Calculating Service Fitness in Service Networks*

Martin Treiber[1], Vasilios Andrikopoulos[2], and Schahram Dustdar[1]

[1] Distributed Systems Group, Vienna University of Technology
[2] ERISS, Dept. of Information Systems and Management, Tilburg University
treiber@infosys.tuwien.ac.at, v.andrikopoulos@uvt.nl,
dustdar@infosys.tuwien.ac.at

Abstract. Inspired by the biological perspective of service ecosystems, we propose to define the fitness of services in service networks. In our work, we show how to calculate the service fitness from the provider perspective using locally available information as a reflection of the position of the service in the service network. For that purpose we define a fitness corridor with upper and lower bounds that confine the service fitness area. After establishing a fitness corridor, we show how to calibrate the fitness calculation parameters to better reflect the service market and how to use the calculated fitness trends for making decisions about the provisioning of a service.

1 Introduction

The proliferation of the SOA paradigm into the highly competitive and volatile world of business has naturally led to the necessity for quickly evolving services that aim to fulfill the changing requirements set by the market. The evolution of services [1,2] and the evolution of the context within which the services operate - the *service ecosystem* [3] - create a number of challenges that need to be addressed.

In this context, we consider a Web service ecosystem a set of potentially overlapping service networks [4] of service consumers. Each service network represents a pool of services that is available to a service consumer. A service consumer selects services from this pool for later use in service compositions or for single service invocations (see Figure 1). From the perspective of the service provider, service networks have fundamentally different characteristics. To illustrate the perspective of the service provider, consider the following example: a company which is providing software services is interested that their offerings reach as many customers as possible. For that purpose the company has a number of disparate services in its portfolio targeted at different customers with different characteristics and needs and may therefore participate in more than one service networks. In that sense, the services offered by a service provider may be part of different service networks.

* The research leading to these results has received funding from the European Community's Seventh Framework Programme FP7/2007-2013 under grant agreement 215483 (S-Cube).

A. Dan, F. Gittler, and F. Toumani (Eds.): ICSOC/ServiceWave 2009, LNCS 6275, pp. 283–292, 2010.
© Springer-Verlag Berlin Heidelberg 2010

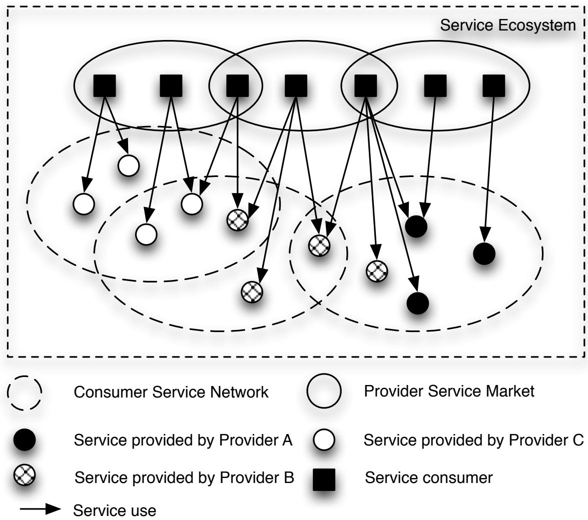

Fig. 1. Service Ecosystem

Consequently, we will refer to the provider's *service market* (Figure 1) as a particular subset of the service ecosystem in which the provider offers his services to a set of consumers. More specifically, a provider's service market can be regarded as a segment of a service network where other (competing) service providers offer similar or equivalent services to a specific consumer set. In contrast to the service consumer, a service provider may have limited knowledge about the service network and can only estimate the size of service markets [5].

As a solution to this problem, we propose the use of a utility function that uses available information at the provider side to calculate a *fitness corridor* for services with regard to service networks and service markets respectively. The fitness corridor defines an upper bound for the maximum service network share that can be achieved with a service with the given information. The calculation of the lower bound requires stochastic or marketing methods that are capable of dealing with uncertain information. The observation of a given service fitness corridor over time serves as input for provisioning decisions for the service provider. For example, (negative) fitness development of a service might lead to the decision to decommission the service. Apart from provisioning, we envision the use of service fitness for prediction of performance trends of the service in terms of market share.

The rest of the paper is organized as follows. We introduce our approach in section 2 and illustrate our assumptions concerning the behavior and structure of service networks. In section 3 we define service fitness and provide the means to calculate it based on the assumptions of the previous section and show its usefulness with a concrete example from an existing company. We conclude the paper with related work in Section 4 and an outlook in Section 5.

2 Service Networks Characteristics

Service networks emerge around businesses and exhibit by their nature a dynamic behavior. Service providers and consumers may enter and leave the service network as the service network expands and shrinks over time, depending on external factors like consumer demand, profit margins, etc. In addition, a particular service provider that participates in the network does not have in principle up-to-date global knowledge of the service network without the presence of a centralized authority for this purpose. As we will discuss in the following sections in more detail, these two factors (constant change and partial knowledge) limit the calculation of fitness to only locally available information.

2.1 Roles and Structures in Service Networks

Traditional Service-Oriented Architectures [6] define three distinct roles, (i) the service provider, (ii) the service consumer and (iii) the service broker (registry). We follow this characterization but focus on the relation between service providers and service consumers. We can distinguish between (i) service providers that offer services to (ii) a set of consumers (or alternatively, *customers* of the service) which are in turn a subset of (iii) the potential customers (see Figure 2).

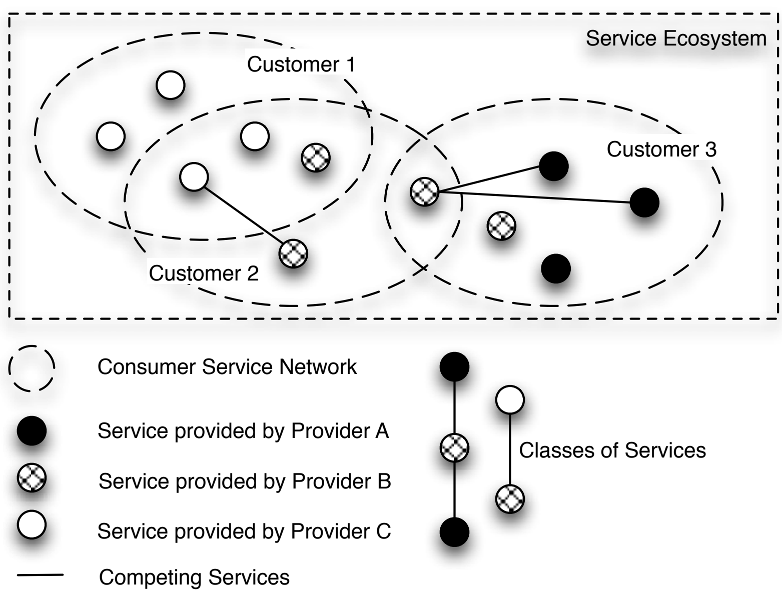

Fig. 2. Emerging classes in service networks.eps

2.2 Knowledge in Service Networks

Crucial for the calculation of service fitness is the available information. In our work, we follow a semi-open approach to define service networks, i.e., we assume that there exists a finite number of service network members which may enter and leave the service network at any time. Furthermore, we assume that there is limited information available for each service network member. Lessons learned from the failure of public service registries [7] impose that the existence of a central entity like a public registry is not very likely which leads to the situation as where very service network member has an *observable universe* in which information about the service network can be obtained. During the life time of a service network we can observe various changes to it: the introduction of new services or service versions, the decommissioning of services, or the entering of new service providers may all result in dramatic changes of service networks. We observe these dynamics in our proposed approach by events that occur in service networks and lead to changes of the service network structure. In particular, we consider the following events that are of interest for service networks:

NS New service - A new service enters the network.
NV New service version - An existing service is modified and a new version of the same service is published. In these cases, the original version can also be available in parallel.
DS Decommission of service - A service is removed from the service network.
DV Decommission of service version - A (older) service version is removed from the service network.
NP New service provider - A new service provider enters the service network and offers new services in the service network.
LP Leaving of service provider - A service provider leaves the network and removes its services from the service network.

The evolution of a service network can therefore be seen as a series of modifications to it, in a similar fashion as the evolution of a service itself can be perceived as a series of unambiguous changes to it [1]. It is important to notice that we explicitly consider *time* of central importance in our approach. We measure time in discrete units that are defined by the service provider, enabling each service provider to monitor his services privately without the need for further synchronization with other service providers.

3 Service Fitness in Service Networks

In this section, we introduce the notion of *service fitness* as a measure of the success of a service provider in a service network. It is important to notice that the notion of fitness of services depends highly on the context. For instance, an ordering service might be fit in the context of computer part supplier networks, but not in customer service networks since the service might be tailored for the requirements of the part supplier service network.

Changes to the context of the services are reflected on the market share of the service and we can therefore observe them as changes to the service fitness. These changes take place within a certain boundary which we refer as *fitness corridor*. A fitness corridor is defined by an upper bound that denotes the best possible fitness (as calculated using the available data) and a lower bound which is calculated with stochastic methods like the Monte Carlo method [8] or with marketing research methods [5]. We propose to use a utility function that takes local information into account and lets service providers calculate the fitness of their services independently from other service providers. In order to calculate the fitness for a given service, we use Equation 1 which gives the basic definition of service fitness:

$$f^m(\tau) = Actual\ Use/Potential\ Use \tag{1}$$

In the formula, τ denotes the *time window* i.e. the time interval for which fitness is calculated, m defines the observation window i.e. how many time windows are included in the calculation of the fitness, the $Potential\ Use$ is the estimated number of possible service customers and $Actual\ Use$ is the number of actual service customers. $f^2(1h)$ for example denotes that the formula will be calculated for $m = 2$ windows of $\tau = 1$ hour each. We always normalize the output of the fitness function to a value of the interval $[0 \ldots 1]$. Depending on the amount of information that we take into account while estimating $Potential\ Use$ we can come with a theoretical upper and lower bound for fitness at any given time interval.

More specifically, we establish an upper bound for the fitness of a given service by collecting information about the status of the service and the perceived status of the network. We use (exclusively) data from the event logs of each service (version) for a given time period. In particular, we are interested in two types of events, (i) the request for the service description (e.g. the WSDL file itself) and (ii) the invocations of the service by different consumers. Using existing tools like Webalizer [1] we can access this information easily. Figure 3 shows the result of the analysis for the services of ikangai for August 2009. By using these available data, a service provider can aggregate the information from the different services and their versions in his portfolio. In order to establish the lower bound of a fitness corridor, we rely on estimation techniques which can be based on market research. In the case of ikangai solutions for example we derived their market size estimation by simply estimating the hits of their homepage and by investigating relevant forums and social networking outlets for interested users. ikangai solutions estimated an average of 500 hits per day during August, thus being well over the actual average number of 201 hits per day.

3.1 Calculating the Fitness of Services

We now show how to apply the basic fitness formula to calculate the fitness of services in service networks. For the upper bound of the fitness function we replace equation 1 with equation 2 that incorporates event log information:

[1] http://www.mrunix.net/webalizer/

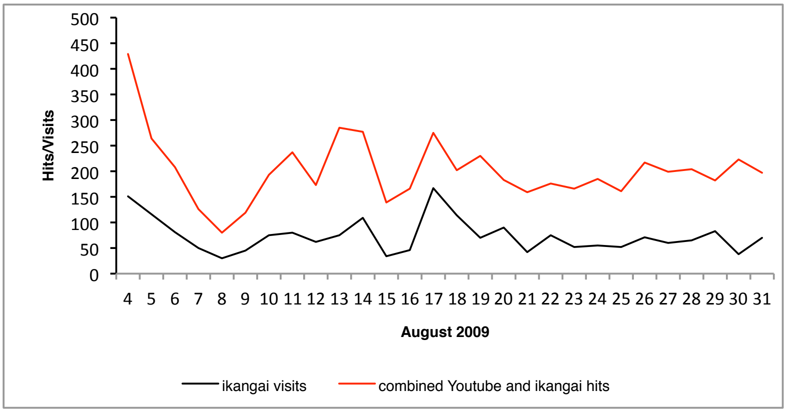

Fig. 3. ikangai web page statistics. The data represents the cumulated youtube hits and ikangai web page visits.

$$f^m(\tau) = \sum Inv(i)/(\sum_m Req(i) + \sum_m q(i)) \tag{2}$$

$Inv(i)$ is a function that returns 1 if a customer i invokes the service at least once during the given time windows m and 0 otherwise. Similarly, $Req(i)$ is a function that returns 1 if customer i requests the service specification (at least once) during m time windows and returns 0 otherwise. *q(i)* returns 1 if customer i invokes the service at least once in the m time windows without a request for a service specification and returns 0 in all other cases. Notice that if global knowledge about the potential customer class is available, then $Req(i) = 1$ and $q(i) = 0$ for all customers and we can calculate the overall fitness of service in a service network.

3.2 Calibration of Service Fitness

To illustrate the effects of different observation windows we've depicted the calculation of service fitness for the same scenario and for three different observation windows. Figure 4. f^1 depicts the use of the last two intervals τ_i and τ_{i-1}, f^{10} (Figure 5) uses the last 10 intervals, and f^∞ (Figure 6) uses all available intervals to calculate the fitness of a service. As shown in the figure, the history of the service has a direct impact on the calculated service fitness by smoothing the fitness function. Notice that we didn't change the function for establishing the lower bound of the fitness corridor. Depending on the dynamics of the service network different observation windows can be useful. In this context we speak of *re-calibration* of the service fitness function, since the past history is not considered in the calculation of the service fitness. For example, if a service provider enters a network with high fluctuation, the service provider might use a small observation window for its fitness function (e.g., f^2). Consider also the case of a service provider that introduces a new service version into the service network.

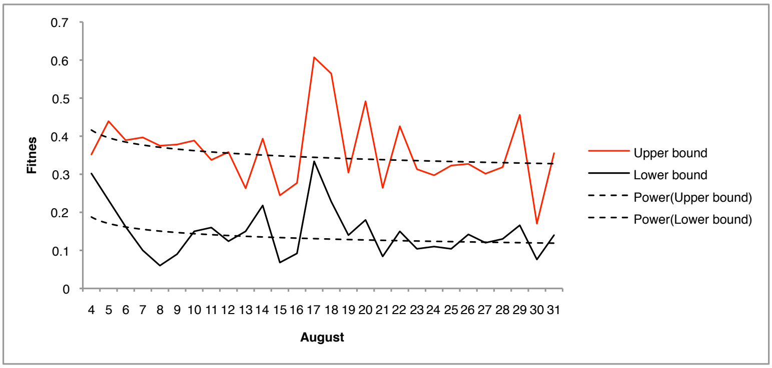

Fig. 4. Fitness corridor using f^1

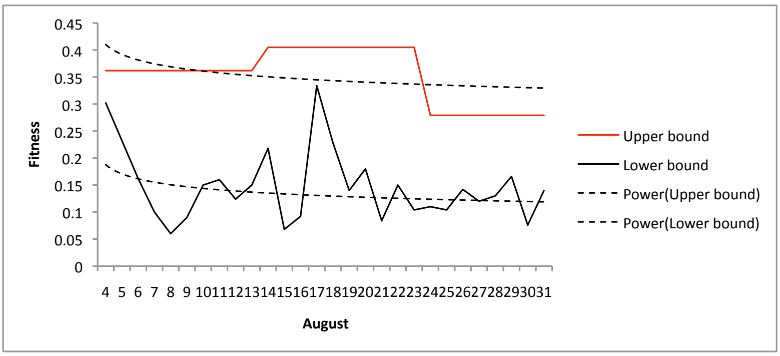

Fig. 5. Fitness corridor using f^{10}

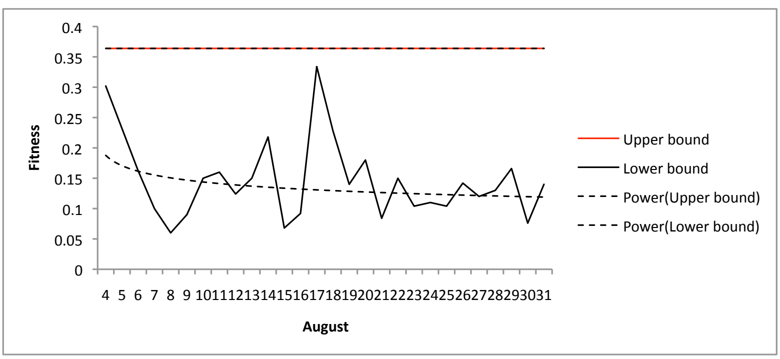

Fig. 6. Fitness corridor using f^∞

The provider is interested in ranking [9] both service versions and thus needs to re-calibrate the fitness function of the older service accordingly.

Another reason for a re-calibration of the observation window is the *fitness of the (fitness) function* itself. The chosen observation window might not return adequate fitness values for the given network dynamics. For example, a provider might use an observation window of only one time window (τ_0) which can result in good service fitness (see Figure 4) but may not be accurate enough to capture the changing dynamics of the service fitness over time.

The length of the time windows is also set by the service provider and is assumed to be of constant over time (in the example being 24 hours). As there is no global notion of time in the service network, each provider can decide on his own about the length of the time interval. Note that the length of the interval is critical for the calculation of service fitness.

3.3 Using Local Fitness for Provisioning Decisions

From the definition of equation 1 it becomes apparent that the closer the service to the upper bound of fitness is, the better the performance of the service with respect to its (estimated) market share is. Additionally, the width of the fitness corridor reflects the actual market share of a service provider. It is worth noticing, that big discrepancies between the upper and lower bounds of fitness require of the service provider either to change the estimation of the service market size or to consider modifying or decomissioning the service.

For prediction purposes we can also select portions of the fitness graph that can serve as decision criteria. For instance, during the observation, we can use functions that predict the time when the service fitness falls below a pre-defined threshold. Depending on the model and on the dynamics of the Web service ecosystem, we can use approximations to calculate the time when a Web service might fall below a critical fitness threshold and requires decomissioning. To illustrate this, recall the example from the previous section. The trend lines obviously show a downward tendency and thus it can be expected that - if the trend continues - the fitness will fall below 10 per cent within the next two months.

Quality of service criteria could also be used for these types of decisions since they offer fine grained metrics for the comparison of services. However, in service networks, such mechanisms may not be applicable due to the lack of information from competitors on the side of the service provider. Even if a service provider constantly monitors the QoS, these might not change. For example, a service may be available with no exception, the response may remain constant, and the service use also may fluctuate within predefined boundaries. However, the service ecosystem may change (e.g., new customers, new service providers, service providers leave) without having impact on QoS attributes. Nevertheless, in business terms, a service that looses market share cannot be considered as competitive (fit) and at some point in time the service might even become obsolete.

4 Related Work

Software evolution has been subject to studies for several decades. In this regard, the work of Lehman [10] [11] and Cook [12] is of relevance for our work. In particular, the postulated constant change of software is reflected by our notion of changing service fitness. As noted by Nehaniv et. al. [13] evolutionary concepts have to be carefully transferred into the realm of software engineering. The same is true for our work with regard to service fitness. Thus, we define service fitness as fraction between use and interest of a given service in a service network.

More general, Value Networks [14] are of interest when business aspects of service networks are studied, i.e., the value that can be generated by such networks. Likewise, from a business oriented perspective perspective, the work of Basole et. al. [15] is of relevance for our work. Their conceptual approach models service value and it's exchange in service value networks. In a similar fashion, Caswell et. al. [16] explain how the concept of value be used to study service networks.

Our work is similar to that latter work in spirit, but differs in the approach. We assume that available service network information is limited and the calculation of the importance or value of a service can only be made with local data. Consequently, we consider service fitness as utility function on the service provider side that can be combined with additional functions, like cost or turnover.

Complementary to our work, Bitsaki et. al.[4] present a service network notation which describes the interactions of service network participants. Our work shares similarities in terms of having the goal to provide methods to calculate the value or fitness of given services in service networks. The main difference is that our proposed fitness function can be combined with other utility functions, like cost or generated value. Furthermore, since complete service network information might not be available, our approach is able to calculate local fitness without the requirement of complete network information.

5 Conclusion and Future Work

In this paper we showed how to calculate the fitness of services from a provider perspective in service networks using locally available information for that purpose. In particular, we used the calculated service fitness to define a fitness corridor with upper and lower bounds that confine the service fitness area. After having established a fitness corridor, we showed how to calibrate the fitness calculation parameters to better reflect the service market and how to use the calculated fitness trends for making decisions about the provisioning of a service.

In future work we are going to evaluate our approach by using a Web service testbed [17] to provide simulations of service networks and illustrate the consequences of changing service fitness with regard to different service selection policies in service networks. A direct application of this process would also allow to model fitness of composite services in service networks which we are going to investigate in future work. Furthermore we plan to combine our fitness utility function with other similar functions (e.g., number of hits or generated value) in order to provide additional metrics for estimating the service fitness.

References

1. Papazoglou, M.P.: The challenges of service evolution. In: Bellahsène, Z., Léonard, M. (eds.) CAiSE 2008. LNCS, vol. 5074, pp. 1–15. Springer, Heidelberg (2008)
2. Treiber, M., Truong, H.L., Dustdar, S.: Semf - service evolution management framework. In: 34th Euromicro Conference on Software Engineering and Advanced Applications, SEAA 2008, pp. 329–336 (2008)
3. Barros, A.P., Dumas, M.: The rise of web service ecosystems. IT Professional 8(5), 31–37 (2006)
4. Bitsaki, M., Danylevych, O., van den Heuvel, W.J., Koutras, G., Leymann, F., Mancioppi, M., Nikolaou, C., Papazoglou, M.: An architecture for managing the lifecycle of business goals for partners in a service network. Towards a Service-Based Internet, 196–207 (2008)
5. Macfarlane, P.: Structuring and measuring the size of business markets. International Journal of Market Research 44(1), 7–30 (2002)
6. Alsonso, G., Casati, F., Kuno, H., Machiraju, V.: Web Services – Concepts, Architectures and Applications. Springer, Heidelberg (2004)
7. Microsoft: Uddi shutdown (2006)
8. Metropolis, N., Ulam, S.: The monte carlo method. Journal of the American Statistical Association 44(247), 335–341 (1949)
9. Ahmadi, N., Binder, W.: Flexible matching and ranking of web service advertisements. In: MW4SOC 2007: Proceedings of the 2nd workshop on Middleware for service oriented computing, pp. 30–35. ACM, New York (2007)
10. Lehman, M.M., Ramil, J.F.: Software evolution: background, theory, practice. Inf. Process. Lett. 88, 33–44 (2003)
11. Lehman, M.M.: Laws of software evolution revisited. In: Montangero, C. (ed.) EWSPT 1996. LNCS, vol. 1149, pp. 108–124. Springer, Heidelberg (1996)
12. Cook, S., Harrison, R., Lehman, M.M., Wernick, P.: Evolution in software systems: foundations of the spe classification scheme: Research articles. J. Softw. Maint. Evol. 18(1), 1–35 (2006)
13. Nehaniv, C.L., Hewitt, J., Christianson, B., Wernick, P.: What software evolution and biological evolution dont have in common. software-evolvability, 58–65 (2006)
14. Allee, V.: Reconfiguring the value network. Journal of Business Strategy 21(4) (2000)
15. Basole, R.C., Rouse, W.B.: Complexity of service value networks: conceptualization and empirical investigation. IBM Syst. J. 47(1), 53–70 (2008)
16. Caswell, N.S., Nikolaou, C., Sairamesh, J., Bitsaki, M., Koutras, G.D., Iacovidis, G.: Estimating value in service systems: a case study of a repair service system. IBM Syst. J. 47(1), 87–100 (2008)
17. Juszczyk, L., Truong, H.L., Dustdar, S.: Genesis - a framework for automatic generation and steering of testbeds of complexweb services. In: ICECCS 2008: Proceedings of the 13th IEEE International Conference on on Engineering of Complex Computer Systems, Washington, DC, USA, pp. 131–140. IEEE Computer Society, Los Alamitos (2008)