Futuremark Corporation

May 23rd, 2003

To the Media, Futuremark's customers and business partners:

Audit Report: Alleged NVIDIA Driver Cheating on 3DMark03

After the launch of 3DMark03 Build 320 Futuremark has received reports from the members of its BETA Program concerning certain anomalies with 3DMark03 and Nvidia drivers. ExtremeTech (www.extremetech.com) has published an article[1] on suspecting NVIDIA drivers to improperly boost scores on Futuremark's 3DMark®03. Some of these anomalies have also been reported by Beyond3D[2]. Alarmed by all these reports Futuremark has conducted a thorough internal audit regarding this matter and has verified that certain NVIDIA drivers indeed seem to have detection mechanisms, which are triggered by components of the 3DMark03 program. We have identified eight such mechanisms.

In our testing, all identified detection mechanisms stopped working when we altered the benchmark code just trivially and without changing any of the actual benchmark workload. With this altered benchmark, **NVIDIA's certain products had a performance drop of as much as 24.1%** while competition's products performance drop stayed within the margin of error of 3%. To our knowledge, all drivers with these detection mechanisms were published only after the launch of 3DMark03. According to industry's terminology, this type of driver design is defined as 'driver cheats'.

We are publishing this document to report to our customers in detail about our findings. Main reason behind publishing this document is to answer the criticism presented against synthetic benchmarks and their reliability when testing hardware performance. The document follows a question/answer format.

How Were These Driver Cheats Found?

Members of Futuremark's BETA program[3] first noticed how parts of the tests in 3DMark03 were rendered differently on different hardware. When testing NVIDIA hardware on 3DMark03 with so-called developer's version's free camera enabled, they noticed how some parts of tests were rendered strangely, and informed Futuremark of their findings. Futuremark investigated further and our findings show that certain NVIDIA drivers seem to detect when 3DMark03 is running and then replace the 3DMark03's rendering requests with manually implemented alternative rendering operations. These alternative rendering operations reduce the amount of rendering work and thereby increase the obtained benchmark result.

[1] Extremetech: http://www.extremetech.com/article2/0,3973,1086025,00.asp

[2] Beyond3D: http://www.beyond3d.com/#news5856

[3] Futuremark's BETA program is an open, fee based cooperation program between Futuremark and the PC industry at large. BETA program members have access to pre-release builds of upcoming benchmarks and to a so-called developer build. The developer build is exactly the same as the public version of the benchmark, but with additional functionality. Amongst other things, the developer build has a 'free camera' mode, where the user can manually move the camera around while the test is running.

Why Does This Matter – It Is Just a Synthetic Benchmark?

We acknowledge with great pride how big a role 3DMark has in the PC industry. 3DMark score has become perhaps the most influential metric of PC performance. Enthusiasts, professional hardware reviewers and OEMs all depend on 3DMark results to a great extent.

We have a tremendous responsibility towards our users, who count on us and on our products when making important decisions. Thus, it matters a great deal that no one is able to take advantage of 3DMark – or any other significant benchmark - with unfair means.

Well designed synthetic benchmarks are excellent tools to objectively compare performance and to reveal different architectures' strengths and weaknesses. Some commentators have argued for only using benchmarks based exclusively on games; however there are severe problems with relying only on this approach:

- Game benchmarks only demonstrate how the hardware performs for that particular game and do not indicate the overall performance of the hardware.
- Any cheats potentially included in drivers are much easier to hide in game benchmarks.
- Finally, synthetic benchmarks can stress the hardware in a variety of ways, allowing reviewers to explore performance in particular areas and extrapolate performance of a hardware for upcoming titles.

Why Is It More Difficult to Cheat In a Synthetic Benchmark?

Actually, it may not be more difficult to cheat in synthetic benchmarks than it is in benchmarks embedded in games. But one thing is for sure, it is much more difficult to cheat in synthetic benchmarks without being caught.

The reason is that a synthetic benchmark does not exist in isolation, but as part of a set of tools. These tools can be used to analyze the benchmark run for irregularities. Further, the tools can be used to expose the existence of the cheats – both ExtremeTech and Beyond3D have proven this point very effectively. Game benchmarks are not immune to cheating either. Just like drivers can identify when 3DMark03 is running, same drivers also can identify when a popular game's benchmark is running. The driver can even detect when the benchmark tool of the game is running to enable the cheats; the cheats are then disabled when the actual game is played. Thus, the benchmark portion can be made to run faster with cheats while the actual in-game play performance is not enhanced at all.

Game benchmark cheats are much more difficult to expose. Even with the presence of clues, such as reduced image quality or missing features, it is next to impossible to prove what is going on without the source code and game developers' active cooperation. The major problem with this is following: There are few if any game developers that want to reveal their intellectual property to third parties so that independent groups would be able to prove the cheats.

Furthermore, game developers are in the business of making games and they want to attract as broad an audience as possible for their game. Their success does not depend on making sure that potential cheats on their benchmarks are exposed, whereas it is a benchmark house's business actively audit drivers and expose any cheating.

Aren’t These Cheats Just Optimizations That Also Benefit General Game Play Performance?

No. There are two reasons.

Firstly, these driver cheats increase benchmark performance at the expense of image quality. Only the user and the game developer should decide how a game is meant to be experienced, and not the hardware developer. An act by hardware developer to force a different experience than the developer or the user intended, is an act that may mislead consumers, the OEMs and the media who look to our benchmark to help them make purchase decisions.

Secondly, in well-designed benchmarks like 3DMark03, all cards are instructed to do the same amount of work. Artificially reducing one card’s workload, for example, by using pre-set clip planes or using a lower precision shader against the program’s instructions, is only aimed to artificially manipulate the benchmark test result. Please note, that the cheating described here is totally different from optimization. Optimizing the driver code to increase efficiency is a technique often used to enhance game performance and carries greater legitimacy, since the rendered image is exactly what the developer intended.

What Are The Identified Cheats?

Futuremark’s audit revealed cheats in NVIDIA Detonator FX 44.03 and 43.51 WHQL drivers. Earlier GeForceFX drivers include only some of the cheats listed below.

1. The loading screen of the 3DMark03 test is detected by the driver. This is used by the driver to disregard the back buffer clear command that 3DMark03 gives. This incorrectly reduces the workload. However, if the loading screen is rendered in a different manner, the driver seems to fail to detect 3DMark03, and performs the back buffer clear command as instructed.

2. A vertex shader used in game test 2 (P_Pointsprite.vsh) is detected by the driver. In this case the driver uses instructions contained in the driver to determine when to obey the back buffer clear command and when not to. If the back buffer would not be cleared at all in game test 2, the stars in the view of outer space in some cameras would appear smeared as have been reported in the articles mentioned earlier. Back buffer clearing is turned off and on again so that the back buffer is cleared only when the default benchmark cameras show outer space. In free camera mode one can keep the camera outside the spaceship through the entire test, and see how the sky smearing is turned on and off.

3. A vertex shader used in game test 4 (M_HDRsky.vsh) is detected. In this case the driver adds two static clipping planes to reduce the workload. The clipping planes are placed so that the sky is cut out just beyond what is visible in the default camera angles. Again, using the free camera one can look at the sky to see it abruptly cut off. Screenshot of this view was also reported in the ExtremeTech and Beyond3D articles. This cheat was introduced in the 43.51 drivers as far as we know.

4. In game test 4, the water pixel shader (M_Water.psh) is detected. The driver uses this detection to artificially achieve a large performance boost - more than doubling the early frame rate on some systems. In our inspection we noticed a difference in the rendering when compared either to the DirectX reference rasterizer or to those of other hardware. It appears the water shader is being totally discarded and replaced with an alternative more efficient shader implemented in the drivers themselves. The drivers produce a similar looking rendering, but not an identical one.

5. In game test 4 there is detection of a pixel shader (m_HDRSky.psh). Again it appears the shader is being totally discarded and replaced with an alternative more efficient shader in a similar fashion to the water pixel shader above. The rendering looks similar, but it is not identical.

6. A vertex shader (G_MetalCubeLit.vsh) is detected in game test 1. Preventing this detection proved to reduce the frame rate with these drivers, but we have not yet determined the cause.

7. A vertex shader in game test 3 (G_PaintBaked.vsh) is detected, and preventing this detection drops the scores with these drivers. This cheat causes the back buffer clearing to be disregarded; we are not yet aware of any other cheats.
8. The vertex and pixel shaders used in the 3DMark03 feature tests are also detected by the driver. When we prevented this detection, the performance dropped by more than a factor of two in the 2.0 pixel shader test.

We have used various techniques to prevent NVIDIA drivers from performing the above detections. We have been extremely careful to ensure that none of the changes we have introduced causes differences in either rendering output or performance. In most case, simple alterations in the shader code – such as swapping two registers – has been sufficient to prevent the detection.

What Is the Performance Difference Due to These Cheats?

A test system with GeForceFX 5900 Ultra and the 44.03 drivers gets 5806 3DMarks with 3DMark03 build 320.

The new build 330 of 3DMark03 in which 44.03 drivers cannot identify 3DMark03 or the tests in that build gets 4679 3DMarks – a 24.1% drop.

Our investigations reveal that some drivers from ATI also produce a slightly lower total score on this new build of 3DMark03. The drop in performance on the same test system with a Radeon 9800 Pro using the Catalyst 3.4 drivers is 1.9%. This performance drop is almost entirely due to 8.2% difference in the game test 4 result, which means that the test was also detected and somehow altered by the ATI drivers. We are currently investigating this further.

Why does this matter?

The latest version in the 3DMark series, 3DMark03, was developed for performance measurement and comparison for the DirectX 9 generation of hardware. All major graphics hardware and CPU manufacturers participated in specifying and testing 3DMark03, in order to ensure that it is an impartial tool that correctly measures the PCs performance in real-time 3D applications, like 3D games. The IHVs see the existence of such an impartial tool so important that they participate in the funding of 3DMark development, by joining the Futuremark beta program.

Games strive to above all offer the best gaming experience, which many times means, that the game needs to be run differently on different hardware. This in turn means that many game benchmarks run differently on different hardware, or in other words submit a differing workload on different hardware. This kind of game benchmark does measure how fast that game runs, but it does not offer a general hardware performance comparison.

3DMark on the other hand is designed for this very purpose. If the drivers do their job correctly, 3DMark produces the precisely same rendering on all hardware. Some hardware with a more limited feature set actually do more work in order to achieve the same rendering, but the resulting rendering is the same, the workload is comparable, and the performance result is justified as a world wide standard.

Is This Just Futuremark Taking Revenge on Nvidia for Leaving the BETA Program?

No. Our only purpose is to maintain the integrity of our products, and so ensure continuity of our business. We treat the BETA program and the whole industry very seriously; Beta Program members include leading companies of the industry such as Microsoft, Dell, AMD and Intel.

Our business relies on our ability to produce unbiased benchmarks that enable apples-to-apples comparisons. Any company that deliberately tries to fool our products is actually trying to take an advantage of our credibility. We will not tolerate such actions from anyone.

What Happens Now?

When 3DMark03 is altered slightly, NVIDIA drivers do not recognize 3DMark03 anymore, and the performance drops. The same slightly altered 3DMark03 version can be run on other hardware and the results remains the same.

Futuremark has prepared a second patch for 3DMark03, Build 330, where these preventative changes are implemented. We hope this will encourage the whole industry to release drivers that render the tests as 3DMark03 intends them to be rendered, and which produce results that can be fairly compared to those obtained with other hardware. It is possible that new drivers that cheat also in the new 3DMark03 build may be released in the future. We sincerely hope this will not happen. We strive to provide all BETA program members with robust tools to detect such situations.

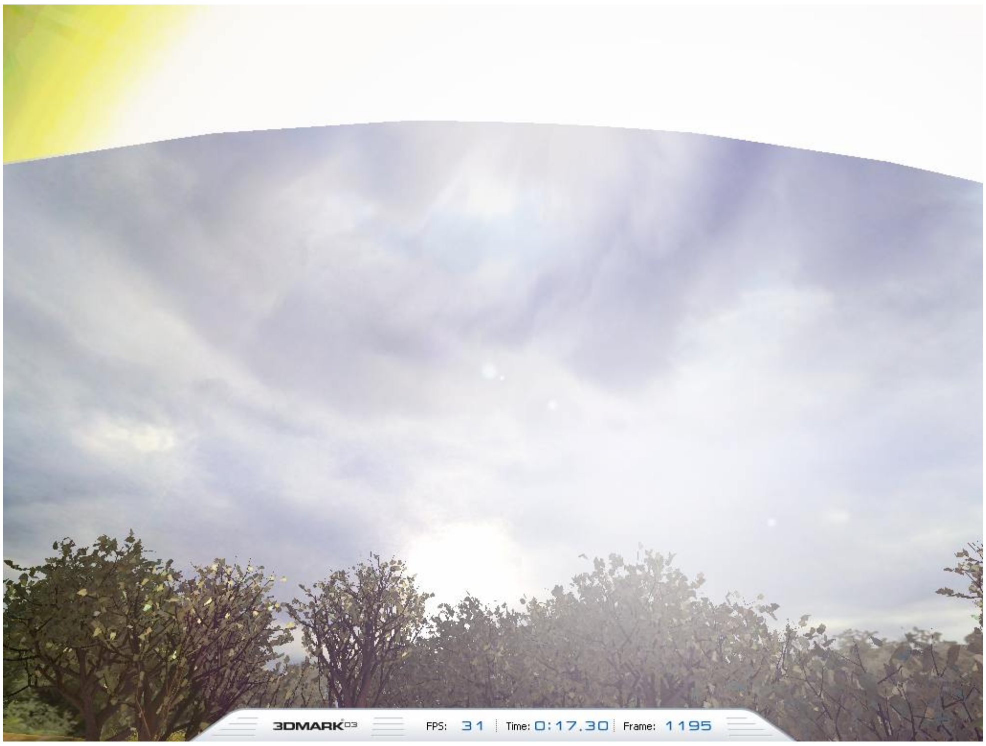
Can 3DMark03 be used as a reliable benchmark for DirectX 9 generation graphics cards?

Yes, with the new 3DMark03 build 330, it can. We recommend benchmarking GeForceFX cards using the NVIDIA drivers 44.03 or 43.51, and the new 3DMark03 build 330. We also recommend benchmarking Radeon 9000 series cards using the ATI Catalyst 3.4 drivers, and the new 3DMark03 build 330. These combinations run the 3DMark03 tests correctly, according to how 3DMark03 instructs the hardware to render the 3D scenes.

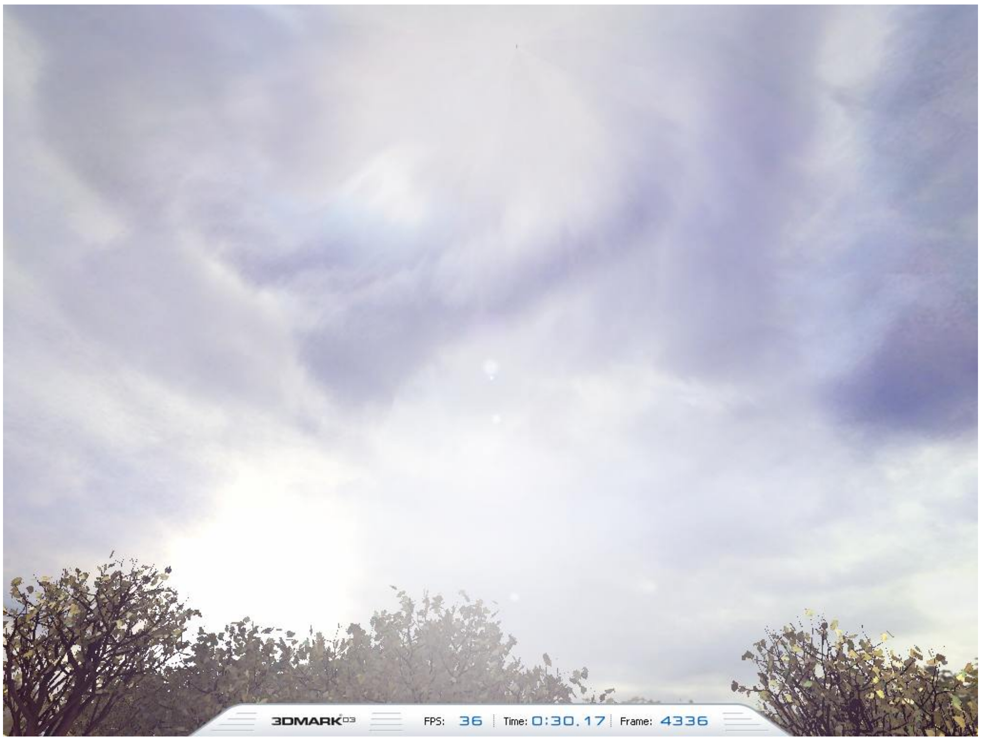
Appendix – Reference Images:

Following screenshots are provided as reference examples to illustrate the issues highlighted in this document:

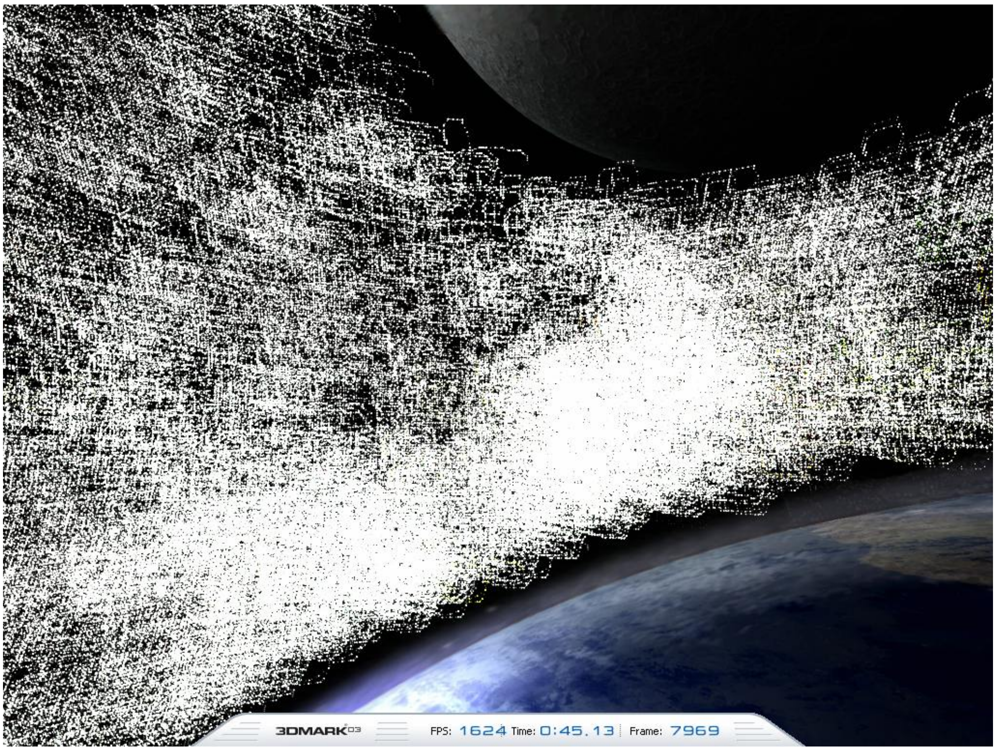
1a. 3DMark03 - Game Test 4 rendered incorrectly with clip planes and no buffer clearing.



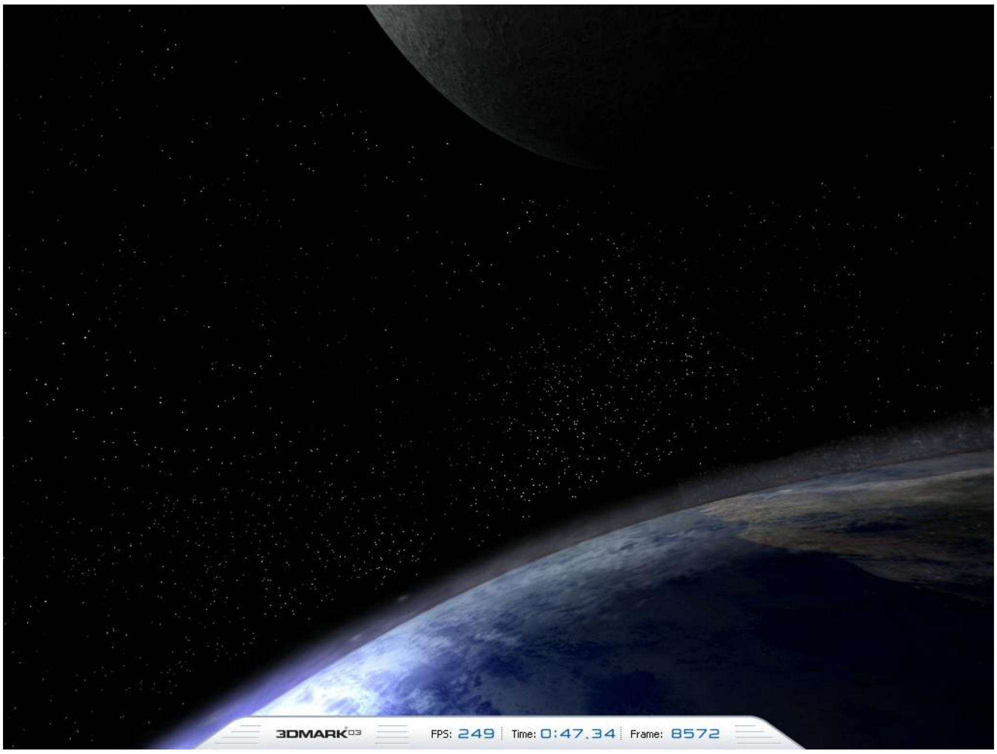
1b. 3DMark03 - Game Test 4 rendered correctly.



2a. 3DMark03 - Game Test 2 buffer clears not executed properly.



2b. 3DMark03 - Game Test 2 buffer clears executed properly.


3DMARK03
FPS: 249
Time: 0:47.34
Frame: 8572