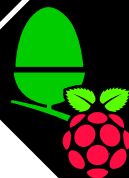


JULY
2020



Vol. 24 N° 6 £5

Archive

The subscription magazine for users of RiscOS:
the Acorn Archimedes and its successors

Newsdesk ... Gavin Smith ... 3
• RiscOS Direct • Impression-X updated • Git for RiscOS
• new games and instant messenger • TCP/IP bounty
• browser updates • date for Wakefield 2021

Southwest show 2020 ... Mark Stephens ... 8

My new Titanium ... Geoffrey Baxendale ... 11

The Cloverleaf project ... Stefan Fröhling ... 14

Building the smallest portable ... Chris Hall ... 16

Making web-books with Bind Gavin Wraith ... 20

XML and direct debits ... Richard Wheeler ... 22

Adding lines to a drawfile ... Chris Hall ... 26

A moral tale, tasting of Pi ... Gavin Wraith ... 29

A glass button, in eight steps
Steve Royd-Marker ... 30

Debugging tools from Spain Bernard Boase ... 32

Network hardware ... Alan Adams ... 34

Thinking ahead... ... Jim Lesurf ... 40

Jim Nagel, 1944–2020 ... the Nagel family ... 43

The company of a kind man ... Mark Moxon ... 48

That's a lot of harddrives ... Bart Nagel ... 50

Your floppies are getting old ... Chris Nagel ... 55

Mac Matters ... Mark Stephens ... 56

Where Acorn leads... ... Paul Beverley ... 58

hint&tips ... QR codes: more than just URLs ... 10

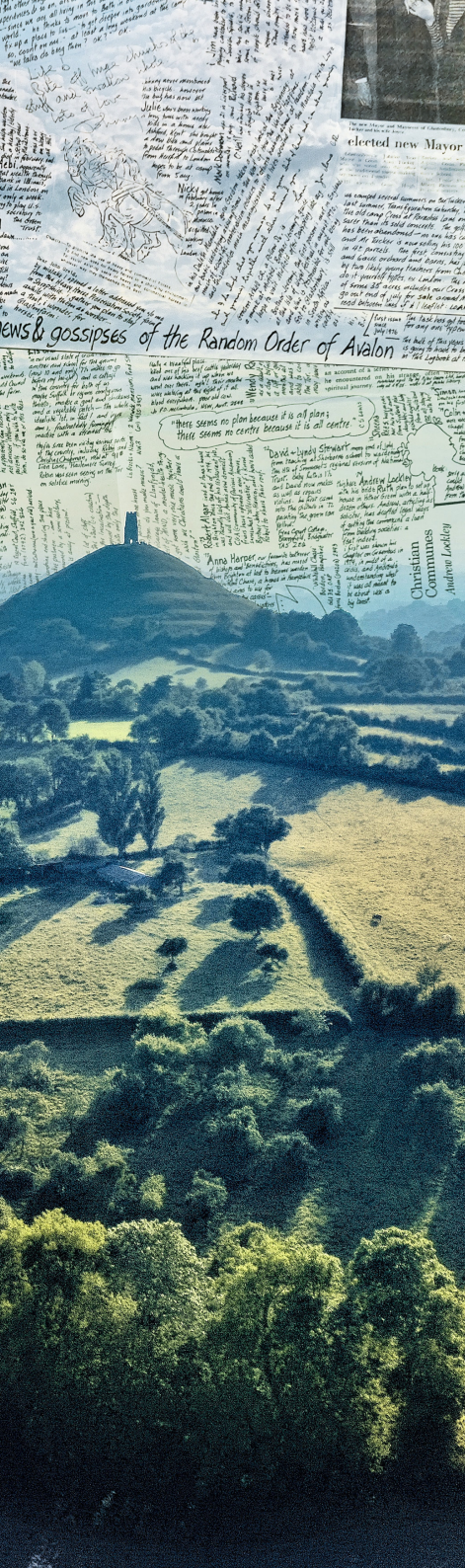
... Escaping a lockup ... 19

... Write XML without writing XML ... 25

... Screensharing from RiscOS ... 39

... Plotting graphs by date in Fireworkz ... 42

... A better welcome from Netsurf ... 54



ednotes

The wrong Nagel – This issue, 24:6, will be the last in the current volume and the final issue under Nagel editorship. But I'm not the Nagel you're used to – I'm Jim's son, Bart. I flew to the UK to help my family after my father died in March [page 43]. One consideration was the future of the magazine: ideally to continue in the hands of a new custodian. During our search, I decided that I would like to produce one final issue as Abbey Press as a tribute to my father and a mark of respect for the magazine and its readership. I've done my best to maintain his style and spirit, using his machine and toolchain. Here it is, and I hope I've met expectations.

Onwards – We are delighted to announce that a new editor for Archive has been found in **Gavin Smith**. Gavin had been working with his consulting editor **Michael Stubbs** on a new printed RiscOS magazine with a view to launch at Wakefield 2020, which was postponed due to the pandemic. The opportunity was presented to instead inherit Archive and they have enthusiastically agreed.

Gavin's first issue of Archive will be 25:1 and its publication date will be announced on the website in due course. The spirit of the magazine will remain: Archive will still be Archive and will continue to be a printed magazine, now produced bimonthly. Gavin intends to send a complimentary printed copy of 25:1 to all Archive members receiving this one, after which existing subscriptions will be fulfilled digitally with the option to upgrade back to a printed subscription. Further information will be detailed in 25:1.

You can contact Gavin at gavin@archivemag.co.uk. Please do so and get acquainted, especially if you are an existing or budding contributor.

Cover – *Glastonbury Tor as photographed by Niklas Weiss (unsplash.com/photos/W2tisCgfu8A), with various handwritten 1970s newsletter paste-ups and prints by Jim Nagel. Composite by Bart Nagel.*

Archive was founded in 1987 by Paul Beverley in support of the Acorn Archimedes computer and its successors using the RiscOS operating system. It is entirely produced with RiscOS software, notably Impression-X and Artworks. Body text is 9½pt Plantin; headings are EFF Eric (Gill Sans). Printed digitally by Direct Offset, Glastonbury. Wordcount this issue: 27,452.

Ed.: **Jim Nagel** Ed.ß: **Bart Nagel** bart@tremby.net

SUBSCRIPTIONS – Archive is moving to a 6-issue subscription model to go along with the new bimonthly schedule. Pricing and payment details are not yet finalized; please see archivemag.co.uk (Prices page) or contact gavin@archivemag.co.uk.

Archive files to download free



Several articles in this issue have files to go with them, as flagged by this graphic in the text. Go to archivemag.co.uk and click on Files.

➤ **An index to your copies of Archive** – On the Files page of the Archive website, click Search to use Kris Adcock's online service. Cumulative Arcscan datafiles are also there.

➤ **All web links** printed in this issue are also on the website in clickable form: see URLs.

Join the online forum – Archive-online is a friendly mailinglist where Archive members help one another with problems. It's gently moderated by Keith Johnson of the East Fleggs. To join, email archive-online@orpheusinternet.co.uk with subject Subscribe.

RiscOS Direct, the distro for new users

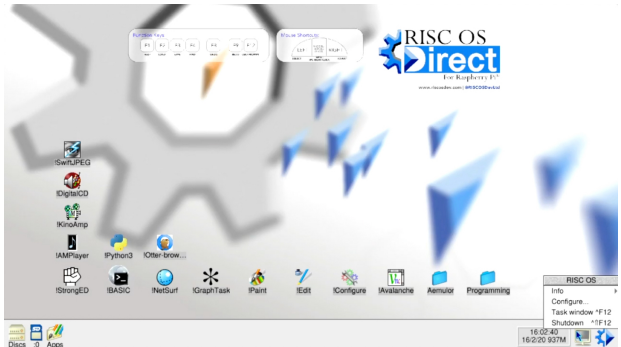
Announced at last year's RiscOS London show, **RiscOS Direct** is now available to download. RiscOS Direct is a distribution of RiscOS for the Raspberry Pi from a collaboration between RiscOS Developments and Wifi Sheep's Tom Williamson, aimed at attracting a new audience. RiscOS Developments are at pains to point out that this is not a fork – this is not a situation akin to the diverging versions of RiscOS from Castle and RiscOS Ltd. As RiscOS Direct shares all its code with RiscOS Open Limited (Rool), it is merely the bundled software and presentation that differs.

Andrew Rawnsley commented, “The OS in RiscOS Direct doesn't contain proprietary code or anything designed to split the OS. It is based on standard RiscOS code, all available through Rool, and Rool roms can be used to update it.”

RiscOS Direct is built from a stock 5.27 rom, but contains a custom theme, preconfigured !Boot and a host of applications to get users started. This helps to ensure that a brand new user to RiscOS isn't immediately lost and left to find software for themselves. Rool's numbering scheme uses even version numbers (5.24, 5.26 etc) to denote “stable” releases, with 5.26 being the first version release under the Apache 2.0 licence, a minimum requirement for the project. RiscOS Direct goes a little further and uses 5.27 to incorporate some Pi-related improvements and key bug-fixes.

RiscOS Direct comes pre-installed with applications and utilities such as DigitalCD, StrongEd, Avalanche and even Python 3 and has a number of icons pinned to the pinboard, including one to drop you straight into BBC Basic.

Andrew said, “There's a lot of content that we haven't been able to include yet due to time and manpower. Most notably a games pack, but there's a lot more software and content that we want to provide too.” riscosdev.com/direct



RiscOS Direct's desktop, with preinstalled apps on the pinboard.

RiscOS Direct, the video series

There are two sides to the RiscOS Direct project, with the other side being a video series on Youtube aimed at helping new users get acquainted with RiscOS. There are currently three episodes: “Getting started with RiscOS”, “Setting up and the user interface”, “Gaming and emulation”. You can find them on the Wifi Sheep channel at youtube.com/wifisheep.

On top of all this, RiscOS Developments now have a stock of SD cards to present free of charge to interested newcomers at Pi Jams and other events, although activities have been somewhat curtailed due to the pandemic.

Impression-X updated again

Work on the modernization and 32-bitting of **Impression-X** continues, with version 5.799 (also known as release 4c) having been sent to subscribers during March and April. Changes include improvements to the native jpeg handling and many bug-fixes.

Impression doesn't currently have its own website but Chris Hall's website (svrsig.org/ImpX.htm) is the next best thing with a comprehensive history of Impression and PDF newsletters keeping users up to date.

As noted in [Archive 24:5](#), while RiscOS Developments have taken over Impression Style and Impression-X, Richard Keefe continues to be the main programmer on the project. It's available at !Store (plingstore.org.uk).

Git for RiscOS

It's currently in an early alpha but RiscOS now has a promising **Git** client from Kevin Swinton (AbbeyCatUK). Git is a distributed version control system created by Linus Torvalds, aimed at coordinating work among software developers. A Git client allows developers to track changes to source code and synchronize with remote Git repositories. Most modern software is developed using Git or similar and RiscOS's entire codebase is no exception. Rool migrated the code from an old source code management system called CVS and it's now available from a modern Gitlab web interface at gitlab.riscosopen.org.

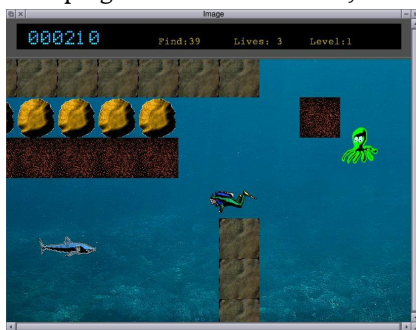
Kevin's client is available with an accompanying PDF user guide at abbeycatuk.info/2020/04/01/risc-os-git but note that it's at an early stage of development.

New games

Virus is a fun little game written by Rick Murray. You play the part of a white blood cell and your objective is to stop the viruses from getting into the red blood cells. There are currently three levels and you win when all the viruses have been destroyed.

No points for guessing where the inspiration for this game came from. Get it for free from !Store.

Scuba Hunter, released at this year's Southwest show, is another new title from



Scuba Hunter, Amcog's new puzzle game.

Amcog. It's an arcade puzzler featuring a diver hunting for treasure by exploring caves, moving boulders and avoiding jellyfish, octopuses and sharks. It's described as having puzzles in the style

of Boulder Dash, and features Amcog's famous music and sound effects. Scuba Hunter is priced just under £10 and is also available from !Store.

Finally, Cameron Cawley has updated his ports of adventure game engines **ScummVM** and **ResidualVM**. These allow a wide range of graphical adventure games to run on RiscOS, such as the *Broken Sword* series, Lucasarts' many point-and-click adventures (*The Secret of Monkey Island*, *Grim Fandango*, *Full Throttle* etc), *Myst*, *Legend of Kyrandia* and even some of the *Might and Magic* role-playing games.

Cameron has managed to get RiscOS back into the lists of publicly-supported releases on these programs' websites, which is particularly significant as it keeps RiscOS in the public eye.

User groups adapt to social distancing

While Covid-19 continues to affect our lives in many ways, several user groups are finding ways to keep meeting, if only virtually. The Midlands User Group met in mid-May via the video conferencing application Zoom. The meeting was described as a test and there has been some discussion on how they might use the technology more regularly in the future.

The Centre for Computing History in Cambridge has hosted the Acorn and BBC Micro User Group several times in the last few years and did so again, virtually, in May. There are a large number of talks available on Youtube from the meeting but of particular interest may be two videos from the founder of the museum, Jason Fitzpatrick, who showed off Steve Furber's "prototype" of the BBC Micro in one video and the museum's very own Phoebe in another. *[On behalf of the museum, Jason also recently took in well over 100kg of Acorn-related press releases and magazines which had been archived at Abbey Press HQ – thanks, Jason. —Ed.B]*

The Wakefield RiscOS Computer Club similarly held their first virtual meeting on June 2 hosted by Steve Fryatt, who looked at RiscOS Direct in more depth. A recording of the meeting is intended to be made available to members in due course.

Chris Cox talks about his time at Acorn

One of the main public faces from Acorn during the 1990s, Chris Cox appeared at shows and headed up the Clan. He visited both Rougöl and Wrocc last year to talk about his perspective of this era at Acorn, including the fateful end of their desktop computer business, and he

shared some great insights and anecdotes at each meeting. They are both on Youtube for your enjoyment. (Hint: The Rougöl video is just slightly higher quality.)

youtube.com/watch?v=ULiSi2-E8-M

Netfetch updated

Netfetch has reached version 5.52 which resolves some flickering issues when using SMTP servers under certain circumstances. It is also updated to use the latest SSL/TLS security, including the FTP component.

New instant messenger client

ChatCube is the first project from Cloverleaf, a three-person team from Germany and Ukraine managed by Stefan Fröhling, and is a modern instant messaging client. Find more information about the Cloverleaf project and Chatcube on page 14.

Rool TCP/IP bounty update

The second part of Rool's TCP/IP stack overhaul bounty has broken five figures. Thanks to 67 generous donations, the bounty now stands at £10,650, with a target of £14,100. "Step 1 shores up some security issues, and in this second part we want the bounty hunter to dig new foundations based on the latest FreeBSD stack, on top of which the final two bounties will build wifi connectivity and a larger IPv6 address space." riscosopen.org/bounty/polls/29

RiscOS 5.28 getting closer

The next major milestone for RiscOS will be 5.28 and Rool now have a webpage which shows the readiness of each target. For example, Beagleboard, IOMD, Iyonix and Pandaboard are all at the green light stage, while Omap5 and Titanium are at amber. The IMX6 and Raspberry Pi are

still at the red stage, the latter mainly due to the challenges of supporting the Raspberry Pi 4B with its USB3 controller. Work continues and you can keep an eye on their progress at riscosopen.org/content/downloads/stable-release-status.

Deep learning for RiscOS

In the first of its kind for the platform, Paolo Fabio Zaino has ported a deep learning C library to RiscOS. On a basic level, deep learning is a subset of machine learning, and uses artificial neural networks inspired by the way the human brain filters information. It allows computers to learn by example, the way our brain does, solving complex problems and classifying images, text or sound.

Paolo provided some practical examples, including facial recognition, car detection and counting in video (for a parking system, for instance) and identification of certain sounds in an audio stream.

The library is only 6K in size and comes with many code examples, along with the source. Paolo says, "This is the first of a set of deep learning libraries I am porting to RiscOS for fun."

For now, the library is only available via Github at github.com/ZFPSystems/genann.

Browser updates

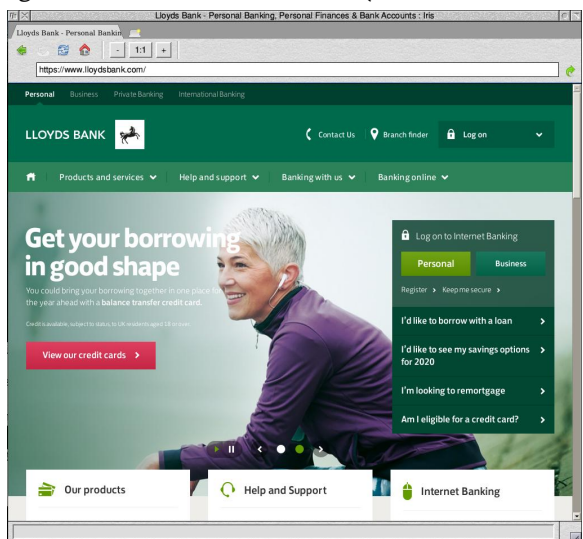
RiscOS Developments have shown recent builds of their two web browsers, **Iris** and **OWB**. The former is a comprehensive cutting-edge port of the Webkit engine which powers many modern web browsers and applications, while the latter is a smaller, faster browser based on older code.

While OWB initially stood out, *The Iris browser can already render many modern websites.*

Iris has taken much of the limelight recently, as it has now progressed to a much more usable state and is compatible with a wide range of websites. Both browsers have native RiscOS user interfaces on top of back ends synchronized with the underlying cross-platform engines. This allows the browsers to track changes made for other platforms, reducing the overhead of keeping them up to date.

RiscOS Developments have made both browsers available to their investors and shareholders, as well as many who have made one-off donations. The next phase will be to open things up to people who supported development by buying the OBrowser CDs at shows and by post over the last couple of years. A final phase will see a more general release of the browsers to everyone. RiscOS Developments stress that the final products will be provided free of charge and open-source for everyone to enjoy.

Meanwhile, owners of R-Comp Interactive's Armbook (which since the last



issue has become generally available) have been given a special preview of Iris in the form of **Pinebrowse**, a special edition provided with that machine to ensure it ships with a Javascript-capable web browser. To quote Andrew Rawnsley, “With many wifi adapters and hotspots requiring login or special configuration via Javascript-based web pages, a portable computer without a compatible browser is going to struggle. Thanks to a little negotiation and horse-trading with RiscOS Developments, we’re pleased to offer Pinebrowse free of charge to Armbook owners to help them out.”

Finally, just before going to press, Chris Gransden announced on the RiscOS Open forums that his **Otter Browser** port had been updated. While the core browser remains unchanged, the new version is easier to run, more compatible with modern machines and has a splash screen to show that it is loading.

Fireworkz Pro

Fireworkz Pro continues to receive regular updates. Version 2.24 was released in March with a number of new spreadsheet functions. Date functions received `DAYS()` and `ISOWEEKNUM()`, while the new mathematical functions are `BASE()`, `DECIMAL()`, `GCD()` and `LCM()`. There are also 17 new engineering functions, such as `BIN2DEC()` and `BITAND()`, and a few existing functions have been tweaked such as `TRIM()` and `VALUE()`, which now accept text strings longer than 200 characters.

Six bug fixes later and Fireworkz Pro is now at 2.24.06 and available from !Store. The free-to-use (but simpler) Fireworkz also received these changes and it’s available on Packman. Similarly, its much-loved predecessor, **Pipedream 4**, also saw updates recently, available on !Store and Packman.

Organizer 2.29

The Southwest show saw the release of Organizer 2.29 with significant changes on the road towards secure VCard contact synchronization. 2.29 sees a major rewrite of the address book, upgrading the VCard format from v2.1 to v3. This paves the way for synchronization of address data with other devices. Other improvements include support for higher-resolution photos, new Ical export scripts, a 32-column 364-day window in the diary and more. Upgrades are available from North One Communications Ltd.

First for free

First, Serious Statistical Software’s statistical analysis software [*there’s a tongue-twister —Ed.B.*], was historically considered one of RiscOS’s heavyweight applications. Following the death of the company’s owner, Robin Edwards, the software has been made available as a free download hosted by the Midlands User Group. The software is 32-bit safe, but some portions may need Aemulor on recent hardware. Interested readers should visit mug.riscos.org/1st/first.htm.

LanMan98 to be open-sourced

Initially announced at the London show last year, but then delayed while awaiting the source code, **Lanman98** (previously sold by Warm Silence Software) was to be released by RiscOS Developments at Wakefield 2020 as a free open-source download. At the time of writing, it still hasn’t appeared on the usual download sites, but readers are encouraged to keep an eye out. In a post on the Rool forums, Andrew Rawnsley confirmed that everything is all set, pending only the time needed to package it for release.

Elesar keep busy and fight the Covid blues

Elesar, developers of the **Titanium** board and of popular programs such as CloudFS, Prophet and Font Directory Pro, have had a busy winter. Not only have they produced updates to several programs, they have also launched a new version of their wifi Hat for the Pi, now with built-in battery-backed clock.

They have also teamed up with Amcog to offer a special bundle of games to help beat the Covid lockdown boredom. Check out their website for ordering information (elesar.co.uk).

Raspberry Pi with 8 gigs

The long-rumoured eight-gigabyte model of the Raspberry Pi 4 is now on sale. RiscOS users are currently unable to take advantage of more than four gigabytes of ram, but one-man coding machine Jeffrey Lee has been working on adding

support to the RiscOS kernel for the “long descriptor” page table format, which will improve support for machines with lots of memory. You can keep track of his work at gitlab.riscosopen.org/RiscOS/Sources/Kernel/-/merge_requests/15.

In the meantime, the eight-gigabyte Pi 4 will work with RiscOS but will be restricted to four gigabytes (still rather luxurious for RiscOS). At the time of writing, the main USB3 ports on all Pi 4 models are non-functional under RiscOS, but the gigabit ethernet port now works well in beta OS builds.

Date for Wakefield 2021 announced

The Wakefield show had to be cancelled this year but the date for 2021 has already been set: Saturday April 24. This will be the 25th show, and will take place 25 years after the first in its regular venue, the Cedar Court Hotel. See wakefieldshow.org.uk for more details.

Southwest show 2020

Mark Stephens

The Southwest show took place on Saturday February 22, which seems a world away now. This was the second year in its new venue at the Arnos Manor Hotel, Bristol. The view is less spectacular than the old venue's was, but the transport links are much better.

As usual, the event was run by Richard Brown (with his daughter roped in) and Andrew Rawnsley. With one year already under their belt, this was a slickly run event with some nice new flourishes.

The show took place in the same two



Like last year, the show was hosted at the Arnos Manor Hotel.

rooms as last year for stands, and with a much larger room (perhaps too large) for the talks.

The stands

All the usual suspects were there, with **AMCS**, **R-Comp**, **Orpheus**, **Rool**, **Steve Fryatt** and **Soft Rock Software** all showing off their wares. It was also really nice to see the welcome return of **MW Software**. No new releases, but Martin was as elegantly dressed as ever.

Some of the main appeals of the shows are the chances to talk to people, to catch up on new developments and to physically try new software and hardware.

Adrian Lees was also back showing off Evince, his fast VNC software which also allows easy keyboard switching similar to Synergy.

Richard Keefe was demonstrating the new printing and imaging features in his new release of Impression-X.

Amcog Games had their full games catalogue including their new Repton-style Scuba Hunter game. Cameron Cawley, meanwhile, had the open-source engine for point-and-click games, ScummVM, ported to RiscOS and Gareth Lock had some nice quiz programs to challenge you.

Drag'n'Drop released the latest edition of their magazine. This is in PDF format so is easily read on any computer.

Organizer had a surprise new release which updates the supported VCard version from 2.1 to 3.0 and has lots of enhancements to the address book. See Newsdesk, page 7, for more.

RiscOS Bits had some nice covers for the Armbook and a proper Armbook logo to stick over the generic Pinebook label.

Chris Hall had his range of projects and was busy trying to get RiscOS up and running on the Raspberry Pi 4.

If you were interested in joining a user group, **Rougol** (London) and **Sroug** (Southampton) were both represented. Andrew Conroy of Sroug had a selection



RiscOS Bits with a range of cases and systems.

of hardware projects running on Raspberry Pi machines.

Wifi Sheep have pivoted their business away from hardware sales to focus on their Youtube channel. They were streaming the event live and had an impromptu recording studio set up in the conservatory for interviews.

Rob Coleman was demonstrating his work on networking for Doom, and there were some really nice demos using BBC machines and the Tube interface to play video and push the BBC way beyond its original capabilities.

John Norris was showing his RiscOS software to learn the art of bell ringing and **Tasty Treats** had their usual range of delicious jams.



John Norris shows his bell ringing software while Mary Norris sells homemade jams and chutneys.

Southwest show highlights

There was, of course, the usual charity stand.

The talks

There were four presenters this year. The talks were filmed and can be viewed online at riscository.com/videos.

Richard Brown

On behalf of RiscOS Developments, Richard showed off RiscOS Direct, which bundles in lots of additional software and tries to provide a more polished experience in order to attract Raspberry Pi users to RiscOS. They have been collaborating with Wifi Sheep to provide some tutorial videos on Youtube to help onboard new users.

Andrew Rawnsley

Andrew showed off the Armbook laptop, then talked about latest work on the Iris and Origyn browsers which will provide better web support to RiscOS (*Newsdesk*, page 6). There was an updated release for RiscOS Developments shareholders, which will hopefully be followed later in the year by a general release for all to freely use, once the code is fully tested and complete.

The new releases behave much more like RiscOS programs with lots of new user-facing features like a hotlist, favourites, Ctrl-F to search the page and so on.



Richard Brown presenting RiscOS Direct for RiscOS Developments.

Steve Revill

Steve, on behalf of Rool, updated us on the new Desktop Development Environment release, plans for RiscOS 5.28 and the current state of work on their “bounties” (sponsored features). There are also plans for an updated version of the user guide for 5.28.

Stefan Fröhling

Finally, there was a bonus talk from Cloverleaf, both talking about the project in general and showing the new Chatcube messaging software. More on page 14.

In photos

Many more photos are available online at iconbar.com/articles/South-West_Show_2020_in_pictures/index1529.html

Mark Stephens

markstephens@idr solutions.com

hint&tip QR codes: more than just URLs

Scanning a QR code can do more than open a webpage. The contained data is usually a URL, but other possibilities include an email address (mailto:), phone number (tel:), calendar event or an entire contact card. It can also be wifi connection info (e.g. WIFI:T:WPA;S:mynet;P:hunter2;;) – handy for guests and visitors – or simply arbitrary plain text.



My new Titanium

Geoffrey Baxendale

I bought the Kinetic RiscPC at the Wakefield show in 2000. Apart from the odd new harddrive and several OS updates it has run well for all that time.

But it was beginning to struggle with Netsurf on modern sites and my increasing store of emails, accessed by **Messenger**. Over the years I have contemplated Beagleboards and Pandaboard etc and I did have a play with a Raspberry Pi, but it wasn't until the Titanium board came out that I felt that here was a "Proper Computer".

I put off buying one for a while but just recently I decided that the time was right, so I ordered one from Elesar together with a suitable PSU and IO shield. CPC supplied a "recertified" Seagate 500G Sata drive and a DVI-to-HDMI lead. After much searching I bought a Coolermaster Elite 110 box from Ebuyer. All companies gave excellent service with fast deliveries.

Done-it-myself

I went down the self-build route rather than buying a ready-assembled unit on the grounds, as Gerald Fitton would say, "of having fun". How did it go? Well, it certainly was, and is, fun.

Assembling the bits into the case was straightforward. The board is a perfect fit. I find PC nomenclature confusing: ATX covers a lot of ground, and I think the board is actually mini-DTX in size, so you need a case that will take that.

Rob Sprowson at Elesar provides quite detailed instructions for setting it up but I found one or two gotchas.



Rear view of Geoff's Titanium. The machine has two DVI ports and can drive two monitors. (Surprisingly for the editor, it does not have the more modern HDMI connections, which feed both video and audio to a monitor.) Six USB sockets, two gigabit network ports – and two serial ports – complete the array.

Screen, not scream

I connected up my monitor via the DVI-D/HDMI lead. It's worth mentioning that the output from the Titanium's DVI socket is digital-only (DVI-D) and does not give an analogue signal for a VGA monitor.

On plugging everything in and switching on... nothing happened! It took me a moment to realize that it's like a PC: you have to press the Titanium's power-on switch, not just turn on the PSU. I was then rewarded with a desktop that was low-res but readable.

I downloaded the self-extracting Harddisc4 image (version 5.22) from the Rool site (riscosopen.org), copied it to a memory stick and inserted it. Again, due to my limited knowledge of PC systems, I hadn't realized that several Fat *[file allocation table —Ed.]* formats exist – the stick was Fat32 and RiscOS couldn't read it. I went back to the laptop and

My new Titanium

reformatted it to the lowest Fat, which I think is Fat12, recopied the HD4 zip and tried again. Success this time: I unzipped it, found **HForm** and initialized the harddrive.

RiscOS found only (only!) 250G but that was OK for me. (I think there is a newer version of HForm that might be better.) I copied the rest of the HD image over and rebooted.

Now I could use Configure to set up a 1920×1080 resolution for the monitor to give me a decent display. Unfortunately the picture was too big due to TV overscan. Eventually in the monitor's own menu I found “just scan” as the aspect ratio; selecting that sorted out my problem. In Configure, I have since set the monitor type to Auto(M2252D) – I didn't realize RiscOS would do that!

Network next

Next came trying to connect to the network. There seemed to be no way of configuring the network interface. After a lot of messing around I eventually put in a support ticket to Elesar for help. Rob came back very quickly to say I need the “nightly beta” build of HD4, not the stable one. Apparently the stable build is missing the program EtherCPSW, which must exist in !Boot.Resources.Configure.!!netsetup.Autosense. With that installed I was able to connect to my network.

I copied **Netsurf** from the RiscPC and ran it, only to have it crash when I clicked the icon. I couldn't see what was wrong, so I asked on the Netsurf list, where a kind soul came back with the solution that you have to go into Netsurf choices and alter the Images setting to Use OS. Only problem was that when I clicked “Images” it

crashed. I solved that by changing the setting on the RiscPC to see what had changed in Netsurf's choices file. I then edited the choices on the Titanium and all was well. In !Boot.choices.www.Netsurf the settings need to be:

```
plot_fg_quality:128
plot_bg_quality:128
```

I added these at the end of the file. So that was Netsurf running and very well too!

Backup system

I use a Synology Nas [*network-attached storage —Ed.j*] for backups and so on, but I could not make head or tail of **Omnicient** and **Filemanager** in order to connect to it. On the RiscPC I used to use **Samba** to connect to Windows, but when that stopped working due to changes in Windows I switched to **Sunfish**. Sunfish is an NFS application for RiscOS [*the clue is the letters NFS in the name Sunfish —Ed.j*]; it worked fine on the Kinetic, so I tried that. No luck – Sunfish wouldn't run.

Searching the internet seemed to suggest that the front end needed recompiling, but I couldn't find a newer version. A plea on Archive-online brought



Front view of Geoff's Titanium in its CoolerMaster case.

a reply from the ever-helpful Steve Fryatt, who gave me the link to the new version of Sunfish: riscosports.co.uk/downloads.html [*Chris Gransden's site —Ed.*]. Google doesn't find everything!

After I set up permissions on the Nas I had access to it via Sunfish. Great.

Email client

I couldn't get Messenger Pro to run: "abort on data transfer". After much messing about I eventually found a post by Andrew Rawnsley on a Rool forum answering the question. One component application within MPro, Espeak, is not 32-bit. I've never used this, so not having it was no problem. I commented out references to it in Messenger's !Run file. I copied my active !Newsdir across from the Kinetic, which took a long time. Then I installed **Popstar** and **Newshound** – and email was up and running!

Textease alas

One particular program I have used a lot over the years is **Textease**. The company that marketed it lost interest in RiscOS, and as far as I know it never got updated for any machines after the RiscPC. As a partial replacement I adopted the original **Ovation** (not Ovation Pro), which I had used in the Archimedes era many years ago. Rick Murray has updated the original Ovation (go to heyrick.co.uk/blog and just under the calendar click Ovation). But Ovation lacks one feature: rotated text. My Textease documents had loads of it.

I knew about Aemulor but suspected it wouldn't run on the Titanium. I downloaded it anyway from !Store, tried it, and I was correct, but it directed me to sendiri.co.uk/aemulor – and, lo and behold, Adrian Lees has provided freely downloadable versions for all the new

machines, including the Titanium! It works, too. Textease runs just fine.

Smooth sailing

Plenty of other software that I use that worked straight away: **PS3** and **RemotePrinterFS** (to print via the network to my laserprinter), **Thump**, **StrongEd**, **Powerbase**, **Labella**, **FTPC**, **Pipedream** and **DPscan**.

To avoid having separate keyboards and mice for Kinetic and Titanium, I used an old KVM switch. It expects PS2 connections so I bought a PS2-to-USB adaptor. This has sockets for the PS2 connectors and one USB output which plugs into the Titanium. Works really well!

May the Force be with you and Don't Panic. ;-)

Update

In the time since first writing this article in 2018 I have found my Titanium machine to be very good.

At the beginning I had a problem with it randomly rebooting. After much testing, even trying a new board, it was solved by a change in the rom made by Elesar. We never did find out what the real problem was – it apparently affected only me – but it has been fine since the rom change.

The update to RiscOS 5.24 went smoothly, though I have never fathomed how to format the harddrive to use its full 500-gigabyte capacity.

I had a play with Linux but decided it was a bridge too far for an old Bear!

Geoffrey Baxendale thebears@onetel.com



Geoff lives at Darwen in Lancashire. His oxymoron of the day in the email he sent to Archive was "pretty ugly". Another one: "political science". Using RiscOS, he is webmaster for the Leyland Society of Model Engineers, leylandsme.co.uk.

The Cloverleaf project

Stefan Fröhling

I initiated the Cloverleaf project to tackle three issues. First, to close RiscOS software and driver gaps. Second, to create a new brand that solely uses RiscOS as its operating system – vital for promotion of RiscOS as a viable choice. And finally, to create the grounds for RiscOS to become well known around the world again.

Substantial funding is needed, which we hope to raise with an international crowdfunding project via the Kickstarter platform. I have teamed up with Andrew Rawnsley. He is responsible for adapting new hardware to RiscOS while I am

responsible for the project management, marketing, public relations and other software implementation. The new hardware will be based on the Rockchip RK3399 system-on-a-chip. Based on Arm's "Big Little" architecture, this chip offers clock speeds up to 2GHz. It integrates a dual- or quad-core Cortex processor with a separate Neon coprocessor, and also includes an Arm GPU which provides graphics acceleration.

This chipset enables us to provide a wide range of hardware. We will offer desktop machines, 11- and 14-inch laptops and eventually also a 27-inch all-in-one computer. At the start we will use existing hardware to minimize risk.

The project will also seek to close known gaps in the

RiscOS software portfolio, with an aim to make RiscOS ready for the mainstream. There are missing drivers for wifi, Bluetooth, NVMe drives, GPUs and sound input, and we intend to do something about this. Breaking the 2G ram limit, multicore processing and 64-bit are also on the agenda for the future.

All RiscOS users will benefit from the Cloverleaf project as we will also produce new programs, freely available and likely open-source. We have already released the first free software, **Chatcube**, which is an instant messenger.

An essential part of a modern

Big magazine, little nutshell

Arm's "Big Little" architecture (stylized big.LITTLE) pairs a fast powerful processor with a slower low-power one. The idea is to make power usage savings where high performance is needed only in short bursts, which is a very common pattern. Both cores have access to exactly the same memory regions, and workloads can be switched between the two cores transparently. Arm says that power usage can be reduced by up to 75%.

This technology reminds your temporary editor of something. A search for some foggily-remembered keywords pulled up the files sent to Dennis Publishing in 1997 for Jim Nagel's Acorn Shopper column of *Computer Shopper* issue 111, which contained an April Fools' Day jape as part of the "Nutshells" section:

- Hot news from Arm Ltd: A revolutionary multiple-processor system called Hand is launched this month. The device, on a single slice of silicon, comprises four 32-bit 400MHz StrongArm processors along with the clever Thumb extension that allows them to link to cheap 16-bit or even 8-bit memory. The price is an incredible £97.04 including Vat. Hand is expected to lead the way in artificial intelligence. Early samples of the device will be available to developers who propose the most innovative applications for it. Contact hand@reality.demon.co.uk.

Pulling out a paper copy revealed that, unfortunately, this didn't make it to print.

operating system, currently missing from RiscOS, is a fully-featured web browser. With great timing, this gap will be filled by RiscOS Developments this year [see *Newsdesk, page 6 and SW show report, page 10 —Ed.BJ*].

A call for support

If you want to support our project please follow our Twitter, Facebook and Instagram accounts and subscribe to our newsletter on riscoscloverleaf.com. And please repost or retweet or otherwise share our content so that we can get more coverage – we can only succeed if as many people as possible hear the word about RiscOS. We want not only to address current RiscOS users but also to open up new markets by spreading the word to international computer magazines, blogs, YouTubers and others.

In the Kickstarter you have many options on how to support us and RiscOS, such as T-shirts and RiscOS support packages, right through to complete



hardware. Each contribution will enable us to develop RiscOS further. We want to work together with RiscOS Open Ltd and RiscOS Developments to unite support for RiscOS as the resources are so


limited. We will even support them financially if we get enough funding in this crowdfunding project. We also plan to provide dealer discounts for our hardware so that we can get wider coverage worldwide. The Kickstarter campaign will be published soon and the actual start will coincide with this year's London show.

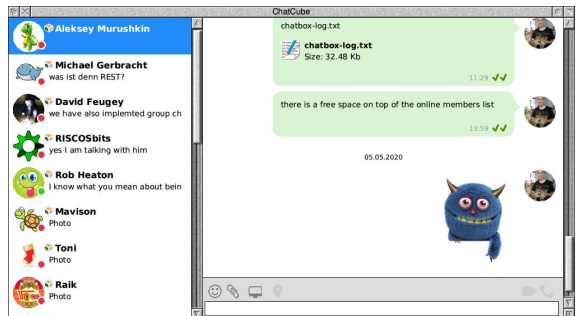
Stefan Fröhling stefanfroehling@yahoo.de



Stefan, from Germany but based in Thailand, is a veteran of Acorn computers. He co-developed the first IDE harddisc controller for the platform and has also developed ray tracing and vocabulary training software for RiscOS.

Multi-protocol instant messaging comes to RiscOS

 Chatcube is an instant messenger for RiscOS, allowing you to chat with other RiscOS users. Chatcube will be the gateway for other messenger platforms – we've already added support for Telegram and, funding permitting, others will follow such as Twitter, IRC, Signal and more. Send text messages, stickers, screenshots and files to your chat partner, have group chats, or ask questions in the public RiscOS chat group. There are future plans to implement voice and video calling too. Currently, all members of Chatcube are



!Chatcube, the new multi-protocol chat client for RiscOS.

publicly visible, so if you have no RiscOS friends yet, now's your chance!

Chatcube is available for download from riscoscloverleaf.com and !Store (plingstore.org.uk).

Building the smallest portable

Chris Hall

I don't know how many prototypes I have made, expecting each to be "the finished product" (whatever that was going to be), but I do now feel I am getting there. It all started in December 2016 after I had played around with small HDMI displays and wanted something more pocket-sized.

So I thought it would be a good idea to describe the construction process in case anyone wanted to follow in my footsteps. My first step is to decide which components I am going to fit into the case and design the case.

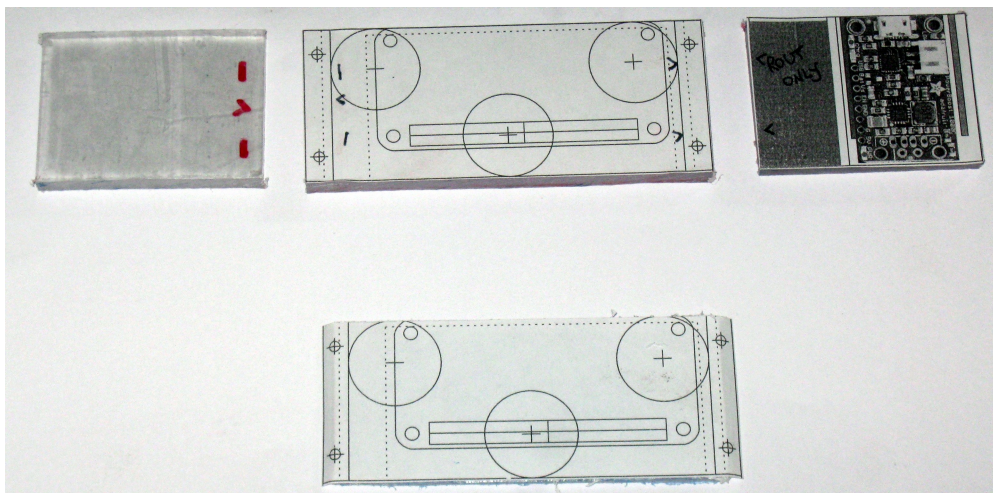
Case design

I have designed cases to hold the Titanium computer (the largest I have done) where I could afford to have each sheet of polycarbonate professionally cut to size. My supplier, Barkston Plastics (also known as "the Plastic People"), will cut 2mm and 6mm polycarbonate sheet to

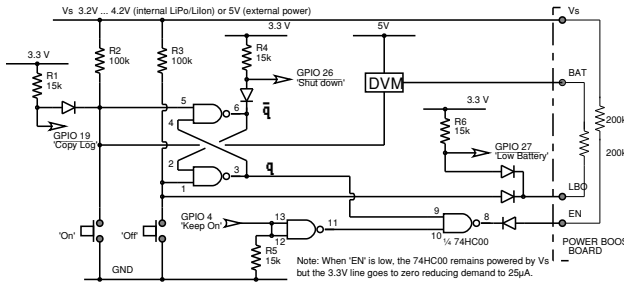
order, charged by surface area with a £3 minimum per sheet and £50 per order. Larger sheets are already close to the minimum price. I have now several sheets 2m long (the maximum length) at varying widths so I can cut the thicker sheets to size with a hacksaw in a single cut, and the thinner sheets using a Stanley knife.

My latest project needs to include a Lithium battery, Powerboost board, Pi Zero W, breakout board, OLED display, GPS module, digital voltmeter and an ADS1015 board. This will allow me to manage battery life, automatically (and tidily) shutting down on low battery and to display remaining battery capacity.

My design layout is shown on the next page. I use this when wiring the breakout board and print the A4 sheet onto a full-page sticky label, cut the pieces to size and stick them onto the cut plastic sheet to indicate where holes need to be drilled (see below). I have 6mm sheet in widths of



There are three 6mm-thick pieces that form the sides and base, and one 2mm-thick piece that forms the shelf. Holes are shown to assemble the sides and base, to add magnetic feet to the base and to mount the Pi Zero. The space below the 40-pin header is cut out to make room for the header pins and solder.

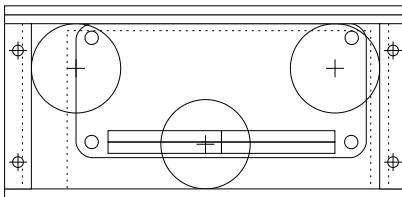


ADS1015 Analogue to digital converter

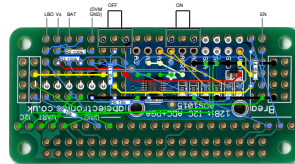


Parts list for case

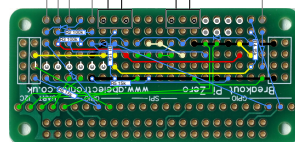
- 6mm x 37 x 90 (2 off)
- 6mm x 37 x 50 (2 off)
- 2mm x 62 x 90 (3 off)
- 2mm x 37 x 82 (1 off)



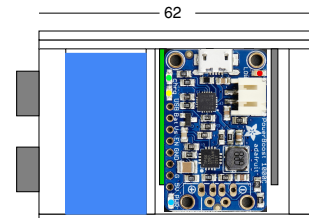
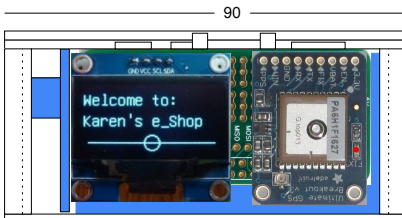
Plan View (6mm top removed)



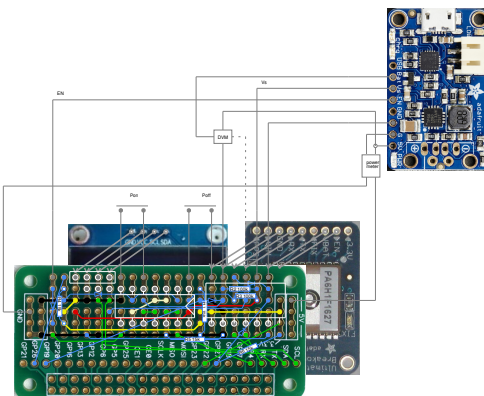
Underside of breakout board with ADS1015



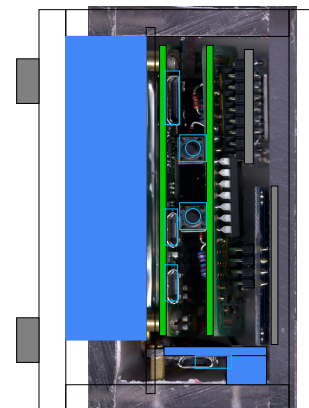
Underside of breakout board without ADS1015



Side View



Back View with connectors



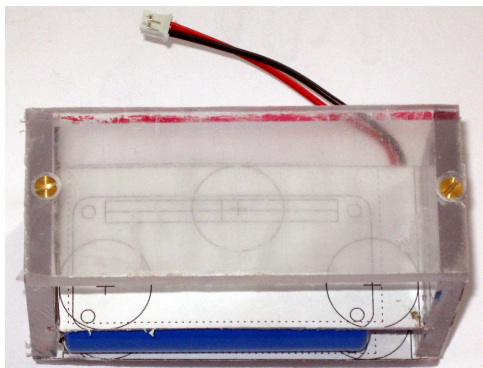
Files: see p.2

The design layout showing circuit board wiring, interconnections, case layout and templates for drilling the case components. It also shows placement of sockets and cut-outs. See full-size images with this issue's files.

Building the smallest portable

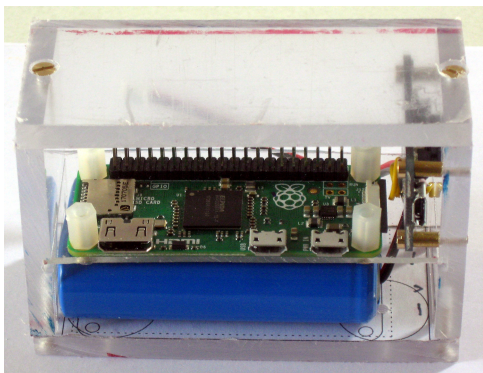
50mm and 90mm, and 2mm sheet in a width of 83mm.

It does not take long to cut the plastic sheets and to drill and tap the necessary holes to assemble the case.



The base, sides and shelf are assembled after about 1½ hours.

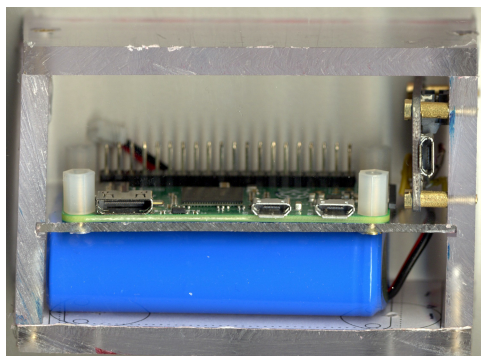
Once it all fits together the holes for mounting the Pi Zero and the Powerboost board can be drilled and tapped.



Another 30 minutes and the major components are mounted neatly.

The front and back of the case (the sockets are all on the back) can be drilled and tapped to suit, avoiding any places that would obscure things which need a clear viewing window, such as the OLED display.

This unit will have a digital voltmeter



A scan of the back of the unit will allow sockets to be placed exactly using a sticky label as the template. There is space to the right of the battery for a GPS extension antenna socket.

fitted to allow the analogue-to-digital converter to be calibrated. This will also read battery voltage and digitize it.

Breakout board

The breakout board provides the essential connections between the Pi Zero and its power supply (provided by the Powerboost board). It also provides sockets for the OLED display, GPS module and ADC board, as well as containing the power control circuitry.

It is laid out carefully because once it's all soldered up it's too late to make changes very easily. There are only a few interconnections between the Powerboost board and the breakout board as follows: GND (black); EN (green); Vs (blue); LBO (white); BAT (white) and 5V (red).

The "on" and "off" switches are mounted on the breakout board directly and the interconnecting wires are arranged with enough slack that the shelf can be slid out to check that it is all working once finally assembled. This takes some thought with surplus wiring accommodated to the side of the battery along with the wiring for the extension GPS antenna and the digital voltmeter module.

GPS, OLED and ADC

Not shown on the circuit diagram are the connections to the OLED display (3.3V, GND, I²C SCLK and I²C SDA), to the GPS module (3.3V, GND, RX, TX, GPS-EN and GPS-BAT) and to the ADC board (3.3V, GND, A0, A1, I²C SCLK and I²C SDA) – these are mostly made via plug and socket on the breakout board.

The I²C connections use a device address: &78 for the OLED display and &90 for the ADC board, sometimes referred to with the 7-bit addresses &3C and &48 respectively).

The serial connection to the GPS module can be used both ways but I just receive data from the module. Pulling GPS-EN low will turn off the GPS module and allow the serial port to be used with another device, if necessary. Battery voltage can be monitored by the ADC board by connecting BAT on the Powerboost board to A0 and GPS-BAT to

A1 (the board can monitor four channels but I am only using two).

The final product

Here we have a RiscOS portable in a case measuring 3×2×4 inches with full battery management: display of remaining life and automatic and tidy shutdown under software control before the battery is exhausted. The battery lasts about 18 hours and its capacity can be extended by plugging in an external powerbank. While stationary it can display the time and date and while moving it will show the speed in miles per hour and log the journey on the SD card, uploading it to a USB pen drive on demand. It consumes 850mW when active and 125μW on standby.

Chris Hall chris@svrsig.org



Retired since 2005, Chris still has a long list of “things to do” – the list doesn’t seem to be getting any shorter. The next project is to publish another book on signalling.

hint&tip Escaping a lockup

After many years away from RiscOS I’d forgotten what to do when the machine seems to have locked up. While producing this magazine I’ve had many such lockups, and in each case had resigned to a hard reboot. It feels odd to thank Impression for having no “undo” feature, but it has meant I’ve been saving and backing up often enough that I haven’t yet lost any significant work to lockups!

Web searches haven’t helped me a great deal with RiscOS matters so far, but this was an exception. A result on riscos.com reminded me that you can interrupt the machine and get a series of “Press Stop to terminate X” dialogues by pressing **Alt-Break**. The memories flood back.

It seems to me this interface could do with modernization. Not only are there no clues about whether a given task is healthy or not, if I click “next task” too hastily and miss my target I need to start over, and now they’ll be in a different order.

Bart Nagel bart@tremby.net



Making web-books with Bind

Gavin Wraith

Everybody knows that computers are magical devices that can take away the tedium of much work. I suspect, however, that there is some subtle law of conservation of intellectual effort that has never been precisely formulated but whose consequences affect us all. It goes something like this: the more powerful and generic a program is, the harder it is to use. Readers acquainted with Jack Vance's works may recognize here the difficulty that sorcerers have in forcing into their brains more than a handful of potencies at any one time.

Now and then I have to write a program. It is a frenzy that consumes me for a few days until it burns out. The attacks come less frequently with age, as my powers of concentration wane. In the days leading up to them, idleness and realism make alliance to warn me of the futility of the idea. Any real project, anything of practical utility, would be way beyond my competence. So these excursions are flagrant self-indulgence – the scratching of an itch. The days are long gone when I could master anything very new, so I tend to stick to things that I could make work in the past, tatting together old rags – I think “code re-use” is the grander jargon. My latest effort is called **Bind** and I would like to explain its purpose and the thinking behind it.

Some while back I began to make a web-book, *Risclua for Beginners*. By “web-book” I mean something whose chapters are web pages, with a separate web page for contents, with a uniform appearance (or “theme”) and with a consistent means of navigating through the pages. Such things have been around ever since HTML

was invented. I soon realized that I was only using a tiny part of HTML, and that it made sense to automate the process of creation. To begin with I used **Weave**, a program I developed specifically for creating marked-up text. I realized eventually that Weave was probably not easy for others to learn. So my next effort was **Infuse**, a program which I described as a device for “pouring content into a container”. The idea was that you made a container by copying a web page whose layout you liked and then replacing all its textual or graphical content by variables; and then you wrote your own content by describing what the variables should stand for, and Infuse would do the heavy work. Then I realized that this was still asking too much of users. I needed to make web-book creation really simple, with no hint of programming effort required, just the writing of text or choice of graphics. That is the motivation for Bind.

How it works

To make a web-book with Bind you first create a directory for chapters. The directory's name is taken as the web-book's title. The names of the files in the directory must start with a pair of decimal digits, followed by the chapter's title. Thus, 57Heinz would describe a 57th chapter called “Heinz”. The chapter numbers must start at 01 and form a consecutive sequence. The chapters are described as a list of sections. Each section begins with the word `!section` followed by optional header text, followed by the text of its body enclosed in braces. Sections cannot occur inside sections. Inside a section you can add links, preformatted code and images. These have the forms `!link url { text }`,

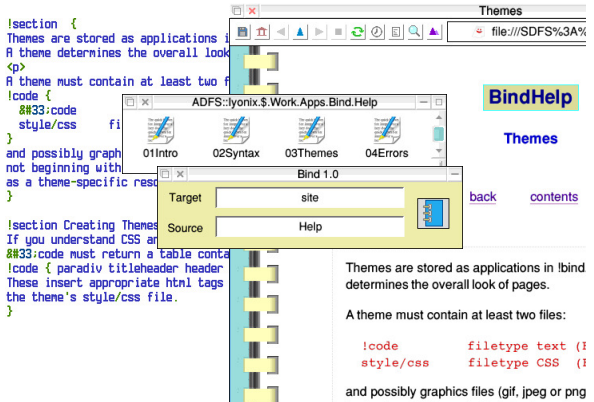
`!code { text }` and `!img filename`.

Images can occur in the text of a link. Text can contain newlines and HTML tags. That is it. Very simple, very limited. When Bind interprets the chapters it remembers the section headers and displays them on the contents page. Bind uses the notion of a “theme” for how things get displayed, and three are provided. Themes are implemented as applications inside `!bind.themes` containing, at least, the files `!code` (Risclua) and `style/css` (CSS). The idea is that all the techie stuff is wrapped up in themes,

leaving the specification of the content as simple as possible. When run, Bind copies all the files inside the chosen theme, apart from those starting with `!`, to the target directory as theme-specific resources.

Once you have created your directory of chapters (the “source” directory) you doubleclick `!Bind`. A filer window opens showing the available themes. You choose one by doubleclicking its icon. This opens a dialogue with two slots and a greyed-out button. Drag the source directory into the lower slot, and the target directory (where you want the web-book to appear) into the top slot. With both slots filled the button will display the chosen theme. Click it for Bind to create the web-book or to update any chapters edited since it was last made. So Bind closely resembles Infuse in its *modus operandi*, and also has a touch of Make in that it leaves old material alone.

Bind and Infuse use Adam Richardson’s excellent Basic program **MultiError** for displaying error messages in the desktop. Risclua is used in Bind for the front-end Wimp task that puts up the dialogue (`!Runimage` and `bin.lib`), and separately for the non-Wimp program that interprets chapters and builds web pages



Bind combines a theme description with content written in a simple language (mainly plain text) to produce web-books.

(`bind` and `bin.xhtml`). The latter can, of course, be used from the commandline (once the system variable `bind$theme` has been set) and hence Bind can be run from within another application if necessary.

You can do it

Bind is partly an experiment. But I have found it a practical tool for making web pages without fuss. In that respect it is like the various “markdown” languages, but much more rudimentary. The only commands are `!section`, `!code`, `!img`, `!link` and the convention of sandwiching cited text between double underscores `__like this__`. If you know HTML and CSS then you can do more, of course – you can design your own themes. But the point is to have a system that lets you go a long way with knowing very little.

Download Bind at www.wraith.plus.com.
Gavin Wraith gavin@wraith.plus.com



An old Wykehamist, a retired Reader in Mathematics, a long-term user of RiscOS, a Lua-freak. Gavin once saw a first edition of Gilbert Murray’s *The Oxford Book of Greek Verse* as part of the decor of a pub in London, and they gave it to him for free when he enquired.

XML and direct debits

Richard Wheeler

At the end of March 2017 I retired. Since 2006 I had been working on designing, developing, implementing and supporting payment systems – specifically for Sepa payments. I thought all that was history, and then what do I find but the article by Paul Porcelijn about Sepa payments (“An XML batch to bank”, [Archive 24:3](#)). Of course *I* knew what it was about, but how many others know what Sepa is?

In the beginning

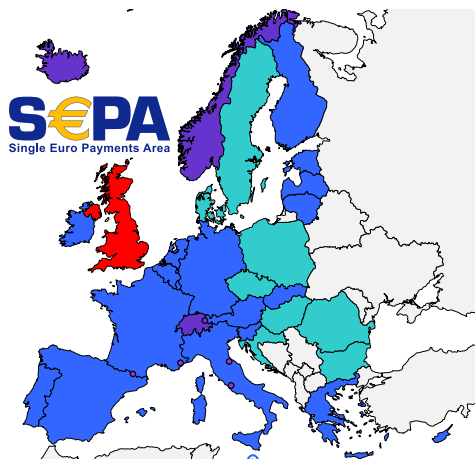
Way back, the EU decided that the costs and difficulties of making cross-border payments in euros were a restraint on trade (and a profitable business for banks). The banks were instructed to make cross-border euro payments as simple and as cheap as domestic payments.

A body called the European Payments Council (EPC) was formed and came up with a set of rules for making euro payments within what was to become the Single Euro Payments Area (Sepa). These rules cover credit transfers, direct debits and card clearing. You can find more than you will probably ever want, or need, to know about Sepa on the EPC website at europeanpaymentscouncil.eu/about-sepa.

Common language: XML

What interests us (well, me – and, I hope, you) is that the transaction interchange must use XML messages based upon ISO 20022, the international standard for financial message formats. These ISO messages are very generalized, and for Sepa there are various restrictions on the values that may be used.

Sepa messages may only be in euros and not any other currency such as pounds



*Sepa's member countries – not all are EU members.
Licence info: archivemag.co.uk/copyright.php#2406*


sterling. On top of that, each country has its own format for bank account numbers. In Sepa messages the International Bank Account Number (Iban) must be used rather than any other account identifier permitted by ISO 20022.

Documentation online

From the ISO 20022 website you can find the schemas (.xsd files) which define the various messages. And, in a semi-comprehensible way, the messages and their flows are described in Message Definition Reports – MDRs. Just to complicate matters, ISO issues periodic updates to the standard, and Sepa is still based on an old version of the standard. (In time, Sepa will catch up and be based on a more current version. I am glad I will not be working on the associated changes.)

Sepa uses messages from the Pacs range (Payment Clearing and Settlement) for bank-to-bank messages, and from the Pain range (short for Payment Initiation) for customer-to-bank messages.

As a customer if you want to initiate a collection – i.e. collect money by direct debit – you need to create an XML file containing a pain.008 message. For Sepa the version you need is pain.008.001.02 (ISO is currently up to pain.008.001.07).

For the full details we can go to the web as everything is publicly documented. The URLs are rather long, so I have sent them to our editor in a file that you can download and just click.  From the ISO 20022 archive we can get the ISO pain.008 schema and the corresponding MDR. (It is large but covers several messages, so be only slightly frightened by the size of it.)

From the EPC Document Store (formerly the Knowledgebank) you can find the latest version of the Sepa pain.008 schema and an implementation guide. The latter is the key document as it is, by and large, readable. But naturally, if you are creating a real file of collections you do so the way your bank tells you.

Schema


Ah, “schema”. Let me digress. If familiar with XML we should know that `<LclInstrm><Cd>CORE</Cd></LclInstrm>` looks valid, whereas

`<LclInstrm><Cd>CORE</LclInstrm></Cd>` does not – the closing tags are out of sequence. But is CORE a valid value in this context? This is where a schema helps us.

An XML schema is a set of rules for validating an XML file. In a schema you can define things like the sequence of elements, whether elements are optional or repeat and the format of elements. If you have an XML file that conforms to a properly-created schema then you know that you are working with high-quality input data. Schemas cannot do conditional validation (if element *X* has value *Y* then some condition must apply to element *Z*)

or perform complex computations like check-digit algorithms, however. We will run into both of these later on.

What the Pain file looks like

What does a pain.008 file actually look like? Paul Porcelijn gave us an example so we should look at his file DDtest/xml. *[Files that go with Archive 24:3 are still downloadable from the Archive website. —Ed.]* You can open this in any text editor (shift-doubleclick or drag the file onto your editor’s iconbar icon). 

We now see four components:

- A file header which identifies this as an XML file.
- A group header which identifies who created the file. For the creditor bank this is who submitted the file (and should be contacted in case of problems).
- A batch header called “payment information” (PmtInf in the XML) which essentially identifies the account to be credited for the direct debits.
- Individual collections, i.e. the direct debits. Each is given an “end-to-end” ID.

The file header

The file header is the first two lines. The first line tells us that this is an XML file. The encoding is UTF-8 so any character from the Unicode character set may be used.

Most data elements in the message have schema definitions which restrict their values. But data elements such as names are restricted only to a maximum length, so accented characters may be used, or Greek for that matter. However, we also have to consider what is allowed

- by the ISO schema;
- by your clearer (which, for Paul Porcelijn, would normally be Equens for Dutch domestic direct debits and EBA Step2 for cross-border direct debits);

- by your bank (i.e. the creditor agent);
- by the debtor's bank; and
- by any other intermediary banks.

Any of these might replace “odd” characters with a default, such as a space or a fullstop or, in extremis, just reject the collection. The only safe characters, the Swift character set, are uppercase and lowercase letters, numbers and 11 others: / - ? : () . , ' + and space.

The group header

The group header gives information about the overall file. You may include a control sum of the total of all collection amounts in the message – but whether your bank will check this is another matter.

The batch header

The batch header tells us about the originator of the collections – who the creditor is and where they want the funds credited. In Paul's test file the account has Iban NL53ING000B1234567.

Normally the account statement shows one credit for the total of all the collections in the batch. The corresponding reference on the statement is the payment information ID from the XML message.

Any unpaid collections will show as contra debits – their statement reference should be the end-to-end ID from the original collection.

There is no Sepa requirement for any of these identifiers to have unique values, but they do help to avoid confusion.

The creditor ID identifies who is issuing the collections. Your bank will help you obtain one when you make arrangements to issue Sepa collections.

The individual collections

Together, the mandate ID and the creditor ID identify the agreement to make collections. If there is any dispute, these

lead back to the underlying contract.

The debtor's bank is not identified, even though Paul's test data shows he has the values (see column AU in MakeDD/xls). This is OK within the EU, but if Paul had a Swiss member (Switzerland is a member of Sepa) he would need to include the debtor's BIC unless his bank can determine it automatically. Instead of

```
<DbtrAgt><FinInstnId>  
<Othr><Id>NOTPROVIDED</Id></Othr>  
</FinInstnId></DbtrAgt>
```

we would see

```
<DbtrAgt><FinInstnId>  
<BIC>CBSWCHZZXXX</BIC>  
</FinInstnId></DbtrAgt>
```

(for a debtor with Citibank in Zurich).

Crossing borders

The BIC is the Bank Identifier Code. These are issued by Swift, the Society for Worldwide Interbank Financial Telecommunication. You will also find your BIC on your account statements.

Looking at the test data, we see that Paul is making collections from Belgium – the EU's idea of making cross-border collections as simple (or as complex) as domestic collections has worked.

Could Paul make a collection against a UK account? Of course he could – and he still can! Back in 2017 the UK was an EU member and so was automatically a member of Sepa. EU membership is not a requirement to be a member of Sepa – Switzerland and Monaco are members, for example. The UK is still a member of Sepa despite the Brexit shenanigans. The UK bank will convert the collection amount from euro to sterling and debit that amount plus any charges from the debtor account. The UK Direct Debit Guarantee will not apply, but similar protections are available through the Sepa Rulebook.

Could Paul make a Sepa collection

against a US account? No – the USA is not a member of Sepa.

Membership database

With all that background the technical problem Paul faced should now be clear. How do we take a membership database in a “sensible” format such as a CSV file and create an XML file of collections? In order for Paul’s bank to accept the XML file for processing it has to, in order of complexity, be valid XML, be schema-compliant and meet various other constraints. Other constraints? That the explicitly-specified number of transactions (`GrpHdr.NbOfTxns`) really does match the number of `DctDbtTxInf` elements, that the requested collection date is within a valid range, that the Ibans’ check-digits verify and so on. A schema cannot enforce any of these things.

RiscOS hero

I think Paul Porcelijn is a brave man to work with XML files within RiscOS. I haven’t looked very hard but I don’t know

of any RiscOS applications that work sensibly with XML files, let alone schemas. (Yes, StrongEd has an XML mode but it appears fairly basic in what it delivers.)

On Windows I still use the free editor Notepad++ with the XMLTools plugin for almost all my work with XML files. No doubt there are other tools that do the job, but Notepad++ did what I needed (and was permitted software where I worked).

When I needed to look into schemas I used XMLSpy – an incredibly powerful tool (and quite costly) – but I only ever scratched the surface of its capabilities.

In the financial world XML-based messages are taking over as the mechanism of choice for delivering a whole range of services. RiscOS, unfortunately, lacks the software infrastructure to take part.

Richard Wheeler archive@maric.info



Richard has both retired and moved house. The new garden and now two one-year-old grandchildren mean that all those RiscOS projects to be done once retired are still waiting!

hint&tip Write XML without writing XML

I’ve written and produced a lot of XML. I used both verbs there for good reason.

At the start, I wrote XML “by hand” – anglebracket by anglebracket. For anything other than the simplest messages, you have to be very diligent, running your output through your validators on every edit, or at some point you’ll certainly slip and break something. If you’re embedding arbitrary data, especially data from users, you also have to be very careful to make sure it’s “escaped” properly, which is easier said than done.

A less error-prone approach to XML, and the one I heavily favour, is to produce it programmatically, such as from a script. Many common scripting languages such as Javascript, PHP and Python have support for this built in. It seems slow and verbose at first, but the certainty of valid XML is well worth the learning curve. You also get the ability to loop through any input data, and perform other tricks. For bonus points, have the same script automatically validate its own output against the schema!

There’s a middle-ground option: to write by hand while avoiding the gnarly XML syntax. To do this, use an intermediate markup language like Haml (haml.info), which is much easier to read and edit and will happily convert to XML.

Bart Nagel bart@tremby.net

Adding lines to a drawfile

Chris Hall

Let us say that you have a drawfile, created by exporting it from another application. You want to add some precise lines to it but otherwise leave the file as it is. One method is to load the drawfile into the **Draw** application, add the lines and save it again. There is a limit to how precisely you can draw curves freehand.

The example I am going to describe adds some lines to a map exported from **RiscOSM**. The lines correspond to a route traced out in the **Anquet** mapping software and saved as an Anquet Export format file, such that the points along the line are given as OSGB36 National Grid references, accurate to the nearest metre.

The map will be the size of a credit card and will thus be at a small scale (1:1,500,000). I will therefore omit minor roads and place names. The exported map is a vector graphic, a drawfile, and the area it covers is precisely described in terms of the latitude at top, centre and bottom and longitude at left and right. Scaling is linear with longitude and with a function of latitude, namely

$$vert = \log(\tan(lat) + (1/\cos(lat)))$$

so that the grid reference of any point can be processed to calculate the corresponding latitude and longitude.

The first step is to create the map background. Using **RiscOSM** with the scale set to 1:1,500,000 and the map size set to 89×65mm, I chose the area I wanted to show and exported a drawfile both with and without map coordinates. The resulting map background is shown to the right.

I realized that the application **Makedraw** was not capable of incorporating an existing drawfile,

just creating one from scratch. It would allow a simple drawfile to be “disassembled” (using **Multitask**) and then recreated but this approach was not suitable here.

I therefore added a few new commands to **Makedraw**:

```
PROCmdInclude("filename")
```

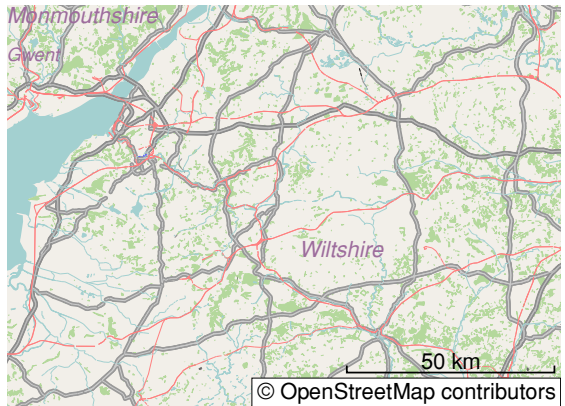
```
PROCmdQuadrantATo(x2,y2)
```

```
PROCmdQuadrantCTo(x2,y2)
```

so that I could add some lines, including a rectangle with rounded corners (see the program segment on the opposite page).

The “include” command examines the file to be incorporated and checks that any fonts it uses have already been correctly defined and have the same font handle in the file that has been built up so far. If not, it prints a warning message.

Makedraw already has a command `PROCmdCircle(x,y,r)` which will draw a standalone circle as accurately as the Bézier curve method allows. Adding rounded corners is something that can already be done, although not elegantly.



To produce this map, I made a map style where railways were shown as thin red lines. I then used the features menu to remove railway stations, minor roads and town names.


```

40 LIBRARY " <MD$Dir>.MakeDraw"
70 PROCmdInitialise(8228192,0)
80 PROCmdSetUnits(mdMM%)
100 REM The drawfile uses font02%
    and font01% so these must
110 REM be defined to match before
    including the existing drawfile
810 font02% = FNmdDefineFont(
    "Homerton.Medium.Oblique")
820 font01% = FNmdDefineFont(
    "Homerton.Medium")
825 REM Now add the fonts I need
830 font03% = FNmdDefineFont(
    "Trinity.Medium")
870 PROCmdStartGroup
880 PROCmdInclude("Wiltshire6")
890 REM Map centred at
    51.34004 N, 2.11497 W
900 REM Includes latitude from
    50.90452 N to 51.77146 N
910 REM and longitude from
    3.0663 W to 1.16364 W
920 REM Drawfile bounds correspond
930 PROCmdDiagramInfo(s%,l%)
940 REM Bounding draw units of
    whole file
950 left%=s%!24:bottom%=s%!28:
    right%=s%!32:top%=s%!36
990 REM Bounding lat and long of
    whole file
1000 LatBV=50.90452:LatCV=51.34004:
    LatTV=51.77146
1030 LatB=FNvert(LatBV):
    LatT=FNvert(LatTV)
1050 LongL=-3.0663:LongR=-1.16364
1070 xorg=0:yorg=0
1170 REM Initialise the conversion
    parameters
1180 PROCinitGB
1185 REM Calculate the LH easting grid
    reference (gx1%) at mid height
1190 PROCConv(LongL,LatCV)
1200 gx1%=xgrid
1270 REM crop marks
8950 DX=3:DY=10
8970 PROCmdLine(DX,DY+55,DX,DY+58):
    PROCmdLine(DX,DY-12,DX,DY-15)
9000 PROCmdLine(DX+83,DY+55,DX+83,DY+58):
    PROCmdLine(DX+83,DY-12,DX+83,DY-15)
9020 PROCmdLine(DX-3,DY,DY+6,DY):
    PROCmdLine(DX+86,DY,DY+89,DY)
9040 PROCmdLine(DX-3,DY+52,DX-6,DY+52):
    PROCmdLine(DX+86,DY+52,DX+89,DY+52)
9080 REM Credit card outline
9350 DX=3:DY=10
9370 PROCmdStartPath(DX+40,DY):
    PROCmdLineTo(DX+80,DY)
9390 PROCmdQuadrantAto(DX+83,DY+3):
    PROCmdLineTo(DX+83,DY+49)
9410 PROCmdQuadrantAto(DX+80,DY+52):
    PROCmdLineTo(DX+3,DY+52)
9430 PROCmdQuadrantAto(DX,DY+49):
    PROCmdLineTo(DX,DY+3)
9450 PROCmdQuadrantAto(DX+3,DY):
    PROCmdLineTo(DX+40,DY)
9470 PROCmdClosePath:PROCmdEndPath
9490 PROCmdRectangle(DX,DY+17,37,18,0)
9510 PROCmdEndGroup
9520 PROCmdSaveDiagram("Test")
9530 END

```

This program initializes memory to contain a drawfile, then uses the PROCmdInclude command to examine an existing drawfile and load it into the memory buffer (see line 880). The font table in the drawfile created in memory by lines 810–830 is checked to make sure it exactly duplicates that in the drawfile being loaded (and gives a warning message if not). Finally the outline of the credit card shape is drawn onto the map.

The commands:

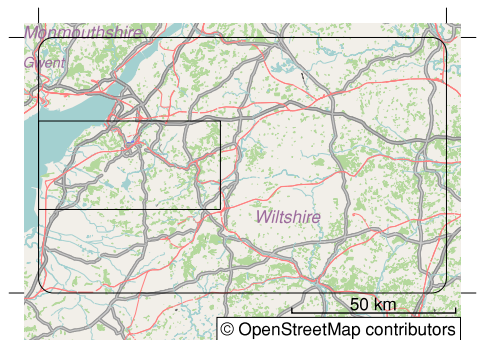
```

PROCmdLineTo(x1,y1)
PROCmdCurveTo(x2,y2,x1+(x2-x1+y2-y1)/
    3.62,y1+(y2-y1+x1-x2)/3.62,x2+(x1-
    x2+y2-y1)/3.62,y2+(y1-y2+x1-x2)/3.62)

```

will draw a straight line to (x1,y1) and then an anticlockwise quadrant to (x2,y2). Note that no trigonometry is needed, just simple arithmetic.

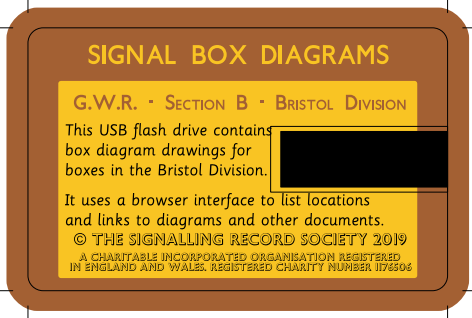
The picture at right shows how the printed image will look on the target USB flash drive (packaged at the size of a credit card) with just crop marks and the outline added using the program above.



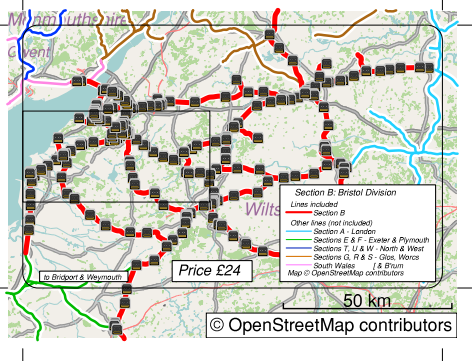
The map with lines overdrawn to show crop marks and the outline of the credit card.

Adding lines to a drawfile

Now to add a few railway lines and signalboxes as well as some explanatory text, and the artwork is complete. The outline of the card is not included in the finished artwork, just the crop marks for alignment purposes.



Finished artwork for each side of the USB flash drive, which flips out to plug in. The contact face (the black rectangle above) can't be printed.



Importing the edited drawfile into **Artworks** gives all vector graphic lines a CMYK process colour, but it currently oversaturates by not using the K (key, or black) component – a known bug in Artworks. The “correct” conversion formulae are these:

$$r = \frac{R}{255}; g = \frac{G}{255}; b = \frac{B}{255}$$
$$K = 1 - \max(r, g, b)$$
$$C = \frac{1-r-K}{1-K}$$
$$M = \frac{1-g-K}{1-K}$$
$$Y = \frac{1-b-K}{1-K}$$

I found a useful facility in Artworks which allows the colour table to be exported. It can then be examined and shows that all of the imported RGB colours from the drawfile have been given CMYK process colours:

```
%%AWColourTable:
...
%%+c p (New colour 90) 0.32 0.6 0.92 0
%%+c p (New colour 91) 0 0.47 0.11 0
%%+c p (New colour 92) 0.36 0.62 0.8 0
%%+c p (New colour 93) 0.02 0.23 0.86 0
```

Note that the K component of the colour (the last value) is zero in all cases.

Assuming that the conversion from RGB to CMY was otherwise correct, we need to set K to the lowest value out of C, M and Y, then complete the conversion with the formulae:

$$C' = \frac{C-K}{1-K}$$
$$M' = \frac{M-K}{1-K}$$
$$Y' = \frac{Y-K}{1-K}$$

A few minutes with **Scicalc** will produce an amended colour table with values like these:

```
%%AWColourTable:
...
%%+c p (New colour 90) 0 0.41 0.88 0.32
%%+c p (New colour 91) 0 0.47 0.11 0
%%+c p (New colour 92) 0 0.41 0.69 0.36
%%+c p (New colour 93) 0 0.21 0.86 0.02
```

The amended table can then be simply imported back into Artworks to make the necessary corrections. A proof-ready PDF can then be exported and sent to the printer.

Makedraw update

Makedraw has been updated to version 2.58 to add the extra functionality described.

Chris Hall chris@svrsig.org

A moral tale, tasting of Pi

Gavin Wraith

A short note, which could be taken as a review, or more usefully as a warning not to jump to conclusions when hardware goes wrong.

In 2012 I bought a 23-inch LG Flatron monitor (cnet.com/products/lg-ips235v/review) to use with a Raspberry Pi. It has no speakers but it shows a really clear picture.

I was very happy with it, until it started going blank for no apparent reason. The picture would return if I sneezed or snapped my fingers.

After a bit of experimentation I noticed that disturbing the HDMI cable from Raspberry Pi to monitor would bring on the blanking. I quickly concluded that something was loose inside the six-year-old monitor. Eventually the blanking became so persistent that I determined to buy another monitor.

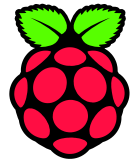
Raspberry-sized

I thought it might be good to try a smaller, more portable, monitor. I wanted one that used the Pi's default 1920×1080 resolution, and chose a 11.6-inch Wimaxit (amazon.ca/WIMAXIT-1920X1080-Portable-Compatible-Raspberry/dp/B076M3RL51) costing £99. It had a beautiful clear picture, but after an hour's use I noticed an irritating high-pitched whine, barely audible, like a nocturnal mosquito inside the net, emanating from its speaker. I could not live with that.

Before sending it back I noticed that it too would go blank if the HDMI cable was disturbed. This made me suspicious. Then, when I jiggled the Raspberry Pi, enlightenment struck. The culprit was not



Neither the old Flatron (above) nor the new Wimaxit (below) was to blame. Cheaper: a new Pi with a non-jiggly socket.



the monitor but the Pi's HDMI socket.

Now, with a different Pi, the good old faithful LG monitor has been reinstated and appears to be working without fault. It is much cheaper to replace the Pi than the monitor.

Leap slowly

The moral is: do not jump to conclusions without exhausting the possible causes.

Gavin Wraith gavin@wraith.plus.com



Words please Gavin. An old friend once asked him if he would like Annick Payne's book on Hieroglyphic Luwian (*Subsidia et Instrumenta Linguarum Orientis*, no. 2) but he already had it.

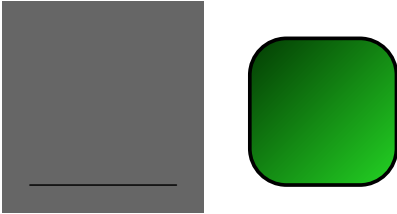
A glass button, in eight steps

Steve Royd-Marker

These drawings were made on a Raspberry Pi using RiscOS and Artworks. *In each illustration, the left side shows the view along the plane of the button (and other artwork to clarify) and the right is our in-progress drawing, showing the button's face.*

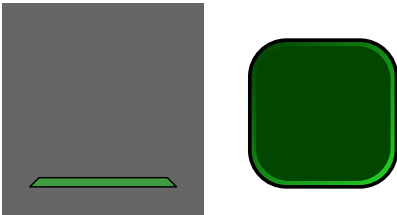
Step 1

To start your drawing, draw your shape with a background colour (somewhat darker than the desired colour) and give it a graduated fill from the darkest to the lightest colour you'll be using. This will already determine where the light is coming from. *The darkest side is the side where the light comes from and the lightest side is in the direction in which the light shines.*



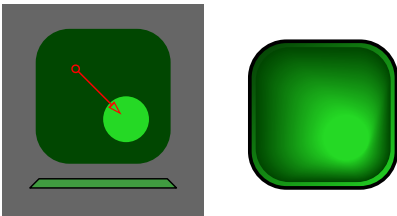
Step 2

Copy the shape and make it somewhat smaller, so the button can appear to have a chamfer on its edges. Use the darkest colour of the previous shape and make it a solid fill. Remove the line colour. *The button will also look good without step 1's shape. It all depends on the purpose of your button – your choice.*



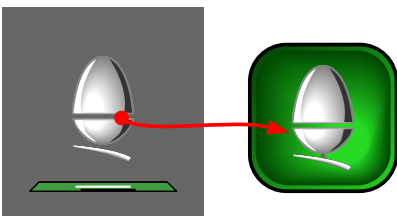
Step 3

Next, draw in the lensing effect of the rounded glass, where the light gets concentrated on one part of the button's background. This is at the opposite side from the light's source. Draw a round shape and give it the lightest tint, then blend it with the background shape in about 50 steps. If your button is very large you may need more steps to avoid banding.



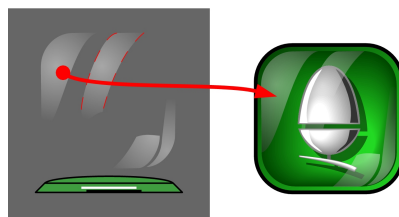
Step 4

Add the object you wish to display on the button and give it a shade so it looks like it is not completely stuck at the back of the button. If giving it a shadow, you need to be aware of the light to offset it the same direction as step 3. The light angle is important when drawing in further light and reflection effects, to make the button look more like glass.



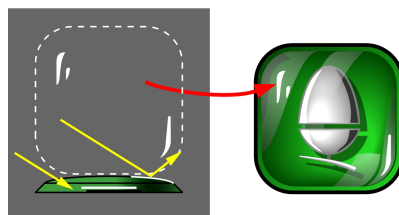
Step 5

To give the button its round appearance, add reflections. It's important to bend the sides of the shapes – the curve really defines the profile of the button and with straight edges it will appear flat. They're made from parts of the original button shape, with graduated fills from black to white and transparency set to 75% in “bleach” mode.



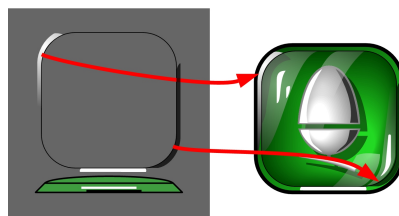
Step 6

To give the button a more lifelike appearance you can draw in a hard reflection, made from previous shapes. Give it a solid white fill and make it much smaller in size. The light that has been reflected in step 5 and 6 on the button's surface can no longer shine on the background to leave a shadow, so reflections always sit on top of the darkest parts of the button.



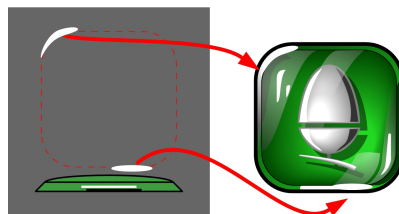
Step 7

You may want to draw in side reflections to bring more attention to the chamfer if you have one. Make a shape to fit the chamfer with a graduated fill from black to white. Be aware of where the light comes from. At the opposite side (close to the lightest side of the button) it's the same, but with a solid black fill and 30% transparency, giving the button extra depth.



Step 8

Sometimes, reflections are so intense that they take more visual room than their physical sizes. This is especially so with photography where the CCD or traditional film can get overexposed. Depending on the size and the purpose of your button you may wish to leave out steps 7 and 8. The smaller and more functional your button, the less detail is required.



Steve Royd-Marker info@markerdesign.be



I've become quite familiar with Artworks. But as with all things you often need to know the “trick” of how to build up an image, to make it appear a certain way.

[Steve's original Artworks file is available on the website for you to download and play with. —Ed.]

Files:
see p.2

Debugging tools from Spain

Bernard Boase

A surprise stand and theatre talk at the 2019 RiscOS London Show came from the University of Cantabria's computer science lab in Santander. Academic staff member Pablo Fuentes described and demonstrated his faculty's teaching aid designed to help computer science students understand and debug assembly language code.

Their reasons for choosing the Raspberry Pi model 1 (single processor core) for this application included:

- Low price and wide availability so a student can have one at home
- Can run standalone with an OS
- Simplicity and ubiquity of the Arm instruction set
- Good specifications of input/output
- No interference from other processors

And for choosing RiscOS:

- Open-source
- Good graphical user interface for interaction
- Direct handling of input/output with low interference from the OS
- Drivers can be tested on the go
- Widely used tools (GCC toolchain)
- Can mix assembly language and high-level code

But the available debugging tools were judged insufficient:

- GDB (the Gnu Project Debugger) could not be made to work under RiscOS.
- The native RiscOS Debugger module did not seem intuitive and had no GUI.
- The Desktop Debugging Tool (DDT) had too expensive a licence for students' home use.
- BreakAid by Rick Murray requires presetting of breakpoints.

!UCDebug

So the answer was to write their own tool in C (for the GUI) and assembly language (for execution of users' code). In use, the application has these windows:



1. Console for commands to UCDebug, changing memory and registers, displaying error messages
2. Instructions in the programmer's code (Arm assembler and machine code) with ability to set and unset breakpoints
3. Memory displayed in words, half-words or bytes (appropriately ordered just like in StrongED's Dump mode)
4. Registers, CPSR and flag mnemonics (changed register contents display in red)

Plus, optionally, windows containing:

- Either the single- or the double-precision VFP (floating point) registers
- The user program's output (STDOUT)

Observations

Q&A at the show emphasized that this debug aid is for student learning, and is not presently aimed at developers who may need another level of functionality, for example a trace file. Also, I have not been able to use this app in practice, so I think there's another article to be written about its hands-on use.

UCDebug displays only ELF (type `&ELF`) executables, and they have to have been linked at the address `&00018000`. (The displays use the `0x` prefix rather than `&` to indicate hexadecimal.)

Since the app has been written with limited knowledge of RiscOS conventions, there are several aspects of its UI that are some way from everyday practice, not to mention from the style guide. One

```
Dbg>help
List of commands:
ld file_name      load the assembly file onto the debugger
tr                execute a single instruction
tr <addr>         execute the single instruction in <addr>
go               execute from the first instruction
go <addr>         execute from the instruction in <addr>
gf              fast execution (beta mode)
gf <addr>         fast execution from <addr> (beta mode)
gd              execute from the first instruction
              without stopping at the breakpoints
gd <addr>         execute from the instruction in <addr>
              without stopping at the breakpoints
gt <addr>         execute without stopping at breakpoints
              until <addr> is reached
br              list the breakpoints currently set
br <addr>         toggle a breakpoint at <addr>
clr             clear all breakpoints
stop           stop the execution
set [s>] <addr>=<val> replace [s>] bytes of memory content
              (default=32bits) at <addr> with <val>
reg <reg>=<val> replace the value of the register
mem           reset the memory view window
mem <addr>    reset the memory view window, starting
              from <addr>
code          reset the code view window
code <addr>   reset the code view window, starting from
              <addr>
Dbg>|
```

```
0x00018088 0xb28D80E4 ADDLT R8,R13,#8E4 ; -"h"
0x0001808C 0x61DB91DF LDRVSSB R9,[R11,#31]
0x00018090 0x0B9D0BAD BLEQ &FE75AF4C ; *** Unpredict
0x00018094 0x7DC45F81 STCVCL CP15,C5,[R4,#516]
0x00018098 0xb746D482 STRLTB R13,[R6,-R2,LSL,#9]
0x0001809C 0x19CF747E STNNEIB PC,[R1-R6,R10,R12-R14]^ ; *** Not
0x000180A0 0x04BC7878 LDREQC R7,[R12],#2928
0x000180A4 0x9E9DA5B0 LDMIB R10,[R0,R2,R3,R8,R9,R11,R12,R14]^
0x000180A8 0x7376A63F Undefined instruction
0x000180AC 0x284E0839 STMCSDA R14,[R0,R3-R5,R11]^
0x000180B0 0xc0755985 LDRGTH R15,[R5],#-149 ; *** Rd=Rn
0x000180B4 0xA503D527 STRGE R13,[R3,#-1319]
```

② Code, here with a breakpoint set (highlighted).



R0	0x2316FB1F
R1	0x0003233C
R2	0x00032204
R3	0x00019000
R4	0x00032380
R5	0x00032330
R6	0x0003558C
R7	0x00036A58
R8	0x00011DA8
R9	0x0003922F
R10	0x04006208
R11	0x04006ED4
R12	0x00000003
R13 (SP)	0x000317FC
R14 (LR)	0x00008AFC
R15 (PC)	0x0001098C
CPSR	0x80000110
FLAGS	N----
CPU Mode	usr
SPSR	

① Console, with the list of commands given by help.

```
0x00019000 CF5324C6 EAEB6CB4 D540C7BE B2F2DDFC ;$.S...l...0...
0x00019010 4F639877 DF3B67A6 E7782EC2 4ED78482 ;w.c.o.g;...X...N
0x00019020 1AA2FB4B 9A050EAF A2DACDCD 7B948BC9 ;K.....{
0x00019030 0EDF2DBF 0C4946D3 EF847708 48C608F9 ;...FE..w....H
0x00019040 6DEAC153 0B73F871 CDFC40D4 8462E809 ;S.mq.s..@...b.
0x00019050 B663CBA7 AAFFAA9E 3D2E77B2 D7033670 ;.C.....w.p6..
0x00019060 F7C1C58A 5F6ABAAC E82EAA1A E37A175E ;.....^..z.
0x00019070 F74645BE 8DF7F0E3 FFB66EC2 B602F1DE ;.EF.....n....
0x00019080 7377C77B D2F74273 4501A25D E3BF1600 ;(.wssB...J..E...
0x00019090 CC2A04D4 74C31AE7 DB2F31DC C5DE200F ;...t..l/..
0x000190A0 D73DA59C A0CE7FCE 686070DC 54277671 ;.....p'hqv'T
0x000190B0 177E2B4B 86DA9912 51B55775 2C2FD666 ;K+.....u.W.QE./..
```

③ Memory, here showing a snapshot from address &00019000.

example is the use of the Menu (middle) button to select and clear breakpoints and to switch between double precision floating point register displays. Another is that some of the “close” icons also quit the app. On the other hand the Select (left) and Adjust (right) buttons are sensibly used for scrolling forward and backward through the code and memory displays.

Tests have shown that it also runs on Raspberry Pi models 2 and 3, and there is a plan to make the app available from Packman.

Download

The application is open source and available from [fuentesp.github.io/UCDebug](https://github.com/fuentesp/UCDebug). You will need to give its Absolute the file type ELF (&ELF) for it to load, and not be

running Aemulor. It is under active development, and v1.8.5 was released on 2019 December 2. Pablo will be pleased to receive any users’ comments by email at pablo.fuentes@unican.es.

The paper *Tecnología low-cost para motivar al alumno* (“Low-cost technology to motivate the student”) by Fuentes et al is in Spanish at [aenui.net/ojs/?journal=actas_jenui&page=article&op=viewArticle&path\[\]=502](https://aenui.net/ojs/?journal=actas_jenui&page=article&op=viewArticle&path[]=502), where an abstract in English is also given.

Bernard Boase b.boase@bcs.org



Bernard is sad to have so many excursions this year cancelled, but is otherwise finding lockdown not so very different from quiet retirement with the usual amount of RiscOS.

Network hardware

Alan Adams

The aim of this article is to explain the various types of network hardware, what they do, and what they don't do. In order to explain this, some elements of networking in general – and TCP/IP in particular – will need to be explained.

Seven-layer notion

Networking software uses a concept called a **protocol stack**. The idea is that you can consider each component of a network as a slice or layer, and a complete network consists of seven layers. Each layer has a dedicated function, and on the local computer communicates with the layer immediately above and immediately below.

Two computers each have a protocol stack, and each layer of the stacks can be considered to communicate with their peer on the other computer.

The top layer is the **application layer**. End-user applications generally connect with the stack at this level, and this means that as far as the application is concerned it is communicating with the application on the other computer directly.

What actually happens is that the messages are sent from the application layer to the next one down, then the next, until they reach the bottom layer, the physical layer. The job of this layer is to send signals (bit-patterns) along a wire, a fibre, or over wireless. The advantage of the layer structure is that the application does not need to know which medium is being used.

TCP/IP (which stands for Transmission Control Protocol and

Internet Protocol) was born before ISO (the International Organization for Standardization) formalized the 7-layer ISO stack, and is usually considered to have four layers. They map pretty well onto single ISO layers or pairs of them, and the same ideas apply.

TCP/IP Model	Protocols and Services	OSI Model
Application	HTTP, FTP, telnet, NTP, DHCP, ping	Application Presentation Session
Transport	TCP, UDP	Transport
Network	IP, ARP, ICMP	Network
Network Interface	Ethernet, fibre, routers, switches	Data Link Physical

Mapping between the 4-layer TCP/IP model and the 7-layer Open System Interconnection (OSI) model.

We will be dealing with the lowest three layers, which are network, data link and physical:

- the physical layer, as I said earlier, is concerned with bit-patterns sent over a physical medium;
- the data link layer is concerned with sending packets to other devices on the medium;
- the network layer is concerned with sending packets to devices on networks, which may be the same one as the local computer, or different ones.

To see where this is leading:

- modems (serial, cable or ADSL) and repeaters work at the physical layer;
- bridges and switches work at the data link layer;
- routers work at the network layer.

Wireless access points and wireless network interfaces work at both physical

and data link layers.

Various equipment sold for home use combines these functions, so a “router” may contain a modem, a switch, a wireless access point, and it does of course contain a router.

Addressing

In order to send a message to another device it is necessary to have an address to send it to. (The exception is for point-to-point links, where there are only two devices, so all messages are for “the other guy”.)

The fundamental address is the **Mac address**. This acronym stands for Media Access Control, and is nothing to do with Apple computer products. It refers to the hardware address of the media access device, which in the most common cases are an ethernet interface or a wireless interface. Thus the Mac address is often an ethernet address. Every ethernet interface has a unique hardware-defined Mac address, shown as six pairs of hex numbers, such as 00:c0:03:ad:89:e3.

All data sent between devices on an ethernet network is sent to the ethernet address of the destination computer. There is also a broadcast form of ethernet address (ff:ff:ff:ff:ff:ff) which is received by all hardware devices. It is used, among other things, when a computer asks for a TCP/IP address using DHCP

[dynamic host configuration protocol: the system where a router automatically dishes out an address when a new device comes along—Ed.].

The transport layer sends and receives data using the Mac addresses.

Each computer will have protocol-specific addresses for each protocol it supports. Generally the only one is TCP/IP, so there will be an IP address. When a computer wants to communicate with

another computer using TCP/IP it puts a message into the stack using the IP address of the destination, and includes its own IP address for replies.

The network layer examines the message, in order to decide which network the destination is in. If it is the same network as the sender, it sends the message directly. However the message has to be sent to the Mac address, so first the equivalent Mac address for the IP address needs to be found. This is done first by checking the **ARP** cache (Address Resolution Protocol), and using the result if there. If it is not found, an ARP request is sent to the Mac broadcast address, basically asking “Who has this IP address?” The reply contains both the Mac and IP addresses of the target, and the result is put into the cache. (The ARP cache can be examined using the `*arp -a` command.)

If the IP address is in a different network then the message cannot be sent directly. Instead the routing table is consulted to find out a device that can pass the message on. This will be a router. The message is sent to the Mac address of the router, and the IP address of the destination. The router looks in its own routing tables, and either sends the message directly to the destination or to another router. In a home connection, it’s almost always to another router at the ISP.

Routing

How does a router, or a host, decide whether a message is for a computer on the same network, or a different one? With TCP/IP, it’s done using the **network mask** (also called subnet mask or netmask). This is a 32-bit number for TCP/IP v4, which is all we will consider here. IPv6 is not yet making its presence felt for home users, because NAT (see later) has bypassed the

problem to a large degree. The mask consists of a string of bits set to 1, followed by a string of bits set to 0. Where there is a 1, the corresponding bit of an address is part of the network number. Where the mask has 0, the corresponding part of the address is part of the host address. If the network numbers match, they are the same network. If not, the networks are different.

The addresses and masks are all 32-bit numbers. For convenience they are usually written in “dotted quad” format: four numbers separated by dots. Each number represents 8 bits, so has possible values from 0 to 255. Many routers have preset addresses of 192.168.0.1, with network masks of 255.255.255.0. In this case the network number is 192.168.0, and the host address is 1. Any messages for hosts whose addresses start 192.168.0 will be sent directly. If the first three octets don’t match 192.168.0 then the message must be sent to a router.

Because the bits of a netmask are contiguous, it is not necessary to write all the bits. All that is needed is to say how many there are. Thus you may see addresses like 192.168.0.7/24, which is the same as 192.168.0.7 with 255.255.255.0 as netmask.

There are two special addresses in any network. If the host portion is all 0s then the address is a network number, and cannot be used for a host. On a network starting 192.168.0 with 255.255.255.0 as its netmask, 192.168.0.0 is the network number.

The second is the **broadcast address**. It has all 1s in the host portion. For the case above it would be 192.168.0.255. A packet sent to this is received by *all* stations on the network. (255.255.255.255 is also a broadcast address. It should go to all hosts on all networks but, for reasons of protecting networks from flooding, is

blocked by most routers, and a lot of hosts ignore it.)

This brings us to a basic TCP/IP rule: to pass between networks a message must go via a router. The corollary is that a message passing through a router has to be to a different network.

There is a fundamental requirement that on a network which is supposed to be a Lan [*local area network*] (i.e. messages do not need to be routed), the network masks *must* match. If not, a host may decide to send a message directly, but the reply gets sent to the router; or a message is sent to the router, but the router thinks it should be sent directly. In both cases there will be no communication.

You may now be able to see why using two routers on a home network is not a great idea. If you use ports which the router thinks are on different networks, then those ports will have to have different network numbers. While you can design a home network like that, it is asking for trouble you really do not need.

What is NAT?

A router for home use generally performs one additional function, which is essential if you use it to connect to the internet, and breaks your network if you use it between parts of your home network – another reason not to use two routers.

The internet long ago ran out of TCP/IP v4 addresses. A 32-bit number has 2^{32} (around four billion) possible values. That might be enough for the current number of directly-connected computers. However, part of that number is network, and part computer. To handle large and small networks, a network number is issued from one of three ranges, called Class A, B or C. A Class A address has a network mask of 255.0.0.0, and there are only 192 of them, long ago assigned to big organizations.

(The rest are used for special purposes.)
Class A networks can have 2^{24} computers.

A Class B address has a netmask of 255.255.0.0. There are 65,534 of these, and a lot more companies than that want them, because they can have networks with up to 65,534 computers.

A Class C address has a netmask of 255.255.255.0, and there are 2^{24} of these, each capable of having 254 computers. These have also run out, because the organizations that could not have a Class A or B network used blocks of Class C addresses instead.

More recently classless networking became common. Put simply, the network part of the address does not have to end on an octet boundary, so you could have a 12-bit network mask, or a 22-bit one. This became possible only when the RIP routing protocol was abandoned, because that protocol made assumptions about the network mask. It is perfectly possible to use a network mask that is not a multiple of 8 bits on your home network, but doing so will confuse anyone you need to ask for help. It's better to stick to a standard one.

TCP/IP v6 was developed to solve the address shortage, but for home networks was overtaken by events. NAT (network address translation) allows a network (company or home) to appear on the internet as a single IP address, that of the router. This address is allocated by the ISP, which owns significant numbers of addresses. Furthermore, in the original address allocation, some networks were reserved for private use, and are always hidden from the internet.

These are Class A 10.0.0.0/8, Class B 169.68.0.0/16 and Class C 192.168.0.0/24 through 192.168.255.0/24. The latter is actually a block of addresses, which can be used as if it were a Class B network. These are the addresses normally used for private

networks, including home ones.

What happens with NAT is that each request going to the internet has both a target IP address and a port number. The port number is associated with a specific service, such as web browsing (80), email (25) etc. The router translates this pair so the router's Wan [*wide area network*] address is used. Thus the technology is more correctly referred to a PNAT (port and network address translation). When the reply comes back the reverse translation is applied, using the translation table set up by the outgoing request. It should be apparent therefore that there is an issue with incoming requests – there is no lookup entry, so the router does not know where to send the message. This is generally a good thing – it acts as a basic firewall, preventing any incoming connection attempts.

It is possible to pass incoming traffic to a designated server, or one designated server for each incoming port number. All incoming requests have to be to the router's own address, with the appropriate port. This has to be configured on the router, and security has to be considered carefully. This setup is often nicknamed as DMZ – “demilitarized zone”. The idea is to have a server in “no man's land” between the internet and the home network. The home network is isolated from attack, and the DMZ server is at risk. It can be managed from the home network without risking the rest of the network.

The hardware

The most basic element is a transceiver, almost never seen separately these days. It is the part of a network interface which handles the electrical or optical signals on the wire or fibre.

Modems are physical layer devices. They take packets sent to them from the

computer and translate them to the format needed by the medium, usually a telephone wire or optical fibre. They don't just repeat the packet, as they usually have to deal with different maximum packet sizes and different forms of addressing. ethernet uses a maximum packet size of 1504 bytes, while ATM, which underlies the telephone network, uses 576 bytes. As a result packets may have to be broken up and renumbered for reassembly.

Repeaters are usually multiport devices, and are more-or-less obsolete (except for gaming adapters, which are two-port repeaters). These contain at least two, usually more, transceivers (physical ports). Any pattern of electrical signal received by any transceiver is repeated blindly on all the other transceivers. For big networks, this results in serious inefficiency, as while one pair of computers is communicating, no other communication can take place. Some years ago 3-Com started marketing these as "hubs", and the term has stuck, causing confusion for those of us who grew up using "hub" as the place where the networking equipment was housed. A lot of people also refer to switches and bridges as hubs, compounding the confusion.

A smarter pair of devices are bridges and switches. Physically they look like each other, and like repeaters. They have at least two ports, usually many more. Home devices tend to have up to 8 ports, commercial ones 24, 48 or even 96. They both do the same job, in subtly different ways: they maintain a lookup table of Mac addresses and hardware port numbers so that a packet arriving for a given Mac address is only sent out on the port where the address is known to be found.

The lookup table is set up by listening. When a packet arrives from a Mac address, that address is added to the list

for the port on which it came in. Until an address has been learnt in this way, it is unknown, and packets for it are sent out on all ports. The size of the lookup table is limited by the memory in the device, and can become overloaded. When that happens the bridge or switch must revert to sending on all ports, reducing efficiency.

The gain in efficiency of a bridge or switch over a repeater comes from two things. Each packet received on a network interface has to be received sufficiently for the destination address to be compared with the local address, before it can either be accepted or dumped. In cheap interfaces, this can result in an interrupt to the CPU, which has to make the decision. Better interfaces handle this in hardware.

The second effect is that separate pairs of ports on the switch or bridge can be simultaneously supporting conversations. In big networks this happens a lot, for example where there are multiple servers. A 24-port switch could be supporting 12 different conversations at once.

Bridges differ from switches in the way they pass on packets. A bridge reads in the entire packet, stores it, then passes it on to the appropriate port. A switch can recognize the destination address while the packet is still being received, and start sending the packet out before it has finished arriving, reducing transit delay.

Both devices can handle ports operating at different speeds, e.g 10M on one and 100M on another. Obviously there is no problem receiving on 10M and forwarding to 100M, but in reverse the packet has to be buffered, because it takes longer to send than to receive.

Routers have already been covered, in some detail. They pass data between different networks, where the network is defined in the design of the protocols concerned.

Wireless devices

There are two basic wireless devices: interfaces, in computers, and access points, which provide the service.

Wireless routers are access points connected to routers, in one box. They frequently also contain switches (to give multiple physical ports) and modems (to connect to ADSL or cable networks).

Access points are designed to have a wired connection to the Lan and wireless connections to their clients. Data from multiple wireless client Mac addresses is forwarded from a single Mac address, that of the wired interface. Some access points can work in other modes as well, or instead. Others cannot. For this reason, using them in ways other than multiple client to one Lan can require careful reading of specs and manuals.

The most useful other mode is “wireless bridge with clients”. This allows a wireless access point (WAP) to link to the Lan via another WAP. For this to work, both need to support bridge mode. The WAP connected to the Lan will be receiving data from multiple computers but from a single Mac address. It needs to be able to handle this. The remote WAP needs to be able to establish a connection

to another WAP as well as conventional clients, and to forward traffic over it. In addition, if single wireless frequencies and network names (strictly, SSIDs) are used, there are issues when clients receive a wireless signal from both WAPs at the same time. It results in the client connection hunting between the WAPs, dropping traffic each time it switches.

To add a wireless connection to a computer without its own working wireless interface, **games adapters** are available. They work like access points in reverse, wireless to the Lan, wired to the client. Once set up they are able to initiate a wireless connection with a WAP without help from the host computer, thus not requiring special software or configuration on the computer. Ideal for RiscOS.

It would be interesting to try connecting a switch and multiple computers to one of these – it might fail, because the adapter would be receiving data from multiple computers on a port where the designers only expected one.

Alan Adams alan@adamshome.org.uk

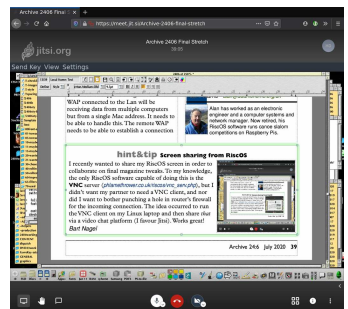


Alan has worked as an electronic engineer and a computer systems and network manager. Now retired, his RiscOS software runs canoe slalom competitions on Raspberry Pis.

hint&tip Screensharing from RiscOS

I recently wanted to share my RiscOS screen in order to collaborate on final magazine tweaks. To my knowledge, the only RiscOS software capable of doing this is the **VNC server** (phlamethrower.co.uk/riscos/vnc_serv.php), but I didn't want my helper to need a VNC client, nor did I want to bother punching a hole in the router's firewall for the incoming connection. The idea occurred to run the VNC client on my Linux laptop and then share *that* via a video chat platform (I favour Jitsi). Works great!

Bart Nagel bart@tremby.net



Thinking ahead...

Jim Lesurf

The basic principle of “copyrights” and “intellectual property rights” (IPR) is a good idea. It allows anyone who writes, plays music, invents something useful and so on, to gain some income from their good work. It also allows authors of such “works” to have the right to be identified as the person who created them and to prevent others from passing off degraded versions. Hence it is good for the reputation of someone who may want an income from producing further works of a similar kind. Note also that although the term “copyright” is common, “copyrights” is more accurate. So, for example, you can specify that you must be identified as the author or that people can – or cannot! – edit or alter the original when making copies available. The multifaceted nature of IPR is significant when considering what follows...

Unfortunately, during the last century or so, the scope and duration of the “protection” offered have both been expanded considerably. And alongside that a number of companies have tended to take over a lot of these rights. The combination of these developments has been an increase in problems – both for authors and for others who might want to access or use their work!

Hidden in plain sight...

Long-term Acorn and RiscOS users will already be depressingly familiar with one aspect of this: the way a lot of excellent old software has ceased to be developed and now can’t even be bought. For example, I’m still happily using an ancient copy of **Composition** (or “Compo” to its users) which was

developed by Rob Davidson and sold initially by Clares. However, newcomers to RiscOS probably won’t know much about it, or ever get a chance to *try* it, because it has become undeveloped unobtainium. Another example at the time I write this is a superb collection of fonts crafted and sold in the past by EFF. I still have the EFF CDs so can use the fonts, albeit with some difficulties because the supplied installer software is ancient. But recent adopters of RiscOS can’t even try the fonts because they aren’t on sale. As an ex-academic I’m also often frustrated by discovering that something I want to read is long out of print and unobtainable.

In addition, a number of old magazine titles and textbooks have ceased publication and often nobody now knows who owns the IPR for their content. Sometimes even the *owner* of the IPR isn’t aware of it. In other cases they do know but have no interest in allowing the work to become available again. Or in some cases they may have a grand plan... which may never come to fruition or succeed. The result is that a lot of useful information becomes difficult to find. Wheels then have to be re-invented and the reputations and intent of the original authors become forgotten. They get nothing; we get nothing. On a legal level it seems clear that this needs tackling at some point. But the good news is that many of us can DIY at least some partial solutions if we wish our own work to remain accessible...

Can you help?

Do you write magazine articles, or computer programs, or other “works” covered by IPR or copyrights? If so, the

key point is to think ahead. When people write for Archive they know it is a form of what tends to be called a “fanzine”. You don’t get paid, but you also remain free to reuse or republish what you wrote at some future date. Similarly, if you write some software and make it available in an open-source way, you can allow others to use, modify and reissue. There are various ways to approach this, but the key point is to realize that when you’ve generated a “work” it is up to *you* to decide its ultimate fate unless you enter an agreement with someone to hand them control over the relevant rights. And when making such an agreement you *can* specify which rights you hand over and which ones you reserve.

For example, many years ago I wrote two textbooks for the Institute of Physics Press. A part of the written agreement made with them was that I would be able to take back the rights to republish the material at such time as it went out of print, and so on. A few years later they decided to cease producing textbooks and they sold most of the rights for their catalogue to another publisher. However, I chose to take back the rights to my own books. I have now made one of them available for free on the web. A number of the articles I’ve written in the past for Archive have also been put up. And I intend to add more such items, including ones I’ve written for various other magazines. This way the information doesn’t cease to be accessible should back issues become impossible to find.

The central concern here is to avoid potentially useful material – be it a book, computer program, music, ingenious idea – from becoming “unavailable”. It may make sense for a commercial magazine or book publisher to put material into a vault and hope that one day someone will pay a fortune for it. But that does neither the

author nor those who would like a copy any good if it then resides, forgotten, in a vault which itself ends up lost and forgotten! In an ideal world the expiry term of copyright would be short enough that this would not be an issue. But in the real world these expiry terms have been grossly extended as the result of lobbying by large media corporations to suit them, not authors or consumers.

So what can be done?

That depends on the circumstances and your personal preferences, but the critical factor is to consider in advance what you’d like to happen. If you want whatever you may have produced to remain useful and accessible, then arrange for it. For example, write a clear and/or public declaration that when you pop your clogs others can freely copy or modify your work. Or ensure that whoever would inherit the IPR from you will use it as you would wish. Even simpler, make what you have produced openly available now! If you are producing material for commercial publication, see if you can make an arrangement where the material either reverts to you (and thus your wishes) after a given time, or you can reuse the work, or give the publisher non-exclusive rights. If producing software, consider an equivalent option of some kind.

Diversity is the spice of life

One other point to note is that although putting material on the web means it becomes accessible, remember that this only applies while that website is maintained. If material is to endure for longer, some form of open “diversity” is likely to be required, so that anyone who wishes can copy and republish, etc. Quite how you’d want to approach this is clearly a matter of personal choice. But the most

Thinking ahead...

important step is to think about it and to avoid good work becoming lost and unobtainable. Again, I can suggest one way to tackle this. For some years now I've been building a website for the UK Hi-fi History Society. A grand name for a dozen or so interested people who want information on this topic to be collected, conserved and to remain available. The central method has been to scan thousands of documents and put the copies openly and freely available onto the web. However, the next key step was to make clear that we encourage others to take copies and then make *them* available by whatever other means they prefer. This diversity makes it possible for the information to endure and remain available even when the original UKHHSoc site eventually disappears!

My personal view is that the timespan

of modern copyrights tends to be absurdly long. If you can't make an income from something being commercially published for, say, 20 years, it seems unlikely that having that extended to a century will help much. But we have a situation where big companies collect copyright material like greedy magpies for no other reason than to avoid older material "competing" with their newly promoted works. This does no good for the reputation or wealth of the creators of older works. Once a work is sold it may be too late. And once you're dead or doolally it may be too late to change your mind. So think ahead...

Jim Lesurf jcgl@audiomisc.co.uk



A fairly extensive "biog" of Jim's work and life can now be found at jcgl.orpheusweb.co.uk/history/ups_and_downs.html

hint&tip Plotting graphs by date in Fireworkz

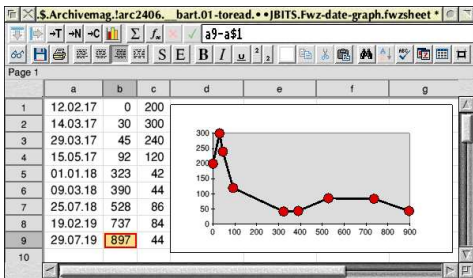
In a 2018 post on the Archive-online mailing list, Jim Lesurf asked how to generate a graph with one axis reading from non-uniformly-distributed date data, as opposite. The response came the next day, and the exchange was saved for a future hint&tip. —Ed.B

12.2.2017	200
14.3.2017	300
29.3.2017	240
15.4.2017	120

In **Fireworkz**, make a new column B and use the simple A1·A\$1 replicated all the way down. A\$1 means keep that 1 constant, while the A1 will increment on each line. So you'll see the result as the elapsed number of days since the first date. Plot accordingly, using one of the X-Y graph formats – **Fireworkz** will produce a drawfile. Use **Draw** to add a few date labels as necessary.

The naive "minus" to give difference between dates is obvious once you've thought of it, but I can find no mention of it in the **Fireworkz** user guide under Spreadsheet Functions (and emailed Stuart Swales, the author, to suggest it for next edition). I can't recall how I ever discovered it – maybe somebody told me on Archive-online!

Jim Nagel



Jim Nagel, 1944–2020

We put out a message to a few online forums saying, in short, that we didn't think we'd write something like the following article and that we were instead asking for anecdotes about Jim to be sent in for curation. This is one of the many responses we received. —Ed._B

I think it would be a shame not to include an article about Jim himself. I'm sure there is *much* more to Jim's life and personality than his RiscOS friends will have seen, and if it isn't too personal (and I appreciate it may be) I think a lot of us would really treasure an insight into the man outside of his interest in all things Acorn.

Andrew Rawnsley

Viola, Chris and Bart Nagel

As an Archive reader you may be familiar with Jim's unique approach to time. We recently found several overdue book notices from his local library postmarked 1957. These cards affirm a number of things about his character: he was always a reader, he'd always reuse and recycle rather than discard (these very notices had been repurposed into dividers for the Dewey Decimal Classification index in his home library) and that he was running late from the start. Combined with insatiable curiosity, a compulsion to keep working at something until it was absolutely right, and a very laid-back temperament, we eventually found that it was best to simply accept that it was a part of him and plan accordingly. This was all compounded in later years by emerging health factors.

But let's step back. Jim grew up in Canada. While studying for his degree in maths and German at the University of Waterloo, he worked on the university newspaper. He took a year out in Winnipeg to attend a church college where he met Viola for the first time. After graduation back in Waterloo he stayed on at the university paper full-time, now in charge as editor. Part of his legacy there was renaming the paper *The Chevron*, a title

which remained until the paper ended some 10 years later. He left this job after a few years to work on a local newspaper, the *Kitchener-Waterloo Record*, and after two more years set off for England. It was to be the first stop on a round-the-world trip, but also ended up being the last.

Although he had only one sister, being part of a sizeable extended family ("cousins by the dozens") could well have contributed to his penchant for connecting his various circles of friends, family and far beyond. This desire for all on earth to live in love and harmony was his passion, and inspired the years he spent in Glastonbury shortly after arriving in England in 1970.

He wrote, at that time, "London may be England's head, but Avalon is the heart: the crossroads of imagination, a place where young see visions and old dream dreams..."



Jim during editorship of the Chevron, circa 1968.

His dream was to incarnate “a worship-centred, land-based, warmly welcoming, rigorously radical new order”; a source of focus and inspiration were the Abbey Ruins at Glastonbury and the various monasteries he’d visited.

Initially taking the form of summer camps before later developing into a year-round project, for some 10 years he worked hard alongside others to realize this dream. Bishops, farmers, abbots and mystics (plus the necessary bank manager) featured in his correspondence, with the aforementioned “young” and “old” flocking to stay. Densely written and creative visitors’ entries are recorded in fat diaries of that era.

While working on the Christian community project he supported himself with jobs as diverse as apple picker, farm worker, warehouse worker, psychiatric ward assistant, sysadmin and teacher. But throughout, he was never without a pen.

In December 1982 he married Viola, a soprano and teacher of singing, and, as Viola sometimes put it to him, established “a new community”: soon there was a mortgage and two young boys, Christopher and Bartholomew. Jim embraced fatherhood with love, pride and creativity. Jim was a people person – and here were two fascinating, new little people. He later embraced grandfatherhood with the same enthusiasm.

After marrying, he worked variously as editor and writer (and sometimes photographer) at publications including

Somerset County Gazette, *The Times*, *Church*

Times, *Amstrad Action*, *Computer Shopper* and, of course, *Archive*. In 1991 Jim founded Abbey Press, which was involved with books and other printed matter in all combinations of typesetting, editing and publishing. Combining these with interminable creativity and a huge range of experience with work, travel and people helped to make him an excellent writer. Future Publishing’s founder once called him “probably the best writer on our staff”.

Jim has been described as a “linchpin of the community”, and it seemed as if everybody in town knew – or at least recognized – “Canadian Jim”. Some of his local involvement included the production of a newsletter for the Glastonbury Conservation Society, door-to-door collection for charity (always raising the most money with his charm and persistence) and having fun with amateur drama. He explored drama a little further in later years with a few extra and bit parts in films and TV shows. He enjoyed the



Jim during a visit with friends in 2015.

novelty of this, even growing out his hair to garner further opportunities.

Involved in the local church, he warmly welcomed any traveller or visitor and was an active member of the church council. He also produced a regular church newsletter and later contributed to the website and to the weekly digital newsletter. He adored singing in harmony – lending his voice to the church choir as well as other groups – and even performed with a choir on the Pyramid Stage at Glastonbury

Festival in 2002. He also took a keen interest in Chris and Bart's musical training and always supported his wife in her concerts and teaching.

Jim travelled widely – sometimes solo, but often with family – with destinations including Europe, Palestine, Africa and, of course, Canada. The last cross-Canada road trip, four years ago, stands out – just Jim and Viola on a leisurely three-week journey from Vancouver Island to Ottawa.

Jim was first diagnosed with prostate cancer in 2011, just in time for him to grow his moustache for “Movember”. Initial rounds of radiotherapy appeared effective but, shortly after returning from the final Canada trip, the illness reemerged in the form of “unwelcome visitors” – secondary cancers. Unrelated health issues dictated that the optimal treatments weren't viable, and those which were had side-effects of their own. He never complained – ever – and rarely mentioned it to others, finding much more significance in life, people, issues and the present.

In a session with his oncologist this last December, he'd said, in his easy way, “Well, let's speak of the elephant in the room: how long have I got?” He was reassured that it was time “measured in years, not months”. Chemotherapy, which he casually called “a new treatment”, started soon after.

Although he never suffered the nausea often associated with chemo, the treatment brought on a deep weariness that never fully abated. He and Viola spent a relaxed last week together in isolation due to Covid-19. He was gone when Viola tried to wake him on March 21. He died peacefully



The Nagels in 2018. Jim and Viola, sons Chris and Bart, plus grandchildren Tifa and Gus and their mother Ephie.

in his sleep. At home, in his pyjamas, in his own bed.

Jim has been described by his friends as a thoughtful, wise, charming and gentle man. His measured dry wit and mischievous sense of humour caused hilarity at many a dinner party. He had a probing mind with a gift for extracting – and recalling – interesting stories from anyone he met. He had a generous heart, with time for everyone. As one family friend put it, “his very presence brought warmth into a gathering”.

The Nagel family

Viola remains at Abbey Press HQ in Glastonbury, a ten-minute walk from Jim's new digs. Chris has a base nearby and works as a professional drummer. Bart is a web developer living in Vancouver, though he produced this mag with the others' help at HQ.

Anecnotes

It was always a pleasure to meet and talk with Jim at RiscOS shows.

Jim was great company and he gave life and levity to a number of the group's post-show meals.

Doug Webb

Jim was one of the first people in the RiscOS scene to make time to connect with my wife Steph (who came over from America, close to Canadian border, when we married). She was rather thrust into RiscOS shows at the deep end, without really knowing anyone. They bonded over tales of Tim Hortons coffee and doughnuts! She would then make him tea and he'd show her photos and tell tales of Canada and Nagel-land etc.

From my own perspective, I was always struck by how helpful, friendly and keen Jim was. As a supplier, he was always interested in what we were up to, and made a real effort to support us via the magazine. I always really appreciated this, because the RiscOS scene needs to pull together like this and support each other.

Andrew Rawnsley

I never met Jim but his personality certainly showed through in his writing, both in the magazine and online. I don't think it needs to be said that he will be greatly missed, that's pretty much a given.

Rick Murray

I remember speaking to Jim the day before one of the shows. He was saying, "I wonder if I can find a printer that will be able to print the magazine tonight". I was just thinking, "this is probably just a tiny bit too late".

Chris Hall

Chris doesn't recall whether the magazines got printed that particular time, but we found this recent email exchange between Jim and the printer regarding another job:

Jim: Here 'tiz, having been proofread and tweaked. Any chance you'll still be able to run this today? Let me know when, and I'll pick up or send a minion.

Murray: That's a 2-hour window you've given us... I mean, I know we're absolutely amazing but... Yes, as it happens, I've done them. I expect at least a knighthood for this please, if you could have a word in the relevant authority's ear. See you shortly (if you're awake).

I didn't have all that many lengthy encounters with Jim: we would talk for a few minutes each time he was planning some work and a schedule would be drawn up. It was a running joke in the office that "Wednesday morning, first thing" would turn into 4:45 on Friday afternoon. Jim could always be relied upon to not only be absolutely last-minute but also totally relaxed about it, which is why none of us ever fell out with him.

My suspicion is that he would spend inordinate amounts of time picking a problem apart so that it would be solved *absolutely*, not just patched over like so many of us would do – the trade-off being that days would be lost and sleepless nights had. In spite of being at the wrong end of these delays, I understood his meticulousness and greatly admired his patience and tenacity.

It struck me early on that Jim considered everything. I remember him expressing surprise that I opened a C5 envelope on the long edge (where the flap is) because it saves effort and an infinitesimal amount of time to open a

shorter edge!

The last time I saw him was when he brought round a bottle of wine for me and my staff to say thanks for a rushed production job. He was such an incredibly good-natured gentleman, always thinking – I mean, properly thinking. I can't help imagining that if everyone on the planet was like that, maybe things would work a lot better and we'd all be less stressed. Well, there would be a few stressed printers – collateral damage.

Those are my abiding memories. I didn't know him well but he was certainly a character and made a big impression on the town.

Murray Wallace, Direct Offset

I met Jim once at a RiscOS show, and our brief chat was one of the highlights on my visit: he came across as someone very easy to talk to, cheerful and kind.

Stephen Gadd

There was something of a recurring joke at shows. I don't recall anyone literally running sweepstakes, but we'd often have a laugh about when or if Jim might appear.

If I'm honest, it masked a modicum of friendly exasperation mixed with a tinge of envy. For a show organizer (or visitor), empty exhibitor tables aren't ideal, as we all work quite hard to make the scene still feel alive, exciting, interesting. So, for example, at the SW show, we tried to plan carefully for the Archive table to make it less obvious if he wasn't there.

For me, as someone with a lot of health issues (doing shows leaves me ill for weeks afterwards) I was always rather envious, in a way, that Jim could get away with it! That people didn't seem to mind and were so accommodating is in no small part demonstrative of the affection everyone had for him. :-)

All that being said, the triumphant moments when he would enter, brandishing the freshly printed magazines, was always an almost magical moment!

Andrew Rawnsley

At the time that Jim was taking over Archive from Paul Beverley, Jim's emails showed a distinct lack of capital letters at the start of sentences and, probably the pedant's worst nightmare, lowercase for the first person personal pronoun. This led me to ask Jim at a RiscOS show whether he was a disciple of e e cummings. Paul explained to the audience the reason for this reference.

Derek Todd

rool brittania

so farewell then jim nagel

i shall miss you

and your wry canadian humour

you took the torch from paul beverley

and kept the light of risc os

flickering throughout

the lean years

of commercial decline

and the bitter os flame wars

until it shone brightly again

in a million raspberry pis

apologies e j thribb

and e e cummings

John Rickman

It was Jim who always put the finishing touches to my Archive articles making them much more readable and much more useful to the RiscOS community. I shall miss his constructive criticism of my writings.

Gerald Fitton

I well remember criticism of Jim about an article reviewing printers, which

appeared in one of the glossy magazines.
[Probably Computer Shopper. —Ed.B]

The ability to superimpose text on photo background was perhaps new. Part of the text was near impossible to read because of the background photo in various shades of grey. This generated a lot of criticism of Jim. He was near outraged and explained that he was the “editor” and the policy at the magazine was for the editor to handle the words but the magazine was then passed to the illustration team and the editor had no control over what they did to the article!

At least with Archive the editor had complete control over the production.

I greatly appreciate what your father did in maintaining the magazine.

Gerald Dodson

A Nagelism: Jim liked to have fun with the English (or Canadian) language. For instance, he would ask for “a narticle” for Archive in reverse to the way that Middle English “a napron” eventually became “an apron”.

Chris Newman

I met Jim at Wakefield in the Thornes Park days, and went on to do some small articles for Archive. I remember him best for his eccentricities:

His “archy and mehitabel” phase when he pretended he couldn’t reach the Shift key and insisted on all lowercase – a disadvantage for a magazine editor!

Then, later, he started his personal campaign against camelcase which, surprisingly considering that earlier phase, seriously offended him.

I took delight in teasing him about both on the Acorn newsgroups. I believe he took this all in good part – we seemed to remain on cordial terms privately!

Having cause to require a French translation for camelcase, I was delighted to find that they make a *jeu de mot* of it, calling it *la casse ChatMot*, which is a homophone for the French word for camel, *chameau*. I don’t recall if I ever shared this with him, but have no doubt he would have been delighted if I had! The French sometimes take their humour to a level above the rest of us!

John Williams

The company of a kind man

Mark Moxon

The early 1990s: a pretty good time to be an Acorn journalist. Between 1991 and 1994 we got our hands on plenty of new era-defining products, like the A5000, RiscOS 3, the A4 portable, Artworks, Impression Publisher, Sibelius 7 and the RiscPC, to name but a handful. We still had a few years to go before the 1998 cancellation of Phoebe, when the writing on the wall finally spilled over into the news pages; for now, the Acorn market still

had an Acorn in it.

In those days Acorn held regular press events, sometimes in Cambridge, sometimes at glossier locations, but always with the same crowd of editors and columnists. It was a very sociable period; this was a time when the Acorn User and Acorn World shows filled entire exhibition halls in Wembley and Olympia, and magazine circulations were high enough that when the press gang got together, rivalry was good-natured rather than cut-

throat. The graphs might not always have been heading in the right direction, but life was good, and *Acorn User*, where I was lucky enough to snag a job, proudly announced on the cover that it was “the best-selling Acorn magazine in the world”. In the world, no less! Confident stuff indeed.

Middle-shelf modesty

But even with a charitable twist, that strapline still read as “big fish, small pond”. The *real* players were the computer magazines that took up prime real estate in WH Smith, the ones at eye level rather than those tucked away on the top or bottom shelves where our Acorn magazines timidly peeked out from behind the ST and Amiga titles. And king of them all – the absolute best-selling computer magazine in the UK, bar none – was *Computer Shopper*.

And *Computer Shopper* had a monthly Acorn column, written by one Jim Nagel. Before I met Jim at one of the press days, I wondered what he might be like. Would he look down on the Acorn market, like the PC and Mac crowd did, as being little more than a niche, almost has-been educational market that big-name developers tended to ignore? Or would he just be a standard technology hack, pumping out words by rote, as the major newspaper technology journalists seemed to do?

Jim, of course, turned out to be none of these. Despite having the largest readership out of all the Acorn journalists – by a country mile – he was one of the nicest and most modest people in the entire press cadre. As I met him at shows, press days and junkets over the years, I’d gravitate towards him as someone almost entirely free of agenda or negativity. Having a long day at Acorn World, having

to field readers’ endless questions about typos in the magazine’s program listings? Look, there’s Jim over in the hospitality area, like an oasis of calm. Let’s go talk to him and recharge our smile batteries. Or perhaps this press day is dragging on a bit? Because let’s face it, the educational market isn’t always the thrill ride we’re hoping for. Don’t worry, Jim’s always got a positive outlook on things – let’s go have a chat with him and have a walk on the bright side. And so it went, until I moved on to other things.

My name’s on the list

The last time I saw Jim was in 2008, at the Acorn Reunited event. This celebration of 30 years since the founding of Acorn was supposed to be a gathering of ex-Acorn employees... but who was there, but Jim Nagel, newly appointed editor of Archive magazine, which he had just taken over from Paul Beverley. I’d done a stint at Acorn Australia and Acorn New Zealand, so I’d managed to wangle an invitation, and Paul had also worked for Acorn in the past, so that’s why he was there... but Jim? Apparently he’d managed to talk his way in, despite a complete lack of Acorn payslips.

But that shouldn’t have surprised me. Who wouldn’t want Jim Nagel at their anniversary bash? I’m sure he just smiled at the bouncers and they figured, just like I had 15 years before, that this was definitely one of the Good Guys.

Thank you for your company over the years, Jim. I enjoyed every minute.

Mark Moxon mark@moxon.net



Mark Moxon was editor of *Acorn User* in the early 1990s and still regularly tinkers with his beloved A410/1. Once an Acorn nut, always an Acorn nut...

That's a lot of harddrives

Bart Nagel

After my dad died, one of my tasks was to secure access to all of his files. We wanted to save anything potentially important like finance records and email; anything sentimental like photos, letters, old schoolwork; and his work, especially that which was still in progress. In all of these cases, the goal was not only to preserve the files, but also to ensure we could still read them in the event that we no longer have a working RiscOS machine.

This process turned out to be an enormous, still-ongoing task. Some backups had been set up but they didn't cover anywhere near everything, even on the main two machines. In addition, we gradually uncovered a huge stack of harddrives whose backup status was entirely unknown to us. Most had notes on them saying when they were bought, or which machine they were pulled from, but not the status of the data!

I took stock of the RiscOS machines: the working machines were an ArmX6 and an Armini. The RiscPC wouldn't boot, the Iyonix booted *sometimes* but would hang whenever it felt the slightest breeze, and the A5000 didn't have networking. The RiscPC and Iyonix both had Compactflash cards, and the ArmX6 and Armini had MicroSD cards, duly added to the list.

Acorn-format drives and Linux

I'm a Linux user, and I'd work with the data most efficiently from the commandline on my own machine. As such, my first thought was to plug the drives into my laptop somehow. I'd read that the `adfs` mount type option promised by Linux's `mount` command is in some way broken or out of date, and so I wouldn't be

able to read the drives that way. Looking back, a properly-set-up emulator might have helped.

My laptop has a MicroSD card slot, so I thought I'd try anyway in case the information was out of date. I took the card from the Armini and put it in the slot. I asked the machine which harddrives it could see, and it reported the card's presence and the sizes of the partitions it could see. Attempts to mount them (taking care to specify read-only) were unfruitful and I gave up. To my dismay, once the card was back in the Armini it would no longer boot! I looked back to the RiscOS machines to read the drives.

Back to RiscOS

The ArmX6 is where dad did most of his recent work, so that was my first choice – it'd likely have all the software needed to access the various file types, plus it had an optical drive and various card readers. Popping the lid revealed that it used Sata devices, which wouldn't help me, but one of them led to a USB device with not only a Sata connector but also regular-sized and laptop-sized IDE connectors – bingo!

I asked Andrew Rawnsley whether the IDE interface in the ArmX6 was likely to do what I wanted it to. It ought to, he said, and in fact he's used it for the same thing in the past. What's more, as a USB device it ought even to be hotpluggable.

So I set to work. I'd been told the drive would appear in the iconbar when it was plugged in. After picking a drive, nothing appeared, even after a reboot. Trying another caused the machine to freeze while the drive was plugged in.

The third drive I tried appeared immediately on the iconbar. Over the next

days and weeks I found that this isn't predictable: a known working drive will sometimes be detected right away, and other times require a reboot.

Still, I could access this particular drive, so I set about finding the best way to back it up. Dad used **Safestore** for automated backups, and exploring its options I found that it knows what a Nas is and has a concept of one-time backups. That sounded just right. I set the options as necessary, then started the backup job. It seemed to be going well but, as one of the larger drives, this was going to take some time. I monitored it for a while then left it overnight, only to find in the morning that it had stopped with an error message. It couldn't copy some file or other, and trying again gave the same result. The inability to copy one file of tens of thousands should not bring an entire backup operation to a halt, let alone with no option to continue! I abandoned Safestore.

I couldn't find any information about whether Safestore actually does anything a regular copy operation wouldn't, assuming I don't want to schedule it or skip any files. So after a few test copies to make sure timestamps were being preserved (yes for files, no for directories: acceptable), I started a full copy of the drive's contents to the Nas over the network, just by dragging and dropping. Unsurprisingly, this failed on the same unreadable file, and I noted its name to investigate later, then hit "skip". There were a few such files, and I found myself wishing there was a "skip all with this error" option, which ideally would log the failed files for later investigation. Without such an option, I either had to stay in the room or check in periodically, potentially wasting a lot of time. Regardless, that was one drive backed up, and others could be handled

the same way.

But what about those drives that weren't detected at all? Another conversation with Andrew yielded a possible cause: older drives didn't quite adhere to the eventual standards, meaning a modern USB interface doesn't know what to do with them. For those drives I'd need an older machine with a real Acorn IDE interface.

RiscPC to the rescue

I mentioned earlier that the RiscPC wouldn't boot. Well, maybe that's a software problem – a corrupted !Boot, perhaps? Will it boot from other drives? I unplugged the Compactflash-to-IDE adapter from the Unipod and attached a drive which had lived in that machine in the past. It booted to desktop. So far so good.

It looked to be an older drive – the system wasn't configured with networking. I tinkered for a while but couldn't get that set up. Perhaps I should have reached out to the community for help – the ability to copy drive contents straight to the Nas would have made things easier – but as it went, I resigned to coping without.

Without network, and given that many drives couldn't be read in the ArmX6, I'd need to find a drive that not only had plenty of free space but also that could be read by both machines. I'd use this as a shuttle, physically moving it back and forth between the two. Another requirement was to be able to connect two drives at once to the RiscPC: the shuttle drive plus the drive which needed to be backed up.

I found a suitable drive after a few tries. (In fact I found two, but the first was unbearably loud. In the end that one got some extra attention from the drill.) I was also pleased to find that the RiscPC would boot from the shuttle drive with another

drive connected at the same time, enabling me to copy from one to the other, with only some minor reconfiguration.

That was the start of the very time-consuming process of backing up this stack of drives. One by one, I would first try to connect a drive to the ArmX6 and reboot. If it showed up – great! – I'd copy its contents straight to the Nas. If it wouldn't, I'd sigh and put it instead into the RiscPC, boot that, copy its contents to the shuttle drive, shut down, hook the shuttle back up to the ArmX6, and copy the backup onward to the Nas.

Curveballs and oddballs

Remember that MicroSD card? That one was simple enough to deal with in the end. I first tried to use the ArmX6's front MicroSD card slot, but nothing seemed to happen. Instead, all it took was a USB card reader. Backing up the USB stick that was in the Armini was even simpler – all I had to do was plug it in.

For the Compactflash cards, I tried the card slot in the front of the ArmX6. It didn't feel right and, on close inspection with a torch, I saw that some pins were bent. I tried instead the Compactflash-IDE interface which had been in the RiscPC (in turn connected to the IDE-USB interface in the ArmX6) and this did the trick.

Two drives came from a Linux machine and were easily read by my laptop via the IDE-USB interface. It was strange to have the harddrive powered by the ArmX6 while its data connection went to my laptop! There were also two 2.5-inch drives from a Windows laptop. These had the smaller 44-pin IDE connector which the USB interface could also connect to.

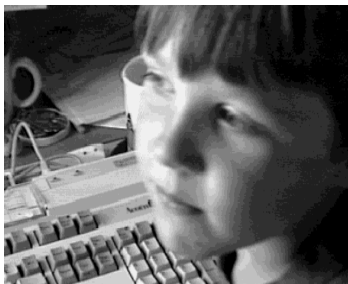
However, I realized that in this particular case I could entirely avoid copying across the network by connecting it directly to a USB port on the Nas. This worked flawlessly.

Now that the RiscPC was running, we had a working floppy drive. While clearing the house we had set aside several floppies whose labels looked interesting. To our surprise the majority of these read just fine, and happily copied to the shuttle drive.

I'd been worried about the Iomega Zip discs, but after finding its drivers on a floppy, the drive worked right away. Most of the Acorn-format discs copied fine. A few were unreadable, and I remember the discs being susceptible to corruption back when the technology was current. Without another option, these discs were destroyed.

So what's left? Two more Acorn-format harddrives I'd been putting off. One was traditional Scsi and I remember dad

getting it when I was small to replace the Archimedes A420/1's original harddrive. He'd had to buy a podule to connect it. Well, by this point my brother and I had already catalogued all the equipment found in the house and knew exactly where to find not one but two Scsi podules! I plugged the Castle one into the RiscPC, and the drive into that, and hoped for the best. No drive on the iconbar, even after fiddling around with configuration. Do I need drivers? I searched our spreadsheet and found we had catalogued a floppy disc labelled "Scsi utilities" from Castle. Before too long I



Original caption: "Bartie and Archimedes are old pals." This photo was taken with a borrowed video camera (thanks, Hugo Fienmes) connected to the Archimedes. And now copied directly from the 1993 Impression document it was found in.

had the drive detected and readable.

Last was the harddrive originally in the Archimedes – 53 whole megabytes! Its interface was unfamiliar to me. Some research revealed it to be ST-506, and that I wouldn't easily hook it up. Unless there was an Archimedes lying around, which we – ah – sold some days earlier. We hadn't been able to get it to boot anyway, so I doubt we could've done more without specialist knowledge.

What about reading the files?

With all the files on the Nas I'll still have access once I'm back in Vancouver, but this is only the start of the story since access to the files doesn't necessarily mean I can usefully read them. For files in many formats I still need a RiscOS machine or emulator as well as the right software.

Email was in **Messenger Pro**. I've already migrated the full archive to Google Mail, which was a saga all of its own, and I documented my final solution in detail at gist.github.com/tremby.

For spreadsheets, dad used **Fireworkz**. I got **RPCEmu** running on Linux, and Fireworkz runs and opens these files without a problem. I can export to other formats in future, should I need to.

Databases were in **Datapower**. This is another package I got to run on RPCEmu. I've failed to get to grips with the software itself so far but, comfortable in the knowledge that I can run it at home, I'll figure it out if and when I need to.

The last important format is **Impression**. Dad was loyal to Computer Concepts since Interword and Wordwise on the BBC and used a lot of their software. He used Impression for 30 years, and so I think it's really important that I



With data safely copied, the drives eventually met the drill bit.

can still open his Impression files. I haven't yet managed to run Impression stably on RPCEmu, but **Ovation Pro** runs and is opening most Impression files. This is better than nothing, but it's unable to handle the very old files, so I'll definitely still try to get Impression working under RPCEmu.

Compression

Speaking of Computer Concepts, many of the very old files were compressed with Computer Concepts' **Compression**, also known as CFS. I couldn't get CFS to run on either the RiscPC or the ArmX6, but searching the web gave me solutions. File Shrinker by David Pilling sounded promising, but it was an open-source decompression project called **Riscosarc** that caught my eye (github.com/mjwoodcock/riscosarc). A commandline program like this has the added bonus that I can script a batch operation directly on the Nas: run a wide search and decompress each

compressed file found, preserving the filetype and original timestamp of the contained file. Please make sure you know exactly what you're doing before trying this, but here's the command I'm running:

```
find . -name '*.d96' -print -exec \  
riscosarc -x -F -d- \  
--delete-archive {} \;
```

Looking to the future

Archiving all of my dad's digital data was a significant undertaking. I jumped through a lot of hoops and am grateful for everything I've managed to preserve. I'm now confident I can usefully read the vast majority of the data and I've also found that the community will be happy to help if I come across something I can't. There's still more to do: while I can read this data, there's much of it that the rest of my family still can't. Photos are at the top of

the list for migration to an online service we can all access from anywhere. I hope to make other forms of data accessible to my family in a similar way.

You may be in a similar position to my dad in terms of digital legacy, with data worthy of carrying forward from old systems and discs to locations and forms with better accessibility, in all senses of the word. I hope this article has given you ideas for how to get started preserving your own data, and maybe even the motivation to start now. You and your loved ones will be thankful you did.

Bart Nagel bart@tremby.net



Bart has spent enough time clicking "skip" now, but waiting for backups allowed him to rediscover some excellent old BBC and RiscOS games.

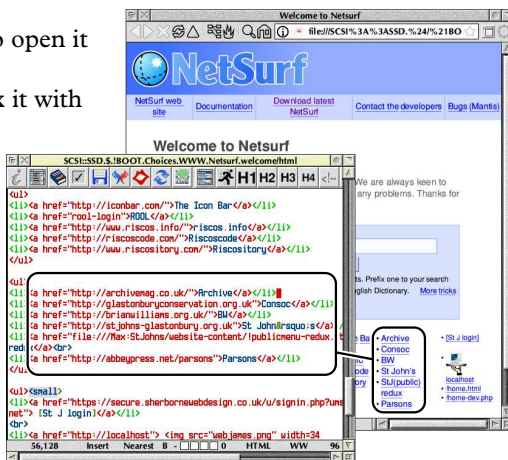
hint&tip A better welcome from Netsurf



The **Netsurf** welcome page – the one that comes up when you click Netsurf on the iconbar – is easy to adapt to your own needs. You might want to add other links of your own, for example, to the ones listed by default. Here's how:

- Open the existing welcome page.
- Shift-Adjust-click its "close" button (the X at top left). The directory where it lives will open: `$.!Boot.Choices.WWW.Netsurf`
- Shift-doubleclick the file `Welcome/html` to open it in a text editor.
- Immediately rename the old file (I suffix it with 0) to preserve it as a backup.
- Make your edits. The HTML is simple and you can copy and adapt existing lines.
- Save it (i.e., this file gets the old name).
- Refresh the Netsurf display (F5 or Ctrl-R or click the "roundabout" icon, fourth from left on buttonbar).
- Repeat the edit–save–refresh steps until you are happy with your result.

Jim Nagel



Your floppies are getting old

Chris Nagel

My understanding is that, even when they were in common manufacture some 30 years ago, the life expectancy of a floppy disc was around a decade. Owing to recent events, we found it necessary to read and copy files from around 200 such discs (a mere fraction of dad's accumulated floppy hoard) – verifying probably half of those – and not one appeared to be corrupt or would entirely fail to read. An unexpected result.

Breaking the mould

In this particular instance, it transpired that the biggest issue with old floppies wasn't the deterioration of the data but rather the deterioration of the enclosure. Even if stored in a responsible and suitable container, we noticed that the disc's plastic casing weakens over time and lends itself to easily cracking under even minor stress. Many of the floppies we dealt with were housed within big black plastic boxes (à la The Fourth Dimension) with their neat disc-sized retaining clips. Unfortunately, it took a few cracked shells to realize that there is a trick to getting an elderly disc out of, or into, such a case.

Pro tip

It seems rudimentary now, but adding a little pressure in the opposite direction of the clips will cause them to open up a fraction – just enough to easily remove or insert a disc from even the tightest restraint. Basically, bend the box back a bit and you won't break your floppy.

More than a save icon

One other issue we initially overlooked was the ability of a machine to actually

read an ADFS disc. Theoretically – using specialized drivers – it should be possible for a modern tower-style computer to do so using an internal floppy drive. However, floppy drives (and, increasingly, optical drives) have been absent from new machines for a number of years, which likely behooves the purchase of an external drive. These are most often connected via USB meaning, thanks to the lack of low-level hardware control on these devices, most modern floppy drives are unable to read Acorn floppies. Fortunately, we had a working RiscPC with which to test and backup discs, but if we had access only to dad's most recent main machine, an ArmX6, reading the old floppies would not have been possible at all.

Time to act

A floppy (like all magnetic storage media) suffers from an unquestionably finite lifespan and, taking this into consideration, we were extremely lucky not to experience any corrupted discs. Thus, it would be prudent to promptly back up any important data to more modern storage. Acting swiftly will also maximize the likelihood of securing hardware capable of reading these discs. Don't forget to ensure you *can* retrieve and read the data from the backup. It might be years before you want it, so please consider comprehensively documenting the process: a backup you can't retrieve isn't a backup at all.

Chris Nagel drums@chrisnagel.co.uk



What Chris Nagel knows about computers could be written with a very large pen on the back of a very small postage stamp. What else were you expecting from a drummer?

Mac Matters

Mark Stephens

The column that ambles around the author's crossover world between MacOS and RiscOS.

Updates since last time...

Last time I had upgraded our office setup to include two Mac machines for testing: a still-zippy 2014 Mac Pro model and a 2018 IMac Pro. These have worked very nicely for the purpose. One of the reasons we run our regression tests on real hardware is for performance. We need these tests to run as quickly as possible and there is lots of I/O. This works very well.

The recent challenge was to update them to Catalina, the new release of MacOS. This changed lots of things: no 32-bit apps, no access to the root directory (where we stored all our test data), things we relied on being in a certain location quietly moved, and so on. We found out all of these when I upgraded my old Mac laptop as a test case and found that things no longer worked.

We cautiously made the changes we needed to our tests first and then upgraded the boxes one at a time. Everything is now running smoothly.

My 2016 laptop had some problems at Christmas and needed to go away to Apple for four weeks, which gave me a good excuse to upgrade to the new 16-inch Mac Pro laptop. These have the next-generation Intel chips and you can now have eight cores on a laptop. I do not notice this machine being significantly faster (until I use the old and now repaired machine which is no slough itself), but having eight cores means things run a lot more smoothly, especially if you have Linux and MacOS running at the same time.

Welcome to Catalina

Apple updates MacOS once a year for free, which makes it much easier to encourage users to update. The 2019 release is called Catalina. The biggest change for me was the fact that it no longer runs 32-bit applications.

I used 3QD's **VirtualAcorn** to provide a virtual RiscOS laptop at work and on the road. This is a 32-bit application, so I found once I updated to Catalina that I could no longer run VirtualAcorn on my Mac.

This MacOS change has been public knowledge for several years so most other applications I use now run as 64-bit applications. I contacted 3QD and their position is explained at virtualacorn.co.uk/news.htm#181019, with some hacky workarounds given at virtualacorn.co.uk/support/articles/164m/index.htm. It seems that a 64-bit version of Virtual Acorn is unlikely to be released unless they see a lot of demand, so if you are in a similar situation you should contact them to show your interest.

Running RiscOS on a Mac

In the meantime, I also run Linux on my Mac and noticed that there was a new release of **RPCEmu** for Linux with much easier networking configuration. This has since been updated again to 0.9.3 (at the time of writing) which fixes some issues and includes some nice extra downloads to make it easy to set up with RiscOS Direct.

RPCEmu is available for Windows, Linux and Mac. The Mac release tends to lag behind other releases which is why I went with Linux.

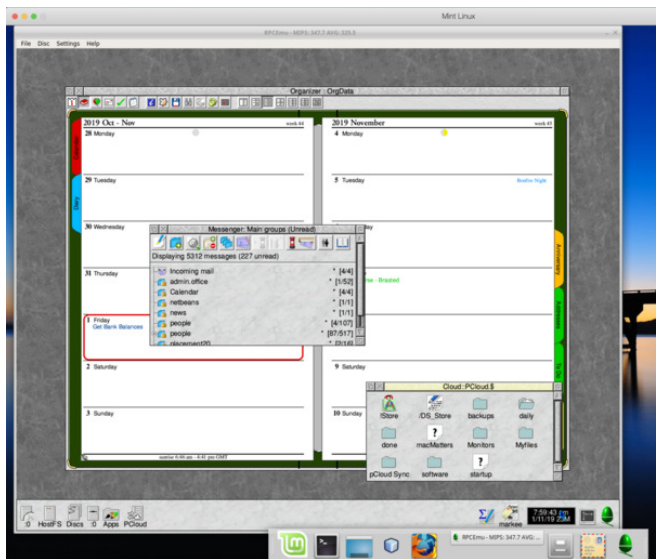
The Linux version takes a few steps to set up and install, but I followed the

instructions and have to say I am impressed! I have had some issues (the filer fails to copy content sometimes), but I now have RiscOS up and running again on my Mac. I can use **MPro** and I share files with my Titanium using Elesar's excellent **CloudFS**. It also simplifies my life in that I run RiscOS 5 on all my systems — I had previously had RiscOS 6 on VirtualAcorn. RPCemu is not a perfect solution: ShareFS does not work, for example, so it is harder to copy files between RiscOS machines.

This setup means I now use my Titanium at home. When at work or on the road, I have RiscOS on my laptop. My email uses Imap, so it is stored in the cloud and accessed from whatever machine I use. CloudFS gives me shared access to my documents and **Organizer** data. Be careful because Organizer does not guard against being accessed from multiple machines simultaneously, but in practice I will only have it open on one at a time.

Another way to use RiscOS on a Mac is with **VNC**. You can run a VNC server on a machine whose desktop you want to share, and then a VNC client on another machine. Connecting that to the server then lets you view and control the server's desktop. Jeffrey Lee has been tweaking a RiscOS version of the server software, which can be found at phlamethrower.co.uk/riscos/vnc_serv.php. Richard Brown of Orpheus kindly configured my router so I could access my Pi remotely, but I find it a little unresponsive for serious use.

At the London and Southwest shows



RiscOS 5 running under MacOS via RPCemu.

[see page 8], Adrian Lees was demonstrating much faster VNC access so I have been pestering him to let me know when it is available.

Signing off

This is my last column under “the current management”. I was very sad to learn of the death of our esteemed editor and we will all miss him.

If Archive continues (and I hope it does), I would be delighted to continue with Mac Matters if asked. Otherwise, I am a regular contributor on www.iconbar.com and I am always happy to chat or advise via email. Happy computing on all your boxes, Mac and RiscOS alike.

Mark Stephens

markstephens@idrsolutions.com



Mark Stephens celebrated 20 years of working for his own company at Christmas. He now claims that he has worked for nobody for most of his career!

Where Acorn leads...

Paul Beverley

Where Acorn leads, the rest of the world will follow – 32 years later.

The apple and the arc

Let me take you back to the birth of Archive in 1987 – well, to its pregnancy, anyway. Acorn invited a number of journalists – including, of course, our beloved Jim – to a confidential meeting, where they showed us the revolutionary Archimedes. Clearly, I had to get on board in some way.

The previous year, I had been to Computer Concepts' Gaddesden Place HQ, where Rob Pickering showed me another revolutionary computer – the mouse-and-pointer Mac Plus and its 300dpi Laserwriter, which could print using en dashes, bullet characters, dingbats and even proper ligatures (single characters like “fi” for “fi” and “fl” for “fl” *[can you tell the difference? —Ed.B]*). The list price of the pair (in 1986) was £8,000, but Apple had a writers' discount scheme, so I bought them for “only” £5,000.

The Mac Plus and the Archimedes came together (so to speak) and Archive was born.

Frustrations & resolutions

However, for each issue, I wanted to take the text and transfer it onto an Acorn floppy disc, which subscribers could search electronically. The ligatures, bullets and en dashes were fine, but I had to use a series of global find-and-replaces to change them

back to “ordinary” text. One day, I looked at my paper-based list of F&Rs and thought, “Surely this could be done by a computer program” and in 1988 or so, thanks to Paul Sprangers, **Convtext** was born.

Fast forward to 2005 and the non-birth of Phoebe, the yellow computer, meant that my computer-sales business was dead in the water. What to do to save the heavily mortgaged house? After editing and proofreading Archive, surely I could sell those services to others, and I went freelance – but that meant using the dreaded Microsoft Word!



Apple Mac Plus (list price in 1986, including Laserwriter, £8,000).

I coped, but after using Convtext for over 16 years, I felt bereft. I turned to my faithful Acorn friends, “Could anyone create a version of Convtext as a VBA macro?” and Philip Marsden very kindly wrote one for me, which I called **PreEdit**. It greatly streamlined my preparations for editing.

I tried to tell the rest of the editing community but they were resistant to this

“new idea” of scripted global F&R, which you and I had been using for 16 years!

PreEdit was a text-only system and, after a couple of years of learning VBA, I rewrote it, calling it **FREdit** (find-and-replace edit) and added, stage by stage, bold, italic, subscript, superscript, fonts, styles, wildcards, case sensitivity; then highlighting, font colouring and switching track changes on and off. Today, FREdit is the single most powerful editing tool available for MS Word, if I say so myself!

...the world will follow

Thankfully, after 13 years, FREdit is being widely accepted. I now have dozens of training videos on my Youtube channel

(with over 750 subscribers) and it is being used in over 50 countries. This week I got myself a paid Zoom account so as to provide a FREdit surgery at 8am for Antipodean editors and 4pm for the North Americans.

So, where Acorn led 32 years ago with Convtext, the world is now following.

Paul Beverley paul@archivepub.co.uk



Sadly, I now inhabit a very different world from that of the Raspberry Pi, so our paths have diverged, but I remember you, my Acorn friends, very fondly, and especially our dear Jim – such a tragic loss! But I will see many of you, I know, when we are allowed to gather again, to celebrate Jim’s life. Bless you all!

Thanks

Thanks from Viola

It is a tribute to Jim’s style that even I, the ultimate technophobe, found pleasure in reading Archive. Jim often asked me to proofread articles as he was putting the magazine together. It was my privilege. I won’t miss him staying up all night to finish, but I will remember his commitment to the Archive community and, despite his tardiness, the support and kindness he experienced there. Thank you.

And thanks from the three of us

We are grateful to everyone who made it possible to produce this issue of Archive:

- **IDRsolutions** for sponsorship – we could not otherwise have printed this magazine;
- **Gavin Smith** for writing up the news;
- all of the other **authors** featured in this issue;
- **Andrew Rawnsley** for many rounds of help with hardware and software;
- **Chris Hall, Richard Keefe and Adrian Lees** for help with Impression;
- **Martin Würthner** for help with Artworks;
- **Murray Wallace** at Direct Offset for being absolutely amazing (expect a knighthood);
- **Kris Adcock, Paul Beverley, Bernard Boase, Richard Brown, Hutch Curry, Gerald Fitton, Julian Fry, Steve Fryatt, Chris Hall, Bryan Hogan, Bimal Jangra, Jim Lesurf, Paul Middleton, Andrew Rawnsley, Mark Stephens,**

- **Chris Williams and Gavin Wraith** for thoughts, advice and guidance;
- **all** those who sent in words, thoughts and anecdotes about Jim, whether or not they were printed;
- members of the **c.s.a** newsgroups, the **Archive-online** mailing list and the **Stardot** forums for help on various topics;
- **you, the readership**, for your continued support of Archive;
- **everyone** who kindly offered to donate – we appreciate your generosity but would rather you continue to support the magazine via its new editor; and, from our hearts,
- **Jim Nagel**

Bart, Chris and Viola Nagel

