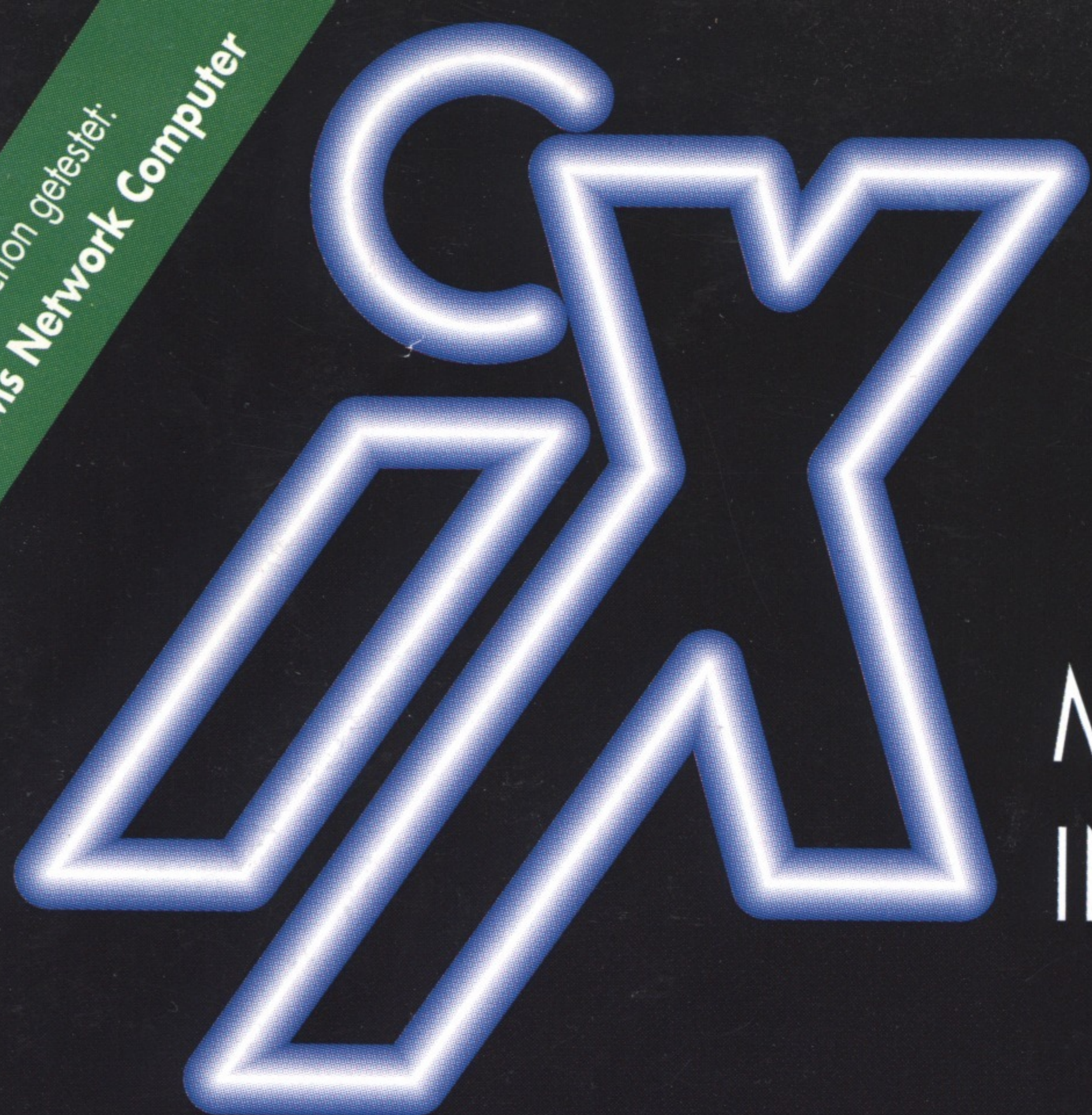


Schon getestet:
IBMs Network Computer

öS 62,- sfr 7,50 **DM 7,50** H 10554

Juli 1997

7



MAGAZIN FÜR PROFESSIONELLE
INFORMATIONSTECHNIK

Verglichen:

Java-Office-Pakete

Applix Anyware und Corel Office for Java

World Wide Web:

ASP-Programmierung für MS-Server
HTTP-Alternative IIOP

PC-Unix:

BSDI Internet Server 3.0
Solaris 2.5.1 x86

DFÜ:

56-kBit/s-Modem

EMail:

IMAP4 statt POP3

LANs:

Tuning für schnelle Netze

Standardsoftware:

KHK Office Line

130 000-DM-Workstation im Test:

SGI Octane MXI

Marktübersicht:

**16 PC-Management-
systeme**



Apaches CGI-Ausführung beschleunigen

Donnerbüchse

Michael Schilli

Eine frei erhältliche Bibliothek bringt dem populären Apache-Server Perl bei: So lassen sich gegenüber herkömmlichen CGI-Scripts Geschwindigkeitsverbesserungen um 1000 % erzielen.



Als CGI-Sprache eingesetzt, hat Perl einen gravierenden Nachteil: Für jeden Request (und das können bei einem populären Web-Server Dutzende pro Sekunde sein) muß der Interpreter das entsprechende Script analysieren, auf Syntax-Fehler überprüfen, in den internen Bytecode übersetzen und schließlich ausführen. Bindet es noch zusätzliche Perl-Module ein, läuft aber relativ kurz, kann das Verhältnis von Lade- zu Ausführungszeit schon mal mehr als 10 : 1 betragen. Für wirklich zeitkritische Anwendungen blieb bislang nur eine Umsetzung des Scripts in C, da übersetzte Programme fast verzögerungsfrei anlaufen.

Mit dem Web-Server 'Apache' und der *mod_perl*-Distribution von Doug MacEachern ist dies jedoch nicht mehr notwendig: Das *mod_perl*-'Plug-in' versorgt den Web-Server mit der Perl-Runtime-Library. So ausgestattet, behält er einmal ausgeführte Scripts startfähig im Speicher und durchläuft den Lese-Check-Compile-Prozeß nur, wenn sich der Inhalt des Scripts geändert hat. Das reduziert die mittlere Ausführungszeit für einen HTTP-Request dramatisch, Performance-Verbesserungen um das Zehn- bis Zwanzigfache sind keine Seltenheit. Damit der Server diese Aufgabe übernimmt, ist die Apache zusammen mit der *mod_perl*-

Distribution zu übersetzen und zu installieren (siehe Kasten 'Installation').

Apache::Registry heißt das Modul, das Perl-CGI-Scripts so performant ausführt. Es wird aktiv, falls ein HTTP-Request auf Scripts in bestimmten Verzeichnissen vorliegt, packt das gesamte Script in eine Perl-Subroutine und führt diese aus. Hieraus ergeben sich einige Unterschiede zum normalen CGI-Betrieb. So sollte jedes Script mit der Sequenz

```
use CGI::Switch;
$query = CGI::Switch->new();
print $query->header();
```

beginnen, bevor es irgendwelche Ausgaben nach *stdout* schickt.

Adressen zehnmal schneller registrieren

Scripts, die Lincoln Steins *CGI.pm* benutzen, laufen fast unverändert mit dem Apache-Plug-in, wenn man statt des Package *CGI* den dynamischen Schalter *CGI::Switch* verwendet, der entsprechend den Anforderungen zwischen regulärem CGI und *Apache::CGI* unterscheidet. Das Listing *reg.pl* zeigt als Anwendung ein Registrierungsskript, das in ein Formular eingetragene EMail-Adressen zeilenweise in einer Datei speichert.

Das in Zeile 7 erzeugte *CGI::Switch*-Objekt bietet eine komfortable Schnittstelle zu eingehenden Formular-Parametern und vereinfacht später die Ausgabe einiger HTML-Tags. Liegt als Eingabeparameter keine EMail-Adresse vor, verzweigt Zeile 9 zur Subroutine *print_form*, die ein Formular zur Eingabe einer EMail-Adresse ausgibt.

Adressen, die der reguläre Ausdruck in Zeile 13 plausibel findet, hängt *reg.pl* an die Datei *email.dat* an, falls sie sie noch nicht enthält (Zeilen 18 bis 31). Wegen der in der CGI-Welt üblichen parallelen Ausführung von Requests muß sich das Script dafür zunächst mit *flock()* den exklusiven Zugriff auf *email.dat* sichern. Paßt die eingetragene EMail-Adresse nicht in das einfache Raster '@_._', gibt *reg.pl* eine Fehlermeldung aus.

Die Subroutine *print_form()* verarbeitet als zweiten Parameter eine Meldung, die sie zusammen mit dem Registrierungsformular in HTML ausgibt. Das Eingabefeld behält eventuell bereits gesetzte Werte bei. Das aus dem Hauptprogramm hereingereichte *CGI::Switch*-Objekt *\$q* bietet die Methoden aus *CGI.pm* und erlaubt eine

Installation

Zur Installation entpackt man `mod_perl-0.97.tar.gz` (aus `CPAN/modules/by-module/Apache`) und `apache_1.2b8.tar.gz` (beispielsweise von <http://ftp.cs.tu-berlin.de/pub/net/www/apache/dist/>) im gleichen Verzeichnis:

```
gtar xzf apache_1.2b8.tar.gz
gtar xzf mod_perl-0.97.tar.gz
```

Die Vorbereitung von `mod_perl` mit

```
cd mod_perl-0.97
perl Makefile.PL
```

wirft dann die Frage

```
Configure mod_perl with ../apache_1.2b8/src ? [y]
```

auf, die mit `y` zu beantworten ist, ebenso wie

```
Shall I build httpd in ../apache_1.2b8/src for you? [y]
```

Ein anschließendes `make` bereitet nicht nur die `mod_perl`-Installation vor, sondern übersetzt gleichfalls die Apache-Distribution und konfiguriert den Server praktischerweise gleich für den Einsatz des Perl-Moduls. Die Installation erledigt ein anschließendes `make install`.

Der Apache-Server selbst benötigt weitere Fürsorge:

```
ln -s /usr/local/etc/apache1_2b8\
/usr/local/etc/httpd
```

richtet einen symbolischen Link ein, der den Pfad zum Server verkürzt und für die folgenden Schritte Tipparbeit spart.

Im `conf`-Verzeichnis liegen die Dateien `access.conf-dist`, `httpd.conf-dist` und `srm`.

`conf-dist`, die an die lokalen Gegebenheiten angepaßt und hierfür in `access.conf`, `httpd.conf` und `srm.conf` umbenannt werden müssen. Falls das Installationsverzeichnis des Apache nicht `/usr/local/etc/httpd` ist, sind folgende Zeilen zu ändern: In `srm.conf`:

```
DocumentRoot /usr/local/etc/httpd/htdocs
```

In `access.conf`:

```
<Directory /usr/local/etc/httpd/htdocs>
...
<Directory /usr/local/etc/httpd/cgi-bin>
AllowOverride None
Options None
</Directory>
```

In `httpd.conf`:

```
ServerRoot /usr/local/etc/httpd
```

Speziell für die `mod_perl`-Konfiguration gilt folgendes: Damit ein unter `/usr/local/etc/httpd` installierter Apache-Server unter `/usr/local/etc/httpd/perl` liegende Perl-Programme als CGI-Scripts dem `mod_perl`-Handler zur Ausführung übergibt, muß folgender Eintrag in die Aliases-Abteilung von `srm.conf`:

```
Alias /perl/ /usr/local/etc/httpd/perl/
<Location /perl>
SetHandler perl-script
PerlHandler Apache::Registry
Options ExecCGI
</Location>
```

So ausgerüstet, fehlt nur noch, den Server zu starten, was

```
/usr/local/etc/httpd/src/httpd -f
/usr/local/etc/httpd/conf/httpd.conf
```

erledigt. Da Apache sofort nach dem Start eine Reihe von Child-Prozessen erzeugt, und es müßig ist, diese in der Testphase stets der Reihe nach zu beenden, legt der Server in `logs/httpd.pid` seine eigene Prozeß-ID ab. Zum Beenden des Servers samt aller seiner Kinder genügt es, einen `kill`-Befehl mit dieser PID abzusetzen:

```
kill `cat /usr/local/etc/httpd/logs/httpd.pid`
```

stoppt also den laufenden HTTP-Daemon samt seiner Abkömmlinge.

Liegt das hier vorgestellte Script `reg.pl` in `/usr/local/etc/httpd/perl`, läßt es sich über die HTTP-Schnittstelle mit einem Browser über

```
//servername/perl/reg.pl
```

aufrufen.

Zur Installation zusätzlich notwendige Module: `CPAN/modules/by-module/WWW/libwww-perl-5.08.tar.gz` `CPAN/modules/by-module/CGI/CGI.pm-2.34.tar.gz`, in Deutschland unter anderem bei

<ftp://ftp.leo.org/pub/comp/programming/languages/perl/CPAN/>

<ftp://ftp.rz.ruhr-uni-bochum.de/pub/CPAN/>

<ftp://ftp.uni-hamburg.de/pub/soft/lang/perl/CPAN/>

ftp://ftp.uni-erlangen.de/pub/source/Perl/CPAN

<ftp://ftp.gmd.de/packages/CPAN>

```
1 #!/usr/bin/perl -w
2
3 use strict;                # Strenge Variablenkonventionen
4 use CGI::Switch;           # 'Umschaltendes' CGI
5
6 my $file = 'email.dat';    # Datei mit Adressen
7 my $q = CGI::Switch->new(); # CGI-Objekt
8
9 if(! defined $q->param('email')) { # Keine Email eingetragen (erster
10     # Aufruf?) => Einführungsseite
11     print_form($q, 'Bitte tragen Sie Ihre Email-Adresse ein.');
```

```
36
37     print_form($q, 'Registrierung erfolgreich. Vielen Dank.');
```

reg.pl speichert per Formular eingegebene EMail-Adressen zeilenweise in einer Datei.

optisch ansprechende Formulierung des HTML-Textes. Gegenüber normalem CGI-Betrieb ergibt sich bei *reg.pl* durch die im Server eingebaute Perl-Bibliothek eine Performance-Verbesserung um den Faktor 10.

Vorsicht bei globalen Variablen

Allerdings birgt der Ansatz, lauffähige Scripts im Speicher zu halten, auch Tücken: So behalten etwa globale Variablen ihren Wert über die Laufzeit des Scripts hinaus bei. Zusätzlich verzweigt der Apache HTTP-Requests alternierend an seine Child-Prozesse – und die Werte globaler Variablen bleiben in deren Adreßraum konstant. Das Script

```
use CGI::Switch;
my $cgi = CGI::Switch->new();
print $cgi->header();
print "globalvar=", $globalvar++, "(PID=$$)\n";
```

liefert pro Aufruf den Wert einer globalen Variablen *\$globalvar* samt der ID des ausführenden Prozesses zurück (\$\$). Mehrfach vom Apache-Plug-in ausgeführt, erzeugt das Script nacheinander folgende Ausgaben:

```
globalvar=0 (PID=29045)
globalvar=0 (PID=29046)
globalvar=0 (PID=29049)
globalvar=0 (PID=29054)
globalvar=1 (PID=29045)
globalvar=1 (PID=29046)
globalvar=1 (PID=29049)
globalvar=2 (PID=29045)
globalvar=2 (PID=29046)
```

Lösung des Rätsels: Erwischt ein HTTP-Request zufällig wieder denselben Child-Prozeß, erhöht sich der Wert der globalen Variablen *\$globalvar* gegenüber dem letzten Aufruf um 1. Man sollte also auf globale Variablen ganz verzichten.

mod_perl bietet übrigens nicht nur eine Performance-Steigerung bei der Ausführung von CGI-Scripts, sondern eine universelle Schnittstelle zum Apache: Mit simplen Perl-Scripts kann man so Server Side Includes steuern, die User-Authentisierung erweitern oder einen intelligenten Proxy-Server bauen. (ck)

MICHAEL SCHILLI

ist Softwareentwickler bei
Black Sun Interactive, San Francisco.

Apache für den Mac

Während es für Windows NT noch keinen Apache-Server gibt, ist für den Mac seit kurzem eine kommerzielle Version erhältlich. Die amerikanische Firma Tenon (<http://www.tenon.com>) bietet ihre Implementierung des beliebten Daemon unter dem Namen 'WebTen' für 495 US-\$ an, eine Demo-Version ist kostenlos über das Netz erhältlich.

Anders als das frei erhältliche Produkt der Apache-Gruppe enthält Tenons Server bereits einen Perl-Interpreter, so daß er entsprechende CGI-Scripts ausführen kann, ohne ein externes Programm zu bemühen. Erweiterungsmodule für WebTen können interessierte Entwicklerinnen selbst erstellen – sofern sie die ebenfalls von Tenon erhältliche Entwicklungsumgebung 'CodeBuilder' benutzen.



- FARSIDE Informationssysteme GmbH beschäftigt sich mit der Entwicklung von Software im SAP- und Microsoft Windows-Umfeld. Die Zielsetzung hierbei ist es, unseren Kunden intelligente Lösungen für die Integration von Desktop- und (Back-) Officeprodukten in die SAP Landschaft anzubieten. Besonderes Augenmerk legen wir dabei auf die Einbindung von auf Internettechnologie beruhender Informationsinfrastruktur in die betriebswirtschaftlichen Anwendungssysteme.

Zur Weiterentwicklung unseres SAP/Internet Integrationsproduktes **WEBsConnector** suchen wir eine/n

Softwareentwickler/in C++

Softwareentwickler

Softwareentwicklerin

- Sie haben bereits praktische Erfahrungen in C++ Projekten unter Windows/Windows NT sammeln können und COM, OLE und Active X sind für Sie mehr als Schlagworte. Ob Sie Ihre Erfahrungen an der Uni, im Job oder in eigenen Projekten gewonnen haben, ist für uns unerheblich. Hauptsache, Sie sind mit Begeisterung bei der Sache, engagieren sich gerne im Team und können auch gegenüber unseren Kunden als kompetenter Gesprächspartner auftreten.

- Wir bieten Ihnen ein interessantes und abwechslungsreiches Aufgabengebiet, ein leistungsbezogenes Gehalt und teamorientiertes Arbeiten bei flexiblen Arbeitszeiten.

Sollten wir Ihr Interesse geweckt haben, richten Sie bitte Ihre schriftliche Bewerbung mit Angabe Ihrer Gehaltsvorstellung, dem frühestmöglichen Eintrittstermin und Lichtbild an:

FARSIDE Informationssysteme GmbH, z.H. Frau Hornschuh, Bergheimer Straße 10-12, 69115 Heidelberg. Für Rückfragen stehen wir Ihnen jederzeit unter der Tel.: 06221/97 12 40 oder per Email: info@farside.de zur Verfügung.

FARSIDE
INFORMATIONSSYSTEME

eMail: info@farside.de
<http://www.farside.de>