Soability

A Model for the Strategic Evaluation of an IT Environment's Ability to Support a Service-Oriented Architecture

Alexander Krähenbühl

Corporate Systems Architecture
Kühne + Nagel
Grosser Grasbrook 11-13
DE-20457 Hamburg

alexander.kraehenbuehl@kuehne-nagel.com

Burkhard Stiller

Communication Systems Group
University of Zurich
Binzmühlestrasse 14
CH-8050 Zurich

stiller@ifi.unizh.ch

Abstract: This document presents the Soability model, which assesses on a strategic level how well an IT environment is able to support an SOA based on Web Services. The model is designed to support the answering of typical questions that decision makers in companies face. By so doing the model addresses a lack of methodological support for stakeholders who need to plan an SOA holistically.

1 Introduction

The introduction of an SOA, e.g. [KR04, BO04], implies fundamental changes to an IT environment and not only technical aspects are of concern, but also organisational ones. The research field as well as the industry constantly improve SOA concepts and products, but there is still a lack of methodological support for decision makers within companies. This is a surprising fact considering that these are the key persons to start or end SOA initiatives. In a collaboration between the logistics provider Kühne + Nagel [KN07] and the Communication Systems Group, Department of Informatics of the University of Zurich [CG07] the Soability model was created to address this issue [KR06].

2 The Soability Model

The Soability model assesses on a strategic level how well the elements of an IT environment are able to support an SOA based on Web Service technologies. An IT environment includes applications, infrastructure, organisation, and exogenous parameters. This document will focus on the application layer exclusively. More details on the other layers as well as further background definitions may be obtained from [KR06].

Soability introduces the notion of a service candidate in order to assess applications. A service candidate is defined as a functionality contained in an application and the access to that functionality. These candidates are analysed in three orthogonal dimensions:

- The solution profile indicates the value of a service candidate in an SOA.
- The access profile mirrors how well the candidate realises a Web Service.
- The simplicity profile shows the demands of a candidate regarding the environment.

3 Data Collection and Evaluation

A concrete Soability model instance must be based on existing IT environment data. Services are highly domain specific and questionnaires are proposed to gather the domain expert knowledge. Based on statements found in the literature, mapping rules have been elaborated that convert the answers into numeric values. For example, application owners were asked how many other applications could possibly use a functionality today, regardless of any technical barriers. Figure 1 shows the respective mapping rule.

$$C_{\text{PotentialUsage}} = f(a) = \begin{cases} 1 & (a = 5+) \\ 0{,}87 & (a = 4) \\ 0{,}73 & (a = 3) \\ 0{,}6 & (a = 2) \\ 0 & (otherwise) \end{cases}$$

Figure 1: Mapping rule derived from statements in the literature

Soability uses equations to mimic the effects that individual values are thought to have in an SOA. These equations are proposals and are, whenever possible, based on statements found in the literature, i.e. [KR04] and [ER04]. Additional assumptions and restrictions are documented and explained in the context in which the model was implemented. For a detailed explanation see [KR06]. Sensitivity analysis was performed for key equations to quantise the impact of the weights chosen on the model output. Solving the equations results in ordinal scale measures. As an example of such an equation, the solution profile is outlined hereafter.

$$C_{\text{SolutionProfile}} = C_{\text{Reusability}} \times \frac{\left(w_1 \times C_{Availability} + w_2 \times C_{Scalability} + w_3 \times C_{Modularity}\right)}{w_1 + w_2 + w_3}$$

The left factor models the value of a service candidate from provider's perspective, which is to reuse the service instead of implementing functionality from scratch. The right factor models the value from a requestor perspective, which is that services are available and future proof. No benefit can be obtained if the provider or the requestor do not perceive any value in the service and the multiplication operation ensures this behaviour. Depending on various preferences and the context in which the model is applied, weights must be chosen for the different properties.

In order to disburden the decision maker from performing the calculations manually, an implementation was made in the enterprise architecture tool System Architect [TE07].

4 Evaluation

During the data collection phase at Kühne + Nagel twelve application questionnaires were filled out by application owners, yielding a total number of 39 service candidates. This sample was chosen to test the model against a wide variety of applications with implementation platforms ranging from Report Program Generator (RPG) to Java , architectures ranging from host to distributed systems, and tasks ranging from core business to reporting.

Statistical analysis showed uniform distributions of the solution and simplicity profiles with correlation coefficients $r = 0.186$ and $r = 0.003$ respectively. The major explanation for this is the selection strategy already mentioned. The access profile displays a correlation coefficient of $r = -0.744$, which is plausible since only few Web Services were contained in the sample. Hence most candidates show no Web Service characteristics, however, some Web-based applications partially do, e.g. by using XML messages. The two actual Web Services contained in the sample score the highest access ratings.

The correlation between the profile was analysed to ensure that they are orthogonal. Evidence for this could be found with correlation coefficients being at $r = 0.36$ or below.

The implementation of the model in the tool System Architect allows for graphical and non-graphical evaluation of the data. Non-graphical analysis includes sorting of profiles to deduce favourable implementation sequences as well as impact analysis [KR06]. A graphical representation of the three profiles allows to understand a service candidate in more detail and to quickly identify stereotypes. This technique is presented in the following subsections.

4.1 Service Stereotype

A service shows high ratings in all profiles, and is thus a valuable solution in an SOA, implemented as a Web Service, and not too demanding regarding specialised infrastructure or organisation. Figure 2 depicts an example found at Kühne + Nagel. “DMS Access” is one of the few established Web Services. It provides access to a document management system storing information valuable for many applications.

4.2 High Potential Stereotype

This service would be a valuable solution in an SOA and it is moderate in its demands regarding the environment. It is simply missing a Web Service interface to be accessed which makes it a good starting point for an SOA initiative. Figure 2 outlines a match found in the sample. “Search Shipments” provides read access to data, which is not sensitive and useful in several domains. It is scalable and modular but may currently not be accessed from outside its owning application.

4.3 Coconut Stereotype

The coconut stereotype contains a valuable functionality but has high demands on the environment so that it is hard or impossible to be turned into a service. This stereotype should not be the target of an SOA project unless the enterprise has gained experience, e.g. with the implementation of the easier high potential stereotype, or has a sophisticated SOA infrastructure at its disposal. Figure 2 presents the coconut “Display Result Tables”, which is comparable to the high potential candidate “Search Shipment” but is more demanding in regards to performance, security, and organisational aspects. Please note how the integrated evaluation of the solution, access, and simplicity profile allows to distinguish these different candidates.

4.3 Fake Service Stereotype

A fake service offers no benefits to an SOA but is implemented as a Web Service. It may result from interest in Web Service technology without an SOA in mind, an unlucky choice for an SOA project, or a former service that has collapsed, e.g. due to the load imposed on it by requestors. Figure 2 shows a match found in the Kühne + Nagel sample. “Gate Booking” was developed as a Web Service for technical reasons, but has no reuse potential within the application landscape.

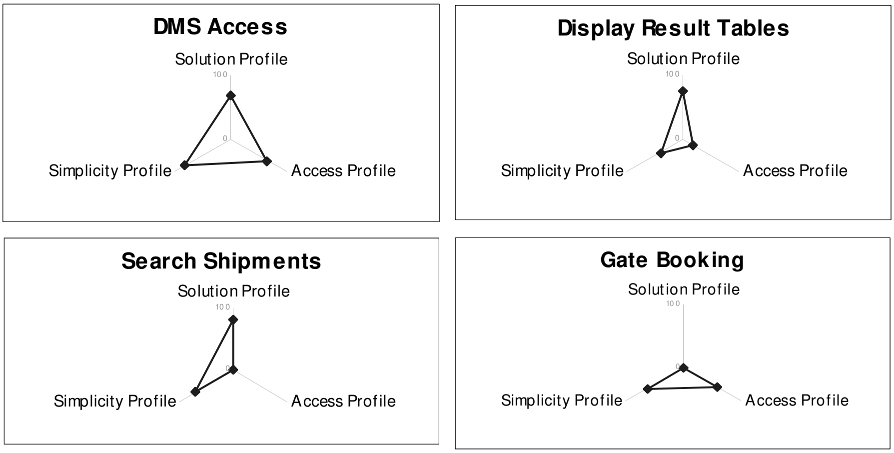

Figure 2: Service Stereotypes Found in the Kühne + Nagel Sample

4.5 Soability Maps

To provide decision makers with a holistic picture of the environment’s ability to support an SOA, service candidates can be depicted in a map. Thresholds for different profiles are used to colour-code service candidates. This makes it possible to identify quickly, which applications are generally not able to support an SOA approach and possible require a fundamental redesign first. By applying the model repeatedly over time, the impact of decisions and the progress of the service-orientation becomes visible.

5 Conclusions

The Soability model was applied successfully to the Kühne + Nagel application landscape. For the given set of service candidates the model produced plausible results that matched the expectations of Kühne + Nagel domain experts. The sample of candidates revealed that the logistics provider has several service candidates that would be valuable services in an SOA. Other candidates require a more sophisticated environment to be leveraged, and some others will hardly ever make sense in this kind of architecture.

The questionnaire-based approach allows to evaluate roughly four service candidates per hour and can be parallelised. The sample could hence be evaluated with little expenditure of time and financial effort, which allows to create a repository filled with SOA-relevant information, on top of which holistic decisions can be made. This is an important advantage compared to approaches which focus on one application at a time and may thus neglect synergies of the big picture.

The complex domain of SOA required several assumptions and simplifications to be made. Thus, the mathematical approach used shall not be confused with precision. Soability is designed to provide a first triage of different possibilities a company has when implementing an SOA but it is no replacement for an in-depth analysis thereafter.

Due to the novelty of the approach, experimental data to work with could not always be found in the literature. In these cases, the Kühne + Nagel case was used to substantiate the rules of measurements and their respective equations. Future experiments are proposed, e.g. based on the mapping rules, to further tune the model.

References

[BO04] Booth, D.: Web Services architecture. http://www.w3.org/TR/ws-arch/ , 2004, site last accessed on 19.06.2007

[ER04] Erl, T.: Service-Oriented Architecture A Field Guide to Integrating XML and Web Services. Prentice Hall PTR 2004

[KN07] Kühne + Nagel (AG & Co.) KG: Internet Presence. http://www.kuehne-nagel.com, site last accessed on 19.06.2007

[KR04] Krafzig, D. et al.: Enterprise SOA: Service-Oriented Architecture Best Practice. Prentice Hall PTR 2004; Pp. 1-10.

[KR06] Krähenbühl, A.: Soability – A Model for the Strategic Evaluation of an IT Environment's Ability to Support a Service-Oriented Architecture. https://www.ifi.unizh.ch/fileadmin/site/teaching/Diplomarbeiten/Abgeschlossene_Diplomarbeiten/Jahrgang_2006/Kraehenbuehl_Alexander.pdf , 2006, site last accessed on 19.06.2007

[TE07] Telelogic System Architect: Internet Presence. http://www.telelogic.de/products/systemarchitect/index.cfm, site last accessed on 19.06.2007

[CG07] Communication Systems Group, Department of Informatics, University of Zurich, Switzerland: Internet Presence. http://www.csg.uzh.ch/, site last accessed on 19.06.2007