



Einführung in Perl 6

Referent : **Gerd Pokorra**
Am Eicher Hang 44
57258 Freudenberg

 (dienstlich): 02 71 / 7 40 – 43 79

Referent

Studium der Elektrotechnik

seit 1988 DV-Systemspezialist

im Hochschulrechenzentrum der Universität-GH Siegen

Systemadministration: VMS -> UNIX

Zuständigkeit:

Systemsoftware, Authentisierung, Cluster

Betriebssysteme: Linux (Fedora, SuSE)

Programmiersprachen: C, Java, Perl

Anwendungssoftware: StarOffice, XML

Einführung

Akronym: **P**ractical
Extraction and
Report
Language

Pathologically
Eclectic
Rubbish
Lister
(krankhaft eklektischer Unsinn Lister)

von Larry Wall, Ende der 80er Jahre

Rakudo Star; Version \geq 2011.07

Jederzeit unterbrechen und Fragen stellen!

Geschichte

Ursprung: Bearbeitung von Textdateien (Logdateien)

Zweck: Systemadministration (über serielle Leitungen / Ost- und Westküste)

Unzufriedenheit mit awk

awk = **A**ho, **W**einberger und **K**ernighan

Freigabe der Perl-Versionen

18.12.1987 Perl 1.000

05.01.1988 Perl 2.000

18.10.1989 Perl 3.000

21.03.1993 Perl 4.000

18.10.1994 Perl 5.000

13.03.1995 Perl 5.001

22.07.1998 Perl 5.005

28.03.2000 Perl 5.6.0

02.09.2000 Perl 5.7.0

26.10.2003 Perl 5.8.1

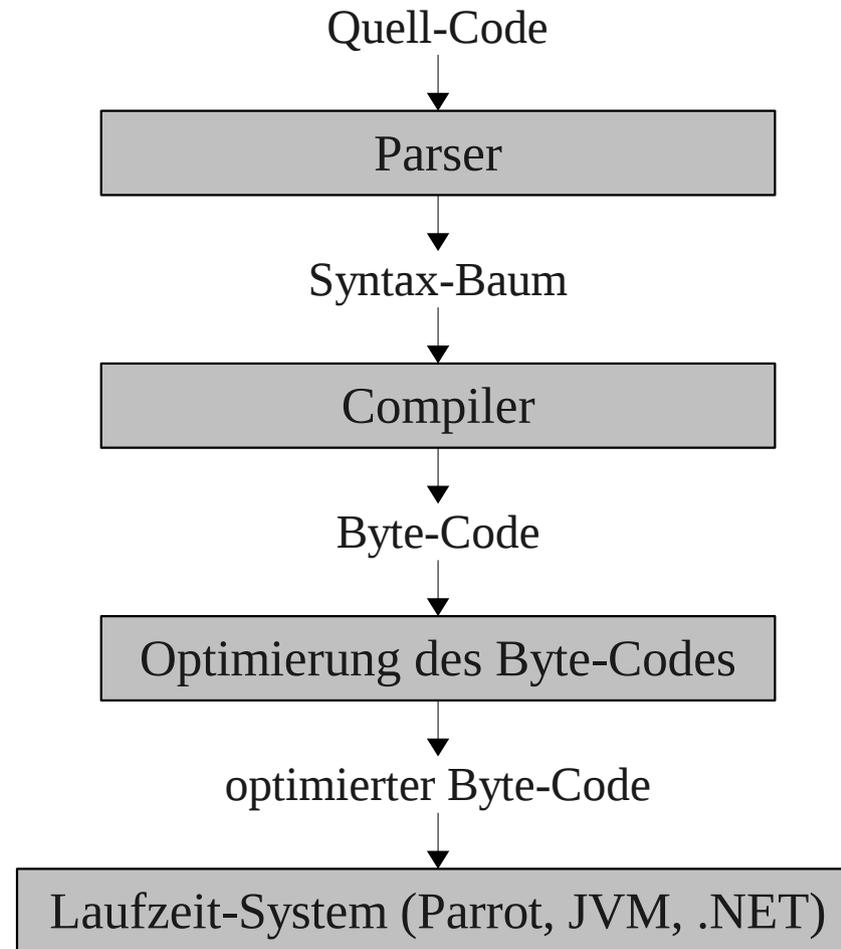
Geschichte

- 26.09.1995 CPAN eröffnet
- 21.02.1996 Perl Journal angekündigt
- 04.11.1996 Ankündigung des TPI (**T**he **P**erl **I**nstitut)
- 19.08.1997 erste Perl Konferenz / San Jose
- 09.10.1998 Larry Wall erhält FSF Award (Preis)

Geschichte

- Entscheidung für Perl 6
 - Juli 2000 bei 4. Perl-Konferenz
 - 361 RFCs reflektieren Brainstorming

Architektur



Eigenschaften

- Skript-Sprache
- viel Funktionalität mit wenig Code
- kleine Aufgaben schnell erledigen (Wegwerf-Skripte)
- Nutzung von Modulen für bestimmte Aufgaben
- läuft auf vielen Systemen
(Voraussetzung: nutzt keine systemspezifischen Funktionen)
- nutzt die besten Features von C, sed, awk und sh

Vorteile

- objektorientiert (oder prozedural)
- keine willkürlichen Limits
- dynamische Datentypen (keine Deklaration)
- modular
- portabel
- interoperabel (C)
- mächtige und leistungsfähige Ausdrücke

Nachteile

Keine!

Nachteile

- langsamer als C-Programme
- kryptisch anmutende Syntax für Anfänger (Novize, Perl-Rookie)
- schwer zu erlernen
- schwierige Fehlersuche

Woher nehmen?

- aktueller Rakudo-Quellcode

```
> git clone git://github.com/rakudo/rakudo.git  
  [<Verzeichnis-Name>]
```

- aktueller Parrot-Quellcode

```
> git clone git://github.com/parrot/parrot.git  
  [<Verzeichnis-Name>]
```

tar-Archive

- Rakudo

<https://github.com/rakudo/rakudo/downloads>

- Rakudo *

<https://github.com/rakudo/star/downloads>

Parrot-tar-Archive

- Offiziell unterstützte Versionen für Anwender:
<http://www.parrot.org/release/supported>
<ftp://ftp.parrot.org/pub/parrot/releases/stable/...>
- Entwickler-Versionen:
<http://www.parrot.org/release/developer>
<ftp://ftp.parrot.org/pub/parrot/releases/devel/...>

Installation

- Installation unter Fedora

```
> yum install rakudo-star
```

- Übersetzen der Sourcen

```
> perl Configure.pl --gen-parrot
```

Komplette Kommandosequenz

```
> git clone git://github.com/rakudo/rakudo.git
  perl6_dir

> cd perl6_dir
> perl Configure.pl --gen-parrot
> make
```

Skripte unter Windows starten

Aufruf:

```
> perl6 Skript-Name.p6
```

Voraussetzung:

die Variable **PATH** enthält den bin-Pfad der Perl 6-Installation

Überprüfen:

```
> echo %PATH%
```

Setzen:

```
> set PATH=C:\Parrot-3.7.0\bin;%PATH%
```

Die wichtigsten Kommando-Optionen

`perl6 -e Perl6Code`

Perl 6-Code: `$a=42;print "\$a=$a\n"`

- Unix

```
> perl6 -e 'my $a=42; print "\$a=$a\n"'
$a=42
>
```

- Windows

```
> perl6 -e "$a=42; print \"\$a=$a\n\""
```

\$a=42

>

Code als Parameter, von Datei oder von der Standardeingabe

Perl-Code von

1. der Kommandozeile mit der Option `-e`
2. aus einer Datei
3. von der Standardeingabe

es findet nur eine Quelle Berücksichtigung

Code als Parameter, von Datei oder von der Standardeingabe

Beispiel für Code von der Standardeingabe:

```
$ perl6  
> print "alf";  
alf> -- Ctrl/D --  
$
```

weitere Kommando-Optionen

`-h, --help:` Optionen anzeigen

`-v, --version:` Versionsdatum von Rakudo

...

Übungen: Skripte starten

1. Aufgabe

Stellen Sie fest, ob beziehungsweise welche Version von Perl 6 auf Ihrem System installiert ist.

2. Aufgabe

Führen Sie den Perl 6-Quelltext

```
print 3+4;
```

auf der Kommandozeile aus.