# VirtualRPC In Use

*Third Edition* *September 2018*

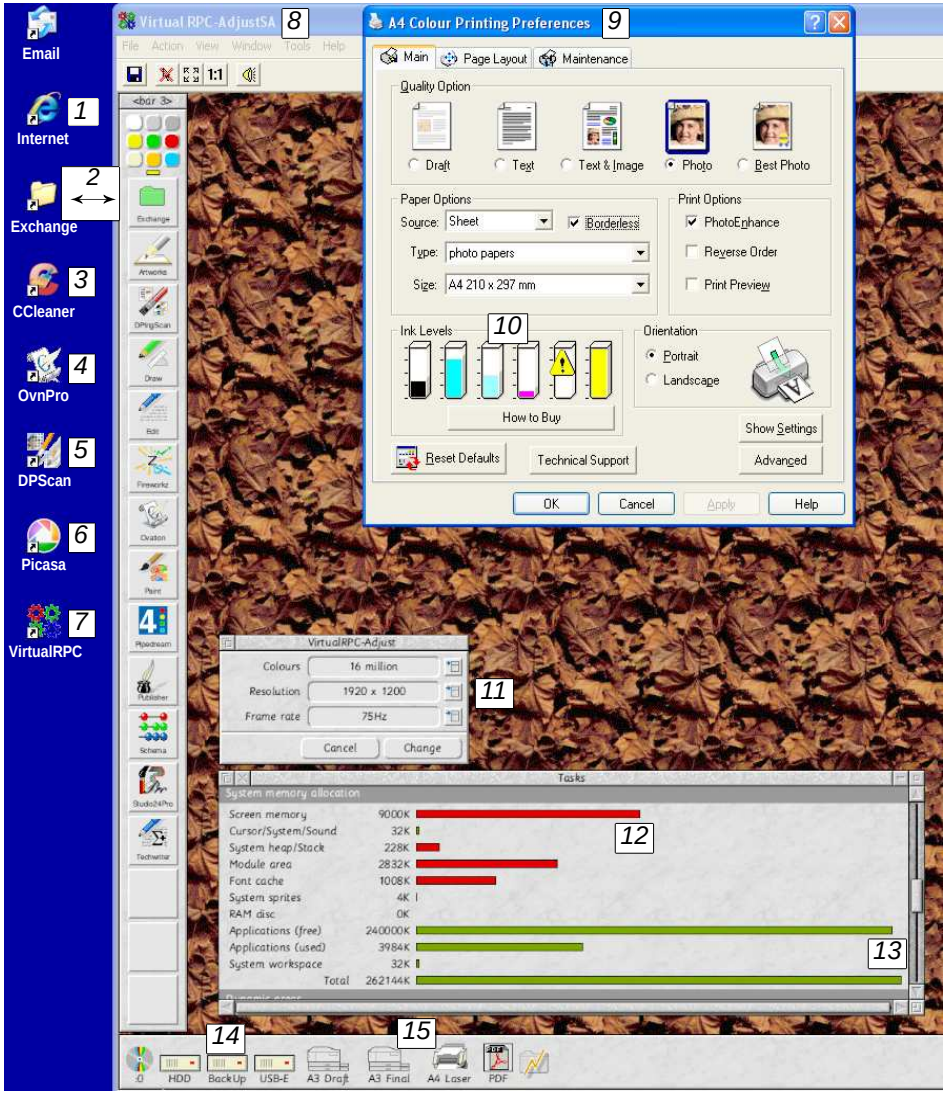
Steve Johnson & Jim Nottingham
*admin@toms12.plus.com*

***Key:***

1 Trouble-free surfing (choice of Microsoft and open-source browsers) – *p73*.
2 Direct Windows ⟷ RISC OS exchange of files, documents, etc. – *pp31 & 38*.
3 Keep your footprints clean (*CCleaner*) – *pp17-18*.
4 Ovation Pro for Windows (*OPW*) – *pp65-66*.
5 David Pilling's Scanning Software for Windows (*DPScan*) – *pp66-69*.
6 High-performance image thumbnailer/viewer (*Picasa*) – *pp74-75*.
7 VirtualRPC (double-click to start up).
8 Choice of RISC OS and ARM core emulations – *p12*.
9 Ultra-high-speed direct printing under Windows – *p55*.
10 'Live' ink cartridge contents display – p50.
11 RISC OS display up to 2560×1600 pixels in 16M colours – *p30*.
12 Available screen memory 8000K (16000K under ARM7500 emulation) – *p13*.
13 Up to 256MB RAM (all variants) – *pp25-27*.
14 High-speed access to all PC storage devices incl. USB/SD, etc. – *pp34-37*.
15 High-speed printing using RISC OS PDF and/or UniPrint, or direct to Windows – *pp51-56*.

# VirtualRPC In Use

*Third Edition* *September 2018*




*Go to Contents*

# VirtualRPC In Use – Edition 3 (Sep 2018)

## Introduction

Shortly after **VirtualRPC** was released, back in September 2003, a group of time-expired military aircraft and systems testers going under the collective name of T.O.M.S. started on a lengthy series of articles in Archive mag called **VirtualRPC In Use**.

The articles were later collated into printed booklets and published by Archive; the inevitable problem being that, over time, their content became partially outdated. So T.O.M.S. completed a general update in 2010 and Edition 2 was published, this time on the VirtualAcorn website.

That was followed by a series of 12 supplementary articles throughout 2011 and, to celebrate VirtualRPC's 10th birthday, a longer article was added in 2013. For convenience, all 13 supplementary articles were collated into a second booklet, and both offerings have since been available for download on VirtualAcorn's *Support page*.

There has been much water under the VirtualRPC bridge since then, affecting what is contained in the booklets. So although there are no serious factual inaccuracies amongst the blether, much of what is included has most certainly become somewhat outdated. And what better time to release this revised Edition 3 of the main booklet than on VirtualRPC's *15th* birthday?

This is also a convenient point to explain that, through a combination of advancing age and general decrepitude, it became high time that T.O.M.S. handed in their testing hats and went their separate ways.

However, 2 of the original team – Steve Johnson and Jim Nottingham – are your scribes for this 3rd edition, so please bear with us if we continue to use "we" throughout (cunningly saving a lot of editing in those areas which don't require much updating...).

To give a flavour of what we've aimed to achieve in this general update, the following topics will be included:

• A brief review of the splendid advances in the range of hardware now available to the many Fans of ARM-Power, along with the equally splendid work of RISC Open in bringing RISC OS 5 to the fore.

• A discussion of what hardware advances have also been introduced into host Macs and PCs to support running VirtualRPC. In particular, many of you have been moving from using Windows XP to Windows 7/8 or, perhaps more commonly, straight to Windows 10, with several knock-on effects which need to be taken into account, both before and when installing, setting up and using VirtualRPC, RISC OS and its application software.

• A look at the ever-increasing range of VirtualAcorn's invaluable *Support articles* which have been introduced in recent years.

• As always, an updated discussion of the ongoing performance comparisons between current ARM-powered hardware systems and VirtualRPC running on contemporary host Macs and PCs.

• A general discussion and update on the very many options for using Windows and/or RISC OS application software, either separately or together, to optimise productivity – 'the best of both worlds'.

As previously, we still have very limited experience of Mac computers and application software, although it must again be stressed that much of the content of this main booklet applies equally to using VirtualRPC with both host Macs and PCs.

Furthermore, much of both booklets is also based on user-feedback, so do please keep that coming in.

© *Steve Johnson & Jim Nottingham*
*admin@toms12.plus.com*

# Contents


| Sect. | Title | Page |
|---|---|---|
| | **Introduction** | 1 |
| **1** | **Performance tests – 2003-2018** | 4 |
| | Testing, testing... | 4 |
| | Range of RISC OS test systems | 5 |
| | Choice of benchmark programme | 5 |
| | The 'test schedule' | 5 |
| | Comments on the tests results | 8 |
| | Conclusions | 8 |
| **2** | **Choosing a Host Mac or PC** | 9 |
| | Intel *Vs* AMD Processors | 9 |
| | Turn-key solutions | 9 |
| | Off-the-shelf solutions | 9 |
| | • Processor speeds (inc. turbo) | 10 |
| | • Number and range of ports | 10 |
| | • Cables, connectors & adaptors | 10 |
| | Manufacturers/reliability/warranties | 11 |
| | Windows XP *Vs* Windows 7/8/10 | 11 |
| **3** | **Choosing the VirtualRPC-*xxx*** | 12 |
| | Adjust or not? | 12 |
| | • Maximum RAM | 12 |
| | • Data DVDs | 12 |
| | StrongARM variants | 12 |
| | Bigger and better displays | 12 |
| **4** | **PC malware issues** | 14 |
| | A typical anti-malware suite | 15 |
| | Firewall | 15 |
| | Anti-virus protection | 15 |
| | Anti-malware scans | 16 |
| | Anti-spyware protection | 16 |
| | 'Adware' killer | 17 |
| | Further privacy measures | 17 |
| | Cookies | 18 |
| | Registry issues | 18 |
| **5** | **Setting up a PC and VirtualRPC** | 20 |
| | VirtualRPC and Win Vista or later | 20 |
| | • User Account Control setting | 20 |
| | • VirtualRPC installation path(s) | 20 |
| | • Windows 'Event Viewer' modules | 21 |
| | **Setting up VirtualRPC** | 22 |
| | The Windows filer | 22 |
| | • Navigating the directory tree | 23 |
| | • Application directories | 23 |
| (5 cont.) | • Drag-and-drop | 23 |
| | • Selecting multiple objects | 23 |
| | • Copying/moving objects | 23 |
| | • Copy to folder | 24 |
| | • Saving files | 24 |
| | • Recovery from finger trouble | 24 |
| | **Customising VirtualRPC** | 24 |
| | Making a selective back-up | 25 |
| | Configuring RAM for RISC OS | 25 |
| | Changing RAM (RO 4.39) | 25 |
| | Setting 128MB RAM (RO 4.02) | 26 |
| | *Setting 256MB RAM (RO 4.02)* | 26 |
| | Sundry RAM considerations | 27 |
| | Changing the hard disc name | 27 |
| | • Renaming under VirtualRPC | 27 |
| | • Renaming under Windows | 28 |
| | Renaming – troubleshooting | 28 |
| | Clearing the errors | 29 |
| | Selecting the startup emulation | 29 |
| | Extra HardDisc drives/folders | 30 |
| | Windows↔RISC OS exchange | 31 |
| **6** | **Setting up RISC OS** | 33 |
| | The default RISC OS display | 33 |
| | Drives, devices and all that | 33 |
| | Default CD drive (CDFS) | 34 |
| | Default 'HardDisc4' | 34 |
| | Default floppy disc drive | 34 |
| | Adding further HostFS 'devices' | 35 |
| | Hard disc/folder mounts | 35 |
| | Optical drive mount(s) | 36 |
| | Plug-in storage media | 36 |
| | Plug-in floppy disc drive | 37 |
| | 'IDE' hard disc | 37 |
| | Drag-and-drop saving | 38 |
| | Share filing system | 38 |
| | RISC OS↔Windows Exchange | 38 |
| **7** | **Transferring files to VRPC** | 41 |
| | • Software on master CD | 41 |
| | • Software for download | 41 |
| | • Software on floppy discs | 42 |
| | • Machine-to-machine transfer | 42 |
| | Compressed folder(s) | 42 |
| | Retaining RISC OS filetypes | 42 |
| (7 cont.) | The HostFS Extensions file | 43 |
| | Using Typefind | 43 |
| | RISC OS filetypes listed | 44 |
| | Generating compressed folders | 44 |
| **8** | **Reliability & compatibility** | 45 |
| | ARM-powered hardware | 45 |
| | PC hardware reliability | 45 |
| | PC hardware compatibility | 45 |
| | Peripheral equipment | 45 |
| | RISC OS software compatibility | 46 |
| | • Getting software to work | 47 |
| | • Reducing RAM | 47 |
| | • Reducing colour depth | 48 |
| **9** | **Printing with VirtualRPC** | 49 |
| | Setting up a Windows printer | 50 |
| | Setting up a RISC OS PDF | 51 |
| | Using UniPrint | 52 |
| | Desktop control of printers | 53 |
| | An elegant icon bar | 54 |
| | Printing from Basic | 54 |
| | Monochrome printing (UniPrint) | 54 |
| | Edit paper sizes | 54 |
| | Ultra-high-speed printing | 55 |
| | 'Printing' to PDF | 55 |
| | Generating landscape PDFs | 57 |
| **10** | **Multiple monitors/displays** | 58 |
| | Replicating a display | 58 |
| | Expanding the desktop | 58 |
| | Multiple monitors (VirtualRPC) | 59 |
| | Triple monitors with VirtualRPC? | 59 |
| **11** | **Sundry VirtualRPC issues** | 60 |
| | Unlock key failure | 60 |
| | 'Broken' installation | 60 |
| | Start-up to a blank screen | 61 |
| | Running VirtualRPC – twice | 61 |
| | Minimising VirtualRPC | 61 |
| | Filer hiccups | 62 |
| | Deleting RISC OS files/folders | 62 |
| | RISC OS→Windows extensions | 63 |
| | Multiple Windows extensions | 63 |
| | Windows keyboard issues | 63 |
| | Three-button mice | 64 |
| **12** | **RISC OS→Windows ports** | 65 |
| | Ovation Pro for Windows | 65 |
| | DPScan for Windows | 66 |
| | • Scanning | 66 |
| | • Image processing | 68 |
| | • Saving images | 69 |
| | Trace for Windows | 69 |
| | Fireworkz for Windows | 69 |
| **13** | **Using Windows software** | 70 |
| | 'Unknown' Windows filetypes | 70 |
| | Printing under Windows | 71 |
| | RISC OS PDF writers | 71 |
| | Windows PDF readers | 71 |
| | Windows PDF writers | 71 |
| | PDF 'Editors' | 72 |
| | Splitting & merging PDF files | 73 |
| | Browsers | 73 |
| | Word alternative | 73 |
| | Image processing/cataloguing | 74 |
| | Bigger and better thumbnails | 74 |
| | Video editing | 75 |
| | Burning CDs/DVDs | 76 |
| | Backing up RISC OS & VRPC | 76 |
| | Selective back-up | 77 |
| | Website authoring | 78 |
| | Applications in the Supplements | 78 |
| | Windows freeware | 79 |
| | 'Bloatware' | 79 |
| | Beware 'Free download!" | 79 |
| | Beware 'scareware' | 79 |
| **14** | **Sundry Windows 'issues'** | 80 |
| | System restore | 80 |
| | Start-up time | 81 |
| | Hard disc defragmentation | 82 |
| | Memory requirements | 83 |
| | Anti-aliasing (not...) | 84 |
| | • (Un)locking the taskbar | 85 |
| | • Tidying the taskbar | 85 |
| **15** | **And in conclusion...** | 86 |

# 1. Performance tests – 2003-2018

*This is a general update of the section in earlier editions, taking into account the very welcome number of post-Iyonix, ARM-powered solutions, plus the parallel updating of Mac and PC host computers. We'll also take the opportunity to expand on the range of comparative performance tests and thin out those which (for reasons to be discussed) previously gave anomalous results.*

Immediately on the initial release of VirtualRPC in 2003 – and very regularly since then – the question was and still is asked: "*How fast does she go, mister?*" The first subjective reports suggested that its performance was roughly on a par with a RiscPC running a 233MHz StrongARM or even Kinetic processor.

### Testing, testing...

But just how fast is it? Before VirtualRPC was released, Paul Beverley (the then long-standing Editor of Archive mag) had published some objective performance test results in Archive (16:5 p15), comparing his Iyonix with a RiscPC StrongARM (202MHz) and a Kinetic 233.

Although Paul's tests were invaluable for potential Iyonix buyers, nevertheless they revealed some unexpected results, at the time suggesting that Iyonix was not performing as well as anticipated, taking into account the more powerful processor and higher clock speed.

We mention this as, in the very extensive performance testing over the ensuing years, involving far, far more powerful solutions both on ARM-powered hardware and VirtualRPC on host Macs and PCs, very similar anomalies arose, to the point that they became almost repeatable.

Fortunately, the reasons for these anomalies were eventually determined and, in essence, boiled down not to the tests themselves, but the test *method* used to run the tests.

Specifically, this involved those test results where a programme written in *Basic* had been used to perform the tests. Indeed, in Paul B's original RiscPC *Vs* Iyonix comparison, he had performed a general mix of tests, some of which used small *Basic* programmes to run each test (e.g. a *Basic* FOR NEXT loop with 10,000,000 iterations).

But for the remainder, he had simply pitted the 2 systems against each other in more practical 'like-with-like' activities of the type which we all undertake on a regular basis, e.g. loading a large textfile into *Edit*, or displaying a complex *Artworks* file.

With hindsight, we can see from Paul's test results that it was invariably those running *Basic* programmes to conduct the tests which were suggesting that Iyonix might be underperforming; the corollary being that Iyonix performed very well on all the more practical 'like-with-like' comparisons. (It was that nice David Holden who eventually explained the phenomenon: in the RiscPC, the *BBC Basic* interpreter sits in one of the StrongARM caches, making it very favourable for *Basic* operations. Lesson No. 1.)

Shortly after VirtualRPC-SE was released, in 2003, a number of early users were reporting that, on the then range of host PCs (typically running Intel Pentium P4 processors), the emulator performance was very respectable, subjectively at least on a par with 233MHz-processor and even Kinetic RiscPCs; *very* pleasing.

Naturally we (with our then chums at T.O.M.S.) soon donned our testers' hats and dived in with some rather more formal testing, the 'test schedule' being based on Paul B's original tests but with a few more day-to-day tests added. The results were published by Archive in the early VirtualRPC In Use articles.



Our own test results almost exactly replicated Paul's findings against his RiscPC and Iyonix, including the performance anomalies he had turned up, so we believe this confirms the underlying problem where particular tests used a *Basic* programme to run them.

But although David Holden had explained the reason for the RiscPC *Vs* Iyonix anomalies, why did they seemingly also apply to some of the Iyonix *Vs* VirtualRPC tests?

This has never been satisfactorily explained but, without exception (and based on oodles of testing over the years), arguably we can reasonably surmise that it seems that VirtualRPC does not work at all well when running *Basic* routines, whether or not as the benchmark programme in use, or if the application is itself written in *Basic*, such as *Shares*. (Lesson No. 2.)

To put this particular sub-topic finally to bed, in late-2016 The Icon Bar published *a wide range of performance data* provided by Chris Hall, comparing early ARM-powered hardware with the newer range including ARMX6 and Titanium.

While this neatly put the ARM-powered solutions into a logical order of comparative performance, unfortunately Chris was able to include only one VirtualRPC dataset – an Intel i5 processor clocked at 3.2GHz in a host Mac computer. Equally unfortunately, the Mac data in Chris's table suggested that VirtualRPC was performing only slightly ahead of the ARMX6, and furthermore was significantly under-performing against the Titanium.

This was worrying as the few datasets we already had by then for VirtualRPC running on contemporary PC hosts (in respect of their processors and clock speeds) were all suggesting better results than the Titanium, so we looked at possible reasons for this significant anomaly. The answer was readily spotted: Chris had used the *RoMark* benchmark programme to control all his chosen test points – and *RoMark* is itself written in *Basic...* (NB Lessons 1 & 2).

In other words, it would appear that the use of the *RoMark* benchmark programme to control the tests was – inadvertently – causing the single VirtualRPC system to seemingly underperform. (The choice of a Mac instead of a host PC is not thought relevant.)

### Range of RISC OS test systems

In completing this updated set of comparative performance tests, we have been very fortunate in talking several punters into undertaking the full range of tests listed below on their own, recently-purchased systems, including 3 ARMinis, 3 ARMX6, 2 Titaniums, another Mac with an i5 processor, and two high-spec i7 processor PCs. The various owners' full cooperation is *very* much appreciated.

### Choice of benchmark programme

For consistency with all the previous testing going back to 2003, initially we decided to repeat the tests on all the newer systems, accepting that in all probability the use of *Basic* programmes to control some of the tests would most likely be subject to the same anomalies. (This replicated the earlier test data.)

However, to balance that, we tried to extend the test schedule with a number of additional 'real life' tests. Unfortunately this was partly frustrated due to incompatibilities with some 26-bit-only software applications not being (natively) compatible with RISC OS 5. The use of *Aemulor* rendered most of the programmes usable but, unavoidably, there was a significant, knock-on performance hit, so the list of extra tests was unfortunately limited.

### The 'test schedule'

A total of 27 tests were conducted on all 18 machines with, for comparison, a further test added to the RiscPCs only. Having been expanded in piecemeal fashion over the years, the test list has unavoidably become rather higgledy-piggledy but was as follows, with the results tabulated on pages 6 & 7:



Return to Contents

1) *Basic* FOR NEXT loop with 10,000,000 steps, using integer counter.

2) 'Lorenz attractor' – *Basic* program plotting some lines on the display using FOR NEXT loops.

3) Standard *Dhrystones* test.

4) In a large *TechWriter* file, changing the font size to force a complete document reformat.

5) Using a macro within a *Fireworkz* file to complete a prime number test.

6) For a hard drive speed test, a multiple file copying exercise from one directory to another; 360 small files in total.

7) As Test 6), repeatedly copying a 30MB sprite.

8) Loading a very large textfile into *Edit*.

9) Hard disc drive performance using the APDL random access file test.

10/10a) As Test 9), using the PowerTec block transfer file test (read and write speeds).

11) & 12) Displaying complex drawfiles made up by graduating between two objects 99,999 times and then copying all the objects and grouping them several times.

13) Compressing and decompressing a number of files using *SparkFS*.

14) Rendering a large *Artworks* file with complex blending.

15) Rotating a 10MB sprite.

16) A *Desk_Bogo* MIPS test; "*a good measure of the raw processing power of the processor*".

17) OCR-ing an Archive double-page spread, 10pt text, four columns/zones, originally using 26-bit *Sleuth 3* (later 32-bit).

18) Converting a 1.1MB mono sprite to a drawfile, using *Trace* v2.13.

19) Rendering (i.e. 'Print to file') Archive 16:12 using the RISC OS *PoScript2* printer

| A | B | C | D | E | F | G | H | I | J | K | L |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Test** | ***Test details (see text)*** | **2× SA202** | **3× Iyonix** | **3× ARMini** | **2× VirtualRPC (PC)** | **3×ARMX6** | ***VirtualRPC (Mac)*** | **2× Titanium** | **VirtualRPC (PC)** | **VirtualRPC (PC)** | **2× VirtualRPC (PC) Vs 2 × Titanium** |
| | | *(Multiple datasets averaged for each class)* | | | Core 2/4 2.4GHz | | (i5 3.2GHz) | | (i7 E7600 3.0GHz) | (i7 E7700 3.4GHz) | |
| 1 | For/Next loop | 5.77s | 2.32s | 2.28s | 3.71s | 1.15s | 1.93s | 0.61s | 1.47s | 1.13s | (1.5) |
| 2 | Lorenz Basic prog | 3.01s | 2.03s | 1.73s | 2.25s | 0.81s | 1.37s | 0.68s | 1.13s | 0.96s | (1.2) |
| 3 | Dhrystones | 395k/sec | 815k/sec | 1072k/sec | 636k/sec | 6737 k/sec | 9251k/sec | 13475k/sec | 16202k/sec | 20255k/sec | 1.3 |
| 4 | 25MB Techwriter file | 38s | 19s | 4s | 4s | 2s | <2s | <1s | – | – | – |
| 5 | FireWorkz | 169s | 35s | 17s | 33s | 9s | 8s | <5s | <5s | <4s | c. 1.1 |
| 6 | Copy 10 directories | 32s | 15s | 3s | <2s | <2s | <1s | 1s | <1s | <1s | c. 0 |
| 7 | Copy 30MB sprite | 41s | 4.5s | 4s | <2s | 2s | 1s | <1s | 0.5s | 0.5s | c 2.0 |
| 8 | Load 25MB text file | 23s | 4.5s | 2s | <2s | 1s | <0.5s | <0.5s | <0.5s | <0.5s | c. 0 |
| 9 | APDL Random access | 2170% | 4295% | 23471% | 311600% | 4371% | – | 86244% | – | – | – |
| 10 | Block transfer | 137% | 1185% | 1431% | 24294% | 2563% | – | 5268% | – | – | – |
| 10a | Power-Tec HDD read/write k/sec | 6083/4774 | 23252/12517 | 62051/48132 | 98946/71385 | 82147/75386 | 117257/35975 | 107247/327812 | 92267/412600 | 112947/399362 | 1.0/1.8 |
| 11 | Screen Draw 1 | 46s | 24s | 8s | 21s | 5s | 18s | <2s | 14s | 11s | (6.22) |
| 12 | Screen Draw 2 | 34s | 18s | 6s | 7.5s | 4s | 7.5s | <2s | 2.5s | 2s | (1.12) |
| 13 | Zip test (JPEGs) | 32s | 13s | 9.5s | <4.5s | 5.5s | <3s | 3.5s | 2.71s | 1.53s | 2.6 |
| 14 | Load Arworks file | 29s | 11.5s | 11s | 3s | 6s | <2s | <2s | 0.5s | <0.5s | c. 3-4 |
| 15 | Rotate 10MB sprite | 8.5s | 3.5s | <2s | <0.5s | <1s | <0.5s | <0.5s | <0.5s | <0.5s | c. 0 |
| 16 | Desk_Bogo mips | 99.1 mips | 367.4 mips | 427.7 mips | 440.5 mips | 592.3 mips | 732.5 mips | 785.2 mips | 832.25 mips | 879.5 mips | 1.1 |
| 17 | OCR A4 page | 101s | 31s | 18.5s | 16.5s | 11s | – | 8.5s | 7.5s | 5s | 1.4 |
| 18 | Trace 1.1MB sprite | 79s | 32s | 21s | 18.5s | 15.5s | 16.5s | 10s | 13s | 11.5s | (1.4) |
| 19 | Render Archive 16.12 (PS) | 223s | 93s | 53s | 21s | 29s | 9.5s | 11.5s | 7.5s | 5s | 1.8 |
| 20 | Render laser (TurboDrivers) | 2m 48s | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A |
| 21 | Ditto (RISC OS PDF) | 15m 15s | 6m 11s (Ae) | 3m 07s | 1m 29s | 1m 51s | 49.5s | 58s | 37s | 29s | 1.8 |
| 22 | Render A4 photo | 8m 15s | 6m 5s | 1m 53s | 1m 15s | 1m 25s | 43s | 48.5s | 36.5s | 31s | 1.4 |
| 23 | USB 1 Vs 2 Vs 3 | 94s (USB1) | 26s (USB2) | 18.5s | 9.5s | 8s | 6s (USB3) | 6.5s | 4.5s (USB3) | 4.5s (USB3) | 1.4 |
| 24 | Load 'real' Artworks file | – | 17s | 8.5s | 4s | 6.5s | 4.5s | 4.5s | <2.5s | 2s | c 2.0 |
| 25 | Convert Artworks to PDF | – | 6m 33s | 1m 53s | 31s | 49s | 15.5s | 18.5s | 8.5s | 6.5s | 2.5 |
| 26 | Load PDF into Artworks | – | – | 31s | 27.5s | 22s | 14s | 11s | 7s | 5.5s | 1.8 |
| 27 | Convert OPro to PDF | – | 8m 30s | 3m 57s | 2m 35s | 2m 08s | 49.5s | 1m 17s | 31s | 25s | 2.7 |
| | *Screen resolution (pixels)* | 800 × 600 | 1280 × 1024 | 1920 × 1080 | 1920 × 1080 | *2560 × 1440* | *2560 × 1440* | 1920 × 1080 | 2560 × 1440 | 2560 × 1440 | |





*Return to Contents*

definition file (PDF), with Colour set.

20) Rendering Archive 16:12 using the RISC OS *HP LaserJet 4* PDF.

21) For comparison (in the RiscPCs only), rendering Archive 16:12 using Computer Concepts' HP *LaserJet 4 TurboDriver* PDF.

22) Rendering a 12.5MB, 24-bit sprite in an A4 *Impression* frame using either a six-colour *PhotoReal* 26-bit driver or *PS3* and *Gutenprint*.

23) Downloading 29 mixed bitmap graphics (total 46MB) from a Smart Media card using a USB card reader (a USB1 *Vs* USB2 *Vs* USB3 test).

24) Loading a complex 25MB vector graphic with transparencies into *Artworks*.

25) Converting the file resulting from test 24 into portable document format (PDF).

26) Re-loading the 25MB PDF file from Test 25 back into *Artworks.*

27) Converting a complex, 175MB *Ovation Pro* document into a PDF file.

### Comments on the test results

Hopefully, the tabulated test results on pages 6 & 7, taken in conjunction with the preceding comments, will be largely self-explanatory.

An encouraging factor is that, where 2 or more computer solutions were assessed in each class, generally the datasets quite closely matched each other, so the table has been simplified by averaging the data within each class. Thus it is possible to pit different solutions against each other, simply by comparing the data in their respective cells, in seconds, k/sec, percentages or mips.

However, at the 'top end', Col L compares the 2 Titanium results (Col I) against a simple average of the 2 'power PCs' (Cols J and K).

The greyed-out cells denote where, on the face of it, the RISC OS performance of the particular Mac/PC running VirtualRPC under-performed compared with the ARM-powered solutions.

Nevertheless, for the reasons previously discussed (Lesson 2), all the indications are that the cause of the evident under-performance is the use of a *BASIC* programme to control the relevant test(s).

Thus we can arguably deduce that, for example, the greyed-out cells in tests 1 and 2 result from the indirect use of the *BASIC* routines to run the tests and, taking previous experience into account, this is most likely the reason why the 2 PCs are apparently under-performing. By the same argument, the cells for tests 11 and 12 (screen draw performance) are also greyed out. But if we compare those with the 'real life', direct load-and-display tests (14 and especially 24), then clearly the anomaly appears to have gone away.

The only other anomaly is the seeming underperformance of the *Trace* test (18) under VirtualRPC but this is entirely compatible with all other *Trace* tests going back to 2003, although the reason has never been identified.

### Conclusions

If we set aside all these avoidable anomalies, then it would appear from this very extensive series of tests that the more recent ARM-powered solutions provide Fans of ARM-power with a very considerable choice, which is splendid. Furthermore, as from the earliest days of VirtualRPC, high-end host PCs continue to provide a significant performance increment. (More on that in the 'Choosing a host' section.)

It also needs to be noted that, although by design VirtualRPC uses only one core in a multi-core processor, sometimes *very* considerable time can be saved by VirtualRPC 'multitasking' with Windows in any or all other available cores.

For instance, once RISC OS has rendered a document, ready for printing (e.g. test 22), the resulting file can be passed to another core for the physical printing by Windows – while RISC OS gets on with something else. Nice one...



# 2. Choosing a host Mac or PC

*After the 2nd Edition of VirtualRPC In Use had been released, VirtualAcorn published a much-expanded article on Buying a new PC which included some particularly good advice on purchasing a PC 'off the shelf'. In this third edition, we'll cover a few happenings post-dating the VirtualRPC article, and add a few snippets from your feedback and our own experiences.*

*So for those of you aiming to buy an off-the-shelf PC and install VirtualRPC yourself, we very strongly recommend that you first study the VirtualAcorn article and then maybe look at the further discussion below.*

Buying a new computer can often entail a considerable capital outlay and it's important to get your choice 'right first time'. For Fans of RISC OS, there's an ever-widening range of both native ARM-powered hardware, and host computers on which to run VirtualRPC.

If you decide to go down the emulation route and purchase VirtualRPC, the next step is to get a host computer to run it on and the choice is either a Mac or a PC. Unfortunately we can't assist with specific ideas on picking an Apple but much of the following discussion will also apply to the Mac alternatives.

With PCs, the basic options are a conventional desktop 'tower', a laptop/notebook/'2-in-1' tablet, or a midi job (a.k.a. 'shoebox').

### Intel *Vs* AMD processors

As for selecting between Intel and AMD processors, VirtualAcorn recommend an Intel product for preference as their architecture particularly suits the VirtualRPC emulation.

### Turn-key solutions

If you're migrating from ARM-power, you may not be familiar with PCs and Windows, so you might prefer to opt for a turn-key solution where VirtualRPC has been pre-installed and configured.

In principle, you can then just switch on and go direct to RISC OS. However, the choice of suppliers is limited but, in the RISC OS community, R-Comp Interactive provide a very good range of interesting PC offerings.

If you start with *www.riscube.co.uk*, you can browse through their *RISCube*/*SpaceCube* 'midi' range which features a selection of dual- and quad-core processor PCs.

R-Comp also offer a range of tower and laptop PC host computers with VirtualRPC pre-installed. Those detailed at the same web address are only representative of what they can supply and they will be happy to discuss particular requirements for screen size, ports, etc.

In Section 4, we'll cover the additional PC 'anti-malware' software which R-Comp routinely pre-install and configure on all their products. We believe they are the only firm which provide this all-inclusive service.

### Off-the-shelf solutions

Provided you're happy to install and configure VirtualRPC yourself (and as we'll see in later sections, it is generally very straightforward), then the choice of desktop, laptop and tablet PCs obtained either through mail order or over-the-counter is rather more extensive.

Although purchasing direct from the chosen manufacturer can sometimes be advantageous, we've always found that a touchy-feely session in a retailer's emporium can pay dividends, especially if it avoids falling into a trap (to be discussed). On that basis, we would strongly recommend a visit to stores if at all possible.

So typically our routine is first to wade through likely models listed on the manufacturers' websites and draw up a shortlist of potentially-suitable machines. But don't slavishly believe everything listed on a manufacturer's website as they can occasionally be a tad 'economical with the truth'.



Return to Contents

This can work both ways. For example:

• **Processor clock speed** – Some laptop manufacturers gaily advertise their high-end processor clock speeds as (say) "3.1GHz" – which is probably true(-ish) – but fail even to mention that the laptop processor is actually being clocked at only 2.5GHz! This may be good for producing impressive numbers for battery life on one charge – but will surely result in a significant performance hit on both Windows and VirtualRPC (taken direct from a disgruntled punter's feedback...).

• **'Turbo-boost'** – In contrast, some high-end processors may be advertised as being clocked at (say) "3.2GHz" – but fail to mention that they may also have a 'turbo-boost' function which might temporarily push up the clock speed to perhaps as high as 4.0GHz. As far as we can tell, VirtualRPC doesn't seem to trigger a turbo-boost function. But if you should hear the usually-quiet processor fan speed increase a notch, that seems to be a clue that the Windows wick has been turned up, typically on a heavy 'number-crunching' job such as rendering a large video file. From our own experience, this can make for quite a significant Windows productivity improvement.

• **Number and range of ports** – Look out for some arguably sharp practice when ports are listed. For example, if a manufacturer's or retailer's website lists (on 3 lines) "6× USB", "3× USB3" and "1× USB Type C", that may well not mean you get a grand total of *10* USB ports... You may instead get typically a 4-way cluster of USB ports on the rear panel – of which only two are USB3 standard – and with the third USB3 and the USB Type C sockets on the front panel, totalling only ***6*** assorted USB sockets in all. Geddit?

Putting these 3 examples together, we suggest also having a look at third-party product reviews and other websites covering the same models as they tend to be rather more forthcoming on detail, plus of course you might also benefit from a closer look and a touchy-feely session in stores.

### Cables, connectors and adaptors

In recent years, a number of different types and sizes of ports have come into being. For example, with the range of laptops and (especially) tablets becoming increasingly slimline, manufacturers have had to resort to fitting smaller versions of otherwise standard ports, purely because of space constraints. For instance, a standard USB or video socket may be replaced by a 'mini' version. Or a multiplicity of video, ADSL and USB ports may be replaced by a single, mutifunction USB port.

Similarly, depending on the graphics card installed in a tower or midi-computer, it may come with several different types of video ports. At the other end, if you plan to use a standard 'HD' TV as your monitor, it will most likely have HDMI socket(s), whereas a dedicated computer monitor may have multiple, different types of video ports.

So you may well have to do some homework to determine exactly what is installed on a particular computer, and to ensure that it is physically possible to interconnect it to your planned peripherals and, if need be, obtain cable(s) with different connectors, or even resort to using one or more adaptors.

(Without wishing to recommend any particular source for cables, connectors and adaptors, we find Am*z*n are a convenient and reliable one-stop shop.)

Even connecting an aging device to a modern one may require an intermediate adaptor, for example to convert an analogue video output on a laptop to a monitor with a digital HDMI socket.

Tablets are likely to have a digital video output, in many cases a 'mini-HDMI' socket, and cables with a mini- and standard-sized HDMI plug at each end will do the trick.

However, some tablets may have a single, mini-'USB type C' socket which does not output video directly. In this case, multifunction adaptors are available which, for example, have

a mini-USB type C plug on a short flying lead to plug into the tablet, with a standard-sized HDMI socket plus multiple USB3 sockets on the adaptor.

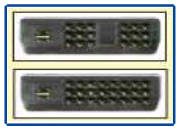
Although this may seem a bit of a hot-potch, user reports are generally complimentary, particularly with respect to the all-important compatibility thanks to the built-in 'plug-and-play' drivers.

Note that RISC OS does not support touch screens, so Windows tablets without a *physical* keyboard and pointing device will require a plug-in alternative such as an *'Air Mouse'* job.

To connect one or multiple monitors to (say) a 'tower' host PC, you need to determine which sockets are on the built-in graphics card. In principle the PC specification should tell you what to expect – but don't be surprised if what you get is not quite what you were expecting!

This isn't necessarily a stopper as if you finish up with a discrepancy, you should be able to source one or more suitable cables. But in this case, you may have to deal with a range of sockets on a video card, including VGA, HDMI, DVI * and/or mini-port(s).

* Be aware of a small 'gotcha' if you are upgrading to a high-resolution monitor (e.g. 2560×1440 pixels; discussed in section 6) and are connecting it to a DVI-D socket on the computer graphics card. To support the much higher data transfer rate required, you need to



ensure that the DVI-D socket has a *full* complement of pins (as in the lower illustration at left) *and* that you use a '*dual link*' DVI-D cable.

### Manufacturers, reliability and warranties

Do note all the good advice in the *VirtualAcorn article* regarding manufacturers. Sadly, most of the better-known names are no longer in the PC market and some other little-known firms have limited 'form' as to quality and reliability.

In particular, in case your PC breaks and needs to be sent away for fixing under warranty, do take on board those named manufacturers which, for reasons which VirtualAcorn explain (from experience), they do *not* recommend.

On the other side of the coin, some makers will provide their own extended warranties. For example, HP currently charge from only £29 for a 3-year warranty *including* collection and delivery; arguably suggesting they're very confident their products are reliable and unlikely to fail.

### Windows XP *Vs* Windows 7/8/10

Today's off-the-shelf machines will most likely have *Windows 10 (Home)* installed. We liked Windows XP a lot but, having now got used to Windows 10, are in general very happy with it (subject to the modicum of extra configuration needed to get VirtualRPC to work; discussed in Section 5).

However, when we came to plug all manner of kit into our first post-XP machine, we couldn't get some of our previous-millennium peripheral equipments to work, typically those going back to RiscPC days. In every case, this was because compatible drivers were not available for the old stuff and, not surprisingly, it's unlikely they will ever be released for Windows 7 or later.

So this is a significant point to bear in mind if, say, you're migrating from ARM-power or Windows XP, with dated peripherals. At today's prices, it may not cost the earth to replace the old kit such as a printer and a separate scanner with contemporary items such as an all-in-one, multi-function printer/scanner which will not only be compatible with Windows 7/8/10 – and VirtualRPC – but also higher-specification.

# 3. Choosing the VirtualRPC-*xxx*

For use on host Macs and PCs, there's a choice of VirtualRPC variants available from VirtualAcorn, each providing diverse levels of RISC OS emulation. A recent addition is a lower-cost/lower-spec entry-level version of VirtualRPC called *VirtualRPC-DL* which can be *D*own*L*oaded from the *RISCOS.com* shop.

The Mac variant is configured to the full, VirtualRPC-AdjustSA standard, but a table comparing the different specifications for the PC variants is on *www.virtualacorn.co.uk/product.htm* and may help you to select a product without further ado. But if that doesn't do it for you, we hope the following additional detail and thoughts may assist.

### Adjust or not?

Which PC variant to select isn't entirely a VirtualRPC decision, but what version of RISC OS you require. The VirtualRPC-DL, -SE and -SA options come with RISC OS v4.02, whereas the -AdjustSA variant comes with RISC OS v4.39. Furthermore, a range of upgrades is available, from v4.02 to v4.39, and onwards to v6.20, all available via *VirtualAcorn* and/or the *RISCOS.com* shop.

(The differences between RISC OS v4.02, v4.39 and v6.20 lie outside the province of this main booklet so we'll leave you to look elsewhere for them, but see also *Supplement 8*, page 30 in the Supplements booklet.)

In practical terms, we're very happy with both the RISC OS v4.02 and v4.39 (Adjust) options as we don't need or use many of the Adjust features anyway, plus a small number of our RISC OS utilities will run under v4.02 but not v4.39. It's swings and roundabouts.

However, looking at the table mentioned above, there are two significant differences which do fall within the VirtualRPC emulation: maximum RAM and the ability to read a data DVD under RISC OS (via CDFS):

• **Maximum RAM** – Unless VirtualRPC is pre-installed and the RAM is configured by the supplier/retailer, all variants come with a default setting of 32MB, drawn from the host computer's main RAM.

The setting in the three (PC) non-Adjust variants can be reconfigured via two separate RAM 'banks', up to a maximum of either 64MB (VirtualRPC-DL) or 128MB (VirtualRPC-SE and -SA), whereas the VirtualRPC-AdjustSA variant (Mac or PC) has a single RAM bank which can be set up to a maximum of 256MB (128MB for the *ARM7500* core).

In all cases, changing the RAM setting(s) is done under Windows but is a straightforward task which is fully documented; we'll look at it in a little more detail in Section 5.

(***Health warning: Please ignore*** *the mention in Edition 2 that VirtualRPC-SE and -SA can be set to 256MB RAM. This is fraught with potential danger, can lead to gnashing of teeth and is frowned upon by VirtualAcorn; see Support article No.158*)

• **Data DVDs** – Provided the DVD/CD drive in the host computer is of the older IDEFS standard, all VirtualRPC variants will tap into one optical drive running under CDFS within RISC OS, so is fine for reading and burning audio/data CDs. In the VirtualRPC-AdjustSA variant, CDFS also supports reading data DVDs, including commercial offerings such as the Archive and Foundation/RISCWorld compilation discs.

However, if you don't have the -Adjust variant – and/or CDFS is not compatible with an *ATAPI* optical drive in the host computer (increasingly likely) – it is still possible to access CDs and data DVDs under VirtualRPC's proprietary filing system (*HostFS*), by having a *HostFS* 'mount' on the icon bar for the optical drive (including an *ATAPI* variant). We'll look at the setting-up procedures in Section 6.

### StrongARM variants

A point to make clear is that, if you have RISC OS software which is "StrongARM compatible", you do *not* need any -SA variant of VirtualRPC to run it. In this respect, it's just like their ARM-powered counterparts where StrongARM-compatible software will still run perfectly happily on ARM 610/710 RiscPCs – or on earlier machines for that matter!

The only time you would need either -SA variant of VirtualRPC is if you happen to have any StrongARM-*only* software, which today involves very few relevant titles.

So what's the point of having an -SA variant? If you look at the *Products* comparison table, you'll see that they have various features not available to VirtualRPC-SE or -DL. Among these are: Windows clipboard support; a pop-up drive manager (making it somewhat easier to handle all the myriad storage devices you can run in the host PC and access under VirtualRPC); automode support for widescreen monitors; and file launching into Windows. Additionally, the *HostFS2* version of the filing system, plus the higher screen redraw speed, together provide a modest performance increase.

### Bigger and better displays

However, there is another notable benefit to having an -SA variant, and this is that there are *three* different RISC OS core emulations included within them: *Arm7* (representing the ARM710), *Arm7500* (emulating the A7000/A7000+ range) and *StrongARM*. The advantage of this isn't immediately obvious but, for us, it proves to be by far the most significant bonus of having an -SA variant, so it may be worth taking a look at it now, although we'll consider setting it up in Section 6.

The original VirtualRPC release came with 2MB 'VRAM' (drawn from the host PC's main RAM), so the possible display combinations were for all practical purposes identical to a RiscPC with a 2MB VRAM chip.

However, an early VirtualRPC upgrade – increasing the effective amount of VRAM to 8MB – was a very considerable improvement and, for example, led to a high-resolution display of 1600×1200 pixels being selectable, without the colour depth dropping from the maximum 16M (screen memory 7500KB, i.e. not far off the 8MB limit). So although this still fell short of the splendid display capabilities of Iyonix, in principle it was extremely welcome. But bearing in mind that the RISC OS display is emulated, there was some evident 'hesitation' in displaying the higher-resolution screen modes with the then processors such as the Pentium P4.

But contemporary processors such as the Core2Duo range and later have more than adequate 'oomph' to support a RISC OS screen mode of, say, 1600×1200 pixels, and also widescreen modes of, say, 1680×1050 pixels, both in 16M colours. Plus there's often enough spare grunt to support even higher-resolution screen modes – the problem then being that the screen memory required would exceed the configured 8MB VRAM limit.

This is where having the three different ARM core emulations in the -SA variants comes in. With ARM-powered machines, the A7000/+ were unique in that they drew the VRAM from the machine's main RAM and the screen memory limit was increased to (we think) 16MB. This is faithfully emulated by the *Arm7500* option within the two -SA variants.

So if we select Arm7500 at VirtualRPC start-up, we can now run in, say, an 'HD' widescreen mode of 1920×1080 pixels (screen memory 8100KB), a 4:3 mode of 2048×1536 pixels (screen memory 12288KB), and perhaps even more, *all in 16M colours* (discussed in Section 6), the only limitation then being the maximum setting of 128MB RAM.

So for this factor alone, we feel the modest extra cost of buying either -SA variant is money *very* well spent.

# 4. PC malware issues

*Once upon a time, the topic of PC malware was a source of near-continual discussion in the computer press, primarily because of the perceived risks to safe operation. However, since Edition 2 was published, there seems to have been a general acceptance that the hazards are much less than previously thought – or perhaps never existed at the high levels then stated.*

*So although this section has had only a modicum of amendment from the previous edition (so that the range of potential risks and solutions are still brought to your notice), maybe we could reasonably apply an element of commonsense on what level of protection we opt to apply.*

*For example, Windows 10 now contains its own built-in anti-malware protection – Windows Defender – and, by all accounts, many people rely on this alone. Whether or not you decide to add further protection is your own decision, but we do advise running though this section to see what alternative software is available.*

Quote: *"... one look at the news should convince you that there are some really nasty people out there and more than a handful of rather stupid ones. Somewhere in amongst those groups are the clowns who write computer viruses. ... There are several unpleasant viruses out there that will wreck your disc-based applications and crash your machine if you give them the chance."*

That quote is taken from (wait for it...) Archive 6:1, October ***1992***, in a review of Pineapple's anti-virus *Killer* programme. So any RISC OS user who complacently scoffs at and criticises Windows for being wide open to malware and stuff, and thinks RISC OS has never suffered from viral infections, simply doesn't recall or know about all the dreadful problems we used to have in those earlier days. Nor the expense of getting rid of the pesky things.

So before delving into this topic, we should first strongly advise potential VirtualRPC purchasers *not* to be swayed by the active anti-Microsoft brigade, a.k.a. 'the knockers'. We're all well aware of – and remain frustrated by – the small but very vocal group of RISC OS users who routinely criticise PCs and (especially) Windows, in the magazines and on forums etc.

It's only when you enter into a dialogue with them that it becomes clear that in many cases they have little or no experience of PCs and Windows, and are either repeating seriously-dated things they've heard or read (i.e. hearsay), or are simply misunderstanding the facts, both of which sometimes can even lead to misrepresentation.

In the past this didn't really matter – they are welcome to their personal views – but over the years we've known of quite a number of people who would otherwise consider VirtualRPC as a replacement for (typically) their aging Iyonix or RiscPC, but who have been frightened off by this unceasing and generally ill-informed criticism. So we'll try to put a sense of perspective into what is being said.

The knockers' manual covers quite a range of topics and we'll address at least some of them in Section 14 but, for the moment, we'll look at the most troublesome: PC malware.

If the doomsters are to be believed, then the instant you get onto the net, all manner of viruses will enter your PC and create havoc, slowing down the computer, filling up your disc, and so on. The screen will fill with unwanted ads, seriously affecting your work and enjoyment. Most worryingly, while you're not looking, nefarious beings will delve into the innermost secrets of your machine and – hey presto! – any semblance of privacy is lost.



Should we believe any of that? Two points can be made here. First, there are many 1.000s of VirtualRPC registered users, the bulk of whom are running it on host PCs – yet when did you last hear of any significant tale of woe caused by PC malware?

Second, the 4 guys manning our previous business at T.O.M.S. ran typically 8-10 PCs between them, on average more than 25 years each (i.e. *over 100 man-years* in total), in both business and domestic settings – and yet (as far as we know), never once suffered from the effects of PC malware. And most certainly far, far less than under early RISC OS.

Surely both points are trying to tell us something? The answer of course is that either the horror stories were simply nothing like as bad as they were painted, and/or we were all successful in protecting ourselves. Which it is simply doesn't matter; we'll presume the worst and make sure we take sensible precautions.

### A typical anti-malware suite

To ensure safe computing and, especially, safe surfing, we could usefully use a small number of anti-malware applications on our PCs, ideally without their becoming in any way intrusive and, if possible, at minimal cost. So in the following sub-sections, we'll look at the types of threat in turn and suggest one or two competent, freeware solutions for each group.

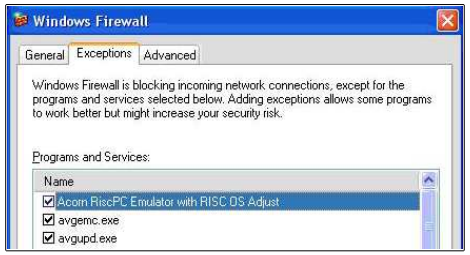
Indeed, PCs running Windows XP or later already had a 'firewall'; more on that in a moment. Similarly, PCs invariably come with a trial version of a commercial anti-virus programme pre-installed, typically *Norton* or *Macafee*. The problems with these two are that they're quite resource-heavy and, being loss leaders, the free updates run out after a couple of months – maintaining them thereafter can prove very expensive. So we suggest perhaps using either one initially but, at a convenient time, replacing it with a freeware alternative.

### Firewall

What a computer firewall actually is and what it does is left a little vague! However, that needn't worry us as all we have to do (in Windows 10) is to select Start › Control Panel › Security & Maintenance, click on Security and check that the Network firewall and the Virus protection (Windows Defender Antivirus; more on that shortly) are both set to On.

If the firewall or Defender virus protection somehow becomes turned off, a warning message will leave you in no doubt that it is.

Second, to enable VirtualRPC to access the net via RISC OS, it needs to be included in the list of 'Exceptions'; i.e. those programmes *not* blocked by the firewall. A window opens to set that when VirtualRPC is run for the first time after being unlocked but, if you hit problems, click on the Exceptions tab and make sure the box is ticked. Other than that, the firewall needs no further attention.




### Anti-virus protection

To protect against a viral attack, we do need to install an anti-virus programme, configure it to ensure it's on the lookout for nasties and –



Return to Contents

*most importantly* – make sure that its virus 'signature' file (which recognises incoming viruses for what they are) is kept up to date on at least a **daily** basis.

Note again that Windows 10 already has the anti-malware programme – *Windows Defender* – built in and that Microsoft recommend that you should run only *one* anti-virus programme as, otherwise, the virus signatures file for one application may be misconstrued as real viruses by another!

However, in our experience, running *Windows Defender* in parallel with any one of the anti-malware programmes discussed below does not appear to have any nasty side-effects. (Although not confirmed, it may be that *Windows Defender* recognises that you have another virus programme running and puts itself into standby mode.)

There's a good selection of other effective, freeware anti-virus titles available and, for preference, we always install the free versions of either **AVG** or **Avast**. These and all the other programmes we'll recommend are available from a number of reliable, freeware libraries such as *www.filehippo.com*

Installing either programme is simply a matter of downloading the installer, double-clicking on it and letting the 'Wizard' do its thing, accepting all the default options offered.

However, up to a few years ago, there was a potential problem with *AVG Free* and, indeed, the other anti-malware programmes we install. This was that, as by default they're configured to download and install any updates *automatically*, there used to be a distinct possibility that this might interfere with VirtualRPC when it was running in full-screen mode.

Normally that resulted simply in VirtualRPC dropping into a window, but there were reports of *AVG Free* interfering with programmes being used under RISC OS.

Arguably the easiest way to prevent this was to disable the automatic *AVG Free* updates and perform manual ones instead. However, in the ensuing years, the then Pentium P4 processors have been superseded by the Core2 and later ranges, with very considerably more computing power available in all but the most basic PCs such as some netbooks running Celeron or Atom processors.

In practice, this seems to have virtually eliminated the interference between VirtualRPC and background activity, such as an anti-malware programme running an automatic scan or update.

### Anti-malware scans

On first installing either programme, it invites you to conduct a full system scan to check if something nasty already exists on the computer and to deal with it.

Thereafter, all the anti-malware programmes will give routine, 'on-the-fly' protection against incoming nasties and, additionally, will update the all-important virus 'signature' files on at least a daily basis. Once again, that could often prove to be intrusive, especially if it did it while you were doing something important. But with the advent of more powerful processors, the background activity now seems to have little or no adverse effect on normal usage.

### Anti-spyware protection

'Spyware' is a bit of a catch-all term for software which seeks to perform privacy-invasive actions on your PC, usually without your knowledge. 'Adware' which downloads and displays intrusive advertisements is a typical (and visible) sub-set of spyware, but others are potentially rather more serious.

Fortunately, as with viral attacks, a range of good, freeware titles are once more available to us, such as **Spybot Search & Destroy** or **SpywareBlaster**, both being available for download from *www.filehippo.com*



In this case, it is permissible and indeed often recommended that you install perhaps a couple of anti-spyware titles so that anything missed by one might well be intercepted by the other. Furthermore, *AVG* now protects not only against viral attacks, but also has its own built-in anti-spyware routines – and still all for free. So on that basis, we install *AVG Free* plus *Spybot Search & Destroy* on our PCs.

(Note that the paid-for *Home* and *Pro* versions of *Spybot Search & Destroy* also include anti-virus protection but that this does not apply to the freeware version.)

### 'Adware' killer

Although not a significant risk to our PCs, 'adware' which clutters up your browser display in particular with unwanted adverts can be a thorough nuisance, but the freeware *Adaware Ad Block* can go some way to improving matters. Against that, many websites depend heavily on income from advertisements, to the extent that they may insist you disable your ad blocker before letting you access their sites.

### Further privacy measures

So far so good. But it has to be accepted that, when surfing the net (regardless of the browser we use: RISC OS or Windows), a great deal of information can be stored on disc relating to our browsing activity, in the form of sites visited, cookies, and so on – even our names and other personal information. This isn't to be entirely knocked though as, in principle, it can greatly speed up re-accessing the same sites and features in the future.

However, it does unfortunately rather give the game away as to who we are, where we've been and what we've done. So despite the best efforts of the anti-spyware programmes, our browsing history can potentially still be compromised. And of course, as we realised following an office burglary, it's all stored on the (stolen) hard discs...

So on that basis, perhaps it's not a bad idea to flush the various internet-related data files every now and again.

On the face of it, this is very easy under Windows as there's a feature for deleting all the temporary storage files, either globally or in selected groups such as the cookies file.

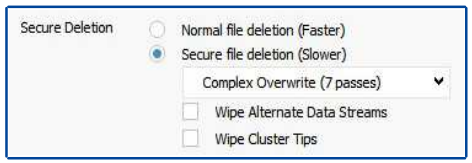
Unfortunately, it isn't conveniently accessible and, in any event, various files called *Index.dat* are *not* deleted (except in Safe mode which also isn't convenient) and they do still contain our 'footprint'. So what's to be done about that?

Yet again there are competent, freeware programmes to do all this. We choose to use the highly-regarded **CCleaner** (previously 'Crap Cleaner'!) and available from – yes, you've guessed it – *www.filehippo.com*

Installing it is just as easy as the others and, in standard use, you simply need to click on the Analyse button (illustrated on page 18), wait for it to scan the various data files, then click on the Run Cleaner button to flush them.

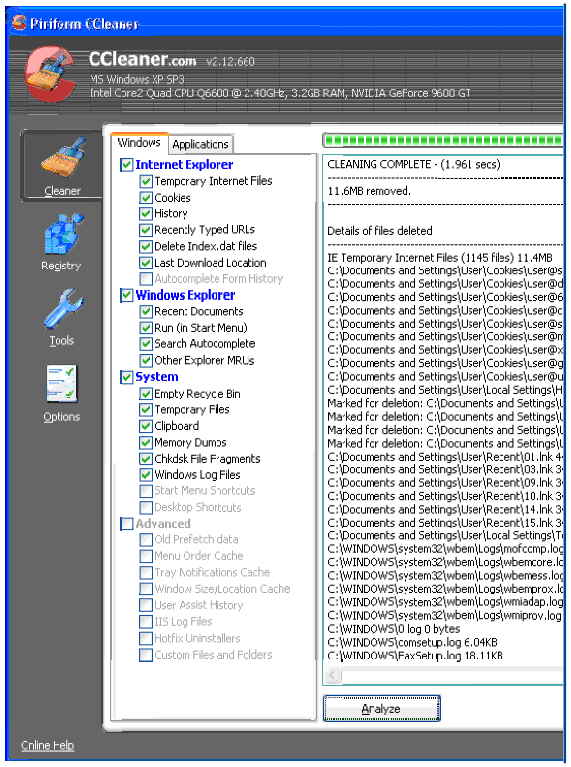
It's important to note that *CCleaner* doesn't just 'delete' the selected data files (which would simply remove their titles from the catalogue but leave the files untouched – and therefore vulnerable). In fact it repeatedly overwrites the files with 'scribble' to render them quite unreadable.

But you do need to pre-configure how many times they are overwritten by clicking on Options and Settings, then making your choice. We always pre-set it to Complex Overwrite on the basis that 7 passes seems to be a reasonable compromise between risk and overwrite speed.






*Return to Contents*




Clicking on the Options button (illustrated at left) and then on Cookies will display a list of all those collected since the last clean-out, under the heading Cookies to Delete (illustrated below).

But if you wish to retain any particular cookies, select them and click on the –> button, when they will be moved across into the Cookies to Keep column.

Similarly, any you've previously kept but now wish to delete can be moved across by using the associated <– button.

## Registry issues

Although the second added-extra feature doesn't come under the malware heading, it's convenient to mention it now as *CCleaner* can also assist.

Windows has what's termed a 'Registry' which, as the name suggests, is a central repository, cross-referencing information about our computer, what application software is installed, which drivers we have, where they are, and so on.

In theory, the Registry will be updated automatically as and when we do something significant, such as uninstalling a programme or updating a driver.

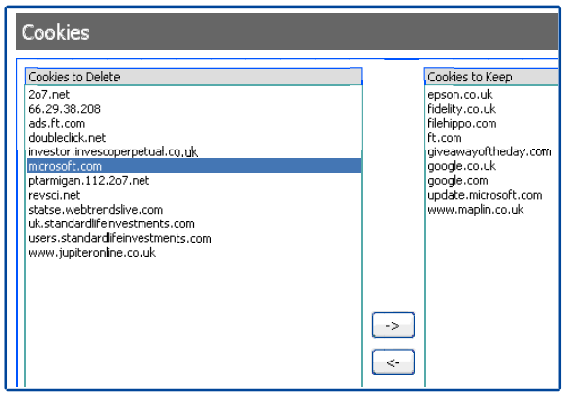
Note in the illustration above that, by default, the tricky *Index.dat* files will be flushed. But if you wish to retain the data in any other files, you can first *UN*tick the relevant box(es) listed in the left-hand column.

## Cookies

There are two added-extra features of *CCleaner* which can be rather handy.

First, in normal operation, you can improve security by setting it to flush all 'cookies' collected since the last time *CCleaner* was run.

However, it can be useful to retain any cookies which you know are 'safe' and which can help to speed things up next time you visit particular websites.



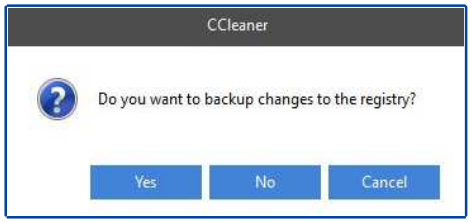

However, just like a real-life registry, it can get out of kilter with what's actually going on – and the knockers would have us believe that a 'broken' registry can wreak havoc.

In our experience, this simply doesn't happen. Indeed, arguably there's a far greater risk in using a commercial but unproven programme (of which there are many) to 'clean' your registry – only to find afterwards that things have gone awry and which even doing a System Restore may not correct (Section 14).

However, if you should suffer symptoms such as the start-up or (especially) the shut-down time becoming dragged out whilst the Registry looks for something that isn't there, it certainly won't hurt to have it checked out *by a known, reliable programme* such as *CCleaner* and any anomalies removed.

In these circumstances, it's very easy to do with *CCleaner*. Simply click on the Registry button (illustrated on page 18), then on Scan for Issues, let it do the scan and then follow the Fix selected Issues prompts to remove the problems. (Don't be concerned if sundry 'Missing or Shared DLL's come up for VirtualRPC-related issues – it doesn't seem to matter!)

The major safeguard with *CCleaner* (which doesn't exist with most of the other offerings) is that, having first listed all the anomalies in the registry, it then gives you an opportunity to save the *existing Registry* file before fixing them.




Saving the file to the default *CCleaner* folder is a convenient location as, if you find things have gone adrift in the 'cleaning', simply double-clicking on the newly-saved file will restore the Registry contents back to where they were.

(As this topic was being drafted, a perfect example of the beneficial use of *CCleaner* cropped up. A recent Windows 10 update clashed with a utility and routinely triggered an annoying error message at shut-down, so the utility was uninstalled and another offering used instead.

However, from then on, every time the host PC was started, the original utility triggered an equally-annoying message to re-install it... So although the uninstall has seemingly worked, evidently a trace was being left behind. Perhaps in the registry? Evidently so as running the Registry option in *CCleaner* and selecting Fix selected Issues did indeed fix the problem. Sorted.)

## Don't be put off...

The fact that this topic runs to 6 pages should *not* be taken to mean that the myriad horror stories about PC malware and registry issues are in any way a significant problem in practice; they aren't. Nor should you let the thought of having to obtain, install and pre-configure the various applications put you off; *www.filehippo.com* is a (free) one-stop shop and the friendly 'wizards' do most of the work for you.

However, someone once thoughtfully asked whether a PC could be obtained with a range of all these freeware, anti-malware programmes already pre-installed and pre-configured? The answer is "Yes": from R-Comp Interactive (page 9).





Return to Contents

# 5. Setting up a PC and VirtualRPC

### VirtualRPC and Windows Vista or later

If you're migrating to running VirtualRPC on a host PC with Windows Vista or later (from either ARM-power or from a Windows XP host), there may be one or possibly more extra actions which you need to take *before* installing and then running VirtualRPC (any variant).

This is to enable VirtualRPC to be successfully installed in the first place, and then to be run either without failures to start up, or without pop-up error messages – or worse...

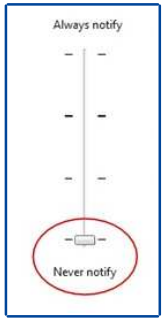
The techie reasons for all this are complex but, fortunately, we can simply implement the necessary action(s) without the need to fully understand what's going on under the bonnet.

These once-only actions – with the techie reasons – are included within VirtualAcorn's increasingly-wide range of *Support articles*, but in essence they boil down to the following:

• **User Account Control (UAC) Setting** – This requirement depends on the date of your master disc; specifically whether or not it pre- or post-dates the 2014 general re-issue discussed on the *Product Registration* page on the Virtual Acorn website. The version of your master disc is easily determined by seeing whether it looks like the Old or New format illustrated below.

If the disc is the New version, then the UAC Setting is automatically fixed during VirtualRPC installation and can be disregarded.

But is it's the Old version, you need to select the Support Article for the version of Windows installed in the host PC (listed in the Product Registration page link above) and follow the procedure as set out in the relevant support article (*122* for Vista; *127* for Windows 7; or *133* for Windows 8 or 10).




In all cases, you should finish up with the UAC Settings slider at the lowest, Never Notify end of the scale (ringed). Finally, close down and restart the computer for the change to take effect.

The end result is that you will now be able to install VirtualRPC, and start it, without Windows throwing a wobbly because it doesn't recognise this strange 'foreign' application.

• **VirtualRPC installation path(s)** – In the early years of VirtualRPC, the application came on 'silver disc' CDs and was set up so that, by default, a new folder called *VirtualAcorn* was generated in the root of *Drive C*: and then VirtualRPC-xxx was installed within the *VirtualAcorn* folder. In other words, the default leafname was *C:\VirtualAcorn\VirtualRPC-xxx.*

In principle, it was (and still is) possible to amend the leafname during set-up but VirtualAcorn firmly recommend that – unless you really know what you are doing – you do *not* alter the default installation path. (This is discussed in more detail in the Supplements booklet, *Supp., 3, page 12.*)

However, as from 5th September 2014, the VirtualRPC -SE, -SA and -AdjustSA variants have included over 200 improvements and have been shipped on 'white disc' CDs.



***Health warning!*** The installation process from the 'white' CD may appear to be unaltered – but it's important to note that there has been a subtle simplification in the default leafname in that the extra *VirtualAcorn* folder is no longer generated and VirtualRPC-xxx is now installed in the *root* of Drive C: – in other words, the leafname has been simplified to *C:\VirtualRPC-xxx.*

Although VirtualAcorn provide an upgrade path to bring your earlier copy/ies of VirtualRPC up to 'white CD' standard, it is not essential that you do so as your existing VirtualRPC installation(s) will continue to work fine, albeit without the improvements,

*But* there is now a potentially significant trap for players. For example, if VirtualAcorn issue an update (for any variant) and you attempt to install it, by default the installer will now try to embody it in *C:\VirtualRPC-xxx* – even though your working copy is still installed in *C:\VirtualAcorn\VirtualRPC-xxx* – and the update will fail to install. Drat...

So even if you elect not to upgrade from the 'silver CD' to 'white CD' standard, you are recommended to move your present copy of VirtualRPC-xxx into the *root* of Drive C: so that the leafname now becomes *C:\VirtualRPC-xxx.*

However, bearing in mind that this is a rather fundamental change, you might prefer to use a 'safe' method (to be discussed in more detail later) which is to select your existing installation (in *C:\VirtualAcorn\VirtualRPC-xxx*) and then press *‹Ctrl-C›* to copy it onto the clipboard, go up one layer of the tree to the root of Drive C: and now press *‹Ctrl-V›* to paste the copy of VirtualRPC-xxx from the clipboard into *C:\VirtualRPC-xxx.*

This will leave the original copy of VirtualRPC-xxx in the *VirtualAcorn* folder – but if you rename the folder as (say) *VirtualRPC-xxx (BackUp)*, you will now have a *back-up* copy of the default VirtualRPC-xxx installation readily to hand.

• **Windows 'Event viewer' (Vista or later)** – Should you hit a problem starting VirtualRPC where the activity icon starts up – but then stalls after a second or so (repeatable) – this could be for one or more reasons and, in the first instance, you will need to work through Support *Article 157* to pin down the culprit 'Faulting module(s)'.

• **Faulting module 'VirtualRPC-xxx.exe'** – This problem is triggered if the PC's audio adaptor is not active and is discussed in Support *Article 141*. An easy solution is to activate the audio adaptor by inserting either speakers or headphones in the appropriate socket, when all should then come good.

• **Faulting module 'Atapicd.dll'** – This topic affects Windows 8 and 10, and will apply if your host PC has an *ATAPI* optical drive installed. During VirtualRPC start-up, the RISC OS *CDFS* drivers will clash with Windows and, after a second or so, start-up will stall. The solution is covered in Support *Article 143* and involves *moving* the VirtualRPC *AtapiCD* folder from its default location in the *PlugIns* parent into a new *PluginsDisabled* folder.

VirtualRPC should now start up without complaint – but of course it will not now be able to access the CD/DVD drive and, if you click on the CD icon, it will return a `Soft-load driver not present` error message. However, data DVDs and CDs can still be accessed by having a *HostFS* 'mount' for the optical drive on the icon bar (page 12). The simple procedure for setting that up will be covered in Section 6.

• **Faulting module 'ntll.dll'** – This problem is discussed in Support *Article 142* and entails setting up VirtualRPC in 'compatibility mode'.

• **Solid State Discs (SSD)** – Many relatively modern ARM-powered computers and host Macs and PCs may have an ultra-fast SSD installed as the main Drive C: along with a conventional hard disc drive (probably fast SATA3) installed as Drive D:



*Return to Contents*

However, although the HDD will have oodles of capacity – at least 500GB, and these days very likely 1TB or more – the SSD capacity may perhaps be 'only' 128GB or 256GB. But this is more than ample to have Windows, VirtualRPC and all your programmes (*including RISC OS applcations*) installed on this ultra-fast device, making best use of the sheer performance of the host PC.

***Health warning!*** For some unexplained reason, a couple of users installed VirtualRPC on their SSD Drive C: – but them moved *HardDisc4* onto the hard disc Drive D: leading to several knock-on problems. The lesson is clear: *Do **not** separate the HardDisc4 folder from the other VirtualRPC-xxx folders* (*Models, Plugins, RomSets,* etc.)

## Setting up VirtualRPC

Assuming you've installed VirtualRPC yourself and have completed any necessary actions covered on pages 20-21 when installing it, there's a very natural inclination to fire up VirtualRPC, get into RISC OS – and totally forget about Windows!

This is perfectly OK *provided* you ensure an anti-malware suite as discussed in Section 4 is installed and that everything is set up to do the all-important, daily check for 'signature' file updates.

And as discussed on page 16, it used to be recommended that the daily updates should be done manually (so as to avoid clashes with whatever else you were doing on the computer, especially if using RISC OS applications), but the advent of much more powerful processors has in effect rendered this safeguard unnecessary, so automatic signature file updates are now accepted as the norm.

During VirtualRPC installation, the user-guide (in HTML format) is conveniently pinned to the Windows desktop. The manual has always been very satisfactory and has been extended over the years. It is also supplemented by the useful Knowledge Base articles in the Support pages at *www.virtualacorn.co.uk/support/articles/index.htm* and, incidentally, absorbing some of the topics discussed in the early VirtualRPC In Use articles published in Archive mag from 2003.

Quite reasonably, the manual assumes the user will have a basic knowledge of PCs and Windows but, in the early years, it became noticeable that a number of problems had surfaced in this particular area.

From forum discussions and our own observations, the underlying reason seemed to be that more and more previously ARM-powered machine owners who were migrating to VirtualRPC had no prior experience of Windows and, perhaps not surprisingly, were being caught out by the sometimes subtle differences, especially in filer actions.

So we'll deal with that in the next part of this section, before going on to have a look at some of the topics in the original articles in Archive mag which are not included in the VirtualRPC manual, including some suggestions for alternative methods of achieving the same ends.

### The Windows filer

Ideally, it would be nice not to have to get involved in using the Windows filer at all but, on a small number of occasions, it is necessary to do so in order to configure VirtualRPC beyond its default condition (e.g. increasing the RAM available to RISC OS).

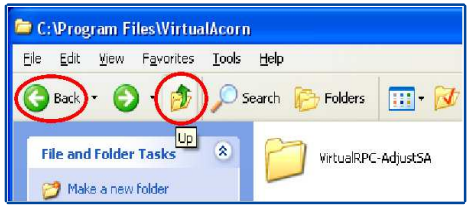
Similarly, some RISC OS settings can be tweaked either under VirtualRPC *or* under Windows and, arguably, the latter option can sometimes be a tad easier to implement.

That said, it is most certainly not necessary to have a detailed knowledge of Windows in order to set up VirtualRPC. If you're familiar with RISC OS, you'll find the Windows filer is for the most part very similar, so in the following sub-sections we'll concentrate on those small differences which potentially can cause hiccups.



A minor variation is that when RISC OS refers to a *directory*, containing files, sub-directories, and so on, under Windows the term *folder* is used synonymously.

• **Navigating the directory tree** – When going up or down a tree in Windows, and in order to close one folder (directory) viewer as the next one opens, it isn't necessary to remember to use the ‹Adjust› button – there isn't one! Going down the tree, we need only to double-click on a folder to open its contents, whilst at the same time the 'parent' folder is closed automatically. Similarly, going up the tree, clicking on either the Back *or* Up icon will automatically close the 'offspring' folder.




• **Application directories** – Similarly, under Windows, it isn't necessary to remember to press the ‹Shift› key in order to open a RISC OS application directory. Indeed, Windows thinks that all RISC OS directories – including applications preceded by a *!* ('pling') in RISC OS directory viewers – are just standard Windows folders, in all respects. This includes not only the main RISC OS *!Boot* file, but even the main '*HardDisc4*' folder. The latter is *not* a special partition (which, as we shall see, makes backing up your RISC OS files a doddle).

• **Drag-and-drop** – Don't believe the knockers who may tell you that Windows doesn't support drag-and-drop. It does – but admittedly nothing like as consistently as the splendid RISC OS feature. The most obvious shortfall is in not being able to save a file to a particular folder, by drag-and-drop. But equally, certain Windows features and application software make far better use of drag-and-drop than do the RISC OS equivalents, so it is rather variable. We suggest try-it-and-see; either drag-and-drop will work (under Windows) in the way you're used to or, if it doesn't, do it another way.

• **Selecting multiple objects** – With no ‹Adjust› button available as such under Windows, it can help to become familiar with the methods for selecting multiple objects prior to copying/moving/deleting etc. If the objects are contiguous, we can either use a conventional select-drag box, or select the first one in the group, press ‹Shift› and select the last object in the group. This selects the whole group, as a package. But if the objects are not contiguous, select the first as normal, press ‹Ctrl› and then select the other(s), singly. Selected objects can also be *de*-selected, individually, by pressing ‹Ctrl› and clicking on them. Once the desired objects are selected, then drag-and-drop, copy & paste, etc, will work as described below.

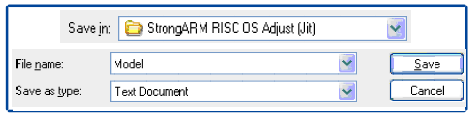
• **Copying/moving objects** – Windows is also not as consistent as RISC OS when copying or moving files from folder *A* to folder *B*, and the small differences are well worth getting to grips with to avoid gnashing of teeth. In essence, under Windows, if you drag-and-drop an object (or package) from one folder to another ***on the same storage medium***, the object with be *moved*, not copied! So to *copy* an object from *A* to *B* – on the same storage medium – it is necessary to press ‹Ctrl› and then use drag-and-drop. But oddly, this is not necessary if copying from one storage medium to another, where drag-and-drop works just as you're used to under RISC OS. So if that seems to you rather confusing (and we fully agree...), a consistent and 'safer' method is to select the object(s) to be copied, press ‹Ctrl-C› to copy to the Windows clipboard, click inside the destination folder and press ‹Ctrl-V› to paste the object(s).



Return to Contents

• **Copy to folder** – Windows has a neat feature which, in principle, can be very useful and saves having to open the destination folder when copying or moving object(s) from *A* to *B*. If you drag an object (or package) and simply drop it *over* the destination folder icon, the object/package will be saved inside it (just like dragging-and-dropping objects over the HardDisc4 icon on the icon bar of an ARM-powered machine). Nice one. However, there is potentially a gotcha to this Windows filer action which needs a little care. If you're moving something into a folder viewer and – inadvertently – drop it over a *sub*-folder, it will be saved into that sub-folder... This can cause much head-scratching as the object(s) will have left the source *A* folder, but will seemingly *not* appear in the destination *B* folder! Don't panic, it will be lurking somewhere inside one of the sub-folder(s).

• **Saving files** – When saving modified files under RISC OS, the easiest way is to press F3, which normally re-saves the file back from whence it came. However, there's no rigorous confirmation (other than the asterisk in the header), so can you be sure? If you'd prefer the 'safer' option (and we always do), you can of course select File › Save document and drag-and-drop the Save icon. Exactly the same argument applies to the Windows filer: to re-save a modified file, the fast ways are simply to press ‹Ctrl-S› or to select File › Save – an additional factor being that as PC hard disc drives are so extraordinarily fast, you may not even spot the briefest flash of the activity LED! So in this case, as a safer option, we always select File › Save As which opens up a dialogue box. Then we can double-check the file name and destination are as intended, before (re-)saving.




When working under RISC OS, and especially under Windows, invariably we use the safer, more rigorous method. The peace of mind it gives – especially if modifying a crucial object such as a VirtualRPC configuration file – more than justifies the few extra moments it takes to do it.

• **Recovery from finger trouble** – Another rather neat feature of Windows has saved us increasingly-decrepit users a lot of inadvertent heartache over the years! Under RISC OS, if you click on a button or icon, the relevant action takes place immediately – whether or not it was the right button or icon... Drat. However, if you click on a button or icon under Windows, the relevant action does not take place until you *release* the select button. So if you should realise that you're about to make a boo-boo, *keep the select button pressed*, gingerly drag the pointer to somewhere 'safe' – and release the select button. With any luck, you will neatly avoid making the aforesaid boo-boo... Phew.

### Summarising so far...

Don't be put off by what may seem the tricky differences in the Windows filer. As ever, the various elements take far longer to describe on paper than to get used to in practice. Nor is it our intention to cover all the ins and outs of the system. But if you're coming to VirtualRPC with no prior experience of Windows, then the above detail should be sufficient to enable you to dig into the Windows tree and configure VirtualRPC for maximum productivity.

## Customising VirtualRPC

Now to the meat of this section; configuring the combination of VirtualRPC and RISC OS to make best use of the various features and to achieve what you expect from them.

Because we'll necessarily have to make at least some of the changes to the VirtualRPC files via Windows, it's important to stress that – even if you make a total hash of it – you can't (physically) break anything; the very worst that can happen is that you'll need to reinstall VirtualRPC from the CD.

### Making a selective folder/file back-up

But to avoid this (and as an added safety precaution), we do recommend that, before making any changes to the VirtualRPC files under Windows, you first make a back-up copy of them, as it's then far easier to replace a corrupted file with the 'clean' back-up copy. To do that, first navigate down the PC's Drive C: tree until you come to the folder containing all the VirtualRPC goodies, i.e. *C:\VirtualRPC-xxx* for the variant you have installed.

The *VirtualRPC-xxx* folder contins all the files for the VirtualRPC installation, including the main *HardDisc4* folder, the *VirtualRPC.exe* run file, plus various configuration folders and files to enable VirtualRPC to do its thing.

If you're just making a simple copy of these configuration files (making a full backup of the complete VirtualRPC installation is covered in Section 13), we suggest making a new sub-folder called, say, *BackUp*, in the same directory. To do that, *right*-click anywhere in the folder viewer and select New › Folder. A fresh folder called *New Folder* is generated which you can now rename to read *BackUp*.

To make a back-up of selected files, press ‹Ctrl› and click on (we suggest) the *Models*, *Plugins*, *VA* and, if installed, the *VAclipboard.dll* folders/files, as these are the ones we'll either fiddle with or can conceivably become corrupted in use. Press ‹Ctrl-C› to copy them to the Windows clipboard, then double click on the *BackUp* folder to open it and, finally, press ‹Ctrl-V› to paste the pre-selected/copied objects into it.

### Configuring RAM for RISC OS

We'll start with setting up the RAM available for use under RISC OS as this does require us to access the relevant file under Windows, so is a good example of observing the various points discussed earlier in this section. And as it happens, the variations aren't covered in detail in the manual, so here goes.

All variants of VirtualRPC for Windows come with the available RAM set at 32MB, by default.

With RISC OS v4.39 (i.e. -AdjustSA), this is set in a single 'bank' and is relatively easy to change, up to a maximum of 256MB.

With RISC OS v4.02 (the -SE and -SA variants), there are two separate banks, each set to 16MB. Normally they may be increased, equally, up to a joint maximum of 128MB.

With the -DL variant (again running RISC OS v4.02), the maximum RAM whch can be configured is 64MB.

But note that, in all cases, the amount of RAM installed in the host PC must be at least *double* that to be used under RISC OS. This was occasionally a significant restriction but modern PCs are very likely to far exceed the requirement without the need to fork out for a larger-capacity or additional RAM module.

### Changing RAM for RISC OS v4.39

First, navigate down the hard disc tree to the main VirtualRPC folder in *C:\VirtualRPC-xxx* and double-click on the *Models* folder. With VirtualRPC-AdjustSA, the *Models* folder will contain three sub-folders, corresponding with the three VirtualRPC core emulations.

Double-click on the one you're most likely to use as a matter of routine (say *StrongARM*), and this will reveal the relevant *Model* file, in this instance the one in the *StrongARM RISC OS Adjust (Jit)* folder.

Having located the relevant *Model* file, double-click on it and, hopefully, it will be auto-loaded into the default Windows text-handling application – *Notepad*. (This is (similar to *Edit* in RISC OS).

***Health warning!*** *It is very important to ensure the Model file is not loaded into any other Windows text editor (e.g. Wordpad) as the Save configuration can be different from that of Notepad and the modified file may not be properly saved in plain text.*

*Arguably the safest way to ensure the Model file is loaded into Notepad is to right-click on the file, select Open with... and then click on Notepad and OK.*

Now we see that the *Model* file is in fact a user-readable plain text-file and, a few items down, the `[Rom]` configuration is set to `16Mb`, followed immediately by a single `[Ram]` 'bank', set by default to `32Mb` (ringed).

```
[Rom]
filename = "Romsets\RISCOS Adjust"
base = 0
range = 16Mb

[Ram]
base = 0x10000000
size = 32Mb
```

You can now edit the `[Ram] size` to suit your needs, up to a maximum of `256Mb` (`128MB` for the ARM7500 emulation), remembering then to re-save the *Model* file using any of the three methods discussed on page 24.

Then when you next start up VirtualRPC (in this example selecting the `StrongARM` option if you have the -AdjustSA version), you should find you have available the total RAM which you've just configured.

(*Note:* A general feature about VirtualRPC running on a powerful, contemporary processor such as a Core2Duo or later, is that RISC OS fires up so fast that you may not even get to see the start-up screen! So if the display blanks out completely for a couple of seconds, *don't panic* – the next thing you'll see is the RISC OS desktop. Then you can double-check the available RAM in the Tasks display.)

### Setting 128MB RAM for RISC OS v4.02

*(This procedure is also contained in the manual; Appendix 2 – Advanced configuration).*

Navigating to the *Model* file within the VirtualRPC-SE or -SA variants is identical to the procedure described for RISC OS v4.39, noting that the -SA variant will also have three optional emulations, so again you need to select the one you're routinely likely to use.

Having located the relevant *Model* file, double-click on it to load it into the Windows *Notepad* programme, as before. The only significant difference is that, this time, there are two separate `[Ram]` 'banks', each configured to `16Mb`, totalling the default 32MB.

In this case, you need to change both `[Ram]` banks, equally *(taking care not to change the* `[Rom] range = 16Mb` *setting by mistake)*, up to a maximum of `64Mb` each, totalling 128MB. Re-save the file and, thereafter, the full 128MB RAM should be available (in the *-SA* version, for the emulation option you select).

```
[Rom]
filename = "Romsets\Riscos4"
base = 0
range = 16Mb  ← Do NOT change!

[Ram]
base = 0x10000000
size = 16Mb

[Ram]
base = 0x14000000
size = 16Mb
```

The procedure for the -DL variant is identical except that the maximum configurable RAM is 64MB.

### Setting 256MB RAM for RISC OS v4.02

***Health warnings!*** *Previous editions of VirtualRPC In Use included an unapproved, At Your Own Risk suggestion on how to configure VirtualRPC -SE and -SA to use 256MB RAM.*

*However, as some users subsequently ran into knock-on problems, this ruse has been formally deprecated and should* ***not*** *be attempted.*

*Similarly, if you decide to use the ARM7500 core emulation (perhaps to run a monitor in extra-high resolution modes as discussed on page 29), do not configure the RAM to more than 128MB as that was the limit for the 'real' A7000/+ machines and which is emulated in the ARM7500 core.*

*RAM settings in general are discussed in more detail in Support Article 158.*

### Sundry RAM considerations

In general, the system used by VirtualRPC to set the RAM proves to be quite robust. But if you do hit any memory problems, either in general or specific to a particular software title, double-check the amended settings in the relevant *Model* file to make sure there's no inadvertent error and, if not, try reducing the maximum configured RAM, from 256MB to, say, 128MB in the case of Virtual RPC-AdjustSA, or less than 128MB in the case of VirtualRPC-DL, -SE or SA.

But don't forget that, if you do make a total hash of it (in which case VirtualRPC may not start up correctly, if at all), you can easily recover by replacing the corrupted *Model* file with the fresh one from the *BackUp* folder generated earlier on (preferably using the copy & paste method, leaving the one in the *BackUp* folder in situ).

### Changing the hard disc drive name

By default, the VirtualRPC 'hard disc drive' (actually a standard Windows folder) is called *HardDisc4*, clearly taking the cue from the standard RISC OS/ADFS naming convention. Although in this case, it uses the VirtualRPC proprietary (and very fast) *HostFS* filing system. Nevertheless, most users will probably wish to change the name to something else and, of course, this is simplicity itself under *ADFS*.

Unfortunately, the simple renaming feature is not available to us under *HostFS* and, equally unfortunately, two factors have led to the procedures being perceived as unduly difficult, so the misperceptions need to be put to bed.

First, VirtualAcorn do not recommend renaming the hard disc – but it must be stressed that this is nothing to do with VirtualRPC as such.

Indeed, if you were to rename the hard disc on an ARM-powered machine (using *ADFS*, *IDEFS* or *SCSIFS*) you would hit almost exactly the same problems, typically `Disc not found` error messages. The reason for this is simply that any instance of a file leafname having an *absolute* reference to *HardDisc4* can no longer be 'found'; hence the error message(s).

*(This is explained in more detail in the article reproduced in the Knowledge Base section at www.virtualacorn.co.uk/support/articles/107/index.htm)*

So in the first place, do be aware that if you change the hard disc name, you'll have to hunt around for the culprit filename(s) – but this is no different to what would also happen on an ARM-powered machine. As far as we know, there's only one VirtualRPC-specific file with an absolute reference to *HardDisc4* – part of *ShareFS* – and we'll consider that shortly.

But if you're still intent on doing it, there are a couple of methods of achieving the object:

- **Renaming the disc under VirtualRPC** – There's no need to go into any detail on the first option as it's covered in the Knowledge Base support *Article 107*. Briefly the procedure is, first, to set up a new *HostFS* 'mount', mimicking the default *HardDisc4* mount in every respect, except for your choice of name. Then if you switch VirtualRPC to `Off` and re-start it (a simple `Restart` is not sufficient) the parallel hard disc drive icon will appear on the icon bar, next to *HardDisc4*, and with your chosen name.

As a precautionary check that all is OK so far, click on both hard disc drive icons in turn and check that the two directory viewer contents (*!Boot*, *Apps*, etc, etc) are identical in every way.

Now comes the teeth-gritting bit... Press ‹Alt-Enter› to drop VirtualRPC into a window, select Tools › Options › Hostfs, click on the default `HardDisc4` (under `Mount`), then click in turn on Delete (!), Apply and OK.

Finally, switch VirtualRPC to `Off`, re-start it and – fingers crossed – it should start up OK with just the *single* hard disc mount on the icon bar – and with your chosen name. But for the moment, don't be concerned if you get one or more of the `Disc not found` error messages described above.

**• Renaming the disc under Windows** – The second option isn't documented and, again, is an At Your Own Risk procedure, but we feel it's much simpler and, once more, even if you make a hash of it, you will not (physically) break anything and we feel it's arguably relatively easy to recover.

First, navigate down the Windows directory tree to the same *Model* file for your installation and load it into *Notepad*, as described on page 25.

Now look for the segment starting `[HostFS]`, a few sub-sections further down from `[Rom]` and `[Ram]`. This lists all the *HostFS* 'mounts' under VirtualRPC which, for the present, consists of just the default `HardDisc4`.

```
[HostFS]
podule = 0
mount0 = "HardDisc4 'HardDisc4' bootopt=2,showext"
```

Noting that the name `HardDisc4` appears twice in this line, it's the *first* instance (underlined) which represents the name which will appear under the hard disc icon on the RISC OS icon bar.

So you can now amend the first `HardDisc4` to whatever you choose, but taking care that you do *not* inadvertently delete the opening " double-quote character, and also avoiding characters which may trigger problems under Windows and/or RISC OS such as a normal space, hard space, an / oblique, and so on. So for this example, we'll call it `MyName`.

```
[HostFS]
podule = 0
mount0 = "MyName 'HardDisc4' bootopt=2,showext"
```

Re-save the *Model* file and, thereafter, when you run VirtualRPC, it should start up OK, with a single hard disc drive icon, but now called `MyName` (or whatever else you've chosen). Once again, do not be concerned if you get the `Disc not found` errors.

### Renaming the disc – troubleshooting

If, after changing the hard disc name, VirtualRPC doesn't start up correctly (setting aside the `Disc not found` errors for the moment), the chances are it will skip the *!Boot* file and display a bare, default RISC OS desktop in a low-resolution display.

In this case, you can drop VirtualRPC into a window, select Tools › Options › Hostfs and double-check all the settings for the (renamed) hard disc mount. A favourite error is to omit to tick the `Bootable` box, which would give these symptoms.

But if that doesn't give the answer, or if VirtualRPC doesn't start up as far as the default desktop, or if you renamed the disc under Windows, then in all cases we would recommend inspecting the `[HostFS]` sub-section of the *Model* file and double-check that it all reads correctly.

That should give you a readily-identifiable answer which can be corrected relatively easily but, if not, the problem (whatever it is) can always be removed by copying the default *Model* file from the *BackUp* folder you generated beforehand (page 25) and overwriting the suspect *Model* file.



### Clearing the errors

Having got VirtualRPC to start up fully, with the renamed hard disc drive, you can now clear the `Disc not found` error(s). One known source is that, in the *VirtualRPC-AdjustSA* variant, *ShareFS* (which is installed and enabled) includes a file with an absolute reference to the default disc drive name `HardDisc4`, so first you need to cater for that.

To load the file into *Edit* (or your active text editor), ‹Shift-double-click› on the main *!Boot* file under RISC OS to open the *Configuration* window and select Choices › Users › Single › Boot › Tasks › AccessSet. Note that the last line in the *AccessSet* file contains '`HardDisc4`'– hence the error message.

If you have no need to use *ShareFS*, REM out this last line and re-save the *AccessSet* file. If you will be using *ShareFS* however, and using the same example as before, amend the last line to read `Share HostFS::MyName.$.Public` and re-save the file. In both cases, this stops the error message appearing at VirtualRPC start-up.

But if you continue to see `Disc not found` error messages, then it's a case of searching through whatever you may have added to the default installation which might have an absolute reference to '`HardDisc4`'.

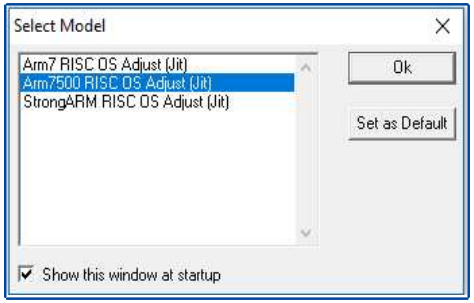
A good starting point is the *Desktop* file but it could be anywhere. We learnt that lesson early on when faced with numerous errors, all caused by putting absolute file leafnames in *NeXTBar*'s Task files. So the lesson here is that, if you do plan to re-name the default `HardDisc4`, do it at an early stage, preferably before adding any user-software etc to the installation, and check things as you proceed.

### Selecting the start-up emulation

At initial start-up, *VirtualRPC-SA* and *-AdjustSA* give you the three choices of start-up emulation discussed on page 13 and, where applicable, you'll probably wish to select the one you're most used to from RiscPC days, i.e. ARM7 or StrongARM. This can be set as the default and, if you wish, you can untick the Show this window at startup box to miss out this step in future.

Both options give 8MB available 'VRAM' so (if your monitor is up to it) you can display up to, say, 1600×1200 pixels in 16M colours.

But we mentioned the possibility of even higher display resolutions being available by selecting the ARM7500 start-up option, the video RAM limit then being (we think) 16MB, and allowing for 1920×1200 or even 2048×1536 pixels, both in 16M colours.




The problem here of course is that, if you've previously *un*ticked the Show this window at start-up box, how can you select the ARM7500 option?! No problem: a fairly simple tweak, via Windows, can restore it. In this case, navigate to the main VirtualRPC directory as before and, this time, double-click on the *VA* configuration file to load it into *NotePad*. At the top, look for line 2 saying `forceselect = 0`.

```
[Model]
forceselect = 0
Name = "StrongARM RISC OS Adjust (Jit)"
```

Manually amend this to read `forceselect = 1` and re-save the file, start up VirtualRPC and you'll see the Select Model window, allowing you to select the ARM7500 start-up option and click on Set as Default.



Return to Contents

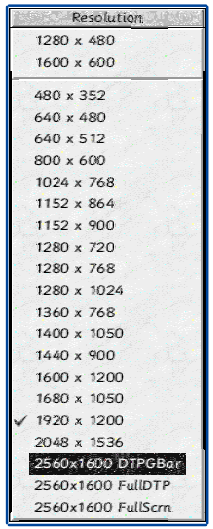
In future, you will have the much higher VRAM available for even bigger and better displays if desired, under the Arm7500 emulation.

Whether you again untick the `Show this window at startup` box depends if you happen to have any StrongARM-*only* software as, if you do, you'll have to start up in the `StrongARM` emulation to use it. So leave the box ticked. But if not (quite likely) it can again be unticked.

(Just a reminder that, if you do select the ARM7500 core option, you may also need to configure the RAM setting to a *maximum* of 128MB as explained on page 27.)

Interestingly, a gallant VirtualRPC user posted an *article on the Forum* which provided three additional monitor definitions to extend the possible display resolutions from the ARM7500 emulation up as far as an extraordinary 2560×1600 pixels!

This involved adding the extra definitions to the existing monitors file on your hard disc, located in *!Boot › Resources › Configure › Monitors › VirtualAcorn › VRPC-SE* (but remember to keep a safety back-up of the original *VRPC-SE* file).




Once modified, re-saved and VirtualRPC re-started, the 3 extra screen modes will be listed in Displays › Resolution on the icon bar.

We found the first two options (DTPGBar and FullDTP) will still run in 16M colours, and gobble up around 14000k Screen memory, but the FullScrn option drops the colour depth to 'only' 32K, so presumably would otherwise exceed the 16MB VRAM limit of the ARM7500 emulation.

Even better, VirtualAcorn have now incorporated a slightly modified version in Support *article 146* which provides a display resolution of 2560×1440 pixels (14400k screen memory). This is excellent as the definition file exactly matches the default '2.5K' display on many contemporary graphics cards and monitors.

This (so-called) 2.5K display in 16M colours is absolutely gorgeous – but it should be noted that all these super-high resolution displays do require a lot of welly from the underlying host PC processor. Nevertheless, Core2Duo processors and the current i3/i5/i7 series seem well up to it; however, the 'old' (by today's standards) Pentium 4 processors find it a struggle and screen redraw performance is noticeably affected.

But who would have thought that we can now run an emulation of the legendary RiscPC in display modes using up to *8*× the original, maximum of 2MB VRAM?! Lovely stuff.

(As an aside, it needs to be appreciated that, on '2.5K' monitors, or hi-res laptop/tablet displays, text is significantly smaller compared with, say, an HD monitor. There's some discussion on the net about this and the general agreement seems to be that a 2.5K monitor resolution display is about the practical limit for text readability. Indeed, Macs provide an option to increase the text size on their 4K and 5K displays to allow for this.

Although Windows 10 has an equivalent option to increase the screen scaling, in steps of 25% (right-click anywhere on the desktop and select Display settings), unfortunately this re-scales everything – text, objects, windows, etc – and will seriously mess up a VirtualRPC *full-screen* display, so cannot be recommended.)

### Extra hard disc drives/directories

As we'll see in the next section, RISC OS running under VirtualRPC can access and make use of all manner of storage media either installed in or plugged into the host PC, among them one or more additional hard disc drives.



Indeed, many desktop PCs now come with an SSD, plus a hard disc drive already installed and configured. If that isn't the case, physically fitting a second drive is a relatively easy task, but then it still needs to be initialised and configured.

The procedure does rather depend on the version of Windows, so isn't really something we can deal with in this booklet, but if you Google '`fitting a new disc drive`', that will offer you a number of helpful options to choose from.

Why fit a second hard disc drive? Apart from the obvious bonus of increasing the total storage capacity, the main advantage as we see it is to give you a convenient method of backing up the main disc drive, in case of data corruption or even failure of the primary drive.

The backup can be either a complete mirror image (including the Windows operating system), or selective areas such as the application software and data files, especially in relation to our activities under RISC OS, and we'll look at that in more detail in Section 13.

But what if there isn't room for a second disc drive, in a laptop/notebook/tablet or (depending what else is installed) in a 'shoebox' PC? In this case, as far as backing up data is concerned, we can achieve the same ends by having one or more directories on the main disc drive, set aside specifically for the backup purposes.

Note that it is *not* necessary to have a separate partition on the disc; one or more additional, standard Windows folders will do the job in providing a backup copy, although of course this does not insure against a total drive failure.

Looking specifically at using RISC OS under VirtualRPC, we will have the main *HardDisc4* 'drive' on the icon bar (renamed if you wish) but it will greatly help to have a second mount on the icon bar for ease of backing up RISC OS software/data, pointing to either a separate disc drive in, say, a desktop PC, or to a separate folder on the primary disc drive in a laptop, or to a portable, plug-in, USB hard disc drive.

More on that in the next section but, if you will wish to have a backup directory for RISC OS, on the primary hard disc drive of the host PC, then for simplicity's sake we would advocate having a second Windows folder, called *HardDisc5*, alongside the main *HardDisc4* folder (reiterating that, as far as Windows is concerned, these are simply standard folders, not special partitions set up on the disc drive).

This is easy to achieve: first, once again navigate to the *VirtualRPC-xxx* folder containing *HardDisc4* and all the other VirtualRPC goodies (page 25) and, once there, right-click anywhere in an empty area of the *VirtualRPC-xxx* directory, select *New* and *Folder*. The new folder appears and we can rename it to, say, *HardDisc5*. For the moment it will be empty but we'll soon add precious info to it.

If we have a second, physical hard disc drive installed, then it would be advisable to put a folder onto it for use as the backup medium. The leafname doesn't have to be the same as on the primary disc drive although, normally, we put *HardDisc5* in the root directory of the second disc Drive D:

So the Windows path where the backup folder can be located is *D:\HardDisc5*.

### Windows ↔ RISC OS exchange

In the final discussion in this section, we'll pave the way for the important topic of transferring objects and data between Windows and RISC OS, and vice versa.

For the moment we won't run though the typical reasons for doing this although, as the



Return to Contents

transfer method is quite simple, and involves setting up a 'common' folder in both Windows and RISC OS (i.e. a single folder on the hard disc drive which is readily accessible by both operating systems), it's convenient to deal with it under this section heading.

The whole thing relies on the factor discussed a few times before: i.e. all directories used under RISC OS are – as far as Windows is concerned – simply standard folders. So to start with, under RISC OS, we need to generate a standard directory somewhere convenient.

That said, it doesn't have to be in, say, the root directory of *HardDisc4*, as we shall provide handy short-cuts to it, under both RISC OS and Windows.

So for the sake of this illustration, we'll put it somewhere out of the way; say inside a directory called *Support*, on the root of *HardDisc4*.

And as we shall be using this 'common' folder to exchange things between Windows and RISC OS, *in both directions*, we'll name it (say) *Exchange*.

Now go across to Windows and seek out this *Exchange* folder, located within the RISC OS *HardDisc4* folder. To do that, navigate as before to the Windows folder containing all the VirtualRPC goodies (page 25).

Double-click on the *HardDisc4* folder – and this will reveal the main *!Boot* directory, the *Apps* folder and all the other directories on the 'drive', including the *Support* folder. Double-click on *Support* to reveal the newly-generated *Exchange* folder.

Next, we use an extremely useful feature of the Windows filer which is to generate a 'short-cut' to the *Exchange* folder, which you can then move to somewhere far more accessible – like the desktop.

To do that, *right*-click over the *Exchange* folder and select the Create Shortcut option. This will generate a new folder, called *Shortcut to Exchange*.

Finally drag-and-drop (i.e. move) this shortcut folder onto the Windows desktop and, if you wish, rename it to simply *Exchange*.

Note that this *Exchange* folder icon will have a small arrow in the bottom-left corner, denoting that it is only a *shortcut* to the *Exchange* folder located in the RISC OS *HardDisc4.$.Support* directory, and *not* the *Exchange* folder itself.

(Indeed, with the exception of the *Recycle Bin* and one or two others, you may well find that all the other icons on your Windows desktop are in fact shortcuts to Windows applications, perhaps located well down the tree.)

Now, whenever you double-click on the Windows *Exchange* shortcut icon, it will reveal whatever objects (applications, datafiles, graphic files, compressed directories, etc) are contained within the *Exchange* folder in the RISC OS *Support* directory.

So when you wish to send anything from Windows to RISC OS, all you need do is to copy (or often move) the object(s) into this *Exchange* folder. Indeed, as we saw in the Copy to Folder discussion on page 24, you don't even need to open the *Exchange* shortcut folder; you can simple drag and drop the object(s) over it.

The corollary is that, whenever you wish to send anything from RISC OS to Windows, you just need to put the object(s) into the 'parent' *Exchange* folder located in the VirtualRPC *HardDisc4.$.Support* directory. But we do need to make it more conveniently accessible under RISC OS and, in the next section, we'll consider various ways of doing that with one or more shortcuts to it.



# 6. Setting up RISC OS

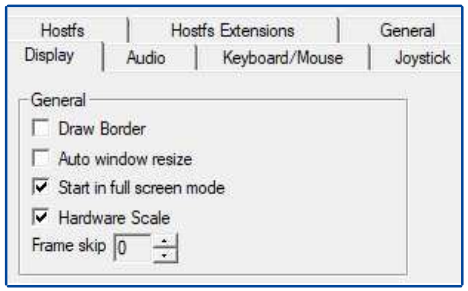
This main topic follows on immediately from the last; i.e. having made various configuration settings, using a combination of Windows and VirtualRPC, we can now finish off primarily by using RISC OS, but with a little further help from VirtualRPC.

A large part of this is already set out in detail either in the RISC OS 4 manual, the VirtualRPC User Guide and/or the *Knowledge Base articles* in the VirtualRPC website Support pages, so we'll concentrate on giving an overview of various topics and adding a little detail to the document-ation. We'll also offer a few thoughts of our own where they've helped us (and others) in the past.

## The default RISC OS display

When you first run up VirtualRPC (any variant), it will present you with a default RISC OS display, in a VirtualRPC window, in a relatively low resolution and with various goodies pinned to the desktop. So before you start adding user-applications and datafiles, you'll probably wish to make display changes to suit you. The order isn't important.

The first might be to arrange for VirtualRPC to start up with RISC OS running in full-screen mode, rather than in a window, and this is fixed simply by clicking on Tools › Options, selecting the Display tab, ticking the Start in full screen mode box and click on Apply and OK.




In future, when you start or re-start VirtualRPC, RISC OS will be displayed in full-screen mode.

Unless you're using a small-screen laptop or tablet, the chances are your monitor can support a much higher resolution than the default 800×600 pixels, so the display can be re-configured in the usual way via the !Boot › Configuration facility (slightly different between RISC OS v4.02 and v4.39).

If you're coming straight from, say, a real RiscPC (without a Viewfinder or Vpod card), expect to be impressed by the range of high-resolution display modes you can choose from. And (if you've installed the additional MDF), including the '2.5K' added-extra. This is all courtesy of the default 8MB 'VRAM' provided by the ARM7 and StrongARM emulations, and double that from the ARM7500 core, so do fill your boots.

The main limitation is the 'oomph' of the host PC processor available to VirtualRPC. So if you should happen to detect any evident hesitation in RISC OS screen redraw performance, try dropping the display resolution and/or the colour depth, by degrees, to find a combination which best suits your preference.

## Drives, devices and all that

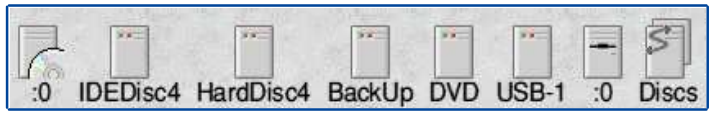
By default, VirtualRPC seemingly replicates its real counterpart and puts icons for a CD drive, a hard disc drive and a floppy disc drive on the icon bar at start-up.

However, these are of course mainly software emulations and differ from the real hardware in various ways which we'll discuss, plus there's a Discs icon present on the icon bar, and which may be new to you.

Just one major advantage of VirtualRPC, compared with the real thing, is that we can so very easily add to, subtract from (i.e. 'remove') and generally re-arrange and configure all manner of existing and additional devices which are either installed in or can be plugged into the host PC.



Return to Contents




All this is without the need to install special RISC OS interface cards and/or proprietary software – and all at zero expense. It's an extraordinarily flexible system which, depending on your needs and wishes, can hugely enhance usability and productivity. So on that basis, it may help to look at a typical range of these individual 'devices' (above) in some detail, under their own topic headings.

### Default CD Drive (CDFS)

Provided the physical DVD/CD optical drive installed in the host PC is an *IDEFS* device, the single CD drive :0 icon is positioned in its usual place at the left end of the icon bar. (Accessing *ATAPI* optical drives as discussed on page 21 will be explained later.)

The *IDEFS* DVD/CD drive is accessed via *CDFS* so can be used to play audio CDs and read data CDs as normal, or write to them using, say, *CDBurn*. Under VirtualRPC-AdjustSA (RISC OS v4.39), data DVDs can also be accessed via this drive.

Reportedly, modern DVD burners are increasingly unlikely to run even standard audio CDs in ARM-powered machines. Fortunately this is rarely the case under VirtualRPC but, if in the unlikely event you were to hit a problem with an optical drive under CDFS, there are always ways of getting round it.

This may vary from simply playing an audio CD under Windows (while continuing to use VirtualRPC in all other respects) to setting up a HostFS 'mount' for a drive which is otherwise inaccessible under CDFS. But we stress this is comparatively rare.

Furthermore – and with no wish to underestimate the excellent *CDBurn* – we would tend to suggest that all DVD and CD burning is done under Windows, for simplicity, reliability, speed and other reasons, and we'll discuss this in Section 13.

### Default 'HardDisc4'

The two main differences between the main disc drive in an ARM-powered machine and the equivalent *HardDisc4* running under VirtualRPC are that the latter is simply a standard Windows folder on the host PC's main disc drive and that it uses VirtualRPC's proprietary filing system called '*HostFS*'.

Although this unavoidably introduces some additional considerations (e.g. renaming the drive and retaining RISC OS filteypes), these are soon overridden by the extraordinarily fast read/write speeds routinely achieved, especially if the host PC drive is a modern, SATA3 variant or, better still, an SSD (see the test data recorded and discussed on pages 6-7).

### Default floppy disc drive

The RISC OS floppy disc driver provided on the VirtualRPC installation CD should be used only if your host PC has an *internal* drive fitted, in which case install it and use the floppy disc drive on the RISC OS icon bar exactly as you would in an ARM-powered machine.

If there is no internal drive, the icon becomes superfluous but can be removed in the normal way by double-clicking on *!Boot* and accessing Discs › ADFS discs, setting the number of Floppy discs to zero, and clicking on Set.

We'll discuss how to set up a 'mount' for a plug-in (USB) floppy disc drive shortly but, if you do, it must be appreciated that this option will *not* read ADFS-formatted discs. So if, for example, you have lots of RISC OS application software on the original master floppy discs, you'll have to revert to other methods to get the software onto *HardDisc4* (discussed in Section 7).



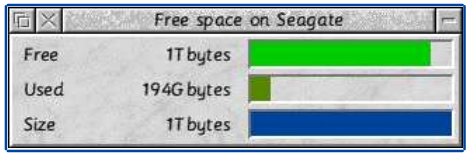
But if you're absolutely determined to have an internal floppy disc drive, then talk to R-Comp as they may still be able to supply a machine to suit your purposes.

### Adding further HostFS 'devices'

We cannot overstate the ease with which all manner of additional devices and storage media can be set up and accessed under RISC OS (running under VirtualRPC).

These can include an extra hard disc drive (or an additional folder on a primary drive), optical drives and all manner of plug-in USB media (compact flash, SD, etc).

In essence, if the device works with the host PC, then there's a very high probability that it will work with RISC OS, under VirtualRPC, without any restriction on capacity, etc (e.g. the 1TB plug-in USB backup drive illustrated).




In principle, all we need to do is to add an extra HostFS 'mount' to access each device. For a while it was necessary to do this by making manual changes to VirtualRPC's Tools › Options › Hostfs facility but, subsequently, an application called *DrivePop* (the 'HostFS pop-up drive manager') was released as part of the VirtualRPC-SA and -AdjustSA variants, very neatly making the whole thing semi-automatic.

We've since found this additional item greatly enhances the flexibility of the overall system, depending on how you treat additional devices in day-to-day usage. For example, if you have a fairly 'static' arrangement with, perhaps, an extra mount for a second hard drive (or folder) and occasionally plugging in a single, USB device, arguably it's just as easy to configure these manually, without running *DrivePoP*.

Alternatively, if you're forever plugging in and unplugging extra storage media, then *DrivePop* simply reduces that to a single mouse click. Or you can use a combination of both procedures. entirely to suit you.

For this discussion we'll assume it's to be done manually. Something to be aware of is that, when adding mounts for further devices, they will appear on the RISC OS icon bar in the order in which you add them, working from left to right, so a bit of forward planning will help.

Although each requirement is covered in detail in the user-guide, we'll look briefly at how it's done, manually, in every case first accessing Tools › Options › Hostfs.

### Hard disc/folder mounts

In this instance (and using the example discussed on pages 30-31), you can either type the full pathname D:\VirtualAcorn\HardDisc5 into the Path: window, or you can click on the '...' (browse) icon and locate the new folder that way.

Now enter the desired name into the Mount: window (calling this additional drive, say, BackUp), leave all three boxes *UN*ticked and click in turn on Add, Apply and OK.

Switch off VirtualRPC, re-start it – and a new mount called BackUp will appear on the icon bar, immediately to the right of the HardDisc4 icon, and giving direct access to the (presently empty) *HardDisc5* directory, in this case on drive *D:* in the host PC.

Why '*BackUp*'? We have this mount on all our VirtualRPC installations. On the desktop machines, it points to the *HardDisc5* folder on the second physical hard disc drive; on the laptop, it points to the extra folder alongside *HardDisc4* on the single hard disc drive.

In every case, we maintain a mirror-image of the complete *HardDrive4* drive on this *Backup*



Return to Contents

drive, including *!Boot*, all directories, application software, utilities, all our datafiles etc. On the desktop machines, this protects against the effects of drive *C:* failure; and on the laptop, it insures against the primary file(s) becoming corrupted. There are also other convenient uses which we'll discuss in Section 7.

### Optical drive mount(s)

You may wish to add a *HostFS* mount for one or more optical drives, accessing DVD/CD read/writer(s) in the host PC. Reasons for this could be:

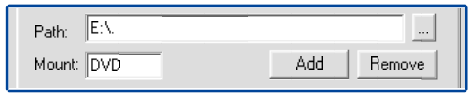
• Although unlikely, an *IDEFS* DVD/CD drive in the host PC might not work under RISC OS via *CDFS*, but we can access it via a *HostFS* mount instead. If this is the case, the superfluous *CD:0* icon can be removed via the !Boot › Configuration › Discs › CD-ROMs facility.

• If the DVD/CD optical drive in the PC is an *ATAPI* device, it will have to be accessed by a *HostFS* mount.

• You have VirtualRPC-SE or -SA (RISC OS v4.02) and also wish to be able to read data DVDs.

• You have a second optical drive in the host PC which you also wish to access.

In all these cases, an extra mount can be provided for an optical drive. Assuming it comes up as drive E: in the host PC, type E:\. into the Path: box (*note the important full stop following the backslash*), put a suitable name in the Mount: box, and click on Add, Apply and OK.




Close down and re-start VirtualRPC, and the DVD 'drive' will appear on the icon bar, to the right of the HardDisc4/5 mount(s). This gives access to the DVD/CD drive (NB that the icon will be as for a standard RISC OS v4 hard disc drive).

Although VirtualRPC supports only one optical drive under *CDFS*, you can have separate *HostFS* mounts for two physical drives in the host PC, if fitted, simply by adding a second path (e.g. F:\.) and a suitable name. But the two mounts should be added individually, closing down and re-starting VirtualRPC each time.

Note that, if accessing a 'RISC OS' CD or DVD via a *HostFS* mount, such as an Archive mag or Foundation/RISCWorld disc, the front-end utility (e.g. *RWold*) will not work. So on the face of it, you can't access the disc contents! However, if instead you first double-click on the *HTML* directory, and then on the *INDEX/HTM* file, that will open the home page on the disc – and away you go.

If there's no disc in an *IDEFS* optical drive and you click on its icon, under *CDFS* you will get a CD-ROM drive empty message. On a *HostFS* mount, you will get an error from the *HostFS* filer saying Error when reading HostFS::DVD.$ – Not Found. This is normal.

### Plug-in storage media

In a similar manner, you can add *HostFS* mounts for one or more plug-in devices, including for example a portable hard disc drive, a memory stick, or a CF or SD card. The interfaces can cover USB devices and/or a plug-in or internal card reader.

If you're making the configuration manually (rather than using *DrivePop*), it's first necessary to determine the 'drive' letter(s) allocated under Windows as shown in the *My Computer* display and use the appropriate letter (e.g. G:\.) for each *HostFS* mount.

A slight complication here is that multi-socket card readers invariably show up either as (typically) four devices – e.g. 'drives' *G-J* inclusive – even though no media are physically inserted. Or alternatively, they don't show up in the *My Computer* display until a medium is actually inserted, when it will then be triggered under 'plug & play'.

So some checking under Windows is first necessary to sort out what drive letter(s) to put in each *HostFS* Path: window.

Other than that, manually setting up the mount(s) is as before and, although RISC OS v4-style 'hard disc drive' icons will appear on the icon bar, clicking on them will access the plugged-in storage media.

Once again, if nothing is plugged into the socket relating to the relevant 'drive' letter, an error message such as Error when reading HostFS::USB-1.$ – Not Found is generated.

### Plug-in floppy disc drive

Setting up a mount for a plug-in (USB) floppy disc drive is relatively simple as, in all probability, it will appear as drive *A:* in the *My Computer* display of the host computer, so typing A:\. into the Path: window of the HostFS mount facility will do the job.

Note that, being a HostFS mount, once again it will appear with a standard RISC OS v4 hard disc drive icon on the icon bar and *not* the relevant floppy disc icon; for this reason we prefer to name the drive 'Floppy' to avoid confusion.

If the mount is added after those for the optical drive/storage media previously discussed, the icon will appear at the right end of the *HostFS* mounts (where the original *ADFS* floppy drive icon is in the figure on page 34.

But if preferred, it can be put (say) immediately to the right of the BackUp hard drive instead, simply by re-arranging the order in which the various mounts are first entered.

Noting that the plug-in USB floppy disc drive cannot read *ADFS*-formatted discs, this of course includes many early RISC OS application software titles supplied on floppy disc(s). However, a simple alternative method of moving files across to VirtualRPC (if you still have access to an ARM-powered machine) is to re-save each application – preferably in a compressed folder – from the hard disc back onto DOS-formatted disc(s).

The *HostFS* floppy disc drive mount will then be able to read the DOS-formatted discs and the files can be uncompressed and transferred onto VirtualRPC's *HardDisc4*. (Compressing the files doubly ensures the RISC OS filetypes will be retained throughout the transfer.)

### 'IDE' hard disc

We'll discuss RISC OS software compatibility in Section 8 but, for the moment, it's safe to say that almost everything you're likely to have, from the earliest days of RISC OS, right up to the very latest releases, will either work as-is, or can be made to run satisfactorily under RISC OS 4.02 (i.e. VirtualRPC-DL, -SE or -SA). A very few utilities may not run under RISC OS 4.39/Adjust under VirtualRPC (or ARM-power).

But there is a knock-on effect with some titles where, for anti-piracy or other reasons, either they will run only from *IDEFS* or they write data back to the original (floppy) disc, and this may not be possible under VirtualRPC.

However, VirtualRPC incorporates a built-in working alternative in providing an optional *IDEFS* 'disc' which appears on the icon bar by default (see the figure on page 34) and clicking on that reveals various files including *ReadME*.

Note that this 'IDE' hard disc (uniquely, a special partition within VirtualRPC) is intended only as a repository for application software tiles which *have* to run under *IDEFS* (*Sibelius* being a rare example) and is *not* intended as an alternative to *HostFS* whose general performance is very many times faster.

So by default, the IDE 'drive' is 50MB in size which should be more than adequate for your

purposes. But if in the unlikely event that you do need more space, the partition size can be increased by reconfiguring a VirtualRPC file. Being a very rare occurrence, we won't cover the procedure here but, if you need the details, please contact us or VirtualAcorn.

If you don't require this 'IDE' drive, the icon can be removed by accessing the !Boot › Configuration › Discs › ADFS Discs facility and setting the number of Hard discs to zero.

### Drag-and-drop saving

A slightly unfortunate feature of *HostFS* needs to be mentioned here. If you drag-and-drop an object over a *HostFS* storage device icon (e.g. HardDisc4), it will *not* be saved on the drive. However, if you drag-and-drop it over the IDEDisc4 icon, or over a RAM disc, the object *will* be saved, as normal.

### Share filing system

The Discs icon shown in the figure on page 34, at the right end of the various mounts, gives access to the *Share* filing system, enabling networking with the outside world which is fully explained in the user-guide.

By default, *ShareFS* is enabled when you fire up VirtualRPC-AdjustSA and we've already dealt with the 'Not found' error message you'll get if you should decide to rename the main HardDisc4.

But what if you don't have a need to network your host PC? Can the now-superfluous Discs icon be removed? The answer is "Yes" and the procedure is covered in the user-guide – but it has never worked for us!

It seems that however many times we *UN*tick the Enable ShareFS box in !Boot › Configuration › Network › ShareFS, someone or something keeps *RE*-ticking the box and the pesky Discs icon keeps reappearing every time we restart VirtualRPC...

The reason for this is there's a file within *!Boot* which reinitialises *ShareFS* at RISC OS start-up. So although not a documented procedure, should you wish to disable *ShareFS* altogether, here's our suggested workaround:

• Open the *ShareFS* obey file in the *!Boot › Choices › Default › PreDesk* folder and disable the re-initialise command by preceding it with a '|' character so that the command now reads `|rmreinit ShareFS` and re-save the obey file. This stops *ShareFS* being reinitialised at startup.

• If not already done (to stop the `Not found` error message if you re-name HardDisc4), open the *AccessSet* obey file in the *!Boot › Choices › Users › Single › Boot › Tasks* folder and disable the final line with a preceding '|' character so that it reads: `|Share HostFS:: HardDisc4.$.Public` and resave the obey file. This will suppress a `File 'Share' not found` error message at start-up.

• Access the !Boot › Configuration › Network › ShareFS option, untick the Enable ShareFS box and click on Set.

• Press F12 and type `unplug sharefs` ‹Enter›.

Phew. That lot should stop *ShareFS* running at start-up and stop the Discs icon appearing. *ShareFS* can be re-enabled if desired by reversing the procedure, the opposite of the `unplug sharefs` command being `rmreinit sharefs` ‹Enter›.

### RISC OS ↔ Windows exchange

On pages 31-32, we laid the foundation for a method of easy transfer of objects of all types between Windows and RISC OS, in either direction. We generated a standard folder on *HardDisc4* called *Exchange* which would conveniently be accessible by both operating systems. The location of this folder wasn't important so, for the sake of illustrating the principle, we put it well down the RISC OS directory tree in, say, the *Support* folder.

Under Windows, we then located the *Exchange* folder and generated a shortcut to it which, for convenience, we moved onto the Windows desktop amd renamed it as *Exchange*. Thereafter, any object(s) saved into it – or simply dragged over the *Exchange* shortcut – would be accessible under RISC OS.

To complete the arrangement, all we need to do now is to provide a similarly-convenient shortcut to the *same* (Windows) folder, but this time via RISC OS.

In practice, there are several alternative ways to achieve this, so take your pick from the following suggested flexible options:

• **'Tiny directory' on the icon bar** – You can arrange for a shortcut to the *Exchange* folder to appear on the RISC OS icon bar at start-up by adding a line to your *Desktop* file saying (for this example) `AddTinyDir HostFS::HardDisc4.$.Support.Exchange`

At RISC OS start-up, this will append the *Exchange* directory shortcut onto the icon bar, immediately to the right of your printer icon. Double-clicking on it will then open the *Exchange* directory viewer.

Two small niggles with the tiny directory idea are that we have to double-click on it to view the *Exchange* folder contents and, unlike Windows, RISC OS doesn't support drag-and-drop over the directory icon to save objects.

• **Pinned to the desktop** – An alternative is to pin the (open) directory to the RISC OS desktop. Then you can drag-and-drop objects into the directory viewer and it will automatically re-size to the right and downwards to suit.

The relevant command can be issued in several ways: a) at RISC OS start-up (this can be achieved by adding a line to your *Desktop* file saying `Filer_OpenDir HostFS::HardDisc4.$.Support.Exchange X Y` where `X` and `Y` are optionally the desired coordinates of the top left-hand corner of the *Exchange* directory); b) by an Obey file in *Tasks*; c) by an Obey file pinned to the desktop or, say, added to *NeXTBar*; d) or by a hot-key combo using *Keystroke*. Fill your boots.

• **TempDir on the icon bar** – If you like *Tempdir*, a suggestion is to arrange to have it fire up at start-up, and for the daily sub-directories to be stored in the *Exchange* directory.

This is done by tweaking the address option in the *MySettings* file within the *TempDir* application to read (for this example) `Setmacro TempDir$ HostFS::HardDisc4.$. Support.Exchange` (changing `HardDisc4` to whatever other name you've given it).

Just like the Windows filer feature, *TempDir* does support drag-and-dropping object(s) over its icon on the icon bar, which will then be saved within the 'today' sub-directory. A very neat option.

As always, clicking on the *TempDir* icon will reveal the 'today' sub-directory contents, while clicking ‹Adjust› over it will open all the daily sub-directories.

This works extremely well under RISC OS but, if there is a small downside, it means we can no longer drag-and-drop objects(s) over the *Exchange* icon on the Windows desktop as they will be saved in the *TempDir* 'root', and not the 'today' sub-directory.

• **NeXTBar shortcut to Exchange** – So the final suggestion, and the one we both prefer. where we routinely use *NextBar* * (v1.23 or later), is to drag-and-drop the main *Exchange* directory into an empty 'pigeon hole' in the *NeXTBar* display and, when the button options window opens, tick the Auto save in directory › Enable box, name the button Exchange and, finally, click on Update.

Now whenever you double-click on this Exchange shortcut 'button', the contents of the main *Support › Exchange* directory will be revealed in the directory viewer.

This RISC OS shortcut facility using *NeXTBar* is *exactly* like the one we set up on the Windows desktop, i.e. it also supports drag-and-drop of object(s) over it, which will then be saved into the main Exchange folder, and with an overwrite warning where relevant.

To summarise, whichever option you select, objects of any type can very easily be exchanged between Windows and RISC OS, and vice versa, conveniently via their respective desktops. If preferred, ticking the Create dated sub-dir box in *NeXTBar* will generate daily sub-directories in the same manner as *TempDir*, but with the same limitation at the Windows end.

** Following the sad death of the author, Paul Vigay, arrangements were made so that, although Paul's website is still active and is accessible via the link on page 39, orders for NextBar – which is shareware – are handled by Digital Phemomena Ltd.*



# 7. Transferring files to VirtualRPC

If just the main VirtualRPC application is installed (excluding the optional and very extensive public domain, shareware, etc, collections as listed in the user-guide), you will still find a lot of extra goodies on *HardDisc4* when you first fire up VirtualRPC.

Some of these will be required to make VirtualRPC work (mainly configuration files within *!Boot*); some will be specific to VirtualRPC but are not essential (e.g. the *DrivePoP* application); the many others provide a basic range of applications and utilities to enable someone coming completely cold to RISC OS to make a good start on getting used to our favourite operating system. These range from *ArcFS* through to *ZipEE* and, in VirtualRPC-AdjustSA, will include RISC OS v4.39 versions of *Draw* and *Paint*.

So if you set about transferring all your usual application software and datafiles onto HardDisc4, there is a bit of a danger of inadvertently overwriting something important – maybe with an older version (even if Newer is selected) – and this needs to be avoided.

For these reasons, whenever loading up *HardDisc4* in a new installation, we always aim to 'clear the decks' as far as we can before starting. For us, this entails moving most of the default goodies installation into a new, temporary directory (funnily enough, called *Temp*) but leaving the all-important *!Boot* file, plus the *Apps* and *Printing* directories in place on the root directory. (The latter because it contains the latest versions of *Printers* and printer definition files, plus a very useful file called *ShowPrinters*; more on that anon.)

That done, we can now set about installing or copying all software and datafiles onto *HardDisc4* and it may help if we run through the various alternatives.

• **Software on master CD** – There's no doubt that, if you already have a title on master CD, this is by far the most reliable of installation options. Usually there will be a front-end installer which will semi-automate the process for you and, most importantly, ensure essential resources such as modules are put in their proper places.

We take the firm view that, if you haven't got a particular title on master CD, then it's very well worth checking to see if one is available, either direct from the authors or via *PlingStore*. That's not only for the ease and reliability of installation but also because it's often a good opportunity of getting a later version of what you already have – preferably the latest. This can also be important where an earlier version can throw up problems (discussed in Section 8) or may not run at all under VirtualRPC.

Plus of course, if you intend to retain and use stuff on a previous machine, getting a second copy is very likely required by the user-licence.

• **Software for download** – Apart from major application titles, some commercial and other software, and most upgrades, can very conveniently be downloaded via the *PlingStore* facility or direct from the software house. This is a very flexible arrangement as, for example, files can be downloaded via a Windows browser if preferred and passed straight to RISC OS via the *Exchange* directory we set up earlier (pages 31-32 & 38-40).

Invariably, RISC OS software downloads will be contained in a compressed folder, most being readable by *SparkPlug* which is included in the VirtualRPC build. This is doubly-convenient as it ensures that RISC OS filetypes are properly retained during the download and transfer (to be discussed).



Return to Contents

• **Software on master floppy discs** – You may have a collection of software on master floppy discs – formatted to *ADFS* of course. If in the unlikely event that you have an internal floppy disc drive, their contents can be transferred onto hard disc as normal. But if in the far more likely event that you have to resort to using a plug-in drive, with a USB 'mount' on the icon bar, the *ADFS*-formatted discs cannot be read.

The easiest way round this is to copy the files from an existing machine onto *DOS*-formatted discs – in a compressed folder – which can then be opened and read via a plug-in drive as normal. The 1.44MB limit per disc isn't likely to be a restriction. But this assumes you do have the other machine and in any event, the more titles you have, the less practical this option becomes.

Note that if you purchase RISC OS software titles on floppy disc from *CJE Micros* (a.k.a. He Who Has it In Stock), in all probability the disc will be formatted to the *ADFS* standard, so you may well be back to the problem of how to transfer the disc contents to your PC.

• **Machine-to-machine transfer** – Provided your previous machine is still available, arguably the easiest method of transferring all your software and datafiles is to set up a local network between them.

This is described in detail in the VirtualRPC user-guide and utilises the Discs icon on the icon bar in conjunction with *ShareFS*, both of which are active by default.

But if the two machines aren't tied together, an intermediate method is to put your files onto a transportable medium such as a plug-in hard disc drive or flash card (assuming the source machine supports USB and you have a card reader for the latter) or, alternatively, by burning a data CD. With the CD option, a bonus is that the disc(s) then become a convenient back-up device for your files.

In all cases we do recommend that, in addition to putting your application software and datafiles on one or more media, you also include a copy of *!Boot* from the source machine.

***Health warning!*** *The reason for that will be discussed later but – whatever you do – don't even think of copying it in its entirety onto HardDisc4 in your VirtualRPC installation. If you inadvertently overwrite the default copy of !Boot, you will almost certainly screw up the operation of VirtualRPC, very likely requiring a re-installation from the master CD.*

### Compressed folder(s)

In all cases, it is recommended that you put the files to be transferred inside compressed folders. This is primarily to ensure the RISC OS filetypes are retained during the transfer but of course it also means you can pack more goodies onto each medium. Indeed, you might well get everything onto one CD.

Having said that, putting everything into a single, compressed folder is not recommended for various reasons; not least being that, if the compressed folder becomes unreadable for any reason, you lose the lot!

So use separate compressed folders, ideally for individual software titles or, perhaps, for a group of utilities or datafiles. Although initially this does require rather more effort, it really does pay dividends in the longer term.

### Retaining RISC OS filetypes

We keep harping on about RISC OS filetypes, so what's that all about? In essence, although Windows has its own, three-character 'extension' for each (Windows) filetype, it knows nothing about the RISC OS filetype system as such. That said, some of the identifiers for the two systems happen to be identical – *jpg* for JPEG images for instance – but generally this isn't the case.



So for example, if we pass a RISC OS *Fireworkz* spreadsheet called *Sheet1* from Windows to RISC OS (say, via the *Exchange* directory) it will arrive not only with the filetype *text*, but also renamed *Sheet1/fwk*, and double-clicking on it will produce pages of scribble in (say) *Edit*. So that's no good.

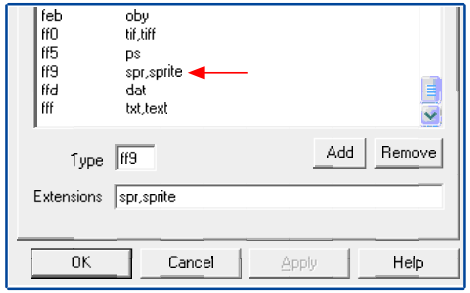
We've already talked about putting objects inside compressed folders and this is a foolproof method of ensuring that RISC OS filetypes are retained. However, that's hardly convenient for routinely swapping single objects between Windows and RISC OS, so VirtualRPC gives us a working method to cater for that.

### The HostFS extensions file

The system is based on *LanMan98* and is fully explained in the user-guide but, briefly, it consists of an *Extensions* file with all the three-character, alphanumeric RISC OS extensions listed against their better-known equivalents (i.e. the ones you see if you press ‹Menu› over a file and select File '(filename)' › Set type. This ensures that, when a file arrives in RISC OS, the appropriate filetype is attached to it.

But for that to work, the filetypes must be contained in the *Extensions* file and the one supplied with VirtualRPC has been added to over the years to cover the majority of those we're likely to use. But if you have something which isn't recognised, fortunately there's a couple of ways rounds it:

• **Add the data under VirtualRPC** – This method is illustrated in the user-guide. As it happens, the common or garden RISC OS *sprite* format had been omitted from the *Extensions* file which came with the original issue of VirtualRPC-SE, so all we needed to do was to put VirtualRPC into a window, access the Tools › Options › HostFS Extensions tab, add the three-digit alphanumeric extension in the Type box (`ff9` in this instance), add the various alternatives for sprites into the Extensions box (`spr,sprite` in this instance – noting there's no space after the comma), and click on Add, Apply and OK. Without having to restart VirtualRPC, a new line `ff9 ... spr,sprite` appeared in the HostFS Extensions listing, in its correct alphanumeric position.




Thereafter, sending a sprite from Windows to RISC OS 'in clear' (i.e. not in a compressed folder) ensured it arrived with the correct file icon and sprite filetype.

• **Add the data under Windows** – An alternative method which some people prefer is to make the changes via Windows. If you quit VirtualRPC, then navigate to the main *VirtualRPC–xxx* folder and double-click on *Plugins › HostFS2*, this shows the Windows *extensions* file. Double-click on that to load it into *NotePad* and it reveals itself as a simple text file. Type in the extra line `ff9 : spr,sprite` in place and re-save the file.

```
faf : htm,html,htl
fb1 : wav
fd4 : mid
feb : oby
ff0 : tif,tiff
ff5 : ps
ff9 : spr,sprite
ffd : dat
fff : txt,text
```

*Hint! If setting up an additional installation, copy your* existing *version of the extensions file (already containing the additional extensions for your software collection)* before *installing VirtualRPC. This ensures that VirtualRPC will 'see' the personalised extensions and will respond accordingly.*

**Using Typefind** – The first two methods have permanent effects but, if you find a



Return to Contents

little-used file arrives in RISC OS with the wrong filetype and it seems hardly worthwhile making a permanent fix, a utility called *Typefind* which is supplied with VirtualRPC will do a one-off job for you.

Simply load it, drag the offending file(s) over its icon, check the findings in the resultant window and, if they make sense, click on Set Filetype.

This saves having to make the change(s) manually and, with multiple files of different filetypes, can be a time-saver. That said, we find it's a little hit-and-miss on occasions.

### RISC OS filetypes listed

The two permanent methods work fine – provided you know the three-character, alphanumeric code to insert in either the HostFS Extensions › Type box (VirtualRPC) or the left-hand column of the *extensions* file (Windows). If not, the chances are it may be included in the extensive listings on: *http://www.riscos.com/support/users/userguide3/book3b/book3_14.html*

There's a seeming anomaly in the *Extensions* file which may raise an eyebrow: the odd-looking line saying `aff ... aff`. In fact this is intentional. The `aff` filetype is correct for the *RISC OS Draw* application and the entry is required to distinguish it from the *Windows* Draw extension.

### Generating compressed folders

Provided you update the *Extensions* file with new filetypes as they become apparent, and/or use compressed folders to transfer blocks of files from Windows to RISC OS, you shouldn't suffer any hiccups due to the loss of RISC OS filetypes.

We recommend using *SparkFS* for generating compressed folders as its Zip option is cross-compatible with the zip format used by Windows. But what if you don't have *SparkFS*?

In that case, *Zipee* (supplied with VirtualRPC) is an alternative, although in practice we've had a few problems with it, especially when compressing large files.

Arguably a better option is to use the built-in Windows feature as, once compressed folders are generated, they can be passed to RISC OS and used as an alternative to those generated by *SparkFS* or *Zipee*, seemingly without any compatibility or reliability problems.

To do this, go to Windows, *right*-click over any unused part of the desktop and select New › Compressed (zipped) Folder (the last option in the sub-menu). This generates an empty, compressed folder on the Windows desktop called *New Compressed (zipped) Folder* – which you'll probably want to rename to something which is less of a mouthful...

However, if you now drag this compressed folder over the *Exchange* shortcut icon (pages 31-32), it will be saved inside. And if you nip across to RISC OS, you'll find it in the *Exchange* shortcut folder (pages 38-40). The compressed folder can now be used just as if it had been generated in the first place by *SparkFS* or *ZipEE*.

Should you need multiple compressed folders, they can still be generated as required under Windows as, automatically, each filename will be given the unambiguous suffix *(2)*, *(3)* and so on.

So if, for example, you plan to move large numbers of applications and datafiles from a previous RiscPC to your shiny new VirtualRPC installation, we'd then suggest generating multiple compressed folders under Windows in the host PC, put them on the transportable medium (e.g. a flash card), take it to the source machine, put all the source objects into the folders, take it back to the host PC and access the folders under (say) the USB 'mount' under RISC OS. This will ensure the RISC OS filetypes are retained throughout.



# 8. Reliability & compatibility

In this section, we'll have a look at some potential problems which you might possibly experience when using the underlying PC hardware and your collection of RISC OS software, both under VirtualRPC. In the latter case, we'll also look at various ways to get particular titles to work.

### ARM-powered hardware reliability

In general, the RiscPC built up a very good name for reliability and we got the impression that it was a well-built machine. But we found that, in practice, once you started packing it with multiple, additional devices such as the PC co-processor, extra drive(s) and podules, that's when the problems tended to arise.

Added to that, we always got the gut-feeling that having to strip down the machines either partially or often fully in order to install almost any hardware upgrade, must have had at least some adverse effects on long-term reliability.

We had an Iyonix for a while and that seemed to be more reliable, possibly because upgrading the innards was a much more straightforward excercise, so it didn't need a strip-down.

We have no experience of more modern ARM-powered hardware but several user-reports are very complimentary about their reliability, performance and and so on. So that's all extremely encouraging for Fans of ARM-power.

### PC hardware reliability

The knockers would have us believe that PCs are very unreliable animals but we simply haven't found that to be the case.

To put numbers to that, for the 8-10 PCs which T.O.M.S. ran at any one time over the last (say) 30 years, in both business and domestic environments, there was only one failure. That was fixed in a day by a local PC fixer, at nil shipping cost.

Of the older PCs which have been replaced, all but two have been handed down to offspring and, as far as we know, are still working.

Why T.O.M.S. had such good experiences with the PCs isn't clear. Perhaps one reason is that they were all of the better-known brands, accepting the small price premium. Or it could well be that, like the Iyonix, the machines are relatively easy to upgrade, without disturbing other components, and certainly not requiring a strip-down.

Admittedly we simply don't know the true reasons but, based on our experiences, for us it has not been true to say that ARM-power is more reliable than PC-power; indeed on balance it is the reverse. So we do feel that anyone being understandably nervous about the reliability of a PC for use with VirtualRPC can be encouraged by this.

### PC hardware compatibility

The only time we've heard of any PC hardware compatibility problem was from a VirtualRPC user who installed an extra RAM chip in his aging PC. He'd fitted a low-cost chip whose specification was less than that already installed, so the whole system performance – including RISC OS – was dragged down.

He got a replacement RAM chip with an improved specification, and that fully restored performance. A lesson for us all.

### Peripheral equipment compatibility

Anyone familiar with early ARM-powered machines will be well aware that expanding their capabilities with hardware upgrades could so often be a frustrating experience, with little guarantee there wouldn't be compatibility problems affecting the peripheral hardware and/or the podule and/or the software application needed to make it all work.



Return to Contents

Against that, our experience of expanding PCs always was and still is that this is invariably trouble-free, primarily because compatibility testing is far more thorough and, generally, adding extra kit doesn't require the installation of a proprietary expansion interface. Indeed, it might well simply be a case of 'plug-and-play'.

Similarly, suitable drivers may well already be built into Windows but, in any event, more often than not the peripheral will come with an installation CD which contains the necessary Windows drivers (but see the *caveat* on page 11 regarding aged peripherals and Windows 7-10).

But will it all work with VirtualRPC? For all practical purposes, the answer in our own experience is a very slightly qualified "Yes".

To give just one common example, plugging any variant of the widely-available range of storage media into a matching socket, or a card reader plugged into the PC, and knowing that RISC OS can almost certainly access its contents via a HostFS mount, generally without the slightest difficulty, really is a major bonus.

Contrast that with the significant problem of getting card readers and media to work with each other, and to work reliably with RISC OS running on an ARM-powered machine.

Similarly, invariably there are no operating limitations with VirtualRPC. So if, for example, you plug an external 2TB hard disc USB drive into the PC, and access it via a *HostFS* mount in VirtualRPC, the full 2TB is available to both Windows *and* RISC OS; perfect for backing up your VirtualRPC installation – or even your complete PC including Windows...

As far as we know, the only incompatibility issue, which used to cause the occasional problem, was that an IDE DVD/CD burner installed in a PC might come up as a `'Broken drive'` under RISC OS. Invariably, the culprit turned out to be the CDFS driver being used – *CDFSSoftATAPI* – and the Second Edition covered the installation of different versions of the driver in the hope of finding one that worked.

However, the more recent versions of VirtualRPC incorporate a later version of *CDFSSoftATAPI* (v1.53) which seems to be far more compatible with recent *IDE* optical drives (whether used by VirtualRPC or ARM-powered machines, of whatever vintage).

But if in the event that you still hit problems with an *IDE* optical drive installed in the PC, or in any case if it is an *ATAPI* device, then you can always access the optical drive via a *HostFS* mount on the icon bar (page 36).

### RISC OS software compatibility

No doubt someone will prove us wrong, but we believe that virtually any RISC OS software title (excluding those which require their own proprietary hardware such as a podule) will either work – or can be made to work – with VirtualRPC-DL, -SE or -SA, i.e. with RISCOS v4.02.

We have literally hundreds of applications and utilities, some going back to the 1980s, and for us the figure is 100%.

With VirtualRPC-AdjustSA (i.e. RISC OS v4.39) a small number will not run. As far as we can tell, they're ones which seek to affect system sprites at start-up, such as Irlam's excellent *Album* thumbnailer. But there are various alternative workarounds which we'll look at in Section 13.

But this is purely a RISC OS issue; you will get exactly the same problems with an ARM-powered machine running Adjust.

An interesting and very welcome point that soon became evident, but which we cannot begin to explain, is that some applications which proved to be annoyingly flaky on occasions on our real RiscPCs and even Iyonix appear to be very much happier running under VirtualRPC, to the point that they're now evidently totally stable.



We're using exactly the same versions of these usually aged software titles – the famed *DiagramIt/GraphIt/Tablemate* trio from Dalriada are typical examples – so this is a mystery, although we know of others who are getting the same vibes.

Perhaps it's a filing system thing (using *HostFS* against *ADFS/IDEFS/SCSIFS*), or that more RAM is available (128-256MB against, typically, 64MB); we know not. But we aren't complaining and, of course, productivity is hugely improved as a result.

But what about the statement on page 46 that some software "can be made to work"? In general terms, there are probably three overall conditions why a particular piece of RISC OS application software won't run straight away under VirtualRPC (usually regardless of the variant):

• **Getting software to work (resources)** – The first situation is where you've copied the application across from another machine, in a compressed folder and so on, but once installed onto *HardDisc4*, it won't start up.

Usually the symptoms are that one or more error messages are thrown up saying '`[so-and-so] module Not found`'. This is the clue. Many applications require additional resources in order to work, generally in the form of modules installed in *!Boot.Resources.!System.* So if you haven't copied across and installed these modules, not surprisingly that's what the error message is telling us.

This is where taking the backup copy of *!Boot* from an 'old' machine (recommended on page 42) comes into its own. In principle, if the copied application complains it can't find a specific module, then it's a case of locating and opening up *!System* on this backup copy of *!Boot* (in the *Resources* directory), finding the relevant module and copying it into the identical location in *!Boot.Resources.!System* in the new VirtualRPC installation.

That should clear the particular error message next time you start up the application. But of course it may then trigger yet another error message, for another module, so you may need to repeat the exercise twice (or more) to clear the problems. However, to reiterate, *do not* try to short-cut the issue by gaily copying all the modules over to VirtualRPC.

That said, there is one area where in our experience a bit of time can safely be saved, and that's with getting a variant of the *Impression* or *Artworks* families to work. In their wisdom, Computer Concepts opted to have their own sub-folder containing all the modules required by their original applications. In some cases any one module was shared between two or more of CC's applications so, not surprisingly, the sub-directory is called *!CCShared*, and it's located in the *!System* directory.

So in order to get *Impression* to work for example, as well as copying the main application folder across from your source machine, it *may* also prove necessary to locate the *!CCShared* folder in the backup *!Boot.Resources.!System* folder and copy that into *!System* on your machine running VirtualRPC.

The same goes for copying *Artworks* across, as an application. But if you have an installation CD for it, then we'd suggest first getting *Impression* to work and then install *Artworks* afterwards, the reason being that *!CCShared* on the *Artworks* CD may install later versions of the shared modules.

• **Reducing RAM** – The second situation has already been mentioned: some 'older' application software won't work properly, if at all, with the full 256MB RAM configured for use with RISC OS v4.39. We know of two titles where this can be the case: *Artworks* v1.xx and *Fresco*. In both cases, reconfiguring VirtualRPC to limit the RAM setting to 128MB seemed to clear the problem.



Return to Contents

However, the other (preferable?) alternative is to upgrade *Artworks* to v2 or later and use a more up-to-date browser (under RISC OS or Mac OS or Windows).

• **Reducing colour depth** – The third situation is that some of your really aged software may not even load, typically coming up with "`Internal error: no trap for stack handler`" stoppers. That always proved to be a niggling frustration for T.O.M.S. when otherwise-valuable software could not be used.

Then the guy with the brain cell suddenly realised that all these titles shared a common heritage: they all pre-dated RiscPC or, to be more precise, 32K and 16M displays and deep sprites. There was the clue. He first selected a 256-colour screen mode before loading each title and – lo and behold – most of them then worked, first time, and just as satisfactorily as of yore.

That left only hiccups with deep sprites to contend with but, in the event, this turned out not to be a major stopper after all. Typically what happened was that the aged software title would load and appear to run OK but if, say, a file containing one or more deep sprites was loaded, problems then arose.

But this wasn't necessarily crucial. In all probability, a deep sprite would appear either with a seriously-incorrect palette (not surprisingly), or to be corrupted in some other way – or not even be displayed.

However, if these symptoms were ignored and the user carried on with what he wanted to do, such as editing a document, and then re-saved it, everything was correctly saved, *including* the deep sprite(s). In other words, the deep sprite problem evidently affected only the display.

This much-belated discovery pays dividends for our purposes. For example, it means we can now load a drawfile containing a deep sprite into, say, *Chameleon* (which is so old that you have to be of a certain age just to recall what it does...), ignore the seemingly-corrupt or 'missing' deep sprite, edit the draw elements in the file, re-save it, import it into a DTP document – and everything is fine. Another problem dealt with, and yet more productivity.

(Why was this not noticed with our RiscPCs? It was because display resolution was then far more important to us than colour depth so, with a 2MB VRAM limit, normally we operated in a screen mode of 1600×1200 pixels in *256* colours and, at the time, never used deep sprites, so these symptoms never arose.)

So to summarise, if you try to run a seriously-aged piece of software and it won't even start up, try dropping the display colour depth to 256 colours and you may then find it will work OK (display resolution doesn't seem to be an issue). And if you have problems displaying deep sprites, try ignoring them and, hopefully, the underlying file can be edited and re-saved without problems.



# 9. Printing with VirtualRPC

*This section applies primarily to running VirtualRPC on a host PC; however, we have been advised that, with the exception of the sub-sections dealing with UniPrint, it is also relevant to using VirtualRPC for Mac OS.*

High-quality printing was always a very important requirement in T.O.M.S.' business activities and, through long correspondence with VirtualRPC users, clearly this also applies to so many other people.

The really good news is that, without exception, everyone is hugely complimentary about the features and facilities which VirtualRPC (and the underlying PC) can and do provide, so it may help potential buyers and users to look at this important topic in detail.

It seems to be generally acknowledged that RISC OS printer definition files (PDFs) have always been limited in scope and capability. But with the enormous range of available printers (and seemingly daily changes), plus the reluctance of manufacturers to release details to RISC OS developers, this is perfectly understandable.

In the past, Computer Concepts' *TurboDrivers* provided a huge boost, as evidenced by the performance data recorded and discussed on pages 6-7. However, both RISC OS PDFs and *TurboDrivers* are limited to 4-colour (CMYK) printing which, for example, meant that for full-colour proofs sent for customer approval, T.O.M.S. had to go to the considerable expense of buying in Chromalin™ prints.

Spacetech's development of their *PhotoReal* drivers to support a very limited range of 6-colour, A4/A3+ printers in 2000 made a huge difference, with 'photo-real' print standards being possible for the very first time under RISC OS. Printing took a while, but the results (on expensive photo-paper) were superb.

Unfortunately, neither *TurboDrivers* nor the *PhotoReal* drivers would run on the Iyonix, even with the help of *Aemulor*, so the flagship ARM-powered machine was in effect a non-starter for us (and it transpires for many other people, for these and other reasons). But some time later the *Gimprint* (and now the *Gutenprint*) series of drivers arrived, available from the prolific MW Software on *www.mw-software.com/software/gutenprint/gutenprint.html*

It isn't possible to over-exaggerate the value of this splendid initiative. All of a sudden – and at long last – ARM-powered users could enjoy the sheer flexibility and productivity, not only of being able to choose from a repertoire of literally hundreds of contemporary printers, but also to produce (so-called) 'photo-real' quality prints from the widening 6-colour range. Even the cost of printers and photo-paper has dropped significantly.

However, there's a considerable limitation (which is fully acknowledged on the Gutenprint web page) and this is in the *very* long times taken to handle the more demanding print jobs. To put a number to that, the website quotes "more than an hour" for an Iyonix to print a single A4 photograph to full, photo-real standards.

The latest ARM-powered machines will most likely take far less than that, although the only test point we have is for an ARMX6 which took 17 mins to print an A4 photo to full photo-real standards. So it can be a general problem if you really don't want the computer to be tied up for perhaps very long periods.

But this limitation simply doesn't apply to VirtualRPC. Even if running on a modest PC, print times using either a RISC OS PDF, or the excellent and more universal *UniPrint*, are a very small fraction of what a stand-alone ARM-powered machine can achieve.



Return to Contents

Furthermore, contemporary host PCs will hand back full control of the computer typically from within just a few seconds for laser printing, up to no more than 60-90 seconds for a full, A4 photo-real print job. And usually there's then sufficient spare 'oomph' to allow continued use of the computer (including VirtualRPC) while, in the background, the Windows-driven printer gets on with the physical printing of the rendered file *and* even other background tasks.

So clearly, printing via RISC OS – from within VirtualRPC and tapping into the host PC printing system – is in a very different league compared with trying to do the same thing on any ARM-powered machine. How is this achieved? We'll look at the two main options in more detail shortly but, first, you need to set up the printer(s) under Windows.

### Setting up a Windows printer

Even if you've never used a PC before, be assured that setting up a printer really couldn't be simpler as it's generally a case of plugging it into a handy USB or parallel port (using a Centronics-USB adaptor cable if necessary), running the manufacturer's CD supplied, starting the Install process, switching on the printer when told to do so and letting the installation wizard do its thing.

Windows 'plug-and-play' will then normally be invoked, the printer will be registered automatically and the relevant drivers will be installed.

There remains only to set up the printer to suit your intended use, via the Printer Properties option, although the configuration can easily be changed between print jobs. Terminology varies between manufacturers but the printer user guide will sort that out for you. So for example, a few mouse clicks might select the print quality – typically Draft, Text, Text and images, Photo or Best Photo (setting the most appropriate dpi print resolution) and the paper size and type – Plain inkjet, Matt photo, Glossy photo, and so on.

As with RISC OS, the paper orientation can normally be left at Portrait as, for landscape-format printing, it will usually be overridden by the application in use. With some inkjet printers, and provided the printer is switched on, there may also be a 'live' display of ink cartridge contents; this feature may also be accessible by clicking on the Maintenance tab.

(As an aside, we find we get *far* more usage out of each ink cartridge as you continue to get 'countdown' readings of, say, 10% and 5% left, making it far easier to gauge when to change individual cartridges. Similarly, a laser printer can often be selected to 'EconoMode' [or similar] to extend the life of a toner cartridge very considerably.)

Just like RISC OS PDFs, it is also possible to install two (or more) 'virtual' printers for the same physical printer, simply by running through the Add a printer wizard for a second time and, when it asks whether or not you wish to replace the existing printer driver, select No and a *second* printer icon will appear in the window (accessed by selecting Start › Control Panel › Devices and Printers).



The reason for setting up a duplicate driver for a single physical printer is exactly the same under Windows as it is under RISC OS: you can have different configuration settings for each one to suit your purpose. So, for example, a colour/photo printer might have one driver set for printing Text and images (4-colour), while the other might be set to Best photo (6-colour, photo-real results on posh paper); the flexible choice is yours.

Note that the Windows printer title will be set up automatically during the installation and, invariably, will be the full printer name – which can be a bit of a mouthful, e.g. "`Epson Stylus Photo R220`". And if you add a second 'virtual' printer, this will have the same long name, suffixed by "`(2)`", and so on.

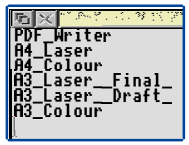
However, just like RISC OS, the name(s) can be changed to something more convenient by right-clicking over each icon, selecting Rename and making the preferred changes. So, using the above colour/photo discussion as an example, the two icon names might be changed to, say, "`A4 Colour`" and "`A4 Photo`" to reflect their intended use. These simplified names will appear whenever you go to print something, allowing you to select between the two (the equivalent of clicking on a particular printer icon on the RISC OS icon bar to select it).

We'll put all these optional features to good use when we look at controlling additional Windows printer functions from the RISC OS desktop.

### Setting up a printer with a RISC OS PDF

Setting up one or more RISC OS printer definition files is fully described in the VirtualRPC user guide but, in essence, we need first to 'Print to file' and then send the resultant file to the Windows printer system for the physical printing. So in the first instance, in the Connections window on the icon bar, we need to change from the default Parallel setting to the (Print to) File option.

Second, we need to enter the printer name into the window adjacent to the File button, immediately preceded by `HostFS::` (without a space and noting the double colons). Exactly what to enter for the printer name is shown by double clicking on the *ShowPrinters* task obey file in the *Printers* folder.




This task window displays all the 'printers' set up in the host PC's Devices and Printers window, noting that the example host PC shown here has a total of 5 physical/virtual printers attached, *plus* a 'Portable Document Format' Writer option; a reflection of the flexibility and sheer productivity one can routinely employ with both the host PC and VirtualRPC.

There are two points to note from the information shown in the task window:

• Although Windows supports the use of multiple 'soft' spaces in file names, RISC OS isn't as flexible and, to ensure things don't go amiss, it replaces all the spaces in the Windows names with the "_" underline character (noting that sometimes there are double "__" underline characters).

• If you change the printer name under Windows (from the default name entered during installation), the task window will reflect the change and the *amended* name is the one to enter into the File window.

So to give a typical example, after we installed our Epson A4 colour printer, initially we used the default printer name so would need to enter the mouthful: `HostFS::Epson_Stylus_Photo_R220` in the File destination window. However, as we later changed the Windows name to the more logical 'A4 Colour', we also needed to change the RISC OS File name to `HostFS::A4_Colour`.

Return to Contents

Once the document is rendered, i.e print(ed) to file under RISC OS, it is sent to the Windows printing system and handled as a 'Red Squirrel' print job. Computer control is handed back to the user far quicker than it would be with any ARM-powered machine and, on a reasonably powerful PC, physical printing then continues, in the background, with barely discernible effects on multitasking.

There are two areas where things can go adrift. First, if having entered the details into the File window, you get a '`Disc not Found`' error message when clicking on OK, that means the connection is incorrect. So double-check what you've typed into the File window, especially to confirm the entry is preceded by `HostFS::` (no space) and any double underline characters have not been omitted from the printer name.

If printing to file seems to go well but nothing is printed (although the printer indicates it is Ready), go to Windows and double-click on the relevant printer icon which opens its document queue window. This will show whether the Red Squirrel print job has in fact arrived, and whether or not is is being held up for some reason, e.g. by a previous file which has failed to print. In which case, right-click over the offender and select Cancel. Once it has been deleted, your job should then print.

One final snippet, which applies equally to ARM-powered machines and VirtualRPC. If there isn't a RISC OS PDF for your particular printer, it's often worth trying the nearest available alternative which can sometimes be made to work, albeit with a bit of necessary tweaking in the Print Setup window, and/or the print margins in the Edit paper sizes window.

For example, our LaserJet 1015 printer doesn't have a RISC OS PDF – but it works with the *LJ1100* PDF. On a RiscPC, it needed quite a bit of tweaking to print a page in the right place but, happily, under VirtualRPC and thence to Windows, this is not required.

### Using UniPrint

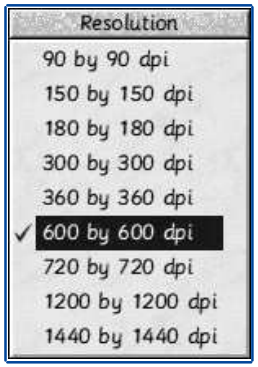
If one or more of your printer(s) do not have a suitable RISC OS PDF available, or if it is limited in some way (the lack of any photo-real capability being a common example), then we would strongly recommend using *UniPrint* from R-Comp (or *UniScan* which incorporates the enhanced print features).

A nice bonus is that, if you have multiple printers connected to the host PC, but not all have suitable RISC OS PDFs available, it's perfectly feasible to 'pick and mix' from the two driver systems.

The principles of *UniPrint* and the setting-up procedures are described in full detail in its user guide and, in our opinion, are slightly easier than installing a RISC OS PDF. In particular, there's no need to be so specific with the printer naming routine as, provided the name you use is unique, *UniPrint* will work fine. We'll look at this plus one or two other nice features later in this section.

As with using RISC OS PDFs, the speed with which *UniPrint* produces the goods is very much faster than any ARM-powered machine, although spooling the print file into the Windows system takes a little while.

But R-Comp stress the very valid point that there's no need to select an ultra-high print resolution if, at the Windows end, this is not reflected in the Windows driver settings. So if the physical printer is, say, a fairly standard 600dpi laser job, then selecting a Resolution setting of `600 by 600 dpi` under RISC OS will give both optimum quality and speed.






If problems arise with *UniPrint*, it's slightly easier to track down the culprit than when using a RISC OS PDF. If document rendering seems to work, but nothing prints out, it's first worth checking to see if the file is stuck in its Scrap directory which is in the *!Boot › Resources › !Scrap › ScrapDirs › ScrapDir › UniPrintVA* folder.

If it is, that means it hasn't been passed onto the Windows printing system, usually because the connection to Windows is incorrect. In that event, go into Windows, click on the UniServer icon in the system tray (a.k.a. the taskbar), click on the Printers and Paths tabs in turn and double-check that the correct printer is selected and that the path matches the above detail, for the variant of VirtualRPC installed. Make any necessary correction and, when you then click on Apply and OK, in all probability the printer will leap into life.

If it doesn't, then proceed as before: double-click on the Windows printer icon to see if the print file has arrived and, if so, whether it is being held up by something else.

### Desktop control of printers

For us, using *UniPrint* has another useful advantage beyond the sometimes restricted features of a RISC OS PDF. If you're using the latter, there's no control over printer options such as paper feed, or selecting between 4-colour and 6-colour photo-real printing, although this may not apply anyway, e.g. for the simple, desktop A4 laser printer used in the earlier illustration.

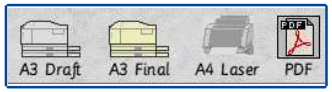
But what if the printer is rather more capable? In our own case, for example, T.O.M.S. used to have a hefty A3 laser printer which had two paper cassettes, normally loaded up with cheapo 80gsm paper for drafts and posh 100gsm coated paper for final printing. Was there any way of selecting between the two directly, via the RISC OS icon bar?

Thanks to a much-appreciated hint & tip from R-Comp, we found it was perfectly feasible to tap into *UniPrint* to do the job. Their cunning plan used the ability of Windows to set up duplicate printer drivers for the same physical printer, as described on pages 50-51.

We could now configure each of the two Windows drivers to select the different features required, in this example selecting between the upper and lower paper cassettes. Then, for ease of identification, the drivers were renamed as preferred; in that example, say `A3 Draft` and `A3 Final`.

Now, under RISC OS, *UniPrint* was set up so as to have two icons on the icon bar for the same physical printer and, using *UniSetup*, `Draft` and `Final` were entered in the two Printer Name icons, this being sufficient for *UniPrint* to be able to distinguish between the two.

Finally, we configured both the 'printers' on the icon bar so as to select the Auto setting in the *UniPrint* Paper feed icons. That lot was saved and, from then on, clicking on either printer icon to highlight it automatically selected the Windows A3 Draft or A3 Final driver, and thus the desired upper or lower paper cassette.




In exactly the same way, duplicate drivers can be set up for a colour printer so as to be able to select between, for example, 'Text and Image' (typically Epson-speak for 4-colour printing) and 'Photo' (for the alternative, 6-colour, photo-real printing), all done from the RISC OS desktop.

Note that this wrinkle can also apply to using duplicate RISC OS PDFs, always provided their features match your requirements. So it wouldn't work for selecting between 4-colour and 6-colour, photo-real printing as the latter isn't supported.



Return to Contents

### An elegant icon bar

Just like any extra HostFS mounts for additional storage media etc, the multiple printer icons we've been discussing will appear on the icon bar in the order installed.

So a little pre-planning is required if, say, you'd like to match the physical placement of printers on your desk (looking from left-to-right) with how their icons will appear on the RISC OS icon bar (page53).

But what's the PDF 'printer' icon all about? Bear with us; all will be explained on pages 55-57.

### Printing from Basic

A somewhat troublesome problem in the early days of VirtualRPC, now largely resolved, was printing from *Basic*. Even in the highly unlikely event that a legacy parallel port is present in the host PC, RISC OS running under VirtualRPC cannot access it *directly*, so printing directly from *Basic* (spooling listings for example) isn't possible.

Reportedly this put a surprising number of *Basic* users off VirtualRPC but, eventually, third parties came up with workarounds. As we're totally unfamiliar with their workings, we can only report that Jim Lesurf's solution is to convert from *Basic* to *Draw* format, using his *DrawGen* utility, and then print out the drawfiles. See Archive 10:6, 10:7 and 10:12.

Brynmor Owen then extended Jim's facilities by adding *DrawGen Basic*, including an extensive library of *Basic* routines and functions. His explanatory articles are in Archive 12:10, 12:11 and 13:1.

Finally, a more universal, commercial solution is now incorporated within *UniPrint* (page 52) which has a 'Text mode printing' feature, supporting direct printing from *Basic*, *Edit*, *Messenger Pro*, *Zap*, etc, to a Windows-driven printer.

Note that, with this solution, a Corpus-style font face is no longer required to be installed in the printer (which is just as well as most modern printers have dispensed with built-in fonts).

### Monochrome printing (UniPrint)

Further mention of *UniPrint* leads us to the topic of using it with monochrome (e.g. laser) printers which do not have a RISC OS printer definition file available, an example being the A3 laser printer discussed on page 53.

For best-quality monochrome printing from such a beastie, from RISC OS using *UniPrint*, one might logically set the latter's Printer options › Quality to Grey, halftoned (fine) which we found gave superb results with spot 'colours', i.e. greyscales.

However, graduated areas (e.g. from *Artworks*) and bitmap images (e.g. *JPEGs*) always looked rather rough, with the boundaries between one greyscale and the next being quite evident, suggesting the number of greyscales produced by *UniPrint* was limited. But R-Comp came up with another cunning plan which was to set a *colour* option – even if printing to a monochrome printer! *Que?*

There are two colour options and we found there are no discernible differences between Colour, halftoned and Colour, halftone (fine). However, either worked an absolute treat with the laser printer for both graduated areas and bitmaps, giving much smoother transitions between different greyscales.

We see no reason why this shouldn't apply to all monochrome printers, whether laser or inkjet; a welcome though unexplained bonus being that, with Colour selected instead of Grey, the print file is only about half the size, speeding up the combined rendering/spooling/printing process by around 40%.

### Edit paper sizes

This sub-topic applies equally to both ARM-powered and VirtualRPC systems where RISC OS comes with a built-in range of standard paper sizes; mainly the A2-A4 series plus some other 'foreigners' However, you can introduce other, user-designated paper sizes by using the Edit paper sizes feature.

We make use of the feature as we often need to produce documents in the USA/Canadian 'Letter' format (215.9 × 279.4 mm – or 8.5 × 11 inches in real money), and 'Tabloid' format (279.4 × 431.7mm – 11 × 17 inches). This is handy as, if you set (say) 10mm margins in the document, it's *just* possible to squeeze draft prints out of most standard A4 and A3 printers for proofing.

### Summarising so far...

To summarise this important topic so far, printing under RISC OS using a standard printer definition file, or alternatively *UniPrint*, provides very fast document rendering.

Depending on the 'oomph' of the underlying host PC, rendering will generally be much faster than Computer Concepts' invaluable *TurboDrivers* or Spacetech's *PhotoReal* driver, and most certainly far, far quicker than, say, using a RISC OS PDF under ARM-power (see the test data on pages 6-7, tests 19-22).

And this doesn't take into account the fact that, once document rendering is completed and computer control is returned to the user, Windows will then get on with the physical printing in the background, generally with little or no adverse effects on whatever else you wish the computer to do. This routinely provides a big productivity bonus.

### Ultra-high-speed printing

But extraordinarily, there's even an ultra-fast(er) printing method! If you first move the source document across into Windows (e.g. in the universal Portable Document Format – the other 'PDF') and then print it *directly*, the sheer speed of response is a jaw-dropper. In fact, until you get used to it, it can be a wee bit scary.

Even with an A4 page sent for maximum-resolution, photo-real printing, the activity icon appears typically for only 2-5 *seconds* before full computer control is returned – after which the physical printing takes place entirely in the background.

Take a moment to think about that: To print an A4 page to full photo-real standards on an Iyonix, using *Gutenprint*, can take an hour or so. The same page, printed directly under Windows, and to exactly the same standard, takes a mere 2-5 *seconds* before full computer control is returned to Windows or to VirtualRPC. Nice one!

We find this gives us extraordinary productivity in use. To give a rather extreme example, one of us was once required (by SWMBO) to produce three copies of a 30-page, A4 sized booklet of much-loved family photographs for doting grandparents. You know the sort of thing.

He prepared the job using *OPro* under VirtualRPC, converted it into a PDF file (details in a moment), passed it to Windows and printed it directly under Windows.

Full computer control was returned *in under two minutes*, following which the printer ran off the 3×30 pages, entirely in the background, *in under two hours* – including replacement of various ink cartridges! The results were outstanding.

How long would printing the *OPro* document tie up a stand-alone ARM-powered machine? On the face of it, many hours continuous running to produce just one copy. So is it even feasible to produce such a multi-page/multi-copy document using ARM-power? Frankly debatable (a polite way of saying not a hope).

### 'Printing' to portable document format

The notion of producing documents under RISC OS and then printing them *directly* under Windows – at ultra-high speed – will usually require the document format to be changed to something which Windows can handle.

For instance, *Ovation Pro* documents can be handled and printed by the Windows port of the same application. *Easi/TechWriter* files can be exported as *Word* documents and printed directly (not necessarily via Word itself; more on that in Section 13). Similarly, *Impression* documents can be passed to Windows, loaded/imported into Ovation Pro for Windows (*OPW*) and printed.

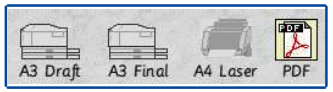
Another example is if your printer supports PostScript; the source document can be converted into PS format under RISC OS using the *PoScript2* printer definition file, or better still *PS3*, and passed to Windows for the physical printing. This works well, although the choice of compatible outline fonts can sometimes be restrictive or troublesome.

But arguably, a far more reliable, easy and universal solution is to convert the source file into Portable Document Format (the 'PDF' abbreviation will be used in this context in the rest of this section).

A significant advantage is that, once the document has been converted into PDF, it can then very easily be double-checked before printing, using any one of a range of PDF readers available for RISC OS, Macs and Windows.

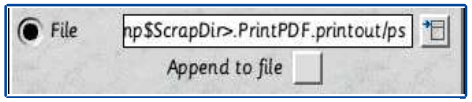
In the early days of RISC OS, PDF was something of a 'foreign' format. But now we can say that converting any document into PDF is very well supported by RISC OS. So in turn, this means that VirtualRPC users can easily – and quite quickly – produce a document and convert it to PDF (both under RISC OS), pass it to Windows and print it directly – at ultra-high speed.

Even with a single-page image file (e.g. a photo), it's often far quicker to do it this way than printing it under RISC OS, and *very* much quicker for a multi-page file. For us, this is so productive that we no longer have a colour printer on the RISC OS icon bar but, instead, replace it with a PDF 'printer'.




There's now quite a choice of methods to produce PDF files under RISC OS, ranging from the application-specific export features of (for example) *Artworks* and *Easi/TechWriter*, universal commercial offerings such as R-Comp's *PDFsuite*, and there's even a universal, freeware solution in Stephen Fryatt's *PrintPDF*. The choice is yours.

Purely through personal choice, we use *PrintPDF* for general purposes, largely because it's so easy to set up. Open the Printers › Printer control window; drag and drop a PostScript printer definition file into it, e.g. *PDFs › Generic › PoScript2* (or set up *PS3*); select it and open the Printer control › Connection window; click on the File option; finally click on the sub-menu icon at the right end of the File box and drag the Printout icon over the *PrintPDF* icon on the icon bar.




Save that lot and, thereafter, whenever you select this PDF 'printer' and print a document, a PDF file will be generated.

The universal PDF writers available for RISC OS all seem to work fine up to the limit of A2-sized documents (matching the default set of paper sizes in the variants of *Printers*). But if you have something physically different, then in principle it's possible to edit the paper sizes in *Printers* to cater for the larger formats such as A1, or even A0.

T.O.M.S. used to do this, with mixed results, and it was easier and certainly more reliable to import the source documents into an application which supported their export, as PDF files, at the extra-large sizes.



So as their huge flow-charts (larger than A0!) needed to be sent to the printers at full-scale, they first dropped the drawfiles into *Artworks* and then exported them as PDFs.

This was fast and routinely successful, again with the advantage that the PDF files could be double-checked before despatch by viewing them in a Mac or Windows PDF reader, this being far quicker and more reliable than using a reader for RISC OS. (PDF readers for Windows are discussed in Section 13.)

For the USA/Canadian 'Letter' and 'Tabloid' paper sizes, PDF files can be produced to those dimensions by using the RISC OS Edit paper sizes facility discussed on pages 54 and 55 and sent to the customer as-is.

### Generating landscape-format PDFs

There is a potential hiccup if your source document to be converted to PDF is both in landscape format *and* does not happen to have any outline text on the page being converted.

In this circumstance, PDF *readers* (RISC OS, Mac or Windows) will by default have a tendency to display the page in portrait format, such that your lovingly-constructed document will be rotated through 90 degrees. In general, the PDF reader will have a menu option to rotate the display back into landscape format, but that doesn't impress a finicky viewer and, naturally, we'd like to avoid this if possible.

The techie reason for all this was clearly explained by that nice Martin Würthner in Archive 21:5 p32 so we won't repeat it. However, he also came up with a very neat workaround, so here are the outline details.

If when you double-check the PDF and find it is being rotated, the trick is to put some 'dummy' outline text anywhere on the landscape page and re-convert it. The PDF *reader* should then interpret that as the signal that you wish it to display the page in landscape format, and oblige.

Note that it is *outline* text which is the trigger. So if, for example, you have a landscape-format drawfile which contains text which has been converted to path, then this will fail to trigger the PDF reader, and the display will be rotated.

As far as the 'dummy' outline text is concerned, with one reservation this this can be anything you like; even a single character will suffice. However, avoid using *italic* text as, for example, Homerton '*Oblique*' doesn't work!

Obviously we don't want the dummy text to show on the page, or be printed, so the other bit of the wrinkle is to set an outline text colour which cannot be seen by the viewer, but which the PDF reader will still 'see' (even if you can't...).

So if, for example, your source document is an *Impression* or *Ovation Pro* file, a simple method would be to include a small frame, type something into it, and then make the (text) foreground colour the same as the (page) background colour, e.g. white. You can't see it – but the PDF reader can...

If the document is multi-page, you may find you need to apply the 'dummy' outline text to every page, in which case it would be easier to add the mini-frame and text to the master page.

Whether or not you need to apply this workaround seems to depend very much on the PDF writer. *Artworks*, for example, doesn't seem to need it but, although PrintPDF has an Auto page rotation custom setting, it's a tad fiddly to set up. So a bit of trial-and-error is in order.

But the workaround discussed above is simple to set up and works for any RISC OS PDF writer, avoiding disappointment if the PDF reader doesn't play the game. This is yet another reason for double-checking your output post-converrsion to PDF, using a Mac or Windows PDF reader, before sending it off.



Return to Contents

# 10. Multiple monitors/displays

### Enter dual monitors

The simultaneous use of more than one monitor per machine is perhaps a new area for the majority of RISC OS users, so we'll consider the "Why and how?" for both Windows and RISC OS running under VirtualRPC. We can split this discussion into two separate elements: first, duplicating a display onto a second screen and, second, expanding (extending) the desktop area over two or more monitors.

### Replicating a display

The main idea behind being able to duplicate a display is if the primary screen is restricted-view in some way. A common example is that, if you have a modest laptop, or a Windows tablet, the display may well be limited both in physical size and in resolution (e.g. 1280×1024 pixels). So very often a window cannot be revealed in its entirety and still remain readable, and having to scroll in one or both directions becomes the norm.

Similarly, even though relatively up-market laptops and tablets have 'widescreen' displays, with an aspect ratio of either 1.6:1, or even 1.77:1, a website display can be displayed full-width but is relatively shallow in height and needs lots of vertical scrolling.

However, laptops invariably have a Video and/or TV Out port, while tablets may have some form of mini-port, in all cases allowing a physically larger display to be plugged in and viewed. But unless you have a sizeable, 'full HD' monitor or TV (screen resolution 1920×1080 pixels), display quality may not be much better than that of the laptop or tablet screen.

Nevertheless, a 'full HD' monitor will give a physically larger and maybe a higher resolution display than the typical laptop screen, so it is well worth setting one up for use when the laptop is back at base, whether that is home or business.

To illustrate this, we routinely plug a high-resolution monitor into the laptop and tablet, the Windows and RISC OS displays then being every bit as good as that of a desktop PC.

Setting up the monitor is normally a trivial procedure, as described on pages 29-30 and 33; and under VirtualRPC, the higher resolution display can then be selected in the usual way.

### Expanding the desktop

On host PCs, Windows enables the desktop to be extended by spreading it over two monitor screens. So for example, a pair of 'full HD' monitors will give you a real (not virtual) working desktop area of 3840×1080 pixels.

Once it's set up, the Windows taskbar and pinboard icons may remain on either one display, or in both displays (Windows 10), but the mouse pointer is free to travel over the full expanse of real-estate, and windows, folder viewers, etc, can be dragged from one screen to the other as required.

How can this be achieved? The setting up and configuration procedures are reasonably straightforward, although they do vary between different versions of Windows. But with Windows 10 for example, all is explained and illustrated in a very clear fashion on *'How to set up multiple monitors on Windows 10'*.

It's relevant to note that the two displays do not have to be the same physical size, nor the same aspect ratio, nor the same screen resolution, as each is set up individually. So, for example, you could use a laptop display for 'Screen 1' and extend the desktop onto a conventional, plug-in monitor for 'Screen 2'.

Running a second monitor with a laptop or tablet is especially handy as the display will normally be larger, at a higher resolution and more suitable for, say, DTP work, while the smaller screen could maybe display a related graphic, or a website, or your email inbox, etc.



### Multiple monitors with VirtualRPC

Do multiple monitors work with VirtualRPC? In principle, yes – and very well. The only limitation is that, when you put VirtualRPC into full-screen mode, RISC OS (with its blue pointer) is constrained to only one display which, by default, is the primary, 'Screen 1' image, e.g. a laptop screen.

But generally you can configure the plug-in monitor to be the primary display when setting up the dual displays by ticking the Use this device as the primary monitor box (on some machines, this option may not be available).

With VirtualRPC running RISC OS full-screen on one monitor, the Windows desktop, task bar and application(s) can still be running and displaying their wares, concurrently, on the other. But the white Windows pointer isn't displayed! No problem: press ‹Alt–Enter› to drop VirtualRPC into its own window – and the Windows pointer reappears.

### Triple monitors with VirtualRPC?

The Second Edition included a link to a posting by R-Comp on *c.s.a.announce on 22 Jan 2010* which covered how they could take things a stage further and support *up to 3 monitors* on their RISCube systems (excluding the RISCube Mini). Furthermore, the *RISC OS* display could be extended over the whole of the much wider real estate.

R-Comp point out that the system requires compatible hardware and special MDFs which they can supply. But being a 'special' the system is not plug-and-play unless it has been set up first.

Although in principle R-Comp can supply an 'upgrade kit' for self-fitting, this does require the user to have some technical understanding and be prepared to make the adjustements needed to get the kit to work correctly.

So if you're interested, by far the best ploy is to discuss your needs with R-Comp to see if they can supply you with a full 'off-the-shelf' system for purchase.



Return to Contents

# 11. Sundry VirtualRPC issues

We've reached the stage where the remaining topics concerning primarily VirtualRPC are relatively short bits and bobs which appeared piecemeal over the life of the original series of Archive articles from 2003.

So having discarded the ones which no longer apply (mainly due to the beneficial effects of subsequent upgrades), we'll update and record the remainder, with a few later additions. They're not in any particular order but we'll start up with a few, erm, start-up hiccups.

### Unlock key failure

Setting aside the few complaints about the principle of VirtualRPC having any anti-piracy feature in the first place, the method used of obtaining and entering an unlock code during the very first start-up seems to be perfectly understandable and satisfactory for most users. Once it's done, it's done.

However, there used to be an occasional hiccup where everything seemed to work for a while – but then VirtualRPC start-up stalled, with the Registration window appearing and with an invalid unlock code message. Inspection of the details in the window showed that the code was correct – but that the host PC seemed to have changed its product ID; hence the stopper.

Possibly because the problem seems to have been dealt with in later versions of Windows, we haven't heard of anyone experiencing it for some time.

Nevertheless, should you happen to hit this problem, VirtualAcorn's advice is simply to shut down and restart the host PC as there's a fair chance that the re-boot will resolve the problem. But if the symptoms are repeatable, you can apply for a replacement unlock code via the recently-streamlined '*RUCHS*' registration system.

However, retain a note of the *old* unlock code for use just in case the difficulty (or whatever caused it) should recur 'in reverse'.

### 'Broken' installation

If VirtualRPC fails to start up, with a Broken installation error message, or a 'green screen' comes up with an error message saying File '&!Boot' not found, usually this means that one or more of the VirtualRPC configuration files, usually the CMOS file, have become corrupted.

Normally, this would require a complete re-installation of VirtualRPC. However, if you nip back to page 25, you'll find we've pre-empted this circumstance by suggesting you make a backup copy of the most likely culprits.

So in principle, it's then simply a case of opening the VirtualRPC-*xxx* folder under Windows, finding the backup folder you generated and then dropping a copy of each backup file over the one(s) thought to be suspect.

We recommend this is done one at a time, using the copy & paste method for safety, and attempting to re-start VirtualRPC each time until it all comes good. We suggest copying over the *VA* file first, then the *Plugins* folder and, finally (if necessary) the *Model* file for the particular emulation you are using (e.g. *StrongARM*).

The order isn't important but, if it does prove necessary to copy across the default *Model* file, you'll then need to re-configure it to take account of any changes you've made, invariably the `[Ram]` setting and perhaps re-naming the original *HardDisc4* mount. So don't worry if VirtualRPC does eventually start up but with lots of `Not found` error messages – these will clear once you rename the main HostFS disc 'drive' (pages 27-29).

And remember that, in VirtualRPC-SA and -AdjustSA, there are *3* different *Model* files (*ARM7*, *ARM7500* and *StrongARM*), so you may also need to re-configure any other(s) you use.



### Start-up to a blank screen

We've already mentioned that, when starting up VirtualRPC on a fairly powerful host PC, you may well not even see the RISC OS start-up screen. The display will blank out for 3-4 seconds but then take you straight to the desktop. Initially this can be a little worrying but you soon get used to it.

(Although not confirmed, we think the reason for this *may* be that, during a full-screen VirtualRPC start-up, the screen mode changes initially from low-resolution (800 × 600 pixels?) to whatever higher-resolution display the monitor is set to (maybe 1920 × 1080 pixels?), and that it takes a little while for the host PC graphics card to make the change.)

However, on rare occasions, initial start-up will appear to proceed OK but seemingly doesn't proceed as far as the full-screen desktop, so you think you're stuck with a blank screen.

This can be cleared by quitting and re-starting VirtualRPC but, normally, that isn't necessary. Instead, press ‹Alt–Enter› to put VirtualRPC into a window, press ‹Alt–Enter› again and VirtualRPC will return to full-screen mode, invariably with the RISC OS desktop now displayed.

Sometimes it's necessary to press ‹Alt–Enter› three times before it all comes good. (This seems to happen very occasionally under Windows 10.) So if in doubt, simply keep pressing ‹Alt-Enter›.

### Running VirtualRPC – twice

Something which catches us wrinklies out quite regularly is that it's possible to start up multiple copies of VirtualRPC. This can happen typically if we're running it, then minimise it to do something under Windows – and simply forget that it's still running (a senior thing). So when we try to start VirtualRPC – a second copy starts up!

Sometimes, it isn't obvious that the second copy is running. But on other occasions, you may get an error message saying `The system was not correctly shut down` which, although not strictly correct, is a clue that you've got the two copies running. In which case quit the second copy in the normal way and maximise the original copy that's already running.

As far as we can tell, nothing really drastic will happen if you do have two copies running. Some users report they've inadvertently corrupted a file but we've never (knowingly) suffered that symptom.

### Minimising VirtualRPC

Where you most certainly *can* corrupt a RISC OS file however is if you're a bit cavalier and leave a document open when you drop VirtualRPC into a window (or even minimise it) to do something under Windows. This can surely happen with the *Impression* family, *Ovation Pro* or, indeed, any application which has an Auto-Save feature. So let's see what's causing it – and how to avoid it.

Take, for example, a situation where you were working on an *Impression* document and you pressed ‹Alt-Enter› to drop VirtualRPC into maybe a small window. Although the text in the window may no longer be legible, it's important to appreciate that both the text application and VirtualRPC *are still running*, albeit in the background. Similarly, the RISC OS pointer is still active.

So if you're doing something under Windows and press ‹Enter›, *and* if the RISC OS pointer happens to be over the active (*Impression*) document, this can and will position the caret somewhere in that document. Then, there is a risk that if you press ‹Enter› under Windows, a new paragraph character will be entered (unseen) into the *Impression* document.

After a while, the *Impression* prompt `Save <document> now?` will be triggered – although you probably won't see it because it's in the small RISC OS window.



Return to Contents

But if you then again press the ‹Enter› key (while still working under Windows), the inadvertently-corrupted *Impression* document will now be saved. Drat.

If our experience is anything to go by, you can quite easily enter multiple, unwanted new para characters into the RISC OS document – and then inadvertently save and re-save the document, several times over. Double-drat.

The obvious precaution is to ensure that all open RISC OS document windows are closed *before* dropping VirtualRPC into a window, or minimising it. Or perhaps better still de-select the Auto-Save feature in the various RISC OS applications you use, such as *Impression* and *Opro.*

But if you're working under Windows with VirtualRPC dropped into a window – or even minimised – and the computer "beeps" at you for no apparent reason, this could well be an audible 'Handle with Care' warning that something (unseen) is happening and which could spoil your day.

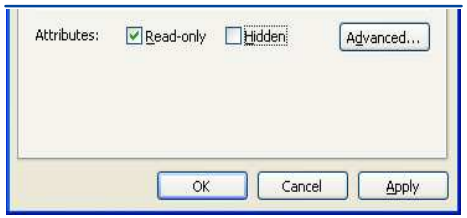
### Filer hiccups

Compared with the myriad problems we've had over the years trying to get IDE and SCSI peripherals, podules and their associated software to work together on ARM-powered machines, it's very pleasing that this unfamiliar animal *HostFS* seems to be very benign.

However, it isn't totally trouble-free and there are a few small hiccups to be aware of. First, the file protection system which we're used to with ADFS etc doesn't work with HostFS.

Specifically, if you click ‹Menu› over a file, then select Access and click on Protected, nothing happens – the Unprotected option stubbornly remains ticked. The same symptom goes for the Public/Private and Access details options.

The reason is that, as far as the host PC is concerned, a RISC OS file (or folder) is a standard Windows object, and will respond accordingly. So if you really do wish to add access protection to a file, it's necessary to find the same file in its Windows folder, right-click over it, select the Properties option, tick the Attributes: Read-only box, and finally click on Apply and OK. What a palaver.




Even then it's a little confusing as, under RISC OS, neither the Unprotected *nor* the Protected option is ticked! However, if inadvertently you now try to overwrite or delete the (Read-only) file, you will get an `Access violation` error message and the file will *not* be affected. So that's OK, and above all 'safe'.

Whether you go to the trouble of making various files read-only in this convoluted way, or leave them 'unsafe', is a personal decision. Being pragmatic souls, we leave them unprotected – but ensure we have plenty of backup copies in case of accident.

The corollary of the above discussion is that, if when attempting to access any file under RISC OS, such as overwriting or deleting it, you get the `Access violation` message, that simply means that Windows is stopping you. The most likely reason is that the file has become read-only, in which case you need to find it via the Windows directory tree and *UN*tick the Attributes: Read-only box.

### Deleting RISC OS files/folders

On rare occasions, you may find that – even if the Windows Attributes: Read-only box is not ticked – you may still get the `Access violation` error message on a file or, perhaps more often, a 'broken' directory. In this instance, the file may well have become corrupted and you find you can do nothing with it, including deleting it.

This is a RISC OS thing, so can of course happen equally on ARM-powered machines. With a '`Broken directory`' you can try to correct it using *DiscKnight* but, if this doesn't fix it, you really *are* stuck with it (apart from re-formatting the hard disc drive...).

But very fortunately, with VirtualRPC, Windows can help us out. If you locate the folder or file within the Windows directory tree, right-click over it and select Delete, then invariably that will force a deletion.

### RISC OS → Windows extensions

We've sorted out how to ensure that RISC OS filetypes are retained when passing objects from Windows to RISC OS, including adding the 'missing' sprite filetype (page 43) – but how about in the other direction?

In general, when passing an object from RISC OS to Windows, the three-character RISC OS filetype will be added to the filename and, in Windows, this comes out preceded by a ',' comma.

So if for example you pass a *Schema* file called *Calc* to Windows, it will appear under Windows as *Calc,c26* (with an 'unknown' filetype icon and note the comma followed by the 3-character RISC OS filetype *c26*).

But if Windows spots it as a filetype which it recognises, then it will respond accordingly. So for example, a JPEG or a PDF will be treated as such and be displayed as *‹filename›.jpg* or *‹filename›.pdf* (note the full-stop) and with an icon appropriate to the filetype. And if you double-click on the icon, the file will normally be auto-loaded into a JPEG viewer or PDF reader.

But for those objects which are not recognised, you can manually change the three-digit extension to something standard, but this time preceded by a '.' full-stop.

An interesting point which we'll look at in Section 12 is that some RISC OS 'native' filetypes such as sprites and drawfiles will be recognised by and can be loaded/imported directly into some Windows applications. And of course, the RISC OS text filetype (`fff`) will be recognised as a text file and can be loaded into Windows lookalikes. More details on that shortly.

### Multiple Windows extensions

A small oddity is that, on occasions, a RISC OS file can end up in Windows with double extensions! For example, if you prepare a compressed folder using, say, *SparkFS* and give it the RISC OS extension */zip,* it appears in Windows as *‹filename›.zip.zip* – whoops.

Not to worry, Windows seems to be quite happy with this (the multiple extensions may be either the same, or different), and will handle the object as normal. Indeed, it can even be attached to an email, seemingly without problems at either end.

### Windows keyboard issues

A fundamental limitation when using a Windows-style keyboard with the host PC is that there is no Print key as we know it and the PrtScr key doesn't have any effect under RISC OS. This is undoubtedly a niggle but one quickly gets used to using the alternative menu options, or (where provided) hot-key combinations, or the Print icon (or button on button bars) within RISC OS applications.

But if a keyboard or button bar shortcut is not built in, there are of course other ways of doing it. For example, for a limited number of our most-used applications, we set up hot-keys for them, using *Keystroke*. In this way, pressing ‹Alt–F10› or –F11 or –F12 will open the application's print dialogue box for *Draw*, *OPro* and *Publisher* respectively.

The only other Windows keyboard incompatibility we know about is that, unlike a RISC OS keyboard, the two ‹Enter› keys cannot be reconfigured individually. So this means for example that our favourite flight simulator (*Interdictor*) cannot be coaxed out of the start-up map display. Another good reason for retaining a real RiscPC...

### Three-button mice

It transpires that some VirtualRPC users still prefer to use conventional, three-button mice, rather than having to press the button on the new-fangled jobs with a scroll wheel.

Finding them on the high street is very unlikely these days but, in the original articles, we mentioned that Maplin still do a suitable three-button USB mouse, and we used them ourselves for a while.

However, we've since had reports that they tended to become unreliable in use, and sadly Maplin are no longer trading anyway. HP still do them – albeit currently at £45 each! – but you may find them on eBay, although we cannot comment on reliability.

For those of you wishing to use the scroll wheel on a conventional mouse *under RISC OS*, this worked a treat on the ARM7000/7000+ ARM-powered computers, and that valuable feature is faithfully emulated in the ARM7500 model.



# 12. RISC OS → Windows Ports

A small number of RISC OS application software titles have been ported to run under Windows and, for the sake of this discussion, we'll assume that VirtualRPC users will already have or otherwise be reasonably familiar with the RISC OS versions.

A common factor is that all can be run as stand-alone applications under Windows and this can be very handy if, for example, you're running the title(s) under RISC OS at home and have a PC at work, without VirtualRPC.

However, in a number of cases, there are distinct advantages to running an application under RISC OS and Windows on the same host PC, so we can limit the extra details to the value of running them under *both* operating systems, their main differences and what needs to be done to get the two versions to talk to each other.

A factor affecting all those which can print documents is that (not surprisingly) printing directly via Windows is extraordinarily fast (page 55).

### Ovation Pro for Windows (OPW)

David Pilling's splendid Ovation Pro for RISC OS (*OPro*) is our most-used and hugely-admired application. So when Ovation Pro for Windows (*OPW*) was released, we quickly leapt aboard.

The whole thing is available for download from *www.davidpilling.net/ovationpro.html* in the form of a 'virtual CD', so from that you can pick-and-choose which bits you need, including a range of applets, import filters and Windows Type1 fonts which are lookalikes of the standard RISC OS set.

The application comes on a very welcome and fully-functional, try-before-you-buy arrangement, so do give it a test run.

Even if you don't have *OPro*, but instead use a variant of *Impression*, you might like to move a sample *ImpDoc* across to Windows, change the *,bc5* extension (with the comma) to *.imp* (with a full-stop) and drag-and-drop it over the *OPW* icon. (Who said Windows doesn't support drag-and-drop?)

The Impression import filter will leap into play and *very* quickly display your *Impression* document in *OPW*, in all its glory – and running under Windows! Be very, very impressed.

When *OPro* users first fire up *OPW*, it immediately looks very familiar, albeit bearing in mind that the standard Windows menu bar will be present, and the button bar arrangement will be as for a default installation.

But therein lies a trap for players: unavoidably, there are detail differences between *OPro* and *OPW* so, unless you dig deep into the detailed user-guide, some things won't appear to work as you expect. A typical example is that the ‹F10› function key for the RISC OS version (Single shift font) moves to ‹F11› under Windows.

If you already have *OPro*, in all probability you will have customised the button bar, the stationery cupboard and so on to suit your purposes and personal preferences. But when first faced with the default *OPW* installation, the first question is how to set about customising the button bar? Especially as your RISC OS variant might use some of your own buttons – and they're in a 'foreign' (to Windows) sprite format...

Here's the good news: *OPW* is not only happy to handle RISC OS sprites and drawfiles in all respects in its documents (but not *Artworks* images) but also has a Custom applet containing a safe repository, where you can copy in the relevant files from your customised *OPro* installation.



Return to Contents

The repository is located in the *C: › Program Files › DavidPilling › Applets › Custom › AResources* folder and, typically, the files you might copy across from *OPro* are:

• The *!Macros* file from the *!OvnPro › AutoRun* directory, giving it the extension *.csc* (with the full-stop).

• The *Buttons22* and *Sprites22* sprite files from the *!OvnPro › !OvnResDir › UK* directory, in both cases giving them the extension *.spr* (with the full-stop).

Finally, you need to set the *Custom* applet to Active (via the *Applets › Applet Manager › Custom* menu) and, when you re-start *OPW*, it will now use all your customised files. You will still need to reconfigure and rearrange the button macros to suit your preferences, but this is far quicker than starting from scratch.

In a similar manner, all your stationery templates (including those in sub-directories) can be copied across from the RISC OS *!OvnPro › Applets › !Statnry › Stationery* directory to the *OPW* Stationery applet folder in *C: › Program Files › DavidPilling › Applets › Statnry › Stationery*.

As far as extensions are concerned, any *OPro* document passed to Windows will have the extension *,b27* attached (with the comma) so that needs to be changed to *.dpd* (with a full-stop), at which point it will display the familiar icon and the document can be loaded in the usual ways.

When passing *OPW* documents from Windows to RISC OS, the correct filetype can be retained by adding the detail `b27...dpd` to the Hostfs *Extensions* file (page 43).

There remains only to match the RISC OS outline fonts in an *OPro* document with the Windows Type1 fonts contained in *OPW*. This is done through the Misc › Font manager menu where Windows fonts can be either ignored, replaced or mapped to their RISC OS equivalents in a very flexible manner. That can potentially be time-consuming however, although it's a one-off procedure and, by default, the basic RISC OS 'set' (31 font faces) are already mapped to their Windows equivalents on the *OPW* CD.

### DPScan for Windows

Originally, *OPW* came with some bundled titles which would work only after the application had been registered. One of these was a Windows port of David Pilling's long-standing *DPIngScan* for RISC OS. We guess the majority of you use the latter, perhaps as your standard scanner controller but also with lots of useful image-processing functions.

The Windows version is named rather more conveniently – *DPScan* – and, in very general terms, does the same jobs and with the same menu functions, It can be downloaded via the *OPW* 'Virtual CD' address on page 65.

Thanks to a *very* generous gesture by David Pilling, *DPScan* may now be used as a stand-alone application and no longer requires *OPW* to be installed and registered for it to work.

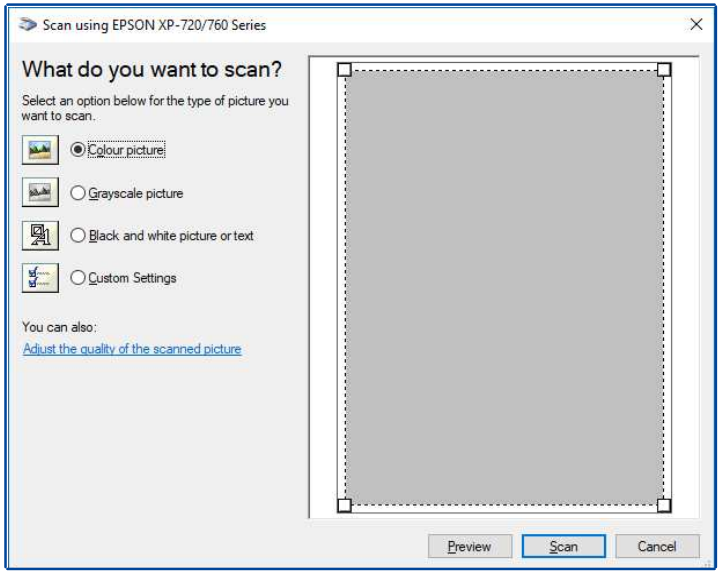
This is a little-known but splendid offering which we can strongly recommend to all. Using the title under Windows is a treat. Although it's a seamless amalgam of the two linked applications (scanning and bitmap editing), it's worth looking at these separately to consider the differences with *DPIngScan*:

• **Scanning** – Before *DPScan* and, later, R-Comp's *UniScan* became available for VirtualRPC users, we necessarily used Epson's own complex, clunky, unwieldy and seriously mis-cast 'Smart Panel' under Windows. Click on the wrong button and we had a fatal crash on our hands.

The scanning side of *DPScan* is a different animal altogether, making good use of whatever driver(s) and control facilities for your scanner you've probably installed in your machine from the maker's CD and, like *UniScan*, giving the option to select Microsoft's Windows Imaging Architecture drivers (WIA) .

In *DPScan*, selecting File › Acquire (or pressing ‹F10›) opens a scanner control window.

A typical example is shown on the right but we must stress this is generated by the scanner manufacturer's software, so may well vary between makes and even models. The example panel shown is the relatively simple WIA menu option offered by Epson but a more complex one with more control is available through the File › Select Source menu option (or press ‹Shift–F10›).




To be fair, we should mention there are also other scanning options available under Windows but, unsurprisingly, we feel much more at home scanning via *DPScan*.

That said, there are a few operating quirks with our Epson scanners (both stand-alone and multifunction) and the simple control window shown. So on the off-chance the same sort of thing might also apply to you, here's the details:

• On clicking on Preview, don't be worried if a "Go away, I'm busy" type of error message appears (repeatable). All this means is, if the scanner hasn't been used for 10 mins or so, it's busy snoozing!

• In our case, the first click on Preview is enough to nudge it into wakefulness (activity LED flashing). Wait until the LED stops flashing, click on Preview again and it will then play ball.

(*Note:* With some other Windows scanning applications, there is no error message, so on the face of it you have a failure to scan. But if you wait a while for the LED to stop flashing, as above, everything should then come good.)

• During the Preview scan, the window can be resized if desired by dragging the bottom-right corner to the point that it fills the underlying display. The image in the Preview window automatically expands to fill the window, making select box adjustments much easier.

• On completion of the Preview scan, the select box position and size are set automatically to what the software thinks is appropriate for the previewed image but, if required, the box can be resized and/or repositioned manually in the usual way.



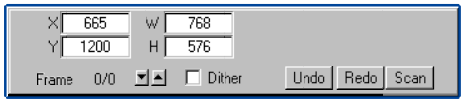


Return to Contents

• *DPScan*'s default main scan resolution setting is only 150dpi which we find is too coarse, especially for monochrome work (e.g. photocopying or OCR). So before clicking on Scan, we first select Custom Settings, set `300dpi` and tweak the Brightness and Contrast sliders as required. The custom settings are auto-saved for future use.

If you also have *OPW*, the latter's Scan applet can control the scanner directly, the control panel(s) and procedures for Select Source, Preview and main Scan being identical to the above description. On completion of the main Scan, the image is imported automatically into the selected *OPW* frame.

• **Image processing** – As you would expect, on completion of the main Scan in *DPScan*, the image is displayed automatically in the frame. This is the exact equivalent of the normal work area of *DPIngScan*, with the same toolbox and info palette.




You can now access all the available image-processing features with which you may already be familiar from using *DPIngScan*, the natural difference being access is via the Windows menu structure rather than clicking the ‹Menu› button.

The *DPScan* Misc menu lists the *Snapper* utility and the Choices sub-menu. The latter is mechanised in a different manner to that in *DPIngScan*; see the Help › Differences file.

[As an aside, if we may add to the "Cropping sprites" saga in Archive 19:2 p36, we find a *very* handy feature in both DPIngScan for RISC OS *and* DPScan for Windows is that, when tweaking the select box prior to cropping, the width and height are shown in the Info palette (W and H).

Or you can simply enter the desired dimensions into the Info palette, press ‹Enter› and the select box will automatically resize itself. The choice of units (pixels/mm/inches/points) is set via the Choices › Page › Units options, in both variants.]

If faced with a simple choice of whether to use *DPScan* or *DPIngScan* for image-processing, the latter may have a slight edge in that, say, when fine-tweaking a select box before cropping an image, the four cursor keys can be used to 'select-nudge' the drag boxes.

(This assumes you have the renowned *IClear* module running – see Archive 17:12 pp37-38 and 18:4 p37. Unfortunately this RISC OS utility doesn't work with *DPScan*.

There is however a neat 're-align' feature added to *DPScan* which, for us, always proves immensely useful. (Well we would say that, wouldn't we? – We thought of it...)

When scanning something, it's very common to finish the main Scan with the resultant image being slightly rotated. This is especially the case with an open book which is often tricky to position correctly.

But this is very easy to fix in *DPScan*: Press ‹Shift–Ctrl› and select-drag a line along a convenient horizontal or vertical reference in the image, such as a line of text, the edge of a picture or (as illustrated below) the horizon. Release the keys and the error is shown in degrees, to two decimal places. Click on OK to accept it and the image is rotated by this angle to remove the error.

As an added bonus, the same process can also be used to correct inadvertent rotation on *any* bitmap imported into *DPScan*, a common example being images from a digital camera which, despite best intentions, often show up one's inability to hold the camera perfectly horizontally or vertically, particularly where a low-resolution LCD viewfinder is used.

Unfortunately, correcting any 'skew' in a pre-existing image imported into *DPScan* will unavoidably result in the complete image being rotated, by the same angle, but cropping off the resulting four triangular bits using *DPScan* will fix that.

• **Saving images**

A significant difference in *DPScan* is that, when you click on File › Save Frame As to save your work, the usual repertoire of Windows image filetypes has two lurking foreigners – Drawfile and Sprite! (The latter is last-but-one in the list so is easily missed.)

These alien (to Windows) filetypes are very handy as, for example, you can scan something, tweak the image in *DPScan* if necessary and then pass it directly to VirtualRPC in a RISC OS-aware format.

However, the transported sprite or drawfile will appear in VirtualRPC as a text file unless you add two more entries in the Hostfs Extensions window, as previously outlined on page 43:

• For sprites, put `ff9` in the Type box and `spr,sprite` (no space) in the Extensions box.

• For drawfiles, put `aff` in the Type box and `aff` in the Extensions box. (That's required to differentiate between the Windows *Draw* extension and the RISC OS *Draw* filetype.)

What's the point of saving in drawfile format? We find it handy if, say, you wish to add captions or arrows to a sprite as they can be applied directly within the RISC OS drawfile.

### Trace for Windows

For all practical purposes, David Pilling's *Trace* works just like its RISC OS counterpart, but converts Windows BMP bitmaps into the improved EMF (enhanced metafile) vector graphics – and does it typically at five times the speed of *Trace* for RISC OS. This is fortunate as although *Trace* for RISC OS certainly produces excellent results, for some unexplained reasons it consistently seems to drag its heels under VirtualRPC (see the performance data on pages 6 and 7).

### Fireworkz for Windows

The Windows port of *Fireworkz* is available for download from *Abacus Training*), although copyright remains with the original authors, Colton Software.

We aren't very familiar with *Fireworkz* ourselves but the variants for both operating systems are reportedly capable and, above all, very user-friendly.

The Windows file extension and the RISC OS filetype are common to both – *fwk* – so documents can be moved easily and directly between the two operating systems, say via the *Exchange* shortcuts.



*Return to Contents*

# 13. Using Windows software

It may seem odd to consider using Windows application software in a booklet devoted to running RISC OS under VirtualRPC, but we've already looked at RISC OS → Windows ports in the previous section and, in some respects, this just follows on directly from those.

And in any event, it's a convenient 'in' to consider what has become a favourite subject for us: the 'Best of Both Worlds', using a combination of both RISC OS *and* Windows applications together to achieve the same ends – but in a generally easier, quicker and far, far more productive manner than using them in isolation. What some people call synergy.

That doesn't mean to say that there will always be advantages as, very often, RISC OS application software is of course the tops and needs no outside help. We wouldn't dream of attempting work under Windows when titles such as *Draw*, *DiagramIt* , *GraphMate*, *TableMate* and *TechWriter* are available to do such a terrific job for our purposes, regardless of the underlying hardware.

But it has to be accepted that there are areas where RISC OS applications simply don't do a particularly good job or, in some cases, don't even go down the roads we wish to take.

As an example, we invite you to have a look at the complex graphic we recently prepared for our local History Group which you will find on their website at *www.pocklingtonhistory.com*

Although the online graphic is at relatively low-resolution for viewing on a monitor (you'll probably need to scroll down to see the bottom half), the high-resolution version was printed and laminated at A1 size and went some way to support winning the Current Archaeology magazine's prestigious, national 'Rescue Project of the Year' competition (2018), with considerable interest from the British Museum.

All this was thanks to the use of Windows applications (for the underlying, high-resolution Ordnance Survey map), the resulting 48Mb sprite then being passed direct from *DPScan* to VirtualRPC, to be overlaid by a mix of graphics and text frames from RISC OS applications: *DiagramIt*, *Draw*, *Ovation Pro* and *PrintPDF*. You know it makes sense...

The sheer beauty of VirtualRPC is that just a couple of key presses – ‹Alt-Enter› – takes you out of RISC OS to the underlying Mac OS or Windows and where, for example, web browsing becomes a great pleasure and not a hit-and-miss affair. Or standard and HD video footage from a camcorder or DVD can be imported, edited and burnt to video DVD.

These are just two examples and there are of course many others. So over the next few pages, we'll aim to cover several Windows applications which can help to streamline Windows-only activities, plus others which will mesh neatly with RISC OS applications to optimise overall productivity. They are in no particular order.

### 'Unknown' Windows filetypes

On page 43, we looked at the HostFS *Extensions* file which ensures that the majority of RISC OS filetypes are retained automatically, plus the *Typefind* utility which can help to identify any unrecognised objects.

An equivalent utility recommended by VirtualAcorn for handling a wide range of unknown Windows filetypes – *Open Freely* – was mentioned in Supplement 13 and does a good job in loading and displaying a wide range of objects with uncommon filetypes where you may not have any suitable Windows applications available.

We have always found *Open Freely* to be invaluable, particularly in accessing the wide range of video and audio filetypes.



### Printing under Windows

We've already discussed at length in Section 9 the whole topic of preparing documents under RISC OS and then printing them using either a RISC OS printer definition file, or *UniPrint*, or alternatively printing them, at very high speed, directly under the Windows printing system.

In all cases – but especially the latter – document rendering and printing speeds are outstanding. Whilst in an extreme case, the RISC OS/Windows combo will sail through jobs such as, say, multi-page/multi-copy full colour documents containing photographs, which (arguably) cannot even be contemplated using contemporary ARM-powered machines.

### RISC OS PDF writers

Using "PDF" in its other guise – portable document format – we also described in Section 9 how a very easy and universal method to get your documents prepared under RISC OS and passed to Windows for printing in a way it can readily understand is to use the PDF facility.

The same goes if we wish to export the document to the Great Outside World for viewing by a customer, for commercial printing, and so on.

Fortunately, we're particularly well served by RISC OS software authors and have a range of alternative commercial and freeware titles for converting our documents to PDF.

This works equally well for ARM-powered machines of course but, for us, the added advantage is that we can so easily pass the resultant PDF file to the host computer and then double-check it for anomalies using a Mac or Windows PDF *reader*, before printing, exporting, etc.

While RISC OS PDF converters are generally excellent, there are the occasional hiccups with, for example, font conversion, and these will be picked up by an external PDF reader.

### Windows PDF readers

For both Macs and PCs, the near-standard PDF reader is Adobe's freeware *Acrobat* which is easy to use and very fast compared with any PDF reader under RISC OS. This is very noticeable where the document contains fonts other than the standard Type1 range and where bitmap characters have to be generated on-the-fly for display. But on a reasonably powerful Mac or PC, this is almost as fast as displaying the document using Type1 outline fonts.

However, there are a couple of potentially adverse features with *Acrobat* in use. First, it's a real heavyweight – even the installer comes in at 33.5MB! So if you'd rather have a 'lite' alternative (for Windows), we can recommend the freeware *Foxit* PDF Reader which weighs in at at 'only' 2.56MB and seems to do everything we're ever likely to need. It's available from *www.foxitsoftware.com/downloads/index.php* so do give it a try.

The second reason (and perhaps because *Acrobat* is available for both Mac OS and Windows), is that *Foxit* has its own font smoothing built in, so doesn't tap into the native Windows facility.

Previously that meant that it had to be user-configured to switch on its font smoothing option. However, contemporary host computers now tend to have graphic cards with enough grunt to provide font smoothing 'on the fly' (the equivalent of the superb 'anti-aliasing' feature in many RISC OS applications), so *Foxit* can now be used as-is.

### Windows PDF writers

Having said that RISC OS gives us a good choice of commercial and freeware PDF writers, you may need or prefer to do the job under Windows (e.g. if the machine you're using doesn't have VirtualRPC).

For this purpose we previously recommended the freeware *CutePDF* which Eric Songhurst advocated in Archive 21:6 p41.



Return to Contents

It can be downloaded direct from the authors or from many others such as *www.filehippo.com* although, as Eric said, you may also need to download/install the open-source *Ghostscript* (for Windows) as well.

The PDF writers are treated as just another 'printer' by Windows in that you can access and set the usual preferences such as paper size, print resolution and so on in the normal way, except that the output is a PDF file, rather than hard copy.

The only other point to revisit is that, if a landscape page in a RISC OS document does not have any outline text on it, PDF *readers* such as *Acrobat* or *FoxIt* may display the page in portrait format instead. That can more often than not be avoided by using the 'dummy text' workaround described on page 57.

However, the good news is that *CutePDF* (and other Windows PDF writers such as *Bullzip*) already take this into account when converting Windows documents with landscape pages into PDF.

### 'PDF Editors'

Shortly after the Second Edition of this booklet was published, some PDF reader authors started to introduce increasingly valuable, additional features to their titles such as *Acrobat* and *Foxit*, for instance adding the ability to highlight text, append 'callout' boxes or add arrows. Of especial note was the feature to introduce hyperlinks to *selected text* (as used throughout this booklet).

The freeware *Foxit* reader soon became (and remains) our favourite application for adding these additional features. All this was discussed in some detail, notably in Supplement 4.

However, it later became clear that other, similar titles (commercial and freeware) were marketing themselves as 'PDF Editors'. This became rather confusing as it was naturally assumed from the titles that (for example), text within a PDF file could be 'edited' in the conventional manner – i.e. by placing the caret within the text and correcting (or otherwise amending) it as required.

It soon became apparent that the authors of these titles were being a tad 'economical with the truth' in that, once text had been embedded and saved within a PDF file, it could *not* then be edited in the normal way (although in principle other methods might be employed to achieve the same ends).

This was discussed in Supplement 13 but, for the moment, it may help to give an overview of what can – and cannot – be done. For example, very soon after the Second Edition and the Supplements were published, our then ISP went down the tubes and we had to transfer to another, with an unavoidable change of email address.

But the 'old' email address was included in several places in the booklet and supplements, and it was 'burnt' into the underlying PDF files. Yet another unfortunate case of Sod's Law, but what could be done about it?

Rather than go back to the source *Publisher* files and amend them – and then go to all the trouble of re-saving them as PDF files, re-activating all the links, and so on, it was much easier to 'edit' the existing PDF files (using *Foxit*) by selecting the Textbox feature, select-dragging a rectangular box over an *old* email address, making both the box and its border opaque (by selecting the background colour to white so that the underlying text was no longer visible), and finally by typing the *new* email address into the (white) text box.

The hyperlink 'target' address needed to be changed from the old email address to the new one, but the text box now overlaying the old address could be copied ‹Ctrl-C› and repeatedly pasted ‹Ctrl-V› over wherever the old email address appeared in both the Second Edition *and* all 13 Supplements, and the resulting, 'edited' PDF files re-saved. So although a tad finicky, a modicum of lateral thinking had solved the problem.

### Splitting and merging PDF files

Although this sub-topic is also covered in detail in Supplement 5, it follows on from the previous discussion so it may help to give a brief overview.

Take for instance a situation where you wish to make significant edits to one or more pages in a PDF file, and which cannot be done realistically in the manner described above.

In that case it may be advisable or even necessary to revert to the original document and make the edits to it. But rather than then having to re-convert the *complete* document back to a PDF file, and to re-activate all the links etc, an easier way is to separate out each page needing to be edited, make the desired changes, re-save it as a PDF page and re-introduce (merge) it into the document.

To do that, we advocate using the freeware *PDFill PDF Tools* to split and re-merge the pages (discussed in more detail in Supplement 13).

### Browsers

There's no need for us to review the perennial problems with RISC OS browsers as the forums continue to include tales of woe from people who simply can't access the pages they want. For VirtualRPC-users, the universal solution is simply to nip across to Mac OS or Windows and do it the (very) easy way. So that's OK then.

*However* – what about the dreaded viruses and malware that leap aboard every time you fire up a Windows browser? From our own experiences, we believe the anti-malware precautions discussed in Section 4 will give you more-than-adequate protection against such nasties, but it must be accepted that the 'standard' browsers – *Internet Explorer* or *Edge* – always have been, and still are, the primary aiming point for the malware brigade.

Yet it must also be appreciated that we don't *have* to use Microsoft's offerings as there's a useful range of alternative and very competent, open-source browsers available to all, the advantage being that (as yet) the malware artists haven't seriously set their sights on them, although of course that may change with time.

So if you feel nervous about using *Internet Explorer* or *Edge* (and we've never knowingly had *any* problems with them over the decades), do have a look at the likes of *Chrome*, *Firefox*, *Netscape*, *Opera* and/or *Safari*, all of which can be downloaded as usual from *www.filehippo.com* and elsewhere.

The use of "and/or" is intentional and signifies that you can have multiple browsers loaded and in use if you wish; as far as we know there's no interaction between them. Indeed, although we used to use *Internet Explorer* with Windows XP, we also had *Chrome* installed as, until Windows 7 or later came along, some sites could not be accessed using *Internet Explorer*, but *Chrome* would do the trick.

### Word alternative

Many people (including ourselves) find that Microsoft's *Word* is simply too big and unwieldy for convenient use, in this context making it that much more difficult for RISC OS *Easi/Techwriter* fans to move their masterpieces to and from Windows.

However, once again, we don't *have* to use *Word* as there are other, highly competent, open-source alternatives for Windows such as *OpenOffice* and *LibreOffice*. Both applications are in fact multi-title suites of semi-integrated database, drawing, formula, Powerpoint-lookalike, spreadsheet and writer applications, the latter being in all respects compatible with the WordDoc format.

We're not familiar with either title in depth but they are impressive and certainly highly thought of by regular users. The initial loading seems to

take a while but, thereafter, are noticeably faster than *Word*. As well as being available direct from the authors, *www.filehippo.com* and others may also have the latest versions.

### Image processing & cataloguing

Being heavily into all things graphic, we find that the topic of image handling and cataloguing is a major must. From the early days, RISC OS was well-served by image-processing programmes, although it was and still is sometimes a case of doing part of a job in one application and finishing off in another, but interchange is rarely a problem.

Everyone will have their favourites but, for us, the long-standing and excellent *DplgScan* plus the splendid *Variations* will between them do almost everything we need.

So it's ironic that, as the years have gone by, effective image *cataloguing* – which once was quite a simple task under RISC OS – has arguably become more and more troublesome. In our experience, there are at least 3 reasons for this insidious deterioration:

• The gradual increase in display resolutions is such that the maximum of 800×600 pixels (SVGA) only a few years ago is now considered 'small screen', and 'HD' widescreen monitors of 1920×1080 pixels, or more, are almost the norm. However, this means that image thumbnails are getting much smaller to view on-screen and are becoming that much more difficult to decipher. Here's a few typical examples:

• In parallel, while once upon a time a 640×480 pixel image from the first-generation digital cameras was considered to be 'high-resolution', today even a 3 megapixel (MP) image is relatively low-resolution, and 6-12MP cameras (including those in tablets and phones) are almost the norm. That's great for image quality – but waiting for RISC OS to generate a directory of thumbnail images 'on-the-fly' can now take a painfully long time. Granted, dedicated image-cataloguers are available for RISC OS but, in our experience, these take a lot of setting up and keeping up to date so, from that point of view, they aren't very productive.

• Following on from the previous point, our favourite thumbnailer for many years was Irlam's *Album* which was easy to set up and keep up to date and, when a directory containing images was opened, the thumbnails were immediately available. But *Album* suffers the same drawback from reducing-sized thumbnails and, in any event, is one of the few applications which will not run under Adjust (on ARM-powered machines or VirtualRPC).

Summarising these three points, it is perhaps inevitable that, over time, thumbnail displays in RISC OS have become relatively poor and rendering them on-the-fly has become increasingly long-winded.

### Bigger and better thumbnails

But fear not, this is where Windows can greatly assist, once again making good use of the fact that any directories full of (say) digital camera images and stored on the VirtualRPC *HardDisc4* are, as far as Windows is concerned, simply standard folders.

So to help with the thumbnailing problem, we strongly recommend an aging but very popular and highly-regarded piece of Windows software: Google's freeware *Picasa* which is available from *http://google.picasa.com* – and of course *www.filehippo.com* and elsewhere.

(Don't worry if, when you download it, it says it's been replaced by something else. Although no longer supported, *Picasa3* still works fine for the purposes discussed below.)



It is easy to install but, when the wizard asks whether you want it to 'look at' images in the whole computer – or just in the *My Documents* folder and sub-folders – *do ensure you choose the latter.* Otherwise, *Picasa* will pick up all sorts of non-essential stuff scattered over the disc and which can take a while to remove from the displays.

Once *Picasa* has completed its scan on being run for the first time, it will have thumbnailed all the images it has found in the *My Documents* folder, but this won't include anything stored elsewhere, namely anything on *HardDisc4*.

No problem: select File › Add Folder to Picasa, navigate to a parent directory containing (say) digital camera images stored on *HardDisc4*, click on the Scan Always button and, finally, click on OK. As you do that, you'll see the greatly-improved image thumbnails being added to the display, at very high speed.

Comparing the clarity of the thumbnails on page 74, produced under RISC OS, with the ones below produced by *Picasa* (both as seen on the identical display), shows there's an enormous improvement.

If all your camera images are contained in the same parent directory, the contents of all *sub*-folders will be thumbnailed and added automatically; if not, add further folders and sub-folders as required.

Thereafter, no manual updating is required as, every time *Picasa* is started, it will quickly re-scan the contents of all selected folders and update the thumbnails for any changes.

Thumbnailing isn't the only thing which Picasa does as there are numerous built-in image-processing features such as red-eye removal and adding lots of effects (some of which may not be available in similar RISC OS programmes). If you do tweak an image and save the changes, the original is retained in a sub-folder, so safety rules apply.

However, if there is a limitation with *Picasa*, it is that it will always re-save the altered image in the JPEG format, thereby increasing or even doubling the 'lossy' effect. So for this reason, we tend to use *Picasa* for image-processing only if the required correction is not available elsewhere. Normally we would use, say, *DPScan* (for Windows) or *DplngScan* (for RISC OS) and then finish off using *Variations*.

But that in no way detracts from the splendid, near-instant thumbnails which *Picasa* provides. However, what about image formats? In essence, anything we have on *HardDisc4* which Windows recognises as a valid format will be accepted by *Picasa*.

So as well as the common or garden JPEGs, this will include lossless bitmap formats such as BMPs and TIFFs. The latter is especially useful for us as it's the native format of our main bitmap application *Studio Pro 24*, and where RISC OS takes an age to produce thumbnails of the large TIF files.

Not surprisingly, *Picasa* will ignore RISC OS-specific file formats such as *Artworks*, drawfiles and sprites. But with the majority of them in (say) clipart folders being relatively small (in KB), it doesn't take long to thumbnail them under RISC OS, on the fly.

### Video editing

Another very popular Windows activity is in taking footage from a camcorder or other video source, editing it to our heart's content and consigning it to (say) video DVD.



Return to Contents

This is where Windows is especially well provided for, with a wide range of video editing application software to suit all requirements. There was even a basic video editing programme called *Movie Maker* bundled with Windows XP which, although not having many of the bells and whistles of the bigger boys, nevertheless was considered to be very capable.

*Movie Maker* is no longer supported but is still freely available for download and will run on later versions of Windows. ***(Health Warning:*** *Be aware that some third-party sources for Movie Maker may trigger a virus message.)*

We covered some of the basics in a series of articles starting in Archive 19:1 p51, but the learning curve is very easy to scale as it's extremely user-friendly, quite intuitive and very much a try-it-and-see program, particularly with the various effects.

*Movie Maker* will also give you a feel for some of the basic video formats you can meet in video-land, including AVI and MPEG.

However, although most editing suites have a DVD-burning capability built in, with *Movie Maker* you had to resort to using a third-party programme to finish off the job.

If you fancy to dig further into the topic, more capable applications with at least some of the extra bells and whistles seem to be mostly commercial, although very occasionally one might crop up on the *www.giveawayoftheday.com* website with a time-limited free licence (more details on page 78).

However, these don't appear to be particularly well regarded, so we got a foot in the door with an early version of Corel's *Video Studio* which has served us very well for many years. Although rather expensive (presently at $64.99), it is nevertheless very good value and can be assessed on a 30-day free trial basis.

### Burning CDs/DVDs

Without wishing in any way to denigrate *CDBurn* which has always done RISC OS users proud, it has to be said and accepted that the task of burning CDs and, especially, DVDs is so much easier and faster if done under Windows.

We routinely use the Windows built-in *Media Player* for ripping audio tracks, sorting out playlists and burning the results to CD, but there are numerous freeware alternatives.

For burning *data* DVDs/CDs, Windows 7 or later has built-in software to do the job. Simply inserting a blank disc into the CD/DVD burner triggers the built-in software. Or if you buy a DVD/CD burner, software is again normally supplied with it (but not if it's been obtained as an OEM component).

For burning *video* DVDs, commercial video editing suites such as *Video Studio* have the facities built in, including a small range of front-end menu-making features.

But if you don't happen to have an editing suite, we use the freeware *StarBurn* which does a very competent job, is compatible with Windows XP or later, and covers virtually all video-format DVDs including BluRay.

### Backing up RISC OS & VirtualRPC

Following on from the mention of the speed and convenience with which we can burn data DVDs under Windows, how about using that for backing up your RISC OS installation?

If you have an ARM-powered machine, a practical method of backing up your software, datafiles and so on is to install a second hard disc drive in the machine and keep a mirror-image of everything on it. This is what we always did with our RiscPCs, and time and again this got us out of trouble. Even if there isn't room for a second, internal drive, a plug-in USB drive could do the job.



With the host PCs and VirtualRPC, we do exactly the same by having a second hard disc drive and using that in exactly the same way, as discussed in Section 6. However, as a double insurance against the loss of datafiles etc (e.g. in case of burglary, as happened to T.O.M.S. some years ago), at intervals we also burn a data DVD containing the *complete* RISC OS installation, i.e. *!Boot*, all application software, all our datafiles, and so on.

That may sound onerous but in fact it's simplicity itself as a bang-up-to-date version of everything just listed is of course already contained within just a single *Windows* folder – called *HardDisc4*!

So it's merely a case of firing up the data DVD burning programme in Windows 7 or later by inserting a blank disc, locating *HardDisc4* under Windows (page 25), drag-and-dropping it into the burn window and clicking on the Burn button. How easy is that?

But you can actually improve on that... Assuming the total contents of *HardDisc4* do not exceed (say) 4.5GB – unlikely? – there's still room on a standard 4.7GB DVD for the rest of a VirtualRPC installation. So instead of burning just *HardDisc4* to DVD, it's just as easy to locate the main *VirtualRPC-xxx* folder and burn a copy of the entire VirtualRPC installation to disc instead.

We do this typically every month or so, normally using a 4.7GB DVD±R disc, although a few pence could be saved by using a DVD±R/W disc. But that may not be so reliable.

The point being that, if something goes badly adrift within your VirtualRPC installation, or even if you're setting up a new machine, you can reconstitute the whole thing from this back-up data DVD simply by copying across the complete works: VirtualRPC-*xxx* and RISC OS (both fully pre-configured), plus all your application software and datafiles.

### Selective back-up (Windows & RISC OS)

R-Comp's *Safestore* back-up programme for RISC OS justifiably earns plaudits from its users and the media, so perhaps it is inevitable that VirtualRPC users ask if there's a similar facility for the host PC, to use in the background and more regularly than the notion of producing back-up discs discussed in the previous sub-topic, perhaps even on a daily basis.

Having tried several Windows titles, only to find that for one reason or another they didn't do what we wanted, we then tried *Karen's Replicator* advocated by Virtual Acorn in Support *Article No. 160*. Although the author Karen Kenworthy died in 2011, her offering is still available for download and works a treat on Windows XP up to Windows 10.

There are more details on pages 38-39 of the *Supplements* booklet but an added bonus is that the application works for both Windows *and* RISC OS. This is thanks to the bonus that the Windows filer simply treats 'RISC OS' directories and files as standard Windows objects, and responds accordingly.

So the only thing we need to do when setting up tasks in *Karen's Replicator* to back up RISC OS directories and files is to 'point' at the same folders and object(s) within the Windows tree.

A point to note is that, with the Edit Settings window open, it's arguably a good idea to tick the Replicate Folder & File deletions? box as this will 'synch' the back-up copy to mirror the original if you delete or rename a file or folder.

The corollary being that, if you leave the box unticked, that will leave all the 'old' files and folders on the back-up medium. On the one hand, that's an insurance if you wish to back-step to recover a previous file or folder but, against that, the back-up medium can overflow with increasingly-outdated objects. The choice is yours.



*Return to Contents*

### Web-site authoring applications

It must be stressed that we have never had to author our own website(s), either for T.O.M.S. business or our personal needs. Thus we have no experience of using any RISC OS web authoring software such as R-Comp's *HTML Studio 6* or similar applications, so apologies in advance should we unwittingly tread on any toes.

However, some years ago, a couple of VirtualRPC users enquired whether we could recommend any (freeware) Windows application which they might use for building their own sites, mentioning that they were looking for something with more 'bells and whistles' than RISC OS titles.

With zero previous experience in which to compare capabilities, we simply did a brief trawl for what might be available in Windows-land, and from ads and reviews soon came across and downloaded the freeware *BlueVoda*.

Despite what is said online about the steep-ish learning curve, we found it *very* easy to get started and, the more we got into it, the more it became clear that anyone familiar with RISC OS 'frame-based' applications such as *Impression* and *Ovation Pro* – plus the concept of introducing text, drawing, bitmaps, etc by drag-and-drop – would generally find it simplicity itself and largely intuitive.

So to prove the point (both for our own interest and to 'sell' to the 2 punters as a potential solution), we took as an example a couple of typical but fairly complex pages in an *Ovation Pro* document, copied and transferred all the data across to Windows (pre-converting things like sprites and drawfiles into their equivalent Window-aware filetypes) and built up a near-identical web page simply by drawing frames where we wanted them and importing the text or objects into each frame by drag-and-drop.

All this proved to be extremely straightforward, primarily because of the similarity with the way that *Impression* or *Ovation Pro* pages are produced. But then we could also easily introduce hyperlinks within the text, and even relatively esoteric added-extras such as audio or video clips, or animated GIFs, simply by adding a relevant icon frame for each feature and drag-and-dropping the source files over each one.

Testing whether the HTML works or not is simply a case of clicking on the 'Publish' icon on the button bar (under Tools), which produces the HTML on-the-fly and opens the web page in your default Windows browser. Any hiccups can quickly be spotted and a brief delve into the *Help* files will most likely resolve any problems.

At this level, it would be difficult to criticise *BlueVoda* so it can confidently be suggested as a means of getting a website up and running, although prior familiarity with RISC OS frame-based applications will greatly assist.

Having got that far, we even wondered whether or not the text could be sub-divided into multiple columns (as in the Ovation Pro original). This proved to be easier than expected in that, having 'drawn' the first column frame, it could then be copied and pasted into place as required, all ready for text to be imported by drag-and-drop. The only limitations (as far as we could tell) were that the text frames could not be linked and that, although they displayed perfectly in Windows browsers, RISC OS browsers would not play ball.

### Applications included in the Supplements

Several other potentially-valuable Windows and RISC OS applications were discussed in the *Supplements booklet* to Edition 2. So rather than repeat the details in this Edition 3, here are some cross-references for you:

Supp. 1 (page 2) – *K-lite* (audio/video codecs).
Supp. 1 (page 3) – Thumbnailing PDF pages.
Supp. 1 (Page 4) – *OCR* for Windows.
Supp 4. (page 13) – *PrintPDF*'s 'Queue' feature.
Supp. 4 (page 15) – *PrintPDF*'s 'Bookmark' feature.
Supp. 4 (page 16) – *Foxit*'s 'smart PDFs' features.
Supp. 5 (page 19) – *PDFill* tools (PDF handling).
Supp. 6 (page 22) – *DeskInst* (save/restore desktop).
Supp. 6 (page 23) – *CloseAll* (closing multiple windows).
Supp. 9 (page 33) – *Panorama* (large-page scanning and 'stitching').
Supp. 11 (page 41) – Oak Draw (NB ***NOT*** 64-bit).

### Windows freeware

We keep referring to freeware and, apart from downloading programmes direct from the authors' websites, many of the titles discussed are also available from *www.filehippo.com*.

But it's worth mentioning that some Windows *commercial* software is occasionally available, either entirely free of charge or, failing that, with a free licence for significantly longer than the usual 30 days trial period.

The word "occasionally" refers to an interesting site called 'Giveaway of the Day'. Specifically, the site makes a piece of commercial software available, either for free or with an extended license, but on a daily basis. The only condition is that you need to download, install *and activate* the software before the day runs out.

There's no support or upgrade paths for the otherwise commercial software titles but, if you have a look at the site on a daily basis, and are selective, some absolute nuggets can be obtained – entirely free of charge.

Sometimes even upgraded versions come up however, replacing titles which may have been offered only months previously. So do have a (daily) look at *www.giveawayoftheday.com* to take advantage of the offerings.

### 'Bloatware'

A final source of potentially-useful Windows software is often disparaged by the 'knockers' of the RISC OS community who are critical of what they see as the rubbishy bloatware, typically bundled with a main application.

We don't generally go along with this as, on occasions, we find there can be some really useful bits of software in there. Just one typical example is that our Epson printer(s) came with a superb label-printing utility for printable discs, giving an extremely elegant touch to CDs and DVDs, without recourse to sticky labels or felt-tip pens.

And more often than not, added extras can be declined during the installation process. But if you find something has been installed inadvertently – or without asking – it can easily be uninstalled and binned using *CCleaner*'s Tools-Uninstall feature (page 18).

### Beware "Free download!"

There are two seemingly free offers to beware of however. First, an advertised "Free download!" often means just that; you can freely download the (commercial) software – but you then need to buy it to use it... How silly.

### Beware 'scareware'

The other dodgy area – commonly called 'scareware' – is reprehensible and frankly might better be termed 'scamware'.

Typically, an advertisement will offer a "Free Scan!" of your Registry. Then when you run the scan, it'll scare you silly with a list of hundreds of Registry 'issues' – all very likely fabricated... Then it offers to fix them for you, often with a "Free download!" offer on the software. But *then* it attempts to charge you to use it... Geddit? **AVOID...**

If you have any residual concerns about your Registry, do consider using *CCleaner* to keep it in good order (page 18).





*Return to Contents*

# 14. Sundry Windows ‘issues’

Having again mentioned those unfortunate ‘knockers’, and having introduced the problem in Section 4, in this final main section we’ll have a look at some of the other aspects of Windows which they do tend go on about and, once again, put things into a sensible and far more realistic perspective.

## System restore

The first topic refers to their criticism of Windows as being very unreliable and clunky; and when things do go wrong, it’s very often a huge task to put things right. But in our experience, that simply isn’t the case and, indeed, generally we find it *very* easy to recover from most problems, such as they are.

The main reason for this is that all versions of Windows have an invaluable, built-in feature called System Restore which, frankly, is sorely missed in RISC OS. But there are various knock-on effects which some people (perhaps because they don’t appreciate the underlying cause) quite needlessly and unfairly consider as being a Bad Thing. So we’ll first have a look at this very positive Windows feature as it really is a great helpmeet and timesaver.

In RISC OS, if you take an action such as installing some new hardware and/or software, and hit problems, it’s very often difficult to step back to where you were. A commonplace predicament is that, especially if the software has an auto-installer, you’re never really sure what’s been added. Or where. And there’s very rarely an Uninstall function.

So if you hit post-installation difficulties, very often you have to rummage through not only the most likely application directory, but also in the main *!Boot* sub-folders such as *Tasks* or *Resources* (especially *!System*), in an attempt to find what files have been added, and to remove or disable them.

And a not uncommon hiccup is where an updated version of a module may well be the culprit. But how do you revert to the previous version if it’s already been overwritten by the updated one? Cue *very* unhappy memories of the ‘wrong’ version(s) of *CDFSSoftATAPI*, which could even stop the machine starting up.

In Windows, this sort of potentially severe problem is routinely much more easily resolved. First, more often than not, the software you’ve just installed will have its own Uninstall function which, in theory, should remove all traces of it, including the support modules and Registry records.

But if that doesn’t quite work, there’s a universal Uninstall feature, accessible (in Windows 10) by pressing Start › Control Panel › Programs and Features and selecting the program – sorry, programme – you wish to uninstall. Alternatively you can use *CCleaner*’s Tools-Uninstall feature (page 18). Sometimes these facilities are more effective than using the individual programme’s feature.

But if in the unlikely event that also fails and you’re still left with troublesome symptoms, invariably you can recover the situation in Windows by using the *System Restore* feature – even if Windows will not start up normally.

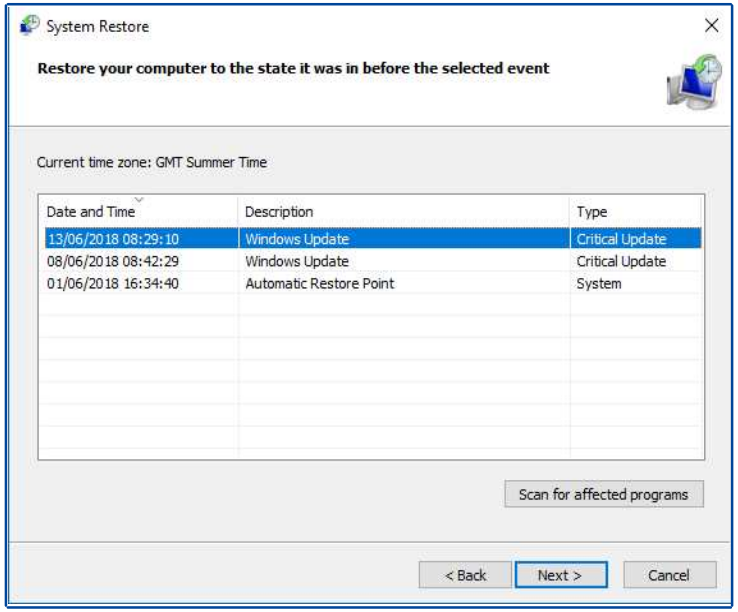
Finding *System Restore* varies depending on the version of Windows installed but, in Windows 10, click on Start › Control Panel › System and select System Protection in the left pane which opens the System Properties window. Then click on System Restore to start the process.

If you’re new to this invaluable get-out-of-trouble feature, you might like to click on the System Restore overview menu option and read up on what it can do for you and how to drive it.



At an early stage, you need to select how far back in time you want to go, usually the point immediately prior to the event which is thought to be causing grief. So for example, if you’ve hit problems following a recent Windows update, and wish to go back to where you were, select the relevant installation event and continue.

In the Windows XP version of System Restore, activities such as installing or uninstalling a piece of software were then listed in the System Restore window illustrated, giving even more flexibility on which event to highlight. This no longer seems to happen in Windows 10, but it still allows you to use System Restore as explained.




We cannot over-state just how valuable the Windows System Restore feature can be. It’s very reassuring to know that, if all else fails, you can use it as a last resort to regain control of the machine, even if Windows fails to start up properly. (See also Supplement 10 which deals with starting up Windows in a ‘safe’ condition.)

If all goes well, you’ll be put back into a trouble-free condition. But if not, you can just as easily step backwards in time by two or even more restore points, maybe pre-recorded over a period of days or weeks, until you get to an operating state you’re entirely happy with.

[*NB*: In that case, the earlier anti-virus and anti-spyware signatures may be outdated, so a quick check is needed to see if more recent updates need to be (re-)installed.]

Note that Windows saves a ‘Restore Point’, automatically, at intervals (01 Jun 2018 in the example above), adding to the range of available points where you can perform a System Restore.

## Start-up time

Let’s face it, your typical off-the-shelf PC used to take significantly longer to start up from a power-off condition than any contemporary ARM-powered machine. We’re talking an absolute minimum of 30-45 seconds and, if Windows was fully loaded up, those times could very easily run to 1-2 minutes.

There might be various reasons for that; for example, the PC may run some additional integrity checks on things like the graphics card and, if the Windows anti-malware is configured to do so, it may run a virus scan.

However, it’s very noticeable that with the more recent, relatively powerful host PCs (with ‘Core i’-series processor clock speeds running at 3GHz or more), start-up times have been very markely reduced to perhaps only 10-12 secs.



Return to Contents

Starting up VirtualRPC needs to be added to that, but now that also takes far less time than hitherto for the full RISC OS desktop to be displayed, with your usual choice of applications running, and is on a par with the same RISC OS start-up scenario on the latest ARM-powered machines.

But for *all* users, it's still necessary to configure your anti-malware suite to ensure that it doesn't drag things out by doing a full virus scan at start-up. The procedure will vary with the anti-malware programme(s) installed, so have a look at the relevant *Help* file.

Nevertheless, the time from starting the host PC from power off to being able to use RISC OS may still feel a tad slow-ish..

However, this is entirely to miss (or to disregard) the elegant and valuable power options which are available to us under Windows and which, when brought into play, make an enormous difference. These are the Hibernate and Standby features, accessible via the Start › Control Panel › Power Options menus.

When configured to do so, the PC will go into Standby mode automatically if it hasn't been used for your pre-selected time, or if you select it manually by pressing a hot key combination, or (if present) the 'Sleep' key. Windows first notes which applications are running and which windows are open, before putting the PC into Standby mode. Typically this takes only 2-5 secs.

Like a TV, the machine can then be left in this condition for as long as you like: minutes, hours, overnight or even days. But pressing the Power button will bring it out of Standby and into a fully-operative condition – with the same apps running and windows open – in a matter of only 2-10 secs, depending on the machine and how busy Windows was before going into Standby mode. In principle you can repeat this process as often as you like.

Interestingly, all this also applies to RISC OS, even if running full-screen under VirtualRPC. So you can drop it into Standby mode in 2-5 secs and later revert it to a fully operational condition, with the same RISC OS apps running and windows open, in only 2-10 seconds.

The Hibernate function is identical in principle to Standby except that, before powering down, it first saves any unsaved documents and then, on re-activating, returns them to the desktop. This additional process adds just a few seconds to the power-down and re-start times.

The other main difference with Hibernate is that residual power consumption is rather less than in Standby mode, so is advantageous to laptop and tablet users. But even if a laptop or tablet is left running in Standby mode, on battery power, you'll have typically at least 7-10 days (often much longer) before you need to recharge the battery.

Putting all that into perspective, powering up a PC (plus VirtualRPC) from a cold start is admittedly time-consuming. However, the availability of the Standby and Hibernate features means it can then be powered down, and later returned to a fully-operating condition, in a matter of moments – and this can be repeated as many times as you wish.

### Hard disc defragmentation

Defragmentation of individual files over a hard disc drive surface does not happen under RISC OS and, undoubtedly, this is a Good Thing. So naturally, the topic of file defragmentation under Windows is a favourite source of horror stories.

Yet again, this must be put into proper perspective and, when looked at in detail, we find most of the reported problems may even be self-inflicted. First, how often do you need to defragment your hard disc drive? Self-evidently, this depends very much on its contents and on how much it's used.

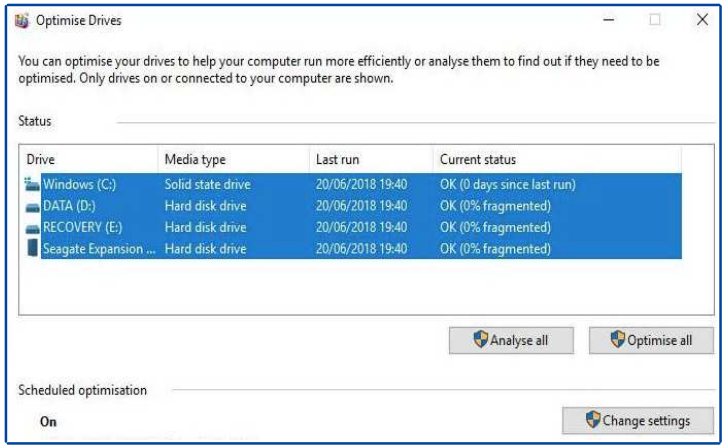


In general, T.O.M.S. found the main drives in their most heavily-used desktop machines (running Windows XP) needed to be defragmented every 12-15 months or so and none of the back-up drives needed to be done for over 3 years.

Even with a drive which is getting to the point of requiring defragmentation, this doesn't normally impact on operation. You may notice a little more drive activity in day-to-day use (e.g. loading or saving a large file which may be fragmented over different parts of the disc) but performance is barely affected, especially if the drive is of the modern SATA variant.

However, for Windows 7 or later, the built-in defragmentation programme is considered to be a significant improvement on the early Windows XP version, and the impression seems to be that you can leave (say) Windows 10 to look after things, in the background and without user-intervention.

But if you wish to double-check your disc(s), or do a manual defragmentation, the procedure is to click on Start and type `defrag` into the window. As you do that (no need to press ‹Enter›), several options are listed, including Defragment and Optimise Dri[ve]. Click on this option to open the Optimise Drives window.




You can now either select any one disc drive and click on Optimise to defragment it or, if you wish, select the first one, press ‹Shift› and click on the last line to select *all* the discs, and finally click on Optimise all to cover all connected drives (including plug-in media). The programme will leap into action and defragment all the selected discs, as a background operation.

Note that, by default, the Window 7 or later defragmentation programme runs every week but, if you find it to be intrusive (on, say, a lesser PC), you can click on Change settings and amend the schedule as desired.

For those of you running Windows XP, the built-in defragmentation programme isn't so good and, in the Second Edition, a much better, commercial programme (*Diskeeper*) was suggested. However, Auslogics' *Disk Defrag* is a recommended, freeware alternative .

### Memory requirements

A favourite way for the knockers to knock a Windows-based machine used to be to scoff at the amount of memory (RAM) it needed to operate effectively. But when we looked into the facts of RAM requirements, they revealed considerable misconceptions which undoubtedly – and quite unfairly – led to misrepresentations when they appeared in print or in forums.

In the earliest days of VirtualRPC, it was very occasionally necessary to ensure that the on-board RAM of the then modest host PCs was at least double the amount allocated to RISC OS. In other words, to ensure VirtualRPC worked with 256MB RAM, the host PC needed a minimum of 512MB.



Return to Contents

However, PCs were very soon fitted with 1-2MB RAM to support the requirements of the later versions of Windows (Vista onwards), plus Windows 7 incorporating the 'AERO Experience' which not only required an increasingly-powerful graphics card but also an additional 1MB RAM.

The other general reason why host PCs require far, far more RAM than their ARM-powered stablemates is to adequately support the inherent needs of whatever application software is to be run on the computer, and which in itself may require oodles of RAM if performance is not to be compromised.

A typical and very popular example is the wide choice of video editing suites available for Windows as discussed on pages 75-76. It has to be appreciated that video footage – in any format – can result in *very* large datafiles; 1-2GB is by no means unusual.

But in order to edit video footage, we first need to load the file(s) to be massaged into memory and, if we are not to rely heavily on virtual memory, the more RAM we have installed, the easier it will be for the machine to cope.

So today, it is quite normal for 4GB of installed RAM to be considered as the working minimum to support such productivity software, in addition to that required by the Windows 10 operating system; and it is not at all unusual for higher-spec host PCs to have 8MB or even 16MB of installed RAM.

To put all this into proper perspective, we should observe that RISC OS and its application software simply do not begin to provide anything like the same productivity and memory-hungry features of some Windows applications. If they did, then RISC OS would of course not only require but benefit equally from these *very* large doses of RAM.

### Anti-aliasing (not...)

Fact: The RISC OS font manager has always had the best outline font antialiasing system in the business.

As far as Windows used to be concerned, the Second Edition went into considerable detail on what actions were necessary to configure the equivalent Windows screen fonts 'smoothing' system to even approach the very high standards set by RISC OS.

Granted, some Windows 'power' application software such as *Photoshop* had and still have true antialiasing built-in (in the same way that *Artworks* for RISC OS can provide antialiasing). But as PC graphics cards and even built-in graphics have become more powerful over the years, it is now quite usual to find that universal antialiasing is the norm, without the need for user-configuration.

(As an aside following on from the previous sub-topic, graphics cards will often now have their own dedicated on-board RAM, typically starting at 1GB and even up to 4GB for the more powerful cards featuring full transparency for windows etc. For unexplained reasons, this sometimes feeds through to RISC OS running VirtualRPC, but to spot transparency on overlapping 'cards' in a desktop game of *Patience* is – to say the least – quite bizarre!)

### Odds & Ends

The preceding sub-headings probably cover the more troublesome issues which the knockers do rather tend to go on about – but of course there are numerous smaller ones which crop up on occasions.

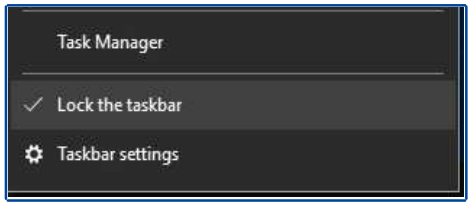
Once again, it's best to avoid your own view of Windows being swayed by them as, routinely, the points of criticism are more often than not based on unfamiliarity with Windows, and not actual shortfalls, so a touch of RTFM and a few mouse clicks will invariably correct these misconceptions.



Here's a couple to be going on with:

• **(Un)locking the taskbar** – Someone complained it was possible to drag the Windows taskbar (a.k.a. the 'system tray') out of position and what a nuisance it was having to reposition it each time. In fact this can happen only if you first unlock the taskbar, allowing it to be repositioned at the left or right side, or at the top or bottom of the display, to suit your personal preference.

Furthermore, it's virtually impossible to unlock it by accident... But to (re-)lock it, simply right-click over an unused bit of the taskbar and click on Lock the taskbar.




• **Tidying the taskbar** – The Windows taskbar is also often criticised for filling up with all manner of mini-icons for applications, (whether or not the programmes are actually running), making the taskbar difficult to decipher. Agreed; and the Second Edition explained the simple procedure for Windows XP to show each icon, or to hide it, to suit your personal preferences.

And should you happen to wish to access a currently 'hidden' icon, clicking on the < (Show) icon will reveal all of those hidden.

The equivalent procedure for Windows 10 is, first, to right-click over an unused bit of the system tray and select the last entry in the sub-menu (Taskbar settings, shown above). The Taskbar window appears.

Now scroll down to the Notification area sub-heading and click on Select which icons appear on the taskbar. The window now lists *all* the application icons, each with an On-Off slide switch. Make your choices and close the window.

Your choice of icons to be displayed should now run along the task bar – with the remainder hidden within the '^' icon (Show hidden icons) at the left end of the line of icons.

A simpler method of hiding a particular icon is to select-drag it over the ^ icon to 'hide' it. Conversely, if you wish to return a particular icon to the system tray, click on the ^ icon, locate the icon you want and drag it onto the system tray, alongside the existing icons.

The order in which the displayed icons are shown in the task bar can be amended as you prefer, again by drag-and-drop – but see the comment on page 30 about doing this when using an extra-high-resolution monitor ('2.5K' or higher). The task bar icons are then so small that moving them around can be a tad tricky.



Return to Contents

# 15. And in conclusion...

Not long before VirtualRPC was released in 2003, the four of us then at T.O.M.S. had as good as decided to leave the world of RISC OS altogether and transfer to Windows. This was because our aging RiscPCs were becoming increasingly unreliable, while the many software incompatibilities with Iyonix, and its inherent hardware limitations, rendered its use as a business tool impractical.

In the event, David Holden suggested we hung on for a couple of weeks as 'something' was about to be announced. Bang on time, VirtualRPC-SE appeared. (Thank you David! RIP.)

Despite the initial hiccups, which were soon cleared, VirtualRPC was a *very* exciting development for us as, in principle, it was immediately apparent that we had a satisfactory long-term solution to our problems. Nothing changes those early views; indeed, in the intervening years since 2003, they have simply been reinforced.

It will be self-evident that we 4 were (and 2 of us still are) *totally* sold on the principle of running our favourite operating system and application software via the medium of VirtualRPC, running on multiple host PCs.

The near-total compatibility with our RISC OS application software, the reliability and usability of VirtualRPC, the splendid high-resolution displays matching the standards of ARM-powered solutions, the reliability and sheer 'oomph' of contemporary multi-core processor PCs – and even the much-maligned (by some) Windows and its application software – all integrating so very well together, make this a simply outstanding 'best of both worlds' combination.

That isn't to say there are no difficulties at all in setting up and running VirtualRPC and the host PC but, when set against the undoubted and innumerable practical advantages, they simply pale into insignificance.

However, there are perhaps three important rules to bear in mind:

• If you're contemplating joining the many thousands of registered VirtualRPC owners, do not allow yourself to be dissuaded by the anti-PC and anti-Windows 'knockers'. Hopefully we've shown in Sections 4 and 14 that their criticism is – invariably – highly-exaggerated scaremongering, with little basis in fact, *provided* we take precautions to protect ourselves from the perceived risks.

• To that end, we cannot expect to fire up the host PC, start up VirtualRPC and spend a session using RISC OS applications, without regard for what might be happening 'under the bonnet'. When setting up the host PC in the first place, it is essential that we install and configure an anti-malware suite of programmes along the lines suggested in Section 4 (or alternatively, obtain a PC with it pre-installed and pre-configured).

• Similarly, it is essential that we ensure the anti-malware programme signature files are kept up to date (page 16). This is in no way onerous as the majority of titles routinely conduct daily updates, in the background.

To summarise: Do enjoy your computing – but keep it safe.

**Acknowledgements**

We mustn't stop this blether without once again offering a huge vote of thanks to VirtualAcorn for providing such an outstanding application, backed up by their splendid support. This has kept us – and we know many others – so well served where otherwise we would have left the RISC OS community years ago. Long may it continue!