



<http://www.eludamos.org>

**No B.S.: The Contemporary Practice of Game Education, Design, and
Development**

Ken S. McAllister, Judd Ethan Ruggill

Eludamos. Journal for Computer Game Culture. 2010; 4 (1), p. 117-122

No B.S.: The Contemporary Practice of Game Education, Design, and Development

KEN S. MCALLISTER AND JUDD ETHAN RUGGILL

This interview is excerpted from a series we conducted in 2010 with Quentin Rezin, a scripter/level designer at inXile Entertainment (<http://www.inxile-entertainment.com/>). Prior to joining inXile, Rezin studied computer science at the University of Arkansas and interned at Disney Interactive Studios/Buena Vista Games. He is currently working on Hunted: The Demon's Forge.

Whereas many developers simply fall into the job, you actually went to school specifically to become one.

Sort of. My goal was always to become a game developer, but the University of Arkansas doesn't have a game-specific degree program. My degree is a B.A. in Computer Science.

Did you essentially craft your own degree, then, making it tilt toward game development?

To a certain extent. The B.A. track in Computer Science gives a good background in computer use and good programming practices, while encouraging exploration of other interests. The electives I chose were things like astronomy, geology, drawing fundamentals, philosophy, literature, and foreign language.

The B.S. in Computer Science, in the Engineering college, is the primary track. The B.A. is a less common choice of majors for computer science students offered through the Arts & Sciences college.

My decision was driven by the freedom it would give me in choosing classes not directly related to the degree program. In my mind, a good game designer needs to be interested in a lot of different areas. Also, I knew I wasn't going to end up working as a programmer, but it's important to have a good base in the fundamentals of programming when working in a software development field.

So if Arkansas did not have a game development degree, how did you know to choose the B.A. over the B.S.? Where did you get your mentoring?

When investigating which school might be right for my chosen career path, I read a couple of books on "breaking into the game industry," plus I sent out 30+ letters to HR departments at companies I admired. I only got one response (the industry was pretty closed to outsiders back then), but it was from a designer who basically backed up what I expected – there really wasn't a "right" school or degree program for designers (she was actually an architecture major, as I recall). So, I looked at a

range of options, including Full Sail (in Florida) and the University of Southern California, and decided that what was right for me was to explore my own interests at my own pace.

Full Sail, for example, started as a film/music industry-directed school, and is more of a vocational school than a traditional university. When I looked into it, the game program was a spin-off of their 3D computer graphics program, so it was focused more on the art and animation side of game development than design. On top of that, I felt that it was important for me to get a true university education to round myself out.

Do you feel like your assessment of your own educational trajectory worked out well for you? Do you regret anything about the foci you chose?

I feel like my choices were perfect for me. I really enjoyed the university experience and the ability to choose classes that weren't necessarily aimed at game development, and I don't feel like my education left me lacking, either. My arts classes gave me experience talking to and thinking like artists, which is helpful in communicating with my team at work, while the computer science courses I took helped me improve my ability to identify and deal with problems. I also learned enough programming to do any scripting I might have to do as a designer (though I'm still nowhere near as skilled as a true programmer), and I've seen – through my experiences with other designers my age – that I didn't "miss out" on design knowledge on account of not having gone to a game development school.

However, university programs typically don't spend a lot of time on group work, project-length assignments, things like that. I was fortunate to be able to take part in a semester-long trial course on game development at the University of Arkansas, but it was self-directed rather than being taught by someone with industry experience. Some of the game development schools that are available these days are more project-focused – the game degree programs at USC and Carnegie Mellon each have a good reputation as far as that's concerned, so I think their students probably get more experience with actual development cycles. All the same, in my experience it seems like that knowledge is something that you can easily pick up on-the-job – of the seven designers on *Hei\$t*, the two Leads had experience and I had worked for six months at Disney, but two of the others had no industry experience or knowledge at all when they arrived at inXile.

What was that like?

My experience as a designer has pretty much been to show up, do the best you can at whatever tasks you've been given (which can sometimes be far afield of what you generally think of as design tasks), and if there's something you can't do, track down someone who can help or take over. One of my co-workers (one of those without prior experience) had pretty significant trouble coming to grips with milestone drift and occasionally being asked to start from scratch on work that he'd done, but I just accepted those as necessary evils of development. It's a flexible field, and you have to stay flexible to succeed in it.

And that kind of free-wheeling is acceptable to management?

For smaller developers, that's sometimes the cost of doing business. We're a relatively small developer for the kind of "big budget" kind of projects we take on. Everybody's got to adapt to the tasks we're given and take on extra responsibilities in order to get the job done. I feel like our development cycles might be smoother if we had the enough employees to specialize, but we don't. Diversity and flexibility are our management structure. Within reason, people can be asked to do anything at any time, and we have to adapt to those demands. On *Hei\$t*, I was actually tasked with writing a significant portion of the game's explanatory text simply because I wasn't doing anything more important at the time. Fortunately, I've had enough experience writing that our primary writer didn't have to re-do what I'd written.

Do you like the fact that you can be called in at any time to do anything? Or would you rather have a more well-defined job?

Hmm... I enjoy being involved in a variety of areas (such as being asked to represent the design team's interests on a special "boss implementation" task force – made up primarily of programmers, hence the need for a designer – or writing flavor text for *Hei\$t*, or being almost solely responsible for the tutorial on that project). At the same time, there are certainly moments when I'm asked to do things that really require a specialist, and I just have to do the best I can.

So, it's a trade-off, just like the one I had to make in deciding to go to a university rather than a game development school. But, again, I feel like I made the right decision. I've learned a lot about the various parts of the Unreal engine that I almost certainly wouldn't have if my job were more restrictive.

inXile seems to have an unusual business model, working on both AAA titles and casual games simultaneously. Could you speak a bit about that model?

Well, the publisher/developer model in general has always been fraught with some uncertainty on the developer end – you've got to always have money coming in, and if your publisher goes under or your game doesn't ship, it's a huge source of stress until another source of income becomes available. The inXile model allows for some more stability for the employees. We almost run like a publisher ourselves: on the one hand, we've got a team dedicated to whatever our current main project is, but by supplementing that with a good number of mobile/small-footprint games done by our other team(s), we're not put at such a disadvantage if something goes sour on the main project. It's operating with a safety net of sorts, although we certainly can't – and wouldn't want to – rely entirely on that backup income.

From the perspective of a developer, I feel like there's still a certain stigma attached to mobile/small-footprint development, but there's also a good chance to end up with low-investment-high-return products, and they're great for experimentation.

What is that stigma?

Mainly that mobile games aren't as "important" as high-profile console or PC titles. The casual market is still (deservedly or not) seen as dumping ground for shovelware as much as it is a market for niche games, experimental titles, and rapid development.

For our company, the keys are to be able to demonstrate the ability to put something cool together (the company put together a "vision target" demo, having the excellent contacts and salesmanship of our CEO, Brian Fargo, and a game idea that sits in an unexplored or underrepresented section of the chosen genre).

Is working on both AAA and casual games simultaneously becoming common in the industry?

To be honest, I'm not sure whether the combined AAA/casual games model is "common," but I suspect it's becoming more common than it used to be. With game budgets ballooning in the current generation, a lot of developers are either focused on working consistently with a single publisher (so that there will always be a contract to keep their employees' wages coming in) or will diversify, either through developing multiple titles or self-publishing smaller projects, like we're doing.

It sounds like, then, that money still is not king of all cosmos in the industry. Otherwise, the massive financial incentives behind mobile platforms would quickly override the cultural capital of dedicated console development.

You've hit on exactly the reason the management has set up the company the way it has. I was once told something like "we do the big budget, multi-year projects because we love to do them and they're what we want to be most well known for, but the money comes from titles like *Baby Pals/Purr Pals* and games developed for mobile platforms." I think that in our case, the shift came from the understanding that unless you're a Top 20 developer who could lose one publisher and be assured of picking up another one right away, things are going to happen from time to time on the big projects that are going to be a serious danger to a developer our size without that extra means of support. I think it was also motivated by the speed boost that the growing number of dependable tools and delivery platforms like Steam, XBLA, PSN, and WiiWare offer in turning a game concept into a realized, shippable product, plus the continuing swell of casual gamers who not only accept but *value* more reasonably-sized products with pick-up-and-play sensibilities.

It seems as if the decades-old question about whether or not computer games are art has now been decided both in and by the marketplace. The answer is definitively no: games are a business that employs people. The industry – not the art – must go on.

From a certain perspective, I suppose that would be the case. I feel that there are times when games are made because "we need to make a game" rather than because a company has a compelling idea. But I certainly don't think that's the driving force behind game development. At the core of every game developer is an instinctual drive to make a product that the end user will enjoy. That's true of producers, HR staff, and publisher employees as well as everyone making hands-on changes to the game. Games *have* to continue to be art for the simple reason that players demand the best possible experience and developers push themselves to deliver it. The game industry is still made up of "artists" (in the general sense, meaning anyone whose job it is to add creative energy to a product) – people who couldn't stop trying to push their art forward if they wanted to.

What would you say to the idea that consumers are increasingly less discerning about the quality of product they consume, and thus they do not need games to be art? They just need them to be cheap, moderately distracting, and reasonably accessible.

I think that if games ever stopped being art, they would necessarily stagnate, something that would be unacceptable to a large percentage of the player base, who would make their displeasure known. Before I got into the industry, I was hearing a lot of rumblings from my fellow players about how industrialized the games that were being released at that time were. I feel that the industry has responded with a fresh burst of creativity in the last several years.

It's interesting to consider whether consumer discernment of products has changed. You can look at titles which have become popular recently, things like *Cafe World* and *FarmVille*, etc., and it's not really as clear where the "art" in them exists. It's entirely possible that they're simply money-making tools with friendly window-dressing, and there might not be any "art" to them. But, at the same time, they facilitate fun for a massive number of players, so it's likely that "what is art?" is a different question for games than for other types of media. It's possible that games, by their very interactive nature and requirement of interaction with the person experiencing them must stretch the category of art into previously unexplored areas. Games like that may be "art" in that they excel at facilitating the experience their users are looking for. After all, as a social meeting-and-interaction experience, the difference between *FarmVille* and *World of Warcraft* may be one of degree rather than of intent or content.

Many people in the industry undoubtedly think the way you describe. And yet, despite their noble intentions, they are likely routinely made to work on things less ambitious in order to "earn a living." If so, then something like functional isomorphism comes into play: if it looks like a duck, and quacks like duck, it is probably a duck. That is, even if the developers all think they want to be artists in the industry, and indeed even if they think they are exercising some of their artistic sensibilities in their work, they are also in fact mostly perpetuating an industry that really only cares about getting product out the door and into the hands of consumers.

Yes, I concede that there's a certain amount of self-delusion there – as an "artist" and a designer, I have to believe that my actions and choices are well-directed toward improving the artistry (and critical and monetary success) of the project I'm working on. That may or may not be the case. But in my view, the creation of almost *any* art is both a balance and a struggle between creativity and salability. I may want to create something of artistic beauty so pure that the reality of it confuses and frustrates its audience. Or I may be making something so assembly-line that it hardly deserves to be called art at all. But the truth, in my experience, is nearly always somewhere in between. There's no line where "making art" suddenly crosses over into "making money." We, as developers, recognize that there's more than one factor in any decision we make. My team may be brainstorming some cool ideas, but we might decide not to implement one just as easily because "it doesn't support the vision" as we might because "it doesn't make good financial/scheduling sense." The difficulty of the matter is that it's a medium of both business *and* expression, and we have to

navigate the (sometimes choppy) seas of challenges in satisfying those two masters on a day-to-day basis. If you follow game news closely enough, it's as filled with developers pushing back a release date to have extra time to improve the end product as it is with projects getting cancelled for financial reasons.

Does the academic study of games ever figure into the way developers/producers conceive of the art and business of game making?

I actually just had an experience this past week where, in order to evaluate the effectiveness of the tutorial level I'm in charge of, I scheduled myself some time to study an article by James Paul Gee titled "Learning by Design: Games as Learning Machines," which presents an excellent series of criteria for evaluating the effectiveness of learning in a particular game.

The academic study of games is very helpful in some situations. I think, ideally, we'd *love* for the answer to be a straight "yes, it definitely helps," but a lot of developers either don't recognize the academic resources that are available to them or don't take the time to investigate them. I'm actually surprised that more of my peers don't avail themselves even of the educational opportunities that "directly" improve our effectiveness at our jobs, like sessions at the Game Developers Conferences.

To borrow what we discussed earlier, one thing that I notice frequently in the artists, programmers, and designers I work with is that almost all do what they can on a daily basis to improve their skills, both because it improves their "art" and because it makes good business sense if they ever find themselves looking for work. For the designers, this generally means digging into academic explorations of high-level design topics such as *Rules of Play: Game Design Fundamentals* (Katie Salen and Eric Zimmerman, Cambridge: The MIT Press, 2004) or *Gender Inclusive Game Design: Expanding the Market* (Sheri Graner Ray, Boston: Charles River Media, 2003), while the programmers read articles and conference presentations on programming techniques and improving the speed of computation-heavy rendering or AI processes, and the artists are always digging into theories of form and the psychology of art.

So there is time for self-improvement during off-times?

Sometimes. Some companies pad the beginnings of their schedules with lots of time for experimentation and coming to a better understanding of the project, which I think smoothes the development process. But not every company can afford that kind of pre-production and training time, so the employees take it on themselves to do what they can. Most of the self-improvement by the people I work with happens during lunch or after work hours. The artists in particular are really enthusiastic about making good use of their free time to learn new techniques, practice, and challenge each other. The designers do what we can with conferences, watercooler discussions of our different philosophies or things we've learned from games we've played recently, and the programmers seem to primarily learn from books and conference reports.