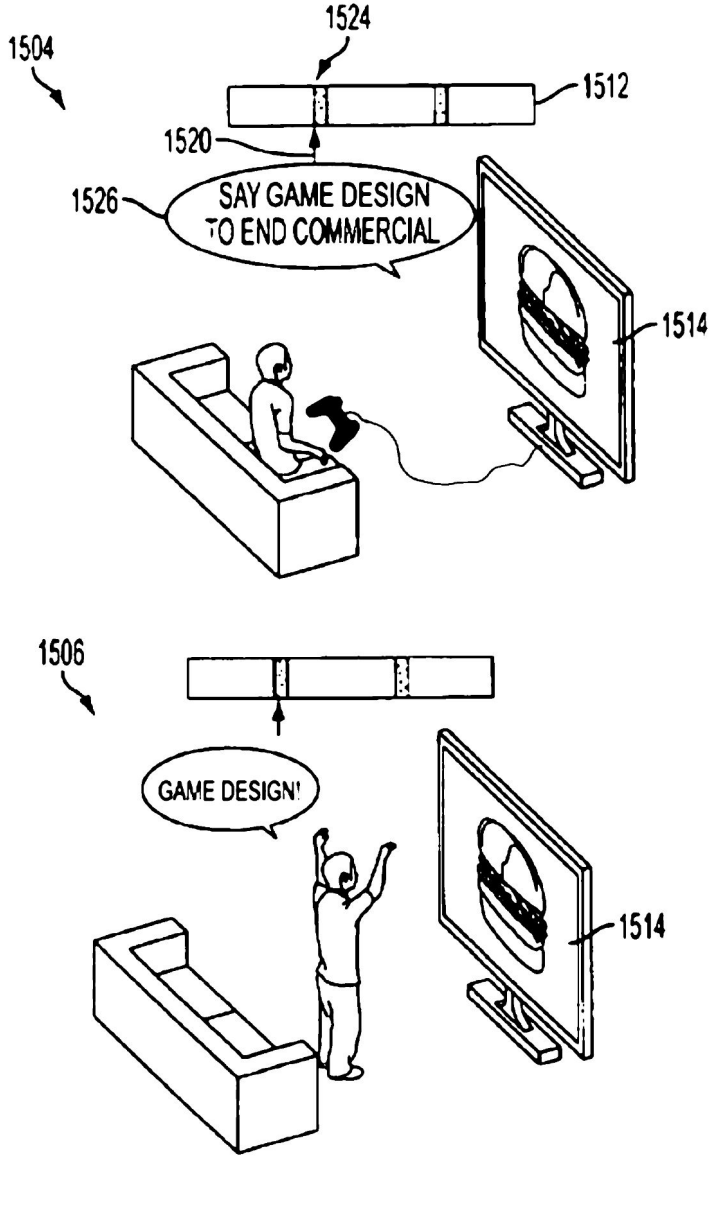
PARADISE Zine



Issue. 1

Video games are art. However, videogames as an act of vandalism are a crime. Every state/province has vandalism laws that apply to video games, and local entities such as cities or counties have anti-videogame ordinances.

Violation of these laws can result in a fine, probation, or a jail sentence.

Paradise does not condone the real life act of vandalism in any form.

Skate Games

Words and illustrations by Mut/Moochi

Design by Mer Grazzini

Recently, Ive been thinking a lot about skateboarding and in particular skateboarding videogames. A big reason is that I've been developing my own skate adjacent game, bruxa, with my friends at xinela. When starting the game we didn't think of it as a skate game, you couldn't even be very expressive with the movement! But most of the work I ended up putting in the game was in making the movement feel more interesting, and thinking about how level design would relate to that, and thinking of how to show feedback/gamefeel/juice of an extremely contrived and arbitrary movement system that arose from following the kind of interactions that made me enjoy playing the game.

— To throw some references that have been really inspiring: Stephanie Boluk and Patrcik Millieux's Metagaming (of course), Bernie Dekoven's The Well Played Game, Tim Rogers's In Praise of Sticky Friction, Andrew Yoders' Playground and Level Design and Frank Lantz's Counting Frames, anything David Kanaga has ever wrote.

Added to that, i've been thinking a lot about how videogames ARE real life (and I think this is nothing new, Juul's Half Real is basically bringing up this argument 20 years ago), and how to embrace more of this (messy) aspect of game, make more games that blur these boundaries, that make clear the arbitrariness of the magic circle. All of this recently culminated and getting to read a series of blog posts (letterclub.games; from other friends that are also in this zine) about the (still very raw) concept of:

Emersion.

By having rules not computationally defined these video games open the doors to see outside of them.

Emersion

, as opposed to immersion

, not to be confused with emergent, which is a characteristic of a system, not the player.

This concept seems to fit like a glove as this way to describe designing towards pushing players to experience the game as reality. To push them towards the outside of the magic circle (instead of the usually more "artistic" approach of shattering the circle altogether).

I have 2 different ways to see emersion right now:

While playing the game*, the player notices something about the real world. It can be about their body, or about their reality. Or about their usual relation to the magic circle.

An example that comes to mind is droqen's asphyx, or die Gute Fabrik's B.U.T.T.O.N. By having rules not computationally defined these video games open the doors to see outside of them, being constantly aware of the possibility of "cheating".

It can also happen that the game continues to exist outside the magic circle. You learn something new about your body, our how the world works, and can relate back to the game.

A classic example of this is the tetris effect, thoroughly explored in The Witness. This is also I believe an unexplored part of movement based videogames, and where my desire to write about skategames came from.

(It's interesting that even when talking about breaking the magic circle, there's still a construct that I am using here, which is "really" playing the game. Maybe it's inescapable).

So, what about skategames?

I think when I say skategames, 2 games will immediately come to mind, Tony Hawk Pro Skater (THPS) and Skate. I have a feeling that people that talk about the former and are thinking about skateness, are already arguing in their heads how Skate is much more skategame than Tony Hawk. I will now make an important, and extremely confusing distinction, skategames vs skate games, or games about skateboarding. (This is inspired by Charles Pratt's also annoying and confusing, but interestingly thought provoking distinction between video games and videogames, which I will not attempt to explain here). The important thing to understand is that the skateboard itself is probably the ultimate skategame, and thus, attempts to recreate it digitally tend to be skategames as well, but most of said attempts fail or sometimes fail forward producing unintended emergent skategames.

The core principle of the skategame is the ability to do tricks, and to express yourself outside of the standard metagame of the game. I believe that standard metagames that are directly related to the tricks themselves is usually the problem* of skategames like THPS (and even Skate). This brings us to the first, and most boring axis of skategames

1. how much do the tricks reflect real life, how real are the tricks. (i.e: does your game have anything like a kickflip?)

*these games have kickflips builtin to the skate object, so even if you can perform a thing that looks like a kickflip, it has nothing to do with what an actual kickflip is, the kickflip is not builtin to the skateboard, nor does the skateboard acknowledge the kickflip, which brings us to the next point.

Almost any realtime videogame can be a skategame (would love if anyone has thoughts on non realtime skategames. dh says: "Magnus Carlsen is like fucking Tony Hawk", expressive narrative). If you can move in realtime, you can probably do tricks (even if they are boring), having skateboards as a benchmark for this is probably the main reason why I chose skateboards and not parkour, or dancing. Its an external analog designed object that you use your body to perform on. And so comes the second axis of skategames:

2. the analog/digital, athleticism axis of maneuvering the skateboard. (i.e: how much of your body are you using to perform said kickflip?)

this is the axis where the argument towards Skate usually stops at, since making an ollie in skate is not a single button press, but a shoryuken-like input, people tend to like to say it's more like skateboarding. I disagree, I think Skate is more like skateboarding because the analog and glitchy inputs make non skateboard tricks possible, tricks that only exist in Skate. (which maybe points to the existence of Skategames, with uppercase S, but I digress).

Finally, a thing that differentiate skateboards from other bodily toys (like juggling or yoyos), is the aforementioned emersion. Skateboards escape its own play, changing how any space is viewed within its bodily constraints. Once you know how to skateboard, any surface can be seen as a surface to do tricks on. In this sense, other skategames to consider are parkour, climbing, slacklining. So, here's the final axis:

3. the exploitation/exploration of the environment with the trick object (i.e.: can you kickflip down these cool stairs?)

the insight in this case is that the more the spaces are designed towards the skateness of the game, the less interesting the game is itself as a skategame. Again THPS vs Skate vs Skate(glitched) are interesting games to think about level design wise. How much undesigned can a level be? (bennett foddy's piece on zk map for stranger is interesting in this regard) How much can a game affect how you look at the real world, how much can it affect other games you play. "if only I could spawn Mario 64's control into dark souls)

To end this, I'll just list things that I think are interesting skategames, and then spit thoughts that I didn't know where else to put.

Interesting skate games:

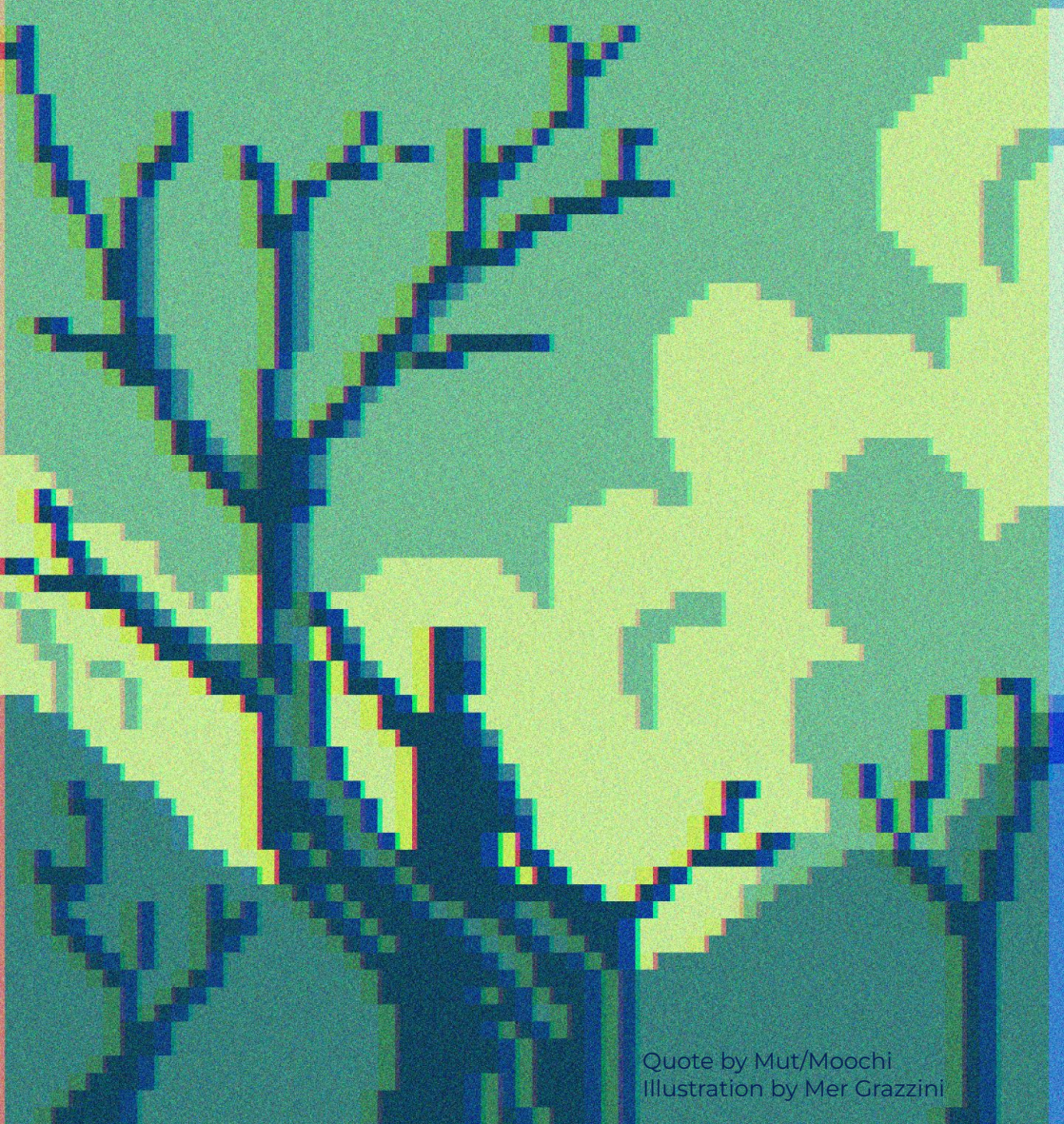
- cs surfing
- titanfall 2
- super mario 64
- call of duty (in particular, no scoping)
- super smash bros melee (combo videos) (movement videos)
- fr0g clan official server 24/7 zk map (for stranger)
- A Kayak for Kyle
- playing snakelike on an arcade machine with 2 trackballs
- getting over it with bennett foddy
- STREET UNI X
- bruxa
- Halo (eg.: tower to tower challenge)
- sebil engineering (maybe a stretch, but there are some expressive movement stuff there)

If you want to see an updated version of this list visit
mut.media/skategames

Stray questions

- how to not acknowledge tricks but still incentivize them?
- how to make it possible to have a game of S.K.A.T.E. without having it be built in?
- is this different than dancing?
- can skategames be competitive? are they just adversarial?
- "bodily" expressive toy with no intrinsic evaluation system
- is falling required in a skate game?
- whats the difference between a slip, a mistake and an error?

Designing towards pushing players to experience the game as reality.
To push them towards the outside of the magic circle.



Quote by Mut/Moochi
Illustration by Mer Grazzini

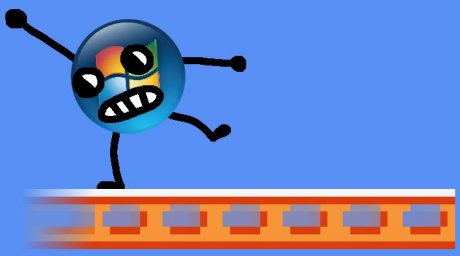
Super Indie Dev Simulator (2021)

Words and drawings by Mer Grazzini

A practical guide for old and new players of the game industry. Short tips with everything you need to know to ~~survive~~ thrive while making games.

Aim for:

- A stable salary
- Croissants
- Seeing the sun (sometimes)
- Fame



Avoid at all costs:

- Windows Vista
- Coding moving platforms
- AI
- Fame

Upgrades:

- A computer that doesn't freeze
- Leds
- A comfortable chair
- 8 hours of sleep expansion pack
- More leds
- Cool toys for customising your desk
- A cat (or dog)

*Note to people who couldn't download the patch "Fixing eyesight", UI might look blurry, you can buy the Glasses upgrade item to fix it.

Interactions:

- How you move
  - Arrow keys
  - Legs (some players might have the wheelchair mod installed)
  - Crawling out of bed

- What you can click
  - Plushies (softy)
  - Drums (noisy)
  - Your mouse (dah)
  - People you like (and like you back)

- What you can not click
  - Your lost dignity (‘cause it’s lost)
  - People you like (and despise you)
  - A rotten orange (I mean you can but, ugh)
  - The fleeting memory of a sunrise with your uni friends, half drunk, and the sound of your laughter, smiling at jokes lost forever in time.
  - Also vapour.

Achievements:

Baby steps: Assist your first jam!

Selfcare: True Happiness (only 2% of players have this achievement)

Pro moves: Mentor a jam.

Nosferatu: Spend the whole night coding.

True dev: Release a game (common)

Greater than its parts: Build a successful team.

Truer dev: Release a game you're proud of (rare)

Egomaniac: Split from your team in a time record (for speedrunners)

The true achievement: Not dying (yet)

Darker than black: Drink 6 cups of coffee in a day.

Free 1,000,000$ game ideas

If you decide to make one of these a reality you will immediately become rich and famous and everyone will love you!

"non uniform gravity skate-like game"

Snake rpg, every fruit pickup is a stat check

When a game renders it turns gamestate variables into an image.

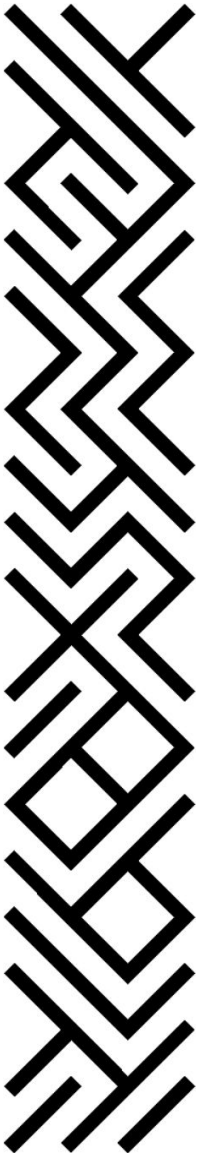
How much can we abstract this step?

Describe gamestate in as few variables as possible and render them using a shader / music / landscape / stories

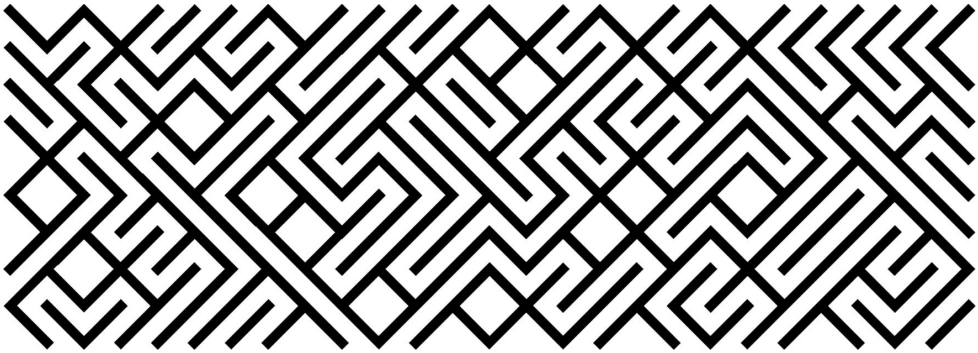
Some games try to be movies, what if a game tried to be a music video instead?

Short thoughts for each day of the month

Curated by Ash

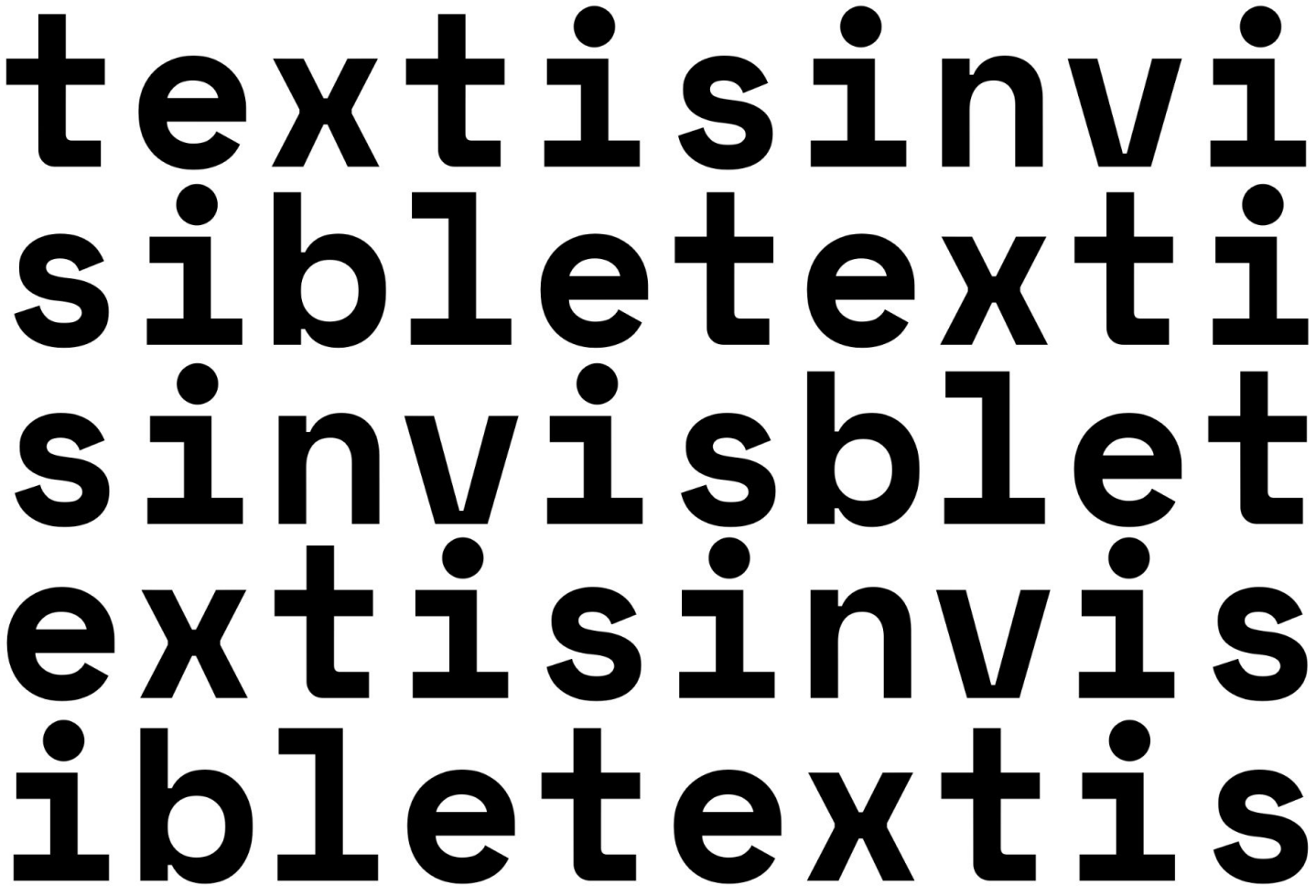


1. Most computer games that call themselves RPGs are LIARS
2. Do not say *the big number*, do not think about *the big number*
3. There are no good games
4. There are interesting emotions and experiences that are not positive in the moment, do not be afraid to make unfun / unsatisfying things
5. Everything can be analysed through the lens of either being a door or a key
6. Делай добро и бросай его в воду (do good and throw it into the water)
7. You need to know what peoples' preferred adjectives are
8. All game mechanics exist on a spectrum where one extreme is about cleaning a mess, and the other extreme is about making a mess
9. You can make an average place feel safe and cozy by making everything outside it scary
10. If you don't look smart enough - make diagrams. If you have nothing that needs a diagram - make shit up

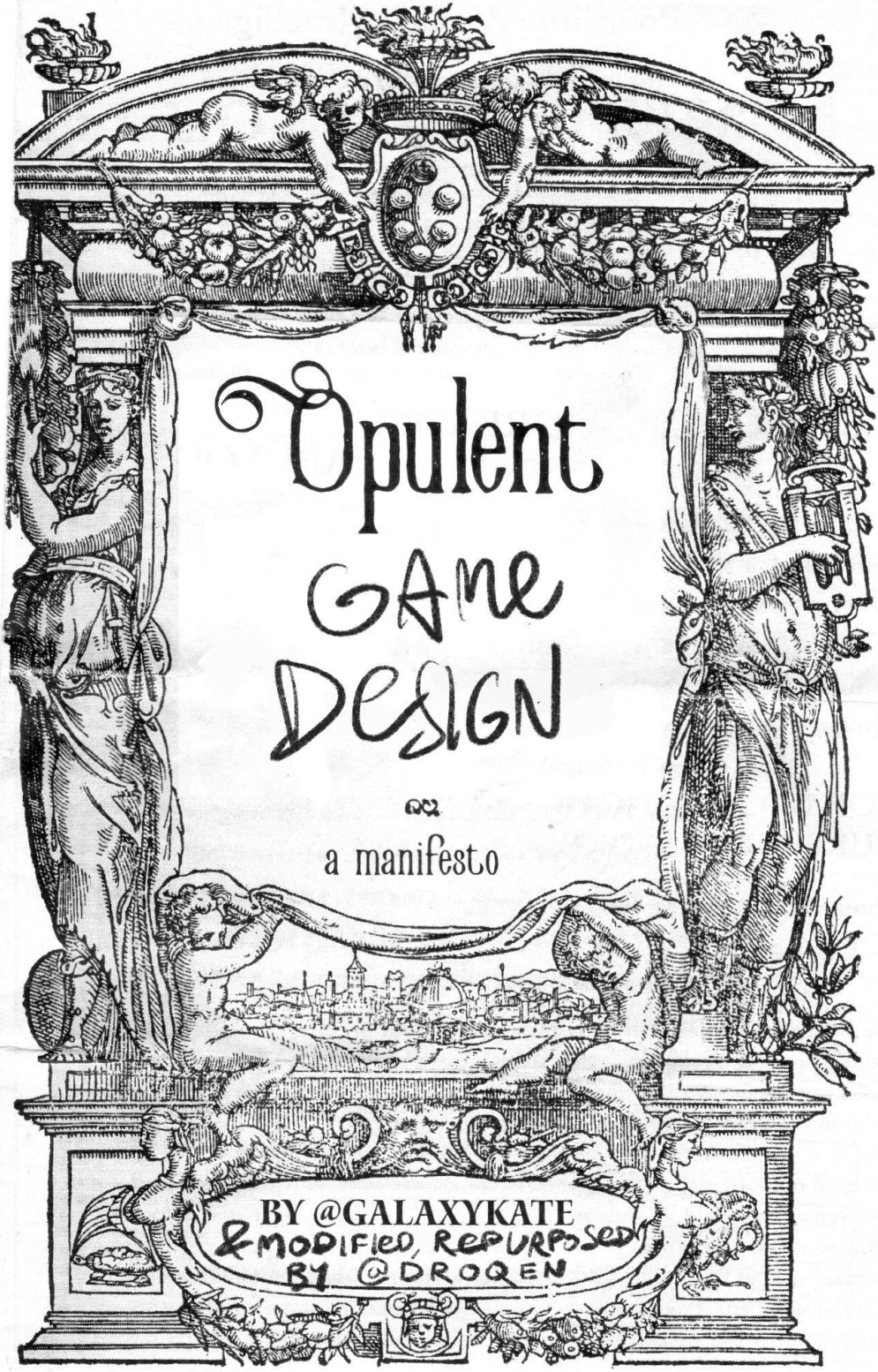


11. All art is theft, there is no art that did not steal anything (stealing is ok)
12. People will always extract a signal from noise, even if there was nothing to extract and they were looking at pure noise.
13. All games contain a skeleton, a red key, a score thing and a magic door
14. Some ideas are perceived as off limits because it is impossible to turn them into a good game. If you stop trying to make a good game you can start engaging with those ideas.
15. Sisyphus is happy because he is playing a roguelike
16. If you see a prescriptionist or an essentialist - RUN. If you ARE one - it is already too late.
17. Mark Brown is not god
18. The mitochondria is the powerhouse of the game design
19. Give players permission to leave
20. All good architecture leaks. But what about games? “All good games glitch” - moochi

21. Apply the Mushroom Mindset to Your Games!
22. Discussions about terminology can get pretty convoluted, eat up all the space because everyone can contribute, and end up creating little new actionable items because it's not like anyone will stop calling things roguelikes
23. Be cringe, but be free.
24. Multiply your estimates by **π** (or 2, suggestions vary)
25. “I can’t understand why people are frightened of new ideas. I am frightened of the old ones” - John Cage
26. There are only three numbers that matter:
zero, one and infinity
27. If a player was sufficiently warned about the consequences of their actions in advance - they should be able to completely fuck themselves over. Just completely and irreparably ruin their experience.
28. Should completionism be encouraged? How can we fight against this formed habit?
29. Engineering is not interesting to anyone except the engineer (and making it interesting takes effort)
30. You can always make something more simple and elegant, but at some point it stops being recognizable and becomes something else entirely


textisinvi
sibletexti
sinvisblet
extisinvis
ibletextis

Quote by droqen
Illustration by Mer Grazzini


Opulent
Game
Design
or
a manifesto
BY @GALAXYKATE
& MODIFIED, REPURPOSED
BY @DROQEN

Poor little GAME DESIGNER!

She scurries around behind the scenes of games.

"Find me the fastest workflow!
"Solve this problem for me, designer
"Route the player to the end of the level without running into pain points!
"But do so UNOBTRUSIVELY. Do so UNSEEN. Never, never, make yourself a spectacle. They do not wish to see you or talk to you. You are just here to do work."

"....Oh, and can you do it faster and with less resources?"

HOW DREADFUL lets make her FABULOUS!

I want to coin the term "Opulent Game Design" for design that takes up **all the resources** it wants, takes **all the attention**, and makes the experience **all about itself.**

FOR NO PRACTICAL PURPOSE WHATSOEVER

Chooseing flattering looks for your

Opulent Game Design

The experience is designed to focus and highlight the entire team.
"Look at how they work! Isn't that so interesting and fabulous"
"Look at how many different kinds of things they can make, and how different they are!"

The goal of opulent design is to dazzle the user with its extravagence, to flood their senses with its variety and charm. Practically, you may want to create collages of the generated content, or generate additional contentaccentuating the main output or adding context (fanciful descriptions, simulated reviews, glitter effects, attractive frames) Create additional characters and features just to point at the design process and exclaim how delightful and clever it is!

Visualization: Lay it all bare

What can the user learn about how the game works? Lay its glorious operations bare before them so that they may marvel at its construction!

Approaches:
Show visualizations of what everything's state currently is, the more shiny and graphical the better!
Let us all say what we're thinking. Why did we make those decisions? Let us expound upon our sagacity and wisdom.
Add hocks that let you track and output the logic that got it to those decisions
Open it up! Leave the backdoor to our brain open, leave the rules and datafiles in plaintext. Allow the suers to admire them, edit them, and reshare them.

I received @GALAXYKATE's "Opulent AI" zine at LOST LEVELS and it took my breath away — but as the years went by I came to realize I was jealous of the Opulent AI. As a designer and an artist, I wanted to be opulent myself. You can be, too.

Cut-and-play exercise

Cut out these strips. Choose two, and design a game using the technique on the back to do the thing on the front

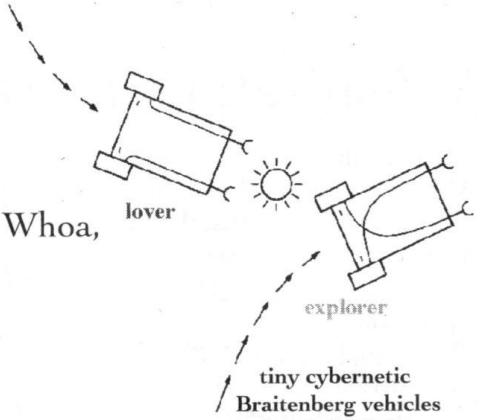
weaving a scarf	choosing beads for a necklace
dealing with loneliness	writing a poem
dancing with you	playing with a puppy
designing a stylish sock	communing with the dead
comforting a distraught friend	creating great artworks

What are games even?

GAMES are a form of ART. Whoa, humanity alert!

GAME DESIGNERS are just ARTISTS.

An artist is something that:
Looks at its world (or hears or listens to input somehow), Thinks (or has some tiny logic), and Acts on the world, by turning on a light or saying something, or performing an action. And that's it!

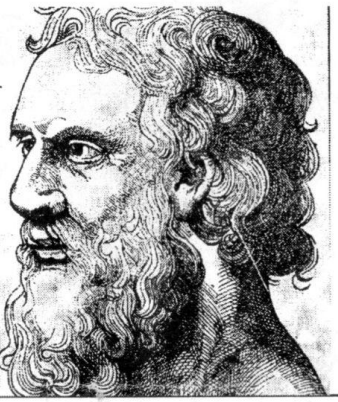


tiny cybernetic Braitenberg vehicles they "see" light, and "think through a single wire"

so I guess they're artists too!

SOCRATES: Or again, in a ship, if a man having the power to do what he likes, has no intelligence or skill in navigation [αρετης κυβερνητικης, aretes kybernetikes], do you see what will happen to him and to his fellow-sailors?

Plato, Alcibiades I*; Benjamin Jowett, translator*



"Game design saves the souls, bodies, and material possessions from the gravest dangers."

—Socrates according to Plato, c. 400 B.C.E.

Neural Networks	Evolutionary algorithms
Finite State Machine	Tracery grammar
Twine	Markov Chain
Edge detection	Pencil-and-Paper ruleset
A-star pathfinding	Cellular Automata

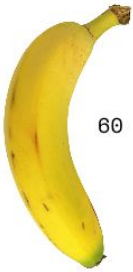
Anthropometric units for game developers

By Ash K

For far too long have game developers been stuck with precise mathematical units of measurement, dictated by technically minded types, or resulting from the quirks of game engines. People have no intuition for how long a frame is, or god forbid any of the many, MANY abstract values encountered at every step. And no matter how much you try to constrain them with a neat mathematically pure metric, people will often use universally understood imprecise measurements anyway: bananas or coins for scale, handfuls for volume, etc.

I propose a solution, a number of novel units of measurement that are genre and tool agnostic, that are in the order of magnitude of most values being used and that people either have an intuition for or are based on things all people do.

8.5 : 1



60 : 1

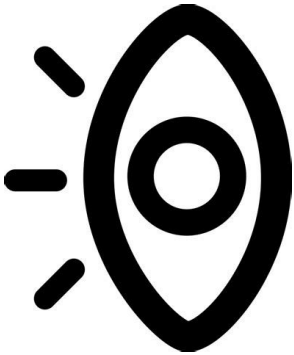
10.2 : 1

Time

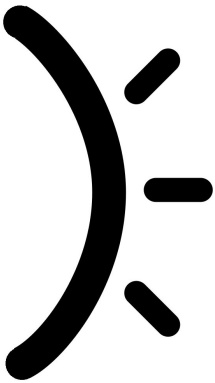
Most small scale time values in games are measured either in seconds, milliseconds or frames. Milliseconds are good for processor time but bad for visual output as players will only see a new image once every 16ms at 60fps. Frames are a whole other disaster as most good games use fps independent physics and animations, and because it is often ambiguous as to what framerate a game is running at when this is mentioned.

✦Blink✦

Introducing THE BLINK. 1 BLINK is equal to 1/12th of a second or 5 frames at 60fps (most common fps values have neat relationships with 60).



Everyone intuitively knows how long a blink is, and it is hovering around the lower bound of what is useful when portraying something visually.



✦Reaction✦

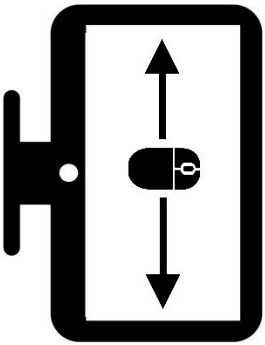
Another useful unit of time is THE REACTION. 1 REACTION is equal to 3-4 blinks, 1/4 - 1/3 of a second or 15-20 frames at 60fps.

While reaction time varies between different people it is still a very useful thing to keep in mind. This value might seem too big at first glance so I urge you to test yourself here. *

*(https://humanbenchmark.com/tests/reactiontime)

Input

Console players have standardised controllers, it is easy to define how fast something should move based on the joystick value. With computer games there is a problem - people have different mice with different sensitivities, each used to their hardware.

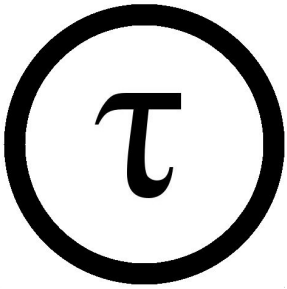


✦Mouse desktop✦

A useful unit of mouse movement could be THE MOUSE DESKTOP. 1 MOUSE DESKTOP is defined as the value change of the mouse' position that is equal to moving it from one side of the desktop to another (it is technically 2 separate units, one for vertical movement, and the other for horizontal).

Everyone is used to moving their mouse on the desktop, the distance from one side to another makes intuitive sense to a player and is what they are comfortable moving within for long periods of time. Asking a player to move the mouse more than a desktop without giving them an opportunity to lift their hand is generally considered to be a dick move.

Direction



✦TAU✦

Everyone knows what a circle is, how much is a quarter of it, a third, etc. Just use TAU (equal to 2*PI) when dealing with radians so that you can intuitively understand what kind of angle you are talking about. (TAU / 4 is a quarter of a circle in radians or 90 degrees)

How to make your own anthropometric units:

1. Ground a unit in some physical aspect of your player / their setup / your environment / world
2. Choose a scale that is appropriate
   (i feel most comfortable when values are in a 0.1 - 10 range, if you find yourself using numbers like 0.0005 or 50000 - the scale is out of whack)
3. Give it a cool name
4. Use it

Example:

1. I am making a platformer where the world is made out of tiles.
2. I ground my unit on a property of the world to make values like “3 tiles length” or “5 tiles per second” easy to grasp compared to something like “48 units”.
3. Values using tile units now end in “_tiles” e.g. speed_max_tiles
4. Values are converted to game engine units by multiplying values in tiles by a const called tile_size

The Deathfeel

by lintilion

It's How It Feels, to Get Slaughtered, Simply Put.

Death is inevitable in 90% of the gamer's exploits, and it's often not even FUNNY.

That's the thing -> Gamer Death can, and should, make you feel things other than "Shit, death!"

It's hard to do that. Death sucks! Loss sucks! Death and Loss come hand in hand, but there are ways around it.

Ways to make death seem like a secret, or a setback, or a Fine Thing.

ANY TIPS?!

- Make the character scream in a funny way.

- Ragdoll physics

- Make dying the true path

- Do Not Return The Player to The Last Save You Mother Fu

- Fast reloads

- Mario Scream

- I liked the way Hylics handled death, I thought that was nea

- Die, Understand finally

- Do not allow the player to Truly Live

- 4K Resolution

- High-Impact Destruction Sequence

- 3 Lives which can be spent on goods and services

- The "3 Tile Rule"

-Some kind of advanced mechanism??

Thank You.

Written and Art
by
Homie "Lintilion" Boon

Doorfesto

Words by Stuffed Wombat
Visuals by Mer Grazzini

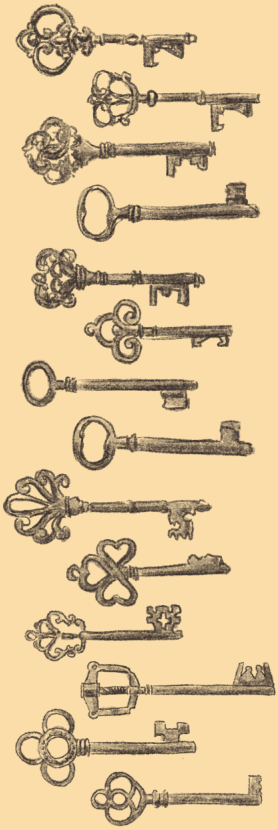
Videogames are made of doors.

Doors are invitations. Stepping through a door and entering a new place, experiencing how it differs from the place that you just left behind is interesting and worthwhile in itself.

A locked door is a promise. It tells you that if you spend more time inside of the game, you will eventually find a key and be allowed to experience a new place.

Locked doors are so effective in keeping people engaged that game designers have been embarrassed about them for a long time, dressing them up in different costumes:

~ A key is transformed into a parrot and a locked door magcially turns into a parrot-less pirates.
~ Dealing with depression becomes a locked door and catching 20 fish becomes the key.
~ An enemy (blocking a door) becomes a locked door and your giant (key-shaped) sword has been the key all along.



To understand how a game works on a large scale, you only have to understand which of its parts are keys, which are doors and how they relate to each other.

Once everything in the game been reduced and abstracted into either door or key, the structure, the pacing, the overall composition of the game becomes clearly visible.

Looking through a single, unified lense, helps you to see a game in its whole-ness, even if that hole-ness is key shaped!

Doors are invitations.

A locked door is a promise.

Corrupted Doorfesto

Words by Droqen over Wombat's first essay
Visuals by Mer Grazzini

Videogames are made of doors.

- Stepping through a doorway and entering a new place, experiencing how it differs from the one that you just left, is interesting and worthwhile in itself.
- Touching a door and imagining a new place and how it differs from the one that you are in now, is interesting and worthwhile in itself.
- Looking at or even thinking about a door is interesting and worthwhile in itself.
- Jiggling the doorknob of a locked door is interesting and worthwhile in itself.

Open doors are invitations to enjoy the pleasure of stepping through a doorway and entering a new place, but they deny you the pleasures of jiggling the doorknob, and of imagining what is on the other side.

Closed doors are tasks.
In order to experience the basic pleasure of stepping into a new space, you must first experience the basic pleasure of opening the door.
Experiencing the new place and opening the door are not the same thing, but they are experiential equals, related partners.

A locked door is a closed door that cannot be opened the obvious way.
It might be taken as an invitation, as a task, as a challenge, or as the promise of a key... in fact a locked door is all of these things. Like an open door, it is an invitation to enjoy the pleasure of stepping through a doorway and entering a new place, but it requires you to enjoy other pleasures first.

Like a closed door, it is an invitation to enjoy the pleasure of opening the door, but it requires you to enjoy other pleasures first.
Uniquely, it offers the pleasure of jiggling the doorknob to discover it is locked, and then to feel the sense of mystery that arises. Unlike a closed door, you cannot simply open a locked door, but most of the time something will communicate that some force is capable of unlocking it.

Sometimes a locked door can be opened if you have the right knowledge. Nobody can say whether this type of door is truly locked, or simply closed.
Let us dive deeper into the **doorfesto corruption**.

The original **Doorfesto v 2** says that 'as stupid as it is, all of game design can be expressed through this simplification..'

- Looking at the sunset is an open door and there is no key.
- Reaching the end of a level opens the door to the next level.
- A wide gap is a locked door, the double-jump-upgrade is the key

If you look at these first two examples closely, however, you can discover that a Closed door is simply two Open doors: the act of opening a closed door is, itself, a "new place".

- "Reaching the end of a level" is an open door and there is no key. It leads to an area containing the also-keyless door to the next level.
- "Killing the skeletons" is an open door and there is no key. It leads to an area containing an open door to "gaining experience points".

If the way to reach the end of the level, or the way to kill the skeletons, is uncertain, then it may instead be a locked door.

A closed door is the *quantum superposition* of an open door and a locked door. You cannot know which one it is until you jiggle the doorknob.

Corrupted Doorfesto's Final Formfesto

As an art form, games have long been defined by locked doors: obstacles which frustrate progress because they might require a key, but you don't know what form the key takes. Sometimes locked doors are invisible.

There has been a lot of discourse around what a game is. Sometimes a game doesn't contain any locked doors, but by wearing the mantle of 'game' it suggests that it *might* contain locked doors. Games are now defined by their *potential* for locked doors. Sometimes what appears to be a locked door is in fact merely closed in a funny way. In rebellion, a game may remove all the locks from their doors, or all the doors from their frames. These things cannot cease to be games, but games are still defined by the existence of locked doors in their neighbours.

To understand how a game works, you need to understand more than its keys and its doors – you must understand the *possibility* of what *appears to be* a locked door, and not just the keys that will unlock it but also all the possible keys that will fail to unlock it.

This will then allow you to view everything through a many-dimensional but unified lens, helping you to see a game in its whole-ness, even if that hole-ness is shaped like a keyring with an infinite number of keys on it!

You gotta learn to enjoy what you do even if there's only three people in the audience, because it matters to them, and mostly, it should matter to you.

Quote and watercolor by Mer Grazzini, lineart by droqen.

The exquisite corpse: what is juice?

We asked five renowned scholars from around the world to submit cutting edge analysis about the concept of juice.

They couldn’t really agree on everything, but we thought you might still be interested in reading a raw version of their debate.

Juice

This part written by ***Pearson A. Deice***

I want to change the way i think about juice in order to approach it from new directions. I am not sure what I would write to accomplish that so I will do this instead:

1. make assumptions
2. come to conclusions based on those assumptions
3. tweak my assumptions until the resulting conclusions make sense

for your reading pleasure each iteration has the changes highlighted

Assumptions, iteration 1

These are my starting assumptions:

1. juice is perceived
2. juice is not mechanics
3. juice is not player behaviour
4. juice affects experience of player
5. juice can be added and subtracted without impacting mechanics

CAN content be juice??? story?? levels? context, graphics and audio???? all juice???
BULLSHIT

juice is a part of all of them, HOWEVER it is the part not thought about consciously all the time while still affecting gameplay. ***(Dante: or simply ...while still affecting perception of what is happening in the game. is that the same as gameplay?)***

therefore:
juice mainly affects the subconscious

Assumptions, iteration 2

point #5 implies juice can affect gameplay. seem aight.
Seems good at first glance. Need to throw shit at it until it breaks.

DO TABLETOPS HAVE JUICE?
YES, in their physical properties. Things that clink, donk and rustle, things that feel nice to the touch. SLAMMING a card on the table and shouting UNO!!! causing your homies to bleed out their ears is undeniably juicy. Same goes for a DM doing funny voices for the goblin shopkeeper.
(Dante: a great example of how juice is many times unnecessary and uncomfortable)

Therefore:
juice can exist in non computer games

Assumptions, iteration 3

#1 & #2 suck, replacing mechanics with rules

Interesting question to #1. Who is perceiving the juice. Probably player, but what if its someone else???
* designers perceiving juice?
* only part of players perceiving the juice while playing an otherwise identical game?
* stream viewer seeing something a player can not??

Productive ambiguity spotted, will hopefully promote interesting discussion and fucked up games?
Therefore:
juice is perceived
juice is not game rules
juice can be added and subtracted without impacting game rules
***the person doing the perceiving does not have to be the person playing the game**

The person playing the game can often perceive more than people just watching the game be played. If you're just watching someone play a game you don't "Feel" the character moving when you press left on your controller.

Assumptions, iteration 4

TF is perceiving anyway??

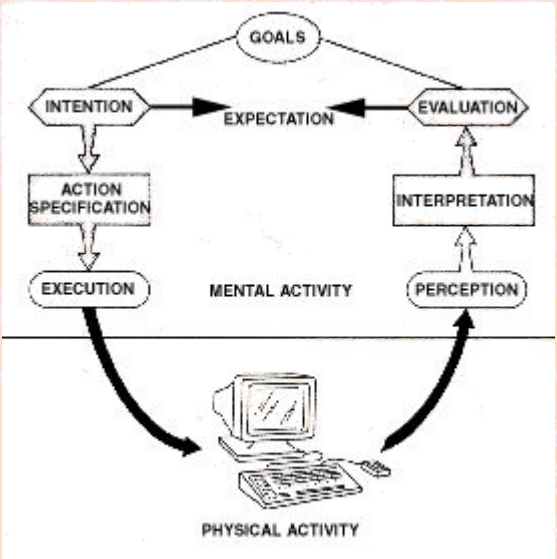
gulf of evaluation & execution (crusty jpeg from the interwebs)

had a large amount of thinking came to conclusions:

juice exist only to those who perceive it

1. ***it is possible for a person to not perceive juice even when they are physically able to.***

2. ***juice can be perceived by people not playing the game directly.***
juice is interpreted and evaluated subconsciously



Assumptions, iteration 5

Does the effect of juice on player need to be positive? Most modern juice makes you feel good. Player might be punished with unpleasant juice that feels worse than if it was not there.
Some people enjoy traditionally unpleasant experiences in moderation, maybe this is actually dope?
(e.g. horror games)

Assumptions, iteration 6

A game can always have more juice, but any implementation of the rules always has a minimal amount of juice intrinsic to the implementation.

Pure rules have no juice, like when a game is played entirely within a players head. ***(Dante: what about this game: you click the mouse and it goes phrhehhwiowwwwshshhh)****(Pearson A. D. : that is a good example of pure juice, no rules)*

Conclusion:

juice can be added without impacting game rules

implementations of rules into something that is playable will inevitably contain some juice

it is possible to have wildly different experiences from different implementations of identical game rules

Assumptions, iteration 7

Returning to juicy assets, any part of any assets not examined directly at the moment is juice. Art is juicy when you see it in the corner of your eye and instantly understand what is going on in that part of the screen, music is juicy when you are not thinking about it. All assets perform a double duty - juice when on the sidelines, content when on the centre stage.

Assumptions, iteration 8

1. juice exist only to those who perceive it
    a. it is possible for a person to not perceive juice even when they are physically able to.
    b. juice can be perceived by people not playing the game directly.
2. juice is not game rules
    a. juice can be added without impacting game rules
    b. implementations of rules into something that is playable will inevitably contain some juice
    c. it is possible to have wildly different experiences from different implementations of identical game rules
3. juice can exist in non computer games
4. juice affects experience of person perceiving it
    a. juice does not have to be conventionally pleasant or evoke exclusively positive emotions
5. juice is interpreted and evaluated subconsciously
    a. Things are only juice when they are evaluated subconsciously
    b. Everything can be juice, some things act as juice significantly more often than others

I'm at a point where i can't find more issues with these. They are NOT PERFECT, I'm just not smart enough to find the rest. Document will be passed to other experts in the field of juiceology in the hopes that they will be able to tear this list to shreds and reassemble it into something better.

Feel

This part written by: Gottfrid

Juice is dead.

Long live feel.

- jan svankmajer

The described juice is about perception, it's about physicality. When we touch an object we feel it's texture, we hear the sound of it brushing against against the skin of our hand. It's weight defines how holding it will strain your muscles. This is feel.

Reality has feel.

Feel creates assumptions. If something is heavy, I'm assuming that if I throw it will fall and make a definitive *thomp* when dropping on the ground. If the texture is smooth, it will slide down a slippery lane of water.

Our bodies are assumption machines. But bytes have no weight, no roughness that our hands feel. The keyboard keys are always the same, the controller triggers produce the same resistance. This cannot be changed through software. We use juice to simulate feel. ***(Dante: We use juice to simulate feel)***

Assumptions, ~~iteration 9~~ feelified

1. ~~juice~~ **feel** exist only to those who perceive it
    a. it is possible for a person to not perceive ~~juice~~ **feel** even when they are physically able to.
    b. ~~juice~~ **feel** can be perceived by people not playing the game directly.
2. ~~juice~~ **feel** is not game rules
    a. ~~juice~~ **feel** can be added without impacting game rules
    b. implementations of rules into something that is playable will inevitably contain some ~~juice~~ **feel**
    c. it is possible to have wildly different experiences from different implementations of identical game rules
3. ~~juice~~ **feel** can exist in non computer games
4. ~~juice~~ **feel** affects experience of person perceiving it
    a. ~~juice~~ **feel** does not have to be conventionally pleasant or evoke exclusively positive emotions
5. ~~juice~~ **feel** is interpreted and evaluated subconsciously
    a. Things are only ~~juice~~ **feel** when they are evaluated subconsciously
    b. Everything can be ~~juice~~ **feel**, some things act as ~~juice~~ **feel** significantly more often than others

The assumptions are now exhibiting some issues. For feel, is not just juice. Feel, the simulated physicality, is affected by the game rules. If wood would sink, instead of float, it would feel different. Our assumptions would be different. In games, it is the rules that we govern over the (non) digital actors that create assumptions in the player.

(Dante: making the thing that floats present as wood is juice. if it is something else, it affects feel, and new juice is needed to fulfill the assumptions created)

Assumptions iteration 10

0. feel exists in digital and analogue space
1. feel exist on a subjective spectrum of perception
    a. it is possible for a person to not perceive feel even when they are physically able to.
    b. feel can be perceived by people not playing the game directly.
2. feel is a combination of many factors
    a. implementations of rules into something that is playable will inevitably contain a feel
    b. it is possible to have wildly different experiences from different implementations of identical game rules
    c. juice inherently creates feel
3. feel affects experience of person perceiving it
    a. feel does not have to be conventionally pleasant or evoke exclusively positive emotions
4. feel is interpreted and evaluated subconsciously
    a. feel is evaluated based on our assumptions
    b. feel creates assumptions
5. Everything can be **feel**

1.b is partially wrong. Not all feel can be "perceived by people not playing the game directly."

Why should we feel?

But why is feel so important? The goal of juice is to accommodate, create feel. Without feel, juice is nothing. But by creating juice, we create feel. Yet, feel is created by more than just juice.

Feel without juice is still feel. Juice, without feel, still becomes feel.

Feel is the queen and juice is a pawn reaching the far side of the board awaiting its promotion.

xXx_Uhm, actually_xXx

by bobo

my colleagues both raise good points
but they are not as lost in semantics
as i am
so I will quickly fix that.

The first confusion is about the term
JUICE.

JUICE just means means
"cheap FEEDBACK".

Just like UI Design,
UX Design and UD Design
is part of what we call FEEDBACK.

FEEDBACK informs the player
about the current gamestate.

Concurrently, point 2.2 of both Lists from before is wrong.

Point 2.2 states that "implementations of rules into something that is playable will inevitably contain some juice"

But

FEEDBACK, which includes JUICE, is added on top of rule, it is a skin, a costume, a mirage, an illusion.

FEEDBACK informs the player about the current gamestate, it does not generate that gamestate.

So point 2.2 has to be dismissed angrily and with many exclamation points.

!!!!!!!!!!!!

(Dante: You forgot about it being PLAYABLE. A playable game has a closed feedback loop with the player, and so has some design decisions made, and so has some juice.)

JUICE was discovered by Joy Stikdottir →
the mythological mother of independent game development in the nordic regions of europe.

Since then, JUICE has spread all over r/gamedev and is therefore hugely influential.*

*citation needed but not desired

JUICE is very popular and everyone wants to have it!

But JUICE is just FEEDBACK!

JUICE is cheaper to implement than most types of FEEDBACK because it deals with the properties of already existing objects.

This cheapness, this connection to the objects of a game often leads to FEEDBACK being confused with FEEL.

Sadly, this happened to my colleague as well and now I will complain about that.

FEEDBACK informs players about the gamestate.

The gamestate is created by rules.

These rules can have some inherent FEEL to them,

The way a character moves through the screen is not driven by FEEDBACK, it is driven by RULES.

The velocity of a projectile is not FEEDBACK, it is the thing that FEEDBACK needs to inform you about.

Rules can not have FEEDBACK.

Bur rules can have FEEL.

To change the FEEL of a game means to change its rules.

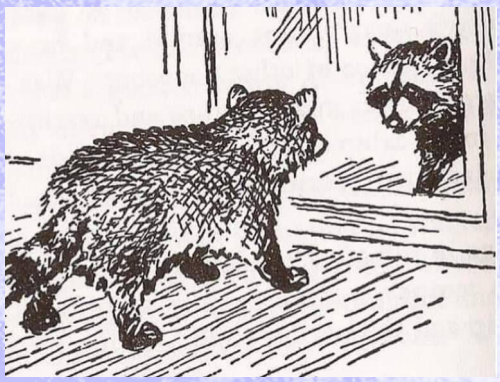
To change the FEEDBACK of a game means to change the FEEDBACK.

There is another debate to be had about all of this, but im done here.

Peace.

Rules don't exist, only juice

by Rurururururraru R. Ru



Since some of my colleagues mentioned prevideogame juice (or feel) lets observe the ultimate juice machine, dice. (This will obviously become a semantic war, but I will try treating most of what the previous colleague said as true, even tho it might not be.)

Dice, allegedly invented by Wan D Sixious many many years ago, as every game designer of respect knows, is the first ever videogame. Dice was originally made with the bones of animals, and the dots only were invented many years later by people finding the original dice, now stained. We know that the configuration of the first die with dots was: 1, 1, 3, 4, 9 and one empty side (which later influenced in the invention of the number zero).

Now lets imagine yourself throwing that dice.

Do you feel it? The juice? [1]

Hmmph, the dice fell out of the table, oh no! [2] It was a 9! [3]

Lets throw it again, but move it more in your hand before throwing it. [4]. Do it!

It fell in the empty side. [5] Lucky for us!

So, how did that feel? If you didn't feel anything, you might not be on with the latest imaginatioin juice and you should seek an especialist (luckily you can just email me, the specialist at rurururururur@rur.ru).

It has multiple moments, which I marked with numbers in square brackets [n], maybe I will explain those in a future section, but for now we gotta think about what were the rules of this interaction.

On a first approach, using mostly intuition and gut feeling, one could think that there are no rules in this interaction whatsoever (This text was originally intended as being a rebutal of my colleagues idea of rules).

"Dice are not a game!", one would say, "dice have no rules." But my colleague, I understand, would counter this, maybe by telling you to do another thought experiment, this time one of implementing a dice virtually (you could try repeating the previous thought experiment even by doing the same thing but in something like tabletop simulator, I just did it and it was pretty cool).

When thinking about this, you start thinking about the rules of dice: it has to have 6 sides, for example. It moves through the screen. it has some (projectile) velocity, and change in speed even. It collides with stuff.

And then you have what my colleague calls feedback, which gets complicated in this dice-only game. For example, is the color of the dice part of the rules or part of the feedback, or the fact that is makes sound when hitting different structures. They seem to be rules, but with inherente feedback. Rules can have inherent feedback, not only inherent feel.

(editors note: this is probably the only objective insight of this essay).

One interesting thing about dice, is that its an object that has so many games made about it that part of its imaginary properties also create rules that are impossible to not think about. It seems to be inherently better to roll a higher number than it is a lower number. There are classic meta rules created around dice, like falling out of the table. These all create different experiences and feelings. Another good thing about this being a thought experiment is that all you had was just feelings, so maybe that's feel?

You can maybe see now what I was doing with those number, just figure that out I guess, for homework (you can send it to the email previously mentioned).

To end this, lets look at the last iteration, and for the sake of it change 'feel' and 'juice' to dice

0. dice exists in digital and analogue space

1. dice exist on a subjective spectrum of perception

    1. it is possible for a person to not perceive dice even when they are physically able to.

    2. dice can be perceived by people not playing the game directly.

2. dice is a combination of many factors

    1. implementations of rules into something that is playable will inevitably contain a dice

    2. it is possible to have wildly different experiences from different implementations of identical game rules

    3. dice inherently creates dice

3. dice affects experience of person perceiving it

    1. dice does not have to be conventionally pleasant or evoke exclusively positive emotions

5. dice is interpreted and evaluated subconsciously

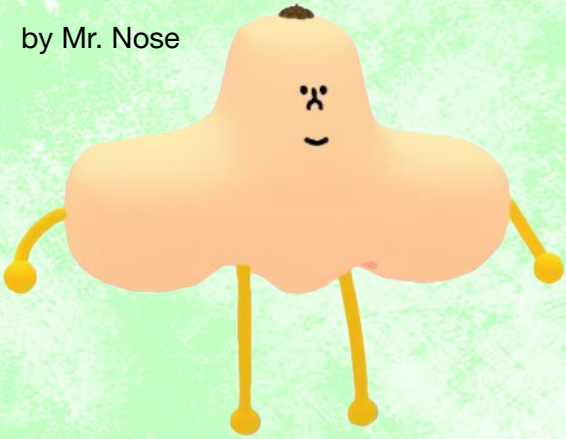
    1. dice is evaluated based on our assumptions

    2. dice creates assumptions

6. Everything can be **dice**

JUICE IS COHERENT FEEL

by Mr. Nose



The thing is, juice is feel is feedback is rules. Juice is when you have fine and detailed control of a thing, when the thing provides enough (often subconscious) cues that you can wield it simply, expertly, intuitively. When small simple actions trigger an avalanche of COHERENT feedback, applications of rules, and opportunities for further control of the system. When the game FEELS a certain way - good or bad (I know, I know, those words don't actually exist in the English language, but I must use them because I don't want to think up actual words.) - because you have been able to so completely plug your body into the control / feedback loop. You become the dice.

JUICE is a way of establishing harmony. You can do it cheaply. you can do it expensively. You can do it wrongly. (another game's juice is not transplantable to every other game). But most importantly, you can do it rightly.

Once again: it is not added to rules.
It is not just feedback.
It is rules.
It is feedback.
It is control.
It is dice.
Juice is perceived dynamics.
Juice is controlled dynamics.
Juice composes a certain feel.

Rather than expressing my love for a thing *at* people, I find myself looking for things to bridge gaps between us.

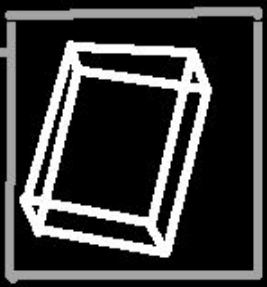
Quote and lineart by droqen, watercolor by Mer Grazzini

By Stickyfr0g

Menu-less Inventory

As the medium of games progresses there is a feedback loop between designers and players. Designers create solutions for large and often abstract problems and players get used to these solutions over time, which then makes these solutions more likely to appear in future games. But just because players are used to any given interaction model doesn't necessarily mean its the best solution to that problem. Yet, once most people are familiar with a system that is good enough, it can be difficult for new games to solve the same problem in a different way without annoying all the people who are content with the way most other games did it. A long time ago I heard someone say:

"If you are going to replace a standard design solution with your own, it had better be worth it."



I think this strikes a good balance between wanting to do new things, while wanting to bring as many players with us as we go in this new direction. And it's from this perspective that I would like to talk about the possibility of creating menu-less inventories.

First of all, it’s not that menu-based inventories are bad. Like most modern game design solutions, they’re good enough, but there are some downsides. One thing that feels odd about most menu-based inventories is how detached they are from the rest of the game. You’re running around having fun and then you need to take out an item so you open up a spreadsheet and scroll around until you find the cell you’re looking for. It doesn’t often match the feel of the other interactions in the game. Of course this isn’t always the case, most menu-based inventories are fine the way they are, but menu-less inventories create all sorts of problems that could lead to new and interesting gameplay.

There are many different ways this problem could be approached. Here’s just a one idea:

If the player has direct control over their hands, then they can interact with any of the pockets, backpacks and containers on their body or nearby.This could allow the player to develop a muscle memory for retrieving things from common places as well as naturally modelling the time it takes to retrieve an item based on how far it is from the hand and how many layers of containers it's inside of. Another thing this could allow is greater depth of inventory management. Items could be organized spatially the way we do in reality, which leaves it up to the player where they store things, how much time they want to spend organizing their items versus just shoving them all inside a pocket or dropping them on the a table

In reality containers are often much less discrete than they are in video games. If your pockets are full but you need to remove another thing from your hands, you try to cram it in. Backpacks are really fun to think about in this interaction model because in reality backpacks are so dynamic. You don't just have pockets; you have side pouches, straps with buckles, bungee cord nets, velcro, you can shove a skateboard between your backpack and your back. Of course, modelling all these different ways to carry items would in ways be approaching a kind of "Simulation Dream" problem and it would be a ton of work. But by removing menu-based inventories, the containers no longer need to be discrete and compartmentalized in the rigid ways that menus are. This is where menu-less inventories get really exciting.

There is a lot more to say about this idea, but I will leave it to just that one example for now. I hope some people find this concept interesting because I think there is a ton of potential to be explored. If you experiment with anything in this area I would love to see what you come up with.

Please send me a message on Twitter!

Thanks for reading.

-stickyfr0g

twitter.com/stickyfr0gg

MONSTER LABORATORY

By Mer Grazzini:

Ziggy (@arrogantgamer) has a 4-year old son (Mu) who's learning to make pixelart alongside his father.

In Paradise, we decided to do a bunch of fanarts of Mu's small monsters.

You're invited to do your own; Just pick one from:

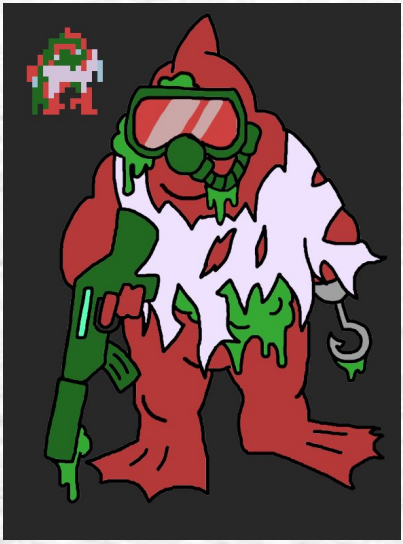
varioustpixels.tumblr.com/

By Kya:

By Q1:

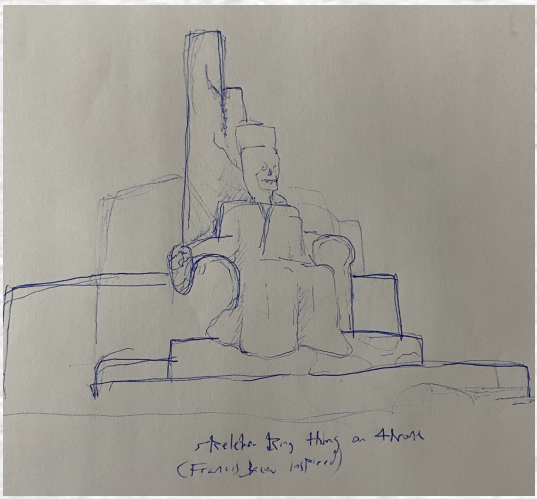
By Kultisti:

By Corey Hardt:

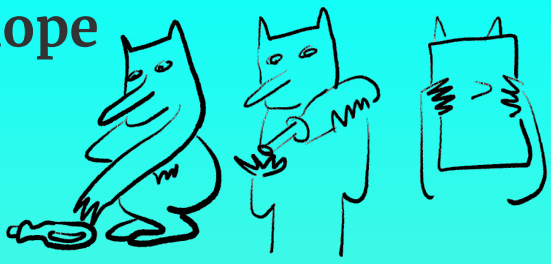


By Pancelor:

By SirMilkman:



One of several reasons analog games give me hope



By Marek

A while back I had a yen to boot up some of my old games and see what kind of changes I've been through as a game developer. I made some old games in klik and play but they no longer run in newer versions of windows - something about being 16 bit executables no longer being supported. I made a bunch of games in Flash in college, but now that's gone too. Thanks to the heroic efforts of the fine folks at the Internet Archive, those can run emulated in the browser, but what will happen when those, too, stop working? It takes constant effort to keep these compatibility layers operating. Will the successes of digital archivists continue forever?

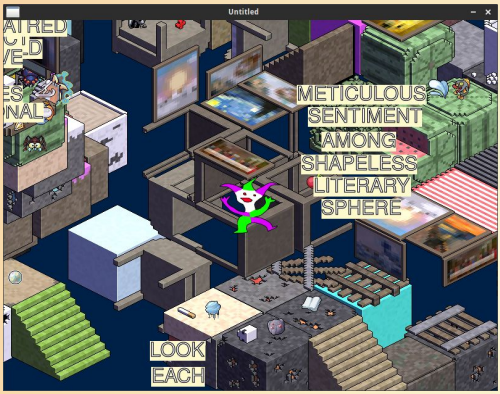
Video games are built on a desert of shifting sand dunes. APIs change, old versions of shared libraries cease to be hosted, services are deprecated. PCs need constant updates to prevent them from becoming the target of hackers, and alongside those updates comes the changing behaviors that cause software to misbehave.

When a painting ages, it does so gradually- layers of grime accrete, resins yellow, the sun fades pigments - but it happens slowly, in a way that we can watch happen as we age. It happens predictably, so we can learn how it happens and make choices when we're painting that will prolong the life of a painting, or restore it. It happens inexorably, so we can accept it's temporal existence.

When a video game ages, it does so all at once. One changed API, or missing library, or new security setting will stop the whole thing in its tracks until someone with very specialized knowledge goes in and fixes it. An unfathomably vast complex of industry create the conditions that allow a video game to execute - from rare metal miners to manufacturers of CPUs, GPUs, PSUs, RAM modules, monitors and keyboards and cases, fabs and dies, to programmers of operating systems and game frameworks, their business and marketing and HR departments. Each entity in this sprawing network has the potential, through pettiness both personal and political, through market incentives, competence or incompetence, with intentions both good and evil, to create a rippling chain reaction that flows through this mysterious medium and renames a single registry value that your game depends on in order to run. There is no solemn dignity for art that is macerated in the grinding gears of progress. Your art will not stand as a testament to hubris, eroded by eons of lonely winds. It will be silently garotted by an update that fixes touch gestures in a sticker app.

The same force that through the green fuse drives the flower does not drive tech's eternal puberty, but you are just as dumb to tell this to Google's support page as you are to the hanging man.

Do your games run on Apple silicon? It's a different instruction set, you know - don't take that for granted. There's some kind of compatibility shim for x86 apps that's already set to be deprecated, and it's up to you- yes, you, the person reading this- to re-build your apps to run on the new macbooks. Unity is certainly not going to do it for you. Your value to Unity (or Unreal, or Game Maker, or whatever) is exactly equal to how much you pay them every year.



Any concessions they seem to make to your workflow are the lurchings of a giant whose limbs stretch beyond the horizon. If you died today, would your grandchildren be able to experience the art you made? Do you have someone to cut a new build before your chosen version of Unity falls out of Long Term Support?

When journalists from the real world come visit our beloved slime bog and interview the rare developer that can stand tall enough to poke their head out of the muck, they (both parties) like to make excuses for games' stinkiness by calling it a "Young Medium". Two thoughts on that: In the British Museum there's a copy of the game of Ur from ~4600 years ago. They won't let you play with that one but with a bit of elbow grease you can whip up your own version and play the same game as the Mesopotamians. Secondly, everyone gets one lifetime to work on their art. I just want to be able to put my games in a box and take them out when I'm older and make some sense of where my head's been at, but I don't want to have to keep debugging build errors on all these dang things for the rest of my life.

Feeding the [game design] creature

by: itsitluk
visuals: metmuseum

The [game design] creature makes games
all day and all night.
But ya also gotta feed it,
yeah, that's right!

Outside! Now! Go and feel the wind!
Peripatetic school walked like all over the place while thinking,
over here, over there, probably over somewhere else too
it can do wonders!

Or take a shower, close your eyes and just breathe for an hour.
That rhymed!
Sometimes the subconscious has the answer
but your mind is just too busy to let it blossom.

Read! Like a lot. Watch forgotten movies, foreign animations!
Listen to music, not the genre you always have rolling 24/7,
but something completely different!

If our creativity grows from new associations between our past
experiences met with the current moment and thoughtflow,
we can feed it! Feed it all the art you can!
[see short thoughts for each day of the month #11]

Play games! (**plural**)
Not the one you have *1000h+* hours in, i mean, it's a really cool game,
but there are like hundreds of other cool games waiting for someone to
play them. That someone could be you.

Seeing the different ways others interpret the word "game"
can help you discover the definition of your own!

Sleep! Sleep! Oh, i can't stress this enough. *Sleep!*
Did I already say sleep? It's important.

During the night you say farewells to *the rational critic* and let *the subconscious art freak* take over, paint yourself flying over the endless skies. Skies born out of your mind.
Many times you'll also discover answers to design problems from there!

Notebook!
This is like the most important thing anyone can ever have ever completely absolutely totally nothing beats this no no no
you gotta have one. And a pen too!

Write it down. What? The idea you had! But it sucked!!!
Write it down! I don't wanna– WRITE IT DOWN, NOW!
You will regret it in the morning if you don't write,
'cause then it'll be long forgotten, gone, perhaps forever.

Empty blank walls, are those doors?
Waiting for your drawings and pictures to open them?
[see doorfesto]

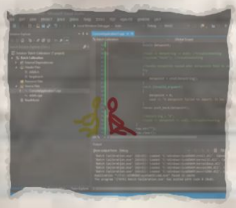
Decorate your walls with stuff that inspires you!
Your very own personal irl moodboard, yeah!
Staring into blankness can get your thoughtwork rolling
but what if that blankness had a colour and shape to it too?

And remember to feed yourself as well!
It is you who is here and who is making games
It's good to take care of that you who is here and-
Making food can also be a game!!!
[see Skate Games]

Immediacy In Game Developement
Or!!
The Length of A Moment

Something I often complain about:
Video games take too long to make, are too difficult to make. This can be a practical concern but what bothers me is the impact it has on self expression. When I write a sentence, it hardly takes longer to create than it does to think, I can instantly export the idea or feeling I had. Things like photography, music, doodling, are also close to that end of the spectrum for me. And though you *can* spend ages tinkering and toiling over these mediums, usually to make longer works, they have a lower **Immediacy** floor, where it's possible to capture quick impressions and thoughts.

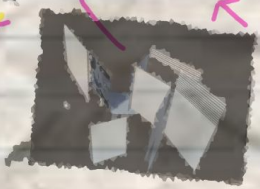
I like doing all of those things too, but I like video games the most! But making a video game, even a very short one, takes considerably longer, and often feels more removed from the experience you get out of it at the end. When I write a song on guitar, I get to hear it while I play it and the build time is the speed of sound. But to make a video game, I usually have to plan things in documents, stare at Microsoft Visual Studio for days, go to a seperate program to make assets in and then drag and drop things in an extremely visually cluttered editor. I feel, or used to feel, like this often results in losing, at least a little, **The Moment** that existed at the start of the idea. This loss of immediacy only gets compounded when working with others.

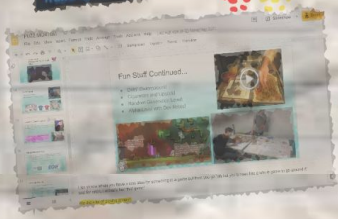


So I've been trying for a while, in certain games, to try to simplify and scrunch down my process to be more immediate and sketch-like. And I'm not alone in trying this! Many great tools have been made to help simplify game development, make it quicker and easier and more minimal. These tools have the added benefit of being more accessible to new (or exhausted old) developers, and things like Bitsy and Puzzlescript have allowed me to make some games I'm really happy with in only a couple hours. And other people have made some really beautiful tiny games in them too that convey to me at least the feeling of immediacy, though I guess I don't know how long they took to make.

Continued!! on page 2!

Recently I was able to push this concept farther than I ever have before. One day while working from home, I started listening to "I heard the sound of violence, far off" by Owen Gilbride. I looked out the window while listening and completely zoned out for a while, watching the snow fall. It was a very transcendent, maybe dissociative **Moment**. I got the urge to try to convey or capture the feeling I had in a game, and with hardly a conscious thought I opened Unity and began to make a game. I decided part way through that I would finish the game when I finished playing the album. It's a fascinating game to look back on for me through this framework of **Immediacy**, because *it's the fastest I've ever made a game*, and when I play it and listen along to the album, I am brought back very intensely to that **Moment** watching the snow fall.

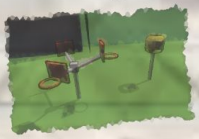




So I made progress on the **Immediacy** front maybe, and I still advocate for playing with that kind of game making and want to try to push it farther for myself in the future. But I also had a thought recently about how longform game making may actually also fit into this lens of **Immediacy** and capturing **Moments**, and it's helping build a better picture of what I want games and game development to be. I was thinking back, as I often do, on the development of Trail Mix. It is *the longest I have ever taken to make a game.* But when I think about it I'm reminded of lots of **Moments** and feelings during its development. It feels like every random thing that happened to us, and every game that we talked about while making it, found their way in somehow.

So rather than capturing a specific **Moment**, at the inception of an idea, maybe a project developed over longer simply captures **a longer moment in time**. It's like a long exposure photograph of the time taken to make the game, and it gets blurrier, but it's all there. This helps me reconcile with the inevitable longness and clumsiness of making bigger games, the notion that all the time and effort spent on them gets embedded and crystalized in the game over the course of its development. I get a similar feeling playing Breath of The Wild shrines, for example, and getting a very strong feeling that different designers with different schools of thought and ideas about level design worked on different groups of shrines.

Having (perceived) contact and communication to individual developers through the fog of a massive AAA game gives me hope and makes me smile. I like games that let you peer through into the person/people who made them. Through recontextualizing this notion of **The Moment** of creation as being something that can be stretched and shared, I feel more able to appreciate both making and playing longer and larger games, though I'm still not done with my quest of trying to improve **Immediacy** in my own work. Maybe this will be a useful notion to you as well, who knows! Thanks for reading.



@-JohnLee_Erg

This zine was brought to you by the many members of Paradise!

Paradise is a small-ish artist collective that gathers many devs from all over the world, united by our love for small games and art in general.

At some point in the past half year we were talking and accidentally stumbled onto the topic of making a half-serious half-jokey zine of game design advice. The idea ended up being well received and a bunch of people have contributed their essays to this zine since! We even had to extend zine size a couple times to ensure there was enough space for everyone.

There will probably be another issue of Paradise Zine, since we're a unstoppable force of art creation, and we looooove discussing stuff and writing essays.

We can't be sure how this zine got to your hands/hard drive, but if you want to know more about us, you can check:

https://twitter.com/paradise_collab

https://paradise-collab.itch.io/

https://youcanneverleave.art/

Be happy. bye 👋


PARADISE
EXIT NOW