

Skeuomorphism and the Mental Model

Haley Young

November 12, 2012

Abstract

This paper discusses the use of skeuomorphism in user interfaces and whether or not it is appropriate in relation to the mental model. The mental model is referring to that of the user and the developer. This paper discusses possible instances when skeuomorphism has no effect or even could be detrimental to the alignment of the user and developer mental models. Overall, the conclusion arrived at here is that skeuomorphism should be used only in situations when it is helpful to this alignment, and if it's not, then the developer should use that freedom to create something held back by absolutely nothing but the mind's creativity.

Contents

1	Background, Preliminary, and Related Work	3
2	Main Content Sections	3
3	Conclusion	5

1 Background, Preliminary, and Related Work

When it comes to describing the mental model, a man named Don Norman is very good at describing it. He says, that there are seven stages of action. First the user must form a goal, then they must form the intention of that goal, then they must specify an action to reach that goal, execute the action, understand the effect that action had, interpret it, and evaluate the outcome of the whole process (discussed in class). When looking at the mental model in this way, this is how the user's mental model of an interface comes into existence. The developer's mental model is that of how the interface works and what it should do in relation to what the user sees and does. For the purposes of this paper, we are concerned with the mental model of the user lining up with the mental model of the developer.

2 Main Content Sections

I remember the first time I experienced skeuomorphism. I was hanging out with my friend and he showed me his new desktop background on his computer. It was literally a carpet covered portable wall, the walls that make up cubicles. He had all of his shortcuts tacked onto the carpet covered wall. I thought this was the coolest thing ever because it was like the shortcuts were real things on a physical wall, rather than just some digital rendition of what we have in real life.

According to Wikipedia, skeuomorphism is defined as “an element of design or structure that serves little or no purpose in the artifact fashioned from thenewmaterial but was essential to the object made from the original material” [1]. So basically an example of this in the physical world would be a phone that looks like a rotary phone, but where one would normally stick their finger in the hole, there's a button. This comes in to play in design models for software today, but more in the form of a debate in the computer science world. People wonder if elements of the mental model really need the skeuomorphic elements or not. The pro-skeuomorphism side says that it allows for a more user-friendly interface, and the anti-skeuomorphism side says that it's completely pointless and even sometimes confusing. This is where the alignment of user versus developer mental models comes in. If skeuomorphism does more to direct the user mental model away from the developer mental model, then skeuomorphism should not be used, but on the other hand, if skeuomorphism helps align user and developer mental models, then it should be used.

The first thing that can be looked at in terms of skeuomorphism is the effect it may have on the user's interaction with the interface. A supporter of skeuomorphism says, “For example, a leather-look texture (with the obligatory stitching) can stimulate memories of how leather feels and smells and, in turn, connotes a feeling of refinement and luxury” [1]. This could be a positive selling point for someone who is more interested in a memorable connotation the interface may provide rather than being concerned with looking cool.

One way to align the user and developer mental models is to give the user a hint as to what the interface is supposed to do. A lot of people see skeuomorphism as useful for this element of design. One person on Tumblr said, “Take advantage of people's knowledge of the world by using metaphors to convey concepts and features of your application. Metaphors are the building blocks in the user's mental model of a task” [2]. If someone can relate to the interface better because of things they have interacted with in the physical world, then this might make their mental model of the interface clearer and closer to the developer's intended mental model. For example, if the interface of a digital calendar looks like a physical paper calendar (one that someone might hang on a wall), a person is likely to understand the tasks the interface is meant to complete. Because this person is easily able to figure out what the interface is meant to do, they have a better understanding of what the developer was intending for the interface.

An example of a time when using the metaphor might be harmful to the alignment of mental models is with the bookshelf that Apple has created. Behar says, “The digital bookshelf doesn’t really work like a bookshelf.... You’re throwing all this extraneous visual noise at me and it’s confusing. My brain, which is used to the physical bookshelf, is confused because of the differences in usability. It’s cute, but not particularly useful” [3]. So in this instance, skeuomorphism is straying the user from the developer’s mental model. The user thinks the bookshelf works exactly like a normal bookshelf, because it looks like one, but in reality, the bookshelf doesn’t have the same characteristics as a normal bookshelf. If the user goes to do something that they can do with a normal bookshelf, but can’t do it with the interface they are currently using, then this is going to cause frustration for the user because their idea of the interface is not what is actually in front of them.

Along with this idea of not being able to do what one expects, a Youtube user says that it is pretty much impossible for an interface to have the ability to do everything someone can do with the physical object. When discussing interfaces which use skeuomorphism, he says, “Those applications look exactly like physical synthesizers and audio work stations, but how does that help you when your infinitely dexterous fingers are trapped behind a less than responsive glass touch screen or the often counter-intuitive mouse? It simply is not the best implementation” [4]. Along with the example of the apple bookshelf, there is no way the user can pick up a stack of books and put them back on the shelf exactly how they want them (maybe even backwards or upside-down) without having all ten fingers on the screen trying to direct the computer to understand what they mean. But, the bookshelf gives off the impression that it should be able to do such things because it looks like an actual bookshelf. Again, this will have a negative effect on the user’s experience. Their mental model of the interface will make the actual interface seem under-qualified for its purpose because the developer’s mental model didn’t include such options.

Another reason not to use skeuomorphism would be if it were to have no effect on the user’s mental model. Some designs (like typewriters) become obsolete. In this case, someone who is a younger user, might have no recollection or even interaction with the object the developer is trying to mimic. Because of this, the use of skeuomorphism in certain situations is just a wasted effort. Jochmann goes along with the typewriter example saying, “Do you think that a manual typewriter reference, a carriage return, will be understood to do what the return key on a keyboard does by people who grew up without typewriters? What appeal would a typewriter even have to them? These are properties you need to account for when choosing a real world reference” [5]. In this way, a developer might have to update their interface just by the simple fact that the skeuomorphic reference is no longer recognizable. The reference might become completely obsolete and therefore have no effect on developing the user’s mental model.

This brings up the question of the intended audience for a specific interface. If the intended audience for an interface is a younger crowd, then using an obsolete object as reference for skeuomorphic elements is not going to be helpful. It is either going to go completely ignored or it will confuse the user just as much as if the developer used a completely new looking interface (the learnability will not be made any better by the skeuomorphic design). In this specific type of situation, skeuomorphism would not aid in the alignment of the user and developer models of the interface because the user would not understand the reference. But, in the case of an older user, referencing an older physical object might be helpful, when Cronin was referencing Apple, he wrote “It is worth noting that Apple is particularly good at targeting this, generally, older and less technologically experienced user group” [1]. If an older person is already being daunted by all the new technology they’re trying to use, seeing a typewriter on the screen can make them feel like they might have a chance at understanding. In this case, the use of skeuomorphism will make the user less likely to give up on the interface. Because of the user’s ability to understand what the interface is supposed to do, in this certain situation,

skeuomorphism could have the effect of more properly aligning the user and developer’s mental models.

Another question to consider is whether or not skeuomorphism is holding the developer’s mental model back from endless possibilities. In this way, the idea is that the user’s mental model of an interface is affecting the developer’s mental model of the interface. If the user is controlling the mental model, then the developer is trying to align their interface with the mental model of the user. This limits the possibilities of the interface to the redundancies of everyday people. The question is, do we take interfaces to the next level, or keep them in the same place out of a desire for the interface to be popular. Microsoft has just recently developed an interface called Modern UI. When India Times published a review on the new Windows 8 operating system, Javed Anwer wrote, “Windows 8 puts a lot of emphasis on Modern UI. And it discourages users from spending time in the Desktop mode, which has been reduced to an app” [6]. The new Modern UI is an example of when a company directly chose not to use skeuomorphism, but their new interface still seems to be popular and highly anticipated. In this example, Microsoft did not allow their mental models to be tied down to the possible mental models of the users of their interface.

In this situation, skeuomorphism might be viewed as a negative thing that would have held back the mental model of the developers of Modern UI, but there may be instances when skeuomorphism should be used to help keep the mental models of both the user and developer in the same arena, no matter how much this may hold back the interface’s infinite possibilities.

3 Conclusion

In conclusion, skeuomorphism can be helpful and harmful in aligning the user and developer mental models of an interface. Skeuomorphism, in this sense, is basically a metaphor between a digital interface and physical object. If the metaphor is not recognized, then the metaphor should not be used. If even further, the metaphor is recognized, but the interface isn’t close enough to the physical object the developer is referring to, then it could cause confusion. But on the other end of the spectrum, if the metaphor is close enough and helps the user better understand the developer’s intentions, then it should be used. The worthiness of skeuomorphism completely has to do the specific interface, the target audience, and the developer’s desire to create something familiar versus new. In this light, a conclusion can be made in reference to each individual interface and how skeuomorphism might help align user and developer mental models, but a blanket statement cannot be made.

References

- [1] Shaun Cronin. Skeuomorphism in interface design. *Webdesign Tuts+*, July 2012.
- [2] edboaden. Reaction against skeuomorphism. *Tumblr*.
- [3] Mikey Campbell. Apple designers reportedly divided over use of skeuomorphic UIs. *Apple Insider*, September 2012.
- [4] VlogginEgan. Skeuomorphism. *YouTube*, July 2012.
- [5] Jakob Jochmann. What skeuomorphism can tell us about design thinking (and Apple). November 2011.
- [6] Javed Anwer. Eight days with Microsoft Windows 8. *The Times of India*, October 2012.