



Code Ownership is a Trap

Gregory Boyd



S. Gregory Boyd is a partner, Co-Chair of the Privacy & Data Security Group and co-Chair of the Interactive Entertainment Group at Frankfurt Kurnit. Mr. Boyd has extensive experience negotiating and drafting software (SaaS), technology, and game development agreements across all platforms. He also advises brands, media companies, and advertising agencies on a large variety of digital matters, including the creation, implementation, and management of privacy and data security programs. The International Association of Privacy Professionals (IAPP) has certified Mr. Boyd as both a Privacy Program Manager (CIPM) and Privacy Technologist (CIPT).

Contacts

Frankfurt Kurnit

New York,
United States

Tel: (212) 826-5581

Fax: (347) 438-2152

gboyd@fkks.com

| The Wise Ones Propagate the Myth

One of the third whiskey pieces of advice I hear “sophisticated” game developers giving young people at conferences is to make sure you “own your code” when you finally get a development deal. Ownership of the code is entirely unimportant and it will often be used as a weapon against developers to trade much more valuable rights. Think of it as simple misdirection, the modestly elevated progeny of a “your shoe is untied” type of distraction. When a publisher has you looking away, they might pick your pocket.

First off, we should admit that developers have less leverage in negotiations than funders. A funder can be an outside investor, traditional publisher, or newer publishing platform that may provide development funds. For the purposes of this short essay, I will refer to all of those as “publishers” and also for this short essay, the leverage difference is always in mind. I am not proposing developers ask for things they would never receive. Developers may be able to receive some or all of items discussed here or else it would not be worth writing about. Most importantly, for this article, developers should stop focusing on what is unimportant or simplistic and keep their eye on what really matters around ownership.

Second, a very brief note on how copyright functions. Copyright is the main protection for game code. One person can “own” the copyright in the code and other rights or subrights relating to ownership can be licensed out to other parties. In the law and in this article, the person that owns the rights and is licensing them out is referred to as “the Licensor” and the person receiving the rights is referred to as “the Licensee.”

| The True Nature of Ownership

As noted above, legally, ownership in code is controlled through copyright. In law school copyright is referred to as a bundle of sticks. How many sticks are in the bundle? There is an infinite number of sticks in the bundle.

This is one of the truly wonderful and meaningful areas of the law where mathematics, philosophy, and psychology land hard in reality to actually affect every game development deal. Just like in the real world, you do not have to be aware of the speed of sound, earth's gravitational constant, or behavioral economics. You can be entirely ignorant of those and they still affect your everyday life. The process works automatically and with a cold efficiency wholly indifferent on your awareness of the rules. The same is true for copyright in game development and its bundle of sticks. With focus and attention you can build something or get ignorantly bludgeoned to death with that bundle of sticks.

What are the sticks? The sticks are all the different ways ownership can be divided. In this little article we will just talk about three of the most important ones. After you know about these, you can start to imagine other sticks in the bundle – and what you can do with them.

I Ownership – Control – Cash

Copyright Ownership

When people think about ownership, the “trap” part is one of the sticks, but not the whole bundle. Plain vanilla copyright ownership in the code is important, but not by itself, and it is not the most important stick in the bundle. The development agreement should include who owns the copyright in the code itself. Preferably, this should be just one party to the agreement. Co-ownership gets messy because in the United States that means that each party can fully and separately exploit the code without accounting for revenue to the other party. Keep in mind, just because a party may not “own” the code, this means very little. For instance, you can be a mere licensee of the code and still (1) have a copy of the code; (2) re-use the code; or (3) make improvements or derivative works (sequels) using the code; or even (4) re-sell, publish, or sublicense the code. All of the things above are little “sticks” in the bundle of copyright that can be licensed out. So, of the three items discussed in this article, Ownership, Control, and Cash Flow – ownership alone is the least important. I would give the publisher copyright ownership of the code every time if the developer gets the correct license rights including the items above and some of the other elements of control and cash flow discussed below.

Control

One of the key sticks in the bundle of ownership is Control and even that has many elements. For the purposes of this article, we can consider three control elements: credit, translation, and sequels.

Who gets credit for making the game, creating the idea, and how is that credit displayed to the world? Many developers have learned the hard way that a publisher can have a larger, longer, and more prominent splash screen on a game opening. Furthermore, one of the people at the publisher could be anointed with a “Director” or “Creative Director” or even “Co-Creator” title. This has to be set out right from the start of the agreement, or the world will never know the truth about who made your game. Furthermore, titles are not empty appellations. If someone at the publisher becomes the Director of your game, you may lose power and final authority over creative decisions. While it might not be obvious at start how credit is related to control, the reality of the relationship is not something that should sneak up on a developer.

Consider translations and localizations of the game, who controls that, who selects the people to do it, who pays for it, and what visibility does the developer have into those costs? Localization is often ignored, but critical to a game’s success. Besides being expensive and complicated, these studios doing the work are your ambassadors to that part of the world. Their translations, feel for the game, and other subtle cultural changes are the difference between success and failure for a title. This is a key element of control and not a decision to make on cost alone.

Sequels are the most clearly felt pain for developers that fall into the simplistic trap of code ownership. One of fundamental rights of telling your own story is having control over how and if to continue that story. Many developers have thoughtlessly given that up. Where possible, make certain the developer has control over how sequels are told. The developer should have creative direction. The developer should have a first right of refusal on developing the sequel. If the developer wants a sequel, but chooses to not develop it themselves, for economic or other reasons, they should have approval over who continues the story. Perhaps they also maintain certain creative input or veto power over story elements. All of this can only be negotiated in advance in development deal. Without a prior explicit agreement, the publisher will usually have these rights by default.

Cash Flow

The flow of money is also more important than code ownership. Understanding cash flow requires understanding what goes into it and how exactly it is calculated. First, there is the royalty. This is often stated as 10% or 20% or [X]% of certain dollars to developer (or publisher). That is only the start of the story. If you get 20%, then ask 20% of what and how is it calculated? Is it gross dollars? Not likely. OK, then it is some form of net dollars. Now listen very carefully. Really. I'm not kidding. Maybe I'll put it right in the middle of a page by itself so you don't sleep through it like everything else.

THERE IS NO STANDARD DEFINITION OF NET REVENUE!

I hope you heard that. There may be a publisher's "way of doing things." Those are always designed to benefit the publisher. There may also be some things that the accounting department adds or subtracts that are difficult to change. None of that matters at the end of the day. Now is your chance to fix it. If there is something that goes on "automatically," then we will account for it here. How much is automatically deducted? 2%? OK then, the developer will just take 2% more of net.

If there is ever some discretionary wiggle room in a net revenue calculation, do not accept a publisher's argument that "it is not in our interest to take advantage of you here." When a developer sees that, then the appropriate response is to say "any deductions not explicitly in the definition of Net Revenue must be prior approved by developer." If it is reasonable, the developer will approve it. If not, then perhaps it should come out of the publisher's portion. Net revenue in the games industry is not a dumping ground of Hollywood Accounting where publishers should be encouraged or allowed to toss in every cost. We owe it to each other as **professionals in the space to anticipate costs and agree on what should be a deduction. There are so many cases where games have the opportunity to rebuild on the mistakes of traditional media. Net revenue is one of those places, but it takes a daily vigilance because the lazy, thoughtless way moves the ball toward traditional media's dumping ground of Hollywood Accounting (which truthfully is not good for either party).**

Keep in mind that if a publisher gets 80% and the developer gets 20%, then 80% is the fair payment for the publisher doing “their job” and any deduction from net revenue is tacitly an agreement that this is at least partly a “developer job.” Net revenue affects developers disproportionately. They are much more cash flow constrained than publishers. Though it is not mathematically or economically entirely accurate, it is instructive to consider that a developer is taking out a loan against future revenue to pay for any net revenue deduction. Whenever a developer gets stuck in a net revenue negotiation, the right question to ask is always – which party fairly has responsibility for this cost? What does the publisher’s 80% pay for? Maybe marketing should be included as a publisher internal cost, not a net revenue deduction. If the developer is paying for “everything” in the net revenue definition that means the 80% publisher split is meant to be internal publisher operating costs plus profit. Is that fair and does it mean the percentages should be moved to account for it?

Keep in mind, that if a developer cannot get the net revenue “wins,” then it should feel empowered to re-visit the split to account for any inequity. Conceding internal accounting process issues in the net revenue definition is fine, but do not leave the fundamental fairness issue unaddressed. Account for it by changing the royalty split.

Now is the time to ask questions and probe deeply. This is the honeymoon period. It is the nicest a publisher will ever be to a developer. Ask so many questions and get them to lay out so much of the important things on paper that they won’t have wiggle room later. This is the entire point of the contract process. Assume that there are large and important elements missing from the contract and make certain those are written out and explicitly addressed. Here it is important say that developers should not chase unicorns or invent specters either. Your attorney will help you understand what the differences are between real and imagined risks.

How publishers treat you now is a sign of what is to come. Are they totally honest now? Do they treat you like you are smart now? Are they forthcoming on internal processes? Do they seem internally organized and able to speak with one voice? Most publishers are risk-averse and disorganized entities that act a bit like super-organisms. Everyone is doing the best they can with the information they have. Everyone that talks to a developer will overstate their authority a bit.

It isn't malicious. It is human nature. Still, we have to get it in writing or it won't matter. Ask people questions. Ask them questions repeatedly. Get them to commit by putting it into the contract. For instance, if they assure you that you will be involved and have final approval over advertising spend, then put it in the contract. If it is not in the contract, it is not real.

Put in a sample calculation. If we are all friendly and the omniscient publisher does this all the time then we won't have any problem just putting it all right down here. There is no real reason a publisher cannot put in a sample royalty calculation based on real numbers with real deductions and talk through various scenarios with the developer. Insist on it.

Consider this sample calculation. Net Revenue equals Gross dollars minus what? Maybe platform fees, license fees, discretionary costs not listed? These definitions can go on for 5 or six lines of text in a real contract. Know what every one of them means and insist on seeing their effect in a sample calculation.

What about advertising? What about internal costs? What about other "soft costs" or costs at their discretion?

Keep in mind that Net Revenue means one thing in practice. What is the developer paying for? It is the developer's account. The publisher is loaning you money against future revenue. Still, if advertising or marketing is deducted that means that the developer is paying for advertising or at least sharing in those costs. Do you have control over the creative, over the spend amount, over the placement? If you don't, is it fair for you to pay for it?

Invoicing and quarterly reports are another place developers can lose money. Today, the standard is real time same account access to numbers. In the past, publishers were able to generate reports on sales and give developers the reports. That made sense because we were dealing with a lot of retail partners worldwide. The standards were set in a pre-internet, pre-social media era. Technology was not as advanced as it is now. Today, we have multiple tracked login access and real-time reporting on everything that matters. That is the standard and put in the agreement that this is how the developer will receive reporting and how the developer will be paid.

Auditing is important. Even if you never use it, you want to have the right to use it. Make sure you list exactly what you can audit. If you have the right to audit something you often don't have to audit it. In a low level dispute, you can write an email and say, our agreement says we could just audit X. So, why don't you give it to us? Then, people are much more forthcoming with providing X. So, think about everything you might want access to in the agreement, especially if you are not able to get real-time account access to all revenue streams.

Consider who gets paid first. This was not a choice at the start of the games industry or even in the first 10 years of this century. Players used to pay retailers and publishers first every time. Now, a developer can insist on having the primary platform accounts for Steam, Apple, Android, etc. in their full control. Clearly with the larger platforms like Xbox and PlayStation the publishers and the platforms will get paid first. Remember that the large publishers are not the only way to do it. The best possible way to calculate royalties and net revenue is to calculate it yourself. There are few things in life more satisfying than writing out a check to your publisher in this context. There are many developers that send publishers money every quarter now – but only if they were thoughtful enough to ask for it in their development agreement.

| Now You Know and They Cannot Fool

In most development agreements publishers allow developers to “own” their code. The developers are very pleased with themselves and go to sleep each night feeling they won their negotiation. In truth, they often give away all or substantially all of everything discussed here including crediting, sequel control, and worst of all real monetary participation. When I was a much younger attorney, this was described to me by venerated “old timer” publisher attorneys this way. “Sure, Developer, you can pee anywhere you want. I just have to hold it for you.” This is terrible on so many levels. It is vulgar, oppressive, and assumes the developer is a man. The statement itself is not nearly as bad as the reality and the effects of that reality on the unsuspecting developer. Maybe we have a chance to do a little better in the coming years.

A final final word, that upon reflection will go on a bit longer than it should. You should always seek the advice of your own attorney. Every situation is different. I wrote this to help you,

knowing full well that one day you (developer) will likely use it against me when I am on the publisher side. That does not mean you will win. Whitman and I have anticipated this:

Do I contradict myself? Very well then, I contradict myself. I am large and contain multitudes. I wish the intractable hell on all of you – of having the publisher and developer literally yelling at you on a conference call, quoting things you have written as evidence for their position. Then, in the same moment, you will truly live and wish you hadn't. Further, I assure you all of your good deeds will be properly punished.

Knowing the rules of the game elevates the discussion and this is my intent. It is immoral and largely no fun to beat up on the ignorant. Economic realities and the myriad contextual elements of the deal will dictate who prevails on any individual point. It is my hope that writing something like this, teaching classes, and gods-save-me-from-ever-doing-it-again WHOLE BOOKS on the subject will let all the parties go into these agreements with their eyes open. It is one thing to “fool” an ignorant developer. I would rather not have that as a standard in the industry going forward. I would rather have it that each side knows we are talking about an infinite bundle of sticks, considers and accepts positions deliberately, with an understanding of the consequences.

It is about the art at the end of the day. We have to feed ourselves and our children, but we cannot forget that it is (or should be) about the art. No matter where we sit at the table, we all choose to be here. I hope there was more to that choice than just making money because there are easier ways to do it. Be honest with each other. Commit to doing what we say we are going to do. Show that commitment by writing it out in the contract. It isn't “the paperwork,” it is the very foundation of THE WORK. Literally every choice in the contract comes down to building the game. Take it seriously. For each choice, is this going to make the game better? Is this better than not having the game made or having it made on a shoestring budget? Are these the people you want to work with? Is this going to be more of your or less of vision? Is it something you can feel good about, including how you treated the people involved? Is this a day well spent or a compromise?

March 2017