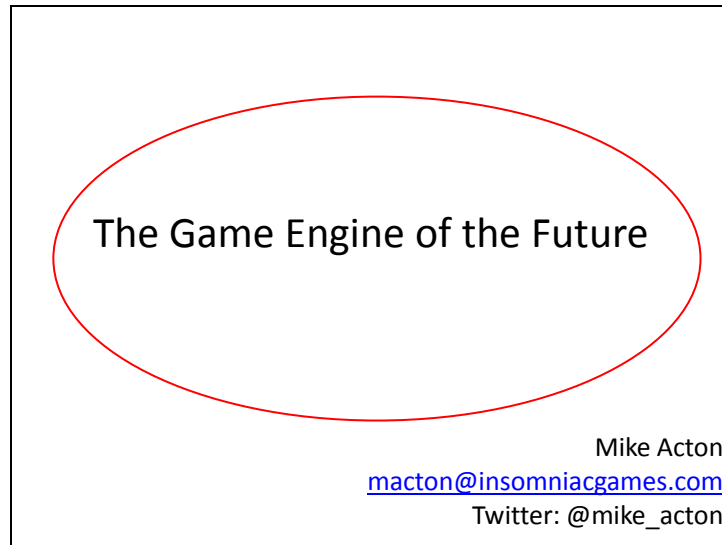


Slide 1



Keynote I gave at the Nordic Game Stockholm Summit. <http://nordicgame.com/?id=93>


Which was on 05 November 2010. It's 09 December today. I think I can remember most of the things I said to add some notes here.

PS: It was 0 degrees at the time. Which apparently is warm in Stockholm.

PPS: I'm sorry to all the people I borrowed images from in this presentation.

PPPS: This is all my opinion. I'm not pretending to represent some larger group of game developers. This is just me.

Slide 2



Why am I here?

So I flew in from Bogota for this. I was on vacation. Nice and warm (kinda rainy though.) Why would I leave for this?

Slide 3

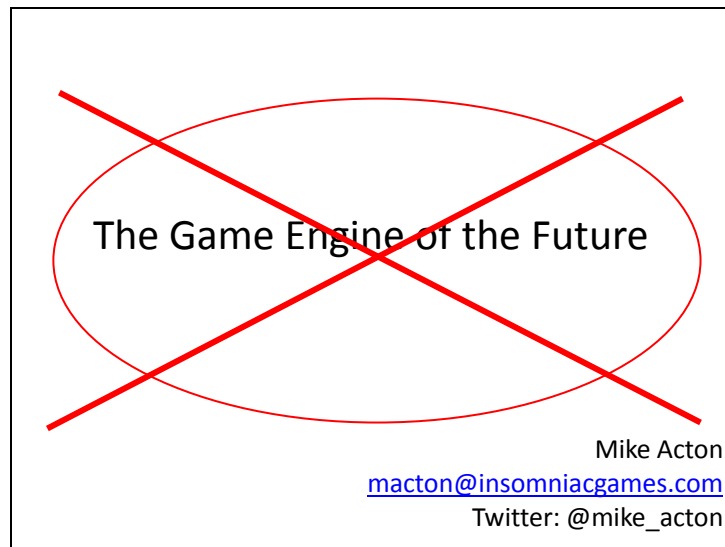


*Say anything you want.*

Right. Because Jacob said I could say anything I want. How often do you get the opportunity to do that?!

Except maybe on Twitter. Follow me here -> [http://twitter.com/#!/mike\\_acton](http://twitter.com/#!/mike_acton)

Slide 4



Since it's "anything I want" – I'm not going to talk about the Engine of the Future.



It's all about the data.

So let's talk about data.

Right. This is it. The fundamental thing. There's nothing else.

Are you a programmer? Your job is not to write code. It's to transform data. That's the only thing that's actually happening. Everything else is just your imagination adding unnecessary extra problems.

I talk about this a lot. Check it out:

[http://www.insomniacgames.com/research\\_dev/articles/2010/1522262](http://www.insomniacgames.com/research_dev/articles/2010/1522262)

<http://cellperformance.beyond3d.com/articles/2008/03/three-big-lies.html>

<http://cellperformance.beyond3d.com/articles/2009/08/roundup-recent-sketches-on-concurrency-data-design-and-performance.html>

Oh, and here's an article from 2006 that I never really finished:

<http://cellperformance.beyond3d.com/articles/2006/04/performance-and-good-data-design.html>

And Aras Pranckevičius from Unity just wrote a rant that's worth checking out too:

<http://aras-p.info/texts/files/UnityBootcamp2-CodeRant.pdf>

My pals over at DICE have some good stuff:

<http://publications.dice.se/>

And since I have your attention:

[http://www.insomniacgames.com/research\\_dev/articles/](http://www.insomniacgames.com/research_dev/articles/)

Slide 6



THE END.

That's it. Go home now.

Slide 7



At this point I still had a bunch of time to fill. So figured I'd tell my own GameDevStory.

PS: It's fun game if you haven't played it: <http://itunes.apple.com/us/app/game-dev-story/id396085661?mt=8>



Slide 8

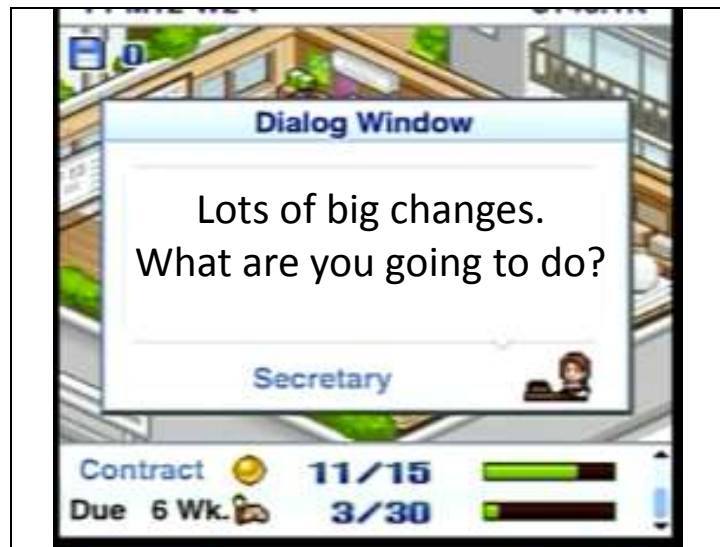


So you may know that at Insomniac we're working on a bunch of new stuff. (If you want to keep up with it, follow @insomniacgames on Twitter)


Slide 9



We're going cross-platform, right? So that means big changes for us in the Core (Engine) group. And we've started working on a new game with a new publisher. And we've announced two more games coming out soon. And we're making all these changes in the middle of a console cycle. And more than that, we're rethinking how we do things and what's important to us as a team.



It's a challenge. It's been a challenge for the whole Core team. We've had to adapt rapidly and really figure out what we're doing quickly.

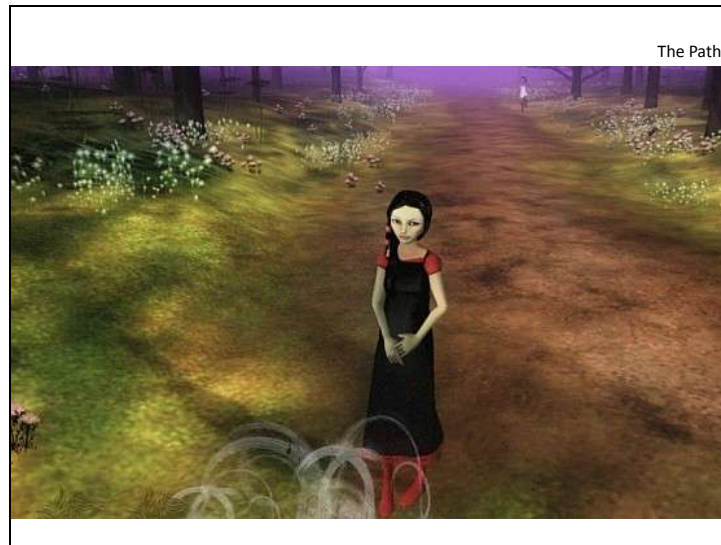


Shared my personal vision with  
team.

So a couple weeks before this talk, I took stock of where I was and shared my personal vision with the team. Why is this work that we are doing important \*to me\*?

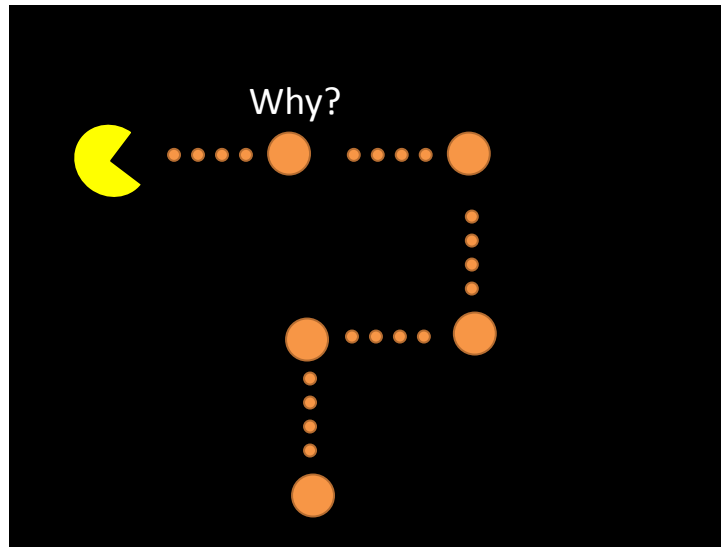
And that's where this presentation comes from.

Slide 12



This is about my own personal path.

PS: Check out The Path <http://tale-of-tales.com/ThePath/>  
I couldn't get it to work on my PC, but it seems pretty cool.



Why do I do what I do?

Why do I care?

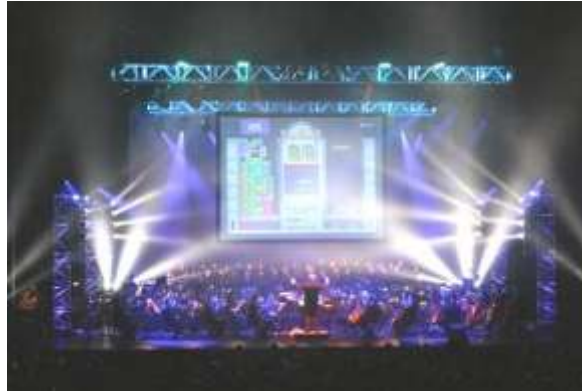
Why do I get up in the morning?

Why?

For the money?



For the fame?



PS: This is Video Games Live

Check it: <http://www.youtube.com/watch?v=JVHGy9XEF9I>





Oh, yeah. Too bad Ms. Pac-Man already has a man.

Creation Ego?



(No offense to Sid Meier. Just the best example.)

Maybe a little...



**Mike Acton Presents:**

Mike Acton's Vision of  
The Engine of The Future

*Presented by Mike Acton*

Hey look! I have my own muppet. The Community team at Insomniac made that for an internal presentation. I have it on my desk. It's damn cool.

Because I believe...

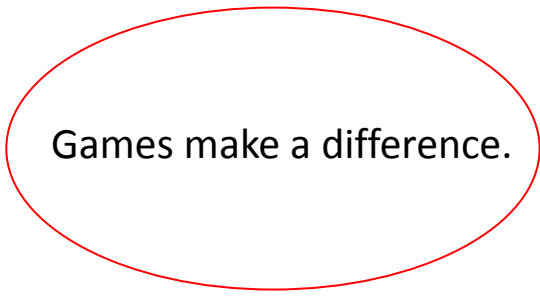


I believe this deeply. They are an important part of who we are as people and as part of our culture.

Imagine a world without games.



It would suck.



Games make a difference.





Seriously. To this very day I work out my route between places based on some concept of number of moves and try to organize things in a loop where I can manage the resources I need:  
Home -> Coffee -> Work -> Gas -> Home

Slide 25



My daughter and I played this game for a year. Together, almost every night. It's a part of our shared history now. And some really fond memories.

Slide 26



My daughter has put THOUSANDS of hours into the Sims. \*This\* is what she's going to think back on fondly when she's my age.



Slide 28



Right so Roger Ebert and a bunch of other people say "Games are not art."

Slide 29



They are wrong.

Not *\*all\** games are art of course. Just like not all movies or books or paintings or whatever. But they are at least as much *\*potentially art\** as anything else.

Slide 30



Suck it.



Games let people participate in their own (or others) fantasies.

Damn, these costumes are impressive.

PS: You might want to check this out too -

[http://www.insomniacgames.com/tech/articles/1107/files/fantasy\\_of\\_the\\_familiar.ppt](http://www.insomniacgames.com/tech/articles/1107/files/fantasy_of_the_familiar.ppt)



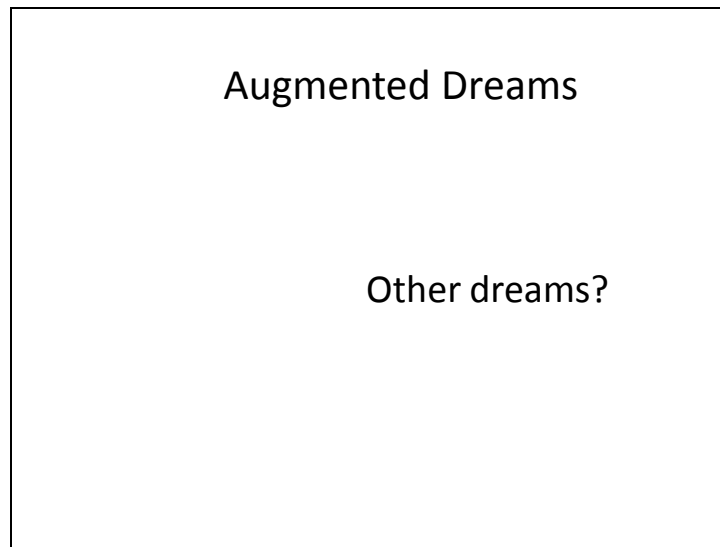


Games are a way to augment your own dreams. Like World Domination.

## Augmented Dreams



Or creating a Civilization in your image. Or building an Empire.



You may have dreams that don't involve taking over the world. I'm pretty simple that way though.



Games are a way of expressing your creativity. Little Big Planet – Look at how much crazy stuff people have done for this!

For instance, check what these guys did: <http://www.teamkaizengames.com/kblog/kfbb-did-a-cool-piece-on-us/>

You can find Josh here: <http://twitter.com/#!/ZookeyTK>

## Interactive Creativity



Pinball Construction Set (1983)

Oh, damn. This was *\*so\** awesome.

And Bill Budge is on Twitter too: <http://twitter.com/#!/billb>

## Interactive Creativity



The Incredible Machine (1992)

Crazy creations!



OMG. Minecraft is nuts. Productivity = 0.

PS: Carl Glave on our team here just gushed about it recently:

[http://www.insomniacgames.com/research\\_dev/articles/2010/152558667](http://www.insomniacgames.com/research_dev/articles/2010/152558667)

Slide 39



Games can also just be an OCD release.



## Business vs. Art

But whenever we talk art, someone has to jump in with the whole “It’s just big business” line.

As an aside (can I have an aside in a presentation that’s pretty much all an aside?) – I hate it when “Indies” go on about how “we” suck because we’re “just in it for the money” and how they’re just “in it for love” – whatever, you fuckers – If I didn’t love this, I wouldn’t be doing it, either. PS: Here’s a ← HUG ?

Don't get hung up...

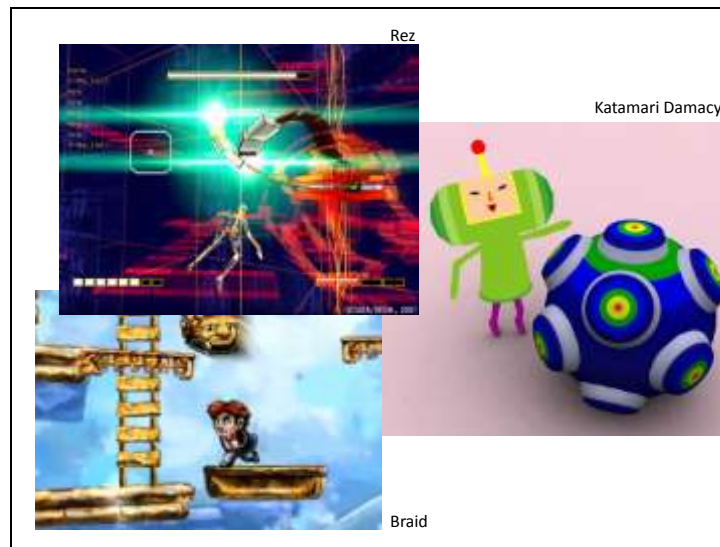
Innovation != Sales

But when talking about “The Art of Games” it’s easy to get caught up in the argument that innovation isn’t rewarded in the marketplace.

(Radical) Innovation != Sales

And there *is* some truth to that. But usually it's *radical* innovation that isn't as successful in terms of sales.

Slide 43



Of course, that doesn't mean it's not important. Or successful. One could hardly argue the real success of these titles. Though sales aren't in the same category as some of the less "innovative" titles.



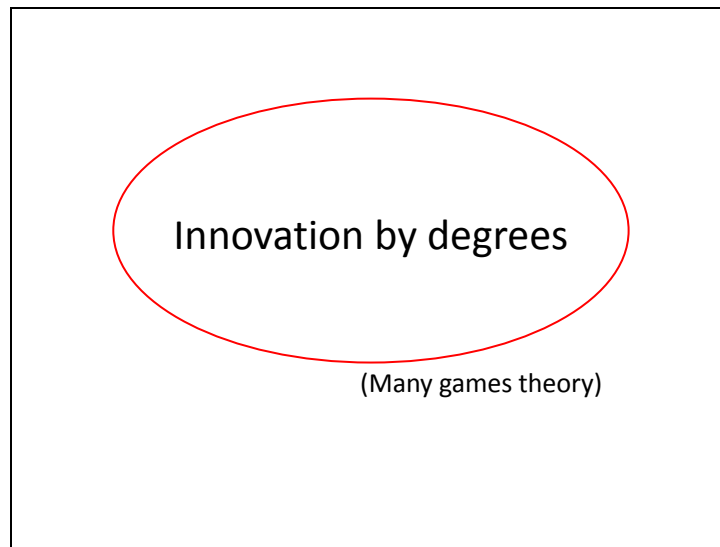
Are we short-sighted?

So a lot of people don't want to make radical design leaps. Thinking it will alienate them from their players. So does that lead to short-sightedness?

Are we short-sighted?

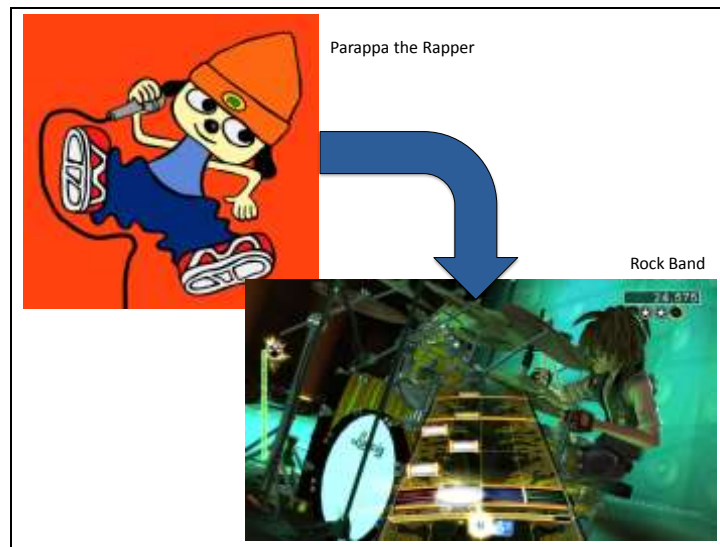
Only sometimes. Mostly impatient.

Sure, a bit. But mostly it's because people forget that they have *\*forever\** to innovate. You don't have to do it all at once. You don't need to feel like you have to change everything now or *\*nothing at all\** - that's short-sighted. And kinda lame.



You have a lot of games in front of you. Between now and some future game, you can make a massive amount of innovation.

Slide 47



Look at Rock Band. I don't think it would have been possible for it to be the success it has been without Parappa (and all the stuff in between), personally. Perhaps you have a different opinion?

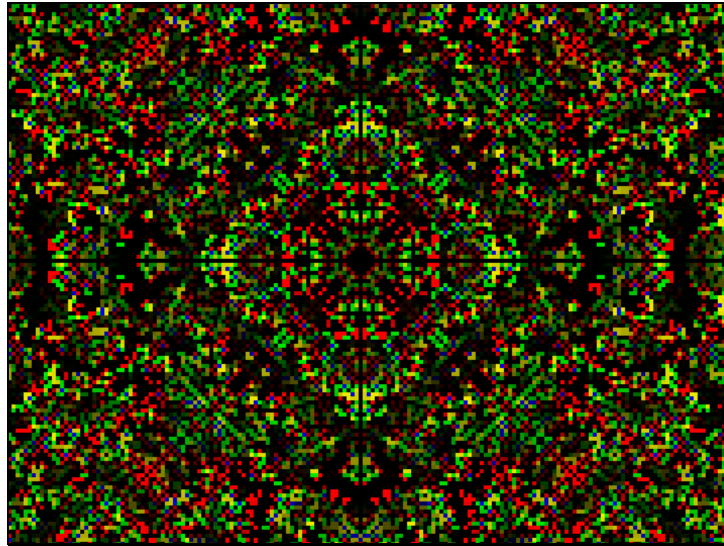




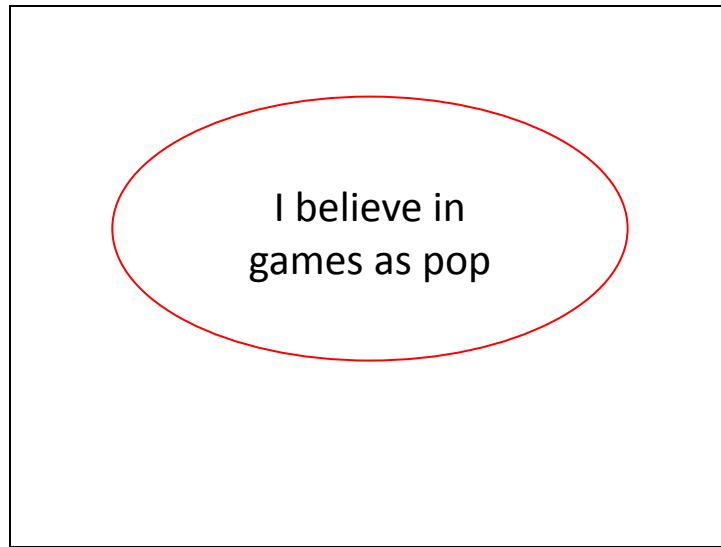
Discover unknown uses.

Another advantage to not trying to do everything at once is that you can learn during the process. It's inevitable that when you put something, even a small thing, in front of some real users that you're going to discover some unknown uses. Machinima is a great example.

Slide 49



It's like Conway's Game of Life. You never know what cool (or crappy) patterns you're going to get until you really just try it.



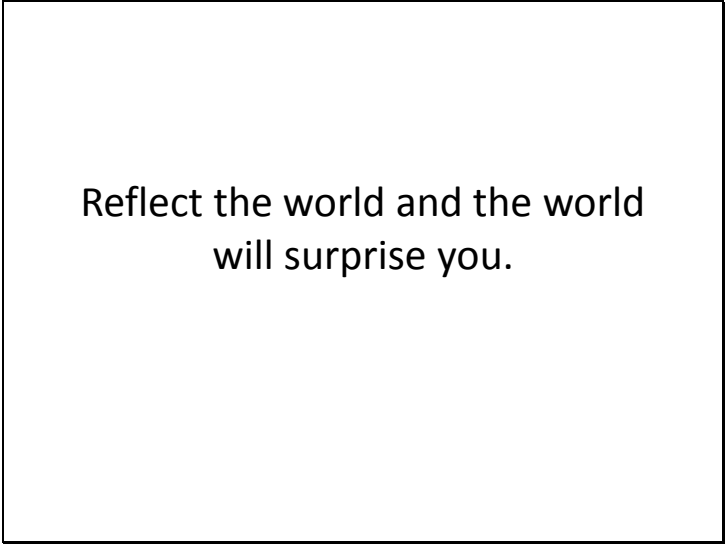
I'm a huge fan of pop entertainment. I believe it plays an important role in society. I believe it serves a role in reflecting us as a culture. It's easy to be dismissive of pop, of course. Especially for fans of things things (music, film, games, whatever) that are perhaps less popular, but in their estimation "better" – But that doesn't diminish the value of pop.

What I'm not saying is that \*all\* games are pop. But largely it is the world that I live in and I love it.

Slide 51

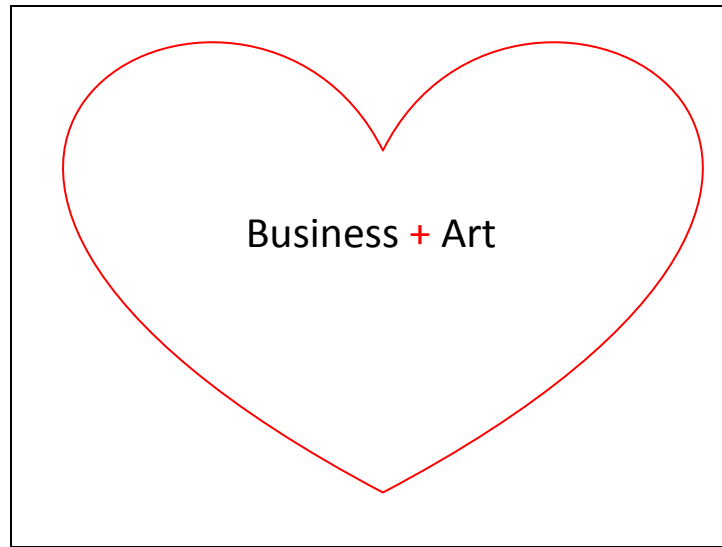


It's like Britney. I'm a huge fan. Or Lada Gaga or Eminem. They are reflecting things about us and tell us a lot about ourselves. Sometimes it's pretty. Sometimes it's really not. But it's always interesting. And it doesn't hurt that you can dance to it.

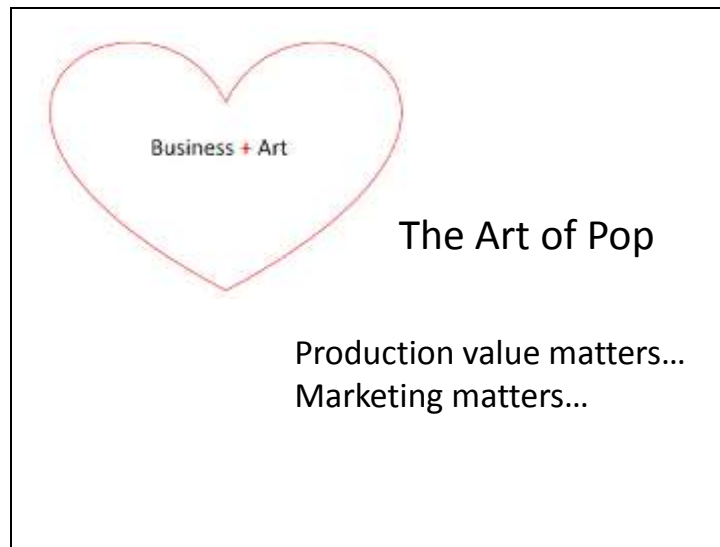


Reflect the world and the world  
will surprise you.

When you try to reflect the world, you will find some amazing things. People are way more interesting than you think. Even teenagers.



Pop is really about the marriage of business and art. It's both those things. And the business itself, of discovering what's important and valuable to an audience and how the world and culture are changing, can itself be an art.



When I talk about pop, the things that help 'sell' it matter for sure. Like pure production value and marketing. But...

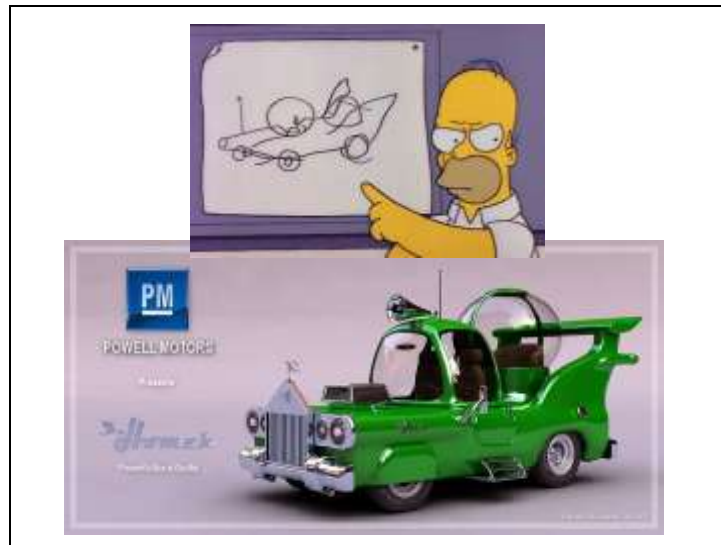


Players Matter.

But the main thing that distinguishes it is that the players matter (or listeners, or watchers, depending on the medium) – You are building something for others in addition to yourself. And understanding their perspective and their role in creation is why it's so very special.

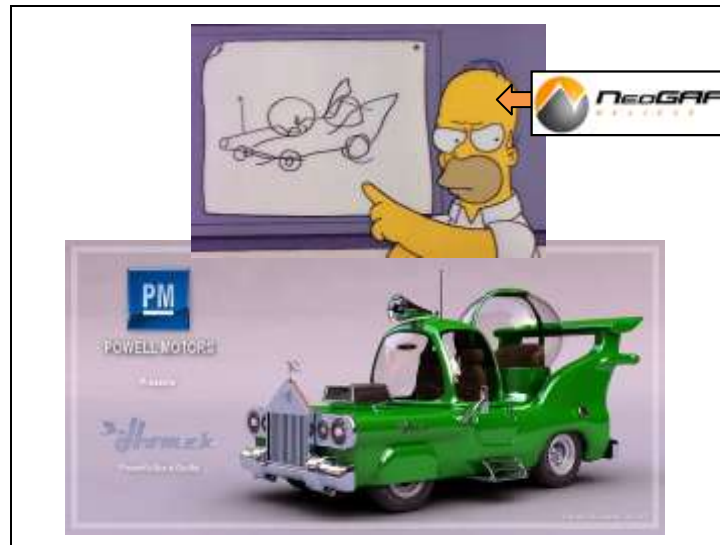


Slide 56

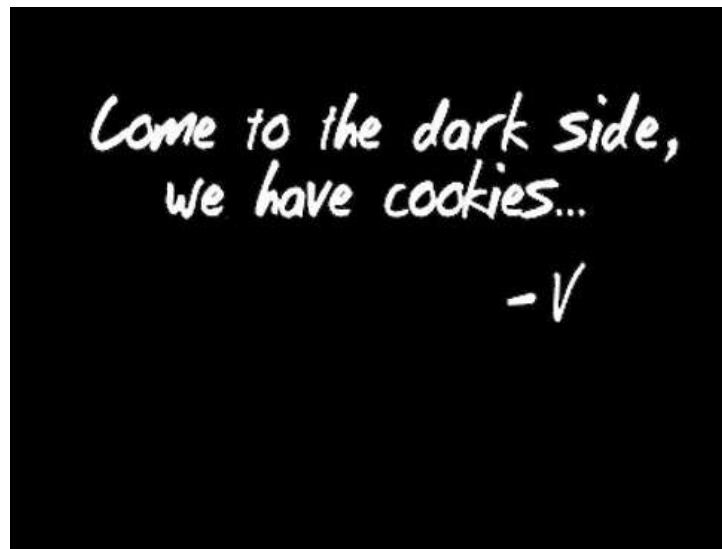


That doesn't mean (obviously) that you just listen to whatever they say and do that. You get craptastic results that way.

Slide 57



And sometimes haters just wanna hate. You have to be careful of that too.



There are definitely dangers lurking in development though...

## Cult of Productivity

It's easy to forget why we're doing this. It's easy to get lost in the day-to-day of tasks, bugs and urgent issues. It's easy to feel like you're making progress just because you're doing \*something\* - Of course the work is necessary, but it's not sufficient.

## Small company vs. big company

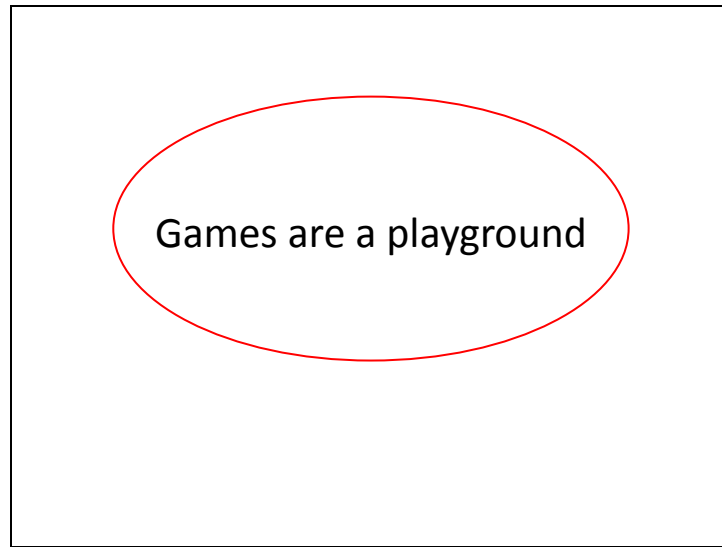
One example is big companies. Big developers. Big publishers. Whatever. It's easy to get caught up in the making and shipping of games each year. And the sales and the marketing. And nonsense on the internet. And metacritic. It's easy to forget what our responsibilities to the art.

We're here because we built on what others did before us. We're absolutely responsible for returning that value to those that follow. How you do that is up to you. It could be gameplay innovation or technical research or visual experimentation.

Some people feel it's "the indies" that need to push the boundaries because that's what their primary motivation is. Sometimes that's true. However, it's the big boys that have both the resources and responsibility to give back the most.

## Individual & Professional

The same holds true for us as individuals and professionals. The more we've taken, the more we need to give back. The more we have the resources as individuals to research and innovate and experiment, the more we have the responsibility to.



Games are also a safe place for a huge variety of research.

## Playground for technology

We do, obviously, get to play with a lot of technology. The Cell processor certainly comes to mind for me personally. It has been a great opportunity to try out new ideas and experiment with concurrent processing, etc. in real practice, under real production conditions.






But there's been a crapload of tech over the years in consoles. You should go check out this guy's image of 222 consoles: <http://greatnscott.deviantart.com/art/222-Console-History-99487643>

From playing with CDs and DVDs and Blurays when they were all still new. To tons of different CPUs and GPUs and approaches. It's always an amazing ride.

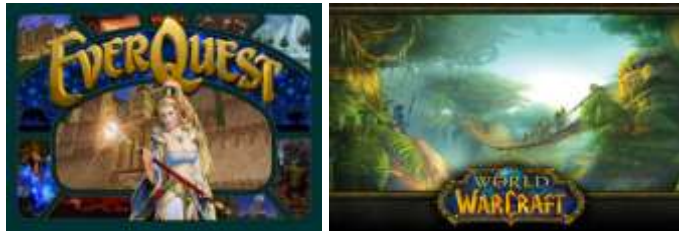


I agree with Alban that these all matter. But I also think a lot of it is because games attract a lot of people who are interested in hard and diverse problems in the first place. I'm sure there are probably other industries that would say the same thing. But for the purposes of this, let's just say I don't care about them. :)



Playground for ideas

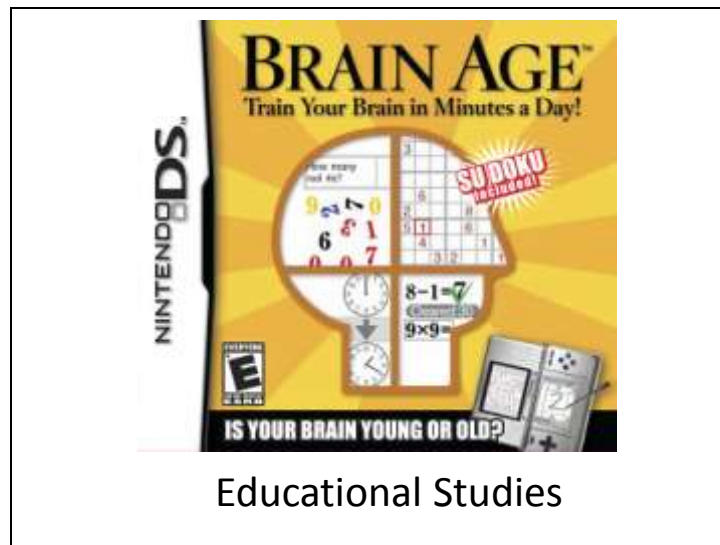
Games are also a place where we can experiment with a lot of different ideas from other areas of study and see how they work.



## Business, Management and Economics

People have learned a lot of really interesting lessons about economies and business behavior from the big MMORPGs like EQ and WoW.

Slide 68



Or experiment with how to engage people's brains a bit more in a fun way.

## Other playgrounds for games

- The Arts
- Human Development
- Cultural Studies
- Labor Studies
- Social Theory, Social Structure and Change

I can hardly think of an area of study that couldn't find an application in games. No matter what you're interested in, there's a way to bring that passion into games.

Slide 70



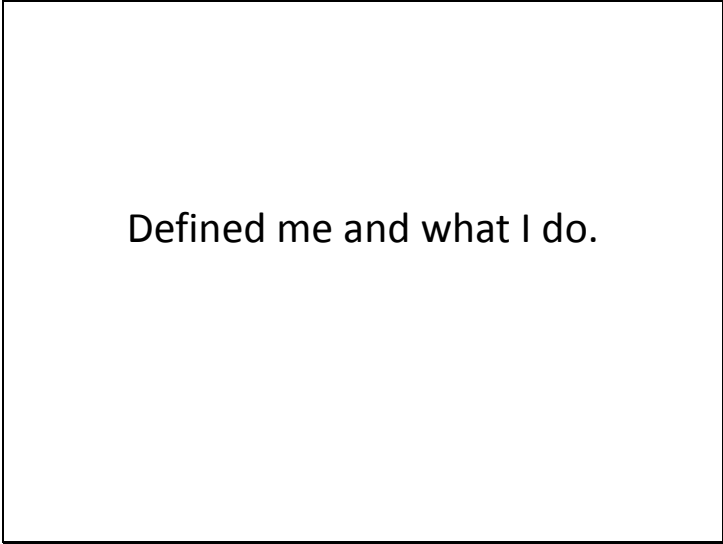
I don't know how we could apply etymology more in games, but I'm sure it's possible. When I saw this, I also thought entymology (which is what I originally thought he wrote.) Both I'm sure can be applied to AI somehow in really interesting ways.

Okay, but...

Why does it matter what I  
believe?

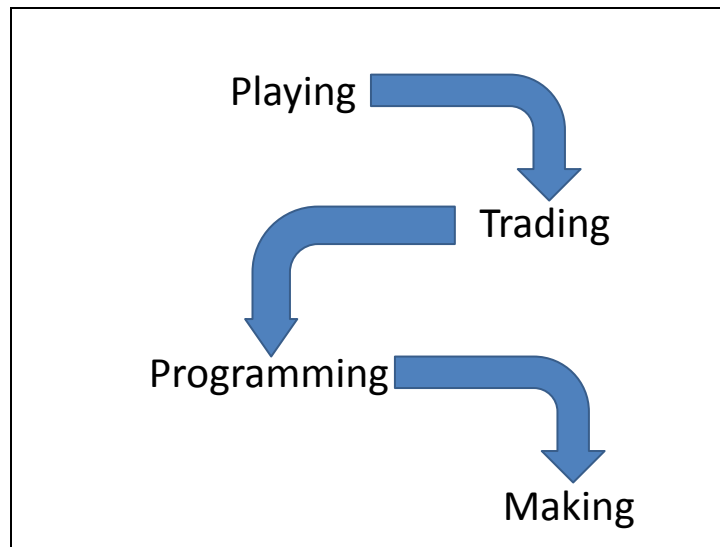
It probably doesn't. But since I'm here...





Defined me and what I do.

Without games I certainly wouldn't be who I am today.



Starting with playing games. Which lead to trading among my friends and becoming a programmer. Ultimately, which brought me back to making games.



Inspired me

But also games, and the industry of development, have inspired me to do the things that I do today.

*"So we need some standardization for data interchange among development tools and products. The more developers that adopt a standard, the better for all of us and our customers."*

-- Jerry Morrison, EA IFF 85

EA's IFF85 document was the first time I had ever seen a game company document an idea and share it for the explicit reason of making things better for everyone. I'm sure it wasn't the first time, but that's not important. It was the first time for me.

Slide 76



And for me personally, I can point to that one document and say that it is absolutely because of how that inspired me that I do some of the things I do today. Like [cellperformance.com](http://cellperformance.com) or our Insomniac R&D pages and Nocturnal Initiative. Or other presentations (like this one!)

*"Simple things should be simple and complex  
things should be possible."*

-- Alan Kay

Interestingly, that document was also the first place I had ever seen this quote by Alan Kay.  
Which I certainly still refer to today.

Color all of my choices  
*every single day*

All of these things make a difference to me. Are important to me. And have profoundly affected who I am and what I do every day.



## Why do I do this?

- Games are important
- Games make a difference
- ...as art
- ...as pop
- ...as a playground
- Games inspire





## What gets in the way?



Haters



Our impatience

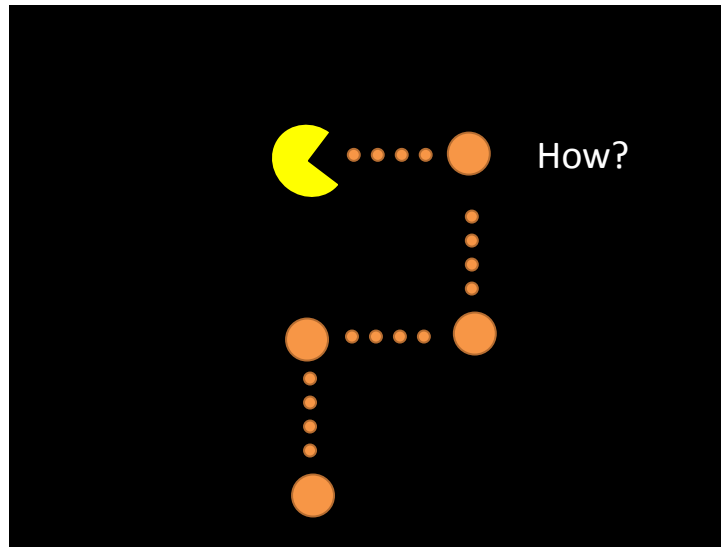


Cult of productivity

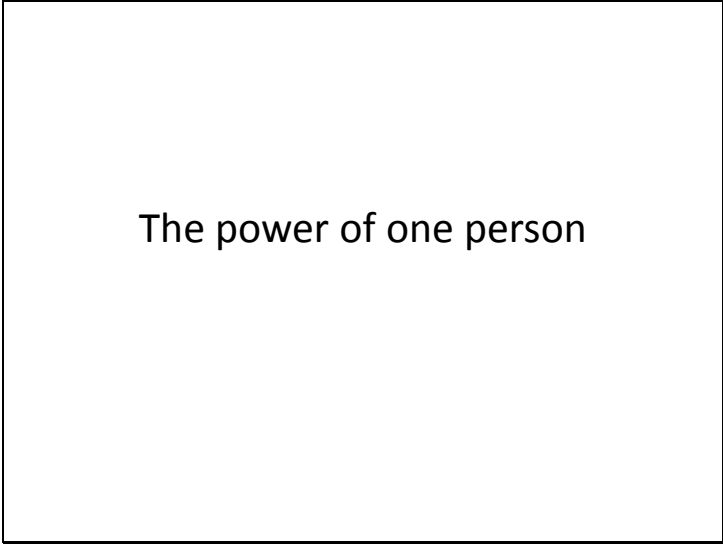


Forgetting why we're here

Slide 81



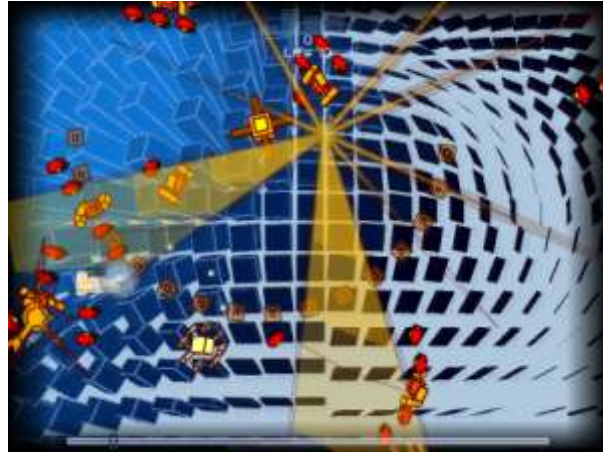
How do we use that? How do we make a difference?



The power of one person


Everyone has the power to make a difference.

Slide 83



Everyday Shooter, Jonathan Mak

Some people use that to show the world what they can do all by themselves. And that definitely has value. PS: You should play this game. I'm a fan.



Impact on a team

But for most of us, it's about how we can make an impact as part of a bigger (and for some, a quite overwhelmingly large) team.

Our job is what we make it.



This is one of the most important lessons I can share. It's not always easy. But you can't get caught up in only looking at what's put immediately in front of you. It's each of our own responsibility to \*figure out\* how we can make the biggest difference in whatever we are doing. And there is always room for that.

Prediction:  
50% - 5 years  
100% - 10 years

Here's what I predict: Write your job description down. What are the specific problems that you work on day-to-day? What are the specific challenges you face?

In 5 years, for 50% of you that job, as described, will no longer exist.

In 10 years, for 100% of you that job, as described, will no longer exist.

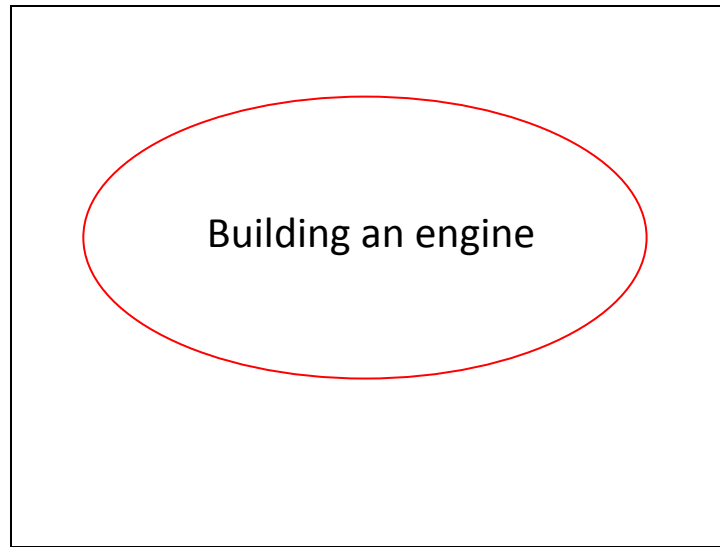
Your job will only be superficially similar, if it's similar at all. The world changes fast. We all need to keep up and change with it.



Growth (2x)

I have this goal. For me personally and for my team. I want us all to grow to be twice as good at our jobs each year as we were the year before. What that *\*means\** for each person may be different. But the sentiment is the same. If you don't grow by leaps *\*constantly\** you will fall behind. And growing and learning is what we exist for. It's what excites us most as people.





I should actually probably talk a little bit about building a Game Engine. How do I bring all of these ideas to my own work?

## Vehicle for passion



I see a Game Engine as a vehicle for passion. For the passions of those that make the engine, for those that use it directly to create a game, and indirectly for the players as well.



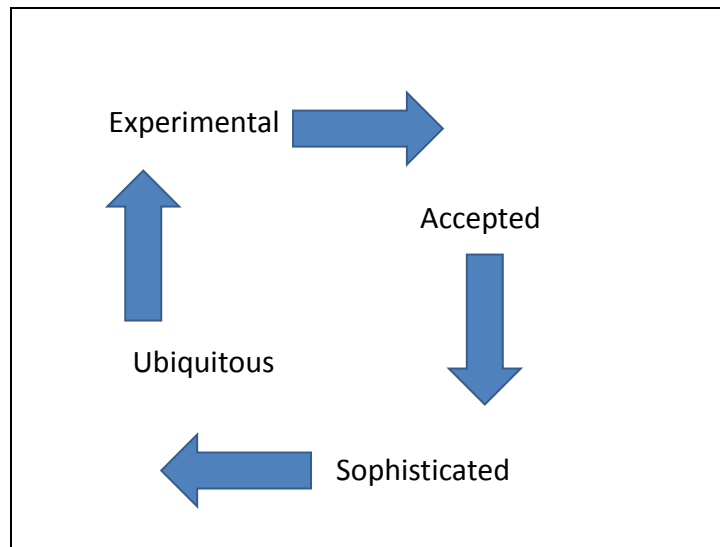
Reflect *developers'* vision

The purpose of an engine is to reflect and enable the developer's vision. Provide the necessary functionality to accomplish what they want to accomplish. Of course, the practice of it is a lot more complex. It should also reflect the vision of the developers of the engine as well. The challenge is to find a way to do both, or balance them. And resolve the practical issues of development (like resource allocation, performance, development costs, etc.)



Technology-driven industry cycle

I think all technology driven industries follow a similar kind of cycle...



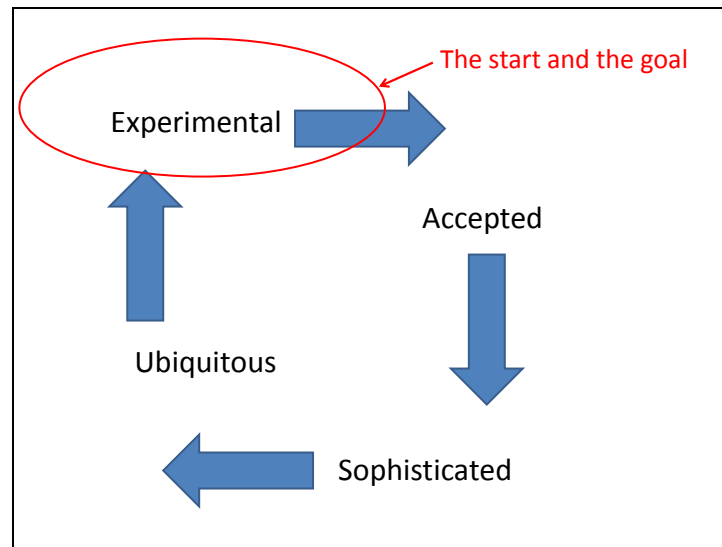
- They start off experimental. Some hacker throws a bunch of stuff together and comes up with something new and cool. Shows it off. People get excited.
- At some point it's not experimental anymore. It's accepted and an industry starts to build up around the idea.
- Over time, the process becomes much more rich and sophisticated. Brand new specialties arise. The complexity of building a project becomes enormous by comparison with the early days. Many, many more people involved.
- However, at some point tools evolve. Work that required a lot of custom work by specialists becomes commodity. Straightforward enough for anyone to do to a useful degree. When those tools evolve far enough, the process becomes much more experimental again. Where many fewer people can make a significant product.

e.g. Movies. Lots of tools now for experimenters. People can pull off very sophisticated effects, compositing, editing, filming, etc. at home and post on to youtube. We've seen a lot of cool stuff.

Games will see the same development. Already started, actually.

While there's always room for the big, sophisticated productions, the amount that can be accomplished by a single person with the right tools is growing rapidly.





The goal is to get back to experimental. And then build on that to solve \*brand new\* problems.

## The Programmer Bias

What gets in our way though? Our biases.

Programmer bias is strong in our industry. Which is the idea that only a programmer can do something. That a task is so sophisticated that it must be programmed by an expert, or it simply cannot be done.

Which, of course, is bunk.

That may be true today for many things. But it gets in our way when we believe that it *\*must\** be true.



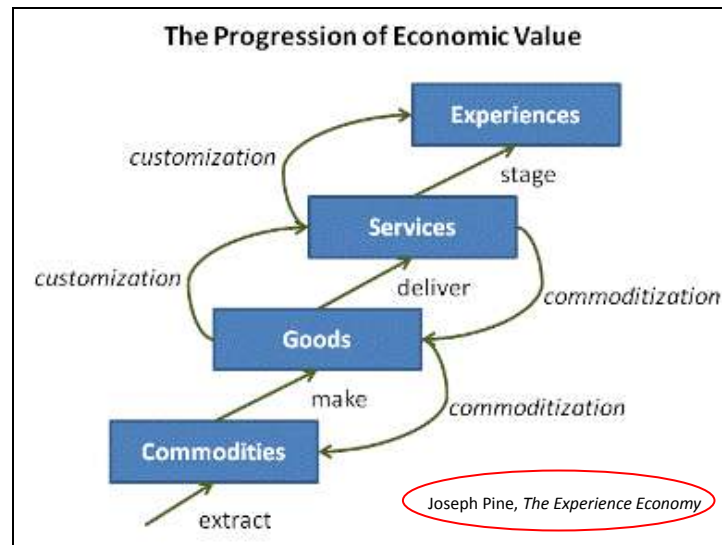
## Negativity Bias

Also, we do have a tendency to be over-critical and negative. We don't always let ideas play out first. We don't assume first that it *\*can\** be done.

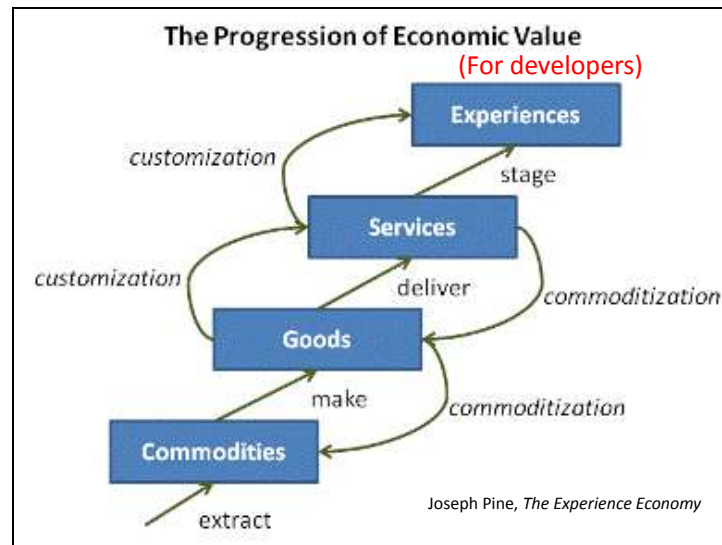
We are cynics by nature.

But we must not forget that it's optimism, a totally unfounded belief that something new can be done, that changes things.

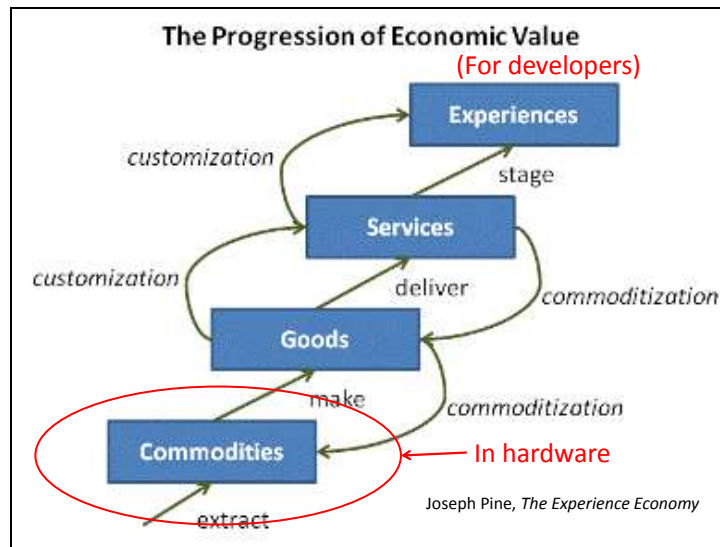
PS: Since I gave this presentation, I wrote a bit more detail on my own biases (including these two) here: <http://macton.posterous.com/personal-biases>



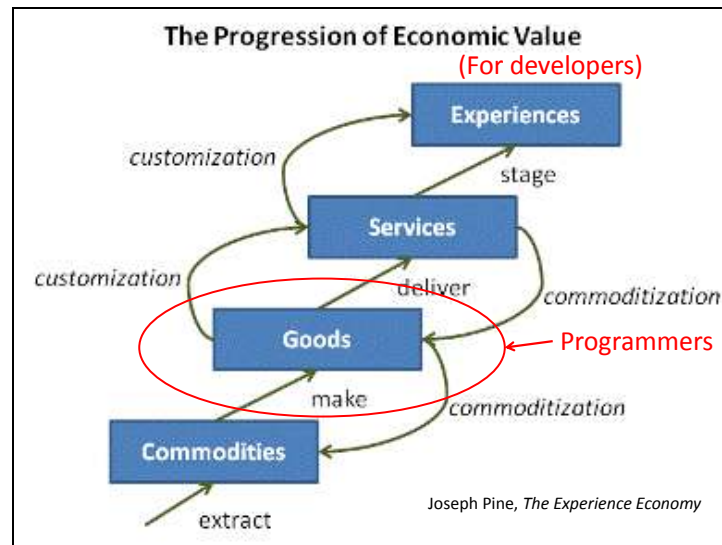
Joseph Pine wrote about the Progression of Economic Value, which also applies to Game Engine development.



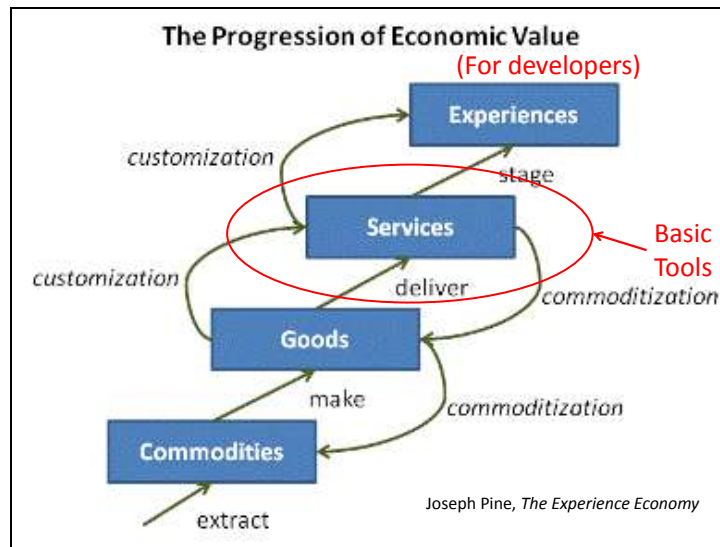
A game engine is an economic engine for developers. It's built on the time and energy resources of the team making the game and transforms that into the value for the player and the vision of the production team.



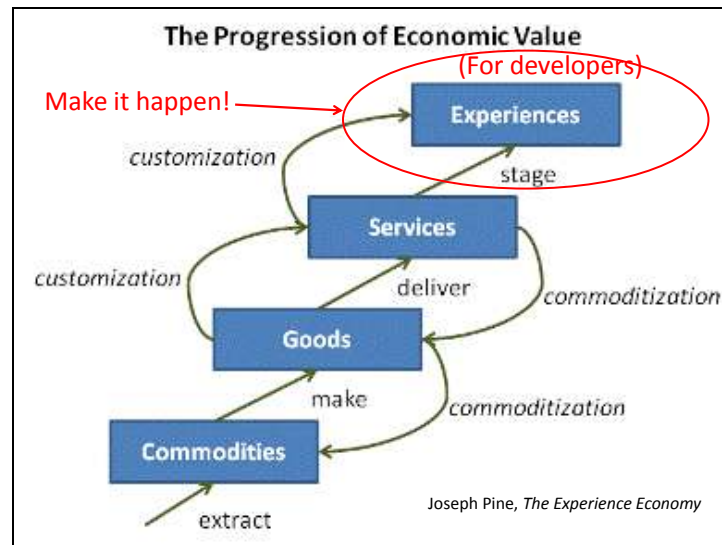
We could build custom hardware for every video game. That's the basic commodity. (And certainly, that's how things started.)



But as the hardware itself is commoditized, programmers customize it to make the games.

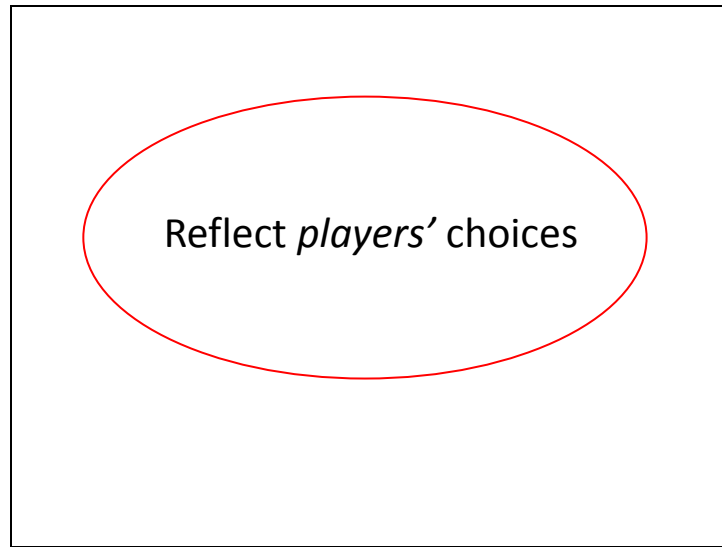


But then, as what the programmers are doing is commoditized, basic tools come on line which can be used to customize the game.



Even further, as that custom work of the tools is commoditized, fewer people can concentrate on building more custom experiences (or individual ones, even) by leveraging the work before.

Obviously this doesn't all happen together. Some parts of game development are very far along in this progression. But clearly, many parts are lagging if the average development cost is any indication.



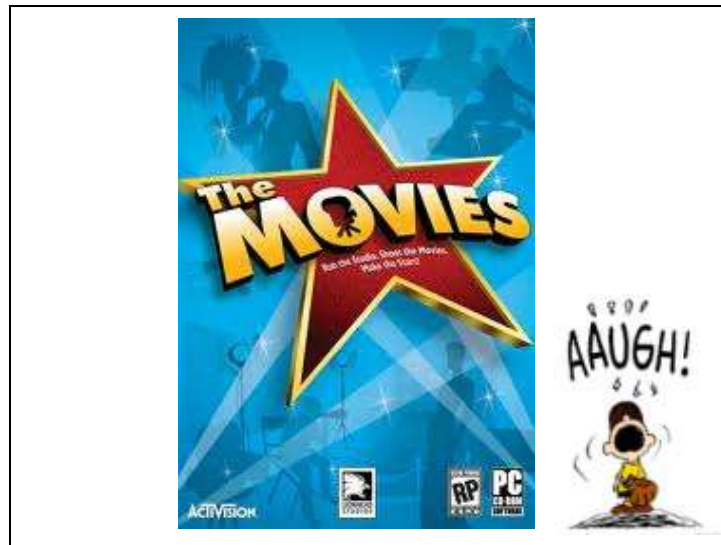
Games, fundamentally, should reflect player's choices. Engines need to enable that.



Realtime = Reflect player choice



Video games are about realtime interaction. The reason to do that is to reflect what the player does. It's the main value of the medium!



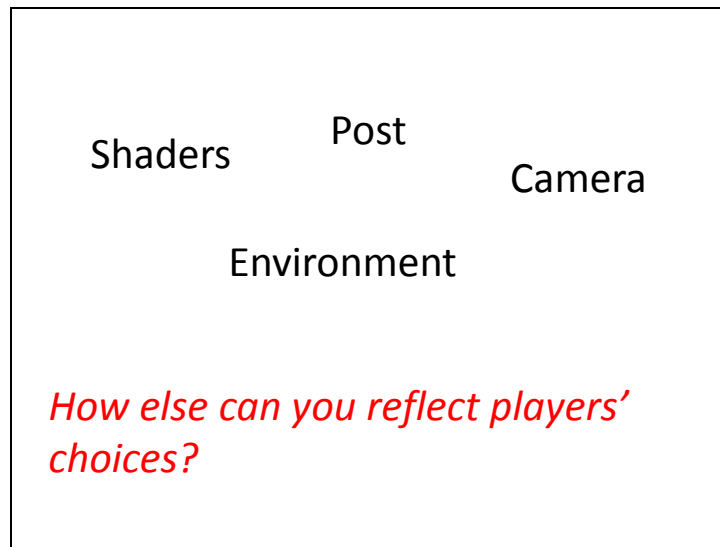
Aside: It really pisses me off when I meet game developers who really (secretly or not) just want to make movies. Go make movies then. Games are a different medium. While they can share things (like having a story) they are fundamentally different.

Video games are NOT an evolution of movies or TV!

Video games are an evolution of other games. People have played games as long as there have been people. Whatever technology existed, people figured out a way to play games with it. The fact that now we play them on the TV or on the computer, or iPad or whatever doesn't change that fact. Games are games.

Movies and TV are about storytelling. That's totally fine.

But while a game *\*can\** tell a story, if that's all it does, it's not a game.



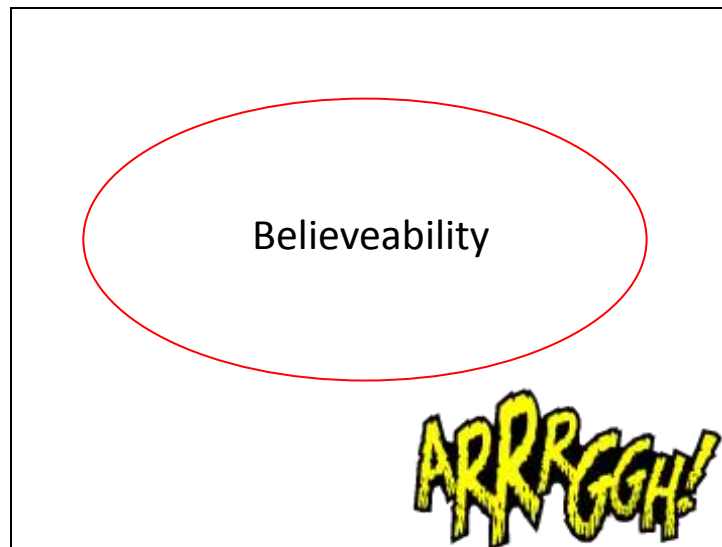
We still have a long way to go to figure all the ways we can reflect the players' choices in games.

We still largely model after a static world that *\*cannot\** reflect the player.

e.g. We model a static building, because in real life, it must be static. So that's familiar and we understand it.

But that building was made with some function in mind. For example, it might have a focal point (stage) and architecturally it's designed to draw your eye there. But in games we can potentially change that focal point based on the choices the player has made and affect the whole architecture.

Lot's of things left to experiment with. Endless!



First off, I don't like argument for "realism" in games. Realism is a fool's errand. And closes a lot of interesting possibilities to boot.

Believability, however, is crucial.



Tobal No. 1 was a great example. Not realistic at all. But the developers did a great job of making it \*believable\* - in particular, the grappling and animations matched the style of the art and the world. Very few titles have done as good a job since at that, IMO.

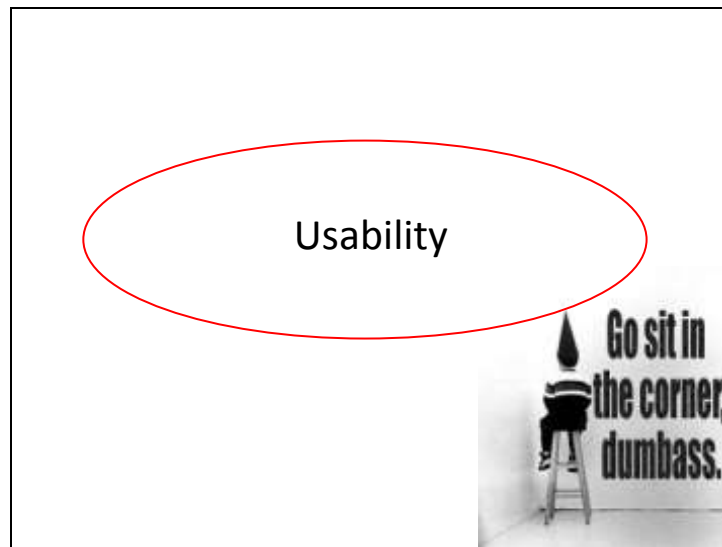


And you can't talk about believability without talking about the Uncanny Valley. Basically, we're close to the bottom right now (for the kind of big budgety adventure or FPS type stuff.) Everything is just a little bit creepy and unnatural. And it's actually getting worse still. Most notably as character models and shaders improve.

An engine needs to be balanced. Also have to focus on the things that will bring you out of this valley and make the final work more believable. Animation is an example, for sure. But it's not the only thing.



It's not \*just\* about the high-res human games, either. Ms Pacman is believable. I accept her in her little world with these ghosts and fruits. Of course, it's largely because it's more abstract art that it's easier to accomplish that. But it's still a consideration in any title.

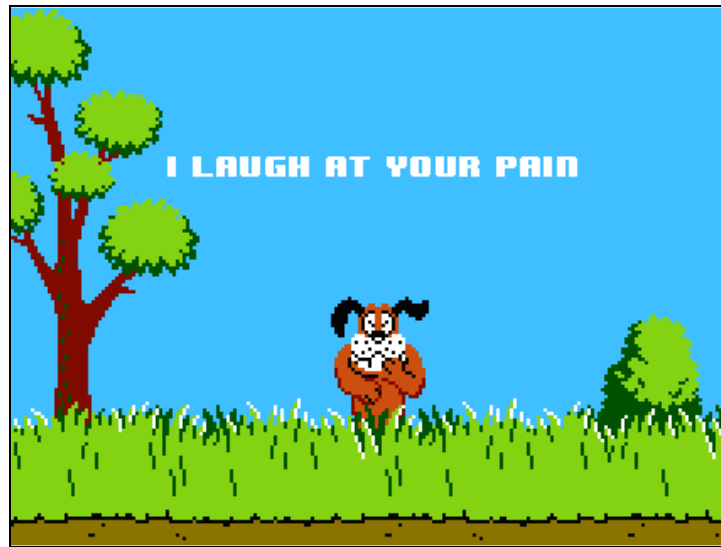


This is a huge (and obvious) lesson. If people can't use an engine effectively, it doesn't matter how good it is under there, no one will ever see it.

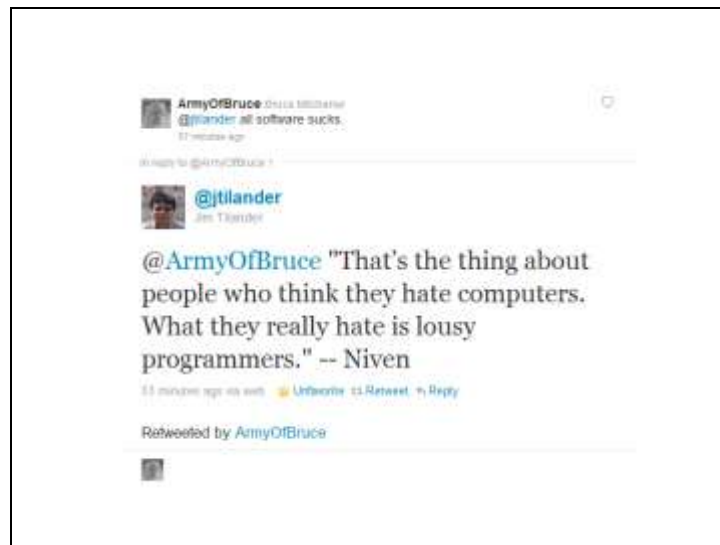
I wrote a bit about my thoughts my personal approach/thoughts on usability. A presentation which I have yet to finish, but the basics are there: <http://macton.posterous.com/unfinished-usability-is-not-random>



Slide 111



It's easy to blame the users. Don't.

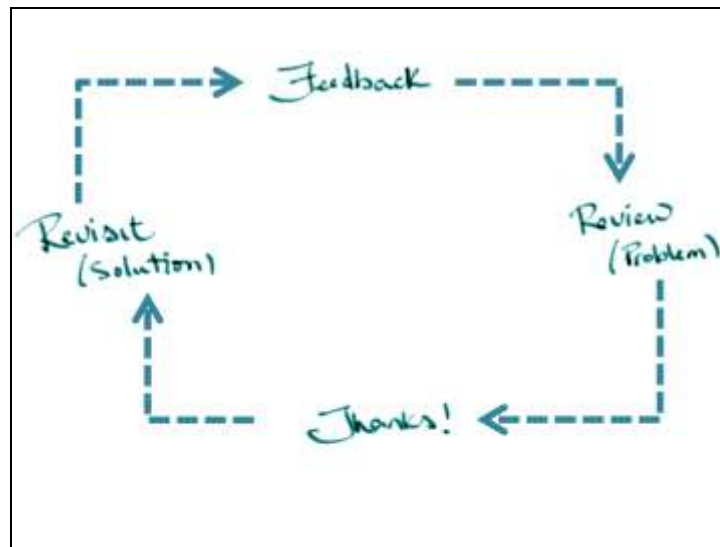


## Usability cart stories

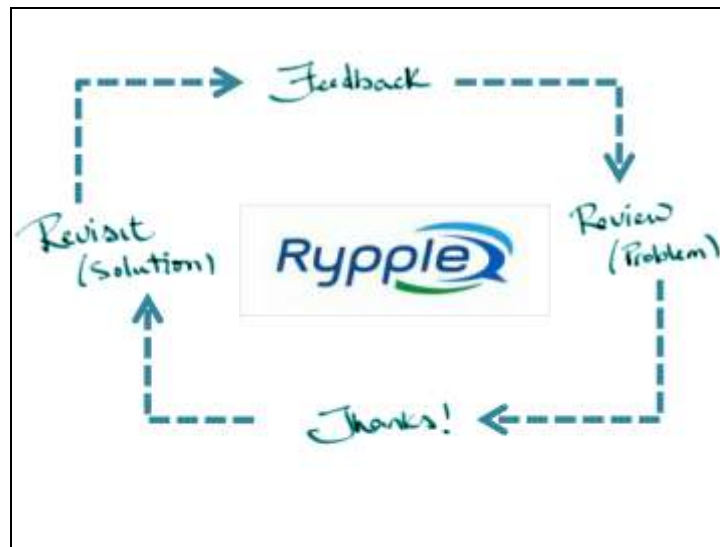
One thing that we've done that's been pretty effective is "usability cart stories" – we have a machine on a cart which we drag over to different people's desks. We let them work with the engine for a couple of hours or whatever and then write a story about what we learned from watching them and share that with the team. Inevitably there's a lot to learn. No matter how many times you do it.



Good feedback is critical to any project.



Personally, while I feel that I've always asked for feedback and criticism, I've not been awesome about the rest of the feedback cycle. In particular with making sure people know that I appreciate their feedback and see what I've done based on it (revisited with them.)



I thought I'd mention that I've been using Rypple internally for gathering feedback and collecting notes.



## How do I make a difference?

- As an individual
- As part of a team
- Building an Engine
- ...reflects developers' passions
- ...reflects players' choices
- ...through believability, usability
- ...and feedback



## What gets in the way?



Not making the most of our jobs



Programmer and negativity biases

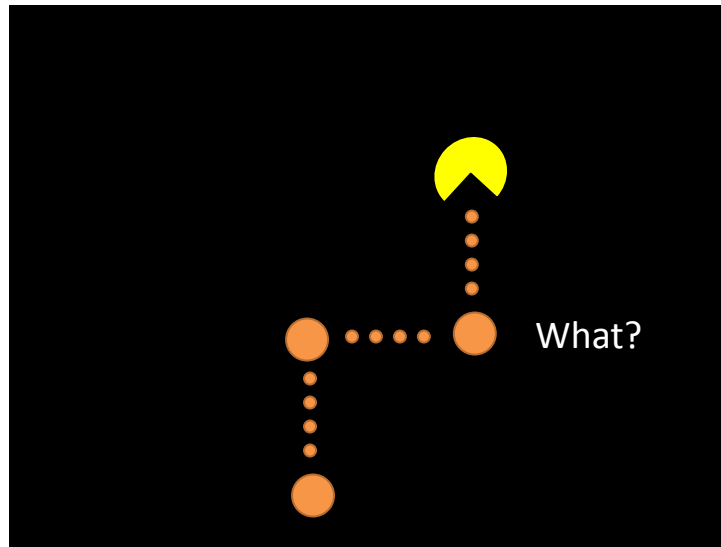


Not focusing on usability

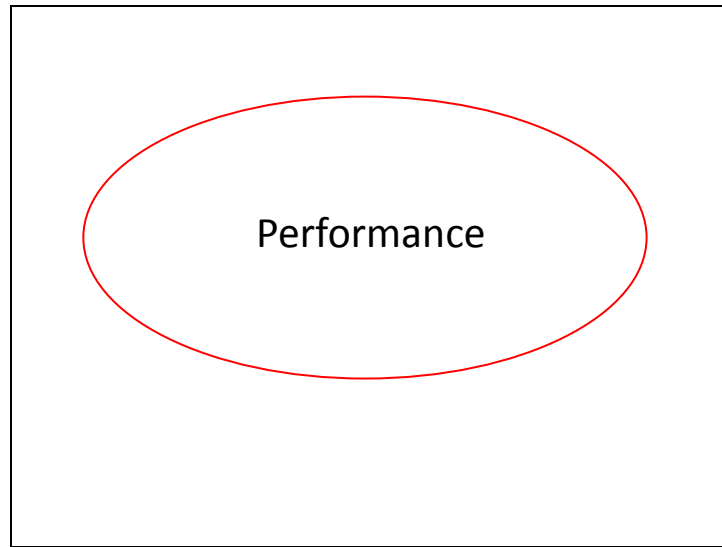


Neglecting the feedback cycle





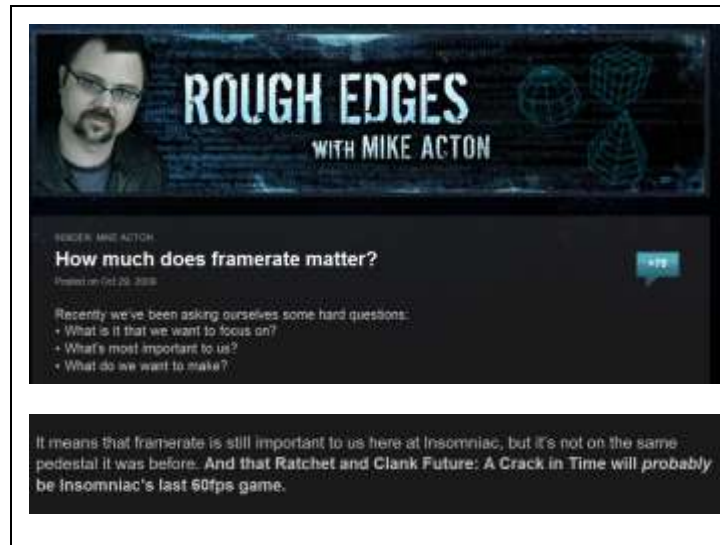
What do I focus on in practice? What do I enjoy? What's important to me?



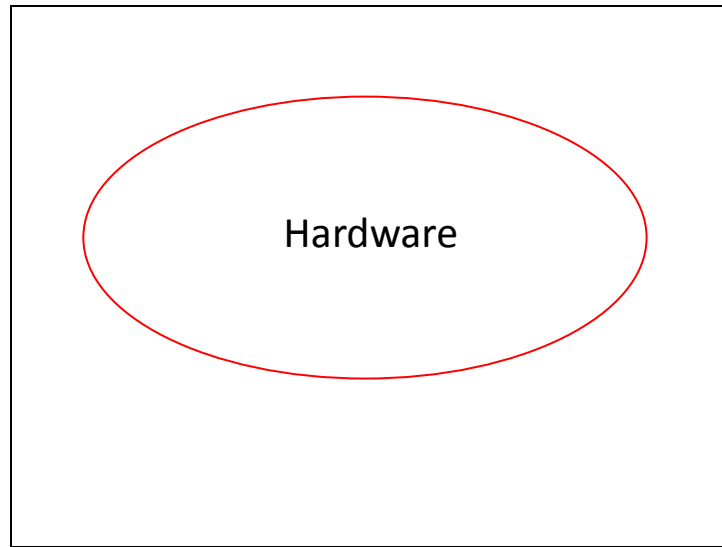
I care about performance. Performance is what enables us to do more. Performance is the currency for creative freedom.

“Never underestimate the power  
of speed.”

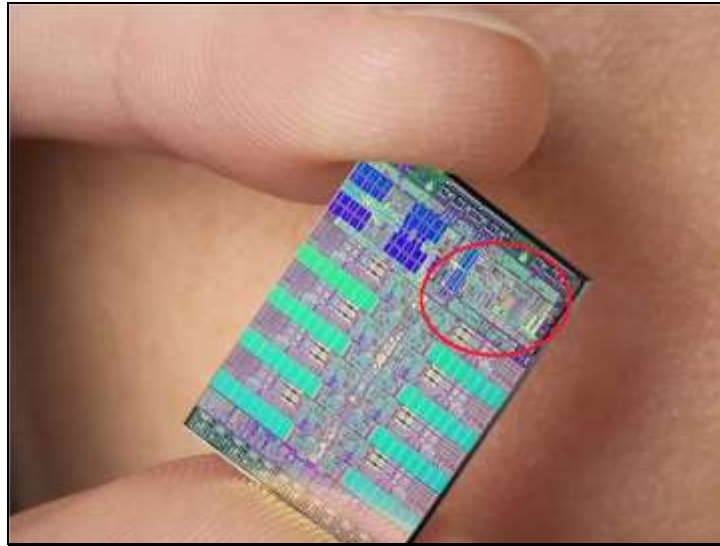
-- Eric Schmidt, CEO of Google



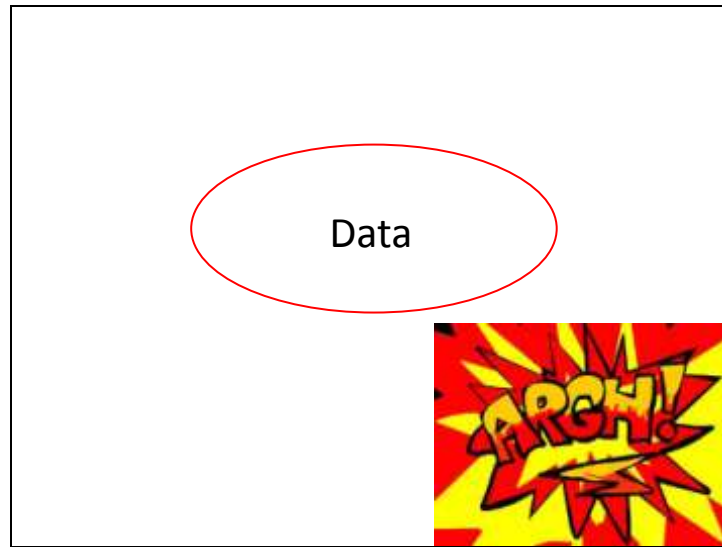
I know someone's going to bring this up. Performance doesn't just mean a faster framerate. It means how you choose to use the performance you have. The only reason this question (60 fps vs. 30 fps) is relevant is because we have had enough performance to make the choice. To many devs, unfortunately, it's not even a choice. There's not enough real performance to even consider an alternative to whatever they have.



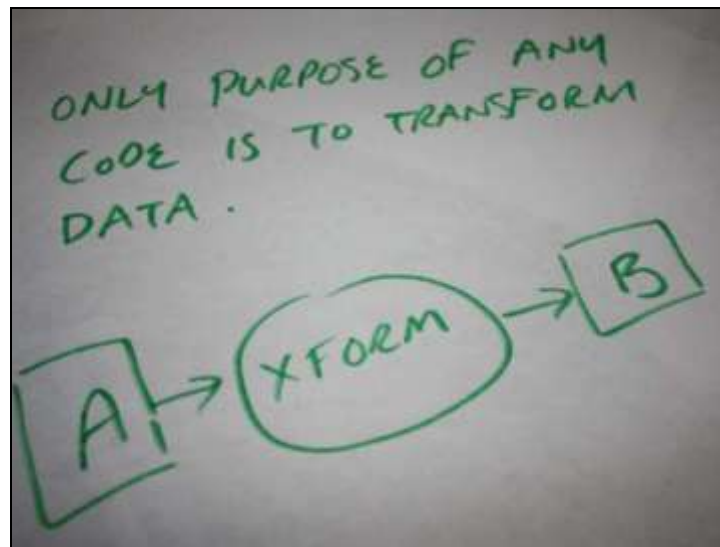
I care about hardware. Software doesn't exist in a vacuum. There is no "abstract software ether." Decisions have to be made based on the real platforms that they're running on. I would hardly make the same choices for software intended to run across all of Google's servers as I would for software to run on the Gameboy. The whole idea that there is a software solution to any problem \*independent\* of the hardware it runs on is pure fiction.



Most of the time even an extremely superficial understanding of the hardware is enough to make better decisions. Just looking at the cell processor, for example, the PPU there is about  $1/10^{\text{th}}$  of the chip. Do you really want all your code on that?



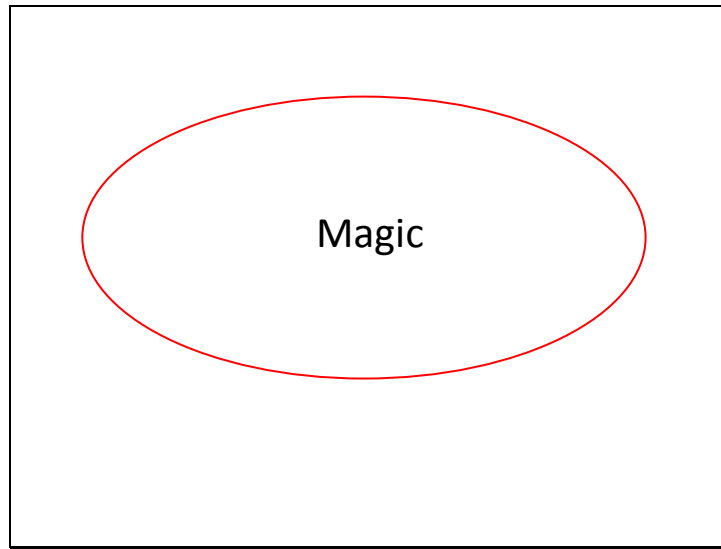
I care about data. It's the only thing that matters. There's just too much code architecture wankery in the world. And not only are the benefits truly dubious, they often make things much, much worse than is necessary. For performance. For maintenance. For memory. For whatever.



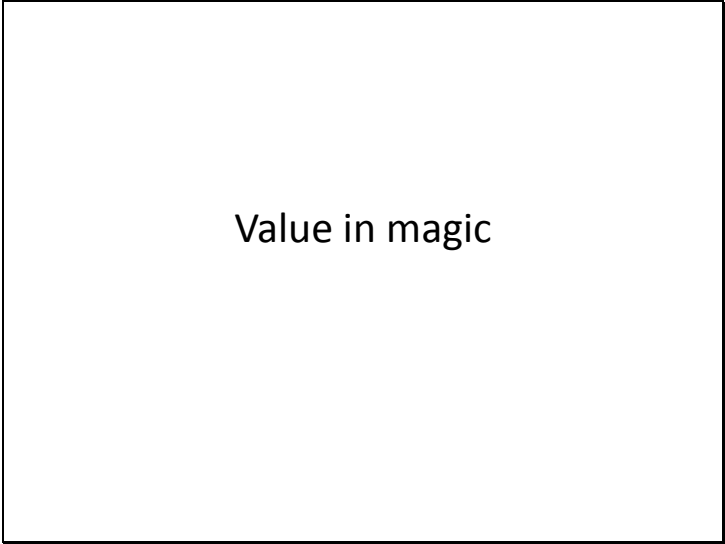
The purpose of any program is to transform data from one form to another. That's all. Nothing else.



Slide 127



I care about Magic.



Value in magic

I *\*believe\** in magic. There's real value in creating something surprising. Or something wonderful that appears to come from no where or nothing.

The magic of  
the unexpected


Slide 130



This is a screenshot of a character hierarchy in Maya. When we saw this, we were very surprised at its complexity. In fact, it's much, much more complex than this since all the math nodes are actually hidden.

But we wanted to see what we could learn from this surprise. We asked the riggers to present to us their process. Of course we learned a ton. But in addition, it really became obvious that they had done something, using very poor tools actually, that could be applied to a lot of different areas. Not just animation.

Which led to some whole new conversations.



## The magic of play

You have to play. Try stuff out. See how it works. Learn what you can from it.

You can't just be locked into the way it is now. Or you'll be obsolete very soon.

So what are some of the things we've played with recently...?



We've looked at webapps a lot for inspiration for our tools. In fact, we've been doing a lot more in Flash and Javascript if you can believe it!

I figure this whole web thing probably isn't a fad. There might be something to it.




We've also looked to the social media sites for inspiration for our own tools and processes.



And, somewhat ironically, the thing we hadn't looked at enough before: We're looking at games. We're looking at how we can use the lessons we've learned from games to make making games better.





## The magic of failure

It's also important to fail.

It's important to let people fail.  
You must embrace it.

If you're afraid of failing, you are paralyzed from doing anything actually interesting.

But of course that does mean that you *\*will\** fail...



Sometimes spectacularly.

Spawn: The Eternal.

I was on this team. This was not an awesome game. I believe involved on this game would say it was a failure. An awful, awful failure.

The first (and only) time I've ever seen the phrase "explosive diarrhea" used in a game review.

But everyone on this team learned a lot from the experience. And we've all gone on to do much better and take those lessons with us.



You have to \*want\* to fail. The trick is to do it as quickly (and cheaply) as possible. Then figure out what you learned and try again.

*“If you don't fail at least 90 percent of the time, you're not aiming high enough.”*

-- Alan Kay



What am I going to focus on?

- Performance
- Hardware
- Data
- The unexpected
- Play



## What gets in the way?



Unoptimizable systems



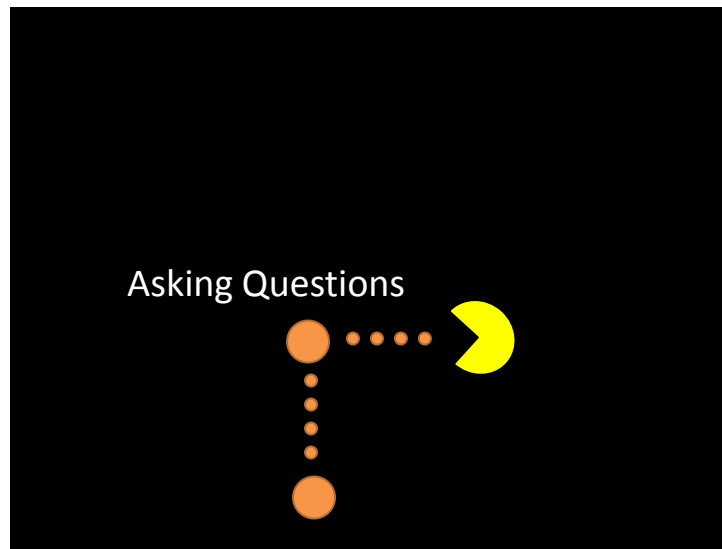
Not understanding the platform



Not understanding the data



Fear of failure



So these are the questions that I asked myself at the start of this...

Why am I here?



What am I most passionate  
about?

What drives me?

What's important to me?

What inspires me?

How do I make a difference?

How can I make a bigger impact?

How can I reflect the world?

How can I reflect the player?



How can I reflect our passions?

How can I overcome my biases?

How can I use feedback better?

Where can I make all new  
mistakes?

What am I going to focus on?

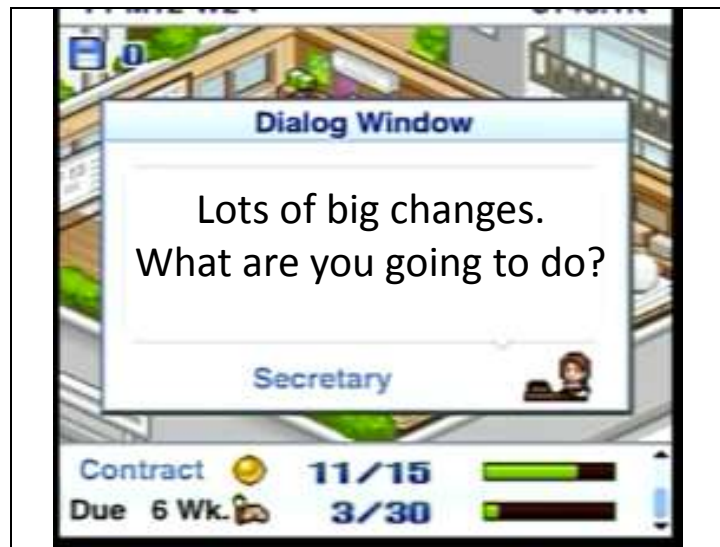
What else is out there?

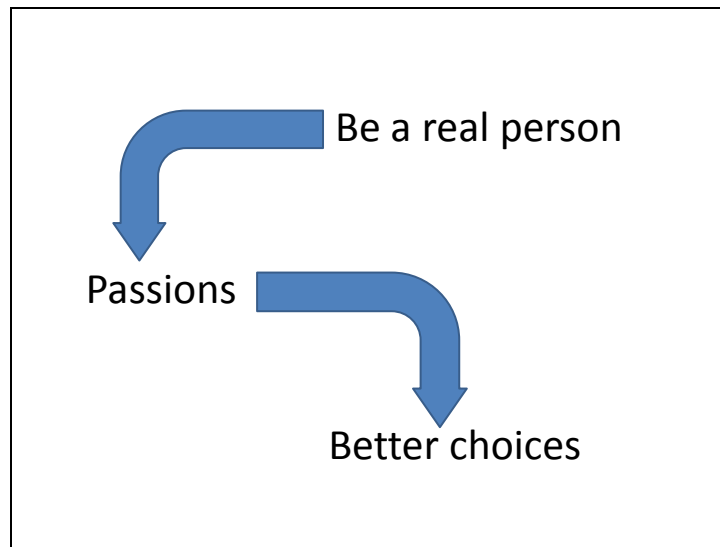
How can I use the unexpected?

Is this fun?



What's the data look like?



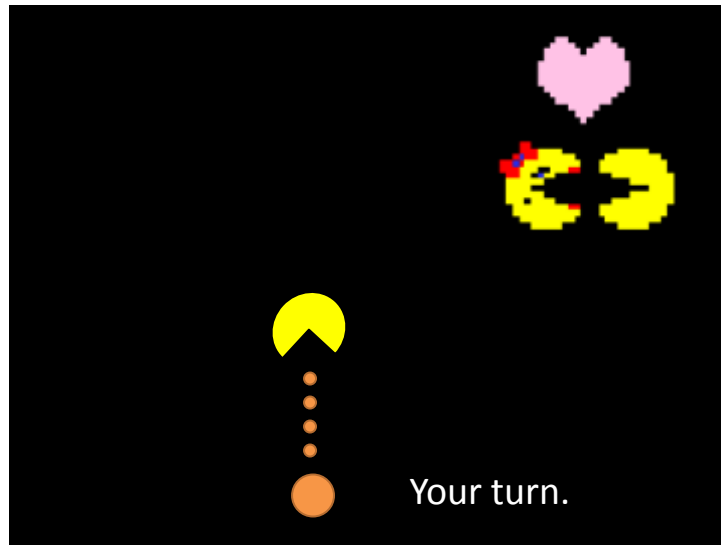


I try to use my passions to drive my choices.

Slide 162



When I do that right, I'm on fire!



How would you answer these questions? How does it affect what you do?

Twitter: @mike\_acton

Email: macton@gmail.com