How to Lose a Dimension: Can the unique advantages of designing games in 2D be utilised to create adaptations which capture the essence of the originals?

Intro

"Losing a Dimension is a project based around the idea that 2D video games and 3D video games can be thought of as two different mediums within games design. Through the creation of 2D adaptations of 3D games, I will be investigating and analysing each medium in an effort to gain a deeper understanding of both, while also creating a detailed resource for other game designers."

That was an excerpt from the poster I submitted as part of my MA project. For this project, I created a series of adaptation documents in which I analysed the design of three video games, before detailing the process of designing a 2D adaptation of them and discussing the thoughts and intentions behind my decisions. The three games I chose to adapt were Resident Evil, Subnautica, and The Elder Scrolls V: Skyrim. I also created working proof of concept demos in Unreal Engine to show how the adaptations would function if they were real games.

Screenshots from Resident Evil 2D (top left), Subnautica 2D (top right), and Skyrim 2D (bottom)

The whole adaptation process was interesting and challenging, encouraging both thoughtful analysis and creative thinking – it also ended up being a lot of fun. For these reasons, I believe that the process is a valuable exercise for anyone with an interest in games design, whether they want to research it, learn about it, or just do it. In order to help out anyone interested in creating their own 2D adaptations, I decided to create this guide, in the same format as my adaptation documents themselves.

How could it work?

This document isn't going to be a step-by-step guide as the process is fairly fluid, without a set path; instead, it's going to be more of a discussion of what you might need to change when creating your adaptation, using my own adaptations as examples. The aspects of game design I'll talk about are

the gameplay and mechanics, the level/world design, the visuals and art style, and the sound. I'll also talk a little about how you should also try creating a proof of concept demo if you can, as it's a great way of seeing if your ideas work in practice.

One more thing before getting into the discussion points: in my adaptations I talk about capturing the "essence" of the original games. To put it simply, I wanted my adaptations to still be instantly recognizable and evoke similar feelings as the originals – I wanted to stay faithful. This was important to me as the point of the project was to see whether we can successfully adapt a game into 2D; staying faithful to the originals while making changes that make use of the new medium's unique advantages is part of the challenge and can also be used to judge the success of the adaptation. Capturing the essence of the original games isn't always straightforward, as usually quite a lot about a game needs to be changed in order to successfully adapt it into 2D. This document will go over the methods I used to capture the essence of my chosen games, and hopefully they'll help you capture the essence of yours.

With all that being said, let's go!

The Gameplay

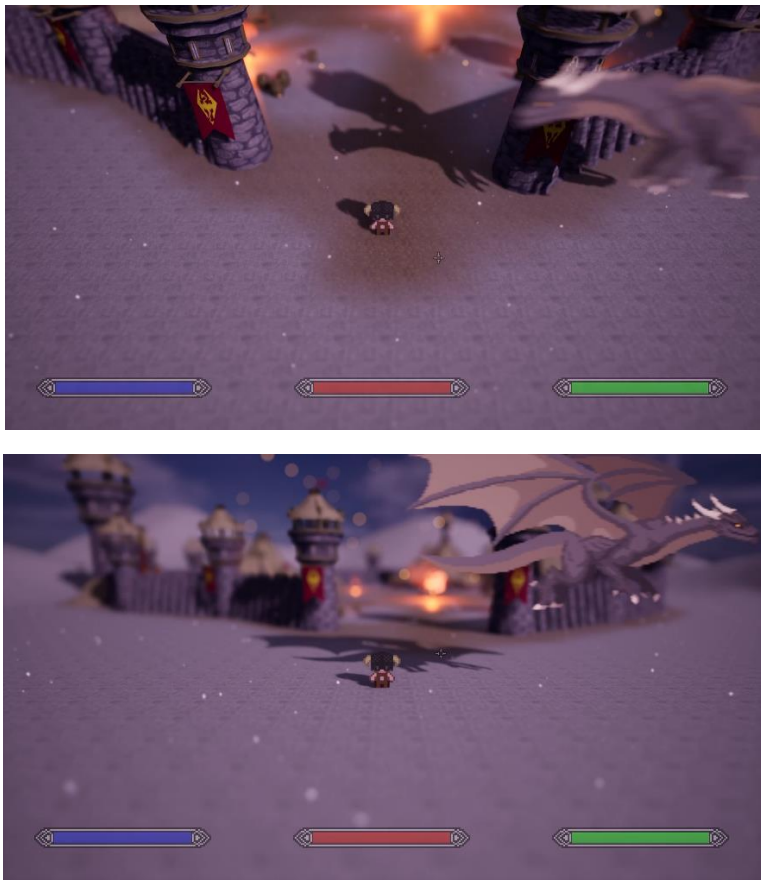
When it comes to gameplay, making a few changes to avatar movement and the controls is essential. This might sound like not much but think about it – avatar movement affects how the player moves through and perceives the playable space, and the controls affect how the player moves their avatar, as well as how they interact with all the elements that are contained within the playable space, and all the game's mechanics. Specific aspects of the gameplay, such as puzzles and even combat for example, can all be thought of as variations of movements and interactions, whether your game is played in three dimensions or two.

In 3D games, the player character usually has free roam of the playable world and can move in all three axes – up and down, left and right, and forwards and backwards. In Subnautica, for example, you can orient the player character in any direction you desire, which is important due to how exploration is a huge part of the game. Contrast this to Subnautica 2D – the player character's movement is locked to only being able to move in two axes, these being up and down, and left and right.

Choosing which axes to remove has a large knock-on effect on the rest of the game, as it affects the layout of your levels and world, what the player can see in the world, as well as how the player moves through the world. If you have main sources of inspiration, this will likely inform which axis to remove, which then affects the games perspective. For Resident Evil 2D, I was largely inspired by retro point and click horror games which led me to make it a sidescroller. For Skyrim 2D, a large source of inspiration was top-down dungeon crawlers such as A Link to the Past. Sometimes the gameplay you want can inform the decision – for Subnautica 2D, I needed the player to be able to dive downwards into the deep ocean, and a sidescroller allowed for this kind of vertical movement.

This choice also affects some of the ways your adaptation will be restricted, which can be tricky to work around. Choosing top-down for Skyrim 2D allowed for the gameplay I wanted, but it also meant the player couldn't see the horizon – meaning I couldn't gaze at the mountains in the distance, or get a good view of dragons, both of which are important parts of the original! This led me to introduce a "sidescroller mode" where the perspective shifts to side-on, and your movement is locked to a different 2D plane as to not get too close to 3D territory. This is a good thing – restrictions like this encourage interesting solutions! Think about what gameplay is crucial to the

original and try to remove an axis with that in mind, but don't be afraid to get creative when problems arise – and trust me, they will!



Top-down mode (top) and sidescroller mode (bottom) in Skyrim 2D

Usually, controls will also have to be changed. Most 3D games have controls for movement, interacting with the world of the game, and for moving the camera. When it comes to movement and interacting with the world, the majority of games are fairly similar – you move the player character with WASD, an analog stick, or a d-pad, and use the face buttons, triggers, bumpers, keys, or mouse buttons to interact with the world (or you use a touch screen for everything like on mobile). What 2D games most often lack, however, is a moveable camera – at least with the same free movement as an FPS, for example. The camera even affects the use of the buttons used to interact with the world, as a lot of the time you'll have to rotate the camera to face whatever it is you want to interact with.

That's not always the case, though. In some third-person 3D games, interacting with certain objects is partially based on proximity – you'll get a context sensitive prompt when you're in range of an interactive element like an NPC, sometimes without needing to rotate the camera. This kind of proximity-based interaction can be a reliable choice for 2D games, especially due to the locked gameplay axes. Additionally, a mouse can still be utilised for 2D PC games to control a cursor used to interact with the world – Stardew Valley is a good example of this. Consoles can also have controllable cursors, but these are a little controversial due to how to make up for a lack of precision, they will often be made slow-moving. This is what led me to consider implementing a pointer (rotated with the unused analog stick) for a hypothetical console version of Skyrim 2D, to allow for the precise aiming and shooting of ranged projectiles and spells – this could all be done with a cursor on the PC version. Depending on what you're adapting, that kind of solution might

work for you as well, but you might have to come up with something entirely new depending on the unique qualities of your chosen game.

It's worth remembering though that most 3D games have a fully controllable camera so the player can look at everything that's in the 3D environment. Sure, it affects how you interact with objects in the game, but that's only because you can approach these objects from any direction imaginable, and the objects themselves can be placed in any location imaginable. The restricted world space and camera of 2D games means you're able to save time when it comes to building the world as you don't have to account for a player being able to view it from any possible angle, but you do have to account for the world being more immediately visible, both of which are important factors when it comes to…

The Level/World Design

Level and or world design is where quite a few changes will most likely need to be made, usually dictated by your choice in perspective. The way I'll approach this section of the guide is as if you've taken a level from your chosen game, and you're trying to come up with a 2D version – this is the approach I took in my adaptations.

Generally, you don't want to be aiming to produce a one-to-one recreation of your chosen level; in fact, in most cases this won't really be possible anyway. It may sound obvious, but levels in 3D games are designed with three dimensions in mind, and the best levels make good use of all three axes. Remove one of the axes from the level, and what are you left with?

Well, it depends on the game you're adapting, and how you've chosen to adapt it. For Resident Evil 2D, I redesigned a good chunk of the Spencer Mansion to work for a sidescroller, meaning that from where you begin in the level, rooms exist either side of you and above you, but not into the foreground and background – the building only has one layer of depth. This meant that a lot of the rooms had to be reshuffled and placed into the one layer of depth I had to work with, resulting in a mansion with a lot of width and a lot of height. This isn't really how mansions are built in the real world, but all was not lost; one advantage of 2D games is that due to how the perspective we have of the game world is so different to how we perceive our own world, there is inherent distance from reality. This lets us get away with more unrealistic or "gamey" features, which, in a game more focussed on realism and immersion through said realism, can undermine the player's suspension of disbelief. We don't have to worry about that. Furthermore, there's the context to consider. The Spencer Mansion is a creepy, spooky place full of unnatural things – an impossible mansion fits with the unnatural, otherworldly theme. Depending on your chosen game, this kind of thing might work for you as well.

Then again, you might just get lucky. Top-down floor plans for buildings, spaces, and indeed levels in games are very common. In Skyrim, the in-game maps of its dungeons are displayed from a top-down perspective. As it happens, Skyrim 2D is a top-down game – how fortunate! Okay, fortunate, but still not necessarily something that made designing a level for Skyrim 2D a walk in the park. I redesigned the Bleak Falls Barrow dungeon and was able to use the in-game map, as well as a much nicer fan-made map, as references, which ended up being extremely useful. Even so, there were aspects that needed changing for various reasons. I ended up making the layout more winding to make up for how having to orient the camera in the original can contribute to getting lost. I added some alternate routes to make the dungeon feel less linear, as the grid-like layout emphasized the linearity. I even had to rotate some rooms, so important points of interest and interactivity were on the most easily visible northern walls. My version of Bleak Falls Barrow was closer to my original

than my version of the Spencer Mansion, but I still made changes that made sense for the shift in perspective.

The lack of a freely controllable camera makes a difference when it comes to level design as well. One advantage is that the player is at your mercy when it comes to what they can and cannot see in the level – you have a much higher level of control over what can be seen by them at any given time. However, it also means you can't make use of landmarks in the distance to catch the player's eye and guide them through the world – at least, not without some out of the box thinking. I planned on wrapping Subnautica 2D's horizontal plane around a cylinder, so the player could see islands and the crashed Aurora spaceship in the distance, and still be able to reach them despite the movement axes being locked. This kind of idea might not have popped up without the limits of the adaptation!

Though removing a dimension can complicate things when dealing with adapting established levels, having one less dimension to worry about can also make level design simpler in general as there are just less variables to consider when it comes to the way the player experiences and moves through the environment. This means that while at first it may stump you, as it did with me, you should find that iteration is quicker and simpler due to the physical restrictions removing a dimension brings to a game's world.

If you're really not getting anywhere, looking to other games for inspiration can make all the difference. If you've decided on your choice of perspective, you can probably think of games that are at least somewhat similar to what your adaptation will be like. Why not have a look at how their levels work? Looking into dungeons in A Link to the Past was vital when deciding what changes to make to Bleak Falls Barrow. My version of the Spencer Mansion was partially inspired by Boo Houses in New Super Mario Bros. - inspiration can come from anywhere!

The Visuals

The visuals – by which I mean graphics and art style – are where you can afford to go wild to a certain extent. There are loads of routes you can go down when it comes to the visuals in a 2D game. You could use flat 2D sprites, 3D models, or even both at the same time – they all have unique advantages and disadvantages.

Using 3D assets in a 2D game can add some depth to what can appear to be quite flat without good use of techniques such as parallax scrolling. However, flat, sprite-based styles aren't restricted by the limits of polygons, which allows for all kinds of unique styles such as pixel art, or even gorgeous hand-painted graphics, all of which can be super smooth without having to worry about the aforementioned polygons, with their quantity limits and rendering expenses that can affect how a game will run.

This all goes hand in-hand with the distance from reality I mentioned earlier – unrealistic art styles are perfectly suited to 2D games as they don't have to worry about realism and the dreaded uncanny valley. Using 3D assets for the environment though – and even 2D sprites in a 3D environment – can allow for the use of dynamic lighting which can add a lot of drama to a scene.

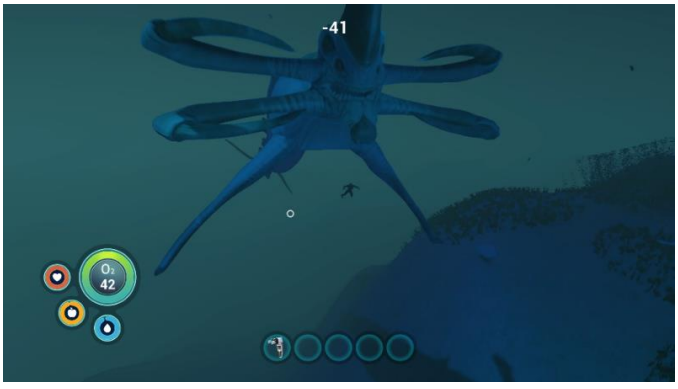
With that being said, it's important to consider which parts of your game you're trying to preserve. For Resident Evil 2D, I wanted to keep the game creepy, as the campy horror elements are what makes Resident Evil… Resident Evil. In the original, the 3D characters, creatures, and interactive elements, are rendered in real time on top of pre-rendered backgrounds. This helps important elements stand out, whilst also feeling unsettling due to how the zombies look a little wrong as they creep around the very still environment. This led me to the style I chose for Resident Evil 2D, in which I used essentially the opposite – 2D characters in a 3D environment. I even went as far as

dressing my fiancée in the character's outfits and taking photos to use as stop-motion animation frames, like in the old Mortal Kombat games. This helped me achieve a similar slightly goofy, yet uncanny feeling as is present in the original while also being able to use highly detailed sprites for characters and creatures!

Screenshot showing my chosen art style for Resident Evil 2D

It's incredibly useful to think about what effects the visuals have on the experience of playing the original game. If you deem these effects to be essential to the experience – like I did with Resident Evil – you'll need to consider how these effects are achieved, and whether or not you can achieve them in a way that makes more sense in a 2D game. Being intentional is important – I could easily have arbitrarily copied the visuals of the original because they work well enough, but using flat 2D sprites with photos of real people made the game look unique while still achieving that essential uncanniness!

One last thing – there's the UI to think about as well. For my adaptations, this mainly meant making changes to the HUD, as I didn't implement every single menu in the game. I took three different approaches in my adaptations. For Resident Evil 2D, I changed it so the inventory was moved to the HUD so it was always on screen, as this is something that many point and click games do, so it's very evocative of the style. It also meant the player was constantly able to see their heart monitor, and their ammo count ticking down, which added extra tension, helping the stress level get closer to that of original. For Subnautica 2D, I kept everything the same as the original, as the HUD displays all the necessary information in a way that both doesn't break immersion and increases excitement and tension due to the presence of the depth and oxygen meters. However, I did remove the compass as the player's view of the world doesn't rotate. Finally, for Skyrim 2D, I kept the style of the HUD the same, but made pixel art versions of the elements to make it more in line with the player and NPC sprites. I also removed the compass, for the same reason as with Subnautica 2D.



Screenshot from Subnautica 2D showing the HUD

Like with everything else, consider what effect the UI and HUD have in the originals, decide whether it's essential to the experience, and make changes based on that. You might not have to make many changes at all, but it's worth considering some, as moving the inventory to the HUD in Resident Evil 2D made a big difference. Like with Skyrim 2D, you might need to alter the HUD elements to match your chosen style even if you aren't changing anything else about it – a retro 8-bit style adaptation might need the HUD elements changing to match.

The Sound

In my Subnautica 2D adaptation document, I wrote a good paragraph about SFX and music in 2D games. I used it as an excerpt in my Skyrim 2D adaptation document, and I'm going to use it here as well:

"Many 2D games take a more bombastic approach to sound – think of the expressive sound effects and catchy, prominent melodies of classic and modern Super Mario Bros. and Sonic the Hedgehog for instance, which go hand in hand with the playful vibes of the game. Additionally, these games aren't focussed on realism (which is part of the reason they work so well in 2D) so don't necessarily need to use music and sound to pull you into the experience – it's more to emphasize the fun. However, there are also those which stray from this approach at points to inject some mystery and even creepiness – Super Metroid is a good example."

There are exceptions this – 3D Mario games feature similarly joyful soundtracks to their 2D counterparts – but it's a good place to start when thinking about changes. Resident Evil, Subnautica, and Skyrim all feature soundtracks which I deemed to be important in achieving atmosphere, so I ended up keeping them in my adaptations as it just didn't make sense to change them. I think sound is underrated in terms of how it makes games feel unique – when it comes to the essence of a game, the music in particular can be inextricably linked to the experience. Luckily for me, the games I happened to choose had soundtracks with both ambient and more involved pieces, which work well in both 3D and 2D.

It really comes down to your unique adaptation. If your adaptation has turned a game into a much more arcade-feeling experience, it might not make sense to include ambient music which doesn't line up with fast paced gameplay. However, if you're worried you've strayed a little far from the original, choosing to keep the music can help tie everything together, as it isn't the be all and end all when it comes to whether or not a game is considered 2D. It's worth noting though that if your adaptation is nothing at all reminiscent of the original when the music is removed, you may need to consider some of the choices you've made thus far. Music is a powerful tool, but it shouldn't be the thing doing all the heavy lifting!

Then we get to the interesting conundrum of where the game's sound effects are being heard from. In 3D games, most sound effects come from a location in 3D space, and are heard in relation to where the camera is located. In first-person games in particular, this means the player hears sounds in relation to where the character is located. This was an issue with Subnautica 2D, as sometimes the camera was too far away to hear sounds that play close to the player character. This led to me making it so ambient ocean sounds are heard in relation to where the camera is, and ensuring crucial sounds (such as PDA alerts) are 2D sounds, able to be heard from anywhere. It really depends on where the camera is located, which wouldn't have been an issue if I didn't implement a camera zoom feature.

Finally, the sound effects themselves honestly probably won't need many changes. Though, like how I mentioned with the UI and HUD, if, for example, you're going for a retro 8-bit style, it might make

more sense to use sound effects more akin to what is heard in those games, but these are case-by-case situations really. As I keep saying, it all depends on your unique adaptation!

The Proof of Concept

I mentioned near the start of this document that creating some sort of proof of concept is a good idea, as it allows you to see whether your plans and ideas, which may sound great on paper, actually turn out great in practice. If you're able to, I'd encourage you to try to put something together in a game engine, whether you use Unreal Engine, Unity, or even GameMaker or Construct.

You don't necessarily need to use an engine though. Paper plans are also a useful way to test out ideas, especially when it comes to level design. Ideas only get you so far and putting them into practice allows you to test them out, while also developing your practical skills. Both idea generation and practical skills are necessary when it comes to creating games, and these adaptations are a great way of putting them both to the test!

Conclusion

We've finally reached the end of the adaptation process – or at least, my adaptation process. I talked about the exact aspects of game design that I've discussed here in my own adaptation documents, and I ended up with three projects that I'm proud of and consider to successfully capture the essence of the original games. With any luck, by thinking over and coming up with an adaptation based on the aspects and tips I've mentioned, you should hopefully also have a project which looks pretty decent!

While the adaptation process is aiming to produce 2D adaptations which capture the essence of originals, while also using the unique strengths of the 2D medium, it's also meant to encourage taking a deep dive into a game's design, and creatively solving the problems that arise along the way. The project is all about how restrictions and limitations can bring about creativity. With that being said, arbitrarily enforcing restrictions to the point where it is to the detriment of your project isn't good for anyone.

Sticking to the limits of 2D worked for my adaptations as the tighter limitations encouraged me to come up with out of the box solutions to problems and was also important in the wider context of the rest of my MA, but that was for my specific project. I'd encourage you to stick to some of the limits of 2D – particularly the restrictions when it comes to gameplay and level design as these are what I found necessitate the most changes – but don't arbitrarily limit yourself if it's not working out for your unique adaptation.

I found it useful to think about whether an old-school, purely flat, sprite-based 2D game could accomplish what I was trying to achieve. If they could – which was often the case, as many older games employed tricks such as cleverly shading certain sprites to emulate depth – I was in the clear. But that's just what I found to be useful. Whatever you choose to do, as long as the adaptation process itself encourages you to think creatively about game design, and above all else, design some interesting games, I believe it's a worthwhile exercise – it was for me.

So – can the unique advantages of designing games in 2D be utilised to create adaptations which capture the essence of the originals? Yes they can, and Losing a Dimension shows how.