Needs-driven service bundling in a multi-supplier setting

de Kinderen, S.

2010

document version
Publisher's PDF, also known as Version of record

Summary

This thesis introduces a methodology that allows for the (semi-)automated composition of (possibly multi-supplier) bundles of commercial services, based upon a complex customer need. As a concrete application scenario, imagine that a customer fills in his internet needs on a website (for example: 'e-mailing with little spam'), whereupon this website responds with suitable bundles of internet services (for example: bundles containing e-mailing and spam filtering services).

Services are often offered as a bundle, a package of individual services that is sold at a single price. First and foremost, such bundles are required because services by themselves often satisfy a need only partially, but together satisfy this need completely. As an example, consider that an e-mail service is often supplemented by a spam filter, so as to avoid that the customer is flooded by unwanted e-mails.

It becomes more and more common to decompose a bundle into its constituent services, and to recompose these services into a bundle that satisfies specific customer needs. Consider packages of internet services. Where these packages used to be generic (usually, the service 'web hosting' was included by default) , today a customer can often choose to compose a package of internet services in line with his specific needs.

At the same time, we see that Information Technology (IT) often plays an important role in the realization and provision of a service. Note for example that IT is intrinsic to the aforementioned internet services. If IT services strongly depend on IT, then it should also be possible to support the selling of these services by IT. One can for example imagine a scenario whereby the customer online creates his own package of services, that upon selection can be provisioned immediately.

In this thesis, we introduce a methodology that allows us to reason, in a (semi-) automated manner, about the composition of bundles of commercial services that satisfy specific customer needs: $e^3service$. Multi-disciplinarity is key to the theory development of $e^3service$: We use theories from service marketing to enable analysis of commercial services and customer needs, and combine these with theories from computer science to formalize the analyses to the extent that the computer can do something with it.

For the automated composition of service bundles two issues are of particular importance: (1) *on a conceptual level*: What concepts play an important role in composing a service bundle, and (2) *on a reasoning level*: How do we employ these concepts, in such a way that we can reason about service bundling?

On a conceptual level, $e^3service$ contributes a formal theory of 'consequences'. Consequence allow $e^3service$ to:

1. express services in terms of the value they provide. This is important because a customer is often not interested in a service itself (e.g. 'e-mail hosting' from the provider KPN) but in the valuable outcomes that this service provide (e.g. an 'sending and receiving e-mail');

2. gradually specify a complex customer need into something that is specific enough to find a service for. Consequences exist on several levels of abstraction, as a result of which a high-level need from the customer (e.g. 'to communicate with someone abroad') can be translated into something that is specific enough to find services for(e.g. an e-mailing service);

3. bundle services that naturally 'fit' together. This is important because the need for one service (for example: 'e-mailing') often implies the need for another service (for example: 'spam filtering')

It is important to note that our theory of consequences is multi-disciplinary. The consequence-concept originates from marketing, and is useful for analyzing customer needs. However, *on its own terms*, a consequence is not formal enough to be 'understood' by a computer. We therefore capture the concept of a consequence in an ontology: A formal conceptual model, used in computer sciences, that allows the computer to eventually do something with consequences.

On a reasoning level, e^3*service* contributes an *interactive* dialogue that allows a customer to *incrementally* create a service bundle in a *semi-automated* manner:

- The creation of a bundle requires an *interactive* dialogue between customer and provider. The customer weighs the positive consequences (for example: an e-mail hosting service) and negative consequences (for example: A price to be paid) of a bundle to eventually decide upon what specific services to select;

- A bundle is created *incrementally*, by presenting consequences to the customer that fit well with the consequences of the services that the customer has already selected. For example: If a customer has already selected a service with the consequence 'e-mailing', it would be logical to also offer him a service with the consequence 'spam filtering'.

- The dialogue is carried out in a *semi-automated* manner. The dialogue is automated in the sense that services are found automatically for a specified customer need. Yet, the dialogue does require human intervention on the part of the customer.

So: we provide a *structure* for the customer dialogue for finding service bundles. This is an addition to existing service bundling approaches, that reason about

service bundling by simply projecting a set of fully specified customer needs to service bundles.

To assess its practical validity, *e*^{3}*service* has been applied to four real-world, non-trivial, case studies in the domains of respectively the telecom industry, the postal industry, the healthcare industry, and in the domain of hosted IT services. In addition, *e*^{3}*service* has been implemented in a software tool to showcase that we can indeed reason about bundling commercial services semi-automatically.