

# ***Perl, das Quickhack-Tool mit Fun-Faktor***

Sprecher: Rene Schickbauer



# *Was ist Perl*

Perl ist eine freie, plattformunabhängige und interpretierte Programmiersprache („Skriptsprache“), die mehrere Programmierparadigmen unterstützt.

Perl setzt auf das Motto “There is always more than one way to do it”

Perl funktioniert auf (fast) allen Unix- und Windows-kompatiblen Plattformen

---

---

# ***Geschichte von Perl***

Der Linguist Larry Wall entwarf Perl 1987 als Synthese aus C, awk, den UNIX-Befehlen und anderen Einflüssen. Ursprünglich als Werkzeug zur Verarbeitung und Manipulation von Textdateien insbesondere bei System- und Netzwerkadministration vorgesehen (z. B. Auswertung von Logdateien), hat Perl auch bei der Entwicklung von Webanwendungen und in der Bioinformatik weite Verbreitung gefunden.

---

---

# *Perl installieren: Unix & Linux*

- wget  
`http://www.cpan.org/src/stable.tar.bz2`
  - `tar xvjf stable.tar.bz2`
  - `cd perl-5.8.8`
  - `./Configure` (und viele Fragen beantworten)
  - `make`
  - `sudo make install`
  - Perlscripts hacken bis der Arzt kommt
- 
-

# *Perl installieren: Windows/Cygwin*

- Ausführen:  
<http://www.cygwin.com/setup.exe>
  - Perl zusätzlich auswählen
  - Ausgewählte Pakete installieren
  - Über Installer ärgern
  - Performance verfluchen
  - Alles wieder de-installieren (per Hand, weil Installer zu dämlich)
  - ActiveState Perl Dev Pro Kit kaufen
  - Perlscripts hacken bis der Arzt kommt
- 
-

# Beispiel: BMP-Files nummerieren

```
use strict;
use warnings;

sub doPad($) {
    my $cnt = shift;
    while(length($cnt) < 5) {
        $cnt = "0$cnt";
    }
    return $cnt;
}

my $count = 1;
opendir(my $dfh, ".") or die($!);
while((my $fname = readdir($dfh))) {
    next if($fname !~ /\.bmp$/);
    my $newname = "new_" . doPad($count) . ".bmp";
    rename $fname, $newname;
    $count++;
}
closedir($dfh);
```

---

---

# *Was ist CPAN?*

CPAN (engl. Comprehensive Perl Archive Network) ist ein weltweit gespiegeltes Online-Repository für Perl-Module, -Anwendungen und Dokumentationen, und bietet die Möglichkeit einer komfortablen Installation und Verwaltung von Perl-Modulen. Heute umfasst das CPAN circa 270 Server auf allen fünf Kontinenten und enthält über 11.400 Module von über 5.700 Autoren, was einem Volumen von annähernd 3,5 GB entspricht.

---

---

# *CPAN Module*

- CPAN ist Modulbasiert
  - Jedes Modul kennt seine abhängigkeiten
  - Das CPAN-Clientmodul ist in jeder Perl-Installation dabei und stellt (u.a.) die CPAN-Shell bereit
  - Die in den Modulen bereitgestellten Dokumentationen werden auch als Webseiten zur Verfügung gestellt
  - Fast alle Module enthalten Selbsttests, die vor der Installation ausgeführt werden
- 
-



# *Wie funktioniert CPAN*

- Modul-Autor stellt das Modul via PAUSE-Server zur Verfügung
  - PAUSE-Server generiert HTML-Dokumentation
  - über 270 Mirrors spiegeln das Modul
  - Test-Server laden und bauen das Modul und reporten Fehler der Selbsttests
  - Benutzer ruft die CPAN-Shell auf und installiert das Modul mit allen Abhängigkeiten
- 
-

# *CPAN-Shell Beispiel*

Aufruf mit:

```
# perl -MCPAN -e shell
```

Beim ersten Aufruf wird der Benutzer aufgefordert, den/die naheliegendsten Mirror-Server auszuwählen

Installation vom Modul HTTP::Daemon  
cpan> i HTTP::Daemon

---

---

# *Was ist ActiveState Perl*

- Kommerzielle, optimierte Perl-Version für Windows (jetzt auch MacOSX)
  - Dev Pro Kit enthält graphischen SDK mit allem, was man von Visual Studio gewohnt ist (ausser, daß der PDK nicht so .NET-mässig langsam ist)
  - PDK enthält ein Jahresabo fürs O`Reilly Safari Bookshelf
  - mit dem PDK kann man Perl-Scripts kompilieren, am Zielrechner keine Perl-Installation notwendig
  - PDK unterstützt u.a. auch Windows Services
- 
-

# *Spass mit Perl: Email Signatur*

```
#!/usr/bin/perl -- #99BoB (C)2004 cavac:prg count drink vessel place act1 act2
@a=@ARGV;$c=$a[0]||99;$b=" " . ($a[2]||"bottles") . " of " . ($a[1]||"beer");$w=" " .
($a[3]||"on the wall") . ", ";do{print"$c$b$w\n$c$b, \n" . ($a[4]||"take one down") .
", " . ($a[5]||"pass it around") . ", \n" . --$c . "$b$w\n\n";}while($c);print"END!\n";
```



# *Spass mit Perl: passwd-Webserver*

```
use HTTP::Daemon;
use HTTP::Status;

my $d = HTTP::Daemon->new || die;
print "Please contact me at: <URL:", $d->url, ">\n";
while (my $c = $d->accept) {
    while (my $r = $c->get_request) {
        if ($r->method eq 'GET' and $r->url->path eq "/xyzzzy") {
            # remember, this is *not* recommended practice :-)
            $c->send_file_response("/etc/passwd");
        }
        else {
            $c->send_error(RC_FORBIDDEN)
        }
    }
    $c->close;
    undef($c);
}
```

---

---

# *Perl@Work: Email senden*

```
use Mail::Sendmail;

%mail = ( To      => 'you@there.com',
          From    => 'me@here.com',
          Message => "This is a very short message"
        );

sendmail(%mail) or die $Mail::Sendmail::error;

print "OK. Log says:\n", $Mail::Sendmail::log;
```

---

---

# *Perl@Work: Server überwachen*

```
use Net::Ping;  
  
$p = Net::Ping->new();  
print "$host is alive.\n" if $p->ping($host);  
$p->close();
```

# *Perl@Work: Server überwachen*

```
use IO::Socket;

$socket = IO::Socket::INET->new(PeerAddr => $host,
                                PeerPort => $port,
                                Proto     => "tcp",
                                Type      => SOCK_STREAM)
    or die "Connect failes to $host:$port : $@\n";

# ... do something with the socket
print $socket "Testpacket\n";

$answer = <$socket>;

# and terminate the connection when we're done
close($socket);
```

---

---



# Perl@Work:GUI mit TK

```
#!/usr/local/bin/perl
use Tk;

# Main Window
my $mw = new MainWindow;

my $label = $mw -> Label(-text=>"Rebooter V1") -> pack();
my $but = $mw -> Button(-text => "Reboot now!",
    -command =>\&push_button);
$but -> pack();

MainLoop;

#This is executed when the button is pressed
sub push_button {
    `shutdown -r now`;
}
```

# *Spass mit Perl: Acme::Brainfuck*

```
#!/usr/bin/env perl
use Acme::Brainfuck qw/verbose/;

while(1)
{
    print "Say something to Backwards Man and then press
enter: ";
    +[->,-----]<
    print 'Backwards Man says, "';
    [+++++++.<]<
    print "\" to you too.\n";
}
```

---

---

# Perl@Work: Inline::C

```
package Math::Simple;
$VERSION = '1.23';

use base 'Exporter';
@EXPORT_OK = qw(add);
use strict;

use Inline C => 'DATA',
            VERSION => '1.23',
            NAME => 'Math::Simple';

1;

__DATA__
__C__
int add (int x, int y) {
    return x + y;
}
```

---

---

# *Rene's Perl@Work Projekte*

Unterstützende Perl-Scripts für die PHP-  
Entwicklung vom AutomationX /  
Kühne+Nagel „Scope“ Projekt

Gewinnt 2005 den österreichischen  
Staatspreis für Transportlogistik

Hauptentwickler für AX: Rene Schickbauer

---

---

# ***Rene's Perl@Work Projekte:***

Simulation Datenkopplung  
Produktionslinien nach SAP-  
Warenwirtschaftssystem für  
AutomationX / Magna Powertrain Ilz

Simulation überprüft das eigentliche  
System vor, während und nach der  
Entwicklung. Durchgeführte Tests: > 50  
Mill. Datensätze; > 1 Mill.  
Datentransfers

Hauptentwickler für AX: Rene Schickbauer

---

---

# ***Rene's Perl@Work Projekte:***

FlexNET/FlexLM Überwachungstool für  
Magna Powertrain Lannach

Historische Aufzeichnung vergebener  
Floating-Lizenzen zur finanziell  
optimalen Ausnutzung vorhandener  
Lizenzpunkte in einem globalen  
Unternehmen mit Standorten in  
verschiedenen Zeitzonen. Anforderung:  
Graphische Auswertung

Entwicklung: Rene Schickbauer

---

# ***Rene's Perl@Home Projekte:***

Perl/TK basierter Level-Editor für  
„BlinkenSisters - Hunt for the Lost  
Pixels“

Spezielle Anforderungen: Drap-and-Drop,  
Nachbildung der BS-RenderEngine,  
komplette Ressourcenverwaltung,  
automatische Generierung von LUA-  
Scripts

Entwicklung: Rene Schickbauer, Wolfgang  
Dautermann, Jörg Ladenhauf

---

---

# *Fragen und Antworten*

„We do, what  
the customer  
wants“

---

---



# *Danke fürs Zuhören*

## Referenzen:

- <http://www.blinkensisters.org>
  - <http://www.perl.org>
  - <http://www.cpan.org>
  - <http://www.activestate.com>
  - <http://www.cygwin.com>
- 
-