



of course

VOLUME 5, NUMBER 1

SPRING 2005

KEEPING GOOD COMPANY BY

Going on a Geek Cruise

ALL HANDS ON DECK

When Open Source has commercial interests

NAVIGATIONAL SKILLS

Do a lot with a little Python



Contents

Geekmas greetings	2
What's new in our training centre?	2
Java course notes now available free	2
Helping out your navigation skills	3
Travelling the world	3
Our libraries of books, both real and virtual....	3
Who looks after MySQL?.....	4
Brag Page.....	5
Seat pricing GUI in Tcl/Tk.....	6
Getting to our training centre	7
You can do a lot with a little Python.....	8
Cruising with the Geek gods.....	10
Latest Versions of Open Source Software.....	14
Make your site's URLs more friendly.....	16
Embedding Layers.....	18
Simplifying complex joins	20
Back Issues	21
Schedule.....	22
Calendar	23
WHC welcomes new employee, Leah.....	24
Apress's distinctive looks filling shelves	24
Distributing open source applications	24

Geekmas greetings

We celebrated Geekmas last November with an open house weekend at our Melksham training centre, and a number of good students-who-have-become-friends came, some travelling quite a distance, to talk Open Source Shop.

On Saturday evening, we adjourned to an Indian restaurant, and even then the geek conversation flowed.

What's new in our training centre?

There's always something new to see and use for regular visitors to our Melksham training centre. Since our last issue, we've installed a new commercial tea and coffee machine that also makes lattes and cappuccinos at a press of a button. We've upgraded our Internet connection yet again to now give a four-megabit download. Building-wise, we're just completing an extension that will give us more space, and giving a new look to the conference room. Just in time too; you'll read elsewhere that we've added staff, and we started 2005 with a full Python course, and the year is only just getting started!



January 2005 brought us more space, as well as other improvements to match our growth



computer training

Well House Consultants

Downloadable Training Notes

We have been writing training notes and presenting courses for a number of years. The notes on this site are released under an "Open Training Notes License" - similar to an open source license - which lets you download and use them at no charge. You may also make and distribute copies, provided that you retain the Open Training Notes license with them and do not charge for the copies or any courses based on them.

- Training notes on Java (Complete 5 day course released)
- Training notes on Object Orientation
- Training notes on HTML
- Training notes on Perl
- Training notes on Python
- Training notes on PHP
- Training notes on Tcl/Tk
- Training notes on MySQL

You can also attend a [public course](#) presented by the author, or arrange for a private course at your office..

Further Information: [Link - why are we releasing these notes?](#)

Java course notes now available free

The complete notes for our "Learning to Program in Java" course are available for download from our sister web site at <http://www.training-notes.co.uk>.

Choose from more than 500 pages of Java notes (and HTML and OO design, too) from our Open training notes site, and even to copy the notes on to your friends and colleagues. The download is completely free; however, you are not to charge for the copies, nor charge for any training you give that makes use of the notes. We call this our "Open Training Notes" license, and the text can be seen in full at:

<http://www.wellho.net/net/whcotnl.html>



Helping out your navigation skills

Our site search now looks through more than 5,000 pages on our web site, and whilst we still try to keep all the information you'll want just two or three clicks away, inevitably it gets harder to see the wood for the trees. How can you easily find what you need?

Suggestion 1

Bookmark our resources index. This is the link farm that we use ourselves to navigate rapidly to our resources.

The first third of the page includes single links to major features such as:

- Training Module information, where you can see the source code of all the examples we use on our courses
- Opentalk Forum, where you can ask programming questions on Perl, PHP, Python and others
- Solution Centre, an area where we place technical papers of long-term interest
- Technical Library, details of the 550 Open Source books we have, with a search facility too
- Horse's Mouth, the daily diary of an open source trainer using tips, techniques and trivia
- Download centre, sample training materials, old issues of our "Of Course" newsletter, and more

Never forgetting we're a training company, the second half of the page is all about our courses:

- Arrangements for getting to Melksham from near and far
- Course descriptions, subjects, prerequisites etc.
- Course joining instructions
- Getting the best from your course
- Using your own laptop at our centre

Finally, we offer you an FAQ containing information you may want, but won't clutter up our main pages:

- Privacy, posting and copyright statements, terms and conditions, etc.
- Information for book authors, publishers, local businesses in Melksham and others who want to sell to us

Suggestion 2

Use the search box that's on the left-hand side of every page. We've tuned our search algorithm so that it reduces duplicate hits, and offers you alternative spellings if it doesn't get many hits with what you enter. More popular pages are listed on the result set before less popular pages.

A WORD FROM GRAHAM ELLIS

Travelling the world to bring you news about the Open Source community

Welcome to the Spring edition of "Of Course". As ever, it's packed with Open Source and training news and we've had to drop more ideas for articles than we've included due to pressure of space.

The last months have been a busy time for us, with a heavy training schedule from London to Liverpool and from Dublin to Guernsey. While this newsletter is at the printers, we'll be in Munich and Manchester too, as well as training at our Melksham base. Our public courses are getting busier; with a maximum of seven trainees per course, we advise you to book in good time. We started 2005 with a course that was filled, with a couple of customers having to wait for next running of the same course.

As well as training, we've been to North America, Asia and Southern Europe in our quest to keep ourselves right up to date, and you'll find a number of technical articles in this newsletter that tell you about PHP and how PHP's author, Rasmus Lerdorf, suggests it be used. We also discuss Perl 6 and how that's coming along (courtesy of a talk we attended given by Larry



Graham cruises with Perl father, Larry Wall

Wall). The world is large and these names rarely make it to these shores; we were very fortunate to have the opportunity to listen to major presentations to small groups by each of them, and we want to share the benefit of our new insights with you.

Our libraries of books, both real and virtual

located in Melksham, Wiltshire and at <http://www.wellho.net/library/index.html>

We've been collecting computer books for years, and that passion is reflected in our training centre library. Consisting of more than 550 computer-related books, and still growing, you'll witness this enthusiasm firsthand at our training centre. But you'll also gain from our efforts as well.

If you have a limited book budget, it's always a difficult choice knowing which book is best for you. At our centre, you'll be able to actually touch, read, use our books to get an idea if this book would be worth having in your own library.

Our website catalogues this collection and ranks all the books on a scale of one (absolute beginner) through five (supergeek). About two percent of the books are singled out as our "hot favourites" and

another six percent are marked as "favourites", so you'll know which of these we'd recommend if you're new to the language.

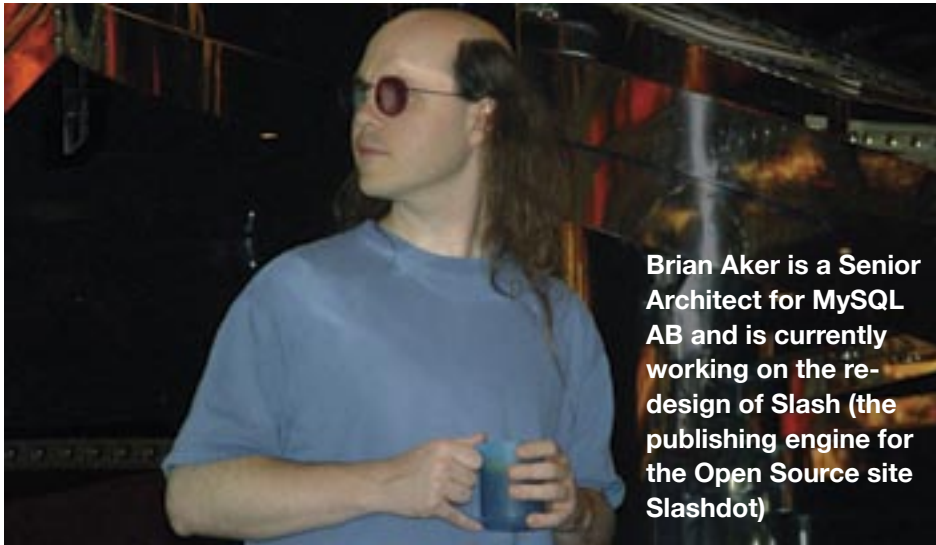
Because we use the technologies we teach to catalogue this collection, we can also show you how we've done it by coming on a course.

Afterwards, you'll be able to check the rankings on which books might suit you best to continue to learn. Links from each book description take you to Amazon (UK or USA depending on where you're located) so that you

can order the book on line. Our site even updates you, taking you to the current edition at Amazon if you've chosen a book that's been replaced.



Who looks after MySQL?



MySQL has become a major player in the database market. In 1999, O'Reilly published their first book on MySQL, but that book was shared with mSQL to give it a potentially larger reader base. Just 6 years later, I have 40 books that are all about MySQL, or cover it as their major topic.

In 1999, the subtitle of the O'Reilly book was "Databases for Moderate-Sized Organisations and Web Sites". Five years later, I listened to Brian Aker in October 2004 describing how he used the database to handle up to 500 transactions per second at Slashdot. And I listened to Monty Widenius and David Axmark describing how MySQL has moved from being an interesting little project into a product that compares favourably to the likes of Oracle.

MySQL is unusual in the Open Source world in that some 90% of its code is written by people who actually work for MySQL AB. With the directors based in Scandinavia, they recruited staff from around the world who have worked enthusiastically on the database. They've now got a geographically spread team, each living wherever he or she chooses.

The key philosophies of MySQL

It should be possible to download, try and install in 15 minutes. Too many open source projects have installation problems, tricks, etc., and that is NOT what they want for MySQL.

The primary intent is for Data Warehousing, handling up to terabytes of data, so it's fast, efficient, and mean in operation.

Design and support is such that you

should not need to upgrade unless you want the new features. Robustness and reliability rather than any form of rocket-science novelties.

Documentation is written with the code and IMPORTANT. And if a question is regularly asked it's added back into the documentation. "I don't like to spend time answering the same question every day, and I like to be able to refer people to a full answer" says Monty Widenius, the originator of MySQL.

MySQL uses extended subset of ANSI SQL:2003 (Similar to most SQLs). Added features are those desired by the community; things are not added if they make it slow or obscure or affect ease of use and reliability.

MySQL uses multiple storage engines (MyISAM tables for quick read/writes, InnoDB for rollback and commit requirements, Heap for memory-based) so that it has the right solution for each circumstance.

Oracle v MySQL

There's a big Open Source (MySQL) v commercial (Oracle) debate. Here are some interesting facts behind the mask.

The financial decision should be based on TCO (total cost of ownership) rather than on license fees. It's always nice to get "something for nothing" but newcomers fear that they'll spend more in the long term on supporting a product that's open source, and there are a number of scary rumours around.

Typically, licensing fees on a commercial

product account for some 25 to 35 percent of the total cost of ownership. So the license fee in a MySQL v Oracle decision is not as vital as you might think. However, TCO for a MySQL system tends to be anywhere from 75 to 90 percent less than the Oracle equivalent. I have no proof of these figures, but I can believe them because of:

- the ease of installing MySQL
- the ease of maintaining MySQL
- the lower DBA costs (training, salary, number of staff)

In justification of a commercial route, conservative buyers often talk about responsibility for code and support. If you want that, you can buy a commercial license for your MySQL. Towards the top of this market, a 24/7 contract with a one-hour response time will cost you \$50k per year, or if you can get that plus your own assigned engineer who knows your setup in great detail and can be reached at any time by pager. Regular visits aren't the sort of thing the typical customer needs, but it's something that the big NYSE companies require where their systems are trading billions.

Licenses available are:

ELA	Enterprise License Agreement
VAR	Value Added Reseller
OEM	Typically an integrator who includes MySQL in a commercial product
EULA	End User License Agreement
GPL	Open Source License

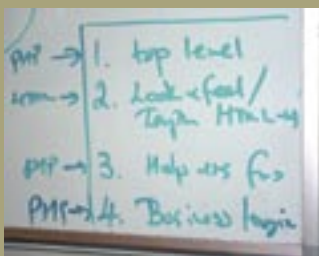
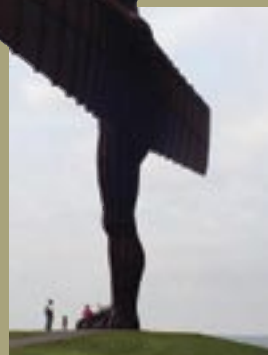
Typically, the GPL works fine for most end users, but it's largely untested in courts or law, and if lawyers get involved with bigger companies, they typically prefer to go for the EULA even though it costs. The general answer to the question "do I need to buy a license" is: If you're for free, then it's free. If you are using it in your proprietary system, then you need a commercial license.

EWeek benchmarks of Jan 2002 (old, but latest available) show performance and latency of MySQL to be virtually identical to Oracle, and well ahead of SQL Server, DB2 and others. "Inspection study shows code quality of MySQL is six times better than that of comparable proprietary code." This means that (for example) a recent decision as to whether to go for another Gamma release of 4.1 or to go production was obvious – it stayed Gamma.

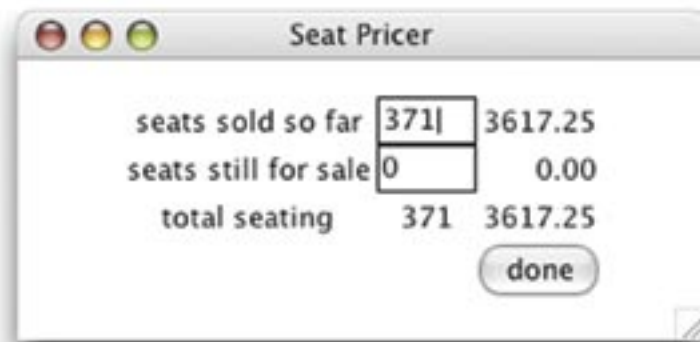
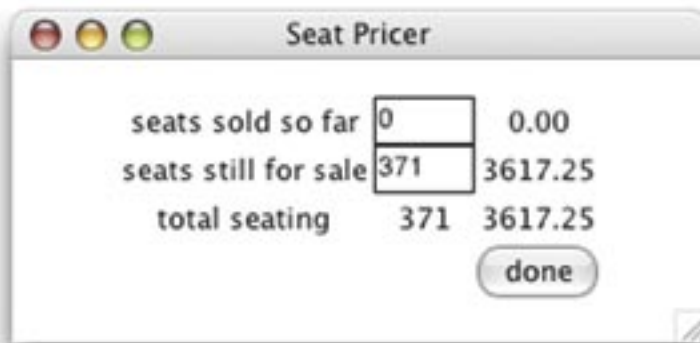
Brag Blog

Ah, it's a hard life, but someone's got to do it.

We have a lot to brag about! These pictures are taken either at our training centre or at their site, or during other business-related trips. It's been very difficult to choose the best, so we hope you enjoy just a mere sampling.



Seat pricing GUI in Tcl/Tk



The Tk Graphic User Interface provides an excellent tool for constructing small tools in just a few lines of code. Here, in 40 lines, is an example that defines 10 widgets and all the interaction you need to price out theatre tickets for a “house” with 371 seats at 9.75 each.

```
earth-wind-and-fire:~/feb05 grahamellis$ wish seats
Total of 120 seats sold for 1170.00
earth-wind-and-fire:~/feb05 grahamellis$
```

How can so much be done in so little code? In Tk, options like **-textvariable** allow the programmer to define an entry button or a label that doesn't need to be altered by his program when a variable changes, Tk automatically treats the label or entry box as a visual representation of the string the variable contains at any instance.

Clever, neat and efficient.

```
# Seat sales

set nsold 0
set total 371

proc calcseats altered {
    global total nsold navail
    global cttotal cavail csold
    if {$altered == "fromsold"} {
        set navail [expr $total - $nsold]
    }
    if {$altered == "fromavail"} {
        set nsold [expr $total - $navail]
    }
    set cttotal [format %.2f [expr $total * 9.75]]
    set cavail [format %.2f [expr $navail * 9.75]]
    set csold [format %.2f [expr $nsold * 9.75]]
}

proc finish {} {
    global nsold csold
    puts "Total of $nsold seats sold for $csold"
    exit
}

calcseats fromsold

label .lsold -text "seats sold so far"
entry .sold -width 6 -textvariable nsold
label .csold -width 6 -textvariable csold
bind .sold <Leave> {calcseats fromsold}

label .lavail -text "seats still for sale"
entry .avail -width 6 -textvariable navail
label .cavail -width 6 -textvariable cavail
bind .avail <Leave> {calcseats fromavail}

label .lttotal -text "total seating"
label .total -text $total
label .cttotal -text $cttotal
wm title . "Seat Pricer"

button .completed -text done -command finish

grid .lsold .sold .csold -sticky news
grid .lavail .avail .cavail -sticky news
grid .lttotal .total .cttotal -sticky news
grid .completed - - -sticky ne
```

Getting to our training centre

DIRECTIONS FROM THE M4

1. Start out at M4 Jct 17 (A350/A429/B4122), Wilts
Follow signs for Chippenham on the A350
2. Malmesbury Road Roundabout
At roundabout take 2nd exit following the A350
3. A350 Badger
At roundabout continue forward
4. Brook
At roundabout continue forward
5. Bumpers Farm (Junction with A420)
At roundabout take 3rd exit
6. Chequers (Junction with A4)
At roundabout take 3rd exit (signposted Poole, Warminster)
7. A350 Lackham
At roundabout take 3rd exit
8. A350 Junction with A365
At traffic signals forward
9. A350 Junction with A3102/B3107
At roundabout take 2nd exit
10. Berryfields roundabout
At roundabout take 2nd exit (signposted Devizes)
11. A350/A365 Junction
At roundabout take 1st exit (signposted Devizes)
12. B3112
At roundabout take the 1st exit onto Spa Road - B3112
13. The Spa
Turn right into cul-de-sac
14. Well House Consultants Ltd
Turn left in last gated entry at 404, The Spa, Melksham (4-storey Georgian)

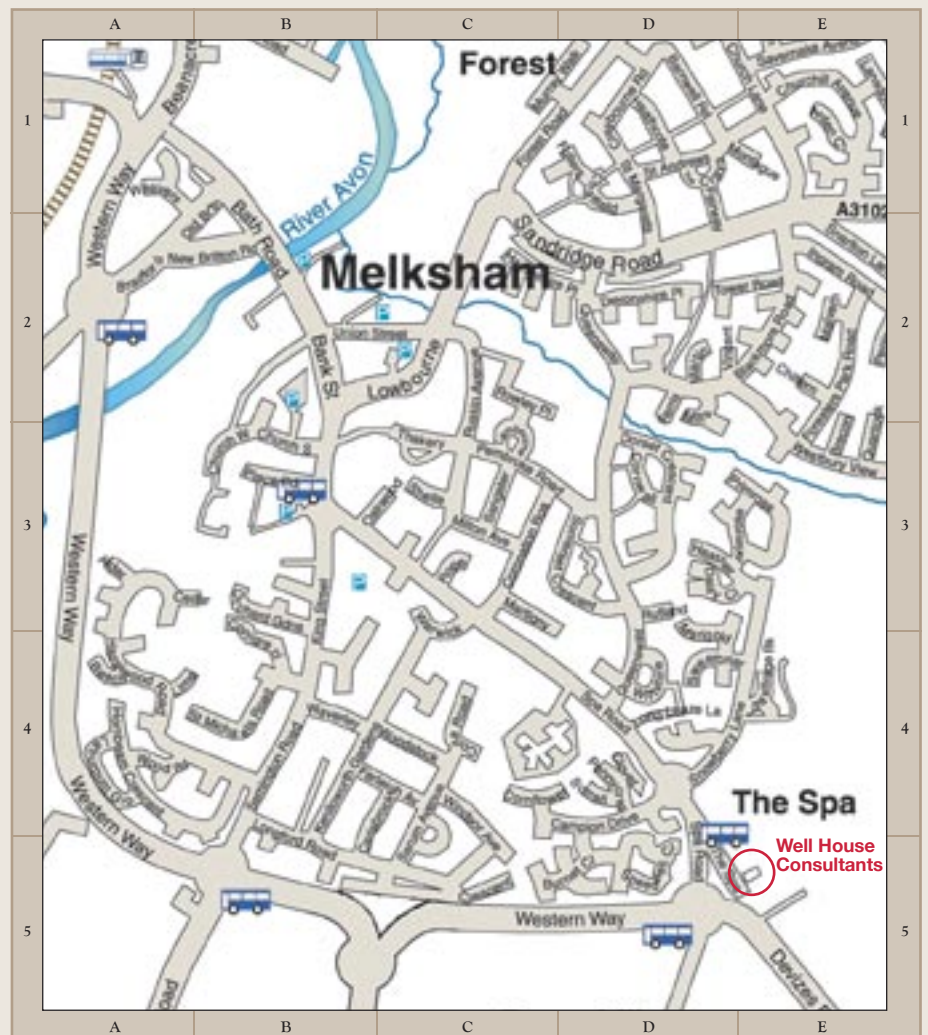
Estimated travel time: 20 minutes in non-adverse conditions

A350

A365

B3112

unclassified



You can do a lot of things

Python's while and else

All the languages we teach have **if** and **else** conditionals, and **while** loops. They're a necessary part of any language, accepted as something of a standard, and we teach them on all the fundamental courses.

Uniquely, Python allows you to add an **else** clause onto the end of a **while** loop. Why? So that you can define a block of code that's to be run upon an exit from the loop caused by the loop's condition becoming false, but that's to be skipped over if the loop exits through a break statement or because of an exception.

Here's an example. This program is going to loop, reading in people who are coming to your teenager's party. If you complete the data entry by pressing the "enter" key without entering another name, the **else** clause on the while is run and your party list is displayed. If you enter "parents" then the party plans are abandoned – you break out of the data entry loop and don't run the else block.

```
running = 1
friends = []

while running:
    said = raw_input("Who's coming to the party? ")
    if said == "parents":
        print "Party plans abandoned"
        break
    if said == "":
        print "Data entry complete"
        running = 0
    else:
        friends.append(said)
else:
    print "Having a party for ",
    print friends
print "program exiting"
```

```
earth-wind-and-fire:~/feb05 grahamellis$ python pywhi
Who's coming to the party? Tom
Who's coming to the party? Dick
Who's coming to the party? Harry
Who's coming to the party?
Data entry complete
Having a party for  ['Tom', 'Dick', 'Harry']
program exiting
earth-wind-and-fire:~/feb05 grahamellis$ python pywhi
Who's coming to the party? Lisa
Who's coming to the party? Leah
Who's coming to the party? parents
Party plans abandoned
program exiting
earth-wind-and-fire:~/feb05 grahamellis$
```



Naming an object

I call my father "Dad" but everyone else calls him "Norman". Same person, different name.

If Dad puts on the hat that he's got from the Christmas cracker before he eats his Christmas pudding, then everyone else sees that Norman has put on his hat before eating his Christmas pudding. Two names for one and the same person.

The same principle applies to the copying of objects in Python (and Java and PHP 5 and other object oriented languages); an assignment statement is just the addition of an extra name to the same object.

Let's see an example in Python:

```
pie = ["apple", "pear", "strawberry"]
pudding = pie
pudding[1] = "apricot"
print pudding
print pie
```

```
earth-wind-and-fire:~/feb05 grahamellis$ python obcop.py
['apple', 'apricot', 'strawberry']
['apple', 'apricot', 'strawberry']
earth-wind-and-fire:~/feb05 grahamellis$
```

By changing "pudding" I've also changed "pie".

In contrast, if I copy a list in Perl (and in PHP 4, Tcl and other languages which are not 100% object oriented), I duplicate the data. Example in Perl:

```
@pie = ("apple", "pear", "strawberry");
@pudding = @pie;
$pudding[1] = "apricot";
print "@pudding\n";
print "@pie\n";
```

```
earth-wind-and-fire:~/feb05 grahamellis$ perl obcop.pl
apple apricot strawberry
apple pear strawberry
earth-wind-and-fire:~/feb05 grahamellis$
```


with a little bit of Python

Using Python as a test bed

Python's a great language for writing "test bed" applications, things which start small with a few lines of experimental code and then grow. Here's an example script to answer some specific questions concerning access to our course description directory on our web server, where we get a new log file several megabytes long each day and it can be hard to see the data you might be looking for.

Here are three sample lines from our logs ... that's out of 700,000 lines in one random month:

```
cleveland.directrouter.com - - [01/Jan/2005:23:57:05 -0800] "GET /fotos/lenew.jpg HTTP/1.0" 200 1241 "-" "-"
cr02.bloglines.com - - [07/Jan/2005:00:22:56 -0800] "GET /horse/index.rdf HTTP/1.1" 304 0 "-" "Bloglines/2.0 (http://www.bloglines.com; 1 subscriber)"
1c4.learn.ac.lk - - [07/Jan/2005:00:23:04 -0800] "GET /course/mqfull.html HTTP/1.0" 200 30843 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows 98)"
```

The whole thing can be done with a simple script written in a few minutes

In that example, the "mqfull" on the last line was generated as a result of the last of the sample records listed above. Enter Python.

```
#!/usr/bin/python
```

```
""" Course page lister.
Looks at all file starting with ac and
treats it as an standard web log file.
Reports on all domains that
have visited a course description page"""
```

```
import os
import re
from hname import *
```

```
shorttable = {}
ctable = {}
lookfor = re.compile("GET\s/course/([a-z]+)")

for filename in os.listdir("."):
    if not filename.startswith("ac"):
        continue # eliminate files that are not logfiles
    for line in open(filename).xreadlines():
        parts = line.split(" ")
# How much of the name / IP address do we report?
        host = hname(parts[0])
        dpsumm = host.getShort()
# Count the host usage and log any course files
        shorttable[dpsumm] = 1 + shorttable.get(dpsumm,0)
        gotten = lookfor.findall(line)
        if (not gotten): continue
        ctable[dpsumm] = ctable.get(dpsumm,"") + " " + gotten[0]
```

```
def byhits(one,two):
    global shorttable
    return shorttable[two].__cmp__(shorttable[one])
```

```
visitors = ctable.keys()
visitors.sort(byhits)
```

```
for browser in visitors:
    print browser,ctable[browser],shorttable[browser]
```

I'm interested in knowing how many lines related to each group of hosts and which particular page(s) they asked for in the /course directory. Here's part of the results:

```
hinet.net  alfull alfull alfull ajfull 441
hispeed.ch alfull ctc alfull 438
on.net     lafull alfull alfull alfull jxfull 434
auna.net   ajfull yp atfull tkfull tbfull 422
netcabo.net atfull mqfull pofull pofull 420
ac.lk      pl mqfull 415
```

A cracker in a snap

Make your own Christmas cracker by using empty rolls of wrapping paper. Buy novelty snaps at any craft shop. Arrange trimmed cardboard tubes as shown, placing the paper snap through all three. Insert novelty items in the middle larger tube and roll all three tubes in the remaining scrap Christmas wrapping paper.



To reuse the code across several programs to work out how much of a computer name or IP address to display, or how to group the hits, put the code into a separate file that can be read within the application. In other words, use a class to avoid repetitive coding and make more efficient use of coding time. The class:

```
class hname:
    def __init__(self,text):
        self.dparts = text.split(".")
        if self.dparts[-1].isdigit():
            self.havename = 0
        else:
            self.havename = 1
    def getShort(self):
        if self.havename:
            if self.dparts[-1] == "uk":
                dpp = self.dparts[-3:]
                if dpp[0] == "demon":
                    dpp = self.dparts[-4:]
            else:
                dpp = self.dparts[-2:]
        else:
            dpp = self.dparts[:]
        return ".".join(dpp)
```

From our cabin looking up, as we leave Venice, we all marvelled at the colours of day's end. It was too beautiful to not be struck by, and too perfect to not capture on film. As it was, a fitting beginning to many more days of picture taking...

...and many more days of being awestruck by the company we had on the boat.

Cruising with the Geek gods



It's a huge decision for an organisation to turn off its revenue stream for a week or two to attend such an event, then to put that stream into reverse by spending on flights, hotels and transfers, the cruise, and the conference itself. But as trainers in these technologies, we need to understand not only what we teach but also the more advanced application of what we teach, and the philosophies of the movers and shakers behind the technologies. Only with this insight can we teach you not only about what the products do today, but about WHY they work like that, and where they'll be going in the future.

continued on page 12





ROBERT SPIER



RANDAL L. SCHWARTZ, Geekcruises.com



RANDAL L. SCHWARTZ, Geekcruises.com

Featured: (far left) David Axmark watching the sunset; (top) Brian Aker talks on MySQL; Casey West and Robert Spier watch as Perl creator, Larry Wall, is welcomed by Neil Bauman; Graham takes notes; (middle) Dave Cross at a market stall; photographer Kevin Gilbert; the enrapt audience; Allison Randal shares Perl 6 developments; Marty Pauley gives Perl points; PHP founder, Rasmus Lerdorf, and Neil Bauman; (bottom) MySQL's Monty Widenius; Randal Schwartz; Ted Ts'o and Stacey Quandt.



RANDAL L. SCHWARTZ, Geekcruises.com



ROBERT SPIER



RANDAL L. SCHWARTZ, Geekcruises.com



RANDAL L. SCHWARTZ, Geekcruises.com



RANDAL L. SCHWARTZ, Geekcruises.com



RANDAL L. SCHWARTZ, Geekcruises.com



RANDAL L. SCHWARTZ, Geekcruises.com



RANDAL L. SCHWARTZ, Geekcruises.com



Geek Cruising

continued from page 10

Sunday afternoon, Venice, and we joined Costa Line's Costa Mediterranea for a Geek Cruise on the Mediterranean. A glorious sunset over the city as we sail past St. Peter's square and smile at the chap clicking away on the balcony of the cabin just above ours who turns out to be one of the co-founders of MySQL AB. And the scene is set for an unusually intensive and mixed week even for us.

At sea, MySQL swell kicks off. Hearing Monty Widenius and David Axmark talking about their "baby", the listener comes to appreciate just how technically astute these gentlemen are, and just how commercially aware they are too. Even though they're in the Open Source world – Eric Raymond's Bazaar rather than his Cathedral – one leaves with the background knowledge that here's a product that can and *will* stand up against the traditional giants such as Oracle and SQL Server. Speaking personally, it was a privilege to listen and it equips me so much better to talk about MySQL when I'm training. I can talk not only of the mechanisms, but the philosophies behind them, too.

Katakolon, Greece. The ship docks, and the passengers transfer to coaches for the drive to ancient Olympia. Like all the other destinations on this cruise, Lisa and I mark Olympia as somewhere we must come back to when we have more time, and it was somewhat surreal to see the very arena where the original Olympics were held, at the same time knowing that mixed in amongst the crowds are some people whose names we mention with great reverence and whom we would never have expected to meet a few years back.

From Greece, onwards to Turkey. A day's cruising with the chance to sample ship foods, gamble and sunbathe. But many of those activities are for others. We're up at first light, in the bar where there's a wireless Internet hotspot specially set up for the Geek cruise and not available for the other passengers. Lecturers and attendees mix as they check their email, upload pictures, and generally stay in touch with base before another intensive day. With any "multi-track" conference, there will be schedule clashes even with the best possible planning. Today, Rasmus Lerdorf (the man behind

continued on page 14

Well House Consultants, Ltd.



(top) the Grand Canal in Venice, our departure city, taken the night before the cruise; (below) "But is it art?" interior of our ship; (right) Graham leads the tour group in Turkey; (bottom) the library at Ephesus



GRANT MCWILLIAMS





The Schwartzian Transform

This Rumplemint, Baileys and Jaegermeister concoction quickly became a mascot of sorts. “This drink was dubbed a ‘Schwartzian Transform’ by my fellow cruisers. Notice the map-sort-map layers,” noted its namesake, Randal L. Schwartz.

The original Schwartzian Transform was named by Tom Christiansen during a Usenet discussion that became the “FMTEYEWTK (Far More Than Everything You Ever Wanted To Know) about

Sorting” in CPAN’s documentation section because it was first pointed out by Randal. Mark-Jason Dominus in his “Perl Paraphernalia” summarises: “Construct a data structure that has both kinds of information, names and dates in it, sort by the appropriate one, and then throw it away again.”

Learn more about the Schwartzian Transform in the third edition of “Programming Perl”, ISBN: 0-596-00027-8.



(left) pillars at the Topkapi palace in Istanbul, built by the Ottomans in the 15th century, and much enlarged since; (right) our very patient waiter on board the Costa Mediterranea, Enrico Golosino; (below) Dubrovnik Croatia [commented photographer Grant McWilliams “I had to run along the wall on a slippery wet rock surface to get that picture because I only had about 10 minutes to see everything.”]; (bottom, left) The Suliman mosque, one of many in Istanbul; (bottom, right) the gala night.

PHOTOS BY GRAHAM AND LISA ELLIS, UNLESS OTHERWISE NOTED



GRANT MCWILLIAMS



Latest Versions of Open Source Software

as of 1 March 2005

Here's our regular table of current releases. As well as the ones listed below, amongst new and ancilliary products that you might also like to have a look at are:

- MySQL Administrator and MySQL Query Browser - from the MySQL team
download from <http://dev.mysql.com>
- Pugscode interpreter for Perl 6 if you would like some pre-exposure to the language
see <http://www.pugscode.org>
- Expect for Windows - a licensed product from Active State
see <http://www.activestate.com>

Perl	5.8.6 Stable	www.perl.com
	5.9.1 Development see http://www.cpan.org for additional modules	
Tcl/Tk	8.4.9 Stable	www.tcl.tk
	8.5a2 Alpha release	
Expect	5.43.0 Stable	expect.nist.gov
PHP	4.3.10 Stable	www.php.net
	5.0.3 Stable see http://pera.php.net and http://pecl.php.net for PHP extensions Note: PHP4 and PHP5 are not 100% compatible. Many PHP programmers aren't even aware of the copying quirks of the old object model and, therefore, the majority of PHP applications will work out of the box, or with very few modifications, under PHP5.	
Python	2.4 Stable	www.python.org
	see http://www.vex.net/parnassus/ for Python resources	
MySQL	4.1.10 Generally Available	dev.mysql.com
	5.0.2 Alpha	
Apache Tomcat	5.0.28 Stable	jakarta.apache.org/tomcat httpd.apache.org
	5.5.7 requires JRE version 5	

Geek Cruising

continued from page 12

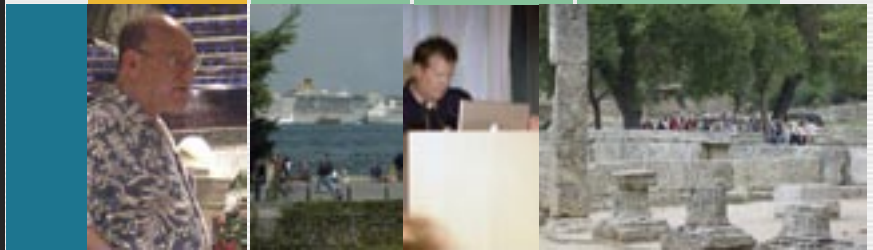
PHP) has kindly offered to run an extra session of "Do you PHP?" and perhaps a dozen of us listen enthralled. With PHP, you can write superb code or a dog's dinner of a solution, and Rasmus gave us his guidelines for the former, in addition to a great deal else including a few answers to "why" questions about the language design that had me wondering, and a view to where he sees PHP going. Rasmus, THANK YOU for that extra session; you don't know what a difference it made.

Turkey is "where East meets West". And there's a feeling of really *being* somewhere as you stand in the grounds of the Topkapi palace in Istanbul, in Europe looking out over the Bosphorus to Asia. And so much to do during the boat's call. On a Geek Cruise, though, we had too much else to do as well, and elected to spend the morning catching up on writing up notes, then relaxed on a half-day tour that took in some of the streets of the old city, the blue mosque, and the Topkapi. We could easily have spent several more days in Turkey, but then we could easily have spent several more weeks on the cruise.

Across the Adriatic, and Allison Randal and Larry Wall are telling us about Perl 6. Perl 6 is coming along, and the foundations are there. For us, it's moving from a spectre that's somewhere in the future towards an exciting prospect that's getting much closer, and Allison and Larry, both of whom have been working on the design and the code, gave us a great deal to think on (and mostly agree with ;-)), so much so that we chose to stay on board and digest their inputs when we docked for a few hours in Dubrovnik, Croatia.

And so, a final evening on board Costa Mediterranea. Enjoying a glass of wine and a technochat with a high energy physics computer administrator from Chicago, a senior member of the White House press corps who knows George Bush on first name terms, and other geeks who came from Finland and Northern Ireland, from Australia, Canada and the USA. Truly a memorable week that we'll look back on long after other memories of 2004 have faded away.

Will you be reading of yet further adventures on board Geek Cruises in the future? We're eyeing the next one now!



www.geekcruises.com

Listen to some of the big names in the development of programming languages and systems software, have a great holiday to some spectacular destinations, and meet with other IT professionals in your field. Sound too good to be true? Well, it's the formula that's successfully applied by Neil Baumann of Geek Cruises. He's now organised and personally run more than 20 events such as "MacMania" and "Photoshop Fling". Destinations include Alaska, Hawaii and the Caribbean, and Neil has just completed his first European Geek Cruise with "Linux Lunacy", "MySQL Swell" and "Perl Whirl" cruising out of Venice to Greece, Turkey and Croatia.

A sampling of upcoming cruises:



MacMania 3.5 and Northern Light

• Baltic Sea, June 30 - July 10, 2005

Linux Lunacy V and Perl Whirl '05

• Caribbean, October 2-9, 2005

Chess Moves

• Eastern Caribbean, January 14-21, 2006

Mac Mania IV and Photoshop Fling III

• Mexican Riviera, February 4-11, 2006



We learned a lot, saw a lot and made many friends. We were reluctant to go home, yet anxious to practise our new skills. It was exhilarating and exhausting. But definitely to be repeated.



Turn your site's URLs into

Have you ever done a search and when you've tried to visit the page that the search engine has found, you just got an error 404 "File not found"? It's probably happened to us all. But with careful web site management and use of tools such as PHP, it shouldn't happen too often. The URL that brings a visitor to your site should be telling you what your user's looking for and not be a hurdle that your visitor must jump in order to access your data.

How can you achieve this?

First easy tip

When you wish to withdraw a URL, don't simply delete the html file concerned. Rather, replace it by a message that declares the page you were looking for has been replaced, and provide a link on to a new location. You'll find such a link on our web site at <http://www.wellho.net/course/py.html>.



Make sure that the new replacement page at the old URL has nothing useful in it that the search engines will keep re-indexing, and in time you should be able to delete it.

Second tip – a little trickier

Configure your web server so that any potential "404" errors get passed on to a script that can interpret them, and offer your visitor something useful.



Configure the .htaccess file with a line like: `ErrorDocument 404 /diverts/d404.php` to achieve this.

Now, analyse what was wrong with the URL the user entered. Does a file with the correct name exist, but with a ".html" rather than a ".htm" extension? Has the user entered a correct file name, but with capital

letters in the wrong place? Does a file of the correct name exist, but in a different directory on your web site? If you find any of these cases do apply, you can divert the user with a few words of warning. Try an example at <http://www.wellho.net/catkin/tb.html>:



The "trick" in our PHP is the call: `header("HTTP/1.1 200 OK");` which tells the web server (and thus the browser) not to treat this as an error at all.

Want to go a step further?

On our web site, we've extended our diversion script to do a web site search if it really can't make out what you're looking for. It takes words from the page name and sees if and where they appear on our web site, e.g. <http://www.wellho.net/iwant/charlie.html>



will offer you a whole menu of links to our cat Charlie.

And some further steps

You can recognise spelling errors and offer sound-alike words. PHP has excellent facilities for searching for similar words,



and can do so within a request even on our web site of 5,000 pages, aided by a MySQL database that we refresh when the web site changes. Here's a search for Liferpool:

<http://www.wellho.net/net/search.php4?search=liferpool>. Part of the request your web server receives is the URL from which your page was linked. Even if you're not going to use this to help, mine your site to provide useful extra data for your visitors, you can log it in your broken link script so that you can find out later where people are coming from to get 404s. If you're clever, you can analyse the referer on all your pages and provide that little extra to your visitor.

Try searching Google for "Perl" and "Oban". The results will give you the URL: <http://www.google.co.uk/search?hl=en&q=perl+oban&btnG=Google+Search&meta=>



and the top link (as I write this, but Google rankings change!) brings you to:

<http://www.wellho.net/solutions/melksham-on-the-oban-road.html>



Can you spot the subtle difference between the two pages? The page when linked to from Google knows this, and it knows you're interested in "Perl" and "Oban" so it gives you a few extra links.

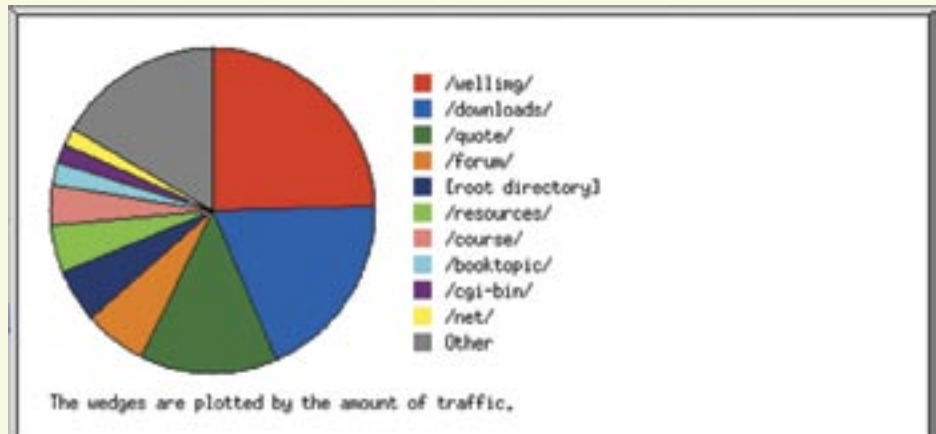
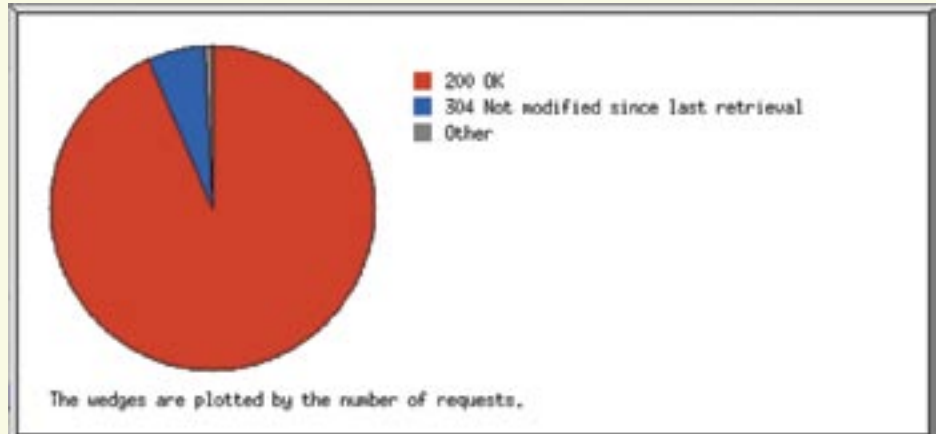
Using the techniques outlined in this article, our complex and dynamic web site returns very few errors. Looking back as I write this, we've returned about 1000 error "404"s out of over half a million accesses in the last 28 days. Of course, being a perfectionist, receiving an error rate of 0.2% still didn't satisfy me so I still checked my logs and discovered that 820 of the errors were caused by a search engine robot that has a bug as it tries to follow certain links.

your customer's friends

Easy-to-read graphs show accesses at a glance

- Red area – good requests which we fulfilled correctly
- Blue area – request for page which hadn't been changed (browser told to use previous copy)
- Grey area – everything else, including pages not found.

Most web servers provide access logs. They're useful for a variety of reasons, including monitoring web site usage, security and marketing purposes. But access logs are huge; for example, we get more than 200 of accesses Mbytes per month. Open Source analysis packages such as Analog will analyse this data for you. We run such an analysis daily, and update a page of pie charts that let us spot unusual activities and trend changes very quickly. Further outputs from analog are



What are people looking at on our web site?

available to Well House Consultants' staff, and those provide us with much more detail. Access is via password-protected pages; we've also got access to the raw log files and the ability to run the log files through Perl or Python scripts to help us look at some facet or other of access to our web site if the need arises.

Web site daily information can be seen at <http://www.wellho.net/resources/accesspie.html>. Email traffic information is also on our web site at <http://www.wellho.net/resources/email.html>.

Embedded Layer

The mountain zebra's stripes are thin - over a few inches, it turns from black to white and back to black again. Although its colours are stark, out in the wild the stripes merge to a greyneess that forms a masterful disguise.

If you're writing web page code using an embedded system such as ASP, JSP or PHP, it's very easy to end up with a page that looks like a zebra - in other words, with a page that switches from program code to HTML and back again so rapidly that the page as a whole is difficult to visualise and maintain. And the only person capable of maintaining such a page is going to be someone who's an expert on both the programming language involved and also on HTML and web page design.

With good initial planning, with most embedded systems you can divide your application into a series of areas, each of which contains a predominance of code of one type, and can be maintained easily by an expert on that type. A good example is the "4 layer model" that Rasmus Lerdorf described during his "Do you PHP?" lecture on our Geek Cruise.

Layer one

The top layer a short and simple file that pulls in all the other layers via require or include calls.

The top layer file is the only one that you place in the document tree on your route server, as it's the point through which users access your web application.

Layer two

The business logic code written in PHP (or Java or VB with other embedded systems) that performs the actual work of your

application - perhaps checking databases for matching articles, posting new text, updating stock levels, etc.

There's little or no web-specific code in the business layer logic, and the code is arranged into a series of functions which can be run as required from within other layers of the model, and perhaps externally too if you want run some of your web site admin tasks through command line scripts or crontabs.

Layer three

Web helpers more PHP code, but this time code that relates to the web display of your application. In here you'll include functions to make your form boxes "sticky" and to generate pull-down menus, functions to validate user inputs, login functions, session handlers and the like. If you plan your web site carefully, your web helper layer can be shared across all your web applications. After all, why re-invent the wheel?

Layer four

web page template the actual HTML of the page, with embedded calls to the business logic and web helper layers. These calls should be very short - the philosophy of PHP is that it provides extra tags, and the four layer model works best if you use the web template layer in this way.

Writing a PHP application is much more than just writing a piece of code to provide some functionality. It's also about writing something that runs well, looks pretty, handles user error well, and is robust, user friendly (and obvious in operation) and maintainable. The four layer model will help you write your application to meet these requirements.

Learning Servers

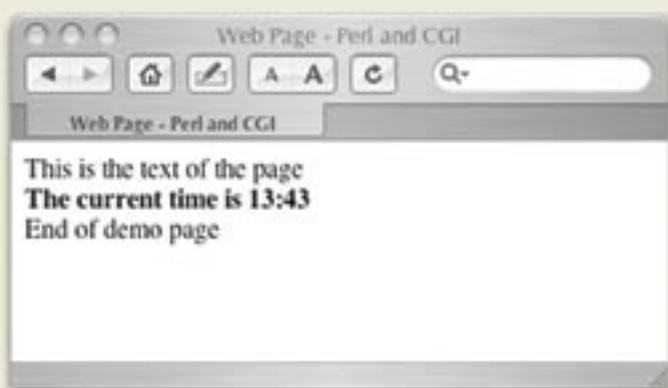


HTML++ simple, yet mind-blowing

I've invented a language. It's called HTML++ and I talk about it during Java and PHP courses. That's because PHP and Java-

ServerPages (JSPs) are forms of HTML++.

If you're writing a program that you're going to make accessible through a browser, there are two ways of doing it. You can write a program that generates HTML, or you can write an HTML page with extra tags that call up programatic elements. That latter is what I've christened "HTML ++". Readers who are familiar with PHP will consider this very basic stuff, but readers who are new to the web, or have only used programs (e.g. Perl with CGI, or Servlets) may find that it's a novel and mind-blowing concept. It is, and it works really well.

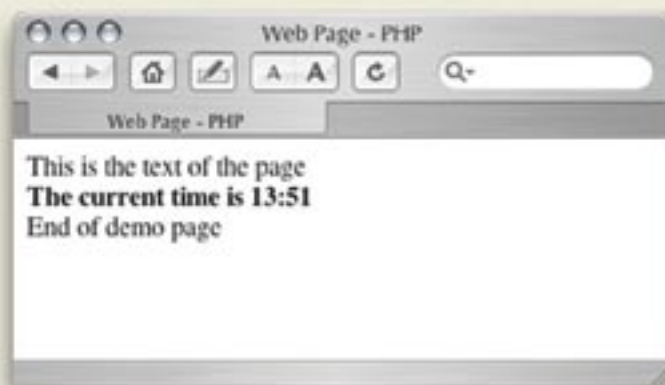


Example - Perl with CGI

```
#!/usr/bin/perl

print "Content-type: text/html\n\n";
@now = localtime();
$tform = sprintf("%02d:%02d",@now[2,1]);

print <<"END";
<html>
<head><title>Web Page - Perl and CGI</title></head>
<body>
This is the text of the page<br>
<b>The current time is $tform</b><br>
End of demo page
</body>
</html>
END
```



Example - PHP

```
<html>
<head><title>Web Page - PHP</title></head>
<body>
This is the text of the page<br>
<b>The current time is <?= date("H:i") ?></b><br>
End of demo page
</body>
</html>
```

Simplifying complex joins

In SQL, you should normalise your data into separate tables to avoid the need to:

- repeat the same data across many rows
- put a whole list of values in a single cell

You'll then join the table back together within commands such as `select` as you retrieve database contents.

A normal JOIN returns you all records where a field in a record in one table matches a field in a record in another table.

A LEFT JOIN returns an extra row for each record in the left-hand table which does not match any row of the right-hand table. In other words, it also includes what we might call "orphan rows" in the left-most table.

Left join

Here's a left join using a table of buildings and residents that live there:

```
select * from bdg left join
res on bdg.bid = res.bid ;
```

name	bid	person	bid	rid
404	1	Graham	1	101
404	1	Lisa	1	102
405	2	NULL	NULL	NULL

Joining three tables

Here's an example of a complex left join, or one that joins three or more tables. An extra table of domain names has been added to the two tables already defined, with some orphan records to help you see what's happening.

```
select * from (bdg left join res on bdg.bid = res.bid)
left join dom on res.rid = dom.rid;
```

name	bid	person	bid	rid	domain	rid	did
404	1	Graham	1	101	www.grahamellis.co.uk	101	201
404	1	Graham	1	101	www.sheepbingo.co.uk	101	202
404	1	Lisa	1	102	NULL	NULL	NULL
405	2	NULL	NULL	NULL	NULL	NULL	NULL

Three way joins

looking for incomplete records

Should we wish to run a report on orphan records only, we can test for NULL fields in fields that could not otherwise have a null value.

```
select * from (bdg left join res on bdg.bid = res.bid)
left join dom on res.rid = dom.rid where dom.rid is NULL;
```

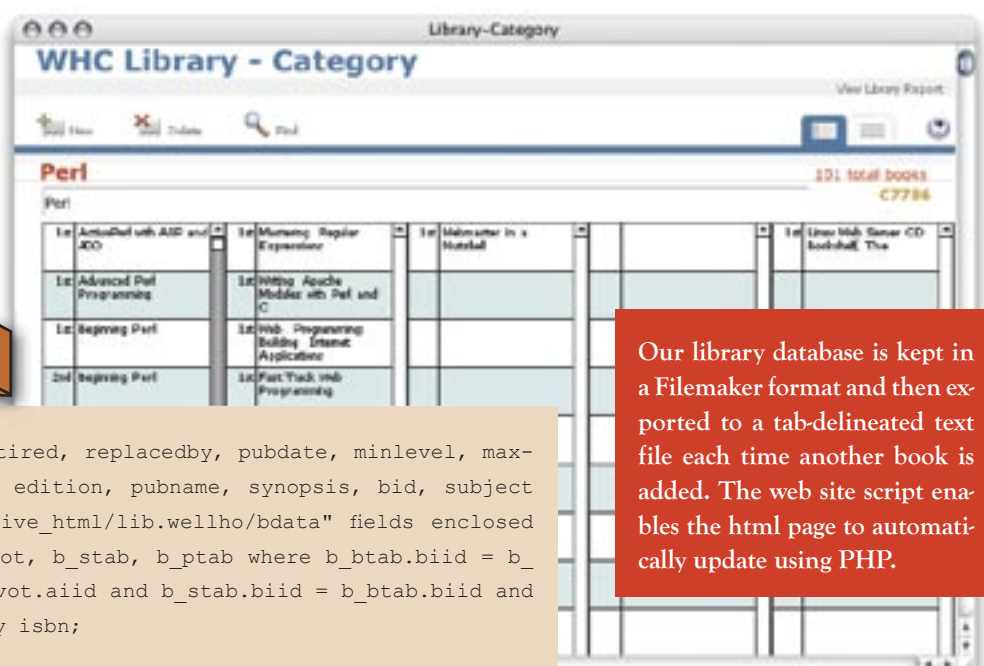
name	bid	person	bid	rid	domain	rid	did
404	1	Lisa	1	102	NULL	NULL	NULL
405	2	NULL	NULL	NULL	NULL	NULL	NULL

Our library is stored in a MySQL database comprising 5 tables:

- Books
- Authors
- A "pivot" table to provide a many to many mapping between books and authors
- Publishers
- Categories

An example of a join using the five tables:

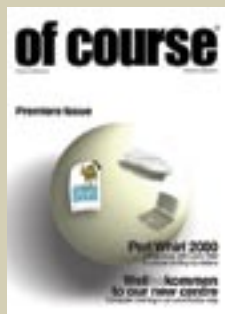
```
select isbn, title, favourite, retired, replacedby, pubdate, minlevel, max-
level, fullname, synopsis, jpeg, edition, pubname, synopsis, bid, subject
into outfile "/Library/WebServer/live_html/lib.wellho/bdata" fields enclosed
by '"' from b_btab, b_atab, b_pivot, b_stab, b_ptab where b_btab.biid = b_
pivot.biid and b_atab.aiid = b_pivot.aiid and b_stab.biid = b_btab.biid and
b_ptab.piid = b_btab.piid order by isbn;
```



Our library database is kept in a Filemaker format and then exported to a tab-delimited text file each time another book is added. The web site script enables the html page to automatically update using PHP.

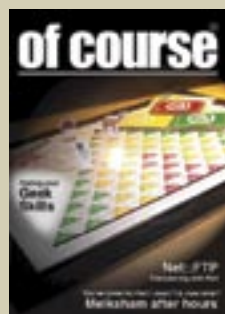
Back Issues

All back issues are available by post to UK addresses at no charge. Please phone or email us for a copy or two.
Copies of recent editions are also available to download from our web site: <http://www.wellho.net/net/ofcourse.html>.



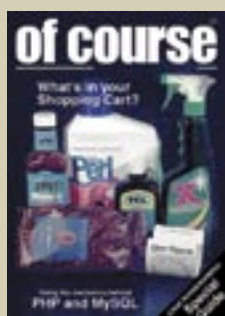
Issue 1
Winter 2000/01
8 pages
Perl Whirl 2000 - An Alaskan Adventure
Wellhokommen,
Bien Venue
Tcl/Tk and Expect
Handling Huge Data
- Sample script
Book Review - The "New Camel"

The London and Dublin connection
Perl for Larger Projects



Issue 2
Spring 2001
16 pages
Quiz - Perl, Java and Tcl
Using FTP from Perl
In and around Melksham
Blistering Speed right to the laptop
What's the latest software version?

Is Open Source right for you?
Focus on Perl Standards
Web logs and market research



Issue 3
Autumn 2001
16 pages
Behind the mechanics of PHP
Java threads for a simple chat room
What is XML?
Relating to MySQL
Access to Gaelic language

Wireless networking
Sessions and State
What's the latest software version?
On the Oban road
Perl 6 - State of the Onion

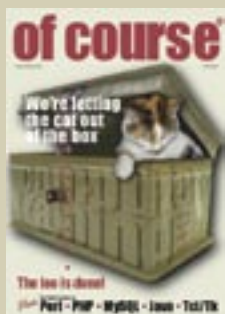


Issue 4
Winter/Spring 2002
16 pages
Keeping data secure
Opinion - IT making the world smaller?
Are you under attack from Cod-ered?

Extracting text from Microsoft Word documents in Perl
PHP - Improved security in release 4.1
Daemons - a story of gods and mortals
Cookies - European law and regulations
The Parrot virtual machine
Case Study - MPC Data embed MySQL in tailored software solution
More Dynamic than an array
Expect / Tcl application
Book review - "The Java Cookbook"



Issue 5
Autumn 2002
16 pages
RegEx
Tcl up to date
Learning Perl as a first language
YaBB
Tk - updates of your screen
Portable PHP code
Using PHP in stand-alone applications
Running scripting languages fast
Open Source Bulletin Board



Issue 6
Spring 2003
16 pages
Perl 6 introduction
Public sector data
When size matters - MySQL text
Data Mining
Setting up online e-commerce
Spring Cleaning
10 relevant projects from SourceForge
Coding tips
Apple get it right to the core
Introducing our loo
Sample application in Perl
Data Protection act - buyer beware
Setting up Mac OS X
A rogue's gallery of recent students



Issue 7
Autumn 2003
24 pages
PHP sticky boxes
Perl 5, Ponie, Perl 6
SQLite
Python and Jython
Opentalk - Tcl, Perl, PHP, Java, MySQL
WHC Library
Broadband update

Can a high speed connection be too fast?
Course materials
Open Source moves forward
Data structure design - link or copy?
Deployment training - new growth area?
Latest stable versions of open source
Course schedule and calendar
Tailoring your web site
Not only faster, but bigger too.



Issue 8
Spring 2004
24 pages
PHP: What do the letters stand for?
Choosing which OS to use
How do I learn OO design?
I refuse to wear THOSE shirts!

What's a polecat?
The hidden costs over a lifetime
Quality matters
Who teaches the teacher?
Python on either side of the pond
Broadband Access for the Traveller
Logging in to the BT Openzone service
Dining out in Melksham
Brag Page
Using Objects in Perl
References and variables in Perl
What do you put into a file?
What defines quality software?
Getting to the core price of Fedora
HTTP Developer's Handbook



Issue 9
Autumn 2004
24 pages
Legal aspects of a web site
Short tags in PHP
The speed of three
A numbers riddle
PHP5: Should you switch?
Latest Versions

Making a web site visible
Responsible for a web site?
Mint under the pillow
Handling huge data
Dynamics of web pictures
Extreme Programming
Back Issues
Schedule
Calendar
OOP DeMYSTiFieD

Schedule

Perl

Perl Programming

5 days £1210

Covers the major features of the language and their use for newcomers. This course now runs in conjunction with Learning to Program in Perl and covers version 5, with footnotes where appropriate to tell you about Perl 6.

20-24 June 2005

26-30 September 2005

12-16 December 2005

Using Perl on the Web

2 days £550

This advanced course takes the Perl programmer through to making full use of Perl on the Web. CGI, mod-Perl, standards. Sessions, cookies, content management, etc.

21-22 July 2005

17-18 November 2005

Perl for Larger Projects

3 days £770

This advanced course takes the Perl programmer through the facilities provided in the language which make it particularly suitable for larger projects. Objects, huge data, SQL databases, XML, efficiency and other topics.

18-20 July 2005

14-16 November 2005

Perl Extra

1 day £330

Subjects not covered on regular public courses.

telephone or email for details

PHP and MySQL

Technology for PHP

1 day £330

Overviews the technologies such as the Web, programming and SQL that you'll need to make good use of PHP. A lead-in day for our PHP Programming course.

4 April 2005

22 August 2005

31 October - 3 November 2005

PHP Programming

4 days £990

PHP is a scripting language used to add executable content to Web pages. This course covers the fundamentals of programming in PHP and Web-related issues such as linking a series of forms together into a session, maintaining state if multiple users are on your site at the same time, cookies, security and other matters. We also cover PHP's interfacing to relational databases in some depth, with a particular study of MySQL.

5-8 April 2005

6-9 June 2005

23-26 August 2005

Object Oriented Programming with PHP

1 day £330

This advanced PHP course introduces you to object oriented programming principles, and teaches you how they are implemented in PHP.

10 June 2005

4 November 2005

MySQL Relational Database

2 days £550

Structure, installation, administration, database design and programming.

11-12 April 2005

13-14 June 2005

1-2 September 2005

21-22 November 2005

Apache and Tomcat

Deploying Apache and Tomcat

2 days £550

How to source, install, configure and support Apache and Tomcat for the deployment of Java web applications.

15-16 June 2005

3-4 October 2005

Python

Python Programming

3 days £770

This course starts at the fundamentals of Python, then goes on to study Python's collections. Functions, modules and packages are then covered, as is the use of and creation of data types (objects) in Python. A short overview of OO principles is included,

as well as an introduction to the development environments and other resources available.

4-6 May 2005

5-7 October 2005

Tcl

Tcl Programming

3 days £770

Fundamentals of the Tcl programming language including Expect, libraries and regular expressions. Further topics covered on the course include writing your own commands (procs), packages, libraries, namespaces and other - you'll leave the course with a thorough grounding in Tcl.

9-11 May 2005

10-12 October 2005

Tcl - the Tk Toolkit

2 days £550

Covers widgets, geometry managers and event handlers of the Tk GUI so you can write and maintain graphic applications.

12-13 May 2005

13-14 October 2005

General

Regular Expressions

1 day £330

Dedicated to pattern recognition and matching, dependent on recognising sequences of characters.

3 May 2005

31 August 2005

Deploying LAMP -

Linux, Apache, MySQL, Perl / PHP

4 days £990

Installing and supporting applications under open source technologies.

31 May - 3 June 2005

19-22 September 2005

Or, take them individually:

Linux Basics day 1 (£330)

Linux Administration day 2 (£330)

Linux Web Server days 3&4 (£550)

Prices valid for bookings confirmed by 31 September, 2005. See Web site for US dollar and Euro prices. Prices do not include VAT.

Calendar

Public scheduled courses are shown in pale olive. Open dates indicated as a guide only and are reserved for private, on-site courses. If you have three or more students and would like a course custom-tailored to your needs, please phone us to make sure the dates are still available. The office is closed for Christmas and Easter holidays.

APRIL 2005

BANK HOLIDAY			Apache	
28	29	30	31	1
PHP	PHP			
4	5	6	7	8
MySQL				
11	12	13	14	15
18	19	20	21	22
25	26	27	28	29

JULY 2005

27	28	29	30	1
4	5	6	7	8
11	12	13	14	15
PERL			PERL	
18	19	20	21	22
25	26	27	28	29

MAY 2005

BANK HOLIDAY	RegEx	Python		
2	3	4	5	6
TCL			TCL	
9	10	11	12	13
16	17	18	19	20
23	24	25	26	27

AUGUST 2005

1	2	3	4	5
8	9	10	11	12
15	16	17	18	19
PHP	PHP			
22	23	24	25	26

JUNE 2005

BANK HOLIDAY	LAMP			
30	31	1	2	3
PHP				PHP
6	7	8	9	10
MySQL		Apache		
13	14	15	16	17
PERL				
20	21	22	23	24
27	28	29	30	1

SEPTEMBER 2005

BANK HOLIDAY		RegEx	MySQL	
29	30	31	1	2
5	6	7	8	9
12	13	14	15	16
LAMP				
19	20	21	22	23
PERL				
26	27	28	29	30



Distributing open source applications easier than you think

If you develop a web-based application using Apache, MySQL and PHP, but then want to distribute it so that anyone can run it on their Windows PC, it might be easier and cheaper than you think. Apache httpd, MySQL and PHP are all available as open source downloads for Windows as well as for Linux. You can configure your system, add a few start-up scripts, then burn the “AMP” of “LAMP” onto a CD. At run time, just a few directories need to be installed on the hard disk of the user’s machine. Once the services start, your user is up and running. If you’ve written the scripts well, all this will happen automatically.

Further details can be found online at: <http://www.wellho.net/solutions/general-windows-running-apache-and-mysql-from-a-cd.html>

WHC welcomes new employee, Leah

I’m delighted to welcome Leah Davies onto our team.

Investment in good resources and efficient systems has meant that we’ve been well able to run the business with just two staff. An excellent opportunity arose to grab someone we much admire to add to the team at a time when we were beginning to feel stretched was just too good to pass up, and Leah joined us around Christmas time.

Leah is a specialist through practical experience of web applications such as YaBB and Simple Machines forums – open source products which are (respectively) Perl- and PHP-based, so she’ll be doing a lot of our web site and contract support work. She’s also very familiar with small company operation and she’ll be looking after our books.

But we’re a small company and each of us

turns our hand to whatever needs to be done; whoever’s in the office – be it Lisa, Graham or Leah – we will be happy to advise you on anything from course suitability to the arrangements for on-site courses.

BOOK REVIEW

Apress’s distinctive looks filling shelves

It’s good to see some old favourite books, and some new, published by Apress whose black and yellow covers have started to decorate the shelves of our local specialist bookstores in recent months.

“Regular Expression Recipes” starts off with a short, 20-page syntax review covering



by Nathan A. Good
published by Apress
ISBN: 1-59059-441-X

regular expressions in Perl, PHP and Python, as well as vim, grep and sed before jumping into a large number of worked examples in each of the languages. The examples consistently give a regular expression to perform

a task such as validating a date or stripping white space out of cascaded style sheets, then explain how the regular expression works, element by element. For newcomers to regular expressions, the examples in this book are an excellent springboard to some really useful tasks, with sufficient explanation to allow the user to tune the examples to meet his specific needs, and to learn further and write his own examples.

With any book of this nature, some of the patterns will vary around the world. An international telephone number dialled for the USA is not of the same format as an international number dialled from the UK, and it’s open to interpretation whether 01/04/05 is the 1st of April or the 4th January. This book is heavily USA-biased. It’ll give you a pattern for a USA Zip code, without even a mention of a UK postcode. But then with a little thought and time and learning from the other examples, you should be able to figure out your own postcode match anyway.

Your Contacts at Well House Consultants:

Administration	Lisa Ellis	lisa.ellis@wellho.net
Technical	Graham Ellis	graham@wellho.net
Accounts/Contracts	Leah Davies	leah@wellho.net
Phone	+44 (0)1225 708 225	0845 8382 404 (Lo-call)
Fax	+44 (0) 1225 707 126	0845 8382 405 (Lo-call)
Web site	http://www.wellho.net	
Postal Address	404, The Spa, Melksham, Wiltshire, England SN12 6QL	

Our office hours are 08:00 to 18:00 Monday to Friday.

You’ll usually reach us at the weekend and in the evenings, too, as phone messages and faxes are emailed to us automatically. If you email us, we’ll receive your email within 24 hours, 365 days a year.

This newsletter is published by Well House Consultants Ltd. and is distributed free. Please contact us if you’re reading someone else’s copy and would like to be added to the circulation list, or if you would like to be removed from our list.

All material in this newsletter is copyright Well House Consultants Ltd., and may not be reproduced in any form or by any means electronic without prior written permission of Well House Consultants except in the case of brief quotations embodied in critical articles and reviews. Well House Consultants has attempted throughout this publication to distinguish proprietary trademarks from descriptive terms by following the capitalisation style used by the manufacturer.