Porting macOS Applications to Linux with GNUstep

An introduction to GNUstep and interview with Gregory Casamento, the project's lead maintainer.

By Petros Koutoupis

In the field of software development, people often find themselves writing code across multiple platforms. When the time comes to port that code and compile it, they keep their fingers crossed and hope for the best. For those who are either coming from an Apple ecosystem or need to develop code for Apple devices, a set of open-source libraries exists that are fully capable of developing or maintaining such code, even when not doing so within macOS.

History Lesson

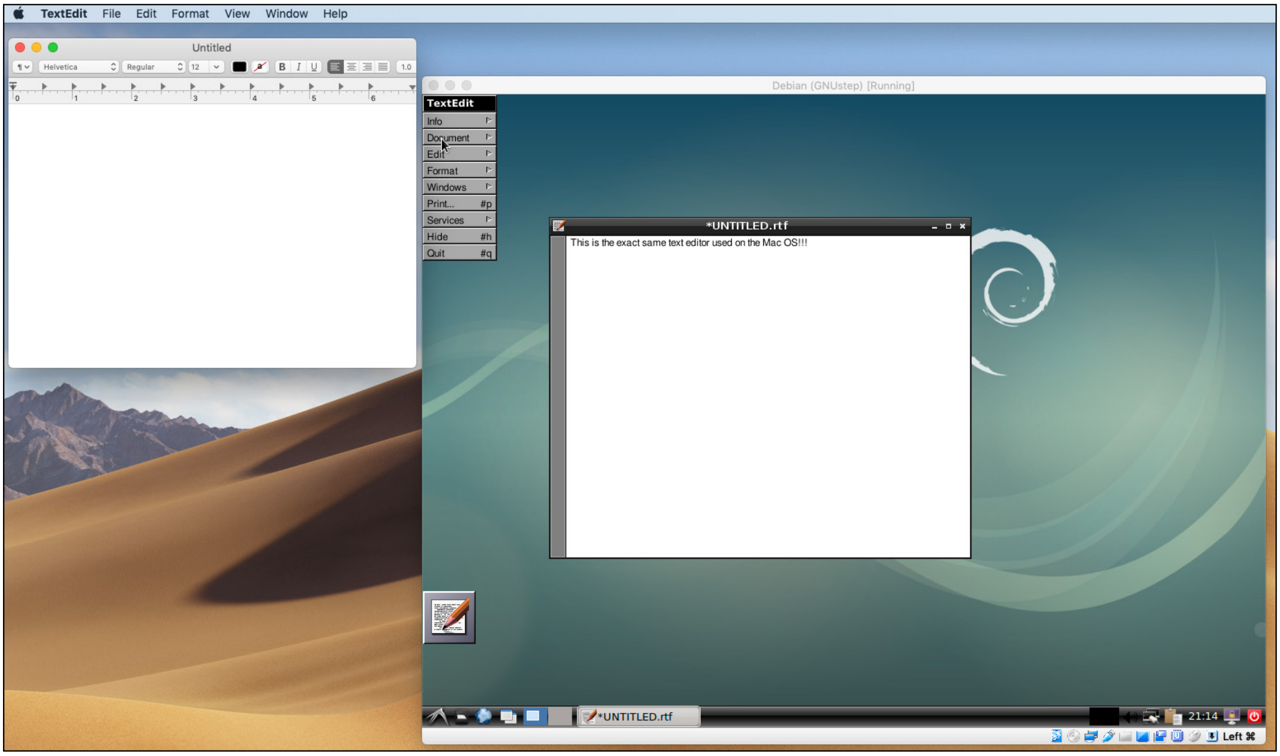
Once upon a time in the 1980s, after founding the then successful Apple Computer company, Steve Jobs was driven out and went on to find a new one. NeXT, Inc., was born in 1985 and was building computers to cater to the growing higher-education market. The foundation of the NeXTSTEP operating system was built entirely on top of UNIX. Its core was a hybrid Mach and 4.3BSD Unix kernel (labeled as the XNU or *X is not Unix*), and everything above it was entirely object-oriented and written in the Objective-C language.

In 1996, Apple announced the acquisition of NeXT, Steve Jobs came back to Apple and the NeXTSTEP ecosystem formed the basis for the newly developed Mac OS X operating system. This same NeXTSTEP code base would continue to be used in later Apple technologies including the iOS, watchOS, tvOS and more.

Fun fact: some of the applications bundled into macOS are descendants of NeXTSTEP applications. Those include TextEdit, Mail and Chess and more.

The Open-Source Effort

Since the days preceding the Apple acquisition, there have been attempts to open-source the NeXTSTEP application programming interface (API). The earliest example was OpenStep. That later would be rebranded as the Cocoa API (post Apple acquisition), and it separated the underlying operating system from the higher-level object libraries. Its primary goal was to offer a NeXTSTEP-like environment for non-NeXSTEP operating systems. The most important thing to



This is the exact same text editor used on the Mac OS!!!

Figure 1. The TextEdit application on both the macOS and in Linux as it was ported with GNUstep.

note here is that while NeXTSTEP offers an entire operating system, OpenStep was nothing more than an API and was ported to operating systems that included Sun Microsystem's Solaris and Microsoft's Windows NT.

Early on, a free software implementation of this API was developed. It was called GNUstep, and to this day, it continues to maintain compatibility with the latest Cocoa (OpenStep) libraries. GNUstep is *not* an operating system, a desktop environment or a window manager. It is only a set of libraries (with development tools to enable a cross-platform development environment) that adhere to the Cocoa API and is licensed under the GNU Lesser General Public License (LGPL) version 3, while the standalone utilities are licensed under the GNU General Public License (GPL) version 3.

Drilling into the Details—an Interview with GNUstep's Lead Maintainer

But, what exactly is GNUstep? I took the opportunity to reach out to the project's lead maintainer, Gregory Casamento.

Petros Koutoupis: Please introduce yourself to our readers and explain your role with the GNUstep project.

Gregory Casamento: I joined GNUstep in 1999 and made some contributions to the gui/AppKit framework on GNUstep. I am the main author of the Gorm (Interface Builder Equivalent) for GNUstep. I became lead maintainer in 2012 and have been since then. My role as maintainer is to encourage people to use GNUstep and to be the main face of the project to the public.

Petros: Is it pronounced "ga-new-step" or "new-step"?

Greg: Pronounced phonetically, it's Ga-new-step.

Petros: So, what exactly is GNUstep?

Greg: GNUstep is a cross-platform framework that can be used on Windows or any POSIX-compliant platform to build your own applications, or alternatively, to port applications from Cocoa on macOS to those platforms.

Petros: Why use GNUstep?

Greg: If you're writing a macOS/Cocoa-based application, GNUstep allows you to maintain one codebase instead of multiple ones, and it allows you to use the native environment rather than some other non-Mac-specific platform, which might not allow you to do everything you might want.

Petros: Can you provide our readers with some common examples? Maybe even mention some of the applications that have been ported over?

Greg: Sure. One such application is called EggPlant, which is made by the nice people at https://eggplant.io. Also there are a number of applications that use GNUstep that are in common use on macOS. Another is PikoPixel. There was, for a while, a company known as Apportable, which was using GNUstep's foundation layer to help port apps over to Android. Their framework is now used by another company known as PocketGems to do much the same thing.

Petros: It's funny that you mention Android. Many folks coming from the world of Apple also are interested in iOS development, and somewhere I read that there may exist some overlap.

Greg: GNUstep's Foundation does have many of the classes and methods from iOS. We also are trying to build our own UIKit implementation. There are currently a number of companies using the Foundation layer of GNUstep to build their apps on non-iOS devices, such as Android.

Petros: Which version of the Cocoa API is it compatible with?

Greg: This is a complicated question. Some of the more used portions of the API are up to spec with about 10.12 or so, but others may have functionality only up to 10.6. It depends on how heavily those classes are used by the companies and people who use GNUstep. You can always help us and join the project and help it get more complete. It's important to remember that we have only about six or seven active developers, and we are building an API that is maintained by a multibillion dollar company.

Petros: Where can our readers learn more about GNUstep?

Greg: You can learn more by following me on Twitter (@bheron), or visit http://www.gnustep.org, http://wiki.gnustep.org or our project page at https://github.com/gnustep.

Drilling into the Details—the Tools of the Trade

How does one begin working with GNUstep on Linux? As with most packages on modern distributions, using that distro's package manager should suffice. For instance, on Debian or Ubuntu, it would look something like this:

```
$ sudo apt install gnustep gnustep-devel
```

This alone will install the libraries, related environmental scripts, development tools (see below) and even some of the ported applications.

You next need to make sure that everything works. Copy the following simple and internet-common piece of code into a file named hello.m:

```
#import <Foundation/Foundation.h>

int main (int argc, const char * argv[])
{
   NSAutoreleasePool * pool = [[NSAutoreleasePool alloc] init];

   NSLog (@"hello world");
```

```
    [pool drain];
    return 0;
}
```

Note that Objective-C filenames use a .m extension.

Before you compile this simple piece of Objective-C code, you'll need to set up the GNUstep environment:

```
$ . /usr/share/GNUstep/Makefiles/GNUstep.sh
```

Compile the file into an executable binary (and yes, that is a long compilation command):

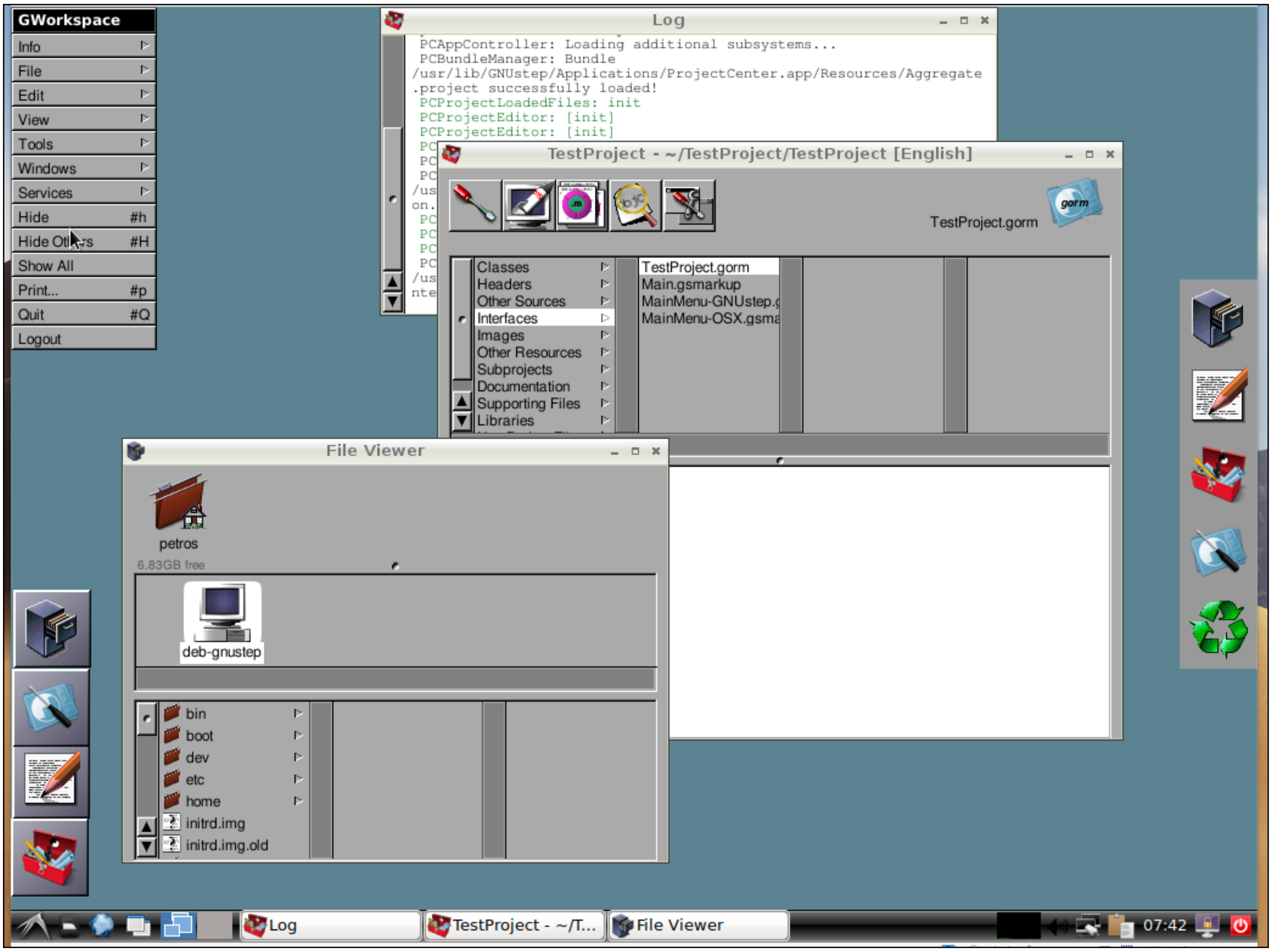
```
$ gcc hello.m 'gnustep-config --objc-flags'
 ↪-I/usr/include/GNUstep/ -L/lib64/ -lobjc
 ↪-L/usr/lib64/GNUstep/ -lgnustep-base
 ↪-fconstant-string-class=NSConstantString -o hello
```

Executing the binary yields the following output:

```
$ ./hello
2019-04-07 15:55:55.029 hello[6796:6796] hello world
```

To simplify the development process, the GNUstep project features an object-oriented IDE (Integrated Development Environment) called Project Center. Using it, developers are able to write all sorts of applications, tools, libraries and more.

The IDE integrates with Gorm, a nice and easy-to-use graphical utility that allows developers to create graphical applications quickly. It reminds me a lot of my days working with Borland C++ Builder (yes, I have been doing this for quite a while). Using a mouse, Gorm relies on drag-and-drop to position and resize objects, such as menus, buttons, lists, tables and so on. You then can connect



```
 PCAppController: Loading additional subsystems...
 PCBundleManager: Bundle
/usr/lib/GNUstep/Applications/ProjectCenter.app/Resources/Aggregate
.project successfully loaded!
 PCProjectLoadedFiles: init
 PCProjectEditor: [init]
 PCProjectEditor: [init]
 PC
 PC
 PC
/us
on.
 PC
 PC
 PC
 PC
/us
nte
```

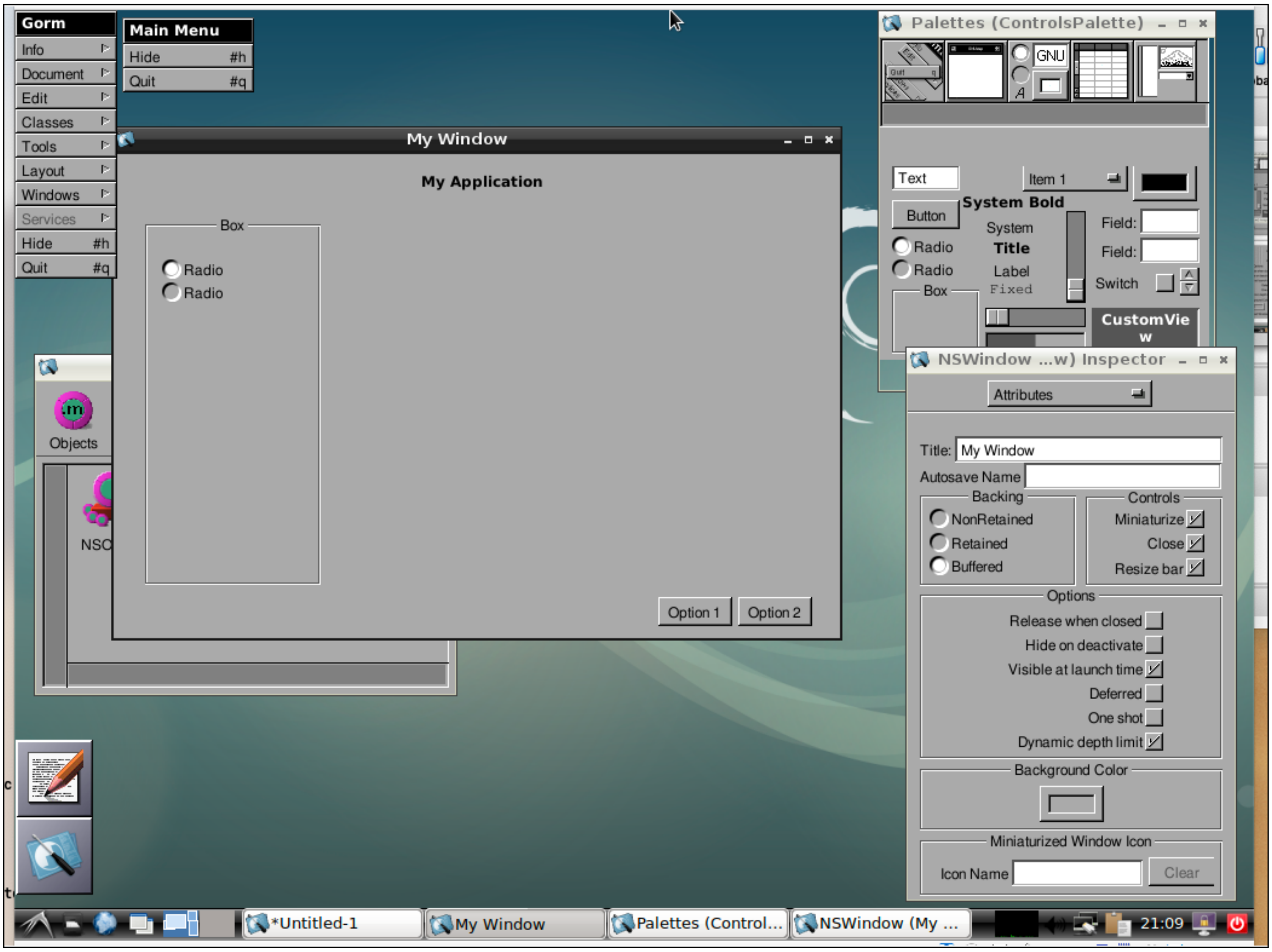
Figure 2. The ProjectCenter IDE

those objects to functions that you write for enhanced functionality.

I know what you're thinking, and no, you don't need to install a special desktop environment to run these graphical applications. Anything you write will work with GNOME, KDE or any other X11-based window manager. To learn more about ProjectCenter, visit its official project page, and go here to learn more about Gorm. Detailed guides and tutorials are available on the project site.

Summary

Your macOS code does not have to die after you switch to Linux. It can live on



Title: My Window
Autosave Name
Backing
○ NonRetained
○ Retained
○ Buffered
Controls
Miniaturize ☑
Close ☑
Resize bar ☑
Options
Release when closed □
Hide on deactivate □
Visible at launch time ☑
Deferred □
One shot □
Dynamic depth limit ☑
Background Color
Miniaturized Window Icon
Icon Name
Clear

Figure 3. The Gorm Graphical Utility

and continue to be supported in both ecosystems. And, even if you don't have the experience to port it to Linux yourself, a small community of dedicated and talented individuals exists in the GNUstep project, who are more than willing to make that happen. ■

Petros Koutoupis, *LJ* Editor at Large, is currently a senior performance software engineer at Cray for its Lustre High Performance File System division. He is also the creator and maintainer of the RapidDisk Project. Petros has worked in the data storage industry for well over a decade and has helped pioneer the many technologies unleashed in the wild today.

Resources

- GNUstep.org
- GNUstep Wiki
- GNUstep Project Page on GitHub
- Gorm

Send comments or feedback
via http://www.linuxjournal.com/contact
or email ljeditor@linuxjournal.com.