Cost Control in Service Composition Environments

Jörg Niemöller, Raphaël Quinet, Roman Levenshteyn and Ioannis Fikouras
The authors are with Ericsson Corporate Research, Service Networks & Applied Technologies, Ericsson Allee 1, D-53134 Herzogenrath, Germany.
joerg.niemoeller@ericsson.com, raphael.quinet@ericsson.com, roman.levenshteyn@ericsson.com, ioannis.fikouras@ericsson.com

Abstract

This paper investigates requirements and proposes solutions for charging of composite services. An analysis of the relations between the service user, service broker and service provider with respect to charging suggests that the price a service broker asks from the user of a composite service is in general decoupled from the price the service broker pays to the service providers of the component services used to create the composite service. Cost control mechanisms as part of the service selection are therefore essential for the service broker. This paper proposes a cost control mechanism based on instantaneous service price prediction.

1. Introduction

The introduction of a new technology in telecommunication networks opens new business opportunities and allows new business models. In this context, the main purpose of charging functionality is the implementation and control of the business model in the network. The charging infrastructure is therefore the technical enabler of business models. With the introduction of a new technology and related modifications in the business model, the charging solution needs to be reviewed in order to identify needs for adaptation.

Service composition technology allows the generation of rich, new services by combining existing basic services with a model-driven approach. A key property of service composition is the data driven selection of constituent services, which leads to very flexible and dynamic service offers. In such a service environment, new kinds of business relationships are expected between the parties that are involved in a service offer. The operator of the service composition environment and its business model is in the focus of this paper. Of special interest is his relation to the users of a composite service on the one side and 3rd party service providers contributing to the composite service on the other side. This paper investigates restrictions and problems that can be expected in order to apply the business model of the composite service provider.

This paper provides a brief overview over the service composition approach that was chosen by Ericsson Research in order to implement service oriented architecture (SOA) in telecommunication networks. Based on this approach the role of the operator of the service composition infrastructure is analyzed with regards to charging for the used services. A solution is presented that allows this operator to control the costs for composite services, while services can be dynamically selected and included. Finally an example further outlines the presented solution.

Business frameworks on top of the service infrastructure are for example proposed by IPSphere [6] and TMForum [7]. Their approach also considers service costs by providing a unified way to reach an agreement between operators and service providers. The cost control mechanism presented in this paper considers service cost directly at instantiation of the components of a composite service. This allows the service selection to be further cost optimized even though it is based on late binding of loosely coupled services.

2. State of the Art

2.1. Composite Services

This paper is based on the model-driven approach to service composition as described in [1]. This composition technology allows dynamic generation of complex composite services by combining basic constituent services. With the presented approach, a model of the composite service is defined by describing the required properties of constituent
services. The concrete services that match the requirements are selected during execution.

Application skeletons implement a composite service by defining the set of participating services, the structure of the composition, i.e. how individual services should be connected in order to form a composite service, and the control flow, i.e. in which order the individual services will be executed.

In order to specify a service to be used as a component of composite services, the application skeleton defines a service template. This service template is a placeholder for a constituent service and describes the service to be used by means of abstract properties rather than by pointing to a concrete service deployment. Therefore these properties constitute requirements on the service to be selected at run-time. They are therefore referred to as constraints of the service selection. For example, the following constraint will select a service that provides the position of a mobile subscriber. It also specifies that only positioning services with a minimum accuracy of 100m shall be considered.

\((\text{function} = \text{'positioning'}) \& (\text{min}\_\text{accuracy} = '0.1\ \text{km}')\)

An application skeleton is not just a chain of service templates defining the set and order of constituent service invocations. In addition, conditional branches in the skeleton flow can be defined, resulting in alternative behavior of the composite service. The related branching conditions can for example evaluate the results of previously executed services.

Due to the abstract descriptions of the constituent services to be used within a service template, the application skeleton constitutes an abstract model of the composite service. Concrete services are selected according to the constraints at execution time rather than at design time. This characteristic is referred to as late-binding. The advantage of this approach is that a composite service can be designed without knowing the exact set of available constituent services. Furthermore, constituent services can be replaced by any compatible service without changing the composite service.

### 2.2. Dynamic Service Selection

The execution environment for composite services consists of the Composition Engine (CE), a Service Database and the Application Servers that provide constituent services. In an IMS context as shown in Figure 1, the composite service is triggered from the Internet Multi-media Subsystem (IMS) using the Session Initiation Protocol (SIP) signaling. The composition engine selects the needed skeleton according to the received trigger and starts interpreting the sequence of skeleton elements. One of the central tasks in the execution of a composite service is finding and invoking concrete services that match the service templates.

The service database has the central role in the service selection process. For each available service, the service database contains a description that combines abstract properties with information about the service location and API. The service database therefore links the model of a service expressed by its abstract properties with a concrete service available on one of the application servers.

The task of the dynamic service selection of composite services is described in Figure 2. The user invokes a composite service which triggers the execution of a composition skeleton in the composition engine. This includes the evaluation of service templates by issuing a service database query in order to find the services that match the selection constraints. The service database provides a set of services that all fulfill the requirements and can be used. Finally, one of these services is selected and invoked. In an IMS context, the service composition engine is an implementation of a Service Capabilities Interaction
Manager (SCIM) as briefly introduced by 3GPP in [5].

The service composition technology described here is designed to operate within a heterogeneous service environment consisting of services from multiple service technologies (e.g. SIP, SOAP Web Services, J2EE). This is achieved by deploying Composition Execution Agents (CEA) on the execution platform as shown in Figure 1. The CEAs are the only part of the presented service composition approach that is related to the technology of the constituent service (e.g. SOAP WS or SIP). They enforce the decisions of the composition engine and link the local service sessions to composition sessions.

2.3. Service Pricing and Billing

The charging and billing solution used in recent telecommunication networks usually follows the architecture and principles as described by 3GPP in [3] and related documents of the 32.xxx series of technical specifications. In a charging environment according to [3], the information about service prices and tariffs is centralized within a charging and billing system. The service providing nodes contain functions that monitor user activities and collect information about service usage. This information is reported to the central charging system where a price is determined and charged to the user. The service nodes have normally no information about the final price of a user activity. Figure 1 shows the allocation of a central charging and billing system in a service composition environment. Here, the IMS network nodes and the application servers provide data about service usage to the central charging system. The prices for constituent services are subject to commercial agreements between the service provider and the operator of a service composition environment.

3. Problems in Composition Charging

In a typical service usage scenario for a composite service there are three distinct roles:

1) A service user, who requests a service from the service broker.
2) A service broker, who applies service composition techniques in order to implement the requested service. This includes selection and invocation of services provided by service providers.
3) A service provider, who provides one or several constituent services.

From the user’s point of view, the service broker is a service provider. The composite nature of a service that is offered by the service broker and the implicitly involved providers of constituent services are hidden from the user. Therefore, the user is not aware that the service he has requested is implemented by selection and aggregation of multiple separate services with the possibility that many parties are involved. This situation is shown in Figure 3. The composite service is requested by the user and provided by the service broker. In order to assemble the requested function, the services A and B are included dynamically. Thus A and B, and respectively the providers 1 and 2, provide the requested service to the user.

From the user’s point of view, the implicit partitioning and distribution of the composite service functionality is transparent. He has a contractual relation only with the service broker. This becomes particularly relevant when considering the charging for service usage as shown in Figure 4. The user only pays for the service he has requested. This is the composite service. The service broker in turn has to pay the providers of the included constituent services. In this model there is no direct charging relationship between the user and the third party service providers.

On top of that, the users of telecommunication...
services must be able to know in advance what to expect to pay for a service. This is a legal requirement in many markets. The service broker, in its role as service provider for the end user, has therefore the responsibility to set and communicate the pricing schemes for the composite services it offers.

When building a composite service, the included constituent services are selected from a pool of functionally compatible services. Although they provide similar functionality, suitable constituent services may differ e.g. in price. The total price of providing a composite service that was assembled based only on functional constraints can therefore vary significantly. As the pricing scheme that was communicated by the service broker on the user side usually has to be stable, it should be decoupled from the prices paid by the service broker to the service providers. Hiding the composite nature of a service means in this respect, hiding the actual costs for constituent services usage from the user. In order to reach positive total revenue, the main problem for a service broker is to ensure that the overall income received from users of a composite service is in total greater than the costs paid to third parties. While occasional losses may be acceptable, the overall balance must be positive. A mechanism that allows the service broker to control its costs in a dynamic service selection environment is therefore required.

4. Cost control by service price prediction

4.1. Service Price Based Constraints

The previous chapter identified the need for cost-control mechanisms for the service broker in order to limit spending on constituent services. One approach would be to negotiate the price for services with each service provider in order to reach a general price level that would allow using the services without further price consideration. This approach can only be successful if flat or very low constituent service prices are reached compared to the price paid by the user for the composite service. In practice, this can only be achieved for standard services, while the main service within a composite service is often a premium service like for example multimedia streaming. On top of this, a certain price pressure on the composite service offer can be expected, which keeps the revenue generating gap between the composite service income and constituent service spending relatively small.

In general, several service providers may be available for a certain type of service. Furthermore many factors influence what provider offers the most adequate service of a certain type. These influencing factors are for example date and time of service usage or user specific properties like the location of a user. The factors that influence the price can be dynamic, which inhibits general and predetermined conclusions about which service is cheaper.

In such a dynamic environment, minimizing the spending side can be achieved by selecting a service according to the price that is valid at the time the service is needed. This implies that the price needs to be considered during the service selection. Hence, price-based constraints can be used to apply price-based conditions, such as for example specifying a maximum approved price for the service to be selected. The following constraint specifies to select a positioning service that must not cost more than 0.1 Euros.

(function='positioning') & (pricemax='0.1 EUR')

Another possibility would be a constraint that requires the selection of the cheapest of all services that remain after the elimination through other constraints:

(function='voice') & (price is min)

Using price-based constraints demands that a service price is actually available at the moment of service selection and therefore prior to execution. This condition is fulfilled for services that are charged flat or per invocation. On the other hand, there are many services for which the charged price depends on usage details like for example the duration of the service usage or the amount of data transferred. For such services, the total price cannot be known in advance at service selection. Only tariffs expressing for example a price per duration or a price per data volume might be available.

4.2. Service Usage Profiles

In order to enable a comprehensive price-based service selection, this paper proposes to predict service prices based on a service usage prediction. The service usage details are the essential unknown variables needed for a price calculation. From an estimate on the service usage, an estimated service price can be obtained.

A service usage estimate basically expresses user habits and is therefore an individual property of the user or user groups. Service usage profiles are introduced in order to formally describe and store service usage habits. Each service usage profile contains the prediction for a certain type of service. For example, the users of one user group mainly do short voice calls of three minutes, while the users of a
different user group are used to talk for more than ten minutes and therefore considerably longer. In order to reach an accurate service price prediction in this example, two different usage profiles are created for the same service of type 'voice' and assigned to the users of either user group. If the expected price of a voice call is needed, the user specific expected usage is taken for the price calculation.

Usage profiles can be obtained for example from the observation and analysis of charging data with statistical methods that are not covered in detail in this paper.

4.3. Allocation of Service Price Definition

Besides the information about the service usage, a mechanism that calculates a price needs a definition of the charging model of this service. As the price can be considered to be a property of the service, it is reasonable to assign the price-defining data to the service description in the service database. This way, a price tag is attached to the service.

The attachment of a price to a service description can be direct by actually storing all data needed for price calculation in the service database itself, or it can be indirect by pointing to an external resource or system where the pricing data can be found or from which the price calculation can be ordered.

The price tag can for example contain the formula to be taken for price calculation. Price influencing parameters would be reflected in the variables of the formula. A rate would be a simple formula with only one variable (e.g., the duration or the data volume used). If no variables are used, the price tag would directly state a fixed price for service usage.

If many variable parameters need to be taken into account in order to reach a sufficiently accurate result, the formula could become complex. In this case it might be more practical to use a structured representation of the price definition, e.g., a script with calculation instructions or an analysis tree consisting of nested tables. Formulas could still be an element of these price definitions combined with additional calculation logic.

With the degree of complexity in the price definition, the accuracy of the estimated price can increase. For easier handling, the complex price definition data might be allocated in a separate database. In this case, the price tag at the service description would contain a link to the actual definition data.

In a charging infrastructure as described in chapter 2.3, the central charging system is the sole reference for price determination. It contains a comprehensive definition of charging models and it is able to consider all user and service related parameters and price influencing factors. This includes the consideration of temporary special conditions, like for example promotions for the user. Considering such dynamic information exceeds the basic definition of charging models. It requires integration into the environment of administrative nodes and user specific databases that are part of the business management infrastructure. The charging system is integrated in such an environment. Using it for price calculations as part of a price estimation process would therefore provide the most accurate results without deviations due to incomplete or not considered data. The accuracy of the estimated price would fully depend on the accuracy of the usage prediction.

4.4. Price Aware Service Selection

With the usage profiles introduced in chapter 4.2 and the definition of service charging as described in chapter 4.3, an environment for price-based service selection is available.

The constraint-based selection of services usually translates constraints into database queries. For price based constraints, the price information needs to be collected or calculated before the constraint condition can be evaluated. If a price based constraint is used in the selection of a service, the selection routine collects or calculates the prices of all services that are still possible matching candidates after the evaluation of other constraints. For each of the remaining services, the expected price is determined separately. Then the condition expressed by the constraint is applied. This might result in eliminating services from the list of candidates (e.g. if they are too expensive). It is also possible to sort this list of candidates in order to prefer the service that best matches the condition (e.g., if the cheapest shall be selected).

As price prediction might need considerable extra processing, it should be placed late in the selection process, in order to minimize the load.

In order to determine the price of a service, the entry of this service in the service database and its price tag plays the central role. It primarily contains all information needed to determine a price. The possible methods were described in chapter 4.3. The entry could contain the price, a rate, a formula, a link to an external storage of charging data or a link to a charging system also specifying the protocol to be used. A service selection routine should be able to handle different price determination methods.

If needed, this price-aware selection process also utilizes user data and usage profiles obtained from their
respective database. Figure 5 shows the additional steps in the service selection and the service database as central broker of service price information.

4.5. Available Standardization

The Diameter Credit Control (DCC) Application as specified by the Internet Engineering Task Force (IETF) in [2] describes an interface for price inquiry towards an online charging system. After sending the details about the user and the used service, the charging system would answer either with the total price for the service or with a rate.

Most charging systems that are applied in telecommunication networks follow the specifications from 3GPP and do not support price inquiry of the DCC application. The interface for price inquiry is not described until 3GPP release 8 of the specification of Diameter Charging Applications in [4]. Adding this interface also to the standardization according to 3GPP is recommended in order to ensure a unique environment for the presented cost control solution.

5. Example

The example composite service “FamilyChat” is implemented by the service skeleton shown in Figure 6. It allows the user to chat with a member of his family that is located nearby. The family members of each subscriber are defined in a database. In order to reach the described function, the “FamilyChat” service determines the caller’s own position and the positions of all family members by multiple invocations of positioning services. The distances are compared and finally a chat session is opened with the family member closest to the caller’s position. The service “GetNearestContact” included in Figure 6 is also a composite service that is called implicitly in order to evaluate the list of family members. This includes several calls of positioning services.

Table I shows the prices for the composite service “FamilyChat” and the available constituent services. The price of the composite service defines the income of the service broker. In order to avoid losses due to the usage of expensive constituent services, the service designer has added price based constraints to the service templates of the positioning and chat services.

In this example, the usage profile of the user defines for the chat type services an expected usage of one hour and 100 messages. The user has registered four members of his family. This implies that the composite service implicitly triggers a positioning service five times in order to determine the position of the user plus the position of each family member. The user actually uses the chat session for two hours and 200 chat messages are sent. According to Table I, the user is charged 2.50 EUR for using “FamilyChat” as described.

Also, without cost-aware service selection, both positioning services and both chat services could be used. The resulting costs can vary between 1.20 EUR (0.00 EUR for “PosB” used at night plus 1.20 EUR for “ChatD” used for two hours with 200 messages) and 3.00 EUR (1 EUR for PosB used five times plus 2.00 EUR for “ChatC” with 200 messages). In this case...
costs for constituent services needed in order to build the composite service can exceed the income of the service broker.

At execution of the service template “GetCallerPosition”, the search in the service database for services that match the constraint-defined property “positioning” provides “PosA” and “PosB” as potential candidates. The price tag for “PosA” directly provides the costs for using this service (0.05 EUR). For “PosB” the price tag contains a script considering the time of invocation. Assuming 10:00 in the morning leads also to the price of “PosB” (0.20 EUR). The first price based constraint sets an upper limit for the approved price and therefore eliminates “PosB” from the list of candidates. “PosA” satisfies all constraints and is therefore selected and invoked. This selection process is repeated for subsequent positioning of all family members.

In the selection of the chat service in the service template “StartChat”, the price-based constraint requires to choose the cheapest services. The price tag within the service database for both services refers to the charging system. Therefore, for each of the chat services a price inquiry is sent to the charging system utilizing usage profiles. User-specific special conditions like promotions are considered. Without finding a promotion for the user, a price of 1.00 EUR is predicted for “ChatC” and 0.6 EUR for “ChatD” leading to a selection and invocation of “ChatD”. If a promotion would be valid for the user with “ChatC” the situation changes and “ChatC” would become cheaper and is selected.

6. Conclusion and future work

Predicted prices may not be most accurate, but as long as they allow a relative comparison of services and usually lead to the selection of the one that actually is cheaper, they are helpful in cost control. Future work could investigate the algorithms that generate accurate usage profiles and assign them to users and user groups.

Besides the service price, other dynamic or soft selection parameters like for example service quality could be considered at service selection. In order to find an overall best service with sufficient quality at a reasonable price, a weighted selection parameter could be calculated and used.

IMS can be expected to become a service ecosystem for telecommunication network operators. Service composition technology will be part of this infrastructure and the described cost control problem becomes imminent. In order to allow comprehensive solutions, 3GPP should introduce an interface for price calculations towards the charging system.

7. References


