A LEARNING METHODOLOGY TO IMPROVE TEACHING AND LEARNING PROCESSES FOR SOFTWARE ENGINEERING TOOLS

Lynne Fowler[1], Maurice Allen [2] and Geoffrey Roy[3]

Abstract* ¾ *Education in the Computing Disciplines *is of particular importance in the relatively new discipline of Software Engineering, which is aimed at the improvement of the production of large, quality software systems. The importance of CASE tool software in supporting the software development process is now becoming clear. Studies show these complex and sophisticated tools have a positive impact on quality and productivity but they have been slow to be adopted by industry: the reasons cited are numerous but include the difficulty in using and in learning to use the tool. Individuals have different learning styles and these differences affect user performance when learning a software package. This research has investigated learning styles and is underpinned by the constructivist theory of knowledge. An online tracking system has been developed to monitor user interaction within Rational Rose, our chosen CASE tool. The purpose is to perform analysis of software use in a quantifiable manner and address the impact of learning styles upon successful usage.*

Index Terms ¾ *CASE, Learning Styles, Online Learning.*

INTRODUCTION

Education and learning are ongoing and dynamic activities. As educators, our teaching and learning styles and methodologies must be continually reviewed to respond to developments in technology and to the changing requirements of society. Within the School of Engineering we are aware of the demands upon our students, particularly the pressures in their first year and the many assumptions made about their abilities. First year students often find the abrupt shift to personal responsibility for managing their learning difficult when entering University [1]. Within the School of Engineering, students are expected to effectively use current technology from the start of their studies and therefore need to be able to master numerous software packages. This is especially so within Software Engineering and Computing programmes. Software packages are constantly changing, being updated and replaced, with an ever increasing degree of complexity. Many professionals in industry who have to maintain state of the art skills also face similar problems to our students in learning these packages.

Software Engineering is a relatively new discipline grown from rapidly expanding demands in IT and computing. CASE, Computer Aided Software Engineering, is defined broadly:

> ".. as tools and methods to support an engineering approach to software development at all stages of the process. By 'engineering approach' we mean a well-defined, coordinated and repeatable activity with accepted representations, design rules and standards of quality." [2].

By looking at CASE tool software, our research aims to address learning issues that can improve our approaches to teaching software packages, along with considering transferable skills users need, to keep up with the dynamic and developing nature of software change. A CASE tool has been chosen because of the issues surrounding their usage in industry and their importance to the development of large complex software systems. Software development costs are increasing and modest improvements in software production would mean significant savings [3]. CASE tools were thought to be the answer to the software crisis [4]-[5] but although they do impact on software production [6], they have been slow to be adopted by industry [7]-[8]. One of the reasons cited for the poor acceptance of CASE tools in industry is the real or perceived complexity and the steep learning curve discussed by Fowler et al [9]-[10]. Vendor price of a CASE tool represents only a small portion of its true adoption cost, when the significant investment in people and time is included [11].

The CASE tool used in this research was *Rational Rose* [12], a professional package, currently gaining increased support and recognition within industry. Within engineering, we need to expose our students to software with similar capabilities to commercial software, as well as meeting our educational requirements. This package achieves these objectives [13].

Usability and learnability is concerned with easy of learning and use of a system. The learnability and usability of CASE tools is therefore of paramount importance and investigating learning issues will help both students in education, particularly in their first year, and professionals within industry.

[1] Lynne Fowler, Murdoch University, School of Engineering, Rockingham Campus, Rockingham, 6168, WA. lynne@eng.murdoch.edu.au
[2] Maurice Allen, School of Engineering, Rockingham Campus, Rockingham, 6168, WA. maurice@eng.murdoch.edu.au
[3] Geoffrey Roy, School of Engineering, Rockingham Campus, Rockingham, 6168, WA. geoff@eng.murdoch.edu.au

CONSTRUCTIVISM AND LEARNING

By investigating the use of learning styles, this research aims to address learning issues that can aid our students in their learning processes as well as identifying teaching strategies for our staff.

The importance of learning styles and their relationship to the construction of knowledge is therefore of paramount importance. The cognitivist approach used in this research has focussed on the styles of learning that apply to either different categories of learners, or the learning of different categories of material, providing insights into individual differences in learning and performance. The challenge is to identify the successful mental modelling strategies of the learner or to modify the learner's approaches to learning [14].

Social constructivists recognise that people and teachers play an active role in the learning process [15]. Vygotsky states that the culture gives the child the cognitive tools needed for his or her development The means by which teachers present information will therefore impact on the student's learning process.

Whilst there are numerous instruments for assessing learning styles, those advocated by Kolb, *Learning Style Inventory*, [16], and Soloman and Felder, *Index of Learning Styles*, [17] are both well known and accepted within education theory [18]. Both instruments provide an efficient way of analysing students' learning styles and complement each other in the information they supply.

Murdoch University opened in 1975 and Foundation Units have been a feature of the undergraduate first year experience. Completing a Foundation Unit forms a compulsory part of the first semester for new students. The broad interdisciplinary focus of these units presents a useful opportunity to assist first year Engineering students to develop a better understanding of their own learning approaches and to form strategies that will be relevant throughout their professional careers. The new Foundation Unit, A115 ***Interactions of Society and Technology*** based at the Rockingham Campus was developed primarily within the School of Engineering in 2000, and was the obvious place to include a study of learning styles [19].

The motivation for this inclusion was based on work by Felder [20] and Laurillard [21]. Felder proposes that it is beneficial to talk to students about their learning styles and the strengths and weaknesses associated with each style. Consequently, all our first year students complete the learning style inventories and analysis as an integral part of the Foundation Unit. This conversational framework identifies the activities necessary to complete the learning process. Teachback and self-explanation are components of a learning dialogue and are taken to be discursive, adaptive, interactive and reflective [21].

Students whose learning styles are compatible with the teaching style of a course instructor tend to retain information better, obtain better grades and maintain a greater interest in the course [20].

Individuals are different and these differences affect how a student performs with learning tasks. Hence individuals having an understanding of their learning style can take positive control over their learning experiences leading to more effective learning practices and study outcomes. Academic staff can use this awareness to develop material and teach in a greater variety of ways [10].

Previous papers [19],[22] discuss the application of the results of the learning style inventories and our latest results are shown in Table 1.

The key points to note from the Kolb results are :

- the diversity of learning styles in our first year students
- the lack of accommodators in our staff types and the heavy percentage of assimilator and converger types. Kolb suggests that a good career for converger types is engineering and for assimilators is teaching
- the lack of accommodators in our 4th year students.
- the close match of the first year male engineering profiles to our engineering staff types
- the divergence of the first year female engineering profiles to our engineering staff types.

The analysis of the learning inventory results has given us a mechanism for identifying pedagogical questions about our students and our teaching and learning methods [19].

EVALUATING USABILITY AND LEARNABILITY

Usability and learnability of a software package are serious concerns for the area of Human Machine Interaction. Designers have for years tried to identify more direct interaction with users during both the design phase and post development stages of a system or software package in order to assess the interface and its usability [23].

Feedback from surveys, interviews, online suggestion boxes, newsletters and online user groups can provide valuable information on a system and whether it meets its design requirements and its acceptability. Heuristic evaluation is an informal method of analysis where an evaluator studies the interface looking for usability problems and passing judgement according to their opinions and experiences [23]. This heuristic method has been extensively used, but is heavily reliant on the evaluator and is very subjective. Co-operative evaluation has also been widely employed whereby a user is observed whilst performing a task and is asked to think aloud [24]. The observational data from this method is hard to interpret. The cost and time factors involved are enormous and finding appropriate users is difficult.

Automatic support and reliable measurable quantities can reinforce and complement results from these methods [25]. Automating the collection of data would allow greater numbers of tests to be performed in a more cost effective and unbiased way.

TABLE I
RESULTS FROM KOLB LEARNING STYLE INVENTORIES 1999-2002

Clients	No. of Clients	Accommodator	Diverger	Assimilator	Converger
1^{st} year Engineering Students	99	9%	20%	31%	40%
1^{st} year Engineers -male	70	9%	7%	40%	44%
1^{st} year Engineers - female	24	8%	58%	13%	21%
Engineering Staff	12	0%	17%	41%	42%
General Arts and Commerce Students 1^{st} year	146	13%	14%	50%	23%
Year 12 all students	81	23%	10%	43%	24%
4^{th} year Engineers Students	18	0%	6%	33%	61%

The purpose of this research is to perform analysis of software use in a quantifiable manner. Our aim is to obtain measurements that can be used to assess the software package environment by tracking the online actions of each user and addressing the relationships between individual learning styles and the successful usage of the CASE tool.

ONLINE TRACKING SYSTEM

An online tracking system has been developed within Rational Rose. The aim of this system is to monitor student movement within the software package and thereby have an automated procedure for measuring the usability of the package. Users of Rational Rose work on diagrams within different views; use case view, logical view, component view and deployment view. By tracking actions and locating the view in which the action is performed, we can track movement between views and diagrams. This movement can then be related to student usage of the tool and patterns of usage can be related to the students learning style.

Rational Rose has a facility called Rational Rose Extensibility [26]. This facility to create add-ins enables clients to customise and automate features through the Rose Extensibility Interface (REI). These add-ins allow customisation in the following areas:

- main menu items
- shortcut items
- custom specifications
- properties
- data types
- stereotypes
- online help
- context-sensitive help
- event handling
- functionality through the Rational Rose scripts or controls (OLE-server), registering for events.

Functionality through scripts or controls allows access at the key points described in Table II.

TABLE II
ROSE EXTENSIBILITY INTERFACE EVENT

EVENT	EVENT NO.	EVENT CODE	DESCRIPTION
OnActivate	1	OAE	start of Rose session
OnAppInit	2	OAI	new copy of Rose
OnNewModel	3	ONM	new model created
OnOpenModel	4	OOM	new model opened
OnNewModel Element	6	ONE	creation of a new model element
OnModified ModelElement	7	MME	modification of model element
OnDelete ModelElement	8	DME	deletion of model element
OnSaveModel	9	OSM	on save of model
OnCancel Model	10	OCA	on click of cancel button
OnCloseModel	11	OCM	close of model
OnDeactivate	12	ODA	Rose shutdown

A model is the name given for a Rational Rose file and an element is an individual item on a diagram within a model. Add-ins can be provided for these events within Rose and the add-in is triggered and executed on occurrence. Initially, Rational Rose scripts were coded for some of the add-in events, but this was too restrictive as some events had to be coded as COM Server Interfaces. Consequently, all the add-ins were coded as Com Server Interfaces. Visual Basic was chosen as the Com-enabled language over C++ due to its simplicity to code. A dynamic link library (.dll) was created and the operating system registry updated. Once the registry had been updated and the COM server registered with the operating system then the add-ins can be controlled from the addin manager pull down menu within Rational Rose. An installer program was used to complete the updates for all our machines in student laboratories.

Each add-in was coded to write to a text file the event, date and timestamp, model name, model element name where appropriate, and the view the event took place in. The five main views logged from within Rational Rose are:

- view 0 = no view – start of Rose
- view 1 = user view
- view 2 = logical view
- view 3 = component view
- view 4 = deployment view

The add-ins are transparent to the users of Rational Rose as they only log information to a text file. The text file is then available for analysis after the Rational Rose session has been completed, Figure 1.

```
" OAE 1 0 view= 0 3/7/2002 10:03:44 AM On_Activate_Event Application_i
" OAI 2 22 view= 2 3/7/2002 10:04:06 AM  On_Application_Initiated "
" OCM 11 53 view= 2 3/7/2002 10:04:37 AM  On_Close_Model  Model_Name=
" OOM 4 118 view= 2 3/7/2002 10:05:44 AM  On_Open_Model_Event  Model_N
" ONM 3 161 view= 2 3/7/2002 10:06:25 AM On_new_model_Event "
" OCM 11 241 view= 2 3/7/2002 10:07:45 AM  On_Close_Model  Model_Name=
" ODA 12 245 view= 2 3/7/2002 10:07:49 AM  On_Deactivate_Event-Rose_sh
" OAE 1 0 view= 0 3/14/2002 10:40:57 AM On_Activate_Event Application_
" OAI 2 31 view= 2 3/14/2002 10:41:27 AM  On_Application_Initiated "
" OCM 11 43 view= 2 3/14/2002 10:41:39 AM  On_Close_Model  Model_Name=
" OOM 4 58 view= 2 3/14/2002 10:41:54 AM  On_Open_Model_Event  Model_N
" ONM 3 68 view= 2 3/14/2002 10:42:04 AM On_new_model_Event "
" ONE 6 111 view= 1 3/14/2002 10:42:47 AM On_New_Model_Element  Qualif
" MME 7 122 view= 1 3/14/2002 10:42:58 AM  On_Modified_Model_Element
" ONE 6 126 view= 1 3/14/2002 10:43:02 AM On_New_Model_Element  Qualif
" MME 7 148 view= 1 3/14/2002 10:43:24 AM  On_Modified_Model_Element
" ONE 6 159 view= 1 3/14/2002 10:43:35 AM On_New_Model_Element  Qualif
" OSM 9 833 view= 1 3/14/2002 10:54:49 AM  On_Save_Model_Event  Model_
" ONE 6 979 view= 2 3/14/2002 10:57:15 AM On_New_Model_Element  Qualif
" MME 7 989 view= 2 3/14/2002 10:57:25 AM  On_Modified_Model_Element
" ONE 6 992 view= 2 3/14/2002 10:57:28 AM On_New_Model_Element  Qualif
" ONE 6 1015 view= 2 3/14/2002 10:57:51 AM On_New_Model_Element  Quali
" MME 7 1190 view= 2 3/14/2002 11:00:46 AM  On_Modified_Model_Element
" MME 7 1195 view= 2 3/14/2002 11:00:51 AM  On_Modified_Model_Element
" MME 7 1202 view= 2 3/14/2002 11:00:58 AM  On_Modified_Model_Element
" MME 7 1208 view= 2 3/14/2002 11:01:04 AM  On_Modified_Model_Element
" ONE 6 1237 view= 2 3/14/2002 11:01:33 AM On_New_Model_Element  Quali
" OSM 9 1264 view= 2 3/14/2002 11:02:00 AM  On_Save_Model_Event  Model
" ONE 6 1350 view= 2 3/14/2002 11:03:26 AM On_New_Model_Element  Quali
" MME 7 1358 view= 2 3/14/2002 11:03:34 AM  On_Modified_Model_Element
" DME 8 1384 view= 2 3/14/2002 11:04:00 AM On_Delete_Model_Element  Qu
" ONE 6 1389 view= 2 3/14/2002 11:04:05 AM On_New_Model_Element  Quali
" ONE 6 1398 view= 2 3/14/2002 11:04:14 AM On_New_Model_Element  Quali
" MME 7 1398 view= 2 3/14/2002 11:04:14 AM  On_Modified_Model_Element
" MME 7 1418 view= 2 3/14/2002 11:04:34 AM  On_Modified_Model_Element
" MME 7 1439 view= 2 3/14/2002 11:04:55 AM  On_Modified_Model_Element
" MME 7 1451 view= 2 3/14/2002 11:05:07 AM  On_Modified_Model_Element
" ONE 6 1461 view= 2 3/14/2002 11:05:17 AM On_New_Model_Element  Quali
" MME 7 1468 view= 2 3/14/2002 11:05:24 AM  On_Modified_Model_Element
" MME 7 1474 view= 2 3/14/2002 11:05:30 AM  On_Modified_Model_Element
" ONE 6 1478 view= 2 3/14/2002 11:05:34 AM On_New_Model_Element  Quali
" MME 7 1483 view= 2 3/14/2002 11:05:39 AM  On_Modified_Model_Element
" MME 7 1488 view= 2 3/14/2002 11:05:45 AM  On_Modified_Model_Element
```

FIGURE 1
SAMPLE TRACKING FILE

The tracking file contains large volumes of data and in this pilot study the data has been analysed by extracting information on events and the Rational Rose view location when the event occurs.

ANALYSING THE TRACKING FILE

The text tracking file is used as input to Excel to produce plots. The analysis has been done by creating an Excel macro to automate this process. Plots of time against *view_calc* have been created for each completed session of Rose, that is from Rose activation to deactivation. Plots of multiple sessions can be displayed and compared, fig 2.

The caculation of *view_calc* is as follows:

*view_calc = ((20 * view code)+ event code)+(session no.*100)*

- view code is the code for each view 0,1,2,3 or 4
- event code is the number for each event 1, 2,3,4,5 12
- (20 * view code) allows for events occurring in each view to span a 20 increment gap.
- session number starts at 0 and is incremented every time Rose is activated as a new session.
- (session number * 100) allows for increments of 100 for each session plot

This calculation is basically extracting and filtering the data by view. The individual graphs show the movement the student makes between views and the individual tasks performed within a view, whilst working on a model, Figure 3 and Figure 4.

By tracking these events we have been able to monitor movement within the package.

ANALYSIS OF RESULTS

A pilot group of seven 3rd year students completed the learning style inventories prior to starting a course using Rational Rose, partial results can be seen in Table III. During the course, these students then were monitored whilst using the CASE tool.

Some immediate patterns can be noted from the results, Figure 3 and Figure 4.

Student 4 is a moderately Active person, a Converger and a strong Visual; as a result his chart shows frequent movement between views, indicating a heavy need to move between diagrams in the Rose model, figure 3.

Student 2 is the only female in the group and a Converger. She is also the only person showing Verbal and Reflective tendencies. However she is strongly Sequential and her chart shows more work and time being completed within a view before moving to another view, figure 4.

Many interactions exist using a complex package like Rational Rose. Having now developed a tool to track and plot the students movement within the package the next important step is to analyse the usage and relate it to student's learning style.

Therefore, the next phase of this research will track students under more controlled laboratory sessions and develop automated pattern matching techniques.

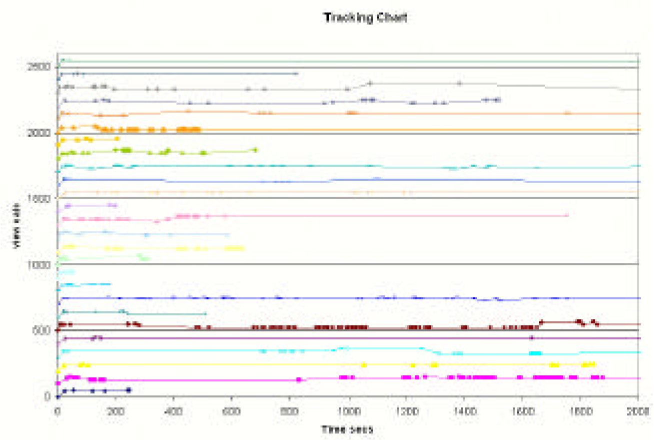

FIGURE 2
STUDENT 1

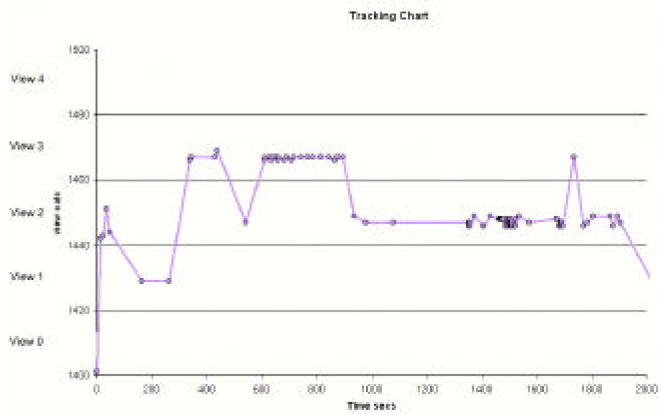

FIGURE 4
STUDENT 2

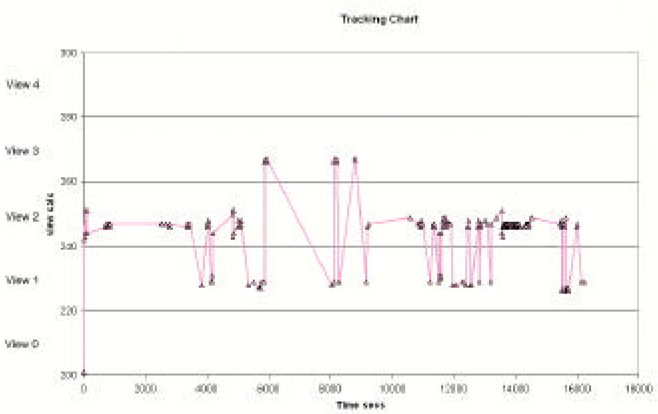

FIGURE 3
STUDENT 4

CONCLUSION

Student awareness of personal learning styles can aid their learning process. The greater understanding students have of their own learning styles the more responsibility they can take for achieving the best outcomes, even in an environment where the teaching and learning approach is contrary to an individuals preferred learning style. Staff can benefit from an increased awareness of their own learning styles and improve the effectiveness of their teaching styles, by varying and mixing their strategies to appeal to a greater range of students.

The inclusion of the learning styles initiative within our Foundation Unit emphasises the importance that we, as a School, feel this knowledge is to students. This topic is proving to be a popular part of the unit, with students and staff commenting on how interesting and useful it is to understand their learning style.

Having now developed and proved our tool to track the usage of Rational Rose, we are now in the crucial position of being able to start to develop automated pattern matching techniques. We can then relate the patterns of movement to learning styles and hence be in a stronger position to identify learning issues involved in teaching these complex tools.

TABLE III

3RD YEAR LEARNING STYLTSLE INVENTORY RESULTS

Student	**Kolb**	**Soloman Act/Ref**	**SEN/INT**	**VIS/VRB**	**SEQ/GLO**
student 1 male	Assimilator	1 ACT	3 INT	11VIS	3 GLO
student2 female	Converger	1 REF	7 SEN	1VRB	11SEQ
student 4 male	Converger	7 ACT	1SEN	11 VIS	3 SEQ

Key 1-3 slight, 5-7 moderate, 9-11 strong

REFERENCES

[1] C. McInnis and R. James, "Getting Started at University:Gap or Gulf? First Years on campus: Diversity in the initial experiences of Australian undergraduates," in *A Commissioned Project of the Committee for the Advancement of University Teaching.* University of Melbourne, 1995, pp. 29-37.

[2] G. Forte and R. J. Norman, "A Self-Assessment by the Software Engineering Community," *Communications of the ACM*, vol. 35, pp. 28-32, 1992.

[3] A. Fuggetta, "A Classification of CASE Technolgy," *Computer*, pp. 25-38, 1993.

[4] R. Pressman, *Software Engineering: A Practitioners Approach*, 4th ed. New York: McGraw-Hill, 1997.

[5] D. A. Gabel, "Technology 1994 software engineering," *IEEE Spectrum*, vol. 31, pp. 38-41, 1994.

[6] J. Iivari, "Why are CASE tools not used?," *Communications of the ACM*, vol. 39, pp. 94-103, 1996.

[7] J. D. Holt, "Current practices in Software Engineering - A survey.," *Computing and Control Engineering Journal*, vol. 8, pp. 167-172, 1997.

[8] S. Dutta, M. Lee, and L. V. Wassenhove, "Software Engineering in Europe: A study of best practices," *IEEE Software*, vol. 16, pp. 82-89, 1999.

[9] L. Fowler, M. Allen, J. Armarego, and J. Mackenzie, "Learning styles and CASE tools in Software Engineering," presented at Flexible Futures in Tertiary Teaching. Proceedings of the 9th Annual Teaching Learning Forum,, Perth, 2000.

[10] L. Fowler, J. Armarego, and M. Allen, "CASE Tools: Constructivism and its application to learning and usability of software of engineering tools," *Computer Science Education*, vol. 11, pp. 261-272, 2001.

[11] C. C. Huff, "Elements of a Realistic CASE Tool Adoption Budget," *Communications of the ACM*, vol. 35, pp. 45-54, 1992.

[12] "Rational Software", 2000, http://Rational.com.au/

[13] L. Fowler and M. Allen, "CASE tools-A basis for developing an online learning methodology," presented at AWCSET '99, 1999.

[14] C. McLoughlin, "The implications of the research literature on learning styles for the design of instructional material," *Australian Journal of Educational Technology*, vol. 15, pp. 222-241, 1996.

[15] L. S. Vygotsky, *Mind and Society: the development of higher psychological processes*. Cambridge (MA): Harvard University Press, 1978.

[16] D. A. Kolb, *Experiential Learning Experience as the Source of Learning and Development,*: Prentice-Hall, 1984.

[17] B. Soloman and R. Felder, "Index of Learning Styles (ILS)".1999:[online] HTTP://WWW2.NCSU.EDU/UNITY/LOCKERS/USERS/F/FELDER/PUBLIC/ILSPAGE.HTML

[18] S. M. Montgomery, "Addressing Diverse Learning Styles Through the use of Multimedia," presented at Engineering Education for the 21st Century: Proceedings of the 25th Annual Frontiers in Education Conference, 1995.

[19] L. Fowler, D. McGill, J. Armarego, and M. Allen, "Quantitative learning conversations: Constructivism and its application to learning in an engineering environment," presented at HERDSA, Perth, WA, 2002.

[20] R. Felder, "Reaching the Second Tier: Learning and Teaching Styles in College Science Education," *Journal of College Science Teaching*, vol. 23, pp. 286 - 290, 1993.

[21] D. Laurillard, *Rethinking University Teaching: a framework for the effective use of educational technology.* London: Routledge, 1993.

[22] L. Fowler, J. Armarego, and M. Allen, "Learner Theory and its Application to Female Learner Support in Engineering," presented at 10th International Women in Leadership Conference, Fremantle Western Australia, 2001.

[23] J. Nielsen, *Usability Engineering*. Boston: Academic Press, 1993.

[24] C. Lewis, "Using the "Thinking-aloud" Method in Cognitive Interface Design," IBM Thomas Watson Research Center, NY 1982.

[25] *Rational Rose 2000e Rose Extensibility User's Guide*, 2000.