

Rules of the Game

Another Five Techniques from Particularly Crafty Designers

Your host

Richard Rouse III @richardrouseiii

with

Hal Barwood www.finitearts.com

Chelsea Howe @manojalpa

Luke Muscat @pgmuscat

Christina Norman @truffle

Damion Schubert @ZenOfDesign



GAME DEVELOPERS CONFERENCE* | FEB 27-MAR 3, 2017 | EXPO: MAR 1-3, 2017 #GDC17

Welcome!

In this session we take five illustrious speakers who you see listed behind me

And then we give them 10 minutes to talk about one of their personal rules about game design.

Rules of the Game 2015

Laralyn McWilliams
@laralyn

“Make Emotional Connections”

Chris Avellone
@ChrisAvellone

“Look for the Silver Lining”

Dan Teasdale
@deliciousbees

“Chunk in Threes”

Kim McAuliffe
@EnameledKoi

“Fight for the Little Things”

Nels Anderson
@nelsormensch

“Don’t Try to Evaluate Your Own Game*”
* But Only You Know What It Should Be

Search “Five Rules” on GDCVault.com

GAME DEVELOPERS CONFERENCE® 2015

MARCH 2-6, 2015 GDCONF.COM



And we’ve done this a few times before. Back in 2015, we had these amazing designers talk

This one is up for free on the vault.

Rules of the Game 2016

Lee Perry “Pizzazz First, Polish Later”
@MrLeePerry

Emily Short “Visualize Early”
@emshort

George Fan “Make Your Enemies Actually Different!”
@TheGeorgeFan

Liz England “Make Actionable Documentation”
@lizardengland

Michael de Plater “Make Your Game Tell Real-World Stories”

Search “Five Rules” on GDCVault.com



GAME DEVELOPERS CONFERENCE March 14-18, 2016 · Expo: March 16-18, 2016 #GDC16

This one is up on the vault but not, as of yet, free to view. But the slides are there! And hopefully it will be free too... soon...

Why Rules?

So why do we care about rules?

Well as game designers our job is making rules.



We like games, after all, that's why we are here. Many of us like the knowableness of rules in a game – they make it work – they make it fair.

Like in the game of Bridge, we like the fixedness of who goes in what order, of how you bid, and how you win contracts.

And it's always clear cut who wins or who loses. Because of those rules.

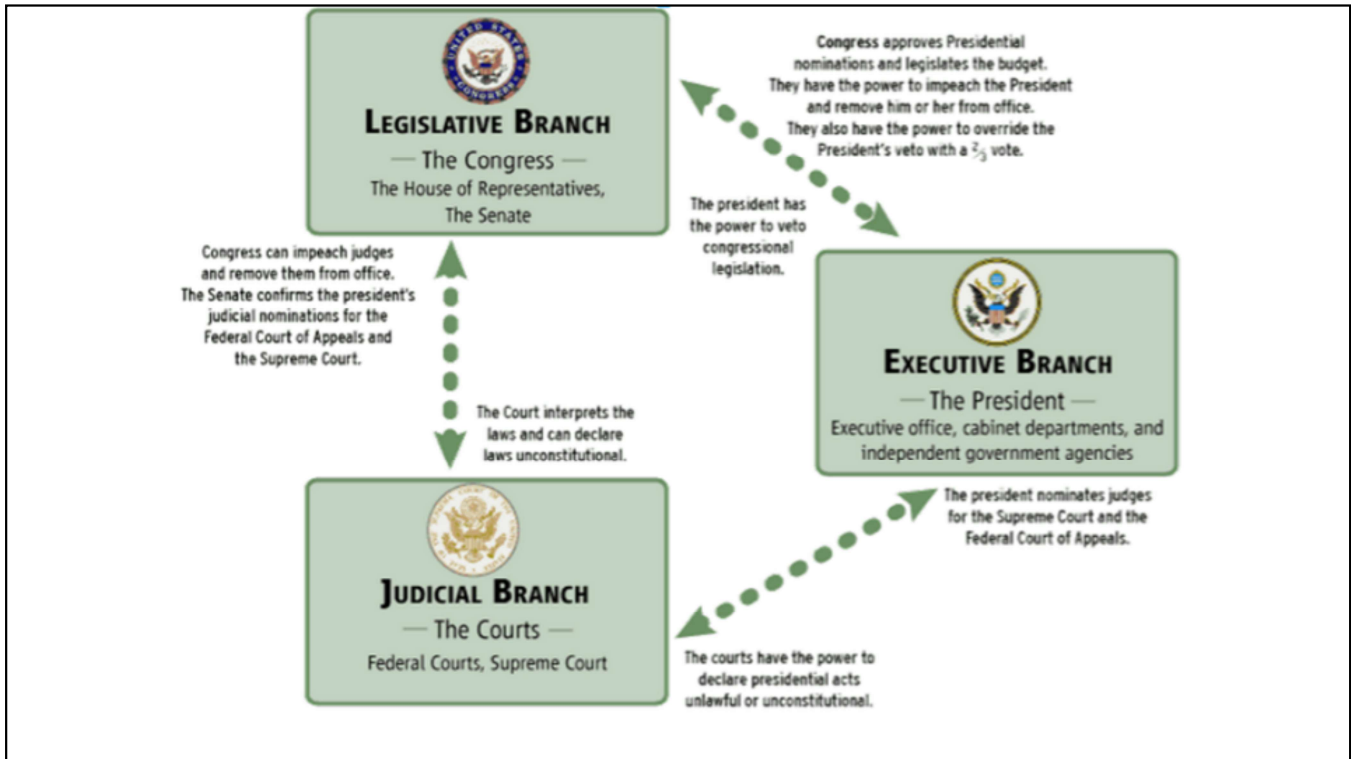


Even in a more complex game, like Diplomacy, the rules becomes harder to understand, and many house rules may exist.

But before playing , you agree on what the rules are and so by the time you are playing, hopefully, everyone agrees on how things work.

Even when they are losing, a player will admit the rules are important. No one likes someone who changes the rules depending on whether they are winning or losing.

We game designers may even appreciate the rules in areas outside of gaming...



... like in our governments.

We may appreciate that when systems are put in place for good reason – that even when someone may go off the rails in one branch, the system, hopefully, keeps them from destroying everything.

And we as designers recognize that these rules are important even when the person we like IS in charge.

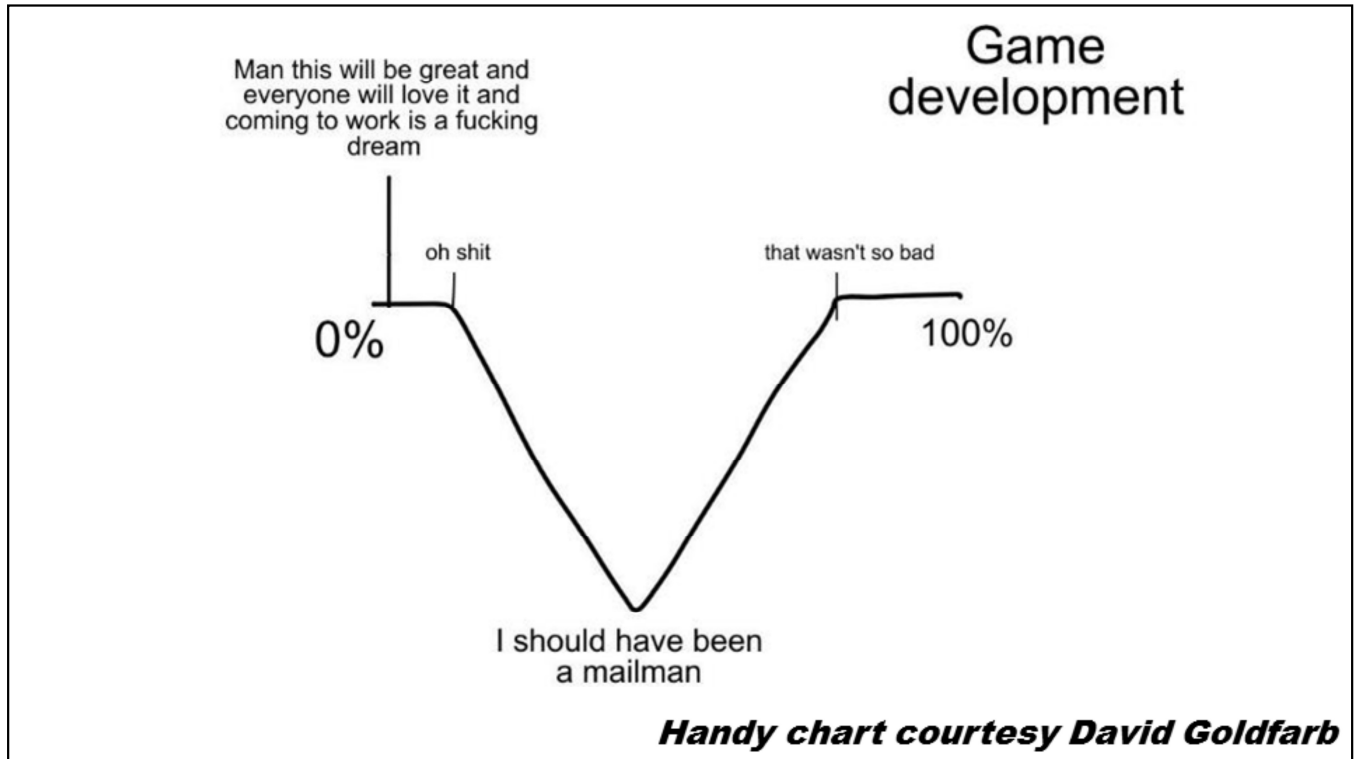


In game development, the reality is there are no rules, not hard and fast ones. Like all creative enterprise, many rules are a matter of personal taste

These are some of the designers who have been highly influential to me over the years, and if you asked them how to make a game, you would get some very different answers

So as you hear these specific rules here today, you may not think they apply to you.

And they may not, at least, not right now.



But maybe they will apply later.

We're all familiar with the dark doldrums of the middle of game development, when everything is horrible

When you are stuck on a problem, down at your lowest point, look back on the rules you hear today to see if they can reframe your thinking and get you out of a rut

This actually happened to me recently – running up to GDC I was working on my game *The Church in the Darkness* and found myself at the very lowest nadir of this chart.



I was working on a new demo for my game, and this is what most of it looked like.

I had stubbed in various parts of the level at their lowest possible pizzazz level, and during this time I was not enjoying working on my level.



But then in the last week I finally added all the fanciness to the level.

This went from an ugly snub to something with life in it – now there's a guy shooting at a pigs head next to cages with dead people. It also became a scene you could interact with, and that changed depending on the story.

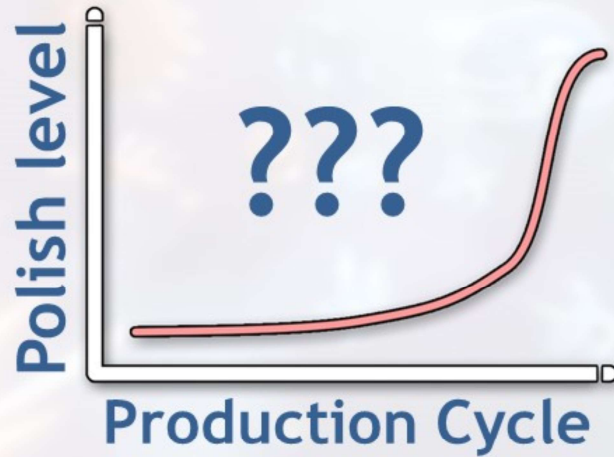
And this took me about an hour to implement.

What if I had just thrown this in earlier? How great would that have been to my morale when I was at my lowest point if I just made my level with a bit more pizzazz.



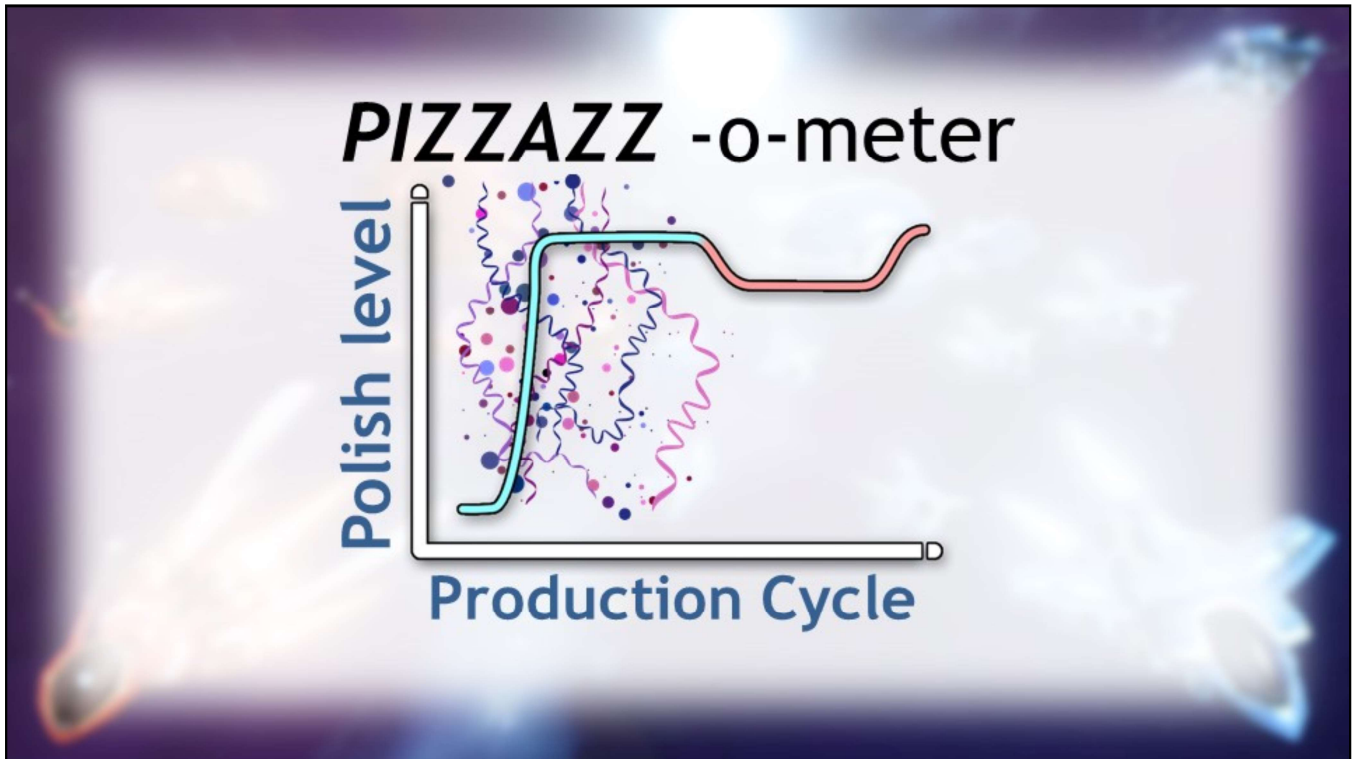
And I should have KNOWN this – because last year one of our speakers – Lee Perry – gave a talk on this very subject.

“SOLO” Polish-o-meter



He talked about that low trough, particularly when you are working solo, that you may forget to add the snazzy stuff because often it's the easy stuff. You're focused on the bigger risks.

But then you run the risk of hating your own game because it's so dull.



But if you add pizzazz earlier, the whole game becomes more enjoyable.

For more on pizzazz and polish, you should definitely go check out Lee's talk.

And the moral here is to listen to these rules not necessarily for them to help you today, but maybe to help you later when you desperately need it.

EVERYTHING

Now, when I do round up speakers for this session, I don't give them a theme – the speakers can pick any rule they want.

But now that I have all the talks, I can see that one has emerged. And this year it's **EVERYTHING YOU KNOW IS WRONG**

Which seems timely, doesn't it?

Our speakers today are going to challenge common assumptions and make us rethink how we develop games.

**EVERYTHING
YOU KNOW**

**EVERYTHING
YOU KNOW
IS WRONG**

Which seems timely, doesn't it?

Our speakers today are going to challenge common assumptions and make us rethink how we develop games.



And though I've been talking about how rules are important, it's important too to remember that many of us got into game development because we wanted to do something different with our lives.

We didn't pick the safe career, or the most lucrative career, we broke away from societal rules and wanted to do something different.

Which means, in a way, you want to break the rules.

And if you've been in game development long enough, you've heard those rules that you may feel are just stifling you.



Like maybe you've head the one about games with peripherals never selling.

Until someone makes the game that does.



Or that no one wants to play a Western Game... those will never sell

Until one does.



Or that adventure games are dead

Until they're not.

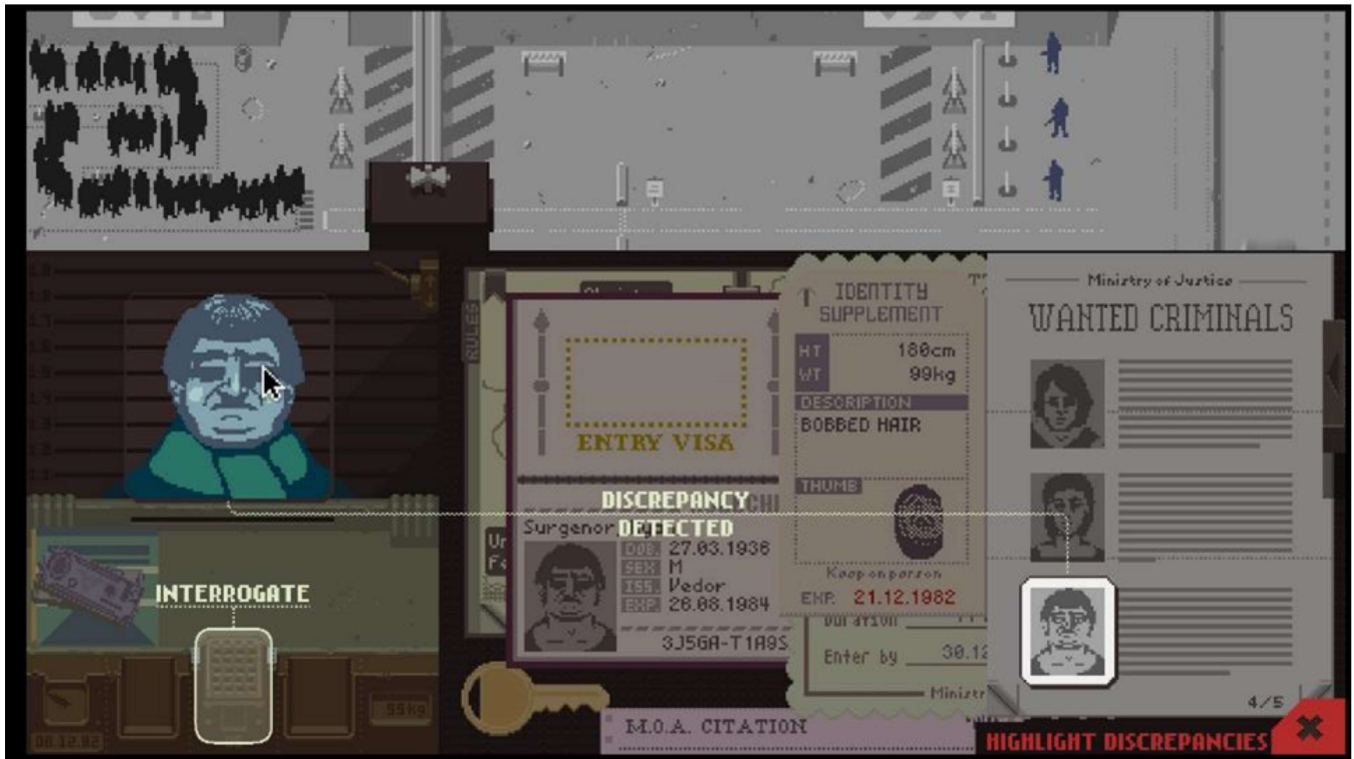


But we as designers can be just as guilty as the business people who tell us what won't sell.

Like when we declare that everyone online is an asshole so we can only make shooters



Or that players only want to win, that everything has to be easy.



Or the dreaded "That doesn't sound like fun."



From all those examples and many more outside the world of games, we know that it's through breaking these "constricting" rules that great new artistic breakthroughs can occur.

**A good rule doesn't
close down
possibilities,
it opens them up.**

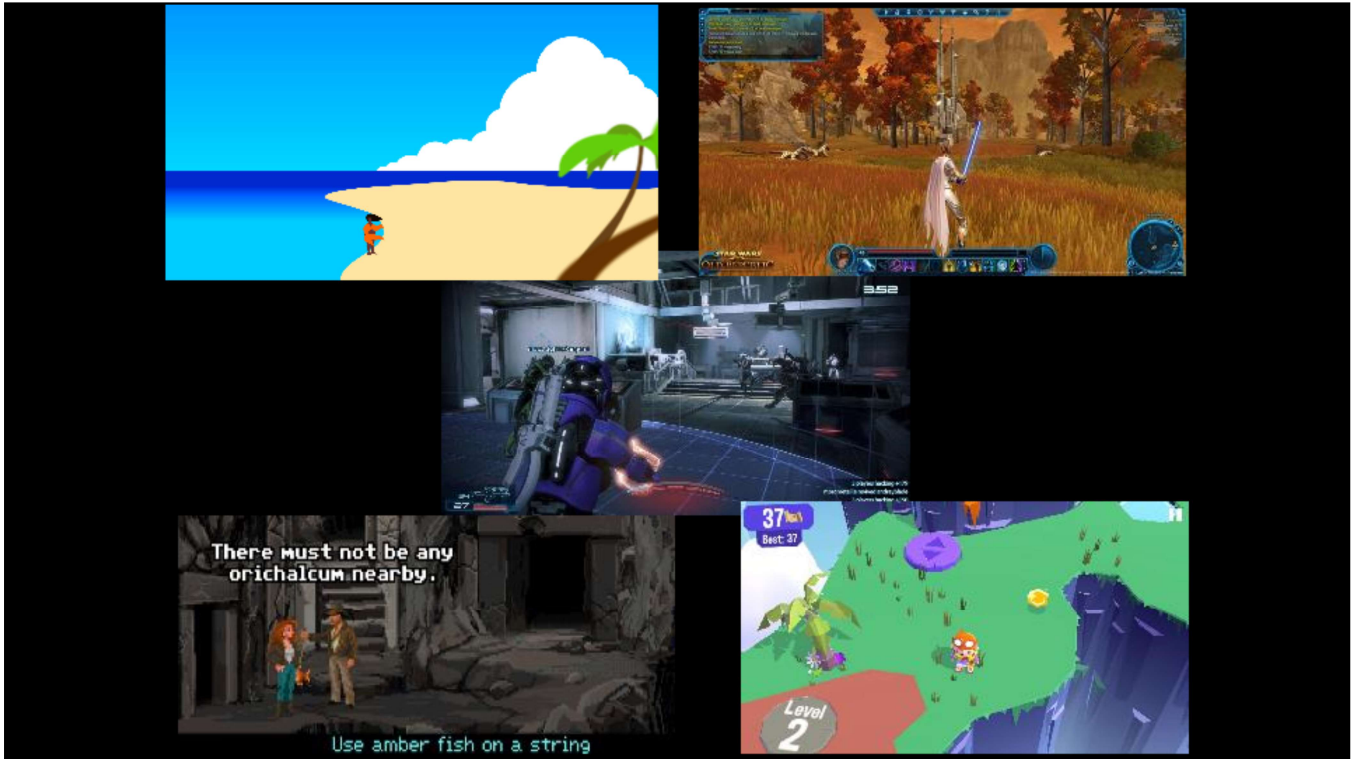
I would agree that a good rule doesn't need breaking – because it opens up the possibility set of what you can do.

It inspires you, and makes you be a more thoughtful creator.

I think the rules we have today are all of this sort.

**EVERYTHING
YOU KNOW
IS WRONG**

So today this is our theme after the fact, and I think these rules are going to open up your world.



Today, we will have a series of accomplished designers each tell you a rule they feel strongly about.

A little later, Damion Schubert is going to remind us who are true player base is.

Christina Norman is going to show you how your assumptions about what you need out of a feature may well be wrong

Hal Barwood is going to suggest the way we've done rules in the past has been wrong

And finally Luke Muscat is going to question the very validity of... well you'll just have to wait for it.

BUT FIRST

CHELSEA HOWE

***Family Guy: The Quest for Stuff,
FarmVille, Scaling the Sky***

**Creative Producteur & Owl Overseer
Owlchemy Labs
@Manojalpa**

First off, Chelsea Howe is going to tell you how you may be thinking of things at the wrong scale

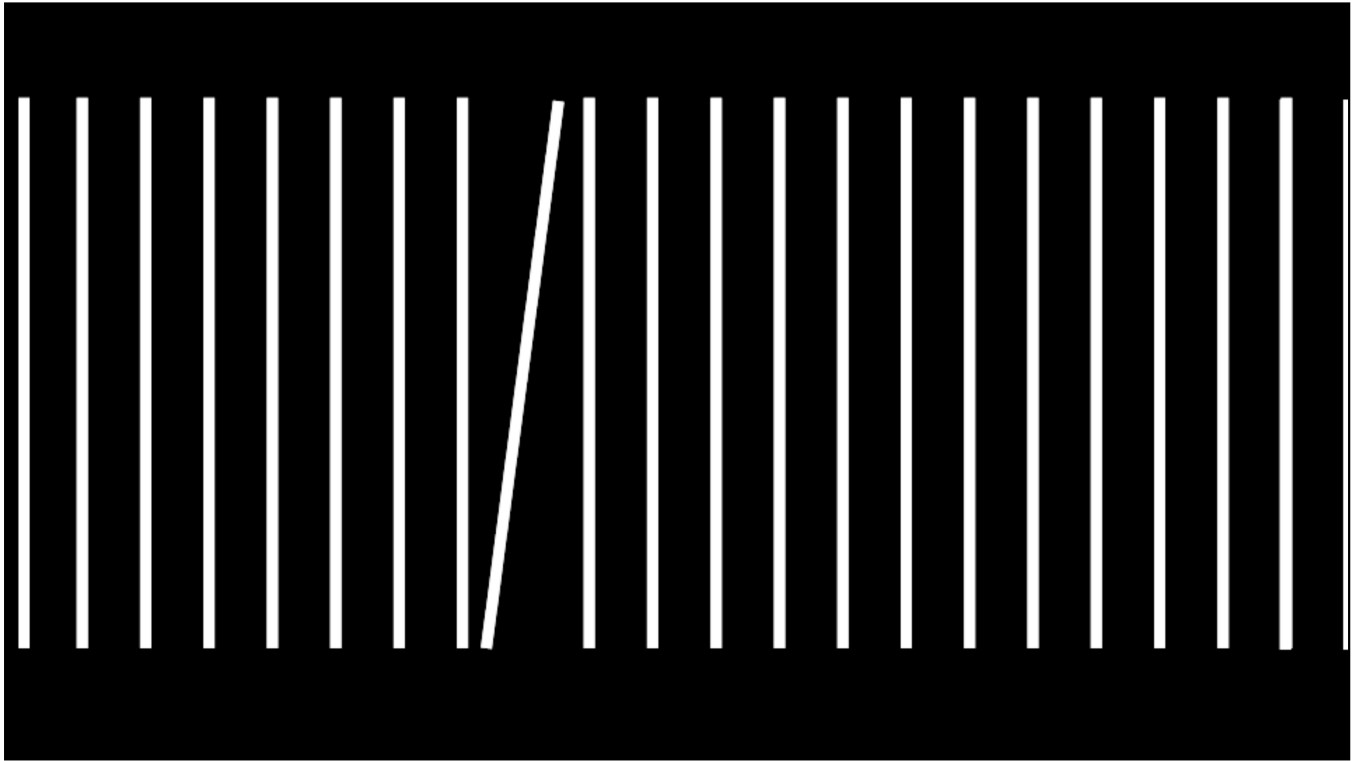
Chelsea likes making games that make a difference. She's currently Owlchemy Labs' creative producteur and owl overseer and co-founder of the Queerness and Games Conference. Previously, she's worked at EA, TinyCo, SuperBetter Labs, and Zynga, experimenting with F2P and consulting on playful experiences. By night Chelsea designs award-winning indie games, runs game jams, and teaches students at places like Coder Dojo and California College of the Arts. Her recognitions include Forbes 30 Under 30 in Games, Fortune's 10 Powerful Women in Games, and Fast Company's 100 Most Creative People in Business.

Design by Fractal

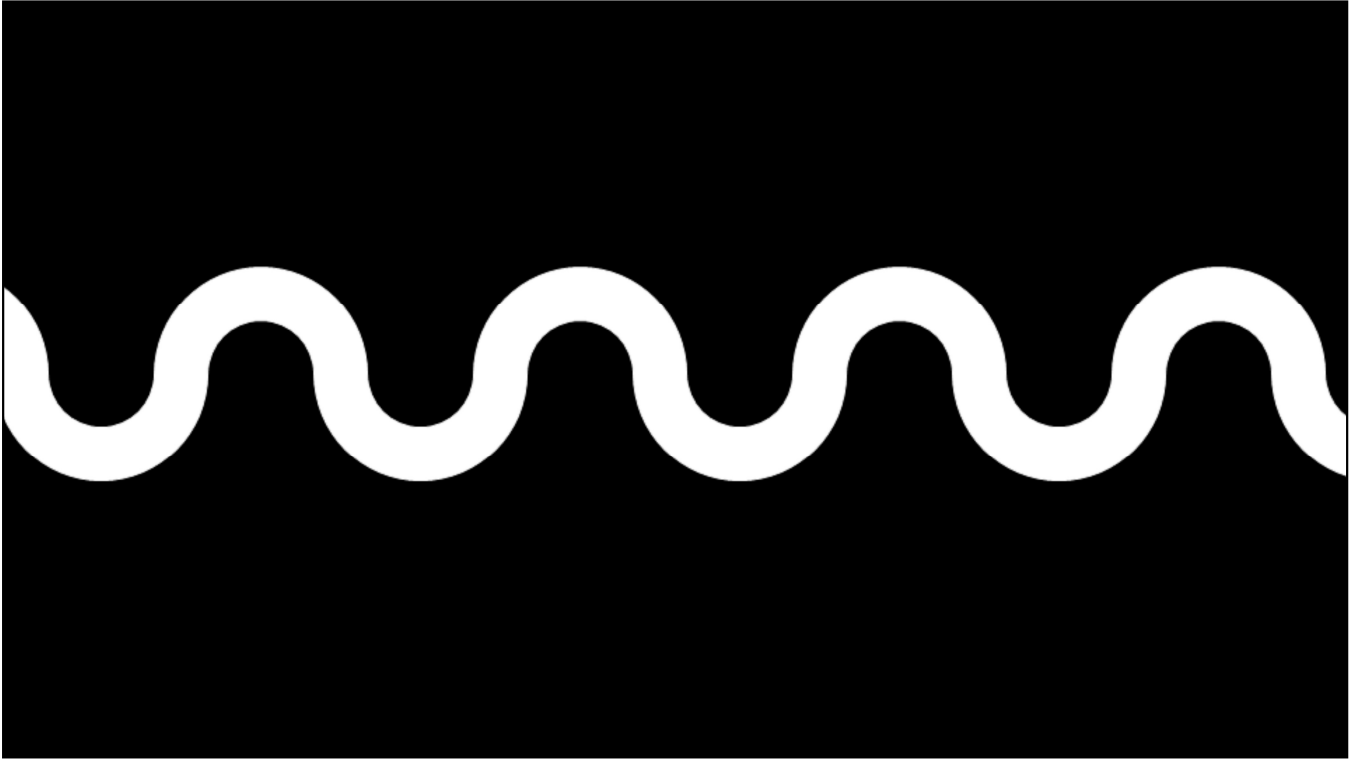
Today I'm going to talk about one of my favorite topics of all times: fractals. My rule, broadly, is "Design by Fractal" though as you'll see fractals are an all around wonderful way to conceptualize, communicate, and create games.

Why Fractals?

OK, so why fractals?



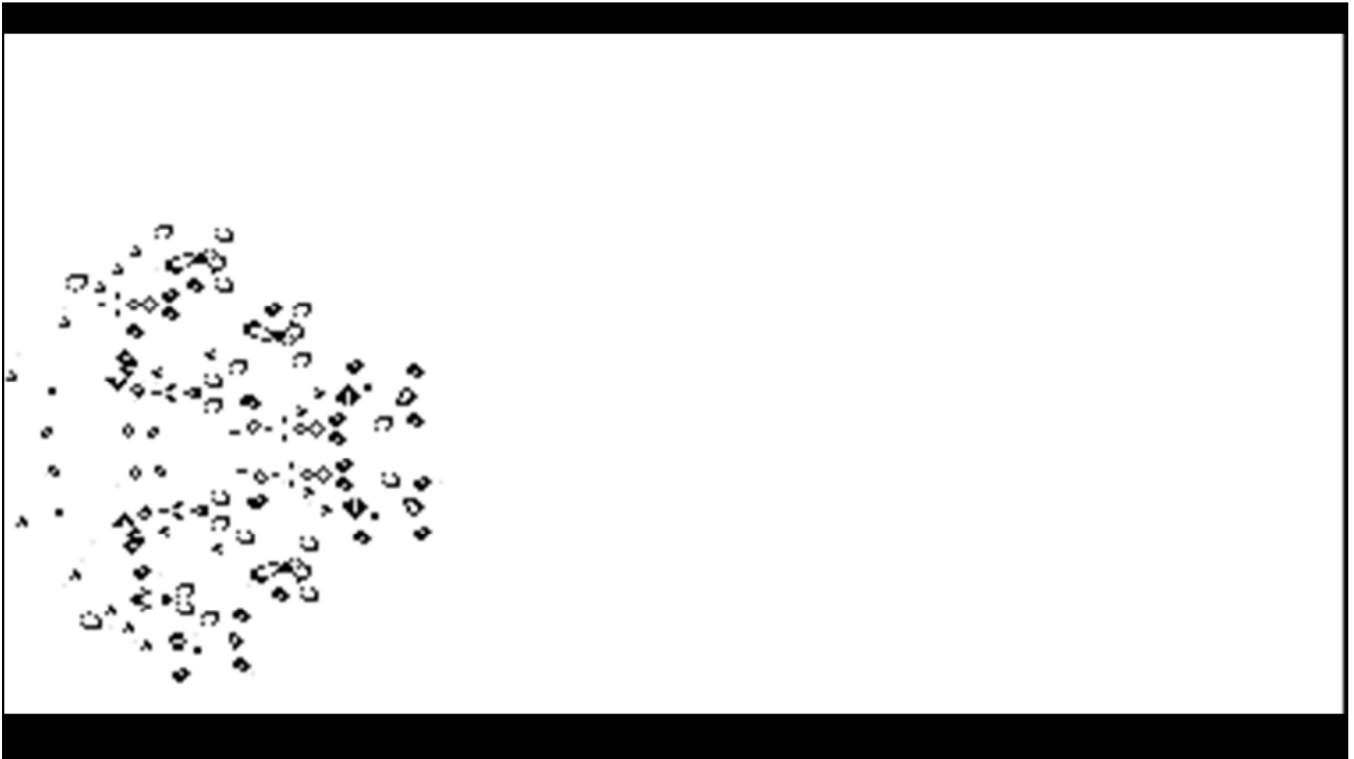
Our human brains are hard-wired to find patterns. Patterns help us cluster and predict the constant flow of data streaming into our brains. They let us survive and act and optimize our performance in an otherwise overwhelming world.



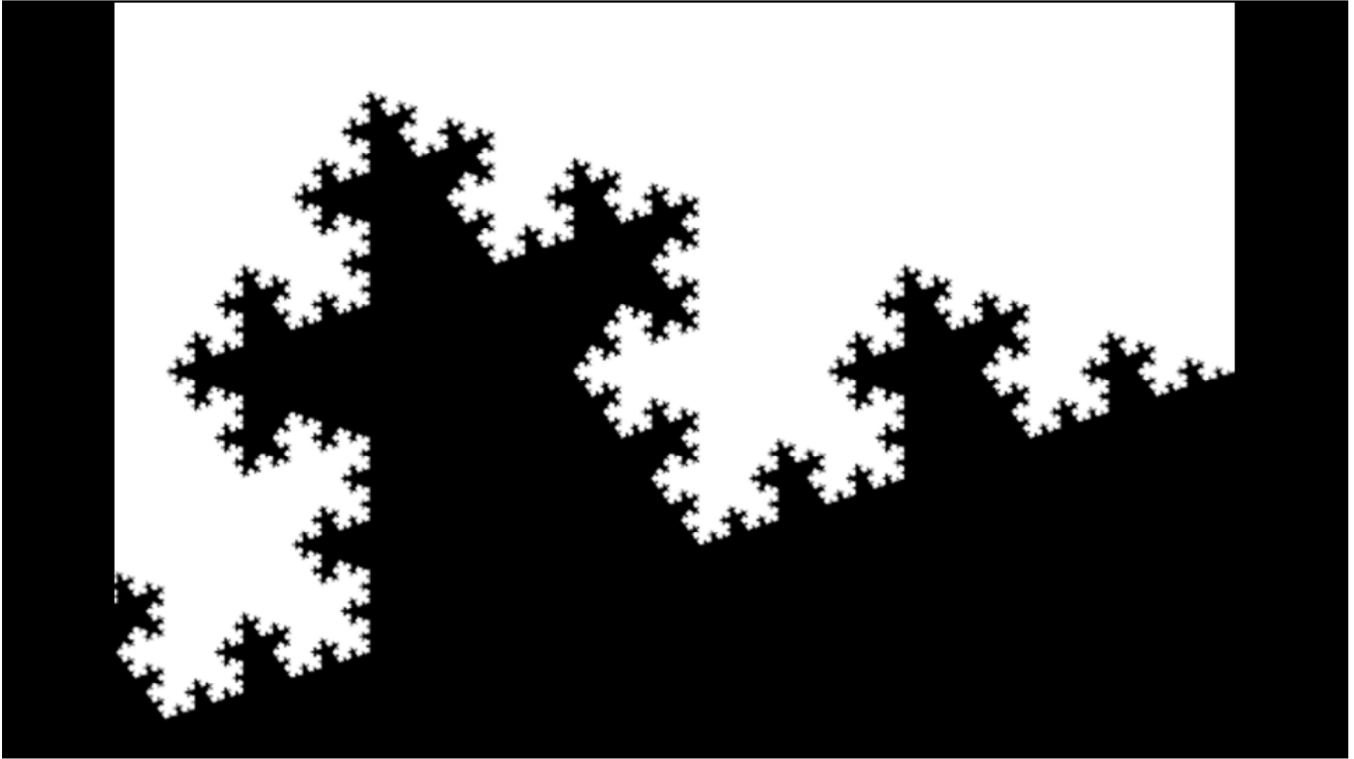
Some patterns are easier to see than others. Songs have choruses, verses, bridges. Novels have the three act structure and hero's journeys. Politics ebb and flow, conservative to liberal and back again.



But games aren't simple nor linear entities. They are vast and complex systems, multi-faceted, multi-layered.



Even games with the simplest underpinnings can have vast and unpredictable results.



And that's the beauty of fractals, too.

Fractals are patterns that exist across scale, that describe infinite systems. The defining property of a fractal is that it is self-similar – that is, you can see the same basic shapes and patterns no matter how far you zoom in or out – and often where you look.

$$Z_{n+1} = Z_n^2 + c$$

And all of that infinite complexity comes from one tiny little set of variables.

All fractals have an equation, an algorithm, that defines their repeating patterns. This simple equations here generates the Mandlebrot fractal.

That single algorithm feeds back in on itself, multiplying onward, outward, upward. With just one underlying and foundational set of variables, an infinite and beautifully resonant pattern emerges.

Game = Fractal

And ultimately, that is exactly what we want with our games. We want coherent design across massive, interrelated systems.

Game = Fractal

Vision = Algorithm

This mean your vision needs to be that algorithm. Everything in the game needs to stem from that core identity.

We use a bunch of different terms for it: a vision statement, a 1-liner, an X statement. But at its heart, it's the aggregate of values from which the rest of your game can be derived.

Game = Fractal
Vision = Algorithm
Pillars = Variables

Usually we designers go a step farther and define a set of keystones; those become the variables within our algorithm.

The clearer those pillars, the better we empower our teams to bring the game to life at whatever scale they're working at. If your team can really grok the algorithm, they can generate code, content, and creativity without designers always looming over them.

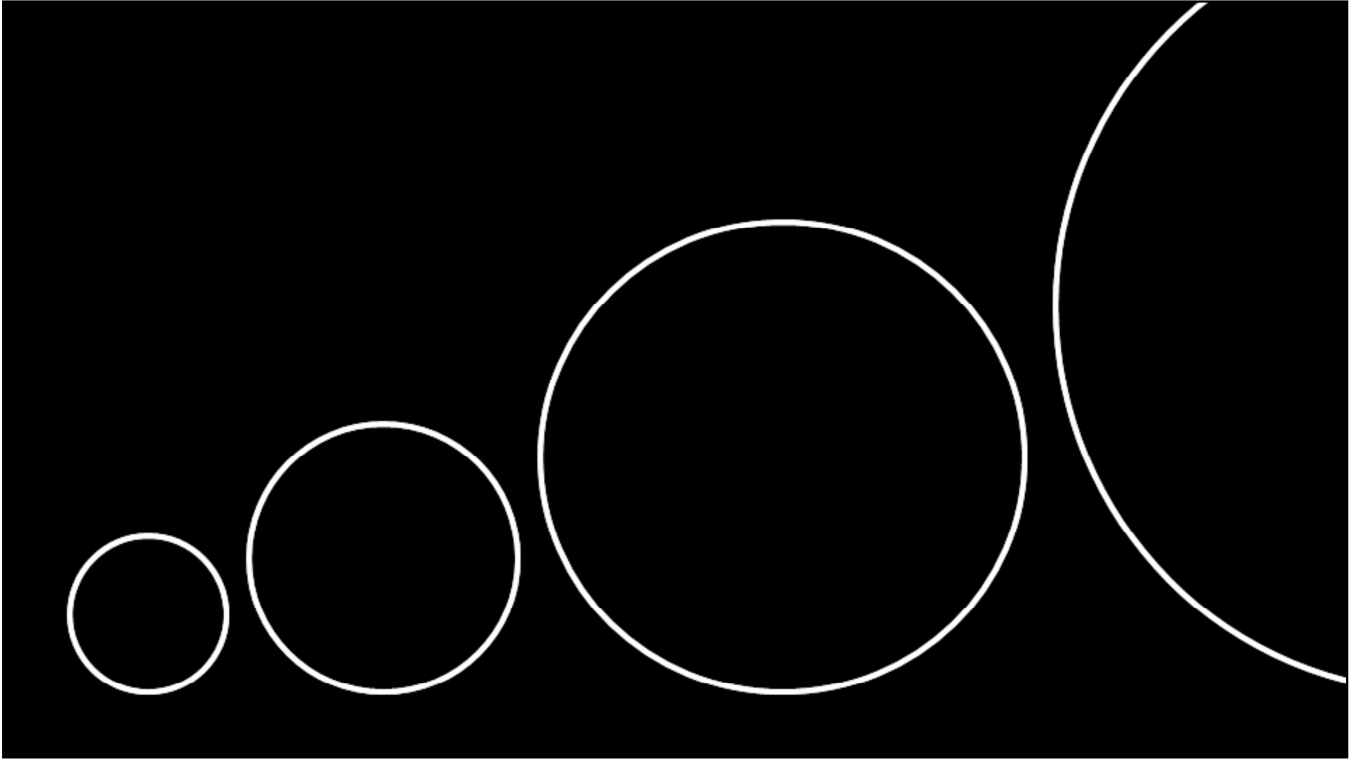
Fractals in Practice

So there's the theory. Let's see a few examples of how it plays out in reality.

Teaching Mechanics

(with fractals)

Fractals help when conveying information to players and swiftly teaching them the core components of your system.



A dominant learning paradigm is scaffolding, where you teach a small concept, practice it until mastery, and then build off that smaller concept into a bigger concept which feeds into a bigger concept. Learning fractals.



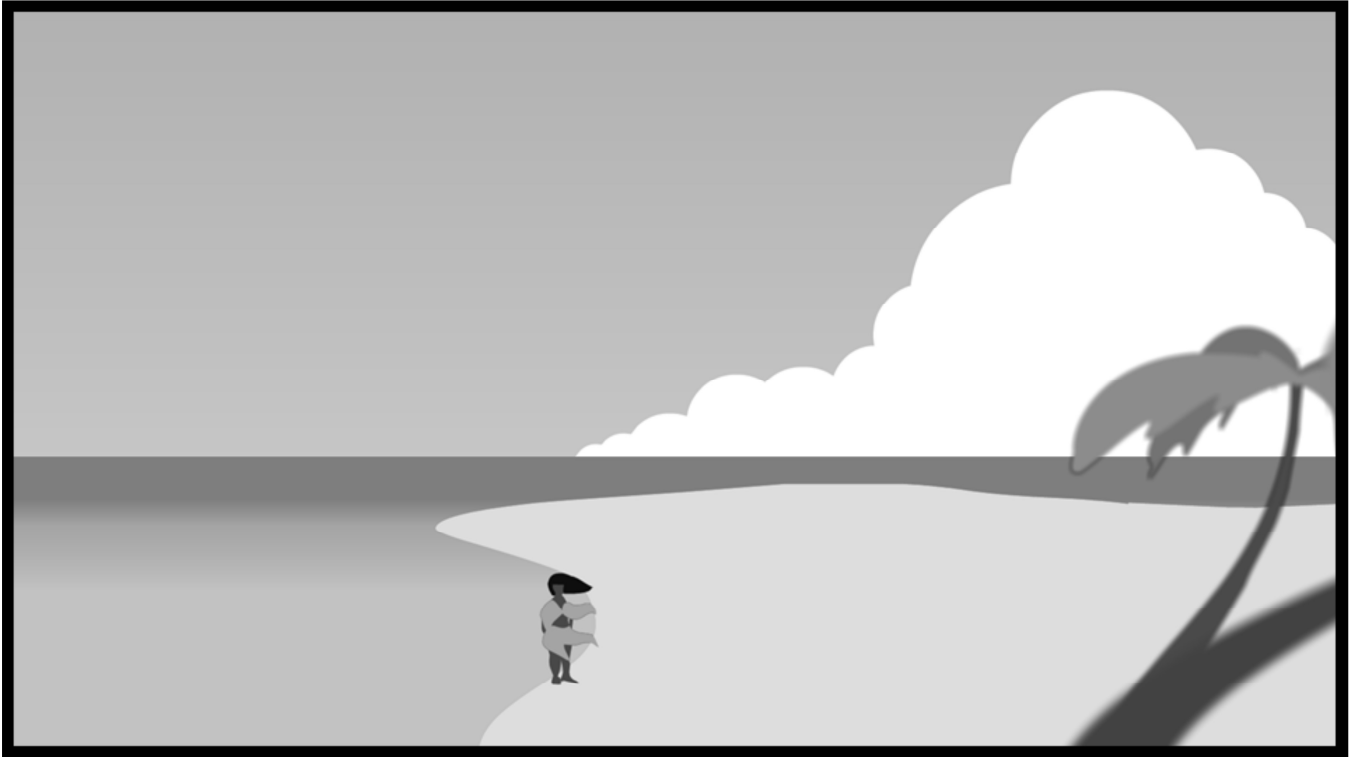
For Family Guy we started by teaching players about a single item - how to get it and use it. Then we introduced characters, who needed multiple items. Then we focused on districts, which held multiple characters. And finally, we showed them the meta game, which involved a multi-district stroyline. We started at the most immediate level of the fractal and zoomed out over time.

Creating Player Emotion

(with fractals)

There's also something magical that happens when players feel that order of magnitude change. There's a moment of awe when your mental model suddenly shifts, expands, or breaks entirely. The possibility space is forever altered. Your conception of the world changes. Fractals invite the sublime.

We can use those scale shifts to our advantage to evoke those more complex feelings in our players.

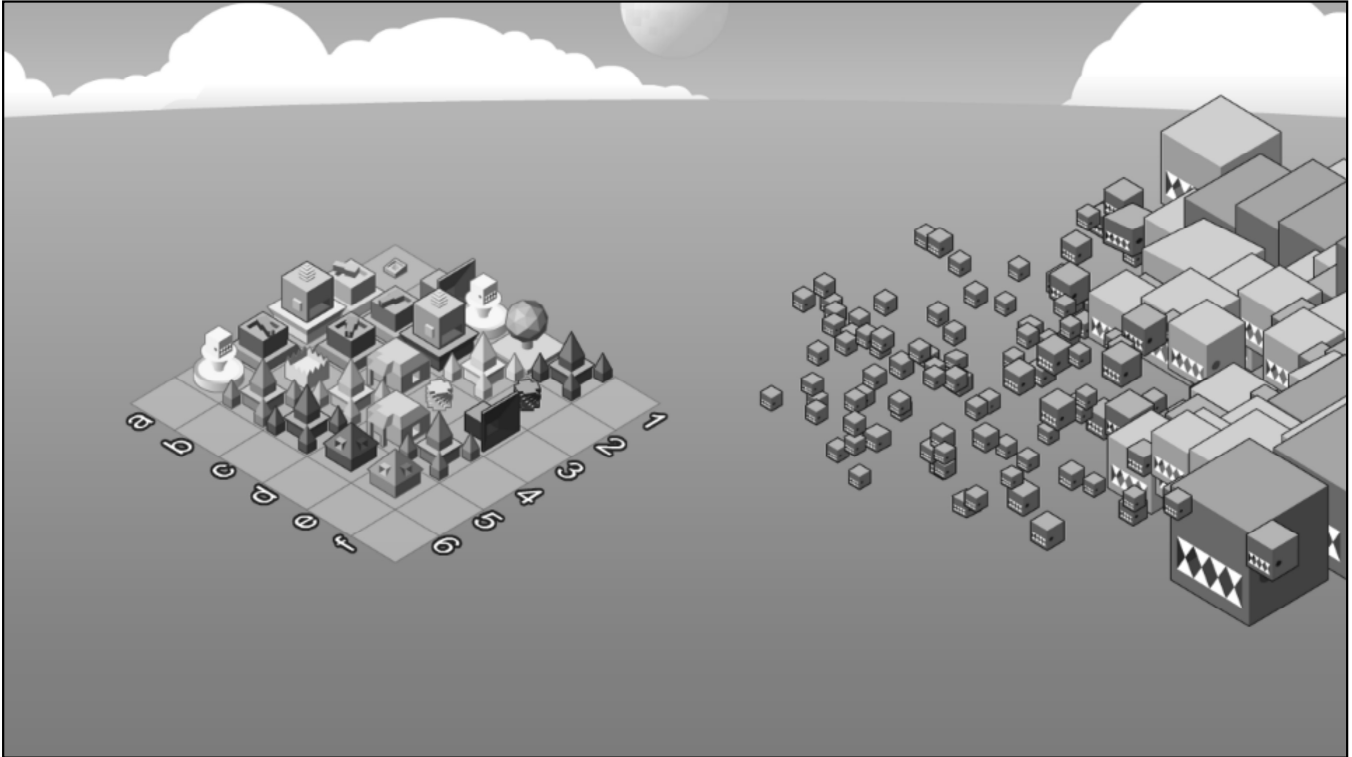


At the end of our indie game, *Scaling the Sky*, players wind up ascending so far into the sky that they pop out of the ocean, right back where they started. It was fascinating watching people's expression on our Let's Plays when they realize they've looped around, when they understand their Mobius world, their repeating pattern.

Creating Stakeholder Emotion

(with fractals)

And of course, powerful emotions aren't just useful to players, but stakeholders too.

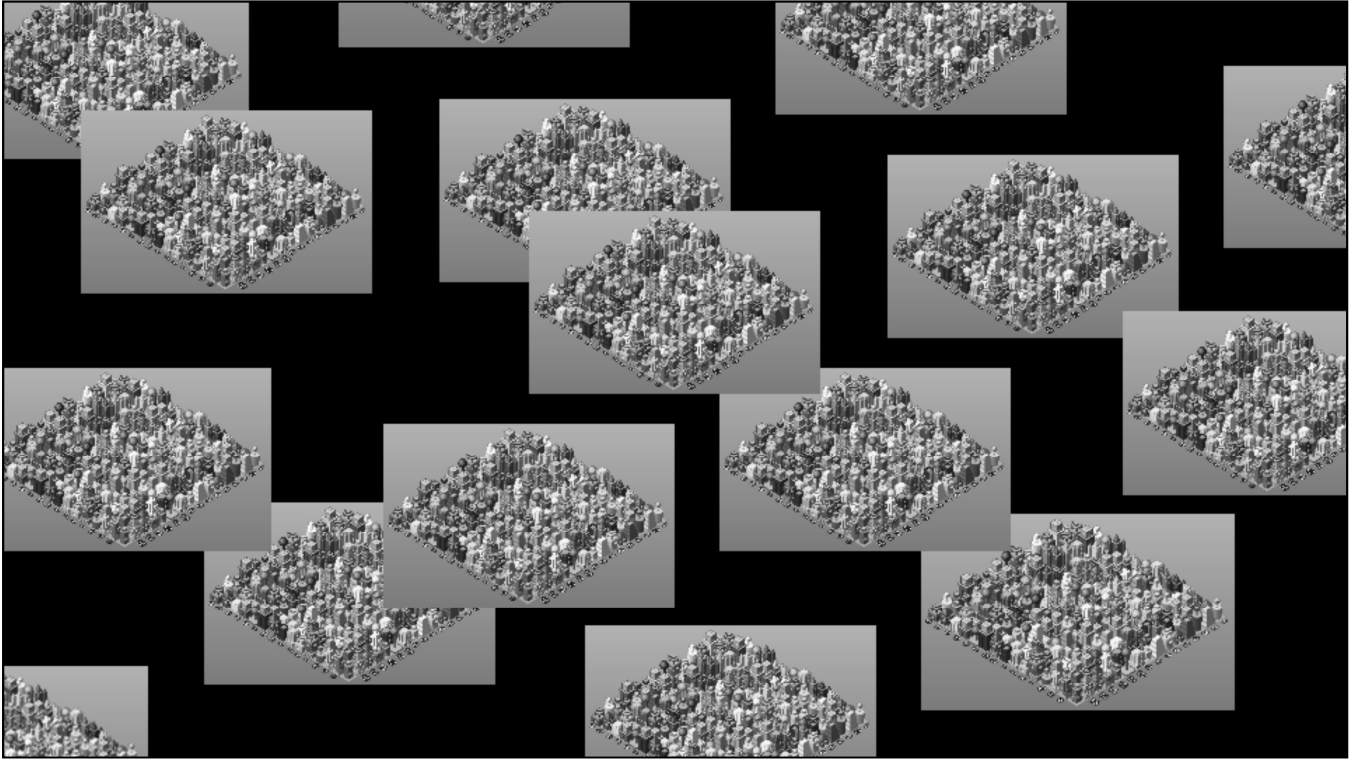


Here's the pitch we used for a game called CityStream. See if you can follow the fractal.

In City Stream, every person in a twitch chat becomes a little dot in the world.

Each dot can gather resources to help build this tower, trying to level it up so it can reach the moon.

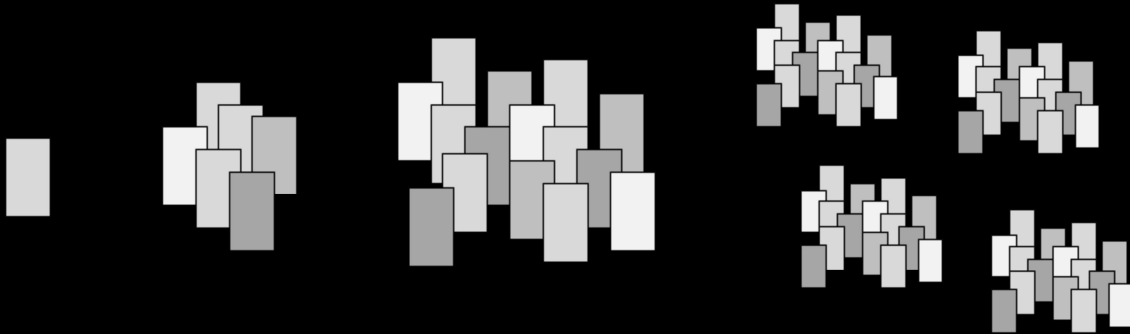
And this tower is actually just one structure in a bigger city that players build and customize and defend over time.



And that whole city they're working on? That's just one city, on one channel, and in fact any streamer on Twitch can launch their own city against someone else's in massive territorial battles for ultimate glory.

Build

Building → City → Society → World



When we subvert expectations we jokingly say “mind-blown”, but that’s exactly what we’re doing: disrupting mental models. Changing people’s schema.

When you pitch, you WANT to blow people’s minds. Designing with fractals means asking what your game would be like if you ramped it up another level, if you zoomed out one more order of magnitude. If you looked to the left and discovered a second, self-similar branch of the system.

Understanding Player Engagement

(with fractals)

And fractals don't just engage people in the moment.

I also use fractals for virtually any element that can hold a player's attention - narrative, difficulty, level unlocks, marketing, content releases

	Small	Medium	Large	XL
Narrative	Dialogue	Quest	Story Arc	Hero's Journey
Release Cadence	Hot Fix	Daily Challenge	Weekly Quests	Quarterly Expansion
Audio	SFX	Character Timbre	Character Theme	Soundtrack
Content Marketing	Teaser Image	Teaser Trailer	Full Trailer	Game Release

A narrative fractal might start with a single line of dialogue or flavor text. Then conversations, then quests, then story arcs, and finally the entire hero's journey.

For content releases you might have daily bug fixes, bi-weekly challenges, weekly quests, monthly feature additions, and quarterly expansions.

Go ahead – any area of your game, anything whatsoever, use fractals to think about how it scales: either over time, in size or scope, or level of attention. Where are your gaps?

Communicating Design

(with fractals)

I know we all love design documentation, and that's another area where fractal like structures are well utilized.

This is a single line about how great my game idea is so that you get the gist and really want to know more

This is a single line about how great my game idea is so that you get the gist and really want to know more.

In an elevator I can give you maybe six solid sentences. That's enough to give you one sentence of lead in, a sentence that's the core idea, three supporting features or pillars that make the game really interesting, and then a final line summarizing why the game deserves your time and attention.

The faster I'm able to communicate the most salient parts of the game, the better the game's chances.

This is a single line about how great my game idea is so that you get the gist and really want to know more.

In an elevator I can give you maybe six solid sentences. That's enough to give you one sentence of lead in, a sentence that's the core idea, three supporting features or pillars that make the game really interesting, and then a final line summarizing why the game deserves your time and attention. The faster I'm able to communicate the most salient parts of the game, the better the game's chances.

This is a single line about how great my game idea is so that you get the gist and really want to know more.

In an elevator I can give you maybe six solid sentences. That's enough to give you one sentence of lead in, a sentence that's the core idea, three supporting features or pillars that make the game really interesting, and then a final line summarizing why the game deserves your time and attention. The faster I'm able to communicate the most salient parts of the game, the better the game's chances. This is a single line about how great my game idea is so that you get the gist and really want to know more. I just read a few more words here to make it look not so copy-pasted.

In an elevator I can give you maybe six solid sentences. That's enough to give you one sentence of lead in, a sentence that's the core idea, three supporting features or pillars that make the game really interesting, and then a final line summarizing why the game deserves your time and attention. The faster I'm able to communicate the most salient parts of the game, the better the game's chances.

This is a single line about how great my game idea is so that you get the gist and really want to know more.

In an elevator I can give you maybe six solid sentences. That's enough to give you one sentence of lead in, a sentence that's the core idea, three supporting features or pillars that make the game really interesting, and then a final line summarizing why the game deserves your time and attention.

The faster I'm able to communicate the most salient parts of the game, the better the game's chances.

This is a single line about how great my game idea is so that you get the gist and really want to know more.

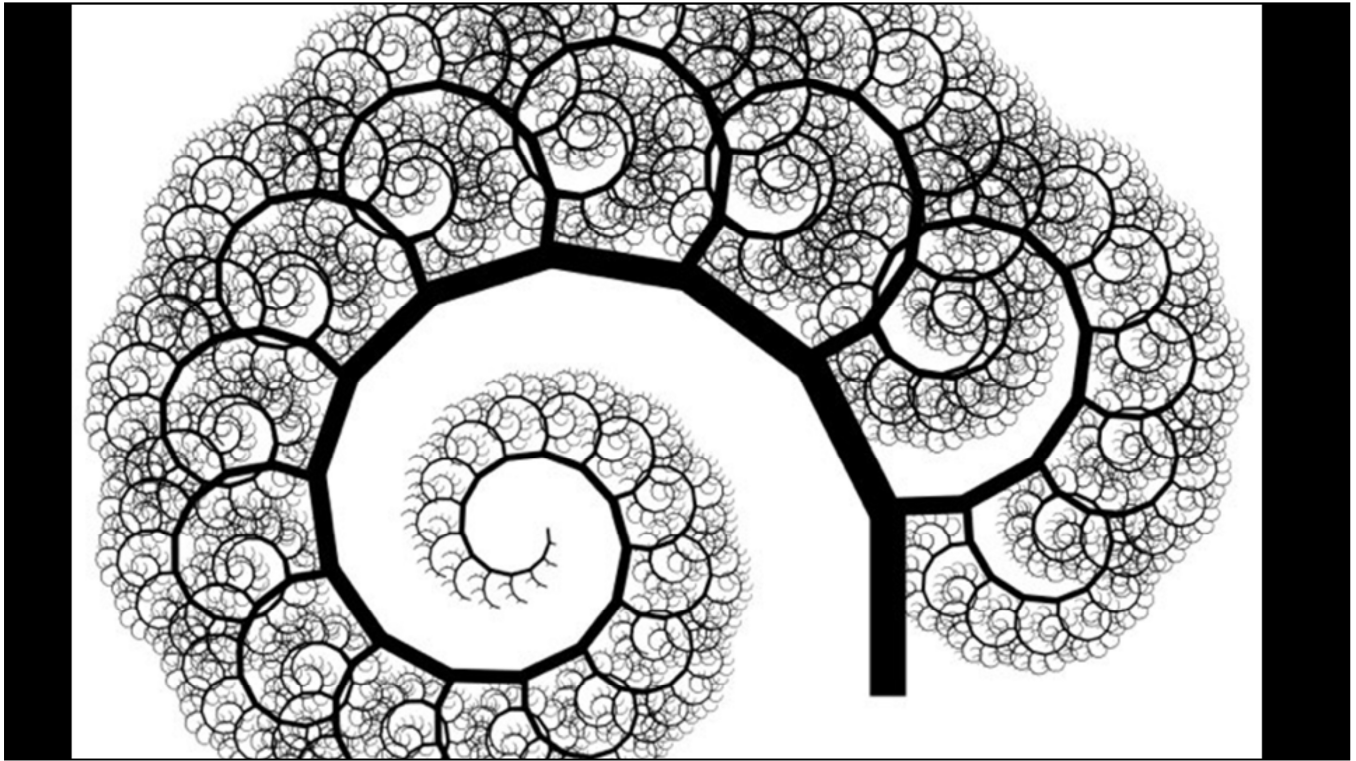
You have the one-liner, the elevator pitch, the one-pager. Then there's the concept deck, the systems doc, all the way to the oft-lamented design bible or wiki. Thinking of them all as the same basic entity, just at scale, helps me keep the ideas consistent throughout. And remembering that different people on a team work at different levels of the fractal helps me figure out the most useful way to talk to them. The global exec team needs the one-liner. The technical artist needs their subset of the design bible.

(I've also noticed some people have a 'default scale' when they think up ideas - I tend to revert to a concept doc; find the scale you're comfortable at and then practice scaling up or down).

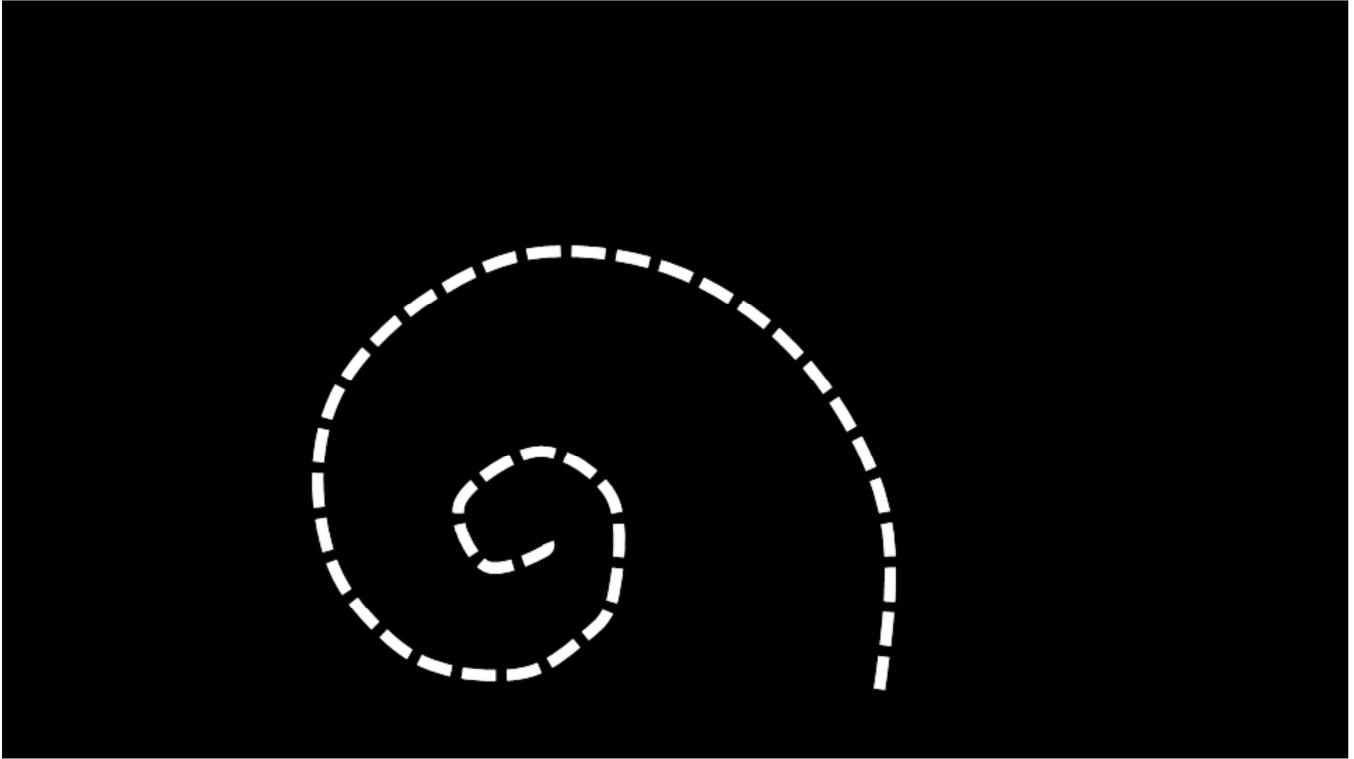
Understanding Other IP

(with fractals)

The good news is that fractals aren't just good for communicating your own IP, but for understanding other IP as well. If you're aware of patterns across scale, you can identify the keystones of other identities – other brands, franchises, experiences.



I've designed for many different licensors and IPs, and unlike most folks I love that work. Because, again, when you can see that pattern, you are empowered to use it in new, unexpected ways.



Once you've seen the consistent themes in a celebrity's twitter feed, instagram, interviews, concerts, filmography, you understand how to infuse that identity into a game.

****Specific Example + Talk about Trust/Bonding/Faith in external partnerships when you can echo their values back to them****

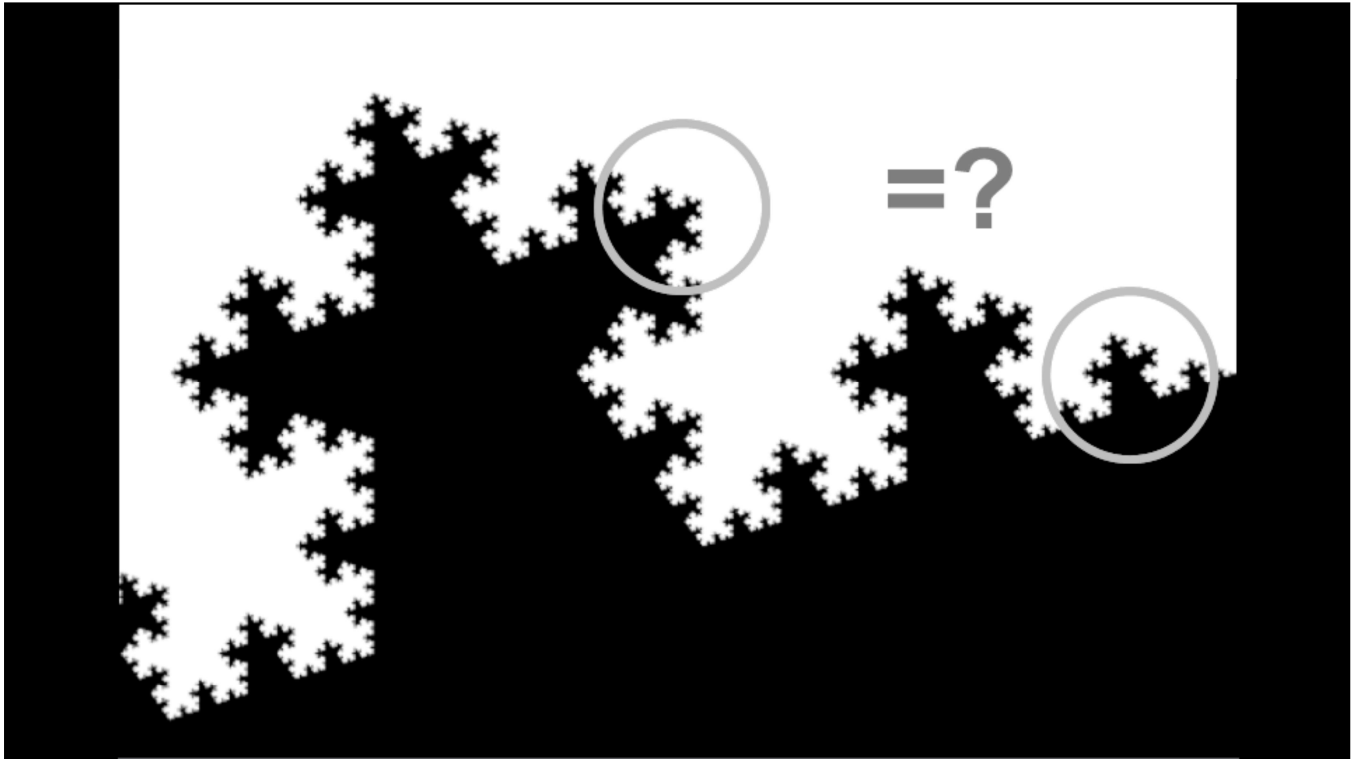
Understanding The Future

(with fractals)

And of course, this isn't just about seeing the current patterns, but looking at how they've changed over time. Because once you understand the way something HAS changed, you're better able to predict how it WILL change.

Where are we going?

When my UX designer and I were coming up with early concepts, we found a curious trend. We saw that despite mobile gamers complaining about a lack of real gameplay - which meant in almost every case moment to moment control and input and strategy - they would almost always use an 'automate' button for that gameplay as soon as it was provided. We saw idle games becoming increasingly popular on flash portals and just starting to leak into the apple charts - games that for the most part play themselves. We saw an overall shift of player attention from the moment to moment interactions to the meta game.



In this case, we saw a broad, general algorithm playing out in many different areas. I mentioned fractals aren't just about scale; they're spatial as well. By seeing a pattern in one area of entertainment, we figured we could apply it to *our* area of entertainment.

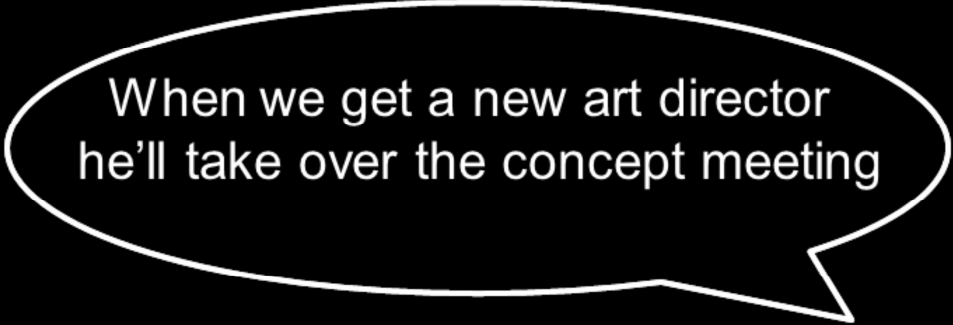
We fused that trend into our concept. Other design teams had been pitching for over a year; our design was greenlit in less than two months.

Understanding The World

(with fractals)

I can't help thinking in fractals at this point. So let's talk about one more application.

Microcosm




When we get a new art director
he'll take over the concept meeting

(hopefully not you)

A team is a small, small microcosm of society. Every team has its own quirks and its own dynamics, but it will always in some ways reflect bigger issues. When small things happen in your team, don't just let them slide. Don't accept them as part of a bigger pattern.

Macro Effect



When we get a new president
they'll restore basic human rights

(hopefully you)

Because change at any scale affects the others. One simple correction can make a difference. When we think in fractals we are no longer helpless individuals; we are simply at a different scale, and we can create vast change.

Thanks

@manojalpa

Thanks

CHELSEA HOWE

***Family Guy: The Quest for Stuff,
FarmVille, Scaling the Sky***

**Creative Producteur & Owl Overseer
Owlchemy
@Manojalpa**

[Richard] This is exactly the type of rule that I think can really help you when your stuck – maybe you need to change your scale and look at your problem again.

DAMION SCHUBERT

*Dungeon Boss,
Star Wars The Old Republic*

**Creative Director
Boss Fight Entertainment
@ZenOfDesign**

Next up, you may know Damion for his many years working on MMOs, culminating as being lead designer on Star Wars The Old Republic. He now works on the mobile RPG Dungeon Boss.

Some years back Damion gave a famous talk on how everything we know about writing design documents was wrong, now he's going to tell us how we may well be looking at our player base wrong...

Damion!



Free is the DEFAULT



When we talk about Free to Play games, the most natural thing to do is to talk about ... how to make money. I mean, that's not unnatural. After all, while many of us are doing this for the passion of making games, we do like to actually be able to pay our rent. And, you know, our bosses generally expect us to build a viable business, for some reason.

But there's been lots of talks about the spenders. You know, the 'whales' (I hate it that we call them that). You wanna hear more about that, there's entire conference tracks – heck, entire conferences – about how to make money.

2%

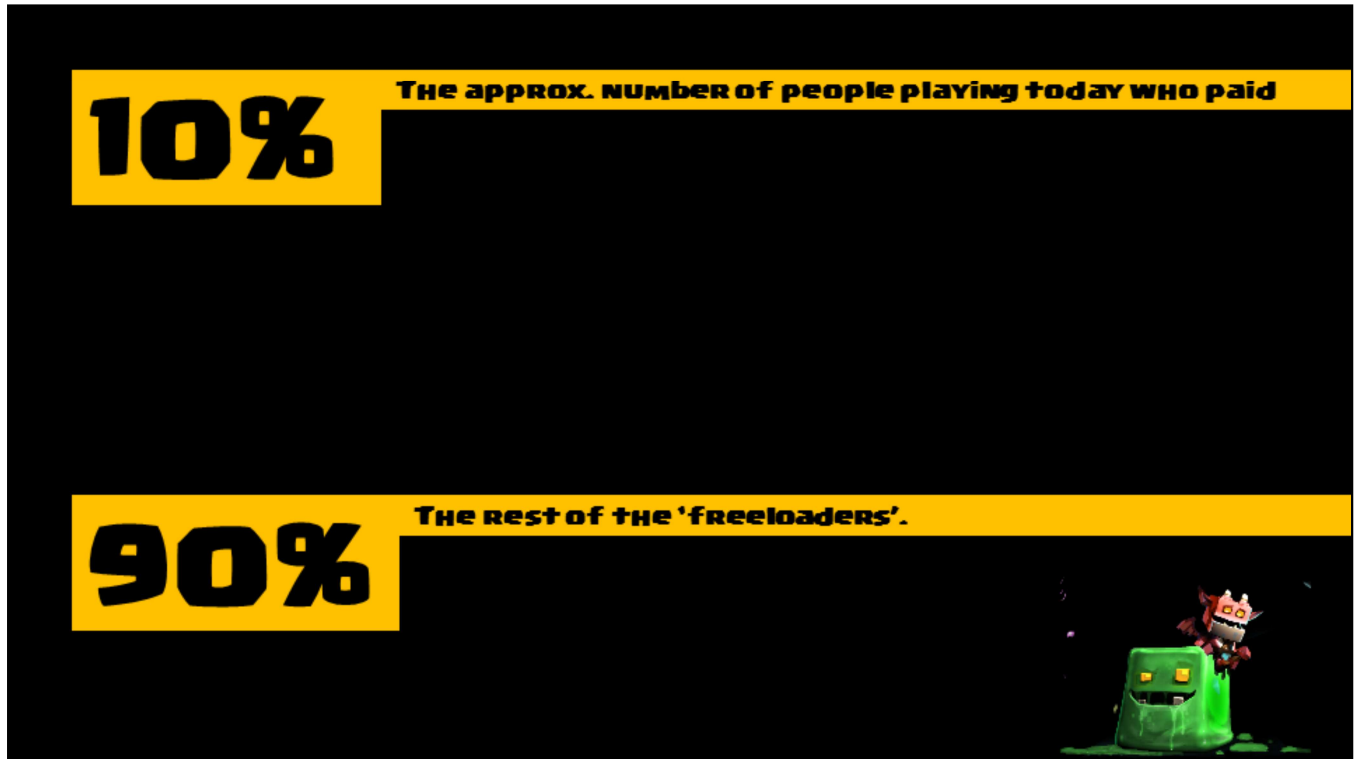
The conversion rate of a successful Zynga game...

98%

The percentage of people who never spend a dime.



I wanna talk about the other edge of the coin. I mean, we know that these heavy spenders spend a lot of money in our games. They are, effectively, subsidizing a whole bunch of other players who aren't paying anything at all. How big is that disconnect? Well, I kept looking for numbers that are actually releasable. Here's a good one. Zynga games, in their heyday, would declare a game a huge success if one of their facebook games had 2% conversion rates – i.e. 2% of their players EVER paid them a dime. That means that the other 98% are, effectively, freeloaders.



I wanna talk about the other edge of the coin. I mean, we know that these heavy spenders spend a lot of money in our games. They are, effectively, subsidizing a whole bunch of other players who aren't paying anything at all. How big is that disconnect? Well, I kept looking for numbers that are actually releasable. Here's a good one. Zynga games, in their heyday, would declare a game a huge success if one of their facebook games had 2% conversion rates – i.e. 2% of their players EVER paid them a dime. That means that the other 98% are, effectively, freeloaders.



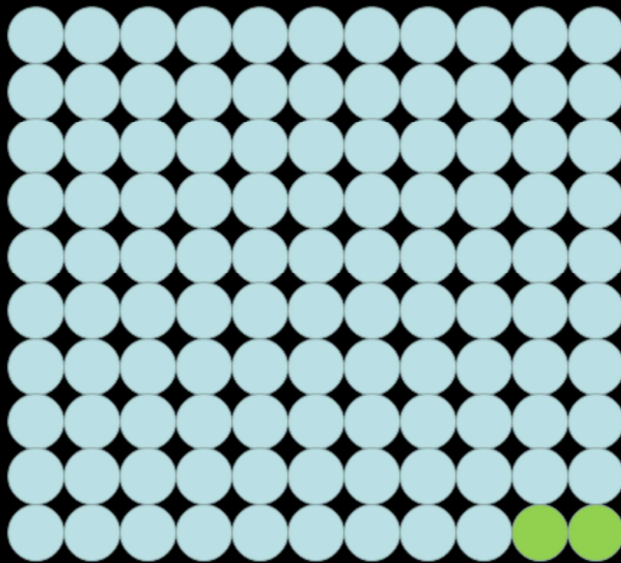
For most people, 'Free-to-Play' means... exactly that.

For most people, free to play means exactly that. They download your game, they play the game, and they do it all intending to never give you a single dime. Even if they love it. These guys have lots of free options for entertainment nowadays. Many of them are kids, who don't have credit cards. Some of them see playing for free as a badge of honor. How many of us played Candy Crush as far as we could go just to see how well we could do?



So what does that mean for us designers? Those of us who are battling the forces of the evil Project Managers who want to turn our customers over and shake them for spare change?

Takeaway #1

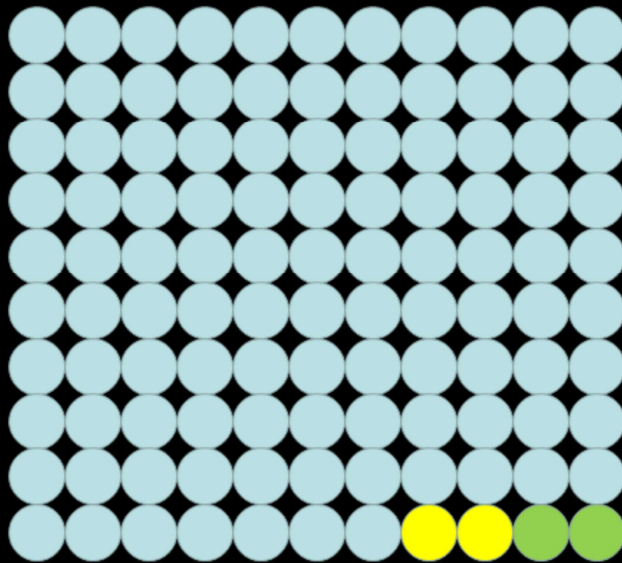


Realize that the bulk of your population will never spend.

Well, first off, designers need to get in the mindset that the massive bulk of your population is playing for free. Heck, remember that 2% number we mentioned before? Let's double that. You're an amazing designer. You have discovered the holy grail of monetization. You're a genius.

.... And 96% of your players are STILL not giving you any money. The massive bulk of your population is still playing for free.

Takeaway #1



Realize that the bulk of your population will never spend.

Well, first off, designers need to get in the mindset that the massive bulk of your population is playing for free. Heck, remember that 2% number we mentioned before? Let's double that. You're an amazing designer. You have discovered the holy grail of monetization. You're a genius.

.... And 96% of your players are STILL not giving you any money. The massive bulk of your population is still playing for free.

Takeaway #2



Virality



Rivals



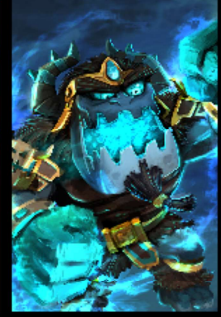
Content



Partners



Fodder



Community

Free Players should add value to YOUR spenders or to the game itself.

And let's keep in mind that freeloaders are not just freeloading. In the worst case scenario, they are making the game viral. You played Clash Royale because you heard about it from someone else. You played Candy Crush to see if you could catch your buddy in the 'saga map'. In Second Life, free players might make amazing content for you to enjoy. In Star Wars: the Old Republic, free play filled out dungeon queue with other players again. And, of course, in most mobile games, free players give plenty of easily stompable opponents for spenders.

You shouldn't want to get rid of free players. Their existence should make your game stronger, even if they never spend. That's what being a F2P designer really centers on.

Takeaway #3



Build and Test the game as if players will never spend.

So if you ARE building a game that's free, great. Sure, you should build, test and focus on things that people spend on, but for the love of god, don't lose sight of the fact of what it's like NOT to spend. Be sure your testers are TESTING what it's like to play without money. Be sure your designers are playing like a miser. Because frankly, if everyone is playing with free currency like the 1%, they are NOT playing the same game as everyone else. And you miss stuff – like playing with characters that haven't been maxed out is REALLY lame because you don't see their special abilities.

One of the things that happened to us on Dungeon Boss was that we didn't keep track of our pop ups. We kept adding game systems that had popups for our non-spenders. When we finally went back and took a look at things, we had just an OPPRESSIVE initial experience into the game, just a few months after ship. We actually spent a milestone going back and streamlining this experience to being sure that new players have a great experience.

Takeaway #4



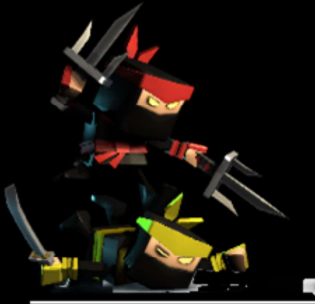
Watch those paywalls - Give People a chance to fall in love with the game.

And about those free players – watch those paywalls. A paywall is when you put a huge ‘cliff’ in the game design, that’s designed to make the player either spend or choose a free alternative – usually grind a few levels, or wait for stamina to refill. One of the most common mistakes I’ve seen is for designers to put up too harsh paywalls too quickly. They go up before players fall in love with the game. When this happens, your brain extrapolates that pacing to the rest of the experience. You do the math, and you realize – wow, this supposedly FREE game is really not free at all! If Free is the Default, then the people who fully intend NEVER to pay will be chased out almost immediately – and your game will be a ghost town.

By contrast, when we converted Star Wars: the Old Republic to F2P, I fought hard – and won – my battle to not put up anything that was even remotely like a paywall on the first planet – that’s 3-4 hours of gameplay, people. Why? Well, my argument was that we had empirical proof, according to EA’s testing lab, that this was the best MMO newbie experience ever released when we were a subscription game. Why screw that up? Instead, we bet that if we gave people 3-4 hours for free, they would fall in love with the game, and they would spend. And in our case, this was the right decision. We ended up GAINING subscribers when we converted

to free to play – because it turns out that we gave players a great, full-featured demo experience, and they WANTED to help out.

Takeaway #4



Matchmaking is vital to keep F2Ps from being stomped too often by spenders.

Remember how I said that F2P players make for good fodder for big spenders? Yeah, well, you have to be careful. It's not a fun experience to get ROFLStomped ridiculously. Now, I'm not saying that free players should always be paired against other free players. However, and this was a painful learning experience on Dungeon Boss, if you have a PvP system where your free players are frequently being paired against players who have spent a thousand bucks, they're usually going to get stomped. If that happens, your free players will cease to interact with that system, which can really cause a cascading spiral of derp. If Free is your default, but your free players stop playing an aspect of your game, that aspect of your game will go into a death spiral.

In many games, you buy a LOT of power when you spend. In Dungeon Boss, the characters in our gem portal are MUCH more powerful than the free heroes, and people who spend have a MUCH easier time acquiring them and maxing them out. Which is fine. What's less fine is when free players face a wall of enemies they cannot defeat.

What you want to do is be sure that players encounter close battles. You want them to feel like if they spent a TEENSY amount, they'd be competitive. But you don't want them to hit a

brick wall. You don't want them to see NO CHANCE of success. Oh, sure, David beat Goliath, but this is a great story because of how fantastically impossible that was. Most of the time, Goliath stomps David so hard that David throws his iPad against the wall.

Takeaway #6



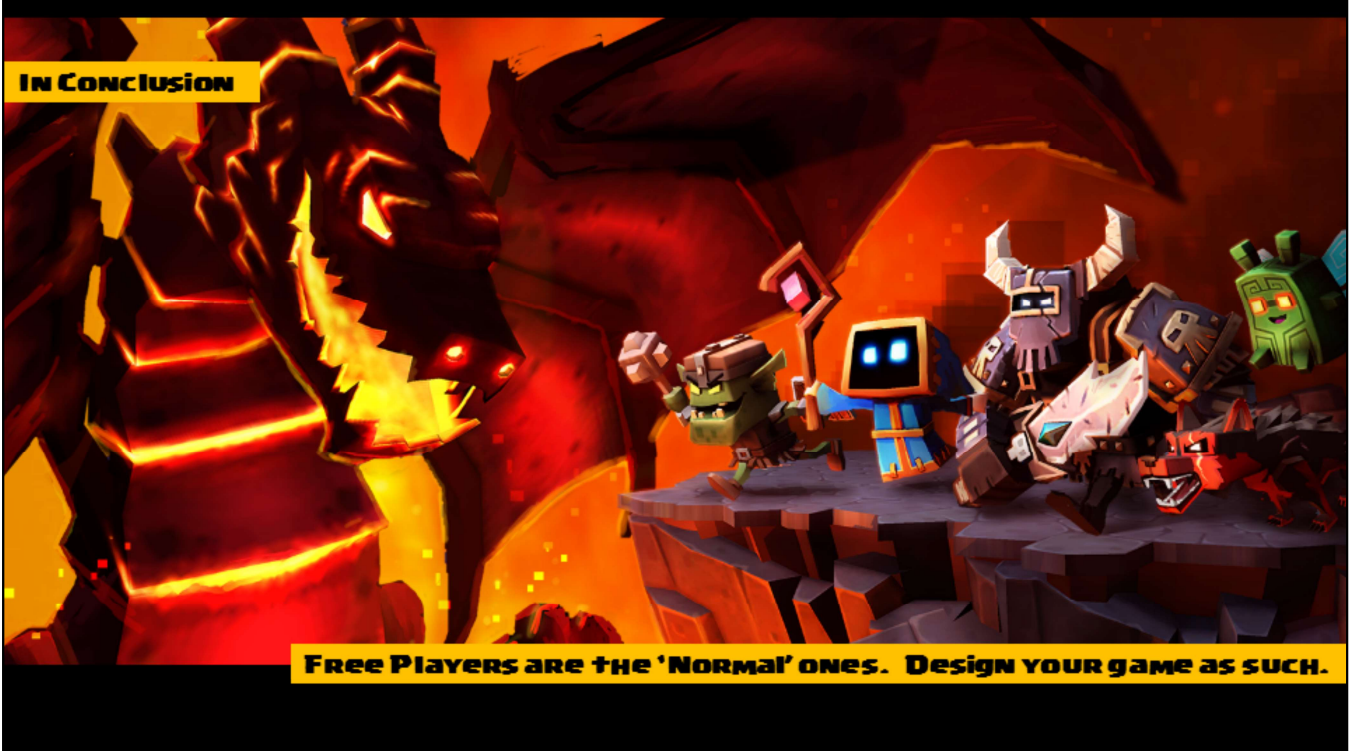
Don't try to make whales. Try to get them to spend a dollar.

Which leads to the last point. If you take all of this together – ensure free players have value to the game, be sure they have a fun time that's not too obnoxious, let them play without spending and then give them a hint of the idea that spending might make them just a LITTLE better and they'd be having a LITTLE more fun if they did, then you have created a game that's ripe for some of those free players to convert.

Now, the vast majority of our money from these games comes from whales – I'm sorry, heavy spenders. But you don't make those overnight. You have to coax them into it. And so, instead of figuring out how to get \$50 bucks from every player, focus on how to get one. Make that spend well worth it. Make it so they want to spend a second, and staircase them up the spend experience.

Now most of them will STILL never spend. And that's normal, and that's fine, get used to it. But if you can get players to spend ONCE, and that spend has obvious advantages, it's much, much easier to get them to ramp up their spend. Still, don't be disappointed if not everyone spends. Remember: Free is your Default. Even the people who spend a dollar are outliers.

IN CONCLUSION



Free Players are the 'Normal' ones. Design your game as such.

DAMION SCHUBERT

*Dungeon Boss,
Star Wars The Old Republic*

**Creative Director
Boss Fight Entertainment
@ZenOfDesign**

[Richard]I think there's a broader story here of being careful about what part of your audience you are addressing.

Even if your game isn't F2P, are you giving the hardcore too much of your attention? Are you worried about people who try to exploit your game when they're in the minority?

Always remember to spend most of your time on most of the players, whoever they may be.

CHRISTINA NORMAN

*Mass Effect 1/2/3,
League of Legends*

**Lead Designer
Riot Games
@truffle**

Next up we have Christina Norman. She was a programmer and then designer at BioWare on the Mass Effect series, and now works on that little game League of Legends

And she's here to say, that problem you think you have? Maybe you don't actually have it the way you think you do.

EMBRACE RADICAL CONSTRAINTS



Lead Designer
Riot Games
@truffle

CHRISTINA NORMAN

Hello, I'm Christina Norman, a Design Leader at Riot Games. I've worked in the industry for 12 years on games like the Mass Effect trilogy and League of Legends.

DESIGNERS HATE CONSTRAINTS

- Time
- Resources
- Technology
- Workflow
- Game Marketing



Lead Designer
Riot Games
@truffle

CONSTRAINTS CAN RESTRICT US

Many designers hate constraints, and that's understandable. Constraints limit what we can do, so it's natural to dislike them.

CONSTRAINTS FOCUS CREATIVITY



*“When forced to work within a **strict framework** the imagination is taxed to its utmost-and will produce its **richest ideas**. Given total freedom the work is likely to **sprawl**.” - T.S. Eliot*

Lead Designer
Riot Games
@truffle

NECESSITY IS THE MOTHER OF INVENTION

But sometimes we do our best work while we're under the strictest limitations. Constraints can inspire us to solve the hardest problems, and to innovate as designers.

THREE CONSTRAINTS SCENARIOS

FRAMEWORK
INTRODUCTION



PROBLEM:
INVENTORY



PROBLEM:
MULTIPLAYER



Game Designer
Riot Games
@truffle

+ FRAMEWORK + TWO IMPOSSIBLE PROBLEMS

Today I'll take you through my constraints framework, and talk about how I use it to solve impossible problems. All of today's examples are real and come from the Mass Effect trilogy.

THREE PANDA NOTATION

HAPPY DEVELOPER
PANDA



Content within constraints

SAD PANDA
DEVELOPER



Frustrated by constraints

SNEAKY
PANDA DEVELOPER



Violating Constraints

Game Designer
Riot Games
@truffle



SIZE OF PANDA INDICATES IMPACT

And I'll be using industry-standard three-panda notation to express how the team felt throughout all this - because constraint management can be stressful. Happy pandas are working comfortably within constraints, Sad pandas feel constraints are limiting them, and Sneaky Pandas are ignoring constraints to solve the toughest problems.

THREE CONSTRAINTS SCENARIOS

FRAMEWORK
INTRODUCTION



PROBLEM:
INVENTORY



PROBLEM:
MULTIPLAYER



Lead Designer
Riot Games
@truffle

+ FRAMEWORK + TWO IMPOSSIBLE PROBLEMS

Let's start with the story of Mass Effect 1

All Possible Games

This diagram is not to scale

All games start with infinite possibilities. Before Mass Effect was Mass Effect, it could have gone in any direction.

All Possible Games



This diagram is not to scale

Like a pure dating sim. I'm sure there's an alternate universe out there where that happened and it's probably pretty cool.

All Possible Games



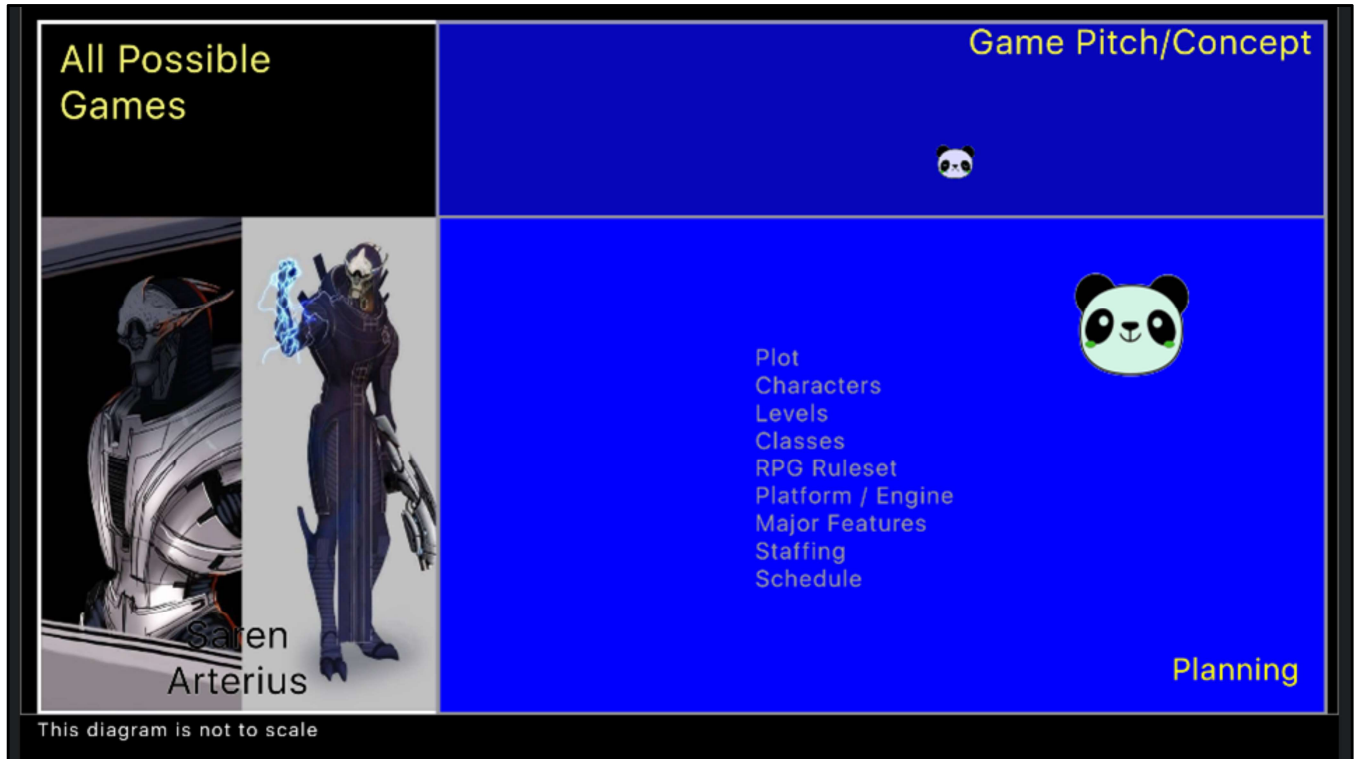
This diagram is not to scale

Game Pitch/Concept

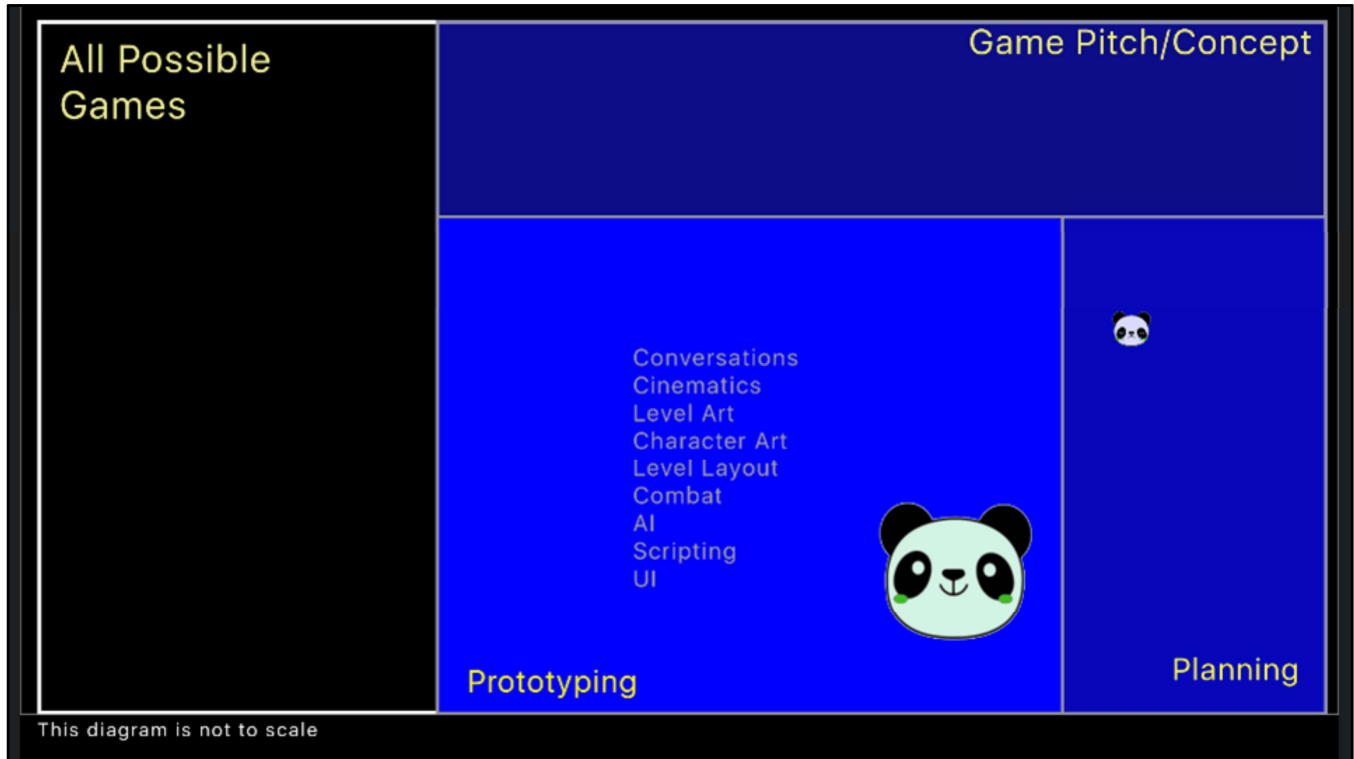


BioWare RPG
Space Opera
Core IP Design
Art Style
Existential Galactic Threat
Explore the Galaxy
Fantastic Sci-Fi Powers
Command a Squad
First Game of Trilogy

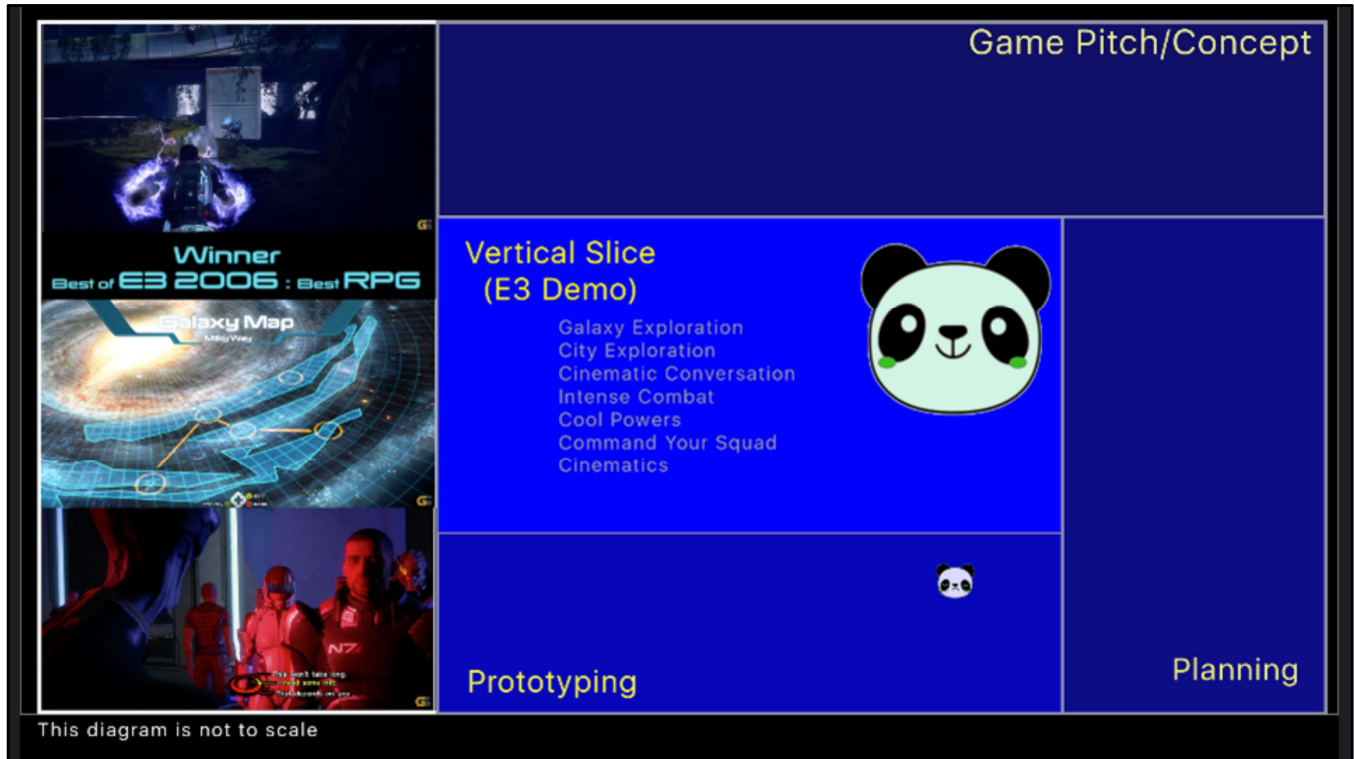
But our team had different ambitions. We wanted Mass Effect to be an Epic Space Opera RPG in which you could explore the galaxy, save the universe, and command an elite squad of super heroes.



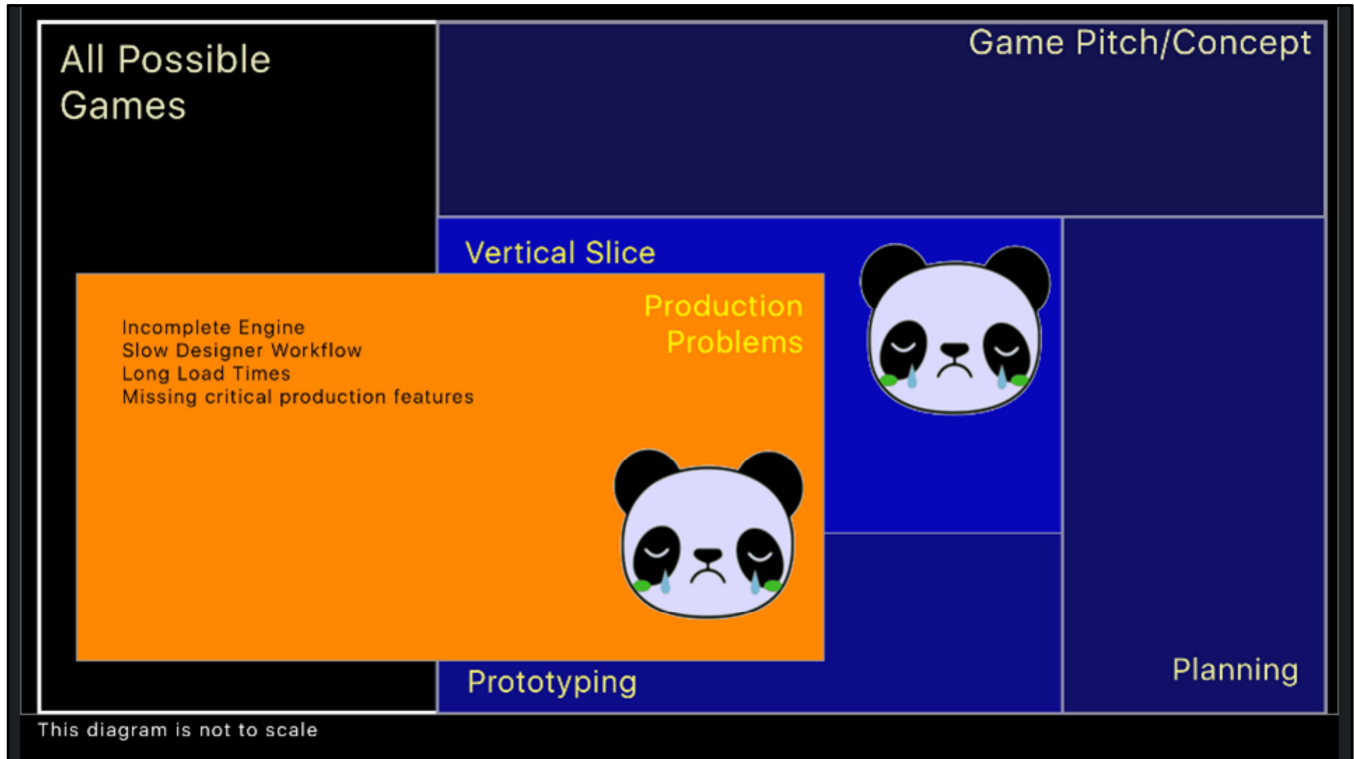
We had a schedule and a budget so we had to get specific. We fleshed out the plot, characters, RPG rules, conversations and gameplay and made some tough calls.



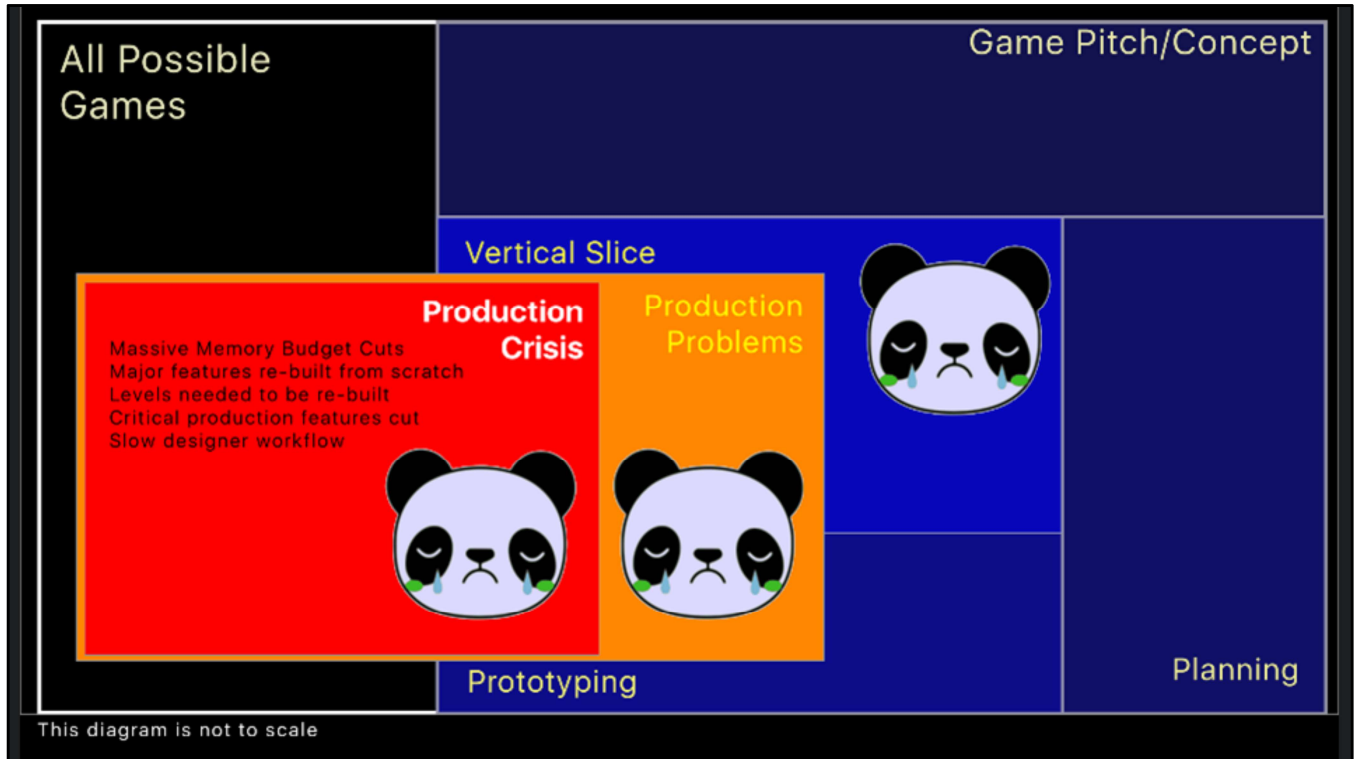
We prototyped key systems, verifying they were both possible and valuable. It was becoming clear what Mass Effect would be, and what it wouldn't be.



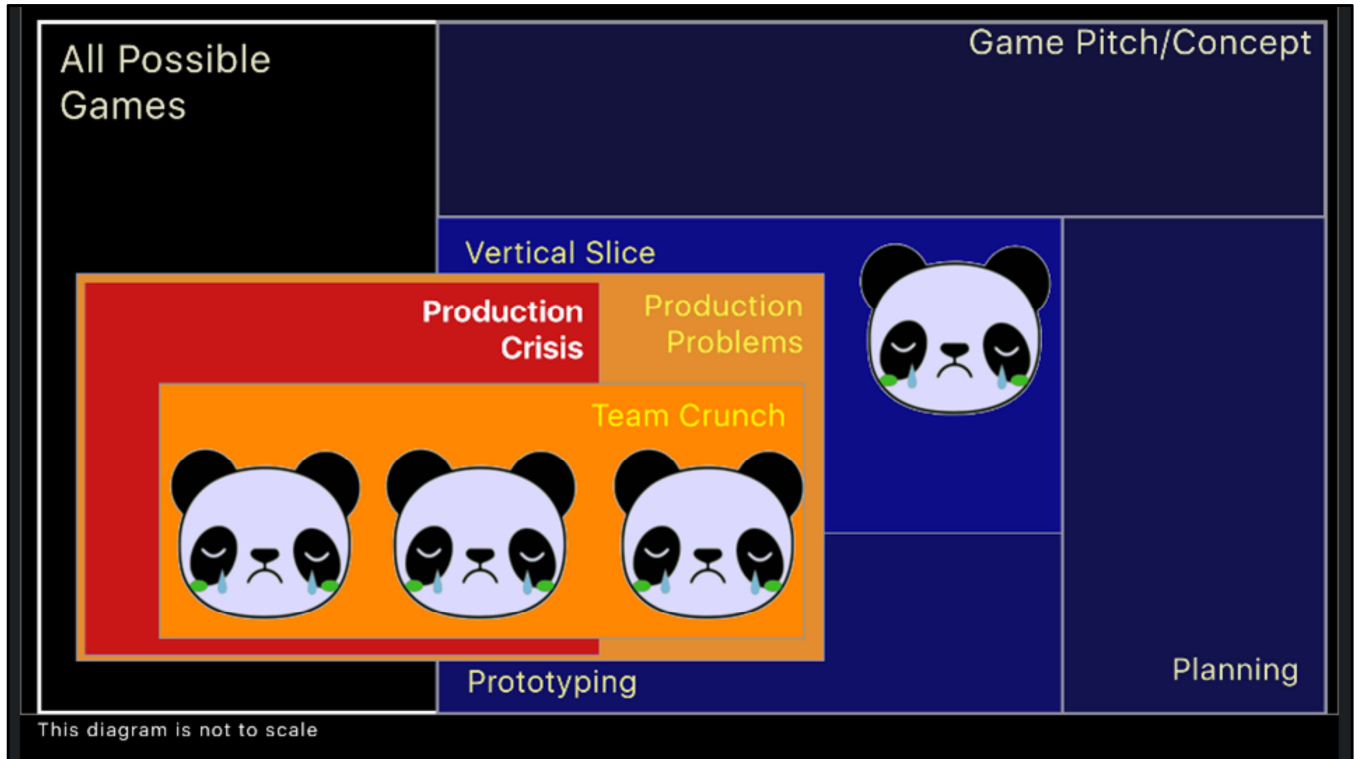
Then E3 happened. We wanted to demonstrate all the key experiences players would have in Mass Effect, like exploration, cinematic conversations, and intense squad combat. This was hard, but the team gave it their all, and we were happy with the results.



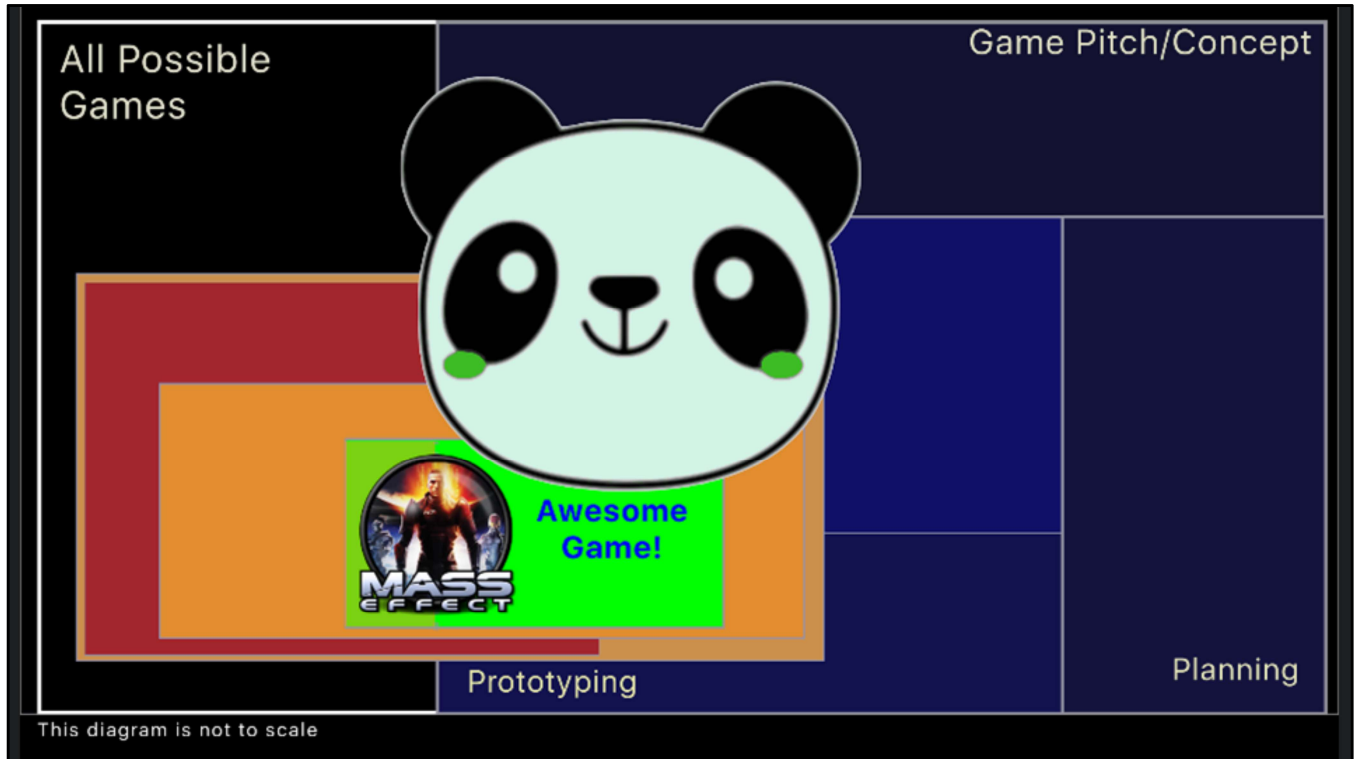
We set out to realize this vision across the entire game but encountered problems executing at scale. Our E3 demo was visionary, but it wasn't real. As hard as we tried we couldn't hit that experience at scale. Engine and workflow problems kept slowing us down. For the first time we felt hampered by constraints that were blocking us from building the game we wanted to make.



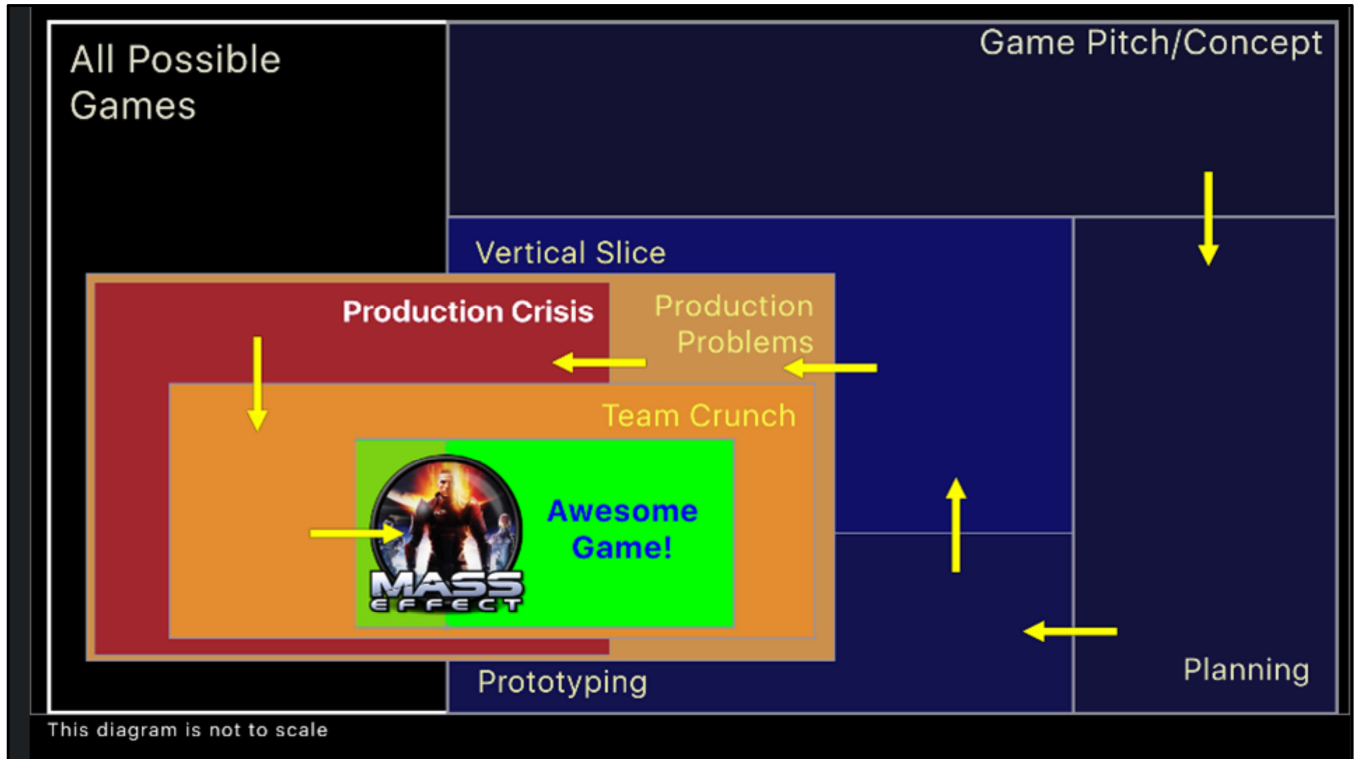
It got worse. We started cutting features and even planets from the game. We had to rebuild existing content and systems to make them more efficient. All this slowed us down. We felt like Mass Effect was slipping away.



So we crunched hard and we crunched long. Through force of will, talent, and focus we dragged Mass Effect back towards where we wanted it to be but the cost on the team was high.



Despite all this adversity we shipped Mass Effect and it was awesome and players loved it. It wasn't everything we wanted, but it was definitely a great game.



Reflecting on the journey from blank slate to Mass Effect, I started to see constraints as a roadmap for progress, and wondered if I could use them to help lead my team towards better games, with fewer sad pandas.

THREE CONSTRAINTS SCENARIOS

FRAMEWORK
INTRODUCTION



PROBLEM:
INVENTORY



PROBLEM:
MULTIPLAYER



Lead Designer
Riot Games
@truffle

+ FRAMEWORK + TWO IMPOSSIBLE PROBLEMS

And we were all focused on making Mass Effect 2 better. We wanted it to be a more refined and polished game, and we wanted to be kinder to ourselves as we built it.

All Possible
Mass Effect 2
Games

This diagram is not to scale

So when we started at the blank slate of Mass Effect 2 we knew we had to address Mass Effect's problems first.

All Possible
Mass Effect 2
Games

PROBLEM: SLOW, CLUNKY, INVENTORY



This diagram is not to scale

And our Inventory was top of mind. It was slow, and clunky, and players used it constantly. It was a clear must fix.

All Possible
Mass Effect 2
Games



Good Inventory System



This diagram is not to scale

So we locked in one constraint early - we'd have a good inventory system.

All Possible
Mass Effect 2
Games



Good Inventory System

Inventory Design

Not Enough Programmers



This diagram is not to scale

And we designed a great inventory system, but we were rebuilding our UI workflow and technology from scratch, and we lost some UI programmers. It became clear that the inventory we wanted was not possible.

All Possible
Mass Effect 2
Games



This diagram is not to scale

Good Inventory System

Lower Scope Inventory

Not Enough Programmers



Inventory Design

Not Enough Programmers

So we designed a lower scope inventory, but that wasn't possible either.



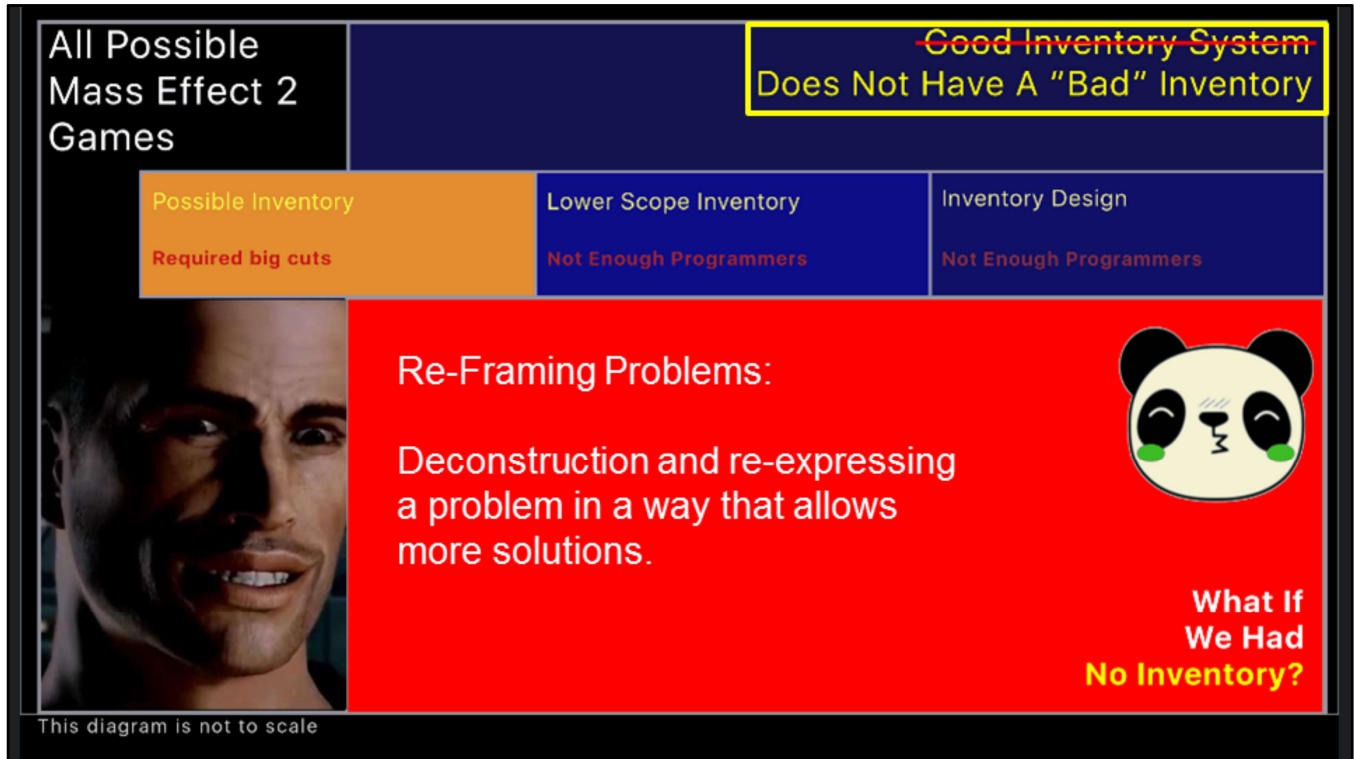
And then we designed a possible inventory, but it wasn't that good. Plus we'd have to cut some great things from the game to even build it. No one was happy with that.



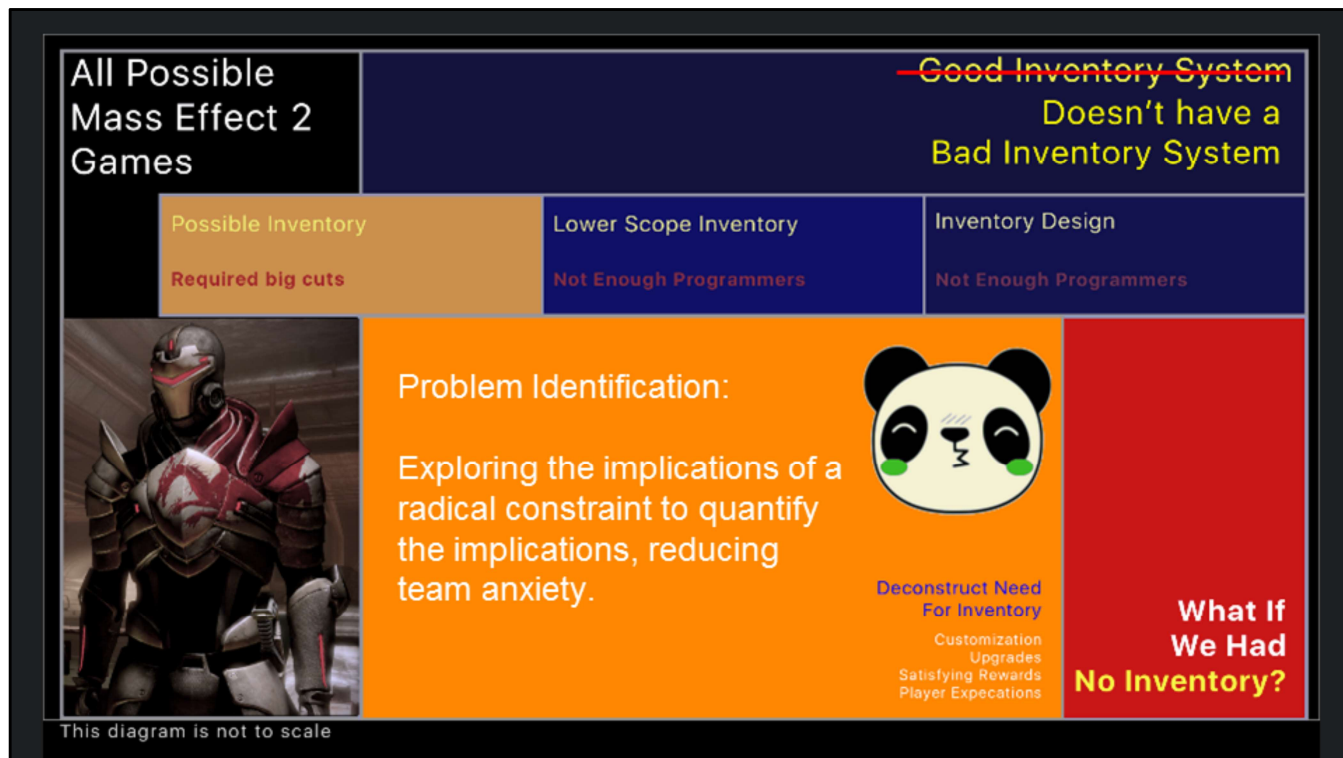
But then I had an idea - if we had to cut something - why not cut the inventory itself?



I proposed this to the team and they were confused. Everyone knew Mass Effect was a BioWare RPG, and BioWare RPGs have inventories. Wasn't the whole point to make sure we had a good inventory?



I proposed that what we really wanted was to not have a bad inventory system. Re-framing the problem this way opened the team's mind.



But people were skeptical - what would Mass Effect without an inventory even look like? Would it still be an RPG? Would it still be fun? These were important questions to answer, so we went deep, deconstructing why we needed an inventory, and identifying the problems we'd have to solve if we removed it.



One by one we proposed solutions to those problems, building confidence in the new direction until we had buy-in from the team.



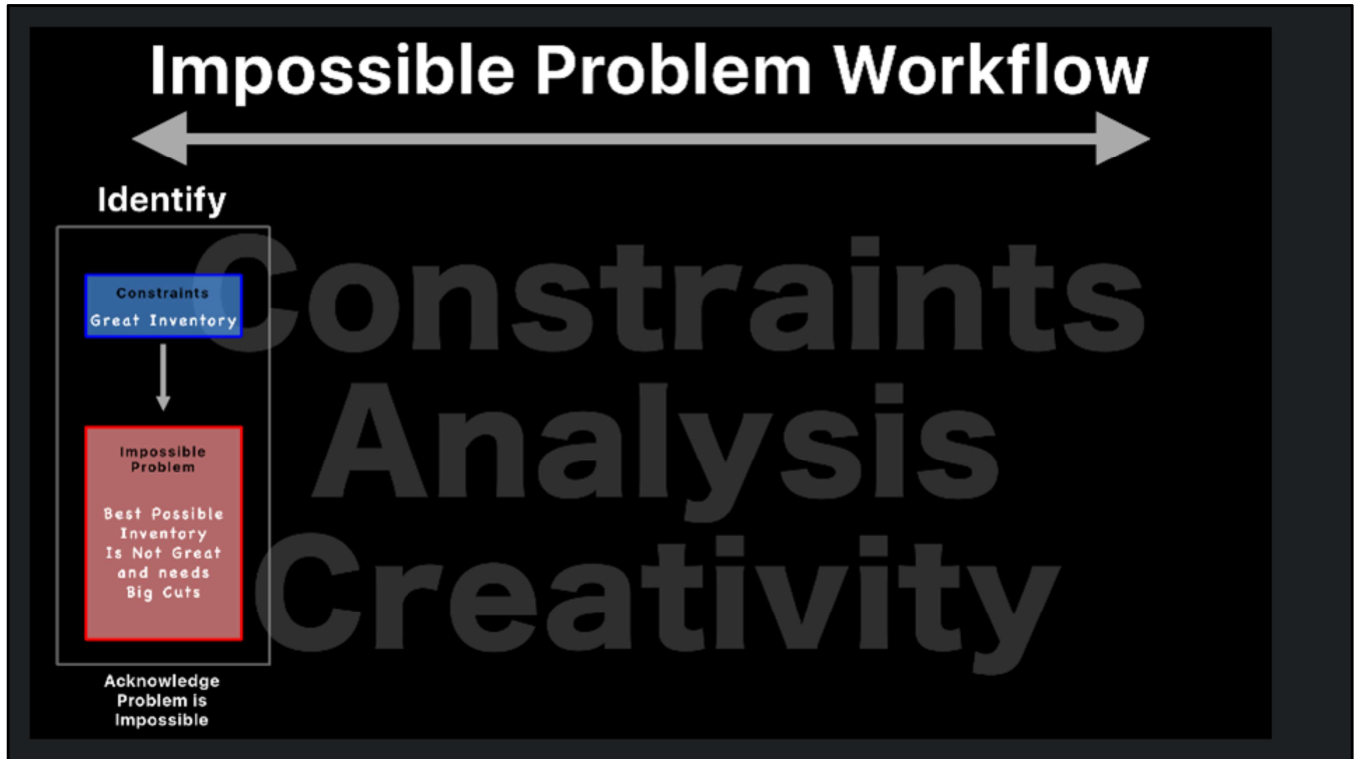
As a result Mass Effect 2 didn't have an inventory and it was a much better game. By going outside of our comfort zone we solved an impossible problem, and unlocked a new line of thinking that lead to a more focused and polished Mass Effect.

Impossible Problem Workflow

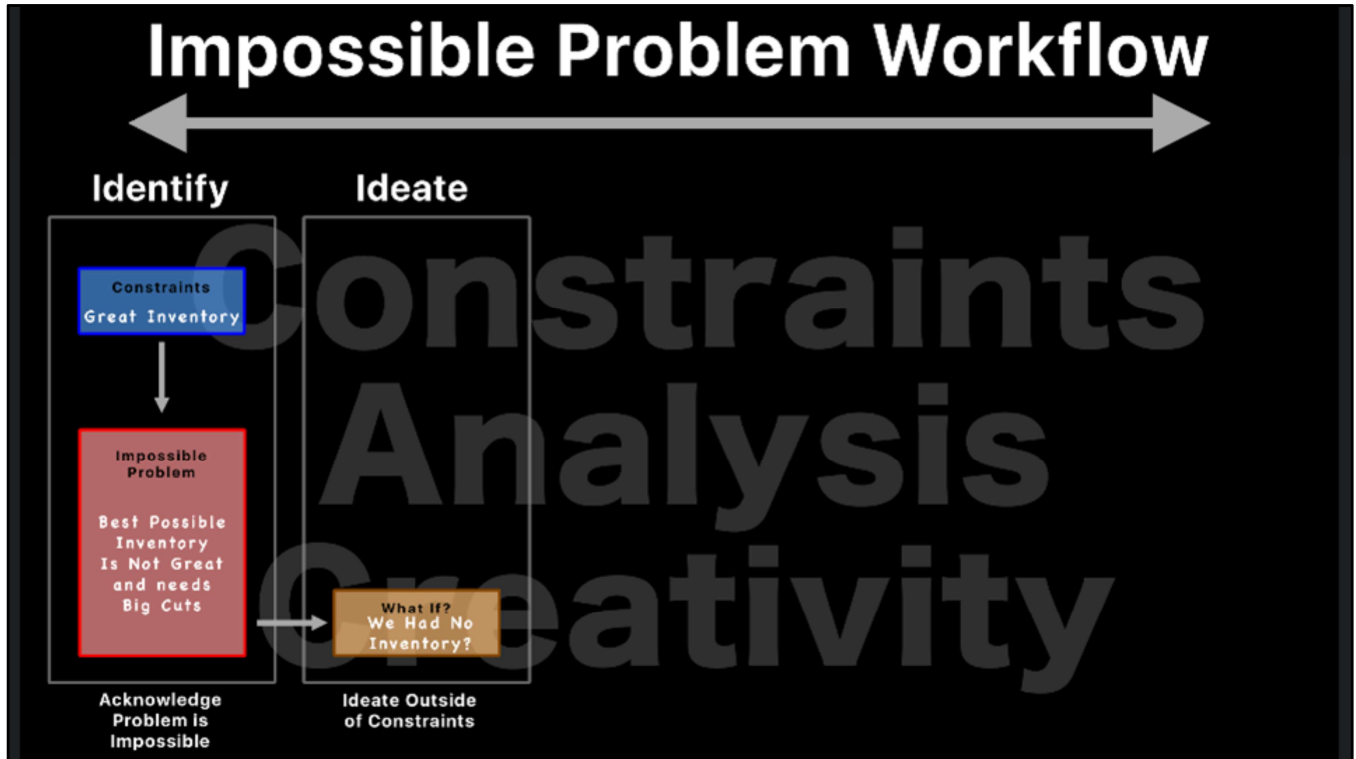


**Constraints
Analysis
Creativity**

I wanted to generalize what we had done, and see if it could be applied consistently. Expanding on my constraints framework, I looked at how different modes of thought had been helpful in solving problems in different ways.

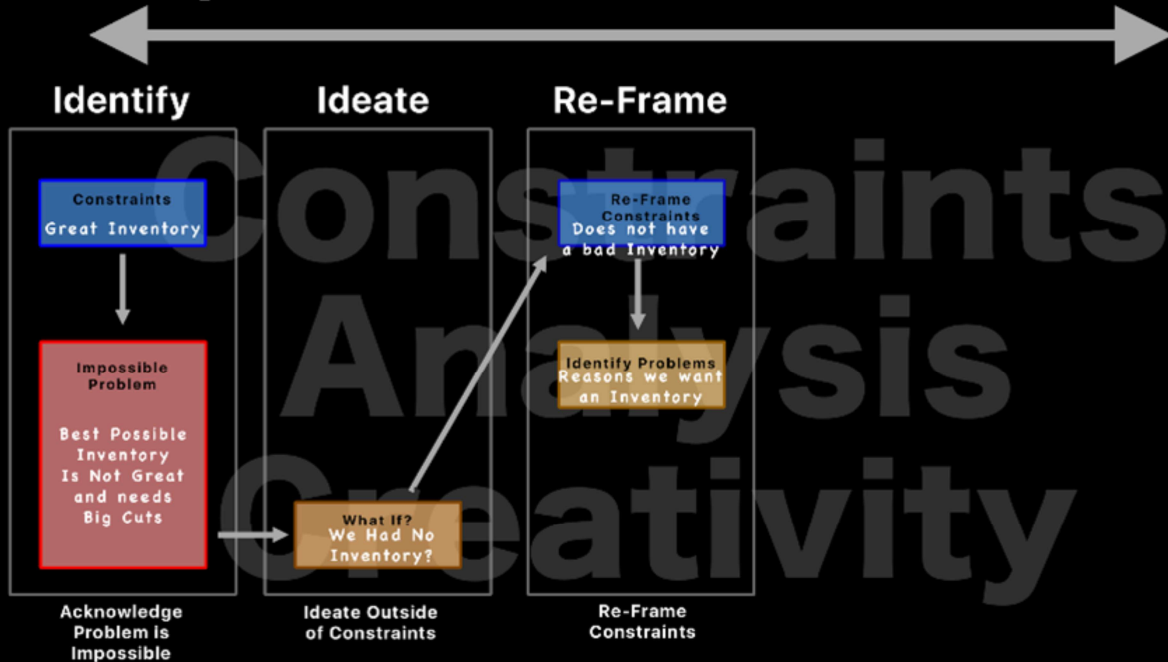


The first step was identifying an impossible problem. Teams don't like to admit when they are blocked. But when repeated attempts to solve problems creatively, and analytically, are failing it's important to acknowledge that you are not trending to success. This helps the team open their mind up to new ideas.



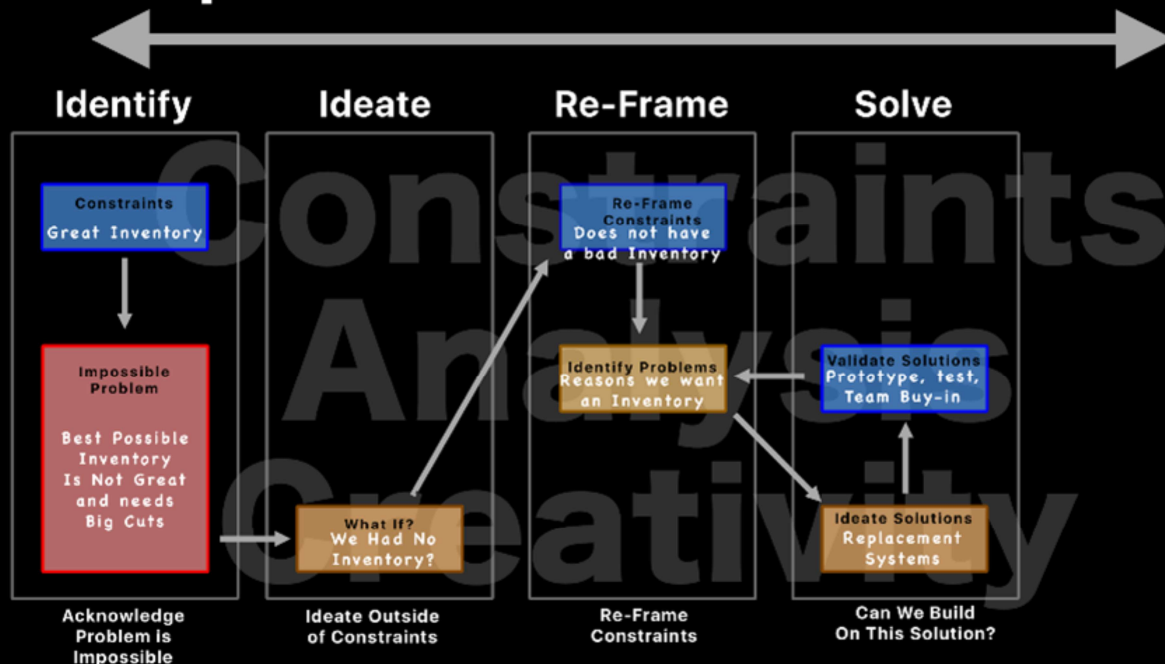
Ideation comes next, and must be free of existing constraints, or even analysis of implications. No buzz-kill pandas allowed. Dangerous ideas must be allowed, and even encouraged.

Impossible Problem Workflow

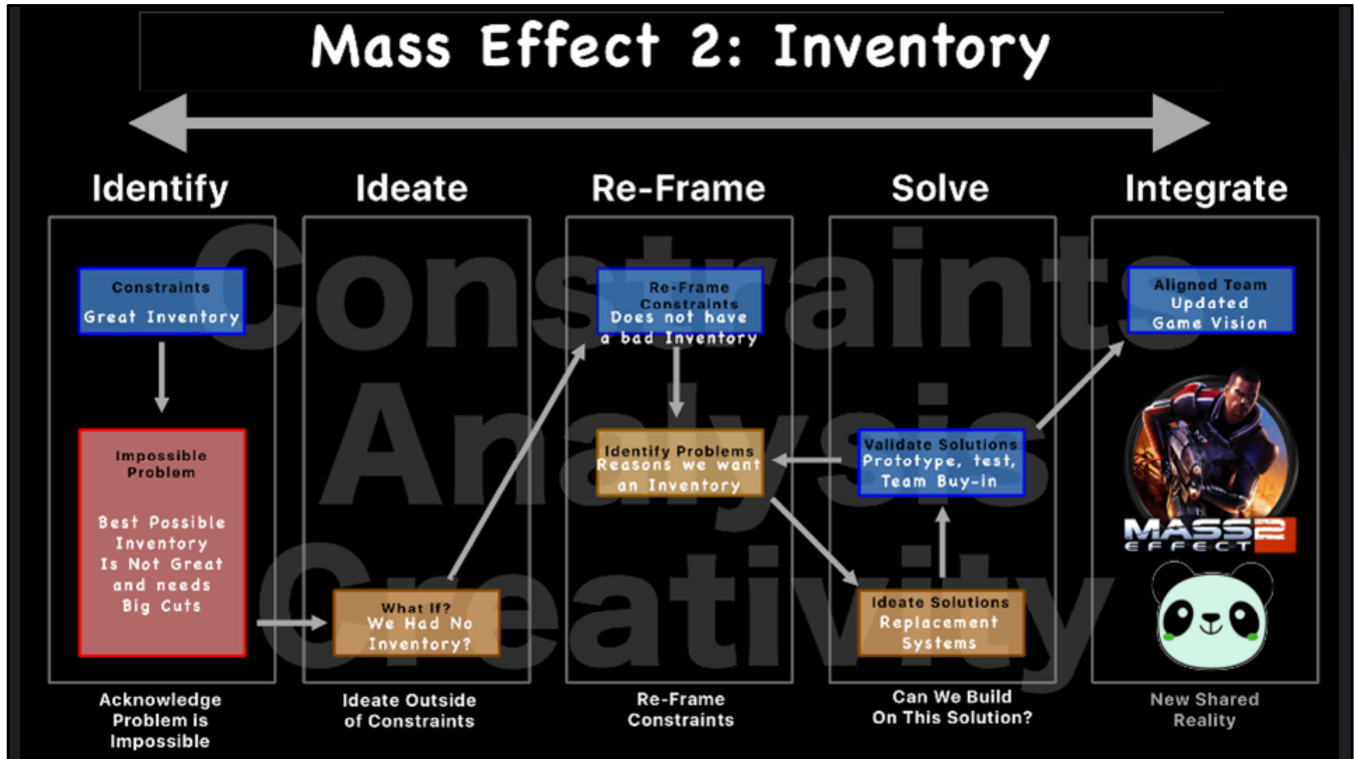


When a promising idea is identified, it's time to re-introduce constraints, re-framing them so the proposed solution is possible. This will raise concerns which should be noted. Additional analysis should be employed to identify other problems. The implicit contract is solving these problems will enable us to move forward in a new direction.

Impossible Problem Workflow



Next you prioritize and solve quickly. With each problem solved you reduce stress in the team, and improve buy-in. The focus is on quickly showing progress, over building shippable code and content.



Once viability is clear enough it's time to seek acceptance so you can return to development. Your team should be happy pandas, excited to work within new constraints.

THREE CONSTRAINTS SCENARIOS

FRAMEWORK
INTRODUCTION



PROBLEM:
INVENTORY



PROBLEM:
MULTIPLAYER



Lead Designer
Riot Games
@truffle

+ FRAMEWORK + TWO IMPOSSIBLE PROBLEMS

I was eager to try this framework which meant I needed an impossible problem to solve. Luckily game dev is full of those.

All Possible
Mass Effect 3
Games

This diagram is not to scale

Coming off the success of Mass Effect 2 we wanted to do something big for Mass Effect 3.

ME3 MULTIPLAYER SCENARIO



Lead Designer
Riot Games
@truffle

AN IMPOSSIBLE PROBLEM

Multiplayer was big, and it was something we'd always wanted to do. At the same time we knew it couldn't feel tacked on. It had to be good. The team was nervous but excited by this ambitious goal.

All Possible
Mass Effect 3
Games



Successful Online Multiplayer
Competitive Market
Audience has Low MP Engagement

This diagram is not to scale

Our initial analysis proved troubling. All our data and surveys said multiplayer was a winner take all market dominated by big PvP games. Even worse, BioWare players had lower engagement in multiplayer than most console gamers, and surveyed players weren't that excited by the idea of multiplayer Mass Effect.

All Possible
Mass Effect 3
Games



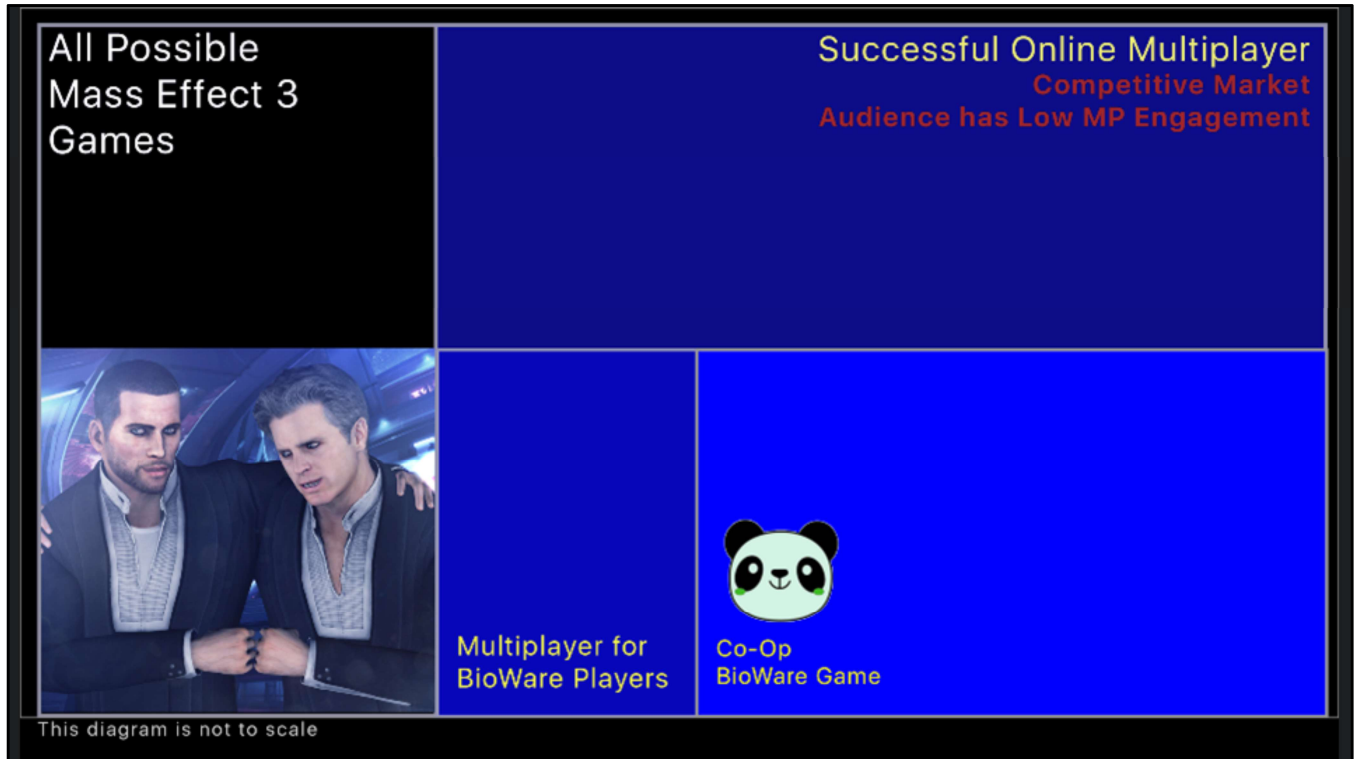
Successful Online Multiplayer
Competitive Market
Audience has Low MP Engagement



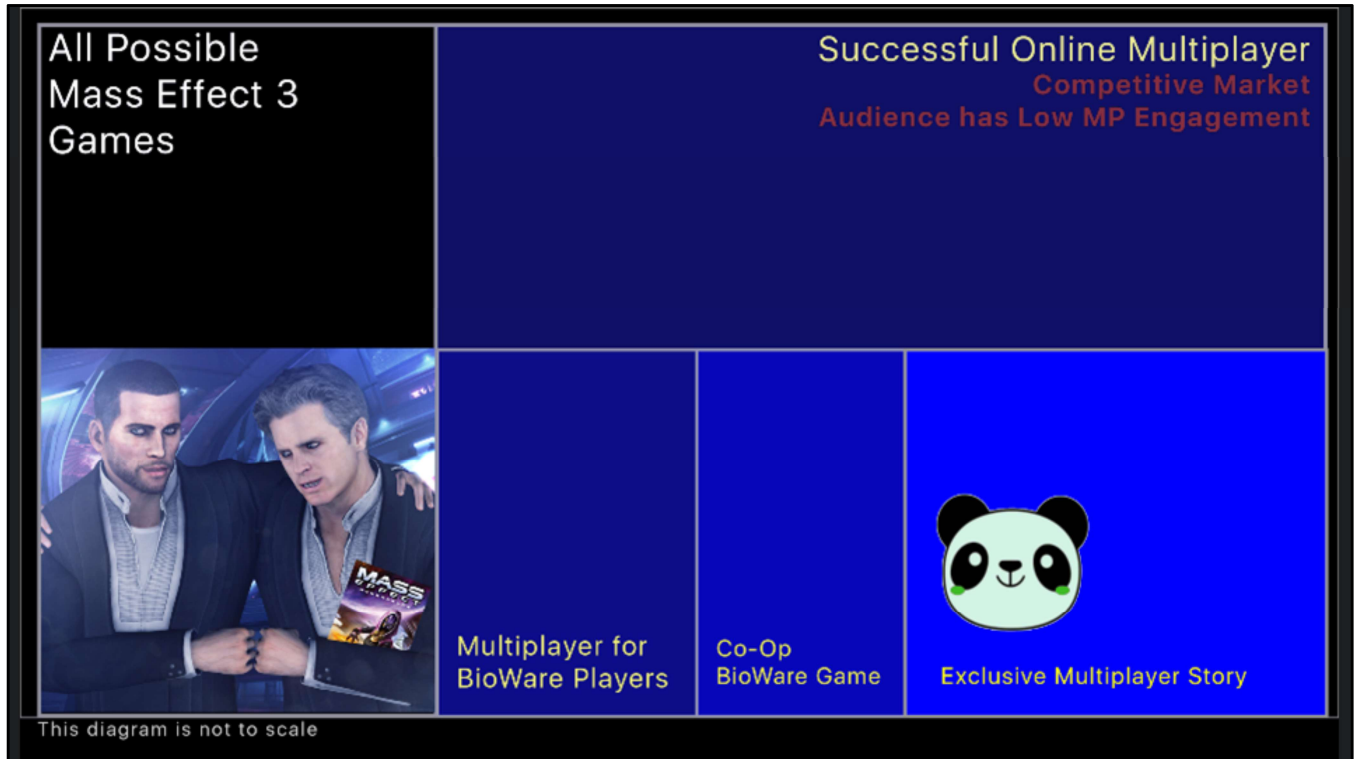
Multiplayer for
BioWare Players

This diagram is not to scale

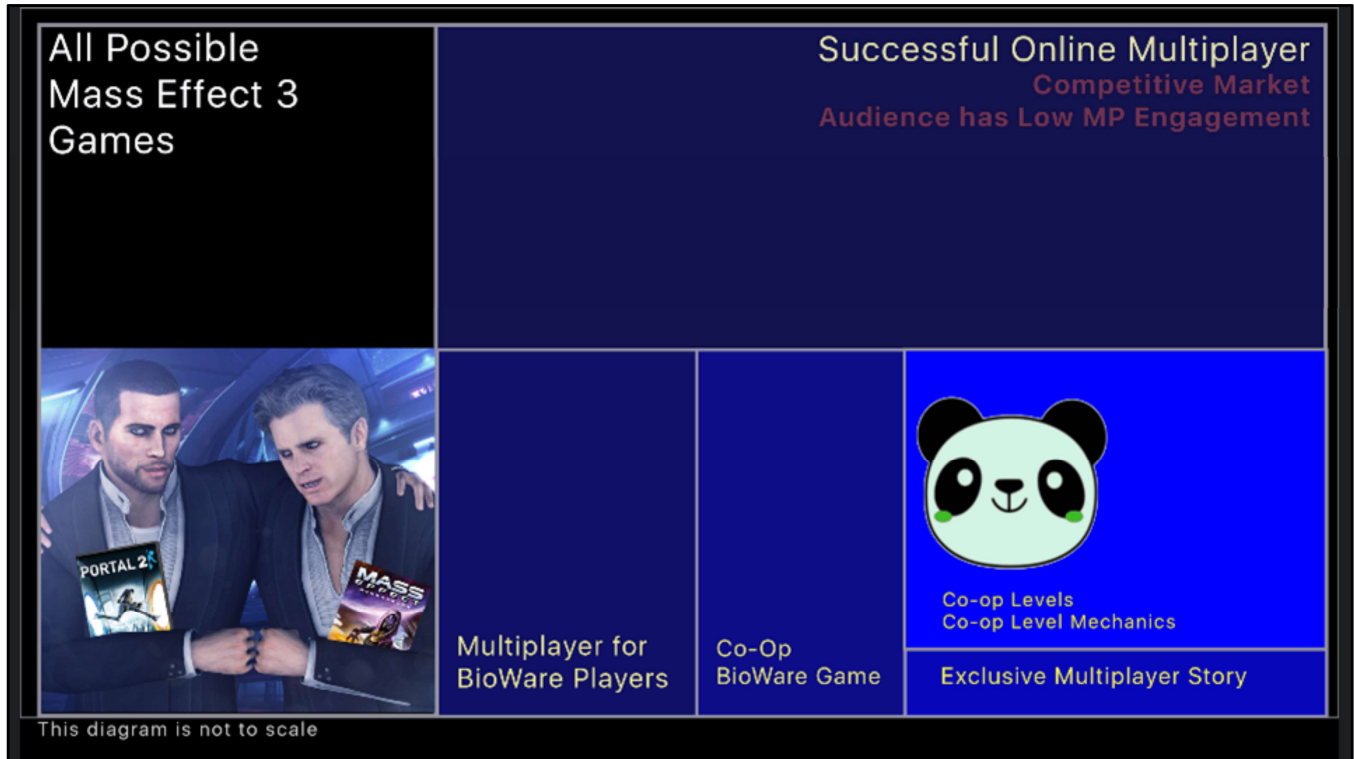
So to make successful high quality multiplayer, we knew we had to build a different kind of multiplayer. BioWare multiplayer.



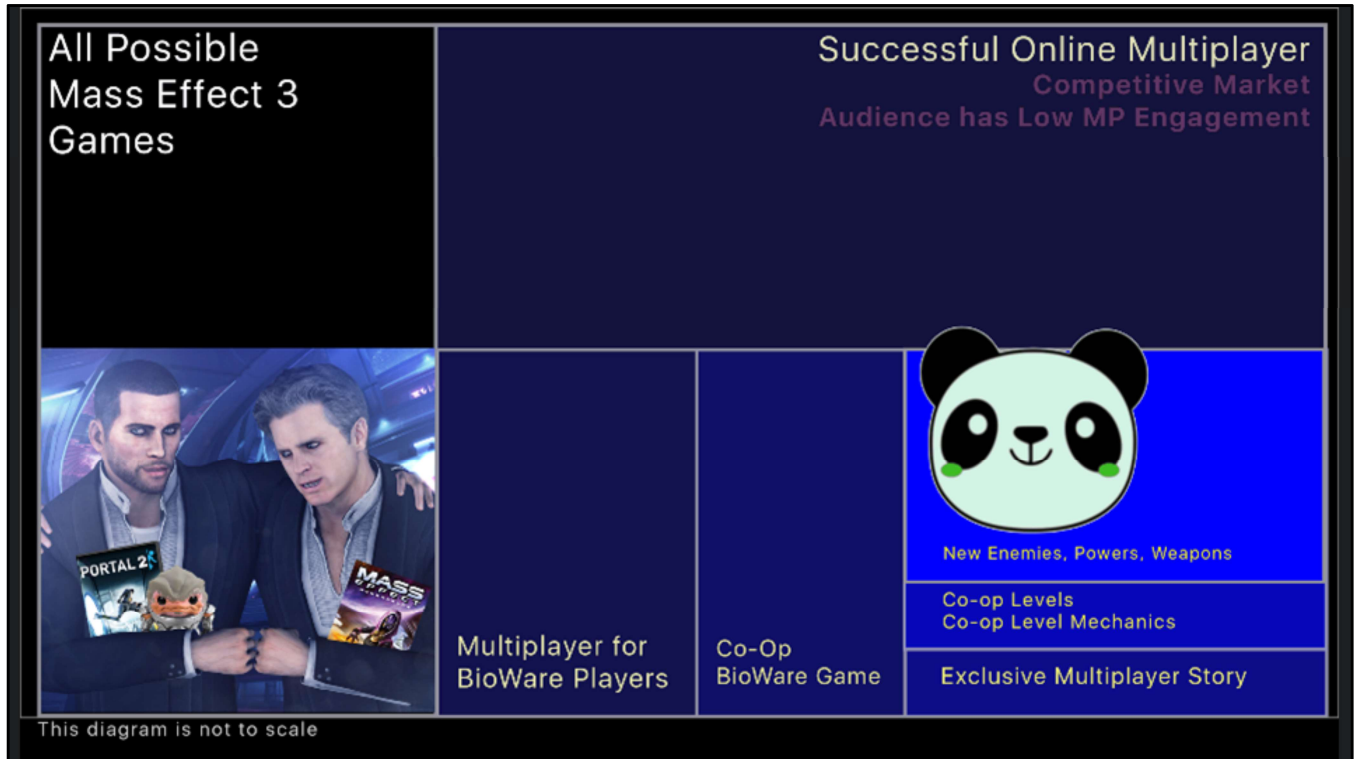
That meant a co-op game with everything BioWare players love.



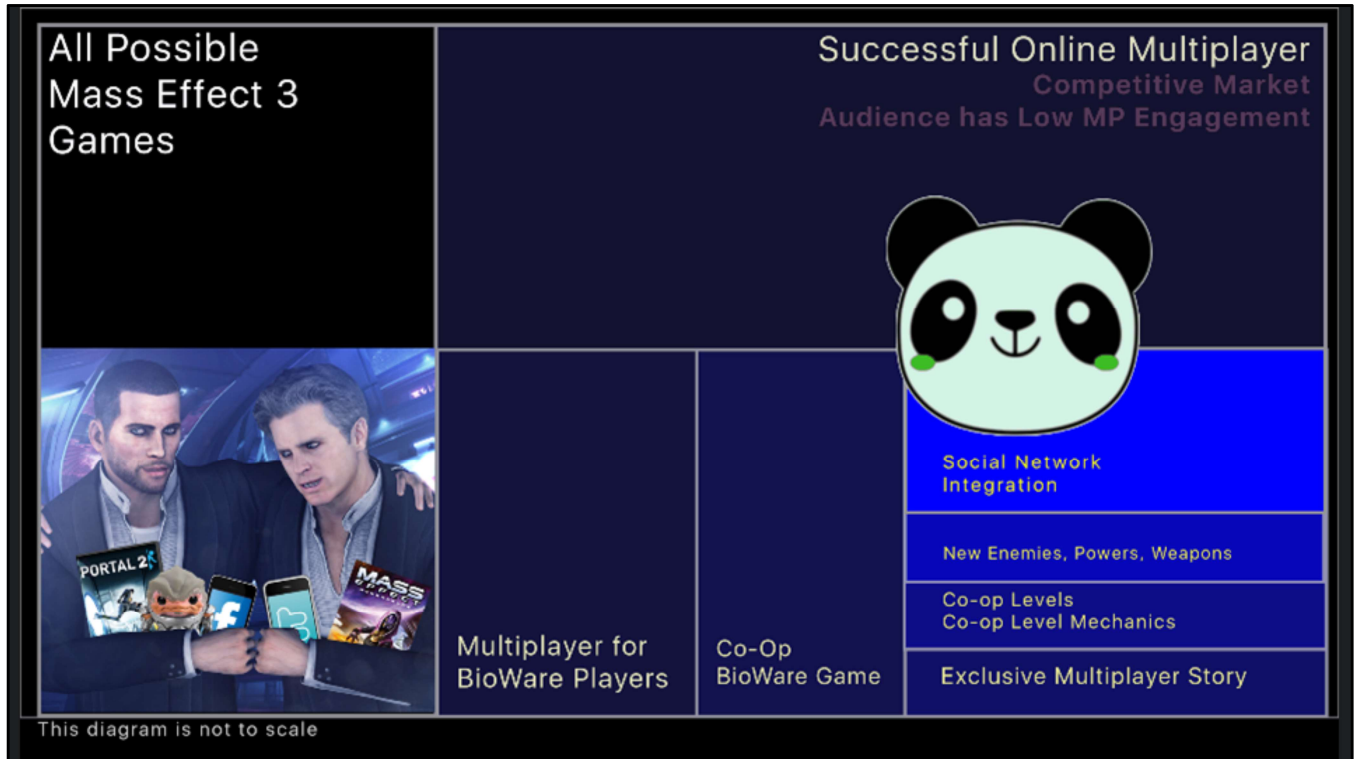
Like an exclusive story



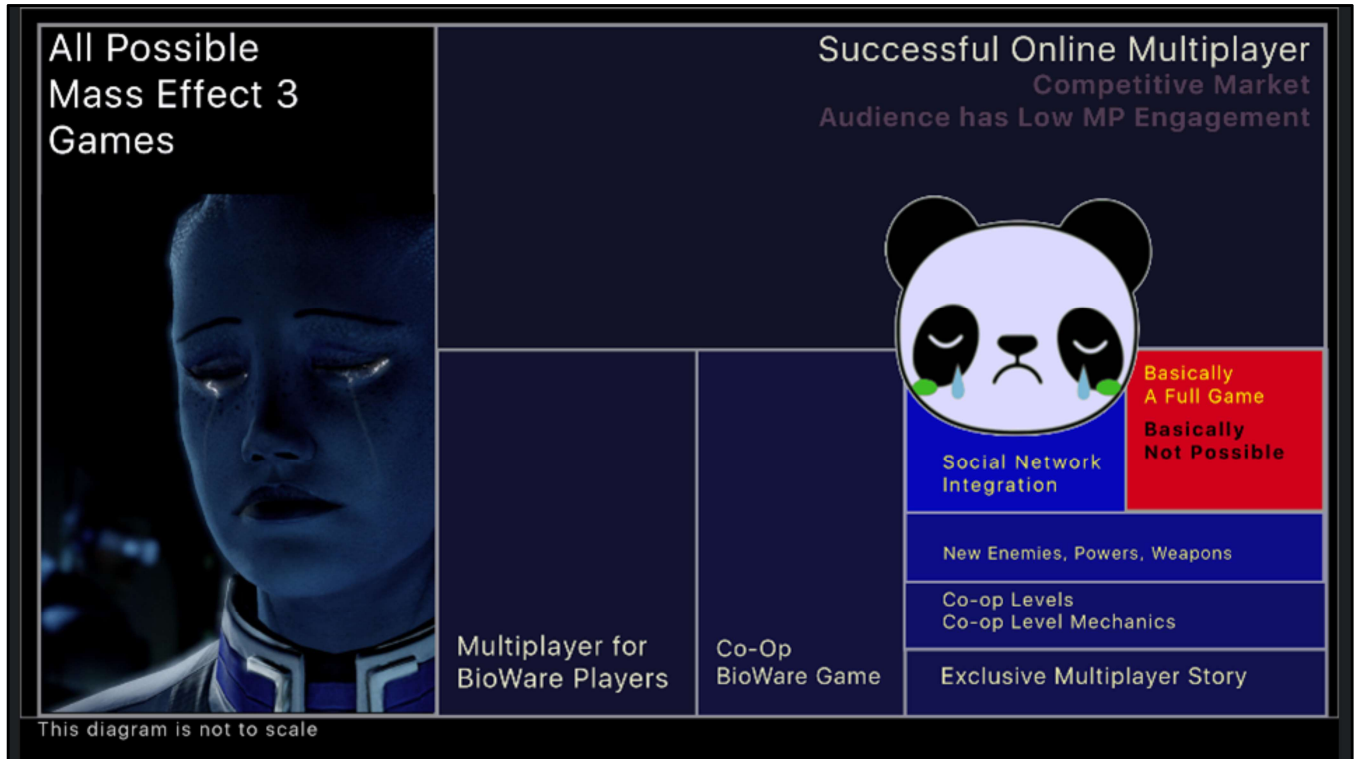
Co-op level mechanics, and levels you couldn't cheese solo.



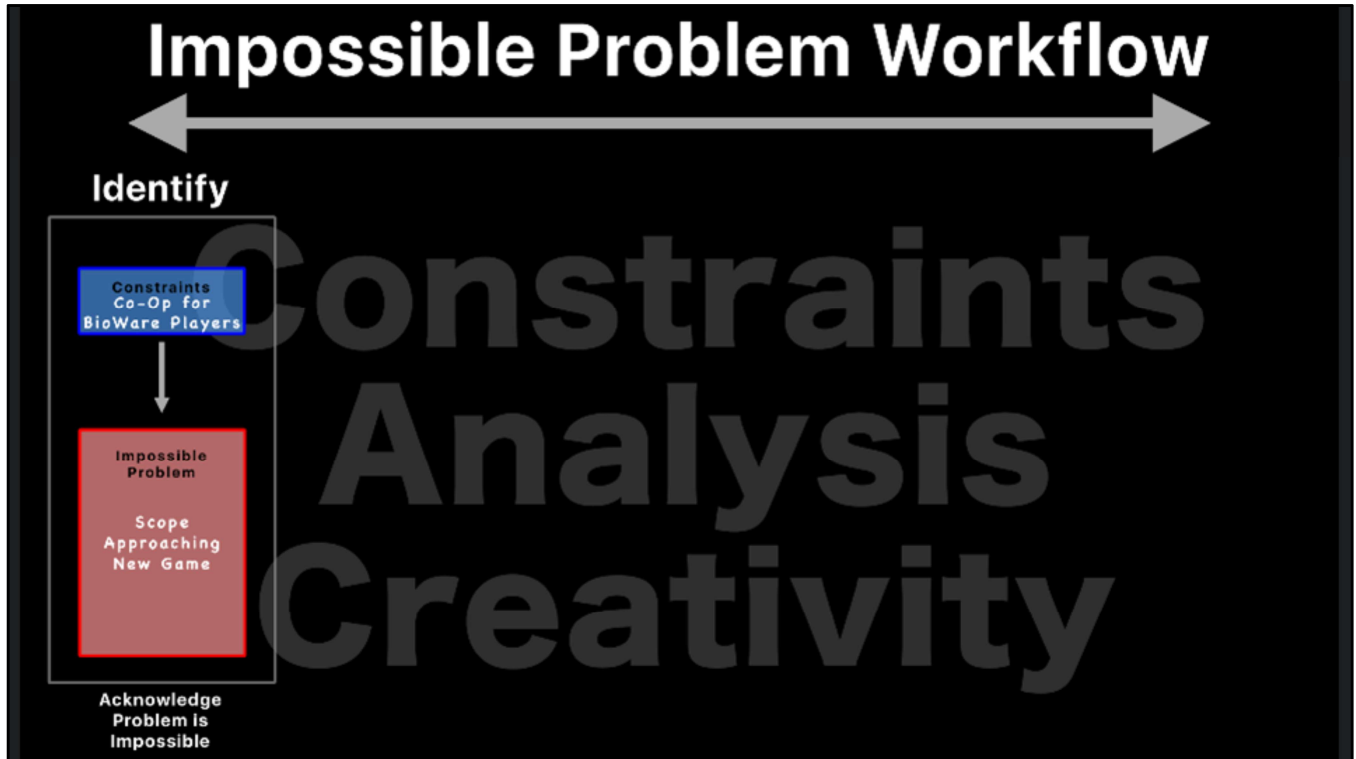
New enemies, powers, and weapons



Plus social network integration so recruiting your friends would be easy.



At this point multiplayer's scope was starting to look like a full game. This was cool, but it wasn't feasible Mass Effect 3 multiplayer. We had another impossible problem, and a great chance to test my framework.



So we acknowledged that our current multiplayer direction would not work. The scope was too high to ship it even with a dedicated multiplayer team.

Impossible Problem Workflow

Identify

Constraints
Co-Op for
BioWare Players

Impossible
Problem

Scope
Approaching
New Game

Acknowledge
Problem Is
Impossible

Ideate

What if co-op
Mass Effect is
super fun?

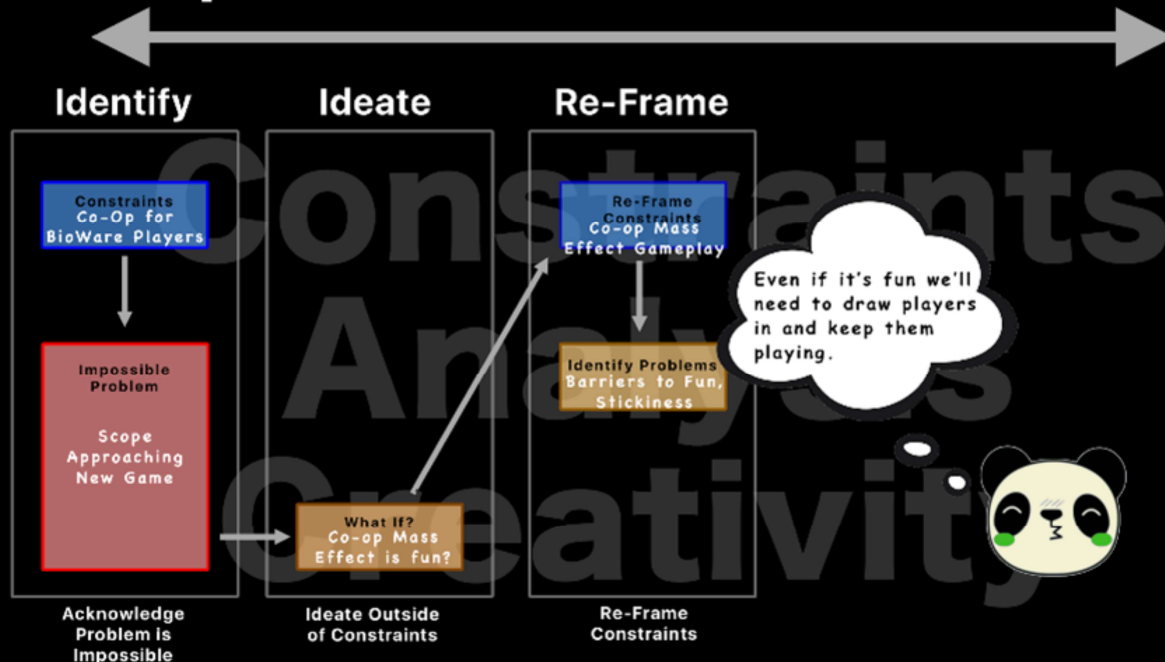
What If?
Co-op Mass
Effect is fun?

Ideate Outside
of Constraints

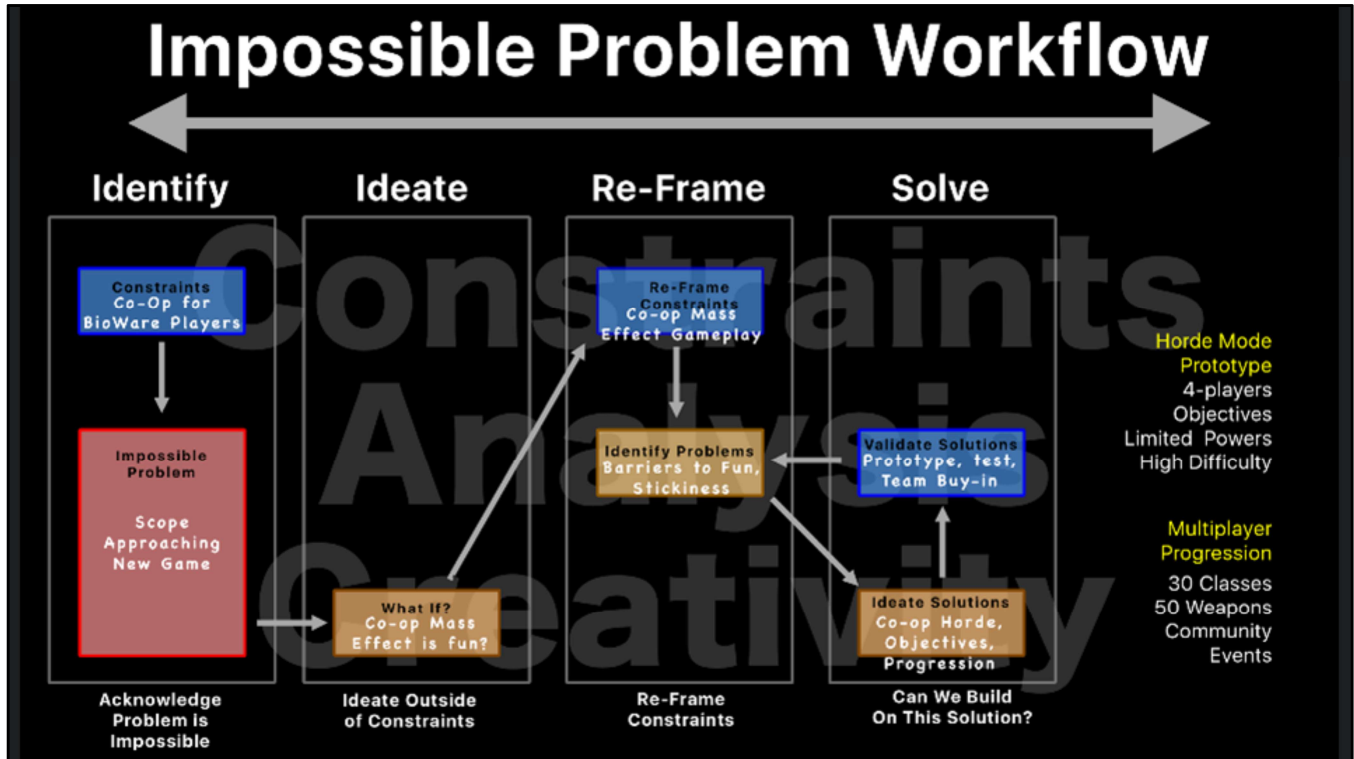


We ideated outside of constraints. What if Mass Effect's gameplay + co-op was really fun? What if we didn't need a new story, enemies, powers, level mechanics, or even new weapons to make it successful?

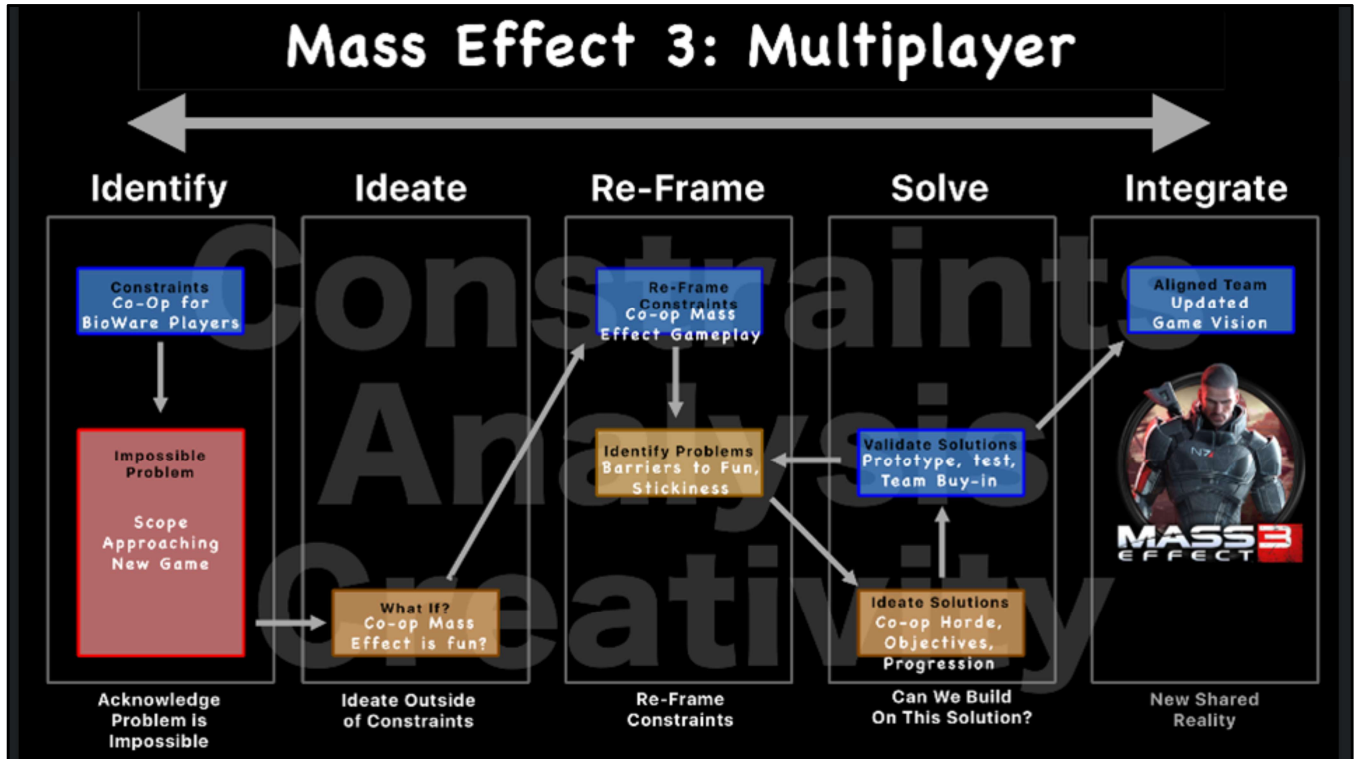
Impossible Problem Workflow



So we re-framed our problem and asked the question - if co-op mass effect was really fun, what issues would we still need to address? We noted those for later.



We prototyped co-op horde mode, and verified it was fun. Based on playtesting feedback we added objectives, limited power sets, and high difficulty. Then we iterated through our problem list, designing progression systems to ensure players had long term goals to motivate engagement.



With a strong prototype, a clear problem list, and designs for engaging progression systems we got buy-in from the team, updated our vision and started moving forward.



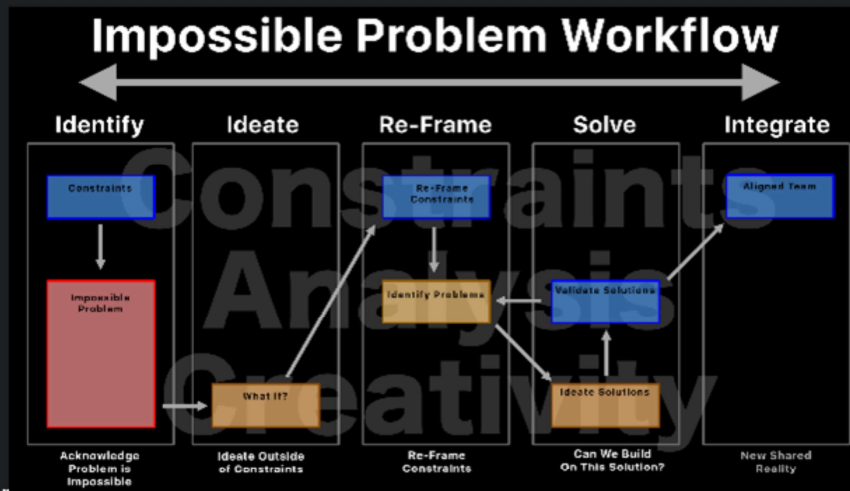
Shortly after this I left BioWare to join Riot, but the Mass Effect Multiplayer team took their prototype and continued to develop it into Mass Effect 3's multiplayer. That same team went on to build Andromeda and I can't wait to check it out.



Players spent more hours in Multiplayer than in Single Player

And just to be clear players loved Mass Effect 3's multiplayer. Not only did they play it, they spent more time collectively in multiplayer than they did in single player. For the third game in a single player franchise that's just crazy.

EMBRACE RADICAL CONSTRAINTS



Lead Designer
Riot Games
@truffle

CHRISTINA NORMAN

Since then I've continued to develop and apply this framework at Riot, using constraints, analytics, and creativity to solve really tough problems. Now it's your turn. Thanks so much for listening, it was awesome sharing my framework and story with you today, I hope this inspires you to solve your impossible problems.

CHRISTINA NORMAN

*Mass Effect 1/2/3,
League of Legends*

**Lead Designer
Riot Games
@truffle**

[Richard] For me this one comes back to remember that sometimes we need to come down to rethinking your true motives.

"We've always done it that way?"

"Yes, but why?"

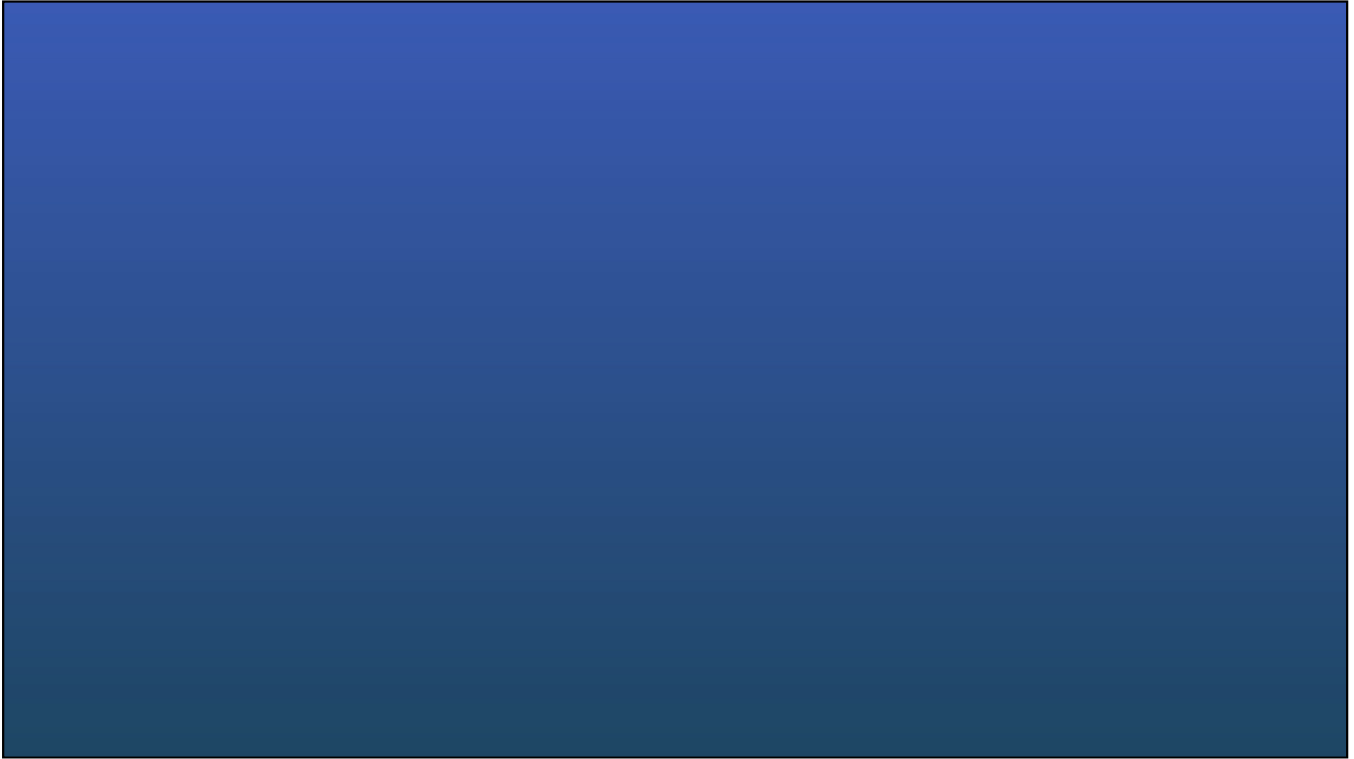
HAL BARWOOD

***Indiana Jones & The Fate of Atlantis
Indiana Jones & The Infernal Machine***

**Writer / Designer
Finite Arts
www.finitearts.com**

Our next speaker is unique in that he was one of the first inspirations for doing this series of talks.

Beyond working on multiple classic games in LucasArts' heyday and the most famous games Indiana Jones has ever appeared in, Hal first did some of the first GDC talks about rules



There are a lot of slides in the following deck. Don't be alarmed — I'm not using PPT's built-in transitions — most of the slides will fly by in an instant as deliberately staccato animations



My rap is about exploiting the possibilities with game characters, so I have a little character here to obey my own rule and personify its explication





QUICK HISTORY



A brief look back . . .



I delivered the first talk on game design rules 16 years ago @ GDC 2001 . . .



I thought designers were already using rules, but in a loose, intuitive, inarticulate way. I imagined that there might be as many as 400 rules sitting out there unexpressed, and I proceeded to examine four that I use. (For the record, those rules — not to be discussed here — were: *Fight Player Fatigue*; *Maximize Expressive Potential*; *Maintain Level of Abstraction*; *Concretize Ideas*.)

QUICK HISTORY



All creative endeavors — visual, verbal, musical, technical, you name it, are so complex they defy perfect understanding

QUICK HISTORY

arts have their rules



Graphics, Writing, Music, Architecture — as a result of their complexity, most arts have generated rules of thumb for guidance.

QUICK HISTORY

arts have their rules
most artists use them



And most creative people need and use them!



QUICK HISTORY
arts have their rules
most artists use them

but I noticed game designers
DID NOT (seem to, anyway)

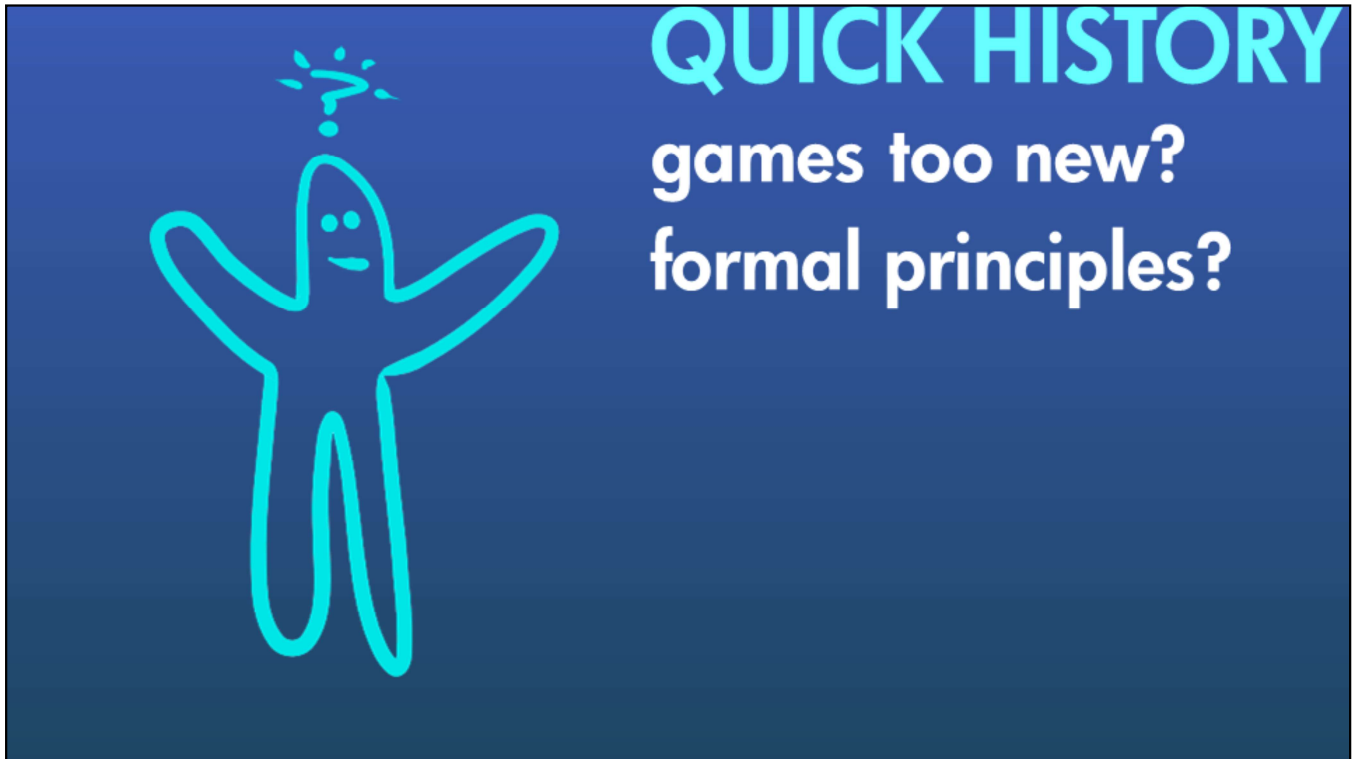
But game designers — at least in bygone days — did not seem to participate.

QUICK HISTORY





Do game designers rely instead on a set of formal principles (as a faction of designers and coders have often advocated) ?



Do game designers rely instead on a set of formal principles (as a faction of designers and coders have often advocated) ?



Or is the attitude just a crusty refusal to submit? The creative process demands freedom! We don't need no stinking rules!

QUICK HISTORY



QUICK HISTORY

rules exist



I thought (and still think) the answer is pretty simple and lucid — rules exist, all right, and there are many . . .



But most of these rules were (and still are) implicit, intuitive — unexpressed, undeclared, unpublicized.



QUICK HISTORY

rules exist
unstated
unreliable !

They are foggy notions at best, and therefore they are unreliable guides.

QUICK HISTORY



QUICK HISTORY



QUICK HISTORY

we're still identifying
rules 16 years later !



When I look back, I think I was on the right track — because, here we are in 2017, still talking about the problem.



162



Let's look at a few of the rules we collected . . .



Savegame systems can be an afterthought; they can require a lot of effort, and so they are sometimes neglected, to the pain and rage of players who are forced to trudge repeatedly through long tracts of a game because they screwed up. Bad savegame systems are barriers, and good ones are big promoters of goodwill.



☀ let the player turn the game off

☀ add a small amount of randomness to AI calcs

Mere décor, for example, should look slightly different from, and more inert than, items that can be picked up, which should all share in turn a graphical emphasis



☀ let the player turn the game off

☀ add a small amount of randomness to AI calcs

☀ things that look alike should behave alike

Mere décor, for example, should look slightly different from, and more inert than, items that can be picked up, which should all share in turn a graphical emphasis



good rules



These rules, like most of the other 100 or so in the 400 Project collection, are perfectly good rules . . .

good rules

small rules



However, they're small rules, also like most of the rest we have collected.

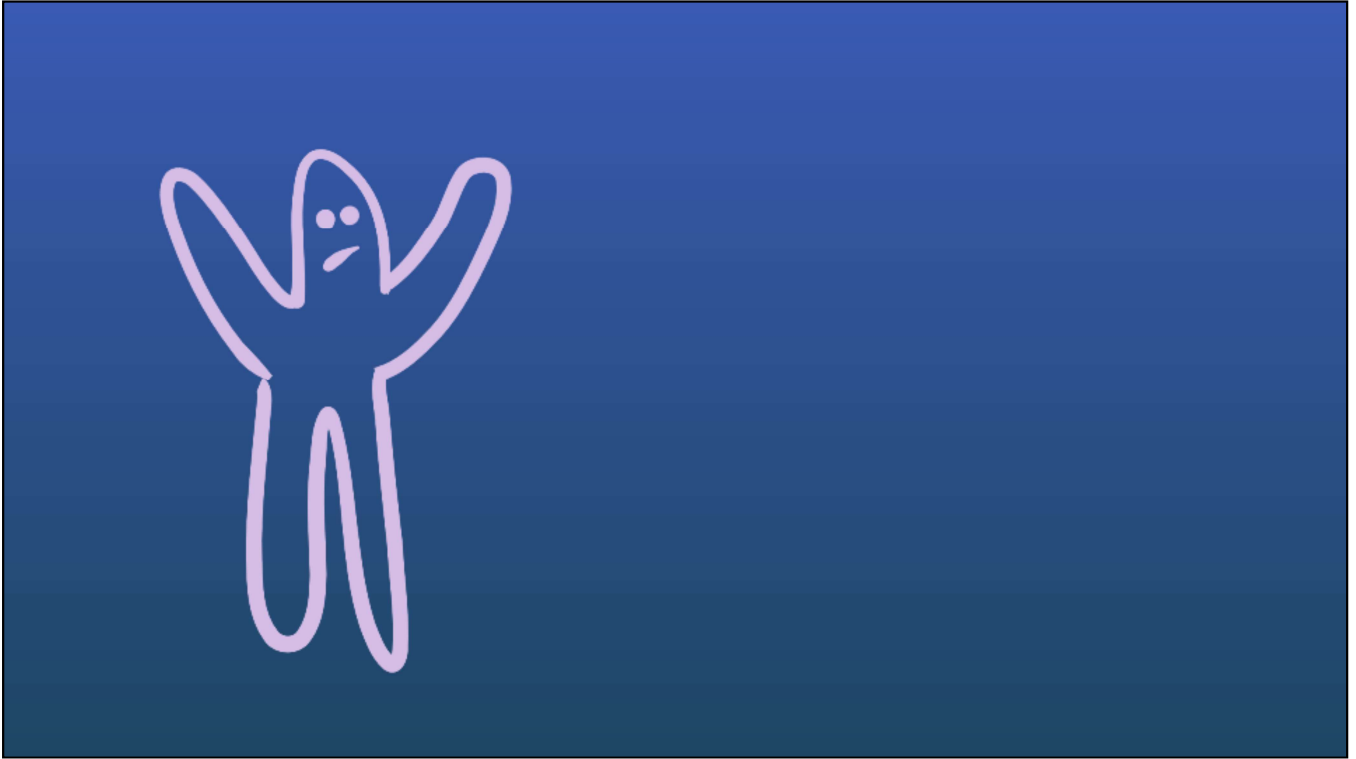


To put it bluntly — they're minor rules. It's hard to imagine any designer using them to help guide overall thought processes.





I've always wondered why our list didn't turn up more expansive rules.





**deep rules are hard
to articulate**

Big rules, important rules, governing rules — well, it's difficult to express exactly what guides one's thoughts — even for experienced designers . . .



**and they may feel
like tricks**

. . . and some developers were (and are) loathe to admit that they would ever resort to what might be seen as a shallow bag of tricks.



rules = wisdom



But rules aren't tricks, and they are not inexpressible. They are actionable summaries of deep principles. They contain the wisdom of prior experience.

**important rules
are BIG rules**



And it's the big rules, rules that cover a lot of ground, rules that are big enough to guide design processes, that are important.



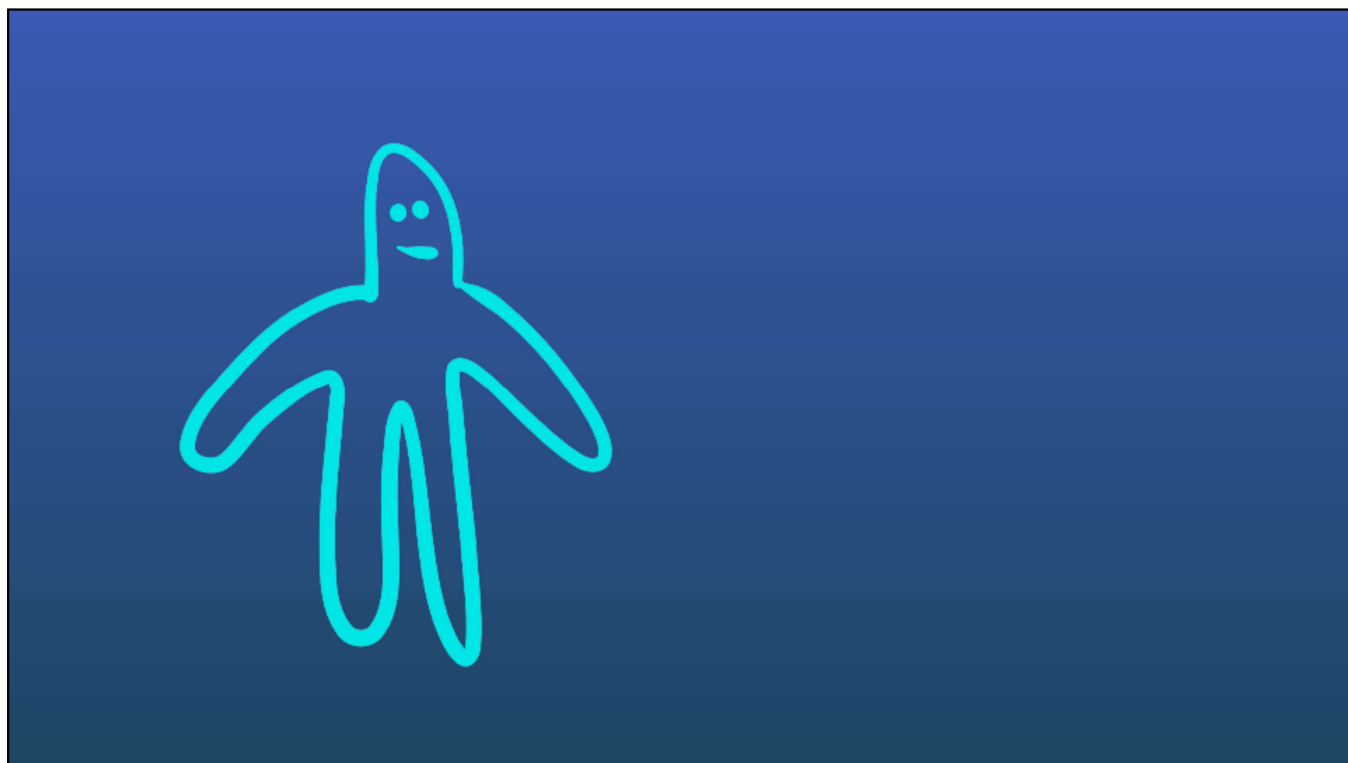


So here's the rule I want to discuss today — it illustrates the possibilities. It's a big rule with wide application.

use characters to solve design difficulties



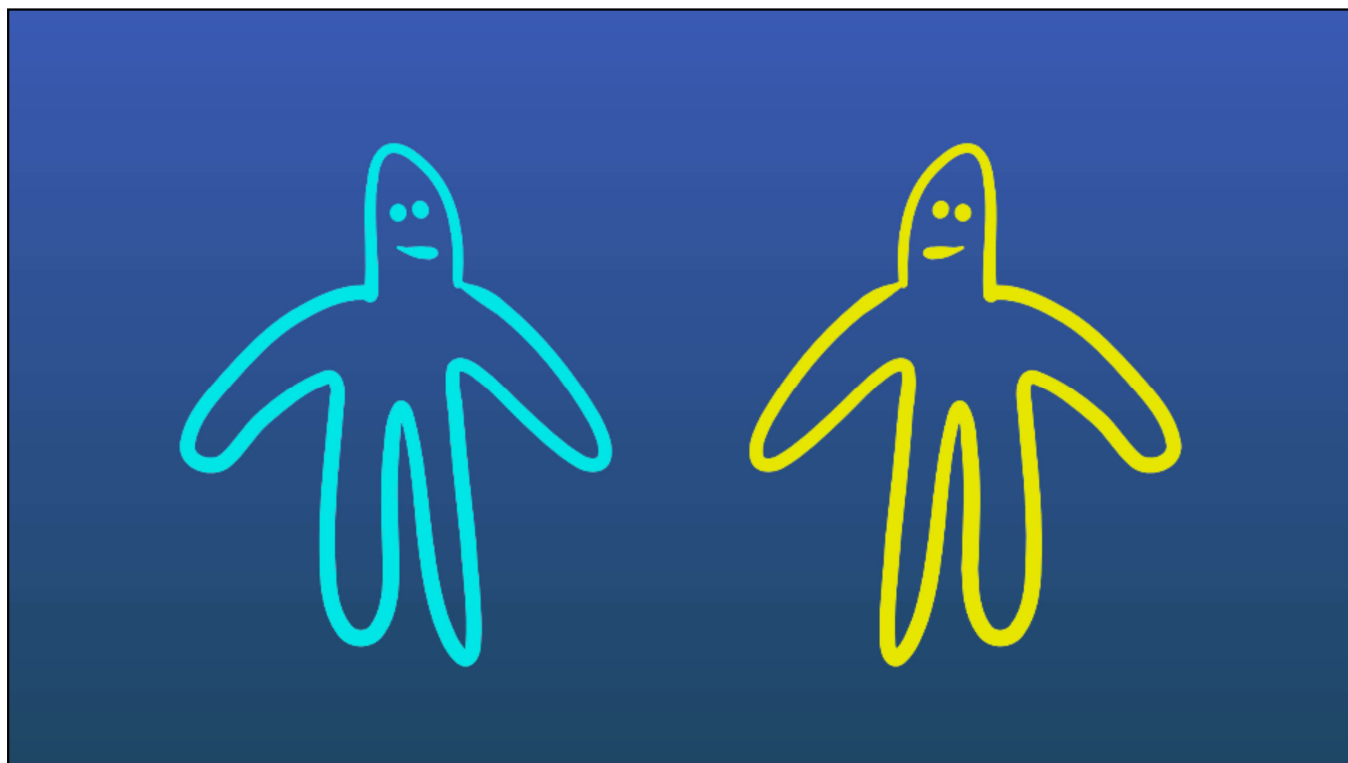
What do I mean → I'm not talking about the elementary notion of populating a narrative game with people: that's a given. How to make use of them is always a problem, however. And what's a little subtle is the idea that many design problems that might be dealt with mechanically can be better approached by turning them into human (or at least sentient) characters.

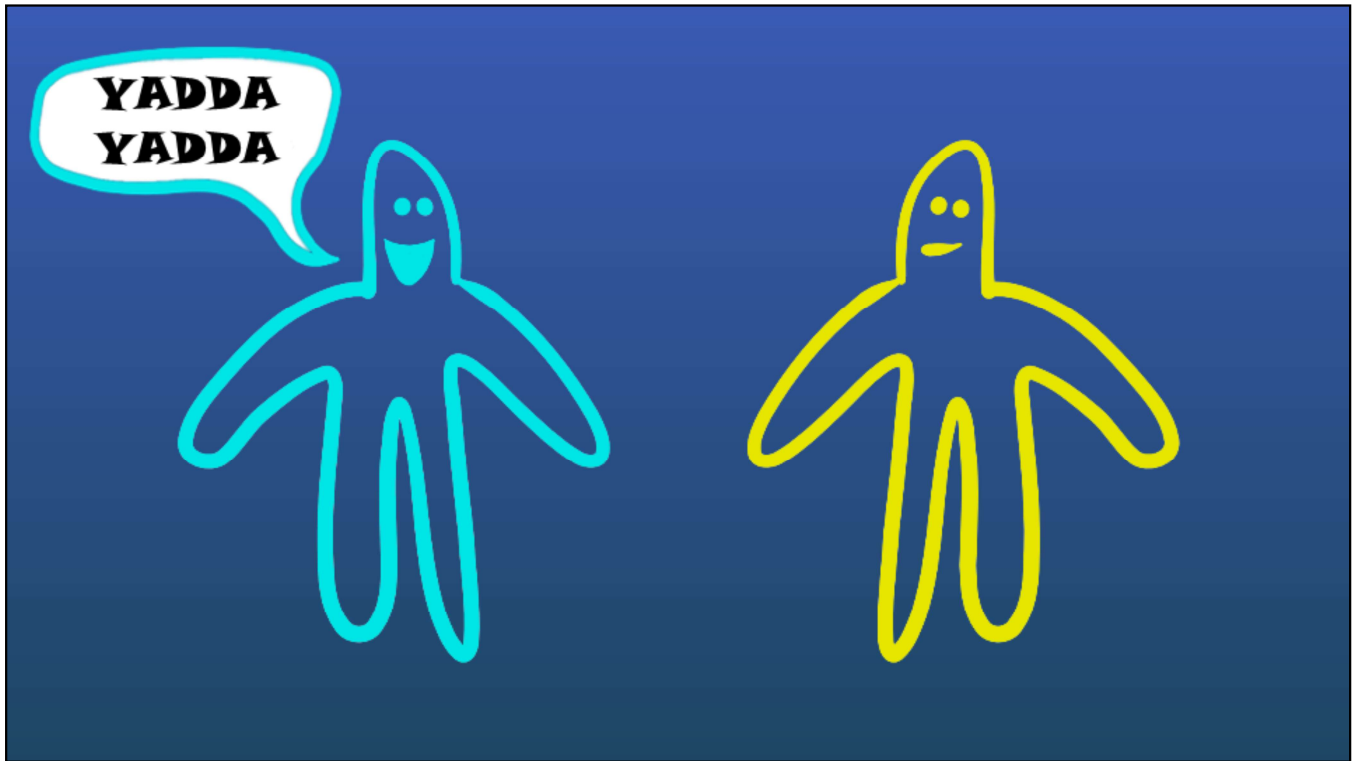


**fundamental to the sort
of games I like**

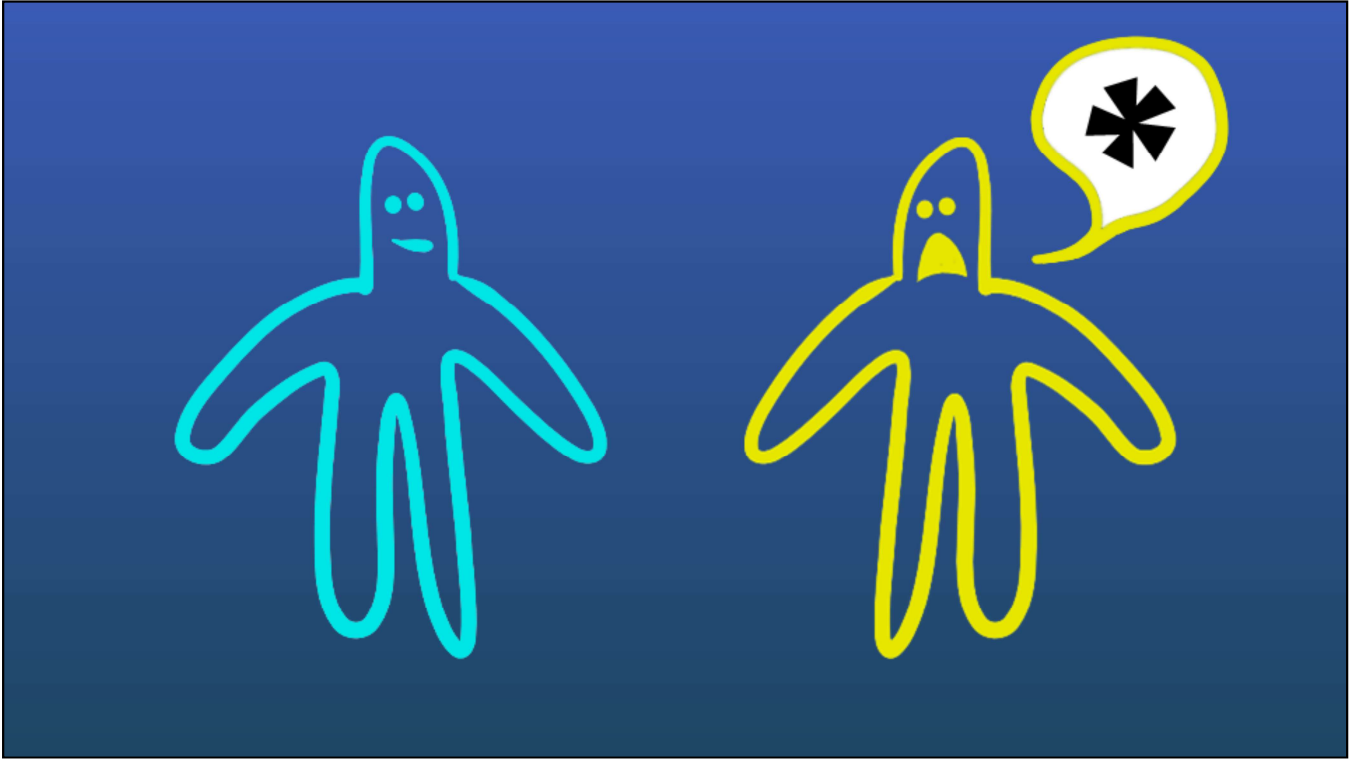


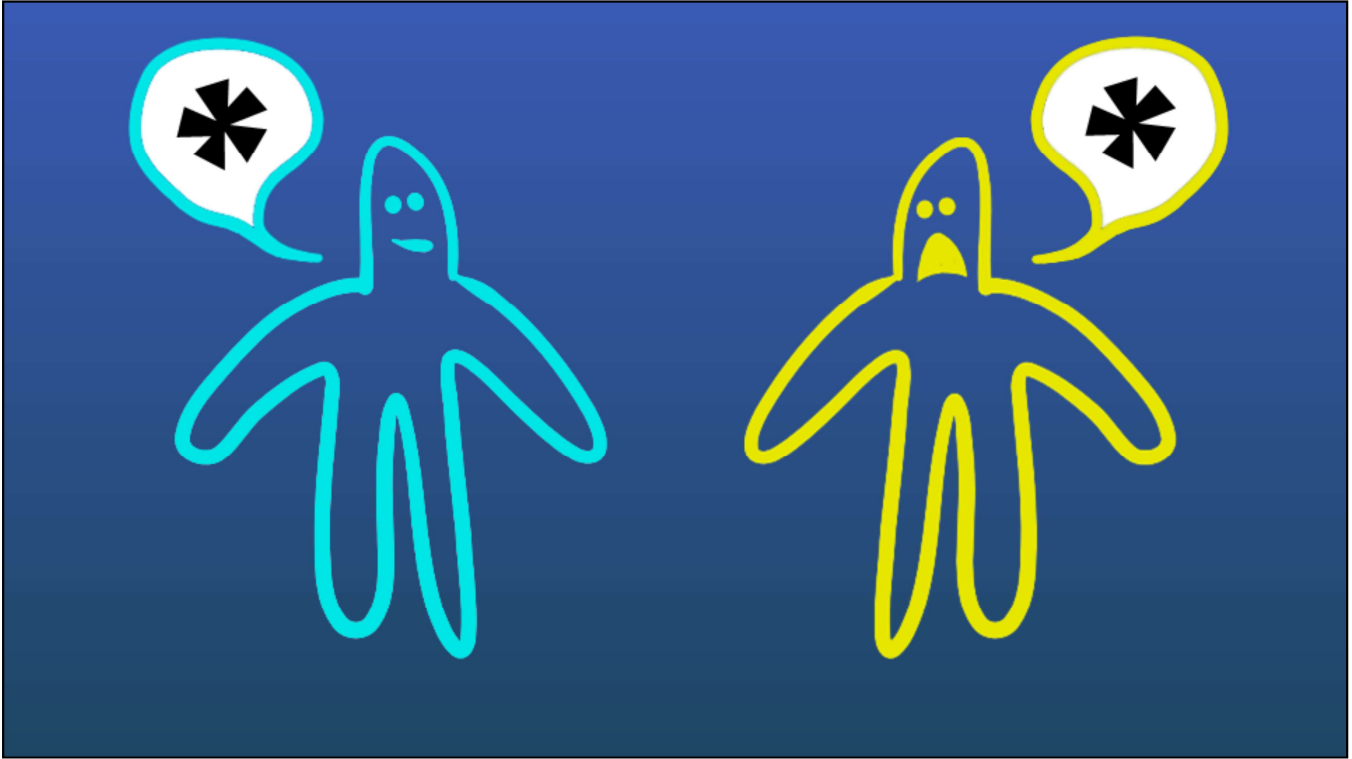
This idea is fundamental enough to govern the thought processes that go into pulling a design together from the word go. It doesn't apply everywhere, but its reach is broad, especially in the genre of games I like to play and build . . .

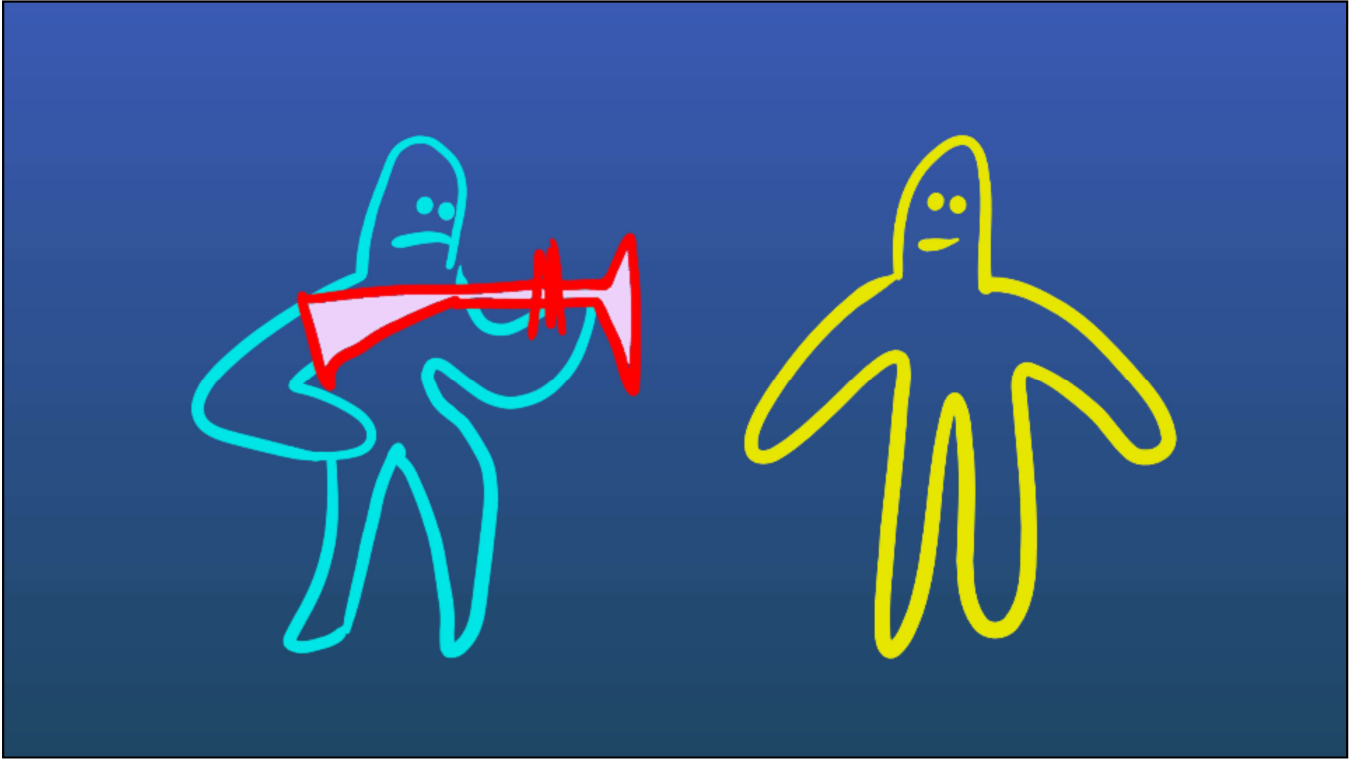




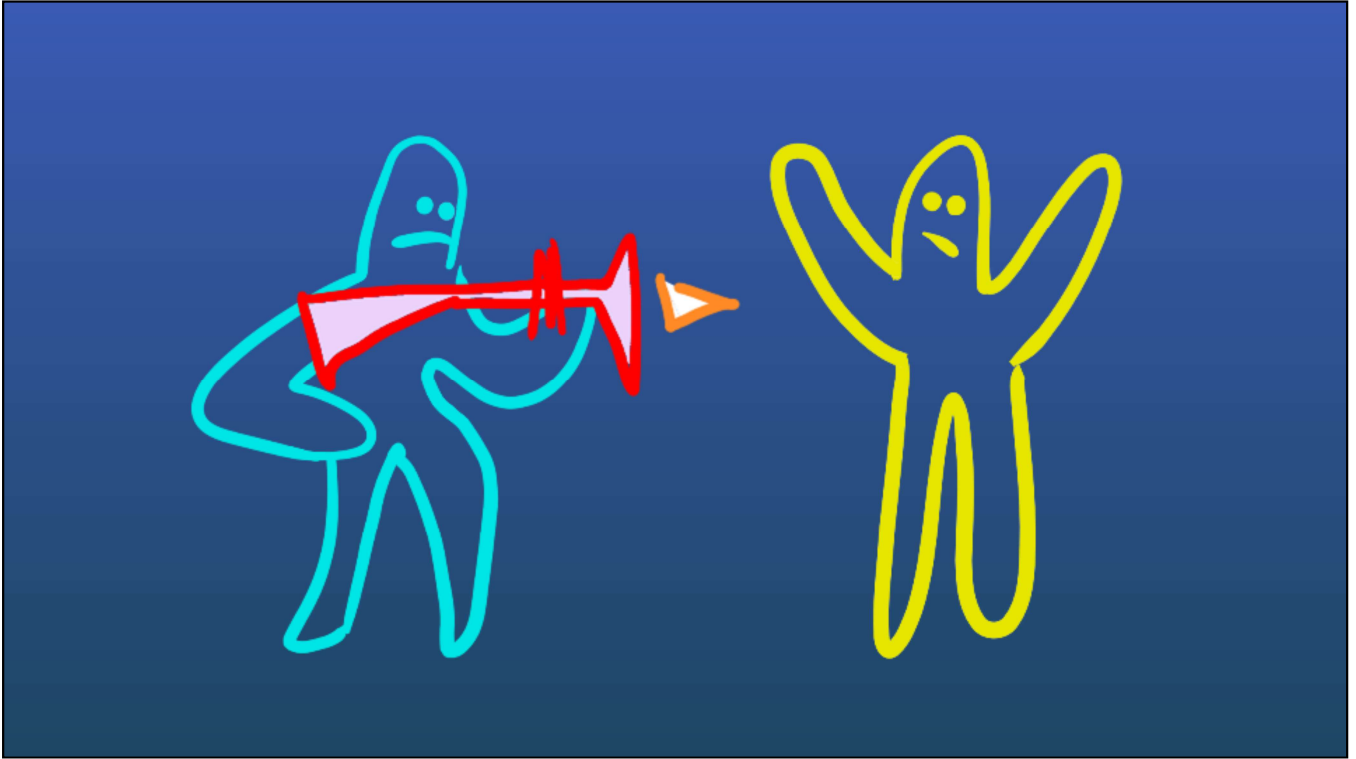
. . . games with lots of character interaction . . .

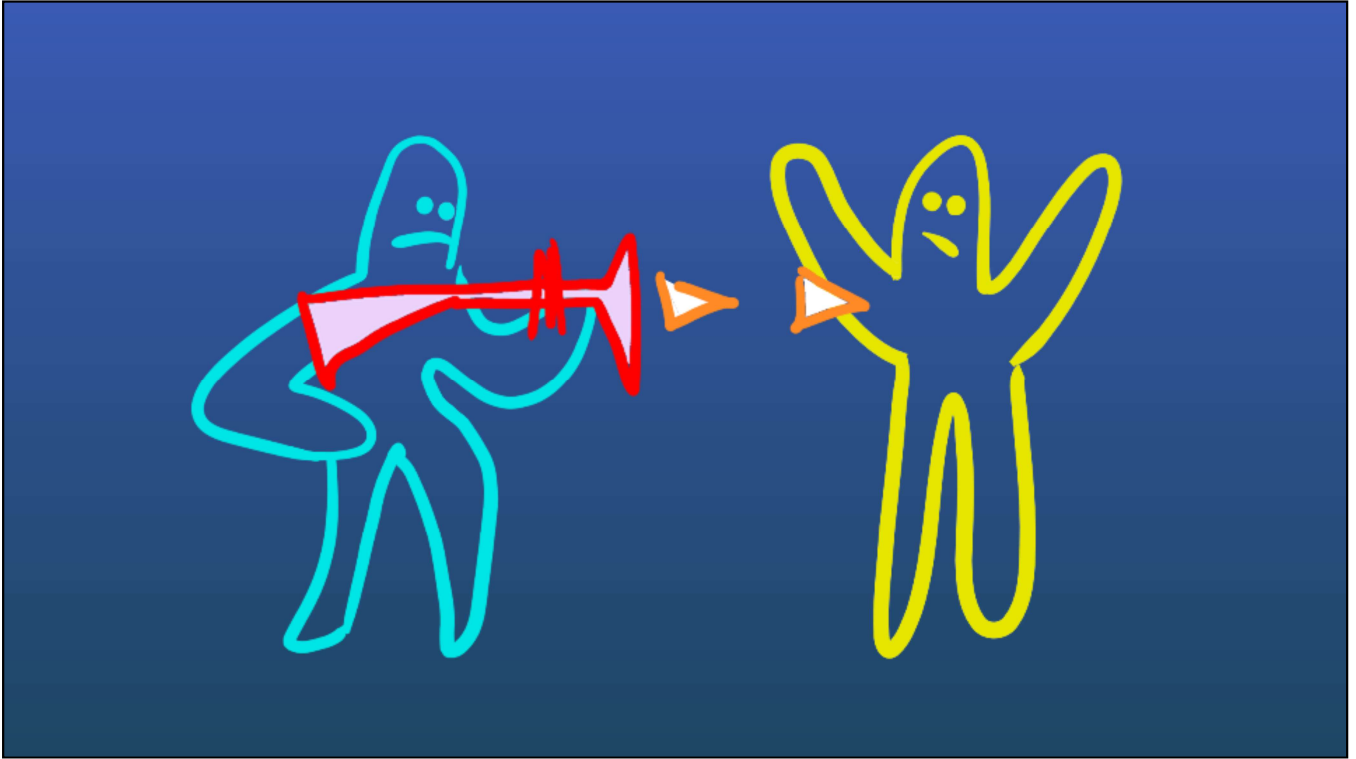


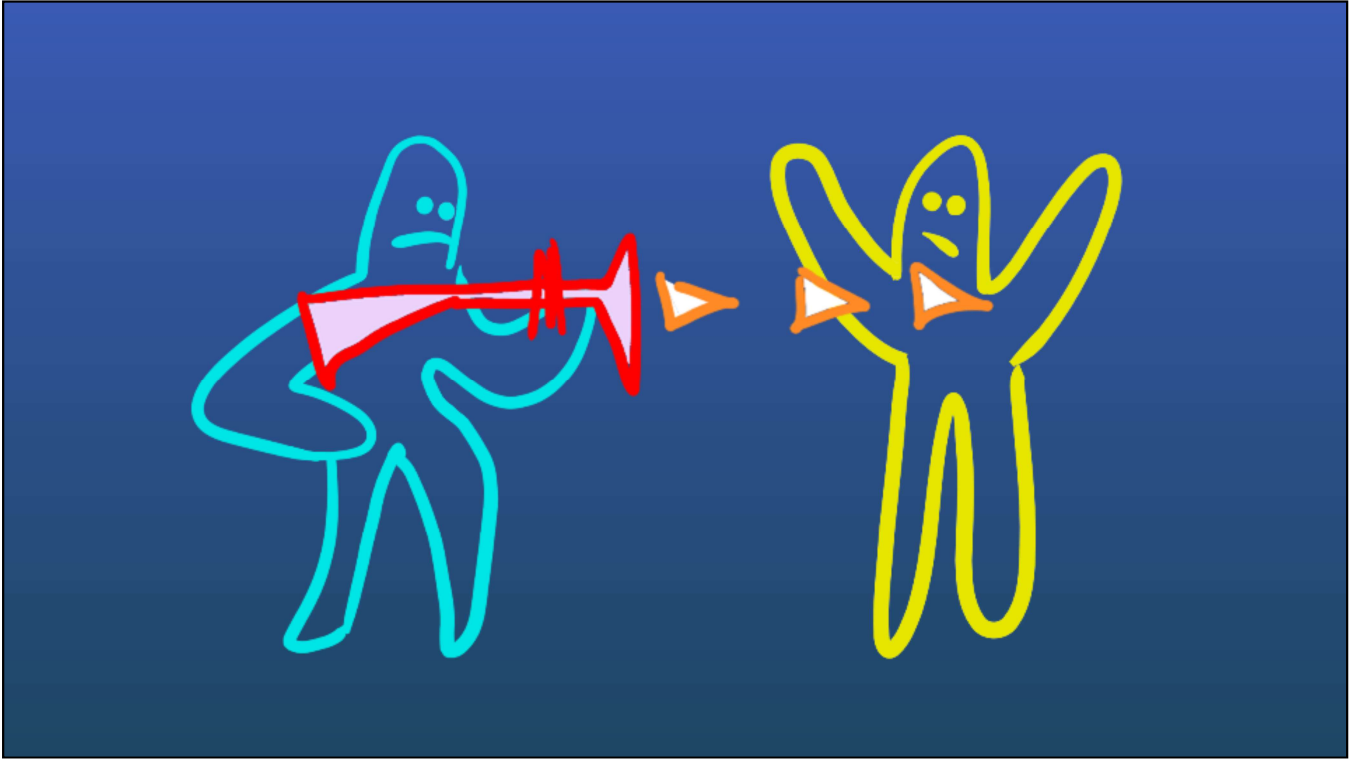


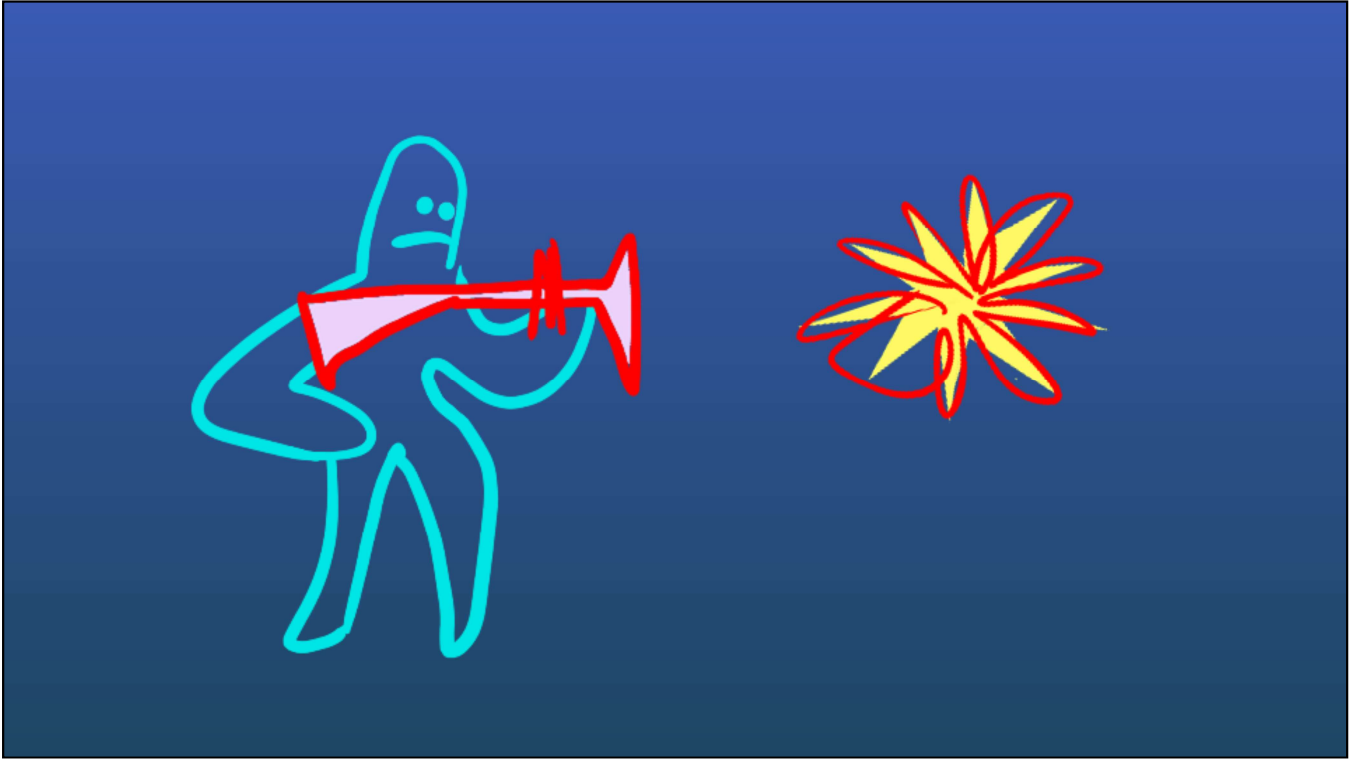


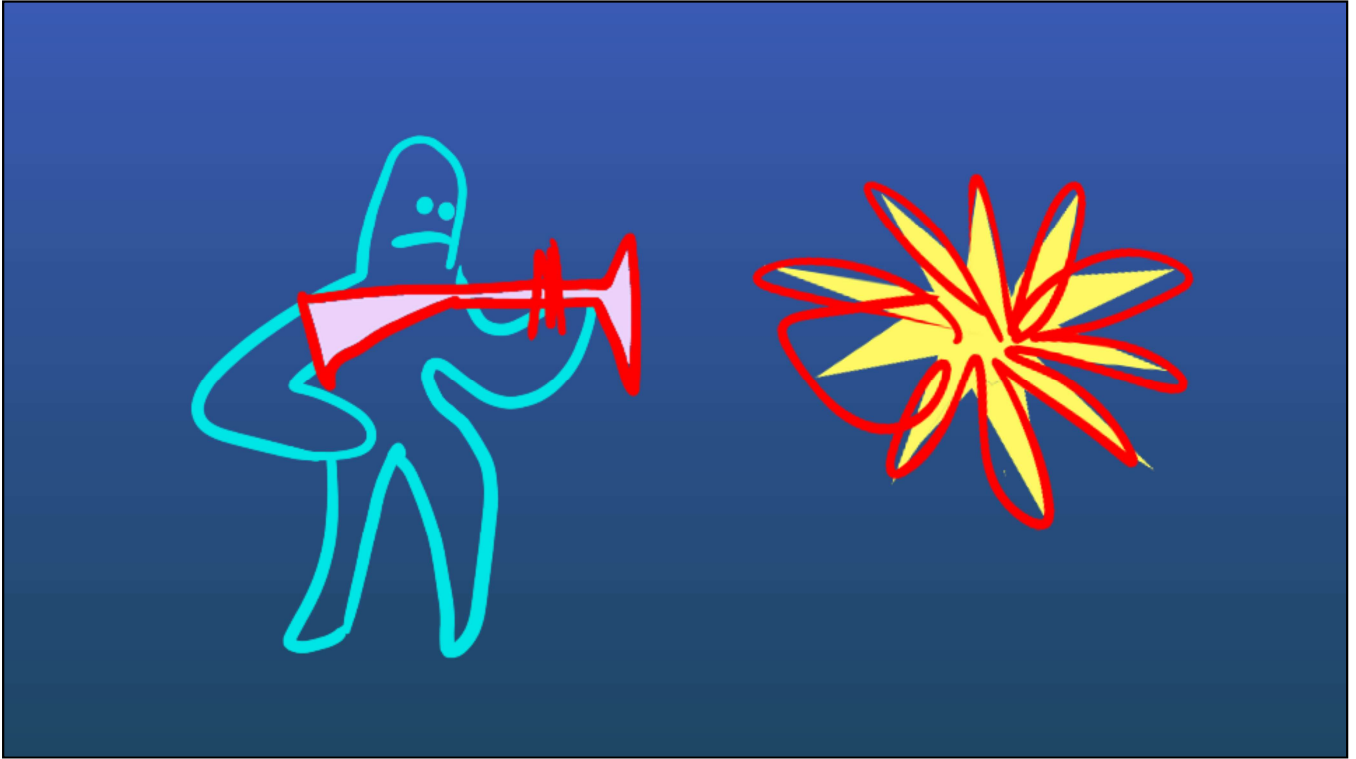
. . . games with some merciless action . . .

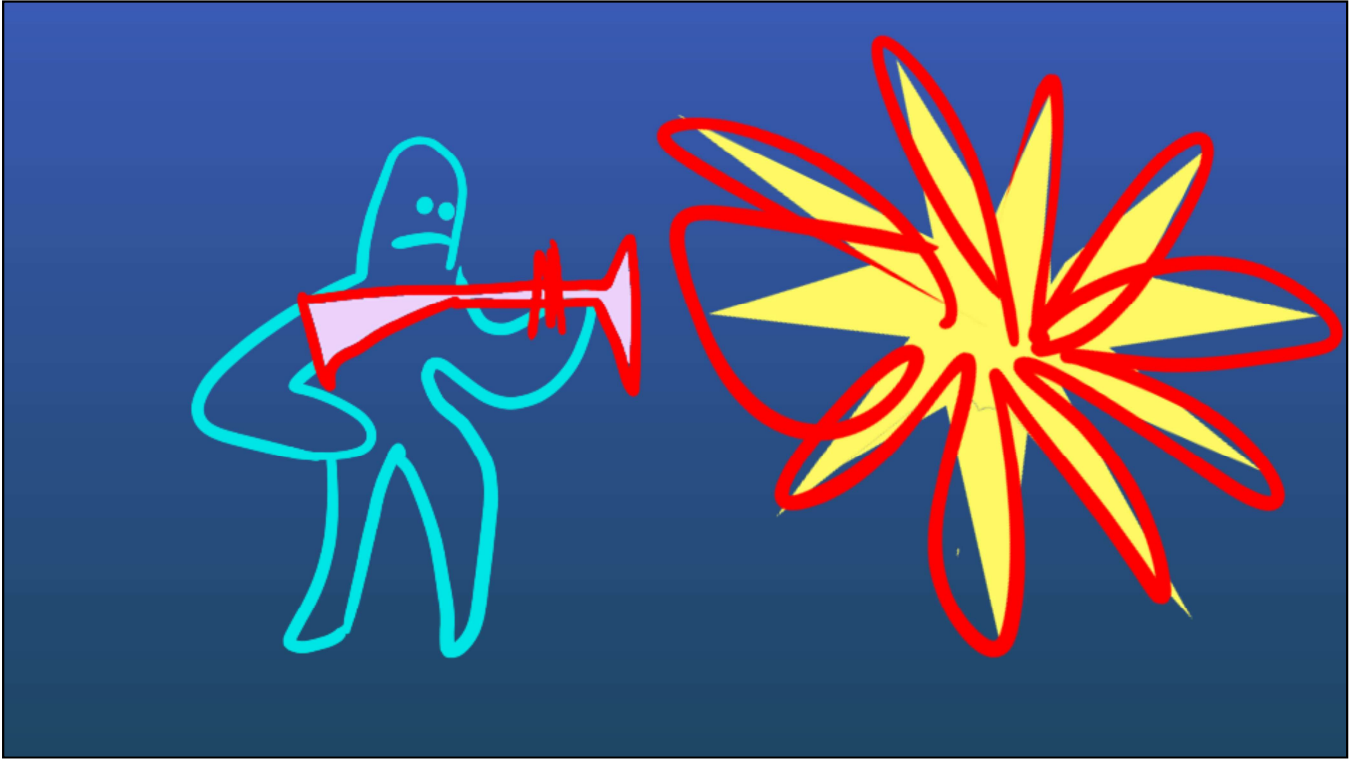


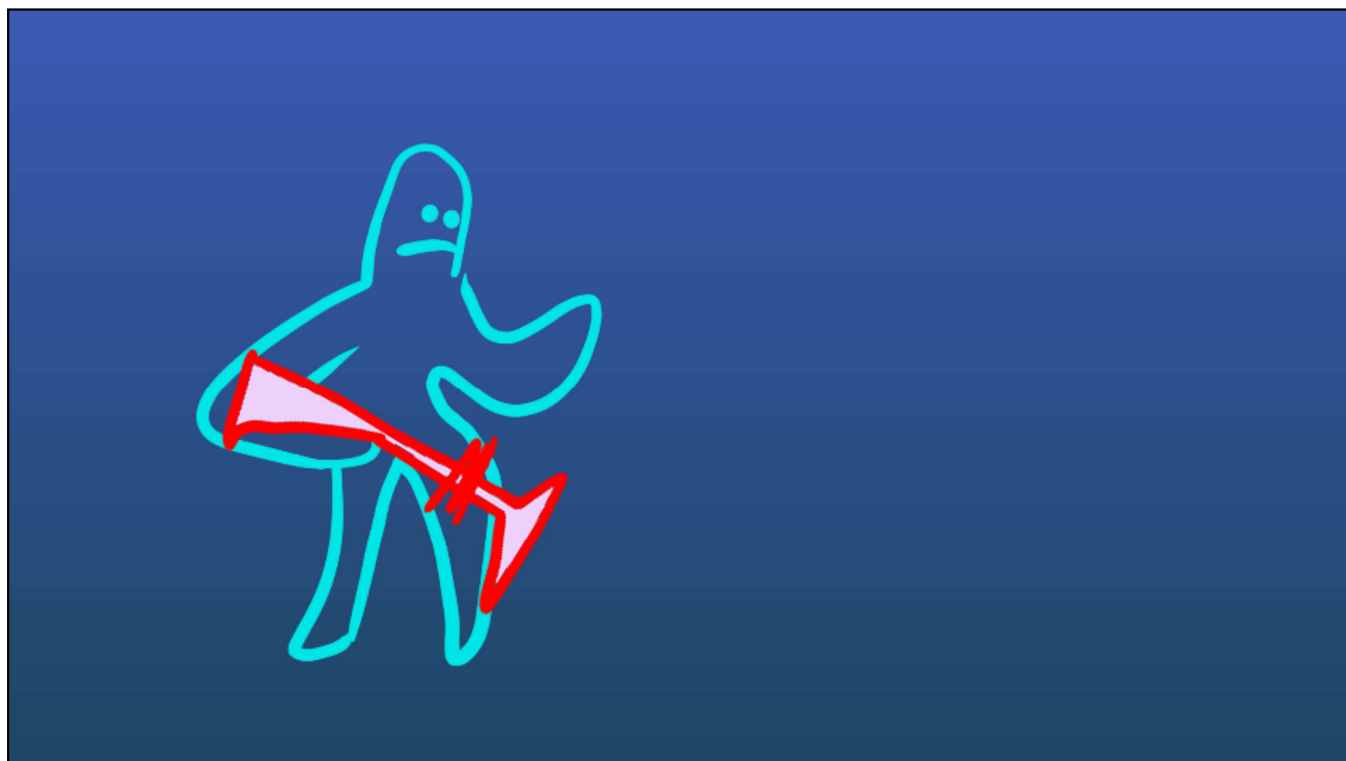


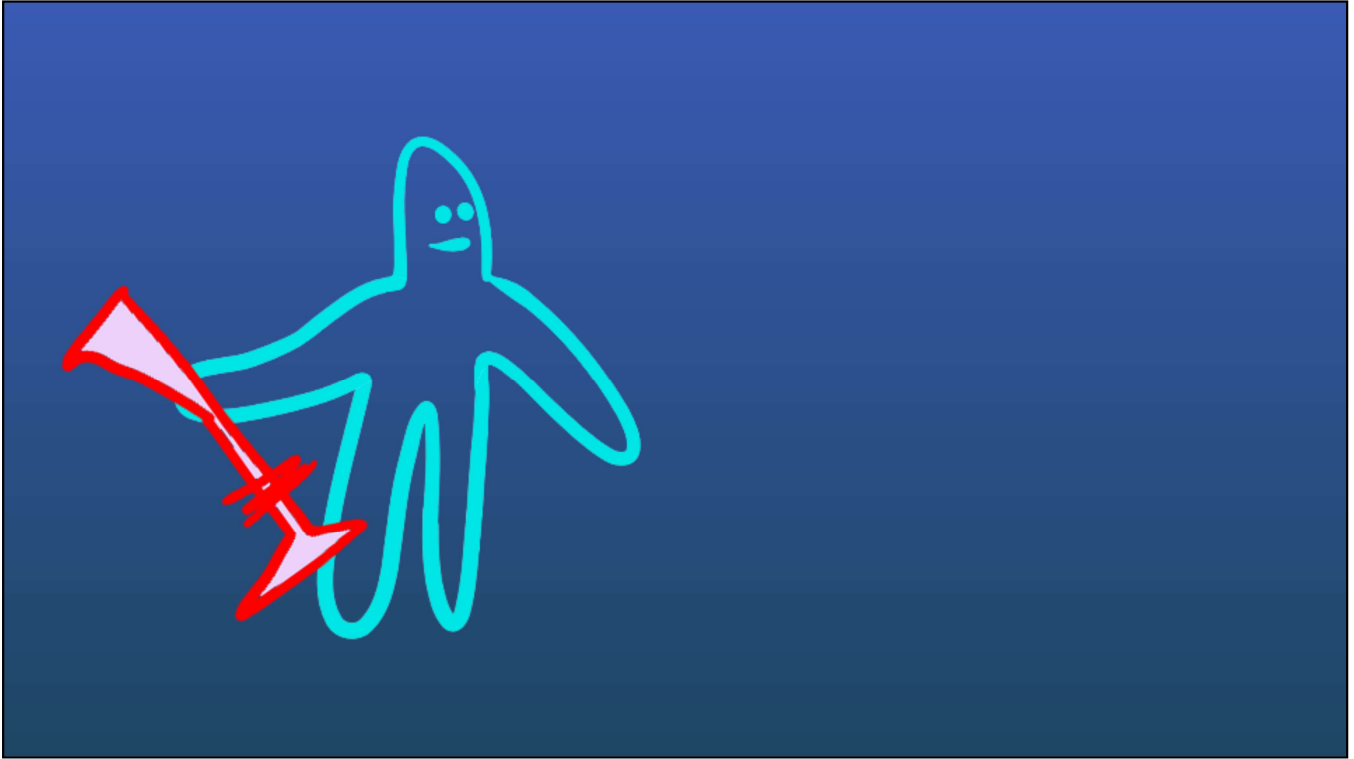








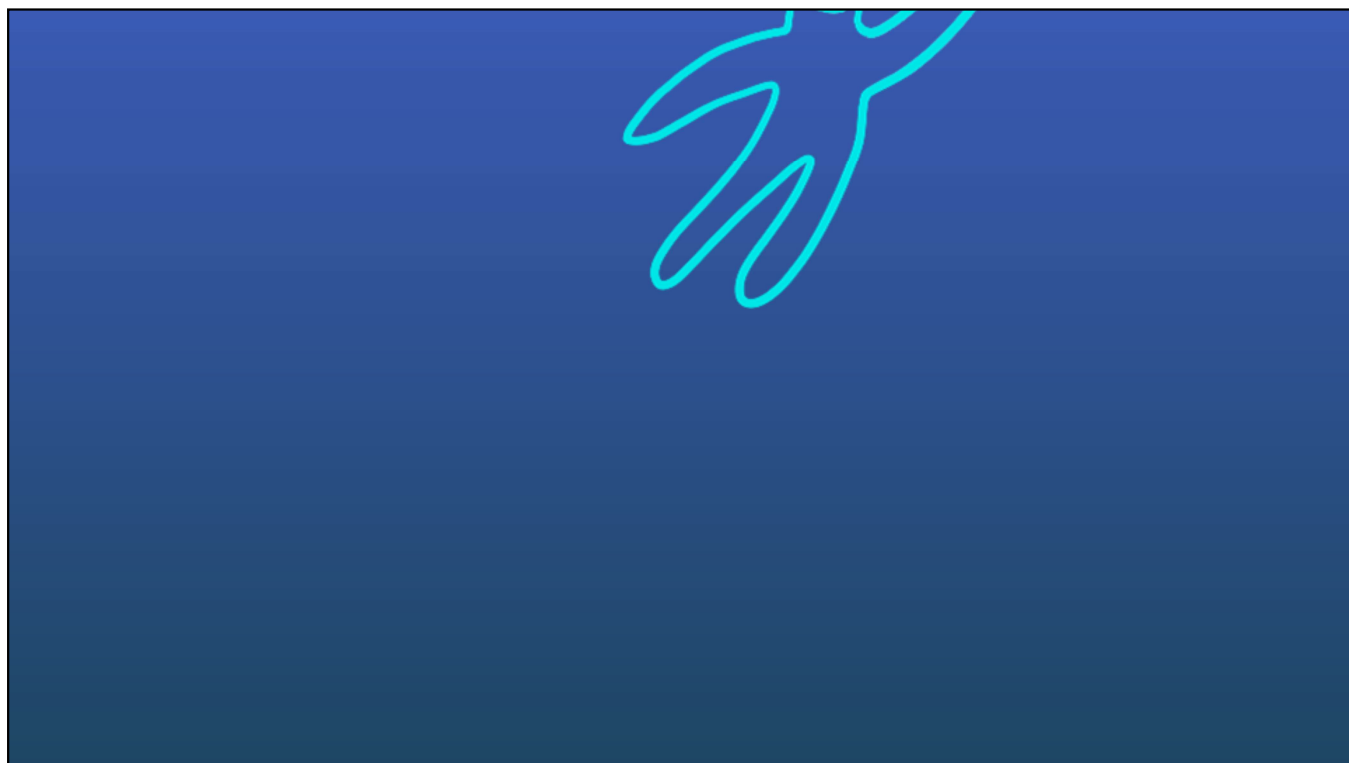


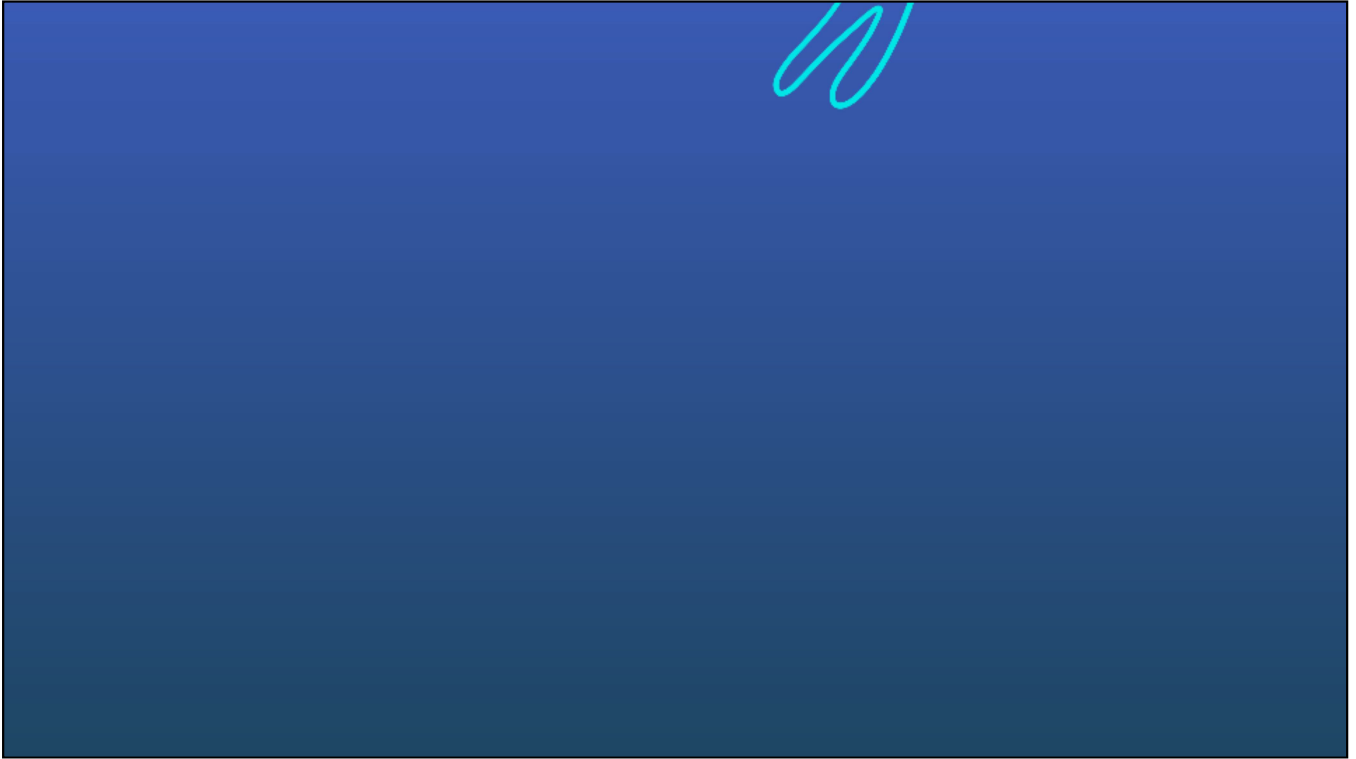


. . . games with exploration and traversal . . .

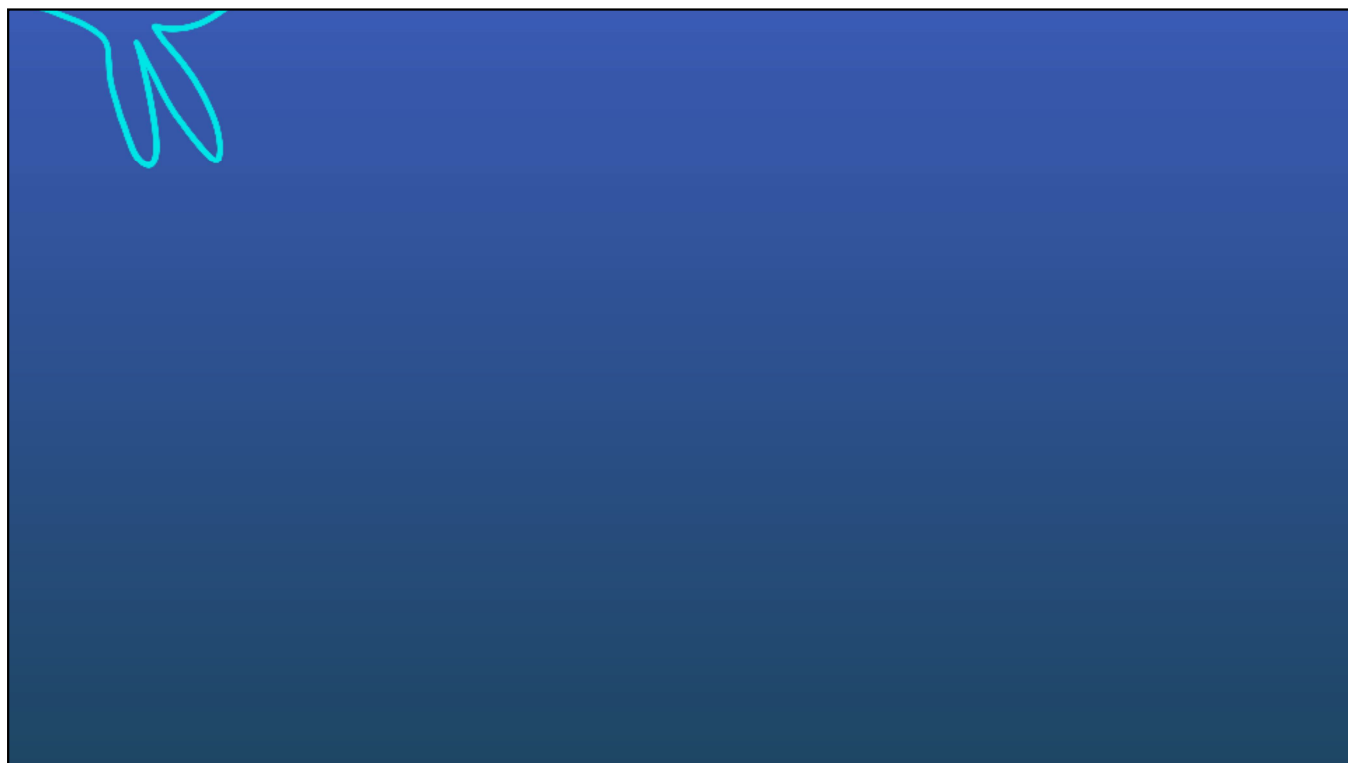




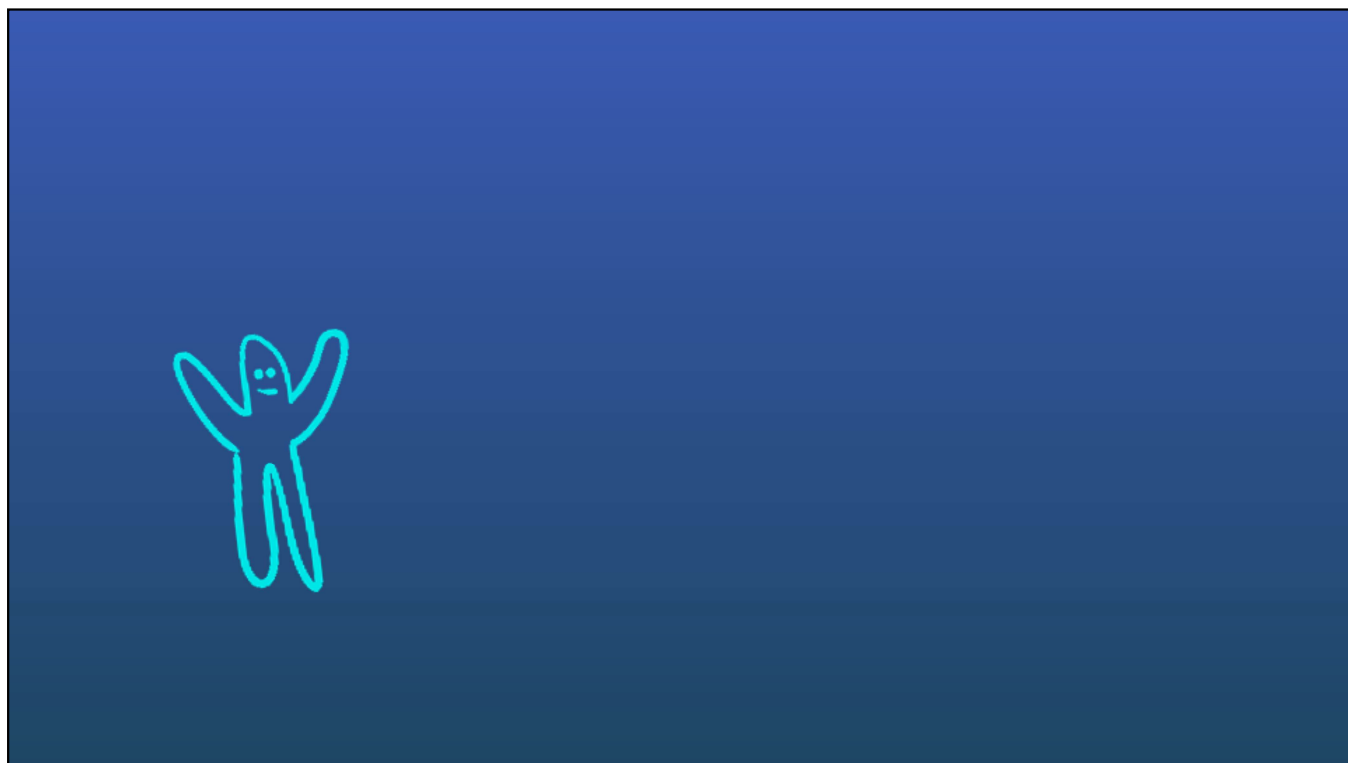
















the domain of ...

Character can only be understood in terms of action; action & traversal are meaningless without a purpose . . .



the domain of ... *narrative games*

. . . character, action, exploration — all 3 mechanics are components of narrative games, the domain for my rule. Such games don't exhaust the field, but they form a large fraction of popular videogames today.

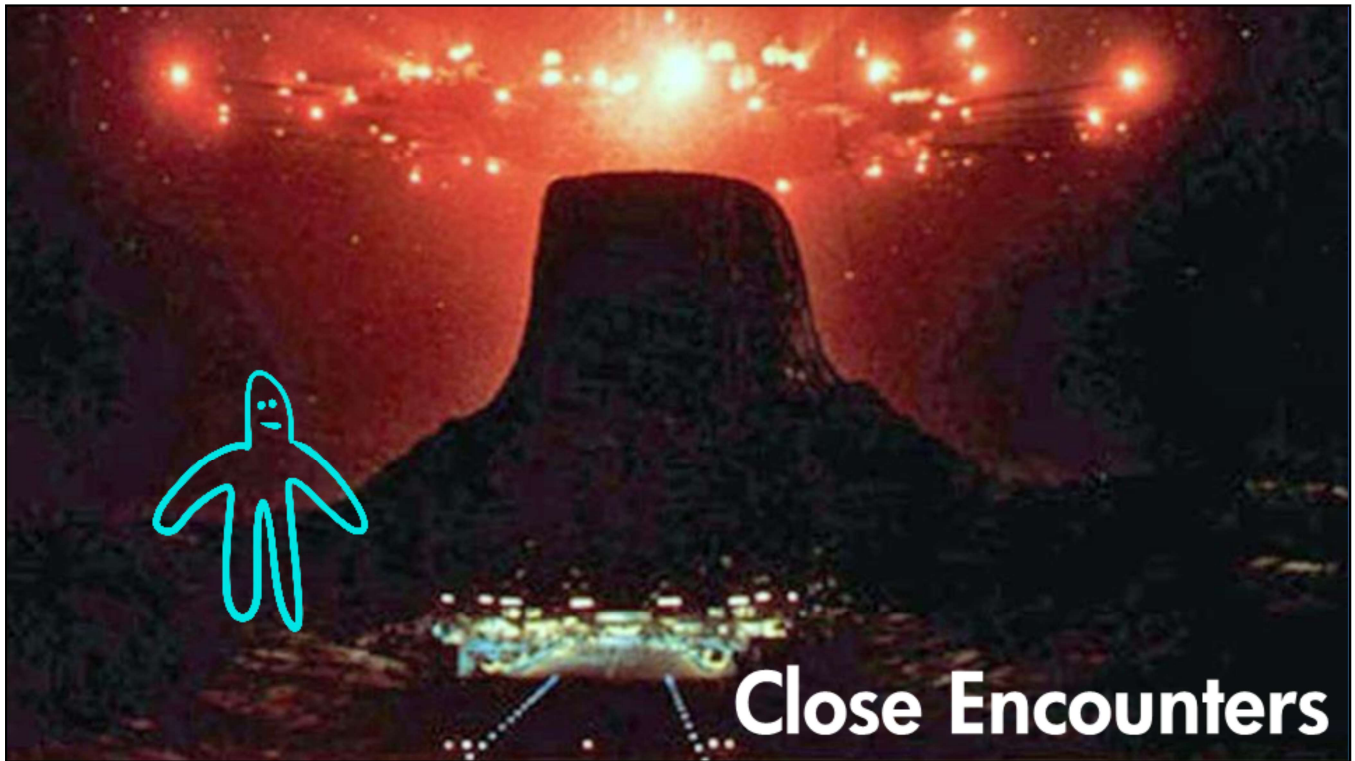


(yup, it comes from my
Hollywood days)

A confession: I brought this rule with me from my Hollywood screenwriting days, where it is well-understood in all its ramifications



A confession: I brought this rule with me from my Hollywood screenwriting days, where it is well-understood in all its ramifications



Let's look at how my rule applies to movies I worked on years ago
— here's *Close Encounters of the Third Kind*



Steven (Spielberg) wanted it to be awe-inspiring, positive, uplifting. The problem is this: awesomeness only goes so far on its own. Soon it wears out, feels dry.



This is where a warm heart must collide with a cool head. Without any sense of sentimentality, but in order to add personal drama to the final act, my partner Matthew Robbins and I invented a character — little Barry — got him abducted by aliens in the first act . . .



. . . and delivered him to his mother's arms in that awesome finale, giving the sequence emotional heft



In our movie *Dragonslayer*, Matthew and I cooked up a story that's all about a romping, stomping fire-breathing dragon. The problem is this: a dragon can be scary, but like any other animal, it's a force of nature, not a dramatic villain . . .



. . . so we invented the creepy King Casiodorus of Urland as the chief bad guy . . .



. . . and his hateful sacrificial lottery, to give the tale satisfying dramatic structure



just storytelling?

Does that mean the rule applies merely to storytelling, merely the narrative backdrop to a game? (For those who are interested, good discussions of the screenwriting trade's practicalities can be found in Laslo Egri's *The Art of Dramatic Writing* and in Jon Boorstin's *Making Movies Work*.)



just storytelling?
not at all ...

Does that mean the rule applies merely to storytelling, merely the narrative backdrop to a game? (For those who are interested, good discussions of the screenwriting trade's practicalities can be found in Laslo Egri's *The Art of Dramatic Writing* and in Jon Boorstin's *Making Movies Work*.)



"show, don't tell"
... in movies

"show don't tell" — a great screenwriting rule



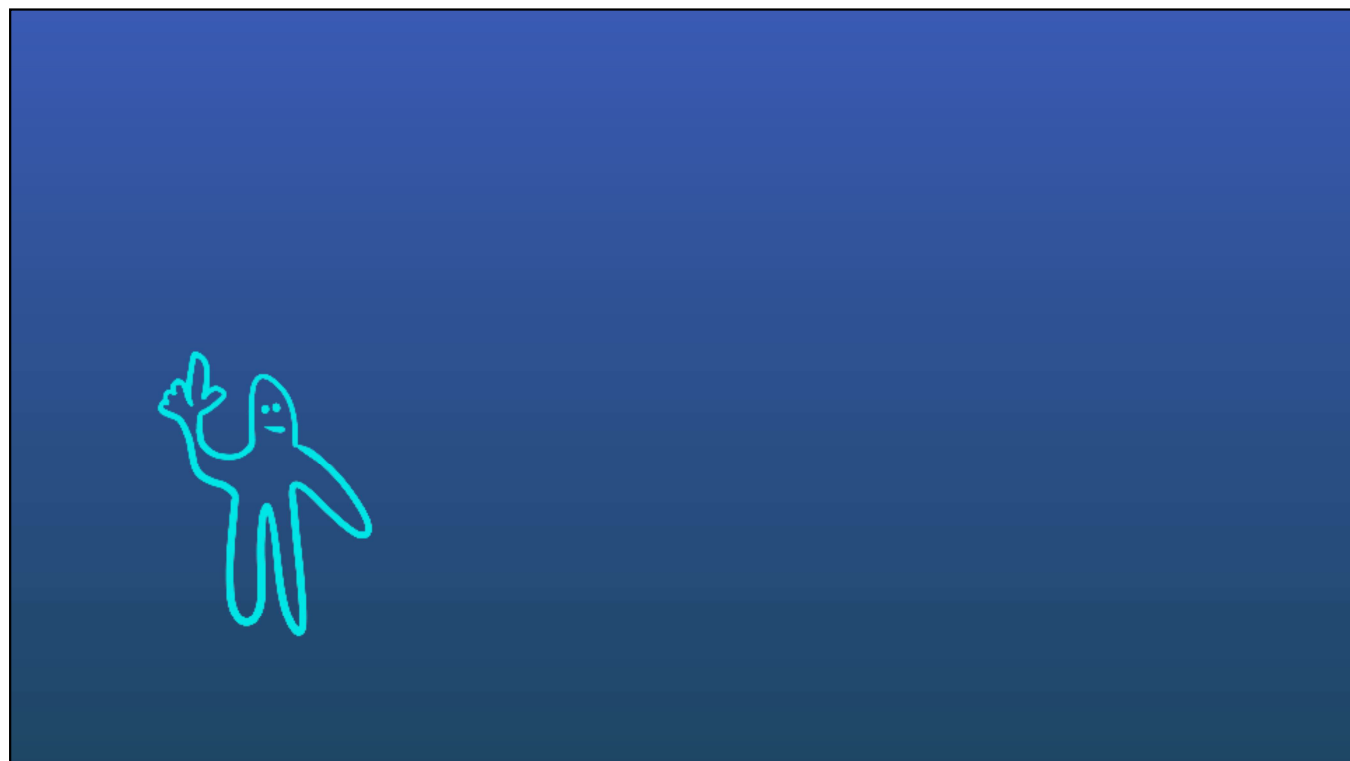
"play, don't show"
... in games

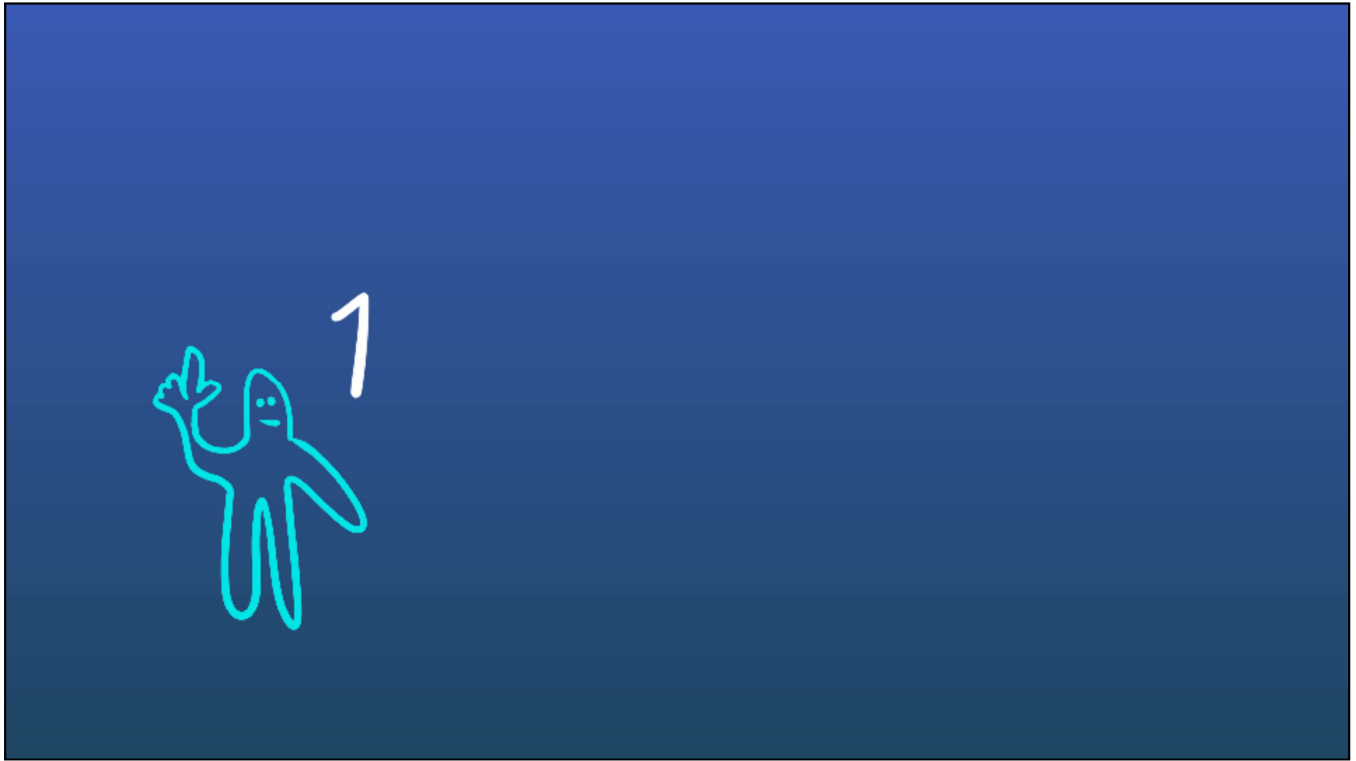
"play, don't show" — an equally sound "big rule" for games



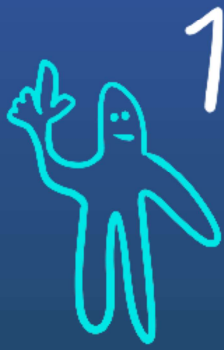
all about *design* — welding story & gameplay

the purpose of my rule is to unify, to weld together, story and game play





Let's look at some examples . . .



problem:

**keep danger alive
before it arrives**

In many games, the heart of resistance is located hours of play away in the final sequences. It's important to keep the game's themes and threats alive and in the players' minds more or less all the time. There are several ways to do that, and here are some from an old game of mine . . .



Indiana Jones and the Fate of Atlantis promises exotic adventure

. . .



. . . so I cooked up a companion for Indy (a must in a Jones story)
— but Sophia Hapgood is a shady former archeologist who
stumbled on an Atlantean necklace years ago. She traded her
digging tools for a stage career as a psychic without being aware
of her find's unhealthy influence

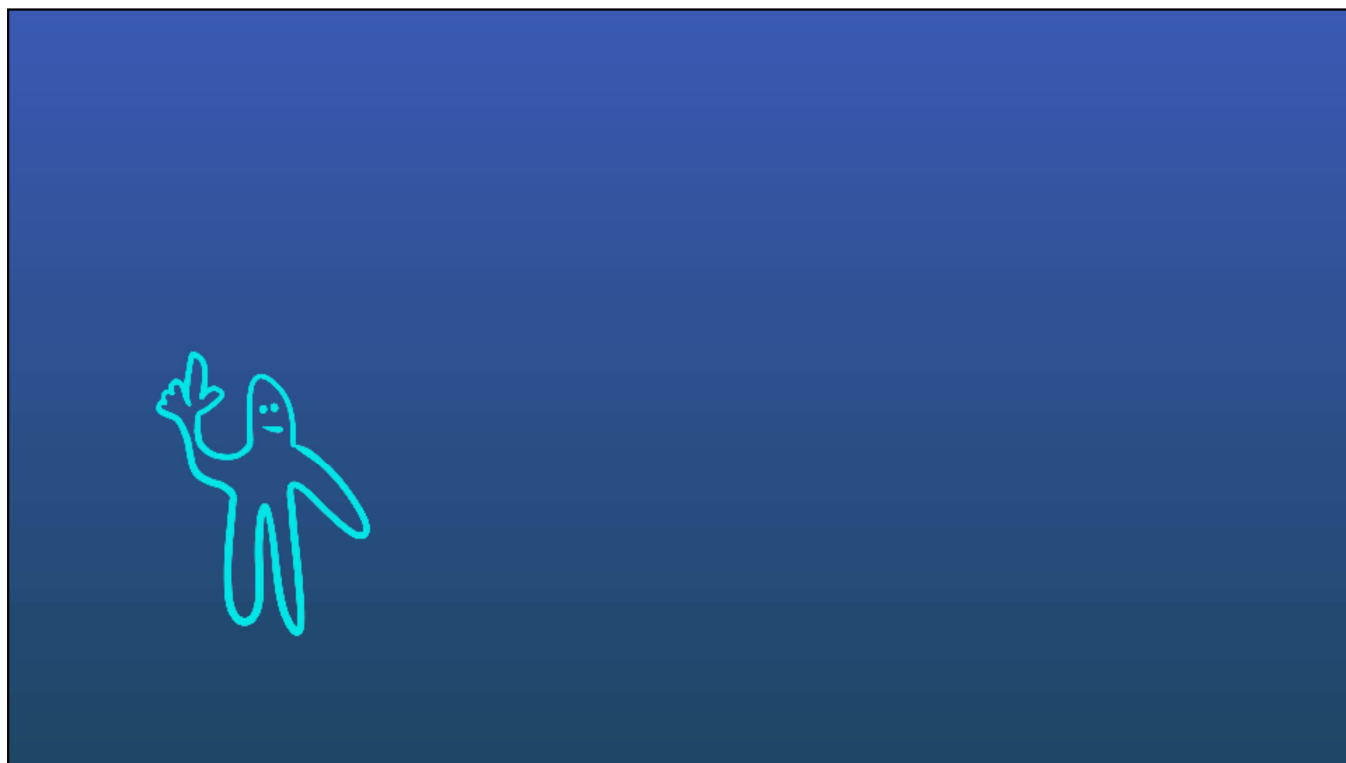


Although Atlantis is far away for most of the game, Sophia carries the threat in and on her person from the beginning



We see that Indy's skepticism is unjustified — Sophia's necklace foreshadows the strange powers of the Lost City long before we get there.







problem:
the uselessness of
Star Wars items

A lot of games (including my own) take a long time to play through. I thought the time commitment drove players away. I had an idea for a replayable narrative game, and LucasArts gave me the go-ahead, as long as it wrapped itself in the *Star Wars* universe

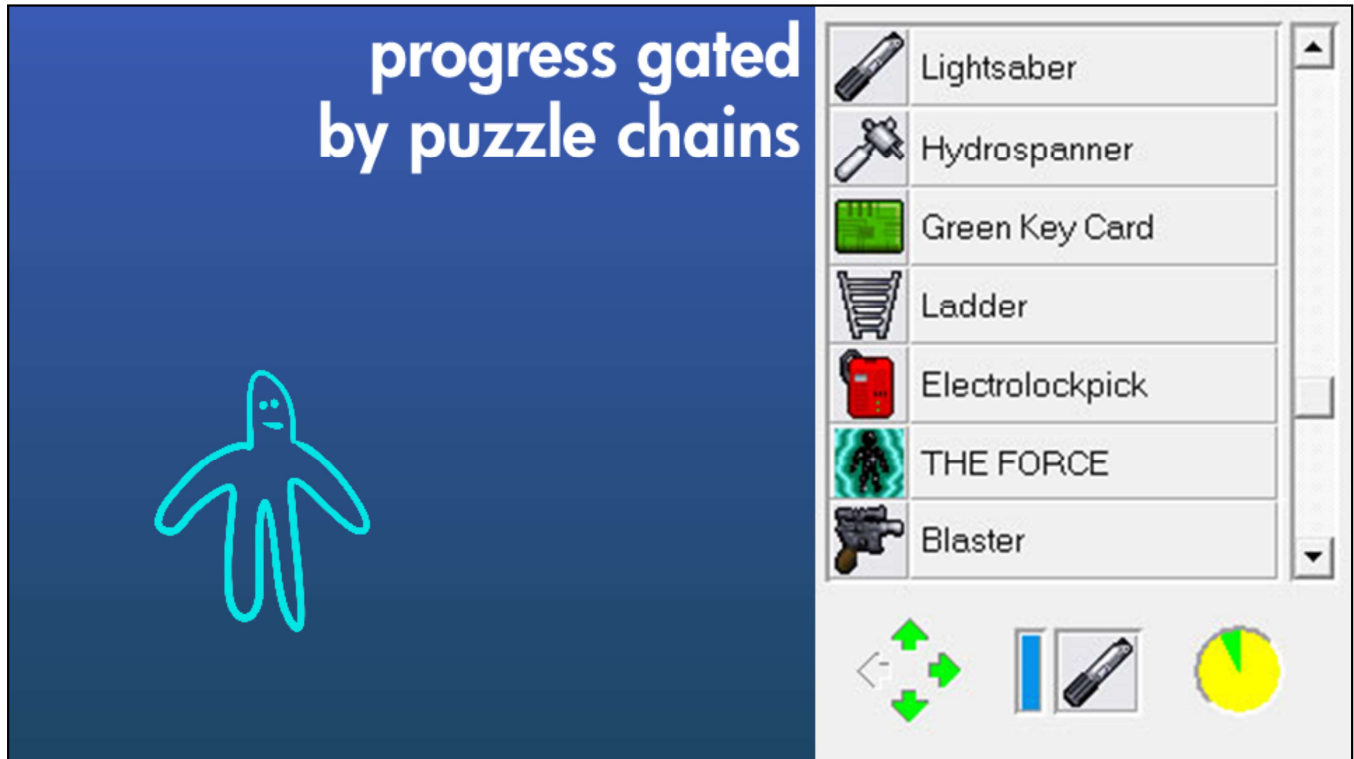


I didn't know it at the time, but *Yoda Stories* was a casual game before the term existed. It's a replayable story game, — 15 action-adventure scenarios that go together in hundreds of different ways. Each scenario can be completed in less than an hour

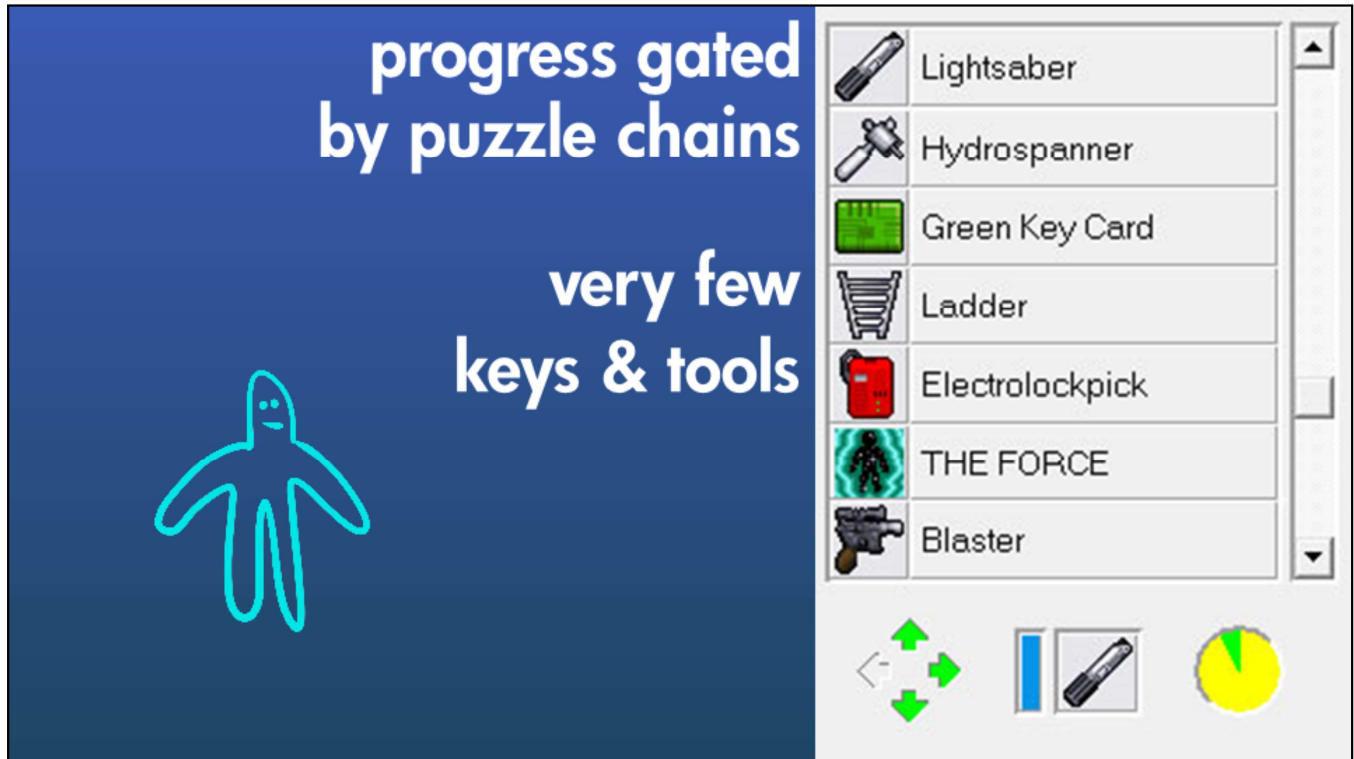


a casual applet

But it's small on your desktop, not necessarily destined to be all that immersive



Progress is governed by puzzle chains. We did the usual — key and tool puzzles . . .



Progress is governed by puzzle chains. We did the usual — key and tool puzzles . . .



But the *Star Wars* universe is shockingly bereft of useful items. We looked hard, without much luck. What are we to do with things like this?



The answer? Add a new category of puzzle — *valuables*. Humans place value on things arbitrarily — anything an NPC claims is worth something automatically acquires worth



Valuables gave us the scope to build hundreds of levels and puzzles and introduce character interaction that drives progress



The cornucopia of silly *Star Wars* stuff allowed us to introduce many transactions into the game . . .

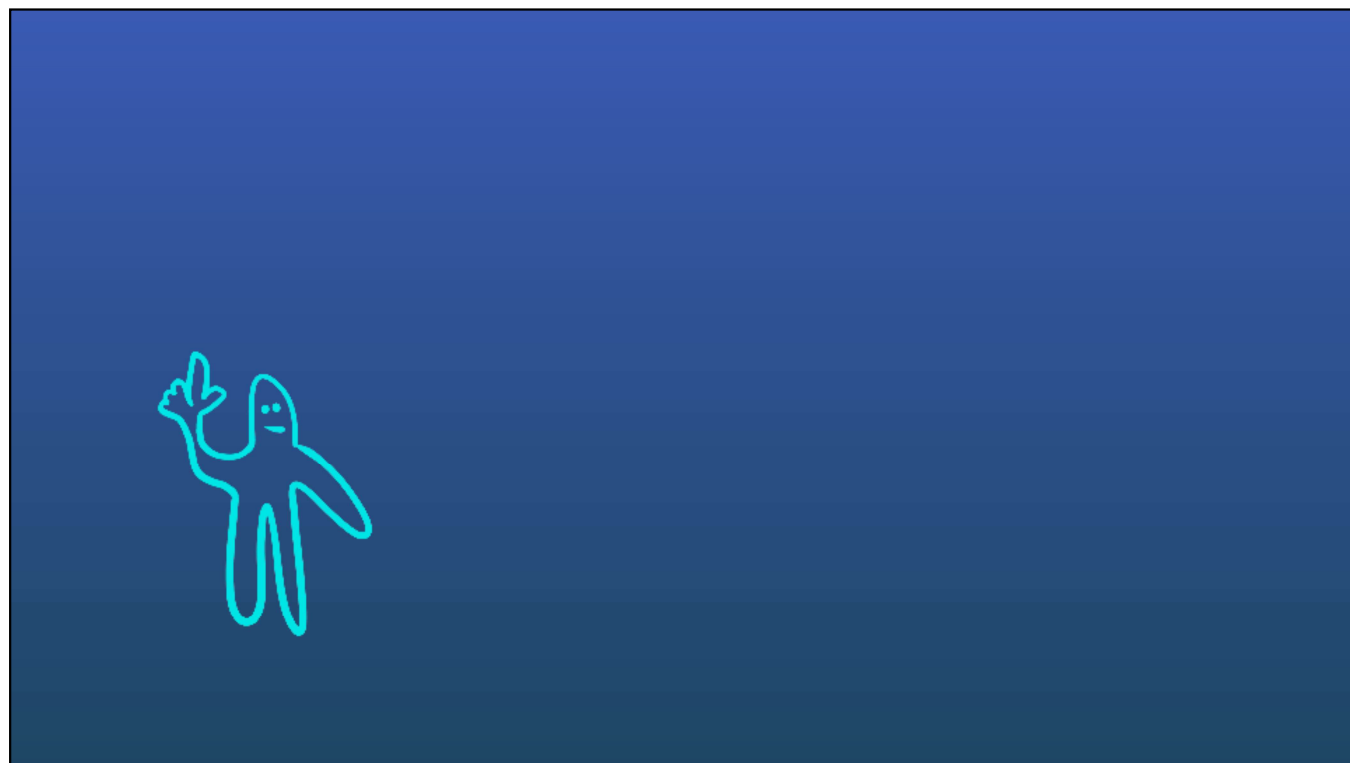






Moreover, populating the game with lots of characters who request and demand things warms up the game and imparts a lively feel to the otherwise cool presentation. . . immersing the player after all







LucasArts had the rights to the *Indiana Jones* franchise, and we wanted to exploit it. The problem was this: with several movies and other games in existence for reference, how best to satisfy a player's expectations?



Indiana Jones and the Infernal Machine — Indy in action. Jones stories are intensified versions of the real historical world, and they always shade into the supernatural. How to express this in a videogame?



In the Shambala sequence, a monastery is overseen by a bent-over old woman . . .



Jones scours the monastery to gather and nurture a flowering plant. When he presents it to her . . .



the plant blooms into a fantastical flower. . . and the old woman is magically transformed into a beautiful princess (well, as beautiful as you can be in 150 polygons) — hinting at deeper and weirder things to come.



We gave Indy a pal — Sophia Hapgood again, now an untrustworthy CIA operative — who helps him out from time to time, and is always around when the game's background needs explaining. Exposition is potentially deadly, always a design problem, but it's acceptable when placed in the mouth of a character.



But exposition is just narration, and we wanted to solidify the character by incorporating her into play. In the Palawan Volcano level, Sophia is kidnapped by Russian *spetsnaz* soldiers.



Meanwhile, Indy finds a locked gate



It opens at the touch of a switch — but it stays open only for an instant — not enough time for Indy to vault across a lava pool and make it through



rescuing Sophia becomes crucial



once free, she's happy to help



Indy whips across the chasm . . .



Sophia pushes ...



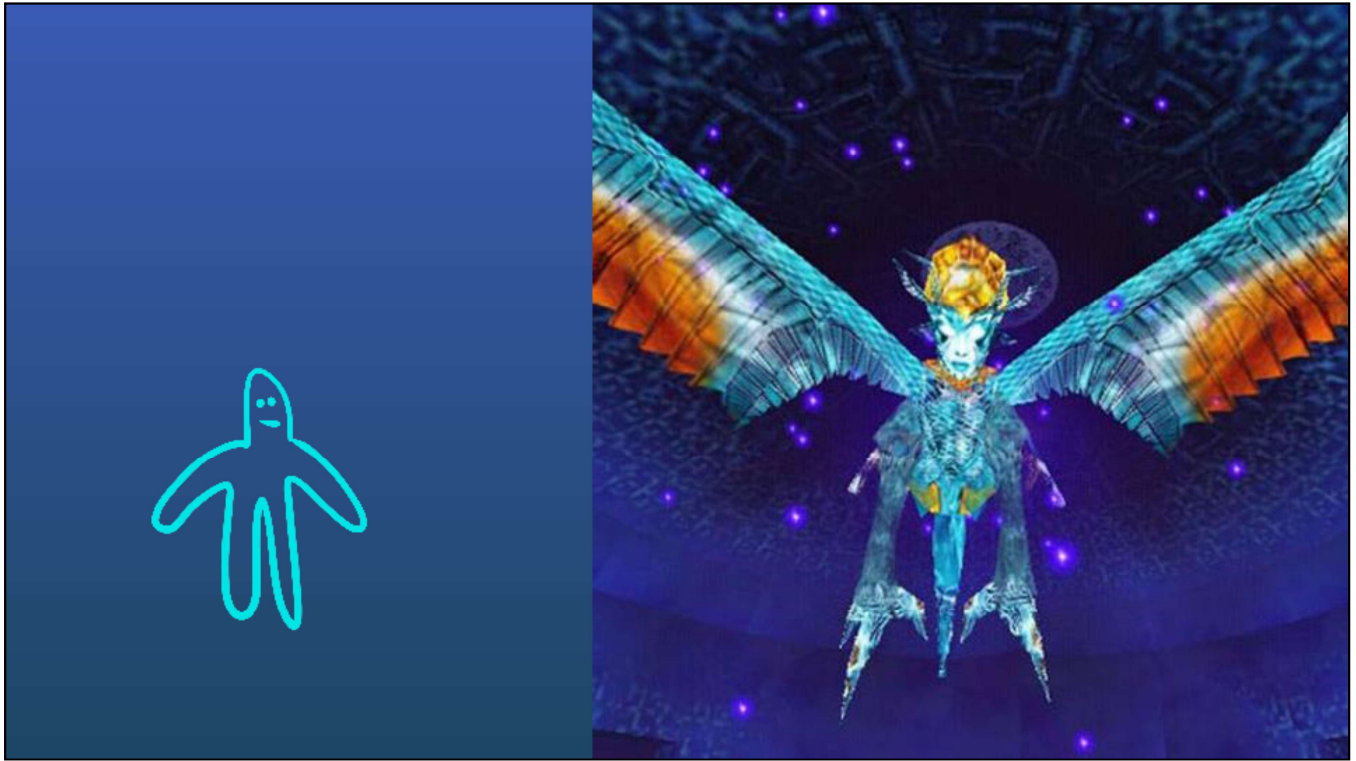
the gate opens . . .



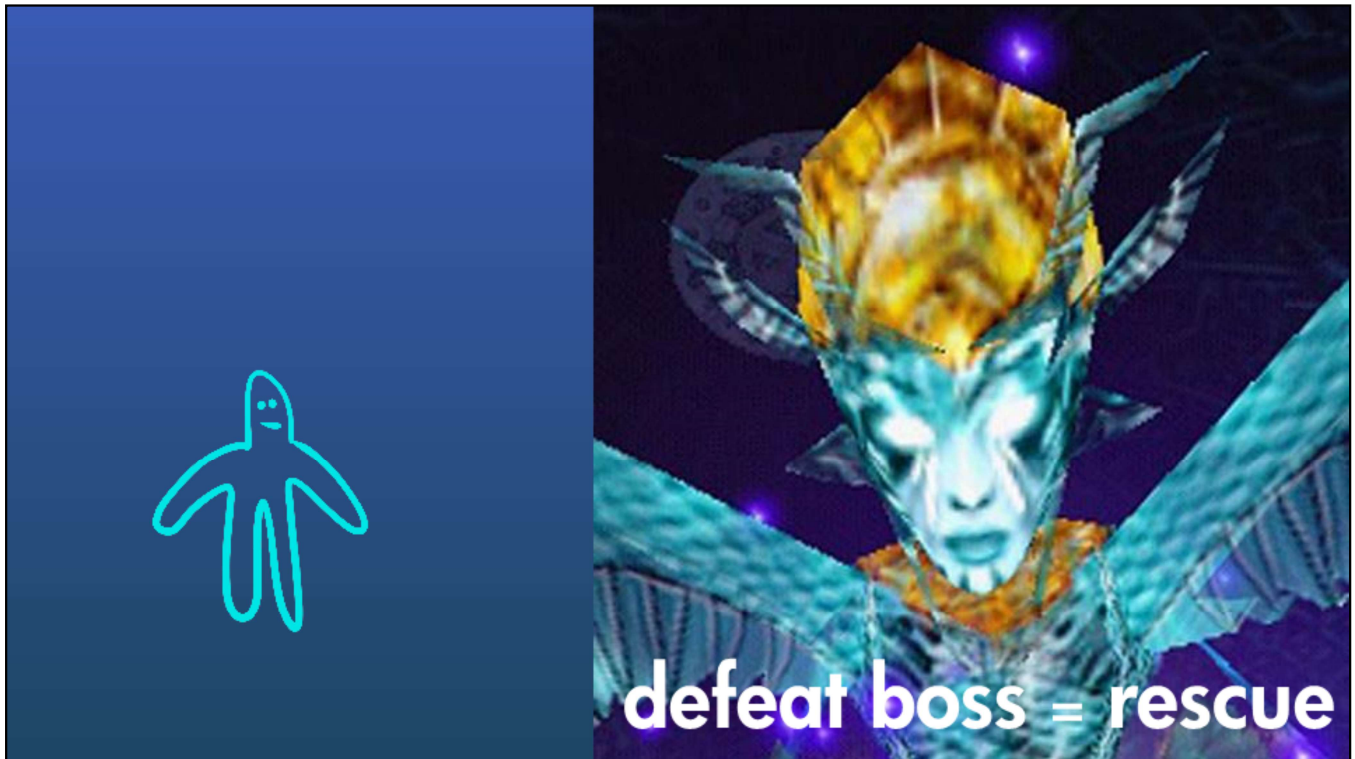
. . . and Indy rushes through just in time for the next phase of his quest. It's a co-op puzzle that anchors Sophia to the action — one of many in the game.



The finale is a boss fight. We needed to make this a supernatural climax, so we created an outrageous boss monsters. But defeating a decent boss is hard work, and this is where players drop out. We wanted players to finish our game. So the final boss, the Winged God Marduk himself, possibly imagining that Sophia is a like-minded menace, melds with her . . .

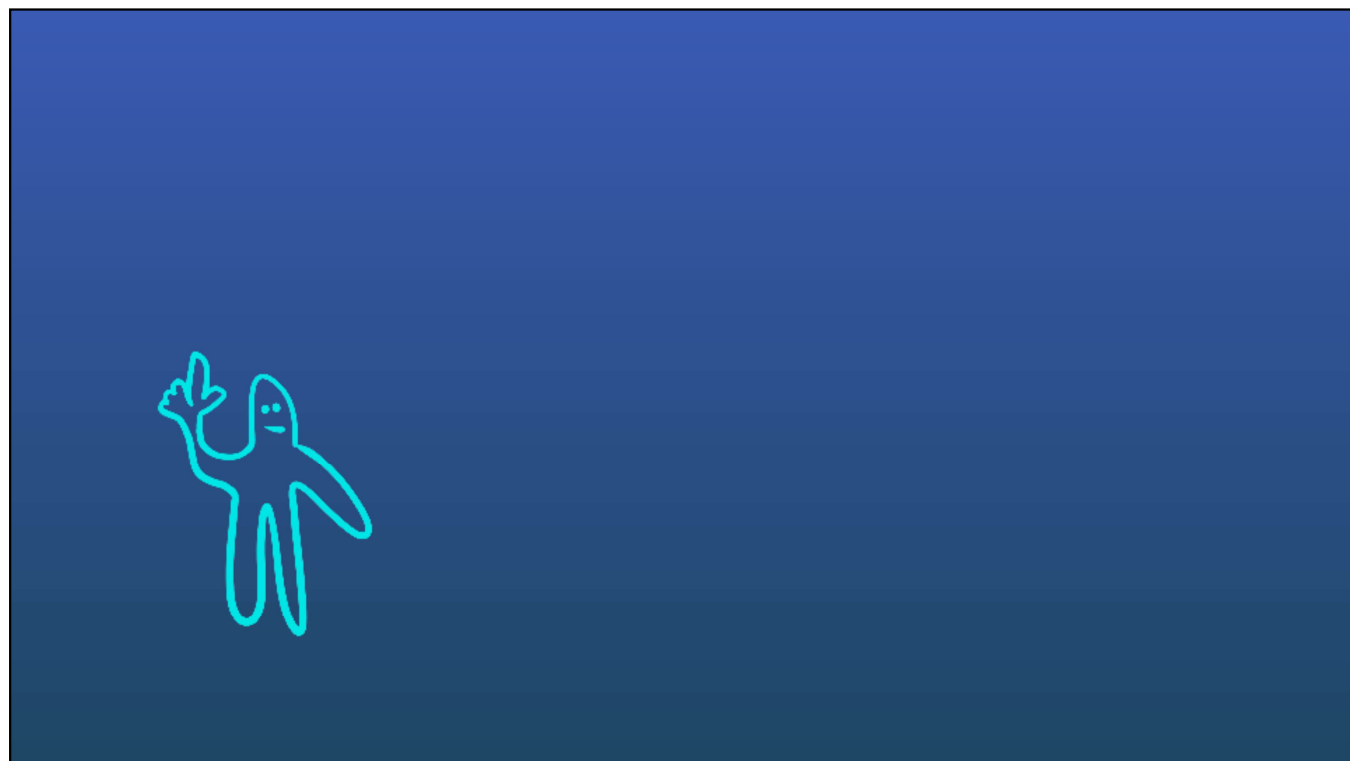


Defeating Marduk doesn't just earn a victory badge . . .



You're given emotional encouragement to win, and when you do, you've resolved a testy relationship and rescued a friend.







4 *problem:* incorporating NPCs in a mechanic

On my own I make small games. As a one-man band I'm constrained for resources, and the result is work that veers into the abstract



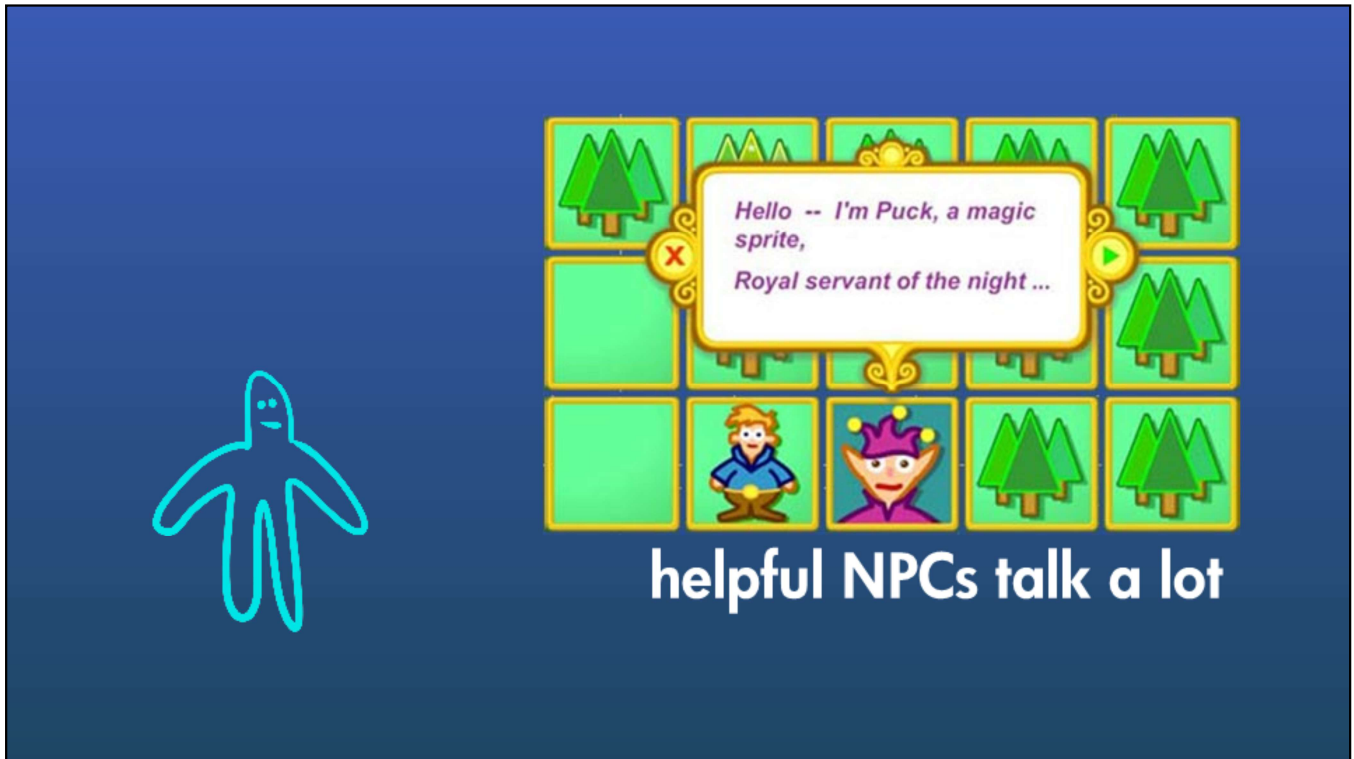
adventure / RPG

Thorn of the Midnight Rose is one of these . . . King Oberon and Queen Titania, late of Shakespeare's *A Midsummer Night's Dream*, are quarreling again. To save their enchanted forest from the evil Moon and his minions, the player must bring about reconciliation by pricking their fingers with a magic thorn.

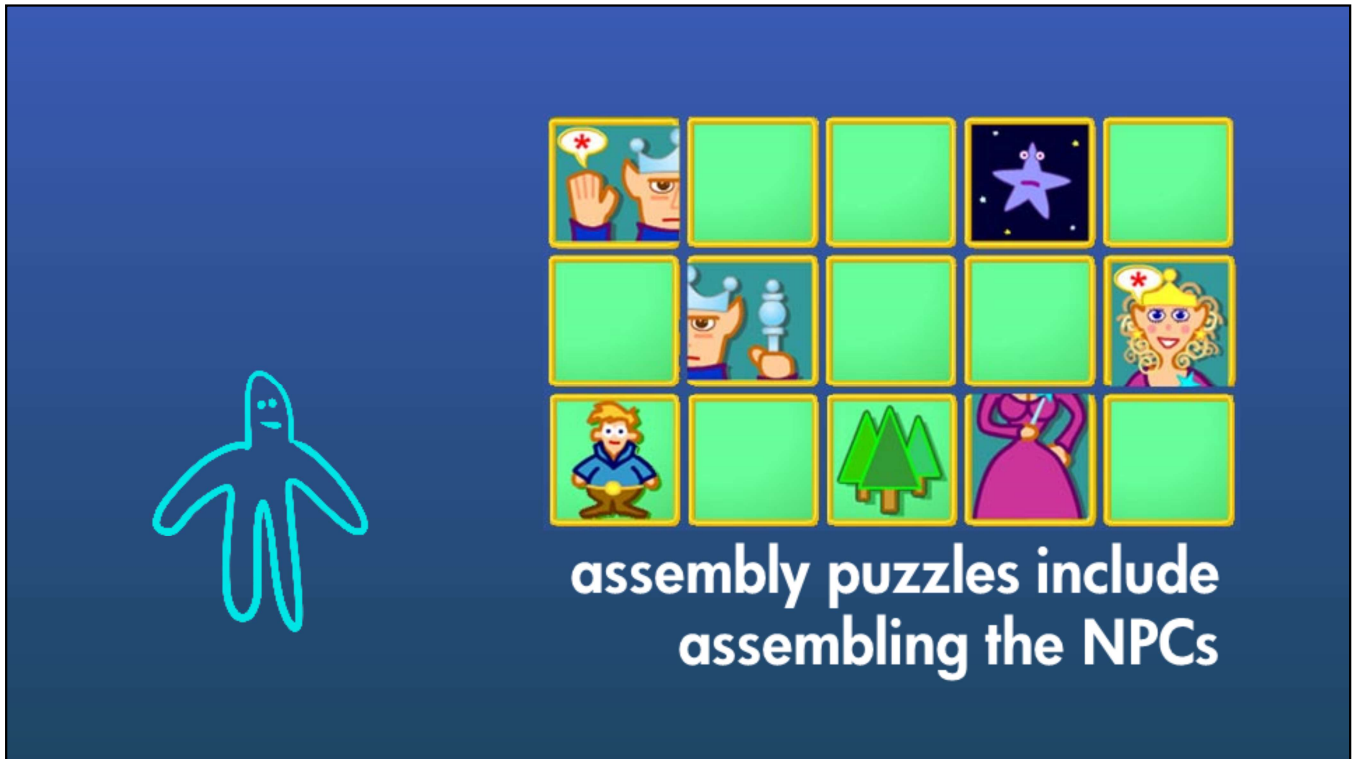


slider puzzle mashup

it's a mashup — an adventure / RPG played out as a sequence of slider puzzles.



The visuals are just icons. It's pretty darn abstract, so I made sure that to include a spirit guide — *Puck* — who has a lot of helpful resources to bestow, plus a lot of advice to offer, giving the game a human feel.



Assembly puzzles are a stock-in-trade of the game. What about the characters? Now that you've got 'em, how do you bring them into play? Assemble them, tying the story directly to the game mechanic . . .



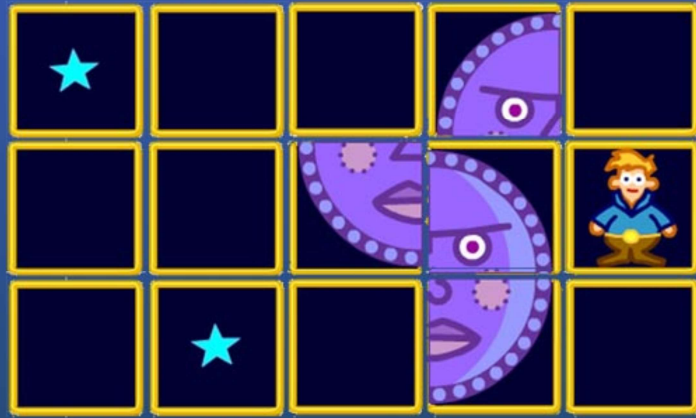
**assembly puzzles include
assembling the NPCs**



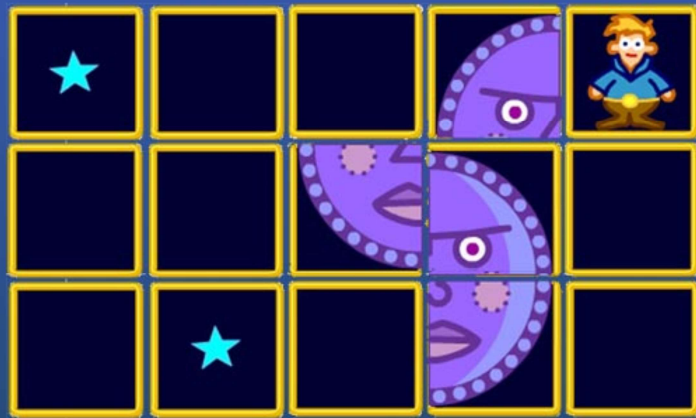
**assembly puzzles include
assembling the NPCs**



The finale starts with a challenging assembly puzzle. First build the boss . . .



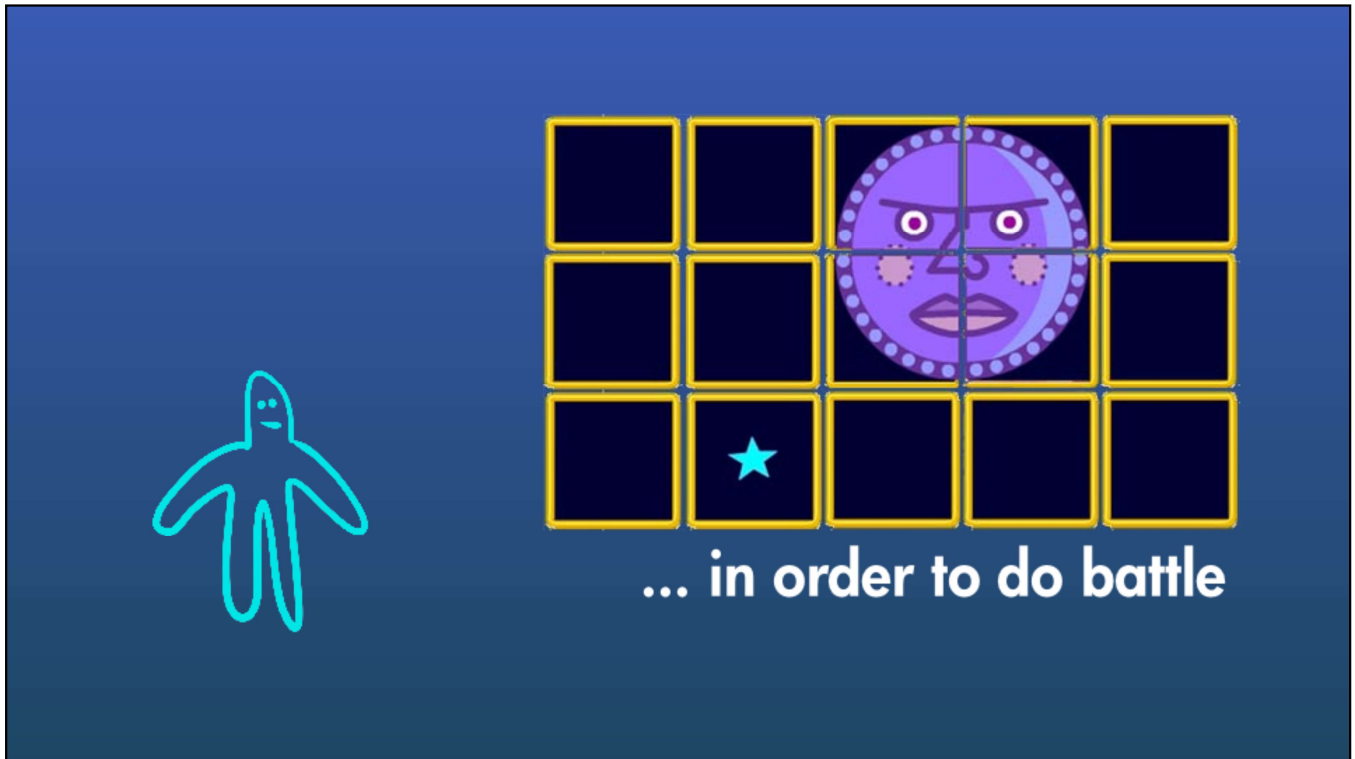
assemble final boss ...



assemble final boss ...

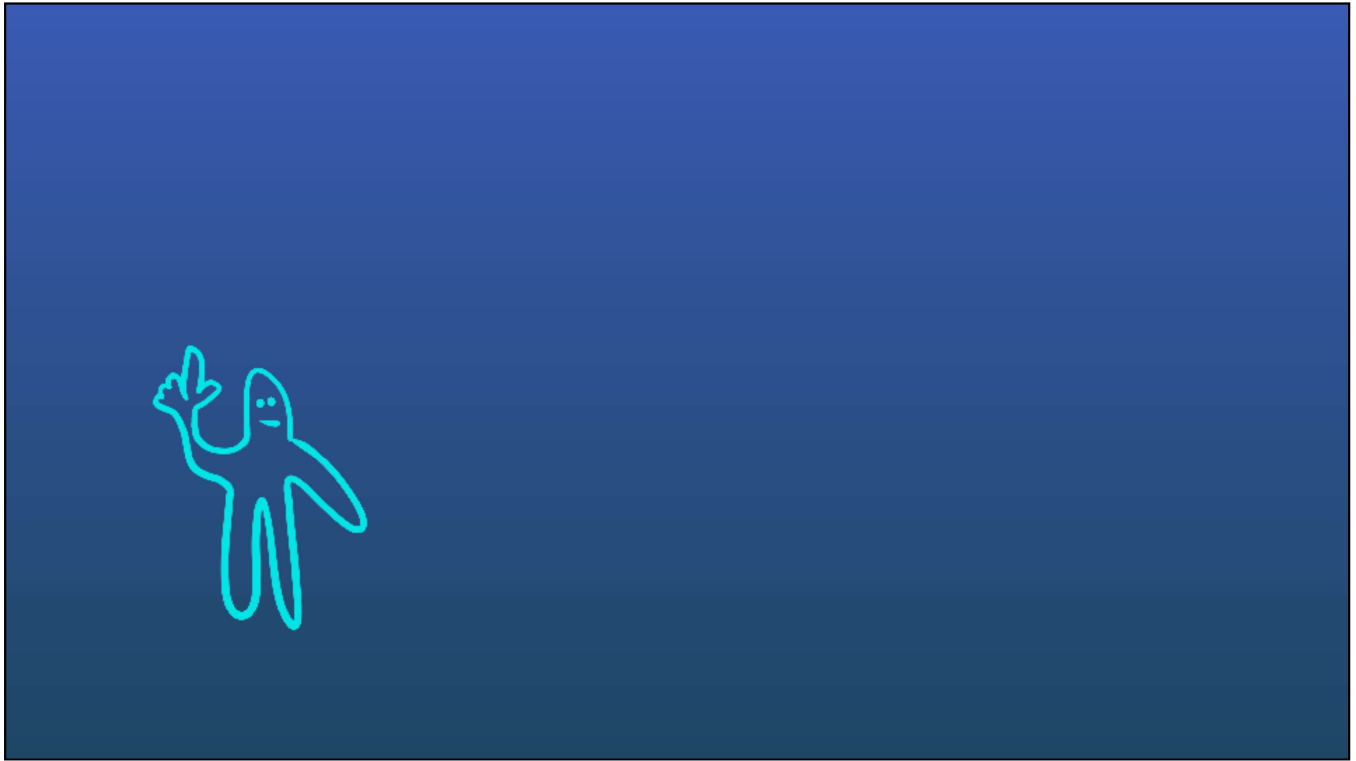


assemble final boss ...



Only after the evil Moon is assembled can the final battle begin.





Finally, if you think the problems I've cited are unique to static, old-fashioned, adventure-y games, think again . . .



The Last of Us

Here's a modern example that's completely the opposite. In *The Last of Us*, when a fungus turns people into zombie-like horrors, and a potential cure appears, the main character could have delivered it across the country in a little glass vial. The experience would have been as dismal as this scene.



Joel & Ellie

But instead, the cure is flowing through the immune system of an adolescent girl — Ellie, a flesh and blood character — and Joel's job is to deliver her across the country, warming the experience



co-op puzzles

And, like Sophia in *Infernal Machine*, Ellie doesn't just hang around. She takes over a winter level and otherwise figures prominently in the game play, as with the co-op rafting puzzle to fetch a ladder seen here.



the takeaway?

What's the takeaway?

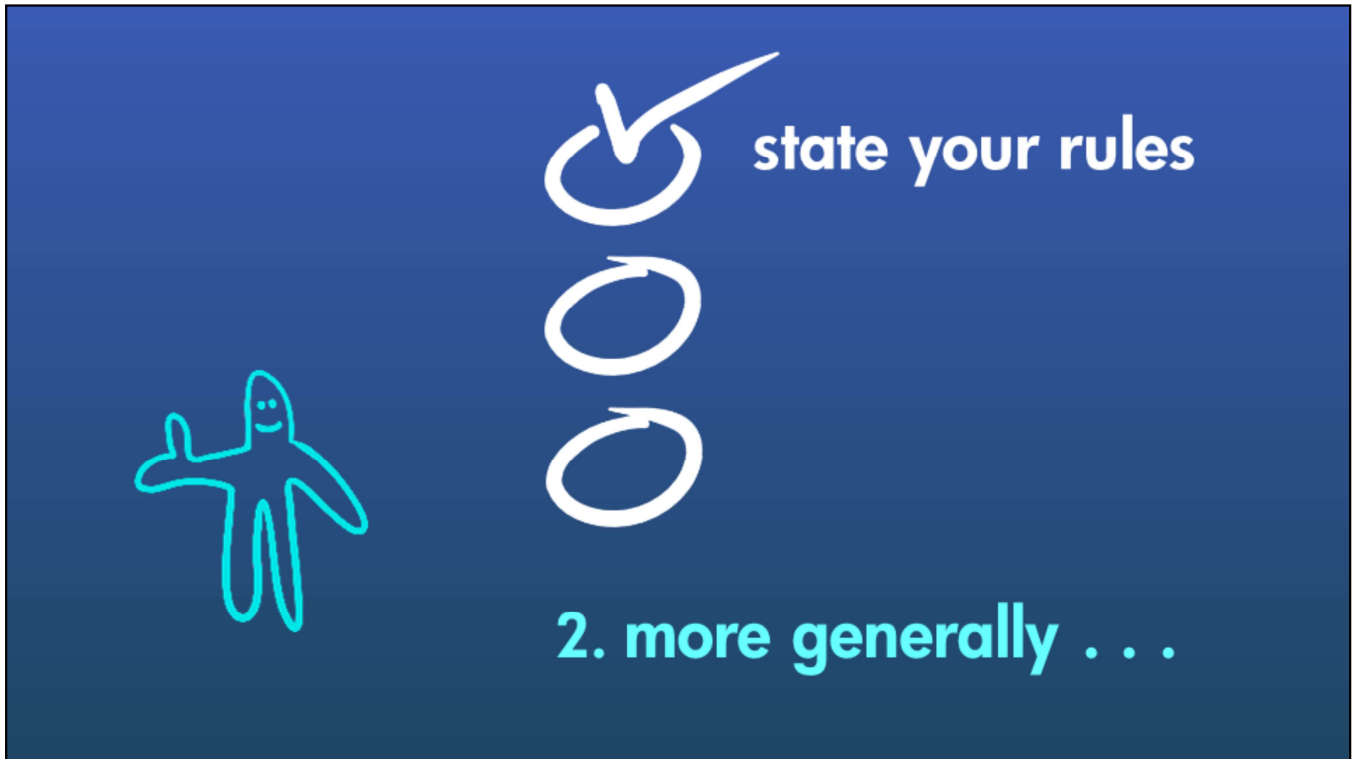


1. humanize your design

Be on the lookout for the many *UNUSUAL* ways you can solve design problems by turning them into characters. In game design, humanizing game elements is never a mistake.



2. more generally . . .



Learn to state your own rules explicitly. Explicit rules are much more powerful than implicit hunches.



Use them deliberately, like a checklist.



state your rules



run a checklist

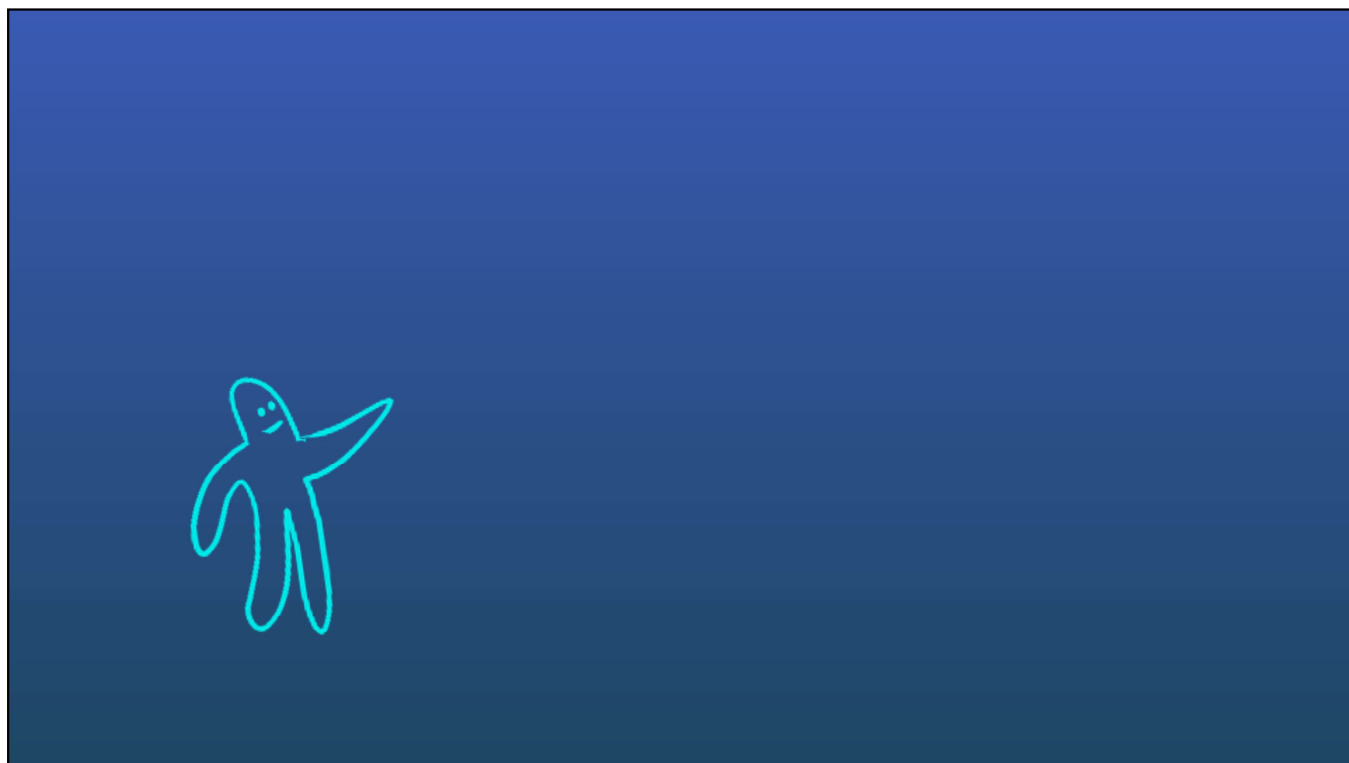


grow design chops

2. more generally . . .

Consciously used and checked, your rules will greatly enlarge your design vocabulary.





www.finitearts.com



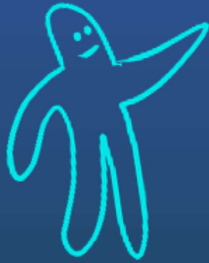
My website . . .

www.finitearts.com



. . . (Finite Arts LLC is my personal service and publishing company) . . .

www.finitearts.com



- see rule collection
- see my old rule talks
- play *THORN*
- play *YODA*



. . . where you can view my older talks about rules, see the 400 Project's rule collection, even play *Thorn* and *Yoda* there if you want . . .

www.finitearts.com



- see rule collection
- see my old rule talks
- play *THORN*
- play *YODA*



. . . where you can view my older talks about rules, see the 400 Project's rule collection, even play *Thorn* and *Yoda* there if you want.

www.finitearts.com



- see rule collection
- see my old rule talks
- play *THORN*
- play *YODA*



. . . where you can view my older talks about rules, see the 400 Project's rule collection, even play *Thorn* and *Yoda* there if you want . . .

www.finitearts.com



- see rule collection
- see my old rule talks
- play *THORN*
- play *YODA*



. . . where you can view my older talks about rules, see the 400 Project's rule collection, even play *Thorn* and *Yoda* there if you want . . .

HAL BARWOOD

Indiana Jones & The Fate of Atlantis
Indiana Jones & The Infernal Machine

Writer / Designer
Finite Arts
www.finitearts.com

To those of us who work in narrative I think there's a beautiful simplicity to Hal's rule – that we almost take it for granted. Yet so many games fail to do it, I'm glad Hal got up here and reaffirmed why personifying your gameplay problems and vice versa is so important...

LUKE MUSCAT

Creative Director
Prettygreat
@pgmuscat

And FINALLY

Luke is the designer of exceeding popular games you may have heard of like Fruit Ninja and Jet Pack Joyride, and now is creative director at his own studio PrettyGreat where they have shipped Land Sliders among other titles.

And EVERY YEAR when we do this session someone says "Yeah, but rules, really?" And this year it's Luke's turn.



Forget Rules
Make **Goals Your King**

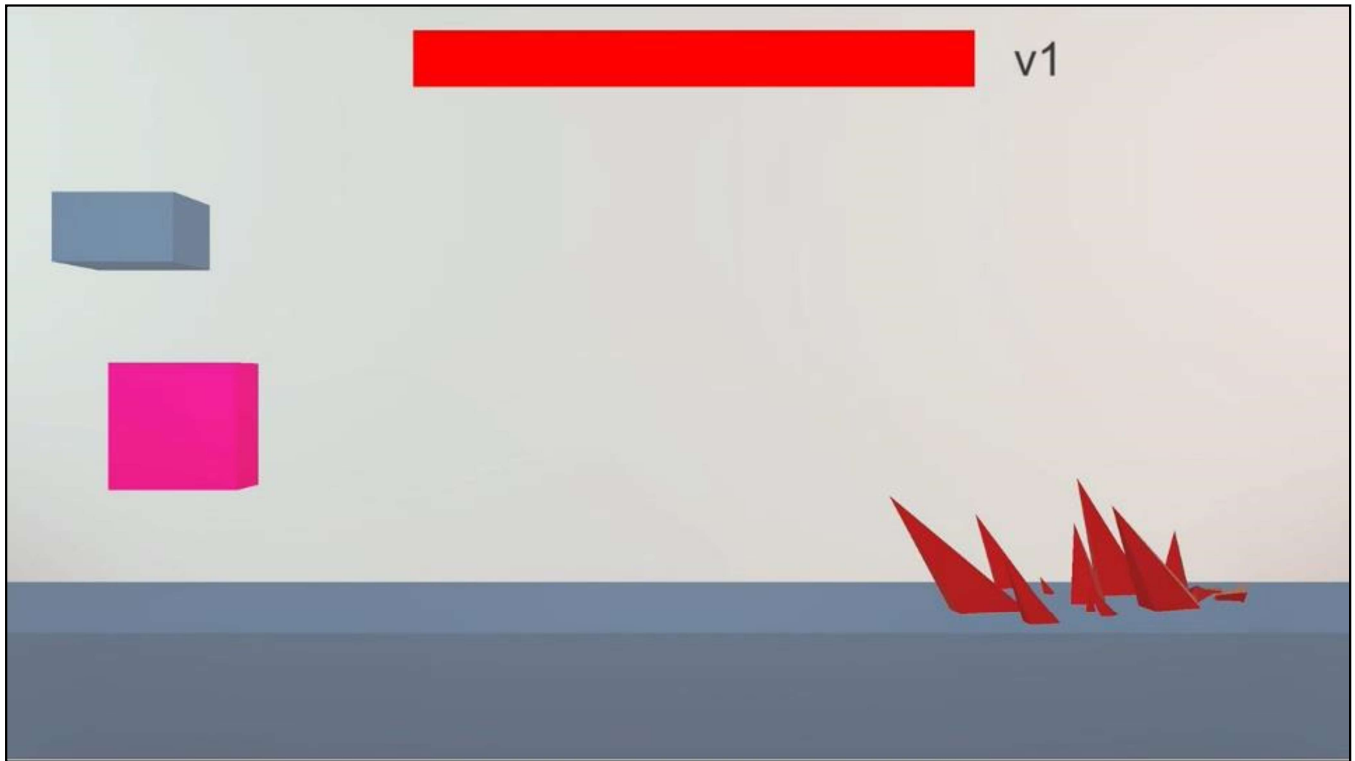
fuck rules





sprint goal: the game is playable

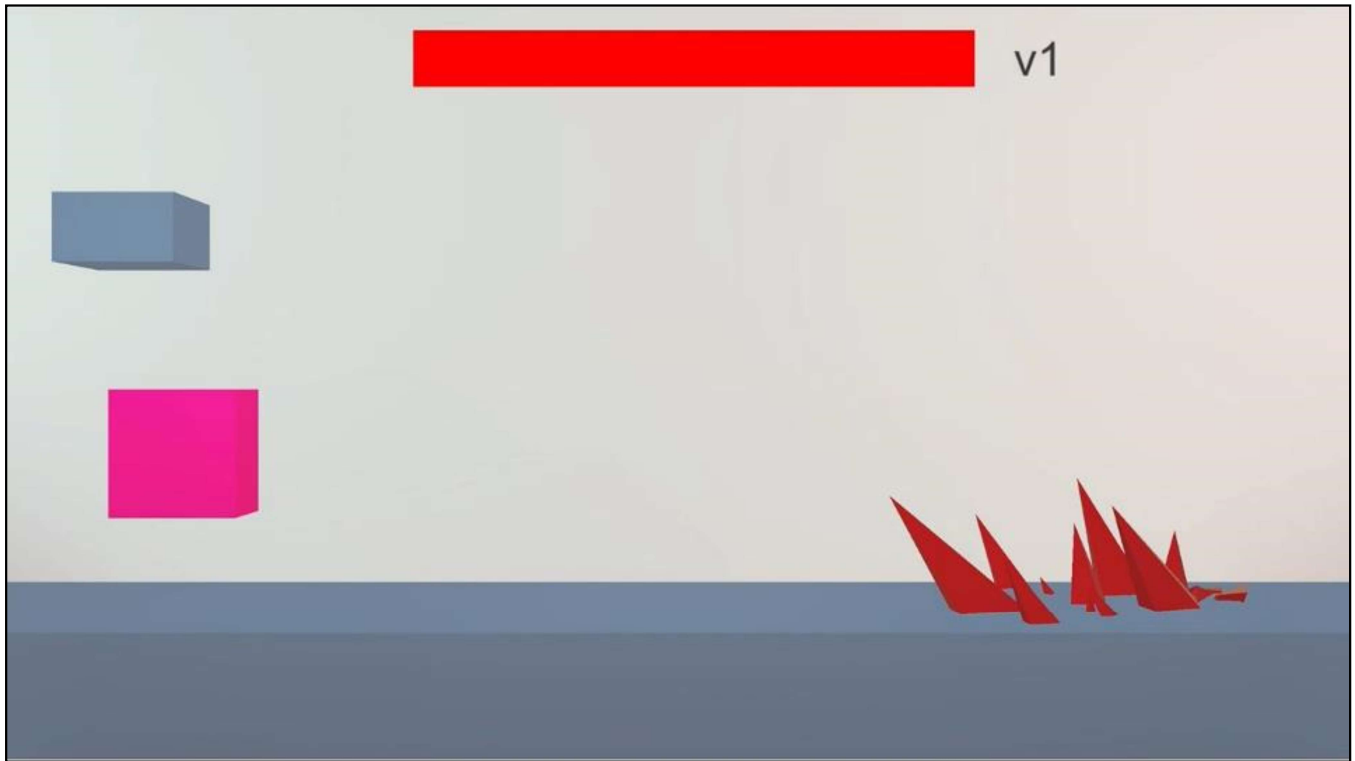
user story	description/tasks	effort
There is a health bar	<ul style="list-style-type: none">[] Player can take ~10 hits[] If health bar == 0, player dies[] displayed to middle of screen	10



Player Feedback



It's about the joy of Mario jumps! Fireballs! Bashing a block!

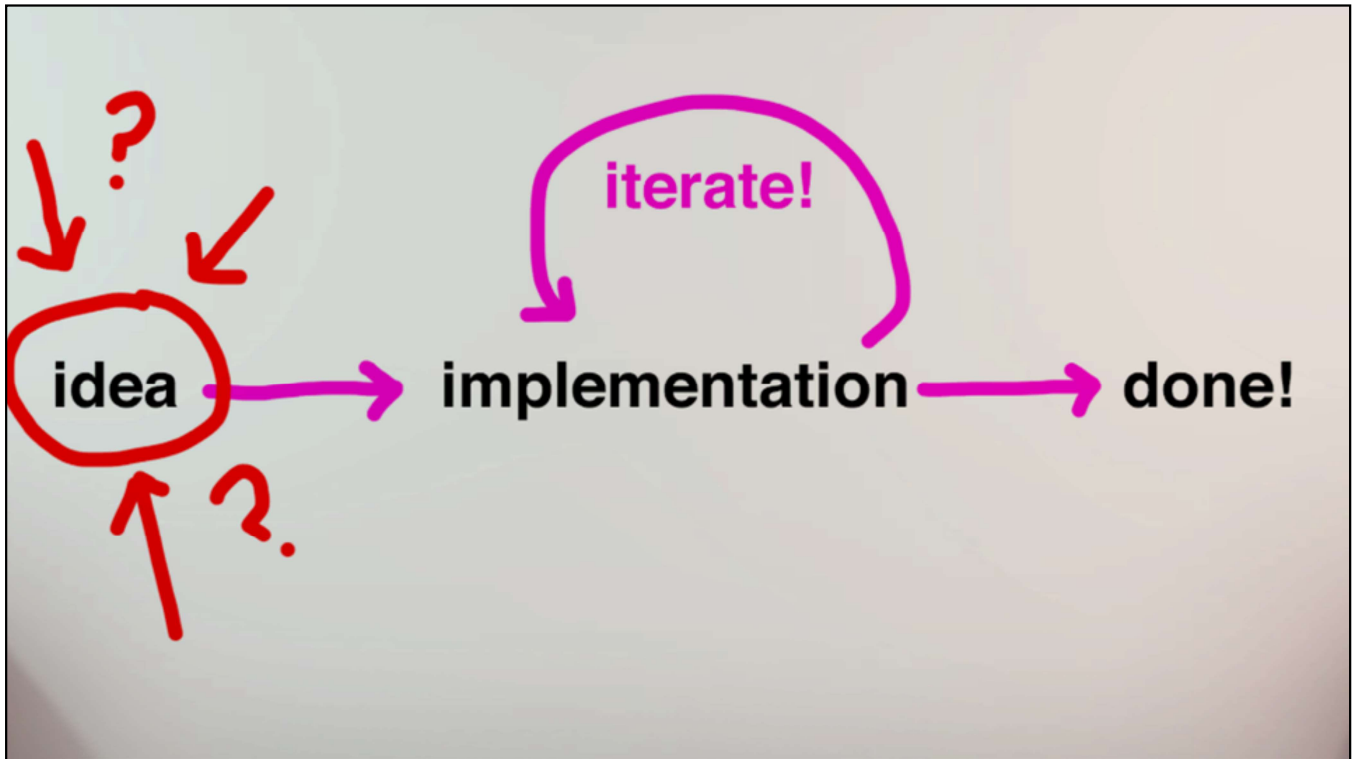




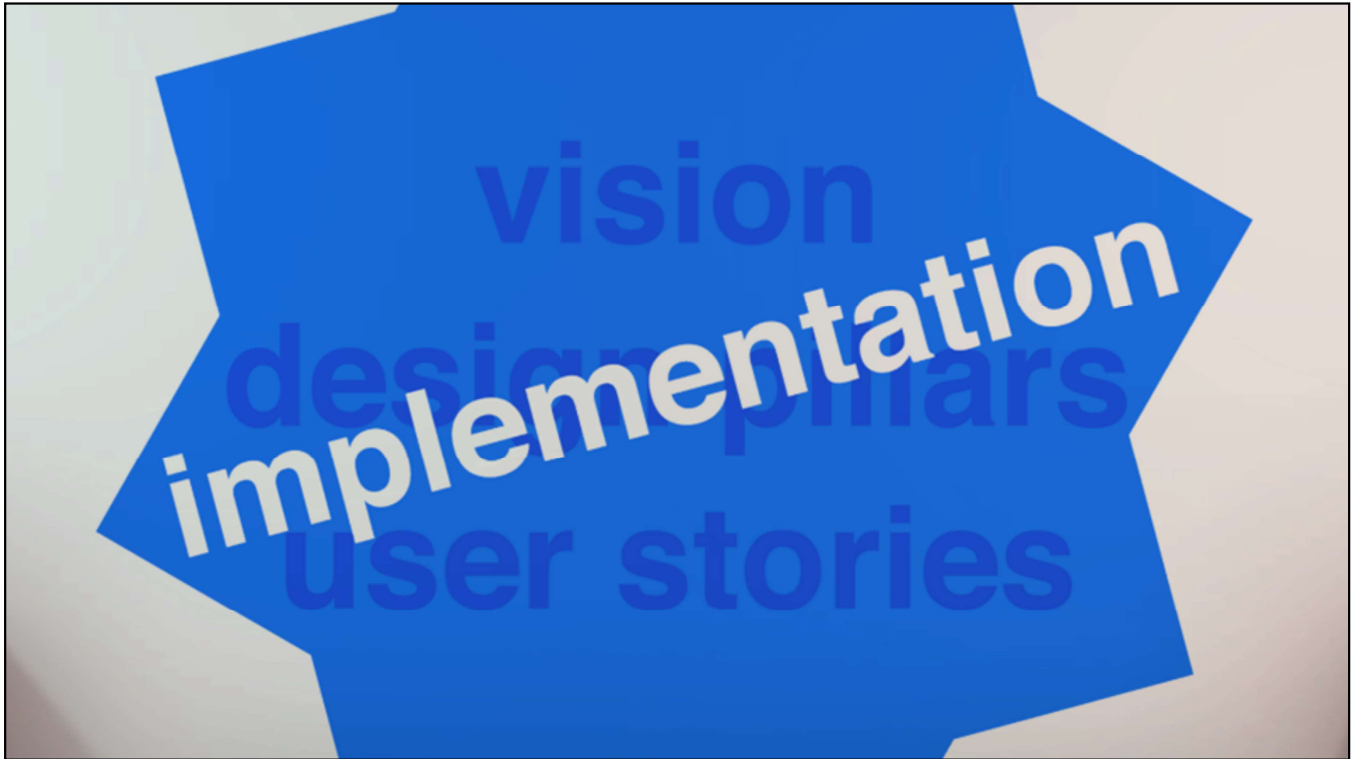
luke



that health bar
4eva



vision
design pillars
user stories



sprint goal: the game is playable

user story	description/tasks	effort
There is a health bar	<ul style="list-style-type: none">[] Player can take ~10 hits[] If health bar == 0, player dies[] displayed to middle of screen	10

sprint goal: the game is playable

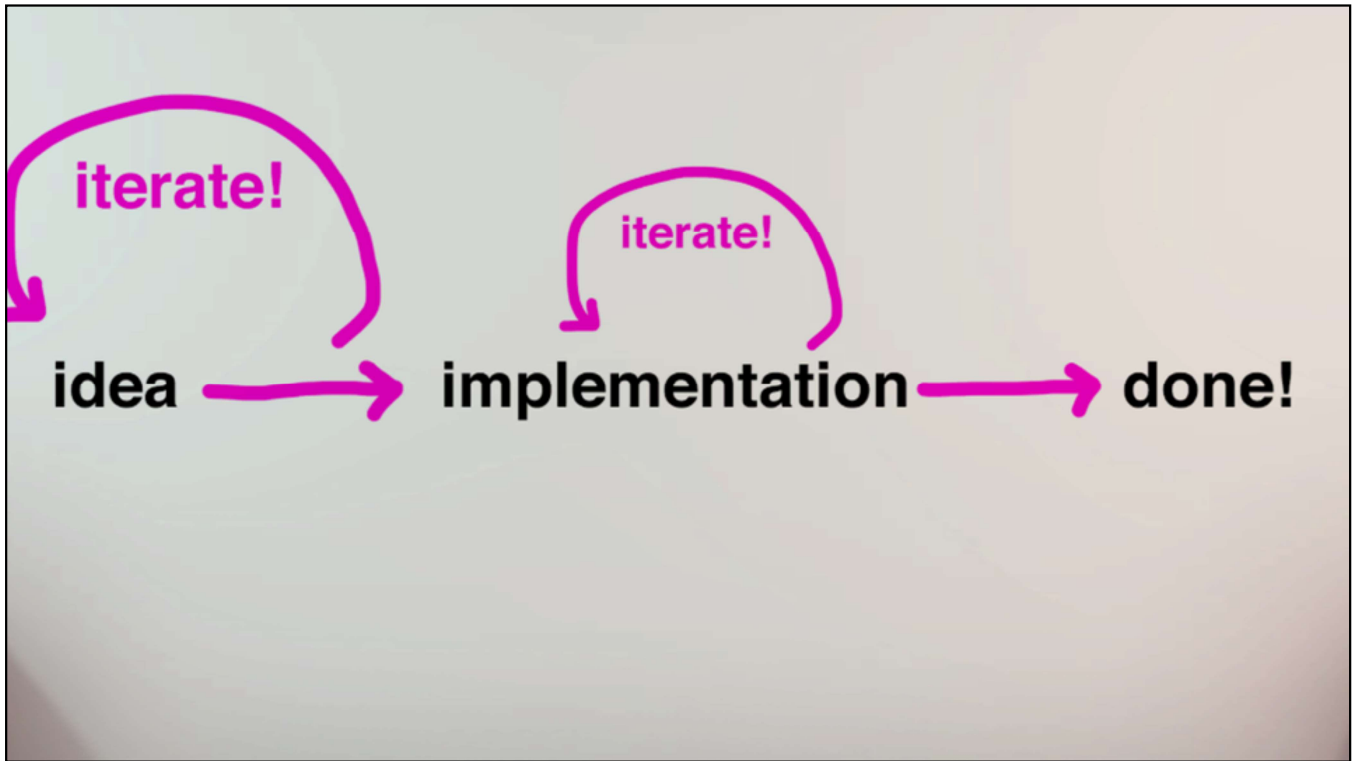
goal	description/tasks	effort
The player can take multiple hits and knows when they will die	<ul style="list-style-type: none">[] Player can take ~10 hits[] If health bar == 0, player dies[] displayed to middle of screen	10

sprint goal: the game is playable

goal	possible solutions	effort
The player can take multiple hits and knows when they will die	<ul style="list-style-type: none">[] there is a health bar?[] COD style screen fringe (can't take too many hits too fast)?	10

sprint goal: the game is playable

goal	possible solutions	budget
The player can take multiple hits and knows when they will die	<ul style="list-style-type: none">[] there is a health bar?[] COD style screen fringe (can't take too many hits too fast)?	10

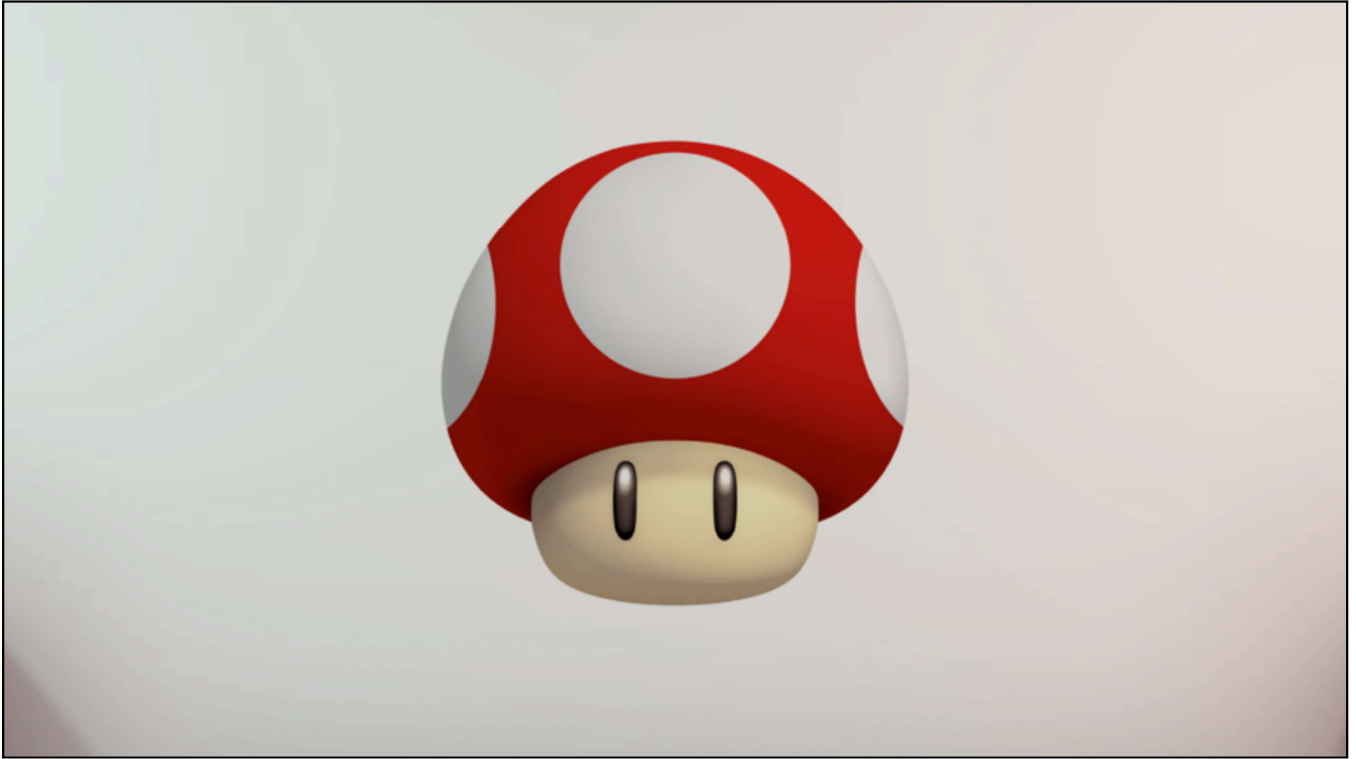




sprint goal: the game is playable

goal	possible solutions	budget
We can create exciting moments where the player barely scrapes through an area	<ul style="list-style-type: none">[] health bar and <3 packs[] every hit you take, death spawns and starts chasing you[] time *= 0.4; for a moment when hit. player dies if time stops	10





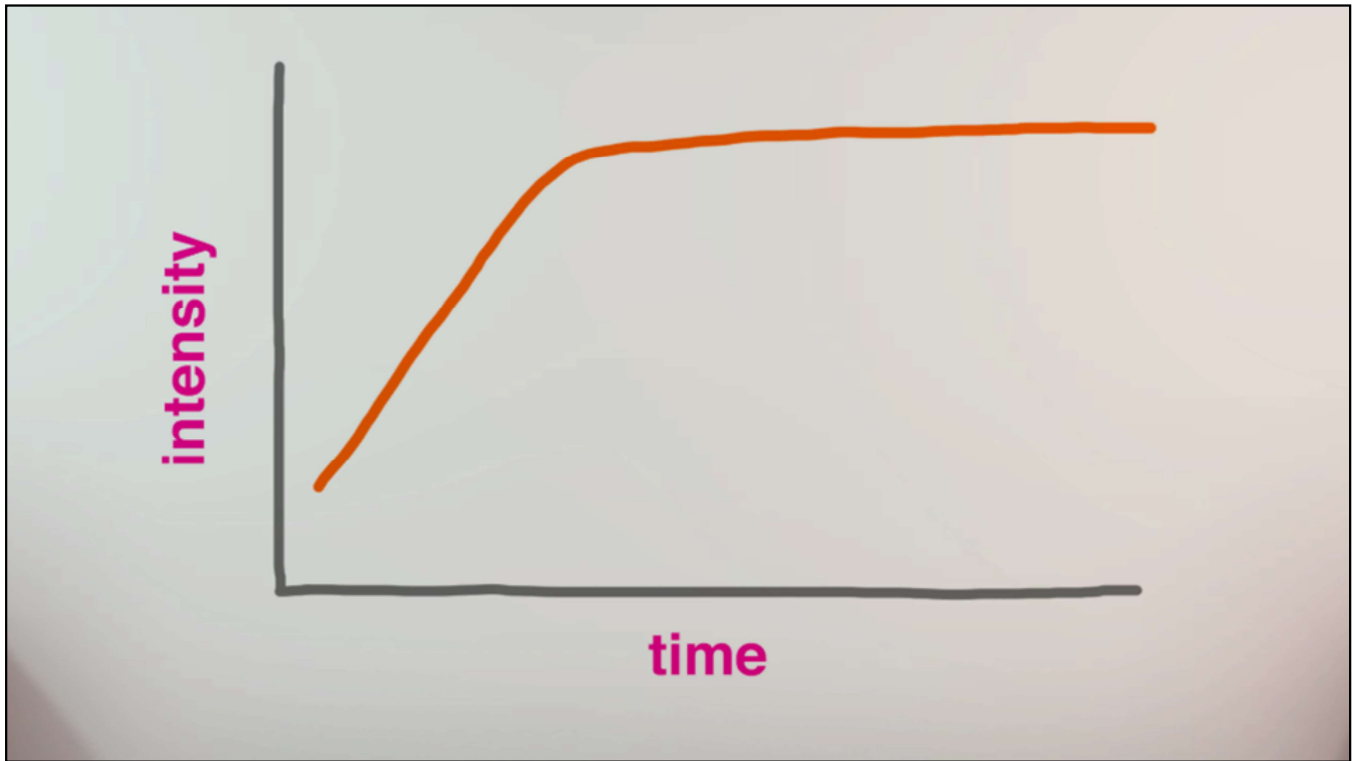
sprint goal: get ready for release!

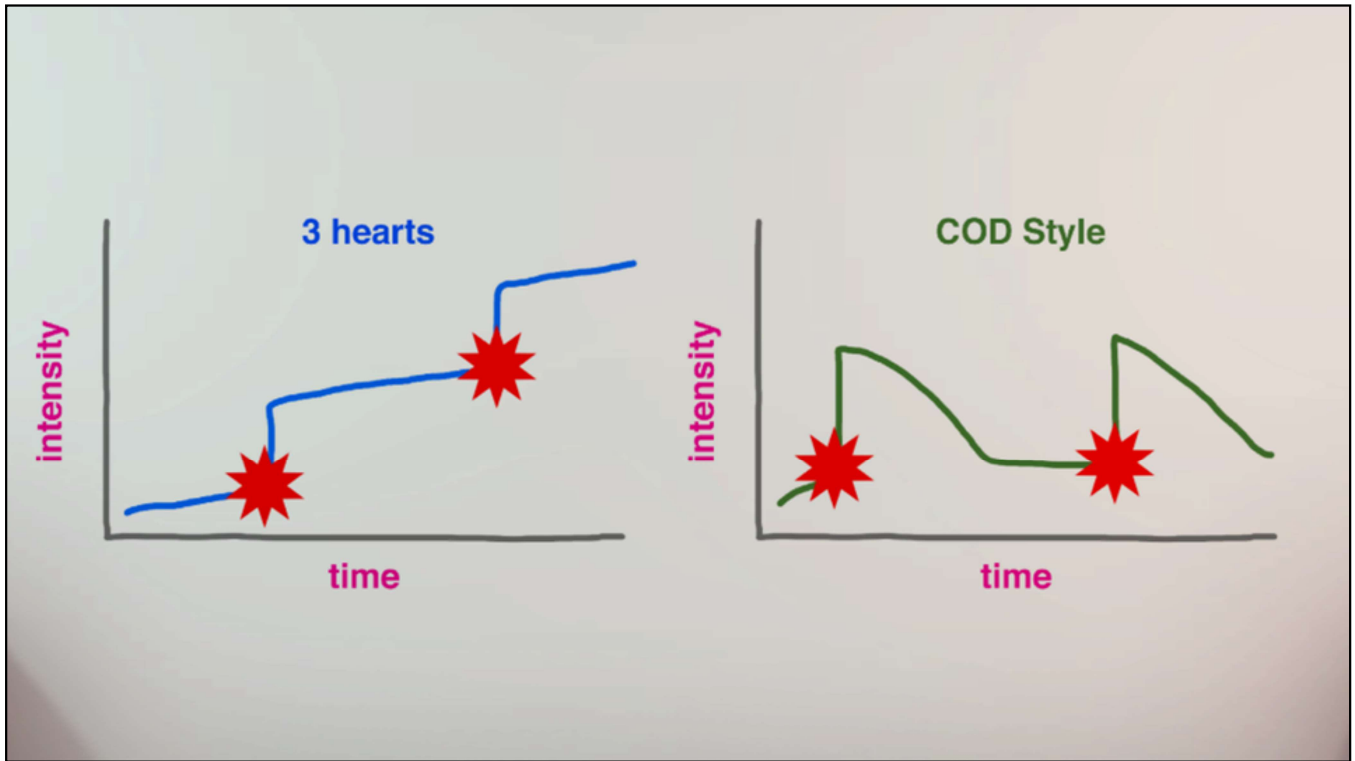
goal	possible solutions	budget
Make the game more interesting	<ul style="list-style-type: none">□ Power ups?□ _('')_/	10



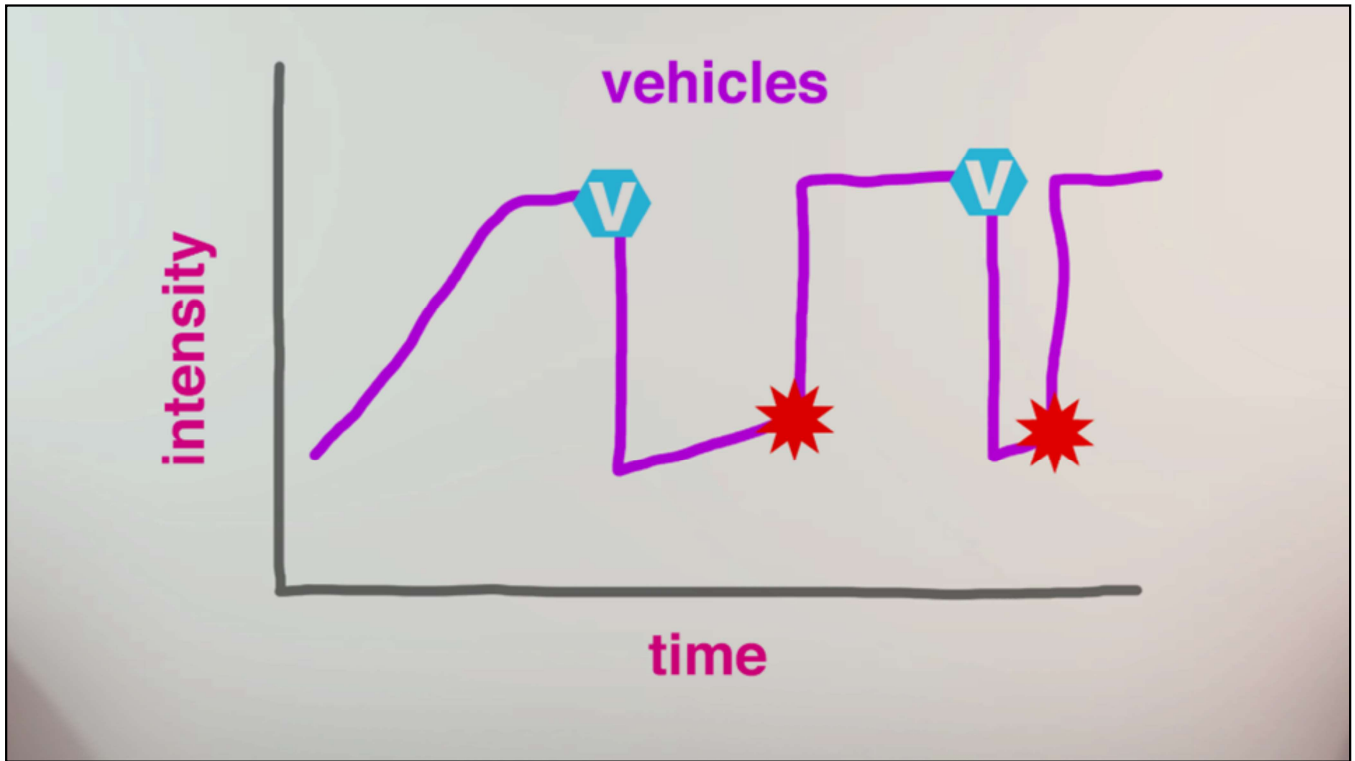
sprint goal: get ready for release!

goal	possible solutions	budget
Give the game a more compelling variety in intensity	<ul style="list-style-type: none">□ Power ups?□ Health bar?□ COD health?	10







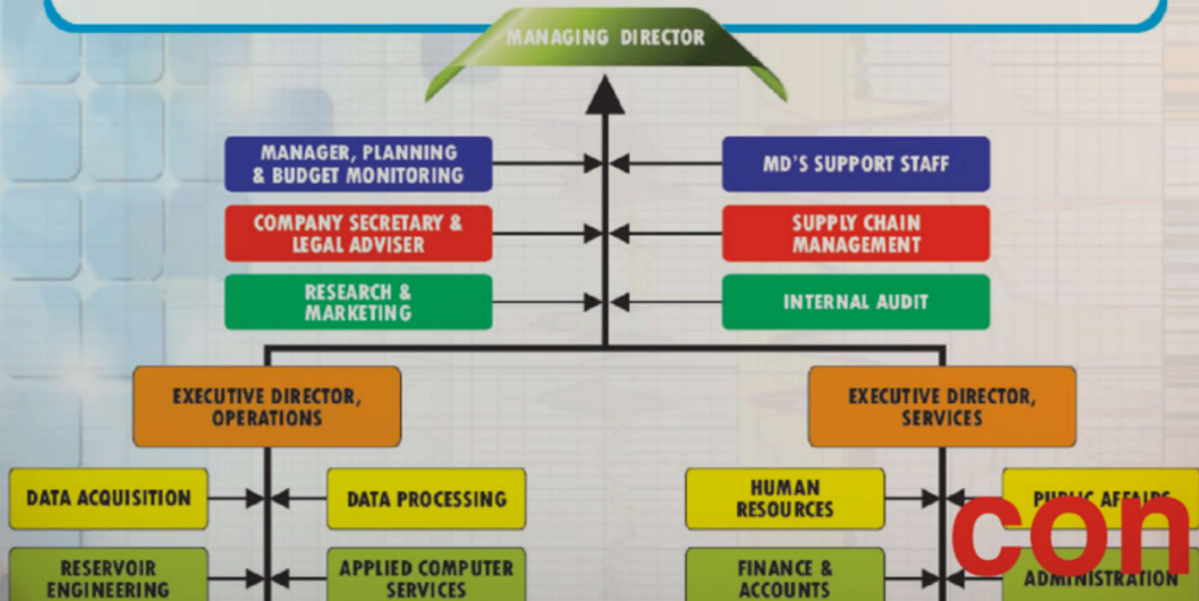






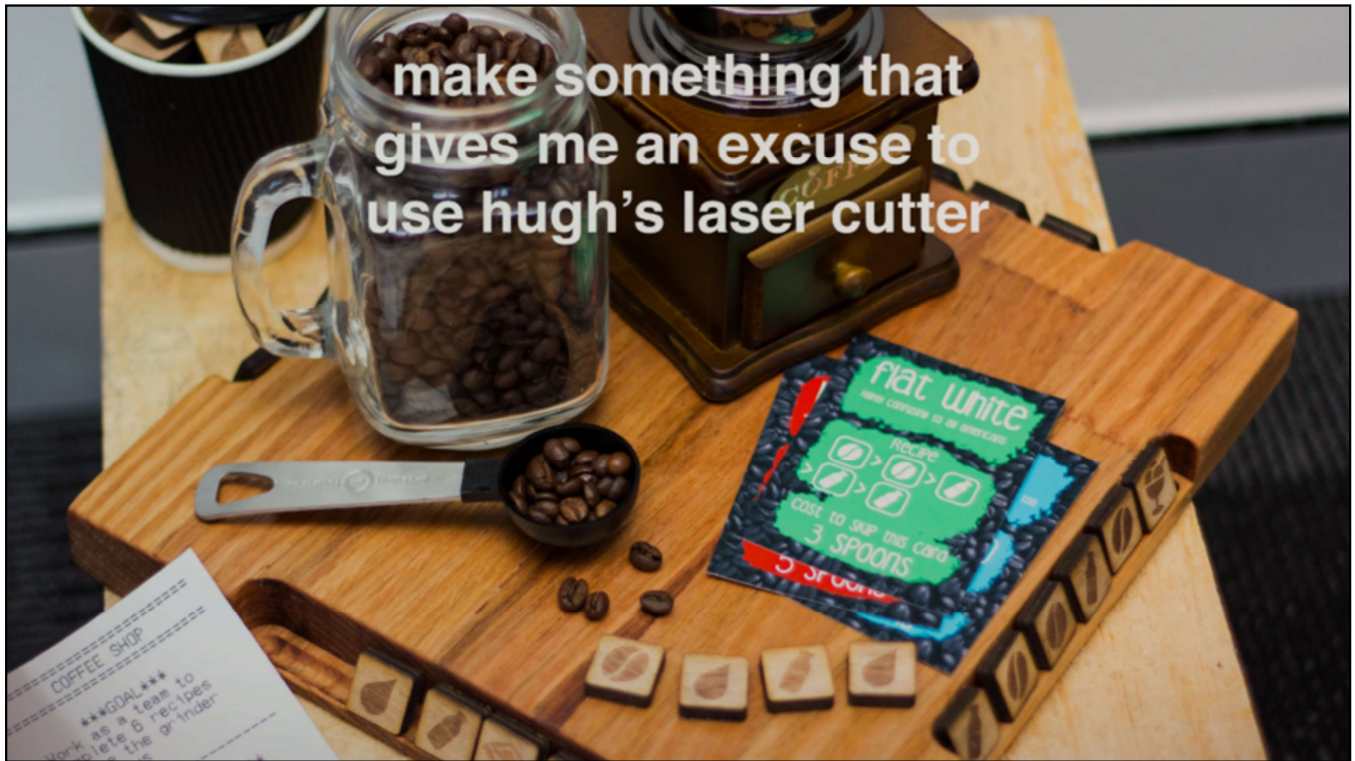
MANAGEMENT STRUCTURE

IDSL is managed by an efficient and dedicated workforce. The Management is structured into OPERATIONS and SERVICES Divisions, headed by a Managing Director



cons







**make a 32 player realtime multiplayer game
that works on a 2G connection**





LUKE MUSCAT

Creative Director
Prettygreat
@pgmuscat

Thanks Luke, for transforming the paradigm entirely.

**EVERYTHING
YOU KNOW
IS WRONG**

So with that I hope you see how our speakers have subverted your way of thinking and given you a new perspective on game design challenges

But I think it's important to remember about our theme is the correlary.

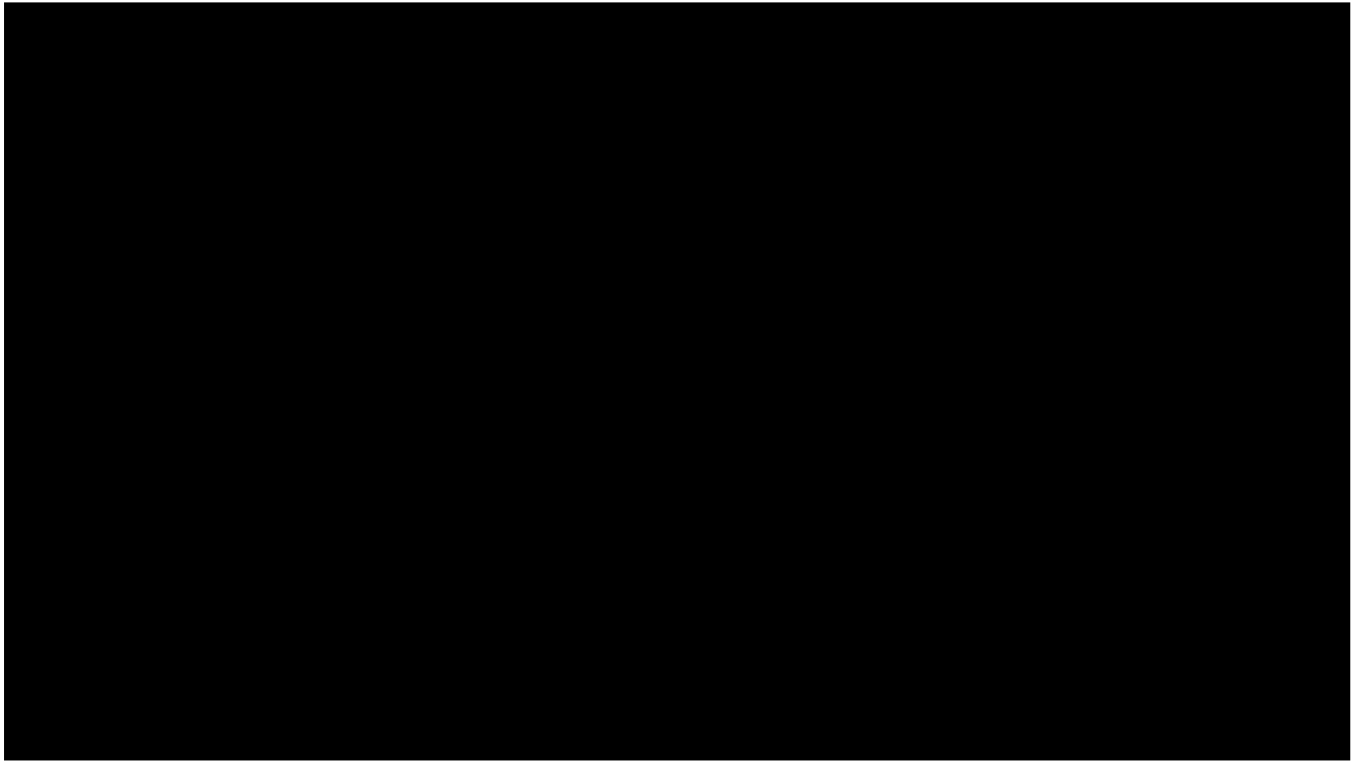
NO ONE KNOWS ANYTHING

Dark musing courtesy of William Goldman

Which is NO ONE KNOWS ANYTHING

These are the rules presented by our speakers here today,
but they may or may not apply to your project or this
particular time or you ever

Because they're not really sure either.



BUT MAYBE when you are in your own darkest moment, look back at these rules

See if one of these can help you of that funk

WHAT ARE YOUR RULES?

But most importantly, I hope you leave this session wondering what YOUR rules are, that you're going to make the games only you can make

Rules of the Game

With your host **Richard Rouse III** @richardrouseiii

Chelsea Howe “Design by Fractal”
@manojalpa

Damion Schubert “Free is the Default”
@ZenOfDesign

Christina Norman “Embrace Radical Constraints”
@truffle

Hal Barwood “Personify Problems”
www.finitearts.com

Luke Muscat “Forget Rules, Make Goals Your King”
@pgmuscat

Slides at: www.paranoidproductions.com

Much thanks to Amy Hennig & Soren Johnson

Thanks to our very helpful advisory board contacts Amy Hennig and Soren Johnson

We're not going to do Q&A but we'll be hanging around up here for as long as they'll let us if you want to come ask us some questions.

Thanks everyone!