An Empirical Investigation of the Utility of 'pre-CIM' models

Keith Phalp and Sheridan Jeary

Software Systems Research Centre, Bournemouth University,
Fern Barrow, Poole Dorset, BH12 5BB
kphalp;sjeary@bournemouth.ac.uk

Abstract

One of the motivations for a model driven development approach is that, by allowing a variety of stakeholders to take part in modelling, projects will be both more efficient than traditional approaches and will produce software that meets the needs of those stakeholders. This will be facilitated by transforming initial (CIM), models to design (PIM) and implementation (PSM). However, it follows that to gain fully from this strategy the initial models, which are the major driver for communication and validation of requirements and business needs, must be appropriate to this usage.

The VIDE project was an EC funded project which produced a successful model driven development tool-set, incorporating a variety of modelling capabilities, at each stage of the MDA process. Aside from support for model transformations, one of the motivations for VIDE was to provide accessible models for those stakeholders representing the client (or business) who may not share the modelling perspective and experience of software engineers.

This paper reports upon an empirical study which attempts to assess whether our proposed 'pre-CIM' models provide a more palatable starting point for users. In brief, our results suggest that the pre-CIM notation provides an accessible starting point for modelling, and enhance the modeller's experience whilst also suggesting that the use of the notation may have a positive impact on the quality of subsequent models.

1.0 Introduction

There has been much work discussing the importance of business and IT alignment [1-3]. However, despite company executives pointing out the necessity of alignment to the future success and competitive advantage of their business [4], there is still no ‘silver bullet’. In addition, there are a large number of stakeholders within any IT development and their views of the way that processes will be supported, the design of the prospective system and the system development process are often very different [5].

The Model Driven Architecture (MDA) has a Computation Independent Model (CIM) which is designed to enable the connection between the domain expert with a set of requirements and the IT architect with technical solutions [6]. This model may then be transformed using ‘minimal’ effort into a Platform Independent Model (PIM) and subsequently a Platform Specific Model (PSM). This transformation allows the (mostly) automatic generation of model or code at the next level, thus giving major productivity gains. The MDA has been identified as having a useful application in the current business environment where competitive advantage is requiring business to respond rapidly to a fast changing environment [7]. However, whilst MDA provides a framework for the development and maintenance of software systems that allows an analyst to describe both business and software assets [8] it is heavily weighted in favour of software assets and there has been little work in the area of model-driven development to encourage the domain expert to invest time and effort into understanding the principles and practices.

The VIDE project attempted to improve the involvement of a variety of stakeholders within early, particularly CIM, modelling phases by providing an accessible modelling toolset. In particular, one of the findings of our work was that existing, even CIM, notations which are at a higher level of abstraction, appear to be a barrier to many who do not share a software modelling background [9]. Therefore, simple accessible notations were developed (as part of the toolset), which would give a more palatable introduction to the modelling and via guided transformations ensure that domain experience was transferred into the software domain. Although such models are clearly part of the CIM phase, they were termed pre-CIM, partly to provoke discussion, but also because they provide a greater potential for usage by non-technical users than many current CIM models, hence moving modelling further upstream towards the business user. This is because current CIM modelling notations are often biased towards the mindset, paradigms and constructs of the software domain. Therefore the non-IT user will find them neither accessible nor intuitive.

This paper, therefore, reports upon an empirical study which attempts to assess whether our proposed ‘pre-CIM’ models do indeed provide a more accessible or palatable starting point for users, and also, as an additional research question, whether their use has an impact on the resultant quality of software models. The paper is organised as follows; Section two provides background on the VIDE tool, section three provides a description of our questions and hypotheses, section four outlines the study design and section five provides an illustration of our statistical analysis, whilst section six discusses findings and section seven offers conclusions.

2.0 Background

The modelling and management of Business Processes has become an important issue for businesses in recent years because, instead of focussing purely on functional or operational aspects, they allow a business to focus on the entire process and thus can add significant value [10, 11]. This approach creates flexibility and often allows the business to adapt to a rapidly changing business environment [12, 13]. There are many process modelling techniques available, but only a few, such as Event-Driven Process Chains [14] and Business Process Modelling Notation (BPMN) [15] have been accepted widely by the Business Process practitioner communities [16]. BPMN was developed by revising many of the other process modelling languages and using their concepts as a guide, for example Unified Modelling Language (UML) Activity Diagrams [17], Integrated Definition Method (IDEF) [18] and Event-driven Process Chains [14]. It was designed to be used by general business users as well as technical developers, whilst it also aimed to be sufficiently formal model to be translated into executable code [19].

However, capturing a domain process and modelling and designing it, is a complex task [11]. The model should be intuitive and easily understood by domain experts, but should not create ambiguity nor allow incorrect inferences to be made [20]. The need is for a graphical language that can capture the process [21] but often this comes with the cost of language complexity [9]; many languages are cumbersome, presenting the users with a large variety of constructs. Zur Muelen notes that Flowcharts in 1958 had 6 basic constructs and 4 extended constructs whereas BPMN in 2006 had 11 basic constructs and 39 extended constructs [9]. Wahl and Sindre [19] analyse BPMN according to the Semiotic Quality Framework and believe that the goal of the notation being understood by both non- technical domain experts and IT professionals is unrealistic. There are 23 different pre-defined elements to represent different types of events. Most of the concepts have their origin in the IT domain and not the business domain and are therefore not intuitive for the business user. Wohed et. al. [22] examine the suitability of BPMN using a patterns evaluation framework. They find a number of ambiguities caused by the lack of formalization and have issues with pools and lanes.

The VIDE project, was a European project financed under framework 6 which produced a successful model driven toolset [23]. To allow the business user access to the tool we developed a simple notation which would capture the information from the domain and which could then be transformed by a business analyst into the early stages of a Business Process Model. The notation involves the concepts of Roles, Activities and Data and the links between them. In addition, there exists a 'Bloop' which is depicted as a cloud. This is something that the business user cannot yet define, or is of an unknown type. It allows the model to gain from increased knowledge of the domain in an iterative manner. The VIDE Domain

Analysis Tool allowed the notation to be used in different levels, thus allowing more detail to be added to the diagram at each phase (see figure 1).

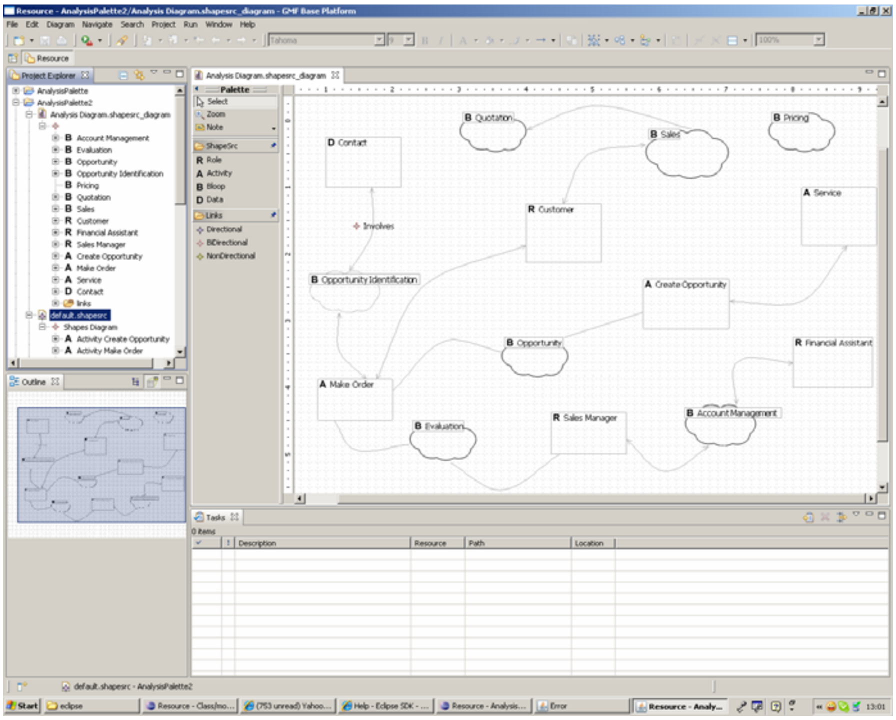

Figure 1: The VIDE Domain Analysis Tool [23]

3.0 Questions and Hypotheses

There were two fundamental research issues that this study sought to address:

1) *Whether the VIDE pre-CIM notation provided a more palatable (less onerous) route into systems development?*
2) *Whether the use of the pre-CIM notation had a positive impact on quality*?

These concepts are reflected in the choice of variables (data collected) and resulting hypotheses. Notably, we attempted to gauge quality by examination of direct attributes of the models created by the subjects, in both a positive and negative manner. We considered the correct identification of objects, mistakes made in identifying object types, the correct identification of keywords, and mistakes made in identifying irrelevant keywords. Hypotheses 1, 2, 4, 5, 6, and 7 (given below) relate to this research question (on resulting model quality). Since our hypothesis was to prove that the pre-CIM notation lowers the barrier to entry it was decided to run a number of separate experiments to test different stages of using the notation on two groups and to make comparisons of the results. Each experiment was based upon a cross selling opportunity scenario, with each group conducting the experiments in different sequences. The two groups would consist of Software Systems undergraduate students within the first ten weeks of their studies with few preconceived ideas or knowledge about modelling.

The barrier to entry is clearly subjective, but we attempted to assess this aspect by asking participants to score the difficulty of the tasks they had undertaken. Clearly, this is a measure of the perception of difficulty rather than an absolute measure of difficulty. However, we believed that this reflected the reality of getting clients involved in the CIM phases of development. That is, if they perceived notations to be difficult or onerous, they would not engage, whereas, in contrast, if they perceived the tasks to be within their capabilities and experience they would readily contribute. That is, the perceived difficulty (as tested by hypothesis 3) is crucial for practical application, as it is the perception of difficulty which often acts as a barrier to greater stakeholder involvement.

3.1 Hypotheses

H1: Group 1, using the pre-CIM notation, will make fewer mistakes in identifying object types (in Task 1 and 2) than the equivalent group without.

H1A: There will be no significant difference in the amount of mistakes made in either group.

H2: Group 1, using the pre-CIM notation, will make fewer mistakes in identifying irrelevant keywords (in Task 1 and 2) than the equivalent group without.

H2A: There will be no significant difference in the number of mistakes made in either group.

H3: Group 1, using the pre-CIM notation perceived (rated) the tasks (Task 1 and 2) as easier (lower difficulty score) than group 2.

H3A: Both groups rated the tasks as equally difficult.

H4: Group 1, using the pre-CIM notation, will identify more correct keywords (in Task 3) than the equivalent group without.

H4A: There will be no significant difference in the number of keywords identified correctly by either group.

H5: Group 1, using the pre-CIM notation, will identify more correct object types (in Task 3) than the equivalent group without.

H5A: There will be no significant difference in the number of object types identified correctly by either group.

H6: The group with pre-CIM will perform better at task five than the group without

H6A: There will be no difference in the performance of groups on this task

H7: Group 1, using the pre-CIM notation, will identify more correct keywords (in Task 4) than the equivalent group without.

H7A: There will be no significant difference in the number of keywords identified correctly by either group.

4.0 Study Design

To enable an investigation into the effectiveness of the pre-CIM notation, and to allow a fair and thorough evaluation, an experimental framework was used. In addition, we required a clear and transparent set of experimental guidance for experimenters, and clear instructions for subjects, typically volunteers. In drawing up the method and procedures, the guidelines outlined by Jedlitschka [24] were explored along with results of a study by Kitchenham [25] using those guidelines. These both highlighted the need for a format for conducting an evaluation which, whilst still retaining sufficient rigour, gives a clear structure and sets out the essential requirements of such studies in a clear and concise fashion. Budgen and Thomson [26] expand upon these ideas, providing detailed descriptions of the experimental processes that they adopted in evaluating a particular CASE tool. For example they highlight the importance of creating suitable experimental protocols, something which would prove particularly valuable to our workshop, since it had to be conducted across different locations.

5.0 Findings

5.1 Description of Data and Statistical Tests

For each of the hypotheses, numerical data was depicted as a standard box plot, with pre-CIM notation results represented alongside the group without. Hypotheses tests were carried out using independent samples t-tests, utilising SPSS version 16. In each case a pre-test of group statistics showed the number from each group (missing data items for each variable being omitted for individual tests rather than across all tests, in order to provide as much data as possible for each test). In addition, in we calculated:

- Levene's Test for Equality of Variances,
- the t-value,
- the calculated degrees of freedom,
- and the significance or p-value for a two tailed test.

Based on our propositions we had taken the positive step of forming single tailed hypotheses. Our tables of tests, generated by SPSS, however, gave a significance assuming two-tailed tests, therefore, we referred to the equivalent single tailed value (half of the table figure for a 5% test of significance).

Rather than explain the data provided in generic terms, a brief example is described below. However, a full set of data, and details of the statistical tests for each hypothesis can be found in Phalp and Jeary [27].

5.2 Sample of tests undertaken

The original third hypothesis was:

H3: Group 1, using the pre-CIM notation perceived (rated) the tasks (Task 1 and 2) as easier (lower difficulty score) than group 2. The alternative hypothesis being **H3A:** Both groups rated the tasks as equally difficult.

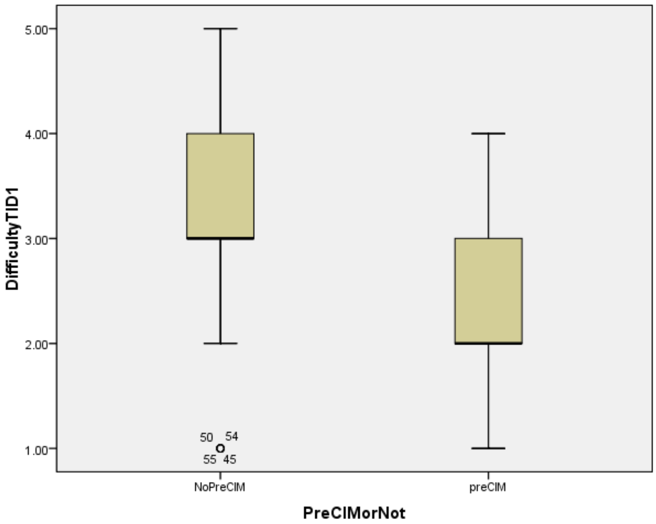

Figure 2: Box plots for Hypothesis 3

The box plot (figure 2) shows that the group with pre CIM appear to do much better. Indeed, the median for the group without pre-CIM is at a comparable error rate to the top of the box (75th percentile) for the group with pre-CIM, and there is little overlap in inter-quartile range.

Group Statistics

	Pair_ID	N	Mean	Std. Deviation	Std. Error Mean
Difficulty TID1	>= 2.00	20	2.3500	.93330	.20869
	< 2.00	38	3.3158	1.21043	.19636

Table 1: Descriptive Statistics for Hypothesis 3

This picture is also borne out by the descriptive statistics where, from examination of mean scores, the pre-CIM group give a much lower mean rating for difficulty, 2.35 (3 s.f.) than the other group, 3.32 (3 s.f.).

Independent Samples Test									
	Levene's Test for Equality of Variances		t-test for Equality of Means						
	F	Sig.	t	df	Sig. (2-tailed)	Mean Diff	Std. Error Diff	95% conf diff Lower	Upper
Equal variances assumed	1.333	.253	-3.110	56	.003	-.96579	.31053	-1.58786	-.34372
Equal variances not assumed			-3.370	48.152	.001	-.96579	.28655	-1.54188	-.38970

Table 2: Tests for Hypothesis 3

Furthermore, the significance of the t-test (0.003) for 2 tailed, (0.0015 for single tailed) shows a highly significant (P<0.01) difference in the perceived difficulty of the tasks (see Table 2).

Ranks

	Treatment	N	Mean Rank	Sum of Ranks
DifficultyTID1	1	39	34.81	1357.50
	2	20	20.62	412.50
	Total	59		

Table 3: Ranks for Non-Parametric Test for Hypothesis 3

As a further confirmation of this hypothesis, and the strength of the effect, we can also test whether this hypothesis holds if we undertake a non-parametric test, where we make no assumptions about the normality of the underlying distribution. In this case, a Mann-Whitney U test, gave a p value of only 0.001 (see Table 3 and Table 4).

This is again highly significant. Hence, it is clear that the group using the pre-CIM notation perceived the task to be less difficult (easier) than the group without.

Test Statistics[a]

	DifficultyTID1
Mann-Whitney U	202.500
Wilcoxon W	412.500
Z	-3.084
Asymp. Sig. (2-tailed)	.002

a. Grouping Variable: Treatment

Table 4: Mann Whitney U Test Statistics for Hypothesis 3

6.0 Discussion of Findings

For each of our hypotheses, the data showed that the group who used the pre-CIM model performed better. This was seen from our box-plots, and from the descriptive statistics, such as the respective medians and means [27]. However, in all cases bar one, (hypothesis 3 which related to perceived difficulty), the results were not statistically significant, though one of these results was very close to the 10% significance level.

Given the consistency of this pattern, that in all cases there appeared to be a positive effect on model quality, in favour of the treatment group (the treatment being the use of the pre-CIM notation against our baseline of no pre-CIM), our belief is that the study undertaken had insufficient power to be able to show the impact of the treatment. In other words, although there appears to be a small positive impact on model quality, the effect is not great enough to be demonstrated within this study. Of course it could be that there is genuinely no impact on quality, but again, this can be taken in a positive light, in that it is clear (from the box plots alone), that the treatment group always do at least as well (of course slightly better here), and that the additional modelling is preferred by the participants. Equally, it could be that our empirical study itself has not allowed sufficient power to demonstrate the effect of the treatment, and with hindsight, in using uneven group sizes we were constrained in the statistical tests that could be applied. This constraint, coupled with the use of pairs, which reduced the number of separate subjects, meant that the power was insufficient to demonstrate statistical significance. Hence, in terms of the original propositions, we cannot provide any significant statistical evidence that the use of pre-CIM models improves the quality of resulting models. Despite this, the results are consistent and promising, and suggestive of the need for further, perhaps larger or revised studies.

It is interesting that the single significant result was for hypothesis (H3) which attempted to gauge difference in the perception of difficulty when introducing the pre-CIM notation. Hence, it appears that, irrespective of the fact that the models cannot be demonstrated to be of higher quality, the subjects felt that the task was easier (or at least reported that is was) when they used the pre-CIM models.

7.0 Conclusions

This paper sought to gauge the effect of introducing a ‘pre-CIM’, notation as part of a series of models within the CIM phase of Model Driven Development. The motivation was that current modelling approaches appear to act as a deterrent factor in the involvement of the very stakeholders who often best understand the needs and requirements for proposed systems development. Hence, by providing a more accessible entry point, we hoped to improve stakeholder involvement within the crucial CIM phases of model driven development.

A crucial aspect of our approach was that our ‘pre-CIM’ models would be perceived, by the modellers, as providing a modelling experience which was less onerous than the alternative. In addition, if such a modelling experience improved involvement, one would hope that it did so without any detriment to the resulting software design. Indeed, in our study we took the even bolder step of suggesting that the models, whilst improving the perception of the modelling phase, might also improve the quality of the resultant software models. This provided two principal research questions: 1) Whether the Pre-CIM notation provided a more palatable (less onerous) route into systems development, and 2) Whether the use of the pre-CIM notation had a positive impact on quality? This was investigated by comparing the performance of two sets of subjects, one using our pre-CIM notation, and one without, across a series of sequential tasks (mirroring the CIM phase and supporting tool chain). Performance on these tasks was scored, and participants were asked to rate their assessment of the difficulty of the tasks for both groups. Data for this performance was analysed visually, e.g., using box-plots, descriptive statistics (such as means and standard deviations) were generated, and tests of significance (principally t-tests) carried out.

In brief, our findings were that the treatment group, using the pre-CIM notation, reported the CIM phase tasks as less difficult than the control group, and that the difference in mean difficulty score was highly significant. In addition, we found that in all other tasks the treatment (pre-CIM) group fared better, producing higher quality outputs, but these did not appear statistically significant improvements (suggesting either a small effect or insufficient power in our study)..

Hence, the main motivator for introduction of pre-CIM notations, reducing the deterrent of existing notations appears to be validated by this study. In addition, rather than concern for the possible negative impact of introducing such notations on software model quality our results suggest that model quality might also be improved (though the effect here was not significant). Of course, as with much software development, questions arise as to whether the increased modelling effort is justified. However, for us, this increased involvement was a specific project goal and one where the context (model driven development tools) promises significant gains in terms of meeting stakeholder needs, improving their understanding and ensuring better quality models and requirements through their active involvement in the production of CIM models.

8.0 References

1. Galliers, R.D.: Strategic information systems planning: myths, reality and guidelines for successful implementation,. European Journal of Information Systems **1** (1991) 55-64
2. Kefi, H., Kalika, M.: Survey of strategic alignment impacts on organizational performance in international European companies. 38th Hawaii International Conference on System Sciences, Hawaii, USA (2005)
3. Bleistein, S., Cox, K., Verner, J., Phalp, K.: B-SCP: a requirements analysis framework for validating strategic alignment of organisational IT based on strategy, context and process. Information and Software Technology **48** (2006) 846-868
4. Luftman, J., MacLean, E.R.: Key issues for IT executives. MIS Quarterly Executive **3** (2004) 89-104
5. Phalp, K., Jeary, S., Vincent, J., Kanyaru, J., Crowle, S.: Supporting stakeholders in the MDA process. 15th International Software Quality Management, Tampere, Finland. (2007)
6. Brown, A.W.: Model driven architecture : Principles and practice. Software Systems Modelling **3** (2004) 314-327
7. Brown, A.W., Iyengar, S., Johnston, S.: A rational approach to model-driven development. IBM Systems Journal **45** (2006) 463-480
8. Harmon, P.: The OMG's Model Driven Architecture and BPM. Vol. 2006. Business Process Trends (2004)
9. zur Muehlen, M., Recker, J.C., Indulska, M.: Sometimes Less is More: Are process Modelling Languages Overly Complex. 3rd International Workshop on Vocabularies, Ontologies and Rules for The Enterprise. IEEE, Annapolis, Maryland. (2007)

10. Giaglis, G.M., Paul, R.J., Doukidis, G.I.: Simulation for Intra- and Inter-organisational business modeling. In: Charnes, J.M., Morrice, D.J., Brunner, D.T., Swain, J.J. (eds.): 1996 Winter Simulation Conference (1996)
11. Barjis, J.: The importance of business process modeling in software systems design. Science of Computer Programming **71** (2008) 73-87
12. Smith, H., Fingar, P.: Business Process Management: The Third Wave. Meghan-Kiffer Press, New York (2003)
13. Ould, M.A.: Business Process Management : A Rigorous Approach. British Computer Society, Swindon (2005)
14. Keller, G., Nuttgens, M.N., Scheer, A.W.: Semantische Processmodellierung auf der Grundlage Ereignisgesteuerter Processketten (EPK)Ver offentlichungen des Instituts fur Wirtschaftsinformatik, Heft 89 (in German), University of Saarland, Saarbrucken (1992)
15. OMG: Business Process Modelling Notation Specification Version 1.1. Vol. 2008 (2008)
16. Recker, J.C.: Why Do We Keep Using a Process Modelling Technique. 18th Australasia Conference on Information Systems, Toowoomba (2007)
17. OMG: Unified Modelling Language Specification version 2.1.2. Vol. 2008. OMG (2007)
18. NIST: Integrated Definition for Function Modeling (IDEF0). Vol. 2008. http://www.idef.com/IDEF0.html (1993)
19. Wahl, T., Sindre, G.: An Analytical Evaluation of BPMN Using a Semiotic Quality Framework. 10th International Workshop Exploring Modelling Methods in Systems Analysis and Design (EMMSAD '05), Porto, Portugal (2005)
20. van Dongen, B.F., van der Aalst, W.M.P., Verbeek, H.M.W.: Verification of EPCs:Using Reduction Rules and Petri Nets. In: Pastor, O., Falcao e Cunha, J. (eds.): Proceedings of the 17th Conference on Advanced Information Systems Engineering (CAiSE'05). Springer-Verlag, Berlin, LNCS 3250 (2005) 372-386
21. Luttighuis, P.O., Lankhorst, M., van de Wetering, R., Bal, R., van den Berg, H.: Visualising business processes. Computer Languages **27** (2001) 39-59
22. Wohed, P., Van der Aalst, W., Dumas, M., Ter Hofstede, A., Russell, N.: On the Suitability of BPMN for Business Process Modelling In: Dustdar, S., Fiadeiro, J.-L., Sheth, A. (eds.): 4th International Conference on Business Process Management, Vol. 4102/2006. LNCS, Vienna, Austria. (2006) 161-176
23. VIDE: Visualize all model driven programming. FP6-IST-2004-033606. (2009) http://www.vide-ist.eu/ [Last accessed 31 January 2010]
24. Jedlitschka, A., Pfahl, D.: Reporting guidelines for controlled experiments in software engineering. 2005 International Symposium on Empirical Software Engineering (ISESE 2005), Noosa Heads, Australia. (2005)
25. Kitchenham, B., Al-Khilidar, H., Ali Babar, M., Berry, M., Cox, K., Keung, J., Kurniawati, F., Staples, M., Zhang, H., Zhu, L.: Evaluating guidelines for reporting empirical software engineering studies. Empirical Software Engineering **13** (2008) 97-121
26. Budgen, D., Thomson, M.: CASE tool evaluation: experiences from an empirical study. The Journal of Systems and Software **67** (2003) 55-75
27. Phalp, K., Jeary, S.: A pre-CIM workshop. Deliverable 11.3 of VIDE EU project FP6-IST-2005-033606 (2009) http://www.vide-ist.eu/ [Last accessed 31 January 2010]