XimenaMiranda4

While giving some thought to the task of comparing physical objects with computational objects, I chose one of each to have in my mind while comparing them because I did not know how to start. I chose a Swiss Army knife and Mario (from Mario Brothers Game), just as an inspiration. . .

I find that one difference is related to manipulation of the objects. In the case of the knife, the object can actually become an extension of your body when you use it as a tool. In the case of Mario, there is an intermediate physical object that you manipulate in order to direct his movements. So in the computational objects, there is a "mental translation" of what you do with your hands to the control, and what the computational object (Mario) does. Actually, there are two steps, because one of them was done for me: you have to use certain language to program the computational object. Thinking about this, my conclusion is that manipulation of a computational object is more distant than that of a physical object.

In other words, I'm thinking that computational objects will always be abstract if the basic definition of abstract is used (having no material existence). This is part of what I like about "digital manipulatives" (Resnick, 1998): they take advantage of computation, but still they are more concrete than the turtle in LOGO. For example, **BitBalls** (Resnick, 1998) are more concrete than the turtle because you can actually touch them and manipulate them directly with physical movements. But they have a Cricket inside that you can program to collect information and exhibit certain "behaviors" (with lights, for example). I think that for certain students and certain topics, the computational objects on a screen are not so helpful (it's not always easy to imagine that you "are the computational object"). In this case, for many students it will be easier to understand the physical principles behind a ball if they can actually throw the **BitBalls** and analyze the information from its trajectory, than to understand by imagining that they "are the ball" on a screen. Of course "being the ball" on a screen will be better for many students than pencil and paper and their imagination (although this will be enough for some students, too).

Searching for a similarity between physical and computational objects, I asked myself if Mario could also be a tool to manipulate objects, like the Swiss Army knife. In a way, Mario is a tool; he can pick up "things" and hit "things". Considering other computational objects, like for example the little arrow used in Windows, it is used to open programs and start them, choose files, etc. The "things" are of a different nature, because they are material in the case of the knife and virtual in the case of the computational object. But the possibility to manipulate, or at least interact with other objects of their own kind is common to both computational and physical objects.

With both kinds of objects, I do think that sheer use of the tool can be a good learning experience. But working on the design of a tool is even better, going with the line of "design education" (Resnick et.al., 2000). Coming from a scientific background, my experience has involved the design of many tools for measuring the environment and making experiments. I've had to use many different materials (including magnets and electromagnets connected to the computer and to a plant to make playbacks of insect vibrations) to design and build my own tools for making experiments in Biology. And as a teacher, I have been inspired by that experience. I don't think the perception of Science being full of blackboxes (Resnick et.al., 2000) is correct, because it is completely the opposite from what I have experienced and what I have seen in my colleagues. I do think that the possibility of designing tools that involve different forms of programming technologies is an excellent tool for learning. For me, this is so more because of the positive arguments of this approach to learning than because of what was said about scientific research and blackboxes.