# Attention:

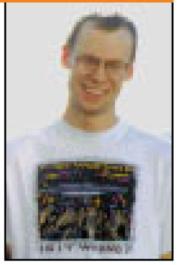
Thank you, and I hope you enjoy the article,

Chris Hecker
definition six, incorporated
checker@d6.com
http://www.d6.com/users/checker

PS. The email address at the end of the article is incorrect. Please use checker@d6.com for any correspondence.

SOAPBOX

by Chris Hecker



# Developer Power and the 'U' Word

I'm going to talk about something I feel very strongly about. It's a subject that might get me laughed off the stage as some kind of wacko, or at best a hopeless idealist. We'll see.

*Editor's Note: This was originally presented as a speech at the Seattle Computer Game Developer's Mini-Conference in November 1997. It has been edited for length.*

This topic is much bigger than texture mapping, rigid body dynamics, or any of the technoid topics I've written about in the past. If you put my thoughts on this topic into product development terms, they would be in the pre-alpha stage, so bear with me, and don't expect a completely polished and airtight presentation.

I'm going to talk about power and control in the game industry.

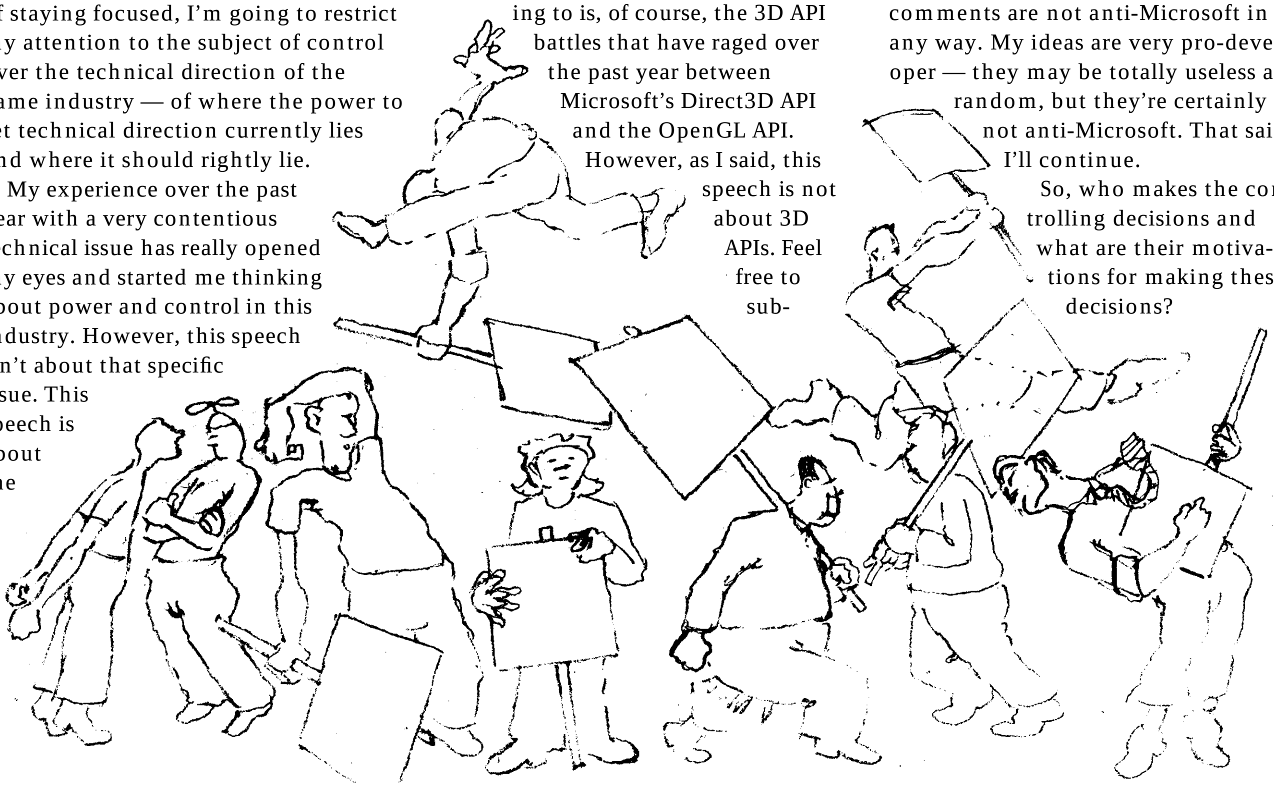
This is a huge subject that permeates every level of this industry, from the artistic to the financial. In the interest of staying focused, I'm going to restrict my attention to the subject of control over the technical direction of the game industry — of where the power to set technical direction currently lies and where it should rightly lie.

My experience over the past year with a very contentious technical issue has really opened my eyes and started me thinking about power and control in this industry. However, this speech isn't about that specific issue. This speech is about the meta-issue that was underlying it at all times, but was never brought to the fore and discussed in any meaningful way. Because I can't claim to have all the answers at this pre-alpha stage, I will simply bring the meta-issue into the spotlight where we can hopefully have an intelligent discussion about it.

The contentious issue that I'm referring to is, of course, the 3D API battles that have raged over the past year between Microsoft's Direct3D API and the OpenGL API. However, as I said, this speech is not about 3D APIs. Feel free to substitute in your own contentious technical issue (whether it's Sega's choice of 3D chip in their next-generation console, Nintendo's choice to use cartridges, Intel's MMX instructions, or something else), and I think the meta-issue will stay the same: who makes the controlling decisions and what are their motivations for making these decisions?

Before I answer this question, I feel compelled to give a sort of disclaimer. A couple of people have tried to paint me as being "anti-Microsoft," and this simply isn't true. It's a shame that I even have to mention this, but I think if you really listen to what I'm saying with an open mind, you'll see that my comments are not anti-Microsoft in any way. My ideas are very pro-developer — they may be totally useless and random, but they're certainly not anti-Microsoft. That said, I'll continue.

So, who makes the controlling decisions and what are their motivations for making these decisions?



*Chris Hecker is the technical director of definition six inc. He can be reached at checker@d6.com.*

With regards to my specific example of the 3D API battles, the answer to the "who" part of that question is Microsoft. Microsoft decided to make Direct3D the "official" 3D game API for Windows. Now, as for their motivations, we could formulate various theories, but as I said, this speech isn't about the 3D API issue. Beyond any specific motivations, what is the underlying motivation at the very base of all the decisions made? There only can be one answer for any well-run, publicly traded company. That motivation is raising the stock price. Of course, there are other secondary motivations (especially at the individual employee level), but at the base of it all, Microsoft, SGI, Intel, Nintendo, Sega, Sony, IBM, and all the rest must make decisions that will raise their stock price, or they'll be out of business.

Should we hate Microsoft for its decision regarding 3D APIs? Should we think that the company is inherently evil or engaged in some sort of Usenet conspiracy hijinks? No. I certainly don't hate Microsoft for making this decision. The company has a fiduciary responsibility to its shareholders, and business plans and technical directions undertaken by Microsoft must uphold this responsibility.

However — and this is the crux — just because something is a good direction for Microsoft or another large and influential company does not necessarily mean that it's a good direction for the game industry. Why should it be? They're completely separate entities. Sure, sometimes (and possibly, the majority of the time) directions taken by large corporations will be in line with what's best for the game industry, but sometimes they won't.

Which brings us to my thesis: it's our failure as game developers when a direction taken by an influential company is not in our industry's best interests.

How can it be our failure when we aren't the ones making the actual decision? Because we failed to stop that decision from being made. We failed to change the decision into something that is best for us as game developers, rather than best for that company's shareholders. Rather than the company's direction changing because we, its customers, demand it, we settle for a different direction because the company demands it. That is 100% backwards from the way it should be. Rather than the industry setting the direction, the company sets direction and the industry is only allowed to provide feedback on that direction — feedback that may or may not be heeded. The final decision lies with the company. Again, 100% backwards.

Remember the last time that you were sitting in your office trying to write to some awful API or instruction set, and you started ranting about how the person who designed it had clearly never written a game and how you could design a better API in your sleep? And remember how your coworker told you to shut the hell up and get back to work because that's the way things are in this industry? Well, guess what. You were probably right and your coworker was wrong.

The secret to turning this situation around is understanding where the real power in the game industry lies. The fact is that power lies in our hands as the creative force driving this industry. All of this money and politics and business exists because we make games that are fun to play and that people buy as a result. Without our games, the industry collapses. That's why we have the real power.

I've seen glimpses of this power in the past year. One time was when I organized the OpenGL letter from game developers asking Microsoft to support OpenGL on Windows 95. Now, you might ask, what good did that letter do, since Microsoft didn't actually end up supporting OpenGL on Windows 95? [This was written before the joint Microsoft/SGI OpenGL announcement in December 1997, so maybe the letter did have some positive effect after all! - Ed.] But you would be overlooking the real effect the letter had. No, Microsoft didn't heed the letter's demands, but the letter did wonders for the morale of game developers interested in using OpenGL, and it let people in isolated pockets know they weren't so isolated in the industry after all. It showed IHVs that developers were serious about using OpenGL for games. And, of course, it brought the press spotlight onto the issue.

The reaction to the letter indicated to me that the power to change the direction of the industry for the better truly does lie in our hands. However, tapping into this power is extremely difficult for two reasons.

First, we only have true power when we act collectively, and getting people to think bigger than themselves or their current project or their company is very difficult. We need to start thinking about what's best for the industry globally and in the long run, in addition to how you can get these bugs fixed or make your milestone on Friday.

If we can't make that leap, then others who can think strategically (such as, say, a large, well-organized company) will control our destiny forever, and we should stop complaining about it. If we're not going to do something about it, then we forfeit the right to complain.

Second, even once people are thinking in this long-term and global mindset, it's incredibly hard to get them to spend any time doing anything about it. I often hear developers say, "Sure, I want to use OpenGL in the long run, but I've just got to ship this game before I can think about it." Well, there might not be an OpenGL in the long run if developers don't actively support it. This is true of any technical issue, not just 3D APIs. There will be a small window in time in which to act for the good of the industry. If we miss it, we'll live forever with whatever technology is being pushed, whether it's the best technology for our industry or not.

Let's make a flying leap and say that I've convinced you that we really do have this power, and you're now thinking globally about various issues that affect your daily job and this industry as a whole. Where do we go from here? I think the raw energy to harness the power is already here and being spent on a daily basis. Every time you rant about some brain-dead technical direction in which we're being led, that's energy that we could theoretically apply to fixing the problem. Most developers rant like that every day, and that's a lot of energy that's just dissipating. We need to focus it.

One way to focus that energy is to form ad hoc interest groups, as I did for the OpenGL issue. I got on the phone and e-mail and organized a ton of game developers, and one of the results of that organization was the letter. This effort worked and it's still working, but it's incredibly tiring. There would be days when I was supposed to be working on our game, but instead I'd spend all day on the phone or answering e-mail. I care deeply about this issue, but I can't let it ruin my company, so I'm forced to budget the time that I spend on it. I like to think that if I could have workled full time at organizing people, there wouldn't even be an issue anymore — we'd have won already. However, there's no way I could spend all of my time on it, so I'm sort of forced to do a half-assed job if I want our company to survive.

I'm not the only one who's tried to organize ad hoc technical committees. Zack Simpson of Titanic Entertainment, who used to be at Origin, tried twice to do this — once with sound standards a few years back, and again with joystick APIs. Like me, Zack found that it takes an incredible amount time to organize people and get results.

One could argue that the reason that it takes so much effort to organize people on these issues is because people really don't care, but I don't think that's true. People do care, and if someone's doing the dirty work of organizing (such as me, Zack, or someone else), then developers are willing to spend a bit of their own energy. It's just that there's such a huge busywork barrier to entry for setting up this leverage that it rarely happens.

So, I think the ad hoc organization isn't a viable model in the long term. I think that we need a more persistent body, one where the initial costs of setting things up can be amortized across a number of issues: a body that has respect so that these companies pay attention to its demands, and a body that has teeth for when the companies don't pay attention.

Now, this is the part where you want to get the rotten tomatoes ready. I'm going to use the "U" word. Union.

Now, of course I don't mean a union in the full Jimmy Hoffa-sense of the term, since this isn't a labor-management situation. Mostly, I use the term union because it seems to have some mystical power these days: people either hate them or love them. Hopefully, I got your attention.

However different our situation may be, if you look at some of the words and concepts that I've been using, such as "a group of people who are powerful together, and powerless when separated," "collective bargaining," and "putting teeth into our demands," you start to see some similarities. If you don't like the term union, you could call what I'm talking about a standards body with the means to enforce its standards.

This would be a body made up of people who have a vested interest in seeing the absolute best thing done for the industry, and that means us developers. If we were well-organized, we could easily force companies to do the right thing, rather than hoping and praying that they do the right thing, as it is now. The teeth we would wield would be everything from simple endorsements, logo programs, and press releases (you'd be surprised by how much these large companies fear bad press!) all the way to boycotts or working with a competing vendor to shut out a disagreeable company.

So now what? How do we move forward? This is where it becomes evident that my thoughts are still in the pre-alpha stage. I can think of a number of organizational models, such as companies pitching in money to fund the organization and developers volunteering their own time. I will think about this some more, and I encourage you to think about it as well. I think the time has come for this idea, but it's not going to happen unless we make it happen.

And now for the requisite quotations:

Voltaire said, "The best is the enemy of the good."

This quote is apt for our situation, but John Miles rephrased it recently for the Direct3D/OpenGL battles, and I like his version even better.

Miles said, "The good-enough is the enemy of the excellent."

I don't know about you, but I'm sick of settling for good-enough when it's well within our power to have excellence. ■