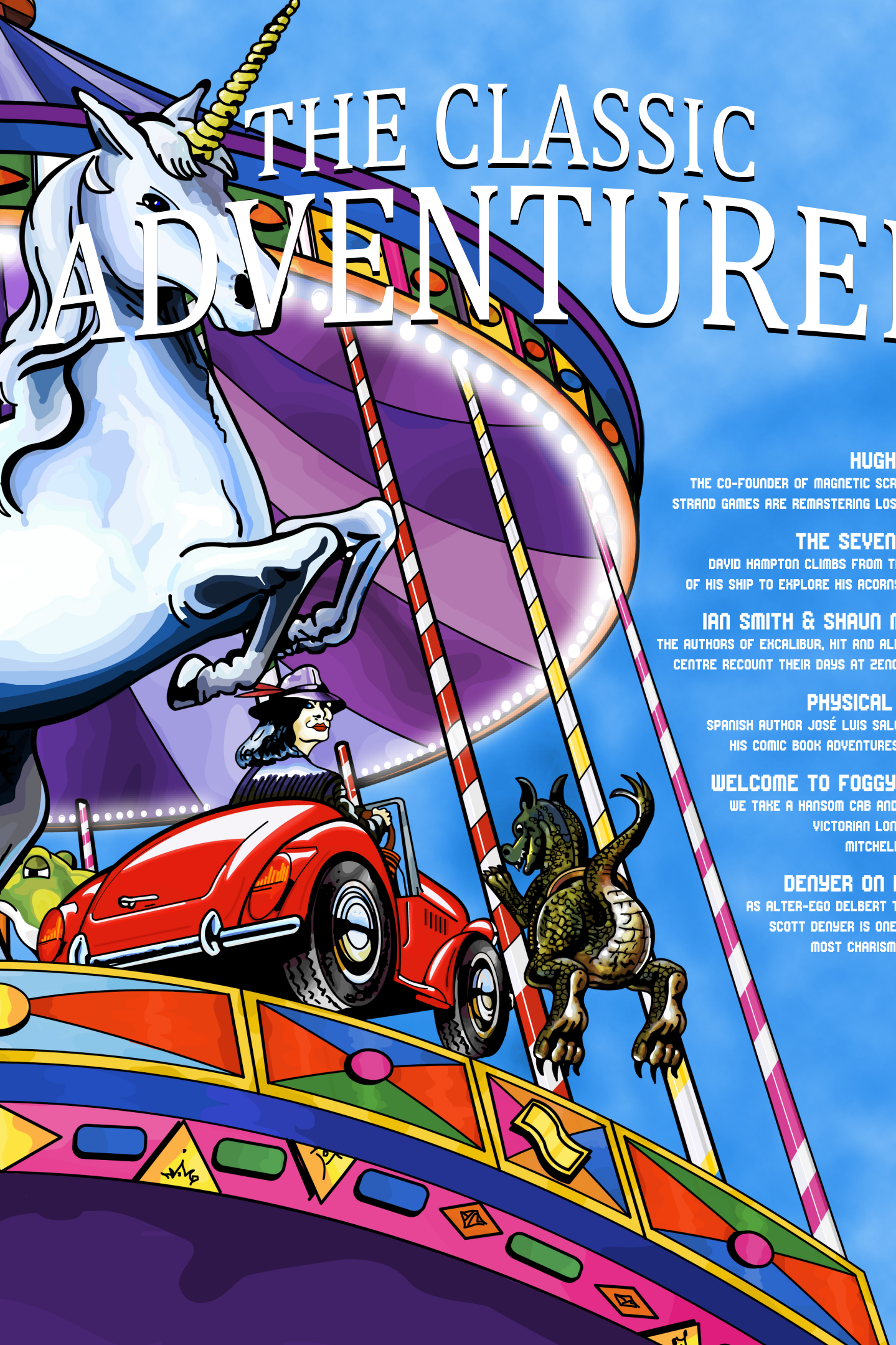
ISSUE 88

THE CLASSIC ADVENTURER



HUGH STEERS
THE CO-FOUNDER OF MAGNETIC SCROLLS ON HOW STRAND GAMES ARE REMASTERING LOST TREASURES

THE SEVENTH STAR
DAVID HAMPTON CLIMBS FROM THE WRECKAGE OF HIS SHIP TO EXPLORE HIS ACORNSOFT CLASSIC

IAN SMITH & SHAUN McCLURE
THE AUTHORS OF EXCALIBUR, HIT AND ALIEN RESEARCH CENTRE RECOUNT THEIR DAYS AT ZENOBI SOFTWARE

PHYSICAL DREAMS
SPANISH AUTHOR JOSÉ LUIS SALGUERO BRINGS HIS COMIC BOOK ADVENTURES TO THE MSX

WELCOME TO FOGGY LONDON
WE TAKE A HANSOM CAB AND INVESTIGATE VICTORIAN LONDON IN PHILIP MITCHELL'S SHERLOCK

DENYER ON DELBERT
AS ALTER-EGO DELBERT THE HAMSTER, SCOTT DENYER IS ONE OF BRITAIN'S MOST CHARISMATIC WRITERS

THE CLASSIC ADVENTURER

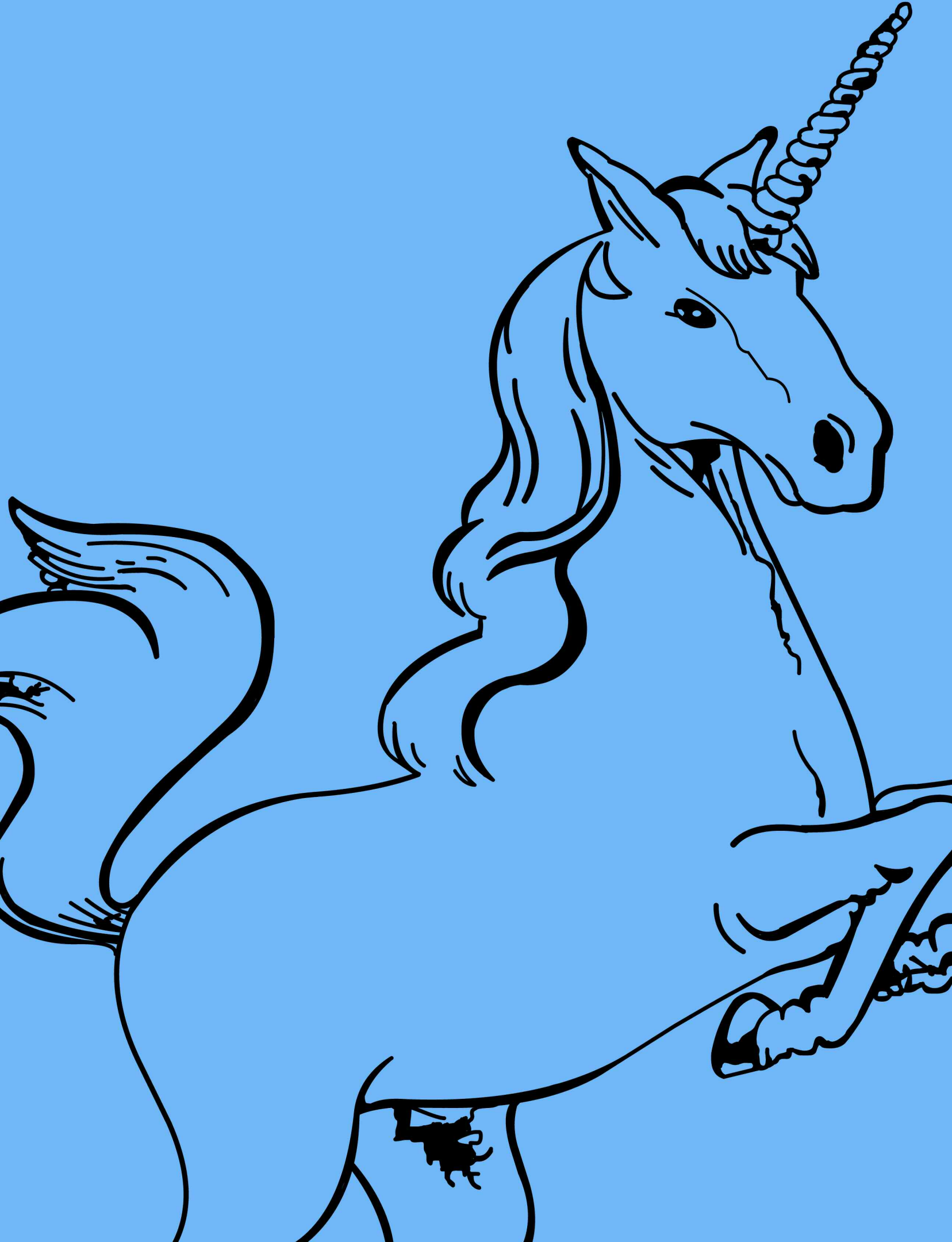
I owned an Acorn Electron as a kid. It wasn't the greatest games machine in the playground, but it did have the best game of all-time, Braben and Bell's *Elite*, and one of the best adventure games of all time, Trevor Hall's *Twin Kingdom Valley*.

For a boy with a fertile imagination, and an obsession with the Fighting Fantasy books, *Twin Kingdom Valley* whisked me through the screen, and into a fantasy world of babbling brooks, Forests, Orcs, Trolls, Goblins, Dragons, Kings and treasure!

I played as many adventures as I could, but It wasn't until I owned a ZX Spectrum and Fergus McNeill's *The Big Sleaze* that I encountered the same immersion with another game. A friend and I spent many weekends hunched over the keyboard, notepad and pen, determined that Sam Spillade would find the missing Maltese Bullfinch.

I'm therefore delighted that both Fergus and Trevor feature in this celebration of classic adventure games, along with many other adventures and authors that transported legions of other kids to far flung corners of their own imagination.

Mark James Hardisty, 2019



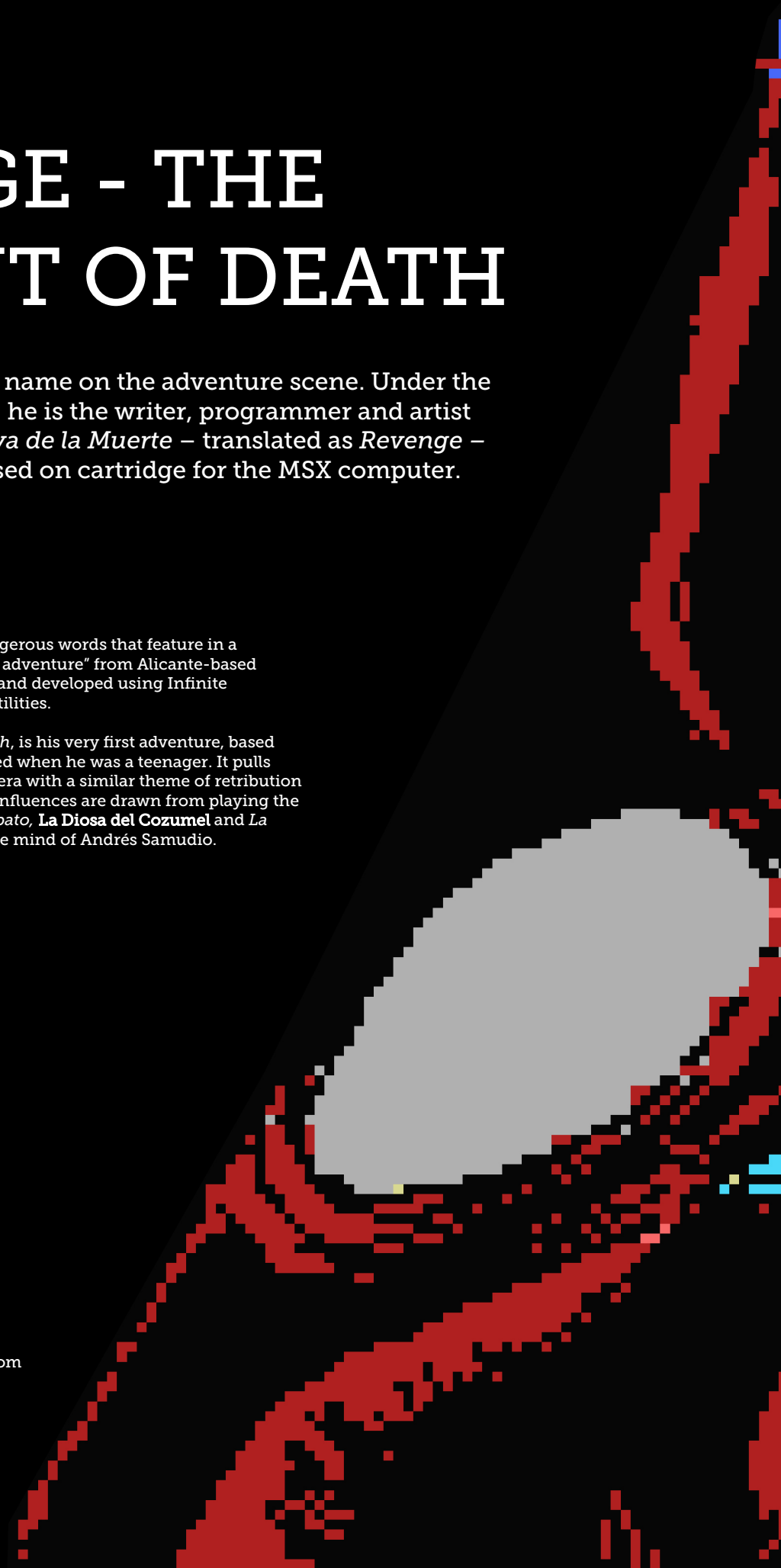
CONTENTS

REVENGE - THE SERVANT OF DEATH

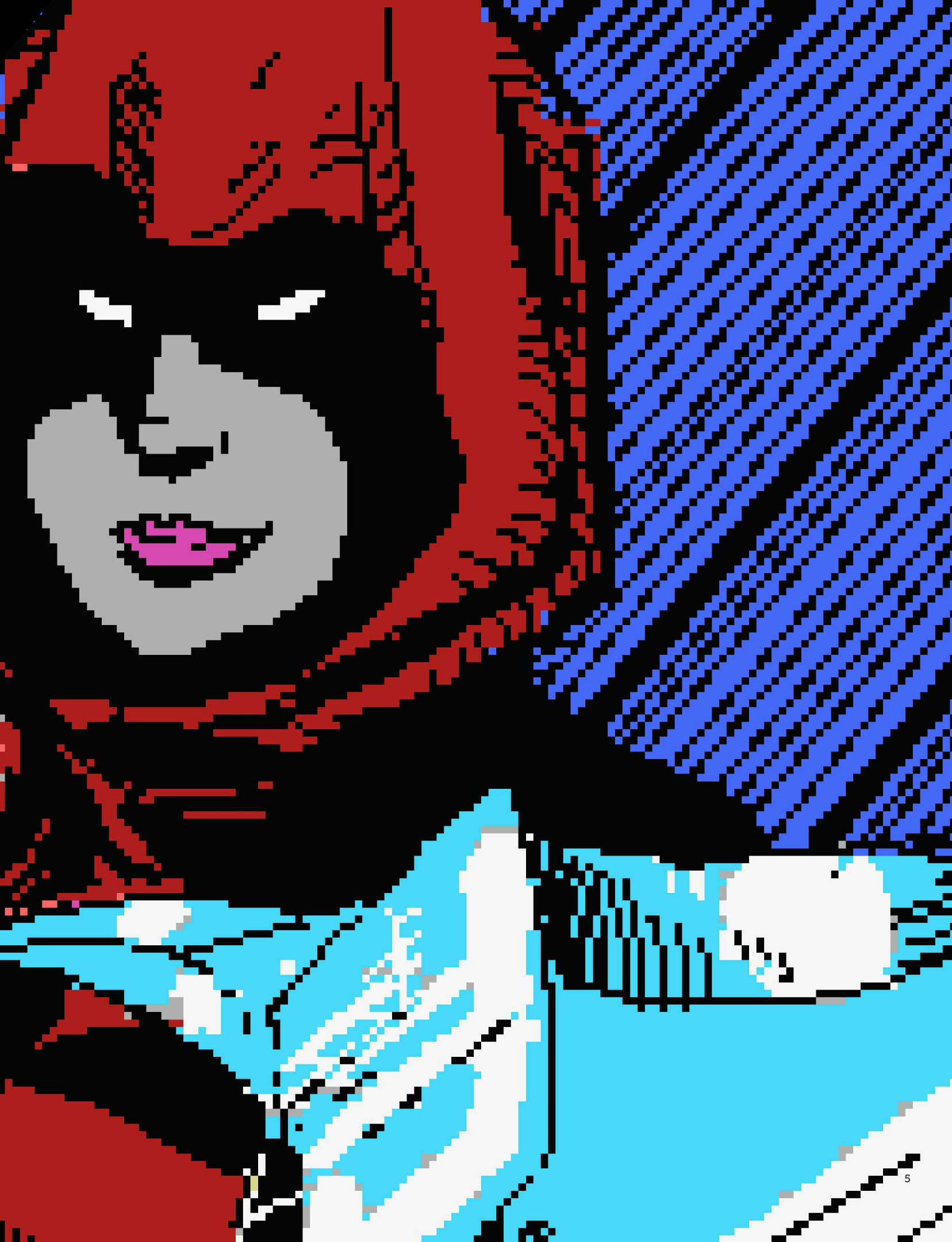
José Luis Salguero is a new name on the adventure scene. Under the banner of Physical Dreams, he is the writer, programmer and artist behind *Venganza – La Sierva de la Muerte* – translated as *Revenge – The Servant of Death*, released on cartridge for the MSX computer.

Revenge. Vengeance. Two dangerous words that feature in a contemporary "conversational adventure" from Alicante-based newcomer José Luis Salguero and developed using Infinite Imagination's *DAAD* suite of utilities.

Revenge - The Servant of Death, is his very first adventure, based upon a graphic novel he created when he was a teenager. It pulls from the many movies of that era with a similar theme of retribution and reprisal. His adventuring influences are drawn from playing the games of Aventuras AD - *El Jabato,* **La Diosa del Cozumel** and *La Aventura Original* – all from the mind of Andrés Samudio.



Format: MSX (Cartridge Only)
Publisher: Physical Dreams
Developer: José Luis Salguero
RRP: €28 - €33
Release Date: November 2019
Contact: physicaldreamsgames@gmail.com



You try to turn your head, but it's too hard... This stranger puts a hand on your head and it seems that he canalizes energy through him to you. Now you can turn your head, even move your hands. You're shocked, because you don't know what the hell is going on and the why of this recovery. The stranger takes a chair and sits in front of you, as waiting you to say something...

There's more...

You can't believe your quick recovery!! You can't avoid to look at the mirror on the wall. You discover that, when putting on the hood, your eyes turn a little redder and your skin whiter... "YOU'RE GOING TO FEED YOU ON BLOOD" the visitor tells you. And he continues, "EACH TIME YOU QUENCH YOUR THIRST REVENGE, KILLING ONE OF THEM, YOUR WEAPON WILL ABSORB THEIR BLOOD." Those words spin in your head and you don't yet know if this is real or an

There's more...

You climb to the roof, and just, from a hole, you see that you have under you one of those mother fuckers, but you can't go down without making a lot of noise and inform the others that you are here. Even though you use this costume, which gives you extra agility or something like powers, you can't put yourself at risk.

What do you want to do?
There's more...

You are at a place like a warehouse, quite big. It seems to be glued to a big flat. You can't go forward because there are 2 guys smoking at the entrance of the warehouse. You have to border it. Something inside you tells you that this 2 are not guilty, so it is not necessary to spill innocent blood. You can go to the EAST and to the WEST.

What do you want to do?

>

You start *Revenge* in a hospital bed where you are recovering from an unprovoked attack on your family. You're in a bad state, but as your consciousness and memory returns you begin to release that your family has fared worse; your husband and son are dead. A stranger soon arrives at your bedside, and bestows superpowers upon your person, hauling you back from a visit to the next life.

The narrative then follows you, as the female super-hero protagonist tracking down your family's murderers as a cape-wielding version of Kick Ass's Hit Girl. It's a mature, and often blood-soaked storyline through dark alleys, warehouses and situations of political and gangland intrigue. With the adult nature in mind, José has stamped the game with a parental advisory warning, due to the frequent use of profanity, and, presumably, from the violent, bloodlust theme itself.

Revenge's stylish location and character graphics, drawn using Photoshop by Selguro, ooze with character and hark back to the game's comic book roots. Its available on cartridge for the MSX range of computers and presents the player with a choice of two different editions of the game at start-up: The original version, conceived to be played in monochrome due to it's darker tone and noir feeling [the preferred option], or players can select the full-colour release, recently published by Salguero [and made free to existing purchasers of the game] if so desired.

Presentation is top notch, using the full capabilities of *DAAD* with the top-half of the window displaying neat, instant loading graphics and the bottom the location descriptions, the input line and the responses. The graphics themselves are loaded using the Maluva plug-in, created for *DAAD* by Uto and are very impressive, having been down-sampled from the original Photoshop images to be compatible with the SCREEN2 resolutions supported by the MSX. Occasionally there's a scrolling bug, perhaps due to *DAAD* itself, when the first element of text disappears from the screen before you can read it. On the MSX cartridge The Classic Adventurer tested there was a slight alignment issue that can be fixed switching from RF aerial cable to the more solid composite output of the MSX.

Anyone taking the time to create additional non-native language versions, as Salguro has, should be applauded. Unfortunately, there are a few flaws in the English version of the game that quickly become apparent in the text itself and also through the rigidity of the parser. The translation is littered with grammatical and structural errors. For example, in one of the opening scenes you see "2 guys smoking at the entrance of the warehouse" and "something inside you tells you that this 2 are not guilty." Trying to take an object, which is labelled as "A not very big stone" results in the response "You have take A not very big stone."

A 21st century adventurer also expects a little more hand-holding too. You are unable to GET objects but must TAKE them. Several items that appear within the game's text cannot be EXAMINEd. Trying to EXAMINE these uninteresting objects results in a time-consuming reprint of the location text, rather then a simple "you see nothing special" message, which ultimately discourages the player from exploring their surroundings. Given how easy it is to add multiple synonyms to the vocabulary and alternative inputs in the database, the parse is exceptionally specific in places, and is often far more strict than the one in the Spanish version of the game. For example, in one puzzle in the English edition, CLIMB UP WALL is needed rather than the more forgiving CLIMB WALL.

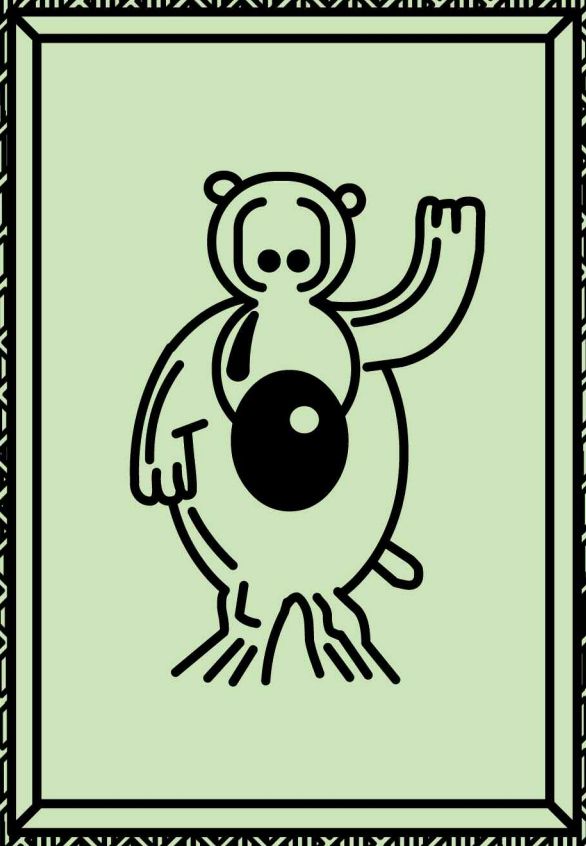
The game would undoubtedly have benefited from the feedback from a wider pool of native-English speakers before manufacture to ensure that the prose, grammar and code was correct. Nevertheless, the physical package that you receive [all produced, packaged and dispatched by Salguero himself] is professionally produced, especially considering that the media Physical Dreams have chosen to distribute the game on is cartridge. Some other retro producers support cheaper cassettes as an alternative, but Revenge comes in a high quality MegaDrive-style box complete with a small introductory manual and a cartridge that feels reassuringly robust.

The cover has more artwork and the manual is professionally printed, but does feel slightly homebrew-y in presentation. The manual gives some hints and tips for playing the game, along with a small section on how to play text adventures for the novice. There's a couple of translation spelling and grammatical errors, but the attraction of the package is in the game, rather than the packaging in this sense.

[Above] *Revenge* is distributed by Physical Dreams for MSX on cartridge in a professionally produced, hard plastic case featuring graphic novel style artwork by the game's author.

There is the promise of English ports to the Commodore 64 and ZX Spectrum in the future, and after that development moves onto the new *DAAD* adventure previewed in the back of the accompanying manual. For that, Salguero promises a more humorous adventure, with the same high-standard of graphics seen in *Revenge* - but for the MSX2 only.

Let's hope Physical Dreams engage with a wider group of English-speaking play testers for the upcoming game and ports. Salguero's work deserves to be more widely appreciated. Beneath the translation errors and playability issues is a solid adventure, with terrific graphics and artwork. It has a good storyline and characters of depth; particularly the protagonist herself who is, as the author intended, more than just an "executioner of revenge" and a "moth who has feelings." Unfortunately, with this many mistakes, and the gameplay gripes, it's hard to recommend, but if you can excuse them, or read Spanish, then pick-up *Revenge*. It's an atmospheric effort with some great-looking visuals.



SCOTT DENYER

Scott Denyer is one of Britain's larger-than-life and charismatic adventure writers. As alter ego Delbert The Hamster he authored some of the genre's more memorable and satirical adventures on the ZX Spectrum, and later teamed up with friend Gareth Pitchford to form a formidable creative partnership.

Scott Denyer was born in Luton and spent his early years flitting between Britain and the Middle East, where his father worked. After the birth of his sister in the early 1980s his family settled back in England and Scott got his hands on a ZX Spectrum computer, first by playing on his cousins' machine before being gifted a rubber-keyed 48K machine in 1984.

[Scott] I was immediately addicted, and all my pocket-money was spent on games and magazines. I used to get Your Spectrum, and then Your Sinclair, and Crash monthly, with the occasional Sinclair User or C&VG. My first foray into programming was by getting type-in magazines like Sinclair Programs and suchlike, then spending hours typing them in, but also designing cassette covers and faux posters for them. My first taste of the fun ahead running my own little software house.

Your first encounter with text adventures was playing Zenobi Software's *Behind Closed Doors*, given away free on a Your Sinclair covertape - is that true?

[Scott] That's not strictly true, I had dabbled in a few adventures previously, but not many, the standards, if you will. I had *The Hobbit*, which I played a lot, and also *Terrormolinos*, *Subsunk* and

***The Neverending Story*. I don't remember having any other adventures, and it wasn't until I stumbled into the mail-order world several years later that I really started to get into them.**

Behind Closed Doors was the game that demonstrated that adventures could be humourous and fun. What else appealed to you about the genre and made it different?

[Scott] [...] As I'd enjoyed other adventures previously. I think it was just that it seemed a lot more accessible than some of the previously played mainstream adventures, more anarchic and silly!

Did you complete it?

[Scott] Of course. I was never the best adventurer, but even I could complete *Behind Closed Doors*!

Did you play any adventures that you didn't like?

[Scott] [...] No others really spring to mind that I played to any proper degree. I must admit that in the early days I preferred a game that had some graphics, but then I was really quite young.

A "Dear Diary" interview for one of the fanzines recalls that you moved on to discover John Wilson's Bulbo character and found Bulbo the Lizard King "even better". Was this on a cover tape too?

[Scott] Yes, I believe it was. Bulbo was obviously more expansive than *Behind Closed Doors*, that being a one-location game and all. I just really bought into the spoof-y nature of those John Wilson games, and I was hooked.

You were studying for a Computer Studies qualification when adventures got in the way. Did you ever achieve a grade?

[Scott] Yes, but not particularly well. I got a C at computer studies. The exam didn't feature many questions about wizards or balrogs.

You were then inspired to write your first adventures in BASIC. Your first commercial game release was *Arnold The Adventurer*, was this your very first game?

[Scott] No, it wasn't Arnold. It was just a basic adventure with a few puzzles. To be fair, I can only remember the way the game looked visually, the typeface, cursor and suchlike. I can't remember anything about the actual game. I imagine it's on a cassette somewhere, but I don't remember seeing it since pre-Arnold days.

To aid your creativity, you purchased a copy of Gilsoft's *The Professional Adventure Writer*. For a teenager that must have been a hefty financial commitment at £24.95. Why didn't you buy the cheaper *The Quill*?

[Scott] I guess it was a big investment, but it was the best utility, an upgrade from the Quill, so it seemed the way to go. No point in buying the inferior software when the upgrade was available. I guess I was around 16 or so at the time.

What impact did Gilsoft's utilities have on you and other indie adventure writers having previously written in BASIC?

[Scott] I can't answer for others, and to be fair I'd only written a BASIC game for my friend to play, but as far as opening up the world of proper adventure writing, the impact was immeasurable. Suddenly, the world of adventure writing was your oyster. It was a wonderful piece of software, and I enjoyed the technical side of programming, the logistical side and problem-solving on that level, as much as the actual game-writing.

In 1990 you finished your very first commercial adventure, Arnold the Adventurer and sent it away to Zenobi Software. What was the inspiration behind the character?

[Scott] I don't remember. I just needed a hero character, and I like alliteration.

Arnold Tanglewood, himself featuring in three games, would be the first in a line of memorable characters to star in your games. Where did the rest come from?

[Scott] I don't remember being particularly inspired by any one thing, just sitting down and trying to come up with ideas, puzzles. Creatively speaking, I tend to get ideas from the everyday, so for example, if I was walking along and I saw a huge dragon attacking a nearby building, it might give me an idea for a plot-point or puzzle! I used to carry a notebook and jot down ideas if they came to me. Sometimes I'd use friends names for characters, just to entertain them and myself.

You authored games originally under a pseudonym Delbert The Hamster, why was that?

[Scott] I had a hamster, called Delbert, at the time, and I was probably inspired by John Wilson calling his label Zenobi after his cat. However, I regretted it pretty much straight away, as it didn't sound very serious, so it tended to guide the style of games that I produced.

After *Arnold*, you sought to publish your own games. The publishing agreement with John Wilson seemed to be productive, why didn't you continue that relationship?

[Scott] Oh, we did. When I wrote *Arnold* I wasn't really too aware of how the mail-order industry worked, and so it seemed easier, but also more prestigious, to ask him to release the game for me. However, I also liked the idea of doing my own thing, I always enjoyed the process of designing posters, marketing, that sort of stuff, so branched out on my own. However, for consistency, I still asked John to sell the *Arnold* sequels.

You created a catalogue of wonderful adventures, how did your style develop over time?

[Scott] I'd like to think that I became a better writer as things progressed, but I can't really say if that was true. I remember I did always try to do something different with each game, where possible, be it a cheeky bit of programming or whatnot. For example, in *Desmond and Gertrude* I liked the idea of playing as two characters, but then I didn't really make full use of that from the playing point of view, so it was rather pointless. In *Brian and the Dishonest Politician*, I really enjoyed creating lots of characters that moved around 'independently' during the game. From a programming point of view, I enjoyed that.

How long on average did it take you to write an adventure? They seemed to be delivered quite regularly?

[Scott] That absolutely depended on the scale of the game, there were some games that I didn't finish.

Did you follow a specific set of adventure game commandments, a set of rules that defined how you developed a narrative?

[Scott] Not as such, but as said I tried to stretch myself or try something new with each game, even if that wasn't immediately apparent to the player. I think my games all had a Delbert The Hamster 'feel' to them, but that was quite inescapable given that they were written by a 17 year-old fool.

So no baseline *PAWS* database from which to start games, i.e.a set vocabulary or locations?

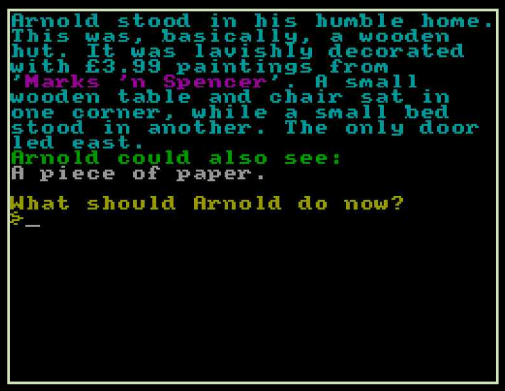
[Scott] Only the very basics, and I would use the same engine if I was writing say, an Arnold game or a Microfair Madness spin-off. Did I write any spin-offs for Microfair Madness, or did Gareth do those himself? He'd know!

How successful were your games?

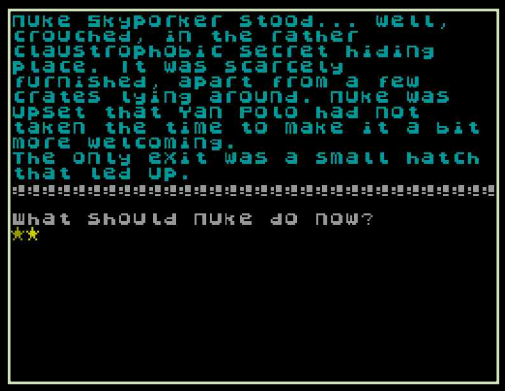
[Scott] Some of them were successful, for what they were, but it

DENYER ON DELBERT

With such a vast catalogue of eccentric and eclectic games, who better to guide readers of the **Classic Adventurer** through his softography than author Scott Denyer? We pick the best the bunch and ask the man himself his recollections of creating these adventuring gems.

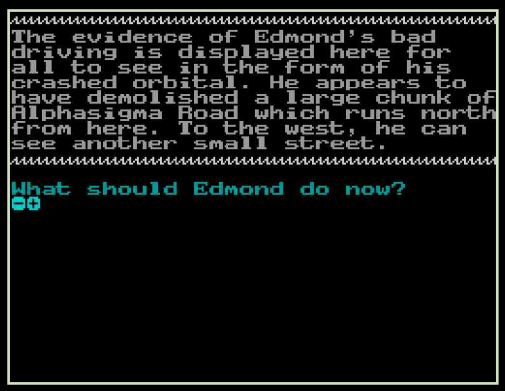


[Arnold the Adventurer] "My first proper adventure, very sweet and innocent, some terrible writing, but it didn't do anyone any harm, did it?"

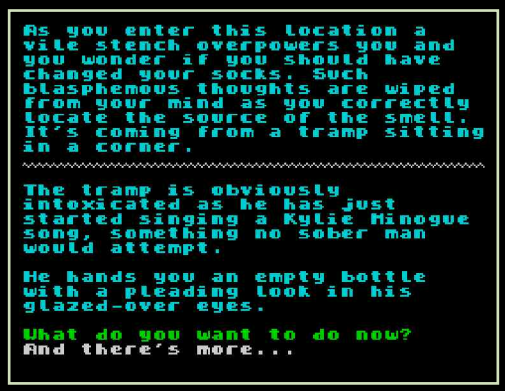


[Star Flaws] "My attempt at a spoof, a bit too juvenile and unsubtle. Still rankles that From Beyond had someone who had never seen Star Wars review the game, it got like 2 stars out of 10, I think. It was worth at least 3!"

[Desmond and Gertrude] "An attempt to write something more epic than *Arnold*, failed at that, but again a simple and unassuming start to the DTHS label."

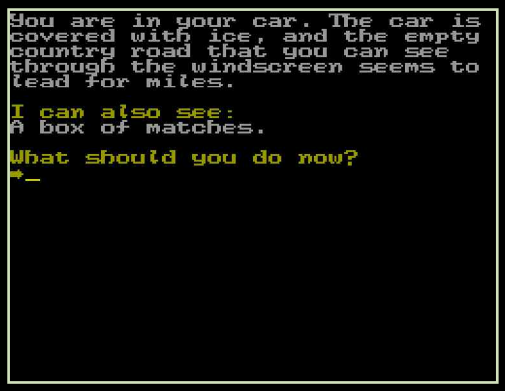


[The Life of a Lone Electron] "At heart a solid puzzle-solving game, but design and theme from Gareth were probably a bit esoteric for some. I remember thinking my 'look' for the game suited it quite well."



[Microfair Madness] "Very well written by Gareth Pitchford, and very well programmed by myself. I used every last byte to get that baby working!"

[Quest for the Holy Snail] "This was an extended version of a mini-game that appeared in *Microfair Madness*. I like how pure and simple it was, with fun puzzles and chucklesome gags, like most of Gareth's stuff."



[Snow Joke!] I" wrote it in one day when we were actually snowed in at home, to a degree. I added it as a little extra on a compilation. I wanted it to be a one-location game like *Behind Closed Doors*, and it was exactly like that, except lacking the humour and originality."

[Larry The Lemmings Urge For Extinction!] "I liked this one, a silly little game where the aim was to kill yourself, and poor Larry getting frustrated at every turn. I gave it a cartoon-y feel."

[Brian and the Dishonest Politician]
Technically much more accomplished and better written. I made it a two-parter, both parts very different, the second part being more run-of-the-mill that the first. The reviews said that the second part let it down as the first part was much more original. There I was trying to give people value for money, I should have just made it a one-parter!

[Grabbed by the Ghoulies] I tried to write a silly, haunted house adventure and I think I succeeded. Lean and mean!

[First Past the Post]
This was one of Gareth's designs, I don't remember much about this one. I think I did program the sequel, Get Me To The Church On Time!, and then Gareth finished the trilogy with Man About The House after I'd quit and sailed off.

entirely depended on the reviews. By that I mean that there were a core group of adventurers who would buy any game I published, completests if you will, but generally the sales figures were almost always based on how well the game had been reviewed. I couldn't tell you how many copies I sold of each game. I certainly didn't feel like I was selling big numbers compared to other software labels.

Aside from designing inlays and other artwork, did you enjoy the logistics of running your own indie label?

[Scott] Very much so, I loved the rigmarole of designing and illustrating covers, printing them, ordering jiffy bags and labels, packaging them off to soon-to-be-disappointed customers.

Which are your favourite and least favourite games, and why?

[Scott] Of my own games, I liked Arnold The Adventurer 3, because I thought each of those games got better, and I enjoyed *Grabbed by the Ghoulies*, which I think was the last complete adventure I finished. As I said earlier, I think my games generally got better as I became more versed in all aspects of the genre, as also as I grew from a spotty teenager into a slightly older less spotty teenager. When I stopped writing, I had written one part of a game called 'Exploits in a Wheelie-Bin', which remembering back I was very pleased with, and thought it was the best thing I'd written. I was also happy when I'd thought of a good puzzle and solution. I was also working on a four-part game called *The Legend of Caldor the Woodcutter,* which was me trying to grow up and write something akin to Larry Horsfield's stuff. I had the main character's wife and kid die in the first part, which is something you definitely didn't get in any of my previous offerings (except maybe Larry The Lemming's Urge For Extinction).

You formed a successful relationship with Gareth where he designed the adventures and you coded them. How successful was that partnership?

[Scott] I think it was very successful in that we had a great deal of fun, and produced the best game I was involved in, Microfair Madness. Gareth was more advanced than me in the pop-culture world, so the game was much more interesting and erudite than anything I'd written up to that point. He also didn't program at the time, so saw no issue in designing aspects of the game that were a pig to program. Still, that's what really stretched me, the programming-problem-solving that I really enjoyed, and ultimately made for a better gaming experience.

What did you see in Gareth's games that you liked?

[Scott] I didn't really know what to expect to begin with, he had put a call-out in Your Sinclair I think asking for a programmer, and I was desperate for material, so I got in touch and he sent me his stuff. As I mentioned earlier, his ideas and puzzles were more advanced than those I was writing, so I was happy for someone else to worry about the 'artistic' part of the game, and I just did the programming. It's funny to think that this was all done by letter, him sending me handwritten text, maps, designs, and then my programming and sending it to him, then him writing back with notes and whatever. It's lucky that we were both really into it, otherwise it could have taken years to finish one game! At that time too that I only had a Spectrum +2, so it was all cassette. I was at one of the Adventure Probe conventions, and I remember one programmer couldn't believe that I was programming using cassette. It was not long after that I upgraded to a +3 with a disk-drive. That sped things up, let me tell you.

You seemed to form a firm friendship, even making fun of Gareth in several games with various characters, including "Pitchfork".
[Scott] Indeed. Proper adult humour, there. There was also a Larry's Horse Field in that game, and numerous others I imagine.

After starting Delbert as a publishing label, you changed your author moniker to "The Spud", why was that?

[Scott] Yeah, I don't know why. Maybe I was angling for a one-word name, like Madonna or Prince? Or The Rock?

You mentioned various Adventure Probe conventions, did you take an activate part in the indie scene? Which other characters were around at that time?

[Scott] Pretty much most people that were about who went to the Adventure Probe conventions. There was also a gang that would meet up sporadically at a pub by Liverpool Street in London, the Hamilton Halls. I always remember that when I pass by even now.

How vibrant was it? We often forget that the indies released so many titles and games around subjects that would never have been considered by mainstream publishers?

[Scott] I think there was a fun camaraderie for everyone, especially as you say because it was off the beaten track. By and large it was a supportive, generous community.

You closed Delbert in November 1992 citing that you were too busy to write games and perform the work of a publisher. What brought about the curtain call, was it a move into a different career?

[Scott] It wasn't the career as such, I just left home to move to London to attend college. There wasn't the time, I was exploring the big wide world.

Were sales still strong enough to continue?

[Scott] Yes, and no. At that stage, it wasn't about the sales, per se. I was never going to be able to retire on it. That said, I guess if I was selling 1000 games a month I might have delayed college for a bit.

Did you still find enthusiasm to continue writing? 1992 was well beyond the commercial life of the 8-bit machiness?

[Scott] I did enjoy it all, but it was a pretty quick drop-off once I'd left home and wasn't involved in the scene any more. [...] I think the technology became too advanced to compete, we were entering the world of decent graphics adventures, so text adventuring was always destined to stay as a niche-market. That said, that's what allows it to continue to this day, as it doesn't need to keep up with the technical times!

Did any game ideas get left unfinished?

[Scott] Yes, those ones I mentioned earlier. 'Exploits' and 'Caldor', which I had programmed some of, plus I'm sure ideas and designs in notebooks and things.

What have you done post Delbert? What is your role now?

[Scott] I worked as an actor for a while, a stand-up comedian, comedy-producer, writer. At the moment I work for a news broadcaster in an operational capacity, and I'm writing a book, a horror novel.

There's been a mini-resurgence of the genre recently, with good IF titles on mobile, several games running through Kickstarter and the indie 8-bit scene returning with Gilsoft's tools and a complete DAAD suite of tools available to budding developers on all platforms. Have you considered re-visiting any adventures or writing any new ones?

[Scott] I've kept my eye it, via social media and whatnot, but I'm too busy at the moment to invest any proper time in it. That said, never say never.

DESERT ISLAND DUNGEONS

Whilst on a long voyage, the nautical steed housing shipmate **Scott Denyer** springs a leak and is forced to beach itself on a desert island. Scott is all alone with his faithful pet hamster and five text adventures to pass the time.

That's a toughie, because with something like games it seems silly to play games that I've already played.

I'd have to go with *The Hobbit*, just because. It's *The Hobbit*.

I'd then go with *The Axe Of Kolt*, because it's a classic by an old pal, Larry Horsfield, and I never completed it, so I could have another crack at that.

Oh, this is difficult, because I didn't play them as much once I started spending all my time programming. Any ideas?

I enjoyed the simplicity of Laurence Creighton's games, just two word solutions to all the puzzles, something like *The Pyramid* I enjoyed, I remember.

I was going to pick a Gareth Pitchford game, as he was a chum, but he never finished *Deception of The Mind's Eye*.

Adventure Writer

Technical Guide

DAAD TECHNICAL GUIDE

Andrés Samudio and Tim Gilberts of Infinite Imaginations have republished a much-needed member of the supporting documentation for the *DAAD* adventure development suite.

The reputation forged by Gilsoft for its industry leading adventure creation tools, *The Quill* and *The Professional Adventurer Writer* wasn't solely established on the quality of the software, but also on the high standard of documentation that was included with their products. Graeme Yeandle and Tim Gilberts' *The Quill* came with an extensive user manual, and with *PAWS* they extended the amount of information available to developers with a comprehensive technical reference guide.

The manuals were superb, covering every topic and every skill level. They held the hand of potential authors, first guiding them through the basics of game design and then explaining the programming and scripting required to implement the logic to power the adventures. Everything that the majority of authors required was at their fingertips in these printed marvels.

In the late 1980s Tim Gilberts left Gilsoft and *The Quill* and *PAWS* behind to focus on his fledgling consultancy business, Infinite Imaginations. One of his biggest contracts was to create an in-house multi-machine adventure writer for Adventuras AD – the legendary 'aventura conversacional' [conversational adventure] developers, founded in Valencia, Spain by Andres Samudio. The result of this work was *DAAD*, combining the capabilities of *PAWS* with Tim's other custom-built utility *System Without A Name* [created for Fergus McNeil's Abstract Concepts label] which produced one of the most advanced adventure tools ever built. Thanks to Andrés, *DAAD* was released into the public domain in 2014, but required the archival work of Tim and adventure aficionado Stefan Vogt to restore the English language version. *DAAD* is now gaining a new audience and a brand new lease of life in the retro adventure community. We're starting to see the arrival of updated ports, feature-extending plug-ins and brand new games from authors such as Gareth Pitchford, John Wilson, Stefan Vogt and Alberto Riera.

To compliment the recovered digital version of the documentation [and to help those old timers that still prefer reading from paper] Carlos Sanchez [with Tim and Andrés' blessing] has self-published the English language version of the *DAAD* technical manual. With its Gilsoft lineage, the quality of the manual is everything you'd expect from Infinite Imaginations and provides comprehensive instructions on how to get to grips with *DAAD's* development environment, interpretor, flags, errors and every other element of the suite.

Its recommended that for any wannabe *DAAD* adventurer writer they cut their teeth with the basics of *The Quill* or *PAWS* to get a good grasp of the programming and CONDACT [conditional action] logic required. After graduating, this 118 page book of goodness will be an invaluable addition to anyone's INVENTORY.

Author: Tim Gilberts
Publisher: Infinite Imaginations
RRP: £2.39 + VAT + Shipping
Buy it from: Lulu
Website: http://www.lulu.com/shop/tim-gilberts/aventuras-ad-daad/paperback/product-24334800.html

DAVID HAMPTON

One of the few Acornsoft text adventures that wasn't from the womb of Cambridge University's Phoenix machines was *The Seventh Star*. Written by David Hampton, the game featured a classic science-fiction setting, devious puzzles and a dose of British humour.

David Hampton was born in 1959 and grew up in the outskirts of Epsom, a small market town in Surrey, England. It was at technical college that he first had the opportunity to get hands-on and learn the inner workings of computers, crafting his programming skills by developing a prime number calculator and an early form of a machine-learning noughts and crosses game.

[David] [It was] far too ambitious for [my] second program but I did have a habit of failing to appreciate how difficult and time consuming something would be – it probably took me six months to get that damn thing to work but I managed it in the end.

As with many university and college students in the 1970s, David's first exposure to text adventures was when he discovered the mainframe version of Crowther and Woods' *Colossal Cave*. He played it with his friends, and the experience left an indelible mark on the young man, inspiring him, as it did many others, to have the enthusiasm to write a game of his own.

[David] I can only describe the feeling as being akin to a child's experience of Christmas – it was a magical feeling, playing the game, having my commands understood and acted on. I played for hours – all through the night on one occasion. I used to think about it a lot, wondering how it worked and gradually thought most of it through.

Towards the end of his studies, the explosion of the British home computer market was about to put small micros into the homes of many thousands of children and adults alike. During his fourth and final year at university, aged 22, his parents bought David a BBC

Micro, and he continued to experiment with programming and refined his coding knowledge by exploring an interest in anti-piracy systems.

[David] [I thought] how do you stop someone from simply loading a program into the computer's memory and then saving it as many times as they wanted onto original-quality tapes? Gradually I developed a protection system; I contacted some software companies and sold it to a company in Devon or Cornwall, I can't remember who. They paid me to protect some of their games and one of those was an adventure. As I looked at the code, the last piece of the puzzle fell into place.

David can't recall the name of the adventure, nor the company to which he was contracted, but by examining the code he could understand its workings, and he began documenting the way in which the engine numerically structured the rooms, dictionary and the rest of the functionality that he required to start writing his own game. He called it *The City of the Seventh Star*, drawing on the many science fiction and pop culture influences that were influencing teenagers and young adults.

[David] One day [I] decided: "I can do this! Let's start right now!". Apart from pinching the room numbers idea, I just wrote it intuitively. I only learned that there was something called an "adventure engine" last week. [...] I realised that the fun of these games was that found items would have unexpected uses that made sense once you had worked them out. I was never particularly interested in Dungeons and Dragons so the references to Sci-Fi films simply gave me a different way to introduce things with unexpected properties, and it was a way to find opportunities for humour as well.

Aliens, space exploration and distant worlds featured heavily in computer games, books and on celluloid of the late 1970s and early 1980s, and *The City of the Seventh Star* cast the player in a somewhat cliched storyline of an astronaut marooned on a desolate planet after their doomed spaceship had crash-landed. It was a very similar plotline to an existing adventure from Acornsoft, Peter Killworth's *Countdown To Doom*, published in 1982.

[David] To be honest I did not realise that the crash-landed spacecraft idea was a cliché – I didn't find out about *Countdown to Doom* until it was too late, which was a naive mistake. The whole thing had started as just a private fantasy about writing an adventure – so I wasn't really focusing on what would make it sell.

Over the coming months David started to add depth to the story and characters, and began to flesh out ideas for individual jokes and puzzles that would go into the *City of the Seventh Star*. He mapped out the adventure, using the traditional approach of pen and paper, adding in detail where needed for each puzzle that had to be solved: It was a very organic process, and although more adventures appeared on the market for the BBC between 1982 and 1983, mainly from the pen of Peter Killworth, Hampton remained quite isolated from external influences.

His personality and sense of mischief pushed him towards injecting as much humour into the script as possible. Contemporary adventures were austere and straight-laced affairs until Fergus McNeil and Delta 4 broke the trend with their satirical take on Tolkien's world in *Bored of the Rings* [published by CRL's Silversoft in 1985]. But it wasn't just the seriousness that David wanted to address. With *City of the Seventh Star*, he wanted to challenge the traditional conception of a linear puzzle chain, attempting to add a greater sense of freedom for the player.

[David] The jokey ideas came first and everything else fitted around them. That's just who I am – I wouldn't dream of doing something like this without trying to make it funny, so this wasn't a deliberate decision, it was a given. I wasn't particularly conscious of this being a shortcoming of other games. What I did notice about the early BBC adventures, though, was that the descriptions were short and unexciting, and I felt I had to improve on that. I also disliked the way some games presented a challenging puzzle early on that prevented you from making progress until you solved it – I wanted my adventurers to have the opportunity to explore lots of the game before they could become stuck. So those attributes were a conscious decision.

Expanding on location descriptions proved difficult. Writing for the most part in inefficient BASIC, David soon ran out of memory and struggled to implement his ambitious plans. He started adding several fast machine code and assembler routines that would speed-up the retrieval of text and invented his own text compression algorithm in order to add the verbose narrative that he sought.

[David] I realised that 8 bits was far more than you needed for a single letter, so I developed a simple mechanism to squeeze more information in. I am a bit hazy now but I think it was just that each byte incorporated information about punctuation - which also meant I could insert a space after the letter if it had a punctuation mark after it. [...] Some specific values would trigger the game to print out a complete sentence, such as the description of a room in a maze that would be used several times. It was nothing especially fancy but it did the job. It required machine code to produce fast responses. I created a utility of sorts – a program that would apply my compression rules to all the descriptions and compress them, so that I could make updates with ease.

With the exception of Melbourne House's *The Hobbit*, It would still be a year or two before mainstream adventures started appearing that used more advanced parsing using pronouns and commands such as GET ALL. David's balance between the traditional and progressive [and the need to award as much memory as possible to the script] retained the standard verb/noun approach of his mainframe influences. He also resisted the temptation of following *The Hobbit's* lead by rejecting the need for illustrations or location graphics.

[David] I felt that the addition of graphics was a massive backwards step. Novels do not need pictures, you get something far richer from your own imagination. The BBC's graphics were primitive, and you end up impoverishing the game as a whole just to put in pictures that add nothing to the enjoyment. I played one of these – *The Hobbit* I think – and hated it. So, there was no way I was going to have pictures in my game. Also, I can't draw.

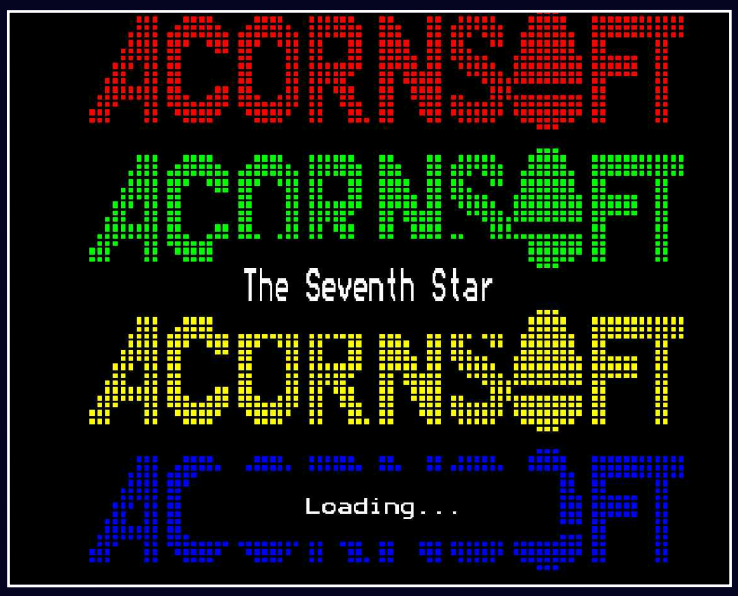
But there was one interesting design feature that did make it into the engine that wasn't seen in many games and wasn't even featured in *The Hobbit* – the inclusion of atmospheric sound effects. The player was treated to the bustle of a busy marketplace, a buzzing electrical chair, a ringing phone and the blowing of the wind through the trees in a wood.

[David] This seemed to me to be a simple and space-efficient of adding a bit more humour and interest, and some sounds worked quite well – as you are drowning, you hear your heartbeat gradually getting slower, for instance. And as nobody else was doing it, that was another plus.

Its refreshing that David wanted to write his own design rules and challenge the traditions and evolving norms surrounding how adventures should be perceived. In April 1984 he penned a letter to Alice, the adventure columnist in the Acorn-specific magazine Micro User, suggesting ways in which adventures should be evaluated by press and players. Alice had stipulated in previous reviews that the perceived time that it took for a game to SAVE could be a sign of a bad adventure, somehow linking the amount of data written to disk or cassette to the parent game's quality.

[David] I think it is ridiculous to judge an adventure game by the size of its save file. It tells you nothing about how much enjoyment you get from playing the game and in any case, you wanted to have as small a save file as possible to make it quick and easy to save and load from cassette. I felt that Alice had her own little axe to grind and her comments were a long way wide of the mark. [...] As far as design rules were concerned, for me it was all about making everything as compact as possible to leave the maximum room for vivid descriptions of the scenery and action. I was a self-taught programmer so I know I had lots of bad habits, but I feel I had a reasonably good intuition for what makes a game enjoyable.

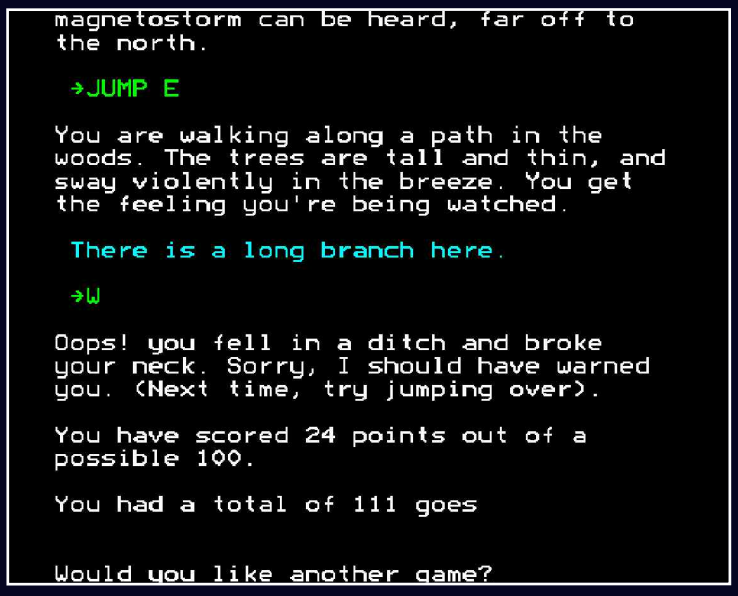
David spent the final six months of his time at university finishing



[Above] The colourful and universal loading screen that accompanied the majority of Acornsoft games on the BBC Micro and Acorn Electron.

[Above] The opening setting to *The Seventh Star* adventure. Make sure that you take all the items that can be salvaged from the wreckage of your craft, including the useful computer.

[Above] David Hampton used a range of British pop culture references throughout his adventure, including a nod to a famous Hollywood alien who needed to use the telephone.



[Above] Something that David would change if he ever revisited the construction of his adventure - removing the occasional instances where death would meet the player without warning.

---

the game, and sinking every waking moment outside of his studies into the development. Naturally, he approached Acornsoft to publish the adventure, since they were the premier publisher on the platform and the obvious first choice as a potential partner in getting the game to market. The iconic brand immediately signed up the game, insisting on one notable change in the process.

[David] I was over the moon; it was a really exciting time. My long-suffering girlfriend [now my long-suffering wife] got to appreciate how much it meant to me when went into shops and saw the game on sale - that was a really special feeling. [...] Acornsoft asked me to shorten [the name] to *The Seventh Star* as they felt this was punchier. Oddly, a couple of the reviewers referred to it as *The City of the Seventh Star*, I have no idea where they got the original name from!

Acornsoft's rigorous playtesting unit set to work evaluating and debugging the game in preparation for release. David was apprehensive as they started to find errors that he hadn't encountered when doing his own testing – aided by one of his university friends he was living with.

[David] I tested the game pretty thoroughly myself and was sharing a house with a friend who had a good go at it, and I felt the game was really well debugged. Acornsoft's testers were really thorough – they did some really unexpected things that I had not anticipated, such as trying to throw the robot.

A few niggles remained in the retail version, but these were mainly concerned with gameplay, rather than coding bugs - such as a lack of objects that are available to examine that appear, or are mentioned in the location descriptions, and the odd case of instant death and the ability not to be able to complete the game when certain actions

SPECTACULAR PROGRAMS
FOR THE BBC MICROCOMPUTER
FROM ACORNSOFT

Autumn 1984

Acornsoft Games 34

Gateway to Karos

Enter a fantasy world through the Gateway to Karos in this exciting adventure game. Your objective is to find the legendary Talisman of Khoronz, a powerful object used in the great battles between the powers of light and darkness. Your search takes you through forests and moors and out to sea. You will find many valuable treasures and useful objects which, with the Talisman, must be collected and taken back through the Gate. Finding your way back will not be easy, and you will encounter many hostile beasts – and also some people who may help or hinder you!

The Seventh Star

Travelling back to earth from your starbase, a violent magnetostorm forces you to crash-land on Seventh Star. The object of this adventure is to return safely to earth. Using the computer as your eyes and hands, you will explore the mysterious city on the planet, searching for the equipment that you need in order to survive long enough to find your way back to earth. A witty adventure with many interesting puzzles and sound effects.

Quondam

For experienced adventurers only, this is a journey through the mysterious world of Quondam. Set in the medieval ages, the magical land you explore is full of wonders, and also full of traps for the unwary. Your objective is to find and collect the treasures scattered through Quondam, and to deposit them in a safe place. Be cautious, though, for many things are not as they seem . . .

Acheton

Acheton is a disc-based game for advanced adventurers. Your journey begins in the deceptively peaceful setting of an English landscape but will lead you on to discover over 400 different locations. Your objective is to survive long enough to find a hidden cave, huge beyond imagination. Once inside, your task is to collect as much treasure as you can find, while exploring this strange and mysterious world.

Draughts and Reversi

Try your hand against the computer with these two popular games:

DRAUGHTS – the traditional game has been faithfully reproduced on screen for you to try your hand at playing the computer. The computer will make sure you follow the rules, insisting that you make all possible jumps or suffer the consequences – the computer has the option of 'huffing' the offending piece.

REVERSI – in this board game each player tries to surround his opponent's pieces, the objective being to gain as many pieces as possible by the end of the game. (Reversi is also known as Othello.)

In both games you challenge the computer, with a choice of difficulty levels, and at the higher levels the machine plays an accomplished game.

Both programs take advantage of all available memory, and so on the Model A the board is displayed in teletext mode, and on the Model B it is displayed in screen mode 1.

[Above] *The Seventh Star* appeared amongst many illustrious adventure stablemates in this Acornsoft brochure from late 1984.

---

are taken in the wrong order.

[David] I regret the three places where you die without warning. A reasonable fix would be to have a reminder at these dangerous locations that as prudent adventurer saves regularly. My other main regret is that I included NE, SW, SE and SW, which means that someone who is stuck has to try so many directions to make sure they have not missed any.

Still, the neatness and concise way in which *Seventh Star* deals with objects does help the player. Every object has a purpose and expanding this to include the ability to examine every single object referenced in the prose would have placed an even greater burden on the ever dwindling availability of RAM. The aforementioned "bug" where the player performs the wrong actions and is unable to complete the game is another regret of David's, and something that should have been signposted better in the final release.

[David] Regarding the ability to make a mistake which means you cannot ever complete the game – yes, I take the point that this can be very frustrating. I think these are probably part and parcel of adventure games but if I were to redesign *The Seventh Star*, I'd have fewer of them.

The game begins with the player stranded on the surface of the planet known as The Seventh Star, just on the outskirts of a mysterious city. The quest is set, to explore the city and survive for long enough to get back to Earth. From the wreckage of the ship, you find the first objects to help on the mission – a bucket, a crowbar and Mu-Mu, a hand-held micro-micro computer, named after the Greek letter that is used to represent 10-6 or "micro" ... for example in µm for micrometre.

David figured that home computers at the time were "micro", so the next logical step for a small machine was a micro micro. As you leave the remains of your spacecraft behind and attempt to enter the city, a sentry demands your name and you have to type it in [though a static citizen number is assigned to you in the game as part of a puzzle]. It adds a little immersion into the game, giving a greater feeling of individuality when you play.

[David] I wanted to surprise people when they see their names being used out of the blue – I didn't know about the concept of Easter Eggs until many years later however. The citizen code and card are a way to give the player an early success and sense of achievement to help motivate them to explore further.

The Mu-Mu also provided cryptic clues when called upon within the game, but David provided a comprehensive player-aid system, granting the ability to type *HELP whenever you became stuck. The code would then direct you to look up a numbered hint, in the game's accompanying documentation, that was specific to the room you were located in. For David, having access to an instant help system wasn't going to spoil the challenge of the game.

[David] I felt that there needed to be away of getting unstuck and so it was important to provide a straightforward way to get you past a mental block. I could never have solved *Colossal Cave* alone – my

friend Rick, who pulled an all-nighter on it with me, found more solutions than I did - some things don't change, he's better at cryptic crosswords too. There's no pleasure in being completely stuck and if someone wants to use *HELP liberally, well, that's a decision they are free to make.

The Seventh Star was released in January 1985 to favourable reviews. Andy Mitchell, in the leading adventure magazine The Micro Adventurer praised David's injection of humour for being one of the ingredients that "lift[ed] it out of the ordinary." He said that the "game has the spark of fun, combined with relatively easy puzzles which appealed to me and kept me battling on towards that final puzzle." He finished the article by congratulating *The Seventh Star's* addition to the Cambridge publisher's catalogue, noting that it would "do nothing but enlarge Acornsoft's prestige and keep their BBC adventures at the forefront of the software market."

In some quarters the writing and puzzle style of Hampton was compared to the great Peter Killworth himself. Of course, the usual critics, as mentioned elsewhere, did remind readers that players had perhaps had enough of the well-trodden science-fiction scenario and clichéd narrative.

[David] I was really pleased by the reviews, except Alice's – but as I mentioned she looked at things in a rather strange way. I don't remember other people saying it was clichéd, though it is a fair criticism. [...] It was unfortunate that *Countdown to Doom,* in particular, came out before my game did; I should have played it and reworked the theme of my own game somehow – but I was too focused on my own efforts at the time. I was pleased that people liked the humour and sound effects and that my game could stand comparison to those produced by people with a lot more knowledge, skill and experience than me.

Still, it was widely reviewed, and David's name appeared in many of the mainstream publications at the time, and the game received praise from all corners including such luminaries as the late Keith Campbell. Campbell, writing in his 1985 Book of Adventure noted the quick response of David's custom adventure engine ["none of those slow responses which we've come to associate with Acornsoft adventures] and commented that "in short, this is a game I thoroughly enjoyed." Mike Gerrard was equally praising, and in Personal Computer News in January 1985 he wrote "*The Seventh Star* is the latest is a fine series of adventures from Acornsoft, well-written and with not a picture to be seen. But once you get stuck in, you don't really miss it."

[David] I loved [receiving good reviews] ... this was all a completely new experience for me and I was somewhat surprised the game was even published – it had just been a hobby really. I have kept all those old magazines.

Acornsoft seemed to shy away from publishing disk versions of its games, and *The Seventh Star* received a standard cassette release. It seems strange that David was using a disk-based BBC to master the adventure and supplied those masters to the publisher, that in the end the medium was ignored for publication. In two of Acornsoft's other releases, *Acheton* and *Quondam* for example, the disk was put to good use for additional content, in the same way that later releases from the Austin Brothers and Level 9 used the additional storage for graphics and other enhancements.

[David] I did prepare a disk version but it was not something that Acornsoft had the time or inclination to release. Having done all that work to get to the point where I had successfully squeezed an immersive game into 32K, I think the prospect of making good use of the 100K or so available on a disk would have been a bit daunting. People would expect more text, more locations, more puzzles, perhaps even – horror! – graphics. So, I would not even have started down that road.

Sales based upon revenue on the face of it seem disappointing, and David received an advance of £750, that he put to good use towards the purchase of his very first home. The game didn't sell well enough to supplement the advance with any further royalty payments.

Worse was around the corner when Acorn Computers hit the financial buffers just after the game's release in 1985. Their new owner, Olivetti decided to disband the software arm of the company and sold its existing Acornsoft contracts and the majority of its back catalogue to Richard Hanson's Superior Software. Superior having released half a dozen text adventures of varying quality in the past had no interest in continuing with the genre, and looked to offload those assets elsewhere. Peter Killworth and his fellow Cambridge professors were able to negotiate the acquisition of their titles, which led to the formation of their own software house, Topologika Limited. For *The Seventh Star*, the end was nigh, and Hampton remembers that the rights to his own game were never offered up for purchase [Richard Hanson confirmed in late 2019 that Superior Software do not own the rights either].

With its trademark Acornsoft packaging and enigmatic painted science-fiction cover, It remains one of the rarest and most sought-after BBC Micro adventure titles, with prices for the game reaching triple figures when it appears on internet auction sites.

[David] The game came out towards the end of Acornsoft's period of pre-eminence. Shortly before its release, they lost their contract with WHSmith, which was a big blow, so the distribution was falling apart and it just came to market too late – that's my excuse anyway. The game did not sell enough for me to earn any money beyond my original £750 royalty cheque, however, there was once an advert from Acornsoft on the back cover of a magazine that listed their top sellers, and *The Seventh Star* was number 2 [behind *Elite*] so it did well for a short time at least. I had two copies for myself; one has been made into a framed picture for my older son, which puts me in a difficult position because it's not fair on my younger son and I can't bear to destroy the second box. I shall keep my eyes on eBay for another copy. It will be like the old J. R. Hartley advert - so there's another cliché for you!

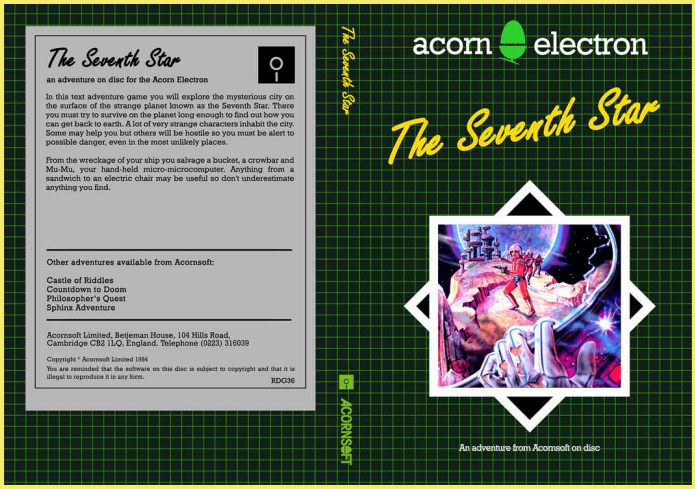
David remains proud of his creation, especially of the characters that he added to the game, such as familiar small brown alien making a telephone call home and pulling on several Hitchhiker's Guide to the Galaxy influences with his own paranoid, or in this case, depressed android.

[David] I got a lot of fun adding in comments that he would make about your performance along the way, and the player has to endure his company for quite some time before getting to the two places where he is needed. My least favourite would be requiring the player to realise they can go DOWN from the central market-place – that has the potential to frustrate. Also, there are three mazes, which is two too many.

At the end of development, Acornsoft paid David another advance to develop an Acorn Electron version of the game. Perhaps they had the port in mind by restricting *The Seventh Star* to a cassette-only release. It was a quick-port for Hampton, finished in June 1985, leaving the storyline, locations, puzzles and text untouched for the BBC Micro's less powerful sibling. To get the game to fit into its more meagre memory, a new method of text compression had to be devised.

[David] It was all about better text compression. I used all available space for the BBC Micro but after the game was published, Acornsoft asked me if I could do an Electron version for cassette tape. That was a big challenge, as I needed to save another 7k. So, I re-wrote the text compression from scratch. The 256 possible values of a byte were mapped to a list of words and phrases such as "walking", "shaft leading down", "North" etc; some values represented combinations of two letters such as "na", "me" "ti" and so on – that meant that many word could be represented efficiently, e.g "name" = "na" + "me" and "time" = "ti" + "me". I also figured out that I did not need 5 bytes for the letters of the alphabet, because some are used far more than others – so I had a little decision tree that would say, "if the first bit is a 1 then it's a vowel and if it is a 0 it is consonant... then look at the second bit" and so on – that meant that the most commonly-used letters could be expressed in only 3 or 4 bits, though uncommon letters needed more. I am a bit hazy on the details! Anyway, it was a combination of these two ideas that enabled me to squeeze out an extra 7k while keeping the original descriptions. I had a machine

ELECTRON ENDOWMENT



In response to the Stardot community's interest in *The Seventh Star*, David Hampton sent his Acorn Electron master disk to David Hitchins who has successfully created a working version of the game. It's hoped through the help of the Acorn community and Dave Moore's Retro Software label a brand-new remastered physical release may soon be available.

code program that would recompress the text every time I edited it, and of course the decompresser that was used when the game was being played was also in machine code – even so, the speed at which the text appeared was only just acceptable, in my opinion. So – yes, that could have been used to provide more text for the BBC micro, had there been the time, inclination and financial incentive for this.

The Electron version was written on David's trusty BBC and tested by a colleague who owned the target machine. It was complete, and in a form ready to be mastered and distributed. The parser ran a little slower than its relation, due to the mismatch in technical details of the machines, but also in the way that the text compression algorithms had to work harder on the less powerful machine.

Now, almost 35 years later the game is still attracting new players and inquisitive minds in the Acorn community. In August 2016, Stardot community member lurkio disassembled the code to *The Seventh Star*, bypassing David's copy protection routines and posting the analysis for all to see.

[David] I am absolutely blown away by it. The people on Stardot seem to be like computer archaeologists, getting a kick out of unearthing the hidden secrets from these old games. All those memories of the fun of creating the game have come flooding back; I have dug out the original hand-drawn map and printouts of the program; I have taken over the dining room table with all the kit and my wife is moaning about all of the time it is taking. So the full 1983 experience is being recreated and I am so grateful for people's interest.

The community have fixed one bug in the game's program, plan a disk release, and have added a fully documented solution and scrolling introduction to the game in order to deliver the definitive version. With the game's engine innards exposed for all to see, one keen observer noted how close its workings were to the other games from Acornsoft, notably *Castle of Riddles*.

[David] To be honest I'm not sure how many different ways there are to code an adventure – I just did what felt instinctively right. You parse the verb and noun, identify special things that are going to happen as a result, update player location and the location of items as appropriate, then repeat. I don't think there is much to it really, except the text compression challenge, which I wrote myself - if I had used someone else's method it would have been better I am sure. In a sense, writing the game was a puzzle to be solved and I wanted to have the satisfaction of solving that puzzle on my own, so I wasn't particularly interested in finding out how others had done it.

It's testament to David's coding abilities, and an incredible achievement for his first game that despite developing his engine in relative isolation, the result was so close to his peer's academically-led efforts. *The Seventh Star* displayed rich writing, a wide-enough vocabulary and a parser that was quick to respond – it was just a pity that it was always compared to *Countdown to Doom*.

[David] I think the biggest thing would be to [...] somehow to fix the similarity to that scenario. Perhaps the player could have wandered off the official path of a Studio Tour in Hollywood, and finds himself locked in after the place has been closed for the winter, with no apparent means of escape... on exploring, he/she discovers that life in the studio continues even when everyone that works there has gone home. But then I would probably be told it's too similar to Toy Story!

Star remains his first and only game. By the time he'd completed the Electron version his personal situation had changed beyond recognition from his student days. He had a stable home life, full-time employment and a burgeoning relationship [that would blossom into marriage and children]. There was no desire to continue ploughing so much time into what was just a hobby – David wanted a better balance in his life.

[David] Bear in mind, too, that Acornsoft had shut up shop, and that I was a lone, self-taught programmer. The technology had moved on apace and producing a competitive game soon required a team of people – I just did not have the resources or skill or contacts to even consider the possibility. It was always going to be a one-off. I feel hugely fortunate to have had the opportunity to write a game during a very narrow period of time during which you could buy a computer that was able to run something worth playing, but not yet so powerful that you needed a whole team of people to take full advantage of its capabilities.

Decades on, his adventure still captivates his children, and with David's interest reignited after the Stardot community postings, he's embraced his creation and has repaired his beloved BBC Micro in order to retrace his steps upon that distant Seventh Star with his family.

[David] To be honest I had not looked at the game, [or] other adventures, since [1985]. The recent flurry of activity on Stardot was prompted by a conversation with someone who attended a training course I was running – she was telling me about John Robertson's comedy show. I was going to send her a link to some information on the game, so I googled it and found [a post called] 'Today I Hacked *The Seventh Star*' which was completely unexpected! Then I discovered that you can play it online, and you don't even need to install an emulator any more. *The Seventh Star* was pretty much a forgotten thing from my past and now I discover that people are still playing these games – I am really grateful for their interest and enthusiasm for the genre.

DESERT ISLAND DUNGEONS

HMS Seventh Star captain **David Hampton** run aground on a deserted island. He salvages his micro-micro computer on which to emulate five text adventures of his choice whilst figuring out how to escape.

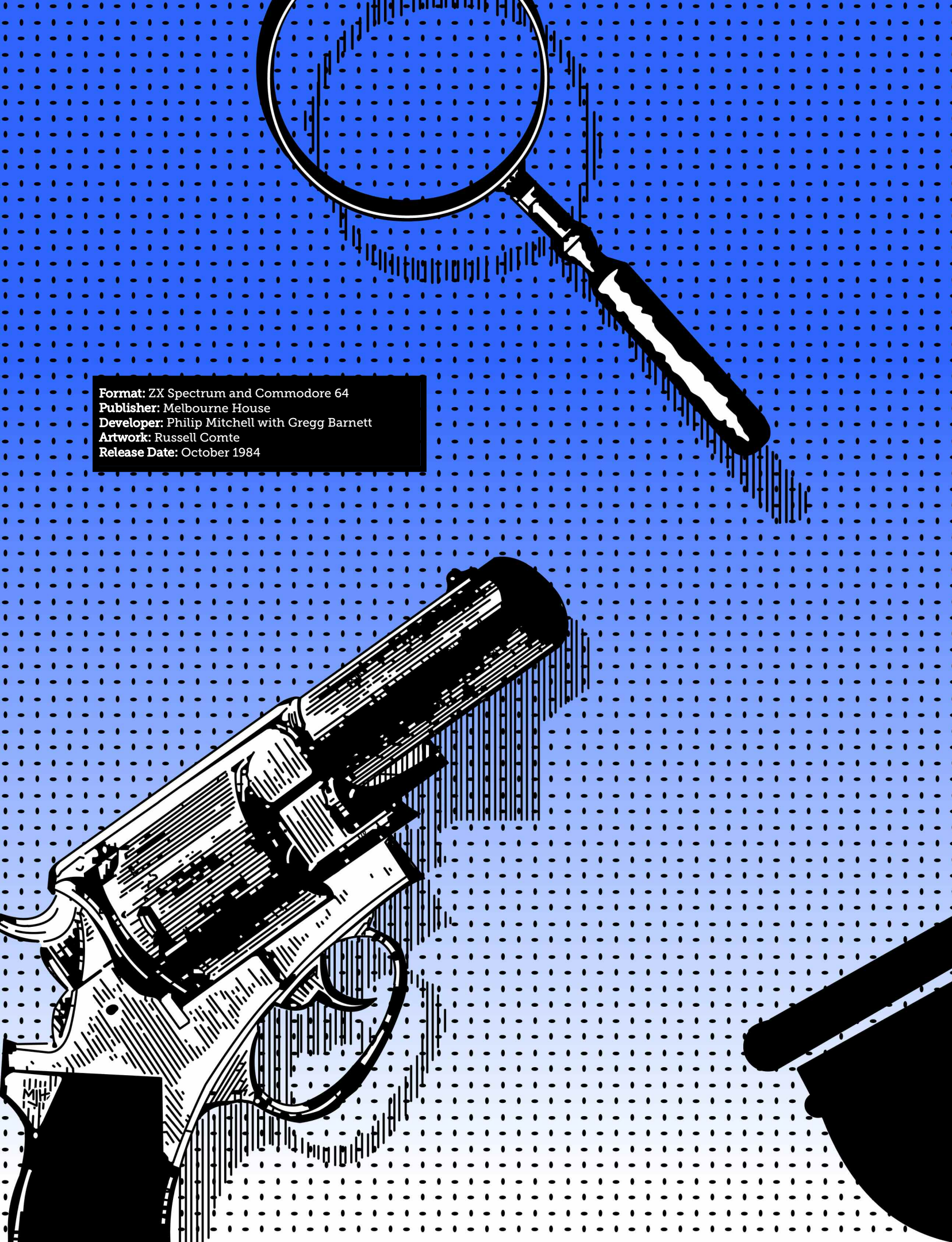
Ok crikey, I don't remember really!

For the full *Colossal Cave* experience I would want to return to the mainframe in the university Engineering Department.

I'd want to play Level 9's *Snowball* again on a BBC micro.

And then all Peter Killworth's adventures, on the Beeb again, to see what his style was like.

I know this is not really a proper answer to your question, but I'd love to re-live the day in around 1975 or 1976 when I first walked into NESCOT and a bearded guy called Julian showed me a room with a massive computer in it. He sat me down at a teletype and pressed something, and it printed the words "Hello, I'm MP3, who are you?". From that day, I was hooked.



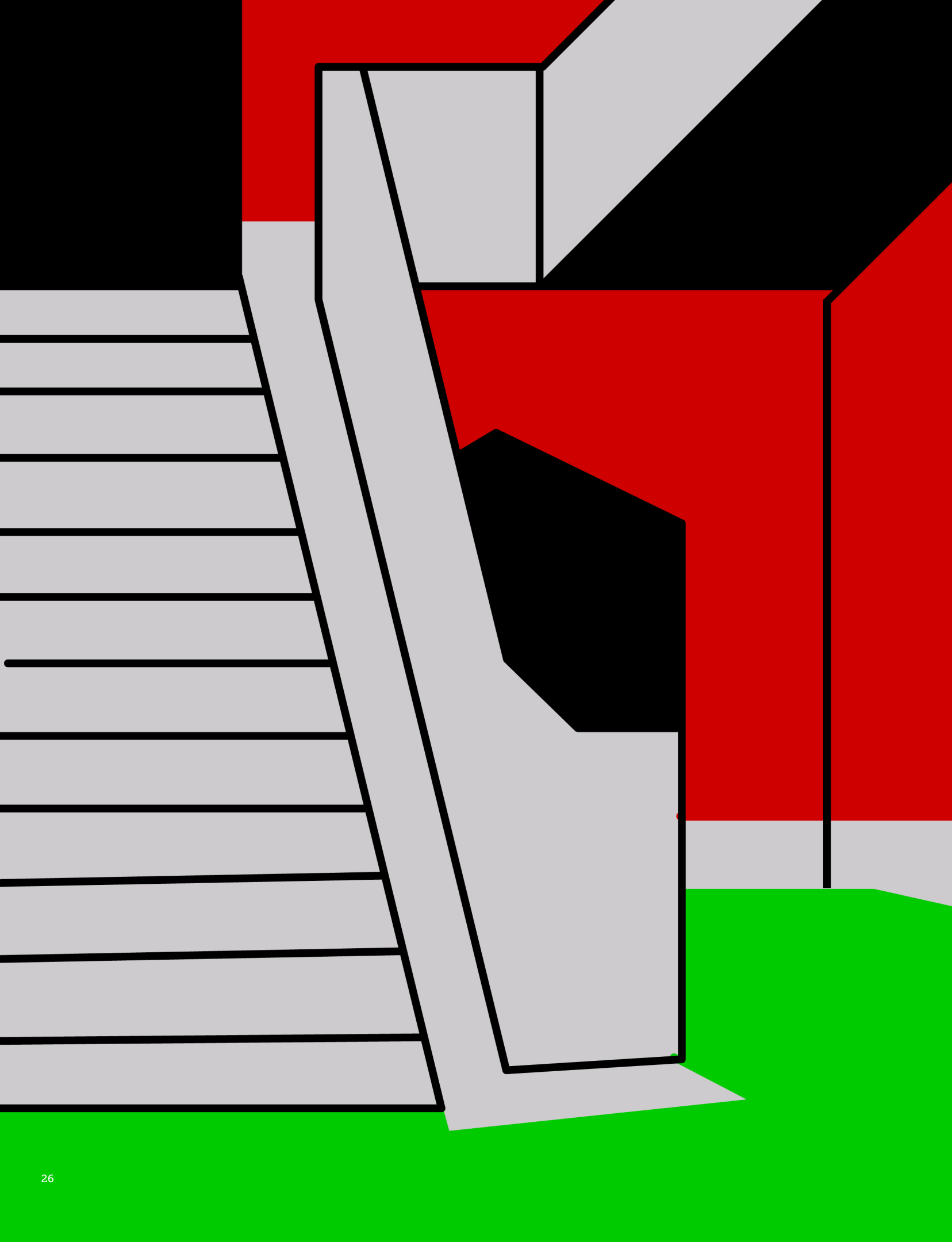
Format: ZX Spectrum and Commodore 64
Publisher: Melbourne House
Developer: Philip Mitchell with Gregg Barnett
Artwork: Russell Comte
Release Date: October 1984

SHERLOCK

In 1984 a revolution in adventuring starred fiction's most famous detective - Sir Arthur Canon Doyle's extraordinary sleuth, Sherlock Holmes. With the sun shining through the windows onto our faces, **The Classic Adventurer** travels the streets of Victorian London in a hansom cab and explores the world of Melbourne House's *Sherlock*.

Fledgling Australian videogame development studio Beam Software was founded to supply the London-based publishing company Melbourne House with software. House had entered the market in 1977, and with co-founder Alfred Milgrom's background in computing they quickly ventured into publishing a range of books for emerging home micros, including the Sinclair ZX80 and ZX81 computers.

To Milgrom's surprise, the books were a runaway success, so he convinced their author William Tang to start writing games for retail, realising that he could sell the same work twice, one on cassette and the other in print. Tang became Beam's first employee and Melbourne House began to build a reputation for quality software with their first releases, including a flicker-free version of *Space Invaders* for the Sinclair machines.



In his formative years, Milgrom had spent some of his time at University playing the original *Adventure*, and through his continuing interest in computing, played the home derivatives created by Scott Adams for his Adventure International label. Milgrom wanted to develop his own adventures, but had the ambition of producing something more advanced telling Home Computing Weekly that "[I] didn't think that the current adventure games were exciting. [So] we set ourselves a more interesting challenge."

That challenge focused on expanding the basic two-word command parsing and simple logical puzzles of Adams' games, and Alfred's ambition grew to developing "the best adventure game ever." The era of bedroom coders had yet to begin, so Milgrom returned to recruit programming talent from the computer science labs at his former university. Along with Philip Mitchell, a female student, Veronika Meglar was offered work on the new adventure project.

Meglar told website L'avventura è l'avventura "Philip [...] and I [...] studied Computer Science, Physics and Mathematics together at Melbourne University, and worked on many group projects together there. I responded to an advertisement at the university looking for a programmer to design and write games part-time."

As part-time employees Meglar and Mitchell delivered the hugely successful and groundbreaking *The Hobbit*, released in April 1983. With sales estimated in some quarters at up to a million units [Milgrom in an interview with Helen Stuckey in 2013 estimates a million sales], it had a seismic impact on adventure authors, the gaming press and consumers – and everyone was hugely excited for what the team would develop next. For everyone concerned, including the developers themselves, it was obvious that the next game in the continuation of the series should be a translation of the next book in the Tolkien Hobbit story; The Fellowship of the Ring. Alfred Milgrom approach the Tolkien Estate looking to secure a deal, only to find that Fantasy Films owned the rights to the three books in the The Lord of the Rings series. Fantasy Films, along with the Saul Zaentz Production Company had secured the media rights in 1976 from United Artists. They produced Ralph Bakshi's feature-length animation The Lord of the Rings, released in November 1978, but it was a critical and commercial failure at the time and fell from public consciousness. Unfortunately for Milgrom and Melbourne House, by 1982 the rights had expired, so he looked around for a property with the same public fondness and brand knowledge that Tolkien commanded and settled on Arthur Conan Doyle's English detective, Sherlock Holmes. The Holmes catalogue - mostly published before the public domain copyright threshold of 1923 [a watershed date pinpointed by copyright lawyer Melville Nimmer in the US] - made it freely available to use and perfect fodder for the genre [though arguments about the Holmes copyright rumbles on to this day].

Discussions around the new Holmes games started before *The Hobbit* was completed. The Beam development team bounced around ideas about how the various Holmes storylines could be adapted, and how their evolving adventure engine could be manipulated to suit the demands of an interactive murder mystery. Before any code was written, Veronika Meglar, the genius behind the unique character interactions of the first game decided that her future lay elsewhere, outside of games, and left the team.

Veronika recalls that "at the end of 1981, I finished my Bachelor's degree. We were beginning to discuss using the Sherlock Holmes mysteries as a next games project; I was not sure that the adventure game engine I'd developed was a good fit for the Sherlock style of puzzle solving, although there were definitely aspects that would translate across."

[Meglar] I did not believe the NPC engine I'd written in Assembler was powerful enough for the kind of interaction that *Sherlock* would

WHAT'S IN A NAME?

Beam Software was named after its co-founders, taking the BE from Naomi Besen and the AM from Alfred Milgrom.

need. The story of Sherlock Holmes is centrered around interpersonal interactions; it seems to me a fundamentally different genre from adventures. The player interactions of *The Hobbit* were under-designed for complex interpersonal problem solving – there were too many special cases and special coding. I felt it required a rethink and redesign with those capabilities in mind, and that it would require a higher-level language such as C to make it feasible.

It was a disappointing turn of events, as Veronika had worked closely with Philip, the two discussing the possibilities of plot and storyline. Mitchell crediting the dawning *Sherlock* narrative to "a lady by the name of Veronica [sic]" in an interview with Keith Campbell in the Computer and Videogame magazine Book of Adventure in 1985. It seemed for developers, even in the early days of the industry, burnout played a part, and Meglar conceded that she was "ready to start something new after a year of coding and debugging in Assembler."

The exact date that Veronika left is up for debate. It's clear from her own memories, and the recollections of Mitchell himself who told C&VG's Keith Campbell that "I was the only programmer on the [The Hobbit] during the last six months", that she certainly did depart before the game was completed. However, in April 1983, in an extensive Melbourne House feature in Home Computing Weekly, Meglar was photographed for the article and listed as still employed as a "software consultant". What is fact however, is that after Meglar's departure Mitchell was left holding the baby and he struggled before *The Hobbit's* release to grasp the engine's code, finish the game, and crowbar in the required location graphics.

Mitchell had joined Beam Software as a full-time employee after graduating from university in 1981. He was in sole control of the *Sherlock* project, and found himself thrust into the limelight as the press clamoured to find out what was next from Melbourne House. He told Popular Computing Weekly in October 1983 that "*The Hobbit* was a starting point. What I want to do is take it a stage father. One of the attractions of Sherlock Holmes is the very involved personalities of the characters, particularly Inspector Lestrade and Dr Watson." The Digital Antiquarian, Jimmy Maher commented that "the core idea [behind *Sherlock*] remained the same: to create a living world populated by other characters going about their business and pursuing their own goals, full of dynamism and emergent elements."

For development, Mitchell turned to his trusty TRS-80 micro, powered by an off-the-shelf assembler package that compiled and compressed his interpreter and database code and squirted it across to the ZX Spectrum computer. Mitchell's coding prowess was such, and his compression algorithms so effective that by the end of development he estimated that the *Sherlock* data was a whopping 500K in size - over ten times greater than the Spectrum's capacity.

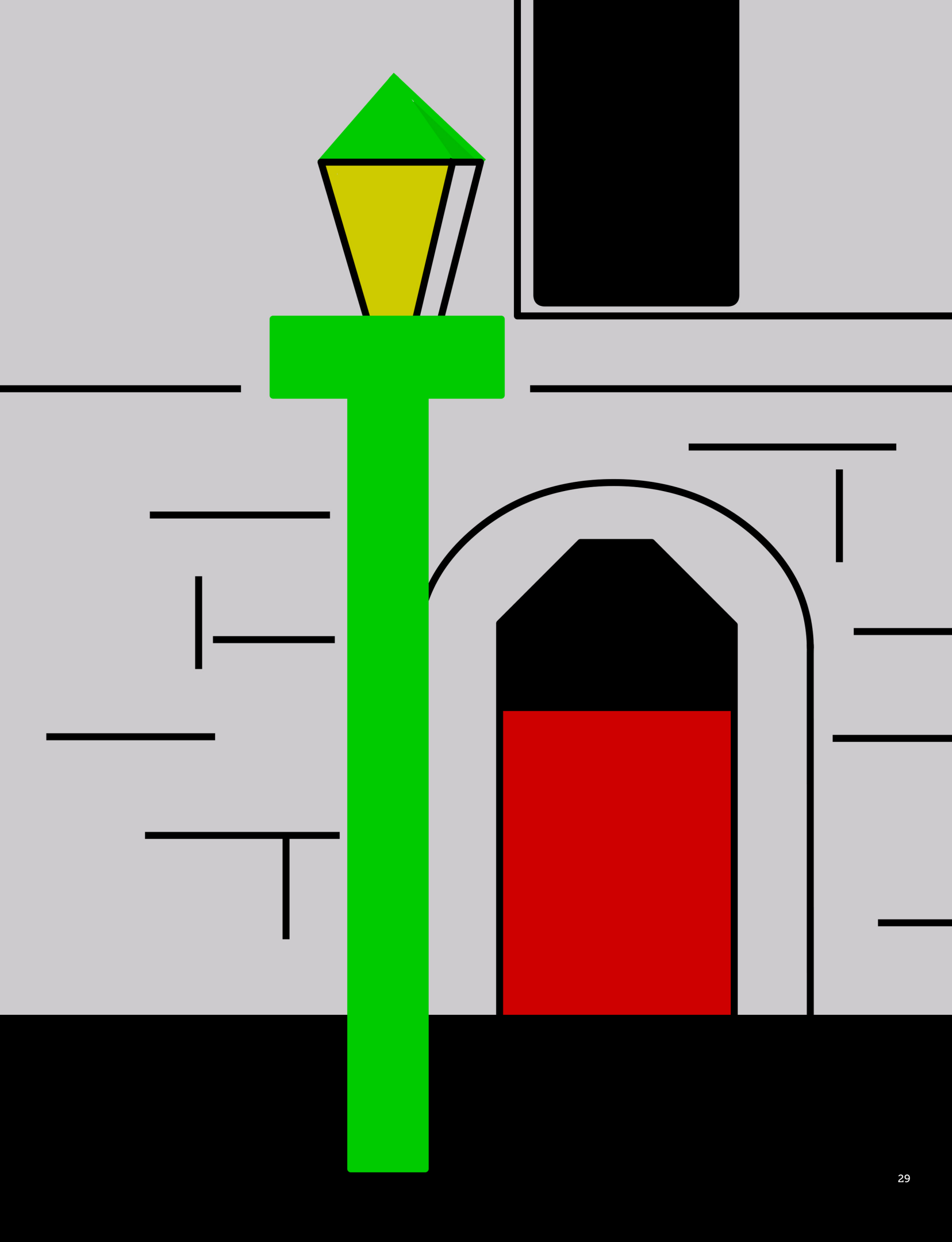
Sherlock was established as a classic murder-mystery thriller, not based on a single Conan Doyle story, but an amalgamation of themes from various sources - though one scenario and puzzle is taken from The Man with the Twisted Lip. Melbourne House's marketing blurb pronounced that "In *Sherlock*, you take on the role of Holmes. With the ever-faithful Watson you will roam freely through the gaslit streets of London and explore Victorian England in your quest to disentangle seemingly inexplicable and unresolvable mysteries. You will follow up clues and confront sinister and elusive characters during your adventure." We find out about those sinister forces after starting the game in Holmes' sun soaked sitting room. His trusty companion Watson is sat reading The Daily Chronicle and finds a most devilish case where two close friends, a Mrs Brown and a Mrs Jones have been murdered with the same weapon in a small township called Leatherhead. Inspector Giles Lestrade has already been dispatched by Scotland Yard to investigate, and so it's up to Holmes and Watson to join him [via a train departing from Kings Cross] and solve the case. As the adventure unfolds we are introduced to a cast of characters including belligerent Hansom Cab drivers, Chief Constable Straker [a namesake taken from The Adventure of Silver Blaze], the Phipps, Strachans and their various servants who all embellish the storyline. As Holmes follows Lestrade around the murder scenes, the game is really afoot as a note is found with the body, signed PF, implicating another character, one Major Percival Ffoulkes [a name used by Barry Grant in his revived 21st Century Sherlock Holmes novels] – and it's up to the player to test their powers of deduction and either prove the Major innocent or otherwise as the case and clues unravel.

Meglar and Mitchell's design ideology, echoed by Maher in his musings about *Sherlock*, had been shaped by the desire to overcome two limitations they'd identified whilst playing the original *Colossal Cave* adventure. Meglar's academic paper on *The Hobbit* - There and Back Again: A Case History of Writing *The Hobbit*, documented the exasperation she felt when encountering *Adventure's* less-than-intelligent beings, and frustration at trying to find the correct words in its restricted vocabulary. She noted that "each Non-Player Character [NPC] was tied to a single location, and always did the same thing. Lastly, you had to figure out exactly the [verb/noun] incantation the game expected; if the game expected KILL TROLL, then any other command – ATTACK THE TROLL, for example – would get an error message."

They loosened the command straitjacket by creating perhaps the most advanced parser seen in an 8-bit adventure to date, dubbing the new technology *Inglish*. The *Sherlock* manual explained that "A subset of English, *Inglish*, first seen in *The Hobbit*, is used in this game to communicate with the program, which uses a large vocabulary of 800 words. Each sentence must have a verb and there are a few simple, and mostly obvious, rules governing the use of adverbs and adjectives. Several actions or sentences can be linked in a manner which allows many different permutations."

The advent of *Inglish* is credited to Stuart Ritchie, a fellow University student and linguistics expert – employed around the same time as Meglar and Mitchell - though Meglar was later to dispute his impact on the project. In There and Back Again, she said "according to what Phil told me at the time, none of [Stuart's] design was used – although I suspect that being exposed to his thinking helped Phil crystallise what eventually became *Inglish*."

The *Inglish* parser used in Sherlock is identical to the one in *The Hobbit*, and although the blurb suggested the recognition of 800 words, the accompanying manual [as it did in *The Hobbit*] recommended around 50, spread across directions, actions, adverbs and prepositions. The functionality complex *Inglish* entered by the



player was simplified by Mitchell's code to a more straight-forward verb/noun combination. As Meglar explained, "No matter what the user entered – TAKE THE SHARP SWORD AND EXCITEDLY HACK AT THE EVIL TROLL, say, he'd convert it to a simple action/target pair to hand off to me: KILL TROLL, or perhaps, KILL TROLL WITH SWORD. Compound sentences would become a sequence of actions, so TAKE THE HAMMER AND HIT GANDALF WITH IT would come to me as two actions: PICK UP HAMMER, followed by a next turn of HIT GANDALF WITH HAMMER."

For all the effort it took to create and code the parser, Mitchell recalled in the Book of Adventure bundled with Issue 27 of C&VG that he found the implementation of *Inglish* within *The Hobbit* unnecessary, but still persisted in expending the energy to enhance it for *Sherlock*. "I think [players] have become used to adventures where you can only use one-word commands" he recalled, "[and] only tend to play *The Hobbit* in the same way. This is a pity because the program can cope with quite long sentences."

The game blurb suggesting that Sherlock will "roam freely through the gaslit streets of London and explore Victorian England" wasn't specifically true. Yes, you will follow-up clues and meet sinister and elusive characters, but there's a single mystery, and the ability to "freely" roam London is greatly exaggerated. As the game's plot widens, the amount of locations that can be visited does increase [often by asking suspects ABOUT their ADDRESS], but the initial destinations are completely railed and linear, controlled by the opening sequence pointing towards the murder in Leatherhead. You can barely go anywhere else but to the train station and off to the countryside. By the way, the ability to move on trains is free, unlike cabs [fares payable in unfathomable old English pounds and shillings], and you can usually end up lost just by standing on the wrong platform and getting on the wrong train. The manual does explain the player's inability to move from Baker Street, "In his adventures, Sherlock Holmes never walked where he could take a hansom cab or catch a train ..... This is in keeping with the character of Sherlock Holmes (and because of the memory limitations of your computer)."

Also, the freedom to walk around, was highly restricted by events that would occur in the game via the real-time clock which ticked onward as the player inputs commands. I'm in two minds about this kind of adventure. Taken at face value, it should play out as in real-life, events happens in the background and at other locations when you aren't there, but it's so difficult in a computer game [especially an adventure with varying player skills] to implement well. For instance, if you don't immediately go to Kings Cross then you don't get to meet Lestrade and you have no chance of completing the game – you'll miss interrogating suspects, discovering their alibi's and fail to acquire the ability to access locations, due to never been introduced to Constable Straker and the various local policemen that guard specific places.

The beginner and uninitiated would certainly spend more time than available in the initial locations and may not get onto the right track [no pun intended]. To combat this Melbourne House included a "Where to Begin" guide with the game, reassuring novices that they "may require some help in entering the world of *Sherlock*." The guide

provided four scripts, each introducing a different challenge, from travelling around London in cabs, to walking through the early set of puzzles and holding conversations with different people.

One other anomaly mentioned in the manual, and one that allows the player to disregard the normal operation of time is that "as Sherlock Holmes you have an inexhaustible capacity to go without sleep once you are deeply involved in a case." This gives the player from the start of the game, on Monday morning in Holmes' Study in Baker Street until 01:02 on Tuesday, when Lestrade finds one of the characters guilty and arrests them – thus ending the game.

In 2012 both *The Hobbit* and *Sherlock* were disassembled by CH as part of his *Wilderlands* and *Foggy London* projects. By running CH's software the inner workings of both games were exposed, revealing the dictionary, the state of objects, and the movement and actions of the NPCs throughout the game map. This visual representation of the inner workings of the game highlight the effects of time [every action has a cost in minutes and seconds] and reveal the computer characters' superfluous need for sleep.

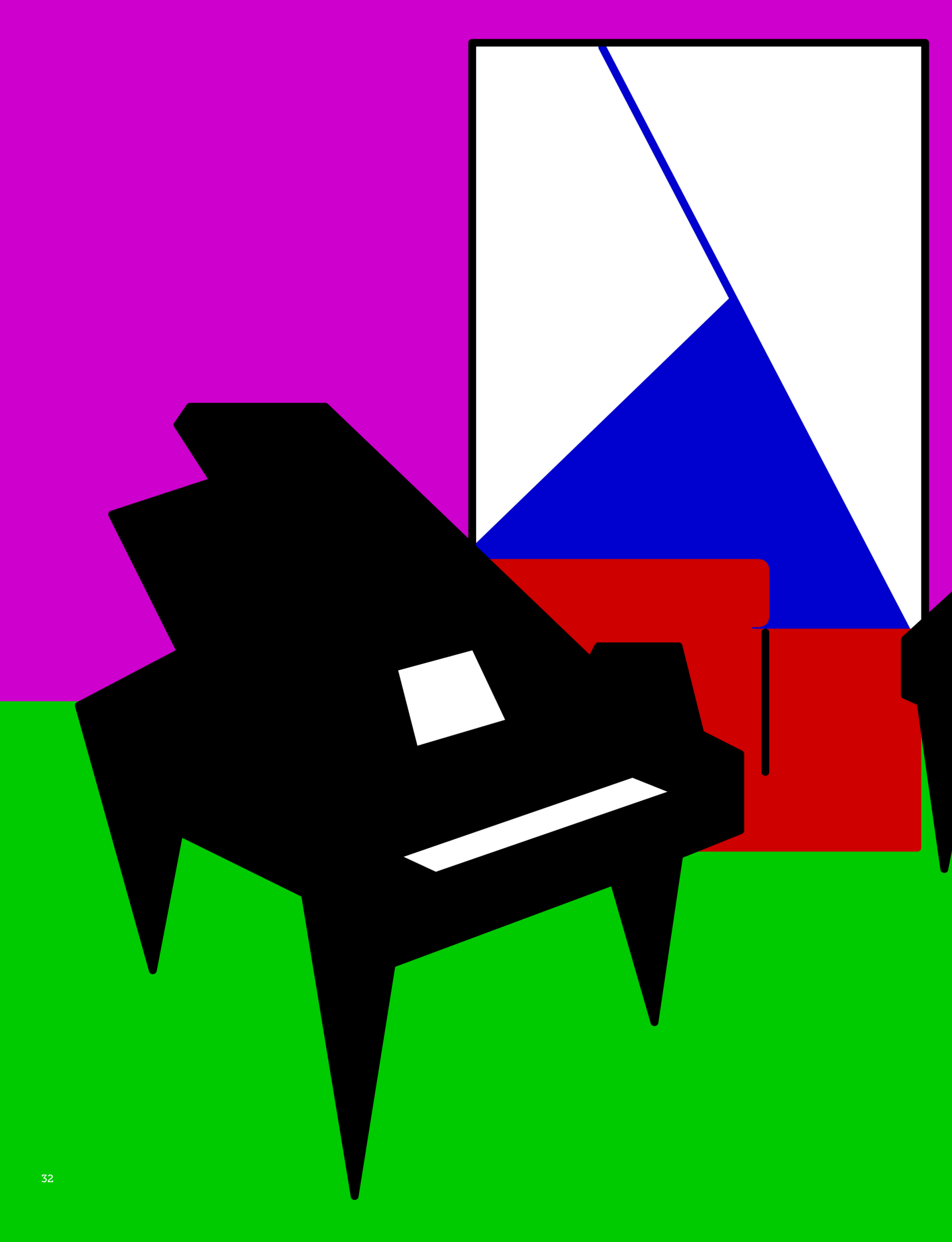
[CH] *Sherlock* is strongly based on the autonomous actions of a number of NPCs, roaming through London and the city of Leatherhead. [...] Therefore, this game is driven by a big time-table, which allows any NPC to do effectively everything the player might do. [...] A funny thing [...] though, they disappear from time to time - obviously resting – [so] you don't know about their doing. Daphne Strachan is the exception: shortly past 21:00 she goes to bed in her room in the first floor of Brown's House. You can visit her there,

SHOOT 'EM UP SIDESHOW

During the development of *Sherlock* and *The Hobbit* both Philip Mitchell and Veronika Meglar worked on many side projects to keep the Melbourne House cash register ringing.

The basic side-scrolling shoot-em-up *Penetrator*, released in 1981, was one of them.

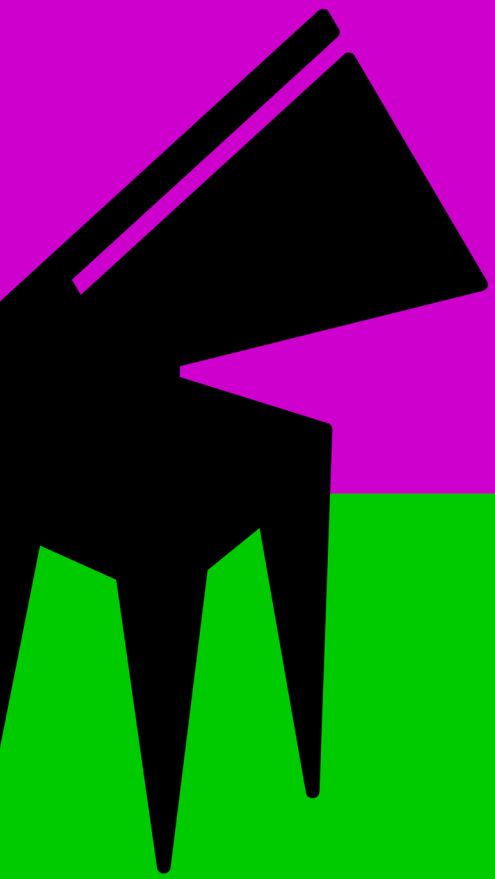
Meglar recalls, "Phil was the clear lead on that game while I worked on some pieces of it, and I think Kerryn may have worked on it a bit too. It was a copy of the arcade game *Scramble*."



finding a "sleeping Daphne Strachan", but can't do anything else ... after all, it's a family game.

Throughout the development it was unclear whether *Sherlock* would feature graphics. At the start, Milgrom cast doubt on their inclusion after his experiences with *The Hobbit*, telling Micro Adventurer in late 83, "we are not sure if we will manage the more advanced graphics together with the more complicated English interpretation and character interaction routines." Expanding the game's dictionary, enhancing the character interaction and including graphics on a ZX Spectrum was going to be a tall order, and for external observers a potentially unachievable ambition. Jimmy Maher, the Digital Antiquarian summed up the insurmountable challenge facing Mitchell, saying, "Throw in the need to have it all reside, with graphics, in 48K of memory, and it started to look impossible." By May of 1984 it seemed graphics were a luxury that *Sherlock* couldn't afford. Paula Byrne, Melbourne House's publicity manager, confirmed that the company had considered culling the number of graphics in the game. In the end, Milgrom made the decision to retain them. Post-release Mitchell debated with Keith Campbell in his Book of Adventure at length, suggesting that the game would have in hindsight been better without the extra memory given up to illustrations. In the end however, "the boss had insisted", he said, "so the boss got graphics".

The game's promotional material continued to over-egg its featured pudding. Adverts and accompanying literature stated that "most" locations had an illustration. The reality was far from the bravado. There are a paltry 14 locations that have images in *Sherlock*, including one that is duplicated for two platforms at Leatherhead Railway Station. They are sparse and half the resolution of *The Hobbit's*, using the traditional plot, line and fill routines that ensured the minimum of memory use possible. Given the detail of *The Hobbit* images, those in *Sherlock* are disappointing, especially considering that the artist responsible, Russell Comte partnered with Melbourne House and Mitchell on the wonderfully [and intricately] illustrated *Mugsy* adventure [a Gangster-strategy/management sim released in June 1984]. But, they are pleasant to look at and used cleverly to add to the atmosphere of the game, breaking up the text at opportune moments. Couple the graphics with the neat screen display, and *Sherlock's* presentation moved up a notch, with a nifty proportional font, a scrolling location window and the real-time clock that displayed the passage of time.



AnimTalk, "the amazing implementation that allows you to instruct other characters in the adventure", first implemented in *The Hobbit* is hugely expanded in *Sherlock*. The brief reads that "You will follow up clues and confront sinister and elusive characters during your adventure." To start, social interactions play an intrinsic part in this game, where pleasantries, or not-so pleasantries define the relationships that characters carry throughout the story. More often than not you'll have to greet other characters by SAYing "HELLO" before they'll even engage with you. For this, Milgrom told Micro Adventurer that *Sherlock* was "another step down the road where it will be difficult to distinguish between the computer and another player."

There's also Watson: He's *The Hobbit* equivalent of Gandalf or Thorin. And, just like Thorin, Watson has his own "sitting down and singing about gold" moments. John Fraser of Micro Adventurer magazine wrote "Watson can be difficult, and he delights in making sarcastic comments such as 'This is brilliant Holmes. I don't know how you do it'." Unfortunately, the novelty soon wore off, and without insisting on Watson to FOLLOW ME, things can take a turn for the worse. Fraser again comments, "The first time I played I got fed up with him and went off on my own. I returned last to see whether he had changed his mind but found a note saying he had gone to see his patients and wasn't available. Another time he stubbornly refused to join me at all." Alan Giles, the Software Project Co-ordinator of Melbourne House admitted in an interview in the September 1985 issue of C&VG magazine that Watson was "virtually useless".

It's interesting to read the accumulated press that *Sherlock* amassed

over time. As well as discussions about the inclusion of graphics, there was a constant battle, as the game evolved, about the exact capabilities of NPCs. Mitchell himself revealed the internal musings of a game designer in his Popular Computing Weekly interview in October 83 – "I haven't decided if [Lestrade] will remain in his office or wander about the game. If I let him roam free then, before you could ask him questions you would have to find him first." With the game widely expected to be released two months after this statement, and with Lestrade playing such a central role in the final retail game it's telling just how much the final game changed before release. Even then it's possible that the game was changed right up until the final master – TV Gamer magazine in December 1984 shows a location image for Daphne's Room that didn't appear in the final game. The NPCs were the one concept that contributed the most to the extended development time.

[CH] In *The Hobbit* the main routine consists of a user input, and after that all the NPCs have one turn [to perform an action], i.e. jumping to the next item in an, often very short, action list. When the player waits for some 30 seconds the game generates a WAIT input, giving some illusion of real time. In *Sherlock* everything is driven by the clock and the calendar. After each tick [i.e. a minute in time] the main loop cycles through a big table looking for actions to perform at this very moment. Player input is just another nuance as time elapses. All the NPCs go on their businesses, no matter what Sherlock does. They hail cabs, talk to each other, interrogate witnesses, ride trains, destroy evidence or move through their respective houses.

This complexity, in culmination with the freeform ruleset applied to all of the objects within the database, led to players experiencing emergent behaviours, where strange and unexpected responses and actions started to manifest themselves. They were christened *HobbitBugs* in the original game, and Mitchell worked to tighten up the more bizarre behaviours in a patched [a rarity for the early days of videogames] v1.2 release of *The Hobbit* commercial product. *Sherlock* contained its own unforeseen behaviours, cunningly nicknamed *SherlockBugs*, and even the game's instruction sheet acknowledged in part that unanticipated things could occur: "Due to the immense size and complexity of the game it is impossible to guarantee that it will ever be completely error free."

It's not difficult to understand why the bugs continued to plague the engine. A contributing factor must have been Mitchell's struggle with Meglar's logic, and the unwillingness to constrain her liberal AI rules which were required to bind together the character relationships the game relied on.

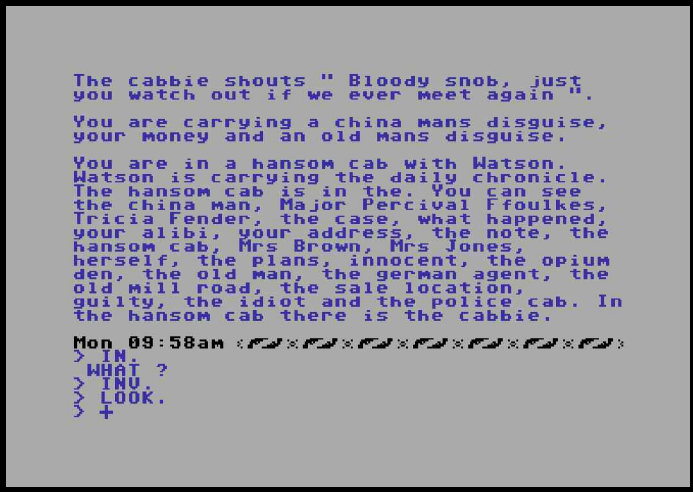
[CN] A new concept is also that of disguises. [...] It might seem a simple thing [but] this poses a lot of challenges for the game: some people might recognize you still, others might not. Though you are in the game's context a different person, you still have to carry your belongings and all the conversions have to shift pronouns and the subject. [...] Tightly connected to the impersonation of the various characters is the implementation of knowledge ("Ffoulkes is innocent", "Brown killed herself",...) which is essential for the course of the game. The programmer(s) had to find a way for this rather great abstraction.

Add into that the meagre 48K available on the ZX Spectrum which was being constantly shifted around like a huge sliding puzzle to accommodate the game logic, parser and graphics. Meglar recalls the sacrifices that had to be made in *The Hobbit:* "We were so limited by memory that we would adjust the size of the dictionary to fit the game into the desired memory size; so the number of synonyms available would sometimes decrease if a bug fix required more lines of code. It was a constant trade-off between game functionality and language richness." Such complexity in logic and code was destined for issues and exceptionally difficult to test and debug.

[CN] [*Sherlock*] was done in Z80 assembly language. Though C-like structures or pointers would have greatly facilitated implementation, those were not available. Worse if something went wrong in the code the game would certainly crash with no hints or traces of the exact problem.

SherlockBugs became a popular topic in the gaming press and players actively hunted for errors or acts of randomness. Tony Bridge, in PCW May 1985, used his column to comment on the phenomenon "[its] the eternal quest for *SherlockBugs*, which took

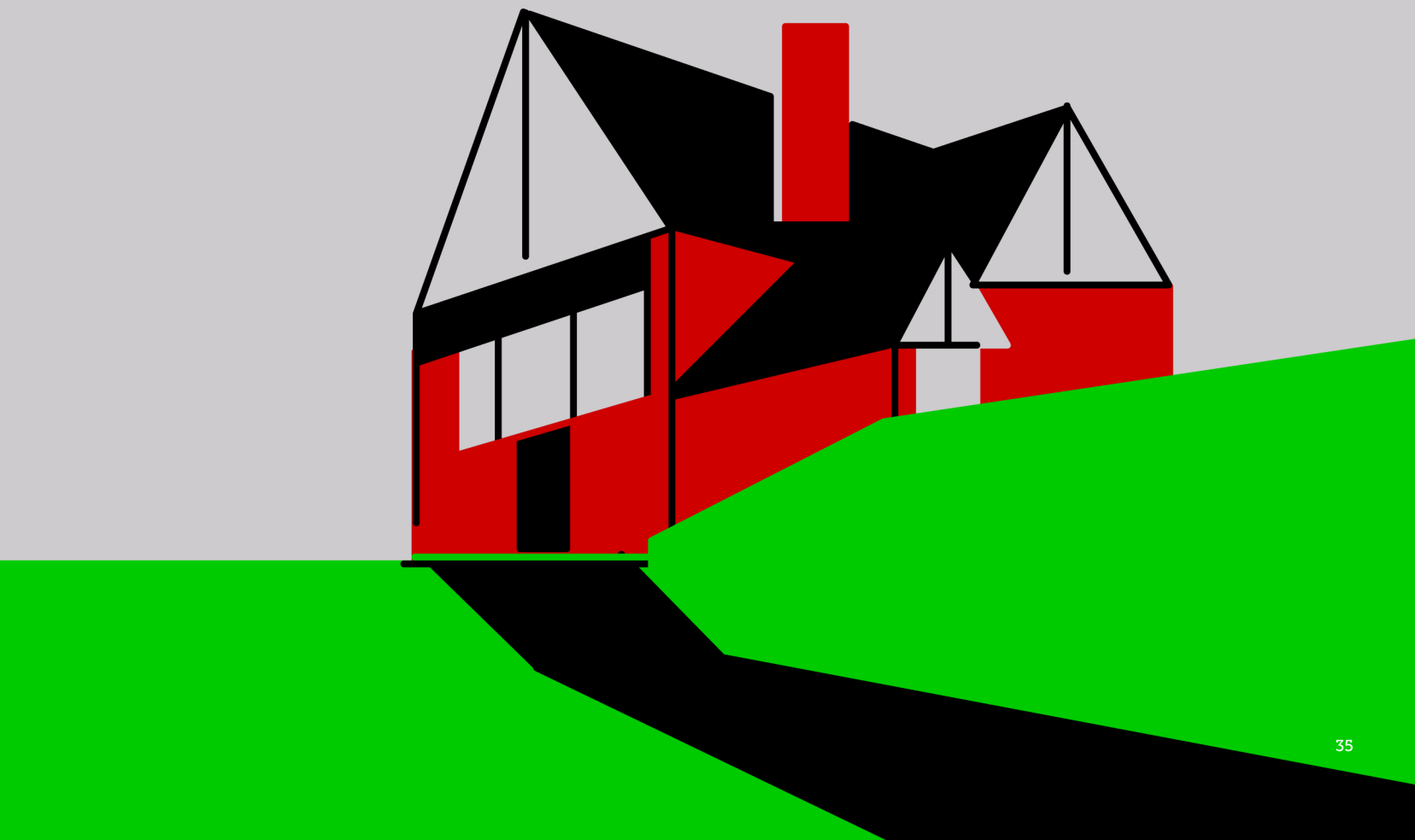
BUGS, BUGS, GLORIOUS BUGS



Both *The Hobbit* and *Sherlock* remain popular, not just because of their epic scale and ambition, but also for the memorable emergent behaviours [or bugs] that both games exhibited. After discovering you could fight the chest in Bag End, or sit on Dr Watson's lap the quest was on for adventurers to find as many HobbitBugs or SherlockBugs as they could.

The glorious example **[above]** is taken from the Commodore 64 version of the game, and involves taking a cab to Kings Cross Road, asking the Cabbie to follow Sherlock and then telling him to get in one of the steam trains which soon whisks him away.

Sherlock is then free to return to his Hansom Cab where a quick search of his inventory exposes a delicious listing of the game's object table.



over from *HobbitBugs* as a relaxing pastime for Melbourne [House] adventurers."

Many blatant bugs and vocabulary frustrations remained in the final version. Two cabs could in the same location or driven inside buildings. Unconscious policeman started talking and stopping your progress despite their incapacitation. NPCs would freeze, and you were unable to GET both disguises that appeared in the game, but could WEAR them. The Play It Again Australian and New Zealand digital heritage and preservation website has a wonderful video with an orchestrated dance between Watson and Holmes in the initial location, getting in and out of armchairs and sitting in each other's laps. Out of memory errors plagued some versions, with Melbourne House stating that conversations were stored in memory to tailor the narrative, but these soon ran out of space and resulted in a crash.

[CN] But this is an insane complex game. Take for example the subroutine which checks if two objects (which includes characters, as these are 'objects' in the game's context, too) can "see" each other. There are so many cases to handle (is object A carried by someone, is he nearby, is it in a container, is this container broken or closed, is this container in another container, is there light, etc.) that of course it will never handle each possible case correctly. This e.g. leads to a lot of strange emergent effects in context with the cabs.

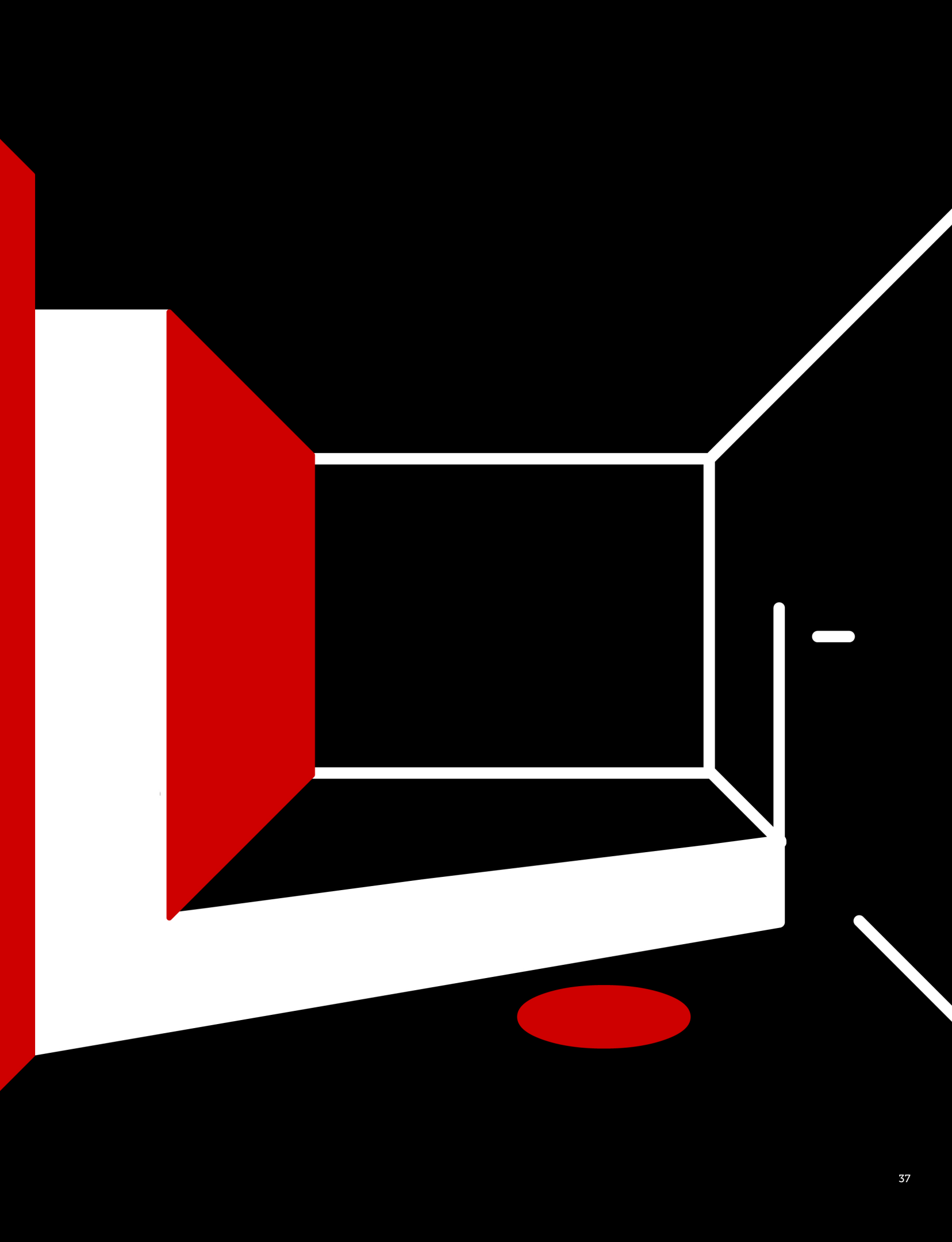
Tony Bridge in PCW magazine in January 1985 confirmed the memory bug, reporting that saying "HELLO" to Watson at the start of the game should be avoided. Mitchell backed up the claim, warning Keith Campbell about talking to the sidekick too much: "His character continues progressively to consume more memory in the computer. So, don't talk to him too much or Watson's knowledge could completely fill the available space and give you an out of memory error!" This was no doubt Mitchell's attempt to develop more advanced AI for the *AnimTalk* feature, and to implement a basic version of an Eliza [a learning natural language processing system] program. Despite that, it's still surprising that given the knowledge of such a flaw, Melbourne House still chose to release it without some failsafe.

In the end, *Sherlock* took an astronomical 18 months to develop, eventually being published for the ZX Spectrum [the Commodore 64 version would come later] in October of 1984.

Production delays were a constant issue. Four months after the game's initial June 1983 announcement, an interview with Popular Computing Weekly assured readers that the Spectrum and Commodore versions would be released in January of 1984. Melbourne House ran several teaser adverts ["Challenging, sophisticated, advanced, extra special. A real adventure - coming soon"] in the first six months of 1984, but press speculation heightened as time passed and no game appeared. It caused a PR headache for Milgrom and Byrne, who dispatched Mitchell to do another round of media interviews. He appeared in PCG magazine in July 1984 giving somewhat nebulous replies about the game's capabilities and content. The interview concluded with the disappointing statement that "Melbourne House have announced yet another delay in the production of *Sherlock Holmes.*"

The amount of time taken is understandable given the circumstances of development for the game, and the environment in which it was developed; a polar opposite to the team-shared workload of *The Hobbit.* Mitchell for the best part was on his own, under immense pressure attempting to decipher the un-documented [or at best minimally documented] code left by Meglar and written in a style alien to his own. It was this code, and not that of his parser that powered the game, so it was crucial that he spent as much time as possible understanding its nuances. Add to this his difficulty with porting Z80 code to 6502 ["converting *The Hobbit* was a horrible job" he told PCW] and his other responsibilities to the ever-growing entity that was Melbourne House. The studio employee number expanded, its games catalogue grew every month and Mitchell himself was juggling extra projects, working on *Melbourne Draw* [with Tang, released in December 1983] and the underwhelming *Mugsy* [released in July 1984].

[CH] Actually if I look at the game as a whole it surprises me it only



took 18 months to create it. Take the plot for example. It consists of two crime cases, suicide and blackmail, conventional riddles (the coded messages) and an action part (the pursuit through London, requiring Sherlock to take the right means of transport to arrive on time). Creating this logical consistent script and making it suitable for micro-computer implementation is fascinating by itself. Then there are the technical limitations of the machine: again they had to fit a whole world with geography, physics and personalities into the 40 kB available on the ZX Spectrum.

Finally, in September 1984 it was announced that *Sherlock* would premiere at the PCW Show in October of the same year. Mitchell [complete with *Sherlock* sweatshirt] attended in person to promote his game, alongside other luminaries such as Scott Adams, who was on a rare visit from the US to launch his new Adventure International *Spiderman* adventure. David Lester's November 1984 Crash magazine report from London's Olympia proclaimed that *Sherlock* had restored Melbourne House's reputation after substandard products *HURG*, *Tetradactyl* and [Mitchell's own] Mugsy. Unfortunately, In the time between *The Hobbit* and *Sherlock*, genre rivals Level 9 had released over half a dozen adventures, including *Lords of Time* and *Snowball*, and had started to push the boundaries of what could be done in 8-bit games. Level 9's A-Code scripting was still missing a compelling NPC engine, and their ideologies felt different – but their storylines and puzzles were tightly crafted. *Sherlock* had to deliver everything and more, but came to market in some rather underwhelming retail packaging. It shipped in an oversized Melbourne House box [with artwork by noted Melbourne designer and AGDA Hall of Fame member Lynda Warner], which enabled it to stand out from the rest of the gaming shelf, but the contents were disappointing. Where *The Hobbit* had scored a hit by including a copy of the Tolkien novel [thus bulking the packaging into an even more oversized box], there was no such gimmick for *Sherlock*. Alongside the instruction booklet and the beginner's guide, the only feelie was a stale typed sheet that contained a fragmented railway timetable, detailing the times of train between various underground and over ground stations [coincidentally one that served Leatherhead].

Nevertheless, media reception was ecstatic. Keith Campbell in C&VG, despite listing a plethora of bugs, said that *Sherlock* was "an intriguing and absorbing game", with the magazine featuring it as one of their top ten games of the year. Derek Brewster, Crash magazine's adventure guru [awarding the game 8/10] was impressed with the game's complexity and that it's "most impressive feature [was] the convincing way in which the leading characters go about their interrogations." To complete the praise, Sinclair Programs heaped accolades on Melbourne House and *Sherlock* branding the game "as exciting, as intriguing and as puzzling as *The Hobbit*."

Unlike *The Hobbit*, that was ported to virtually every conceivable platform, including the Oric, MSX and BBC, Melbourne House only converted *Sherlock* to the Commodore 64 [and a fabled version for the ZX Spectrum on Wafadrive]. The C64 version [see boxout] hit the shelves a month or so after the Spectrum version, and featured several new location graphics, music, and its own implementation of *SherlockBugs*. Given the additional development time given to port the game to the Commodore 64, a second bug-fixed Spectrum release didn't appear, despite World of Spectrum and other Spectrum repositories reporting that a second revision of the game was released.

[CN] I found that both files [v1 and v2 on the repositories] contained the same code [that was] maybe packed a little bit different in the container-files. This would fit my information that I have not read of any error-corrected release. As Mitchell and Megler used the loading screen of *The Hobbit* to indicate a version "1.2" Mitchell might have done so in *Sherlock* too, [but it] wasn't the case.

Not even the Amstrad CPC464, a machine that would pose little difficulty for the programmers, because of it's shared Z80 processor, received its own version of Sherlock. Perhaps the reason was that the ZX Spectrum version failed to live up to the expectations in terms of sales, or it could have been a result of Mitchell's disillusionment with the project. The game packaging prophesied that the average completion time would be "several months", though internally the company estimated that it would take over a year to figure out. To Melbourne House's surprise, it was solved within 6 months. Northumberland brothers David and Paul Cunningham sleuthed their way to success by playing the game every weekend for three months, telling C&VG magazine that "we weren't allowed to play it during the week because of our homework." Even though it was vogue at the time to offer rewards for the first person to complete games [Domark with *Eureka* for example], Melbourne House hadn't offered a prize, but recognised an opportunity for a good piece of PR. They whisked the two teenagers, along with C&VG Adventure luminary Keith Campbell to London for an all-expenses paid slap-up meal at the Sherlock restaurant.

After *Sherlock*, Mitchell developed the truly epic *Lord of the Rings*, *Shadows of Mordor* and *The Crack of Doom* [that was only released in the US] but didn't remain in the games industry long enough to be part of the move from the 8-bit era to 16-bits and PC: A pity given his many interviews where he expressed a desire to see those machines and capabilities emerge, and his imaginations for what the micros could offer his adventure writing. "I don't think we will see much advance on the sorts of things being achieved now on 89-bit micros" he confirmed to PCG magazine, "when we have 16-bit machines [...] there will be [...] more scope for AI in adventure games." *Sherlock* no doubt bumped up against the confines of 48K, and given 512K of RAM and access to random-access media on the ST and Amiga I'm sure he could have contributed to a Magnetic Scrolls or latter-day Level 9.

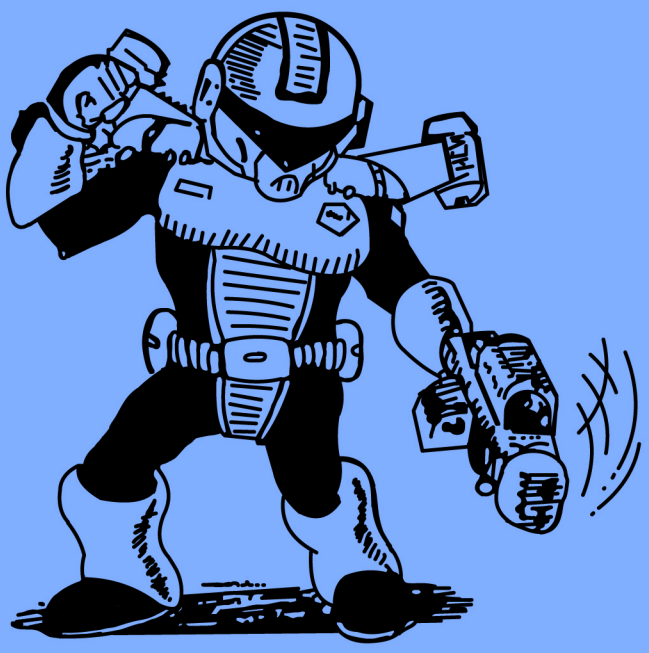
[Opposite] Crash magazine featured Sherlock Holmes and Dr Watson on the cover of its third issue, published in April 1984. The artwork by Oliver Frey incorporated a reference to Melbourne House's *HURG [High level User friendly Real time Games designer]* alongside Sherlock to promote a huge feature on the Australian publisher.

CONVERSION CAPERS

After the pain of single-handedly converting *The Hobbit* to the Commodore 64, Philip Mitchell enlisted the help of Melbourne House for the port of *Sherlock*.

Gregg Barnett had been recruited by Alfred Milgrom as Beam's Commodore 64 programmer and his first job was to revisit Mitchell's version of *The Hobbit*. After creating a second, more elaborate adaptation with sound and graphics [making use of the Commodore 64's disk drive] he went onto port other Spectrum classics *Hungry Horace* and *Horace goes Skiing*. Despite those huge titles, Barnett is probably best remembered for his sublimable beat-em-up *The Way of the Exploding Fist*.

The C64 version of Sherlock suffered from its own bugs [**see elsewhere**], included an extra location image, and carried an intriguing glimpse at a high resolution, and more detailed version of the game on its box artwork, perhaps from a non-functioning mockup.



IAN SMITH & SHAUN MCCLURE

Ian Smith and Shaun McClure have been friends since 1978. They both grew up in Mexborough, Doncaster and attended secondary school together, albeit a year apart in age. Both loved the new and exciting world of home computing, and it was natural progression for them to collaborate in creative ways using the new technology.

As with many adventure authors, it was the arrival of Sinclair's affordable ZX range of home micros that was the catalyst for Shaun and Ian's endeavours. Ian's coding prowess came from typing in games from the pages of popular magazines such as Sinclair User and Sinclair Programs, and Shaun's artistic flair was apparent from an early age; and he had wanted to pursue a career in illustration, primarily for book covers.

They developed a much-loved graphic text adventure called *Excalibur: Sword of Kings* [**see Issue 07**], building a custom adventure engine programmed by Ian that incorporated a clever bitmapped graphics routine to draw artwork created by Shaun.

Excalibur was released by Alternative Software in 1987 and was a modest success for the Pontefract outfit earning both Shaun and Ian enough of a royalty to incentivise working on more adventures together. Their next game, *Hit*, short for hitman delved into the world of Film Noir, and told the story of Ricky Swift, set in the era of prohibition, who sets out to protect an old school friend, the current Mayor of Chicago from failing foul to a gangster's assassination contract.

[Ian] We had decided to make the plot a little more complex than *Excalibur*, perhaps as a reaction to reviewers' comments about *Excalibur* being too simple. We toyed with a few different names for

[Above] A speakeasy scene from *Hit*. Ian Smith and Shaun McClure's clever graphics engine disassembled illustrations into component parts, meaning they could be duplicated, mirrored and in many cases reused over and over again.

the title, but we liked the term *Hit* and so that's what we ran with. We did not release it under any other alternate titles.

Hit firmly placed itself into the adventure noir genre, alongside other games of the category, including Fergus McNeil's seminal *The Big Sleaze* and Mastertronics *Play It Again Sam*. Another game, *Mugsy*, more of an animated interactive comic strip than a traditional text adventure, from the pen of *The Hobbit* co-author Philip Mitchell, was a big influence.

[Ian] In planning and developing the game, we had been inspired by watching classic films such as The Maltese Falcon, The Big Sleep and Casablanca. [I] had also played the Spectrum game *Mugsy* and that must also have remained in our consciousness. The characters/locations within the game were not based on any real-life individuals or places.

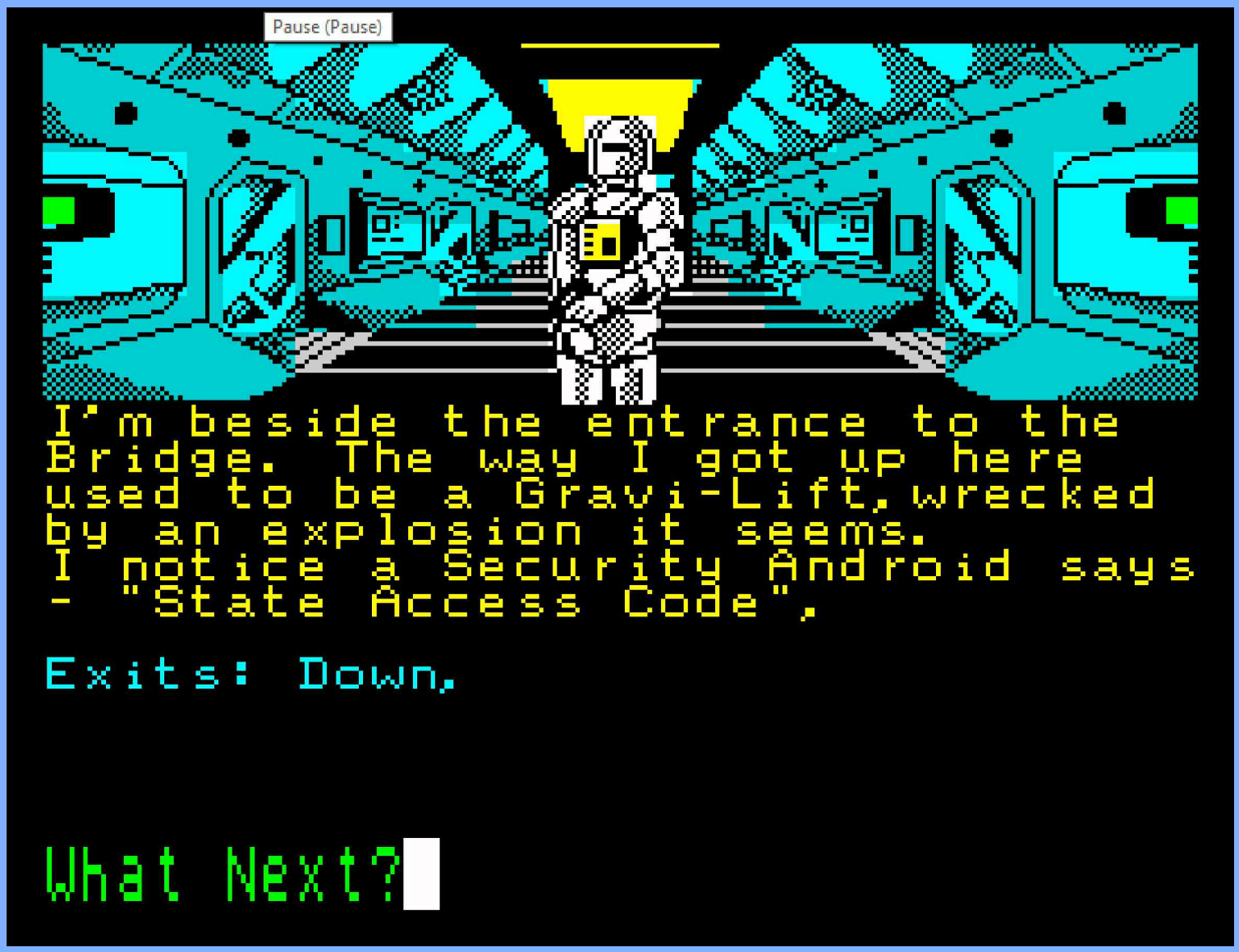
[Shaun] [Ian's] new thing [...] was Film Noir, and I think he'd even bought a director's chair with the actual word "Director" painted on the backrest. He'd become a proto-film buff and so I suggested that the next one was our take on a Bogart movie. [I was] aware of *Mugsy*, but hadn't played [it] or *The Big Sleaze*.

Ian had drafted in his Dad, Tom in the latter development of *Excalibur* to help get the game published. Tom brought with him a useful toolset and changed the duo's approach to programming. His background was extensively in engineering and was a draftsman by trade. He was mathematically gifted and applied his knowledge in transforming the algorithms that Ian had used to compress Shaun's graphics in the original game. In the next incarnation of their adventure engine, they both wanted to squeeze as many graphics into the game as possible.

[Ian] Despite being unfamiliar with assembly language, [Tom] examined the sections of coding we used to compress and store the game's graphics and came up with more efficient ways of saving memory space. I still do not understand to this day how some of his mathematical formulas worked, but they enabled us to store more graphics than we had previously thought possible. Having said all that, *Hit* remains a relatively short game due to the restrictions of the

CORRECTIONS AND CLARIFICATIONS

Many ZX Spectrum repository sites mistakenly credit Ian as being the author of CRL's *The Warlock's Treasure* and that *Excalibur*, *Hit* and *ARC* were authored using *The Professional Adventure Writing System*.



[Above] Ian and Shaun's final adventure *Alien Research Centre* is a game of high polish, terrific artistry and technical accomplishment. With the help of Ian's dad, Tom, the game was able to squeeze in a huge amount of graphical locations and even digitised sound effects.

48K Spectrum's memory.

[Shaun] Tom was a clever guy and thought of ways for us to make memory compression techniques mostly - his background was in electronics. He also found a few "oops" things in the game - like the mayor being drunk, when the game was set in prohibition time [and] the well-stocked bar that was in the game. We were still kids you have to understand, and we'd not considered things like that!

There's plenty of polish and refinement to the graphics in *Hit*, with the additional attention to detail really showcasing the Spectrum's high-resolution graphics. The clever used of dithering – a technique that used crosshatch patterns to produce shadow effects and other colour tricks - exhibiting that Shaun was progressing as an artist.

[Shaun] Yeah but it was also a much grittier game - you can tell because the houses look like they're in Yorkshire instead of New York. [...] I have been labelled "Stipple lord" [stipple being the creation of a pattern simulating varying degrees of solidity or shading by using small dots] more than once in my career though.

Hit was ready for publication in 1989. In Britain, the appetite for text adventures on the high street was waning, especially with mainstream publishers. The 8-bit market was shrinking and the emergence of 16-bit computers and games that took advantage of their enhanced capabilites such as *The Pawn* marginalised independent authors still using 8-bit tools further.

Shaun, Ian and Tom whilst searching for a publisher for *Hit* worked on their next game, called *Alien Research Centre* [*ARC*]. It was typical science-fiction fayre, set onboard a stricken space station [the Alien Research Centre] that requires repair. To escape from the station, and back to the safety of Earth the adventure player would encounter a squad of malfunctioning security robots and a horde of alien beings, probably created by the centre's experimentation, who have been able to escape and roam free due to its damage.

[Shaun] We basically just liked Sci Fi, and it was a genre we'd not done yet.

[Ian] I think we were inspired by a number of horror sci-fi films and books, chief amongst them being Alien, Dark Star, The Thing and Inseminoid.

Shaun initially designed the game, but in the documentation Ian and Tom take centre stage for their contribution to the programming and technical aspects behind the endeavour.

[Ian] I honestly cannot remember who was responsible for the majority of the game design or puzzles. However, my dad Tom was on the credits largely for his technical contributions.

BELOVED SEQUEL

In August 2019, John Wilson penned a sequel to *Alien Research Centre*, entitled *Return To Oblivion* which included a brand new piece of Shaun McClure artwork.

[John] I loved *Alien Research Centre*. [...] Sadly the game never reached the audiences it deserved but those who played it always liked it and, in my opinion, there should have been a sequel. More than 30 years later I took it upon myself to produce a small tribute to Ian Smith and Shaun McClure in the form of Alien Research Centre 2 - Return To Oblivion. Its not a patch on the original, but it should still provide a moment or two or entertainment - famous last words!

To download *ARC2* visit John's Zenobi website:

http://www.zenobi.co.uk/2019/08/alien-research-centre-2.html

Both *Hit* and *ARC* displayed a more mature approach to narrative, a hint that both Shaun and Ian, with the help of Tom, were growing as writers. In terms of a technical achievement, *ARC* stands head and shoulders above *Excalibur* and *Hit*, really pushing the capabilities of the Spectrum and demonstrating the evolution of Ian's programming talents and Shaun's mastery of the Sinclair machine's difficult graphics system. *ARC* was supremely polished, with a gorgeous custom font face designed by Shaun, and they somehow managed to cram in even more detailed and varied graphics, into a far bigger location count than both its predecessors.

[Ian] The enhanced graphical capabilities were predominantly a result of my dad's contribution. His mathematical and analytical skills meant we could include much more content with ARC than our previous two titles.

[Shaun] Compression was getting better and we made the graphic area slightly smaller.

ARC featured sound effects - something of a rarity - even at this late development of text adventures, having sound was highly unusual [Gilsoft's much used tool *The Professional Adventure Writer* didn't really support sound, apart from a basic BEEP action, and attempt at effects or music had to be done with external routines]. Ian used a programming trick to shake the whole screen to accompany the noise of the player firing a disrupter weapon in the game, and when taking on the Vapour Wraith and other aliens. In one part of the game, various digitized samples can be triggered, making the accomplishment even more impressive.

[Ian] Memory fails me (pun intended), but I do remember enjoying the challenges of coding sound and special effects on the ZX Spectrum. A lot of experimentation took place! [...] There is at least one sampled sound in there, and possibly more than one. That was one of the elements I enjoyed playing around with from a coding point of view. The Spectrum had very limited sound capabilities, so you could not expect to create the sound of a dying creature's death throes using a sequence of BEEP commands. There is at least one creature in the game whose last "words" were a result of yours truly doing my best strangled roar into a microphone, and then digitally manipulating the recording.

The quality of the location graphics moved up a notch. The symmetrical engine first developed for *Excalibur* that saved so much memory made a third outing in *ARC*, with Ian and Shaun making clever use of their "transfers", or independent objects such as chairs and tables, to obfuscate the fact that the base images were mirrored. The way that the pair worked on the game to produce the imagery didn't change. They rarely worked side-by-side instead opting for Shaun to work at home creating illustrations and delivering them on cassette in SCREEN$ [the bitmap and colour attribute data] format to Ian. He then used his ever-evolving programming algorithms to take that data and squeeze it into the engine. With the more ambitious and compact encryption process, both *Hit* and *ARC* managed to introduce a detailed cast of characters, including the Garganadon, a beautifully drawn alien.

[Ian] I think they hark back to those films I mentioned earlier. We were both deeply in love with the sci-fi and fantasy genres, and particularly the darker side of those genres. And we both have vivid imaginations, so conjuring up a strange and dangerous character was something we relished. As for the game feeling like it moved up a notch graphically, that is due to Tom's graphics compression formulas, and Shaun's artistic talent.

Surprisingly, one aspect of presentation that did disappoint the player, was the sparse endings to both games – a screen of text.

[Ian] You know, in all our games I don't think we paid much attention to the endings. In retrospect, that is a shame, as the player deserves a better reward.

After they finished *ARC*, finding a publisher was proving difficult.

[Ian] [...] The market for adventure games was in decline and getting the games published at all felt like an achievement.

They turned to a contact of Shaun's, John Wilson of Zenobi who was still strongly supporting independent adventure game authors. Shaun had a good relationship with John, having provided him with illustrations and loading screens for games in the past. Ian, Shaun and Tom approached Zenobi Software with an eye to getting both games published.

[Ian] Shaun already had a relationship with John Wilson of Zenobi Software, having provided some loading screens for him for other games. So it was a straightforward step getting in touch with John. He liked *Hit*, and agreed to distribute it.
[Shaun[He was one of the people that replied to my request to do free work. I worked on *Fuddo and Slam* and *Bulbo and the Lizard King*. Being Scottish he had no choice but to take my free artwork. He's remained in touch on and off ever since.

John enjoyed both adventures and appreciated the graphical work that Shaun had produced. He bundled them both together as a single release, making the offering commercially stronger given that *Hit* [making it in effect a freebie title] was a much shorter and more linear game than *ARC*. With the trademark Zenobi packaging, John bundled a delightful two-page graphic comic that introduced the narrative of the game.

[Ian] I had forgotten about that! Looking at the comic again, a lot of effort was put into it by Sean Doran. I believe he was one of the contributing artists for a number of Zenobi titles. I like the comic, and think it adds greatly to the overall atmosphere of the game.

Hit and *ARC* were Tom and Ian's last adventures. With respect to Zenobi, the disappointment of not being able to find a high-street retailer such as Alternative Software for adventures showed them that the writing was on the wall. Shaun moved on to work for Sheffield based development studio Wyse Owl Software to create graphics for their work-for-hire titles, and Ian went to University.

[Ian] I think Shaun and Tom would have continued with another game in the series, but I wanted a "proper" job and the stability that goes with that. Sometimes I regret that decision, but who knows which path is the right path to follow in life ... it is something of an adventure after all.

Shaun is currently working on a new adventure game using Chris Ainsley's *Adventuron System*. Ian has no plans at all to return to adventuring, but has enjoyed observing the revival of interest in his titles, including the remake of *Excalibur*. They are both still in touch, occasionally seeing each other when they can, despite the distance and demands of family life.

DESERT ISLAND DUNGEONS

Lost at sea, **Shaun McClure** and **Ian Smith** are washed onto a remote desert island, with only each other and five text adventures to pass the time while hoping to be rescued.

Ian Smith

I am relying on memories and feelings for games I played decades ago. All of the games were on the ZX Spectrum platform.

Snowball and ***Lords of Time*** by Level 9 - I found the location descriptions in these adventures really transported me to another place and time. The puzzles also seemed to me at the time to be very mature and took a lot of effort and thinking time to solve. I never completed either game, but they kept me hooked for quite a while. They also had a sense of humour which resonated with me.

Adventure 1 by Abersoft - This was a remaking of the 1970's mainframe based *Colossal Cave Adventure*. I don't think I was aware of that fact at the time of playing, as we didn't have the internet back then. I don't know how faithful it is to the original, but I remember loving this version. The locations were so evocative, and they genuinely conjured up a feeling of claustrophobia and being trapped in a remote underground location. Again, I never completed this one.

The Hobbit by Melbourne House- I think I loved this one more than all the other adventure games I played at the time. I even remember being too sick for school one day, but somehow getting away with playing this game all afternoon. It helped that I was a huge fan of the Tolkien books to begin with, and I felt this game interpreted the book's atmosphere very well. I also liked the AI non player characters who displayed an element of free will. Though they could be very frustrating! I liked *The Hobbit* so much, I actually completed this one!

Bored of the Rings by Delta 4 - Given my previous choice, this one might be a surprise. I imagine some Tolkien fans would have found this title to be sacrilegious, but I found it immensely amusing. It really appealed to my sense of humour. I don't think any other adventure game made me laugh out loud. And for that reason alone, I would have to include it in my list of desert island adventures. However, I also think it was entertaining and well written, and had enough reverence for the original books to keep their fans from being alienated.

Shaun McClure

Probably the *Zork* series on PC, I just like them.

THE QUEEN'S FOOTSTEPS

Davide Bucci's latest adventure is a fast-paced thriller of intrigue, set during the height of European interest in Egyptology at the start of the last century.

Davide Bucci is a relative newcomer to the modern text adventure scene. His first game, *Two Days To The Race* was released last year and passed the majority of the community by. Now David is back with his second game, *The Queen's Footsteps* set in Italy in 1904.

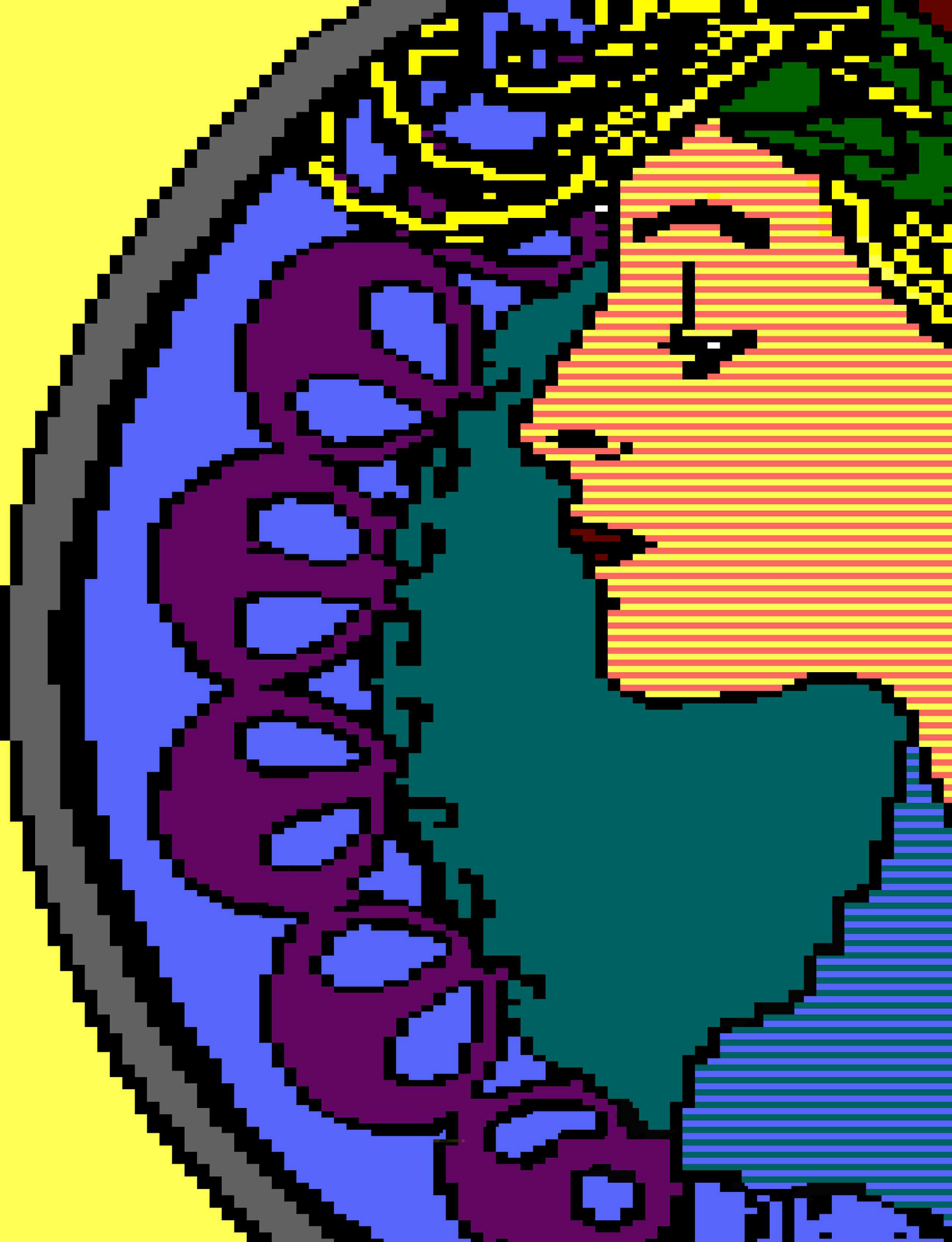
You play Emila Vittoni [the second appearance of Davide's female protagonist], an archaeologist returning from her latest dig, alongside colleague Ernesto Schiaparelli, the director of the Egyptian Museum of Turin. The dig has turned up many treasures, and the duo have documented, and crated boxes of artefacts excavated from the ancient tomb of Queen Nefertari Meritmut [the first of the Great Royal Wives of Ramesses the Great]. The artefacts are sent ahead first by stream ferry, and then by train, heading for the museum. Before leaving herself to follow them, Emila accepts an invitation to join a sumptuous reception organised by wealthy art lover, Eugenio Collovati, Count of Raligotto.

The adventure begins the day after the party, with Emila in her hotel room, packing, ready to leave. On exit from the hotel, she's handed a telegram, sent by Schiapearelli, warning that several of the treasures have gone missing during transit. She must step out onto the streets of the town, running into unknown dangers, a cast of characters and intrigue in this four-part adventure in order to find out what happened.

The Queen's Footsteps is written using *The Adventure Writing System* [*AWS*] - a tool created by Aristide Torrelli to easily write interactive fiction games. Davide has written his own utility to convert from *AWS* into C which enables him to create games for multiple home computers. The result is an adventure that has a nippy engine and a nice customised and proportional text display on the majority of machines. He's based his adventure on real-life locations, characters and events, and thus is able to set the scene with some very detailed and well-written prose.

Footsteps is getting a physical release, courtesy of Poly.Play who continue their much-appreciated support of the retro game community, and the text adventure genre in particular. It is available on a range of home computers and formats, complete with lavish artwork, a MicroSD card with disk images and a bunch of feelies including a poster and pocket mirror in a linen bag. Shipping is due to start later in April 2020, but with the worldwide COVID-19 panic still in full swing at the time of writing, expect substantial delays in any pre-order.

Author: Davide Bucci
Publisher: Poly.Play
RRP: From €25,00 + Shipping
Buy it from: https://www.polyplay.xyz
Download: https://darwinne.itch.io/the-queens-footsteps



Format: ZX Spectrum and Commodore 64
Publisher: Melbourne House
Developer: Philip Mitchell with Gregg Barnett
Artwork: Russell Comte
Release Date: October 1984

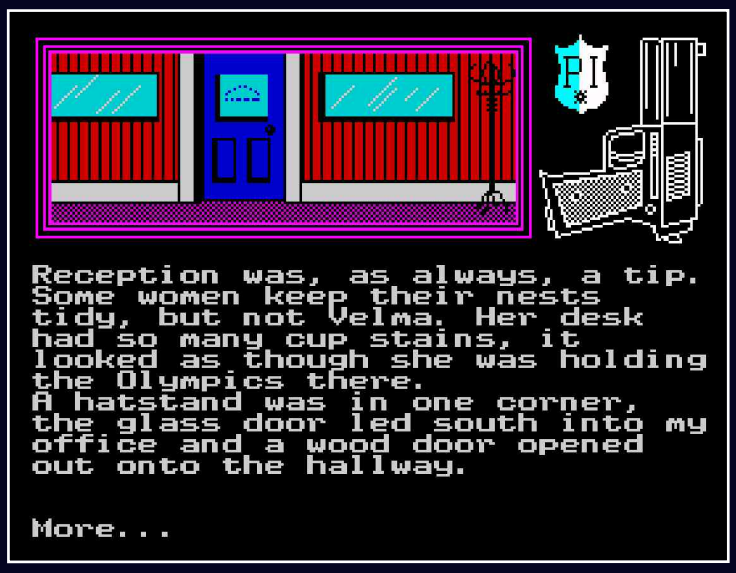
POST MORTEM

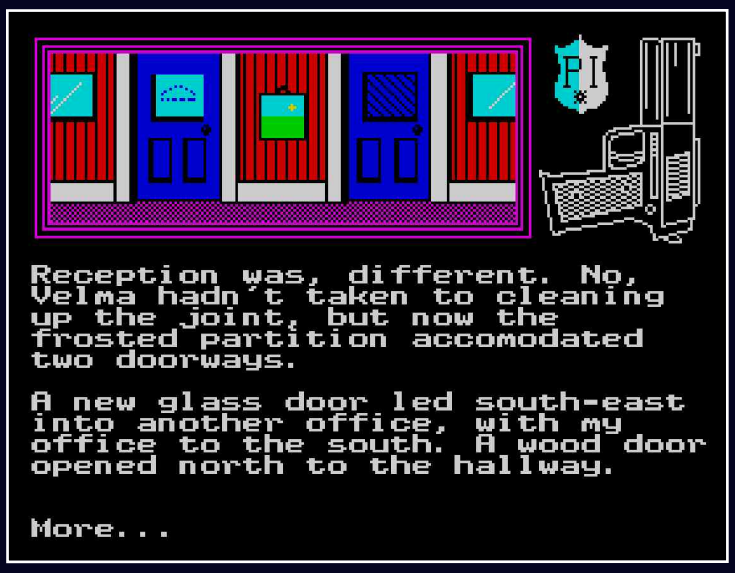
The Big Sleaze 2.5 was released in 2018 as a homage to Fergus McNeil's de facto detective noir adventure game. In a world of streamlined software tools and integrated development environments, **The Classic Adventurer** looks at the trials and tribulations of developing an 8-bit adventure game for the ZX Spectrum in the 21st century.

Detective adventures have always been popular, and I fell in love with the genre playing them on the Acorn Electron and ZX Spectrum during my teenage years. Sherlock Holmes was a popular figure to portray, with Melbourne House being one of the first publishers to bring Sir Arthur Conan Doyle's character to life in Philip Mitchell's *Sherlock* [**see elsewhere in this issue**]. 8th Day were another house to digitise the deerstalker in their excellent *The Raven* [penned by Stephen Kee and Alan Bolger], and when players moved on to the PC, Electronic Arts provided us with one of the best remembered games of the Holmes franchise with *The Lost Files of Sherlock Holmes: The Case of the Serrated Scalpel*.

Others authors penned gumshoe detective storys of the 20s and 30s, including Zenobi's *Hit* [see elsewhere in this issue], Atlantis' *Mafia Contract* and Mastertronic's *Play It Again Sam*. No other adventure captured my imagination more than *The Big Sleaze* written by the maestro himself, Fergus McNeil. I spent so many hours playing the game; it was one of the few adventures that I was able to make logical progress with. A friend and I completed the game in one long weekend during our teenage years, finally rescuing the dame, her father and sending King Kong tumbling to his demise taking the German Agent and his "hench-thing" with him.

I longed for a sequel, but Fergus moved on from Delta 4 and the satirical side of adventuring to something more serious with his





[Above] Capturing the look and feel of the original *The Big Sleaze* was key to providing the essence to any sequel. As well as duplicating the original game's font, the screen layout with P.I. badge and gun was painstakingly recreated. Spillade's office reception appears as a location in both games, with the original's graphics being reused and repurposed in this second outing.

Abstract Concepts label. After his collaborative title with Anna Popkiss, *Mindfighter*, the bottom fell out from the text adventure market and he never returned to write another game. He's now busy applying all the skills of his craft into being a hugely successful and respected author, although the crime and detective theme is still much loved in his DI Harland series of books.

At the start of 2018 the interest in adventures was flourishing. I wrote a blog post in January 2018 stating my new year intentions. One of those was to write a text adventure, by getting to grips with *The Quill* or *PAWS* – something I failed to do when I was a teenager. I had a few ideas for themes, but really wanted to explore the world of Fergus' B*ig Sleaze,* and set my sights on a sequel of sorts - more a homage and tribute to the original, as I 'd never be able to replicate the writing prowess and humour of the master himself.

Plot, Characters and Locations

The Big Sleaze is set in archetypical 1930s New York and I couldn't imagine Sam Spillade moving from the city, so wanted to set the sequel in the same location. My research started with a raft of reference material, from cultural pulp fiction comics and books from authors Enrique Sanchel Abuli and Jodi Bernet, through to traditional classics in print and film such as Dashiell Hammett's The Maltese Falcon and The Thin Man and a more modern take in James Ellroy's 1990 novel LA Confidential. Then I moved onto factual reference guides such as old maps and period tourist guidebooks of the city and region. These enabled me to write down locations, plot ideas, object ideas and characters that could be explored. The period books helped to shape characters, in terms of how they dressed, acted, and the sort of language they used.

I started to write down locations, paying attention to their descriptions in the reference material, but also taking specific details from the visual aids provided by photographs of images on screen. Photos certainly help to add textual richness and a depth of atmosphere to descriptions, for example when the player finds themselves on FDR Drive, I added the visual image taken from a photograph of tug boats easing along the East River in my final game. Travelling by car was vital to getting around in the original game, so I started to map out the connections between individual locations and districts and started to do more research into other forms of travel including the underground and overground modes of transport. Accurately modelling a train network and timetabling was a little beyond the ambition I wanted in the game, and so I stuck with Spillade still using the car. I wasn't sure how accurate Fergus had been with the movement of the PI between locations, but thought he would bring some semblance of realism. Turns out I was wrong. After looking at each of the journeys taken by Spillade, from location to location in his car he always drove via "a shortcut that I know" via another place. I presumed that this "place" really would be a shortcut, but on examining the route taken against a real-world map I realised that Fergus had introduced a clever joke to those that would only get it with some knowledge of the city's topology: These "shortcuts" were in fact always miles out of the way, thus Spillade would always, without exception reach his destination "some hours later" even if it was in the neighbourhood. I should have known.

Puzzles

Type in "how to write adventure games" into a search engine of your choice and you receive links about coding. There are very few resources out there that actually discuss the nuances of designing a game, what makes a good plot, how to pace the plot and move it forward and what makes a good and bad puzzle, or indeed what makes a good and bad adventure. There's plenty of advice on how to do the aforementioned in a linear way, such as in a film, but for a branching adventure I had to take a different approach.

I started by evaluating the games I liked, and played the games that I could find on the same topic and theme to analyse the things I mentioned before, pace, setting and puzzles. *The Big Sleaze* itself was the obvious benchmark but I also made notes from *Sherlock*, *Here Dies Whatever*, *Mafia Contract*, *Matt Lucas*, *Play It Again Sam*, *The Boyd Files*, *Detectiveland* and *Make It Good*. With walkthroughs in hand, I'd play through, but investigated what the adventure did if you strayed from the winning path, by doing the wrong thing – and noted how forgiving the titles were and how much hand holding they gave.

There's a treasure-trove of information in old books, magazines and fanzines. I assessed reviews of games from stalwart adventure luminaries [who all knew a thing or two about good adventure design] such as Tony Bridge, Mike Gerrard, Derek Brewster and Keith Campbell and poured over interviews with original authors such as the Austin Brothers, Scott Adams, Peter Killworth and Fergus McNeil. Accessing magazines is easy thanks to the Internet Archive [and it has a great text search facility], and thankfully they have a full Micro Adventurer collection [the adventure bible of the period] and a few odds and sods of other adventure journals such as Adventure Probe and Red Herring. Fergus has of course been interviewed many times throughout his career, and with the recent reconnaissance of retro gaming, he's appeared in modern magazines such as Retro Gamer and popular podcasts such as the Retro Hour [episode 53].

Perhaps the most useful reference, was The Adventurer Gamers Manual, written by the late Bob Redrup. Bob wrote an adventure gaming column called The Mad Hatter for The Micro User and he drew on his many years of experience breaking down several games into their composite parts, specifically the art of the puzzle.

Structure

With my raft of locations, potential characters and design considerations I set about writing the actual narrative. I felt that writing the game as a linear-ish story helped me judge whether the flow felt right, after all, we are supposed to be creating interactive-fiction. There's plenty of references for traditional writers that applies to those creating adventures- such as story beats and narrative advancement. It really helped to write *BS2.5* as a mini novella, padding out areas that I felt need embellishing, and adding extra areas of narrative to describe potential sub-plots, or branching to cover what happened if the characters strayed from my planned plotline. Once I had something I felt worked, I need to think about how to implement it.

Implementation

My fondness adventure memories were made on an Acorn Electron and ZX Spectrum, so I preferred to develop a game for both machines. *The Adventurer Creator* [*The Graphic Adventure Creator* on other micros] and *The Quill* were the best-known tools, with *The Quill* being the undisputed superior product and capable of supporting the game I wanted to write. Unfortunately the available memory to an Electron programmer using one of the utilities would prove too restrictive, so the ZX Spectrum was the only viable platform. On the Speccy *The Professional Adventure Writer* superseded *The Quill* [and incorporated all of its expansions such as *The Illustrator* and *The Patch*], and I have coveted it ever since its appearance in issue 40 of Crash magazine. It made sense to use *PAWS* to take advantage of all those features.

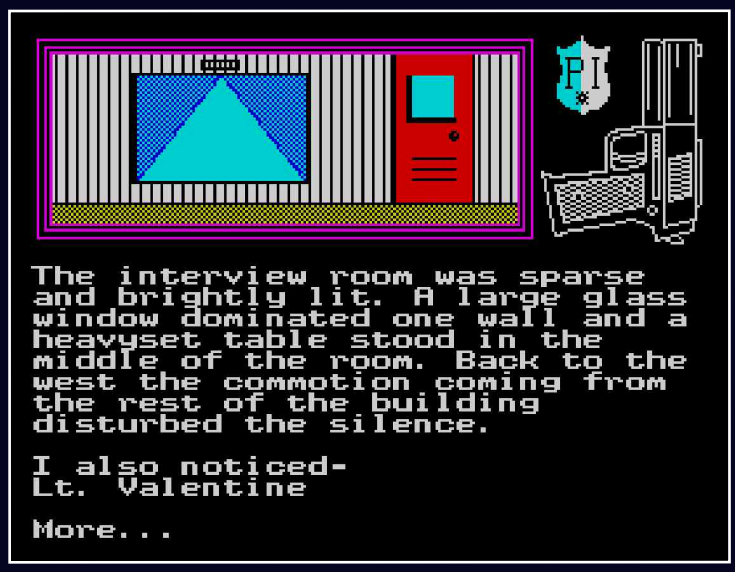
Johnathan Needle's *Spectaculator* is the emulator of choice when it comes to the ZX Spectrum, and it supports all models of the machine, including the ability to run the disk version of *PAWS* on +3 [the ZX Spectrum with a built-in disk drive] hardware. Running on the +3 with 128K of RAM would enable every available feature overlay [something that you had to juggle around if limited to the 48K version to save memory] to be loaded and still allow me to target the base 48K machine. There's a small amount of *PAWS* learning material out on the web, including tutorials from Larry Horsfield serialised in Sinclair User, Chris Hester's Adventure Coder magazine [that has a featured called *PAW* Prints] and Mark Cantrell's Process One 'zine. Since *PAWS* is built upon the scripting language used in *The Quill* you can adapt materials specific to the earlier utility, such as Simon Avery's Using The Quill: A Beginner's Guide, without much hassle. Starting with the excellent manuals that Gilsoft include, I began mapping locations and objects and adding numbers to each of them to ease reference. To aid my development I created printable structured templates that allowed me to record these entries more efficiently.

Considerations

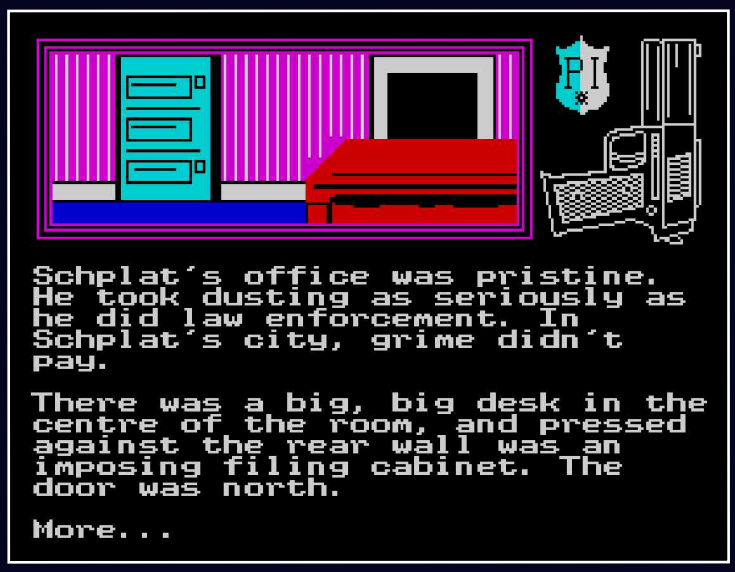
There is an expectation when creating a modern text adventure that traditional frustrations of early games are removed. I wanted to remove some of those, no instant deaths, allowing shortcodes such as X to be used for EXAMINE and signposting puzzle solutions. There's other annoyances that I find, that are usually dependent on the author and their construction of games and puzzles. Is SEARCH the same as EXAMINE? Is SEARCH CAREFULLY the same as SEARCH? These are just some pedantic constructs that frustrate me as an adventurer. For examine in the *Cranmore Diamond Caper*, EXAMINE TRACKSUIT doesn't reveal anything of interest, yet SEARCH TRACKSUIT finds a card. In the mind of the player, are the two really different? In the *The Jack Necklace* by the late, great Jack Lockerby and released by River Software [and later Zenobi Software] you can TAKE the newspaper but you can't EXAMINE it, nor READ it without picking it up first. Only when you OPEN it does a clue arise. Would you really read a newspaper without opening it, or not notice something inside it without doing so? There is a leap in faith in programming and puzzle composition here.

Considering and implementing minutiae like this forces the player into a mindset where they have to CAREFULLY examine everything, just in case they miss something. It's the same with adventures that have some puzzles that require a SEARCH of a location to undercover details. The inclusion of both possibilitys instills an unwanted behaviour in the player - it's akin to a badly designed modern point-and-click adventure that has you searching ever single pixel on the screen for a point of interested.

We can then continue to apply these implementation decisions to a wide range of problems. How do you standardise communicating



[Above] In an early scene from the game, The Big Sleaze 2.5 draws heavily on film noir, and the raft of period detective and cop films of the era.



[Above] Humour was an essential ingredient to every Delta 4 adventure. Here, Schplatt's love of housekeeping results in one of the better play-on-words contained within the text.


ONE WAY
West 39TH St
8 AVE

[Above] A one-off physical mockup of the game, complete with inlay artwork and in-box "feelies".

---

with other characters, a complex Sherlock-style SAY TO WATSON "TELL ME ABOUT CHRONICLE", or a simpler "ASK WATSON ABOUT CHRONICLE" or a "TALK TO WATSON" that has a defined conversation thread. OPENing, CLOSEing and LOCKing and UNLOCKing of doors have been used as puzzles themselves in other adventures. *Doomsday Lost Echoes* implements a neat feature, similar to its Access Card puzzles – if you have the correct card or key then it makes presumptions on the part of the protagonist and either inserts it or opens a locked door without asking the player to do so. In the original *The Big Sleaze*, the game forced you to TOUCH WIRES in the car every time before being able to travel. Good game design should support that once you'd solved that particular problem the protagonist now knows what it takes to start the car. I decided to get around this particular programming quandary in *BS2.5* by having the keys stuck in the ignition!

Mirroring the original

I wanted the game to feel like an extension of the original game. The obvious place to start was matching the screen layout and font thus giving the impression of a continuation to the series. I used *Spectaculator* to grab various screenshots and used *Photoshop* to measure the various attributes of the layout - what size the location graphic was, colours used, and the various plot and line requirements to compose the accompanying PI badge and gun.

For the font, by default, the Spectrum displays characters from a location in memory determined by the value of bytes in addresses, 23606 and 23607. This either points to the built-in ROM character set, or a location with the user-defined character set. Thankfully, *The Quill* uses the stock method for displaying fonts, and I was able to rip *The Big Sleaze* font from the game using the emulator's tools.

The last step of theming the game was to include the various subtle routines that Fergus had added to create additional charm - such as an occasional change in weather that was relayed to the player, or the indication that time was moving forward with the announcement via the "bong-ing" of various clocks "somewhere" in the city. The final part was duplicating the player prompts, such as the basic What Now? prompt, complete with the user-defined graphic and Fergus' neat use of colour.

Packaging Art

As with the theming within the game, the physical artwork that I wanted to include had to match or mirror the style of the original. My artistic skills are limited, so I found a stock gangster image taken from one of the reference materials I'd collected. I combined the gangster's body with the Piranha artwork of Spillade and embossed it over a building taken from another reference comic. The last element was a US street sign of the era, taken again from one of the source comics with the street names changed to reflect locations in the game.

I wanted to include a dedication to Fergus, both in the game and reflected on the packaging artwork so I recreated the DELTA 4 logo for the rear and included a small piece of "humourous" text as a quotation.

Feelies

Physical props are a reminder of the glory days of adventuring. They were a staple and popular feature [and a trademark] of Infocom's adventures. Very few British adventures included significant "feelies" until the arrival of Magnetic Scrolls who included a novella with *The Pawn*, but then enhanced each subsequent games with more offerings - for example, *The Guild of Thieves* came bundled with A Bank of Kerovnia account card, die, the "What Burglar" magazine and the Kerovnia Guild of Discrete Entry And Removal Operatives contract. With *The Big Sleaze 2.5* I was aware that the game would primarily be downloaded and played digitally, so the physical feelies may be something that would be overlooked by the casual player. With that in mind I created a series of "feelies" that added to the game's theme but weren't required to solve the game or key to the story – unlike, say, the answerphone message on cassette with Magnetic Scrolls' *Corruption*.

I designed a PI Badge, a New York postcard [adapted from an original design found online], a Museum of Metropolitan Art Poster [for the oncoming exhibition of the McNeil diamonds] and a beautifully shaped flyer [again from an original found online] for the Star Lounge situated in the Waldorf Astoria Hotel. The lounge was going to feature in the game, but in the end wasn't included – but I still enjoyed making the feelies and including the fictional acts that starred in the show - all adventure related. A glimpse at the Star Lounge's show listing would reveal such bands as Anita and the Sinclairs and Balrog John and his Zenobi Orchestra.

Launch

I did a little bit of tweeting to make the game available for download and play. I hadn't polished the game, nor really playtested it, but the best place for that would be in the hands of fellow adventurers. I pushed an "alpha" version, and created a holding page where the game could be downloaded hosted under The Classic Adventurer domain. It was sometime before The Classic Adventurer was really starting to gain traction, so the number of downloads has been a very small amount. With that in mind, it would have been better to push the game around the various forums and adventure groups, such as the vibrant community on *Facebook*.

What worked?

So, what worked? I hope I managed to retain some of the charm and the look and feel of the original game. There's no doubt it looks like a *Sleaze* game, with the familiar layout and graphics, and the font that is so perfect. I was never going to match the wit and wisdom of Fergus, who is a genius, but I made an effort in all ways to bring in those similarities of style where I could.

Perhaps the best part of the game, unsurprisingly, is the physical version that I printed and built myself. The artwork and accompanying feelies would match any retail product that was published back in the day. John Wilson of Zenobi kindly made a comment that he would have seriously considered releasing the game on his label given a little bit of a polish and brush up.

What didn't?

The game was a thorough learning process, so a lot didn't work. The puzzle design wasn't as strong as it should be. Perhaps the weakest was the effort that had to be made to explore a bedroom, and the twist where you have to FOLLOW one character to progress in the game. If you failed to FOLLOW or lost the character then it was game

over – there should have been more indication of that set of instructions, or at least some way to get back into the plot after that. The FOLLOW example was also one of the very few elements where I tried to implement a NPC. Getting to grips with programming the process tables in *PAWS* meant that they were basic and simple routines. I'd have liked more time to explore adding depth to their personalities within the game.

There were also other *PAWS* problems too. Getting to grips with the way that alternate nouns had to be coded, and general problems from using an emulated Spectrum as the IDE – these involved the complex way that the Spectrum keyboard operates to input symbols and other characters, and the control codes that can be entered to change font colour. Trying to implement these using an emulator relied on having the virtual keyboard open at all times. Add into that the quirk of the hardware running to a crawl when location text or message text was too long for the input buffer to handle. It all resulted in a lot of frustration.

In the end the gameplay and coding problems could have been ironed out in subsequent revisions of the game, but it wasn't heavily played, and a handful of those players provided feedback so it was shelved at that time.

What's Next?

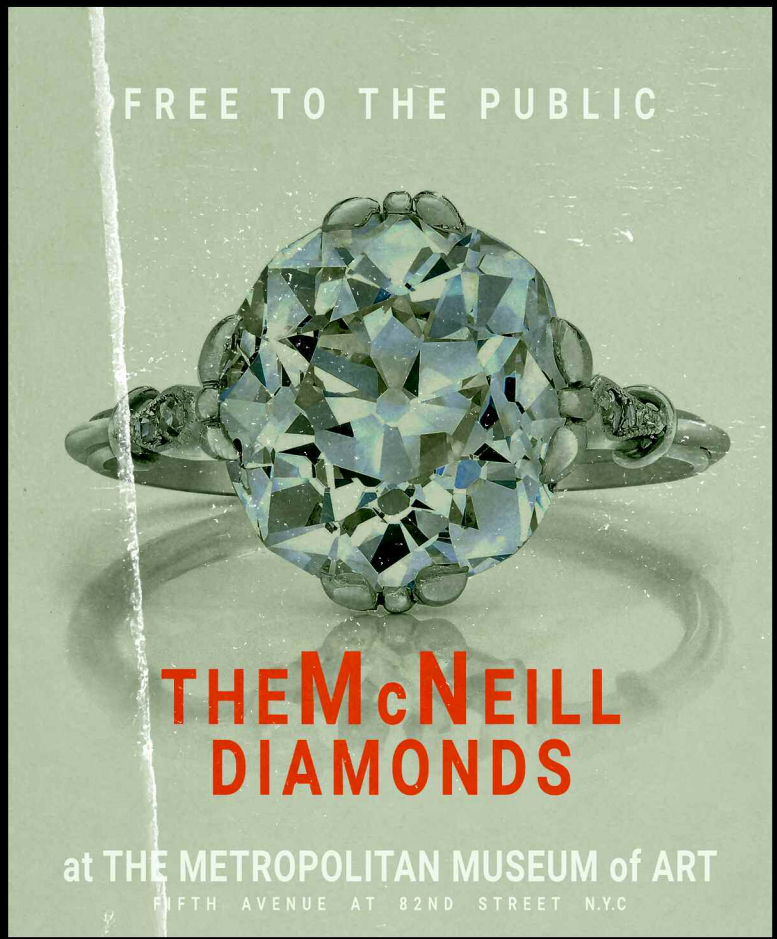
I'll certainly return to writing a new game, perhaps continuing with *The Big Sleaze* world, or delve into something completely different. This year has seen the launch of the brand new Spectrum Next computer with a promised enhanced version of *PAWS*, and it would be a good motivation to code another game that targeted this new and exciting addition to the Speccy family.

DAAD continues to gain traction, so if it supports the Spectrum Next it would enable me to get around some of the IDE difficulties [being able to use and editor in DOS] and frustrations in using *PAWS* directly in an emulator. *DAAD* would also enable the easier implementation of graphics [a pain in *The Illustrator/PAWS*] into a game.

NPCs always fascinated me in games, and I'd love to explore creating a more living and expansive world, such as those imagined by Veronika Meglar and Philip Mitchell, Trevor Hall, Linda Wright and the multi-character genius creations of Mark Cantrell.

There's no doubt there's never been a better time to write an 8-bit or 16-bit text adventure. The communities are friendly and vibrant and thanks to a handful of enthusiasts, more and more fan sites, interviews, reviews, new games, documentation, hints, tips, manuals and utilities are continued to be unearthed, archived and developed.

[Below] A closer look at the Starlight Roof and Metropolitan Museum of Art promotional "feelies".



DESERT ISLAND DUNGEONS

As the SS Classic Adventurer undertakes its final voyage, Captain **Mark Hardisty** picks five text adventures to accompany him on the long and treacherous journey.

Twin Kingdom Valley, Acorn Electron

My first computer was an Acorn Electron, so it will always hold a special place in my heart. It was bundled with *Sphinx Adventure*, but it was sheer fantasy and majesty of Trevor Hall's *Twin Kingdom Valley* that really fired my imagination and started my love affair with the genre.

The Greedy Dwarf, Acorn Electron

I never owned it, but did play it on a friends Electron. I never progressed past the first few locations either, but I coveted its evocative and exotic box art and for nostalgic reasons alone it's going into the suitcase.

The Big Sleaze, ZX Spectrum

Fergus McNeil and Delta 4's finest hour. Brilliantly written, flawlessly programmed and a wonderful sense of humour. Utter genius.

Bulbo and Lizard King, ZX Spectrum

I didn't have a great deal of disposable income as a teenager, and spent the meagre income from my paper round on computer magazines. What a delight, therefore, it was to find this introduction to John Wilson's loveable character included for free on a Your Sinclair covertape.

Sherlock, Commodore 64

I'd never played *The Hobbit*, so when I experienced Sherlock for the first time it delivered a totally immersive adventure and an enthraling story of murder and mystery that would unravel from the moment you asked Watson to read the Daily Chronicle. *Sherlock* is probably one of the best, albeit it buggy, adventures on 8-bit computers.

HUGH STEERS

In 1983, **Hugh Steers** along with **Anita Sinclair**, Ken Gordon and Rob Steggles formed Magnetic Scrolls, a British text adventure giant that produced storytelling and technology to rival anything produced by the US powerhouse Infocom. Now, with several of the original team, Steers is back with Strand Games, on a mission to remaster their masterpieces for a modern audience.

Ken Gordon, Rob Steggles and Hugh Steers grew up together in the London suburb of Woolwich. They were all in the same class in school, and Steggles often referred to his friends as being the "computer whizz kids", as they both took an interest in computing from that early age.

[Hugh] Computers and technology are today so ubiquitous that it's hard to imagine growing up in a world without them. Schools didn't properly have a "computing department", but instead, rather primitive computer science lessons, where the actual computing was done by sending punched cards off by post and getting the, usually negative, results back in the next lesson! [...] Imagine the amazement of being a kid growing up during the explosion of the home computing era. [...] You had a range of "exciting" games made by, each turn, re-printing little ASCII pictures on a paper roll, including *Hangman*, *Hunt the Wumpus* and *Animal*.

Hugh credits the rapidly evolving microcomputer scene as a major reason for their interest in computing. It was an exciting time for fans of technology. In Britain, in the early 80s, all manner of micros were hitting high street shelves, many of them commercial failures as rapid developments meant each computer was quickly overtaken by the next competitor. Another reason for their comradeship, according to Steggles was a shared love for Dungeons and Dragons, but Steers didn't fully embrace the fantasy storytelling worlds of Gary Gygax and Dave Arneson.

[Hugh] Rob did play a fair bit of it. He was very creative and able to adapt dynamically - as you would need to be to make interesting gameplay from random dice throws. I never really liked Dungeons & Dragons, I think randomised gameplay does not work, or at least does not produce optimum use of material. D&D gameplay relies heavily on the skill of the person hosting it rather than from the rules.

Instead of a human, Hugh preferred a computer to be the gamemaster, and started to explore the structured rules and logic of computer-based adventures, especially those from the Massachusetts giants Infocom. The more he and Ken delved into the genre, the more they were captivated.

[Hugh] [...] Of course, *Zork* was impressive, but I actually preferred the mystery games of *Deadline*, *Suspended* and *Witness*. For me, a real-world setting seemed much more interesting than a fantasy

world. Additionally, these games introduced characters into the plot. I always felt this gave games much more depth and scope. Everyone played Level 9's games and *The Hobbit* from Melbourne House, too of course.

It was through their explorations, and a shared acquaintance called Tony Lambert that they were introduced to a young woman called Anita Sinclair. Sinclair was another budding computer enthusiast who, at 18 was a little older than Steggles and Gordon, and on the cusp of leaving school. By this stage, Anita was already working for several local businesses as a freelance programmer.

[Hugh] Anita was inspired by the ongoing explosive development of microcomputers. She has always had an entrepreneurial flare and saw the incredible opportunity in technology at the time.

[Anita Sinclair] [...] We wanted to write a game, [but] we weren't certain that we wanted to write an adventure game. There was a game out at the time called *Hack*, that was a Unix based game, and I wanted to go a graphical version of that, because I thought it was amazing.

[Hugh] *Rogue/Hack/Nethack* [were] classic dungeon crawls. [...] These games have considerable variation in play and are a constant source of surprise - as all sorts of things can happen, in what was after all, just a text-based dungeon. It would be nice to have open-ended possibilities in a text adventure game.

[Anita] Ken was kind of ambivalent, but Hugh really wanted to write a parser. Once he got his teeth into his parser and showed me what it could do – which was amazing at that time, it was a no brainer that we should write adventure games.

[Hugh] Most of the early experiments were firstly to recreate the classic Scott Adams style, two-word parser. After that, it was a question of how this could be improved upon, e.g. four words "put key in bag" and then issues of how ambiguities could be solved. One of the problems, of course, was that the target machine in those days was very limited indeed. We're talking of 16K machines, out of which the parser is just a minor part of the whole product. Even a two-word game parser can be inspiring. You can do a lot with just two-word, VERB NOUN combinations, and when the machine actually know what you want to do, it's really quite cool. It's basically AI, albeit fairly limited.

Akin to a cast of a real-life role playing game, each protagonist brought with them a special ability; with Steggles' literacy turn of mind and creativity ["we saw Rob as an author that also had the talent to develop the dynamic type of fiction needed for an interactive story"]; Steers and Gordon and their logical and structured programmer brains; and Sinclair – glamourous, with a huge grasp of technology, business acumen and contacts – and a small stream of revenue from her freelance projects that generated funds for their fledgling enterprise.

[Anita] We had no money, we literally had no money so we would do anything we could to generate a bit of cash to pay for the computers, the office, and whatever else. Luckily Ken and Hugh lived with their parents, and they were very generous.

In 1983 they set up shop in a small office in Eltham in south-east London and called themselves Magnetic Scrolls. Along with the

freelance work, rumoured to be several unattributed arcade games written by Anita, they coded their first exploratory adventure together. Magazines of the time ran with a story about a rather tongue-in-cheek game called *The Willy Affair*. According to the articles, Ken and Hugh, in the midst of their A-Levels, sat up through the night, ironically before the Computer Science examinations, to create an adventure in which a limp and then subsequently erect "willy" played a large, ahem, part. The risqué subject matter was apparently down to Anita's substitution of said phallic term when she couldn't think of the word she wanted. "So, if I say to someone 'Can you pass me the Willy', anyone who knows me will know I mean the coffee cup or the box of disks or whatever," she embarrassingly told Your Computer.

[Hugh] Ah well, this is clearly fiction, because there was no computer science A level syllabus when I was studying! It's one of those myths that keeps getting recycled. When you're a games company and especially one making interactive fiction, all sorts of ideas are constantly being bandied about. It was a kind of test; come up with a wild and crazy idea and see if it works. However, there were actual games started or researched by Scrolls that were never made, and bits of these have been recovered from the tape archives. If I do happen to recover material for this *Willy Affair,* I'll definitely let you know.

In January 1984, Sir Clive Sinclair announced the new QL Professional Computer, from a typically extravagant launch event held at the InterContinental Hotel in London. The QL was the next big thing, taking home computers into a new generation powered by 16 and 32-bit CPUs.

[Hugh] Clive Sinclair felt that the QL could be really successful. It was an affordable 32-bit home computer when all the others were only 8-bit. The QL predated both the Atari ST and Commodore Amiga in the 32-bit space. [...] I think Sinclair did reach out to all sorts of avenues for his QL machine. We were one of them and we were also optimistic about it. It was luck that Anita knew Clive Sinclair. Consequently, we got some early QL machines to develop on.

[Sinclair] They were free. Because Clive Sinclair who was a friend of mine, and is still a friend of mine, gave me a bunch of them. [... Clive] was an incredibly clever man and could see the future in a way that other people couldn't see the future.

With Steggles and Sinclair writing the adventure, the text, the puzzles, the story and so on, Steers and Gordon quickly got to grips with the QL's new 68000 CPU, porting their evolving adventure toolkit from Hugh's ageing TRS-80 computer. The new suite of utilities and language was given a rather tongue-in-cheek name, to fit in with their London location – *ELTHAM*, it was short for *Extra Low Tech Highly Ambitious Metacode*. Within the Scrolls team it was known as the *System*.

[Hugh] The, so called *System* is a reference to two parts of the technology; the virtual machine, and the game engine. The virtual machine CPU was based on the M68000 architecture. [...] We called our version *ELTHAM*, which contained a subset of the M68000 instructions plus some new ones we invented. By emulating a variant of the 32-bit M68000 on an 8-bit machine, you are emulating "upwards". In other words, several 8-bit instructions are together making one 32-bit instruction - which is obviously the right direction to go for efficiency. In contrast to *ZIL* [*Zork Implementation Language*, a programming language developed by Infocom], you are

emulating an 8-bit architecture with, at the time, another 8-bit architecture, which isn't great, and even worse nowadays; you're emulating 8-bit with 32-bit which is even worse. The *ELTHAM* quasi 68k CPU was virtualised so it could run as 32-bit within an 8-bit address space. Banking in and out bits of code as needed.

The underlying adventure language was as complex as the explanation of the virtual processing taking place. *ELTHAM* separated the game logic from the interpreter onto objects, so that doors, items, characters, etc, within the game possessed the same attributes so they could be carried, opened, sat on, contain other objects, etc.

[Hugh] This is the other meaning of the *System*, which is the body of logic which ran the world simulation. The principle is that, with enough properties specified as data, the world could be simulated to the extent that you could move about and manipulate objects with just data and no new code. Once this is achieved, it remains to add the story. [...] The world simulator was there from the start. The data model was designed at inception. Of course, the world model did evolve as the games became more ambitious. But in all cases, representational data was accommodated within the original data structures that, from the start, had spare areas for expansion. Virtualisation did not happen on the QL because it had enough memory anyway. This was added later to allow games to be accommodated on 64K machines like the Commodore 64.

In the end, the Magnetic Scrolls *System* was an interpreter along with a series of source files that outlined the adventure logic and the game's scenario. Wrapped around them was a tool called *Fred23* that controlled the output of words needed by the parser, and extracted the text, the locations and their connectivity, along with the game objects and all of their properties.

[Hugh] In theory you could design the bare bones of a game, write a *Fred* file and emit a game that was playable without writing any code. But in practise you got a simulation that did not have any game play events or story. [...] You had to write the story in terms of triggers and hooks that existed in the system. For example, if a cookie had an "edible" property then, by default, you could eat it. It would then disappear, but nothing else would happen. Unless, for example, its edibility was poisonous, in which case you'd get ill and die. This would all happen automatically in the simulation. But say your cookie was magic with the words "eat me" on it, you'd hook the "eat" trigger, and add your own game story logic. You'd have the choice to return to the parent logic or not. The former would dispose of the cookie, for example.

As time progressed, their first adventure, *The Pawn* was beginning to take shape. Hugh continued to develop the *System* and the parser, with Ken evolving the Virtual Machine code. Rob shaped the storyline, adding the "*fredding*" and Anita addressed the business problems whilst mysteriously finding bug after bug in the engine.

[Hugh] We never figured out exactly how she did that! [...] In later games, we were organised enough to separate these parts – mostly! But *The Pawn* suffered from being the first game where everything was new. What I described as the *System* was in development at the same time as the game itself. This caused a number of circular problems and it wasn't 100% certain whether a feature would work well enough for a game idea or whether the game idea was even a good one in the first place. Consequently, some parts of the game worked well and others not so much. In several cases, there were parts of the story we would have liked to expand upon and improve, but there was no space at the time to accommodate it.

With her strong links with Sir Clive, and the game running on the QL, Anita thought that her namesake would help to publish *The Pawn* though his software label. Unfortunately, Sinclair Research, due to the troubled production of the new computer ran into financial difficulties. The failure to find a market for their C5 electric car deepened their cash flow woes, and the route to market looked to be closed. As one door closed, another opened and Jeremy "Jez" San [game programmer and entrepreneur who founded Argonaut Software] introduced Anita to the new head of Telecomsoft's Firebird label, Tony Rainbird.

[Hugh] Tony was awesome. He was the kind of businessman that you don't get today.

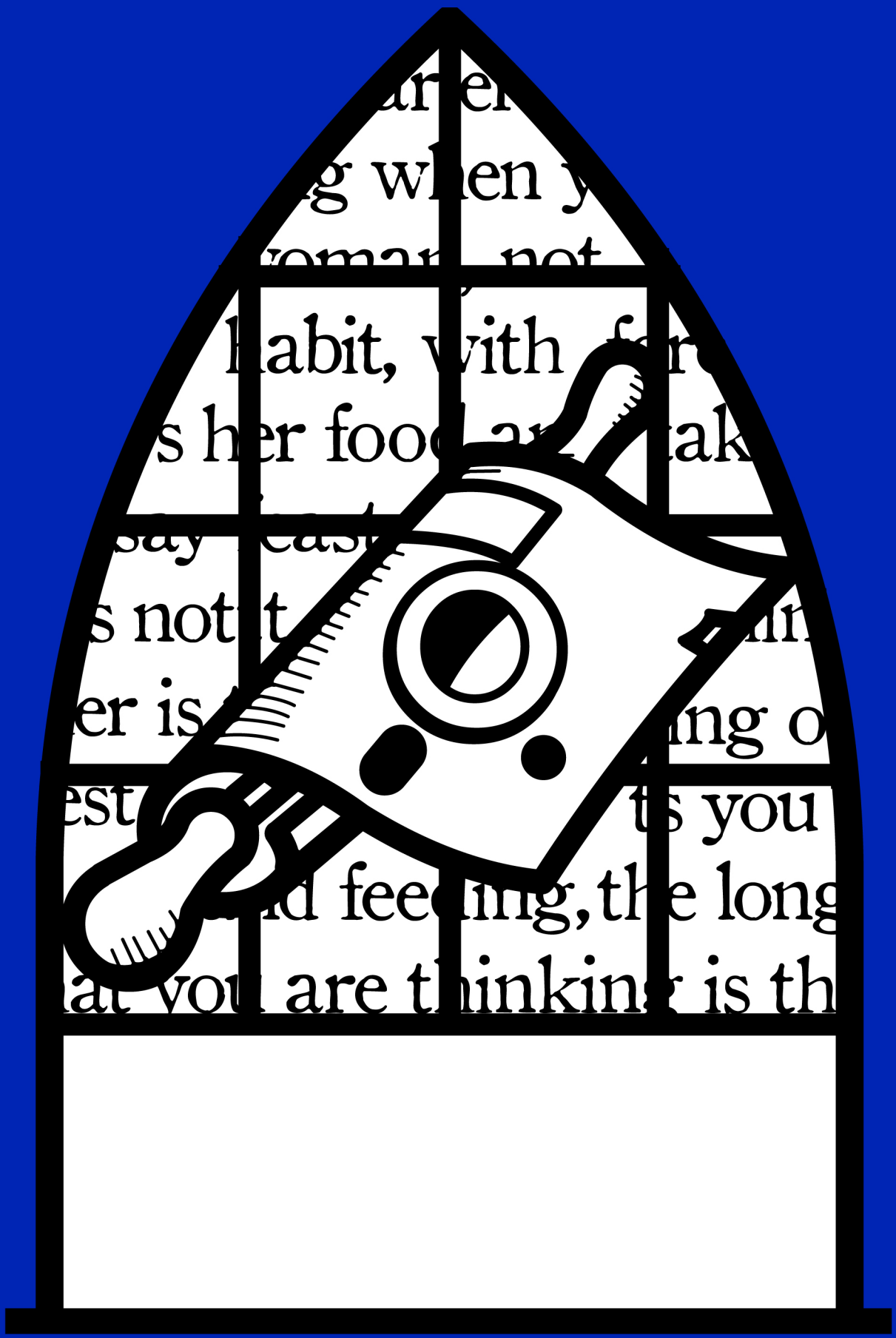
Anita showed Tony a demo of *The Pawn* and he quickly realised the potential of the product. By chance, rather than design, Atari were showcasing its new computer, the ST at the Consumer Electronics Show around the same time [January of 1985]. The ST boasted the same 68000 chipset as the QL, but with 512K of RAM, an industry standard floppy disk drive, and graphics capable of drawing from a palette of 512 colours.

[Anita] [The ST] became our focus. Of course, with the STs came the graphics. That then changed the nature of [our] games [that] originally were going to be text-only adventures and then they became text and graphic adventures. Tony said it had to have graphics, and he introduced me to David [Pringle] who ran a company called Oxford Digital Enterprises that produced music programs. He had in his employ an artist he wasn't using called Geoff Quilley who lived in Cornwall. [...] David relinquished Geoff to me and he did 90% of the graphics in *The Pawn*. We had to design user interfaces, and that became quite fun, I got into that. The way that the graphics scrolled was great fun to do.

[Hugh] I think we would have added graphics anyway, but the ST was also the machine we used to author the graphics. So, in that sense it was vital for graphic production. [...] Anita was convinced, correctly, that we had to have graphics. She even invented some of the image compression methods. The graphics were always meant to be illustrations, rather like you would get in a book. It was natural to assume people would appreciate images within interactive fiction in the same way, although some players expected the graphics to change with the situation, whereas we saw them as situation snapshots.

After being almost instantly forgotten on the QL, the ST version thrust Magnetic Scrolls into the limelight. The castle courtyard or snowman graphics were seen running on every shop demo machine and featured in the pages of every newsstand print magazine. It had the wow factor, the killer app appeal, and brought into sharp focus the jump in technology and capabilities from the 8-bit machines to their new 16-bit overlords. The team went into full brainstorming mode to enhance *The Pawn* for the new platform. Rob jotted his extensive ideas onto reams of A1 sized paper and took it to the office. Anita, Ken and Hugh all mucked in with plot and puzzle ideas.

[Hugh] The engine itself did not change significantly, although by



then of course it had evolved somewhat. By this time, it was possible to selectively include modular parts of the engine that were needed for a particular story. [...] The biggest challenge of the 16-bit machines like the Atari ST and Commodore Amiga was how to make use of their relatively vast memory and faster CPU speeds.

The Pawn, was released in 1986, was set in mythical Kerovnia, a world in turmoil from the power struggle caused by its waning king. You started the game wearing a wristband, unable to break free from Kerovnia, unable to pass its borders still wearing it. It was a game of kidnapped princesses, useless princes, sub-plots, dragons, hobbits, trowels and a character called Jerry Lee Lewis who you bribed with beer.

Anita set the scene to C&VG magazine: "You wake up one morning to find yourself out in the middle of a field of flowers. It's pretty and all, but not exactly what you've come to expect at 8am. You need to remove the bracelet someone has popped on your wrist, but the problem is that it won't come off. In fact, only one person can remove it, so you need to search and find him or her in order to get home".

Despite its impact, *The Pawn* remained a flawed adventure, and felt more like a technology demo than an airtight, highly polished adventure. Some puzzles seemed obscure, created solely to make use of the parser, and the characters certainly felt as they'd been dropped in for comedic effect, rather than knitted into the fabric of the plot. The Scrolls team were attempting to find their feet, both as writers and technology creators.

[Hugh] Back in those days, you had to develop your own technology and algorithms for everything. We invented our own text and image compression methods, for example, as well as developing our own build tools for the whole system. [...] To some extent, also both the story and the system were being developed at the same time for *The Pawn*. It's easier to understand how weird parser-style puzzles might get put in. Anita was always asking if the parser could handle such-and-such. We'd have a go at making that work. And, of course, if it could be done then we had to have a bit of the game that used it!

It was a monster hit for Rainbird and received rave reviews across the press, with Keith Campbell commenting specifically on how the fledgling company had set a new standard in adventure creation. "No longer will it be viable to knock out a quick cheap-and-nasty [adventure] with *The Quill*", he noted, "*The Pawn* is so good that to compare it in the same category as many of the adventures on the market [would be unfair]."

[Sinclair] If it hadn't been for Geoff who knows how successful we'd have been really, because the first thing people saw of the game was the graphics.

[Hugh] It's true that the graphic illustrations made the games special and differentiated Scrolls from other companies. We endeavoured to make at least a few really high-quality pictures for each game. This was not especially easy as making images look realistic with only 16 colours and chronically low resolutions was a huge challenge. Artists did not paint strokes with a brush like in *Photoshop* today, instead they clicked on individual pixels one at a time. It needed that much attention to get a masterpiece, and some of the results still look fantastic today. Geoff Quilley did a superb job!

After their Telecomsoft partnership announcement at the Winter Consumer Electronics Show [CES] in Las Vegas in January of 1986, Anita Sinclair and Tony Rainbird returned stateside later in the year to promote *The Pawn* at the summer CES event in Chicago. Atari, under the new ownership of Commodore founder Jack Tramiel was treading the boards, heavily promoting its new ST range of computers, keen to maintain its momentum and initial healthy sales.

Anita recalls that the Scrolls booth was ten-deep with people constantly trying to get hands-on time with the game. Such an interest didn't go unnoticed by the Commodore owner, and Tramiel tried to gain an audience with the astute Sinclair, only to be told he'd have to wait his turn.

[Hugh] I wasn't there for that, but I understand it was actually more of a misunderstanding. [...] [At CES] I used to hang back and observe what people typed into the machine. This gave me ideas of that people wanted to do. For example, *The Pawn* actually has two parsers. A structured one and another unstructured one. If you communicate with a game character, you'll get the unstructured one. You can get some surprisingly weird responses - some even accurate! At shows, I'd mainly test this on people and see their reactions. The unstructured parser was removed for later games to save space.

The Pawn received a staggering ten releases across the home computers available on the market at the time. It seemed a strange decision to force a quart into a pint pot, having to create a text-only version for the Spectrum or Atari XE for example, and not focusing resources to push the 16-bit versions to a technical pinnacle. The 8-bit versions generally went under the radar. The sumptuous narrative and parser was retained, but they were text-only affairs for most systems, or restricted to disk-based releases for those machines that could support them. It was their 16-bit counterparts that stole the limelight, and for text adventures, we'd moved into the era where graphics sold games.

The accolades flooded in, with *The Pawn* taking home the C&VG Golden Joystick Award for Best Adventure, and Adventure Game of the Year from Crash magazine. The game proved successful overseas too, winning The Golden Tilt award from the leading French games magazine, and awards from the German magazines Happy Computer and 64'er.

[Hugh] [My favourite parts] are the surreal ideas such as the overall concept that you are not actually the real player. Your appearance in *The Pawn's* Kerovnia totally upsets what was otherwise supposed to happen. I haven't seen that idea developed elsewhere. [...] On the downside, *The Pawn* was always meant to be surreal and this didn't work for some people. Emily Short describes the game as, "not good: not well designed, not well paced, not well hinted, not well written, not even well punctuated."

Regardless, it was the poster boy, the killer app, and delivered Infocom-beating technology – matching their sophisticated parser but adding a larger game world and jaw-dropping graphics. It was a true British game changer. Anita's business acumen and drive kept *The Pawn* in the news, and her influence over journalists meant that magazine after magazine were queuing up to dedicate plenty of column inches to the game, and to the lady herself. Digital Antiquarian Jimmy Maher commented "But graphics were just one of *The Pawn's* not-so-secret weapons, the other being the potent comeliness of Ms. Anita Sinclair. The British press, who had the most regular access to Anita and her charms, were the most smitten." It was the obvious move, but also a clever one, that Anita became the face of Magnetic Scrolls. She was a rare commodity, a supremely articulate operator, who handled the demands of fronting the industry's next hot property with aplomb.

[Hugh] Totally! Anita was, and is of course, a brilliant businesswoman. She has a natural instinct to "know" what will work and what won't. It was much rarer, then, to have a woman head up a technology company, and this obviously really helped. [...] We were all more than happy to let Anita to all the talking, and don't forget, the press was always up to their tricks as well and she was good at outmanoeuvring them.

Their success powered Magnetic Scrolls forward and enabled the team to invest in new hardware technology to support their industry leading software. In collaboration with British Telecom, they purchased a DEC Microvax which centralised development as everyone could work from a terminal connected to a single server.

[Hugh] Now we could be organised and use automated source code control and versioning. That's the system needed to ensure two people don't edit the same file at the same time. Nowadays this is standard, but back then, if you couldn't centrally automate it, you had a manual system like having little paper cards in a box - like borrowing a book from a library or worse you pass around a "teddy bear" whereby only the holder of the bear can edit anything! You can imagine how annoying and inefficient this would be.

Their next title, *The Guild of Thieves* was well into production during *The Pawn's* public relations campaign, and the team had started to expand to accommodate the development of multiple games. Anita had around 12 full-time staff, split between programming and art, with 4 freelance writers at her disposal. Though the technology was beginning to mature, it was always pushed forward and went hand-in-hand with the storytelling.

[Hugh] It was never just about writing adventures. As Rob says, we wanted to experiment with every new game. Only *The Pawn* and *The Guild* were set in Kerovnia, all the others were different. Each new idea was a challenge. Some games came out better than others because we always wanted something new and untested. It would have been way too boring to just churn out *Guild 2*, *Guild 3*, etc. We might as well have given up and all got jobs at Tesco.

Guild's plot and puzzles were astonishingly written in a single afternoon in the pub by Rob. He'd phoned Ken and Anita about doing another scenario and argued to deviate from their original plan for another fantasy game.

[Hugh] No, we didn't want another fantasy game at the time and definitely not a sequel. We always wanted something new, a risky strategy, but necessary if we were to push the frontiers and make these games more mature and have wider appeal.

They compromised on another story set in Kervonia, about a wannabe amateur burglar and his attempts to become a fully-fledged member of the professional Guild. Whereas *The Pawn* showcased its surrealism, and convoluted solutions, *Guild* was more straight-forward, and old-school adventure with its emphasis on logical puzzles, treasure hunting and simpler character interaction.

[Hugh] Games take ages to write. It took us about a year to write most games. Although, it's true you could have the main idea and sketch out a sort of outline concept in a day. But, that's the easy bit and, for sure, it will change! [...] Actually the *Guild's* "collect the treasures" concept turned out to be hugely popular. *The Guild* was probably the most successful game Scrolls made. Also, the open-ended nature of *Guild*, where you could simultaneously tackle several puzzle frontiers was brilliant.

Jinxter followed – Scrolls' answer to Infocom's *Enchanter*, with Anita's sister Georgina providing some of the back story and text for the included novella.

[Hugh] By *Jinxter*, we wanted to try developing games in parallel and it seems sensible to try to expand the writing. The idea didn't work out too well, but maybe that was just bad luck. Not everything goes smoothly. [...] You had to have the primary game idea, then the main outline as the game "backbone". Elements of the main story could be broken down into steps you have to do, and then those actions could be turned into puzzles. When it came to the puzzles, ideas were banded about from everyone. It was important to technically validate the puzzle concepts, or you'd get stuff that couldn't be actually implemented.

[Sinclair] None of us could really write and Rob was off at University. We needed people that could write - My sister is a historian, and she could write. We used her, Michael Bywater and various people that could write. They didn't necessarily construct stories, but they could turn programmer ideas into the written word. She worked on an early version of *Jinxter* and the first draft of the novella that came with.

Jinxter was well named - its protracted development and a rumoured disagreement between Anita and Georgina led to much of the early

[Above] Rainbird's first release, *Starglider*, debuted the publisher's iconic blue boxes alongside Magnetic Scrolls' *The Pawn*. Every subsequent adventure used the same lavish packaging until the release of *Fish!* and their Virgin-Mastertronic title *Wonderland*.

work being rewritten. Georgina's drafts were polished and reworked by Bywater who, with a limited amount of time before the game's expected release, had to crowbar his changes into pre-existing data structures and puzzles.

[Hugh] Yes it was [rewritten]. All games had a working title which would be changed once the game was finished. *Jinxter* was developed as *Green Magic*, for example and *Corruption* was called *Assassin*. The working title reflected the genre of the game. So, *Assassin* was a thriller and *Green Magic* meant it was fantasy. The early version of *Green Magic* didn't gel. Michael helped add a certain style to the game and, of course, *Jinxter's* outrageous sense of humour!

As *Jinxter* was in the final throes of development, Rob Steggles got his wish to move away from fantasy worlds, and returned to write *Corruption*, a story of dodgy finances and underhand dealings in the City of London. He'd only contacted Anita and Ken for a job reference but was asked to come and work full-time in their new offices, located near London Bridge. "I thought it was now time to branch out into a different genre and I settled on the idea of a thriller set in the City of London [and] Ken and Anita liked the idea" he told L'avventura è l'avventura.

[Hugh] We wanted to try our hand at something different. Rob and I basically wrote *Corruption* single handed in about 9 months, except for the artwork of course. We both would have really liked to expand many parts of the game, but it was always a battle to keep it within the memory constraints. Rob wanted to try his hand at programming, he wrote all the code to drive the office elevator. It worked, but sadly we had to remove it for the final version, just to fit the game in memory. It was the only non-vital piece of game logic. Sorry Rob! When we remaster the game, we'd like to add quite a few new things!

Steers took over full programming duties for *Corruption*, even though Steggles contributed those routines that were eventually dropped. With *Corruption*, it was gameplay and narration and the addition of intelligent non-playable characters that conveyed specific and significant changes in the engine.

[Hugh] I added some system changes to *Corruption* to drive game characters. Each one had a goal-solving script which drove their behaviour, it was quite advanced for the time. The characters could actually solve problems themselves, which was a kind-of AI. I remember several times; the characters would outsmart us. For example, they'd find a shortcut to get from A to B faster than they should or perhaps start going for a stroll in parts of the game they weren't supposed to be. Then they'd go crazy sometimes too!

Magazines reported that *Corruption* was heavily modified to have a better control over the passage of time to support the non-playable characters, and have every action incur a cost measured in minutes and seconds. These changes played a significant part in the game's puzzles and story.

[Hugh] Since each character had their own behavioural AI goals, it was important to map out the interactions of who was where and when at any given time. Of course, this was not a constant because the player could interfere. For example, if a character was looking for, say a document, the player could drop this document in their path or even stop them and hand it over. This meant that perhaps a character could move on to their next goal earlier and be somewhere sooner than they would otherwise have been. Sometimes this would cause conflicts with the story that would give us a headache.

Since *The Pawn*, Scrolls had followed Infocom's lead by including "feelies" with its games. They became ever more elaborate as more titles were released, and *Corruption* also had one of the best "feelies" of the genre - an audio cassette tape with an answerphone message that held clues to the game.

[Hugh] With any game, there's a lot of scope to what could be included as feelies. Cost was a major factor, so it was Ken and Anita that used to explore what we might be able to put in the box as feelies on a tight budget. Adding a working cassette tape was quite expensive, so this was quite an achievement. [...] I always liked what they came up with. Beer mats in Jinxter, cassette tape in Corruption and a credit card in Guild. That's in addition to the usual box content, the manual, posters, booklets, etc.

ANITA'S ADVENTURE

Visit any website that features *Corruption*, including the Magnetic Scrolls memorial, and you'll more than likely find the widely held misconception that the game's working title was *Upon Westminster Bridge*.

In fact, *Upon Westminster Bridge* was an Anita Sinclair murder mystery adventure, with multiple playable characters and super-intelligent non-playable characters that had a brand new scripting language to control them.

[Hugh] *Upon Westminster Bridge* was not at all connected to *Corruption*. In fact, it was the working title for a new game and would probably have been renamed if finished. Yes, unfinished copies have been recovered from the backup tapes!

Corruption was a hugely underrated adventure. Jimmy Maher summed up the game by saying that "Steggles captured the zeitgeist in a bottle. This being the height of Margaret Thatcher's remade and remodelled, hyper-capitalistic Britain." The critics agreed, and it went on to win the prestigious Personal Computer World (PCW) Game of the Year award. But there was a state of diminishing returns for text adventure games. *Corruption* had a particularly low-profile launch, and the industry had passed the peak in adventure sales. Some even speculated that *The Pawn* had seen the peak of the market.

Fish! [**see Issue 03**] was Magnetic Scrolls' final release with Rainbird, and fell victim to the disintegration of the publishing relationship between the two entities. BT had put Telecomsoft up for sale, and Tony Rainbird decided that his future was elsewhere, leaving in the run up to the announcement to pursues his own business interests. *Fish!* was a completely surreal game, and arguably one of Magnetic Scrolls' best, with its humour aimed at a maturing audience.

[Hugh] Firstly, the design and writing of *Fish!* was the joint efforts of John Molloy, Pete Kemp, Phil South and Rob Steggles. It was developed in parallel with *Corruption*. That meant I was not available to add new *System* features for *Fish!* So, the thing that impressed me the most was that *Fish!* was made without direct input from the original coding team. Definitely the *Fish!* team did a fantastic job.

It was poorly marketed, and came packaged in a small box, much smaller than the lavish and glossy sized cardboard that was an expected experience with previous Rainbird releases. Its limited commercial success also could be put down to its somewhat marmite approach to humour.

[Hugh] It's rather difficult to say, even with hindsight. The trouble was commercial text adventures were a much smaller part of [Rainbird's] market. [...] Not all games sell as well unfortunately, and that doesn't always reflect relative quality. *Fish!* was a great game, it had its own sense of humour - as did all the games. For sure, it could have been promoted better, but that's true of anything. [...] We're currently remastering *Fish!* We've got most of it working already. We'd like to fix some minor gameplay issues and hopefully add some new features too.

In 1988 Microprose purchased Telecomsoft, and the Silverbird, Firebird and Rainbird labels - including the rights to Magnetic Scrolls games. It formed one of the biggest UK based software companies and should have heralded a new era of investment and promotion for

Anita, Ken, Hugh and the team. For a year or so Scrolls disappeared from the press radar with Ken retiring *ELTHAM* and working on brand new technology for their next game. Anita and Microprose were awkward bedfellows. The unhappy marriage came to an end in January 1990 when Anita signed a deal with Nick Alexander of Virgin/Mastertronic.

With Virgin, Magnetic Scrolls unveiled *Wonderland* [Based on Lewis Carroll's Alice's Adventures in Wonderland novel] in July of the same year, with a story by freelancer David Bishop running on a brand-new cross-platform, graphical multi-window interface called *Magnetic Windows*.

[Hugh] Wonderland had that surreal mixture of fantasy that we liked. It's also one of those books where people want to imagine being there and find out what might have happened if they'd chosen some other action than *Alice*. Also, it was out of copyright.

Magnetic Windows offered an iconographic representation of available compass movements, objects that could be interacted with, the player's inventory, as well as a story windows that operated as a traditional text adventure parser. Its innovative features were powerful; it had a map that drew itself as you explored and the system also added animation support for the graphics. Each window could be resized, open, closed, and moved around on screen.
It was a huge undertaking for such a small company with limited resources and was a root and branch attempt to change the way that people played text adventure games. It kept the complexity of the more traditional experience, with the usability of a GUI. It was the direct opposite approach of a new style of interface that had begun to edge traditional adventure games from the market - the point and click revolution under LucasArts.

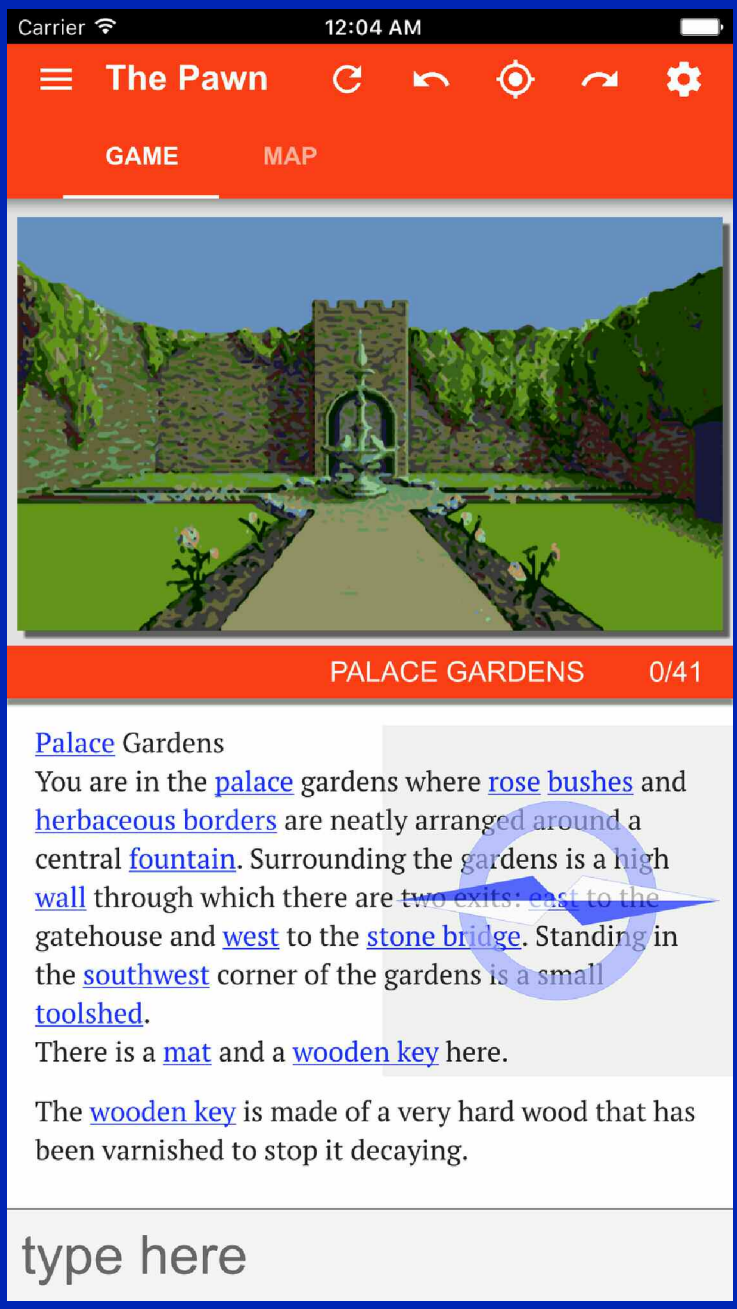
[Hugh] [...] During the late 80's it was obvious to us that we needed a graphical environment. We based *Magnetic Windows* on *X-Windows*, it was a colour system, although in practise all 16-colours had to be used for the picture, leaving just black and white for the UI. [...] [Point and Click] was discussed, but the feeling was that point and click was a totally different kind of game. Specifically, it lacked narrative and depth. It's not that we thought point and click was going to be unsuccessful, it's just that we didn't want to go down that route.

[Anita] We almost certainly made the wrong call with that, but our publishers wanted to do it, and David Bishop was on the ball and the game looked really good, with animations, and was easier to play from a point-and-click perspective but without sacrificing any of the text adventure game play. It was the wrong call, but a shame.

Wonderland was the last adventure from Magnetic Scrolls, its costly development and insufficient sales virtually bankrupting the company. Despite the argument that *Magnetic Windows* was attempting to reinvent the wheel, the adventure itself was very well reviewed in the press. Keith Campbell, a stalwart Magnetic Scrolls [and Anita] fan, proclaimed that the game "had all the fine detail that has become the hallmark of Magnetic Scrolls", and that the game was "nigh on unbeatable at the moment, I cannot see it being bettered for at least a couple of years."

They used the *Magnetic Windows* system to revamp and repackaged their back catalogue, releasing *The Magnetic Scrolls Collection* in 1991. Unfortunately, the tide had turned, and Virgin ended the publishing relationship due to poor sales.

[Hugh] At the time we all knew text adventures had had its day, or so we thought. Infocom sadly packed up and we were next. To some extent the belief that point-and-click was the new thing was a self-fulfilling prophecy. The vacuum of professional interactive fiction was filled by the hobbyist one. This proved people still wanted to play it. This all happened just before the internet went mainstream. The internet might have saved commercial interactive fiction because you didn't need a publisher. Working with a publisher meant you had to pander to their concerns too. They didn't like experimental works, they wanted mainstream cuddly stuff. Oh yeah, and they took most of the revenue! Having said that, IF needed to change and evolve. Even with the internet, selling games made from the same technology wouldn't have worked. They needed to be bigger,



[Above] The iconic *The Pawn* - remastered and improved by Stand Games to work on modern PCs and mobiles with a vastly improved user interface.

MAIL ORDER MYTHOLOGY

Tony Rainbird left to form Official Secrets, a mail order adventure company that commissioned Magnetic Scrolls to produce *Myth*.

Myth was a limited-edition, lavish mini-adventure [with sparse graphics] written by Paul Findlay that explored Ancient Greek mythology through the eyes of Poseidon.

friendlier and derive their entertainment from things other than puzzles.

It was the end of an era but Scrolls weren't alone. Level 9 and Infocom had both folded, and Legend Entertainment setup by former Infocom staff had attempted their own [if somewhat basic] reimagining of a GUI based system [similar to *Magnetic Windows*] for adventures *Timequest* and *Gateway*. By 1993 they'd deserted text adventures too. One final hurrah for Ken Gordon was producing an RPG called *Legacy* for Microprose, but this was developed after Scrolls was wound-up.

[Hugh] Well, if I could go back in time, I would tell everyone that *Windows 3.0* will suddenly appear before *Magnetic Windows* was property finished. [...] I still would have not chosen to develop point-and-click, but instead of developing *Magnetic Windows,* build a new authoring system that makes games 10 times easier to write and targets 1MB+ systems. Then I would make 10 games per year instead of one! But more seriously, what is interesting is that "adventure games" did not really die after point-and-click nor even with the explosion of 3D games that followed. There always was and still is a non-trivial market for interactive fiction. The fact that people still read books is testament to a demand for well written narrative.

After the closure in 1991, the Magnetic Scrolls team disappeared from the videogame limelight for almost three decades. Several fan sites emerged with the propagation of the internet, and the odd interview with some of the mainstays, such as Rob Steggles, Michael Bywater and Phil South appeared - but the original founders remained elusive.

In 2018, a new website and publisher, Strand Games, emerged posting information regarding Magnetic Scrolls' history and the promise of remastered games. Strand's About Us page announced that Hugh Steers, along with Stefan Meier, the curator of if-legends.org, were intending to preserve the history and masterpieces of the British developer with the support of several members of the original team including Anita Sinclair, Ken Gordon, Rob Steggles and Servan Keondjian.

[Hugh] I wanted to get back into interactive fiction. As a stroke of luck, we managed to get our hands on the original Scrolls backup tapes. We thought, as part of a bigger endeavour we'd also like to restore the games from the original source code, make them work on modern computers and mobile devices as well as publish the source code.

Though the Magnetic Scrolls brand was registered as a domain name, the new company moniker, Strand Games, reflected the objectives of the new enterprise, being the "strands" of "History, Technology and Games."

[Hugh] History is bringing back the retro games, Technology is work we currently doing on a new IF gaming system and Games, refers to our republishing of the original games and new game material we hope to develop and publish. It would not be appropriate to call it Magnetic Scrolls as people would be misled to believe we will be making sequels to the original games. That's not the case, although we haven't ruled that out.

After several blog posts that discussed the games and technology, 9th May 2018 saw the release of the first public beta of Strand's modern adventure platform, and a reimaging of Magnetic Scrolls' first masterpiece, *The Pawn*.

[Hugh] We managed to remaster *The Pawn* without the source code and [that] was quite horrible. However, I'm pleased to announce that since recovering the source code, we've now published an updated version of *The Pawn* that is built from the original code and the hacks have gone! [...] There was a lot of positive feedback, but we were also surprised by expectations for the new UI, so a lot of features were included. You can change fonts and colour themes, there's a navigation compass, a dynamic map, touch links, inventory sidebar, input word suggestion, sound effects and many more. We're adding new features to the UI all the time.

[Anita] I'm terribly pleased [about the re-releases]. It is nice that the games are still there, and people still remember them. They were a little slice of life, and a good slice of life. [The new games] are amazing. It's better to play the original, but given 99.99% percent can't do that, given the fact you can now get one from the Appstore and play it is excellent.

The re-emergence of one of Britain's most illustrious names has so far gone under the radar. Mainstream game sites and retro magazines such as Retro Gamer have hardly given Strand a mention, and as of yet the company doesn't have a social media presence.

[Hugh] [...] That's OK right now. It's great that we're remastering the originals, but one of the main objectives of Strand is to develop new material. At present it looks from outside like we're only restoring the retro games. But that's not all we are doing. For example, we've learnt a lot from doing the remasters of what people expect from the UI. And a revised version of this will become the basis of our new platform, so it will debut with a well-tested feature set. Hopefully, at some point, we'll have something new to publish. And then it will time to shout about it.

Releasing text adventure games into the mobile sphere has proven difficult for many developers, especially given that the platform does not lend itself well to typed instructions. Several other developers have moved towards Interactive-Fiction with selective and branching narratives rather than pure text adventure, given the difficulty in designing an efficient UI that enables text entry.

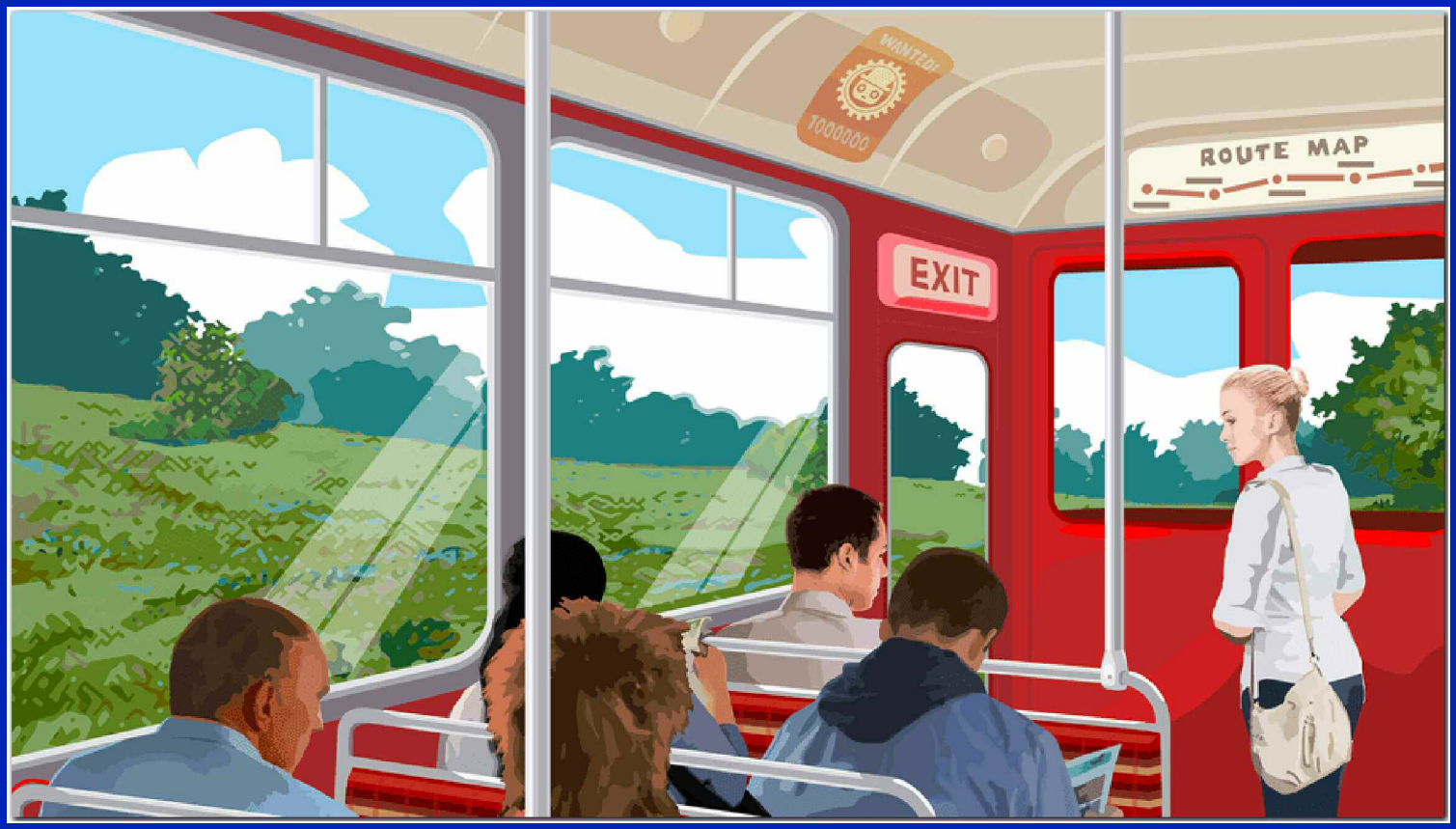
[Hugh] The challenge for narrative based interactive fiction is to make the human interface as easy to use as possible. Text entry is often a bind on desktops as well as mobile, where it is especially problematic. We have tried to alleviate difficulties with text entry as much as possible; elements of the main text are clickable and these "links" perform sensible operations like "look", "get" etc. An on-screen navigation compass and a map make it easy to navigate. And when you do use the text entry box, there is context sensitive word suggestion and re-editing of previous commands. Our plan is to try making the bulk of commands touch based but provide free-form text entry for the rest.

The games are being remastered from the original Scrolls code – with the team editing the original *ELTHAM* code in various ways to make gameplay improvements and fix bugs. For example, the ending in *Jinxter* was famously cruel and has been fixed for the remaster, and thanks to Stefan, there's an exciting addition of multiple game endings. The *System* has been replaced by a new framework called *Braham* that has an independent UI and a communication component that links the game engine with the front-end; the *Interactive Fiction Interface*, or *IFI* for short.

[Hugh] With this approach, the front end can be connected to different back-end game engines. So, one such "back-end" is the Magnetic interpreter for Scrolls games. So, we not using any new tech to reanimate these, and new games will manifest via a different back-end engine altogether. Future game development will not proceed as an extension of the old Scrolls *System* at all.

Anyone who has been following the regular blog posts on the Strand website will have a glimpse into the herculean effort it has been for the team to recover any semblance of data from the age-old Magnetic Scrolls backups – most of which that have been stored on decades old floppy disks and Digital Audio Tapes [DAT]. There's a mix of frustration and fun in the process, and Hugh has gone to great lengths to preserve the data, even calling on the help of DEC [the original manufacturer of Scroll's mainframe system] specialist Rob Jarratt.

[Hugh] This is a lesson for people who archive material onto tape; after many years tapes actually absorb water and become unreadable. However, it turns out if you're lucky you can temporarily restore the readability by baking the tapes in an oven! For those interested, you put it on about 50 degrees for 7-8 hours and slow cook it! After that, if you're really lucky you just might be able to read the tape, but usually you'll also have to thread it into an opened-up tape reader as it will generally need constant cleaning as the tape sheds oxide onto the



[Above] As part of Strand's history project they are working on a restoration of *Jinxter*, entitled *Jinxter Revived*. One of the few illustrations that the team didn't feel lived up to expectation was the bus image from the original game [**top**]. For the remaster, [bottom] they have remade the image with a more modern context, that is scalable as needed for modern high-resolution devices.

mechanism as well. Rob Jarratt did an excellent job of restoring the tapes for us.

It highlights the wider implications of lost history as more and more people and organisations become aware that there hasn't really been a strategy for preserving and archiving information from the computing era. Strand are fulfilling an important role in digital archiving, sometimes finding that paper backups of source code are in better condition than digital or analogue ones.

[Hugh] I see this whole problem as how to archive what I call "modern computer history". It's the stuff still in people's heads but not yet considered museum material. It's one reason we wanted to restore the original tapes. [...] The job isn't over! We've recovered nearly all the original game sources. We'd like to try recovering the rest of the tapes. Some will have junk on them, but I'm also hoping to find source code to various ports, like the Z80 and 6502 versions. There was also an emulator that ran backwards, I'd like to find that. And of course, there might be versions of the original artwork too.

Not content with games for modern platforms, the arrival of the Spectrum Next has seen a resurgence of interest in Magnetic Scroll's back catalogue. Several previews builds have appeared online and at various retro shows of Next versions of Strand's games, showcasing a mixture of Spectrum text and Atari ST graphics. Programmer Stefan Bylund has been brought in by the team to convert the graphics from the ST and Amiga versions to run natively on the Next, but Hugh plays down any prospect of a physical release.

[Hugh] It's an exciting project because it remakes the Spectrum but adds new features as well. Apparently, there might be some new graphics modes and it will be interesting to see if we can do anything to the original and remastered images for the ZXN. [...] In theory the remaster binaries should work with the Magnetic interpreter. Nevertheless, a few changes are necessary, but we're hoping to get the remastered editions on the ZXN, as they have many bug fixes as well as several game improvements.

The Guild of Thieves Remastered was released on 18th December 2018 - the first Strand game completely rewritten using the original source code. The original Scrolls build system was written in C, so has been recompiled to run on modern platforms [including Windows, Linux, Mac and Android and iOS devices]. The flexibility of using such a high-level and future-proof language means that Strand can feed the original source code into these tools and rebuild the binaries.

[Hugh] In theory at least!

For *The Guild Of Thieves* remaster, significant bugs from the original game have been squashed, and several gameplay tweaks have been made to make the game more accessible and less hostile to the novice player.

[Hugh] For example a long-standing bug in *The Guild* meant you didn't have to win the "rat race". Now you do! Then, in the original, when you finally got the lute, you'd trip over and break it - without even a clue! Not anymore, insanely cruel stuff like that has been reworked. [...] Which leads me to two ways we can improve gameplay. Sometimes we've slightly edited the text messages to add missing subtle clues, and sometimes we've changed the logic so events make more sense. We can also make the text look nicer using colour and bold/italic highlighting.

Strand have also added the popular feature found in many other retro-inspired remakes with the ability to switch from "modern" mode back to "classic" mode. This toggles bugs, and the appearance of graphics, from the upscaled and resampled illustrations that Strand have been experimenting with, to the original classic pixel graphics from the originals. In addition, the UI can suggest words within certain locations and situations – great for the novice, but an annoyance to some purists.

[Hugh] Word suggestions and fixing confusing locations have all been well received. The suggestions never give away information or spoilers, which is good. The upscaling was necessary to bring the pictures to what is still considered small! The originals were only 255 pixels wide and the upscaled ones are now 1024. [...] Some people actually prefer the original pixelated images. Although you can disable our upscaling, the hardware will scale itself using interpolation so it will try to smooth the blocks. We recently added a feature to disable this in classic mode to try to bring the original pixelated images to the screen as faithfully as possible.

STRAND SOURCE CODE

Eventually all of the original source, including fixes and any other recovered assets will be made open source. Currently Strand have made *Brahman*, with a small demonstration project available via gitlab.

See **gitlab.com/strandgames**

Adventure writing is going through a renaissance, with the release of the entire Gilsoft and Infinitive Imagination library of utilities, and new tools such as Chris Ainsley's *Adventuron* going from strength to strength. One of the driving factors of these technologies has been the free of charge, public access to them, and Strand are keen to venture into this space in the future.

[Hugh] We're working on a high-level authoring tool that will compile into *Knowledge Language*. The *KL* runtime will constitute a new back-end game engine that will run new material. [...] The *KL* system and new authoring tools will all be made freely available when ready. Anyone will be able to use these to make and publish their own games. [...] People will definitely not want to use the old Scrolls authoring system, unless they are either totally mad or want to be driven totally mad, I can assure you!

It's so refreshing to see that the majority of the original Magnetic Scrolls team are behind the project. It's testament to Ken, Anita and Hugh that the team share such strong bonds. Two of the most enthusiastic members of Scrolls have been Ken Gordon and Rob Steggles, who are both looking to get involved in the future.

[Hugh] I'm going to force Ken to fix his own bugs from 35 years ago and rename our new authoring tool *Fred24*, give it to Rob Steggles and watch him scream! [...] But seriously both Ken and Rob are enthusiastic with the Strand project. Rob has offered to author perhaps a sequel to *The Guild* using the new technology when it's available.

Since his Classic Adventurer interview [**see issue 03**] in 2018, Phil South has hinted of a return to adventure authoring, and is rumoured to have started writing a sequel to *Fish!*

[Hugh] We are indeed working with Phil. I'm supposed to get my finger out and develop, at least a beta of the new authoring system for him. He has some great ideas for *Fish2!* Let's hope we can make this a reality.

And as for Anita's hint that they all may one day reunite for another game?

[Sinclair] I wonder whether Hugh, Ken and I should lock ourselves in room and come up with something new .....

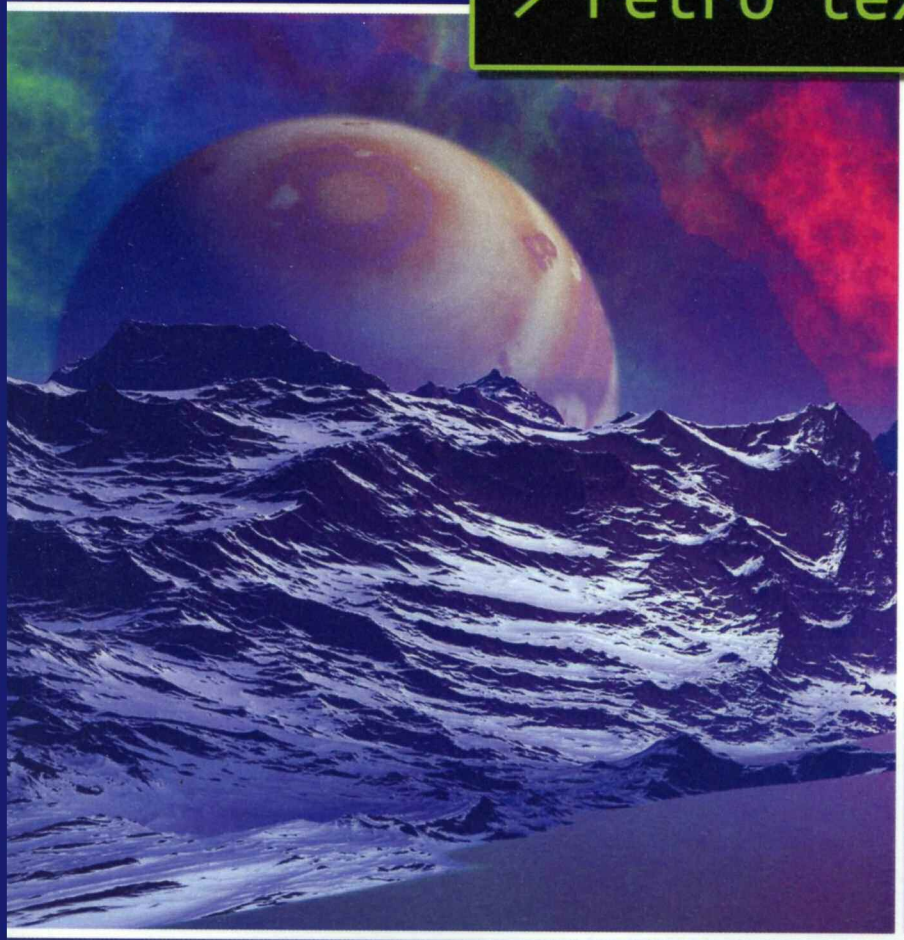
Hugh is very positive.

[Hugh] Anita supports the project and helped us with artwork for *Jinxter*. It remains to be seen whether we can tempt her back into working on new material. Of course, if she starts talking about writing the *Willy Affair* then she'll just have to go. I'm sorry, but that's just how it is!

Author: Huw Collingbourne
Publisher: Bitwise Books
RRP: £10.10
Website: http://bitwisebooks.com

THE LITTLE BOOK OF ADVENTURE GAME PROGRAMMING

> retro text adventures



WRITE INTERACTIVE FICTION
CREATE OBJECT HIERARCHIES
MOVE PLAYER AROUND A MAP
PUT OBJECTS IN CONTAINERS
TAKE & DROP TREASURES
SAVE & LOAD GAMES
LISTS & DICTIONARIES
OVERRIDDEN METHODS
C# PROJECTS SUPPLIED
EXAMPLES IN JAVA & RUBY

BONUS: DOWNLOAD THE SOURCE CODE
DETAILS INSIDE

THE LOST ARTS OF THE IMPLEMENTERS IN C# AND OTHER LANGUAGES

bitwise books

HUW COLLINGBOURNE

THE LITTLE BOOK OF ADVENTURE GAME PROGRAMMING - RETRO TEXT ADVENTURES

With a deluge of adventure creation tools and engines to choose from, The Golden Wombat of Destiny author **Huw Collingbourne** makes a case for writing your own in his new book The Little Book Of Adventure Game Programming.

The Quill, The Professional Adventure Writer, The Graphic Adventure Creator, INFORM or *DAAD* – these are just a few of the adventure creation tools available to any modern wannabe adventure author wishing to write their own classic adventure game. These tools are hugely popular because they take the heavy lifting from creating games, and remove the need for a high level of programming proficiency by providing a tried and tested framework, usually in fast assembler or machine code, or the modern compiled equivalent. It means that authors can concentrate on writing a compelling narrative and puzzles by using the tool's straight-forward and simpler built-in scripting languages.

In the search for more power and flexibility, several authors have implemented their own adventure engines, not just to add their own features and capabilites, but as a way of learning how to program [see Line by Line Labyrinths in Issue 07]. One of those authors is Huw Collingbourne, who coded his own adventure engine for the cult 80s adventure *The Golden Wombat of Destiny*. Now an established author and tutor, Huw has recently published a brand new book, The Little Book of Adventure Game Programming that distills his programming knowledge into a concise guide for those wishing to take the plunge in writing their own parser.

What inspired you to start writing this book?

[Huw] Writing an adventure game is a great way to learn to program. The first big program I ever wrote, back in the early '80s was an adventure game called 'The Golden Wombat Of Destiny'. Since then I've taught programming in magazine columns, online courses and books and I often give small adventure game samples to inspire people to have a go at writing a game. So many people have contacted me to ask for a more in-depth guide to game-writing that I developed a video-based course which proved to be surprisingly popular. [...] The next logical development was to explain all the principles in a book.

What are your adventuring influences?

[Huw] Mainly Infocom games. Zork, of course, but also their space-exploring game, Starcross and other games such as the very weird Alice-in-Wonderland-Meets-The-Atom-Bomb game, Trinity. I love the ability to explore huge landscapes, try out silly things usually get some kind of meaningful reply. The graphics conjured up in my mind by a text adventure are better than the graphics shown on screen in modern games! It's the same difference between reading a book and watching a film.

How long did it take to write the book?

[Huw] Probably about thirty years. I mean, it took maybe only about four or five months to type out all the words. But the book is based on all the code and tutorials I've written going way back to adventure games I wrote in Object Pascal and Java in my old columns in PC Plus Magazine and Computer Shopper many, many years ago.

THE LITTLE BOOK OF ADVENTURE GAME PROGRAMMING

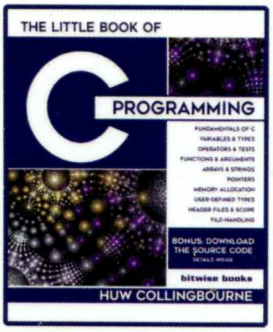
JUST THE STUFF YOU REALLY NEED...

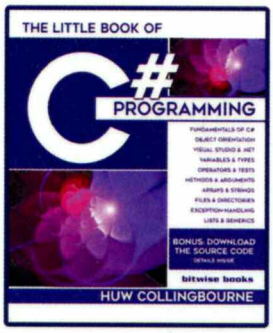
The fastest, easiest way to learn to program object oriented text adventures. This book will teach you the real secrets of adventure game programming. You will learn how to create Rooms and Treasures, how to let the player take and drop objects and how to save and restore games using serialization. The projects (which can be downloaded *free*) are written in C#. Advice is also given on translating to other languages such as Java, Object Pascal and Ruby.

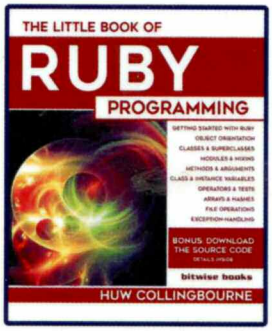
The Little Book Of Adventure Game Programming gives you *just the stuff you really need* to get straight to the heart of writing retro text adventures without all the fluff and padding.

About the author

Huw Collingbourne is the author of the cult text adventure game, *The Golden Wombat Of Destiny*. Huw has been a programmer for more than 30 years and is a well-known online programming instructor. He has written programming columns for numerous computer magazines such as *PC Plus* and *Computer Shopper*. He is the author of programming books on a range of subjects including *The Little Book Of C*, *The Little Book Of C#* and *The Little Book of Ruby*.







bitwise books
www.bitwisebooks.com

ISBN 9781913132088
90000
9 781913 132088

Why do you think that adventures are a good staple for learning programming?

[Huw] Two reasons. 1) A text adventure can be written from start to finish by a single programmer. You don't need teams of people and you don't need lots of graphics libraries or huge 'game frameworks' like Unity. And 2) an adventure game is a perfect project for an object-oriented language. It forces you to get to grips with creating class libraries, overloading and overriding methods, serializing data in order to save and restore games and much more. Even though writing a game may sound like a trivial activity, it really isn't. As my book explains, it requires the programmer to make use of a very broad range of serious and sometimes quite tricky programming techniques.

Who do you hope buys your book, who is it aimed at?

[Huw] Obviously, any programmer who is interested in writing retro text adventures should, I hope, find it of interest. However, even a novice programmer can start work on a game following the guidance in the book. Remember, I was a beginner programmer myself when, all those years ago. [...] I didn't have any help in writing that game and I made a lot of mistakes that could have been avoided if only I'd had access to a book such as the one I've published.

The book does make an assumption that the reader has a basic knowledge of programming, or programming concepts. What would you recommend they read first?

[Huw] Most of the examples in the book are written in C#. For a complete beginner, I'd recommend studying my other book, 'The Little Book Of C# Programming' either before tackling my adventure game book or else, at the same time. [...] On the other hand, a programmer who already has some knowledge of C# would be able to dive right into the adventure game book. All the techniques are fully explained, just not in quite the same detail as in the Little Book Of C#.

Are you planning on any updates or revisions in the future?

[Huw] Let's wait to see if readers request any specific additions. The book is already 'complete' in the sense that it shows how to develop a basic, fully functioning game system with all the appropriate class hierarchies, data-saving features and so on. I've left it up to the reader to figure out which puzzles to add. I have considered the possibility of writing a separate books giving walkthroughs of certain standard types of puzzle – mazes, treasures with 'magical' abilities and so on. Maybe that's something I'll do on YouTube. Though really I think you have to leave something for the individual programmer to figure out. Programming the puzzles is where the fun and creativity come in.

Are you planning on any new adventure writings – perhaps adding more complexity to the concepts found in this book?

[Huw] Again, I would need to hear from readers. I think the concepts explained in the book are all sufficient to write a complex game. There's nothing I've 'kept back'. While I only develop a game with a fairly small number of rooms and treasures in the book, adding on more doesn't add to the difficulty of programming. Similarly extending the class hierarchy by adding on new types of object is not difficult once you've understood the problems associated with inheritance, encapsulation and dealing with generic collections of objects. I've even shown how to write a simple English language parser and while that is only capable of dealing with very simple phrases, the techniques I explain could easily be adapted to deal with more complex sentences.

What about the actual "art" of writing an adventure game – something that's usually missing from these types of book and something that you briefly touch later – the formation of storylines, narratives, what makes a good puzzle, or a bad puzzle, or good and bad design rules? I'd like to see an adventure book that really looks at what makes a good adventure over the structure of the engine.

[Huw] I have considered the possibility of writing a separate books giving walkthroughs of certain standard types of puzzle – mazes, treasures with 'magical' abilities and so on. Maybe that's something I'll do on YouTube. Though really I think you have to leave something for the individual programmer to figure out. Programming the puzzles is where the fun and creativity come in. [...] I am not entirely convinced that this is the sort of thing a book on programming can, or should, teach. A book on programming should stick to the nitty-gritty details of programming technique. I don't really think it is my business to try to teach creativity. If people want to write an adventure game, I'm assuming they want to give free rein to their own ideas. This is where the inner novelist comes to the fore. To any game writer, I'd say: give me something original. Don't slavishly copy other games. Have your own ideas, invent things, surprise the game player. As a writer (I've written for books and magazines for even longer than I've been programming), I detest 'style guides' and 'how to write' books. As a programmer I would be equally suspicious of programming books that told me how I should plot my game.

Have you tried any other adventure engines, such as 8-bit engines Quill/PAWS, INFORM or the recently completed release of DAAD?

[Huw] No, I never have. I like to have full and absolute control over what I program. The fewer engines, libraries and frameworks that intrude upon that, the better. I have, however, written games (some quite small, admittedly) in all kinds of different general-purpose programming languages including C#, Java, Pascal, Object Pascal, ActionScript, Smalltalk and Ruby. All those languages provide all the features I need.

Will you be making a return to writing adventures in the future?

[Huw] Watch this space.

My thanks to the generosity of every contributor, who gave their time to humour me and answer questions they've been asked a thousand times before.

Special thanks to many friends, retro acquaintances, and text adventure geeks, including the usual kindness and help from:

Fergus McNeill
Tim Gilberts
Gerrard Sweeney
Gareth Pitchford

A non-exhaustive list of references and other useful information:

Books, Magazines and Fanzines

Retro Gamer Magazine, Future Publishing
Twilight Inventory, Gareth Pitchford
Spectrum of Adventure, Thomas A. Christie, Extremis Publishing
Adventure Coder, Chris Hestler
Micro Adventurer, Sunshine Publications

Documentaries

GET LAMP, Jason Scott
The Hunt for 'The Hobbit's' Missing Hero, Great Big Story

Websites and Blogs

Mobygames
Spectrum Computing
Lemon64
The Digital Antiquarian
The World of Spectrum
Internet Archive
Amiga Magazine Rack
The Classic Adventures Solution Archive
Stardot Forums
Games That Weren't

Research Papers

There and Back Again: A Case History of Writing The Hobbit, Veronika Megler

Artwork

Sherlock
Lynda Warner and Melbourne House

Sherlock Holmes Crash Magazine Cover
Oli Frey, courtesy of Roger Keen

The Seventh Star
Acornsoft

Alien Research Centre and Hit
Zenobi Software, courtesy of John Wilson

The Pawn (QL Illustration)
Jenny Tylden-Wright

Jinxter and Magnetic Scrolls Logo
Magnetic Scrolls and Rainbird

The Classic Adventurer
Written and designed by Mark James Hardisty

About the author

Mark James Hardisty is from Sheffield. His weekly pilgrimage to Just Micro as a child left him with an indelible love for Gremlin Graphics.

You can find Mark at @hardistymark, where he tweets about games, getting kids coding, The Cannonball Run, and his favourite game - *Elite* on the Acorn Electron.

This work is dedicated to:

My wonderful family – my mum Val, my beautiful wife Helen, and daughters Amelia Rose and Kitty Mae.

Fergus McNeill, a genius, and one of the kindest and humblest people I have had the pleasure of meeting. Thank you for *The Big Sleaze*.