

Philip's Story – Exploring peer assessment

July

I am a teacher/manager at the Polytechnic working in the game design area. I take a module on programming which, to be frank, is a very dull course, very technical. Each week the students have a small programme they have to design, building up programming skills to help them work on group game design projects towards the end of their course. I have both adult learners (CET) and students straight from school (PET) in this class. The PET students seem very passive and are often lost, but they don't ask questions. I try to draw them out. I aim to create good visual PowerPoints to help them understand the different coding techniques they need. But I am afraid of the silence after I ask a question. I would like to get them to think, to be more self-empowered as learners, to initiate questions when they don't know something.

I have not been trained as a teacher. I have been working in the gaming industry for a number of years. When I came to the Polytechnic, I had a short course to help me understand the module syllabus system, how to break down topics for lessons, how to ensure each lesson covers what is required and that the outcomes are met. Although I have done further professional learning at the Polytechnic, I want to discover other ways. I am open to new ideas and ways of thinking.

August – Tools for Learning Design workshops

The Tools for Learning Design workshops provide me with space to think and explore aspects of teaching and learning that I have not had the luxury to consider before. I have the opportunity to discuss with people coming from diverse experiences and perspectives of the system. There are some very experienced and knowledgeable people here, who are deeply questioning their assumptions about teaching and learning as part of the workshops. I want to understand more about pedagogy and learning. Sometimes I am lost but other times I have clarity. I am having different sorts of conversations here than in my workplace, and it is helping me to see things in new ways. I am being asked to think about what I really value as an educator.

My goals: to be a better educator, think different, leave no students behind, question the fundamentals, encourage innovation.

I am now questioning the notion of academic grading. Can there be more depth to the whole process of grading than simply giving a lifeless alphabetical letter to denote a student's performance?

I remember my own diminishing experiences of being streamed when young into the bottom classes and thinking I was not capable of an academic career. I struggled and struggled to prove this wrong. I realise that I am a certain sort of learner – one who learns by “doing”, rather than reading theory. If given a chance to do, I can learn.

Bill talked about the joy of learning – learning takes place when the learner is excited, interested and willing to learn. Where is the joy of students in learning programming? Where is the joy with the grade that they get? Perhaps I can mitigate any negative effects of the single alphabet grade I give them by providing descriptive comments. But this is an enormous task for a teacher of 40-50 students.

Sue: As I listen to Bill and Philip talk, I realise that we are just opening the door of the issues of assessment. Sometimes we are so focused on the actual measuring we don't question what we are measuring, what we are valuing. We don't consider the role of assessing as one that can be shared by teacher and students to help build up better understanding of what it is that we are doing. I suggest that the notion of peer and self-assessment may give us a new way to look at the issue.

Philip

Yes, what are the values behind the assessment?

What makes a good programmer?

Is there a contradiction between these two statements?

Is the assessment done based on a learning outcome or based on the process of learning?

I am beginning to see that exploring peer assessment may help to empower the students in giving them some control over the process of grading, as well as helping them to see more deeply into the processes of what they are doing.

My research question: What does peer assessment mean for students? What are the responsibilities of a student doing peer assessment?

I aim to trial some peer assessment with the students.

Sue

Philip has asked me to be his critical friend. I realise how much his journey resonates with my own. Like him I came into teaching from industry, without formal qualifications. One of my jobs was computer programming which I found quite boring, and can relate to how his students might feel. As a beginning teacher, I was interested in peer assessment, and each year I trialled more sophisticated nuances.

So my role as Philip's critical friend is partly mentor (someone who has gone through this before) and partly someone who can encourage him to explore issues and what he values through dialogue. However, I can't but help to be enthusiastic. I need to be careful I don't bulldoze him into taking on something just because it is close to my own interests.

As I talk and give examples from my own trials of peer assessment I am reminded how so many teachers are very negative about it. It takes time to build student capacity and comfort with this – they can be initially anxious about others judging their work, and they don't have the skills to make good judgements. It is important to start small, to start encouraging students to think about their thinking and processes and what they are valuing in their own products. This may be helped through asking students what they are thinking in conversation, by adding a reflective question about the processes they used to their assignments, or providing proformas for peer assessment.

Early September – Conversation between Philip and Sue

Philip: "I have been so busy. There is no time to plan and think about this once I am back working in the Polytechnic. There are so many other things to organise. But I want to get peer assessment happening. So each week when students create their programme they give it to another who looks at it and gives feedback. But I need something to help them know what to look for."

Sue: "It is important to think about the different goals that you want peer assessment to achieve, as this will help you think how you might scaffold it for students."

Philip: "I want students to

- be exposed to other programmes and approaches to programming
- develop an appreciation for what makes a good programme – make hidden values regarding programming explicit
- develop skills of judgment which they can use in their own programming
- be exposed to different perspectives about their own programming and its value."

Sue: "So what do you value in a programme? Do you make that explicit for students?"

Philip: "Well, the programme has to work, but it is more than that. I haven't really thought this through before and I probably haven't made this explicit. The programme needs to be understandable and efficient. A really good programme may have some innovative or surprising element that can be used in other programmes."

Sue: "So based on this, you can create a *proforma* that teases out these different elements and students can then give a rating of one to five. Perhaps a column where they have to write a comment about what they specifically valued and what they think could be improved. Perhaps at the bottom could be a reflection comment where they think about what they have learnt about coding as a result of this peer assessment process. But I think it is important that you start modelling in discussions about what you think makes a good programme."

Sue: I spent some time with Philip, helping him to tease out the different things he valued in programming. I decided to create a sample proforma as I know, when beginning, it is hard to imagine what one may look like. I was in two minds about this as I think the struggling to create it yourself forces you into thinking it through, and I wanted Philip to have that experience himself. How much was I helping, pushing or filling? This conversation was very pragmatic – aiming to create something to solve a particular problem.

Criteria	Rating	Comments: What do you value? What could be improved?
1. Does the programme work? <i>To what extent does the programme do what it is supposed to do?</i>	0 1 2 3 4 5	
2. Is the programme coding understandable? <i>Is the programme well structured? Does it have line breaks between segments and explanatory comments? Does the code do what the comments say it does? Do you understand this coding enough to work with it?</i>	0 1 2 3 4 5	
3. Is the programme efficient? <i>Does it use processing power efficiently e.g. parallel processing?</i>	0 1 2 3 4 5	
4. Does the programme have the X-factor? <i>Does it have innovative or surprising solutions, did it enable discoveries that could be used for other possibilities, did it build on and value-add others' previous solutions in novel ways?</i>	0 1 2 3 4 5 “niftiness”	
Overall mark	A B C D	
Reflection: In assessing the programme, what have you learnt about coding? What can you use in the future?		

November – conversation between Philip and Sue

Philip: “As usual, I have been so busy. It is hard to find time to think about this. I realise that to do peer assessment was just too big a leap for me. Part of the reason is that each week, my lessons are planned out – PowerPoints followed by

practicals. I barely have time to cover what is needed. To provide time for getting students oriented towards peer assessment has just been impossible. It is too big a learning curve for them and, I think, for me.”

Sue: “Peer assessment is something that you need to invest time in to develop student skills if you want it to be empowering and not stressful for the students. This might mean less content initially and more process. Once the students are used to it, then the whole learning process becomes much faster as they have clear criteria that they are working with. It might mean a redesign of the course next time to give you some space up the front. But I guess the question is, what can you do now? I am interested in understanding what it means to be a good programmer – what that experience is like. Perhaps that can help us to think about this in another way.”

Philip: “While the programming may be dull, there are moments of joy when everything works. Serious concentration can move the coder into the “zone” where you start solving problems and seeing that things are getting there. Not all students get their programmes to work and they often flounder around, never reaching the “zone”. It is more often a struggle. If the problem is too big, it is hard to get a handle on it. Coders work with bits, they use procedural lockstep techniques, they are logical, they problem solve and they need to imagine how this relates to a bigger whole. As a coder, I am used to seeing a whole problem and knowing how to “divide and conquer” – to create the smaller questions or bits. They become doable. This is a typical mindset of the coder. But we also need to know how to piece the bits back into the whole.”

Sue: “Are you, as a teacher, also making the bits for the students? Do they ever practice working out how to cut bigger problems into bits?”

Philip: “You are right. This is exactly how I have been thinking about the course. In giving the students small components they can manage. In doing so, I have actually taken away important problem-solving skills they need. I have been thinking with a very procedural way of knowing – step A, B, C result. Give them specific skills or tools for each step.”

Sue: “What are the dispositions that you think programmers need?”

Philip: “Definitely the ability to problem solving, to persist, to try new things. But there are other skills – how to use online networks to help find answers and to even know how to frame questions to ask such networks.”

Sue: "It sounds like there are several layers of learning needed for your course:

- Coding skills
- Dispositions (persistence)
- Programming processes (networking with others, ways of problem solving, judging against criteria, being able to break down bigger problems)

How might different coding exercises also teach larger lessons about programming processes? How can you leverage off typical misconceptions?"

Philip: "Ah yes, there are definitely elements that I can use here. I begin to see how didactic I have been. Really, I want students to move into self-directed learning, to move beyond their culture as passive learners. I have been bogged down by the mindset of cutting things into bits. It is not so much about me creating course materials; it is about creating questions that enable deep problem solving. It is about encouraging greater interaction with each other which can arouse their imaginations.

In the past, I focussed on "doing" as a way of learning. But I also realise how much I am learning through these conversations. Conversation is a tool for learning. How can I use it with my students?"

Sue: "Perhaps the first step is to try and have a conversation with your students, perhaps one-on-one, about what they are thinking, how they are going about doing their programmes, and what strategies they are using?"

Sue: This conversation helped us to explore new ways of thinking about the problem.

By exploring the barriers to peer assessment, we actually discovered some fundamental assumptions in the design of the course which arose from seeing programming as coding rather than problem solving. If the aim is to develop a good programmer then it is important to really understand what it is that makes a good programmer. All of it.

Looking deeply at assessment helps to break the mindsets we have about how courses need to be. But what can Philip do with this?

December – Philip

Although I have been frantically busy I have tried to squeeze some student reflections in on their programming processes and strategies. First, I tried the approach of just directly asking my students in class. The response I got is not quite unexpected – no response or they are simply too shell-shocked to respond to such open-ended questions.

So what I tried next was to pose some of these questions in an assignment report I asked them to write about so I could get an idea of what's in their heads as they attempt to do their assignments.

- What were your learning strategies when tackling with learning how to cope with SDL_Net?
- Do you find SDL_Net easier or more difficult than using raw socket programming?
- What are some of the problems you encountered in this assignment?

I think some of the feedback I got from this is quite interesting. I realised a few alarming things like, for example, a lot of them are missing some critical tools knowledge which they should have already known at least a year ago! I also find interesting the difference between a student who “gets” programming versus another who’s still not quite there yet. The student who’s not quite there yet will usually find the raw socket programming technique simpler and easier to work with than the more complex SDL_Net compared to a student who is more advanced in understanding. I think this has been a very useful peek into their brains to help me better understand how they learn, and thus craft a better approach in my delivery in future teaching.

Sue

Philip sent me six responses of varying lengths and reflectivity to his reflection questions from his class. Despite not being able to answer Philip’s questions in class, all students actually made a good effort at unpacking their processes, showing a clear ability to do this. So this may be a very useful first step for others who are facing the brick wall of silence in their classes. I noticed that some of the more capable students are more self-directed: using networks to ask for help, doing on-line tutorials and being able to synthesise solutions from various

sources. The less capable ones seemed to ask a single person for help and have them explain exactly how to do it.

I now wonder whether these students would be more capable and confident of discussions on their processes in class since doing this assignment. Could the more self-directed students be good mentors and models for the less self-directed ones? I wonder how Philip might share some of the strategies people use to solve their programming problems to build up deep problem-solving processes. Initially, such conversations seem very strange, but the more capable reflectors can actually help pull the less reflective people along.

February – Philip

I have been continuing to build in conversations and reflections about learning strategies and thinking, and students are now more aware. The students are at the stage of giving presentations to each other on their projects, and will be giving feedback to each other that goes towards the assessment. It has proven to be very useful for student learning. It helps them to see things together, to be able to see other points of view and to start realising their own mistakes.

While I would like to bring this across to other courses that I teach, the difficulty is that I am not the main course manager for other courses. Although I would like to have a team in our section to continue this across a broader range of our courses, it is very difficult to discuss this with colleagues. Time is a big issue for me, but others coming from traditional approaches to teaching also pose a problem. It would involve a considerable change in mindsets about programming and about learning.

Acknowledgements & Background



This story was constructed by Dr Sue Stack (pictured, top) from a participant's conversations, emails and artefacts in the Tools for Learning Design project. While fictionalised it represents the conversational relationship between the participant and Dr Stack and how they used conversation to both achieve deeper understanding. It draws from narrative inquiry research methods to help the reader reflect on their own journeys. This is part of the Tools for Learning Design project led by Dr Helen Bound (bottom) and Dr Stack. We would like to thank the participant for allowing his story in all its vulnerabilities to be shared.



This story and the four others that are found in the Tools for Re-imagining Learning website convey the participants' questions, issues, processes and journeys. These stories have been slightly fictionalised with changed names to provide anonymity. They aim to express the authentic voice of the participants through using a conversational writing style.

The Tools for Re-imagining Learning website is a resource for trainers, curriculum and learning designers, and training leaders in the Singapore Continuing Education and Training sector interested in deepening understanding of their practice to create innovative and enlivening possibilities for their adult learners.

The Tools for Re-imagining Learning website and the Tools for Learning Design project overview can be found at www.ial.edu.sg.

For more information on the Tools for Learning Design project or the Tools for Re-imagining Learning website (content), please email Dr Stack at susan.stack@utas.edu.au or Dr Bound at helen_bound@ial.edu.sg.

We welcome questions or feedback on this publication, the Tools for Learning Design research report or the Tools for Re-imagining Learning website (layout or technical issues). Please email research@ial.edu.sg.

Copyright © 2012 Institute for Adult Learning

Published by the Institute for Adult Learning (IAL), Singapore
Research Division
1 Kay Siang Road, Tower Block Level 6, Singapore 248922, www.ial.edu.sg

This publication remains the copyright of the IAL. Users are free to download this publication for personal reference but any reproduction of content, either wholly or in part, requires the express permission of the Director, Research, IAL. For queries and clarifications relating to this publication or the use of it, please email research@ial.edu.sg.

About the Institute for Adult Learning

The Institute for Adult Learning (IAL) aims to contribute to the competitiveness of Singapore by developing an effective, innovative and responsive Continuing Education and Training (CET) sector that is able to meet the needs of industries and the workforce. It achieves this by raising capabilities, catalysing innovation, and leading research in workforce learning.

About the Centre for Research in Learning

The Centre for Research in Learning (CRIL) undertakes research that seeks to understand and develop the processes and practices of learning, teaching, assessment and curriculum design in the CET sector from multiple perspectives, settings and contexts. We work with those taking part in research projects, engaging practitioners in the research process and thereby developing communities of practitioner researchers.