



Skriptsprachen im Vergleich

Python vs. Perl



Skriptsprachen im Vergleich

Agenda

1. Überblick Skriptsprachen
2. Überblick Python
3. Überblick Perl
4. Gegenüberstellung
5. Vor- und Nachteile
6. Zusammenfassung



Skriptsprachen

Überblick
Skriptsprachen

Überblick
Python

Überblick
Perl

Gegen-
überstellung

Vor- und
Nachteile

Zusammen-
fassung

- Scripting: Zusammenfassung von Kommandos zu einem Script (Drehbuch), das in einer Datei gespeichert wird und wiederverwendet werden kann
- Verzichten oft auf bestimmte Sprachelemente, z.B. Deklarationszwang
- Heute z.T. mächtige und vollwertige Programmiersprachen für nahezu alle Aufgaben nutzbar
- Skriptsprachen sehr unterschiedlich!



Abgrenzung zu Programmiersprachen

Überblick
Skriptsprachen

Überblick
Python

Überblick
Perl

Gegen-
überstellung

Vor- und
Nachteile

Zusammen-
fassung

	Skriptsprachen	Programmiersprachen
Einsatzziel	Anwendungen aus existierenden Bausteinen zusammensetzen	Neuentwicklung von Anfang an
Einsatzbereich	Oft speziell, z.B. Reporting mit Perl	Oft allgemein
Ausführung	Interpretiert oder Bytecode	Meist in Maschinen-Code übersetzt und ausgeführt
Typisierung	Typfrei oder schwach typisiert	Oft stark typisiert
Stärke	Schnellere Anwendungsentwicklung (10-20% der Codegröße)	Effiziente Ausnutzung der Ressourcen

Quelle: verändert nach Pfahler/ Thies 2009: 18



Einsatzgebiete

Überblick
Skriptsprachen

Überblick
Python

Überblick
Perl

Gegen-
überstellung

Vor- und
Nachteile

Zusammen-
fassung

- System- und Netzwerkadministration, Automatisieren von Abläufen
- Datenbankanwendungen
- Dynamische Web-Anwendungen, Client-Server-Anwendungen
- Erweiterung von Systemen und Anwendungen
- Experimentelles Programmieren, Prototyp-Entwicklung
- ...



Eigenschaften

Überblick
Skriptsprachen

Überblick
Python

Überblick
Perl

Gegen-
überstellung

Vor- und
Nachteile

Zusammen-
fassung

Features

- Automatische Speicherverwaltung
- Optional: Objektorientierung, Modulkonzepte, etc.

Mächtige Datenstrukturen für ausdruckstarken Code

- Dictionaries/ assoziative Arrays

„Höhere“ Sprachmittel

- Kontrollstrukturen

Bibliotheken und domänenspezifische Features

- Allgemein (reguläre Ausdrücke, etc.)
- Web-Anwendungen (HTML-Integration, etc.)



Skriptsprachen im Vergleich

Agenda

1. Überblick Skriptsprachen
2. Überblick Python
3. Überblick Perl
4. Gegenüberstellung
5. Vor- und Nachteile
6. Zusammenfassung



Entstehung Python

Überblick
Skriptsprachen

Überblick
Python

Überblick
Perl

Gegen-
überstellung

Vor- und
Nachteile

Zusammen-
fassung

- Entwickelt als objektorientierte Sprache von Guido van Rossum
- Benannt nach Monty Python
- Anfang 1990: Beginn der Entwicklung
- Januar 1994: Version 1.0
- Oktober 2009: Version 2.6.4
- Python 3 in Entwicklung?



Guido van Rossum





Einsatzgebiete

Überblick
Skriptsprachen

Überblick
Python

Überblick
Perl

Gegen-
überstellung

Vor- und
Nachteile

Zusammen-
fassung

- Web-Applikationen-Frameworks (Zope)
- Anbindung von Datenbanken
- GUI Entwicklungen
- Rapid Prototyping
- oft Bindeglied für XML-Datenverarbeitung und Visualisierung



Merkmale

Überblick
Skriptsprachen

Überblick
Python

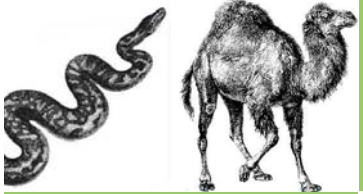
Überblick
Perl

Gegen-
überstellung

Vor- und
Nachteile

Zusammen-
fassung

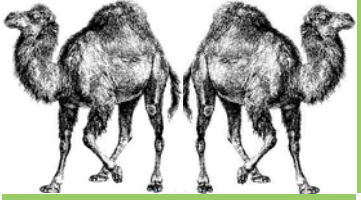
- objektorientierte Skript- und Programmiersprache
- rein dynamische Sprache (keine Typdeklaration notwendig)
- plattformunabhängig (Unix, Windows, Mac OS, etc.)
- Generell: einfach und übersichtlich
- „Batteries-included“: umfangreiche Standardbibliothek
- Python leitet den Programmierer zu strukturiertem Programmieren an!



Skriptsprachen im Vergleich

Agenda

1. Überblick Skriptsprachen
2. Überblick Python
3. Überblick Perl
4. Gegenüberstellung
5. Vor- und Nachteile
6. Zusammenfassung



Entstehung Perl

Überblick
Skriptsprachen

Überblick
Python

Überblick
Perl

Gegen-
überstellung

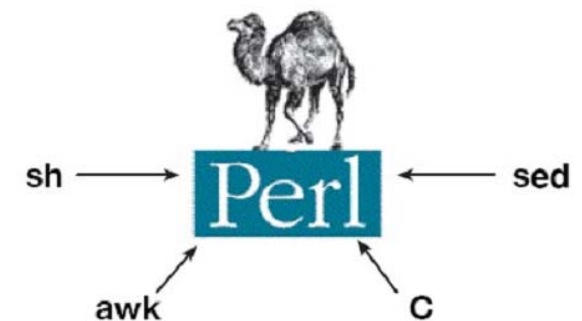
Vor- und
Nachteile

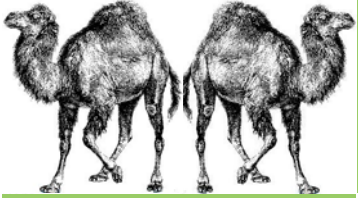
Zusammen-
fassung

- **practical extraction and report language**
- Entwickelt von Larry Wall
- Dezember 1987: Perl 1.0
- Oktober 1994: Perl 5.0
- Aktuelle Version 2009: 5.10.1
- Zukunft: Perl 6 (neue Sprache)



Larry Wall





Einsatzgebiete

Überblick
Skriptsprachen

Überblick
Python

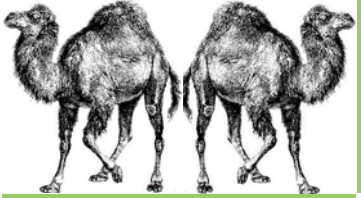
Überblick
Perl

Gegen-
überstellung

Vor- und
Nachteile

Zusammen-
fassung

- Ursprünglich:
 - Datenbankabfragen,
 - Werkzeug zur Verarbeitung und Manipulation von Textdateien insbesondere bei System- und Netzwerkadministration
- Heute:
 - Auch Entwicklung von Webanwendungen



Merkmale

Überblick
Skriptsprachen

Überblick
Python

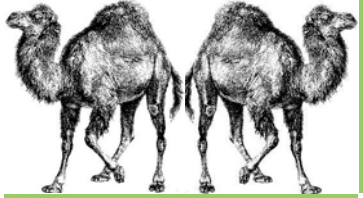
Überblick
Perl

Gegen-
überstellung

Vor- und
Nachteile

Zusammen-
fassung

- „Schweizer Messer“ für Systemadministratoren
 - Bsp. Auswertung von Logfiles
- Schnelle Problemlösung: gut für quick prototyping
- Größtmögliche Freiheit für Programmierer
- Mehrere Wege: TIMTOWTDI „There is more than one way to do it“
- Einfach und möglich: „Perl makes easy jobs easy and hard jobs possible“
- kontextsensitiv



Perl-Kultur und Spass

Überblick
Skriptsprachen

Überblick
Python

Überblick
Perl

Gegen-
überstellung

Vor- und
Nachteile

Zusammen-
fassung

■ Kreativer Programmcode → Wettbewerbe und Traditionen

- Golf
- Poesie
- Obfuscation
- JAPH
- Perligata
- Acme

```

#
sub j(\$(\$(\$(
    0;while($$P=~s:
    $V+=('!eq$1)-32:31
    )$V+=ord( substr(
    $P,0,1,""))-74) sub a(
    my($I,$K,$ J,$L)=0_
    ;$I=int($I*$M/$Z);$K=int(
    $K*$M/$Z);$J=int($J*$M
    /$Z);$L=int($L*$M/$Z); $G=$
    J-$I;$F=$L-$K;$E=(abs($
    G)>abs($F))?$G:$F;($E<0) and($
    I,$K)=$J,$L;$E|=0.01
    +int($I*$F/$E)
    +int($I*$F/$E)
    Z=$z|.01;map(
    $H=$_;$I=$N=$H;$K=$O=$H;while($H){$q=ord
    substr($H,0,1,"");if(42==$q){$J=$H;$L=$H}else{
    $q=9;$J=$q-$L/9;$L=$q-9*$J-4;$J=4;$J+=$I;$L+=$K;a($I,$K,$J,$L);
    ($I,$K)=$J,$L)a($I,$K,$N,$O)0;my$T:map($y=$_);map($T.-$D->($y
    ->{$_}?)$y:' ')-59)..59);$T.="n"}(-23..23);print"e[H$T]"$w=eval
    require Win32::Console::ANSI;$b=$w?'1;7;':'';($j,$u,$s,$t,$a,$n,$o
    ,$h,$c,$k,$p,$e,$r,$l,$C)=split//,'T*JSK8Iag*PJ[*J@wR)*JR)*QJ[*J'
    'BA*JQK8I*JC)KUZ)BAIJT)*Q[R?-R[e]'RI'.'Tn*JQ)wRAI*JdnR8QAU)wBKT'
    'Jn*JEI*EJR*QJ)*JR*DJ@IQ()*JSe*JD[n)*JPe*'. 'JBI/RI)T8@?PcdnfgVBCRcP'
    '?ABKV])*JWe*JD[n)*JPe*JC?8B*JE);Vq*OJQ/IP['. 'wQ)*JWeOe(n*EERk8;P'
    'J*JC)/U*OJd[OI@*BJ*JXn*J>w)U)Cwq*OJc8KJ?O[e]U/T*QJP?)*JSe*JCaTe'
    'QIAKJR)*JV)wRAI*J?)T)*RJoJI[3;U)Uq*PM[wW]U)UCT*DM*5J'. '2P[2'
    'PZa[}UKVgogK9K*QU[}n[RIB*EH@IddR[Q]T)T)T3o[dk*JE'. '[2\U'
    '(T)*JPKTKK)*OJ[QIO[PIQIO[}gUKU)k*JE+J+J5R5AI*EJ00'. 'BCB*
    'DNKKJIR[Q+*EJO*ER';sub h($)=qw(%εθx)[int rand
    4];map(printf "%e[%b;%dm",int(rand 6)+101-60* $w
    |[0];system( "cls")if$w ;($A,$S)= ($_[1], $
    _[0]);($M,
    0,)= split ' ');for( $z=256
    :$z>0; $z --$S){$S*= $A;p 0,} sleep$
    [2];while ($_[3]εε($ z+=$ S) <=256){
    p0,}){"" "32)7D$3" .")AG". "$u)OG"
    ."$s)WG" ."$t", "" "24)" " "IJ$a"
    .")1G$n" .")CO$o" .")GG$t" .")QC"
    ."$h" .")G$e" .")IG" ."$z",
    "32)?" .")H$P)FG$e)QG$r". ")ZC"
    ."$1", "28)(LC" ."" .""
    "$h):" .")J$a)EG". "$c"
    .")H" .")C$k)ZG". "$e"
    .")" .")d$e", "18" .")("
    .")D:" .")C" )}(h(16 ,1,1,0
    );h(8, .98,0,0 );h(16 ,1,1,1)
    ;h(8.0 ,0.98,0, 1); redo)###
    #wzitten
    060204 by
    
```

Quelle: http://de.wikipedia.org/wiki/Just_another_Perl_hacker



Skriptsprachen im Vergleich

Agenda

1. Überblick Skriptsprachen
2. Überblick Python
3. Überblick Perl
4. Gegenüberstellung
5. Vor- und Nachteile
6. Zusammenfassung



Unterschiede

Überblick
Skriptsprachen

Überblick
Python

Überblick
Perl

Gegen-
überstellung

Vor- und
Nachteile

Zusammen-
fassung

Python

- One obvious way to do it
 - Sehr lesbar
 - Wenige Spezialfälle
 - Allgemein konzipiert
 - Weniger Freiheit → weniger Verantwortung

Perl

- More than one way to do it
 - Sehr kompakt
 - Viele Abkürzungen
 - Auf Scripting zugeschnitten
 - Viel Freiheit → Verantwortung!



Beispielprogramm in Python

Überblick
Skriptsprachen

Überblick
Python

Überblick
Perl

Gegen-
überstellung

Vor- und
Nachteile

Zusammen-
fassung

```
#!/usr/bin/python

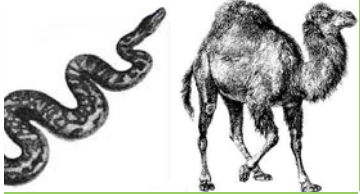
# Dieses Programm durchsucht die Passwd Datei und
# gibt alle vorkommenden Login Shells aus

passwd = "/etc/passwd";           # Passwd Datei

# Unterfunktion
def process_passwd(file):
    tmp = {}                       # temporäres Dictionary
    f = open(file)                 # Datei öffnen
    for line in f.readlines():      # zeilenweise lesen
        arr = line.strip().split(":") # CR/LF entfernen, in Liste splitten
        tmp[arr[6]] = None         # Login Shell in Dictionary schreiben
    f.close()                       # Datei schließen
    shells = []                    # leere Schell-Liste erzeugen
    for sh in tmp.keys():           # Schleife über alle Indizes
        shells.append(sh)          # Schlüssel an Liste anhängen
    return shells                  # Array mit Shells zurückgeben

# Aufruf der Unterfunktion. Array mit Login Shells wird zurückgegeben.
shells = process_passwd(passwd)

# Alle Shells in einer Schleife ausgeben
for sh in shells:
    print "Login Shell:", sh
```



Beispielprogramm in Perl

Überblick
Skriptsprachen

Überblick
Python

Überblick
Perl

Gegen-
überstellung

Vor- und
Nachteile

Zusammen-
fassung

```
#!/usr/bin/perl -w
# Dieses Programm durchsucht die Passwd Datei und
# gibt alle vorkommenden Login Shells aus
my ($sh);
my $passwd = "/etc/passwd";          # Passwd Datei
# Aufruf der Unterfunktion. Array mit Login Shells wird zurückgegeben.
my @shells = process_passwd($passwd);
# Alle Shells in einer Schleife ausgeben
foreach $sh (@shells) {
    print "Login Shell: $sh\n";
}
# Unterfunktion
sub process_passwd {
    my $file = shift(@_);            # Übergebenes Argument
    my ($line,$sh,@shells,%tmp,@arr); # lokale Variablen
    open(F,"<$file");               # Datei öffnen
    while( $line = <F> ) {           # zeilenweise lesen
        chomp($line);                # CR/LF entfernen
        @arr = split(/:/,$line);     # String in Array splitten
        $tmp{$arr[6]} = 1;           # Login Shell in Hash schreiben
    }
    close(F);                        # Datei schliessen
    foreach $sh (keys %tmp) {         # Schleife über alle Schlüssel des Hash
        push(@shells,$sh);           # Schlüssel in Array schreiben
    }
    return(@shells);                 # Array mit Shells zurückgeben
}
```



Skriptsprachen im Vergleich

Agenda

1. Überblick Skriptsprachen
2. Überblick Python
3. Überblick Perl
4. Gegenüberstellung
5. Vor- und Nachteile
6. Zusammenfassung



Allgemeine Vorteile

Überblick
Skriptsprachen

Überblick
Python

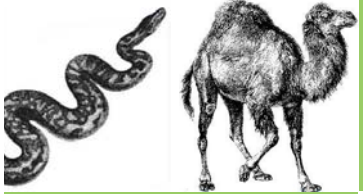
Überblick
Perl

Gegen-
überstellung

Vor- und
Nachteile

Zusammen-
fassung

- Schnelle und einfache Erstellung von kleineren Programmen
- Je nach Sprache Formalitäten aktivierbar
- Kurze Entwicklungszyklen und kompakter Code
- Mächtige Konstrukte, hohe Flexibilität
- Testing ist relativ leicht



Vorteile Python - Perl

Überblick
Skriptsprachen

Überblick
Python

Überblick
Perl

Gegen-
überstellung

Vor- und
Nachteile

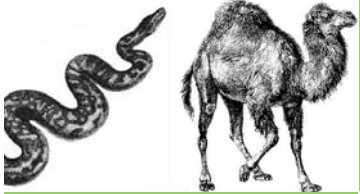
Zusammen-
fassung

Python

- Leicht zu lernen , gut lesbar und leistungsfähig
- Gut skalierbar
- Java-Integration mit Jython relativ einfach
- Zur Erstellung von Prototypen geeignet

Perl

- CPAN: sehr umfangreiches Perl-Modul-Archiv
- Sehr ausdrucksstark und flexibel
- Mächtige Tools für Dokumentation und Testing
- Für (fast) alles geeignet



Nachteile

Überblick
Skriptsprachen

Überblick
Python

Überblick
Perl

Gegen-
überstellung

Vor- und
Nachteile

Zusammen-
fassung

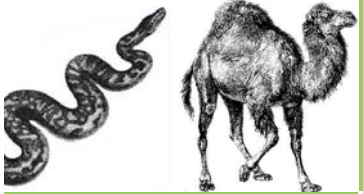
- Für systemnahe Programmierung weniger geeignet
- Höherer Speicherverbrauch

Perl

- Code kann schwer lesbar sein → Spaghetticode
- „There is more than one way to do it“

Python

- Strukturierung durch Einrücken der Zeilen
- Noch nicht so viele Module wie in Perl



Skriptsprachen im Vergleich

Agenda

1. Überblick Skriptsprachen
2. Überblick Python
3. Überblick Perl
4. Gegenüberstellung
5. Vor- und Nachteile
6. Zusammenfassung



Zusammenfassung

Überblick
Skriptsprachen

Überblick
Python

Überblick
Perl

Gegen-
überstellung

Vor- und
Nachteile

Zusammen-
fassung

- Nicht für systemnahe Programmierung, gut für „kleine“ Aufgaben
- Schnelle Entwicklung, langsamere Laufzeit
- „Know your tools“
- Spezialgebiete Perl:
 - Unterstützung von regulären Ausdrücken
 - Dateimanagement
 - Report-Funktionen
- Spezialgebiete Python:
 - Datenstrukturen
 - Objektorientierung
 - Lesbarkeit des Codes



Kriterien für die Sprachauswahl

- Sprach-Features: Ausreichend und zur Aufgabe passend?
- Verfügbarkeit von Erweiterungen, Modulen, Frameworks
- Support, Community-Unterstützung?
- Umfang des Projektes, evtl. spätere Erweiterbarkeit
- Externe Bibliotheken benötigt?
- ...



Beispiel: Geoprocessing in ArcGIS

```

7% *Untitled*
File Edit Format Run Options Windows Help

import sys, string, os, win32com.client

gp = win32com.client.Dispatch(„esriGeoprocessing.
GpDispatch.1“)

gp.SetProduct(„ArcEditor“)

gp.OverwriteOutput=1

gp.AddToolbox(„C:/GIS/ESRI/ArcGIS/ArcToolbox/
Toolboxes/Analysis Tools.tbx“)

punkte_Buffer_shp = „N:/daten/punkte_Buffer.shp“
punkte_shp = „N:/daten/punkte.shp“

gp.Buffer_analysis(punkte_shp, punkte_Buffer_shp,
„100,000000 Meters“, „FULL“, „ROUND“, „NONE“, „“)
  
```

```

7% *Untitled*
File Edit Format Run Options Windows Help

use Win32::OLE;
my $gp = Win32::OLE->new(„esriGeoprocessing.GPDispatch.1“);
$gp->SetProduct(„ArcEditor“);
$gp->{OverwriteOutput} = 1;
$gp->AddToolbox(„C:/GIS/ESRI/ArcGIS/ArcToolbox/
Toolboxes/Analysis Tools.tbx“);
my $punkte_Buffer_shp = „N:/daten/punkte_Buffer.shp“;
my $punkte_shp = „N:/daten/punkte.shp“;
$gp->Buffer_analysis($punkte_shp, $punkte_Buffer_shp,
„100,000000 Meters“, „FULL“, „ROUND“, „NONE“, „“);
  
```



Quellen und weiterführende Links

- Vorlesungsunterlagen „Skriptsprachen zur Realisierung dynamischer Webanwendungen“ URL: <http://ag-kastens.uni-paderborn.de/lehre/material/skriptsprachen/> (Letzter Zugriff: 16.12.2009)
- Freude A.C.H. (2008): Perl, PHP, Python – Ein Vergleich. URL: <http://alvar.a-blast.org/vortraege/scriptsprachen-vergleich.pdf> (Letzter Zugriff: 16.12.2009)
- Wehner, V. (2003): Perl und Python. Hausarbeit im Rahmen des Hauptseminars Programmiersprachen. Uni Passau
- [http://de.wikipedia.org/wiki/Python_\(Programmiersprache\)](http://de.wikipedia.org/wiki/Python_(Programmiersprache))
- <http://de.wikibooks.org/wiki/Python-Programmierung>
- <http://www.python.org/>
- [http://de.wikipedia.org/wiki/Perl_\(Programmiersprache\)](http://de.wikipedia.org/wiki/Perl_(Programmiersprache))
- <http://de.wikibooks.org/wiki/Perl-Programmierung>
- <http://www.perl.org/>