



Using AppleScript Studio to improve Illustrator.

Mark Chapman fixes a clipping problem using the interapplication communication abilities of AppleScript Studio and wraps his problem-solver in a flexible and informative interface.

By Barbara Gibson

“Now that we have AppleScript Studio,” says AppleScript expert Mark Chapman of Chapman Consulting Group, “scripters can build beautiful Mac OS X-consistent interfaces that guide a user’s experience in running the script — something that’s not possible using AppleScript alone.”

Chapman, known for his work using AppleScript to automate workflows and repetitive tasks for clients in graphics, prepress, multimedia and elsewhere, is currently using AppleScript Studio to create an application that fixes a bug in certain types of Adobe Illustrator files.

The Built-in Correction

“If you’re a designer and you’re placing Illustrator files into layout programs such as Adobe InDesign or QuarkXPress,” Chapman explains, “the edges of the Illustrator file will often be clipped. To compensate for the bug, people would have to manually create a new box in Illustrator slightly larger than the group of objects in the document and insert the box into a background layer. So when an Illustrator file was placed into InDesign or Quark, the new box would prevent the clipping of the artwork. It was a tedious thing to have to do, completely unrelated to graphics work.”

In Chapman’s application, AppleScript takes a look at the Illustrator document, automatically creates a new box slightly larger than the group of existing visible objects and places the box in the background as a new layer. “It doesn’t affect printing,” says Chapman. “It just forces Illustrator to make a larger bounding box when the file is exported.”

Guiding the User

The AppleScript Studio interface, Chapman says, also guides the user’s experience in working with the script in a way that’s not possible with the script alone. “For instance,” he explains, “with AppleScript Studio you can provide more information. You can have a help window that pops down, menus, buttons, visually display database connectivity and other interface items that can all contribute to a clearer interface.

“The AppleScript Studio interface I’m building is a simple one, but it still gives users different options about how they want to handle a file, where they want to save it, whether they want to batch process groups of files and so on. In AppleScript, you were pretty much limited to a long list of dialog boxes, or you had to go to a third-party interface



The Final Interface. The application automatically creates a new box slightly larger than the group of existing visible objects and places the box in the background as a new layer.

Guiding the Developer

“AppleScript Studio is wonderful for creating interfaces,” Chapman says. “It provides Aqua elements and guidelines so you can build interfaces that are consistent with the OS X look and feel. Helpers are built right into the Interface Builder application of AppleScript Studio. And Studio is great for building this kind of application because it lets me create any number of interface designs — designs that I can review with a client before completing the application.

“Once we determine the best design for an interface, I can wire it to the tasks I require by attaching scripting code to the various objects and possible actions in the interface.” The finished product, Chapman says, will be indistinguishable from other applications written in any number of languages. “Much of the Mac OS X Cocoa application functionality is also built right in by AppleScript Studio. I get these additional features without having to write a line of code.”

Improving Usefulness

Chapman is beginning to build older scripts into AppleScript Studio applications for clients who are migrating to Mac OS X. “AppleScript Studio can expand the usefulness of our older code,” he says, “both by adding an interface to solutions previously lacking one and by expanding the feature set of applications in the stable, forward-looking Mac OS X environment.

“In AppleScript Studio, you can build applications with

builder, which isn't available on Mac OS X."

Learning AppleScript Studio

The first step to learning AppleScript Studio, says Chapman, is to learn AppleScript. "And there really isn't a better way than to take a two or three-day class from [TECSoft](#). That will give you a solid basis from which to move forward." CD and online versions of this training are available as well.

Also very helpful are books such as Danny Goodman's "AppleScript Handbook" (included with TECSoft's training); "AppleScript in a Nutshell" by Bruce W. Perry; and books on scripting for individual applications, such as "Adobe Illustrator Scripting" by Ethan Wilde, and "AppleScripting InDesign" and "AppleScripting QuarkXPress," both by Shirley Hopkins.

Once you understand AppleScript, you can enter the AppleScript Studio world. "AppleScript Studio has several layers of complexity. Part of being in the AppleScript Studio world is learning to focus on areas of interest to you, because there's a lot in Project Builder and Interface Builder that will be useful only to hard-core programmers."

When exploring AppleScript Studio, Chapman recommends that you:

- Visit the [AppleScript Studio](#) website
- Work through the tutorial in [Building Applications with AppleScript Studio](#).
- Join the [discussion list](#) for scripters and developers who are using AppleScript Studio. The list focuses on the use of the AppleScriptKit classes and on the use of Project Builder and Interface Builder to create AppleScript applications.

technology fully transparent to the user; users don't know that the application was written in AppleScript or Objective C. You also can mix technologies in AppleScript Studio and take what's best in each and use it within the Mac OS X environment of Interface Builder and Project Builder.

"That's one of the big advantages of AppleScript Studio," Chapman adds. "You not only get this interface capability but you get a lot of development resources that you wouldn't otherwise have. All the Cocoa support is right there, already in Studio. It saves you a lot of programming and development, because you're tapping into an engine that already provides everything you need."

More Power, More Accessible

Despite his reputation as an AppleScript pro, Chapman doesn't see himself as a techie. "I just like solving problems with great solutions," he says simply. "I was a philosophy major and a theology graduate student and I do AppleScript development. You don't have to have a technical background to do AppleScript or AppleScript Studio. Like a lot of things Apple does, Apple is making more powerful tools more accessible to more people."

Coming Soon

Chapman's application will be available soon on the [Adobe Studio Exchange](#).

Chapman Consulting Group

serves clients in graphics, prepress, multimedia, digital video, DVD creation and the Internet. The company also develops small and large scale custom AppleScript solutions to automate workflows and publishing processes that rely on everyday production tools such as InDesign, QuarkXPress, FileMaker, PhotoShop, Illustrator and others.

[Site Map](#) | [Search Tips](#)

Visit the Apple Store [online](#) or at [retail](#) locations.
1-800-MY-APPLE

[Contact Us](#) | [Privacy Policy](#)
Copyright © 2002 Apple Computer, Inc. [All rights reserved.](#)