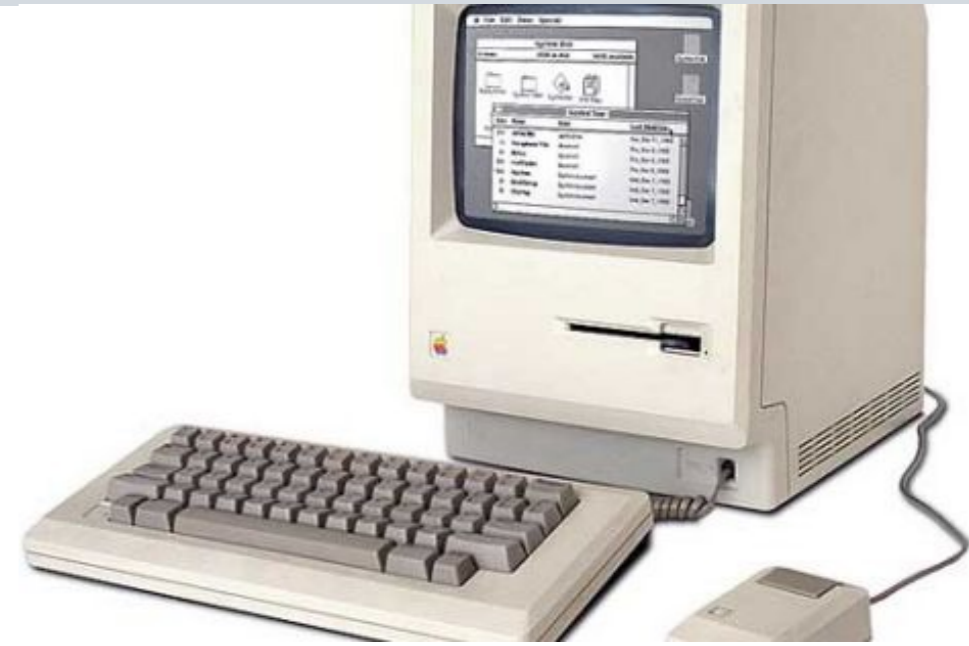


NEXT LEVEL VERSION CONTROL:

NOW IN 3D!

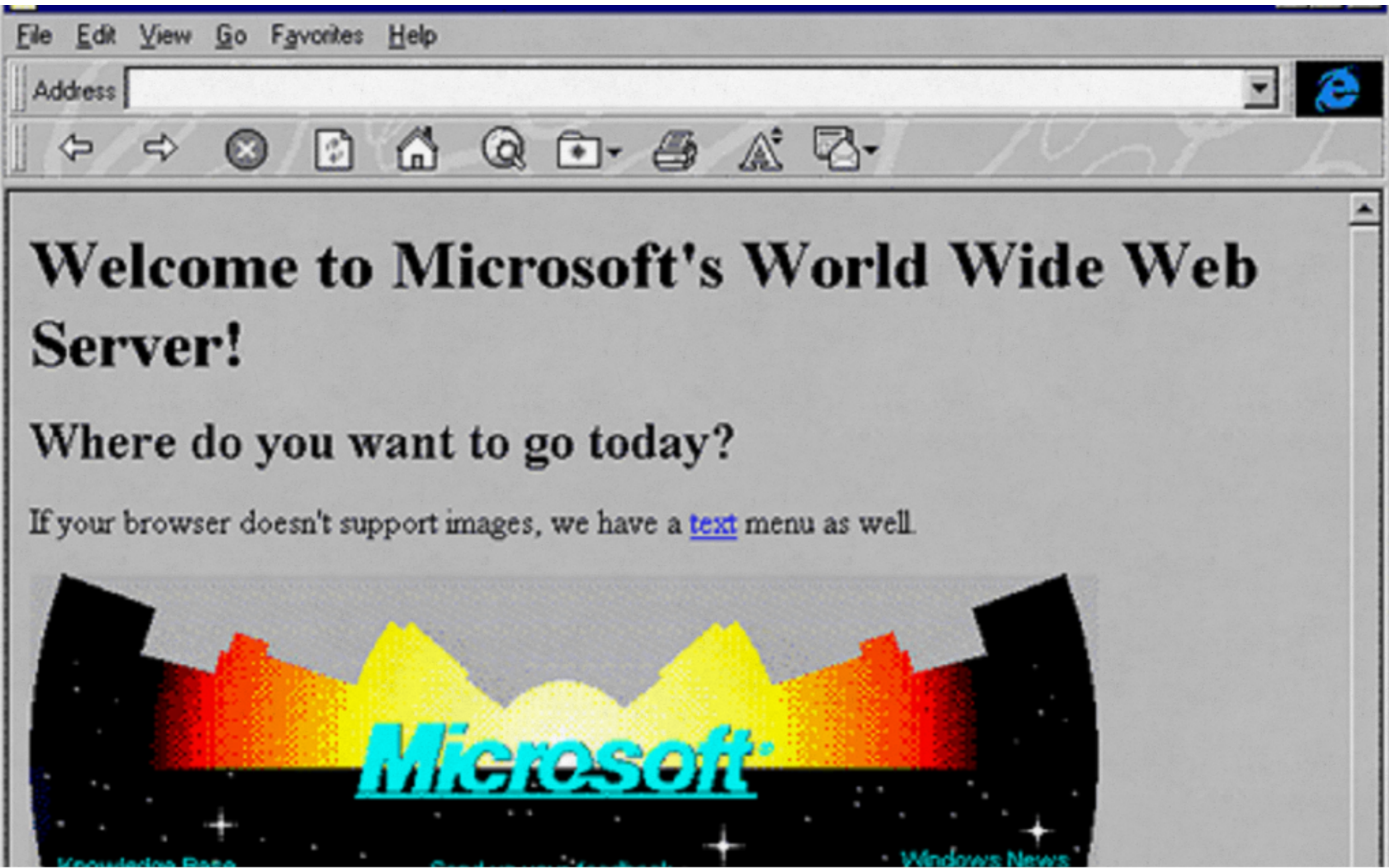
Dr. Jeremias Rößler

Wer hat vor
1990
Software entwickelt?



Erinnern Sie sich?






```
public client static void reFreshCallerDataSource(Object _callerFormDataSource,
boolean _recordExists,
boolean _doReSearch)
{
if (_recordExists)
{
if (_doReSearch)
{
callerFormDataSource.reSearch(true);
}
}
callerFormDataSource.reRead();
callerFormDataSource.reRefresh();
if (callerFormDataSourceHasMethod(_callerFormDataSource, identifierStr(reReadLines)))
{
callerFormDataSource.reReadLines();
}
callerFormDataSource.active();
}
}
```



err.c



etharp.c



init.c



ipv4



ipv6



mem.c



memp.c



netbuf.c



netdb.c



netif.c



netifapi.c



pbuf.c



raw.c



snmp



sockets.c

Gott seit Dank: CVS



err.c



etharp.c



init.c



ipv4



ipv6



mem.c



memp.c



netbuf.c



netdb.c



netif.c



netifapi.c



pbuf.c



raw.c



snmp



sockets.c

file:///home/joshua/dev el/cpp/cps111/bigcalc/apstring.cpp | file:///home/joshua/dev el/cpp/

File Difference Settings Help

Viewing: **apstring.cpp** Previous File Next File Files

/home/joshua/dev el/cpp/cps111/bigcalc/apstring	/home/joshua/dev el/cpp/cps111/bigcalc/apstring.h
234 apstring operator + (char ch, const apstring &	43
235 // postcondition: returns concatenation of ch &	44 // indexing
236 {	45
237 apstring result; // make string equivalent	46 char operator[](int k) const;
238 result = ch;	47 char & operator[](int k);
239 result += str;	48
240 return result;	49 // modifiers
241 }	50
242	51 const apstring & operator += (const apstring
243 apstring operator + (const apstring & str, char	52 const apstring & operator += (char ch);
244 // postcondition: returns concatenation of str	53
245 {	54
246 apstring result(str);	55 private:
247 result += ch;	56 int myLength; // leng
248 return result;	57 int myCapacity; // capa
249 }	58 char * myCString; // stor
250	59 };
251	60
252 apstring apstring::substr(int pos, int len) con	61 // The following free (non-member) functions of
253 //description: extract and return the substr	62 //
254 // at index pos	63 // I/O functions
255 //precondition: this string represents c0, c1,	64
256 // 0 <= pos <= pos + len - 1 < n.	

Lass uns mal die Version *1.234* auschecken...

Gott seit Dank: CVS?

Gott seit Dank: SVN

Quiz:
**Wer schreibt auf Anhieb
fehlerfreien Code?**

Gott seit Dank: SVN?

Git seit Dank: DVCS

file:///home/joshua/develop/cpp/cps111/bigcalc/apstring.cpp | file:///home/joshua/develop/cpp/

File Difference Settings Help

Viewing: **apstring.cpp** Previous File Next File Files

/home/joshua/develop/cpp/cps111/bigcalc/apstring	/home/joshua/develop/cpp/cps111/bigcalc/apstring.h
234 apstring operator + (char ch, const apstring &	43
235 // postcondition: returns concatenation of ch &	44 // indexing
236 {	45
237 apstring result; // make string equivalent	46 char operator[](int k) const;
238 result = ch;	47 char & operator[](int k);
239 result += str;	48
240 return result;	49 // modifiers
241 }	50
242	51 const apstring & operator += (const apstring
243 apstring operator + (const apstring & str, char	52 const apstring & operator += (char ch);
244 // postcondition: returns concatenation of str	53
245 {	54
246 apstring result(str);	55 private:
247 result += ch;	56 int myLength; // length
248 return result;	57 int myCapacity; // capacity
249 }	58 char * myCString; // storage
250	59 };
251	60
252 apstring apstring::substr(int pos, int len) con	61 // The following free (non-member) functions of
253 //description: extract and return the substr	62 //
254 // at index pos	63 // I/O functions
255 //precondition: this string represents c0, c1,	64
256 // 0 <= pos <= pos + len - 1 < n.	65

Wo hat sich der
Code *geändert*?

Wo hat sich das
Programm *geändert*?

retest-core/pom.xml

Hunk 1 : Lines 38-44

Stage hunk Discard hunk

```
38 38     <dependency>
39 39         <groupId>log4j</groupId>
40 40         <artifactId>log4j</artifactId>
41  -         <version>1.2.9</version>
41  +         <version>1.2.14</version>
42 42     </dependency>
43 43
44 44     <dependency>
```

static

vs.

dynamic

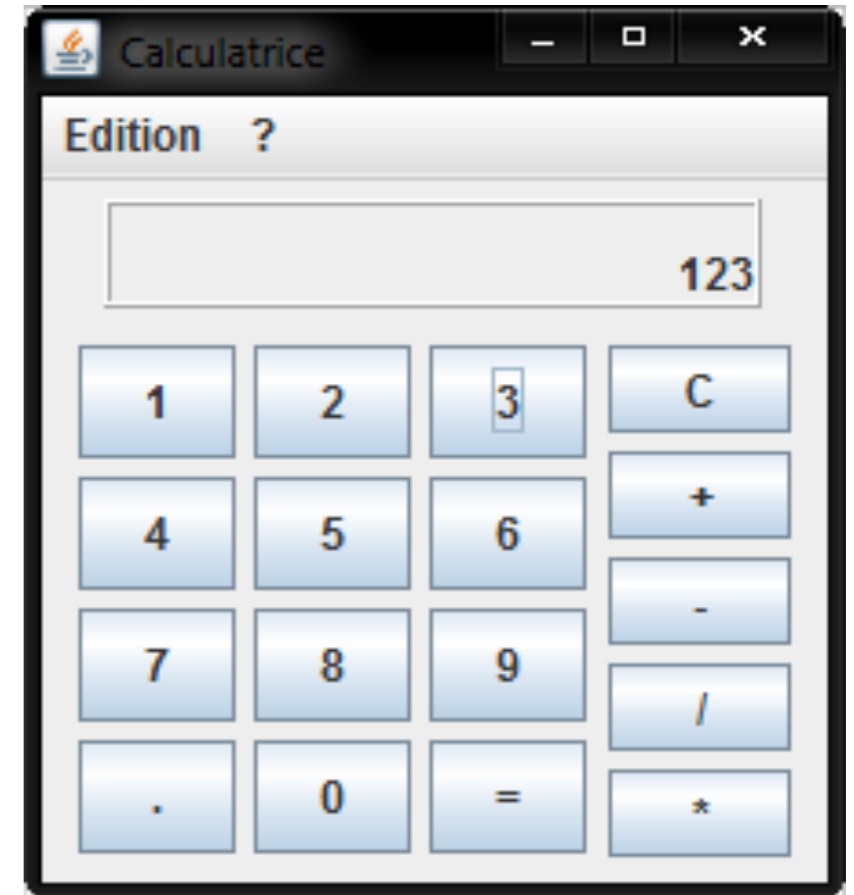
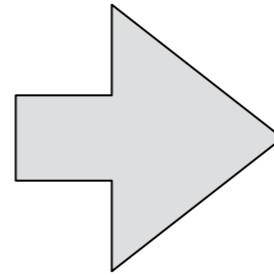
```
int getRandomNumber()  
{  
    return 4; // chosen by fair dice roll.  
              // guaranteed to be random.  
}
```

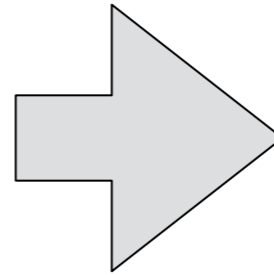
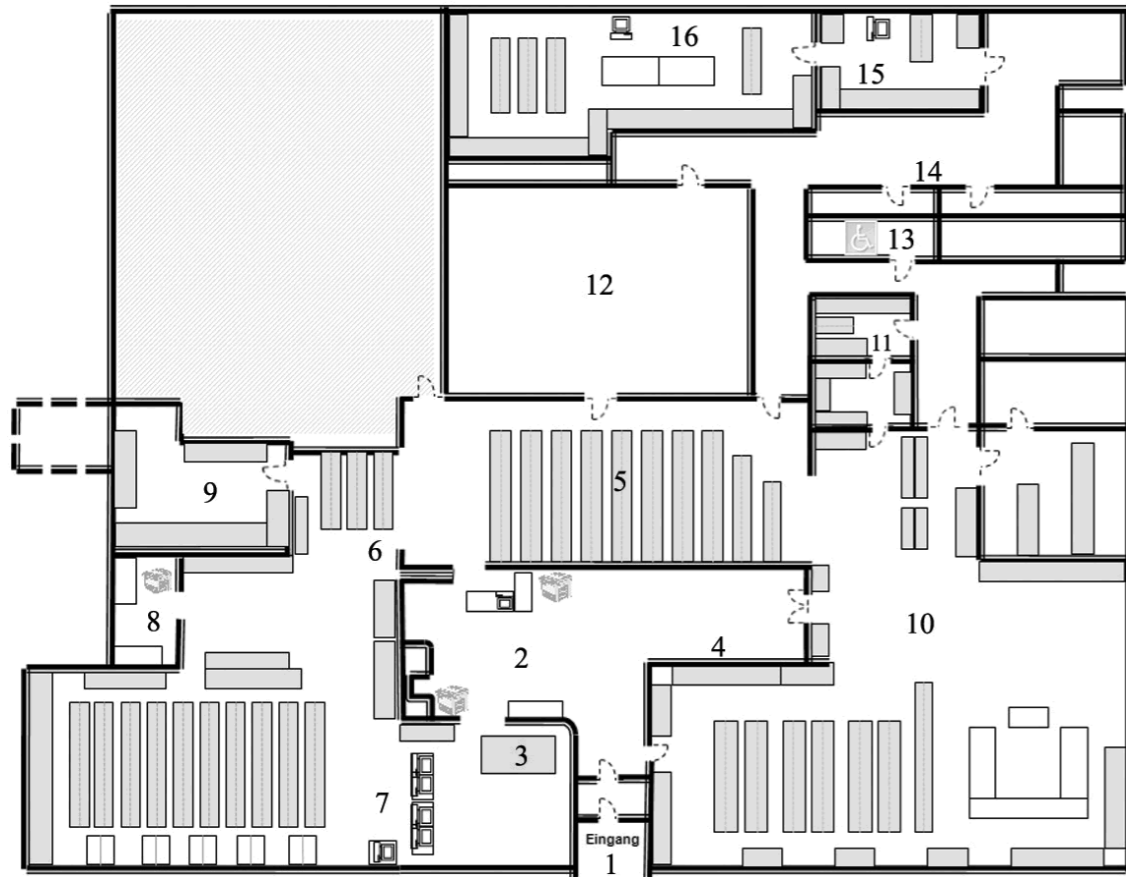
“ Beware of *bugs* in the above code;
I have only *proved* it correct,
not *tried* it.

Donald Knuth



```
function date_mysql2ger($t) {  
    list($year, $month, $day) = explode("-",  
    return sprintf("%02d.%02d.%04d", $day,  
    }  
function date_ger2mysql($datex) {  
    list($day, $month, $year) = explode("-",  
    return sprintf("%04d-%02d-%02d", $ye  
    }  
function timestamp_mysql2ger($t) {  
    return sprintf("%02d.%02d.%04d",  
        substr($t, 6, 2),  
        substr($t, 4, 2),  
        substr($t, 0, 4)  
    }  
function date_mysql2ger($datex) {  
    list($year, $month, $day) = ex  
    return sprintf("%04d-%02d-%02d", $ye
```











Die Lücke schließen: *automatische* Tests

GUI-Tests

automatisiert?!

1. Generation:

Capture/Replay

Untitled (untitled suite) - Selenium IDE 2.9.0 *

Base URL

Fast Slow

Test Case
Untitled *

Command	Target	Value
open	/	
clickAndWait	link=Häufige Fragen	

Command:

Target:

Value:

Select Find

link=Häufige Fragen link

//a[contains(text(),'Häufige Fragen')] xpath:link

//ul[@id='nav']/li[2]/ul/li[2]/a xpath:idRelative

//a[@href='http://www.retest.de/fa...'] xpath:href

//li[2]/a xpath:position

Runs: 0

Failures: 0

Log Reference UI-Element Rollup

clickAndWait(locator)
Generated from **click(locator)**

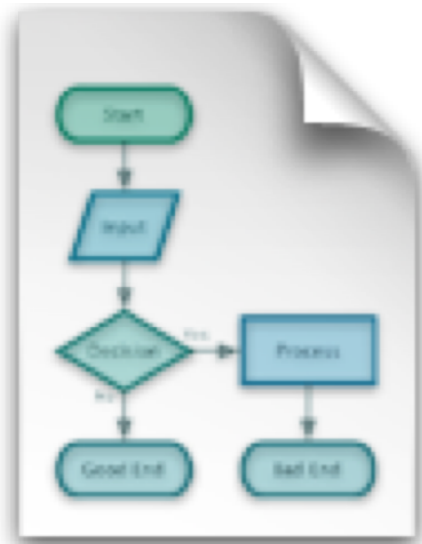
Arguments:

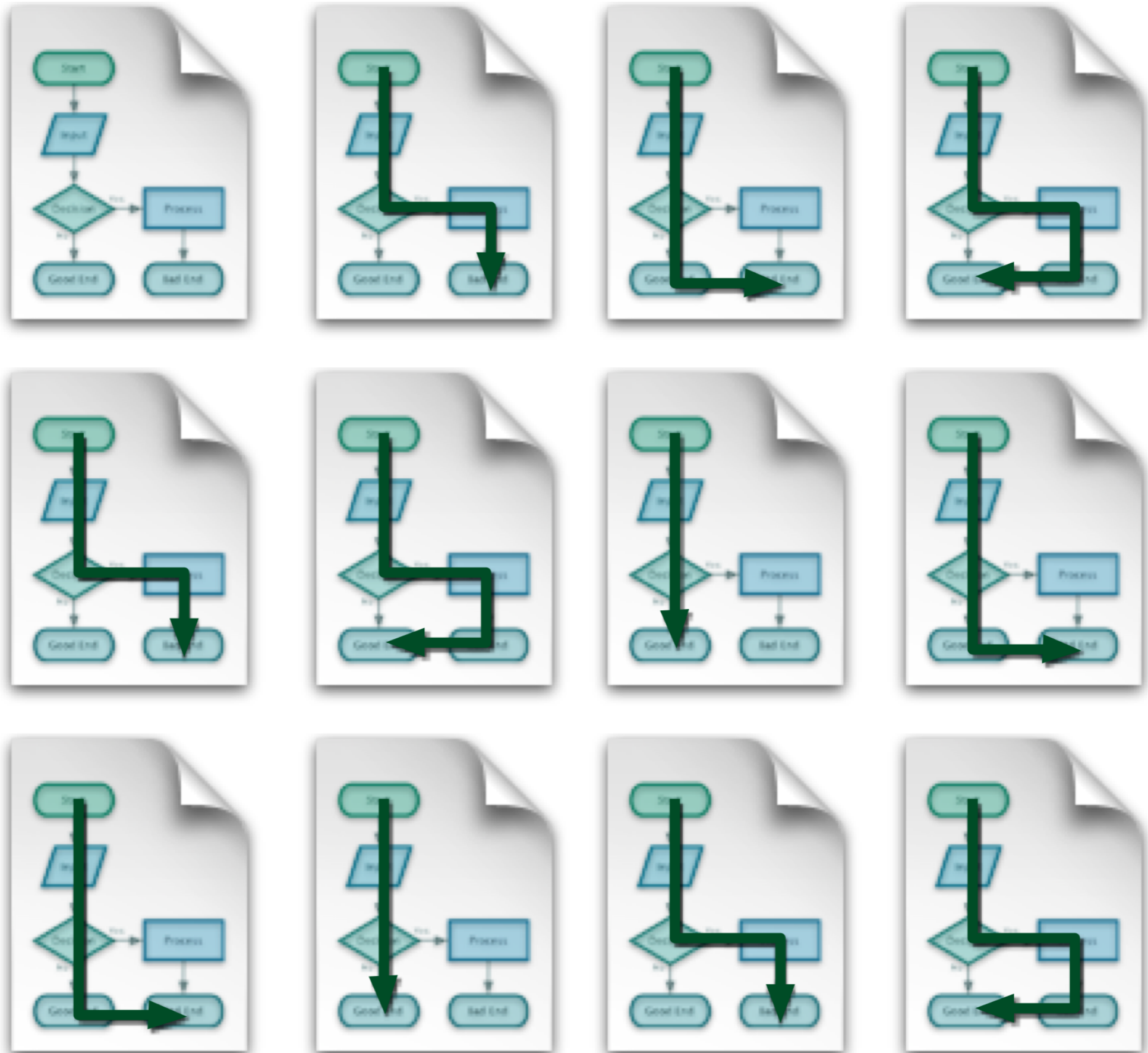
- locator - an element locator

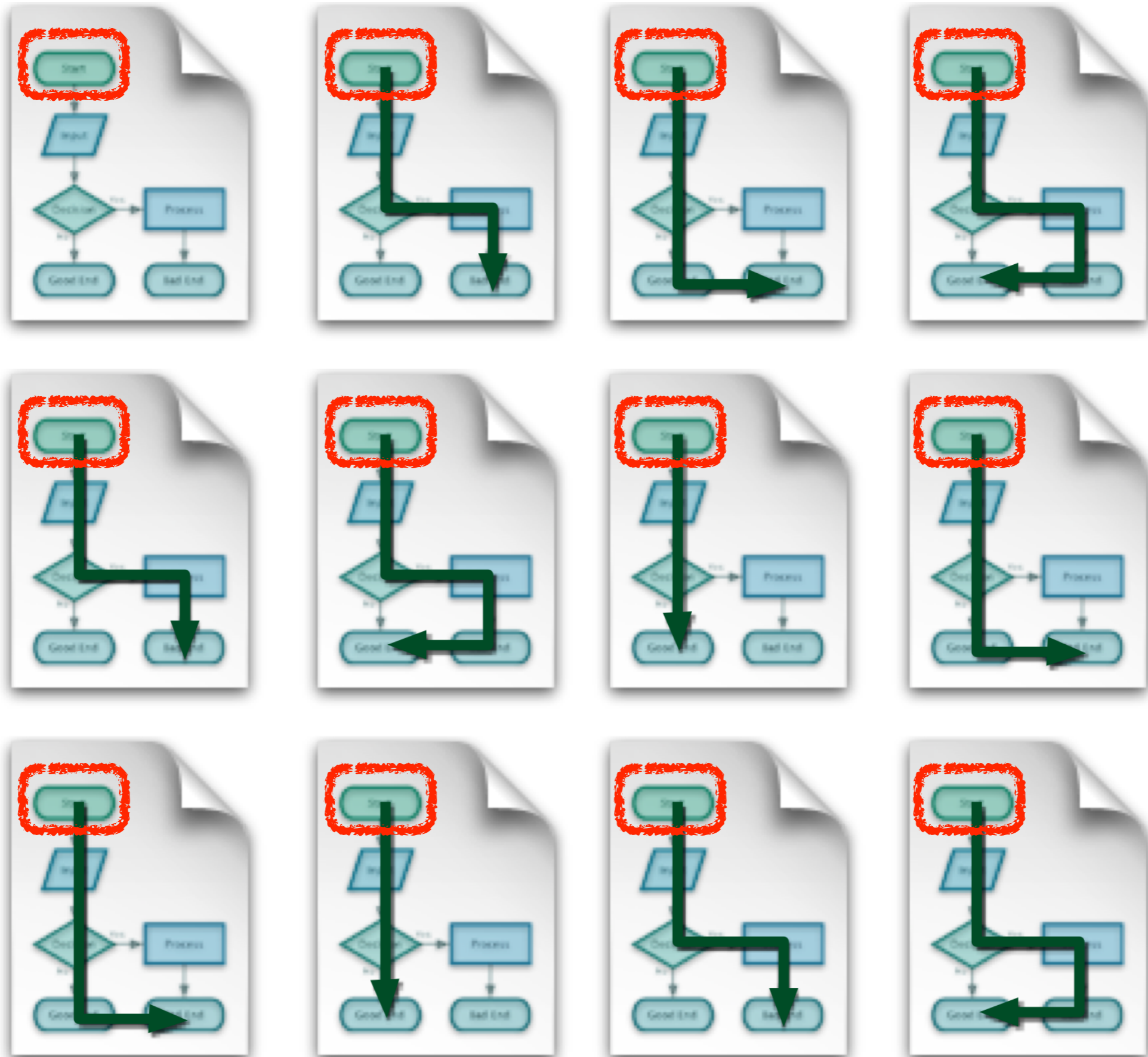
Clicks on a link, button, checkbox or radio button. If the click action causes a new page to load (like a link usually does), call `waitForPageToLoad`.

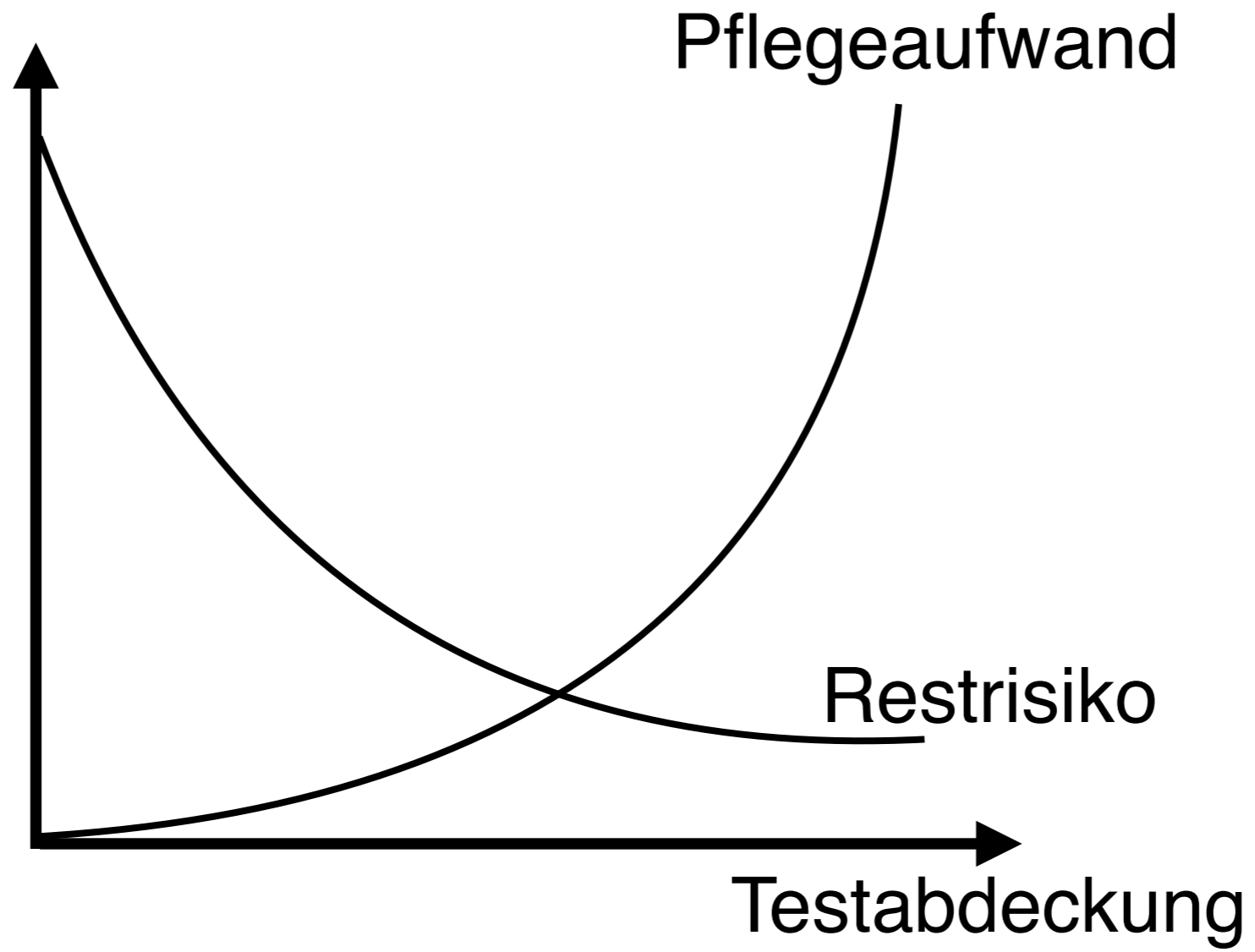
Problem:

Software Änderungen









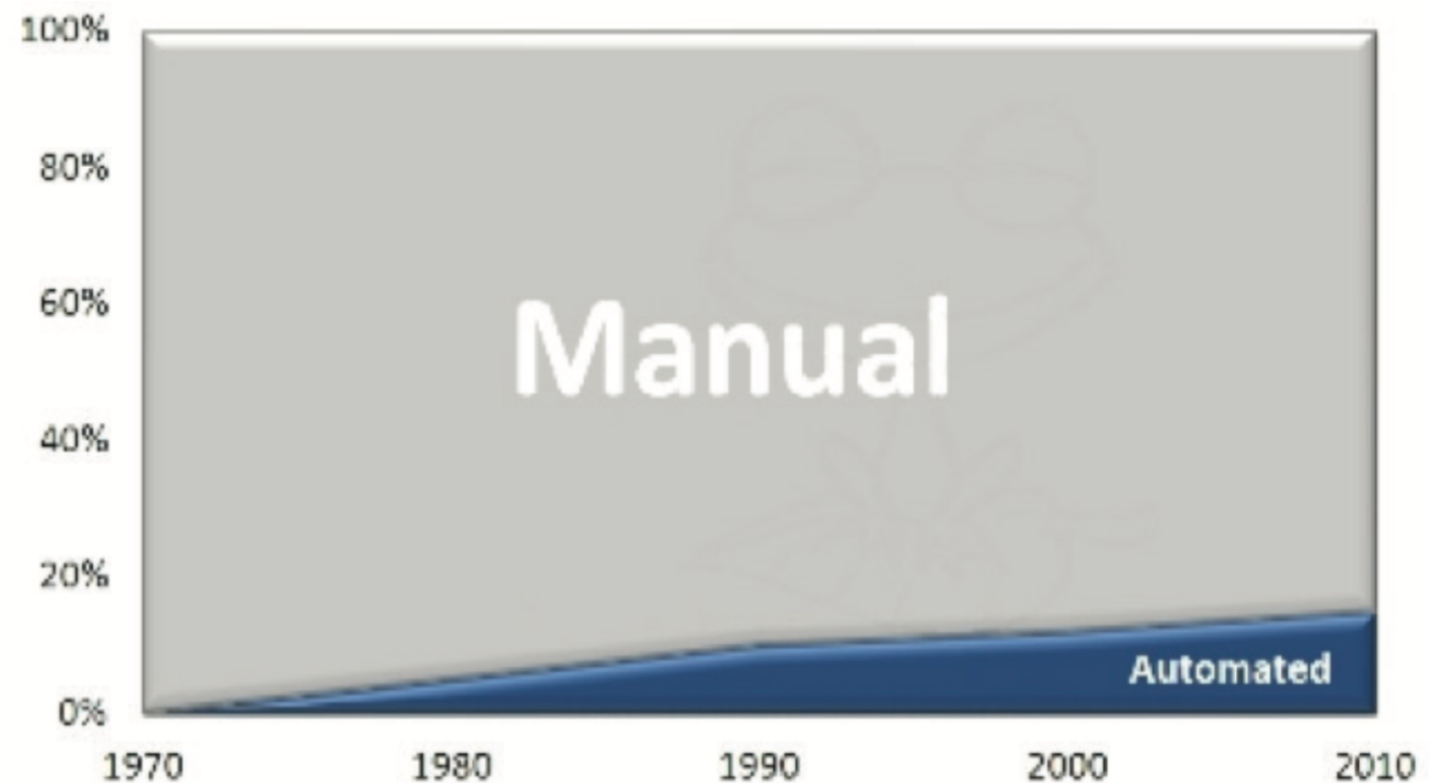
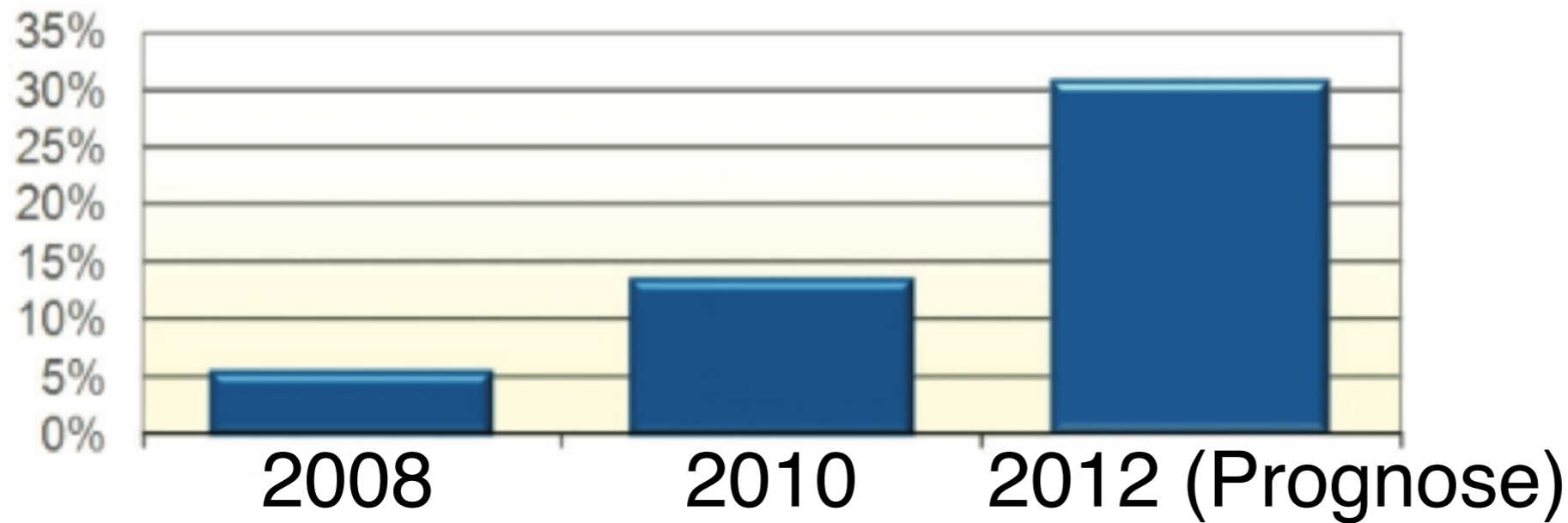
Redundanz in den Tests

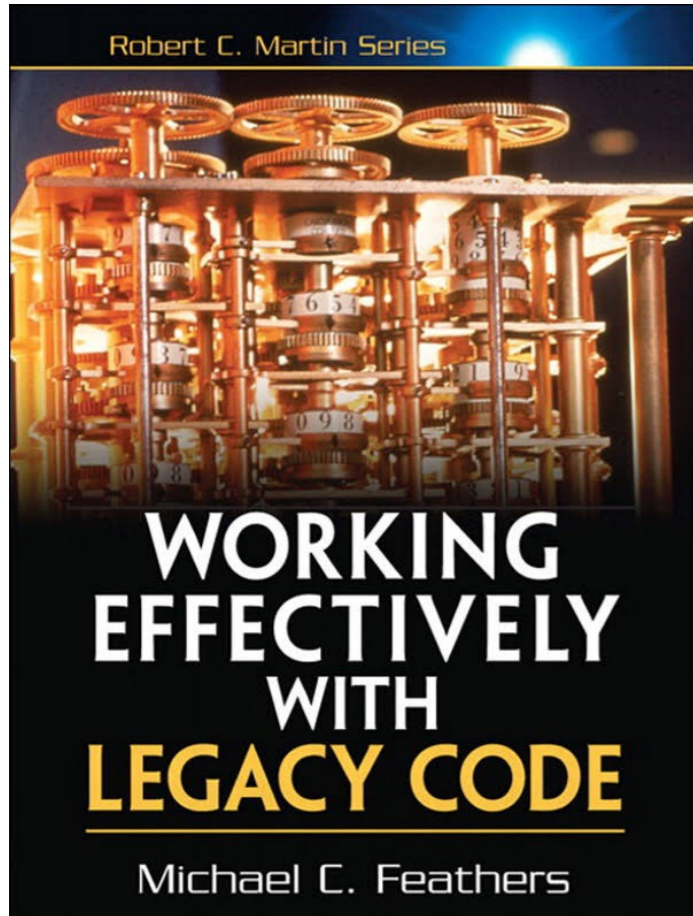
2. Generation:

Test Automation Frameworks

2. Programmierprojekt

Test Automatisierung





Legacy Code



Code without Tests

Legacy

Erbe

Hinterlassenschaft

Vermächtnis

Altlast

Legacy

“ Legacy code is *valuable code* that
we feel *afraid to change*.

J.B.Reinsberger





Foto: Fabian Zetterberg

“

Code without *tests* is *bad code*.

Michael Feathers

Es ist egal, wie ...

... gut geschrieben der Code ist

... schön der Code ist

... objektorientiert der Code ist

... entkoppelt der Code ist

Tests lassen uns Verhalten schnell
und *verifizierbar* ändern.

Ohne Tests wissen wir nicht,
ob der Code besser oder
schlechter wird.

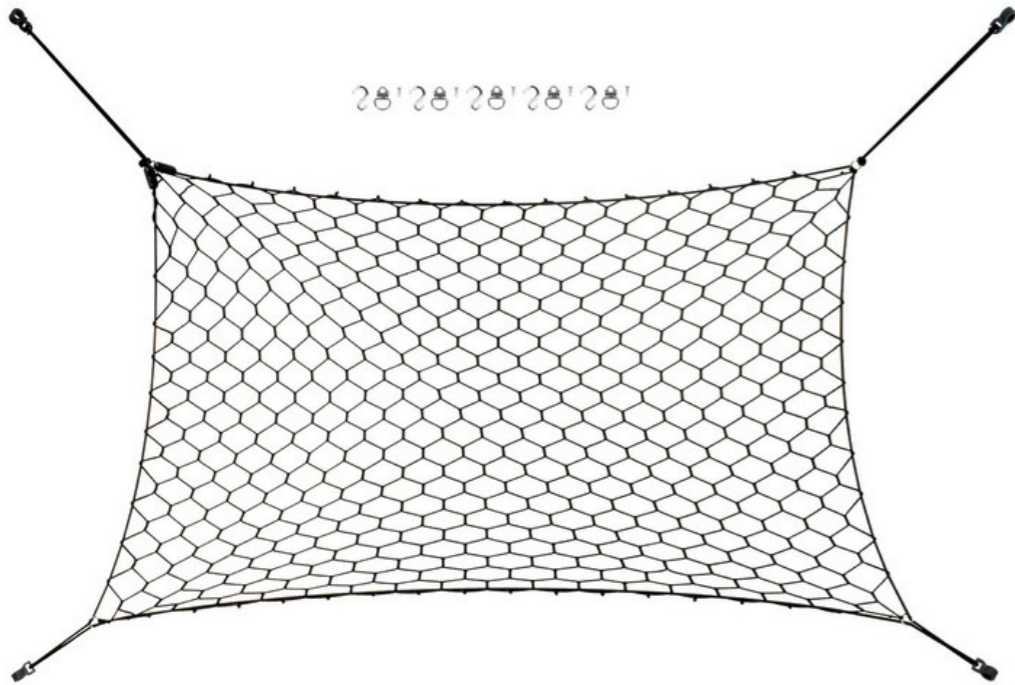
Dann schreiben wir halt Tests!

starke Kopplung

geringe Kohäsion

Code nicht testbar

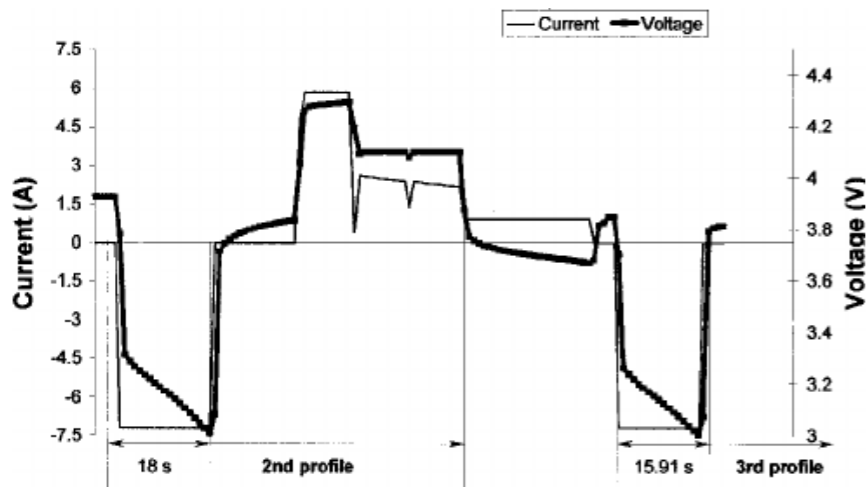
viel zu teuer!



Sicherheitsnetz

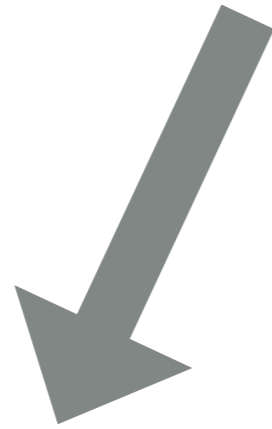


Golden Master



Characterization Test

Wozu testen wir?



Nach Implementierung?

$$1 + 1 = 3 \quad \times$$

$$2 + 2 = 4 \quad \checkmark$$

Auftragsbearbeitung

Dr. Kaufmann
 Refraktion: Herr Test
 Brillenart: Feinbrille
 Datum: 21.10.2004
 Auftrag ist fertig bis: 29.10.2004

R	Sph	Cyl	Achse	ADD	PD	NH	Pr. 1	Basis	Pr. 2	Basis	HSA	Visus	A	Betrag
	-1,50	-0,50	178°											224,50
														185,20

1/1 Inventur Artikelste Ausfragen Export Duplizieren Etikett

Datum EK: 06.04
 Kalkulation: 1.17931
 EK-Schlüssel: 20000

BUCHEN: Sachkonten Allgemeinbeleg

Deutsche Asphalt HW/FFM
 Erlöse HotLine - Wartung
 EUR 0,00 Erlöse HotLine - Wartung
 EUR 142.747,17 H

Konto: 00208
 Gegenkonto: 5210
 SSK: 9 [16,00% MwSt in 100]

Vertragsnr. 00184 als Buchung
 Option bis: 05.02.2008
 gebucht am: 04.02.2008

Anreise - Daten
 Anreise (Mon): 01.02.2008 (4 Tage)
 Abreise (Die): 05.02.2008

Preise und sonstige Informationen zur Buchung
 Errechneter Mietpreis: 199,00 EUR
 Summe Sparten: 19,00 EUR
 Provision: -24,28 EUR
 Versicherung: 5,68 EUR
 Pauschalkosten: 7,66 EUR
 sonstige Kosten: 616,76 EUR
 Endpreis: 656,76 EUR

Teilnahme: 201006 Mustermeier Daniel; 3: Definitiv

Name: Mustermeier Daniel
 Vorname: Daniel
 Anrede / G: Herr M
 Zusatz: M
 Sprache: D
 Strasse/Nr.: Bahnhofstrasse
 L/PLZ/Ort: CH 6313

Zuweisung	Rechnung	Budget	Kosten	Korrespondenz
	Teilnehmerin / Teilnehmer	4450		
	Name	Mustermeier	Kursnummer	201006
	Vorname	Daniel	Kursjahr	2000/01
	Anrede / G	Herrn	Periode	
	Zusatz	M	Kürzel / Fach	KB-1C
	Strasse/Nr.	Bahnhofstrasse	Stufe	1C
	L/PLZ/Ort	CH 6313	Kursnummer	201006
	Geb.-Datum	00.00.00		
	Interessen			
	Status	3: Definitiv		
	Status-Grund			
	Status-History			

Personendaten

Vorname: Jürgen
 Nachname: Cimbollek
 Straße: Musterlee 33
 Land/PLZ/Ort: 0224 2221
 Anr. Adresse: Musterlee 33
 E-Mail: reinstein@hans.testmann.d

Wohnorte bearbeiten

Lfd. Nr.: 6
 Land: D
 Territorium: SA
 geogr. Gebiet: Altmark
 aktuelle PLZ: 29410
 Ort: Salzwedel
 bei: Salzwedel
 alt. Straßenna.:
 Wohnort: Am Karlsfeld 9
 Grund-Umzug: von 1995 bis ????

Jahreszeugnisse Realschule

Name, Vorname: Mustermann

Kopfnoten: Verhalten: 2

Leistungen in den einzelnen Fächern:

Ethik	4	NWA	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
Gemeinschaftskunde		Technik	3
TOP Informationstechnik	2	TOP WVR	0
TOP Tech. Arbeiten		TOP Social. Engagement	2

Auswahl
 Erster Schüler: 14 | Schüler Nr.: 6 von 25 | Letzter Schüler

Notendurchschnitt: 3,4

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

Kunde TELEKOM

PR-Gruppe 1: 3 IT-Projekte
 PR-Gruppe 2: 3.1 neue IT-Systeme
 PR-Gruppe 3:

Wunschtermin: 09.02.06 bis 25.05.06
 Kalkulierter Termin: 21.05.06 bis 05.01.07

Parameter	Dauern	Aufwände	Kosten
Budget	680 h	85.000,00 €	
Soll	97 h	699 h	81.070,00 €
Ist	132 T	82 h	4.560,00 €
Rest	30 T	617 h	74.510,00 €
Gesamt	229 T	699 h	81.070,00 €

EK-Wert: -11,04 €
 Lagerort: Zuschnitt
 Artikel ist Best.: Ja
 Bestell.Bem.: Bitte Eloxieren
 Bemerkung: Spezial Alu für Demomuster

Pos-Zelle: 7 / 675

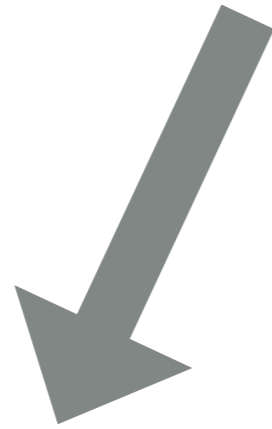
Querkontraktionszahl mue	0,3	Querkontraktionszahl mue	0,25
Längenausdehnungskoeffizient	1,15E-5	Längenausdehnungskoeffizient	9E-6
Zulässige Fügetemperatur Tzul	600 °C	Zulässige Fügetemperatur Tzul	400 °C
Dichte rho	7,8 kg/dm³	Dichte rho	7,35 kg/dm³

Beschwerden

Datensatz: 1 von 3860

Druck	Benutzer	Top	Kommentar
02	MEIER		Angebot 201000428 gemalt
37	0105	FRISCH	Hochgehalt, braucht mehr Zeit, ziehen erstmal um -> nächstes Jahr
24	0110	FRISCH	Hochgehalt, wenig Fragen zur Feedbackseite erhalten, um bearbeitet, evaluieren
15	RUEHRIG		Standard-E-Mail mit Info zu Produkt als PDF geschickt
46	RUEHRIG		Anfrage nach Produkt
39	0115	MEIER	Hochgehalt, möchte Angebot
28	0103	FRISCH	Hochgehalt, Zeit überlegen -> Auf. Fabr.

Wozu testen wir?



Nach
Implementierung?

$$1 + 1 = 3 \quad \times$$
$$2 + 2 = 4 \quad \checkmark$$



Nach
Änderung?

$$1 + 1 = 3 \quad \checkmark$$
$$1 + 1 = 5 \quad \times$$



```
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?



“ “ High test/code ratio
is a symptom of high coupling in the code.
Or *wasteful testing*.

Kent Beck

DIFFERENCE-TESTING

Capture/Replay

+ Golden Master

+ Versionskontrolle

= ReTest



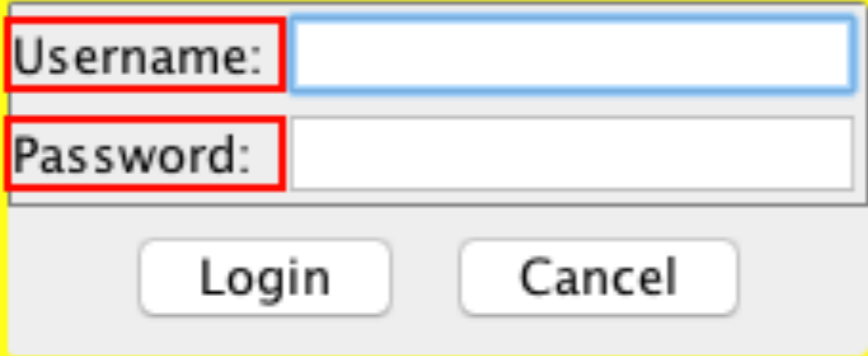
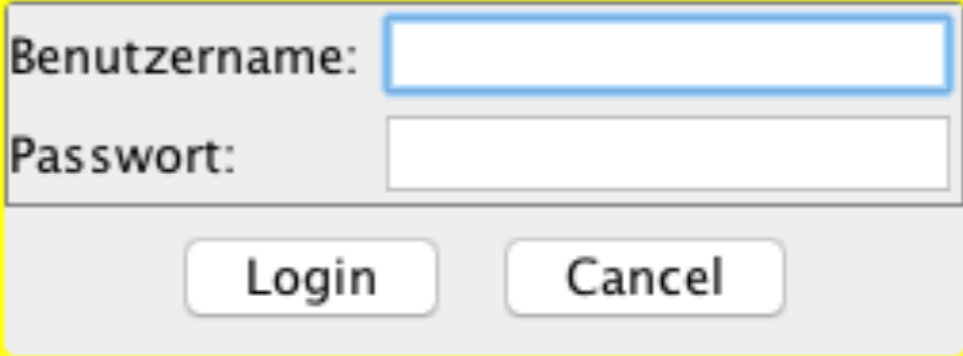
file:///home/joshua/dev el/cpp/cps111/bigcalc/apstring.cpp | file:///home/joshua/dev el/cpp/

File Difference Settings Help

Viewing: **apstring.cpp** Previous File Next File Files

/home/joshua/dev el/cpp/cps111/bigcalc/apstring	/home/joshua/dev el/cpp/cps111/bigcalc/apstring.h
234 apstring operator + (char ch, const apstring &	43
235 // postcondition: returns concatenation of ch &	44 // indexing
236 {	45
237 apstring result; // make string equivalent	46 char operator[](int k) const;
238 result = ch;	47 char & operator[](int k);
239 result += str;	48
240 return result;	49 // modifiers
241 }	50
242	51 const apstring & operator += (const apstring
243 apstring operator + (const apstring & str, char	52 const apstring & operator += (char ch);
244 // postcondition: returns concatenation of str	53
245 {	54
246 apstring result(str);	55 private:
247 result += ch;	56 int myLength; // leng
248 return result;	57 int myCapacity; // capa
249 }	58 char * myCString; // stor
250	59 };
251	60
252 apstring apstring::substr(int pos, int len) con	61 // The following free (non-member) functions of
253 //description: extract and return the substr	62 //
254 //	63 // I/O functions
255 //precondition: this string represents c0, c1,	64
256 //	65 return & operator[](int k);

252 apstring apstring::substr(int pos, int len) con
 253 //description: extract and return the substr
 254 //
 255 //precondition: this string represents c0, c1,
 256 //

Expected	Actual
<p style="text-align: center;">Login-Window</p> 	<p style="text-align: center;">Login-Window</p> 
<p style="text-align: center;">Password: text = Password:</p>	<p style="text-align: center;">Passwort: text = Passwort:</p>
<p style="text-align: center;">Username: text = Username:</p>	<p style="text-align: center;">Benutzername: text = Benutzername:</p>

Attribute	Expected	Actual	Affects
<input type="checkbox"/> text	Password:	Passwort:	Components: Password: Tests: login.actions
<input checked="" type="checkbox"/> text	Username:	Benutzername:	Components: Username: Tests: login.actions

Demo!

Untitled (untitled suite) - Selenium IDE 2.9.0 *

Base URL

Fast Slow

Test Case
Untitled *

Command	Target	Value
open	/	
clickAndWait	link=Häufige Fragen	

Command:

Target:

Value:

Select Find

link=Häufige Fragen	link
link=Häufige Fragen	link
//a[contains(text(),'Häufige Fragen')]	xpath:link
//ul[@id='nav']/li[2]/ul/li[2]/a	xpath:idRelative
//a[@href='http://www.retest.de/fa...]	xpath:href
//li[2]/a	xpath:position

Runs: 0
Failures: 0

Log Reference UI-Element Rollup

clickAndWait(locator)
Generated from **click(locator)**

Arguments:

- locator - an element locator

Clicks on a link, button, checkbox or radio button. If the click action causes a new page to load (like a link usually does), call `waitForPageToLoad`.

Attribute	Expected	Actual	Affects
<input type="checkbox"/> text	Password:	Passwort:	Components: Password: Tests: login.actions
<input checked="" type="checkbox"/> text	Username:	Benutzername:	Components: Username: Tests: login.actions

Software-Änderung als Grundparadigma

Einfache Erstellung

Einfache Pflege

Capture/Replay

+ Golden Master

+ Versionskontrolle

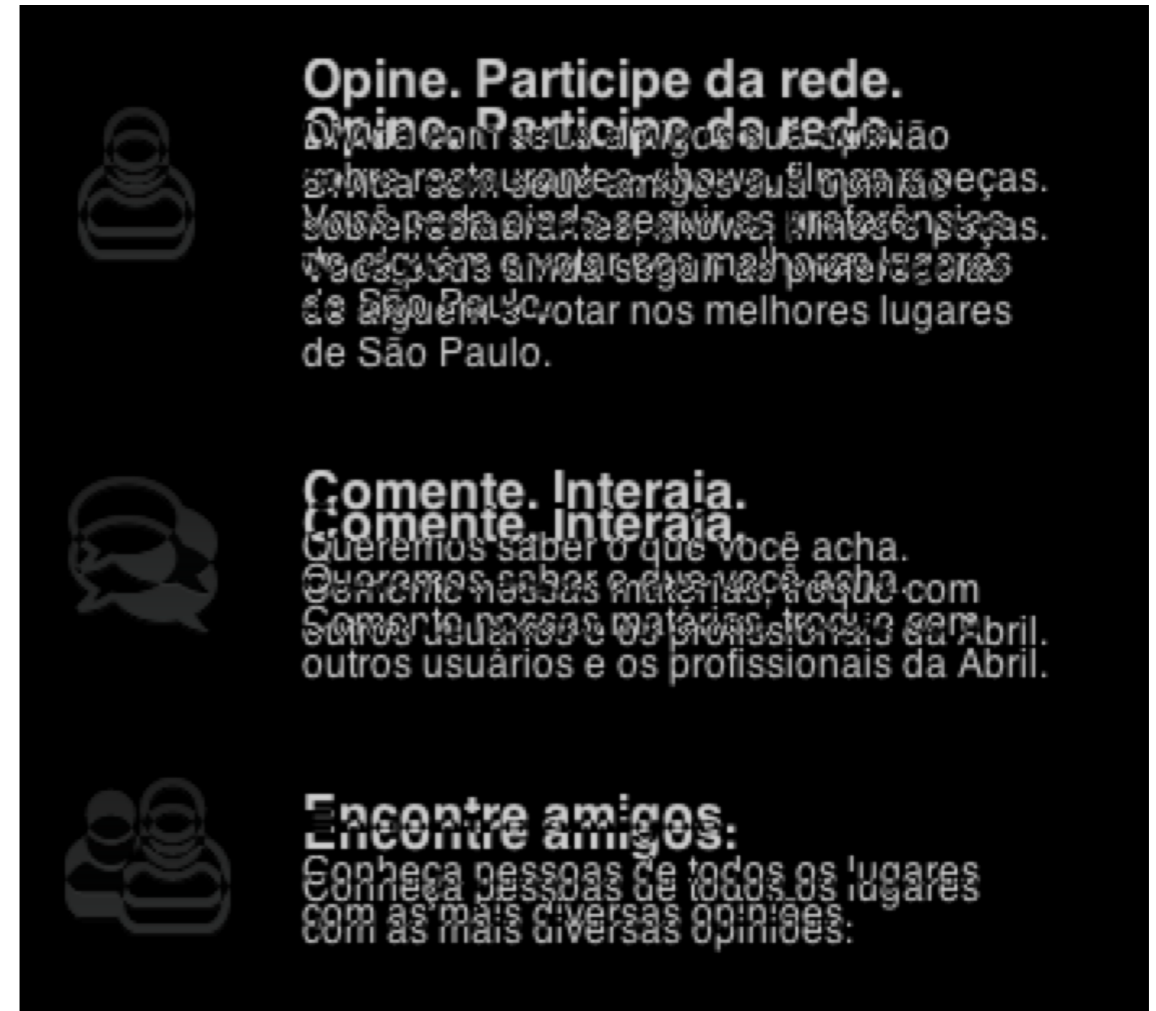
= ReTest



```
[TestMethod]
public void TestPurchase()
{
    // Create a new receipt
    var r = new Receipt();
    // Add 1 candy bar at $.50
    r.AddItem(1, "Candy Bar", 0.50);
    // Add 2 sodas bar at $1
    r.AddItem(2, "Soda", 1.0);
    // verify the receipt
    Approvals.Verify(r.ToString());
}
```



ApprovalTests



automated-screenshot-diff

Next generation frontend regression testing tool <http://www.retest.de> — Edit

3 commits 1 branch 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find file Clone or download

roesslerj committed on GitHub Add some initial info Latest commit d25df90 4 minutes ago

CONTRIBUTING.md Create CONTRIBUTING.md 4 hours ago

README.md Add some initial info 4 minutes ago

README.md

ReTest

is a next generation frontend regression testing tool that combines

Capture & Replay with Golden Master testing and a VCS-like approach to test maintenance.

More info can be found at <http://www.retest.de>



fully automated progress testing
software engineering efficiency

Ende 2016

Interesse?

Unser Service ist momentan noch nicht für jeden Kunden verfügbar, aber wir arbeiten daran.

Wenn Sie Interesse haben, tragen Sie Ihre Email ein und wir melden uns sobald wie möglich bei Ihnen!

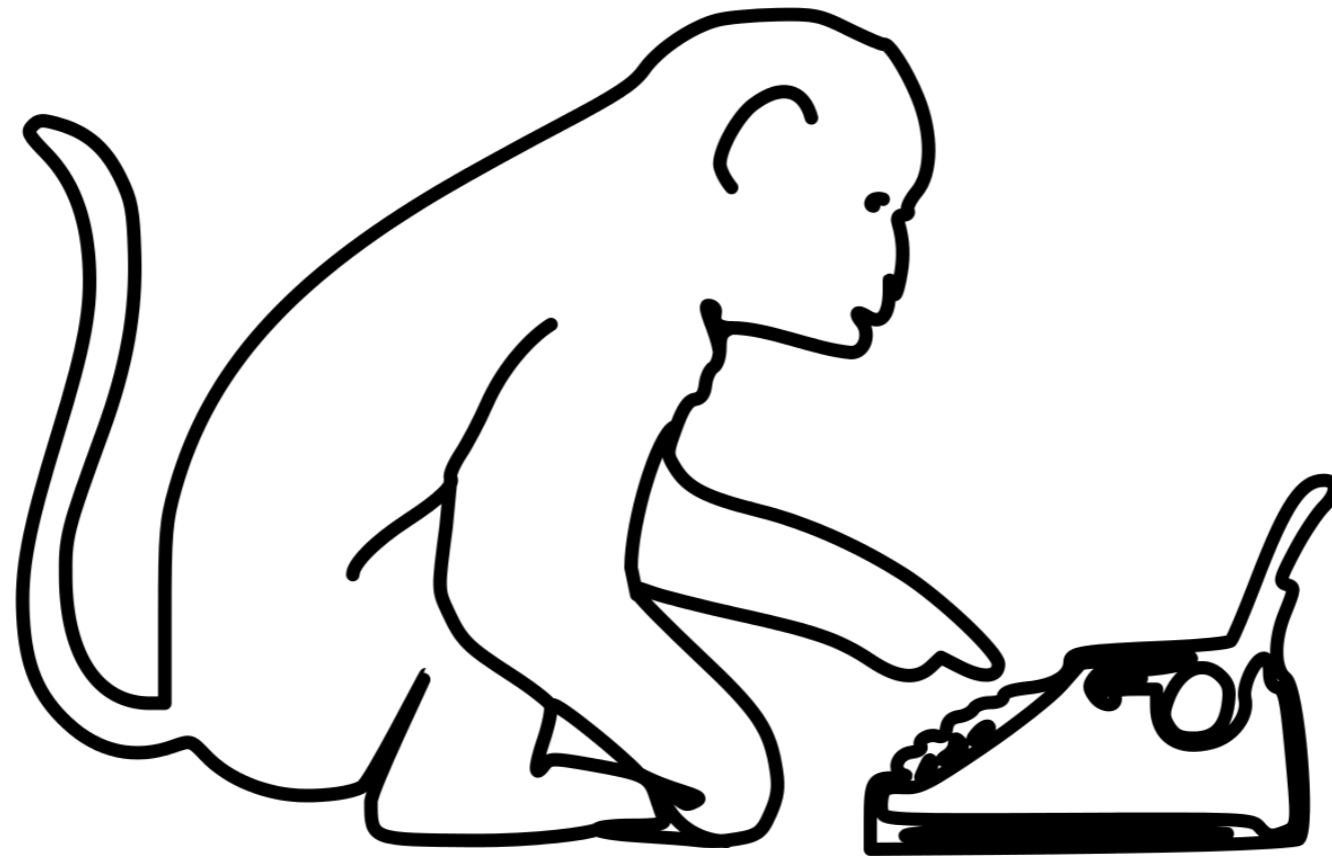
  A form consisting of a light gray input field with the placeholder text "E-Mail Adresse" and a small red icon of a person with a checkmark. To the right of the input field is a green button with the text "Anmelden" in white.

Wir werden Ihre Email-Adresse nur verwenden um Sie zu kontaktieren, sobald unser Service für Sie verfügbar ist.

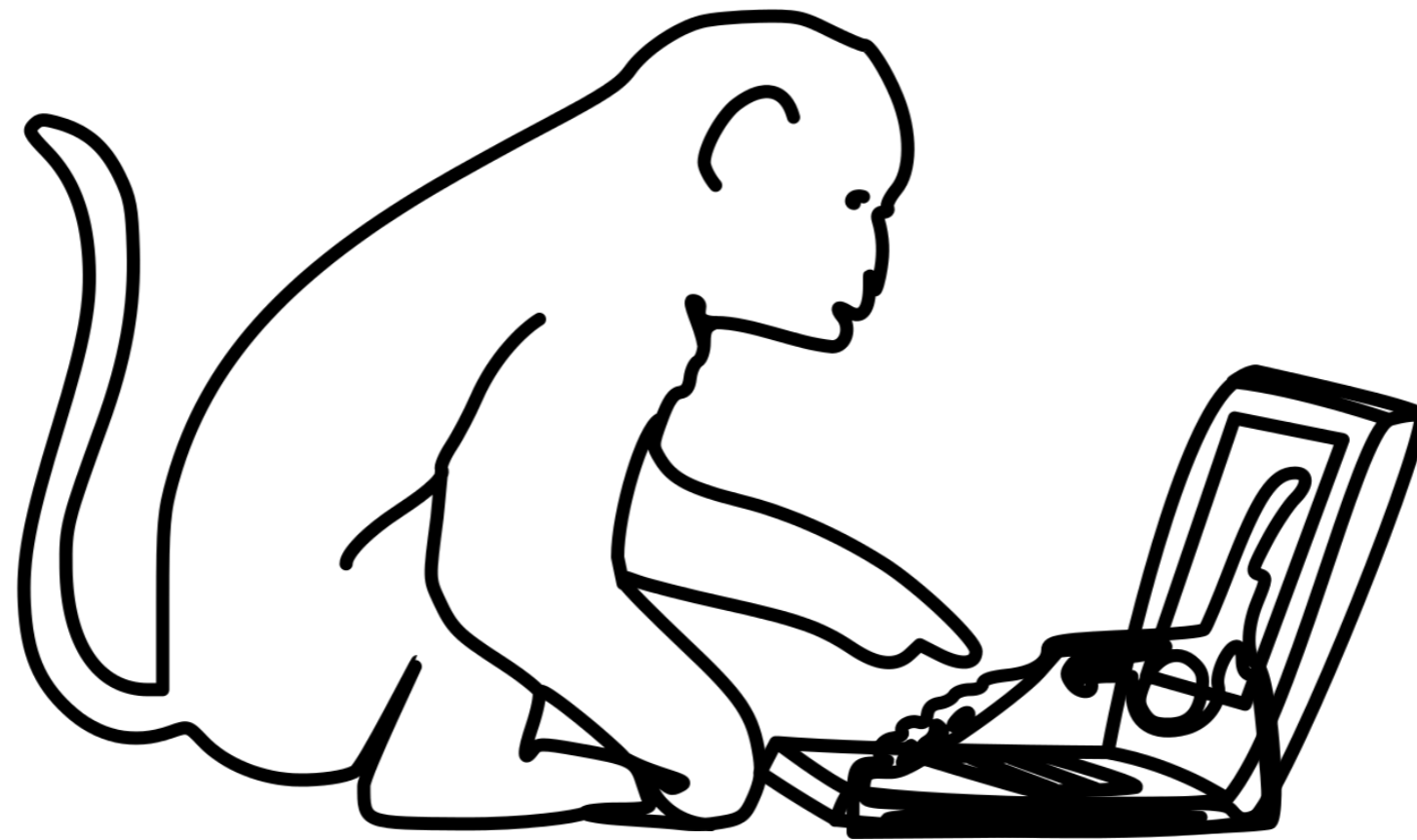
Wir werden Ihre Email-Adresse nicht weitergeben und nicht missbrauchen. **Versprochen!**

www.retest.de

**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...

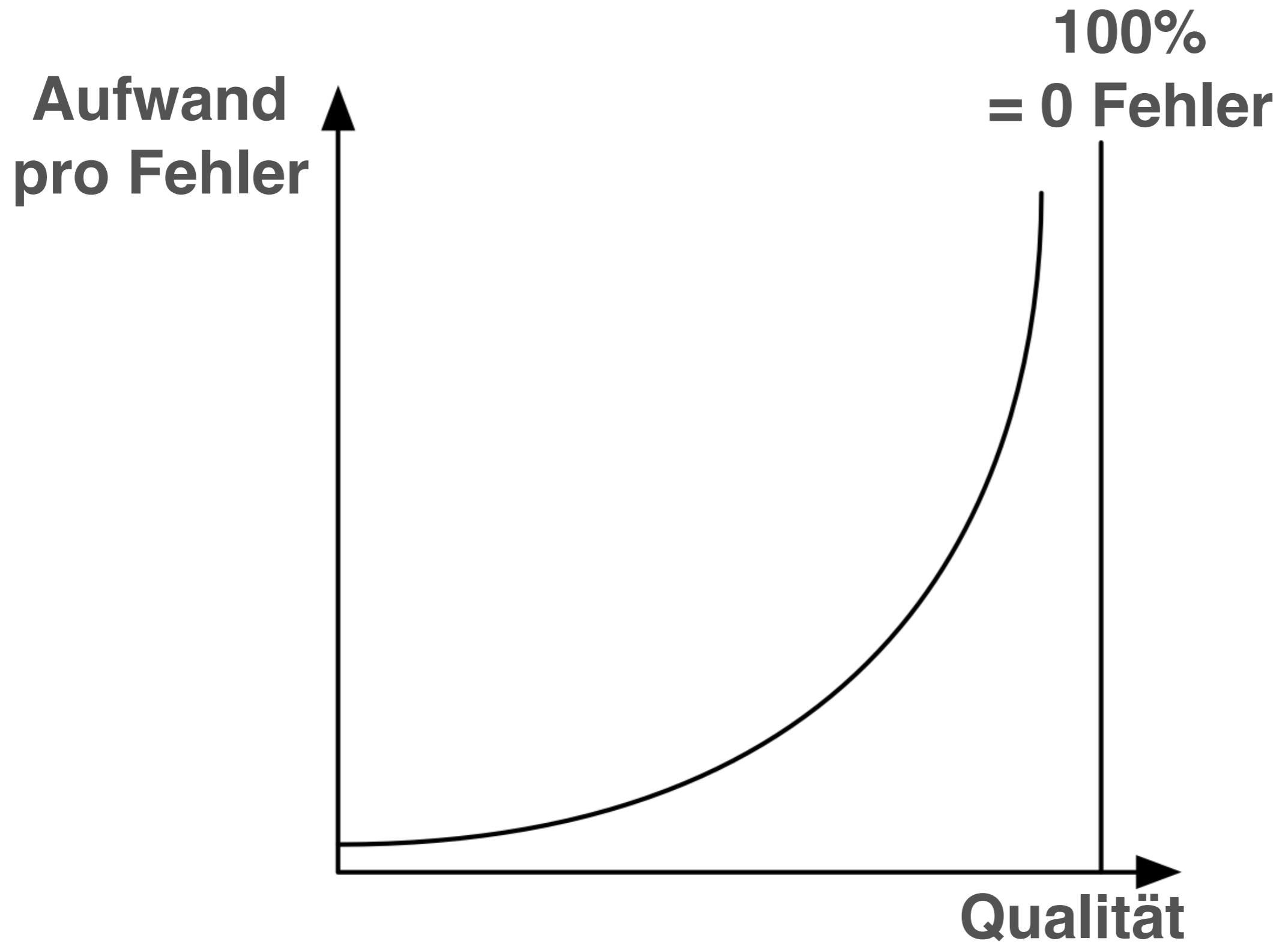


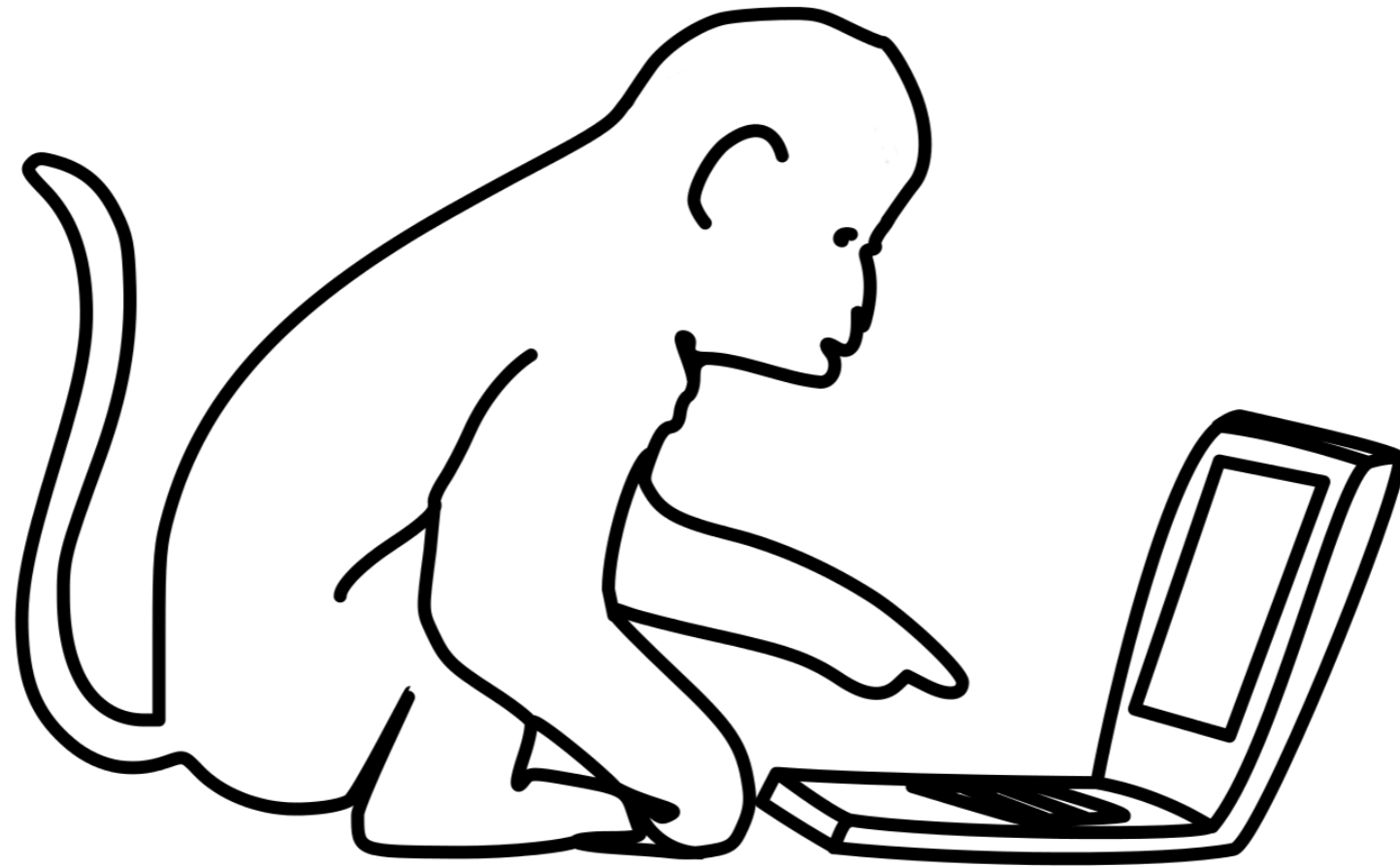
Infinite Monkey

```
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





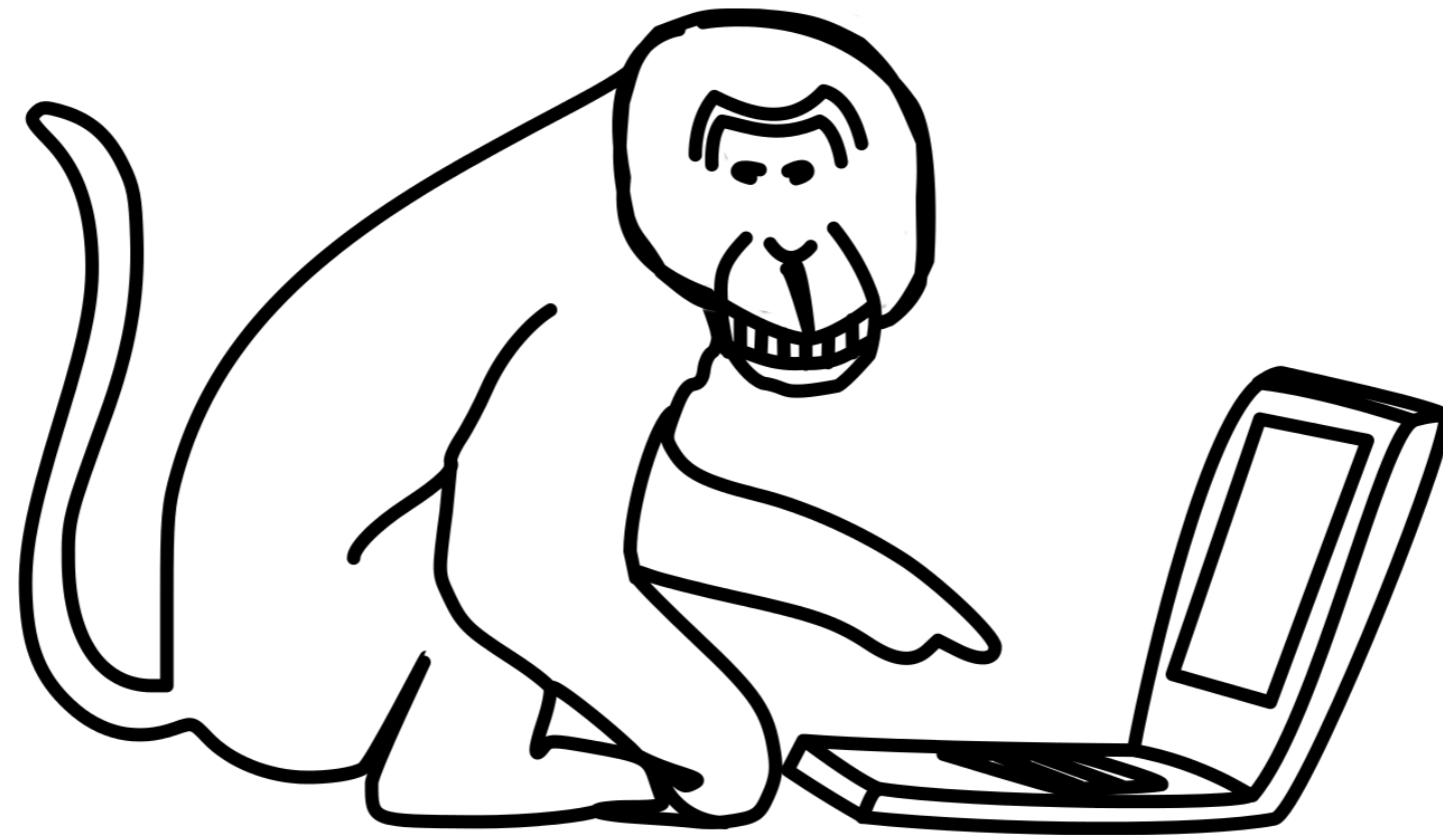
Intelligenter Affe

```
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

Intelligenter Affe



Demo!

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?

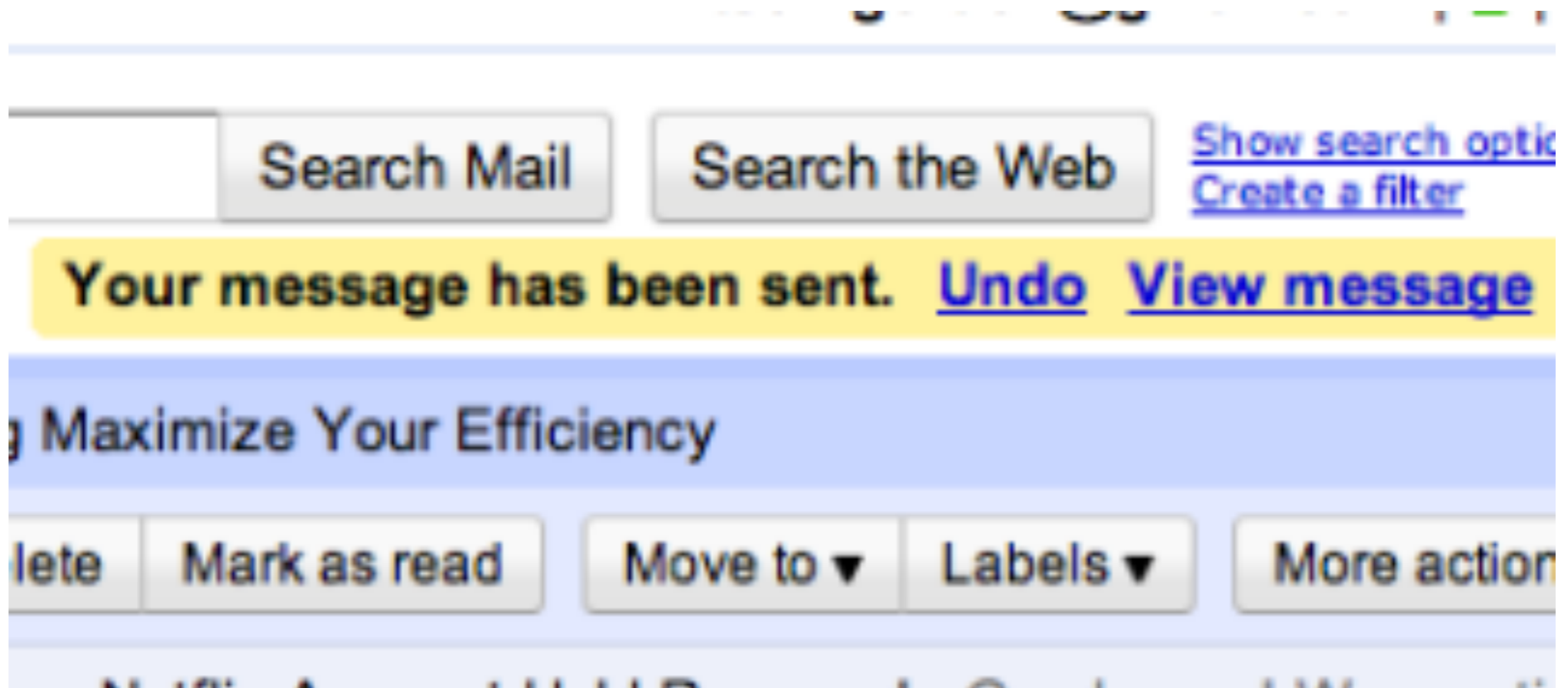
```
.  
..  
text.c  
other.files
```

```
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



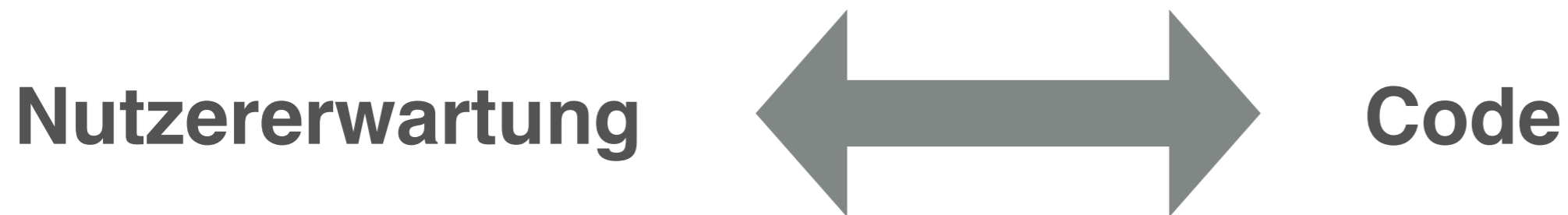
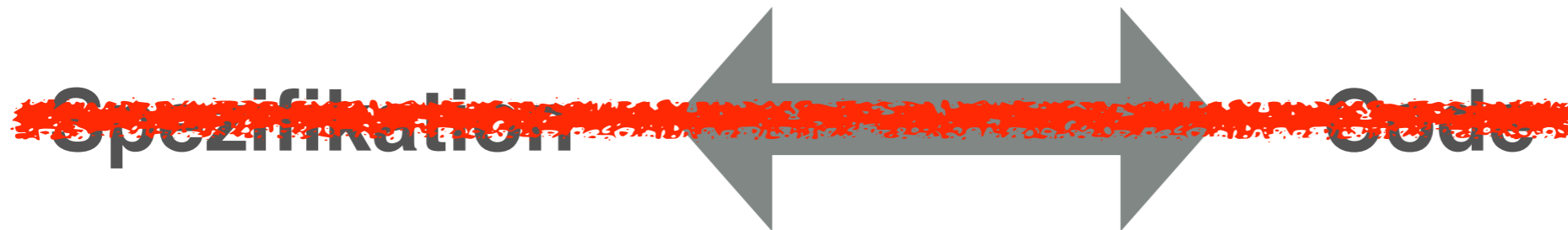
Code

Nutzererwartung



Code

Monkey Testing



Wozu testen wir?



**Nach
Implementierung?**



**Nach
Änderung?**

The image displays a collage of various software application windows, demonstrating different user interface elements and data handling capabilities.

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

Technical Specification Dialog

Querkontraktionszahl μ	0,3	Querkontraktionszahl μ	0,25
Längsausdehnungskoeffizient	1150,5 1/K	Längsausdehnungskoeffizient	10,6 1/K
Zulässige Fugtemperatur T_{zul}	100 °C	Zulässige Fugtemperatur T_{zul}	100 °C
Dichte ρ	7,8 kg/dm ³	Dichte ρ	7,35 kg/dm ³



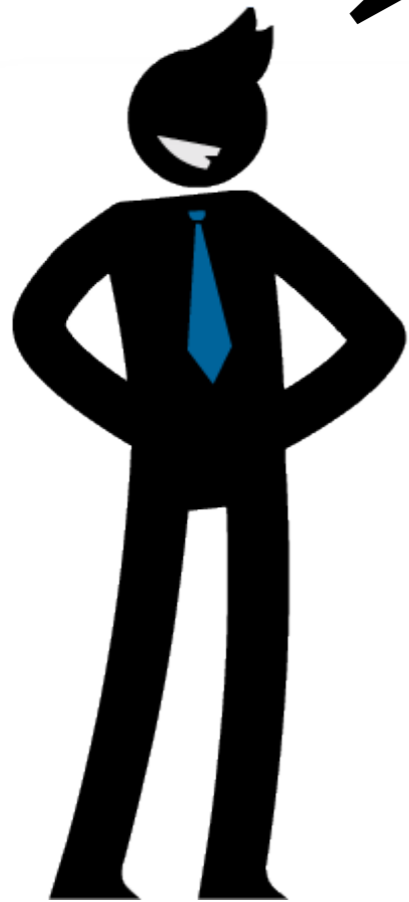
WE WANT YOU!

The ReTest logo, consisting of a green stylized arrow pointing upwards and to the right, followed by the text "ReTest" in a dark blue, sans-serif font.

ReTest

<https://github.com/retest/retest>

Machen Sie den Test!

The main ReTest logo, consisting of a green stylized bar chart icon to the left of the text "ReTest" in a dark blue font.

www.retest.de