Bennett Foddy on the pitfalls perfectionism

March 1, 2018 -

As told to Brandon Stosuy, 4122 words.

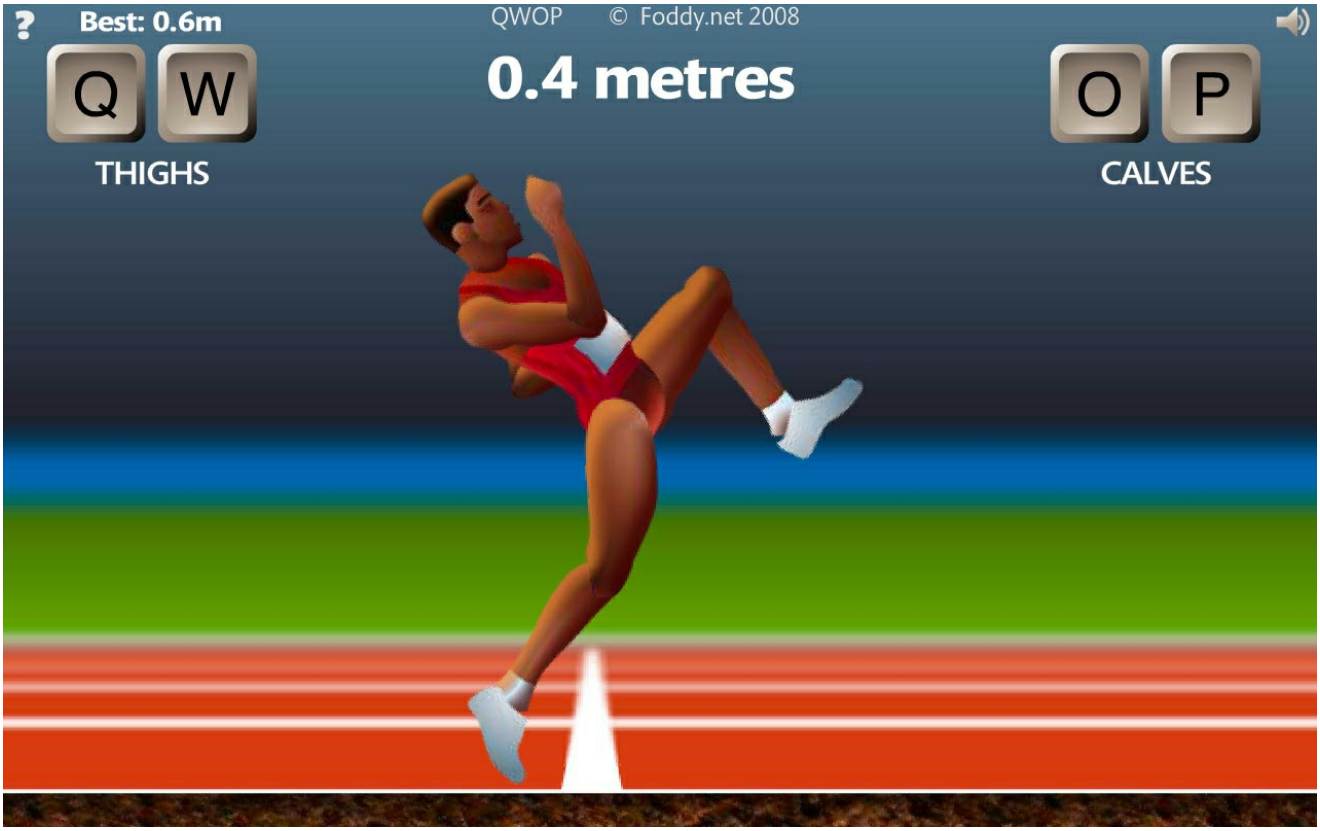
Tags: Games, Process, Focus, Education, Success, Independence.

You trained as a moral philosopher, and you played in the band Cut Copy. What made you gravitate towards making games and teaching about games?

I've always been more comfortable in an intuitive problem-solving space than a painstakingly logical one. I think that's true in my video game practice. There are certain people who make video games in an engineering way and a programming way. That's valid, but you can also make games in a free-associating, intuitive, freewheeling style. That's the kind of creative space I'm interested in.

As a musician, I was the bassist. I was not writing the songs. So that's a different kind of creative expression. It's one where you're rehearsing and you're trying to pull out creative expression from a rote task which, again, is totally valid and something that some people do beautifully. But, where my own creative process is most satisfied, where I'm most functional, is in a place where I'm able to make intuitive connections and work emotionally and in a flexible way.

Maybe it's weird to have picked games out of all of the creative disciplines because it's not the most free. You're constrained by technology and you're constrained by engineering in some ways, but I just have a deep love of games. Out of all the things I was involved in, that was the one that won me over.



I imagine there being more immediacy to games than those other disciplines. Is there a thrill to being able to make something, put it out, and get feedback right away?

Yes, especially with the web. The web enables a form of expression in creative arts that it doesn't enable in science or philosophy, or something like that. When you write a philosophy paper, the process is something like this: Come up with the idea, do a bunch of research, draft up a "version one" in the argument, workshop that, start showing it to people. Then they give some feedback, you make some changes, and when you're completely 100% happy with it you start sending it off to journals. The journals are all looking for the greatest philosopher in the universe so you get more rejections than acceptances.

Each time that you send it off, it has to be sent around to peer reviewers; that can take six months easily. They can often come back with a rejection at that stage. So, just getting a place to accept a perfectly good paper can be a process that takes two years. Then after it's accepted, it can be another two years before it arrives in print. Then it can be another two or three years or four or five or six years before anybody's responded to it in print.

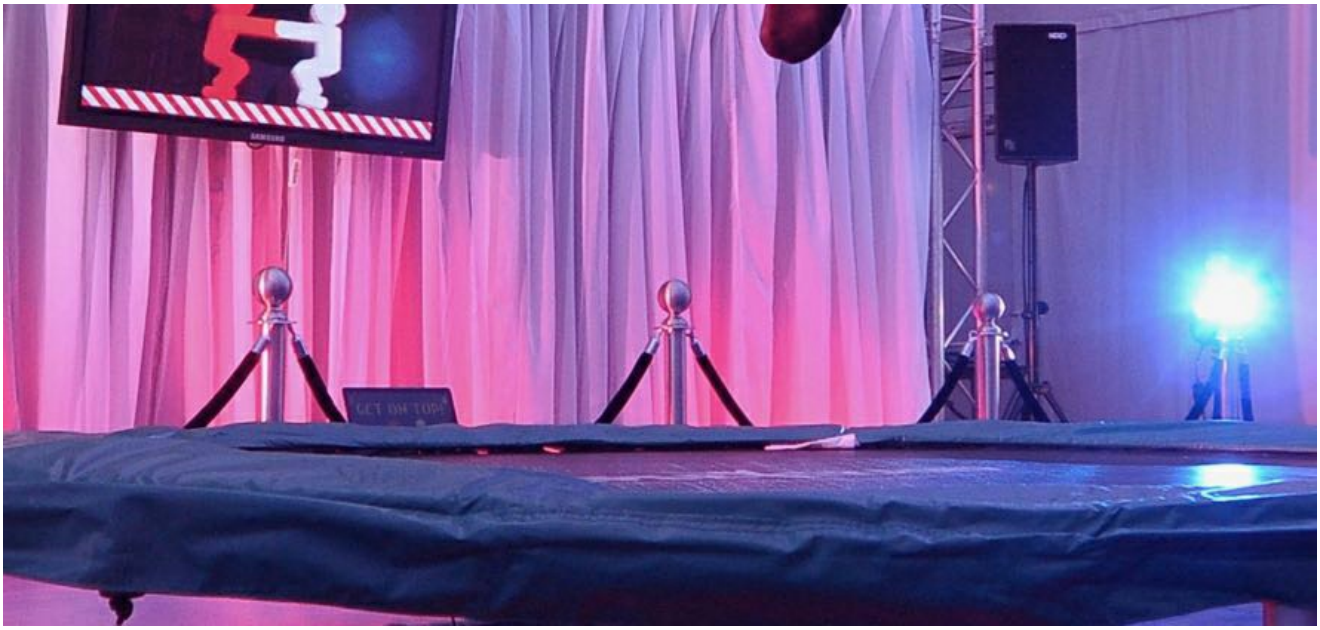
In the worlds I've been involved in—music, philosophy, science, games—it's all best done in dialogue with other people. I don't think any of these things are best done as single-player experiences. It's better to put something out there that's flawed and have people respond to it and criticize it and then come back to their criticisms with another piece of work than to try to sit and endlessly polish something until it's completely beyond criticism. I don't believe it's better to release something that is beyond criticism then to release something that's flawed. It's better to allow room for dialogue.

In academia, you're not really in dialogue until 10 years after you come up with your original idea. With a game, if I talk to someone and they give me an idea for a prototype, I can code that up in the afternoon and it can be online that day. I can have 100,000 people check out this prototype that day. There's almost nothing in the world that's as direct as that. Obviously, if you're a member of the press or a YouTuber or something like that, you've got that as a platform, but as a creative person to get creative ideas out there, it's just so amazing. That's motivating for me.

As more people know your work, do you find yourself being more cautious about just putting something out there?

I only keep about one in five of the prototypes that I make. A lot of them get killed in the crib. But, with the right framing you can let people know, "Oh hey, this is something I spent three hours on." Manage expectations. I feel comfortable that people generally understand the difference in a thing you worked on for a week or a month. It gets fuzzy for players of games as soon as you get over the six month mark; to most players, it then goes into a category of "professional product," and they don't care at all whether you worked on something for six months or seven years. It's the same kind of product to them. But in the smaller scale, I think it's very easy for people to understand that there's going to be rough edges on something that they know was made in three hours.

It's one of the hazards of creative work, that there's that kind of paralysis of feeling pressure. A lot of the time, my most talented students feel a lot of anxiety about their first release. I'm always saying to them what was said to me when I was starting out: Nobody is waiting for your first release. Nobody is expecting it to be good. You can get a lot of encouragement, but that doesn't mean anybody is going to be disappointed with its shortcomings.



On the other hand, as people gain more of a following, they may feel pressure to meet expectations. “Everybody loves my previous thing and I don’t want to disappoint people. I don’t want anyone to say, ‘Oh, he’s jumped the shark.’” There is a kind of paralyzing fear that can come from that.

Early on in my philosophical career, I was on the path to having a severe case of that. Before I was a grad student, I was working as a research assistant for a philosopher. He wanted me to come up with ideas for papers and pitch them and then co-write them with him. He wanted to publish as much as possible and he wanted to do it collaboratively. He said, “Your job is to facilitate publication.” But he was also very busy and had a lot on his plate.

One time I had an idea for a paper. I wrote not much beyond a list of bullet points, just ideas for arguments, and I sent it to him to get his “ok.” Like, “Should I work on this?” Instead of reading it, he forwarded it to the review panel of a major journal. Without my knowledge, they sent it off for peer review to famous professors in the field. I was so mortified. I just thought, “Ah, this is the end of my life. Everybody’s going to think I’m a joke. I can never come back from this.” I was so stressed out about it, but nothing happened and I kind of forgot about it. I thought they must have just thrown it immediately into the trash. Then, something like eight months later, I get this email that’s like, “We’ve reviewed your paper and we found it could be published with revisions.” [*laughing*]

I’m like, “No shit it needs revisions.” I didn’t wind up publishing it, but it was important to me to have my work be realized and realize it’s really not bad and it’s not a problem. If you release mediocre work nothing bad happens. The model of somebody who only ever releases flawless gems is so rarely born out in reality.

Do you share your work with people while you’re working on it?

We’re always yelling at students for not play-testing their games. People generally avoid tasks that are emotionally difficult even if they’re necessary. Showing people your work and getting them to play it is emotionally difficult.

Video games, as an industry, is very much in favor of vigorous play testing as a way of improving your design. You make some changes, you play test them, you observe the play tester’s data, and you iterate your design based on the play tester. That’s the basic loop of game design and development. It’s a set of ideas that comes out of the software industry, originally—”user testing.”

I’m trying to strike the right balance between letting people play the work, not being overly protective of it, and at the same time not letting that influence my thinking very much at all. It’s a very difficult balance to strike.

Trying to settle into the right habit where I’m mainly responsible for play testing my own game, but not phobic about releasing it.

As someone gets more and more skilled at programming, how do you avoid going down endless technical wormholes?

With any creative art that has that technical component—music and painting, filmmaking, and especially with games where the technical component is so much at the forefront—there’s this risk. It’s not just perfectionism. Even if you’ve managed to steer away from perfectionism, there’s a process by which you gradually become more and more enamored with craft. As you get better able to make the computer do what you want it to do to get your ideas down, you become confident that the idea that you originally had can be turned into a living video game. That is its own kind of hazard.

Even if I’m not being perfectionistic, I can easily fall in love with technical proficiency and virtuosity and the desire to perfectly architect my code and to have the best looking grass in my game. To have perfectly optimized code, maybe I’ll write my own game engine that does it exactly how I want. It’s just an endless time sink. It can be where all of your energy on a project goes because there’s really no end to good craft in any field. It can be a thing that blocks you from releasing work, but I also think it can become overwhelmingly a thing that becomes the focus of the work and by-and-large not what’s of interest to the audience.

Caring about craftsmanship is laudable and good, but I think in most cases your audience doesn’t care about it as much as you think they do. Especially as your appreciation of craftsmanship improves over time. The more that you build video games, the more you become attuned to what people are doing well from a craft perspective. When you play other people’s games you want to do that yourself and you learn how to do it. It’s all super attractive. And you just have to remember and connect to the fact that the average player just really doesn’t care.

If you look at the top games of the last year *PlayerUnknown’s Battlegrounds* would be the big one that’s up there that’s made the colossal amount of money. It’s ugly, unfinished, and full of bugs. And it has been for the entire duration of its popularity. Why is it popular? It’s got good ideas and it shows some kind of love and some weirdness and it’s direct. There’s a lot to love about that game, but craftsmanship is not part of it.

I think of games primarily as works of art. But if it doesn’t run on my computer, it’s not a work of art, it’s just a lump of ones and zeros. It doesn’t do anything. And so ultimately there are some clear lines where you’ve crossed over where a little bit of attention to craft would have helped.

As you get more expert in the creative practice of making games, your aesthetics, your sense of what counts as sufficiently good software is constantly shifting. Things get more achievable for you, and as you look at more people’s work and compare your own work to it. That’s just a process I find myself constantly trying to defend against.

When I was working on my most recent game, *Getting Over It with Bennett Foddy*, I was aware the whole time that I had to willfully resist my desire to rathole myself on small things. So in it, I have some kind of model for a table that's part of the level that you're playing in and the model has too many polygons and too many materials. What I want more than anything is to spend the whole day optimizing the model in Maya, or something like that. I want to fix every single thing about it, every single vertex. That's not progress. That's not part of being creative. It's just me settling into a very comfortable technical process where I should be focused on a creative one.

My process there was to be extremely strict with myself and try to notice when those things were happening and cut them off immediately. But it's sort of unnatural in a way. What would be natural for me right now would be to settle into nothing but a black hole of solving small technical tasks forever.

As you've gone on, has getting better with time management helped you survive? It's pretty common for students, and people just starting, to pull all nighters. That doesn't seem sustainable.

I was exactly like that as a student. I did everything at the last minute. And, yeah, a lot of my students are like that. A lot of the time they're just over-scoping. It's a very classic problem, and it's a problem that is solved somewhat by experience. You start to learn how long things take. But there's some things that are just much easier to estimate than other things.

Getting Over It was very unusual both as a game and as a software product in the sense that it's mostly statics. There's this one system which is the way that you move the guy's body, and all of the obstacles in the world, give or take, are static. They don't move and react to you. There's nothing you can do to influence them in any way.

That drastically reduces the combinatoric explosion of problems that you can have in making a game. I shudder to think what it's like making a game like *Minecraft* or *Spelunky*. Every little system that you make a change to can influence every other system, and you can get all these kind of runaway bugs that you completely didn't ever think that you could have.

That's the stuff that makes estimating the time in engineering really difficult. So part of being able to know what I was doing was about having designed something that was mostly statics.

I imagine that it’s the same for some other creative arts. For example, for painters I’d say, “How long will it take you to finish this painting?” If somebody’s halfway through a painting, I think they can probably give me a pretty good estimate. There are some things where there’s not going to be too many shocks along the road.

It was interesting to me how nailing a bunch of that stuff down involved being able to make it work in a freer and more experimental way. I had the security of knowing that it wasn’t going to spin out of control.

Something that’s in my work now: If I can’t clearly see something that I’m trying to build, I don’t proceed. I’m at peace with the fact that sometimes, through the process of building something, the concept will change. I’m not rigid about that, but I do feel like if there isn’t a unified guiding light at each point of the design and development of a game, your risk of ending up with purple play dough is high.

You know, you keep on adding more and more colors of play dough together and it always winds up the same color. That’s definitely a risk of an iterative taste way of design process, something that doesn’t really have concepts at its core, or at least a skeleton of clear unifying concepts.

For the work that you do, what are the most valuable resources? Do you come up with ideas when you’re surfing the web? Are you pulling from things you’ve read?

I don’t remember if this is an influence from being in a sample-driven electronic band, but my view has always been that if you don’t have any good ideas, what you should do is hit the stacks. In music, you go and listen to some old records. I definitely do that in games, and I always have.

I’ve been in love with video games since I was two years old. I still play a lot. I feel like there’s this huge diverse evolutionary tree and a lot of branches bloomed and died. There’s a lot to be revisited and reused. Something I worry about is erasure of ideas in games, and I think that being literate and conversant in the ideas of the whole history of the form is the way you can keep that alive and pay respect to it. That’s a huge part of where I’m coming from with my ideas.

The other thing is the community in New York amongst students and other indie game developers… I’m constantly just playing new work as well. I’m a huge proponent of trying to stay current in whatever field that you’re in.

I had a moment that was very impactful on me. In about 2008, I was talking to a musician from Australia. He was classically trained, and he was doing really well. While talking to him, I thought about all of the classically trained musicians that I knew from when I was in college. I would always go to parties of wannabe concert violinists. I had this clear memory of being at a violinist’s party in sophomore year of college. There was this weird pallor over the whole thing because they all know that they’re not going to get to be what they want to be. There’s just not enough spots. You can only have that small handful of concert violinists, soloists, in the world. Maybe it’s not like this in that scene, maybe it was just the party I was at, but it felt like a weird jealous and downbeat vibe.

So I was asking this guy knowing that he must have been to school with similar people in similar situations. I asked him: “What do the people that you knew in college think of your hit record?” He was quiet for a little while and then he was like, “Those people don’t listen to music.” [*laughing*] That really rung true for me.

I don’t want to be like that. I want to stay current. I want to be abreast of what other people are doing. I feel like if I want anybody to play my games, if I have any hope of, especially other designers are playing my games, then I have to play their games. It’s something I’ve put a lot of time into. It doesn’t feel like an imposition. When you’re busy it is, but you need to make time for it. It’s not something that fits into a professional person’s life very well especially since we culturally define games as down time. We define them as wasted time.

If you’ve got a deadline looming, even if you’re a game developer, it can be really difficult to make time to play games that are coming out. We’re in a weird period of time where on Steam there’s like 15 new indie games minimum every single day. You can’t possibly play them all, and you start to feel like, “I’m not even gonna bother to try. I’d be better off spending the time working on my own game.” I’m against that. I think it’s on you as a creative person to stay current in the field. That’s where I think a lot of my ideas come from. Hopefully, my games are in dialogue with the things that are coming out right now as well as older things that are forgotten.

People describe your games as difficult or complicated. Do you embrace that?

There are things about difficult games that I’m interested in. For instance, it’s easy to produce certain kinds of emotions, a sense of high-stakes. But, hopefully they’re not ever difficult to understand. I think even when my games are surreal or absurd, it’s easy to read, hopefully, what I’m trying to do with them. I will say: I don’t think that you need to be able to be good at my games in order to get the value out of them. To be able to get the

value out of them, you don't need to be good at them. Richard Hogg is a designer of games like *Hohokum*. He described my games as being indifferent to the player. I think that's where it's helpful to break down what people mean by "difficult."

One of the things about my games is that they're indifferent to whether you get to the end or not. That's really kind of a reasonably usual emotional tone for video games. When you're making free games, or even if you're framing a bigger game like *Everything* in the right way, the sense of being able to play it for a while and just stop, there's nothing incomplete or shameful, or failure-oriented about that. It's a complete way to play a game.

It's one thing if you're trying to make a video game that tells a story, and then it's something that clearly is a design failure and a play failure because you never get to the end of the story. People walk out of a movie or they walk out of a musical performance; it's a failure in that kind of sense. But we have other sorts of media where that's just not the case.

If I'm walking through an art gallery looking at paintings, it's not like the painting fails if I didn't stand there for a certain amount of time. I can digest as much of it as I want. Something that I have tried to bring out in my games is definitely a sense that you can stop, partly as a way of reframing what it means for them to be difficult, but partly because I think there's kind of an interesting tension there because I want people to think about why they're not stopping with my games.

I've heard people saying that they've played *QWOP* for three days without stopping. And they know there's nothing at the end to see. Yes, there's an end of the track, but it's not like when you get to the end that's the moment when you can be like, "Okay, I'm done with this game." If you're playing it for three days straight, you are kind of addicted to it.

To me, it's interesting to work in a medium that has this sort of addictive nature to it, this compelling nature. But also to make work that's like, "Hey, you don't have to play this. You can just walk away at any time." And then the player's like, "No. I've got to do this" That changes their relationship with the game.

I don't want to ever feel like I'm begging people to play a game. There are games where you try to quit and it's like, "Are you sure you want to quit?" Like, "No. Don't." That's too thirsty. You shouldn't be framing it that way. I'd prefer the feeling to be like it's a party that doesn't end just because you left.

Bennett Foddy recommends:

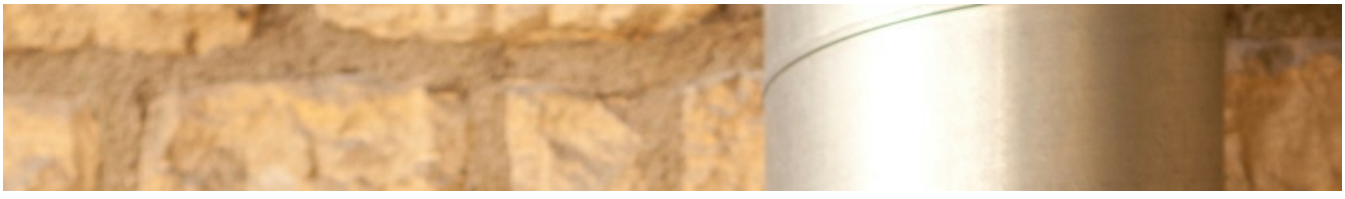
An installation of MAME or SDLMAME configured just the way you like it, with system ROMs for all the computer and console platforms. Then you can play almost any game created from 1975 to 1995, and learn only one arcane emulator interface.

An installation of FS-UAE to cover the one main blind spot of SDLMAME, the Commodore Amiga

Any videogame by Justin Smith, who has the exact punk élan that I aspire to.

Any videogame by Stephen Lavelle, who has the deep artistic seriousness that I admire but will never possess.

Warpdoor.com, a firehose of contemporary videogame inspiration.

Name
Bennett Foddy

Vocation
Game Designer, Teacher