# Human Interface Note

**Note #3** **Dueling Metaphors: the Desktop & HyperCard**

Written by: Tom Erickson January 1990
(Supersedes Human Interface Update #14)

Discussion of the differences between the metaphors of the Desktop and HyperCard.

Metaphors help users form a coherent model of an application's human interface. In an interface with a well-chosen metaphor like the Desktop, users find it easy to predict the results of an action or to figure out which action produces a desired result.

While a single, clear metaphor aids human-computer communication, mixing metaphors may cause significant problems. Even two metaphors which work well separately may interfere with one another when they are used within the same human interface.

The Desktop and HyperCard metaphors can interfere with one another. This document describes the conditions under which such interference can occur, and what can be done to avoid it.

In the Desktop metaphor users do things by pressing rounded-rectangle buttons, choosing menu commands, and double-clicking (opening) icons. Each type of control object has a distinct, carefully-defined appearance, as well as a different method of access. You can tell how to use a control object just by looking at it.

In HyperCard, buttons are the principle control objects, but in HyperCard, a button can look like anything—an icon, an item in a list, a push button, a menu item. Since a click is the only way of starting an action, the appearance of a button is less important than in the Desktop: the user knows that all HyperCard control objects respond to a single click.

When elements of the Desktop and HyperCard metaphors are combined, confusion may result. In an interface with a mixed metaphor, a user can no longer predict the result of clicking an item in a list or clicking on an icon. Does a click select the object, as in the Desktop metaphor, or does it launch an action, as in the HyperCard metaphor? Clicking on an icon to select it and having it launch an action because it's acting like a HyperCard button is—at the very best—disconcerting. Such unpredictability destroys the comfortable feel that is essential to a good human interface[ Begin Footnote ] --- Also see Chapter 1 of *Human Interface Guidelines: The Apple Desktop Interface* (Addison-Wesley, 1987)—in particular, the principles of Consistency and Perceived Stability.--- [ End Footnote ]. Users confronted with such unpredictability are likely to become lost, confused, and unhappy with your product.

Do **not** mix the Desktop and HyperCard metaphors. If you're writing a HyperCard stack, don't include icon-like buttons that must be double-clicked. If you're writing a Desktop application, don't include (to take a real example) a house icon that takes the user somewhere when it's clicked, like the Home Card button in HyperCard. Desktop applications should not contain HyperCard-like interface elements.

The most important point is this: It should always be obvious whether the user is in a Desktop application or a HyperCard stack. And this means obvious at a glance; users should not have to read text or remember whether an application or a stack was launched. If the context is obvious, the user knows the result of a click—without having to think about it.