

# CloneVDI v4.01 Release Notes

---

Copyright (C) 2020, Don Milne.

All rights reserved.

See LICENSE.TXT for conditions on copying, distribution, modification and use.

Don Milne (aka user mpack on forums.virtualbox.org)

12<sup>th</sup> February 2020

**PLEASE READ THIS DOCUMENT CAREFULLY, IT SHOULD CONTAIN ANSWERS TO MOST QUESTIONS.**

But, if you can think of anything else that should go in here then do let me know.

Intended audience: users of Oracle's "VirtualBox", a Virtual Machine application.

**Addendum:** CloneVDI v1.20 added VHD read support, v1.30 added VMDK read support, later versions added other formats. However I do not intend to go through this document replacing every occurrence of "VDI" with "VDI or VHD or {etc}". You may assume that, unless I specifically say otherwise, then anything you can do with a VDI as input, you can also do with the other formats as input. That specifically includes keeping the old or generating a new UUID, compacting while copying, and enlarging the drive. The only thing to remember is that the output from CloneVDI is always a dynamically expanding VDI, the other formats are supported for reading only.

**USAGE INSTRUCTIONS FOLLOW THE FAQ SECTION.**

## Legal Disclaimers

Basically, you got this tool for nothing, so I accept no responsibility for anything that goes wrong. While I made every effort to make this tool safe and reliable, I offer NO guarantee that I succeeded. The software may be used by anyone at his or her own risk.

## FAQ

Q. What does CloneVDI do?

A. At a basic level, it makes copies of (i.e. clones) Oracle VirtualBox Virtual Disk Image (VDI) files. It can also convert Microsoft's VHD and VHDX formats, VMWare VMDK, Parallels HDD (v2), raw disk or partition image files, and even physical Windows drives - into a VDI file. Note that CloneVDI is not a GUI for VBoxManage, it is entirely stand-alone, using my own code for reading and writing VDI, and for reading the other supported formats. This means that CloneVDI can work when VBoxManage is not working or not even present.

At a less basic level, the primary focus of CloneVDI today is the things it can do to a disk image file **while** it is being cloned, such as optimizing the block order, compacting the filesystem, or enlarging the logical drive size.

Q. How do I install it?

A. It came as a .zip, just create a folder and unzip the contents into it. After that you can optionally create a shortcut to link your desktop to the CloneVDI.exe file inside the target folder.

Q. What's the point? - VirtualBox has cloning built in as standard!

A. When CloneVDI was first created in 2009, VirtualBox only supported cloning from the command line, which many users disliked. I also wanted to add a number of diagnostic and optimization features that VirtualBox did not provide. And, I wanted the flexibility to add more features in the future, independent of what the VBox developers chose to do.

In particular :-

- I wanted a simple GUI interface (I later added a command line interface as well, to satisfy those strange people who like that sort of thing!).
- I wanted better feedback from the tool, in particular an accurate answer to the question, "so how long is this going to take?".
- Speed. I wanted to see if cloning could be any faster. In a test I performed, the "VBoxManage clonehd" command took 101s to clone a 1.6GB VDI. My CloneVDI tool did the same in 68 seconds. But, I find that Windows disk I/O speeds are highly variable, so that could have been dumb luck.
- I wanted more flexibility regarding the UUID (a signature embedded in VDI and most other virtual disk formats, which VBox uses to identify them). By default I generate a new UUID for the clone (as "VBoxManage clonehd" always does), but I also have an option to keep the old UUID. USE THIS FEATURE WITH CARE - ALWAYS GENERATE A NEW UUID UNLESS YOU KNOW WHEN/WHY THE OLD UUID CAN/SHOULD BE RETAINED.

- I wanted more flexibility regarding the drive size. CloneVDI has an option to increase the virtual drive size of the clone, without disturbing the existing partitions. There is a separate option to expand the main partition too (for certain filesystems only) to occupy the increased space on the drive.
  - ==> VirtualBox added a limited disk "resize" feature to VBoxManage v4.0.0, however I feel that CloneVDI's version is still preferable for many reasons: the short form of which is that CloneVDI is more capable and yet easier to use.
- I wanted to try optimization of the clone on the fly, i.e. detect blocks which are no longer used by the guest filesystem, and discard those from the clone. This feature requires me to interpret the contents of the drive, which entails more risk of an error and is therefore optional (see the "Compact while copying" checkbox). The easier and entirely safe optimizations already performed by "VBoxManage clonehd", such as defragmenting the block order and skipping zero blocks is of course also done by CloneVDI, regardless of the setting of the compact option.
- I wanted to try other optimizations, such as adjusting the target VDI header size to ensure that guest filesystem clusters are aligned on host cluster boundaries, for faster disk I/O.
- People sometimes have problems getting VBoxManage to perform a particular cloning operation. When that happens it is useful to have an independent tool you can try instead.

VirtualBox finally added GUI support for cloning (both VMs and bare disks) in v4.1.0. This represented an excellent step forward, and for most everyday uses the integrated VirtualBox feature is now more convenient to use than the CloneVDI original. However, CloneVDI continues to excel when cloning disks in unusual circumstances, e.g. when the VBox media registry refuses to accept a disk, or it has a broken snapshot chain, or when the user wants to preserve the UUID, or when the user wants to perform some other operation at the same time, such as compaction or disk enlargement. And, CloneVDI continues to serve as a useful "second source" for cloning and backup making tasks.

Q. Why clone at all? - Why not use Snapshots instead?

A. By all means use snapshots if you like them. But, snapshots worry me. I worry that they are inherently fragile: a series of links in a chain, and if any link in the chain is broken then the chain itself may be rendered useless. By contrast a clone is entirely independent of the original on which it was based. Damaging the original will not affect the clone at all (or vice versa). Also, it's easier to move VDIs to a different host, or to a backup drive, if they don't have snapshots. Finally, it's easier to switch back and forth between numerous variants of a VM if they are done as clones.

Q. Will it corrupt my virtual disks?

A. I don't see how. CloneVDI does not modify the original file (it always makes a copy, then modifies the copy), so you can always revert back to the original provided you haven't deleted it. If you do intend to delete the original then I suggest that you use the clone for a while first, to make sure that everything is stable.

As far as I know, I don't corrupt the clone either. However nobody is immune to bugs, and it's quite possible that if you try something unusual then you'll hit a problem. If you find a repeatable bug then do please let me know via the "CloneVDI tool - Discussion & Support" topic on [forums.virtualbox.org](https://forums.virtualbox.org).

Q. Can it clone VDIs with snapshots?

A. Yes, CloneVDI v2.00 now has the ability to clone a snapshot VDI (the .VDI files with the strange names that live in your \<VM>\Snapshots folder). It will follow the chain of dependencies back to the base VDI and then create a clone which includes all the data from the point in time represented by the snapshot. This feature originally worked with VirtualBox native (VDI) snapshots only. Support for cloning VMDK and VHD snapshot chains was added in CloneVDI v2.10.

Snapshots in other formats (such as Parallels .hdd) are still not supported.

Q. Does it include any spyware, adware, trojans - or anything of that ilk?

A. No.

Q. Does it modify the Windows registry, modify system files etc. etc.?

A. No. It does not use the registry, and the only files it modifies are the ones you tell it to modify, plus its own .INI file (stored in the same folder as the .exe).

Q. Do you have a version that runs on Linux?

A. I know very little about software development or distribution on a Linux platform. However as of v1.42 the source code has been released under a permissive FreeBSD-like license, so perhaps someone else will do a port. In the meantime I know that the tool works well under Wine on the popular Linux distros<sup>1</sup>. This makes sense, as CloneVDI is very much a "vanilla" Win32 application. It may also be possible to run CloneVDI inside a Windows VM, accessing VDIs via shared folders, although I've heard reports that this does not always work - CloneVDI doesn't care what folder a file

---

<sup>1</sup> I have since been able to confirm that CloneVDI works well under Wine on Mac OS X hosts as well.

is in, so there may be something about the VM shared folder interface that doesn't like very large files.

Q. What about non-English language versions?

A. Version 1.42 added localization support, with initial support for English, Dutch and German. V2.01 added a French translation. If you would like to contribute a translation then please see the file "CloneVDI.rc" in the source code, as well as the files "resource\dutch.rc" and "resource\german.rc" to see how others have done it. If you do a translation then please let us know in the discussion topic at [forums.virtualbox.org](http://forums.virtualbox.org). For now I'm happy to do the work of integrating these translations with the other code.

Q. Why do you have a copyright notice? Am I not allowed to make copies, give to friends etc?

A. A copyright notice is a standard feature even on free and open source software, because that is the only way the terms of distribution can be controlled. See the file LICENSE.TXT for an explanation of those terms.

Q. Do you plan to make it open source?

A. Already done. The software became open source in v1.42, under the terms of a permissive FreeBSD-like license.

Q. What is this UUID thing that the dialog box mentions?

A. Simply put, the header of every VDI file includes an identifying number, called a UUID, which VirtualBox uses to identify VDI files and tell them apart. VirtualBox requires every registered VDI to have a unique UUID, and new VirtualBox users often encounter the problem that if they simply use the host operating system file copy function to copy a VDI then VirtualBox refuses to let you register the copy, because of conflicting UUID signatures. VHD files also have UUIDs, VMDK files usually (but not always) have them too. RAW files obviously don't have UUIDs, nor do Parallels .HDD images. If the input format doesn't have a UUID then CloneVDI has no alternative but to generate one, and the "Keep UUID" option would be ignored.

Q. I've used CloneVDI to clone my source disk. How do I run the cloned VM under VirtualBox?

A. In fact you don't have a cloned VM, all you have is a cloned hard disk which you are now free to either mount in an existing VM, or create a new VM around. A more precise answer to this question depends on how and why you made the clone.

- If you cloned a source VDI to a destination VDI with the same name, same UUID, and in the same folder then you don't need to do anything else - from now on VirtualBox uses the new VDI in all VMs which used the original VDI.
- If you want to create a new VM around the cloned VDI then do the following :-
  1. If the clone is still called "Clone of xxxx" then rename it to something more meaningful that reflects the purpose of the new file.
  2. Run VirtualBox and tell it to create a new VM; when you get to the hard disk creation step then simply select "Use existing" and use the offered dialog box to browse to the cloned VDI.
  3. NOW STOP AND THINK - DO NOT RUN THE NEW VM YET. Instead go carefully through the settings, making absolutely sure that the old and new VMs have the same settings, especially for network MAC address, hard disk controller type and IO APIC setting. Failure to do this properly may result in the guest failing to boot, or (with Windows guests) you may have to reactivate - which is a big deal since your other VM is presumably already activated with a different hardware profile. [ If you make this mistake then your only fix is to clone the original again - and get the settings right this time before you boot! :-) ]
- If you intend the clone to run on a different host then the previous paragraphs still apply, but the UUID issue is less important.

Q. Does this tool encourage copyright violation or software license infringement?

A. I've added this question because at least two people have suggested that it might. My answer is that this tool is no more or less of an encouragement to infringe copyright than is the "file copy" function provided by your host operating system. The only difference is that this tool tries to produce an optimized copy rather than an identical copy. I don't see how an optimization step encourages anyone to infringe any law or license. For the avoidance of doubt I do NOT condone the use of this tool to violate either copyright law, or the terms of a software license.

Q. I gave it a fixed disk and the clone was dynamic! I couldn't see an option to force the fixed format.

A. I don't believe that the fixed format is necessary or useful, and I don't want to encourage novices to consider it a legitimate option, hence I don't support it for output. The very idea of compaction is useless on a fixed drive, and cloning itself is much less appealing if every clone occupies a substantial fraction of the host drive. My design style is to keep the interface simple, if necessary by omitting features which I see as not very useful. If you disagree then you are free to continue using the VBoxManage tool or perhaps write your own tool.

## USING CLONEVDI

Just run the executable and a simple dialog box is displayed. The fields on this dialog box are discussed below.

### Source and Destination filenames

**Source filename:** enter the full path and filename of the original VDI, VMDK, VHD, VHDX, HDD or RAW file. There is a Browse button beside this field to help you find the right file. Once you have selected an existing file then several other fields on the dialog will be enabled.

**Dest filename:** the full path and filename you would like for the clone. Once you select a source file <filename> then this field defaults to "Clone of <filename>". The extension is forced to "VDI", since CloneVDI currently cannot write other formats.

NOTE: it is perfectly ok for the source and destination path\filenames to be the same. In that case the tool will first write the clone to a temporary file. Then, if all goes well it will clean up by renaming the old file to "Original <filename>", then it renames the new file to <filename>. If you do this then you probably also want to use the option to keep the old UUID, that way VirtualBox Media Manager should not notice the change. [ Obviously this does not apply when converting VHD,VMDK or RAW, since the output filename will always have a VDI extension, hence the filenames can never match ].

CloneVDI remembers the contents of the source and destination fields between sessions.

### Options

Select "**Generate new UUID**" if you intend to use both original and clone on the same host PC. Selecting "**Keep old UUID**" is a possibility if you plan to delete the original file or use the clone on a different PC. There are many other circumstances in which you may prefer one or other of these options. If in doubt then "Generate new" is probably safest (that option is compatible with VirtualBox "clonemedium" behaviour). Note that if you use the "Keep" option then *only* the creation UUID is inherited by the clone. The "modification" UUID is given a new value. If you were unwise enough to clone and replace the base VDI for a snapshot chain then this should ensure that the error is not catastrophic (VirtualBox should complain - correctly - that the base VDI has been modified, at which time you should restore the original VDI).

Select the "**Increase virtual drive size**" option if you want to (err..) increase the virtual drive maximum size. If this option is selected then you can enter a new drive size in the edit box beside it. Normally CloneVDI expects the new size to be at least as large as the old size, however starting from CloneVDI v3.00 you may now enter a smaller size which CloneVDI will honour, provided it is large enough to encompass the last partition on the drive. **WARNING WARNING:** If you tell CloneVDI to shrink a drive, it will shrink a drive. On your own head be it! If something has hidden data beyond the end of the partitioned areas of the drive then that something will probably stop working: GPT partitions, Windows Dynamic Disk, and some software dongle schemes may all store data beyond the partitioned areas.

The "**Increase partition size**" option is only available when the "**Increase virtual drive size**" option is selected. If the "**Increase partition size**" option is selected then CloneVDI will automatically increase the size of the main partition to fill the new drive size. This feature only works at present on FAT16, FAT32 and NTFS partitions, however it has side effects which may be useful to you even with other guest filesystems (see the notes on increasing partition size later on).

The "**Increase partition size**" feature works in a simplistic way, and doesn't work at all for some filesystems. An alternative is to let CloneVDI enlarge the drive, then run a third party tool inside the VM to adjust the partition size as required. I know that the free "GParted live CD" works well for this: see "<http://sourceforge.net/projects/gparted/files/gparted-live-stable/>" – download the latest ISO and mount it in your guest as a virtual CD, then boot up the VM, after ensuring that the CD/DVD drive comes first in the boot order.

The "**Compact drive while copying**" checkbox enables a feature whereby the tool identifies unused blocks(\*) in the guest filesystem, and omits those from the clone. This would be most useful if a large amount of file data has been deleted inside the guest. This saves you from having to run SDelete or its Linux equivalent before making the clone, to get the same result. This feature only does something if CloneVDI recognizes the guest filesystem, and at present the only supported filesystems are NTFS (tested with NT4 and later), EXT2/3/4, FAT16 and FAT32. Also, I can only recognize the filesystem inside a partition if the disk uses a traditional MBR partition map or GPT partitions. I also have experimental (not heavily tested) support for Windows Dynamic Disk. I don't yet support Linux LVM partitioning.

(\*) All the clusters that fall inside a VDI block must be unused in order for the block to be considered unused. VDI blocks are usually 1MB, which typically covers 256x4K guest clusters. You might get better results by defragmenting inside your guest VM first.

## Buttons

"**Partition Info**" shows the partition map from the MBR of the source VDI.

"**Header Info**" shows information taken from the VDI, VHD or VMDK header. RAW files obviously have no header to show, but CloneVDI internally treats a RAW disk image as a VMDK with a single flat extent, so what you will see is the synthesized VMDK header.

"**Sector viewer**" lets you view source disk sectors in hex format, and/or dump sectors to a file.

"**Proceed**" tells the tool to begin the cloning operation.

"**Exit**" causes the application to close.

## Resizing Partitions

As mentioned above, if you select the "Enlarge disk" feature then you also have the option to select the "Increase partition size" feature. These two features in combination will :-

1. Enlarge the drive to the selected size.
2. Identify the main partition on the source drive.
3. "Slide" any partitions which follow the main partition up to the end of the drive.
4. Increase the size of the main partition filesystem to use the space made available after step 3. (Step 4 only happens for FAT16, FAT32 and NTFS partitions).

Notice that in step 2 I say "main partition" rather than "boot partition". With modern boot managers the fact that a particular partition is marked as bootable is no longer a reliable indicator that this is the main partition. Hence, what I do instead is assume that the largest partition on a drive is the main partition, and that is the partition I enlarge further. Although I expect this assumption to be true in most cases there will of course be times when it is not, in which case you will need to use a third party partition manager, as in previous versions of CloneVDI. I could have added dialogs to prompt the user for this information, but I didn't want to complicate the user interface.

Although complete functionality (including step 4) is only available for FAT and NTFS partitions, steps 1-3 may be useful even for other filesystems. For example a Unix hard disk consisting of a main partition followed by a swap partition will have the swap partition moved to the end of the enlarged drive, making it easier for a third party partitioning tool to enlarge the main partition.

Sorry Linux users, but ExtX partitions cannot be expanded using this feature. I did originally intend that "Increase partition size" would work for ExtX partitions too, but it turned out to be harder to do than I expected, much harder than for the other supported filesystems. It's possible that I simply don't "get" ExtX well enough to see how it could be done easily.

## VDI Repair

Starting with v3.00, CloneVDI now has the ability to repair some common forms of corruption in VDI headers. Please note that this feature **ONLY** applies to VDI. You must look to other tools if you have a corrupted VMDK, or VHD etc.

For several years now I have been regarded as the guy to go to when you have a corrupted VDI. I have become familiar with the regular patterns in a VDI header, plus the most common forms of corruption that occur. This has allowed me to be able to guess what the lost header should have contained. I do not have the same intimate "feel" for other image formats, which is why this new feature only exists for VDI.

I have now transferred my techniques into code so that the procedure can be automated.

**THIS FEATURE REPRESENTS A BEST EFFORT TO RECOVER YOUR DATA, BUT NOTHING IS GUARANTEED.** Test the clone thoroughly before relying on it. And if you want to be *really* sure of recovering your data then nothing beats a recent backup copy.

As background, a VDI consists of a pre-header mostly containing a signature string, then a true header containing important information about the VDI (UUIDs, offsets and disk sizes), then a blockmap (which lists positions of image blocks), then a number of 1MB disk image blocks containing the data on the virtual drive. In a dynamic VDI not all blocks need be present, nor are they required to be in linear order: the blockmap tells you how to untangle them.

CloneVDI now has the ability to test for damage, and if necessary replace, the preheader and header. It cannot repair damage to the blockmap or image blocks.

If only the preheader needed repair then the cloned VDI is as good as the original.

If the header proper needs repair then the replacement has to be given new UUIDs: you will have to unregister the old media in VirtualBox and register the new clone. Also some guest software can be affected by the change of UUID, e.g. you may have to edit the grub boot script for Linux guests.

Happily most cases of VDI corruption only seem to affect the first cluster of the file (typically the first 4K), so header repair nearly always works.

**NOTE FOR THE UNWARY:** If you accidentally deleted your VDI and tried to recover it with an undelete tool and now come looking for CloneVDI to magically make the recovered VDI work, then you are probably wasting your time. In my experience "undelete" tools do not work with very large files (multiple MBs), and the people selling them for that purpose are snake oil merchants, pure and simple. In the cases I've seen, the file content from beginning to end was complete garbage - not a single byte of the supposed VDI actually came from a VDI, and so, naturally, nothing can be done with it.

**TO USE THE REPAIR FEATURE:** select the source VDI as normal. The usual validation result will confirm that the file is corrupted. Note also that the "Proceed" button has changed to "Repair". So click the Repair button. CloneVDI will attempt to infer a replacement header. If it succeeds then it will continue to clone the source VDI, and the clone will have a repaired header. If repair is not possible then you'll get an error message and cloning will be aborted.

Other notes:

- CloneVDI defaults to "Keep UUID" when repairing a VDI. Obviously this only applies if the old header was recoverable (damage to preheader only).
- If the source header is repaired then cloning proceeds normally, so you are free to set other CloneVDI options, such as Compact, Resize etc.

## Cloning Snapshots

(NOTE THAT THIS SECTION CHANGED SIGNIFICANTLY FOR CLONEVDI VERSION 2.04)

(NOTE2: THIS SECTION TALKS PRIMARILY ABOUT VDI, HOWEVER VHD AND VMDK SNAPSHOTS HAVE BEEN SUPPORTED AS WELL SINCE CLONEVDI VERSION 2.10)

In this section I will use the term "snapshot" somewhat loosely (and inaccurately) to refer to any differencing VDI file, i.e. an incomplete VDI file which requires the presence of at least one parent VDI to get a complete disk image. What users generally think of as snapshot files are the ones with strange names such as "{1234.5678.abcd.etc...}.vdi" and which generally live in a "Snapshots" folder which contains all other snapshots created for the same VM. One file missing from this folder is the base VDI, which always acts as the first link in a snapshot chain, and is needed to complete the virtual disk image. In VirtualBox v4 the base VDI is stored by default in the parent folder of the Snapshots folder. In previous VirtualBox versions the base VDI was stored in your "<userdoc>\.VirtualBox\HardDisks" folder.

If you select a snapshot (child) VDI file as the source then CloneVDI will scan the remaining VDI files in the same folder to see where they fit in the snapshot chain. Also, if the folder happens to be called "Snapshots" then the parent folder will also be scanned, and any VDI files found there will receive similar treatment.

This snapshot chain reconstruction method works well with the new VirtualBox v4 folder structure, and it also provides the simplest possible method of creating a flat VDI file from a collection of snapshot files: just make sure to put them all in one folder, including the base VDI, and CloneVDI should have no problem creating a clone of any one of them.

In the CloneVDI dialog you should usually select the newest snapshot VDI as the source file, thus ensuring that the merged VDI contains the most up to date information.

If you still receive an error message that "differencing images are not supported", then it means that a valid VDI snapshot chain couldn't be created from the files you provided. Bear in mind also that only VDI, VMDK and VHD snapshot chains are supported.

## Using CloneVDI to do a P2V ("Physical To Virtual" Disk Conversion)

CloneVDI now recognizes when you have used a Windows disk device name as the source filename (for example, "\\.\PhysicalDrive0" identifies the boot drive on most Windows hosts), and slightly modifies its behaviour in that case to make it possible to clone the physical host hard disk to a VDI file. Basically, all it does is use a different method to get the "source file size", otherwise everything works as before. Be warned that this feature is extremely rudimentary, so it is up to you to use it sensibly. In particular you should have no other applications running (certainly nothing that creates and/or writes to files), and the clone should be written to a different drive. On some Windows hosts you may also need to run CloneVDI with elevated privileges ("Run as administrator") since this feature requires CloneVDI to have sector level (read only) access to the source drive.

## Virtual Disk Container Support

### VDI

As far as I know, I support all variants of the basic VDI file format, from the oldest Innotek format to the latest Oracle format. When writing I write the latest VDI format, which most recent VirtualBox versions should support. CloneVDI always writes a dynamic VDI.

### VHD(X)

VHD is supported for input only. I support fixed and dynamic VHDs using any block size.

Split VHD files are not supported: if you have a split VHD then you should download one of the many available free file split/join tools and join the segments into one file before processing it with CloneVDI. If you are comfortable with the Windows command prompt then the "copy" command can also be used to join files (do some web research first, you need to get this VERY right!).

Version 2.50 added VHDX read support. They have similar names - Microsoft clearly wants to leverage VHD brand recognition - but internally VHDX is a completely different (and better) format. At present CloneVDI has only been tested with dynamic VHDX files. Fixed size images should work as well. Snapshots are not yet supported in this format.

### VMDK

I support monolithic and split VMDK files having FLAT, ZERO or SPARSE extent types. Due to lack of clear documentation I do not yet support the strange extent types you may find on an ESX server. I do not support compressed VMDK files.

### Parallels HDD

I could find no open specification of the current Parallels .HDD format, the only information I found was for the Parallels v2 format as implemented in qemu-img (I did not use any QEMU code, I simply used them as a source of file structure information). On the upside, all of the Parallel's appliances I've found online do seem to use the V2 format.

### RAW

A RAW file is a simple uncompressed image of a disk or partition, the file must have a .raw or .img extension, otherwise (since it has no header) CloneVDI has no other way to reliably identify the file as a raw image. A whole disk image is best, but if you give CloneVDI a partition image then it will try to identify the partition contents and then reconstruct a full disk image around it (adding a new MBR and track0 at the front, and adding padding at the back to create a rounded drive size). One must bear in mind however that CloneVDI is trying to reconstruct important missing information and there is NO guarantee that the result will work. Also I must apologise to Linux fans, since this ability to convert a partition image will not work at all with a Linux partition image, because every working Linux disk I've seen has multiple partitions and puts a lot of extra data in track0 (grub boot stuff) - all of which makes it impractical to reconstruct a whole disk starting with an image of just one of the necessary partitions. Put another way, it MAY be possible, but I don't know enough about Linux to do it yet.

## Converting VHD/VHDX/VMDK/HDD/RAW to VDI

Be warned: all that CloneVDI does when you give it a VHD/VHDX/VMDK/HDD/RAW file is convert one media type to another media type. There is no guarantee that you'll be able to use the media **content** without problems inside VirtualBox. For example, let's say you convert an XP.VHD that previously ran under Virtual PC 2007: the same install of XP may not run under VirtualBox right away, due to the different hardware simulations on the two platforms. Even if XP does run it is very likely you will be prompted to reactivate.

Activation on the new platform may give you "Windows Update" problems if you continue to use the same XP license on the old platform. These same issues would arise if you mounted the source file under VirtualBox without converting.

## "Sector Viewer" feature

When developing CloneVDI I found myself often wanting a convenient way to view guest disk sectors, and when testing support for RAW files I also found myself in need of a reliable and friendly tool for creating raw files by dumping disk sectors from a VDI (usually I needed entire disks or partitions dumped to a flat file, and you may (or may not!) be amazed at how unfriendly some of the freely available tools are!). I had had a "Sector editor" feature in mind since day one (the button was always there, but grayed out), so I finally got around to implementing it in v1.40; though I decided to make it a sector viewer rather than a sector editor. Non-technical users can safely ignore this feature entirely, for you it has nothing more than curiosity value. Technical users may find it an interesting tool for analysing guest filesystems. The ability to dump sectors to a file may also be useful, for example this can be used to export an entire disk or partition to a flat file.

## Command Line Interface

Version 1.41 of CloneVDI added support for a command line interface. A total of three run modes are therefore supported:

- Normal GUI mode, which is when CloneVDI is run from the Windows desktop without arguments.
- GUI mode with command line, when CloneVDI is run from the Windows desktop *with* arguments.
- CLI mode, which is when CloneVDI is run from a command console with arguments.

For completeness, I should note that if you run CloneVDI from the command console *without* arguments then it runs in normal GUI mode. The CLI mode usage information is as follows :-

Usage: CloneVDI <source file name> {options}

<b>Option Name</b>	<b>Effect</b>
<b>-o or --output</b>	Sets output filename. A filename should follow as next argument. The output filename defaults to "\"Clone of \""+<source> if this option is not selected.
<b>-k or --keepuuid</b>	Output file inherits UUID from source. Default is to generate a new UUID.
<b>-e or --enlarge</b>	Enlarges drive max size. Intended size should be provided as next argument, e.g. 256.00MB, 64.00GB.
<b>-r or --repart</b>	Enlarge the main partition to fit extra space (ignored if --enlarge option is not also set).
<b>-c or --compact</b>	Enables compaction feature (supported guest filesystems only).
<b>-h or --help</b>	Displays this usage information.

Options can be grouped, e.g. -kce or --keepuuid+enlarge. Option parameters should follow, and the order of the option parameters should match the order of the options within the option group.

Here are some example command lines, all of which do the same thing :-

CloneVDI "My Virtual Disk.vdi" -o "My Clone.vdi" -k -e 64.00GB -c
CloneVDI "My Virtual Disk.vdi" -o "My Clone.vdi" -k -e "64.00 GB" -c
CloneVDI "My Virtual Disk.vdi" --output "My Clone.vdi" --keepuuid --enlarge 64.00GB --compact
CloneVDI "My Virtual Disk.vdi" -okec "My Clone.vdi" 64.00GB
CloneVDI "My Virtual Disk.vdi" --keepuuid+compact+output+enlarge "My Clone.vdi" "64.00 GB"

You should bear in mind that the Command Line Interface (CLI) feature is just a different way to access the features of CloneVDI, the features themselves still work in the same way. For example, it is legal to make source and destination filenames exactly the same, this would be handled exactly as if you had filled the dialog box in the same way, i.e. you will end up with a clone with the selected name, and the original file renamed to "Original "+<filename>.

CloneVDI is an example of a "dual mode" console/GUI application. This presents a small problem when running from the Windows command console: the console expects an application to be *either* a GUI application, or a console application, and not both. A field in the EXE header says which of these types a particular application is. In the GUI case the console launches the application as an independent process and does not wait for completion. In the CLI case the console does wait for completion of the command.

CloneVDI is an application which supports a CLI but is marked as a GUI app, hence the command console will normally launch it as an independent process and not wait for completion. In most cases this is no big deal: CloneVDI will do what you asked it to do (without a start dialog) and will terminate itself when finished. However any text output by CloneVDI will overwrite the next command prompt, which is not very elegant (you have to press enter once CloneVDI is done, to make CMD repeat the command prompt).

If you don't like the above behaviour then the Windows command console supports a special run command "start /wait <appname>" which forces it to wait until <appname> terminates. So, just prepend the example commands given above with "start /wait " and everything should work ok. Alternatively, running CloneVDI inside a batch file also seems to make the system wait for completion (I've provided an example batch file - see CloneCLI.bat - which you may want to use). I can't speak for how other scripting languages will handle the situation. I guess you'll let me know.

There is one small quirk that you should probably be aware of, which is that CloneVDI treats the source folder as the target unless told otherwise. For example if you type the command like this:-

```
CloneVDI "\Source Folder\Source Name.vdi" --output "Dest Name.vdi"
```

Then CloneVDI creates "Dest Name.vdi" in the source folder, *not* the current folder (if different). To

change this you need to explicitly give the required target folder in the output filename.

Incidentally, script writers may wish to know that CloneVDI returns an error code of 0 to the shell if all goes well, and a non-zero result code if there was an error. If run from the command console then there should also be an error message in the latter case.

## Notes on Language Support (Localization)

After careful thought I decided that CloneVDI would not adjust itself automatically to a locale, since I was not convinced that the preferred language need always be the same as the locale. So, if you want CloneVDI to use a different language then you need to add a "Language=n" line to the [Options] section of the "clonevdi.ini" file (or modify the option if it is already there). 'n' is a number indicating the preferred language. Currently supported numbers are 0=English, 1=Dutch, 2=German, 3=French. English is the default if the option is not present or the language number not recognized. Incidentally, there is no significance in the numbering, I simply allocate new numbers in the order that translations are sent to me.

## Other .INI File Features

The .INI file is used to store source and destination filenames from the last session, and also includes the Language=n option described above. A new addition in v2.00 is an optional line "Compact=n", which sets the startup default state of the "Compact" switch in the dialog. n=0(off) is the default if this .INI line is not present; "Compact=1" (or any other non-zero value) causes the compact option to be enabled on startup. This .INI file setting has no effect on CLI mode (compact is always disabled by default for CLI mode).

## Release History

Ver.	Date	Description
4.01	12 <sup>th</sup> Feb 2020	Fixed some crashes when certain options were used in command line mode.
4.00	3 <sup>rd</sup> Jan 2020	Added experimental support for GPT partitioned drives. This required extensive changes throughout the code and some dialogs, hence the major version bump.
3.02	17 <sup>th</sup> May 2018	Some debugging trace statements were left enabled in the command line parsing module.
3.01	24 <sup>th</sup> Dec 2017	Slight tweak to VDI repair function to make it work for more cases. Recreated icon to look better on hiDPI displays.
3.00	5 <sup>th</sup> July 2017	<p>New VDI Header Repair functionality added.</p> <p>I now allow you to shrink a drive, provided you don't make the drive smaller than the end of the last partition. TODO: needs additional checks for Windows Dynamic Disk, and GPT partitions. Thank heavens I don't do this in-place!</p> <p>Fixed an incompatibility between CloneVDI's compaction feature and the EXT4 filesystem with the INCOMPAT_64BIT flag set (e.g. Debian 9 installer now sets this feature flag by default).</p>
2.51	4 <sup>th</sup> Dec 2015	Fixed a bug which caused the block map size to be wrong when creating VDIs larger than 256GB in logical size.
2.50	10 <sup>th</sup> Oct 2015	<p>Added experimental VHDX read support. I now align the block map and the image on 1MB boundaries, as VirtualBox does - this is more suitable for SSDs, 4K sector drives, and modern OS's which no longer align partitions on 63 cylinder boundaries.</p> <p>Also a few minor fixes.</p>
2.10	9 <sup>th</sup> Sep 2012	<ul style="list-style-type: none"> <li>o Minor fixes, "theme aware" manifest now embedded. I now align the largest (rather than the first) partition on a 4K VDI file boundary.</li> <li>o Fixed a bug in the code which eliminates "between partitions" data when compacting. In rare circumstances this could discard the MBR! Circumstances were: MBR partition slot 0 empty (v rare by itself), another MBR slot not empty, compaction enabled, first actual partition begins at LBA &gt;= 2048 (i.e. 1MB or beyond). In this case the first 1MB block was eliminated.</li> <li>o Snapshot cloning now works for VMDK and VHD snapshot chains too (experimental).</li> </ul>
2.05	20 <sup>th</sup> Jun 2011	Wrong compiler structure packing option broke my VHD read code.
2.04	16 <sup>th</sup> May 2011	<p>Fixed another bug in NTFS partition enlargement code - hopefully the last important one!</p> <p>Significant changes made to how snapshot cloning is done.</p>
2.03	14 <sup>th</sup> Oct 2010	Minor tweak to validation code to recognize VirtualBox 3.2.10 (Oracle tagged) VDI headers.
2.02	1 <sup>st</sup> Mar 2010	Fixed bug in NTFS "grow partition" code. French translation completed.
2.01	8 <sup>th</sup> Feb 2010	Fixed bug in COW module which affected "enlarge partition" feature.

		Added French translation.
2.00	30 <sup>th</sup> Jan 2010	<p>New features</p> <ul style="list-style-type: none"> <li>o Support for Parallels .HDD format v2.</li> <li>o Support for cloning VDI snapshots, merging differenced data into a flat clone.</li> <li>o Rudimentary support for cloning physical drives and partitions P2V).</li> <li>o Support for Windows Dynamic Disk partitions.</li> <li>o Added .ini option to control state of "Compact" option on startup.</li> </ul> <p>Fixes</p> <ul style="list-style-type: none"> <li>o Actually, this might qualify as a new feature. CloneVDI now identifies data in unpartitioned space on the drive as unused. Previously it could only recognize unused space inside a partition which used a recognized filesystem. It simply hadn't occurred to me that unpartitioned space might contain data that the user now expects to get rid of (e.g. because the user shrank a partition using GParted).</li> <li>o Changed method of calculating time to completion to take the results of compaction into account in advance. Obviously this only makes a difference if the compact option is selected. This also means that CloneVDI calculates required disk space more accurately, no longer assuming that the clone will be the same size as the original.</li> <li>o Fixed a poor design feature which caused slow performance when converting RAW files (including physical drives).</li> <li>o Fixed a problem with handling VMDKs having a large number of extents (e.g. a 256GB drive split into 2G extents). The old code used to assume that 32 extents was enough.</li> </ul>
1.43	10 <sup>th</sup> Dec 2009	Implemented fix for a problem with enlarging drives smaller than 8GB.
1.42	12 <sup>th</sup> Nov 2009	Added localisation support (see "thanks" for translators).
1.41	31 <sup>st</sup> Oct 2009	Allow ".IMG" as a synonym for ".RAW". Added command line interface. Several minor usability tidy-ups.
1.40	14 <sup>th</sup> Oct 2009	Added read support for RAW files, plus the "Sector Viewer" feature.
1.30	7 <sup>th</sup> Oct 2009	Added experimental read support for VMWare VMDK files.
1.22	21 <sup>st</sup> Sep 2009	Bugfix: VHD read support STILL failed on files >4GB.
1.21	21 <sup>st</sup> Sep 2009	Bugfix: VHD read support failed on files >4GB.
2.20	20 <sup>th</sup> Sep 2009	Added ability to read VHD files.
1.11	17 <sup>th</sup> Sep 2009	Minor bugfix - a temp file of 0 bytes was being left behind sometimes (I never bothered to release this version).
1.10	15 <sup>th</sup> Sep 2009	Implemented "Compact while copying" feature for NTFS, EXT2/3/4 and FAT16/FAT32 guest filesystems.
1.00	8 <sup>th</sup> Sep 2009	Original version.

## Other Contributors

My thanks to the following people :-

- Sasquatch @ forums.virtualbox.org for providing the Dutch translation of the CloneVDI resource file.
- Erich N. Weitzel (a.k.a. Etepetete @ forums.virtualbox.org) for the German translation of the CloneVDI resource file.
- Gregory (a.k.a. Gregoo @ forums.virtualbox.org) for the French translation of the CloneVDI resource file.
- MarkCranness @ forums.virtualbox.org for contributing a better looking application icon.
- Makoto Matsumoto and Takuji Nishimura for making available an open source permissive licensed C implementation of their "Mersenne Twister" pseudo-random number generator algorithm. CloneVDI itself makes very minor use of this, but I have found the code useful in several applications (in fact this is now my standard PRNG module).
- Chris Waters (a.k.a. cgwaters @ forums.virtualbox.org) for going to a lot of trouble to provide me with enough information to reproduce and fix a bug in the NTFS partition enlargement code.

## Third Party Copyrights, Licenses, Patents et al...

Just one of these that I know of...

The module "Random.c" implements the Mersenne Twister pseudo random number generator. The code is a slightly modified version of an original file "MT19937.c" which I found on the web a few years ago. The authors original copyright statement is shown below.

I emphasise that the following copyright statement applies to "Random.c" only.

----- snip -----

Copyright (C) 1997 - 2002, Makoto Matsumoto and Takuji Nishimura,

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The names of its contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

----- snip -----

## LICENSE.TXT

(This is the essentially the "FreeBSD License")

The CloneVDI executable and, with exceptions noted below, all individual CloneVDI source files are :-

Copyright (C) 2010, Don Milne.  
All Rights Reserved.

The license terms are described below. These conditions apply to the application as a whole, as well as individually to each source file.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistribution of source code must retain all copyright notices, this list of conditions and the following disclaimer.
2. Redistribution in binary form must retain all copyright notices, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

### Exceptions

"Random.C" is Copyright (C) 1997 - 2002, Makoto Matsumoto and Takuji Nishimura. See Random.C itself for the author's original copyright notice and license terms.

### Also

After much thought I decided to make the following a request, rather than a requirement. If you create a "fork" of CloneVDI then please call it something other than "CloneVDI". I make no claim of ownership on the word itself, I simply don't want to be receiving support questions for software I'm not responsible for.