



MAGAZIN FÜR PROFESSIONELLE
INFORMATIONSTECHNIK

Anschluß für die Firma:
**Internet-Provider
in Deutschland**

WWW: XML kommt

Grundlagen, Programmierung, Werkzeuge

World Wide Web:

Suchmaschinen

Site-Indizierer im Vergleich

Clustering:

Linux als Supercomputer

Betriebssysteme:

Solaris 7 kostenlos

RISC-Systeme:

400-MHz-SPARC-Server

High-End-3D von HP und Sun

Kommunikation:

EMail-Programme für Linux

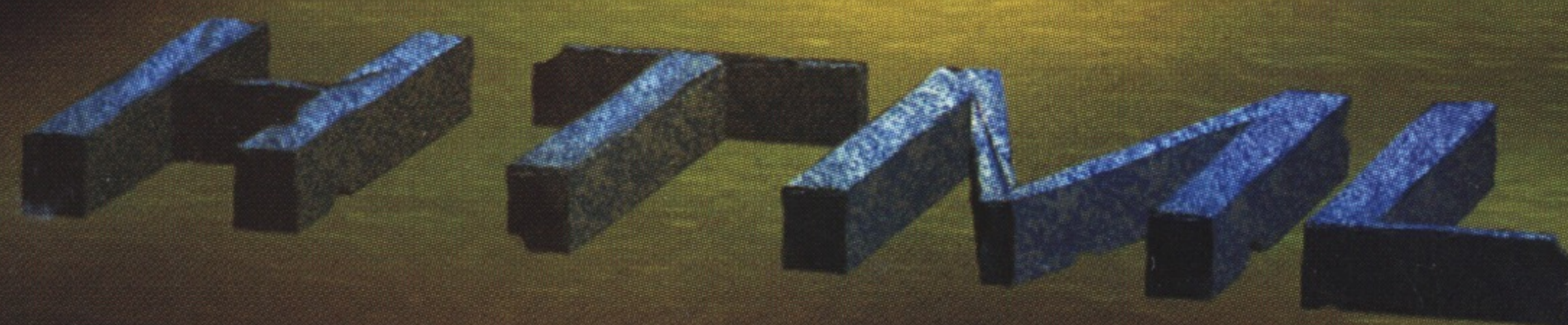
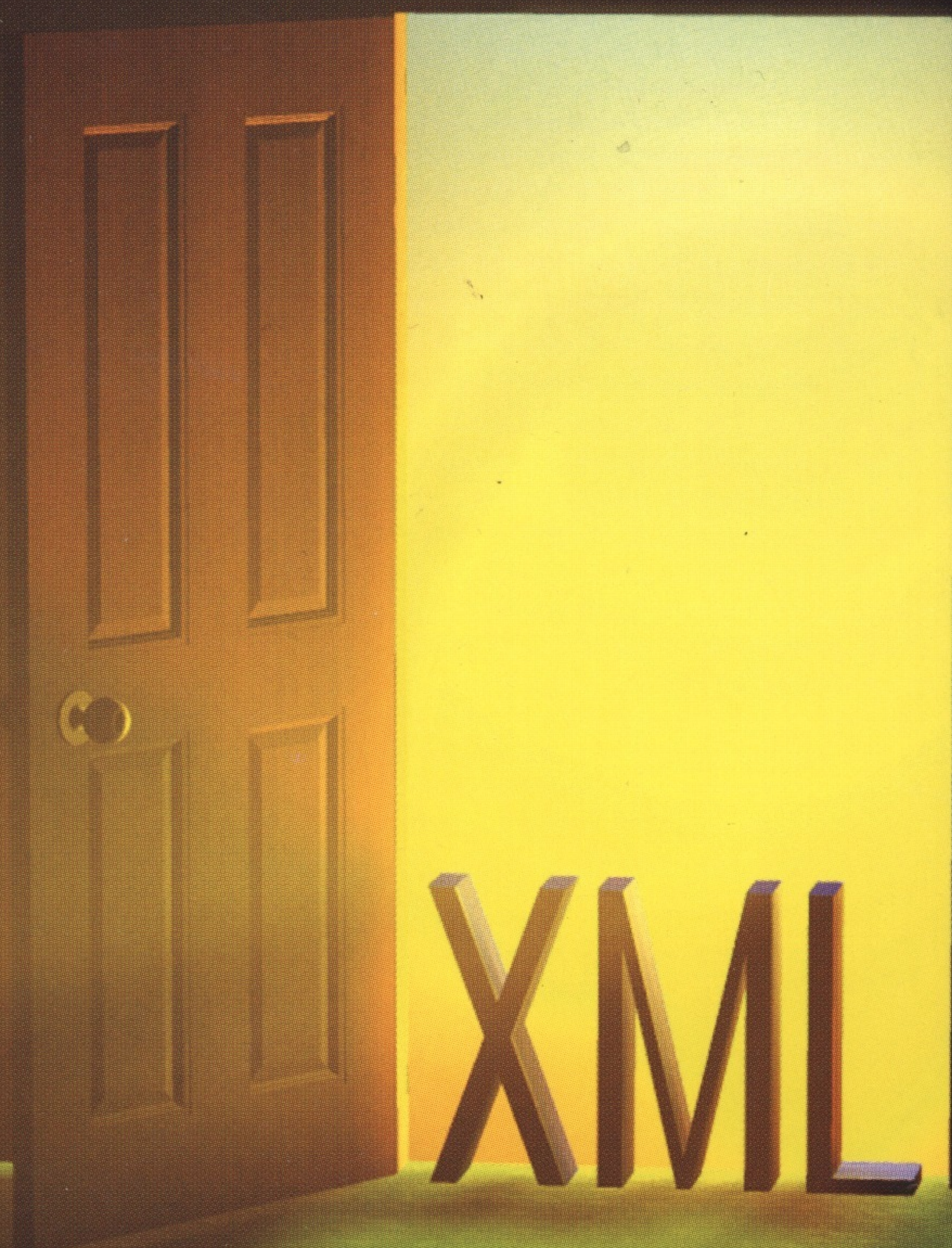
Gefälschte Massen-EMails

Softwareentwicklung:

OpenGL Volumizer

Generative Programmierung

Web-basierte NT-Verwaltung mit ADSI



Ob nun ein Werbebanner oder ein einfacher Link auf eine andere Web-Seite verweist: Jeder Webmaster möchte gerne wissen, wer welche Knöpfchen auf seiner Website drückte, um fremde Seiten zu erobern – und sei es nur, daß die Anzeigenabteilung mit den Erfolgszahlen ihrer Werbebanner angeben kann. Ein ordinärer Link, als HTML-Code

```
<A HREF="http://remote.host.com"> Klick mich! </A>
```

oder als Bestandteil der Sequenzdatei des in der letzten iX vorgestellten Bannerservers [1], zeigt auf <http://remote.host.com>. Klickt der Anwender darauf, springt der Browser dorthin, ohne der bisherigen Seite adieu zu sagen – und demnach weiß dort niemand, daß dieser Vorgang überhaupt stattgefunden hat.

Ein für den Web-Surfer unsichtbarer Zwischenschritt bringt die Lösung: Der Link zeigt nicht auf die anzuspringende Seite, sondern auf ein CGI-Skript der heimischen Website, das den Vorgang protokolliert und anschließend dem Browser mittels einer Redirect-Anweisung vorgaukelt, das Dokument habe seinen Standort gewechselt. Daraufhin verzweigt jener anstandslos zur eigentlichen Seite:

```
<A HREF="http://mysite.com/cgi/go.cgi/http://remote.host.com">
Klick mich! </A>
```

ruft das Skript *go.cgi* im *cgi*-Verzeichnis der darstellenden Website *mysite.com* auf und gibt ihm als *PATH_INFO*-Parameter die Zeichenkette '<http://remote.host.com>' mit. Er veranlaßt es, den Browser des Anwenders auf <http://remote.host.com> weiterzulotsen und anschließend in aller Ruhe den Vorgang in eine Log-Datei zu protokollieren.

Einheitliches Format für Log-Dateien

Diese Datei könnte man auf alle möglichen Arten füllen, doch da die gesammelten Daten nicht dort vermögen, sondern mit frei verfügbarer Software später ansprechend dargestellt werden sollen, benutzt das CGI-Skript das erweiterte 'Common Log File Format'. Daran halten sich die meisten Server beim Schreiben von Log-Dateien.

Klickt etwa ein Benutzer des Rechners *ppp.isp.com* auf der Web-Seite

Link-Klicks mit CGI zählen

Ausreisekontrolle

Michael Schilli

Die Benutzer einer Website registriert die Log-Datei des Servers, die populäre Analyseprogramme detailliert und farbenfroh darzustellen wissen. Doch wieviel Abenteuerlustige wandern über die eingearbeiteten Links zu externen Seiten? Diese Zahlen erfaßt nur der Click-Through-Tracker.



<http://referer.com/index.html> den Link <http://www.remote.com/index.html> an, schreibt *www.remote.com* folgende Zeile in seine Log-Datei:

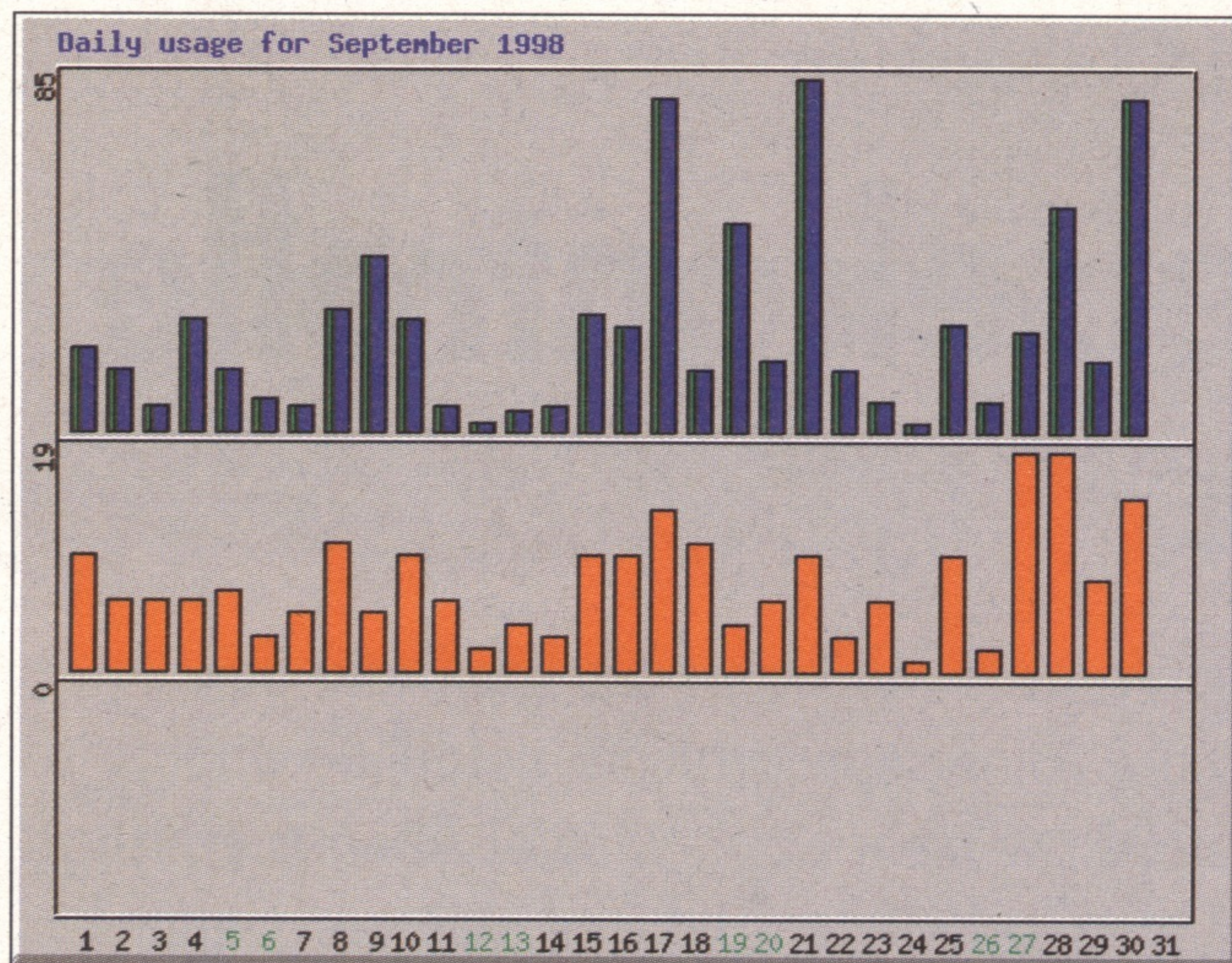
```
ppp.isp.com - - [14/Oct/1998:20:32:04 -0700] \
"GET /index.html HTTP/1.0" \
200 1234 http://www.referer.com/index.html \
"Mozilla/3.04Gold (X11; I; Linux 2.0.30 i586)"
```

Die insgesamt neun Felder des Eintrags haben folgende Bedeutung:

- Name oder IP-Adresse des anfragenden Rechners. Ist der Benutzer über PPP beim Service-Provider angemel-

det, steht hier gerne mal ein 'dynamischer' Name, der die dynamisch zugewiesene IP-Adresse widerspiegelt.

- Ein Bindestrich.
- Die UserID, falls es sich um eine paßwortgeschützte Seite handelt, falls nicht, ein weiterer Bindestrich
- In eckigen Klammern das aktuelle Datum des Zugriffs mit Zeitzoneangabe relativ zu GMT.
- In einem in Anführungszeichen eingeschlossenen String: Zugriffsmethode (meist GET oder POST), angeforderte Datei, Protokollversion.



Die obere Grafik zeigt für jeden Tag die Zahl der 'Durchklicker' (Surfer, die Links drücken), die untere die Zahl zugreifender Rechner (Abb. 1).

Top 30 of 81 Total URL's					
#	Hits	KBytes	URL		
1	62	9.31%	0	0.00%	http://www.perlmeister.com/german/perl/gotoperl/
2	41	5.50%	0	0.00%	http://www.linux-magazin.de/ausgabe.1998.08/Pounder/pounder.html
3	40	5.36%	0	0.00%	http://www.ix.de/ix/artikel/1998/07/133/artikel.html
4	33	4.42%	0	0.00%	http://www.linux-magazin.de/ausgabe.1998.10/Expect/expect.html
5	33	4.42%	0	0.00%	http://perlmeister.com/resume.html
6	27	3.62%	0	0.00%	http://www.linux-magazin.de/ausgabe.1998.09/Kalender/kalender.html
7	27	3.62%	0	0.00%	http://www.heise.de/ix/artikel/1997/08/150/artikel.html
8	27	3.62%	0	0.00%	http://www.ix.de/ix/artikel/1998/10/154/artikel.shtml
9	26	3.49%	0	0.00%	http://www.addison-wesley.de/katalog/query.pperl
10	23	3.09%	0	0.00%	http://www.heise.de/ix/artikel/1997/05/166/artikel.html
11	23	3.09%	0	0.00%	http://www.linux-magazin.de/ausgabe.1998.04/CGI/cgi2.html
12	21	2.82%	0	0.00%	http://www.buchkatalog.de/kod-bin/suche.exe
13	20	2.69%	0	0.00%	http://www.heise.de/ix/artikel/1998/01/157/artikel.html
14	17	2.29%	0	0.00%	http://www.heise.de/ix/artikel/1998/02/140/artikel.html
15	17	2.29%	0	0.00%	http://www.linux-magazin.de/ausgabe.1998.03/CGI/cgi1.html
16	14	1.89%	0	0.00%	http://www.heise.de/ix/artikel/1997/07/154/artikel.html
17	13	1.74%	0	0.00%	http://www.linux-magazin.de/ausgabe.1998.06/CGI/cgi4.html

Webalizer liefert auch eine Liste der von der Website aus angeklickten Links, sortiert nach Häufigkeit (Abb. 2)

- Statuscode (200 für OK, 404 für Document not found, et cetera).
- Anzahl der übertragenen Bytes (1234).
- URL der referenzierenden Seite, falls vorhanden (erweitertes Common Log File Format).
- Art und Version des Browsers, Betriebssystem mit Version (erweitertes Common Log File Format).

Gibt der im Listing vorgestellte Click-Through-Tracker seine Log-Daten auch in diesem Format aus, interpretieren bekannte Analyseprogramme wie *webalizer* oder *analog* die Daten, als kämen sie aus der Datei eines Web-Servers. Die Abbildungen 1 bis 3 zeigen eine Auswahl dessen, was der *webalizer* (Installation siehe Kasten 'Webalizer-Installation') aus dem Datensatz zaubert: Säulengrafiken, die die stündliche und tägliche Verteilung eingehender Benutzerklicks darstellen, die Top-30-Liste angeforderter URLs, eine Liste der beliebtesten Browser und vieles mehr.

Listing *go.cgi* zeigt die Implementierung des protokollierenden CGI-Skripts. Zeile 10 legt die Datei fest, in die *go.cgi* seine Protokolldaten schreibt. Sie muß für den Benutzer, unter dessen Namen der Web-Server läuft (meist nobody) beschreibbar sein.

PATH_INFO als Umleitungsschild

Ruft man ein CGI-Skript *http://yada.yada.com/cgi-bin/script* als *http://yada.yada.com/cgi-bin/script/some/path*

auf, liegt ihm nach dem Aufruf der Pfad */some/path* in der Umgebungsvariablen *PATH_INFO* vor – so will es das CGI-Protokoll. Zeile 12 in *go.cgi* holt den Parameter, Zeile 13 schneidet das führende / ab und Zeile 15 beendet das Skript, falls kein Pfad vorhanden ist.

Läuft alles wie vorgesehen, leitet die Ausgabe von Zeile 17 den anfragenden Browser an die neue Adresse weiter. Solche Redirect-An-

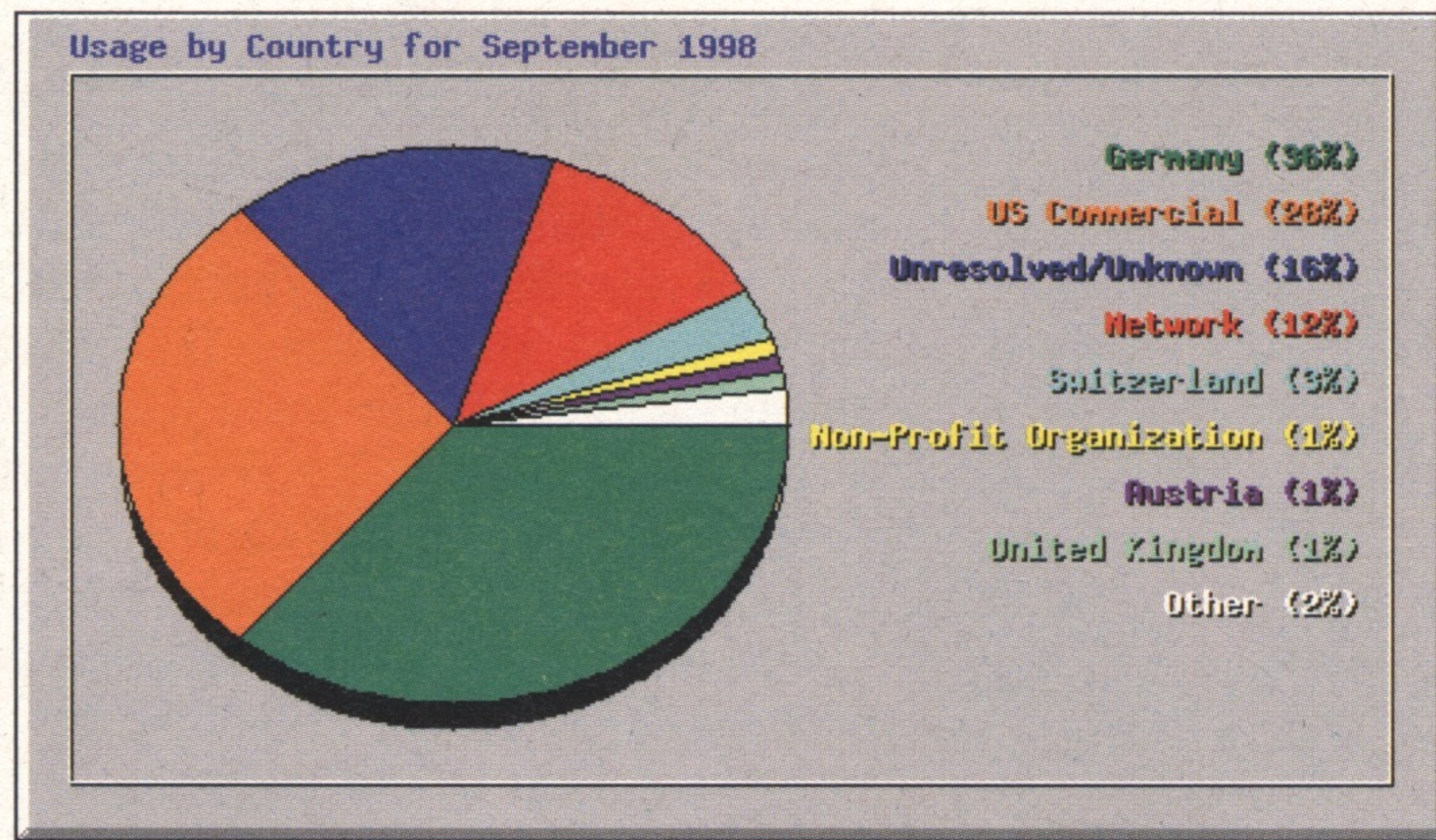
weisungen verarbeitet letzterer, ohne daß der Anwender davon Wind bekommt (nur bei genauem Hinsehen bemerkt man, daß sich der angezeigte URL kurzfristig verändert). Da das Schreiben in die Log-Datei einige Zeit in Anspruch nehmen kann, die Kommunikation mit dem Browser zu diesem Zeitpunkt aber bereits erledigt ist, erzeugt *go.cgi* nun mit *fork()* einen Child-Prozeß, der das Log-Schreiben übernimmt, und beendet

LISTING

```

1 #!/usr/bin/perl -w
2 #####
3 # Click-Through Tracker
4 #
5 # 1998, schilli@perlmeister.com
6 #####
7
8 use Socket;          # For address reversal
9
10 my $LOG_FILE = "/services/http/DATA/go.log";
11
12 my $path = $ENV{PATH_INFO};
13 $path =~ s#/#g;
14
15 die "Redirect Path not set" unless defined $path;
16
17 print "Location: $path\r\n\r\n"; # Issue redirect
18
19 my $newpid = fork();   # New Child process
20
21 # Parent exits on successful creation of child
22 exit(0) if (defined $newpid && $newpid != 0);
23 # child running from here
24 close(STDOUT);         # We're done sending
25 log_request($path);    # Log redirect
26
27 #####
28 sub log_request {
29 #####
30     my $url = shift;
31
32     open(LOG, ">>$LOG_FILE") ||
33         die("Cannot open $LOG_FILE: $!");
34
35     flock(LOG,2);       # Lock exclusive
36     seek(LOG, 0, 2);    # Seek to end
37
38     my @months = qw(Jan Feb Mar Apr May Jun Jul
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

Besucher einer Website nach Ursprungs-Domains (Abb. 3).

sich anschließend. `close(STDOUT)` sorgt dafür, daß der steuernde Web-Server sich nicht mehr um das Skript kümmert.

Die Funktion `log_request` mit dem zu notierenden URL als Parameter hängt einen Eintrag im 'Extended Common Log File Format' an die Log-Datei an. Um gegen parallel laufende Web-Server-Prozesse gewappnet zu sein, sichert sie die Datei vor dem Schreiben mit `flock()`. Diese Funktion synchronisiert den Zugriff auf eine Datei, indem sie jeweils nur einem einzigen Prozeß oder Thread den Zugriff gewährt und alle anderen, die ebenfalls einen `flock()` darauf versuchen, so lange blockiert, bis der Eigentümer die Sperre wieder löst. Aus der aktuellen Uhrzeit entsteht ein String vom Format

14/Oct/1998:20:32:04 +0000

(ist die gewählte Zeitzone nicht GMT, ist der String in Zeile 45 entsprechend anzupassen), und die Umgebungsvariablen `$HTTP_REFERER` und `$HTTP`

`_USER_AGENT` liefern gemäß dem CGI-Protokoll die referenzierende Seite und den Browsertyp des Anwenders. Da `$REMOTE_ADDR` nicht den Namen des aufrufenden Rechners beinhaltet, sondern dessen IP-Adresse, versucht die Funktion `reverse_lookup` mit ihr als Parameter den Rechnernamen zu bestimmen und gibt diesen im Erfolgsfall zurück, andernfalls bleibt es bei der IP-Adresse. (ck)

MICHAEL SCHILLI

arbeitet als Web-Engineer für America Online Inc., San Mateo. Er ist Autor des 1998 bei Addison-Wesley erschienenen Buches 'GoTo Perl 5'.

Literatur

- [1] Michael Schilli; Rotationsprinzip, Anzeigen schalten im Web iX 1/99, S. 147 ff.
- [2] Randal Schwartz; Click-Through Tracking in Perl; WebTechniques 5/98, S. 28 ff.

Webalizer-Installation

Der Webalizer, ein Analyseprogramm für Log-Dateien von Web-Servern, steht unter <http://www.mrunix.net/webalizer/download> in der Version 1.20 zur kostenlosen Abholung bereit. Für die gängigen Unix-Varianten gibt es Binärversionen, die man nur in einem neuen Verzeichnis entpacken muß. Neben dem Programm `webalizer` steht dort die Konfigurationsdatei `sample.conf`, die nach `webalizer.conf` kopiert werden muß. Folgende Einträge sind anzupassen:

LogFile-Pfad/zur/Log-Datei
OutputDir Pfad/zum/Ausgabe/Verzeichnis

`webalizer` liest die zu analysierende Log-Datei ein, verarbeitet ihre Datensätze und legt das Ergebnis monatsweise in mehreren Dateien (*.html und *.gif) im definierten Ausgabeverzeichnis ab. `index.html` in diesem Verzeichnis bietet einen Ein-

stiegspunkt, um durch verschiedene Monatsdaten zu stöbern.

Soll der Webalizer statt der voreingestellten monatlichen Zusammenfassungen die Daten täglich aktualisieren, ist folgendes zu beachten: Damit er nicht immer wieder die gesamten Daten seit Monatsanfang durchorgeln muß, bietet er inkrementelle Verarbeitung. Dabei 'merkt' sich das Programm die bereits ausgewerteten Log-Daten in einer sogenannten Cache-Datei, so daß es nur noch mit neuen Daten gefüttert werden muß. Um dies zu aktivieren, ist

Incremental yes

in `webalizer.conf` einzustellen, und schon kommt er mit täglich eingespeisten rotierten Log-Dateien zurecht.

<http://www.heise.de/tp>



Wir hängen unser Fähnchen nicht in den Wind

Telepolis bietet ein breites Spektrum an Berichten über die wachsende Netzkultur. Aspekte von Politik, Wissenschaft, Wirtschaft, Technologie, Kunst und Kultur werden kritisch und vielseitig aus der Netzperspektive behandelt. Neben aktuellen Artikeln, Reportagen und Kommentaren finden Sie profunde Auseinandersetzungen zu Schwerpunktthemen wie Netz-Ökonomie, Netzkunst, digitales Kino, Künstliches Leben, Internet & Politik oder Architektur.

TELEPOLIS

MAGAZIN DER NETZKULTUR