The Dog, Hen and Corn...and post-completion errors

Paul Curzon, Middlesex University, Interaction Design Centre

The great essayist in Natural History, Stephen J. Gould is famous for using baseball examples in his essays about evolution, due to his passion for the game. He does this in an essay on the way pictures of snail shells are often printed back to front in text books. His argument is that mental errors are unforgivable and the essay searches for a reason why it might have been done on purpose to absolve the culprits.

> *"Baseball players make a proper distinction between physical errors, which can happen to anyone at any time and should engender no shame and mental errors – bonehead judgements – forgetting the rules – which should never occur ... No excuses possible."*

No excuses possible? This is a common view: mental errors are beyond the pale.

Gould's view and, if he is to be believed, that of Baseball players is a dangerous view for a computer scientist to hold. Mental errors can also happen to anyone and should similarly engender no shame. The common belief otherwise can lead to deaths. When airlines or trains crash it is often blamed on "pilot error" or "driver error" and that is the end of the story. The pilot or driver responsible is a villain. One class of airline crash involves the pilot flying the plane straight into the ground despite no apparent physical problems. This kind of error is called "Controlled Flight into Terrain". The ultimate bonehead judgement surely? The fact that such errors deserve a name is an indication of how common they are. This suggests there is some underlying cause that means that they are bound to happen eventually. If this is so, then the focus moves to the system designer to prevent them.

The stereotypical view of a computer scientist is someone who locks themselves away with their computer and avoids other people. In fact, software engineers and other computer scientists involved in software development need to understand human psychology as well as technology. As computers become pervasive, becoming embedded in everyday objects like washing machines and fridges this becomes ever more important. To see why, let's start with a puzzle. Solve it, writing down your answer, before you read on.

A farmer is on her way to the market with her trusty dog, Shep. The market is on the other side of a deep ravine. Getting to market always used to involve a long detour down the ravine and back up the other side. However, the local inventor who lives at the edge of the ravine decided to rig up a rope-pulley contraption to allow them to get straight across. It consists of a rope slung between pulleys on either bank, with a seat just big enough for one person hanging from it. By local convention the seat is always left at the market side where the owner lives so that it is easy for her to take it in each evening (After all she is not charging anyone for the service). On arriving at the ravine when going to market the farmer pulls the seat across from the far side using the rope. She gets in, hugging Shep, who goes with her everywhere, tightly on her lap. She then pulls herself across the ravine and continues into the village.

On the occasion in question she buys a new hen and a sack of corn. Returning home later in the day she arrives back at the ravine, and quickly realises she has a problem. She can only carry one thing across with her at a time on the seat. She will have to make more than one trip. That is usually the case when she travels to market. This time it is worse than that however. If she leaves the hen and the corn alone on either side, the hen will eat the corn. Similarly if she leaves Shep and the hen together on one side the dog will worry the hen and may mean it stops laying eggs. Shep is not interested in eating corn so it will come to no harm with him. Write down the series of steps (the algorithm in computer science jargon) that she must take to get everything across uneaten and continue on her way. We will get back to the answer later.

In the mean time I am in trouble again. I made the most wonderful peanut butter and tomato pasta sauce for dinner. I timed it to perfection: it was just ready as my wife walked in from work. As I was serving it onto the plates she told me all about her nightmare day. We ate it and everything was going fine until we returned to the kitchen to sort out the Strawberries I had bought for pudding. It was then that she noticed I had done it again. I forgot to turn the gas ring off. How could I be so stupid? I know it is important. I could burn the house down, gas us and even without such disasters think of all the energy I waste (and saving the environment is supposed to be important to me!), and yet just when I least expected it, I did it again. Bonehead! I cannot claim I did not know. Maybe I should stick to microwave cooking. I never leave that on after serving the food!

Whenever I make that mistake I try to persuade my wife it is not completely my fault. It may be a mental error but it is not a bonehead one. After all, why don't I ever make the mistake with a microwave oven, only a gas oven? Does the gas affect my memory? Or is it that the design of the microwave prevents me from doing it. This kind of error is a feature of the way my (and every other human's) brain is wired. We are prone to that kind of mistake. Not every time of course but often enough to be a pain. Never done it? Have you ever forgotten to switch off your car headlights on arriving at a destination, returning later to find you have a flat battery? Or left the petrol cap on the roof of your car at a garage? Have you taken the copy from a photocopier and forgotten the original? Have you used a vending machine and forgotten your change? They are all, along with many other similar and common mistakes the same error, psychologically. They even have a name: they are called "Post Completion Errors".

Here is one answer to the puzzle above (the other possible answer is similar):

- Farmer travels across with the hen (Dog left with Corn but that is okay).
- Farmer returns.
- Farmer travels across with Dog.
- Farmer returns with Hen (as otherwise dog will eat it)
- Farmer travels across with corn.
- Farmer returns (leaving dog and corn again)
- Farmer travels across with the hen.
- *Farmer sends the seat back to the other bank.*

You may (or may not) have forgotten the last step – returning the seat. If you did forget it, you just made a post-completion error. The goal was to get everything across so the farmer could continue on her way. However, once that was done you had to put the seat back to the side it came from. If you made the mistake you cannot claim you did not know. It was there in the instructions, and you were even told why it was important.

What is it that makes all these errors in apparently different circumstances the same? They all involve trying to complete a primary goal (get to a destination, fill up with petrol, make a copy, getting chocolate, getting to the other side of a ravine with a dog, a hen and a sack of corn, etc). However in achieving that goal something about the way things originally were must be disturbed and then put back the way it was (lights that were on switched off again, put the seat back to the right side of the ravine, *etc*). The thing that makes a post-completion error possible is that these extra tidying up tasks must be completed after the actual goal has been achieved. You must remember to do them after you have achieved the thing you set out to achieve. Humans have an unfortunate tendency to concentrate on the main goal and forget about such completion tasks. The name post-completion error arises because they all involve failing to do something after completing the thing that the person had actually set out to achieve.

In a series of experiments that must have been fun to take part in, Michael Byrne and Susan Bovair showed that this error is not just a random, bonehead mistake but can be made to happen even in laboratory conditions. They also found that they could control the likelihood of it occurring. They showed that making the mistake depended on at least two things: a person's working memory capacity, and how much else they had to think about at the same time. Your working memory is your short term memory: it is the immediate memory you have for recent things, for things you are currently working on, but that you do not need to remember for long. If I tell you my phone number so you can go away and dial it, you store the number temporarily in your working memory. A month (or even 10 minutes) later it will be gone, but you should be able to retain it long enough to make the call. The greater your working memory capacity, the longer the telephone number you will be able to remember.

The post-completion error experiments involved subjects taking the role of a Star Trek helmsman (that is the fun bit!) using a computer simulator. The most complicated task was to fire a phaser to destroy a Romulan ship. Firing the phaser involved first charging a phaser bank, setting the focus, turning on tracking, tracking the ship and firing and then once destroyed turning off the tracking. The post-completion part is remembering to turn off the tracking. Two different versions of the phaser controls were used. In one, the helmsman had to remember to turn off the tracking after seeing they had destroyed the ship. In

the other (control) version they were not told whether the Romulans had been destroyed until tracking was switched off. That meant that they only discovered they had achieved their goal once the whole task was done. The trial for real did not start until the participant had been trained and shown they could do the task without error. The subjects then did a series of trials (also doing a simpler task setting the shields in between). Errors that the participants made of whatever kind were recorded.

The participants made many post-completion errors with the first design that made it possible – including 13 out of 14 participants making that mistake on their first training trial despite the manual explicitly telling them what to do at that point. More to the point the errors were made more frequently than can be explained by it being a random error. If it is a bonehead error, the world is full of boneheads! Whilst with the second control design where post-completion errors could not be made, post-completion errors disappeared, other errors were just as likely. Clearly the design of a computer artifact can prevent or encourage human error.

In a further, similar experiment, also using Star Trek tasks, Byrne and Bovair proved that working memory was instrumental in the making of post-completion errors. This time, the capacity of the helmsmen's working memory was measured before the experiment. They discovered that the better the working memory a person had, the less likely they were to make post-completion errors. The need to do the completion task has to be remembered in working memory while you do the rest of the task. A person can only remember a limited number of things at once. If they have limited working memory capacity then they are more likely to forget completion tasks.

The importance of working memory was demonstrated further. During the experiment some participants were required not only to kill Romulans but also do other mental tasks at the same time – remembering information given through headphones. This required working memory. Those people who were mentally loaded in this way were more likely to make post-completion errors – as their working memory was filled with other tasks, the completion task was more likely to be forgotten. If enough else was happening, even those with the largest working memory capacity would make post-completion errors. Experience only had a minor effect suggesting the errors may not be something that training can get rid of. Making post-completion errors is a feature of the way that the human mind works. Individuals cannot be personally blamed for making them. In the wrong situation anyone can make them. That gas ring was not my fault, honest! The fact that I was being so sympathetic, listening to all my wife's worries probably meant I was filling my limited working memory with all the concerns of her day at the time, increasing my chances of making the error.

If training cannot get rid of the errors can anything? Yes! Designer's can often completely eradicate them and can at least make them less likely to occur. Byrne and Bovair demonstrated this in the lab, but it is demonstrated every day in high streets and in homes up and down the country. Buy a different car and you may find you no longer forget to turn off the headlights if the new one beeps when you open the door with the headlights on. Good design. You may not forget your change from that vending machine if it gives the change before the chocolate. Good design again.

There are several approaches that designers of computers, hi-tech gadgets and other everyday objects can take to reduce the likelihood of this kind of error. The first possibility is to eradicate it altogether by redesigning the order of actions to be taken in the interaction. Post-completion errors occur because there are still things to do after the goal has been achieved. Ensure that the goal cannot be achieved until everything else has been done and the problem disappears. When the first cash machines appeared in walls around the country, the sequence you had to follow was: 1) Insert your card and type in your PIN. 2) Select the amount. 3) Take the money. 4) Take the card. Guess what. People walked off with their money (the point of using the machine), leaving behind the card. They were making post completion errors. Leaving a debit card lying around is not a good thing to do. However, it does not happen any more in Britain at least, and it is not because people have learnt from their mistakes. The design of the machines in Britain (though not in all other countries) have changed. Now you always get the card back before any money. The order of doing things has been changed. The new order is 1) Insert your card and type in your PIN. 2) Select the amount 3) Take your card. 4) Take the money. With this ordering you cannot take the money and leave the

card - you can of course forget the money but as that is your goal it is far less likely to occur, though still possible if you are distracted.

The Phaser controls in the Star Trek experiment used a variation on the above approach. Tracking could not be turned off before the Romulan ship was destroyed, so the goal would always be achieved before the completion task was done. Instead what was controlled was the helmsman's knowledge of whether the goal was achieved or not. The helmsman could not find out whether they had fulfilled their goal of destroying the ship or not without doing the completion task first.

Rather than just change the order of things a more radical solution is to eliminate the completion tasks. Completion tasks arise because the state of the world has had to be disturbed to achieve the task. Alter the interaction design so that the perturbation does not happen in the first place and the problem goes away. This approach is taken with some credit card payment machines - phones and petrol pumps, for example. If you give up your card by inserting it in the phone at the start of a phone call, by the time you put the receiver down you are likely to have forgotten all about it and make a post-completion error. A solution is to not give up the card in the first place. Instead of inserting it, swipe it. The state is not perturbed and there are no completion tasks to do at the end as you never let go of the card. A post-completion error that occurs commonly in everyday life, just because the task is done so often, is that of forgetting change: when using coffee machines, ticket machines, etc. Credit cards are a way of redesigning the task to remove the error – with a credit card you do not need to remember change as you do not over pay in the first place (a different post-completion error arises instead – but we just saw how to fix that). In fact the example of change shows that post-completion errors are not to do with high-tech gadgets as such – like me you have probably walked out of shops in the past without your change too. If you are lucky in that situation the solution is that the shopkeeper runs out into the street and prevents you leaving without it. If you are not lucky they pocket the money.

Another way of making post-completion errors impossible is to automate the completion tasks. They may or may not still be done after the goal is completed, but are done by the gadget rather than being left to the person. This can only be done if there is a way of telling when the goal has been achieved. This is the solution that ensures I have never forgotten to switch off the microwave. Because of the way microwaves work and the danger, unlike with a normal cooker, when you open the door, it automatically switches off. You cannot physically get the food out without turning it off.

Another solution is not to alter the tasks or their order but to give a warning. This is the approach often used in cars over headlights. When you open the car door the loud beep, given if the headlights are still on, is hard to ignore. On hearing the beep you remember the lights. Unlike the microwave solution, giving warnings is not foolproof, however. The shopkeeper chasing out of the shop after you is also this kind of solution, but you may have disappeared before they get to the shop door or you might ignore the shouts, thinking they were directed at someone else. The warnings have to occur quickly enough and be insistent enough that you cannot miss them or ignore them. The person being warned also has to understand what they mean. I often use hire cars and every so often I have a car that beeps if I leave the headlights on. The first few times I had no idea why it was beeping. When I got out and shut the door, the beeping went away. That's OK then! Problem gone? No the lights are still on. Why do the cars not just switch the lights off for you? Presumably because there are situations when you need to leave a car with its headlights on, such as if you break down.

The same cash machines that now give you cash after your card do still in fact allow the same mistake to happen when used for other tasks because they use the warning solution. I recently opened a new deposit account. A really helpful member of staff of the bank showed me how to use the machine with my new card. Getting money out involved taking the cash last (one post-completion error solved), but making a deposit involved taking the card last (Oops). The last thing the bank clerk said was “Oh and do not forget to take your card back after you have made the deposit – people are always forgetting their cards”. We know why. The solution adopted this time was for the card to be taken back at the end and just print a message on the machine saying "Please take your card" whilst after a slight delay emitting a beep. In that situation the solution is obviously not good enough as the person may already be walking away and their attention may be elsewhere. They may be in a noisy street so barely hear the beep. Photocopiers sometimes use beeps

to prevent us forgetting the original, but by then I have often walked away. The warning approach works better in the car as it takes time to climb out of the car after the beep has started. As it is a loud beep it is hard to miss it, unlike a written message. A solution that apparently works in one situation may not in another that appears to be the same.

If messages and beeps cannot guarantee a human doing what they are told, more physical reminders can be used. In supermarkets, the cashiers often have to sign on and off, using a key to lock and unlock the cash till. On finishing their shift it would be easy to forget to remove the key – it is a post-completion error. Making it would leave the till vulnerable to thieves. Avoiding the error occurring is important. The solution is to attach the key to the till operator's belt. They cannot physically leave the till without removing the key and so locking the till. The completion action must still be done and it must be done last. However it cannot be omitted (except perhaps intentionally by the till operator removing it, but then the problem would not be a post-completion error). This kind of solution is known as a forcing function. The design forces you to do the right thing as you physically cannot do the wrong thing. Petrol caps are often designed in a similar way – they are fastened to the car so that you cannot leave them behind. However, in this case it is only a partial solution – you can still forget to screw it back in place. The ramifications of making the error have at least been lessened though.

So what is the solution for the design of the seat for getting across the ravine? Altering the order does not work – the seat must be returned last or the person will be stuck on the wrong side. It could be eradicated altogether by just scraping the convention that the seat must be returned to the inventor's side after use. That may not be fair on her, however. Why does she take the seat in every night though? If it is just to keep the seat out of the weather, perhaps the seat could stop in a hut (like a cable car) on either side so that it did not need to be put away. Then it could just be left on either side. Too expensive perhaps? Alternatively, if the inventor's side is lower, perhaps gravity could be used so that the chair automatically returns to that side when you let go of it. Maybe an automated recording could remind the person to return the seat whenever it got to the other side, though we have seen that is not "foolproof". Which is the best solution may depend on things we do not know about. Perhaps you can think of an even better design. Or maybe in this instance, the inventor does not mind occasionally pulling the seat back, so it is not worth the bother and expense of solving the problem. Leaving the post completion error there is always a possible solution, but if so it should be chosen with an understanding of the issues involved. If that is the solution chosen, just don't call anyone who then makes the mistake a bonehead!

Human error is pervasive. We are not infallible but not all errors are random. Some, like post-completion errors, happen for systematic reasons, are persistent and cannot easily be eradicated with training. However designers can design in ways that reduce the likelihood of them occurring. Many designers do understand this aspect of human psychology and design appropriately. However, the number of situations where such errors are still possible suggests that not everyone has got the message. As computers become pervasive, it is computer professionals who are the designers. Many of the situations where we make these errors are minor irritations, though a major reason for bringing in computers in the first place is to make things better for people, not more irritating. What is more, in many situations it really does matter. People can die. It is therefore increasingly important that such professionals understand human psychology as well as technology. We have to hope that those responsible for the systems that really matter, such as nuclear power plant, airline flight deck software and air traffic control systems, do. Computers are good at following plans, but the humans that use them are not. Computer scientists really do need to know psychology too.

Further Reading

1. M.D. Byrne and S. Bovair (1997), A Working Memory Model of a Common Procedural Error. *Cognitive Science*, 21(1):31-61
2. S.J. Gould (1997). *Dinosaur in a Haystack*, Chapter 16, Left Snails and Right Minds pp206-207. Penguin.

Acknowledgement This chapter was inspired by discussions with and unpublished research by Ann Blandford on designing to avoid post-completion errors.