

CLANN

May 98

NEWSLETTER

JAVA 1.0.2 Compliant

Exclusive
interview
Stephen
Streater

Preview
the RPC II
at the
Wakefield
Show

**Its time to
operate**

Take a look at the Unix
operating systems

Catch up
on the
Acorn
South
West Show

**What's in a
name?**
Your chance
to be
involved with
Acorn's
forthcoming RPC II

Find out
the latest on
Acorn URL
handler

CONTENTS

ACORN NEWS

The strengthening of the dealer channel in preparation for future product launches.

First shipments of !Browse & RisCafe.

Acorn World 98 Update

NEWS

PAGE
3

ACORN GOES SOUTH WEST

If you didn't make it, find out what went on at the last Acorn show



PAGE
5

HAVING WORDS WITH...

Find out the latest secrets from a special Clan Member.



PAGE
6

OPERATION ACORN

Interested in UNIX?
The two operating systems are featured



PAGE
8

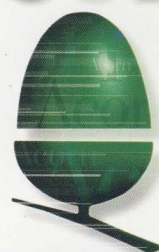
UNDERGROUND

An in depth look at the URL handling of the Acorn's latest Browser



PAGE
10

Acorn



Please note that all Clan enquiries should now be directed to Acorn on

email
clan@acorn.com

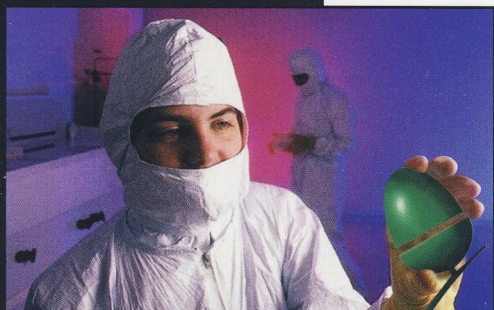
Phone
01223 725925

Mail
Clan Acorn
645 Newmarket Road
Cambridge
CB5 8PB

The clan magazine is a quarterly publication to clan members and developers in the Acorn Community

Contributors include Neil Carson, Dave Walker, Kerri Davies & Arthur King

Designed and produced by Creations
01438 747446



News

Acorn Group and Citrix Sign Licensing Agreement for ICA Protocol®

A definitive licensing agreement with Citrix Systems, for Citrix's Independent Computing Architecture (ICA®) has recently been signed. Under the agreement, Acorn will integrate Citrix's ICA protocol, the emerging industry standard for thin-client/server computing, with all future NC operating systems and all appropriate Internet appliance products. Technology from Acorn is now capable of being integrated into enterprises running almost any network platform.



Acorn's relationship with Citrix supports both companies' vision of reducing the costs associated with computer and system management.

Acorn has also outlined its intention to create a next generation, inexpensive thin-client device that is fully optimised for Citrix's ICA thin-client/server architecture. Using Citrix's ICA technology, Acorn's thin client products will extend application access while offering seamless integration into existing mixed operating system environments, from legacy mainframe systems to Microsoft® Windows NT® networks and ntranets.

Acorn thin clients running the Citrix ICA protocol will offer the full functionality and performance of a networked Windows-based PC without the need for local processing. This enables users to access 32-bit Windows-based applications executing 100% on Citrix WinFrame® servers, and offers a single-point control to MIS managers for deploying, managing and supporting applications and optimum performance for end-users.

This economical solution is ideal not only for the corporate enterprise market that is looking to lower total cost of ownership and system maintenance, but also to other sectors such as the education market which are often bound by budget constraints.

RPC II at Wakefield 98

Making its debut at Wakefield this year will be the long-awaited Risc PC II. Code-named Phoebe, the new machine will be exclusively on show in a preview room built chiefly for the demonstration of the prototype.



Throughout the show the room will be open only to WACG and Clan. In addition, a number of Phoebe engineers will be on hand to answer any technical questions and provide a feedback on the specification of RPCII. The demonstration also sets out the history of the Phoebe's development.

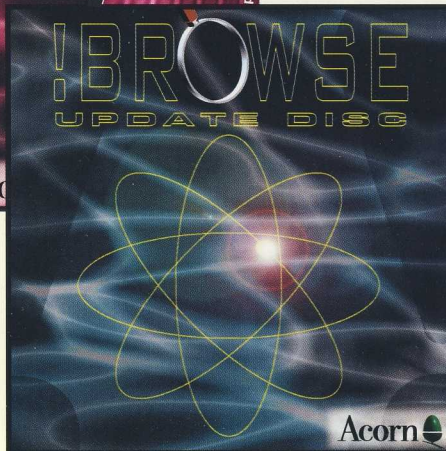
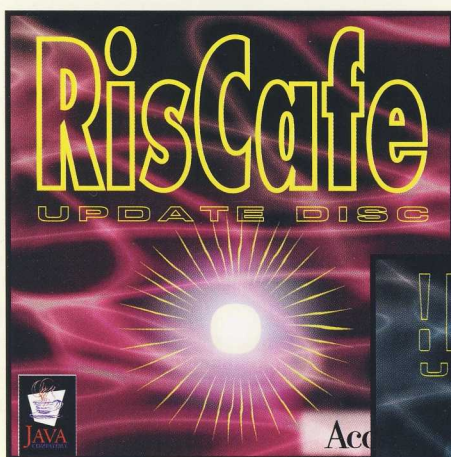
The event takes place at Thorne Park Athletic Stadium, Horbury Road, Wakefield, on 16 May and 17 May 98. Opening at 10 am on both days. Advance tickets can be obtained from WACG Show, 95 Cumbrian Way, Lupset Park, Wakefield WF2 8JT. Cheques are to be made payable to WACG Show.

Update on Discs

After a successful launch of the beta CD originally launched at Acorn World 97, updated versions of !Browse and RiscCafe are now available from your local dealer.

Both RiscCafe and !Browse have been significantly enhanced and contain several updates including full Java 1.0.2 approval, improved Browser Toolbox and on line documentation. The updates authorise the user to continue using the product after the official termination date and enables access to future updates upon release.

More information is available from your local dealer.



NEWS

Power to the People

With another successful year over, the fortune of the Acorn platform now looks even more promising. Alongside a rapid employment of additional marketing and sales personnel, Acorn has restructured its dealer channel to improve the flow of information through to new and existing Acorn users.

Whilst supporting the dealer channel with sales tools and materials, the restructuring also empowers dealers to communicate and promote the platform into areas Acorn has not previously targeted. In addition to providing value added marketing campaigns, Acorn will work closely with a number of dealers to adopt a network community of opportunity.

Other developments include a contract with *Acorn User* magazine to bring a range of promotions, special offers and features over the next year. The strengthening of the relationships with dealers who regularly advertise in *Acorn User* has also been a major breakthrough in enhancing the overall Acorn industry.

— ACORNUSER —

Revelation Demo Party

The 2nd official Acorn demo party will be taking place on the Saturday at Acorn World 1998.

The party will be a chance to meet the demo crews, talk coding and design, and check out all the latest releases running on everything from A3010s to StrongARM Risc PCs. The highlight of the party will be the demo competition taking place at 2.30pm in the Acorn World seminar theatre, to discover the best demo of 1998. The top prize is a Risc PC II computer system kindly donated by Acorn, along with other prizes for runners up.

How to submitting your entry to the competition

The names of the group and/or person who created the submission should be clearly marked on the discs, as well as in a help file on the submitted disks.

Save your entry onto one or more floppy disks. (No archiving)
Write your name on the disc label. Include a covering letter:
i). Name ii). Address iii). Phone number iv). Email address

All entries for the competition MUST be received no later than the end of the Thursday before the show. Entries should be sent to the following address:

Revelation Demo Party, 160 Heronswood Road, Welwyn Garden City, Herts, England, AL7 3EU.

Proposed floatation of ARM Holdings PLC

Acorn Group PLC has announced that its 38% associate, ARM Holdings PLC, is now listed on the London Stock Exchange and quotation for American Depositary Shares on the Nasdaq National market in the United States.



Established in 1990 from Acorn's own microprocessor group, ARM designs and licenses high-performance, low-cost, power-efficient RISC microprocessors, and related technology and software. Other corporate shareholders in ARM are Apple Computer (38%), VLSI Technology (6%) and Nippon Investment and Finance (6%). In the year to 31 December 1997, ARM grew its revenues by 59% to £26.6m and posted pre-tax profits of £5.0m. ARM's net assets as at 31 December 1997 were £12.5m.

Filings submitted to the LSE and the SEC have initial filing price of £5.75 per share. All corporate shareholders are proposing in the offering to sell 16.7% of their existing holdings and to grant the underwriters an option to purchase up to an additional 2.2% to satisfy market demand. Acorn currently holds 15m shares.

Prizes

The winner will become the proud owner of a Risc PC II computer system. 2nd place contestant will receive a year's subscription to *Acorn User* magazine.

Demo Competition rules

Must run on:

- SA Risc PC, <2Mb VRAM
- Must be StrongARM compatible
- Must be previously unreleased
- No interaction



All screen modes used must be accompanied by mode definitions supplied with the demo. Demo must start with a simple double click from the desktop and must exit cleanly to the desktop after running. Max size - must come on no more than 4 HD disks.

Playing time - max 5 minutes. No more than one entry per group/person. In the case of disagreement, the adjudicator's decision is final. All entries are available for use by Acorn for their CD compilation. ■

Stop Press.....

This year's Acorn World 98 has now been scheduled for 16th - 18th October 98 at London Arena, full details will be announced in the next issue of *Acorn User*.



Acorn goes...

SOUTHWEST



Acorn Southwest Show described by an Acorn enthusiast

The Acorn Southwest Show took place on Saturday, 7 February 1998, at The Webbington Hotel, Loxton, North Somerset.

What a lovely location! Much bigger than I'd expected from the motorway view, and bigger in the exhibition hall. A lot happening, still plenty of room to move around. I'm a long time Acorn enthusiast and have run Acorns in education for years, and so I wandered round with considerable interest.



Here are some observations as I walked around the Show.

TAG HyperStudio - I use this on both Acorn and PC platforms. They had excellent demos. I like the new animated path cartoon objects and they

were able to give me useful technical answers to some cross-platform questions. A good buy for a school, as we're doing a big CD-ROM project this way. Useful for staff and pupil use. (I was wandering round taking piccies with the Mavica digital camera as I went. Ideal for hypermedia projects, etc. I used it for the Hyperstudio project to photograph ALL the pupils!)

The game 'Doom' sold by R-Comp Interactive was running native on Acorn. Whatever next? Looked fun, though...

Acorn's latest Archimedes World magazine featuring the eMate. Lots of interesting articles...

Xemplar were plugging the eMate. A number of schools are trialling them and my junior colleagues may adopt a class set. It's great for robust entry of text and numbers. Good connectivity. However, I'm a bit concerned it doesn't fit well to an Intranet/Internet future and is quite expensive.

Speech software and Pluto Newsreader by Jonathan Duddington looks a huge improvement, as I use Termite. It's on my 'must get' list.

Impact Database by Circle Software. I've wanted to know more and found lots of help here.

APDL had collections of CD-ROMs, PD fonts, etc. Excellent prices on Zip 230Mb drives and software, Atapi + IDE interfaces, A3020 120Mb +IDE interfaces.

Raspssoft showed off their invoicing software. Could be interesting as a possible class activity.

Sound Venture had an add-on for TextEase. It was interesting talking to Peter about linking Sonar and his 'Talking Books' projects based on these programs. Now know what 'turdus' is.

Serious Statistical Software showed off school performance figures analysis. Great package for statistics.

iSV Products had eMates, special CD prices, etc. I was most interested in DrawWorks.

Yellowstone had fluorescent everything! T-shirts included. Mozart software caught my eye and there were very cheap 'Big Foot' drives too.

Sherston were doing a great job promoting educational software. Their catalogue shows how well Acorn allows software to target the British education scene.

Electronic Font Foundry showed off their fonts, keyboard layouts etc. A tested and trusted supporter of Acorn's sophisticated use of DTP. ExpLAN showed off their Bible software, which is highly relevant to schools; especially church-based ones like mine.

CJE had cheap big IDE drives, cheap RISC PC 32M memory, Tekkie CD, CJE 133 'fastest PC card'. Nice. Nice...

The Casio digital cameras demonstrated by IMS address a slightly different market to my Mavica. Good targetting in terms of what schools may be able to afford. Gave good results, resolution, colour rendering. I'm very tempted. Also Pocket books, Kiyeko CD adaptation, etc. Lot of things of interest.

Argo Interactive had a show offer to get online including a 33.6k modem and one month online, including free headphones, etc.

I Liked Langdale project by Creative Curriculum Software (lived there, done that...), and Magic Maths, Conjuguez, etc.

Werewolf Software - RAMplify looks useful to freeze programs and to free memory. Just the ticket if you need it, if you see what I mean.

I finally tracked down ArcSimp by Mijas. Best thing since sliced bread if you remember analogue computers or differential equations.

Castle Technology look like they are bringing out a CD writer which will be very welcome (one of my PC things I'd hate to be without). Xemplar/Acorn. Good to see prototype Risc PC II, eMate and other goodies. Use of Acorn machines as thin clients via Citrix Winframe to NT Windows servers is very interesting for schools being pressurised towards the Wintel side. Good to see Mike Gilbert at his perky best, if hoarse after a busy day. Keep up the support in the IT education newsgroup, Mike!

Not forgetting the BARUG stand, manned all day and meeting lots of enthusiasts old and new, and generally spreading the message of Acorn's power and sheer efficiency.

Oh, and Micro Laser Design. Nice stand, very helpful. Supply lots of good kit and advice, not forgetting an A3 colour laser printing service. They did my school magazine from Ovation last year at an excellent price. Local and worth supporting.

The lecture area was comfortable, with good projector facilities. The main concourse gave room to circulate even with hands full of goodies, carrier bags, etc. I'd come again another year...

John Fisher, ICT Co-ordinator, Taunton School, Taunton, Somerset TA26AD

If you'd like to see some photographs of the event, you can access them at <http://www.argonet.co.uk/acornshow/swest>. ■



Having words with...

Steve Streater
Eidos PLC

By Arthur King

Stephen Streater is no stranger to the Acorn community. His company, Eidos, started off in 1989, developing digital video compression technology using Acorn Archimedes A540 workstations. This was originally applied to a professional off-line video editing package, called Optima. A simpler version, called Eidoscope, was sold to Acorn users. Digital video is now a central feature of the best computer games and Eidos was quick to see the value of its technology. Through some astute business strategies, Eidos PLC now controls most of the best-known computer games software brands in the UK, producing such blockbusters as Tomb Raider, featuring the famous Lara Croft. Eidos retains its Acorn links, in fact if you inspect the movie files in Tomb Raider, you will see they are in fact Replay movies. Stephen Streater is now a very wealthy and successful man and he is a Clan member. We had a chat with him recently:

CLAN MAGAZINE: You must be the Clan's richest and most successful paid-up member. Your success has come through the general UK games industry, which has little direct association with Acorn, and yet you continue to devote a great deal of time and effort - much of it personally - to Acorn-based products and projects. What is it about Acorn that feeds your enthusiasm?

STEPHEN STREATER: If you talk to people who have to use PCs, they almost universally say that they think it is a terrible platform, but they have to use it because "everyone else" does. Well, I don't have to use a PC, so I don't.

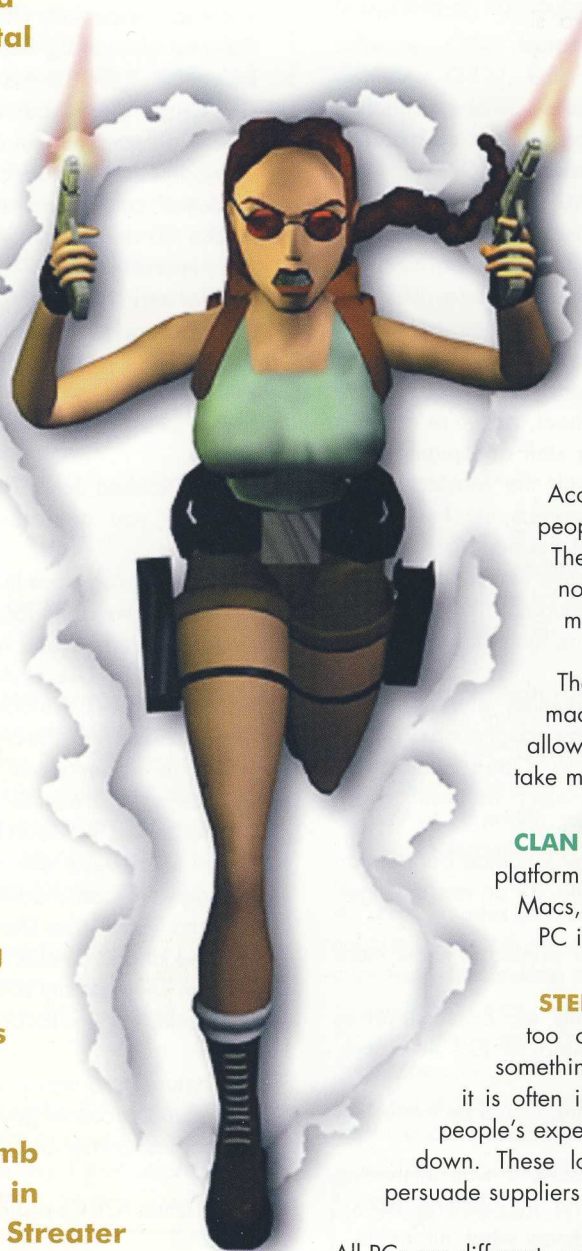
Acorn machines are designed to allow people to do what they want to with them. The system is designed to help you along but not force you to do things in a way which may be inappropriate.

The high level of compatibility between machines, even over long time periods, allows me to take on longterm projects which take many years to progress to fruition.

CLAN MAGAZINE: What is it about the Acorn platform which distinguishes it from PCs and Macs, what makes life easier for you on a Risc PC in comparison?

STEPHEN STREATER: PCs and Macs are too complex to use in reliable systems. If something doesn't work in a realtime application, it is often impossible to understand why not. Also, people's expectations of PCs is that they tend to break down. These low expectations make it very hard to persuade suppliers to fix their problems.

All PCs are different, so you never know how well something will work. This is very important in realtime applications, and increases the development time significantly.



Perhaps most importantly, Acorn is just down the road, and in the same time zone. This makes it very easy to resolve any issues we may need to. All our suppliers are also local, so we get a much better service from them when we need it.

CLAN MAGAZINE: There are rumours that some of the multimedia sequences in Eidos games are put together using Acorn equipment. Is this true and, if so, how widely is Acorn-prepared material used in Eidos titles?

STEPHEN STREATER: My team does all their R&D on Acorn machines. The video compression used in the Eidos games is tried and tested first on Optima, with the wide variety of videos edited on Optima. We usually keep the files in Replay format when they move to other machines, because it is a simple and efficient format. A lot of customers in other companies want the videos in their native formats (such as Apple QuickTime) because these are compatible with their native video editing and special effects tools. The tools we sell for games and multimedia companies are for Macs and PCs, because those are what our customers own. Most development in Eidos is on PCs or games consoles, though quite a few people at Eidos have Acorn machines at home.

CLAN MAGAZINE: We first heard of Eidos when the Optima professional off-line video editing system was created. I guess you had little inkling that your start-up company would grow to swallow up a large chunk of the UK games industry. Just how did that happen?

STEPHEN STREATER: I knew it could end up doing anything, so I am not surprised. Long-term R&D is always a difficult story to sell in the City. There are many things which made these things possible. Having a publicly quoted company was a major part in this. The editing system was the product we used to get the listing. I always made sure the company had no debt and lived within its means. This meant that I could take my time to look for the people with the right skills and experience to take the company forward. The person I found and who deserves the most credit for this is our Chief Executive, Charles Cornwall. A lot of friendly shareholders have given us a hand when we needed it as well.

CLAN MAGAZINE: Optima was clever technology in need of more powerful technology. 30MHz ARM3s in Archimedes A540 boxes has since been replaced by 233MHz Risc PCs. With some ten times the horsepower, what has this enabled you to achieve in improving Optima?

STEPHEN STREATER: We have better picture quality and data rates. Also, the systems we sell are much cheaper now, because StrongARM enables us to use a simple digitiser. People at the Acorn World Show were saying that the picture quality looked as good as VHS, and reviews are now saying we exceed VHS; but I think we need one more step to get beyond SVHS and this good enough for a mass market.

CLAN MAGAZINE: Optima is a professional product - can you tell us which programmes we might commonly watch on TV are

prepared with the help of Optima systems? And is there an update to the 'junior' product which used to be sold by Computer Concepts?

STEPHEN STREATER: The programmes are too numerous to quote them all - some are listed in our brochure. Perhaps 1,000 programmes were made last year on Optima.

The issue with a junior version is the cost of video capture. I think consumers would expect a finished programme. Optima itself is designed to output a text file (edit description list or EDL) like PostScript, but for video, and only professionals have access to the equipment needed to make use of this. In due course, the cost of digitising will fall and people will be able to afford a home Optima.

CLAN MAGAZINE: At Acorn World we saw the latest project from Eidos - Acorn Eye, which is claimed to be video over ordinary modem connections as you've never seen it before. What can we realistically expect from Eye technology in the future?

STEPHEN STREATER: The Eye is designed to have fully programmable hardware and software, and to be as cheap as possible to make. So people will be able to have any applications they want which make use of realtime image processing. We are currently planning a videophone and a security device.

CLAN MAGAZINE: What dreams do you have for the future of computer technology which Acorn might have a hand in delivering in the next five years?

STEPHEN STREATER: Cheap, reliable and fast computers. Then I can make some new things out of them. StrongARM could be made into a rather nice parallel super computer... ■



OPERATION ACORN

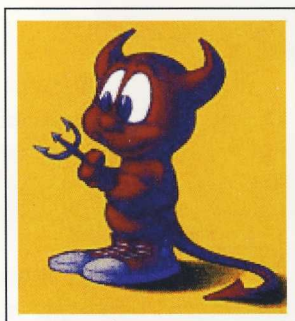
The aim of this article is to give readers some background on two free operating systems for Acorn machines, namely ARMLinux and RiscBSD. This introductory column looks at the historical aspects of the operating systems, what hardware they run on, and gives a brief overview as to their functionality. In future issues, we will look at the user interfaces available, as well as an overview of how to use the operating systems in dedicated applications, for example serving network computers.

History

Both ARMLinux and RiscBSD are often known as UNIX operating systems. It's very hard to define what UNIX really is. Going back to the 1940s, all computers were strictly single-user. Programmers would schedule several hour's use on a machine. When the programmer's time allocation came round, he or she would go up to the machine with a collection of punched cards, which would be fed into the machine by an operator. After submitting a job, it could take several hours for an output to be returned to the programmer - in most cases, the result would be an error. This made program debugging one of the most time-wasting activities that anyone could perform. One misplaced character on a punched card could waste hours of a person's time. Although this practice continued for a while, it was viewed as being very counter-productive. With this in mind, researchers at MIT joined forces with Bell Labs and General Electric to create a next-generation computer system called MULTICS, which stood for *MULT*(iplexed) *I*(nformation) and *C*(omputing) *S*(ervice). However, MULTICS did not succeed - the development tools required to write the OS were delivered too late and were poorly written, and the system was designed to allow hundreds of people to use a machine significantly less powerful than an A310.

Bell Labs pulled out of the project, which was generally regarded as being far too ambitious, leaving a man called Ken Thompson with nothing to do. He decided to write a stripped-down version of MULTICS on his own on an old PDP-7 minicomputer. He succeeded, and generated sufficient funds to support his development. The system was at the time known by researchers at Bell Labs as UNICS, the *UN*(iplexed) *I*(nformation) and *C*(omputing) *S*(ervice). This name was changed to UNIX at a slightly later date. Development continued, and gained the

support of a now well-known person called Dennis Ritchie, together with Ken's entire department at Bell Labs. It was ported to other machine architectures, and was rewritten in a new language called B. B was a stripped-down version of BCPL, which was in turn a simplified version of CPL (which never worked). This rewrite failed, due to weaknesses in the language, so a new language was designed to support yet another rewrite of the operating system: Logically, the new language, being the successor to B, was called C. Development of UNIX gradually continued, and became widely used in universities on powerful PDP-11 computers. The source code for the operating system was widely distributed, the result being that individuals from all around the globe started fixing bugs and adding enhancements to it. By the mid - 1980s, UNIX was in use on minicomputers everywhere.



Then, in 1984, AT&T (the holding company of Bell Labs) was broken up and decided to set up its own computer subsidiary. This new subsidiary released the first commercial UNIX product, System III. System IV never happened, and later on System V was released. System V was then replaced by releases 2, 3, and finally release 4, now commonly referred to as SVR4. Each new release added new features but had the side-effect of making the system more and more bloated. Early on in the development of UNIX by Bell Labs, it was acquired by the University of California at Berkeley. Using funding from DARPA (the Defence Advanced

Research Projects Agency), researchers from the Computing Science Research Group (CSRG) at Berkeley were able to modify and enhance the freely available source code, culminating in the first Berkeley Software Distribution, 1BSD. This was then followed by 2BSD, 3BSD and 4BSD, which all offered major enhancements, such as the addition of virtual memory, new filing systems and, for the first time, TCP/IP networking. These enhancements, together with the continued free existence of source code, caused many workstation manufacturers to base their core operating systems on the Berkeley distribution, such as Sun's SunOS.

The final release from Berkeley was 4. 4BSD, which was then developed solely by the free community. Two main strands exist, FreeBSD (for IBM PC computers) and NetBSD (an enhanced portable version of 4.4BSD, available for many machine architectures) on which RiscBSD is based. Back in 1991, a student called Linus Torvalds (then studying at the University of Finland) created a simple task switcher for PCs. This grew into an operating system in its own right, called Linux, which reimplemented the UNIX interface. Linus' operating system proved extremely popular among free developers, particularly because of the use of the GNU General Public Licence (GPL), which ensures availability of source code for anyone who wishes to obtain it, even for commercial products. Linus, together with an



increasing number of developers from all around the world, developed the Linux operating system into a full-blown multi-user environment capable of running on IBM PC computers. Because it contains no historical 'legacy' code, Linux is often noted for its efficiency compared to other similar systems. More recently, the OS has become much more portable in nature, leading to versions for several machine architectures, including the ARM - unsurprisingly called ARMLinux. Horses for courses!

ARMLinux and RiscBSD are both available for a variety of Acorn systems, and are finding their way into several commercial products; for example, the Digital EBSA-285 uses ARMLinux as its reference OS, and similarly the Digital Network Computer is distributed with RiscBSD. Generally, like most computing platforms, regular users of a particular operating system tend rapidly to become evangelised by what they know and love. BSD operating systems still tend to be preferred in 'workhorse' serving situations, whereas Linux has proved extremely popular in a whole variety of areas. To the end-user, there is little to choose between operating systems at present: Linux is a faster operating system in general, whereas RiscBSD has a more efficient floating point implementation; Linux's filing system is faster, whereas the BSD implementation is more recoverable, etc. Both offer largely the same collection of facilities. Highlights include:

- multiple users, either local or remote
- TCP/IP networking
- pre-emptive multitasking and thread libraries
- virtual memory
- demand paging of applications
- shared libraries
- X-Windows client/server graphical user interface.

For more information on how to get the operating systems and how to install them, two web sites have been provided at the end of this text. It is worth noting:

1) You will need to repartition a hard disc. Ideally, this should be a new disc; if not, you will need to move all RISC OS data off the existing disc, run !HForm, and then put it back on again.

2) Installation of both operating systems is not a trivial job, and may require patience as well as a big 'X' on a nearby wall for banging one's head against when things don't quite work as well as expected.

3) Care and attention are required; do not try to install one of these operating systems when drunk, or after a good night out!

Although these points may scare some people off, on the positive side help is available from several internet mailing lists, where existing users can assist with installation difficulties, and the maintainers of the operating systems can hold people's hands. The two operating systems utilise different hardware combinations. At the time of writing, to the author's best information, these are:

ARMLinux

IDE: Motherboard

ICS Serial: Serial Port Dual Serial, Atomwide 1,2,3 High Speed

Serial SCSI: Acorn, Cumana SCSI I, Serial Port EcoSCSI, Oak SCSI

Misc: Iomega Parallel ZIP

Ethernet: Ether1, Ether3, EtherB, EtherH

Machines: A540, A440, A440/1, A300 series, A3000, R140, R260, R225, A5000, A4000, A3010, A3020, A4, RiscPC, A7000, EBSA110

Processors: ARM 2, ARM250, ARM3, ARM610, ARM710, StrongARM SA110

RiscBSD

Ethernet: Ether1, Ether2, Ether3, EtherB, Ether5, EtherM, EtherH

SCSI: Acorn, Cumana SCSI II, PowerTec SCSI II, MCS Connect32, Oak, Morley uncached

IDE: Motherboard, RapIDE, Simtec, ICS

Machines: A7000, A7000+, RiscPC, Digital Shark, VLSI RC7500

Processors: ARM610, ARM710, ARM700+FPA, ARM810, StrongARM 110

Answers

Because ARMLinux and RiscBSD are operating systems in their own right, they do not require the assistance of RISC OS in order to operate; this does not, however, mean that your computer is tied to only one operating system. Both operating systems will happily coexist with RISC OS on a machine, and will offer a choice of operating system at boot time. Quite probably, another concern that people may have is: 'What can I do with these free operating systems?' A large variety of free software is available. One particular area where these operating systems perform very well is networking. They can talk to Apple or Microsoft networks, as well as the native TCP/IP, and can perform as either flexible clients or powerful multi-user servers. Other useful tools available include:

- editors, such as the famous GNU Emacs
- graphic manipulation tools, such as the PBM Toolkit, xv, ghostview, xfig (for vector graphics)
- typesetters, such as LaTeX and troff
- SQL databases, such as Postgres95
- development tools, compilers available include Ada95, C, C++, Fortran, Java, Modula, Perl, Tcl/Tk etc.
- multi-player video games....

the list is almost endless. In future columns, we will look in more detail at the variety and nature of the many tools and applications that are available. But you cant:

- run RISC OS applications such as Sibelius or Impression
- WYSIWYG desktop publishing, spreadsheets, etc.
- drag and drop data easily between applications
- experience a completely consistent user interface shared by all programs.

Do not let this worry you, however: A combination of both RISC OS and a UNIX operating system on the same machine can give you access to one of the most productive IT environments known to man!

For more information...

- ARMLinux: <http://www.arm.uk.linux.org/~rmk/>
- RiscBSD: <http://www.causality.com/riscbsd/>



"Correctly Addressed"

Continuing in similar vein to the Nested Window Manager and Browser Plugin details from the previous articles, in this issue we examine Acorn's new URI handler which ships as part of the !Browse distribution.

Up until last year, no RISC OS specification existed which described a standard method for different applications to communicate URIs between themselves. URIs can be thought of as a superset of URLs; in fact, a URL is an example of a particular limited form of URI ("limited" in that it is hardwired to a particular server or page). Readers who write WWW pages according to the HTML 3.2 or 4.0 specifications will already have already encountered URIs although they may not have known it at the time; the "-//IETF//DTD HTML//EN" argument supplied to the <!DOCTYPE ... > tag is a URI. Applications may wish to pass URIs around, for example, to enable an address book to request that a Web browser display someone's homepage.

To date, several third party developers have independently solved this problem in a variety of

different ways, as there was no centrally published, universally available standard for developers to work to. This document defines such a standard.

The 'central resource broker' detailed below will be extended in the future to provide mechanisms to enable more efficient handling of URIs. For example, data may be passed to an appropriate application based on the type of data as opposed to simply the method specified for retrieval of the data, as is often the case with URLs. This too will be via a service interface to the central broker.

The software takes the form of a RISC OS relocatable module, entitled 'AcornURI'. This is a generic, OS-level software component that could sit equally beneath a text editor which was aware of the form of URIs or beneath a Web browser or mail / news reader.

The module is suitable for use on RISC OS 3.10 upwards, and should be stored in !System.310.Modules.Network as 'URI'.

Programmer's interface

The application programmer's interface to the services provided by the Acorn URI handler is detailed in the following sections. This interface will be enhanced in the future to provide a more comprehensive set of services; it is therefore worth emphasising that only those details and features of the interface specified in the following sections should be considered to be supported. Any behaviour which is not specified below should be considered to be an implementation feature of a particular version of the software, and as such liable to change, alteration or omission without notice.

The following have been allocated for the use of the Acorn URI handler:

Module name URI

SWI prefix	URI
SWI chunk	&4E380
WIMP message chunk	&4E380
Error code chunk	&810A00
Service call	&A7
Filetype	&F91

All environment variables containing the string _URI_ (i.e. matching *_URI_*)

URI 'handles' are utilised to identify a specific URI request when communicating with the URI handler; tasks may assume nothing about these handle values, other than that they identify a particular URI to the handler for the period of their validity.

URI SWIs

URI_Version (&4E380)

On entry
RO = flags:

bit	meaning if set
0-31	reserved (must be 0)

On exit
RO = current version * 100

Interrupts
Undefined

Re-entrancy
Undefined

Use
This SWI is used to inquire of the URI handler module's version number, and should be used to check for a suitable version being present before using the facilities provided.

The number returned is of the form (major version * 100) + minor version.

URI_Dispatch (&4E381)



On entry

R0 = flags:

- bit meaning if set
- 0 inform caller of result (\Rightarrow R2 valid)
- 1 check only, don't process (R0:0 must be set)
- 2 don't attempt external process startup
- 3-31 reserved (must be 0)

R1 = pointer to 0 terminated URI string
R2 = 0, or source task handle if bit R0:0 is set and the caller is a WIMP task

On exit

R0 = flags:

- bit meaning if set
- 0 request rejected, URI won't be dispatched
- 1-31 reserved (must be 0)
- R2 = task handle of URI handler
- R3 = handle of this URI (request identifier)
- All other registers preserved.

Interrupts

Undefined

Re-entrancy

SWI is not re-entrant

Use

This SWI is used by an application to pass a URI string to the handler for dispatch, or checking for the presence of a potential servicer. Dispatch provides for optional requesting of a success/failure indication (R0:0 set) via a WIMP message (URI_MReturnResult) or service call reason code (Service_URI_ReturnResult) - necessary since the dispatch of the URI occurs asynchronously.

If R0:0 is set, module clients must signal that a URI_MReturnResult message is not necessary by setting R2 to 0. In this case, only the service call will be sent out. Conversely, WIMP task clients must specify a valid task handle in R2 - in this case, only the WIMP message will be sent out.

When requesting a check only (R0:1 set), it is an error not to set R0:0 and fill in R2 as described above.

The URI will be copied to the URI handler's workspace, optionally transformed (future enhancement, such as canonicalisation), then relocatable modules will be offered the chance to handle the URI via service call &A7 with an appropriate reason code (Service_URI_Process); if the service call is unclaimed, then a User_Message_Recorded WIMP message will be broadcasted (URI_MProcess), offering other tasks the chance of handling the URI; if neither of these mechanisms elicits a response, then the request will be deemed to have failed (in so far as active tasks are concerned).

If R0:2 is clear, then the 'fallback' position of checking a subset of the environment variables will be used to attempt to start a suitable task to handle the URI. The handle ceases to be valid at this point if notification has not been requested, irrespective of whether or not the URI has processed.

If R0:0 is set, the originating task will be informed of the results of the dispatch process (via a User_Message_Recorded WIMP message URI_MReturnResult if R2 contains a valid task handle, or service call Service_URI_ReturnResult if R2 is zero). If the message is not acknowledged or service call claimed, the handle will cease to be valid; otherwise, the originating task becomes responsible for indicating that it no longer needs the URI by calling SWI URI_InvalidateURI.

URI_RequestURI (&4E382)

On entry

R0 = flags:

- bit meaning if set
- 0-31 reserved (must be 0)
- R1 = pointer to buffer to hold URI or 0 to read required size
- R2 = length of buffer or unused (if R1 = 0)
- R3 = URI handle

On exit

R2 = offset into buffer of terminating null, or size of buffer required (if R1 = 0 on entry)
All other registers preserved.

Interrupts

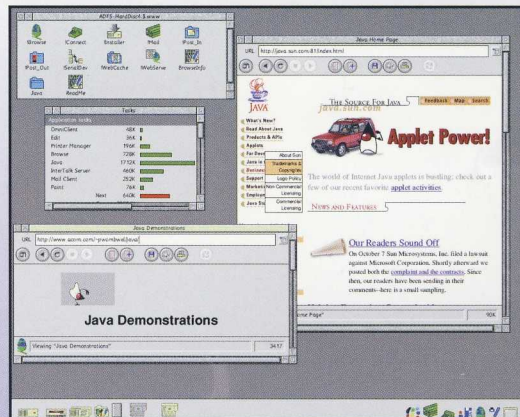
Undefined

Re-entrancy

SWI is not re-entrant

Use

This SWI is used to inquire what size of buffer is required to hold the specified URI (if R1 is zero on entry), or to pass details of a buffer into which your task desires the URI to be copied.



If this is successful, then R2 should be equal to the size of the buffer: if the buffer specified on entry is not large enough, then R2 will be returned negative (indicating the number of unreturned characters), and the string returned in the buffer will

still be zero-terminated i.e. buffersize-1 characters of the string are returned.

URI_InvalidateURI (&4E383)

On entry

R0 = flags:

- bit meaning if set
- 0-31 reserved (must be 0)
- R3 = URI handle

On exit

All registers preserved.

Interrupts

Undefined

Re-entrancy

SWI is not re-entrant

Use

This SWI is used to mark the specified URI



as being invalid.

URI service calls

Service call &A7 has been allocated for the use of the URI handler; the following sub-reason codes are defined for the use of external applications. All other service call reason codes are reserved: a module may assume nothing about these, and should always ignore unrecognised reason codes - never claim such service calls.

A deliberate degree of similarity exists between the WIMP messages and the service calls, since both provide essentially the same functionality; clearly, messages will be convenient in environments where service calls are not and vice versa, hence the duplication of functionality between the two.

Reason 0: Service_URI_Started

URI handler started

On entry

R0 = 0 (reason code)
R1 = &A7 (service call)
R2 = flags:

bit meaning if set
0-31 reserved (must be 0)

On exit

All registers must be preserved -
the call must be passed on.

Use

This service call indicates that the URI handler has started. It is intended for more specific use defined in future versions of this specification.

Reason 1: Service_URI_Dying

URI handler dying

On entry

R0 = 1 (reason code)
R1 = &A7 (service call)
R2 = flags:

bit meaning if set
0-31 reserved (must be 0)

On exit

All registers must be preserved -
the call must be passed on.

Use

This service call indicates that the URI handler is dying. It is intended for more specific use defined in future versions of this specification.

Reason 2: Service_URI_Process

Process or check URI

On entry

R0 = 2 (reason code)
R1 = &A7 (service call)
R2 = flags:
bit meaning if set
0 check URI only, do not process
1-31 reserved (must be 0)
R3 = pointer to URI string
(readonly access)
R4 = handle of this URI

On exit

R1 preserved or 0 to claim
All other registers preserved

Use

This service call indicates that the URI handler has been requested to dispatch the given URI for either processing (R2:0 clear), or just checking (R2:0 set). The URI string is held in the URI handler's workspace; this buffer must not be written to - if it is, behaviour is undefined. It is intended that modules should inspect the string at the given address, and if they decide they can process the given URI, claim the service call. If R2:0 is set, this is all that is required.

However, if R2:0 is clear, i.e. process URI, then a call to SWI_URI_RequestURI to obtain a local copy to work with must be made; this step may NOT be omitted, since the internal buffer is not guaranteed to remain valid after return from the service handler.

If a module cannot process the given URI, it must pass the call on with all registers preserved to allow the remainder of the dispatch mechanism to function.

Reason 3: Service_URI_ReturnResult

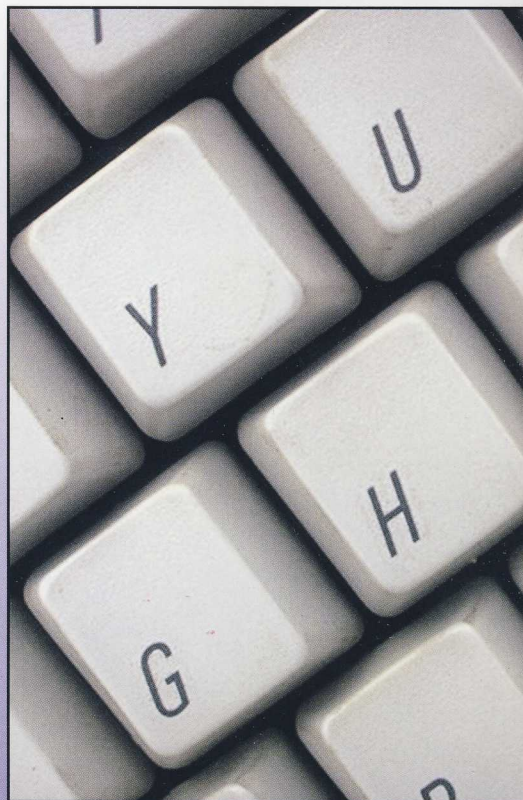
Return result of a dispatch

On entry

R0 = 3 (reason code)
R1 = &A7 (service call)
R2 = flags:
bit meaning if set
0 0=> URI was claimed for processing
1=> URI was not claimed for processing
1-31 reserved (must be 0)
R3 = undefined (reserved (0))
R4 = handle of this URI

Use

This service call is used by the URI handler to return result status information to a



requesting module. The module requests the service call when it calls the URI_Dispatch SWI; it must set R0:0 and R2=0 on entry. Such modules must remember the URI handle returned in R3 by this SWI or they cannot later determine if the service call was meant for them or another client; any client setting R0:0 on entry to URI_Dispatch must see if it recognises the URI handle in R4, and if so, claim the service call. If it does not recognise the handle, it must not claim the service call. Any clients which never set



R0:0 on entry to URI_Dispatch can ignore the service call.

Only success or failure is indicated, though this is likely to be enhanced in future.

WIMP messages

Message &4E380: URI_MStarted URI handler started

Poll block

R1+20 = flags:
 bit meaning if set
 0-31 reserved (must be 0)
 R1+24... undefined (reserved)

R1+24... undefined (reserved)

Use

This message is broadcast (User_Message) to indicate that the URI handler is shutting down. It must not be acknowledged - information only.

Message &4E382: URI_MProcess Process or check URI

Poll block

R1+20 = flags:
 bit meaning if set
 0 check URI only, do not process
 1-31 reserved (must be 0)
 R1+24 = pointer to URI string
 (URI internal buffer)
 R1+28 = URI handle
 R1+32... undefined (reserved)

Use

This message is broadcast (User_Message_Recorded) to indicate that the URI handler has been requested to dispatch the given URI for processing, or check if any task can process the URI.

The URI string is held in the URI module's workspace; this buffer must not be written to - if it is, behaviour is undefined.

It is intended that applications which can process URIs should inspect the string at the given address to determine if they can process the URI. If R0 bit 0 is clear, you must then call SWI URI_RequestURI to obtain a copy to work with - this step may not be omitted, since the buffer given is not guaranteed to remain unaltered.

If an application is able to check or process the given URI, then it should acknowledge the broadcast by sending a URI_MProcessAck message to the URI handler, thus preventing it being passed on to other applications, otherwise it must not acknowledge the message.

Message &4E383: URI_MReturnResult
 Return result of a dispatch

Poll block

R1+20 = flags:
 bit meaning if set
 0 0 => URI was claimed for processing
 1 => URI was not claimed for

processing

1-31 reserved (must be 0)
 R1+24 = URI handle
 R1+28... undefined (reserved)

Use

This message is used by the URI handler to return result status information to a requesting task. Only success or failure is indicated, though this is likely to be enhanced in future.

Message &4E384: URI_MProcessAck
 Acknowledged URI_MProcess

Poll block

R1+20 = flags:
 bit meaning if set
 0 Check URI only, do not process
 1-31 reserved (must be 0)
 R1+24 = pointer to URI string
 (URI internal buffer)
 R1+28 = URI handle
 R1+32... undefined (reserved)

Use

This message is used by clients of the URI handler to indicate to the URI handler that they can claim or process a given URI, thus preventing it being passed on to other applications. Claimants just change the message type to &4E384 (URI_MProcessAck) and copy the supplied my_ref field into your_ref, then send the message back to its originator (ie. the URI handler).

* Commands

*Desktop_AcornURI
 Starts the URI handler

Syntax
 *Desktop_AcornURI

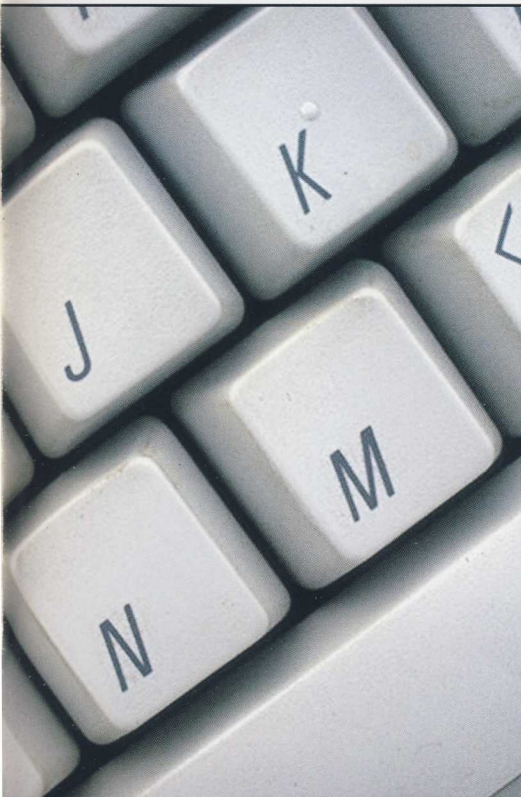
Parameters
 None

Use

*Desktop_AcornURI starts the Acorn URI handler. Do not use *Desktop_AcornURI, use *Desktop instead.

Help text

Do not use *Desktop_AcornURI, use *Desktop instead.
 Syntax: *Desktop_AcornURI



Use

This message is broadcast (User_Message) to indicate that the URI handler has started up. It must not be acknowledged - information only.

Message &4E381: URI_MDying U R I handler dying

Poll block

R1+20 = flags:
 bit meaning if set
 0-31 reserved (must be 0)



Example
 *Desktop_AcornURI
 Use *Desktop to start AcornURI

Related commands
 None

Related SWIs
 None

Related vectors
 None

***URLinfo**
 Display information about the
 URI handler

Syntax
 *URLinfo

Parameters
 None

Use
 *URLinfo produces status information
 from the Acorn URI handler

Help text
 URLinfo produces status information
 from the Acorn URI handler.

Syntax: *URLinfo

Example
 *URLinfo
 URI_taskhandle: 4b4016d8
 URI chain start: 021cc844
 URI handle: 022b60d4
 (action:00020000)
 'http://www.acorn.com/'

Related commands
 None

Related SWIs
 None

Related vectors
 None

***URIdispatch**
 Try to launch a URI

Syntax
 *URIdispatch <uri>

Parameters
 uri: the uri to be launched

Use
 *URIdispatch tries to launch a given URI. No
 indication is given of whether or not the
 launch succeeded.

Help text
 URIdispatch tries to launch a URI.
 Syntax: *URIdispatch <uri>

Example
 *URIdispatch http://www.acorn.com/

No: Base + 2
Cause: An empty URI string is supplied
 (e.g.to URI_DispatchURI).

Name: Error_URI_BadHandle
No: Base + 3
Cause: A bad URI handle has
 been supplied.

Name: Error_URI_BadFile
No: Base + 4
Cause: Reported when there is an error
 accessing a URI file.

Generators of the errors are as follows:



Related commands
 None

Related SWIs
 URI_Dispatch

Related vectors
 None

Generator	Returns
URI_DispatchURI	Error_URI_NoMemory
	Error_URI_BadURI
URI_RequestURI	Error_URI_BadHandle
URI_InvalidateURI	Error_URI_BadHandle
*URIdispatch	Error_URI_NoMemory

Finally, the WIMP task may generate (through
 a standard WIMP error box)
 Error_URI_NoMemory and Error_URI_BadFile.

URI handler errors

The URI handler has a error chunk base of
 &810A00. Currently defined errors are:

Name: Error_URI_NoMemory
No: Base + 1
Cause: There is not enough memory to
 complete an operation.

Name: Error_URI_BadURI

Use of the URI filetype

URI files have the filetype &F91, with the
 text equivalent 'URI'. The URI handler will
 deal with such files appropriately when the
 file is double-clicked upon (currently, it
 dispatches the URI inside the file - see the
 file format description below). Applications
 must not set an Alias\$@RunType variable



for the URI filetype, nor must they deal with DataOpen messages for this filetype. Applications may respond to DataLoad messages for the filetype as they see fit.

URI files consist of a series of lines of characters. Lines are ended by any number of control code characters (ASCII code less than 32) or the end of the file. All lines in a file do not have to end in the same way provided each individual line ends in a valid manner. Other white space is not ignored, hence a single space character (ASCII code 32) followed by ASCII code 9 does count as a line containing a single space followed by a line end marker.

URI files support comments. Comment lines start with a '#' (ASCII 35) and end in the same way as all other lines. Comment lines are not counted; any file reader that happened to keep track of the line number it was on should not increment the counter for a comment line. A URI file may contain any number of comment lines, but automatic file generators are encouraged to keep comments to a bare minimum to keep file sizes down. Generator code must never create special comment lines which mean something to accompanying reader code - comment lines are always skipped by the reader code and never parsed, beyond identifying them as comments.

The line ending type of a URI file is not fixed as a specific control code or sequence of control codes (e.g. CR+LF) to allow simple generation from a variety of sources, including manual authoring. Given this latter possibility, it is important to stress that unlike, say, HTML, the URI file format is rigorously defined and must be adhered to. Incorrectly formed files are not guaranteed to work correctly with either the Acorn URI handler or applications which support it.

That said, the use of ASCII code 13 followed by ASCII code 10 (CR+LF) to end lines is strongly encouraged as this is a common line ending type supported by many different editors on many platforms. ASCII code 9 (tab) could also be used to give the file a better visual appearance in the editor - it is still an end of line as far as the file reader is concerned. This convention provides the potential for greater convenience for the end-user, but must NOT be assumed in file reading code!

Currently defined formats:

Supporting URI module version
(None at present)

Line number 1 Contents:

'URI' - this must be present before any comments or other information.

Line number 2 Contents:

Text equivalent of the earliest module version number (as returned by URI_Version) that would fully understand the file contents; e.g. '5' for v0.05 (any number of preceeding '0's are also valid). So if lines were added to this file format to produce a version 6 file, this implies that URI v0.06 is required to understand those extra lines, even though v0.05 would still understand lines 1 to 4.

The first general release version of the URI handler will adopt a version number of 1.00, so the first URI files will start with '100' in this line.

Line number 3 Contents:

A fully specified URI; v0.05 of the URI handler does not attempt to canonicalise URIs, though future versions may. If this line contains only one character with ASCII code 42 ('*'), the file does not contain a URI and should be ignored (this is to allow future file formats to hold non-fully specified URIs on later lines that could be canonicalised by the URI module, without breaking legacy file reading code).

Lines 1 to 3 are required in a minimal URI file. Any other lines may or may not appear.

Line number 4 Contents:

A title string to associate with the URI. Again, if this line contains only one character with ASCII code 42 ('*'), the file does not contain a title string. Processors wishing to display title information alongside a URI may well use the URI itself instead, in this case.

Future file formats will be backwardly compatible with this one, so clients should only check the version number of the file to know what sort of contents to expect. So for example, if a version 100 aware application encounters a later version file, it

can assume that the first 4 lines of the file are as described for the version 100 file; though there may be other lines which clearly it cannot understand, and must ignore.

For example, the file format rationale may be easier to understand given the possibility of a future format - version 101, say - which allowed non-fully specified URIs in line 5 which can be canonicalised, and a preferred external process to start in line 6.

The file could look like this:

```
-Start of file-
URI
6
```

```
*
Acorn Group PLC
www.acorn.com
```

```
<Browse$Dir>.!Run
-End of file-
```

Use of URI environment variables

Currently defined variables are of the form:

Alias\$Open_URI_<scheme> <file to run>

for example,

```
Alias$Open_URI_http <Browse$Dir>.!Run
Alias$Open_URI_ftp <FTPClient$Dir>.!Run
```

If a variable such as the above is defined, then the task it names will be run. If this is successful, the URI will be redispached in the normal way, so the task has the opportunity of dealing with it.

A comma separated list of handlers may be specified, so applications must always add to the contents of the variables. At present, only the first item in the list is used, though this may change in future versions. ■



NAME THAT COMPUTER

What does the new RPC II mean to you?

- Now is your chance to personalise the next generation of computers by suggesting a name that will suit the forthcoming release.

The winning name will give a new identity to the machine and should relate to the new range of upgraded features incorporated.

The name will be used to globally position the RPC II and will be integrated on the computer casing, printed materials and a full marketing campaign.

Please send your suggestion by 1st May 98 to:

Clan Acorn
Acorn House
645 Newmarket Road
Cambridge, Cambs CB5 8PB

Alternatively you may email your suggestions to
Clan@acorn.com

The chosen name will be revealed at the Wakefield Show

Acorn 