The power-law distribution indicates that there can be a preferential attachment [13] for the service selection, that is, consumers intend to select the popular services to form a new composition. This is because consumers usually believe that they can learn from the historical usage of the services and build a new composition with potentially better quality by reusing popular ones. Meanwhile, most compositions invoke small number of services which means that the complexity of the compositions in the ecosystem is still low.

2) Dynamic metrics

a) Arrival of services and compositions

- Method and Observation

In order to quantify the causality between the arrival of services and compositions, we sort the services and compositions chronologically and then count the new services and compositions in each month to form the increase series (IS) for services and compositions.

\[
S_{ln} \text{ is the IS for services and } S_{ln}(i) = j \text{ indicates that there are } j \text{ new services introduced in the } i^{th} \text{ month.}
\]

\[
C_{ln} \text{ is the IS for compositions and } C_{ln}(i) = j \text{ indicates that there are } j \text{ new compositions created in the } i^{th} \text{ month.}
\]

There are 78 months in the time span of our data set (from June 2005 to November 2011), and therefore the length of these two series are 78.

Granger causality is a statistical concept of causality which was introduced by Granger in 1960s and has been widely used in the economics [14]. In order to employ the Granger Causality test, the Augmented Dickey-Fuller test (ADF test) [15] is used for the unit root test to check whether these two series are stationary. As shown in Table II, we find
that \( SIn \) is not stationary where the p-value equals to 0.3938 (>0.05) and \( Cln \) is not stationary where the p-value equals to 0.2139 (> 0.05). The first difference data of \( SIn \) which we refer as \( FDSIn \) is not stationary where the p-value equals to 0.1136 (>0.05), while the first difference data of \( Cln \) which we refer as \( FDCIn \) is stationary where the p-value equals to 0.0001 (<0.05). The second difference data of \( SIn \) which we refer as \( SDSIn \) is stationary where the p-value equals to 0.0001 (<0.05).

Thus we employ the Granger Causality test between \( FDCIn \) and \( SDSIn \). From the result we can see that \( SDSIn \) does Granger Cause \( FDCIn \) where p-value equals to 0.0964 (<0.1) and \( FDCIn \) does not Granger Cause \( SDSIn \) where p-value equals to 0.6652 (> 0.1).

**Analysis**

In fact, \( SDSIn \) is the acceleration of the increasing number of services while \( FDCIn \) can be considered as the growth rate of the increasing number of mashups. Thus from the Granger Causality test we can conclude that the rapidly increasing number of services in the ecosystem will promote the increasing interest in mashups while the impetus of the increasing number of mashups to the releasing of new services is not apparent.

b) **Concentration rate of services**

**Method and Observation**

Concentration rate (CR) of services indicates the heterogeneity of the number of compositions per service (i.e. the heterogeneity of Figure 4 (b) histogram). We use two indicators: the Herfindahl-Hirschman indicator (HHI) [16] and Pareto indicator (PI) that are defined as follows, respectively:

\[
HHI^+ = \frac{N \cdot \sum \left( \frac{f(i)}{N} \right)^2 - 1}{N-1} = \frac{HHI - 1}{N \cdot \left( \frac{1}{N} \right)^{-1}}, N = \sum f_i(i) \tag{7}
\]

\[
PI = \frac{\sum_{i=20\%} f_i(i)}{\sum f_i(i)} \tag{8}
\]

where \( f_i \) is the quantity-frequency distribution of the number of compositions per service. Note that if the QF distribution is uniform, \( HHI \) will be \( 1 / N \); if all services are invoked by the same number of compositions, \( HHI \) will be 1. Therefore the value of \( HHI^+ \) will vary between 0 and 1. \( PI \) indicates the share of the top 20% elements in QF.

 Apparently, the larger the \( HHI^+ \) is, the more homogeneously the services are invoked by compositions. The larger the \( PI \) is, the more services are invoked by few compositions.

We can get the quantity-frequency distribution of the number of compositions per services by year (i.e., the data in Figure 4 (b) divided by year). Based on these QF distributions, we calculate the two concentration indicators as well as the cumulative distribution (each cumulative distribution meets the power-law distribution, similar to Figure 4 (b)). Figure 5 (a) shows the PI indicator increases from 74.07% by 2005-12 to 88.99% by Nov-2011; the \( HHI^+ \) indicator decreases from 0.329 by Dec-2005 to 0.147 by Dec-2009 and then increases from 0.153 by Dec-2010 to 0.176 by Nov-2011. Furthermore, the absolute value of the exponent (EAV) of each CF distribution in magnitude increases from 0.811 by Dec-2005 to 0.932 by Nov-2011.

**Analysis**

From the observation, a few popular services capture higher probabilities to be selected into new compositions (significant power-law distribution). However, the increasing absolute exponent value shows that the strength of the preferential attachment appears to have weakened during our study period (from Sept-2005 to Nov-2011). This indicates that although the popular services gain higher probabilities to be selected for the new compositions, the consumers pay more and more attention to the services with small number of compositions. Meanwhile, more and more new services begin to appear in the compositions (191 new services in 2011), and although most services appear in a few compositions (PI indicator is increasing), several services become more and more popular (\( HHI^+ \) indicator decreases from 2005 to 2009).

c) **Average number of services per composition**

**Method and Observation**

Sorting the compositions by the creation time, and then calculating the average number of services of the existing compositions at each creation time, we can get the variation curve of the average number of services per composition. From Figure 6 (a), we can see that after a short period of time (by 500, Feb-26-2007) in concussion, the average number of services per composition increases steadily over time.

**Analysis**

Average number of services per composition is an indicator of the complexity of the ecosystem. The increasing average number of services per composition shows that the

<table>
<thead>
<tr>
<th>Data</th>
<th>Exogenous</th>
<th>P-value</th>
<th>Stationary</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIn</td>
<td>Constant, Linear Trend</td>
<td>0.3938</td>
<td>F</td>
</tr>
<tr>
<td>FDSIn</td>
<td>Constant</td>
<td>0.1136</td>
<td>F</td>
</tr>
<tr>
<td>SDSIn</td>
<td>None</td>
<td>0.0001</td>
<td>T</td>
</tr>
<tr>
<td>Cln</td>
<td>None</td>
<td>0.2139</td>
<td>F</td>
</tr>
<tr>
<td>FDCIn</td>
<td>None</td>
<td>0.0001</td>
<td>T</td>
</tr>
</tbody>
</table>

**TABLE II. UNIT ROOT TEST FOR SERVICES AND COMPOSITIONS**
ecosystem starts to deal with more complex requirements. We observe that, in the inaugural phase of ProgrammableWeb, most of the compositions only consist of a single service. As time goes, more and more new compositions appear combining multiple services to fulfill more complex needs. For example, the composition Epicenter Document Management created in Aug-2010 uses services such as Zoho, Google Spread sheets, Google Docs List, Google Client Auth, Amazon S3 and Amazon EC2 to help the centralized information storage, sharing and archiving for the business and department management.

\[ n = 963 \]

where \( n \) is the number of services which are used during either month \( t \) or \( t + 1 \).

So we get the average dynamicity of the service usage pattern based on auto-correlation dynamicity.

\[ D_a(t,t+1) = \frac{\sum_{1\leq m\leq t_m} D_a(t,t+1)}{t_m - 1} \]  (11)

where \( t_m \) is the number of the snapshots during our study period.

(2) The cosine-similarity dynamicity of the service usage pattern in two adjacent months can be defined as follows:

\[ D_c(t,t+1) = \frac{\sum_{1\leq m\leq t_m} D_c(t,t+1)}{t_m - 1} \]  (13)

where \( t_m \) is the number of the snapshots of the network.

Figure 7 (a) shows the service usage pattern variation, measured by the auto-correlation dynamicity and the cosine-similarity dynamicity. We can see that the increasing trends for both dynamicity indicators with the growth of the ecosystem. Furthermore, we only consider the top \( n \) popular services (in terms of the number of compositions) and then re-calculate the average variation. As shown in Figure 7 (b), the average dynamicity of the usage pattern decays very fast.

\[ D_c(t,t+1) = 1 - \frac{|Sv(t) \cap Sv(t+1)|}{|Sv(t) \cup Sv(t+1)|} \]  (12)

where \(|Sv(t) \cap Sv(t+1)|\) is the number of services that are used during both months \( t \) and \( t+1 \), \(|Sv(t) \cup Sv(t+1)|\) is the number of services which are used during either month \( t \) or \( t+1 \).

So we can get the average dynamicity of the service usage pattern based on cosine-similarity dynamicity.

\[ D_c = \frac{\sum_{1\leq m\leq t_m} D_c(t,t+1)}{t_m - 1} \]  (13)

where \( t_m \) is the number of the snapshots of the network.

Figure 7 (a) shows the service usage pattern variation, measured by the auto-correlation dynamicity and the cosine-similarity dynamicity. We can see that the increasing trends for both dynamicity indicators with the growth of the ecosystem. Furthermore, we only consider the top \( n \) popular services (in terms of the number of compositions) and then re-calculate the average variation. As shown in Figure 7 (b), the average dynamicity of the usage pattern decays very fast.

\[ D_a(t,t+1) = 1 - \frac{|Sv(t) \cap Sv(t+1)|}{|Sv(t) \cup Sv(t+1)|} \]  (10)

where \(|Sv(t) \cap Sv(t+1)|\) is the number of services that are used during both months \( t \) and \( t+1 \), \(|Sv(t) \cup Sv(t+1)|\) is the number of services which are used during either month \( t \) or \( t+1 \).

So we get the average dynamicity of the service usage pattern based on auto-correlation dynamicity.

\[ D_a = \frac{\sum_{1\leq m\leq t_m} D_a(t,t+1)}{t_m - 1} \]  (11)

where \( t_m \) is the number of the snapshots of the network.

So we get the average dynamicity of the service usage pattern based on cosine-similarity dynamicity.

\[ D_c = \frac{\sum_{1\leq m\leq t_m} D_c(t,t+1)}{t_m - 1} \]  (13)

where \( t_m \) is the number of the snapshots of the network.

Figure 7 (a) shows the service usage pattern variation, measured by the auto-correlation dynamicity and the cosine-similarity dynamicity. We can see that the increasing trends for both dynamicity indicators with the growth of the ecosystem. Furthermore, we only consider the top \( n \) popular services (in terms of the number of compositions) and then re-calculate the average variation. As shown in Figure 7 (b), the average dynamicity of the usage pattern decays very fast.

\[ D_c(t,t+1) = 1 - \frac{|Sv(t) \cap Sv(t+1)|}{|Sv(t) \cup Sv(t+1)|} \]  (12)

where \(|Sv(t) \cap Sv(t+1)|\) is the number of services that are used during both months \( t \) and \( t+1 \), \(|Sv(t) \cup Sv(t+1)|\) is the number of services which are used during either month \( t \) or \( t+1 \).

So we get the average dynamicity of the service usage pattern based on auto-correlation dynamicity.

\[ D_a = \frac{\sum_{1\leq m\leq t_m} D_a(t,t+1)}{t_m - 1} \]  (11)

where \( t_m \) is the number of the snapshots of the network.

Figure 7 (a) shows the service usage pattern variation, measured by the auto-correlation dynamicity and the cosine-similarity dynamicity. We can see that the increasing trends for both dynamicity indicators with the growth of the ecosystem. Furthermore, we only consider the top \( n \) popular services (in terms of the number of compositions) and then re-calculate the average variation. As shown in Figure 7 (b), the average dynamicity of the usage pattern decays very fast.

\[ D_c(t,t+1) = 1 - \frac{|Sv(t) \cap Sv(t+1)|}{|Sv(t) \cup Sv(t+1)|} \]  (12)

where \(|Sv(t) \cap Sv(t+1)|\) is the number of services that are used during both months \( t \) and \( t+1 \), \(|Sv(t) \cup Sv(t+1)|\) is the number of services which are used during either month \( t \) or \( t+1 \).

So we get the average dynamicity of the service usage pattern based on auto-correlation dynamicity.

\[ D_a = \frac{\sum_{1\leq m\leq t_m} D_a(t,t+1)}{t_m - 1} \]  (11)

where \( t_m \) is the number of the snapshots of the network.

Figure 7 (a) shows the service usage pattern variation, measured by the auto-correlation dynamicity and the cosine-similarity dynamicity. We can see that the increasing trends for both dynamicity indicators with the growth of the ecosystem. Furthermore, we only consider the top \( n \) popular services (in terms of the number of compositions) and then re-calculate the average variation. As shown in Figure 7 (b), the average dynamicity of the usage pattern decays very fast.
there are less services used in common. Actually with more services introduced into the system, the diversity of services provides more choices for users, which in turn results into the increased dynamicity. This can be considered as an indicator for the prosperity of the ecosystem. In the meanwhile, there are several services that are strongly positioned in the ecosystem and used almost in every month. For example, the service group which consists of the top 6 popular services (“Google Maps”, “Twitter”, “Flickr”, “YouTube”, “Amazon eCommerce”, and “Twilio”) stays stable during our study period. In fact, these services are the most important services in their respective categories (“Mapping”, “Social”, “Photo”, “Video”, “Shopping” and “Telephony”).

B. Service-Service network (s-s network)

1) Static structure

a) Portion of the service-service network

• Method and Observation

Service-Service network is a network in which nodes represent services, edges represent the relations between services and weights represent the collaboration times (i.e., how many times they show up in the same composition) between services. Figure 8 shows a portion of the s-s network with the edges whose weight is among the top 51 in the overall network and the nodes connected by these edges. The size of a node is proportional to its degree in the overall s-s network and an edge’s width is proportional to the number of compositions that share the two services on its ends. The green diamond represents the services with the top 10 degree in the overall s-s network. The yellow circles represent services that connect with these top ten services while the blue ones represents the services that do not connect with them directly.

Figure 8. A portion of the s-s network.

• Analysis

From Figure 8, we can see that services Google Maps, YouTube, Flickr, Twitter, Amazon eCommerce and Facebook are the core of the s-s network as these services form a clique in which each service not only has a high degree (top 10) but also directly connect with each other with a high weight (top 51). Services Twilio and Twilio SMS collaborate with each other many times (ranked as 4th in terms of the edges’ weight). Actually they are provided by the same provider.

This indicates that there are some core services in the ecosystem. Improving the quality of these services and facilitating the interactions between these services will enhance the usability of the ecosystem.

2) Dynamic metrics

a) Edge weight centrality

• Method and Observation

Using the method to analysis the CF distribution of the composition-service network (see Subsection III.A), we can get the CF distributions of the edge weight by each year. Figure 9 (a) shows the CF distributions by Nov-2011 plotted on log-log axes and Figure 9 (b) shows the dynamic of the absolute exponent value of the CF distribution of the edge weight.

Figure 9. The cumulative distribution of the edge weight. (a) the cumulative distribution of edge weight by Nov-2011. (b) the dynamic of the absolute exponent value of the CF distribution of the edge weight

• Analysis

From Figure 9 (a), we can get a significant power-law distribution of the edge weight with an absolute value 1.58 of the exponent in magnitude for the Nov-2011 snapshot. This demonstrates that a few services pairs are very common in the service-service network, for example, Twilio and Twilio SMS shown in Figure 9. Thus there is a preferential attachment that the used service pair will gain a high probability to be chosen in new compositions. This is because that the users would learn from others and a higher collaboration number indicates a higher trustworthiness of the service pair, so the users may copy the same services pair if they need the similar functionality. Furthermore, from Figure 9 (b), we get a decreasing absolute exponent value and this shows that the strength of the preferential attachment becomes stronger during our study period. This indicates that the pair services may adapt to work with each other in a better way and the consumers may further duplicate the collaboration pattern among these services in new compositions. This phenomenon is referred to as copying process [18].

b) Degree centrality

• Method and Observation

<table>
<thead>
<tr>
<th>Date</th>
<th>N</th>
<th>&lt;k&gt;</th>
<th>&lt;mk&gt;</th>
<th>&lt;kmax&gt;</th>
<th>&lt;wmax&gt;</th>
<th>Cm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan-07</td>
<td>210</td>
<td>27.34</td>
<td>0.131</td>
<td>56.33</td>
<td>58.92</td>
<td>91.40%</td>
</tr>
<tr>
<td>Jan-08</td>
<td>325</td>
<td>23.95</td>
<td>0.074</td>
<td>64.42</td>
<td>70.84</td>
<td>90.15%</td>
</tr>
<tr>
<td>Jan-09</td>
<td>454</td>
<td>20.9</td>
<td>0.0466</td>
<td>69.17</td>
<td>76.19</td>
<td>87.00%</td>
</tr>
<tr>
<td>Jan-10</td>
<td>624</td>
<td>23.04</td>
<td>0.031</td>
<td>88.44</td>
<td>97.48</td>
<td>87.28%</td>
</tr>
<tr>
<td>Jan-11</td>
<td>772</td>
<td>23.54</td>
<td>0.032</td>
<td>102.97</td>
<td>115.11</td>
<td>87.31%</td>
</tr>
<tr>
<td>Nov-11</td>
<td>963</td>
<td>22.56</td>
<td>0.023</td>
<td>111.05</td>
<td>123.87</td>
<td>85.88%</td>
</tr>
</tbody>
</table>

TABLE III. THE SERVICE-SERVICE NETWORK METRICS BY QUARTER
We compute a set of network metrics which is commonly employed in weighted network analysis [19]: the number of services in the network \( N \); the average degree \( <k> \); the normalized degree \( <nk> \) which is the average degree divided by the maximum possible degree; the mean of the average nearest neighbors’ degrees per service \( <k_{nn}> \); the mean of weighted average nearest neighbors’ degrees per service \( <wk_{nn}> \); and the size of the largest component \( Com \). Since we consider dynamics of the s-s network here, we capture six snapshots in Jan-07, Jan-08, Jan-09, Jan-10, Jan-11, and Nov-11, respectively. Table III shows the summary of these metrics which are calculated for each of the 6 service-service network snapshots.

- **Analysis**

  From Table III, we can see that the \( <k> \) and \( <nk> \) are gradually shrinking. \( <k_{nn}> \) and \( <wk_{nn}> \) are increasing and \( <wk_{nn}> \) is always larger than \( <k_{nn}> \). This makes sense because the popular services will not only connect with many services but also collaborate with other services together most of the time. The decreasing \( Com \) demonstrates that many peripheral services are invoked individually. In the future, the compositions which invoke the peripheral services and the core services in the main component can be suggested as an innovation direction.

  **c) Network assortative mixing**

  - **Method and Observation**

    For the snapshot of Nov-2011, we calculate the \( <k_{nn}> \) of each service and then we can get the relation between \( <k_{nn}> \) and \( k \) in the network, as Figure 10 (a) shows. Figure 10 (b) shows the dynamic of the mixing pattern indicator \( r(g) \) [20] of the service-service network snapshot in month.

- **Analysis**

  From Figure 10 (a), we can see that \( <k_{nn}> \) has a negative correlation with \( <k> \) which shows that the network is a degree-disassortative network [19]. Also, as shown in Figure 10 (b), the \( r(g) \) values during our study period are all less than 0 which also confirms the disassortative characteristic of the service-service network. This property indicates that the services with a lower degree intend to build connections to the services with a higher degree. Actually, when a new service is released into the ecosystem, it’s more likely that it will work with the most popular existing services (i.e., those that have a higher degree). This trend becomes more obvious as the \( r(g) \) decreases along with time.

V. **Related Work**

This paper is an empirical study of a mashup ecosystem, i.e. ProgrammableWeb. We try to introduce a network-based method to analyze the static structure and dynamic evolution of the ecosystem. Our work is at the cross-road of the service system and the complex network analysis.

On one hand, complex network analysis is a powerful tool to understand the large scale networks [21]. Many studies have been done to analyze networks such as scientists’ correlation network, mobile phone calling network [7], the Internet [8], the email network [4] as well as the metabolic network [9]. Metrics such as connectivity, clustering coefficient, average minimum path distance, the small-world and the scale-free properties are investigated. Furthermore, graph sequences have been used to describe the snapshots of networks to study the evolution of networks [8].

On the other hand, the increasingly growth of Web services has attracted much attention. Some works focus on the evolution of the intra-dependencies of one service and inter-dependencies among services [22, 23]. There are some existing works on analyzing ProgrammableWeb. Yu and Woodard presented a preliminary result in studying the properties in ProgrammableWeb and proved that the cumulative frequency of APIs follows power law distribution [24]. Weiss and Gangadharan analyzed the complementary nature of APIs in ProgrammableWeb and found that the complexity of the mashups drives the development of the mashup platform [11]. Wang et al emphasized on mining mashup community from users’ perspective by analyzing the User-API and User-Tag network in ProgrammableWeb [25]. Our previous work [10] studied the usage pattern of services in scientific workflows stored at myExperiment based on the social network analysis.

Compared with the existing studies regarding the ProgrammableWeb data set, our work not only quantifies the static structure but also pay special attention to the dynamic metrics of the ecosystem. This can not only help to understand the current usage and evolution trace of the ecosystem, but also help to build a more realistic evolution model for the ecosystem to predict its future behavior.

VI. **Conclusion**

This paper presents the methodology to quantitatively characterize a service-mashup ecosystem and an empirical study on ProgrammableWeb. We present the motivation of the evolution of this service-mashup ecosystem and then several networks are constructed for our study. Applying the network analysis technologies, we comprehensively explore the ecosystem and investigate both the static structure to understand the current usage pattern and the dynamic metrics to track the evolution of the ecosystem.

In our study, we find in the service-composition network, the increasing number of services in the ecosystem can
inspire the new compositions. This suggests that the managers of the ecosystem should pay attentions to increase the diversity of the services. However, the highly concentrated distribution of the service popularity suggests that the service providers should think about how to improve the usage of their services. Also, the reuse rate of services is low and the advanced use of many services together is still rare. This indicates that the service-mashup ecosystem is still under-utilized and calls for an effort to improve its usage.

Meanwhile, although the usage patterns of the ecosystem are very dynamic, there are several services, such as Google Maps, YouTube, Flickr, Twitter, Amazon eCommerce, Facebook, Twilio and Twilio SMS, which form the core of the ecosystem. These services are not only used frequently but also constantly together. Thus improving the quality of these services and facilitating the interactions between these services will enhance the usability of the ecosystem.

Furthermore, consumers of the ecosystem intend to select the popular services and service pairs, which can be referred to as the preferential attachment for the service selection and the copying process for the compositions. Also the disassortative characteristic indicates that the services with low usage intend to build connections to those with a higher one. These selection mechanisms will help us to build a more realistic network-based evolution model for the ecosystem.

The hidden knowledge we discover in this work can help understand the current usage pattern and the evolution trace of the ecosystem. In the future, we will try to develop a rigorous theoretical framework to trace and predict the evolution of the service ecosystem, and to suggest how to improve the system.

ACKNOWLEDGMENT

This work is partially supported by the National Natural Science Foundation of China (No.61174169 and No. 61033005) and the National Key Technology R&D Program of China (No. 2010BAH56B01).

REFERENCES: