



# *Fundamentals of Exceptional Interface Design*

## **Dr. Don Levan**

President, Vanguard Custom Software  
NY State Licensed Clinical Psychologist  
FileMaker Certified Developer

## *The Business Problem*

The design of a successful software application rests on three pillars: the developer's understanding of the business and personal goals the program is to address, the design of the user interface and interactions, and the quality of the programming and data structures.

## *Impact of a Successful User Interface*

An application with a well-designed user interface can help a person achieve their personal and business goals, feel satisfaction and effectiveness, and at times, enter a state of flow. Flow is an experience of full absorption during which time slows, distractions fall away, and inner worries and doubts are silent (Csikszentmihalyi, 1990).

Research has also shown that a well-designed user interface fosters a positive emotional attachment to the product and brand loyalty. "Attractive things have been found to make people feel better (which facilitates creative thought and the ability to learn), and are judged to work better" (Norman, 2005).

## *Impact of a Problematic User Interface*

Poorly designed interfaces can cause distraction, feelings of anxiety, frustration, and helplessness (which narrow thought and diminish ability to learn), physical injury, and a diminished sense of self.

## *Goals of this Presentation*

This seminar will provide an understanding of how to design the right tool, and how to make the resulting application easy to learn and efficient to use.

*Successful user interfaces are those that are easy to learn, and pleasurable and efficient to use.*

*Meeting these goals requires that you understand the needs and abilities of the people who will use your software.*

## Key Points

- Identify as clearly as you can the goals of both the people who are funding the application and those who will use the application.
- Support the creation of an accurate Mental Model.
- Make all possible actions visible.
- Exploit natural and social constraints and mappings.
- Provide clear feedback about the results of every action.
- Facilitate quick scanning.
- Use clear language.
- Minimize disruptions.
- Provide feedback within the locus of attention.
- Optimize to support efficient habits.
- Make it modeless and monotonous.
- Do not require memorization.
- Provide cues to facilitate recognition.
- Put more frequently used controls closer.
- Less frequently used or dangerous controls should be placed farther away.

## Design the Right Tool

Every application, interface, or interface change should begin with an understanding of the goals being addressed and the problems being solved (Goodwin, 2009).

Start by identifying and interviewing the stakeholders. Ask open ended questions to determine: the overriding goals for the project, the goals of each workgroup or user type, the existing solution, problems with the existing solution, common task scenarios faced by each user type, and requirements derived from the task scenarios.

Develop the design by iterating through a process of sketch, prototype, feedback, and refinement. Establish a design framework which addresses the larger goals and scenarios. Then focus on the details.

## Make it Easy to Learn

### Support the Development of a Mental Model

People create internal mental models to explain the workings of the world, other people, and events (Norman, 1989). Based on what we can see and the responses to what we do, these models help us adapt to new situations and use the things we encounter.

Help people learn your application by making all possible actions visible and providing clear feedback about the results of every action.

Improve the visibility of an interface by making all buttons look like buttons, placing labels near the fields they identify, and ensuring that the user can differentiate between labels, editable fields and non-editable fields, buttons, and other controls. Utilize natural and cultural mappings and constraints to help users immediately understand what every control does.

Help the user understand by providing feedback in response to their actions. Feedback is most important when something is happening that the user cannot see.

### Facilitate Scanning

When forced to make a decision in a novel situation, people do not first gather all available evidence and consider all possible options. Instead, they quickly scan (or satisfice) to find the information they need and do the first thing that does not appear to have problems (Krug, 2006).

Help the user scan by creating a visual hierarchy. Make the most important things stand out, group related elements, align objects

on a grid, and remove excess visual “noise.” Consider each element and remove anything extraneous. Some items should just be deleted, others can be moved off the main interface and presented as the user needs or requests them.

Make your language as clear as possible. Omit needless words.

Use icons sparingly, and then only to represent nouns (actions are too difficult to represent in a tiny image). Use text instead, or pair icons with text.

## *Make it Efficient to Use*

### **Do not interrupt**

Humans have only one Locus of Attention, or can only attend to one thing at a time. The interface must not become a source of interference. The application must work no matter how narrowly the user is focusing (Raskin, 2001).

Minimize disruptions by eliminating unnecessary prompts and dialogs. Before using a custom dialog, consider whether there is a way to eliminate the interruption. Instead of asking the user to confirm their decision, design the system so that all actions can be reversed (Raskin, 2001).

If a user is absorbed in a task, they will likely not see status messages added to the screen. You can only ensure that a person receives feedback if it is provided within the area of their focus. The best way to know what the user is seeing is by conducting user testing. This can be very brief and consists of watching a person use your interface while performing application tasks.

### **Encourage the Formation of Habits**

Habits are crucial to human development, as they allow us to perform complex actions without conscious thought. If you repeat an action more than a few times, that action will become habitual and you will be able to perform it without thinking (Raskin, 2000). Habits can become so strong that a person can lose the ability to control them.

Well-designed interfaces encourage habits which allow a person to work faster and enter a flow state. Poorly designed interfaces engender habits which can slow a person down or cause them to do things they do not intend.

Habits are the reason that confirmation dialogs are often useless. Any dialog that repeatedly presents the same text will soon become habitual (and automatically dismissed).

To support good habits, be consistent in the look, feel,

placement, and function of elements. Create a logical tab order on each layout, and add frequently used scripts to the Script Menu. Be careful not to change the order of scripts once set.

Be careful not to introduce modes in the functioning of elements. When a button, widget, or keyboard shortcut functions differently depending on the state of the system, it is said to have different modes (Raskin, 2000). Modal interfaces are a major source of confusion, and at times can cause damage to a user’s work.

If you need the same element or button to perform different actions, create a Quasimode, or a user maintained mode (Raskin, 2000). Strive to create interfaces that are “monotonous”, or have “one and only one” way to achieve any desired result.

### **Provide Cues for Recognition**

Human beings have two major types of memory: short-term memory of that which has just happened, and long-term memory of past people, places, events, and thoughts. Short-term memory is limited, fragile, continuously replaced. If an item has importance or is repeatedly rehearsed, it may be transferred to long-term memory.

Items in long-term memory must be retrieved or “remembered” to be used. Retrieval happens either through recall or recognition. Recall from long-term memory tends to be slow and error prone. Recognition is easier for most people than recall because all the knowledge they need is present “in the world” (Norman, 1989).

To help users recognize the parts of your interface they have already learned, be consistent in the placement and function of elements, make all possible actions visible, use natural mappings and constraints, and help the user regain context by tracking their state and returning them upon startup to the last record and layout visited. In addition, the limits on memory suggest that it is best not to present information in a custom dialog or window that the user has to remember and use elsewhere.

### **Don’t require great precision**

Fitts’ Law states that the time it takes to acquire a target on the screen (i.e., a button, field, or menu) is a function of the distance to travel and the size of the target.

Use Fitts’ Law to your advantage by making more frequently used controls and buttons larger and closer. Less frequently used controls, and those which you do not want the user to accidentally activate should be smaller and placed farther away.

## Interaction Design Resources and References

### Books, Webpages, and Articles

Apple Human Interface Guidelines, <http://developer.apple.com/mac/library/documentation/UserExperience/Conceptual/AppleHIGuidelines/XHIGIntro/XHIGIntro.html>

Cooper, Alan. (2004). *The Inmates are Running the Asylum, Why High-Tech Products Drive us Crazy and How to restore the Sanity*. SAMS.

Cooper, A., Reinmann, R., and Cronin, D. (2007). *About Face 3: The Essentials of Interaction Design*. Wiley.

Goodwin, K. (2007). *Designing for the Digital Age*. Wiley.

Krug, Steve. (2006). *Don't Make Me Think! A Common Sense Approach to Web Usability, Second Edition*. New Riders.

Levan, Donald M. (2007). *Psychology of Effective FileMaker Pro Solution Design* FileMaker Advisor Magazine. December/January 2008, p. 18.

Lidwell, W., Holden, K., Butler, J. (2003). *Universal Principles of Design*. Rockport Publishers, Inc.

Norman, Donald. (1989). *The Design of Everyday Things*. Currency Doubleday.

Raskin, Jef. (2000). *The Humane Interface, New Directions in Designing Interactive Systems*. Addison-Wesley Professional.

Tufte, Edward. (2001). *The Visual Display of Quantitative Information, 2nd Edition*. Graphic Press.

Tidwell, Jenifer. (2006). *Designing Interfaces: Patterns of Effective Interaction Design*. O'Reilly.

### The "Craft of FileMaker" Workshop

Interested in learning more? Join Don Levan and Ernest Koe (of the Proof Group) in a three day seminar on the craft of FileMaker Development.

Hosted in New York City in October, 2010, this workshop will feature a combination of lectures and small group experiences to help you dramatically improve your abilities to understand the problems to be solved and design and code effective solutions.

For more information send an email to: [craft\\_seminar@vanguardcs.net](mailto:craft_seminar@vanguardcs.net)



**VANGUARD**  
CUSTOM SOFTWARE

**V**anguard Custom Software was founded by a psychologist with a passion for solving problems.

Our mission is to create simple applications which solve complex problems. We incorporate software development best practices, interaction design, and cognitive psychology to create data driven applications and websites that are efficient for business and easy for people. We employ only FileMaker Certified Developers.

Please contact us for assistance on your next project.

**Don Levan, Psy.D.**

President, Vanguard Custom Software  
NY State Licensed Clinical Psychologist  
FileMaker Certified Developer

917 842-2911  
[www.vanguardcs.net](http://www.vanguardcs.net)  
[don@vanguardcs.net](mailto:don@vanguardcs.net)