

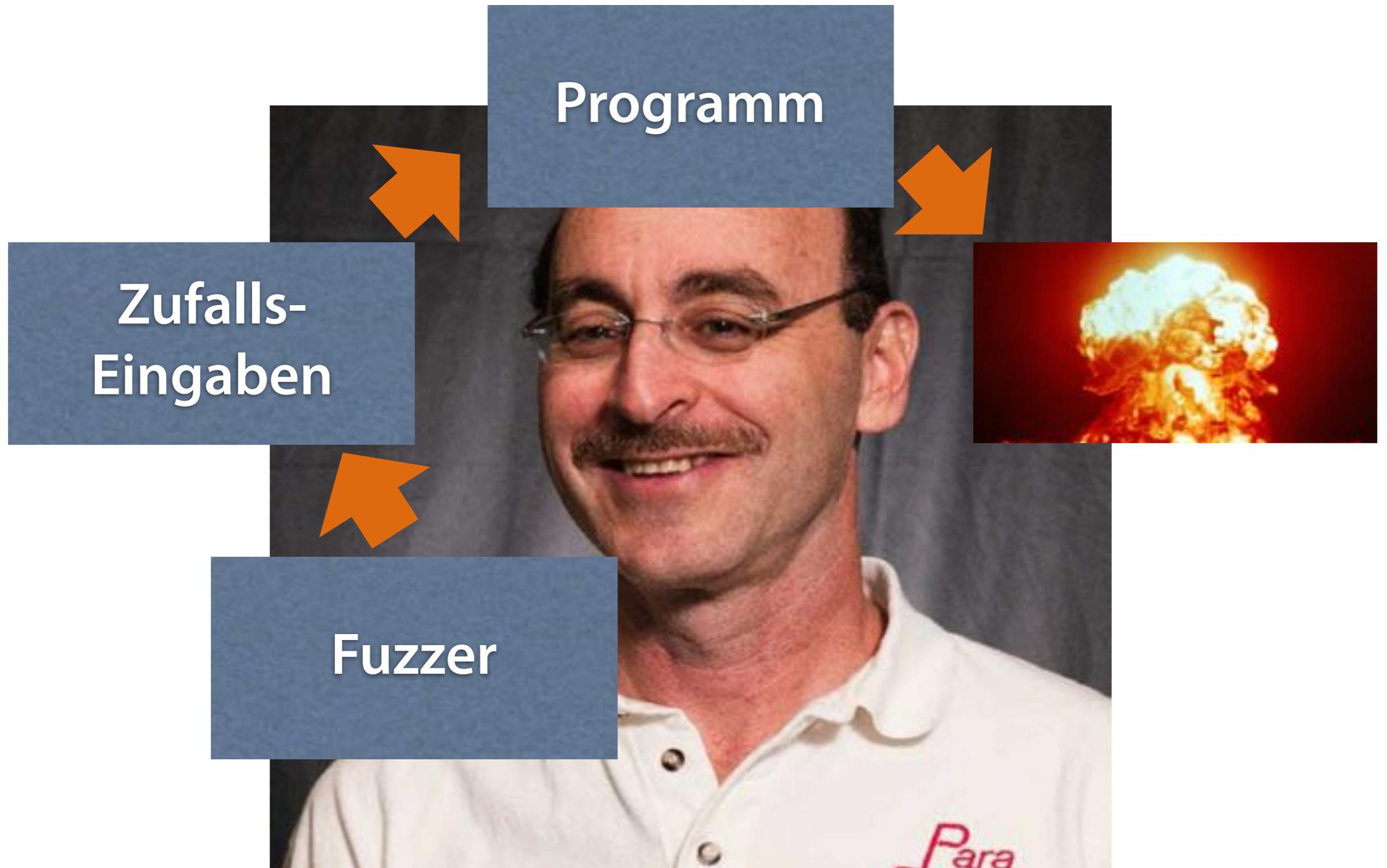
# Testgenerierung + Fehlersuche

Software-Praktikum  
Andreas Zeller, Universität des Saarlandes



# Fuzzing

Zufallstesten auf Systemebene (1989)



# Fuzzer-Ausgabe

[ ;x1-GPZ+wcckc];,N9J+?#6^6\e?]9lu2\_%'4GX"0VUB[E/r  
~fApu6b8<{%siq8Zh.6{V,hr?;{Ti.r3PIxMMMv6{xS^+'Hq!  
AxB"YXRS@!Kd6;wtAMefFWM(`|J\_<1~o}z3K(CCzRH  
JIIvHz>\_\*. \>JrLU32~eGP?lR=bF3+;y\$3lodQ<B89!  
5"W2fK\*vE7v{' )KC-i,c{<[~m!]o;{.'}Gj\ (X}  
EtYetrpbY@aGZ1{P!AZU7x#4(Rtn!q4nCwqol^y6}0|  
Ko=\*JK~;zMKV=9Nai:wxu{J&UV#HaU)\*BiC<),`+t\*gka<W=Z.  
%T5WGHZpI30D<Pq>&]BS6R&j?#tP7iaV}-}`\?[\_ [Z^LBMPG-  
FKj'\xwuZ1=Q`^`5,\$N\$Q@[!CuRzJ2D|vBy!^zkhdf3C5PAkR?  
V hn|3='i2Qx]D\$qs40`1@fevnG'2\11Vf3piU37@55ap  
\zIyl"'f,\$ee,J4Gw:cgNKLie3nx9(`efSlg6#[K"@WjhZ}  
r[Scun&sBCS,T[/vY'pduwgzDlVNy7'rnzxNwI)(ynBa>%|  
b`;`9fG]P\_0hdG~\$@6 3]KAeEnQ7lU)3Pn,0)G/6N-wyzj/  
MTd#A;r

# fuzzer.py

```
import random

def fuzzer():
    # Strings up to 1024 characters long
    string_length = int(random.random() * 1024)

    # Fill it with ASCII 32..128 characters
    out = ""
    for i in range(0, string_length):
        out += chr(int(random.random() * 96 + 32))
    return out

if __name__ == "__main__":
    print fuzzer()
```

# UNIX-Werkzeuge fuzzen

- Nutze Fuzz-Ausgabe als Eingabe für lex:  
`$ python fuzzer.py | lex`
- Nutze Fuzz-Ausgabe als Eingabe für grep:  
`$ python fuzzer.py | grep x`
- Nutze Fuzz-Ausgabe als TeX-Dokument:  
`$ python fuzzer.py | tex`

# 1989 Paper

## **An Empirical Study of the Reliability of UNIX Utilities**

*Barton P. Miller*  
*bart@cs.wisc.edu*

*Lars Fredriksen*  
*L.Fredriksen@att.com*

*Bryan So*  
*so@cs.wisc.edu*

### **Summary**

Operating system facilities, such as the kernel and utility programs, are typically assumed to be reliable. In our recent experiments, we have been able to crash 25-33% of the utility programs on any version of UNIX that was tested. This report describes these tests and an analysis of the program bugs that caused the crashes.

# Ergebnisse

Utility	VAX (v)	Sun (s)	HP (h)	i386 (x)	AIX 1.1 (a)	Sequent (d)
adb	●○	●	●	○	-	-
as	●			●	●	●
awk						
bc				●○		
bib			-	-	-	-
calendar				-		
cat						
cb	●		●	●	○	●
cc						
/lib/ccom				-	-	●
checkeq				-		
checknr				-	-	
col	●○	●	●	●○	●	●
colcrt				-	-	
colrm				-	-	
comm						
compress					-	
/lib/cpp						
csk						

deroff	●	●	●		●	●
diction	●	-	●		-	●
diff						
ditroff	●○	●	-	-	-	
dtbl			-	-	-	-
emacs	●	●	○	-	-	
eqn		●	●	●		
expand					-	
f77	●		-	-	-	-
fmt						
fold					-	
ftp	●	●	●	-	●	●
graph					-	
grep						
grn			-	-	-	-
head					-	
ideal			-	-	-	-
indent	●○	●○	●	-	-	●
join		⊕				
latex			-	-	-	-
lex	●	●	●	●	●	●
lint						
lisp		-		-	-	-
look	●	○	●	●	-	●

# Ergebnisse

Utility	VAX (v)	Sun (s)	HP (h)	i386 (x)	AIX 1.1 (a)	Sequent (d)
adb	●○	●	●	○	-	-
as	●			●	●	●
awk						
bc				●○		
bib			-	-	-	-
calendar						
cat						
cb	●		●	●	○	●
cc						
/lib/ccom						●
checkeq						
checknr						
col	●○	●	●	●○	●	●
colcrt						
colrm						
comm						
compress					-	
/lib/cpp						
csd	●○	○	○	-	○	○
dbx		*	-	-		
dc				○		
deqn		●	-	-	-	-
deroff	●	●	●		●	●
diction	●	-	●		-	●
diff						
ditroff	●○	●	-	-	-	-
dtbl						
emacs	●	●	○	-	-	
eqn		●	●	●		
expand					-	
f77	●		-	-	-	-
fmt						
fold					-	
ftp	●	●	●	-	●	●
graph						
grep						
grn			-	-	-	-
head						
ideal						
indent	●○	●○	●	-	-	●
join		⊕				
latex						
lex	●	●	●	●	●	●
lint						
lisp		-		-	-	-
look	●	○	●	●	-	●

Table 2: List of Utilities Tested and the Systems on which They Were Tested (part 1)

● = utility crashed, ○ = utility hung, \* = crashed on SunOS 3.2 but not on SunOS 4.0,  
 ⊕ = crashed only on SunOS 4.0, not 3.2. - = utility unavailable on that system.  
 ! = utility caused the operating system to crash.

Utility	VAX (v)	Sun (s)	HP (h)	i386 (x)	AIX 1.1 (a)	Sequent (d)
m4				●		
mail						
make			●			
more					-	
nm						
nroff				●		
pc				-	-	-
pic			-	-	-	-
plot	-	○	●	-	-	-
pr						-
prolog	●○	●○	●○	-	-	-
psdit				-	-	
ptx	-	●	●	○		○
refer	●	*	●	-	-	!●
rev				-	-	
sed						
sh				-		
soelim					-	
sort						
spell	●○	●	●	○	●	●
spline					-	
split						
sql		-			-	-
strings						
strip						
style	●	-	●		-	●
sum						
tail						
tbl						
tee						
telnet	●	●	●	-	●	○
tex			-	-	-	-
tr						
troff	-	-	-			
tsort	●	*	●	●	●	●
ul	●	●	●	-	-	●
uniq	●	●	●	●	●	●
units	●○	●	●	●	●	●
vgrind	●		-	-	-	
vi	●		●	-		
wc						
yacc						
# tested	85	83	75	55	49	73
# crashed/hung	25	21	25	16	12	19
%	29.4%	25.3%	33.3%	29.1%	24.5%	26.0%

Table 2: List of Utilities Tested and the Systems on which They Were Tested (part 2)

● = utility crashed, ○ = utility hung, \* = crashed on SunOS 3.2 but not on SunOS 4.0,  
 ⊕ = crashed only on SunOS 4.0, not 3.2. - = utility unavailable on that system.  
 ! = utility caused the operating system to crash.

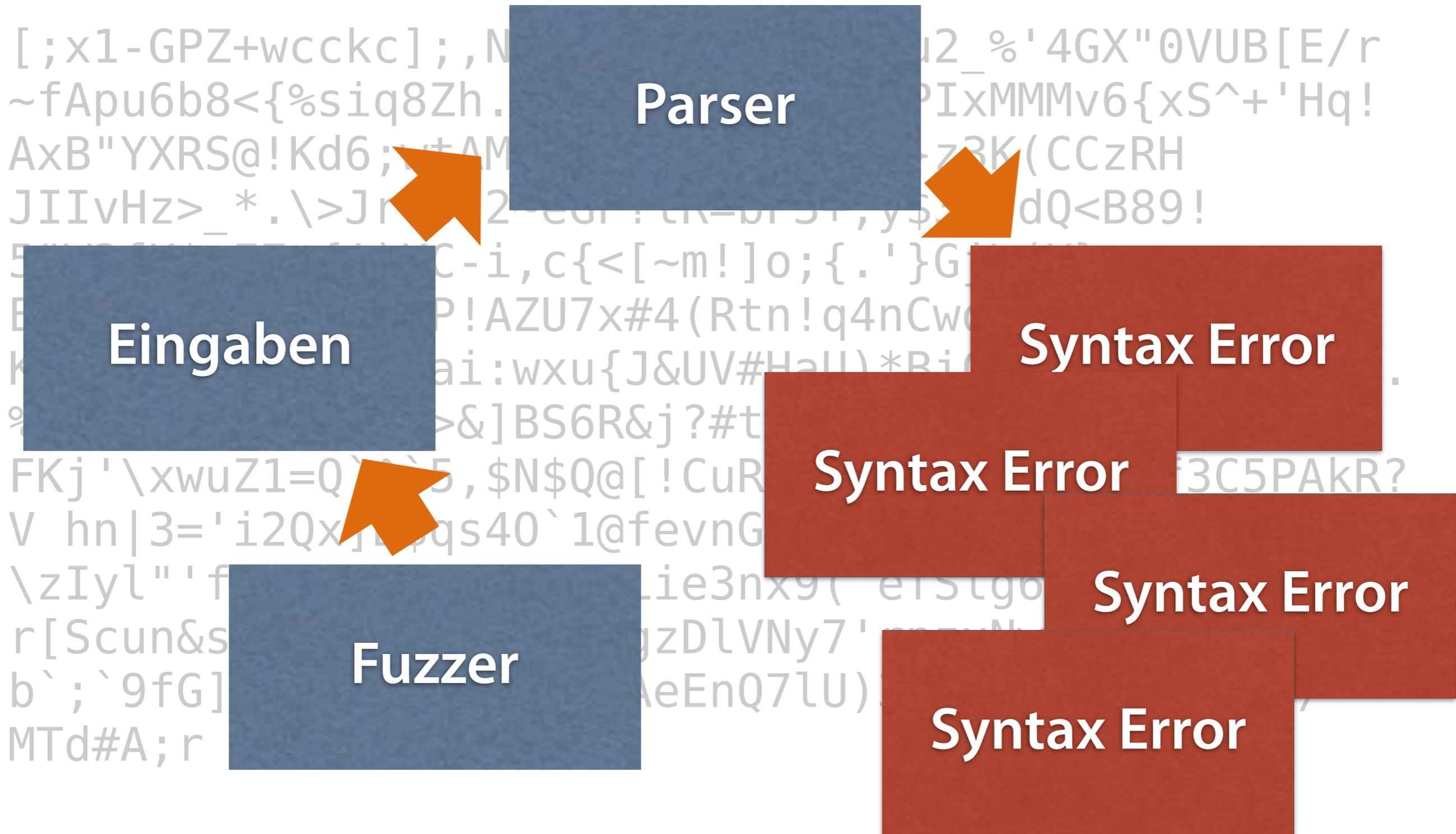
# Einen Parser testen

- Wir möchten einen *Parser* testen – ein Programm, das eine *Sprache* annimmt (z.B. Programmcode)

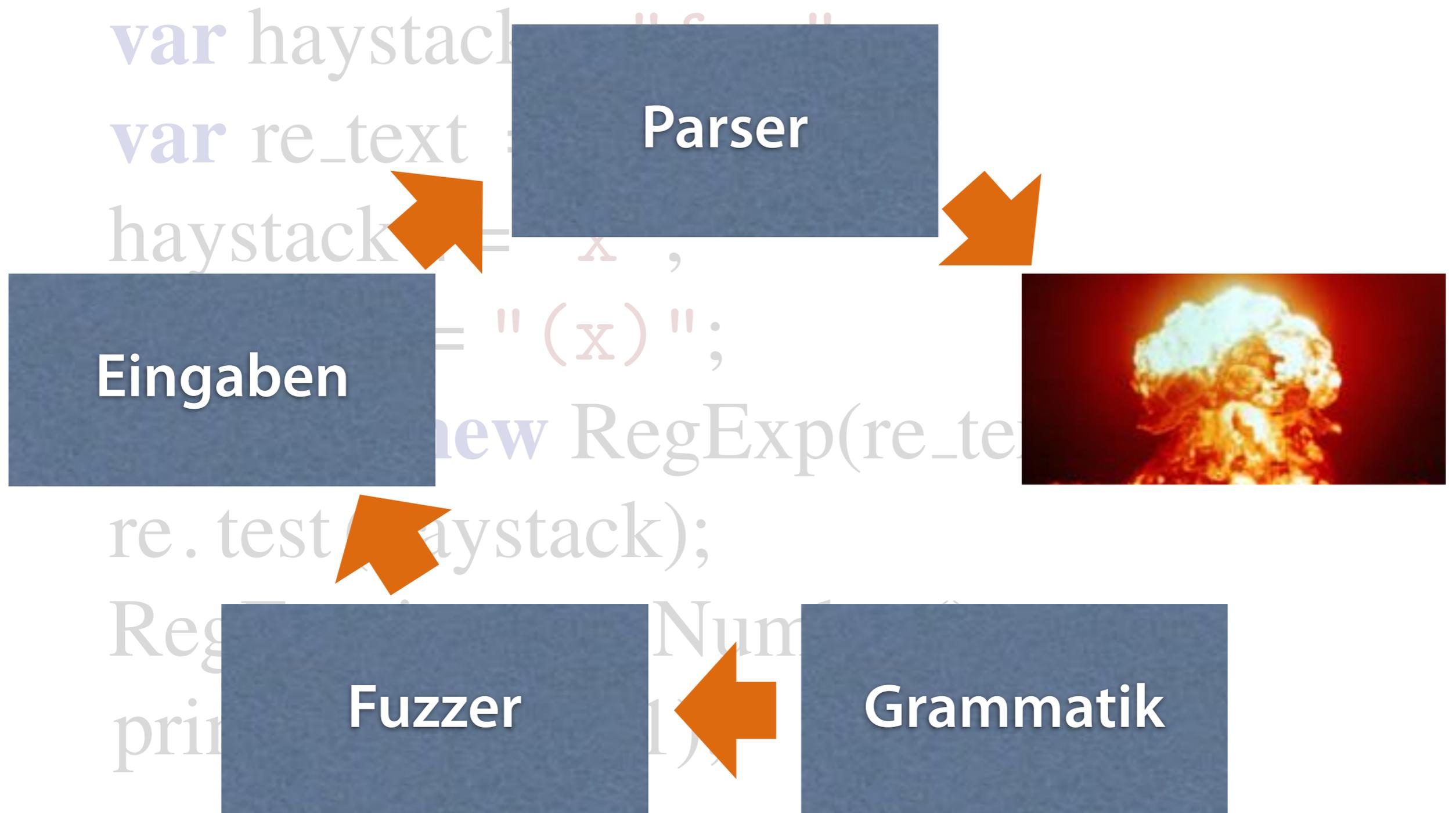


Parser

# Standard-Fuzzing



# Grammatik-Fuzzing



# grammar-fuzz.py

- Wir codieren eine *Grammatik*, die arithmetische Ausdrücke *as strings* erzeugt
- $\$START$  wird zu  $\$EXPR$ ; dieses kann zu  $\$TERM$ ,  $\$TERM + \$TERM$ , etc. expandieren

```
grammar = [  
    ("START", "EXPR"),  
  
    ("EXPR", "EXPR + TERM"),  
    ("EXPR", "EXPR - TERM"),  
    ("EXPR", "TERM"),
```

# grammar-fuzz.py

```
grammar = [  
    (" $START", "$EXPR" ),  
  
    (" $EXPR", "$EXPR + $TERM" ),  
    (" $EXPR", "$EXPR - $TERM" ),  
    (" $EXPR", "$TERM" ),  
  
    (" $TERM", "$TERM * $FACTOR" ),  
    (" $TERM", "$TERM / $FACTOR" ),  
    (" $TERM", "$FACTOR" ),  
  
    (" $FACTOR", "+$FACTOR" ),  
    (" $FACTOR", "-$FACTOR" ),  
    (" $FACTOR", "($EXPR)" ),  
    (" $FACTOR", "$INTEGER" ),  
    (" $FACTOR", "$INTEGER.$INTEGER" ),
```

```
(" $FACTOR" , "+ $FACTOR" ) ,  
(" $FACTOR" , "- $FACTOR" ) ,  
(" $FACTOR" , "( $EXPR )" ) ,  
(" $FACTOR" , "$INTEGER" ) ,  
(" $FACTOR" , "$INTEGER.$INTEGER" ) ,
```

```
(" $INTEGER" , "$INTEGER$DIGIT" ) ,  
(" $INTEGER" , "$DIGIT" ) ,
```

```
(" $DIGIT" , "1" ) ,  
(" $DIGIT" , "2" ) ,  
(" $DIGIT" , "3" ) ,  
(" $DIGIT" , "4" ) ,  
(" $DIGIT" , "5" ) ,  
(" $DIGIT" , "6" ) ,  
(" $DIGIT" , "7" ) ,  
(" $DIGIT" , "8" ) ,  
(" $DIGIT" , "9" ) ,  
(" $DIGIT" , "0" )
```

```
( "$INTEGER", "$INTEGER$DIGIT" ),  
( "$INTEGER", "$DIGIT" ),
```

```
( "$DIGIT", "1" ),  
( "$DIGIT", "2" ),  
( "$DIGIT", "3" ),  
( "$DIGIT", "4" ),  
( "$DIGIT", "5" ),  
( "$DIGIT", "6" ),  
( "$DIGIT", "7" ),  
( "$DIGIT", "8" ),  
( "$DIGIT", "9" ),  
( "$DIGIT", "0" )
```

```
]
```

```
def apply(term, rule):  
    (old, new) = rule  
    # We replace the first occurrence;  
    # this could also be some random occurrence  
    return term.replace(old, new, 1)
```

```
def apply(term, rule):  
    (old, new) = rule  
    # We replace the first occurrence;  
    # this could also be some random occurrence  
    return term.replace(old, new, 1)
```

```
MAXSYMBOLS = 5
```

```
def produce():  
    term = "$START"  
    while term.count('$') > 0:  
        # All rules have the same chance;  
        # this could also be weighted  
        index = random.randint(0, len(grammar) - 1)  
        new_term = apply(term, grammar[index])  
        if (new_term != term and  
            new_term.count('$') < MAXSYMBOLS):  
            print new_term  
            term = new_term  
    return term
```

\$EXPR

\$EXPR - \$TERM

\$EXPR + \$TERM - \$TERM

\$EXPR + \$TERM \* \$FACTOR - \$TERM

\$TERM + \$TERM \* \$FACTOR - \$TERM

\$TERM + \$TERM \* -\$FACTOR - \$TERM

\$FACTOR + \$TERM \* -\$FACTOR - \$TERM

-\$FACTOR + \$TERM \* -\$FACTOR - \$TERM

—\$FACTOR + \$TERM \* -\$FACTOR - \$TERM

—\$FACTOR + \$FACTOR \* -\$FACTOR - \$TERM

——\$FACTOR + \$FACTOR \* -\$FACTOR - \$TERM

——\$FACTOR + \$FACTOR \* -\$FACTOR - \$FACTOR

——+\$FACTOR + \$FACTOR \* -\$FACTOR - \$FACTOR

——+-\$F

——+-\$I

——+-\$D

——+-2

——+-2 + \$INTEGER.\$INTEGER \* -\$FACTOR - \$FACTOR

——+-2 + \$INTEGER.\$INTEGER \* -+\$FACTOR - \$FACTOR

——+-2 + \$INTEGER.\$INTEGER \* -+\$INTEGER - \$FACTOR

——+-2 + \$DIGIT.\$INTEGER \* -+\$INTEGER - \$FACTOR

——+-2 + 3.\$INTEGER \* -+\$INTEGER - \$FACTOR

——+-2 + 3.\$INTEGER \* -+\$INTEGER - +\$FACTOR

——+-2 + 3.\$INTEGER \* -+\$INTEGER - +\$INTEGER

——+-2 + 3.\$DIGIT \* -+\$INTEGER - +\$INTEGER

——+-2 + 3.5 \* -+\$INTEGER - +\$INTEGER

——+-2 + 3.5 \* -+\$DIGIT - +\$INTEGER

——+-2 + 3.5 \* -+1 - +\$INTEGER

——+-2 + 3.5 \* -+1 - +\$DIGIT

——+-2 + 3.5 \* -+1 - +5

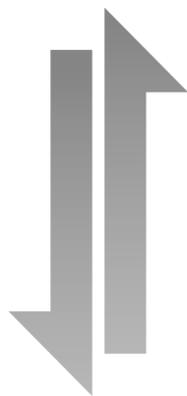
-----+-2 + 3.5 \* -+1 - +5

# LangFuzz (2012)



- Fuzz-Tester für Parser
- Nutzt *Grammatik*, um Eingabeformate zu beschreiben
- Nutzt Grammatik, um *bestehende Eingaben* zu verarbeiten

# Anwendungsgebiet: JavaScript



- Kontrolliert ein Angreifer den *JavaScript-Interpreter*, kontrolliert er den gesamten Browser

# JavaScript-Grammatik

## If Statement

*IfStatement*<sup>full</sup> ⇒

**if** ParenthesizedExpression Statement<sup>full</sup>

| **if** ParenthesizedExpression Statement<sup>noShortIf</sup> **else** Statement<sup>full</sup>

*IfStatement*<sup>noShortIf</sup> ⇒ **if** ParenthesizedExpression Statement<sup>noShortIf</sup> **else** Statement<sup>noShortIf</sup>

## Switch Statement

*SwitchStatement* ⇒

**switch** ParenthesizedExpression { }

| **switch** ParenthesizedExpression { CaseGroups LastCaseGroup }

*CaseGroups* ⇒

«empty»

| CaseGroups CaseGroup

*CaseGroup* ⇒ CaseGuards BlockStatementsPrefix

*LastCaseGroup* ⇒ CaseGuards BlockStatements

*CaseGuards* ⇒

CaseGuard

| CaseGuards CaseGuard

*CaseGuard* ⇒

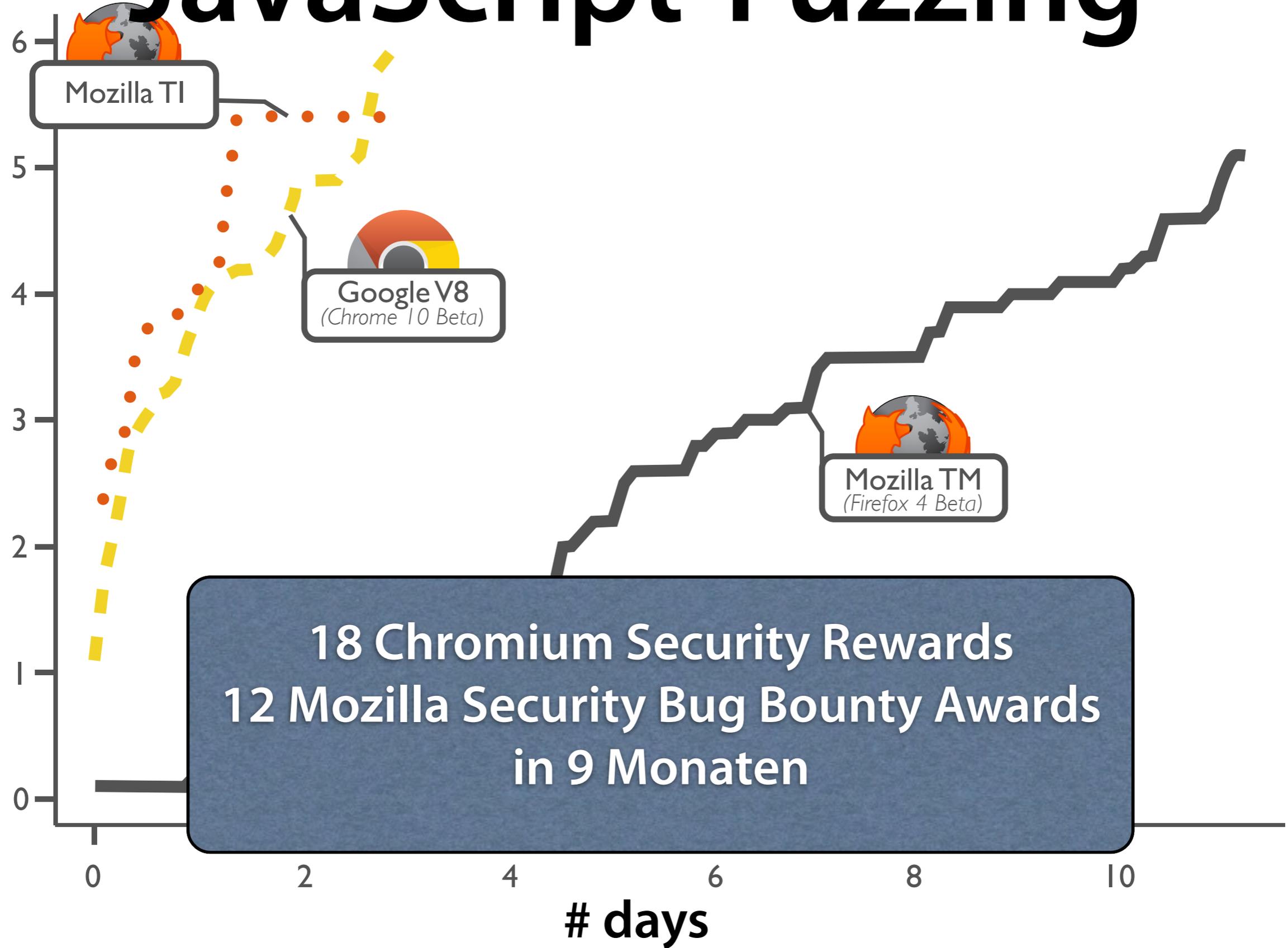
# JavaScript des Todes

```
var haystack = "foo";  
var re_text = "^foo";  
haystack += "x";  
re_text += "(x)";  
var re = new RegExp(re_text);  
re.test(haystack);  
RegExp.input = Number();  
print(RegExp.$1);
```



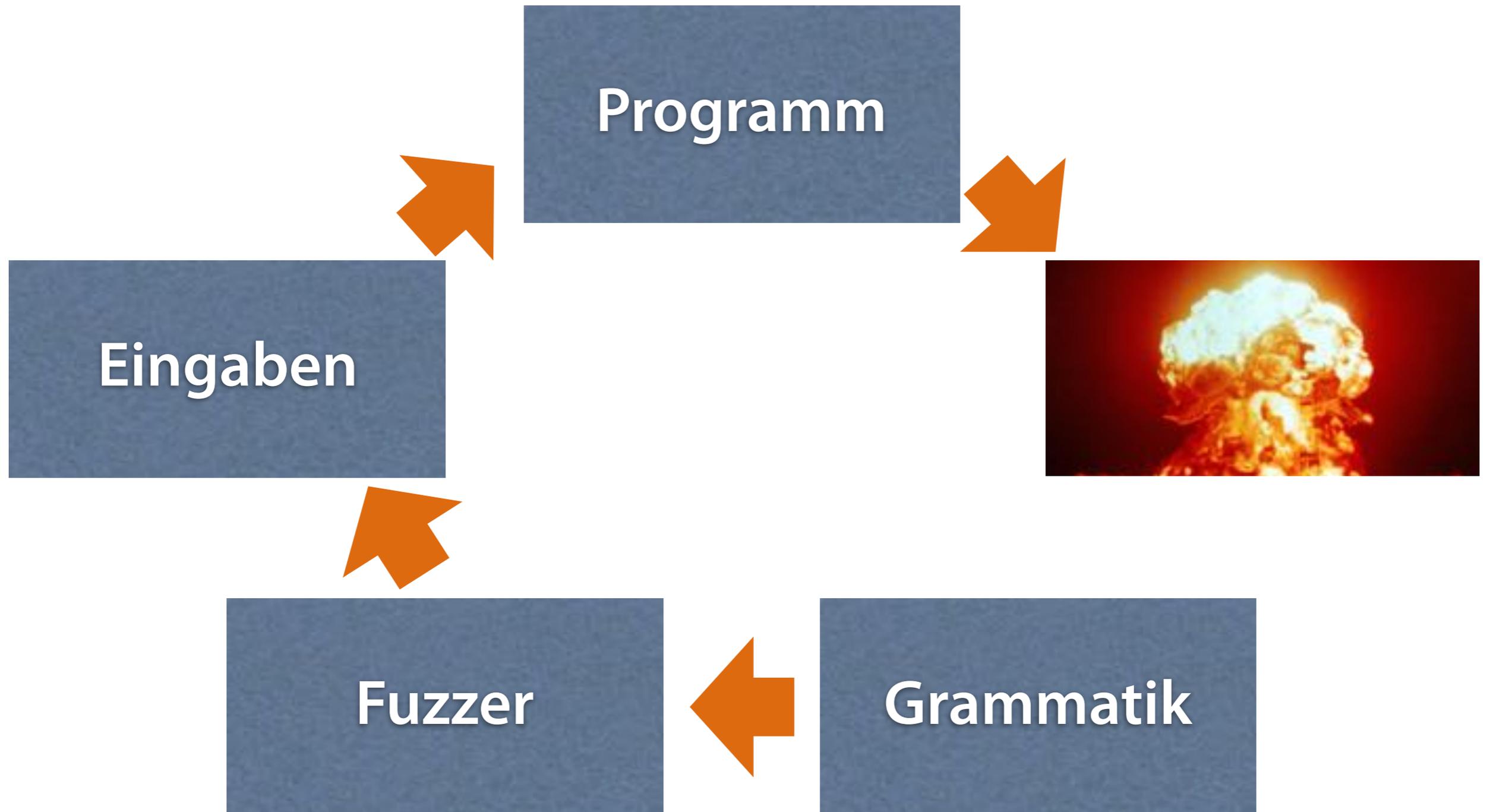
# defects

# JavaScript-Fuzzing



**18 Chromium Security Rewards  
12 Mozilla Security Bug Bounty Awards  
in 9 Monaten**

# Deep Fuzzing



# Quelle der Grammatiken



# Grammatiken lernen

<http://user:pass@www.google.com:80/path>



Programm

# Grammatiken lernen

`http://user:pass@www.google.com:80/path`

`http`

– Protokoll

# Grammatiken lernen

`http://user:pass@www.google.com:80/path`

`http`

– Protokoll

`www.google.com`

– Rechnername

# Grammatiken lernen

`http://user:pass@www.google.com:80/path`

`http`

– Protokoll

`www.google.com`

– Rechnername

`80`

– Port

# Grammatiken lernen

`http://user:pass@www.google.com:80/path`

`http`

– Protokoll

`www.google.com`

– Rechnername

`80`

– Port

`user pass`

– Login

# Grammatiken lernen

`http://user:pass@www.google.com:80/path`

`http`

– Protokoll

`www.google.com`

– Rechnername

`80`

– Port

`user pass`

– Login

`path`

– Seite

# Grammatiken lernen

`http://user:pass@www.google.com:80/path`

`http`

– Protokoll

`www.google.com`

– Rechnername

`80`

– Port

`user pass`

– Login

`path`

– Seite

`:// @ : /`

– Trenner

# Grammatiken lernen

`http://user:pass@www.google.com:80/path`

`http`

– Protokoll

`www.google.com`

– Rechnername

`80`

– Port

`user pass`

– Login

`path`

– Seite

`:// @ : /`

– Trenner

verarbeitet in  
*verschiedenen*  
*Funktionen*

gespeichert in  
*verschiedenen*  
*Variablen*

**http://user:password@www.google.com:80/command?foo=bar&lorem=ipsum#fragment**

java.net.URL.set(Protokoll, host, Port, authority, userinfo, path, query, ref)

| .....  
param: Protokoll

| .....  
param: host

| .....  
param: Port

| .....  
param: authority

| .....  
param: userinfo

| .....  
param: path

| .....  
param: query

| .....  
param: ref

`http://user:password@www.google.com:80/command?foo=bar&lorem=ipsum#fragment`

`java.net.URL.set(Protokoll, host, Port, authority, userinfo, path, query, ref)`

| .....  
param: Protokoll

| `http` .....

param: host

| .....  
param: Port

| .....  
param: authority

| .....  
param: userinfo

| .....  
param: path

| .....  
param: query

| .....  
param: ref

| .....

`http://user:password@www.google.com:80/command?foo=bar&lorem=ipsum#fragment`

`java.net.URL.set(Protokoll, host, Port, authority, userinfo, path, query, ref)`

| .....  
param: Protokoll

| `http` .....

param: host

| ..... `www.google.com` .....

param: Port

| .....  
param: authority

| .....  
param: userinfo

| .....  
param: path

| .....  
param: query

| .....  
param: ref

`http://user:password@www.google.com:80/command?foo=bar&lorem=ipsum#fragment`

`java.net.URL.set(Protokoll, host, Port, authority, userinfo, path, query, ref)`

param: Protokoll

`http`

param: host

`www.google.com`

param: Port

param: authority

param: userinfo

`user:password`

param: path

param: query

param: ref

`http://user:password@www.google.com:80/command?foo=bar&lorem=ipsum#fragment`

`java.net.URL.set(Protokoll, host, Port, authority, userinfo, path, query, ref)`

param: Protokoll

`http`

param: host

`www.google.com`

param: Port

`80`

param: authority

param: userinfo

`user:password`

param: path

param: query

param: ref

`http://user:password@www.google.com:80/command?foo=bar&lorem=ipsum#fragment`

`java.net.URL.set(Protokoll, host, Port, authority, userinfo, path, query, ref)`

param: Protokoll

`http`

param: host

`www.google.com`

param: Port

`80`

param: authority

param: userinfo

`user:password`

param: path

`/command`

param: query

param: ref

`http://user:password@www.google.com:80/command?foo=bar&lorem=ipsum#fragment`

```
java.net.URL.set(Protokoll, host, Port, authority, userinfo, path, query, ref)
| .....
param: Protokoll
| http .....
param: host
| ..... www.google.com .....
param: Port
| ..... 80 .....
param: authority
| .....
param: userinfo
| ..... user:password .....
param: path
| ..... /command .....
param: query
| ..... foo=bar&lorem=ipsum .....
param: ref
| .....
```

http://user:password@www.google.com:80/command?foo=bar&lorem=ipsum#fragment

```
java.net.URL.set(Protokoll, host, Port, authority, userinfo, path, query, ref)
| .....
param: Protokoll
| http .....
param: host
| ..... www.google.com .....
param: Port
| ..... 80 .....
param: authority
| .....
param: userinfo
| ..... user:password .....
param: path
| ..... /command .....
param: query
| ..... foo=bar&lorem=ipsum .....
param: ref
| ..... fragment
```

http://user:password@www.google.com:80/command?foo=bar&lorem=ipsum#fragment

```
java.net.URL.set(Protokoll, host, Port, authority, userinfo, path, query, ref)
| .....
param: Protokoll
| http .....
param: host
| ..... www.google.com .....
param: Port
| ..... 80 .....
param: authority
| ..... user:password@www.google.com:80 .....
param: userinfo
| ..... user:password .....
param: path
| ..... /command .....
param: query
| ..... foo=bar&lorem=ipsum .....
param: ref
| ..... fragment
```

http://user:password@www.google.com:80/command?foo=bar&lorem=ipsum#fragment

```
java.net.URL.set(Protokoll, host, Port, authority, userinfo, path, query, ref)
| http...user:password@www.google.com:80/command...foo=bar&lorem=ipsum...fragment
param: Protokoll
| http.....
param: host
| .....www.google.com.....
param: Port
| .....80.....
param: authority
| .....user:password@www.google.com:80.....
param: userinfo
| .....user:password.....
param: path
| ...../command.....
param: query
| .....foo=bar&lorem=ipsum.....
param: ref
| .....fragment.....
```



URL ::= PROTOCOL '://' AUTHORITY  
 AUTHORITY ::= USERINFO '@' HOST

```
java.net.URL.set(Protokoll, host, Port, authority, userinfo, path, query, ref)
| http...user:password@www.google.com:80/command·foo=bar&lorem=ipsum·fragment
param: Protokoll
| http.....
param: host
| .....www.google.com.....
param: Port
| .....80.....
param: authority
| .....user:password@www.google.com:80.....
param: userinfo
| .....user:password.....
param: path
| ...../command.....
param: query
| .....foo=bar&lorem=ipsum.....
param: ref
| .....fragment.....
```



```
URL ::= PROTOCOL '://' AUTHORITY PATH '?' QUERY '#' REF
AUTHORITY ::= USERINFO '@' HOST ':' PORT
PROTOCOL ::= 'http'
USERINFO ::= 'user:password'
HOST ::= 'www.google.com'
PORT ::= '80'
PATH ::= '/command'
QUERY ::= 'foo=bar&lorem=ipsum'
REF ::= 'fragment'
```

# URLs

`http://user:password@www.google.com:80/command?foo=bar&lorem=ipsum#fragment`  
`http://www.guardian.co.uk/sports/worldcup#results`  
`ftp://bob:12345@ftp.example.com/oss/debian7.iso`



```
URL ::= PROTOCOL '://' AUTHORITY PATH ['?' QUERY] ['#' REF]
AUTHORITY ::= [USERINFO '@'] HOST [':' PORT]
PROTOCOL ::= 'http' | 'ftp'
USERINFO ::= /[a-z]+:[a-z]+/
HOST ::= /[a-z.]+/
PORT ::= '80'
PATH ::= /\[/[a-z0-9.\//]*\//
QUERY ::= 'foo=bar&lorem=ipsum'
REF ::= /[a-z]+/
```

# INI-Dateien

```
[Application]
Version = 0.5
WorkingDir = /tmp/mydir/
[User]
User = Bob
Password = 12345
```



```
INI ::= LINE+
LINE ::= SECTION_LINE '\r'
      | OPTION_LINE  ['\r']
SECTION_LINE ::= '[' KEY ']'
OPTION_LINE  ::= KEY ' = ' VALUE
KEY ::= /[a-zA-Z]*/
VALUE ::= /[a-zA-Z0-9\ ]/
```

# JSON

<JSON Dateien>



# Dateien

```
JSON ::= VALUE
VALUE ::= JSONOBJECT | ARRAY | STRINGVALUE |
        TRUE | FALSE | NULL | NUMBER
TRUE ::= 'true'
FALSE ::= 'false'
NULL ::= 'null'
NUMBER ::= ['-'] / [0-9]+/
STRINGVALUE ::= '"' INTERNALSTRING '"'
INTERNALSTRING ::= /[a-zA-Z0-9 ]+/
ARRAY ::= '['
        [VALUE [',' VALUE]+]
        ']'
JSONOBJECT ::= '{'
        [STRINGVALUE ':' VALUE
        [',' STRINGVALUE ':' VALUE]
        +]
        '}'
```

# URLs testen

```
URL ::= PROTOCOL '://' AUTHORITY PATH ['?' QUERY] ['#' REF]
AUTHORITY ::= [USERINFO '@'] HOST [':' PORT]
PROTOCOL ::= 'http' | 'ftp'
USERINFO ::= /[a-z]+:[a-z]+/
HOST ::= /[a-z.]+/
PORT ::= '80'
PATH ::= /\[/[a-z0-9.\//]*\//
QUERY ::= 'foo=bar&lorem=ipsum'
REF ::= /[a-z]+/
```



```
http://6F35:Pkt5v@2.5/,,
http://.g:8
http://C.Ta.2./p.,//1.#14cq5
http://.37...g:776/,,
http://.:07//,.8B,#eUN027
http://87.:2117//?&=&&38#207
http://S1t26c:7223i@.1...:16207
ftp://wb428:lr@00.8y.#5W7V9U2
ftp://012304:xt9Ut@k:285?250===K
```

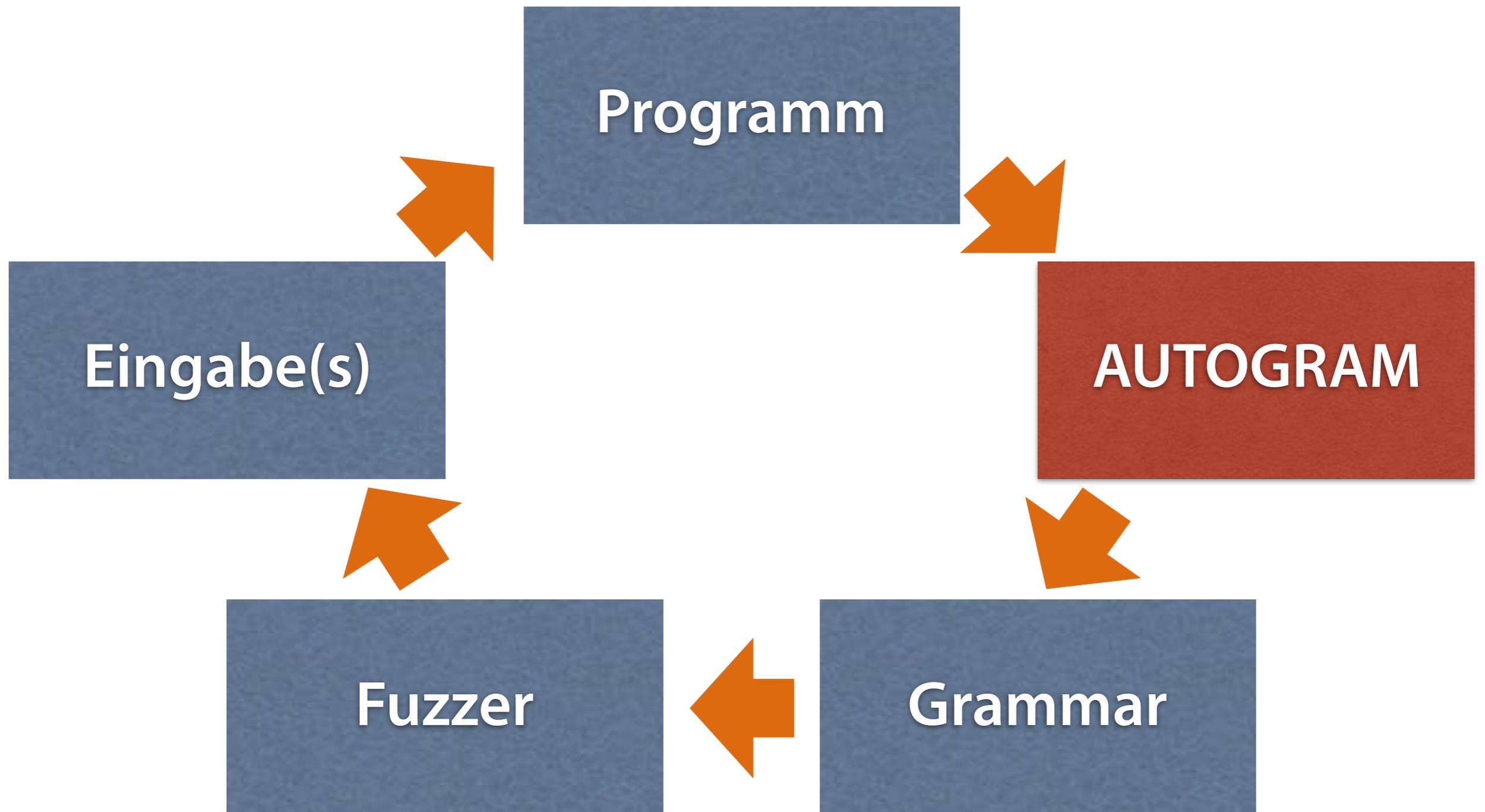
http://mE:26Ciu@.8...:1528/8,,2,,?===&r&  
ftp://rW:L@0H....:8111/7,,g/,  
http://D...C  
http://2.6j0:032277  
http://x1f0...:332334?&==2==&  
http://3u8Wabn:tN@m:3592#36  
ftp://2.8...:9161208/..?=&=9#5F  
ftp://.n:7945457//?9  
http://Jy:98/9,3?===&#1q  
http://G42:7Nz596e@6.4b//F/,?&I=0  
ftp://.697..?===SU=  
http://3d00:ud@.1dF9/2q//5  
ftp://.d5...8:646#D  
ftp://62ql1:40P63@4.:321727?=  
http://.//,,/  
ftp://8zN3xl:3499l8@t036./,3?=&=40  
http://B7j85D3:NvPd7M@.8.p.:5/,,#e7JS  
http://t4...:124///6,G.?=&&=#3F2Qx  
http://YP6:zKG@.:946775?=#Zb7  
http://./,31,,F.#693  
ftp://7V:c4748C2@.//...?&&&&2R  
http://...:40123?=r=&7I  
ftp://.74:4773362/.A#Et  
ftp://67:3g5YNi@.5M.2...:06716?&=#3W758V6  
ftp://i:cqj97@..2..3:362287?&=&7f5#4  
http://1:l@N..6..i:667//,,6,  
http://70o0:518@3:4791089#962  
ftp://zA35Qsu:56@..5...:997/,.  
ftp://8.../5?&n#7i1C7G3  
ftp://2:fm0@J.:6208/,Z/H#3GZ747b  
http://2:7p54n14@8r09.1  
ftp://XK3438:w169KkU@..5R.8?=6g



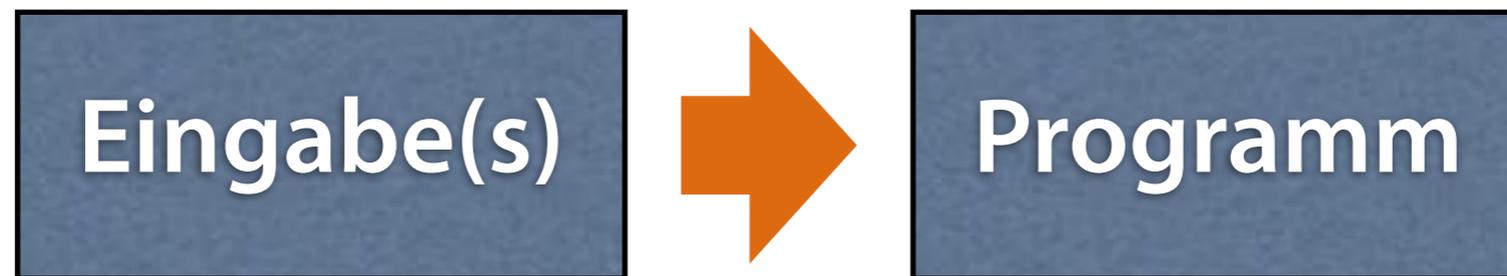
# Evaluation

<b>Subject</b>	<b>Accuracy</b>	<b>Completeness</b>
<code>java.lang.URL</code>	82.3%	100.0%
Apache Commons CSV	100.0%	100.0%
INI4J	64.6%	100.0%
Minimal JSON	100.0%	100.0%

# Grammatiken lernen



# Dynamische Prüfungen

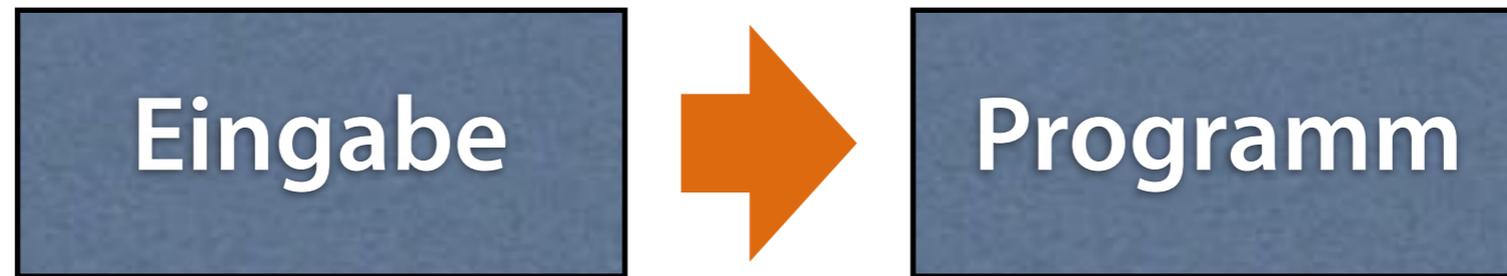


*xyzyzy*



- prüft auf Ziffer
- prüft auf "true"/"false"
- prüft auf ""
- prüft auf '['
- prüft auf '{'

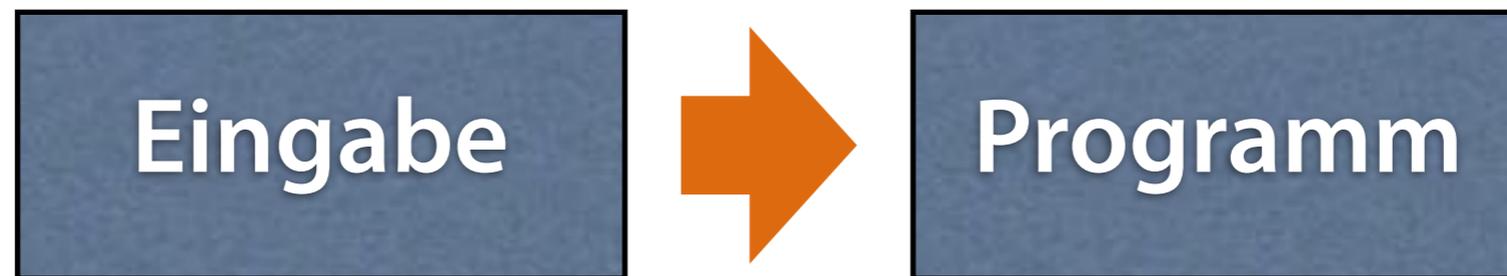
# Dynamische Prüfungen



0



# Dynamische Prüfungen

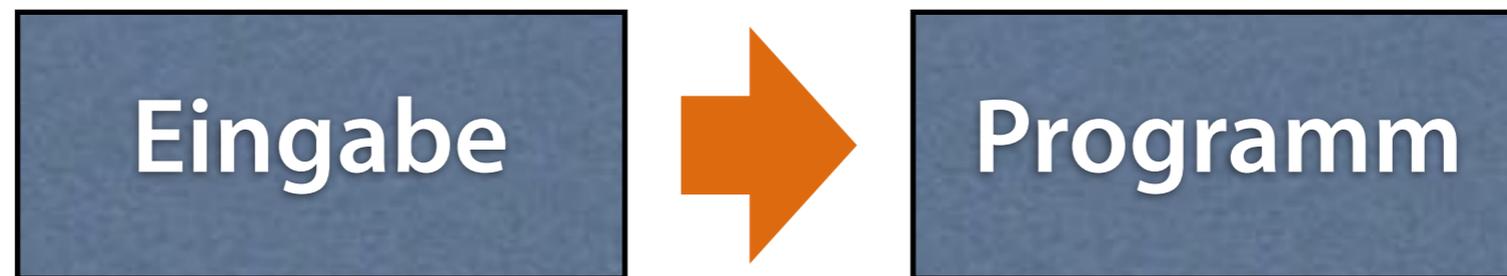


0



- prüft auf Ziffer
- prüft auf "true"/"false"
- prüft auf ""
- prüft auf '['
- prüft auf '{'

# Dynamische Prüfungen

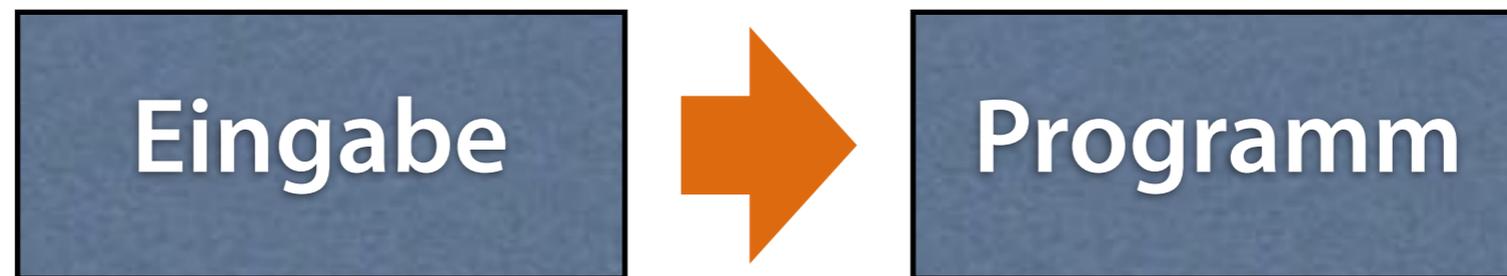


0



- prüft auf Ziffer
- prüft auf "true"/"false"
- prüft auf ""
- prüft auf '['
- prüft auf '{'

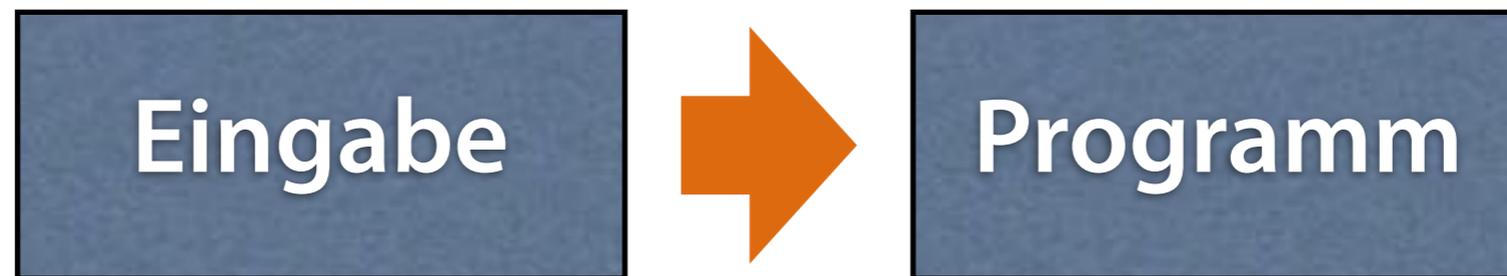
# Dynamische Prüfungen



true



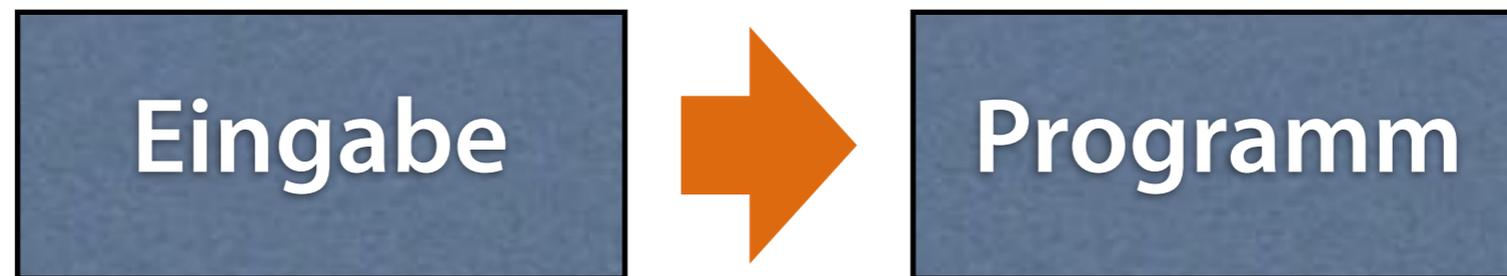
# Dynamische Prüfungen



false



# Dynamische Prüfungen

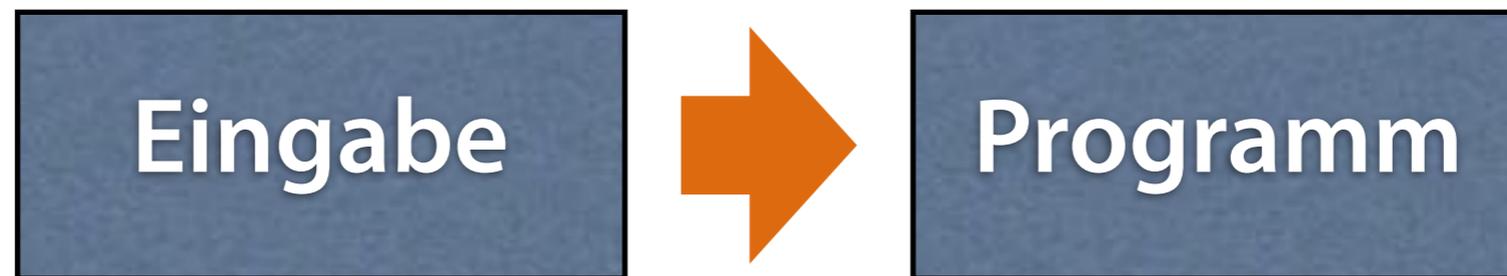


false



- prüft auf Ziffer
- prüft auf "true"/"false"
- prüft auf ""
- prüft auf '['
- prüft auf '{'

# Dynamische Prüfungen

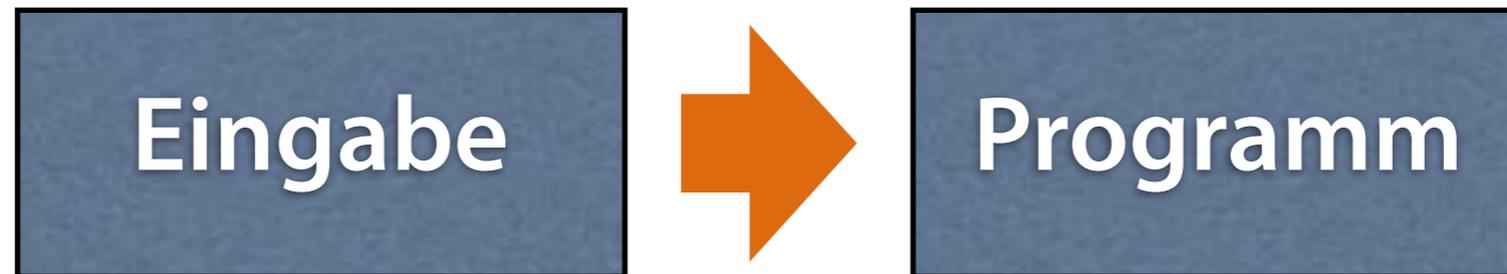


false



- prüft auf Ziffer
- prüft auf "true"/"false"
- prüft auf ""
- prüft auf '['
- prüft auf '{'

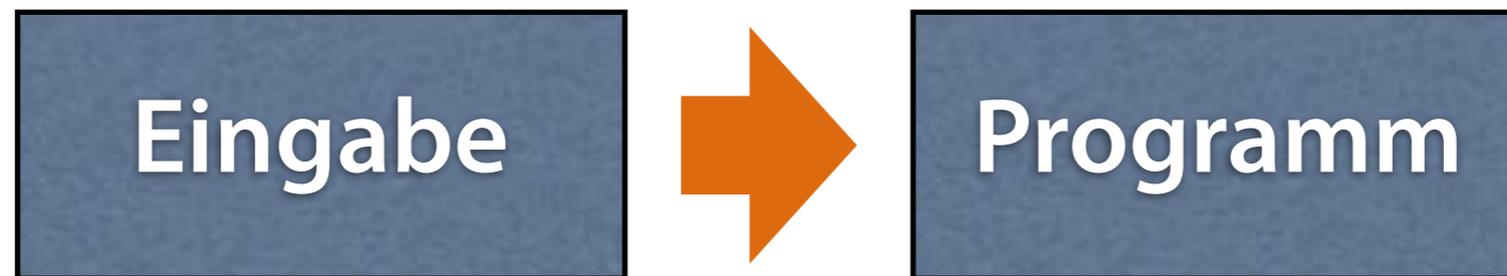
# Dynamische Prüfungen



||

X

# Dynamische Prüfungen

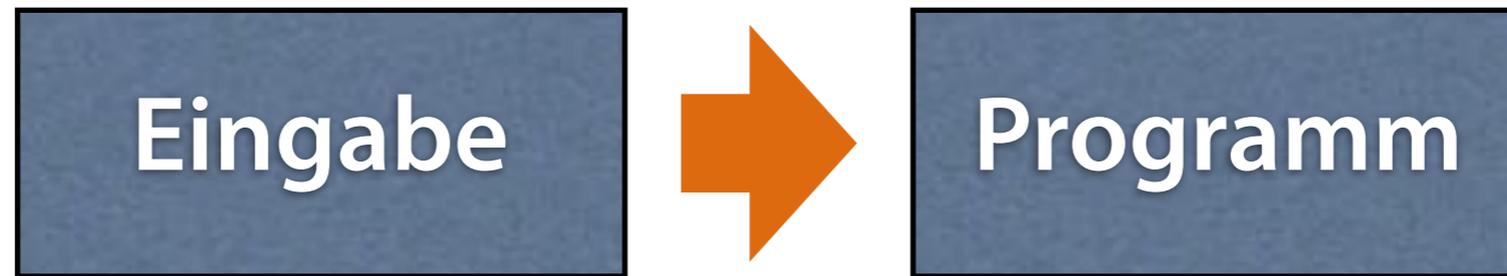


||

X

- prüft auf ""
- prüft auf \'\'
- prüft auf andere Zeichen

# Dynamische Prüfungen



||||



# JSON

# Dateien

<JSON-Dateien>



```
JSON ::= VALUE
VALUE ::= JSONOBJECT | ARRAY | STRINGVALUE |
        TRUE | FALSE | NULL | NUMBER
TRUE ::= 'true'
FALSE ::= 'false'
NULL ::= 'null'
NUMBER ::= ['-'] / [0-9]+/
STRINGVALUE ::= '"' INTERNALSTRING '"'
INTERNALSTRING ::= /[a-zA-Z0-9 ]+/
ARRAY ::= '['
        [VALUE [',' VALUE]+]
        ']'
JSONOBJECT ::= '{'
        [STRINGVALUE ':' VALUE
        [',' STRINGVALUE ':' VALUE]
        +]
        '}'
```

# Fuzz Testing

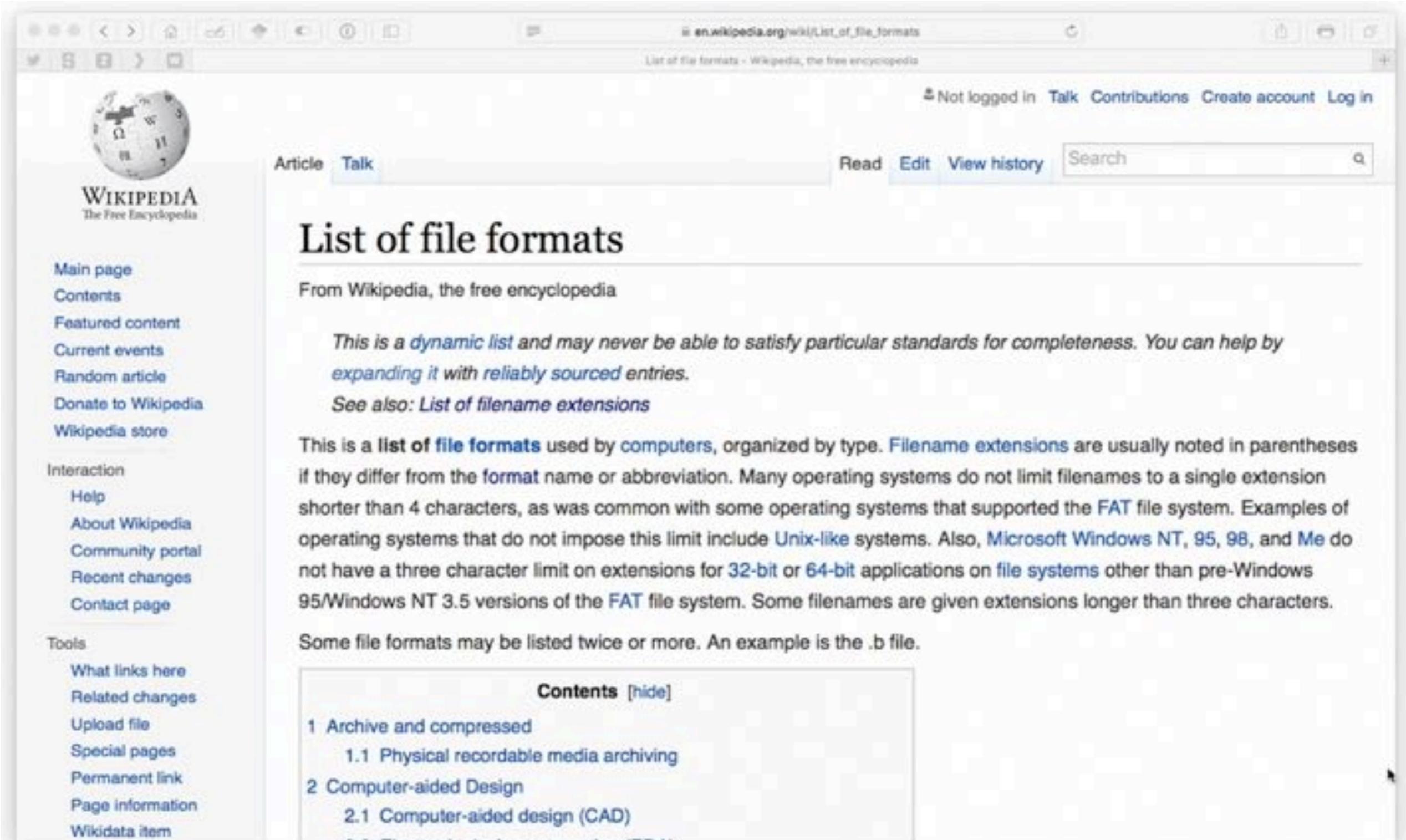
<nichts>



Programm



# Dateiformate



The image shows a screenshot of the Wikipedia article titled "List of file formats". The browser address bar shows the URL "en.wikipedia.org/wiki/List\_of\_file\_formats". The page header includes the Wikipedia logo, the text "WIKIPEDIA The Free Encyclopedia", and navigation links such as "Main page", "Contents", "Featured content", "Current events", "Random article", "Donate to Wikipedia", and "Wikipedia store". The article content begins with the heading "List of file formats" and a sub-heading "From Wikipedia, the free encyclopedia". A note states: "This is a *dynamic list* and may never be able to satisfy particular standards for completeness. You can help by *expanding it with reliably sourced entries*." Below this, it says "See also: *List of filename extensions*". The main text explains that this is a list of file formats used by computers, organized by type, and notes that filename extensions are usually noted in parentheses if they differ from the format name or abbreviation. It mentions that many operating systems do not limit filenames to a single extension shorter than 4 characters, as was common with some operating systems that supported the FAT file system. Examples of operating systems that do not impose this limit include Unix-like systems, and Microsoft Windows NT, 95, 98, and Me. It also notes that some file formats may be listed twice or more, with an example of the .b file. At the bottom, there is a "Contents" section with a [hide] link, listing sections like "1 Archive and compressed" and "2 Computer-aided Design".

en.wikipedia.org/wiki/List\_of\_file\_formats  
List of file formats - Wikipedia, the free encyclopedia

Not logged in Talk Contributions Create account Log in

Article Talk Read Edit View history Search

## List of file formats

From Wikipedia, the free encyclopedia

*This is a **dynamic list** and may never be able to satisfy particular standards for completeness. You can help by **expanding it with reliably sourced entries**.*

*See also: **List of filename extensions***

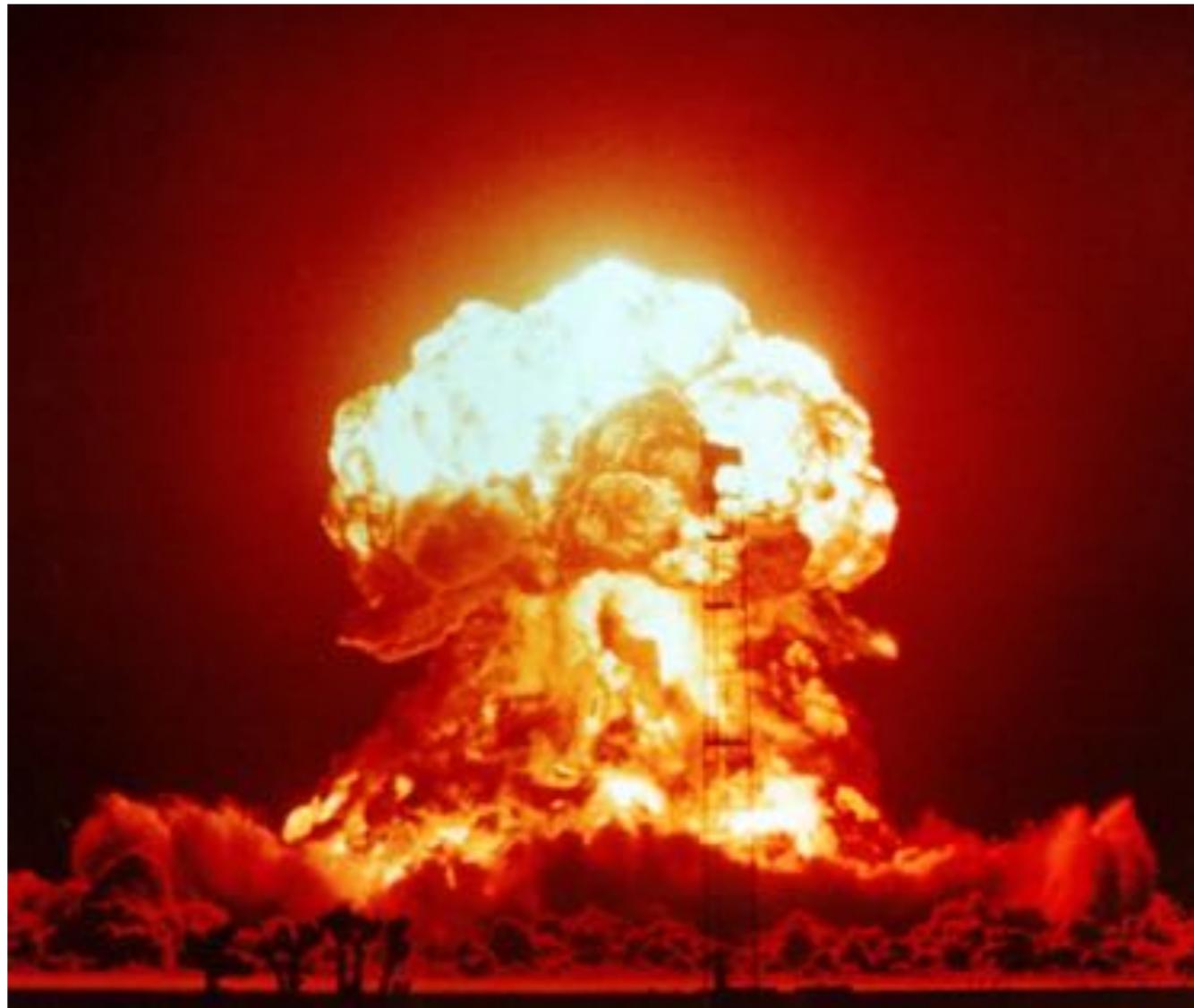
This is a **list of file formats** used by computers, organized by type. **Filename extensions** are usually noted in parentheses if they differ from the **format** name or abbreviation. Many operating systems do not limit filenames to a single extension shorter than 4 characters, as was common with some operating systems that supported the **FAT** file system. Examples of operating systems that do not impose this limit include **Unix-like** systems. Also, **Microsoft Windows NT, 95, 98, and Me** do not have a three character limit on extensions for **32-bit** or **64-bit** applications on **file systems** other than pre-Windows 95/Windows NT 3.5 versions of the **FAT** file system. Some filenames are given extensions longer than three characters.

Some file formats may be listed twice or more. An example is the **.b** file.

### Contents [hide]

- 1 Archive and compressed
  - 1.1 Physical recordable media archiving
- 2 Computer-aided Design
  - 2.1 Computer-aided design (CAD)
  - 2.2 Electrical design automation (EDA)

# Was nun?

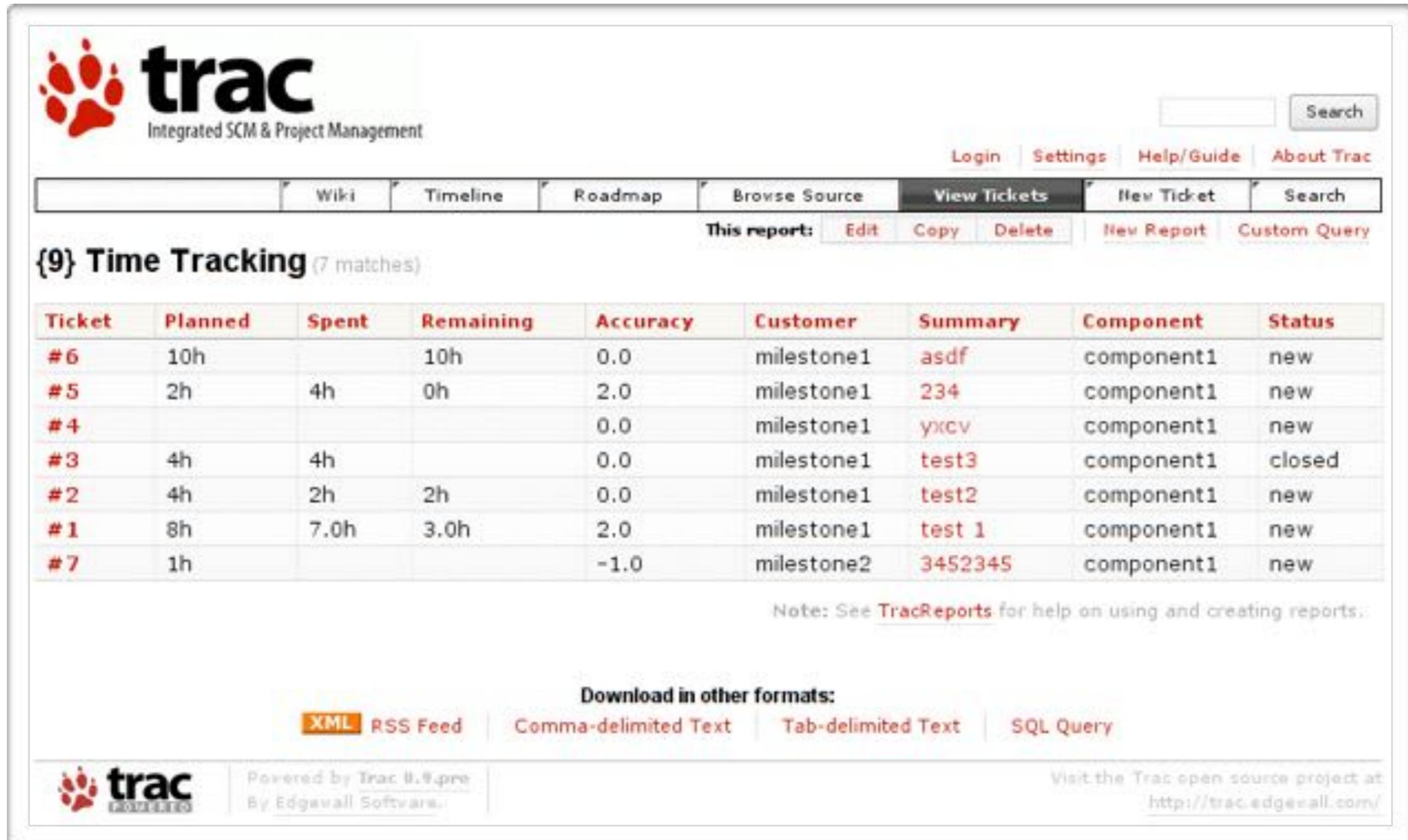




# Systematische Fehlersuche

<b>T</b> rack the problem	<i>Problem verfolgen</i>
<b>R</b> eproduce	<i>Reproduzieren</i>
<b>A</b> utomate	<i>Automatisieren</i>
<b>F</b> ind Origins	<i>Ursprünge finden</i>
<b>F</b> ocus	<i>Fokussieren</i>
<b>I</b> solate	<i>Isolieren</i>
<b>C</b> orrect	<i>Korrigieren</i>

# Problem verfolgen



The screenshot displays the Trac web interface. At the top left is the Trac logo (a red paw print) and the text 'trac Integrated SCM & Project Management'. To the right is a search box with a 'Search' button. Below the logo are navigation links: 'Login', 'Settings', 'Help/Guide', and 'About Trac'. A secondary navigation bar contains 'Wiki', 'Timeline', 'Roadmap', 'Browse Source', 'View Tickets' (which is highlighted), 'New Ticket', and 'Search'. Below this bar are report management options: 'This report: Edit Copy Delete New Report Custom Query'. The main content area shows a report titled '{9} Time Tracking (7 matches)'. Below the title is a table with the following data:

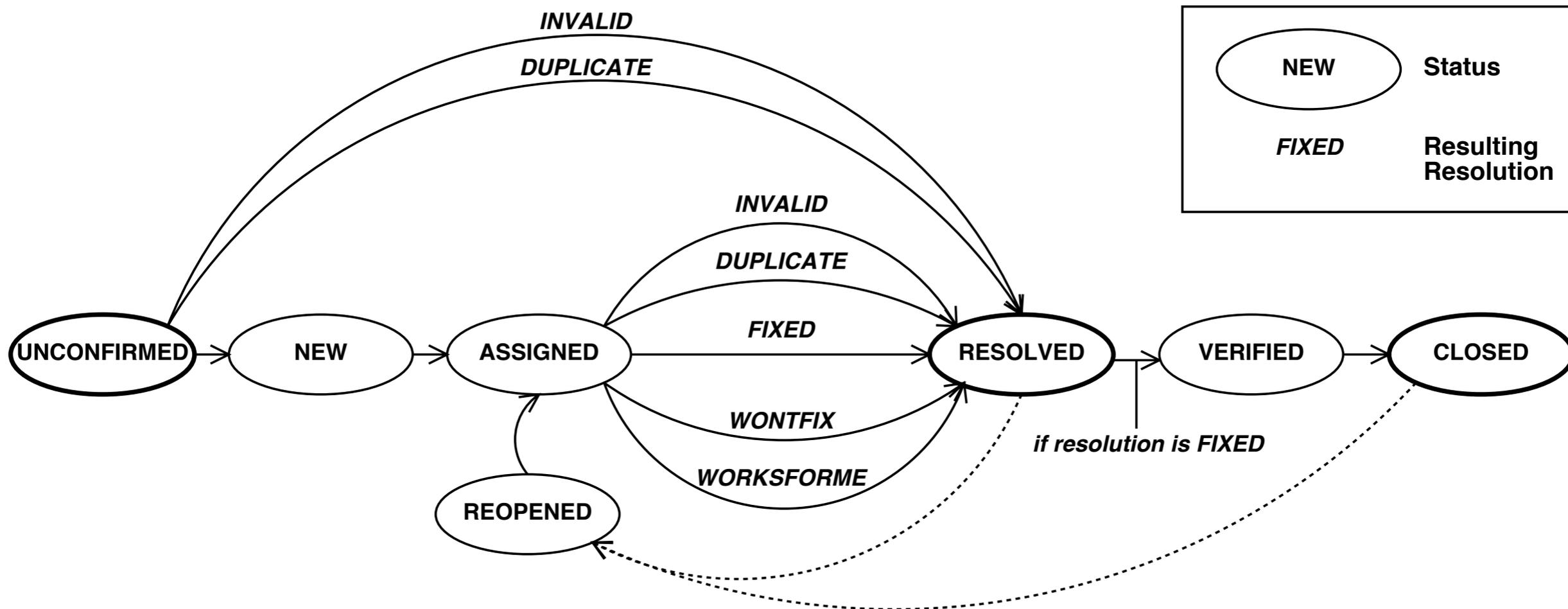
Ticket	Planned	Spent	Remaining	Accuracy	Customer	Summary	Component	Status
#6	10h		10h	0.0	milestone1	asdf	component1	new
#5	2h	4h	0h	2.0	milestone1	234	component1	new
#4				0.0	milestone1	yxcv	component1	new
#3	4h	4h		0.0	milestone1	test3	component1	closed
#2	4h	2h	2h	0.0	milestone1	test2	component1	new
#1	8h	7.0h	3.0h	2.0	milestone1	test 1	component1	new
#7	1h			-1.0	milestone2	3452345	component1	new

Below the table is a note: 'Note: See [TracReports](#) for help on using and creating reports.' At the bottom of the report area, there is a section titled 'Download in other formats:' with links for 'XML RSS Feed', 'Comma-delimited Text', 'Tab-delimited Text', and 'SQL Query'. The footer of the page contains the Trac logo, the text 'Powered by Trac 0.9.pre By Edgewall Software.', and a link to the Trac open source project at <http://trac.edgewall.com/>.

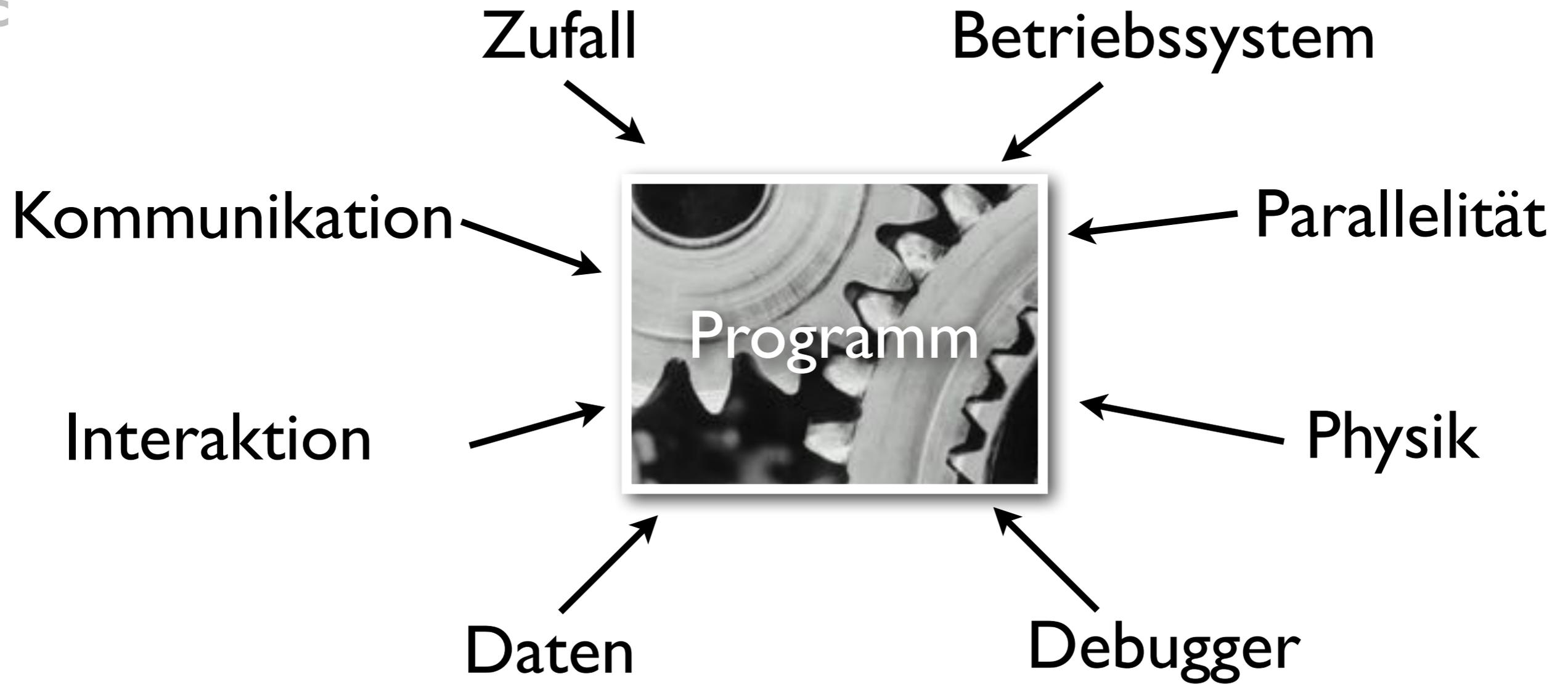
# Problem verfolgen

- Jedes Problem wird in die Fehler-Datenbank eingetragen
- Die Priorität bestimmt, welches Problem als nächstes bearbeitet wird
- Sind alle Probleme behoben, ist das Produkt fertig

# Lebenszyklus eines Problems



# Reproduzieren



# Automatisieren

```
// Test for host
public void testHost() {
    int noPort = -1;
    assertEquals(askigor_url.getHost(), "www.askigor.org");
    assertEquals(askigor_url.getPort(), noPort);
}
```

```
// Test for path
public void testPath() {
    assertEquals(askigor_url.getPath(), "/status.php");
}
```

```
// Test for query part
public void testQuery() {
    assertEquals(askigor_url.getQuery(), "id=sample");
}
```

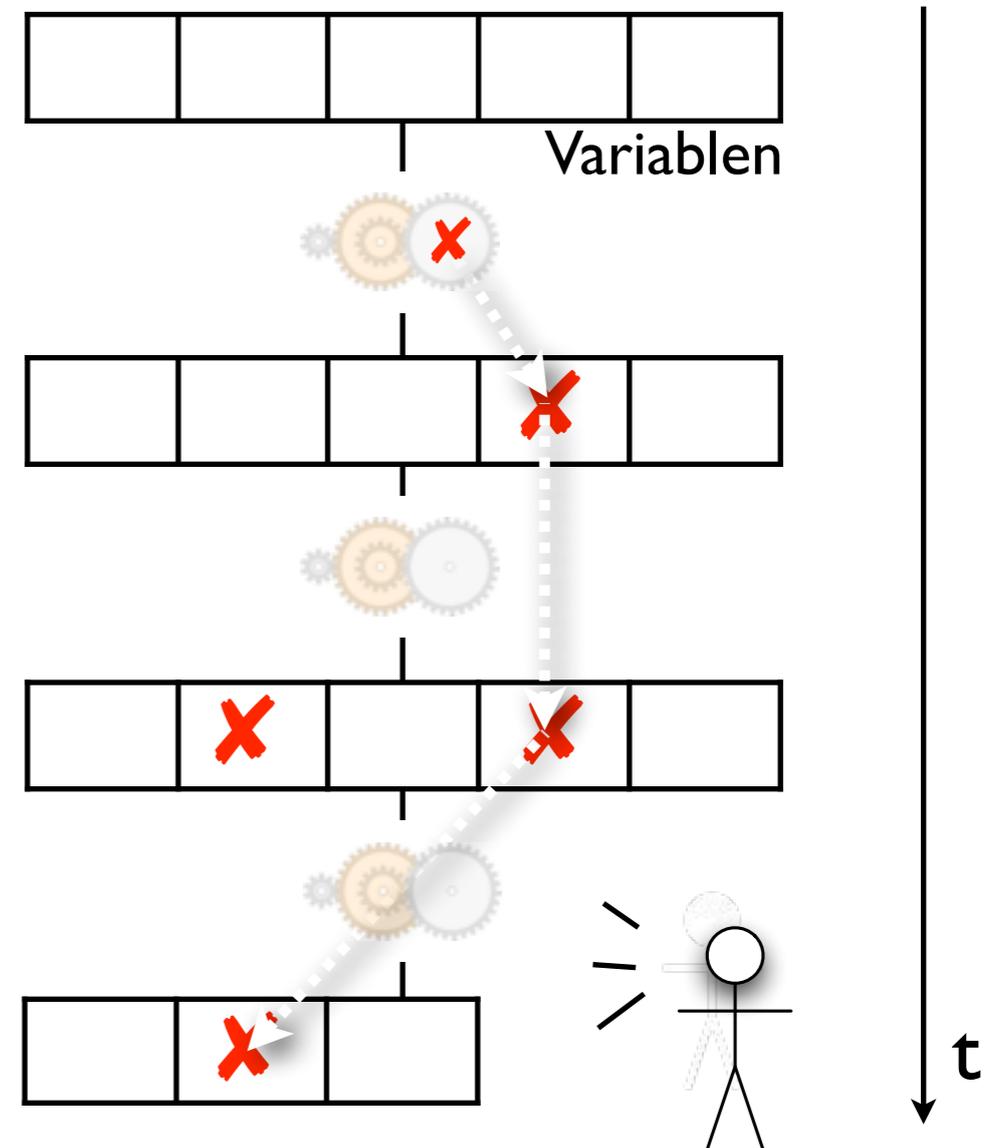
# Automatisieren

- Jedes Problem sollte automatisch reproduzierbar sein
- Dies geschieht über geeignete JUnit-Testfälle
- Nach jeder Änderung werden die Testfälle ausgeführt

# Ursprung finden

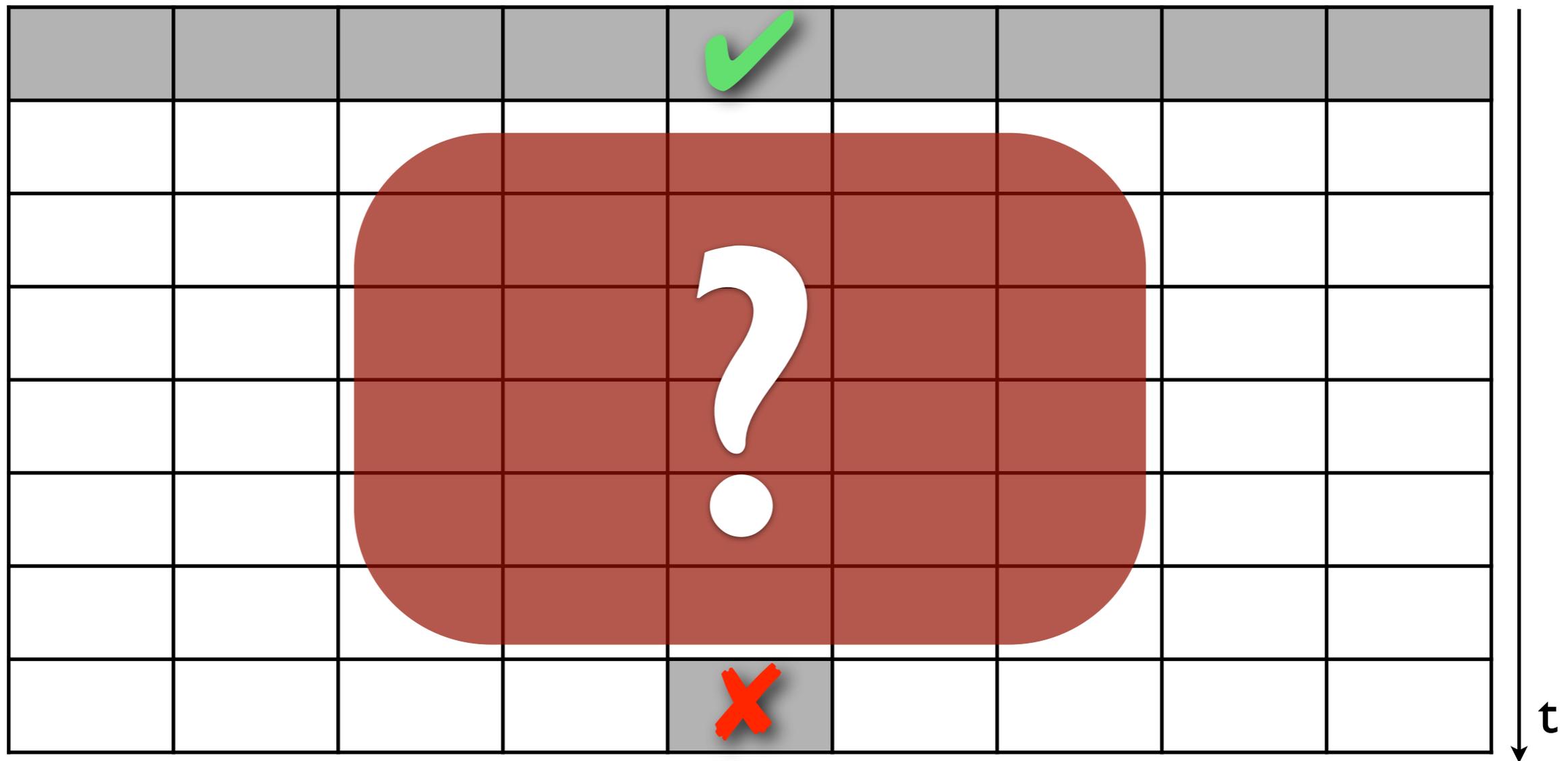
1. Der Programmierer erzeugt einen *Defekt* – einen Fehler im Code
2. Der ausgeführte Defekt erzeugt eine *Infektion* – einen Fehler im Zustand
3. Die Infektion breitet sich aus...
4. ...und wird als *Fehlverhalten* sichtbar.

Diese Infektionskette müssen wir *brechen*.



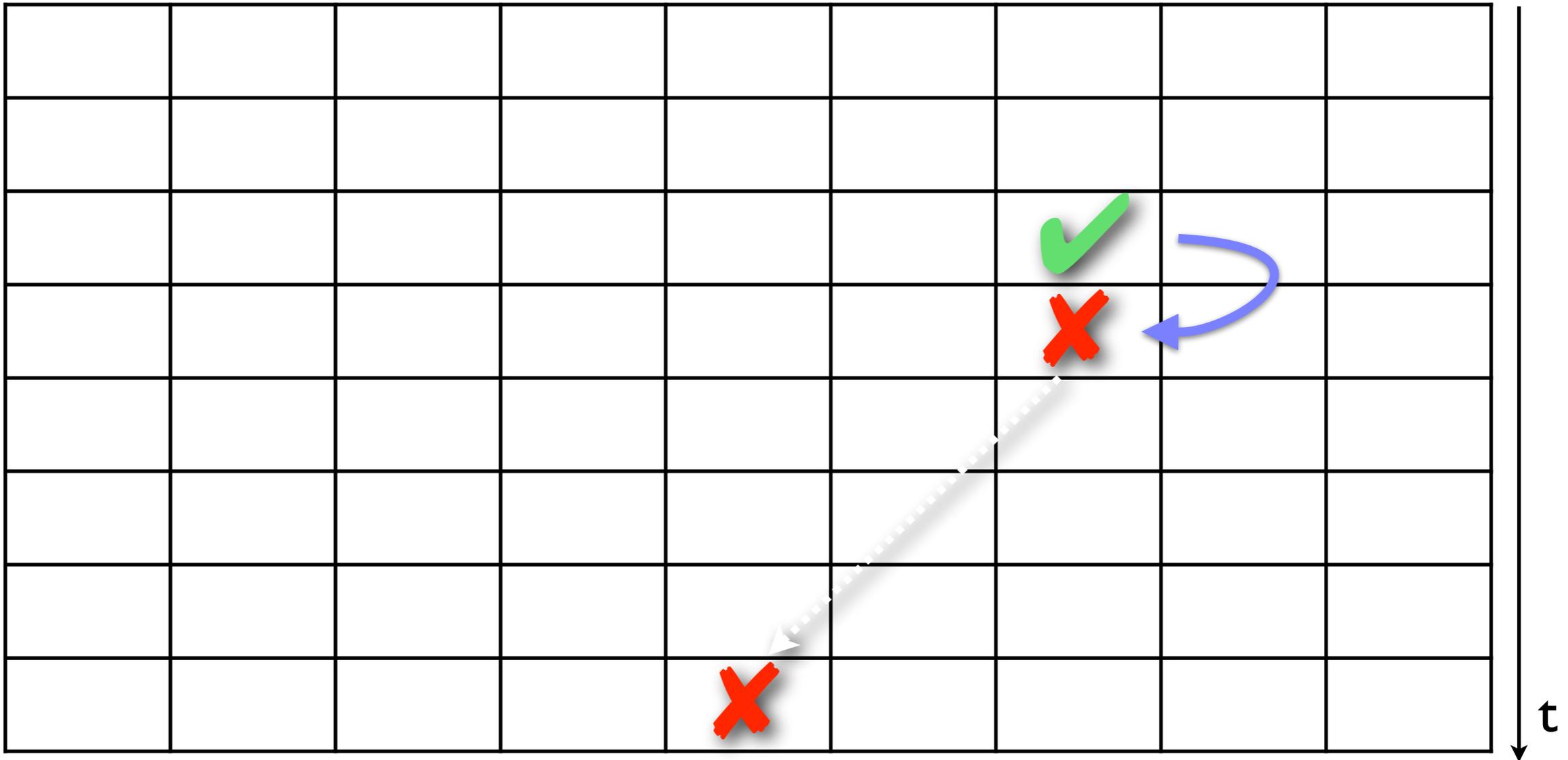
# Ursprung finden

Variablen

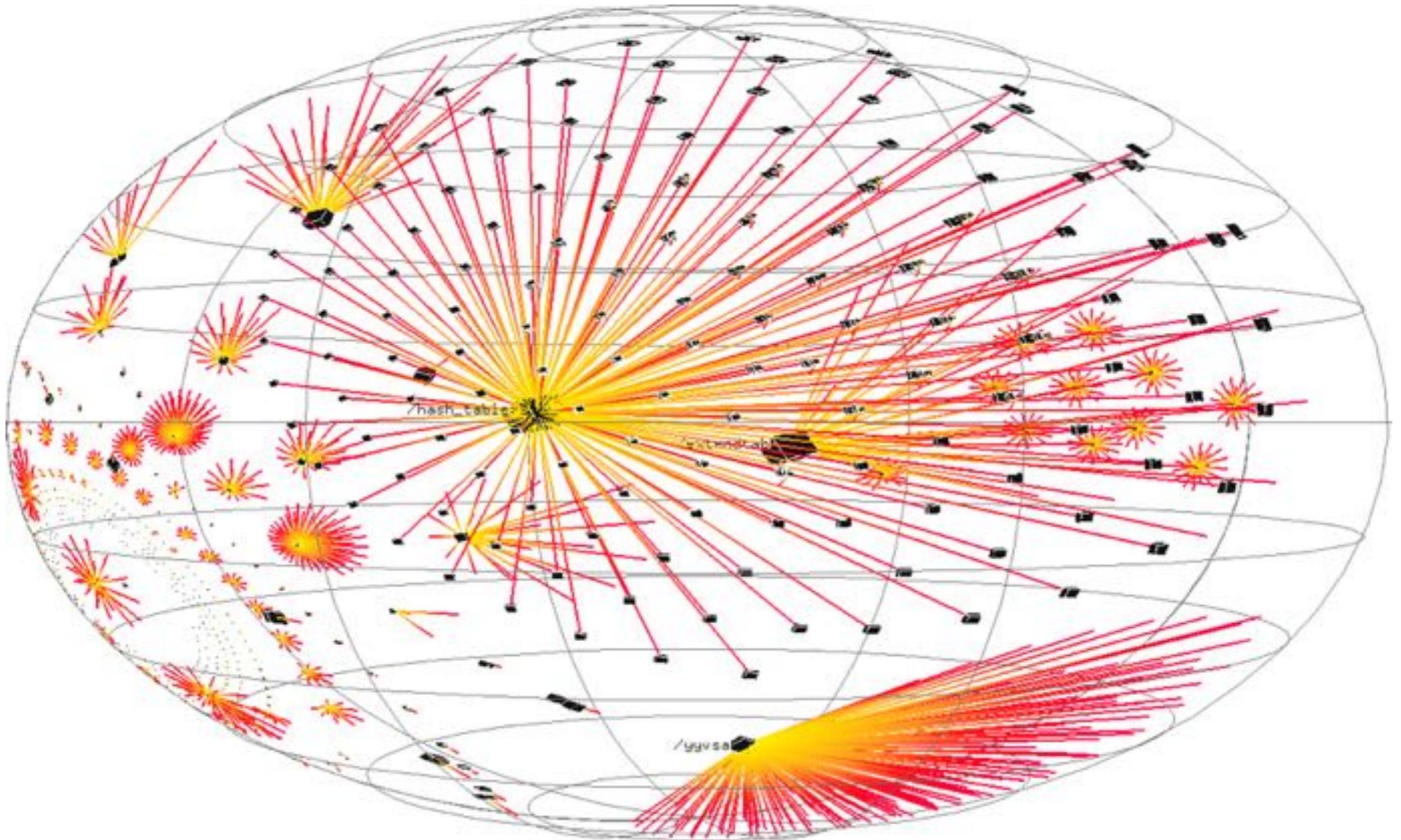


# Der Defekt

Variablen

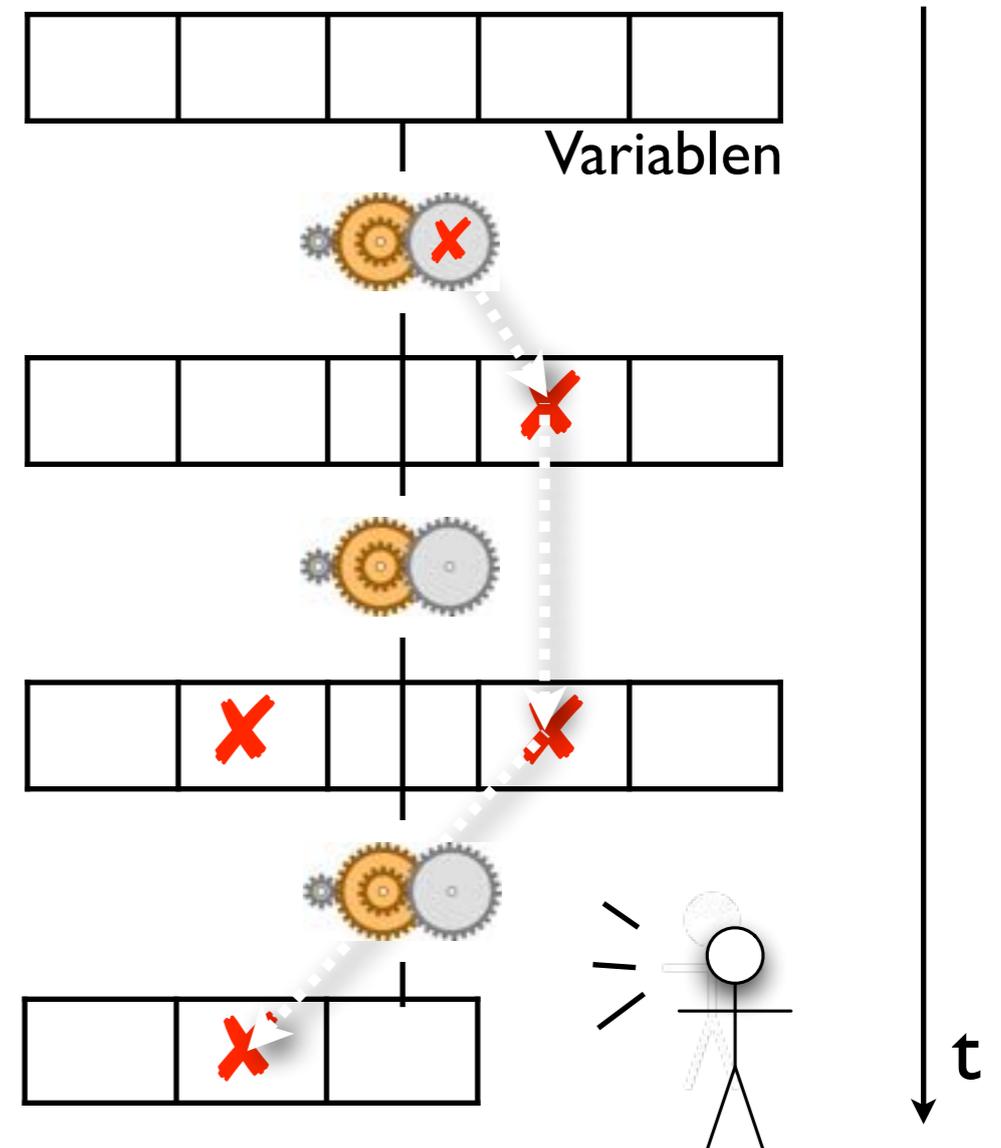


# Ein Programmzustand



# Ursprung finden

1. Wir beginnen mit einer *bekanntem Infektion* (etwa am *Ende der Ausführung*)
2. Wir suchen die Infektion im *vorherigen Zustand*



DDD: /public/source/programming/ddd-3.2/ddd/cxxtest.C

File Edit View Program Commands Status Source Data Help

0: list->self

```

list->next      = new List(a_global + start++);
list->next->next = new List(a_global + start++);
list->next->next->next = list;

```

**STOP** (void) list; // Display this

(List \*) 0x804df80

```

delete list;
delete list->next;
delete list;
}
// Test
void lis
{
    list
}
//-----
void ref
{
    date
    dele
    date
}

```

DDD Tip of the Day #5

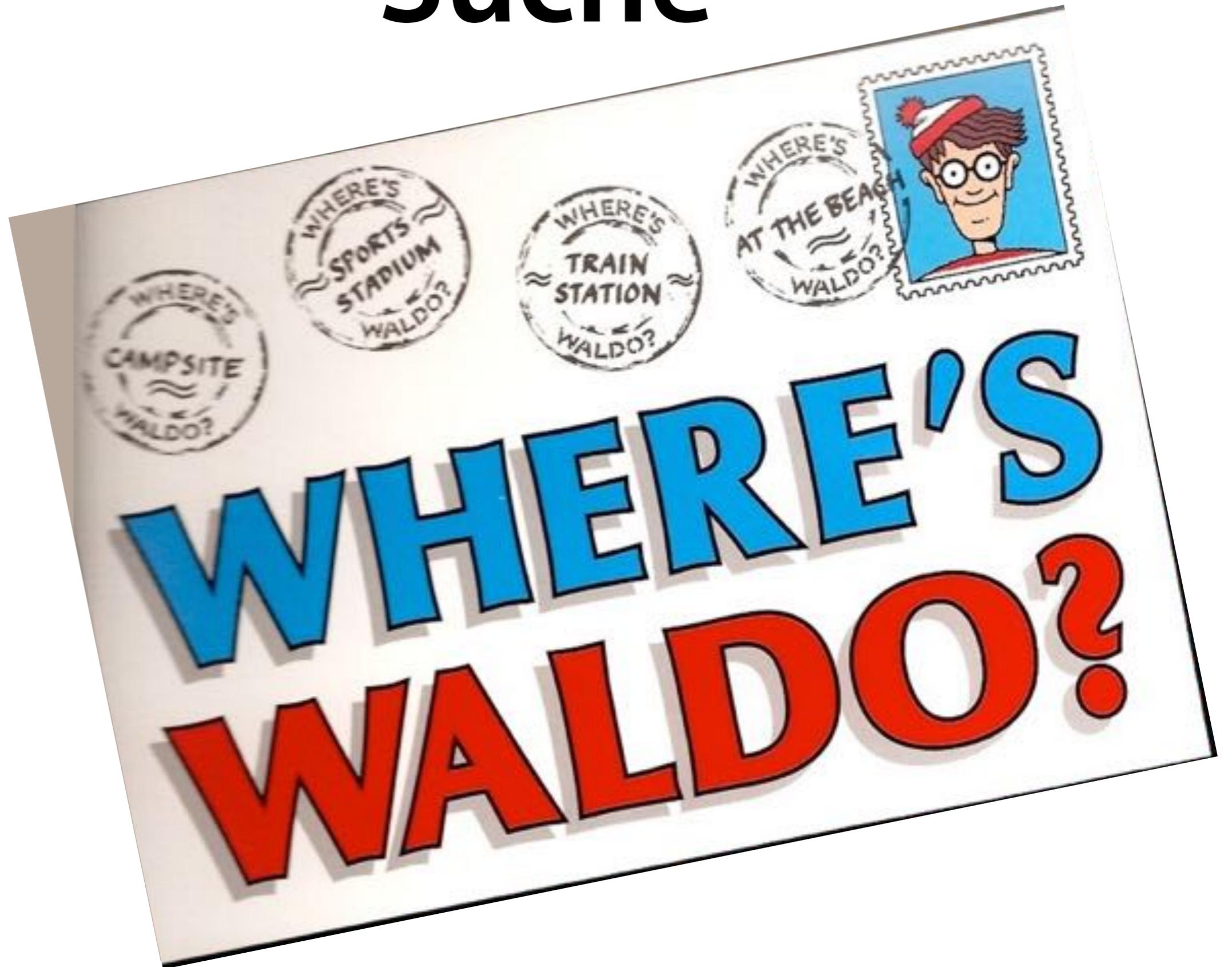
If you made a mistake, try **Edit→Undo**. This will undo the most recent debugger command and redisplay the previous program state.

Close Prev Tip Next Tip

(gdb) graph display \*(list->next->next->self) dependent on 4  
(gdb) [

Δ list = (List \*) 0x804df80

# Suche





# Fokussieren

Bei der Suche nach Infektionen konzentrieren uns auf Stellen im Zustand, die

- *wahrscheinlich falsch* sind (z.B. weil hier früher Fehler aufgetreten sind)
- *explizit falsch* sind (z.B. weil sie eine *Zusicherung* verletzen)

Zusicherungen sind das effektivste Mittel, Infektionen zu finden.

# Infektionen finden

```
class Time {
public:
    int hour();        // 0..23
    int minutes();    // 0..59
    int seconds();    // 0..60 (incl. leap seconds)

    void set_hour(int h);
    ...
}
```

Jede Zeit von 00:00:00 bis 23:59:60 ist gültig

# Ursprung finden

```
bool Time::sane()
{
    return (0 <= hour() && hour() <= 23) &&
           (0 <= minutes() && minutes() <= 59) &&
           (0 <= seconds() && seconds() <= 60);
}

void Time::set_hour(int h)
{
    assert (sane()); // Vorbedingung
    ...
    assert (sane()); // Nachbedingung
}
```

# Ursprung finden

```
bool Time::sane()  
{  
    return (0 <= hour() && hour() <= 23) &&  
           (0 <= minutes() && minutes() <= 59) &&  
           (0 <= seconds() && seconds() <= 60);  
}
```

sane() ist die *Invariante* eines Time-Objekts:

- gilt *vor* jeder öffentlichen Methode
- gilt *nach* jeder öffentlichen Methode

# Ursprung finden

- Vorbedingung schlägt fehl = Infektion *vor* Methode
- Nachbedingung schlägt fehl = Infektion *nach* Methode
- Alle Zusicherungen ok = keine Infektion

```
void Time::set_hour(int h)
{
    assert (sane()); // Vorbedingung
    ...
    assert (sane()); // Nachbedingung
}
```

# Komplexe Invarianten

```
class RedBlackTree {  
    ...  
    boolean sane() {  
        assert (rootHasNoParent());  
        assert (rootIsBlack());  
        assert (redNodesHaveOnlyBlackChildren());  
        assert (equalNumberOfBlackNodesOnSubtrees());  
        assert (treeIsAcyclic());  
        assert (parentsAreConsistent());  
  
        return true;  
    }  
}
```

# Zusicherungen

				✓				
✓	✓	✓						
✓	✓	✓						
✓	✓	✓						
✓	✓	✓						
✓	✓	✓						
✓	✓	✓		✗				

↓ t

# Fokussieren

- Alle möglichen Einflüsse müssen geprüft werden
- Konzentration auf wahrscheinlichste Kandidaten
- Zusicherungen helfen schnell, Infektionen zu finden

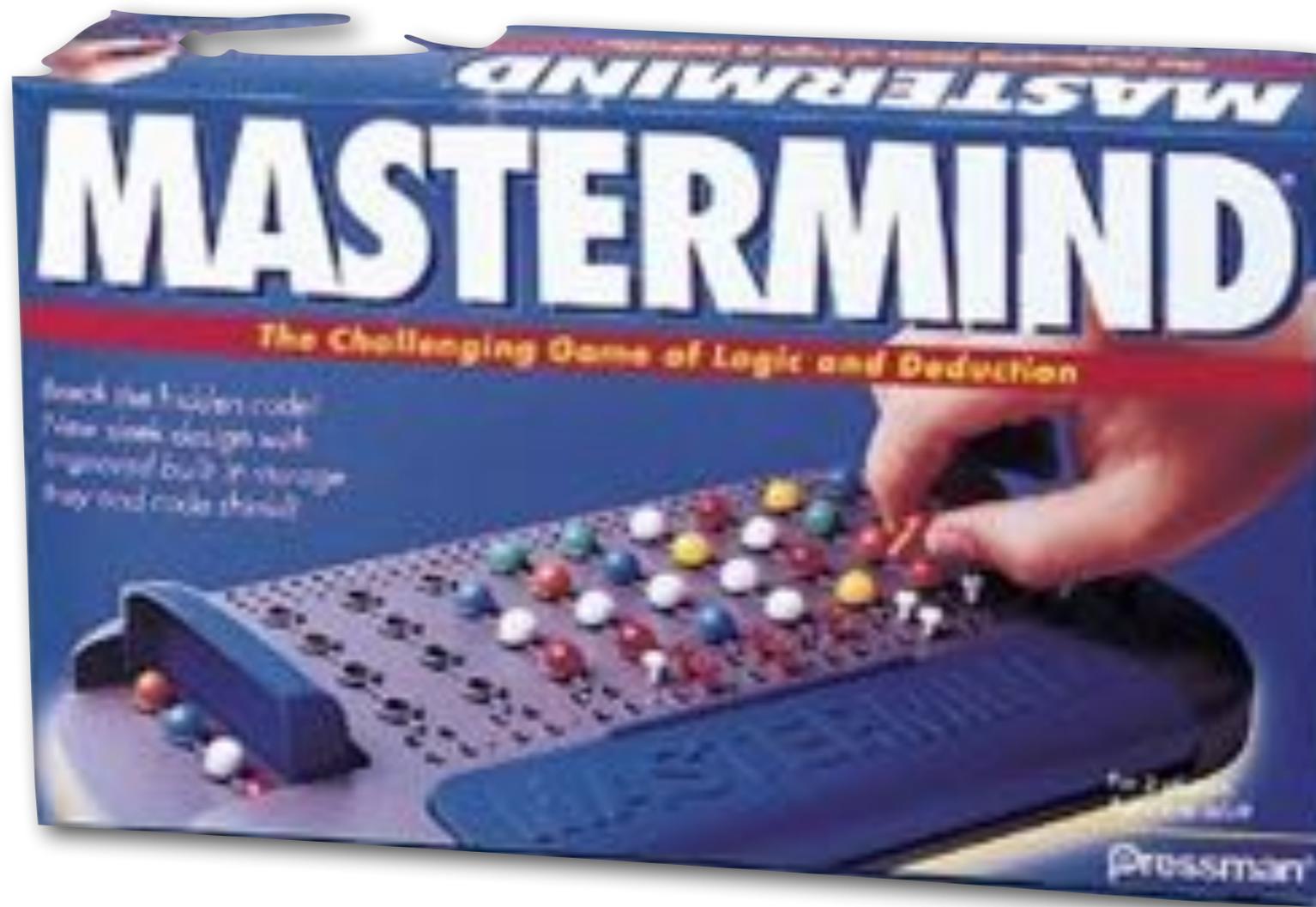
# Isolieren

- Fehlerursachen sollen *systematisch* eingeeengt werden – mit Beobachtungen und Experimenten.

# Wissenschaftliche Methode

1. Beobachte einen Teil des Universums
2. Erfinde eine *Hypothese*, die mit der Beobachtung übereinstimmt
3. Nutze die Hypothese, um Vorhersagen zu machen.
4. Teste die Vorhersagen durch Experimente oder Beobachtungen und passe die Hypothese an.
5. Wiederhole 3 and 4, bis die Hypothese zur *Theorie* wird.





# Explizite Hypothesen

Hypothesis	The execution causes $a[0] = 0$
Prediction	At time $t$ , $a[0] = 0$ should hold.
Experiment	At time $t$ , $a[0] = 0$ holds.
Observation	At time $t$ , $a[0] = 0$ as predicted.
Conclusion	Hypothesis is confirmed.

Wer alles im Kopf behält,  
spielt Mastermind blind!

# Explizite Hypothesen



# Isolieren

- Wir wiederholen die Suche nach Infektions-Ursprüngen, bis wir den Defekt gefunden haben.
- Wir gehen *systematisch* vor – im Sinne der wissenschaftlichen Methode
- Durch *explizite* Schritte leiten wir die Suche und können sie jederzeit nachvollziehen

# Korrektur

Vor der Korrektur müssen wir prüfen, ob der Defekt

- tatsächlich ein *Fehler* ist und
- das Fehlverhalten *verursacht*

Erst wenn beides verstanden ist, dürfen wir den Fehler korrigieren.

# The Devil's Guide to Debugging

Finde den Defekt durch Raten:

- Verstreue überall Debugging-Anweisungen
- Ändere den Code, bis etwas funktioniert
- Mache keine Kopien von alten Versionen
- Versuche gar nicht erst zu verstehen, was das Programm tun soll.

# The Devil's Guide to Debugging

Verschwende keine Zeit damit, dem Problem auf den Grund zu gehen

- Die meisten Probleme sind ohnehin trivial

# The Devil's Guide to Debugging

Benutze die offensichtlichste Reparatur:

- Repariere nur das, was Du siehst:

```
x = compute(y)
// compute(17) is wrong – fix it
if (y == 17)
    x = 25.15
```

Warum sich mit `compute()` beschäftigen?

T  
R  
A  
F  
F  
I  
C

# Erfolgreiche Korrektur



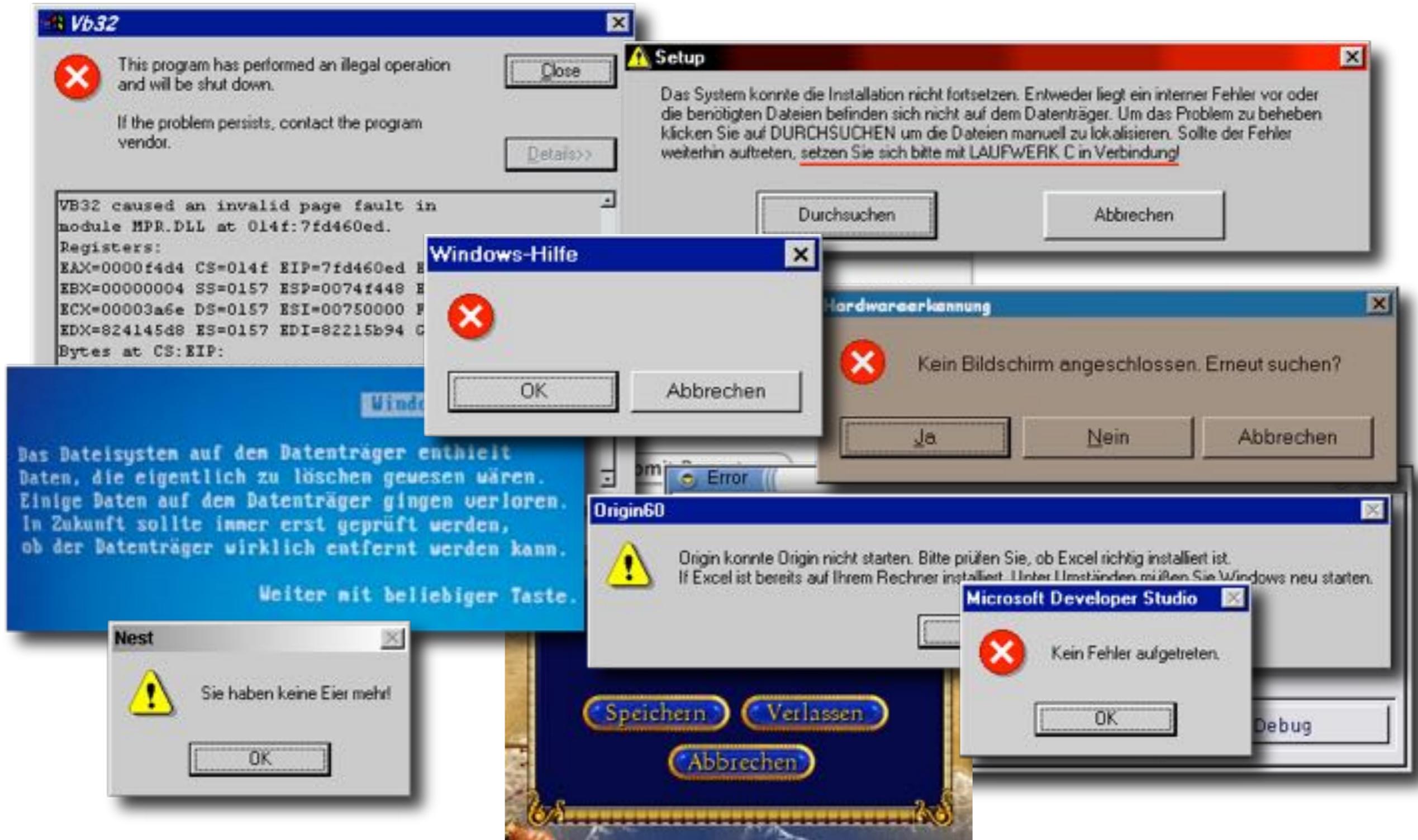
# Hausaufgaben

- Tritt das Fehlverhalten nicht mehr auf?  
(Falls doch, sollte dies eine große Überraschung sein)
- Könnte die Korrektur neue Fehler einführen?
- Wurde derselbe Fehler woanders gemacht?
- Ist meine Korrektur ins Versionsmanagement und Problem-Tracking eingespielt?

# Vorgehensweise

<b>T</b> rack the problem	<i>Problem verfolgen</i>
<b>R</b> eproduce	<i>Reproduzieren</i>
<b>A</b> utomate	<i>Automatisieren</i>
<b>F</b> ind Origins	<i>Ursprünge finden</i>
<b>F</b> ocus	<i>Fokussieren</i>
<b>I</b> solate	<i>Isolieren</i>
<b>C</b> orrect	<i>Korrigieren</i>

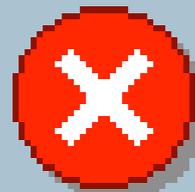
# Das Problem



# Was ist ein Problem?

- Ein *Problem* ist alles, was vom Benutzer als solches wahrgenommen wird
- Entwickler müssen dafür eine *Benutzer-Perspektive* einnehmen können

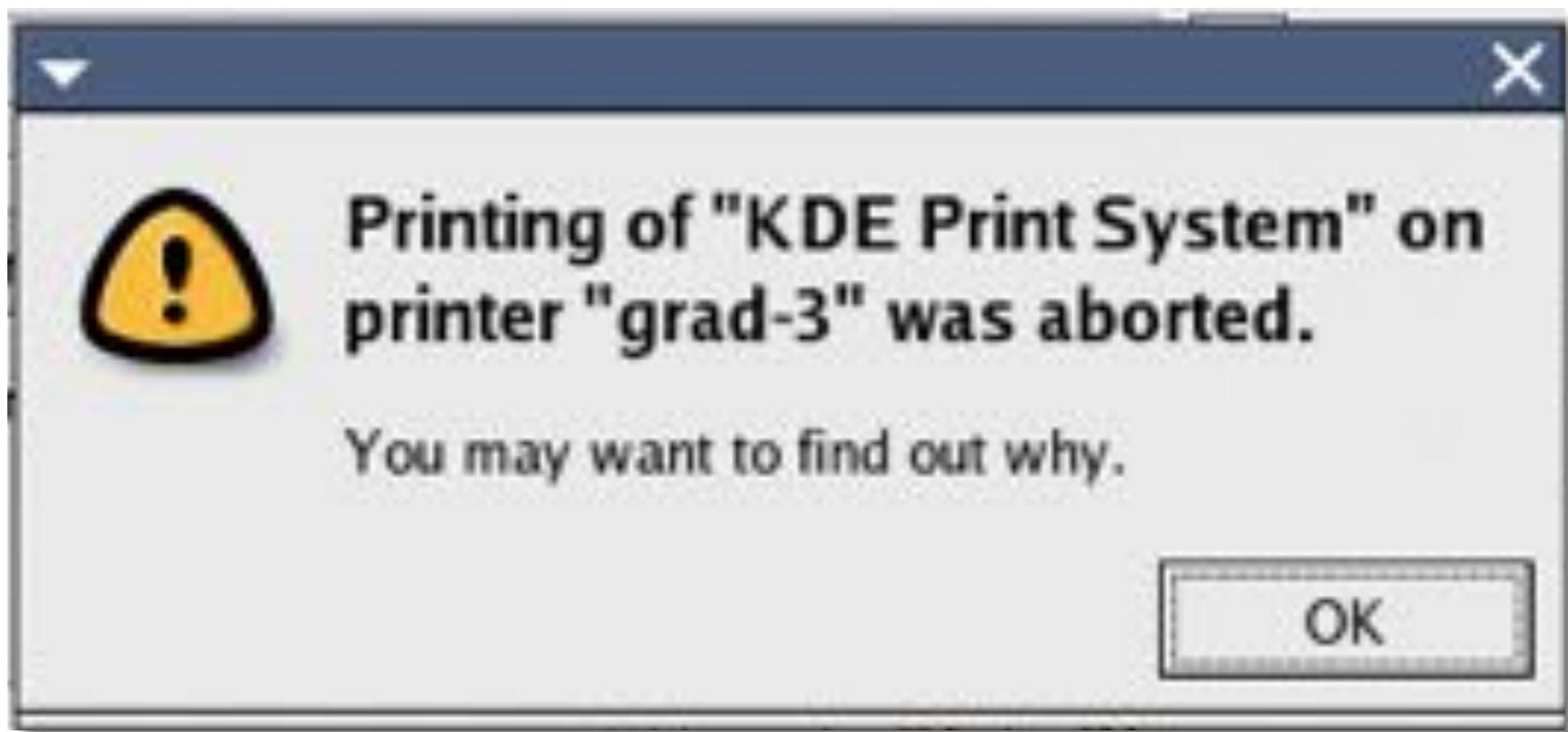
Microsoft Visual Basic



System Error &H80004005 (-2147467259). Unspecified error

OK

Help



**Printing of "KDE Print System" on printer "grad-3" was aborted.**

You may want to find out why.

OK



Eudora



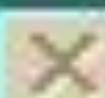
There has been an error transferring your mail. I said:

MAIL FROM:<mmcclinc@vt.edu>  
and then the SMTP server said:  
503 Polite people say HELO first



OK

This Really Happened...



Type mismatch

OK

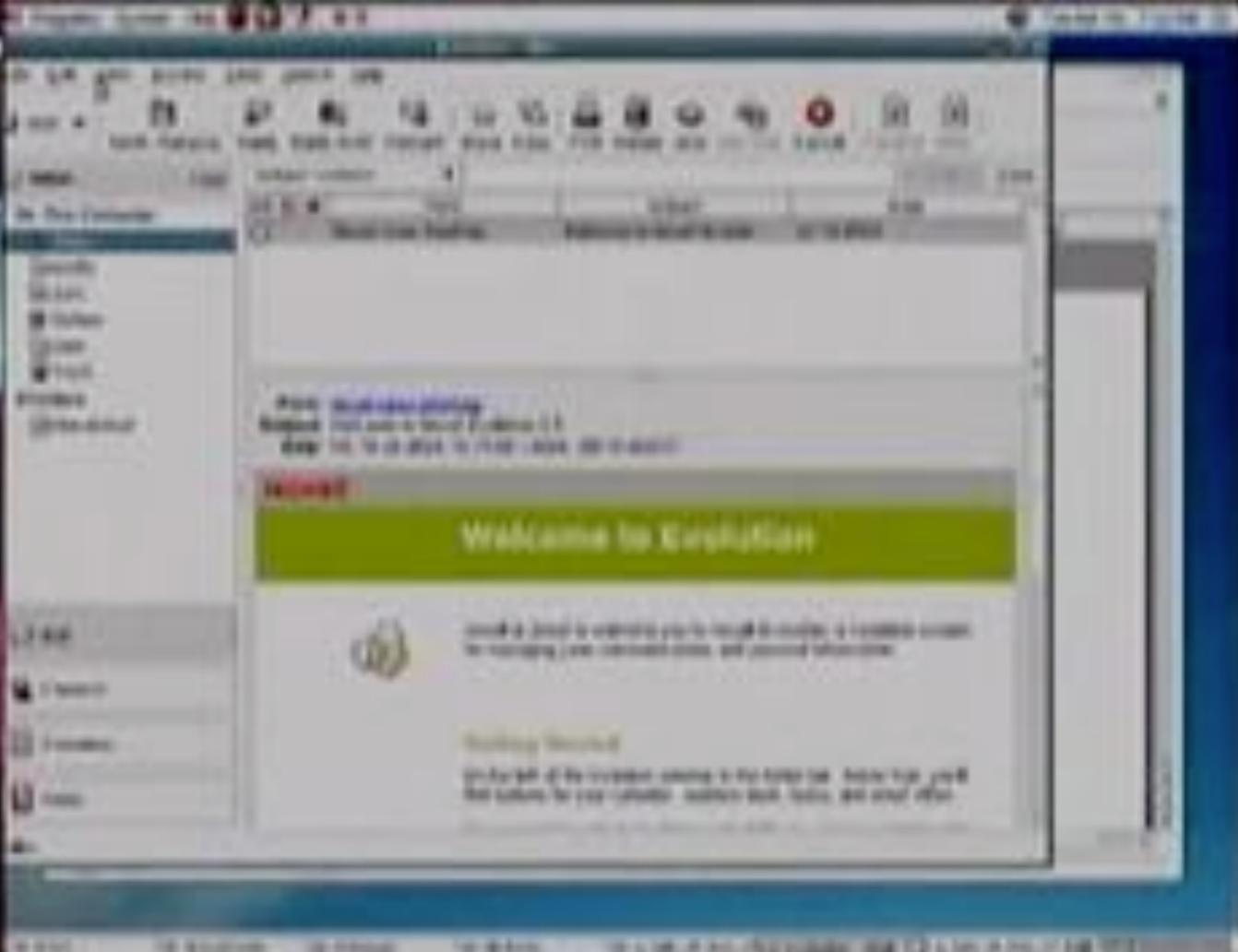
```
$ ssh somehost.foo.com
```

```
You don't exist, go away!
```

```
$ _
```

# Was ist ein Problem?

- Ein *Problem* ist alles, was vom Benutzer als solches wahrgenommen wird
- Entwickler müssen dafür eine *Benutzer-Perspektive* einnehmen können
- Lösung: *Test mit echten Benutzern!*

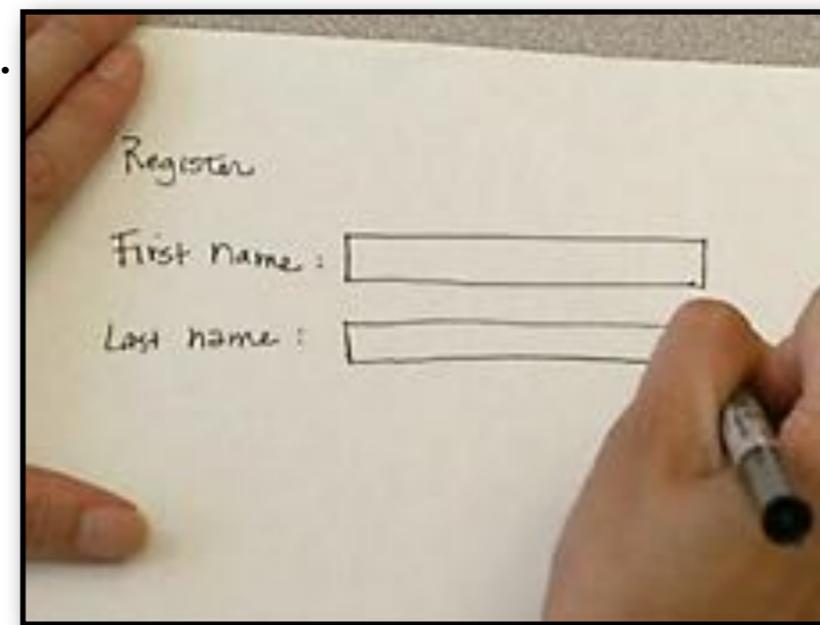


VIDEO LOSS  
26/03/28  
02:26:35

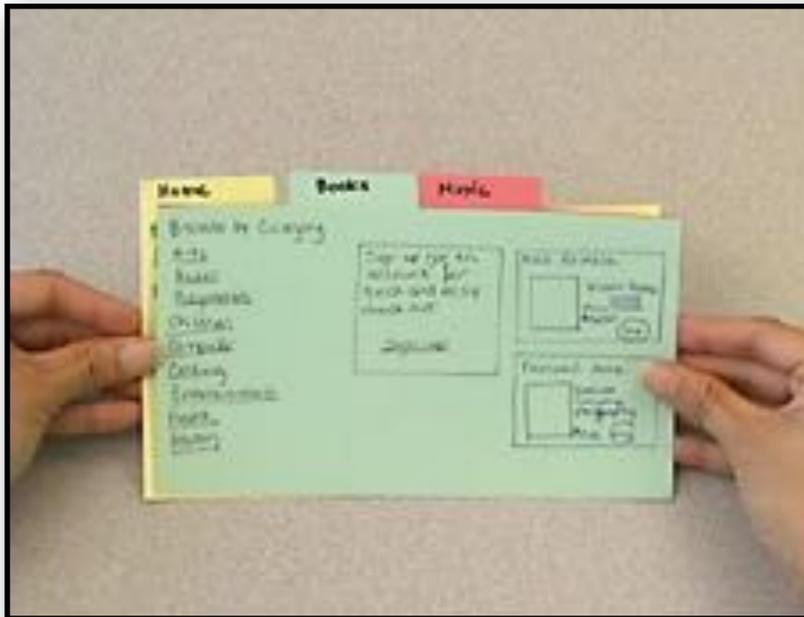
# Paper Prototyping

Credits: Nielsen Norman Group

1.



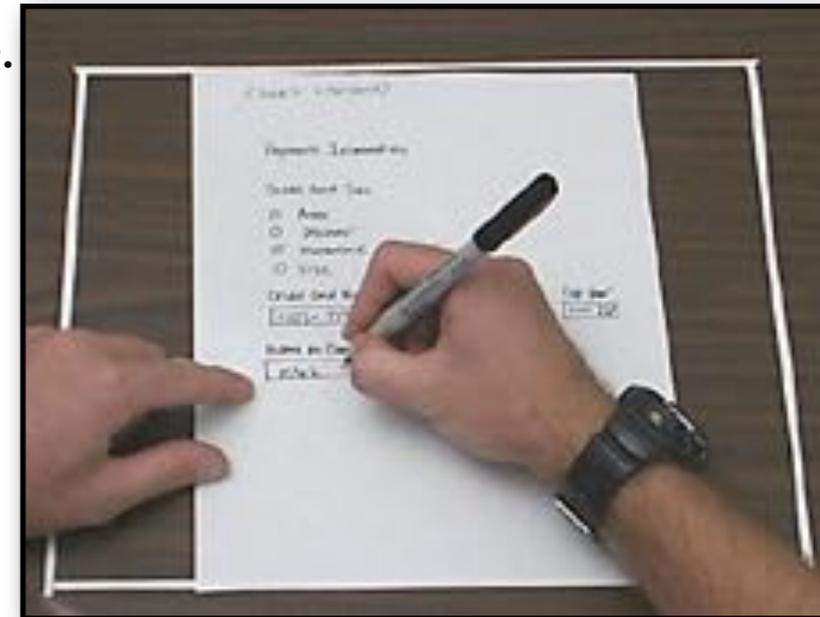
2.



3.



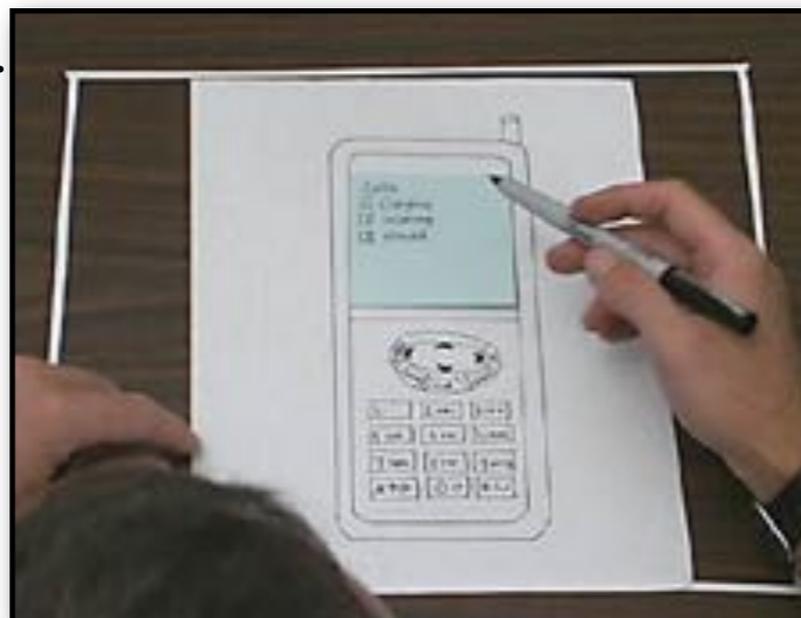
4.



5.



6.



7.



# Register

First Name :

Last Name :

**Market**

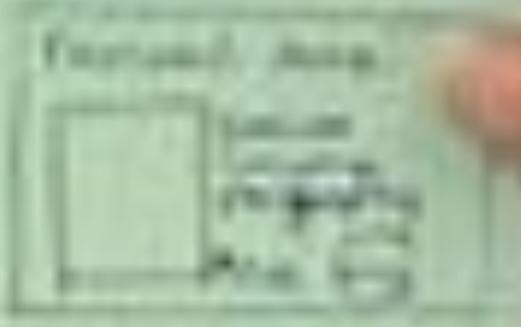
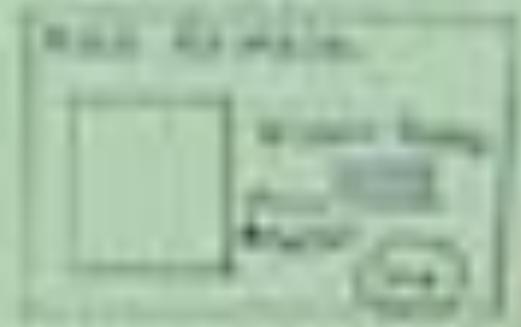
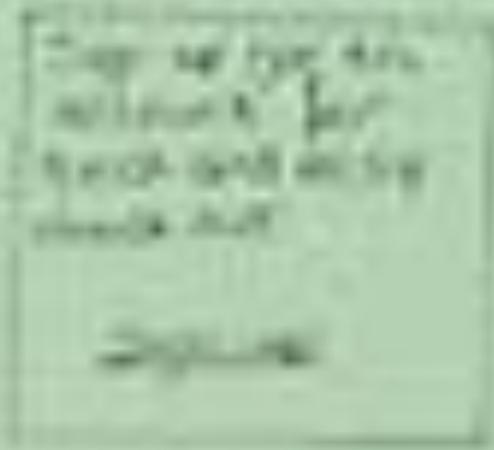
**Books**

**Media**

Credits: Nielsen Norman Group

**Books by Category**

- 4-11
- 12-17
- 18-24
- 25-31
- 32-38
- 39-45
- 46-52
- 53-59
- 60-66
- 67-73
- 74-80
- 81-87
- 88-94
- 95-101
- 102-108
- 109-115
- 116-122
- 123-129
- 130-136
- 137-143
- 144-150
- 151-157
- 158-164
- 165-171
- 172-178
- 179-185
- 186-192
- 193-199
- 200-206
- 207-213
- 214-220
- 221-227
- 228-234
- 235-241
- 242-248
- 249-255
- 256-262
- 263-269
- 270-276
- 277-283
- 284-290
- 291-297
- 298-304
- 305-311
- 312-318
- 319-325
- 326-332
- 333-339
- 340-346
- 347-353
- 354-360
- 361-367
- 368-374
- 375-381
- 382-388
- 389-395
- 396-402
- 403-409
- 410-416
- 417-423
- 424-430
- 431-437
- 438-444
- 445-451
- 452-458
- 459-465
- 466-472
- 473-479
- 480-486
- 487-493
- 494-500
- 501-507
- 508-514
- 515-521
- 522-528
- 529-535
- 536-542
- 543-549
- 550-556
- 557-563
- 564-570
- 571-577
- 578-584
- 585-591
- 592-598
- 599-605
- 606-612
- 613-619
- 620-626
- 627-633
- 634-640
- 641-647
- 648-654
- 655-661
- 662-668
- 669-675
- 676-682
- 683-689
- 690-696
- 697-703
- 704-710
- 711-717
- 718-724
- 725-731
- 732-738
- 739-745
- 746-752
- 753-759
- 760-766
- 767-773
- 774-780
- 781-787
- 788-794
- 795-801
- 802-808
- 809-815
- 816-822
- 823-829
- 830-836
- 837-843
- 844-850
- 851-857
- 858-864
- 865-871
- 872-878
- 879-885
- 886-892
- 893-899
- 900-906
- 907-913
- 914-920
- 921-927
- 928-934
- 935-941
- 942-948
- 949-955
- 956-962
- 963-969
- 970-976
- 977-983
- 984-990
- 991-997
- 998-1004
- 1005-1011
- 1012-1018
- 1019-1025
- 1026-1032
- 1033-1039
- 1040-1046
- 1047-1053
- 1054-1060
- 1061-1067
- 1068-1074
- 1075-1081
- 1082-1088
- 1089-1095
- 1096-1102
- 1103-1109
- 1110-1116
- 1117-1123
- 1124-1130
- 1131-1137
- 1138-1144
- 1145-1151
- 1152-1158
- 1159-1165
- 1166-1172
- 1173-1179
- 1180-1186
- 1187-1193
- 1194-1200
- 1201-1207
- 1208-1214
- 1215-1221
- 1222-1228
- 1229-1235
- 1236-1242
- 1243-1249
- 1250-1256
- 1257-1263
- 1264-1270
- 1271-1277
- 1278-1284
- 1285-1291
- 1292-1298
- 1299-1305
- 1306-1312
- 1313-1319
- 1320-1326
- 1327-1333
- 1334-1340
- 1341-1347
- 1348-1354
- 1355-1361
- 1362-1368
- 1369-1375
- 1376-1382
- 1383-1389
- 1390-1396
- 1397-1403
- 1404-1410
- 1411-1417
- 1418-1424
- 1425-1431
- 1432-1438
- 1439-1445
- 1446-1452
- 1453-1459
- 1460-1466
- 1467-1473
- 1474-1480
- 1481-1487
- 1488-1494
- 1495-1501
- 1502-1508
- 1509-1515
- 1516-1522
- 1523-1529
- 1530-1536
- 1537-1543
- 1544-1550
- 1551-1557
- 1558-1564
- 1565-1571
- 1572-1578
- 1579-1585
- 1586-1592
- 1593-1599
- 1600-1606
- 1607-1613
- 1614-1620
- 1621-1627
- 1628-1634
- 1635-1641
- 1642-1648
- 1649-1655
- 1656-1662
- 1663-1669
- 1670-1676
- 1677-1683
- 1684-1690
- 1691-1697
- 1698-1704
- 1705-1711
- 1712-1718
- 1719-1725
- 1726-1732
- 1733-1739
- 1740-1746
- 1747-1753
- 1754-1760
- 1761-1767
- 1768-1774
- 1775-1781
- 1782-1788
- 1789-1795
- 1796-1802
- 1803-1809
- 1810-1816
- 1817-1823
- 1824-1830
- 1831-1837
- 1838-1844
- 1845-1851
- 1852-1858
- 1859-1865
- 1866-1872
- 1873-1879
- 1880-1886
- 1887-1893
- 1894-1900
- 1901-1907
- 1908-1914
- 1915-1921
- 1922-1928
- 1929-1935
- 1936-1942
- 1943-1949
- 1950-1956
- 1957-1963
- 1964-1970
- 1971-1977
- 1978-1984
- 1985-1991
- 1992-1998
- 1999-2005
- 2006-2012
- 2013-2019
- 2020-2026
- 2027-2033
- 2034-2040
- 2041-2047
- 2048-2054
- 2055-2061
- 2062-2068
- 2069-2075
- 2076-2082
- 2083-2089
- 2090-2096
- 2097-2103
- 2104-2110
- 2111-2117
- 2118-2124
- 2125-2131
- 2132-2138
- 2139-2145
- 2146-2152
- 2153-2159
- 2160-2166
- 2167-2173
- 2174-2180
- 2181-2187
- 2188-2194
- 2195-2201
- 2202-2208
- 2209-2215
- 2216-2222
- 2223-2229
- 2230-2236
- 2237-2243
- 2244-2250
- 2251-2257
- 2258-2264
- 2265-2271
- 2272-2278
- 2279-2285
- 2286-2292
- 2293-2299
- 2300-2306
- 2307-2313
- 2314-2320
- 2321-2327
- 2328-2334
- 2335-2341
- 2342-2348
- 2349-2355
- 2356-2362
- 2363-2369
- 2370-2376
- 2377-2383
- 2384-2390
- 2391-2397
- 2398-2404
- 2405-2411
- 2412-2418
- 2419-2425
- 2426-2432
- 2433-2439
- 2440-2446
- 2447-2453
- 2454-2460
- 2461-2467
- 2468-2474
- 2475-2481
- 2482-2488
- 2489-2495
- 2496-2502
- 2503-2509
- 2510-2516
- 2517-2523
- 2524-2530
- 2531-2537
- 2538-2544
- 2545-2551
- 2552-2558
- 2559-2565
- 2566-2572
- 2573-2579
- 2580-2586
- 2587-2593
- 2594-2600
- 2601-2607
- 2608-2614
- 2615-2621
- 2622-2628
- 2629-2635
- 2636-2642
- 2643-2649
- 2650-2656
- 2657-2663
- 2664-2670
- 2671-2677
- 2678-2684
- 2685-2691
- 2692-2698
- 2699-2705
- 2706-2712
- 2713-2719
- 2720-2726
- 2727-2733
- 2734-2740
- 2741-2747
- 2748-2754
- 2755-2761
- 2762-2768
- 2769-2775
- 2776-2782
- 2783-2789
- 2790-2796
- 2797-2803
- 2804-2810
- 2811-2817
- 2818-2824
- 2825-2831
- 2832-2838
- 2839-2845
- 2846-2852
- 2853-2859
- 2860-2866
- 2867-2873
- 2874-2880
- 2881-2887
- 2888-2894
- 2895-2901
- 2902-2908
- 2909-2915
- 2916-2922
- 2923-2929
- 2930-2936
- 2937-2943
- 2944-2950
- 2951-2957
- 2958-2964
- 2965-2971
- 2972-2978
- 2979-2985
- 2986-2992
- 2993-2999
- 3000-3006
- 3007-3013
- 3014-3020
- 3021-3027
- 3028-3034
- 3035-3041
- 3042-3048
- 3049-3055
- 3056-3062
- 3063-3069
- 3070-3076
- 3077-3083
- 3084-3090
- 3091-3097
- 3098-3104
- 3105-3111
- 3112-3118
- 3119-3125
- 3126-3132
- 3133-3139
- 3140-3146
- 3147-3153
- 3154-3160
- 3161-3167
- 3168-3174
- 3175-3181
- 3182-3188
- 3189-3195
- 3196-3202
- 3203-3209
- 3210-3216
- 3217-3223
- 3224-3230
- 3231-3237
- 3238-3244
- 3245-3251
- 3252-3258
- 3259-3265
- 3266-3272
- 3273-3279
- 3280-3286
- 3287-3293
- 3294-3300
- 3301-3307
- 3308-3314
- 3315-3321
- 3322-3328
- 3329-3335
- 3336-3342
- 3343-3349
- 3350-3356
- 3357-3363
- 3364-3370
- 3371-3377
- 3378-3384
- 3385-3391
- 3392-3398
- 3399-3405
- 3406-3412
- 3413-3419
- 3420-3426
- 3427-3433
- 3434-3440
- 3441-3447
- 3448-3454
- 3455-3461
- 3462-3468
- 3469-3475
- 3476-3482
- 3483-3489
- 3490-3496
- 3497-3503
- 3504-3510
- 3511-3517
- 3518-3524
- 3525-3531
- 3532-3538
- 3539-3545
- 3546-3552
- 3553-3559
- 3560-3566
- 3567-3573
- 3574-3580
- 3581-3587
- 3588-3594
- 3595-3601
- 3602-3608
- 3609-3615
- 3616-3622
- 3623-3629
- 3630-3636
- 3637-3643
- 3644-3650
- 3651-3657
- 3658-3664
- 3665-3671
- 3672-3678
- 3679-3685
- 3686-3692
- 3693-3699
- 3700-3706
- 3707-3713
- 3714-3720
- 3721-3727
- 3728-3734
- 3735-3741
- 3742-3748
- 3749-3755
- 3756-3762
- 3763-3769
- 3770-3776
- 3777-3783
- 3784-3790
- 3791-3797
- 3798-3804
- 3805-3811
- 3812-3818
- 3819-3825
- 3826-3832
- 3833-3839
- 3840-3846
- 3847-3853
- 3854-3860
- 3861-3867
- 3868-3874
- 3875-3881
- 3882-3888
- 3889-3895
- 3896-3902
- 3903-3909
- 3910-3916
- 3917-3923
- 3924-3930
- 3931-3937
- 3938-3944
- 3945-3951
- 3952-3958
- 3959-3965
- 3966-3972
- 3973-3979
- 3980-3986
- 3987-3993
- 3994-4000





Classroom Management

Support Strategies

Teacher Role

- (i) Plan
- (ii) Prepare
- (iii) Monitor
- (iv) Evaluate

Classroom Management

Support Strategies

Teacher Role

Support Strategies

Page No. \_\_\_\_\_  
Date \_\_\_\_\_



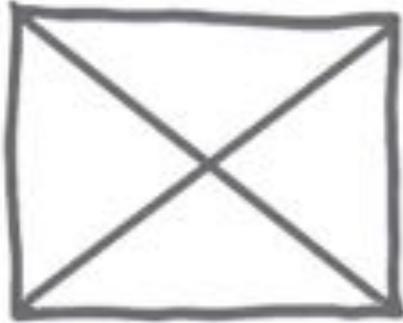


12:34  
100%



1	2	3
4	5	6
7	8	9
*	0	#





## Profile Name

245 Blackfriars Road

Ludgate House

London, SE1 9UY

Email: [firstname@surname.com](mailto:firstname@surname.com)

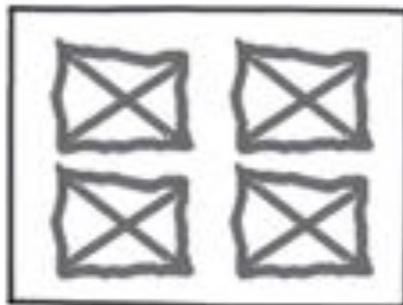
Telephone: 0207 955 3705

### Categories

Lorem ipsum  
dolor sit  
amet  
dolor sit

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi consectetur nibh feugiat urna elementum facilisis. Nullam diam arcu, lobortis ut tincidunt vel, suscipit quis lectus. Praesent interdum sapien in nisi tempor vestibulum. Mauris nec mauris sapien. Nam laoreet nisi non magna iaculis vitae convallis lorem porttitor.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi consectetur nibh feugiat urna elementum facilisis. Nullam diam arcu, lobortis ut tincidunt vel, suscipit quis lectus. Praesent interdum sapien in nisi tempor vestibulum. Mauris nec mauris



### Attachments

- [Lorem ipsum dolor sit amet.](#)

# Wireframing

# Systematische Fehlersuche

<b>T</b> rack the problem	<i>Problem verfolgen</i>
<b>R</b> eproduce	<i>Reproduzieren</i>
<b>A</b> utomate	<i>Automatisieren</i>
<b>F</b> ind Origins	<i>Ursprünge finden</i>
<b>F</b> ocus	<i>Fokussieren</i>
<b>I</b> solate	<i>Isolieren</i>
<b>C</b> orrect	<i>Korrigieren</i>