**Polska O. V.,  Kudermetov R. K.**     *Zaporizhzhia Polytechnic National University, Zaporizhzhia, Ukraine*

**Zolotukhina O.A.**     *State University of Telecommunications, Kyiv, Ukraine*

**Shkarupylo V. V.**   *National University of Life and Environmental Sciences of Ukraine, Kyiv, Ukraine*

## A UML PROFILE FOR QUALITY-BASED WEB SERVICE SELECTION USING LOGIC SCORING OF PREFERENCE METHOD

*Abstract: In the modern information infrastructure, Web services have a very important role. Web services are one of the main tools for information creation, delivery and processing in society and economic life; for the formation and further evolution of the information space. Among the existing and constantly growing number of Web services, in a natural objective way, there are many Web services with the same or similar functional properties. This makes it possible to select among them such Web service that best satisfies the consumer's requirements and is most suitable to consumer's preferences on quality characteristics, for instance, cost, reliability, response time, etc. On the other hand, the consumer is not always possible to decide which Web service is the best from his point of view, because Web service has multiple quality characteristics. In this regard, many researchers formulate the problem of quality-based Web service selection as a multi-criteria decision-making problem and integrate the facilities for the decision-making into decision support systems. One of the methods of multi-criteria decision-making is the Logic Scoring of Preference (LSP) method. Given the current trends in the transition from document-based to model-based systems engineering, it is expedient to provide the design templates (meta-models or profiles) for the decision-making support systems developers.*

*In this paper, a UML profile to design the quality-based Web service selection systems using the LSP method was proposed. Standard UML extension mechanisms (stereotypes, tag value definitions, and constraints) to capture the basic quality properties of the Web service and the rules of applying the LSP method operators for this profile were used. The efficiency of the proposed profile by the examples of modelling the LSP aggregators to aggregate the Web service quality characteristics was demonstrated. The correctness of the models created using the proposed profile in the MagicDraw modelling environment was verified.*

*Keywords: UML, profile, quality of Web services, Logic Scoring of Preference.*

**Польська О. В.,  Кудерметов Р. К.**     *Національний університет «Запорізька політехніка», Запоріжжя, Україна*

**Золотухіна О.А.**     *Державний університет телекомунікацій, Київ, Україна*

**Шкарупило В. В.**     *Національний університет біоресурсів і природокористування України, Київ, Україна*

## UML ПРОФІЛЬ ДЛЯ ВИБОРУ ВЕБСЕРВІСУ ЗА ХАРАКТЕРИСТИКАМИ ЯКОСТІ З ВИКОРИСТАННЯМ МЕТОДУ ЛОГІЧНОГО ОЦІНЮВАННЯ ВПОДОБАННЯ

*Анотація: У сучасній інформаційній інфраструктурі вебсервіси відіграють дуже важливу роль. Вебсервіси є одним з основних інструментів для створення, доставки та обробки інформації в суспільстві та економічному житті; для формування та подальшої еволюції інформаційного простору. Серед існуючої та постійно зростаючої кількості вебсервісів, природно, є багато вебсервісів з однаковими або подібними функціональними властивостями. Це дає можливість вибрати серед них такий вебсервіс, який найкраще відповідає вимогам споживача та найбільш підходить за вподобаннями споживача щодо якісних характеристик, наприклад, вартості, надійності, часу відгуку тощо. З іншого боку, споживач не завжди може вирішити, який вебсервіс є найкращим з його точки зору, оскільки вебсервіс має безліч характеристик якості. У зв'язку з цим багато дослідників формулюють проблему вибору вебсервісів на основі якості як багатокритеріальну задачу прийняття рішень та інтегрують засоби для прийняття рішень у системи підтримки прийняття рішень. Одним із методів багатокритеріального прийняття рішень є метод логічного оцінювання вподобання (LSP). Враховуючи сучасні тенденції переходу від інженерії систем на основі документів до системної інженерії на основі моделей, доцільно надати шаблони проектування (метамоделі або профілі) для розробників систем підтримки прийняття*

рішень.

У цій роботі було запропоновано профіль UML для проектування систем вибору вебсервісів на основі характеристик якості із використанням методу LSP. Використовувались стандартні механізми розширення UML (стереотипи, визначення значень тегів та обмеження) для охоплення основних властивостей якості вебсервісу та правил застосування операторів методу LSP для цього профілю. Продемонстровано ефективність запропонованого профілю на прикладах моделювання агрегаторів LSP для агрегування характеристик якості вебсервісу. Було перевірено правильність моделей, створених із використанням запропонованого профілю в середовищі моделювання MagicDraw.

*Ключові слова:* уніфікована мова моделювання, профіль, якість веб-сервісів, логічне оцінювання вподобання.

**Польская О. В., Кудерметов Р. К.**   *Национальный университет «Запорожская политехника», Запорожье, Украина*

**Золотухина О.А.**   *Государственный университет телекоммуникаций, Киев, Украина*

**Шкарупило В. В.**   *Национальный университет биоресурсов и природопользования Украины, Киев, Украина*

## UML ПРОФИЛЬ ДЛЯ ВЫБОРА ВЕБ-СЕРВИСА ПО ХАРАКТЕРИСТИКАМ КАЧЕСТВА С ИСПОЛЬЗОВАНИЕМ МЕТОДА ЛОГИЧЕСКОЙ ОЦЕНКИ ПРЕДПОЧТЕНИЯ

*Аннотация:* В современной информационной инфраструктуре веб-сервисы играют очень важную роль. Веб-сервис является одним из основных инструментов для создания, доставки и обработки информации в обществе и экономической жизни; для формирования и дальнейшей эволюции информационного пространства. Среди существующего и растущего числа веб-сервисов, естественно, есть много веб-сервисов с одинаковыми или подобными функциональными свойствами. Это дает возможность выбрать среди них такой веб-сервис, который наилучшим образом соответствует требованиям потребителя и наиболее подходит по предпочтениям потребителя относительно качественных характеристик, например, стоимости, надежности, времени отклика и тому подобное. С другой стороны, потребитель не всегда может решить, какой веб-сервис является лучшим с его точки зрения, поскольку веб-сервис имеет множество характеристик качества. В связи с этим многие исследователи формулируют проблему выбора веб-сервисов на основе качества как многокритериальную задачу принятия решений и интегрируют средства для принятия решений в системы поддержки принятия решений. Одним из методов многокритериального принятия решений является метод логической оценки предпочтения (LSP). Учитывая современные тенденции перехода от инженерии систем на основе документов к системной инженерии на основе моделей, целесообразно предоставить шаблоны проектирования (метамодели или профили) для разработчиков систем поддержки принятия решений.

В этой работе был предложен профиль UML для проектирования систем выбора веб-сервисов на основе характеристик качества с использованием метода LSP. Использовались стандартные механизмы расширения UML (стереотипы, определения значений тегов и ограничения) для охвата основных свойств качества веб-сервиса и правил применения операторов метода LSP для этого профиля. Продемонстрирована эффективность предложенного профиля на примерах моделирования агрегаторов LSP для агрегирования характеристик качества веб-сервиса. Была проверена правильность моделей, созданных с использованием предложенного профиля в среде моделирования MagicDraw.

*Ключевые слова:* унифицированный язык моделирования, профиль, качество веб-сервисов, логическая оценка предпочтения.

### Introduction

There exist a large number of Web services with equal or similar functions on the Internet and consumer can select one that appropriates to the needed for him functionality. However, the Web service consumer may have different requirements for Web service quality characteristics, for instance, cost, response time, throughput, security, availability, etc. These characteristics are usually called Non-Functional Properties (NFPs) or Quality of Service (QoS) [1, 2, 3]. The problem of selecting a Web service based on QoS can be formulated as the problem of multi-criteria decision

making (MCDM) [4]. Many studies are devoted to QoS-based Web service selecting as the problem of MCDM [5, 6, 7]. Several studies [8, 9] were devoted to solving this problem using the Logic Scoring of Preference (LSP) method, proposed by J. Dujmović [10, 11]. The LSP method differs from other MCDM methods in that it allows more subtly consideration of user preferences to criteria (properties) by using the logic relations and conditions between properties of the object under the selection. The list of these relations and conditions are 'mandatory', 'sufficient', 'optional', 'simultaneity' and 'substitutability'. The use of these criteria characteristics together with such human perception characteristics as properties importance and preference, make it possible to more accurately evaluate the quality of the selectable object and, respectively, to make a decision close to optimal.

The number of criteria and aggregation functions in the professional use of the LSP method can reach several hundred. After building the hierarchical list of elementary criteria in the LSP method, the process of constructing an aggregation schema for these criteria follows, taking into account various logical conditions and user preferences. The aggregation of criteria done in a stepwise way. Groups of logically related criteria are aggregated together, creating subsystems of criteria. By repeating this process, the global criterion is obtained. This global criterion is an indicator of the overall suitability of the selected object. Due to stepwise aggregation and the possibility at each step to take into account the logical relationships between the criteria at each step, such a process of building a general aggregation schema is in a sense adjustable. Thus, the construction of a general schema for LSP aggregation in each specific case should follow certain logic of aggregation of elementary criteria and subsystems of criteria, taking into account the preferences of the decision maker. A problem of this kind with a large number of criteria cannot be reliably solved intuitively. These circumstances served as the main motivation for beginning the development of a UML profile, which, using stereotypes and associated with stereotypes tagged values and constraints can help in modelling, building and verifying the LSP aggregation schemas. This UML profile (called hereinafter QoSLSP profile) covers domain of the use of the LSP method for QoS-based Web service selection.

### 1. Analysis of literature and problem statement

The general-purpose visual modelling language UML is the powerful tool for specifying and documenting of the system's artefacts while the model-driven complex system development, verification and validation. Along with the UML universality, it has customization mechanisms for modelling specific domains (for example, business processes or real-time systems) that may be more appropriate. A set of UML extension mechanisms such as stereotypes, tag definitions, and constraints for specializing model elements allow customized extensions of UML for particular application domains. These UML extension mechanisms are grouped into a special package called a UML profile. This profile can be embedded or imported and applied to the elements of the domain model. A brief introduction to UML profiles and a discussion of the capabilities and benefits of UML profiles can be found, for example, in [12]. In a systematic review of UML profiles [13] can be seen the analysis of practical definitions and styles of the UML profiles.

Since the topic of Web service quality is relevant, many researchers are worked in the development of Web service quality models. The foundational documents on Web services quality are "Web Services Quality Factors Version 1.0" (WSQM) proposed by the consortium OASIS [14] and specification "UML Profile for Modeling Quality of Service and Fault Tolerance Characteristics and Mechanisms, v1.1" (QoS&FT) adopted by consortium OMG [15]. In WSQM the Web service quality factors and sub-factors have been defined. The definitions of these factors are very important for an unambiguous understanding of the Web service quality attributes by the stakeholders. WSQM only considers the conceptual side of quality characteristics and divides them into the business quality characteristics and the system quality one. At the QoS&FT specification, a UML QoS profile has been proposed that contains the key stereotypes, tag definitions and constraints needed to model the Web service quality attributes. This profile is very useful and is applied by many researchers for Web Service quality modelling. In paper [16] the QoS&FT profile

extended with concept and constructs from the proposed Quality-Value-Dependency-Priority (QVDP) model. This extension introduced a class to express the rules that define priorities of the quality characteristics for optimization when Web service is selected. In [17], the same authors separated the QoS&FT profile to make an explicit distinction between the requirements of the requester and the provider's proposals regarding the quality characteristics of the Web service. Thereon two quality models were used to select the best Web service using fuzzy multi-criteria decision analysis. The comprehensive survey of Web service quality models and meta-models (SQMM) was presented in [18]. The main conclusion of this fundamental work is that no SQMM satisfies all the requirements, and this lack prevents the widespread use of SQMM in Web services management systems. This conclusion must be the motivation for extending the existing SQMMs and developing a new one. A UML profile for Modeling and Analysis of Real-Time and Embedded systems (MARTE) issued by the OMG [19] has a powerful sub-profile QoS, which compliance with the UML profile QoS&FT. The MARTE profile includes the BasicNFP_Types library with data types for specifying the nonfunctional properties and mechanisms for define custom types. These tools are quite applicable to specifying the QoS properties of Web services. The disadvantages of MARTE are that there exist only a few tools supporting this profile and a large specification (close to 800 pages), which can hold back potential users [20].

As a result of the literature review, it can be stated that there is no UML profile QoS-based Web service selection using the LSP method.

A detailed description of the LSP method can be found in [11]. Here is provided only the information required developing the UML QoSLSP profile. In the LSP method, aggregators such as Graded Conjunction/Disjunction (GCD), Conjunction Partial Absorption (CPA) and Disjunction Partial Absorption (DPA) are mainly used to aggregate criteria. The schemas of these aggregators are shown in Fig. 1. The criteria to be aggregated are divided into three types: mandatory, sufficient and optional. The mandatory criteria are the criteria that must be satisfied. The sufficient criteria are the criteria that can completely satisfy the requirements regardless of the degrees of satisfaction of other criteria. The optional criteria are desired criteria. GCD aggregators are used to aggregate criteria of the same type. GCD aggregator is implemented using the weighted power mean (WPM) function:

$$E = (w_1 E_1^r + \ldots + w_m E_m^r)^{1/r},$$

$$0 \le w_j \le 1, \ \sum_{j=1}^{m} w_j = 1,$$

where $E$ is an output of GCD aggregator; $E_j$ is the $j$-th input criterion, $w_j$ is the weight of $j$-th input criterion and the parameter $r$ is a real number used to express logical relation between the input criteria. The main logical relationships between the criteria are set by the *andness* parameter. The values of *andness*, their corresponding $r$ parameter values, symbols for designating GCD aggregators, and the types of criteria aggregated by these aggregators are presented in Tab. 1.

*Table 1*

Symbols and parameters of GCD aggregators

| *andness* | Symbol of aggregator | $r$, (*m*=2) | $r$, (*m*=3) | Criterion type |
|---|---|---|---|---|
| 0 | D | +∞ | +∞ | Sufficient |
| 1/16 | D++ | 20.63 | 24.3 | |
| 1/8 | D+ | 9.521 | 11.095 | |
| 3/16 | D+- | 5.802 | 6.675 | |
| 1/4 | DA | 3.929 | 4.45 | |
| 5/16 | D-+ | 2.792 | 3.101 | Optional |
| 3/8 | D- | 2.018 | 2.187 | |
| 7/16 | D-- | 1.449 | 1.519 | |
| 1/2 | A | 1 | 1 | |
| 9/16 | C-- | 0.619 | 0.573 | |
| 5/8 | C- | 0.261 | 0.192 | |

| 11/16 | C-+ | -0.148 | -0.208 | |
| 3/4 | CA | -0.72 | -0.732 | |
| 13/16 | C+- | -1.655 | -1.55 | |
| 7/8 | C+ | -3.51 | -3.114 | Mandatory |
| 15/16 | C++ | -9.06 | -7.639 | |
| 1 | C | $-\infty$ | $-\infty$ | |

To aggregate a mandatory criterion with an optional criterion, the CPA aggregator is used (Fig. 1b). To aggregate a sufficient criterion with an optional criterion, the DPA aggregator is used (Fig. 1c). The weights $W_1$ and $W_2$ for CPA/DPA aggregators depend of penalty and reward, which set the decision maker. The weights $W_1$ and $W_2$ can be determined from special tables, developed by J. Dujmović [21]. Fragment of the table for determining $W_1$ and $W_2$ weights for CPA aggregator, composed of aggregators A and CA is shown in Tab. 2 [11].
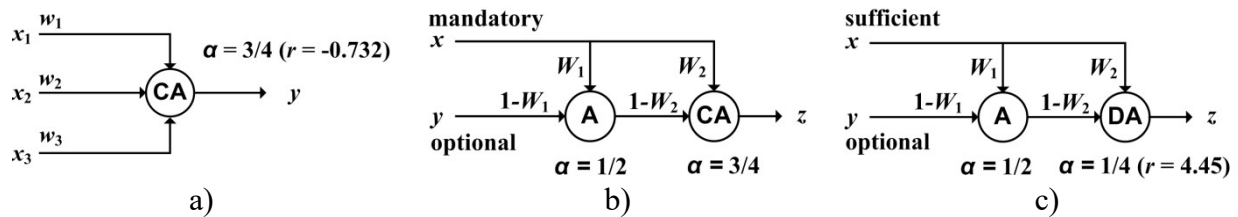


Fig. 1. The main LSP aggregators examples: a) GCD, b) CPA, c) DPA

*Table 2*

Fragment of the table for W1 and W2 determining

| $W_1$ $W_2$ | | Penalty, % | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| | | **50** | **45** | **40** | **35** | **30** | **25** | **20** |
| Reward, % | **5** | 8 87.5 | 9.5 87 | 13 86.5 | 17.5 85.5 | 19.5 87 | 30.5 83 | 36 84 |
| | **10** | 18 73.5 | 22.5 72 | 27 71.5 | 32.5 70.5 | 40 68.5 | 49.5 65 | 61.5 58.5 |
| | **15** | 26 60.5 | 31.5 58.5 | 37 57.5 | 44.5 54 | 53 49.5 | 63 41.5 | 74 26.5 |
| | **20** | 32.5 48 | 38.5 45.5 | 44.5 42 | 53 37 | 62 29 | 71.5 16 | |
| | **25** | 37.5 37 | 44 33 | 51 28.5 | 59 21.5 | 67.5 10.5 | | |

The reviewed in this section UML profiles can be used to describe the QoS in Web service selection problems solved by MCDM methods that have a simple aggregation function (for example, Simple Additive Weighting method). Using the LSP method to QoS-based Web service selection requires taking into account the strict logical relationships between the quality characteristics and performing their stepwise hierarchical aggregation. A UML profile can help to reduce logical errors in this process by fixing the main properties of QoS in tagged values and impose constraints on relationships between them. In addition, the UML profile can provide automatic verification of the developed QoS aggregation LSP schema when using suitable UML modelling tools.

**The purpose of this work** is to develop a UML profile that allows creating 'well-formed' models for QoS-based Web service selection using the LSP method.

## 2. UML QoSLSP profile

The QoSLSP profile refers to the M2-level of a four-layer meta-modelling architecture, which means the user's model elements on M1 and M0 levels can be instantiated via stereotyping. The QoSLSP profile has been defined using standard extension mechanisms: stereotypes, tag definitions and constraints. The proposed profile includes three main concepts of the domain area of the

interest: QoS criterion, LSP aggregator and relationship between them. The first two are mapped as the stereotypes, and the third is presented as an association between these stereotypes. Stereotypes, main tag definitions and constraints of the QoSLSP profile are shown in Fig. 2.
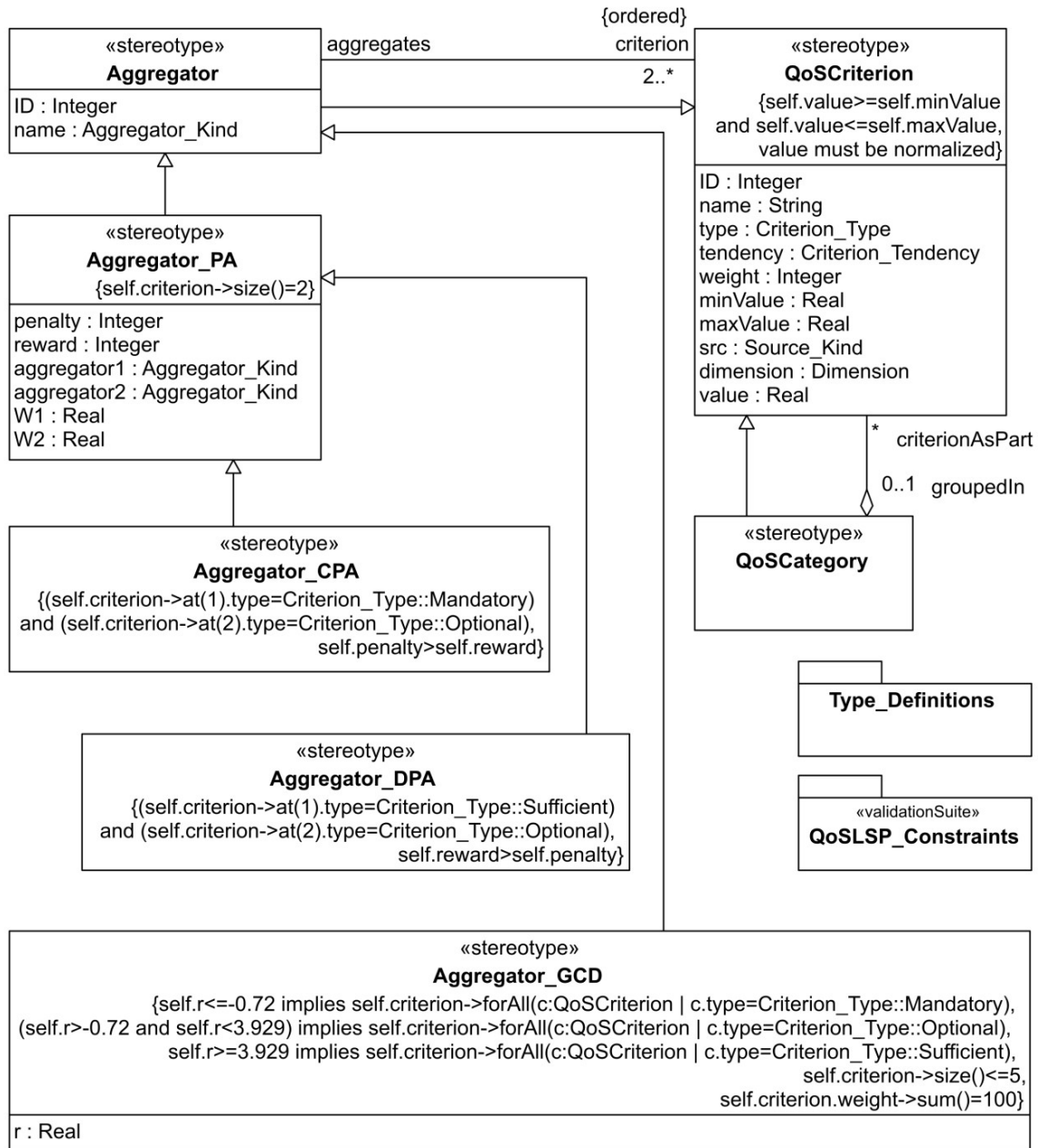


Fig. 2. The QoSLSP stereotypes

The semantic peculiarities for three aggregator types are expressed through the tag definitions and constraints and by the generalization relation (inheritance). The most general aggregator meta-model is defined as the «Aggregator» stereotype, which has only two tag definitions: the ID tag and the name tag. The ID tag has the Integer type. For the tag name the type is defined as the enumeration «Aggregator_Kind», which includes the enumeration literals denote all kinds of aggregators (list of GCD type aggregators, CPA and DPA aggregators) (Fig. 3). The stereotype «Aggregator» generalizes the properties of all aggregators. An auxiliary stereotype «Aggregator_PA» (Partition Absorption) generalizes the properties of the CPA and DPA aggregators. This stereotype has the following tags:

      − Penalty and Reward tags specify the penalty and reward, consequently;

– `aggregator1` and `aggregator2` tags must be used to indicate which aggregator kinds form the CPA or DPA aggregators (Fig. 1b, 1c);

– `W1` and `W2` tags specify the weights $W_1$ and $W_2$, which the user determines from penalty and reward values (e.g. from Tab. 1.).

| «enumeration» **Criterion_Type** | «enumeration» **Source_Kind** | «enumeration» **Aggregator_Kind** |
|---|---|---|
| Mandatory<br>Optional<br>Sufficient | meas<br>eval<br>calc | D++<br>D+<br>D+-<br>DA<br>D-+<br>D-<br>D--<br>A<br>C--<br>C-<br>C-+<br>CA<br>C+-<br>C+<br>C++<br>CPA<br>DPA |

| «enumeration» **Criterion_Tendency** | «enumeration» **Dimension** | |
|---|---|---|
| benefit<br>cost | [invokes/s]<br>[ms]<br>[%] | |

Fig. 3. The QoSLSP type definitions

In the model, the user can use `Penalty` and `Reward` tags as the work tags, then `W1` and `W2` tags can be used only for informational purposes or not at all. Similarly, if the user already knows the $W_1$ and $W_2$ weights then he can use the `W1` and `W2` tags as the main ones, and the `Penalty` and `Reward` tags as informative.

In QoSLSP profile the set of constraints associated to stereotypes are defined. The constraints impose the restrictions on the meta-model elements and help to create a 'well-formed' model, which can be verified by appropriate UML modelling tools. For this purpose, the QoSLSP profile constraints have been expressed in Object Constraint Language (OCL) [22, 23]. For example, the «Aggregator_PA» stereotype has constraint which restricts the number of inputs (input criteria) for CPA and DPA aggregators. The OCL expression for this constraint can be written as `{self.critrion->size()=2}`.

The stereotypes «Aggregator_CPA» and «Aggregator_DPA» inherit all the properties of the main stereotype «Aggregator» and auxiliary stereotype «Aggregator_PA», i.e. the tag definitions and the constraints. These stereotypes only impose additional restrictions on the input criteria characteristics and the values of defined by user the penalty and the reward. So, according to the CPA aggregator definition and the Tab. 1, its first input criterion must be criterion of mandatory type, the second criterion must be optional type, and the penalty value must be greater than the reward value. For the DPA aggregator first input criterion must be sufficient type, second criterion must be optional type, and the reward must be greater than the penalty.

The «Aggregator_GCD» stereotype is a direct inheritor to the «Aggregator» stereotype. The GCD aggregator is recommended to use only for aggregating the same type criteria. The GCD aggregator kind is determined by the *andness* value, which sets the value of the parameter $r$ for the WPM function. In this regard, with the «Aggregator_GCD» stereotype the corresponding constraints are associated that restrict the possible types of input criteria depending on the parameter $r$ value ranges. If the parameter $r$ is less or equal to –0.72, then types of all input criteria must be 'mandatory'. If the parameter $r$ is greater or equal to 3.929, then types of all input criteria must be 'sufficient'. If the parameter $r$ is greater than –0.72 and less than 3.929, then types

of all input criteria must be 'optional'. The stereotype «`Aggregator_GCD`» is also associated with the constraints requiring the sum of criteria weights must be equal to 100 and the number of input criteria must be less or equal to 5. The last constraint is in a sense arbitrary and the user can change the allowed number of input criteria in the constraint text. Nevertheless, many studies have shown that the number of characteristics evaluated simultaneously depends on the experience and professionalism of the evaluator. And to obtain the relevant assessments about the evaluated characteristics, it is recommended to evaluate no more than 5-7 characteristics at the same time.

Another basic stereotype of the QoSLSP profile is the «`QoSCriterion`» stereotype. It includes required and informative tags that characterize the QoS criteria. Descriptions of the «`QoSCriterion`» stereotype tags are presented in Table 3.

*Table 3*

Tags of «`QosCriterion`» stereotype

| Tag | Type | Remarks |
|---|---|---|
| `ID` | `Integer` | Required |
| `name` | `String` | Informative |
| `type` | «`Criterion_Type`» | Required. Refers to enumerated literals `Mandatory`, `Sufficient`, `Optional` |
| `tendency` | «`Criterion_Tendency`» | Informative. The criterion tendency can be either `benefit` or `cost`. 'Benefit' tendency means that higher value of the criterion is better than the lower one (e.g. throughput). 'Cost' tendency means the lower value of criterion is better than higher one. |
| `weight` | `Integer` | Weight must be greater than 0 and less than 100. If the criteria aggregated by the GCD aggregator then weight is required and is not used otherwise. |
| `minValue`, `maxValue` | `Real` | Required/Informative. These tagged values are used for normalizing the value of the criterion. |
| `src` | «`Source_Kind`» | Informative. Enumerated literal `meas` means that the criterion value is obtained by measuring, `eval` – by evaluation and `calc` – by calculating. |
| `deimension` | «`Dimension`» | Informative. This tag specifies the dimension unit of the criterion value. |
| `value` | `Real` | Required/Informative. |

Sometimes the QoS criteria need to be grouped into QoS categories – composite criterion made up of multiple criteria. For this purpose, the QoSLSP profile provides for the «`QoSCategory`» stereotype. This stereotype is associated with the «`QoSCriterion`» stereotype by the aggregation relation. Since the composite criterion is also a criterion, the «`QoSCategory`» stereotype inherits all the properties of the «`QoSCriterion`» stereotype.

Here there is a need to clarify the use of the «`QoSCriterion`» stereotype. Depending on the purpose and stage of modelling, the «`QoSCriterion`» stereotype can be used for UML modelling of both a QoS input attribute of LSP aggregators and for modelling an QoS criterion. The difference is that the value of the QoS input attribute is not normalized, and the QoS criterion value is normalized. If the «`QoSCriterion`» stereotype is used to stereotyping the QoS attribute modelling element, then it is assumed that the tagged value '`value`' should be normalized and tagged values `minValue` and `maxValue` are required. If the «`QoSCriterion`» stereotype is

used to stereotyping the QoS criterion modelling element, then it is assumed that the tagged value 'value' is normalized and tagged values 'value', minValue and maxValue are informative.

The «Aggregator» stereotype and therefore all of its child stereotypes inherit the properties of the «QoSCriterion» stereotype. This relationship is necessary because the output of an aggregator can be one of the inputs of another aggregator. The stereotypes «Aggregator» and «QoSCriterion» are connected by the association relationship. The role criterion of this association has the multiplicity 2..* and imposed standard UML constraint {ordered}. This constraint required because the input criteria for CPA and DPA aggregators must be ordered: the first criterion must be mandatory (for CPA aggregator) or sufficient (for DPA aggregator); the second criterion must be optional.

### 3. Application of the QoSLSP profile

This section demonstrates an application of the QoSLSP profile for modelling a complex LSP aggregator. To synthesize the aggregation schema of the LSP method, a hierarchical tree of the Web service quality attributes was used. This tree is shown in Fig. 4. The formation of the criteria tree was carried out taking into account the existing standard of quality factors for Web services [14] and a dataset of quality characteristics of real Web services [24]. The rationale for choosing the Web services quality attributes and the hierarchical tree formation procedure was described in [25].
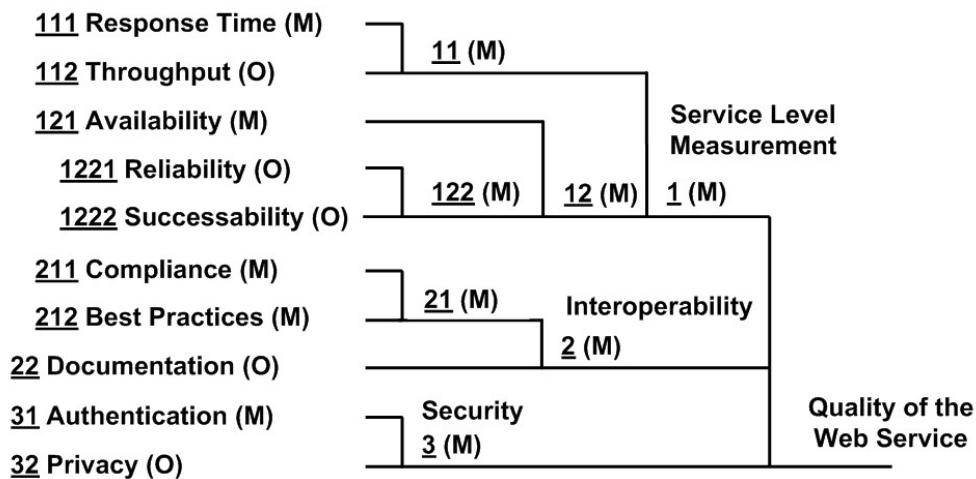


Fig. 4. The tree of the Web service quality attributes, M – mandatory, O – optional

The modelled composite aggregation schema is shown in Fig. 5. To synthesize this aggregation schema, it is necessary to select the appropriate aggregation functions for each group of the quality attribute tree according to Tab. 1. So, to aggregate the mandatory attributes, the decision-maker should determine the degree of simultaneity of these criteria. In this case, the *andness* degree should be in the range from 3/4 to 15/16. For instance, if the operation C+- was chosen then a parameter $r$ of the WPM function equal to -1.655. Then the decision-maker should determine the relative importance of the aggregated attributes, i.e. their weights. Similarly, for the aggregation of optional attributes, it needs to choose the *andness* degree in the range from 5/16 to 11/16, select the GCD aggregator and assign the weights of the aggregated attributes. To aggregate the mandatory attribute with the optional attribute the CPA function is used. To do this, you need to select two GCD aggregators (see Fig. 1b) and the corresponding penalty and reward. Aggregation schema synthesis modelled in this work described in detail in [25]. Other examples of composite aggregation schema building can be found in [11, 26].
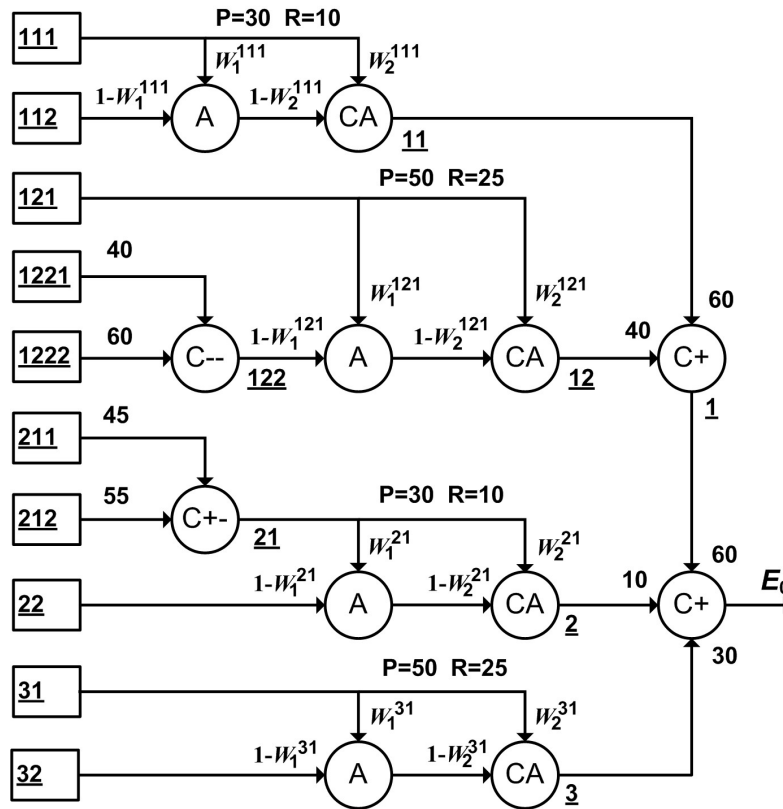
Fig. 5. The composite aggregation schema

Figure 6 shows how the stereotype «QoSCriterion» may be applied to define the attributes of the WebService class. WebService class attributes, which are Web service quality characteristics, are stereotyped (marked) by the «QoSCriterion» and «QoSCategory» stereotypes. This means that each of the WebService class attributes has associated tagged values specified by the tag definitions belonging to the «QoSCriterion» stereotype. Using these tagged values, a developer can set the values of quality attributes common to all instances of the WebService class, and using constraints to control whether the tagged values of class instances are satisfied to conditions imposed.



| WebService |
|---|
| «QoSCriterion»ResponseTime : Real{ID = 111, dimension = [ms], maxValue = 150.0, minValue = 50.0, src = meas, tendency = cost, type = Mandatory} |
| «QoSCriterion»Throughput : Real{ID = 112, dimension = [invokes/s], maxValue = 25.0, minValue = 5.0, src = meas, tendency = benefit, type = Optional} |
| «QoSCriterion»Availability : Real{ID = 121, dimension = [%], maxValue = 100.0, minValue = 60.0, src = eval, tendency = benefit, type = Mandatory} |
| «QoSCriterion»Reliability : Real{ID = 1221, dimension = [%], maxValue = 100.0, minValue = 40.0, src = eval, tendency = benefit, type = Optional, weight = 40} |
| «QoSCriterion»Successability : Real{ID = 1222, dimension = [%], maxValue = 100.0, minValue = 20.0, src = eval, tendency = benefit, type = Optional, weight = 60} |
| «QoSCriterion»Compliance : Real{ID = 211, dimension = [%], maxValue = 100.0, minValue = 30.0, src = eval, tendency = benefit, type = Mandatory, weight = 45} |
| «QoSCriterion»BestPractices : Real{ID = 212, dimension = [%], maxValue = 100.0, minValue = 40.0, src = eval, tendency = benefit, type = Mandatory, weight = 55} |
| «QoSCriterion»Documentation : Real{ID = 22, dimension = [%], maxValue = 100.0, minValue = 20.0, src = eval, tendency = benefit, type = Optional} |
| «QoSCategory»Authentication : Real{ID = 31, maxValue = 6.0, minValue = 1.0, src = calc, tendency = benefit, type = Mandatory} |
| «QoSCriterion»Privacy : Real{ID = 32, maxValue = 1.0, minValue = 0.0, src = eval, tendency = benefit, type = Optional} |
| +serviceFunction( message : String ) : Integer |

Fig. 6. The WebService class with stereotyped attributes

Figure 7 shows how the «QoSCriterion» and «QoSCategory» stereotypes may be used to modelling the complicated quality attribute of Web service. If a certain quality attribute is formed as a function of several elementary attributes, the QoSLSP profile allows one to model such a category, which, inheriting the properties of the QoS criterion, can be further used as an ordinary quality attribute.
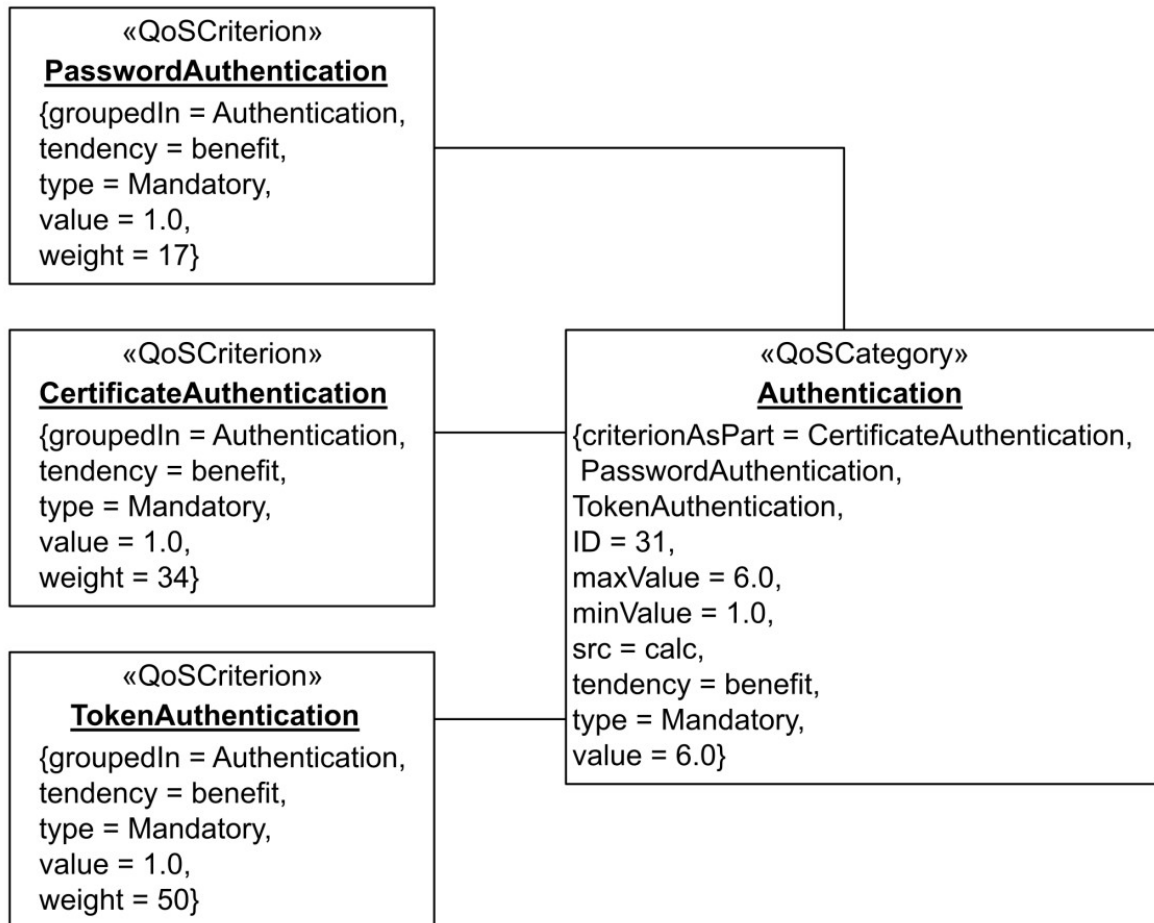


Fig. 7. An example of modelling a category of quality attributes

Figure 8 shows a diagram of modelling the part of the LSP aggregation schema shown in Fig. 5. 'Reliability' and 'Successability' criteria are both optional, so they must be aggregated by one of the GCD aggregators. In this case, the aggregator C-- is used, this aggregator may be realized by WPM function with the parameter $r = 0.619$. The result of this aggregation has, in turn, all the properties of the QoS criterion, and the type of this criterion is also optional. To aggregate this result with the mandatory criterion 'Availability' the CPA aggregator must be applied. To implement this CPA aggregator the GCD aggregators A and CA are chosen. The penalty and reward for this CPA aggregator are set (50:25) and according to Tab. 2, the weights W1 and W2 are set. The correctness of the UML and QoSLSP profile applying for modelling aggregation schemas, fulfilment of UML general constraints and QoSLSP profile constraints was verified in the MagicDraw modelling tool. To verifying the LSP aggregation schemas models for compliance with the QoSLSP profile constraints, a rule package QoSLSP_Constraints was created, which contains the references to all OCL constraints of the QoSLSP profile.
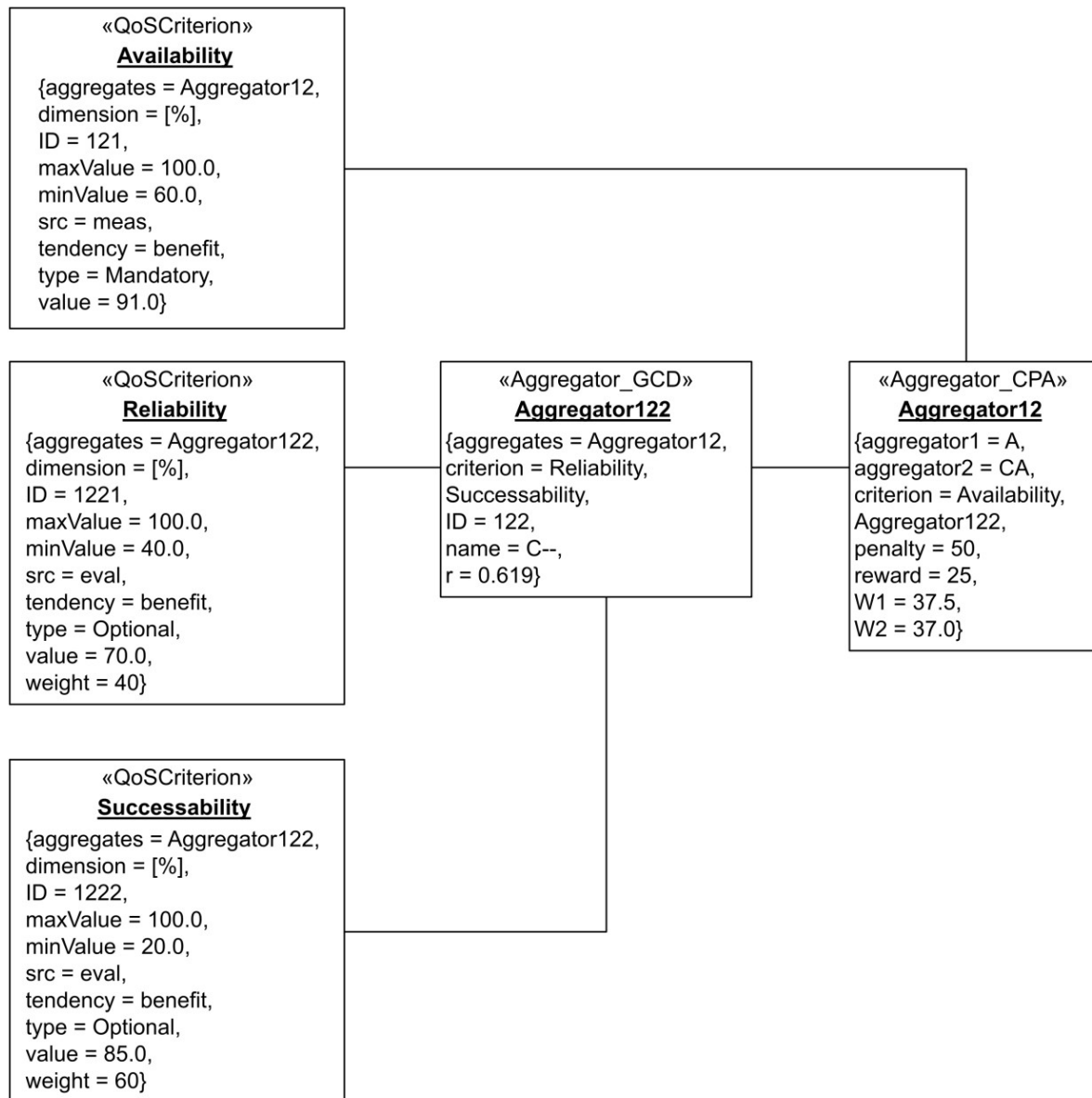
Fig. 8. An example of modelling the part of the LSP aggregation schema

## 4. Discussion

As can be seen from the modelling results (Figs. 6, 7, 8), the proposed QoSLSP profile enables customize UML for the model-driven synthesis of complicated LSP aggregators. The profile takes into account key aspects of the Web service selection process using the LSP method. The profile includes the definitions of key tags required by both the Web service quality concept and the LSP method that uses these tags to a logical score of preference. These include such tags for representing quality characteristics as relative importance, the requirement to criteria simultaneity or the enabling of criteria substitutability (weight, mandatory, sufficient and optional requirements). They also include characteristics that can be used for other MCDM methods (weight, value, ranges of value, tendency and dimensions). With the help of OCL constraints imposed on the types of input aggregated criteria, the issue of control over the correct use of LSP aggregators is solved. This ensures that extreme combinations such as joint aggregation of the criteria different types using GCD functions, high *andness* and optional input criteria or low *andness* and mandatory input criteria are avoided. These extreme combinations may arise with the decision maker's insufficient experience with the LSP method. Given the simplicity of implementation, the QoSLSP profile can be easily supplemented with new tags and constraints or even is used as a base for developing new

LSP-oriented profiles, for instance, for modelling families of the Uniform GCD aggregators (UGCD) and General GCD aggregators [11], because the current QoSLSP profile version is intended only for modelling of the WPM.17 aggregators, which have medium granulated *andness*.

### Conclusions

The QoS-based Web service selection remains in the interest of many researchers, as evidenced by numerous scientific publications. The issues of further formalization and clarification of the concepts of Web service qualitative characteristics are urgent. The development of the patterns, meta-models and profiles for modelling both the Web service quality attributes and the procedures for their assessment can provide great support in solving issues on this topic. In this context, the contribution of modelling languages such as UML and its profiles can be significant.

In this work, a UML profile QoSLSP for modelling the QoS-based Web service selection using the LSP method was developed. The results of this profile verification showed that this profile produces reasonably accurate models of Web service QoS criteria and LSP aggregators.

The scientific novelty of this work is that the QoSLSP profile covers simultaneously both the Web service quality characteristics and the LSP method aggregators for evaluating these characteristics. Such profile was proposed for the first time judging by the publications authors are aware of.

Prospects for further research are expanding the capabilities of the QoSLSP profile by using the BasicNFP_Types library of the UML MARTE profile and the stereotype «Requirement» of the UML profile SysML [27] to manage Web service quality requirements.

### References

1. Becha, H., & Amyot, D. (2015). Consented consumer-centric non-functional property description and composition for SOA-based applications. *International Journal of Web Engineering and Technology, 10*(4), 355-392. doi:10.1504/IJWET.2015.073949

2. Kaewbanjong, K. & Intakosum, S. (2015). QoS Attributes of Web Services: A Systematic Review and Classification. Journal of Advanced Management Science, 3(3), 194-202. doi:10.12720/joams.3.3.194-202

3. Dee Castro, C. F., & Fantinato, M. (2018). *Dictionary of Non-Functional Requirements of Business Process and Web Services* Technical Report 003/2018, Graduate Program of Information Systems, Univ. of São Paulo Retrieved from https://www.researchgate.net/publication/

4. Tran, V. X., Tsuji, H., & Masuda, R. (2009). A new QoS ontology and its QoS-based ranking algorithm for web services. *Simulation Modelling Practice and Theory, 17*(8), 1378-1398. doi:10.1016/j.simpat.2009.06.010

5. Maheswari, S., & Karpagam, G. R. (2018). Performance evaluation of semantic based service selection methods. *Computers and Electrical Engineering, 71*, 966-977. doi:10.1016/j.compeleceng.2017.10.006

6. Hosseinzadeh, M., Hama, H. K., Ghafour, M. Y., Masdari, M., Ahmed, O. H., & Khezri, H. (2020). Service selection using multi-criteria decision making: A comprehensive overview. *Journal of Network and Systems Management, 28*(4), 1639-1693. doi:10.1007/s10922-020-09553-w

7. Kazar, O., Rezeg, K., & Belouaar, H. (2017). Web service selection based on TOPSIS algorithm. Paper presented at the *Proceedings of the 2017 International Conference on Mathematics and Information Technology, ICMIT 2017, 2018-January,* 177-182. doi:10.1109/MATHIT.2017.8259713

8. Yu, H. Q. and Molina, H. (2007). A modified Logic Scoring Preference method for dynamic Web services evaluation and selection. *Service Oriented Computing : 2nd European Seminar for Young Researchers, Leicester, 11-12 June 2007,* 87-93. Retrieved from https://www.cs.le.ac.uk/events/yrsoc2007/downloads/proceedings.pdf

9. Yu, H. Q., & Reiff-Marganiec, S. (2008). A method for automated web service selection. Paper presented at the *Proceedings - 2008 IEEE Congress on Services, SERVICES 2008, PART*

*1* 513-520. doi:10.1109/SERVICES-1.2008.8

10. Dujmovic, J. (1996). A Method For Evaluation And Selection Of Complex Hardware And Software Systems. *Int. CMG Conference.*

11. Dujmović, J. (2018). Soft computing evaluation logic the LSP decision method and its applications. *Soft computing evaluation logic: The LSP decision method and its applications* (pp. 1-880) doi:10.1002/9781119256489

12. Fuentes-Fernández, L., & Vallecillo, A. (2004). An Introduction to UML Profiles. Retrieved from https://www.academia.edu/9333573/An_Introduction_to_UML_Profiles

13. Pardillo, J. (2010). *A systematic review on the definition of UML profiles* doi:10.1007/978-3-642-16145-2_28 Retrieved from www.scopus.com

14. OASIS (2012). Web Services Quality Factors Version 1.0. Retrieved from http://docs.oasis-open.org/wsqm/WS-Quality-Factors/v1.0/cos01/WS-Quality-Factors-v1.0-cos01.html

15. OMG (2008). UML Profile for Modeling QoS and FT Version 1.1. Retrieved from https://www.omg.org/spec/QFTP/1.1

16. Jureta, I. J., Herssens, C., & Faulkner, S. (2009). A comprehensive quality model for service-oriented systems. *Software Quality Journal, 17*(1), 65-98. doi:10.1007/s11219-008-9059-2

17. Herssens, C., Jureta, I. J., & Faulkner, S. (2008). *Capturing and using QoS relationships to improve service selection* doi:10.1007/978-3-540-69534-9_25 Retrieved from www.scopus.com

18. Kritikos, K., Pernici, B., Plebani, P., Cappiello, C., Comuzzi, M., Benrernou, S., . . . Carro, M. (2013). A survey on service quality description. *ACM Computing Surveys, 46*(1) doi:10.1145/2522968.2522969

19. OMG (2019). UML Profile for MARTE Version 1.2. Retrieved from https://www.omg.org/spec/MARTE/1.2

20. Selic, B., & Gérard, S. (2013). Modeling and analysis of real-time and embedded systems with UML and MARTE. *Modeling and analysis of real-time and embedded systems with UML and MARTE* (pp. 1-287) doi:10.1016/C2012-0-13536-5.

21. Dujmović, J. (1979). Partial absorption function. *Journal of the University of Belgrade, EE Dept., Series Mathematics and Physics,* 659, 156-163. Retrieved from http://www.jstor.org/stable/43668110

22. OMG (2014). Object Constraint Language. Version 2.4. Retrieved from https://www.omg.org/spec/OCL/2.4/

23. Cabot, J., & Gogolla, M. (2012). *Object constraint language (OCL): A definitive guide* doi:10.1007/978-3-642-30982-3_3

24. The Quality of Service for Web Services (QWS) Dataset v2.0. (2019). Retrieved from https://github.com/qwsdata/qwsdata.github.io/releases

25. Polska, O. V., Kudermetov, R. K., & Shkarupylo, V. V. (2020). The approach for QoS based web service selection with user's preferences. *Scientific papers of Donetsk National Technical University. Series "Problems of modeling and design automation",16*, 19-27. doi: 10.31474/2074-7888-2020-2-19-27

26. Dujmović, J. J., De Tré, G., & Van de Weghe, N. (2010). LSP suitability maps. *Soft Computing, 14*(5), 421-434. doi:10.1007/s00500-009-0455-8

27. OMG (2019). OMG Systems Modeling Language (SysML) Version 1.6. Retrieved from https://www.omg.org/spec/SysML/1.6/