At some point with a keynote there’s a special, excruciating request: “Can we get a title for your talk?” Like many of you I’m beyond busy, so I did what I normally do in this situation – I make something up. I wrote to Carrie, how about this idea: what will great serious games be like? That sounded very keynote-y. (That’s the added pressure on speakers when they put “keynote” to your talk.) And she wrote back, GREAT, I can’t wait to hear it. I replied to the email, “yeah, me too.”

When I sent this idea, I had spent a good portion of my spring discussing and exploring the question of where serious games should head. Frankly, these perspectives apply as much to where I think not just serious games but all game should head, especially the kind of games we will be playing a lot of or asking people to play a lot of.

I also wanted a title that implied that we don’t really have these games yet. That’s not entirely true, of course. There’ve been some really interesting projects and semi-successes, especially in the last few years. I’d like to think the successes include one or two of the games I’ve worked on, but that’s debatable.

For the most part, I don’t think we have had an epic serious game success yet. First off, I always say this every time I give a talk. All games are serious. I recently berated a friend who was working with a media company on serious game. Before the serious game was ready, the company put a free to play game on their web site, and said “come waste some time.” I couldn’t believe she was allowing her company to write that about games.

Throughout the history of game studies, we have been struggling with the idea that games are not productive. I don’t think any time I’ve spent with games has been particularly wasteful. The only time I think a game is wasteful is when it sucks. Or when I suck at it. Many academics and people in the game industry complain that the word serious games sucks, and it should be this or it should be that. Eventually the word serious goes away. But it’s not going to go away if we’re promoting the notion that games are wasteful in and of themselves. If you don’t build games up I’m going to have to keep going around Washington advocating serious games, but won’t be my fault.

So, all games are serious.

I require myself and everybody else who speaks at my conferences and events to begin their talk with a list of the games they are currently playing. It’s hard to claim that you can...
talk knowledgably about games if you can’t produce this slide.

My Current List Contains: Limbo, Goldwalker, Uncharted, Mine2049er (8-bit Atari), Red Dead Revolver, Ogs (iPad) and Minecraft!!!

I like to see this slide because it gives me ideas of the games I should be playing. Gives the audience an idea of what I’m doing, what I’m about.

My four year old loves Limbo. Seeing this slide gives you some sense of how I am as a parent. He thinks it’s funny when daddy dies in it. He thinks it’s hilarious. I try to play a lot of different games. This is important if you’re a game designer.

So I gave two talks this Spring that I want to recap quickly. At Game Developer’s Conference, I gave a quick talk about the Reimagining Drill and Skill games. This approach is antithetical to the push for big games teaching hardcore concepts and deep understanding. Using games for drill and skill is not necessarily something people disagree with. There is an interesting opportunity to make really good new generation drill and skill games on finite topics that can change how we practice behaviors.

We know that practice as pedagogy can work. The more you practice something, the better you get at it. Practice grows neurons. In the games for health field, a key challenge is sustaining behavior change to overcome bad habits. Practice can help. Say what you will about the testing phenomenon in schools, but a huge issue for why kids don’t perform better in schools is that they really don’t study enough. It’s just that simple.

I can lecture all day about how something in algebra works, but if I can make a game that shows you, that would be better. There is a lot of learning that still requires (necessitates) skill development, especially in things like math. For my kid right now, it’s addition. He’s six. He needs to master the visualization of math things – like number lines. I was working with him to show him how to use a number line and then add to it. He can see conceptually what’s actually happening.

At the end of the day there’s another part of him that just needs to learn 3+4 and 4+3 over and over again so that he doesn’t have to go like this (gestures), he just says SEVEN. Because later on, he needs to be able to say “seven” and move on with his life. I spent some time looking at that idea and what it would mean. What would be the types of supporting frameworks that would actually make these types of games work?

The other talk I gave this Spring was to Google, titled Games Everywhere: The Larger Role for Web Platforms and Services for Games & Serious Games. You can find the talk on YouTube (http://www.youtube.com/watch?v=XPaCwjhZ2aY).

I felt honored to be invited. It was an awesome experience to go to Google. It forced me to think harder about the technical challenges I’ve been encountering working with some of my corporate clients. They have very different notions about the type of problems the software we are building for them needs to solve.

The background (covered in the Google talk) is really important if you are thinking of building serious games for large scale organizations because there are things they will tell you once you get in and start working with them that change the paradigm of the actual software you will have to develop. I ended up calling for a “game application engine” that can remotely render secure gameplay. I’ll touch on many of those elements in this keynote.

Preparing the Google talk forced me to think a bit harder about some of the technical threads I’ve been working on, especially as I was engaging with my own corporate clients, who have very different ideas about software requirements.

A game for a corporate client is browser-based. It is HTML-based. It’s not going to rely on native executables. It will need to hook in to corporate data streams. Native apps on desktops in corporations are going away. I believe a huge swath of gaming and serious gaming is going up into the cloud in the form of rendering directly to the browser or browser-based embedded aps.
Causal Modeling to Foster E-Participation in the Policy Decision-Making Life-Cycle

Digital Game based Learning for Undergraduate Calculus Education: Immersion, Calculation, and Conceptual Understanding