Primer on CD-R

The following is intended to be parts of a primer on recordable Compact Disc formats. I have favored simple expression over technical detail. For that, I urge you to check other resources, as found in the URLs. In particular, the information from vendors is authoritative, where the following is **not.** The focus is on PC's and Windows; those with Unix or Mac systems are welcome to translate to the extent possible. (I try not to write about things I don't know.)

Inputs are invited for corrections and for additional topics. Best of all would be a draft for posting here; it will get you a byline and international exposure. (Great for the resume, you know.)

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Before you Begin

If you have already begun to use CD-R, you know all of this. It is not for you. Please skip it.

If you want to argue with any of this material, please do it somewhere else.

In short, the following is all so obvious that I'm embarassed to write it down. Still ...

---

1. General

1. Most of what you 'hear' about CD-R is false.
2. A CD-ROM reader cannot write to CD-Recordable (CD-R) media.
3. A CD-R writer cannot write an erasable (CD-RW) blank unless it is designed to do so.
4. Any writer can read any type of disc which it can write. That does not mean that it can read any particular disc which it has written. The ability to read depends on the medium and its contents.
5. No CD-ROM device can read any form of DVD or DVD-ROM.
6. No CD-R device can write any form of DVD or DVD-ROM.
7. Any kind of file which can be stored on your hard drive can be copied to a CD-R.
8. A file which is too big to be written to a given blank cannot be written to that blank.
9. Certain types of information on a CD-ROM must be written in particular ways. Those include the files of Video CD (VCD) and CD Digital Audio (CD-DA).
10. An image of the contents of any drive is not the same as the contents themselves. To recreate a CD-ROM from its image, the image must be transformed by software, not simply copied, to the CD-R.

---

2. Packet writing

1. Packet writing (using UDF) requires a drive designed to write packets and packet-writing software supporting that drive.
2. Fixed-length packets can only be written to an erasable blank in a writer designed to write packets and to write erasables - and requires software made for the purpose.
3. If a drive is not supported for a given function by a given program, it cannot be used to do that job with that program. Exception: In some cases, a drive can be made to look like another; that will almost certainly void any warranty and may lead to unexpected and even unsolvable problems.
4. Packet writing cannot be used to create a CD-DA or to create an exact copy of a pressed disc.
5. If two programs require contradictory values for a parameter (such as Auto Insert Notification), they cannot both be active in a single operating system at one time. Similarly, if they require different versions of a single DLL or other component, they cannot both be active at the same time.

3. Media

1. 'Media' is a plural noun. Its singular is 'medium'.
2. Two blanks which have the same coloring may or may not behave the same way.
3. There are no absolute rules for what medium is best for any specific purpose.
4. The manufacturers and distributors of media are under no obligation to tell you what they put into their packages or to tell you when they change those contents.
5. Erasable (RW) and write-once media work in entirely different ways.
6. The upper (non-recordable) surface of a CD-R is delicate and must be handled with care.
7. Never use a pencil or a ball-point or other hard-tipped pen to write onto a CD-R.
8. In the present state of the art, two write-once media which meet the specifications may behave very differently in a recorder.
9. The various colors of write-once media indicate differences in manufacture but do not qualify one as generally 'better' than another.

4. Readers

1. Any device designed to read a CD, CD-ROM or CD-R is a reader. Every CD-R writer is a reader.
2. There is no known way to write to a specific medium with a specific writer and be sure that it can be read on all readers.
3. A reader designated MultiRead should be able to read an erasable; some which are not so designated may be able to do so as well.
4. Some readers will not read CD-R at all.
5. While there are reasons to prefer a reader using either SCSI or EIDE, each can perform all functions of the other.
6. A given reader has the right to reject any given CD-R.
7. A reader which is not a writer normally will read only a closed session.*
8. An audio player will normally read only the first closed audio session on any disc.*

5. Writers

1. While there are reasons to prefer a writer using either EIDE or SCSI, each can perform all functions of the other.
2. Many features supported by some writers may not be supported by others. Some are precluded by hardware, some by firmware. In general, a feature not supported by your hardware cannot be implemented on your hardware.

3. A hardware manufacturer may claim capability requiring software or media not supplied with the product.
4. Not all writers are manufactured by the company whose name appears on their packaging.
5. A writer not designed to write erasable media will not write erasable media.
6. A writer not designed to write packets (UDF) will not write packets.
7. In general, any modification of the hardware or firmware of a writer not sanctioned by the manufacturer will void any warranty and may result in unexpected and even unrecoverable problems.

---

6. Audio

1. There are no files on an audio CD.
2. The apparent .CDA files are a fiction of the operating system.
3. If you see an audio CD as having WAV files, you have an artificial environment which is likely to cause problems when you work with that disc unless you know what is going on.
4. Compact Disc - Digital Audio (CD-DA) is the fixed and inviolate format for sound which will play on a conventional stereo. It is 44.1 Ksps, two-channel, 16 bits, uncompressed PCM. With trivial exceptions, no flexibility is allowed and you cannot put more onto the disc and still play it back in a conventional machine.
5. WAV is a file type which mastering software will convert to CD-DA when writing a CD-DA.
6. MP3 is a file type which some mastering software will convert to CD-DA when writing a CD-DA.
7. Whether WAV or MP3, a file which your software will not convert (e.g., at an incommensurate sampling rate) cannot be written by that software to CD-DA.
8. Performance of a reader on audio is usually not the same as its performance on data.
9. Converting a 44.1 Ksps WAV file to CD-DA introduces no noise, distortion or coloration to the sound. Any differences you hear are due to your ears and your hardware.

---

7. Summary

1. When in doubt, **read.** Sources include this site, the FAQ and other sites linked from the URLS page here.
2. CD-R technology is evolving rapidly. What was true yesterday may be false today; what is scheduled for tomorrow may never happen.
3. The more you understand about an immature technology such as CD-R, the more likely you are to recover from any problem.
4. CD-R problems may arise from hardware, software, firmware or procedures. Some derive from the motherboard, the adapter, reader, writer, operating system and other hardware and software on the system. Many arise from interactions of components.

5. The most common cause of failure in writing CDs is the action or inaction of the operator.

---

* This condition is true by specification. It is possible for a manufacturer to create an exception so it may be false for a specific piece of hardware.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Make Your Own CD's

*In slightly modified form, this article was published in **User Friendly** the journal of the Los Angeles Computer Society in October 1997.*

A quiet revolution in technology began near the end of 1995 with the release of the first practical consumer CD recorders. That revolution has been continuing as both hardware and software have evolved. By now (late 1997), do-it-yourself CDs and CD-ROMs are quite reasonable goals for the home user and are routine tools for many small offices.

The recording process

A CD-Recordable (CD-R) disc differs fundamentally from a CD or CD-ROM. A 'pressed' disc is made from a glass master with the information stored as pits and lands. That master is pressed against a softened plastic aluminized to give a shiny, metallic surface which reflects the reading laser. The signal generated is very strong and unambiguous. In a CD-R, a flat plastic substrate is plated with gold and covered with a dye layer. The dye, which may be gold, green or blue, has the property that when exposed to an appropriate, strong laser light, it changes state. In one state, it is nearly transparent to infrared; in the other, it is more nearly opaque. The reading laser passes through the dye layer twice: once on the way to the gold layer, once on the way back. The resulting signal does not show as great a difference between a zero and a one as does the pressed CD, but it is still enough for most readers to decipher. The quality of the signal from a CD-R depends on the medium used, the writing laser and the reader. As a result, there is no 'best' CD-R blank or 'best' writer.

Compact Disc audio

A Compact Disc Digital Audio (CDDA or CD for short) consists of a continuous stream of bits translated in a CD player into a digitized waveform which is then converted to analog for listening. Also on that disc is a Table of Contents (TOC) which consists of tiny files describing the start point and the length of the recording. They are specified in terms of recording 'blocks' on the disc; each block is 2K long and there are about 330,000 of them on a normal CD or CD-ROM. It is not important when listening to the music, but is very important when creating a CDDA to remember that the sound is not stored in files at all, but in that stream of bits. (The book of specifications for CDDA was originally bound in red and is now universally known as The Red Book; other colors apply to standards for other formats.) A redbook sound must fit the standards in detail and be recorded at 44.1 KHz sampling at sixteen bits' resolution and two channels.

CD-ROM

A CD-ROM differs fundamentally from a CDDA in that its information is stored in conventional files of the sort familiar to computer users. One of the jobs of CD recording software is to translate a redbook bit stream into WAV files and back. Using any standard CD-R package, that conversion is transparent to the user, but for technical reasons is very important. The process of retrieving the digital bit stream from a

CDDA is called Digital Audio Extraction (DAE). Not all CD-ROM readers can do DAE; some do it only very slowly, very poorly, or both; and some do it quickly and well. Therefore, not all readers are equally good at supplying information for subsequent recording if you want to make a CD-R of favorite selections from your CD library. Incidentally, the directory to the information on a CD-ROM is not stored in a FAT (File Allocation Table) or other familiar file system; it's in the TOC. One key job of the Microsoft Extensions in MSCDEX is to make the CD-ROM's TOC look like a FAT to the operating system. That's why drivers are loaded either in DOS or in Windows (or both) to enable your computer to 'see' your CD-ROM drive.

Variations on the themes

As if CDDA and CD-ROM were not enough, there are variants which mix sound and data, CD Video interactive, Photo Discs and more. Different programs support more or fewer of those formats; if you have special needs in these areas, you must find the recording package which does the job you need. There are also two major innovations in recording which may lead to new uses of CD-R when they are fully integrated in software. Packet writing allows you to dump a collection of files onto a disc when you wish but to delay building the TOC until you are ready. The other major change is the rewritable CD-R. A packet writer using a read/write (RW) will have software to mount the drive as though it were a removable hard drive or large Zip. The next step is already sampling: DVD-RAM. With that, a rewritable packet system will hold gigabytes of data!

A final warning: At this writing (September, 1997), CD-R is not a science or a solid technology. It is essentially an art form, with the advantage that once you have the formula for what you're trying to do, it tends to work repeatedly. However, the tools of that art are changing at remarkable speed. New hardware and new software emerge monthly and the truths of today will be questionable or false in a few months. However, so far it has worked well enough for me to have seen five titles pressed into 4,000 CD-ROMs - and to have at least five more projects in active development.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Getting Started in CD Recording

This is a personal set of recommendations on starting with your new CD Recorder. The fundamental idea is K.I.S.S. - Keep It Simple, Stupid.

1. Install your system - follow instructions.
No, you don't have to do it this way. You can use your existing SCSI adapter, run other software, terminate in ways that are not recommended. And all of that **may** work. But just for the heck of it, start off by following the instructions. Then, if it doesn't work, you call up the manufacturer and get useful information on fixing it.

2. Prepare your system to burn a CD-R.
If you're running Win95, check the information at the Adaptec WWW site and configure the OS as directed there. Running CD Creator in Win 3.x, increase the buffer settings in CD Creator to maximum values. In NT 4, run Easy CD Creator. If the software that comes with your drive does not run in your preferred OS, don't run it there.
Clear out enough space on your HD to be sure you can burn a CD-R. To be certain, use a drive (IDE or SCSI/AV) with 1,000 MB free. Ideally, use a separate drive from that on which your temporary files are stored. Make certain that all drives that may be used are fully defragmented.

3. Burn a simple data disc first.
Even though you will ultimately be doing much more difficult things, start out with the easy case: All data, Mode 1, files on the hard drive. For example, back up all of your C: drive files that will fit and that are not open while you burn. That means, exclude your CD-writing software folder/directory and the Windows and system directories. It's useful to have that backup disc anyway, but the main reason for this is to be sure that you **can** burn a disc.
While you're at it, make that first disc Disk At Once (DAO). Again, when you can do that, you may want to try Track At Once (TAO) and multiple sessions to see what they do. Try Digital Audio Extraction (DAE) from your CD-ROM reader or from your CD-R; play with copying a CD-ROM, then a CD; make an ISO file on the HD and burn that; and otherwise check the bells and whistles. But do those one at a time, **after** you know that the basic setup works.

4. One at a time, add functions and modify configuration to make the system work as you wish.
You will probably be able to run the CD-R on the same SCSI adapter that manages your other devices, even those (Jaz drive, HD, CD reader) that will be in use with your CD-R. You probably will succeed with various positions on the chain for the reader and with various terminations. But to attempt a complex installation first would mean that the error messages you get will be unique to your system - and none of the manufacturers involved will be willing or able to help you solve the problem. If you make the system work in the manufacturer's configuration, then move one step at a time, you can find out why one of them fails and fix it. If you do three at a time ... forget it!
The same rule holds with other modes, mixed formats, audio files compiled from multiple sources and the other sources of flexibility you will ultimately want for full exploitation of your hardware. It will cost you a few blanks, but will save you endless hours of frustration and save the people who are willing to help you even more.

5. Don't use untried shortcuts.

It would be wonderful if your 12x IDE CD reader fed digital data to your CD-R, but it probably won't. Since you have a SCSI adapter for the CD-R, get a reasonable SCSI reader to do DAE instead of beating your head against an uncooperative manufacturer's wall of stubbornness. Or use your CD-R as a reader and make an ISO. Sure, it takes a little longer, but it works. And if you're going to write at 2x, it doesn't matter whether your reader will run at 4x or 12x - it won't read files any faster than they can be written. (Note, too, that the fact that your reader will work at high speed for other functions does not mean that it can do DAE comparably quickly. An 8x reader may not be able to extract digital audio at all; if it does, it may only do so at 2x, or even at 1x. If you **must** do high-speed DAE, ask around to find a reader that may do it, optimize your system - and hope.)
The same rule applies to your hard drive. It may well be possible to use a single, gigabyte drive for CD-R and the rest of the things you want of your system, but why? Another gigabyte drive will set you back $100 or so, and that's a bargain. So is getting enough RAM.

The fundamental requirement for a successful burn on a CD-R is maintaining a steady flow of data. Anything that interrupts that flow will create a coaster instead of a usable disc. Turn off anything you can on your computer that is not needed to burn the disc: screen savers, AV monitors, fax receivers must not be active if you want to make a disc you can read. Your disc must be defragged to avoid wasting time while the heads seek the next data item. If you're copying a lot of small files, you will want to make an ISO file for the same reason. Again, you **may** get away with violating those rules, but until you know your system and its capabilities, don't take the chance.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Getting Smart(er)

There are many ways to get information about CD Recording - or almost anything else - on the Internet. This note is intended to list some of them with indications of when and how to use them. It provides guidelines, not rules, but you are probably here for guidance, so take them for what they're worth.

Accountability

Accountability means that the information you get is reliable and that the source is responsible for its accuracy. The highest level of accountability is the **statement of the manufacturer** or the publisher of a product. If the company says: "This drive works with that software" and it doesn't, you can recover damages. That's real accountability and the reason that many manufacturers of hardware and publishers of software are circumspect about what they assert and careful to post disclaimers.

The next level of accountability is information posted at a **WWW site** by responsible third parties. Clearly, a FAQ such as Andy's excellent one for CD-R at http://www.cdrfaq.org/ is near the top of the list. My CD-R site at http://www.mrichter.com/ (where this page is posted) is also accountable because I make the effort to keep it accurate and I respond to corrections and suggestions. In other words, the sites are accountable because Andy, his colleagues and I are. You have to judge whether other pages are reliable or not - there's no official imprimatur to say: this is truth and that is opinion. I recommend three WWW sites for general CD-R information: the FAQ for a reference book (I use it all the time); Adaptec's site for the best information on their products and their use; and this primer for information on why things work as they do.

A lower level of accountability, but one still of great value, is the **Adaptec mailing list.** As with the WWW pages, its virtue is in its persistence. In general, neither Adaptec nor those who post to the list vouch for the correctness of what's there, but because the list is persistent over time and because its archives are presented in searchable form, any error is likely to be caught and corrected. Along the same lines, much misinformation is trapped by the listowners before it reaches your inbox.

One step above mere rumor and surmise are the **newsgroups.** There is one primary newsgroup for CD recording: alt.comp.periphs.cdr. Two others, comp.publish.cdrom.hardware and .software, have much less traffic and seem to be havens for newbies and people with nothing useful to do. Nevertheless, there are sometimes good questions and useful answers in the comp.publish groups. However, before using any newsgroup, please take a look at the section below for guidelines.

Finally, there is private **e-mail.** Those of us who publish our real e-mail addresses are inviting questions and follow-ups, so feel free to post. Please try to use a bit of judgement when you do. I have received e-mails (I kid you not) saying: I just bought a bare drive - how do I use it? If your question requires a lengthy answer, I am much more likely to post it to this primer than anywhere else, so look for the answer here first. Scan the Adaptec site for their writeup and for the archives of the list. It doesn't hurt to check the FAQ, either. For some strange reason, I get perturbed when a question with answers readily available is asked of me for the thirty-fifth time in a week. "What's the best medium?" is likely to inspire a very uninspiring response.

Newsgroups

On-topic and cross-posting

Make way for my gripe line.

There is so much overlap among the three major CD-R newsgroups that there's no point in complaining about a software question being asked in the .hardware group. However, as you wander farther off-topic - into DVD recording, hard drive optimization, spread-sheet software and so on, you are ever less likely to get a useful answer and ever more likely to generate resentment among the people who want to help. (The people with nothing better to do are happy to discuss matters about which they know nothing.) To find out what's relevant to a newsgroup, *read it.* Scan through the current posts to see if they focus on your concerns - it is not enough to have one post in forty peripherally related to your issue.

Cross-posting is seldom appropriate for a question and must be done correctly (the easy way). It makes sense to cross-post an announcement, but usually one appropriate newsgroup is the place to post a question. If it's as urgent as smoke coming out of your floppy drive, crossposting makes sense; otherwise, pick one spot and if you don't get a reply in a few days, try another.

Correct crossposting is easy: compose one message and post it to all the newsgroups you feel **must** get it. That way, all replies will thread together and can be correlated with one another. If you write separate posts to each of your victims ... er ... target groups, then each will have its own thread and no one will know how to relate them to one another. In addition, someone who wants to help - particularly to correct an error in your problem statement - would have a difficult and ultimately impossible task. She would have to know all the newsgroups to which you posted and would have to subscribe to each in order to make the correction.

When I see an off-topic post, I either ignore it or say: "The question is off-topic - this group is for CD recording as indicated in its name. If you post to a relevant group, you should include [some specifics missing in the original post.]" When I encounter an improper cross-post, I reply "If you must crosspost, please do it correctly. Reply posted to ..."

I am routinely accused of being a "netcop" for those observations. The most interesting thing about that accusation is that those protecting others' right to misuse the resources never provide any useful input - even of as limited use as mine. They will discourse on my motivation and ancestry, but somehow never address the question at hand. I suppose it makes them feel good, so I don't complain.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Terms of Art

This is a selective list of terms relevant to CD-R with explanations rather than definitions. Organized and authoritative information may be found (as usual) in the FAQ at http://www.cdrfaq.org/ and at Roxio's site. Much of this information is controlled by standards; many of those standards are controlled by the International Standards Organization (ISO). Those pertaining to Compact Disc are in the evolving document, ISO 9660.

NOTE: I look forward to getting information to correct any errors and will consider suggestions or requests for additional entries. I am not interested in arguments about facts, particularly if 'somone' told you ... If you read the INet long enough, someone will tell you whatever you want to hear. You will 'hear' that a CD-ROM can write CD-R, that LaserDiscs are digital, and that Saddam Hussein and Benjamin Netanyahu are lovers. I am not interested in rumor, wishful thinking or anything but traceable fact.

Not Relevant - LaserDisc and CD-V

LaserDiscs and CD-V's are analogue, not digital. Their video and audio data are not read or written on a computer. Digital data on a LD are encoded into analogue to be recorded. Regardless, these media are not readable or writable with CD-ROM or CD-R.

Slightly Relevant - DVD

DVD used to stand for Digital Video Disc or for Digital Versatile Disc. It now stands for DVD. It is a medium with great potential for entertainment and for data storage - and is important in the development of CD-R. However, DVD is in flux right now and what it will become is TBD.

DVD - Movies

The DVD players in your local electronics superstore are the video counterpart of CD players. The format used on DVD movies is MPEG 2. First, that is not the same as MP2 - MPEG 1 Level 2 - which is an audio format. MPEG 2 is implemented in hardware, not software. Creating an MPEG 2 file at this point takes a workstation, hardware and software that would buy a small house in most parts of the world. This is not a job for a PC and probably will not be for some time.

DVD-ROM

A single DVD can hold over 5 GB of data and a few applications see value in exploiting it. While this is not the place to go into technical depth, it is worth pointing out some important technical differences between DVD-ROM and CD-ROM. Maybe it's enough to say that DVD's may hold two or more tracks on each side and may be written on both sides of the disc.

DVD-ROM readers can read DVD movies, DVD-ROMs and pressed CD formats. The first generation cannot read CD-R of any flavor, but later generations are Multiread and will read CD-R and CD-RW. Please note (again) that additional hardware - an MPEG 2 decoder - is required to play a DVD movie on a DVD-ROM.

DVD-R

There is a DVD-R on the market It writes DVD movies (built on a workstation) and DVD-ROMs.. It is commercially available for about $17,000. If you're interested in buying or using one, find an expert; I can't afford even to think about it.

DVD-RAM

There are two competing formats for DVD-RAM and first production units (under $1K) are on the market now. It is logical that at most one will win out eventually, but no one knows which format - or whether it will be another mechanism altogether. As I understand the standard format, it will read DVD movies and DVD-ROMs. A 2.6-GB blank (which may be readable in DVD-ROM) costs about $20 today; a 5.2-GB blank, which is not readable in other hardware, is double-sided and costs about $40.

DVD-RW

Logically, DVD-RW should include DVD-R - but it doesn't. It should be a variation of DVD-RAM - but it isn't. And it should have a single standard - but guess what? Fortunately, the competitive units are not yet available; if you want to learn about it at this stage, please look somewhere else. I have all I can do to keep up with commercial products.

Compact Disc

Now we get into the real world. Some of these things have been around long enough to be pretty well understood and even to be in regular use. The first topic is formats - in particular, those used on pressed discs. Note that discs are pressed from glass masters. The masters are made from tapes. The tapes **may** come from CD-R's or from other sources; the pros are not limited to starting with CD-R any more than cassettes have to come from cassette masters or video tapes from VHS tape masters. Therefore, the pros can break the 'rules' we follow in CD-R.

CD Audio

Properly called CD-DA (Compact Disc - Digital Audio), this is the sort of thing you drop into your Discman. Information about it is defined in the Red Book - a formal standard which is explicit and (largely) followed. The Red Book (named for the color of its cover) calls for a standard format which is not subject to argument; for convenience, I use redbook as a word for that format: uncompressed Pulse-Code Modulated (PCM) data sampled 44,100 times per second in two channels of 16 bits each and ordered and interleaved in a particular manner. With reordering, that format becomes RAW or SND on a computer. Wrapping appropriate header and footer information around a RAW or SND file produces an AU or WAV file. As discussed elsewhere in this primer, software which extracts digital audio produces WAV files; authoring software writes CD-DA from WAV files. It is worth repeating here that a conventional CD player reads exactly one closed session on a CD; anything recorded after the first session is invisible to it and if that first session is not closed or is not audio, it will not be read at all.

CD-ROM

CD-ROM (Compact Disc - Read-Only Memory) is used to describe any CD format in which data (rather than audio tracks) are written. Hybrid modes incorporate both data and audio and are discussed below. A data track is a session and consists of a Table of Contents (TOC) and data. The TOC is translated into a FAT16 file system by the CD Extensions (such as MSCDEX). The files are any flavor of computer file. More information elsewhere in this primer addresses multiple data sessions and related matters.

Hybrid Modes

On a CD-Xtra disc, a CD-DA session is followed by one or more data sessions. Since the first session is audio and closed, it will play on an ordinary CD machine. The subsequent sessions read as they would on any data CD-ROM. To satisfy the standard for CD-Xtra, the audio session and first data session must be written at once and some additional folders and data must be recorded; however, for many purposes it is sufficient simply to record a closed CD-DA session, leave the disc open, then add one or more data sessions which do not import the first. Mixed Mode is similar to CD-Xtra except that the data session(s) is written first and the resulting disc plays only in a CD-ROM or special player. NOTE: The standard requires that the audio of a hybrid mode be written TAO; some software violates the standard to support emulating pressed discs.

CD Recording

Home recording of CD's in various formats has been practical since about 1995. The technology is evolving rapidly, various packages of hardware and of software implement different features more or less in compliance with the standards - and confusion is increasing even faster than brands and models.

CD-R

Compact-Disc Recordable (CD-R) uses a blank consisting of a polycarbonate base with a layer of dye under a metallized surface covered with a lacquer. Clear and blue dyes may be used, resulting in gold or green appearance; the metal layer is gold with those dyes. A different blue dye is overlaid with aluminum, giving a silver/blue disc. When exposed to intense laser light, the dye changes opacity to the less-intense illumination used to read a disc.

CD-RW

Compact Disc - Rewritable uses a different mechanism for recording. The sensitive layer is a silver-colored alloy which changes state (crystalline to amorphous) when exposed to intense laser light and which changes back to some extent when exposed to different illumination. Those state changes cannot be repeated indefinitely; the specified number of cycles for successful operation is currently 1,000. Although an erasable blank is fundamentally different from the kind used to write once, a CD-RW recorder will write CD-R without penalty.

Only some players - CD audio or CD-ROM - can read erasable blanks. CD-ROMs designated Multiread should read them reliably; others may or may not. Some audio players which read write-once CD-R media will also read erasables; some will not.

Packet Writing

Where CD-R and CD-RW were designed to make a disc which behave like a pressed CD or CD-ROM, packet writing is designed to allow writing comparatively small amounts of data in the way that one writes to a floppy drive. Two different systems are used, one for erasable blanks, the other for write-once; they are discussed in this primer under Packet Writing. However, it is important to note the following, which apply to reading in drives other than the writer.

1. CD-DA cannot be written in packets; a packet disc cannot be read in an audio player.
2. A packet-written disc may be finalized to a format readable under Win95 or Win NT, but cannot be read at all in DOS or Win3.x.
3. A write-once packet disc may be readable without finalizing in Windows 98 and an imminent Apple system.
4. An erasable packet disc requires a driver to be read. It cannot be finalized.

E-mail me at cd-r@mrichter.com

Return to Mike's home page

Media - Design and Durability

Let's take a look at how CD media are made and at what that means for their durability. There are darned few hard data available on many of these matters and even fewer reliable numbers, so apply a substantial fudge factor - and give me better numbers if you can find them. Please remember in what follows that the light used to read a CD is infrared and not visible to the human eye. What you see is not what the reader sees, so do not be misled. I am also neglecting here the whole business of the grooves pressed into all discs and containing both positioning information and code for the type of medium.

To begin with, we should examine a pressed disc. Essentially, it begins with a fairly thick layer of tough, clear plastic. The top surface of the plastic has been inscribed with pits and is covered with a very thin layer of metal. On top of that is a thin layer of that plastic, then the graphics which are silk-screened on the very top. Going back to that inscribed top of the bottom piece of plastic, it is usually created by pressing a metal stamper made from a glass master into the surface. Where the result is flat (parallel to the face of the disc), the reflection from the metallized layer is strong; where there is a slant to the plastic, the illumination is scattered and there is no significant return signal. Thus, the maximum brightness is high and the minimum is quite low.

An erasable disc (CD-RW) is similar in some ways to the pressed disc. The top and bottom layers are the same, but the plastic is smooth. The big difference is that instead of a metallized layer which simply reflects the light uniformly, the erasable uses an alloy which changes state when illuminated strongly. In the crystalline state, the alloy reflects light quite well. But it can also be switched to an amorphous state in which its reflectance is substantially lower. The frequency of illumination for reliable reading of an erasable is different from that for a pressed disc or for a CD-R, so special hardware is needed in a reader to allow it to get the signal back. It is also significant that the change of state is not completely reversible. After a number of cycles, the alloy becomes stubborn at a spot and insists on being either crystalline or amorphous, ignoring all the urging that the writing laser may offer. As a result, the disc will gradually develop errors over repeated erasures; eventually, they will be too numerous for error correction to deal with and the disc will be a coaster.

A write-once disc (CD-R) uses yet another arrangement. The layers are similar: thick plastic, smooth surface, metallized reflecting layer, thin plastic over the top. The essential difference is that the thick plastic layer has a dye in it which changes color when illuminated by the writing laser. Before illumination, it may be relatively opaque to the reading frequency; after illumination, it will lighten. The dye is deliberately unstable (otherwise, it would not change transparency) in two different senses. One is that it is able to be written - to respond to the writing laser by changing color - only for about five years; the other is that over time and depending on handling, the dye can fade. When working right, the dye absorbs some of the light when it would ideally be clear and allows some through when we want it to be opaque, so it does not offer the contrast of a pressed disc.

Writing and labelling the disc

The top plastic layer on a pressed disc is pretty sturdy - not as durable as the thicker bottom layer, but pretty good especially when it has a nice coat of silk-screen ink on it. An erasable is usually pretty good in this respect as well; not as resistant to mistreatment as a pressed disc, but a lot better than a typical CD-R. A write-once blank may have a durable top surface added or not. A durable surface, like one advertised for 'long life' or used to support ink-jet printing, is still much more vulnerable to scratches than is the top of a pressed disc, but the least expensive blanks, with no additional protection or surface printing, are the most fragile of all.

In general, if you want to write onto an uncoated disc, you should use a pen made for the purpose. TDK has one available for about $3. Many uncoated discs will take writing from the felt-tip (**not** the metal-sheathed ultra-fine) Sharpie, but there is a slight risk that the solvent in the Sharpie's ink can etch some plastics which may be used. Other pens may well be safe, but why not hedge your bets and either find out from your medium's manufacturer or stick with something made for the purpose? As for pencils and ballpoints: you might as well use a dentist's drill or a sandblaster.

Another option is to apply an adhesive-coated paper label such as are available from Avery, Neato, Stomper and others. That's a fine solution, but there are some risks. Obviously, the label needs to be pretty well centered to avoid problems in high-speed readers. Also obviously, you don't want loose adhesive to foul up your reader or its optics. However, the biggest problem is the adhesive used. It must not let go. If it does, the label will peel away in part, snag in the drive and potentially peel some of the adjacent plastic layer. And that is disaster. If you scratch the thick layer moderately, there will probably be no effect at all. But an uncoated disc has a remarkably thin upper layer and even a tiny scratch or hole will disrupt the metal layer below; a disturbed reflecting layer means no reflections, which means no data - dead disc.

Lifetime

Okay, now you're prepared to handle the disc with due respect for the ultrathin lacquer layer on top. How long will the disc last if you don't sandblast it? If it's a pressed disc, the answer is likely to be 'forever'. That is, there is no decay mechanism known for the CD's sandwich of plastic and metal if it is made well. We may learn better some day, but for now when we find a CD that has gone bad, we suspect that it was made badly, that something led to oxygen reaching the metallizing layer or some extraordinary event - like a fission bomb - turned the plastic opaque.

Write-once media are another story and one which is less understood. Remember, that dye which holds the information is unstable. Time alone will cause the dye to fade or to grow opaque. Either way, it will mask the data. From accelerated tests, manfuacturers will claim life expectancy of 100 or 200 years, but they don't know. On top of that, the real life depends on how you store the disc. Intense light - sunlight or most other forms - will change the dye. After all, the only difference between the sun and the writing laser is that the laser concentrates its light much more finely. How much exposure does how much damage is ... unspecified. Even worse is uncertainty about storage temperature. Life tests are accelerated by holding the disc at a higher temperature than normal; no one seems to be willing to speculate on what happens to that 100-200 year estimate if the disc is not held at 20 C (or whatever).

The erasable disc's decay is probably something like that of the write-once. There have been reports that intense illumination (in one case, with an ultraviolet PROM eraser) will erase an RW and that seems quite likely. My guess is that cosmic rays will also lock the alloy in one state, but I have found no data from manufacturers or others on the matter. But the real killer is that 1,000-cycle estimated life. Well, I certainly don't expect to write and erase the blank a thousand times - come on, be sensible! Unfortunately, you may be doing more erasing than you think. In particular, if you are writing fixed-length packets, the area which holds the directory (think of it as a FAT, though it isn't) is written and rewritten each time you modify a file. It does not take a lot of scrubbing of that part of the disc to reach a thousand cycles to become unreliable. By now, you probably will not be surprised to know that we don't know how to assess that vulnerability, to predict how much use of a disc written in fixed-length packets will kill it.

De Beers claims that 'A diamond is forever.' But diamond is an unstable form of carbon under normal conditions and degrades from the time it is mined. A CD-R or CD-RW treated well may last effectively 'forever', too - as long as we will care about it (though not as long as a diamond). Mishandled, 'forever' for a CD-R may be measured in minutes.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

A Happy Medium

While I do not mean to shock you (prepare yourself, please): the CD-Record process is not perfect.

Now that you have recovered, let's take a look at what is imperfect and what can be done to minimize at least part of the risk. A commercial (pressed) CD or CD-ROM consists of a thin layer of aluminum inside a block of plastic. That aluminum has pits and flat areas; the flats reflect the laser light to provide the signal. That signal is normally very strong because the aluminum reflects almost all of the laser light shining on it, the pits reflect almost none, and almost anything will read the difference quite easily.

In contrast, a CD-R has less contrast. Recording a CD-R means subjecting a layer of dye to enough laser energy to make it change its state. If you leave a color photograph in the sun long enough, it will change color. Its dyes have changed state. The same sort of thing (though much better controlled) is going on in the CD-R. The dye then allows more of the light of a reading laser to pass through the area where it has been written. That light is reflected by the layer of metal behind the dye and passes through again. As a result, most of the reader's light is passed by the area which has been written and most of it is blocked by the area which has not had a state change . Even under the best conditions, the bright spots are less bright and the dark ones less dark than on a pressed disc. Reliable reading then becomes more difficult on a CD-R than it is on a commercial disc.

How well a blank will record in **your** CD-R depends on several factors. How well it plays back depends on others. Essentially, if a medium is 'right' for your recorder, it will change state very effectively when it has been written as you intend. However, if you write at a different speed, it may not work as well - it may no longer be as well tuned to the energy the laser lays down. (Perhaps you are recording more slowly; if the areas that should have changed state did so effectively at a higher speed, they can't change more when hit with extra energy. However, the parts that should have stayed opaque might get enough spilled energy at the lower speed to change a little - perhaps enough to make them a little less black when read out. Change to a less sensitive medium and you may have better results.)

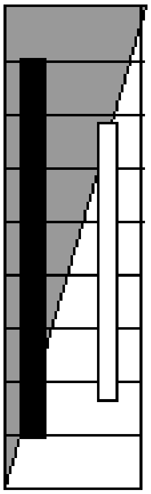


Figure 1

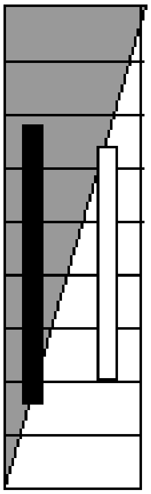


Figure 2

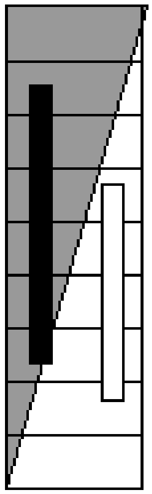


Figure 3

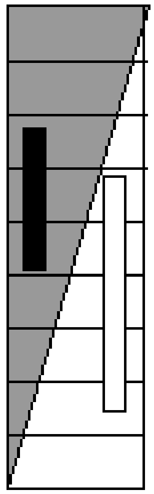


Figure 4

The pictures above are supposed to show the range of laser signal received from the disc when illuminated for reading. Perfect reflection would be the solid white at the bottom of each Figure; perfect absorption would be the solid gray at the top. The signal available in each case is the black bar on the left within each Figure; the detection ability of the reader is the white bar on the right. Figure 1 represents a good, pressed CD and a good reader. The maximum signal is much more than is needed for a the reader to see a one; the minimum signal is much less than that needed to be recognized as a zero. Figure 2 suggests what happens with CD-R: the signals are closer to the thresholds of the reader, but still unambiguous.

In Figure 3, a less sensitive reader (needing more light) is looking at a CD-R which produces less signal at both ends. Notice that here a one is ambiguous; it lies between what the reader would call a zero and a one, so that there is a significant chance that a bit will be misread.If you are reading a megabyte file, there will be about 4 million ones and about 4 million zeros. If the chance that a one is misread is only one in a million, there will be about four errors in reading the file. That means that four times, there will have to be a reread; it's unlikely that the reread will be wrong again, but the reading process has been slowed a great deal. Of course, if the one is even weaker or the reader needs a little more signal, the chance of error goes up until the reread also has a significant chance of failure. Especially if the disc is imperfect - perhaps a spot of dust - it can become completely unreadable.

Figure 4 represents what happens with an erasable (RW) disc. Notice that the signal level for a one becomes much lower; the dye is never as nearly transparent as it is for a CD-R. As a result, a conventional reader cannot make sense of the disc at all. In order to read both RW and conventional discs, the reader must adjust its parameters based on the signals it receives. Such dual-mode readers are becoming available at this time (Fall 1997), but few are on the market. If RW discs become common, the demand for dual-mode readers will be significant and they will become the standard for new drives.

Finding a happy medium

Now that we can see **why** it is the combination of writer, medium and recorder that makes a difference, we can look at how to determine what combinations will make you happy. If you have a writer, a reader and some blanks, you are ready to burn them and try them. If the result leaves you unhappy - reading is slow, audio is noisy or whatever - clearly, any of the three components can be changed. A writer costs about $400; a good reader will set you back about $100; blank media run around $3. Which one would **you** change first?

There are three major kinds of media with substantial variations in two of them. Relatively few companies make blanks, but many sell them; those who do not make their own buy them from one of the manufacturers. The major differences in formulation are indicated by the color of the writing surface: gold, green or blue. There is more variation among the different varieties of gold or of green than between different colors on average, but at least you can be sure that you have a different formulation if you change colors. Because that is what you will want to do: try different media until you get what you want.

Let's assume that you're going to go with one of the major brands - perhaps one that you expect to find available consistently and at low cost. Buy a few blanks for your first runs. If you expect to copy audio CD's on the fly from your reader, do one of them. If you often write data-only CD-Rs from your hard disc, do one of them. Then check out how the discs perform on the different readers you may want to use. For example, check your audio CD-R in your car player if you expect to use it there. The mobile units tend to be more fussy about CD-Rs than good home audio players or computer CD-ROMs. If everything works well in all of your expected applications, lock in on that brand and variety (if you can tell the variety from the package). Buy a supply of blanks and keep a few in reserve so that even if a new batch turns out to be bad, you still have some that you can rely on.

There are many ways to determine how good a copy is. There are thorough tests which are tedious and simple ones which go quickly. One of the simplest takes advantage of the fact that a poor track will often need to be reread. The rate at which errors occur in blocks of data (called BLER) is what you want to determine. When you insert a disc into a reader, the activity light goes on while the Table of Contents (TOC) is being read. If the BLER is high, even reading the TOC will take some time; if the TOC cannot be read reliably, the drive will take a long time to determine the problem and then may spit the disc out. Suppose that you have just made a copy of a disc that you own. When you put the original into the drive on which you plan to read, the drive light turns on for five seconds. Put in your copy; if the light goes on for five seconds, the BLER is very low and the combination is good. If it is on for, say, nine seconds the combination is marginal, but not hopeless - look for something better.If it's on for thirty seconds (even if eventually it goes out), the combination is useless and you need to scrap that medium and look for something else for that job. Remember, too, that because one medium works best for you at one speed for one purpose and in one reader, you can't be sure that it will be the best choice when any of those changes.

Wrapup

If you have gotten to this point, you should know the answers to questions often asked in the newsgroups and elsewhere:
1. What's the best medium (variations: for my drive, for audio, for mixed mode)
2. What's the best writer (or reader) - sometimes specifically for audio or data or mixed format
3. Why won't this reader work on discs that work elsewhere? Is it broken?
4. Why won't my writer read discs that it writes?

Since you can answer all of those and many more, you are an expert. Just as I am. It doesn't take much to become an expert, but let's not spread the word. :-)

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Media

The medium - the blank onto which the recorder writes - is one key to successful CD-R. Unfortunately, there is not the consistency one would wish among the various types. Several different forms are available at this writing (March 1999), distinguishable by the color of the active surface - the one that goes down and onto which the laser writes. They are gold, green and blue. None of them is 'best', and the gold and green vary greatly among brands. Even a single brand may have different product, depending on the actual manufacturer. There are also two different colors of metallizing layer: silver and gold. (The silver may either be silver or platinum.) The active color which seems to be green is actually blue, but made to look green by reflection from the gold metallizing.

It is the combination of medium and recorder that determines how well your discs come out. If someone finds that Brand X makes great audio discs, you may find them terrible because your recorder does not use them as well. Thus, you may want either to stay with the hardware manufacturer's recommendation or with the results reported on the newsgroups or the mail list. Note, too, that an audio disc that plays well in some CD machines may not be acceptable in others and that what works best for data may not handle audio as well.

Most media on the general market are of essentially the same length. There are slight variations, but whether the package is marked 650 MB, 680 MB or something else, it is almost certainly about the same size as any other with similar markings, as attested by the claim of 74 minutes. The longer discs (80 minutes), shorter ones (63 minutes) and shortest ones (smaller diameter, about 15 minutes) have been hard to find and often disproportionately expensive; the 80-minute discs are also reported to be harder to write successfully. Otherwise, the apparent difference in capacity is largely a marketing difference in counting megabytes and is of no practical interest. There is slight variation in actual length - a matter of a few seconds - but that is seldom important and may not be consistent even within a brand. There is a page on this in the Files section of this site with some measured lengths and a link to another such table.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Longevity

One of the key questions about CD recording is: how long will it last - where 'it' is a write-once or erasable blank either before or after recording. There is a little information about write-once blanks which I am trying to summarize below and a lot less about erasables, so what you find here has a lot of guesswork.

A write-once blank uses a dye which is not entirely stable over time either before or after it is written. Using accelerated life tests, manufacturers estimate that a type of disc will remain writable for between five and ten years after manufacture. After that, the probability of write errors will increase - I suspect because the power requirement will become non-uniform and regions will not record properly. After being written, the areas which have been written should hold well, but the unwritten areas will effectively by written (perhaps by cosmic rays?) and the disc will develop excessive errors in a century or two.

Two comments are critical. One is that there are assumptions about handling of the discs which can be significant. If you expose any blank to intense sunlight for a period of time - whether written or not - you will find that it changes visibly and functionally. That does not mean that you need to keep the discs in the dark, but that exposure to heat or light will accelerate the breakdown. With normal handling, a written disc will probably outlast you and certainly your interest. The other note is that accelerated life tests are performed by overloading the medium with the forces which are **believed** to cause the decay. I have found no information on the accuracy of those tests - how well the acceleration relates to the real world. In short, I don't trust the numbers that come out as meaning anything about the life that would be measured without acceleration. I would not choose one brand over another because it claims longer life. Needless to say, none of the media we use today has had a century or even a decade of storage to validate the tests that are reported.

Erasables work quite differently and are subject to different forms of decay. I have been able to find neither claims for life expectancy nor accelerated life tests on them before or after writing. It is clear that the story is more complex than for write-once if only because we know that erasables lose reliability when put through erase cycles. My guess is that the distinction in bit-sized regions between crystalline and amorphous becomes lost, that the regions become more or less locked in one state or the other when written repeatedly. If that's true, then the same effect will presumably limit the life when the disc is exposed to heat, light or cosmic rays. Does that mean that erasables will last longer than write-once or not? Yes, it does. It means that their expected life is longer or shorter - but I cannot determine which.

A final comment seems appropriate. The failure modes known and hypothesized here do not lead to the disc suddenly 'breaking' in any sense. Reading a CD entails correcting errors. As the disc degrades, the rate of errors increases. An unwritten blank may become gradually less consistent in reacting to the write process so that regions are not written uniformly. Regardless, you are not likely to find that a spindle becomes suddenly useless the day after its tenth birthday or that an erasable which has worked beautifully for 999 erase cycles turns transparent on the 1000th. Instead, there will be a gradual increase in the risk of a faulty data read or a gradual degradation in high-frequency quality or other degradation in audio.

E-mail me at cd-r@mrichter.com

Return to Mike's home page

Rebadging - Who Made This?

There are only a few manufacturers of CD-ROM and CD-R(W) drives - far fewer than there are brands on the market. Many of even the biggest names, such as Hewlett-Packard, do not make drives at all. Rather, they contract with manufacturers to deliver drives branded with their names. We speak of the process as "rebadging". In the process, they may have specific features built in for their purposes or they may use off-the-shelf product with a new faceplate.

The information needed to write a disc depends on the features you want to use. A program with limited application, such as one which writes only audio or which only copies a CD or CD-ROM, is not sensitive to many of those features. One which needs to create CD-Text **and** to write data from a HD in DAO **and** to create a VCD **and** to do all the other things a full-function program sets out to accomplish must know the specific properties of the hardware it is to use. Therefore it needs the specific drive string and that means that a rebadged Plextor is not a Plextor, but an HP, IOmega or whatever.

Further complicating the matter is that the rebadgers may have different drives under the same "hood" at the same or different times. One may be bought from Plextor, another from Ricoh and so on. They are different drives to the mastering program, but the user has no way to know that. Neither, in fact, does the software publisher until the drive or at least the essential information comes to them. If the manufacturer cooperates, the information may be there before the drive hits the market; if not, they have to find a way to get that particular drive (in the sense of "under the hood") from commercial sources. There is no way that a programmer can support a drive he does not know. Some programs with less capability can get away with less - whether the mechanism came from Plextor or Ricoh may make no difference for the limited application. So when a publisher says that Drive X is supported by Version N, they mean that the drive version(s) they know are supported. They do not mean that the one you buy this week is supported unless its the same mechanism that bore that name last week.

Incidentally, you will see evidence of this problem even with limited-function software. In the newsgroups, you will find arguments over whether the FlyByNight model 12 supports a feature. One person says it doesn't; another says it does. Of course, they're likely to be right. Both of them. They have different drives with no external indication of the difference.

Before you ask: the same holds true with blank media. The package of RitesAll 16x media you buy at your neighborhood feed store today may have the same discs inside as the next one on the shelf or the package you buy tomorrow - or it may not. The only external indication you have **may** be the country of origin, usually on a sticker attached to the outside. That alone is a good indicator: this discs are from Taiwan, those are from Malaysia - they're not likely to be the same formulation. You cannot judge by the appearance of the blank except in a negative sense. That is, if they look different in color or surface finish, they will almost certainly perform differently, but if they look the same, they may or may not be equivalent.

Now, a CD-R(W) reports itself to the operating system as a "drive string" - a set of characters which identifies it so that its features can be known and exploited. Two drives which look the same to the user but are different inside will report different drive strings. So the unsupported unit which shares a model number with a supported one differs where the computer sees it - in the drive string. The computer cannot see the writing on the box or even on the drive itself; it knows the drive by its string and supports it or not based on what's in that string.

Similarly, the recorder recognizes the blank not by what the packager wrote onto the disc or its package but by what is written into the disc itself in the ATIP. For further discussion of that, please see the page A Blank is not Blank. In this case, the ATIP limits what can be done with the blank, but two different formulations may have identical information in their ATIPs.

E-mail me at cd-r@mrichter.com

Return to Mike's home page

A Blank is not Blank

No - this is not a trick. The fact is that although a blank has nothing written on the medium, it does contain information - information which can be very important to its use.

A brief recap on the construction of a writable disc. From bottom to top, the three layers are a polycarbonate plastic; a metal; and a lacquer (plastic) top coat which may be overcoated or printed. A write-once blank has a dye in the polycarbonate; the dye changes its infrared transmissivity when written with a high-power laser. An erasable uses an alloy for the metal layer; that alloy changes state between crystalline and amorphous when written, thereby changing its reflectivity.

That polycarbonate slice of the sandwich has a spiral groove pressed into the plastic with a stamper. The laser beam is servoed to stay in the groove as it writes. The tighter the spiral, the greater the total length along the spiral and the longer the disc can record and play. Unfortunately, a tighter groove is harder to follow accurately, so both writing and reading an 80-minute blank are more demanding than the same operations on a 74.

The spiral is not plain; it is modulated with a wobble to give the laser tick marks to locate its place along the spiral. That regular modulation or jitter is further modulated with digital information in what's called the ATIP. Specifically, information in the ATIP includes

- Manufacturer
- Writable/Rewritable
- Dye type
- Spiral length in blocks
- Rated speed
- Audio

Since the length of the spiral is pressed by the same stamper which encodes the ATIP, that information **must** be correct. Everything else is true or not depending on whether the stamper is used by the manufacturer who had it made to press the intended medium. Ideally, the stamper is tuned to exactly the material of the blank - but there's no guarantee of that or of the manufacturer of the blank being the one who had the stamper made.

Only a writer can read the ATIP because only a writer cares about what's in it. If the disc is so badly corrupted that it cannot stabilize in the drive, the ATIP cannot be read, so the writer doesn't even know whether the disc is erasable. Obviously, competent software won't attempt to erase a write-once disc, so that's one way to kill a rewritable medium. Dye type is of little practical interest; whatever is encoded there is overruled by the result of power calibration when the burn begins.

Rated speed is important when specified. If there is no specification, an erasable is always written at 2x; otherwise it may be rated for 4x, so erasable discs not coded to permit 4x writing will not allow it. Similarly, only discs encoded for "audio" can be written in the standalone writers (otherwise, they are identical with that manufacturer's conventional blanks). It is not clear what happens when a standard, write-once disc without a speed specification is to be written; at least in general, it appears that whatever you try will be attempted, but that does not mean that it will work.

Finally, we come back to groove length. The minimum inner and maximum outer radii for a CD are specified in the standard. The manufacturer's rated number of blocks in the ATIP is the amount that can be written in that space, allowing for the runout (leadout) track. In practice, the spiral goes beyond the maximum radius, so there is more room than the manufacturer allows - but it is room at the expense of the design maximum of travel on the writer and the reader.

Writing beyond the rated number of blocks is called **overburning.** It is a somewhat risky operation for reading and for writing. If the disc is fully written with overburning, then the runout track goes beyond the maximum radius. Then the reader, which needs the runout track for operation, may not be able to read it and may be unable to sync on the disc. That's one way that a disc can work in one reader but fail in another. At least theoretically, it's possible for overburning to damage a writer by forcing it to travel farther than its design permits. In short, overburning is risky; it can pay off in some cases, but if you use it, you're on your own.

E-mail me at cd-r@mrichter.com

Return to Mike's home page

Erasable Media

Erasable (RW) blanks are fundamentally different from write-once media. They may look similar, but that is misleading.

A write-once blank (CD-R) uses a dye which changes composition when exposed to strong light of the right frequency. That change is not reversible; the new form is stable and absorbs the reading laser differently from the original. An erasable disc (CD-RW) uses an alloy instead of a dye. When exposed to light of the right properties, it changes from crystalline to amorphous; another form of exposure switches it back to crystalline. The **reflectance** of the alloy depends on its form. Essentially, as a crystal it reflects the read laser as though it were a dull mirror; in amorphous form, it scatters the illumination. Instead of getting white and black (like a pressed disc) or near-white and near-black (like a write-once), an erasable provides light and dark gray and needs more sensitivity in reading.

As a result, only drives marked MultiRead are able to read your erasable reliably. Very few audio CD players will read RW discs. Otherwise, you can use mastering software on an erasable exactly as you would on write-once and expect the same results. That makes erasable media valuable for a couple of reasons. If you're trying something you have not done before, your failures will not fill the trash with non-recyclable plastic. More to the point, the variation in erasable blanks is much less than that in write-once. Therefore, if you cannot tell whether a failure is due to your system or your medium, try an erasable.

UDF is implemented quite differently on erasable and write-once media. For details on that, please see the pages on packet writing. In summary, a CD-RW written in fixed-length packets cannot be finalized to ISO 9660 Level 3, but it can be read in a MultiRead drive with an appropriate driver installed.

Erasing

When an area of the disc is erased, it is converted to be all crystalline - or at least, as nearly so as possible. In practice, there is always some amorphous material (as there is some crystalline material left when written to be amorphous). With each write/erase cycle, a bit or a region of bits becomes more nearly balanced between amorphous and crystalline. As a result, after a number of cycles, the two states become less distinct to the reader and the ones and zeroes approach 'halves'. Nominally, a disc will survive 1000 such cycles, but individual regions may fail sooner. During formatting for fixed-length packets, the software can determine that a region is failing and mark it to be bypassed when writing. However, there are some critical regions - such as where the TOC begins - which cannot be bypassed. Without packet writing, bypassing is not possible.

When any disc is to be recognized by a drive, either by being inserted with AIN on or by being selected by software, the drive attempts to find out what sort of beasty it's dealing with. To do that, it must find the start of the TOC and read data from it. If it gets nothing at all, then a writer assumes that it has a blank; a reader keeps trying indefinitely. Let's assume that we have inserted an erasable with a bad TOC into the drive so that we can erase it. If the TOC is bad enough, even a writer will not be able to make sense of it and will keep reading indefinitely. It gets enough to know that the disc is not blank, but not enough to figure out what it has. That disc defies erasing because erasure cannot start until the TOC is read.

If the TOC can be read well enough to let the drive stop, but not so well that it can be deciphered, you have an 'Unrecognizable Format'. It needs to be recognized so that a quick erase is possible - one which clears the TOC to allow writing without clearing the whole disc. You may not intend to use quick erase, but the software is not psychic, so it will not let you start doing anything until it knows how to do what you might intend. It **could** accept the condition and simply gray out the quick option, but I have found no program which does that. Fortunately, that problem disc may be erasable with a program such as Super Blank (linked from this site) which offers no option but always does a full erase.

Is all hope lost for the disc which will not stabilize? Perhaps not. There are three options, none of which is certain but each of which has been reported to succeed occasionally.

- **Fool the drive** Start with a good disc, prepare to do a full erase, but just before you hit the last OK, swap it for the faulty one.
- **Give it a sunburn** Expose the disc to strong sunlight for some hours.
- **Treat it like an EPROM** Expose the disc to a PROM burner.

When (if?) quantitative data are accumulated, I'll be happy to post them. When erasable blanks cost $5-10 each, recovering them was worth the effort and I had some success with the first two methods. Now that $2 will buy an erasable blank, I'm more likely to discard a flaky one.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Erasing

Let's get the obvious out of the way first: you cannot erase a write-once (non-erasable) disc. That does not mean you need special hardware or non-commercial software or a secret trick. It means that the changes in the dyes of a write-once blank are not reversible.

Restricting ourselves to erasable (CD-RW) media, we can look at how erasing is done. In first approximation, it works just as we write to those discs. A region corresponding to a bit is heated with a laser until it (more or less) melts, then refreezes in either mostly crystalline or mostly amorphous state depending on the heat/cool cycle used. The result is more or less reflective, hence a one or a zero.

In quick erasure, the TOC is reset to suggest that the disc is blank, but in fact the rest of the disc still has its information. This is not simply resetting one byte of the filename as on a hard drive, so recovery tools will not work to restore the information. Neither is it such complete erasure that a backup program, such as TakeTwo, will be able to use the disc. In short, Quick erase is great for routine use, but if it does not work, a Full erase is needed.

In full erasure, the whole surface is cycled and the information is really, really (honest-to-gosh) gone. A disc which was used for packets may not switch happily to mastering if you use Quick Erase - but Full Erase will probably do the trick. (I would be happy to tell you when one works and when you need the other, but I haven't figured it out yet. It seems to depend on the medium, it may depend on the recorder, but it definitely depends on more information than I've been able to collect so far. Watch this space; some day it may have the answers.)

Finally, there are two more levels of erasure - one that is likely to work and one that should not, but sometimes does. Super Blank is linked from this site. It is made to erase a batch of blanks at once on multiple SCSI drives. However, it will work on only one drive at a time if you wish and it does support most ATAPI (EIDE) drives - so try it. If the disc is sufficiently fouled up, it may report serious errors and start flashing at you. If you ignore that and let it proceed for forty minutes or so, it usually does erase the disc. Super Blank does only Full erase and theoretically does it no better than any other full-erase program. In practice, it seems to work where others do not.

Another solution that should not work - but may - is solar assist. When you put a disc into a drive, job number one is to determine what the disc is. To do that, the TOC is read and the runout track is checked. If the drive is a reader, it does not need a runout track, but in any drive, there must either be a TOC or no TOC at all. If there's a part of a TOC or a faulty one, the drive will keep looking... Forever. If a disc won't stop spinning in a reader, try it in your writer; if there's no runout track (blank, open session, etc.), it will settle down there. But if the TOC is fouled, it will not stabilize even in the writer and violent means are called for.

On occasion, a 'dead' erasable - one which would not stabilize - can be made to look blank enough to be erased by exposing it to strong sunlight for a few hours. Whether it's heat, ultraviolet or black magic, it sometimes works. No sun? Try an EPROM programmer or other intense UV source. No guarantee, nothing but hope that something might save that scrap of metallized plastic.

A final note on Adaptec; the story may be similar with products of other publishers. There are two ways to erase in Easy CD Creator 4. A separate CD Eraser is included with Direct CD; it only offers Quick erase. For the option of Full or Quick, you need to use the Erase function on the CD menu of the ECDC core program. Since Full erasure is needed primarily for DCD functions (including the DCD-related TakeTwo), this is characteristic forgetfulness. You have Full capability where you don't need it, but not where you do.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Disc Speed and Rating

There are three distinct parts to this note: some words on terminology, information on write-once media and critical data on erasable blanks. Terminology first: What's with this business of "erasable" and "write-once"; aren't "CD-RW" and "CD-R" good enough. Frankly, no, they're not.

The terms "CD-R" and "CD-RW" are applied to both the drives and the media that they write. In itself, that ambiguity is confusing, but even worse is the fact that it leads to a set of misunderstandings, particularly by newbies. I believe it's the reason that some do not recognize that you can master an erasable. There have even been some who don't know you can burn a write-once disc in a CD-RW drive. So, live with it: I use "erasable" and "write-once" for media, "CD-R" and "CD-RW" only for writers.

Write-Once Media

As noted elsewhere in this primer, there is a place in the ATIP of every blank for specifying the maximum and minimum write speeds. However, that is a capability, not a requirement. No manufacturer of media is required to fill in those data and many do not. If the blank does not establish a minimum, you are free to write - or to try to write - at 1x. If it does not specify a maximum, then you can run your drive at its max speed. The problem is that that may not produce a valid burn.

There is an optimum speed for writing any blank in any burner. Fortunately, that will tend to be a broad optimum and not to be very sensitive to the burner used. So if the optimum speed in your Yamaha is 8x then that medium will probably burn well at 4x and 12x and I'm likely to get good results in my Plextor at those speeds as well. However, that disc may not write well at 2x, might not work at all at 1x, may have high error rates at 16x and be prohibitively bad at a higher speed.

The result is that there is no blanket answer to the question of whether a blank certified for 12x will work at 16x (or 20x or ...). If it is not limited in the ATIP, you need to find out its performance for yourself or trust the reports in the newsgroups. Remembering that there are errors on every disc (though most are correctable), you can see that someone else saying it's (not) good enough for his needs doesn't promise you anything about your criteria.

Erasable Media

When erasable discs came along, they could be written at only one speed: 2x. So the rule for writing the ATIP was different from that for write-once. If there is no value specified for minimum, the lowest write speed is 2x; if none for maximum, the highest speed is 2x. So far so good, but when the manufacturers wanted to get practical speeds on erasables, they came up with a new alloy. The dynamics of melting and freezing this metal are quite different from those for the 2x or 1x-4x media and the new stuff simply won't work below 4x. These blanks are termed "HS" for High Speed and drives suitable for writing them are similarly marked "HS". At this writing, HS media and drives are typically rated 4x-10x. Non-HS media and drives operate at 1x-4x.

Here comes the fun: the two types overlap at 4x. So someone with a non-HS writer may believe he can use HS media at 4x. Nice thought, but it's wrong. They simply won't work and if he's lucky, his software will tell him that when he tries. If he's not lucky, he may write or format without getting anything written or formatted. Even that's not as bad as using a non-HS disc in an HS drive. Typically, that does work, but only for a little while. All erasable media seem to be forgetful and to lose their data after some period of storage. The old 2x-only blanks seem to hold the information for many years; 1x-4x are less reliable; HS are pretty sad in this respect, but still should work for testing over a few months. But if you write an HS disc in a non-HS drive, you have a good chance that it will have forgotten most of that information the next day, perhaps on the next insertion.

The conclusion is that most experienced users of CD-RWs are very selective about where they use erasable media. Even more, they make sure that they use a tested brand of the right type: HS or not. Erasing is a good thing; forgetting is not.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

How Many Bits??!!

In a discussion of error protection, Guy G. Sotomayor, Jr ggs@shiresoft.com explained how playing back a CD-ROM at 1x requires reading the disc at more than 4 megabits per second. If you are concerned about matters such as making copies accurate to the bit level or if you want to understand some of the reasons behind the imperfections of CD-R, this section is worth the time it takes. Before going into his text, let me note that a 'symbol' is a character or a value - basically, just an 8-bit byte. Each sample of a redbook waveform consists of sixteen bits for each of two channels: 32 bits, 4 bytes or 4 symbols.

> The first thing to remember is that the data on the CD is stored as 14 bits/symbol rather than just 8. There are several reasons for this, but it makes reading the bits easier. Here are the criteria for selecting which patterns can actually be used:
> 1. 11 is not allowed
> 2. 1001 is as close as two 1s are allowed to get
> 3. 100000000001 is as distant as two 1s are allowed to get (ten 0s).
>
> Each 14-bit symbol is separated from its neighbor by 3 bits, called merging bits, coupling bits, connecting bits or packing bits. They allow the 3 rules above to be applied continuously. Remember at this point we just have a string of bits -- there is still nothing indicating where bytes start and stop. To recap, 8 bits is represented by 17 bits - a little over 2x.
>
> Now, we look at how data is actually stored on the CD. Everything depends upon a 588-bit frame. The 588 bits are organized as follows:

Description	Bits/each	Total bits
Sync Word	24 bits + 3 padding bits=27	27
Control Word	17	17
Audio Samples	6 samples, 2 symbols ea	17 x 12=204
Error Correction Q	17	17 x 4=68
Audio Samples	6 samples, 2 symbols ea	17 x 12=204
Error Correction P	17	17 x 4=68

> Total 588.
>
> To figure out how many 588-bit frames per second we're dealing with we go back to what we see on the output of the CD (namely 176400 bytes per second). A stereo "frame" is 24 bytes, so if we take 176400 / 24 we get 7350 frames per second. Now take 7350 x 588 and you get 4321800 bits per second. Q.E.D.
>
> BTW, the P, Q, G, etc channels are derived from "stacking" the control words (98 of them to be exact) to get the encoded data. As you will no doubt notice, audio data has ECC applied to it. This is the CIRC that I spoke of previously. This is **always** there. Data because of its more exacting nature has an additional layer of ECC applied to it to further reduce the chance of an error. Hope this clears it up!

Footnote from Mike:
There are additional issues including the nature of that 'additional layer of ECC' and the numeric values for error rates in audio and in data. Since ECC discussed by Guy is always imposed, I usually do not consider it in the discussions in this primer. The only specifications I have found for accuracy on CD-R suggest a bit error rate of one in a million for audio and one in a billion for data. Both appear to be worse than we experience in practice, particularly the one for data; that extra layer of ECC should cut the rate by a lot more than a factor of a thousand. Error rates for pressed discs are **much** lower than those for CD-R.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Writing Speed

One of the more common questions in CD-R is: What's the best writing speed? The answer is, again, the one that works best for you. This page is intended to provide insight into why that is so.

First, having the laser work 'faster' or 'harder' is not the issue. It is designed to work in a specific fashion and its capabilities for speed and power are within its designed use. The issue on the laser is akin to that of a CPU: if the computer is on, it is spending its cycles doing either useful work or nothing. The computer will not last longer by having it spend more time doing nothing. The laser will not work better by being used more gently - assuming that the drive is not overheating, etc. In what follows, I write of dye, but the same sort of thing is true for the alloy in an erasable (with extra complications due to heat transfer).

Think of the old Memorex "Is it live" commercial in which the glass was shattered by a sound. That 'demonstration' had the note matching the glass' resonant frequency. (At least in principle; it was an ad, after all, not reality.) If the note had been a little higher or a little lower, it would have had to be much louder to shatter the glass, but it could still have been done. The dye in a CD-R is akin to the glass. It is designed to work best with the write laser energy supplied in a particular way. If it is provided faster or slower, it will still do the job, but less efficiently. That lack of efficiency will show up as a dark area being less dark (because the energy was not used efficiently) or a light area being darker (because of scatter from an adjacent dark area). To a significant extent, that is corrected by power calibration, but a given type of disc will still write better at one speed than at any other in a given recorder. If the optimal speed is 4x, it will probably still be very good at 2x and 6x, but may not work as well at 1x and 8x.

An erasable disc works by heating a tiny area of an alloy until it melts, then allowing it to harden either as a crystalline or an amorphous material. Since the alloy conducts heat, the cycle of heating and cooling depends on how much energy is dumped into it how quickly and how the heat is conducted away. In order for the melting to be localized and for the cooling to be controlled, early erasable media could be written **only** at 2x. Recently, media have been developed which can be written at from 1x-4x and the range may be increased with time. Even so, there is a best writing speed for that heat/cool cycle in any given writer.

The limitations of write-once media are different so that media which work over speed ranges of 8-to-1 or even 12-to-1 are available. Even so, there is a best speed and the farther one operates away from that speed, the higher the error rate. At first thought, it would appear that the dye would react based on the total energy, not the speed with which it is applied, but that's not the way the process operates - at least, not completely. If you are familiar with Silly Putty, you know that you can apply a lot of force to it slowly and it will simply deform - ooze out of the way. Apply the same force quickly, perhaps by dropping it onto a hard surface, and it will bounce. But hit it with a hammer and it will shatter. The resilience of the chemical bonds is similar in some ways to that of the Silly Putty; the changes depend both on how much energy is applied and how quickly.

Unfortunately, home recorders do not let us get a handle on the BLER (bit-level error rate), so we cannot read the uncorrected errors. Therefore the only way we know how well the writer, speed and medium combine is by looking for cases when the errors are so many that they get through the ECC. Since we only see those errors when we read the disc, uncertainties due to the reader enter as well. Please see the pages on media in this primer for some illustrations (of a sort). In summary, some readers are better tuned to a darker recording and some to a lighter one, so different readers will have different error rates from a single disc.

This is *not* an exact science - it's closer to psychology than to rocketry, at least at this stage of its evolution.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Counting on Errors

A fundamental difference between storage on CD-R and on a hard drive is that the CD-R or CD-ROM has a much higher error rate than we are accustomed to. Without substantial error correction, CD-R would either be much less attractive - slower and offering less storage - or not usable at all. There is a low-level of ECC (Error Correction Code) built into CD recording which is below the horizon of any test equipment and which we will ignore for that reason.

BLER - BLock Error Rate - is specified for CD recording. Essentially, it is the probability that a bit which is to be written to the disc is read back incorrectly. If you send a master in to be pressed, the BLER will be determined; if it is too high, you will be asked for a replacement master to ensure that the master made from your disc is accurate. About 13% of the space on a data disc is consumed by block-level error correction bits. Because both video and audio playback can tolerate an occasional glitch which would be unacceptable in a program, block-level is skipped for CD-DA and VCD. Those formats then get about a 13% boost in storage space.

It is possible to read and write raw data to a CD-R - data in which the block-level bits are seen separately. In normal operation, the hardware processes those bits with the 2K data block to deliver only corrected information, but some readers will deal with raw bytes on command. That capability has been exploited for some time in copy protection, where a block is deliberately made bad by altering the block-level correction bits. Some consumer recorders and readers allow you to do the same thing. That can be exploited to circumvent that form of copy protection or to create your own.

In normal copying operations, each block is corrected in the reader and fresh correction bits are generated in the writer. When you create a disc image, the bits are generated in your software; that's why an image of 650 MB of data occupies 750 MB. (It is also why you cannot simply copy an image to a disc and have it run; in that case, another set of ECC bits would be created.) In raw copying, the disc is read as though Form 2 and error handling is left to the software. Any reader which can do DAE can supply raw data, but not all support the software commands that make it practical for data. Clearly, any writer will allow raw write since it will write CD-DA or an image, neither of which has block-level ECC added in the writer.

The question then arises: When should I use raw copy and when 'cooked' (error-corrected)? For normal operations, cooked is the answer. When you copy raw, any errors are propagated to the next generation - and any new ones are added. As a result, the chance of an undesired, uncorrectable error goes up. However, if you are trying to copy a disc protected through deliberate errors, raw mode is the only practical choice. (An expert - such as the one who created the protection in the first place - can hex-edit an image. That is a game the novice will lose.)

There are programs such as CD-R Diagnosis, Exact Audio Copy and CD Speed (all linked here) which will report error rates of CD-DA discs. In order to choose the best medium for your purposes, a starting point is to write CD-DA on each line of blanks at each speed and to see which gives the best results. That doesn't make much sense if you bought a special deal on ten CD-Rs, but it may be worthwhile when planning to buy a spindle of fifty or more. The test is simple: start with a clean audio CD, make copies at each speed you want to try, then run one of the programs on each test disc in each reader for your computer. A few hours of testing can save a lot of grief downstream.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Reading Errors

We like to think of the computer reading and writing information perfectly. In fact, we seldom even think about that at all because perfect operation is so common. Unfortunately, perfection is not achievable by man. In Islam, that reality is recognized by requiring that a flaw be woven into each 'Persian' rug. In computers, real hardware provides all the errors we need.

According to the specification, CD-R errors start out as being not more than one in a million. Sounds good, right? 99.9999% accuracy should satisfy even the most critical person - but it won't do for computers. There may be 8*650*1024*1024 bits on a CD-ROM. If so, 5,000 of them may be faulty and reading that disc will be a **major** problem. So almost 15% of the space on the disc is 'wasted' in coding which helps automatic error correction - ECC or Error Correction Code. With ECC, the specified error level is cut by a factor of 1,000 so that there would be only five bits wrong on a disc. Even those may be transient so that when an uncorrectable error is found the sector is reread in the hope that good information can be retrieved. In addition, most discs will exceed the specification significantly. Even so, if rereading is necessary the data flow is reduced substantially. The read mechanism must stop its rapid progress, back up and start the sector over.

That saving ECC layer is not applied to audio. If it were, the longest playing time on a CD-DA would be about 64 minutes instead of 74. Similarly, on a VCD there's no ECC on the video files. The idea is that the extra playing time costs only slight errors on playback and that you won't notice the ones that sneak through any error correction applied on the analogue side. For video, the fault is usually a momentary spot in the picture and no correction is used. For audio, the player has some limited ECC of its own, but if it is used there can be a loss of brilliance in the sound.

Another key factor in the quality of information read from a CD-ROM or CD-R is the quality of reading. Some errors derive from the way the track is formed on the disc and are related to the term 'jitter'. Even more significant is that, in general. the faster the data fly past the pickup, the greater the chance that a bit will be misinterpreted. With a pressed disc and ECC, that is not significant. Listening to a CD-DA or watching a VCD occurs at 1x regardless of the drive, so again speed errors don't matter. However, when you extract digital data through DAE or in raw form from a VCD, the story becomes less happy. All other terms being equal, the faster you run DAE, the greater the number of ticks and pops - bit errors - you will encounter.

There is also a quality issue here - and in this, I am providing personal opinion, not objective data. With the increasing pressure from the marketplace for faster and faster drives, other things are being sacrificed. The general public neither knows nor cares about DAE, so the ability to get a good signal without ECC is not a selling point on CD-ROM readers. Therefore the manufacturer will work to find a way to claim 24x for a poorly made drive which might work well if designed for, say, 12x and provide competent DAE at 1x or 2x. In practice, the 24x will be slowed by the need to reread even data and may deliver very little more in effective speed than a 12x, while being unable to provide satisfactory DAE even at 1x. Even a pressed CD-DA may sound dull on that drive, though typical computer speakers are unlikely to show that since they, too, seem aimed at bass boom instead of clean high-end response. And as has been reported elsewhere, running a disc at high speed increases vibration and mechanical noise.

Finally, a CD-R is not a pressed disc and it does not perform as well. In particular, a pressed disc has great contrast between its strongest reflection and its maximum scattering of the reader's light. A CD-R or CD-RW has a much lower maximum signal and a much higher minimum signal for the reading diode. As a result, the chances of reading errors from a CD-R or CD-RW are higher than for a pressed disc and a copy of a CD-DA copy (especially audio) is likely to sound noticeably worse than the original. Of course, the better the hardware, firmware and software are for the job, the better the results are likely to be. But there is no free lunch, there is no guarantee of perfect writing or reading, and if you do not take that into account as you choose your reader and test your CD-Rs, you may regret it.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Isolating Errors

When a burn goes bad, especially when it fails repeatedly, the question arises: is it hardware, firmware, software, operating system, configuration or medium? Or is it something else?

That question cannot be answered simply and often requires extended analysis. Unfortunately, that analysis is very difficult by remote control - if your hands are not on the system, the job reduces to judging a beauty contest blindfolded. (Come to think of it ... no, you know what I mean.)

Using an erasable

Assuming that your problem is with mastering and your drive will write erasable discs, get a few of a good brand. Burn something to each disc to be sure that it works well in your system, then do a full erase and file the discs away.

When you have a problem burn, pull out one of the erasables and repeat the burn there. (You'll erase it later, so nothing will be lost except some time.) If the problem does not appear on those discs which you know are good, it's almost certainly the medium or the speed. If the problem occurs with a write-once at the speed you used for the erasable - it's your medium.

Yes, that test is too easy, but it works. Why not use another write-once blank and forget the erasable? Because it wastes a disc (no big deal at current prices) and because the other line may have other problems. This way, you know the disc is good.

Using CD-R Diagnostic

When you analyze a disc with CD-R Diagnostic at its most detailed level (Complete Scan), it comes as close to measuring BLER (Block Level Error Rate) as your hardware allows. In many cases, it actually gets down to that point. It reports the total errors it found and those which are not corrected. That is a very powerful set of data. The total error count is a measure of the quality of the burn. It takes a pretty poor burn or a physically damaged disc to get an uncorrected data error, but the total shows you how hard your drive has to work to get good results. Note that comparing bytes does not tell you how much error correction was done. As a result, it doesn't let you know how close you are to problems.

Try using different media and different speeds to see how error rates vary. You probably have two readers since a writer is also a reader. Try the same disc in both to see how dependent the writer/speed/medium combination is on the reader you use. When you try a new medium, don't just assume that if it works, it's good or even that if the files match with WINDIFF or another program you can count on the disc. Get below the level of corrected errors to see the best way to use that line of blanks.

CD Speed and Exact Audio Copy

These two programs work somewhat differently and report differently in extracting audio from your drives. Like CD-R Diagnostic, they provide insight into how well your readers work on any particular writer/speed/medium combination. Running either program, you will quickly discover that the quality of

DAE can vary a great deal with the nature of the original disc. Very few readers will deliver the same results on a CD-R that they do on a pressed disc. When you change any of the parameters of your CD-R burn, the performance of your reader(s) is likely to change as well.

Before the problems arise

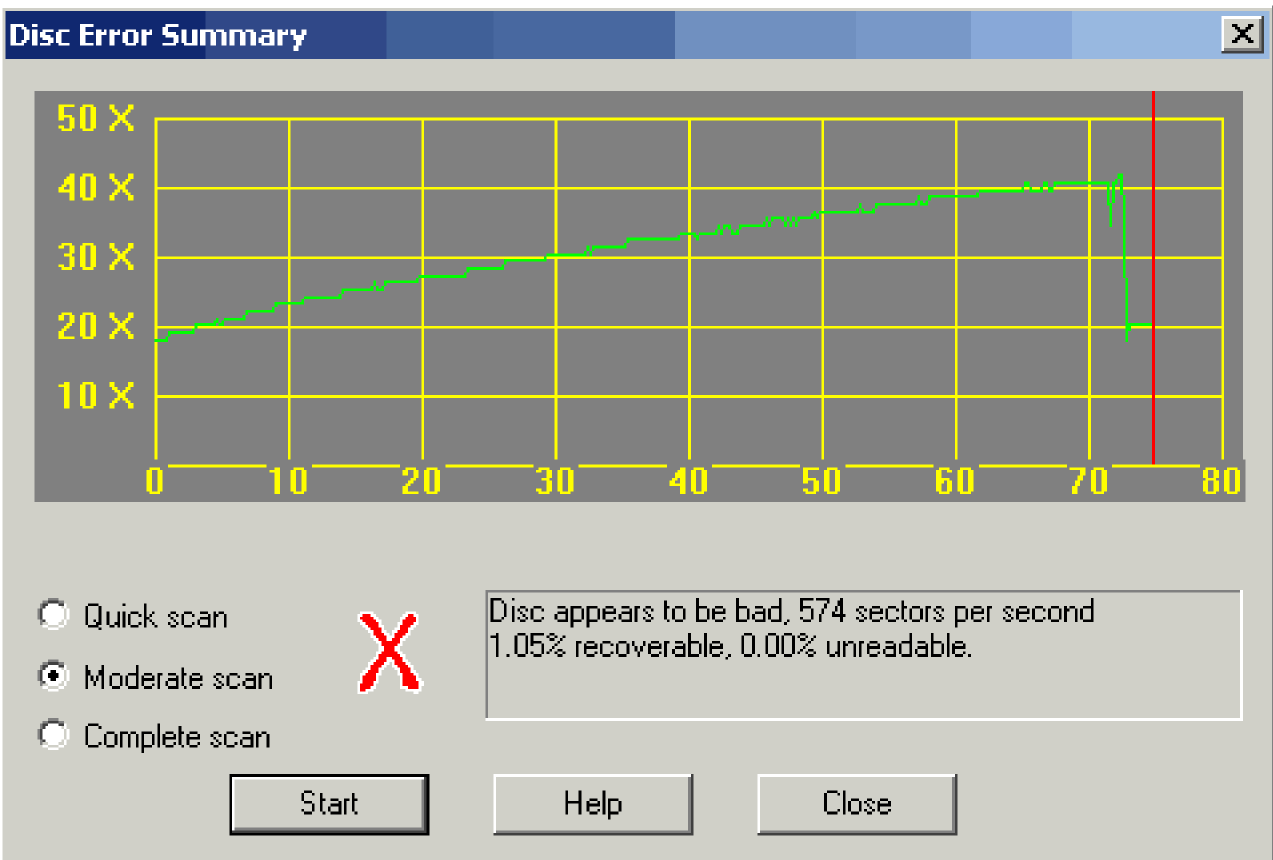
Don't wait until a critical operation fails to find out that you have a problem. Get what you need to do the analysis when everything is working right. Check out the erasables and learn the software you will need. When the crisis comes, you will be ready to deal with it. The same idea leads to another conclusion: don't wait until you run out of blanks to pick up more. Make sure you have enough on hand for any anticipated project when you order or buy your next batch. Otherwise, when you must get XYZ gold/green 80's for the job, they will be unavailable - and you'll be stuck. That's not the time to research other lines to see what you can substitute.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Interpreting Tests

Before beginning, I will admit that I have cheated a little. First, I used a disc with properties I know from previous samples. It's a "BrandX" 700-MB which writes well at 12x and 8x to about 650 MB, then picks up soft errors. The point at which errors appear varies from one disc to another and there are occasional "hard" errors which show up in the first blocks, but as you'll see they're all invisible to the simple "go/nogo" tests most people use. The second way I cheated is to use the big brother of CD-R Diagnostic, CD-R Inspector. I chose that because it provides a graphic output which is easier to see and use than Diagnostic, but everything you see below can be done with Diagnostic if you watch what's going on as you test. The disc had about 658 MB of data and it reads just fine in all my drives - a little slow in some, but without serious complaint.

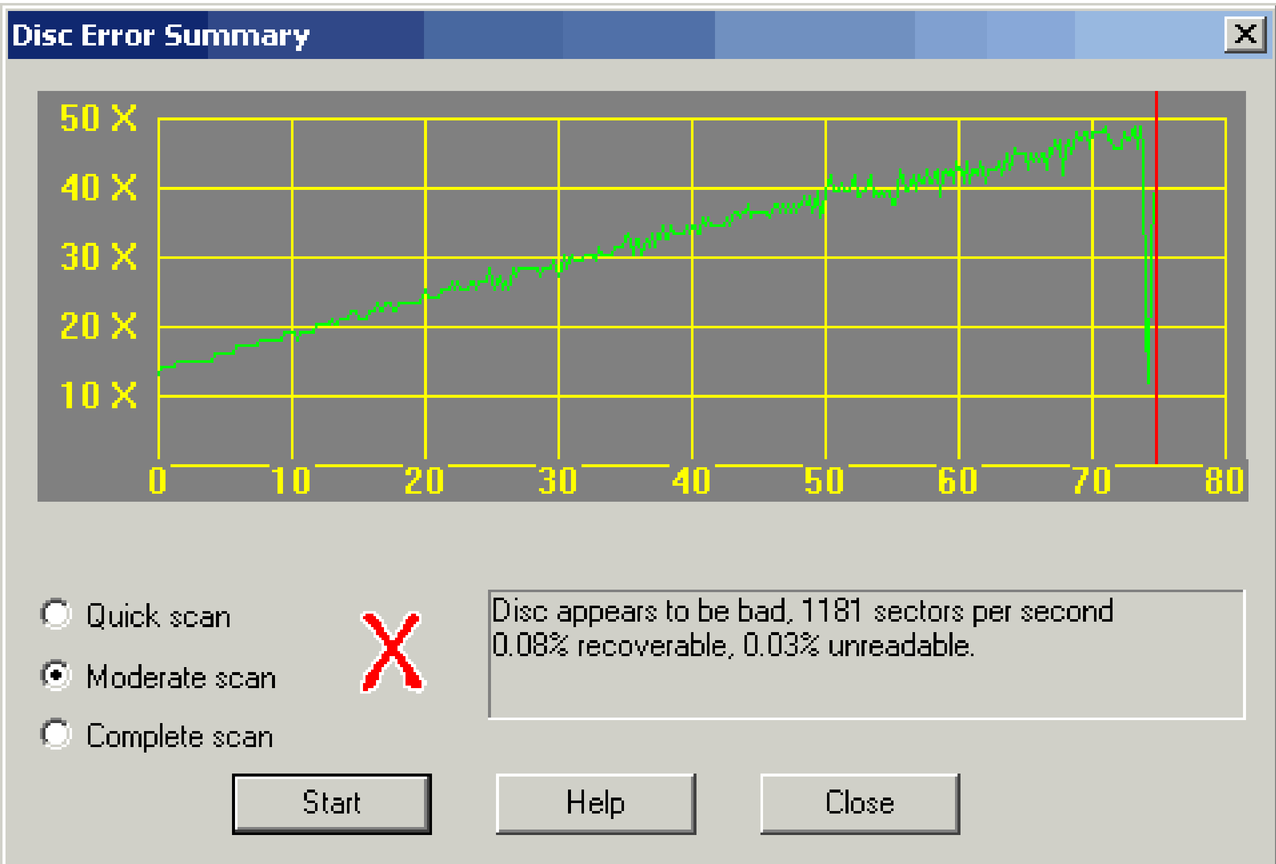
Plextor 40Max



Let's begin at the end: 0.00% unreadable. That's gratifying. There are no fatal errors; all that were detected were corrected by the error-correction code. The bad news came just before that. 1.05% recoverable errors means that the ECC was needed on 1% of the sectors read. We got through - but would we have been so lucky on another reader?

There's more information here. When the soft errors show up, the drive slows down. Those flaky blocks are reread to verify the ECC. In fact, the drive has a maximum speed a bit over 40x, but that's lost on this disc because of rereading. Incidentally, the test was repeated after the others were run and the results were identical.

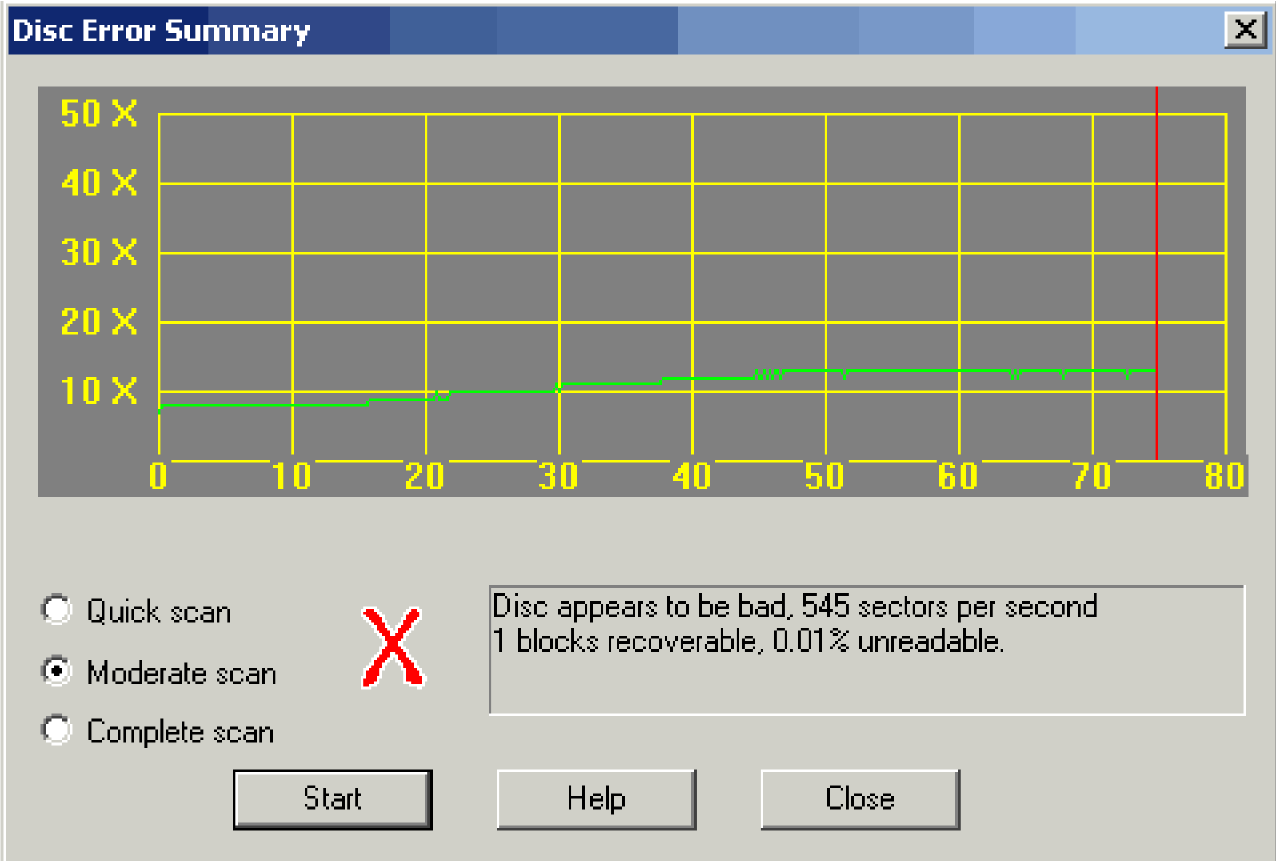
Plextor 12/10/32



Now we're getting 0.03% unreadable - errors which are not corrected. In fact, those occurred in reading the TOC and turned out not to affect the ability to recover the files. How do I know? I checked the data against the source files with WINDIFF (free from Microsoft). WINDIFF found **no** differences at the byte level. The errors were there, but it was possible to get past them - in this reader. There were fewer recoverable errors with this reader on the same disc used in the 40Max. Which, in my interpretation, means that this reader has slightly different sensitivity in reading a burned disc. As it happens, it's better tuned to the particular way the disc was written.

Note that this drive, which supposedly has maximum read speed of 32x, actually reaches well over 40x before the soft errors slow it down. Overall, it's twice as fast as the 40x. It starts at a lower speed but ramps up more quickly and doesn't hit those soft errors (which make it slow down) until later in the burn.

Plextor 412



With this drive, we get the best results of all - except for the "unreadable" (i.e., uncorrectable) errors. Again, they did not interfere with WINDIFF ultimately recovering all the bytes correctly, but that was good luck. The errors are not really unrecoverable, but too difficult to read reliably for CD-R Inspector to regard them as valid. Again, the "unrecoverable" errors were in the first blocks, which probably explains the slow start.

Conclusion

There should be no surprise here. This disc is not well matched to the writer; in fact, it is probably not well matched to any of my writers from the tests I've run on them. For critical use, it's a "bad" line of blanks. However, simplistic tests - checking that the files are readable - would indicate no errors at all. If it had been recorded with audio instead of data, it would sound fine on any of the drives. However, the errors would be there and concealed by the electronics. On another reader - perhaps a car CD player - it might well fail in several ways. It might not track well; it might produce noise or sound dull; it might not progress correctly from track to track.

Does all that matter to you? Maybe not. If you know that the line works well when written in your drive and read in your players, you may not care. But the quality is missing and if you're relying on this disc to carry information to someone with a reader you don't know or to behave in someone else's CD player, you're running a major risk.

With the low price of quality discs today, I can't justify using these 700-MB blanks for anything that matters. I have others which cost about fifty cents each and check out free of all errors in all of my readers. They are well matched to my writers and at 14 MB per penny, they're less expensive than these "free" ones.

YMMV - Your Mileage May Vary.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Keep It Clean

This note has nothing to do with X ratings. Put whatever information you wish onto your discs, but it won't get there unless the laser can see to write it.

Most of us have encountered a pressed disc - CD or CD-ROM - which would not play. On examination, it may disclose the reason: a gouge, a smear of grease, a blob of dirt or something else which keeps the laser from focussing on the pits. A simple rule: if the laser cannot see the information, it cannot read it. If it cannot read it, your stereo or your computer cannot make use of it.

The solution is usually simple enough: polish out the gouge or clean off the disc, then try again. Voila! It's fine.

Unfortunately, life is not so easy when you're writing instead of reading. If the laser encounters that blob or smear when writing the disc, the information is lost. The name for the result is: coaster. There are hundreds of millions of bytes on a CD-R. If all are good, it's a disc. If a few are bad, it may be a coaster.

If you're lucky, the coaster isn't all bad. Maybe just one faulty file, maybe a few. Or perhaps the glitch is in the TOC and nothing can be read - or it's detected in trying to work with a UDF disc and it can't be finalized. At worst, it becomes an unreadable mess and everything is lost.

That doesn't happen to me!

True - most of the time, it doesn't. You remove the shrinkwrap and take a dust-free blank from a jewel case. You burn it and - no problem. Or you take the blank from a spindle and drop the cover back on. After burning, if you get a speck of dirt, you wipe it off. So, no problem - right?

Maybe.

There's that disc you wrote with 200 MB and now you want a second session. Or the packet disc to which you add a few MB of downloads each week. Hey, it's been in the almost-closed jewel case over by the ash tray; what's the problem?

Oh.

Well, maybe you can keep it dust-free. In the real world, the best way to keep the disc free of dust and dirt is to clean it before **each** burn. Remember, you don't get to clean it after you find the problem, as you do when reading. Well, that's not true; you can clean the coaster so that it's bright and shiny in your mobile. You just don't get to clean it off and then use it as though nothing had happened - as you do when reading it.

What to do about it

First, let no coaster go undiagnosed. The first thing to do is to look at the writing surface under a strong light with magnification if needed. Look at the circle where the area you've written ends. Look closely for the bit of a hair that came at just the wrong spot. You may not find it - it could have fallen off or slipped to another spot on the disc. Or there may be another cause this time. But if you don't know what made **this** coaster, then whatever it was will be ready to create more for you.

Another move - highly recommended - is to look before you burn. Keep a can of 'compressed air' and a clean, lint-free cloth handy. Examine the writing side before you insert the disc to be sure there is nothing that can be blown away or wiped off. It makes sense to blow the surface clean before you look; it will get most of the schmutz (a term of art; see any good computer dictionary) so there will be less to see and less to miss. And even though the cloth was clean and lint-free, if you use it, examine and blow the disc afterward.

Note that these precautions are far more important when you are reusing a disc than when you are burning a new one. Even if dirt got under the shrinkwrap, all you lose is that blank. If this is a second session, you risk the first; if a UDF, you risk it all. In fact, I speculate that one reason UDF discs mysteriously go bad is that in one of the writes or one of the reads of the directory, a bit or two was lost. That's all it would take to make a directory unreadable and a data disc into a coaster.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Tick - It's the TOC

The Table of Contents (TOC) on a CD-R is the repository of all the information required for your OS to understand the pattern of bytes on the disc. Essentially, it contains everything needed for your system to figure out what information is where and in what format, so that your request to receive information can be satisfied.

For example, on an audio disc the data are not stored in files, but in a continuous stream. The TOC informs the OS (or the player) that a specific block is the beginning of a track. When you play a track, a command is given to go to that block and to begin to retrieve the bit stream which follows. Note that the track itself - in the audio sense - need not begin where the TOC says it does, but that is irrelevant to the player. There are also subcodes which can be used to interpret how to backspace and so on - but they are beyond the present scope. (And if you want to read that as meaning that I don't understand them, I couldn't argue.)

When you go to write a Disc At Once (DAO), all the information needed for the job is available when you start. Mastering software assembles that information and decides what is to be in the TOC. When everything is ready, the laser is positioned and turned on and writing begins. The **first** thing written is the TOC which says: this is a closed disc of a single session and here is the information on what you will find where among the following bytes. The laser then proceeds to write those bytes in sequence and in accordance with the information saved to the TOC. When the last byte is written, the laser turns off and you have a good disc. If that burn is interrupted after the TOC is written but before the end, everything is accessible up to the failure. An audio disc which fails when track 5 is being written will be usable for tracks 1-4 and maybe for the written part of 5 - but what happens when the track runs out depends on the player.

When you write Track At Once (TAO), the procedure is quite different. The writer begins by inscribing a special area on the blank called the Program Memory Area with information needed to close the session later. Then it records track information and writes the track itself. When it's time to close the session, the writing laser turns off, the mechanism returns to the PMA and the information inscribed there is read back to close things up, record how many tracks there are in all and, in general, to complete the TOC. So if the burn fails before the PMA is read (or if the PMA cannot be read at all), the TOC is not complete and a reader cannot make sense out of it. However, not all is lost! Although the reader needs a complete TOC to do its job, a writer is prepared to make do with less and, in fact, has what it needs in the part of the TOC which is complete to retrieve the valid data.

Suppose that you have written your disc - and the software reports errors. If the error has to do with reading, then it is almost certainly the inability to read the PMA when it came time to close the session. The disc is trash, but you **can** retrieve the information from it in a writer. That information may not include what was on the last track, so if it was a data session you're in trouble. But if it was an audio session, you should be able to recover all the tracks but the last. Note that the particular error message you get is critical here. If it says buffer underrun, the data are not there so they can't be read and (in TAO) the TOC was not written, so you couldn't find them if they were. If it reports a communication error, it might well have occurred when seeking for the PMA; in that case, your writer may actually be able to read everything on a TAO disc. Finally, if a communication error came at the end of a DAO burn, it may mean only that the runout track was incomplete. Since most modern readers need very little runout, the disc may work perfectly for you although it is 'too long'.

E-mail me at cd-r@mrichter.com

Return to Mike's home page

Backup

There are three distinct types of backup to CD-R:

- Copy selected files from a computer system to disc
- Duplicate a CD-DA or CD-ROM
- System backup for restoration if needed

Copy selected files

There's not much to say about backing up your current working files from your HDD to a CD-R. There are basically two approaches, used by different programs. CDRWin typifies one: replicate a folder on the CD-R with fully qualified paths to each file. The other approach, used by ECDC and others, is to organize the CD-R like a separate drive, with its own folders (invented by you or dragged from the source) and with files pulled from various places at your pleasure. You might replicate a folder if you want to remove it from the drive and be able to restore exactly as it had been. You might mix and match with ECDC in order to organize scattered files into a single project.

Duplicate a disc

Of course, there are the easy ones which simply copy with a mastering program. Slightly more difficult to handle are those which appear to be oversize because of a distorted TOC (see the page on Oversize discs); those can usually be duplicated through a disc image. But there is another whole world of 'protected' software which cannot be duplicated in a straightforward way. Unfortunately, I cannot help with that.

There are moral issues of piracy, right to maintain an archival copy, licensing for use relative to ownership, and so on. They are important to me and I don't know how I'd deal with them if I were able to cover this subject at all. But I do not play computer games and very rarely copy any disc except those I create, so I simply have nothing to share. Hunt around the newsgroups for the pirates' dens - look for alt.binaries and warez - if you wish to learn about this subject.

System backup

Again, I plead ignorance: I do not know enough about Mac OS or Linux to offer anything of value. So this deals only with backup of DOS/Windows - and even then, DOS and Win 3.x are simply assumed to go along when the backup is being made. For that, there are two different approaches to backup: file level and disc image.

File-level backup transcribes each file with its fully qualified name. It operates in a manner similar to XCOPY - and XCOPY can do it. So can PKZIP, which will also compress the files and reduce the number of entries needed for the final TOC. Where position of the file on the disc does not matter, that's all that you need. Unfortunately, a full system backup includes some files whose position **is** critical. The system files with which your computer boots and certain files used in copy protection are characteristic of position-sensitive information.

Drive imaging means compressing the information on a drive so as to preserve positioning as well as all data. It is also needed to capture certain files which are opened when you enter 32-bit Windows and are therefore inaccessible to ordinary programs.

There are two approaches to imaging a drive or a system: capturing directly to CD-R or capturing to a conventional drive (hard disc or removable) for mastering later to CD-R. Direct capture requires some form of packet writing. It is slow but convenient; it requires a writer which supports the type of packet writing the backup program uses. There are many programs which use one or both of those approaches, including Ghost from Symantec, Veritas' Backup Executive and PowerQuest's Drive Image. The most flexible program I have seen and the one I use routinely is TakeTwo, part of Adaptec's Easy CD Creator 4 and above. For that reason, I use it here as an example of the options and their uses. Other programs provide subsets of its features which may be sufficient for your needs and desirable for other reasons - such as being bundled with your drive or having a user interface which you find easier to navigate.

TakeTwo backs up from within Windows either directly to a supported CD writer or to any partition - even the one which is being backed up. Writing directly, it automatically spans discs and formats them if required during the backup. Writing to another device, it permits you to specify the size of each constituent file, so you may use it to write to Zip cartridges or to HDD files which will later be mastered to CD-R.

TakeTwo restores a drive from a cold boot, using a single floppy which it creates for you. While that disc works 'as is' on most systems, it may not on yours, so test it before relying on it. If you need a driver to access your storage medium it may not be supplied on the disc; then you will have to add it yourself. Since the drivers included handle all Adaptec SCSI interfaces and most ATAPI devices, the odds are that they will be sufficient. The program also supports simple retrieval of files and folders from within Windows, allowing you to recover them to any location you choose.

Again, the above is not intended to be an endorsement of any specific program. TakeTwo is my choice because of the way I have configured my system and the drives and devices I employ. Other people prefer other software for reasons which make sense to me. You may well be one of those "other people."

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Oversize Discs

The capacity of a 'normal' CD-R, like that of a normal CD or CD-ROM, is 650 MB of data or 74 minutes of audio. Sometimes, one runs across a source disc that seems to be larger or a CD-R that claims to have more recording capacity. Those phenomena may be real or illusory and they can be handled in various ways. This note is intended to provide guidance in copying the uncopy-able and similar issues. One suggestion I do make is: Don't break the law. Backing up a disc you own is apparently legal. Saving your own files onto CD-R is certainly OK. Otherwise, please be careful; I don't want to be an accessory to theft.

Estimating size

First, capacity of a disc is measured in megabytes - but how large is a megabyte? Is it a million bytes? a thousand kilobytes? or 2^20 bytes? A thousand kilobytes is 1,024,000 bytes or about 2.5% more than a million bytes; a true megabyte (1,048,576 bytes) is about 5% more than a million bytes. Since there is no enforcable standard for the word 'megabyte', manufacturers of CD-R's get away with claiming almost 5% more capacity than is 'really' there. It's an accounting trick that will not let you store one extra character on the disc.

There are CD-R's with more than 74 minutes' recording capacity. The maximum possible is a shade under 80 minutes. There are also 63-minute discs, but while they are said to be more reliable than 74's, they are again hard to find and no less expensive than the longer ones. Even the 80-mm blanks with about 15 minutes' capacity may reappear. (There are also recordable minidiscs, but this is complicated enough already!) The result is that under all reasonable circumstances, you will be recording on a 74-minute, 650-megabyte CD-R. If you need an 80-minute blank, you will need to hunt it down.

Where did the megabytes go?

A data CD-R gives the impression that it is written with a FAT and structure just like that of a hard drive, but it isn't. As a result, there is information stored on the CD-R that takes away capacity in hidden ways. The amount 'wasted' depends on the type of recording being made. More to the point, CD-R software does not usually estimate the capacity of the disc precisely. A similar phenomenon occurs with audio CD-R's. As a result, you may be 'cheated' out of some tens of megabytes. Even worse, you may not know about it until you have burned the disc - and find that it cannot be closed.

The size of the Table of Contents (TOC) which consumes CD-R capacity depends on parameters you set when recording. To minimize the wasted space, record DAO. If that doesn't solve the problem, try writing an image to your HD. That image is a byte-level preview of what will be written to the CD-R, so the software need not estimate whether or not it will fit but can actually count bytes and sectors to prove it. Another test (not a perfect one) is to do a test write. Most cases of trying to write more than the disc will hold will be caught in the process and all you will have wasted is the time for the test. You may even run a test on an ISO - just to be certain. As usual, different writing software has different ways to do these things, so check your manual.

When you are preparing to write a CD-R, your software may estimate that you do not have enough room. Some programs simply refuse to proceed. Others inform you and blithely try to do the impossible when you tell them to. If you are near capacity, write the image, then test writing it to the disc you want to use. If that works, cross your fingers over a rabbit's foot and burn your copy. And remember what happened to the rabbit.

Disc length and Overburning

Although standard discs are marked 650 MB and 74 minutes, in fact they vary significantly in actual length. The true size the manufacturer designed in - as blocks or another measure - can be found with appropriate software such as the Disc Information of Easy CD Creator 3.x and CD Information of ECDC 4.x.

Overburning refers to stealing some of the space beyond the manufacturer's intent to hold data. The leadout/runout track is standardized at about 13 MB, but not all readers need the full length. With some hardware and software, you may use that space for data or audio - but you should be aware that that may leave the disc unreadable in some drives. This procedure is risky because it leads to apparent success and works often, then can let you down when the disc is most critical to you. I suggest that you look into other choices - such as an 80-minute blank - before overburning even if your system has the capability. Still, if you are in a critical situation and *must* find an extra couple of megabytes, it may be worth trying.

The manufacturer specifies in the ATIP the number of blocks which the disc is prepared to hold. Given the parameters of the spiral, that should take the write laser to the specification limit. In fact, the spiral probably goes somewhat farther than that and you can probably get away with a shorter runout track than specified - so it is possible to overburn.

Overburning has nothing to do with SCSI or EIDE. As I understand the limitation, some drives use the ATIP information to refuse burns beyond the block count and others leave enforcement to the writing software. There is no command to "overburn".

The software commands the drive to write specific information to specific blocks of the disc. That is not on a block-by-block basis because of positioning, but in principle, the software says: put this there - and the hardware does it. There are several cases of interest for a standard (650-MB) blank.

1. The software says: write in a block within the ATIP count and the hardware does it
2. The software says: write in a block beyond the ATIP and the hardware refuses
3. The software sees that the write would be beyond the ATIP and does not command it
4. The software commands a write beyond the ATIP and the hardware tries to do it

Case 1 is not overburning at all - it is the normal condition. Case 2 means that the hardware precludes overburning Case 3 means that the software precludes overburning Case 4 means that overburning is attempted - leading to four additional cases:

- 4a) Everything goes fine, the burn completes and the disc is successful
- 4b) The burn goes beyond the available spiral and part of the write does not occur

- 4c) The carriage hits a stop and part of the write does not occur
- 4d) The burn runs out of spiral **and** hits a stop

4b-d may also be successful burns since most readers will be happy with less than a full runout track. Any of them is more likely to have created a coaster and, depending on the design of the drive, 4c and 4d may do physical damage.

There are two situations of interest to add to the above when the ATIP indicates a blank longer than 74 minutes. Obviously, no drive will accept an arbitrarily large block number - the maximum value will certainly correspond to a blank shorter than 7400 minutes. There are old drives which were designed with a 74-minute maximum; they will not burn 80-minute blanks at all and the software will not - or, at least, should not - allow you to write beyond that limit. A special case is one group of Sony designs used in some Sony and HP drives. They simply ignore commands beyond a certain point, corresponding to about 78:30 on an audio disc.

In all of this, you need to remember that the reader has only a very limited vocabulary when talking to the computer. It does not shake hands on each transaction or report regularly on the status of its operation. When it knows that something specific has failed, it says so; otherwise, it runs open-loop, trying to implement the commands it receives. It does not report the power calibration it achieved, only that it failed to find one. It does not report the level of its internal buffer, only the fact that it has emptied. And so on.

Oversize source

There are two distinct cases here: a source CD or CD-ROM which is too big to copy, and a set of files under your control which exceeds the capacity of the CD-R. In fact, there's a middle ground of considerable interest.

First, some CD-ROMs are effectively copy protected and cannot be copied at all. (More precisely, some can be copied but the copy won't run.) The mechanisms include physically altering the medium as well as creating software conditions that some or all CD-R software will not handle. If you wish to backup such a CD-ROM and if you have reason to believe that the protection is effected only in the data (not by the physical medium), try other writing software to see if it can do what your preferred program won't. That may cost you some coasters, but you must expect to pay a price for circumventing the vendor's efforts. (Incidentally, there may be information in a newsgroup or mailing list that will be of assistance.)

Next, there are tricks that may be played by the CD-ROM publisher with files and directories that make the amount stored appear to be larger than it is - sometimes by a factor of two or more. If you make an image of the source disc and find that it fits onto your CD-R, you're home free. Incidentally, those tricks may be used for purposes other than to inhibit copying, so paranoia may not be appropriate.

In the category of files under your control, the solution is obvious: Control the files. Save some for another disc. If you are archiving, consider zipping text, database and other low-density types. Resize images or use a bit more compression. For audio, drop a verse or a chorus or repitch the selection. 10 MB is about 1.5% of the capacity of a CD-R. If everything you did were 98.5% efficient, you wouldn't be likely to complain. So, don't do it here, either.

Finally, if you have a disc that is stubbornly just a bit too big to copy, don't waste your time trying to figure out how the publisher did it. Retrieve all the files to your HD and remove the read-only attribute on the data. Now you have control and can apply the same tricks you would if you had created the files in the beginning. Perhaps there's a directory full of information for a platform or OS you don't use; kill it. There may be 500 graphics files you don't care about; scan for any worth having and dump the rest. Use your imagination and your good judgement. (One useful trick if you have a Jaz drive or a blank HD partition: create your reduced disc on that drive and run it. If it works, burn it; if not, make the appropriate adjustments.)

A Note on Microsoft's Way

It should not be a surprise to you that Microsoft knows their operating systems - and how to break the rules that they impose.

Much of MS's software is cross-platform: it runs on different systems with only moderate changes depending on the host and its configuration. An easy way to develop that code is to have the common elements in one set of folders developed by one group and have the parts particular to a specific platform developed by a group of specialists in that platform. For ease of control, the platform developers are probably given read-only access to the common code in folders that appear to belong to them. In effect, the developer for a DEC system builds her own CD-ROM using the common code and her own interface to the Alpha. She can then create a CD-R for the Alpha and test it. However, before release the Alpha and Intel (I386) packages need to be combined onto a single CD-ROM for distribution.

It is easiest to keep all the references to common code in the platform-dependent parts unchanged. To accomplish that, MS can and apparently does fake the references in the TOC. That is, the common code appears to be in both the Alpha and the I386 folders. It is actually present only once, but the table which references it gives the appearance of providing two copies. In many cases, that results in a disc which seems to have much more than 650 MB on it. Note that this illusion is developed as a programming convenience, not to provide 'copy protection'. You can probably back up a MS disc which appears to be oversize either by dropping unneeded platforms from your copy or by making an image and burning from there.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Audio Basics for CD-Recording

One of the functions many people seek with CD-R is creating audio CD's. Software for creating CD-ROM's supports that desire, but each product varies in the ways in which the job is done. For information on your software - **read the manual.** This note is intended to cover the basics of audio file formats and problems.

Jump to Recording with D/A/D conversion
Editing a WAV file
Recording on the fly

CD Audio and WAV files

A selection on an audio CD is very nearly in WAV format. It is a variant of a specific type (redbook) of WAV file: 44.1 KHz sampling, stereo, 16 bits depth. Unlike an ordinary data file on a CD-ROM, the sound file does not contain error correcting codes (ECC) within it to handle data lost in transfer. Instead, a layer of ECC is provided on the disc. One of the ways in which CD players vary in sound quality is that they are more or less effective in using ECC to repair errors in reading the audio.

NOTE that an audio player is required to read only the first closed audio session on a disc. To avoid problems of poor readability, mysterious noises and so on, the beginner should not try to leave the disc open when closing the first audio session. You may write the disc in DAO, write the whole session at once in TAO, or write repeatedly *leaving the session open* until the last track, when you should close the disc. When you understand CD Extra and are willing to experiment, multisession can be fun.

Getting a WAV file

There are two different ways to create a file on a CD-ROM: an audio CD track or a WAV file. Only the former will play in a CD player. However, CD-R software will convert a WAV file (CD Creator will take one at less than redbook standard) into the necessary format and will read the audio file from a CD into a redbook WAV file on your hard disc. In that process, your reader may run at anywhere from 1x to its maximum speed. The higher the speed of reading, the more chance that some information will be lost. Since there is no effective error correction, those faults will show up in the WAV file. Fortunately, that file is on your hard disc and you can listen to it. If it sounds good there, it should sound good when converted to the CD format and recorded. If the copy on your disc has clicks or other problems - don't give up.

The first step is to persuade your software to read the CD at a lower speed. First, you need a player which supports Digital Audio Extraction (DAE). Your CD-R certainly will do that; a CD-ROM reader may or may not. Most SCSI readers provide DAE; very few IDE readers do, and then only with the right software. If your reader does not support DAE, you can use D/A/D conversion or your CD-R. Let's assume we have DAE. Using whatever tools your software provides, cut the read speed down until you get a clean WAV file. That tells you how fast you can run with that CD. Now check with other CD's to find out whether your first one was unusually good (or bad). Ultimately, you will decide that you can use a particular speed reliably. Don't try to cheat on it: If your reader won't give you a good signal at 4x, no software or CD-R media or whatever will improve on that.

Note that you are now beginning to understand why some 8x CD-ROM readers are slower than some 4x. They may run at 8x under ideal conditions, but if they aren't able to retrieve information reliably, even error correction may not be enough to get a good file. So the system rereads that file until it comes out right. There's no standard around to say what 8x (or any other speed) means - except that under some circumstances as many as 8x150 KB/sec will come from the drive. Whoopee!

D/A/D

The information on a CD or CD-ROM is stored in digital format. To listen to that signal, it must be converted to analog (a.k.a. analogue to us old-timer purists). The conversion is done in the reader and uses the ECC of the disc to a greater or lesser extent. If you have put an appropriate jumper from your reader to your sound card (so you can listen to a CD while you 'work'), you can record from that with controls on your mixer. If not, any CD player can be connected to the Line In jack on your sound card - as can any other appropriate source of analog audio. The quality of the signal going into your sound card then depends on the quality of the CD reader/player you are using.

On the sound card, an analog signal is converted to digital - and with that comes a set of problems. Essentially, converting between analog and digital signals is a matter of approximation. D/A is hard enough, but A/D is tougher still. The result is that only very good and very costly equipment will do first-rate D/A/D. Is your present sound card good enough for you? No one can answer that but you. Would the BrandX SuperSound do enough better to be worth the cost? 'Better' depends in part on your taste, and 'enough' is meaningless to anyone but you. Perhaps someone on the mailing list or in a newsgroup can give you insight, but don't expect easy answers. There aren't any.

At least, if you do use A/D/A instead of DAE, you will use the CD's ECC. If a CD you want to use insists on crackling on DAE at 1x, you may have no other choice.

Editing a WAV file

One of the advantages of making a WAV file through DAE or D/A/D is that you can edit it. Tools for that purpose are included in the audio section of the links at this site. Those editors will let you modify the sound in many ways: denoising, adding effects, repitching, cloning, ...

There are some tricks to this operation. One is that some software figures that whatever's tucked onto the end of the file for its own purposes won't do any harm. As far as the computer is concerned, that's so. However, on a CD those bytes become a 'click', and for some strange reason most people don't like extra clicks and pops on their recordings. Here, the solution is easy: don't end your editing with such a program. If you want to use an editor which you know leaves the click behind, do it. Then open the file in another editor that you know doesn't make popcorn (such as GoldWave) and save it. The click will be left behind.

Recording on the fly

"On the fly" means writing the CD-R from the source (here, a CD reader) without going through the extra step of writing to the hard drive. Let's dispose of the obvious first (not that you don't already know it, but for the benefit of those who don't). You cannot read from the CD-R while you are writing to it; therefore, you need a separate CD reader to record on the fly. Recording on the fly requires that the data being read are digital, so your separate reader must support DAE. If you don't want to use a second, DAE-capable reader, you cannot record audio on the fly. Given current prices for a SCSI CD reader under $100, it doesn't take many coasters to pay for a second drive - and you have a SCSI adapter already for your CD-R, so you probably don't need to buy one of those.

When you go to record on the fly, your software assumes that you can run at full writing speed without a problem. So you're tempted to think that your CD-R that writes at 2x will work fine with a reader that runs at 2x or above. You may be disappointed. For audio, you should already have determined with the WAV tests above that your reader is only reliable at 1x - even though the box says 2x, 4x, or 45.7x. The same thing that keeps a WAV file from sounding good will guarantee ticks and pops when you record on the fly. Therefore, you must slow down on-the-fly recording to the highest speed at which your reader will work. Of course, if that speed is faster than your CD-R can write, you can't beat the CD-R hardware by getting a faster reader. There's still no free lunch.

All the other stuff

Recording a CD is much like recording a CD-ROM. Some combinations of media, hardware, firmware and software work better than others. If gold/gold media work better for you for CD-ROM, they will probably be better for CD, too. Just because your hardware vendor's bundled software has a good interface for CD-ROM does not mean that you will find it ideal for audio (or *vice versa*). There are at least three choices of CD-R software on each major platform and each does every part of the job differently from every other.

There is no 'best' hardware, software or medium. The mailing list and newsgroups will provide you with unauthoritative, inaccurate, contradictory, incomplete and irrelevant information. Still, it's better than buying and learning all those products.

At least I think it's better.

From someone who knows: Pauses in Audio

From: Bart Lynn - blynn@eng.jvcdiscusa.com

As an Engineer for a CD manufacturer, I feel that I can try to explain the differences in the pause times on CDs. Red Book (the Compact Disc Specification for Audio) states that the Pflag must be 2 ~ 3 seconds. Therefore, since Absolute time starts at 00:00:00, the start of track 1 can be from 00:02:00 to 00:03:00. Of course, there can be silence after that point, so there can be in essence a lot of pause.

Most PC/Mac software sets the start of track 1 at Atime 00:02:00, since Yellow Book (Compact Disc

Specification for normal CD-ROM discs) specifies that the pause be just 2 sec. At JVC, we start at 2 seconds Atime, then we offset the first track 1 second, prior to the start of the music (CD-ROMs start at exactly 00:02:00). Each additional track is offset 5 frames (30 SMTPE frames to 1 second). The end time of the last track is moved forward 1 second. The reason is that many old CD players (from like 1985, up to the triple beam ones) search for the start of the track by the Pflag, not the absolute time. The new players can be extremely accurate in searching, where the older ones cannot. As a manufacturer, we have to try and supoort all makes and models, so we have to provide a slight tolerance for them. Additionally, the record label/artist sometimes requests that we provide additional pauses, especially for classical music, or live recordings.

Bart Lynn
JVC Disc America

Extensive additional material is available at http://www.westnet.com/~gsmith/ - but let me caution the unwary that the subject is complex. Those pages may be difficult to digest, but can be worth the effort.

E-mail me at cd-r@mrichter.com
Return to CD-R home page

Analogue Audio

Let's start with the basics. Sound is a pattern of pressure that we sense - usually thought of as sound waves in the air. The usual way to deal with sound waves in electronics is to create an electrical analogue by exposing the sound waves to a microphone. The signal from the microphone can be amplified and recorded - still in analogue form - on audio tape or on a vinyl disc. It can also pass through an Analogue-to-Digital Converter (ADC) to become a digital signal. One of the functions of a sound card is to provide an ADC to deliver a digital signal corresponding to an original pattern of pressure. Another function is to synthesize a digital signal from instructions in the form of a MIDI or similar file. On the other side, the sound card provides Digital-to-Analogue Conversion (DAC) which is then used to drive tape recorders, speakers or other analogue devices. The inputs used for the purpose may be WAV files, MIDI synthesis or other **digital** signals the card accepts. Finally, the sound card allows mixing of its various analogue signals on input and/or on output. So the card will allow you to mix the analogue signal provided from its CD input with a synthesized sound from a MIDI file to drive your speakers with the combination. And a "full-duplex:" card allows such an analogue signal to be digitized while being played.

There are two ways to get sound from your CD reader. First, there's a headphone output with a volume control. Run a cable from that output to line in on your sound card, and you're ready to go just as you would be if the source was a tape deck. Even better (especially on internal drives) is to connect a special, analogue cable from the reader's plug to one marked CD on your sound card. The sound is the same at the two outputs, but with the internal connection you don't have to remember to twiddle the headphone volume control and you don't have that cable hanging outside the case. Finally, most CD readers support Digital Audio Extraction - DAE - which gets a page of its own in this primer. Recording from the CD as analogue uses the DAC of the reader to generate the signal, then the ADC of the sound card to convert back to digital. Each of those processes is imperfect and can degrade the sound. However, if your DAE is not perfect, the analogue connection may sound better to you - so, use it! Only the purists will insist that you should put up with what you don't like because It's the Right Way. One more point: if you have two CD sources, such as a reader and a writer, do **not** try to connect them both to the same jack on your sound card. Either get a card with dual stereo inputs or run that cable from headphone out to line in. (Connecting them both to the same input **may** work or may appear to, but can have unfortunate consequences.)

Let's look more deeply into the question of recording from CD-DA. Since you are likely to have both options, should you use DAE or analogue recording? The answer already given is: use what works for you. But there is more to it than that. Assuming that you **do** have the choice (good DAE and an analogue connection), when should you use analogue? First, some recordings have an extra bit set for preemphasis. That is a shift in the frequency response away from flat. Under some circumstances, it can let the producer record a **LOUDER** signal. If it's set, then DAE from the disc will sound too bright and lacking in bass. If all you're going to do is write that extracted file back to a CD-R, you can live with preemphasis as long as your software allows you to set the bit when you record. (Adaptec's PC programs do not.) If you want to edit the WAV file, you will definitely want to correct the preemphasis first; for that, you will find it easier to record analogue instead of ripping the track with DAE.

Another factor is error correction, which in general is beyond the scope of this page. The important thing about it is that errors recorded onto a CD-R or read from a CD-DA track will get through DAE without correction. But when the reader sends the signal to one of its analogue outputs, it passes it through filtering circuits which clean up some of those errors. The more cleanup the circuits do, the more they alter the sound quality. One reason that a copy of a CD-DA track may differ from the original is that the filters have more to do (CD-R has more audio errors than a pressed disc) and therefore change the sound more. Should you use analogue or DAE on a CD-R? Whichever sounds better to you - but be aware that they will have some differences in sound.

There's a lot of good material at http://homepages.nildram.co.uk/~abcomp/lp-cdr.htm with details not covered in this low-level page; stop by for more when you're ready.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

What's this DAE stuff, anyway?

There may be nothing about CD-Recordables more confusing than Digital Audio Extraction. So, let me try to explain what it's all about and why it is an issue. As usual, corrections are invited; I am not an expert in this field and can be accused of sharing my ignorance, not my knowledge.

In the beginning, the compact disc format was defined for digital audio - CDDA. The rules for CDDA are defined in a standard called The Red Book (not the Little Red Book - that's the one with Chairman Mao's wisdom). CDDA says that there's a small file (in the computer sense) on the disc which identifies the tracks that follow. Then there are the tracks with a precisely specified sequence of bits in which two channels of audio at 16 bits per channel and 44,100 samples per second are interleaved. Those tracks are not files - they are simply tracks with strings of bits. The only files on the disc are those little pointers which tell the laser where to go to get a track, how long it is, etc.; they could hold a bit more information, but since few CD players would use it, few publishers bother to put it in.

The difference between a track and a file is worth looking at. After all, a WAV file may be an interleaved pair of strings with the same properties as the track. But the WAV file contains information about its contents, for example, the fact that it's sampled at 44.1 KHz; it also has an associated CRC (cyclic redundancy check) which lets the retrieving software determine whether it has been read correctly. When stored on a CD-ROM, it has an extra layer of error correction; the track has a minimal level in the hardware that we'll discuss later. In short, the WAV file carries overhead, 'wasted' space, with information not in the CDDA track.

When a CD player recognizes an audio track, it routes the bit stream to its DAC - digital-to-analogue converter. There the minimal layer of error correction is applied to patch up some errors in a disc that's in good shape (it cannot do much more) and the analogue sound output is generated. Since an error is usually in a single sample and a sample lasts less than 25 microseconds, you're not supposed to notice it. However, we would like to capture that digital data stream to our system in order to record it onto a CD-R and perhaps even to edit it. That means we wish to take raw data extracted from the audio track and convert them into a file that can be stored, modified and written with a computer. To do that, the reader must feed the digital data to an output that the computer can read. If the drive will provide that information, it is giving you Digital Audio Extraction, DAE. Not all CD-ROM readers do that, so some simply cannot provide the signal that's needed. You can run any software you wish, if it won't provide the output, the output won't be there.

In some cases, the firmware of the reader can be changed to provide DAE; in others, it's a hardware problem - period. In addition, something must tell the drive that the computer wants the DAE output - that the signal should be diverted from the DAC to the digital output. It can also tell the reader how fast to try to read the CDDA stream. Another critical need when the computer is trying to convert the stream of bits into a file is that they come through on time and in sequence. Again, if there's a bit of inconsistency in the audio stream, we probably won't hear it; but the computer is not as forgiving as the human ear and it wants each sample to be meaningful, not just 90% or 99% or even 99.9%. So where a sample or two can be dropped every so often while listening to a CD, none can be lost if the data are to go into a file.

Thanks to all the error correction on digital data, a CD-ROM reader may run at 4 or 8 or 20 times the speed that a CDDA is supposed to play. But without that correction, the audio stream may not be able to run at full speed. In fact, on some supposedly high-speed CD-ROM drives, CDDA may not even extract at 1x - 150 KB/sec. In other cases, there are so many errors that the resulting file is unacceptable. Again, the problem is not the software but the reader's hardware and/or firmware. All that the software has done so far is to command that particular reader to read CDDA as digital data to the appropriate output port, receive the data stream, and slow down the reader if the data are not usable to build a file.

Finally, the software comes into play to do a bit more. It takes the received data stream of a track, converts it to a WAV file and writes it to the hard drive if that is being used as an intermediate. By writing to the HD, the DAE is allowed to be slower than would be needed to write directly to the CD-R. If you do not go to the HD, then noise or something else may slow the reader enough to underrun the buffer on the writer (it is not getting data fast enough to keep writing) and your CD-R is another coaster. Writing the data to the HD buffers the information so that varying read speed is acceptable. In addition, you can listen to the WAV file that the software records to confirm that it's good enough to write to your valuable blank. Of course, if you are sure that the reader and the CD-ROM are good, or if you are in more of a hurry to get a copy than you are in need of a good one, you are free to burn 'on the fly' - directly from the reader to the writer.

Most readers do not use even the little error correction available when performing DAE. Only the Plextor family consistently corrects the data stream on the way to the digital port. And few drives but the Plextors extract CDDA as quickly as they do digital data. Those using Plextors for DAE are not praising them because Plextor is paying them; they are simply sharing with you their best advice on saving money and time. You are free to ignore their (our) recommendations, but don't expect a whole lot of sympathy when you do. And I am sure that they/we will be happy to let you know when another vendor delivers quality signals from a quality product. Until then, we will have to restrain ourselves from saying: 'I told you so' when you find that you spent more for your 24x EIDE than you would have four a 4+ and cannot get decent DAE even at 1x.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Recording from an Analogue Source

Okay, folks - here we go on how to record your favorite LP's, cassettes, 45's, DAT's, 78's and so on to CD. These notes end with getting a clean WAV file. From there, you use advice elsewhere in this primer or wherever to make your CD-DA, MP2 or MP3, or???

Sources

Most analogue audio sources deliver about one volt output. One important exception is a turntable; most cartridges for vinyl or shellac put out less than a hundredth of that level. If your cartridge is crystal or ceramic (few are), you may be able to connect your turntable as though it were a tape deck. However, in general you need to **preamplify** the output of your cartridge in order to record it on your computer. (**NOTE:** the preamp also corrects the frequency response of a cartridge.) You can buy a separate preamplifier at Radio Shack or any other general-purpose electronics store or build one with a circuit from my Files page. If you can't find one for a turntable, one made for a microphone may do. Even better is to use a receiver or a high-fidelity preamplifier. They may show up at a pawn shop or a swap meet. If you have a choice on a stereo unit (receiver or preamp), get one with two sets of tape input/output connectors. The output to use is one pair of connectors marked to go to TAPE REC. Output from your sound card goes to the corresponding TAPE MONITOR inputs to the preamp. In my setup, I have two high-level sources (open-reel decks) connected to TUNER and AUXILIARY, a cassette deck on TAPE 2 (REC and MONITOR), the sound card LINE IN connected to TAPE1 RECORD, the sound OUTPUT connected to TAPE 1 MONITOR, and the amplified speakers connected to the preamp OUTPUT. I can record from either open-reel deck or the cassette or phono to the sound card or from phono, an open-reel deck or the computer to the cassette deck. Just one warning: watch out for feedback loops when you're using the cassette deck.

For some purposes, it's sufficient and more convenient to record the turntable output to tape, then to feed the tape signal into your sound card. There are some other sources to discuss: Digital Audio Tape (DAT), MiniDisc (MD) and turntable. DAT and MD provide analogue outputs and can be connected just as tape decks are. Some also provide digital output; those can be connected digitally if your sound card has SPDIF interface and if the signals are (or can be made) compatible. But that's 'way, 'way beyond the level of a primer (which translates into: Mike doesn't know about that stuff).

Getting ready

Now that we have some sound going into the sound card, we need to look at getting it recorded. The first step is telling the card what input to read and to set its level. Unfortunately, most people are stuck with the absurd Voyetra mixer that comes free with Windows. If you have a better one, use it. (If you're a programmer who wants to do a service for humanity, how about making a freeware replacement?) Start the mixer - usually with a right-click on the speaker icon in the tray; otherwise, it's Volume Control in Accessories. Now, go to the Options menu and select Preferences. Click on Record and make sure that all the inputs are checked before you click OK. Now you have sliders for each input available to you. Pick the input you want. (CD is for the analogue connection of your reader or writer; unless you want that, you should probably select Auxiliary - which is their substitute for LINE IN.) Now, set the slider near the top of its range. You have just told the sound card to record from the selected channel at about the right level. DO NOT close the mixer - you're going to need it again.

Now the sound card is ready to go - but something has to tell it what to do with the selected signal. The Windows applet called Sound Recorder will do for starters, but it records to RAM. When it runs out of RAM (which usually doesn't take long), it's done. So you will probably want another program - such as Adaptec's Spin Doctor, one of the WAV editors or CDWAV (see the URLs page). Each package works differently from the others, but all are similar. Select the Record function, set mono/stereo, bit depth (usually 8 or 16) and the sampling rate (options are usually spelled out). If you are going to make a CD-DA, it's best to capture directly to redbook format: stereo, 16 bits and 44.1 KHz. If you record with other settings, you must convert with some program, which is slow and inconvenient - and very slow if conversion is done well. Most programs default to redbook; Spin Doctor only works that way.

You're not yet ready to record, though. Your recorder s/w has three choices: it can use only memory (as the applet does), it can use a TEMP file (either in the Windows default directory or one you select in the program), or it can write directly to your output file. Check that program for the way it works. Now, you have to set the recording level. If you overrecord, the result is painful even if it only lasts for a short time. The analogue signal is fed to the ADC (Analogue to Digital Converter) on the sound card. The ADC can only provide its maximum signal - all 1's. The result is that any excess signal is clipped, hard and brutally. Don't do it! If you record at lower level, you can leave it there or you can use a WAV editor or Spin Doctor to correct the level. Before you start recording, put your recording s/w and the mixer in non-overlapping windows on your screen. Find a passage in your source which is as loud as it gets. Activate the indicator of your s/w - sometimes it can be done without starting to record; sometimes, you will have to start some form of recording (which you will later throw away). Adjust the level on the mixer so that the level indicator on your s/w is always below maximum. If it's only 80% or 90% of maximum, that will be fine; those correspond to two and one db loss, respectively, which is not significant (and which will still give you much more signal-to-noise ratio than your system needs.)

Recording and processing

Now you're ready to record. Hit the record button (specify your capture file if necessary), then start your source. I strongly recommend that if you want more than one selection from a side of a tape or an LP, you record the whole side at once. You can always split and edit the selections later with a WAV editor; for splitting a long file, CDWAV can't be beat. These operations use a lot of space on your hard drive and may take a lot of time. Most editors will require at least twice as much space as the file you're editing. If you've captured a 30-minute side, you will need a spare 700 MB or so to hold redbook-format WAV files for editing. As you will discover quickly, lots of RAM and a fast CPU will help a lot, too.

When the file needs to be cleaned, still more time is needed. Removing clicks and hiss can be handled more or less automatically by programs such as CoolEdit and DART Pro, but that work takes a lot of processing. Even light treatment can take longer than playing the file. Different programs and different settings take varying time and give varying results. I use many different programs for my sources, depending on the initial quality and the importance of the recording. For example, I like the way that DART Pro removes clicks, the way CoolEdit handles hiss and noise, and GoldWave supports manual editing. I have spent hours correcting a single, four-minute cut. (No, I'm not rational about that, either.)

Along the way, some programs may complain about the headers produced by others. For safety's sake, I strongly suggest that you have StripWave (see the files page) available.

I can't tell you how to process your files - that's up to you and your source material. Anything you do to the signal will degrade it; if what you do is more important than what you lose, you come out ahead. The least damage is done by manual editing, but that is excruciatingly slow and painful. (Literally painful; an hour or two of editing will teach you more than you want to know about tense arm muscles.) The tools available today can be used with care to produce minimum loss and satisfying results. Misused, they can turn a favorite recording into mush.

Spin Doctor

In many ways, the easiest program for analogue recording is Spin Doctor. Select the source, select the target, set the cleaning options, and let it go. Still, there are some things you need to take into account to get the best results.

First, all the analogue sources are the same, regardless of the different lines from which you select. You still have to specify the source with the mixer. The different options give you different icons on the screen and may let Adaptec (some day) tune the cleaning operations to the kind of source being used. Next, if you select any of the options - cleaning or equalizing volume - the program will record the signal to the HD, then process it, then produce the output WAV file. It is designed to be allowed to do its 'thing' without interruption; let it have its way. I strongly recommend that you do any processing in a separate operation. First, record the tracks to HD without processing. Then do some cleaning, balancing or whatever by running Spin Doctor from one HD file to another. Finally, burn the finished WAVs to CD-R. Spin Doctor permits you to go from analogue source directly to CD-R, but that simple step often fails. The problem is that any tiny speed error, any hitch in the process will create a coaster. That's because the source is (by definition) running at 1x - real time. The recorder needs a steady stream of data and has a limited buffer. So if there is any difference between the speeds, or if you use an erasable blank (which must be written at 2x), you will have a coaster. It may work for you; if you want to try it, feel free. But safety and flexibility (opportunity to split tracks, edit, whatever) suggest that you go through your HD. Remember, you have to cue perfectly, cannot correct a mistake and cannot adjust levels (or anything else) except where it will be heard forever after.

E-mail me at cd-r@mrichter.com
Return to my CD-R home page

It doesn't sound good!

Okay, now that you've burned a Compact Disc Digital Audio (CDDA) - why doesn't it sound good? We have to begin with how the information is recorded to understand how it can go wrong.

There is a directory of sorts on each CDDA which identifies and locates the individual tracks. A track is located by its starting block and its length. There are about 330,000 blocks of 2K each on a standard blank; each one represents somewhat more than 10 milliseconds of sound. If you start a CD without specifying a track, it begins to play at the first byte of the first block and keeps on playing through to the last. There is no information on the track about the track - for that reason, it is not a 'file'. It's just a sequence of bytes.

If you write a track to the disc that is not an integer number of blocks, there are bytes to be written which don't come from your source (WAV) file. Most modern WAV editors will fix that by making sure that the number of bytes is a multiple of the block size (2K). If yours does not, there will be a click at the track transition for the random bytes that fill the last block. Solution: break the tracks on block boundary using appropriate software. (CDWAV from Mike Looijmans is linked from this site. It will split a long track into short ones on block boundaries.)

If you write a disc using Track at Once (TAO), the laser is turned off between tracks. When it is turned on, it writes a gap of two seconds before the start of the next track - that is dictated by the standard but there is software which allows you to violate the standard. You can also avoid that gap by recording in DAO - Disc at Once - where the laser burns continuously through the full set of tracks. Not all CD-R's support DAO, some do not support it well, and some which do support it are not implemented for DAO by some software. In general, if you have DAO problems, check the software from Goldenhawk; it does whatever can be done along those lines on any hardware yet manufactured. Or if you are really hard up, check the DAO page in this primer.

That covers getting a WAV file onto the CD-R in good shape. How about creating the WAV file or copying directly (on the fly) from a source disc? If you are having problems copying on the fly, first extract the files to your hard drive as WAVs and check them there. If the files sound bad, the problem is in your reader, not in your writer. (If you have problems with the last tracks of a CD, don't judge by extracting only the first ones.) You should look at the extracted files in a WAV editor to see if there are sharp spikes which you might not hear easily - they are the clicks and pops. In that case, please turn the 'page' in this primer and go to Snap, Crackle and Pop. Then you can get into the **really** complicated parts by reading about the complexity of WAV Files.

Skips and repeats

The CD writer needs a continuous supply of data. If something interrupts that flow for a significant time, the buffers can empty and underrun. If the interruption is shorter, a buffer - particularly that in the reader - may not cause an immediate underrun but may supply either zeroes or a repeat of the last information. In that case, you get a silence (zeroes) or a repeat lasting a fraction of a second.

Of course, the right way to fix such a problem is to ensure that the buffers do not empty. If you are writing on the fly from a reader, check the lights for regular operation. If you see a significant inactivity, then a burst you may have spun down (typically, between tracks) and if spinup is not fast enough the flow is interrupted. Another cause is a slightly damaged source disc which must be reread on the fly. If the problem arises when writing from the HD (very rare, indeed), then something is interrupting HD access and you should check for the usual villains: FastFind, anti-virus monitoring or another concurrent program.

If the problem occurs when writing on the fly and you have no remedy for the cause, extract the files to the HD and burn from there.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

DAO - SAO - TAO

Most of us are familiar with audio CD's and the tracks of music which they contain. When mastering a CD, those tracks may either be laid down separately, with the laser turning off between each pair; or in a continuous flow. If the laser writes **Track at Once** (TAO), a gap of two seconds is introduced (set by the Redbook standard) to ensure proper synchronization. If the tracks are written continuously in **Disc at Once** (DAO), no gap is required. There are schemes for approximating DAO by adjusting the gap in spite of the standard; the results may or may not be satisfactory depending on the player used and there is always a small gap with that approach.

DAO also has a role in data recording: masters for mass production are required to be in that format. Jerry Hartke, President of Media Sciences, Inc. http://www.mscience.com/ has provided the following on why pressing plants require DAO masters. Note that going through the 8-mm intermediate adds to the cost of mastering and that it will also (slightly) change the size of the disc being pressed.

> Mastering facilities must have perfect source data or their laser beam recorder will abort and ruin an expensive glass master. Track-at-once recording leaves "link-blocks" at the end of lead-in and at the beginning of lead-out. These are read as defects, or errors, by the LBR. Disk-at-once recording has no link blocks. Many mastering houses have learned from a rather gruesome history, and transfer information from a CD-R to 8-mm tape using methods that remove problems such as link blocks. Mastering is then conducted from the perfect image on the 8-mm tape, not from the "bad" CD-R.

Not all CD-Recorders are capable of DAO and not all software supports even those that can do it. The best information on DAO recording is provided by Jeff Arnold at http://www.goldenhawk.com/ If you believe that you will want DAO either for mastering pressed discs or for making CD-DA's without gaps, check there **before** you pick your recorder.

SAO - Session at Once

Session at once is a relatively new capability used on some CD Extra discs. With SAO, an audio session is written without intertrack gaps, just as though it were DAO. The difference is that only the **session** is closed, so one or more additional sessions can be written. Since an audio player can only see the first session of a disc, it makes no sense to write audio after SAO (unless you want to play the disc only on your computer).

Not all software supports SAO; Easy CD Creator 4.0 and above does, but earlier versions did not. Similarly, not all writers support this mode. Many which handle DAO, such as the Ricoh 62xx, are not compatible with SAO. Of course, that may be changed with firmware and new drives are almost certain to support SAO.

E-mail me at cd-r@mrichter.com

Return to Mike's home page

Snap, Crackle and Pop - noisy CD's

You have just 'backed up' your favorite CD to a CD-R. You go to play it and - it's full of noise. Snaps, pops and crackles all over the place. Must be bad software or a faulty blank, right?

Wrong. There's a chance that it comes from a dirty or noisy CD; if that's the case, clean it or change it. But the most probable cause is your hardware and its Digital Audio Extraction or DAE. There's a page on the subject of DAE here. This page is devoted to how to tell whether that's your problem and what to do if it is.

The telltale symptoms of faulty DAE are that the noise is in an extracted WAV file and that the amount of noise varies over the disc (usually worse as the track number goes up). There are three ways around the problem: read at alower speed; change the reader; or transfer through analogue. Note that a drive may extract digital audio at higher than rated speed, at rating, only at 1x or even at less than 1x, the speed at which it plays CDDA. And if your preferred reader does not do acceptable DAE (or doesn't do it at all), you should try your writer as a source.

Control of reader speed depends on the extraction software. Some programs, such as Plextor Manager, give you substantial control and let you select from 1x to the drive's maximum. Most do not and automatically select the highest possible speed. On many drives, that speed is too high for the sound to be free of noise. (Why is it that Plextor - all of whose readers except the 6x do DAE at maximum speed essentially without error - is the company that helps you use the lower speed you don't need on their drives. Unfortunately, the drives that need such a Manager can't be used with Plextor's and their manufacturers offer no equivalent.)

Replacing the reader is costly, but it is the best solution. No, you do not need to use a Plextor. There are many other good drives out there and you may be able to find one that costs less and is good enough for you. However, I have seen many posts that said: *I finally gave up and bought a Plextor; I should have done it months ago.* And in case you wondered: I get no kickback from any of the manufacturers I name. They've never offered, so I haven't even been tempted.

Finally, you can try recording a WAV with your sound card instead of DAE. Just make sure that you use the CD Player as the input by selecting it in the mixer and setting an appropriate level. (Alternative: run a cable from the player's headphone out to the sound card's line in.) If you have no appropriate s/w, you can try the Windows applet, but it will only record to memory so you may not get much music unless you have a lot of RAM on your system. Any of the WAV editors (links to several are on the URLs page) will do better for you. There are several reasons why you may be unhappy doing it this way. The most serious is that your are using digital-to-analogue conversion (DAC) in your reader and ADC (guess!) on your sound card. Each of those is imperfect and the artifacts that they create compound one another. A less important difficulty is that the sampling rate you get depends on your PC's clock, so the playing speed may be slightly higher or slightly lower than the original. However, because of the interaction of similar frequencies in non-linear processes (boy, he knows big words), resampling can produce artifacts such as subtle beats or recurrent noise. Still, it costs nothing to try this approach and if it works, don't fix it.

E-mail me at cd-r@mrichter.com

Return to Mike's home page

WAV Files

Let me begin by assuring you that computers are simple. The people who set standards and write programs for them are not.

A WAV file consists of three elements: a header, audio data, and a footer. The header is mandatory and contains the specifications for the file - information on interpreting the audio data - and optional material including copyright. The audio data are - um - audio data in the format specified by the header. The footer is optional and, if present, contains other annotation. I like to call the combination of header and footer the 'envelope' of the data. Usually, the data in a WAV file take the form of PCM bit streams. A simple one would be 8-bit samples taken 11,025 times per second from a single (monaural) channel. Then within the envelope would be a RAW file - the sequence of those 8-bit bytes in order and ready to be fed to a sound card's Digital to Analogue Converter (DAC) to play on your speakers. There is a well-defined way to put the other formats into the RAW file, which then goes into the envelope containing the information about how to make sense of the bytes which follow. What gets written to a CD-R is very much like the RAW file, with one important modification: the order of bits in each byte is reversed. So your CD-R authoring program can take a 44.1-KHz, 16-bit, stereo WAV file (i.e., in redbook format) out of its envelope, switch the order of bits, and write it to your disc.

Other formats can be stored in a WAV file. Apart from changing the PCM parameters, you can put a compressed stream into the envelope. Whatever is in the WAV file, your CD-R program must convert it to redbook in order to write CD-DA (Compact Disc, Digital Audio); and if the disc is not in CD-DA, it won't play on an ordinary audio CD machine. In many cases, converting from another format to CD-DA will require substantial processing. As a result, you may not be able to keep up with the writing speed of your CD-R. Since the program does not know how fast your computer is or how much processing will be needed, ECDC (at least) always converts a non-redbook WAV to redbook in a temporary file before writing it to your disc. ECDC also doesn't want to run out of disc space during that conversion, so it converts one file at a time, writes it to the CD-R, then converts the next. Writing one track at a time, it obviously cannot write DAO.

Now we get to the messy part. Not all programs follow all the rules all the time. As a result, the envelope may not be quite what it should be to make a valid WAV file. When that difference confuses the next program reading the file, it may refuse to open the WAV file or it may need to be told the format of the audio data, or it may just interpret some of the envelope information as audio data. For example, the 16-bit version of DART Pro appears to convert a valid header with annotation to an invalid one and does not recognize footers at all. If you feed it a WAV file from a program like GoldWave, which creates an envelope with valid annotation in the header and footer, it produces an output file which is almost unreadable. Fortunately, GoldWave will open it - when you tell it what format it has. Unfortunately, the envelope information DART Pro did not respect are still present - as data within the audio file. Those can be heard as initial and final clicks and can be seen in the first and last milliseconds of the file. If you snip those data out of the file, you can now Save As WAV and get a proper file again. Another option is to run StripWave. That program will strip the footer (it is unnecessary) before you start the offending program. Run it after that program has mishandled the header and it will strip out the excess audio - the initial click. NOTE: DART Pro is not the only offender and the 32-bit version appears to leave annotation where it is.

Another problem program is Plextor Manager. It also fails to put the right information into the header so that data appear in the audio stream. Again, StripWave will fix it easily - but you may not need to do so. The effect of the faulty envelope varies depending on the next program to see it. Some authoring programs are able to strip the faulty data the same way that StripWave does, so the simple error will not show up on your disc. Or, if it does, it may be lost when it is read from the disc. So once you have a bad envelope, you may or may not get an initial click or a final click. And you may get it on DAE but not when you listen to the disc, or hear it when you play the disc but not see it on DAE. If you do have the faulty file and process it before writing it to CD-R, the processing (WAV editing) program may refuse to open the file, may embed the data in the audio stream so that the click is guaranteed to show up later, or may correct the problem for you.

The bottom line is that you either set up a procedure which you know works for your software and your needs, or you figure out for yourself when and how you strip the faults from the WAV files. Note that StripWave is shareware and links to it and to other interesting tools are in the URLs at this site.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Block Boundaries

Now to zero crossings and splitting on block boundaries. Sit back - this is not a ten-word issue. And in all this, remember that what you are hearing is a bitstream, not distinct tracks. One more caveat: I know Sound Forge and many other WAV editors only by reputation. The ones I use regularly are GoldWave and CoolEdit Pro.

A track is a collection of sequential blocks of that bitstream and is designated in the TOC. Tracks must be split on block boundaries. Addressing in the TOC is in terms of blocks and you cannot split more finely than that. Each block on an audio CD is 1/75th second. For SMPTE, it's 1/30th, for other purposes, it's other sizes, but we're talking CD-DA, so it's 1/75th.

If the split occurs on a block boundary and if you burn DAO, then the sound is continuous across the split and there is no click or silence or other artifact - the bitstream is continuous and the sound is uninterrupted. If your split is not on the block boundary, then the block has to be filled with something.

By splitting at a zero crossing, the program presumably fills the rest of the block with zeroes for you. That's fine - if you want a silence which may be as long as (almost) 1/75 second. Sound Forge is right that you won't get a click, but you will get a momentary dropout, sort of an inverted click, if the zero crossing did not occur during an extended silence. The advantage of splitting at a zero crossing is not for writing to CD-R but for listening to the WAV. If you pick a high-amplitude signal to end a block (on the screen), then play it, you will hear a distinct noise at the end. The signal has dropped abruptly to zero. If you do the same thing at a zero crossing, you do not hear the click. Again, this is for listening to the WAV directly, not for recording in CD-DA where the block must be filled.

If you use a program which does not split on a block boundary, *something* must be used for the missing bytes and now there is no supply of zeroes for the job. I am told (I haven't verified it) that the program may pick up whatever debris is around from the stream of data from your source. It might even be the next few milliseconds of the audio - but in that case, they will be repeated. At best, it might fill with zeroes (see the issue above), but regardless you will not have continuous sound across the split. So the way to get a clean split for continuous play is to split the track at the block boundary. CDWAV and GoldWave force you to do that. In allowing for alternate frame rates, CoolEdit gives you the option to vary it. My understanding is that Sound Forge provides the option as well. And zero crossing has nothing to do with block (frame) boundaries.

E-mail me at cd-r@mrichter.com
Return to my CD-R home page

Cleaning Audio

The following is representative of the way that I clean up audio source files. It is intended simply as a guideline for those new to the process; many other programs and approaches can be used and different sorts of faults on the original call for different tools and techniques. Programs referred to in the text are generally available from links here.

Capturing

If the source is monaural, begin by recording it monaurally. If it has limited frequency range, do not sample at too high a rate. Many of the steps which follow are disproportionately difficult in redbook format (44.1 KHz, stereo); if you can work at 22.05 KHz monaural, life will be much easier. I usually record with CDWAV (linked here). It permits me to capture as I wish and to split the tracks from a tape or LP while saving them. I also use CoolEdit for the purpose since I like its level indicator or GoldWave for quick manual editing. I rarely use Spin Doctor for poor sources because it permits recording only in redbook.

If my source does require full fidelity and stereo, then Spin Doctor lets me pre-clean the file during capture. That is usually good enough to eliminate the need for denoising and declicking the whole recording, but manual intervention is likely still to be necessary for an LP source. Because of the nature of the work I will be doing, automatic splitting is rarely useful. In addition, if a single source tape or LP has several tracks, it is usually easier and better to do the automated denoising and declicking before splitting the tracks. In that way, the general sound will not vary from one track to the next.

First pass - Outliers

The audio should be captured at a low enough level so that overload is not a threat. In general, if you are going to clean the source, it starts out poor enough so that capturing a few db low will not harm the result after normalization. If the source has a serious problem - such as a scratch - I like to remove the extreme spikes by hand before running anything else. Without question, GoldWave is the best way to manipulate the waveform by hand. Where CoolEdit requires moving each sample or very brutal treatment (snipping or zeroing), GoldWave allows you to redraw the waveform with the mouse. Another advantage of GoldWave (especially Versions 4 and above, in beta as I prepare this) is that it will open virtually any file and provide you with access to edit any header and footer which sneaks into the audio stream. Those are the usual sources for clicks at the beginning or end of a track and various programs will introduce or eliminate them. Note that even StripWave will not necessarily handle footers if they prove to be a problem. However, I use Strip Wave when one program's output will not immediately load into another - as when a CDWAV file is to be read by DART Pro.

Ticks and pops

Step two is to remove the clicks, ticks and pops of modest amplitude. Having tried several programs for the purpose, I have settled on DART Pro as the most effective and easiest to use. In particular, the Test option in DART Pro 32 is consistent and very informative. The similar capability in CoolEdit is available only in some of its restoration tools and is much more awkward to use. Do not overclean your track! In general, if you try to eliminate all the transient noise, you will distort the music. Any automated denoising should be adjusted to leave the music at nearly full quality even if that means some noise is left behind. In particular, count on removing some leftover ticks after a noisy track is passed through DART Pro. Again, GoldWave is the way to smooth out those residuals.

Hiss, fixed tones and equalization

My choice for these operations is CoolEdit. If there is a persistent tone in the track, spectrum analysis will identify it and a notch filter will cut it down to size. But before beginning any track-level operation, I like to normalize it - again, in CoolEdit. Many tracks are normalized to 100% so that the peak level is just 0 db - maximum signal. However, I usually normalize a highly compressed track to only 80-90% so that its average level balances well with a fully normalized track that has a wide dynamic range. Similarly, I cut the level for a track which is quiet by its nature; I don't want a serenade and a march to play back at the same level.

After reducing fixed tones - whines, hums and the like - I listen to the track for frequency balance and for hiss. In general, I am willing to put up with more hiss than others prefer, but if I must cut it I use previews in CoolEdit to be sure I don't take out too much sound in the process. Then I judge the frequency balance of the result and decide on any general curve I want to apply. Again, I tend to change the signal as little as possible and I use the very flexible options in CoolEdit for that job. (GoldWave is similarly powerful and flexible.)

Preparation for recording

CoolEdit offers excellent tools for resampling, mixing and splitting tracks and similar manipulations. In general, I finish off the track in that program and use the Edit menu option to change the file parameters. For example, I may produce a redbook WAV for CD-DA; a 22KHz, stereo, 16-bit track for MP3; and a sample at 11 KHz, monaural and 8 bits for posting at my WWW site. GoldWave will reformat for me, but not as well and without all the flexibility of CoolEdit.

This process sounds tedious and for best results from poor sources, it is. However, it can also be quite rewarding. It may take an hour or more to extract a listenable result from a well-worn disc recorded in 1903, but it may then become a valuable resource not only for my pleasure but also for scholars and music lovers for generations to come.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Sound Balancing

Your perception of the loudness of an audio track depends on several factors. One is the absolute loudness, another is the dynamic range, a third is the kind of material.

The maximum signal you can record on a digital system is fixed by the number of bits available. If an analogue system is overdriven, the result is a gradual increase in distortion; digitally, saturation is a wall and even a slight step beyond it is highly disturbing. (There are, of course, some kinds of 'music' which are designed to be distorted; I have nothing to contribute for them.) So the maximum level you want to set is something which does not go beyond maximum loudness.

The second factor is dynamic range - the range from the loudest to the quietest passage in the program. At a symphonic concert, that can be as high as about 100 db - for a very quiet audience and some very loud passages. CD recording at 16 bits offers 96 db range. A very good tape deck can deliver about 60 db. But in many environments, use of wide dynamic range is undesirable, so the actual dynamic range on a recording may be compressed to 40 db or less. Different recordings will vary in the amount of compression used - and the average sound level depends more on that than on the absolute maximum you have to keep to 100% (16 bits).

The third factor is the program content. I don't know about you, but I want a march to sound louder than a lullaby. So even after you set the maximum and mean levels by normalizing and compressing, you have to consider how loud you want the result to be.

All is not as bad as this sounds. In general, a given label will be pretty consistent across its recordings of a single group or even of a single kind of music. But as you try to mix a greater variety, you face the problem that the software doesn't know what compression makes sense (or how best to implement it) - and it certainly does not determine how loud **you** want a given track to seem on a given compilation. The automatic level setters do what they can, but they are basically limited to setting the maximum level. The rest must be done to your taste by you. A good WAV editor, such as CoolEdit Pro will let you shape the compression on a track-by-track basis as well as permitting you to adjust level and keep from hitting saturation. It will also denoise, alter channel and frequency balance, and do many wonders - with a lot of time and effort on your part. To avoid that, choose the tracks for your compilation with some care and take what you get with automatic balancing.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Clicks - a Recap

This section is essentially redundant with information in three other parts of this primer: Digital Audio Extraction (DAE), Disc At Once (DAO), Snap, Crackle and Pop and It Doesn't Sound Good. The purpose is not to beat the message into you, but to indicate the **process** by which a problem is analyzed.

Clicks throughout the tracks

These noises usually arise on tracks extracted digitally (ripped) from a CD-DA. They come from faulty DAE or occasionally from a damaged source disc. While a better reader is the right answer, other choices include reading from your writer, slowing down DAE, and recording through an analogue connection. You can hear such clicks on a WAV file extracted to your hard drive. Note that the low signals generated by CD-R compared with pressed discs make some sources of clicks more likely; as a result, a setup you find acceptable with commercial originals may be too noisy for you if you are ripping from a CD-R.

Click at the beginning of a track

The source here is usually data being read as audio. There are different ways that the WAV envelope can be written and some software assumes that some of those options are not in use. In that case, they pick up some bytes of data as though they were audio and encode them into the CD-DA. If you look at the first milliseconds of the WAV (from whatever source) in one or two WAV editors (linked from my page of URLs) you can see the noise as a spike; some editors will give you a strange error message in this case because they think that the WAV file is a raw format. In that case, they may or may not show the glitch. If you get that message, that editor will save the file without the click. In any event, some audio players should reproduce that click if you listen carefully to a WAV file.

Click at the end of a track

A track must begin at a block boundary, where each block is 1/75th of a second. If the last block of a track is incomplete, it will be filled in by something else. Depending on the software, it may be filled with zeroes or with whatever bytes happen to lie next on the disc from which you are writing. In the latter case, you are likely to get a click. A click from this cause will **not** be audible on the WAV file you are going to write since the block limitation is encountered only when writing to the CD-R. Most good tools for splitting tracks enforce the block boundary, but the more powerful ones (such as CoolEdit and Sound Forge) let you choose a size different from the nominal. If you do not use 1/75th second for the sector size, you can produce this problem with even the best software.

A second cause is similar to that of a click at the beginning. The WAV envelope around an audio track may include a footer as well as the header and that footer may be read as audio data. If so, you may again be able to hear the click in some WAV players.

The third source for a click at the end of a track comes from a click at the beginning: if some bytes of header are read as audio and if the audio data started as an integral number of blocks, then the last block being written has only as much audio as got pushed out of the first block by the unwanted data. The result may be a click or may be silence; you can identify it easily in a WAV editor by the fact that it usually lasts for 100 milliseconds or more where other sources are usually much less than a millisecond.

Click between tracks

This is an oddity that occurs only when you record a continuous program without using DAO. Some software allows you to violate the spec requirement of two seconds between TAO tracks. You may even be able to run it down to what appears to be zero - but it isn't. In the short intertrack gap which is produced, the signal is zero. If the continuous program had sound at the track split, the brief, sudden silence may be audible and may seem to be a click. If you tried a 'zero' intertrack gap, this is a likely cause and can be fixed only by writing in true DAO.

Analyzing your problem

So you get clicks - and want to know what to do about them. The first step is to listen to them to find out where they occur. The second is to use information such as that above to identify the cause. If that fails, ask in the Adaptec mailing list or the newsgroups for assistance - but be sure to report what you learned from your own analysis.

Once you find the cause, what do you do about it? The best choice is to eliminate the cause, not the effect. If your mastering software does not like the header from your WAV editor, change one of the programs until you have a compatible pair; if your reader doesn't do good DAE, use a different reader or a different speed. Another choice is to clean up the results. Many of the sources can be fixed with a program like StripWave, which eliminates most problems with the envelope. You can use a WAV editor to remove those clicks manually. You can do generalized cleaning in powerful WAV editors or standalone programs such as WAVClean, DART Pro and DCART. Spin Doctor (part of Easy CD Creator) has a simplified cleaning capability as well.

Are there other causes? Sure, but I don't know enough about them to write them up here. Fortunately, they are also uncommon. Frankly, the specifics of dealing with clicks is less important to me here than exposing the process used to track down errors. If you post - to tech support, the mailing list, the newsgroups or by e-mail - that you have clicks on your tracks, you haven't said enough for anyone to help. The same is true when you report that you can't burn a data disc. You must do your homework before you can get any help. Unless you bring in a personal consultant, you're the only one who can isolate the cause - or at least reduce the possible causes to one or a few of the many potential sources.

E-mail me at cd-r@mrichter.com

Return to Mike's home page

MP3 to CD-DA what's going wrong?

This note is a synthesis of some basics and some material elsewhere in this primer to see what may go wrong when you make a conventional audio disc from MP3s. There's nothing new here, but putting it together may be worthwhile. We will walk through the whole process even though you may only deal with a small portion; imperfections can come from others' contributions. Some other starting notes: different programs deal with deviations in different ways so something that works when you listen with WinAmp may fail when you burn with MusicMatch - or vice versa.

Let's say we start with a cassette of a concert. We digitize it first at the redbook standard of 44.1 Ksps, 16 bits, stereo, uncompressed PCM. Then we split it into tracks and encode it to MP3. (Some programs hide the PCM, some write it out as a WAV, but all which allow splitting or editing go to PCM along the way. If you doubt that, watch how much storage is used while you're recording.) Now you take those MP3s, make an audio layout and burn the disc in DAO so there are no gaps.

But there are. And maybe there are noises in them. And some of those files were not accepted in the layout. Some were accepted, but turn out to be only four seconds long. Others have glitches or bits chopped off the end. What's going on?

We avoided one error: sampling at the wrong rate. Still, when you get a file from the Internet or from a friend, it might have been created at a different sampling rate. If it came from a DAT, it may be at 48 Ksps. Regardless, some programs demand 44.1 Ksps stereo at 16 bits while others are more tolerant, but none will buy 48 Ksps (or 16 or 32) because they cannot resample on the fly. As with other operations which come up later, they might be able to do a competent job for writing at 1x on the fly, but they don't know how fast you'll want to write or how much CPU power will be available, so they avoid the issue by making you do the resampling. But we were smart; we bypassed that problem by recording right in the beginning.

When we split the PCM/WAV file, where did we do it? With most programs - CDWAV, GoldWave, CoolEdit - you split on block boundaries. Other programs are more flexible; with CoolEdit Pro, for example, you can redefine a block (frame) from its nominal audio value of 1/75th of a second to handle film and other rates. Some simply cut where you tell them to, ignoring blocks. If the track ultimately going to the CD does not end on a block boundary, the software will fill it in so it does. Needless to say, it doesn't fill it with music - just silence.

There's another trap when that PCM is made into a WAV. That is adding non-audio data. Like MP3, WAV can carry additional information. However, when you put that in for other purposes, your later programs may not recognize it as a chunk of data and make it into audio. That will not only mean the wrong length of track (not ending on a block boundary), but will also make a very noticable click.

Now we compress the tracks to MP3 and get ready to use them. Along the way, we added the information we'll want in our MP3 player - artist, selection, that sort of thing. Out comes just the thing - a 128-Kbps file that sounds close enough to the tape to suit our needs. Well - maybe. First, it's not really **at** 128 Kbps. When you told the program to encode for that rate, it set some parameters so that it would get close, then charged ahead. Depending on your encoder, you'll be off by a little. That doesn't matter in most ways and not at all when you listen, but it does mean that the size of the file is not exactly 128 Kbits per second (plus overehad). That becomes more significant still if you encoded with VBR - Variable Bit Rate - which squeezes a little more quality from the compressor.

Now let's make that layout and let's include this introduction your friend sent over. Oops, wrong sample rate, have to resample in an editor. Okay, now why does this extra track seem to be only four seconds long? Oh, it was encoded with VBR which our mastering software won't convert - back to the audio editor or a separate decoder to make it a PCM (WAV) for the layout. We start the burn and the program stops on another file; back to the editor to find the data dropout (just a bit or two - what difference does it make? I can't even hear it) and snip it out. Now it burns and what am I hearing instead of a nice continuous concert? Gaps where the MP3s decoded to incomplete blocks thanks to the losses in compression, bit rate deviation and the like. Clicks from misplaced ID3 information (courtesy of some compressors) or from having silence where sound ought to be. (Huh? Howcum? Well, suppose you have a steady tone and introduce a very short silence. You hear it turn off, then on again - a click.)

One more 'goodie' is that some tracks may be cut off and others may have noticeable silence added. That comes from another "feature" of some encoders: poor reporting of the playing time. Remember, the mastering program cannot play the file to determine how long it really is and the compression is only approximate, so the only way the program knows that this track is 03:24:12 (minutes : seconds : frames) is by reading it from the header. The program has to believe the header whether it wants to or not.

If you've gotten this far, I trust that you will understand why **you** are the only one who can determine why the disc doesn't sound good and pin down the actual source(s) of your trouble instead of blaming your writer, your mastering software, or any other single component of the process. You should also have a good idea of some of the inaudible differences among mastering and encoding programs.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Archiving Audio

First, let's be clear: this is about saving audio files **you** have worked on. Where and how you archive a copy of that CD you bought on eBay is a whole other subject. This is about that surreptitious concert tape you spent half a day rebalancing and de-hissing; the rare LP you spent hours de-clicking and selectively compressing. Now you have a batch of WAV files and an audio CD, maybe a spare CD just in case. That's not necessarily good enough, though; what are the options for making a copy that will last and will give you the same quality CD-DA next year that you burned today?

Saving as CD-DA

Hey, you've got that spare audio (CD-DA) disc; you can just run a copy from that. Right?

You can, but that may not be your best choice. The potential problem is that there may be slight errors on that disc, noise you do not hear when you listen to it but that will show up on a copy. Even if there are none now, they may develop over time or with handling and they may show up when you upgrade to a drive that doesn't like the medium you used. The key here is that when you save as audio, you gain about 13% in capacity but pay for that by dropping a layer of error correction. The odds are that you'll be okay, but you can hedge your bet with one of the other options.

Saving the WAVs

Okay, if you save the WAV files, you get back that blessed layer of error correction and gain extra confidence. Unfortunately, you lose 13% of capacity, so that if you started with a 74-minute disc, you need to store about 740 MB and ECC or not, it won't fit onto one disc. You could write it to two discs, which makes re-creating the CD-DA tedious; you could use a "99-minute" blank, but that will run the risk of failure 'way up. So if your home-made disc is less than about 65 minutes, you're home free, but otherwise you have your choice of poor options if you want to save as WAV.

Saving an image

It turns out that this has the same size problem as saving WAVs with little advantage. For example, if you decided you wanted to touch up a track, you would have to extract all the tracks to WAV, modify one, then make your new CD-DA and a new image to save. Extracting those tracks can mean burning a fresh CD-DA and running DAE or running a program such as CD-R Diagnostic or ISOBuster. The one advantage of archiving as an image is that it lets you make additional CD-DAs very easy: just double-click the saved image to fire up the appropriate mastering program, ready to burn.

Saving with lossy compression

Compressing to MP3 or one of the other formats using perceptual encoding can give you back that 13% - and a lot more. Many people are happy with redbook audio compressed to 128 Kbps - a factor of eleven or so. Some want to cut their losses and use 256 or even 384 Kbps. Even so, there are losses and at the least, they may leave you wondering whether you've thrown away something you will want some day. It's also worth noting that high-quality compression, such as high-rate with LAME, are far from speedy.

Saving with lossless compression

"Lossless" compression? What's that about?

We know that the usual compressors - ZIP, RAR and the like - do little or nothing to save space in a WAV file. But there are compressors for audio which cost neither quality nor money. The one I've run is Monkey's Audio, which is quick, simple and provably lossless. That is, you can take your favorite WAV file, compress it to APE, then decompress it and you will have a bit-for-bit match to the original. Because lossless compression throws nothing away, it does not shrink files as much as MP3 does, but it will save more than enough space to put even an 80-minute CD-DA onto a 650-MB disc. The one drawback is that you must decompress back to WAV to use the file. If one of the schemes becomes popular, perhaps the CD-mastering programs will accept it for direct writing.

The medium that holds the message

The time you spent creating this audio masterpiece is worth a fair amount to you; don't sacrifice it for a few pennies of medium. Testing labs which have the equipment and no bias will tell you that the longer the blank, the higher the error rate. That doesn't mean that you will hear the difference - even if CD-DA loses one layer of error correction, there are others beneath it. What it does mean is that a 90 or even an 80 is less reliable than a 74 and that you run a higher risk of having an error when you need that file. This is also a case where you may want to invest the extra dime or dollar in a Taiyo-Yuden or Mitsui disc - the best you have found for your writer - just to be sure.

Links to the programs in this page are here at this site.

E-mail me at opera@mrichter.com
Return to Mike's home page

Correcting Pitch

In most cases, the pitch of your source is correct - that is, the recording is at the pitch and has the duration intended by the performers. However, the farther back the original was made, the more likely that it is now at the wrong pitch regardless of the form in which you find it. In the first three decades of sound recording, a '78' would be recorded (usually) somewhere between 60 and 95 rpm! Later, errors became less frequent and less extreme, but they still occurred. The best way to solve the problem for transfer to your computer (and to your CD-R) is to change the pitch at the source - with a variable-speed turntable or tape player. There are also variable-speed CD players, but they offer control only at their analogue outputs; whether you would prefer to record from a corrected analogue signal or from off-pitch DAE is up to you. If you are using a variable-speed player, then, you simply adjust it until the pitch is right to you and record through the line input of your sound card. So from here on I assume you're starting from a fixed-speed source and you have to correct pitch digitally. For convenience, all sampling rates below are in KHz.

Determining pitch

Well, this should be easy, right? Maybe so, maybe not. It depends on your source and on your ear. First, we assume that whatever pitch error exists, it is constant through the recording. That was not strictly true in the earliest days, but correction for changes during the recording is beyond the scope of this primer (i.e., I don't know how I would do it if I had to).

Step one is to record a short selection in which the pitch can easily be determined. I recommend doing that in monaural at a reduced sample rate, perhaps 22.05. It makes handling the file easier and should serve your purposes as well as working with a file four times the size. That file is loaded into your favorite WAV editor. For this purpose, I find GoldWave the best choice because its speed controls are the most comfortable for me. All you need to do in GoldWave is to adjust the playback speed until the pitch on your test file checks with your pitchpipe or other reference. Now note the error - the difference between the playback speed and that at which you recorded. For reference, a semitone is about 6% in speed; half an octave is about 40%.

Resampling

One way to correct pitch is to record at the nominal sampling rate you will be using, say 44.1, and then to resample to correct. For example, if your 22.05 test played back best at a rate of 23.3 KHz your WAV is about a semitone flat. In GoldWave, you can simply record the whole selection at 44.1 sampling, then transpose it down a semitone. The same thing is done slightly differently in other editors, but the effect is the same: approximately six samples are thrown away out of every hundred and the others are adjusted to fit. GoldWave gives you no control over how that adjustment is done and uses a simple, quick algorithm for the purpose. CoolEdit Pro and other WAV editors provide control in the form of pre- and post-filtering and relatively complex interpolation schemes. Since that interpolation is done 44,100 times per second per channel in this example, even a fast CPU will take substantial time to do a good job. Only you can determine how much time you want to spend processing and how important the errors are.

Off-Sampling

(Note: this term is my invention. If you have a better one, please let me know.)

Most audio capture programs allow you to pick from a few sampling rates. Typically, they are 48, 44.1, 32, 22.05 and lower. However, Mike Looijmans' CDWAV is not so choosy. In addition to its presets, you may type in any integer sample rate you would like. If the playback is best at 6% higher speed than recording, then you can record at a sampling rate 6% lower than your target and if you have a way to interpret the result as 44.1, you would be on pitch. With the right tools, that's straightforward.

Record in CDWAV at your shifted sample rate - here 41.5. Record the selection as a .PCM file. PCM is a raw format (other extensions are RAW and SND). Now, open that file as PCM in CoolEdit and lie to the program - tell it that the rate is 44.1. In this case, lying is good for you. All you need to do now is to clip off the click in the header (a real PCM doesn't have a header, but CDWAV recorded it with the WAV envelope, which has to go). Save the file as WAV - and you're done. No resampling, so no approximations or delays.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Sound Cards

To tell the truth, this subject is peripheral to CD recording and I've been reluctant to add it. But questions come up so often that it seems to be necessary - and for many, the performance of the sound card in the PC is essential to getting good results on CD-R.

First, I shall neglect some of the factors important to users of computer audio for other purposes. The primary interest for CD-R is the ability of the sound circuitry to encode PCM - Pulse Code Modulation - from analogue. There is a special case in which the sound card serves a purely digital function - S/PDIF; it will be dealt with at the end. In addition, the focus here is on CD-DA, Compact Disc - Digital Audio.

Pulse Code Modulation

In PCM, a collection of computer bytes corresponds to a single, specific point on an analogue waveform. The frequency with which those samples is taken is the sample rate, usually designated in samples per second but written (imprecisely) as KHz (KiloHertz). Of course, one issue is whether there are one or two channels of audio involved - monaural or stereo sound. Another is how many bits are used for each sample collected; the usual values are 8 or 16. If the encoding is set up to collect 8,192 samples per second (8 KHz) from a monaural signal and to encode at 8 bits per sample, the resulting PCM stream takes 8 K bytes per second (bps). If samples are collected to be recorded in redbook format at 44.1 KHz, stereo and 16 bits, the PCM stream takes 176.4 Kbps. That rate is labelled 1x for audio. (Because of error correction, the corresponding data rate is 150 Kbps.)

Noise Floor

One key function of the sound card is to convert the analogue input signal it receives into digital data. The quality of the result depends essentially on two factors: the noise floor and the accuracy of ADC. Noise floor means the noise introduced by the card itself and the leads which provide input when there is no signal. It determines a level of hiss due to the card as opposed to what may be received from a tape or other input. For comparison, a very good cassette tape can deliver about 60 db SNR - Signal to Noise Ratio. That means that its noise floor is 60 db below its maximum signal. Because of quantization (the resolution capability of a digital signal) the SNR for 16 bits such as from a CD is about 96 db. A sound card which puts its quality into gaming and MIDI may deliver something like 50 db - in other words, it may provide more noise than a good, Dolby B cassette. A less expensive sound card which focusses on WAV quality may deliver 75 db. Where those cards typically cost less than $100, for $200-500 or so, one can buy still higher quality, up to about 82 db. Beyond that, the experts use outboard converters to avoid the electrical noise inside the computer's case.

ADC - Analogue to Digital Conversion

The input to a sound card is typically an electrical analogue of a sound wave. The card's job is to convert that signal to a digital one of the proper number of bits and at the proper rate. The conversion process is more or less accurate depending on how well the card is designed and built. Not surprisingly, the cards with low noise floor also have high-quality ADC. High-end cards may encode to 20, 24 or more bits. That allows substantial editing without losing quality before the bit depth is reduced to 16 for writing to CD-R. In general, mastering s/w will create redbook audio from 8- or 16-bit samples, but not from others.

Sample Rate

Similar to but worse than bit depth, sample rate is under your control and must be chosen with care. The highest frequency which can be captured in digital form is half the sample rate: 44.1 K samples per second cannot record more than 22.05 KHz audio. However, there are significant effects at lower frequencies, so it is desirable to work at a higher sampling rate when practical. Most DAT recording is done at 48 KHz; 16 and 32 KHz are common choices for lower-fidelity sources; professional mixing is often done at 96 KHz. However, those rates are not easy to convert to 44.1 for CD-R, so they will require processing outside the mastering software before they can be burned to CD-DA. Like adjusting for non-standard bit depth, that is a job for a WAV editor. Because the file size and the difficulty of editing increase with sample rate, processing low-fi sources at 22.05 KHz monaural is much more efficient than the same process in redbook - 44.1 KHz stereo. Fortunately, most mastering s/w will convert that to redbook without bothering you (though not on the fly).

File Formats

Rather than repeat the information elsewhere in this primer, I will simply summarize it. A PCM file may be raw or wrapped in an envelope with a specific format name. Raw files typically have extensions PCM, RAW or SND. A raw file carries no information except the stream of bytes. If you put data on that stream to specify details such as sample rate, bit depth and mono/stereo, then you must use an envelope or wrapper such as WAV, AU or AIFF. That envelope permits other information (e.g., copyright) but does not require it. Considering only the WAV file (it's all I know), the wrapper can also hold compressed files, but those are **not** PCM. ADPCM and MPEG Layer 3 are typical compressed formats.

Inputs and Outputs

For convenience and for mixing, the sound card will accept several different inputs. Typically, one comes from CD-ROM, one from Line In (an external connector) and one from Microphone (usually monaural). There may be additional CD-ROM connectors on the card, but unless there are separate sliders for them on the mixer, they probably are all shorted together and simply provide different types of plugs. You should **not** connect two signal sources to a single input of your sound card. It probably won't sound good, but it definitely will not load the sources correctly. If you want to connect both your writer and your reader to a single card, you can find a card with distinct inputs (separate sliders), install a switch for the two inputs or use a cable from an analogue output such as headphone to Line In.

In order to hear a signal from your sound card, you must enable it on your mixer. However, what you hear is not necessarily what you will record. In order to record that signal, you must enable it (and set the record level) on that same mixer. On the standard Windows mixer, you go to Options, Properties, Recording in order to reach the controls you need. The next time you boot, your settings are lost and you must go there again! Other mixers make the controls more accessible and sticky, but they are not usually available except with their corresponding sound cards. An Ensoniq mixer is available only with a Creative Ensoniq card, for example.

S/PDIF designates a direct digital connection between a device such as a MiniDisc player or a CD-ROM and your computer. It is usually implemented in a high-end sound card. Note that not all outputs of

CD-ROM drives which **say** digital or look digital will mate correctly with an S/PDIF card. Like DAE, S/PDIF avoids DAC and ADC altogether and allows you to remain in the digital domain. In addition, many sound cards implementing S/PDIF also convert from 48 to 44.1 K samples per second on command. Another advantage is that an S/PDIF input allows you to connect that second CD-ROM safely. But by now, we have gone beyond the scope of a primer. (And anyone who suggests that that means it has exhausted everything I know on the subject has been cheating!)

Recommendations

As usual, I make no comments on equipment I do not know. I have replaced three sound cards since getting into CD-R: a Creative AWE 32, an AWE 64 and a Yamaha. (The last meant replacing the motherboard on which it was located.) In all cases, I found both the noise floor and the quality of ADC insufficient; the Yamaha also proved almost impossible to control with the standard mixer. My current systems use an Ensoniq PCIAudio (no longer available) and its first cousin, the Creative Ensoniq PCIAudio. The cards and the mixers are different (and not interchangable), but both deliver high quality at low cost. I have not tried the Creative Live, a Turtle Beach, Card D, or any of the dozens or hundreds of alternatives.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Audio Losses - pressed vs. CD-R

At first glance, there should be no reason for loss of fidelity between an original disc and a CD-R copy. In practice, there are often reports of such degradation. For example, Philips reports at their WWW site in cddd3610e.pdf that you can expect the following differences at the analogue line output, with similar values for headphone.

Line Out	On pressed CD	On recordable CD
Amplitude Linearity	1.5 dB (20 Hz - 20 kHz)	2.5 dB (20 Hz - 16 kHz)
S/N-ratio	81 dB (84 dB A-wtg)	80 dB (82 dB A-wtg)
Total Harmonic Distortion + Noise	65 dB (1kHz)	55 dB (1kHz)
Channel separation min.	70 dB (20 kHz) min.	65 dB (16 kHz)

They also state - emphatically

THE SOMEWHAT REDUCED AUDIO QUALITY WHEN PLAYING BACK AUDIO TRACKS ON CD-R DISCS HAS NO RELATION TO THE DIGITAL QUALITY OF THE AUDIO TRACKS AS THEY HAVE BEEN RECORDED ONTO THE CD-R DISC.

Those numbers are pretty good, but why don't they match? Why is CD-R playback inferior to that from a pressed disc? I have no proof, but offer the following guess. If someone has information about it, please let me know.

The signal read from a CD-R is inferior to that from a pressed disc. Either the maximum brightness is less or the minimum blackness is brighter - or both. As we know, a layer of error correction is saved by using Mode 2 Form 2 for CD-DA. So we have a weaker signal and less correction, hence more errors. That means that the circuitry on the analogue side - the part that feeds line out and headphones - will be making more corrections. Those corrections will have exactly the kind of impact in the table. My guess is that Philips recognizes the losses and adjusts the circuitry so that it does not attempt to pass an inferior signal.

Regardless of why or how it happens, it's clear that Philips acknowledges that pressed discs deliver better performance on the analogue output - sound better - than recordables. Even though other manufacturers may not be explicit about it, you can bet that they, too, cannot deliver the same performance from recordable and pressed discs. Does that mean that the digital signal is better and that you will get better response from SPDIF or from DAE? Probably not; the cause is still there - errors in the read signal. The effects should be the same in the sense that there will be uncorrected extraction errors. It seems likely that some drives will have fewer than others, just as some do DAE faster or with fewer audible errors than others.

E-mail me at cd-r@mrichter.com

Return to Mike's home page

The CDDB and Protected Audio Discs

IANAL - I am not a lawyer - so please take any legal implications below to a real attorney before you act on them. Similarly, I am not a cop; I have no concern whether you are pushing on - or across - the limits of copyright. Finally, my interests are in the sort of music to which none of this applies so I am reporting inferences from what I've read far more than my own experience.

The Compact Disc Data Base (CDDB) is a database of information about Compact Discs. I trust that the rest of what you read here will not be **quite** as simplistic as that. The CDDB contains timing information about commercial audio CDs. It uses a well-defined format to store the length of each track and information about the track (title, artist, etc.) and about the disc on which it is published. It is used by having a program describe a disc to the database in terms of the duration of its tracks. The database then reports the disc and track information back to the program that asked. The duration of a track is measured in minutes, seconds and frames - seventy-fifths of a second. Looking only at the values for seconds and frames, there is pretty nearly uniform (random) distribution over 4500 possible values. If there are fifteen tracks, then there are 4500^15 patterns of values. As long as no one has set out to spoof the system, two discs with the same pattern can be relied on to have the same contents.

An obvious question is: where is that database we are referencing? As long as it has the right structure and contents, it can be anywhere - and in fact it is distributed at Internet sites and (in fragments) on personal computers. Until recently, there was a master copy referenced freely over the Internet. At this time (July 2001) there is dispute over who owns the database, who owns its information, who can access it, who can duplicate it and on and on. It seems inevitable that litigation and disputation will continue longer than this page will last, so I have to leave you to find the CDDB you want to use for your applications.

Now we get to the question of protecting audio discs. The first issue is: what is being protected? Essentially, the publishers are trying to prevent digital copies; whatever the legal situation, they tolerate analogue copies. Of course, an analogue copy goes through DAC (Digital Analogue Conversion) followed by ADC (Analogue Digital Conversion - in case you hadn't figured it out), losing quality in each step. Whatever way the publisher mungs up digital extraction to prevent piracy, he cannot keep you from making an analogue copy or the disc simply wouldn't play. You can therefore get a good - good enough? - copy of a protected disc by playing it and capturing its output with a full-duplex sound card or a program such as TotalRecorder (linked from this site).

Now let's put the pieces together. If you make an analogue copy, count on it - your disc will not be recognized confidently by the CDDB. Your capture of the duration of a track simply won't match that of the protected source. That will give you the job of entering track information, but the copy will be made and will play. Unless you're selling the result, the publishers won't call in the cops.

E-mail me at cd-r@mrichter.com

Return to Mike's home page

A Note on Video

The two applications which drive many aspects of personal computing are video and games - and they are obviously closely related. With the availability of inexpensive capture devices such as IOmega's Buz, many are jumping into video and becoming frustrated in the process. This note is intended to address the beginning user of digital video.

Like audio, most video begins with an analogue signal. If your camera generates a digital signal directly, most of this page does not apply. For the rest of us, some capture device is required which converts the analogue information to digital as your sound card does with audio. In general, the capture is to a lightly compressed format. (Without compression, you would have to record at around 30 MB/sec. That would both load your system very heavily and rapidly use up all available space on a hard drive.) The lightly compressed format is usually compressed further offline with a codec (encoder/decoder) that limits the quality of the resulting file. One popular format is whitebook MPEG, which is used to generate VCD. In that form, the data rate is about 1x in CD-ROM terms, meaning that the signal has been compressed 200:1. In that format, a CD-R can hold about an hour of video. At 50:1 compression, a CD-ROM can hold only about 15 minutes of video. Obviously, the more compression used, the lower the resulting quality of both video and audio.

In order to play back a video, the playing computer must have the same codec installed as was used to create it. Many such are available and whenever you install one to encode, you will be able to play it back. However, the person to whom you send it may not have that codec - and in that case will need to find it and to install it before she can watch your product. Intel has several Indeo codecs and there are many others with modest advantages from one to the next. Those codecs usually generate AVI files on a PC (MOV is more common on Mac). AVI and MOV are different envelopes wrapped around the same raw data stream - as WAV and AU are envelopes for audio data. Because of the variation in AVI codecs, it may be most convenient to compress your video to MPEG, which is consistent across platforms and has a standard codec. You can produce MPEG with other products, but many find the Xing encoder most convenient and reasonably priced.

A key factor is the size of the file you generate. Since your original recording will be an AVI even if you are going to compress it to MPEG, you are limited by the specification of AVI. The definition of that format limits it to 2.1 GB total size. While that seems immense, in video terms it is quite limiting. If your original capture is compressed 50:1 (which is very high), an individual capture must be less than 45 minutes. Compression to MPEG is similar to compression to MP3 in audio - and similarly makes editing in that format almost impossible. In order to edit your video, you will need to operate with an AVI and use a tool such as Adobe Premiere or ULead MediaStudio. Editing is very demanding on your system, so be prepared to add large, fast HD's, plenty of RAM and powerful processors if you want to deal with ten minutes' worth at one time. However, it is important to recognize that a high-performance capture card is of little value for home video. If you have a very high quality digital signal, you may throw away 90% of it in compression. If you start with a half-frame source, you will 'only' throw away 60%. The results will be indistinguishable. (Obviously, the rules and the systems are very different for broadcast quality.)

No one can tell you what picture quality is good enough for your purposes. But once you know what you want, do not overbuy to achieve it. High quality in capture means major system revision; modest quality will let you know how much you are willing to invest and give you satisfying results.

E-mail me at cd-r@mrichter.com

Return to Mike's home page

MPEG - Picture and Sound

There is a set of standards for video labelled MPEG. Some apply to video; others apply to audio. This note will cover only the basic features; for details see the usual sites as linked under URLs. Like all lossy compression schemes, the quality of the output of MPEG encoding is less than that of the original video or audio source. In general, the user has enough control to select a balance in compression between loss of quality and reduction of file size, but one should not hope for miracles. Experimentation is needed to determine what looks and sounds right for you.

Video

MPEG video was to be available in three flavors. MPEG 1 can be produced and played back (encoded and decoded) in software alone and on a fast system can provide reasonable quality at 352x240 pixels or about half of the usual minimum screen size. MPEG 2 is more efficient, supports higher resolution and requires specialized (and costly) hardware to encode and moderately priced hardware to decode; it is currently used for DVD playback. MPEG 3 is still higher quality and was intended for high-definition television (HDTV), but it has been suspended until requirements can be fully defined.

MPEG 1 is readily produced on a PC and is suitable for recording from and to VHS videotape. 'Readily' must be taken with a large grain of salt, however, since software encoding takes many times real-time. The MPEG codec is quite efficient in storage requirements and provides a high-quality picture at reasonable file sizes by video standards. A properly encoded MPEG 1 file can be turned into VCD (Video CD) with Easy CD Creator and played on a suitable set-top player. More useful for many is that it can be played on a personal computer and even accessed by a WWW browser with a suitable helper application. Many other codecs (coder/decoder) are available for video; their products are usually in the form of avi or mov files depending on whether they are packaged in Windows AVI or Mac/Windows QuickTime envelopes.

Audio

MPEG Layer 1 is a moderately compressed audio format which is used in some video to reduce decompression load. Layer 2 is usefully compressed and can be generated easily in software encoders. The usual extension for MPEG Layer 2 is .mp2. Layer 3 is still more highly compressed; the shareware program l3enc (with its companion l3dec) appears to be the most popular choice for that process. Other mp3 encoders are available or on their way.

MP2 and MP3 playback require an appropriate application; usually both are provided in a single program. One with several advantages is the inexpensive shareware WinAmp (and the companion MacAmp) from Nullsoft (http://www.nullsoft.com/) It offers a great deal of flexibility so that good results can be obtained on a slower computer; a graphic equalizer with some compression of dynamic range; and even the ability to export an MPEG file in decompressed form in the wav format. Since CD-DA is uncompressed, an MPEG file cannot be the starting point for recording an audio CD-R. An MPEG can be stored on a disc, but to make it useful in a CD player, it must be decompressed to wav before burning.

Streaming

Streaming video and audio refers to the ability to play the picture and sound as it is being received. In general, that requires special capability on the server side and is not available from a CD-R. The formats used for streaming include RealAudio, RealVideo and TrueVoice. The RealMedia capabilities are very inexpensive, but require that the user either download or buy the player program. Efficiency and quality for video are less than MPEG, but more than some video codecs; RealAudio options are quite attractive, but no converter is available to produce wav files from ra. In addition, the author of a RealMedia file may mark it so that from a streaming server it cannot be saved to disc by the user.

E-mail me at cd-r@mrichter.com

Return to Mike's home page

Video CD

VCD is a well-defined and specific format, much like CD-DA for audio. Like audio, it requires a precise implementation of file structure so that it will play back correctly on a VCD or DVD machine which does not plug into a computer.

The VCD has one or more MPEG-compressed data streams recorded without the usual ECC of files. As with CD-DA, it is assumed that any errors will be unnoticed in the result. Unlike CD-DA, the data stream looks to be a true file and can be read (as a DAT) to your computer. However, since it does not have ECC, it can be too large to write back to a disc as a conventional file - and would not be recognizable in a VCD player if it were. Many MPEG players will play back DAT files as well as MPG's, even though they are slightly different internally.

To create a VCD, two steps are required. First, a whitebook MPG must be generated, then it must be recorded with software designed for that purpose. There may be boards or devices which create whitebook MPG's directly, but since those files are very difficult to edit, the preferred starting point is an editable AVI. The AVI from a capture device is most often edited with ULead's Media Studio or Adobe's Premiere; the latter is the choice of most serious users. The edited file can be saved with your choice of codec or those using Premiere and the Xing plugin can render directly as a fully compliant MPG. The ULead output is not fully compliant, but is acceptable to VCDC 3.5b and above. Both hardware and software encoders for MPEG are available. The Xing software implementation is relatively fast and consistently generates true whitebook code when the VCD option is selected.

The VCD format requires specific folders and files and the DAT itself must be recorded without ECC. Therefore, a dedicated program such as VCD Creator (a component of Easy CD Creator Deluxe) is required. VCDC will not record a non-compliant MPG file, so you will not waste a blank by discovering non-whitebook format after the fact. Note that relatively few VCD or DVD players will handle CD-R. Early Sony models accepted them without complaint, but later models (except the very high-end) do not. Similarly, first-generation DVD-ROM players do not read VCD's recorded on CD-R. Second- and later-generation DVD-ROM is MultiRead and should handle either CD-R or CD-RW. That is particularly advantageous in the VCD format because of the time and effort required in preparing and editing material. With audio, one can simply DAE the source to WAV files, then rearrange them and burn them to CD-DA. Theequivalent compilation capability in VCD requires a mastering program which will read DAT (as VCDC does). Otherwise, one may either generate one's own files from AVI or copy a disc in its entirety. If you are creating a VCD, it makes sense to burn an RW for evaluation before making one or more CD-R's for use. (Some standalone players which will not take write-once media will play erasable.)

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Correcting an MPEG for VCD

One question raised frequently is how to put MPEG files from various sources onto a VCD. Frequently, the answer is: you cannot - at least, not practically. That usually results in cries of protest: someone must have a simple way to 'fix' the file. The CD recording publishers are recognizing the VCD format straitjacket, so programs such as the VCD creator in Easy CD Creator are more tolerant than their predecessors and are even capable of compressing to VCD from AVI.

VCD is a strictly defined format - just as CD-DA audio is. If your original audio file is sampled at 44 Ksps instead of 44.1 Ksps, the only way you can make it into an audio disc is to resample it to 44.1. Similarly, NTSC video for VCD must be 29.97 fps. Other rates will play as MPG without a problem, but will not make a VCD. Rates that are frequently encountered on the Internet are 30 and 15 fps; 29.97 is a rare choice.

In some cases, a program such as FlasKMPEG will fix the problem; otherwise, if the original MPG does not conform, it must be opened in a suitable editor and recompressed in exactly the right format. First, there are not many editors which will do that; the best editors begin with AVI, not with MPEG. However, ULead Media Studio 5.2 will do it. Unfortunately, compression is a lossy process. In general, you begin with a file nearly conforming to the standard and end with one which has double the losses. While it is fully conforming, it looks bad. That's not because the compression process is done badly but because double compression is bad.

Returning to the audio analogy, if you download an MP3 sampled at 32 Ksps and want to convert it to CD-DA, you must first decompress it. You may do that by opening it in a WAV editor which accepts that input, such as GoldWave, or by opening it in a separate program, such as WinAmp, and saving it as a WAV. Once decompressed, the file must be resampled, typically in a WAV editor. The resampled WAV is then compressed again to MP3. Decompression is lossless. Resampling introduces problems depending on how well it is done; quick resampling can produce audible artifacts while precise conversion takes a long time. Finally, the resampled file is compressed to MP3. Whether the three steps are done in three, two or only one program, there are three stages of loss: the original compression, resampling and the second compression.

If I were forced to make a nonconforming MPEG conform, I would use two programs - in part because I prefer Xing's compression to ULead's. I would open the file in Media Studio and save it as an AVI, assuming it was not larger than 2 GB in that format. Then I would compress it with Xing. (**Note:** : Compression is *very* demanding of the computer. A 400 MHz Pentium II runs overnight on a 10-minute clip. That's another reason for using Xing - it's not as slow as Media Studio. The third reason is that Xing offers VCD settings which are absolutely compliant, where you have to set your own parameters correctly in Media Studio.

The conclusion I suggest from all of this is that the payoff for forcing conformance is too great to be worth the costs: software, learning curve to use it, time to process and quality loss. If your initial MPEG is not in whitebook format, you probably do not want to convert it to a VCD.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Compression and Decompression

Confusion in a Bottle

In large measure, this page is a recap of primer material which has proved hard to decipher. In part, that is a natural result of poor naming and confusing concepts. Regardless, let's try again.

The purpose of compression is to use computing power to give the brain the impression of more signal than is really there. The effectiveness of compression is measured by the amount by which the rate is reduced and the loss of quality in the result. The codec (the algorithm which compresses and decompresses the signal) uses a model of human perception to deliver a given data rate from a specific input stream.

Two kinds of video compression have been standardized; another has been dropped and a fourth is on its way. MPEG 1 demands a moderately powerful computer or low-cost dedicated hardware and delivers a moderately compressed picture with acceptable sound. It is used in many home systems and fits well with CD-ROM capabilities. MPEG 2 requires enough more from the computer so that it is not practical without hardware playback support; however, it produces high-quality video, typically the best available in the home. MPEG 3 was to have been even better, but it was dropped when digital High-Definition TeleVision (HDTV) was defined. MPEG 4 is under development now and will probably emerge with digital HDTV in the next few years.

Three levels of compressed audio are defined to support compressed video. They are MPEG Level 1, Level 2 and Level 3 (abbreviated by their extensions, MP1, MP2 and MP3) - and must **not** be confused with MPEG 1 and so on. In particular, MP1 is used for low-fidelity sources, MP2 for moderate fidelity and MP3 for high fidelity. Although audio compression level may be chosen independently of video compression, MPEG 1 usually uses either MP1 or MP2 depending on the video data rate to be achieved. If the video is being compressed a lot, then audio quality will be lowered to keep the data rate down and the advantages of higher Levels are lost. In general, a video encoder will choose the 'best' amount of audio compression for the target data rate of video compression. In some cases, there are specific standards (such as those in the White Book for MPEG for VCD). In others, you may adjust the balance for yourself. MP3 is the level used for DVD and other MPEG 2 applications.

Finally, let me note that compression is much more demanding than decompression. Typically, an MPEG 1 video designed to play back on a Pentium 200 without hardware assist in real time, will encode many times slower even with an efficient software encoder. Encoding MPEG 2 requires substantial hardware to be practical at all. At this writing, MPEG 2 encoding hardware is more expensive than the computer with which it works.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Common Files and Uninstalling

There are many complaints about the difficulty of uninstalling software of various types. This page addresses only one aspect which is particularly painful - shared files - because it affects CD-R operations. The end point is worth raising first so that those who already understand are spared the lecture. It is that some software components cannot be uninstalled because their necessary predecessors cannot be recovered. Therefore, the only practical way to get rid of them is by restoring a backup.

A program installed into Windows will typically have three groups of files: executables stored in its own folder(s); overlays, libraries, data and related files also located in dedicated folders; and common elements, often DLLs (Dynamic Link Libraries), which are located in such folders as Windows and Windows\System. The common elements are available to all applications. (There is a fourth group, elements common to programs of a family or a vendor. They share properties with the local ones and the common ones as should be apparent from what follows.)

One of the (in)famous collections of common files is the ASPI layer which translates between Windows and SCSI or SCSI-emulating devices. The separate modules form a set of which some are needed for any such communication and others only for selected jobs. The particular modules depend on your OS: NT/2K and 9x/Me use different components. Microsoft ships an outdated but functional ASPI layer with the OS - one which is sufficient for basic operations but cannot properly support CD-R.

When you install a version of ECDC prior to 4.02, an updated layer is installed with it if possible. "Possible" means that it will not overwrite a "newer" module with one which appears to be older. So if another program installed part of the ASPI layer with a later date, the update from ECDC will not replace it. The result is an inconsistent layer which can be corrected by removing the inconsistent files by hand. [Sidebar: There is another problem which arises when a program puts a version of a common file into its own folder. If that program runs before something else loads the common file, the local version will be brought into memory and will keep the system from loading the common one. In that way, your system may test fine for a consistent layer but at times have an inconsistent one - a clashing component was loaded by a 'rogue' program. Note that this is only one of the ways that ignoring programming rules can introduce problems. A more common one is a publisher modifying one of the common elements incorrectly and overwriting a valid version with its own.]

After installing a program which modifies the ASPI layer, what can be done with the files in the layer if you uninstall? They cannot be deleted because the other ASPI functions would be lost. The old ones can't be put back in because they're no longer there to restore. So the updated layer is left behind. 'Way back in the third paragraph, I mentioned that the ASPI layer interfaces for SCSI devices and for those which emulate SCSI. That emulation is used for IDE (which is why your IDE controller shows up as a SCSI device) and for many others. If you install a scanner, it may choose to update modules of the ASPI layer. Since the scanner manufacturer usually neither knows nor cares about CD-R, that "update" may kill a perfectly acceptable set of common elements. Uninstalling the scanner does not solve the problem because you can't recover the lost (valid) components with uninstall. The problems with rogue installs ulitmately reached the stage where Adaptec gave up on maintaining sanity. ECDC from 4.02 does not use the ASPI layer. The necessary communication is built into the local software and bypasses whatever is in the common files. That allows the CD-R software to run in the face of uncontrolled modifications being introduced by others. It's an unfortunate resolution since it leaves the chaos to be solved by everyone else and it wastes space on your hard drive, but it's understandable. The bottom line is that if you want to do a real uninstall, you had best restore from a backup.

You *do* make a complete image backup before any such installation, don't you? I was sure you did.

*P.S. - I'm not ignoring the partial backups of GoBack and other programs. They *may* work depending on the code and the situation. Restoring from backup *does* work - reliably and repeatedly.*

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Storage on a CD-R

Some of the following is adapted from a post to AdaptecCDR by Dick Langley, DickLTech@WWDB.org My thanks to him for permission to modify and to use his material. And since I modified it substantially, don't blame him for the mistakes - they're mine!

Let's start with an audio CD, just as the developers of CD-ROM did. CD-DA tracks are made up of blocks. There are 75 blocks per second of music. There are 588 samples per block; each has 16 bits for each of the two channels, so it takes 4 bytes for each sample or 2352 bytes per block. Now, the accuracy of reading from a CD is not nearly as high as from a hard drive or RAM. For audio, that's not much of a problem, but when the disc is used for data (as a CD-ROM), drop a bit or two and there goes the whole program. To reduce the rate of data errors and to prove that we're computer types and think in powers of two, they decided to use the same block to hold 2K (2048) bytes of data and to take the remaining 304 bytes for error correction. So a block on a CD-R is 2352 bytes long and can hold either 1/75th of a second of redbook (CD-DA) audio or 2K of data.

Audio

In documentation of audio and video processing, the word 'sector' is used for a chunk of information. By default, that chunk occupies 1/75th of a second and fits into one block. Because sector comes up later, I'll stick to blocks, thank you - and you can translate when you read about audio editors and such. And you are also now prepared to do your own arithmetic to see why 74 minutes of audio fills the same blank that holds 650 MB of data, or to understand why, when you extract a CD-DA to your hard drive, it won't fit into 650 MB. And all that neglects the cost of the header on a WAV file. On a CD-DA, there is no header. The information is in a continuous stream of bytes from one block to the next. There's a TOC to tell the player at which block each track begins, but there's no audio file on a CD-DA. (The .CDA you see with Explorer is a fiction.) To create a WAV file from the data stream, a drive capable of Digital Audio Extraction (DAE) supplies the byte stream to your software. That software converts the stream into WAV format, complete with header. DAE is done by dedicated programs (see the Files and URL pages at this site) or by your CD-R authoring program. Converting from WAV for CD-R is done by the authoring program.

If the information in a wave file is not an exact multiple of 588 samples, software pads to a complete block with nulls - silence.If you're recording a concert (for instance) and you want your selections to be addressable as different tracks via the TOC, those tracks should be multiples of a whole block. As you slam two tracks together, the sound (crowd noise for instance) should be constant when transitioning from one track to the next. If you don't have a true multiple of sectors, the last sector of a track will be padded with nulls or no sound. (In some older editors, the padding is not done with zeroes and whatever bytes were lying around will be read onto your disc). As you play your new CD and you transition from one track to the next, you may hear a click (where the sound you hear goes to nothing or to random bytes, then back again). It becomes annoying to hear these clicks between tracks. To solve this situation, software offers quantization. When you're cuttin', pastin', and slicin' tracks apart or together, it will do it in multiples of 588 samples to give you whole blocks. So if you have that concert that you've broken up in tracks, it sounds like one continuous track when played. Mike Looijmans' CDWAV (on the links page) was designed to cut on block boundaries. As you look for software that manipulates wave files, you should look for this "feature". Of course, that means your split may be moved. If you can hear the difference when it's shifted no more than 1/150th of a second, try analogue recording.

Data

Now for data blocks. Again, there is a problem stemming from the fact that there's no File Allocation Table on a CD-ROM. There **is** a TOC and most of what a FAT needs is contained in the TOC. So OS extensions are used to translate the TOC to a FAT - on a PC, to a FAT which looks like the one you would have on a 650 MB hard drive. Information on a FAT16 hard drive is stored in up to 64K sectors; the size of each sector is the total size of the drive divided by 64K, then bumped up as necessary to the nearest power of two. So a 650-MB hard drive would have sectors of 16K bytes each. Since each file must begin on a sector boundary, it will waste whatever space is left unused in the last 16K sector.

Ah, but things are better on a CD-ROM with its 2K blocks, right? No - that would be too simple. Since the OS extension must address the CD-ROM as though it had a FAT, it must look for the file to begin on a 16K-sector boundary. A 4K file wastes 12K on a 650-MB hard drive or a CD-ROM. On average, each CD-ROM file wastes half a sector or 8K. When you add up the file sizes for a disc you are about to burn, you must remember to allow for that wasted space. If you are writing from a drive which has 16K sectors - such as a Jaz or a FAT16 hard drive with at least 512 MB but less than 1 GB - you can use Explorer and Properties to determine exactly how much of your blank will be needed. If your source drive has a different sector size, you will have to do your own arithmetic or approximate using 8K wasted per file - and a little more just in case. *There's more on this subject in the page on Packet writing.*

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Packet Writing

As usual - but perhaps more so given my ignorance - this material is unofficial. In the hope that it's helpful, buckle up for a hard ride. Also as usual, it is aimed at Windows users; Mac users should be aware that much of it needs to be adapted to their/your needs.

ISO 9660, Level 1/Level 3

The International Standards Organization controls standards generated by industry. The standard for CD-R is numbered 9660 and covers the agreed rules for recording all sorts of CD's. Not all manufacturers of hardware and software look on those standards the same way: some violate them freely, some implement them in part, some try to adhere to them strictly. When a publisher like Adaptec tries to implement them fully and a hardware vendor does not (names omitted to avoid assigning guilt), problems arise. The user complains that Adaptec does not do something the drive manufacturer claims to do and everyone is likely to be both right and wrong. Adaptec says: the h/w does not implement the standard; the h/w maker says: it does what it needs to do; the user says: it's not my fault. There are many solutions available. Another software package may be used which is less stringent about the rules; a different device may be used which follows them more closely; the user may give up on the capability.

ISO 9660 Level 1 covers the formats we have known for a while. They are CD-DA, CD-ROM Mode 1, CD-ROM Mode 2, Mixed and CD-Xtra. In the first three, the disc may be written Disc At Once (DAO) if the hardware and software support the capability. CD-DA is covered in other pages of this primer under Audio. CD-ROM Mode 1 was designed for single-session recording; Mode 2 is slightly less efficient and was designed for multisession. In fact, either may be used for either single- or multi-session - the difference is that while only very old drives balk at multisession Mode 2 discs, some moderately old ones won't take multisession Mode 1. Both Mixed and CD-Xtra modes combine CD-DA and CD-ROM on a single disc. Simply put, CD-Xtra permits playing the audio on a conventional CD machine since the audio tracks come first.

On a conventional CD of any flavor, the first information the reader sees is the Table Of Contents (TOC). In CD-DA, the .CDA 'file' specifying each track is a fiction that a file manager (like Explorer) uses. In a single-session, closed disc, the TOC contains information which is interpreted by a DOS extension as being a directory. The rules for converting TOC information (including the fact that everything on the disc is read-only) are embedded in the extension, such as DOS's MSCDEX. On a multisession disc, the TOC for the first session tells the system where to find the next one - or whether the current one is the last, or whether the disc is now closed. DAO is necessarily single-session (which is why one cannot write DAO tracks in Mixed or CD-Xtra formats).

In order to support UDF, the TOC - first information on the disc - had to change. As a result, MSCDEX and other extensions cannot read packet-written discs even after they are closed to ISO 9660 format - the information that the extension expects to see is simply not visible. The solution was to invent Level 3. Functionally, it looks the same as Level 1 to the user, but it requires a different extension. That extension is available in Windows 95, Windows NT, and Apple System 8, so that they can read Level 3 discs. It is not in DOS, Windows 3.x or other systems. **This has nothing to do with your hardware or application software.** Someone may write a new extension, but Microsoft may not be interested and there's not much money to be made from the effort to inspire a company to do the job. If you write one, you will be much loved by many users, but probably not well paid for it.

Universal Data Format (UDF)

Now comes packet writing - which has two distinct flavors. The idea behind writing packets is that if one does not worry about the TOC, one can write data to a CD-R in blocks or packets. The information that goes into each packet needs to be accessible by the writer, but not necessarily by an ordinary reader. By writing an appropriate set of functions into the heart of the OS - approximately at the level of the extensions to DOS - a vendor can provide packet capability for writers with suitable hardware and firmware. However, there is no TOC in the Level 1 or Level 3 sense; that information is on the disc in other ways. That takes extra space on the disc and means that those data must be interpreted when a UDF disc is inserted into the drive.

At one time, Sony had its own format for packet writing and provided a driver to permit a UDF disc to be used in a reader; that appears to have been dropped in favor of the standard discussed below. CeQuadrat also has a form of packet writing in PacketCD; its capabilities have not been reported and I cannot test it because it runs only in Win95. Adaptec's DirectCD implements the UDF standard and works in Win95, Win98 (so far) and NT 4. For the following, I will speak only of DCD and its implementation of UDF. If someone can provide information onother realizations, I'll be glad to append it.

In the first version(s) of UDF, only conventional, non-erasable media were considered. For them, a very efficient system could be devised. At the head of each packet is the information on the file it represents including its length (which, of course, implies where the next packet begins). Each packet contains an integral number of blocks; the only space wasted is for that header and for the leftover bytes of the last 2K block. A variable-length packet can be converted to ISO 9660 Level 3 format because all the bytes of the file are in complete, consecutive blocks. When the session is closed to Level 3, the TOC is written in a more or less conventional fashion, but data interpretation requires that the OS ignore the old header information. In other words, the TOC is similar to that of Level 1, but both the TOC and the data are different enough to require changes that make the old extension useless. (Note that the Level 3 TOC may be written either to close a session or to close the disc. If the session is closed, another UDF session can be started and finalized; when a disc is closed, the TOC indicates it and will not permit reopening.)

A note on capacity: Variable-length packet writing is very efficient. Because of the small amount of space wasted, it is possible to write more blocks than can be accomodated by a FAT16 for a 650-MB disc. If you fill a variable-length packet disc with many files of the right size, it may not be possible to close it to Level 3. That means you won't be able to read it except in your writer - the session is still open and a CD-ROM cannot read an open session.

When UDF is applied on an erasable blank, a new option is available. In addition to being able to write variable-length packets and to erase the whole disc, you can write, read and erase individual fixed-length packets. Now the CD-RW begins to resemble a conventional floppy, where the packets are the sectors, at least from the point of view of the user. That capability is reflected in DCD 2.x. However, one problem of erasable blanks is that they support a limited number of erasures - nominally, a thousand. Another problem is that when one rewrites on any drive, data may be written over previous entries. Overwriting results in 'scrubbing' or disproportionate reuse of a single region of the disc; it also leads to fragmentation of files so that their constituent blocks are no longer necessarily contiguous. For fixed-length packets, the consequences include high overhead to keep track of all the blocks and inability to convert the disc to any Level of ISO 9660. So an erasable blank formatted for UDF offers only 494 MB of storage, slow access (about the equivalent of 1x) and inability to make the disc readable in a conventional player. Neither Microsoft nor Apple will be reading fixed-length packets in their promised OS's, so those discs are and will remain for a while useful only in the writer or in a MultiRead reader with a suitable driver (such as the one now available from Adaptec's WWW site).

If you think about it for a moment, that sounds much like the case for a very large floppy - which is a good analogy to keep in mind. Both a floppy and fixed-length packets waste about 30% of raw capacity on formatting; neither can be read in a device which cannot write it; each is relatively slow; and if you have a catastrophic failure during writing, you may lose the whole disc. Of course, the UDF disc is about 350 times bigger than a floppy, but that's a (rather important) detail.

Adaptec has not chosen to offer the user the option to format an erasable for variable-length packets. Rightly or wrongly, they find little sense in using them on CD-RW, but therefore face a real potential for user confusion. As long as they are the only contenders in the general multi-OS marketplace, we can only ask them to re-evaluate. If they get a serious competitor, the market may make them change their minds.

How UDF is implemented - and when it's safe

When a CD-R is formatted for variable-length packets, space is committed for the TOC, but no TOC is written. Nothing else of importance is done, so the process is quick. When a file is written to the disc, it becomes a packet which is headed by the information corresponding to what would be in its FAT entry, including its length so that the location of the next packet can be determined. At the same time, DCD holds in memory a virtual FAT (VFAT) which gives access to that file just as though it were on a conventional disc. Write another file and the VFAT is updated appropriately. When the disc is ejected, the VFAT is removed from memory and nothing is written to the disc.

If the session is closeded when the disc is ejected, the information for the TOC is written to the disc from the VFAT; the packets themselves are not altered. The TOC is then accessed by Win 95 or NT 4 through the usual extension. Until the disc is closed, there is no usable TOC information on it - which is why it cannot be read conventionally. A disc with a closed session has a TOC in Level 3 format; that TOC may point to the place another TOC will be written or may say: this disc is closed. During the time that a packet is being prepared and written, a catastrophic system failure would cause the write to be incomplete and might make the disc useless; at all other times, even if the system is powered down with the disc in the drive, the information is safe. The laser is off, everything needed is already written and nothing can happen. Apply power, return to your OS, and DCD reads the packets and forms the VFAT.

With fixed-length packets, the process differs in some significant ways. The key fact is that when the disc is ejected the VFAT is written to the disc so that the location of the packets forming each file can be traced. Logically, one would ask: Why not write the VFAT each time the disc is written. The answer is, simply, that that would increase the nuber of writes to the space that it occupies and would kill the disc more quickly. The VFAT in memory becomes a FAT on the disc. Since the runout track was written, a suitable driver then gives access to the files. That driver is provided by the UDF Reader from Adaptec. Since a fixed-length-packet disc is always RW, that driver is only useful in a MultiRead drive or a writer capable of reading erasable media. A fixed-length-packet disc still in the drive after having been written must be ejected (and have the VFAT written) to give access to the new files. If power fails or the system is reset before ejecting the disc, the new information will be lost and the disc may be unreadable. With luck, a utility to recover what can be recovered will become available, but at this writing the disc is vulnerable to such a failure.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Data in the Mix

Now that you have mastered recording data and audio separately, you are ready to tackle those great interactive discs which mix the two - and maybe other stuff as well. There seem to be many ways to do it, but what are they? how are they used? how are they written? Unfortunately, with one exception they are complicated, used with difficulty and written in various ways. Since this is a primer, not a comprehensive treatise, it will cover the easy way and give some clues to the others which you are free to research if you wish.

Before getting into that, it's important to keep in mind some properties of CD readers. First, almost all CD-ROM drives are multisession so will see the **last** session on a disc and all that are linked to it. Second, all audio-only players are single-session so will see only the **first** session on a disc. A player which handles multiple formats, such as a standalone DVD player, should be single-session at least for audio and will probably be single-session for everything. However, one which also handles MP3s may be mutlisession, UDF-compatible and just about anything the manufacturer makes it.

Another factor which has not been significant until now but is for this page is that properties such as Mode are associated with a track, not a session. Usually, each session consists of tracks of a given type - you would not drop a data track among audio tracks. However, there are cases here where that's exactly what is done.

Enhanced CD

This is the way to mix audio and video which works reliably with almost all drives and is easy to implement. It has no official name but is often identified with CD Extra; as you will see later, it's not CD Extra. In this mode, an audio session is written first with as many tracks as you wish. That session is closed but the disc is left open so that it can be followed by a data session. The two sessions are **not** linked. You create separate volumes. When placed in a single-session reader such as an audio player only the first session (audio) is seen and the disc plays as a conventional CD-DA. The same disc in a multisession CD-ROM shows up as data and can Autorun and otherwise operate as though a pure data disc. In fact, to get to the audio session, you must use a tool such as Session Selector or ISOBuster to make the unlinked CD-DA visible to the computer.

What if you want to access the audio from the data side? Simply put, you cannot. There is no link between the two volumes, so neither knows that the other is there, let alone what's in it. Even if you had the link, you would have no way to address the tracks from the data side. One solution is to include MP3 versions of the audio tracks on the data side; then the sound can be played as though there were a link without spending a great deal of space on it.

Defined combination modes

There are many of these and only a few will even be discussed here. Further information is available in the CD-R FAQ at http://www.cdrfaq.org/ and in the excellent Technical Library of PDF files at http://www.cinram.com/ Compared with the authoritative sources in the colored books, both are easy reading - but compared with making a "Enhanced CD", they are heavy going.

CD Extra

CD Extra is a defined format with a number of audio tracks in a first session followed by a data session. It has an extra folder and some files to accomodate player capabilities no longer used or needed and essentially obsolete. For all practical purposes, it is easier to use the method above under " Enhanced CD" and save those extra files.

Mixed Mode

In Mixed Mode, a single session is written with a data track followed by audio tracks. Because it is a single session, the audio can be accessed from the data side with appropriate software - but that software is not generally available and involves information not accessible to the casual user. This mode is out of favor because the data track comes first. Most modern audio players will skip a data session, but some may not and the resulting sound can be catastrophic. Mixed Mode is still used for some interactive programs written by experts and intended for restricted players, not for a Discman.

Video CD

A VCD is similar to a CD Extra disc except that the audio tracks are replaced by video files (DAT) in Mode 2 Form 2. Because the folders, files and pointers are explicit in this mode, a specialized program is needed. General-purpose mastering programs such as ECDC, WinOnCD and Nero provide VCD capability.

Writing in the pregap

The standard requires that an audio disc begin with a two-second "pregap" before the first track. The location of that first track is specified in the TOC, so it can be made longer than two seconds - the equivalent of 300 KB - without confusing most players. By setting a larger pregap, information can be stored and hidden from 'normal' operations. Conventional mastering programs do not give you access to the pregap and do not provide ways to use it if you get there.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Arithmetic 101

This note is intended to serve two purposes: give you some idea of the limitations in recording data files and show you how simple arithmetic can be used to investigate problems you might otherwise ascribe to hardware, software, media, or an evil spell.

Problem: I have a simple layout with a lot of files to burn as data to a CD-R - and it doesn't work! Somebody said to defrag the hard drive, but why should I do that? Do I need an AV SCSI drive for this sort of simple job? No - all you need is to think about what's going on and to apply some logic.

Suppose we start with 40,000 small files. How small? first guess would be about 16K each; if we make them all just under that, they will fit onto a standard blank. In fact, how much smaller than 16K they may be makes little difference and can make the situation worse, as you will see. Let's assume that I'm writing them from a defragged drive and that I have good caching of that drive so that its directory stays in RAM. Then retrieving that file takes a check of the directory cache, one seek to the file start and 16K data transfer.

On a typical hard drive, you can transfer 3 MB/sec or more and seek time is around 10 msec. Transferring 16K will then take 10 + 16/3 or about 15 msec. If the files are smaller, the number could be as low as 10 msec - the seek time. That means that the drive will transfer 70-100 files per second and that 40,000 files will take 400-600 seconds (7-10 minutes) to write.

Now some more arithmetic. If you write a 650 MB disc at 8x, you will be writing for a bit more than 9 minutes. It takes 10 minutes to fetch the data and 9 minutes to write them without stopping - **underrun!** Suppose the files are smaller? Oh, then you will be writing less, that will take less time and **underrun!**

Now, suppose the disc with the files is fragmented. Each fragment requires a seek. If it's fragmented badly, then many fewer files will cause the same problems as a lot of files on a defragged disc. If we make an image on a defragged disc, there is only one seek required (to the start of the image) and you could write at 20x - regardless of the number of files or their fragmentation.

Please notice the following:

- This has nothing to do with your mastering software
- It has a little to do with the size of the files
- It has nothing to do with your medium
- It has a lot to do with fragmentation
- There is no maximum file count other than that of the FAT16 filesystem in which the CD-ROM is interpreted
- The recommended solutions work: defrag or make an image (or both)
- A faster drive will help a little, but not enough in most cases to pay off

We can carry this a bit further still and move from the HD to a fixed-length packet disc. Here, each packet is 32K and they are deliberately separated on the disc to reduce scrubbing when files are deleted or replaced. So each 32K read requires one seek to the start of the packet and 32K or less of data reading. On a CD-ROM, seek time is relatively long. Your assignment, should you choose to take it, is to find the seek time on your drive(s), do the arithmetic and see if you can understand why it takes so long to read a fixed-length packet disc and why the rotation speed is not the issue.

E-mail me at cd-r@mrichter.com

Return to Mike's home page

MP3 Encoding

The need for audio compression has been apparent for some time. Even with a high-speed connection, the 10 MB/sec of redbook audio is hardly acceptable for streaming over the Internet. For some years, MP3 has been the compression scheme of choice even though it is lossy; lossless compression can rarely save more than 30% of the size of the file, where 90% reduction is common with MP3.

A bit of history first. The Motion Picture Experts Group (aha! that's where MPEG comes from!!) covers audio, video and still-image formats. The audio formats are labelled MPEG Level 1, Level 2, etc. Each is a superset of the lower ones so that MP3 includes MP2 which includes MP1. Each also uses "perceptual coding" in which less significant components of the audio are sacrificed for smaller file size. Note that as more compression is used, maintaining constant quality means more computing.

With the Fraunhofer Institute's development of L3ENC, high quality and substantial compression were realized. A serious listener with excellent source material can hear the degradation of the Fraunhofer engine even at high rates, but the penalty is not great and storage requirements are reduced in many cases by an order of magnitude - a payoff well worth the price to most users. For similar compression at low rates, other engines produce more artifacts and poorer frequency response; again, they may be acceptable to many users. In all that follows your judgement on matters of sound quality, cost and ethics should be applied; I am trying only to report the facts as I have observed them.

The Fraunhofer Institute focussed development on the MP3 encoding algorithm and owns the rights to the one in general use. (WARNING: I will **not** enter debate on the moral or legal implications of algorithm patents or of copying commercial material.) For the rates of interest to MPEG, everything up to Level 3 is handled by the high-quality Fraunhofer codec at least as well as by anything that has shown up since. When one moves to the higher sample rates and higher bitrates of Level 4 (and above?), the rules are different.

Originally, the Fraunhofer codec was licensed at a substantial fee for a higher-speed, lower-quality version and a very steep one for the full package. An individual license for general-purpose use on your computer cost of the order of $300 either directly from them or from companies which licensed it to wrap in their own shell. I signed on then, buying the AudioActive Studio Pro at a painful price. That installed the Fraunhofer codec into Windows and let me run programs such as ECDC which used it to encode audio. Along came a number of other codecs with their own algorithms. They were priced low - or free. Some were extremely fast, though not very good. Still, they satisfied many users' needs and they killed the market for the Fraunhofer.

So Fraunhofer offered a different way to use the high-quality encoder: contain it within a program. That brings the license down well under $100 and products such as CoolEdit can afford to offer an MP3 plugin and AudioActive Studio Pro can be priced competitively. Unfortunately, that means that their codec is not accessible to other programs - it is not installed into Windows. I suppose one could still license the codec directly from Fraunhofer, but it is probably still priced out of reach. So lower-quality codecs are available which may well install into Windows for you. I had some cluttering my system before my recent move to Win2K. For now I'm prepared to run CoolEdit Pro with the plugin and AudioActive Studio Pro and not compress audio in programs such as ECDC. When I get a good solution, I do not want to have to figure out how to remove garbage, even garbage I paid for in my more ignorant days.

High-quality codecs are available as freeware - both BLADE and LAME are generally regarded as superior to Fraunhofer above 128 Kbps - but I have not found an implementation which installs one into Windows. If someone has a method for Win2K, please let me know. I have both BLADE and LAME on my system for use in Exact Audio Copy, for example. Note that they demand substantial computing power; on most machines, they run far slower than real time.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

MP3 and the Perfectionist

To begin at the beginning, MPEG stands for Moving Picture Experts Group. That body establishes standards for digital video and audio. We are concerned here with the standards for the audio layer in the MPEG1 format. MPEG2 is in use today, but is not related to the MP3 files; they are MPEG1 Layer III audio. Simplifying the situation, different layers impose different loads on the decoding software - the program which converts the MPx file to uncompressed Pulse Code Modulation (PCM) audio to drive the reproducer. Layer III - MP3 - is consistent with modern low-cost dedicated packages and with Pentium-class CPU's. Layer 2 - MP2 - is less demanding; its performance is not adequate for it to be considered quality reproduction. In theory and usually in practice, system and software reproducing a given Layer will handle any lower Layer.

Simplifying again, the MPEG standard for a Layer specifies the playback of a file encoded for that Layer and leaves encoding to the developer. MPEG audio employs perceptual encoding and is lossy. That is, it compresses the data stream by throwing away information which the encoding algorithm 'believes' will affect the listener least. The decode side of the codec (code/decode algorithm) is usually pretty simply implemented from the standard; there are some differences which will be discussed below, but in general the playback algorithm is not an issue.

The encoder is a different matter altogether. There are three parameters input to any encoder to control the way it process the file: channels, sample rate and bitrate. Channels simply means monaural or stereo; in general, the encoder will provide 'stereo' channels (identical content) from a monaural original, combine two original channels into one, or leave the count unchanged. Sample rate is simply the number of samples of the data per second. Bitrate dictates the size of the encoded file. Those factors are interrelated.

Rates in common use for digital processing include 44100, 48000 and 96000 samples per second. Even though it violates convention, those are usually shortened to 44.1, 48 and 96 Ksps - which will be done here. (The standard for CD encoding is published in a book with a red cover; uncompressed, 44.1 Ksps, 16-bit, stereo signals are conveniently referred to as 'redbook' in its honor.) The more serious problem, representing Ksps as KHz, will not be accepted here. As is well known, the maximum frequency which can be encoded digitally is one half the sample rate, providing an incentive to take more samples to preserve high frequencies. In the octave below that limiting frequency, phase shift can substantially alter waveforms even though amplitude loss may be acceptable. For various reasons, it may be desirable to have the sample rate in an MP3 file different from that of the original. Many MP3 encoders will resample for the user. Some are limited to commensurate rates such as downsampling 96 Ksps to 48. Others will handle incommensurate rates and can accomplish the more difficult task of resampling 48 Ksps to 44.1. The quality of resampling may be significant here. Simply put, samples are created or destroyed in resampling and that may be done with slow, careful algorithms or quick, simple ones. Needless to say, the effects are audible if one is willing to listen.

Bitrate dictates the size of the finished MP3 file per minute of audio. CD-quality audio is defined in the standard and requires about 175 Kilobytes per second (175 KBps). MP3 bitrates are specified in Kilobits per second (Kbps - note the lower-case 'b'). The rate most often used for CD audio is 128 Kbps - which corresponds to 16 KBps or about one eleventh of the redbook rate. Other rates are frequently used, usually 64 or 256 Kbps though lower compression (higher bitrates), but 128 Kbps is usually assumed. As you would expect, throwing away 80-95% of the original information will impact the sound of the MP3. There are many encoding algorithms on the market. Some, such as BLADE, are optimized for lower compression than usual and will be neglected. Of the others, both subjective and objective tests indicate that the patented algorithm developed by the Fraunhofer Institute is the most pleasing. That is unfortunate, since it is by far the most costly to license - $300 versus $15 or less.

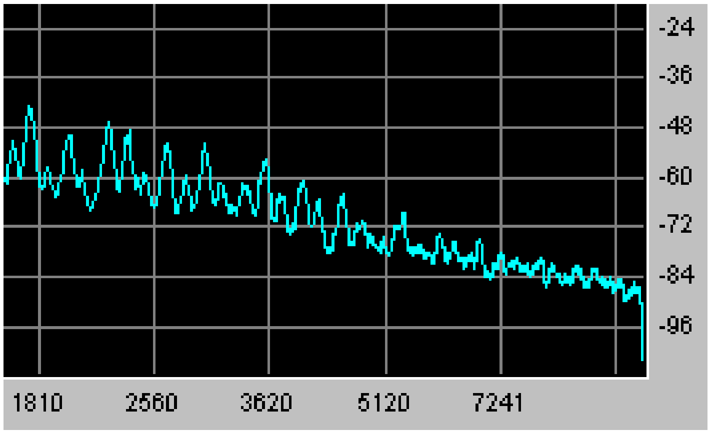


Figure 1:
Original Wave

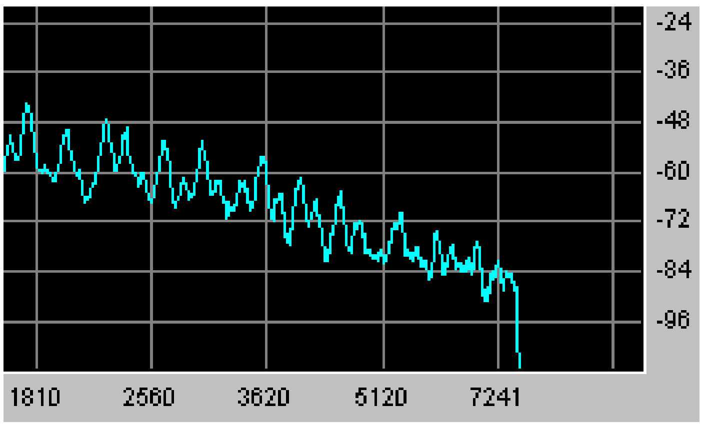


Figure 2:
Fraunhofer

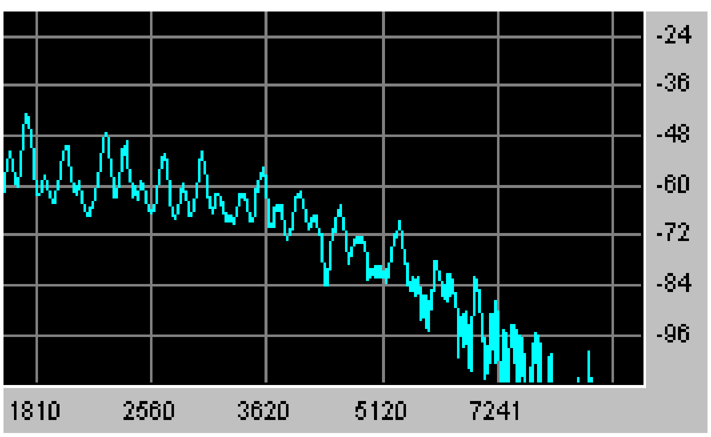


Figure 3:
Encoderx

Quality is a lot easier to hear than to describe. Absent an objective measure, there are two parameters of primary interest: preservation of audio spectrum and avoiding artifacts. In practice, they go together and codecs which maintain spectrum best tend to introduce the fewest audio disturbances. The most common form of artifact is a metallic tone which includes some narrow resonances; it's one of those things you'll know when you hear it but it defies description. It appears to originate when signals near the maximum frequency are encoded differently from those near them in pitch. The Fraunhofer codec gives very nearly flat response up to a frequency at which it cuts off abruptly. Another codec may nominally extend the response another half octave, but in the process substantially distort the response curve. Figure 1 shows the spectrum of a 22.05 Ksps monaural original file. Figure 2 shows the same file after encoding with the Fraunhofer high-quality option and decoding back to PCM. Figure 3 is the same with another algorithm. Note that the spectra are quite similar up to about 1.5 octaves before the cutoff at 11.025 KHz. The Fraunhofer remains nearly flat until it falls off a cliff, making no attempt to encode the last half-octave. Another algorithm provides some output in that half-octave, but is 12 db down at 8 KHz where the Fraunhofer is within about 1 db up to its cutoff. In short, the Fraunhofer provides good encoding within its frequency range where the alternative generates artifacts amid some signal on significant overtones. Needless to say, the perception of the sound of the samples is very different even to an untrained listener.

If artifacts are audible in the encoded file, the only solution with a given algorithm is to limit the frequency response of the original. Needless to say, that limiting must be more severe, perhaps by a full octave, with an inferior encoder. Another solution is to increase the bitrate or, in the case of variable bitrate, to increase the maximum bitrate, but either of those would increase the file size.

There are listeners who find even an inferior codec's 128 Kbps file identical to a redbook source. With any audio acuity at all, a listener should be able to recognize loss in the Fraunhofer at that rate in A/B comparison with the source; she will be significantly disturbed by the artifacts of a poor encoder, though she may be unable to describe the faults in the sound.

One point in all of this is that evaluating an MP3 player as a high-fidelity device is not as simple as evaluating CD players. It is necessary to know not only the sound of the source - often, an audio CD - but also the encoding algorithm that was used. Many of us are familiar with the influence of engineering when transferring a master analogue tape to CD; the same original can sound quite different depending on the equipment used and the engineer's judgement. Fair comparison of two reproducers requires that they be loaded with the same MP3 file, not simply compared using whatever file it may have against the original. There are MP3 playback products which 'compensate' for the faults of the encoder they are intended to follow. That is a kind of inversion of the deservedly deprecated Dynagroove process in which the original was distorted to correct for faulty reproduction. As with any high-fidelity product - video, audio or otherwise - the ideal is to maximize quality of each element in the chain, not to correct in one for expected faults in another.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Photo Facts

Contrary to popular opinion, not everyone records only games, audio, video and data onto CD-R. Some people put photos on as well. Since Kodak's proprietary PhotoCD format for discs is no longer available (but see the section on formats for word on their files), the following is generic for contnuous-tone images, not specific to CD recording. Still, it seems worth a page of exposition. Be warned, though: there's a fair amount of elementary arithmetic to follow.

Resolution

We are concerned here with printing from the computer, not by optics onto photosensitive paper. Unless you use a dye-sublimation printer (where dots and pixels are equivalent), we must begin with some treatment of pixels. A pixel here is an array of printer dots which represents a shade of a color. Since conventional printers only produce one 'shade' of dot in a given color, in order to have varying shades we must have a different number of dots considered as a group. That group is termed a 'pixel'. A pixel of 8x8 dots offers 65 shades of color (0 through 64 dots printed in the block). The extra dot is usually ignored so we speak of 64 shades for an 8x8 pixel. The original image probably has an eight-bit byte to represent that color, which means that there are 256 values available; to represent all of them, one would need a 16x16-dot pixel.

The resolution of a printer is specified in dots per inch. The manufacturers fudge a little sometimes, so a "1400-dpi" printer may only deliver that resolution on special paper with a particular sort of test pattern - but that's another story altogether. Suppose we have a true, 600-dpi printer. If its output is broken into 16x16-dot pixels, there are 37.5 pixels per inch - 600/16. That's about twice as coarse as a newspaper photo, four times worse than a good magazine's output. In short, it's too coarse to be acceptable. For a quality image, effective resolution of 150 pixels per inch (ppi) is a good target.

Photographic resolution is a somewhat different story. Photo resolution is measured in lpm: line pairs per millimeter. There are two dots in a line pair and 25.4 millimeters to the inch, so good film in a good camera can deliver 100 lpm; in computer terms, 5000 dpi! (Now you know why film scanners have such high resolution.) Incidentally, a photographic print is usually considered sharp at 7 lpm or 350 dpi. No doubt, that was a factor when laser printers came out for text (no gray scale needed) at 300 dpi. Note that a dot in photography, as in dye-sub printers, does convey shading. As a result, that 7-lpm, 350-dpi print corresponds to something like 3000 dpi in an ink-jet or laser printer.

Digital cameras make a useful reference point and it may be worth using their resolution - in megapixels (MP) with three colors each - as a reference. Here's a short table of values relating those continuous-tone pixels to performance of films, printers, computer screens and television sets.

Application	**MP**
35-mm frame @ 100 lpm	35
8x10 photo print @ 7 lpm	10
8x10 dye-sub print @ 300 dpi	7.5
8x10 print @ 150 ppi	1.8
4x5 print @ 150 ppi	0.45

640x480 computer screen	0.3
320x240 VHS playback	0.075

File Formats

There are two broad classes of bitmap image format: lossy and lossless. The common lossless formats include PCX, TIFF and GIF, any of which is quite usable for image storage and retrieval. GIF is usually used for images with only 256 colors, not enough for continuous-tone quality; if the 24-bit version is considered, the same rules apply as for PCX and TIFF.

In a continuous-tone image such as a typical photograph, lossless compression has little payoff. Since an uncompressed TIFF is probably the most universally recognized format, its requirement of three bytes (one per color) per pixel is a good reference point. As a result, it will take more than 100 MB to store all the information in a 35-mm film frame. Even a good 8x10 print at 150 dpi - magazine quality - is over 5 MB. Ouch!

There are now competing schemes for compression of continuous-tone images. By far the most common is JPEG. JPEG compression is quite similar to that of MP3: detail is sacrificed in ways least noticeable to the viewer in order to save storage space. The amount of compression is controllable in software. The more you save, the greater the chance of visible artifacts. The artifacts show up as pixellation of colors - regions of gradual shading turn into blocks of solid colors - and ragged diagonals. Most users seek about 10:1 compression as their compromise, but you should play with your own graphics program to see what its controls offer in price and payoff.

In general, a single compression can be chosen so its effects are acceptable. However, a second compression, even with the same settings, will result in serious image degradation. If you are delivering finished images which will need no further processing, JPEG is a good choice of format. If you are storing the images for editing later on, uncompressed files are far better. If that sounds familiar to those who have done (lossy) MP3 compression, it should.

The Kodak PhotoCD format is a modified form of JPEG with very similar properties. With appropriate software, the image can be retrieved in any of a number of sizes - a minor convenience when resizing in an inexpensive image editor is so convenient. (It made a big difference when 8 MB was a lot of RAM.) The drawbacks to that format are many: access often requires specialized software; the format is not easily displayed and is not suitable for the WWW; the file size is greater than that of a JPEG for the same maximum dimensions and compression. In short, unless you can write the nearly obsolete format for PhotoCD set-top boxes and use one, this file format has no advantage.

Storing on the CD-R

Here, the options are yours and there is no 'best' answer. On a PC, you may want to create an index with ThumbsPlus or a similar tool, though the ThumbsPlus CD writer is prohibitively expensive for personal use. Adaptec's Easy CD Creator Deluxe includes PhotoRelay with useful capabilities not only for slide shows but for video and audio as well. WinOnCD will let you write a VCD with up to 99 still photos. You can also write your own HTML for cross-platform capability.

Before you start your imaging project, clarify for yourself what the purposes will be. If you just want to have the pictures display well on the computer's screen, a 20KB JPEG will do for each and a standard CD-R will hold more than thirty thousand of them. In contrast, you'll have room for only half a dozen full-quality 35-mm frames on such a disc.

E-mail me at cd-r@mrichter.com

Return to Mike's home page

Short File Names Long

One of the rules for using data in a computer is that fully qualified names must be unique. That is, there can be at most one file with the name D:\d1\d2\d3\filename.xyz.abc and so on. If there were more than one, when you asked for a specific file the system would not know which to give you. That's true whether it's a Short File Name (SFN - maximum of eight legal characters, a period, and maximum of three legal characters) or a Long File Name (LFN - and with too many variations in legality to provide here). "Fully qualified" means including the full path from the drive letter through all the intevening directories/files to the file name. For these purposes, note that a directory or folder looks just like a file when viewed from the folder or directory at the next higher level.

In various places on your system, you probably have several files named read.me - or readme.txt or whatever. Suppose you wanted to put two read.me files into one directory. The OS would, quite properly, refuse. It can't put the second one there with the first because the fully qualified names would be identical. That would also be true with longfilename.txt - the fully qualified LFNs would be identical. The major problem arises when one has different fully qualified LFNs but one needs to have SFNs which are visible in and acccessible on a system using 8.3 names. MS decided to write the SFN logically, as 6~k.3 where k is a single character counting 1 to 9 to a to z (and on from there to 5~10.3, etc.). Now, there is an error in that process - which you will discover if you try to retrieve file longfi~c.txt, but that's not the point here. The point is that the fully qualified SFN must be unique in any system.

The SFN is not a property attached to the file. When a file is written to a directory (folder), an SFN is created. For an 8.3 name which is unique, the SFN matches the LFN. For a non-unique 8.3, you get the usual question: Do you want to overwrite? The same is true when you have unique and non-unique LFN's. The problem comes when you have a unique LFN but there is already a file in the directory/folder with the SFN that that LFN would generate.

In folder D1, you have longfilename4.txt with SFN longfi~7.txt - don't ask why, you just do. In folder D2, you have longfilename6.txt with SFN longfi~7.txt - perfectly feasible. Now you want to put those two files into otherwise empty folder D3. If the SFN's were preserved, the OS would have to refuse to put the second file into the folder - its fully qualified SFN would be the same as the first one's. However, you would have no way to repair the problem since you can't alter the SFN without renaming the LFN. So MS did the only logical thing: they did not preserve SFN's. When you write a file to a directory, if the LFN's are unique, a new SFN is generated in which the character after the tilde is invented. It's really the only way to do it.

If you have bothered to read this far, then play with the process to see how it works. One interesting variation is to write into an empty folder longfilename.txt - and then to try to write in longfi~1.txt which would have the same SFN (since the SFN of an 8.3 file is the 8.3 name). Also notice what happens when you write the LFN's in different sequences. It's all perfectly logical - but also very inconvenient for users who want to use SFN's in an LFN environment.

I would blame Microsoft for doing it wrong if I could figure out how to do it better.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Formats - Mastering and Packet

Settle back - this gets complicated. It is also based on my understanding not on authoritative sources. The results seem to work - the explanations you will see (and, I hope, follow) lead to the right sort of properties all around. However, anyone with authoritative information is invited to post to me privately so I can correct my errors.

As you will see, the physical properties of CD-ROM are significant in one area: there is no way to send the read head to a chosen spot on the disc. Instead, it must go to an approximate location, pick up synchronization bits, and then start writing based on that sync. Each time the head (laser beam) starts a read, it must do it from that space 'wasted' for sync.

The structure of the data and the TOC for a normal data disc (ISO 9660 Level 1) is really quite simple. Each file is written as a continuous stream of data; the TOC provides information on the start point and the size of the file. Mastering software (also called pre-mastering and authoring and typified by CDRWin and Easy CD Creator) supplies a stream of bytes to the disc for those files in sequence. To facilitate getting to each file, it needs to be tied to those sync bits; the effect is space wasted in much the same way as it is on a FAT16 - though the lost space is not really for clusters at all.

When you use variable-length packets in writing to a CD-R, two cases come up. If you write one file at a time, the packets (limited in size by the buffer of your drive) are written consecutively; if you have multiple, concurrent writes, each packet is written with a chunk of one file, but the next packet may be a chunk from another file. The packets are written consecutively - no gap between them - but they may be interleaved. Note that since you need to address packets separately, you need sync space for each. However, that's not usually critical since a variable-length packet is limited by the size of the buffer internal to your drive. If the file is small, it needs just one sync block. If it's big, the block is small enough compared with the total packet size to be negligible. Effectively, you're right back where you were in FAT16 or mastering.

When you finalize the session, the TOC is written in Level 3 format, which means that each file's pieces are properly connected. A runout track is also written during finalizing (closing) a session or the disc. So the net effect of variable-length packet writing is not very different from mastering when the disc or session has been finalized. Its advantage is, of course, that files can be written whenever you wish; you need not accumulate them between writes, though you need to pay the usual end-of-session penalty for the runout track.

When you use fixed-length packets, formatting is used to break the disc space into packet-sized chunks and a map is prepared to hold information on how the chunks make up the files. Every write to such a disc of a file larger than one packet is deliberately fragmented to help keep one area from being scrubbed (written, erased, rewritten) excessively. Those writes require separate seeks and complex bookkeeping on the map. So fixed-length packets take space for the packet delimiters and for the map. Writing them takes a lot of work in the CPU and memory (for the map) and space on the drive (for the seeks). As a result, the overhead includes bytes for each packet to sync the writing to the appropriate area of the disc. Finally, that map is not a FAT or even a TOC, so making it comprehensible requires that you install a translator (the UDF reader) if you want to read the disc on a conventional MultiRead drive. Since the runout track was written when you formatted, the disc needs no finalizing (in fact, none is possible) for it to work in a reader with that translator installed on your system.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Multiple Sessions

Settle back, folks. This gets sort of complicated.

Track, session, volume, disc - confusion

First, let's take a crack at understanding discs, sessions and tracks, working from the bottom up.

A track is a stream of bits with a starting point identified in a Table of Contents (TOC). In audio, a track is usually a song or other identifiable chunk of music or speech. In data, one usually writes a single track with all the data of a session; the capability is there to write multiple tracks, but there is no reason to do so and it wastes space.

A session is a collection of tracks (one or more, maximum 99 per volume) with a TOC at the front and a runout track at the end. If there is more than one session on a disc, the first will point to the start of the second and so on until the last. A session is said to be 'open' if there is no runout track and the TOC still permits adding tracks. Since an open session has no runout track, it cannot be read in an ordinary CD-ROM or CD player; it can only be used in a writer. When the runout track is added, the TOC points to it and a reader can then know when to stop reading; until then, it doesn't know. Each session (after the first) takes about 13 MB - essentially for the runout track. The first session takes about 20 MB, but that is considered when the disc is designated as "650 MB" or some other length.

"Volume" is seldom used in CD recording these days, but it's a useful concept which does show up occasionally. A volume is a collection of one or more sessions, none of which is linked to a session outside of itself. Usually, a disc has a single volume. To create a second volume on a disc, add a session to a disc with one or more existing sessions - and do not link them to one another. The result is as though you had separate discs on one, but access is a problem.

Neglecting the multiple-volume issue, a disc is said to be closed if its last (or only) session has a TOC saying that the disc is closed. On modern CD-ROMs, there is no need to close a disc except to protect it from having a confusing extra session added. It is a sort of write protection against adding sessions. In addition, some older drives can be confused by an open disc, so it's wise to close the disc when you are through adding sessions to it.

Single Session and audio players

Some early CD-ROM drives and all audio players are single-session. They 'see' and use only the first session on the disc. There could be a multi-session audio player, but it is not part of the specification and all portable, mobile and home players are single-session. As a result, once you close an audio session, whatever you write after that will not be recognized except in a CD-ROM drive. That introduces a problem if you want to drop tracks onto a disc in different sittings (i.e., at different times). While the first session is open, you can play the disc only in a **writer.** But if you close the session, then any sessions you add later will not be playable in an audio unit. You may not like the situation, but those are the rules and that's all, folks!

Critical to all of this is the fact that audio players must be inexpensive. The standard is designed for the lowest common denominator of players, a brain-dead box which can read the TOC to count tracks and their lengths and nothing more. Of course there are fancier versions which can (and do) play tricks, but the design of CD-DA is built around the mindless machines.

Multisession in various forms

Given the above, there are only a few logical ways to lay out multiple sessions on a disc - and many illogical ones with special purposes. Logically, if you're going to put CD-DA onto a disc, you put it into the first session. Put it anywhere else and an audio player won't see it. Data before audio makes sense only for CD-ROM readers, and in that case the only advantage of true CD-DA tracks is that they use disc space better than would WAV files (thanks to error correction). Such arrangements and other non-logical ones can be used on game discs where all rules are made to be broken. They're off-topic here.

They are also off-topic for some programs which may not support creating or copying discs with such layouts. If you are interested in programs for "backing up" game discs and other copyrighted materials, you may find useful information here and usable programs from major publishers, but piracy is not my purpose and you should look elsewhere for expertise on those matters.

There is a special sort of multisession disc used for Video Compact Disc (VCD); that is covered in other pages. Complicating the matter further is Compact Disc Video (CD-V); on such a disc, the video is in **analogue** format as on a LaserDisc. Because of that, it cannot be played in a CD-ROM drive or copied as a VCD would be.

Multiple data sessions

Until packet writing came along, the only useful way to write part of a data disc at a time was in separate sessions, paying the 13-MB penalty for each one after the first. Packet writing is not universal and a disc written in packets cannot be read in DOS or early Mac OS's, so multisession data discs still make sense. All modern CD-ROM drives are multisession and will read such discs without trouble. Although you should be able to mix modes and even ISO 9660 levels, safety says: don't. Data mode 2 (XA) was devised for multisession, but with today's hardware you can use either mode without penalty.

After the first session, you may either import a previous session when creating a new one or not. The simplest course when adding sessions incrementally is to import the immediately previous session automatically. That simply links the old TOC into the new one so that your resulting directory will include the old one. You may have an option at that point to delete files from the first session or to move them around among the folders. In any event, all you are moving are pointers to the data, so no significant space is wasted. Note that if you do not link the previous session, you create a new volume. The old volume (or the old session) is accessible then only with software written for the purpose, such as Adaptec's Session Selector or Nero's volume mounter.

Audio before data - CD Extra

Suppose you create a disc with some audio tracks in a first session, then an unlinked second data session. Drop that disc into an audio player and listen to it as a normal audio disc; since the second session is not visible to the (single-session) audio player, you're home free. Drop the same disc into a (multisession) CD-ROM and the audio will be ignored so you can access the data as though there were no audio there at all. You can still reach the audio by selecting it with appropriate software, but for practical purposes you have two separate discs.

When the idea was first thought up, the goal was to have enhanced players which would use the data session for menuing. A standard was developed for such a CD Extra disc which included a logo and a menu. If you create a true CD Extra disc, a folder and some files will be constructed to satisfy that standard, but they are unlikely to be of value for your needs. As a result, it's usually sufficient and appropriate to create two sessions - a closed audio session followed by an unlinked data session.

Mixed Mode

Mixed Mode is not really multisession, but it is multi-mode. The first track is data in Mode 2 Form 1; subsequent tracks in the session are audio - CD-DA. The CD-ROM drive can then see the data and the the program in the data track can easily access the audio in those which follow. The problem arises when you put that disc into an audio player. The better ones will recognize that the first track is data (the TOC says so) and it will be skipped. Unfortunately, that least common denominator drive - and you cannot know which it will be - is too dumb to know that the track is data and will try to play it. The result will be a scream of high-volume noise.

Note that if you put a Mode 1 data disc into even the stupidest audio player, it would not read it. Since it cannot make sense of the TOC, it will just complain; it won't try to make sound out of noise. A disc in Mixed Mode gets around that 'problem' - and can damage your hearing as a result. Needless to say, Mixed Mode is not recommend for the novice; it may be supported by your mastering program, but it should not be used.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Packet Writing - Intermediate

Despite appearances, the following is a simplified account of storage on a DirectCD disc. For the full story, check ECMA 119 and 167 at http://www.ecma.ch - or the equivalent from ISO if you can afford their fees. I am not the source of this information, but its author is fully knowledgable and has done an admirable job of making it understandable. If you doubt that, try the ECMA documents.

In order for data to be readable on a standard CD-ROM, the disc must contain three things: a lead-in (containing the table of contents or TOC), the data area (containing the tracks), and a lead-out (containing nothing). The TOC is NOT the same as it would be used for a file system; it only describes what is in the data area - like track start and stop positions. An "open" disc or session contains only the data area since the ultimate track layout is not known (new tracks can be added) until the session is closed. These data are stored in a temporary location called the PMA. When the session is closed, the drive gets the temporary information from the PMA and writes it in the lead-in as the TOC. Again, this has nothing to do with the actual contents of the data area. Move that disc to a CD-ROM drive, and the CD-ROM will find the TOC and tracks and is able to perform reads or to play the associated audio. In a multi-session disc, you would have multiple lead-in, data, lead-out sequences with each lead-in except the last linking via some pointers to the next. The CD-ROM drive starts at the first lead-in and follows it through to the last one to locate the current data set. A CD-R/RW drive does all this, but is able to read the PMA as well so it can find the open session data. Now, when using incremental, or packet writing, the link, run-in and run-out blocks are added whenever a packet is written; they are required due to the way that data are encoded onto the disc. It also has nothing to do with the lead-in, data, lead-out business either. Do these link blocks cause problems for CD-ROM's? Some, but MultiRead is supposed to mandate the ability to deal with them.

So how does this all apply? When using CD-R, DirectCD will write into the data area of the open session, appending packets as needed. In this state there is no lead-in or lead-out so a CD-ROM cannot read the disc. When the disc is converted to ISO-9660 format, we write the ISO image to the disc and close the session. We could simply close the session and still be readable on a CD-ROM because all a CD-ROM cares about is the lead-in, data, and lead-out. Nobody could get anything useful from it without a reader, but the drive would be happy. ISO 9660 is useful because of the wide availability of ISO readers. In the case of CD-RW, we actually format for all the data and close the session right in the beginning. That is why it is readable (in theory anyway) from the start. With both CD-R and CD-RW, the file system is written into the data area of the disc, but with RW the session containing the data is closed and therefore readable on a MultiRead CD-ROM.

With DirectCD, data are written in discrete units (packets) that may be as large the recording drive's buffer. Because of the way data are encoded onto a CD, each packet is proceeded by a series of blocks (link and run-in) and followed by a couple of blocks (run-out). One way of visualizing this is to think of the packet being encased in some "insulation". The net result is that there are some additional blocks between consecutive packets of data. For example, if we have to write a 3-megabyte file on a drive with a 1-megabyte buffer, the data will likely end up written in three discrete pieces or packets, each packet being one megabyte in size and each pair being separated by the linking blocks. In order to read the data correctly, the reading software must be able to skip over the linking blocks that separate each packet. MSCDEX cannot do this. It assumes that all files will be written in one continuous extent. When it encounters files written in multiple pieces it tries to report them as multiple files. The Win95 and WinNT readers are able to skip ahead as needed. This is really the distinction between ISO level 1 and level 3. In level 3, the files can be in different pieces (multi-extent) while in level 1 the files must be single extent.

Another wrinkle added by a file system implementation is that we are able to handle multiple threads of execution writing at the same time. This means that we may have data interleaved on the disc as a result of receiving interleaved data from the operating system. This would also require multiple extents. Finally, the fixed packet size used for RW is 64K or 32 blocks (blocksize=2K). Including the link blocks and the total amount written becomes 39 blocks (1 link, 4 run-in, 32 data, 2 run-out) or 78K. This is also why there is so much overhead for direct overwrite on RW media, approximately 18% waste!

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Why Packets?

Packet writing is new, flaky and ill-supported. Why in the world would anyone need it?

In fact, practically no one **does** need it. But it may make your life easier if you use it well. It will certainly make it harder if you use it badly.

Variable-Length Packets

With variable-length packets (yes, there should be a better term), you can write arbitrarily large files to a CD-R as they become ready to write. Each day, you download from a mainframe all of your correspondence, transaction history, presentation drafts and other information to get you started fresh the next day. That information updates your HD - but drop a copy onto a CD-R for permanent record and to be sure you can start at full speed tomorrow even if the network goes down. A few megabytes - or even tens of megabytes - need to get easy, non-volatile storage in chunks. A perfect application for variable-length packets. Or you have fifty pages to scan and OCR tonight and a batch-job program which will do it and will save both images and text as it goes. Let those pages be output to a UDF CD-R; when everything looks good the next day, finalize the disc to ISO 9660 Level 3 and file the disc. I update one WWW site each week with one to five MB of files. Each week, I archive the site to UDF so I can return to my mistakes later and apologize appropriately.

Variable-length UDF is rarely an alternative to conventional CD-R. Basically, it's the tool for what we used to do with floppies and now might do with Zip discs - if the did not cost a dime per megabyte. The price of the CD-R is nearly a hundred times less, so we can feel happier about saving history with it. But when you have truly critical data or very large blocks to deal with at once, packet writing is not your best choice for primary storage. In the example of HD update, primary storage is on the HD and the UDF disc is backup. In the case of OCR, the final output on your HD is the primary product; the UDF is a way to verify what was done and to repeat the OCR even if the originals are not to be scanned again. Once closed to Level 3, the UDF disc is as reliable as any CD-R and almost as versatile, so `wasting' the 15 MB or so to finalize should be routine.

Fixed-Length Packets

Fixed-length packets are written only to erasable blanks and provide a very different sort of environment and very different potential. Think of a fixed-length UDF disc as a slow drive like a very large floppy - and try to think of its use as we used floppies before HD's became common. (That was not so long ago for Apple users.) You can install a program to such a disc and run it from there even as it updates its files. If the program is designed to be used that way, it will be CPU-intensive, not disc-intensive, and will manage DLL's and overlays differently from the way it would on a fast HD. There are few programs today written for that sort of thing, but three possibilities should be considered for CD-R use: cleaning and compressing audio and compressing video. Those are programs of modest size which write relatively little information after a lot of processing. On most computers, MP3 compression of a redbook WAV takes about as long as it does to play the WAV. The file can be read from a conventional CD-ROM (at about 150 kilobytes/sec) and execution and output can use fixed-length packets, writing 15 KB/sec. That format is written at 2x, so the delay for slow writing does not matter much.

Using fixed-length packets where variable-length makes sense (e.g., for backups) is quite possible, but is usually a false economy. You can work that out for yourself, considering intial cost, loss of space to formatting, slow write and - most important for many - inability to finalize. Fixed- and variable-length packets are good solutions for some situations, but can be frustrating or costly if used incorrectly.

E-mail me at cd-r@mrichter.com

Return to Mike's home page

The UDF Session

Things seemed complicated enough when we were mastering a disc. Now, with UDF and packet writing, they're getting more confused. Formatting, finalizing - what's going on?

The cases are very different for write-once and erasable media - variable- and fixed-length packets. I pleaded with Adaptec to use different terms when UDF was being introduced, but I was overruled. Something about a standards committee.

When you 'format' a write-once blank (CD-R), you don't really format the disc. You simply establish the directory for an unusual sort of session. Just as you can add audio tracks to an open session one at a time, you can add data files to an open UDF session one at a time. Since the session is open, it has no runout track and can be read only in a writer. Since it is not in a format known to the OS, it can be read only in a writer capable of UDF with a UDF program such as DCD operating. (In fact, a program such as CD-R Diagnosis can also interpret the format, but that is because it does so the way that DCD does.)

When you add a file to a disc in variable-length packets, it goes to the UDF 'driver' part of DCD - a part you don't see but which has been installed when you boot up. That driver translates the file into one or more packets and writes them to the disc as a sort of 'track'. (Because the laser beam must be positioned for this task in a way not needed for mastering, not all drives will permit packet writing; that's why some drives do not support UDF.) After a while, you may want to make your disc readable in a CD-ROM drive, not merely in your writer. Then you close the session with a visible part of DCD. When you do that, you create a TOC similar to the one made when you master a disc - but not identical in form or content. Because closing a variable-lenght packet session changes the type of the directory, it is sometimes called "finalizing".

When you master a disc, a file is written in contiguous blocks - the file is read sequentially. When you write variable-length packets, the packets may *not* be in sequence. If you are copying two large files during the same period, they may be interleaved. So the format used when you close the session is ISO 9660 Level 3, not the standard Level 1 (I haven't found Level 2 yet in my reading - it may show up yet). Because it is not Level 1, it cannot be read in DOS, Win 3.x, early Mac OS's and so on. When you close a session, you have the option to close the disc. If you do, it is no longer writable. If you do not, then you can add another session by "formatting" again. That will open a fresh directory space and link the session to its predecessor. Since the earlier session is in Level 3, you should not be able to master to the disc to add a Level 1 session - the TOC would become hopelessly confused. In fact, ECDC does not let you do so; I don't know about other software.

NOTE: As you will see in the page of this primer on Psychic Software, the program does not know whether you will be closing the session or the disc and does not make allowance for it. Therefore, you can write so much to the disc that there will not be room to close the session.

Now a word on behavior with fixed-length packets. When you format an erasable blank (CD-RW), you actually do format the whole disc. You write markers to delineate the segments in which packets will be written and finish it up with a runout track at the end of the disc. As a result, the disc can be read in a CD-ROM drive - but only if that drive is MultiRead (can read an erasable) and has a translator from UDF to a format the OS will recognize. Of course, that translator takes the form of another of those 'hidden' parts of DCD. It actually reads the information about the disc's contents so that it presents the disc to the

OS as though it were a sort of oversize floppy. The translation is done in RAM - which introduces a whole set of other issues. The bottom line is that a disc in fixed-length packets is always single-session and the session is technically closed although you can still add and delete data.

Isn't it nice that this stuff is so simple?

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Compression Caution

At least two programs (PacketCD and ECDC 4.x) offer compression on fixed-length packet discs. Under the right circumstances, that can be a distinct advantage, recovering more space than UDF overhead consumes. Under other circumstances, it can be a curse.

Like any lossless compression scheme (PKZip, Drivespace), the effectiveness of compression depends on the compressiblity of the files being stored. That varies widely. Typical numbers are suggested in this table:

Filetype	Ratio
Database / spreadsheet	5-10
TXT, DOC	3-5
Executable	1.5-2
WAV	1.1
ZIP, JPG, MPx	0.9

The overhead needed for compression must be paid even on an incompressible file, so zips, JPEGs and MPEGs of all sorts (including MP3s) tend to take more space on a compressed drive than on an uncompressed one.

A minor factor on most systems is that compression and decompression take processing time. That is seldom significant compared with the time needed to store the data, but it can be noticeable in some cases. A more serious issue is that the compression scheme is not standardized. If you have one system on your computer, you may not be able to read a compressed disc from another, but you will be able to read an uncompressed one. And while you should not try writing a disc formatted by one program in another, you **must** not attempt to do so if the disc is compressed.

By far the most serious implication of compression comes up because the software is not psychic. The only way it knows how much compression will be achieved is to do it. Theoretically, it could compress everything to some work area, measure it and report back to the user before writing to the disc - but can you picture any user putting up with that? The effect of the compression program's ignorance is that an initial space estimate may say that you have a gigabyte available on the disc, but if you write 520 MB of Zips, JPEGs and MP3s, you overfill it and it will not be usable.

No one is cheating you out of anything. The program is working as well as possible. But if you ask the impossible - accurate prediction of effective compression - you won't get it. With any recording process, it is wise to be conservative, to avoid using the last few bytes or even megabytes of storage. With compressed UDF drives, even more conservatism is justified. Once you know how much compression you can achieve on your typical files, you can decide whether to use the feature or not.

If you use compression, please do so with care.

Come to think of it, care is a valuable contribution to any CD-R effort. It never hurts.

E-mail me at cd-r@mrichter.com

Return to Mike's home page

Buffer Underruns

Most of the information here is also in other parts of the primer, some in greater detail. Still, the questions about buffer underrun - sometimes called "failure of streaming" - persist. So, let's consider it as a separate matter in order to help you diagnose it if it occurs and to cure it.

"Buffer underrun" simply means that the buffer supplying data to the writer has emptied before the data are completely written. Since mastering a disc requires that the burn of any track (at least) be continuous, interrupting the flow means - coaster time! (Because of the way that packets are written, buffer underrun cannot be a problem there.)

There are two sorts of buffer involved in CD-R: one within the writer and one which the mastering software may create from system RAM. Not all programs do it. Recent versions of CDRWin, ECDC and WinOnCD certainly do; others probably do as well, but you may want to check your preferred program. In a reasonably well configured system, the amount of buffer made from system RAM is much greater than that within the drive. Even more to the point, there is no feedback from the drive to tell the software how much buffer is being used. The result is that if you have a display of buffer level, it is reporting only the system RAM being used. That may go down to zero safely - you still have whatever is in the drive - but usually if it drops, complete underrun (which will be reported to the software) is only seconds away and unavoidable. For that reason, if you are trying to analyze the buffer underrun, it is useful to record the way the buffer behaves before the fact. Fortunately, underrun is usually found during test as well as while burning, so it can be analyzed without wasting a blank.

The normal pattern for buffer level is that it will build up quickly before the data writing begins, then hold steady - typically well over 90% - until just before the last data are to be written. Then the buffer drains to write the last information to the disc. Remember that pattern, it will be important.

One way to prevent a buffer underrun is to make the buffer so large that it cannot be emptied during the burn. Since that would mean something like a gigabyte of RAM, that's not the solution most of us choose. An alternative is to use the hard drive as a buffer - which will usually work if the problem comes when recording from another source, such as a CD-ROM - and we'll see where that is used in what follows.

Okay, if we can't put the whole disc into RAM, the objective is to keep the buffer filling more or less as fast as it is emptying. Since we're writing at between about 150 KB and 2 MB per second (1x data to 12x audio), that may be no problem at all or a severe burden on the system. If you are simply writing too fast for your source to keep up, the buffer level will ramp up to 100% initially, then gradually fall to zero - and continue to fall invisibly as the buffer in the drive is drained to produce an underrun. The solution here is either write at a lower speed or get your source to run faster. Depending on how you have configured your system, there are various ways to make the source drive run faster, but they may be more expensive or less convenient than you like. The simplest way to get your hard drive to run faster is to avoid using it for any heavy lifting while you burn. If you're close to your system's speed limit, shutting down other programs may be necessary.

What can slow down your source drive? If it's a hard drive, fragmentation will do a really good job of that; see the page on Arithmetic 101 for the effects of fragmentation and of dealing with large numbers of files. The fix in that case is simple: defrag the drive and (if you have a lot of files) write from an image.

If you're reading from a CD-ROM instead of a hard drive, you need to check both the nominal speed of what you're trying to do (for example, a lot of "high-speed' drives only do DAE relatively slowly) and the actual speed given the disc you're trying to read.

Three factors typically slow the reader below the speed you expect it to deliver. One is consistently high error rates - typical of a CD-R or a scratched or dirty pressed disc. Rereading is very slow; even a modest error rate can cut the effective speed of your reader, even reduce it below 1x. That is not known to the software until the burn begins, when nothing can be done about it. If you suspect that cause, you can check the disc with CD Speed or simply make an image (where the read speed doesn't matter) and burn from that. If this is the cause of the underrun, the buffer level will fall gradually, but usually not smoothly, in a pattern similar to that of simply having too slow a source drive.

The second factor which can slow the CD-ROM reader is a severe scratch or similar localized damage which causes it to read well most of the time but to have trouble in one area. In that case, the buffer will stay up until that region is encountered, then fall rapidly as the reader re-reads to try to get valid data. You will see the read light flash as you watch the buffer drain. You can try writing from an image, but more often you will have to repair or replace the source disc.

The third cause is that the source drive simply stops for a period; when it starts up again, if it does not reach operating speed quickly enough - underrun! Most high-speed readers have a spindown time to save the bearings. If the drive is not reading information for that period, it is allowed to spin down, then when queried it will spin up. A CD-ROM being used for copying is not reading between sessions of a multisession disc. The pattern here is that the read light goes off while the buffer drains at the end of the first session - and the drive begins to spin down. You can deal with that problem in several ways: increase the spindown delay; write at higher speed (so the time to change sessions is reduced); write the second session to your hard drive and copy the first from the CD-ROM, the second from the hard drive; make the copy in separate sessions, one for each on the source; and so on. (To find out about changing spindown delay, check with the manufacturer of your reader. Using unapproved methods can damage the drive and void the warranty.)

Note that a similar condition can occur on a hard drive if its sleep time (typically, on a laptop) is set too short. There's another way that the buffer can be drained while you write from the hard drive: some other program may come along and override the high priority the mastering program set. What sort of program would do something that stupid? FastFind, some anti-virus monitors and other programs which insist on reading all your drives - including the one you're writing. Since the drive cannot read and write at the same time, if a very high priority read command is received, the writing will stop and the buffer will be drained so the read can be performed. The solution is simple and has no known alternative: kill the monitoring program.

In case you have not guessed, these are not all the ways that a buffer underrun can occur and far from the list of ways you can cure the problem. However, they should be enough to help you find out the cause of your problem and to fix it. Since you are the person who sees what's happening, you are in the best position to remedy any difficulties; fixing it by remote control through e-mail or even voice is extremely difficult.

E-mail me at cd-r@mrichter.com

Return to Mike's home page

Firmware

Firmware is the program that's on a chip in most devices you use with your computer. It is definitely in your CD-R. Originally, such firmware was stored in Read-Only Memory (ROM) of a flavor called Programmable ROM (PROM). The information is hard-wired into the chip and can be changed only by changing the chip itself. In a later version, the chip could be burned (actually burned in the sense that metallic links could be blown out) by relatively high currents. That made it a lot cheaper to upgrade the information. Of course, by now we were writing to read-only memory, but that's computerese, not language. It got worse when people figured out how to avoid actually burning the metal and developed EPROM (Erasable PROM using ultraviolet light) and EEPROM (Electrically Erasable PROM). Still, it is a way of storing information into your hardware on how the device is to listen to and to respond to the commands it receives. It is, in a word, the sole storehouse in the device of information on what to do with inputs from its host.

The PROM in your CD-R holds firmware. If the manufacturer supports doing so, it may be written from your computer instead of needing to be done by a service center. To do it, the computer must tell the adapter how to supply the signals needed and the card must have the higher-current supplies ready to drive it. Some cards do and some don't; some manufacturers can and will let you do it, others don't. The key is that if you burn the wrong information into your device, it will lose the ability to listen to the computer to get the right information in. There is no way around it but to have it serviced and if you have burned really bad stuff, the manufacturer and service people may simply say: Tough luck - we told you not to do it and you ignored us. You voided your warranty and any implied contract, so it's your problem. Next time, follow the rules.

One brief note: there are two components to the upload you will try to use. One is the binary data which go to the device - your CD-R. The other is the loader which is the program that tells the adapter what to do and how to do it in order to get the information into the device. When the loader runs, you will see some strange stuff on your screen: a version number that has nothing to do with the firmware version, apparent error messages which are not errors, etc. None of that matters. When flashing is complete and you reboot, you will see the new firmware revision as the SCSI buss is polled. At that point, you are free to sigh with relief.

Nothing else that you do with a CD-R has the potential to do as much harm as burning firmware. If you use a program from an unsupported site, you are taking the risk. If you are very lucky, it will work. If you are reasonably lucky, it will do no harm. If you are typically lucky, you will still be able to burn the right firmware when it comes out. If your star is not in the ascendant, you may have made an expensive paperweight.

Software from an unauthorized site is the stuff of Alice in Wonderland. Eat it or drink it and something dramatic will happen. Sometimes, it isn't really bad at all.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

The ASPI Layer

In the early days of CD-ROM, every manufacturer had his own interface and connections to the host. Standard interfaces, ASPI and CAM (which will operate ASPI devices and is functionally equivalent for Windows platforms) were devised and manufacturers of SCSI devices either conformed or wandered out of the market. The situation with IDE devices was similar, but the standard that was developed for them (ATAPI) made them look to the system like SCSI and allowed them to work through ASPI. For that reason, your computer's IDE adapter and devices may show up under the SCSI controls. More information on SCSI and drivers can be found at
http://www.europa.com/~diogenes/DSMFDSM/#CAMT
http://www.lionsgate.com/Home/Baden/public_html_index/SCSI/SCSIgame.txt

The ASPI Layer is a collection of programs to realize the ASPI interface. Those DLLs work together and must be consistent and complete to avoid problems in communicating and interpreting commands. Some vendors package a subset of the DLLs with their products - scanners are notorious for this.Often, they will grab an old file (one that's good enough for their needs) and package that with their own software - but with the then-current date. It will then install even though a later version is on your system and it will not be removed readily, again because the date is misleading. (Okay, it's a lie!)

Note that there is an approach to interfacing the components via miniport drivers which does not require the ASPI layer; there is none in Windows 2000, for example. In general, CD-R gets along fine without an ASPI layer if the code is written appropriately for Win2K. However, code which is not fully Win2K-compliant may use an ASPI layer instead. That can work - if the layer is complete and consistent. Many support programs do rely on the ASPI layer, so one is needed for them even if not for programs such as Easy CD Creator 4.02 and above.

ASPICHK from http://www.adaptec.com will analyze your system and report the consistency and version numbers of your ASPI components. If it reports a problem, you may try Adaptec's repair program - but that runs into the date problem and may not resolve inconsistency. At this writing, the latest ASPI layer (1021) appears to be solid. However, it is subject to corruption if some device is installed which replaces parts of the package.

The variety of CD-R failures due to a faulty ASPI layer is remarkable and defies detailing. Any repeating error message dealing with communications to or from your writer should trigger a verification with ASPICHK. It sometimes appears that programs that really do not need the ASPI routines install them (badly) just to be sure. With ASPICHK and the Adaptec files, ensuring a working ASPI layer is painless. To steal from an American commercial: Just do it!

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Labelling

Now that you have produced your shiny silvery or golden disc, how do you mark it? By this time, you are probably not surprised to learn that that is not a simple question. Before going into the specifics, some words of warning are needed. **DO NOT** use ball point, Post-It notes, or any conventional glue or tape.

Construction

The bottom of the disc seems intuitively to be the sensitive part, but 'tain't so. The reflective layer just below the upper surface is far more delicate than the thick plastic on the bottom. Your reader will make quick work of a scratch or dust speck on the bottom, but anything which disturbs the metal foil will make the disc usesless.

In the least expensive discs, the upper layer is little more than a plastic varnish over the foil. Solvents used to suspend ink and hard tips used to lay the ink down can wreak havoc. Some discs have a more durable coating applied to the top - various names are used to suggest that they are less vulnerable to damage. They are still fragile, but not to the extent of the bare discs. In general, a printed surface suggests a stronger one and one designed for you to write on the disc (underlined spaces for date, etc.) is a pretty sure bet. Still, a soft nib and water-soluble ink are desirable. Finally, there are discs made for ink-jet printers. They have thick, usually painted surfaces and can be written much more freely.

Pens

By now, this is easy, right? You want a soft-tipped pen with water-soluble ink for complete safety. The TDK version is available from several suppliers of bulk discs for about $3; Apogee has another at a similar price. An art supply store can probably provide you with a range of altenatives in different colors and line widths.

It's inevitable that people ask about the excellent Sanford Sharpie line. Simply put, there's been no report of failure from the use of Sharpies, but the ink is not water-soluble so there's a chance that it's a problem waiting to happen. I use the standard Sharpie all the time on coated discs, but even though the maker of my inexpensive discs assures me that the Sharpie will work, I haven't chosen to try it there. I figure that the TDK costs me a tenth of a cent or so per disc and at that rate I can afford to splurge. <G>

I will offer one warning on the Sharpie and others: be aware of the tip and its potential for problems. Sanford's Ultra Fine Point is surrounded by a metal collar. If I want a fine line, I use the TDK with its semi-rigid tip. Perhaps I'm being overcautious, but I see no point in risking data loss for something as simple as this.

Labels

There are several manufacturers of labels and of positioning devices for CD-R. Avery's 5824 are unique in two respects: they have an ingenious positioning scheme which means you have high reliability without an additional hardware; and you can get them from any competent stationer. They are expensive so I recommend them for those who only want to use labels on rare occasion. They are also a good starting point for beginners since Avery provides templates for many programs and supports the 5824 with their Label Pro software.

There are also branded labels from Neato, Stomper and others plus unbranded ones from independent sources. There are two essential issues in selecting such labels - but only one can be determined easily. That is whether the printing surface is suitable for your printer and purposes. If you want to print photographic quality with an inkjet, you may have real problems finding suitable stock. The other problem is the adhesive. You would think you can use some sort of super glue - but the adhesive, like ink, must not contain any solvent which would attack the plastic or seep through to damage the reflective layer.

The adhesive bonds the paper label to the disc's upper surface - the one that's so vulnerable to damage. If it does not hold tightly enough, the label can begin to peel. You cannot remove it (that would pull off some of the lacquer layer and probably destroy the reflector and the disc); you cannot press or glue it back down again; and you cannot use the disc with the label pulling off. The best choice is to try to stick it down long enough to make a copy.

If the label is not well centered and reasonably flat, it will unbalance the disc - particularly in a high-speed reader. You get one and only one chance to position the label and if your eye or your hand is off, you may turn a perfectly good recorded disc into a coaster. For that reason, the major makers of labels sell positioning devices. Once you have graduated from Avery and have selected the brand of label stock you want to use, the positioner is a wise choice. Note, too, that labels vary widely among manufacturers and even from a single source. As a result, you may need to find the right label stock for your needs.

Direct printing

There are several vendors with printers adapted from low-cost inkjet models. I have not used any of them, but the reports are consistent: they work well for a while, then critical parts wear out from the heavy load of a CD-R where the design is intended for paper.

There are at least three quality direct printers. Primera (formerly Fargo) and Rimage have been in the business for some time and Marcam has recently appeared as a contender. However, these devices are costly ($1300-$4000 list).

Manufacturers of bulk CD-R systems seem to prefer the Primera Signature (relatively low cost) for high-volume applications. It uses standard ink-jet cartridges and delivers modest resolution on the disc. Note that the silk screening used on pressed discs offers only about 150 dpi; while 300 dpi may be marginal for a letter, it looks good on a CD-R. The Primera prints onto special discs for which a premium of the order of ten cents is typical; that means that they are desirable for both cost and quality when volume is high enough.

The Rimage printer is much more costly still but uses wax transfer and can print on any disc surface. It is built with metal where Fargo uses sturdy plastic and seems suitable for very high volumes.

If you are patient and watch carefully, you may be able to get a good deal on a CD-R printer. For example, Primera recently upgraded their Signature product and offers refurbished units at a substantial discount from their list prices. Nevertheless, it takes a substantial printing load to justify $1,000 or more to replace a $3 pen or a package of labels.

Software

Check out DISCUS at http://www.magicmouse.com/ for an attractive and inexpensive program for direct writing to the disc, labels and inserts. To capture the titles or tracks, your choice of mastering program is the best starting point.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Psychic Software

One of the limitations of CD-R software is that it is not psychic - in particular, it is not able to determine what you intend to do or what you will do; it is neither ominscient nor prescient.

Suppose you are making a layout, say of data. The mastering program does not know whether you will be adding an audio layout to make a CD Extra or Mixed Mode disc. It does not know if you are going to make an image or if you plan to burn to a 74-minute blank, an 80, or to one which is already partially written. In fact, when you make up the layout, even you may not know what you will do with it; you might think you're going to burn a standard blank but then decide to add it to an unfinished disc or to use an 80.

While prescient software might decide for you, the stupid stuff we use does not. Instead, it **estimates** the amount of space required based on the most common assumption: you're making a layout which you intend to burn to its very own standard blank. Even if you already know what you're going to do with the layout, the software is not privy to that information. As a result, it only does what you have told it to do (make the layout) and informs you of what it knows (how it would fit on a standard blank).

In another situation, you might be writing fixed-length packets to a disc. The software knows how much has been written and how much you are about to write, so it can tell you if there is room enough for the files you are adding. But it has no idea of when or whether you are going to finalize the session. If it anticipates that you will finalize, then it should lie to the OS and claim to be full when it would no longer have space for the runout track. If it knows that you won't want to read the disc in a normal drive, then it should tell the OS the truth. Lacking psychic powers, it tells the truth. The result is that you can overfill the disc - write so much that you cannot finalize it. Heck, you knew all along that you wanted it to be readable in other drives; what kind of dumb program knows less than you do?

E-mail me at cd-r@mrichter.com
Return to Mike's home page

SCSI or EIDE?

As I frequently reply to private inquiries: I use EIDE for hard drives and SCSI for everything else, so I am not able to advise those who prefer EIDE for CD-R or CD-ROM. Let me begin with an observation. Many users have switched from EIDE to SCSI for CD-R and CD-ROM. Some had a few initial problems with the move, but none has reported going back. I have seen none who went the other way and stayed with EIDE. It is not that SCSI is 'better' than EIDE, parallel port or USB. The fact is that the SCSI interface was designed for the sort of operation that CD's demand and is well suited to it.

CD-ROMs began on SCSI as did scanners, removable hard drives and tape drives. The SCSI interface manages the devices attached to it for the computer, offloading processing from the CPU and minimizing interactions among those devices. I have been asked to help people who hang scanners, printers and capture cards from a parallel port or who try to make CD-R, CD-ROM and hard drives work with EIDE. I anticipate similar questions on USB as soon as the interface becomes more popular. On occasion, there is a solution which satisfies all demands, though often it requires a high-powered EIDE adapter - and is still not as satisfying as a SCSI interface which would have cost little or no more. (In general, any PCI card will do for SCSI, though extremely high speed readers will benefit from a fast one.)

If I am asked for a recommendation on a new system, I always suggest SCSI. That's because I anticipate that a user who initially expects only to add one or two peripherals will inevitably be attracted to add others - and run into problems. Then switching to SCSI is costly (replacing existing peripherals). One complication is that the demand for cheap solutions has driven low-cost SCSI devices off the market; the products that remain are addressed to the committed user and the professional.

Many users have succeeded in running two HDD's, CD-ROM and CD-R on EIDE, but not without experimenting with primary/secondary, master/slave and DMA settings. I sit back in amazement at their willingness to spend time and effort and rejoice in knowing that my 2910 card happily supports Yamaha and Ricoh writers, an Ultraplex reader, Jaz, Zip, ScanJet and, until recently, Sony DAT.

In mathematics, there is a term for the process in which many are caught up: local optimization. At each step, with each addition to the system, the 'best' way to go is apparent. Yet after a number of such steps, if one stands back to look at the whole configuration, the result is far from optimal. It is hard to recommend that someone move to SCSI from EIDE, but those who jump to that interface always seem to wish they had done it sooner.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

All Things SCSI

The following is in response to some of the many general questions that arise from time to time on the Adaptec CD-R users list regarding SCSI. The information is not definitive and I claim no responsibility for it being 100% correct but I feel it covers most areas with sufficiently accurate information to make it interesting and therefore worth reading.

AHA2940UW. You must not use all three connectors at the same time. If you do so you will violate the SCSI spec. by turning your SCSI bus into a T. Only use two connectors at any one time.
Technical Note. The SCSI Bus is an end to end bus, i.e., it does not support branches. There is no allowance within the SCSI spec. for SCSI Bus branches (although there is a stub allowance). Therefore, as the 2940UW and other SCSI controllers utilize 3 connectors, for connectivity purposes only, you must not use all three at the same time. All three connectors are physically wired to each other, no buffering or re-clocking circuitry is present between the connectors. For the AHA2940U2W and the new AHA2940UW Pro you can use all three connectors, in fact on the U2W you can use all 4. This is due to a SCSI extender chip, the AIC3860 fitted to both cards. This little device creates an isolated extension of the main SCSI bus allowing you to use the extra connector plus electrically isolate devices between the main bus and the extended bus.
Technical Note. The SCSI bus is still one logical bus on the U2W and UW Pro, therefore you still must ensure that each device has its own SCSI ID and that each bus, primary and extended, is terminated correctly.

Termination. Only terminate the ends of the SCSI bus, not the middle or only one end but both ends of the bus. Do this either by enabling terminators on the device or by terminating the cable/external connector.
Technical Note. For single-ended SCSI there are two types of terminator, Active and Passive. Passive Terminators are to be found on older SCSI I type peripherals and can be identified as a row of three Single In Line (SIL) resistor packs, normally stamped with 220/330. This refers to the resistive value of the packs in Ohms. Due to their nature of operation, passive terminators should only be used if the speed of the SCSI bus is no greater than 5MB/s. If the speed of the bus is above this as in Fast SCSI then active terminators should be used. Active terminators give a cleaner signal on the SCSI bus and are therefore suited for use at higher bus speeds.

SCSI IDs. Try to leave 0 and 1 for disk drives. It is not strictly necessary to do this but traditionally 0 and 1 have been reserved for disk drives and it won't do you any harm if you continue this tradition.
Technical Note. SCSI IDs have a priority order; if two devices wish to gain access to the bus at the same time the device with the highest priority ID will gain access first. SCSI ID 7 has the highest priority followed by 6, 5, 4, 3, 2, 1 and 0, then by 15, 14, 13, 12, 11, 10, 9 and 8 for Wide SCSI devices.

Wide/Narrow SCSI. You can mix wide and narrow devices on the same bus but you may need to use special converters/terminators like the ones from HYPERLINK http://WWW.TMCSCSI.COM Try to convert from wide to narrow, that way is easier to implement and the wide devices can run ... well. wide. Keep the SCSI ID of the controller at ID 7, moving it above ID 7 will result in the narrow devices being unable to see the controller. Adaptec and other controllers negotiate for Wide SCSI transfers on a per device basis, the wide part only being used during actual data transfer.
Technical Note. Address lines are used on the SCSI bus, one for each ID. If you have a narrow device it

will not be equipped with the necessary address lines to see above ID 7.

Disconnect/Reconnect. This feature is normally enabled; it allows multiple devices to overlap transfers/commands on the SCSI bus and therefore maximizes the available SCSI bus bandwidth; its what makes SCSI suitable for multitasking environments. However, it can also cause a good run in coaster manufacture. Try disabling it as part of your normal trouble shooting. This setting can also be set on a per device basis on some SCSI controllers.
Technical Note. The term disconnect refers to a SCSI device's ability to disconnect **logically** from the bus. This is not a physical disconnect!. Allowing devices to logically disconnect frees up the bus to allow other devices to gain access and perform data transfers and makes more use of the available bus bandwidth. Its benefit is appreciated most in multitasking environments with multiple peripherals attached to the bus.

Synchronous Transfers. If this were a perfect world then every device ever produced would handle the request for Synchronous Transfers correctly. However, it isn't and they don't. If you are trouble shooting, then try disabling this option in the SCSI card's BIOS. CD-ROMs, CD-Rs, Tapes and Scanners have all been known to handle the controller's request for Sync. Transfers poorly. Today's hard drives are immune to this but ... well, just beware. Again this option is enabled on a per device basis so this allows devices to run at different speeds on the SCSI bus.
Technical Note. Synchronous Transfers are negotiated between devices just like Wide SCSI transfers. Different rates can be supported on the same bus at the same time. The negotiation involves a speed parameter defining a minimum period of time between the leading edge of successive REQ (request) or ACK (acknowledge) signals plus an offset parameter for the number of REQs received by a device before it has to respond with an associated ACK. Each REQ-ACK handshake defines the transfer of one byte or one word of data transfer. All Data transfers on SCSI can be performed at the full burst rate of the bus if that rate has been negotiated beforehand between devices, however command, message, status phases, etc. are performed in asynchronous mode no matter how fast the burst rate of the bus is.

Ultra SCSI & Ultra2 SCSI. Ultra SCSI is another option that can be enabled on a per device basis. Apart from increasing the speed of the bus, it has also increased the number of headaches experienced around the world by users and Tech. Support people alike! Theory goes that you can mix Ultra and non-Ultra devices on the same bus. Many people do and it all runs OK for the lucky ones. Because of the cable limitations imposed by just one device running at Ultra SCSI speeds, my suggestion if you do have issues is to A) first make sure your cable is A1 OK; B) make it as short as possible; C) separate the Ultra devices from the non-Ultra by installing another card. Ultra2 SCSI, on the other hand, is far easier to implement. It offers another step forward in SCSI speed to 80MB/s burst and also offers an electrically different signal scheme called Low Voltage Differential or LVD. This signaling scheme suffers none of the major issues seen with Ultra SCSI (short cable length vs number of supported devices, etc.) but it is electrically incompatible with Ultra SCSI devices in its native form. This becomes a non-issue as Ultra2 devices employ a scheme where they can detect if they have been installed on a non-LVD bus (Ultra or FAST) and revert to the maximum speed and electrical characteristic of that bus thus making them backward compatible with previous generations of SCSI.
Technical Note. The term Multimode I/O cell is often used when referring to LVD devices' ability to detect what type of bus they are attached to. However, since LVD devices are generally not compatible with the SCSI High Voltage Differential bus, only HVD devices should be used on a HVD bus.

Cables. Not all cables are created equal. Ultra/Wide and Fast/Wide SCSI cables are different. They have

different specifications. Make sure yours comply with whatever you are running. External cables and connectors can cause you issues if they are not up to spec. You only get what you pay for! It is asking too much when you spend $100s on the latest all singing/dancing PCI controller and expect it to drive a device via a substandard cable.
Technical Note. Ultra2 cables are essentially the same as Ultra cables. OK, they are not the same! But if you have a stable working Ultra SCSI system and wish to move to Ultra2, then the cables you currently have will work in most cases, the only reason to upgrade would be if your system was on the tolerance edge.

Bus Mastering. Bus-mastering PCI cards greatly ease the worry/pain of CD-R work. Sure they are not necessary and many people do use ISA cards and non-mastering controllers, like the early AHA2920 but at the end of the day they are affordable, like the AHA2910 / 2930 and give you that good piece of mind feeling.
Technical Note. Bus Mastering or Bus Mastering DMA (Direct Memory Access) refers to an I/O device's ability to control a data transfer directly into your computers main memory without CPU intervention. This makes better use of the available CPU bandwidth by not saturating the CPU with interrupts that a non-mastering card would generate. Today all PCI SCSI cards feature BM DMA; it is a specific requirement for the Microsoft PC99 initiative. The early Adaptec AHA2920 was a re-badged Future Domain PIO (Programmed I/O) Card but all later Adaptec cards feature BM DMA. ISA cards are to be phased out as part of the Microsoft PC99 initiative (PC99 Basic Requirements, Chapter 3, page 66 Section 3.28). This means that all new systems sold after 1 January 2000 should be sold without an ISA slot or slots if the system vendor wishes to use the PC99 logo.

Fixes. If it isn't broken then don't fix it!

If you have anything to add to the above, correct or dispute then please email me in a constructive manner at nealg@freenetname.co.uk You may distribute the above as you wish, all I ask is that you acknowledge the Author !!
Neal Gibbons

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Remembering History

This note explores some of the effects of standardization of the CD formats and the issues raised since the standards were established. If someone will supply specifics, I'll be glad to include them in an update. A very knowledgable friend has added some notes, offset here. Note that the driving force in what follows is the audio CD; CD-ROMs were forced to the same sort of parameters for compatibility.

It may be difficult to believe, but the Compact Disc did not even exist in the 19th century. In the early 1970's, Sony, Philips and a number of other companies began to plan for a commercially viable medium for digital audio. Even I had some slight input in arguing (through AES) for a higher sampling rate than the 44.1 Ksps which was standardized. Like other aspects of the design, the sampling rate was a compromise dictated by conflicting desires for maximum playing time, highest fidelity, fabrication ability, and costs on several levels. Note that at this stage, there was no concept of home recording and no prospect of the technology we use today on a routine basis.

> 44.1 Ksps was the highest sampling rate possible in mass-producible silicon in the late 70s and early 80s, when the standard was set. In fact, 16 bits of resolution turned out to be rather more difficult than was expected -- the first Philips players had oversampled 14 bit D/A converters. A sidebar: folklore says that Akio Morita, the founder and chairman of Sony, specified a 74 minute playing time so that the Beethoven Ninth could play without interruption. However, according to Kees Schouhammer-Immink, Philips' top scientist on the team of Sony and Philips engineers which designed the CD (and who is a friend of mine), that story is revisionist fiction. The diameter of the CD was originally specified to be the diagonal dimension of the compact cassette, and was then rounded to an even number. Given the sampling rate, bit depth, error correction and practical optical density at the time, the result was a playing time of 74 minutes. But that isn't particularly sexy, is it? Another sidebar: because quadrophonic equipment and software were being sold to consumers during the time that the CD specification was written, the original spec allowed for four-channel discrete audio, halving the playing time. Quad failed commercially before the CD was released, and no four-channel CDs were ever produced. [MR: The bit is still there to implement quad some day.]

The critical component in the CD was and is the laser used to read the disc. If we had solid-state lasers when the spec was being developed, I do not recall them. I believe we were still using the gas lasers which were in commercial distribution in the early LaserDisc players.Whichever sort was the target, it would operate in the near-infrared at a wavelength of about 0.8 micron. (For reference, visible light is in the range from .4-.7 micron.)

For the Compact Disc to be practical, commercial-grade manufacturing would have to be used - upgraded to some extent, but not qualitatively changed as for instrumentation in scientific laboratories. That dictated a spot size for reading of twice the wavelength - 1.6 micron. In order to position the spot, they devised the idea of a spiral pressed into the plastic. The spiral groove had a pitch not tighter than 1.6 microns and a superimposed modulation - a wiggle - to allow the laser to servo to a desired position. In order to establish the position of the laser, leadin and leadout (runin/runout) standards were set for the start (center) and end (edge) of the disc. Now the readers could be manufactured in accordance with those constraints and details such as the minimum radius for the start of the track and the maximum radius for its end. Over time, manufacturing technology advanced so that the tolerances could be held at very low cost and players that at first sold for a few hundred dollars are now available - with extra features - at a tenth the price.

Let me step back a few years to the first commercial use of a pressed optical medium, the LaserDisc (LD). In some respects, an LD is like an overgrown CD. It breaks the signal up into frames and records them around the disc in the form of pits in plastic which are read by overlaying a reflective layer. The most significant difference is that the LD is written in Pulse Width Modulation (PWM), where the length of the pit is the analogue of the signal; a CD uses Pulse Code Modulation (PCM) in which the pits embody a digitized signal. Another difference is that the twelve-inch diameter of the LD dictated enough thickness for two platters to be sandwiched together. The CD is smaller and thinner, so it cannot be double-sided.

> It could have been specified to be double-sided, but the process of bonding the two halves of a laser video disc at that time did not lend itself to efficient manufacturing and was not fully reliable. Further, the playing time was judged to be adequate without requiring manual or automated disc flipping. [MR: In the event, the choice was wise since it avoided the infamous Laser Rot.]

Part of the standard is the optical location of the reflective layer from the (physical) bottom surface. The polycarbonate medium for the disc was and is the best choice for cost, produceability and transparency; significantly, it has the highest practical index of refraction so that the optical distance was the greatest practical multiple of the physical distance between the layers. Given the properties of the plastic and the specified focal distance (from the optical path length), the thickness of the plastic is defined. Above the metallizing is a lacquer to seal it, then optional printing up to a maximum specified thickness. (The first players used top loading and could have handled arbitrary thickness, but the specification set a limit which was less than twice the thickness of the plastic.) As a result, players have evolved with other loading mechanisms that will not tolerate excessive thickness; some slot loaders will balk at a disc with a paper label applied.

That is the simplest demonstration of our limitations in exploiting advances in technology in CD. If we have a tighter pitch, the many inexpensive players in use today will be unable to maintain tracking and will balk. In fact, we have that today with discs which exploit the full specification and run up to eighty minutes of audio and some players which cannot track them. Much longer discs could be produced now by using a shorter wavelength. In fact, they **are** available. We call them DVDs. Of course, a DVD exploits other capabilities the CD does not have: two sides, multiple layers, and a variety of formats for audio, video and data among them.

Other possibilities which would be allowed by modern technology but are not in the standard are to start the spiral closer to the hub, to use shorter runin/runout tracks and to write closer to the edge. Each of those is feasible but would obsolete large groups of players old and new. In fact, with overburning, you can push the limits. Two factors come into play: the diameter of disc which is usable and the length of runout track.

Onto the wavering spiral pressed into a blank is impressed another modulation, the ATIP. On a CD-R(W), the ATIP reports the length of the spiral, the nature of the recording medium (type of dye or the alloy for an erasable), intended writing speeds and so on, including the code indicating royalty payment for the Digital Audio blanks used in standalone recorders. Except for the length of the spiral, none of the information is certain. A stamper made for one manufacturer to be used with one dye type can press blanks for another manufacturer using another dye. Even the length of the spiral is only a minimum; the actual length may be greater and the extra length may be usable - but the manufacturer does not guarantee that. The length in the ATIP is the point at which the spiral reaches the spec limit for maximum writing radius. The spiral almost certainly goes beyond that and the dye covers all the way to the edge, so why not use more? Simply, the writer mechanism is designed to go to that limit; it will go farther, but at some point it will fail, perhaps catastrophically. The reader will also reach a stop - a limit of some sort. That may simply mean that it will not read farther or it may suffer mechanical failure. Some writers and some software permit overburning - writing beyond the limit in the ATIP - and some do not. If you overburn, you now know the risks you take.

Just to wrap up, we need to look at the runout track with respect to overburn. Since a reader does not use the ATIP, the runout track provides a necessary reference for it to know where the disc ends. If a reader cannot find the runout track - it's missing or too short to be recognized - then it cannot read the disc. The runout is the last thing written; if you overburn, the runout may be eliminated, shortened or written beyond the farthest point the reader can track. Again, as you shorten the runout, you increase the risk that some reader will not be able to handle your disc.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

The Intertrack Gap - and Burn Proof

This tale must begin with a bit of CD recording history. Another aspect of the tale told under Remembering History in this primer is the effect of writer tolerances on some of the features and limitations of modern recording.

One part of the specification not addressed here is the requirement for a two-second (150-block) gap before the first track of any CD-R. Instead, we're looking at the space between two tracks, particularly (at first) on an audio disc. Why are those two seconds "wasted"? Simply because the tracks are written one at a time.

When CD-R was being defined, the problem of positioning the laser was severe. It was simply not possible to tell the laser to go to a specific point and to expect it to find that point on the inscribed wavering spiral. That's still true, but the uncertainty in positioning and the amount of information needed to get synchronization is much less with modern production techniques than it was twenty or so years ago.

The original mode of writing audio was Track At Once - TAO. A track would be written, the laser would turn off, the next track would be prepared, the write would be readied by servoing the laser to the point where the write was to begin, the laser turned on and the next track was written. To be sure that the track would begin as intended, the space from its predecessor was set to 150 blocks.

Then Disc At Once was added to the capability. In DAO, the TOC is written first to indicate where each track is to be regarded as starting. Then the laser turns on and burns all the data as a single track with no interruption where the tracks are separated and no required gap. Note that some players have more trouble finding the start of an interior track on a DAO disc than they do with TAO - the reader would like that gap for synchronization and on a disc which it does not read easily it may have to seek. That's particularly true for older audio players, but it can happen even with a new one with a poorly matched medium.

Someone then realized that a gap might still be desirable between some tracks in DAO, so an adjustable "gap" was introduced; it's not really a gap, but a specified period of silence introduced before the next track starts. Some software will also allow you to adjust the intertrack gap when writing TAO, which is useful for drives which don't support DAO, but is both imperfect (there is **always** a gap) and in violation of the specification.

So, what does this have to do with Sanyo's "burn proof" recording, since licensed by other manufacturers? First "burn proof" does not mean protected against all failures (and certainly not protecting a blank from being burned - recorded - at all). It is a means of reducing the effect of buffer underrun. If the flow of data to a recorder is interrupted long enough to empty the buffer, the recorder reports the last full block written and the software holds off until the buffer reloads. Then it begins to write from the end of the last block to complete the track. The process works because manufacturing tolerances have been improved so much that the gap between the end of one block and the start of the next can be less than the space needed for the reader to recognize a break. There is still a gap, the laser does stop writing, but it is not recognizable because the specification allowed for an even longer one to be ignored.

Back when buffer underruns were common, "burn proof" would have been a boon. Today, an underrun is rare and indicates a serious problem in system configuration which should be addressed, not simply patched over. In a sense, it may be unwise to use the "burn proof" feature because it keeps you from knowing you have a problem to fix. Of course, when the feature saves you a disc, you will feel differently.

E-mail me at cd-r@mrichter.com

Return to Mike's home page

Readin', Writin', Speedin'

Before we get to matters of reading and writing speed, there's a bit of housecleaning to take care of.

In the old days - when a big hard drive offered 250 MB in a full-height 5.25" bay - there was a problem holding tolerances as the drive heated up with use. Those drives occasionally took time out for thermal recalibration. Drives which did contnuous recalibration were able to support continuous data streams, so were designated A/V (audio/visual). Modern drives have much tighter tolerances, so they all recalibrate continuously and don't stop working just for that. Unfortunately, the term 'A/V' has been retained to designate very high-speed drives. But what we care about is that you do **not** need an A/V drive for CD-R. However, remember that if you let your hard drive go to sleep, it will have to spin up again. Now, that's out of the way.

Writing

Okay, how fast should you write? For most users, it is not important. If you are in production, turning out copies of your own Greatest Hits album, then you care. But since modern mastering software uses only a small amount of a modern system's power and permits you to do almost anything while burning, it is no longer a big deal. If you are going to burn a half dozen discs a day or fewer, you will feel less pain waiting for a 4x burner than you will paying for a 16x. It's also a lot easier to find inexpensive media which write well at lower speeds.

Drives are getting faster all the time, so a 24x is better (quicker) than a 16x, right? Not exactly. Up to 16x, speeds do go up uniformly and as expected. But above that, the manufacturers start playing the same game they do with readers: 24x means a maximum speed of 24x, but the job won't be done in half the time a 12x takes. In fact, achieving the higher speeds requires that the laser stop and start to shift write speeds, so all such writers have buffer underrun protection. In itself, that's good, but it can also mean that you don't get the speed you expect. With a high-speed drive, your reader may not be able to keep up (see below) but you think you're being saved by underrun protection. Well, you're not likely to get a coaster, but that protection costs write speed and you may well find that your 24x writer is slower for some jobs than a 16x or even a 12x.

Reading

This primer is about CD-R. If you want a high-speed reader for some other reason, fine - but that is not of interest here. You want a CD-ROM reader to read discs for writing. In order to write "on the fly" - directly from reader to writer through mastering software - your source must supply a steady stream of data as fast as your writer will write them. The hardware provides some buffering of those data, but that is only enough to handle a short transient, typically 6-7 seconds. Modern mastering software provides additional buffering from spare system RAM. Still, eventually the buffer will be drained if the source dries up long enough.

To be sure that your source does not lag behind the demand, it must be able to supply data continuously at the needed speed. A rule of thumb is that the source should be twice as fast as the writer. In general, a CD-ROM drive rated at more than 8x does not deliver constant speed. The speed on the innermost part of the disc may be less than half that on the outermost. As a result, a drive rated at 24x may not be able to provide data fast enough to keep a high-speed writer happy. Complicating the matter further is the fact that audio extraction is often substantially slower than data transfer. Some drives do not do DAE at all; others may be rated 24x but deliver digital audio at 2x, 1x or slower.

Surprisingly, there can also be a problem with a reader which is **too fast.** A very fast reader may fill the buffer very quickly, then sit in idle while the buffer is draining. If that takes long enough, the reader may spin down. When the buffer needs to be refilled, the source starts to spin up again. If it takes too long to supply the data, the laser can be starved and the dreaded buffer underrun may appear. There are many ways to address that problem if it occurs for you: reduce the software buffer, slow the reader or increase its spindown delay. In fact, it may also be solved by slowing the write; that would give more time for the buffer to be reloaded. This problem will only occur when writing on the fly, it is detectable during test, and it can be monitored by watching buffer utilization if your software displays that information.

Emptying the cache

In a Windows system, writing to a drive is usually cached - buffered in RAM. That speeds up normal operations substantially and is necessary in practice if your system is to run well. However, it introduces a potential for disaster in some forms of CD recording. In particular, if you are writing to a packet disc, a lot of information is being held in RAM. Typically, that will include much of the information you want to store and (for fixed-length packets) the directory to all of the disc's contents. If you're in a hurry to get the disc out and eject it manually or shut down the computer without ejecting it at all, some or all of that will vanish. Needless to say, recovering what was in RAM instead of on the disc can be a problem.

When you shut down Windows, if you're smart you let the software establish equilibrium first. If you're writing packets, the same rule applies. Use the software to eject the disc; it will ensure that the pending information is written where it belongs. Save a few seconds by forcing the disc out of the drive and you may lose data, the directory or even the whole disc.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Recording DVD's

First things first: we're speaking of DVD Movie, aka: DVD-Video. DVD-ROM is handled the same way as any other data source. DVD-Audio will be a whole other set of issues when it settles down. So the question is, essentially, how does one record a DVD Movie onto some recordable, flat medium.

Part one of the answer is that you can capture the movie in analogue or by some digital mechanisms to your hard drive. Whether that capture is AVI, MPEG1 or MPEG2, it can be processed to a form suitable for CD recording: VCD, SVCD or what have you. The result will be at best a collection of CD-Rs playing chunks of the original, usually in poorer quality and in violation of copyright. For example, a 160-minute DVD Movie will convert to three VCD's at less than VHS quality and without the supplements, alternate angles, subtitles and other goodies of the original. If that's what you're after, go to it and keep the copyright cops away.

The following material was provided by a person knowledgable in the industry and represents a snapshot as of early 2000. The industry is changing steadily and we may not be able to keep this page up to date. Still, we'll try.

The thoughts of creating, storing and reproducing long video material with high quality on a single disc are very seductive, and some consumers may be tempted to invest in this technology in order to make their own DVD-Videos. Regrettably, the state of the technology for creating one's own DVD-Videos is nowhere near the ease of use and (relatively) low cost of creating CD-Audio and CD-ROM discs.

First, there's the matter of disc standards. At present, there are four different systems for recording directly onto a DVD-format disc. One, Pioneer's DVD-R, creates discs which can be played in many standalone DVD-Video players, but both the recorder and the blank discs are expensive and are designed for use by professional DVD authoring facilities for testing purposes. The other three systems are intended for consumer use. Only one of these systems is currently available: DVD-RAM, promoted by Matsushita. This system creates discs which can be re-recorded. Two types of DVD-RAM blanks are available: single-sided discs which hold 2.6 gigabytes, and double-sided ones which hold a total of 5.2 gigabytes (2.6 gigabytes per side). DVD-RAM discs must be in a special carrier in order to be used in a DVD-RAM drive. A single-sided DVD-RAM disc can be removed from the carrier to be played in a conventional DVD or DVD-ROM unit.

There are two other systems: DVD+RW, developed by Philips, Sony and Hewlett-Packard, and DVD-RW, developed by Pioneer. Only one such, the Pioneer DVD-RW DVR-1000, is shipping at least in Japan. DVD-RW is also a re-recordable format, but the discs hold more than the DVD-RAM discs and don't require a carrier. Further, DVD-RW is not backward compatible. DVD+RW is compatible, does not use a caddy and is supported by Philips, Sony, et al.

Even with current hardware, though, one still has to capture the video and audio material, digitize it (if it is not already digital), compress the video in MPEG-2 form, and confirm that the video and audio are accurately synchronized before writing a DVD-Video disc. It is possible to make a DVD-Video disc without menus, chapter entry points and the like, and the above process results in such a disc. However, we generally want to be able to access intermediate points in a long video without having to "fast forward" to them. To accomplish this, one must create "chapter stops" at the access points, and menus which allow one to skip directly to a key moment must be created. These are not trivial issues. I am also not discussing subtitling and other features which are usually taken for granted on commercial DVD-Video discs.

Until recently, the hardware and software for the above functions cost US$40,000-$50,000 for entry-level systems, and considerable skill is required to use them. Astarte, a German software company, developed considerably less expensive DVD-Video authoring software. This software and its developers were recently acquired by Apple Computer. Sonic Solutions, one of the primary developers of professional DVD authoring software, has announced lower cost systems, as well. Perhaps it will be possible in a few years for consumers to create their own DVD-Video discs as easily as they can now create CD-Audio and CD-ROM discs. However, for at least the near term, consumer-level recordable DVD systems are only suitable for computer data storage and retrieval.

E-mail me at cd-r@mrichter.com

Return to Mike's home page

AutoEverything

One of the most commonly asked questions about CD-R is: How do I start my CD-R automatically. The answer is simple for some cases, but can get tangled - usually in the case you want.

Auto Insert Notification (AIN)
AIN indicates to the operating system that a disc is present in a drive. If AIN is on, each drive is checked for its contents. That check is important in UDF for the system to recognize how to take control; AIN must be on for UDF software to operate reliably and stably. However, reading your drives when you are burning can be disastrous; most modern mastering software will control AIN so it may be left on safely when running those programs. AIN is necessary for Autoplay and Autorun to work; the OS cannot know to start the disc unless it knows that a disc has been inserted for it to start.

TweakUI
Microsoft offers a program called TweakUI which provides several nice features which probably should have been built into the Control Panel. One of the nicest panels in the program is Paranoia, which gives you an easy way to turn the autostart functions on and off. Since a registry hack would be needed to do the job otherwise, TweakUI is the only mechanism I recommend for controlling Autorun and Autoplay.

Autoplay
If you turn Autoplay on and insert an audio disc, your default CD player will be activated and begin to play the disc with the first track.

Autorun
This can be complicated and it's worth checking Microsoft's site if you're getting fancy or want an icon. For the simple case where you want to execute a file RUNME.EXE, you simply use a text editor such as Notepad or Wordpad to create a file, AUTORUN.INF which you place in the root of your disc. The text need only be:

```
[Autorun]
open=runme.exe
```

That's fine if the RUNME.EXE is in the root, but it also works if the program is in a folder as long as you use the fully-qualified relative pathname for the file. OUCH! No, that's not complicated, but the words are. What it means is that instead of "open=runme.exe", if runme.exe is in the folder direct\direct1 you write "open=direct\direct1\runme.exe" - whew! Similarly, if you want RUNME.EXE to start with a specific file, say START.FIL, then you write "open=runme.exe start.fil" - again using fully qualified relative pathnames as appropriate. ("Relative" means without specifying the drive letter. That's important since you want the disc to autorun in whatever drive is being used and you cannot be sure what drive letter it will have in all cases.)

Well, what do you do if you want to run whatever program may happen to be associated with a file, say FILE.FIL on the user's computer - but don't know what program that is or where it's located? Microsoft has a partial solution to that in Windows 9x:

```
[Autorun]
open=start file.fil
```

Note that you do not control the program that runs because you don't know what (if anything) the user has associated with the FIL extension. She may use MediaPlayer for RealMedia files or she may have some version of RealPlayer installed; she may use Netscape's or Microsoft's browser for .HTM.

Another problem is that your user may be running Windows NT, which does not recognize "open=start". There are several programs for doing that, including SHELLOUT.EXE at this CD-R site. There are also menu programs which give the user options on starting up once one is autoplayed from the disc. Try searching for freeware or shareware choices for "setup" or "start menu" or similar terms at your favorite source, such as www.download.com.

Where to go

For the real skinny on Autorun, try
http://msdn.microsoft.com/library/psdk/shellcc/Shell/Shell_basics/Autoplay_intro.htm
If you cannot reach AIN simply by right-clicking the drive letter and looking at its properties, see your OS tips here in the primer; a link for TweakUI is there as well, though as operating systems evolve, you would do well to check your OS installation disc first. Above all, feel free to try this stuff; it is simple and demands no fancy software until you start looking at tiered menus with background sound ...

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Tips for Win 9x Users

Jeff Arnold offers:
The following tip can greatly improve CD-R performance under Windows 95. This could prevent buffer underruns and other common problems from occuring. By default, Windows 95 does excessive file caching. This is a waste of CPU time, memory, and possibly disk space, since you will never go back and use anything in the cache. To fix this problem, do the following...
1) Open the file SYSTEM.INI with a text editor (this file will usually be found in C:\WINDOWS or C:\WIN95).
2) Find the section of the file called "[vcache]".
3) Add the following lines **after** the "[vcache]" line.
minfilecache=512
maxfilecache=4096
4) Save the file changes.
5) Reboot your PC.

NOTE: If you have any problems with this setup, you can just remove the lines and reboot your system again.

NOTE: If you have changed the cache and are having problems with DCD recognizing a formatted disc, try backing out this change.

IMPORTANT: Adaptec provides valuable information consistent with Jeff's - and freeware to make things work. For example, you can and should pick up ASPICHK from them. It determines the modules in your ASPI layer and reports whether or not they need to be updated. If you need to update, you can do it with freeware from the same site.

Other thoughts to consider - and not **just** for Win95:

Where possible, put your pagiefile (swap file) on a different physical drive from the one that you are using to store your data.

Defragment your HD before you read data or build an image from a CD.

Defrag the HD from which you will be writing before starting the burn.

A friend reports:
It appears that Win95 is not shy about putting duplicate drivers here and there. You may only see this in "safe" mode, so bring Win95 up in Safe Mode, go to System Properties, and look at Device Manager. If redundant devices appear that weren't visible in regular mode, deleted *all* the hard drive devices, floppy drive devices, and redundant devices ports, etc. Then reboot several times until Win95 finds all the hardware again and installs only *one* copy of each.

Autorun and Autoplay

While packet writing needs AIN, this in itself does not enable Autorun (or Autoplay). For that purpose, I suggest using TweakUI from Microsoft's Power Toys. At this writing, it can be found at http://www.microsoft.com/NTWorkstation/downloads/PowerToys/Networking/NTTweakUI.asp

---

E-mail me at cd-r@mrichter.com
Return to Mike's home page

NT Notes

Windows NT 4 provides a fine environment for CD-R. Operation is quite similar to that in Win95, but a few comments are worth noting.

NT 4 Server

To use NT Server for burning CDR's successfully you need to use regedit and change the following registry setting:

> HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control \Session Manager\Memory Management

Find a DWORD entry "LargeSystemCache=0". If the value is set to "1", file caching has higher priority than the application code in the fight for real memory. This has the nasty effect that when you handle large files, like a cd image, or a stream of files to be burnt, NT will page out the cdr-software executable (and everything else, except the kitchen sink) to the hard disk. Problem is that in the next instant NT finds that it needs to page the executable back in as it is actually running and handling these large files NT made room for... So the system will keep bouncing the program code in and out of memory, competely annihilating the system throughput, until you end up with a coaster. If I remember right, is this value set to 0 (as desired) as default on NTWS, while it is set to 1 (coaster city) as default on NTS. It is quite possible that the install (or use) of the Server resource kit caused this change to the "standard server value". This tip might be useful for anyone wanting to use NT Server when making cdrs. I run Server myself, and have no problems burning provided I do this single change to the registry.
Provided by Glenn, glennb@algonet.se

Iain Jenkins, ijenkins@btinternet.com offers the following variation on the above and additional data:

The change can be implemented using control panel as follows:
Go to control panel - network - services - server click properties
optimization change from the default "optimize for file sharing" to "maximize throughput for network applications" dismiss boxes
job done on reboot!

Another tip for setting the DMA on/off on NT for EIDE devices (it doesn't appear on the drive settings as in W9X):
Extract program DMACHECK.EXE from your service pack (3up) disk, and run it - it will give you the chance to control the DMA setting. If it doesn't give you the choice, you can still fall back on regedit to give you the desired results (it doesn't always allow DMA to be set due to incorrect device detection)
Go to: HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Atapi\Parameters\Device X (where X is your hard disk/CD no.) look for a key DmaDetectionLevel (It may need to be created) DMA set as required:
0x0=off
0x1=on
0x2=forced

Also, check the file ATAPI.SYS for at least version dated 1/98.

Auto Insert Notification

Again, this takes editing the registry.

HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\Cdrom

Enable AIN by setting the value of Autorun to 0x00000001 (1)

NOTE: This in itself does not enable Autorun (or Autoplay). For that purpose, I suggest using TweakUI from Microsoft's Power Toys. At this writing, it can be found at http://www.microsoft.com/NTWorkstation/downloads/PowerToys/Networking/NTTweakUI.asp

ARTICLE-ID: Q168113 TITLE : Using Windows 95 PowerToys with Windows NT 4.0 Additional Information. There are two other keys that can affect this functionality:
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer
NoDriveTypeAutoRun=0x00000095
HKEY_USERS\.DEFAULT\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer
NoDriveTypeAutoRun=0x00000095
The correct value for each is 0x00000095.

Making a Boot Floppy (NT4 or 3.51 workstation)

There are times when Windows NT won't start, and you know why. If this occurs, you can go through the usual procedures to restore the system, but since you know what's wrong, you can get running again more quickly if you have a boot floppy. To create one, insert a floppy disk into Drive A. Right-click on the floppy disk icon and choose Format. Format the disk using Full Format (just to be safe). Now open Windows NT Explorer and click on the root folder (usually C:\). Copy the following files to the floppy disk:
Boot.ini
Ntdetect.com
Bootsect.dos (for dual start-up installations)
NTLDR
Ntbootdd.sys (if it's in the root folder, copy it)

If you don't see these files in your root folder, choose View, Options in Windows NT Explorer. Select the radio button labeled Show All Files. Click Apply and then OK. Now you should see the files. If you don't, press F5 and look again. Using the boot floppy, you can boot into your damaged system and make the appropriate repairs.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Standalone Recorders

This page was provided by a correspondent. To the extent that I have information on standalone recorders, the information is correct. I also take full responsibility for the footnote. For the rest, I am only the relay for the following.

> My experience has been with three models: Philips CDR-765, Tascam CD-RW 500 and HHb 800. The basic principle is that consumer models such as the Philips will only work with Digital Audio or "audio" discs. An audio disc is basically the same as a data disc except that it has a flag in the ID info track of the ATIP which identifies it as an "audio" CD. It carries a higher price tag because of the royalty added to consumer audio media like cassettes, VHS tapes etc. to reimburse the "industry" for copyright infringement. Data CDs are cheaper because they are not considered media for consumer use where copyright infringement is going to be a primary concern. Professional level CD-Rs can use data discs because of the anomalous law that says that assumes only consumer equipment will be used for infringement. The Philips could be tricked by swapping a conventional data disc for a Digital Audio disc at the right point in the cycle; these units are no longer on the market.
>
> Pro and consumer recorders also differ in the handling of SCMS - the "copy bit" with values of 00 for unlimited copies, 01 for one more digital copy, 11 for no more digital copies allowed. Consumer audio hardware will write 01 on the first burn, and then on a copy of anything sourced as 01 will write 11 so no more copies can be made. Pro audio gear will almost always write 00 and ignore 01 and 11 so you can make copies through optical or coax digital connections; AES/EBU digital connections don't pass SCSM info so are always ready to copy.
>
> There's also a new level of standalones, the so called "prosumer" - hybrids of consumer and pro hardware. Priced in the $500 - $700 range, you get lower cost of the initial unit, ability to use data discs, and in some cases no SCMS. The Tascam 500 is a studio unit (about $750 new) which I hooked up with a coax digital connection to my CD player for copying. Since it records in real time, a 74-minute CD takes 74 minutes to create. It syncs track IDs so the CD will be ID'd exactly like the original - very handy for letting it record and coming back an hour later to switch discs.

Footnote: There have been reports that Digital Audio media do not record correctly in some computer CD-Rs. While their higher price without payoff makes them unlikely choices when not needed, you should be aware that they may not work in your computer-based recorder.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

Configuring Storage

This page is about hard drives and partitions. If you already know all there is to know on the subject or if my recommendations - which are only slightly related to CD recording - are of no interest, please move on.

The current fashion in computer systems is to deliver them with a single, large drive formatted in a single partition. Simply put, that's a recipe for inefficiency and an invitation to poor practice. The first time you try to back up 40 GB or to search 20,000 files, you'll appreciate both factors - and if you decide to defrag that partition after some weeks of fragmentation, you'll be lucky to finish the job overnight.

Run two physical drives

Writing from one defragged drive to another is quick. The heads progress on each as quickly as they can and information is transferred at nearly the sustained rate of which the hardware is capable. Writing within one drive, even between two partitions, requires that the heads seek from each read sequence to the corresponding write. Seek time is **very** long compared with read and write times. For example, backing one partition up to another on a different drive is at least five times faster for me than backing it up to the same drive. Every time you move a large file - for example, to write the Undo for a WAV file you're editing - you will see the difference.

Make partitions for use

There are three sorts of information stored on most computers:

- Operating system and essential utilities
- Applications programs
- Large-scale working data

Operating systems are getting close to a Gigabyte. Add in the utilities to support your hardware, the pagefile/swapfile (on the boot drive to handle error logs) and the like and you need a 2-GB C: partition. Some people want Gigabytes of application code, but for the most part they are simply holding applications they think they may want to consider starting to learn some day (when they'll be outdated, but who's counting?). Another way to bloat the applications partition is to throw in all the sample files and tutorials the publisher provides; just because the file is on the installation disc does not mean that it needs to be on your hard drive. If you look at the programs you use, you're likely to find that they, too, will fit into 2 GB with room to spare. (Of course, many applications use data only sparingly. Your accounting data, for example, logically stay with the programs that write them.)

Now for all those MP3s, WAVs, TIFs, JPGs, MOVs, AVIs, ... you absolutely have to have. That's cool and there's no reason not to have them accessible on your hard drive when storage is so inexpensive. The fact is that those files are not used the same way as your OS and apps. They are not searched for the same reasons and don't require the same sort of backup. Good practice says that when you've written some hundreds of MB since your last backup of those files, you write a CD-R with the new ones and mark the disc so you can find the files again. Logically, you make big - okay, **BIG** partitions for them. You don't back up those partitions, you back up their files. You don't search thousands of DLLs to find the MOV you want; you search the partition with a few dozen MOVs, AVIs and so on.

Since you're reading this primer, you are presumably also writing CDs. If your writing from a hard drive, you known that that partition should be defragged. In fact, the easy way to do it is to make one or two partitions of 1-2 GB each and to leave them empty as a rule. When it's time to write that disc, just put the source files into the empty partition - no defragging needed because the files don't fragment when you write them that way. Even if you have managed to fragment that storage, defragging such a drive is quick and sure.

My primary system

At this writing, I have two, 20-GB EIDE drives on my system. Each has three partitions: two are FAT16 (2 GB) and one is NTFS - it could be FAT32 just as well. Drive 0 has C:, E: and G:. Drive 1 has D:, F: and H:. G and H hold current backups of D: and C: respectively and can be reached from a DOS boot floppy. C: is the boot partition with OS and such; D: is the application partition. G and H have plenty of space left to hold the files to write a CD. E: and F: are the work spaces. My downloads go into a dedicated folder there and are checked for viruses regularly. Audio and video files are held there while I'm working with them and while waiting to be backed up in logical chunks to CD-R. I put my TEMP folder there as well both to speed up file transfers and to be sure I won't run out of space. And by having two such partitions on different physical drives, most of my work with large files runs at high speed.

One footnote is worth adding, I hope. The documentation will assure you that storage efficiency goes up with FAT32 and NTFS. That is certainly true - but it doesn't matter. Today, you buy Megabytes of storage for a penny; how much is it worth to save a few tens of Kilobytes per application? Like backup software, partitioning for backup is very cheap insurance. I back up C: and D: at least once a week; it takes about ten minutes total and I can burn the CD-R copy when and if I wish. I don't back up the other partitions, but I do keep CD-R copies of the important files they contain and can rebuild either easily if needed. If I have a head crash on either drive, I can be back on line in half an hour with the full system by restoring what's needed from the other drive. (Admittedly, there can be a problem with installs since the last backup, but I can reinstall anything I put in over the past few days.)

Finally (at last!), if you buy the thoughts above and want to modify your configuration along similar lines, you will find PowerQuest's Partition Magic an easy way to do the job.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

A 650-MB WWW Site

Let me begin by saying that my primary use for CD-R is not that of most others. Before I bought my first writer, I defined what I wanted to do with it and how. The result has been (through September 2000) twenty-one CD-ROMs pressed for me and distributed through the Internet to a small but appreciative 'market'. This note traces the logic leading to the selected format(s) and suggests tools for following that lead.

Format: Why and How HTML

My objective has been to make files in various formats easily accessible to users with limited computer skills. I quickly reduced the options to two: Adobe Acrobat's PDF and HyperText Markup Language - HTML, the language of the World Wide Web. Acrobat offers several distinct advantages: well-defined formatting with esthetic values such as multicolumn text, explicit placement of graphics and full-text search. It shares with HTML the ability to link elements and to operate freely across platforms. HTML won out for me for several reasons, notably the ability of the user to run a browser she already knew and the flexibility to adapt that browser to her particular needs. If she is running a monochrome display, monochrome options are at her fingertips; if she needs larger type, she may simply set her defaults. In addition, creating and editing HTML is easy with tools of moderate cost and ready availability and, perhaps most telling of all, one can easily embed a PDF into an HTML file, but not HTML into PDF. Finally, PDF would have required the user to acquire a reader and to install it onto the computer; I wanted a self-contained, machine-independent configuration.

Because I want my discs to read on as wide a range of platforms as possible and with the least limitation on personal configuration, I use only the most primitive forms of HTML. Fonts are set with H1-H6 rather than by specifying them. There are no frames and very few tables; in fact, inline and background graphics are used to supplement text instead of to replace it. As a result, a user can browse any site or CD-ROM I create with virtually any browser and retrieve from it whatever her tools support. Another plus for HTML is that if a format begins to be supported, I can add it to my 'site' and the user can get it on her own for her platform. In a word, I can deliver the product we both want without requiring the user to adapt to my means of delivery.

To create the HTML, I use HoTMetaL Pro, a powerful language processor which implements far more than I need. One of its virtues is controllable enforcement of rulesets. I can pick the flavor of HTML I want to use and have the program enforce it as I write or after the fact. You will not be surprised to find that I use the most limited ruleset available and keep it on at all times. If you prefer a less costly approach, such as the shareware HotDog, that's fine, too. However, I do **not** recommend that you use HTML export or authoring from programs not dedicated to the purpose. They tend to produce code which looks just right to you - and fails miserably when seen on another system. In fact, they throw away many of the advantages which attracted me to HTML in the first place. They often implement the specific font choices you made when you wrote instead of leaving options open for the user. I want to create an open shop with its wares laid out for all visitors to see and to use; those programs tend to organize the shop into the specific pattern you see on your screen and to force visitors to see it your way.

My Recipe - Flavor as you wish

Unlike a site on the WWW where a few megabytes are stored, the CD-ROM offers 650 MB and needs careful organization. My discs are organized with a home page that provides direct access at the top to the major components, technical and legal information as required, and advice at the bottom on special needs for configuration and helpers. Because I work on a PC and most of my customers are on PC's, I develop first for that machine and then through beta test verify operation on other hosts (both Mac and Unix). However, I do nothing that I can avoid which would keep a user out. On my latest discs (read about them here), I do require MP3, which is not built into all browsers.

I include on each disc everything needed to get up and running. For a novice on a PC (running Win 95 with Autorun ON), the disc is inserted into the reader and autoruns. I bought a royalty-free license for I-View, a compact and flexible browser designed for use off-line. I obtained permission from the authors to include freeware not only for WinAmp but also for a suitable Mac browser and helper. A Mac user can run an existing browser (or may look at a MACREAD.TXT file in the root) to install the included browser and helper on his own machine. Of course, if the user prefers another browser and is willing to add the MP3 or other helpers required, that option is also open. Again, everything is kept as simple as possible; elementary menus are used instead of frames or tables; the user is given the greatest possible flexibility.

With HTML, cross-referencing is my job before I create the disc, but that access is easily supported. Suppose I want a record of my family's history and have video clips, still photos and audio recordings prepared for inclusion. I can supply a page of Uncle John with a portrait and text and with links to sound and video clips and to scenes in which he figures prominently. I can link to that page from a page with the family tree or from each page of narrative text in which his name is mentioned. I can even take a family picture and provide a link from his area of the photo to 'his' page. Note that the same page or the same file may be linked from each point where it seems fitting to me.

Special cases and methodology

On three of my CD-ROMs, I incorporate the history of my primary WWW site. That is easily done by taking the individual pages that have been posted and linking them either chronologically or by subject. The result is a large, instantly accessible site on the CD-ROM. Note that I have to be careful because of the way most browsers operate. For example, I use playlists of MP3 files, but neither MSIE nor NN will play them from the disc. I-View has no trouble doing so because it is designed for the purpose; to make a conventional browser work, it would have to know where to find the CD-ROM. The usual browsers load the playlist into the TEMP folder, then look for the referenced files there when they are acutally on the CD-ROM. On a PC, they want the letter of the drive (which, of course, I do not know on the user's machine); on a Mac, they could use the volume label, but for some reason they do not. Only by testing can those cases be found and even there you need to be sure to use a variety of hosts and of configurations to be confident. (I goofed on one disc and some files are only playable on machines with more than 32 MB of RAM; I made a change which imposed that requirement after testing was complete and have been embarassed by the awkward workaround for the finished discs.)

My way of building the disc relies on my own system hardware and preferences; yours may vary. I start with an outline of what I want to do on that disc and an overall map of its organization. I break that down into pages and groups of pages to be written. As I prepare material for each page, I write the supporting HTML and check it on that level. I store the pages on a Jaz and periodically make a CD-R copy for backup. (Typically, I have several projects in development and dedicate a Jaz cartridge to each. Then I can work on that project simply by selecting the appropriate cartridge.) My home page for the disc is built in outline form; fleshing it out is the last step and exploits what I have from previous efforts.

Periodically, I run HTML Power Tools to verify my linkages. I skipped that step on my prototype; as a result, it has both duplicate files and broken links. When I have a viable version of a new project, I send it out to testers who are knowledgable in the computers they use and the subject matter. While they evaluate the work, I write the home page and work on other projects. As the testers feed back, I return to the job with a fresh eye and make changes as appropriate. That is also a time when I may add another cross index or additional text, but not additional files unless I'm prepared to ask for retest. I also contact the person who will press the disc and prepare the graphics and other materials needed for production. When the process is complete, I again run HTML Power Analyzer to verify the layout, then burn a couple of CD-Rs to check yet again on my PCs. Finally, I drop the discs into the mail to the pressing plant and erase the Jaz. I interact with my broker, Digital Bim, by e-mail and by telephone. About a month after mailing the masters, the finished product is delivered and ready for distribution. For your information, the cost of having discs pressed is roughly $600 for 500 copies, $850 for 1000 copies.

E-mail me at cd-r@mrichter.com

Return to Mike's home page

Interoperability

"Interoperability" means a design supporting operation on multiple platforms or in multiple environments. In general, it means giving up features restricted to specific computers, operating systems or applications in favor of finding more users. Interoperability is seldom universal and is generally expensive: you have to give up a lot to increase your market.

If your CD recording is for your own use, interoperability is worth very little to you - and you are invited to skip the rest of this page. Or not. You know what computers and other devices you intend to have run your discs, so where's the issue? Simply put: do you know how you'll want to use them next week? Do you know you won't be installing a new OS, buying a new player or otherwise changing your base of operations?

The most nearly universal format for CD recording is Compact Disc - Digital Audio, CD-DA, redbook music. The standard is rigid and universal; there are extensions such as CD-Text, but adding them does not cost operation on equipment which does not support them. The problem with CD-DA is a consequence of its interoperability - inflexibility. If you want to play a CD of 1925 monaural 78s, you have to use the same audio parameters required to get year 2000 stereo. There's room on that disc for at least four times as much music without noticeable loss of quality (22.05 Ksps, monaural), but the standard will not permit it. If you write MP3s instead of CD-DA, you can put about 45 hours at 32 Kbps onto a CD-R which would hold only 74 minutes of redbook audio - but then you lose interoperability with all those audio CD players. That may be quite acceptable to you - you have an audio player which handles low-rate files without complaint. However, that won't help your neighbor whose MP3 portable demands 44.1 Ksps; or the guy down the block whose CD player only plays CDs - CD-DA recordings.

The problem shows up in many areas other than audio format. One is file naming. Your MP3 player, like your computer, displays information with which you can select what you want to do. Different players have different rules for that display, for the handling of folders and for tolerating multisession recording. Different operating systems on computers have varying rules for filenames as well. Fortunately, all common systems will handle strict ISO 9660 naming: 8.3 with a restricted character set (upper-case letters, numerals and the underscore). All MP3 players so far will handle a single-session disc without folders and eight-character names having an extension of MP3. As you stray from those rules - longer names, folders, additional characters, multisession - you lose platforms to play your discs.

If you want to include text and graphics on a disc, you have many ways to format it. Of course, TXT is the most readable. As with ISO 9660 filesystem, TXT is nearly universal, though very limiting. Other common formats are DOC (Microsoft Word) and PDF (Adobe Acrobat). If a sufficiently restricted set of capabilities for either one is used, free downloadable programs make those readable on most platforms. Another choice (my own preference in most cases) is HTML. But with any of those, you can write an elaborate version which is readable only with some versions of Word, with an advanced Acrobat reader or with a specific browser. It is your option whether to use features which enhance your product and restrict its use by others. For example, different browsers implement various features of HTML and supersets of HTML which others either treat differently or regard as errors. Most will handle mouseover (where the display changes when you pass your mouse over a target area), but some will not so you will lose the vision-impaired and others with graphics turned off or unavailable. On a simpler level, there is a small set of diacritical marks - accents and the like - which are handled easily in standard HTML, but go beyond them (to true single- and double-quotes, for example) and some browsers will give unintended results.

It is your choice when you design your project to build in interoperability. The point of this note is to induce you to think about the question at the beginning. Once the disc is designed, it is difficult or even impossible to extend its usability.

E-mail me at cd-r@mrichter.com
Return to Mike's home page

A New Medium - New Ideas

CD-R technology can allow us to realize familiar things in familiar ways. We can back up CDs and CD-ROMs, we can save archives to a nearly permanent medium easily and economically. We can transcribe tapes and LPs to CD's. We can even back up our computer systems. Yet there are potentials here which are not yet being exploited - at least, not extensively - that offer new opportunities.

Browse your disc

One architecture for a CD-ROM is that of a WWW site. If that format is used, one simply browses it with one's choice of browser on any appropriate platform. The directory is not printed on the traycard - it is in the HTML with which one accesses the files. No looking up the track number or filename; just click on the link. I have been using that architecture for several published CD-ROMs, with excellent results. I am using a variety of formats - HTML, GIF, JPEG, WAV, MP3, MPG - to get a rich blend of content on a single disc.

Disc Jockey Time

We all know the virtues and the limitations of the CD: great sound limited to less than 80 minutes; continuous or selective play with the sequence predetermined on the disc or requiring programming into the player. But there is a viable alternative allowing more music to be stored on a disc and accessed in whatever order is appropriate for the job.

Suppose that the tracks are recorded on a CD-ROM, not in CD-DA format, but in WAV or MP3. Then playlists can be constructed which preselect track sequences for specific purposes: an hour of slow dancing to end an evening, twenty minutes of increasingly upbeat tracks leading to the climax of the evening, three hours of Glenn Miller for the 'mature' audience, and so on. If special needs arise, one list can be modified in real time or a new one can be constructed while another is playing. A single CD-ROM can then hold ten hours or so of music to fill a full gig.

Playback? Instead of carrying a stack of CD's and CD-R's to the party and hoping that the host's CD machine is good enough, carry one disc and a laptop you know. Connect its line out to the local sound system and you have full control right from your computer. With a program such as WinAmp from http://www.nullsoft.com/ you can call up a graphic equalizer to tone things down or to tune the sound to the local system. WinAmp and other programs for WAV and MPn also support simple scripting and both sequential and quasi-random playback.

Footnote: Two at a time!

So, does it make sense to put both of those ideas together? You bet it does! When I'm working on a disc, I write the HTML and the files - usually, MP3 and JPEG - at about the same time. I use only relative addressing and provide a folder for each logical collection with all required files. Then I can check things out with browsers on a folder level. When I'm ready, I add a home page linking to the HTML in the folders, add I-View, WinAmp and whatever else is appropriate, run some checks with analysis tools and play with it a bit - and then burn away.

E-mail me at cd-r@mrichter.com
Return to Mike's home page