

£3

ALL FORMATS

Wireframe

LIFTING THE LID ON VIDEO GAMES

Are video games art?

Issue 16 £3
wfmag.cc



Groan

The Full Monty

Paintings come to Pythonesque life in *The Procession to Calvary*

Out of the box

The escape room genre returns

Going indie

From triple-A to startup duo

Point-and-click

Exploring the handmade world of LUNA



UPGRADE
TO LEGENDARY



AG273QCX
2560x1440

144 HZ

1 MS

Curved

QHD

FreeSync2

amazon

BOX

ebuyer.com

AOC
GAMING

f @aocgaming

t @aoc_gaming

i @aocgaming

aocgaming.com

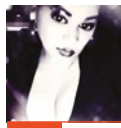
Beyond Designed Experiences

Over a chill autumn weekend in 1997, my best friend and I crushed *Castlevania: Symphony of the Night*. We took turns, staying awake, fuelled by Pizza Hut and Tahitian Treat, for as long as we could. One of us would sleep while the other kept playing. It's not exactly what people think of when they think of couch co-op. At each hand-off, we'd have to recount the adventure up to that point. Detailing plot beats, boss encounters, zones that had been traversed. We'd forget things. Doorways to come back to. What was really going on with Maria and Richter. We embellished others (*no way* he didn't take a single hit fighting Death). We were telling the story of the game not as it was or as intended, but as we perceived it. And in the end, we had a wholly synthetic experience. It was a kind of magic that I've been chasing in games ever since.

"Game design is experience design. You try to scaffold and support the experience you hope the player has with your text," is a quote from game studies and design professor Dr. Todd Harper that I've been thinking about a lot lately. It's right. At the same time, I've become much more interested in how players can co-opt and even dismantle that scaffolding, not to take ownership per se, but to create alternate, unexpected experiences.

In the concluding hours of Waypoint's most recent Save Point stream, Rob Zacny and Austin Walker took turns playing *XCOM 2* while the other was blindfolded. Neither one got to see the actions of the other. In effect, creating a kind of exquisite corpse playthrough in a tactical war game, where coordinated action and an understanding of deployment is often crucial. Even if the designers conceived *XCOM* as being capable of played cooperatively, it surely wasn't like this.

But as long as games have existed, once we've exhausted the designed experiences, we've sought newer ones. Often this is through self-directed



DIA LACINA

Dia Lacina is a queer indigenous trans woman writer, photographer and founding editor of CapsuleCrit.com, a monthly journal dedicated to microgenre work about games. She tweets too much at [@dialacina](https://twitter.com/dialacina)

play, whether it's by exposing the limits of the game and exploiting them, imposing our own additional constraints, or outright cheating.

I'm also thinking of Walker's permadeath streams of *Breath of the Wild*. *Speedrunners*. Or the innumerable games I've modded, made myself immortal in, granted myself the powers of flight or to walk through walls. Or how we've all seen just how many cheese wheels our computers can send rolling down the side of a mountain in *Skyrim*.

None of which is intended, but which extend our experiences beyond the scaffolds provided. On one hand, it's to exercise our sense of ownership over media, or achieve greater value for money. Just as often, however, it's our desire to stay in worlds we love, to savour and understand these games, their construction, and what they mean to us. And sometimes, these experiences leave the digital boundaries of the games themselves and enter into our communal consumption, discussion, and the sharing of these experiences with one another. Streaming, Let's Plays, criticism, game photography and filmmaking, fan-created works, and forum discussion all extend and alter the intended experience of playing a game, not just for the players, but also those who are traditionally thought of as non-players. One only has to look at how the experiences built into *Dark Souls* have extended far beyond the scope of the original design (and dominated whole discourses). While those actively participating in a game outside of its confines might not necessarily be growing sales, those experiences are no less real or important.

None of which is to say that those scaffolded experiences are less valuable, either: there would be no game without them. But understanding that a well-designed scaffold can just as easily become a playset for the adventurous player to explore and thrill themselves on far beyond the original intent – it's a celebration of that design, rather than an abnegation of it. 🎮

#16



Contents



6

Attract mode

Interface



18

06. The Procession to Calvary

A point-and-click adventure with a Monty Python edge

10. Road To Guangdong

Pootle through China in a unique road trip drama

12. LUNA The Shadow Dust

A hand-crafted game inspired by Ursula K. Le Guin

16. Incoming

Detectives, dark fantasy worlds, and killers in pointy hats

18. Going indie

Former Guerilla Cambridge devs discuss their leap of faith

22. Escape Rooms

A neglected genre makes a welcome return

44. Joffa Smith

Exploring a ZX Spectrum legend's life and games

50. id Software

The Texan studio behind Doom and Quake profiled



50



22



56

Toolbox

- 28. CityCraft**
Considering real-world materials, and simulating them in a virtual city
- 30. The power of focus**
Developer advice from a Splinter Cell and Battlefield veteran
- 32. Save/Load**
A guide to saving and loading player data in Unreal Engine 4
- 40. Source Code**
Code your own Gradius-style Option sprites

Rated

- 56. Observation**
Systems management has never felt so tense
- 60. Void Bastards**
Jon Chey's strategy-shooter was well worth the wait
- 62. Mordhau**
A gleefully bloody medieval combat simulator
- 64. Persona Q2**
Bid farewell to the 3DS with a solid JRPG spin-off

WELCOME

As I write this, the internet is aflutter with the news that Keanu Reeves has a surprise role in CD Projekt Red's forthcoming *Cyberpunk 2077*. The revelation comes hot on the heels of *Death Stranding*'s recent eight-and-a-half minute trailer, which featured such entertainment industry lights as Norman Reedus, Mads Mikkelsen, Guillermo del Toro and assorted other Hideo Kojima chums scowling their way through an opaque sci-fi action epic. We're in a new golden age of celebrity video game appearances, it seems; forget about the likes of *Hollyoaks*' Gemma Atkinson and martial arts sensation Gina Carano in *Command and Conquer: Red Alert 3* – it's hard to remember a time when quite so many fancy actors and directors were last willing to be seen in a video game. Go back 20 years or so, and you probably wouldn't have expected, say, Stanley Kubrick and Tom Cruise lining up to appear in such a patently surreal game, even if it was helmed by one of the industry's most familiar names. It's further proof, surely, that it's cool to be associated with video games in 2019.

Certainly, it's all a far cry from the time when Gordon Ramsay apologetically showed up in his own Wii game (that'd be 2008's *Hell's Kitchen*), and his digital avatar looked like someone had placed a blonde wig on top of an angry pink space hopper. We live in a brave new gaming world, readers. Enjoy the new issue!

Ryan Lambie
Editor





Pythonesque

We catch up with Scottish developer **Joe Richardson** and his point-and-click adventure made from Renaissance art

There have been numerous attempts to capture the spirit of Monty Python in a video game over the decades, but few have matched the comedy troupe's irreverent, anarchic tone quite as well as Joe Richardson's *Four Last Things*, a point-and-click adventure made from recycled Renaissance art.

With his upcoming sequel, *The Procession to Calvary*, Richardson is hoping to do the same again, with players this time controlling a new lead character who, having survived a holy war, and prompted by the actions of Heavenly Peter at the end of the last game, travels to a region known as the South, again assembled from a collection of Renaissance paintings.

The Procession to Calvary is the third of Richardson's games to feature his collage art style. The first, *The Preposterous Awesomeness of Everything*, was his final illustration course project at the Camberwell College of Arts. Unlike *Four Last Things* and *The Procession to Calvary*, though, *The Preposterous Awesomeness of Everything* was made up of images he'd taken of himself – including some nude photos – that he used to assemble the primitive cast of characters.

"I liked the art style," says Richardson. "I think it suited the game, but it wasn't pretty to look at. So, when I was doing that, I thought, basically, what if the subject matter wasn't my ugly face? What if I wasn't using disgusting subject matter, but beautiful subject matter? So that

was basically the whole idea for *Four Last Things*. This art style, but using beautiful Renaissance art as the starting point."

Public domain

When *Four Last Things* released, the game's artwork drew immediate comparisons to the animation style of Monty Python's Terry Gilliam, whose influence Richardson argues was unintentional, at least to begin with.

"I'm obviously aware of Terry Gilliam," says Richardson. "I'm a Monty Python fan... so when people started saying it, I was obviously very happy. That's a huge compliment. And I think subconsciously it probably was something that helped towards me finding that style. But it wasn't a conscious thought before I started."

Like many of Gilliam's madcap animations, everything in *Four Last Things* and *The Procession to Calvary* is

▼ After experiencing a terrible nightmare, the protagonist of *Four Last Things* must repeat seven deadly sins to gain entry to the church and seek forgiveness.





A Painting of Three Naughty Boys

▲ Like *Four Last Things*, *The Procession to Calvary* will feature a fair bit of metacommentary – the art used to make the game also exists in the same world as framed paintings.

constructed from materials gathered from the public domain. In this case, that includes Northern and Italian Renaissance paintings and recordings of classical music.

“I’m really not very knowledgeable [about Renaissance artists],” admits Richardson. “I went to art school, for what that’s worth... And so, I knew Bosch and Bruegel. But that was all I knew. So, the process of making *Four Last Things* – a lot of it was just learning loads about it and realising there was so much more to it.

“The artwork is slightly easier [to find],” he continues. “Most of that comes from either Wikimedia Commons, or a lot of galleries now are putting these awesome hi-res scans of their catalogue up in the public domain. But the music is pretty difficult to find. And not only finding music, but then... I’ve got to find a painting with characters playing appropriate-ish instruments to make it look like they’re playing the music.”

To compensate for this, Richardson allows himself some extra wiggle room. While the artwork in *Four Last Things* and *The Procession to Calvary* is usually from a specific period in history, the music is actually from a much wider timeframe.

“The music comes from a bunch of different online archives,” states Richardson. “The main ones are Musopen and the International Music Score Library Project. I’ve spent hours trawling through these sites.

▼ By the nature of moving towards using more Italian Renaissance artists, there is a lot more religious imagery on display in *The Procession to Calvary*.



“In *Four Last Things*, the music comes from all over Europe and ranges from genuine medieval compositions all the way up to stuff from the 1800s – I used Erik Satie’s *Gymnopédie No. 1* in the last scene. I don’t have anything quite that modern in *The Procession to Calvary* yet, but it’s a similarly broad spectrum. I basically just try to find stuff that ‘sounds appropriate.’”

Artwork first

Another effect of assembling a game from pre-existing art is that it alters the production timeline significantly. Writing dialogue and story usually comes much later on in the process, after tracking down all the relevant art, constructing scenes for players to walk around in, and adding 3D effects like parallaxing.

“People are always a bit surprised when they find out just how backwards my design process is,” says Richardson. “So, for *Four Last Things* and now *The* ▶

“I do the artwork and then start thinking about puzzles, because I can’t decide to use a pig and then not be able to find a pig”

Procession to Calvary, I have the entire game’s artwork and all the rooms.

“I do the artwork and then start thinking about puzzles, because... I can’t decide to use a pig and then not be able to find a pig. Or decide to do a puzzle where you have to roll a wheel, then not be able to find a rollable wheel. So, I find things that are going to be [interactive] in the artwork and design puzzles around that. And then have to cobble together the story last.”

Alongside this ‘backwards’ approach, Richardson also places a strict rule on how he constructs the environments in his games, with the idea generally being to preserve as much of the original painting as possible in the finished project. This means that he won’t excessively remove objects from the environment or try to flatten areas to create pathways for the player to walk on.

“For each scene, I find between two and five nice paintings of a similar perspective and subject matter, and then find a way to pin those together without destroying any of them,” he explains. “I also try not to add too many superfluous bits, so items



^ Every environment in *The Procession to Calvary* is assembled from multiple paintings that have been meticulously pinned together.



you pick up from the scene will usually be things that were already in those paintings, and I’ll just cut them out and fill in the background behind them so they can be put in and taken away again... It’s a lot of matching colours. Changing the hue and saturation and brightness and contrast to get things [right].

“Some people think I’ve just taken some Renaissance art and saved myself a bunch of time,” he continues. “But pinning them together in a way that like feels like a cohesive scene and giving it depth is the majority of the work. And animating is an absolute nightmare.”

Puzzling solutions

Because of this approach to game design, the puzzles in Richardson’s games often end in fairly ridiculous and obvious solutions that tend to poke fun at the player for overthinking them.

“It’s hard to do too many things that change the scene,” he says. “Because picking up an item – I don’t just have to



^ The environments in *Four Last Things* are made from paintings from the Northern Renaissance, which favoured more naturalistic environments.



▲ There are numerous musicians in the game, but the player will only jam with other lutists.

cut that item out. I've got to, like, fill in the gaps you make in the background from cutting the item out. So, to try and do things that have a huge impact on the environment would just be so much work."

One great example of how he gets around this can be seen in *Four Last Things*, when players must find a way to commit the sin of sloth. Here, the solution is almost insultingly easy for players to actually perform: you simply need to leave your keyboard alone for a few seconds, after finding a place to lie down, and that's pretty much it. It's a simple puzzle that relies on some fairly straightforward logic, but it's one that has a hilarious payoff as a ridiculously overdramatic sin meter pierces the silence, condemning your immoral activity.

"My puzzles are mainly gags," argues Richardson. "I'm not trying to make difficult adventure games. That's what I used to like to play, back when I played adventure games when I was a kid. I think now I've lost my taste for that. If I get stuck in a game, I give up. I'm more interested in the art and the story, and letting the story flow is important. And having puzzles that are too difficult to block that, I think, detracts

more than it adds. So, they're just more light-hearted."

A more recent example of this type of humour in Richardson's games is the new sword mechanic in *The Procession to Calvary*. Players can use the protagonist's sword in order to skip puzzles they may be stuck on. It's a bold design step, and one that could lead to players getting through most of the game without picking up a single item.

"The idea is that you're going to be told at the start of the game not to use the sword," says Richardson. "So, you're going to potentially be able to just murder your way through your first playthrough in 20 minutes. And then probably not get the ending you're looking for and realise that it might be a good idea to try and go back and not be quite as murder-y on the second attempt. But exactly how the reward for not doing that is going to play out, I haven't entirely decided yet."

A conversation with God

Another recognisable trademark from Richardson's games is their reliance on self-referential comedy. For instance, in both *The Procession to Calvary* and *Four Last Things*, the paintings used to construct the world also exist within the game world itself, which means puzzles can sometimes require players to refer

back to the original artwork. This is the case in *Four Last Things*, where you have to track down the name of the artist who was responsible for painting the protagonist's face, in order to answer a riddle and secure an important document for a creepy lawyer.

"It's something I have to be careful with," says Richardson. "Because – if it was just for me – I would really up the meta gags. I really like them. But I know it doesn't go down well for a lot of people."

"I do really like them, though. And I'm thinking about the possibility of a conversation with God in *The Procession to Calvary*, which would get very meta. I always remember there's a bit – I think it's in the Alasdair Gray novel *Lanark* – where the main character talks to the author and it gets super meta, and then down the side of the pages there are all these references to things he's sort of stolen or just listed for no reason. I loved it. And I'd like to do maybe just one bit where it gets really meta. But again, I've not got there yet." 🗨

***The Procession to Calvary* releases on 15 July for PC.**



Road to Guangdong

Keep rollin', rollin', rollin'

Info

GENRE
Road-trip-'em-up

FORMAT
PC / XBO

DEVELOPER
Just Add Oil Games

PUBLISHER
Excalibur Games

RELEASE
Out now (early access)

'An emphasis on forethought and planning' isn't going to be a back-of-the-box bullet point for any major triple-A title coming this (or any) year, so thank crickey for the likes of *Road to Guangdong*. Just Add Oil's road trippin' story tells the tale of lead character Sunny and her journey through the eponymous Guangdong province in the south of China, pootling through a stylised 1990s version of the former Canton on her way to see family, pick up recipes from them, and ultimately win their blessing to run the family restaurant back home.

There is, of course, a bit more to it than that – you're not looking at map-clicking fast travel in *Road to Guangdong*: there's a lot of driving to be done. Making your way from one point of interest to the next is where the main challenge comes into play, because, like most youngsters the world over, Sunny's car is... not great. Prone to breakdowns – because of wear and tear, but also because of how you might drive

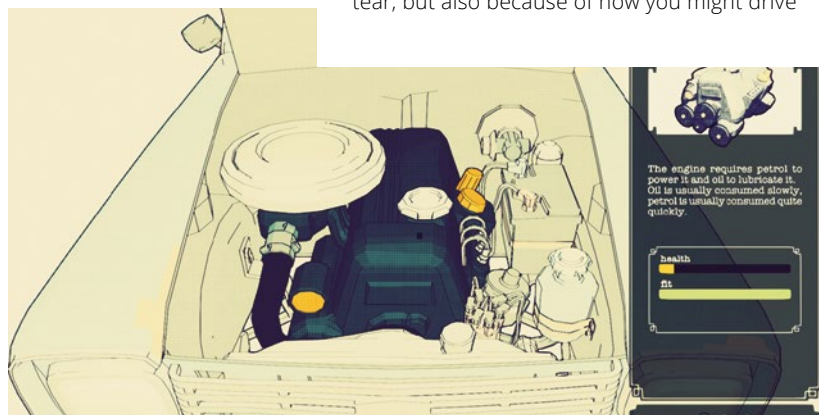
the thing – there's a lot of said breakdowns, repairs, and attempts to keep things running from petrol station to petrol station on your way to see the next relative. So just like a real road trip in a car not fit for purpose, that's where the 'forethought and planning' comes into play.

So how does an idea like *Road to Guangdong* come about, then? Alex Darby, game designer and programmer, explains: "I saw this as a chance to do something off the beaten track and pitched the hare-brained idea of retelling the classic Chinese folk tale *'Journey to the West'* as a slow driving game set in 1990s China, but with a sort of 'buddy movie in the oeuvre of Wes Anderson' angle." So far, so un-Chinese. While the will and interest was present for Darby and publisher Excalibur Games, the designer knew it had to go a bit deeper than childhood viewings of cult TV classic, *Monkey!*

"When Excalibur were interested," Darby continues, "we realised straight away that, despite our interest, as middle-aged British white men [we] had a very limited understanding of Chinese culture, and so our main priority was to find a writer with Chinese heritage who could give the game an authentic voice. We found Yen, and she's taken it in new directions we could never have done without her."

Yen Ooi, a PhD student and lecturer at the University of Westminster, as well as short-story author and novelist, guided the themes and tone of *Road to Guangdong*; away from a vague retelling of a story understood through the eyes of an outsider and into the realms of authenticity. "For me," Ooi explains, "life itself is like a road trip where we're always getting from

▼ Just like my 2002 Mini One, this engine is Not In Good Health.



point A to point B, so setting 'life stories' in a game like *Road to Guangdong* seems natural.

"Many of us now live far away from our families, and we rarely have opportunities to get together, other than at the occasional holiday," she continues. "So, the visits we have are usually bittersweet, where the moments together are poignant even if the events are trivial and sometimes stressful! And when we part again, it leaves us wanting more. I think it is this feeling – the affinity and nostalgia of family visits that brings out the very relatable human emotions that will hopefully come across while playing *Road to Guangdong*."

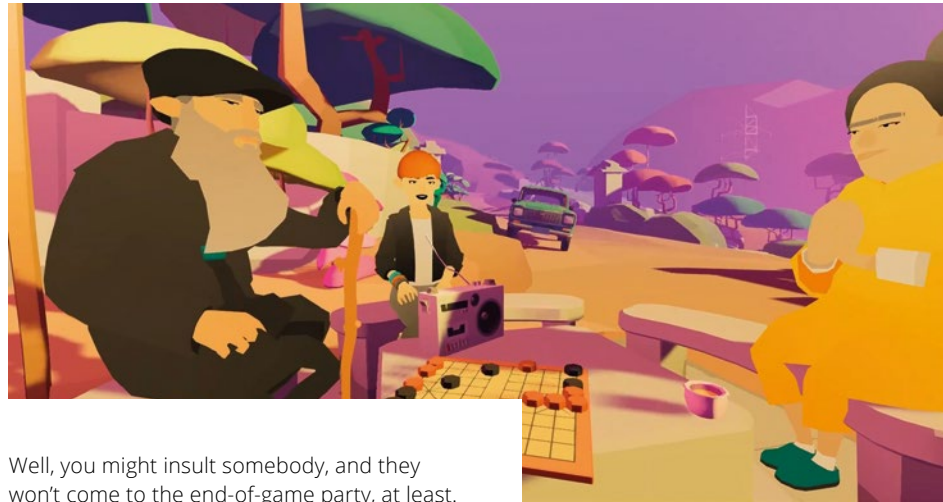
That relatability shines through in the game's road tripping aspect, as well as its familial bonds. Even those of us not so close to our family unit should know the pleasure – and pain – of the extended car journey, and this framing is something Darby is keen to develop: "I think that – as a narrative setting – road trips allow this sort of 'forced but believable' proximity between characters for prolonged periods," he says. "In the context of driving, there's a

shared burden of concentration, boredom, and resignedness to being sat there, which almost necessitates the dropping of emotional barriers.

"I think this situation really lends itself to exploring the kind of topics which otherwise get overlooked but which often relate to deeper human truths, needs, or desires – I reckon maybe 75% of the really important decisions I've made with my wife have been made on long car journeys – getting married, having a child, getting work done on the house, etc."

There's an emphasis on the long, drawn out (dare we say 'boring?') drives, but it's the meetings with the family along the way that make up those core, decisive moments where you impact the story and divert the narrative.

▼ The long road ahead... just remember tiredness kills, yeah?



Well, you might insult somebody, and they won't come to the end-of-game party, at least. "The choices you make in the story do have consequences," Darby explains. "For example, if you lie, that might upset someone, and mean that they decide not to give you their recipe; or that they refuse to give Sunny their blessing as the restaurant's new owner; or maybe they will just refuse to come to the Spring Festival celebration meal."

So it's not a case of moral burdens forcing you down the path to being a monster, but "more emotional choices," Ooi says. "They're not different from the choices we face day-to-day with close friends and family. You could

be a tyrant and cause all your family to be upset, but it'll mean a very quiet Spring Festival reunion meal!"

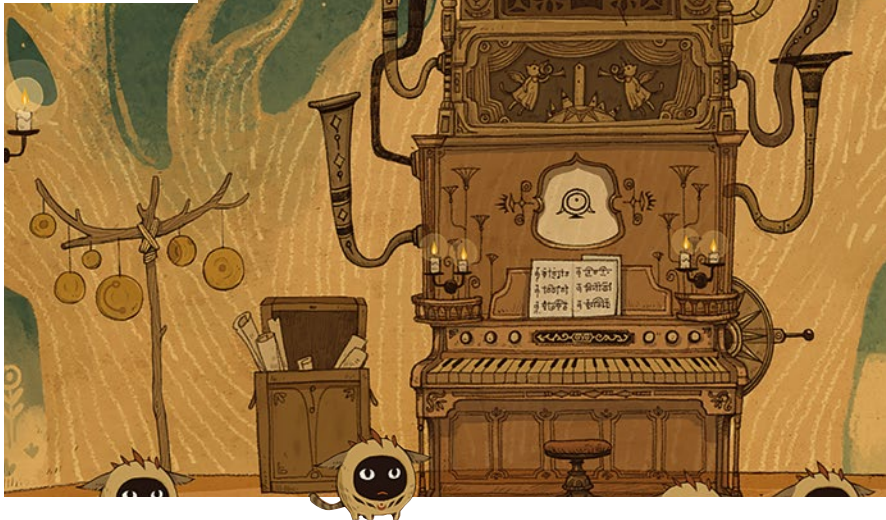
Road to Guangdong is out now on Steam early access, with a full release set for later in the year on both PC and Xbox One. But once it's out on the road, what does Darby think he's going to get up to? "Lots of sleep mostly, I think!" he laughs. "I have real trouble switching off, and working from home means that I have a tendency to 'just quickly finish one bit' of the game which means I regularly end up going to bed well after midnight, and I have to be up at 7am to get my son to school!" 🙄

▲ As you progress, things open up more and allow more choice of where to go next.

"Life itself is like a road trip where we're always getting from point A to point B"

BANGERS

While *Road to Guangdong* sees you fixing a lot of problems with the car at the side of the road, there are some things you'll need more help for. "Some stuff – like replacing an engine – requires a garage, and also costs for the price of the labour as well as the parts," Darby says. "If you break down and can't repair it yourself, it's very costly to get the car towed to a garage, so you really want to avoid that. This gives the game a sort of economy of keeping an eye on the car's various parts and balancing the cost of travel and buying new parts."



Absolutely gorgeous

Lantern Studio talks Ursula K. Le Guin, Ghibli love, and hand-animating *LUNA The Shadow Dust*

Info

GENRE
Point-and-click

FORMAT
PC / Mac / Linux /
iOS / Android

DEVELOPER
Lantern Studio

PUBLISHER
Coconut Island
Games

RELEASE
Summer 2019

Some puzzles require the two characters to split up, with changes to one room impacting the objects in the other.



It's hard not to fall in love with the 2D-animated art style of *LUNA The Shadow Dust* – the debut point-and-click adventure game from Lantern Studio. From the cute Ghibli-inspired character design to its mysterious locales and detailed animations, the project immediately draws the eye with its charming visuals.

The game is the brainchild of the London-based freelance animator Beidi Guo – whose previous work includes projects for BBC Children in Need and Elton John. Together with a small four-person team located across three continents (Europe, North America, and Asia), she's been working on *LUNA* since 2015, back when Lantern Studio first formed.

In *LUNA*, players switch between two characters – a humanoid boy in a rabbit hat, and a shadowy cat-like creature – as they explore a mysterious tower. Each room in the tower holds a new puzzle for players to solve, with most focusing on teamwork and interactions with your environment, such as pulling levers, moving boxes, and illuminating light sources.

"We don't have an inventory system," says Guo. "The characters, they only interact with their surroundings. And we don't have any pop-ups. All of the environment is fixed. Because we don't have any inventory, all the characters need to have even more sets of animation for them to interact with all those objects. So if we have a chair, we need to make sure the chair or the stone or the box they are about to step on is the same height. So there are a load of design details that we need to document or follow."

BEGINNING THE ADVENTURE

The initial idea for *LUNA The Shadow Dust* came from a student film Guo made in her final year at university, called *The Plenilune*. The animation tells the story of a man who pilots the moon around Earth, and his long-distance relationship with his family back home. Both feature a similar art style to one another and reuse the same imagery, including the ancient tower and the pale moon.

"In that film, the story is very different from *LUNA*," says Guo, "but I really like some of the elements. It has a mysteriousness to it. So I picked some of the key elements from that student film and then rewrote a new story – which later became the story for *LUNA The Shadow Dust*."

Back in early nineties China, buying video games was considered a luxury and a huge personal investment. Luckily for Guo, though, her mother had a job in the computer department of her company, meaning she had in her possession a brick-ish IBM laptop that she



▲ Teamwork is important in *LUNA*. For example, you'll need to switch to the cat-like creature to reach higher ledges and find secret passageways that only they can fit through.

could play games on. Guo poured hours into playing point-and-click adventure games like *The Neverhood* on this machine – an experience she credits with kickstarting her love of animation and video games. With such a huge personal attachment to the genre, therefore, it made perfect sense for Guo to set about building *LUNA* as an adventure game in the same style.

On top of this, there were also some other influences that came from outside the world of gaming. These include the works of the acclaimed Japanese animation house, Studio Ghibli, and American fantasy author, Ursula K. Le Guin.

"I've been a big fan of Studio Ghibli films since I was very young," says Guo. "The way we tell stories in *LUNA* was influenced by their films. I also like to read fantasy books, so the story for *LUNA* definitely takes inspiration from *The Earthsea Quartet* by Ursula K. Le Guin. I really like the setting of her fantasy world, which is not simply a tale of good vs evil, but rather drawing attention to the importance of the balance between the two."

"There's a certain charm that can only be achieved by hand-drawn animation"

Guo points to one quote in particular from *A Wizard of Earthsea* as having greatly impacted her work on *LUNA*. It reads: "To light a candle is to cast a shadow." According to Guo, the relationship between light and shadow is a recurring theme throughout the entire game, represented in the puzzle design and the relationship between the boy and his animal companion.

A HANDS-ON APPROACH

LUNA also features an entirely 'wordless' approach to storytelling. Instead of dialogue, story and emotion are conveyed to the player through hand-animated cutscenes, intricate sound design, and musical motifs.

▼ *LUNA* unfolds without a word of dialogue or on-screen text, allowing the sound and visuals to shine.



"During the cutscene, we'll have proper animated storytelling, like short animated films," Guo explains. "So, the music and animation work together to tell a part of the story. We have seven cutscenes throughout the whole thing. In total, it adds up to about 20 minutes of 2D animation. That's quite a lot of work for a studio the size of ours.

"The [animation] can be very intense and very time-consuming," she continues. "[But] there's a certain charm that can only be achieved by hand-drawn animation. The process is very hard, because we only have one person. But we break it down into very short tasks to make sure I don't get too overwhelmed by the workload."

Alongside the demands of animating everything by hand, organisation also presented an issue for the studio. Having a team spread out across multiple time zones meant coordinating meetings was difficult at times, with members of the team active during different hours.

"Communication is a big challenge," says Guo. "Just to set up a time for a meeting sometimes is difficult. Someone has to stay up really late or get up really early. But we work it out and we also found an advantage of this setup, because we can all attend a game convention in different locations at the same time. So it kind of has its pros and cons... To be honest, I would prefer if everyone could work in the same office, but... reality is unpredictable sometimes."

LUNA The Shadow Dust is due to release this summer for Windows and Mac. You can already play a brief demo of the game on Steam, which takes you through the first half hour and some of its absorbing early puzzles. 🎮



▲ The game takes place in a mysterious tower, filled with puzzles and ancient writings. An overview map will detail your progress as you travel between each room.

COMPOSING LUNA'S SOUNDTRACK

For the game's soundtrack, Susie Wang, *LUNA*'s composer, took a broad approach to her work. Rather than composing individual tracks for specific areas, like level themes, she instead wrote motifs for the two main characters and then arranged different variations of each to match the tone of new environments and communicate emotional story beats.

Headlines

from the virtual front



01. Stadiams get details

Google's Stadia... game... thing... has clarified just what it is and how much it'll be, and it's seen a divided reaction. In short, it's less the Netflix of games, more the Amazon Prime of games. Launching in November, you'll pay a monthly subscription of £8.99 to get access to 4K60 streaming of a bunch of games, beginning with *Destiny 2*. New titles, however, still have to be purchased in the old-fashioned way before you can stream them – it's not an all-inclusive subscription cost, but it's still a decent entry point for those fearful of consoles and computers.

More awkwardly, the initial availability of Stadia is limited to a £119 hardware bundle or Pixel 3 phones, so it's not really breaking down any barriers of entry just yet. By 2020 we should see both a free version of the service – limited to 1080p and minus those subs-inclusive games – and Stadia working on more than just the specific hardware it uses in 2019. Will it change the world? Unlikely. Will it make more people play great games? The hope's still there.

02. AMOS returns

AMOS The Creator's creator is back creating more AMOS. Say that six times fast. OK, so this isn't new news particularly, but it only just popped up on our radar: François Lionet, creator of legendary BASIC dialects STOS and AMOS (for Atari ST and Amiga, respectively), announced earlier this year he would be returning to the 'first game engine' he created 30 years ago.

AMOS 2 will be fully compatible with original AMOS and STOS creations, so anything you might have hacked together using its contained BASIC development environment all those years ago will work on the new version. But, of course, it doesn't stop there: AMOS 2 will produce cross-platform JavaScript and HTML5 code for PC and mobile platforms, as well as supporting modern things like controllers. You can find out more and support Lionet's efforts over on Patreon: wfmag.cc/amos2



03. Coherent

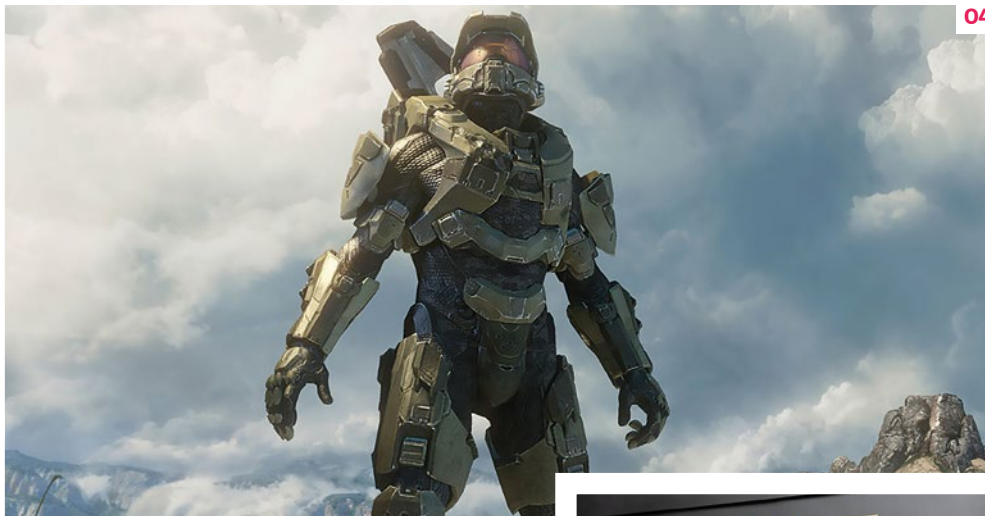
Dino Patti, co-founder of *Limbo* studio Playdead, and Unity co-founder David Helgason have announced a new venture: Coherence. A cloud-based, open-source platform allowing development teams of any size to create persistent online games, even if they're not very experienced. At least they're the claims being thrown around at this very early point. Joining the two founders is Peter Björklund as CTO, whose experience as lead networking programmer on DICE's Frostbite engine is sure to come in handy.

Coherence aims to allow for rapid prototyping and real-time interactions with projects as and when they're being made, thus helping to democratise the entire process. Improbable's SpatialOS has already got a foothold in similar territory, but the Swedish upstart has entered the fray with some confidence. It's the games, though, that will matter after all the bluster.



Baldur's Gate 3 coming from Larian Studios; *DnD* glee erupts

Cyberpunk 2077 out 16 April, 2020; has Keanu Reeves in it



04

04. Halo GoT no incest

The upcoming *Halo* TV show will be like *Game of Thrones*, according to those in charge – but not *like that*. Speaking on the Academy of Interactive Arts and Sciences' *Game Maker's Notebook* podcast, the *superbly* named Kiki Wolfkill, head of transmedia and entertainment at 343 Industries, explained what the focus will be for the upcoming Little Chef adaptation. Master Chef. Whatever he's called.

Game of Thrones is the name on all the planning notes, it seems, with the show held up as the gold standard for what an episodic serial should be. But while the scope, scale, and complexity of HBO's gore-and-carnal desires-'em-up is the focus, the *Halo* show will not bring in George R. R. Martin's go-to shocker: incest. "A lot of the background of *Halo* is this sort of political drama," Wolfkill explains, but she points out that "no incest [is] planned at all for this show, I'll say that. If you're looking for that, you won't find it here."

The *Halo* Show, as it isn't called, will arrive on the Showtime network... at some point. Hopefully it'll be worth the years-long wait.



05

05. Playdate, or not Playdate

Playdate, the little handheld with the crank announced recently, has already run into a big ol' pile of controversy after its creators began aggressively defending the device's copyright. Well, aggressively is the wrong word. 'Overzealously', maybe. See, there's a small event in Los Angeles that's been running a fair few years by the name of – can you guess? – Playdate.

Some time in 2018, an employee at Panic, creators of the handheld, mailed organisers at the event with a heads-up, saying it was working on something that could cause name-based confusion. Again, in 2019, Panic emailed with the polite-but-entitled suggestion the Playdate festival definitely change its name, as Panic had trademarked the Playdate name. What fun! Well, it's all been sorted out, and Playdate can be called Playdate while Playdate will remain Playdate and... oh, I feel so sleepy. Moral of the story? Point out people are being overzealous on Twitter. It helps.



06

06. Doom on NES, naturally

Simple as that, really. A chap by the name of TheRasteri – on YouTube, at least, it's not his real name – posted a video of id's legendary *Doom* running on a bog-standard Nintendo Entertainment System. We might have seen the FPS masterpiece on the SNES, calculators, even a printer – but it being on NES feels like a step too far.

But hark! It's not as clear-cut as it seems, and some cheating was done. Turns out TheRasteri actually used a technique not too dissimilar to how the Super FX chip functioned in SNES games, replacing the Star Fox-powering add-on with... dare we say it... a Raspberry Pi. The Pi itself runs *Doom*, while the NES displays the graphics directly through its picture processing unit. It's cheating, sure, but it's really cool, really interesting cheating. Next, *Doom* on the Amiga, please. What do you mean they already did that? Crikey.

Elder Scrolls 6 'has to last ten years', says Todd Howard



Xbox body spray announced. It's... Lynx, obviously

Incinerator

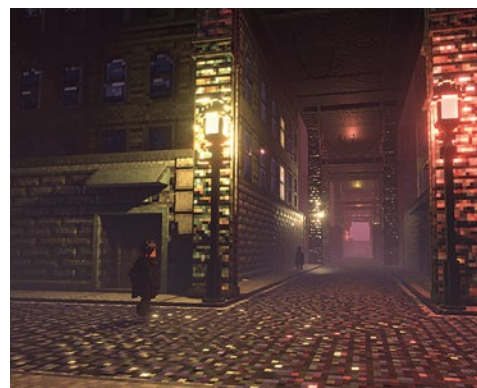


Wattam ↗

Much-loved eccentric Keita Takahashi hasn't yet made a game that's matched the cult success of *Katamari Damacy*, but maybe *Wattam* – a project that's been in development for at least four years – will be the title to reintroduce his idiosyncratic style to a new generation. Like Takahashi's *Noby Noby Boy* before it, *Wattam* almost defies description: you take control of the Mayor, a cuboid character who makes friends by detonating the explosives he keeps hidden under his hat. For some reason, blowing up friends – an assortment of rocks, cakes, vegetables, and other curious beings – makes them incredibly happy, and doing so allows the Mayor to unlock more friends, each with their own unique abilities. A seed character can plant itself in the ground and grow into a gigantic tree, which can, in turn, inhale other characters and 'poop' them out of its branches as sentient fruit. In other words, *Wattam* sounds more like a surreal sandpit than a game, but Takahashi's also said in interviews that there's a story and greater purpose to *Wattam* that he's keeping tucked under his hat for now.

Shadows of Doubt ↘

Described by designer Cole Jefferies as "a detective stealth game somewhere between *Deus Ex* and *Sherlock Holmes: Consulting Detective*", *Shadows of Doubt* takes place in a neo-noir city made from millions of tiny voxels. As a private investigator, it's up to you to quietly roam the city's streets and dimly-lit rooms in search of a serial killer – and because the game's world is procedurally generated, any one of its AI citizens could be the culprit.



↖ Structure

A multi-directional shooter hailing from Russia, *Structure*'s premise sets it apart from its 2D brethren: you control a small, spherical robot exploring an artificially intelligent mass that has formed around a remote planet. There are enemies to blast, as you'd expect, but there are also locked doors to hack into, and what appear to be physics-based puzzles to solve. The distinctive, silhouette-based visuals also look great.





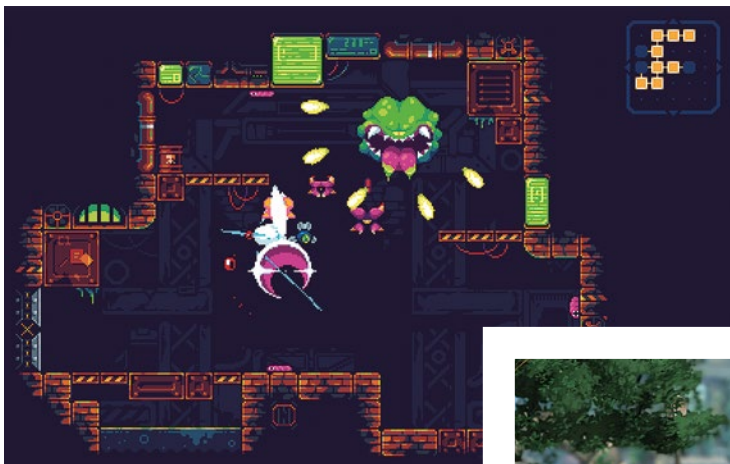
Unbound: Worlds Apart ↗

A once leafy world has turned into a scorched wasteland in this terrific-looking 2D platformer where, in a riff on games like *Giana Sisters: Twisted Dreams*, it's possible to switch between two versions of

the same space in order to solve puzzles and navigate hazards. *Unbound* recently soared past its Kickstarter goal, so expect to see lots more about this one ahead of its launch in May 2020.

Blasphemous ↘

Team 17 recently announced this 2D Metroidvania, in which a sword-wielding warrior named The Penitent One roams a grim fantasy landscape relieving monsters of their heads and arms. Expect huge amounts of gore, and some outlandish boss battles: one scene in the reveal trailer sees the pointy-hatted hero fight a gigantic, blindfolded baby. And because the game has a deadly serious, death metal tone, the baby is named Expósito, Scion Of Abjuration.



Baldo ↘

On the face of it, Italian developer NAPS Team have managed to craft a Japanese-inspired action RPG in the vein of Level-5's *Ni no Kuni* – no mean feat, considering the latter game was made in collaboration with the mighty Studio Ghibli. *Baldo* is a fantasy adventure that takes in giant monsters lumbering through painterly forests, villages full of friendly, whimsical characters, and youthful heroes pressing switches to solve puzzles. Whether it'll offer its own individual twist on the genre remains to be seen, but what the studio has produced so far certainly looks pretty.

ScourgeBringer ↗

Anyone who feels like they've seen enough pixel-art ninja games over the past few months should look away now; for the rest of us, *ScourgeBringer* looks like another promising action-platformer. With its fast movement, tough-looking level designs, and constant player deaths, *ScourgeBringer* provides a caffeine-rush fusion of *Ninja Gaiden*, *Super Meat Boy*, and *Spelunky*'s procedurally generated caverns of doom.





Making the jump to indie

Polygon Treehouse tell us what it's like to go from a major triple-A studio to making games as a two-man indie team

WRITTEN BY
GARETH DUTTON

◀ The titular Rōki is a monster who steals Lars, Tove's brother, which starts Tove on her reluctant adventure to get him back.

▼ The devs inside the Polygon Treehouse: Alex Kanaris-Sotiriou and Tom Jones.

W

hen Guerrilla Cambridge shut down in January 2017, it had been running for 19 years. Among its staff were veterans Alex Kanaris-Sotiriou and

Tom Jones, who at the time were working on *RIGS*, a first-person shooter in development for PlayStation VR. The pair had been at the studio for over a decade, and the shutdown came as quite a shock. Today, though, Alex and Tom are keen to take positives from the experience.

"It was tough, we'd been there a long time, but equally you have to be philosophical about these things," Tom argues. It left the pair with a decision to make: secure a job with another triple-A dev and continue working in a large-scale studio, or make the jump and go indie. Within weeks of Guerrilla Cambridge's closure, Alex and Tom had made their decision.

"We saw it as an opportunity to do something else that we might not otherwise have made the leap to do," continues Tom. "We were afforded the opportunity to go and do something new. We both had opportunities to go and work in studios again, but we [both] felt like doing something in the indie scene – essentially doing something where we could be more selfish in what we were making."

CONSTRAINTS

Polygon Treehouse's story is one that's becoming increasingly common: talented, enthusiastic developers leave the triple-A scene in order to join or start their own indie companies. An indie setup affords its members a greater creative freedom seldom possible in larger studios. Smaller teams often lack the resources enjoyed by larger studios, but these limitations positively encourage creative solutions.

"Working for Sony gives you access to big teams and big visions and money," explains Tom, "but it also brings a lot of challenges, such as constraints in game design. Suddenly, when you're two guys working from home making a game that you've come up with, it's incredibly different. That's very liberating, but also very challenging.

"At Sony, we made big, blockbuster games with big teams, and that required a lot of effort, especially with regard to art assets. That's not to say that we hated it – we enjoyed those projects – but we don't have a team of 20 artists or an outsource department, so we can't make a game like that. ➔





SMALL IS BEAUTIFUL

The creative agility of a smaller team can better accommodate the ebb and flow of the artistic process, and it's where a lot of innovation stems from in the indie scene. It's a freedom unique to a small team structure.

"One thing that is very quick is the decision-making process," explains Alex. "We're able to move exceptionally quickly in terms of creating characters and getting things up and running. That's one of the nice things about being more hands-on with the work – when you have to make those calls, it gives you a lot of momentum to spring forward."

"We wanted to make something that felt epic, but we can't do it in that triple-A art style – it's just physically and financially impossible. Having that constraint from the start allowed us to find creative solutions, hence why we went for something clean and graphical and colourful, with no normal mapping or sophisticated lighting. [*Röki*] is really focused and something we can turn around very quickly, as well as being something that is very consistent throughout."

For Polygon Treehouse, the advantage of being away from the triple-A environment is that you don't have those same professional pressures coming from higher up. This is especially important in a creative industry.

A game with a huge budget is a clunking machine; once the design is in place, there's only so much steering that can be done once it has begun its journey. And the machine doesn't stop – if you just aren't feeling creative some days, it's not like you can pop into the boss's office and say 'You know what? I'm not feeling it today, I'll see you tomorrow.'

"The freedom to be able to not try and brute-force the creative process is really powerful," says Alex, "because you can't brute-force the creative process; it's one of those things where you will just bang your head against a brick wall.

"That's a really powerful thing, to be able to say, 'I'm just going to stop. I'm going to stop, go for a walk, or I'll take the rest of the day and come at it tomorrow.'"

BUILDING RÖKI'S WORLD

Alex and Tom's first game under their new studio banner, Polygon Treehouse, is *Röki* – an adventure game with a setting inspired by Scandinavian folklore. You take on the role of a young girl named Tove, and delve into a magical fantasy world filled with mystery and uncertainty – quite the departure from the fast-paced, twitch-based virtual reality world of *RIGS*. It almost feels as though the pair were compelled to create something that was the polar opposite of their previous work – although Alex ensures me that such a creative left-turn wasn't necessarily a conscious decision.

"We did want to make something that

was non-violent," he agrees, "and something that was more personal and had more emotion in it – something that told

"We wanted to make something epic, but we can't do it in that triple-A art style"

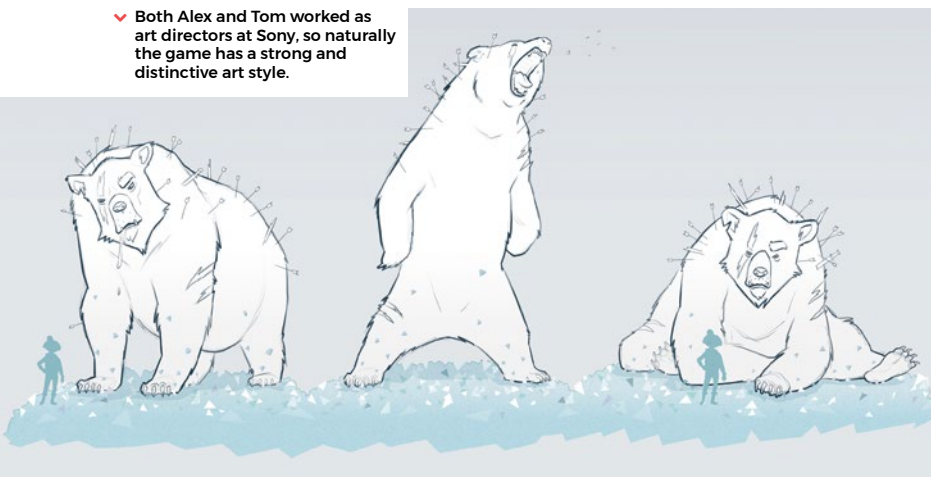
more of a story. In terms of design principles and visual design, even though the visuals are really different, a lot of the same principles still apply. You still have silhouette and readability of characters, frequency of detail, composition, and so on. Even if the visual styles are very different, the building blocks and considerations are the same. I think that it's those principles that have made the art style of *Röki* so appealing."

BREAKING THE MOULD

During our conversation with Polygon Treehouse, our chatter drifted from the relative serenity of *Röki* to the violence more commonly seen in triple-A titles, such as the forthcoming *The Last of Us Part II*. Unusually, Naughty Dog's forthcoming sequel will – at least according to a Kotaku interview with director Neil Druckmann – attempt to deal realistically with the consequences of violence. In short, the bloodshed in *The Last of Us Part II* is intended to give the player pause rather than exhilarate. But in the context of a triple-A game designed to sell millions of copies, is such an innovation even possible?

"The gameplay trailer [for *The Last of Us Part II*], felt like the visuals had gone so far from being a faceless baddie," Alex argues. "Not just in how well they were modelled and textured but also how well they were animated; they felt so alive that when someone plunged a machete into someone's neck... it felt kind of wrong."

✓ Both Alex and Tom worked as art directors at Sony, so naturally the game has a strong and distinctive art style.





▲ The environments in *Röki* draw heavily from Scandinavian folklore. The intricate detail and sense of place are inspired by the team's love of classic point-and-click adventures.

"I think it will be interesting to see whether Naughty Dog are trying to achieve the goal with *The Last of Us [Part II]* of making something that repulses people. If they do, then that's really good and potentially really powerful. People who are more introspective might look at that and think, 'Wow, this game is really challenging me.' My cynical head, however, wonders how many people will enjoy it solely for the ultra-violence."

THE FUTURE OF INDIE

Players and developers alike are being drawn to indie games in increasing numbers. The Nintendo Switch has been a hotbed of indie games by itself; titles are being developed specifically for the Switch, while older titles, previously only available via Steam, have had new life breathed into them for the console.

"The indie scene is an opportunity for fresher experiences that aren't governed by the same expectations as a triple-A, open-world game, where you can spend 100 hours doing stuff," enthuses Tom. "You can make a two-hour-long game, it can just be side-scrolling or whatever, and that's fun and liberating. There's a real freshness to that scene because it allows incredibly creative people to do something that means a lot to them and get it out there."



▲ The pair created an emote system for the characters, which allows for a broad range of emotions to be expressed, depending on the situation.

"Unity is free for a certain team size, Photoshop is ten pounds a month, Maya you can get a light version for 30 pounds a month," he continues. "So it's not that much money at all compared to when we were at uni, and you had to buy licenses that cost thousands of pounds in total. It's crazy how now you can just start making things and put it out there. Even if you're still at school, you can put stuff out there and get feedback."

For years, small-scale, low-budget indie games have brought us innovation and creativity, simultaneously hindered and inspired by financial constraints. But we're also seeing the rise of the high-end indie studio, started by passionate creatives who cut their teeth on bigger, more mainstream games. With *Röki*, Polygon Treehouse is bringing together a mix of industry experience, budget, and creativity to forge a beautiful-looking game that dares to stray from the beaten path. 🗨



▲ Tom worked on a variety of projects with Sony, including *Killzone 1* and *2*, *Heavenly Sword* and, more recently, VR title *RIGS*.



▲ Alex started working with Sony as an artist as soon as he left university, and had been with them for over a decade.

OUT OF THE BOX

The return of the escape room

A new generation of games prove there's still life hiding in an oft-overlooked genre

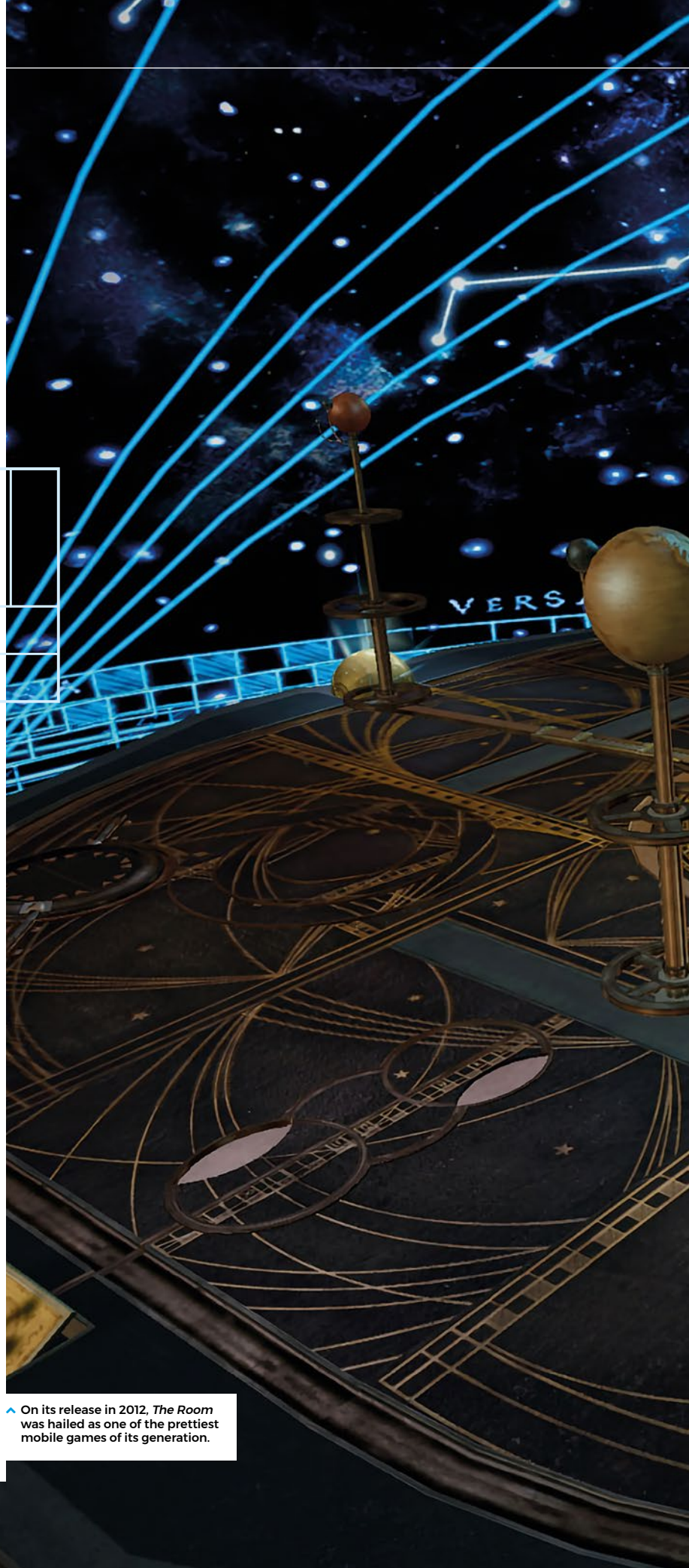
WRITTEN BY
STEVEN T. WRIGHT

V

iewed from the lofty vantage of 2019, the cramped red chamber that helped kickstart a micro-genre doesn't exactly ring out as a timeless classic.

Rather, Toshimitsu Takagi's *Crimson Room*, made in 2004, comes off as a dusty relic of a forgotten era of browser-based gaming, a compact puzzle defined by tedious pixel-sniping and the twisted reasoning that we might charitably term 'walkthrough logic.' While the game might not hold up to modern tastes, it still stands as a defining example of one of the most popular forms of the puzzle game, the so-called 'escape room.' But while the term might give readers flashbacks to entire weekends demolished by Flash portals like Newgrounds and Armor Games, over the past decade, some of these tiny puzzlers have managed to make the jump from your browser window to your pocket – and they've never been better.

^ On its release in 2012, *The Room* was hailed as one of the prettiest mobile games of its generation.





These animal guests are the denizens of the titular *Rusty Lake Hotel*, the original premium entry in the series.



Search 'escape room' on your mobile device's app store, and you're assaulted with a barrage of derivative mediocre-ware with names like *Action Escape* and *Room Escape Game*.

While these free or low-cost examples might scratch a riddle addict's itch in the same way that a decent crossword puzzle might, the allure of a truly great escape room game comes both in its puzzles, and also the story it tries to tell.

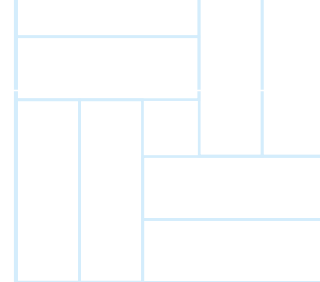
Bored office workers didn't flock to *Crimson Room* or its sequels because they adored clicking a curtain five times to make a key fall out; they stuck with it because they adored the bizarre iconography, the keen sense of foreboding.

That's the essential element – the delicious mystery of why you're trapped inside the room in the first place – that a new wave of 'premium' escape room games have deployed to great effect. And no series has pulled this off quite as well as *Rusty Lake*.

A LYNCHMAN LAKE

Created by development partners Robin Ras and Maarten Looise, *Rusty Lake* began life in 2015 as the free *Cube Escape* series, hosted on the same Flash portals that first made the form famous.

The early games in the series, like *The Lake*, offer up a similar style to the likes of *MOTAS* and *Crimson Room*, forcing the player to ➤



“We sometimes joke that we’re making our version of something like *Twin Peaks*”

scour a space for tools and keys and MacGyver them together to solve increasingly Byzantine conundrums. The difference comes in the quaint, small-town horror trappings that Ras and Looise choose to soak their puzzles in, which they say comes largely from the filmography of beloved auteur David Lynch. *Crimson Room* tasks you with putting together a projection box that reveals the location of a hidden safe; *Rusty Lake* has you pull a corpse out of a pond with a fishing rod, only to dig inside to get an artefact you need to summon a demon.

“We sometimes joke that we’re making our version of something like [Lynch TV series] *Twin Peaks*,” says Looise. “We liked the existing escape room games on the Flash portals, like *Crimson Room*, but we thought the gameplay could use a bit more storyline and atmosphere. From the start, *Rusty Lake* was always meant to be a world where many stories could take place, a bit like *Twin Peaks*.”

From the beginning, both Ras and Looise felt that the explosion of cheap, subpar games had savaged the reputation of the escape room genre such that they wanted to do something

FLASH FORWARD

Many Flash developers had difficulties translating their success onto the mobile platform, but Ras and Looise say that it was far easier than they expected. “I’m always surprised that some Flash developers never made the transition,” Looise says. “It’s all about marketing. We always made sure to include reasons to come to our website in the game, and we would announce the mobile versions on there. Technically, the struggle isn’t getting the game on the phone, it’s all about making sure your users know that it’s an option.”

that clearly stood apart from the pack. In order to build interest in their games, rather than simply building sequel after sequel, they decided to build a fictional universe and populate it with a cast of distinct characters, releasing the first two games in the series just days apart in April 2015. And they made sure to communicate that intricate interconnectedness to players from the beginning: entering a code from the second entry into a safe in *The Lake* allows you to view an alternate ending.

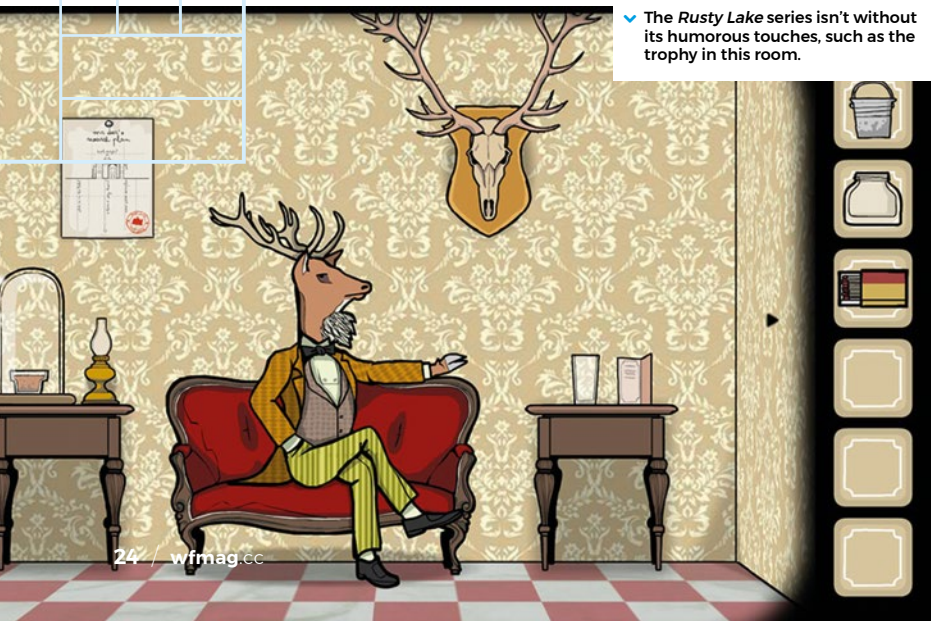
After releasing four more games in 2015 to a growing fanbase, the duo decided to ask their community if they were willing to pay for a more ‘premium’ experience. As Looise recalls, it went far better than they expected. “They basically said, ‘Well, yeah, of course. You’ve given us six games for free already,’” he says, laughing. “With that, we were able to build *Hotel*, which had five rooms, and was a way more complicated thing. And we’ve kept up the same pace for a while now. A few free games, then a premium. It’s worked out well.”

A CHINESE BOX

While the team behind *Rusty Lake* takes the mantle of the escape room with pride, not all developers are quite as eager to hold it aloft. Take Fireproof Games, the studio behind the much-acclaimed *The Room* series, which takes the central ‘confined space’ concept and augments it with a highly tactile interface and impressive production design. Director and co-founder Barry Meade agrees that the series’ simple name might lead players to expect to jiggle keys or demolish doorknobs as they carve their way out of a prison-like space, but he says Fireproof’s series isn’t so much about breaking out of a place, it’s about breaking into an object, like a safe. “We didn’t set out to make an escape-the-room game at all,” he says. “The original idea was to create a game about breaking into unusual boxes by manipulating hidden switches and locks... I think the name makes people think it’s more of an escape-the-room game than it is. But we do put a lot of effort into ensuring the player is in a cool environment that they get to navigate, too.”

It’s easy to look at the gleaming mahogany wood panelling and hear the satisfying ‘click’ of futzing with one of its puzzle boxes and conclude that, aesthetically, *The Room* was a few steps ahead of the infinite runners and pseudo-

✓ The *Rusty Lake* series isn’t without its humorous touches, such as the trophy in this room.





^ Some of the *Rusty Lake* games are less connected than others. *Rusty Lake Paradise* is considered standalone, and takes place on a remote island.

GREAT LAKES

Now that the *Rusty Lake* universe has expanded to include short films, and with the scale of the endeavour tumbling ever-outward, the twosome say that keeping the lore of the games straight can prove daunting as well. "A lot of stuff has happened, and we've made a lot of games," says Looise, laughing. "We sometimes have to check our own Wiki. We don't want to get anything wrong. We go to conferences, and they ask us questions. Sometimes it feels like we don't even know the answers."

gambling games that made up the mobile space in 2012. But for Meade, the unique cachet the puzzle genre enjoys among those who aren't necessarily the most gaming-inclined isn't a result of any developer or game – it's an innate trait that those of a curious bent all share. He compares *The Room's* riddles to other creative pursuits, like chess, or board games, or even toys like a Rubik's Cube. "Solving a puzzle is pure thought – you can zone out to them. I think there's a Zen quality to a good puzzle, where the world drops away, and all you have is the problem floating in front of your face. I think people enjoy that, the comfort of it. And of course the solving of the problem – people enjoy getting things right after an initial struggle where you think, 'How the hell do I do this?'"

SETTING A PACE

Ras and Looise concur with this approach. To them, the most difficult part of making a new entry in their series is trying to tune the difficulty. As they rocketed past a dozen games, Ras admits that the tendency is towards complexity, to keep long-term fans who have mastered all the previous games sufficiently challenged. However, since the duo also feel that accessibility is a cornerstone of the series, they have to scale back some of the sprawling scope of the premium entries – which can sometimes include out-of-the-box concepts, such as time travel – for the free games, as they're intended to be gateways to the entirety of the series.

"We just held a birthday party in our office for some kids," Ras says. "We showed them some of the puzzles in our new game, *Paradox*, and we were really impressed by how quickly they were able to solve some of them, then run into a wall,

and then solve some more. They're the audience we like to consider when making our games, since we want everyone to be able to play them."

"There are pacing issues in a lot of puzzle games," adds Looise. "In our games, we always try to give you that sense of breakthrough every ten minutes or so, so you feel like you're making progress. We don't want people to just click everywhere, but if that's what it takes, then that's sometimes what happens."

After spending five years promoting what they call "traditional" escape room gameplay, Ras and Looise admit they're ready to experiment with some new mechanics. While they're quick to point out that these new games will still involve puzzles and escaping, they say they're prototyping new concepts that they're excited to implement in their next project. But while they love the point-and-click gameplay that won them a devoted fanbase, they say that, after 13 games, they feel like they've almost exhausted their bag of tricks at this point. Still, they think they've made quite a mark.

"When we started, we wanted to raise the reputation of escape room games, because we thought the potential for storytelling there was greater than what had been done," Looise says. "I feel like we've done that. With *Roots*, which was sort of a collection of all the cool things we've done in the games, I think it all came together. The next thing we make will still be a kind of escape room, but I'm excited to see what we can do next." ^W



^ Intricate contraptions controlled by a touchscreen are the signature hook of *The Room* series.

All Hail King Rob!



STEVE MCNEIL

Steve met a clever nerd, and he thinks you should know about him.

“Rob remains relatively unknown but, thanks to his most recent work, he’s approaching the point where it’ll be hard to ignore what he’s doing”

In one of the first ever *Go 8 Bit* live shows back in 2013, an unassuming nerd called Rob Sedgebeer approached me afterwards, and mentioned he’d created something we might like to use in the show. He’d built a version of *Pong* which the audience could control using their phones. It had been used once before on stage, at the brilliant Festival of the Spoken Nerd night run by Helen Arney, Steve Mould, and Matt Parker, but since then had sat on a shelf.

The way the game worked was, the audience connected to a closed WiFi network Rob rigged in the room, picked which team they were on, and then he beamed two buttons to their phone – an up arrow, and a down arrow. Each time the ball approached their paddle, they voted on which way it should go and, if the majority were correct, it whizzed into place and hit the ball. Democracy Pong, essentially.

Of course, as fans of democracy (‘the people’ famously never make bad decisions), we loved the idea and booked Rob in for the next show. He turned up early, set up his network, and waited – poised for the moment we would unleash this new tech upon the world. Then it didn’t work. That was awkward.

But Rob persisted. The second time, it worked with about six people. Then crashed. But, just like porridge, on the third attempt, it was *just*


right. Since then, Mr Sedgebeer has advanced the technology, adding more complex games, and even beaming full games to the audience – everything from the earliest retro games, to 3D first-person shooters and VR flight sims. All of this, without the need for the user to install an app – simply done in a browser, on any device, with any operating system.

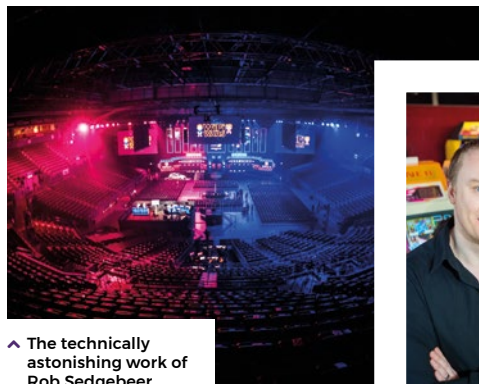
Despite having a residency at the Royal Institution (where he broke a Guinness World Record with his tech), Rob remains relatively unknown but, thanks to his most recent work, he’s approaching the point where it’ll be hard to ignore what he’s doing. Man of the Year 2020? You heard it here first.

For the last six months, Rob tirelessly rebuilt his code from the ground up, to allow it to run on cloud servers, and automatically scale itself up to accommodate larger numbers of connections. Previously, the most the show had comfortably worked with was hundreds, but Rob opened it up to all 8000 attendees at the ESL One *Dota 2* event at Arena Birmingham. And it worked.

But what does all this mean? Well, in the context of the above event, it means arena-scale multiplayer gaming events. But he’s also got the show working on Twitch, allowing people watching anywhere on the planet to compete with each other. He can even beam video of their gameplay back to the show as part of the broadcast.

Imagine a live, global broadcast, where millions watching can interact with the show, compete in games and quizzes, and literally *win the show*. As someone who works in TV, the notion of ‘appointment-to-view’ programming (i.e. shows people tune into for their premiere airing) is a big deal. So I’m very glad I met him.

Find out more about Rob’s work at wifiwars.co.uk or join in online at twitch.tv/king_rob_uk 



^ The technically astonishing work of Rob Sedgebeer.



Toolbox

The art, theory, and production of video games

28. CityCraft

Real-world materials to consider in your game's city

30. Power of Focus

Why game design is about doing fewer things better

32. Saving and loading

Making a save/load function in Unreal Engine 4

38. RPG Maker plugins

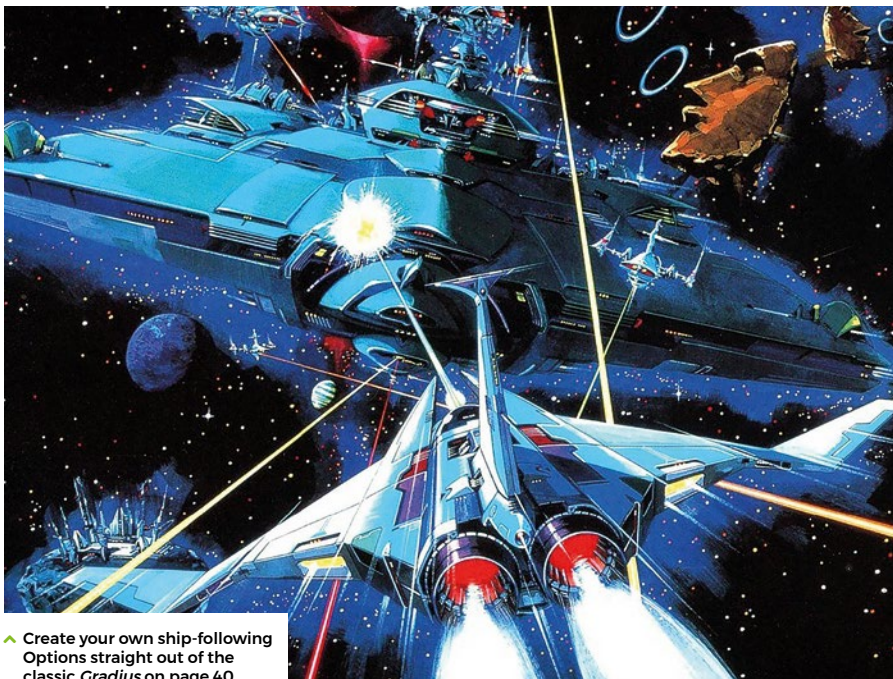
The free plugins you can use to make your game stand out

40. Source Code

Recreate the ship-following Options from Gradius

42. Directory

Game dev events coming up this summer



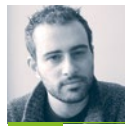
^ Create your own ship-following Options straight out of the classic *Gradius* on page 40.



^ The concrete, steel, and glass that make up a real city are an important consideration in your game's metropolis - see page 28.

CityCraft: The building blocks of virtual cities

From marble to steel, and brick to titanium, considering real-world materials is key to building a digital city



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

Monumental Stones

Architecture and materials aren't uniform across classes, or for all types of buildings. The richer a building's owner, or the greater the building's importance, the more expensive its construction, and thus the more impressive its materials. Stone houses were rarer than wooden ones during the Middle Ages, and usually signified their owner's status, whereas Rome's marble and stone public buildings stood out from the brick residences of the masses. More strikingly, the palaces and pyramids of the Pharaohs soared over the mud-brick buildings of ordinary mortals.

Cities are a physical and enduring manifestation of societies. Their walls, edifices, and infrastructure shape the flows of human activity, and comprise the foundation that allows civic life to flourish. Buildings aren't simply conjured out of thin air, so the materials we use to create them are an important factor, determining how our cities look, feel, smell, and even sound.

Construction materials are the city's building blocks, dictating its shape, colour, and temperature. These materials determine whether a city can easily burn, whether it will survive an earthquake, or even whether its roads will become slippery in the rain.

Unsurprisingly, then, building materials are another key element that we have to try and simulate when creating our game cities. Whether you're finalising a masonry texture for a medieval settlement, or building the heat-resistant window frames of a near-future, eco-friendly building, materials should always be a key consideration.

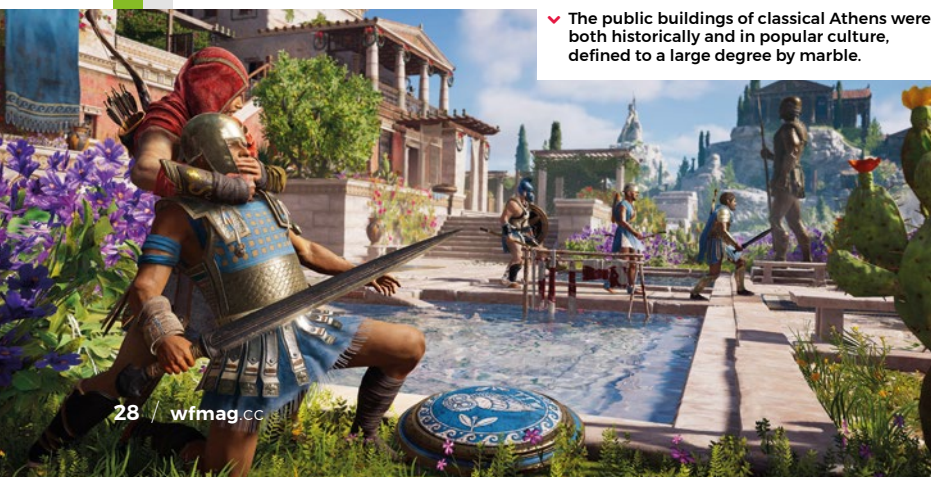
LOCATION AND TECH

The farther back into history we gaze, the less choice civilisations had when deciding what to build with. Not only was transporting the large quantities of wood, earth, or ore required for a settlement essentially impossible over long distances, but even the ability to mould metal or blow glass was a technological advance that took centuries to achieve. Location and technological capabilities, therefore, mattered in the creation of the built environment. The evolution of transportation effectively shortened distances, and, up to a point, liberated construction from geographical constraints, even if some early societies had already achieved such monuments as Stonehenge or the pyramids of Giza with limited technical means.

In the pre-modern era, local quarries and forests were often used to build towns and cities. Wooden houses and temples, straw roofs, rock walls, and mud huts were all common, and employed nearby materials that were easy to shape. The techniques used were varied, and constantly improved in a continuous dialogue with architectural styles and urban ideals. Major advances would eventually include the use of bricks and plaster, as well as better and faster stone cutting that allowed for walls, columns, and arches. The use of concrete in the Roman Empire was widespread, while mastering iron forging, insulation, and eventually, the reinforcement of concrete were crucial breakthroughs.

So, generally speaking, when deciding on the materials to use when creating your game city, you should always consider where and when its building would have taken place. What resources

✓ The public buildings of classical Athens were, both historically and in popular culture, defined to a large degree by marble.





▲ Glass and steel, the materials that shaped the 20th-century skyscraper, also provide Spider-Man with urban canyons to traverse.

would have been used? What tools were available? Were architecture and engineering developed around existing materials? How far could luxury materials be transported? Could all classes afford, say, stone, or was it a privilege of the elite? Did a nearby volcano provide the area with an abundance of volcanic rock to turn into plinths?

Brick, in all its varied guises, is a safe choice, as it's been widely available across the globe for thousands of years, and still used in modern buildings. Then again, historically speaking, epic architecture was routinely constructed from more durable materials, and often used resources considered exotic, expensive, and rare. Even today though, due to both transportation costs and local traditions, location still matters. White marble was used to construct public buildings in ancient Athens, but it's still a recognisable part of the city's architectural character. Similarly, Polynesian construction has widely used the same type of timber for almost a thousand years, and New York still likes to exploit the layer of dark granite on which it was built.

ARCHITECTURE AND PLANS

Architects, city designers, and urban planners have to take the availability of materials into consideration, and design their spaces accordingly. Just as brutalism is a style borne of the raw texture of concrete, it was precast concrete and steel that allowed for the curvaceous shape of the Sydney Opera House.

Complicated shapes also demand complicated calculations: the curves of Bilbao's Guggenheim Museum would have been all but impossible to design without the computing technology of

the 1960s. Materials, engineering technology, and knowledge have always been closely tied to each other. Wild, post-modern architecture demands light, durable materials and similarly strong mathematics.

If your virtual city and its construction capabilities allow for skyscrapers, concrete office buildings, massive stone pyramids, or brick railway stations, these must also be reflected in its wider planning to ensure they can function properly. Furthermore, the use and colours of your dominant

materials will determine how the city will sit in its environment. Will it be a yellow dried-brick town, barely discernible in the desert surrounding it? A stone village hiding on a mountain slope? A basalt castle dominating the green fields of the countryside? Or a row of white high-rises by the sea?

The choice of materials can also influence the microclimate of a city. Paving slabs, for example, are more heat-resistant than asphalt, while concrete slabs are more durable in extremely low temperatures. Streets and pavements are, after all, constructed from available materials, too. At the architectural level, an abundance of glass towers in a warm, sunny climate can have wider repercussions; air conditioning raises outside temperatures and energy demand. By contrast, Arabian and Persian architects traditionally used wind towers (or wind catchers) to cool the interiors of their buildings.

In short, thinking about the ways architects use materials like steel, glass, concrete, and brick in the real world will give your virtual city an added layer of richness and realism. 🌐

“Construction materials are the city's building blocks”



▲ Cobblestone has been in use for millennia, and is still relatively popular in the construction of Europe's pedestrian roads.

The Eternal Brick

Humans have been using bricks for over 9000 years, making them one of the oldest known building materials. The very first bricks (discovered around ancient Jericho) were made of mud and dried in the sun, and eventually improved by the Egyptians, who constructed theirs with clay and straw. The durable fired brick has been in existence since 3500 BC, but it was the Romans that introduced the mobile kiln, and the ancient Chinese who used bricks to pave roads. In Industrial Revolution Britain, Henry Clayton's brick machine was capable of producing around 25,000 bricks per day.

The power of focus

Game development is as much about what you don't put in as what you do put in, Reid writes



AUTHOR
REID SCHNEIDER

Reid is the producer of *Splinter Cell*, *Battlefield Vietnam*, *Army of Two*, *Batman: Arkham Origins*, and *Batman: Arkham Knight*. Follow him on Twitter: [@rws360](#)

In 2005, I headed to EA headquarters for an event known as a Franchise Review. These took place each quarter, with the aim being for executives to provide feedback on games that were currently in production. The tone of these reviews could vary wildly. Sometimes they were constructive, other times they could feel like the snark of *American Idol* crossed with the terror of a firing squad. I experienced both sides. On this specific day, I sat in on the presentation for EA's marquee title for that year: *The Godfather*.

The Godfather was being developed at EA's Redwood Shores. This studio, which would later become Visceral, was where the company was founded, so there was a ton of pride in the group. When EA obtained the *Godfather* license, its goal was to make a 'GTA killer': the management at Redwood had decided that the studio would become the world leader in open-world adventure games.

The executive producer took to the stage to start the presentation. It began with a 'target gameplay' video: a pre-rendered piece designed to show what the gameplay would eventually look like. After the footage ended, there was

nothing short of a standing ovation (in all fairness, it really was an impressive video). Next began a presentation, where features from *Grand Theft Auto III* were listed next to those from *The Godfather*. The meaning was clear: *The Godfather* would top all of them.

At one point, there was even mention of an 'SPB Ratio', which meant 'Stuff Per Block' – or the number of objects in the world the player could interact with during a play session. This wasn't a metric I'd ever heard before; that someone had calculated it was pretty wild nonetheless. In fairness to the team, they *needed* to sell the game in this way; the costs of building an open-world game, in the Bay Area, were astronomically high. It's also worth noting that there were some tremendously talented developers working on the game, and they had a vision. A big vision.

LOFTY GOALS

As is normal with game development, though, the challenge came down to execution, and realising all those big ideas. Their feature list not only had to match that of *GTA*, but exceed it. Unfortunately, the weight of that feature list, combined with the pressure to deliver the title in a given timeframe, meant that *The Godfather* would never achieve its lofty goals. While it spawned a sequel, EA would ultimately part ways with the license to focus on its own properties.

A few years earlier, while I was working at EA, I had the opportunity to work on a much more focused title that would ultimately go on to exceed sales expectations. I was working with DICE on *Battlefield Vietnam*, the follow-up to *Battlefield 1942*. It's important to note that this title was far less anticipated than a project like *The Godfather*, and was far smaller in scope. *Battlefield Vietnam* had originally started as a



▲ EA's *The Godfather*: alright, we guess, but not quite the 'GTA killer' its studio had envisioned.

▼ A focused experience, and all the better for it: *Battlefield Vietnam* was one of the series' high points.





^ **Battlefield Vietnam featured a number of firsts, including big green helicopters.**

'mod', but as it grew in traction and quality, EA realised it could be a full game on its own. Fortunately, the game was priced competitively, and not positioned as a full sequel to the ground-breaking *Battlefield 1942*.

While we were working on *Vietnam*, the team had ideas for how to make it bigger. Fortunately, as it was being developed at a satellite studio, DICE Canada, we were constrained by the number of people who could be developing it, and the budget that could be spent against it. This ultimately worked to our advantage, as it meant we needed to be super-focused, and *Battlefield Vietnam* could never become a 'kitchen sink' game – that is, a title where you throw in every feature known to be popular at the time. This meant no single player, no console support, and no big changes to the engine. This focused development was precisely what made the game successful: it was a game that knew exactly what it needed to be. That's not to say its development was simple or easy – it wasn't.

"It's better to do fewer things better than lots of things at an average level"

LESS IS MORE

Over the course of that project, we built a new foliage/tree system called Overgrowth, and brought helicopters (along with their crazy physics and controls) to the *Battlefield* series for the first time. Overgrowth was needed to create the feeling of being in the dense and claustrophobic jungles of Vietnam, with scores of other players. The feature was unknown, risky, and scary, but ultimately worked due to the talented developers that laboured on it. Ultimately, we were able to ship the game, without a marked increase in the PC minimum

spec required, and it went on to achieve both commercial and critical success.

The point, ultimately, is to always be focused when building your game. It's better to subscribe to the 'less is more' adage, and do fewer things better than lots of things at an average level. That being said, this is much more difficult than it sounds; it's easy to get excited about an idea that might not be critical to your project and

want to start working on it immediately. When this happens, try to think critically about whether or not this feature really matters, and more

importantly: does your team have the capability and time to fully realise it? It's way more painful to cut a feature after work has been invested in it than at its inception. In the end, games, like any entertainment project or investment, are an informed bet. Always be mindful that it doesn't matter what you did yesterday: all that matters is what you do next. ☺

Killing floor

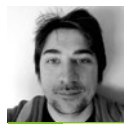
As a side note, anytime you hear a phrase like 'GTA killer', be warned. More often than not, the label's doomed to failure, as an established incumbent – like the boundary-pushing *Grand Theft Auto III* – has usually benefited from being the first of its kind. If you're going to 'kill' a rival game, your offering needs to be so far beyond the current champion that winning borders on impossible. As you've probably gathered, creating such a game isn't easy.

> **Battlefield Vietnam's updated engine proved so effective that a mod was released so that several Battlefield 1942 maps could run on it.**



Make a Save/Load function in Unreal Engine 4

Saving and loading data in your games needn't be a headache in Unreal Engine 4 with SaveGame



AUTHOR
RYAN SHAH

An avid developer with a strong passion for education, Ryan Shah moonlights as KITATUS – an education content creator for all things game development.

From reading data out of audio cassettes on a Commodore 64 to pushing your high score into the cloud, saving and loading data has long been a vital part of gaming.

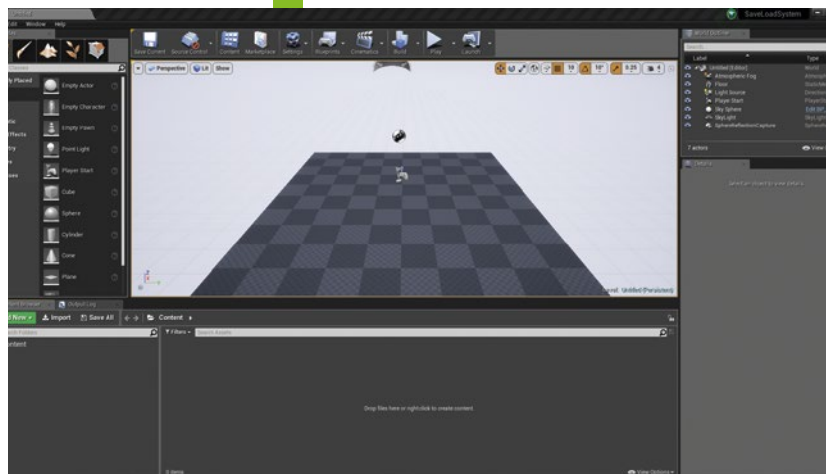
Most, if not all game engines available today offer a developer tool for saving and loading data – be it Unity's Persistence/PlayerPrefs system, or Unreal Engine's SaveGame class. Today, I'll show you just how simple it is to use Unreal Engine 4's SaveGame system, and how you can implement it in your own projects.

In Unreal Engine, everything is based on inheritance. Everything can be traced back up the family tree to a handful of core base classes that establish the dynasty that you use in your UE4 projects. You'll generally make edits to classes in one of two ways: by either overriding the existing class, or creating a child based upon the class you want to edit.

So how do we know which one to use? Due to the nature of the source code, you're usually fine to start by selecting a new child class as opposed to immediately overriding the behaviour of the existing class. This is because the classes have been designed with this inheritance in mind. When it comes to the SaveGame system, everything we need has been set up ready for us to use. There's no need to craft our own solution or override the existing one, as it has been designed to be fully exploited by any children created from it.

In fact, the SaveGame class was designed to be versatile enough to have children in not only C++ but also the Blueprints Visual Scripting system. I want to show you how to use the system in both C++ and Blueprints, so you're not limited in your options when it comes to using them. We'll start with the Blueprint implementation, and then move over to the C++ variant once we understand how it all works.

▼ This window of Unreal Engine is the hub for everything you'll be doing in the engine. The viewport is your window to the in-game world.



GETTING STARTED

First, we're going to need a new Blueprint. You can create a new Blueprint by heading into the Content Browser (usually found at the bottom of the main view of our project) and selecting the green Add New button that sits on the top left of the browser. From here, select Blueprint Class.

You're then going to be asked what parent class you'd like. This is where the inheritance we talked about earlier comes into play. Keen eyes will notice that there's no SaveGame parent class immediately visible in this window. To find the SaveGame class, use the search field to find the base class in the source of the engine. Once you've found SaveGame, select it and press OK to generate a child of the SaveGame class.

And that's it! Thanks for reading. All right, it might not be quite that simple, but we now have a fully functioning save and load system in our project. This class, when asked to, will store and load data from the player's hard drive (usually in the **Appdata/Local** folder on PC, or the **/ApplicationSupport** folder on Mac). To better understand what this class is doing, we need to feed it some data to see it in action.

Go ahead and double-click the created asset to load into the Blueprint editor. In here, we're going to need to add some variables. The variable list is usually visible on the left-hand side of the Blueprint editor if you're in the Full Blueprint View, or in the list in the compact view. For this example, we'll create a Vector, Bool and Int. Go ahead and create those now by pressing the + icon in the Variable section of this Blueprint. Once the variable has been created and named, use the Variable Type dropdown menu to select the correct variable types. For this example, I'll be naming these variables as follows:

Vector = PlayerLocation
Bool = isPlayerDead
Int = PlayerLives

Once your variables are created, you can go ahead and close this Blueprint.

We want to test the functionality of our created Blueprint to ensure it can save and load the data correctly. To not overcomplicate things for this tutorial, it's best to put our test in the Level Blueprint. Alternatively, you can create a

Blueprint Actor (as long as you remember to place it into the scene), but I'll be using the Level Blueprint for the purpose of this guide.

GOING DEEPER

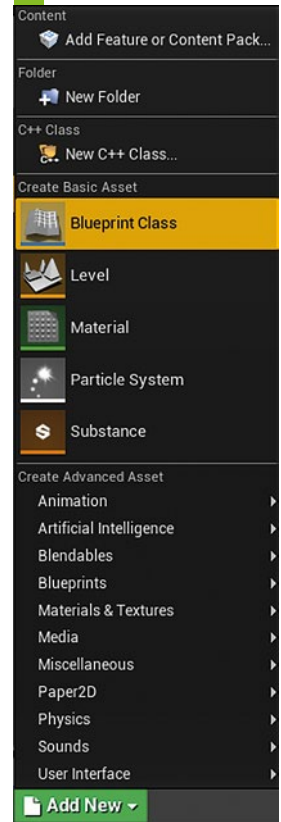
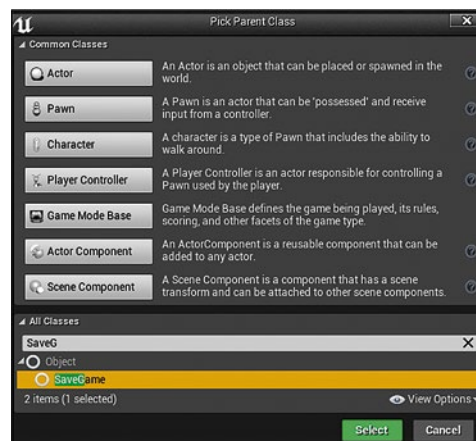
To head into the Level Blueprint, you'll see a Blueprints button above the viewport on the main engine screen. Click this button, and a dropdown will appear with the option Open Level Blueprint. Select this to open the Level Blueprint.

Take a look at **Figure 1**, and copy the Blueprint into either your Level Blueprint or the custom Blueprint Actor you created.

Let's quickly go over what this code is doing. We're using Begin Play, which means our code will fire as soon as whatever the code

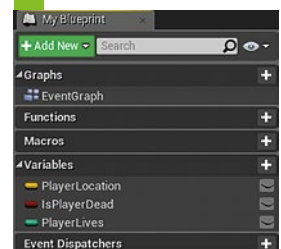
is attached to exists within the world. We are checking DOES SAVE GAME EXIST with the Slot Name of PLAYERVARS and a User Index of 0. What this is doing is checking "Does PLAYERVARS exist for player 0?".

The user index is used for multiple versions of the save file. This is mainly used in titles that have multiple save slots, such as a *Legend of Zelda* or something similar. PLAYERVARS is the name of a variant of our SaveGame system. That's right – it's yet another child. We don't directly access this version of our system, but instead, use functions to save it and load it indirectly. This is so we don't do anything Unreal Engine doesn't want us to do with this spin-off of our class. ➔

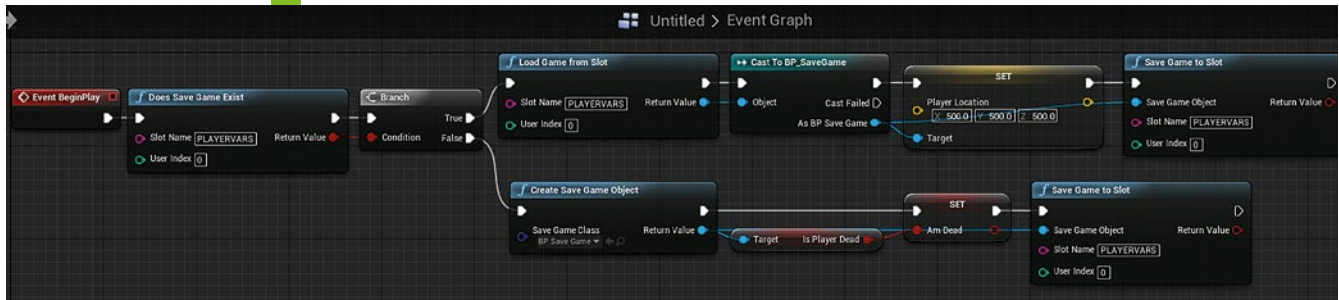


▲ There are a number of objects and classes you can create with the Add New menu. Alternatively, you can right-click in the Content Browser.

▼ **Figure 1:** This is the code we're going to create to deal with saving and loading in the Level Blueprint.



◀ Be sure to select the class you want to inherit from if you can't see the green Select button.



▲ Load the SaveGame, cast to our specific version, get and set your variables, and then save your changes.

Asking the question 'Does X exist' is going to return a true or false value, so the branch is put in place to hear the answer to that question. From here, we act based on if the SaveGame exists or if it doesn't. Let's start with if it does.

From the TRUE output of the Branch, we are using the Load Save From Slot node to load the version of the SaveGame we need, using the same inputs we used when we were trying to see if this existed in the first place. Because of how the Blueprint nodes were originally created, this node knows the data to load but doesn't really need to know the parent class of the data, and so puts the data

in a generic SaveGame object (this is a fancy way of saying the parent of our created SaveGame system).

We need the data back because we need to talk to our created

SaveGame directly, so drag from the output pin of the Load Game node, and in the search field that appears, type in 'Cast'. Select the Cast that corresponds to your created Save Game; for example, 'Cast to MyAwesomeSaveGame'.

From the output we now have, we have the power to either update the data via SET nodes (Saving) or using the data via GET nodes (Loading). To do this, drag out from the output pin and type 'Get' (or 'Set') with the name of the variable you want to use. From here, you can do whatever you like with the data.

Did you use SET nodes to update the data in the SaveGame? What you actually did was update a temporary copy of your SaveGame.

This is for optimisation and safety reasons.

All this means is that when we're ready, we have to send our changes to the actual SaveGame that is going to be saved to the hard drive.

In order to achieve this, all we have to do is head back to the output pin of the cast we created and type in 'Save' in order to save the data. You'll notice it's time to name the slot name and user index we'll use the next time we load this blueprint.

Fantastic: now we can load existing data, change, and save it! We're three-quarters of the way there. Now for the missing piece of the

puzzle: for us to be able to load data, the data needs to be created in the first place. At the end of our 'Load' code (which is used to update the existing values), we create the data. This is the marker that lets

Unreal Engine know, 'Hey, we're parking in this location. If somebody asks for us - this is where we'll be.'

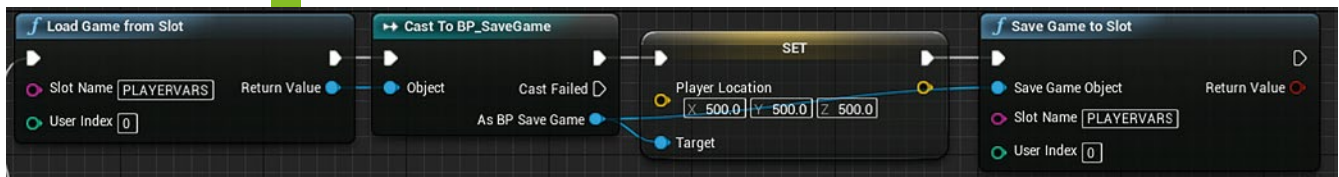
We'll fix this by heading all the way back to the FALSE output of the Branch. If you recall, this Branch is asking 'Does the save game exist?'; what we're about to add is a node for when the answer is false (or 'no, it doesn't exist' in pseudo-code). There's a node in Blueprints called Create Save Game, so we're going to use that. The node asks for a base class to base the created SaveGame on, so plug the SaveGame class we made into this node. From here, we need to set the slot and user index just like we've done twice now - so go ahead and do that.

“Fantastic: now we can load existing data, change, and save it. We're three-quarters of the way there”

ADDITIONS

You can add macros, functions, and events to SaveGame blueprints to further extend their functionality. Want to only allow saves under certain conditions? How about create a dynamic struct (a collection of different types of data) and save it on the fly? All of this is possible thanks to the versatility of the SaveGame system. You can even implement interfaces (a function that can be triggered elsewhere in your code) and dispatchers (which fires interfaces somewhere else in your code) for even more functionality.

▼ Create the SaveGame, get and set the variables you need, and then save this version to a slot of your choosing.





It's possible to update your local variables when creating the SaveGame object. When a SaveGame object is called, it will output the value currently stored. If this is a new SaveGame instance, it will use the default values supplied when you put together the SaveGame class.

Once that's done, we can immediately start using GET and SET from our newly created SaveGame, just like we did in the 'Load Game' part of our code.

Once you've made the changes to the variables contained within, just like with the Load Game code, drag from the output of the SaveGame object (which isn't in a cast this time, it's immediately available) and Save the data to a slot. Congratulations: you've written a fully functioning Save/Load system!

SAVE/LOAD IN C++

Sometimes you'll want even tighter control over your code, or you might want the faster compile times that come with C++ over Blueprints. Let's replicate what we just created, but in C++. If you want to attempt this, please ensure you have Visual Studio and the required Unreal Engine 4 prerequisites installed. To get started, we need to go to the main window of our Unreal Engine project. In the top left, select File > Add C++ to Project.

You have two choices here – you could create a Blank C++ class (which doesn't inherit anything by default) or you can search for the SaveGame system. For this guide, we're going to select None, but you're more than welcome to select whichever one suits your coding style. Name our new class whatever you like (I'm naming mine cMyAwesomeSave) and press the Create button.

You'll have to wait a few moments whilst Visual Studio compiles the new code.

We're first going to deal with the header file. For those new to C++, this file is home to all the functions and class declarations we want to use in our CPP file. C++ follows an 'Include what you use' mentality, so it's good practice to include anything you're going to need within your header file. Find the existing **include** in your H file (Line #5, **"#include "CoreMinimal.h"**), and on line 6, add **#include "GameFramework/SaveGame.h"**.

This is so we have access to all the SaveGame functions, macros, and variables in a similar fashion to how our Blueprint version worked. I will now give you a code snippet showing how your header file should look, so go ahead and copy it now – making changes based on your needs of the SaveGame system.

```
#pragmaonce

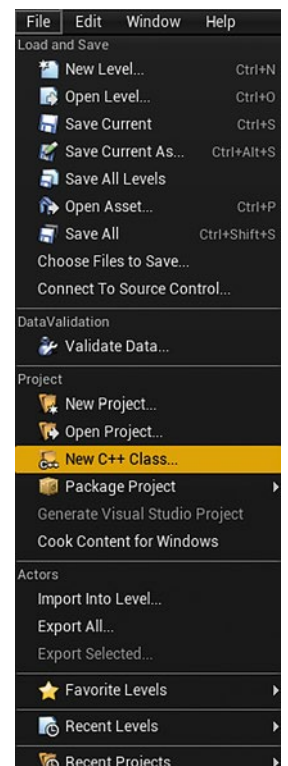
#include"CoreMinimal.h"
#include"GameFramework/SaveGame.h"
#include"cMyAwesomeSave.generated.h"

UCLASS() // Include the UCLASS macro to tell
the engine that this extends a base class and
cannot be directly instantiated in the world: it
must belong to something.
```

SAVING SPACE

Did you know that you can make casts 'Pure' in Blueprints? If you know for sure that the cast you're using will always be correct and that you won't need the 'Cast Failed' output, right-clicking the Cast node and selecting 'Convert to Pure Cast' will turn the node into a pure function. This saves a nice little bit of room in your Blueprint, since it doesn't require its own execution pins.

New C++ Class is found in the File Menu at the top of the main editor view.



Name Your New Class

Enter a name for your new class. Class names may only contain alphanumeric characters, and may not contain a space. When you click the "Create" button below, a header (.h) file and a source (.cpp) file will be made using this name.

Name: SaveLoadSystem (Runtime) Public Private

Path: Choose Folder

Header File: C:/Users/conta/Documents/Unreal Projects/SaveLoadSystem/Source/SaveLoadSystem/Public/cMyAwesomeSave.h

Source File: C:/Users/conta/Documents/Unreal Projects/SaveLoadSystem/Source/SaveLoadSystem/Private/cMyAwesomeSave.cpp

You can press the Public or Private button to change the location of your generated C++ to follow C++ standards better.

> The naming screen allows you to set the location of your newly created files. Most Unreal projects follow the Public / Private directory coding standard by default, which can be achieved by pressing the Public / Private buttons.

Name Your New Actor

Enter a name for your new class. Class names may only contain alphanumeric characters, and may not contain a space. When you click the "Create" button below, a header (.h) file and a source (.cpp) file will be made using this name.

Name	MyAwesomeActor	SaveLoadSystem (Runtime)	Public Private
Path	C:/Users/conta/Documents/Unreal Projects/SaveLoadSystem/Source/SaveLoadSystem/		Choose Folder
Header File	C:/Users/conta/Documents/Unreal Projects/SaveLoadSystem/Source/SaveLoadSystem/Public/MyAwesomeActor.h		
Source File	C:/Users/conta/Documents/Unreal Projects/SaveLoadSystem/Source/SaveLoadSystem/Private/MyAwesomeActor.cpp		

```
class [YOURPROJECTNAME]_API UMyAwesomeSave :
public USaveGame // Replace [YOURPROJECTNAME]
with your project name (removing the [ and ].
// The : public USaveGame at the ends tells
the engine that we're inheriting from
USaveGame.
{
    GENERATED_BODY() // Tells the engine
to inject the additional functions and type
definitions it needs to understand our code.

public: // This means that variables and
functions listed below are visible to all
classes and not hidden away.

    UPROPERTY(EditAnywhere,
BlueprintReadWrite) // Adding
UPROPERTY(EditAnywhere, BlueprintReadWrite)
tells the compiler that the next line is
variable that can be seen and edited even in
Blueprints.
        FVectorsgPlayerLocation;

        UPROPERTY(EditAnywhere,
BlueprintReadWrite)
        boolsgIsPlayerDead;

        UPROPERTY(EditAnywhere,
BlueprintReadWrite)
        uint&sgPlayerLives;

        UPROPERTY(EditAnywhere,
BlueprintReadWrite)
        FStringSaveSlotName; // Both
SaveSlotName and UserIndex don't use the sg
prefix. These are because they are required to
```

```
save our game to a slot and to a user index.
        // The above variables are optional
but both SaveSlotName and UserIndex are not.
        UPROPERTY(VisibleAnywhere)
        uint8UserIndex;

        UMyAwesomeSave(); // This is our
constructor so we can pass default values when
this is created.
};
```

With the code provided, you'll have to replace [YOURPROJECTNAME] with your project name in capitals, so in my case, class [YOURPROJECTNAME]_API would read class SAVELOADSYSTEM_API. You'll also have to replace the cMyAwesomeSave references with whatever you named your class on creation. Note that U appended to the start of cMyAwesomeSave. This tells the compiler that this is not an actor class.

Now let's head into the CPP file.

```
#include "cMyAwesomeSave.h"

// This is the constructor, which is fired
when this is created.
UMyAwesomeSave::UMyAwesomeSave()
{
    SaveSlotName=TEXT("ChangeMe");
    UserIndex = 0;
}
```

We only need to set the constructor here, to use the correct SaveSlotName and UserIndex, just like our Blueprint nodes for saving and loading did. You'll see that all we're doing here is setting default values that we can edit once we load up our SaveGame anywhere in our project. If

you wanted, you could replace your Blueprint version of SaveGame with this created C++ version – but let's go deeper, and save and load from another C++ class.

ACTING CLASSES

Go ahead and create another C++ class from within the engine. This time, I'm going to pick an Actor class, but you're more than welcome to pick anything you'd like to – we're only going to show you how to interface with your SaveGame system within another C++ class. Remember, though, that if you're following the code included here and you don't pick an Actor, your code will have to be slightly different in parts (you shouldn't inherit from an Actor if you're not an Actor).

“You've made a Save/Load game system in Blueprints, C++ and Blueprints, and pure C++”

First add two `includes` to the H file – (`#include [YOURSAVENAMEHERE].h`) and (`#include “Kismet/GameplayStatics.h”`)

```
#include“CoreMinimal.h”
#include“GameFramework/Actor.h”
#include“cMyAwesomeSave.h”
#include“Kismet/GameplayStatics.h”
#include“MyAwesomeActor.generated.h”
```

At the bottom of our H file – before the `}` – add a `SaveToSlot` void, `LoadFromSlot` void and an `example` variable created in your SaveGame (for me, that will be an `FVector`).

```
voidSaveToSlot();

voidLoadFromSlot();

FVectorExampleVector;
```

Head on over to the CPP file and fill out the two functions.

```
voidMyAwesomeActor::SaveToSlot()
{
    UMyAwesomeSave* SaveGameRef =
    Cast<UMyAwesomeSave>(UGameplayStatics::
    CreateSaveGameObject(UMyAwesomeSave
```

```
::StaticClass())); // Create our Save Game
and store it as “SaveGameRef”
    SaveGameRef->sgPlayerLocation=Exampl
eVector; // Set the Save Game version of our
Variable here.

UGameplayStatics::SaveGameToSlot(SaveGameRef,
SaveGameRef->SaveSlotName, SaveGameRef-
>UserIndex); // You can use a custom Slot
Name or User Index instead – Just like the
Blueprint version!
}

voidMyAwesomeActor::LoadFromSlot()
{
    UMyAwesomeSave* LoadGameRef =
    Cast<UMyAwesomeSave>(UGameplayStatics
::CreateSaveGameObject(UMyAwesomeSave
::StaticClass())); // Load our Save Game and
store it as “LoadGameRef”
    LoadGameRef = Cast<UMyAwesomeSave>(
UGameplayStatics::LoadGameFromSlot(LoadGameR
ef->SaveSlotName, LoadGameRef->UserIndex)); //
Load the Slot / User just like we did in the
Blueprint.
    ExampleVector=LoadGameRef-
>sgPlayerLocation; // Update our Variable!
}
```

Now you can call these any time you like within this Actor (such as in the `Begin Play` function) by using `LoadFromSlot()` and `SaveFromSlot()`.

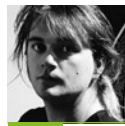
We've finished! You've created a full Save/Load game system in Blueprints, C++ and Blueprints, and pure C++. From here, you could extend the system's functionality by adding your own functions, such as only allowing a save once a player has lost five lives. The possibilities are almost endless! 🗨

TESTING

Want to test to ensure your code is working correctly? Add `Print String` nodes at the end of the two code execution chains to see if they actually fire. Want to go even deeper? You can right-click nodes to add or remove breakpoints to pause the game at that exact location. Perfect when hunting for bugs or fixing issues.

Standing out with RPG Maker plugins

How to improve your role-playing game with free plugins from the RPG Maker MV community



AUTHOR
ANNA BLACKWELL

Anna Blackwell is a freelance game journalist and hobby game designer in both the digital and tabletop worlds. Check her out at blackwellwrites.com



^ **Final Fantasy IX was the last to use the ATB system before the series switched to Conditional Turn-Based combat.**

From its debut as Mamirin on the PC-88 computer in 1988 to its most recent update, RPG Maker MV has provided JRPG fans with the toolset to realise their stories and designs. But with the market getting more and more crowded, how do you make your game stand out? With plugins. And art, music, and other things that are equally important – but today, we’re looking purely at plugins.

If you’re developing an RPG Maker game, odds are you’re a hobbyist or indie designer who just wants to make something without getting bogged down in engine design. I can appreciate that, and I can also appreciate that we’re all a bit skint these days, so I’ve curated some useful free plugins that will help your game stand out from the crowd.

“I’ve curated some useful plugins to help your game stand out”

But before we dive into the rest of the plugins in the list, you’ll want to have a look at the incredibly useful ‘Core’ plugins by programmers Yanfly and Victor Sant, as most of the plugins require or expand on code from these cores.

MEAT AND BONES

The Victor Engine collection of plugins by Victor Sant requires the Basic Module (wfmag.cc/rpgbasic) for any of Victor’s plugins to work, so make sure to have that installed if you download any of his plugins.

Yanfly’s Core Engine (wfmag.cc/rpgcore), while not required, is an immensely handy plugin which lets you take control over the resolution, item/gold/exp limits, as well as

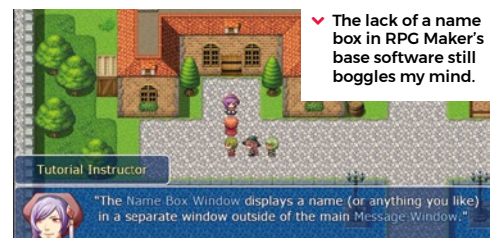
providing quick access to the console during testing. An important first step to standing apart from the crowd is to change up how the game looks and give it a professional full-screen.

Yanfly’s Battle Engine Core (wfmag.cc/rpgbattle) changes up RPG Maker MV’s slow, methodical combat with simultaneous damage resolution, better mouse control, and better menu placement, as well as more attractive animations. I can’t recommend this enough.

For the more story-focused designer, Yanfly’s Message Core (wfmag.cc/rpgmessage) adds name boxes, coloured and scalable text, and a tonne of functionality to dialogue that’s surprisingly absent from the base software.

This helps to give dialogue that visual novel feel with timing code giving speech a more varied rhythm.

The Yanfly and Victor Engine plugins should work together as long as they aren’t affecting the same code, and are available for your version of RPG Maker MV. Make sure to check all this before installing plugins. But now, on to the good stuff!

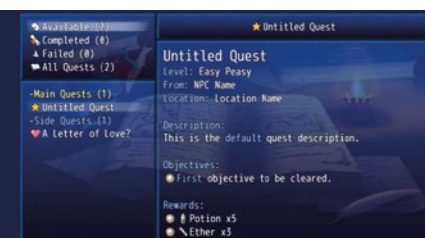


^ **The lack of a name box in RPG Maker’s base software still boggles my mind.**

Tutorial Instructor
“The Name Box Window displays a name (or anything you like) in a separate window outside of the main Message Window.”



^ **The Quest Journal’s simple looking, but necessary for making an RPG feel like an RPG.**



COMBAT EVOLVED

An important part of any JRPG is the combat, and while RPG Maker MV provides developers with the ability to easily change between front-on and side-facing battles, the out-of-the-box selection just doesn't cut it anymore: the turn-based battle system is functional but, frankly, a bit boring. Thankfully, the RPG Maker community has been working on new battle systems and helpful additions to the base system.

As the battle system of choice for the *Final Fantasy* series from *Final Fantasy IV* through *IX*, it's no surprise that both Yanfly and Victor have made their own versions of the Active Turn Battle System (wfmag.cc/rpgactive). Notable for its love-hate reception among users, the system gives each character a bar that fills up over time, prompting the player to input their turn as soon as the bar is filled. This makes combat more pressing, requires quick thinking, and lends combat a greater tension.

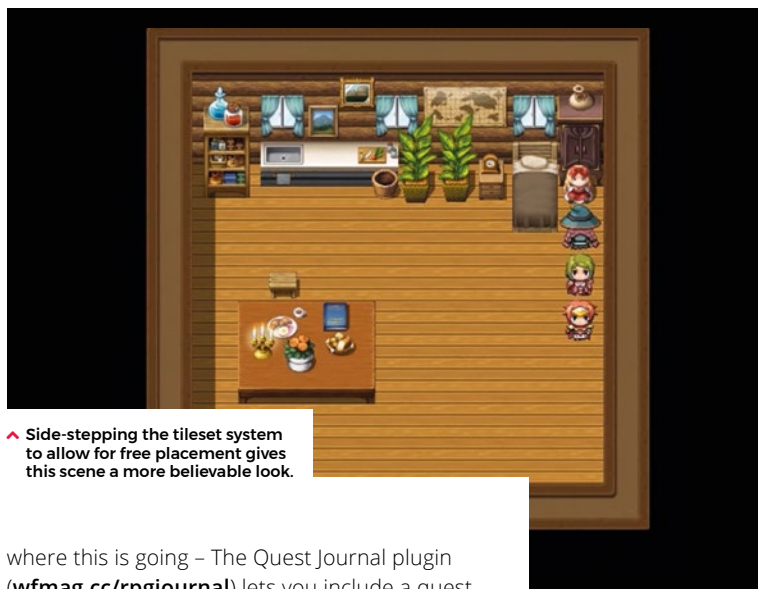
For a lighter, less pushy Active Time Battle System, there's also the choice of the Conditional Turn-Based System (wfmag.cc/rpgcondition) found in *Xenosaga I to III* and *Grandia III*, which pauses the bars when you're making a turn.

MOVEMENT AND JOURNALS

Combat isn't the only thing that makes or breaks a JRPG, and while RPG Maker MV provides a substantial map editor, it has one major flaw: everything is tile-based. Tileset A is used for creating the 'land' while B through E are used to add doodads which are stuck to this tile system. Thankfully, Yanfly has put together the Grid-Free Doodads (wfmag.cc/rpgdoodads) plugin which allows for free placement of static and animated objects to give scenes more life.

Speaking of more life, thanks to the power of plugins, it's also possible to break characters out of their antiquated cardinal movement system with Diagonal Movement (wfmag.cc/rpgdiagonal). It's not much, I know, but it'll at least let your players feel a little freer in their movements. Be aware, though, that including diagonal movement means you'll have to create diagonal spritesheets for every actor you intend to let move in those directions.

Finally, one of the most important parts of any RPG: the quest journal. It's often left out of hobby RPG Maker titles as it's a hassle to incorporate and – well, you can already tell



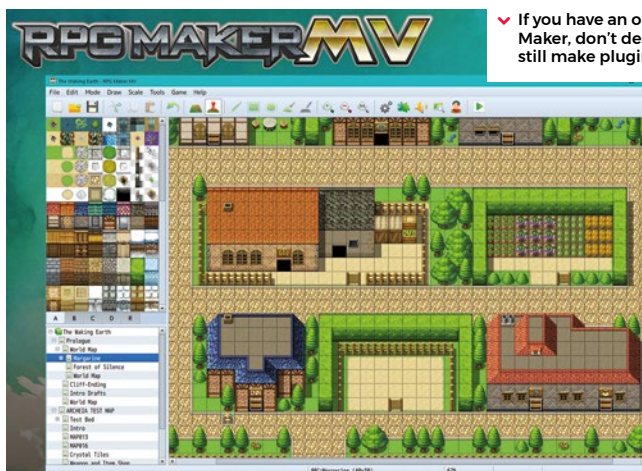
▲ Side-stepping the tileset system to allow for free placement gives this scene a more believable look.

where this is going – The Quest Journal plugin (wfmag.cc/rpgjournal) lets you include a quest screen in the menu. With it, you can “adjust the quest's title, display a difficulty level, remind the player who and where that quest is from, add various dynamic descriptions explaining the quest,” and lots more besides. The base plugin gives 100 quest slots, but an extension plugin can be installed to increase this number and really give quests a professional look. You can also use the Map Quest Window (wfmag.cc/rpgwindow) – a useful little addition that allows you to track your current quest and updates it as you complete objectives.

There are hundreds of other plugins that let you hack RPG Maker MV (or its predecessors) into whatever shape you could want. And if you're looking to get into game development as a hobby, but are worried that you're lacking a particular skill, don't worry – the RPG Maker community has created vast collections of plugins, art, and music for you to use. 🎵

Plugins, DLC, art, and more

The Yanfly team and Victor Sant are just a couple of great members in the RPG Maker community. Others, like HimeWorks, Fallen Angel Olivia, Moghunter, SumRndmDde, and Galv are out there making excellent plugins. If you need art, itch.io and specifically Tyler Warren's massive selection of Battlers should certainly help. And of course, there are over a hundred DLC packs available on the RPG Maker MV Steam page, just in case you thought you were lacking options.



♥ If you have an older version of RPG Maker, don't despair: many developers still make plugins for VX Ace.





◀ Options first appeared in 1985's *Gradius*, but became a mainstay of numerous sequels and spin-offs, including the *Salamander* and *Parodius* series of games.



Gradius' ship-following Options



AUTHOR
RIK CROSS

Learn how to create game objects that follow the path of the main player sprite

First released by Konami in 1985, *Gradius* pushed the boundaries of the shoot-'em-up genre with its varied level design, dramatic boss fights, and innovative power-up system. One of the most memorable of its power-ups was the Option – a small, drone-like blob that followed the player's ship and effectively doubled its firepower. By collecting more power-ups, it was possible to gather a cluster of death-dealing Options, which obediently moved wherever the player moved.

There are a few different ways of recreating *Gradius*' sprite-following, but in this article, I'll show you a simple implementation that uses the player's 'position history' to place other following items on the screen. As always, I'll be using Python and Pygame to recreate this effect, and I'll be making use of a spaceship image created by 'pitrizzo' from opengameart.org.

The first thing to do is to create a spaceship and a list of 'power-up' objects. Storing the power-ups in a list allows us to perform a simple calculation on a power-up to determine its position, as you'll see later. As we'll be iterating through the power-ups stored in a list, there's no need to create

“A small, drone-like blob that followed the player”

a separate variable for each. Instead, we can use list comprehension to create the power-ups:

```
powerups = [Actor('powerup') for p in range(3)]
```

The player's position history will be a list of previous positions, stored as a list of (x,y) tuples. Each time the player's position changes, the new position is added to the front of the list (as the new first element).

We only need to know the spaceship's recent position history, so the list is also truncated to only contain the 100 most recent positions. Although not necessary, the following code can be added to allow you to see a selection (in this case every fifth) of these previous positions:

```
for p in previouspositions[::5]:  
    screen.draw.filled_circle(p, 2,  
    (255,0,0))
```

Each frame of the game, this position list is used to place each of the power-ups. In our *Gradius*-like example, we need each of these objects to follow the player's spaceship in a line, as if moving together in a single-file queue. To achieve this effect, a power-up's position is determined by its position in the power-ups list, with the first power-up in the list taking up a position nearest to the player. In Python, using `enumerate` when iterating through a list allows us to get the power-up's position



Sprite-following Options in Python

Here's a code snippet that creates *Gradius*-style Options in Python. To get it running on your system, you'll first need to install Pygame Zero – you can find full instructions at wfmag.cc/pgzero

```
# set screen width and height
WIDTH = 800
HEIGHT = 800

# create spaceship and a list of 3 powerups
spaceship = Actor('spaceship', pos=(400,400))
spaceship.speed = 4
powerups = [Actor('powerup') for p in range(3)]

# create a list of previous positions
# initially containing values to the left of the spaceship
previouspositions = [(spaceship.x - i*spaceship.speed,spaceship.y) for i in range(100)]

def update():

    global previouspositions

    # store spaceship previous position
    previousposition = (spaceship.x,spaceship.y)

    # use arrow keys to move the spaceship
    if keyboard.up:
        spaceship.y -= spaceship.speed
    if keyboard.down:
```

```
spaceship.y += spaceship.speed
if keyboard.left:
    spaceship.x -= spaceship.speed
if keyboard.right:
    spaceship.x += spaceship.speed

# add new position to list if the spaceship has moved
# and ensure the list contains at most 100 positions
if previousposition != spaceship.pos:
    previouspositions = [(spaceship.x,spaceship.y)] +
previouspositions[:99]

# set the new position of each powerup
for i,p in enumerate(powerups):
    newposition = previouspositions[(i+1)*20]
    p.pos = (newposition[0],newposition[1])
```

```
def draw():
    screen.clear()
    spaceship.draw()
    for p in powerups:
        p.draw()
```

✓ Plotting the spaceship's position history.



✓ Power-ups following a player sprite, using the player's position history.



in the list, which can then be used to determine which position in the player's position history to use.

```
newposition = previouspositions[(i+1)*20]
```

So, the first power-up in the list (element 0 in the list) is placed at the coordinates of the 20th ((0+1)*20) position in the spaceship's history, the second power-up at the 40th position, and so on. Using this simple calculation, elements are equally-spaced along the spaceship's previous path. The only thing to be careful of here is that you have enough items in the position history for the number of items you want to follow the player!

This leaves one more question to answer; where do we place these power-ups initially, when the spaceship has no position history? There are a few different ways of solving this problem, but the simplest is just to generate a fictitious position history at the beginning of the game. As I want power-ups to be lined up behind the spaceship

initially, I again used list comprehension to generate a list of 100 positions with ever-decreasing x-coordinates.

```
previouspositions = [(spaceship.x -
i*spaceship.speed,spaceship.y) for i in
range(100)]
```

With an initial spaceship position of (400,400) and a `spaceship.speed` of 4, this means the list will initially contain the following coordinates:

```
previouspositions = [(400,400),(396,400),
(392,400),(388,400),...]
```

Storing our player's previous position history has allowed us to create path-following power-ups with very little code. The idea of storing an object's history can have very powerful applications. For example, a paint program could store previous commands that have been executed, and include an 'undo' button that can work backwards through the commands. ☺

LIST COMPREHENSION

List comprehensions are a way of creating a list, using other iterables (a sequence of objects). This is really handy for creating large lists and/or lists where the elements follow a pattern. List comprehension has been used twice in this sprite-following example; for creating a list of three power-ups and for quickly creating a list of 100 previous spaceship positions.

```
>>> squares = [i*i for i in range(5)]
>>> squares
>>> [0,1,4,9,16]

>>> numbers = [2,3,4,5,6,7,8,9,'J','Q','K','A']
>>> suits =
['Hearts','Clubs','Spades','Diamonds']
>>> playing_cards = [(n,s) for n in
numbers for s in suits]
>>> playing_cards
>>> [(2, 'Hearts'), (2, 'Clubs'), (2,
'Spades'), (2, 'Diamonds'), (3, 'Hearts'),
...]
```

Directory

Upcoming events for game developers

Courtesy of Ukie, here's a selection of game dev events coming up around the UK this summer

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

✓ You'll find over 40 speakers from around the esports industry at this year's ESI London.



➤ **UK-China Workshop, Goldsmiths,**
London, 25 June

Exploring immersive art and tech, this free workshop also aims to forge creative partnerships between the UK and Shanghai. Will include short talks, networking sessions, and demos.

wfmag.cc/china-workshop

➤ **Nova Games Conference 2019,**
Gateshead, 27 June

The Nova Games Conference is an annual event in the North East of England, focusing on themes, topics, and talks which can educate and inspire programmers, designers, artists, and more.

wfmag.cc/nova-games

➤ **Guildford.Games Festival,**
Guildford, 27-29 June

A city key to the UK games industry is getting its first ever Guildford Games Festival. Here you'll find industry talks, workshops, plus a multiplayer event hosted by Supermassive Games.

wfmag.cc/guildford-fest

➤ **Game Dev Heroes Awards 2019,**
Brighton, 9 July

Returning for 2019, Game Dev Heroes recognises and celebrates the work of the people behind the scenes of the games industry.

wfmag.cc/dev-heroes

➤ **Develop:Brighton 2019,**
Hilton Brighton Metropole, 9-11 July

Hear from the industry's leading lights, get up to date with the latest tools, techniques, and industry trends, plus network with over 2000 game dev professionals.

wfmag.cc/develop-brighton

➤ **Ukie Hub Crawl: Getting Investment Ready,**
Dundee, 16 July

Come and join fellow games developers and publishers and identify new opportunities and strategies to support the growth of your business.

wfmag.cc/hub-crawl-dundee

➤ **Esports Insider:**
London 2019, 16-17 September

ESI London will bring you up to speed, feed your ideas, and help you find the right partners and funding. In short, it'll provide all the esports knowledge you'll need in one place.

wfmag.cc/esi-london



The UK's BRAND NEW FILM MAGAZINE

TRY
**THREE ISSUES
FOR £9.99**
with the code 'RASPBERRY'

THE LATEST RELEASES REVIEWED
FILM STORIES
A NEW FILM MAGAZINE, FROM THE CREATOR OF DEN OF GEEK
EAST IS EAST, WILD WEST & REPRESENTATION
CONFESSIONS OF A CINEMA MANAGER
THE INCREDIBLE, OVERLOOKED BOX OFFICE RUN OF BETTE MIDLER
AUTISM IN THE MOVIES
FEATURING ROMESH RANGANATHAN, DEBBIE MOON, HOPE DICKSON LEACH, ANNA SMITH

webscribe.co.uk/magazine/filmstories



The Art and Code of Joffa Smith

The late Jonathan 'Joffa' Smith's work on the ZX Spectrum inspired a generation of designers. Kim pays tribute to his endless creativity



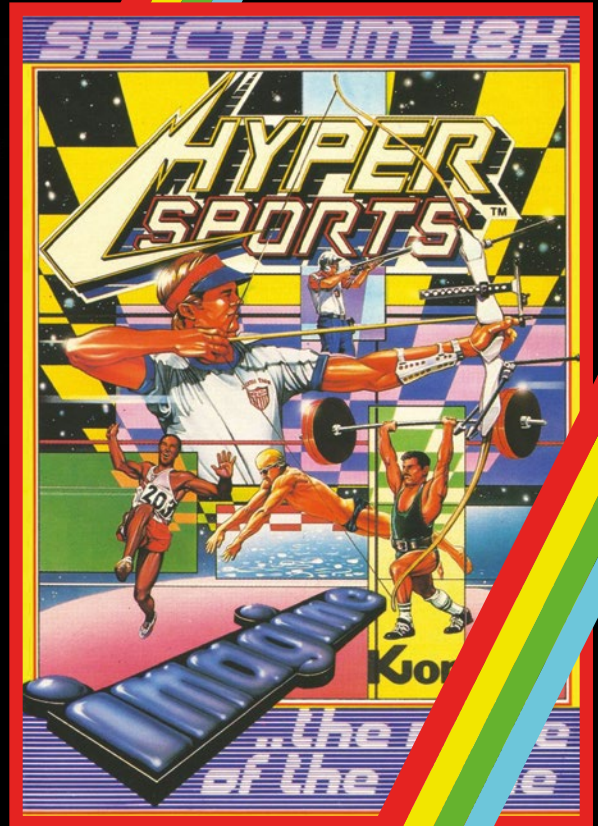
WRITTEN BY
KIM JUSTICE

If he hadn't ended up falling into programming, it's plausible that Jonathan 'Joffa' Smith, the brain behind such ZX Spectrum classics as *Cobra*, *Green Beret*, and *Hyper Sports*, would have gone into filmmaking – specifically, animation. He was one of the first school kids to submit a short animated flick, *The Thorn*, for his art O-level, and may have been accepted into the National Television & Film School if it wasn't for a job offer from Ocean Software.

The 1984 documentary *Commercial Breaks* shows Joffa in the process of being hired, as he demonstrates a game to studio bosses David Ward and Paul Finnegan while studiously trying to avoid the TV cameras. And yet this shy teenager – a self-proclaimed 'sociophobe' – was among the rarest of breeds. It's often thought that coding and graphics are two entirely separate disciplines, with little crossover between them; in the Spectrum era, Joffa was among the best in both areas.

"He was a great all-rounder, programmer, artist," recalls programmer Jim Bagley, who worked with Joffa at Liverpool-based studio Special FX. "He could make the Speccy do amazing things. The Speccy was his specialist subject." ➔





Interface

The Art and Code of Joffa Smith



That first game Joffa presented to Ocean would end up on the shelves as *Pud Pud In Weird World*, and it showed how creative he could be with tiles and sprites. While much of Joffa's later work involved taking arcade classics and making remarkably faithful conversions of them on the 48K Spectrum, *Pud Pud*'s entirely original art and premise stood out. You control a bright yellow pudding-esque figure in a maze filled with other assorted puddings that you must eat, an angered woman roaming around who'll kill you outright if you touch her, and plenty of strange artefacts – a huge skull here, a broken Speccy there, plus an unforgettable 'Joffa Says R.I.P.' gravestone.

There are some surprisingly large, multi-tile graphics in *Pud Pud*. Indeed, the hefty number of objects in Joffa's games never compromised

“The technical prowess on display in Joffa’s games was, for the time, staggering”

their speed and immediacy – he always aimed to make the best arcade-style games he could. Later original works, such as *Firefly* for Special FX – his personal favourite – would raise the stakes even higher, especially once the 128K Spectrum came out and gave him even more memory to play with.

For Joffa, that memory provided an even bigger canvas, which was how he preferred to work. He summed up his preferences in a 2009 interview for *The Retro Brothers* in typically enthusiastic style.

“You can say to someone that they have a completely blank canvas, and they say yeah [fine],” Smith said. “And then they find out what a completely clear sheet of nothing is, and they run. No maths subroutines, no auto screen manipulation, no help... ‘Calling’ a development kit routine to find the square root of a number isn’t programming. Take control. Work out how to do it yourself, and do it fast! That’s programming.”

SMOOTHLY DOES IT

The technical prowess on display in Joffa's games was, for the time, staggering. With *Cobra*, he managed to create a Spectrum game with smooth parallax scrolling – a remarkable feat, given that the system that had no dedicated graphics hardware. Then there was the interrupt-driven ‘plip-plop’ music routine he created for Doug Burns' *Ping Pong*, which made the Spectrum sound as though it had a second music channel working alongside its standard single-channel beeper, as well as a drumbeat (this routine was originally going to be in Joffa's own *Green Beret*, but he ran out of memory).

One game that Joffa often liked to mention was *Mikie*, a great conversion of a fairly obscure Konami coin-op which featured what he described as the most complicated graphics routine he ever wrote.

The game itself looks simple, but there's an awful lot going on under the hood to create a classroom full of kids and flashing hearts, not to mention a kiss-chasing hero and an angered teacher, all moving smoothly around the screen. When Ocean company director Jon Woods saw

ON THE WEB

Joffa could be found at many places on the internet in his last years, from his official webpage, where he shared various thoughts on his games (now sadly closed), to the *World of Spectrum* forums, where he frequently posted under the name of Frobush. He also had a small YouTube channel full of various vids, from his old art O-level animation to him showing his distaste for *Terra Cresta* by chucking a watering can at what appears to be a Microdrive cartridge containing a prototype of the game.



JOFFA'S OTHER PROJECTS

While gaming is where he made his money, Joffa's talents often branched out into other fields. He continued working on animation in his spare time, and also spent a fair while working as a DJ in Blackpool in the nineties (a period he would often refer to as his "wilderness years"). Most curiously of all though, he produced the animated computer displays for series two and three of the cult TV comedy series, *Red Dwarf*, a project that he apparently crunched through in all of two weeks.



▲ Joffa didn't like being photographed, but he was dragged into this picture, taken for a magazine back in the Special FX days.

that Joffa already had two screens running on the game after just a day or two of development, he was so astonished that he jokingly told Joffa to – and we're paraphrasing here – go forth and multiply.

In those days of relatively short dev cycles and inflexible deadlines – generally led by a company's desire to get its latest game advertised in the press – Joffa often wished he had at least a week or two more to work on some of his games. With the exception of *Terra Cresta*, however (a game he considered unfinished and riddled with bugs), Joffa was, for the most part, generally pleased with what he managed to achieve on the Spectrum.

I'M THE CURE

Of all of these games, *Cobra* is perhaps the most worthy of a closer look – not just because its parallax scrolling is the most obvious example of Joffa's coding brilliance on the system, but also because of the game's general creativity and weirdness. He and musician Martin Galway had the pick of two films to create a game on – either *Top Gun* or *Cobra*. They didn't care much for either of them, but Joffa thought that *Cobra*, an action flick starring Sylvester Stallone, was the best vehicle for the game he truly wanted to create: a *Mario*-esque side-scroller for the Spectrum. As a game, *Cobra* is nothing like the movie it comes from, featuring all sorts of

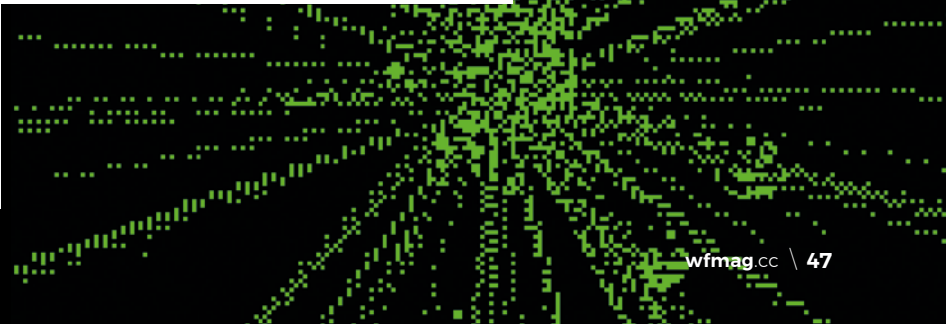
strange enemies, a damsel that needs to be rescued in each stage, and a rubber duck that represents how much ammo you have left in your weapon, not to mention music cues and boxing gloves that refer back to another, much more famous set of Stallone movies. The game takes a generic licensed product, adds a big slice of irreverent, somewhat British humour alongside some fast and furious arcade play, and comes up with something truly special.

MAKING IT WORK

For the more technical-minded, though, it was *Cobra's* smooth scrolling that warranted attention. Horizontal scrolling, in particular, was tricky on the Spectrum, because it lacked any dedicated graphics hardware; there was no equivalent to the Commodore 64's VIC-II chip, meaning scrolling was often choppy, and most games stuck to flick screens instead of dealing with slow, jerky gameplay. Joffa managed to figure it out, however, and *Cobra* benefitted hugely from his expertise. In order to create the game's signature scrolling, he coded a bit →



▲ A still from *The Thorn*, in which a dog is getting the Bart Simpson treatment. Animation's loss would be gaming's gain.





^ In this still from a YouTube vid, a probably quite rare *Terra Cresta* Microdrive cart is just about to be struck by a watering can.



to run at half-speed to a blank block in order to create a 'beam' – a stack of code that would continuously draw new rows of graphic tiles without any of the flickering you would often get in games that attempted horizontal scrolling.

The construction of the tiles themselves would also produce a parallax effect that, again, was quite rare on Spectrum games of the time. Joffa himself was, as ever, somewhat modest about the effect, preferring to single out games like *Mikie* for their graphics routines and saying that, as ever, if he'd had more time he could have done more – such as having a bitmap image in the background.

"Another week, and it would have been *awesome*," he told *The Retro Brothers* in 2009. "But you have to let go, or rather it gets taken from you. That's it! Gone. No game testing. Just gone. But this is other people's money and if they say stop, you stop. And there's a back catalogue of stuff to be done before Christmas."

START TO STAB

Joffa was never too precious about his work – he was always happy to show others how he achieved a certain effect, or coded a routine.

Creating games was its own reward – something that allowed his personality to shine through, in ways that he found trickier in normal social interaction.

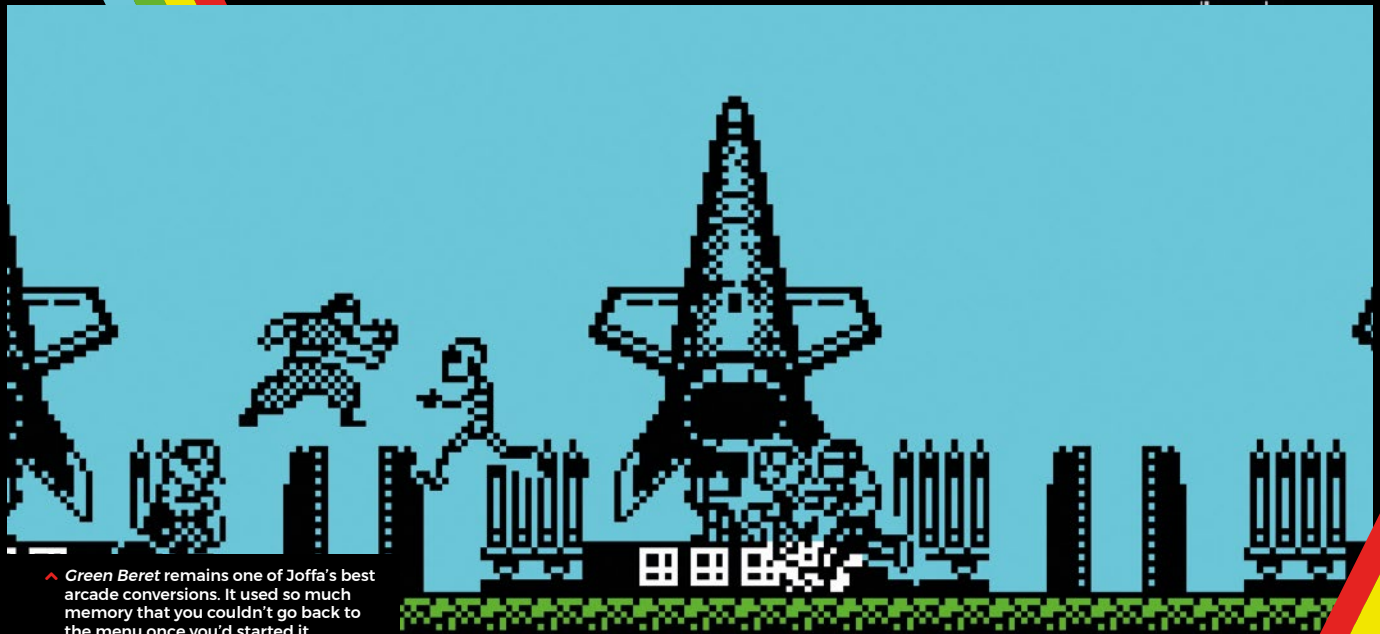
Even in conversions of arcade games such as *Green Beret* or *Hyper Sports* lie little signature touches here and there; iconic commands, such as 'STAB TO START'; his little intros, or his backwards name that, being by his estimation the most boring name anyone could possibly have, he was always trying to make more interesting. Many of those who worked with Joffa over the years recall that he had a spark of fun about him, and how good he was to work with.

"He was awesome," remembers Jim Bagley. "He showed me his push-scroll routine that he'd used in *Cobra*, which I then used in *GUTZ* for the main game screen, and I made some modifications so it could do a big block for the corridors between levels. He was really helpful, had plenty of time for everyone, and was always good for a laugh."

GAME UNDER

In later years, this humour – along with the technical quality and playability of his games – would continue to be celebrated on the internet, which was another place where Joffa was quite comfortable speaking about what he'd done, or what he was going to do. He was enigmatic and






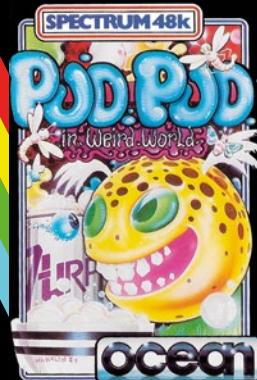
^ *Green Beret* remains one of Joffa's best arcade conversions. It used so much memory that you couldn't go back to the menu once you'd started it.

kooky, yet friendly and approachable on various forums. They, and many others, were shocked to hear of his passing in 2010 after a short illness, aged just 43 – far too young, and still with so much more to give.

A solo homebrew project called *Saucer*, which he'd worked on for years, would, alas, go unfinished. But through the encouragement and insight he gave in later years, wrapped up in his style and eccentricities though it may have been, he was able to inspire many generations of Spectrum coders, from grizzled veterans to nascent homebrewers.

It's a shame that he missed out on the renewed interest in retro games we've seen in recent years, especially those on the Spectrum – it would have been fascinating to see what he'd have come up with on a machine like the Spectrum Next. But for those who once worked with him, such as Jim Bagley, part of what they're doing now involves taking a little piece of Joffa along for the ride, and ensuring that his legacy continues to live on.

"I like to put humour in the games I make," Bagley says, "and that was one of the things he loved doing... that, and just being different, thinking outside the box. He's missed an awful lot – he was hugely respected and liked by everyone who knew him. He was such a creative genius, and it remains a sad loss to the gaming industry, and to the players." 



< The box art for *Pud Pud In Weird World*, as with many Ocean games, was a creation of the late and much-missed Bob Wakelin.

ON MODERN GAMES



Joffa was always much happier working on the systems he was accustomed to, and wasn't particularly into modern games. He had this to say about them: "Yeah, I think games *have* evolved. It's a shame that everything, development-wise, has gone all big budget and Hollywood. Swings and roundabouts, I guess. There's room for simple stuff, too. And I get the feeling that the tide is turning... It's wonderful that the retro scene is really taking off. And I'm proud to be a part of it."

Developer Profile

id Software

A look back at the renegades of Doom

It's a bit of a shocker that July 2019 marks the first time, ever, id Software's Quakecon has left the protective confines of the United States and made its way to Europe – more specifically, London. This is a massive studio with serious international appeal; one of the core foundational devs of the first-person shooter genesis; beloved and respected in equal measure. But it's taken 23 years for the team to pack its bags and take the pageantry and... shooting... of Quakecon outside US borders. What better time to look back at the studio named for one-third of Freud's model for the human psyche? What better time, indeed.

The story of id Software begins, as most things do, somewhere else – Softdisk. A software company based in Shreveport, Louisiana, it paid the bills in a number of ways, including through the sale of a monthly game disk subscription service known as Gamer's Edge. This service was the brainchild of one John Romero, who was working at Softdisk alongside John Carmack, Adrian Carmack, and Tom Hall. Those who know where this is going: don't ruin the surprise.

So this team of fine gaming minds was put to work on quickly developed games to be shunted out on a monthly basis. This wasn't enough for the fearsome foursome, who decided to work on their own project outside of work hours, but while using work computers. Eventually, one *Commander Keen* was produced in 1990; a fine little smooth-scrolling platformer the likes of which had only



really been seen on console to this point. Ideas From the Deep, as the team called itself, had its first release... and its first brush with controversy.

Turns out using work computers for out-of-work projects is sometimes frowned upon, and Softdisk wasn't happy with what the team had been moonlighting on. While John Carmack and the rest of the Gamer's Edge team left Softdisk to officially form id Software, Softdisk brass pushed its now former employees into providing regular titles for the Gamer's Edge disks for its first year, at a rate of \$5000 per game. It wasn't glory-riddled work, but it ended up helping id with a lot of prototyping and, basically, 'mucking about' and toying with development.

EARLY ID-EALS

With the need to produce pretty much a game a month, id's output on its own projects was understandably limited for that first year. Eventually, they were able to put together a few follow-up

▼ *Rage 2* shows id still knows how to make shooting things fun.





Commander Keen episodes with original publisher Apogee (and id fanboy, Scott Miller) at hand, and it looked like the team was going to settle into a rhythm of making cartoony, silly, fun platformers for the PC. Well, it would have done were it not for their work on the likes of *Hovortank* and *Catacomb 3-D* as part of the Softdisk arrangement.

These early first-person shooters laid the groundwork for what would become *Wolfenstein 3D*, one of the most influential, bar-raising titles ever released. Smooth and quick, with intelligent enemies and a thick atmosphere of tension and, frankly, silliness, *Wolf 3D* was unlike anything else gaming had seen before. Ideas From the Deep might have formed in 1990, but it was 1992 when id Software truly became 'a thing'.

BOLD ID-EAS

From that point, there was no turning back, beyond the odd foray into traditional first-person dungeon-crawling RPGs in the late noughties. id very much became the FPS studio, cranking out the absolute legend that is *Doom* – still one of the best games ever made, and we will resolutely stand by that until either hell freezes over or invades our solar system via an experiment with portal technology on Mars.

And the bar-raisers just kept coming – not just good or great games, but epoch-making, industry-shaking titles. *Quake* in

“Quake III pre-empted fast-paced, arena-based FPS esports by a good decade-plus”

1996 with its fully 3D, texture mapped, and buttery-smooth visuals, along with John Carmack's work on standardising what we now take for granted in online play, through his tinkering with TCP/IP networking models. *Quake III* pre-empted fast-paced, arena-based FPS esports by a decade or so. *Doom 3* wasn't the greatest game, but lifted the medium with its id Tech 4 engine, while *Quake Live* was

another example of the studio being ahead of the curve, (originally) offering free-to-play before it was a dirty term.

In recent years, id has been shunted to the side by other younger, more vital studios. Its acquisition by Bethesda saw the original development renegades – a team that made a hacked version of *Super Mario Bros. 3* on PC and sent it to Nintendo to try and curry favour – somewhat neutered. The majority of the big names left, most notably genius-level tech wrangler John Carmack in 2013 (genius-level designer John Romero had left back in 1996), and id fell into relative obscurity, the butt of a few jokes about its then-upcoming *Doom* revival being nothing more than empty promises that could never live up to the hype.



^ *Doom II*'s hidden boss, of sorts: the head of John Romero, impaled on a spike. Oh, id.



^ *Keen* displayed a... well, keen sense of humour.

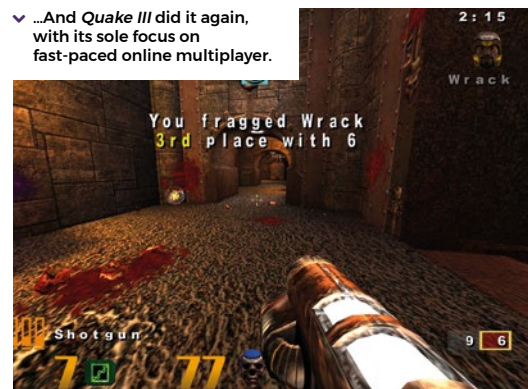
Doom 2016 destroyed the hype and set in motion the modern id era. *Rage 2* was a co-production, but had id's fingerprints all over it, while *Quake Champions* aims to revive the love of online *Quake* we once felt through to our bones – and the upcoming *Doom Eternal* will, hopefully, see a continuation of the fantastic work the studio did on the series' revival.

This isn't the id Software we once knew; backed into a corner and making things just to see if it could. This is a corporate entity – a subsidiary of Bethesda. But it's still a vital team, making vital games, and offering the kinds of experiences that, frankly, nobody else even tries to make. ☹

^ *Quake* revolutionised the FPS genre...



^ ...And *Quake III* did it again, with its sole focus on fast-paced online multiplayer.

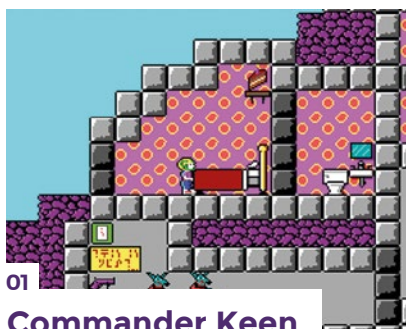




Beyond Wolfenstein

10 games that led to id's doom

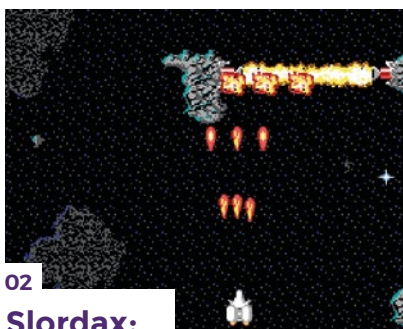
Before id changed the world, there were efforts like these



01 Commander Keen in Invasion of the Vorticons

PC – 1991

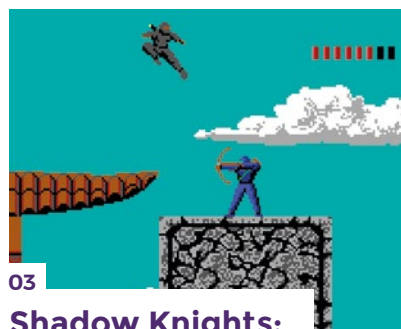
Commander Keen's adventures burst onto the DOS gaming scene from Ideas From the Deep, later id. John Carmack's adaptive tile refresh system allowed PCs of the day to display smooth scrolling for the platformer, something not really seen before outside of consoles.



02 Slordax: The Unknown Enemy

PC – 1991

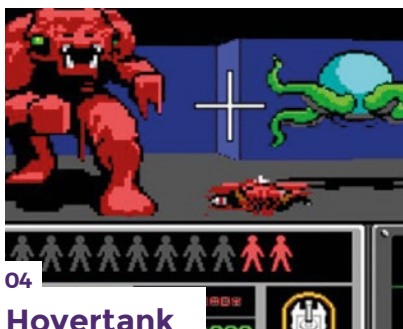
Another exercise in smooth DOS scrolling, *Slordax* took things in a more vertical direction. While technically a release from Softdisk, with the team leaving soon after the game's release to go it alone as id, it's fair to lump *Slordax* in with the pile marked 'early output' for the studio. Still, the shoot-'em-up had its charms.



03 Shadow Knights: The Shogun of Death

PC – 1991

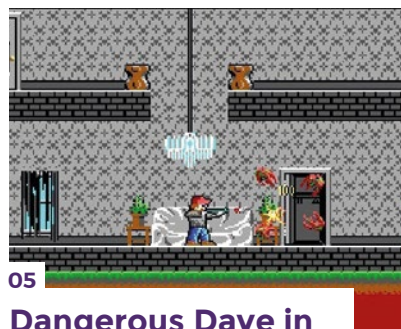
Developed in conjunction with *Slordax* (with Keen being made outside office hours), *Shadow Knights* brought *Ninja Gaiden* to DOS... in a way. Another smooth-scrolling, console-style experience on PC, it nonetheless fell into relative obscurity post-launch and is rarely mentioned these days.



04 Hovortank

PC – 1991

Whether you add a suffix to the title (*3D* or *One*, depending on your version), it matters not – *Hovortank* is still the first unsteady step for a dev team which would go on to take over the world. This engine, made by John Carmack in six weeks, would go through plenty of iterations and tweaks, but eventually ended up being the basis for *Wolfenstein 3D*.



05 Dangerous Dave in the Haunted Mansion

PC / Mobile – 1991, 2008

Sequel to 1988's *Dangerous Dave*, coded by Romero, *Haunted Mansion* pivoted from its *Mario*-clone origins and into more familiar id territory. That being: blasting zombies with a shotgun. An improved version of the *Shadow Knights* engine made for, once again, some lovely smooth DOS action.

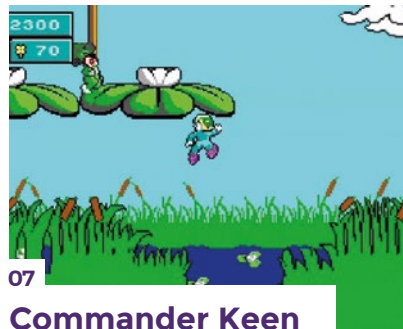




06 **Rescue Rover**

PC – 1991

A step out of id's comfort zone – and yet another title made to fulfil the studio's obligations to Softdisk – *Rescue Rover* was a puzzle game. Involving block-pushing and robot-avoiding – all in pursuit of saving a stranded pooch – this was a decent enough game to while away a bit of time. A sequel followed soon after, though the series stopped dead there.



07 **Commander Keen in Keen Dreams**

PC / Mobile / Switch – 1991-2013

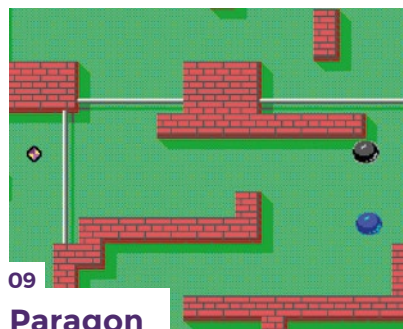
Keen's fourth episode was a prototype for what id wanted to do with the series outside its pesky Softdisk obligations. Released to little fanfare, *Keen Dreams'* surreal elements went largely ignored by the masses, and it wasn't until later re-releases that more eyes got to see what id had attempted to do with this 'lost' episode.



08 **Catacomb 3-D**

PC – 1991

A bit of polish and texture mapping brought the *Hovertank* engine into a new era (or month) with *Catacomb 3-D*. Bundling the early FPS experience forward plenty of steps, *Catacomb* saw that typical smooth id rendering, a delightful mix of violence and maze exploration and – possibly for the first time in the genre – let you see your hand on-screen. Bar-raising.



09 **Paragon**

PC – 1991

Originally created by Nick Peck for the C64, id ported *Paragon* (aka *Street Ball*) to DOS for the Softdisk deal. A fun puzzle game of sorts, *Paragon* has you launch a ball at set angles and bounce it through levels, collecting items along the way. It's worth including here mainly to point out how few people know, or care, about the game showing up in id's back catalogue.



10 **Commander Keen in Aliens Ate My Babysitter**

PC – 1991

id's final entry to the *Keen* series, released outside of the Softdisk deal, was notable not for any real technological advances, but more because it marked the end of the team's efforts in youth-focused game development. A sequel was being worked on, but something called *Wolfenstein 3D* happened to take over...

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



Review

Observation

Info

GENRE

Puzzle / Narrative

FORMAT

PS4 (tested) / Windows

DEVELOPER

No Code

PUBLISHER

Devolver Digital

PRICE

£19.99

RELEASE

Out now

Emoting in a tin can

O

bservation is both captivating in its translation of the tense silences and terrifying peaks of horror cinema, and a clear parable on how unethical it

is to force sentient AI to carry out repetitive, obtuse busywork. I came away from my time awed at how it balanced incisive, personal claustrophobia with expansive cosmic horror. I also came away feeling a bit sorry for my operating system, and only partly in the ways I think the game wanted me to.

Observation puts you in the role of SAM. That's Systems Administration and Management, by the way – the onboard computer of the space station, *Observation*. The game opens on the spinning interior of the station's European arm, to floating debris and the crackling, panicked voice of Dr. Emma Fisher. Clearly, something's gone badly wrong.

It's up to you to, uh, administer and manage some systems, then. After all, that's what they built you for.

To call what you'll be doing for the next five or so hours 'puzzles' is less accurate than to call them 'tasks'. *Observation* has a specific set of things it wants you to do in a specific order to advance the plot. A few of these require you to use your brain, but mostly you'll just be using your ears, your eyes, and your thumbs.

This sounds a bit dull, right? It's not great, admittedly, but the beauty of *Observation* is how UI, sound design, and story work together so that even performing mundane tasks can make you feel every bit the helpful, advanced AI companion you're supposed to be. It's less about the complexity of navigating these sleek science-fiction interfaces, more that there is so much detail and care put into both the visual design and interconnectivity of SAM's many systems that the interior of the station feels plausible and tangible. When that plausibility and tactility is used to terrify, *Observation* is at its strongest.

Then there's Dr. Emma Fisher (voiced by Kezia Burrows), who carries the entire game. *Observation's* understated humour makes it easier than you'd expect to care about the fate of an AI, but this is Emma's story above all, and I felt her every panicked inhalation and relieved sigh as if it were my own. There's a naturalistic imperfection to Emma's dialogue and performance. The result of her speaking as much to herself as she is to SAM, oscillating between allowing herself to forget and realising again that, technically, she's completely alone.

If *Observation* has the visual flair and tight pacing of a Hollywood flick, its focus on the personal makes it feel firmly like a British indie film. It's fantastical without feeling artificial or contrived, even in details as seemingly

REVIEWED BY
Nic Reuben



▲ Strong voice performances mean communicating with the crew is a highlight throughout.



▲ Individually designed interfaces for the station's many functions add variety to simple, repetitive tasks.

▲ Glitches and other visual anomalies are used to create an atmosphere of creeping dread.

small as remembering that characters in peril don't always speak in complete sentences. As the situation deteriorates, there's a sense that SAM becomes emotional, as well as technical, support. It's believable, human, and completely captivating.

Here's something crucial, though: the dangerous situations Emma finds herself in are all dictated by the plot. There are no fail states, effectively, or none I encountered. This is where the cinematic influence starts to overshadow *Observation's* interactivity, as it's sometimes difficult to buy into the importance of opening an airlock quickly when you know you can comfortably leave everything suspended in zero-G to go and make a consequence-free sandwich.

If the puzzles are neither all that interesting or all that tense, though, there is something ominously pleasing about dutifully carrying them out and being told 'good job, SAM' over and over in Dr. Fisher's gentle South Walian lilt. I got the sense that *Observation* was making a commentary on my self-assumed autonomy as a player here. The game never golf-clubs you in the face with it the same way *BioShock* did, but there are things that made me question the act of playing games. To think the ways in which we derive enjoyment from repetitive tasks, and how there's something unquestionably... robotic about the whole thing.

There are plenty of other questions *Observation* provokes, too. What's the nature of consciousness? How is our relationship to film affected if we view the camera lens as a

conscious participant, rather than an impartial observer? Why is it so much fun to annoy doomed astronauts by repeatedly pretending you didn't understand their requests? Honestly, this last one is great, especially as there's extra dialogue recorded for when Emma gets exasperated over you failing to remember very simple commands. Again, I'm beginning

to understand where my desktop is coming from.

You've probably noticed from the screens, but safe to say *Observation* is, pun intended, a real looker.

Photorealism isn't something I feel all games should strive for, but between the artful glitches and static and its environments, *Observation* is what all those adorably terrible FMV games from the nineties should have been. The production values are (kill me) stellar across the board. The notable exception being the – otherwise excellent – motion capture on characters' mouths, which tend to flap about randomly like the tentacles of an eldritch horror far more terrifying than anything in the actual game.

Ultimately, *Observation* is something you want to share with a friend who doesn't play games, just to demonstrate how incredibly absorbing the medium can be. But then, halfway through thinking about how exciting it all is, you'll start to recall all the repetitive tasks you have to complete along the way.

Observation's still worthwhile, and your friend's still going to love it, but the thought's enough to give you pause. 🤖

“It's believable, human, and completely captivating”

HIGHLIGHT

Kezia Burrows' understated, naturalistic, and utterly convincing performance as Dr. Emma Fisher gives *Observation* its emotional anchor. It's both a risk and great bit of unconventional storytelling to centre a game around a protagonist you don't actually play as, and *Observation's* story does some interesting things with this disconnect.

VERDICT

In drawing influence from cinema, *Observation* often forgets to make the most of its interactivity. This is the most gripping space admin job you're likely to hold this year, but it's still an admin job.

76%

HIGHLIGHT

The Space Between is deliriously uneasy in its final moments, thanks, in large part, to a skin-crawlingly stellar score. A tea kettle screams. Martial horns resound. There's a repeated noise that sounds like Cthulhu plopping his mighty tentacles on the bass end of a piano keyboard. Good stuff.

▼ *The Space Between* begins inside a blanket fort, and returns to this important location throughout the game.

SEPARATION AND CLOSENESS.

Review

▲ Separation and closeness are the game's core themes, explored in text (as in this conversation with Clara) and subtext.



The Space Between

The tension builds in a first-person narrative experience

Info

GENRE
Narrative... thing
FORMAT
PC (tested)
DEVELOPER
Christoph Frey
PUBLISHER
Christoph Frey
PRICE
£4.60
RELEASE
Out now

REVIEWED BY
Andrew King

VERDICT

Spookier and stranger than a game about something as clichéd as 'the walls we build' has any right to be.

74%

The *Space Between* is a melancholy, surreal, first-person narrative experience that looks like a PS1 game shot with a 1998 Sony Handycam then touched up with a coat of Microsoft Paint. If you've played indie horror games like *Paratopic* or *September 1999*, this will be covering familiar territory. Honestly, though, it's the most unsettling of the bunch.

The game's itch.io page bills it as being about "the walls we build for us," but the more high-concept summary is that *The Space Between* follows architect Martin Melanson, who stipulates that he be allowed to live in each structure he builds. That quirk dates back to childhood; in the opening moments, we play as Martin squirreled away inside a blanket fort. A friend, Daniel, asks him to come out, but Martin refuses. Daniel suggests that they put their hands together through the blanket. Martin obliges, and Daniel asks him whether he feels the blanket or his hand. Both, Martin replies.

This game is fascinated with that answer and its implications; that the metaphorical walls separating us from others don't actually fully wall us off, but instead, colour our perspectives. Frey extends this metaphor through a variety of objects: a coffin, a stage, an apartment, and most disturbingly, a blanket (I'll let you see it for yourself).

The game mostly focuses on conversations between Martin and Clara, a woman he meets smoking outside her apartment.

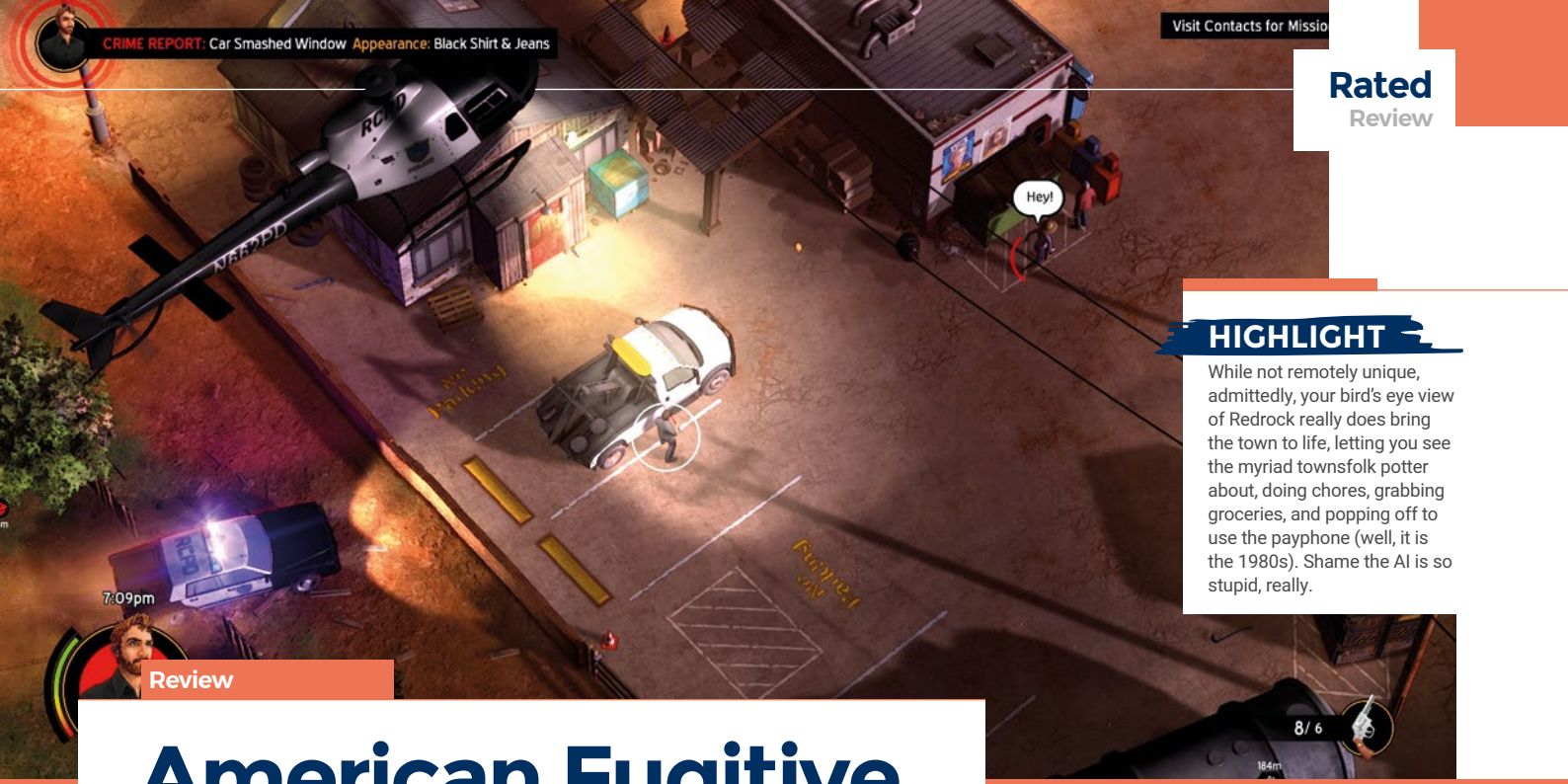
There are some weird design choices that make these conversations falter at times. The camera is wrested from your control, locking onto Clara's inexpressive face, and you can't control the pace of conversation. Additionally, dialogue isn't voiced, but each line is presented in the same white font, so it's sometimes difficult or impossible to know which character is saying what. There are only three characters, so I would love to see Frey patch in colours to differentiate each one's dialogue. Additionally, the game defaulted to playing windowed on my PC without an option menu to adjust the display.

But, minor issues don't detract from what *The Space Between* does so well. The thematic threads Frey weaves early in the narrative come back to reap dividends in the uneasy climax.

The game may shy away from gore, but the mundane images the camera does linger on – paired with a disturbing score – make the conclusion a late nineties-set nightmare. 🕸



▼ Players get a brief look at the theatre that Martin is building (and living in).



CRIME REPORT: Car Smashed Window Appearance: Black Shirt & Jeans

Visit Contacts for Missio

Rated
Review

HIGHLIGHT

While not remotely unique, admittedly, your bird's eye view of Redrock really does bring the town to life, letting you see the myriad townsfolk potter about, doing chores, grabbing groceries, and popping off to use the payphone (well, it is the 1980s). Shame the AI is so stupid, really.

7:09pm

Review

American Fugitive

And are you OK, are you OK, And?

F or a guy trying to clear his name, Will Riley is making some questionable choices. While I can't say I know how it feels to be accused of a crime I didn't commit – let alone the injustice of being sent down for murdering my own father – I'm inclined to think busting out of jail and picking up a life of petty (and not-so-petty) crime probably isn't the best way to go about it.

That said, I don't suppose many (any?) of us are picking up *American Fugitive* for its gripping story or social commentary. Fallen Tree Games' bombastic adventure is an unashamed love letter to the original *Grand Theft Auto* and, to its credit, it's stuffed with delicious carnage and chaos. Yes, it's a little derivative and no, it's not particularly cerebral, but so what? *American Fugitive* is all about the action, baby, and it delivers that by the bucketful.

The slick, isometric view peels back the layers of Redrock County, a seemingly sleepy US town in which you'll be tasked to help out an array of contacts in order to help uncover who killed your pa. Your associates include your dodgy brother-in-law, femme fatale Ana, and a dubious undertaker, and while the missions are chiefly fetch-quests, they're enjoyable enough, often requiring the careful casing of buildings before you ransack the place and run off with someone else's family silver.

There's a lot of truly lovely mechanics in the mix, too, from Will's near-constant need to update his wardrobe – the cops can accurately,

if inexplicably, describe your outfit from the other side of a wall sometimes – to respraying key vehicles to thwart their APBs. Both Riley himself and the vehicles he uses are attributed 'wanted stars', and as navigating the town without driving into the townsfolk requires Herculean efforts, you'll find yourself swiping clothes from unattended washing lines more often than you'd like.

The problem is further compounded by Redrock's idiotic inhabitants, who are forever stumbling into your way, and the top-down view that makes you repeatedly rear-end innocent

drivers. With no mini-map, there's no way to anticipate these hurdles, which means even the most law-abiding tasks will likely draw the attention of the rozzers. Most vehicles drive

like they're on ice – the physics are outrageously floaty – and the shooting mechanics also feel a little too loose and unremarkable. The numerous unlockable upgrades help mediate some of these issues – you can increase your inventory space, learn to restrain hostages, and so on – but not enough to make *American Fugitive's* gun- or vehicle-play ever feel truly satisfying.

Despite the copy-and-pasted vehicles (I was once idling at a stop sign with what I swear were four identical cars), repetitive quests, a weirdly intrusive stamina gauge, and one too many cringy 'jokes' (is it really that funny having to respray a foe's truck pink or putting Will in a dress?), *American Fugitive* might not be particularly challenging, but it's undeniably good fun. 🗣️

“An unashamed love letter to the original GTA”

Info

GENRE

Open world

FORMAT

XBO (tested) / PC / PS4 / Switch

DEVELOPER

Fallen Tree Games

PUBLISHER

Curve Digital

PRICE

£17.99

RELEASE

Out Now

REVIEWED BY

Vikki Blake

VERDICT

An imperfect, but perfectly adequate, crime time adventure.

64%

Void Bastards

Shut that door

Info

GENRE
FPS, Roguelike

FORMAT
PC (tested) / XBO

DEVELOPER
Blue Manchu

PUBLISHER
Humble Bundle

PRICE
£24.99

RELEASE
Out now

REVIEWED BY
Ian Dransfield

HIGHLIGHT

The way citizens are given visual 'audio' cues – a 'tap, tap' pops up on screen as you're walking through a door, for example – is a wonderfully elegant way of telling (and teaching) the player what it is they need to prepare themselves for. 'Squelch' isn't scary. 'Stomp' is worrisome.

✓ You spend a lot of time looking at the map, figuring out your next move (and door to lock).



W

ith 37 seconds of oxygen remaining, a grand total of seven bullets to my name, an irradiated body, and mere metres to go before reaching

the solace of my boarding (and escape) craft, S.T.E.V, I hear it. It's behind me. It alerts me. Warns me of my doom. "Oi, knobhead!" I die, shot in the back by a mutated child with an attitude problem. Ah well, time to rehydrate another convict and start over, methinks.

You can throw a million and six different labels at *Void Bastards*, the latest from Jonathan Chey's Blue Manchu. It looks like *Borderlands*, it plays like *BioShock* and *System Shock*, and there's a chunk of *FTL: Faster Than Light* crushed up and sprinkled on top. There's even forgotten rap satire and Northern comedy god Devo in there, unless my ears really are trying to fool me on encountering the game's 'Juve' enemies. It's a lot of influences along with a lot of assured design, all crammed into one package. But really, *Void Bastards* is one thing: door management. Brilliant, beautiful, funny, and stressful door management.

See, you make your way from ship to ship in an unfriendly part of the universe, scrambling through half-functioning wrecks on the hunt for useful scrap with which you can build something – many things – to help you escape this galactic limbo you find yourself in. Each ship

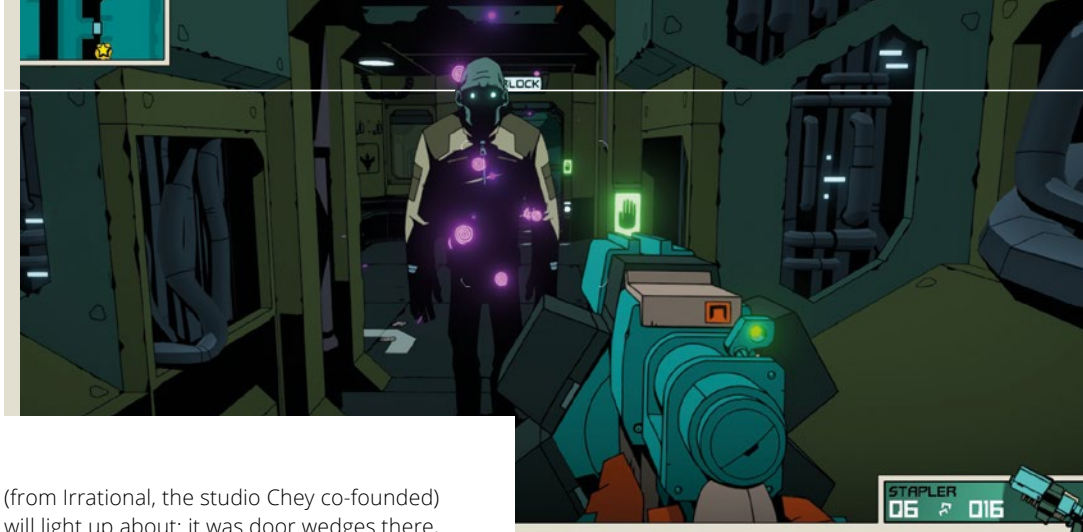
is procedurally generated, made up of a type (it could be a fuel tanker or an ambulance ship, maybe one full of cat robots or a luxury liner riddled with loot), and pieced together with different rooms in a classic tile-based format. At the basest of levels, your job is to get The Thing and not get killed while you're doing it. How best to do that? Door management.

You're taught from *Void Bastards'* opening tutorial that doors are (usually) your friends. Citizens – the mutated, void-spewing beasts infesting every ship you enter – are either unable to open doors full stop, or cannot open locked doors. As such, in a game where you're up against it most of the time, low on ammo, without the luxury of recharging health, and being pursued by a dogged little foul-mouthed child – it pays to engage in smart door management.

If it sounds like I'm getting a bit wrapped up in that aspect, then... well, you're right. I am. But this is a factor any player of *SWAT 4*

✓ It's often not a good idea to stand and fight.





◀ It's not always aggression; sometimes citizens are on your side. Rarely.

(from Irrational, the studio Chey co-founded) will light up about: it was door wedges there, it's locks with a stressful countdown to engage in here, but the thinking is the same. Check out the level presented to you at the start. Know where you need to go. Know how you're going to get there. And make sure you're funnelling those antagonists wherever you can, so you can control them to the best of your abilities.

Door management – I promise I'll stop saying it so much now – is a key element of *Void Bastards*, but it's not the main draw, as much as I am both obsessive about it and *want* it to be key. It's just a highlight for someone with a desire to control the destiny ('direction') of others like me. Others will see the broad selection of weapons and gadgets on show and realise this is a game that wants you to experiment with your approach. Others will adopt a stealthy approach and sneak their way as much as possible through a ship, distracting and confounding enemies and causing as little physical mischief as possible. If you're playing on easy, you can even try and play *Void Bastards* like *Doom* – though anybody trying that on the harder difficulties will just run out of ammunition and die a lot.

Which handily brings us to the roguelike aspects of *Void Bastards*: you die a lot, and you're meant to. Couched in a genuinely quite funny story, players take on the role of multiple prisoners, one after the other, randomly spawned when the previous con meets their end. You'll build up a collection of bits and pieces to construct items with, perks, and ammo, eventually feeling like you're actually a bit of an established force in the universe; a one-person army capable of facing down anything.

I won't say *Void Bastards* gets incredibly difficult, but that whole 'facing down anything' thing doesn't come true. As you progress, it does get harder. More enemies, tougher enemies, more environmental hazards, ships with periodic power outages forcing you to route back to the generator room multiple times, less oxygen in your tank, and so on and so forth. Every step forward you take, *Void Bastards* takes one of its own. But with every one of those steps you learn

– when to be careful, when to stand and fight, when to run away, and when to engage in some smart door management. I didn't say I'd *stop* talking about it.

There are inherent limitations borne of *Void Bastards'* procedurally generated nature –

ship layouts can be rather samey – and after a while, a sense of tedium does start making itself known. Even with repetition as an acknowledged part of the

game's story, having to go through things again after losing a prisoner becomes a bit annoying on the 25th occasion. And there's that thing that comes with roguelikes, in that sometimes it's just patently unfair. So yes, the game does have a few issues holding it back from true greatness.

But *Void Bastards* is great fun. It's a confidently made game, aspiring not to be like all of those other titles it draws influence from, but to be its own impressive little package. It's not quite as deep as it first seems, but given the right approach, that can work in its favour – this is one to be played over and over, changing up your playstyle and attempting new, different things each time. I wouldn't say it shifts any paradigms, and not just because that's an awful turn of phrase, but *Void Bastards* does earn itself a place in my list of Recommended Games, subsection 'door management'. 🗨

“Void Bastards is one thing: door management”

VERDICT

It won't keep you playing forever, but you'll enjoy the time *Void Bastards* does give you.

78%



▲ The comic book presentation is superb throughout.

Rated
Review



Review

HIGHLIGHT

Mordhau blends the ludicrous and the intense seamlessly. In one life I survived being disarmed twice: the first time by an enemy accidentally killing their teammate, the second by an ally jumping from a cliff for a 'death from above' attack, dying a split second later from the fall damage. I was too busy laughing to notice my head roll off my shoulders a few seconds later thanks to an enemy with a Zweihänder.

▲ Melees are huge, loud, messy, and so much fun.

Mordhau

The Black Knight always triumphs

Info

GENRE
Medieval combat
FORMAT
PC (tested)
DEVELOPER
Triternion
PUBLISHER
Triternion
PRICE
£23.79
RELEASE
Out now

REVIEWED BY
Joe Parlock

VERDICT

What *Mordhau* lacks in content, it makes up for in silliness, brutality, and excitement. You're not going to get an experience like this anywhere else.

80%

Poets, novelists, and artists alike have all dealt with the horrors of war. The iron tang of blood in the air, the screams of the wounded, the seconds before your death stretched out over an eternity. They rarely show a bard prancing around the battlefield, playing the lute and taunting friend and foe alike in a foppish voice. *Mordhau* does, and I don't think I've laughed this much at a multiplayer game in years.

As a spiritual successor to *Chivalry: Medieval Warfare*, *Mordhau* is all about waging war with swords, axes, bows, horses, and siege weaponry. You're able to create your own character and kit them out with the very best in 12th-century technology, unlocking new cosmetics and upgrades as you rise through the ranks.

Through a detailed combat system that relies on the type of swing, grip, and stance, the combat has so much nuance that mastering it is a daunting proposition. For those who've played *Chivalry*, it'll seem familiar, with old tricks like swinging your mouse in the opposite direction to delay an attack being just as much a part of *Mordhau*, but it's still got enough of its own

mechanics to feel like a total overhaul, rather than a clone. Most weapons have an alternative grip that come with their own advantages. Horses are a big part of the battlefield, with players riding around and dealing massive damage at the cost of becoming the single biggest target in play. Siege weaponry can be found around the maps and used to rain death down on the enemy team (and unlucky teammates).

For all of its technical gameplay and sizeable learning curve, *Mordhau* manages to be fun even for newcomers, thanks to how incongruously silly it is. Disarming an opponent can be more literal than expected, as lopping off limbs to make them drop their weapon is a totally viable strategy, as is distracting the enemy with some fancy lute playing before smashing their skull wide open with it. It's a frenetic, lively game that just exudes fun from the start of the match to the end.

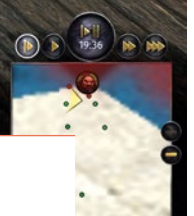
The only major complaint to be made about *Mordhau* is its lack of playable content. It includes three modes in standard matchmaking – Battle Royale, Horde, and a capture-point mode called Frontline – but only Frontline really allows for the best elements of *Mordhau* to shine through. There's also a ludicrously small map pool of seven, with three of those exclusive to custom server deathmatch modes not on the normal matchmaking rotation.

Mordhau is a stellar game. It is a bit slim in its offerings at present, but you'll be hard-pressed to find an experience like it anywhere else. Daft yet deep, hilarious yet brutal, *Chivalry* has finally been dethroned as the go-to for multiplayer medieval chaos. 🐘



HIGHLIGHT

As complex and involving as the politics are, there's still nothing quite like charging armoured cavalry through a flaming catapult breach in a city's walls, underneath the bloodshot haze of a setting sun. Both the campaign map and the fields and cities you'll do battle over are alive with gorgeous, painterly detail.



AND AT CHANGBAN
6 595
AUTUMN 20

Review

Total War: Three Kingdoms

A masterful reinvigoration of the Total War series

After letting your players order giant rats to drop nuclear weapons on hordes of pirate vampires, how do you make blokes with spears exciting again? *Total War: Three Kingdoms* has an answer to that. Rather than staying true to the letter of history, you get lost in the romance of it. Tense duels between immortal folk heroes. Legendary friendships and clandestine conspiracies. Song-worthy victories and crushing defeats. *Three Kingdoms* doesn't come close to the sheer variety and spectacle of the *Total War: Warhammer* series, and that's absolutely fine. Instead, it masterfully reinvigorates the fundamentals of a two-decade-spanning franchise. The result is the most gorgeous, sleek, and accessibly complex *Total War* yet.

For those new to the series, *Total War* is a game of three elements. The first is a sprawling, geographical campaign map divided into provinces and settlements. On the campaign map, players upgrade their own settlements, conquer their opponents', and recruit and manoeuvre their armies. The second is grand, real-time battles. Grand matches of elaborate rock-paper-scissors featuring thousands of individual combatants, where cavalry charges, flanking manoeuvres, and morale come together to reward both authentic historical tactics and a healthy serving of the finest cheese.

The third element – and the one *Three Kingdoms* makes the greatest advancements

in – is the diplomatic relationships not only between factions, but between the named characters in those factions. Faction leaders have distinct personalities, influencing their diplomatic decisions. Your own betrayals and honourable acts echo across China, affecting your future relationships. Righteous characters may balk at your underhanded decisions, while warlike generals grow dissatisfied at languishing in extended peacetime. Attempting to enact a diplomatic treatise provides a detailed numerical breakdown of affecting factors – everything from how much of a strategic threat you pose to how much backstabbing you've done in the past.

All of this is conveyed through a sleek and readable UI, gorgeous painterly environments, and sound design that jumps from tranquil folk to pounding war drums. The series has had its fair share of shonky releases,

but the worst I've experienced is two or three soft crashes in 40 hours of play, and a handful of frame drops when battles got extremely hectic. If you're looking to put your rig through the gauntlet, there's even an 'ultra' unit size option, just to make the battles even more spectacular.

There's far, far more to *Three Kingdoms* than I'm able to go over here, but it's an experience that grips from the outset and only gets more enjoyable as you pick up the nuances. If you're new to the series, start with this. If you've been a fan of *Total War* in the past but taken a long break, this is just what you need to rekindle the romance.

“The most sleek and accessibly complex *Total War* yet”

Info

GENRE
Real Time Strategy

FORMAT
PC (tested) / MacOS / Linux

DEVELOPER
Creative Assembly

PUBLISHER
Sega

PRICE
£44.99

RELEASE
Out now

REVIEWED BY
Nic Reuben

VERDICT

The battles may seem like a downgrade from the *Total War: Warhammer* series, but almost every other aspect has been improved. A new benchmark for a venerable series.

82%

HIGHLIGHT

Normally, only the protagonist has the power to wield multiple personas. *PQ2*, however, lets everyone equip a second persona, which grants them additional skills as well as boosts their HP and SP. Essentially, you needn't rely on Mitsuru for ice attacks or default to Morgana as healer, but can diversify your party skills based on what personas you gain and fuse.



As certain characters bond in side quests, they'll learn new attacks called Unison – they look pretty flashy but also trigger randomly.

Info

GENRE
JRPG
dungeon crawler

FORMAT
3DS

DEVELOPER
P Studio

PUBLISHER
Deep Silver

PRICE
£34.99

RELEASE
Out now

REVIEWED BY
Alan Wen

VERDICT

A fitting 3DS swansong – and the closest Nintendo fans have to a *Persona* RPG, at least for now.

77%



Review

Persona Q2: New Cinema Labyrinth

A spin-off that sends the 3DS out on a high

▲ *PQ2* only has Japanese audio. Considering the Western versions of *P3* and *P4* only had English, it's a shame those familiar voices don't get to reprise their roles.

Switch owners crying out for a *Persona 5* port long before Joker came to *Super Smash Bros.* may protest against *Persona Q2* only being available on a console heading for retirement. But as you navigate this first-person dungeon crawler, drawing out maps and drag-and-dropping symbols onto the touchscreen, it's apparent that this simply wouldn't translate to a single screen without heavy compromise. As the 3DS's last major release, it makes effective use of its dual screen, a functionality I'm sad to see Nintendo is closing the book on after almost 15 years.

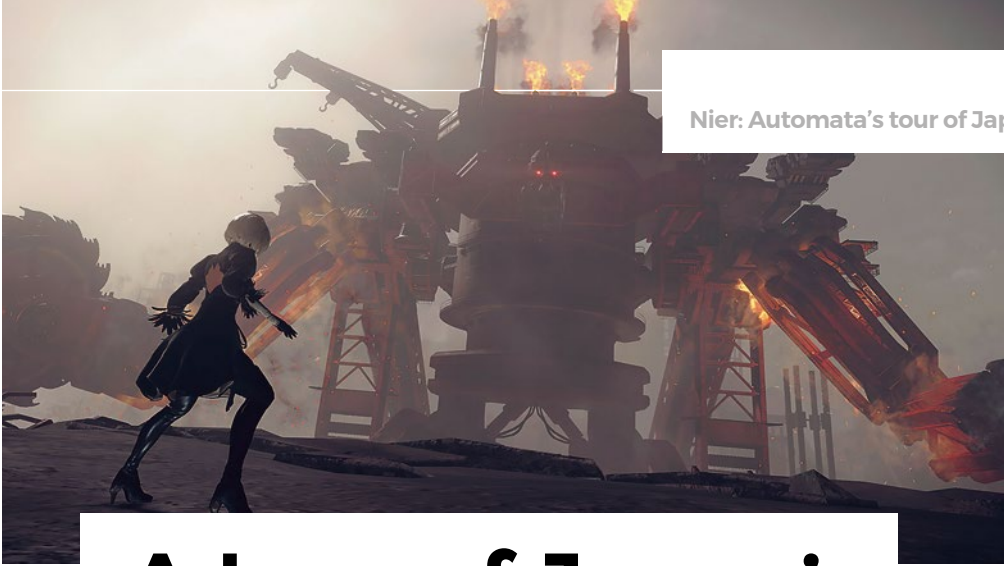
Of course, *PQ2*'s real draw is the fanservice of having the *Persona* series' SEES, Investigation Team, and Phantom Thieves come together in a mash-up adventure. Much like this universe, however, personal arcs and conflicts are suspended in time as you focus on helping newcomer and amnesiac Hikari escape a mysterious cinema filled with labyrinths based on different film genres, though each shares the central, albeit heavy-handed, theme of individuality versus conformity.

While Hikari isn't a playable character, this turns out to be a relief, since the final tally already has a whopping 25 characters – not including navigators Futaba, Rise, and Fuuka. When you can only have up to five members in your party, chances are most of the cast will fall by the wayside unless you're an ardent completionist. Nonetheless, the script does its admirable best

to include everyone, especially when it goes on banter tangents that trigger side quests, which can yield new items or skills. In the absence of the social elements from the mainline entries, these frequent moments of levity where characters bond or butt heads are a charming alternative.

Even if you're not a *Persona* fanatic, there's still a deep and hardcore RPG to enjoy. You aren't just wandering through the same labyrinths with different skins; each have their own unique mechanics and puzzles, the latter often taking the form of tougher FOE enemies you're meant to avoid (and who still put up a fight when you've levelled up significantly hours later). As skills cost HP or SP, and replenishing items are a scarce resource, turn-based battles are about exploiting enemy weaknesses, which gives your character a boost that lets them use their next skill for free. Continuing a boost chain becomes the key to lasting longer in a dungeon before you're forced to escape back into the hub to rest and restock.

Just beware of some nasty difficulty spikes, especially for bosses. Whether it's getting in multiple attacks in a single turn or a late surprise phase, some frankly take the cake. A few can feel like unfair battles of attrition; that early healing items aren't enough to get you back on your feet certainly doesn't help. But you'll want to get back on your feet, because whether it's listening to the catchy soundtrack, featuring all the vocalists from past *Persona* games, or indulging in some tongue-in-cheek callbacks, you couldn't ask to be trapped in a cinema with better company. ☺



A tour of Japan's game history

Nier: Automata's opening compresses decades of Japanese action games into a few glorious minutes

As video game openings go, I'd argue that *Nier: Automata's* is up there with the most captivating. In a planet's upper atmosphere, a squadron of futuristic fighter jets is gradually whittled down by enemy fire until just one craft remains. The lone pilot battles on, diving through clouds and scudding along a sunlit ocean, avoiding bullets as she goes. Then the craft transforms into a hovering, bipedal mecha, capable of blasting other mechanoid enemies out of the sky. Before we know it, the mecha we've just gotten used to flying has exploded into scrap, and we're controlling a sword-wielding android, hacking and spinning her way through a factory full of angry robots...

These opening moments pass by in such a dervish of chaos and spectacle, it's easy to miss just how smoothly it whisks us through a condensed tour of Japanese game history.

The first scene with the fighter jets is essentially a 2D, vertical shooter, like a beefed-up *Galaxian*; next, we're taken into the early eighties with a *Macross*-style transforming mecha fighting off bullets from all angles, recalling multi-directional shooters like *Time Pilot* or *Thunder Force II*. Then the camera takes us behind our craft, giving us a 3D rail-shooting experience like *Star Fox*; meanwhile, the rate of enemy fire increases to resemble a bullet hell shooter from the turn of the millennium. And then we land on solid ground for a frenetic bit of brawling akin to *Devil May Cry* or PlatinumGames' own *Bayonetta*. To cap it all off, the set-piece ends with the kind of wild boss

battle we've come to expect from Far Eastern action games: a fiery duel with a monstrosity that looks like an angry mechanical digger.

Did PlatinumGames intentionally stage *Nier: Automata's* opening to remind us of so many other games from its country's past? Possibly not, but there's no denying its impact, or the confidence of its execution. That opening also serves as an exhilarating entry point to a game

that, once you get into the meat of its post-apocalyptic RPG loop, requires a fair bit of patience as you explore, forage, batter oddly vulnerable-looking rank-and-file enemies, and generally improve your stats. In those occasional

moments of frustration (an awkward map system, copious invisible walls), *Nier: Automata's* opening slab of action echoes in the mind, both pointing ahead to similarly thrilling moments still to come in the game, and also pointing backwards, to some of the finest moments in Japan's proud gaming history. 🗨

"There's no denying its impact, or the confidence of its execution"



^ Whether you're tumbling through the sky as a fighter jet, or fighting a gigantic boss with a samurai sword, *Nier: Automata's* opening is one for the ages.

Wireframe Recommends



Infinite Space

NINTENDO DS

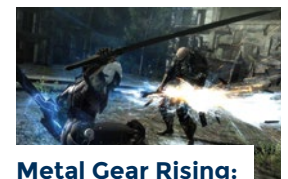
People don't really talk about PlatinumGames' RPG-RTS space sim all that much these days, but it's well worth a second look. A dinky space opera in the palm of your hand.



The Wonderful 101

WII U

Yes, it had its flaws, but Hideki Kamiya's superhero pile-on was one of the most imaginative action games on the Wii U. We're still crossing our fingers for that fabled Switch port.



Metal Gear Rising: Revengeance

PS3, XBOX 360, PC

For sheer mayhem, *Revengeance* was difficult to beat, even if its connection to the *Metal Gear* series was tenuous. Hacking dinosaur-like robots to pieces in slow-motion with a massive sword? Poetry. Sheer poetry.



Far Cry 3

Not the first, certainly not the last, but it made outposts *great*

UBISOFT MONTREAL / 2012 / MULTI

Far Cry 3 wasn't even the first game in the series to feature outposts – they'd been a staple since day one. This isn't a case of a game introducing a brand new feature that changed all of gaming forever. No, what *Far Cry 3* did with outposts was take what was already there, and make it – by our calculations – approximately ten billion times better.

Where outposts were once vague affairs – more a collection of enemies and a few shacks than anything else, with said enemies respawning the minute you dared look away for more than half a second – *Far Cry 3* made them a core pillar of the game. These islands were in the grip of a menace. You needed to liberate them. You did this by claiming land as your own, thus spreading your benevolent (or malevolent) whims and freeing the people. Outposts in *Far Cry 3* are, by any stretch of the imagination, a very straightforward gameplay mechanic.

But *what* a mechanic. Where *Far Cry 2* made you cry tears of fury almost every time you encountered a detachment of enemies, the third game made it a challenge – an exercise in balletic fury, or a clusterfuck of epic proportions. Here's a set area, it has a set number of enemies (and potential reinforcements), you know how the mechanics work (disabling alarms, blowing up barrels and so on): now get to it.

A microcosm of tight, well-made game elements – both a test for your abilities as a player, as well as a testing ground for

ways to actually play the game. One attempt might see stealth and slinking the order of the day, while the next brings with it rocket-propelled carnage on a grand scale. There's no wrong answer – only the chance to try again if it goes awry.

And, best of all, once an outpost is liberated, there's an actual impact in the game world – it's taken over by your team, the map becomes less hostile, new missions unlock, better weapons become available. It all meshes so utterly perfectly that it's easy to overlook just how well made *Far Cry 3*'s outposts are.

Spin-off *Blood Dragon* hammered home how much fun it was to liberate outposts by basing the entire game around doing just that, with little in the way of anything else to bother with.

Sure, it did get a bit samey after a time, but all you'd have to do was mix up your approach again and it felt fresh once more. Outposts are one of the top examples of a solid mechanic, done well – one that elevates the experience beyond the sum of its parts and makes it into a vital, exciting and, above all, fun thing to play.

Which is exactly why outposts are now one of the most overdone elements of any game, not least of which in Ubisoft's own output. That's not to say they're pale imitations – *Metal Gear Solid V*'s take on outpost conquering is some of the best in any game, period. But when everybody's jumping on the bandwagon, it does become rote – and in turn, a bit boring. A killer feature? Absolutely. A cautionary tale? Definitely. 🗨

“It meshes so perfectly that it's easy to overlook how great outposts are”

Next Issue

ON SALE 4 JULY

“It’s about as close to a massive triple-A game as you can get **WITHOUT** abusing employees!”

Don't miss out!
Subscribe
PG54

Team Meat's Tommy Refenes talks exclusively to us about Super Meat Boy Forever

Also

- ▶ Introversion: the Prison Architect studio's fall and rise
- ▶ The design genius of couch co-op games
- ▶ Death and depth in A Place for the Unwilling
- ▶ How games can create atmosphere with words

Editorial

Editor

Ryan Lambie
Email ryan.lambie@raspberrypi.org

Features Editor

Ian Dransfield
Email ian.dransfield@raspberrypi.org

Sub Editors

David Higgs & Vel Ilic

Design

criticalmedia.co.uk

Head of Design

Lee Allen

Designer

Harriet Knight

Contributors

Anna Blackwell, Vikki Blake, Rik Cross, Konstantinos Dimopoulos, Gareth Dutton, Kim Justice, Andrew King, Dia Lacina, Steve McNeil, Joe Parlock, Nic Reuben, Reid Schneider, Ryan Shah, Alan Wen, Steven T Wright, Jack Yarwood

Publishing

Publishing Director

Russell Barnes
Email russell@raspberrypi.org

Director of Communications

Liz Upton

CEO

Eben Upton

Advertising

Commercial Manager

Charlie Milligan
Email charlotte.milligan@raspberrypi.org
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 Rasperry 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).
ISSN: 2631-6722 (print), 2631-6730 (online).





JOIN THE PRO SQUAD

1ms

144Hz

Free Sync



RED EAGLE™

GB260HSU | GB270HSU | GB2760QSU



GET IN THE GAME

1ms
75Hz



Free Sync



BLACK HAWK™

G2288HS | G2530HSU | G2730HSU | GB2530HSU | GB2730HSU



IMMERSE YOURSELF IN THE GAME

4K



Free Sync



GOLD PHOENIX™

GB2888UHSU



ENTER A NEW DIMENSION

2560
x
1440

1ms
75Hz

Free Sync



SILVER CROW™

GB2783QSU