On the impact of layout quality to understanding UML diagrams

Störrle, Harald

Published in:
2011 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)

Link to article, DOI:
10.1109/VLHCC.2011.6070390

Publication date:
2011

On the Impact of Layout Quality to Understanding UML Diagrams

Harald Störrle
Department of Informatics and Mathematical Modeling,
Technical University of Denmark
Richard Petersens Plads, 2800 Lyngby, Denmark
hsto@imm.dtu.dk

***Abstract*—Practical experience suggests that use and understanding of UML diagrams is greatly affected by the quality of their layout. However, existing experimental evidence for this effect is been weak and inconclusive. In this paper, we explore two explanations. Firstly, we observe that the visual qualities of diagrams are more prominent in earlier life cycle phases so that the impact of layout quality should be more apparent in models and diagram types used there, an aspect not studied in previous research. Secondly, in practice, good layouts use many different heuristics simultaneously whereas previous research considered them in isolation only. In this paper, we report the results of a series of controlled experiments using compound layouts on requirements analysis models. With very high significance, we find a notable impact of the layout quality measured by different aspects of cognitive load.**

I. Introduction

The Unified Modeling Language (UML) has been the "*lingua franca of software engineering*" for a long time now. It is a generally help belief that visual languages are somehow superior to textual languages ("a picture says more than a 1000 words"), and that this is also true for the UML. In fact, many people connect the success of UML with the fact that it is primarily visual. However, there are actually few research results to support this belief. There *is* a large body of experimental results on the layout of UML class diagram and how it affects human understanding and problem solving, but the findings are ambiguous, and sometimes unintuitive. In particular, only very small effects have been found in vitro. For instance, Eichelberger and Schmid note that "*We could not identify [...] a significant impact [by diagram quality].*" (cf. [10, p. 1696]). On the other hand, practical experience in industrial software projects suggests a much higher impact of good or bad layout. We offer two explanations for this.

Firstly, different parts of UML are used during different phases of the software development life cycle. Dobing & Parsons [4] show that the two most commonly used UML diagram types used for technical purposes (i.e., late life cycle phases) are class and sequence diagrams, while the two most commonly used digram types for requirements analysis (i.e., earlier life cycle phases) are use case and activity diagrams. However, previous layout research has very much focused on class diagrams. Based on our own industrial experience, we hypothesize that inter-personal communication is more prominent and extensive in earlier life cycle phases. Therefore, the quality of a diagram layout should have a larger impact for the model types used there, and thus we should *expect* to see less impact of diagram layout in previous work which did not focus on such models and diagrams. Instead, researchers have previously focused on class diagrams which are more often used for technical tasks of later life cycle phases (cf. [4]). Therefore, in this paper, we study models created in requirements analysis projects. In particular, we study the use case and activity diagrams most important for that purpose. We also study analysis-level class diagrams as a benchmark.

Secondly, previous work has been preoccupied with creating results suitable to feed into the development of diagram layout algorithms. Thus, empirical research has identified the various individual quality criteria, formally defined them as layout quality metrics, and studied them in isolation. Such criteria may be the number of line crossings and bends, the joining and placing of arcs of certain types, and so on. This is doubtlessly a great contribution to creating better automatic layouts, but such knowledge is no effective help for human modelers trying to create "good" diagrams. In the workplace, modelers are quite content (and effective) to apply informal and even vague guidelines, when instructed to do so. Therefore, in a "good" realistic diagram, many layout heuristics are applied *simultaneously* in a more or less consistent way. On the other hand, in a "bad" realistic diagram, we will find little or no care for diagram layout, resulting in close to random scattering of notational elements on the diagram pane. Examples of this latter type of "layout" are readily found in the diagrams of novice modelers (e.g., freshmen) as well as in articles published in academic and industrial publications by respected professionals. Since we are more interested in the practice of diagram layout than in the development of algorithms, in this paper, we study the compound effect of many heuristics applied together to create "good" diagrams. Using a series of controlled experiments with 77 subjects, we find a notable impact of the layout quality measured by different aspects of cognitive load. Our results are highly significant.

II. Related Work

The layout of graphs (in the mathematical sense) has been a longstanding research challenge, both with respect to auto-

matic layout and to various aspects of usability, e.g., diagram comprehension, user preferences, and diagramatic inference. Based on the rich knowledge on general graphs, research on the layout of UML has started with those of UML's notations that are closest to graphs, namely, class diagrams (cf. [24], [8], [11], [31], [18]), and, to a lesser extent, communication diagrams (see e.g. [17], [21] who use UML 1 terminology). Other types of UML diagrams, in contrast, have only attracted little interest so far (e.g. use case diagrams [9], or sequence diagrams (cf. [1], [30]). While there is some work on the Business Process Modeling Notation (BPMN, see [6]), there seems to be no empirical work whatsoever on UML activity diagrams. Arguably, however, the two notations are similar enough to transfer insights from one to the other.

Research on UML class diagrams has mostly focused on the isolated impact of individual and minor layout criteria such as line bends, crossings, and length. Unsurprisingly, each of these individual criteria has little or no impact. The more elusive higher levels like applying layout patterns, respecting the diagram flow, and the correspondence between the diagram and the message it is supposed to convey seem to have not yet been studied empirically at all.

The main focus of previous work on UML diagram types and their layout has been with one of four aspects: diagram comprehension (cf. [26], [27], [20], [21] and/or user preference (cf. [18], [29]), automatic layout (cf. [8], [11], [16], [9], [5]), or one of a variety of diagram inference tasks, e.g., program understanding based on visualizations (cf. [30]), or the role of design patterns in understanding (cf. [27], [28]).

Most research uses controlled experiments for their research and evaluate user performance using paper questionnaires, or online surveys. Only a few contributions have used other methods, most notably eye tracking (see [2], [31], [27]). After using both methods for essentially the same experiment, Sharif et al. have concluded that these two methods are mostly complementary wrt. comprehension tasks (cf. [25]). Thus, eye tracking is only favorable for a tightly restricted set of research questions, in particular when taking into account the considerable cost and effort involved. Having said that, most questionnaire-based approaches employ only very few subjects in their experiments, typically in the range of 15 to 30, with the notable exceptions of [26], [19] and [1] involving 45, 55 and 78 subjects, respectively.

III. "Good" layout of UML diagrams

In this section, we will briefly review the knowledge on aesthetic criteria for the layout of UML diagrams. A detailed discussion of aesthetic criteria for class diagrams is found in [8, p. 54–65], a recent survey of empirical results on layout criteria is found in [10]. Wong and Sun [30] provide an overview of these criteria from a cognitive psychology point of view, along with an evaluation of how well these principles are realized in several UML CASE tools. Purchase et al. discuss aesthetic criteria with a view to the layout of UML class and communication diagrams (cf. [18], [17]) and also provide sources to justify and explain these criteria (cf. [20]). Eichelberger [7] also discusses these criteria at length, and shows how they can be used in the automatic layout of UML class diagrams.

The layout of UML diagrams is governed by four levels of design principles. First of all, there are the general principles of graphical design and visualization that apply to all kinds of diagrams, and probably any kind of visualization. For instance, in a good layout, elements should not obscure each other, the Gestalt principles should be respected, text should be shown in a readable size, elements should be aligned (e.g., on a grid), and there should be sparing and careful use of colors, and different fonts or styles. The "Physics of Notation" could be used to organize these factors (cf. [13]).

Second, there are layout principles applying to all structures that can be considered as a graph, mathematically speaking. Thus, good layouts should avoid or minimize crossings, bends, and length of lines. Most of the empirical research on UML diagrams focuses on principles from this level, e.g., [24], [8], [11], [31], [18].

Third, there are layout principles that apply mostly only to the notations like those found in UML. For instance, diagrams with some inherent ordering of elements should maintain and highlight that ordering as visual flow. Visual clutter should be reduced by introducing symmetry when possible. For instance, similar edges should be joined, similar elements should be aligned and grouped, and so on. In UML, this means that if a class has several subclasses, it might be helpful to group and align the subclasses and join the arcs indicating the inheritance-relationship. Another application is found in activity diagrams, where several consequences of a decision could be aligned and grouped.

Fourth, there is the level of pragmatics, that is, support for underlining the purpose of a diagram in order to better address the audience. Items may be highlighted by color, size, or position to guide and direct the attention of spectators. On this level, rules and guidelines from lower levels may be put aside to better serve the paramount purpose of conveying the message and telling whatever story the diagram designer intends to tell.

In order to develop algorithms for creating automatic layouts that are perceived as being helpful (or "good") by human modelers, detailed knowledge about the individual criteria, their relative and absolute impact, and their formalization is needed. So, it is not surprising that most of the empirical research on UML diagrams has so far focused on studying individual principles, with an emphasis on the second group (cf. [24], [8], [11], [31], [18]). For instance, work by Purchase et al. has shown that there are many such criteria with varying degrees of impact (see [18], [17], [20]), though all of them seem have a rather small impact with findings that are not or not highly statistically significant. Also, the ranking and contribution of these criteria may vary across different diagram types. Even between class and communication diagrams, which are rather close relatives as far as concrete syntax is concerned, [18, pp. 246] shows notable differences in the ordering and impact of layout criteria. Thus, other notations that share even less

commonalities with class diagrams (e.g., activity, use case, or sequence diagrams) may need a completely different set of criteria.

For humans creating diagram layouts, on the other hand, a set of comparatively vague guidelines together with some instruction is often good enough for practical purposes. Humans may (and will) mix and match criteria from all three levels as appropriate and create what they *and their peers* perceive as high quality UML diagrams. Of course, there is still a large degree of subjectivity in this definition, but it does capture the intuition.[1] Therefore, in the remainder of this paper, we will thus call a diagram (layout) *good*, if it mostly adheres to the criteria from all these levels, and *bad* if it mostly violates them. Unfortunately, elaborating or quantifying the notions of "good" and "bad" layout are beyond the scope of this paper. Generally speaking, in terms of the four levels of layout rules described above, if a diagram layout does not (significantly) violate any of the rules on the first two levels but adopts the rules described in the latter two levels we call it a "good" layout.

In contrast, "bad" layouts will violate some or all of the rules given on the first two layers. Since all diagrams have been created with the same tool, a minimum level of quality is maintained anyway, e.g.consistent coloring, font sizes, alignment and so on. Also, "bad" layouts ignore rules from the latter two levels. That does not necessarily mean that these rules are not partially respected, but they are not consistently followed. Some examples are provided in Appendix A, a sample questionaire can be found online at www.imm.dtu.dk/~hsto/vl4/q1.

IV. Experimental setup

We used [15] as a guideline for our experimental setup. We presented subjects with paper questionnaires showing one UML diagram and ten questions on the diagram, recording four categories of answers (right, wrong, "don't know", and no answer), time used, personal preference, and subjective assessment of layout quality. The dependent variables are accuracy and speed of comprehension, and preference. The independent variables are the experience level of the participants (beginner/advanced), the diagram type (class, use case, activity), the diagram size (small/large), and, of course, the layout quality (good/bad). Altogether, we ran three experiments with 78 participants. The main purpose of the first experiment was to validate the experimental setup, the questionnaires, and the instructions, to estimate the time required, and to explore learning and carry-over effects. Minor adjustments have been made to the setup for the second and the third experiment. In the remainder, we will focus on the setup of the second and third experiment. The details of the setup are discussed below; a summary of the experimental setup and study design is shown in Fig. 1.

[1]This will also be confirmed by the empirical results we discuss below: they exhibit both a wide variance in subjective assessment of quality and, on average, a strong preference for "good" diagrams, cf. Fig. 2 (c, d).

A. Model population

The models used in the experiments have been created by students as part of their coursework in a requirements engineering course taught by the author. These models belonged to one of three case studies and have been prepared by teams of 4-7 students over a period of twelve weeks with an approximate effort of 600-800 working hours for each model. For each case study, two or three teams worked in parallel; for each case study, the model of the team achieving the highest grade was selected.[2] This procedure ensured several desirable properties.

Firstly, by using models created by students undergoing the same course and being awarded the same grade, very similar levels of modeler capability and model quality may be assumed. Furthermore, the models used exhibit a large degree of methodological homogeneity in that they are very similar in terms of model structure and size, model and diagram usage, and frequency distribution of diagram types. Also, in the models used in our experiments, model elements had their original, semantic-bearing names, whereas in some previous experiments this vital aspect seems to have been deliberately eliminated by giving meaningless synthetic names to model elements (cf. [10, p. 1697]).

Secondly, due to the project oriented nature of the course, the evaluation criteria, and the fact that the evaluation is carried out by practitioners rather than academics, we can assert that the models underlying our experiment are realistic in the sense that their size, quality, and purpose is very close to industrial reality. Finally, all of these models used exist at the same stage of the software life cycle, namely requirements analysis.

In contrast, all earlier works seem to have used only a single case study and model, and most work has been carried out on models at the design or implementation level. Also, there is no indication in previous work as to how close to the reality of practical software development the underlying models are.

B. Diagram samples and questions

From each of the three models selected from the model population, we chose one large and one small example of class, activity, and use case diagrams with particularly good or bad layout. The size of a diagram was measured by the number of graphemes in the diagram. The quality of layout is measured by the adherence or non-adherence to a number of layout rules discussed below in detail. This step yielded three models (one from each case study) for each of the six buckets, that is, the categories of small/large diagrams of types class/activity/use case. So we arrived at 18 diagrams altogether which were then trimmed to have approximately the same size in each of the categories. We then derived two variants from each diagram exhibiting good and bad layout (i.e., two different treatments), respectively, yielding 36 different diagrams. Some examples are provided in Appendix A, a sample questionaire can be found online at www.imm.dtu.dk/~hsto/vl4/q1.

[2]The grades were awarded by an external censor, not the teacher.

Fig. 1. The experimental setup and study design.

A catalog of ten questions was developed for each of the three diagram types. These catalogs have then been adjusted to the other five diagrams of the same type, e.g., changed the model element names used in the diagrams, changed the expected answer to questions, or adjusted to the diagram size. These 18 sets of similar questions were then combined with the 36 diagrams to form 36 different sheets with one diagram and ten questions each. For each of the 18 models, there are two sheets with the same questions on the same model appearing once in a good, and once in a bad layout.

For the first experiment, different permutations of five different sheets were created to validate the questionnaires, estimate the time required, and to explore learning and carry-over effects. For the second and third experiment, four systematic permutations of nine sheets each were created such that each participant had at most five good or bad layouts, five small or large models, and exactly three models of each of the three types. No participant of any of our experiments was asked to answer two sheets with different layout of the same model.

C. Participants and completion rates

The participants for our experiments were recruited among students from different computer science classes at the Danish Technical University in Lyngby. All students participated voluntarily with no reward or threat and under complete anonymity, i.e., it was clear to students that their performance had no influence whatsoever on their grades, for instance. For the first two experiments, participants came from two parallel 1st year Computer Science BSc. courses on OO software development using UML. From now on, we will refer to this group of students as "novices". The experiments were run towards the end of the term. There were 21, and 22 participants in the first two experimental groups, respectively. Immediately before the experiment, all participants received a ten-minute introduction to those parts of the UML that were covered in the experiment.

For the third experiment, participants came from a Computer Science MSc. course. All participants had just completed a course on requirements engineering using UML worth 10 ECTS points. There were 34 students in this group, which we will refer to as "experts" in the remainder. Altogether, 6290 questions were asked, 6153 of which were answered, and 5487 of them with an answer other than "don't know", which is a completion rate of 97.8% for any answers and 89.2% for answers other than "don't know". When looking at completion rates of the core questions (i.e., without demographic, assessment, and time), the completion rate is somewhat lower (see Table I). Half of the participants worked between 20 and 40 minutes on the second and third experiment (durations for the first experiment are not comparable).

TABLE I
DEMOGRAPHIC DATA ON THE PARTICIPANTS OF THE THREE EXPERIMENT.

	male	female	all	completion rate (core questions)
novices	40	3	43	80.0 %
experts	30	4	34	84.4 %
all	70	7	77	81.9 %

V. OBSERVATIONS

We present our observations for comprehension (accuracy, response time), and preference. A summary is given in Fig. 2, the exact figures are provided in Table II. Data analysis and presentation was done using R [22].

Fig. 2 (a, b) shows box plots of the correct and wrong/missing answers on good and bad layouts (indices $+$ and $-$), respectively. Obviously, there is a fairly large variance between subjects, and there is a tendency of subjects giving the right answer to questions. At first sight, Fig. 2 also seems to show that the scores for correct and wrong/missing answers are very close together for both good and bad layouts. This would indicate that there is no (big) difference between the comprehension of good and bad layouts as far as accuracy is concerned. Keep in mind, however, that the bars in the box plots represent *medians* rather than means. Thus, looking more closely at the figures, we see that there is actually a positive impact from good layout on the mean scores

Fig. 2. Summary of the measurements (left to right): density of treatment a; results for accuracy; results for preferences; results for response time. Indices $-$ and $+$ to treatments a through f indicate bad and good layouts, respectively. The bars in the box plots indicate medians rather than means.

(see Table II, top). Obviously, the distributions for right vs. wrong/missing answers are symmetric; we include the latter only for presenting the benefit (last column).

Fig. 2 (c, d) also shows box plots of the subjective assessment subjects offered for good and bad layouts. Clearly, the variance is rather large, ranging over the complete spectrum in the case of subjective quality of bad layouts ($c+$). Still, of all the aspects considered, these show by far the largest advantage of good layouts (see Table II, middle).

Finally, Fig. 2 (e, f) shows box plots of the average time subjects spent on answers and the average time subjects spent on answer relative to the number of correct answers. Due to the scaling, it is quite obvious that the response times for good layouts are smaller (i.e., better) than for bad layouts, although the size of this effect is not much larger than the benefit yielded in terms of scores as discussed in the previous section (see Table II, bottom).

To sum up, the impact of good layouts over bad layouts shows up consistently across a variety of different metrics: an increase in correct answers (+7%), a reduction in wrong/missing answers (-13%), higher preference (approx. +30%), and lower response times per answer/correct answer (-7% and -17%, respectively). The absolute size may appear to be small at first sight, but compared to the miniscule effects found for individual layout criteria (e.g. [20]), this was to be expected.

TABLE II
MEASUREMENT DETAILS OF THE BOX PLOTS PRESENTED IN FIG. 2.

Accuracy (a, b)

answers	bad layout μ_b	bad layout σ	good layout μ_g	good layout σ	benefit $\mu_g - \mu_b$
right	6.35	2.07	6.76	1.94	+6.5%
wrong/missing	3.65	2.07	3.24	1.94	-12.7%

Preference (c, d)

rating	bad layout μ_b	bad layout σ	good layout μ_g	good layout σ	benefit $\mu_g - \mu_b$
diagram quality	5.54	2.74	8.06	2.12	+31.3%
diagram clarity	5.61	2.74	7.81	2.27	+28.2%

Response time (e, f)

s/answer	bad layout μ_b	bad layout σ	good layout μ_g	good layout σ	benefit $\mu_g - \mu_b$
all answers	22.72	10.85	21.06	8.25	-7.3%
right answers	38.37	24.39	31.68	15.77	-17.4%

TABLE III
ANALYZING THE MEANS AND STANDARD DEVIATIONS FOR IMPACT OF EXPERTISE LEVEL.

Accuracy (correct answers)

right answers	bad layout μ_b	bad layout σ	good layout μ_g	good layout σ	benefit $\mu_g - \mu_b$
novice modelers	5.94	1.98	6.34	2.03	+6.7%
advanced modelers	6.76	2.07	7.22	1.73	+6.8%
	+13.8%		+13.9%		

VI. INFERENCES

Plotting the density function shows a highly skewed and partly ragged distribution (see the density plot of treatment a+/a$-$ on the left of Fig. 2). Also, the Shapiro-Wilk test showed very low p-values for scores on correct answers (approximately 10^{-10}) for both good and bad layouts. Thus we conclude that our measurements cannot be considered normally distributed, so that we use the Wilcoxon-test rather than the t-test for testing our hypotheses. Also, this rules out a straightforward ANOVA analysis; given the complex experimental setup, developing a suitable generalized linear model is beyond the scope of this paper and has to be deferred to future work.

Since previous empirical work has struggled to measure effects of significant size attributable to layout improvements by single quality criteria, we first check for the existence and

order of magnitude for the compound effect and formulate the hypothesis $\mathbf{H_{0,1}}$: *Modelers perform equally well on diagrams with good and bad layouts*. We break down the notion of comprehension into accuracy and speed, and further into the number of correct vs. wrong/missing answers for accuracy, and time per answer vs. correct answer for speed and test the four hypotheses that there is no difference between performance for good and bad layouts for these aspects, respectively. We can reject all of them with at least high significance (see Table IV). We thus conclude that $\mathbf{H_{0,1}}$ can be rejected.

Another way of looking at the performance of users is to ask them to assess the difficulty of the tasks subjectively. So we formulate the hypothesis $\mathbf{H_{0,2}}$: *Modelers show the same preference for good and bad diagrams*. We measured preference with two independent questions asking for assessments of layout quality and diagram clarity. Testing the two respective hypotheses with the Wilcoxon test showed, that they may be rejected with at least high significance (see Table IV). Thus, we reject $\mathbf{H_{0,2}}$.

Previous work has found differences between experts and novices. Generally speaking, experts perform better than novices (cf. [3], [26]), and they seem to apply different strategies to diagram understanding (cf. [31], [26], [27]). Based on a literature survey and a discussion of the meaning of "expertise", Schrepfer et al. [23] hypothesize that novices should benefit more from good layouts than experts. So we formulate the two hypotheses $\mathbf{H_{0,3}}$: *Expert and novice modelers exhibit the same performance for good and bad diagrams, respectively* and $\mathbf{H_{0,4}}$: *Novice performance increases more than expert performance from good diagrams as opposed to bad diagrams*.

We computed the individual benefits in scores of right/wrong answers (see Table III). The individual benefit from good layout is almost identical for both groups of students ($\approx 7\%$), and the distance between novices and experts is the same for both kinds of layouts ($\approx 14\%$). Observe that the effect of experience appears to be twice as big as the effect of layout quality. Using the Wilcoxon test as before, we can reject $\mathbf{H_{0,3}}$ with very high significance, but we do not have sufficient evidence to reject $\mathbf{H_{0,4}}$ (see Table IV). One explanation for this surprising finding is that the advanced students that we tested in the "expert" group did actually not satisfy the definition of an expert, i.e., the experience levels of the two student groups were not different enough to show the expected effect.

TABLE IV
TESTING DIFFERENT ASPECTS OF COGNITIVE LOAD, WE REJECT THE HYPOTHESIS THAT GOOD LAYOUTS DO NOT IMPROVE USER PERFORMANCE AND ASSESSMENT.

HYPOTHESIS	P-VALUE	SIGNIFICANCE
$\mathbf{H_{0,1}}$: same user performance for good/bad layouts wrt.		
... correct answers	0.003	**
... wrong answers	0.002	**
... time per answer	0.061	*
... time per correct answer	< 0.001	***
$\mathbf{H_{0,2}}$:same user assessment of good/bad layouts wrt.		
... layout quality	$< 10^{-15}$	***
... diagram clarity	$< 10^{-15}$	***
$\mathbf{H_{0,3}}$ same performance for good/bad layouts by experts/novices wrt.		
... correct answers	< 0.0001	***
... wrong answers	< 0.0001	***
$\mathbf{H_{0,4}}$: novices benefit more than experts from good layouts		
... correct answers	0.39	-
... wrong answers	0.24	-

A. Discussion

We draw three main conclusions from our experiments. First, we could measure a notable effect of "good" layout on cognitive load, in particular when using subjective assessments. The effect we found seems to be strictly larger than the one found in previous experiments. We believe this is caused by using all available heuristics whenever applicable instead of trying to isolate effects of individual criteria. Another explanations has been offered by Eichelberger and Schmid who have attributed the absence of findings in their experiments to a small number of subjects. While Purchase et al. (who also find little to no effect) report population sizes similar to those in our experiments, they only measure individual layout criteria, and they seem to ask their subjects many fewer questions than we do.

Second, we have taken four different measurements that can all be understood as aspects of cognitive load (cf. [14]). While all these measurements show similar effects, the size of the effects found vary considerably. This is in line with previous findings of low correlations between subjective cognitive load and objective user performance (cf. [12]). Nevertheless, subjective assessments of cognitive load have been found to be very reliable indicators of the objective difficulty of a task. That could imply that the tasks provided in our experiments are so easy that they are well within the capabilities of our subjects, or that subjects have compensation strategies. Repeating the experiments with harder questions, under time pressure, or with additional secondary tasks may shed light on this question.

Third, novice modelers seem to benefit much more from good layouts than expert modelers. Similar findings have been made repeatedly in different contexts (see e.g. [23] or [31], [26]), so this is no surprise. However, the magnitude of the advantage experts have over novices may hold two interesting implications. On the one hand, it may be possible to develop a standardized test based on our experiments to assess the level of UML capability, similar to standard IQ tests. This may be a very helpful instrument in academic teaching and commercial UML certifications. On the other hand, Yusuf et al. have found characteristic differences in the strategies for understanding UML class diagrams, as employed by experts and novices (center-out vs. top-left to bottom-right, respectively, see [31]).

B. Threats to validity

a) Internal validity: Great care has been taken to provide systematic permutations of diagrams, questions, and sequences thereof to avoid bias by carry-over effects ("learning"). Any such effects would occur similarly for all treatments and, thus, would cancel each other out. Subjects have been assigned to tasks randomly. We can also safely exclude bias through the

experimenter himself, since there were only written instructions that apply to all conditions identically.

b) External validity: The selection of the models and diagrams may be a source of bias. However, we applied objective and rational criteria to the selection, and compared to previous similar studies, we used three different diagram types (rather than just one or two), a competitively large number of models, and very realistic models. The layouts for the models were, to a large degree, used-as-found, that is, they were created under realistic conditions by people unconnected to these experiments. On top of that, our study is based on a comparatively large number of subjects. So, the present study is certainly among the best validated among studies of its kind and we expect our results to be valid for UML models *in general*, i.e., we expect a markedly higher degree of external validity than previous contributions can claim.

VII. Summary

In this paper we presented three controlled experiments on the impact that the quality of layout has on the comprehension and preference of UML use case, class, and activity diagrams. In contrast to previous work, our approach focuses on human-made layouts rather than layout metrics and algorithms: here, we studied the combined impact of many of the layout criteria that had repeatedly been studied in isolation before. We observed a marked beneficial effect of "good" layouts to several distinct aspects of cognitive load. In particular, novice modelers benefited far more than advanced modelers. Our experiments exhibited a high level of validity through comparatively large numbers of subjects, models, diagrams, and tasks. Also, the models underlying our study are realistic in terms of their origin, size, structure, and so on. Finally, while previous work had focused on design and implementation level diagrams (i.e., class and interaction diagrams), this study focused on analysis level diagrams.

It seems likely that the results obtained here carry over in a similar fashion to other software engineering diagram types such as the remaining ten UML diagram types and SysML, but also to completely unrelated notations such as BPMN, the IDEF family, or the ARIS family of notations. However, this requires further empirical studies that would also replicate our experiments.

Acknowledgments

I would like to thank Lars Bogetoft Pedersen for letting run the experiments in his lectures. Also, I would like to thank the DTU Data Analysis center for their support, most notably Bjarne Kjær Ersbøll and Henrik Spliid.

References

[1] C. Britton, M. Kutar, S. Anthony, T. Barker, S. Beecham, and V. Wilkinson, "An empirical study of user preference and performance with UML diagrams," in *Proc. IEEE 2002 Symp. Human Centric Computing Languages and Environments (HCC/LE)*. IEEE, 2002, pp. 31–33.

[2] S. Y. Dawoodi, "Assessing the Comprehension of UML Class Diagrams via Eye Tracking," Ph.D. dissertation, Kent State University, 2007.

[3] A. De Lucia, C. Gravino, R. Oliveto, and G. Tortora, "Data model comprehension: an empirical comparison of ER and UML class diagrams," in *Proc. 16th IEEE Intl. Conf. Program Comprehension (ICPC)*. IEEE, 2008, pp. 93–102.

[4] B. Dobing and J. Parsons, "How UML is used," *Com. ACM*, vol. 49, no. 5, pp. 109–113, 2006.

[5] T. Dwyer, B. Lee, D. Fisher, K. I. Quinn, P. Isenberg, G. Robertson, and C. North, "A Comparison of User-Generated and Automatic Graph Layouts," *IEEE Txn. Visualization and Computer Graphics*, vol. 15, no. 6, pp. 961–968, 2009.

[6] P. Effinger, N. Jogsch, and S. Seiz, "On a Study of Layout Aesthetics for Business Process Models Using BPMN," in *Proc. 2nd Intl. Ws. Business Process Modeling Notation (BPMN)*. Springer Verlag, 2010, pp. 31–45.

[7] H. Eichelberger, "Aesthetics of class diagrams," in *Proc. 1st Intl. Ws. Visualizing Software for Understanding and Analysis (VISSOFT)*. IEEE, 2002, pp. 23–31.

[8] ——, "Aesthetics and automatic layout of UML class diagrams," Ph.D. dissertation, University of Würzburg, 2005.

[9] ——, "Automatic layout of UML use case diagrams," in *Proc. 4th ACM Symp. Software Visualization (SOFTVIS)*. ACM, 2008, pp. 105–114.

[10] H. Eichelberger and K. Schmid, "Guidelines on the aesthetic quality of UML class diagrams," *Information and Software Technology*, vol. 51, no. 12, pp. 1686–1698, 2009.

[11] M. Eiglsperger, "Automatic layout of UML class diagrams: a topology-shape-metrics approach," Ph.D. dissertation, Universität Tübingen, 2003.

[12] D. Gopher and R. Braune, "On the psychophysics of workload: Why bother with subjective measures?" *Human Factors*, vol. 26, no. 5, pp. 519–532, 1984.

[13] D. L. Moody, "The Physics of Notations: Toward a Scientific Basis for Constructing Visual Notations in Software Engineering," *IEEE Trans. Software Engineering*, pp. 756–779, 2009.

[14] F. Paas, J. E. Tuovinen, H. Tabbers, and P. W. Van Gerven, "Cognitive Load Measurement as a Means to Advance Cognitive Load Theory," *Educational Psychologist*, vol. 38, no. 1, pp. 63–71, 2003.

[15] S. L. Pfleeger, "Experimental design and analysis in software engineering," *Annals of Software Engineering*, vol. 1, no. 1, pp. 219–253, 1995.

[16] H. C. Purchase, "Metrics for Graph Drawing Aesthtetics," *J. Visual Languages and Computing*, vol. 13, no. 5, pp. 501–516, 2002.

[17] H. C. Purchase, J.-A. Allder, and D. A. Carrington, "Graph layout aesthetics in UML diagrams: user preferences," *J. Graph Algorithms Applications*, vol. 6, no. 3, pp. 255–279, 2002.

[18] H. C. Purchase, D. Carrington, and J.-A. Allder, "Empirical Evaluation of Aesthetics-based Graph Layout," *J. Empirical Software Engineering*, vol. 7, no. 3, pp. 233–255, 2002.

[19] H. C. Purchase, D. A. Carrington, and J.-A. Allder, "Experimenting with aesthetics-based graph layout," in *Proc. Intl. Conf. Theory and Application of Diagrams (Diagrams)*, ser. LNAI, M. Anderson, P. Cheng, and V. Haarslev, Eds., no. 1889. Springer Verlag, 2000, pp. 489–501.

[20] H. C. Purchase, L. Colpoys, D. A. Carrington, and M. McGill, *UML Class Diagrams: An Emprical Study of Comprehension*. Kluwer, 2003, pp. 149–178.

[21] H. C. Purchase, L. Colpoys, M. McGill, and D. Carrington, "UML Collaboration Diagram Syntax: An Empirical Study of Comprehension," in *Proc. 1st Intl. Ws. Visualizing Software for Understanding and Analysis (VISSOFT)*. IEEE Computer Society, 2002, pp. 13–22.

[22] R Development Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2011. [Online]. Available: http://www.R-project.org

[23] M. Schrepfer, J. Wolf, J. Mendling, and H. A. Reijers, "The impact of secondary notation on process model understanding," in *The Practice of Enterprise Modeling (PoEM)*, J. Persson, A.and Stirna, Ed. Springer Verlag, 2009, pp. 161–175.

[24] J. Seemann, "Extending the Sugiyama algorithm for drawing UML class diagrams: Towards automatic layout of object-oriented software diagrams," in *Proc. Intl. Conf. Graph Drawing (GD)*. Springer, 1997, pp. 415–424.

[25] B. Sharif and J. I. Maletic, "An empirical study on the comprehension of stereotyped UML class diagram layouts," in *Proc. 17th IEEE Intl. Conf. Program Comprehension (ICPC)*. IEEE, 2009, pp. 268–272.

[26] ——, "The effect of layout on the comprehension of UML class diagrams: A controlled experiment," in *Proc. 5th IEEE Intl. Ws. Visualizing Software for Understanding and Analysis (VISSOFT)*. IEEE, 2009, pp. 11–18.

[27] ——, "An eye tracking study on the effects of layout in understanding the role of design patterns," in *Proc. 2010 IEEE Intl. Conf. Software Maintenance (ICSM)*. IEEE, 2010, pp. 41–48.
[28] ——, "The Effects of Layout on Detecting the Role of Design Patterns," in *Proc. 23rd IEEE Conf. Software Engineering Education and Training (CSEE&T)*. IEEE, 2010, pp. 41–48.
[29] J. Swan, M. Kutar, T. Barker, and C. Britton, "User Preference and Performance with UML Interaction Diagrams," in *Proc. 2004 IEEE Symp. Visual Languages and Human Centric Computing (VL/HCC)*. IEEE, 2004, pp. 243–250.
[30] K. Wong and D. Sun, "On evaluating the layout of UML diagrams for program comprehension," *Software Quality Journal*, vol. 14, no. 3, pp. 233–259, 2006.
[31] S. Yusuf, H. Kagdi, and J. I. Maletic, "Assessing the Comprehension of UML Class Diagrams via Eye Tracking," in *15th IEEE Intl. Conf. Program Comprehension (ICPC'07)*. IEEE Computer Society, 2007, pp. 113–122.

APPENDIX

The following figures show some sample diagrams from our questionnaires. The class and activity diagrams are considered medium sized, the use case diagrams are considered small. Diagrams Fig. 3, Fig. 5, and Fig. 7 show "good" layouts, and diagrams Fig. 4, Fig. 6, and Fig. 8 show their respective variants with "bad" layout.

Fig. 3. Small use case diagram with "good" layout.

Fig. 4. Same model as in Fig. 3 with "bad" layout.

Fig. 5. Medium class diagram with "good" layout.

Fig. 6. Same model as in Fig. 5 with "bad" layout.

Fig. 7. Medium activity diagram with "good" layout.

Fig. 8. Same model as in Fig. 7 with "bad" layout.