Dueling Interaction Models
# of Personal-Computing and Web-Computing

Matthias Müller-Prove, Frank Ludolph

Abstract

Today the user of personal computers is facing several inconsistencies which originate from an unresolved situation between two competing interaction models. The WIMP desktop model was developed nearly 30 years ago at Xerox Parc and Apple Computer. The web model became popular in the mid 1990s and has profoundly changed business and the perception of social relationships. Contradictions between these two models have a severe negative impact on human-computer interaction.

1. Introduction

The interaction model of modern personal computing was established and standardized with the success of the Apple Macintosh. Although the system had a couple of commercial predecessors – Apple Lisa and Xerox Star – it is fair to say that the introduction of the Macintosh in 1984 was the starting point of the desktop publishing revolution. Ever since, all systems follow the same set of concepts: their graphical user interfaces use windows and icons to mimic an environment that was the physical desktop of the office worker 25 years ago. Documents are stored in a folder hierarchy much as real paper documents are filed in folders and cabinets and archives. Commands are typically organized in a menu structure which is operated with a mouse; for practical reasons laptops have replaced the mouse with a trackpad. In general, all devices that move the cursor on screen and transmit clicks to the computer are called pointing devices. The term «WIMP desktop» summarizes the systems that adhere to these principles.[1]

Ten years after the introduction of the Apple Macintosh another branch of interactive systems became popular: the World Wide Web. Its inventors combined the ideas of Ted Nelson's hypertext from the mid 1960s with the transmission capability of the Internet to transfer HTML documents between computers – quite literally around the world. The class of programs that display web pages is called web browsers. The first browser was Nexus – developed by Tim Berners-Lee on a NeXT computer at the particle physics laboratory CERN in Geneva. He had a working prototype with a graphical user interface by December 1990. The program could display images (in separate windows), and had hypertext functionality built-in. Underlined anchor text could be activated with a single mouse click, which caused the targeted page to be loaded and displayed in a new window. Nexus was also capable of editing HTML pages, a feature that nearly all other browsers failed to implement ever since. Robert Cailliau recalls:

> «Usually a prototype is riddled with problems and the version that follows is a great improvement. With the World Wide Web the opposite is true. Tim's browser set the

[1] WIMP is an acronym for windows-icons-menus-pointing devices. It was probably invented by UNIX hackers, who liked the connotation of the term.

> standard for everything that followed and nearly a decade later no other browser has been able to match it.» [1, p.192]

NCSA Mosaic and the subsequent versions of the Netscape Navigator by Marc Andreessen became the most popular browsers of the 1990's because they were available and easy to install for Windows, Macintosh, and Unix. They introduced the behavior of replacing the web page in the current browser window with the loaded page after making a hyper-jump [3, p.39].
The action of using windows like clip-on picture frames and activating hyperlinks on single-mouse clicks seem to be just subtle modifications to the WIMP desktop interaction model. In fact they have a tremendous impact on the general user experience on the PC. The unresolved conflicts lead to the situation that we are facing today. This paper will compare the interaction model of personal desktop computing with the general interaction model of world wide web browsers.

2. Interaction Models

An interaction model is used to describe how the user's actions correspond to changes in the system model and vice versa. It is the continuous flow of information between the user and the machine that constitutes the interaction with the PC. The interaction models for WIMP desktop systems on the one hand and the web on the other hand utilize quite different elements.

2.1. Interaction Model for WIMP Desktop Computing

The basic interactive elements for desktop computing are windows, icons, menus for the command structure, and the pointing device for operating the widgets on screen. The desktop metaphor is the guiding principle that helps the user build a mental model about her personal working space in the computer.
On a very fundamental level the WIMP desktop is *object-oriented*. This term needs to be explained, because we are not talking about software engineering principles. By using the desktop metaphor a set of real objects – like folders and paper documents – is transferred to the virtual space on the computer screen where they are represented as icons and windows. However, the illusion on screen is perceived as real. The user can touch the objects (with the mouse) and move them around, open and edit them, file them into folders, or put them in the trash can. As a result, a real interaction with virtual objects emerges.
Alan Kay's formula, «Doing with images makes symbols» depicts the relationships between physical actions, electronic images, and the logical effects [2]. *Doing* is the literal manipulation of things on screen. *Images* are the icons and windows of the graphical user interface. And *symbols* indicate the logical consequences of actions. One example: The act of moving a document icon onto a folder icon with the mouse changes the location of the document within the folder hierarchy. The latter is an abstract concept that just exists in the conceptual model of the system.
Ben Shneiderman calls this kind of interaction with objects on screen «direct manipulation» [6]. User actions directly cause effects on the object and immediate feedback is provided to the user.
In order to convey this intimate relationship between the user and digital objects, the system even emulates the spatial persistence of the real world. In the same way as a real object stays where it is, icons and windows keep their position as long as they are not touched by the user. Breaking this principle would also break the illusion of objects and the idea of ownership over those digital objects.

2.2. Interaction Model for Web-Computing

The interaction model for the web uses the same set of input and output devices as the WIMP desktop systems: keyboard, pointing device, computer display. This is no surprise since web browsers are developed as desktop applications. But they are used to access an entire new universe that has nothing to do with the desktop environment outside the browser window. The WIMP elements themselves are far less important and used in different ways.

Once the user enters the web, a window is no longer the representation of a specific document. It is just a frame for the current page in the path of visited web pages. As soon as the next hyperlink is activated it replaces the window's entire content. The limited influence of websites on browser windows makes it difficult to use windows in a meaningful way, especially given the advent of pop-up blocking. Windows can therefore be neglected as core elements of the interaction model for the web.

In the same sense, icons do not play a role in the web. The web has no objects that need to be represented by icons. This is due to the fact that the desktop metaphor does not cover web interactions, and other object-oriented metaphors have not been utilized for the web. The "icons" that are used on web pages are substitutes for text. Logos, home icons, and arrows to guide the user to next and previous pages do not have the quality of desktop icons that can be directly manipulated.

Menus are not a constitutional property of the web experience either, because websites can not choose to utilize the browser's menu structure for their own purposes. The browser's menu deals mostly with navigation and view management, as well as limited text editing support for text fields and areas that may appear within a page. Some large websites have implemented menu and menubar structures to simplify navigation, but they do not have the rich command structure of desktop applications which typically support content creation.

We are beginning to see pressures for additional menu capabilities with the advent of Ajax technologies and their ability to support highly interactive page content. These menus frequently take the form of a toolbar with some drop down menu-like elements that appears below a page header area.

3. The Duel

Having two interaction models in place – one for the desktop and the other for the web – confuses the user, which results in the inferior usability of the entire system. Two examples illustrate the different approaches taken by the models.

The desktop interaction model uses a single click for selecting an icon or for positioning the text cursor. The only instances where a single click triggers an action is for non-selectable and non-editable items such as menu items and push buttons. Apple's Lisa Desktop Manager set the standard for the double click in 1983. It has been introduced to simplify the activation of the most likely menu command – typically the Open command [5]. Opposed to that, a *single* click is sufficient to *open* a hyperlink in web browsers. Positioning the text cursor is considered less important because web pages cannot be edited like text documents anyway. But text is conceptually something different than objects and control elements. The result is that additional mental effort is required of the user because whether a single or double click is required to open an item depends on the context. Also the developer – regardless of web or desktop application – has to check for more error conditions. For instance if a single clicks activates the command, the second click of a double-click event should not immediately trigger the command a second time or have some other unintended side effects.

The second example is about preventing the user from making fatal mistakes, such as closing a document without saving its content. Desktop applications with a strong binding between the file on the hard drive and the document window display an alert, «The document has been modified. Do you want to save your changes?» Browser windows close without any confirmation – regardless of the state they are in. Web pages can contain important user data that is lost without any chance to recover.

4. Summary

Web browsers give access to a branch of graphical user interfaces that do not use windows, icons, and menus as their main elements. To that extent, websites have non-WIMP graphical user interfaces. Their interaction model is fundamentally different from the desktop model in a way that the knowledge of operation cannot be leveraged from the desktop to the web. Consistency and familiarity can only flourish in the user experience of the combined system of desktop and web if the models are compatible with each other. They are not, because the mode has to be considered in order to predict the effect of the next user action. This extra mental effort causes problems because humans do not pay attention to the surrounding context once they are focused on their activity; they lose sight of the fact that they work in a browser and transfer their experience with desktop applications to build expectation on using web applications. In many cases this is the reason for errors and sometimes even loss of data.

Recent progress in web technology enables the designers to deliver rich and interactive user experiences [4]. E-mail and calendaring are examples for applications that are available for the desktop and the web. This will fuel the conflict between desktop and web even more, as the tasks become more indistinguishable in still different interaction contexts.

5. About the Authors

Frank Ludolph was a member of the Xerox Star team and a co-developer of the Lisa Desktop Manager. He has also worked on Sun Microsystem’s Starfire and Looking Glass projects.

Matthias Müller-Prove played a significant role in designing the user interface of the web editor Adobe GoLive before he joined Sun Microsystems to work on OpenOffice.org in 2002. He was honored with the special award of the Wolfgang von Kempelen Prize 2005 for Computer Science History.

6. References

[1] GILLES, J. / CAILLIAU, R., How the Web was Born. Oxford University Press, 2000

[2] KAY, A. C., Doing With Images Makes Symbols: Communicating with Computers. Apple Computer, Inc., Cupertino, CA, 1987. (Video, 97')

[3] MÜLLER-PROVE, M., Vision and Reality of Hypertext and Graphical User Interfaces. University of Hamburg, Department of Informatics, 2002: Report FBI-HH-B-237/02. http://www.mprove.de/diplom/ (March 2007)

[4] O’REILY, T., What Is Web 2.0 – Design Patterns and Business Models for the Next Generation of Software. 2005 http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html (March 2007)

[5] PERKINS, R. / SMITH KELLER, D. / LUDOLPH, F. E., Inventing the Lisa User Interface. In: Interactions 4(1) p. 40-53, 1997

[6] SHNEIDERMAN, B., Direct manipulation: A Step Beyond Programming Languages. IEEE Computer, 16 (8), p. 57-69. 1983