# A Semantic and Information Retrieval Based Approach to Service Contract Selection*

Silvia Calegari, Marco Comerio, Andrea Maurino, Emanuele Panzeri, and Gabriella Pasi

University of Milano-Bicocca,
Department of Informatics, Systems and Communication (DISCo), Milano, Italy
{calegari,comerio,maurino,panzeri,pasi}@disco.unimib.it

**Abstract.** Service contracts represent the agreement between the service provider and potential service consumers to use a specific service under given conditions; for each service multiple service contracts are available. In this paper we investigate a new approach to support the service contract selection by exploiting preferences both explicitly defined by a user and implicitly inferred from his/her context. The core of our approach is the use of multi-constraint queries expressed on punctual values and on textual descriptions. Both semantic-based and information retrieval (IR) techniques are applied. Experimental evaluations show the effectiveness of the proposed approach.

## 1 Introduction

The visionary idea of Service-Oriented Computing (SOC) is a service ecosystem where application components are assembled with little effort into a loosely-coupled network of services to create agile applications that might span organizations and computing platforms [1]. One of the building block of SOC is *service discovery* that is the activity of locating a machine-processable description of a service that meets certain functional requirements [2].

Since more than one service is likely to fulfill the functional requirements, some ranking mechanisms are needed in order to provide support for the (semi) automatic selection of a restricted number of services (usually one) among the discovered ones. Broadly speaking we can identify two phases in the service discovery activity: the first one is devoted to identify services that satisfy the functional requirements, while the second one (also called *service selection*) is in charge of ranking retrieved services according to non-functional properties (NFPs) that represent the description of the service characteristics (e.g., availability, performance, price) that are not directly related to the provided functionality. As in the real world also in SOC ecosystem, NFPs can be enclosed in *service contracts* representing the agreement between the service provider and potential service consumers. In the last years, increasing research efforts are

* This work is partially supported by the Italian MoSeForAgroFood project (funded by the Lombardy region), and by the SAS Institute srl (Grant Carlo Grandi).

G. Kappel, Z. Maamar, H.R. Motahari-Nezhad (Eds.): ICSOC 2011, LNCS 7084, pp. 389–403, 2011.

aimed at defining solutions for service contract management [3]. For each service, multiple service contracts are available and each service contract can be offered to specific user categories under predefined applicability conditions. More specifically, constraints based on NFPs consist in the specification of *contractual terms*. They can be expressed by numeric values defined in different units (e.g., price in Euro or in USD), or by qualitative values (e.g., trust is *high*, software is *open source*). The *service contract selection* is the activity of ranking service contracts according to the constraints on NFP explicitly specified by the user, and/or implicitly inferred from user information.

Service contract selection is definitively one of the most important enabling factors for supporting flexible and dynamic business processes and agile applications. Nevertheless, conversely to the real world where contract selection is a largely studied problem [4,5], this activity has not yet been extensively investigated, and current approaches [6,7,8,9] lack, among others, in providing support for the formulation of user requests, in the evaluation of applicability conditions and in managing the heterogeneity of NFPs that can be specified in a service contract. The management of NFPs is a complex task since there exists no standard terminology for describing these properties. This means that service providers and consumers specify their service contracts as they wish, thus raising the term ambiguity problem when multiple services governed by different contracts are utilized. In fact, similar properties may have different names (e.g., in different languages or domains) or the same name may refer to different properties (e.g., in different domains a property may have different implications). This current lack of agreed terminology, combined with a major absence of trust in claims about service contracts, renders service contract selection difficult if not impossible in commercial organizations.

In this paper we present a new approach to service contracts selection based on the exploitation of preferences explicitly defined by a user and implicitly inferred from his/her context and the use of both semantic-based and information retrieval (IR) techniques to rank service contracts. In particular, the main contributions of our approach are:

– *multi-constraint query formulation*: the constraints on NFP composing the user query are defined by considering preferences explicitly specified by the user, and implicitly inferred from user information (e.g., personal information specified at registration-time, historical information related to formerly service used). The constraints can be either expressed as data constraints or keyword-based constraints and they can be defined on a wide set of NFPs.
– *hybrid approach to service contract ranking*: the ranking of service contracts is based on the combination of semantic and information retrieval techniques to evaluate the degree of matching between contractual terms and user preferences.

The rest of the paper is organized as follows: Section 2 presents the state of the art of service contract selection and related fields. Section 3 describes the proposed approach. In Section 4, and 5 our hybrid service contract selection approach is described. Sections 6 and 7 present an exhaustive example and the experimental evaluations of our approach. Conclusions and future works conclude the paper in Section 8.

## 2 State of the Art

The agreement between a service provider and a service consumer can be established by using different approaches (e.g., policies [10] and service level agreements [11]). Even if some differences exist among these approaches, the common term *service contract* is generally used [3].

Currently, service contract selection is executed by either non-semantic or semantic approaches. Non-semantic approaches (e.g.,[6,7]) are characterized by a high efficiency but low precision due to the management of only syntactic service contract descriptions. The evaluation of degrees of matching between requested and offered contractual terms related to qualitative NFPs is reduced to the syntactic comparison among values, raising semantic misunderstandings and inefficient selections. Semantic approaches (e.g., [8,9]) are based on automated reasoning techniques on service contract descriptions. These techniques are particularly suitable to mediate different terminologies and data models. Therefore, reasoning techniques can be used for the evaluation of degrees of matching of contractual terms related to qualitative NFPs in order to exploit semantic relations between NFP values. However, the evaluation based on logical reasoning is characterized by a low efficiency since many reasoners show poor effectiveness when dealing with non trivial numeric functions (e.g., weighted sums) which are needed to manage more properties at the same time.

The most important problems in both semantic or non-semantic approaches above mentioned are: (i) *expressivity* as the possibility to evaluate qualitative descriptions by means of logical expressions on ontology values, and quantitative descriptions by mean of expressions including ranges and inequalities; (ii) *extensibility* as the possibility to define parametric degree of matching evaluation by customizing evaluation functions and (iii) *flexibility* as the possibility to perform evaluation in case of incomplete specifications.

For example, the approaches in [6,7,8] present some limitations in expressivity and extensibility. The NFP-based service selection approach proposed in [6] considers the evaluation of qualitative properties but it does not consider the semantic relations among property values. The framework described in [7] allows the definition of requested contractual terms only using the *equal* operator and the selection process is simplified and modelled as the problem to maximize the difference between prices (associated with each contract using a pricing function) and score (associated with each requested contract and stating the maximum price for which the costumer is willing to carry out the trade). Finally, the semantic approach in [8] is applicable only for properties characterized by ordered scale values and fixed tendencies (e.g., the cost must always be minimized) limiting the freedom of the user in defining his/her preferences.

The approach to Web service selection based on the usage of axioms for requested and offered contractual terms defined in [9] lacks in flexibility. The exploitation of axioms supports complex and conditioned term definition (e.g., if the client is older than 60 or younger than 10 years old the invocation price is lower than 10 euro) but forces the user to specify all the necessary information (e.g., age) in advance.

In [12], an hybrid approach to Web service contract selection that combines logic-based and algorithmic techniques and offers high levels of expressivity, extensibility and flexibility is proposed and tested. The limitation of the approach is that applicability conditions on service contracts are not evaluated and the approach lacks in providing support for the formulation of user requests. In this paper, we extend the approach in [12] by means of IR techniques.

## 3 The Proposed Approach

The aim of the whole service contract selection process is to propose to the user a list of service contracts ranked according to his/her preferences. The process is composed of set-up time and run time activities and it is based on the software architecture shown in Figure 1. At set-up time, the user interacts with the registration module in order to create his/her *user profile*. At run time the user specifies preferences on functional and non-functional properties in order to perform the service discovery and the service contract selection. At set-up time, during the registration, the user selects from a list one of the pre-defined profiles expressed in natural language. The pre-defined profiles help the user to provide relevant information on generic characteristics (e.g., spoken languages, used devices). Then, the user completes the registration by inserting personal information such as (i) his/her personal data, (ii) his/her agenda in order to know at which time the user is located in a particular location, and (iii) preferences

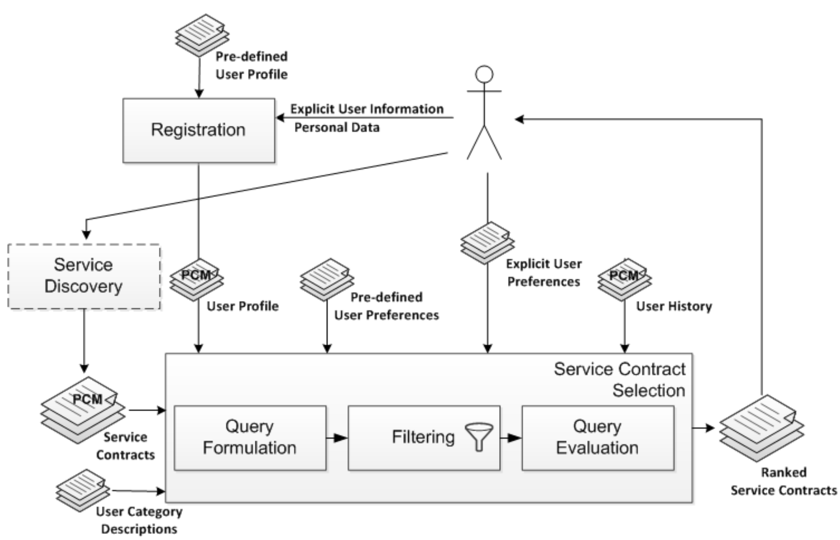


**Fig. 1.** The proposed approach to service contract selection

on specific properties such as the preferred payment method. Preferences are specified by means of textual descriptions. The user information gathered in this phase (i.e., personal information, selected profiles and textual descriptions) are jointly considered to define the *user profile*. It is worth noting that our approach to build the user profile is not tailored to any specific context model, and several context models can be applied to our approach.

Once the phase of information gathering is completed, the user can access to the next phase. At run time, when the user looks for a service offering a specific functionality, the service discovery component (not discussed in this paper) is invoked. Each discovered service is associated with different service contracts representing different NFPs, and applicable to different user categories. In order to support the user in choosing the contract that best complies to his/her preferences, the service contract selection module is invoked. This module is composed of three components that support:

- *Query Formulation*: the user selects pre-defined preferences (e.g., *I want to receive information on my mobile phone*) from a list, and he/she personalizes them by writing a text into a textual area, like the following *"I want a blanket insurance on the service delivery"*. User preferences, user profile and information extracted from the user history (i.e., information on past interactions between the user and services) represent the *contextual user information* that are used to formulate the multi-constraint query.
- *Filtering*: service contracts are filtered complying to the user category and contextual user information. The result is a set of *filtered service contracts*.
- *Query Evaluation*: the multi-constraint query is evaluated against the filtered service contracts. A ranked list of service contracts is returned to the user.

In our approach, we adopt the Policy Centered Meta-model (PCM)[1] as a meta-model for service contracts and contextual user information specifications. As shown in [13], the PCM outperforms other models by supporting: (i) expressive descriptions addressing qualitative contractual terms by means of logical expressions, quantitative terms by means of expressions including ranges and inequalities and, (ii) structured descriptions that aggregate different term descriptions into a single entity with an applicability condition.

The Query Formulation phase will be detailed in Section 4, whereas Filtering and Query Evaluation will be described in Section 5.

## 4 Multi-constraint Query Formulation

The simplest query formulation allows the user to select a query from a predefined list. This list is made up of the most frequent user queries, and each of them is formally defined by the service provider in the PCM format in order to easily represent the query constraints. But each predefined query is presented to

[1] The PCM formalizations in OWL and WSML-Flight are available at: http://www.siti.disco.unimib.it/research/ontologies/

the user as a textual description to ease the selection process. A query is formulated by means of constraints on data values: we allow the specification of both *precise* and *flexible* constraints. Precise constraints are specified on a selected attribute by a specific value of the attribute domain, e.g. *insurance=damage*. Flexible constraints can be specified on attributes with a numeric domain by a linguistic label which constraints the values of the attribute domain, e.g. *price = at most 40 €*. Formally, such a linguistic label is associated with the membership function of a fuzzy subset of the domain. Additional details related to the definition of query constraints are presented in Section 5.

In Listing 1, a PCM-formulation of the pre-defined query *"I'm looking for a CHEAP (i.e.,* price at most 40€*) delivery service by having an INSURANCE on damage"* is shown. In the above example, both a flexible (*price=at most 40€*), and a precise constraint (*insurance=damage*) are specified.

**Listing 1.** An example of user query in PCM format

```
nonFunctionalProperties
   dc#description hasValue "Request Instance for Logistic Operator"
endNonFunctionalProperties

pcm#hasNfp hasValue requestedPrice
pcm#hasNfp hasValue requestedInsurance

instance requestedPrice memberOf nfpo#PriceRequest
pcm#hasExpression hasValue requestedPriceExpression

instance requestedPriceExpression memberOf nfpo#PriceExpression
pcm#hasOperator hasValue pcm#atMost
pcm#hasParameters hasValue 40
pcm#hasUnit hasValue nfpo#euro

instance requestedInsurance memberOf nfpo#InsuranceRequest
pcm#hasExpression hasValue requestedInsuranceExpression

instance requestedInsuranceExpression memberOf nfpo#InsuranceExpression
pcm#hasOperator hasValue pcm#exist
pcm#hasParameters hasValue ins#damage
```

At this point, a user can personalize the selected query in three ways: (1) by modifying the pre-defined constraints, (2) by adding further constraints, and/or (3) by adding a short textual description. This way the user can provide more details and/or refine the constraints about the required service contracts. As an example, the value assigned to the precise constraint *insurance=damage* can be replaced by *insurance=fire&theft*.

Once the user has completed the formulation of his/her query, some additional constraints are automatically added based on the information obtained from the personal context, where constraints on both the user information and the user history are examined. The analysis of the user information (stored at registration time) determines additional precise constraints like the list of information channels that can be used to deliver information to the user. Instead, from the user history implicit user preferences are extracted, such as how many times a specific service contracts has been employed by the user in the past (for an example see Section 6).

# 5 Filtering and Query Evaluation

As explained in Section 4, the *PCM-based multi-constraint query* is composed of two types of constraints: constraints on punctual values (data), and constraints on textual descriptions. In the following subsections, we will describe how the query is evaluated for filtering and ranking the service contracts.

## 5.1 Service Contract Filtering

The first step executed by the query evaluation process is the service contract filtering; such filtering is based on the user category affiliation and aimed to filter out from the set of service contracts the ones that do not relate to the current user. This is done by matching the user category to the contract applicability and then by removing service contracts that require categories that the current user does not belong to. For each service contract a category is defined by a set of applicability conditions (e.g.: user age, VAT owner) that a user must have. A user is associated with a category if and only if all the conditions are respected. As an example, if the category called *SeniorUser* has the applicability condition *"User must be at least 65 years old"*, and the current user is 35 years old, then service contracts related to this category will be filtered out.

## 5.2 Constraints Evaluation

The NFP expressed in a service contract are defined by both specific data (such as prices, insurance, ...) and textual descriptions. With each query constraint a constraint evaluation function (in short CF) is associated. The evaluation of a constraint produces a matching degree, in the interval $[0, 1]$, between the constraint itself and a service contractual term. In the following sub-sections, the evaluation functions are described in relation to the different types of constraints.

***Flexible Constraints on numeric data values.*** As explained in Section 4, we allow the specification of flexible constraints on data values. The evaluation of these constraints, formally defined as fuzzy subsets of the considered attribute domains, is performed by means of membership functions that express the compatibility between the flexible constraints and the related attribute domains. We define a membership function as a parametric linear function the value of which is in the interval $[0, 1]$. An example of membership function for the *price=at most 40€* constraint is depicted in Figure 2: service contracts with prices lower (or equal) than the required one (e.g., 40€) will have a matching degree of 1, whereas service contracts with prices higher than 60€will have a matching degree of 0. The flexibility of the adopted solution is for the range of values between (40€,60€) where the matching degree will decrease as the price will continue to increase.

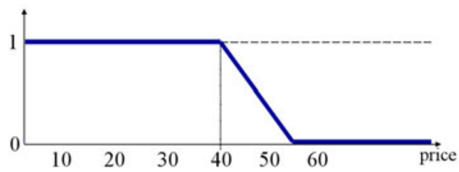


**Fig. 2.** Constraint evaluation function for *"at most 40"* constraint

***Concept-based constraint evaluation.*** Service contracts could include NFPs that use concepts to represent their values. For example, the *Insurance* NFP assumes values (e.g., blanket, fire&theft) characterized by relations among them (e.g., a blanket insurance includes a fire&theft insurance). The evaluation of concept-based NFP constraints makes use of an ontology/thesaurus that maps all the possible values with all the relations among them. The matching degree is evaluated by the distance from the required value and the one provided by the service contract. Given the taxonomic hierarchy, the matching degree is maximum if the value required by the user is the same or a descendant of the one provided by the service contract. As opposite, if the required value is an ancestor of the provided one, the resulting matching degree is calculated according to the distance between the two values and it is normalized in the $[0, 1]$ set. Examples of concept-based constraint evaluation are in [12].

***Set-based constraint evaluation.*** A third type of values that can be associated with NFP is called *Set-based*. Set-based constraints are defined by a set of values that have to be matched against the offered set based NFPs. As an example, the matching degree for the *InformationChannel* constraint is computed by applying the following formula:

$$CF_{InfoChannel}(sc, q) = \frac{|q_{InfoChannel} \cap sc_{InfoChannel}|}{|q_{InfoChannel}|}, \tag{1}$$

where $sc$ is a service contract, $q$ is the user query, $q_{InfoChannel}$ is the set of the Information channels specified in the user query, and $sc_{InfoChannel}$ is the set of the Information Channels provided by the service contract. The matching degree for the Information Channel will be in the set $[0, 1]$, where 1 represents a fully satisfied constraint and 0 will be returned for service contracts that do not provide any of the required characteristics in the user PCM-based query.

***Keyword-based Evaluation.*** For the service contract textual description evaluation an IR approach is adopted to compute the user based query and the service contract matching; the relevance is estimated by a matching degree (in the set [0,1]) between the user needs and the textual description. The service contract description is a plain text that describes the service functionalities and characteristics in natural language. To index the textual description, simple IR techniques are applied, such as keyword extraction (words and terms are identified and extracted from the service contract textual description), and stop words removal (the terms that are non-significant or do not provide any meaning are removed), respectively. For sake of simplicity we do not give a formal description of each of the above IR techniques, for more information and further details we recommend the reader to refer to the IR literature [14]. A user query is specified as a set of keywords that represent the main features that the service contract should have.

The previously cited IR functionalities enable to estimate the *relevance* degree between the user query keywords and the keywords extracted from the service contract description. To this aim we use a classical IR model for relevance evaluation called the *Vector Space Model* that represents each set of terms (or keywords) as vectors and can evaluate the relevance degree by the *similarity* between two vectors using a vector distance such as the Cosine similarity. It is worth noting that in reference to the term ambiguity problem raised in Section 1, some term disambiguation techniques [15] could be applied either at the indexing phase or at the query formulation. We will address this issue in a future research.

### 5.3 Overall Degree of Matching

The proposed aggregation function, a linear combination where the previously described constraint evaluation functions are aggregated to compute the overall service contract score, is defined by the following formula:

$$DoM(sc,q) = \frac{[\sum_{i=1}^{nc} CF_i(sc,q)] + CosSim(\overrightarrow{sc},\overrightarrow{q})}{nc+1}, \tag{2}$$

where $nc$ is the number of constraints, $CF_i$ is the constraint evaluation function for the query constraint $i$ and $CosSim(\overrightarrow{sc},\overrightarrow{q})$ is the service contract textual description evaluation performed using the Cosine Similarity on $\overrightarrow{sc}$ (i.e., keyword vector related to the service contract), and $\overrightarrow{q}$ (i.e., the keyword vector of the query). In the Formula 2 the overall service contract degree of matching is calculated as the average of the query constraint evaluation scores.

## 6 An Exhaustive Example

The logistic operator is the domain chosen to provide a complete example of the approach described in this paper. In the logistic operator domain a service provider offers one or more facilities to potential users. For instance, a service provider can offer freight transportation of cumbersome goods and traceability information to the consumers through different channels (e.g. SMS, e-mail, phone call). A transportation service is characterized by a set of functionalities, and it is associated with one or more service contracts. Furthermore, a service contract contains one or more *contractual terms* and it is addressed to specific user categories. Examples of NFPs on which contractual terms can be defined are:

– *payment method*: how the user can perform the payment (e.g., credit card, electronic transfer, cash on delivery);
– *insurance*: the type of prevention applied to the service;
– *price*: the amount of money that must be paid for the transportation and the traceability service;
– *hours to delivery*: the number of hours required for the service fulfilment;
– *information channels*: the channels (e.g., SMS, e-mail, phone call) used to send traceability information to the user.

Table 1 shows a set of service contracts for two hypothetical providers defined on the basis of the above mentioned NFP list. For example, (i) *pay-flex* offers maximum flexibility with respect to payment methods; (ii) *high-trace* is characterized by maximum flexibility with respect to information channels and languages; (iii) *secure* offers a maximum insurance coverage; (iv),(v) *fast-plus* and *fast* support fast transportation, and (vi) *cheap* performs transportation at lower price. Each contract is characterized by advantageous/disadvantageous contractual terms (e.g., the *fast* service contract offers a fast delivery but at an higher price).

**Table 1.** Examples of service contracts traceable freight transportation services

| **Provider A** | | | | | | |
|---|---|---|---|---|---|---|
| **Contract** | **PayMeth** | **Insurance** | **Price** | **HToDel** | **InfoC** | **Vector** |
| *pay-flex* | credit card, elect.transf, cash | Fire&theft | 30 | 24-48 | SMS, e-mail | *traceability, cheap, english* |
| *high-trace* | credit card, elect.transf | Fire&theft | 35 | 48-72 | SMS, e-mail, call | *traceability, english, italian* |
| *secure* | credit card, elect.transf | Blanket | 35 | 48-72 | SMS | *secure, traceability, english* |
| **Provider B** | | | | | | |
| **Contract** | **PayMeth** | **Insurance** | **Price** | **HToDel** | **InfoC** | **Vector** |
| *fast-plus* | credit card | Fire&theft | 40 | 12-24 | SMS | *fast, traceability, english* |
| *fast* | credit card | Fire&theft | 40 | 24-36 | SMS | *fast, traceability, english* |
| *cheap* | credit card | - | 20 | 72-96 | SMS | *cheap* |

Each service provider specifies the user categories that can access each offered service contract; such user categories are usually hierarchical in the sense that a higher level category includes the facilities of a lower level category. Examples of user categories are sketched in Table 2. The affiliation to the *BusinessOne* category is addressed to users that are VAT owners and mobile phone owners, instead the *BusinessPlus* is dedicated to users who respect all the BusinessOne conditions, and who have also used service contracts offered by a specific provider for at least 30 times in the past. *SilverUser* and *BronzeUser* categories have memberships conditions defined exclusively on the number of historical service utilizations. Finally, the *SeniorUser* category presents a condition on membership based on the user's age. Notice that BusinessPlus/BusinessOne and SilverUser/BronzeUser are hierarchical categories (e.g. a BusinessPlus user is also a BusinessOne user, but not viceversa).

**Table 2.** Examples of user categories

| Contract | Category | Condition |
|---|---|---|
| *pay-flex* | *BusinessPlus* | VAT owner, mobile phone owner, 30 shipments |
| *high-trace,secure* | *BusinessOne* | VAT owner, mobile phone owner |
| *fast-plus* | *SilverUser* | 20 shipments |
| *fast* | *BronzeUser* | 10 shipments |
| *cheap* | *SeniorUser* | $\geq$ 65 years old |

Let us suppose that the customer "Mario Rossi" has interacted several times with our system by selecting the appropriate service contracts for his specific tasks. In particular, the user has used *high-trace* and *secure* contracts from Provider A for 5 times and *fast* contract from Provider B for 20 times. The identification of the categories for Mario Rossi with respect to the service contracts is performed. The information considered from the user profile are the user's age, as well as the VAT and mobile phone information. From the history, the information that he has used for 10 times a service from ProviderA reserved to Business One users, and for 20 times a service from ProviderB reserved to Bronze users are considered. Thus, by analyzing the above user context and these conditions, the selected categories for Mario Rossi are: *BusinessOne, SilverUser,* and *BronzeUser*, respectively. For the filtering phase, the service contracts listed in Table 1 are filtered by using the previously obtained user categories affiliation. In Table 3 the user category affiliation has been associated with the related Service Contract. Thus, the service contracts *cheap* (with *SeniorUser* category) and *pay-flex* (with *BusinessPlus* category) will be filtered out and they will not be further analyzed in the ranking process.

**Table 3.** Example of service contracts category filtering

| Provider | Contract | Category | User Membership |
|---|---|---|---|
| Provider A | *pay-flex* | *BusinessPlus* | no |
| Provider A | *high-trace* | *BusinessOne* | yes |
| Provider A | *secure* | *BusinessOne* | yes |
| Provider B | *fast-plus* | *SilverUser* | yes |
| Provider B | *fast* | *BronzeUser* | yes |
| Provider B | *cheap* | *SeniorUser* | no |

Let us now suppose that Mario Rossi interacts with the system to formulate a query. He selects the following pre-defined query: *"I am a user who needs to perform a transportation of a valuable good. I am looking for a FAST (at most 48 hours) delivery service having a blanket INSURANCE. I would like to receive TRACEABILITY information about the transportation"*. The user decides to modify the query by introducing some specific constraints; in particular he modifies the flexible constraint FAST into *at most 24 hours*, and he adds new constraints not defined in the query, such as *price at most 40€*, and *payment method=credit card and electronic transfer*. The user's query is enriched with the information provided by the user at registration time: the user is interested in *secure* and *cheap* services and his preferred information channels are *phone call* and *e-mail*.

For sake of simplicity, the evaluation process of each constraint will be described for one of the contracts listed before: the *fast-plus* contract. The same evaluation process will be then applied to the other contracts. By considering the *fast-plus* contract its NFPs evaluations are commented here below:

- **Hours to Delivery:** the evaluation of this NFP produces a matching degree of 1, since the service contract provides the delivery in 24hours as requested in the user query. The matching degree is calculated as explained in Section 5.2.
- **Insurance:** the *Fire & theft* insurance type provided by this contract is a subset of the insurance type required by the user (*Blanket*); the matching degree is 0.33 as the given contracts covers only one third of the *Blanket* one (the *Blanket* insurance is composed by the *Fire & theft*, *Damage* and *Loss* sub-insurances).
- **Price:** the service contract price (40€) is equal to what required from the user: the matching is fulfilled and the constraint evaluation function is 1.00 according to Section 5.2.
- **Payment Method:** the payment methods offered from this contract match only partially the user query: the *fast-plus* contract provides only the *credit card* method. The matching degree is 0.50.
- **Info. Channel:** the service do not provide any of the user requested Information Channels, for this reason the constraint evaluation function of this constraint is 0.00.
- **Description:** the service contract description contains only the terms *traceability* and *fast* defined in the user query. The estimated relevance degree is 0.50.

The GDoM matching degree for the *fast-plus* contract is finally evaluated as described in Formula 2; where $\sum_{i=1}^{nc} CF_i(sc, q) = 2.83$, and the final $DoM$ degree is evaluated as $DoM(sc, q) = \frac{2.83+0.50}{6} = 0.555$.

## 7 Experimental Evaluation

In order to assess the effectiveness of the proposed service contract selection strategy we adopt the normalized discounted cumulative gain (NDCG) measure [16]. This metric has been designed in order to compare IR methods with respect to their ability to favour relevant search results. DCG, *discounted cumulative gain*, measures the gain of a document based on its position in the result list. The gain is accumulated from the top of the result list to the bottom with the gain of each result discounted at lower ranks. In our scenario a result refers to a service contract, and for our tests we adopt the modified NDCG formulation proposed in [17]. This modification explicitly models a judgment value in addition to the ranking obtained after the application of the methodology presented in Section 5, and it normalizes the DCG values by comparing them with respect to an ideal rank. The ideal rank is obtained as an agreement of a pool of experts. In detail, we asked to 3 experts, given both a set of queries and a user profile, to independently indicate a judgement for each service contract with the consequence to obtain 3 ideal ranks. After this, the experts have indicated a common assessor on them to provide a unique ideal rank.

Given a ranked result set of service contracts $\mathcal{S}_r$, and an ideal ordering of the same set of service contracts $\mathcal{S}_i$, the (DCG) at a particular rank threshold $k$ is defined as $DCG(\mathcal{S}_r, k) = \sum_{i=1}^{k} \frac{2^{jdg(i)}-1}{log(1+i)}$, where $jdg(i)$ is the judgement (0=Bad, 1=Fair, 2=Good, 3=Excellent) at position $i$ in set $\mathcal{S}_r$.

The ideally ordered set $\mathcal{S}_i$ contains all service contracts rated for the given query sorted descending by the judgement values. Formally, the NDCG at a particular rank threshold $k$ is defined as:

$$NDCG(\mathcal{S}_r, k) = \frac{DCG(\mathcal{S}_r, k)}{DCG(\mathcal{S}_i, k)}, \tag{3}$$

Higher NDCG values correspond to better agreements with human judgements.

### 7.1 Experiments

To the best of our knowledge there is no benchmark defined to compare different service contract selection tools, consequently we have simulated the interaction of a user with our system. Thus, we have defined 32 service contracts from 5 distinct providers, and the NFPs on which contractual terms have been defined are those specified in Section 6. We asked to a user to perform three queries by increasing the complexity of each request. This means that for each new query a new constraint has been added. In details, the first query, i.e. $Q_1$= *"I am looking for a SECURE and FAST delivery service"*, has been selected by the user from the provided list without specifying any further detail. As indicated in Section 4, for each pre-defined query the constraints on attributes are identified; in query $Q_1$ the constraints are $insurance = blanket$ and $delivery <= 48\, hours$, respectively. In the second query, $Q_2$, the user specifies his/her meaning for the data *fast* as 24 for indicating a delivery services at the most of 24 hours. At the end, for the third query, $Q_3$, the user adds the data "traceability" in the free-text area in addition to 24 as a punctual value for the data-field *fast*. In order to show the effectiveness of our approach, we have performed simulations in different conditions by considering for each of them the above three queries as follows: (i) without considering the user context model (i.e., "case 1"), (ii) only by having the user information taken at registration time (i.e., "case 2"), (iii) only by considering the user history (i.e., "case 3"), (iv) only by considering information based on punctual values (i.e., "case 4"), (v) only by analyzing information obtained from textual descriptions (i.e., "case 5"), and at the end (vi) all the information provided in the previous steps that characterize the system described in this work (i.e., "case 6"). Thus, we have compared and evaluated the six approaches by applying the NDCG metric at various rank cuts (@5, @10, and @20). Fig. 3 shows the NDCG average values obtained for the above cases at different @-cuts.

By analyzing case 1, it emerges how the knowledge of user information allows to obtain better results in all the other cases where no additional information is considered with respect to the queries. In our system the usage of data prevails with respect to the textual description, and this implies a better performance in case 2 and case 4 with respect to case 5. Another consideration can be made by analyzing the use of the history information, case 3, where lower values are obtained with respect to the information taken at registration time. This means that our strategy gives more importance to the personal information of the user (i.e., his/her role/job, info languages, ...). By considering all cases our method (case 6) outperforms the other ones. This means that the proposed methodology produces higher NDCG as it preserves the ranking given by the ideal ranking better than the other cases.

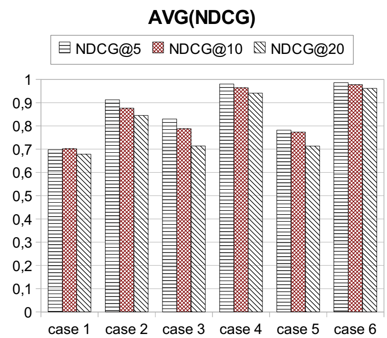


**Fig. 3.** Comparison of all the considered cases

# 8 Conclusions and Future Works

Service contract selection is an important factor to enhance service discovery. In this paper we have proposed a novel approach to support service contract selection based on semantic and IR techniques. The approach exploits precise and flexible preferences both explicitly defined by a user and implicitly inferred from his/her context. The user's preferences on the NFPs are formulated by means of a multi-constraint query that is used to filter and rank the service contracts offered by discovered services. The filtering is performed by evaluating the user categories, and the ranking is performed by aggregating the single constraint matching degrees of each service contract. Experimental results show the effectiveness of our approach to rank 32 service contracts from 5 distinct service providers according to 3 multi-constraint queries formulated by the user.

Our future research will address the problem of building a large benchmark of real service contracts to make comparative evaluations of different approaches possible. Moreover, we are investigating how to handle the management of qualitative NFPs (e.g., security and trust) which cannot be directly quantified. Finally, we are also studying how to integrate our approach with the aggregated search of data and services presented in [18].

## References


1. Papazoglou, M., Traverso, P., Dustdar, S., Leymann, F., Kramer, B.: Service Oriented Computing Research Roadmap. In: Dagstuhl Seminar Proceedings 05462 (SOC) (2006)
2. Toma, I., Roman, D., Fensel, D.: On describing and ranking services based on nonfunctional properties. In: Third International Conference on Next Generation Web Services Practices (NWESP 2007), pp. 61–66. IEEE Computer Society, Washington, DC, USA (2007)
3. Comerio, M., Truong, H.-L., De Paoli, F., Dustdar, S.: Evaluating Contract Compatibility for Service Composition in the se$CO_2$ Framework. In: Baresi, L., Chi, C.-H., Suzuki, J. (eds.) ICSOC-ServiceWave 2009. LNCS, vol. 5900, pp. 221–236. Springer, Heidelberg (2009)
4. Wang, Y., Tang, B., Huan, Z.: Study on contract selection decision based on hybrid supply chains. In: Proc. of the Inter. Conf. on Measuring Technology and Mechatronics Automation. ICMTMA 2010, Washington, DC, USA, pp. 572–575 (2010)
5. Talluria, S., Leea, J.Y.: Optimal supply contract selection. International Journal of Production Research 48(24), 7303–7320 (2011)
6. Yu, H.Q., Reiff-Marganiec, S.: A method for automated web service selection. In: Proc. of the Congress on Services (SERVICES), pp. 513–520 (2008)
7. Lamparter, S., Ankolekar, A., Studer, R., Grimm, S.: Preference-based selection of highly configurable web services. In: Proc. of the 16th International Conference on World Wide Web (WWW 2007), pp. 1013–1022. ACM, New York (2007)
8. Wang, X., Vitvar, T., Kerrigan, M., Toma, I.: A QoS-Aware Selection Model for Semantic Web Services. In: Dan, A., Lamersdorf, W. (eds.) ICSOC 2006. LNCS, vol. 4294, pp. 390–401. Springer, Heidelberg (2006)
9. Garcia, J.M., Toma, I., Ruiz, D., Ruiz-Cortes, A.: A service ranker based on logic rules evaluation and constraint programming. In: Proc. of 2nd Non Functional Properties and Service Level Agreements in SOCWorkshop (NFPSLASOC), Dublin, Ireland (2008)
10. Bajaj, S., Box, D., Chappell, D., et al.: Web Service Policy 1.2 - Framework (2006), `http://www.w3.org/Submission/2006/SUBM-WS-Policy-20060425/`
11. Keller, A., Ludwig, H.: The WSLA framework: Specifying and monitoring service level agreements for web services. Journal of Network and Systems Management 11(1), 57–81 (2003)
12. Palmonari, M., Comerio, M., De Paoli, F.: Effective and Flexible NFP-Based Ranking of Web Services. In: Baresi, L., Chi, C.-H., Suzuki, J. (eds.) ICSOC-ServiceWave 2009. LNCS, vol. 5900, pp. 546–560. Springer, Heidelberg (2009)
13. De Paoli, F., Palmonari, M., Comerio, M., Maurino, A.: A Meta-Model for Non-Functional Property Descriptions of Web Services. In: Proc. of the IEEE International Conference on Web Services (ICWS), Beijing, China, pp. 393–400 (2008)
14. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press (2008)
15. Stokoe, C.: Word sense disambiguation in information retrieval revisited. In: ACM SIGIR, pp. 159–166 (2003)
16. Järvelin, K., Kekäläinem, J.: Cumulated gain-based evaluation of ir techniques. ACM Transaction on Information Systems (TOIS) 20(4), 422–446 (2002)
17. Agrawal, R., Gollapudi, S., Halverson, A., Ieong, S.: Diversifying search results. In: WSDM 2009, pp. 5–14. ACM, New York (2009)
18. Palmonari, M., Sala, A., Maurino, A., Guerra, F., Pasi, G., Frisoni, G.: Aggregated search of data and services. Information Systems 36(2), 134–150 (2011)