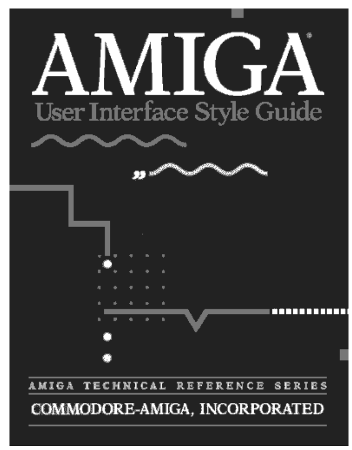
From the Desk of Jeff Scherb

The Amiga is receiving more attention from professional users these days, which makes the user interface issue even more important. Amiga applications are now competing with Mac and Windows applications. The vast majority of applications for those platforms conform to guidelines initially set down by the respective manufacturer. As a result, it is relatively easy for a Windows spreadsheet user to learn how to use a Windows word processor because the user knows how to operate the user interface.   The commands and menus are already familiar.



The new *Amiga User Interface Style Guide* describes the standards and conventions for the three Amiga user interfaces -- the GUI, the CLI, and ARexx. This book took over a year to develop and is the result of months of research into existing user interface conventions on a variety of platforms. CATS coordinated with several key developers to help form a set of practical and reasonable guidelines for the Amiga user interface.

You might wonder what the drawbacks are for not following user interface standards. No, the Interface Police are not going to show up at your door one morning -- the drawbacks are much more subtle than that. Not following commonly accepted practices makes an application's user interface less predictable, making it harder for the user to learn how to use the application. If the application's interface is too unpredictable, within a few minutes the user may give up trying to use the application. This leads to lower market acceptance -- one penalty of not following user interface standards.

Another disadvantage of not following the user interface standard is that it increases the amount of code a developer has to maintain. Release 2.0 provides some standard GUI elements through standard system libraries (such as the standard requesters of *asl.library*). Not using these standard libraries will produce unconventional (and possibly confusing) GUI elements that you will have to update and correct yourself, rather than system standard GUI elements that Commodore maintains as part of the Amiga system software.

The Style Guide is part of the new Release 2.0 ROM Kernel Reference series published by Addison-Wesley. The book should be available by the time you read this. Developers will be able to get the Style Guide through CATS. The book should also be available in bookstores. At a retail price of $21.95, everyone involved in creating applications for the Amiga should have one.

CATS has several other development aids in the works. We're working on a user interface design, protoyping, and code generation tool which should be available by DevCon in September. Currently known as *AppBuilder*, this tool allows you to quickly build an interface which follows the guidelines set forth in the Style Guide. After you're satisfied with the look and feel of the interface, *AppBuilder* can generate C or assembler source code. The code can be compiled and linked with the *AppShell*, which provides standard event processing code. With the combination of *AppBuilder* and *AppShell*, you'll be able to spend your development time on the functionality of your application instead of on gadget creation and housekeeping.

The Amiga is growing up. It is finding its way into more professional markets, and increasingly competing with and being compared to the Mac OS and Windows. I encourage all of you to review your applications in light of the information in the Guide and plan for interface improvements in your next release.

-JS