Specifications of the Sole Computer

Travis Brown

September 5, 2007

1 What is it?

The Sole Computer is the concept of a portable computer which is the only computer the average person requires. It is designed to be used in every way a digital device is currently used, except perhaps as a cell phone. That is it will be your MP3 player, it will be your web browser, you will watch video with it, you will chat online with it, you will write papers with it, you will run spreadsheets with it, you'll code with it. This is to be accomplished by combining a flexible portable device with the ability to make us of additional interface and processing equipment at a moment's notice as it becomes available. To this end there are four major components I envision.

The primary component is the portable central CPU. This is the component which contains the most important data and holds the working set. The three other components are designed to augment the portable CPU by adding IO capabilities such as large monitors, keyboards, mice, speakers and wired network. Furthermore the other components also provide additional computational power, memory and electrical power.

The second component is a desktop dock. This dock is unlike traditional docks in that it contains a graphics accelerator, one or more processors, RAM, a network connection and the other items people currently view as necessary for a desktop computer. The desktop dock is used by plugging the portable CPU directly into it by a fast bus. The desktop dock then becomes the head for the portable system while the portable system is in direct control. The portable system does dynamic clustering to make use of the processors, RAM and graphic accelerators in the desktop dock. The final things the desktop dock offers to the portable component are other media options such as a flash reader, printers, DVD drives, etc. Of course while docked the portable unit will recharge its battery.

The third component is the laptop dock. This is similar to the desktop dock in what it provides, except that it is in a portable form factor. This is to allow for activities which aren't suitable for what a current desktop is used for, perhaps text editing while on the move, but for which the handheld interface is insufficient. The primary purpose of the laptop dock is the provide battery power, keyboard, mouse and large screen. Additional processors and memory may or may not be available or put into use depending on the requirements of the application at hand. When very little in computing resources are required they can be left powered down to increase battery life.

The final component is an augmentation less of interface and more of computing power and data storage. This is a server component. The server component provides bulk data

storage in a central location and can be used as a CPU server when network conditions are appropriate. This is intended to allow free sharing of docks within a household. An example usage would be a household with two docks and four Sole Computer users. Now it would be inconvenient if the large data archives could not be shared between the users, even if they didn't require the use of the dock. It may also be that one wants to run some large job or download in the background while somebody else uses the dock. In these situations the server becomes useful. In the business sphere the advantages become more obvious as a central location to provide data storage and extra computing power, allowing cheaper docks to be used for the desktop where a fast network exists.

2 Specifications

2.1 Handheld

The handheld unit is the central device in the system of the Sole Computer. This is to allow a single person to only need to own a single computer. The complexity of operating multiple computers is too great for most people and current computers do not provide optimal physical user interfaces (keyboards, screens and mice) in every situation. The handheld is first and foremost a handheld device. It must provide sufficient power to operate as is reasonable in it being a handheld. Beyond this it must simply be powerful and flexible enough to scale when provided with additional resources via docks.

CPU	400-600MHz high efficiency processor. Probably ARM or PPC.
RAM	512MB-1GB of RAM running at half the processor clock.
Storage	16GB-64GB Compact Flash or similar
Dimensions	133 mm X 73.9 mm X 21.5 mm (Nintendo DSLite sized)
Case	Very similar to the Nintendo DSLite, but instead have the clamshell be a simple cover and fold onto the back of the unit.
Video	A single Full face, dual mode (BW and Colour) LCD. Highest possible resolution.
Input	Touch screen sensitive to fingertip and stylus. Power button.
Network	Generic Wifi (802.11b, 802.11g, etc.). Perhaps GSM connectivity.
Sound	Attached via Bluetooth. Specifically a Bluetooh earphone/mike pair.
Keyboard/Mouse	Bluetooth.
Battery Life	• Approximately 6 hours open faced, LCD on colour. • Approximately 12 hours open faces, LCD Black and White. • Approximately 20 hours closed faces, but not sleeping (MP3 player or tour guide mode). • Approximately 160 hours sleeping. • Certainly Lithium-Ion.
Docking Connector	Dual Firewire 3200.

2.1.1 CPU

The advantage of docking to make use of additional resources is that the handheld can have a very slow processor. This is evidenced by the fact that I think 400-600MHz is sufficient for most operations. I find the reasonable as few people will be able to stand using the small screen for tasks which require complex and resource intensive software.

The processor needs to be low power to have any change of attaining the battery life I find worthwhile. This eliminates x86 based processors from the running. If energy efficiency was the only goal an ARM or perhaps MIPS would win easily, however there is an additional consideration. With the docking system the handheld clusters with a much more powerful group of processors. In order to make the most effective use of wattage on the handheld all programs must be compiled where possible down to the actual ISA in use on the handheld. Now the docks have sufficient power to emulate the ISA in the handheld, but doing so limits the computing power that can be offered there. Thus the PPC architecture remains viable as it has a relatively simple ISA that is well suited to high efficiency processors while still being scalable enough to compete with modern desktops.

I think going with a PPC is most likely as there are variants which are designed for all the power ranges we require: handheld, laptop, desktop and server. PPC is also low level enough that it is reasonable to have a low cost hardware implementation, this is in contrast with some sort of bytecode.

We could go with ARM if we had a 16 core desktop version, that should provide sufficient power.

2.1.2 RAM

The reason such a large amount of RAM is available for a handheld is that real applications will be run from time to time, one big advantage of this system is that if you really need to you can open that final report and make a few minor changes. Also being flash based swapping out must happen infrequently so there should be sufficient memory to keep a significant working set in memory.

It is probably useful to have the ability to completely power down some memory on the fly as well as clocking down the memory to save power when not needed.

2.1.3 Storage

For this device to have any battery life at all Flash is basically a requirement. I am aware of no other storage medium of the density which will be required that uses less power. With sufficient engineering it should even be possible to disable certain chips or part of chips of flash and thus save even more power.

Now the reasoning behind a minimum size of 16GB. In order for this device to be usable as the sole full fledged computer a person own their must be sufficient storage to contain most programs as well as a significant portion of the user's data. I find it reasonable that with care every application which has some use upon a portable device could be contained within three gigabytes of space. I believe that this is reasonable because my Applications directory on my Mac laptop weighs in at 7GB, though that contains some data and many of

those applications are not used, or used irregularly. Even more useful evidence is that the entire operating system size on my desktop is less than 2GB.

The limitations in storage size will present a few challenges in ensuring that we have a lean system, this will most likely require library magic to eliminate any and all redundancies.

2.1.4 Dimensions and Case

I have chosen the size of a Nintendo DS Lite as the starting point because I feel that it is a fair size to fit all the necessary components, fits well within the hand, had a usable screen area and it portable. Now where the Nintendo DS Lite has a clamshell design with two screens the handheld would instead have a single screen with a fold over cover.

Coming to the materials, though a plastic case could work, I would think instead that a metal case would provide positive wear and durability.

2.1.5 Screen

There will be a single screen for this device, it will cover as much of the internal surface as possible. Doing other than maximizing the screen size doesn't seem worth it, who wants to waste space, especially when using a small handheld device.

The screen will also be the primary input device. Thus it obviously needs to be a touchscreen. Though a multi-touch screen would be preferable I fear that it won't be feasible. A single point touch screen, similar to the Nintendo DS Lite, will be workable, but may require more work on the user interface. In being similar to the Nintendo DS screen a stylus is going to be necessary, but I like to think that the DS Lite also has that problem handled.

The screen should take a cue from OLPC and be a dual mode BW and colour screen. The resolution should be as high as possible and in the ideal case fully usable in full daylight and without a backlight often. I would like to see at least a 150 DPI, but 300 would be ideal.

2.1.6 Network

No computing device in this modern world is complete without network connectivity. Now the handheld will be slightly different than a laptop in that it will only have wireless connectivity built in, other network forms need to be provided by the docks.

Obviously 802.11b/g needs to be included, 802.11a may also be worthwhile, but it depends on the power and space costs. The other important form of network connectivity is Bluetooth. Bluetooth will be the primary connectivity option for peripherals. Any keyboard, mouse, microphone or speakers will need to be connected over Bluetooth. Doing this will maintain a clean case design and reduce the number of internal chips needed to get the work done. The only problems this design decision may bring up is the inability to connect digital cameras, but perhaps a USB-Bluetooth converter would be created to bridge this minor gap.

Now an options that would be amazing, but possibly very difficult is GSM connectivity. If we could manage to get this worked out this machine would truly be the only computer, and perhaps eventually the only electronic device, a person needs during a regular day.

Now the dock concept is key to the capabilities of the handheld. If not for docks it would be impossible to have a useful single computer in a handheld format. In order make the dock useful a highspeed connection is required. For this I have chosen Firewire. I choose this for

a couple of reasons. Foremostly it is high bandwidth, currently available versions operate at 800 Mbps though there is a specification for bandwidths up to 3200 Mbps. This should be fast enough for every communication need. A secondary reason is that Firewire allows cross device DMA. This will allow display buffers to be DMA'd between the handheld and the dock with ease and high performance.

However, single Firewire is still insufficient. DMA could easily fill a single Firewire bus and makes intrahandheld networks impossible. By going with dual Firewire there will be ample bandwidth and creating daisy-chained Firewire networks becomes simple.

2.1.7 Battery Life

Probably the most difficult engineering problem of the whole concept is getting the required amount of battery life out of the handheld. Though portable gaming platforms can get incredible battery life (over ten hours in some cases), those are less powerful machines. I feel that the battery lives are important though, especially if we want these devices to be the only computer a person requires.

To help with these battery life figures the handheld will need to be designed to powerdown anything not in use, this includes Flash, individual ICs and perhaps clusters of circuits within ICs. The long sleeping time requirement should be easy to accomplish since everything will be solid state and perhaps RAM can be Flash backed so only a wakeup circuit need be kept powered.

The 20 hours closed but awake I think will be a very useful mode. Used to MP3 playing, directions, voice controlled computing or as a cell phone this is a great challenge and will require clocking the processor and RAM down, possibly down to dozens of Megahertz.

The Black and White mode is intended to be used more for reading than real world and should be possible with appropriate video hardware which keeps the screen working while RAM and the CPU is disabled, similar to the OLPC.

2.2 Laptop Dock

Though a ubiquitous handheld is useful in many situations, there exist situations where it is insufficient, but portability is important. It is for these sorts of situations where a laptop dock would come in handy. The dock basically provides everything a laptop has: larger screen, keyboard, touch pad, CD/DVD Drive, ethernet, modem, USB, a microphone, speakers, RAM, faster processors and additional battery.

Basically you carry the handheld around everywhere and bring the laptop dock when you are on an airplane perhaps. The handheld should also be able to charge off the dock's battery.

It is thought that you just stick the handheld into the side or back or front and have it instantly recognize the dock and make use of the additional resources provided.

2.3 Desktop Dock

There are also times when you need even more power, better interfaces or just don't want to deal with all that a laptop entails. For these times there should be a desktop dock. The

desktop dock will provide all the same resources as the laptop dock, but more so. The screen will be larger, perhaps even multiple screens, better 3D acceleration, more RMA, more and faster processors, etc. There should be multiple models to serve all needs, from those who just word process, to those who do image processing and video editing.

The other thing the desktop dock will provide is additional storage. Even 64GB will not be sufficient for many people, especially when games and media become available.

The general design of the dock should be unobtrusive. Some models should be like the Apple iMacs, all in the screen. Others should be like a Shuttle, small and easily kept out of the way. Finally for those whose needs are endless there should be a model similar to a full size tower.

3 The Fun Stuff

Now people create for two reasons: money and fun. The money comes out of success. The fun comes out of interesting challenges and problems. This sort of product has those in spades. I believe all of them to be solvable, but it will take some ingenuity.

3.1 Hardware

A product like this has hardware which is challenging to create. Firstly it must be reliable, losing your sole computer for any length of time can be devastating. Secondly it must be affordable. It must also be power efficient.

The challenges related to these are dealing with powering down ICs and portions of ICs while being able to reactivate and initialize them in small fractions of a second. Related to this problem is one of space. The handheld will be small and requires lots of functionality to be useful. This will probably lead down the road of a custom ASIC or two which provide low complexity functionality which would otherwise require too much valuable space.

3.2 Software

Though the hardware has a few challenges which will keep an electrical engineer busy for a while the true power of the device is based in the software. In many ways the hardware will be limited, CPU power as an obvious example, but the software will be required to supply the moon.

3.2.1 Operating System

Now I think it is fairly obvious after some consideration that our choice of base Operating system is rather limited. Of the options Linux stands out as having proved its ability to span the power spectrum and furthermore there are known partial solutions to many of the problems we will face. This is not to say that we can rule out other free Unix variants before ample research.

The major issues which need to be dealt with on the Operating System front is hardware support, automatic clustering, connected dock device detection and adaptation and efficiency.

Supporting the hardware we design will be relatively easy, the challenge will be supporting the power saving modes, which will require specific hardware to be deactivated, reactivated often while including RAM within this set of hardware.

Automatic clustering support is especially critical. Without the ability to make use of the dock's RAM and CPU resources the handheld will be mostly useless as a central computer. There is hope however, the OpenMosix project appears to do most of what we need, the only real problem is automagic configuration and non x86 processor support.

Dock device detection is important because we will want to make full use of any screen, input device, network, optical media and fixed media which are available, knowing that it may be removed at any time. This will require scalable subsystems which can cross network boundaries.

The efficiency of the base components is critical. There is very little processing power to waste and space is also at a premium. This will require heavy optimization and some research into way of reducing code bloat while not becoming too inefficient. A solution to this library duplication problem, as in the case of C++ templates, could lead to a side solution which is marketable.

3.2.2 User Interface

The great difficulty with this project is the creation of a UI toolkit which is easy to use and thus create applications, yet also makes those same applications work equally well on a small touch screen, to a large multi-touch desktop, to a traditional mouse and keyboard interface. This is an interesting UI challenge and gives ample opportunity to fix all those UI mistakes one sees.

Especially important are resolution independent widgets and touchable interfaces. Though there are also problems with changing to BW or greyscale display modes. I am thinking that it will be necessary to move away from the desktop model of interaction simply because the handheld has insufficient space and changing display modes often, as when docking, would cause utter disarray of the desktop.

3.2.3 Applications

No platform will ever be of any use if it doesn't have applications. We'll need some. Even more than that, we'll need equivalents to all the regular things such as web browsers and office suites. We'll also need something new, something that couldn't be done before. What this is I am not sure, but surely something lends itself to the sole computer of a person, but perhaps using any application anywhere on all your data is sufficient.