

# **PROMOTING FAIRER GRADING OF GROUP BASED ASSESSMENT USING COLLABORATIVE IT TOOLS**

**Wayne Ellis, Mark Ratcliffe and  
Benjy Thomasson**

# Promoting Fairer Grading of Group Based Assessment Using Collaborative IT tools

Wayne Ellis, Mark Ratcliffe, Benjy Thomasson  
Department of Computer Science  
University of Wales, Aberystwyth, UK  
([wwe8@aber.ac.uk](mailto:wwe8@aber.ac.uk))

## Abstract

Group working is widely seen as an important aspect of personal and academic development but yet its assessment is problematic. Whilst awarding a group mark is straight forward and a real time saver, assessing individual contributions is much more difficult as it is largely impossible to account for individual effort. This unfairness negates the more widespread use of group working in an academic system that by definition looks to identify individual merit. From a student perspective, the inaccuracies can lead to weaker students doing less work as they feel they can get away with it, whilst other students might work harder to make up the shortfall. This paper discusses current approaches to overcoming the problem of unfair assessment and suggests a new approach aimed at revolutionising the grading of group work.

## 1. Problem

The assessment of group working is notoriously difficult, particularly with respect to individual accountability (Slavin, 1990). In most cases without this accountability, individual contribution is difficult to identify let alone assess. Documented effects such as the free-rider and sucker effects (Johnson and Johnson, 1990) show how these shortcomings not only impact the accuracy of marking but can also be detrimental to student learning. After group assessment both the assessor and the assessed can feel dissatisfied with the assessment process. Consequently, many academics feel that they cannot justify significant use of group work as part of formal assessment, even though they are well aware of the advantages collaboration can generate. Indeed, group working is actively promoted and required by professional accreditation. What is needed is a method to enhance the measures of individual performance.

## 2. Why Use Group Based Assessment?

The problems of group working for assessment are clear, so why do academic institutes insist on using group projects?

The purpose of Colleges and Universities is not only to teach their students knowledge about the courses they attend, but to also prepare the students for the real world. This is particularly important in vocational courses where students are learning the skills for a particular job, such as Information Technology, Teaching *etc.* Group working is a reality of the professional world; it cannot be escaped from and is the main reason for requiring group

project in courses. Professional bodies such as the British Computer Society actually include group projects for accreditation purposes.

There are many other advantages to group working. These range from building heterogeneous relationships (Webb, 1982) to encouraging students' responsibility for learning (Baird and White, 1984). Most notable for assessment is that group working allows assignment of more challenging problems, and promotes a more positive attitude toward the subject matter, thereby encouraging students' responsibility for learning (Baird and White, 1984), leaders aiding strugglers (Swing and Peterson, 1982) and strugglers improving leaders (Sharon and Shaulov, 1990)). Improving student learning is paramount for academics and therefore group working cannot be ignored due to its shortcomings.

### **3. Overcoming the Problems**

The main factor in solving the problem of unfair grading is individual accountability. The understanding of who contributed to the final submission, how much they contributed and the value to the group for the contribution, allows an assessor to formulate a fairer grade for the individual's contribution. This is often an impossible task to achieve. How can academics gather this level of understanding? Without constant supervision, which is just not possible, the necessary information is not available and so there is no full solution.

#### **3.1. Current Approaches**

In the absence of full data, various other techniques have been adapted to partially address the problem.

##### **3.1.1. Peer distribution of grades**

One method is the idea of allowing group members to assign the grades to the individuals in the group. This works as follows:

1. *Grade the work and assign a "project grade".*
2. *Take the project grade and multiply by the number of group members to give a "group score".*
3. *Give the group the "group score" and allow the group to decide on a fair distribution of the grades. In this way, the group assign each member a grade that has been jointly decided as a fair grade.*

This method has much credibility, but also some flaws. For example, if a group of 10 individuals receive a "project grade" of 70% where only 5 of the members put in any significant amount of work, then if the group agrees that the 5 members deserve a 90% grade each, this leaves the remaining 5 members a 50% grade. The members with a 50% grade have a considerably good score for no real contribution to the group, so without the hard working members being awarded greater than 100%, the non-working students will still receive a grade above the level they deserve for their efforts. This technique has the additional problem that a student may be awarded a mark far in excess than that of the project, which in relation to other projects might

be grossly unfair. Applying this American technique to a system that scorns marks much above 75% can obviously raise difficulties in assessment boards.

This technique can cause serious issues in peer pressure and even lead to bullying for grades. In smaller groups it is much more likely that the students putting in less work may try to pressure the more deserving student to equally share the grades rather than distribute the grades fairly. This second scenario would hopefully not be an issue in Universities with adult students, yet many feel that this is an important issue in this methodology.

### **3.1.2. Personal reports**

A second method is to ask group members to hand in a personal report with their group project. The purpose of the report is to allow the marker to try to identify what the individual student has contributed to the group, what the individual has learnt and how the students' evaluate the group.

This method can have good results, but is subjective. The method requires that students are truthful in what they claim. It is also possible that students do not put claim to their extra work due to the restrictions put on the project (such as a maximum number of hours spent on the project).

Students may also be unwilling to discredit their peers even if their peers did not contribute significantly; they may just feel uncomfortable in doing so or they may be concerned that they themselves or the group may be penalised for poor group management or for attempting to discredit a co-worker.

The only satisfactory option is for continual monitoring by an academic through regular reporting. This is a fairer system, but is very intensive on the part of the academic and even then they will not see everything that is going on. When there are very high student numbers, it is often too intensive for academics to monitor. This method is unfortunately very subjective, being restricted to what the academics see or are made aware of by the group of students.

## **3.2. New Approach**

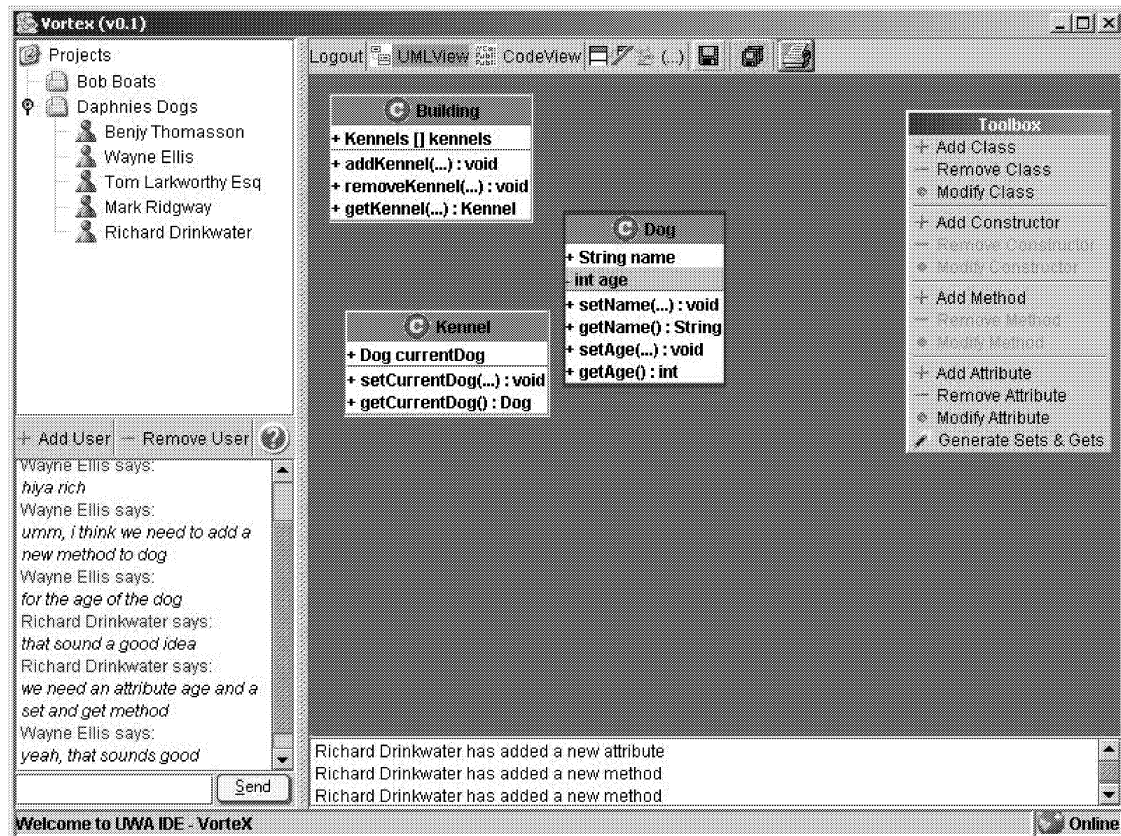
VorteX (Ratcliffe, Thomas, Ellis and Thomasson, 2003), a new approach jointly undertaken by the University of Wales, Aberystwyth and Khaydor Ltd, provides an environment that gives a variety of metrics to help enhance the fairness of group assessment during software development. Using VorteX, an instructor is able to monitor the actions performed by group members during collaborative working exercises; all actions undertaken by each individual member are recorded in a history log that is available for later analysis. Unlike other work involving data capture (MacGregor, Thomas and Woodman, 2001), VorteX records high-level actions; it is not interested in keystrokes.

VorteX began as a collaborative design environment in an attempt to identify perceived weaknesses in the education of software engineers. Following a series of presentations at International Conferences (Ratcliffe, Thomas and Woodbury, 2002) it became clear that the pedagogical processes involved in

teaching novice engineers were in need of some enhancement. Too many students were failing to gain a basic understanding of software development. Vortex was therefore designed in an attempt to establish what exactly was happening in the minds of the student learners. Presented as a software development environment to its users, it set about answering two fundamental questions: what were their first steps in undertaking designs and how did assistance from demonstrators impact these developments? Capturing user interaction and their design developments were fundamental to this investigation. Introducing collaboration into the equation was simply an attempt to make their decisions more explicit. Having to explain their actions to colleagues meant that the information could no longer be transferred directly from their heads to their own designs; it would have to be passed through their peers too.

In order to have a formalised design (which would be easier to parse), Vortex was designed to use UML, a standard diagram notation for developing software projects. At any stage the students can view the code to be generated from the UML and can edit it using the built in text editor. In essence Vortex provides all of the functionality that would be required by first year engineers; this was considered vital in convincing students to use it.

To encourage collaborative design Vortex enables students to be designated as belonging to one or more project groups. These teams are fundamental to Vortex and represent all of those members who have access to a particular development. In addition, the members of a group are then able to chat to each other using a built in MSN Messenger style chat tool. This metaphor allows each student to see which other participants are active at any time enabling communication to start with the click of a button. To ensure that no communication is lost, all conversations are captured and are linked in to the design developments. As they develop their designs, each group member is kept informed of developments as Vortex updates both the graphical and textual view of the designs. As students edit the design, add or remove components, they are asked to justify their changes through a non-intrusive data capture window. Students are encouraged to participate in this aspect of the design capture through the incentive of automatic documentation generated from the captured explanations. All actions carried out during the project are immediate, so group members always have the most up to date design.



**Figure 1: The Vortex Environment**

In addition to the main development tool shown in Figure 1, VortexX provides a number of server-side tools that provide additional functionality.

Using the server side front-end, instructors can create, update and delete administrative details within the system. To assist in running collaborative work, VortexX provides a module, assignment and team hierarchy for the organisation of students and courses. Once assignments have been created and assigned to modules, the students can be assigned to their groups. Students are then able to access their own projects and communicate with their own teams. Demonstrators are special users that can join any projects they wish.

The web-based interface can also be used to provide snapshots of the current designs as being progressed by all student groups. This facility enables an instructor to quickly assess the relative progress of the individual teams identifying which, if any, need specific attention. Once an individual team has been selected, extensive information can be provided to help analyse both the group's progress and individual contributions. Such information would be laborious to obtain by manually analysing the accompanying history logs, but the customisable metric software available through the administration tools enables accurate identification of individual contributions to the group work.

Using the main Vortex tool, student members can, at any time, view the current project status but the real strength of Vortex lies in its stored information. Using an associated animator, individual team members (and their instructors) can view the entire build up of the design showing each action step by step. Team members who may have missed a session or two are now able to quickly catch up on recent developments and, in a similar way, groups are able to carry out reviews of recent changes.

The real power of the Vortex animator is the ability that it provides instructors to view and assess detailed progress of the entire group project. Every step made by the team members can be analysed in the finest detail. This can include individual modifications to the design as well as the associated discussions that were being held at the time of the changes. Returning to one of the original questions inspiring Vortex, this feature allows an instructor to view the impact of information given to them by teaching instructors. All too often, students are given advice on an ideal solution rather than being informed of how they can improve on their current design. This might help them make progress but they are often left wondering what was wrong with their original work.

### **3.2.1.What Can Be Analysed?**

From analysis of the usage histories, Vortex is able to ascertain much information about the effort and value of contributions by individual members. This information becomes paramount when assessing the individual group members and includes: -

- Metrics on the start and end times of the project phases;
- Who did what, when and for how long;
- What changes were made;
- Percentage of individual contributions to the group as a whole;
- The ability for an instructor to replay work step by step for detailed analysis.

This information is displayed to the supervisor of the group assessment as a number of pie charts, gantt charts, and bar charts. In order to maximise the usefulness of this data, the metrics are customisable so that the supervisor can decide exactly what information should be returned. In this way they can select or reject actions such as logging in and out, chatting, and editing classes, attributes or methods.

The information provided by Vortex is intended to aid the assessment and to reduce the workload of the instructor. Rather than replace traditional methods, the information is aimed to support and accelerate the process.

#### **4. Preliminary Results**

The project is only in its infancy and full deployment of the tool is still a couple of months away, yet already we have some interesting results. Vortex has already been used on a group of first year Computer Science students. In our first test case, a project specification was deployed on a class of 115 students but resulted in only 20% of students producing a design anything like that expected by the lecturer. Providing information such as this can really help a lecturer assess the progress of a class. Further analysis appears to indicate that as a course progresses, the designs produced by students get closer and closer to those of the lecturer teaching the course. Whether this means that the students are producing better designs or simply that they develop an understanding of what the lecturer wants is still up for investigation.

Over the next few months as the captured designs are analysed we expect to gain more information that will challenge our perceptions of how successful courses really are. It is already clear that to be most effective, Vortex needs to be deployed to other institutions that use different methods and styles of teaching.

#### **5. Conclusion and Further Work**

This approach is not suggesting that current methods be replaced. Metrics can sometimes be misleading, but with the use of the new approach and the extent of the new information available to assessors, it allows a greater level of certainty to be achieved. Assessors can for example determine the extent to which claims made by students in their own critical appraisals or individual contribution reports are embellished.

Vortex is currently being used in the Department of Computer Science at the University of Wales, Aberystwyth to facilitate and assess collaborative design of computer software. Although this prototype application is used specifically for software development, the system applies equally to all collaborative activities. Over the next 12 months, the project is to be expanded to venture into the many other areas that collaboration is used both in academia and industry. The aim is to offer the same level of information to as many other fields as possible, so that everyone can benefit from the advantages of fair grading.

Vortex is also being extended as part of a PhD project to assist students in object oriented design through the use of a case based reasoning system. Through the collection of the collaborative designs (Ratcliffe, Thomas, Ellis and Thomasson, 2003) an intelligent case base is built up from many histories of different groups carrying out the project. The case based system will be able to offer students and groups automated assistance, through the analysis of previous designs.



## References

Baird, J., and White, R. (1984) *Improving Learning Through Enhanced Metacognition: A Classroom Study*. Paper presented at the annual meeting of the American Educational Research Association, New Orleans.

Johnson, D. W., and Johnson, R. T. (1990) *Cooperative Learning and Achievement* in Sharan, S. (ed.), *Cooperative Learning: Theory and Research* New York: Praeger.

MacGregor, M., Thomas, P., and Woodman, M. (2001) *AESOP (An Electronic Student Observatory Project)*, ITiCSE 2001, Innovation & Technology in Computer Science Education, Canterbury, UK.

Ratcliffe, M., Thomas, L., Ellis, W., and Thomasson, B. (2003) *Capturing Collaborative Designs to Assist the Pedagogical Process*, ITiCSE 2003, Innovation & Technology in Computer Science Education, Thessaloniki, Greece.

Ratcliffe, M.B., Thomas, L.A., and Woodbury, J. (2002) *Improving Motivation and Performance Through Personal Development in Large Introductory Software Engineering Courses*. 15th Conference on Software Engineering Education and Training, Kentucky, USA, 108-115.

Sharan, S., and Shaulov, A. (1990) *Cooperative Learning, Motivation to Learn, and Academic Achievement* in Sharan, S. (ed.) *Cooperative Learning: Theory and Research* New York: Praeger.

Slavin, R. E. (1990) *Cooperative Learning: Theory, Research and Practice*, Needham Heights: Allyn and Bacon.

Swing, S., and Peterson, P. (1982) *The Relationship of Student Ability and Small Group Interaction to Student Achievement*, American Educational Research Journal 19 259-274.

Webb, N. M. (1982) *Student Interaction and Learning in Small Groups*, Review of Educational Research, 52 (3) 421-445.