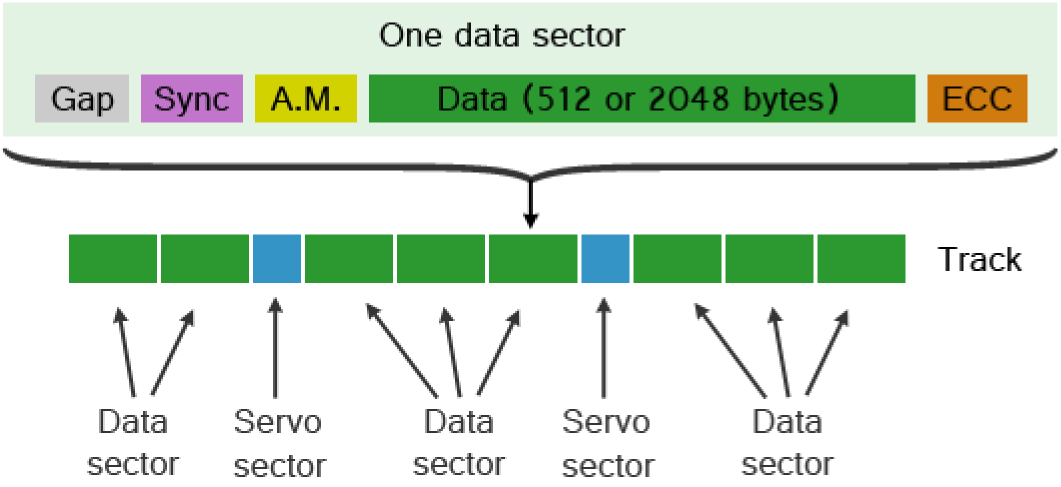
Bad Sector Recovery

January 2nd, 2013 atola insight, Dmitry Postrigan

Hard drives are built in a way so that they never return *unreliable data*. This means that if the hard drive cannot guarantee 100% accuracy of the data requested, it will simply return an error and will never give away any data at all.

Understanding Bad Sectors

General causes for bad sector formation are physical or magnetic corruption. Physical corruption is easy to understand – it occurs when there is physical damage done to the media surface. Magnetic corruption occurs when hard drive miswrites data to a wrong location. While the latter may seem to be less damaging, it is actually as dangerous as physical damage, since miswritten data may not only damage adjacent sectors, but also servo sectors.



Regardless of the cause of damage, there are several possible outcomes:

- Address Mark field corruption
- Data corruption
- ECC field corruption
- Servo sector corruption
- Or any combination of these

What is common with all these types of corruption is that your Operating System or normal data recovery tools cannot read the data from those sectors anymore.

Let’s find out exactly what happens when a tool tries to read a sector that has one of the above mentioned problems.

Address Mark corruption

When Address Mark is corrupted, the hard drive simply cannot find the requested sector. The data might still be intact, but there is no way for the hard drive to locate it without the proper ID. Some modern hard drives do not actually use sector ID or Address Mark in the sector itself; instead, this information is encoded in the preceding servo sector.

Data corruption

To verify data integrity, a hard drive will always validate it with the Error Checking and Correction algorithm using the ECC code written after the data field (see above diagram). When data is corrupted, the hard drive will try to recover it with the same ECC algorithm. If correction succeeds, the drive will return the sector data and will not report any error. However, if correction fails, the drive will only return an error and no data, even if the data is partially intact.

ECC field corruption

Although this is rare, the ECC code can also get corrupted. In this case the drive reads perfectly good data from the sector, and checks its integrity against the ECC code. The check fails due to the bad ECC code, and the drive returns an error and no data at all, because there is no way to verify data integrity.

Servo sector corruption

There are up to a few hundred servo sectors on a single track. Servo sectors contain positioning information that allows the hard drive to fine tune the exact position of the head so that it stays precisely on track. They also contain the ID of the track itself.

Servo sectors are used for head positioning in the same way a GPS receiver uses satellites – to exactly determine the current location. When a servo sector is damaged, the hard drive can no longer ensure that the data sectors following the servo sector are the ones it is looking for, and will abort any read attempt of the corresponding sectors.

How Bad Sector Recovery Works

Once again, hard drives are built to never return data that did not pass integrity checks.

However, it is possible to send a special command to the hard drive that specifically instructs it to disable error checking and correction algorithms while reading data. The command is called Read Long and was introduced into ATA/ATAPI standard since its first release back in 1994. It allowed reading the raw data + ECC field from a sector and returning it to the host PC as is, without any error checking or correction attempt. The command was dropped from the ATA/ATAPI-4 standard in 1998; however, most hard drive manufacturers kept supporting it.

Later on, when hard drives became larger in capacity and LBA48 was introduced to accommodate drives larger than 128 GiB, the command was officially revived in a SMART extension called SMART Command Transport or SCT.

Obviously, since the drive does not have to verify the integrity of data when the data is requested via the Read Long command, it would return the data even if it is inconsistent (or, in other words, the sector is “Bad”). Hence, this command quickly became standard in bad sector recovery.

There is also another approach which is based on the fact that some hard drives leave some data in the buffer when a bad sector is encountered. However, our tests have shown that chances of getting any valid data this way are exactly zero.

Debunking Bad Sector Recovery

So, to “recover” data from a bad sector, one would simply need to issue the Read Long command instead of “normal” Read Sectors command. That is really it! It is so simple any software developer who is familiar with hard drives can do it. And, sure enough, more and more data recovery tools now come with a Bad Sector Recovery option. In fact, it has come to the point when if a tool does not have a bad sector recovery feature, it automatically falls into a second-grade category.

Error checking and correction algorithms were implemented for a reason, which is *data integrity*. When hard drive reads a sector with the Read Long command, it disables these algorithms and hence there is no way to prove that you get valid data. Instead, you get *something*, which may or may not *resemble* your customer’s data.

Tests in our lab had shown that in reality, by using this approach you will get much more random bytes than anything else. Yes, there are cases when this approach allows recovering original data from a sector, but these cases are extremely rare in real data recovery scenarios, and even then, only a part of the recovered sector will contain valid data.

Even when we got some data off the damaged sector, what exactly should we do with its other (garbled) part? And how exactly do we tell which part of the sector has real data in it and which is just random bytes? Nobody is going to manually go through all sectors in a HEX editor and judge which bit is valid and what is not. Even if someone did, there is no way to guarantee that what they see is valid data.

And this is where the real problem starts.

Dangers of Read Long approach

Imagine a forensic investigator recovering data off a suspect’s drive while the drive has some bad sectors on it. To get more data off the drive, the investigator enabled Bad Sector Recovery option in his data acquisition tool. In the end, his tool happily reported that all sectors were successfully copied, so he began extracting data from the obtained copy.

When looking for clues, he found a file that had social security numbers in it. He then used these numbers in one way or another for his investigation.

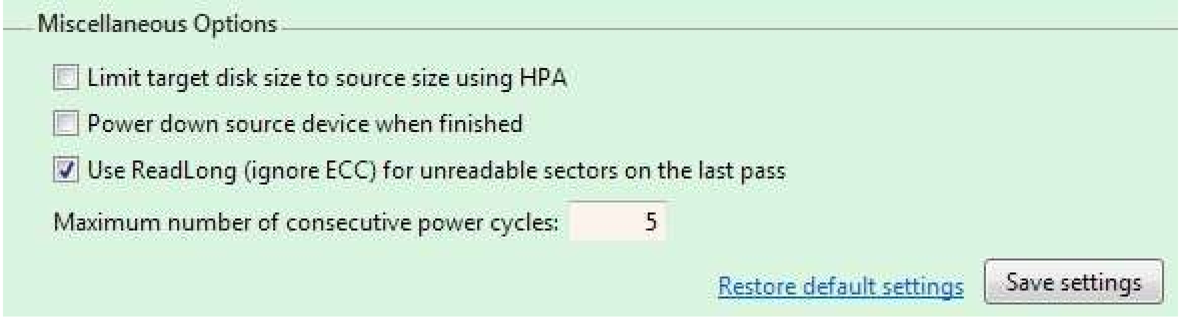
What he did not know is that one of the sectors that contained these numbers was recovered via the Read Long command, and some bits were flipped (which is very common for this approach). So, instead of 777-677-766 he got 776-676-677, causing him and other people a whole lot of unnecessary trouble. Another example: recovering a damaged file system, even slightly altered data in an MFT record can mislead the file recovery algorithm and in the end do much more harm than if there was no data copied at all in that sector.

Once again, error checking and correction algorithm is in place for a great reason. There is absolutely no magic in bad sector recovery; it is impossible to recover something that just isn’t there.

There are tools that claim better bad sector recovery because they utilize a statistical approach, an algorithm where the tool reads the bad sector a number of times and then reconstructs the “original” sector by locating the bits that occur most often in the sector. While these tools claim this approach *could* improve the outcome, there is no evidence to back up the validity of such claims. Furthermore, re-reading the same spot many times while the hard drive is failing is a good way to cause permanent damage to the media or heads.

So what about Atola Insight?

Like all high end data recovery tools, Atola Insight supports bad sector recovery via the Read Long approach.



Miscellaneous Options

□ Limit target disk size to source size using HPA

□ Power down source device when finished

☑ Use ReadLong (ignore ECC) for unreadable sectors on the last pass

Maximum number of consecutive power cycles: 5

Atola Insight 3.0 which is going to be released in January 2013 has even more profound functionality. Again, we are one important step ahead of competition: the locations of recovered sectors are automatically stored in the case management database. After imaging is complete, Atola Insight 3.0 automatically marks all files that contain sectors recovered with the Read Long command.

This way the operator has the ability to disregard such “unreliable” files and manually verify file integrity if it is an important one.

Once again, if you are after valid data, avoid using any bad sector recovery algorithms. These algorithms will never offer data integrity no matter how complex their implementation is. If you absolutely must recover data from bad sectors, make sure you use a tool that properly accounts for these recovered sectors.

We always advise our customers to avoid using a bad sector recovery option until absolutely required. In Atola Insight you can always create an image without bad sector recovery first, try recovering files, and, if unsatisfactory, go back to Imaging and improve the image by enabling new options, including a bad sector recovery, and running the imager only on bad sectors.

When it comes to bad sector recovery, make sure your data recovery tool offers this level of flexibility.