



IGDA Contract Walk-Through

- First Edition -

April 22, 2003

www.igda.org/biz

Table of Contents

Introduction	3
Contract Essays	4
Cross-Collateralization	4
Definition of "Net Sales"	7
Derivative Works	12
Game Documentation and Trade Show Demos	14
Localizations	18
Ownership of IP (Game Code – Developer vs. Publisher vs. Licensor)	23
Ownership of IP (Tools and Technology)	29
The Reserve	33
Lawyers Panel Bios	37
About the IGDA	39

The IGDA Business Committee

The International Game Developers Association is the independent, non-profit association established by game developers to foster the creation of a worldwide game development community. The IGDA's mission is to build a community of game developers that leverages the expertise of our members for the betterment of the industry and the development of the art form.

The IGDA Business Committee's mandate is to empower the development community with business knowledge and in the process allow developers to make better games.

The goals of the Business Committee are as follows:

- Enable developers to build stronger, more successful companies
- Provide knowledge and business support resources
- Increase the perception of game development as a credible business and raise the profile of game developers as viable companies
- Improve the publisher/developer relationship
- Improve the retailer/developer relationship

Additional information on the IGDA and the IGDA Business Committee can be found at

<http://www.igda.org/biz/>

<http://www.igda.org/committees/business.php>

http://www.igda.org/committees/business_members.php

Introduction

Welcome to the IGDA Contract Walk-Through! The second* most difficult thing any developer has to contend with is contract negotiation.

Game development is a creative and technical challenge. Making the deals is a legal and business necessity. Few developers in the earlier stages of their careers have the training, knowledge and experience to navigate complex contracting issues.

Contracts are long and getting longer. As the business grows, development budgets and staffing begin to resemble those for feature motion pictures, publisher risk attached to each project grows, and contracts get longer and more complicated.

In late 2002, the Business Committee of the IGDA suggested a project that could help developers better understand game development contracts. The Committee envisioned a living resource that could grow over time and serve as a primary reference tool for developers.

The IGDA Contract Walk-Through breaks down the development contract into a collection of issues, all of which must be addressed and negotiated. Each issue is described, analyzed, alternatives suggested, and sample language provided. There is enough detail to cover the issue, but it is presented briefly and cogently for a *non-lawyer* audience.

Each article is written by a volunteer attorney; all experienced practitioners who regularly represent game developers in contract negotiations. (See page 37 for bios.)

The Contract Walk-Through is an ongoing effort of the Business Committee, and the attorneys' panel. It is anticipated that new sets of essays will be added twice each year. After an essay has been published, it will be annotated by members of the IGDA Business Committee and by other well known developers, who will add their own comments based on their experiences dealing with publishers and the issue in question.

Over time, the Contract Walk-Through will grow to be a comprehensive analysis of development contract issues, a place for developers to go to better understand this complex part of the business.

In reading the Contract Walk-Through, it should be remembered that this is presented for general informational and educational purposes and is not to be taken as legal advice. Every situation is unique and developers are always best served by working alongside their own experienced legal counsel.

If you have issues you would like us to consider, or other comments on the project, send them by email to biz@igda.org, subject "Contract Walk-Through."

Jim Charne
Law Offices James I Charne
Santa Monica, CA

* The FIRST most difficult thing any developer has to contend with is, of course, the challenges of game development!

Cross-Collateralization

by Jim Charne

Introduction

“Cross-collateralization” may be the longest word in a development contract. It means that advances paid in connection with one game can be recouped (recovered by the publisher) from royalties earned in connection with sales of other versions of that game or other games.

Sample Clause

Here is a typical cross-collateralization clause from a development contract:

“Advances paid to or on behalf of Developer may be recouped by Publisher from any royalties payable hereunder, or payable in connection with any other agreements between Developer and Publisher.”

Discussion

Cross-collateralization as a concept can be applied very broadly, or very narrowly (or not at all). When a developer contracts for multiple SKUs of the same game under a single development contract, publishers will commonly seek to cross-collateralize all SKU’s for royalty purposes. This is the narrow application.

Publishers may argue that in simultaneous multi-SKU development, resources are shared and it is not possible to accurately allocate among the SKUs for recoupment purposes. Publishers may also claim they take a greater risk in financing simultaneous development of the game on multiple systems. Rather than attempt to find a workable allocation, the publisher reduces its financial risk by recovering all its development costs from one big royalty pot.

Ray Muzyka

Personally, I recommend avoiding cross-collateralization in any form if at all possible. It may not be possible for a newer developer to avoid it, however.

Ron Moravek

Its important to note that in most cases this is not an area of the contract that is written clearly and explained well. More than likely, it is assumed and hidden in lawyer “mumbo jumbo”. It is best to have this written clearly and accounted for in the royalties area. If you do, “good work” for the publisher, most likely they will renegotiate with you to make each one product specific.

Binu Philip

As horrible as that sounds, it is indeed true. I’ve seen this in some contracts. However, it seems to be an easy clause to get rid of. Cross collateralizing multiple titles effectively pushes back potential royalties for a very long time. Each game should have its own risk profile. Cross collateralizing multiple games means the developer might actually be paying for the second game if the first one earns out. Cross collateralizing multiple SKUs of the same game however, is fair. For example, it is fair that Spiderman the Movie PS2, GCN & Xbox are all cross collateralized. However, it is not fair for Spiderman the Movie 1 and Spiderman the Movie 2 to be cross collateralized.

Cross-collateralization – cont'd

At its broadest, publishers may attempt to cross-collateralize projects developed under separate development contracts. In the example language above (adapted from a real contract), all royalty streams could be retained by the publisher to recoup advances paid in connection with this contract.

Cross-collateralization reduces publisher risk. In the narrow multi-SKU application above, the publisher may decide to support a hardware platform that is early in its life-cycle, as well as more established platforms. If the riskier platform never achieves critical mass (for example, the Sega Dreamcast), the publisher can still recover development costs by applying royalties earned from the best selling SKU against advances paid for the failed system version. At its broadest, the publisher can tap multiple royalty streams from multiple projects to recover development costs, even on products that may never come to market, or achieve only limited distribution!

Cross-collateralization pushes the time when a developer could see royalties farther into the future, and reduces the possibility that any royalties will be earned. When royalties from the better selling SKU are used to recoup lesser-selling SKUs, money that might otherwise go to the developer is retained by the publisher.

Cross-collateralization as a concept comes from the music industry. Historically, the advance and recoupment model appears in 99+% of recorded music artist contracts. Recording artists sign long term, multi-album contracts which are entirely cross-collateralized. Advances paid for recording of any cd, production of any video, or for other recoupable expenses can be recouped from royalties payable from sales of any cd under the deal. However, the music and games industry are fundamentally different. Game developers typically do not sign long term exclusive agreements, game publishers do not invest heavily in career “artist development” of game developers, and developers do not enjoy ancillary and potentially enormous income streams such as personal appearances and music publishing.

For this reason, cross-collateralization in a developer agreement should be resisted. While the concept is commonly found in standard publisher agreements, it may be possible to narrow its application, or remove it from the final agreement altogether.

Josh Resnick

We have never agreed to this type of cross collateralization in our contracts. Fortunately, we have been able to successfully make the argument with publishers that it is not reasonable to cross-collateralize completely separate product lines. We have, however, allowed publishers to cross-collateralize between SKUs of the same Product.

Joe Minton

I don't have a problem with signing cross-collateralization when dealing with multiple SKUs of the same game that are in development at the same time. In all other instances, and for 10 years, we have successfully stricken cross-collateralization from our contracts. The importance of not crossing separate games cannot be over emphasized.

Ray Muzyka

However, if you can get large advances for all of your projects (building in profit to your advances), you can essentially make the issue of cross-collateralization moot (since you wouldn't care as much about royalties in that case), assuming of course that your royalty advances are recoupable but not repayable.

Cross-collateralization – cont'd

Any final agreement is a negotiation, a compromise between the parties, and the success a developer will have in approaching cross-collateralization will depend on how the publisher views the risk, potential for success of the game or games, reputation of the developer, and publisher financial model.

Similarly, in approaching a negotiation, the developer should weigh the relative importance of this issue, look to its relationship with the publisher, and determine whether cross-collateralization is a priority issue in an agreement.

If the decision is made not to contest this clause, or if the developer can narrow, but not remove it altogether, it may be possible to revisit the issue at a later time; for example, when negotiating a contract for further work. The developer may find it enjoys a greater bargaining position when the publisher comes back for more work.

© 2003 Jim Charne. All rights reserved.

Martin de Ronde

I would argue that in the broad sense, it should definitely be resisted. In the narrow sense however, we as a company would be able to live with it, as long as there are clear indications in the contract when royalty calculation is carried out (quarterly, annually) and after which time royalties have to be paid out, irrespective of any cross-collateralization clause. For instance, if a publisher indefinitely postpones the release of the GC version (with PS2 version and Xbox version already released), a developer wouldn't be entitled royalties as the cross collateralization calculation can't be carried out.

Ray Muzyka

Publishers may resist removing the cross-collateralization clause at a later date, especially if the new work being proposed would be cross-collateralized against the earlier work and if the earlier work has yet to generate profits (often the case when a developer signs up the next project).

Definition of "Net Sales"

By Don Karl

Introduction

A developer typically finances the development of a game through development fees and advances but the "profits" that the developer hopes to earn come in the form of royalties. The definition of "net sales" is an important provision of the development agreement because royalties or contingent compensation are calculated as a percentage of "net sales" (or "net receipts", "net revenues" or other similar term).

Sample Clause

"Net Sales" is almost always a defined term, usually in either the definition or royalties section of the development agreement. The definition provides for the calculation of net sales by reducing the total revenues received by a publisher from sales of a game by various costs and deductions that the developer and publisher agree are appropriate. For example:

"Net Sales" means all amounts received in connection with the sale, lease or license of units of the Game, less the following amounts: (1) taxes on sale, lease or license, such as sales, use, excise, value-added and other taxes; (2) costs of insurance, packing, custom duties, shipping and similar charges, if not reimbursed by customers; (3) costs of manufacturing and packaging not to exceed \$ ___ per unit; (4) all royalty and/or license fees payable to proprietary system licensors and/or third party licensors for the right to publish and manufacture units of the Game; and (5) amounts for credits, returns, refunds, price protection or promotional allowances. The returns reserve shall be calculated as follows: For each quarterly payment to be made by Publisher to Developer, Publisher shall withhold twenty percent (20%) of each payment as a reserve for returns of units of the Game.

Binu Philip

Net sales is a very important concept to understand because it will have a great impact on royalties earned. It will be very frustrating to create a hit game and not get paid the royalties you deserve because you didn't really understand net sales and the various deductions. Some large publishers include deductions that others don't. That is why you can't compare a royalty rate from one publisher to the next without understanding net sales and the way each publisher calculates deductions.

Ron Moravek

An important distinction in Net Sales is the difference between "actual costs" and "costs." Publishers can assign "costs" to a project but not actually spend the money. I.e. they allocate 4% of revenue to marketing but the money never actually gets spent. I have seen this several times in contracts where a product has been on the shelf for 5 years and the publisher is still taking 4% for marketing costs.

Ray Muzyka

One approach with the definition of net sales that can work well is to simplify it as much as possible (i.e. reduce the number of possible deductions, and decrease the royalty rate proportionately). Having fewer deductions usually benefits the developer in that there are fewer 'outs' available to publishers later on.

Definition of “Net Sales” – cont’d

Publisher shall retain such returns reserve for such quarter's sales as a reasonable reserve against charges, credits, or returns. Publisher shall pay to Developer on a quarterly basis the reserve which was withheld two (2) quarters earlier. The amount to be paid shall be the full amount less any amounts actually reimbursed or given as credit to Publisher's customers.

Discussion

The negotiation of the definition of net sales centers on two primary areas. First, are royalties calculated on an amounts received basis (i.e., a cash basis) or an accrual basis (i.e., a royalty is calculated when the publisher ships or invoices a shipment and then books an account receivable). Second, which items should be deducted from gross revenues to arrive at the net amount on which royalties are calculated. With respect to cash vs. accrual, the industry standard has become that publishers almost always use a cash basis for calculating "Net Sales" but, depending upon the relative bargaining power of the developer, it is sometimes worth discussing. With respect to deductions, there are a number of categories of deductions – each with varying degrees of justification – and some more easily quantified than others.

Taxes and Shipping Costs. The first category of items is deductions for direct costs of sales. These include sales, use and other similar taxes on sales which are usually paid at the point of sale and are never received by the publisher, as well as payments for insurance, packing, customs duty, shipping and similar charges that a publisher pays directly or reimburses to its customers. These are appropriate deductions from gross revenues because the publisher may never receive, or if it does has to promptly reimburse, the amounts attributable to these expenses.

Credits, returns, refunds and price protections. The deduction of amounts that the publisher credits or refunds to distributors and customers as a result of returns or as promotional allowances is also appropriate. Royalties should not be paid on sales that are subsequently cancelled, as is the case with returns.

Joe Minton

I find that some developers focus on the royalty percentage far more than the definition of net sales – when the percentage number is totally irrelevant until the net sales wording is nailed down. For example, in one of our deals, the publisher did not deduct anything except for return / price protection. They paid a lower percentage number, but it equaled a higher number under different definitions (and allowed zero room for monkey business).

Definition of “Net Sales” – cont’d

However, many publishers will not be satisfied with a simple statement that the calculation of net sales requires the deduction of the amount of refunds and credits because, unlike taxes and shipping costs, the level and amount of returns is not immediately known to the publisher. Since most of a game's sales typically occur in the first few quarters after release, a publisher does not want to be in the position of having paid royalties on initial high sales only to later discover that it has overpaid due to high returns. As sales decline over time, the publisher may not be able to fully offset the returns against declining future royalties because the royalties are less than the returns. In that case, the publisher would have to obtain a refund of the royalty overpayment from the developer.

To avoid this situation, it is common practice to create a reserve for returns equal to a percentage of each royalty payment, calculated on a basis consistent with the accounting for royalties (usually quarterly). For each quarter during which royalties are to be paid, a percentage of those royalties will be retained by the publisher for a period of time that should allow the publisher to determine whether the net sales on which it is paying royalties will be significantly reduced by refunds. The negotiation of return reserves focuses on the percentage to be retained and the period of time for which the reserve is held before the amount withheld, after taking account of returns, is "liquidated" (paid to the developer). Typical percentages range from 15% to 25% range though publishers may attempt to get a higher amount, particularly for the first six months of a game's release. The period of time for which the reserve is held is usually six months.

Note that sometimes the reserve for returns is not in the definition of net sales but separately appears in the royalty payment sections.

Price protection occurs when a publisher grants a retailer (through a credit memo) the right to reduce the wholesale price for a game, which then allows the retailer to mark down the retail price of the game on its shelves. As the game moves through its sales cycle or, if it is not immediately successful, the publisher receives pressure from retailers for price protection. The publisher and developer heavily negotiate this issue because it directly affects the publisher's profit margins and the developer's royalties. Many publishers require the developer to "share in the pain" by allowing the publisher to deduct the amount of price protections granted to retailers from gross sales in the calculation of "Net Sales."

Ray Muzyka

The period could be as long as a year, but try to get the shortest possible time for returning the reserve to reduce the net present value of the reserve that is held. An alternative is to get reasonable rates of interest on the reserve for the time it is held.

Definition of “Net Sales” – cont’d

Limiting and seeking to cap price protections is a good strategy for a developer when the publisher insists on being able to deduct the amount of price protections.

Manufacturing Costs. Publishers almost always demand a deduction for the costs of manufacturing the game units, including all materials, replication and packaging. The inclusion of manufacturing costs essentially makes the net sales number a measure of "gross profit" in the accounting sense. If manufacturing costs are included, a developer should attempt to negotiate a per unit maximum deduction for these costs so that there are no surprises. This is achievable because a publisher has all of the necessary information to accurately estimate these costs.

Sometimes a publisher will ask to exclude revenues from sales below manufactured cost. The publisher will argue that for sales of units well into the life cycle of a game at a price below such cost, it should not have to pay a royalty if it is losing money on such sales on a per unit basis (which ignores the fact that the publisher may already have earned substantial revenues to cover all of its manufacturing costs). This issue is usually addressed by excluding from net sales all revenues from sales at prices below a certain percentage of wholesale price. This should not be agreed upon in a definition of net sales in which manufacturing costs are also deducted (the publisher would be double-dipping!).

Third party license fees. Publishers seek to pass through to developers all license fees payable to proprietary system licensors (such as Sony) or to third parties who have granted a license in connection with the particular game. Platform license fees are appropriate deductions as are most license fees paid for technology or content though not if the publisher is also the owner of the content or technology (such as the game development divisions of motion picture companies). Developers should be alert to situations where "license fees" are "paid" by the publisher to an affiliate or related company.

Marketing costs. A very important category is marketing and advertising costs. The deduction of these costs essentially reduces the net sales number to a net profit number since almost all related costs, other than overhead, will have been deducted to arrive at net sales. Marketing costs are hard to define and limit since many expenses can be characterized as advertising or marketing costs. As a result, the deduction of marketing costs can effectively eliminate a developer's opportunity to earn royalties.

Joe Minton

This is also a good argument for getting the publisher to not deduct the COGs (cost of goods). Since they know the cost, they can just factor it in when determining the royalty percentage.

Ray Muzyka

“Double-dipping “ can be avoided with a minimum defined royalty per unit of \$0.

Ray Muzyka

Note that sublicensing in distribution could result in a reduced royalty rate payable to the developer so developers should be on the lookout for this as well.

Matias Myllyrinne

Such “license fees” should be capped (if legitimate expenses exist for e.g. distribution in different territories) or excluded all together.

Martin de Ronde

Many publishers are multinationals and most of the clauses speak of these “affiliates” and “license fees paid to affiliates”. Since this provides the opportunity for publishers to hide ‘net receipts’ within the group, it is something developers should be wary of.

Definition of “Net Sales” – cont’d

A developer should not agree to a deduction for marketing costs. If a publisher insists, a developer can respond by seeking to: 1) limit the number of deductible items; 2) exclude one or both of two “big-ticket” items - coop advertising and marketing discount funds (“mdf”); and 3) agree on a per unit maximum deduction. In addition, the royalty rate should be reviewed, and adjusted if necessary, to ensure that it is the “going rate” for a deal in which net sales is reduced by marketing costs.

Conclusion

The negotiation of “net sales” and similar terms is critical to determining whether and how much royalties will be paid if a game performs well. A developer should make sure that it understands each deduction made to arrive at net sales and that each of those deductions make sense for his or her game project.

© 2003 Don Karl. All rights reserved.

Ray Muzyka

Refusing to allow marketing costs to be deducted might have the unfortunate side effect of decreasing the marketing spend. I would push for a higher royalty rate and allow certain marketing costs to be deducted.

Joe Minton

I know definitively that this clause is used by at least one major publisher to swing royalty payments away from publishers that they don’t like or aren’t still working with.

Martin de Ronde

Note that apart from the above, quite clear deductions, publishers also tend to include vague things such as ‘agent fees’ or ‘indirect production costs’. Developers need to be careful...

Matias Myllyrinne

Bundling... in some cases this is legitimate but in others it can be used to cheat the developer. E.g. if the publisher is selling on the developer's game and one of its own (where it is not paying royalty) they will be tempted to do the following: the fair market value of games is e.g. \$25 wholesale for both. Publisher will make a deal with wholesale where by they pay \$35 for the publisher’s game and \$15 for the developer’s game. Any kind of piggy-backing (with hardware or other games) needs to be approved by the developer. If this is not possible the developer must insist that the allocation of revenue sales price) for its title is representative of prevailing market prices.

Derivative Works

By Trevor Fencott

Introduction

“Derivative Work” refers to a product that is based, even partially, on a previous game. Typically, this means it is based on the story-line and characters of the previous game. Varying from country to country, the term may have an underlying statutory definition that supplements any contractual definition. However, it is often defined in the development agreement to include sequels, prequels, localizations and ports to different platforms. More recently, the definition may explicitly include rights such as ancillary or merchandising rights (e.g. movies, television programs, toys, collectible cards, strategy guides, etc.).

Sample Clause

The following is a definition of derivative works from an actual development contract:

“[Derivative Works] means works within the definition of “derivative works” as defined in Section 101 of the U.S. Copyright Law (17 U.S.C. Section 101) which are based, in whole or in part, upon the Work(s) or a prior version of the Work(s). The term “Derivative Works” shall include, without limitation, prequels, Sequels, Localizations and Ports.”

Discussion

The concept of a derivative work is relevant to developers in the following two ways: (i) determining future royalties (i.e. who gets paid for what) and (ii) determining future work (i.e. what rights and obligations the developer may have to develop any derivative works).

Generally speaking, the holder of the copyright in the original game has the exclusive right to develop derivative works based on that game (or collect royalties on such derivative works if they were created by others).

Scott Miller

Developers should not give up too many IP rights without a fight, assuming the design of the IP is yours. The developers with the best financial strength in this industry got that way by creating and maintaining ownership of their IP. Never forget that.

Matias Myllyrinne

Despite the importance of this topic, developers should not be led to believe that non-game derivative works are financially significant. Most aren't. Movie rights maybe, strategy guides etc. - no. However, if the IP is with the developer he should always have a veto right on any derivative works, to protect the consistency and integrity of the brand.

Ron Moravek

This area is really only important if the developer owns the IP. If the developer does own the IP its important that the developer maintain the right to refuse development of any other products using the IP. If the developer does not have this, there is no benefit to owning the IP. Having said that I think its fair that the publisher (if this is the first product of a new IP and the publisher is paying for development) benefit if the developer goes ahead and makes a movie, toys, or other products. I have seen royalty rates to publishers range from 30-50% of the Developers net in these cases. Obviously, interactive electronic products are different as the publisher is going to want to maintain publishing rights.

Derivative Works – cont'd

For this reason, publishers typically attempt to control all intellectual property rights to a game (including the right to create derivative works). A publisher may argue that this comprehensive intellectual property ownership is necessary in order to reduce their financial risk. Since the game can be exploited in numerous ways (sequels, movies, merchandise, etc.) the numerous royalty streams increase the chances that the publisher will make a profit on the game.

The grant of the right to create derivative works should be carefully considered by the developer. The right to create derivative works is valuable (consider the number of sequels, strategy guides and movies currently on the market). A developer may be able to negotiate a passive royalty on derivative works, such as ports or sequels, subsequently created by the publisher (if the game is an original work by the developer) or trade the grant of the right to create derivative works for more favorable terms in other areas of the development agreement. For non-game related derivative works (e.g. action figures or film adaptations), a developer may be able to negotiate a higher royalty rate from the publisher since such products require little or no expenditures on the publisher's part.

Aside from the issue of what constitutes a derivative work, another important issue for a developer is who actually gets to do the work of developing any derivative works. It is in the publisher's interest to resist giving the developer the exclusive right to create derivative works of the game (i.e. sequels, prequels, ports, etc.). For example, in the future, the publisher may be able to find a lower cost studio, or take projects in-house for development.

A developer should use the current development agreement to secure as much of the work associated with the development of any future derivative works of the game as possible. There are a number of contractual provisions that can assist the developer in keeping their development pipeline full.

It may be possible to qualify the word "exclusive" based on the products that fall within the definition of a derivative work. The publisher may be willing to consider granting the developer the exclusive right to do any port work, for example, while leaving the issue of sequel development to be determined at a later date.

Binu Philip

Most movies that are made from game IP's don't pay anything significant for the use of the IP. The IP owners typically do it in hopes of increasing the profile of the IP which should improve the game sales.

Binu Philip

It is important to get a good passive royalty on ports and sequels that use your work. This partially ties into protecting your IP. If you own your technology and tools, the publisher will have to pay you a passive royalty to use that code if they are doing a game with it. Passives can be very lucrative.

Joe Minton

We've been able to get a passive royalty quite easily when we do not own the rights ourselves.

Ray Muzyka

Try to get the rights secured to develop spin offs without the obligations to do so; this can be a double-edged sword otherwise.

Derivative Works – cont'd

Other contractual provisions the developer may wish to consider include: (i) rights of first negotiation, (ii) rights of first refusal, (iii) rights of last offer and (iv) so-called “lock-up” rights.

A right of first negotiation means that the publisher and developer agree that should the publisher decide to create a derivative work, the publisher has an obligation to approach the developer first and negotiate a potential development agreement for the work. The time frame for negotiation is typically anywhere from 20 to 30 days, after which time, if no agreement has been reached, the publisher is free to shop the development work to other studios. This is a relatively common provision.

A right of first refusal means that if the publisher decides to create a derivative work, they are obliged to offer the work to the developer first. Similarly, a right of last offer means that the publisher has an obligation to return to the developer and give them a last chance to match any existing offer. In either case, if the developer turns down the work then the publisher is free to shop the project to other studios. Since the publisher typically sets the commercial terms of the agreement (which may not be reasonable to the developer) these types of provisions do not offer the developer substantial protection.

A “lock up” typically means that if the game hits a certain sales threshold, the publisher will grant the developer the exclusive right to develop certain derivative works (e.g. a sequel). If the threshold was not met, the publisher will be free to shop any future work to other studios. This is often a very useful compromise position. However, defining a threshold that is satisfactory to both parties may be challenging.

Conclusion

As with all agreements, negotiation is largely determined by the relative bargaining power of the parties. The developer should consider the relative value of derivative works and future work on such derivative works in the context of the development agreement as a whole, and assess whether the issue is a priority.

© 2003 Trevor Fencott. All rights reserved.

Ray Muzyka

These clauses are really only fully effective if the royalty rate floors for the ROFR, ROFN, or ROLO are defined in advance. In other words, a ROFR with no royalty rate could have limited utility as it doesn't define the offer that needs to be made by the publisher to the developer.

Scott Miller

If the publisher is recouping advances, then *you* are paying for the project's development cost, and not the publisher. With the majority of projects, even those that do not sell as anticipated, the publisher makes enough to at least break even, *even* if they do not fully recoup advances. So, when they play the card that says they're taking all the risk, know that it's a joker.

If your studio is paying for the project's development (because the publisher is recouping advances), then you have a compelling case to retain IP ownership. If the publisher is stuck on owning the IP, then suggest that they pay for the development of the game, too. If they say they are, then kindly point out that if that's the case why are they recouping advances. (Watch them squirm!) It's only fair that whoever pays for development, owns the product. It's hard for them to dispute this if presented to them.

Again, if your studio is paying for development, you have every right to maintain IP ownership. The publisher will push hard to get it, but instead, offer them any of these other things:

- The right to the first sequel, at a 20% improved royalty deal.
- Matching rights to a second sequel.
- Merchandising rights to each game they publish.
- Finally, be willing to take a little less in royalties if you can, in order to keep your IP rights. If you end up creating a valuable game brand, then it's worth almost any sacrifice in the short run.

Game Documentation and Trade Show Demos

by Thomas H. Buscaglia, Esquire

Introduction

Game Documentation and Trade Show Demos are often not given serious consideration in, or omitted from, developer contracts. Though their impact is often overlooked, the failure to consider these elements when negotiating a developer/publisher contract can result in alienated end users and delay in the timely delivery of milestones and advances throughout the term of the development agreement.

Sample Clause

Game Documentation and Trade Show Demos, if present in the contract, are usually included in the Deliverables portion of the contract and may also have specific references similar to the following:

Game Documentation

“The Developer will provide content for the end-user documentation including an instructional manual, quick reference card, and any other documentation related to the gameplay and back-story.”

Trade Show Demos

“The Developer will provide Demo versions of the game for trade shows such as E3, ECTS, SCoRE sufficient to demonstrate the then current playability and functionality of the game in a positive manner. Developer will also, at (Publisher’s or Developer’s) expense, provide sufficient personnel in attendance to operate and maintain the Demo and to participate in media contact for interviews and other promotional events.”

Tom Crago

The whole concept of the trade show demo, or indeed of any form of marketing material, can be unpalatable to developers. We tend to think solely in terms of completing our game, often preferring to ignore the necessity for creating these materials. They are, of course, a critical ingredient in a successful game, and developers who have recognized this fact have enjoyed increased sales and exposure as a result. So the first step, even at the time of entering into a development contract, is to realize that you will have to spend many, many hours creating demos and marketing materials to support your Publisher and to assist them sell your game. You need to plan for this from a resource allocation perspective, and of course you need to budget for it. It will take time and it will take money, and for that reason it needs careful treatment in the contract.

Matias Myllyrinne

Trade show demos that are done by the developer, give the developer ammunition for the future. This is overlooked in the section. By building relationships with buyers and most importantly media, the developers create a valuable network and loyalties that can be leveraged. Being able to get your own message across without relying on the publisher’s PR and marketing machine allows for a soap box in conflict and changing publishers without losing press relations.

Ron Moravek

All marketing and demo material requirements should be built into the schedule and agreed upon by both parties prior to that milestone beginning. If the deliverables change as a result of the publisher wanting something different or in addition then, they have to pay for it. The best way to account for this is through a “Change order Request”. Basically, the publisher sends a written request, the developer gives a quote on additional costs, if any, then the publisher agrees or disagrees and the game moves forward.

Game Documentation and Trade Show Demos – cont'd

Discussion

Game Documentation and Trade Show Demos are deliverables that are well outside the core business of the developer, making their game. But these elements are essential to the successful sale of the game. Contracts that overlook these elements can result in significant problems when the need for these deliverables arises, especially for small teams on a tight budget.

Game Documentation

Game Documentation is the written material included with the game media. The background story for the game and world it occurs in, as well as the “how to play” user interface instructions, are included here. Obviously, the end users need to have the user interface and play elements explained. Who the characters in the game are and what the game story is about are also often essential to understanding the gameplay. So too is a narrative background for the world the game occurs in. Often much of this material is contained in the Project document (“Game Bible”) delivered to the Publisher at the commencement of the relationship. The problem here is that often the Game Bible descriptions do not provide sufficient information to give the end user what they have come to expect - a compelling story about the game and its world.

Developers are usually adequate at explaining how to play their games. However, development teams may not include gifted copywriters able to compose a well-written description of the game world, let alone provide a compelling narrative. Also, too often these matters are left until the last phase of the development cycle. When crunch time hits, the last thing the developer needs is to take a member of the development team off of debugging or the completion of that last game element in order to draft the manual and other game documents. Cut and paste from the design document and a superficial back-story often results. Unfortunately, poor work here can seriously effect the end users initial impression of the game. And often, that first impression of the game may make or break the games ability to succeed in a crowded competitive marketplace.

Tom Crago

How many times have we been left disappointed by the marketing and support that our games have received? If we want to address this, the best way forward is to be proactive.

Ray Muzyka

Making game documentation and trade show demos should probably be considered at least a part of the developer's core business. Often times, publisher marketing staff are busy with multiple projects and they may not have an in-depth understanding of all projects they are working on. It could be difficult for them to optimally write or coordinate the writing of game documentation or do optimal trade show demoing without the assistance of the developer.

Josh Resnick

We usually never sign up to write the manual since invariably we run out of time. Typically, we'll provide the publisher with play instructions and guidance, but let them hire and manage a marketing professional or copy writer to add the polish and sizzle.

Game Documentation and Trade Show Demos – cont'd

Make certain that as a developer you understand the scope of your responsibilities concerning game documentation and pay the same through attention to it that you do to all the other aspects of your game. If necessary, out-source the documentation to a professional copywriter. Just be sure that the documentation provided with your game enhances, rather than hinders, your game's chance of success.

Trade Show Demos

Trade Show Demos are required to preview your game to the press and retailers. The press helps create the necessary "buzz" to drive initial sales. Previews by retailers establish the initial orders for your game. So, these demos are as important to the developer as they are to the publisher. The Trade Shows also provide an opportunity for the developer's spokespeople to provide interviews with on line and printed media to help promote the game prior to its release.

Trade Show Demos are not the demos that are included on Game magazine CDs. They are often not even fully functional. In early stages a Demo may be comprised only of a few key elements presented at the show by a development team member. At later stages of Development the Show Demos may be a fully functional "hands on" demo that can be played by those previewing the game. In either case, these Show Demos present a huge opportunity to promote the game. But they also require a substantial diversion from the straightforward completion of the game. It is the diversion from the game development that must be taken into account when determining the impact these Show Demos will have on the schedule of milestone deliverables and the cost of development.

It is not uncommon for a publisher to expect the developer to provide several key development team members to accompany the demo. The problem here is that the week or two that it takes to polish the Show Demo and the 4-5 days at the show can cut deeply into the resources available to the development of the game itself. Sure taking a few days off to go to E3 or ECTS sounds like fun. But, in reality, it just means a lot more work for the core members of the development team.

Martin de Ronde

The core business for any developer is to create games that sell, not just create games. A trade show demo should just be just part of the normal development schedule if a developer is doing things correctly. A Change Request procedure (which should be in any contract) wouldn't hurt. If a sudden request for a internal demo comes from the publisher, a Change Request procedure can effectively deal with this.

Matias Myllyrinne

Developers should seek to gain a veto clause that prevents the publisher from leaking material or showing the game without the developer's consent.

Tom Crago

It's a hard thing to estimate, but I think it's safe to factor in an extra 5% in the budget for the creation of marketing materials and the like. On a big project, this is a lot of money, but if you're prepared to spend this time you will reap the rewards. We have people preparing marketing materials that the publisher hasn't even asked for, just so we can paint our games in the most favorable light. Stuff like promotional websites, Flash movies and the like.

Game Documentation and Trade Show Demos – cont'd

It distracts from completing milestones on schedule and, if not properly taken into account when setting the timetable for deliverables in the development contract, can easily throw the development of the game over a month behind schedule. The result, not only can the publisher become upset because the deliverables are late, necessary advances are also delayed resulting in unexpected financial stress on the developer. So, the time necessary to meet these additional requirements, and what that time means in terms of your budget, should be taken into account when estimating completion time for key milestone deliverables and the contract value.

Conclusion

Make sure that as a developer you understand your obligations regarding both the Game Documentation and Trade Show Demos before you sign your development agreement with the publisher. If a publisher presents a developer with a contract that fails to take into account either the Game Documentation or the Trade Show Demos, as some do, it is up to the developer to raise these issues and make sure that they addressed in a realistic manner in the contract. If not, the impact of these two items will result in unforeseen issues when the time or ability to negotiate them has passed.

© 2003 Thomas H. Buscaglia. All rights reserved.

Tom Crago

The worst situation is to have a critical milestone, say on the 1st of June, and then in March to realize that the entire development team is going to need to spend two months putting together an E3 demo that you didn't realize you needed to provide. It can actually jeopardize projects, because even if you complete the demo successfully, suddenly your June 1 milestone is two months behind.

Localizations

By David S. Rosenbaum

Introduction

“Localizations”, or “localized versions” are versions of a computer or video game which are translated from the primary language in which the product is to be developed (usually English) into other languages for distribution throughout the world.

Sample Clause

Here is a typical contract clause from a development agreement that addresses localizations:

Developer shall provide the Game to Publisher in the English language in US NTSC and PAL formats, and Developer shall incorporate into the software for the Game up to five (5) localized versions in single-byte character languages (“Localized Version(s)”) designated by Publisher, at no additional cost to Publisher. Developer will supply to Publisher (in document and computer readable formats), all reference, instruction and other associated textual materials or other materials of the Game, and Publisher shall, at its expense, translate, produce and deliver to Developer all text to be displayed on screen, including, without limitation, user instructions related to the Game, screen text and fonts, and audio tracks if applicable. Developer shall at its expense, recompile the translated and localized materials so that the Game is in the formats and languages specified by Publisher and is otherwise suitable for use in the countries designated by Publisher. For additional single-byte character language Localized Versions, and for any double-byte character language Localized Versions (e.g., Japanese), Publisher and Developer will negotiate in good faith additional advance payments for such localizations. Whenever possible, Developer will use its good faith efforts to complete the foreign translations simultaneously with the completion of the English language version of the Game.

Ron Moravek

It’s really important up front to outline the number of localizations the developer is committing to develop. Generally 3-5 is a good number depending on the size and budget. It’s also good to outline the costs of additional localized versions upfront so there is no argument at a later date. \$15K for each additional version is a good number these days. Of course, if you can get more, all the better.

Tom Crago

In my experience, localizations are always negotiable. The default position is that you’re signing on, for the agreed amount, to make a game in one language. If the publisher wants you to provide additional languages, then this should be at an additional cost.

Localizations – cont'd

Discussion

In order to maximize revenues and a return on its investment, game publishers will look to distribute games to the widest possible audience around the world. In order to serve the worldwide audience, games need to be produced in multiple languages. As such, game developers will be called upon to produce multiple language versions, or “localize” a game for multiple language markets.

The requirement to produce localizations is commonly included in game development agreements and developers need to be familiar with the localization process when both bidding for a development contract and entering into same. Typically, localizations deal with language translations. However, other requirements may also need to be considered. For example, if a game features violent or sexual content, the game may need to be edited for content in order to be approved by local censorship authorities for distribution. In Germany, fantasy games and martial arts games are scrutinized closely for the amount of violence and, in fact, the depiction of red blood is discouraged; often, red blood is changed to green or other fanciful, non-realistic color. Whether the issue is merely language translation or content editing, a developer must properly allocate and budget time and resources in order to deliver the game as required under the development agreement.

These are the principal questions to be asked:

1. How Many Languages: the number of languages is a function of how widely the publisher intends on distributing the game. The sample clause above suggests five translations. These are commonly, French, German, Italian, Spanish and either Dutch or Portuguese. Some publishers may ask for as many as 10 – 15 localized versions. As this is a time/resource issue, the developer needs to accurately budget its expense to produce these versions.

Tom Sloper

For a developer, this is a 2-edged sword. In terms of royalties and sales, you want a publisher to cover as many languages as possible. But in terms of workload, you want fewer languages. It often happens that by the time of code release, the publisher has changed its plan for how many languages it needs for international sales.

Binu Philip

It is a good idea to include verbiage that allows the publisher to get additional translations from the developer at a set fee per language. Each translation takes work to do by programmers, and it is fair that the publisher pays it.

Localizations – cont'd

2. Are the Languages Single-byte or Double-byte Character Alphabets: this is a significant point. To reconfigure a game for a single-byte language (such as the languages noted above) does not typically require significant re-programming of game assets whereas a double-byte alphabet, which includes Asian languages, such as for the Japanese market, does require significant re-programming of game assets. Again, for a developer, the issue is time and resource allocation and to produce a double-byte localization is more time consuming and thus more costly.

3. Who Will Prepare Translation Materials: typically, the larger publishers will be responsible for translating the text and audio into the designated languages so that the developer merely has to re-integrate localized assets into the core game to replace the primary language assets. This process is typically at the Publisher's expense, although the developer should be certain that the Publisher isn't treating this expense as recoupable out of royalties. If a publisher does not have such resources and looks to the developer to undertake such task, then the developer will need to increase its development budget to accommodate both text translation and production of dialogue to replace the primary language assets. This will also impact the development or milestone schedule as more time will be required to produce these items, usually by engaging subcontractors to produce such materials.

4. When Will the Localized Versions be due: as the sample clause notes, publishers will want the localized versions to be delivered so that they may be released on a day-and-date basis with the primary version. If the publisher is tasked with preparing the translated text and audio files, then a developer's ability to deliver localized versions is directly dependent on the timeliness with which the publisher delivers the localized assets so a development agreement with such a publisher should not obligate the developer to deliver the localized versions by a date certain without also acknowledging that developer's ability to meet such date is conditioned on timely delivery from the publisher. Also, to the extent milestone payments are allocated to delivery of localized versions, then the agreement needs to reflect the publisher's obligation to timely deliver the localized assets. A developer should be wary of a milestone payment schedule that puts a disproportionate amount of a development advance on the delivery of localized versions.

Tom Crago

This is always a nightmare. As a developer you set certain critical paths, and receiving translation materials from the publisher by the specified date is absolutely critical. It never goes to plan though, and the translations usually arrive late and in patches. We've also experienced another curious phenomenon. The translations seem to change right up until submission. It's as though the initial translations are provided by someone from outside the games industry, and it's not until the QA process that the testers of the foreign language versions realize that maybe the translations aren't quite as appropriate as they might be. Then, at this incredibly late stage, we need to change the text or voice-overs in the game, exposing ourselves to a range of other potential bugs and breakages. There's nothing much we can do about this expect plan for it and build in buffers so we don't miss our dates.

Joe Minton

Push to see if they really plan to do this. We've seen many instances of the publisher's "same day" really meaning a month later.

Localizations – cont'd

5. Content Issues: as noted above, if the violence or sexual content in a game is such that the publisher believes the game will not be admitted for release in any country, the game will have to be edited to qualify the game for release. This then requires careful planning as early as the design document stage as these edited versions will have to be treated and developed as independent versions of the game which may significantly impact the development budget and development schedule. As noted above, a developer should be wary of a milestone payment schedule that puts a disproportionate amount of a development advance on the delivery of versions which are being produced for specific countries for censorship reasons. Censorship clearances require the mutual cooperation of the developer and publisher and allocating fees not otherwise directly attributable to unique localized versions may work to a developer's disadvantage especially if through no fault of the developer, such a version is not approved for release by the applicable regulatory agency.

© 2003 David S. Rosenbaum. All rights reserved.

Ownership of Intellectual Property (Game Code – Developer vs. Publisher vs. Licensor)

By Daniel O'Connell Offner

Introduction

Who owns the intellectual property rights to the game code is almost always one of the most significantly negotiated areas between the developer, publisher and licensor. The parties usually try to be extremely careful about who owns what with respect to the copyrights and other rights to the software. Depending on the type of game there are a variety of different approaches to the ownership issue. A game using licensed content is different from a game where the developer is working with publisher content or a publisher game to create a sequel or port or where the developer is creating an original game and licensing it to the publisher.

Sample Clauses

Sample 1 - Developer License With a Restrictive Covenant:

Subject to the terms of this Agreement and except as set forth herein, Developer hereby grants to Publisher and its subsidiaries the exclusive license, during the Term, to use, manufacture, sell, distribute, license, sublicense and otherwise vend the Game, Software, Engine, Content, and Ancillary Products, and their ports, conversions, sequels and derivative works thereof throughout the Territory, and the right to file an excised or redacted version of this Agreement with the U.S. Copyright Office and the right to file the first and last twenty-five (25) pages of the object or source code of the Game in the name of the Developer in the U.S. Copyright Office and similar foreign offices to secure a copyright registration for the Game in the United States and in other countries throughout the Territory; provided, however, Publisher does not have the right to exploit such aforesaid rights to the Software independent of the Game or Engine (i.e., the source code shall not be licensed, sold or distributed in an uncompiled form unless it relates to an Engine license exploitation). [Please note that this language presumes the Developer keeps its Engine and grants Publisher an Engine license.]

Richard Iwaniuk

Developers should get cost estimates for copyright/TM requirements before they agree to it in the contract.

Ownership of IP – Game Code – cont'd

Notwithstanding anything to the contrary contained herein, Publisher shall only use, manufacture, sell, distribute, license, sublicense or otherwise vend the Content in a manner which is compatible with the Game.

Subject to the exclusive license, Developer retains all ownership of the intellectual property rights to the Game, Software, Content, Engine, and Ancillary Products.

Developer may license the Engine to a third party at anytime; provided, however, Developer may not accept payment from a third party with respect to any such license prior to Publisher's acceptance of the final Milestone. Notwithstanding anything to the contrary contained herein, no game using the Engine may be released by any party other than Publisher during the twelve (12) months following Publisher's acceptance of the final Milestone.

Sample 2 – Publisher Work For Hire With a Carve Out for Pre-Existing Developer Materials:

"Developer Materials" means Developer's Pre-existing Proprietary Technology including pre-existing Source Code, to be provided by Developer on Exhibit _.

"Ownership" All work performed by Developer and all right, title and interest therein and thereto, including, without limitation, the Work, the Work Product, and all copyrights (including all renewals, extensions, revivals and resuscitations thereof), trademarks, trade secrets and patents, shall be the sole property of and shall be credited to Publisher; provided however, Developer shall retain ownership of Developer Materials. To the extent possible or required under applicable law, including, without limitation, the U.S. Copyright Act, the Work, Work Product, results, products and proceeds of any and all Services (collectively, "Results and Proceeds") created, produced or worked upon by Developer shall be considered "Works Made For Hire," specially ordered and commissioned by Publisher for use, without limitation, as part of an audiovisual work

Joe Minton

Be VERY careful of a clause like this. We often use our core engine on more than one game at a time (offering an important economy of scale in this tough-as-nails business). We've always been able to get publishers to agree to a much stricter definition (i.e. "Sci-fi, story-based 3rd person shooter") in addition to the time limit.

Ownership of IP – Game Code – cont'd

If such Results and Proceeds are not legally capable of being considered as Works Made For Hire, then, in such event, Developer hereby irrevocably and exclusively grants, transfers and assigns to Publisher in perpetuity, throughout the universe in all languages, all right, title and interest, including, without limitation, copyright, and all extensions, renewals, revivals and resuscitations thereof, Developer has or may have in or to such Results and Proceeds. In the event that under any current or future copyright law of any jurisdiction worldwide, any of the rights in or to the Results and Proceeds are subject to a right of termination or reversion, to the extent and as soon as legally permissible, Developer shall accord Publisher rights of first negotiation for thirty (30) days, and last refusal for fifteen (15) days, to match any third-party offer in connection therewith. Developer will provide Publisher with reasonable assistance to further evidence Publisher's Intellectual Property rights, and Developer will make no claim inconsistent with Publisher's Intellectual Property rights in and to the Results and Proceeds. All value and goodwill accruing in connection with the Work will inure to the sole benefit of Publisher. With respect to any so-called "moral rights," Developer hereby unconditionally waives such rights and the enforcement thereof. Developer represents and warrants that the Services are to be solely performed by full-time employees of Developer pursuant to "work-made-for-hire" agreements, which have been fully executed by both parties prior to commencement of any services hereunder and will supply a copy of each fully executed "work-made-for-hire" agreement to Publisher within five (5) business days of full execution of this Agreement. Publisher also agrees that any Third-Party materials or code licensed by Developer in the performance of this Agreement will remain the properties of their respective owners, but that Developer will represent and warrant that Publisher will receive a perpetual, worldwide, royalty-free license and right to utilize any such materials or code licensed by Developer for the Developer Materials in the Work.

Ownership of IP – Game Code – cont'd

"License to Developer's Materials" To the extent Developer's Materials are used in connection with the Work, Developer shall grant and hereby grants Publisher a non-exclusive, irrevocable, fully paid right and license throughout the universe and in all languages, with the right to grant and authorize sublicenses, to exercise all Intellectual Property rights in and to Developer's Materials in connection with the Work, including without limitation, the right to make, use, copy, perform, modify and create derivative works thereof, and distribute Developer's Materials in connection with the Work, including the right to adapt the Work for use on the Platform.

Sample 3 – Licensor Work For Hire Language With A Carve Out:

Licensor or its assigns shall own all Intellectual Property Rights in and to the Deliverables, Title, the Property and any Derivative Works made from the Property, whether or not used in the Title, including all packaging, advertising, promotional and other artwork used in connection with the promotion and distribution of the Title. These rights include all rights in and to the entire "look and feel" of the Title, all visual displays, scripts, dialogue, literary treatments, concepts, characters, backgrounds, environments, and other elements visible to the Title's users; all sounds, sound effects, soundtracks and other elements audible to the user; and all methods in which the user interacts with the characters, backgrounds, environments or other elements of the Deliverables and Title. PUBLISHER/DEVELOPER acknowledges and agrees Licensor shall be the exclusive owner of these rights as a work made for hire. PUBLISHER/DEVELOPER acknowledges that PUBLISHER/DEVELOPER' use of the Property shall not confer or imply a grant of rights, title or interest in the Property or good will associated therewith. Licensor's Intellectual Property Rights shall be indefeasible and irrevocable and shall not be subject to reversion under any circumstance, including cancellation, termination, expiration, or breach of this Agreement.

Ownership of IP – Game Code – cont'd

Notwithstanding the foregoing, Licensor or its assigns will not claim any right, title or interest in or to any pre-existing software tools and/or any game engines (pre-existing or developed for the Title) owned or controlled by PUBLISHER/DEVELOPER and used in the Title. Such tools and engines shall not be deemed to be part of Licensor's Intellectual Property Rights and shall be owned exclusively by PUBLISHER/DEVELOPER.

All materials created hereunder shall be prepared by an employee of PUBLISHER/DEVELOPER under PUBLISHER/DEVELOPER' sole supervision, responsibility and monetary obligation. If third parties who are not employees of PUBLISHER/DEVELOPER contribute to the creation of the materials, PUBLISHER/DEVELOPER shall obtain from such third parties, prior to commencement of work, a full written assignment of rights so that all right, title and interest in the materials, throughout the universe, in perpetuity, shall vest in Licensor.

Discussion

Sample 1 illustrates a very pro-developer ownership situation, e.g. where the developer is creating and self-funding a game. Here it is explicit that the developer owns all rights to the source and object code of the game, including the developer's engine. The copyright to the object code to the game is encumbered only by the exclusive license to the Publisher while the copyright to the source code or engine is encumbered only by a restrictive covenant. The developer has sufficient "clout" here to also require the Publisher to license its engine if the Publisher wants to use the engine for another game application other than the "game." The result of the exclusive license is that the developer cannot license another party to publish the game. However, the developer can license another party the right to use the engine or source code to make a competitive game subject only to a time limitation of when that game can be released. The clause itself only restricts by time period – no genre limitation – and no specific reference to competition.

Ownership of IP – Game Code – cont'd

Sample 2 illustrates a situation which is the opposite of Sample 1. Here a publisher is asking a developer to take the publisher's content, object and source code and create a sequel or port or an original game.

Everything that the developer creates here is owned by the publisher, which includes the developer's source code. The intent of the language is to make the publisher the owner of the copyright to the source and object code to the game that is created. While the publisher is allowing the developer to keep its pre-existing tools and technology, the publisher is requiring a worldwide, perpetual, fully paid up license to use those materials in future versions of the game. Effectively, even though there is a carve out for the pre-existing materials, the publisher owns everything the developer creates in connection with the game and can use the developer's pre-existing materials in the future to create new games without having to pay the developer anything.

This situation is different from Sample 3 where the developer retains its tools and technology, including the new tools and tech built or created for the new game. Also, in this situation the Publisher avoids having to pay a re-use fee or technology license fee, if the Publisher wants to have another party do a sequel to the original game that the developer creates. Developers typically negotiate these points heavily.

Sample 3 illustrates a situation where a licensor, such as a film studio, is granting a license to the publisher or a developer to make a game based on a film studio property. Here the licensor is attempting to make sure that it owns all rights to the expressions of the property, i.e., the film, and to a degree therefore the copyright to the object code to the game. The licensor is pretty clearly letting the publisher or developer keep the rights to its pre-existing tools, technology and engine and to anything new that will be created by the publisher or developer. The result is that the publisher or developer here probably owns the source code to the game, and the licensor cannot use the source code to make a sequel or port to the game without going to the publisher or developer and obtaining a license.

Licensors such as studios sometimes go this route, because they are more interested in protecting the property and all expressions or forms thereof than trying to be in either the development or publishing business, although that is changing.

Ownership of IP – Game Code – cont'd

Licensors, such as film studios, are moving into publishing and also becoming aware of the cost of not owning the underlying source code for sequels and derivatives. Accordingly, studios are beginning to ask for pre-negotiated re-use fees and licenses to the underlying source code and materials created by the developer, so that it can make or have other parties make sequels, prequels and derivative works.

For a more detailed examination of ownership of intellectual property relating to tools and technology, please see the section by David Anderson.

© 2003 Daniel O'Connell Offner. All rights reserved.



Ownership of Intellectual Property (Tools and Technology)

By David Anderson

Introduction

A developer's ability to retain ownership to its tools and technology, including its engine and other rendering software, is crucial to its ability to develop other games. (This discussion focuses on a standard "work for hire" development agreement in which the publisher or a licensor will own the finished product, as opposed to a "licensed development" agreement in which the developer retains ownership to the entire game.)

Sample Clauses

In most standard "work for hire" development agreements, the retention of the developer's intellectual property rights usually follows the "ownership" or "work for hire" provision in favor of the publisher.

Sample: Developer shall own all right, title and interest in and to (i) the computer engine contained in the Product as previously developed by Developer (the "Engine") (if such Engine is, in fact, utilized in connection with the Product); and (ii) all software utilities, editing/compiling tools, data formats or compression methods, algorithms and interface routine and general computer software design practices (collectively, the "Background Materials") which have been developed or are now being developed by Developer and/or its contractors, outside the scope of this Agreement and which will be included in or used in the creation of the Product in order to expedite its development. Developer will provide Publisher with a list of all Background Materials to be incorporated into the Product within 30 days following the completion of the Product, or as Publisher may otherwise reasonably request; provided the content of such list shall be subject to Publisher's review and will be negotiated in good faith by the parties.

Binu Philip

It is not fair to give up ownership of an engine especially if that engine existed prior to the publishing contract or even partially before the publishing contract. Tools & Tech IP ownership is one of our deal killer issues. We always walk away from deals where we can't own our Tools & Tech IP.

Ownership of IP – Tools and Technology – cont'd

Developer hereby grants to Publisher a perpetual, worldwide, exclusive, paid-up and royalty-free license, including the right to sublicense, to use and exploit the Engine, including, without limitation to modify, enhance, create derivatives, advertise, promote, distribute, publicly perform and display and sell the Engine, in connection with the Product, as well as Enhancements, Converted Versions and Sequels to the Product (as such terms are defined below in this Agreement). A list of the Background materials is attached to this Agreement as Exhibit 1 and incorporated herein in full.

Discussion

As can be seen from the example above, Publishers usually try to own or control as much of the finished product as possible.

One thing to bear in mind is to confirm that some type of carve out exists for the developer's tools and technology, since some publishers have been known to not include the language in their standard form agreement. If this happens, an unwary developer could inadvertently sign an agreement in which at the end of development, the publisher would have a strong argument that it owned the developer's tools and technology.

Even when a publisher isn't attempting a "land grab" and agrees to a carve out of the developer's tools and technology, there are a couple of issues that warrant attention.

For instance, in the Sample the carve out refers to an engine "previously" developed by developer. Again, this arguably means that anything that the developer develops for this product is owned by the Publisher. Developers would be advised to attempt to strike the word "previously" from the definition of tools and technology so that tools and technology also includes any improvements and additions made to the tools and technology during the course of development. Some publishers will agree to this and some won't. In the event that the publisher insists on owning these "improvements" a developer should push strongly for a license to these improvements so that they can continue to develop future games based on "their" latest tools and technology.

Joe Minton

We've always been able to make this list very broad so it covers everything with no questions.

Joe Minton

It is crucial to keep the license to these "improvements and additions", I believe. The argument that the publisher is now getting the benefit of all of the former work on the engine that was made on past projects is usually a strong enough one to win this.

Ownership of IP – Tools and Technology – cont'd

In addition, the Sample clause contains a license to the developer's tools and technology. In the example above, the publisher has limited the license to the product, enhancements (add-ons), conversions, and sequels to the game. However, it is not uncommon for a publisher to include language to the effect of "any other game" in this grant of rights. This is another area where a developer needs to pay close attention to the grant of rights that it is making.

Also, the license in the Sample clause is "royalty-free." Again, this is another example of the publisher looking to get as many rights as possible up front. Some publishers will argue that at least with respect to sequels, ports and conversions that it should be entitled to a royalty free license in light of the payments being made for the development of the game at hand. Many developers successfully argue that if they are not the developer of a sequel, port or conversion that uses the developer's tools and technology that they should be entitled to some type of passive royalty.

Another issue that frequently arises during the negotiation surrounding the ownership of tools and technology is a restrictive covenant. For instance, a publisher may say, "Sure you can own it, but you can't use it for anyone else's game." That's nice, but in its broadest sense doesn't do the developer much good. If a publisher is going to try to force a restriction on the developer, the developer should attempt to limit the scope of the restriction, in both terms of time and genre. For instance, a developer may be more willing to agree to a restriction that doesn't permit the developer to use its tools and technology in connection with a game in the same genre as the current game for a period of six months from the release of the current project.

The ownership of developer's tools and technology is often a hard fought area. What the developer needs to keep in mind is that often the reason the publisher has come to the developer is because of developer's tools and technology. Bearing this in mind, developers should strive to prevent a "land grab" and to attempt make sure that they own their tools and technology after development of the current project so that they are able to continue to build on their code library.

Ownership of IP – Tools and Technology – cont'd

While every contract is unique and each negotiation is the result of the parties' relative bargaining power, the ownership of developer's tools and technology, which is often nothing more than two or three sentences carving these rights out from publisher's ownership rights, is an area that should not be simply glossed over, and generally, is not an area where a developer should easily give in to the publisher.

Developers live and die by their tools and technology, and that's why it is important for the developer to fully understand the full effect of these types of clauses and to clearly grasp which party owns what, what rights if any are being granted, and the extent of any restrictions being placed on the developer as a result of the agreement.

© 2003 David Anderson. All rights reserved.

Joe Minton

In fact, in this day and age, it can kill a studio to not have rights to an engine (even if their own engine is also hooked up with middleware). Further, the publisher rarely uses the developer's engine without them (excepting engines built for this purpose like Unreal), meaning that the publisher really isn't giving anything up by including this language. Fight hard for it.

Ray Muzyka

Of course, tools and technology, while important, are only one factor that a developer needs to consider - they don't represent an insurmountable barrier to entry for the publisher or other developers and hence can't be counted on to force the publisher to work with the developer on sequels or future projects.

The Reserve

By Don Thornburgh

Introduction

In a typical game development contract, the “reserve” is a percentage of the gross royalties earned by the developer which are withheld by the publisher for some period of time. These reserved funds provide the publisher with a safety net against certain variable costs such as returned goods, forced price reductions and defective units, each of which otherwise would have undermined the publisher’s bottom line. Instead of applying future royalties against such losses, publishers seek to protect their investment by maintaining a reserve. In short, this is a “pro publisher” clause, but there are steps which a developer can take to minimize its impact.

Sample Clause

“Royalties shall be payable at the Royalty Rate (defined elsewhere) on gross sales of all licensed products, less allowances for returns, the aggregate of which shall not exceed ten percent (10%) of gross sales (the “Reserve”).”

Discussion

The sample clause sets up the concept of the reserve, but there are really three essential components to any reserve: (i) the percentage of royalties which the publisher can withhold, (ii) the types and extent of the publisher’s costs which can be paid from such reserve, and (iii) how and when any amount remaining will be liquidated and paid out to the developer.

The sample clause contains a typical reserve in terms of size (ten percent), but that is not the whole story. Ten percent of what amount, gross sales or royalties due? It is essential to be absolutely clear as to reserve calculation so neither party faces an unwanted surprise.

Matias Myllyrinne

Developers with substantial leverage may seek to push for a reserve free royalty. This may mean compromising on the royalty rates received but will add transparency and minimize the ways in which publishers may hold royalties.

Ray Muzyka

Typical ranges for reserves may vary quite a bit, from 7.5 to 20%.

The Reserve – cont'd

What costs is the reserve applicable against? Various publisher costs can be included, and should be spelled out in detail. The developer's goal, however, is to limit this list to returned merchandise, or possibly also to discounted merchandise. Note also that it is usually preferable to set a fixed percentage, rather than trusting the publisher to be reasonable when the phrase "reasonable reserve" is employed.

In addition to clear language describing what the reserve can be applied to, the developer may wish to insert additional verbiage expressly *excluding* certain costs. The following is an example of such a clause.

"Except for those expressly provided for in this paragraph, the Reserve shall not be applicable against, and there shall be no deductions for, any other costs, including but not limited to, cash discounts, costs or expenses incurred in the manufacture, distribution, sale or advertisement of the licensed products, or for uncollected bills."

With the introduction of such language, a developer should expect the reserve to be applied only against product returns, and to the extent that such returns are less than the total dollars withheld by the publisher, the money should be paid to the developer.

Payment of the reserved amounts can be accomplished in several ways. The preferred approach, from the developer's perspective, is to having rolling payments. In other words, the reserved amounts are withheld during one period of the contract and distributed in a later period. For example, if the publisher is paying royalties to the developer quarterly, an amount (10% of gross sales in the sample clause) is withheld during the first quarter, but the remainder (if any) is paid during one of the subsequent quarters. Developers prefer only a 90 or 180 day turnaround for liquidation of undepleted reserves, but many publishers will insist that reserves be liquidated only on an annual basis.

Joe Minton

Royalties due is what we are used to seeing. Publishers generally push for 10-15%, although I've seen the number go up to 30%. While I want as low a number as possible, I'm more concerned with the liquidation term. (If it is 3 or even 6 months, I'm not going to care as much about a high reserve.)

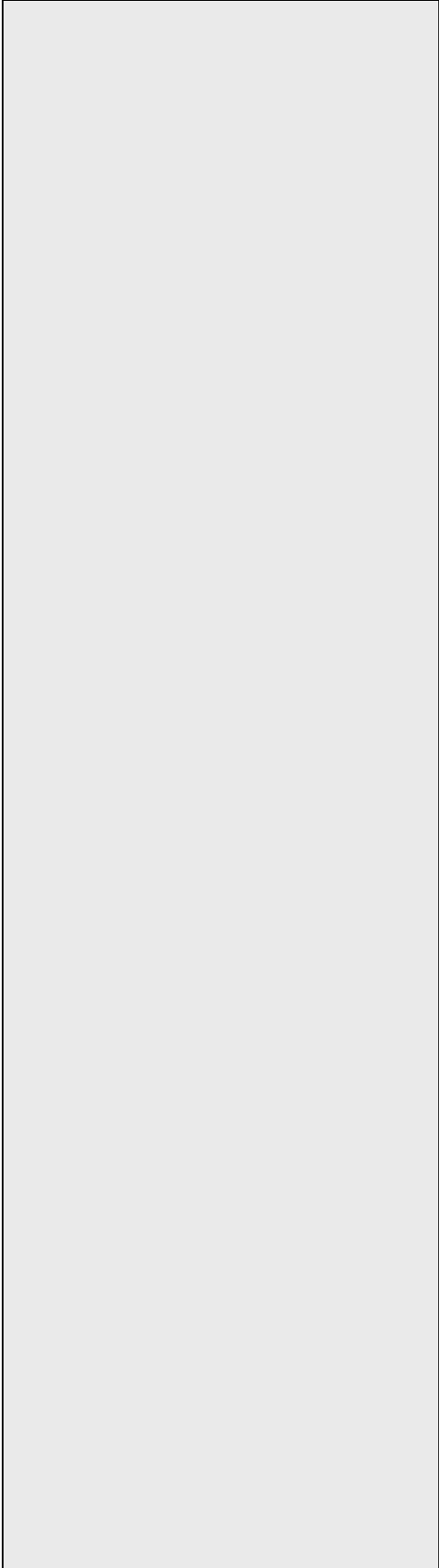
The Reserve – cont'd

Successful negotiation of this point depends largely upon the developer's overall bargaining position with the publisher, but circumstances can give rise to some compelling arguments on this issue. For example, if the reserve is to be applied primarily to returned merchandise, a developer should examine the likely timing and basis for returns. Are sales going to be primarily through direct sales (higher returns), or via ordinary retail channels (fewer)? Also, who should bear the risk of customer dissatisfaction? Is the title based upon an extremely popular property, and the developer is simply bringing it to a new platform, or is the property/content unproven, and the developer is being engaged to try to make it a hit? Arguably, the developer should assume more of the return risk in the first instance, but perhaps the publisher should take the risk in the second. The argument can cut both ways, but with some creative arguments, developers sometimes can reduce the impact of burdensome reserves. In short, the reserve size and terms are tied to the amount of risk the publisher is taking. Since certain favorable factors (an established game, retail channel, etc.) reduce that risk, a developer may be able to negotiate better reserve terms in such cases.

Conclusion

In summary, the developer's three primary goals with regard to reserves should be (i) to minimize the total amount the publisher can withhold, (ii) to limit the types and extent of publisher costs which can be paid from the reserve, and (iii) to expedite payment of reserve amounts.

© 2003 Don Thornburgh. All rights reserved.



Lawyers Panel Bios

Jim Charne – Chair, Lawyers Panel

Jim Charne, chair. Jim Charne is a California, New York, and New Jersey lawyer who has provided legal representation for clients in all phases of interactive software entertainment since the mid-1980's. He entered the industry in 1983 as a producer for Activision, and served as VP Legal & Business of console developer Absolute Entertainment. Jim was President of the Academy of Interactive Arts and Sciences (AIAS) from 1999 to early 2001 and is a voting member of AIAS, IGDA, and the Recording Academy (NARAS). Jim has been chair of the Legal and Business tutorial at GDC since 1998, and writes "Famous Last Words," a monthly column on games contracting issues for www.igda.org. He can be reached by email at charne@sprintmail.com or on the web at www.charnelaw.com.

Dave Anderson

Dave Anderson is a partner with the firm of Offner & Anderson, P.C. in Los Angeles, California. His practice focuses on intellectual property matters relating to the interactive entertainment industry, mobile gaming, traditional merchandise licensing, and trademarks. In connection with the interactive entertainment industry, Offner & Anderson represents both publishers and developers in connection with all aspects of the industry. Dave can be reached by email at danderson@offneranderson.com or on the web at www.offneranderson.com.

Tom Buscaglia

Tom Buscaglia practices technology law in Miami, Florida www.game-attorney.com. He obtained his law degree, cum laude, from Georgetown University. Tom represents game developers in all aspects of their legal and business needs. He authored a chapter entitled "Effective Developer Contracts" for the *The Secrets of the Game Business* book and published a series of articles on GIGnews.com for rookie game developers. Tom was a presenter at the 2002 GDC, San Jose, and is a guest lecturer at Full Sail in Orlando, Florida. Tom recently formed Games-Florida, a non-profit, to expand the Game development industry in Florida www.games-florida.org, sits on the Board of the Digital Media Alliance of Florida and coordinates the South Florida IGDA Chapter. Tom is also Supreme Warlord of FaTe's Minions www.f8s.com.

Trevor Fencott

Trevor Fencott is an attorney at Goodmans LLP, Canada's Entertainment Law Firm. His practice focuses on corporate, securities and intellectual property law in the New and Interactive Media industries. He currently assists content developers, publishers, agents and the investment community with a variety of matters that include financing, strategic transactions, intellectual property, licensing, software development and publication agreements. Trevor is a founding Committee Member of the International Game Developers Association (IGDA), Toronto chapter, and currently serves on the Game Development Program advisory boards of both Sheridan College and Algoma University.

Don Karl

Don Karl is a partner in the Emerging Companies and Technology Group of Perkins Coie LLP's Los Angeles office and works with clients in the technology, entertainment and new media industries. Don practice in the videogame industry principally involves the representation of developers in connection with mergers and acquisitions, financings and other strategic transactions as well as development agreements. In November 2002, he represented Angel Studios, Inc., in its sale to Take-Two Interactive. Don is a frequent speaker on entertainment/technology topics and has spoken at the Game Developers Conference and E3.

Dan Offner

Dan Offner is a partner at Offner & Anderson, P.C., a law firm in Los Angeles, California. He practices intellectual property and corporate law for videogame publishing, developer, and toy company clients.

David Rosenbaum

David Rosenbaum is a member of Fischbach, Perlstein & Lieberman, counseling clients in the interactive entertainment, music, motion picture, television, publishing and licensing industries. In the interactive industry, Fischbach, Perlstein represents many leading game publishers, game developers and technology licensors. From 1980 to 1992, David was a counsel at Paramount Pictures, where he handled a wide variety of legal matters from marketing and distribution of motion pictures to licensing of character and entertainment properties; in 1986, he became Vice President, Legal Affairs for Paramount's Marketing Division and was in charge of the legal affairs of Paramount's consumer products licensing operations.

Don Thornburgh

Don Thornburgh is an attorney and founder of Strategic Law Group in Los Angeles, California. His practice focus is intellectual property in technology, entertainment, multimedia products and services, and Internet issues. He has advised a diverse client base, including those engaged in the development of computer games, online entertainment, film making, music and sound effects, Web-based services, software licensing, and wireless technologies. Prior to forming Strategic Law Group, Don worked in the entertainment department at Loeb & Loeb, and before that in the technology group in Brown Raysman's L.A. office. He earned his B.A. from Pomona College and his J.D. from the U.C.L.A. School of Law. Don can be reached by email at don@strategiclawgroup.com or on the web at www.strategiclawgroup.com.

Commentors

Tom Crago – CEO, Tantalus Interactive

Martin de Ronde – Commercial Director, Lost Boys Games

Richard Iwaniuk – Director of Finance, BioWare Corp.

Scott Miller – CEO & Founder, 3D Realms

Joe Minton – President & CEO, Cyberlore Studios

Ron Moravek – COO, Relic Entertainment

Ray Muzyka – Joint-CEO, BioWare Corp.

Matias Myllyrinne - Business and Finance Director, Remedy Entertainment

Binu Philip – President, Edge of Reality

Josh Resnik – President, Pandemic Studios

Tom Sloper – President, Sloperama Productions

About the IGDA

The International Game Developers Association is the independent, non-profit association established by game developers to foster the creation of a worldwide game development community. The IGDA's mission is to build a community of game developers that leverages the expertise of our members for the betterment of the industry and the development of the art form. Do the right thing and join the thousands of members, studios and partners that help make this mission a reality.

Personal Membership

The IGDA membership is made up of programmers, designers, artists, producers and many other development professionals who see the importance of working together to advance games and game development as a craft. Your involvement is critical to the success of your career, the IGDA and our industry.

By joining the IGDA, you join a worldwide community of game developers that shares knowledge, insight, and connections. From local chapter meetings, to online discussions, to committee output, the IGDA provides invaluable information and resources.

Studio Affiliation

Your team is your most valuable asset. As a Studio manager, you can reward and inspire your development team by affiliating with the IGDA. By joining the Studio Affiliation Program, a studio provides all of its employees with personal IGDA memberships, allowing them to connect with their peers and grow professionally and personally. In addition, Studios receive their own unique benefits and discounts, all while showing support for the community. Refer to the back cover of this report to see all the great Studios that are part of the IGDA.

Industry Partner

Your organization is essential to game development. Make a difference in the community you've helped to create by becoming an IGDA Partner. Send the message to game developers that your organization supports the growth and development of games as an art form, and backs the community at its roots. Gain exposure with IGDA members for whom game development is a way of life. The IGDA upholds the common agenda of game developers and the game industry. Be a part of that agenda by becoming an IGDA Partner.

Make a difference:

www.igda.org/join