# A Qualitative Investigation of UML Modeling Conventions

Bart Du Bois[1], Christian F.J. Lange[2],
Serge Demeyer[1], and Michel R.V. Chaudron[2]

[1] Lab On REengineering, University of Antwerp, Belgium
{Bart.DuBois,Serge.Demeyer}@ua.ac.be
[2] Dept. of Mathematics and Computer Science, Technische Universiteit Eindhoven
{C.F.J.Lange,M.R.V.Chaudron}@tue.nl

**Abstract.** Analogue to the more familiar notion of coding conventions, modeling conventions attempt to ensure uniformity and prevent common modeling defects. While it has been shown that modeling conventions can decrease defect density, it is currently unclear whether this decreased defect density results in higher model quality, i.e., whether models created with modeling conventions exhibit higher fitness for purpose.

In a controlled experiment[1] with 27 master-level computer science students, we evaluated quality differences between UML analysis and design models created with and without modeling conventions. We were unable to discern significant differences w.r.t. the clarity, completeness and validity of the information the model is meant to represent.

We interpret our findings as an indication that modeling conventions should guide the analyst in identifying what information to model, as well as how to model it, lest their effectiveness be limited to optimizing merely syntactic quality.

## 1 Introduction

In [1], a classification of common defects in UML analysis and design models is discussed. These defects often remain undetected and cause misinterpretations by the reader. To prevent these defects, *modeling conventions* have been composed that, similar to the concept of code conventions, ensure a uniform manner of modeling [2]. We designed a pair of experiments to validate the effectiveness of using such modeling conventions, focusing on their effectiveness w.r.t. respectively (i) defect prevention; and (ii) model quality. We reported on the prevention of defects in [3]. Our study of the effect of modeling conventions on model quality forms the subject of this paper.

In the first experiment, we evaluated how the use of modeling conventions for preventing modeling defects affected defect density and modeling effort [3]. These modeling conventions are enlisted in Appendix A, and have been discussed previously in [1]. This set of 23 conventions has been composed through a literature

[1] A replication package is provided at http://www.lore.ua.ac.be/Research/Artefacts

T. Kühne (Ed.): MoDELS 2006 Workshops, LNCS 4364, pp. 91–100, 2007.

review and through observations from industrial case studies, and concern abstraction, balance, completeness, consistency, design, layout and naming. These conventions are *formative*, in that they focus on specifying *how* information should be modeled, rather than specifying *what* should be modeled.

Our observations on 35 three person modeling teams demonstrated that, while the use of these modeling conventions required more modeling effort, the defect density of resulting UML models was reduced. However, this defect density reduction was not statistically significant, meaning that there is a (small) possibility, albeit small, that the observed differences might be due to chance.

This paper reports on the second experiment, observing differences in representational quality between the models created in the first experiment. We define *representational quality* of a model as the clarity, completeness and validity of the information the model is meant to represent. Typical flaws in representational quality are information loss, misinformation, and ambiguity or susceptibility to misinterpretation. This study investigates whether models created using common modeling conventions exhibit higher representational quality.

The paper is structured as follows. The selected quality framework is elaborated in section 2. The set-up of the experiment is explained in section 3, and the analysis of the resulting data is discussed and interpreted in section 4. We analyze the threats to validity in section 5. Finally, we conclude in section 6.

For space considerations, the description of the experiment has been reduced to its essence. A more elaborate discussion of the experiment is provided in [4].

## 2 Evaluating Model Quality

Through a literature review of quality models for conceptual models, we selected Lindland's framework for its focus on clarity, completeness and validity. Lindland's framework relates different aspects of modeling to three linguistic concepts: syntax, semantics and pragmatics [5]. These concepts are described as follows (citing from [5]):

**Syntax** *relates the model to the modeling language by describing relations among language constructs without considering their meaning.*

**Semantics** *relates the model to the domain by considering not only syntax, but also relations among statements and their meaning.*

**Pragmatics** *relates the model to the audience's interpretation by considering not only syntax and semantics, but also how the audience (anyone involved in modeling) will interpret them.*

These descriptions of the concepts of syntax, semantics and pragmatics refer to relationships. The evaluation of these relationships gives rise to the notion of syntactic, semantic and pragmatic quality. We note that the effect of UML modeling conventions on syntactic quality has been the target of our previous experiment [3], and is therefore not included in this study.

In [6], Lindland's quality framework is extended to express one additional quality attribute. *Social quality* evaluates the relationship among the audience interpretation, i.e. to which extent the audience agrees or disagrees on the statements within the model.

With regard to representational quality, we are less interested in the relationship between the model and the audience's interpretation – indicated by pragmatic quality – than in the relationship between the domain and the audience's interpretation, as the former is unrelated to the information the model is meant to represent. Accordingly, we will not observe pragmatic quality, but instead introduce an additional quality attribute, *communicative quality*, that targets the evaluation of the audience's interpretation of the domain.

### 2.1 Measuring Model Quality

Lindland's quality framework evaluates the relationships between model, modeling domain and interpretation using the elementary notion of a statement. A *statement* is a *sentence representing one property of a certain phenomenon* [6]. Statements are extracted from a canonical form representation of the language, which in UML, is specific to each diagram type. An example of a statement in a use case diagram is the capability of an actor to employ a feature.

The set of statements that are relevant and valid in the domain are noted as $D$, the set of statements that are explicit in the model as $M_E$, and the set of statements in the interpretation of an interpreter $i$ are symbolized with $I_i$. We say that a statement is *explicit* in case it can be confirmed from that sentence without the use of inference. Using these three sets, indicators for semantic quality (and also pragmatic quality, that we do not include in this study) have been defined that are similar to the concepts of recall and precision:

**Semantic Completeness** (SC) is the ratio of the number of modeled domain statements $|M_E \cap D|$ and the total number of domain statements $|D|$.

**Semantic Validity** (SV) is the ratio of the number of modeled domain statements $|M_E \cap D|$ and the total number of model statements $|M_E|$.

Krogstie extended Lindland's quality framework through the definition of *social quality* [6]. The single proposed metric of social quality is:

**Relative Agreement among Interpreters** (RAI) is calculated as the number of statements in the intersection between the statements in the interpretations of all $n$ interpreters $|\bigcap_{\forall i,j \in [1,n]} I_i \cap I_j|$.

Similar to semantic quality, we introduce the following metrics for communicative quality:

**Communicative Completeness** (CC) is the ratio of the number of recognized modeled domain statements $|I_i \cap M_E \cap D|$ and the total number of modeled domain statements $|M_E \cap D|$.

**Communicative Validity** (CV) is the ratio of the number of recognized modeled domain statements $|I_i \cap M_E \cap D|$ and the total number of statements in the interpretation of interpreter $i$ $|I_i|$.

Communicative completeness and validity respectively quantify the extent to which information has been lost or added during modeling.

The difficulty in applying the metrics for semantic, social and communicative quality mentioned above lies in the identification of the set of model statements ($M_E$), and interpretation statements ($I_i$). In contrast, the set of domain statements ($D$) is uniquely defined and can reasonably be expected to have a considerable intersection with the set of model and interpretation statements. Accordingly, we choose to estimate the sets of domain statements, model statements and interpretation statements, by verifying their intersection with a *selected* set of domain statements ($D_s$).

Semantic validity cannot be approximated in this manner, as it requires an estimate of the set of statements that lie outside the set of domain statements ($|M_E \setminus D|$). Nonetheless, the resulting set of estimates for semantic, social and communicative quality allows to assess typical representational quality flaws as information loss (semantic and communicative completeness estimates), misinformation (communicative validity estimate) and misinterpretation (social quality estimate).

## 3 Experimental Set-Up

Using the classical Goal-Question-Metric template, we describe the purpose of this study as follows: **Analyze** UML models **for the purpose of** evaluation of modeling conventions effectiveness **with respect to** the representational quality of the resulting model **from the perspective of** the analyst/designer **in the context of** master-level computer science student.

Using our refinement of representational model quality presented in the previous section, we define the following null hypotheses:

$H_{0,SeQ}$ – UML analysis and design models composed with or without modeling conventions do not differ w.r.t. *semantic quality.*

$H_{0,SoQ}$ – UML analysis and design models composed with or without modeling conventions do not differ w.r.t. *social quality.*

$H_{0,CoQ}$ – UML analysis and design models composed with or without modeling conventions do not differ w.r.t. *communicative quality.*

### 3.1 Experimental Design

In this study, we use a three-group posttest-only randomized experiment, consisting of a single control group and two treatment groups:

**noMC – no modeling conventions.** This group of subjects, referred to as the *control group* were given UML analysis and design models that were composed *without* modeling conventions.

**MC – modeling conventions.** The subjects in this treatment group received UML analysis and design models that were composed using the list of modeling conventions enlisted in Appendix A.

**MC+T – tool-supported modeling conventions.** Subjects in this treatment group received UML analysis and design models that were composed using both a list of modeling conventions and a tool to support the detection of their violation.

### 3.2 Experimental Subjects, Tasks and Objects

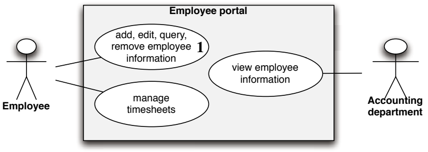
The experiment was performed using pen and paper only. Each student was provided with (i) a hardcopy of all diagrams of a single model; (ii) a questionnaire; and (iii) a vocabulary.

A total of 27 MSc computer-science students participated in the controlled experiment. This experiment was performed in the end of 2005 at the University of Mons-Hainaut and at the University of Antwerp (both in Belgium). We evaluated the subjects' experience with the different types of UML diagrams using a questionnaire. All subjects had practical (although merely academic) experience with the diagrams required to answer the questions.

The questionnaire contained a single introduction page that described the task. Another explanatory page displayed one example question and its solution, elaborating on the steps to be applied. The example question, illustrated in Table 1, asks the participant to verify whether a given UML analysis and design model confirms a given statement. As an argument for the confirmation of a statement, the participant should be able to indicate a diagram fragment dictating that the statement should hold. In case such a fragment can be found, the participant annotates the fragment with the question number.

**Table 1.** Example question and supporting diagram fragment

| Nr | Statement | Confirmed | Not Confirmed |
|---|---|---|---|
| 1 | The software system should support querying employee information. | O | O |


Employee portal
add, edit, query, remove employee information
1
view employee information
manage timesheets
Employee
Accounting department

The main part of the questionnaire asked subjects to evaluate whether a given statement was explicitly confirmed by the given model. Only two options were possible, being either "confirmed", or "not confirmed". The questions[2] asked allow to estimate semantic, social and communicative quality. We have identified over 60 statements that are relevant and valid in the domain, derived from the informal requirement specification for which the subjects of the first experiment composed the UML models. From this set of 60 statements, a selection of 22 statements was made, comprising the set of selected domain statements $D_s$.

For each experimental group ($noMC$, $MC$, $MC+T$), a representative set of three UML analysis and design models was selected from the set of output models

[2] An elaborate discussion on the different categories of questions is provided in [4].

of the first experiment. The selected models serve as experimental objects, and were representative w.r.t. syntactic quality, defined as the density of modeling defects present in the model. These UML models – modeling a typical application in the insurance domain – consisted of six different types of UML diagrams used for analysis and design. The frequency of each of the diagram types in each model is provided in Table 2.

**Table 2.** Frequency of the diagram types in each model

| | **noMC** | | | **MC** | | | **MC+T** | | |
|---|---|---|---|---|---|---|---|---|---|
| **type** | $no_2$ | $no_4$ | $no_8$ | $MC_2$ | $MC_4$ | $MC_5$ | $MC+T_4$ | $MC+T_6$ | $MC+T_{10}$ |
| Class Diagram | 6 | 1 | 6 | 8 | 1 | 1 | 11 | 1 | 5 |
| Package Diagram | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| Collaboration Diagram | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| Deployment Diagram | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| Use Case Diagram | 7 | 1 | 5 | 0 | 3 | 5 | 6 | 5 | 1 |
| Sequence Diagram | 6 | 26 | 10 | 3 | 39 | 14 | 8 | 56 | 15 |
| total | 20 | 28 | 16 | 11 | 43 | 20 | 26 | 23 | 64 |

As the different models used synonyms for some concepts, a glossary was provided indicating which names or verbs are synonyms.

### 3.3 Experimental Procedure

The procedure for this experiment consisted of two major phases. First, in preparation of the experiment, the semantic quality of each selected model was assessed. Second, two executions of the experimental procedure (runs) were held to observe subjects performing the experimental task described in the previous subsection, thereby assessing the models' communicative and social quality.

*Assessment of semantic quality.* This assessment was performed by three evaluators, and did not require the participation of experimental subjects. The three evaluators were the first two authors of this paper, and a colleague from the first authors' research lab. After an individual assessment, conflicts were resolved resulting in agreement on the recognition of each selected domain statement in each model.

This evaluation procedure provided the data to calculate the semantic completeness and semantic validity of each of the nine selected models.

*Assessment of social and communicative quality.* Each experimental run was held in a classroom, and adhered to the following procedure. Subjects were first randomized into experimental groups, and then provided with the experimental material. Subjects were asked to write their name on the material, to take the time to read the instructions written on an introduction page, and finally to complete the three parts of the questionnaire.

No time restrictions were placed on the completion of the assignment.

### 3.4 Experimental Variables

The independent variable subject to experimental control is entitled *modeling convention usage*, indicating whether the model was composed without modeling conventions (*noMC*), with modeling conventions (*MC*) or with modeling conventions and a tool to detect their violations (*MC+T*). The observed dependent variables are the estimators for semantic completeness (SC), communicative completeness (CC), communicative validity (CV) and relative agreement among interpreters (RAI), as defined in section 2.1. As these variables are all calculated as ratios, we express them in percentage.

## 4 Data Analysis

Table 3 characterizes the experimental variables across the experimental groups.

**Table 3.** Statistics of the experimental variables

| | | **Overall** | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Hyp.** | **DV** | **mean** | **MCU**[1] | **Mean** | **StdDev** | **Min** | **Max** | $H(2)$ | **p-value** |
| $H_{0,SeQ}$ | SC | 62.6% | noMC | 66.7% | 13.9% | 54.5% | 81.8% | 0.4786 | .7872 |
| | | | MC | 59.1% | 9.1% | 50.0% | 68.2% | | |
| | | | MC+T | 62.1% | 6.9% | 54.5% | 68.2% | | |
| $H_{0,SoQ}$ | RAI | 59.6% | noMC | 66.7% | 15.6% | 50.0% | 81.8% | 1.1556 | .5611 |
| | | | MC | 59.1% | 20.8% | 36.4% | 77.3% | | |
| | | | MC+T | 53.0% | 17.2% | 40.1% | 72.7% | | |
| $H_{0,CoQ}$ | CC | 76.9% | noMC | 82.7% | 14.1% | 61.0% | 100.0% | 2.7298 | .2554 |
| | | | MC | 74.5% | 16.0% | 36.0% | 93.0% | | |
| | | | MC+T | 72.5% | 13.1% | 53.0% | 92.0% | | |
| | CV | 85.0% | noMC | 87.0% | 7.9% | 75.0% | 100.0% | 1.5235 | .4668 |
| | | | MC | 85.9% | 10.6% | 60.0% | 100.0% | | |
| | | | MC+T | 81.5% | 8.9% | 69.0% | 92.0% | | |

[1] **M**odeling **C**onvention **U**sage.

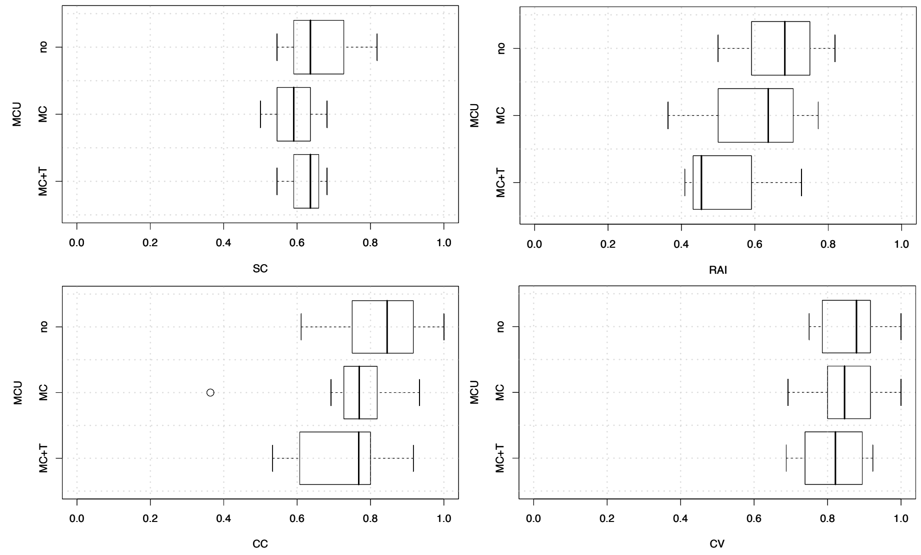
**Semantic Completeness (SC)** – The semantic completeness of models composed without modeling conventions was somewhat higher, within a margin of 10% (see top left figure in Table 4). I.e., the models from group *noMC* described slightly more modeling domain statements. However, the *noMC* group also exhibits a larger standard deviation.

**Relative Agreement among Interpreters (RAI)** – There was considerable higher (about 14%) agreement among interpreters of the models composed without modeling conventions (see top right figure in Table 4). However,we also observed considerable standard deviations in Table 3 in all treatment groups.

**Communicative Completeness (CC)** – The communicative completeness of models composed without modeling conventions was somewhat higher (around 10%) than that of models composed with modeling conventions.

**Communicative Validity (CV)** – The communicative validity is approximately equal between models composed with and without modeling conventions, as illustrated in in the bottom right figure in Table 4).

**Table 4.** Variation of SC, RAI, CC and CV across experimental groups



To verify whether the differences among experimental groups are statistically significant, Kruskal-Wallis test results are appended to Table 3. This test is a non-parametric variant of the typical Analysis of Variance (ANOVA), and is more robust with regard to assumptions about the distribution of the data, as well as unequal sample sizes ($\#noMC$=10,$\#MC$=9,$\#MC+T$=8). Moreover, the assumptions of at least an ordinal measurement level, independent groups and random sampling were also satisfied.

Table 3 indicates that the group differences concerning semantic, social and communicative quality are not statistically significant at the 90% level. Accordingly, we must accept the hypotheses stating that the UML analysis and design models composed with or without modeling conventions do not differ w.r.t. semantic, social and communicative quality.

## 5 Threats to Validity

*Construct Validity* is the degree to which the variables used measure the concepts they are to measure. We have decomposed representational quality, the main concept to be measured, into semantic, social and communicative quality, and have argued their proposed approximations.

*Internal Validity* is the degree to which the experimental setup allows to accurately attribute an observation to specific cause rather than alternative causes.

Particular threats are due to selection bias. The selection of statements from the domain $D_s$ could not have introduced systematic differences, and the selection of model was performed as to be representative w.r.t. syntactic quality.

*External Validity* is the degree to which research results can be generalized outside the experimental setting or to the population under study. The set of modeling conventions was composed after a literature review of modeling conventions for UML, revealing design, syntax and diagram conventions. Our set of modeling conventions contains instances of these three categories.

## 6 Conclusion

Based on the results of this experiment, we conclude that UML modeling conventions focusing on the prevention of common UML modeling defects (as reported in [1]) are unlikely to affect representational quality.

We interpret our findings as an invitation to study the application of modeling conventions of a different nature. Conventions are needed that clarify which types of information are relevant to particular future model usages. Such modeling conventions might suggest the modeling of a type of information (e.g., features, concepts, interactions, scenarios) consistently in a particular (set of) diagram type(s). We hypothesize that this uniform manner of modeling different types of information is more likely to optimize semantic and communicative quality, as these types of information are the subject of their evaluation.

## References

[1] C.F.J. Lange and M.R.V. Chaudron. Effects of defects in UML models - an experimental investigation. In *ICSE '06: Proceedings of the 28th International Conference on Software Engineering*, pages 401–411, 2006.

[2] C.F.J. Lange, M.R.V. Chaudron, and Johan Muskens. In practice: UML software architecture and design description. *IEEE Softw.*, 23(2):40–46, 2006.

[3] C.F.J. Lange, Bart Du Bois, M.R.V. Chaudron, and Serge Demeyer. Experimentally investigating the effectiveness and effort of modeling conventions for the UML. In *O. Nierstrasz et al. (Eds.): MoDELS 2006, LNCS 4199*, pages 27–41, 2006.

[4] Bart Du Bois, C.F.J. Lange, Serge Demeyer and M.R.V. Chaudron. A Qualitative Investigation of UML Modeling Conventions *First International Worskshop on Quality in Modeling* at MoDELS 2006

[5] Odd Ivar Lindland, Guttorm Sindre, and Arne Solvberg. Understanding quality in conceptual modeling. *IEEE Softw.*, 11(2):42–49, 1994.

[6] John Krogstie. *Conceptual Modeling for Computerized Information Systems Support in Organizations.* PhD thesis, University of Trondheim, Norway, 1995.

[7] Friday, November 10, 2006 at 5:06 pmWilliam R. Shadish, Thomas D. Cook, and Donald T. Campbell. *Experimental and Quasi-Experimental Designs for Generalized Causal Inference.* Houghton Mifflin, 2002.

## A Modeling Conventions

Table 5 enlists the modeling conventions employed in a previous experiment. These conventions were used by two of the experimental groups (*MC* and

$MC + T$) while composing UML analysis and design models. As the resulting models were used in this experiment, it is relevant to recapitulate these conventions.

**Table 5.** Modeling Conventions

| Category | ID | Convention |
|---|---|---|
| Abstraction | 1 | Classes in the same package must be of the same abstraction level. |
| | 2 | Classes, packages and use cases must have unique names. |
| | 3 | All use cases should cover a similar amount of functionality. |
| Balance | 4 | When you specify getters/setters/constructors for a class, specify them for all classes. |
| | 5 | When you specify visibility somewhere, specify it everywhere. |
| | 6 | Specify methods for the classes that have methods! Don't make a difference in whether you specify or don't specify methods as long as there is not a strong difference between the classes. |
| | 7 | Idem as 6 but for attributes. |
| Completeness | 8 | For classes with a complex internal behavior, specify the internal behavior using a state diagram. |
| | 9 | All classes that interact with other classes should be described in a sequence diagram. |
| | 10 | Each use case must be described by at least one sequence diagram. |
| | 11 | The type of ClassifierRoles (Objects) must be specified. |
| | 12 | A method that is relevant for interaction between classes should be called in a sequence diagram to describe how it is used for interaction. |
| | 13 | ClassifierRoles (Objects) should have a role name. |
| Consistency | 14 | Each message must correspond to a method (operation). |
| Design | 15 | Abstract classes should not be leafs. |
| | 16 | Inheritance trees should not have no more than 7 levels. |
| | 17 | Abstract classes should not have concrete superclasses. |
| | 18 | Classes should have high cohesion. Don't overload classes with unrelated functionality. |
| | 19 | Your classes should have low coupling. |
| Layout | 20 | Diagrams should not contain crossed lines (relations). |
| | 21 | Don't overload diagrams. Each diagram should focus on a specific concept/problem/functionality/... |
| Naming | 22 | Classes, use cases, operations, attributes, packages, etc. must have a name. |
| | 23 | Naming should use commonly accepted terminology, be non-ambiguous and precisely express the function/role/characteristic of an element. |