The Effects of Conceptual Consistency on the End User’s Mental Models of Multiple Applications

John W. Satzinger
Southwest Missouri State University

Consistent user interfaces across applications are thought to facilitate ease of learning and use because a user can draw on existing knowledge when learning any new computer application. Although empirical research has confirmed that transfer of learning occurs when procedural rules for multiple applications are consistent, no research has been reported that examines the effects of consistent versus inconsistent conceptual models across applications on the accuracy of the user’s mental models of the applications. Applications from a variety of sources available to an end user might conform to interface standards for their “look” and “feel;” however, consistent conceptual models across applications (conceptual consistency) are still not assured. This paper reports the results of a laboratory experiment that tested the effects of conceptual consistency across applications on user knowledge, performance, and satisfaction when two integrated cooperative work applications were learned and used by student participants. No differences in performance or satisfaction were found; however, participants learning the inconsistent applications formed more accurate mental models of the applications. Schema theory is used to explain the results, and some implications when users initially learn multiple applications are discussed.

Information system managers are increasingly adopting interface design standards for end-user applications and client-server organizational systems (e.g., Apple, 1987; Berry, 1988; Microsoft, 1992). The primary motivation for interface design standards is ease of learning and use resulting from consistent user interfaces (Shneiderman, 1987; Mayhew, 1992). Consistent user interfaces across applications are thought to facilitate ease of learning and use because a user can draw on existing knowledge when learning any new application (Kieras & Polson, 1985; Bennett, 1986; Polson, 1988). However, multiple applications of a variety of types and from a variety of sources are being integrated on the end user’s desktop, making interface consistency difficult if not impossible to achieve (Nielsen, 1989).

Despite the acceptance of interface consistency guidelines and standards, very little empirical research has been reported that examines the effects of consistent user interfaces across applications. Much of the reported findings have focused on transfer of learning (Kieras & Polson, 1985) from one application to another application of the same type, usually when command languages or procedural rules are inconsistent (Karat et al., 1986; Foltz et al., 1988; Polson, 1988). Only one study reported transfer of learning from one application to a different type of application, but interface consistency was not manipulated (Ziegler et al., 1986). Two studies have investigated the effects of interface consistency when a user interacts with multiple applications or modules (Kellogg, 1987; Satzinger & Olfman, 1998), but again the main interface consistency manipulations were procedural rules across applications.
There is a need for additional research on interface consistency because the user interface is much more than the procedural rules and devices used to issue commands. Moran (1981) defines the user interface as consisting of everything the user comes into contact with while using the application — physically, perceptually, and conceptually. Therefore, the user interface includes the underlying objects and relationships manipulated by the application, which Foley and van Dam (1982) call the conceptual model of the application. For example, a cooperative work application might include objects such as work groups, people, projects, schedules, messages, and documents. Learning this application requires understanding these objects, understanding how they are related, and understanding how they might be manipulated to complete work tasks regardless of the way the application is actually implemented on the computer. The conceptual model is therefore the “heart” of the user interface (Akscyn, Yoder & McCracken, 1988), and the objects, relationships, and possible operations from the conceptual model are reflected throughout the implemented application.

This paper reports the results of a laboratory experiment that tests the effects of consistent versus inconsistent conceptual models across applications when a user learns and uses multiple applications. Two cooperative work applications were developed specifically for the experiment to manipulate the consistency of the conceptual models across applications while controlling the consistency of other interface attributes. The applications were integrated, learned together in one session, and used concurrently to complete the experimental tasks. A multiple application research model based on end-user training research was used to design and control the experiment (Bostrom, Olfman, & Sein, 1990; Satzinger, 1994).

The remaining sections of the paper are organized as follows. Section 2 discusses the user interface, interface consistency, and how the conceptual model of an application is reflected in the user interface. Section 3 draws on schema theory to describe how consistent conceptual models across applications might affect user knowledge, user performance, and user attitudes toward the applications. The hypotheses for the experiment are then presented. Section 4 describes the research method, and Section 5 presents the results. Section 6 discusses the results, including implications for managers and for researchers concerned with understanding the effects of consistency when a user learns and uses multiple applications.

**The User Interface and Interface Consistency**

The user interface and interface consistency are complex constructs (Moran, 1981; Grudin, 1989). Generally accepted models of the user interface are based on Foley and van Dam's (1982) language model of human-computer interaction (c.f., Bennett, 1986; Gerlach & Kuo, 1991; Marcus & van Dam, 1991). In the language model, the interface consists of a conceptual model and two separate languages, an action language and a presentation language, used by the user and the computer to communicate. Human-computer interaction is therefore modeled as a dialog.

The conceptual model of an application includes the intrinsic objects from the task environment of the user, the relationships among the objects, and the operations possible to manipulate the objects. The conceptual model is a logical model which does not describe how the application is physically implemented, although it is sometimes confused with the designer's conceptual model, which usually does include a brief description of the implementation (Moran, 1981). Figure 1 shows a written description of a conceptual model for an application used in the experiment reported in this paper. Objects, relationships, and possible operations are apparent, but no references are made to the way the user would actually interact with the computer.

The rest of the user interface involves the two separate languages that are used to communicate about the objects in the conceptual model: one used by the user to communicate to the computer (the action language) and one used by the computer to communicate to the user (the presentation language). The two separate languages are designed based on the communications requirements that follow from the conceptual model. The action language is used to tell the computer which operations to perform, and the presentation language is used to ask the user about the requested objects and operations and to provide the resulting information.

**NUCLEUS** is used to create work groups and the information they use. Work groups consist of members selected from lists of people in the system described by a work group name. An information structure is created to store various types of work group information. Any member can create an information structure and assign access to other work group members. Work group members then use the information structure by either browsing through it or typing in additional information. Any person can review work groups to see a summary of the work group. Work groups and information structures can be modified or deleted as needed.

---

**Figure 1: Description of a Conceptual Model**
Related Content

Analysis of User Involvement and Participation on the Quality of IS Planning Projects: An Exploratory Study
[www.igi-global.com/article/analysis-user-involvement-participation-quality/4148?camid=4v1a](www.igi-global.com/article/analysis-user-involvement-participation-quality/4148?camid=4v1a)

A Hybrid Case Based Reasoning Model for Classification in Internet of Things (IoT) Environment

Developing User Profiles for Interactive Online Products in Practice
[www.igi-global.com/chapter/developing-user-profiles-interactive-online/76796?camid=4v1a](www.igi-global.com/chapter/developing-user-profiles-interactive-online/76796?camid=4v1a)

User Acceptance of Voice Recognition Technology: An Empirical Extension of the Technology Acceptance Model
[www.igi-global.com/chapter/user-acceptance-voice-recognition-technology/18233?camid=4v1a](www.igi-global.com/chapter/user-acceptance-voice-recognition-technology/18233?camid=4v1a)