Programmers are human too!

12th *February 2006 (adapted from a longer article originally written in April 2003)*

The case for the prosecution

I've just finished reading Alan Cooper's landmark book *The Inmates are Running The Asylum*. Cooper gives some clear insights into the failings of modern computer software and computer-driven products and to how we can correct these failings.

I agree with his main thesis on 'cognitive friction', the 'dancing bear' syndrome, the importance of interaction design and his persona-based approach. However, I think the chapter on Homo Logicus is misguided.

Cooper asserts that programmers in general are effectively a separate human species, mentally similar to robots but with additional juvenile human characteristics. He claims that programmers think in ways that aren't normal and are driven by a strong desire to control and master complex systems rich in cognitive friction. They prefer difficult systems that they only can master and revel in their interactive difficulties. He concludes that this species can't be trusted with good interaction design, and are incapable of understanding how users will interact with software.

The cross-examination

Homo Logicus is an extreme stereotype. Of course there are programmers with each and all of the characteristics mentioned, but are they the norm?

The book is attempting to outline a desirable future where interaction design is the driving force for good software development. In order to achieve that goal, programmers need to be 'on-side'. Cooper needs to convince them and make them a willing participant in his philosophy. Homo Logicus comes across as a biased attack on programmers which unfairly labels them and offers them no opportunity to change.

Why not express the same issues but couch them in an inclusive language? Let's make programmers participate in this process. Let's use their talents to help improve interaction design.

It is true that programmers master complex systems packed with cognitive friction. It is also true that there is satisfaction to be gained in prevailing against the adversity of bad design. But this is often not by their choice, but by the choices forced upon them by the situation or external constraints.

The defence

Programmers are driven by the *need* to master tools and systems developed by others, most of which are badly designed for their use. It is not that they desperately *want* to master these difficult tools or that the tools inspire loyalty. They are dancing bears for programmers.

Partly because there is a general dearth of interaction designers, programmers can be forced into assuming that role. When facing deadlines, strong demands about 'quality', weak and weak or no demands on interaction design; is it any wonder that they stick to what they're familiar with?

We do desperately need more interaction designers in software development, but we must swell the ranks from programmers as much as from managers and business analysts. Cooper admits that he was once a Homo Logicus himself, and now he is a leader in interaction design. There's hope for us all.