

£3
ALL FORMATS

Wireframe

LIFTING THE LID ON VIDEO GAMES

THE BEAUTIFUL GAME

The studios evolving the football genre

PLUS! Code your own
Sensible Soccer homage

BLOODLINES 2

Inside Hardsuit Labs' toothsome sequel

ALTERED VISTA

Games that recreate old user interfaces

TOOLBOX

Tips on world-building, narrative design, and more

DOUBLE STRIKE

YOUNG SOULS: THE BRAWLER WITH ATTITUDE TO SPARE

Issue 35 £3 | wfmag.cc



JOIN THE PRO SQUAD!



G-MASTER™
MONITORS 4 GAMERS
by **iivama**

RED EAGLE™

GB2560HSU¹ | GB2760HSU¹ | GB2760QSU²



Panel	TN LED / 1920x1080 ¹ , TN LED / 2560x1440 ²
Response time	1 ms, 144Hz, FreeSync™
Features	OverDrive, Black Tuner, Blue Light Reducer, Predefined and Custom Gaming Modes
Inputs	DVI-D ² , HDMI, DisplayPort, USB
Audio	speakers and headphone connector
Height adjustment	13 cm
Design	edge-to-edge, height adjustable stand with PIVOT



gmaster.iivama.com

We absolutely need escapism right now

This is a very weird time to be writing about video games, to be honest. As I type this, Europe is increasingly on lockdown to help halt the spread of COVID-19. By the time it's published, we may very well be taking similar measures in the UK, where I live.

Suddenly, everything I had to say about games a few days ago – about *Animal Crossing* and the immense value of the chill game, or about how my anticipation for *The Last of Us 2* has been tainted by not-unexpected but nonetheless depressing reports of overwork at Naughty Dog – seems rather trivial.

But actually, video games are something that can genuinely help in these unprecedented times. With millions worldwide stuck at home, spending many hours playing games is perhaps finally going to be understood by the wider world as a vital escape rather than a waste of time. I started playing *Animal Crossing: New Horizons* for review a couple of weeks ago and since then, it's gone from a welcome, calming thing to play while hiding from my children in the toilet to a treasured oasis from the state of the world right now. Instead of staring at horrifying news live blogs on my phone, I'm planting orchards and watering tulips on a nearly deserted island. How's that for self-isolation?

Anyone who plays games will know their immense escapist power – their ability to take you, just for a while, into a different life in a different place. It's the ease with which games can lift you from your real-world circumstances that makes them such an important refuge for people struggling through a mental health downturn or difficult times of life. Whether you find that escape through tending a *Stardew Valley* farm, disappearing into *Breath of the Wild's* Hyrule for a while, or powering through ten consecutive *Apex Legends* matches, the effect is the same: in hard times, games can be an excellent distraction.



**KEZA
MACDONALD**

Keza is The Guardian's video games editor, currently on leave with her two small children, and with a desire to play a lot of *Animal Crossing: New Horizons*.

And not just that – games are also an excellent source of social contact without any physical contact. I predict a significant uptick in *World of Warcraft* subscriptions and hours spent in *Warframe*, *Fortnite*, and *Destiny* in the coming weeks and months. This has already happened in China, where the Hubei lockdown led to a huge uptick in online gaming at the beginning of the year, with 150 million people playing mobile game *Honor of Kings* in one day. Game streaming viewership was also way up, and all major esports tournaments were played online.

Some developers have started adapting to this situation, extending free trial periods or making changes to online games. Even *Pokémon GO*, the ultimate game about getting outside, has made some adaptations – incense, the in-game item that allows you to spawn Pokémon wherever you are, has been made available for almost no money (1 Pokécoin), and Niantic has extended the duration of its effect as well. It also takes far fewer steps now to hatch Pokémon eggs, meaning you can theoretically do it by marching around your apartment.

If I end up stuck at home with my small children for several weeks, I'll be spending whatever spare hours they allow me hanging out in *Animal Crossing*, where I plan to spend many more hours chopping wood, making furniture, and fishing with friends for some social connection instead of gazing into the hell-portal that is social media during a pandemic. This kind of 'simulated' socialisation is exactly the kind of thing that my parents rolled their eyes at when I was well into *Guild Wars* as a teenager, but now it's got the potential to help us all cope if we end up in coronavirus prison.

This is a time where games can have a gigantic positive impact by helping people stay the hell at home under very depressing circumstances. If studios or publishers can afford to make games available cheaper or for free, now would be an amazing time to do it. 🙏

#35



Contents



6

Attract mode

Interface



22

- 06. Young Souls**
The co-op brawler fighting goblins and prejudice
- 10. Bloodlines 2**
Hardsuit Labs on reviving a cult classic for the modern era
- 14. News**
The latest headlines from the virtual front
- 16. Incoming**
Brawling robots, delivery drivers, and nineties video stores

- 18. Liquid football**
The studios transforming the beautiful game
- 22. Archaic interfaces**
Indie games that recreate yesterday's UI spaces
- 44. In the beginning**
Turochamp, and other pioneering video games
- 50. Tecmo**
Profiled: the studio behind Ninja Gaiden



Toolbox

- 28. Design principles**
Inside the mind of the computer programmer
- 32. Substitute Soccer**
Code your own top-down football game
- 38. Hyper-screenplays**
Exploring the world of game design documents
- 40. Source Code**
Remake Hyper Sports' skeet shooting minigame

Rated

- 56. Zombie Army 4**
The schlocky series comes back stronger than ever
- 58. Coffee Talk**
A visual novel that captures the daily grind
- 61. The Pedestrian**
The signs point to a truly absorbing puzzler
- 64. Backwards compatible**
HDMI solutions, and other retro hardware delights

WELCOME

As Keza MacDonald already pointed out on page three, COVID-19 is now a present – and genuinely worrying – reality for many of us. And with much of the western world self-isolating to avoid spreading the virus, it's clear that the economic impact of the pandemic will be considerable, too. The cancellation of events like E3 and EGX Rezzed (which you can read more about on page 14) are going to have a profound impact on our favourite hobby; indie developers, who rely on those events to make deals and get the word out about their games, will be particularly affected, as are charities like the National Videogame Museum and the Centre for Computing History, which have both been forced to close due to the coronavirus situation.

Thankfully, the industry's pulling together to help those affected: the GDC Relief Fund was set up in early March to assist indie devs who've lost money from the cancelled expo, and gamedev.world is launching a fundraiser from 27 March to 3 April to boost donations – you can find out more about that at wfmag.cc/gdc-relief. There are things you can do to help those museums, too: you could join the NVM's Patreon at patreon.com/thenvm, or you could make a donation to the CCH on its website (computinghistory.org.uk).

Above all, though, stay safe, stay calm, and look after yourselves. Like Keza, I'm going to be spending a lot of spare time immersed in the soothing world of *Animal Crossing: New Horizons*.

Ryan Lambie
Editor





Fighting Fantasy

Battling prejudices by day and goblins by night in *Young Souls*

With a core team of just two people, it's fitting that France-based 1P2P Studio's debut title is a two-player game. The studio's name also evokes the '1P' and '2P' names from old coin-op games, which is also appropriate, given that *Young Souls* takes its cue from Sega's classic 2D beat-'em-up, *Golden Axe*. It might seem like a turn to old-school fare for 1P2P co-founder Jérôme Fait, whose career has included much larger-scale projects, including a stint at French MMORPG company, Ankama. Nonetheless, Fait points out that *Young*



Souls is more than just an old-fashioned brawler, since it also incorporates more than a few modern elements. With the word 'dark' in the title, you might expect a *Dark Souls* influence here, especially when it comes to harsh but fair combat – though Fait says the game's influences also hail from elsewhere. "More recently, we were inspired by *Dragon's Crown*," he says. "We liked its structure; we liked the fact that you mix dungeons and beat-'em-up gameplay with RPG mechanics. So we tried to recreate the same kind of things, but with a [similar] modern city setting."

The game's split into two worlds, with its modern-

day town acting as a hub for training, exploration, and acquiring new gear in the daytime, before you journey to a medieval fantasy world to beat the snot out of a horde of goblins by night. Employing a day-and-night cycle to distinguish these two worlds has a certain *Persona* ring to it, though the design choice has another motivation, according to Fait. "The day cycle was created for the feeling that becoming a hero takes time," he says. "You know, you're not just becoming a hero in only one day. This is kind of an origin story of how you begin as marginals and then become heroes."

That gradual process of becoming a hero is naturally tied to the concept of any RPG where levelling up and acquiring



more powerful gear is crucial to advancing further into dungeons with more powerful adversaries. At the centre of it, though, are the *Young Souls*' protagonists: twin siblings Jenn and Tristan. As Fait points out, they're outsiders – a status neatly illustrated in an early scene when someone in town snidely calls them “ginger weirdos.” While games have had their fair share of redheads in the past, this definitely feels like the first time we've noticed an overt case of what some parts of the world, including France, consider the last acceptable form of prejudice. “We didn't want to play with that aspect at first,” Fait admits, mentioning that he had picked their hair colour purely as an aesthetic choice to make them stand out in a positive way. “But these kids are also considered outcasts. They're not really liked in the city, and people in town use



Attract Mode Early Access

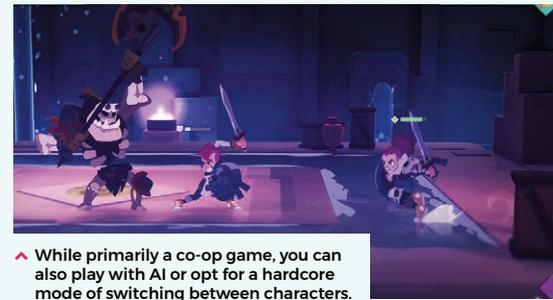
^ That's a cute moped to get around town, though whether or not you can take it underground with you is another matter.

everything to bully them, from their hair to the fact that they're orphans.”

In the face of all this, it's up to the duo to save the town that rejects them, as unbeknownst to the rest of the human population, a goblin race from deep underground is slowly digging its way up to invade the surface. On a more personal level, the twins are also out to rescue their kidnapped adoptive father, the Professor, who's housing a portal in his secret lab that transports them to goblin-infested dungeons. With a story penned by Matthew Ritter, whose credits include Telltale Games' *The Walking Dead* as well as titles such as *Life is Strange 2*, there definitely appears to be more to *Young Souls* than just dungeon-brawling.

Combat is certainly the main focus for the demo, however, and it feels positively slick in its animations, aided by a winning pastel cartoon aesthetic that complements publisher Dotemu's previous releases, like *Wonder Boy: The Dragon's Trap*. The action's fairly straightforward, as you progress to an area and find the door sealed until you've defeated all the spawning enemies. As well as attacking with weapons, you can also block with your shield, roll to dodge, and make use of magic spells, all of which are governed by stamina and mana gauges respectively. These grunts aren't pushovers either, but most also leave behind a generous dollop of hearts, and there seems to be no limit on how often you can revive your teammate to carry on the fight.

Defeating goblins will also earn you loot, or more specifically a mixture of resources from tokens to runes that can be used to exchange better equipment in town. (It's amusingly apt for a game featuring teenagers that, aside from all the expected fantasy gear, buying cool sneakers is also a way of altering your characters' stats.) It's too early to tell just how in-depth the



^ While primarily a co-op game, you can also play with AI or opt for a hardcore mode of switching between characters.

“These kids are considered outcasts. They're not really liked”

loot system and customisation will be, though Fait says that the game's crafting system has been purposely designed to be simple. “In fact, we wanted a kind of easy co-op gameplay,” he explains. “And when you play in co-op, you don't really take as much time thinking about what you're trying to make as you might when playing single-player. So we want something easier to understand.”

After three years of development, the main focus now is on getting the gaming balance right before its 2020 release. But *Young Souls*' vibrant and likeable visuals, accessible yet challenging co-op gameplay, and a desire to tell a deep narrative sounds like a winning combination. Hopefully, it'll also make Jenn and Tristan poster children for redheads everywhere. ➔

READY TO RUMBLE

The 2D brawler has been having something of a resurgence recently, from the remaster of former Xbox Live Arcade cult hit *Castle Crashers* to *River City Girls* that gives a refreshing gender-flip to the long-running Japanese *Kunio-kun* series. Incidentally, *Young Souls*' most formidable rival comes from its own publisher Dotemu and developer Lizardcube's hotly anticipated revival of a Sega classic with *Streets of Rage 4*. It's good healthy competition though, while *Young Souls*' light RPG elements also allow it to stand out.

Attract Mode

Early Access



◀ Nothing like finding a portal in your house that takes you into a world filled with angry goblins.

Having gone hands-on with *Young Souls*, it only seemed right that we follow up our play session by firing a few more questions at the dynamic duo behind the game. Here's what Jérôme Fait and 1P2P co-founder Baptiste Martin had to say.

Why and how was *Dragon's Crown* such an inspiration? What else did you look to?

The structure and the rhythm of *Dragon's Crown* inspired us, in addition to its mix of genres. We love the instant satisfaction provided by the arcade/beat-'em-up gameplay, and adding a modern touch by mixing it with hack-and-slash elements spoke to us a lot.

From a gameplay point of view, we wanted something less conservative, so we moved away from this model. *Young Souls* is deeper, with much less emphasis on button mashing and promoting a freer and smarter approach to combat.

For the mood of the game, we went looking for games at the opposite end of the spectrum, like *Night in the Woods*. We're fans of the independent productions of the 2010s, which also influenced us a lot. *Young Souls* is the result of mixing it all up!

There seems to be a lot more focus on story than you might expect from the brawler genre – why is that?

We love to work on arcade games – they're both accessible from the get-go and demanding in the long term. But to us, the gameplay alone is not enough – we like to tell stories, which brings more 'soul' to the game. The story supports the action, reflects the player's progress, and stimulates their imagination beyond what

the mechanics offer by focusing on the characters and journey.

It's something we felt both by working on narrative games – survival horror, specifically – and our gamer experiences on triple-A titles, offering both accessible gameplay and a story-driven experience.

“We love to work on arcade games, but to us, the gameplay alone is not enough”

A good inspiration for us in this action-narrative genre was definitely *God of War*, especially the first and last games. The games are great, really tight, and at the same time, their stories bring an extra layer, beyond the damsel in distress trope usually found in classic beat-'em-up games such as *Double Dragon*, *Final Fight*, and so on.

With *Young Souls*, we wanted to reflect the player's experience through the script and develop the characters accordingly.

The initial idea was to feature a co-operative mode, so we took this opportunity to centre the story around two characters with well-developed, complementary personalities, a story that revolves around twins, an initiation quest with teenagers.

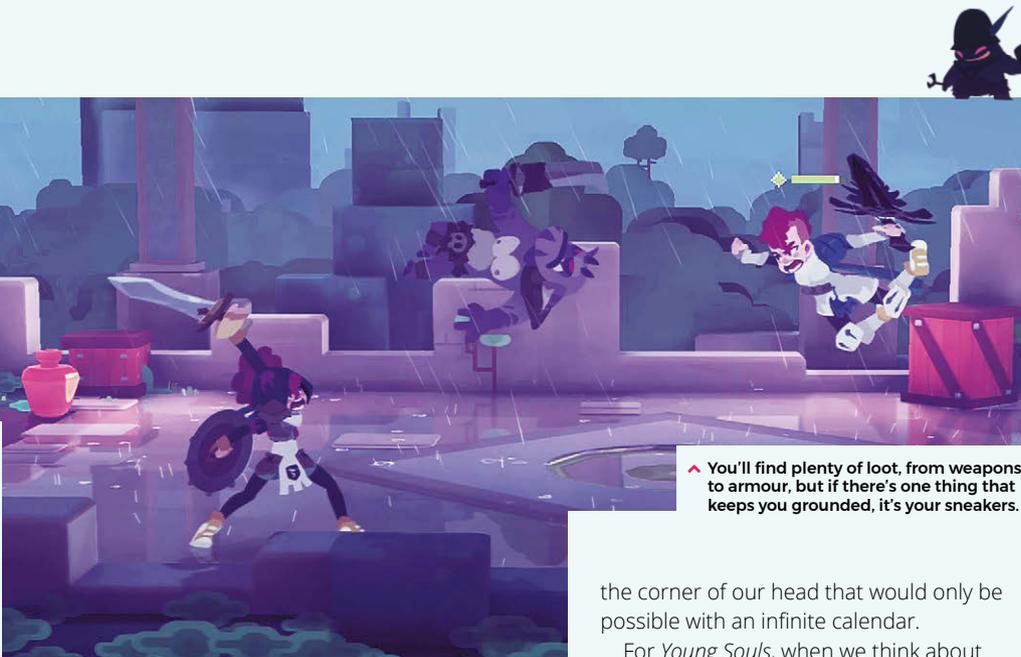
How challenging is it to find the balance between presenting players with something familiar when it comes to the genre, but also bringing in new elements to keep things fresh and engaging?

As with the game's genre, we mix ideas that are quite common in the eyes of the players but that have rarely been associated. This allows the player to be on familiar ground, without giving them an impression they've already played the game.

We respect a few key codes of the genre, recognisable elements for the player – old-style 2.5D gameplay, progression in dungeons, and so on. The new elements are there to bring originality and a personal touch. We wanted to push the game in two real ways: deep gameplay, using different weapons with their own movesets and gameplay mechanics, stamina and mana management, and crowd control; and through its content 'volume', providing a wide range of tools and skills to unlock – weapons, spells, accessories, mounts, and

▼ Combat takes influence from the likes of *Dark Souls*.





^ You'll find plenty of loot, from weapons to armour, but if there's one thing that keeps you grounded, it's your sneakers.

so on – so the player does not need to be a fighting game purist to enjoy it. Many different approaches are possible to make progress.

The difficulty of this design lies mainly in teaching. Making a rich game like this requires careful going, and finding the right balance between introducing interesting features and the timing of their introduction into the game progression. You can't unlock everything too fast, [otherwise] the player would be overwhelmed by the number of mechanics to learn. And because we want to offer accessible gameplay, we have to introduce elements gradually.

It's difficult, because you have to test the game a lot and not hesitate to change things here and there, especially in terms of the game structure, which has evolved a lot during the development.

Three-plus years of development is a long time for any game – how much has the project changed since day one?

In the end, the initial ideas have changed little; it is mainly the project ambitions that have grown over time.

Our publisher, The Arcade Crew, has been very supportive. They gave us the means to push the game much further than we had originally planned, both in terms of quality and production volume.

Was there any feature you had to pull from the game?

No matter the scope of the game, you are always tempted to add more than you can do, so we keep this fantasy version in

the corner of our head that would only be possible with an infinite calendar.

For *Young Souls*, when we think about it, we're not that far from our idealised version. We put a lot of things aside, but in the end, today's version is pretty close to our ideal one.

You're a core team of two – how have you found it working in such a way? Would you recommend it?

The core team is indeed made of two people. We also worked with five freelancers and a few interns, to varying degrees of investment in the project.

Two of us covered the game direction, the game design, the animations, most of the environments, and the code.

We worked with freelancers, experts in their fields, to fill our gaps and push the game as far as possible in terms of quality – they helped with audio, character design, concept art, storyboard, and FX. We worked exclusively with people we knew well to avoid spending too much time on management tasks.

With few people on the team, the production is long but qualitative. Communication is smooth – we have been working together for 15 years – so it means we're not wasting time in sterile debates, because the project direction is clear and we are on the same wavelength.

But this small team also induces a huge amount of work and responsibility; it is also necessary to be able to change tasks frequently, to make both hemispheres work. It suits us perfectly, we knew we wanted to do it like that since day one, but really it depends completely on the developers' personalities. 🗨

Young Souls releases later in 2020.

STORYTELLING MASTERS

Young Souls is attempting to be a beat-'em-up with a story people will actually get invested in, which is at odds with the classics of the genre. In the interest of scientific truth-seeking, we looked through the competitors of the Golden Era of scrolling brawlers to see just what epic tales were being told back then, and what *Young Souls* has to compete against.



Double Dragon

Billy's girlfriend Marian is kidnapped by the Black Warriors gang, so he teams up with his brother Jimmy to beat up literally everyone they encounter.



Double Dragon II

Billy's girlfriend Marian is murdered by the Black Warriors gang, so he teams up with his brother Jimmy to beat up literally everyone they encounter.



Final Fight

Haggar's daughter Jessica is kidnapped by the Mad Gear gang, so he teams up with her boyfriend Cody and friend Guy to beat up literally everyone they encounter.



Streets of Rage

A gang has taken over the streets, which are now filled with rage. Three vigilante ex-cops use their bare knuckles to beat up literally everyone they encounter.

WHAT WE DO IN THE SHADOWS

Cult classics aren't supposed to get sequels, but here we are with *Vampire: The Masquerade – Bloodlines 2*. Time to suck it up

It's still a surprise to remember that, in fact, yes, *Vampire: The Masquerade – Bloodlines 2* is a thing that exists and is releasing in 2020. A sequel to 2004's game of the same name (minus that identifying number) really does feel like the stuff of impossible dreams, even now, mere months ahead of its launch. And with that self-imposed hype comes pressure: Hardsuit Labs, developer of *Bloodlines 2*, only has one launched title under its belt, 2012's free-to-play FPS *Blacklight: Retribution*. *Vampire: The Masquerade* has a committed fanbase. *Bloodlines*, Troika's original game, has something of a rabid fanbase. If Hardsuit gets it wrong, the community will – somewhat fittingly – be out for blood.

So it is that we find ourselves in a situation where there's real trepidation about a much-wanted sequel. When a game releases, withers on the vine, but lives on through community support and intense fandom – as *Bloodlines* did – there's rarely a real hope of anything official following it up. But that announcement, those first looks at the sequel that *does* exist, that release date offering an actual time and place you'll be able to play it. It's all a bit too much. Surely it's a fever dream? Surely it can't go well? Surely something has to go wrong? We spoke to Russell Nelson, CTO and co-founder at Hardsuit Labs, to find out why, actually, there's a real confidence this could be the impossible sequel *Bloodlines* fans have been craving.

How long has *Bloodlines 2* been in development?

The moment we found out that Paradox had acquired *World of Darkness*, Ka'ai Cluney, our creative director, went to work. He contacted Brian Mitsoda, who was the lead writer on the original *Bloodlines*, and who he had worked with previously, and asked if he wanted to make another *Bloodlines* game. Brian was in so the two of them literally worked out the setting and main story beats over a weekend and a bottle of whiskey. Ka'ai then went to Andy [Kipling], CEO of Hardsuit Labs, and asked if he knew anyone at Paradox.

We pitched the game to the team at Paradox in February 2016, much to their surprise and delight. It's extremely rare for a small game studio to spend their own time and capital on developing a pitch to a single company. It's high-risk, and Paradox honestly didn't quite know what to make of the request. But after the pitch, they knew everything they needed to – that we could make this game.

Pre-preproduction began in March 2016, and we were officially green-lit in October 2016.

What was the mood like in the studio before revealing the game's existence?

Our biggest fear was that we'd announce the game and people would respond 'So what?'. But that genuinely wasn't the case. When we did closed-door playthroughs before the announcement, people would walk in, see the opening splash screen

with the logo and respond with genuine excitement. That was a huge indicator to the team that *Bloodlines 2* would be well received. This is the first original game made by Hardsuit Labs, and it is the follow-up to a cult-classic. While in development, you are so close to the game that all you see are flaws – what you don't see is the reaction of a first-time player, especially the reaction of a fan who loved the original game.

Have things changed around the studio much since beginning the project?

We have tripled in size since the start of the project, and there are always challenges associated with growing that quickly. Each challenge has given us an opportunity to learn, and the important thing is that we keep moving forward. One key learning was that we needed more producers, so we have expanded a lot in that regard. Production helps keep people focused on important tasks and avoid randomisation. Production can be a force multiplier when it comes to focus. ➔



▼ *Bloodlines 2*, as you might expect, avoids most natural sources of light.





How does Hardsuit's experience with *Blacklight* feed into creating a single-player RPG?

There are quite a few foundational things we were able to take from the experience making *Blacklight: Retribution* and apply them to *Bloodlines 2*. Things like optimisation, user flows, first-person camera controls, and hierarchies in messaging. Player progression and in-game economies can translate well, but the player motivation is different. Most of *Bloodlines 2* was built from the ground up with a team crafted around the RPG experience.

Were there any tricks you learned from work on *Blacklight* that transferred over to *Bloodlines 2*?

Run on the lowest-end platform you intend to support as much as possible. More importantly, make certain your tools are friendly to your developer colleagues that are creating content; nothing is more important than removing the friction of adding an asset to a game. Also, playtest with as many diverse groups as you possibly can. There's so much that goes into making a game that it's easy to miss something critical, but having many different eyes, ears, and appendages on the game can help.

How does *Bloodlines 2* factor in the tabletop RPG it is based on? Does it stick strictly to the rule set, as well as to the overall narrative themes?

The first *Bloodlines* game was very true to the tabletop, many of the systems and interfaces being direct translations from the tabletop game. This wasn't always successful in a video game environment



^ Friend or foe? You decide. Wait, not that. He's definitely a foe.

since the two formats and experiences are quite different. For *Bloodlines 2*, we are less interested in a one-to-one implementation of the tabletop systems and more interested in what is at the core of those mechanics: being a vampire.

Bloodlines. But more importantly, we are more conscious and sensitive to certain subject matter in this game. For example, the Malkavian clan deals a lot with mental illness. This was used mostly for comedic effect in the first game, but we have a

“We looked at the things we wanted to keep and where we could improve on the first game”

When [we were] developing the core gameplay of *Bloodlines 2*, we [remained] in constant contact with the tabletop team while they were working on V5 of *Vampire: The Masquerade*. Many of our systems were developed together.

While the game is a sequel, is it still fair to say it's quite the departure from the original *Bloodlines*?

Throughout development, one of the game's core pillars is to be a true successor to *Bloodlines*. That means everything we do is in service to that goal. It's been great having Brian Mitsoda back to both write and guide the game – his involvement ensures we stay on track to achieve our ambitions for *Bloodlines 2*.

The major differences between *Bloodlines 2* and the original game that came out in 2004 are the technology and subject matter. We are using Unreal Engine 4 to develop the game, which is incredibly powerful and allows us to do things that weren't remotely possible with the original

greater understanding of mental illness and how it impacts people now. This is a subject we are not taking lightly, and we want to give the Malkavians' representation of mental illness the respect and dignity it deserves in *Bloodlines 2*.

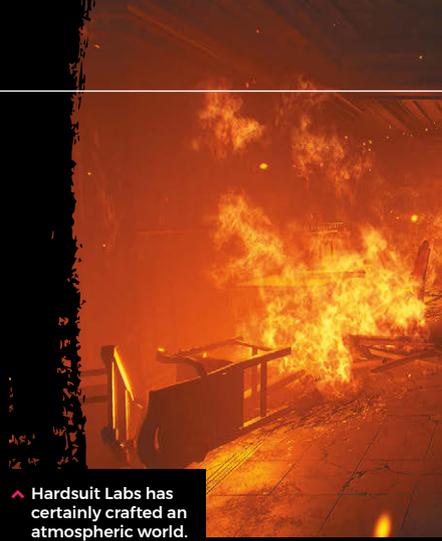
How is the representation of vampires being modernised or updated? Is it even being modernised/updated to begin with?

The mechanical parts of being a vampire are timeless – drinking blood, immortality, aversion to daylight – but it's up to the player to navigate how to do those things in a modern setting. *Vampire: The Masquerade*, and the larger *World of Darkness*, have always explored what it means to be a monster. How does being immortal impact your daily life and outlook on the future? What does feeding on human blood mean to you ethically? How do you approach relationships with other monsters, especially when resources are on the line? How can you survive in a world with increasing connection and isolation?

Bloodlines 2 takes place in the modern day, just as the first one did. There have been some major societal shifts in the 15 years between games, and we've been able to integrate them into *Bloodlines 2*. For example, the Masquerade is more difficult to maintain out in public because of the abundance of mobile devices and cameras. Anyone with a smartphone can take photos of vampire shenanigans and immediately show it to the world, resulting in harsh consequences for the player. But we balance that with areas where the player can perform those things outside of public view, either where they won't be seen or where people no longer care. Being a vampire hasn't changed all that much, but the ability to *be* a vampire is something we get to leverage in the playable spaces of the game.

What's been something you were champing at the bit to change, upgrade, or fix from the original *Bloodlines*?

We looked very carefully at the first game and reviewed what was needed for the sequel. We looked at the things we wanted to keep and where we could improve on the first game. The combat experience was at the top of the list of opportunities for improvement. We really wanted to emphasise the vampire strength, agility, and abilities in combat to make the player feel like a monster. When you're a vampire, weapons are meant to be fleeting accessories, your fists tear through human



^ Hardsuit Labs has certainly crafted an atmospheric world.

flesh like tissue paper, and vampiric abilities – or Disciplines – are powerful and fulfilling to use. Players can customise their preferred combat methods – they can even create a character that can resolve some situations through manipulation and coercion to get what they want.

What's been the most challenging aspect to bring into the sequel from the tabletop original?

Player choice and reactivity is at the core of pretty much everything we do. Tabletop RPGs are great because of how free-form and choice-driven they are, which gives players a lot of freedom. The players can do anything at any time. Video games are more restrictive because of the format and the technology, but we can still design for enhancing player choice and reactivity. The player may not have the same amount of freedom as the tabletop, but we can get very close with good design.

When developing the game, we always keep player choice in mind. How would a player with a non-combat build get through this encounter? Can we have more entrances to this building that take advantage of the player's Discipline choice? If the player puts all their points into this one skill, how will that impact their game experience? The player gets to decide the



^ Death, unsurprisingly, plays a central role in the world of the Masquerade.



^ Unarmed? Sure, but also you're a vampire. Tends to help a smidge.

monster they want to be, and our goal is to have the game react accordingly.

It has to be strange dealing with the hype/expectation surrounding *Bloodlines 2* – how do you deal with that strange mix of pressure and lack of pressure?

You concentrate on making the game you want to make. While we felt the pressure from expectation early on, the more we worked on it, the more secure we were in knowing we were making a *Bloodlines* game. While the first game didn't sell well at launch, over the years it has become a cult classic that has millions of fans all over the world. To this day, you'll be hard-pressed to find a fan of RPGs who hasn't at least heard stories about the greatness of *Bloodlines*. If the response from the gaming community at large is any indication, I think we have a hit on our hands.

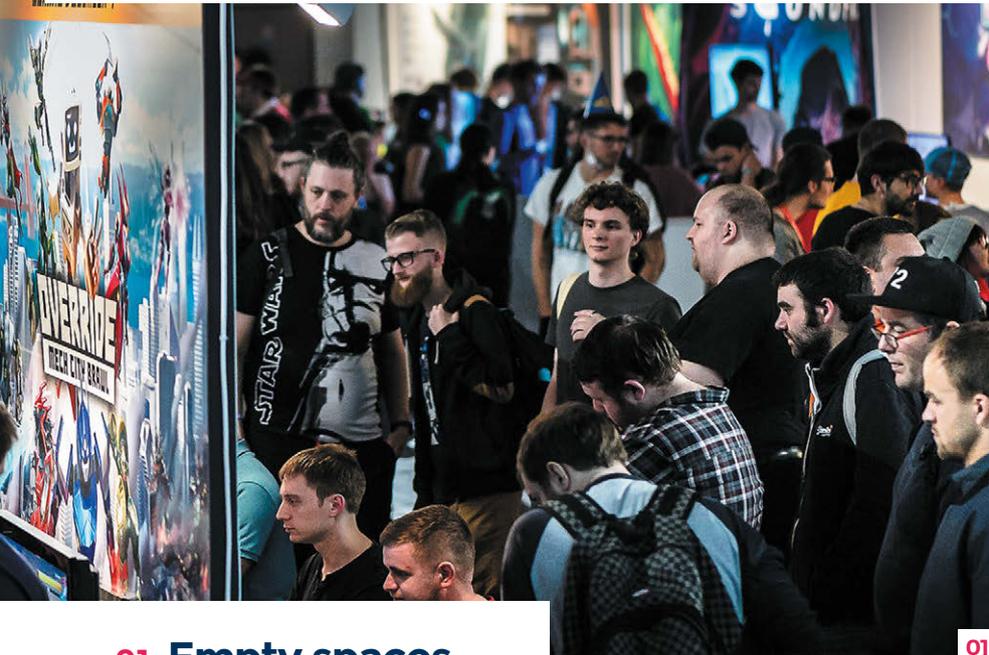
What are your hopes for *Bloodlines 2* on release? What do you want to see it achieve?

We want to deliver a true successor to *Bloodlines*, and everything that goes with it – something fans of the first one and players new to the franchise will enjoy. We want people to have meaningful conversations about the story and the choices they made. We, as a team, understand how much the original means to people (because it means a lot to many of us), and we're not taking that lightly. 🍷

Vampire: The Masquerade – Bloodlines 2 releases on PC, PS4, and Xbox One later in 2020.

Headlines

from the virtual front



01. Empty spaces

Cancellations related to the ongoing COVID-19 pandemic continue, understandably, with events large and small around the world being pushed to later in the year – or cancelled altogether. One of the biggest to fall has been E3, with 2020's event entirely cancelled by the ESA in an attempt to help limit the virus's spread. This will be the first time since the convention's creation there hasn't been an E3 held, to put that in some context. UK-based events like EGX Rezzed also saw a delay imposed, with the indie-focused gathering pushed back to the summer by event organiser and EuroGamer owner ReedPOP, and the Video Game BAFTA awards show was changed to a streamed event.

02. Silent Hills

It's a rumour, take it with a barrel full of salt, but the whispers are pointing to not one, but two brand new *Silent Hill* games in the works over at Konami. First reported by Rely on Horror, the murmurs say writer/director Keiichiro Toyama, creature designer Masahiro Ito, and composer Akira Yamaoka are working in conjunction with SIE Japan Studio to create a soft reboot of the original *Silent Hill*. And the second game? Well, that's *Silent Hills*, obviously, with Sony allegedly operating as relationship counsellor between Konami and Hideo Kojima in order to get the latter involved with the project. It does all sound too good to be true, but we can live in hope.

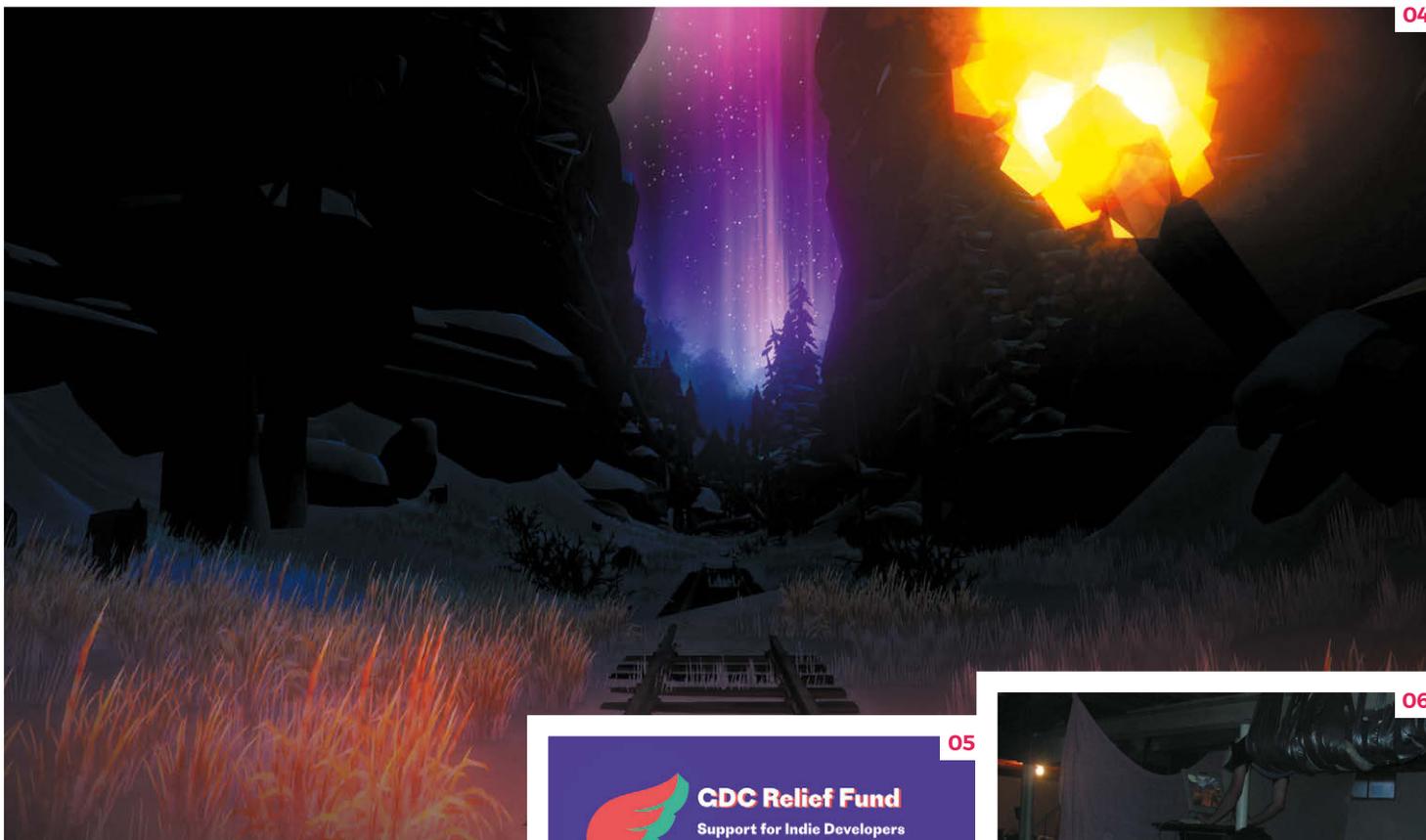
03. Breaking walls

Minecraft is where we build hastily constructed houses only to see them explode shortly afterwards when some cactus-looking thing meanders into it, right? Well, not quite. Adding to that feeling of 'not quite' is the recent announcement from Reporters Without Borders, the non-profit focused on press freedom around the world. The organisation has announced The Uncensored Library, a 'safe haven for press freedom' located inside a *Minecraft* server which contains banned books and articles from around the world, free for anyone to read. Check it out here: wfmag.cc/unclib



Super Mario Lego sets announced; joy ensues

Michael Chu, lead writer on *Overwatch*, departs Blizzard after 20 years



04

04. GeForce Maybe Later

Some big publishers have pulled their titles from Nvidia's GeForce Now streaming service, which we took a look at last issue (in short: it's good at times, poor at others). But now it seems some devs – specifically Hinterland Studio – are pulling their games from the service for very interesting reasons. Basically: Nvidia doesn't seem to have asked permission to put some games up, at least going by what Raphael van Lierop, Hinterland founder, wrote on Twitter. "Nvidia didn't ask for our permission to put the game on the platform, so we asked them to remove it. Please take your complaints to them, not us. Devs should control where their games exist." Oopsie.



05

05. Sweet relief

The cancellation of GDC hit some indie devs hard, with unrecoverable costs mounting up and net losses expected. WINGS Interactive, a new label focused on funding games from diverse teams, has teamed up with the likes of Google, Redbeet Interactive, and more to offer the GDC Relief Fund. The fund will offer financial support to developers negatively impacted by GDC's cancellation, as well as providing things like remote pitch meetings and other support services. Have a look here: wfmag.cc/GDCr



06

06. And finally...

If you've been on the internet in the last 15 years or so, you'll likely have seen the image above these words: a *Counter-Strike* LAN party with one particular attendee finding a free spot above everyone else, duct-taped to the ceiling. While the questions of why and how have been discussed by those involved previously, a new documentary – *Internet Legends: Duct-Taped Gamer* – will go into more detail than ever before about this one, pure moment in time, featuring interviews with those involved. You know you want to see more: wfmag.cc/duct

2K to make non-sim NFL games; EA simulates rage



COD streamer Carl Reimer shoots energy drink; banned from Twitch

Worm



The Big Con ↗

Playing as 'awkward teen' Ali, *The Big Con* sees you trying to raise money – by means nefarious or otherwise – in order to save your family's video store from gangsters. Also, it's very nineties. Very nineties. Ostensibly an adventure game with a few skill tests thrown in there – for pickpocketing and the like – this one has a wonderful visual style to it, and a sense of humour to boot. Rad.

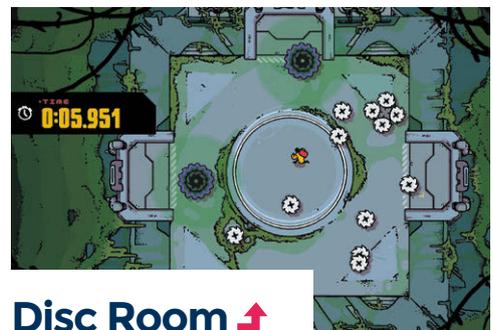


Metal Revolution ↗

Metal Revolution stands out for two reasons: first, it's a striking-looking thing, with bold, cartoonish colours backing up the quick and smooth robot battling action. Second, its striking-ness is being made to suit newcomers to the fighting genre as well as the experts – not a common move in the competitive fighting game scene. That's nice. Will it be any good? Errm... no idea.

When The Past Was Around ↘

Gorgeous hand-drawn art is what draws you into this point-and-click adventure, telling the tale of a woman in her 20s and her attempts to give her lover some kind of form. Said lover is some kind of shadow being, but also an owl-person. Look, quite honestly, it's not one of those that explains incredibly well, so check out the demo over here: wfmag.cc/wtpwa.



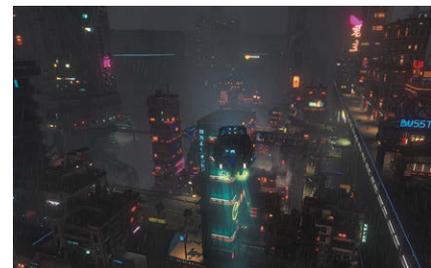
Disc Room ↗

In the year 2089, a giant disc will be found in orbit around Jupiter, apparently, and a team of scientists will have to investigate. What this will result in is what we'd call an action-avoidance romp mixing in puzzles and extreme violence as you navigate rooms aboard the disc, avoiding ever-increasing numbers of spinning blades, and trying to figure out just what in the blue heck is going on. And that's a future historical document for you.



Cloudpunk ↴

Leaning heavily on story and exploration, *Cloudpunk* sees you taking on the role of a delivery driver of the future. It's the stories people tell and the things you discover along the way that are the real hook here, but it doesn't exactly hurt that *Cloudpunk* looks absolutely phenomenal.



Command & Conquer Remastered ↴

Why yes, this is a fair old chunk of space to dedicate to a re-release double-pack (with extras) – but there's a certain allure about *Command & Conquer* that makes it impossible to overlook. Coupled with a genuine sense that EA – and developer Petroglyph – is actually listening to what people want from their re-releases of 1995's *Command & Conquer* and 1996's *Command & Conquer: Red Alert*. In short: we want it to work on modern systems (check), we want higher resolutions (check, up to 4K), we want original developers involved (check, Petroglyph was formed by Westwood Studios refugees), and we want the original music (check,

and some new tracks from original composer Frank Klepacki).

It's all sounding good: the classic original games, their expansions, a plethora of extras, even a re-recorded EVA voiceover by original talent, Kia Huntzinger. But the best thing out of all of this is that it isn't just empty nostalgia – the games still hold up. Right now, it's not clear how much work is being done on balancing things that have been left as-is for 25 years, and maybe things will just be presented as they were, but the opportunity is there for this to be the genuine ultimate *Command & Conquer* collection. Well, of the first two games, at least.

Amnesia: Rebirth ↴

The trope of using amnesia as a storytelling device is well worn, but there are exceptions we can make – one of them is most definitely for when Frictional Games returns with a full sequel to its utterly terrifying *Amnesia* series, in the shape of *Rebirth*. With this being the first time the series' original developer has returned to the franchise since the first game in 2010, hopes are high – and with Frictional popping out the brain-rending *SOMA* a few years ago, it's clear it hasn't lost touch with what makes us frightened. Described as “fresh, but not revolutionary” by Thomas Grip, creative director on the game, it sounds like the studio is sticking with its familiar path of scaring the living bejesus out of anyone who plays its games.



THE BEAUTIFUL GAME | THE STUDIOS REDEFINING FOOTBALL

We meet the developers intent on changing the rules of the football video game

WRITTEN BY
SEAN MARTIN



Football, soccer, kicky-net-goal, whatever you want to call it: whether you enjoy the sport or not, it's impossible to deny the impact it has had on gaming culture. Since the launch of *Atari Football* and *NASL Soccer* in the 70s/80s, football games have grown to become a huge market, with both *FIFA* and *PES* consistently selling in the millions. Inevitably, they owe a huge debt to the sport they're based on.

It often seems like the popularity of these games is based on consistency, sticking to the rules, and channelling an authentic sporting experience.

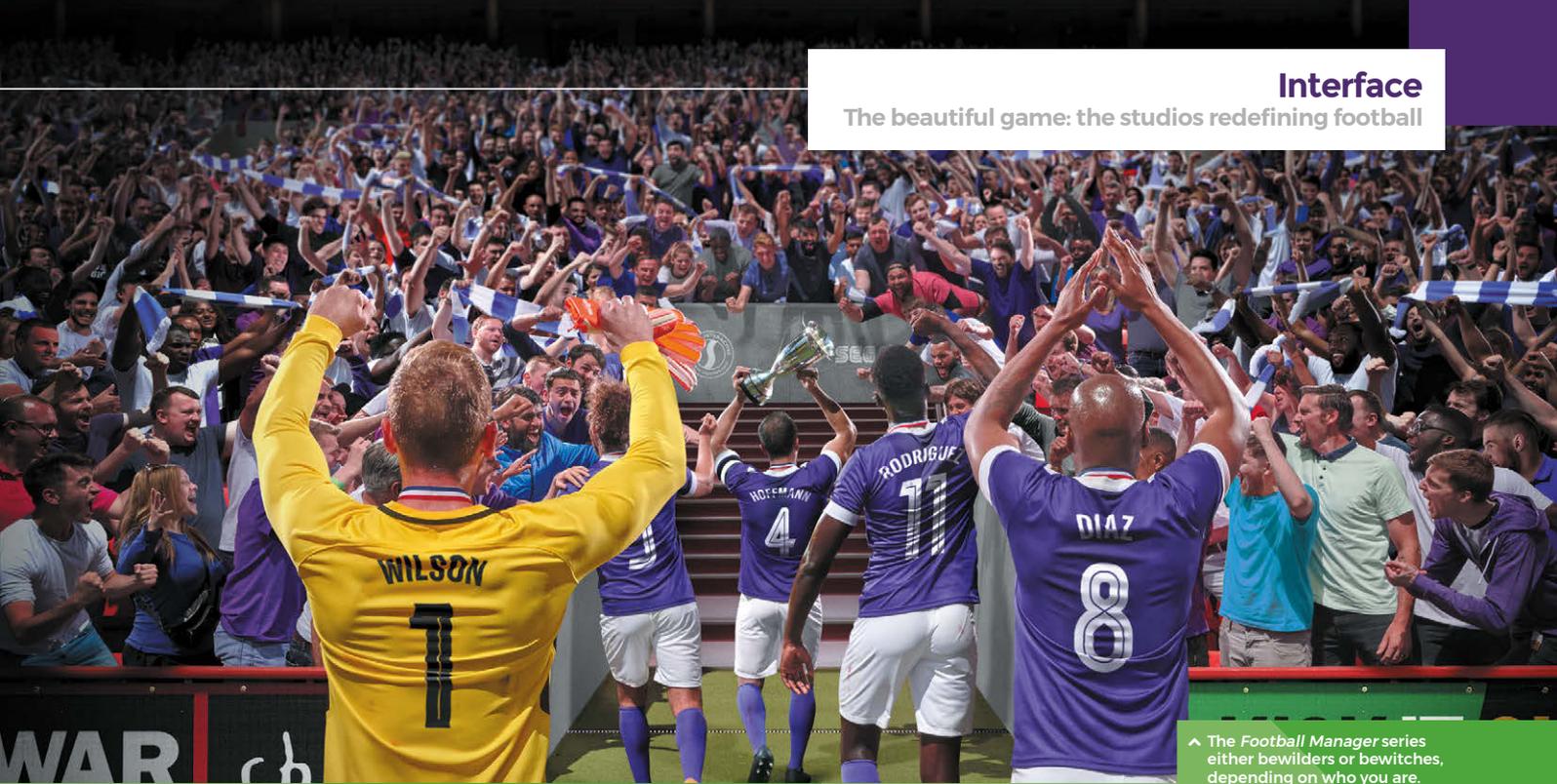
So it's important to ask: when creating a representation of a popular sport, which abides by a set of rules, is there room for innovation? Is there room for different perspectives within the football genre – whether objective, political, or even satirical? All of the following games have achieved this. They've found their own success, while also proving the genre might not be as heavily defined as it seems.

BREAKING THE RULES

For those who don't enjoy football, derision has become an art form of its own, turning the language of the sport

against itself. I can't help but be reminded of Mitchell and Webb's 'Watch the Football!' sketch, with its talk of "Constant! Dizzying! 24-hour! Year-long! Endless football! Every kick of it massively mattering to someone, presumably!" So it wouldn't be unusual to think this perspective had no place within football games, but you'd be wrong.

"I can't stand football. Not interested in the slightest, and I've always been slightly annoyed at how pervasive it is in our society," says Dan Marshall, creator of *Behold the Kickmen*. "I dislike nice pub chats being ruined by people howling



^ The *Football Manager* series either bewilders or bewitches, depending on who you are.

at a television, and I hate feeling like the bad guy because someone's conversation opener was, 'So, who do you support?'"

Part satire, part arcade reimagining, *Behold the Kickmen* pays no attention whatsoever to football's established rules. Offside sees half of the field hilariously close off; the pitch is circular, and scoring goals nets you cash. All of this

"BEHOLD THE KICKMEN PAYS NO ATTENTION TO FOOTBALL'S RULES"

unfolds against a backdrop of nonsensical jeering and amped-up electronica.

"It was nice to be able to just free-wheel some of the rules, especially when they broke the flow of the game," explains Marshall. "I understand about 'throw-ins', but they stopped the game and were boring, so I just made the ball stay in play at all times. I'm aware offside is a hilariously complicated rule, because I've seen people on telly stumble over it for *decades*, like it's the funniest thing in the world. But again, there was no point recreating it properly, and

I hit on the idea of just doing it stupidly, and simply."

Behold the Kickmen excels because it dares to look beyond the established rules of football – like, say, *Sensible Soccer*, it distils the sport's elements into a fast, competitive arcade game. "It was all about making something that, at its core, is quite dull, into something upbeat and enjoyable," reflects

Marshall. "In an hour-and-a-half of real football, the score might be 1-0, which is one interesting event happening in all that time, which arguably isn't brilliant for a video game. So amping up the effects, the music, all that stuff, really helped give the whole thing a shot of adrenaline."

While *Behold the Kickmen* is also a satire at heart, featuring a central narrative which channels every sport story trope imaginable, its mockery of football isn't its greatest success. What *Kickmen* does best is show the benefits of playing fast and loose with the

rules, reimagining established precedents, and most of all, valuing fun over authenticity. As Marshall says, "I always quite liked the idea of playing a football game, but the actual 'football' part of it turns me off."

THE MAGIC OF FOOTBALL

Football's popularity – what Marshall calls its 'pervasiveness' – is difficult to define, even to those who've spent their whole lives playing. It's a subject often covered in the literature surrounding the sport, and one that the game *Football Drama* seeks to channel. "You can refer to two different phenomena with 'the magic of football': →



^ Dan Marshall is part of BAFTA award-winning indie studio Size Five, and developer of *Behold the Kickmen*.



^ Miles Jacobson OBE is director of *Football Manager* studio Sports Interactive.

POETRY IN MOTION

For *Football Drama* co-creator Pietro Polsinelli, the inspiration for his game came from two writers: Uruguay's Eduardo Galeano and Argentina's Osvaldo Soriano. "For Galeano, football in its simplest form – kids running after a ball – is the 'definition of happiness'. The spirit of these writers is to approach football as a sort of dream, like a Borgesian man's tribe ritual, whose sense goes beyond language. These writers convey their attempts at defining football through short stories and episodes, masterfully written and full of evocative power."



Interface

The beautiful game: the studios redefining football

its role in society and what happens in a match," explains *Football Drama's* co-creator, Pietro Polsinelli. "The entire narrative dimension of the game is dedicated to the first aspect; for the second [the matches], I took a minimalistic approach. This means giving a lot of space to the player and their imagination."

In *Football Drama*, you play as the fledgling manager of Calchester Assembled, seeking to guide your club to victory through the turbulent waters of league football.

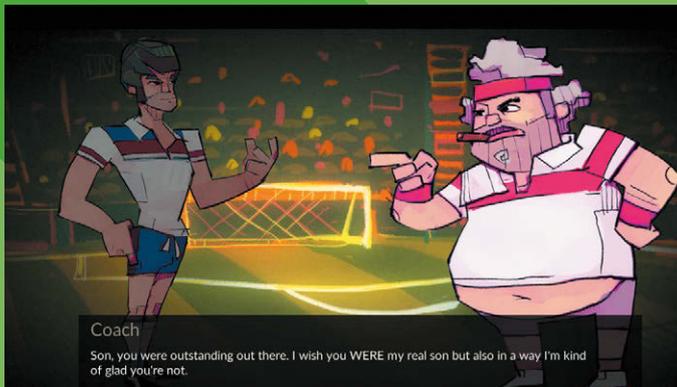
But it's easy to see why *Football Drama* is so divisive: it sheds the direct control of mainstream football games in favour of unpredictability, narrative choice, and a focus on mysticism.

"HUMOUR IS AN ESSENTIAL COMPONENT OF A HEALTHY FOOTBALL PASSION"

Football Drama might also be the first sports game ever to have a karma system, inspired by the *I Ching*, an ancient Chinese text of divination: "The *I Ching* is the second mirror I use in the game (the other being your cat), and it represents your attempts to deal with the flow of life and sport events, that is both inscrutable and always there to be interpreted," says Polsinelli. "I'm trying to make the player think about the illusion of control, which the game of football, with its unpredictability, shows so clearly. This, of course, irritates some players, who expect the game to be some kind of deterministic football manager, but it proudly isn't."

tarot-like playing cards. Each game in *Football Drama* is like a divination – you play your cards hoping the players will listen, you use your limited means of control as best you can, and you hope that if the conditions are correct, and the stars align, magic will happen. It's an idea that channels the mysterious question of what makes good football.

The game also includes elements of satire, presenting a world of corrupt managers and clichéd commentators – and it's here that *Football Drama* is at its most special. While *Behold the Kickmen* uses satire to distance itself from the sport, *Football Drama's* satire is affectionate. "I think that



^ Behold the Kickmen's story parodies the classic sport underdog narrative.

> In Behold the Kickmen, you earn cash each time you 'Do a goal'.

Karma and Kaos are the two choice-based currencies which determine "your coaching/managerial style." Both present their own path to winning the championship, and change the effectiveness of the game's



^ Football Manager grants players a great deal of control over every facet of the managerial experience.

PENALISING PLASTICS

"The planet is something that Tim Breach and Nat Cooke at Sega are just as passionate about as me, so trying to come up with some packaging that was more eco-friendly was an obvious step," explains Miles Jacobson. "While I'd love to take all the credit for it, Tim and Nat did the vast majority of the work on it with their manufacturing partners. Now it's my job to persuade the other entertainment industries to do similar – we're saving 20 tonnes of plastic just from our niche game. Imagine what happens if a home entertainment company comes on board, or a music company, or Xbox or Playstation? It's just the right thing to do."





▲ *Football Drama* pokes fun at the sport, but its intentions are good.

humour is actually an essential component of a healthy football passion," reflects Polsinelli. "Football mixes sacred ritual with playfulness, and it works when the two go together somehow. One of the most praised features of the game by football fans is that they finally found a game where writing and illustration are filled with that profane spirit that is a part of football."

POLITICS IN PRACTICE

Football Manager has become a beloved series since its 2004 launch, as well as one of the most successful football sims ever. Throughout that time, developer Sports Interactive has also championed the in-game representation of many political movements within the sport itself. "We made a decision more than 20 years ago to put the Kick It Out logo in the game just to help promote their work. They're a great organisation who've worked tirelessly to kick racism – and now other

forms of discrimination too – out of football," says Miles Jacobson, studio director of Sports Interactive. "We've also supported many other causes – some financially, like War Child, who get a donation for each PC game we sell, some with in-game adverts, like GamesAid, SpecialEffect, and four other charities each year. The ads in the game are also in-situ – they appear on ad hoardings around the side of the pitch the same way they do at football grounds, but are clickable so people can find out more info – and lots of people do exactly that, which is great."

At a time when some major developers appear to be actively trying to disengage with political issues, Sports Interactive's record of grappling with these topics and creating representation in their games is refreshing to see. Depicting real-world issues that affect the sport also feeds into the game's sim status. "It is about creating a realistic world – the suspension

of disbelief is essential to everything that we do, but I do think it's educational too," Jacobson explains. "Brexit was added because it's part of the real world, and will affect football in the UK – so I spent time working out all the possibilities of what could happen football-wise post-Brexit and tried to work out the percentage chances, consulting with some academics and politicians to ensure it seemed sane. In the real football world, most of these scenarios are still on the table, and it's still a subject I find myself talking about far too often!"

It can be easy for games to claim a disconnect with the real world, but *Football Manager* shows us how a sports game can act as a positive influence on the sporting culture on which it's based – whether through charity, representation, or by educating players about the issues and injustices facing the sport. But such an approach can also spark controversy, Jacobson points out. "I've now got used to getting hate mail,

tweets, death threats... Threats can come from anything – even people losing matches. Players coming out in-game was probably the worst for hate mail. Although I got quite a bit from Brexit, too, mainly from people who didn't like the tiny percentage chance – so small that I doubt anyone saw it in-game – of Northern Ireland splitting from the UK. And lots of love from some people in Scotland who liked the idea of Scotland breaking away because of Brexit – even though as a proud Brit, I don't like the idea of them splitting away at all."

Games like these show the power that perspective plays in changing the genre, whether it's disliking a sport, reimagining the rules to create something wholly fresh, or taking a stand on political issues and committing to realism. When some of the most popular games in the football genre seem slow to innovate, it's exciting to see developers showcase how rich and multifaceted the sport can be. 



THE NOSTALGIA AND DISCOMFORT OF ARCHAIC USER INTERFACES _

A LOOK AT THE APPEAL OF RECREATING YESTERDAY'S
UI SPACES FOR TODAY'S INDIE GAMES _

WRITTEN BY
ANDREAS Inderwildi



^ Nathalie Lawhead,
creator of *EIGTBOK*.



^ Kyle Seeley, creator
of the *Emily is
Away* games.

In the last couple of years, there's been a slew of indie games that use mock desktop environments or other UIs as their main setting. Here, narratives are experienced in purely virtual spaces as we rifle through text files, talk to characters via instant messaging, or navigate a detailed simulacrum of the internet.

Many of those games have an almost archaeological feel to them, since they take inspiration from UIs that have been outdated for years or even decades. They hark back to early text adventures, DOS, Windows 98, and the social media platforms of the 2000s.

"The aesthetic era that I usually draw heavily from is eighties to late nineties computer software," says Nathalie Lawhead, creator of the darkly humorous interactive zine, *Everything is Going to be OK*, which explores virtual spaces full of tortured cartoon animals and glitchy desktop environments. "I know a lot of arguments could be made about 'nostalgia', but I think that's kind of dismissive or missing the point of why there's so much appeal in older software. Early software was just starting to find itself. People were exploring ways of conveying what a UI even is, and how you should interact with one. I feel like with the loss of customisation, experimentation, and the way freeware used to thrive, we

now have these very dead, highly functional corporate environments where the focus is on productivity rather than all the exciting things that are possible on computers."

Capturing an era when things used to be different "is a way of keeping these philosophies alive," says Lawhead.

EIGTBOK's desktop environments are lived-in spaces with idiosyncratic quirks and files that contain poems and diary entries that deal with serious personal issues. As a result, it's easy to feel like an intruder into someone else's private space. "If you give someone a new machine, they start virtually inhabiting it," Lawhead says. "Notes, pictures, files saved in odd places, website history... It's kind of like your virtual home. Software and UI has its own language, too, so you can convey these emotions through the way things glitch, break, function against expectation. UI is really an under-appreciated narrative tool. UI isn't just an interface, it's an actual space."

VIRTUAL HOMES

In Kyle Seeley's *Emily is Away* series, digital spaces are similarly personal. His games recreate the interfaces of the instant messaging and social media sites and programs of the 2000s and ask us to inhabit the role of a high



> In *EIGHTBOK*, apps and files have a life of their own and communicate their anxieties to us.

WHY UI?

"The fact that there was always something pushing us to be more open-minded makes it easier for such work to be embraced," Nathalie Lawhead says. "I remember doing this exact same thing in 2007 and people were much more indignant. Like, why in the world would you even want that in a game? You work on a desktop, why make a game about that?"

schooler chatting with friends and navigating the treacherous waters of teenage relationships. "I chose to mimic AOL Instant Messenger because that's where I grew up," Seeley says. "I know what it's like to talk with a crush on AIM until three in the morning, to juggle multiple conversations, or to go Facebook official with someone."

Seeley calls *Emily is Away* the "social media equivalent of my hometown," to which he'd returned after college prior to creating the first *Emily* game. "My nostalgia for being home, and the awkwardness that came with it, seeped into my work."

Playing the *Emily is Away* games is an intensely nostalgic experience for anyone who was a teenager during the 2000s, but its nostalgia comes with a sting. "Having grown up, we can relate to the teenage main character, but we aren't them in 2006," says Seeley. "And ultimately, that rift between the person we are now and the person we were in high school comes to a head. The player often chooses to talk to Emily, to reach out and fix things, but the person we were

then doesn't allow us to. The player is forced to rewrite their messages and slowly lose touch with Emily. And the game ends on that note of discomfort, of not being the person we wish we could have been."

A BROKEN CYBER UTOPIA

Hypnospace Outlaw places you in the role of an Enforcer in HypnOS, a kind of alternate version of the internet set in a fictional 1999. It's your

job to surf the quirky, garish homepages of HypnOS users and report abuses. Its creator Jay Tholen shares Lawhead's appreciation for early digital spaces: "I love

the short window of time when people do weird, interesting things with new technologies that are just hitting the mainstream."

Tholen's HypnOS is a janky, glitchy, sometimes broken space that bristles with creativity. It's a callback, he says, to a past that never existed: "People were being sold this dream of a digital global village with ads containing 3D ▶

"'NOSTALGIA' IS DISMISSIVE OF WHY THERE'S APPEAL IN OLDER SOFTWARE"



▲ Jay Tholen, creator of *Hypnospace Outlaw*.

Interface

The nostalgia and discomfort of archaic user interfaces



▲ In *Emily is Away Too*, characters message us links to sites like YouToob or Facedook, which open directly in our browser.

MY LITTLE PONY

"The way *Pony Island* uses mechanics outside the main application, like when you receive a fake Steam message, definitely inspired some of that stuff in my own games," says Kyle Seely. "Chat logs exporting to your desktop, using your browser to view retro YouToob, and so on. And there's actually a *Hypnospace Outlaw* Easter egg in *Emily is Away Too*! You can talk to hypnobot and create your very own hypnopage!"



▲ Daniel Mullins, creator of *Pony Island*.

avatars flying around together across psychedelic landscapes. There was a very real sense that the internet was going to change communication forever – which did happen – and bring people together by fostering understanding across cultures – not quite there yet. I think once folks finally did buy in and sign on, it became clear that the tech and people pushing it weren't working towards that goal."

Hypnospace Outlaw may have been inspired by this kind of promise, but it's no cyber utopia by any stretch. As a HypnOS Enforcer, we witness first hand how the policies and carelessness of the company often change the digital lives of HypnOS denizens for the worse: "A major story beat in *Hypnospace Outlaw* centres around the impact that seemingly minute changes can have

"TURNING THAT SOFT, FUZZY FEELING INTO SOMETHING UNSAFE"



▲ The digital spaces of *Pony Island* are teeming with demonic spirits trying to manipulate us.

on users and communities," Tholen says. "This sometimes comes in the form of glitchy system updates or security oversights, but pain is more often caused by a blindness to the idea that their users may take the online space they're inhabiting seriously. It may be all they have."

GHOST IN THE MACHINE

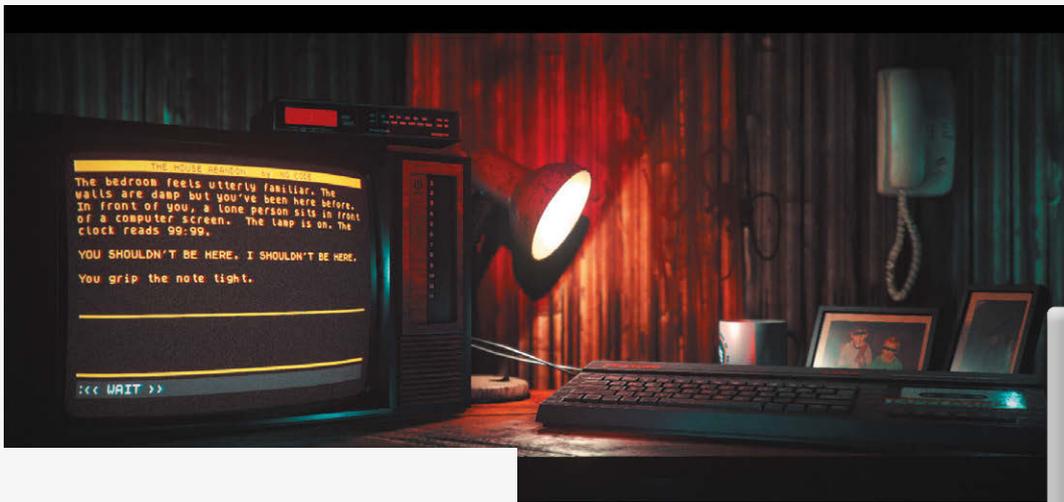
If games like *EIGTBOK*, *Emily is Away*, or *Hypnospace Outlaw* toy with the darker side of nostalgia, *Pony Island* embraces it completely. Here, nostalgia strays into outright horror as we fight for our soul on the battlefield of an arcade machine built by Satan himself. Creator Daniel Mullins tells me he "wanted to the player to keep 'digging' deeper into the system from the high-level game down to the most basic code of its innards. The desktop is somewhere between those. The aesthetic of the early 1990s seemed to be the result of balancing between being able to represent enough on screen while still appearing antiquated."

It's a balancing act that pays off, as the antiquated nature of the game's interfaces feeds into the creepiness. "Archaic interfaces are good at creating discomfort and even horror due to their unintuitive, sometimes unfriendly, and ambiguous nature," he says.

The deliberate crudeness of its aesthetics only enhances this sense of discomfort. Even though they're rarely associated with each other, there's a strange kind of resonance between its occult iconography and the lo-fi graphics through which they come to life. "A creepy icon or image is only made more disturbing when it is distorted or improperly reproduced in some way, yet still distinguishable," says Mullins. "Lower resolution graphics have a way of being creepy by depicting something disturbing in low fidelity and letting the player mentally fill in the rest. Also, the glitchiness of the digital space in *Pony Island* allows for imagery to appear and disappear quickly, which can also make it more disturbing."

THE HORROR OF NOSTALGIA

The House Abandon, part of the game anthology *Stories Untold*, also dives into the horror of old



technology. But instead of an early nineties arcade machine, we find ourselves sitting in front of an old computer running an eighties text adventure game. At first, it seems to be a game about a protagonist returning to his parents' holiday home, but nostalgia soon takes a sudden left turn when the familiar becomes strange.

"A combination of a love for all things imperfect and a background in graphic design means I get to explore and recreate these things I was in awe of as a child," says Jon McKellan, founder and creative director of No Code. "Not just the angle of text adventure games, but the whole experience; the bright, flickering, multi-coloured loading patterns and noises, the phosphor glow of the CRT – all of these details are erased from today's technology."

Having grown up in a family in which the main source of entertainment was playing video games together, McKellan subverted his childhood nostalgia for "these magical boxes" in *The House Abandon*. "Turning that soft, warm, fuzzy feeling into something that feels distinctly unsafe was really exciting for me to create around," he says. "What was safe was now unsafe, and the memories you uncovered as a result of interacting with these machines weren't as nice as you hoped. We set up the expectation of 'this is a nice safe space to play a game in'. It looks cosy. Then, we pull that space apart."

THE HISTORY OF SOFTWARE

There's a sense of a genre emerging among these explorers of old interfaces. Tholen speaks about the "heap of shared inspirations from when most

▲ In *The House Abandon*, the space outside the screen is non-interactive, but small changes to it reflect what's happening within the text adventure.

of us were in our formative younger years."

"It's only going to grow as more young game devs who were inspired by their digital childhoods start making games," Seeley says. Lawhead feels "like we're all part of a creative discourse. A lot of older work led up to this and laid the groundwork for current titles so that they can even be understood. I would hope that we can build a history of such work."

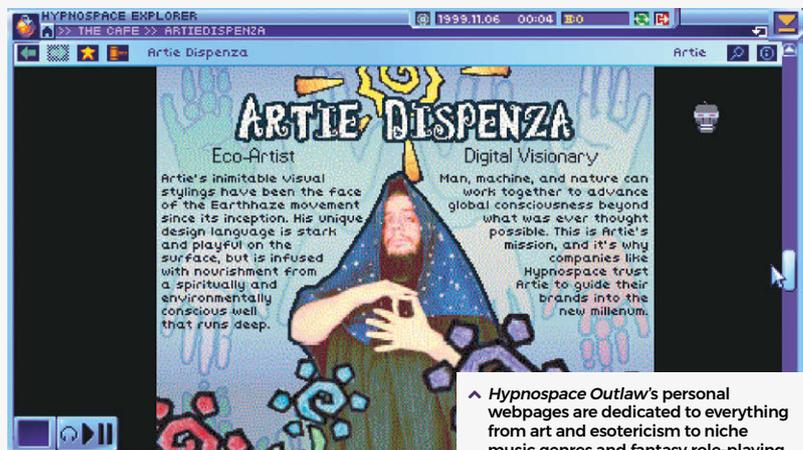
The history not just of video games, but of software in general is a shared concern in all their work. Together, they keep alive the history of how we used to interact with technology, with all the nostalgia and discomfort it entails. [w](#)

ALIEN TECH

Speaking about his experiences working on *Alien: Isolation*, Jon McKellan says that its "lo-fi interfaces looked cool and interesting, but also added to the horror. The display might fail, the program might not load, and you're left with a serious problem. That concept has definitely stuck with me all the way from then, through *Stories Untold* and into our latest game, *Observation*."



▲ Jon McKellan, creative director of No Code and co-creator of *The House Abandon* and *Stories Untold*.



▲ *Hypnospace Outlaw's* personal webpages are dedicated to everything from art and esotericism to niche music genres and fantasy role-playing.

Ceci n'est pas une game



LOTTIE BEVAN

Lottie's a producer and co-founder of award-winning narrative microstudio Weather Factory, best known for *Cultist Simulator*. She's one of the youngest female founders in the industry, a current BAFTA Breakthrough Brit, and founder of Coven Club, a women in games support network. She produces, markets, bizzes and arts, and previously worked on *Fallen London*, *Sunless Sea*, *Zubmariner*, and *Sunless Skies* as producer at Failbetter Games.

“Turn games one way and they glitter with starlight”

*“All art is at once surface and symbol.
Those who go beneath the surface do so at
their peril.
Those who read the symbol do so at their peril.
It is the spectator, and not life, that art
really mirrors.
Diversity of opinion about a work of art shows
that the work is new, complex, and vital.”*

– Oscar Wilde

That last line might comfort developers whose games have not been met with ‘overwhelmingly positive’ reviews. It’s the penultimate line that gets me,

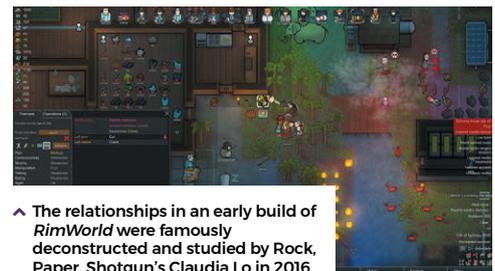
though. When you play a game, are you experiencing some sort of external experience – is *Detroit: Become Human* a game ‘about’ human nature and second-class citizens, or is it ‘about’ provoking the player to think about these things? Do we consume external narratives when we play games – is there one definitive experience – or do we close a feedback loop, completing an otherwise incomplete piece of art in the act of playing it? The latter is what literary reader-response theory believes. And it’s interesting when you bring that theory to games.

It’s true that games comprise a limited set of themes and mechanics, and can be described in factual terms. But they come alive in players’ hands: the ultimate experience of playing a game cannot exist without a player in the first place. If games need a subjective human to realise themselves, there cannot be objective experiences. No definitive readings or true reactions. There is only art, blended with the human interacting with it at the time.

It’s important to remember this when it comes to critique, particularly when it’s controversial. Take indie designer Tynan Sylvester’s public dust-up with Rock, Paper, Shotgun over a journalist’s interpretation of *RimWorld*’s code. The points the journalist made were valid and formed a coherent, interesting discussion.

But the article was not empirical. The journalist brought an existing belief-system (that simulation games define their own world rules, and force their creators to concretise human experience in rigid ways) to *RimWorld*’s code (an attempt to codify sexuality in C# with a fun player experience at the end). This, to be clear, is great: person-meets-art is what almost all artistic critique *is*. What’s *not* great is when people accept that critique as some sort of empirical, definable truth. *RimWorld* is *not* a game which furthers existing and oppressive gender relationships between men and women or erases bisexual men. It can absolutely be discussed in the context of a simulation game’s ethical and essentialist restructuring of infinitely complex human reality, but it doesn’t change *RimWorld* itself. RPS rightly went beneath the surface and read the symbols. But the resultant piece is a reflection of the spectator, not of life.

Writing a piece in a games magazine about the subjectiveness of pieces in games magazines is ironic. And to be clear, subjectivity doesn’t make reviews, opinion pieces, or other evaluations of games any less meaningful. What it does do is highlight the prismatic quality of what we all do. Turn games one way, and they glitter with starlight. Turn them another, and they reflect the dawn. Whatever you feel about a game – and whatever critics or players or Metacritic or whoever else feels about it – remember games are shiny things, and more often than not what’s said about them doesn’t objectively reflect the game. It reflects the gamer. 🗨



▲ The relationships in an early build of *RimWorld* were famously deconstructed and studied by Rock, Paper, Shotgun’s Claudia Lo in 2016.

Toolbox

The art, theory, and production of video games

28. Design Principles

Howard Scott Warshaw on the programmer brain

30. CityCraft

Using walls and boundaries to define game cities

32. Substitute Soccer

Learn how to code a top-down football game

38. Video game scripts

Game design documents, and how they're used

40. Source Code

Recreate Hyper Sports' shooting minigame

42. Directory

Game AI: useful courses and resources



^ Code a *Sensible Soccer*-style top-down football game with our guide on page 32.



✓ Find out how walls and boundaries can be used in your own game cities. See page 30.

The principles of game design

Analytical, a complex relationship with rules...
Howard takes us inside the mind of the programmer



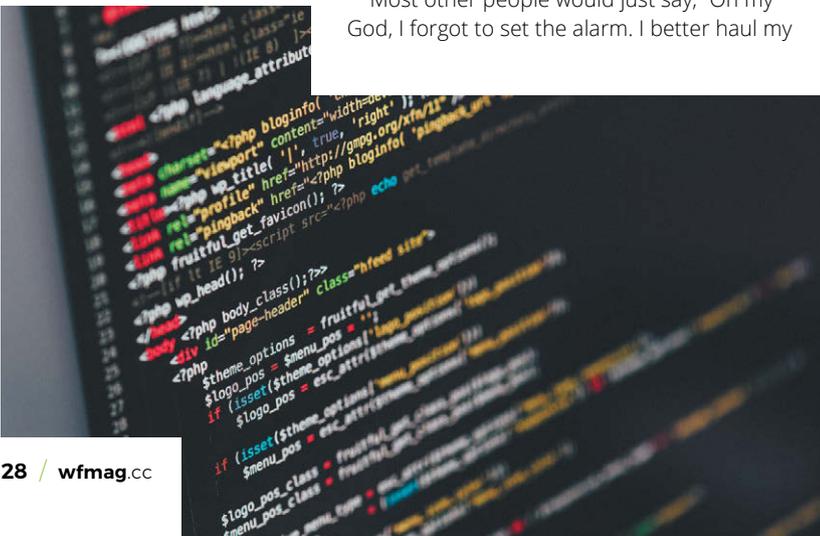
AUTHOR
HOWARD SCOTT WARSHAW

Howard is a video game pioneer who authored several of Atari's most famous and infamous titles. onceuponatari.com

▼ Tea? Coffee? Water?
Whatever it is: use a
quality mug.



▼ Pictured: inside the mind
of a programmer. Maybe.



A computer program running on a computer is a system. Programmers who create, modify, and repair them are systems engineers. Systems engineers analyse systems to determine their capabilities and limitations. Then they figure out how to best achieve goals within those systems. This is something all human beings do. We are all systems engineers, it's just that most of us don't do it explicitly with specialised language.

For example, a systems engineer might wake up and say, "Hmm, I discovered a bug in my set-up procedure last night, resulting in a current circumstance outside the range of acceptable outcomes. To recover to a nominal situation, I must derive and execute an optimal (or near-optimal) strategy for transporting myself from my present location to a predesignated alternate location in minimal time. This strategy must assimilate and accommodate the current state of all relevant environmental parameters."

Most other people would just say, "Oh my God, I forgot to set the alarm. I better haul my

ass to work. Oh crap – traffic!" Same situation. Same reaction. Different language.

Now let's look at programmers. Programmers are more than systems engineers; they are a rapidly growing segment of the population. We are increasingly likely to run into programmers in our day-to-day lives, so it's probably worthwhile getting to understand them a bit more.

Computer programming is a complex and conceptual undertaking which happens entirely in the mind. Where else can engineers build working systems without building muscle mass? Since computer programs must handle *all* possible inputs, programmers must spend their time trying to think up every perverse thing that could happen, regardless of likelihood. This includes the absurd, the incongruous, and the unseen consequences of complexity. Programmers are responsible for handling it all correctly (even things they don't know about or can't imagine), and every miss is a bug that may reflect poorly on their reputation. Ever try planning for unforeseen events? It can be very stressful. This is a tough place to live. Keeping your head in this place too long can lead to a phenomenon known as Programmer Brain, and it isn't pretty.

When you only think in terms of exceptions, you can lose touch with ordinary happenings and the regular flow of life. Many programmers fall into this trap. They're so busy trying to protect the software from extraordinary cases that they lose their sense of proportion. Most people tend to focus on the usual stuff, figuring they'll deal with the exceptions as they arise. This works pretty well, since common cases are the vast majority of life. It's not so good for programmers



though, because they have to deal with the exceptions before they happen.

Programmers tend to be people who like order and love exceptions. This can make them annoying, because they love to point out the one obscure counter-example to your otherwise sound explanation. Who doesn't love a good round of "Spot the logical flaw"? Many non-programmers, that's who.

SECOND-GUESSING

Programmers have an interesting relationship with rules. They both respect and can't stand rules. Specifically, they like sensible rules and will not tolerate nonsensical rules. Of course, nonsensical rules are defined as rules they don't feel are worth listening to. They don't phrase it that way; they say those rules are outdated, illogical, contrary to stated policy, or just stupid. But once you translate the nerd-speak, that's the bottom line.

Programmers are also looking for the ultimate elegant solution. Not only must they handle every possibility, but they also want to be as efficient as possible. To this end, software people are constantly spouting, critiquing, and reforming ideas before settling on an approach. They're chasing the question: "Is this the best solution?" (Fun fact: the efficiency of a programming effort is typically measured by how nicely it solves the problem, not by how long it takes to create the solution. Time to creation may be ignored entirely – by the programmer, not their manager.)

Once a path is finally chosen, the second-guessing and reworking begins. Warning: doing this for extended periods may have side effects. The impact ranges from the comic to the tragic. My wife sums it up nicely with: The Cabinet Story...

"Programmers tend to like order and love exceptions"

For her, it was simple. "Sweetheart, could you please move the office cabinet to the den?"

"Absolutely." Done.

Twenty-five minutes later, she comes by the office. I'm striking a pensive pose, staring intently at the open (but as yet untouched) cabinet.

"What are you doing?" she asks.

"I'm thinking about the best way to do this."

"You take the books and games out, move the cabinet, then put them back in. That's the best way to do it."

"But where do I put them in the meantime? Should I move them to the den as I take them out or move them later? Do we really need all this stuff, maybe this is the time to dump

some of it? Oh, there's backgammon, we used to play backgammon, that was a lot of fun. Or maybe I can carry it without having to remove everything. What's the

minimum stuff I can take out and still move the cabinet? Maybe we should get a handcart, that will save time on loading and unloading."

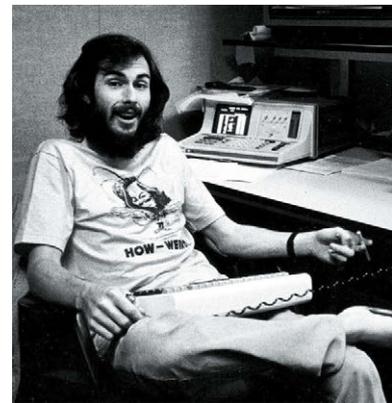
"Save time? Seriously? It's been half an hour.

You could have done it already!"

"Look, planning time is just overhead. I'm trying to optimise here!"

She disappears down the hallway, but her sigh of exasperation lingers. In moments like this, I'm glad she loves me, but this is what it looks like when Programmer Brain kicks in. It's a place where my approach to the task becomes more important than completing the task.

Remind you of anyone you know? Programmer Brain isn't a side effect, it's a full-on world view, and a fundamental way of being. It becomes the systems analyst's system of analysis... for better or for worse! 🤖



▲ Howard Scott Warshaw, in full Programmer Brain mode at Atari in the early eighties.

Game programmers

Those wacky computer programmers. They do have their quirks and eccentricities. However, there is a subset (or inheriting class) of computer programmers that's even wackier: video game programmers. Game programmers have all the wackiness of regular programmers, but their neurotic engines rev at higher RPMs. That's because regular programmers only have to meet a spec. Game programmers must hit their spec and still clear one more hurdle: the game has to be fun. Want to drive a programmer insane? Give them a specification with subjective criteria for success. When are we done? When a bunch of middle-schoolers say so.

Civic boundaries and city walls

Building walls and defining the edges of an urban space can be a creative process, and full of game design potential



AUTHOR
KONSTANTINOS DIMOPOULOS

Konstantinos Dimopoulos is a game urbanist and designer, combining a PhD in urban planning with video games. He is the author of the forthcoming *Virtual Cities* atlas, designs game cities, and consults on their creation. game-cities.com

Even the most outlandish fictional cities need boundaries. Limits, geographical or societal, have forever separated urban spaces from their hinterlands, and their surrounding 'natural' or rural spaces. Such borders, whether fuzzy and porous or concrete and precise, have delineated all human settlements throughout history, and have served to distinguish life in the city from life in the countryside. In Europe, during the Middle Ages and the Renaissance, in particular, walls were considered such an integral element of what constituted a city that they formed an essential part of its formal definition.

Of course, boundaries don't necessarily have to be as emblematic and impenetrable as the fortified walls of Carcassonne in France, and neither do they have to be the marvels of engineering that protected Constantinople. Urban space can easily be defined by fences, administrative lines, rivers, moats, steep cliffs, ceremonial borders, signs, a greenbelt, or even the last row of its houses.

THE COMMON WALL

The wall, in its myriad historical guises, is the combined result of defensive requirements, engineering might, available materials, architecture, ideology, planning, and geographical location. Obvious defensive functions aside, walls commonly provided shelter from strong winds and the (perceived or real) dangers of wild animal attacks. They also firmly established dominance, served as metaphysical or ceremonial boundaries, and were a physical expression of the city's borders. Embodying civic

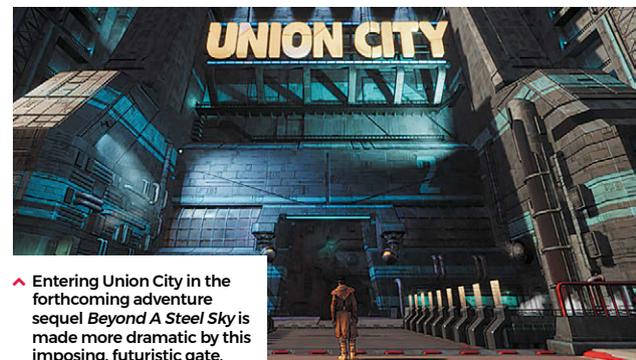
borders turned walls into political instruments of control. Not everyone was allowed passage in or out of a city, and the traffic of goods and people was controlled at guarded gates. City gates were in most cases the locations where tolls were collected, and were the structurally weaker points during a possible siege.

Several types of city walls have existed, ranging from the wooden constructions of the bronze age to the elaborate bulwarks and curtains of Renaissance city walls – all of these can serve both the fiction and the mechanics of a video game world. Walls fulfil the level design demand for boundaries, can act as believable world borders, are excellent centrepieces for epic battles, and can effortlessly provide tense siege or stealth opportunities, as well as a strong civic character.

The realistic fortifications of Whiterun in *Skyrim* are a prime example of walls designed to stage a game's most intriguing battles, while also providing a narrative functioning. Whiterun's walls

The Walls of Civilization

One of the staples of Sid Meier's constantly evolving *Civilization* series is, unsurprisingly, the concept of urban walls. It's a gameplay element with an almost exclusively defensive function which fits in nicely with the abstracted geographies of the series, while also adding a sense of historical authenticity. Aptly, in most versions of *Civilization*, walls lose their defensive bonuses sometime after gunpowder is invented; with the exception of *Civilization 6*, where the option to upgrade walls to the more abstract 'city defences' is given after a tech level is achieved.



^ Entering Union City in the forthcoming adventure sequel *Beyond A Steel Sky* is made more dramatic by this imposing, futuristic gate.



▲ The walls of Croatia's Novigrad, and the imagined demands of their construction, inspired much of the local geography in *The Witcher 3*.

convey local history, showcase its architecture, hint at the relative decline of the city, and describe its class structure. What's more, city walls can obscure the splendours of the city within it, and even offer beautiful views. Getting past them can require a stealth or charisma-focused challenge to sneak past guards, or wearing a disguise, like the one needed to enter Gerudo Town in *The Legend of Zelda: Breath of the Wild*.

Walls are inevitably shaped by available technology and local materials; the walls of Novigrad, Croatia were constructed from stone brought in via river from nearby mountains. And though getting into architectural details is beyond the scope of a two-page article, let me mention some of the defensive wall's key elements. At its simplest, it consists of a sturdy enclosure and gates. Its top should be accessible to the defenders, and it commonly features parapets, gates, towers with arrow slits, and provisions for war machines such as catapults and ballistae. Inspiration can be found in Roman rectangular walls, Baghdad's circular fortifications, medieval castles, and the star-shaped defences of the Renaissance. A fantasy setting could easily incorporate magic, fire or other elemental forces, and exotic materials in its architecture, though all walls have to evolve to counteract new siege techniques.

NOT A WALL

Ancient Egypt, whose security was guaranteed by the Pharaoh, had no need for walls to protect its cities, whereas during the rise of European modernity, major city walls were demolished and turned into ring roads or parks. Despite old definitions, a wall isn't an essential part of urbanism; defence isn't an urban function found throughout history, and stark divisions between city and countryside aren't always required. Legible limits to cities both real and virtual, though, must be defined, and this is why futuristic versions of walls are so popular in science fiction. Energy shields encircled the urban centres



on Frank Herbert's desert planet of *Dune*, and Brutalist walls enclosed Union City in *Beneath a Steel Sky* and its forthcoming sequel. A dome is another relatively common reinterpretation of the wall in sci-fi, but is usually employed more as a shelter against a deadly environment, and less as a means of defence or population control.

Extreme environments do, after all, enforce definite boundaries on the settlements within them, and can introduce a sense of constant danger or complete isolation. The underwater skyscrapers and tunnels of the *BioShock* series, *Mass Effect's* Citadel city, which is surrounded

by the void of deep space, or a domed metropolis on Mars come with their own predefined and absolute limits. Being thrown out of such places would result in death, while the environment itself would constantly threaten to invade the urban space.

In friendlier settings, defining city limits isn't a difficult thing to do. An administrative line, for example, can be drawn anywhere. Restricting access beyond the said line to avoid modelling a whole world is the more taxing problem. Creating the edges of a game world while avoiding invisible walls (whether disguised or not) is an often demanding process. We have to achieve a harmony of fiction and limits. In a contemporary setting, I would, for instance, use train lines, highways with heavy traffic, and waterfronts to limit an open-world city. A steep hill, a waterfall, and a dense forest that could procedurally go on forever would also work, as would unending sand dunes. A besieged city could be blocked by barricades, rubble, and military checkpoints, a horror one by the unending labyrinth encircling it, while fantasy settings could employ imaginative supernatural borders – like heaving walls of the freshly summoned undead. 

“A wall isn't an essential part of urbanism”



▲ Fog, monsters, chasms, and a blocked tunnel all work together to keep players trapped inside the hellish town of *Silent Hill*.

▼ The walls of *Skyrim's* Whiterun feel real and defensible because they're so well researched and realistic.



Inner City Walls

Designers of imaginary cities should not be afraid to use walls, as well as other types of edges, within their cities. The magnificent Forbidden City in Beijing is a fine example of a city-within-a-city separated by a wall, as were several castles dotted around the urban centres of medieval Europe. A succession of walls, mostly concentric, can also be found separating the levels of Tolkien's Gondor, as well as the districts of *Skyrim's* Whiterun. Besides acting as elements organising urban space, city walls also function as subsequent defensive layers.

GOAL!

Make a goal-scoring football game

As seen in *Code the Classics*, here's how to make your own version of *Sensible Soccer*



AUTHORS
EBEN UPTON, DAVID CROOKES,
AND ANDREW GILLET

Released in late 2019, *Code the Classics* shows you how to recreate an array of video game mainstays using Python and Pygame. You can, of course, buy your own copy at wfmag.cc/CtC, but we share one of our favourite games from the book with you here. With that in mind, may we present to you: *Substitute Soccer*. If you follow the instructions in this guide, you'll end up with a great little game of footie. It features both one- and two-player modes, as well as three difficulty settings. Each team has seven players – and as the pitch is larger than the game window, the viewport scrolls on both the X and Y axes.

A SENSIBLE INSPIRATION

Developed by Sensible Software and released in June 1992, *Sensible Soccer* initially appeared on the Amiga and the Atari ST before later being converted to other platforms. It focused mostly on European club football, while some versions had made-up player names and included some fictitious custom football teams. With its top-down perspective and arcade-like flow, the game had great pace and required a high level of skill.

The first thing designer Jon Hare and his team did when creating the game was look at perfecting the controls. They thought about how people would play the game and the type

of controller they'd use, before starting to think about the best way for the action to evolve. "Every game, whether football or otherwise, should be designed around the hardware, so that's what we did well with *Sensible Soccer*," Hare explains. "We designed the controls around the limitations of the Amiga hardware, which was that eight-direction joystick and a single button. All the best games are designed that way, and it helped to get things right."

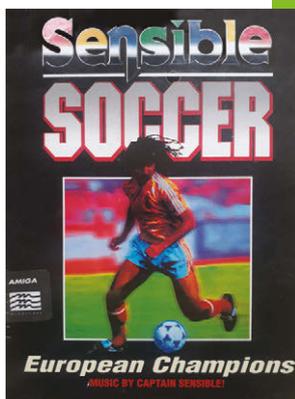
With the controls in place, the developer was then able to deconstruct the real-life game of football so that it could be recreated in pixelated form. Basing the game on an existing sport provided a set of ready-made rules, allowing Sensible Software to concentrate on the skills needed by players and on the best ways of providing them with an adrenaline rush.

It's why some games get away with a pared-back approach: you don't necessarily need leagues, cups, and tournaments, for instance, to make an

entertaining football game. "We concentrated on the actual gameplay when we created *Sensible Soccer*," Hare says. "We didn't go down the path of flashy presentation and the emphasis on style over substance, which we eventually saw when *FIFA International Soccer* was released in 1993. We wanted a game where the player had to chase the ball and use skill to keep it. We certainly didn't want to replicate a televised game. We preferred

"Every game should be designed around the hardware"

▼ If you're going to take influence from somewhere, take it from the best.



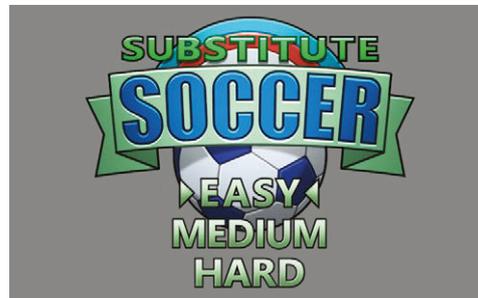


to make players feel like they were on the pitch rather than [being] an uninvolved member of a TV audience.”

SUBSTITUTE SOCCER

To make our tribute to *Sensible Soccer*, we need to bear a few important things in mind first. We'll start by looking at the classes which contain the bulk of the game's code. **Game**, **Ball**, **Player**, and **Goal** are all pretty self-explanatory – although we should note that the **Player** class is used by each of the 14 football players on the pitch, only one or two (depending on the game mode) of which are controlled by a human player at any one time. Whereas in some other games an instance of the **Player** class is the manifestation of the human player in the game, in *Substitute Soccer* it makes more sense to think in terms of a particular team corresponding to a human player, rather than a specific player on the pitch. **Difficulty** is used to store and refer to a number of parameters which are chosen based on the difficulty level. **Controls** deals with control inputs (arrow keys and **SPACE** for player 1, **WSAD** and left **SHIFT** for player 2). **Team** stores an instance of the **Controls** class, which determines the controls for the relevant player – computer-controlled teams use the value **None** here.

The **Team** class also keeps track of the **active_control_player** – the player on the pitch currently being controlled by a human, indicated by an arrow over their head. When they don't have the ball, a human player can switch the ➔



◀ The menu screen enables the selection of number of players and one-player difficulty level.



VISUALISATION QUEST

With so many computer-controlled players running around at once, it can be hard to verify that the AI code is doing what it's supposed to be doing – especially given that players can change roles multiple times within the space of a few seconds. One moment a player might be acting as a goalie, then he might be one of the 'lead' players trying to tackle the ball owner, then he might be marking a player on the opposite team. One way to help confirm that everything is working as it should is to use debug visualisations. Near the top of the code, you'll see a number of constants such as `DEBUG_SHOW_TARGETS`. When set to `True`, `Game.draw` displays a line from each player, showing the position they're currently running towards. Debug visualisations can also be useful for learning how a game works. Try turning on each one – preferably one at a time.

CHALLENGES

- Display a different message from the usual 'GOAL!' if an own goal is scored.
- Try enlarging the game window so you can see more of the pitch at once. Notice how certain aspects of the user interface do not display correctly after this change. How would you fix this?
- The game includes code allowing players to mark players on the opposing team – however, this is currently only enabled for computer-controlled teams. Try enabling this marking behaviour for human teams. Consider how this affects the gameplay and whether you feel the game is better or worse having made this change.
- Give each player a name, and display this above their head by using Pygame Zero's `screen.draw.text` method. You could also generate stats for each player, altering their speed in different circumstances – e.g. when they are running with or without the ball.
- Simulate a full-length match, including swapping ends at half-time. Display a timer on the screen indicating number of minutes played. The timer could advance at, for example, one minute of game time for every five seconds of real time. Make sure that the timer is stopped during the kick-off phase – otherwise a player could run down the clock by refusing to kick-off!

`active_control_player` using the same key they use to kick the ball. We use a class named `MyActor` here, inherited from Pygame Zero's `Actor` class, which is used in part to deal with scrolling and making sure objects are displayed at the correct position on screen. We also use Pygame's `Vector2` class. Instances of this `Vector2` store the X and Y components of a vector, and the class defines a number of useful methods such as `length` and `normalize`, and allows us to subtract one vector from another – which is necessary when we want to work out the position of an object in relation to another, or the distance between two objects.

The `MyActor` class defines the attribute `vpos`, which stores an object's position using a `Vector2`.

Having made the decision to store positions in this way, it's vital to remember to always access or change an object's position via `vpos`, and not via Pygame Zero's usual methods of accessing `Actor` positions, such as `pos` or `x` and `y`. The only time we set those is when it's time to actually draw the object on the screen. We also need to be careful when we want to copy the contents of one `Vector2` to another. As with any class in Python, instances of `Vector2` are reference types. Therefore, copying should be

done like this – `v2 = Vector2(v1)` – rather than this: `v2 = v1`. The latter means that the variable `v2` will refer to the same object as `v1`, so changing one will change the other.

The `Game` class creates and maintains objects for the teams, players, goals, and ball. Its `reset` method is called at the start of the game and after each goal – it recreates the players, ball, and goals, and also decides the initial positions of the players on the pitch. And amongst other things, `Game.update` detects goals being scored and assigns players to mark one another

(including assigning a goalie on hard difficulty). If a team has the ball, it also chooses either one or two (depending on difficulty level) 'lead' players from the other

team, who will try to intercept the ball.

The final part of the code uses a simple state machine system to process interactions with the main menu, and trigger updating and drawing of the current `Game` instance. There are also a number of helper functions to do with ball physics, targeting, and angles. In this game, the numbers 0 to 7 are used for angles, with 0 representing up, 1 up and right, 2 right, and so on. Hence we have our own custom sine and

“When a computer player has the ball, there are two decisions it has to make each frame”





✓ The 'game over' screen shows the winning team and score.



cosine functions which work with those angles, as opposed to degrees and radians.

COUNTING THE COST

When a computer player has the ball, there are two decisions it has to make each frame – which direction to run in, and whether to kick the ball. These decisions are made with the help of the **cost** function. Given a position on the pitch and a team number, it calculates the number representing how good or bad it would be for the ball to move to that position – the lower, the better. The cost value is calculated based on the distance to our own goal (further away is better), the proximity of the position to players on the opposing team, a quadratic equation (don't panic too much!) causing the player to favour the centre of the pitch and their opponents' goal, and a 'handicap' value.

The **cost** function is called in two places. First, when a player with the ball is deciding where to run, **cost** is called five times, each time being past a position indicating where the player would be if they were to move slightly forward in a particular direction – the five directions being straight ahead, left or right 45°, and left or right 90°. **cost**'s optional third parameter, **handicap**,



is used to slightly discourage the player from making turns – this ensures that the player doesn't exhibit unrealistic behaviour such as repeatedly turning left and right within the space of a fraction of a second.

cost is also called when a player is deciding whether to pass the ball to a teammate. A piece of code in **Ball.update** tries to find a suitable player to pass to, but the pass only goes ahead if the cost value for the target player's location is less than the cost value for the current player's location. ➔

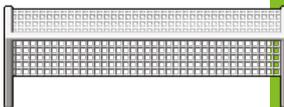
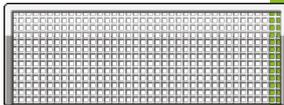
I AM VERY INTELLIGENT

Substitute Soccer's AI has a computer-controlled player deciding their next action each frame, based on a series of conditions. The first is whether anyone currently has the ball. If so, it must be owned by either that player, someone on their team, or someone on the opposite team. Each scenario has its own corresponding behaviour. Alternatively, it might be that no one has the ball. This could be because kick-off is about to take place, in which case no players are allowed to move, other than the player who is about to take the kick-off. Otherwise, all players currently considered 'active' (i.e. are within 400 pixels of the ball on the Y-axis) will attempt to intercept the ball by looking at its trajectory and calculating where to run to have the best chance of obtaining it.



RED team : **BLUE team**

^ The banner used for the scores at the top of the screen.



^ The goalposts for the bottom and top of the pitch.

RUNNING THE GAME

You'll find the full code for *Substitute Soccer* on Wireframe's GitHub – simply head over to wfmag.cc/CTC1-soccer. To get the game running on your system, you'll need to install Pygame Zero – you can find full instructions at wfmag.cc/pgzero.

```
import pgzero, pgzrun, pygame
import math, sys, random
from enum import Enum
from pygame.math import Vector2

if sys.version_info < (3,5):
    print("This game requires at least version
3.5 of Python. Please download
        "it from www.python.org")
    sys.exit()

pgzero_version = [int(s) if s.
isnumeric() else s for s in pgzero.__
version__.split('.')]
if pgzero_version < [1,2]:
    print("This game requires at least version
1.2 of Pygame Zero. You are"
        "using version {pgzero.__
version__}. Please upgrade using the command"
        "'pip install --upgrade pgzero'")
    sys.exit()

WIDTH = 800
HEIGHT = 480
TITLE = "Substitute Soccer"

HALF_WINDOW_W = WIDTH / 2

LEVEL_W = 1000
LEVEL_H = 1400
HALF_LEVEL_W = LEVEL_W // 2
HALF_LEVEL_H = LEVEL_H // 2

HALF_PITCH_W = 442
HALF_PITCH_H = 622

GOAL_WIDTH = 186
GOAL_DEPTH = 20
HALF_GOAL_W = GOAL_WIDTH // 2

PITCH_BOUNDS_X = (HALF_LEVEL_W - HALF_
PITCH_W, HALF_LEVEL_W + HALF_PITCH_W)
PITCH_BOUNDS_Y = (HALF_LEVEL_H - HALF_
PITCH_H, HALF_LEVEL_H + HALF_PITCH_H)

GOAL_BOUNDS_X = (HALF_LEVEL_W - HALF_
GOAL_W, HALF_LEVEL_W + HALF_GOAL_W)
GOAL_BOUNDS_Y = (HALF_LEVEL_H - HALF_
PITCH_H - GOAL_DEPTH,
```

```
        HALF_LEVEL_H + HALF_
PITCH_H + GOAL_DEPTH)

PITCH_RECT = pygame.rect.Rect(PITCH_
BOUNDS_X[0], PITCH_BOUNDS_Y[0],
        HALF_
PITCH_W * 2, HALF_PITCH_H * 2)
GOAL_0_RECT = pygame.rect.Rect(GOAL_
BOUNDS_X[0], GOAL_BOUNDS_Y[0],
        GOAL_
WIDTH, GOAL_DEPTH)
GOAL_1_RECT = pygame.rect.Rect(GOAL_
BOUNDS_X[0], GOAL_BOUNDS_Y[1] - GOAL_DEPTH,
        GOAL_
WIDTH, GOAL_DEPTH)

AI_MIN_X = 78
AI_MAX_X = LEVEL_W - 78
AI_MIN_Y = 98
AI_MAX_Y = LEVEL_H - 98

PLAYER_START_
POS = [(350, 550), (650, 450), (200, 850),
(500, 750), (800, 950),
        (350, 1250), (650, 1150)]

LEAD_DISTANCE_1 = 10
LEAD_DISTANCE_2 = 50

DRIBBLE_DIST_X, DRIBBLE_DIST_Y = 18, 16

PLAYER_DEFAULT_SPEED = 2
CPU_PLAYER_WITH_BALL_BASE_SPEED = 2.6
PLAYER_INTERCEPT_BALL_SPEED = 2.75
LEAD_PLAYER_BASE_SPEED = 2.9
HUMAN_PLAYER_WITH_BALL_SPEED = 3
HUMAN_PLAYER_WITHOUT_BALL_SPEED = 3.3

DEBUG_SHOW_LEADS = False
DEBUG_SHOW_TARGETS = False
DEBUG_SHOW_PEERS = False
DEBUG_SHOW_SHOOT_TARGET = False
DEBUG_SHOW_COSTS = False

class Difficulty:
    def __init__(self, goalie_enabled, second_
lead_enabled, speed_boost,
        holdoff_timer):
        self.goalie_enabled = goalie_enabled
```

```

        self.second_lead_enabled = second_
lead_enabled
        self.speed_boost = speed_boost
        self.holdoff_timer = holdoff_timer

DIFFICULTY = [Difficulty(False, False, 0,
120), Difficulty(False, True, 0.1, 90),
              Difficulty(True, True, 0.2, 60)]

def sin(x):
    return math.sin(x*math.pi/4)

def cos(x):
    return sin(x+2)

def vec_to_angle(vec):
    return int(4 * math.
atan2(vec.x, -vec.y) / math.pi + 8.5) % 8

def angle_to_vec(angle):
    return Vector2(sin(angle), -cos(angle))

def dist_key(pos):
    return lambda p: (p.vpos - pos).length()

def safe_normalise(vec):
    length = vec.length()
    if length == 0:
        return Vector2(0,0), 0
    else:
        return vec.normalize(), length

class MyActor(Actor):
    def __init__(
(self, img, x=0, y=0, anchor=None):
        super().__init__(
(img, (0, 0), anchor=anchor)
        self.vpos = Vector2(x, y)

    def draw(self, offset_x, offset_y):
        self.pos = (self.vpos.x - offset_x, self.
vpos.y - offset_y)
        super().draw()

KICK_STRENGTH = 11.5
DRAG = 0.98

def ball_physics(pos, vel, bounds):
    pos += vel

```

```

        if pos < bounds[0] or pos > bounds[1]:
            pos, vel = pos - vel, -vel

        return pos, vel * DRAG

def steps(distance):
    steps, vel = 0, KICK_STRENGTH

    while distance > 0 and vel > 0.25:
        distance, steps, vel = distance - vel,
steps + 1, vel * DRAG

    return steps

class Goal(MyActor):
    def __init__(self, team):
        x = HALF_LEVEL_W
        y = 0 if team == 0 else LEVEL_H
        super().__init__(
("goal" + str(team), x, y)

        self.team = team

    def active(self):
        return abs(game.ball.vpos.y - self.
vpos.y) < 500

    def targetable(target, source):
        v0, d0 = safe_normalise(target.
vpos - source.vpos)

        if not game.teams[source.team].human():

            for p in game.players:
                v1, d1 = safe_normalise(p.
vpos - source.vpos)

                if p.team != target.
team and d1 > 0 and d1 < d0 and v0*v1 > 0.8:
                    return False

            return target.team == source.
team and d0 > 0 and d0 < 300 and \

```

▼ Player sprites for both teams, for every animation frame and direction.



GET THE FULL CODE

The full Substitute Soccer listing can be found on Github at wfmag.cc/CTC1-soccer 🌐

Hyper-screenplays: why don't we read video game scripts?

It's not quite as simple as writing a script:
Tony explores the realm of game design documents



AUTHOR
TONY JEFFREE

Tony is Wireframe's game writing and narrative design columnist. He's also creative director of Far Few Giants, and you can find his work via tonyjeffree.co.uk or [@tonyjeffree](https://twitter.com/tonyjeffree) on Twitter.

```
... * Jason_Hogger
000 * Hogger_jesseand = true
001 * Hogger_jesseand = true
002 * Hogger_jesseand = true
003 * Hogger_jesseand = true
004 * Hogger_jesseand = true
005 * Hogger_jesseand = true
006 * Hogger_jesseand = true
007 * Hogger_jesseand = true
008 * Hogger_jesseand = true
009 * Hogger_jesseand = true
010 * Hogger_jesseand = true
011 * Hogger_jesseand = true
012 * Hogger_jesseand = true
013 * Hogger_jesseand = true
014 * Hogger_jesseand = true
015 * Hogger_jesseand = true
016 * Hogger_jesseand = true
017 * Hogger_jesseand = true
018 * Hogger_jesseand = true
019 * Hogger_jesseand = true
020 * Hogger_jesseand = true
021 * Hogger_jesseand = true
022 * Hogger_jesseand = true
023 * Hogger_jesseand = true
024 * Hogger_jesseand = true
025 * Hogger_jesseand = true
026 * Hogger_jesseand = true
027 * Hogger_jesseand = true
028 * Hogger_jesseand = true
029 * Hogger_jesseand = true
030 * Hogger_jesseand = true
031 * Hogger_jesseand = true
032 * Hogger_jesseand = true
033 * Hogger_jesseand = true
034 * Hogger_jesseand = true
035 * Hogger_jesseand = true
036 * Hogger_jesseand = true
037 * Hogger_jesseand = true
038 * Hogger_jesseand = true
039 * Hogger_jesseand = true
040 * Hogger_jesseand = true
041 * Hogger_jesseand = true
042 * Hogger_jesseand = true
043 * Hogger_jesseand = true
044 * Hogger_jesseand = true
045 * Hogger_jesseand = true
046 * Hogger_jesseand = true
047 * Hogger_jesseand = true
048 * Hogger_jesseand = true
049 * Hogger_jesseand = true
050 * Hogger_jesseand = true
051 * Hogger_jesseand = true
052 * Hogger_jesseand = true
053 * Hogger_jesseand = true
054 * Hogger_jesseand = true
055 * Hogger_jesseand = true
056 * Hogger_jesseand = true
057 * Hogger_jesseand = true
058 * Hogger_jesseand = true
059 * Hogger_jesseand = true
060 * Hogger_jesseand = true
061 * Hogger_jesseand = true
062 * Hogger_jesseand = true
063 * Hogger_jesseand = true
064 * Hogger_jesseand = true
065 * Hogger_jesseand = true
066 * Hogger_jesseand = true
067 * Hogger_jesseand = true
068 * Hogger_jesseand = true
069 * Hogger_jesseand = true
070 * Hogger_jesseand = true
071 * Hogger_jesseand = true
072 * Hogger_jesseand = true
073 * Hogger_jesseand = true
074 * Hogger_jesseand = true
075 * Hogger_jesseand = true
076 * Hogger_jesseand = true
077 * Hogger_jesseand = true
078 * Hogger_jesseand = true
079 * Hogger_jesseand = true
080 * Hogger_jesseand = true
081 * Hogger_jesseand = true
082 * Hogger_jesseand = true
083 * Hogger_jesseand = true
084 * Hogger_jesseand = true
085 * Hogger_jesseand = true
086 * Hogger_jesseand = true
087 * Hogger_jesseand = true
088 * Hogger_jesseand = true
089 * Hogger_jesseand = true
090 * Hogger_jesseand = true
091 * Hogger_jesseand = true
092 * Hogger_jesseand = true
093 * Hogger_jesseand = true
094 * Hogger_jesseand = true
095 * Hogger_jesseand = true
096 * Hogger_jesseand = true
097 * Hogger_jesseand = true
098 * Hogger_jesseand = true
099 * Hogger_jesseand = true
100 * Hogger_jesseand = true
```

^ This is what a real, machine-readable game script can look like. A lovely easy read - I wonder why these don't often get published separately...

v Film scripts are everywhere, easily accessible, and endlessly studied. Why isn't the same true for games?

Anyone who wants to write a game has at some point asked the following question of a game writer, narrative designer, or teacher: It'd be useful to see what video game scripts should look like, do you have any examples of them?

This is an innocent question, and a very sensible one. It is also cursed. The resounding answer from almost anyone in the know, regardless of how helpful they want to be, is a pained face and an uncomfortable "Well, it's not quite that simple..."

To understand the answer, we must first consider what TV, film, and play scripts really are. Forget for a moment that we sometimes read or study them separately from the end product. They're a prototyping tool. Let me explain:

Let's say I want to make a movie. The most immediate way of doing that is to pick my location, imagine some characters, hire actors and film crew, then go and start filming. All lines would be written as we go, acting would be intuited by the actors and guided by the director. Shots would be

improvised on set. We'd then wander to the next location and repeat. Eventually, a story would start to take place, but we'd likely have to reshoot earlier scenes, much like an author revising their book.

IT'S ABOUT TIME, AND TIME IS MONEY

This would be a prohibitively expensive way to make a film. Imagine the sheer amount of money that'd be wasted on totally avoidable things like plot errors and character inconsistency. So instead we start with a screenplay, which describes the film in words only from start to finish, such that a human reader can imagine the film without the expense of shooting it. They can then assess, edit, and rearrange elements quickly and cheaply until they're absolutely happy with it. Once this written prototype is done, it's much safer to start spending money on equipment and talent.

So why don't we see these 'written prototypes' for games? The answer is, actually, you do. They're called game design documents (GDDs), and they're a human-readable document which describes every aspect of a game in such a way that a reader can imagine the game without the expense of making it. They are, in fact, more potent than screenplays, since they not only describe the story and game mechanics of a project, but also key elements that traditional screenplays don't paper-prototype, such as descriptions of music, art direction, and plans for required technology. Just like a screenplay, everyone on a game team is referring to the GDD for guidance on





◀ Most game writing is done during production, not as a plan beforehand. It therefore doesn't really exist separate from the game.

what's required for their part of the project, and smart developers continually update the GDD, reflecting the new needs of the project over time. Sometimes they get so extensively detailed that we start to call them bibles.

VERY DIFFERENT INDUSTRIES

The real reason these aren't distributed as widely as screenplays is simply that the games industry does not have a social-Darwinist underside, wherein people publish GDDs speculatively online in the hope a game studio picks up their idea. Instead, GDDs are written collaboratively by the key creatives in an existing team, and there's never really a need to expose them to the public. However, a quick search on Google will yield a good number of GDDs from popular games for you to peruse.

But even then, the story sections of GDDs aren't written in any kind of screenplay format. They're usually detailed summaries, more like a film's Wikipedia page than its script, and it's very rare for them to include things like actual lines of dialogue.

The reason for this is that game scripts are also often inherently non-linear. Consider the *Civilization* games, where all of the writing is inside a host of unique, situational interactions with other faction leaders, most of which are absent from each individual playthrough. You can summarise the breadth of each leader, but could you possibly represent this in such a way that you could sit at home and just read the entire script for *Civilization*? Probably not, but we're not finished adding problems yet.

MAN VERSUS MACHINE

Real video game scripts need to be machine-readable. They are not a plan for a human to read, they are a file (or asset) that is part of the actual software itself. The game needs to be able to chop lines up, extract the data about each of them, then use them in-engine. In our *Civilization*

example, each passage would need an ID number of some kind, so that it can be found easily by the computer. It would need to be tagged with the speaker, which we're used to in screenplays, but also with the voice file, which is likely to be obscurely named. It would then need a list of preconditions against which the game can check itself. For example, is this passage about a military surrender? If so, the game must know it can only display this passage during wartime, if the enemy faction's morale score is low, and when the player is winning.

Reading this document would be, frankly, quite horrible, and is unlikely to make any sense unless an inordinate amount of additional work is done in adding plainly written context to every passage. Think of it like this: if a film script is like a cake

recipe, trying to read a game's script is like trying to read a slice of the cake.

The real game script, the one that's actually part of the game, is written in whatever format the writing and programming teams can agree on, usually a compromise that's slightly too technical for the writers, and a bit too human for the game engine. What this actually looks like depends entirely on the type of project and the individuals responsible for it.

Sometimes you'll use a programming language like Ink, Yarn, or Harlowe, and your writers will have to learn it. Other times, the writers will simply write lines in a massive spreadsheet, and the programmers will have to find a way to parse that into the game. Large developers sometimes use specialist software that bridges this gap, such as articy:draft.

So if you're going to write a game, look at GDDs rather than searching for screenplays of video games. Then consider what machine-readable format is best for your unique project, and start the real writing in that. Which machine-readable format is best for what type of game? That'll have to wait for a future column, but it'll probably begin with "Well, it's not quite that simple..." 🙄

“Real video game scripts need to be machine-readable”

Exceptions

“But what about *The Last of Us*?!” I hear you cry. Well... OK. There is a small minority of (usually) very high-profile games in which the story and line-delivery are filmic enough that it can be worthwhile to prototype the story like a screenplay, purely as a human-readable script. It's rare, but it happens. The recently released script book for Insomniac's *Spider-Man* game is a great example of this.

♥ In *The Last of Us*'s script 'prototype', Neil Druckmann used blue text to differentiate gameplay descriptions from standard action.

```

SARAH
Uh... what was that?

Sarah turns to the window. She can see the Austin skyline lit up against the night sky. A plume of smoke rises out of the street.

Next to the smoke a giant explosion erupts. Several of the buildings go dark.

A moment later the window rattles from the after shock.

SARAH (CONT'D)
Oh god.
(beat - panicked)
Dad?

Sarah leaves Joel's bedroom

[GAMEPLAY] Sarah runs downstairs. Through the front window she sees a row of police cars speeding and wailing down the street.

SARAH (CONT'D)
What is going on?

Towards the back of the house, the door to the guest room is propped open. Sarah shows the door open and heads inside.

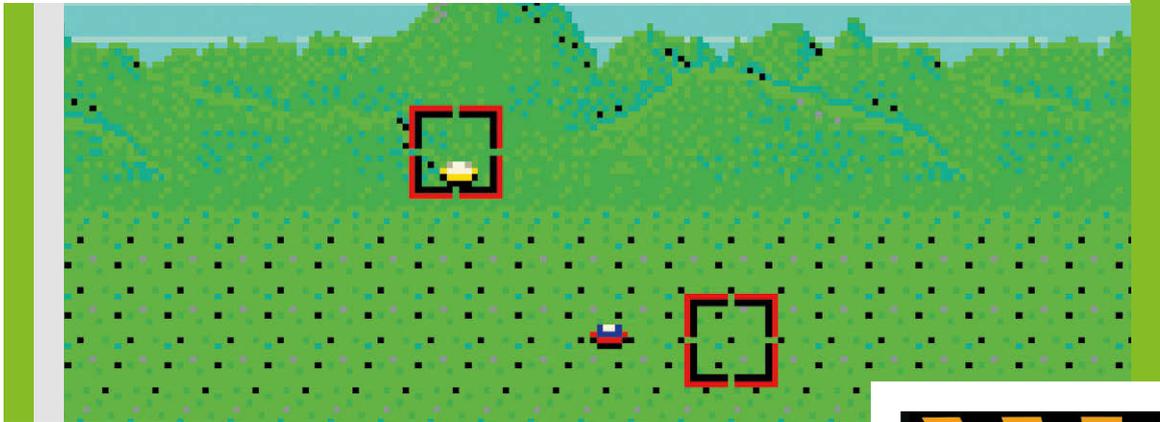
INT. GUEST ROOM - NIGHT

Sarah stands in the doorway to the guest room. On the opposite end of the room is a glass door leading to the backyard. The glass door is open and the blinds gently shift in the wind.

She starts towards the door when Joel rushes into the house and slams the door shut behind him. He breathes heavily.

SARAH
There you are.

Joel spins to face Sarah.
    
```



✓ Like its predecessor, *Track & Field*, *Hyper Sports* had two run buttons and one action button per player.



^ *Hyper Sports*' Japanese release was tied in with the 1984 Summer Olympics.

◀ When the clay targets appear, the player uses the left and right buttons to shoot either the left or right target respectively.



Source Code

Code Hyper Sports' shooting minigame

Gun down the clay pigeons in our re-creation of a classic minigame from Konami's *Hyper Sports*



AUTHOR
MARK VANSTONE

Konami's sequel to its 1983 arcade hit, *Track & Field*, *Hyper Sports* offered seven games – or events – in which up to four players could participate. Skeet shooting was perhaps the most memorable game in the collection, and required just two buttons: fire left and fire right. The display showed two target sights, and each moved up and down to come into line with the next clay disc's trajectory. When the disc was inside the red target square, the player pressed the fire button, and if their timing was correct, the clay disc exploded. Points were awarded for being on target, and every now and then, a parrot flew across the screen, which could be gunned down for a bonus.

To make a skeet shooting game with Pygame Zero, we need a few graphical elements. First, a static background of hills and grass, with two clay disc throwers each side of the screen, and a semicircle where

our shooter stands – this can be displayed first, every time our `draw()` function is called. We can then draw our shooter (created as an Actor) in the centre near the bottom of the screen. The shooter has three images: one central while no keys are pressed, and two for the directions left and right when the player presses the left or right keys. We also need to have two square target sights to the left and right above the shooter, which we can create as Actors.

To make the clay targets, we create an array to hold disc Actor objects. In our `update()` function we can trigger the creation of a new disc based on a random number, and once created, start an animation to move it across the screen in front of the shooter. We can add a shadow to the discs by tracking a path diagonally across the screen so that the shadow appears at the correct Y coordinate regardless of the disc's height – this is a simple way of giving our

game the illusion of depth. While we're in the `update()` function, looping around our disc object list, we can calculate the distance of the disc to the nearest target sight frame, and from that, work out which is the closest.

When we've calculated which disc is closest to the right-hand sight, we want to move the sight towards the disc so that their paths intersect. All we need to do is take the difference of the Y coordinates, divide by two, and apply that offset to the target sight. We also do the same for the left-hand sight. If the correct key (left or right arrows) is pressed at the moment a disc crosses the path of the sight frame, we register a hit and cycle the disc through a sequence of exploding frames. We can keep a score and display this with an overlay graphic so that the player knows how well they've done.

And that's it! You may want to add multiple players and perhaps a parrot bonus, but we'll leave that up to you. 🐦

Download
the code
from GitHub:
[wfmag.cc/
wfmag35](https://wfmag.cc/wfmag35)

Skeet shooting in Python

Here's Mark's code snippet, which creates a skeet shooting game in Python. To get it running on your system, you'll need to install Pygame Zero – you can find full instructions at wfmag.cc/pgzero.

```

from random import randint
gameState = shootTimer = score = 0
shooter = Actor('shooter', center=(400, 450))
frameLeft = Actor('frame', center=(320, 350))
frameRight = Actor('frame', center=(480, 350))
skeets = []
def draw():
    screen.blit("background", (0, 0))
    if gameState == 0:
        for s in range(len(skeets)):
            if skeets[s].x > 0 and skeets[s].x < 800 and skeets[s].
frame < 4:
            skeets[s].draw()
            screen.blit("shadow", (skeets[s].x-20, 400-(skeets[s].
life/2)))
            shooter.draw()
            frameLeft.draw()
            frameRight.draw()
        else:
            screen.draw.text("ROUND OVER", center = (400, 300),
owidth=0.5, ocolor=(255,255,255), color=(0,255,0) , fontsize=80)
            screen.blit("overlay", (0, 0))
            screen.draw.text("SCORE:" +str(score), center = (400, 550),
owidth=0.5, ocolor=(255,255,255), color=(0,0,255) , fontsize=80)
            screen.draw.text("PYGAME ZERO SKEET SHOOT", center = (400, 55),
owidth=0.5, ocolor=(255,255,255), color=(255,0,0) , fontsize=60)
def update():
    global shootTimer, gameState
    if gameState == 0:
        if len(skeets) == 100: gameState = 1
        if randint(0,100) == 1: makeSkeet(700)
        if randint(0,100) == 2: makeSkeet(100)
        if shootTimer == 0:
            shooter.image = "shooter"
        else: shootTimer -= 1
        for s in range(len(skeets)):
            skeets[s].life += 1
            if skeets[s].frame > 0 and skeets[s].frame < 4:
                skeets[s].image = "skeet"+str(skeets[s].frame)
                skeets[s].frame += 1
            if skeets[s].x < 320 and skeets[s].dir == "right":
                skeets[s].distToLeftTarget = 320 - skeets[s].x
            else: skeets[s].distToLeftTarget = 999
            if skeets[s].x > 480 and skeets[s].dir == "left":
                skeets[s].distToRightTarget = skeets[s].x - 480
            else: skeets[s].distToRightTarget = 999
            targetLeft = getNearestSkeetY("left")
            if targetLeft > 0: frameLeft.y += (targetLeft-
frameLeft.y)/2
            targetRight = getNearestSkeetY("right")
            if targetRight > 0: frameRight.y += (targetRight-
frameRight.y)/2

```

```

def on_key_down(key):
    global shootTimer
    if (shootTimer == 0):
        if key.name == "LEFT":
            shooter.image = "shooter_l"
            shootTimer = 10
            checkShot("left")
        if key.name == "RIGHT":
            shooter.image = "shooter_r"
            shootTimer = 10
            checkShot("right")
def makeSkeet(st):
    skeets.append(Actor('skeet', center=(st, 370)))
    s = len(skeets)-1
    skeets[s].frame = 0
    skeets[s].life = 0
    skeets[s].distToLeftTarget = 999
    skeets[s].distToRightTarget = 999
    endpoint = 800
    skeets[s].dir = "right"
    if st > 400:
        endpoint = 0
        skeets[s].dir = "left"
    animate(skeets[len(skeets)-1], duration=3, pos=(endpoint,
randint(-200,250)))
def getNearestSkeetY(leftorright):
    y = 0
    dist = 999
    for s in range(len(skeets)):
        if leftorright == "right":
            if(skeets[s].distToRightTarget < dist):
                dist = skeets[s].distToRightTarget
                y = skeets[s].y
        if leftorright == "left":
            if(skeets[s].distToLeftTarget < dist):
                dist = skeets[s].distToLeftTarget
                y = skeets[s].y
    return y
def checkShot(leftorright):
    global score
    sounds.shot.play()
    for s in range(len(skeets)):
        if leftorright == "right":
            if skeets[s].collidepoint((frameRight.x, frameRight.y))
and skeets[s].frame == 0:
                score += 1000
                skeets[s].frame = 1
        if leftorright == "left":
            if skeets[s].collidepoint((frameLeft.x, frameLeft.y)) and
skeets[s].frame == 0:
                score += 1000
                skeets[s].frame = 1

```



Handy game AI tutorials and resources

If the cunning AI in our Substitute Soccer game left you wanting to learn more about the subject, here are some suggestions

GET INVOLVED

Do you have an online tutorial you'd like to share with readers? Have you created an online resource that other game developers might find useful? Maybe you have a local code club you're keen to promote? If you have something you'd like to see featured in the Directory, get in touch with us at wfmag.cc/hello

▼ *Pac-Man* used simple algorithms to give its ghosts individual movement patterns. There's a bug in the code, though - you can read Don Hodges's geeky explanation at wfmag.cc/pac-bug.



▶ The total beginner's guide to game AI

GameDev.net's primer is a good place to learn about the concepts of artificial intelligence in games, from basic decision-making to pathfinding to finite state machines.

wfmag.cc/beginner-ai

▶ Pac-Man's ghost AI

If you want to learn how to create AI characters with personality in just a few lines of code, then *Pac-Man* is worth studying in detail. GameInternals' deep-dive is an ideal place to start.

wfmag.cc/pac-ghost

▶ Reinforcement Learning in Python

Mauro Comi shows you how to develop an AI Bot capable of learning how to play that old Nokia mainstay, *Snake*, with Python and Pygame.

wfmag.cc/snake-learning

▶ Make an AI-powered Unity game

This Udemy course shows you how to use C# and NavMesh to make an AI spaceship that can explore a three-dimensional planet.

wfmag.cc/ai-ship

▶ AI in Unreal Engine 4

Learn how to use controllers and behaviour trees to create a simple AI character that roams and attacks enemies in Unreal Engine 4.

wfmag.cc/unreal-attack

▶ Pathfinding in Unity

Never heard of A* Search or Dijkstra's algorithm? This Udemy tutorial teaches you these concepts, and how to implement them in a 2D maze.

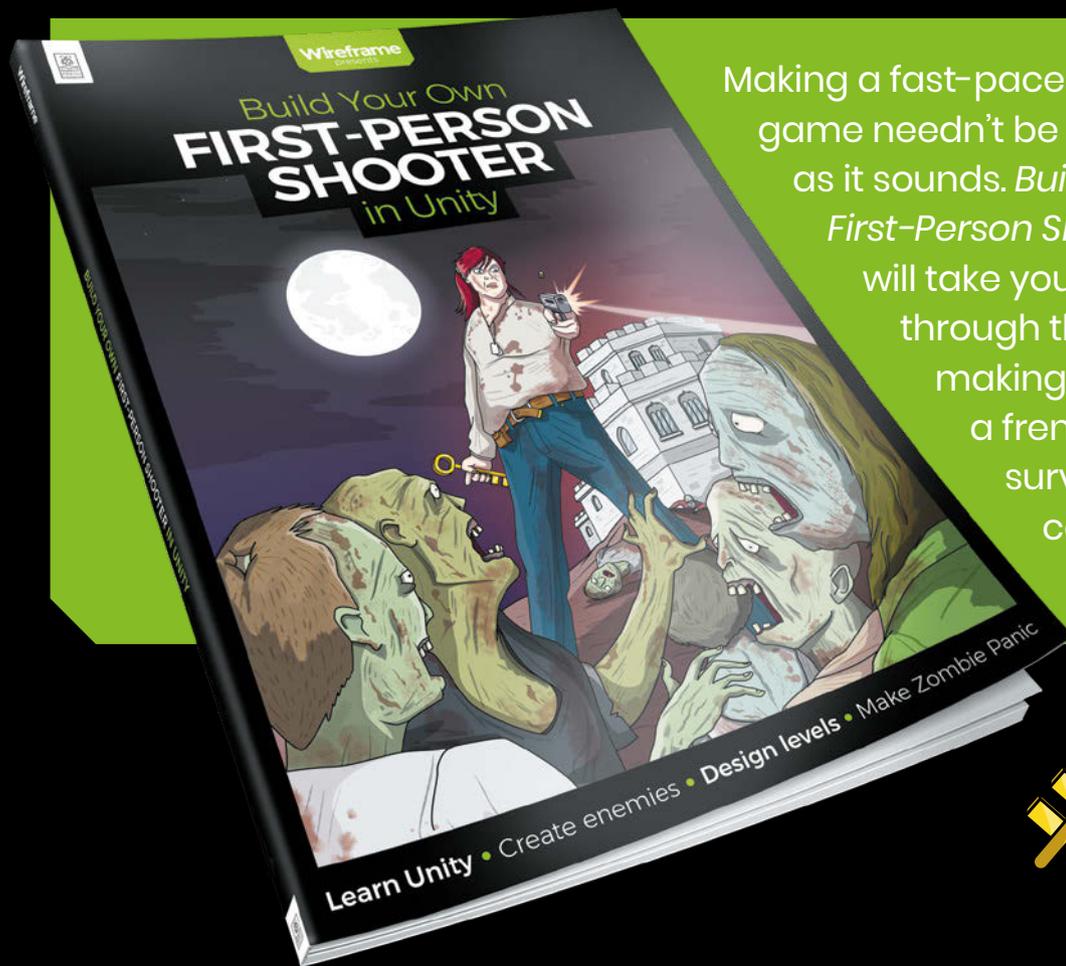
wfmag.cc/unity-path

▶ Advanced AI for games

Another Udemy course, this time about Goal-Oriented Action Planning to create smoother and smarter non-player character movement.

wfmag.cc/ai-goap

Build Your Own FIRST-PERSON SHOOTER in Unity



Making a fast-paced 3D action game needn't be as daunting as it sounds. *Build Your Own First-Person Shooter in Unity* will take you step-by-step through the process of making *Zombie Panic*: a frenetic battle for survival inside a castle heaving with the undead.



IN THE PROCESS, YOU'LL DISCOVER HOW TO:

Set up and use the free software you'll need

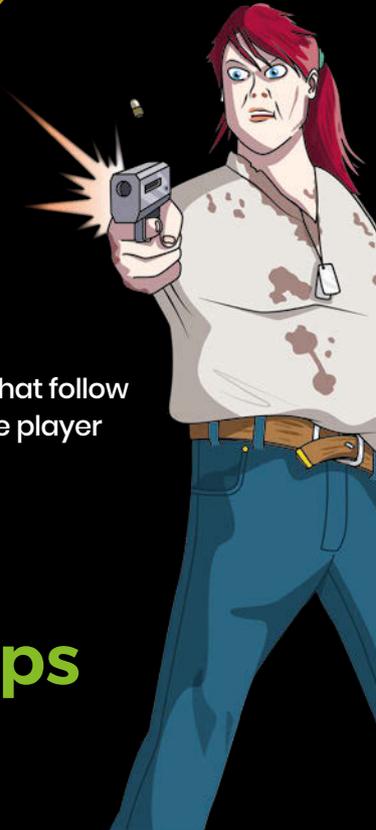
Create and texture 3D character models

Make enemies that follow and attack the player

Design a level with locked doors and keys

Extend your game further, with tips from experts

Available now: wfmag.cc/fps



IN THE BEGINNING

Exploring the earliest video games

How did the games industry really start? With bored mathematicians and a need to sell new-fangled computers

WRITTEN BY
PJ EVANS

The tale of Andy Capp's Tavern in Sunnyvale, California may now be more myth than truth, but we do know some solid facts about what happened in 1972. A few days after a local company called Atari had installed a bizarre-looking upright box next to the pinball tables, the tavern's owner called them to complain the machine was already broken. Distraught engineers quickly arrived to find out what had happened to their precious machine. On opening the cabinet, they discovered the coin holder was overflowing with quarters, jamming the mechanism. *Pong* was a hit.

Although some doubts have been cast over this story's authenticity (including accusations that the holder had been deliberately filled up to impress the landlord), it provides a great opening moment for the history of video games. Here, it's going to be our closing scene. What happened before *Pong* captured the world's imagination?

Let's start with some ground rules: what is a video game? For this particular timeline, let's exclude the electro-mechanical automatons that were popular in the late 19th century onwards – these normally required some human interaction to work. The Mechanical Turk, built in 1770, astounded the high and mighty of Europe by playing chess with seemingly true artificial intelligence. In the mid-1800s, the truth was discovered: an

operator with a complex control system sat inside the device, operating the hands of the 'Turk'. We'll focus on inventions that use true computation to provide entertainment and challenges.

If we're looking at the history of computer games, that means we need computers. The answer to the question 'What was the first computer?' involves a lot of debate and beard-stroking, but we can safely say that nothing qualifies before 1943 and Alan Turing's invention of Colossus – a code-breaking computer based at Bletchley Park, which is often cited as the first of its kind. We'll give an honourable mention to the well-named 'Cathode-ray tube amusement device' patented by Thomas Goldsmith in 1947. Although it used no computation and was never actually built, it was designed for video-based entertainment, but we're not counting it as a) no computer and b) it didn't exist.

THE GAME THAT NEVER WAS

Continuing on the theme of things that never existed, we come to one of the first-ever chess computer programs, Turochamp. It's no surprise that chess features heavily in the early history of computer games,



given the nature of the forefathers including, of course, Alan Turing. Turing and his colleague David Champernowne designed Turochamp, the first complete computer program that could play a game of chess, in 1948. That no computer capable of running the program existed was a minor mishap. Despite attempts to implement the code on the Pilot ACE and Ferranti Mark 1, the only way Turochamp was ever 'run' was by Champernowne playing his wife, manually going through the code step-by-step. The algorithm won. Sadly, the original code is lost but a new version, based on Champernowne's notes, was recreated in 2012 and played Garry Kasparov, losing in 16 moves to the Grand Master.

BERTIE AND NIMROD

Two candidates for the first video game appear in 1950 and 1951. In what appears to be a complete coincidence, both were created for the same purpose, to explain computers to the masses at major exhibitions. In 1950, the Rogers

“Gaming was being used as a bridge between the public and the seemingly impenetrable world of computer science”

Majestic company displayed 'Bertie the Brain' at the Canadian National Exhibition in Toronto. Designed by Dr. Josef Kates, it was designed to show off their latest technology, especially Additron tubes. Bertie could play tic-tac-toe (noughts and crosses) using true computation to decide the next move. It was good, too, with operators often having to dial-down the 'intelligence' so younger players stood a chance. Sadly, the advent of solid state transistors a few years later meant the Additron tube never amounted to much.

Just one year later in 1951, Ferranti premièred 'Nimrod' at the Festival of Britain. This imposing giant grey slab of a machine could play *Nim*, a seemingly simple game that quickly spirals into billions of different mathematical



▲ Actor Danny Kaye beats Bertie the Brain. The operators had secretly set Bertie to the easiest possible skill level to avoid embarrassment.

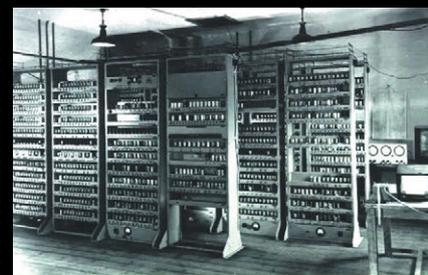
combinations. People flocked to see this modern marvel and try to beat it. Most were no match for the blisteringly fast and clever responses.

In both cases, gaming was, and not for the last time, being used as a bridge between the public and the seemingly impenetrable mathematical world of computer science. Rogers wanted to sell Additron tubes and Ferranti their range of computers. People visiting Nimrod could purchase a book explaining how the computer worked, which included the following passage:

“It may appear that, in trying to make machines play games, we are wasting our time. This is not true as the theory of games is extremely complex and a machine that can play a complex game can also be programmed to carry out very complex practical problems.”

Neither featured full displays as we know them; in both cases, output came in the form of light bulbs. Regardless, we can definitely consider these computer games. Sadly, in an all-too-common footnote of computer history, both companies were swept up by the speed of progress and spiralling costs, blinded from the legacy of what had been created. Both machines were dismantled and are lost to history.

Later in 1951 came a more complex game. Yes, it's chess again, but this time →



THE RETURN OF EDSAC

When it comes to preservation, computing history is littered with disappointment. Bletchley Park's Colossus machines were scrapped for parts and to maintain secrecy. Likewise, expensive components necessitated the destruction of EDSAC; many parts were auctioned to recover costs and fund EDSAC 2. Luckily, you'll soon be able to play *OXO*, complete with rotary phone input, at The National Museum of Computing, Bletchley Park, where volunteers are now in the final stages of building a replica EDSAC.

Interface

In The Beginning: exploring the earliest video games



TEXT GAMES

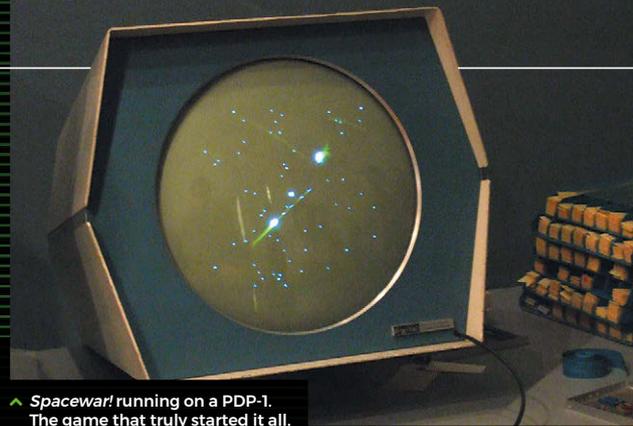
Although computers were becoming more commonplace in universities, expensive CRT displays were out of reach of the average student. In the mid-1960s, text games started to become popular, their call-and-answer format being well-suited to paper-based output. Games like *Hamurabi* (1968) and the first iteration of *Lunar Lander* (1969) were text-based, running on the revolutionary PDP-8. 1971 gave us *The Oregon Trail*, an educational game that would remain popular well into the 1980s. 1975 saw the release of *Adventure* (aka *Colossal Cave*) and a genre was born.

a program was successfully implemented on a Ferranti Mark 1. Loosely based on Turochamp and designed by Ferranti employee Dietrich Prinz, it could solve 'mate in two' problems. Input and output were via a teleprinter as cathode-ray tubes were in their infancy and very expensive.

NOUGHTS, CROSSES, AND TUBES

EDSAC (Electronic Delay Storage Automatic Computer) was built in Cambridge, UK in 1947, and ran its first program in 1949. It is often cited as the first general-purpose stored-program computer. Although many pub arguments have, and will, continue to debate this claim, it was certainly the first reliable machine. The project was funded by Lyons, the Starbucks of their day, who had the foresight to see that these new-fangled computer things could help with the ludicrous logistics and financial calculations involved in baking and delivering ten million meals a week.

In 1952, Alexander Douglas, studying at the University of Cambridge, was preparing a thesis on human-computer interaction. As part of his research, he implemented a noughts and crosses algorithm using EDSAC, making use of a Williams tube (an early form of CRT) which could display the contents of an area of memory as dots on a screen, a concept we still use today. By manipulating this system, Douglas created a noughts and crosses board and in turn a playable video game he called *OXO*. A rotary phone dial was used to enter moves, and the



▲ *Spacewar!* running on a PDP-1. The game that truly started it all.

results were displayed on the tube's 35 × 16 display.

Around the same time, Christopher Strachey was working on the Turing-designed Pilot ACE and decided to implement a simple game of draughts (aka checkers) in order to learn the machine's operations. This also used a rudimentary CRT screen. Whoever ran their code first is unknown. A later improvement to Strachey's code by Arthur Samuel at IBM incorporated machine learning, creating the first primitive AI system.

HIGINBOTHAM TO SERVE

As we've seen, board games were the first games to be implemented, and that's not surprising given the mathematical types designing and using computers. The first sport to make an appearance was a take on the bat-and-ball genre. The delightfully named *Tennis For Two* appeared in 1958 at Brookhaven National Laboratory in Upton, New York, who were gearing up for the annual visitor's day. William Higinbotham, head of instrumentation, was tasked with showing-off their facilities, including the prized Donner Model 30 analogue computer. He decided to adapt a simple test program that drew an arc on an oscilloscope. Working with technician Robert Dvorak, the pair adapted the arc code into a side-on tennis game that could be controlled using two paddles. This may well be the first time a computer game was created solely for entertainment; after all, Bertie and Nimrod were there to sell technology. *Tennis For Two* was a huge hit, with hundreds lining up to play this remarkable creation. An improved version with a larger screen was shown the following year before it was dismantled. In 2008, Brookhaven researchers rebuilt the game to celebrate its 50th anniversary.

(SPACE)WAR IS DECLARED

The 1962 game that arguably kicked off the games industry was *Spacewar!*. This



▲ *Tennis For Two* set up and ready to play at Brookhaven National Laboratory. The first interactive game designed solely for entertainment.

game is not only significant because of its influence, but also for the attitudes it fostered towards software development. *Spacewar!*'s father was Stephen Russell, a member of the Tech Model Railroad Club (TMRC), a group that met at the Massachusetts Institute of Technology (MIT). TMRC had a philosophy of openness and experimentation, and are often described as the first true 'hacker' group. Their policies of openness and collaboration influence the open-source movement to this day. Russell, along with his fellow TMRC members, had become transfixed with MIT's new PDP-1 computer. Although several demonstrations existed to show off the power of the machine, none were truly interactive or demonstrated all its capabilities at once. *Spacewar!* was designed to do just that, a fun experience that pushed the PDP-1 to the limit. The screen showed two ships that could move and fire at each other.

"Douglas created a noughts and crosses board and in turn a playable video game, OXO"

News of the game quickly spread across American universities. Improvements were made, such as hyperspace mode (sound familiar, *Asteroids* fans?), and ports were made to more powerful computers. It was during this period that the game was demonstrated to Nolan Bushnell, although the exact dates are disputed.

BIRTH OF A LEGEND

By the 1970s, reduced component costs and increasing computation power made computer games accessible to the general public for the first time. Nolan Bushnell had formed Syzygy Engineering with Ted Dabney in order to produce a derivative of *Spacewar!* that could be played in penny arcades and bars. They saved money by designing purpose-built logic circuits, an approach that would be common in the early days of arcade machines.

The resulting *Computer Space* (1971), in its futuristic cabinet, sold over 1000



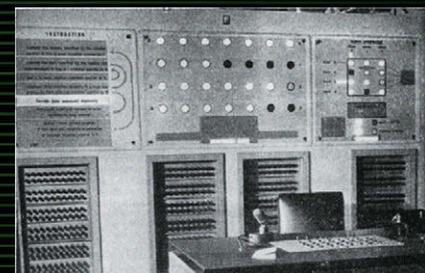
THE 'BROWN BOX'

As computers started to spill out of universities into businesses, and the cost of electronics continued to fall, Ralph Baer continued to develop his idea for playing games on a television. He was convinced that TVs could be used for games as well as watching programmes. By 1967, his 'Brown Box' prototype proved it was feasible. Magnavox licensed the technology and developed it into the 'Odyssey', released in 1972. Baer is often referred to as the 'father of video games'.

units but was not the runaway success Syzygy was hoping for. After a couple of variants were released, Syzygy closed in 1976, shortly after being reborn as Atari. Baer's Magnavox Odyssey had gone on sale shortly after *Computer Space*'s release, and computing as entertainment was now entering the mainstream.

New Atari hire Allan Alcorn was given a 'warm-up' exercise by Bushnell. He was asked to create a two-player table tennis game featuring two paddles and a moving 'ball', viewed from a top-down perspective. The inspiration probably came from the game *Tennis* for the Magnavox Odyssey; indeed, the similarity between the two eventually resulted in Magnavox filing a lawsuit against Atari. Bushnell denied that he was inspired by *Tennis* on the Odyssey, and said that *Pong* was similar to a game on the PDP-1. (The case was eventually settled out of court.)

After designing the custom circuitry and making significant improvements to the original specification in order to make the game harder to play, Alcorn's machine was fitted with a pinball-style coin slot with an improvised holder made from a milk jug. The prototype *Pong* arcade machine was installed in Andy Capp's Tavern in August 1972. A few days later came that fateful call from the bar's owner – and with that, a legend was born. 🎮



✓ A rare photo of Nimrod at the Festival of Britain. The player sat at the desk with the 'board' and Nimrod's moves were shown on the panel of light bulbs.

✓ A publicity shot for *Computer Space*. The radical design was intended to make it stand out from the regular pinball tables.





^ *Inspector Waffles* is built in Unity, but with some custom tools Yann has made to manage and import dialogue.

Interactive

Inspector Waffles

Meet the French developer behind a distinctly hairy point-and-click murder mystery

Are you a solo developer working on a game you want to share with Wireframe? If you'd like to have your project featured in these pages, get in touch with us at wfmag.cc/hello

“If there's one thing I like more than strong milk,” the titular cat detective says at the start of his game, “it's solving mysteries.” A noir murder mystery with its tongue planted firmly in its cheek, *Inspector Waffles* is the work-in-progress of Yann at Goloso Games in France. It contains most of the elements you'd expect from a Chandler-esque thriller: a wealthy industrialist has been murdered in mysterious circumstances, there are numerous suspects with secrets

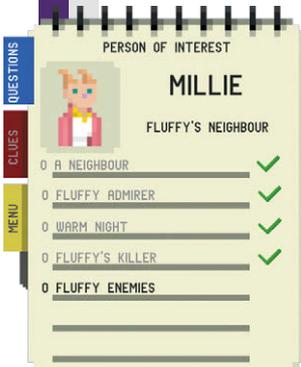
“I started making games six years ago because I was bored at work”

to hide, and there are hints of a connection to a Trumpian billionaire named Maples, whose looming skyscraper dominates the game's early scenes.

Inspector Waffles takes all this and filters it through a point-and-click adventure populated by pixel cats and dogs and laced with gentle humour. Waffles, the scruffy, milk-addicted detective, has a curious mother fixation which results in some amusing phone calls – she also provides a clue or two if you get stuck – while background details are filled with puns: paintings by Meowgritte, a classic novel named *The Great Catsby*.

Yann has clearly enjoyed crafting this pet-noir world, and game design is, he tells us, something of a release from his day job as a software engineer. “I started making games six years ago because I was bored at work,” he says. “I'd been working on a project for six months in a big company, and then, for a stupid reason, the project was cancelled, and all of my work came to nothing. I wanted to make sense of the code I'd developed, and I love video games, so I gave it a try and haven't stopped since.”

The creative spark that led to *Inspector Waffles* began at a game jam three years ago, where the theme was, Yann says, “cats and dogs” (“Without the game jam, I'd probably have never done a game with animals, to be honest,” he confides). Inspired by the point-and-click adventures he played growing up – among them, *Broken Sword* and *Indiana Jones and the Fate of Atlantis* – Yann began thinking about the kind of story he could tell within these parameters. “I thought, ‘I could do a point-and-click game, with detective game mechanics. It would be fun to have a case with a cat who fell from a window and died, because they're supposed to fall on their paws.’ That's how *Inspector Waffles* started.”





◀ "I'm trying to make a feel-good game, with empathy and a strong friendship between a cat and a dog, albeit with a noir atmosphere," Yann tells us.

▼ Yann doesn't consider himself a pixel artist ("I'm trying to improve almost every day," he says), but the game's characters are packed with charm.



▲ In true murder mystery fashion, *Inspector Waffles* begins with a dead body and a rapidly growing list of suspects.



When it comes to designing *Inspector Waffles'* mysteries, meanwhile, Yann has been careful to come up with puzzles that don't leave players scratching their heads – or worse, infuriated when they realise how obscure the solution is.

In the game's first chapter, for example, a door can be opened by improvising a new handle from two random yet logical objects – just the kind of satisfying 'a-ha' moment you need in an adventure game. "I love games like *Monkey Island* and *Day of the Tentacle* – they have this comedic atmosphere where puzzles can be very obscure," Yann says. "And players are OK with things like washing a dirty car to make it rain. *Inspector Waffles* has some comedic aspects too, but I tried to make this game more as a detective movie in which you played the main character. Obscure puzzles won't fit in it, so the game isn't hard if you pay attention to the dialogue and look closely at the backgrounds. And for players who like harder challenges, there will be some story-related secrets to find."

WAFFLY VERSATILE

As for working out the game's structure, Yann says that he starts with the story first, with the puzzles and other elements springing from it. "I always start by writing the whole story, like a movie scenario: a sequence of places to visit, important dialogue, and the mandatory objects players will find. It's vital for me to feel, before anything else, that the story is good and

interesting, at least from my point of view. Then, I design the interrogations, what players should learn about it, and imagine how Waffles will find clues and items in the game. And at this point, when it feels ready, I create a flow chart with every possible interaction, and see if it could work. After that, I finally draw the art and write the rest of the dialogue."

Three years in, Yann says the game's code is roughly 90 percent complete, and playing the demo (which you can sample for yourself at wfmag.cc/waffles), *Inspector Waffles* certainly feels polished. The main things left to do, Yann says, include finishing off some remaining backgrounds and completing the dialogue, but he hopes to have the game done by the end of the year. Fitting *Inspector Waffles* around his working life hasn't always been easy, Yann admits, but his desire to see the finished game has kept him motivated. "I'm working on my games during breaks, evenings, and sometimes weekends – that's why *Inspector Waffles* goes slowly," Yann explains. "It's hard to have two jobs at the same time, to be honest. I'd love to become a full indie someday... *Inspector Waffles* has been in my life for quite a long time now, and even if I love working on it, it's normal to sometimes feel the need to do something else. But, at the same time, I can't wait to see the full story not only on paper, but also in a video game, like I started to imagine it three years ago." 🐾

FILM NOIR

Inevitably, *Inspector Waffles* leans as heavily on film and TV as it does on video games. "Recently, I watched *Mindhunter*, and it inspired me a lot," Yann says. "*Inspector Waffles'* plot was already written when I saw it, but the characters, detectives, and serial killers are incredibly well done."

Back in the realm of games, Rockstar's *L.A. Noire* proved to be a big inspiration. "The investigation mechanics, the atmosphere, the sidekicks... It's one of my favourite games ever. You felt like a real detective, and that's what I'm trying to do with my game."

The buddy-cop aspect of Yann's game, on the other hand, was drawn from an older source: eighties thriller, *Lethal Weapon*. "I created Waffles and Spotty, the sniffer dog, as a comedic duo. Waffles is smart, sarcastic, and doesn't follow the rules. On the other hand, Spotty is loyal and naive – he's a good boy. They have just one thing in common: they both doubt a lot about how good they are at work."



Developer Profile

Tecmo

The firm behind *Ninja Gaiden* and more

Taking in cleaning equipment, yacht sales, expensive lawsuits, and multiple name changes, developer and publisher Tecmo's history is a long and sometimes bewildering one. But while the story behind the Japanese firm is tricky to sum up in just a few hundred words, the mark it's made on gaming is much easier to pin down. Its *Ninja Gaiden* series has remained a mainstay (off and on) since the late eighties; the *Dead or Alive* series of fighting games is still going strong; even its 8-bit American football game, *Tecmo Bowl*, has retained a loyal following well into the 21st century.

Founded by Yoshihito Kakiyama as Tehkan Ltd in 1967, the company first started making video games in 1981 with *Pleiades*, a fairly unremarkable fixed-screen shoot-'em-up in the vein of *Galaxian*. Through the eighties, however, Tehkan – which essentially became Tecmo in 1986 – fostered a hotbed of

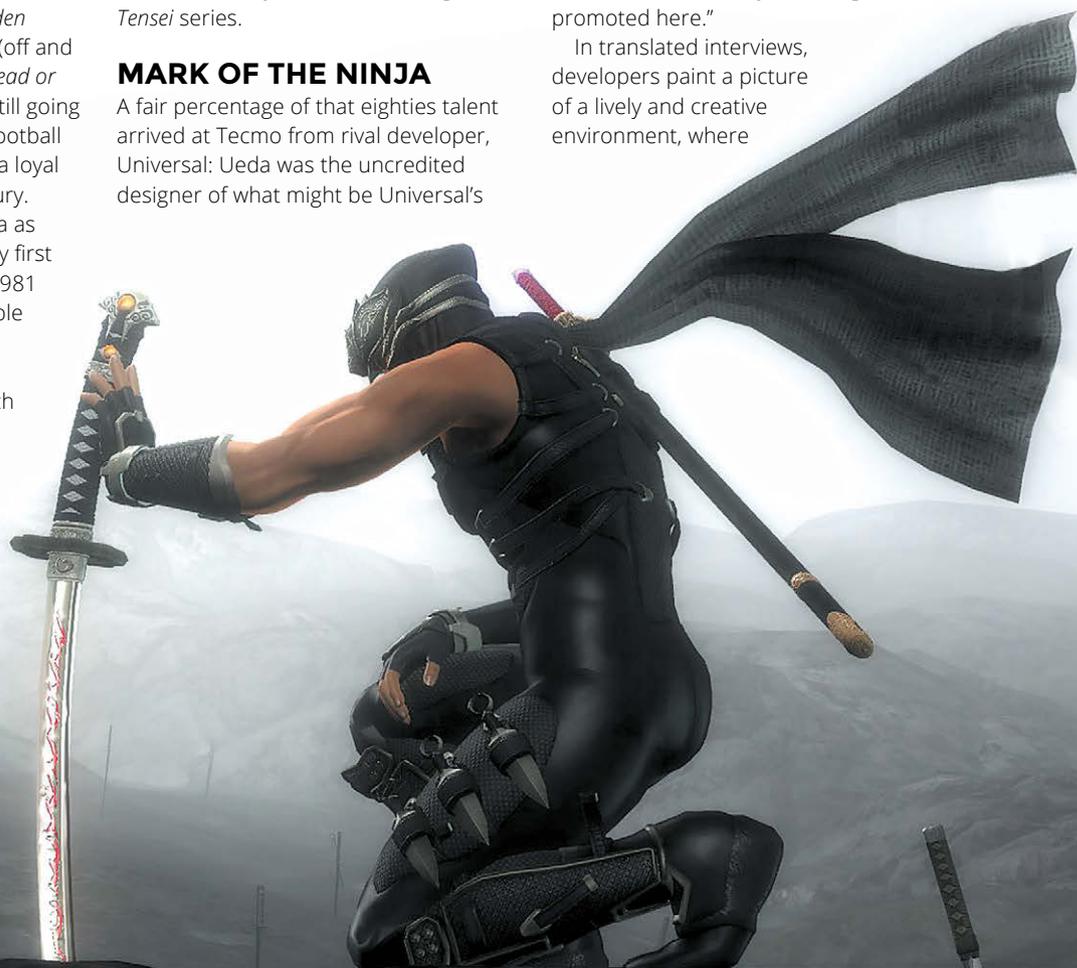
game design talent. Michishito Ishizuka, who programmed Tecmo's hit *Tehkan World Cup*, later went on to co-found developer Escape (later Westone Bit Entertainment), perhaps best known for the *Wonder Boy* series; Kazutoshi Ueda, who oversaw the development of Tecmo's arcade game, *Bomb Jack*, eventually went on to found Atlus, and worked on early entries in the *Megami Tensei* series.

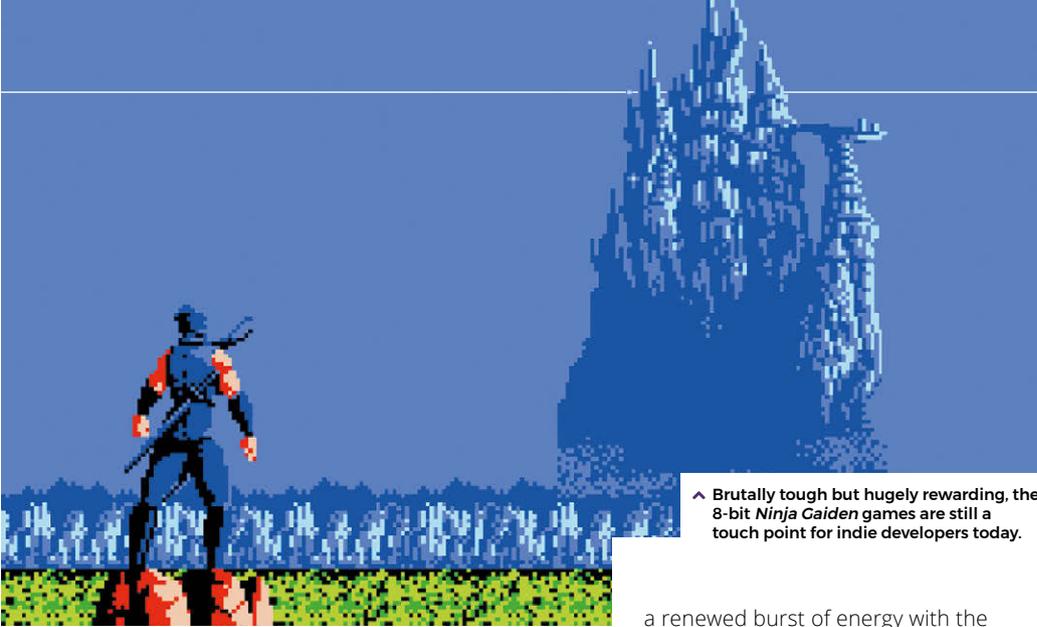
MARK OF THE NINJA

A fair percentage of that eighties talent arrived at Tecmo from rival developer, Universal: Ueda was the uncredited designer of what might be Universal's

best-remembered game, *Mr. Do!* – an unabashed clone of Namco's *Dig Dug* – before he landed at Tecmo. By the middle of the decade, Ueda was joined by around 20 former Universal staff – perhaps as much as half of Tecmo's workforce. According to Michishito Ishizuka, this led to a saying among the company's staff: "If you didn't come from Universal, then you can't get promoted here."

In translated interviews, developers paint a picture of a lively and creative environment, where





▲ Brutally tough but hugely rewarding, the 8-bit *Ninja Gaiden* games are still a touch point for indie developers today.

small teams worked quickly on games over the course of a few months; from this fruitful period came such cult titles as *Bomb Jack*, *Solomon's Key*, *Rygar*, and *Star Force*. *Ninja Gaiden*, meanwhile, got its start when Tecmo boss Yoshihito Kakiyama noted the growing ninja craze in America, and set two teams to work on different projects with the same name. The first was a belt-scrolling arcade game in the vein of *Double Dragon* – it was a violent brawler where the central character could hang from bits of the scenery and kick bad guys clear across screen. The NES version of *Ninja Gaiden*, on the other hand, was completely different, and largely unrelated to the coin-op aside from the look of its athletic hero. A rock-hard action platformer which made innovative use of cutscenes to tell its story, *Ninja Gaiden* was among the finest third-party games ever made for Nintendo's console, and spawned a wealth of sequels and ports.

Tecmo had begun to flag financially by the mid-nineties, before it received

“Tecmo had begun to flag financially by the mid-nineties”

a renewed burst of energy with the founding of Team Ninja, a design group headed up by the now legendary Tomonobu Itagaki. Handed the task of making a rival to Sega's *Virtua Fighter*, Itagaki and his team came up with *Dead or Alive* – a one-on-one brawler whose success, particularly on the PlayStation, helped pull Tecmo back from the financial brink. Team Ninja continued to make some of the best games for Tecmo over the next decade or so, from

numerous *Dead or Alive* sequels to the bloodthirsty *Ninja Gaiden* revival in 2004.

Clouds began to form for Tecmo in the 2000s, though, first with the untimely passing of Kakiyama, its president and co-founder, in 2006. In its own way, the resignation of Itagaki two years later was an even bigger blow; his \$1.4 million lawsuit against Tecmo for alleged unpaid bonuses prompted 300 other Tecmo employees to file a similar high-profile case of their own.

With financial woes once again beckoning, Tecmo eventually merged with another long-standing developer and publisher, Koei, in 2008, which effectively signalled the beginning of



▲ The *Ninja Gaiden* series came back in style in the 2000s, with *Ninja Gaiden Black* (pictured) being a high point.

the end for the firm. Team Ninja and its other divisions still exist under the Koei Tecmo umbrella, but Tecmo essentially went the way of the dodo in 2010. As is so often the case in the games industry, Tecmo's creations – *Ninja Gaiden*, *Dead or Alive*, *Fatal Frame*, and yes, even *Tecmo Bowl* – have far outlived the studio that first made them. 🗨

▼ Tecmo were on the ropes in the mid-nineties. The success of Team Ninja's *Dead or Alive* helped put it back in contention.



Tecmo's Touchdown

Sports games were common enough even in the late eighties, so it's worth reiterating just how unusual it is that *Tecmo Bowl* – and particularly its 1991 sequel, *Tecmo Super Bowl* – still have such a dedicated following. The NES American football game struck an addictive balance between fast action and quasi-realistic simulation, and its timeless quality means that players are still arranging (and streaming) *Tecmo Super Bowl* tournaments almost 30 years later. According to a 2016 article on sports website The Ringer, veteran players who once appeared in the game are still getting fans come up to them and compliment their in-game counterpart's football prowess. "It cracks me up," said former Miami Dolphins linebacker, John Offerdahl. "They have a higher opinion of me than they probably should."



10 Tecmo titans

A selection of the studio's finest moments



01

Bomb Jack

Arcade / various – 1984

Key to this fixed-screen platformer's success was the feel of its controls. Jumping and gently gliding around as the titular hero made avoiding enemies and grabbing bombs a flowing and addictive pastime, and the concept's simplicity meant that its better home computer ports (particularly the ZX Spectrum version) were just as much fun.

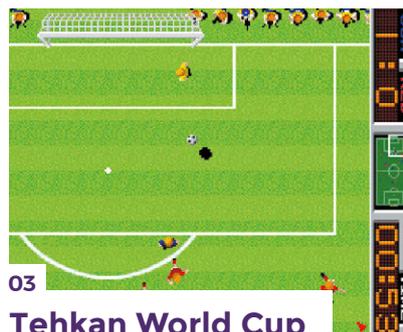


02

Star Force

Arcade / NES / various – 1984

Tehkan's attempt to jump on the Japanese vertically scrolling shooter craze was decent if unremarkable, but it does have a small yet important place in video game history. Hudson Soft handled the MSX and NES ports, and appeared to like it so much that they made *Star Soldier* – a superior, unofficial successor that sparked a long-running series.

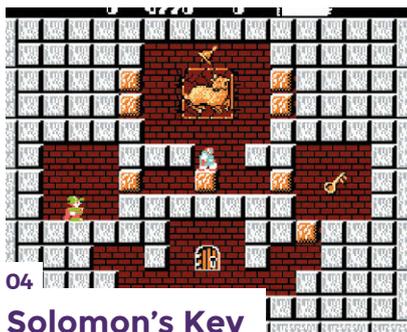


03

Tehkan World Cup

Arcade – 1985

This top-down rendition of the beautiful game really came to life in two-player mode, with the trackball controls and sheer speed of each match – which lasted all of 90 seconds – giving it a sense of sweaty-palmed competitive urgency. Other arcade-style football games, not least the terrific *Sensible Soccer*, offer at least a small debt to this one.



04

Solomon's Key

Arcade / various – 1986

This fixed-screen platformer owes a debt to earlier games like *Lode Runner*, with its central character's ability to create and destroy blocks to form platforms and open up pathways. The need to collect keys to unlock doors gives the game a more cerebral feel, though, and it soon became one of the most widely-ported Tecmo titles of the 1980s.



05

Rygar

NES – 1986

The original arcade game – a run-and-gun affair – was entertaining enough, if a bit lacking in long-term appeal. The NES version really took things up a notch, with more weapons and plenty of exploration. Of particular note is *Rygar's* surreal fantasy theme: it's all giant mutant turtles and ghouls, while its hero's default weapon is a giant, deadly yo-yo.



06

Ninja Gaiden

NES – 1988

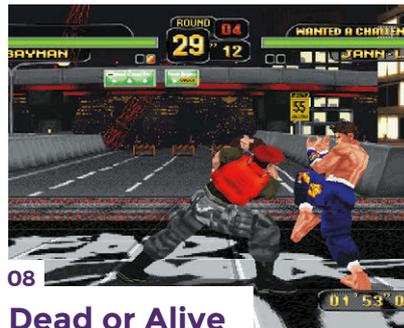
Like *Bomb Jack* four years earlier, it was the precision of the controls that made *Ninja Gaiden* satisfying, even as it murdered you for the five-thousandth time. Jump, cling to a wall, back-flip over to a street sign, then vault to a higher platform, before fatally slashing a villain with your sword... it still feels taut, thrilling, and perfectly judged today.



07 Tecmo Super Bowl

NES – 1991

A sports game that, as we've already seen, is still being played avidly decades later, *Tecmo Super Bowl* displayed an admirable dedication to realism – it featured all 28 NFL teams – but also just enough arcade-style outlandishness to make it enjoyable for those less steeped in the sport's sweaty minutiae. Little wonder it's still popular all these years later.



08 Dead or Alive

Arcade / PlayStation – 1996

The game that helped revitalise a flagging Tecmo, Team Ninja's brawler was a slick and exciting alternative to its competitors, with its grappling moves being a welcome differentiator. The series' puerile dedication to jiggling bits of anatomy has thankfully ebbed at least a bit in more recent years, allowing the weightiness of its combat to come to the fore.



09 Fatal Frame

PS2 / Xbox / PS3 – 2001

Although less celebrated for their work than, say, Capcom or Konami, Tecmo had form in the horror genre, having released the less-well-known *Deception* in the nineties. *Fatal Frame's* nifty ghost-fighting camera mechanic added freshness to the haunted house proceedings, and director Makoto Shibata's use of pace and atmosphere made the game genuinely scary.



10 Ninja Gaiden

Xbox – 2004

Some territories blanched at the dismemberment and gore in Itagaki's reboot, but there was no denying that *Ninja Gaiden* was a thrilling – and suitably tough – return for the series. Some camera issues aside, Ryu Hayabusa's leap to the 3D hack-and-slash format felt like a satisfying and logical progression.



Subscribe today



SAVE
49%

wfmag.cc/subscribe

13 issues for just £20

Subscriber benefits

- > **Free delivery**
Get it fast and for free
- > **Exclusive offers**
Great gifts, offers, and discounts
- > **Great savings**
Save up to 49% compared to stores

Introductory offer

Rolling monthly sub

- > **Low initial cost** (from £4)
- > **Cancel at any time**
- > **Free delivery to your door**
- > **Available worldwide**

Subscribe for 12 months

Receive all 26 issues

£40 (UK) £75 (USA)

£65 (EU) £75 (RoW)

Offers and prices are subject to change at any time



Digital subscriptions from

£1.99



Visit wfmag.cc/subscribe or call **01293 312192** to order

Subscription queries: wireframe@subscriptionhelpline.co.uk

Info

GENRE
Third-person
shooter

FORMAT
PS4 (tested) / PC
/ XBO

DEVELOPER
Rebellion
Developments

PUBLISHER
Rebellion
Developments

PRICE
£39.99

RELEASE
Out now

Zombie Army 4: Dead War

Fight of the living dead

The hordes of *Zombie Army 4: Dead War* personify the horror of the Second World War – in a pulpy, ham-fisted sense, yes, but the metaphor is very much there.

The shuffling undead's repeated grunts and moans give way to a sinking feeling of being overwhelmed as they shamble towards you, endlessly, relentlessly. These are not modern zombies, sprinting around like Olympic athletes and with intelligence beyond what we've traditionally known them for. These are traditional zombies (who are also Nazis): dangerous in numbers, and relentless in their pursuit of human flesh.

Adopting the tone of a camp 1980s horror flick, *Zombie Army 4* is rife with quippy one-liners, ravaging undead crowds, and an obscenely high amount of gore. Its absurdity feels like something you'd catch on the Horror Channel at 3am on a Saturday. Rebellion's game prides itself on how much higher it can take the levels of silliness with each new scenario, and doesn't ever hold back from paying homage to that strange corner of late night/early morning pop culture we've all experienced at some point in our lives.

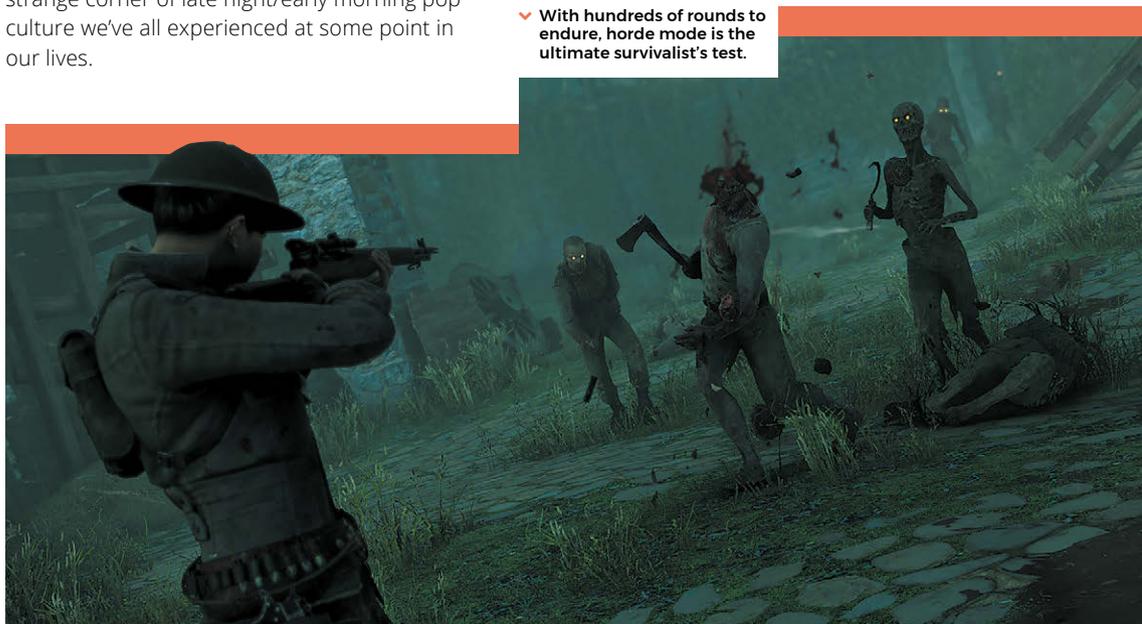
This is backed up by a musical score covering jazz, metal, progressive rock, and more besides. Composer Nick Brewer takes huge inspiration from Italian band Goblin and its soundtrack for 1978's George Romero-directed *Dawn of the Dead*. The funky beats of each track enhance the goofy, old-school atmosphere the game goes for while simultaneously paying homage to the godfather of zombie fiction.

Zombie Army 4's cast of misfit soldiers is diverse, all with their own qualities breathing uniqueness into them. While their personalities aren't clearly expressed in the game itself – rather, relegated to cutscenes and flavour text in menus – their individuality is apparent enough during play thanks to the different stats each one possesses. Some are better snipers, while others are more proficient in melee weapons, for example. These positives and negatives, when put next to each other, encourage co-operation between every member of the team, demanding everyone works together and uses their core strengths.

▼ With hundreds of rounds to endure, horde mode is the ultimate survivalist's test.

HIGHLIGHT

The campaign features a pub crawl of different locations to visit including a volcano, zoo, and even a castle straight out of hell itself. Each chapter introduces new enemies and mechanics to deal with, keeping the story fresh for the entirety of its run.





▲ The ridiculous weapons, such as the blunderbuss, make for some truly incredible action sequences.

This works well alongside the game's deep RPG levelling feature which constantly refines your character as you play. Through pretty standard upgrade trees and skill systems, there's a huge diversity of builds that can be configured to fine-tune a character to your personal play style. Perhaps you're the brute, getting up close and personal with the zombies, soaking up all the aggression; or maybe you're the group's marksman, staying back in safety and picking off key targets through a scope; or you could be the bait, constantly attracting the attention of the shuffling hordes and featuring as a stumbling buffoon of a main course.

On purpose, obviously. Ahem.

This element shines brightest in *Zombie Army 4*'s horde mode, in which players face increasingly difficult enemies in a tough, claustrophobic environment. When it's all about survival, and you're not focused on a specific objective or getting from A to B, the diversity in team ability comes to the forefront, and you really learn how well you all work together – or how styles can clash. Horde mode offers something completely different from the campaign and is, really, much better for players who want a truer co-operative experience.

While it has been designed as a multiplayer game in the most part – it's a multiplayer spin-off of what was originally a single-player only series in *Sniper Elite*, after all – *Zombie Army 4* is completely playable alone. Here it becomes a more frantic affair, as with no one to watch your back you're far more careful about defending certain areas, or hesitant to advance through levels too quickly. Rebellion accounts for this shift by adding an option to scale the number of enemies depending on how many human players are present, so difficulty never spikes drastically higher or lower if your friends can't join you in a session. It also means smaller teams can bring in more zombies than they may be comfortable with for an additional challenge, so it works both ways nicely enough.

“Zombie Army 4 is rife with quippy one-liners”

This flexibility for those playing alone ends up being more of a good thing because of one of *Zombie Army 4*'s main flaws: lag. No matter how stable the connection, every hit on an enemy takes a fraction of a second longer to register than it does when playing solo. It's ultimately inconsequential, especially if you don't even bother playing online, but it is noticeable enough to become a downer on the full experience.

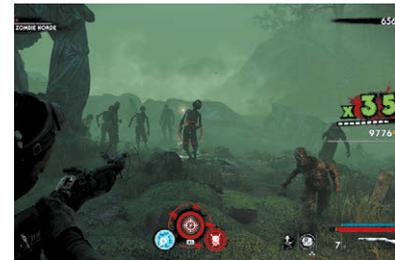
But that in no way should be read as saying *Zombie Army 4* is anything other than a very good game. The intensity of its battles – even with those slow-footed, shambling corpses – is seriously impressive at times, and each encounter with a new horde can leave you gasping for breath. It's one of Rebellion's best shooters to date; a natural evolution from the studio's ever-improving *Sniper Elite* series and – as with the last few *Zombie Army* spin-offs – a theme that fits like a glove.

It's easy to say zombies are overdone – the past 15 years has made this incontrovertible in all facets of entertainment – but that's not to say something can't ever come along and remind us of how to do it properly. Robust shooting mechanics coupled with fine presentation sees *Zombie Army 4* raise the bar for horror-shooters, with its undead Nazi hordes proving, once again, a fine match-up for the Second World War theme. It's imperfect, but one of Rebellion's best. 🗨



▲ No developer does sniping quite like Rebellion.

▼ Your kills build up a combo meter that rewards you with more experience points.



VERDICT

By leaning into its schlockier influences, *Zombie Army 4* comes out stronger than ever.

78%

Rated
Review



HIGHLIGHT

Sometimes a customer orders something you haven't made before, and you need to try to figure out the ingredients. I found myself googling the real-life recipe for certain drinks on my phone, and it was easy to imagine my barista character doing the same thing, discreetly, under the counter.



Jorji
I've just got a call, I've got to go.

Review



^ This cat man wouldn't look out of place tending a farm in a Miyazaki movie.

Coffee Talk

Fine coffee, but not damn fine coffee

Info

GENRE
Visual novel

FORMAT
Switch (tested)
/ XBO / PS4
/ PC

DEVELOPER
Toge
Productions

PUBLISHER
Toge
Productions

PRICE
£11.69

RELEASE
Out now

REVIEWED BY
James O'Connor

VERDICT

A small, pleasant visual novel with low stakes and fun characters that could stand to ask for more from its players.

58%

Coffee Talk understands the appeal of a nice warm drink. You play as a barista, and from your cosy first-person perspective behind the game's bar, you can see the perpetual rain of the city outside. As customers come in and order drinks, it's up to you to figure out the right ingredient combination to satisfy them, and then to listen to the stories of their lives.

Your interactions with the world of *Coffee Talk* are limited. There are no dialogue options, no movement, no real choices beyond which drinks you make and serve. In any given scene, you're only asked to serve a few drinks; most of your time is spent reading, clicking through each new line of dialogue, and sneaking glimpses at your in-game phone to check your growing drink database or change the music. It's wise to pay attention to characters, to try to remember their tastes and what they ask for, but the punishments for getting orders wrong are extremely light. You're mostly there to be a supportive ear for the game's cast of characters.

Coffee Talk is set in a fantastical version of Earth where humans co-exist alongside orcs, elves, werewolves, vampires, cat-people, and other mythical beings, but the dramas that unfold between characters are, at their core, relatable and human. A quarrelling couple might consist of an elf and a succubus, but their problem – that they come from racist families – is only slightly heightened by the fact that one of them is risking their immortality for the relationship.

The script is pleasant enough, but not always particularly engaging, as *Coffee Talk* intentionally

keeps the stakes low for most of its running time. The fate of the world doesn't rest on your barista's shoulders, and while you become privy to some characters' secrets, there's no option to do anything with them. There's little incentive to serve your customers anything other than what they asked for, assuming you can figure out the recipe (which will always consist of three items out of a possible nine you have to work with).

The characters are the game's strength, with each of them feeling distinct by the game's end, and the script has the good sense to take their struggles seriously. Even the characters played for laughs, like an alien in a spacesuit who's come to Earth on a breeding mission, are given enough pathos that when the game ended, I was surprised to find myself feeling sad to say goodbye.

There's a pleasant warmth to *Coffee Talk*, but it's also just a little too slight for its own good. It's like instant coffee and skimmed milk on a rainy morning – comforting, but it's hard not to wish for something stronger. ☹️

^ The cast is small, but each one feels distinct in design and personality.



Heli
Why would you do the mating ritual if you do not want to have offspring?

HIGHLIGHT

The realisation of a world I was already so familiar with brought a smile to my face on many occasions, and its tiny interactive biomes are, at times, spellbinding. The swamps and castles all feel like home, and I often found myself swivelling the camera around to zoom in on the smallest of details.

- ◀ The Gelfling Gurjin needs to survive, so let's move him out of the way and get on with the fight.
- ✓ Sure, he's in the palm of your hand, but SkekMal is still a horrible foe, and an evil one to boot.



Review

The Dark Crystal: Age of Resistance Tactics

Fighting fantasy

Muddled is probably the best word for this tactical outing. On the surface, everything looks pretty, and it's evidently true to the 1982 movie and The Jim Henson Company's series. The Gelflings are proud and loyal; the evil Skeksis still creep me out. Look past the style, and you'll find a robust system offering several options in its teams and compact, diorama-like levels. Each character leans towards a certain playstyle – fighting, healing, magic – though their skills can be changed around by using a job system that allows for new abilities and some level of micro-management. This system makes it easy to create 'all-rounders', however, which removes the need to balance a team – you won't need a specific healer if you can give a fighter a back-up healing spell. In other words, the tactical action lacks the depth needed to shine in a world of *Mutant Year Zero* and, to a lesser extent, *Mario + Rabbids*.

The combat can often feel easy, too – in some stages, there isn't much need to put on your tactical hat at all. I completed a few areas, for example, by simply moving a Gelfling or Fizzgig next to an enemy and wailing on them. There's a sense that the game wants you to play on the high difficulty setting, which I tried – but I felt the AI simply scaled the damage up, rather than offering a new, smarter challenge to overcome.

It is wonderful to interact with these familiar worlds though, and some areas offer a unique win condition to break up that simplistic combat. I loved pushing the Hunter Skeksis, SkekMal, into a biting pit to weaken him, or uncovering texts in the Gelfling library in order to progress. The world of Thra certainly comes alive in your hands; it's recreated brilliantly on the Switch, and its intimate, nostalgic fantasy world really suits the system.

The story gradually loses its some of its charm, however, when you're already aware of the next stage of the plot, which retells the Netflix series' story almost beat for beat. *The Dark Crystal: Age of Resistance Tactics* relies heavily on the player's familiarity with the franchise, and it's a pity the game doesn't stray far beyond the familiar: the game needed to tell a new story, perhaps one that complemented the series rather than hewed so closely to it.

Ultimately, I'm not entirely sure who the game's aimed at. Die-hard fans will undoubtedly enjoy the details on offer in the game's world, but even they may be disappointed by the plot's familiarity; tactical strategy game buffs, on the other hand, will find little here to get their pulses going. *The Dark Crystal: Age of Resistance Tactics* is by no means a bad game, but at the same time, it doesn't offer much that will satisfy either of those camps. 🗣️

“There's a sense the game wants you to play on the high difficulty”

Info

- GENRE**
Turn-based tactics
- FORMAT**
Switch (tested) / PC / PS4 / XBOX / Mac
- DEVELOPER**
BonusXP
- PUBLISHER**
En Masse Entertainment
- PRICE**
£16.99
- RELEASE**
Out now

REVIEWED BY
Daniel Lipscombe

VERDICT

Lacks the crystalline polish needed to become as memorable as the rest of the franchise.

52%

Rated
Review

HIGHLIGHT

While it stands out in absolutely no other way, *Astra Exodus* does at least have some semblance of character thanks to the simple act of putting faces to things. Every department under your control has a gurning mug attached to it, and some alien designs are pretty natty. Admittedly this is pushing things a bit to call it a 'highlight'.

Review

Exciting space combat falls by the wayside, instead replaced with plodding nonsense you will skip each and every time.

Astra Exodus

4X? More like 4zzzz

Info

GENRE
4X

FORMAT
PC (tested)

DEVELOPER
Atomic Kaiser

PUBLISHER
Slitherine

PRICE
£25.99

RELEASE
Out now

REVIEWED BY
Ian Dransfield

VERDICT

Absolutely competent at its best. Far more often it's utterly prosaic.

50%

It's a disease, I think – every time a 4X game (eXplore, eXpand, eXploit, eXterminate) pops up, I'm all over it. *Astra Exodus* fits the bill, and it's always good to give any new entry in the genre a go in case it happens to bring something new to things, mixes it up, 'pulls a *Galactic Civilizations II*', as it's known (nobody says this). *Astra Exodus* does nothing new, and plenty old, and is a fine insomnia cure.

Ignoring the semi-frequent bug that popped up and rendered the entire universe in blinding white, as well as the plentiful typos, spelling errors, and grammatical mistakes littering the game, there's little about *Astra Exodus* that's outright bad. This certainly isn't something a reviewer would, could, or should pile on regardless of how much of a release that might be. This is a game that is *absolutely competent* when everything's running as it should – and therein lies the problem. It obviously needs to be a lot more than that.

You set up your galactic homestead and spread out, colonising new planets and running the day-to-day operations as you try to get some kind of population on these new (mostly) empty rocks. Along the way, you encounter alien races and engage in diplomacy with them, in which they appear to behave in an oddly binary way – either your absolute bestie, or as though they saw you kick a puppy once. It does feel like there's little in-between. You research things

with nary a tooltip in sight (prevalent elsewhere, it's fair to point out), and ship-to-ship combat can be carried out in a top-down tactical format. Honestly, you'll skip it every single time after doing it a couple of times, because it's so slow, and your decisions on the fly impact very little. Pretty much every feature in *Astra Exodus* is one that absolutely should be there, and one that you're happy is there. Equally, pretty much every feature in *Astra Exodus* is one that induces zero excitement, offers nothing new, and does little of note. It is, in a word, boring.

Maybe you've never bothered with the *GalCiv* series, perhaps *Stellaris* isn't for you, and it might well be that *Endless Space 2* has passed you by. There's the chance you don't know what *Master Of Orion 2* is, either. If you're in that peculiar situation and don't want to rectify it by picking up any of the just-mentioned titles, for whatever reason that might be, then sure – *Astra Exodus* will fill a hole in your life. You'll be able to mindlessly tap on the end turn button and spend much longer than you'd expect hoping your population's construction output improves, while tweaking your ship designs for no discernable difference to their performance in battles. Everything from 4X bingo is ticked off, present and correct. But you shouldn't do that – just get one of the others, or if you already have them, just stick with them. *Astra Exodus* is uninspired in the extreme, and catastrophically mediocre. ☹

“Honestly, you'll skip tactical combat every single time after doing it a couple of times”

▼ A whole galaxy of possibilities?
Not really, no.



HIGHLIGHT

There's always a lot going on behind the scenes in *The Pedestrian*, and you'll miss a lot of the details (and a few cute references) if you don't halt the camera between scenes and train your eyes to inspect the nooks in the background as well as the puzzle-laden foreground.

▲ The game's thoughtful presentation extends to its cracking soundtrack.

Review

The Pedestrian

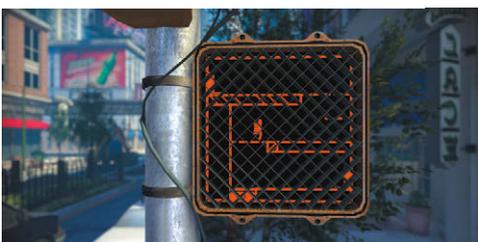
Sign of the times

It's always a delight to find a puzzle game that has stumbled onto an exciting new premise. The past few years have allowed for a surprising amount of experimentation within the heavily iterated genre, and now *The Pedestrian* gets to join the ranks of other mind-bending oddities like *Superliminal* and *Manifold Garden*.

The Pedestrian's puzzles centre on who – or what – you play as: a symbol, hopping between street signs in various urban vistas. To progress, you need to connect signage and create entryways to advance through. As the game strolls along, it introduces new concepts like picking up keys, supplying power to gates, and descending through elevators. The novelty of moving between signs remains as the puzzles start to become more like Daedalian jigsaws, asking you to connect disjointed bits of signage to create a mini-level on the face of cardboard boxes or a restaurant blackboard, all leading back to a branching puzzle hub. It's here that *The Pedestrian* could have used some more iteration, the kind I'd be delighted to see in a sequel – there were situations where the signage was just hanging in the air without any bolts attaching it to the background.

This is at odds with some of the game's better puzzles, where you cross streets and move

✓ The best puzzles in *The Pedestrian* take place on believable structures, using the environment to draw you into the world.



between genuine signage that doesn't suspend your belief that life is happening around them. I imagine this is a consequence of the designers butting heads between the minimalist symbolic approach and providing a decent amount of difficulty, so you don't just race by... I empathise. Regardless, you'll never dwell on that too much because the puzzles are far too compelling.

Because the signs are movable, it often feels like you're arranging a collage of the thoughts in your brain as you play, leading to idiosyncratic eureka moments. Yet, it's the tiniest morsels of detail in *The Pedestrian* that often prove to be the most delightful. Whenever you press pause, you'll zip to a TV hiding somewhere within that level to establish a charming continuity between scenes.

The camera moves like a one-take piece of cinema and the dynamic Pixar-esque soundtrack swells and recedes with the motion. The game's pacing is fantastic, and the plot always picks up to provide new scenery as you're figuring your way out of the few dry puzzles. It's hands-off for the most part as the game makes you figure out its puzzles without any overbearing tutorials, yet some of the mechanics introduced in the late game – such as the sign-freezing paint system – could have certainly used some extra in-situ explanation, or at least offer a means to review past solutions.

The game is ultimately tied together by one of the most brain-breaking, mouth agape endings in recent memory, which evoked genuine comparisons to Valve's instant classic *Portal*, and demands to be seen. An absolutely essential pick-up, I sincerely hope *The Pedestrian* enters the esteemed annals of puzzle history. 🗨

Info

GENRE
Puzzle
FORMAT
PC (tested)
DEVELOPER
Skookum Arts
PUBLISHER
Skookum Arts
PRICE
£15.49
RELEASE
Out now

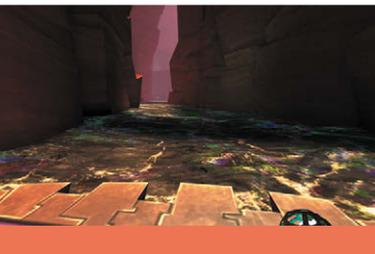
REVIEWED BY
Jordan Oloman

VERDICT

A whip-smart puzzler with a killer ending, *The Pedestrian* is one of the best one-shot puzzle games of the past few years.

84%

- ▶ Throughout the game, you'll encounter other stranded pilots who trod the same path you walk - except they all failed.
- ▼ Unlike other VR games, *Eclipse: Edge of Light* lets the player actually walk around - no teleporting here.



Review

HIGHLIGHT

Drip-feeding the story to the player via the surroundings is an interesting choice, and one I ended up enjoying more than I thought I would. It felt rewarding, and telling the story in real time is certainly more organic than using text or cutscenes.

Eclipse: Edge of Light

Stop and smell the alien roses

Info

GENRE

Walking sim

FORMAT

Switch (tested)
/ PS4 / PSVR
/ Oculus Rift /
Mobile

DEVELOPER

White Elk Studios

PUBLISHER

White Elk Studios

PRICE

£11.29

RELEASE

Out now

REVIEWED BY

Donogh Moore

VERDICT

A mysterious exploration game, and another intriguing indie added to the Switch's roster.

62%

Eclipse: *Edge of Light* is a pensive, first-person exploration game developed by White Elk Studios. Comprised of former *God of War* developers, White Elk Studios originally released *Eclipse* as a VR game back in 2017, when it won various 'best mobile VR game' accolades. So... how does a three-year-old mobile VR game work on the Nintendo Switch?

Eclipse is a walking sim that sees the player crash-landing their spaceship on an alien planet. After coming into contact with a mysterious, orb-like artefact, you have no choice but to set off and explore this clearly once-inhabited place. Along your journey, the artefact shows you remnants of a past civilisation, and of the path of corruption that turned a once-benevolent messiah into a power-consumed despot.

The story is central to the game, and is communicated through the player's interactions with the various statues and monuments they encounter. While these monoliths don't necessarily need to be scanned as you progress, the story will be communicated to you via no other means, so it's worth stopping and smelling the proverbial alien roses every once in a while.

On hearing the phrase, 'mobile-VR-game-turned-console-FPS-walking-simulator', you're probably questioning just how well it all plays.

Like a lot of the genre, the pace the player moves at feels slow at first, which is initially frustrating. As the game progresses and the story unfolds in front of you, however, the pace seems more and more deliberate; after all, if you *could* walk any faster, you may skip through and miss important aspects of the narrative.

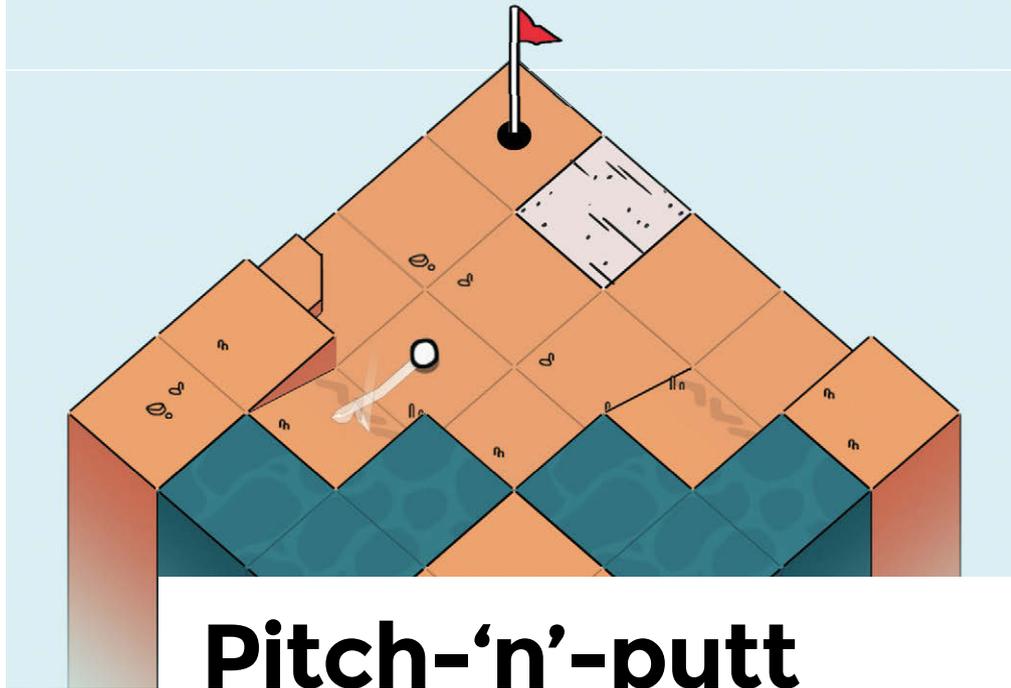
Throughout *Eclipse*, the mysterious artefact you pick up post-crash acts as your trusty one-size-fits-all tool. Not only can you throw the solid, metal orb, but it also grants you abilities such as

telekinesis and a jet pack-style propulsion, as well as scanning monuments and hidden items needed to progress. It's a functional system and sometimes a neat gimmick, but ultimately nothing special.

Visually, things are a bit all over the place, ranging from being quite pretty one moment (the murky, oil-coloured water is a treat to look at), to very much looking like a 2017 mobile game running on a 2020 console, with jagged, sharp textures jutting out of every corner.

Eclipse: Edge of Light is a pensive game with a deliberately slow pace that offers players room to breathe and reflect, all the while unearthing a mysterious and intriguing story over the course of about three hours. While at times it feels like the narrative is dangling a carrot before your nose and leading you on, the entire experience is ultimately an enjoyable one. 🗣️

"You have no choice but to set off and explore this clearly once-inhabited planet"



Pitch-‘n’-putt

One finger for puzzling par is all Ian needs in Golf Peaks

Golf games are a right sort – your straight-faced simulations, riddled with real-world sponsorships and people and courses and balls (loads of fun), your less serious games leveraging the whole fun of thwacking things long distance over the constraints of petty concerns like ‘realism’ (also loads of fun), and plenty more. Afterburn’s *Golf Peaks*, released way back in the misty past of 2018 on PC and mobile, certainly fits into that latter description, and I can’t stop playing it as a result.

Obviously, it’s a golf game. It’s got ‘golf’ in the name, which is usually a dead giveaway of something at least intending to include golf in it. That said, it’s hard to call *Golf Peaks* a *golf game*. No, this is a puzzle game – it’s about moving around squares using limited movement cards, trying to get to a goal. It just happens the thing you’re moving is a golf ball, and the goal you’re moving to is a hole. It’s about as close to golf as you’d get if you lived in a society where golf hadn’t been invented yet.

Initially, I found that to be a bit of a let down – I wanted a golf game for my fingers, something to while away a few minutes here and there on, wallop some balls about, scream ‘Fore!’ in a greengrocers, that sort of thing. But that passed, because this is a really good fun puzzle game, that solid mix of straightforward premise and depth of complexity in the situations you’re presented with.

I’m not going to stick with it with any real commitment, mind – there are no differing approaches to be found to these solutions, no individual thought celebrated. I’m not going

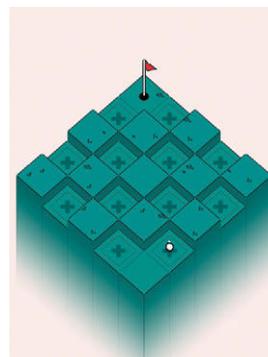
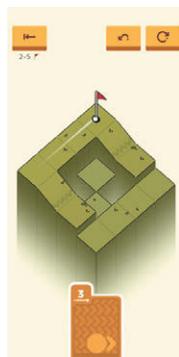
to be slicing something out of the deep rough but somehow finding it pinging off the pin to set up an easy birdie. You either don’t get to the hole and reset (or go back a step), or you do get to the hole and you move on. That’s it. The challenge isn’t specifically in figuring out the spatial puzzles, in sussing when and where you should use a specific movement card – this one hits the ball two spaces forwards, that one chips

it two and has a one space run-on – I’ve found it’s actually more about resisting the temptation to just resort to trial and error.

You can, if you want. If you want to remove almost every aspect of thought from *Golf Peaks*, you can just

systematically try out every potential move in a level until you get it right. You won’t be punished for that. I won’t, mind. I’ll stick with this one-finger putting odyssey for a while, until I get bored, and I’ll have got my money’s worth when that happens. A testament to great mobile game design, indeedy-doddy. 🍷

“I wanted to scream ‘Fore!’ in a greengrocers, that sort of thing”



Wireframe Recommends



Everybody's Golf

PS4, MULTI

If you’ve ever played the series more than once, you’re surely in love. *Everybody’s Golf* isn’t the pitch (and putt) perfect simulation some might want, but it is, far more importantly, a brilliant golf game.



Sensible Golf

AMIGA, PC

Sensible Software’s great failure still lingers with a bit of a stench, to be honest, but all the same, *Sensible Golf* is still a nice time capsule to a world just before the studio could do no wrong.



Desert Golfing

IOS, ANDROID, PC

A minimalist demi-art project crossed with golf (in a desert), *Desert Golfing* is simple, utterly captivating, and – still, six years later – bereft of any end goal. Fabulous stuff.

Backwards compatible

WRITTEN BY IAN "DYNAMO" DRANSFIELD

HDM-why?

I've nattered briefly about this before on these pages, but it's worth going into a bit more detail: the one big problem we all have when rooting old consoles out from storage (aside from them not working at all) is in TV connections. Basically, your old cables just don't plug into modern TVs, in the most part – and even if they do, it can be hard to get anything showing on-screen. So what can be done? Well, thanks to the efforts of people far smarter and less lazy than I: quite a lot. Your options depend on what console you're trying to play, and prices (and effort levels involved) can range from not much to a hell of a lot – but here are a few of those options.

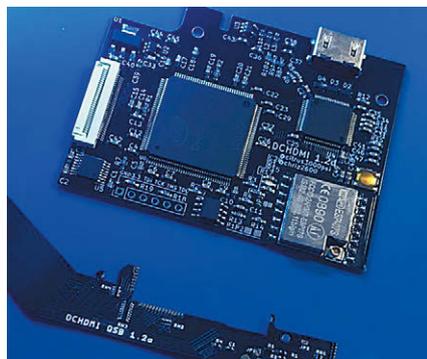
The god-tier stuff comes in the form of Black Dog Technology's HDMI solutions – currently available is the DCDigital, for Dreamcast; an FPGA-based board you install into your console (or pay someone to do, because it's difficult), resulting in digital signal drawn directly from the

system and a modifiable output up to a full 1080p resolution. It's around £120 for the unit, but it is *extremely good*. A PSone version is in the works right now, too. Similarly, a few different outfits are offering GameCube HDMI solutions, the newest of which don't require any modifications at all and just plug into the digital out port on the console – look to Kaico's £60 device for a good balance of price and performance there.

Elsewhere you can pick up a plethora of different HDMI cables for retro consoles, of varying quality. They tend not to offer a 'pure' signal, instead grabbing the existing analogue output and converting it to an HDMI/modern telly-compatible picture,

but they're very budget-friendly (around £18–45) if you look to the likes of Pound and Kaico's ranges and – unless you're an absolute purist – do the job well enough. Those looking for something better and willing to pay more (either side of £50) should have a gander at the RAD2X cables from Retro Gaming Cables, as their built-in RetroTINK hardware means you're getting a far better image quality for the price.

If you want to convert existing cables, you can pick up all manner of cheapo SCART-to-HDMI adapters from online stores – they tend to work, but that's about as positive a thing as you can say for them. But for £10, who can argue? If you do want proper, quality conversion from your old console cables though – and have £130-ish to burn – look for an Open Source Scan Converter (OSSC). This device can handle any console outputting an RGB signal, converting it to an HDMI-compliant one, and with plenty of tweaking that can be done besides. It is, in short, brilliant. Hope that helps a bit.





Dreamy

Continuing the hardware-y theme of this issue's BC, I've had my gaze drawn to this little beauty: the Dreamcase, a replacement shell for the Dreamcast, made out of metal and looking surprisingly sleek. The one downer, aside from the £115 price (not including

postage), is that it's only compatible with units using the GDEMU ODE, so anyone with an original GD-ROM drive is out of luck. Still, those with the money and the means can have a look on either Beharbro's or TR Fight Stick's sites for the new shell.

Mini delay

The PC Engine Core Grafx Mini (or TurboGrafx-16 Mini in the States) has been delayed owing to ongoing concerns surrounding the COVID-19 (coronavirus) outbreak. The virus' impact on manufacturing and shipping facilities in China is the main issue that has caused the delay, and at the time of writing Konami hasn't announced any revised release date for the mini console. As with most things related to the outbreak, it's understandable, and for the best. We can wait a bit longer.



BS2

Or bootleg PS2, if you will. Yep, more hardware – this time it's a thoroughly enticing bit of kit: a handheld PlayStation 2 made in small batches by Chinese sellers. What makes it different from your usual batch of retro gaming handhelds is that it isn't powered by an onboard Raspberry Pi, a device running Android, or anything of the like. Nope, it's actual PS2 hardware, chopped up to fit inside a handheld case, and with some custom PCBs made to make it all play nicely together. Is it official? Absolutely not. Does that matter to me? Good golly gosh, no. The PS2 is still relatively difficult to emulate – as in, you won't find any portable devices that can do it justice – so the chance to have a pocket PS2, even if my pockets need to be massive to carry it, is intensely tempting. OK, so there are some legal grey areas to navigate, given it has no physical disc drive

and the original console only used physical media – none of this digital malarkey – so you'll have to make your own peace on that. Honestly, the only thing holding me back from picking one up is the price, with units costing around £460 each. Let's... stick to those £50 Android devices for now, yeah? 🙄





Mortal Kombat

It's the pun-lover's killer feature: the fatality

MIDWAY / 1992 / ARCADE, MULTI

W

hen you're talking about the game-changing features, as we do here in the Killer Feature section, it's the temptation to make every one of them supremely important and all-encompassing; the sorts of things that alter gaming forever and, usually, make it a better place. But sometimes that initial thinking is just 'wouldn't it be funny if we said fatalities were a Killer Feature, because they are literally a feature that makes you a killer'. Bit of a peek behind the curtain for you all, there.

But then, on sitting down and thinking about it, *Mortal Kombat's* fatalities did have a huge impact on the world of gaming in a couple of ways. First up, the more obvious one, in that they changed how matches in fighting games would end – the coup de grâce was introduced: an act of absolute victory for the winner and total humiliation for the defeated as their heart was ripped out/head was ripped off/head was punched off/head was exploded by lightning/they were set alight by a demon/they were set alight by a kiss/they were uppercut quite high. It was a bold, at the time shocking, full stop on matches in a genre where traditionally someone just fell over and you moved on to the next bout.

It had an impact, and there were plenty of copycats – even to this day, the modern *Mortal Kombats* are capable of raising the hackles of the finest Mary Whitehouse impersonator with

how genuinely grotesque they can be. But in this element of their being at least, the fatality has been relatively enclosed in its influence.

No, it was *Mortal Kombat's* violence – capped off by the fatalities – that led in part to the formation of the ESRB, the United States' video game ratings board. Any ratings body would have the potential to be censorial in nature, especially if operated as a wing of government – but the controversy around *Mortal Kombat* (and some other titles) led to the formation of the self-regulating body we have today. As you might expect, a self-regulating body is much more favourable to the industry at large than an outside organisation could ever be, so the deck has ever since been stacked in gaming's favour, content-wise.

There are lines that won't be crossed – especially with the US's puritanical views on sex and nudity – so the ESRB can be an effective force in policing the content that comes out of the video gaming world. But generally speaking, its existence has proven a positive for how much games are allowed to get away with. Complaints still arise, but there's an official body present to comment; there are standards that can be pointed to and facts and figures that can be doled out to placate. You see, thanks to *Mortal Kombat's* fatalities in 1992, *Mortal Kombat* is still free to include fatalities in 2020 and beyond. Weird how some things work, but that does make those death moves a true Killer Feature. 🗿

“An act of absolute victory for the winner and total humiliation for the defeated”

Next Issue

ON SALE 9 APRIL

PHOGS!

Two heads are better than one in Bit Loom Games and Coatsink's canine puzzler

Don't miss out!
Subscribe
PG54



Also

- ▶ Your Story: make an FMV game with Twine
- ▶ Clicktastic: the rise and rise of idle games
- ▶ Game design and the grieving process
- ▶ The growing influence of tabletop games

Editorial

Editor

Ryan Lambie

Email ryan.lambie@raspberrypi.com

Features Editor

Ian Dransfield

Email ian.dransfield@raspberrypi.com

Sub-Editors

David Higgs & Vel Ilic

Design

criticalmedia.co.uk

Head of Design

Lee Allen

Designer

Harriet Knight

Contributors

Lottie Bevan, David Crookes, Konstantinos Dimopoulos, PJ Evans, Andrew Gillett, Tony Jeffree, Daniel Lipscombe, Keza MacDonald, Sean Martin, Donagh Moore, James O'Connor, Jason Oloman, Olly Smith, Eben Upton, Mark Vanstone, Howard Scott Warshaw, Alan Wen

Publishing

Publishing Director

Russell Barnes

Email russell@raspberrypi.com

Director of Communications

Liz Upton

CEO

Eben Upton

Advertising

Commercial Manager

Charlie Milligan

Email charlotte.milligan@raspberrypi.com

Tel +44 (0)7725 368887

Distribution

Seymour Distribution Ltd

2 East Poultry Ave, London EC1A 9PT

Tel +44 (0)207 429 4000

Subscriptions

Unit 6, The Enterprise Centre, Kelvin Lane, Manor Royal, Crawley, West Sussex, RH10 9PE

To subscribe

Call 01293 312192 or visit wfmag.cc/subscribe

Subscription queries

wireframe@subscriptionhelpline.co.uk



This magazine is printed on paper sourced from sustainable forests and the printer operates an environmental management system which has been assessed as conforming to ISO 14001.

Wireframe magazine is published by Raspberry Pi (Trading) Ltd, Maurice Wilkes Building, St. John's Innovation Park, Cowley Road, Cambridge, CB4 0DS. The publisher, editor, and contributors accept no responsibility in respect of any omissions or errors relating to goods, products or services referred to or advertised in the magazine. Except where otherwise noted, content in this magazine is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported (CC BY-NC-SA 3.0) license. ISSN: 2631-6722 (print), 2631-6730 (online).





CQ32G1
2560X1440

144 HZ	FreeSync	1 ^{ms}
	Curved	QHD

BEND YOUR REALITY

Available at:



ebuyer.com



aocgaming.com



[@aoc_gaming](https://twitter.com/aoc_gaming)



[@aocgaming](https://www.instagram.com/aocgaming)



[@aocgaming](https://www.facebook.com/aocgaming)