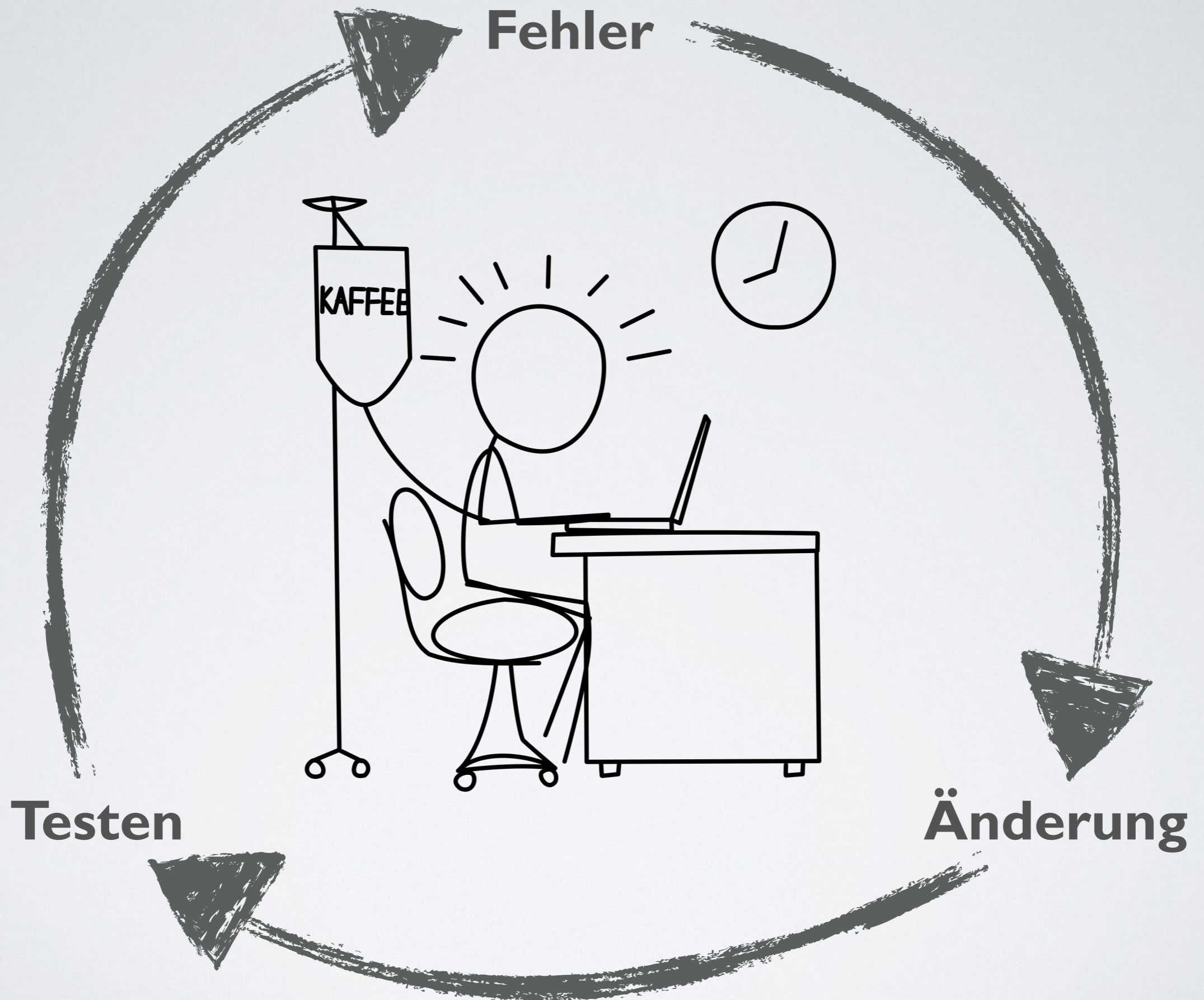


BEI UNSTESTEN
LAUTER AFFEN!

DAS ENDE DER BANANENSOFTWARE!

Dr. Jeremias Rößler



UMFRAGE

Wer hat manuell getestet?

Wer hat manuell regressionsgetestet?

Wem hat das Spaß gemacht?

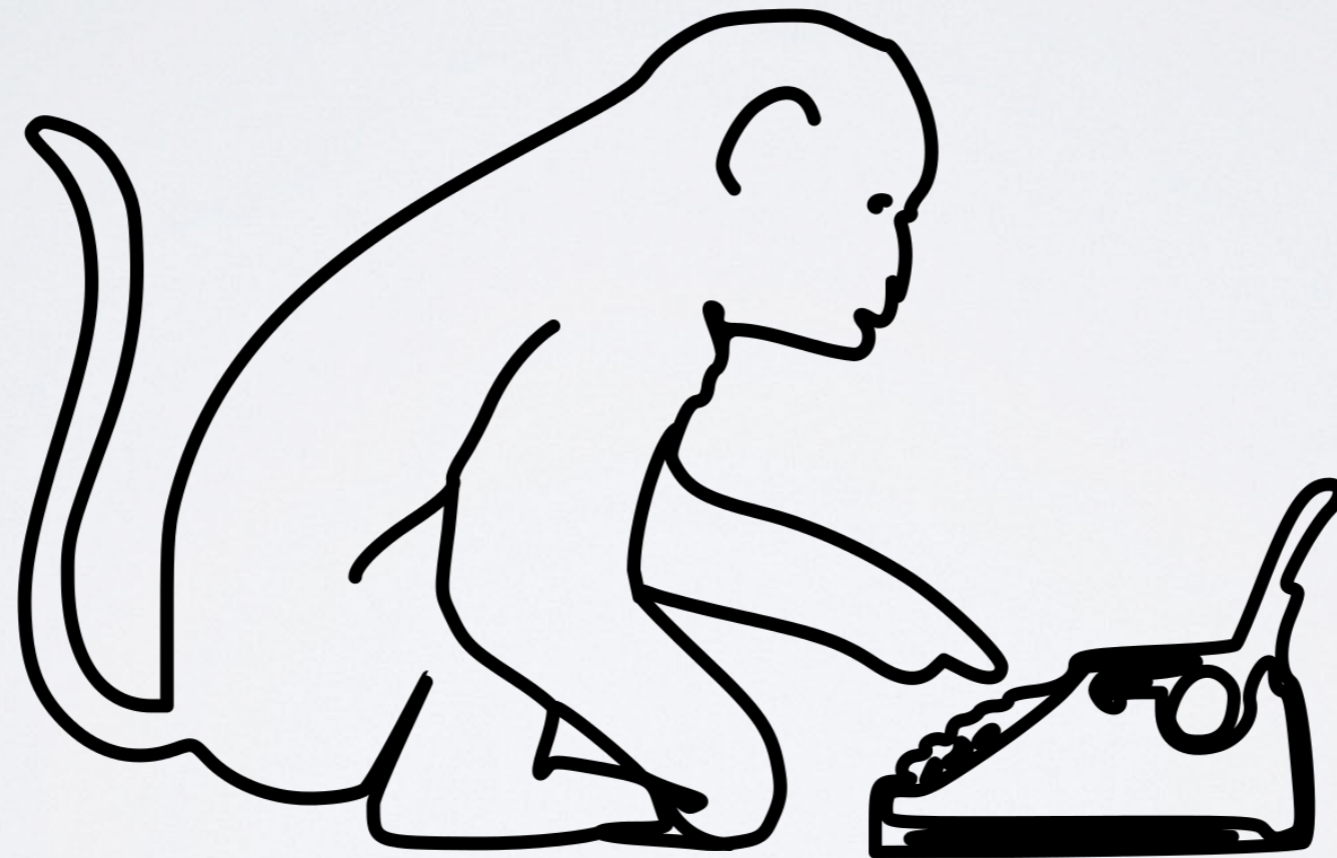
keine Automatisierung

automatische
Automatisierung!

keine Zeit



**Infinite-Monkey-Theorem:
Wenn eine Affe nur lange genug auf einer
Schreibmaschine tippt,
schreibt er irgendwann alle Werke von Shakespeare.**



**Wir ersetzen die Schreibmaschine
mit einem Computer..**



Infinite Monkey

```
1. public static void main(String... args) throws Exception {
2.     Robot robot = new Robot();
3.     while (true) {
4.         robot.mouseMove(random.nextInt(maxX), random.nextInt(maxY));
5.         robot.mousePress(InputEvent.BUTTON1_DOWN_MASK);
6.         robot.mouseRelease(InputEvent.BUTTON1_DOWN_MASK);
7.         robot.delay(200);
8.         for (char inputChar : randomString().toCharArray()) {
9.             robot.keyPress((int) inputChar);
10.            robot.keyRelease((int) inputChar);
11.            robot.delay(10);
12.        }
13.        robot.keyPress(KeyEvent.VK_ENTER);
14.        robot.keyRelease(KeyEvent.VK_ENTER);
15.    }
16. }
```

Infinite Monkey

DEMO

Vollautomatisches Regressionstesten.

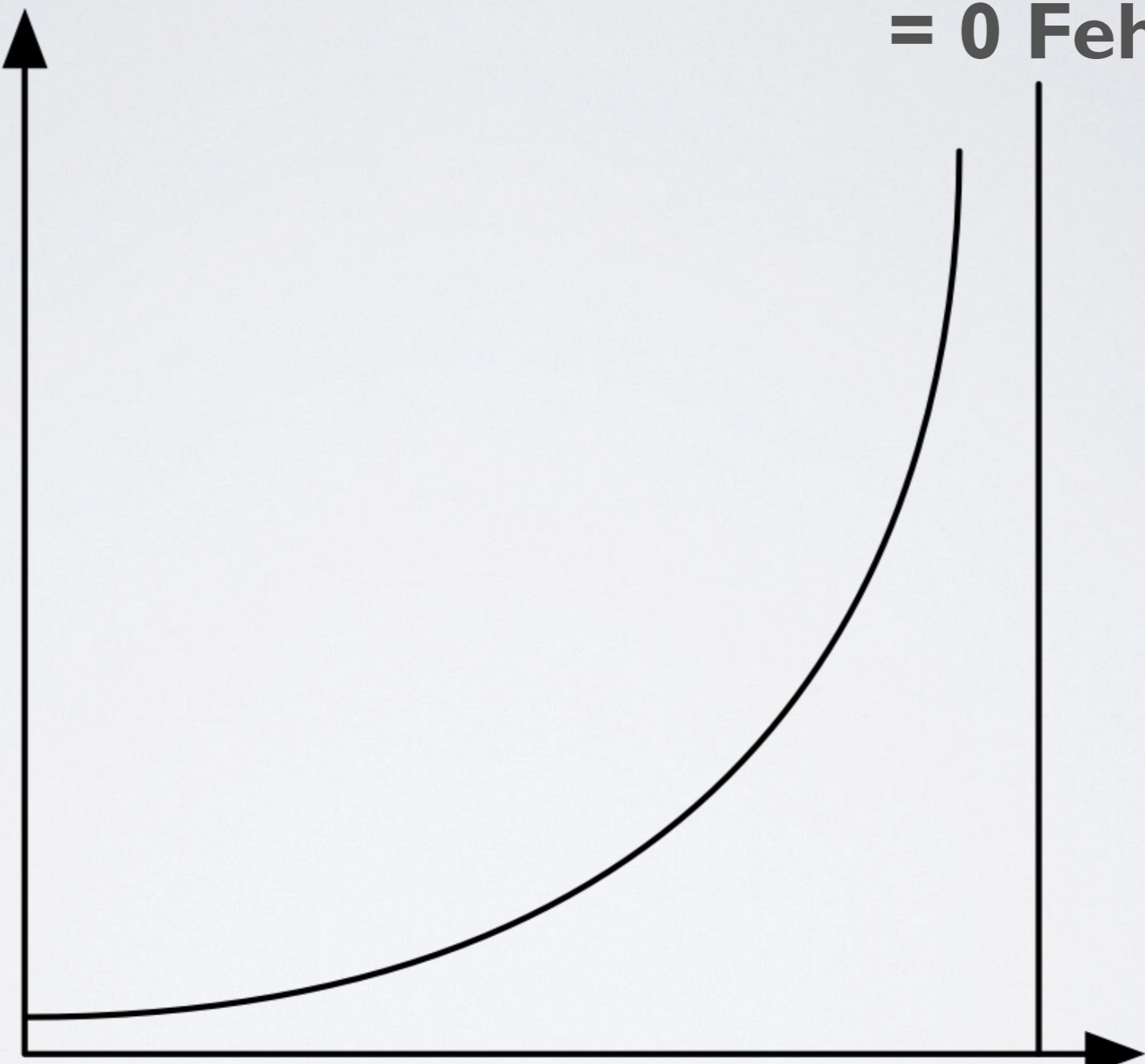
Weniger Risiko, weniger Stress, weniger Kosten!

Haben Sie genug vom Testen?

Direkt aus der Forschung bietet Ihnen
ReTest als bisher einziges Produkt
seiner Art vollautomatisches
Regressionstesten auf Ebene der
Benutzerschnittstelle. Damit stellen Sie
sicher, dass keine unerwünschten
Seiteneffekte mehr ins Projekt kommen.
Robustheitstesten von Corner Cases
bekommen Sie gratis dazu.
Mit ReTest behalten Sie die Kontrolle!



**Aufwand
pro Fehler**



**100%
= 0 Fehler**

Qualität

Dummer Affe



Intelligenter Affe



Intelligenter Affe

```
1. public static void main(String... args) throws Exception {
2.     WebDriver driver = new FirefoxDriver();
3.     driver.get("http://www.retest.de");
4.     while (true) {
5.         List<WebElement> links = driver.findElements(By.tagName("a"));
6.         links.get(random.nextInt(links.size())).click();
7.         Thread.sleep(500);
8.         List<WebElement> fields =
9.             driver.findElements(By.xpath("//input[@type='text']"));
10.        WebElement field = fields.get(random.nextInt(fields.size()));
11.        field.sendKeys(randomString());
12.        Thread.sleep(500);
13.    }
14. }
```

Intelligenter Affe

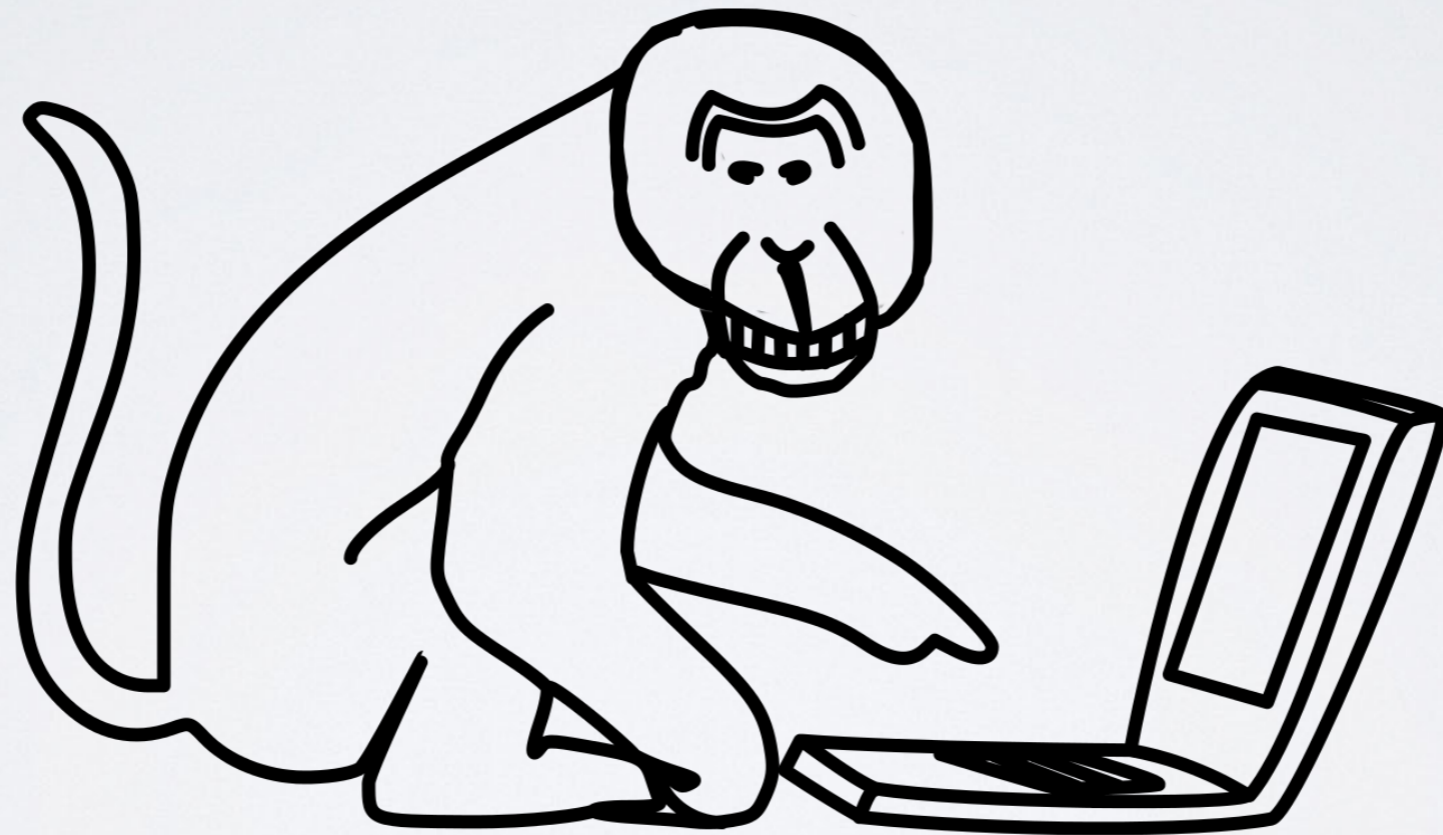
DEMO

Vollautomatisches Regressionstesten. Weniger Risiko, weniger Stress, weniger Kosten!

Haben Sie genug vom Testen?
Direkt aus der Forschung bietet Ihnen
ReTest als bisher einziges Produkt
seiner Art vollautomatisches
Regressionstesten auf Ebene der
Benutzerschnittstelle. Damit stellen Sie
sicher, dass keine unerwünschten
Seiteneffekte mehr ins Projekt kommen.
Robustheitstesten von Corner Cases
bekommen Sie gratis dazu.
Mit ReTest behalten Sie die Kontrolle!



Intelligenter Affe



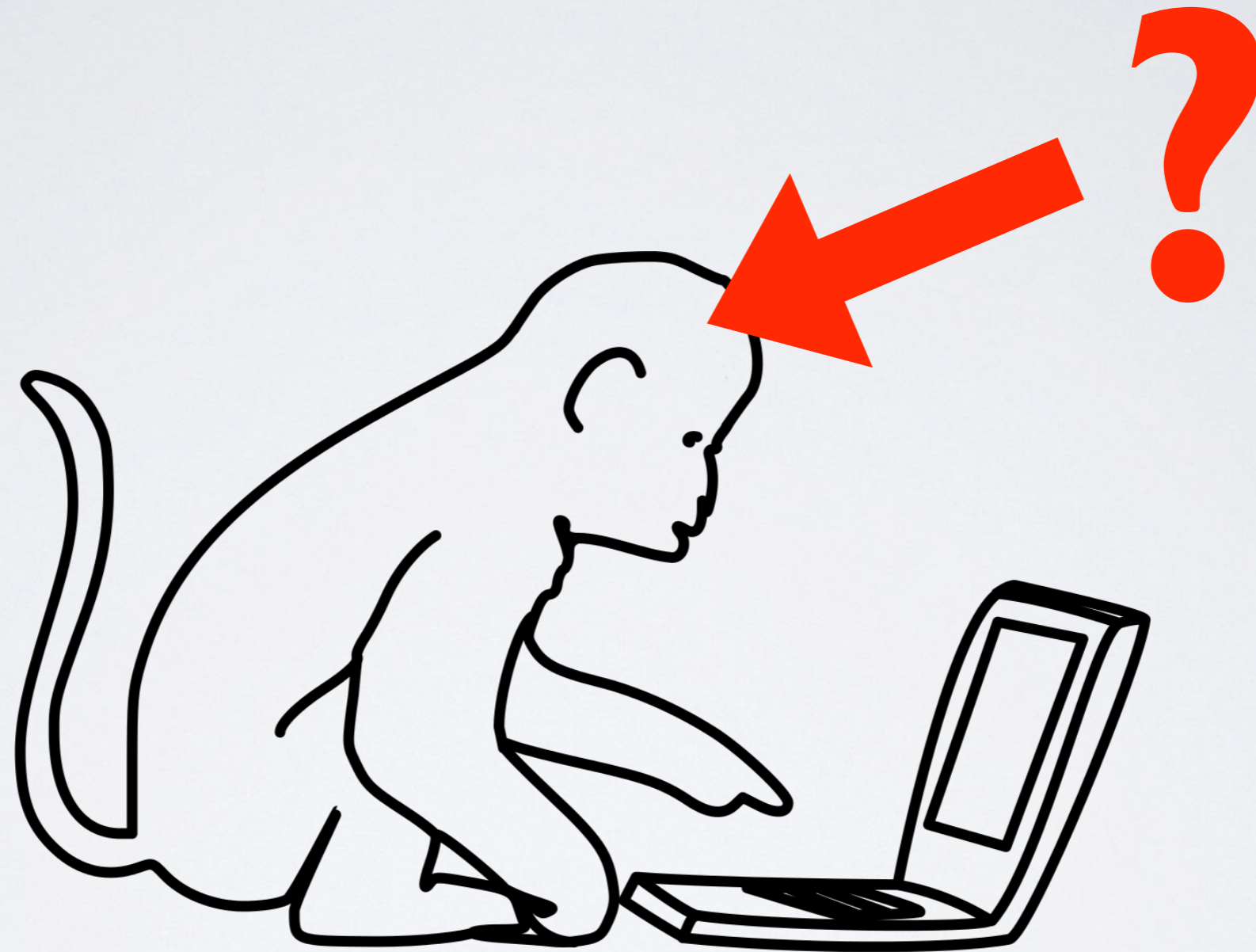
Open Source Produkte

NetFlix Chaos Monkey

gremlins.js

UI/Application Exerciser Monkey

Intelligenter Affe



HumiCalc with Uncertainty

File Options Help

Configuration

Temperature Scale: ITS-90 Carrier Gas: Dry Air Mode: Normal

Equilibrium Over: Water Apply Enhancement Factors: Known: Dew Point

Known Values (Standard u)

Dew Point: 10.0 ±0.0577

Temperature: 25.0 ±0.025

Pressure: 14.7 ±0.025

Calculate

Units

Temperature: °C Pressure: psia Vapor Pressure: Pa Density and Abs Humidity: g/m³ Enthalpy: J/g

Calculated Values (Expanded U with 95.45% Confidence)

%RH	38.7340793	±0.3212	Specific Humidity	0.0076028	±6E-005
Frost Point			Absolute Humidity	8.962178468	±0.0694
Dew Point	10.0	±0.1155	Dry Air Density	1169.837556	±4.0342
PPMv	12314.00364	±105.31	Moist Air Density	1178.799735	±4.033
PPMw	7661.045263	±65.521	Saturation Temperature		
Grains/lb	53.62731684	±0.4586	Saturation Pressure		
Enthalpy	44.63021277	±0.1744	Wet Bulb Temperature	16.10904106	±0.0559
SVP@Tt	3169.90395	±9.4493	Mixing Ratio by Volume	0.012314004	±0.0001
SVP@Td	1228.13339	±9.5028	Mixing Ratio by Weight	0.007661045	±7E-005
SVP@Ts			Percent by Volume	1.216421347	±0.0103
F@Tt,Pt	1.004109348	±1E-005	Percent by Weight	0.760279987	±0.0065
F@Td,Pt	1.003863842	±1E-005	Vapor Mole Fraction	0.012164213	±0.0001
F@Ts,Ps			Dry Air Mole Fraction	0.987835787	±0.0001

Jahreszeugnisse RBE

Name, Vorname: **Mastmann Emil**

Kapitel: Verhalten: 2 Mitarbeit: 2

Leistungen in den einzelnen Fächern u. Fächerverbänden:

Engl.	4	Mat.	4
Deutsch.	4	Physik.	4,3
Englisch.	4	Chemie.	3,9
Mathematik.	4	Biologie.	3,7
Geschichte.	0	Sport.	3
EwG.	3	Musik.	4
Erkunde.		Bildende Kunst.	4
Gemeinschaftsle...		Technik.	3
TOP Informatiksch...	2	TOP WVR.	0
TOP Tech. Arbeiten...		TOP Sozial. Engagement.	2

Arbeitsgemeinschaften: AG Auswahl

Fehltag: 5 unentschuldig; 0 Fehlzeiten; 2 unentschuldig; 0

Bemerkungen: Versetzt: Emil war für die Klassenkasse zuständig.

Verstärkt: Ja / Nicht verstärkt: Nein / Versetzt nach §10(3): Ja

Eintrag: 3,4

Lagerverwaltung WN 2000 Dienstag, 5. Juni 2007 Jahr/Woche 07/23 Demo Version GmbH

Lager-Liste

Artikel: ALU00700 Spezifikation: Dicke: 10.000 in mm Bestand: 2.000 Kg

Spezial: Prof: Flach N. Benutzt: 0.000

Umschung: 1.0000 ppa in arb Spec. Dichte: 2.700 kg/dm³ Grundfläche: 0.0400 dm²

EK-Preis 1: 0,52 €/Stück EK-Preis 2: 0,56 €/Stück

Legenot: Durchschl. Min. Menge: 2.000

Artikel ist Best.: Ja B. Datum: 05.06.2007 B. Menge: 10.000 Kg Aktive Bestellungen: 1 St.

Bemerkung: Spezial-Alu für Demonstrator

S.F.K. GmbH 2003 - Geschäftsjahr 10-2003 - Buchstamm 01.01.2003 - 31.10.2003

Buchen: Sachkonten Allgemeinbeleg

Konto: 5210 Eigenkapital

Beleg: 03.223 OP Anz.Nr.: 03.223 Buchungsdatum: 01.10.03 Belegdatum: 01.10.03

Umsatz: 2.320,00 Soll Haben

Steuerbetrag: 300,00

Konto	SKN	SKZ	Beleg	Datum	Soll	Haben	Text	Faktura	Anz.Nr.
1610	6700		147	13.10.2003	30,60		Gewerbesteuer Rückz. 2001	147	
1610	4895		147	13.10.2003		500,00	Spesen		147
1610	4915	1	146	09.10.2003		107,70	Telefon		146
1610	0		145	08.10.2003	1.035,00		10-Punkte		145
1610	4270		144	06.10.2003		369,00	144 09.03		144
1610	4830		144	06.10.2003		966,44	144 9.03		144
1610	4895		143	02.10.2003		20,50	OP		143
1610	4895		143	02.10.2003		500,00	Erhaltung		143
Summe Primärkonto					7.260,30				

S.F.K. GmbH 2003 - Geschäftsjahr 10-2003 - Buchstamm 01.01.2003 - 31.10.2003

Buchen: Sachkonten Allgemeinbeleg

Konto: 5210 Eigenkapital

Beleg: 03.223 OP Anz.Nr.: 03.223 Buchungsdatum: 01.10.03 Belegdatum: 01.10.03

Umsatz: 2.320,00 Soll Haben

Steuerbetrag: 300,00

Konto	SKN	SKZ	Beleg	Datum	Soll	Haben	Text	Faktura	Anz.Nr.
1610	6700		147	13.10.2003	30,60		Gewerbesteuer Rückz. 2001	147	
1610	4895		147	13.10.2003		500,00	Spesen		147
1610	4915	1	146	09.10.2003		107,70	Telefon		146
1610	0		145	08.10.2003	1.035,00		10-Punkte		145
1610	4270		144	06.10.2003		369,00	144 09.03		144
1610	4830		144	06.10.2003		966,44	144 9.03		144
1610	4895		143	02.10.2003		20,50	OP		143
1610	4895		143	02.10.2003		500,00	Erhaltung		143
Summe Primärkonto					7.260,30				

Lagerverwaltung WN 2000 Dienstag, 5. Juni 2007 Jahr/Woche 07/23 Demo Version GmbH

Lager-Liste

Artikel: ALU00700 Spezifikation: Dicke: 10.000 in mm Bestand: 2.000 Kg

Spezial: Prof: Flach N. Benutzt: 0.000

Umschung: 1.0000 ppa in arb Spec. Dichte: 2.700 kg/dm³ Grundfläche: 0.0400 dm²

EK-Preis 1: 0,52 €/Stück EK-Preis 2: 0,56 €/Stück

Legenot: Durchschl. Min. Menge: 2.000

Artikel ist Best.: Ja B. Datum: 05.06.2007 B. Menge: 10.000 Kg Aktive Bestellungen: 1 St.

Bemerkung: Spezial-Alu für Demonstrator

Jahreszeugnisse RBE

Name, Vorname: **Mastmann Emil**

Kapitel: Verhalten: 2 Mitarbeit: 2

Leistungen in den einzelnen Fächern u. Fächerverbänden:

Engl.	4	Mat.	4
Deutsch.	4	Physik.	4,3
Englisch.	4	Chemie.	3,9
Mathematik.	4	Biologie.	3,7
Geschichte.	0	Sport.	3
EwG.	3	Musik.	4
Erkunde.		Bildende Kunst.	4
Gemeinschaftsle...		Technik.	3
TOP Informatiksch...	2	TOP WVR.	0
TOP Tech. Arbeiten...		TOP Sozial. Engagement.	2

Arbeitsgemeinschaften: AG Auswahl

Fehltag: 5 unentschuldig; 0 Fehlzeiten; 2 unentschuldig; 0

Bemerkungen: Versetzt: Emil war für die Klassenkasse zuständig.

Verstärkt: Ja / Nicht verstärkt: Nein / Versetzt nach §10(3): Ja

Eintrag: 3,4

S.F.K. GmbH 2003 - Geschäftsjahr 10-2003 - Buchstamm 01.01.2003 - 31.10.2003

Buchen: Sachkonten Allgemeinbeleg

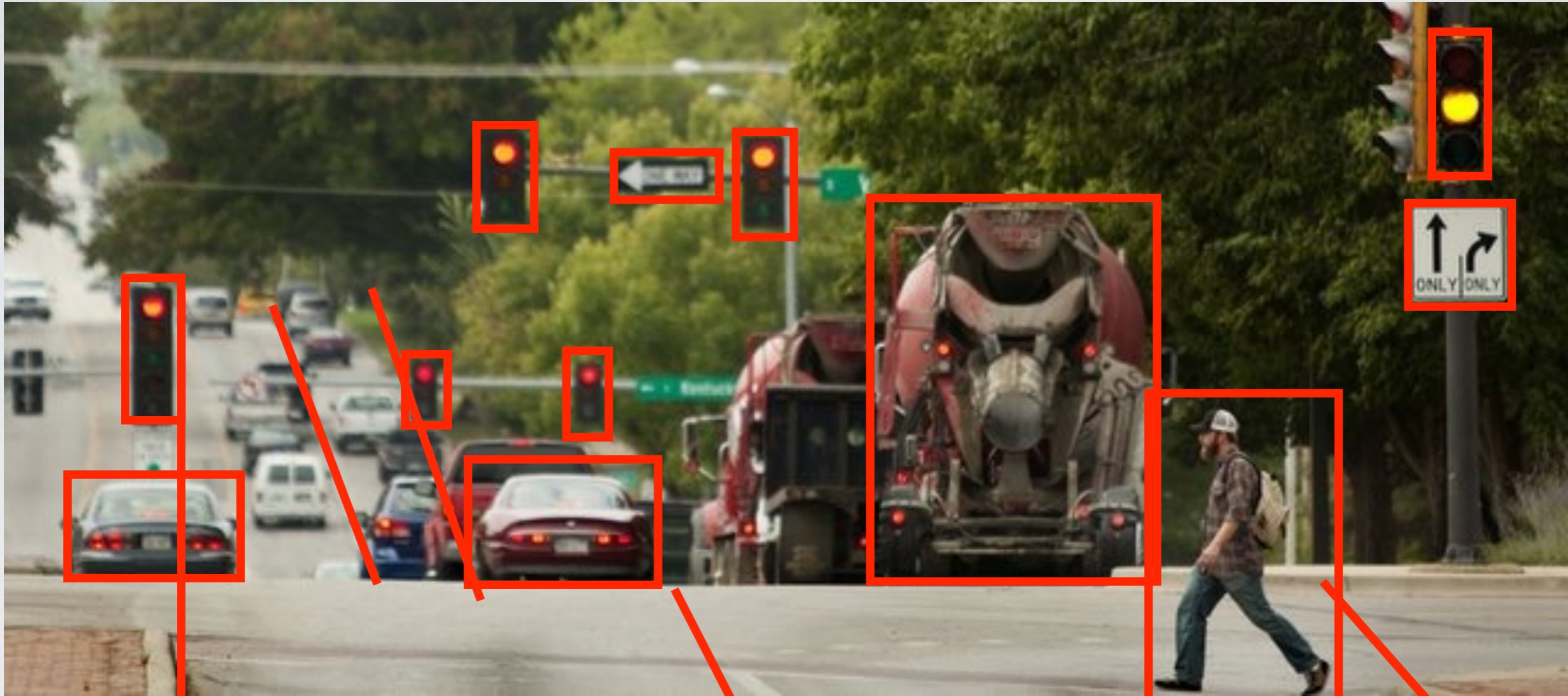
Konto: 5210 Eigenkapital

Beleg: 03.223 OP Anz.Nr.: 03.223 Buchungsdatum: 01.10.03 Belegdatum: 01.10.03

Umsatz: 2.320,00 Soll Haben

Steuerbetrag: 300,00

Konto	SKN	SKZ	Beleg	Datum	Soll	Haben	Text	Faktura	Anz.Nr.
1610	6700		147	13.10.2003	30,60		Gewerbesteuer Rückz. 2001	147	
1610	4895		147	13.10.2003		500,00	Spesen		147
1610	4915	1	146	09.10.2003		107,70	Telefon		146
1610	0		145	08.10.2003	1.035,00		10-Punkte		145
1610	4270		144	06.10.2003		369,00	144 09.03		144
1610	4830		144	06.10.2003		966,44	144 9.03		144
1610	4895		143	02.10.2003		20,50	OP		143
1610	4895		143	02.10.2003		500,00	Erhaltung		143
Summe Primärkonto					7.260,30				



Problemraum

**Problem
raum**





Jahreszeugnisse RBE

Name, Vorname: **Mastmann** **Emil**

Kapitelnummer: 2 Mitarbeit: 2

Leistungen in den einzelnen Fächern u. Fächerverbindungen:

EPK	4	Mat.	4
Deutsch	4	Physik	4,3
Englisch	4	Chemie	3,9
Mathematik	4	Biologie	3,7
Geschichte	0	Sport	3
EwC	3	Musik	4
Erkunde		Bildende Kunst	4
Gemeinschaftslehre		Technik	3
TOP Informatik	2	TOP WIR	0
TOP Tech. Arbeiten		TOP Sozial. Engagement	2

Anwahl

Erster Schüler: 4 | Letzter Schüler:

Eingabemodus: **Noteneingabe nach Schülern** | **Noteneingabe nach Fächern**

Notenberechnung: **3,4**

Lagerverwaltung WN 2000 Dienstag, 5. Juni 2007 Jahr/Woche 07/23

Lager-Liste

Artikel: ALU00700
Spezifikation: Dicke 10,000 in mm
Bestand 1: 2.000 Kg
Bestand 2: 1.952 in

Umschung: 1.0000
EK-Preis 1.0: 5,52 €/Stk
EK-Preis 2.0: 5,96 €/Stk
Lagerort: Durchwahl

S.F.K. GmbH 2003 - Geschäftsjahr 10-2003 - Buchrahmen 01.01.2003 - 31.10.2003

Buchen: Sachkonten Allgemeinbeleg

Deutsche Asphalt HUPFH EUR 0,00 Erlöse Hotline-Wartung EUR 142.747,17 H

OP Anz.Nr.: 03.223 | Buchungdatum: 01.10.03 | Belegdatum: 01.10.03

Konto	Saldo	Haben	Haben	Text	Faktura	Anz.Nr.	
1610	6700	147	13.10.2003	30,66	Gewerbesteuer Rückz. 2001	147	
1610	4895	147	13.10.2003	500,00	Speisen	147	
1610	6915	1	146	09.10.2003	107,76	Telefon	146
1610	0	145	08.10.2003	1.875,00	HP-Prüfung	145	
1610	4270	144	06.10.2003	369,00	144 09.03	144	
1610	4830	144	06.10.2003	966,44	144 9.03	144	
1610	4895	143	02.10.2003	25,50	OP	143	
1610	4895	143	02.10.2003	500,00	Erhaltung	143	
Summe Primärkonten		7.260,30					

S.F.K. GmbH 2003 - Geschäftsjahr 10-2003 - Buchrahmen 01.01.2003 - 31.10.2003

Buchen: Sachkonten Allgemeinbeleg

Deutsche Asphalt HUPFH EUR 0,00 Erlöse Hotline-Wartung EUR 142.747,17 H

OP Anz.Nr.: 03.223 | Buchungdatum: 01.10.03 | Belegdatum: 01.10.03

Konto	Saldo	Haben	Haben	Text	Faktura	Anz.Nr.	
1610	6700	147	13.10.2003	30,66	Gewerbesteuer Rückz. 2001	147	
1610	4895	147	13.10.2003	500,00	Speisen	147	
1610	6915	1	146	09.10.2003	107,76	Telefon	146
1610	0	145	08.10.2003	1.875,00	HP-Prüfung	145	
1610	4270	144	06.10.2003	369,00	144 09.03	144	
1610	4830	144	06.10.2003	966,44	144 9.03	144	
1610	4895	143	02.10.2003	25,50	OP	143	
1610	4895	143	02.10.2003	500,00	Erhaltung	143	
Summe Primärkonten		7.260,30					

Lagerverwaltung WN 2000 Dienstag, 5. Juni 2007 Jahr/Woche 07/23

Lager-Liste

Artikel: ALU00700
Spezifikation: Dicke 10,000 in mm
Bestand 1: 2.000 Kg
Bestand 2: 1.952 in

Umschung: 1.0000
EK-Preis 1.0: 5,52 €/Stk
EK-Preis 2.0: 5,96 €/Stk
Lagerort: Durchwahl

Jahreszeugnisse RBE

Name, Vorname: **Mastmann** **Emil**

Kapitelnummer: 2 Mitarbeit: 2

Leistungen in den einzelnen Fächern u. Fächerverbindungen:

EPK	4	Mat.	4
Deutsch	4	Physik	4,3
Englisch	4	Chemie	3,9
Mathematik	4	Biologie	3,7
Geschichte	0	Sport	3
EwC	3	Musik	4
Erkunde		Bildende Kunst	4
Gemeinschaftslehre		Technik	3
TOP Informatik	2	TOP WIR	0
TOP Tech. Arbeiten		TOP Sozial. Engagement	2

Anwahl

Erster Schüler: 4 | Letzter Schüler:

Eingabemodus: **Noteneingabe nach Schülern** | **Noteneingabe nach Fächern**

Notenberechnung: **3,4**

S.F.K. GmbH 2003 - Geschäftsjahr 10-2003 - Buchrahmen 01.01.2003 - 31.10.2003

Buchen: Sachkonten Allgemeinbeleg

Deutsche Asphalt HUPFH EUR 0,00 Erlöse Hotline-Wartung EUR 142.747,17 H

OP Anz.Nr.: 03.223 | Buchungdatum: 01.10.03 | Belegdatum: 01.10.03

Konto	Saldo	Haben	Haben	Text	Faktura	Anz.Nr.	
1610	6700	147	13.10.2003	30,66	Gewerbesteuer Rückz. 2001	147	
1610	4895	147	13.10.2003	500,00	Speisen	147	
1610	6915	1	146	09.10.2003	107,76	Telefon	146
1610	0	145	08.10.2003	1.875,00	HP-Prüfung	145	
1610	4270	144	06.10.2003	369,00	144 09.03	144	
1610	4830	144	06.10.2003	966,44	144 9.03	144	
1610	4895	143	02.10.2003	25,50	OP	143	
1610	4895	143	02.10.2003	500,00	Erhaltung	143	
Summe Primärkonten		7.260,30					

___ / ___ / ____



Geburtsdatum

Geburtsdatum ✕

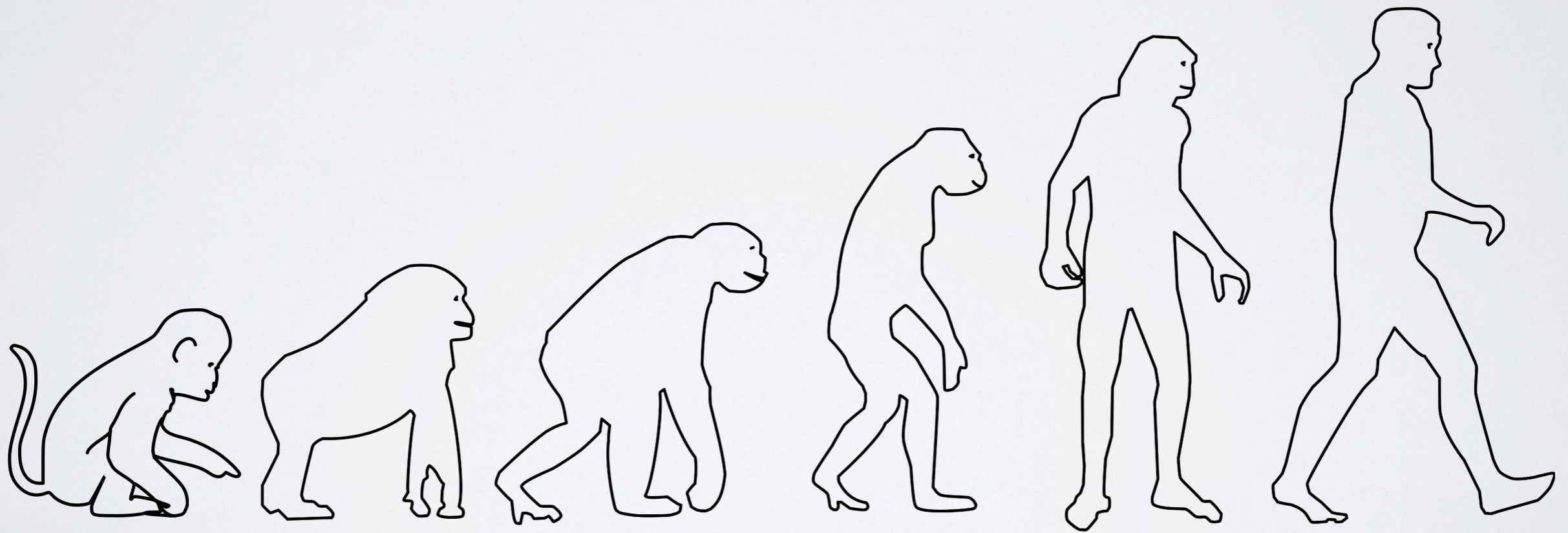
◀ **December 1980** ▶

S	M	T	W	T	F	S
30	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3

Intelligenter Affe



Intelligenter Affe



Türme von Hanoi



© André Karwath aka Aka

Türme von Hanoi

```
1.class TuermeVonHanoi():
2.     def __init__(self):
3.         self.A = [6, 5, 4, 3, 2, 1]
4.         self.B = []
5.         self.C = []
6.
7.     def AtoB(self): self.B.append(self.A.pop())
8.
9.     def AtoC(self): self.C.append(self.A.pop())
10.
11.    def BtoA(self): self.A.append(self.B.pop())
12.
13.    def BtoC(self): self.C.append(self.B.pop())
14.
15.    def CtoA(self): self.A.append(self.C.pop())
16.
17.    def CtoB(self): self.B.append(self.C.pop())
18.
19.    def valid(self):
20.        return all(self.A[i + 1] < self.A[i] for i in range(len(self.A)-1)) and \
21.               all(self.B[m + 1] < self.B[m] for m in range(len(self.B)-1)) and \
22.               all(self.C[n + 1] < self.C[n] for n in range(len(self.C)-1))
```

Türme von Hanoi

Rekursiv

```
1.  def bewege(schritte, i, a_name, b_name, c_name):
2.      if (i > 0):
3.          bewege(schritte, i-1, a_name, c_name, b_name)
4.          schritte.append('tuerme.' + a_name + 'to' + c_name + '()')
5.          bewege(schritte, i-1, b_name, a_name, c_name)
6.      return schritte
7.
8.  def algorithmus():
9.      return bewege([], 6, 'A', 'B', 'C')
```

Türme von Hanoi

Manuell

```
1.  schritte = [\
2.  'tuerme.AtoB()', \ #[654321][][] -> [65432][1][]
3.  'tuerme.AtoC()', \ #[65432][1][] -> [6543][1][2]
4.  'tuerme.BtoC()', \ #[6543][1][2] -> [6543][][21]
5.  'tuerme.AtoB()', \ #[6543][][21] -> [654][3][21]
6.  'tuerme.CtoA()', \ #[654][3][21] -> [6541][3][2]
7.  'tuerme.CtoB()', \ #[6541][3][2] -> [6541][32][]
8.  'tuerme.AtoB()', \ #[6541][32][] -> [654][321][]
9.  'tuerme.AtoC()', \ #[654][321][] -> [65][321][4]
10. 'tuerme.BtoC()', \ #[65][321][4] -> [65][32][41]
11. 'tuerme.BtoA()', \ #[65][32][41] -> [652][3][41]
12. 'tuerme.CtoA()', \ #[652][3][41] -> [6521][3][4]
13. 'tuerme.BtoC()', \ #[6521][3][4] -> [6521][][43]
14. 'tuerme.AtoB()', \ #[6521][][43] -> [652][1][43]
15. 'tuerme.AtoC()', \ #[652][1][43] -> [65][1][432]
16. 'tuerme.BtoC()', \ #[65][1][432] -> [65][][4321]
17. 'tuerme.AtoB()', \ #[65][][4321] -> [6][5][4321]
18. 'tuerme.CtoA()', \ #[6][5][4321] -> [61][5][432]
```


Türme von Hanoi Möglichkeiten

$$\geq 2^{63}$$

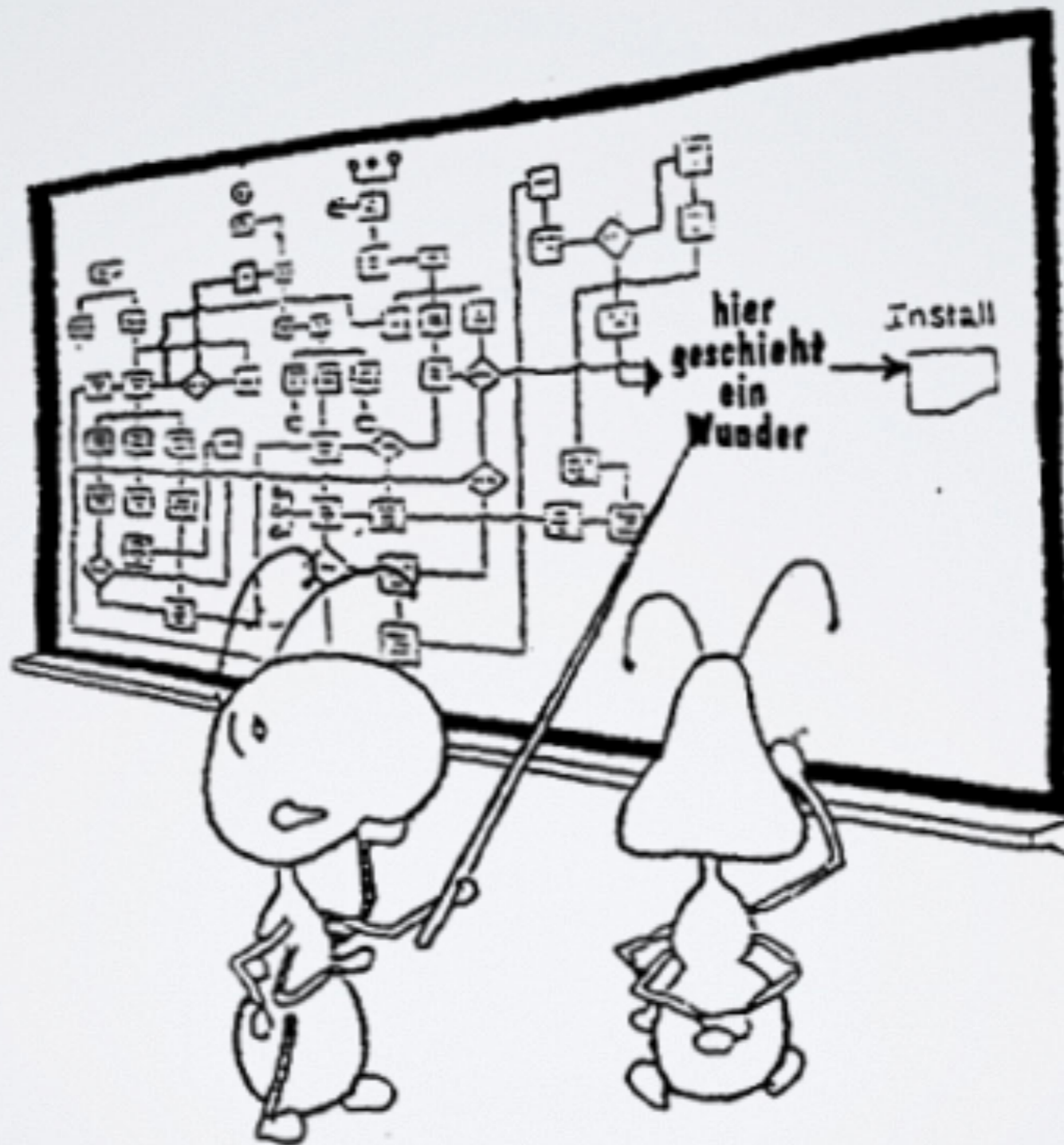
Türme von Hanoi Möglichkeiten

9.223.372.036.854.775.808

Türme von Hanoi Möglichkeiten

292.471.208 Jahre

Türme von Hanoi Automatisch?



Türme von Hanoi

Zielfunktion

```
1. def min_zielfunktion(tuerme): return (21 - sum(tuerme.C))
```

Türme von Hanoi

Stochastischer Bergsteigeralgorithmus

```
while (min_zielfunktion(loesung) > 0)
  neue_loesung = addNextStep(loesung)
  if (better(neue_loesung, loesung))
    loesung = neue_loesung
```

Türme von Hanoi

Stochastischer Bergsteigeralgorithmus

DEMO

Türme von Hanoi

Stochastischer Bergsteigeralgorithmus

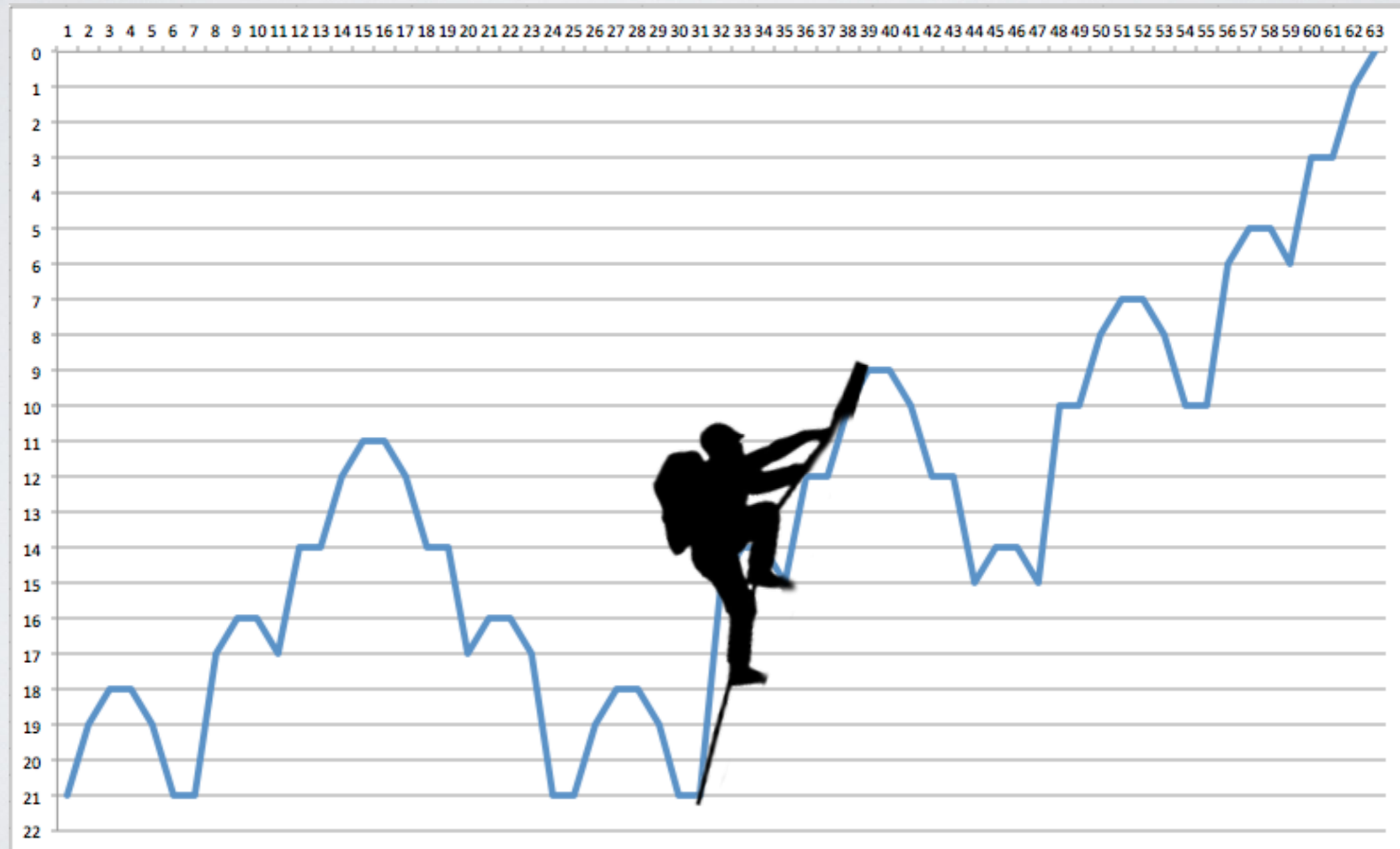
`['tuerme.AtoC()'] -> [65432][][1], Anzahl Schritte: 1, Zielwert: 20`

Türme von Hanoi

Zufalls-Neustart Stochastischer Bergsteigeralgorithmus

```
nach_1000_versuchen_neustarten:  
  while (min_zielfunktion(loesung) > 0)  
    neue_loesung = addNextStep(loesung)  
    if (better(neue_loesung, loesung))  
      loesung = neue_loesung
```

Türme von Hanoi Suchraumlanschaft



Türme von Hanoi

Zufallsbewegung

```
while (min_zielfunktion(loesung) > 0)
  neue_loesung = addNextStep(loesung)
  if (better(neue_loesung, loesung)
      or 50/50_chance)
    loesung = neue_loesung
```

Türme von Hanoi Zufallsbewegung

DEMO

Türme von Hanoi Zufallsbewegung

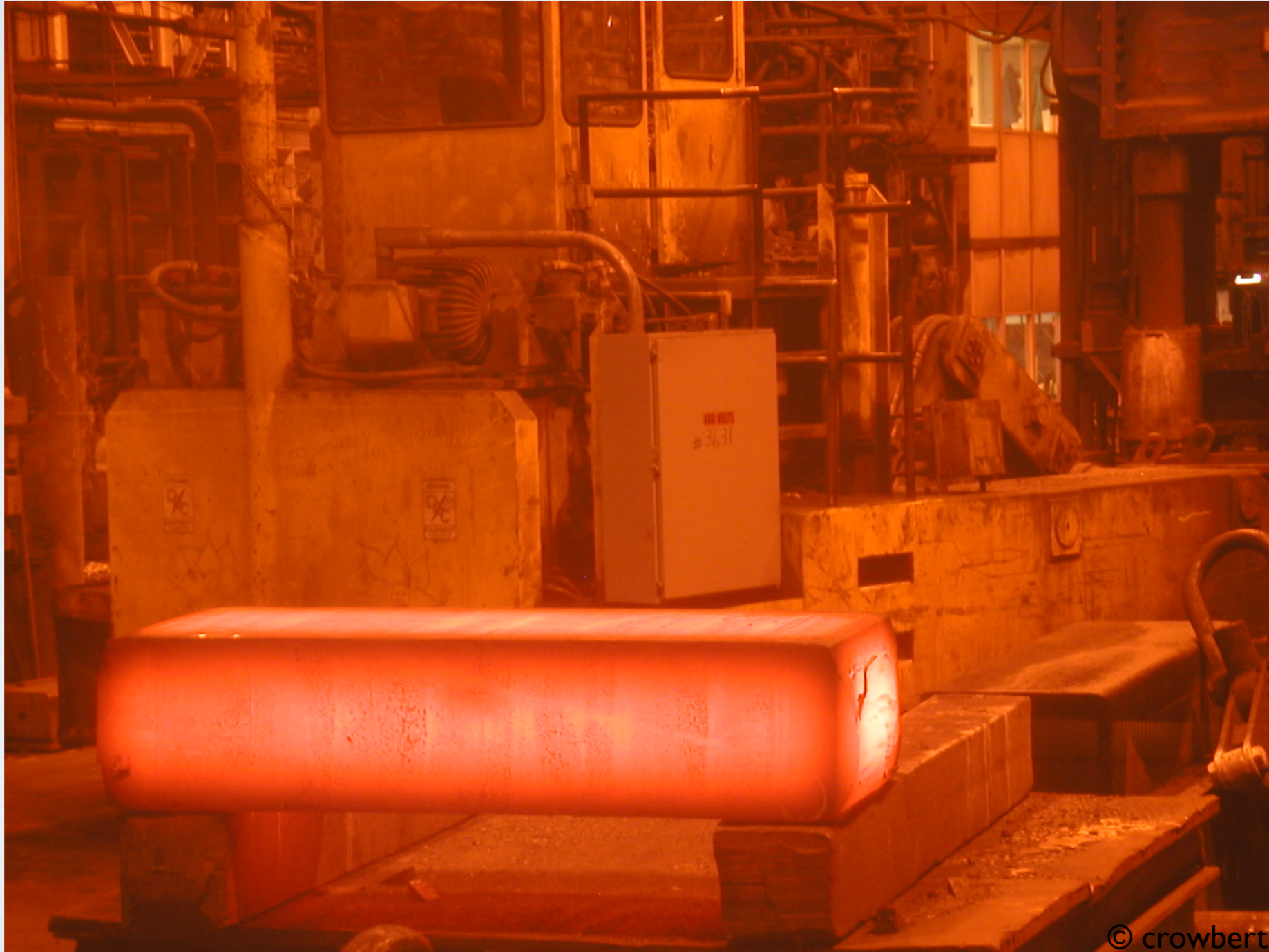
```
[tuerme.AtoB(), tuerme.BtoC(), tuerme.CtoB(), tuerme.BtoA(),  
tuerme.AtoB(), ...] -> [51][][6432], Anzahl Schritte: 2018, Zielwert: 6
```

Türme von Hanoi

Simulierte Abkühlung

```
while (min_zielfunktion(loesung) > 0)
  neue_loesung = addNextStep(loesung)
  if (better(neue_loesung, loesung)
      or decreasing_chance)
    loesung = neue_loesung
```

Türme von Hanoi Simulierte Abkühlung



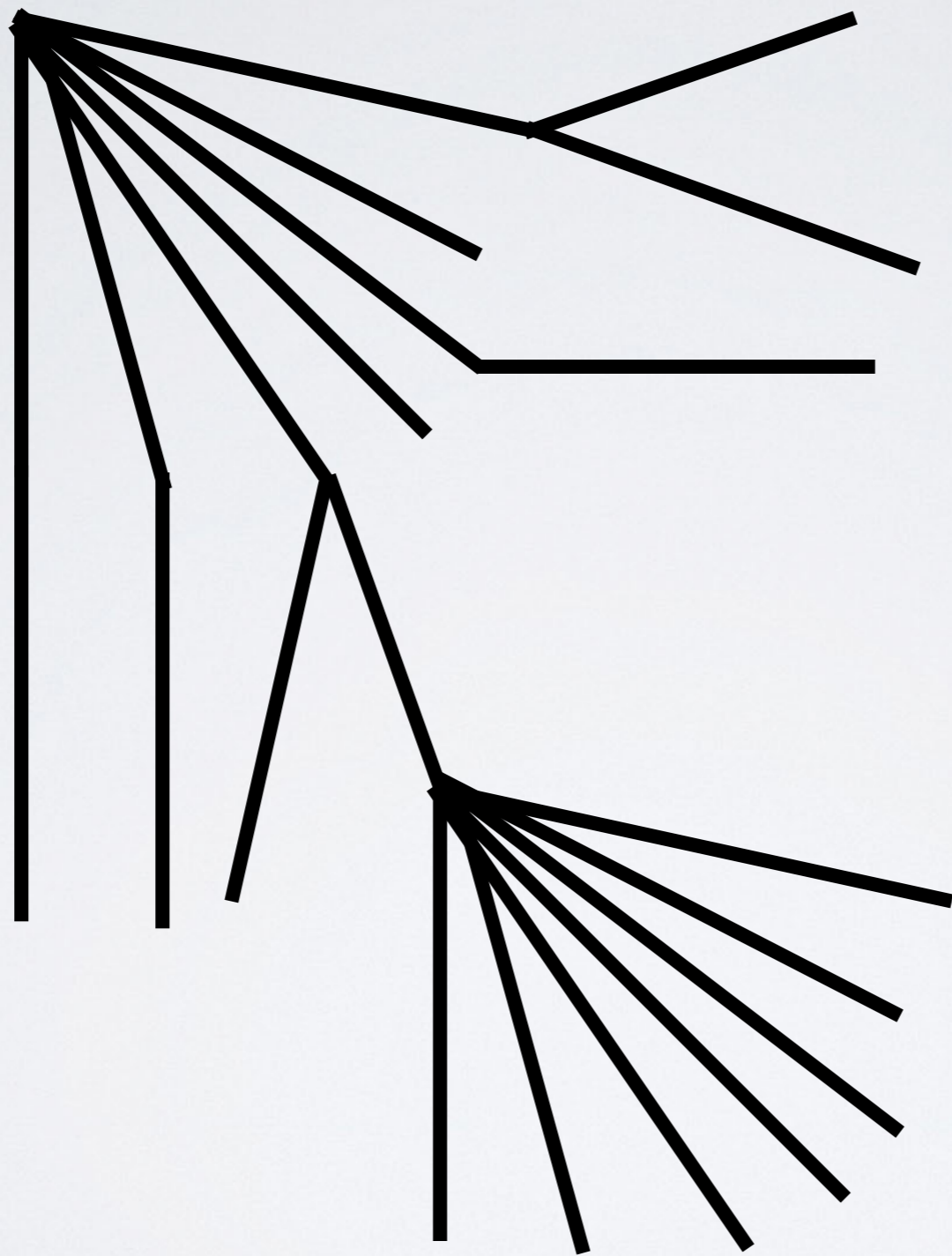
© crowbert

Mehrere Suchen gleichzeitig?

Türme von Hanoi

Balkensuche

```
loesungen = []
beste = get_beste(loesungen)
while (min_zielfunktion(beste) > 0)
    neue_loesungen = addNextSteps(loesungen)
    loesungen = getXBest(neue_loesungen)
    beste = get_beste(loesungen)
```

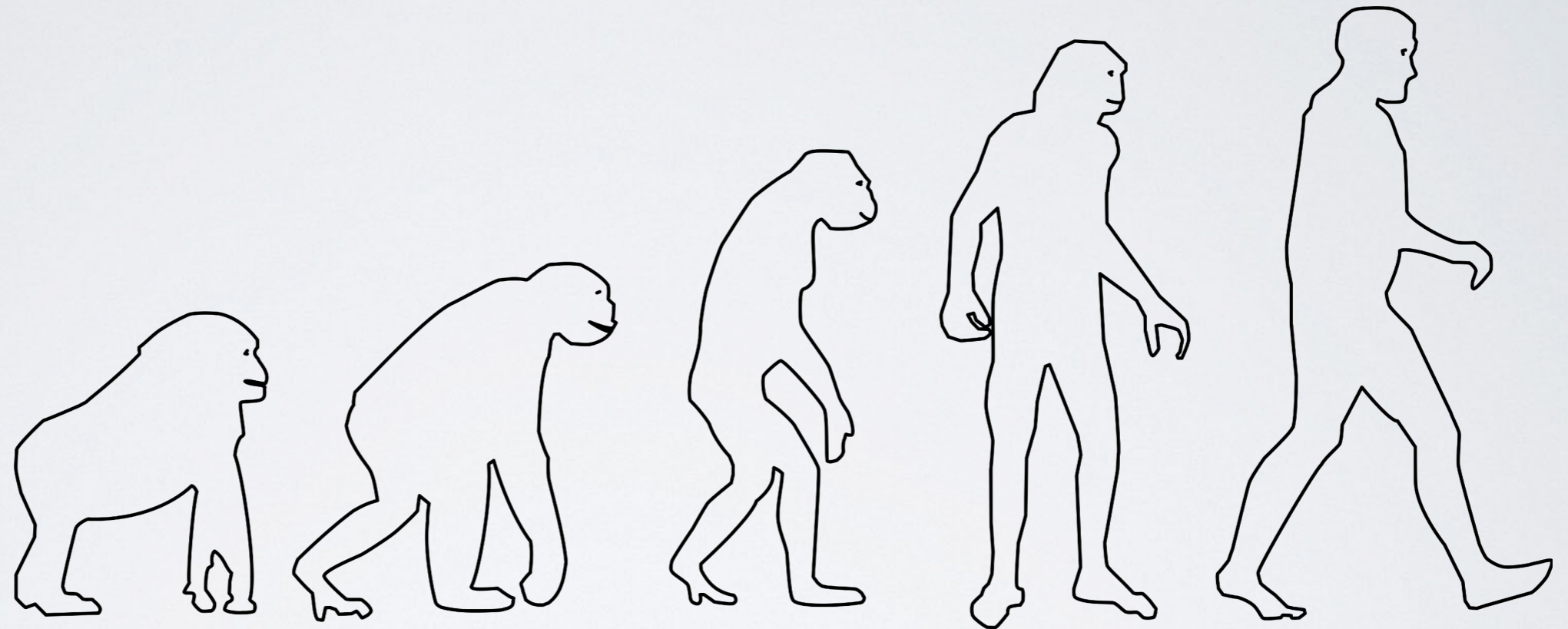


Türme von Hanoi

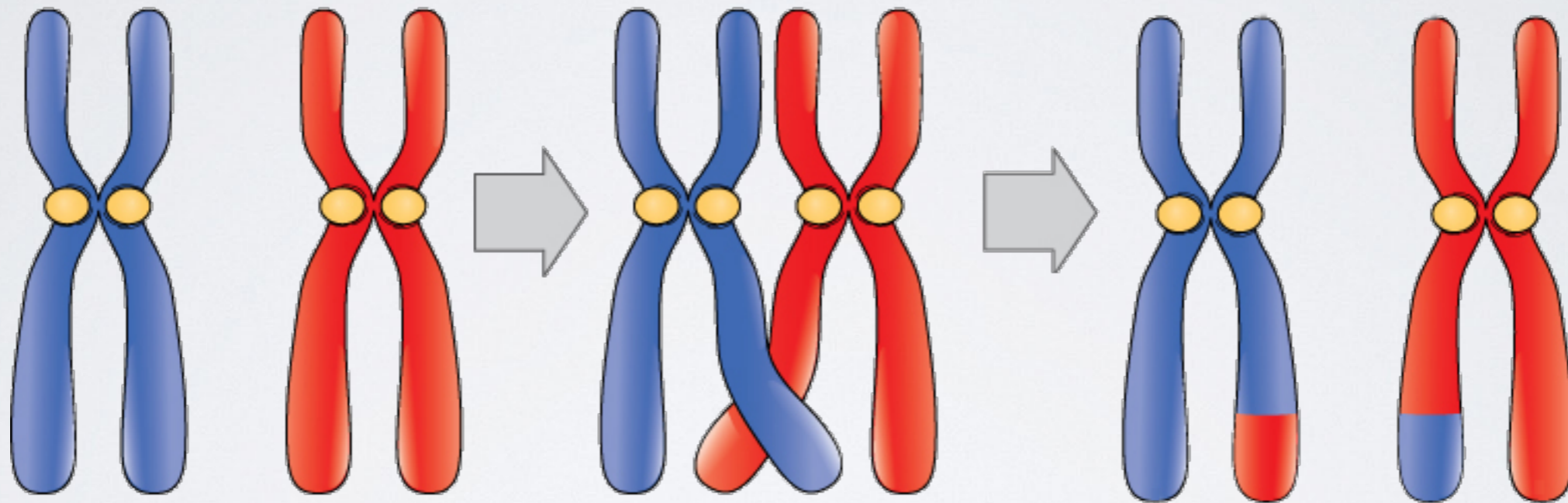
Balkensuche

```
loesungen = []
beste = get_beste(loesungen)
while (min_zielfunktion(beste) > 0)
    neue_loesungen = addNextSteps(loesungen)
    loesungen = getXBest(neue_loesungen)
    beste = get_beste(loesungen)
```

Türme von Hanoi Genetischer Algorithmus

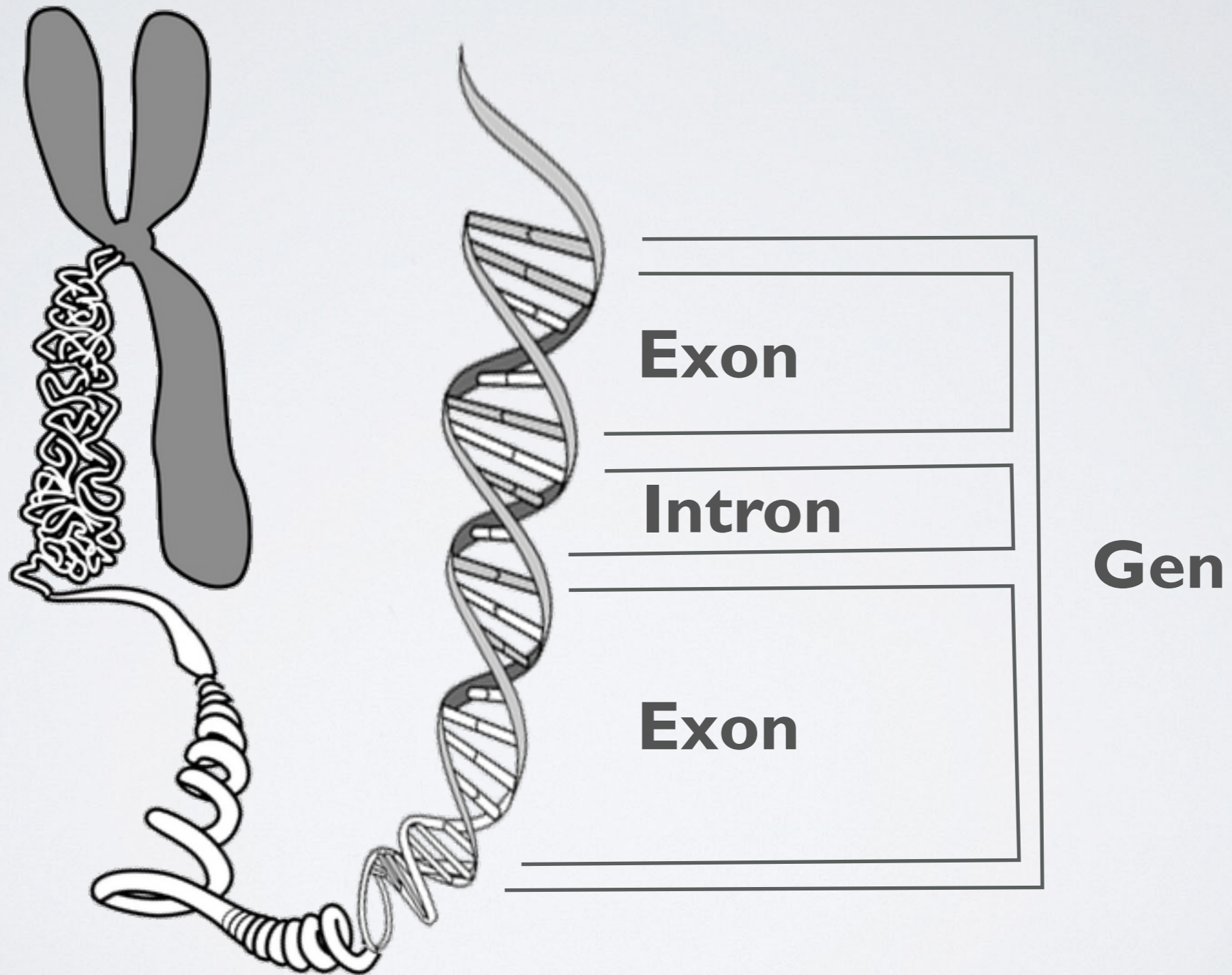


Türme von Hanoi Genetischer Algorithmus



© Robert Bear and David Rintoul

Türme von Hanoi Genetischer Algorithmus



Türme von Hanoi

Genetischer Algorithmus

```
loesungen = []
beste = get_beste(loesungen)
while (min_zielfunktion(beste) > 0)
    neue_loesungen = crossover(loesungen)
    neue_loesungen =
        addNextSteps(neue_loesungen)
    loesungen = getXBest(neue_loesungen)
    beste = get_beste(loesungen)
```

Türme von Hanoi

Genetischer Algorithmus

DEMO

Türme von Hanoi

Genetischer Algorithmus

```
[tuerme.AtoB(), tuerme.AtoC(), tuerme.BtoA(), tuerme.AtoB(), tuerme.BtoA(), ...] ->  
[][][654321], Anzahl Schritte: 1727, Zielwert: 0
```

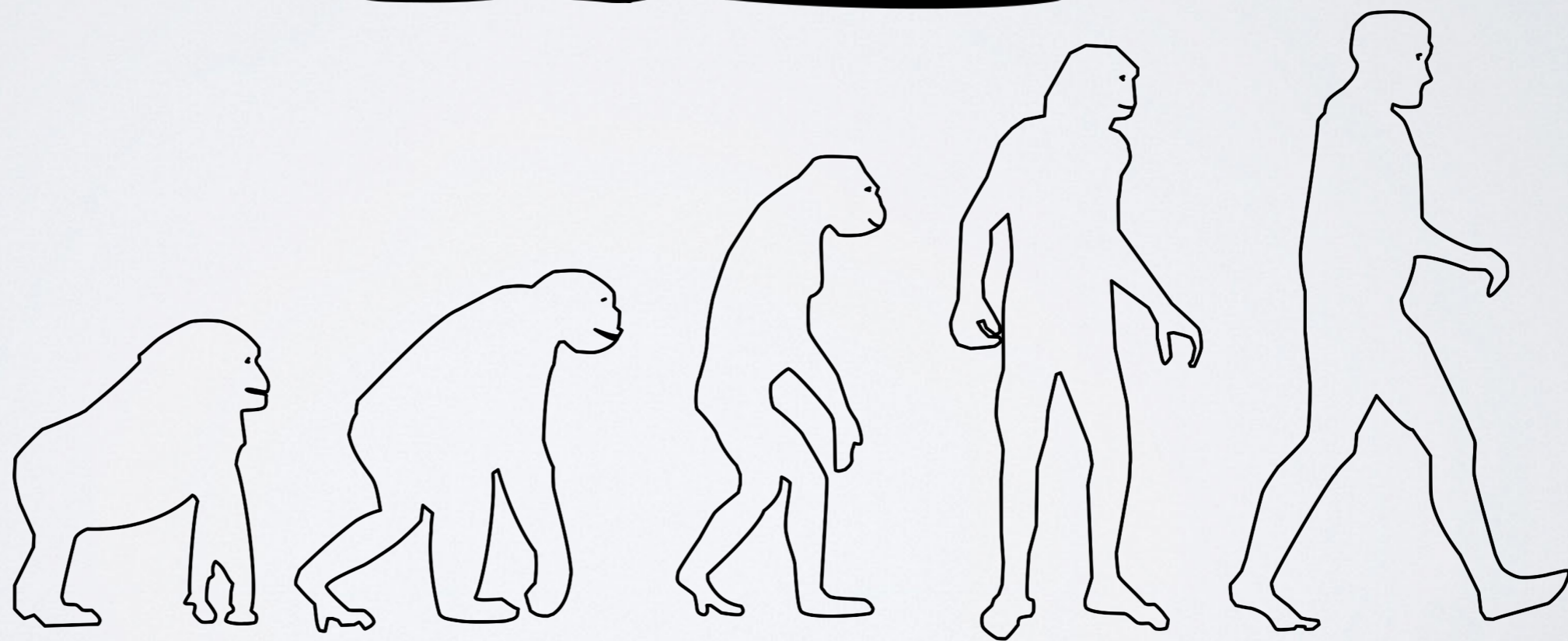
Genetischer Algorithmus

Effizienz

66 Sekunden

statt

150 Mio. Jahre



Welche Fehler kann der Affe finden?



When is a bug not a bug?



When it's a feature!

Is it a bug?

```
1. def auth(username, password):
2.     if username == 'admin' and password == 'geheim':
3.         return True
4.     if hash(password + get_salt(username)) == get_pwd_hash(username):
5.         return True
6.     return False
```

Is it a bug?

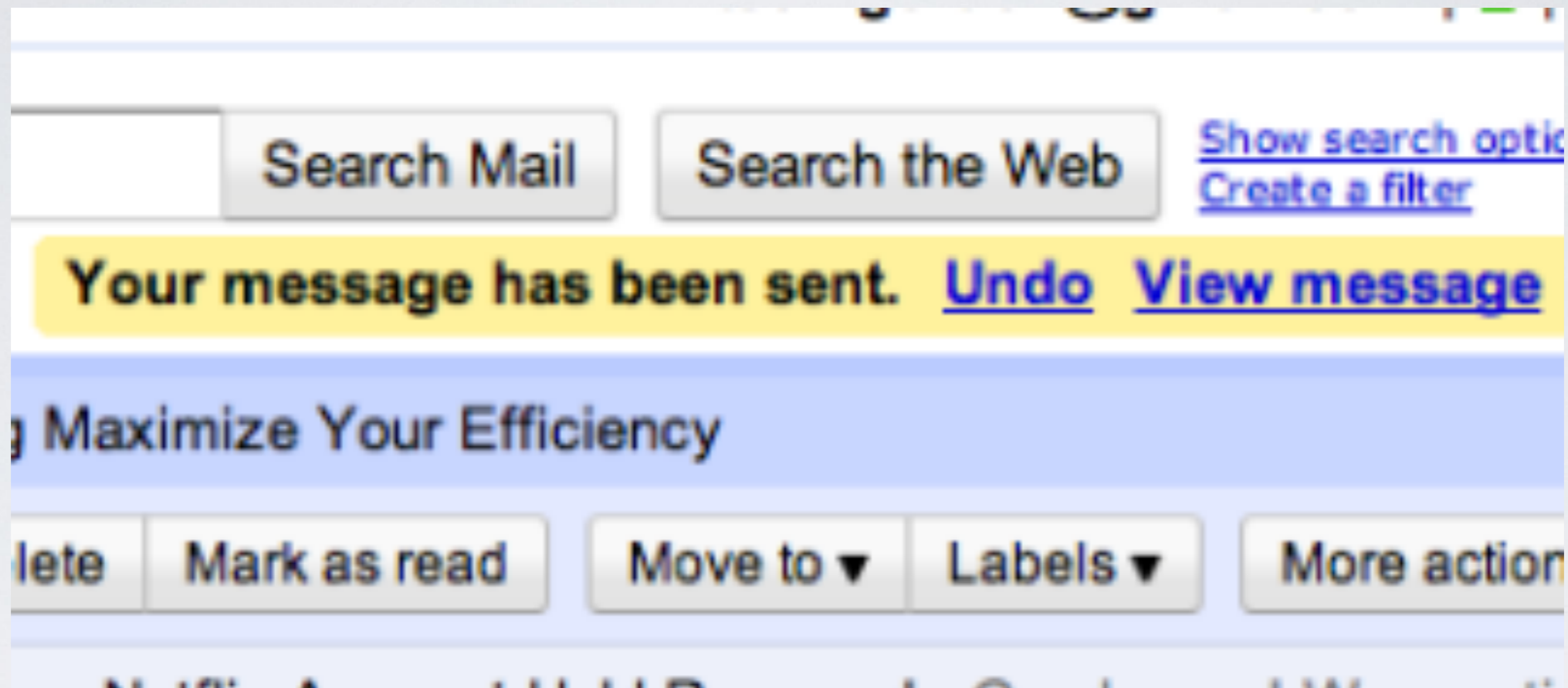
```
.  
..  
text.c  
other.files
```

```
1. if (name[0] == '.') continue;
```

Is it a bug?



Is it a bug?



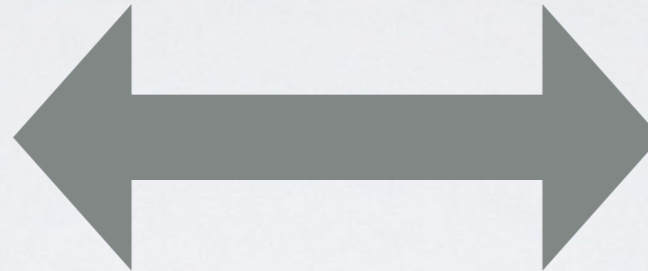
What is a bug?

“ Without specification, there are no bugs
— only surprises.

Brian Kernighan

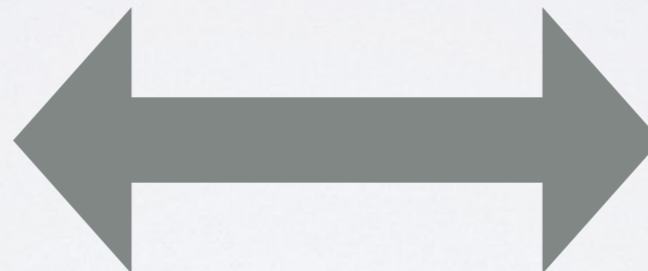
What is a bug?

Spezifikation



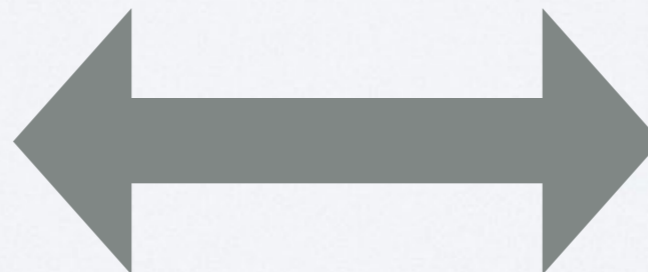
Code

Modell



Code

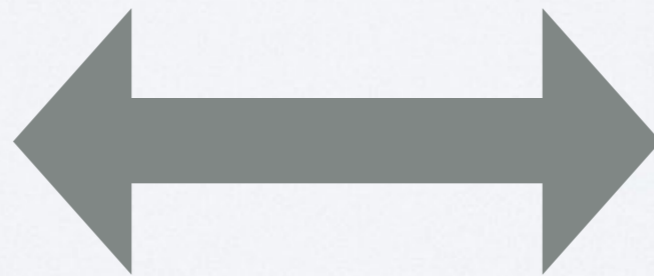
Nutzererwartung



Code

Monkey Testing

Nutzererwartung



Code

Wozu testen wir?



**Nach
Implementierung?**

TableSelectionDemo

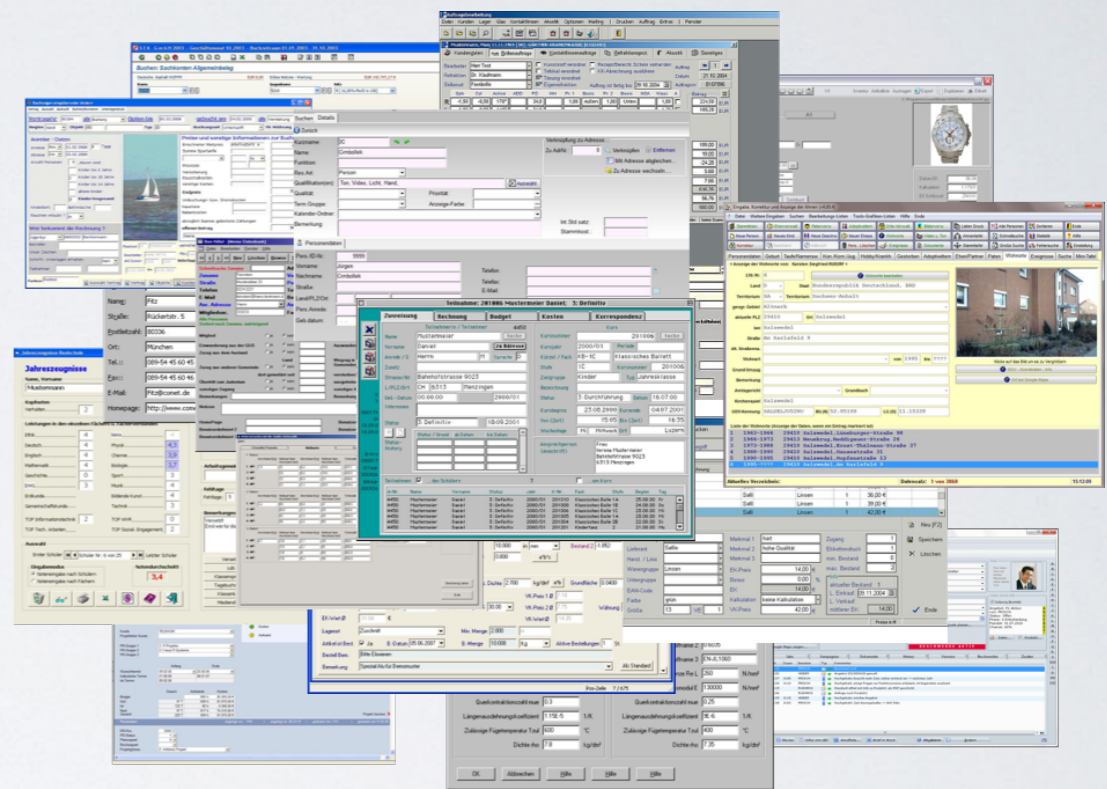
First Name	Last Name	Sport	# of Years	Vegetarian
Kathy	Smith	Snowboarding	5	<input type="checkbox"/>
John	Doe	Rowing	3	<input checked="" type="checkbox"/>
Sue	Black	Knitting	2	<input type="checkbox"/>
Jane	White	Speed reading	20	<input checked="" type="checkbox"/>

Selection Mode

- Multiple Interval Selection
- Single Selection
- Single Interval Selection

Selection Options

- Row Selection
- Column Selection
- Cell Selection



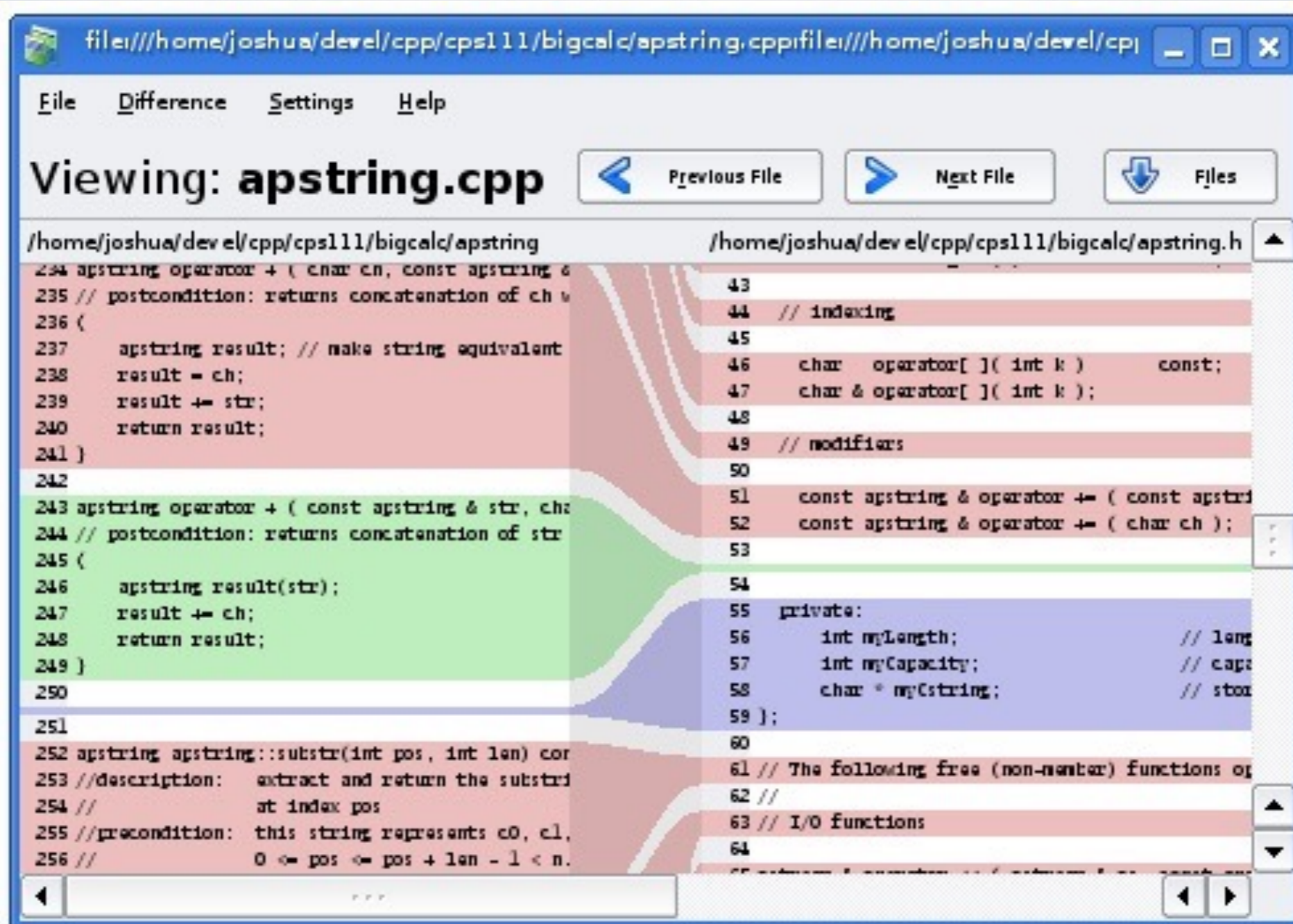
Wozu testen wir?



**Nach
Implementierung?**



**Nach
Änderung?**



**Wenn man nur einen Hammer hat
sieht alles aus wie ein Nagel...**

```
public void testSomething() throws Exception {
    // here be some test code ...
    UIElement element = driver.findElement("path/to/element");
    assertEquals(null, element.getAction());
    assertEquals("action", element.getActionCommand());
    assertEquals(true, element.isEnabled());
    assertEquals(true, element.isFocusable());
    assertEquals("Lucida Grande", element.getFont().getName());
    assertEquals(13, element.getFont().getSize());
    assertEquals(0, element.getFont().getStyle());
    assertEquals(23, element.getHeight());
    assertEquals(null, element.getIcon());
    assertEquals(0, element.getMnemonic());
    assertEquals(null, element.getPressedIcon());
    assertEquals(true, element.isSelected());
    assertEquals("label", element.getText());
    assertEquals(119, element.getWidth());
    assertEquals(27, element.getX());
    assertEquals(191, element.getY());
}
```



Ihr Vollautomatischer Regressionstester.

Weniger Risiko, weniger Stress, weniger Kosten!

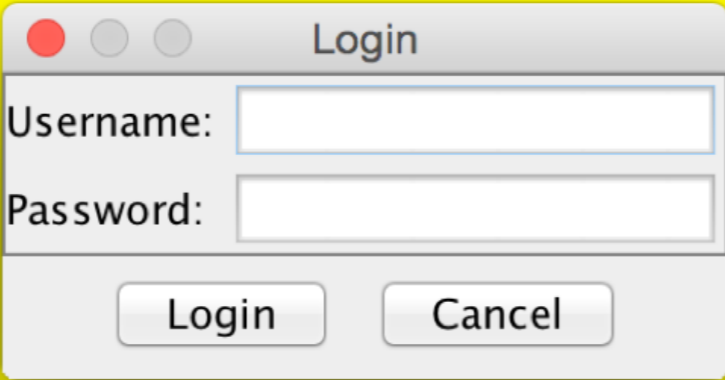
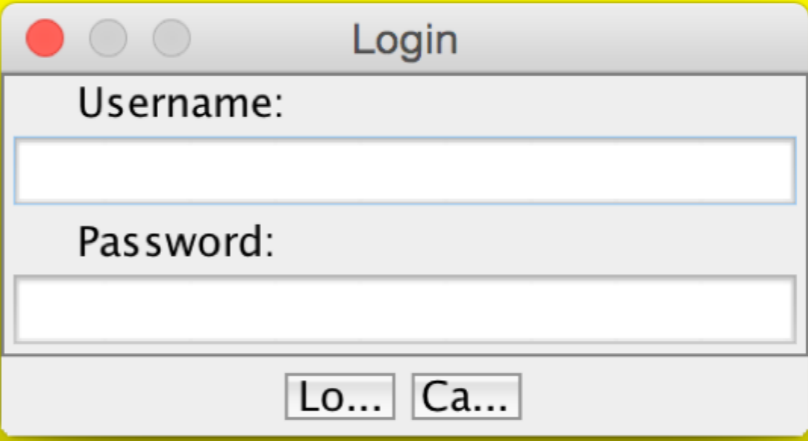
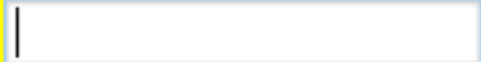
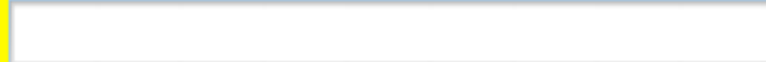
Haben Sie genug vom Testen?

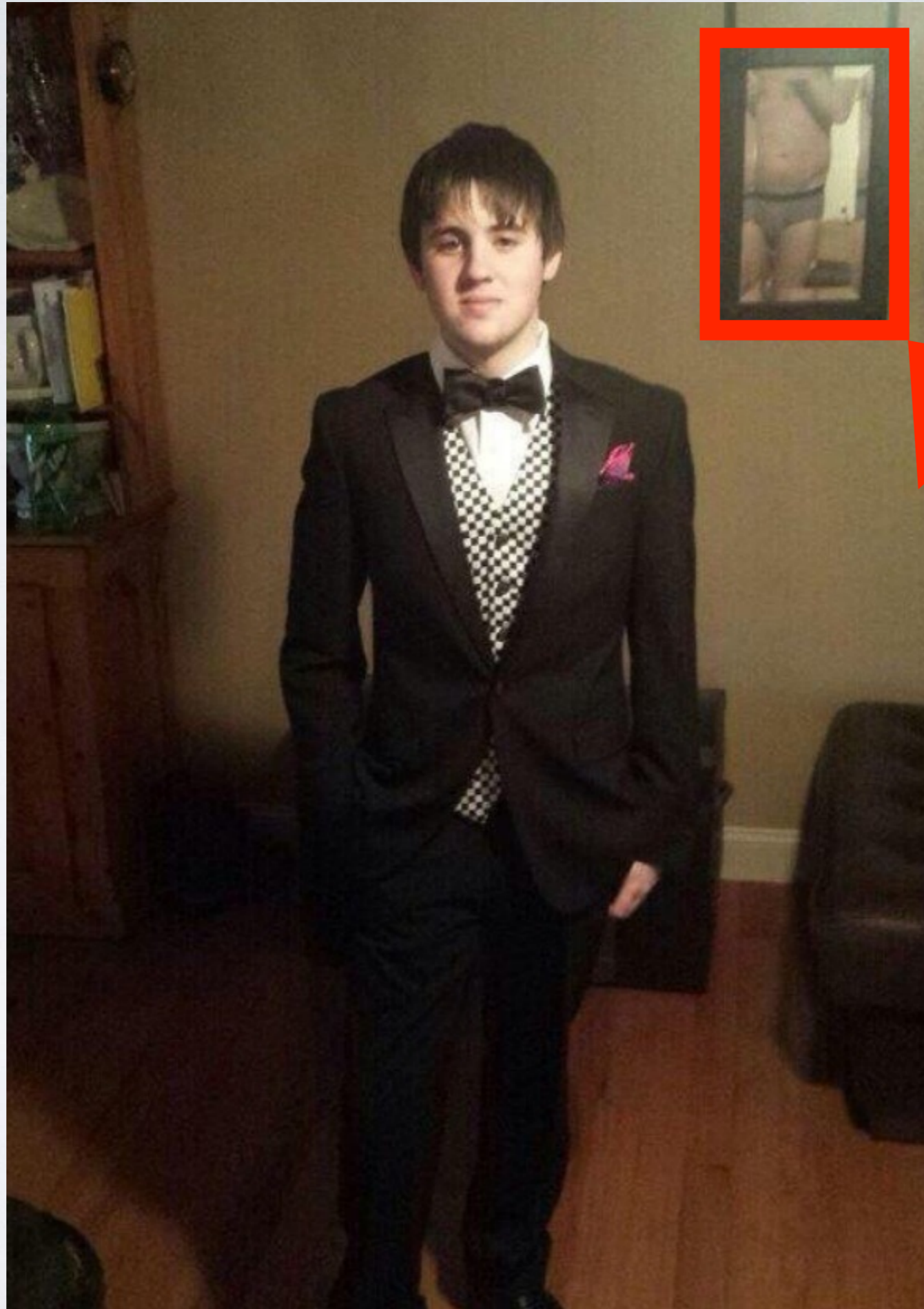
Sowohl manuelles Testen, als auch das manuelle Erstellen und Pflegen von automatischen Tests sind unbeliebte Zeitfresser. Direkt aus der Forschung ist ReTest als bisher einziges Produkt seiner Art Ihr



Haben Sie genug vom

Testen?

<input type="checkbox"/> LoginDialog		
<input type="checkbox"/> Label Username	<p style="text-align: center;">Username:</p> <p>x-Coordinate: 0 y-Coordinate: 5</p>	<p style="text-align: center;">Username:</p> <p>x-Coordinate: 24 y-Coordinate: 0</p>
<input type="checkbox"/> Field Username	 <p>x-Coordinate: 71 y-Coordinate: 1 Width: 254px</p>	 <p>x-Coordinate: 1 y-Coordinate: 23 Width: 543px</p>
<input type="checkbox"/> Label Password	<p style="text-align: center;">Password:</p> <p>x-Coordinate: 0 y-Coordinate: 44</p>	<p style="text-align: center;">Password:</p> <p>x-Coordinate: 24 y-Coordinate: 48</p>



Ausführung Wiederholbar?

**Separate Testumgebung
mit voller Kontrolle**



Monkey Testing

Coverage Maximization

Saving GUI Results

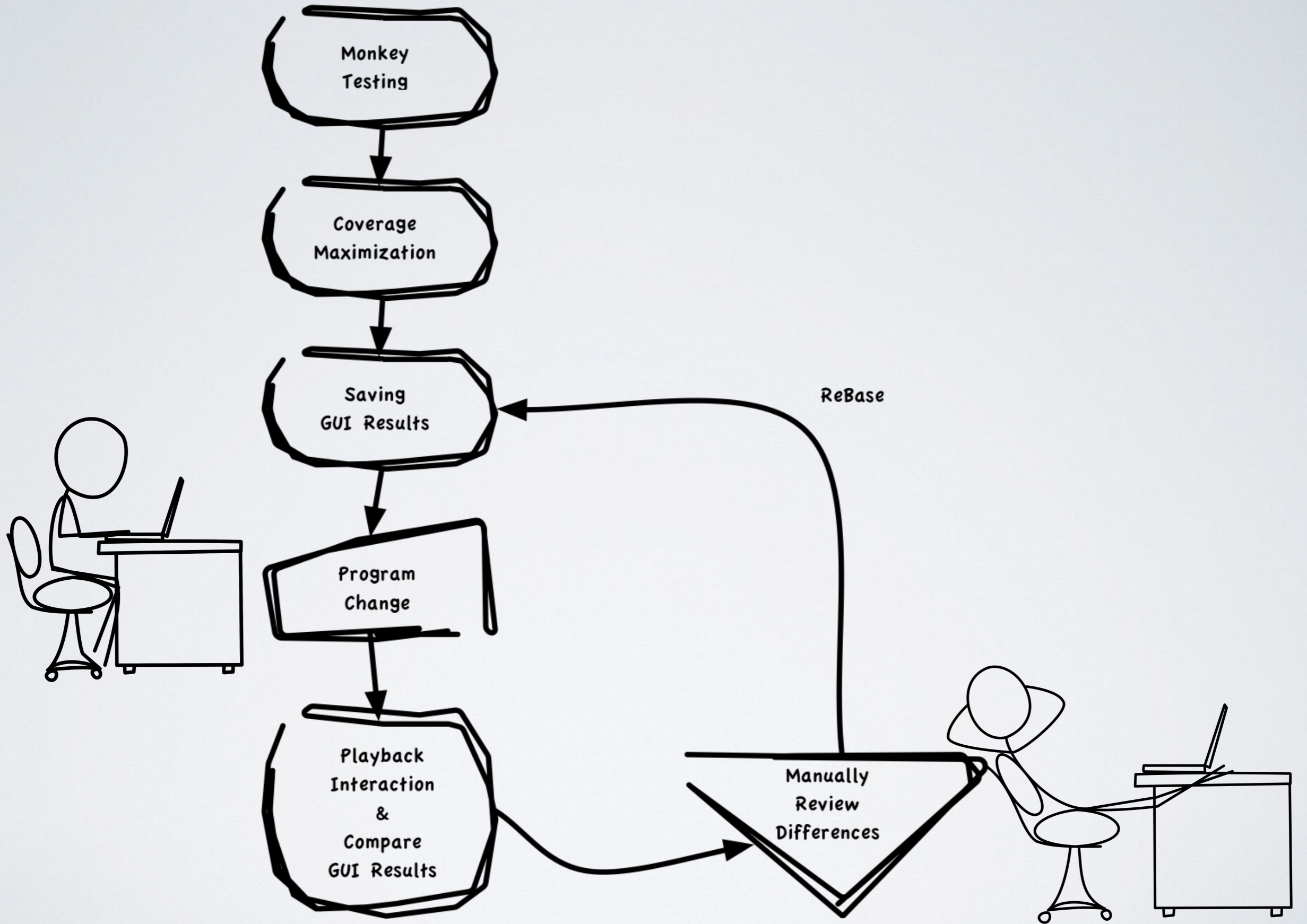
Program Change

Playback Interaction & Compare GUI Results

Manually Review Differences

Repeat







ReTest



Aktionssfolge aufzeichnen



Starte Aufzeichnung



Starte Anwendung

Aufgezeichnete Aktionsfolge

Aktion

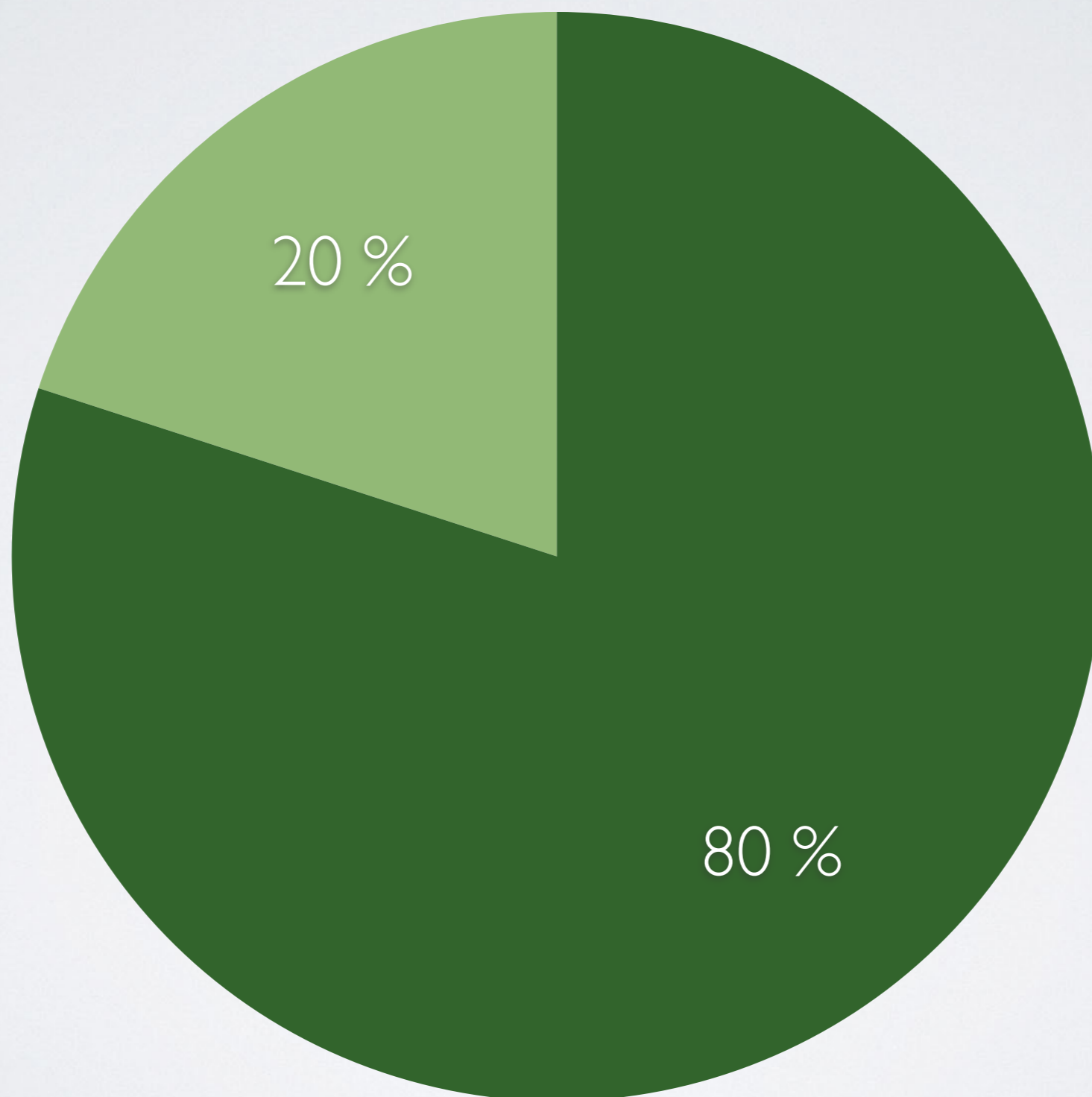
Export Test

Neu...

Speichern

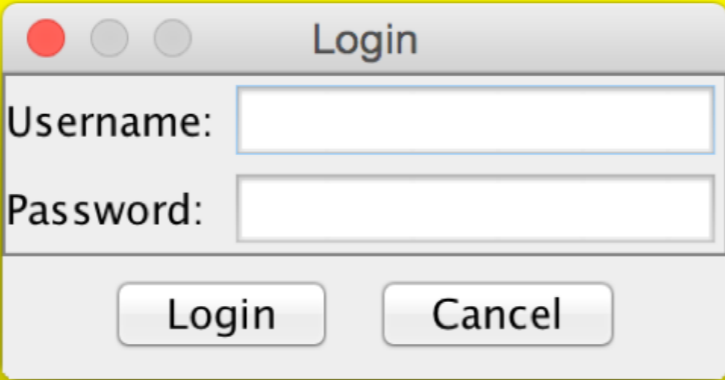
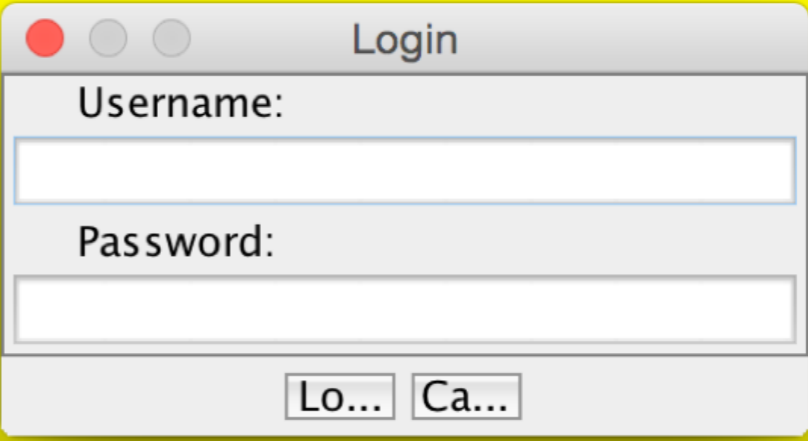
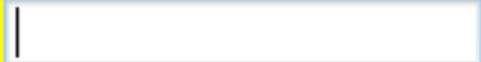
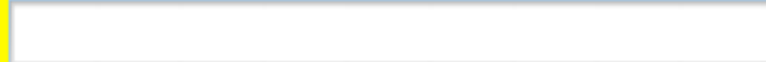
Bereit. Drücken Sie auf 'Starte Aufzeichnung'.

GENETISCHER ALGORITHMUS



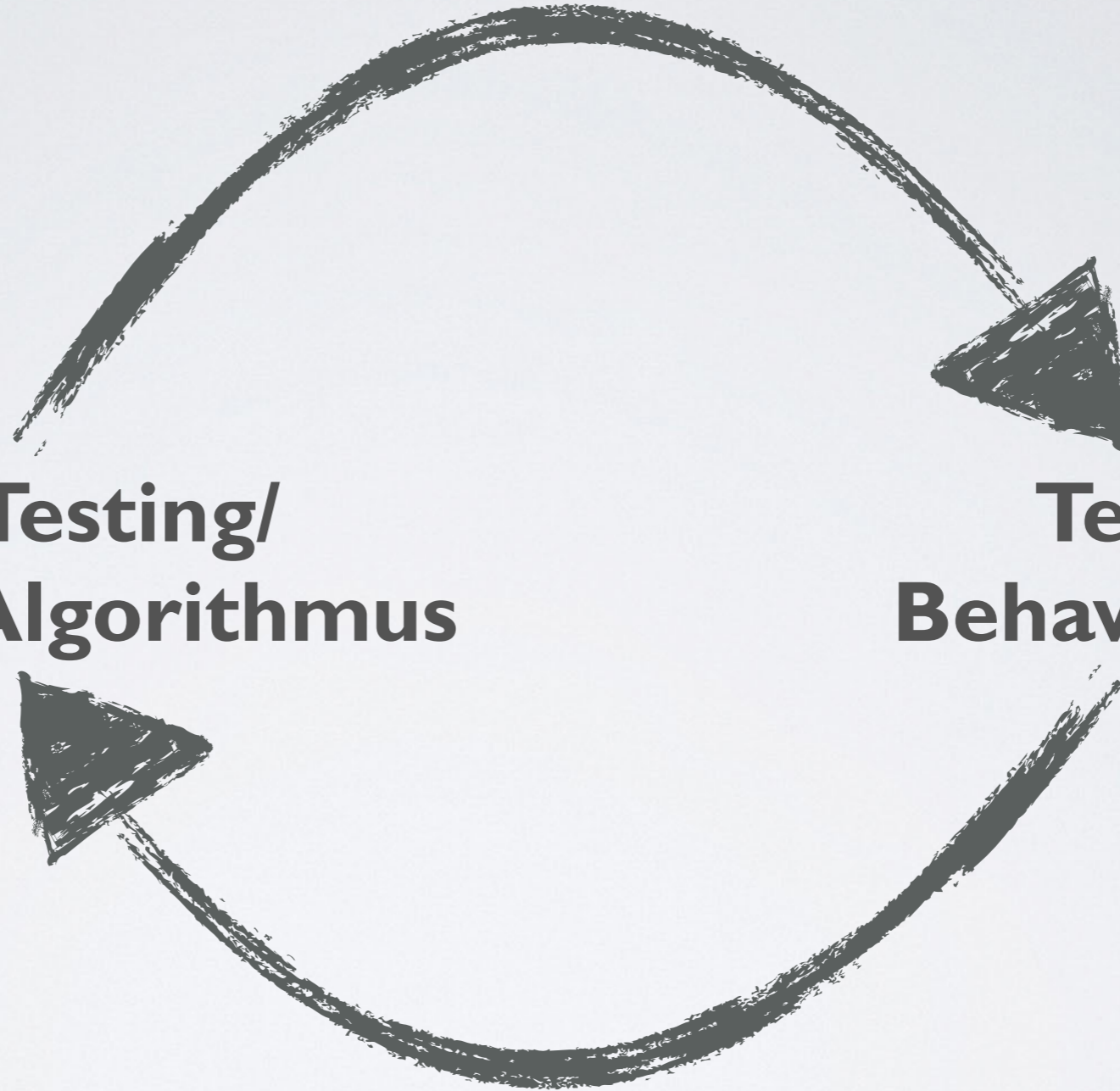
Coverage

festes Set an Tests

<input type="checkbox"/> LoginDialog		
<input type="checkbox"/> Label Username	<p style="text-align: center;">Username:</p> <p>x-Coordinate: 0 y-Coordinate: 5</p>	<p style="text-align: center;">Username:</p> <p>x-Coordinate: 24 y-Coordinate: 0</p>
<input type="checkbox"/> Field Username	 <p>x-Coordinate: 71 y-Coordinate: 1 Width: 254px</p>	 <p>x-Coordinate: 1 y-Coordinate: 23 Width: 543px</p>
<input type="checkbox"/> Label Password	<p style="text-align: center;">Password:</p> <p>x-Coordinate: 0 y-Coordinate: 44</p>	<p style="text-align: center;">Password:</p> <p>x-Coordinate: 24 y-Coordinate: 48</p>

**Monkey-Testing/
Genetischer Algorithmus**

**Testing/
Behavioral Diff**



Summary

Monkey-Testing:

billiger

schneller

besser

multipliziert manuelle Tests

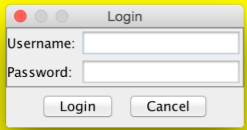
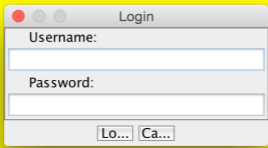
Behavioral Diff

kein Pflegeaufwand

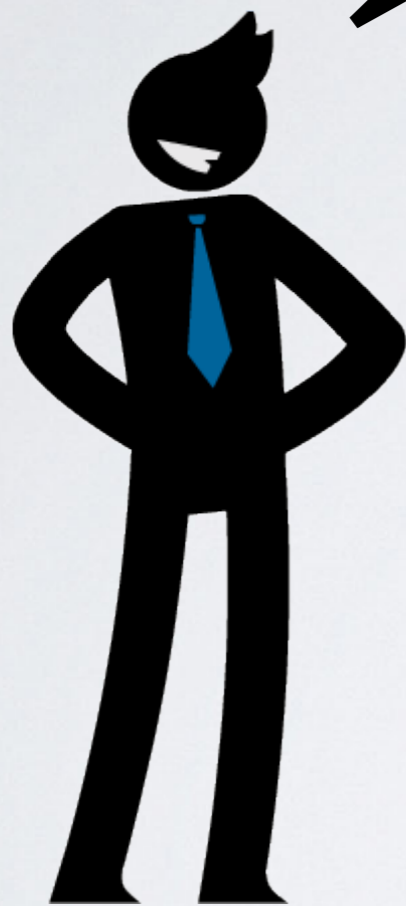
kein wasteful testing

trotzdem “vollständiges“ Testen



<input type="checkbox"/> LoginDialog		
<input type="checkbox"/> Label Username	Username: x-Coordinate: 0 y-Coordinate: 5	Username: x-Coordinate: 24 y-Coordinate: 0
<input type="checkbox"/> Field Username	<input type="text"/> x-Coordinate: 71 y-Coordinate: 1 Width: 254px	<input type="text"/> x-Coordinate: 1 y-Coordinate: 23 Width: 543px
<input type="checkbox"/> Label Password	Password: x-Coordinate: 0 y-Coordinate: 44	Password: x-Coordinate: 24 y-Coordinate: 48

Machen Sie den Test!



 ReTest

The logo for ReTest features a green icon on the left consisting of a curved line with several vertical bars of varying heights, resembling a bar chart or a stylized 'R'. To the right of this icon, the word 'ReTest' is written in a sans-serif font. The 'Re' is in green, and 'Test' is in dark blue.

www.retest.de