Abstract
The Unified Modeling Language (UML) is becoming the lingua franca in software development. One of the reasons of its popularity is the possibility to enrich the set of modeling elements of the language via the language’s extension mechanisms. Such a customization of the language leads to a growing complexity of language’s constructs. To improve understanding of such extensions, a set of measurements for them is required. This paper proposes three sets of metrics useful in measuring the size of the extensions and measuring the size of designs with stereotypes. The metrics are used in our research in order to compare various extensions with respect to their size and complexity and in order to provide a means for comparing quality of extensions.

1. Introduction
The Unified Modeling Language (UML, [1]) was designed in such a way that it consists of standard modeling elements contained in the language specification and a set of special modeling elements for extending the language. The extension mechanisms are based on the notion of stereotype – a classifier allowing for redefinition of the standard modeling element’s meaning. Extending of the language is gaining its popularity mainly due to the customization efforts in such contexts as the Model Driven Architecture (MDA, [2]) or a lightweight version of meta-modeling [3]. An important aspect in such language engineering effort is how to size and compare stereotypes and their groups – profiles. In such activities as language engineering, the metrics for stereotypes and profiles are significantly important as they measure the size of a profile or a stereotype while measuring productivity or quality.

Although there are several metric suites for object-oriented designs, none of them provides stereotype-specific or profile-specific metrics. This causes problems since the nature of stereotypes is different than that of other design elements, e.g. the most similar ones: classes as stereotypes have no operations. Furthermore, the meaning of such elements as constraints is different in the context of stereotypes. The metrics presented in the paper are intended to complement existing metric suites, and thus the relationship of our metrics and a sample of existing suites is discussed at the end of the paper. The metrics shown in this paper are used in our research on language customization (c.f. [4]) for such purposes as:
- comparing the size of various profiles/stereotypes,
- comparing the quality of profiles after normalizing them by their size, and
- refactoring profiles and stereotypes.

The paper is structured as follows. Section 2 presents a short overview of the definition of the notion of stereotype and profile. Section 3 presents the metrics for sizing stereotypes, while Section 4 presents the metrics for sizing profiles. Section 5 presents a discussion on the applicability of the existing metric suites for sizing profiles and stereotypes and Section 6 presents conclusions.

2. Definition of stereotypes and profiles
This section describes the way in which stereotypes and profiles are defined in UML 2.0, which is required for further discussions. The considered definition of a profile and a stereotype notions is taken from the UML 2.0 Infrastructure specification [1], where the full details of the syntax and semantics of these notions are defined. Fig. 1 presents a simplified version of the definition of stereotypes. The figure shows the necessary information that is required for further definition of metrics.

![Fig. 1. Overview of the stereotype definition in the UML 2.0 meta-model.](image-url)

Stereotypes are defined as a subclass of the meta-class Class, so they are allowed to possess all properties of
classes in the UML (i.e. they are special kind of classes in UML 2.0). As every class, each stereotype might have attributes, which are also known as tag definitions (in the same way as in UML 1.x). The instances of these properties can be called tagged values – we shall use these terms in order to stress that these properties are part of stereotypes. Stereotypes are allowed to participate in associations only with ordinary classes, never with stereotypes (as it is stated in the official UML 2.0 documentation). Stereotypes can be associated to other stereotypes only if there is an association between the two base meta-classes extended by these stereotypes. Stereotypes can have constraints since they indirectly inherit them from the Namespace meta-class. The constraints attached to stereotypes express additional well-formedness rules of the stereotyped base elements.

Profiles in the UML are a subclass of the Package meta-class; they can contain the same elements as ordinary packages and additional elements: stereotypes, as it is presented in Fig. 2.

```
Package

Profile

metaclassReference

metaModelReference

::Constructs::ElementImport

::Constructs::PackageImport

Stereotype

ownedStereotype
```

Fig. 2. Excerpt of the Profile definition in the UML 2.0 specification.

Profiles contain stereotypes and need to have references to the reference meta-model or single meta-model elements. As ordinary packages they can also import elements from other packages (although it is not shown in Fig. 2). A set of constraints can be defined for profiles, since they inherit indirectly from the Namespace meta-class. This kind of constraints is not taken into consideration while defining the metrics in this paper, because they are the same constraints as constraints attached to the standard model elements. These metrics are part of the suite for measuring stereotyped designs, presented in Section 5.

### 3. Stereotype metrics

The metrics presented in this section measure the size of stereotypes; they possess the theoretical properties of size measures according to the Briand et al.´s framework [5]. This suite is targeted for the stereotypes as defined in UML 2.0 specification [1], nevertheless, the metrics still apply to previous version of UML.

The common definitions used throughout the section are:
- $S$ – a stereotype $S = \langle C_s, T_s, B_s \rangle$
- $C_s$ – a set of constraints defined by stereotype $S$
- $T_s$ – a set of tag definitions of stereotype $S$
- $B_s$ – a set of base classes of stereotype $S$.

The definition of the metric suite is based on the set theory. Each subsequent section defines a metric, and presents its purpose.

#### 3.1. Weighted number of constraints

The first metrics of the size and complexity of a single stereotype is the number of constraints which are defined as part of the stereotype definition. Since the constraints are in the form of logical expressions that are evaluated to either true or false, the metrics need to associate the complexity of the expression with each constraint. The complexity should be evaluated in an empirical way, and in the course of the paper, the complexity of the constraint $i$ is denoted as $c_i$ ($c_i \in [0...1]$). The weighted number of constraints attached to a stereotype ($NoSC – number of stereotype constraints$) is given as:

$$NoSC = \sum_{i=1}^{[\cdot]} c_i$$

For the sake of simplicity, if the constraints are equally complex – with respect to some criteria (for example the number of quantifiers in the expression) – then the complexity of each constraint can be assumed to be 1. Further empirical work is needed to define a way of measuring a complexity of logical expressions in OCL.

The large number of constraints indicates that the stereotype is very specific and that it can be used only in a very restricted manner. Such stereotypes are usually used to completely redefine the semantics and usage of the base model element (c.f. [6]). This metric was used in the study presented in [7].

#### 3.2. Weighted number of tag definitions

Each stereotype can have a set of tag definitions (or properties). Each tag definition is a kind of a meta-property of the base element. Its type can be a complex meta-model class or a simple data type. Therefore, with similar assumptions about the complexity of a tag definition, the weighted number of tag definitions ($NoSTD – number of stereotype tag definitions$) can be given as follows:

$$NoSTD = \sum_{i=1}^{[\cdot]} t_i$$

where:
tᵢ – the complexity of the i-th tag definition tᵢ ∈ [0…1],
Since the tag definitions can be seen as additional meta-
attributes of the extended – base meta-model elements, their
complexity is very much different from the complexity of
constraints. Therefore, the tᵢ is advised to be equal to 1,
unless there is a need to distinguish one of the tag
definitions as more complex than others – e.g. because of
its complex type.
The larger the number of tag definitions defined in the
stereotype, the more complex and the larger the stereotype
is. The increasing number of tag definitions in the design
leads to a growing complexity of using the stereotype. The
user needs to understand the meaning of more tag
definitions. This metric was used to indicate the need for
refactoring of stereotypes in our study presented in [8-10].

3.3. Number of base classes the stereotype can be
attached to
Stereotypes are meant to extend the base model elements.
Therefore, for each stereotype, a set of base elements is
defined. It denotes the elements to which the stereotype can
be applied. The metric of the number of base classes
(NoBC) is particularly important from the perspective of
the generality and reusability of the stereotype. It is given
by the formula:

\[ \text{NoBC} = |B| \]

The more entities are affected (higher NoBC), the less
predictable the application is and therefore the more general
the stereotype should be. The large number of base classes
indicates stereotypes which are not well defined for two
reasons:
• they extend meta-classes which are too specific
(another, more abstract meta-class should be chosen
instead),
• the constraints defined with the stereotype might be
evaluated in all valid contexts as there might be
differences which meta-attributes are valid for a given
application of the stereotype.

4. Profile metrics
The set of metrics presented in this section allows
assessment of the size, complexity, and cohesion of a
profile. These metrics have been used in our studies
presented in [7-11]. Using these metrics resulted in
refactoring of profiles and comparing quality of profiles
after their normalization w.r.t. size.

4.1. Common notation
The following elements are used in the subsequent
definitions:
• P – a profile P = <PEₚ, Iₚ, Tdₚ>
• PEₚ – a set of elements in profile P
• PSₚ – a set of stereotypes defined in profile¹ P
• Iₚ – a set of packages and elements imported by
profile P

4.2. Weighted number of defined stereotypes
The main measure of the size of the profile is the number
of stereotypes (NoPS) defined in the profile. It is given
by the formula:

\[ \text{NoPS} = \sum_{i=1}^{PS} s_i \]

where:
\[ s_i \] – the complexity of the i-th stereotype.

The higher number of stereotypes in the profile
indicates the higher complexity of it. The profiles with
large number of stereotypes have been observed to be
used for heavy-weight language customization and
indicate that the profile needs to be evaluated by a
language engineer (c.f. [6, 7]).

4.3. Number of profile imports
The number of imported profiles, packages, and elements
provides a measurement of the reusability level of the
profile. The profile which is self-standing is more
reusable. The metric is defined as follows:

\[ \text{NoPI} = |I| \]

The high number of imports indicates that the profile is
less reusable and it is more volatile to changes in the
imported packages and profiles.

4.4. Number of other elements in the profiles
In case of heterogeneous profiles [12] – that is profiles
which contain both the model level elements (for
example data types) and meta-model level elements (for
example stereotypes) – the set of metrics should
incorporate also the number of elements that are not on
the meta-model level. The metric is given by the
following expression:

\[ \text{NoPO} = \| e \| e \in P \wedge e \notin PS \| \]

If the number of the elements is significantly larger than
the number of stereotypes, then perhaps this profile
should be split into a model library (a package) and a
profile in a strict sense.

4.5. Profile cohesion
The ratio of the number of stereotypes over the number of
all profile elements can provide a means of assessing
whether the profile is for language extension or more like
a model library [12]. The ratio, called Profile Cohesion
(\(PCoh\)), is given by the following formula:

\[ PCoh = \frac{|PS|}{|PE|} \]
The most specific profiles, the so-called “strict profiles” [13], are used in extending the language capabilities. Less specific profiles are often used as model libraries, which contain reusable modeling elements, and not elements dedicated for instantiation in user models (c.f. [14]).

### 4.6. Model library cohesion

In a similar way to the profile cohesion, a related metric can be defined – Model Library Cohesion (MCoh). The metric is defined as follows:

\[
MCoh = \begin{cases} 
\frac{NoPO}{NoPS + NoPO} & \text{iff } NoPO + NoPS > 0 \\
0 & \text{iff } NoPO + NoPS = 0
\end{cases}
\]

Profiles with the high value of MCoh should be split into two things: a profile (with stereotype) and a model library (with classes). The profile should import the model library, which makes the types from the library available in user models after applying the profile.

### 5. Measuring the designs with stereotypes

None of the existing and well-established object-oriented metrics take into account the existence of stereotypes. Therefore, some atomic metrics need to be defined for designs where stereotypes are used. The set of metrics is to be seen as an add-in to the existing metrics suites for UML models.

Common definitions:
- ME – a given model element
- \( S_{ME} \) – a set of stereotypes applied to ME
- \( C_{ME} \) – a set of constraints defined for ME
- \( D \) – the set of all elements in the model

#### 5.1. Number of stereotypes attached to an element

To obtain information about how complex a model element is, the metrics of a number of stereotypes applied to model elements is given. The measurement is given by a formula:

\[ NoAS = |S_{ME}| \]

The value of the formula is the number of elements in the set of attached stereotypes.

#### 5.2. Number of applications of stereotypes

In other words: how many times was a single stereotype applied in this model.

\[ NoSA(s) = |\{e|e \in D \wedge S_{el} \cup s \neq \emptyset\}| \]

The metric is the number of elements in the set which contains stereotyped elements in the design.

### 5.3. Design specificity

Another important aspect is the ratio of stereotyped model elements to the non stereotyped model elements (Spec – specificity of the model). The metrics provides information of how specific the design is. If the ratio is close to 1, it means that the model is very specific and the meaning of the model is domain-specific, where the domain is defined by the profile. Therefore, perhaps the standard model manipulation methods should not be used for this model.

\[ Spec = \frac{|\{e|e \in D \wedge S_{el} \cup s \neq \emptyset\}|}{|\{e|e \in D \wedge S_{el} = \emptyset\}|} \]

For the designs that are intended to be used for specific purposes and thus with the profiles applied, the value of Spec should be as close to 1 as possible. It means that each element in the design is stereotyped and that each has specific semantics in the domain for which the element is supposed to be used.

### 5.4. Number of stereotypes per element

For the overall design we need to calculate an overall metric of the density of stereotypes for the design. The metric is defined as follows:

\[ NoSpEL = \frac{\sum_{i=0}^{\|D\|} NoAS_i}{|D|} \]

where:
- \( NoAS_i \) – the value of NoAS metrics for element \( i \)

### 6. Relationship with the existing metrics suites

There are several existing metric suites for UML designs measuring the structural properties of the UML models. The different metric suites were investigated whether some of the metrics can be applied to stereotypes and profiles. The results of the investigation are presented in Table 1. The plus sign in the table cell indicates that the measurement could be used as it is defined, whereas the minus indicates that it cannot be used as it is defined. If a metric cannot be used, it can be caused by the fact that stereotypes do not have elements required for the calculation of the metric (for example operations) or it can be caused by the fact that the metrics does not give meaningful information about the measured stereotype. In order to compact the results stereotypes and profiles...
are presented in the same table although none of the metrics could apply to both.

<table>
<thead>
<tr>
<th>Suite</th>
<th>Metric</th>
<th>Ster.</th>
<th>Prof.</th>
</tr>
</thead>
<tbody>
<tr>
<td>C&amp;K [15]</td>
<td>Weighted Methods per Class (WMC)</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Depth of Inheritance Tree (DIT)</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Number of Children (NOC)</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Response for Class (RFC)</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Lack of Cohesion of Methods (LCOM)</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Coupling Between Obj. (CBO)</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>OOA metrics [16]</td>
<td>Number of Responsibilities for Class (CL1)</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Number of Dependencies (CL2)</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Depth of Inheritance Tree (CL3)</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Number of children (CL4)</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Number of distinct classes depending on the class (CL5)</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Extent of usage of classes from outside the package (PK1)</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td></td>
<td>Extent of usage of classes from the package in other packages (PK2)</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td>MOOD [17]</td>
<td>Method Hiding Factor (MHF)</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Attribute Hiding Factor (AHF)</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Attr. Inheritance Factor (AIF)</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Method Inheritance (MIF)</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Coupling Factor (COF)</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Polymorphism Factor (PF)</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Reuse Factor (RF)</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 1. Applicability of other metrics for stereotypes

It should be noted that the values of the metrics in Table 1, if applied to stereotypes/profiles, are usually different than typical values for these metrics when considered for classes/packages.

7. Conclusions

In the paper, three metric suites are presented – a suite for measuring stereotypes, a suite for measuring profiles, and a suite for measuring stereotyped designs. We use these metric suites in order to baseline profiles and stereotypes with respect to size and complexity. We also use the suites for assessing how specific a given model is – how many elements are stereotyped.

In the course of our research on stereotypes and profiles we found that the existing metric suites for object-oriented designs have limited applicability for measuring stereotyped designs and profiles/stereotypes. Since stereotypes and profiles are elements from the domain of the modeling language, and not from the domain of the modeled system, they need to have dedicated metric suites. The presented metrics helped us to use software engineering methods to work with profiles and stereotypes.

Our current efforts are directed towards validating the metrics empirically in a context of a controlled experiment.

References