

# How I Built a Home Computer Lab, Changed My Life and Saved the Earth.

Robert Andresen

*This paper describes the choices, problems and solutions the author encountered while building a home computer lab, utilizing corporate equipment that is no longer powerful enough to run production workloads. Training classes alone won't help build new expertise. What is learned in an expensive corporate class can easily be forgotten through non-use. Discarded hardware can be used to maintain and enhance skills in "new technology". We'll examine options for obtaining hardware and software to set up a small home network and to build expertise in new technologies that will allow one to keep a job with the current employer or find a better one elsewhere.*

## **Introduction**

Change is inevitable. This author, like many other IT professionals, started a career when IBM mainframes were dynamic and growing. We learned and grew in this environment, increasing our value and earnings with this expertise. As time passed, it became more and more apparent that this platform was no longer popular and was not growing in terms of number of jobs and opportunities available. Companies thought of mainframe platforms as legacy, perhaps to be replaced eventually, but not an area on which to spend much money. Other technologies became the favorite children.

The question became: what to do? Stay with the expertise built over the years or learn a new technology? Some in this position chose to leave IT altogether and start a completely different career. Another choice is to try to stay in the technology of expertise, but this means competing for an ever-shrinking number of positions as companies turn away from the old technology. Even companies that still run the legacy technologies have fewer and fewer people doing the work with one person wearing many hats, having many areas of responsibility. To find a new job working in old technology may take a long time.

Yet another choice is to learn the new technologies, stay current and follow the technology for which corporations are budgeting. There are choices on how to learn these new technologies. In a perfect world, companies would offer training to their valuable employees to keep them from becoming obsolete. There may even be a corporate value statement that

says: "People are our most valuable asset." But I digress.

## **Seminars and Training Classes**

There are many opportunities to take professional classes for new technology. Typical classes last from several days up to a week or two. It is sometimes difficult to be away from the day job for even a week, and these classes tend to be expensive enough that an employee would prefer that their company pay for this training. But there is another problem besides the cost and time away from work: It is very easy to forget the new things learned unless one immediately starts to use the new skills.

If the company will not approve a weeklong training class, would it be useful enough to pay for it out of one's own pocket? A weeklong class could cost several thousand dollars, plus travel. To save money, find something local. But also consider how much material will be covered over the course of the class.

Commercial classes tend to cover a lot of material very quickly. If the information isn't fully understood at the beginning, one may become lost later on as the class builds on the early material and progresses to more advanced material. Many people learn better when the training is spread out over a longer period. If the class meets once or twice a week after work, lasts for over a dozen sessions and has assigned work which is evaluated, participants will have a much better chance of retaining material compared to a one week whirlwind of training.

## **College classes**

In the real world one must realize that “no one will look after your own self-interest other than you.” So, how to train oneself to take control of the future? For many, a very good choice is local colleges. One of the best values is a community college. These schools were originally set up to offer low-cost, two year associates degrees for high school graduates unsure of what to do or unable to afford a traditional four year college. These colleges soon realized that there was an untapped market and resources available. The school was empty at night; younger students would take few, if any night classes. Working adults needed degrees and additional expertise, but could only attend classes after their workday was over.

College of DuPage, a local community college offers classes in Java, C#, .NET, Visual Basic, UNIX, Linux, Windows, networking and more. Many other local colleges have similar offerings. All of these are popular and growing technologies. The colleges have machines in their labs to run these technologies; however, students usually need to travel to the college to use the machine and, once the class is over, there is no longer access to work just learned. Unless one keeps using what has just been learned on a regular basis one quickly forgets. Applicants won't be an attractive candidate on a job interview with theoretical and poorly remembered knowledge of this new technology when compared to other candidates who use it every day.

Wouldn't it be great to have access to all these new technologies at home? It would be easier to do class homework, experimenting with different approaches. But more importantly, once the class is over one still can work with what has been learned. Regular repetition of using the technology will allow one to understand it much better, understanding subtleties that are impossible to know as a short time casual user.

## **Computer Lab**

A computer lab at home! Won't that be expensive? Won't that take over my house? Won't my significant other veto this? What is the feng shui for home computer labs?

Most IT professionals have at least one computer at home; many have several in a home network, with DSL or cable access to the Internet. The interesting thing to consider is why those computers are in the home. Many went out and bought a first machine to be a general use family computer. Over time that machine was significantly slower than current models on the market, and we bought a newer machine, but even though we tried to palm the old machine off on other family members, everyone wanted to use the

newer, faster machine with more memory and a larger hard drive.

Another problem: if there are excessive applications installed on a machine, processing slows down considerably. Product installs usually add services that automatically start when the machine boots. It's also not too useful to install several products on single machines that normally communicate with each other over the network. The learning experience is better if the products are running on several machines which are connected over a home network. Lastly, the machines sitting around the house tend to be x86-based. This is fine to learn Windows or Linux-based products, but useless to learn AIX or Solaris on Sparc.

Companies run software across many servers; there should be multiple machines in the home lab, but what if there is no spare room to set up the traditional server farm. Even so, unless the significant other is a computer geek, they may have other ideas for that room. Very few people have the space or the interest to put rack-mounted equipment in their house or apartment.

## **KVM switches**

The first approach is to use a KVM switch. This way there is only one keyboard, one mouse and one monitor switch between multiple machines. The choices are:

- Powered vs. not powered
- Number of machines supported
- Connections supported (OS/2, USB)

A power supply generally means that machines currently not selected still think they are connected to the “human interface” hardware, and there will be a sequence of keystrokes that will jump the monitor/keyboard/mouse to another machine. A less expensive choice was a four-port Inland KVM switch that had a rotary switch to select the machine connected. This device had a problem with Linux machines seeing the mouse after the device switched away then came back. There were many hits on Google with other people having the same problem.

A better, though slightly more expensive choice was a Belkin OmniView SE 4 port switch, which used a power supply. Another popular brand is Linksys, which some folks in my company use at corporate. The lost mouse problem went away. My suspicion is that with a power supply the switch makes each machine think the devices are still connected, whereas the mechanical switches make some machines or O/S think the mouse has been unplugged.

Another feature of a powered KVM switch is that one can switch between machines using the keyboard. The Belkin allows hitting scroll lock twice, then up or down arrow depending on which order one wants to cycle through the machines. The Belkin also supports chaining multiple switches together. It is easy to daisy chain another 4 port Belkin and, then, be able to switch between eight machines.

The number of machines the switch needs to support is another consideration. Switches are available for two, four, eight or more machines. A two-port switch will be outgrown almost immediately. Many of the larger switches are designed to be rack mountable, not suitable to sit on a desk. The Belkin can sit anywhere on a desk, the keyboard keystrokes will be used to switch sessions. It is unnecessary to access its buttons; only 10 inches wide and 6 inches deep, much smaller than a rack mount unit.

The downside of the KVM switch is that all the machines need to be fairly close to the switch and the keyboard/monitor. Another choice is to use remote desktop from the main machine to access machines anywhere on the network. This way the machines can be placed on a shelf in the basement or in a broom closet. PCs aren't as touchy as some machines, but there are environmental concerns to consider. Basements that flood would be bad, of course, but even dampness or condensation should be avoided as well. Dusty places will cause the fans and airways of the machines to clog and possibly overheat.

If the main machine is running Windows XP, use remote desktop to access other Windows machines, even if they are running Windows 2000, with Terminal Services. If the remote machines are running a flavor of UNIX, such as Linux, AIX or Solaris there are several different choices. For text only terminal one can telnet or use PuTTY (<http://www.chiark.greenend.org.uk/~sgtatham/putty/>), a free secure shell program which runs on Windows or UNIX. For a graphical interface to X Windows, one can use the commercial product Hummingbird Xceed or its competitors. If a commercial product is too expensive, consider using VNC (<http://www.csd.uwo.ca/~magi/doc/vnc/index.html>), which is free under a GPL license. VNC allows access to remote Windows/Solaris/Linux/Mac from Windows/Solaris/Linux/Mac/DEC Alpha. It is stateless at the server side so connect from one machine, then go to another machine and pick up that connection where you left off.

## **Networking**

Another issue with the machines is how to network them. If only one machine is hooked up to the DSL or cable ISP (internet service provider), PPPoE is

installed to provide signon and authentication information on that computer. A router is needed to run multiple machines on a home network. The Ethernet cable from the modem (cable or DSL) should be plugged into a port on the router and the machine cables must also be plugged into the router, which will need to be configured to do the PPPoE authentication. Good news is that most routers have a web browser interface to do this configuration.

But what kind of router to get? Considerations include:

- Wired vs. Wired and wireless
- Speed
- Number of ports

If laptops, wireless is more useful. If there is no easy way to string Ethernet cables between places where computers are located, wireless is the solution. But which speed? The two common wireless routers were 802.11b, which run at 11Mbps (million bits per second) and 802.11g which run at 54Mbps. By now, the 11Mbps aren't sold new. The first time user would prefer to buy the faster router. When considering replacing older routers, consider the following:

- DSL 1.5Mbps
- Cable 3Mbps
- 802.11b 11Mbps
- 802.11g 54Mbps
- Wired 100Mbps
- D-Link\* 108Mbps

The 802.11b is over three times faster than cable, and the cheapest option, wired Ethernet is almost twice as fast as the fastest wireless. The moral of the story is, unless a fractional T1 is run to the house, the 11Mbps wireless will work just fine to the Internet. The difference is how fast machines on the local network will talk with each other. If frequently copying large files from the laptop to a home server, the faster wireless router will make a difference.

The other choice is Cable vs. DSL. Cable is theoretically faster than DSL, but that does not guarantee that it will be true in practice. Numerous Internet articles discuss network latency of both, finding cable to have more intermittent and longer latency than DSL. (Latency is the time it takes for data to make a round trip, send a packet, and get an acknowledgement.) If it is necessary to connect to

---

\* D-Link advertises that if using a D-LINK 802.11g wireless router and connecting with a D-Link 802.11g wireless card, their compression will hit speeds of up to 108Mbps.

servers back at work, slow latency will make interactive terminal sessions have bad response.

Which service is better, DSL or cable. Personal anecdotal evidence compares AT&T advanced DSL with a next-door neighbor who has Comcast cable. The AT&T network is secured, his is not. On occasion the laptops pick the wrong wireless signal and end up connected to his network. There is not a recognizable difference in how fast the Internet is no matter whose network the laptop connects to. (Though AT&T advanced DSL is advertised as being up to 3Mbps, the same as cable)

The first wireless router was a Linksys 802.11b which worked fine except for one annoying problem; it seemed to lock up intermittently. It had to be powered off and on to reset it, sometimes several times a day, sometimes once every other day. Even upgrading the firmware didn't fix the problem. The firmware didn't seem to have any newer updates; perhaps because it was an older model. Many similar complaints were found while searching the Internet for fixes or workarounds.

The fix deployed by this author was to upgrade to a D-Link 802.11g router and the problem went away. Is this because D-Link is better than Linksys or because the D-Link is newer? It's hard to say. After searching the internet there were fewer complaints about D-Link than Linksys. This was surprising as Linksys is owned by Cisco, a well known company many people respect. Nonetheless, when the problem went away there is less incentive in looking for the cause.

But the network setup may not be done with just a wireless router. The home routers have four wired ports. If there are more than four machines there is a bit of a problem. Also, if putting a couple machines in a closet is it advisable to run a long cable from the router to each machine? At some point a hub becomes necessary. Run one long cable to the machine closet, plug it into a hub, and then plug each machine into the hub as well.

What are the hub considerations? Mainly the number of ports. Home hubs range from four to eight ports. Linksys has a five-port hub that allowed more computers on the local network. Somewhere along the way one of the ports got fried. Linksys offered to replace the hub if a receipt could be shown; unfortunately the receipt had long ago been thrown away. The moral of this story is to keep all receipts. If the purpose of the home computer lab is for a business, the receipts should be retained for tax purposes. The next step was to replace the Linksys 5 port hub with a NETGEAR® 8 port hub. The Linksys hub is still being used in another part of the network.

Once there is a network infrastructure, using wired, wireless, or more likely a combination of both, configure IP addressing. DHCP works fine for laptops with wireless devices. For computers and printers plugged into a wired port assign constant IP addresses. For the most part these are servers that will have a constant address. To telnet to a machine or print to a network printer, it would be annoying to constantly change the IP address after every power down.

This home network does not have its own domain name or domain server. There are concerns about a work laptop not being able to connect to the home network if they have different domains. Currently, there are no problems on the network that setting up a domain would solve. The outside world cannot yet see the home lab. For the most part, simply use the IP address in lieu of a host name. To access machines by hostname instead of IP addresses, simply update the hosts file on the local machine to know which IP addresses belong to which machines.

### Computers

At this point we see the network infrastructure can be set up for the home lab, but we have not yet gotten to the meat of the discussion: where to get the different machines. One could go to the neighborhood super-mega-monster store and buy lots and lots of PCs, but that would be very expensive and would be limited to Windows, Linux-capable, and maybe a Mac. But if there is a need to run Solaris or AIX, the mall won't suffice.

Another source of technology is eBay. Companies lease computers. When the lease was up, they lease newer, faster more powerful machines and dump the old machines. As processors keep getting faster and faster, machines several years old are viewed as obsolete. But are they useless? Should they all go into landfills? Or be crushed or stripped for scrap metals? (This is the "save the earth" part of the paper in case you might be napping)

A corporate server that needs to handle hundreds or thousands of transactions a minute needs to be fast enough to handle that workload. A home lab won't be running production workloads; the most work might be a big compile. It might take longer than it would on a corporate server, but there's no reason to sit there waiting for it, log on remotely, then go work on another machine and come back later. Machines with processors around 1GHz works satisfactorily as long as they have 512Mb or more memory. Hard drives running at 5400 or 7200 RPM help speed machines considerably, too.

The machines that come off corporate lease are bought up by computer recyclers. Some of these companies sell them on eBay. With luck, a local company will let you pick up the computer and save on shipping. For some of these machines the shipping costs as much as the machine. Once a local recycler is located, visit them and ask about the kinds of equipment they typically get. They may be able to call as soon as they get the kind of machine for which you're looking. A local recycler occasionally gets DEC/VAX equipment, something to consider if more DEC assignments come up in the future.

What kinds of machines are available on eBay and for how much? Here are some real world examples:

- Compaq 930Mhz P3 (Fedora Core 4)
  - 512MB of memory
  - 14G hard drive
  - \$51 plus \$16 shipping.
- Dell Optiplex GX110 866Mhz P3 (Windows 2000 server)
  - 512MB memory
  - 15G hard drive
  - \$49.51 plus \$27.59 shipping
- IBM Netvista 1.3GHz P4 (Windows XP Pro)
  - 128MB memory (had to upgrade)
  - 40GB hard drive
  - \$90 plus \$25.57 shipping
- Sun Sparc Ultra 5 270MHz (64bit) (Solaris 10)
  - 384 MB memory
  - \$31 plus \$17 shipping
- IBM RS6000 7043-150 43P (AIX 5.3)
  - 1G memory
  - 18G hard drive
  - \$250 plus \$55 shipping
- Apple G4 PowerMac 533MHz (Mac 10.3)
  - 768MB memory
  - 40GB hard drive
  - \$235 plus \$26 shipping

Anytime one considers buying something on eBay, it is important to do some research to find the best price for the item. NexTag.com, buy.com and shop.com are good sites that compare prices across many vendors. Frequently these vendors are selling the item new or refurbished with a warranty. Then, check eBay to see whether an appreciably better deal is possible. If something new is available for sixty dollars, it doesn't make sense to buy it used for fifty plus shipping. Many auctions have the price go higher than another vendor is offering the same item. That is part of eBay's success; the adrenaline kicks in and people bid higher

than they should and the process is addictive for some. So beware.

So far, most of the things bought on eBay or from recyclers work fine, with a few exceptions. Some guidelines:

- Buy from sellers with a high positive feedback rating as well as a high feedback score (number of items sold). They are most likely to stand behind defective products and refund the money if it doesn't work.
- If returning an item, send it back registered. An amazing percentage of shipments don't get delivered if there is no receipt notification.
- Read the item description very, very carefully. Watch out for the words "as is" or "for parts". There seem to be a number of people using eBay in lieu of garbage pickup.
- Watch the shipping price. Some computer equipment can be very cheap but very heavy. In those cases look for a local seller that will allow pick up of the item. An HP LaserJet 5 printer went for \$25. It was being sold by a local recycler who allowed local pick up for \$5--rather than paying \$50 for shipping. They even threw in a network card and an extra toner cartridge for another \$10.
- Be careful about buying memory. Some systems, notably Apple and those using RDRAM are extremely picky about which memory will work. A good vendor is memoryx.com. They have an ActiveX application that will check a Windows computer and give lifetime guaranteed memory recommendations. Their prices appear to be very competitive.

When building a home computer lab, remember that we can sometimes learn more from our failures than our successes. It helps to repeat this when things go wrong. Think of the cowardly lion in the Wizard of Oz:

*"I do believe in spooks, I do believe in spooks..."*

There is the saga of the IBM Netvista P4 1.3GHz with a 40GB hard drive and 128MB of memory. It cost \$90 plus \$25 shipping on eBay, so far not so bad. But it didn't have enough memory to run the applications wanted. The rule with computers is similar to cars and houses: let someone else restore and upgrade them. Memory is getting cheaper over time, but RDRAM is still relatively expensive.

The first effort to upgrade involved buying some RDRAM memory on eBay. The new memory was added to the existing memory. The machine failed to boot. The new memory was tested by itself--same result. A memory slot was cracked by hurriedly swapping the memory cards in and out. Things continued from bad to worse.

At this point, it seemed to be a good idea to return the bad memory and replace the motherboard. (Soldering motherboards is beyond the scope of this paper and the author's expertise.) There appeared to be the same motherboard on eBay. For those keeping score at home, it was an ASUS P4T-M. Well, it was similar, but not quite the same. There are literally hundreds, if not thousands, of motherboards. The main problem was that the front power switch on the Netvista had a socket that didn't match the pins on the new motherboard.

So if we buy a new ATX case, it should have the correct pins, right? Another \$35 out the door for a new case, but it, too, had a different socket for the front switch. The computer wouldn't boot unless this front switch is pressed. So the computer still won't start! Fry's Electronics had a switch which would work, but didn't fit anywhere on the front panel. Purists may disagree with drilling a hole in an expansion slot cover and mounting the switch in the back of the computer, but that was the least destructive solution.

But the machine still only had 128MB of memory, which is where this saga began. Another eBay search found 512MB of RDRAM memory for \$100. Notice the memory was more than the original price of the machine. The lesson here is: buy the machine that has what you want rather than trying to upgrade. If that isn't possible then at least find a machine that takes cheap memory.

The final challenge was to come up with a host name for a machine built with the parts of several other machines with an ugly switch sticking out of an expansion slot cover. The jury-rigged switch is reminiscent of the main character of Mary Shelley's novel with the bolts sticking out of his neck. The question remains, if we name a machine "Frankenstein", are we required to wait for a lightning strike to reboot the machine?

## **Software**

Now we have machines, they are on the network and accessible either from the KVM switch or remote desktop, but what can they do? How do we get the software we want to run? The easiest is Linux and all the open source projects. Download not only the

operating system but also all kinds of applications that will run on the various forms of Linux.

The first important concept for Linux is to distinguish between the Linux kernel and the Linux distribution. Linux is an open standard; no one can sell it as a product. There are companies that build special distributions of Linux and charge for the install media and support. Since we are in it to learn we'll do our own support and if we're connected to the Internet we can download the install media for free.

There are many Linux distributions, popular ones include:

- RedHat (Fedora is their free distro)
- CentOS – A free distro based on Enterprise Redhat
- SuSE – Novell now owns them
- Debian
- Gentoo
- Mandriva/Mandriva
- Slackware

Each release of each distribution will have a Linux kernel at some release. Kernel 2.6 is the current version.

Some of the open source software that is available:

- OpenOffice: An office suite similar to commercial versions
- MySQL: Relational database
- Mozilla Firefox: Web browser
- GIMP: Image editing

The list goes on and on. Check here for a more detailed list: <http://www.linux.org/apps/index.html>. Depending on the reasons for building a home lab, it is possible to do everything with free Linux systems and open source code.

Prior to Linux becoming so popular, there were several open source Unix distributions built on the Berkley Software Distribution (BSD) version of Unix. They grew out of the sixth edition of Unix released by the University of California, Berkeley back in the late seventies. The famous quote to remember is: "Two culturally revolutionary things came from Berkeley: Unix and LSD. This is not a coincidence."

--anonymous

- FreeBSD
- NetBSD
- OpenBSD
- PC BSD
- M0n0wall (designed for firewalls)

Continuing on an exploration of free software, there are some vendors who will permit a download of their commercial software for personal use and learning. Sun will allow Solaris to be run this way. And there are two versions, one for Sparc machines, another for x86 processors. Thus Solaris could be run on an ex-Windows machine, but verify that any applications to be run under Solaris have also been ported to the x86 platforms as well.

Oracle will also allow a download of their database for learning and personal use too. All software downloads are free and each comes with a Development License that allows use of full versions of the products only while developing and prototyping applications. They have versions for Solaris (Sparc and x86), Linux, AIX, HP-UX and Windows.

What about Windows? A licensed copy of whichever version of Windows must be purchased/available for each machine. Check each machine for Certificate of Authenticity (COA) stickers. If the machine has a COA for the desired version of Windows, simply load that version. If the version is older, purchase an upgrade. Most business software will run on Windows 2000 Professional, which has a fairly small footprint (make sure there is sufficient memory) and will be cheaper than buying Windows XP or 2003 server.

This leaves IBM. My company is an IBM partner, which allows us to run a number of their software products for training and development purposes. IBM will allow some of their products to be downloaded and run for evaluation, but this usually is only for thirty days. Some of their Linux products have free downloads for educational purposes. IBM will also allow you to sign up with PartnerWorld or developerWorks, but there is a qualification process. If your company uses IBM products, check with the IBM rep to see what options they offer you as for educational or evaluation use of their software.

For commercial software, consider looking at a used software shop or on eBay. Frequently an older release is available at substantial savings over the most current release at the big retailers. Even when paying full price for the latest software, it is going to be back leveled in six months to a year. Unless there are plans to upgrade to the latest release, buy the older release and save the extra money.

Another source of less expensive software is available to students. The only requirement is that you are enrolled in an accredited high school or college. Most college bookstores have the shrink-wrapped software sitting on the shelf, already discounted. Just show a student id.

The cheapest software is open source, of course. Not only available to install on Linux systems, much of it has been ported to Windows and Mac. OpenOffice approaches the ease of use and features of commercial office suites, even adding extra features like being able to write Acrobat (.pdf) files. Sourceforge is a good starting point to look for open source software. Also, search Google for open source software, there are sites like: <http://www.theopencd.org/>, which is a collection of high quality Free and Open Source Software. The programs run in Windows and cover the most common tasks such as word processing, presentations, e-mail, web browsing, web design, and image manipulation.

If open source software is of interest, keep in mind that each project is constantly looking for volunteers to help develop and test their software. Participation can not only help give back to the open source community, but can gain new expertise that looks much more credible on a resume than taking a seminar in a new technology.

### Futures

As stated in the beginning of this paper, this author originally started out with IBM mainframes. My job still requires me to install software on these machines. Once one is no longer a systems programmer, it's easy to start forgetting much of the details and internals of z/OS. Even when working on a z/OS system with a client, one is not usually granted any of the authorities to do many functions. Wouldn't it be nice to have a z/OS system as part of a home lab? Few of us could afford the price, space or electricity required for a real mainframe.

There is Hercules, a software implementation of the System/370, ESA/390 and z/Architecture mainframe architectures that can run under Windows, Linux or UNIX on Intel, Alpha, Sparc, and Mac hardware. Thus, a PC can emulate:

- System/360
- z990 running in
  - S/370 mode
  - ESA/390 mode
  - z/Architecture mode

Hercules allows free use and distribution under its Q Public License. Hercules does the emulation, but does not provide the IBM operating systems. Either get a license for the system you want to run or use an older system, in public domain. What is public domain? According to the Hercules web site: <http://www.conmicro.cx/hercules/hercfaq.html#1.01>

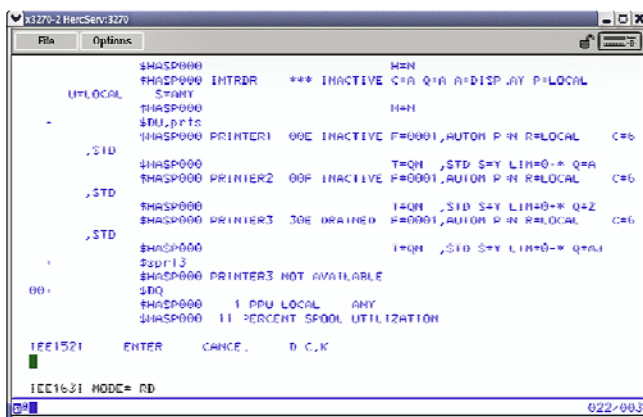
*“Most 3rd party operating systems like Linux/390, z/Linux and TELPAR are covered under their own free license and can therefore run under Hercules without any legal problems.*

*OS/360 (PCP, MFT and MVT) is in the public domain, as far as we know. The status of OSeS for which IBM did not charge a license fee is somewhat murky; these include MVS 3.8, VM/370 release 6, and DOS/VS release 34.”*

The speed of the PC required depends on what you want to run. Users have reported that OS/360, MVS 3.8, VM/370 will run on a Pentium 300Mhz with 32MB of memory. More recent systems will require faster processors and more memory. Linux/390 or OS/390 have been run with light workloads on 2GHz multithreaded processors, or multiple processor machines. The 64-bit systems, zLinux or z/OS, put an additional strain on the processor because they need to emulate 64-bit operations on a 32-bit processor. The suggestion for 64-bit systems is to use a 64-bit processor with a 64-bit version of Linux.

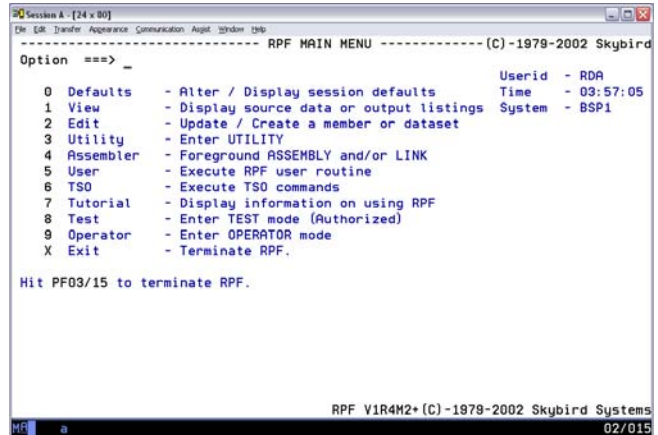
The memory recommendation is to set S/390 main and expanded storage high enough to eliminate paging, and then have enough real memory to support the S/390 memory plus the requirements of the operating system Hercules is running under. Get a hard drive large enough to hold the emulated DASD devices. Keep in mind that a 3390 model 2 is about 2GB, so an 80GB 7200 rpm drive will hold about two strings of 16 devices, plus will have room for Hercules and Linux.

There are a few choices for public domain IBM mainframe software listed on the Hercules website: <http://www.conmicro.cx/hercules/>. Modern IBM operating systems require a license. The newest version of MVS which is considered public domain is 3.8J. This site has links to download a tur(n)key system. Here is its MVS console in Figure 1:



**Figure 1, Hercules MVS console.**

IBM has allowed MVS 3.8 to become public domain, but the rest of the program products that ran under it are not. Thus ACF2, CICS, DB2, IMS, ISPF, and SDSF are not available on this free distribution. There are SHAREWARE utilities such as RPF, which approximates some of the functions of ISPF see Figure 2, RPF shareware utility screen.



**Figure 2, RPF shareware utility screen.**

This is a very old operating system, pre-XA, but the MVS 3.8J public domain system runs quite well on a 1GHz Intel processor with 512MB of memory. Back in the day no one expected to be able to run a mainframe at home. Now we all can.

### **Conclusion**

A home computer lab can assist one in gaining expertise in new technologies, which is important in today's volatile job market. It need not cost very much and is a better use of old technology that overloading landfills. When planned properly, the lab will not require a lot of space, especially if you already have a desk with a keyboard and a monitor. The most important thing to remember is its fun to work with new technology.