

McNits: Mike's Palm Programming Nitpicks, Part 4

by Mike McCollister

- Originally published circa April 2003 on the “Handheld Computing Developer” website.
- This article can be found at <https://MikeMcCollister.com/palm/articles/mcnits4/>.
- As this article are quite dated, most of the links will be invalid.

Introduction

Do you remember your teenage years, the time of your life that “awkward” was more than a word, it was a way of life? If you can program a computer, chances are you understand perfectly. This is where Palm OS is right now. It has just started its adolescence, the time that it is trying new things, vying for acceptance, getting into trouble, and trying to find out what it wants to do in the future. It was once a cute, darling little operating system that everyone liked (who ever saw an ugly baby or at least admitted that to the mother?) but it is now growing up.

Palm OS 5 is a transition operating system. On the surface it looks like any other Palm OS device, but the internals have changed drastically. Included is support for a whole new class of ARM processors, a high-resolution API, sampled sound and many other features too numerous to list here. While these are all well and good, there are still some drawbacks to OS 5 such as lack of multi-tasking and the same old user interface just to name a few. The next major release of Palm OS (rumored to be named Palm OS 6) promises to improve upon these issues.

During this transition period, your programs will have to adapt. In most cases you will need to do nothing, and it will be simple to verify that your program works with Palm OS 5 and future versions. However, bad programming practices will most likely come back to haunt you and cause you and your user's great pain. As the Palm OS and its developer community learns and then graduates from adolescence, a better system awaits the Palm OS user community. These nitpicks will help you realize what is awkward and what is acceptable.

User Interface Nitpicks

- Include high resolution bitmaps and icons. This is getting to be especially important for Palm OS 5 devices and other devices that conform to the new Palm high resolution API. Look at how games look better with the high-density bitmaps and see for yourself.
- Be sure to compress your icons as the current compiler tools support this. This saves a fair amount of space, especially since applications can have up to ten icons.
- Be sure to compress your bitmaps. Programs with many bitmaps will see a significant savings in program size.
- Have transparent backgrounds on icons and bitmaps. Who knows what color the background the user has chosen with one of the many color scheme programs out there? Also make sure that your bitmap will look good no matter what color the background is. A simple technique is to give the bitmap a white perimeter.
- Make sure that you respond to `frmUpdateEvent` and redraw your form. If you do not respond to this event, there may be some missing text on your screen. The debug versions of the Palm OS simulator and emulator force this issue as a debugging technique.

- When you have a long list that displayed, make sure that the current selected item is visible. You can do this with the following code:

```
LstSetTopitem(pList, (item / LstGetVisibleItems(pList)) *  
LstGetVisibleItems(pList));
```

- If you have a long list, make sure that the width of the list takes into account the up and down arrows. If you are clever you can write a routine that tests if the list is longer than the display can hold and widen the list by the required amount.

Compatibility Nitpicks

- All Palm OS 5 devices run on some variation of the ARM processor, which is a totally different architecture than the Motorola 68,000 based DragonBall processors used on previous Palm OS devices. The ARM processor can do more with less power (Watts or battery power) than the older processors. However, legacy programs cannot take full advantage of the muscle of the ARM processor and in some rare cases some programs may run slower on an ARM processor since old 68,000 programs are emulated on the ARM processor.

Palm OS 5 includes the ability to increase the performance of a program using ARMlets. An ARMlet is an ARM native code segment linked into a 68,000 Palm program. To maintain compatibility with old 68,000 programs Palm OS 5 has a built-in 68,000 instruction set interpreter or emulator (PACE or Palm Application Compatibility Environment) that converts said instructions to ARM native code. While this has proven to be fast, some code simply cannot run fast enough in this emulation layer, hence the inclusion of ARMlets. Many applications, such as MP3 players, 3D first person shooter video games and arcade game emulators, include ARMlets to accelerate computationally extensive code.

If your application has a critical path, use the ARMlet capability of Palm OS 5. Admittedly, this is more difficult to program than writing traditional Palm OS code, but it can be worth the effort. Until Palm OS 6 arrives, this is the only way to get the true performance of the ARM processors in Palm OS 5.

- Have your programs work properly with Graffiti 2. As of Palm OS versions 4.1.2 and 5.2 Graffiti 2 will be the standard text input method. Graffiti 2 can have multiple strokes to get one letter. For example, the letter “i” will be a vertical line followed by a dot above that line while the letter “/L” will be only a vertical line. This ambiguity can be a problem if the letter “/L” is a menu shortcut so avoid using “/L” as a menu shortcut. For more information, check out [Graffiti 2 Compatibility Tips FAQ](#) from the PalmSource developer area.

Bug Nitpicks

- Get to know your compiler and learn how to turn on all warnings. Doing this may prevent you from having a bug that happens only under the rarest of conditions. For example, without the proper warnings enabled, you might be using uninitialized variables which can cause all sorts of problems.
- Use the Palm OS 5 Simulator. This is the best way to verify that your application will work on a Palm OS 5.x device. You can find it at <http://PalmOS.com/dev/tools/simulator/>.

- Use the **debug** version of the OS 5 Simulator. I have been lazy in the past and did not use the debug version very much. In fact, one of my programs had many warnings on the debug version of the simulator and I dismissed those warnings since I was doing something “special”. Later I regretted that decision. After fixing those warnings some of the strange problems that I was having disappeared. The debug version of the OS 5 Simulator checks for many common problems that may prove to cause problems on actual hardware.
- If you are using the debug version of the Palm OS 5 Simulator and your code is running too slow with PACE, try the non-debug version of the Palm OS Simulator. You will be surprised on how much faster PACE is. However, don't get into the habit of using the non-debug version as you may miss some obscure bug.
- Those memory leaks that you ignored are not tolerated too well in OS 5. Fix them.
- Occasionally you will get a strange bug report from a user that you cannot reproduce. Don't give up too soon on that bug report. Find out if the problem is really your application or some third-party conflict. Finding the source of the problem will not only make your application more stable but you will gain the respect of your users. You never know if this bug is preventing someone from purchasing your application. If you cannot find the source of the problem, keep track of it and go back to it periodically until it is mitigated.

Support Nitpicks

- You may want to include several screen dumps in your documentation or on your web page. This can be a good thing as it may help the user better understand how to use the program. Whatever you do don't use JPEG as your compression scheme for those screen dumps. It will make it look lousy and unprofessional. Stick with GIF or PNG since there will be no quality loss due to the lossy JPEG compression, as shown in *Figure 1*.
- There will be times when you get an e-mail from a disgruntled user. That e-mail may be rude or contain obscenities. When that happens don't respond immediately with a knee-jerk reaction. Step away from the computer, take a deep breath, and cool down if need be. Later you can write a polite response to the individual.



Figure 1: Screen Capture with Bad JPEG Compression

Extension (Hack) Nitpicks

- If you have a Palm OS extension that can be converted to a notification, then do so. You can do many things with notifications that required hacks in the past. While this may limit your program to working in OS 3.5 or greater, but it will work with Palm OS 5 devices. Palm OS 5 does not support extensions, or hacks.

Audio Nitpicks

- Not every Palm OS 5 device includes support for streaming sound. Be sure to check for the appropriate feature.
- Palm OS 5 has some performance issues when playing 8-bit sounds while using the audio callback function. For best results use 16-bit samples with little endian byte ordering to prevent any choppiness that you may have experienced if using 8-bit samples. If possible, convert your 8-bit audio on the fly to 16-bit little endian samples using ARMlets to save space.
- Just because Palm OS 5 includes support for sampled sound does not mean that you should add sampled sounds to your applications. Sometimes going overboard can be awkward and be a real turnoff to a potential paying customer. As most teenagers eventually learn, just because you can do something does not mean that you should do it. However, if your application is a game, then add the sampled sounds. This can add a new dimension to your game over the simple blips and bleeps.

- 11110 -