# Chapter 21

# Virtualization of Legacy Instrumentation Control Computers for Improved Reliability, Operational Life, and Management

Jonathan E. Katz

## Abstract

Laboratories tend to be amenable environments for long-term reliable operation of scientific measurement equipment. Indeed, it is not uncommon to find equipment 5, 10, or even 20+ years old still being routinely used in labs. Unfortunately, the Achilles heel for many of these devices is the control/data acquisition computer. Often these computers run older operating systems (e.g., Windows XP) and, while they might only use standard network, USB or serial ports, they require proprietary software to be installed. Even if the original installation disks can be found, it is a burdensome process to reinstall and is fraught with “gotchas” that can derail the process—lost license keys, incompatible hardware, forgotten configuration settings, etc. If you have running legacy instrumentation, the computer is the ticking time bomb waiting to put a halt to your operation.

In this chapter, I describe how to virtualize your currently running control computer. This virtualized computer “image” is easy to maintain, easy to back up and easy to redeploy. I have used this multiple times in my own lab to greatly improve the robustness of my legacy devices.

After completing the steps in this chapter, you will have your original control computer as well as a virtual instance of that computer with all the software installed ready to control your hardware should your original computer ever be decommissioned.

**Key words** Legacy hardware, Virtual computers, System reliability, Mass Spectrometry, VirtualBox, Cloning, Systems management

## 1 Introduction

Scientific measurement devices with attached control computers often have a viable life that greatly exceeds the supported life of the operating system that hosts the vendor software. It is also often the case that the vendors have either gone out of business or have discontinued support for their older devices in this same time frame. Even for the vendors still in business, the process of, for example, reinstalling and reconfiguring a mass spectrometer, an LC system and its associated autosampler is not a fast nor simple process.

Lucio Comai et al. (eds.), *Proteomics: Methods and Protocols*, Methods in Molecular Biology, vol. 1550,
DOI 10.1007/978-1-4939-6747-6_21, © Springer Science+Business Media LLC 2017

Many labs often have multiple deployments built on (now) legacy operating systems, such as Windows XP, for which there is no driver support under more current versions of Windows. In spite of being well past their amortization, these devices still provide invaluable daily service. A failed computer alone (all too common an occurrence) should not be deciding factor forcing procurement of replacement funds.

In this chapter, I show you how to take the required steps to virtualize a legacy control computer. A virtual computer is computer that is created in software that runs on a host computer. In its most common deployment, there is a definition file that describes the features of the computer (the memory, the processor type, what interface cards are available, etc.) and there is an image file that contains the contents of the hard drive. Modern computers are more than capable of storing, simultaneously, the definition and image files of multiple virtual computers. While there is a minor performance penalty for running the virtual equivalent of a real computer, this deficit is greatly outweighed by the administrative and logistic advantages. Effectively, a virtualized computer is entirely recapitulated by a single file; a file as easily copied, moved, or otherwise manipulated as any other. Physical computers are often different from one another, but, from the perspective of the operating system and the applications, every virtual computer with the same definition file looks the same.

When deploying a new physical control computer (say to replace one with a failed motherboard), it requires installation of an operating system, all the applications and then configuring all the components—hours of work. On the other hand, to deploy a duplicate of a configured virtual machine requires copying a file and starting the virtual machine—minutes of work.

Creating a new virtual computer instance is fairly straightforward and not unlike deploying a new physical computer. It requires installation of an operating system, installation of applications and then configuration of these components. However, for legacy control, this has the same problems as installing a new physical computer—copies of operating systems need to be found, original control software and drivers needs to be installed and configured, etc. The scope of this chapter, however, is the virtualization of an *existing* installation.

Your currently running machine is already installed and configured, so, all the installation steps are already done! **This is the inherent advantage to virtualizing a currently running machine**. However, virtualizing existing Windows installations has some other complications as compared to creating a new virtual instance. The primary complication comes from the fact that Windows has a concept of the underlying hardware and virtualization will change the hardware that the operating system is expecting to see—Windows interprets these changes as errors. There are two primary forms these errors take. One is licensing—if Windows sees too many changes to what it perceives as the underlying hardware, it will assume that it has been copied and generate an error; Windows will boot, but it will require reactivation. The other error is that if the underlying hardware changes, Windows will assume it has been corrupted and will not successfully boot. Both of these issues will have to be addressed. The final complication to address here is that of virtualization peripherals—the cards and ports on the computer that the applications need to use. The serial ports, the network interfaces, etc. all need to be made available to the virtual computer. There is one final error that can occur, the original computer may not have appropriate drivers installed to match what is presented by the virtual hardware; this tends to be rare for XP virtualizations.

As an example, I will be using a 2007 install of an LTQ-Orbitrap with a CTC-Leap autosampler and an Eksigent 2D LC system. The control computer is running Windows XP SP3. This configuration presents some nice challenges in that it requires the use of 2 network interface cards as well as several RS-232 serial connections. I have used very similar steps to virtualize spectrophotometers, DNA spotters, and gel documentation systems. While the presented process focuses on the virtualization of Microsoft Windows based installations (primarily XP) using VirtualBox virtualization software, the essential steps remain unchanged for different operating systems and are as follows:

1. Prepare the live machine to be virtualized.
2. Cleanly shutdown the live machine.
3. Duplicate the raw hard drive data to an image file contains the complete image of the hard drive.
   (a) This will usually be done by booting an alternate operating system on the live machine, attaching an external hard drive and executing a duplication job. Sometimes it will be required (or desired) to remove the current live hard drive and duplicate it on a different machine.
4. Prepare the host computer to run the VirtualBox Software
   (a) Adjust BIOS settings.
   (b) Install VirtualBox.
5. Convert the hard drive image file into an image readable by VirtualBox.
6. Create an initial hardware definition and attempt booting the virtualized machine.
7. Finish configuration and perform required post-virtualization processes.
   (a) Re-validate windows.
   (b) Configure peripherals.
   (c) Check application configurations.

## 2 Materials

1. Software—Within this chapter, there are many referenced tools that are to be downloaded from the Internet. Because of the uncontrollable volatility of Internet links, we have made available an uncontrollably volatile mirror site for these applications. The mirror link is:
   https://drive.google.com/folderview?id=0B41aLALlbwPWX2FQRjVqMjJoeFU
   Conveniently accessible by its volatile short link:
   https://goo.gl/3MDo5Y
   This mirror should only be used as a last resort when the official channels are unavailable.
2. Host computer.
   (a) Hardware—Most modern computers should be sufficiently powerful to virtualize a computer from a couple of years or more in the past. There are two important considerations. The first is that the available RAM should be at least 1–2 GB more than the RAM that is installed on the machine you are virtualizing. The second is that the computer should have free hard drive space at least the size of the hard drive of the computer you are virtualizing. For example, if I am virtualizing a machine with a 500 GB hard drive, I find a host computer with a 750 GB hard drive installed usually more than sufficient.

   (b) Operating system—My preference in virtualization software is VirtualBox from Oracle. Any operating system that can run VirtualBox will be sufficient, this includes Linux, Windows, and Mac OS.

   (c) Virtualization software—VirtualBox is available for free download from [1]; for Linux distributions there is often an easy install process, *see* **Note 1** for an Ubuntu example.
3. External hard drive—An external USB hard drive that is at least the size of the hard drive you wish to virtualize. Note, if you are virtualizing a 32-bit machine, it is probably best to use a 2 TB or smaller external USB drive.
4. External drive reader (optional)—For some people and for some significantly older computers, it is difficult to create the image of the hard drive from the machine that you wish to virtualize. This can happen for a number of reasons—the computer might not be able to boot external media, the computer might have severely limited resources, etc. In these cases, it is often easier to remove the hard drive and duplicate the contents on another machine. To do this, it is recommended to obtain a multi-format USB hard drive reader. I have had good

success with the SATA/IDE to USB converters made by Sabrent, however, other models should work similarly well. One that mounts SATA, PATA, and IDE will fulfill most needs.

5. External boot media (e.g., CD-R, USB flash drive)—If you are going to create your disk image on the existing physical computer, you will need to boot into an alternate operating system to allow for the duplication of hard drive contents without corruption. Depending on your physical computer, this can be either a blank CD-R or USB stick prepared with a bootable copy of an operating system suitable for making the image copy. Typically a 2 GB USB drive is more than sufficient.
6. External USB->Serial adapters (possibly required). It was very common for older devices to use RS-232 serial ports as their means of connection to a host computer. Sometimes the live computer would use "multiple port serial cards" to increase the available RS-232 ports. It is common for your new host machines to have less of these ports available. Further, it is sometimes difficult to virtualize the multiple-port RS-232 cards and even the onboard RS-232 port. These issues are readily solved with USB based RS-232 interfaces. I have had much success with the TRENDnet TU-S9 USB to RS-232 interfaces, however, other models should work similarly well.

## 3 Methods

1. Prepare the physical computer to be virtualized. Much of this section mirrors the unsupported protocols referenced from VirtualBox [2]:
   (a) The primary hardware incompatibility that stops successful Windows XP virtualization is that the Windows hard drive driver does a check of the attached hard drive controller; if the controller has changed, Windows will stop the boot process. The easiest remedy is to relax these checks before creating a hard drive image; VirtualBox provides a tool on their website (MergeIDE) that greatly simplifies this process. This is not a problem with Windows98.
   - Obtain MergeIDE (noting capitalization) from: https://www.virtualbox.org/attachment/wiki/Migrate_Windows/MergeIDE.zip
   - Extract the contents of MergeIDE.zip.
   - Execute MergeIDE.bat on the live computer ("double click" works fine for this. Be sure to double click "MergeIDB.bat" and not "MergeIDE.reg").

   (b) Cleanly shutdown the live computer.
2. Create a raw ("dd") copy (i.e., "image") of the live machine's hard drive. In this example, we will boot the existing live com-

puter with either a CD or USB containing a copy of OSFClone. This is the least invasive process and, in practice, perhaps the most useful technique but it does require the live computer to be able to boot off of secondary media such as a CD-ROM or a USB flash drive. While we give the OSFClone example here, this can also be performed from any live Linux distribution as described in **Note 2**. While least optimal, if the computer cannot boot from secondary media, it will be required to physically take the hard drive out of the live machine and then, on a different machine, use an external drive interface to create the image. This is also the required process if the live machine is having problems booting off of CD-R's and USB sticks. If needed, the process of removing a hard drive to image it is described in **Note 3**.

Below is the method for creating a full hard drive copy (i.e., creating a hard drive image) on a live system using OSFClone.

(a) Determine which bootable media type will work best on your current live computer. While more convenient, older machines often have difficulty booting from USB media and are often more successful at booting from CD-ROM.

(b) Go to http://www.osforensics.com/tools/create-disk-images.html and download either the.iso (to create a bootable CD-ROM) or the.zip (to create a bootable USB).

(c) Follow the instructions on the osforensics website to burn the image to your chosen media. For example, in Windows 7, for the.iso, right-click and select burn image, for the.zip, unpack it then run the ImageUSB tool. In Linux, one might chose to use "brasero" to write the.iso to a CD-R disk.

(d) Power off the live computer if not already.

(e) Attach the USB external hard drive you will use to hold your hard drive image.

(f) Boot OSFClone on your live computer using the CD-R or USB you created. For this to be successful, you may need to enter the BIOS to change the boot order (*see* Fig. 1 and **Note 4** for some further description of this).

(g) Figure 2 shows the boot process for OSFClone. You will be presented with a prompt "**boot:**" at which point you should press the enter key.

(h) You may be presented with a prompt to "**see video modes available**", it is ok to just press the space key here to continue.

(i) Once OSFClone is started, select the option to "**image**" a complete drive (type the number **2** then press the **enter** key; please note that this interface does not use the mouse or the arrow keys. You will need to type the values you are

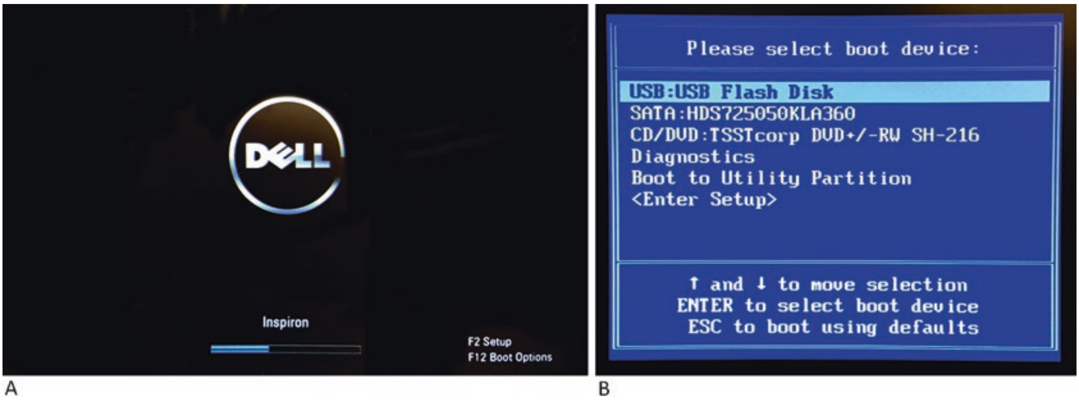
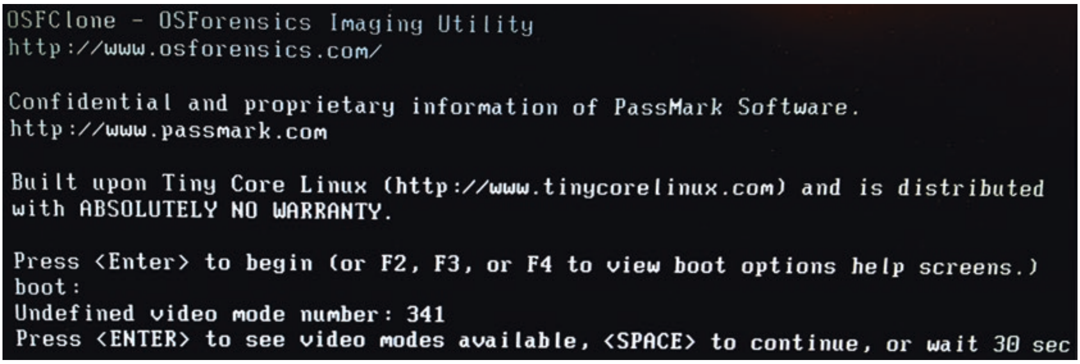

DELL
Inspiron
F2 Setup
F12 Boot Options
A
Please select boot device:
USB:USB Flash Disk
SATA:HDS725050KLA360
CD/DVD:TSSTcorp DVD+/-RW SH-216
Diagnostics
Boot to Utility Partition
<Enter Setup>
↑ and ↓ to move selection
ENTER to select boot device
ESC to boot using defaults
B

**Fig. 1** Booting from secondary media. Panel (**a**) shows a representative example of a splash screen when a computer is powered on. Pressing F12, as prompted by this screen brings up the menu shown in panel (**b**). From this menu, the system can be booted from prepared USB flash drives, removable CD drives, or other media



```
OSFClone - OSForensics Imaging Utility
http://www.osforensics.com/

Confidential and proprietary information of PassMark Software.
http://www.passmark.com

Built upon Tiny Core Linux (http://www.tinycorelinux.com) and is distributed
with ABSOLUTELY NO WARRANTY.

Press <Enter> to begin (or F2, F3, or F4 to view boot options help screens.)
boot:
Undefined video mode number: 341
Press <ENTER> to see video modes available, <SPACE> to continue, or wait 30 sec
```

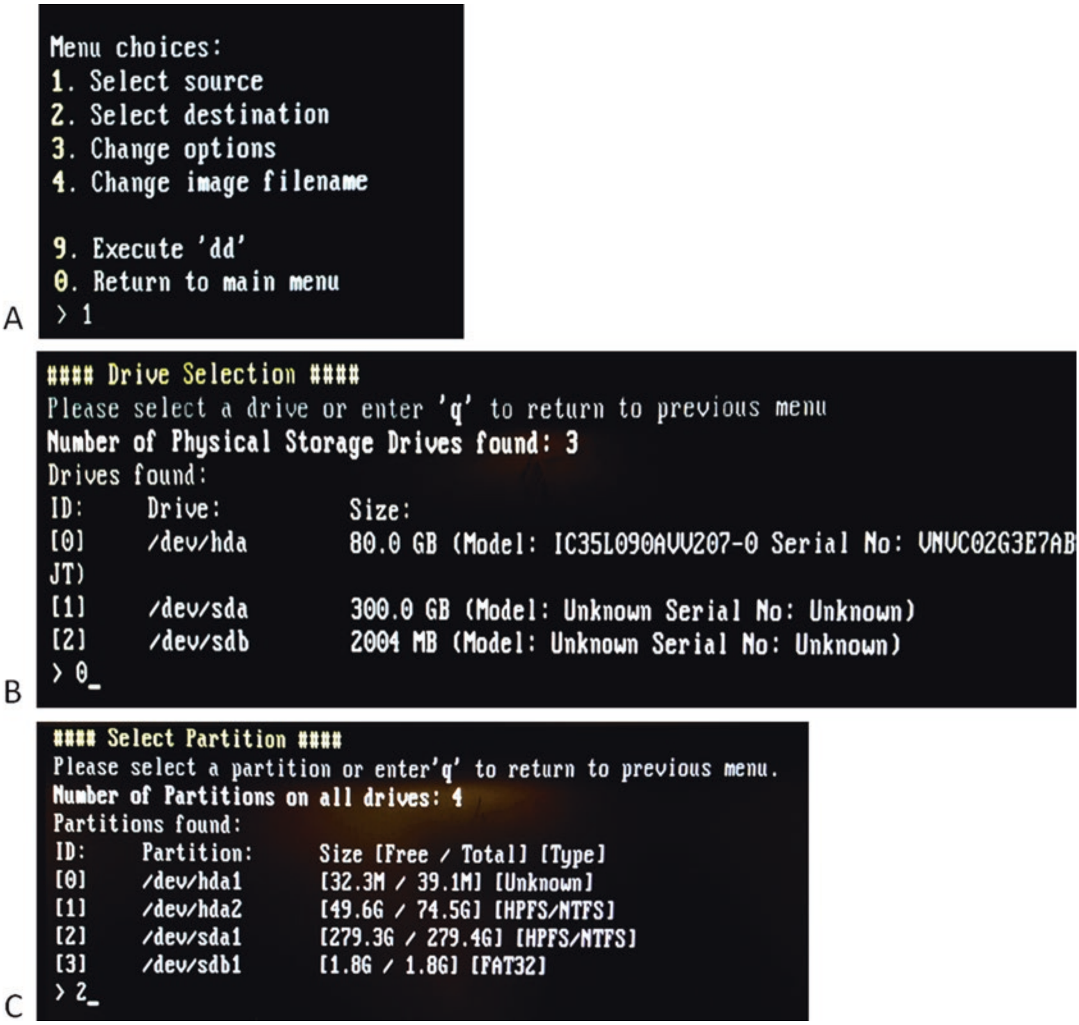
**Fig. 2** Booting into OSFClone. OSFClone is built on an operating system with very rudimentary drivers. As such, it is entirely keyboard controlled. In this screen shot, the user was prompted to press enter at the "**boot:**" prompt and then asked to press the space bar to boot without further defining the video card

prompted for and press enter). Image will copy the entire hard drive into a file (in contrast to "clone" which would duplicate one device onto another device). You may be presented with options of which "**format you wish to use**" and you should select "**dd**" to perform this imaging operation (i.e., type **1** and press the **enter** key).

(j) You will now need to select a source and destination. *See* Fig. 3 as an example.

- Type **1** and press **enter** to select the source; most likely it will be "/dev/hda"—confirm this on your system; you will be able to confirm this based on the reported size (it will match the size of your “c:” drive). Return to the option menu.

- Type 2 and press enter to select the destination; most likely it will be “/dev/sda1"—confirm this on your system; you will be able to confim this based on the reported size (it will most likely be the largest size available). Note that for the destination of an image operation, the partition will be mounted and a file will be created in that partition.



```
Menu choices:
1. Select source
2. Select destination
3. Change options
4. Change image filename

9. Execute 'dd'
0. Return to main menu
> 1
```

```
#### Drive Selection ####
Please select a drive or enter 'q' to return to previous menu
Number of Physical Storage Drives found: 3
Drives found:
ID:     Drive:          Size:
[0]     /dev/hda        80.0 GB (Model: IC35L090AVV207-0 Serial No: VNVC02G3E7AB
JT)
[1]     /dev/sda        300.0 GB (Model: Unknown Serial No: Unknown)
[2]     /dev/sdb        2004 MB (Model: Unknown Serial No: Unknown)
> 0_
```

```
#### Select Partition ####
Please select a partition or enter'q' to return to previous menu.
Number of Partitions on all drives: 4
Partitions found:
ID:     Partition:      Size [Free / Total] [Type]
[0]     /dev/hda1       [32.3M / 39.1M] [Unknown]
[1]     /dev/hda2       [49.6G / 74.5G] [HPFS/NTFS]
[2]     /dev/sda1       [279.3G / 279.4G] [HPFS/NTFS]
[3]     /dev/sdb1       [1.8G / 1.8G] [FAT32]
> 2_
```

**Fig. 3** Setting Source and Destination in OSFClone. OSFClone uses a command line interface to set all the parameters. Initially, to set the source, type “1” and press enter (as shown in panel (**a**)). You will be presented with a screen similar to panel (**b**). Most likely you will select the option corresponding to /dev/hda—confirm this based on the device size. Then you will be back at panel (**a**) and you can select option 2 for the destination. Presented with a screen similar to Panel (**c**), you will probably select the option for /dev/sda1 or /dev/sdb1; confirm this by matching the size of your external hard drive to the option you wish to select

(k) Start the **dd**. After it is complete, shutdown the computer and remove the external drive containing your image.

3. Prepare the host computer to run the virtual computer

   (a) It may be required to enable virtualization extensions within the BIOS; this is often indicated by the error message "VT-x is not available." The process to enable Intel virtualization extensions is described in **Note 5**.

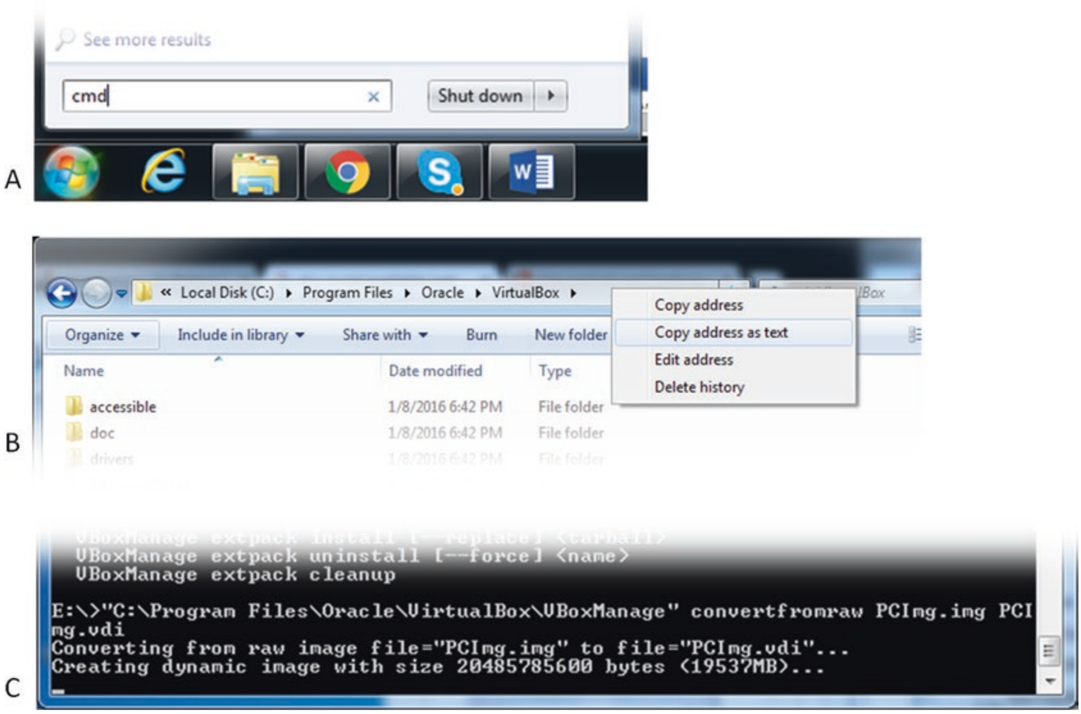
   (b) Boot the host machine and install VirtualBox which can be obtained from http://www.virtualbox.org/ (Ubuntu install described in **Note 1**).

   (c) If you have USB based peripherals, you will also want to download and install the VirtualBox extension pack from virtualbox.org; the downloaded file is then installed from within VirtualBox -- File->Preferences->Extensions. (**Note 1** also describes how to match the extension pack version in Ubuntu.)

   (d) Attach your external hard drive that contains the dd image file.

   (e) Convert the hard drive image file to a VirtualBox image. This is best done from the command line. In Windows, this process is described below (*see* **Note 6** for the Linux equivalent commands).

   - Press the "Start Button", type "**cmd**" and press enter to open a command line interface (Fig. 4a).
   - Execute the VBoxManage command to perform the conversion. The concept is:
   - *VBoxManage convertfromraw image.img out-drive.vdi*
   - where "VBoxManage" is the name of a command installed when you installed VirtualBox, "convertfromraw" is the action you want the command to perform "image.img" is the name of the image that you created and out-drive.vdi is the name of the VirtualBox compatible hard drive. All of the commands and file names will need to reference the full path to the folder where the commands or image files reside. It is easy to get the full path by navigating to the directory within the Windows file explorer and then right clicking and selecting "Select address as text" (Fig. 4b). Below is an example of the command as it might be typed. Note that the VBoxManage portion had to be in quotes because the path had a space in it. At the **C:\>** prompt type:
   - "C:\Program Files\Oracle\VirtualBox\VBoxManage" convertfromraw E:\diskcopy.img D:\HD.vdi


A
See more results
cmd
Shut down
B
« Local Disk (C:) ▸ Program Files ▸ Oracle ▸ VirtualBox ▸
Organize
Include in library
Share with
Burn
New folder
Name
Date modified
Type
accessible 1/8/2016 6:42 PM File folder
doc 1/8/2016 6:42 PM File folder
drivers 1/8/2016 6:42 PM File folder
Copy address
Copy address as text
Edit address
Delete history
C

```
  VBoxManage extpack install [--replace] <tarball>
  VBoxManage extpack uninstall [--force] <name>
  VBoxManage extpack cleanup

E:\>"C:\Program Files\Oracle\VirtualBox\VBoxManage" convertfromraw PCImg.img PCI
mg.vdi
Converting from raw image file="PCImg.img" to file="PCImg.vdi"...
Creating dynamic image with size 20485785600 bytes (19537MB)...
_
```

**Fig. 4** Converting the dd image file into a VirtualBox image file. Panel (**a**) demonstrates how to open a command line to type commands. Press the "Start" button in the lower left, type "**cmd**" and press enter. Panel (**b**) demonstrates how to copy the complete path for the command line from within Windows Explorer. Panel (**c**) shows a sample of how the command window looks when executing the VBoxManage command under Windows 7

- replacing the paths with your paths; this assumes that the removable hard drive with your disk image was mounted as the E: drive and the path where you want to store the VirtualBox hard drive image is just D:\ and that VirtualBox was installed in C:\Program Files\Oracle (Fig. 4c).
- It is recommended that the.vdi image be placed on a physically installed hard drive, or, if it is to be on an external USB hard drive that USB 3.0 be used.

4. Start VirtualBox and select the option to create a new virtual machine. Select the appropriate type (e.g., "Microsoft Windows") and version (e.g., "Windows XP (32-bit)"). For memory, a reasonable starting setting is to match the setting on the machine you are virtualize—all things being equal, 2 or 3 GB is often a good starting point for XP machines. Note that for Windows 98, a maximum 512 MB memory setting avoids some compatibility issues. When asked for hard drive options, select the .vdi file that you created above.

5. Before starting the new virtual computer, connect all USB peripherals you want to control. In the VirtualBox list of virtual machines, select the machine you created, click on settings, click on USB. On the right hand side, click the icon of the USB connector with the "+ "symbol. Add all the devices that should be attached to that computer (e.g., USB based LC Pumps). If you are using USB to RS-232 interfaces, do not forget to connect these as well. You do not need to add mice or keyboards as they are handled in a separate fashion by VirtualBox.
6. Set up the network interfaces. For most machines, the default settings (e.g., one attached interface set as "Attached to: NAT") is sufficient. This will allow your virtual computer to share the host network connection in a similar fashion to connecting a computer to a router that is attached to your home internet feed. In our example, however, the Thermo-Fisher mass spectrometer uses a second network interface for control. To make this work properly in the virtual machine, select Network->Adapter 2 and then set it to be "Attached to: Bridged Adapter" and then select the network interface attached to the mass spectrometer router. Note that for Windows 98 installations, selecting the network interface type as "PCNet-PCI II" will usually work better than the default option.
7. Start the Virtual Machine; if Windows hangs on a black screen, power off the virtual machine and try enabling **I/O APIC**. In the VirtualBox list of virtual machines, select the machine you created, click on settings, I/O APIC will be found by navigating to System->Motherboard->Extended Features.
8. If Windows asks you to reactivate, use the license key that is associated with your installation. **Note** 7 provides some pointers if Windows does require activation and your license key does not work.
9. Install the guest additions. The guest additions allow your virtual machine to more gracefully handle the graphics display as well as sharing keyboards, mice, and files with the host computer. Installation is performed on the running guest machine by choosing the "Devices" option from the menu bar, selecting to insert the "Guest Additions CD" (*see* Fig. 5) and then installing the software from within your virtual computer. You may be prompted to do this automatically after you "insert" the CD, or, you may have to navigate to the CD folder to execute the install script.
10. Check your system application settings to make sure they are interfacing to the appropriate network cards and RS-232 COM ports.

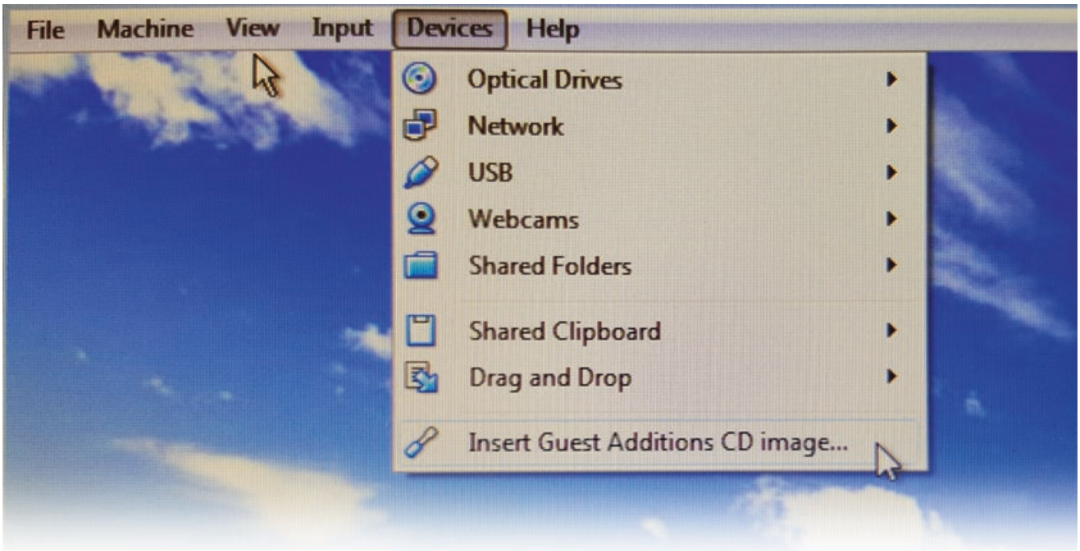


**Fig. 5** Installing Guest Additions in Your Virtual Machine. Within the started virtual machine, select Devices->Insert Guest Additions CD. Then install the guest additions as prompted

11. Enjoy your easy to manage new virtual instrumentation control computer (**Note 8** describes how to export your new virtual machine suitable for rapid redeployment)

## 4 Notes

1. Ubuntu install of VirtualBox.
   (a) VirtualBox exists in the software repository for Ubuntu. While this facilitates installation, this is often not the most current copy of VirtualBox that is available; this will usually not present a problem, but, it does mean that you will have to look through the download archives to find the appropriately version-matched extension pack.
   (b) First, open a terminal window. Clicking on the top left icon to open the Dash, type **terminal** and click on the resulting icon. Or use the hotkey shortcut of "**ctrl+alt+t**".
   (c) Type "**sudo apt-get install virtualbox**" and enter your password when prompted.
   (d) Start up virtual box (*VirtualBox*), check the version, download the extensions pack from http://www.virtualbox.org/ that matches the version you installed from the repository.
2. Imaging a removed hard drive on a Linux computer.

(a) Remove the hard drive from the computer to be virtualized.

(b) Open a terminal window on your Linux machine (*see* **Note 1(b)**).

(c) Become "root" (e.g., type "**sudo /bin/bash")**.

(d) Using your USB based external drive reader, attach the hard drive to the Linux machine.

(e) Figure out which device id was assigned (type **dmesg** and look for the last message, you will see something like "/dev/sde" referenced—sda is the first hard drive, sdb the second, and so forth. In this example, we will assume that this drive was assigned as /dev/sde.

- If you are booting Linux on the machine to be virtualized, this is often /dev/sda.

(f) Type **mount** and see if any partitions from the live drive were auto-mounted, if so, unmount them—e.g., **umount /dev/sde1** repeat for all mounted sde partitions.

(g) Connect the external USB drive that you acquired to hold the copy of the hard drive of the live computer.

(h) Figure out which device id was assigned (type **dmesg** and look for the last message, you will see something like "/dev/sdf" referenced. In this example, we will assume that this drive was assigned as /dev/sdf.

(i) Type **mount** and note the partitions from sdf (in this example) that have been mounted and where. Often this will take the form of /media/user/xxxx-xxxx-xxxx-xxxx/. If sdf has not automounted its partition, you can do this by hand: **mkdir /tmp/out; mount /dev/sdf1 /tmp/out** This assumes that the first partition on the external USB drive is the one you want to use and it was assigned as "sdf".

(j) This is dangerous. If you are unsure of the device and path names, you can accidentally overwrite data! Proceed with caution. From the root command prompt type the imaging command:

```
dd if=/dev/sde of=/tmp/out/file.dd bs=4M
    conv=noerror,sync
```

where "/tmp/out" should be replaced by the path identified in step 2.9. The "conv" options are meant to intelligently handle read errors if they were to arise; noerror means the copy will continue and sync means to replace error reads with the appropriate number of 0's to match the amount of data that was not correctly read.

3. Imaging a removed hard drive on a Windows computer.

(a) Remove the hard drive from the computer to be virtualized

(b) Using your USB based external drive reader, attach the hard drive to another Windows machine

(c) Connect the external USB drive that you acquired to hold the copy of the hard drive of the live computer

(d) Download and install the HDD raw copy tool from HDD Guru (http://hddguru.com/software/HDD-Raw-Copy-Tool/). I use the "portable Windows executable" that does not require installation.

(e) Run HDD Raw Copy, select the source hard drive—this will easily be findable because it will be a "USB" device and have the size you are expecting. For destination, select "FILE" and navigate where you want to store the image. **For file type, be sure to select "Raw Image (dd image)" versus compressed image**.

(f) Initiate the copy.

(g) Cleanly unmount the external USB drive and the mounted hard drive. If you are uncertain of how, you can just shutdown your computer.

(h) Remove the USB drive containing the hard drive image and remove the mounted hard drive. The drive with the disk image does not need to be removed if it is on the machine you are going to use as the host computer.

(i) If desired, reinstall the hard drive in the original physical computer.

(j) Windows does provide a tool to convert a physical device to a .vhd image file ("disk2vhd"), however, .vhd files are not the best to use with VirtualBox so conversion to. vdi-will still be required.

4. Configuring BIOS to boot from CD or USB before the hard drive.

   (a) For most instances, the example show in Fig. 1 will be sufficient—press the prompted for key and then select either the USB or CD as appropriate.

   (b) For some versions of the BIOS, you will need to find the options for the boot order and then set the CD or USB to boot before the hard drive.

5. Configuring BIOS settings to enable virtualization extensions

   (a) If you are running an AMD CPU, hardware virtualization is usually turned on by default. Intel machines tend to be a mix of default enabled and default disabled.

   (b) To check and change settings, you need to be in the BIOS configuration screen. Turn on the host computer and, on the initial splash screen, select the option to activate the "Setup"/BIOS menu. This is often the "F2", "DEL" or "ESC" key—look at the splash screen to see if they tell you the appropriate key; you can also perform an Internet search for the make and model of your computer. This can vary considerably, for example, for some Lenovo laptops, there is a little physical button that needs to be pressed during the boot process.

   (c) BIOS menus also vary considerably, you are looking to enable "Intel VT-x" aka "Intel Virtualization Extensions" aka "Intel Virtual Technology".

6. Converting a dd image into a vdi file in Linux.

   (a) Open a terminal window.

   (b) Become "root" (e.g., type **sudo /bin/bash**).

   (c) Type **VBoxManage convertfromraw /media/user/xx/image.dd ~user/out-drive.vdi** replacing the paths to the input and output file with the appropriate fully qualified pathnames.

7. Windows reactivation

   (a) Unfortunately, the new virtualized Windows will sometimes require reauthentication. This is complicated because Windows XP and earlier are no longer supported by Microsoft. Often the reauthentication can be accomplished by using the original license key. If, however, the original key will not validate Windows, there is no supported reactivation option. Many people have found success in a work-around that involves editing the Windows registry, resetting the countdown timer and then preventing the operating system from updating the countdown timer value. There are many sites on the internet that describe this process [3–5].

8. Comments about managing your new virtualized computer.

   (a) Some functions may work better if you install the VirtualBox guest additions. In the running virtualized computer select Devices->Insert Guest Additions CD Image. The image will be downloaded and inserted into the virtual CD drive. At this point, you will either be prompted to install the software, or, you can select the CD drive from within the virtualized computer.

   (b) Backups and redeployment are easy. With the virtual computer shut down, in the VirtualBox main console, select "File->Export Appliance". This exported appliance (i.e., a single file that contains everything required by the virtual image) is easily imported into other VirtualBox installations. The exported files can be archived and instantiated as needed.

(c) Recovery from configuration mistakes and from viruses is now trivial! After your machine is running, shut down the machine then, from the VirtualBox main console, select "Machine->Clone". To save disk space, select a "linked clone". Then, always use your new clone instance of your virtual machine. If a configuration gets corrupted, just start up the original instance and all the changes are undone.

## Acknowledgments

I thank David B. Agus, The Lawrence J. Ellison Institute for Transformative Medicine at USC and the USC Center for Applied Molecular Medicine for generous institutional support. Also, I thank Lara Bideyan for her critical review of this document and the testing of the protocols described within.

## References

1. VirtualBox download site: https://www.virtualbox.org/wiki/Download
2. https://www.virtualbox.org/wiki/Migrate_Windows
3. http://community.spiceworks.com/how_to/show/3381-how-to-fix-windows-xp-activation-after-a-windows-xp-repair
4. http://www.americancomputerenterprises.com/downloads/how_to_activate_windows_xp_witho.htm
5. http://www.technotrait.com/2012/02/27/activate-windows-xp-without-genuine-key/