



This is the TPC's first all-electronic issue of the newsletter. Let me know what you think. And if you have ideas for improvement or better yet ideas for submissions of articles, be sure to let me know about them, too. May you all have a very happy, prosperous, and exciting Year of the Ram!

—Mike Lloret

Volume 20, Issue 1

January 2003

*Users Helping Users
with Solutions
in Computing*

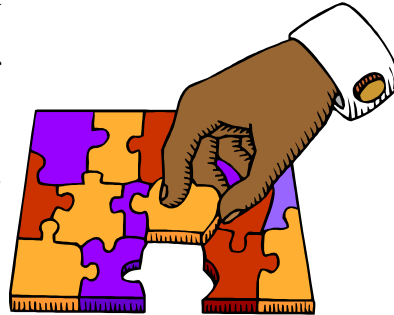
Searching, matching & replacing successfully - Part 1 by Kurt Keller

Could be useful for other stuff, couldn't it?

Did you ever use the command `dir *.txt` on a DOS or Windows machine? Have you ever typed `ls *.tar` on Unix? What is the asterisk standing for? Exactly. It means any number (0 or more) of any sign, or shorter: anything. So `*.txt` actually means anything followed by `.txt`. This can be as little as `.txt` itself or it can be `music.txt` or `this.txt-and-that.txt` or even something more complicated.

Wouldn't it be wonderful if this could be applied to more than only

file listings? And wouldn't it be terrific if it was a bit more flexible? Well, the same concept can be applied to more situations and it is much more flexible. What you're look-



ing for is called Regular Expressions, or Regex for short, and is available in many utilities, many programming languages and built into many applications. You

just need to know it is there and how to use it. If it is not built into the application where you need this, then there is probably not much I can do for you. Your best bet would be to export the data to some file and use some utilities to massage the file. If the problem is rather that you don't yet know what all these hieroglyphs mean, then I might be able to shed some light onto it.

When talking about Unix, then I feel comfortable and can join the discussion. When the topic is Windows, I

(Continued on page 4)

Got Computer Problems?

Join the Club!

The Tokyo PC Users Group is Japan's largest English speaking personal computer club, with a particular interest in bilingual computing.

Users of any platform or OS, of any ability, from occasional web surfers to genuine geeks and gurus should check out our on-line newsgroups and monthly meetings. We all have techniques to learn and tricks to share.

Inside this issue:

Midnight Writer	3
Architectures & Servers	7
Absent Minded Perfesser	8
Ionic Column	9
Babbitoon	14

About the Tokyo PC Users Group

www.tokyopc.org

Here you can find out about upcoming meetings and events, catch up on past meetings you may have missed, browse the newsletter online, and find out about all of the membership services.

(Contributors are always welcome. Mail your commons, suggestions, or interest in helping out to:

tpc-webmasters@tokyopc.org.

TPC Mailing Lists

The group provides mailing lists that mirror each of the newsgroups. Send a blank message to: info-lists@tokyopc.org to receive a catalog of the current lists and complete information on how to subscribe.

TPC Newsgroups

The group runs its own private news server that you can access over the internet from anywhere in the world. There are newsgroups for everything from help requests and Windows troubleshooting to non-computer topics like movies and dining around Tokyo. Set up your newsreader to point to news.tokyopc.org or visit <http://www.tokyopc.org/tpc/newsgroups.html> for a "no-setup" way of trying out the service.

Snail Mail Correspondence

Tokyo PC Users Group, P.O. Box 103, Shibuya-ku, Tokyo 150-8691, Japan

E-mail Correspondence

Mail: president@tokyopc.org, or any of the listed officers with questions or suggestions for the TPC.

Membership

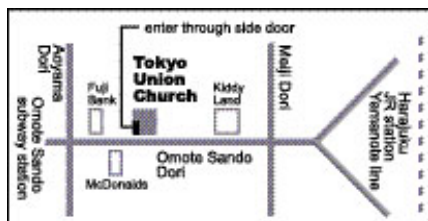
Individual: ¥10,000 per year. Includes newsletter, admission to regular monthly meetings, priority registration to SIGs and workshops, and other benefits.

Corporate: ¥25,000 per year. Full details:

www.tokyopc.org/tpc/join_tpc.html

Monthly Meetings

Meetings are the first Thursday of the month (2nd Thursday in case of Holiday conflict) in the basement of the Tokyo Union Church. Doors open at 6:30pm and the meeting begins at 7:00pm. Fee for non-members is ¥1000. See the map below.



From Omotesando station, take exit A1. When you reach the street, with the Hanae Mori Building on your left, continue downhill to the Tokyo Union Church on the corner of the third side street. Turn left at the church and use the side entrance inside the carport. Take the stairs down to the basement to the meeting room. Note that the TPC is not affiliated with any religious group. (Most of us head down the road to Shakey's for beer and pizza after the meetings.)

Executive Committee

President: Barbara Manning
barbara.manning@tokyopc.org

Vice President & Reviews Editor:
S. Patrick Eaton
Spe@l-sako.com

Treasurer: Anthony Whitman
03-3770-1513
tonyw@gol.com

Editor & Past President: Mike Lloret
090-9380-3508
lloret@gol.com

Publicity & Newsletter Publisher:
Justin Gardiner
justin@fka.att.ne.jp

Corporate Liason: Andrew Shuttleworth
andrew.shuttleworth@tokyopc.org

Disk Librarian: Mike Kato
mikekato@gol.com

APCUG Rep: Ken Cotton
(03) 3984-9600/4487 (tel/fax)
kc@gol.com

Programming SIG: Robert Altena
Robert@RobertAltena.com

Webmaster Committee: Sajjad Zaidi
sajjad@vgkk.com

TPC Exec Meetings are held on the Monday following the General Meeting. All are welcome to attend.

Bank Details

Sumitomo Mitsui BC (Sakura) Bank,
Akasaka Branch #825
Regular Account #7174919
Name: Tokyo IBM PC Users Group

I have been told that I'm a stubborn man. I prefer to think that I'm not easily deterred from taking a path once I've decided on it. Whichever way you choose to look at it, many people will be surprised to find me still at the editing helm of the newsletter in its first all-electronic issue. I argued and plotted against doing away with the paper version of the AJ for years, and have several times said that such a move would also require a search for a new editor. I had—and have—several of what I think are excellent reasons for my adamant stance, and I believe that they're still as valid as they ever were. You are welcome to discuss them with me over a few beers some evening, but the fact is that I have finally come around to agreeing that although good, they're no longer good enough. Quite simply, the TPC cannot afford to print and distribute the printed version any longer.

With membership markedly lower than it was in the bubbly good old days, the group's cash is reduced. If we don't want to join the ever-increasing host of bankrupt organizations, we need to cut costs and raise income. A great deal of time and effort is being expended by the group's management to try to raise funds, but meanwhile the printing and distribution of the newsletter remained the highest costs by far, and so we have reluctantly decided to publish only an electronic version.

I may be stubborn, but I'm not vindictive without good reason, and I prefer to try to be part of the solution instead of adding to the problem, so I have decided to stay on in the job. For now the plan is to make a PDF version of each newsletter available to members only for the month it is published, and subsequently to move it to the general public viewing space on the website.

One of my concerns has been that submissions will become rarer now, and indeed a couple of contributors have indicated they're not so interested in providing material for an all-electronic publication. On the other hand, another couple of people have said that they'd continue as before, which makes me very grateful, indeed. One of them, stalwart former editor/publisher and long time columnist David Parry, gives his views on the change in his column elsewhere in this issue.

From a work point of view, for me there's relatively little change except somewhat less immediacy about deadlines. It's a bit more troublesome converting



The Midnight Writer

By Mike Lloret

the finished newsletter to PDF format, but no more so than it was to burn the files to a CD, or to print camera ready masters as was done before that. And delivering the files to the TPC server minders should be much easier than it was getting them to the printer in time. One advantage is that any layout oddities in the PDF version will be either my fault or Justin's; there had been an increasing number of mismatches lately between my version and the printed version, apparently because the printer didn't have the default printer set properly, so that shouldn't be happening again.

This issue I'm doing both the editing and the layout work, so any errors are mine. You can expect some further tweaking in the next few issues as we strive to come up with what will look best both on the website and in the printouts that some readers may want to make. I'm always eager to hear suggestions for improvement.

I'm also always eager to receive submissions of articles, reviews, anecdotes, PC advice, or whatever you think might be of interest to other readers. My colleagues in the executive committee listed on the facing page are eager to hear any suggestions for improvement to the TPC, and especially any ideas for ways of increasing the membership and thus the group's income. In whatever form it comes, your input is valued.

Next time, I'll probably give you my impressions of a couple of new pieces of gear I've acquired: a Mitsubishi RDT175M-A monitor and an HP psc all-in-one printer. They were bought to replace a printer and monitor each of which died suddenly recently, and so far I'm happy with both of the new additions. We'll see if I continue to be. Meanwhile I wish all of you a very exciting and happy new year. Stay tuned.

(Continued from page 1) **Searching**

don't know much more than the average Joe (most secretaries know Word, Excel and all that stuff way better than I do) and when talking about Mac, OS/390, Novell and others, I have no idea at all. So in my examples I'm using mostly Unix tools and commands. If you want to go along with the examples, you don't need to have a Unix machine at home. If you're running Windows, then you can install Cygwin (<http://www.cygwin.com/>) which is a free Unix environment running in a window under Windows. It is simple to install and you don't need any Unix knowledge to install and run it. If you prefer to set up a Unix machine, you might want to try FreeBSD (<http://www.freebsd.org/>).

The very basics

With regular expressions, there are some signs which have a special meaning, just like the asterisk mentioned in the directory listing example. There are more than one regular expression engines and not all the tools support the all of the special signs, but the basics usually are quite universal. Let's start right with an example.

We do have an email address book. This is a simple text file named `aliases.txt`, plain ASCII with one email address per line:

```
AJ editor <editors@TPC.org>
John Doe <jdoe@hotmail.com>
Alice Springs <alice@yahoo.co.au>
TPC President <president@tpc.org>
TPC program director <programs@tpc.org>
Bob Hacker <badboy@ftpc.org.net>
Dave Cybercop <hunter@angels.net>
```

Now we're informed that the Tokyo PC Club changes its domain name from `tpc.org` to `tokyopc.org`, because the Transaction Processing Performance Council so badly wants to get the domain `tpc.org` and has offered the Club a lot of money for it. We now could simply open the `aliases.txt` file with a text editor and change every occurrence of `tpc.org` into `tokyopc.org`. If our `aliases.txt` file only consists of the seven entries above, this is no problem. But the whole

file does have around 700 addresses and unfortunately I don't have a secretary to whom I could assign the task to adapt my aliases file. Here a few tools which understand regular expressions can help us.

Regular expressions are always about matching some text. The better we can describe what we want to match, the more successful we're going to be. Regex is always line oriented, that is the whole match must occur on a single line to be recognized (exceptions are possible with certain tools but very rare). As there is exactly one email address per line in our `aliases.txt` file, we use the command `grep` (Global Regular Expression Print) to show us all the `tpc.org` addresses in the file. So first we must know what text we want to match. Should be `tpc.org`, right? So let's try:

```
%> grep tpc.org aliases.txt
TPC President <president@tpc.org>
TPC program director <programs@tpc.org>
Bob Hacker <badboy@ftpc.org.net>
```

Well, not quite what we expected. First of all the AJ editor's address is missing. This is because regular expressions usually are case sensitive. The command line switch `-i` for `grep` makes regex support case insensitive and will solve that problem. But why do we have Bob Hacker in this list? After all `ftpc.org` is not the same as `tpc.org`, right? Well, the dot is one of the special characters and means any character at all. So a single dot matches anything, but not nothing. If we want to match a real dot, we need to take the special meaning away from the dot, we need to escape it. This is being done by preceding it with a backslash. To prevent interpretation of the backslash by the Unix shell, we'd better put the whole regular expression between single quotes (I spare you the detailed explanation of this at the moment, just believe me for now). So with this added knowledge lets try again:

```
%> grep -i 'tpc\.org' aliases.txt
AJ editor <editors@TPC.org>
TPC President <president@tpc.org>
TPC program director <programs@tpc.org>
```

"The better we can describe what we want to match, the more successful we're going to be."

Cool, that's the list of all the tpc.org addresses we need to adapt. But I'm not yet satisfied with the regular expression we used. What if I had the email address <mike.oldfield@another-tpc.org> in my aliases.txt file? It would be matched as well because it contains the string tpc.org. We do know much more about the string we actually want to match than what we put into our regex. It is an email address and the domain is tpc.org. There will always be an *at* mark right in front of it. By including the at mark in our regex, we can further safeguard what we'll be matching. Add the following line to your aliases.txt file to see whether I'm having you on or telling you the truth:

Mike Oldfield <mike.oldfield@another-tpc.org>

```
%> grep -i 'tpc\.org' aliases.txt
AJ editor <editors@TPC.org>
TPC President <president@tpc.org>
TPC program director <programs@tpc.org>
Mike Oldfield <mike.oldfield@another-tpc.org>
```

```
%> grep -i '@tpc\.org' aliases.txt
AJ editor <editors@TPC.org>
TPC President <president@tpc.org>
TPC program director <programs@tpc.org>
```

Now we have safeguarded one side of the string we want to match. And now you expect me to also take precautions at the end of the string, right? Hey, you're getting the hang of it, good. Add another line to your aliases.txt file:

Donna Summer <donna.summer@tpc.org.tw>

When checking the syntax on our aliases.txt file, we see that all the email addresses are enclosed in angle brackets. If this is really so, it will help us to form a more reliably matching regular expression. I don't want to strain my eyes visually checking my own 700 lines aliases.txt file to see whether really all the email addresses end with an angle bracket. Instead I use another regular expression to confirm it. First we need to know how many lines there are in our aliases.txt. We count them by piping the whole file into the wc (Word Count) utility. The -l command line switch is used to only count lines:

```
%> cat aliases.txt | wc -l
9
```

And now we count how many lines end with closing angle brackets:

```
%> grep '>$' aliases.txt | wc -l
9
```

With the help of grep we filter all the lines which have a closing angle bracket (>) just before the end of the line (\$) and pipe this output into wc which then counts the lines for us. And good luck, the number is the same as with the last command, which means that all the lines end with closing angle brackets. So our final grep command looks like this:

```
%> grep -i '@tpc\.org>' aliases.txt
AJ editor <editors@TPC.org>
TPC President <president@tpc.org>
TPC program director <programs@tpc.org>
```

As you can see, the output is correct even though we now have the additional two 'troublemaker addresses' in our file, so our matching expression should be fine and reliable. With this output you can open your aliases.txt file and look for these entries to adapt them. No need to check each line in the file separately and possibly miss one.

Lazy people need to know a little bit more

I'm going to do something else, though. I'm using regular expressions to actually do the whole work for me. On one hand I'm lazy and on the other hand, even though I'm a fast typist, my typing is not so reliable, I make too many mistakes. So I'm going to use another of those handy little Unix utilities: sed, the Stream Editor. As most other tools and editors that support regular expressions, it does have a substitute function.

What you match by a regular expression (or parts of it, if you want to) can be substituted with something else. sed does, however, not have an option for case in-

"On one hand I'm lazy and on the other hand, even though I'm a fast typist, my typing is not so reliable, I make too many mistakes."

(Continued from page 5) **Searching**

sensitivity. So in order to match both upper and lower case I need to use character classes. When you want to match any one of a bunch of possible characters or signs, you can put the list of possibilities within square brackets. For example [abc] will match any one of the letters a, b or c. So if I want to match either an upper or lower case t I can use [Tt], for either case of the letter p it would be [Pp] and so on. The same can be used with grep:

```
%> grep '@[Tt][Pp][Cc]\.[Oo][Rr][Gg]>' aliases.txt
AJ editor <editors@TPC.org>
TPC President <president@tpc.org>
TPC program director <programs@tpc.org>
```

I'm not going to explain the whole sed command, but if you have been following along, it should not be too difficult to at least guess what is going on:

```
%> sed 's/@[Tt][Pp][Cc]\.[Oo][Rr][Gg]>/@tokyopc.org>/' aliases.txt >aliases.new
%> cat aliases.new >aliases.txt
%> rm aliases.new
```

Even though I'm using regular expressions often, I wouldn't dare doing this without using grep this way first to control what is being matched.

The whole thing could also be done in a couple of seconds reliably with regular expressions in my favourite text editor, vim. For those of you knowing vi or vim:

```
:% s/@tpc\.org>/@tokyopc.org>/igc
```

More to follow

As the sed and vim examples show, knowing how to use regular expressions can simplify your life and job a lot in many situations. Even adapting my 700 line aliases.txt file wouldn't take more than a couple of seconds with regular expressions and vim. Without regular expressions it would be a tedious, error prone and time consuming task. We have only slightly scratched the surface of what regular expressions can do.

Some very important concepts, such as quantifiers, or backreferences in substitution have not even been mentioned yet. Watch out for part 2 of this introduction to regular expressions.

References

Cygwin - free Unix under Windows
<http://www.cygwin.com/>

FreeBSD - free Unix
<http://www.freebsd.org/>

ViImproved - vi clone with many enhancements
<http://www.vim.org/>

P I N B O A R D
<http://www.pinboard.com/>

H i g h T e c h S a m u r a i
<http://kurt.www.pinboard.com/>

"...knowing how to use regular expressions can simplify your life and job a lot..."



Multi-tiered Architectures and Applications Servers (Part 1)

By David Smith

The last few years have seen a lot of change in the software industry. More and more software tools have become available to ease the job of the internal IT development staff. Typically these tools give a foundation of common functions that most applications use. This frees the developers to work on the real problems to be solved, often referred to as “business logic”.

The “application server” is one of these foundation products. An application server provides a basic framework to run a business application in. It handles housekeeping chores like startup and shutdown, networking and failover. The programmer develops modules that execute within the application server’s environment. In some ways, Oracle acts like an application server – it provides a framework for working with data and you can develop applications that execute inside of it.

Application servers have been around for six or seven years, however they were not heavily used until the introduction of Java and the Enterprise Java Beans (EJB) specification in 1998. Java really enabled application servers to take off; application servers based on C and C++ could be crashed too easily by poorly written code.

Multi-tiered architectures

In order to understand the role of an application server it helps to understand the kind of systems architecture it is designed to fit into. When designing large scale IT systems we talk about multi-tiered architectures today. A multi-tiered architecture is one where the functionality of the application is spread across multiple layers or tiers, each typically executing in its own server.

A single-tiered architecture is one where everything executes together in one monolithic server. A example of this is the classic Oracle configuration where us-

ers at dumb terminals interact with the SQL*Plus command line interface. In this example, the application is coded in PL/SQL and all of the application resides together in the Oracle RDBMS.

First generation client-server systems are two-tiered architectures. The user has a PC or workstation that runs a client program that interacts with the RDBMS. Also, many first generation web systems are two-tiered. The web server connects to an RDBMS and the code on the web server executes queries, inserts, etc. with the database.

Finally, more and more larger scale web sites and internal applications are moving to a three-tiered architecture. In this architecture, the web server or client program makes requests to a server that executes the application logic (often referred to as “business logic”). The intermediate server executes queries, inserts, etc. with the database. Using an application server allows a developer to concentrate on the “business logic” rather than all of the housekeeping details involved in that intermediate server.

Why would a developer want to move to a three-tiered architecture? Although the three-tiered architecture is more complex it also brings significant benefits.

First, scalability. In a single-tiered application, all of the business logic executes in the RDBMS. As the application logic grows more complex, more CPU power and memory must be added to the RDBMS server machine to handle processing the application logic, even if the number of actual database operations does not grow. In a two-tiered application, more of the processing can be pushed out to the client machines, however the client machines cannot share information. Also, updating the software on all of the client machines can be difficult.

“Although the three-tiered architecture is more complex it also brings significant benefits.”

(Continued on page 13)

The Absent Minded Perfesser

By John Philips



Crawling back to Windows

Once upon a time, before this column was born, I wrote something for Mike the editor about how I was switching from Windows to Macintosh. I had finally given up trying to get Arabic script up in Japanese Windows. Windows 2000 had promised the ability, but had not come through for personal users. Everyone I knew with the same problem had been telling me to just get a Mac. Some of them even laughed at me. I finally got a G4 with OS 9.2(J) and was very satisfied with it. I wasn't about to give up (or away) my Windows and Linux machines, but I did appreciate having a more flexible alternative.

"It was a perfect operating system for a multilingual family like mine. Or so I thought."

Then OS X came out. It was even better. I could change the operating system language from Japanese to English and about a dozen others. I could even create multiple users with different OS languages, so that I could log on straight into Japanese, English, or almost any other major language in my OS. It was a perfect operating system for a multilingual family like mine. Or so I thought.

OS X had no support for right to left languages. Not only was Arabic completely missing, but so was Hebrew. I spent some time on the help boards that Apple so generously provides. I was far from the only person who was upset, and I was far less upset than some. Of course you could still use right-to-left lan-

guages in the OS 9 shell, but it was a hassle to boot up the 9.2 shell every time you wanted something in such a language. And most of the major software vendors were already well advanced porting their popular software to OS X.

Sure, one big draw of Macs is their graphic and video capabilities, but another has been their multilingual ability. Mac had always been way ahead of Windows in supporting a wide variety of languages and writing systems on its computers. It was Unicode compliant long before anything Microsoft put out was. It was the platform of choice for anyone using a variety of languages, especially those requiring non-Roman alphabets.

Emphasize the "was" in that last sentence. While OS X was a unique blend of Macintosh's user friendliness (think Japanese bar hostess) with UNIX-like stability, flexibility and power (think Russian gymnast), it no longer matched Microsoft's multilingual ability (think US high school student).

Windows XP finally gave users the ability to enter Japanese and Arabic in an English operating system. In fact, it automatically downloaded whatever language abilities you needed, so you could just surf to the UN homepage <<http://www.un.org/>> to get set up automatically to handle Arabic, Russian, Chinese, or any of the other half-dozen UN of-

ficial languages. XP did ask you about it, but you didn't have to ask it, much less try to search a Microsoft database for the right files to install. Of course having Bill Gates' minions taking over your computer the way XP does left many people rather nervous, but that's another story, perhaps for another column. And I had other problems with that XP computer, some of which you may know about if you read the TPC newsgroups regularly.

Apple got the multilingual message fairly quickly, though. OS X.2 (aka "Jaguar") supports left-to-right languages, and in fact is fully Unicode compliant. In addition to the many languages you can use directly, and even make your OS work in, there is a "character palette" you can use to search for and insert characters. You can search by Unicode block, Unicode table, or glyph table. You can even save your favorites so you can find them easily. If it's in Unicode, and you have a font for it, you can use it in Jaguar. Finally there was a way to insert Arabic script Hausa language characters like `___*` [Unicode 069F]. Maybe now I can finally finish the Arabic script Hausa webpage I have been meaning to finish for years: <http://www2.gol.com/users/philips/Ajami.htm>.

Of course there are still problems. For one thing, there was no free upgrade to X.2. Had I known that such was the case, I, like many users, would have held off buying X.1 and just wanted for X.2, which had what we were really looking for. Furthermore, Unicode is an incredibly complex and constantly evolving standard and the issue of which characters should be added is being discussed on many fora around the net. It's not easy to keep up, even if you hang around the Unicode website:

<http://www.unicode.org/>.

I've given up talking about "switching" operating systems. I've come to realize that all major operating systems have their advantages and disadvantages. I like the idea of working with all of them. I've never been interested in Mac or Linux simply because I resented Bill Gates' money. I have to admit that his near success in monopolizing the computer OS market has had the effect of stifling innovation—Microsoft hasn't taken the lead in such things as multilingual ability—and breaking up Microsoft into separate OS and applications companies might have been one of the best things to happen to the computer industry. Still, if there's one thing the world in general—and yours truly personally—needs more than a Unicode compliant operating system, it's an effective malaria vaccine. That is one thing Bill Gates is way ahead of the competition in developing: <http://www.gatesfoundation.org/storygallery/malariamvi.htm>

John Edward Philips, Ph.D.
<http://www2.gol.com/users/philips/>

** [On my Win98 system, I was unable, even after several hours of searching for and downloading numerous font sets, and updating Windows and Office 2000, to get this character to appear in the text. The character is shown below, as a graphic, and is described on Unicode.org as "Arabic character **tah** with three dots above, Old Hausa". —Mike]*



David Parry

Ionic Column *in exile*

Englishman David Parry lived, worked and played in Tokyo from 1980 to 1994 and was a member of TPC from 1986. He was Newsletter Publisher from late 1988 to early 1990 and began the Ionic Column in 1992. This column even won a prize and an honorable mention back in 1992. Currently based in Düsseldorf and working as a translator, he returns to Japan electronically via the Internet.

The last Ionic promised more of the online exchanges with **Marc Prior**, but that will have to be held over for a month. Time was short for both of us, and in my case I had to finish a pair of translations before leaving for holiday, leaving the office barely before the witching hour. Marc has also been busy from what I hear, and for both of us it looks as if 2003 will be more the same, but he did outline his plans for possible future articles on a Linux theme. Work permitting, and of course the AJ has to take second place to anything that brings in money. While it is nice to have a moment of literary and online fame, more mundane considerations often get in the way of the artistic muse.

Save the trees

A message came from **Mike Lloret** just a few days before I left to tell me that the "dead trees" edition of the AJ was to be axed, if you will pardon the metaphor. As always, it comes down to money. The

sad truth is that it is cheaper to distribute electrons than paper. This move is not altogether unexpected, since it had been mooted before more than once, and other organizations that are in a similar situation are doing or will do the same thing in all likelihood.

Already obsolete when new

The big advantage of a Net presence for any organization that is primarily using a newsletter in its literal sense, as a source of news about future events, is that a Webpage can be constantly updated to include last-minute changes. A hiking club that I was a member of in Tokyo now publicizes its events primarily in the Internet for that very reason, and in the past the compliant had always been that the newsletter was out of date the moment that it went to the printer! That is not quite true in itself, but the lead time for the printing and distribution meant that it took at least a week before it plopped through the mailbox, and no

changes or additions could be made except by phoning around.

Hard facts on hard copy

The cost issue is very simple: the most expensive way to keep a permanent hard copy record of a document is to print it out on an inkjet. And I would not bank on it being permanent either, since ordinary black ink for inkjets is supposedly not all that permanent, certainly less so than toner. A laser printout is the next on the scale of costs, followed by photocopying. The latter is the only reliable way to get double-sided printing, which is a rare feature in most computer printers. As it happens, one of the reasons that I got my **HP 980 DeskJet** is that it can print on both sides, but it takes quite a while to do so. Printing out my **Trados** manual from the CD was a question of selecting about 30 pages and then going for lunch. While laser printers have dropped in price and generally got faster, making 8 to 10 ppm printers quite affordable, you are really looking at a monster for network use before you get the option of double-sided printing. Interestingly, the prices for color laser printers have dropped out of the stratosphere, but against that, I can only say that I have never seen one installed anywhere. Color still seems to be the province of inkjet printers for all but the more specialized users.

Mindless but error-prone

But I digress. Copiers are cheaper than laser printers, and the fancier models can print on both sides, collate the results, and in some cases even staple the pages together. These sort of copiers are mainly used by advertising companies or consultants needing, say, 30 copies of a 50 page report, and on a frequent basis. Secretaries are too expensive for a mindless but error-prone operation such as collating, whereas I have more or less happy memories from the past of collating, folding and stapling newsletters and publications of various kinds by hand. The next step was offset printing, taking a carefully pasted-up printout of our magnum opus to a jobbing printer and getting a finished publication back. The economics of offset printing in general are that it gets much cheaper per copy if the print run is larger, and the minimum that is worth doing is around 100 copies. That may have changed with some of the new digital printing technologies that I have heard about and seen at **Photokina**, but I have no direct experience of them. Also, one problem remains: if you want a regular type of publication that

is folded in the middle and stapled or fastened, you are caught in the law of four; the number of pages (sides) must be a multiple of four. The copy has to be stretched or pruned to fit, or more devious methods such as a change in font size are resorted to. Take a look at past copies of the AJ and you will see what I mean.

Carpe diem

Back to the question of paper. If the newsletter is not so much for news, but what I might call representational purposes, then perhaps a case can be made for a paper edition. If the newsletter is retrospective, describing what was done in the recent past and summarizing it, with a faint whiff of preserving the moment for a probably uncaring posterity, then paper can make a claim for being the better medium, and just possibly a longer-lived one. Digital data has the potential to be immortal, since it can be copied indefinitely without any loss of quality or accuracy. A tape recorder was the best method to transfer music by analog means, but even the costliest equipment running at high speed with the best quality tape added its own noise and errors, plus a small but significant amount of tape hiss with each copy. Old films and photographs lose quality each time they are copied, the colours and contrast shift, the shadows become darker and blocked up or the brighter parts, the highlights, become washed out, fine details are lost. Old films from the 1960s and 1970s shown on TV exhibit quite noticeable deterioration from what I remember from viewing them in the cinema when they came out, and I do not think that I am just suffering from a bad case of nostalgia. Newer films shown in their digital versions also lack the characteristic "noise" from dust, scratches, and the tramline scratches that develop from repeated passes through a projector. Look and compare.

Rushing into print

Are there any other advantages for paper, apart from it being a kind of "vanity publishing" for the authors and contributors? By which I mean the sort of thing whereby aspiring poets and novelists would get a printer to typeset and bind their text so that they could say that their work had got into print, albeit entirely at their own expense and perhaps for a readership in single digits. These days such people can indulge their literary vanities with a Webpage, or perhaps they are enthusiastic contributors to blogs. The kindest thing that I can say personally about

(Continued on page 12)

(Continued from page 11)

blogs is that they are very democratic, allowing one and all to have their say. At least they are not as messy as spray-painted graffiti, but I leave to the reader to decide on the relative quality of such offerings. And to ponder the oft-repeated and now oft-ignored injunction never to rush into print. Of course, your humble author more often than not has to speed this column on its way with indecent haste to meet the already overdue deadline, and a number of typos that accidentally found their way in, like blobs of airborne detritus that seem to have a fatal attraction for freshly applied paint, but a little thought can avoid at least the most egregious examples.

A cup of tea off the top of my head

Speaking as one who has to wrestle with words for a living, it is all too clear that many of the online postings I have seen were made by people who seem to be blissfully unaware of the difference between the written and spoken style, and the "off the top of your head" postings that grace the Internet are not quite in the same league as the "stream of consciousness" school of literature. To be fair, I have read very few publications of that type, and in all honesty that style is not my cup of Flowery Orange Pekoe, but I would hazard a guess that it requires a certain amount of skill, writing experience and literary discipline, however contradictory it may sound. Researchers have already noted a number of differences between the style of e-mails and snail-mail, and within the Internet itself there are also many different styles that are current. The TPC forum is (quasi) moderated, so we get less of the witless, name-calling flame wars that make the Usenet intolerable for all but the thick-skinned.

But what sort of progress do we see in the Internet? One wonders how long it took from the discovery of the telephone to the invention of the obscene call. Luckily it took a very long time before the telemarketers discovered the marvelous ability of the telephone to intrude into one's home at any time, an ability that was fortunately held in check for a long time due to the cost factor. But now we have very cheap telephony, for local calls at least, and the Internet is to all intents and purposes free. And anonymous, with a little ingenuity. It did not take long for the results of this fortuitous combination to appear.

Brave new insults

I'll comment briefly on the new trend in anonymous abuse first, inspired partly by comments I have seen in a local newspaper about the use of **SMS** to send short and abusive messages to users of mobile phones. To date I have never sent an SMS, since I can do my work by e-mail instead, and an SMS message is also not as cheap. But I read that schoolkids send each other abusive and anonymous messages by SMS. Back in the good old days, it was either a mob of bullies who found safety in numbers and resorted to name-calling from a distance, or somebody would write up a suitable epithet on the blackboard. The latter nearly always ended up with the culprit getting caught, even if they did not confess, since the teacher often recognized the handwriting, and the number of suspects was in any case quite small. Not so for a telephone call that has lost its caller ID. Or somebody in Usenet with a disposable Hotmail address. The anonymity of the new media makes it hard, if not impossible, to track down the sender.

Scanning the world

This anonymity helps and hinders. A number of less politically-enlightened countries object to their citizens reading or viewing certain materials, or passing them on, but even they cannot arrest a posting that has been anonymized by passing through a couple of remailers that strip off all identifying headers. As yet I have not bothered to use such methods, but they are described in a number of sources. In response, a number of security organizations are endeavoring to scan all e-mail that passes through the Internet to look for keywords in text, for example, The object is either to haul somebody into a police station and violate his human rights or else to nobly free the world of obnoxious terrorists, depending on one's point of view. There were some interesting postings recently online at TPC by **Chris Case** concerning the efforts of the US authorities to sift all possible sources of information and to sniff out putative terrorists with profiling techniques, and an article by a database expert who pointed out that the job required more computing power than we have—currently, at least—and that the results would yield a huge number of false positives. True enough; we have all seen how our names and mailing addresses get transmogrified as they are passed around from one mailing list to another, and computers have problems with variations in addressing that a human can allow for.

I'll conclude with a weird aside on the world of computers and the Internet. We have all seen how the Internet has made it easier for people to get in contact with each other. Or perhaps, to be more precise, to pass around porn and to pester with sales solicitations. At any rate, it allows people with the most off-beat of interests to get in touch. A pair in Germany made news recently. One could say that they were indeed made for each other, and also perhaps for the Darwin Awards. At any rate, I find them both even weirder than **Michael Jackson**, my previous yardstick for the wonderfully whacky. As a police psychologist noted, there have always been people who wanted to be killed, and from time to time there are people with cannibalistic tendencies, and for the first time there was a case that combined both. In this case the Internet acted as Cupid. Curiously, both the participants in this event were in the computer business, the diner as a computer technician, the dinner as a chip developer at **Siemens**, and both lived for little else but the online chat rooms, having increasingly little connection with the non-virtual world. I suppose that this is evolution of a kind, but not exactly what we had in mind two decades ago when the PC revolution started, or when the Internet started to take off less than a decade ago.

And in case you are wondering, the German police are slightly at a loss as to what to charge the surviving man with. Cannibalism as such is not listed as a crime, and the victim seems to not only have consented but to have requested this bizarre fate. And this is well documented from the e-mail both parties.

And in case you are wondering, part 2, how the German police stumbled upon the still hungry man-muncher, he posted another message in the same, ahem, vein a year after disposing of his former soul-mate and an alert doughnut-muncher decided to investigate.

Next month I'll be back to the bits and bytes and the postings from Marc Prior.

Comments or feedback or more information? A chance to be famous? Contact me at DAParry@t-online.com or mail@daparry.com. (Note: the WWW address listed in previous articles no longer exists.)

(Continued from page 7) **Architectures**

Second, separation of functionality. The three-tiered model allows the data access logic (talking to the database) to be separated from the business logic. The presentation logic which interacts with the user is separated from the first two functions. This lets you split the project across multiple groups. If you have separated your functions properly, you also reduce the risk of making changes to the code. In a one or two-tiered application, changing the "presentation logic" (the part that formats data for the user and interacts with the user) runs the risk of breaking the core components because the presentation logic and the application logic are mingled together.

Third, security. Having the web server connect directly to the database can lead to major security holes. Putting an application logic server in between can allow you to make a much more secure architecture. For proper security you must design the API to the application logic properly (narrow range of functions, don't pass SQL code through the API) and place each tier onto its own network separated by firewalls.

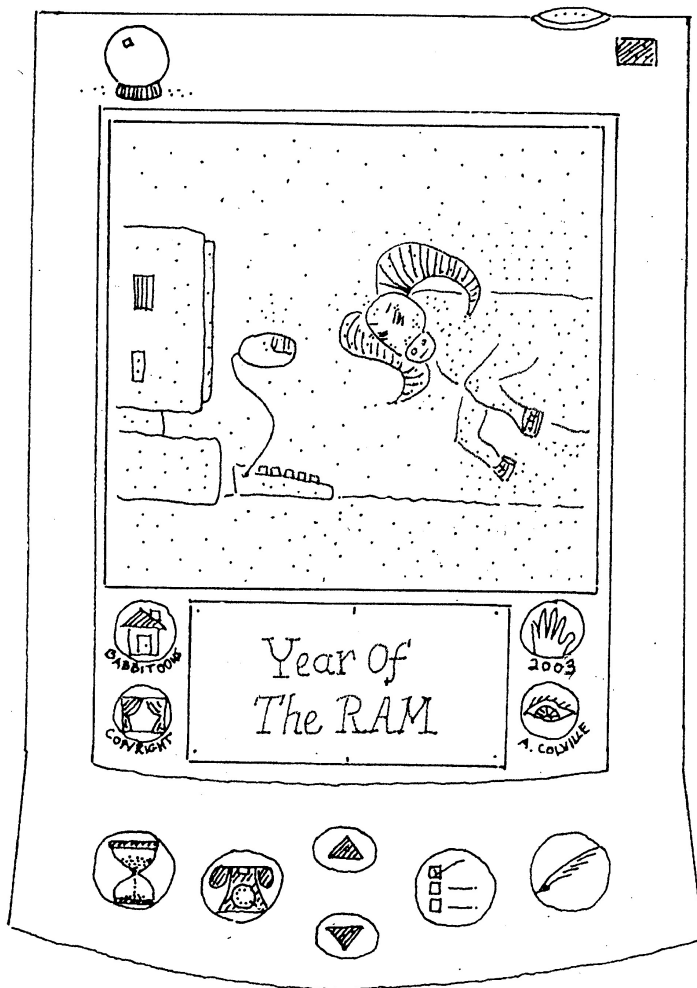
Fourth, flexibility. It's possible to develop multiple clients for the application logic. One could be a GUI interface for internal users, another the web based interface and a third could be applications that do automated tasks.

Three tiered applications aren't for everyone. Before diving into one you should have a clear design for what you are doing and a lot of programming and architecture expertise.

Next Time

Next time we will look at how the application server actually makes your life easier in the three-tiered environment.





CABLE & WIRELESS