

Program Development Environment for Handheld Devices and PCs --- Masa Kasahara

Masa has spent a great deal of time searching for the ideal platform to run his music education programs. He has looked at many handheld devices that would provide the portability required to practice anywhere. Important requirements include light weight, commercial availability, ease of use, ease of developing programs, and a seamless development environment. Recently, Masa began investigating the Allegro multi-platform game library for single source development for most operating systems running on today's PCs.

Brief Review of Computer History: Application Centric

- The value of the system is ultimately determined by the applications it runs
- Not by the hardware capabilities
- Not by the operating systems
- Unless you are a computer designer

The Requirements for Music Education Software

- It must be available all the time (24/7) since music requires a full time attention, much like learning a foreign language
- It must be easy to use
- It must be reasonably priced for students' sake
- Preferably, it's fun to use – much like a game

The First Requirement Demands Portable Devices

- Reasonable screen resolution (240x240 minimum)
- Sound capabilities
- Reasonably priced
- Non proprietary environment – many gaming devices require complicated licensing terms
- Not heavy to carry

Pocket PCs with Windows CE (Mobile included): Advantages

- Small enough to carry around. It's even better if the device is a phone as well. Why? You carry around the phone anyway. No additional hardware.
- Windows – Anybody can write software without special licensing terms
- Not too expensive

Pocket PCs with Windows CE (Mobile included): Disadvantages

- Windows, but not exactly the same as desktop Windows in terms of programming and maneuvering
- Not too expensive, but it's not a regular item to own
- MIDI stack is not implemented. You have to use a third party library
- It requires different binary sets for different CPU architectures... A pain to manage!

Desktop PCs (including notebooks) with Windows: Advantages

- Widely available
- Windows – Anybody can write software without special licensing terms
- The same data set can be shared between mobile and desktop PCs
- It is nice to have a desktop version so that anybody can experiment it (marketing perspective)
- Real device emulation is interesting, but more computation intensive

Desktop PCs (including notebooks) with Mac OS X

- It is an important platform for music-related software because they still have a huge user base.
- Mac OS X – Anybody can write software without special licensing terms.
- iPod Touch is still question mark in terms of software distribution. Also, how close is it in terms of program development?

Desktop PCs (including notebooks) with Linux

I don't think it's an easy target for commercial software. Linux is not designed for binary compatibility in mind, which means you have to prepare different binary sets for different distributions. This could be a nightmare. Windows are very good at this. It is still possible to write a single binary file which runs on all Windows (95/98/ME/2000/XP/Vista).

My Initial Target Environment

- Microsoft Windows for desktops since they are the primary PC environment
- Windows Mobile for the same development tool set (Visual Studio 2005 Professional)
- Windows Mobile and desktop Windows – seamless integration

What I have learned so far

- Windows Mobile is quite a bit different from a desktop counterpart in terms of programming.
- It is nearly impossible to write one set of programs for both desktop and mobile devices.
- In particular, a MIDI stack is missing in Windows Mobile. You have to purchase a library or use an open source counterpart. This is a problem if you develop proprietary software.

What I have learned so far (continued)

- Windows Mobile is attractive for Windows developers since they share the same tool set. Otherwise, I tend to think they are more different than the same.
- Visual C++/MFC (Microsoft Foundation Class) is not easy to understand. It is a good environment where many programmers develop applications under a supervision of knowledgeable senior programmers. Otherwise, the consistency among applications can be difficult to achieve.

What I have learned so far (continued)

- .Net is even worse. It might solve technical difficulties, such as memory management issues, but it requires the installation of .Net for older Windows. It makes impossible to write a program which runs on all the Windows (95/98/ME/NT/2000/XP/Vista)
- If Windows Mobile is different from Windows Desktop, why not looking into cross platform tools?

Allegro

- Allegro is a cross platform game library.
- It covers most PCs, if not all. It seems it even covers old DOS with special graphics cards.
- Allegro is not limited to programming games. There are people who use Allegro for serious applications.
- It's important to match compiler versions. Otherwise, you suffer from syntax changes.

Allegro (continued)

- Totally license free environment. It is great if you develop proprietary software.
- This means you can port a part of the library to other unsupported platform, such as Windows Mobile. For example, MIDI stack.
- So far, I installed Allegro in Linux and Windows. Mac OS X will follow.
- I will port necessary parts to Windows Mobile.

In Summary

- By utilizing Allegro, I have one set of source for most operating systems used today.
- For mobile devices, such as Windows Mobile or iPod Touch, I will strip down the main version. This is still faster than developing an application from scratch.

Disclaimer

- This is created best to my knowledge at the time of writing. It is, however, not guaranteed to be 100% accurate. I shall not be held liable for either direct or indirect damages by your applying the information contained here. You are on your own.

Copyright Notice

- This work is copyrighted by Masa Kasahara.
- You are granted to make copies and share with your colleagues for educational purposes.
- All the images used are either my own, if not specified, or in public domain with sources indicated.
- **Third-Party Trademarks:** Brand and product names are or may be trademarks or registered trademarks of their respective holders. All rights with respect to those trademarks or registered trademarks are reserved by their respective holders.