Wireframe

LIFTING THE LID ON VIDEO GAMES

FREE POSTER!

Wartales

Turn-based tactics from the makers of Northgard

Portal

Recreate Valve's iconic teleporting mechanic in Unreal

moonglow bay

Fishing, food, and friendship: the inside story of 2021's most relaxing RPG

Speakeasy

RichCast is an upcoming platform designed to let just about anyone make voice-activated, interactive experiences. Co-creator Philip Oliver tells us all about it
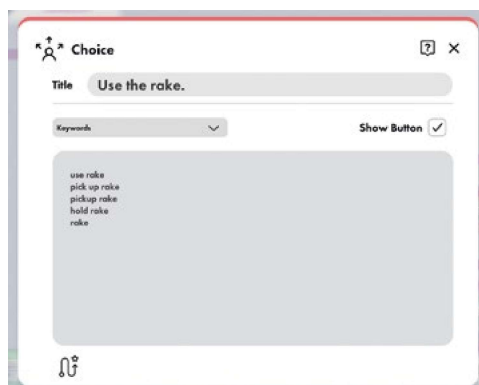
WRITTEN BY
RYAN LAMBIE

The explosion of creative, left-field indie games we've enjoyed over the past decade or so has been thanks in no small part to the growth of platforms like GameMaker, Unity, and Unreal Engine. Those tools have greatly lowered the barrier to entry for a generation of indie developers: without them, we may never have seen such games as, say, *Hotline Miami*, *Night in the Woods*, or *Hello Neighbor*. But what about potential storytellers who have no experience in programming? You may not need a computer science degree to make a piece of interactive fiction in platforms like Twine and Ink, but even those solutions require users to get to grips with their respective scripting languages.

> RichCast's voice recognition isn't flawless at present, but you can help it interpret the user's various spoken inputs by providing lots of alternatives for it to listen out for.

RichCast's interface, on the other hand, is almost entirely visual. It allows users to create interactive experiences using simple tiles and connectors, so if you can understand and draw flow charts, you'll be able to make games in RichCast – specifically, voice-activated, narrative games. *Miami Heist* is one early example game built in RichCast: it's an adventure where your objective is to break into a dodgy billionaire's mansion and steal one of his prized possessions, and you navigate your way around by speaking simple commands ('use rake', for example) into your mobile phone or PC. Narration and actors are brought to life with synthesised voices, while captions on the screen give you a better idea of the commands you can use to interact and progress.
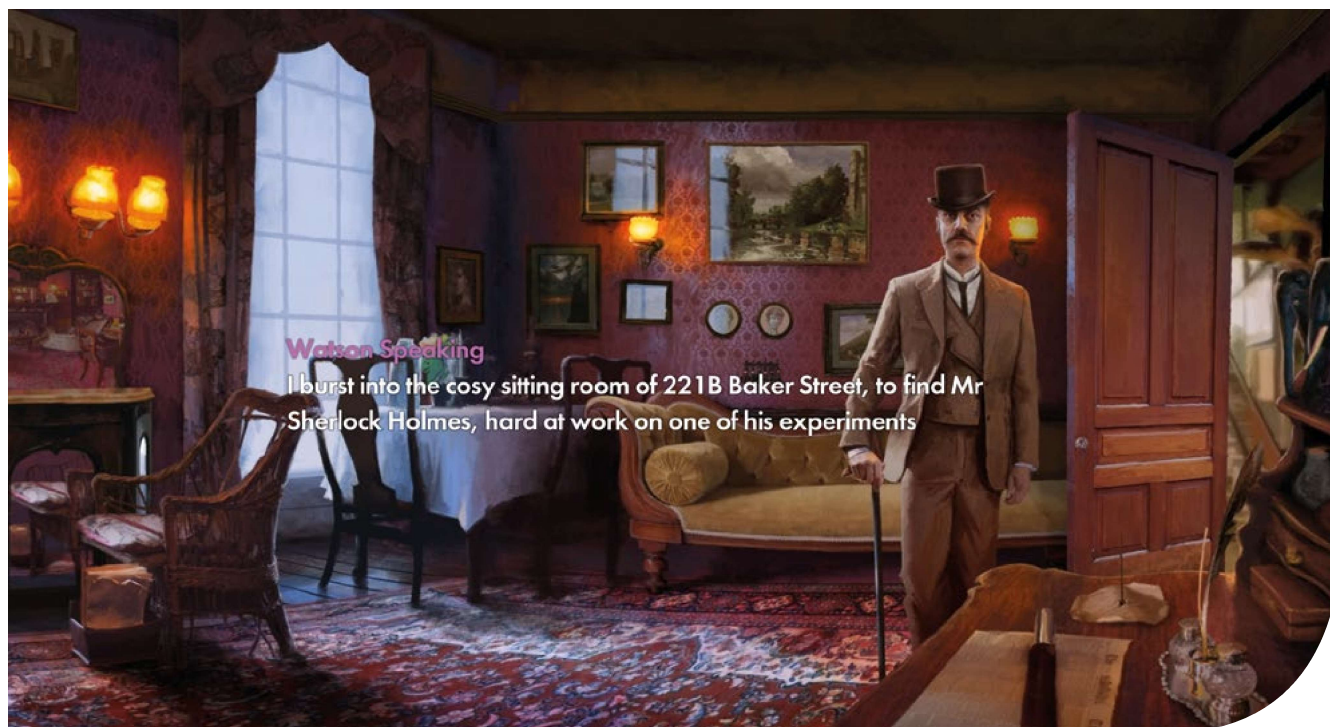
Open *Miami Heist* up in RichCast's editor (or 'Integrated Creative Studio'), and you'll immediately get an idea of how the tile-based system works. 'Speak' tiles contain the lines of script that the game reads out; 'Choice' tiles provide the branching paths that allow players to issue voice commands and choose their course through the game. Discrete areas in your project – whether they're chapters or individual scenes – are stored in separate pages, which can in turn be connected up to create the narrative flow.

With your tiles in place, you can then drag connective arrows between them, creating a sequential network of scene-setting images,

Watson Speaking
I burst into the cosy sitting room of 221B Baker Street, to find Mr Sherlock Holmes, hard at work on one of his experiments

sound effects and snippets of dialogue, player choices, and potential outcomes. In theory, it's akin to Unreal Engine's node-based Blueprints scripting system; in practice, it's much simpler, with its clean colours and chunky interface closer to the beginner-level programming language, Scratch. RichCast's system is a flexible one, though, and it's possible to create quite complex scenarios with the handful of tile types currently programmed into the platform – you can use Logic tiles, for example, to create conditions ("If the player's spoken to character X, then go to scene Y", for instance), or strings that keep track of items in the player's inventory.

RichCast is the latest venture from Philip and Andrew Oliver – the veteran British developers who've brought us such classics as the *Dizzy* series of adventures, *Super Robin Hood*, and *Grand Prix Simulator*. According to co-creator Philip Oliver, RichCast began with the idea of creating a platform specifically for non-programmers: "From the original concept, we said we need creators and writers to be able to use this system easily," he tells us. "We said, 'The minute you go to a script-based language – doesn't matter what that language is – it puts people off.' And so we were saying it has to be a graphic user interface, and the natural way to go then is to say, 'It's got to be a flow chart.'"

"We need creators and writers to be able to use this system easily"

When work began on RichCast, the process of creating its visual language was, as much as anything else, a process of paring things down: keeping the interface as clean and simple as possible, with a grid-based system that allows the user to create complex game loops that, with the aid of connective tiles called Junctions and the odd comment pinned here and there, can still be kept tidy and easy to comprehend. In design terms, Oliver compares RichCast to the Acorn's usage of the RISC (reduced instruction set computer) architecture in its computers. "Everybody was going more and more complicated, and then Acorn invented the RISC system and said, 'Actually, why don't we go simpler, and you'll just use more of them?'. So, in our case, that's little tiles – we've simplified what's normally programming. Other people do flow charts, like in Unreal, but they've always had nested systems inside each node, whereas, in RichCast, one tile is only one function. You're either going through it or you're not – it's binary. We've made it really simple to use."

Panivox, the company the Olivers and fellow industry veteran Neil Campbell formed to create RichCast, have big plans for the platform. They see it as an all-in-one system for voice-activated games and apps – an ecosystem where creators ➡

⌃ *Sherlock Holmes and the Eminent Explorer* **is an example of the kind of interactive mysteries that you could make in RichCast.**

More than fiction

RichCast isn't only for making games. As its name implies, it could also be used to make interactive podcasts – with voice-enabled commands for selecting chapters, for example, as Philip Oliver explains. "Hopefully we'll eventually get something like that: someone does a podcast, they break it into pieces, drop it into RichCast, and then index through it. We haven't got an example of one yet, but it's something we might commission just to prove it can be done easily."
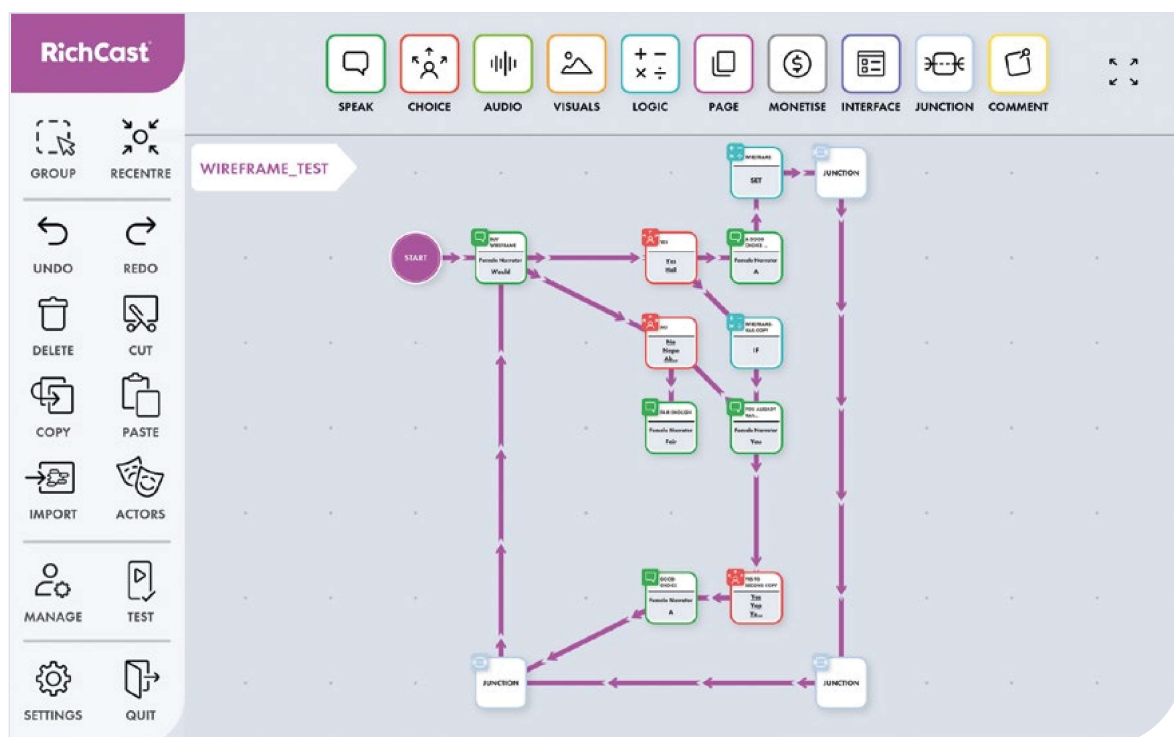
SPEAK The text from your game's script is stored in Speak tiles, and will be read out by your chosen synthesised 'actor'.

CHOICE Player interactions, from simple 'yes' or 'no' responses to more complex commands, are created with Choice tiles.

LOGIC You can use Logic tiles to track and update variables in your game, such as whether a player has picked up an item.

JUNCTION These tiles are essentially empty, but are useful for keeping your flow of tiles and connections neat and tidy.
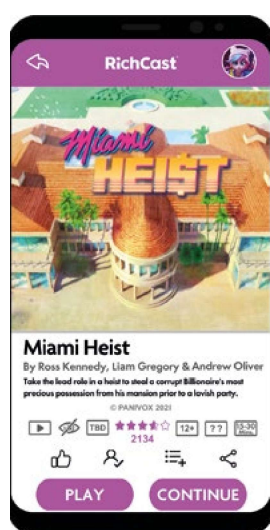
∧ Voice-activated games are a relatively novel experience, though some may feel a bit self-conscious about playing them on a crowded train, say.

will be able to publish their work and earn money from it. The details surrounding RichCast's monetisation are still being nailed down, but broadly, there'll be a range of subscription options for creators, ranging from just a few dollars per month at one end to around $50 per month for commercial developers, while players will either be able to purchase games using the platform's in-game currency (called RichCoins) or via a monthly subscription. There'll also be a free-to-use 'hobbyist' tier for creators, though this subset of users won't be eligible for any kind of revenue share; only subscribed creators will get a varying share of royalties from the sale of their game, depending on their chosen level of subscription. Philip Oliver compares RichCast to an entertainment platform like Amazon Prime or Apple TV, "where there's free content, but the majority of experiences have a price attached".

"What has been pointed out is how good it is for game jams"

Launching an entire creative platform does, of course, have risks attached to it, and Philip Oliver is cognisant of the challenges ahead. "First of all, there's lack of content," he says. "Hence why we have a content strategy, which is to commission people [to make games], then to do some competitions to motivate people to play with it and use it. We know that most of that stuff will be amateurish, but it will also be very random. And actually, random's good, because it might mean, 'Woah, you've done something really different and unexpected.' Then, hopefully, over time, there'll be a big enough user base, and we bring in monetisation, where writers will say, 'I want to write something serious [for RichCast] now, because there's an audience here for it.'"

It goes without saying, too, that RichCast is tightly focused on making a specific type of experience at present: it's not a platform like

Panivox co-founders Philip Oliver (left), Neil Campbell (middle), and Andrew Oliver (right).

Unity where you could make anything from a 2D platformer to, say, a 3D, open-world horse opera.

"The limitations in the short term are that it's not really designed to have lots of 3D models and sprites moving around the screen," Oliver says. "It's not a programming language to do that. It's purely for fiction and non-fiction stories at the moment. However, now that we've got quite an interesting and unique programming system, theoretically in the future it could be taken to those areas. It's a limitation at the moment, but we've made what we've originally envisaged – it's just that now we're going, 'Hmm, we can do all these other things as well.'"

In its current form, RichCast will, its creators hope, attract the kinds of would-be interactive fiction writers that are looking for an alternative to platforms like Episode and Twine. At the time of writing, Panivox has started to contact potential designers for the platform, and has

recently taken on three community managers to help shepherd what they expect will be a coming wave of creators and their content. "We're reaching out to more writers, and as of last week, we've started contacting writing groups, and going out to university lecturers, all of whom have said, 'This is the best programming language to get students into our games course.' We knew it'd be useful for universities, but we hadn't appreciated how useful... in fact, what has been pointed out is how good it is for game jams – in 24 or 48 hours, you can do a lot in RichCast."

Ultimately, RichCast's future lies with its users: if the platform can build up the ecosystem it needs to gain traction, then we could eventually see a diverse array of voice-activated games that go far beyond anything the Olivers originally envisaged. "We want novices to get their stuff on the platform, have people look at it, and then realise they need to get better," Oliver says. "All developers start that way – me and Andrew, our first games were pretty poor. We got paid very little for them – and rightly so, quite frankly – but then you say, 'Ah, the way to get more eyes, more money, is to improve the quality.' So with RichCast, I think we'll see what we often do with user-generated content. Some people just blow your mind with what they create."

RichCast will launch on PC/Mac in December, with iOS and Android to follow in early 2022. Sign up for Early Access at **richcast.com**. ⓦ

Improving Tech

According to Philip Oliver, the technology that underpins RichCast simply wasn't there even a few years ago. "When speech recognition was introduced in the noughties, half the reason it was terrible was because microphones were bad – the actual input signal was appalling. Whereas Alexa is so good – you can stick an Alexa in a room with the TV on and music playing and it still gets you every time. The mics are getting so good, and the software's better at filtering out noise. We're using third-party libraries, so a lot of this is down to them to keep improving. I'd say it's pretty good now, but it'll only get better. The same's true of artificial voices – in two to three years, it'll give real voice actors a run for their money."
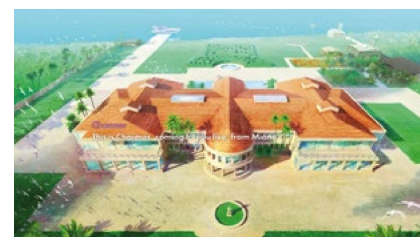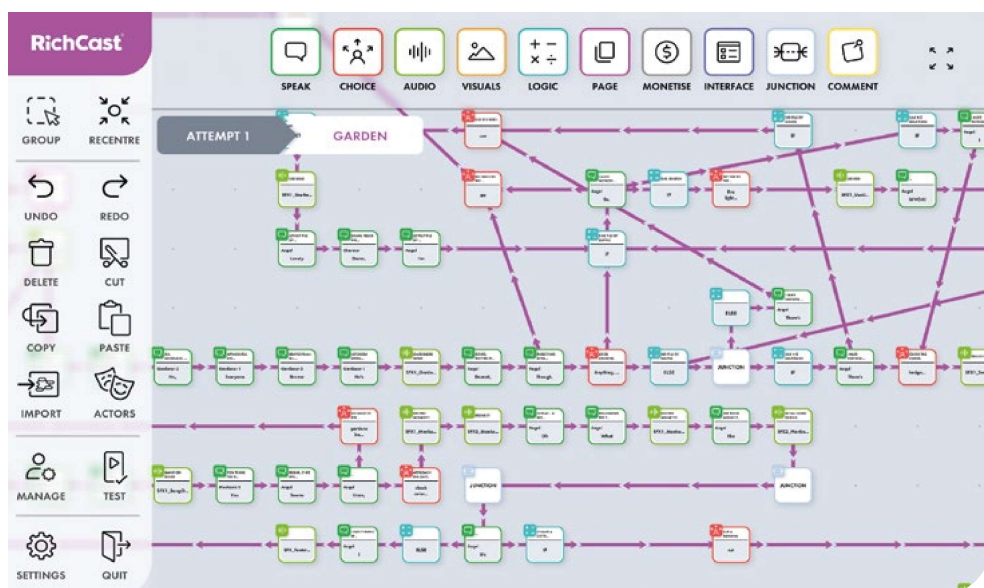
‹ A lengthier adventure like *Miami Heist* requires a hefty network of tiles and links.

FULL DISCLOSURE
Wireframe editor Ryan Lambie suggested an idea for a RichCast game, and to his surprise, Panivox liked it. It's currently a work in progress.