Conference Postscript for Software Development as Learning

The Presentation

When I was preparing *Viewing Software Development as Learning* for the ACCU conference I found something was missing. I couldn't put my finger on it but it was, in fact, I re-wrote the presentation 3 times before I came up with the one on this website.

Having done the presentation I now know what was missing. One of the, perhaps the only, messages I wanted to make in the presentation was: *if you don't act on something you learn you haven't learnt*. And that was the missing element from the presentation.

For anyone sitting in the room watching me speak on the subject of learning, and hearing me emphasis that true learning requires action there was no action. So, where they really learning?

I'd always intended to make the presentation interactive but on the day luck intervened and gave me the ingredient that was missing: **a flip chart**.

When I watch presentation at conferences I'm often aware that we run out of time and don't get to the end of the presenter's presentation. When I speak to presenters before the presentation they are often concerned about whether, or not, they have enough material. Presenters and audiences don't like empty space.

I'm just the same. I over produce my presentations and include too much material. One solution to this - and one I try to do but have yet to master - is to design the presentation in such a way that sections can be dropped on the fly. The problem with this is: if sections can be dropped easily, how important are they to the presentation? Does their existence imply that the presenter is padding his or her material? Or that they are trying to deliver too many messages?

As is so often the case *less is more*.

When delivering *Viewing Software Development as Learning* I delivered the introduction - proposed my hypothesis - and I went over the back ground material (about a third of the presentation). I reached the slide which asks "Ideas from the audience”. This was always intended to lead a section where the audience proposed ideas.

I moved from the PC and slide projector to the flip chart and asked the audience to propose ways in which we could enhance learning - specifically when developing software. The audience was great, the audience came up with the 30 ideas listed below. This just goes to show that: Humans are Learning Creatures, hopefully, by those present learnt through their participation and will take many of these ideas back to their workplace.

The Results

1. Supportive company with a culture of learning
2. Focus our learning activities
3. Learn then act - application reinforces learning
4. Give time to learn (but be aware of the risks of taking too much time)
5. Provide Safety and Slack (see Tom De Marco)
6. Open Systems
7. Provide Feedback (e.g. peer review)
8. Share ideas and learning
9. Failures - be willing to fail, allow others to fail
10. Take away ideas from others, take ideas back to others
11. Retain your people - knowledge accumulates in employees who stay with the company
12. Understand risks and dangers, undertake experiments (but be careful with the time, too much can kill as surely as too little)
13. Do more stuff - variety helps learning
14. Appreciate different learning styles (e.g. start by Google “David Kolb Learning styles)
15. Recognize Quality
16. Reflection - take time to reflect - how do we reflect? when do we reflect? what is the trigger for reflection?

- Annual personal reflection - triggered by annual performance review or participation in a professional development scheme
- When Walking the Dog - triggered by the daily dog walk
- When pain backs you stand back and take time - e.g. you resign from a death march project
- Preparing a presentation

- Invoice Cycle - triggered by the monthly need to issue clients with invoices. was the money worth it? would I do that job again?
- Keeping a dairy
- Keeping a Blog
- When driving - could be dangerous!

17. Brown Bag lunches
18. Don't over stress people - with work or forcing them to learn
19. Learn what's new
20. Set Goals
21. Recognize and appreciate *Communities of Practice*

- Check out John Seely Brown, http://www.johnseelybrown.com/ and specifically http://www2.parc.com/ops/members/brown/papers/orglearning.html

22. Good communication - open up communication, value communication, allow others to communicate

- One person told this example, one Swedish company provides relaxed space in the Stockholm offices where people can sit on a sofa while the drink coffer or eat lunch, here they can chat with others, even people who work on different projects. However, these facilities don't exist in the companies UK offices. There are more breakthroughs and innovative ideas in Sweden than the UK.

23. Allow ignorance

- Allow others to be ignorant, don't make people think they *must know this* or to *pretend to know*
- Allow yourself to be ignorant, don't pretend you know everything, don't assume you know what someone is about to tell you, listen to them

24. Value Learning
25. Trust
26. Just do it!
27. Motivation - we learn better when we are motivated
28. Know when to stop learning - sometimes you already know the answer and you need to *Just do it*
29. Teaching - make it part of your job to teach others
30. Retrospectives - check out http://www.retrospectives.com/ and consider *Heartbeat Retrospectives http://www.twelve71.org/blogs/rachel/archives/000752.html*

Finally, one which wasn't proposed by the audience but I added myself:

31. Ask Questions - Don't stop when you get a first answer

- If I hadn't repeated my question "How do we enhance learning?" there would be 30 fewer ideas on this list.

The next question

This list is not enough by itself, how do you introduce these ideas into the organization? This is a question of change. Change is the most important issue facing software developers and development organizations.

I can't give you all the answers here and now, I can suggestion you go and read *Fearless Change* by Linda Rising and Mary Lynn Manns.

The Thanks

Many thanks to everyone in the audience, I can't remember all the names so rather than offend someone by missing you out I won't name anyone. I couldn't have done this without you!