All games start as an idea, something like "Wouldn't it be cool to be a space marine and blow up zombies on Phobos" or "Wouldn't it be cool to be a pilot in a starfighter involved in an epic struggle to overcome the oppression of a star empire gone bad" or "Wouldn't it be cool to drive modified street cars on Tokyo streets at night." These idea sparks are often the source of long conversations between developers late into the night at the studio. Another potential starting point for a game is a licensed property; i.e., "make a RPG/RTS/action game using XXX license." (Fans may want to play that license specifically. Major licenses include Star Trek, Star Wars, D&D, WWE, Lord of the Rings, and Harry Potter.)

This article discusses how to turn the structure that your business context and your game ideas provide into a game concept worthy of fleshing out into a game design document.

Business Context Shapes Design, Or Does Design Shape The Business Context?

First of all, I am not asserting that having your business context in hand will act as a magical tool that will turn any game idea into a well-thought-out game concept. It is only an important aid to assess the requirements that your game idea is implying. Some game ideas (such as the faithful recreation of Middle Earth where the whole world is modeled with strong AI, 3D graphics capable of great indoor and terrain rendering, where an unlimited number of players can join in together on both sides of epic conflict between good and evil) cannot be reconciled with the business parameters of two artists and a programmer looking to break into the industry, who have six months of living expenses available to them on their collective credit cards. That Middle Earth concept is an example of a game that will dictate the business parameters. If we take the business parameters of two artists and a programmer, they might want to recreate an arcade classic on the Nintendo Game Boy Color or Advance, use it to secure their first deal, and then move on to more ambitious projects.

For many game projects there is a middle ground where the business parameters and the game idea go back and forth and refine each other. Perhaps the developer pitches a massively multiplayer game to a publisher who is wary of the costs and risks behind massively multiplayer. From these talks it is quite possible the developer will end up creating a game that exploits a license the publisher has rights to and features a much more modest multiplayer feature set. This is not an acceptance of a mediocre plan; rather it is a mature development of the idea into a viable concept. A viable concept is a game that people with capital believe will be seen through to completion, with a high probability of favorable reception in the market to overcome the inherit risk in game making.

Reconcile the Business Context and Game Idea Early

This process of refining the game idea and business context is the earliest stage of a game project. All projects reconcile their business contexts and the game idea at some point. Tragically, for too many projects this reconciliation only occurs after the project manifests itself by
underperforming, usually by missing milestone dates. Some projects have a painful reassessment where senior management allocates more funds and grins and bears it. For other projects, senior management interprets this late reconciliation as an unpleasant surprise presented by an immature development team and consequently cancels the project.

Allocating more funds and time to a project is a common occurrence, and because it is commonplace, too many developers think it does no harm to themselves and no significant harm to the publisher. That is fallacy; when a publisher is forced to spend additional dollars and push back the release of the title, there are many negative impacts.

First of all, the publisher must extend additional money to the developer. This is an obvious point, but it means that these funds are unavailable towards the development of another title with another developer or (worse for this title) funds may be drawn from the marketing budget to pay for this overage.

The second impact is that the publisher has to delay when they will be able to start recouping their investment and see a profit that they can put to work in future games.

The third problem is that the marketing effort is deflated as the awareness for the game is now ill-timed, and it will be difficult for the game cover that marketing was able to secure for your game last quarter to have real value 18 months later. Right or wrong, the developer is the vendor and the publisher is the customer, and the adage that the customer is always right holds firm in this case, with the developer being tarnished by the reputation of poor estimating capability.

Another reason to avoid going back for extra money and time from your publisher is that the business deal will never improve. A loss of royalty points is common; sometimes you will see a shifting of intellectual property rights. In the extreme sometimes the developer agrees to an assignment of equity in the project to the publisher. In the case of shifting equity to the publisher, the developer is strongly advised to get full value for that equity; no matter how small an equity stake the publisher takes, it will make all other publishers avoid doing business with the compromised developer for fear of a conflict of interest and confidentiality concerns.

The developer is also losing time by going over his time budget, and spending more time on a project with the business deal worsening is not a good goal.

The final reason to avoid a late reconciliation of the business context and game idea is to prevent team members from becoming disillusioned and moving on to another company.

At Taldren we have released Starfleet Command, Starfleet Command: Gold, Starfleet Command: Neutral Zone, Starfleet Command 2: Empires at War, and Starfleet Command: Orion Pirates in less than two years. At the same time we gathered more fans and have always produced a profit for our publisher. Many of our employees are loyal to Taldren because of the steady pace of release; they know their work will be released and not wasted.

### The Effects of a Slipped Game

1. Less working capital for the publisher.
2. The total advance is tied up longer than expected.
3. Marketing dollars are often wasted as the hype bugle is blown too soon.
4. The developer’s reputation almost always suffers.
5. The business deal never improves for the developer.
6. The developer loses the opportunity to work on other titles.
7. Team members are in danger of becoming disillusioned, and the team may suffer uncomfortable turnover.

Ion Storm has to be the most infamous example of the consequences for late reconciliation of the business context and the game idea. Ion Storm was founded around John Romero, who is credited with the design of Doom—perhaps the greatest PC game ever. The UK-based Eidos was flush with cash, and John Romero left id just as Quake was entering its final stages towards release. Eidos needed to put the surplus capital from the Tomb Raider series to work, as all businesses must do. Tomb Raider was so successful that Eidos needed to get into a number of
games, but established top developers were already booked, so Eidos would need to go with a less established development house. The idea of taking advantage of the designer behind Doom and creating a new development house is not a bad idea; in fact it is a good idea. Experience, a built-in fan base, and a great story for the media would create an environment that would be conducive to game development, one would think.

Ion Storm was founded with the vision statement that design is king. Even this is not a bad idea; treated properly this would mean that Ion Storm would capitalize on its core strength -- game design embodied by John Romero -- and take advantage of existing game engines. Looking at how Ion Storm interpreted their vision statement would reveal where Eidos made their mistake. Ion Storm used the vision statement, design is king, to treat game development as a pure art form and lost respect for a strong development process. Ion Storm's marquee project Daikatana suffered all of the ills described above. Whole engine retooling caused massive delays and required Eidos to double the already overgenerous advance of $13 million to $26 million to keep Ion Storm's three projects rolling.

Daikatana did not just lose face in the game press, it became the material for much derision, and even the local Texas newspapers saw the poor management at Ion Storm as a good story for a series of columns. Ion Storm not only suffered crippling turnover, but some employees helped feed the negative press storm by leaking to the press ugly internal email. John Romero was forced to hand over the company to Eidos, and their games shipped to little success. Ion Storm's Dallas office has been closed by Eidos to what amounts to a large write-off of Lara Croft earnings and a reputation for Eidos to overcome. In fact the quieter Ion Storm Austin studio run by Warren Spector, which shipped the critically acclaimed Deus Ex, is now looking for a shiny new name to operate under to distance that studio from the ill-fated Dallas studio.

The sad thing is that John Romero really can design games; just play Doom any day and you will see how amazing a game it was and still is. And Eidos turned on the cash to set up the game for greatness. It is just heartbreaking, really, to think about the potential of Ion Storm and to see it fall for lack of rigorous development methods.

What can be worse than either pumping more money into a late project or canceling a project? Shipping it. It should never be done, but almost every large publisher has shipped a game well before it was finished. I don't mean just with bugs; I mean before critical parts of the game were complete.

Descent to Undermountain from Interplay is a classic example of a game that was shipped too early. The idea behind Descent to Undermountain was to take advantage of two key assets of Interplay: the Advanced Dungeons and Dragons license and the mega-hit Descent. Management at Interplay decided it would be a snap to plop down some fantasy environments, characters, and monsters to bash. Management decided the Descent game engine would be ready for immediate development into another title. Most publishers do not have a strong technical director available for code review. Yet at the same time many publishers also negotiate the terms of the publishing deals to either own the software engine behind the game or have a license to the software engine. Descent to Undermountain was a case where the revenue opportunity was so large as to prevent an objective review of what it would take to get the game done. The original business parameters for this title called for a budget of only six months of four developers' time. No established development house was chosen to do the job; rather an ambitious independent contractor programmer stepped up, and various artists at Interplay contributed to the project. No project manager was allocated. Let me share with you what Gamespot thought of the results of this game after it slipped to three years and six times the original budget:

From Gamespot review of Descent to Undermountain:

But somewhere along the line something went horribly wrong, and now gamers are asking themselves two questions. The first arose merely out of befuddlement: How could the company that produced Fallout also be responsible for one of the lousiest games to come down the pipe in quite a while? The second, though, addresses a much more serious issue: Why did Interplay ship the thing when it wasn't even close to being the sort of cutting-edge product the hype machine had led us to believe it would be? ...There's probably no way to learn the answer to the first question, but-thanks to some very frank members of the Descent to Undermountain team-the answer to the second is now common knowledge. The game went out...
when it was scheduled to go out (in time for a Christmas release) even though it wasn't ready. That's not just me speculating; that's precisely what a member of the DTU team stated in a recent post on Usenet.

When a project is three years in the making and six times the original budget, there is tremendous pressure to just ship the game. At the time, Interplay was receiving a huge amount of attention for Descent to Undermountain; everyone wanted a truly 3D dungeon romp. (Dungeon Siege, the first really 3D dungeon romping game, and BioWare's Neverwinter Nights, which is a more detailed 3D implementation of D&D, weren't released until 2002.) Interplay thought at the time that with all the hype, maybe, just maybe, the early sales in the first few weeks would be large enough to recoup a significant portion of the costs. It was also Christmas time when 40 percent or more of our sales as an industry happen. Interplay had three choices:

1. Ship it now.
2. Cancel the project altogether. (Remember lost money really is lost, and it is best not to chase it.)
3. Find a real AAA development house and start over with a new large budget and two years more of development time. (Really the same thing as canceling the project.)

Unfortunately for Interplay at the time, canceling the project or starting over with a new developer appeared to be more expensive than shipping the title. Let us see what Gamespot thought of this decision to ship the game:

From Gamespot review of Descent to Undermountain:

The lesson to be learned should be obvious: If you're gonna ride the hype machine, you'd better deliver the goods. Sadly, DTU doesn't even come close-and the worst part is that sometime over the next year or so we'll probably see this same story played out all over again.

So what have we learned today? That pushing a product out the door before it's ready makes loyal customers angry; that game developers should keep at least one eye on what's going on in terms of technology when working on a new game; and that if you buy Descent to Undermountain after reading this, you get what you deserve.

Descent to Undermountain shipped in a condition that was far below the industry standards of the time, Diablo and Quake II. The hype behind this game also crushed it. It is just possible that if Interplay had developed this title quietly, hard-core fans of AD&D and/or Descent might have bought 20,000 copies and been patient for a patch or two. I am not saying this is a great idea, but it is better than a hype storm. This is a poor way of doing business; the game industry shows time and time again that the mega-hits are just games that offer straightforward gameplay with strong production values. Wacky or niche games or poor craftsmanship are not rewarded. Just make a few quality titles and you will spend a lot less money in development, and your individual titles will have more capital to work with.

Descent to Undermountain was a perfect case where the game idea and the business parameters were in conflict. If Interplay wanted a title in six months and had only a modest budget to accomplish it with, then Interplay should have commissioned the developers of Descent, Parallax, to create a cool expansion pack for Descent and they should have contented themselves with the sales of an expansion pack. Perhaps it was perceived that with Descent II already in development at the time, it would have been competing for sales. The other option was for Interplay to allocate the funds they were to later plow into Stonekeep II and hire a top developer to create a 3D dungeon romp of quality. Stonekeep II would later go into production for five years and then be cancelled. You must create a game that is compatible with your business context or fail.