

Level Design Lessons

...from Professor Video Games, Anna Anthropy, 2009-2019.

These essays were first written and published on my website in 2009. They're mostly focused on platform games. I guess that's what I was doing at the time. They've been cleaned up and revised after a decade that I have spent (in part) teaching game design in a real actual classroom, and they reflect me, you know, growing as a designer and becoming smarter and stuff.

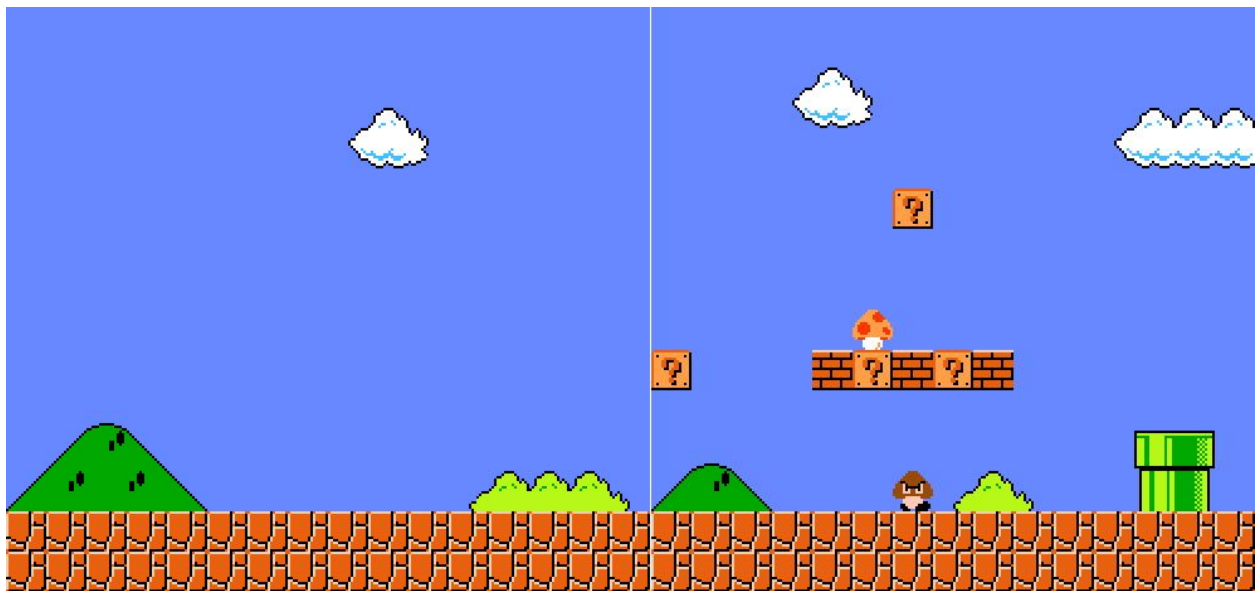
Most of the screenshots here were edited from maps found on vgmaps.com, and were compiled by Jon Leung, Jyouiisan, RyuMaster, Rick Bruns and S&F Prod. Thanks!

Feel free to reproduce this document, assign it to your own students, etc etc. Enjoy!

To the right, hold on tight. *Super Mario Bros. 1-1.*

Games are shaped by rules. Certain rules give the player liberty to change the game state in specific ways: I call these “verbs”– they represent the player’s vocabulary for communicating with the game. If Mario is the main character of *Super Mario Bros.*, “jump” is his primary verb, with the most important button on the controller dedicated to it. The idea of opening a box by jumping into it makes little sense outside of *Super Mario Bros.*, but since the player’s first means of interacting with the game is via the verb JUMP, almost all of the game’s interactions have been designed around jumping. Mario can jump into a box. Mario can jump on a goomba.

The first question of level design is how to communicate these interactions to the player. As a teacher, I often see students rely solely on an instructions screen to explain their game, and not consider how level design can support the player’s learning.



The first two screens of *Super Mario Bros.*, designed by Takashi Tezuka and Shigeru Miyamoto, are designed to communicate the most critical rules of the game to new players. Consider that Mario is the game that popularized the “platformer” as a genre, and most of its players at the time of its release would be encountering these ideas for the first time. That requires thoughtful design!

Let’s start with the very first screen. Open sky, flat ground, Mario. The framing of this screen communicates a lot: Mario appears at the far left of the screen, facing right, and the ground stretches away in that direction. The framing invites the player to move

towards the right. There are no other objects on the screen, nothing to threaten the player as they try out the controls for the first time, learning how to move Mario on screen. When they move far enough to the right, the screen begins to scroll, revealing new things. Go right to advance.

As Mario goes right, two new characters scroll on-screen. There's a flashing ? block, which is interesting, and the game's first threat: A little brown mushroom with angry eyebrows, tromping toward the player of its own will. This is the first time-sensitive event in the game, though this goomba is fairly slow compared to later obstacles. But it's moving from right to left, counter to what we've been taught progress looks like. In order to continue to advance, the player must jump over, or onto, this little guardian.

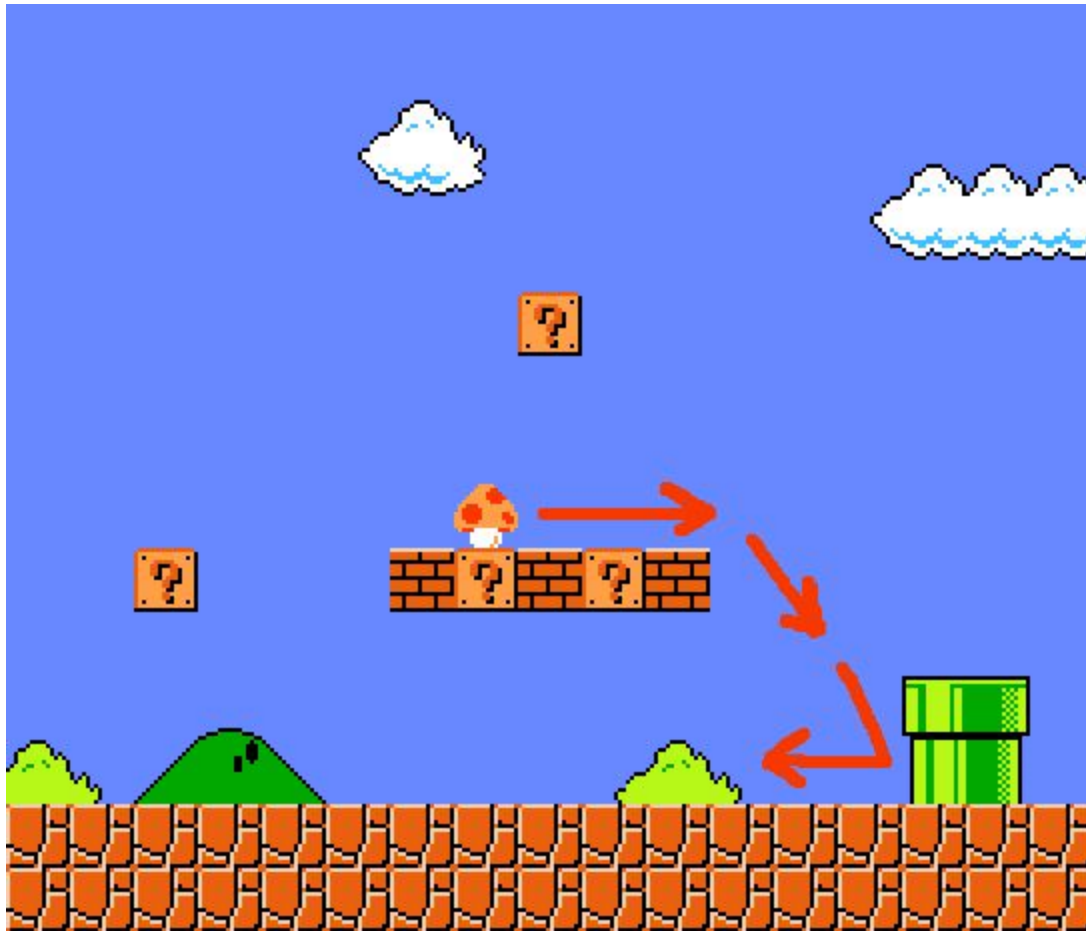
This is a recurring pattern in the design of Mario games, one that can continue to be seen even in recent games like Mario Odyssey. Give the player an open, safe area to learn their verbs in, then provide a bottleneck that asks them to apply those skills. (Look at the level progression of the first two kingdoms of Mario Odyssey for great examples of this.)

There are a few possible outcomes of the player's first attempt to jump over the goomba: Mario might not jump soon enough, or might land too early, and lose a life to the goomba. Or Mario might land on the goomba, defeating it. The goomba is slow-moving, and a new player is probably pretty slow-moving too, so this is not an unlikely outcome. Mario might even hit the ? block from below, causing a coin to pop out. Each of these outcomes provides information.

In [an interview](#) coinciding with the release of Super Mario Maker, Tezuka and Miyamoto revealed that the goomba had been introduced to the game primarily to fit the needs of this early screen. The koopa, which turns into a kickable shell when Mario jumps on it, was just too complicated to explain these basics of the game with. They needed the simplest version of a Mario enemy: Little guy, walks slowly, defeated when pounced on once.

After the goomba's been dealt with, the player discovers some more blocks in the sky, simple bricks and glowing ? blocks. The player might have already tried jumping into the first ? block, but these blocks are arranged to create a platform: Not only can you jump into them from below, you can also try standing on them from above. If Mario

jumps into the leftmost ? in this pattern, the second ? he's encountered, something dramatic and new happens.



A mushroom rises from the box and speeds to the right, along the platform. When it reaches the end, it falls, bumps the pipe, and reverses direction, and heading back toward the Mario who just released it. The player is given plenty of time to observe the mushroom's behavior before it reaches Mario. While later mushrooms have to be chased down (this is why they're so fast, to reward the player who can catch them) this mushroom comes to Mario. If the player does nothing else after hitting the block, Mario will still get his mushroom.

What if Mario tries to jump over the mushroom? It comes toward him just like the goomba did, and that thing was definitely dangerous. If Mario jumps, he'll most likely bump his head on the platform above and fall back down into the mushroom's path. Because of the mushroom's speed and the low ceiling, it's much harder to avoid than the goomba. The mushroom does have some semiotic qualities that mark it is as

different from the goomba: The mushroom doesn't appear until Mario touches a block, giving it a connection to the player's actions. It starts out traveling toward the right, the direction that Mario advances in.

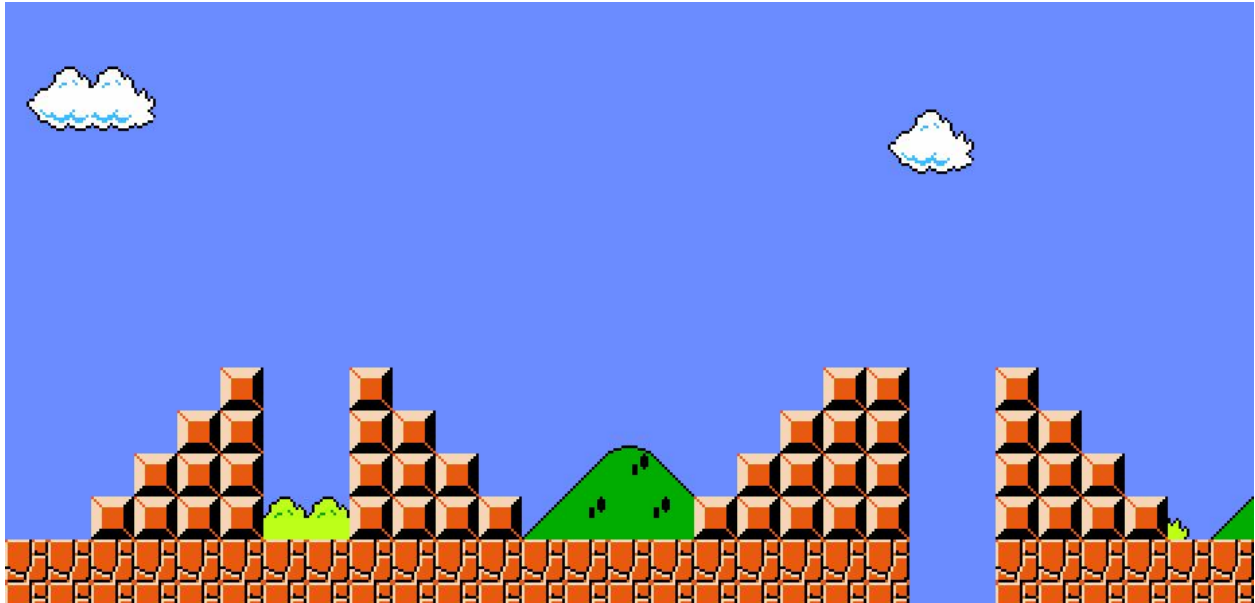
When the mushroom reaches Mario, a fanfare plays and Mario grows into big Mario. If Mario jumps while large, the player may discover that they can now shatter the bricks in the previously-impervious platform. Why might they jump? Well, there's another ? block hanging above the platform, and so far, the player's been rewarded every time they struck one. Breaking the bricks gives Mario additional paths toward it.

The pipe on the right is the final guardian of this area. It's a roadblock, preventing Mario from simply running past the blocks without exploring them a little. It also subtly develops Mario's jump verb. In order to get over the pipe, the player has to either hold down the button to have Mario jump higher, or to jump from on the platform. Either way, the player has to move Mario horizontally while also moving him vertically. (Both the goomba and the ? blocks could be "solved" just by jumping while standing in place.) To make sure the player has time to get the hang of it, there's no threat here: The pipe prevents further progress, but doesn't hurt Mario.

What ideas has the game introduced or reinforced so far? Here's a quick list: Mario progresses by traveling right. Sometimes he needs to jump to progress. He can use his jump to kill or avoid enemies, to hit blocks that make coins and power-ups appear, to smash bricks when big to open new passages, and to climb over obstacles. This is just two screens of the game, but they're very dense. Super Mario Bros. shipped with an instruction manual, but the level design here manages to reinforce all these ideas non-verbally.

This essay, first written in 2009, owes a debt to Jeremy Penner and Eric-Jon Tairne. Jeremy's piece Breaking the Law of Miyamoto informed my understanding of Super Mario Bros.' level design and Eric-Jon was the first person I ever saw use the word "verb" to describe a player mechanic.

Mario ascending a staircase. Super Mario Bros. 1-1.



These two adjacent screens are very similar. Each has a staircase leading up to a jump. There are slight differences between the two, though. The most obvious is that the first jump is over solid ground, the second over a deadly pit. The other difference, less obvious in-game but moreso when seeing them right next to each other like this, is that the stairs leading to the second jump are one block wider.

This is a great example of the dynamic I described before. A safe area to practice in with no threat, followed by a chokepoint designed to test the skill the player's been practicing.

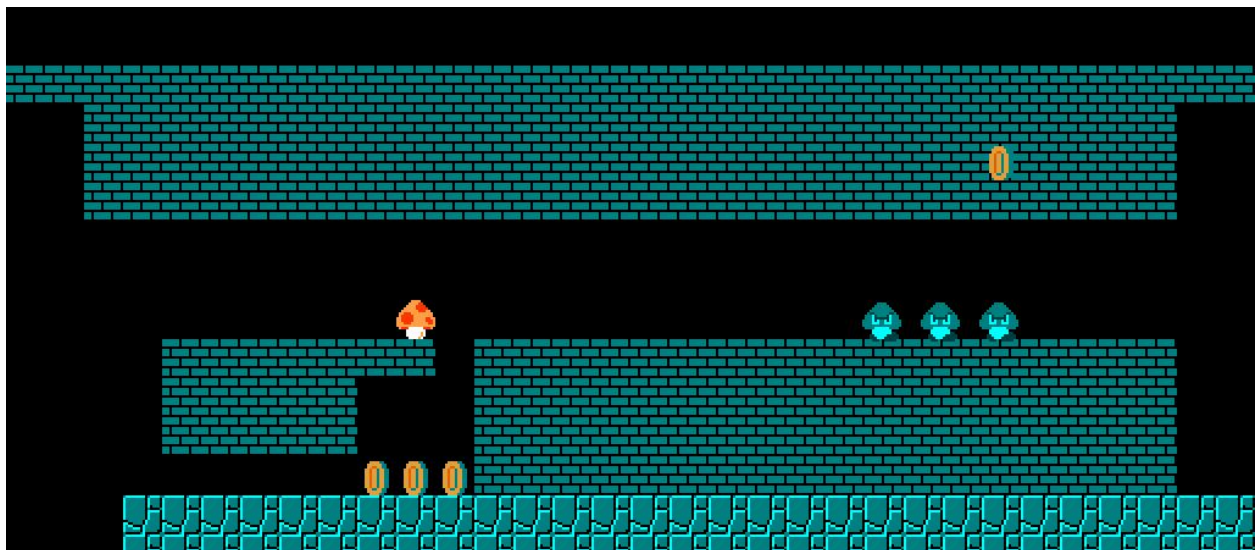
So why is the second one wider? Theory one is simpler: These screens were initially the same, but Tezuka and Miyamoto decided the second one (the one with stakes) needed to give the player a bit more room to build up to their jump. They didn't bother to change the first screen to match because it's not a big deal if the player falls there.

Theory two is complete conjecture, and hinges on the idea of learning by failing. By giving the player the opportunity to fall into the safe pit, the game is letting the player explore the boundaries of this setpiece. If they have the opportunity to try and fail here, that's practice. That's learning. When they encounter the second version, with the *real* pit, they'll know what failure looks like and understand the stakes. At the same time,

the designers subtly make the challenge easier than the first time by extending the top step.

The designer is dead. This is just me trying to retroactively understand decisions that might have been intuitive, creative, or deeply intentional, and were probably a combination of all three. No choice is made in a vacuum – even small level design choices impact the shape of the player’s experience. (Neither Miyamoto nor Tezuka are currently dead, to my knowledge. Don’t worry! That was a metaphor.)

Low overhead. *Super Mario Bros. 4-2.*



We’ll start with the right side of this area, the goomba hallway, because that’s the fun part. This is the equivalent of a Mario lateral thinking puzzle. The goombas march toward Mario down a three-block tall hallway. They take up a third of the vertical space here; so does little Mario. Big Mario takes up two-thirds of the space – meaning there’s uncomfortably little head room for jumping when Mario’s big.

The ceiling is made of breakable blocks, though, which means that the more Mario jumps, the less ceiling there is. The clever solution here is to break blocks to create head room for jumping over the goombas in anticipation of their arrival. (Their slow forward march gives the player a decent amount of time to prepare, but also a deadline.) There are also ten+ coins hidden in a block in the ceiling to reward players who continue to explore.

This setpiece is naturally the most fun as Big Mario. small mario has enough jump space that it's still a challenge, but a more straightforward one. Fire Mario can just blast through. (Big Mario and Fire Mario also have the option of just taking the damage and using Mario's recovery time to pass the remaining goombas.) But as a designer we want to give the player every opportunity to discover the interesting solution.

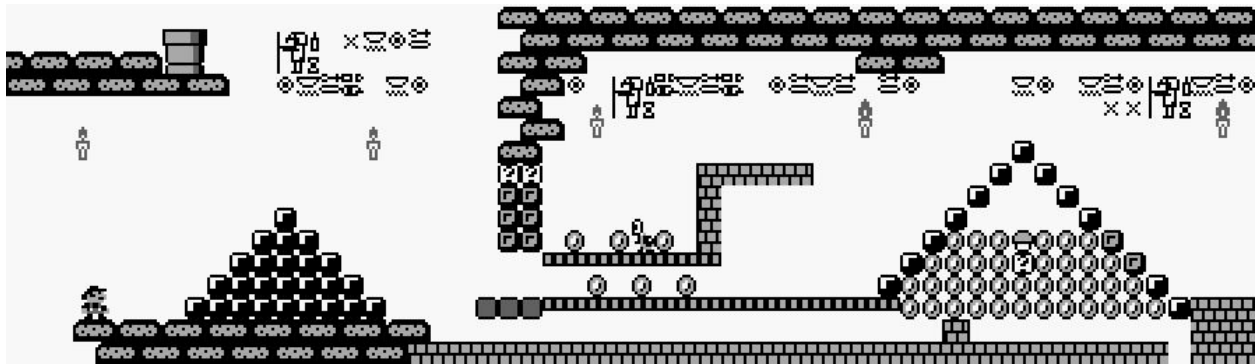
That's why there's a hidden mushroom at the start of the area. Not an exceedingly well-hidden mushroom: The block it's hidden in stands out like a sore thumb, and there are coins underneath to encourage the player to explore the area. There are even two ways into the little nook where it's hidden.

Well, small mario has two ways to enter: through the one-block-high crawlspace or through the "exit" hole to the right of the block. Even if the player doesn't intentionally search the suspicious block, it's easy to imagine them bonking it by accident while trying to exit the nook. Big Mario's only option to explore the area is to drop down the one-block-wide hole. While running, Mario will pass over gaps this small without falling. A Big Mario who can't fit through the front entrance will have to deliberately stop themselves to drop down through the back door.

Because of the power-up system in Super Mario Bros., the same block that gives little mario a mushroom will instead give Big Mario a fire flower, giving him the horizontal attack that makes this section trivial. This power-up area is arranged to be more accessible to small mario than to the big guy.

Super Mario's level design often emphasizes the affordances of a powered-up Mario, making the player feel more powerful as they unlock Mario's abilities. I like this bit because the game rarely supports the affordances of small mario: The ability to fit into areas his grown-up form can't.

Pyramid scheme. *Super Mario Land 1-3*.



Like the last example, this sequence from Satoru Okada's *Super Mario Land* plays with Mario's size-changing abilities, offering alternative paths for Big Mario and small mario. This set of screens appears midway through level 1-3. The first three levels of the game have a vague desert / "Egypt" theme that isn't really obvious until this stage. Each of these stages reproduces a familiar level theme from the original *Super Mario Bros.*: The flat level, the "treetops" level, and the "castle" level. This is *Super Mario Land*'s take on a castle.

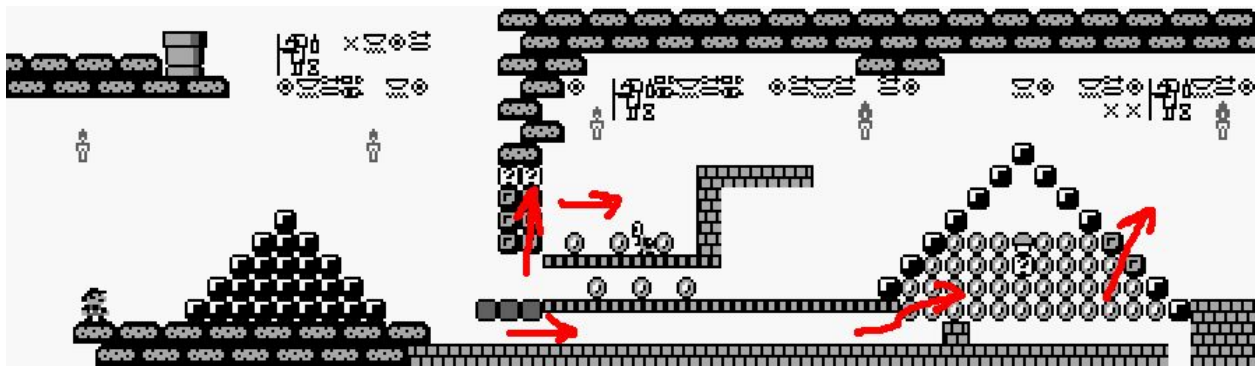
On the Game Boy, Mario is drawn a little bit taller than a block, but when figuring out where Mario can fit, the same size constraints apply as in *Super Mario Bros.*: small mario is the height of one block, Big Mario is the height of two. The beveled gray blocks in the above image (the ones underneath those ? blocks, for example) can be destroyed from below by Big Mario. When Mario is Big, touching an enemy will revert him to small. Small Mario is closer to death, but presents a smaller target for enemies.

This area halfway through level 1-3 offers a choice of three routes. The Upper route requires Mario to smash through some blocks and fight a Nokobon (that's the turtle's name in this one). The Middle one offers some coins. The Bottom one has a one-block-high entrance, and leads to a pyramid-shaped treasure chamber. *Super Mario Land*'s camera scrolls to the right, and prevents Mario from turning back to try other paths. The left edge of the screen acts as an immovable wall.

Two of these routes might be blocked off depending on Mario's state. Only Big Mario can smash through the blocks to access the Upper route. Only small mario is short enough to fit through the entrance to the Bottom route. Both Marios can access the Middle route, the least rewarding one. Centrism rarely pays off.

Most of the time, small mario is a less desirable state to be in. He can't break blocks, is only one hit from death, and can't use the superball weapon. (The superball is Mario Land's version of the fire flower. It's also more interesting than the fire flower. I said it.) In this area, though, small mario is able to access the most lucrative path through the level.

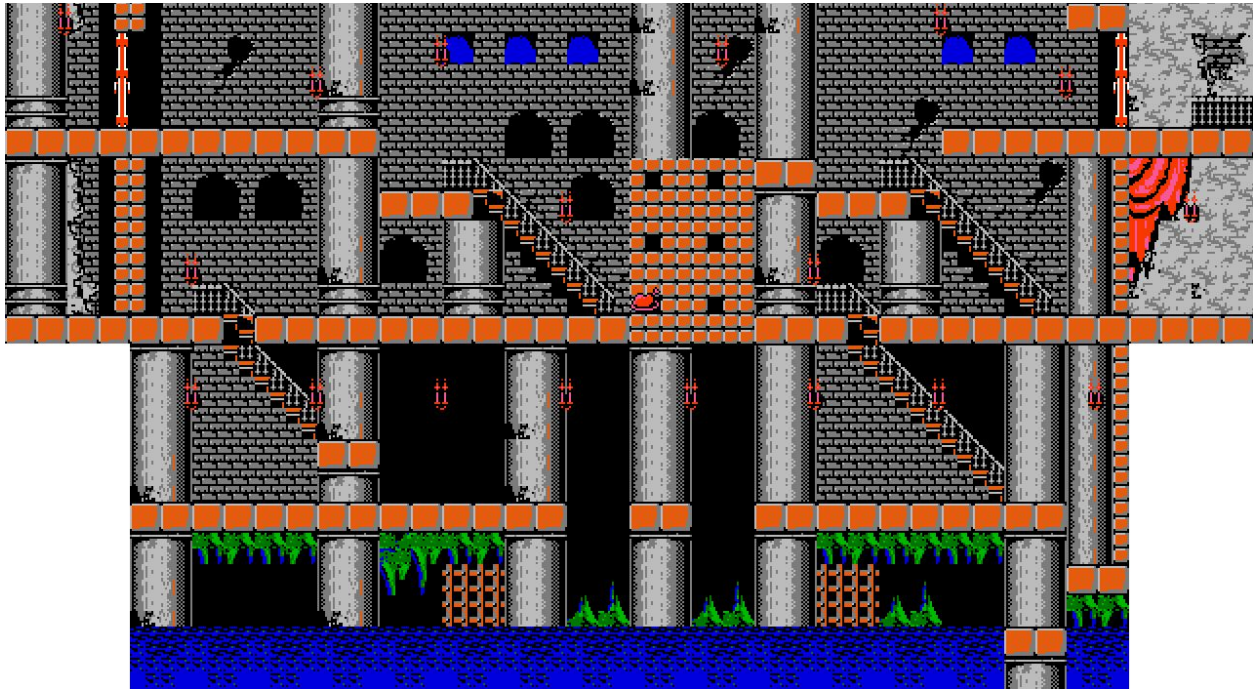
The coin stash is only visible after the player's advanced far enough that they can't go back and pick again. Cutely, the lower path is the only one that doesn't have treasure visible at the entrance. The player might miss it the first time, get defeated by the end-of-level boss, and remember the coin room when they start over at the half-way point. (Also cute: there's no immediate way to damage Mario and make him small right before the path branches.)



That treasure room, though. It contains 38 coins (a third of the amount needed for an extra life) and a mushroom. In fact, Mario can only leave by turning big and smashing his way out (via the two breakable blocks on the right side). The pit on the right does two things. First, it introduces time pressure by giving the mushroom a place to escape to. Second, it gives Mario a way out if he fails to catch the mushroom. *Grim*. The alternative is being stuck in the pyramid until the game's timer runs down. A fate worse than death.

Visually, shaping the treasure chamber into a pyramid is fun. It non-subtly underscores the "Egypt" theme of the levels (and more pyramids can be seen throughout the first three levels of the game, including at the left side of this image). But giving Mario the opportunity to traverse a maze of passages to find a secret passage into a pyramid filled with treasure tells a little story that is even more evocative of the theme. Mario is a tomb robber.

The shape of water. Castlevania, level 1.



I used to ask beginning game design students to design a one-screen platform level. Most students' first levels are the same. Go to the right, up, back to the left, up, and back to the right. The S-shape. The S-shaped level is designed to fill the area. It doesn't really communicate anything thematically or narratively about the space. It doesn't meaningfully use verticality beyond stacking pieces of level on top of each other. It's okay, they're freshman. They learn.

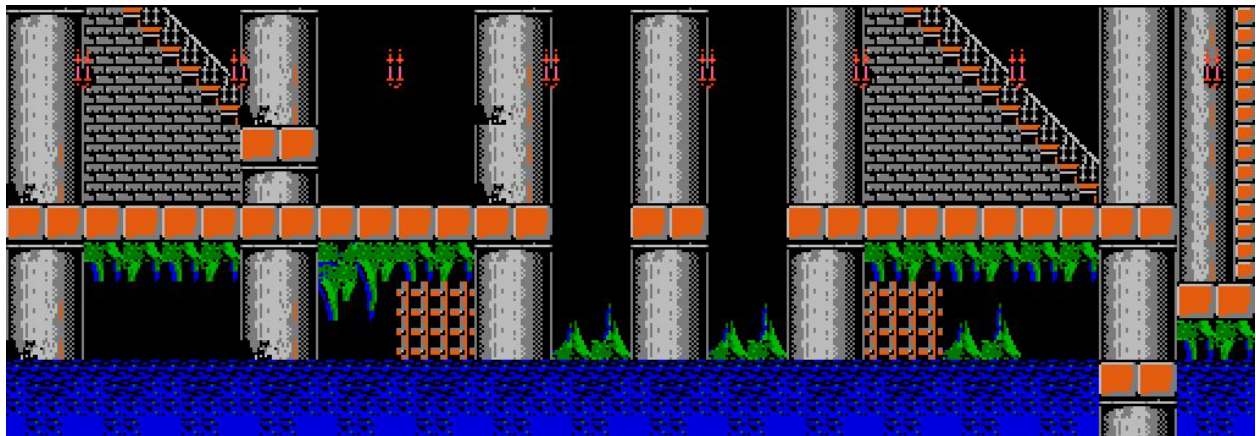
Let's consider the shape of Castlevania's first level. As a disclaimer, some levels of the game make use of the S-shape. This game is fairly linear. The first level, though, complicates the one-side-to-the-other path in a way that's interesting and fairly memorable. A callback to this level appears in almost every Castlevania sequel, and it always emphasizes the vertical basement dip midway through the level.

Simon Belmont, the game's protagonist, spends most of the level traveling from left to right, hitting things with his sword. He occasionally jumps to destroy candles and collect items, or faces a branching path where he can take the stairs to a higher platform or continue along the floor. For the most part, though, the shape of the level is horizontal.

About two-thirds of the way through the level, that horizontal movement is interrupted by an impassable wall. Visually, there's clearly a path on top of the wall that continues to the right, but Simon can't jump high enough to reach it. Famously, if the player whips the wall in frustration or confusion, a secret hunk of meat is discovered which heals the player. There's no further progression here, though.

Instead, the player must go back to go forward. Unlike in Super Mario (a game many of Castlevania's early players would be familiar with), the player can turn back to the left. In many later levels, the player will alternate traversing right and left in order to progress. Here, Castlevania introduces the player to the idea that forward will sometimes be to the left.

At the left is a staircase that the player would have seen when they entered the room. It leads to the moat under the castle.



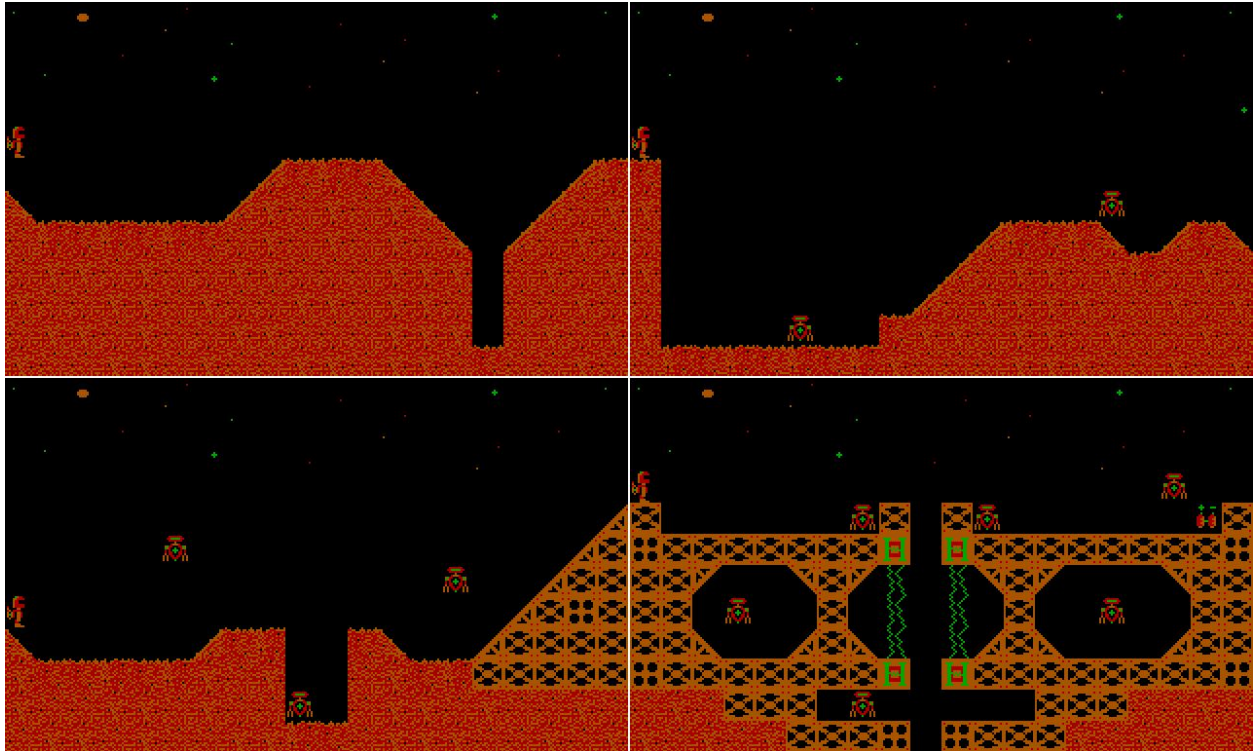
The moat is unique from the rest of the level visually and mechanically. Visually, it features deep water, dripping slime, and Black Lagoon-style fishmen who don't appear anywhere else. Mechanically, it develops the player's jump verb more strongly than anywhere prior. The fishmen spit balls that must be jumped over, and the moat includes two of the game's first deadly bottomless pits.

The visual design of the moat is doing a good job of making sure the player is aware of the danger these new obstacles present. Tall columns visualize where the platforms begin and end. The background tiles are switched out for a flat black background that contrasts well with the columns and creates vertical lines that span the entire height of the screen, drawing the eye down to the water at the bottom. This is a strong example of visual design supporting mechanical design.

The shape of the level and the placement of the moat area does a lot to support the feeling of the castle as a *place*, not just a collection of platforms and enemies.

Castlevania plays on a lot of horror movie tropes, trying to create a strong sense of setting. As the player progresses from level to level, the game repeatedly shows them where their current location is on the castle map. The moat scene gives this level a spatial dimension. What's under the castle keep? It's not more identical hallway, it's a basement moat. And if I go up from there? Back to the keep, the same room I left. Castlevania's map is spatially cohesive in a way that Super Mario's isn't. (I can go through a pipe anywhere in a Mario level and come back up anywhere else in the level.) The first level goes literally out of its way to demonstrate this.

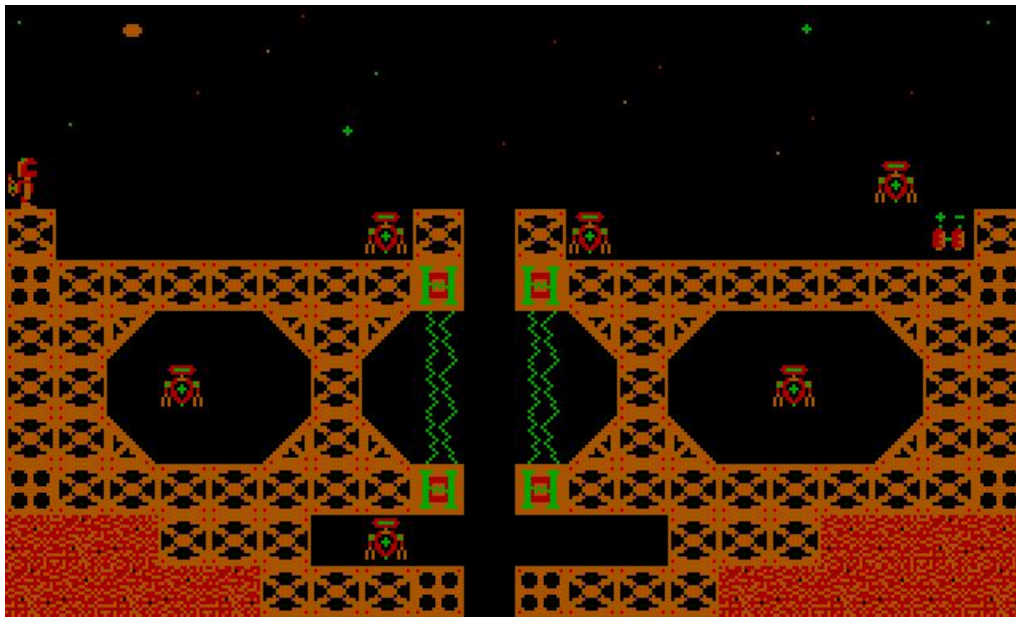
Silhouettes of Mars. *The Monuments of Mars!*, volume 1.



Todd Replogle's *Monuments of Mars* has some great examples of doing a lot with a limited design palette. *Monuments* is an episodic shareware game for MS-DOS. *Monuments* uses DOS' CGA (Color/Graphics Adapter) video mode, which means the game gets exactly four colors on screen at a time. CGA has several palettes, but *Monuments* is set entirely in red / brown / green / black. Given those colors, the Red Planet is a good match, but it's still a muted and samey-looking palette (the red and brown are way too close visually). That the game's opening screens work at all visually really testifies to the power of silhouette in screen composition.

Those first four screens cover the protagonist's journey across the surface of Mars to the alien "monument" where most of the episode will take place. Unlike *Mario* or *Castlevania*, *Monuments of Mars* doesn't have a scrolling camera. (MS-DOS wasn't designed primarily as a graphical game platform, unlike the NES, so most DOS games of the early 90s featured chunky tile-based scrolling or none at all.) Each of these screens is self-contained. The player travels from the starting position to the ending position, and then the game jumps to the next screen.

The composition of these screens uses visual continuity to make these independent levels feel connected to each other rather than a series of isolated challenges. Put them side by side, and the edge of one screen seems to match up with the next. This is most apparent in the second screen, where the player begins on a high ledge, and in the third screen, where we see the ground transition from dirt to the metal that will make up screen four.



Also helping to sell the feeling of a journey is visual contrast between the silhouettes of the “craggy” region and the “alien” one. The first three screens make use of irregular shapes (hills, drops, dramatic changes in elevation) to create the feeling of a more “natural” landscape. (It was designed by a creator, Todd Replogle, of course. But it feels less designed.) In contrast, screen number four – the entrance to the alien monument – uses symmetry to create the feeling of an artificial setting. On the microscale, the natural world is full of symmetries; On a larger scale, however, it’s hard to find the straight lines and boxes of architectural design in a natural landscape. If you look close you’ll notice that even the tiles making up the monument are symmetrical, while the pattern on the dirt tiles is more erratic and irregular.

This is why S-shaped levels don’t fly. The visual composition of a level gives the player their first impression of a level: What kind of place it is, where I should be going, whether it’s dangerous or safe. And, of course, whether I’m outside or inside, on a hill or an alien monument.