

Tools for Interface Crafters: Static and Dynamic Palettes; Tools for Interface Crafters: Static and Dynamic Palettes

A palette is a special display that holds one or more reusable objects. You can drag these objects from the palette to your application's interface. There are two types of palettes: static and dynamic. To the user, they seem identical, but the differences are many.

Static palettes are built as a project and have code defining their objects; dynamic palettes include no special code—they're unique configurations of (mostly) standard OpenStep objects. Consequently, static palettes must be compiled, but you can create dynamic palettes on the fly, without writing and compiling code. Objects on static palettes can have inspectors and editors, which dynamic-palette objects cannot have.

Creating static palettes (and their inspectors and editors) is a more complex process than creating dynamic palettes, but the resulting product has more value added to it. For example, if you want to store a button that has the title OK, you use a dynamic palette because the change involves only the Interface Builder Inspector panel. However, if you want to store a custom subclass of NSButton, you use a static palette. A static palette can store both the button and your custom code.

Dynamic palettes are a great convenience. You can save collections of your objects, with or without their interconnections, to a dynamic palette at any time. You can save dynamic palettes and store them in the file system, just as you do with the traditional compiled palette. You can remove the palette from the Palette window and, when you need it again, just load it back into Interface Builder.

The possible practical uses of dynamic palettes are numerous. You can use them to:

- Store collections of often-used view objects configured with specific sizes and other attributes.
- Hold windows and panels that are replicated in your projects (such as Infopanel).

- Store versions of interfaces.
- Keep interconnected objects as a template that you can later use as-is or modify for particular circumstances. For instance, you could store a group of text fields and their delegate, or a set of controls and their connections to a controller object.

You can also use dynamic palettes for prototyping and group work. For instance, you could design an interface or a part of an interface, store the objects on a dynamic palette, and then mail the palette file to all interested parties.