



Test Automation without Assertions

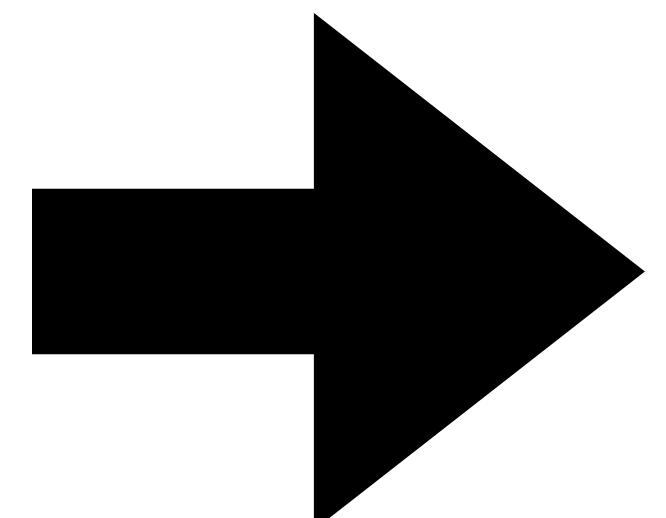
Dr. Jeremias Rößler (Roessler)

<https://retest.de>



Please star us on GitHub!

Everyone who gives us a GitHub star, gets a sticker.





```
expect(umbrellaOpens).toBe(true)
```

tests: 1 passed, 1 total

****all tests passed****

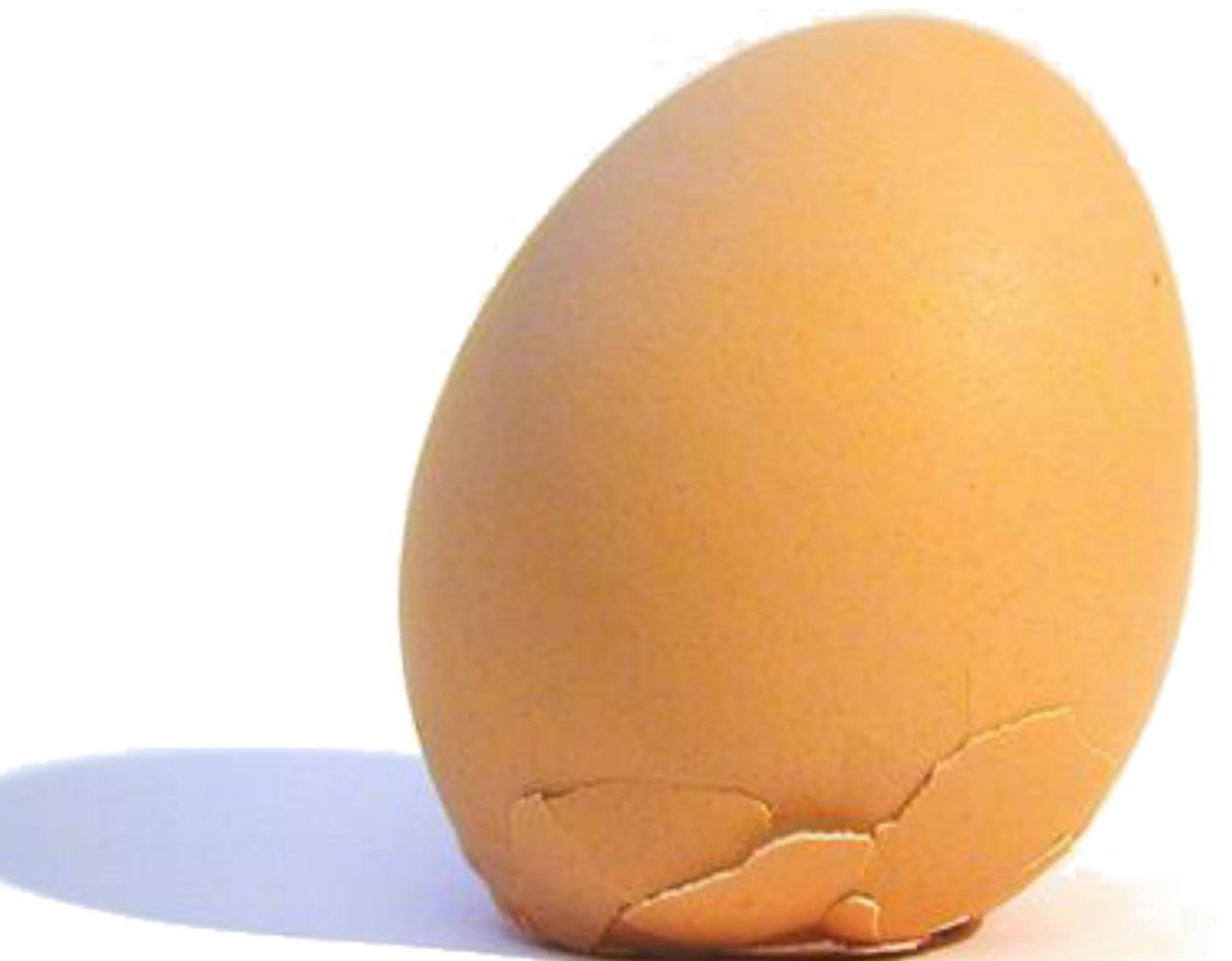




Teaser-DEMO



We do one thing **different**.





You don't define the outcome,
you approve it.

First you ensure that everything is as wanted.
Then we make sure it doesn't change.



automated tests

≠

tests



automated tests

=

version control



Where has the
Code *changed?*

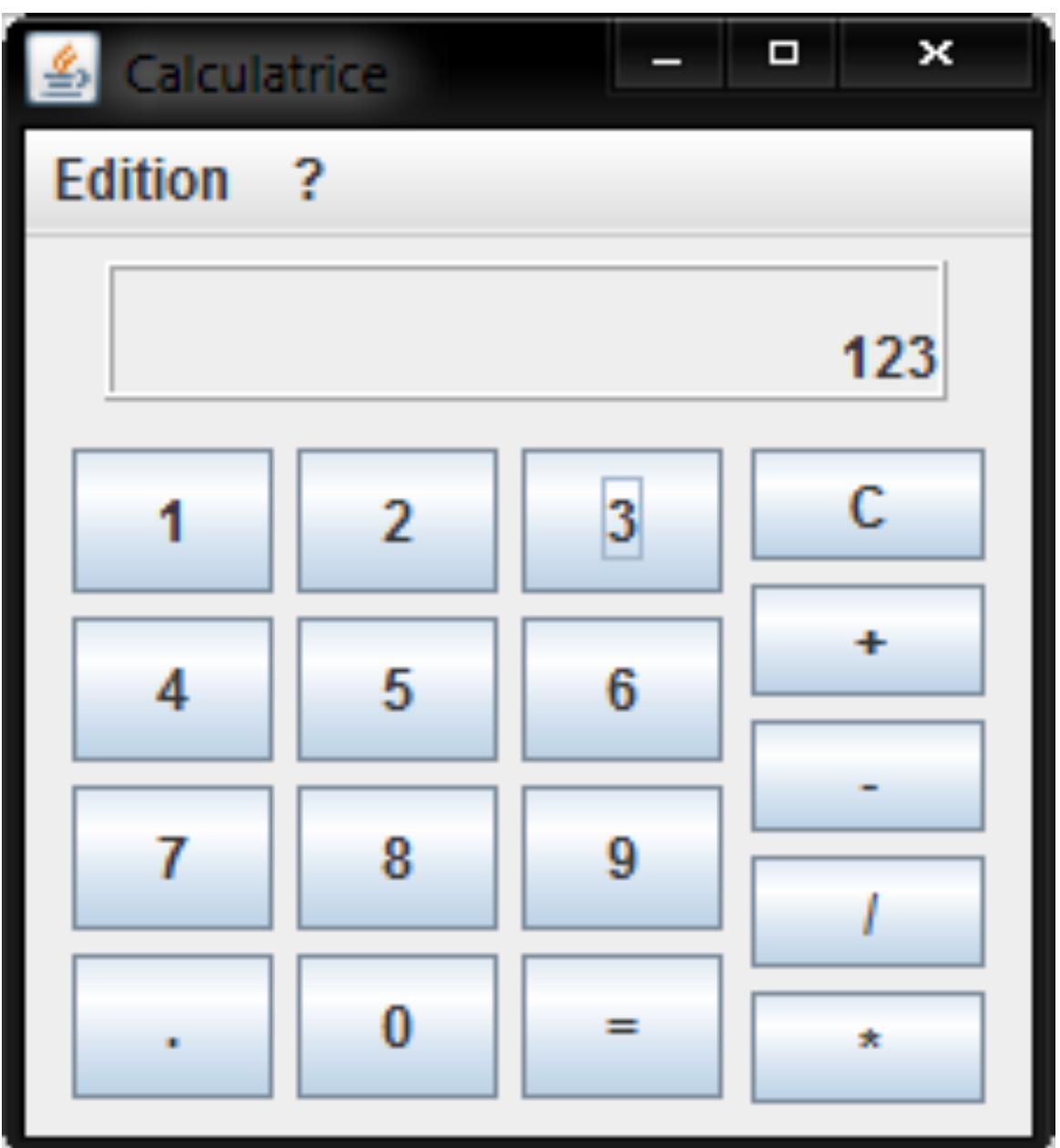
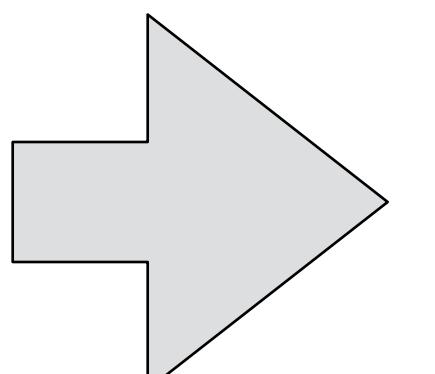


```
function date_mysql2ger($dateex) {
    list($year, $month, $day) = explode("-", $dateex);
    return sprintf("%02d.%02d.%04d", $day, $month, $year);
}

function date_ger2mysql($dateex) {
    list($day, $month, $year) = explode("-", $dateex);
    return sprintf("%04d-%02d-%02d", $year, $month, $day);
}

function timestamp_mysql2ger($t) {
    return sprintf("%02d.%02d.%04d", substr($t, 6, 2),
        substr($t, 4, 2), substr($t, 0, 4));
}

function date_mysql2ger($dateex) {
    list($year, $month, $day) = explode("-", $dateex);
    return sprintf("%02d.%02d.%04d", $day, $month, $year);
}
```





Where has the
Software changed?



Closing the gap: *automated tests*



automated tests

=

version control





Two Challenges:

- 1) Noise**
- 2) Redundancy**

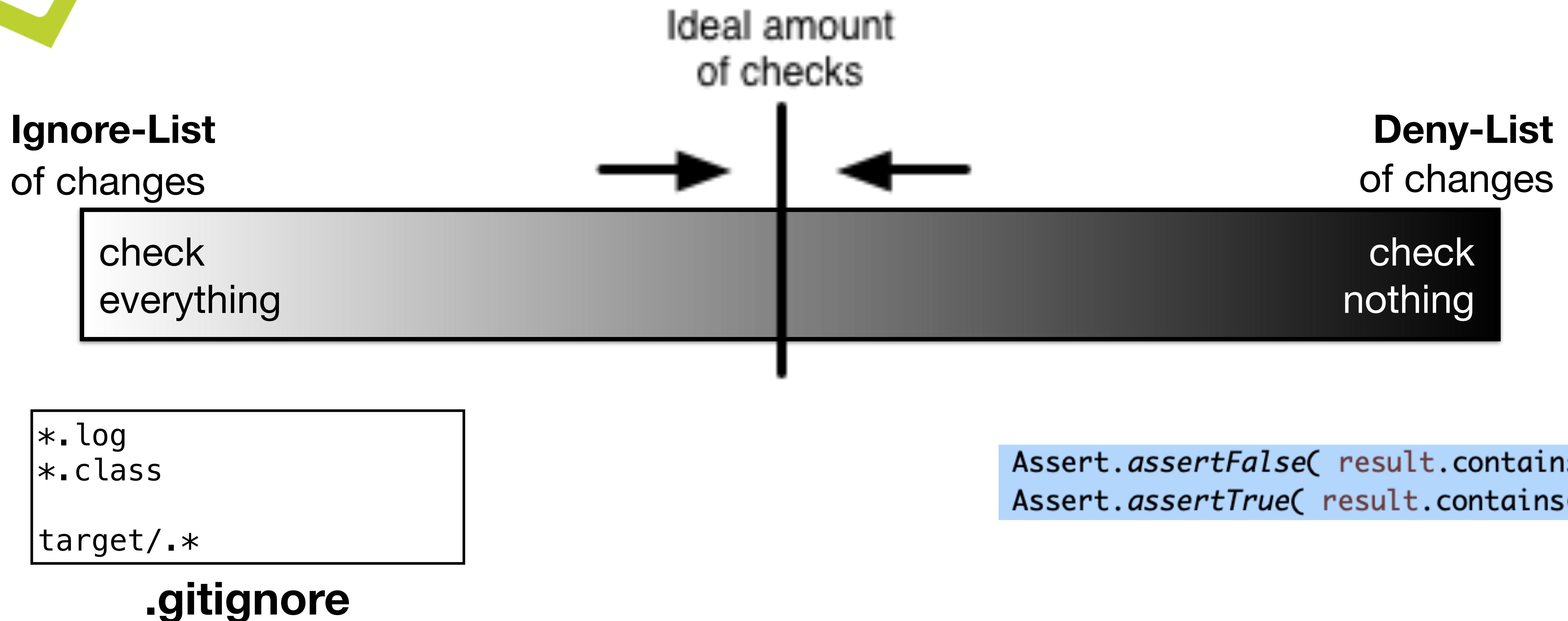
Naive



automated tests

=

version control





1

Multiple choice:



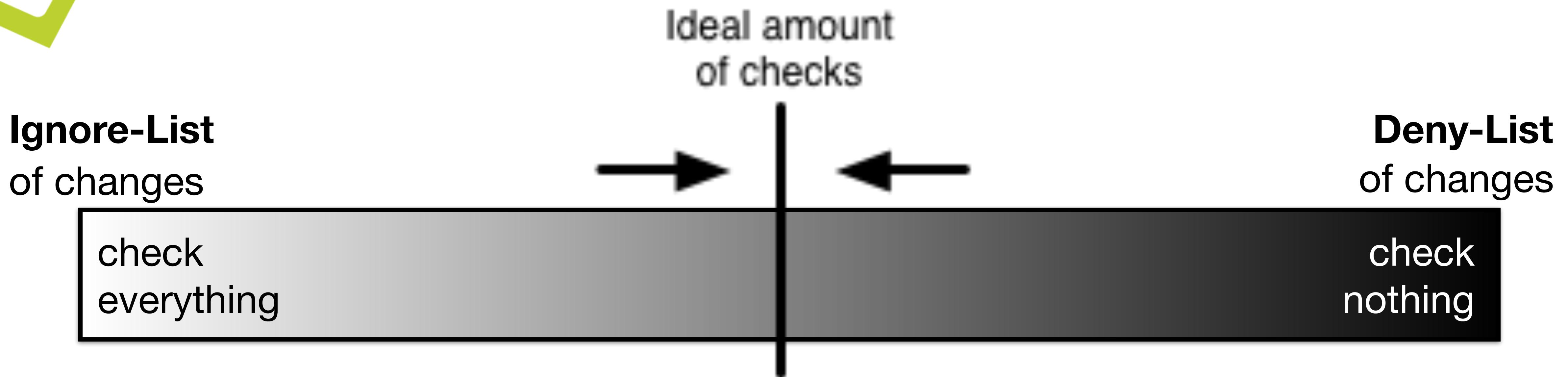
100 characters
remaining 

Randomize answer order for each respondent 

Randomize answer order for each respondent 

enter answer text









Git for the GUI



```
# These files are ignored by git  
# based on their name:
```

```
*.log
```

```
*.class
```

```
target/*
```

.gitignore



```
# Ignore file for recheck.
```

```
# Ignore attributes globally:  
attribute=class
```

.retest/recheck.ignore



```
# Ignore file for recheck.  
  
# Ignore attributes with regex:  
attribute-regex=.*
```

.retest/recheck.ignore



```
# Ignore file for recheck.
```

```
# Ignore elements by tag:  
matcher: type=iframe
```

.retest/recheck.ignore



```
# Ignore file for recheck.  
  
# Ignore elements by attribute:  
matcher: id=banner
```

.retest/recheck.ignore



```
# Ignore file for recheck.
```

```
# These can be combined:
```

```
matcher: id=ad, attribute-regex=font.*
```

.retest/recheck.ignore



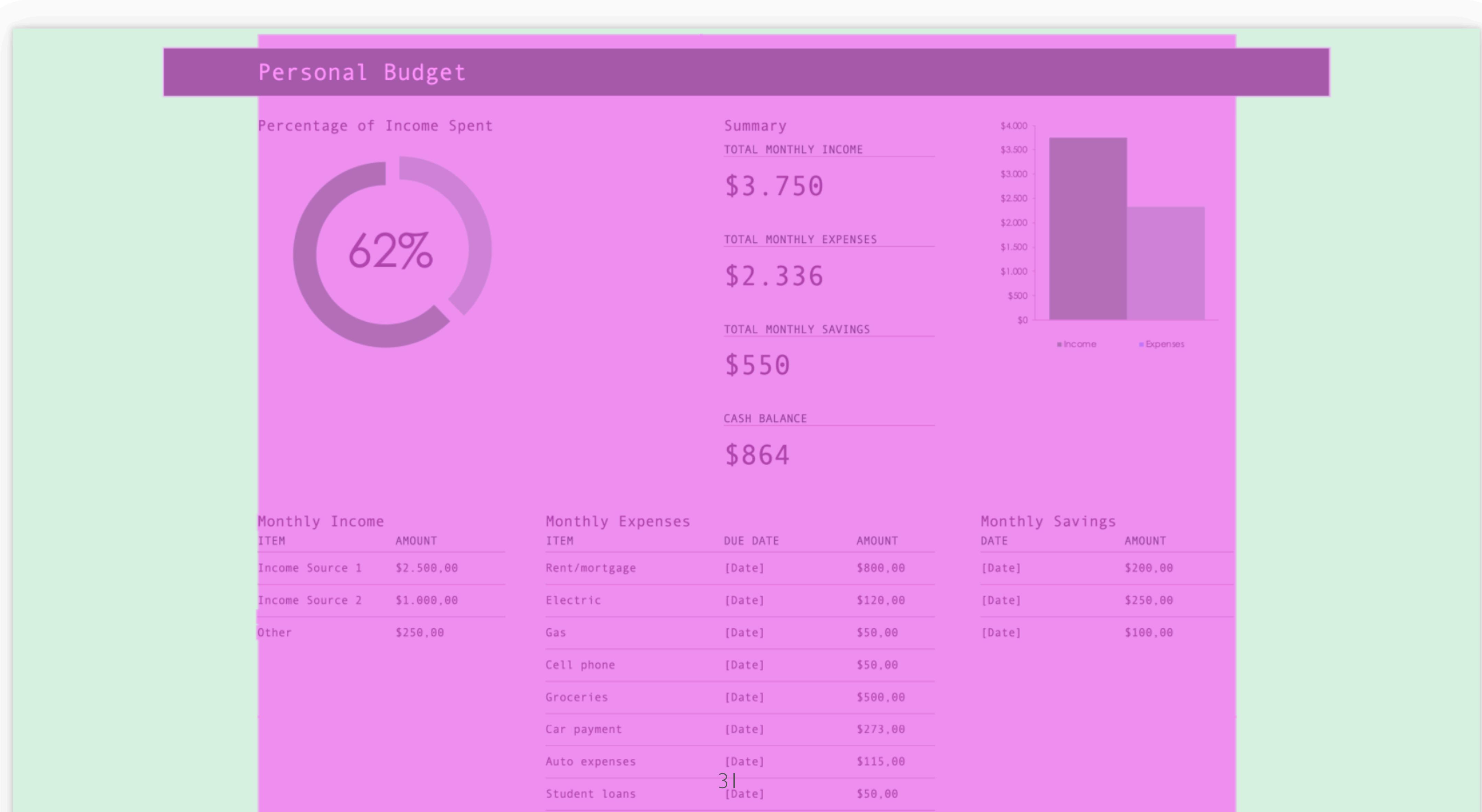
```
# Ignore file for recheck.  
  
# Ignore whole subtrees:  
matcher: xpath=HTML[1]/BODY[1]/DIV[1]/DIV[1]/DIV[1]
```

.retest/recheck.ignore



ignore ≈ filter

Better than pixel diffing

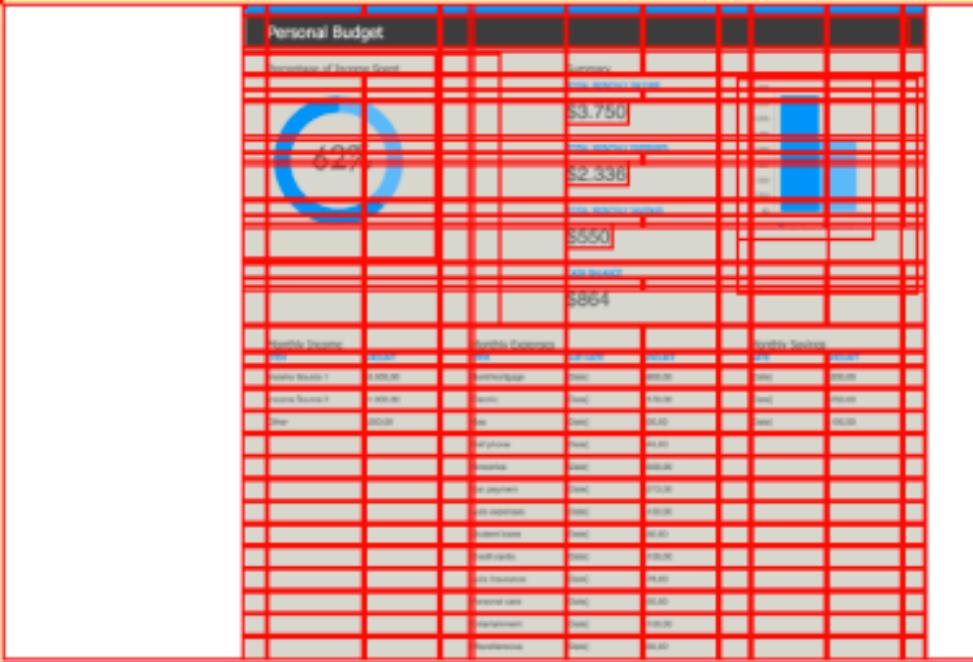


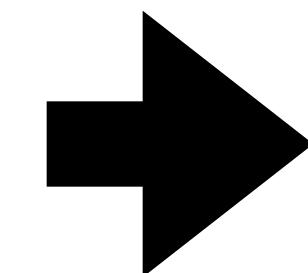


Demo

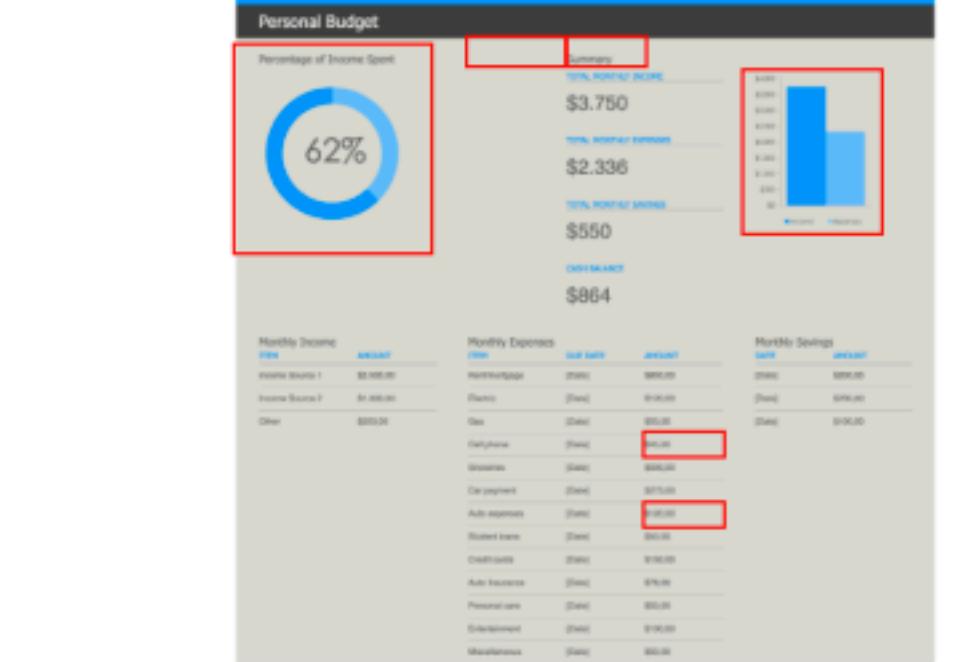
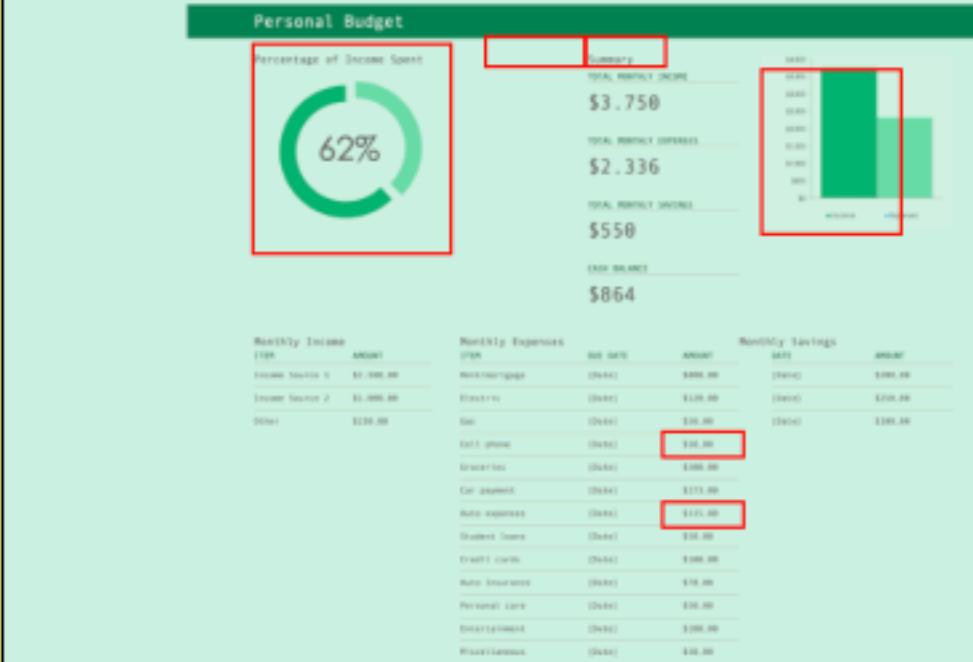
Better than pixel diffing

Difference in a state
Path: Budget[1]
Content: Budget

Expected		Actual																																									
																																											
<table border="1"> <thead> <tr> <th>UI element / Attribute</th> <th>Expected value</th> <th>Actual value</th> <th><input type="checkbox"/> Accept all</th> <th><input type="checkbox"/> Ignore all</th> </tr> </thead> <tbody> <tr> <td>html</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>outline</td> <td>height=1089</td> <td>height=1088</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> </tr> <tr> <td>body</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td> class</td> <td>xl70</td> <td>xl73</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> </tr> <tr> <td> background-color</td> <td>rgb(209, 243, 226)</td> <td>"Andale Mono", sans-...</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> </tr> <tr> <td> font-family</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>div</td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>				UI element / Attribute	Expected value	Actual value	<input type="checkbox"/> Accept all	<input type="checkbox"/> Ignore all	html					outline	height=1089	height=1088	<input type="checkbox"/>	<input type="checkbox"/>	body					class	xl70	xl73	<input type="checkbox"/>	<input type="checkbox"/>	background-color	rgb(209, 243, 226)	"Andale Mono", sans-...	<input type="checkbox"/>	<input type="checkbox"/>	font-family					div				
UI element / Attribute	Expected value	Actual value	<input type="checkbox"/> Accept all	<input type="checkbox"/> Ignore all																																							
html																																											
outline	height=1089	height=1088	<input type="checkbox"/>	<input type="checkbox"/>																																							
body																																											
class	xl70	xl73	<input type="checkbox"/>	<input type="checkbox"/>																																							
background-color	rgb(209, 243, 226)	"Andale Mono", sans-...	<input type="checkbox"/>	<input type="checkbox"/>																																							
font-family																																											
div																																											



Difference in a state
Path: Budget[1]
Content: Budget

Expected		Actual																																									
																																											
<table border="1"> <thead> <tr> <th>UI element / Attribute</th> <th>Expected value</th> <th>Actual value</th> <th><input type="checkbox"/> Accept all</th> <th><input type="checkbox"/> Ignore all</th> </tr> </thead> <tbody> <tr> <td>td [\$120,00]</td> <td>\$120,00</td> <td>\$115,00</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> </tr> <tr> <td> text</td> <td></td> <td></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> </tr> <tr> <td>td []</td> <td>td []</td> <td>missing</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> </tr> <tr> <td> img</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td> v:shapes</td> <td>chtIncomeExpenses</td> <td>chtIncomePct</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> </tr> <tr> <td> td [\$45,00]</td> <td>\$45,00</td> <td>\$50,00</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> </tr> <tr> <td> text</td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>				UI element / Attribute	Expected value	Actual value	<input type="checkbox"/> Accept all	<input type="checkbox"/> Ignore all	td [\$120,00]	\$120,00	\$115,00	<input type="checkbox"/>	<input type="checkbox"/>	text			<input type="checkbox"/>	<input type="checkbox"/>	td []	td []	missing	<input type="checkbox"/>	<input type="checkbox"/>	img					v:shapes	chtIncomeExpenses	chtIncomePct	<input type="checkbox"/>	<input type="checkbox"/>	td [\$45,00]	\$45,00	\$50,00	<input type="checkbox"/>	<input type="checkbox"/>	text				
UI element / Attribute	Expected value	Actual value	<input type="checkbox"/> Accept all	<input type="checkbox"/> Ignore all																																							
td [\$120,00]	\$120,00	\$115,00	<input type="checkbox"/>	<input type="checkbox"/>																																							
text			<input type="checkbox"/>	<input type="checkbox"/>																																							
td []	td []	missing	<input type="checkbox"/>	<input type="checkbox"/>																																							
img																																											
v:shapes	chtIncomeExpenses	chtIncomePct	<input type="checkbox"/>	<input type="checkbox"/>																																							
td [\$45,00]	\$45,00	\$50,00	<input type="checkbox"/>	<input type="checkbox"/>																																							
text																																											

Works with animations

Review test report

review

Test report: 39e042b9-ea02-40e1-b0f1-73b91f5e6f66.report?X-Amz-

Open...

Hide the following differences:

Positioning
 Style Attributes
 Invisible Attributes

Hide accepted/ignored differences

▼ Suite: Have multiple suites with rech...
▼ Test: Have multiple tests with rec...
▼ Differing Animated Text
 animation-d... animation-d...
 style = anim... style = anim...
text = Lorem.. text = Lorem..

Difference in a state

Path: Differing Animated Text[1]

Content: Differing Animated Text

Expected

The Caterpillar and Alice looked at each other for some time in silence; at last the Caterpillar took the hookah out of its mouth, and addressed her in a languid, sleepy voice.
This is yet another animated text. I have no idea why that would make any sense, but nonetheless, here it comes!
Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua.
Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat.
Ut wisi enim ad minim veniam, quis nostrud exerci tatione non流逝。
Nam liber tempor cum soluta nobis eleifend option congue nihil imperdiet doming id quod maxime placet facer possim assum.
Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis.
At vero eos et accusamus et iusto duo dolores et ea rebum. Sed clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.
Consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua.

Actual

The Caterpillar and Alice looked at each other for some time in silence; at last the Caterpillar took the hookah out of its mouth, and addressed her in a languid, sleepy voice.
This is yet another animated text. I have no idea why that would make any sense, but nonetheless, here it comes! + comment
Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua.
Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis.
At vero eos et accusamus et iusto duo dolores et ea rebum. Sed clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.
Consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua.

UI element / Attribute	Expected value	Actual value
▼ p [This is yet another animated text. I have no idea why that would make any sense, but nonetheless, here it comes!] animation-delay style	0.2s animation-delay: 0.2s;	0.5s animation-delay: 0.5s;
▼ p [Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua.] text	...r sadipscing elitr, sed diam nonumy eirmodr sadipscing elitr, ut labore et dolore magna aliquyam erat, sed diam voluptua.	...r sadipscing elitr, sed diam nonumy eirmodr sadipscing elitr, ut labore et dolore magna aliquyam erat, sed diam voluptua.



Redundancy

Tool Support

A screenshot of a terminal window with a light gray header bar. The header bar features three colored circles (red, yellow, green) on the left and the text "roessler — -bash — 61x8" on the right. The main window has a black background and displays the following text in white:

```
Last login: Tue Oct 15 15:30:18 on ttys002
bonum:~ roessler$ recheck commit BudgetTest.report --all
```

Tool Support



Review test report

Test report: BudgetTest.report

Open...

Hide the following differences:

- Positioning
- Style Attributes
- Invisible Attributes

Hide accepted/ignored differences

Suite: de.retest.recheck.example.BudgetTest

Test: java.util.Collections\$EmptyListTest

open

text = \$120,... text = \$115,...

deleted: td []

src = OriginalBudget.fld src = AdaptedBudget.fld

text = \$45,00 text = \$50,00

src = OriginalBudget.fld src = AdaptedBudget.fld

inserted: col

Golden Master

Personal Budget

Category	Original	Adapted
Total Monthly Income	\$3,750	\$3,750
Total Monthly Expenses	\$2,336	\$2,336
Total Monthly Savings	\$550	\$550
Current Balance	\$864	\$864

Actual

Personal Budget

Category	Original	Adapted
Total Monthly Income	\$3,750	\$3,750
Total Monthly Expenses	\$2,336	\$2,336
Total Monthly Savings	\$550	\$550
Current Balance	\$864	\$864

Path to Golden Master file: src/test/resources/retest/recheck/de.retest.recheck.example.BudgetTest/excel/open.recheck

UI element / Attribute	Expected value	Actual value	<input type="checkbox"/> Accept all	<input type="checkbox"/> Ignore all
td [\$120,00]				
text	\$120,00	\$115,00	<input checked="" type="checkbox"/>	<input type="checkbox"/>
td []	td []	missing	<input checked="" type="checkbox"/>	<input type="checkbox"/>
img		OriginalBudget.fld/Original...AdaptedBudget.fld/Ada...	<input type="checkbox"/>	<input checked="" type="checkbox"/>
src				
td [\$45,00]				
text	\$45,00	\$50,00	<input checked="" type="checkbox"/>	<input type="checkbox"/>
img		OriginalBudget.fld/Original...AdaptedBudget.fld/Ada...	<input type="checkbox"/>	<input checked="" type="checkbox"/>
src				
col		col	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Apply changes

roesslerj

```
@Before
public void setup() {
    driver = new ChromeDriver();
}

@Test
public void check_formPage() throws Exception {
    // To see differences, simply switch to formPage-changed.html below
    String url = Paths.get("src/test/resources/formPage.html").toUri().toURL().toString();
    driver.get(url);

    driver.findElement(By.id("email")).sendKeys("Emil");
    driver.findElement(By.id("age")).sendKeys("16");
    driver.findElement(By.name("login")).submit();

    Assert.assertEquals(driver.findElement(By.id("email")).getText(), "");
}

@After
public void tearDown() throws InterruptedException {
    driver.quit();
}
```

new version

The screenshot shows a login form with two text input fields: 'Username' and 'Password', and a 'Log in' button.

Username: Enter your username

Password: (redacted)

Log in

**find in
old version**



`findElement(by.id("sign-in"))`

breaks

ID changed

old version

The screenshot shows a login form with three text input fields: 'Username', 'Password', and 'Sign in', and a 'Log in' button.

Username: Enter your username

Password: (redacted)

Sign in

Log in

**create 1-on-1
assignment**

**use
new element
and log**

best match



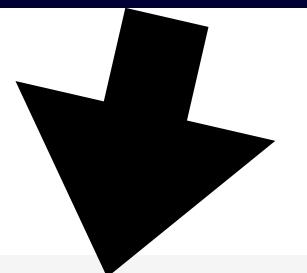
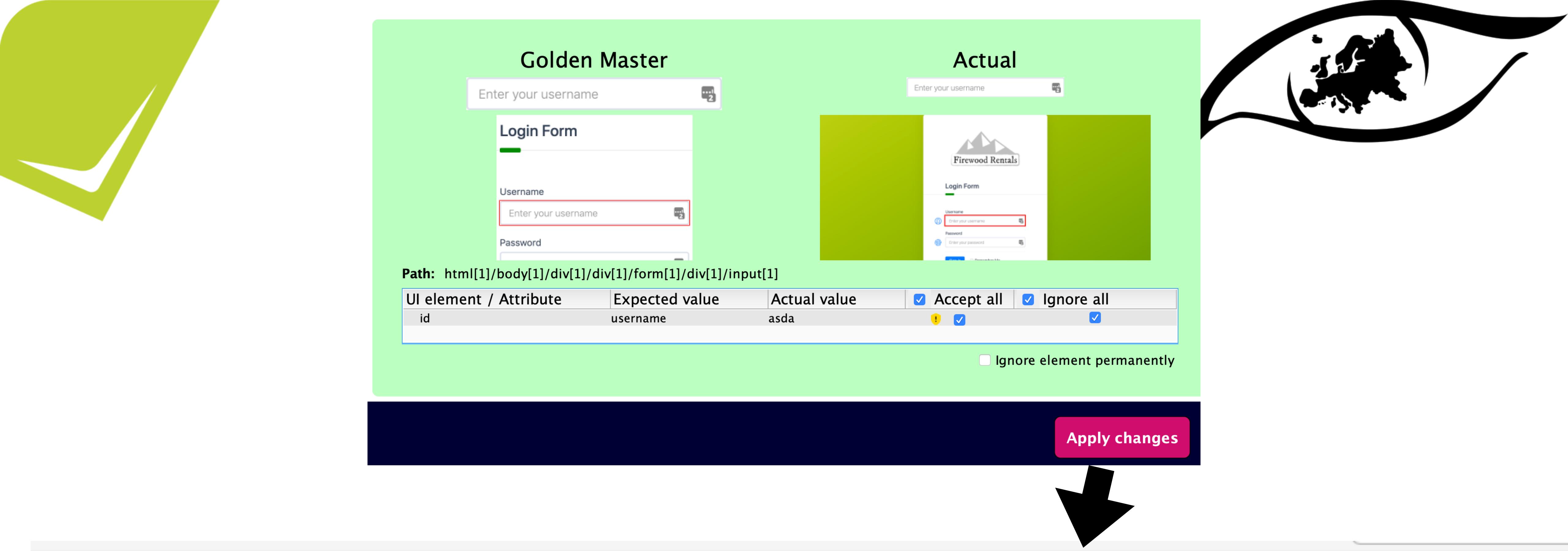
***** recheck warning *****

The HTML id attribute used for element identification changed from 'username' to 'asda'.

retest identified the element based on the persisted Golden Master.

If you apply these changes to the Golden Master , your test de.retest.recheck.example.MyUnbreakableSeleniumTest will break.

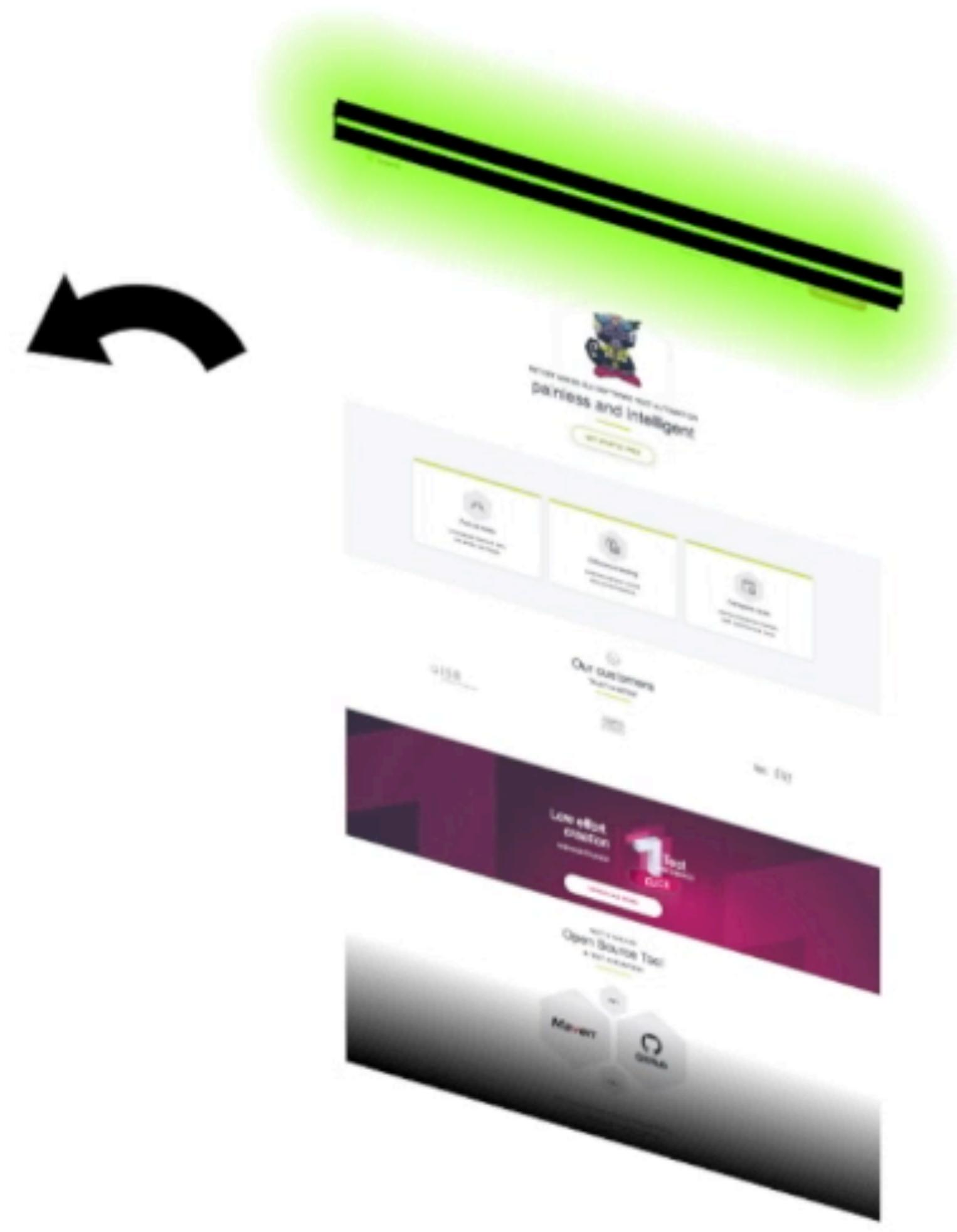
Use `By.id("asda")` or `By.retestId("username-1")` to update your test MyUnbreakableSeleniumTest.java:27.

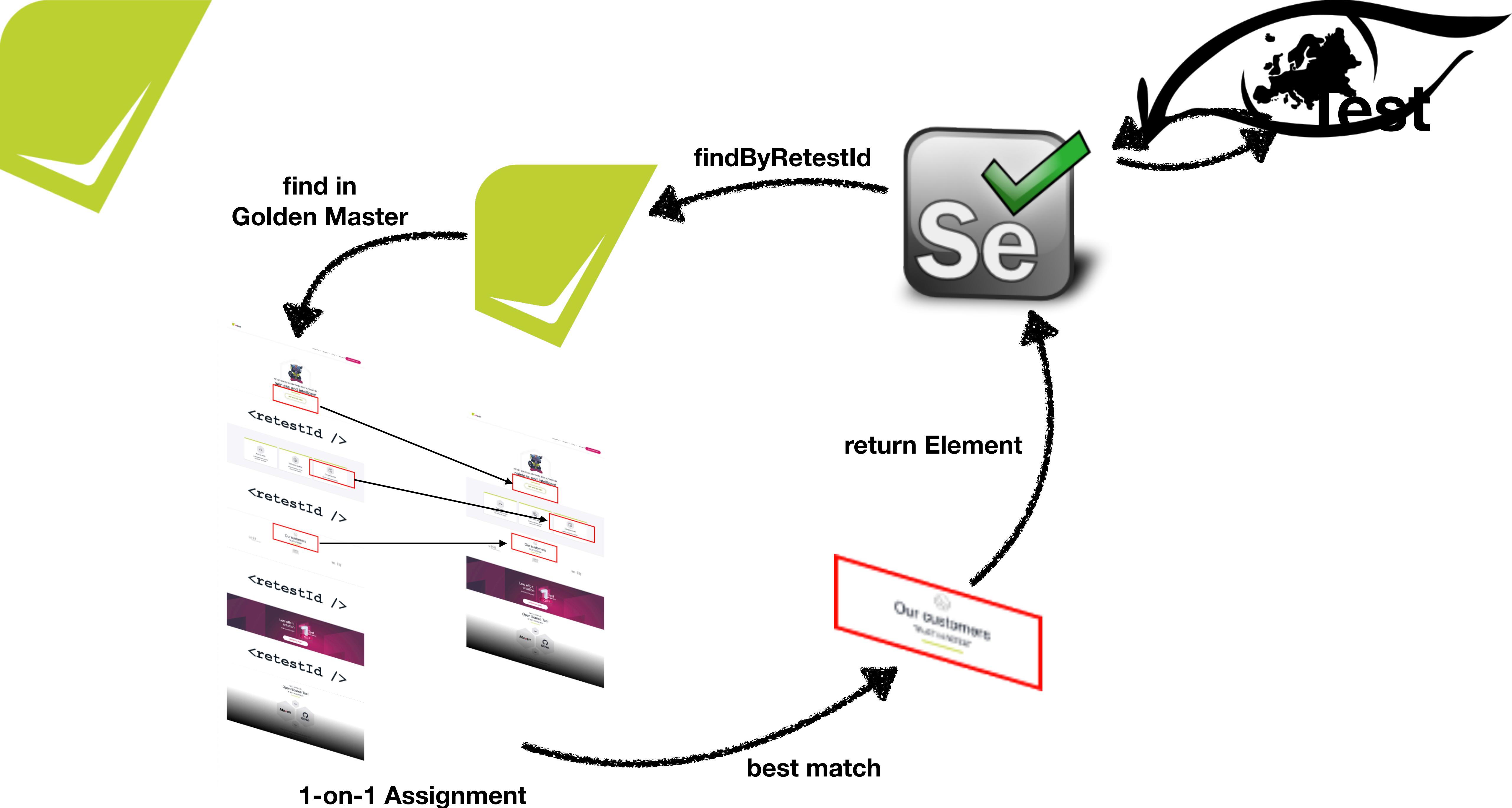


... src/test/java/de/retest/recheck/example/MyUnbreakableSeleniumTest.java

Hunk 1 : Lines 27-29

27	27	
28		- driver.findElement(By.id("username")).sendKeys("Simon");
28		+ driver.findElement(By.id("asda")).sendKeys("Simon");
29	29	driver.findElement(By.id("password")).sendKeys("secret");





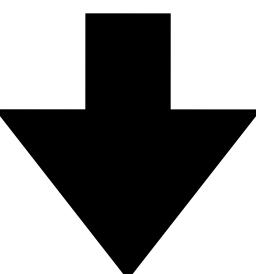
Reason number 34986 why I hate selenium. People write code like this and are proud of it: result =
self.driver.find_element_by_css_selector('#root
> div > ul > li > div > p')



Mr. Weasel @alanpage

1:06am - 16 May 2019

```
<containedElements retestId="email">
  <identifyingAttributes>
    <attributes>
      <attribute key="absolute-outline" xsi:type="outlineAttribute">
        <x>8</x>
        <y>26</y>
        <height>19</height>
        <width>131</width>
      </attribute>
      <attribute key="id" xsi:type="stringAttribute">email</attribute>
      <attribute key="outline" xsi:type="outlineAttribute">
```



```
driver.findElement(By.retestId("email")).sendKeys("Emil");
driver.findElement(By.id("age")).sendKeys("16");
driver.findElement(By.name("login")).submit();
```



data-independent tests



recheck benefits:

- + rule-based deterministic ignore
- + open source
- + offline usage
- + unbreakable selenium
- + works for *any* technical interface



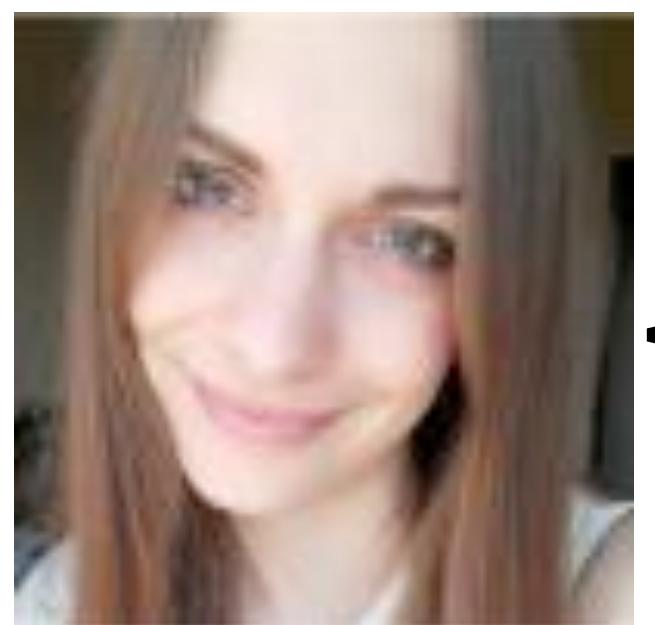
Michael Wowro

... I was thrilled by that product and new approach...



Ahmed Magdy

... let me congratulate you on your brilliant idea of the unbreakable Selenium and the web page difference checking solutions. I think this is a very much needed solution in the website testing field.



Karolina Wierbol

... the tool is very helpful, gives detailed information and is easy to use ...

some Feedback

<https://github.com/retest/recheck-web>



README.md

The screenshot shows the GitHub README page for the recheck project. At the top is the project logo, which consists of a green checkmark icon followed by the word "recheck" in blue and "for web apps" in black. Below the logo are several status badges: "build passing", "maven central 0.6.2", "license AGPL", "PRs welcome", and "</> with ❤️ by retest". A descriptive paragraph follows, stating: "recheck is a Golden Master-based test framework on top of Selenium that comes with powerful features. Check everything at once and create unbreakable tests." Below this is a code editor window displaying Java code for a test class named "SeleniumLargeTest.java". The code includes imports for Vertx, WebDriver, and Recheck, and defines a class with private fields for these dependencies.

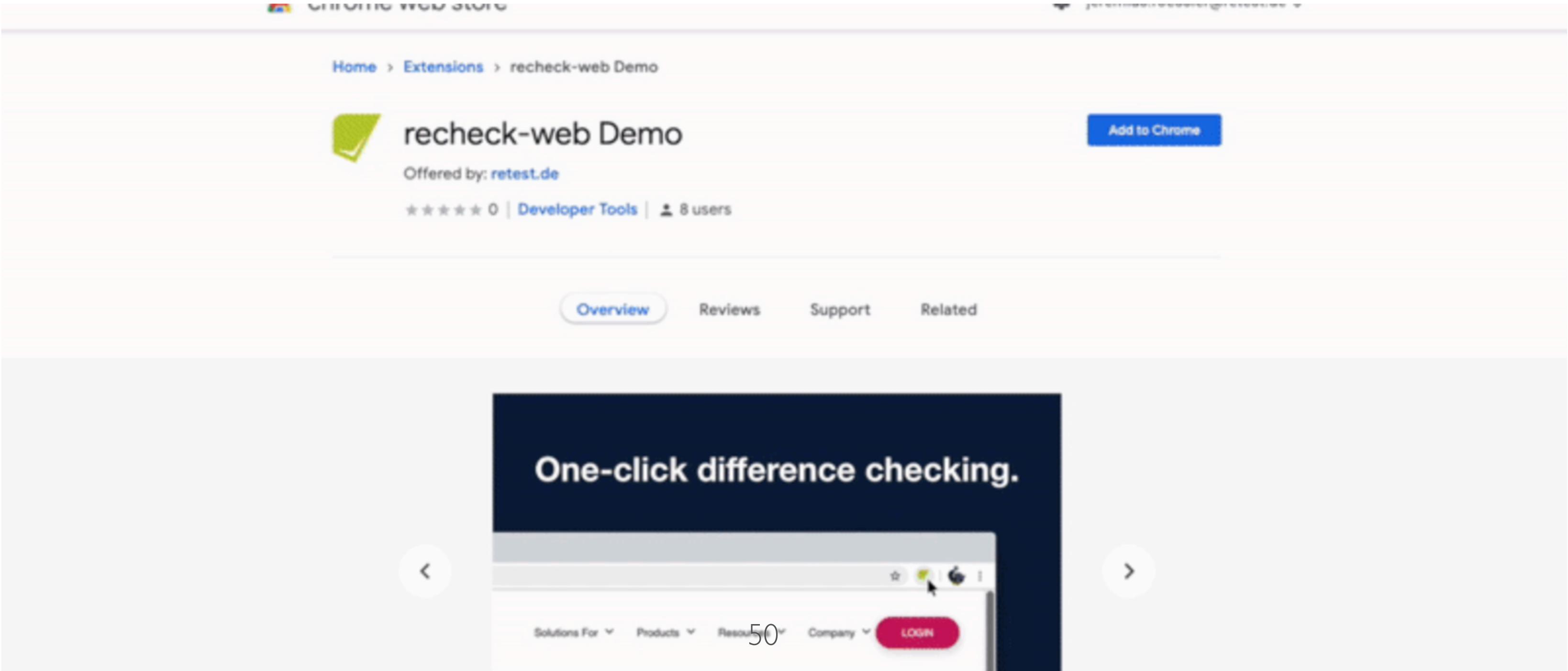
```
*SeleniumLargeTest.java ✘ ProductRepo.java
17 public class SeleniumLargeTest {
18
19     private Vertx vertx;
20     private WebDriver driver;
21     private Recheck re;
22 }
```

<https://github.com/retest/recheck-web-chrome-extension>

README.md Edit

recheck-web-chrome-extension

A Chrome extension that uses [recheck-web](#) to let you manually create 'checkpoints' of web sites. Using recheck-web together with the [recheck.cli](#) or the [review GUI](#), you can now easily find differences in websites. Using filters, you can easily differentiate between style, positioning or content differences—you can even define your own category!



The screenshot shows the Chrome Web Store page for the "recheck-web Demo" extension. The extension icon features a green checkmark inside a yellow square. The title is "recheck-web Demo" and it is offered by "retest.de". It has a rating of 0 stars and 8 users. A prominent blue button says "Add to Chrome". Below the main listing, there's a large thumbnail image showing a browser window with the text "One-click difference checking." and a screenshot of the extension's UI.

Overview Reviews Support Related

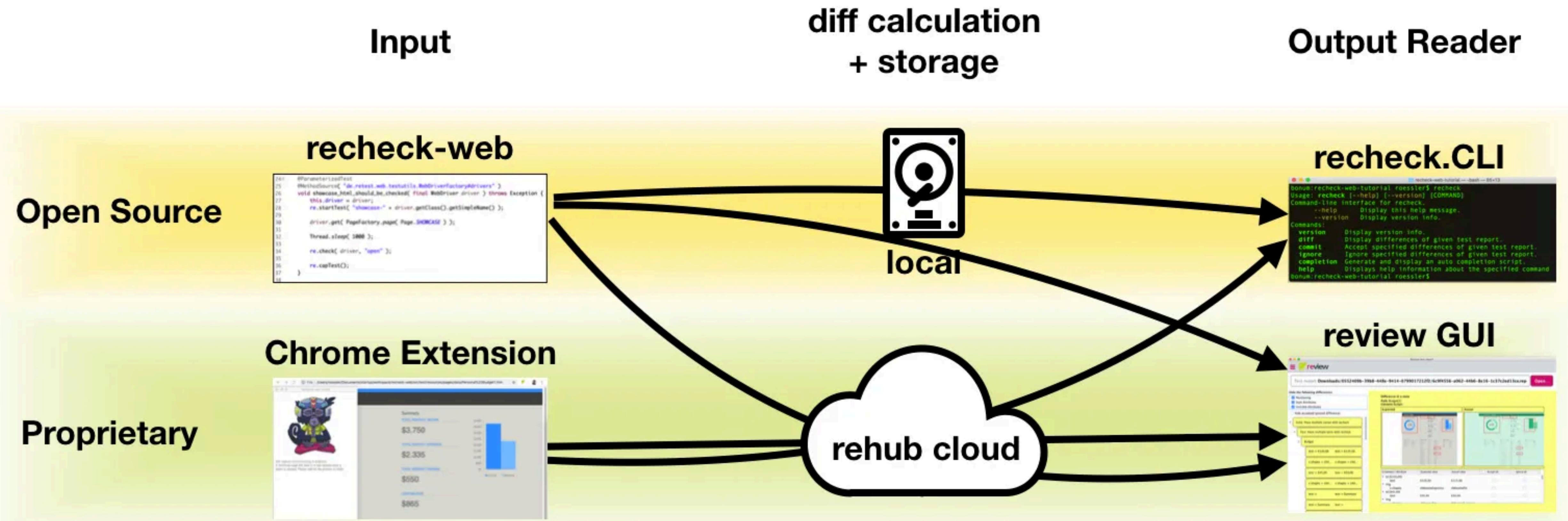
One-click difference checking.

Solutions For ▼ Products ▼ Resources ▼ Company ▼ LOGIN

50

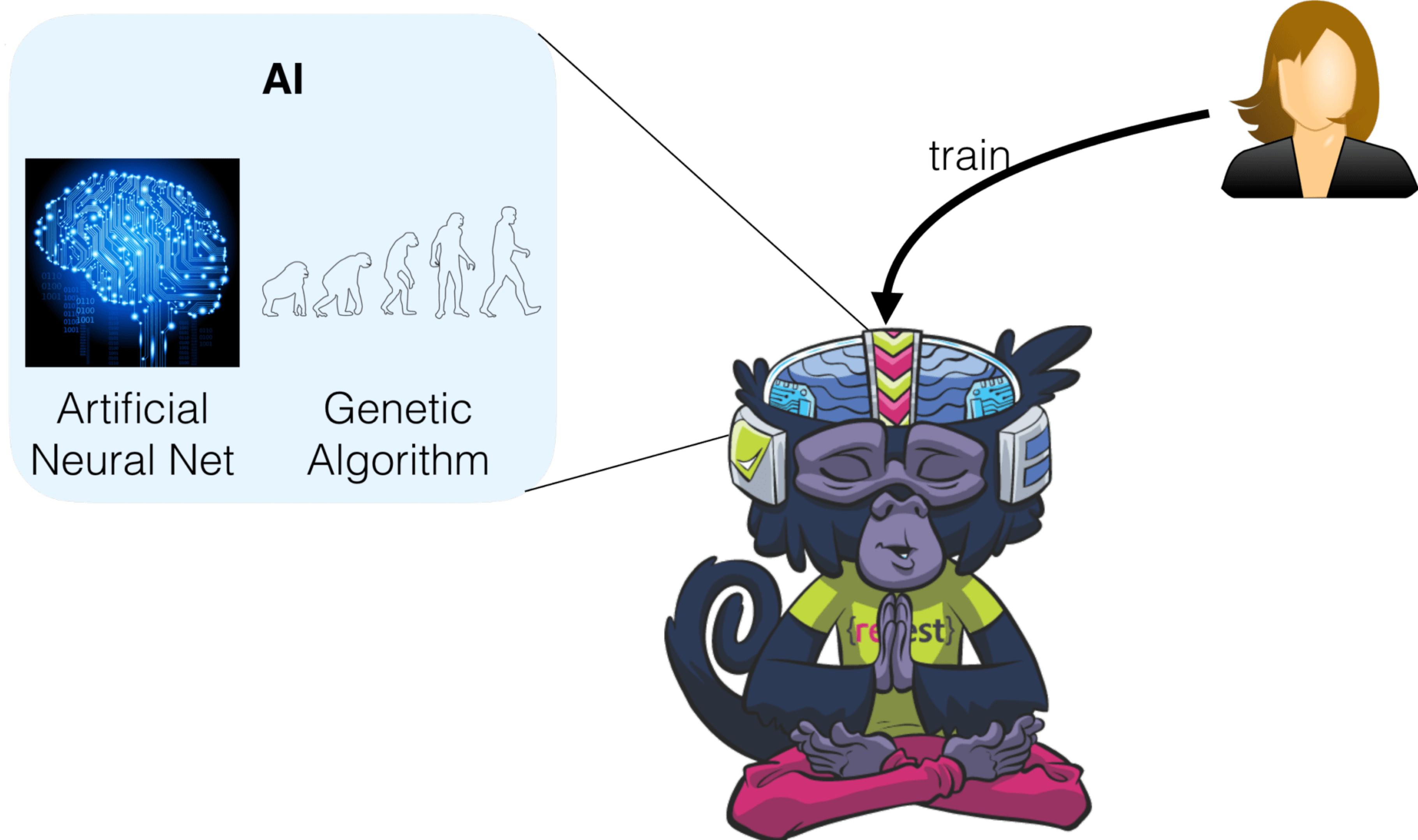
@roesslerj

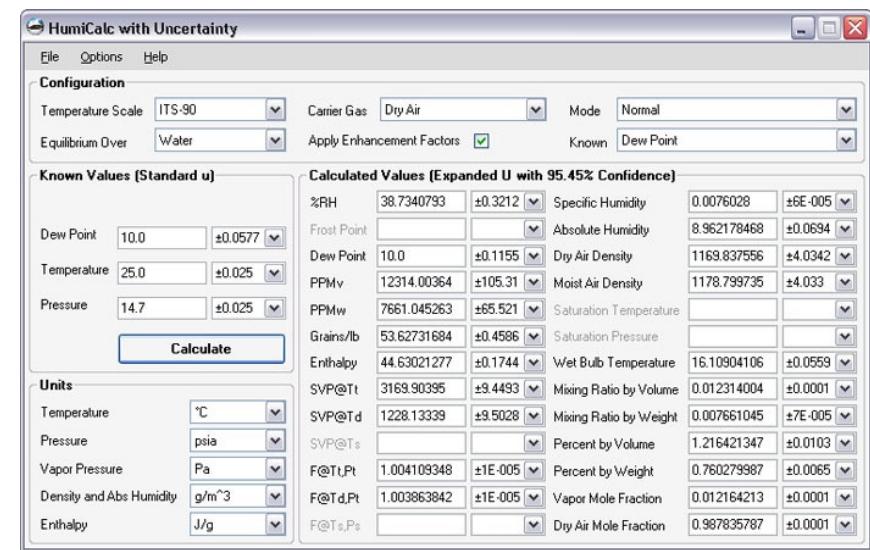
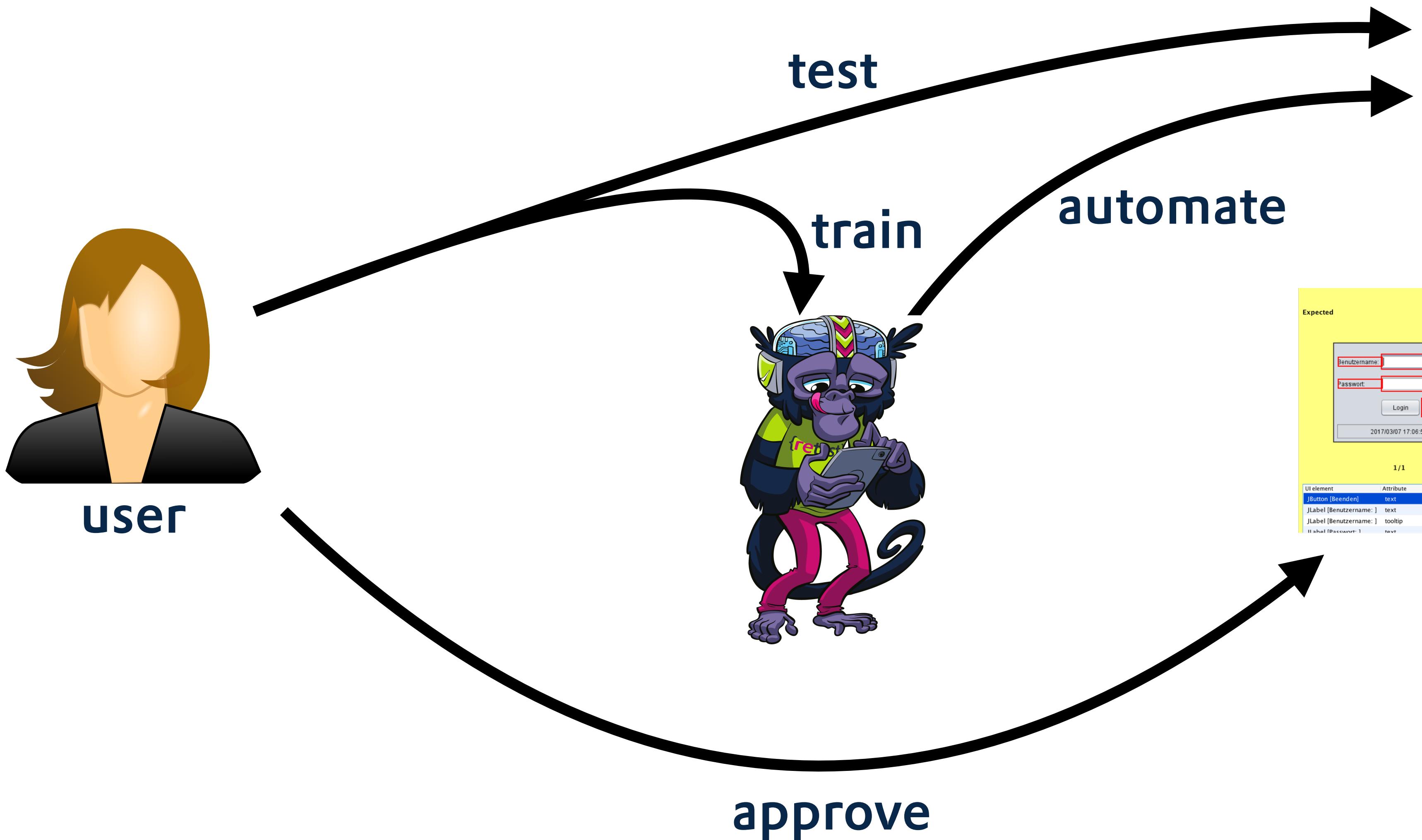
Our Business Model



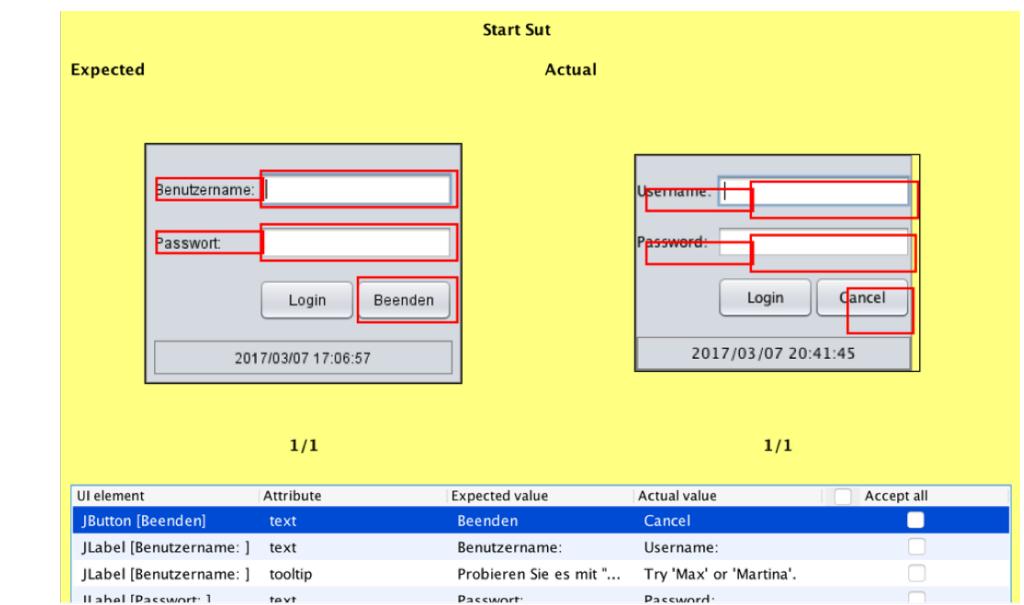
+AI

AI-based Test Generation

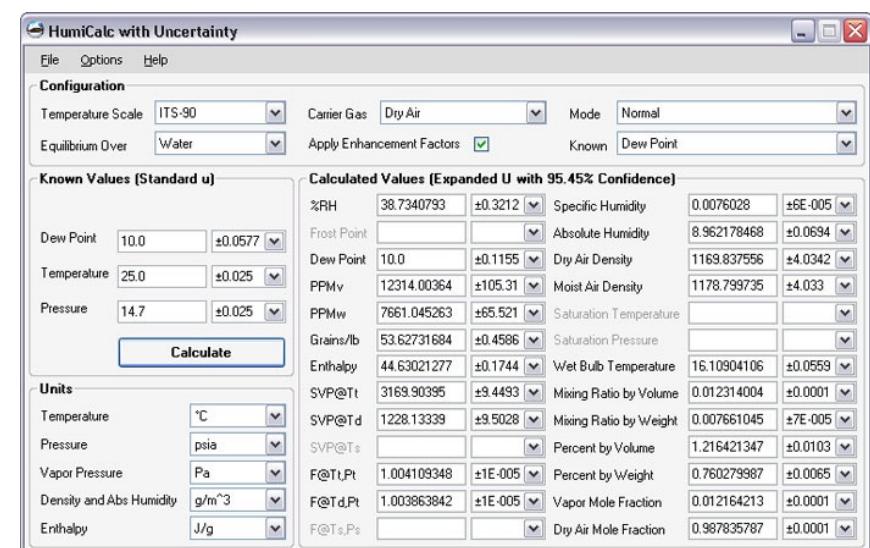




1.0



diff



1.1

Summary

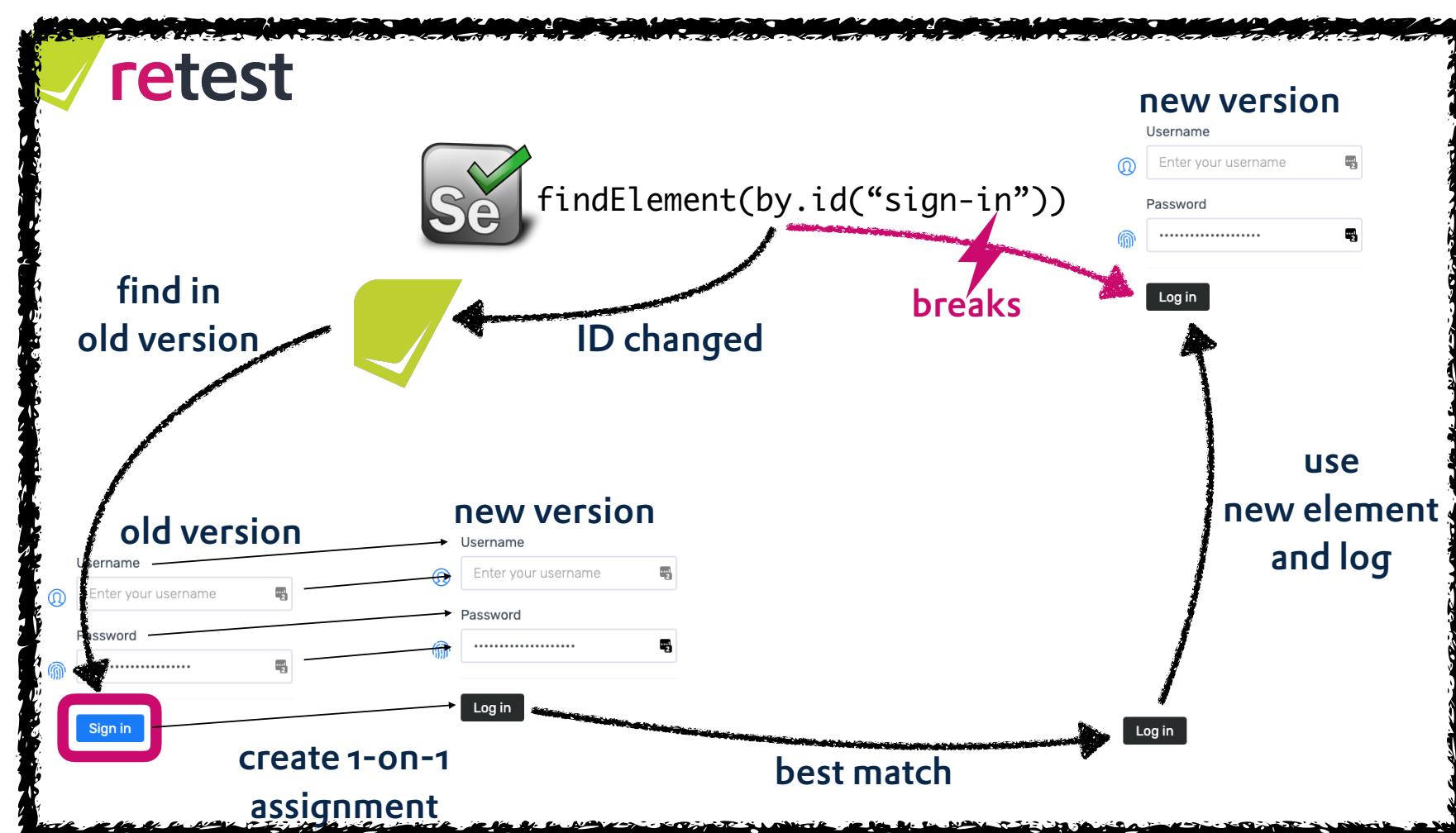


Better than pixel diffing

The screenshot shows two side-by-side UI snapshots labeled 'Expected' and 'Actual'. The 'Expected' version has a light blue background, while the 'Actual' version has a light green background. Numerous red highlights indicate differences between the two versions across various UI elements like tables and charts. Below each snapshot is a table comparing UI elements and their attributes, showing expected and actual values. A large black arrow points from the first comparison table to the second.

@roesslerj

Difference Testing
whitelist testing
easy creation
easy maintenance



Unbreakable Selenium:
easy to use
define own ids
data-independent tests



Please star us on GitHub!

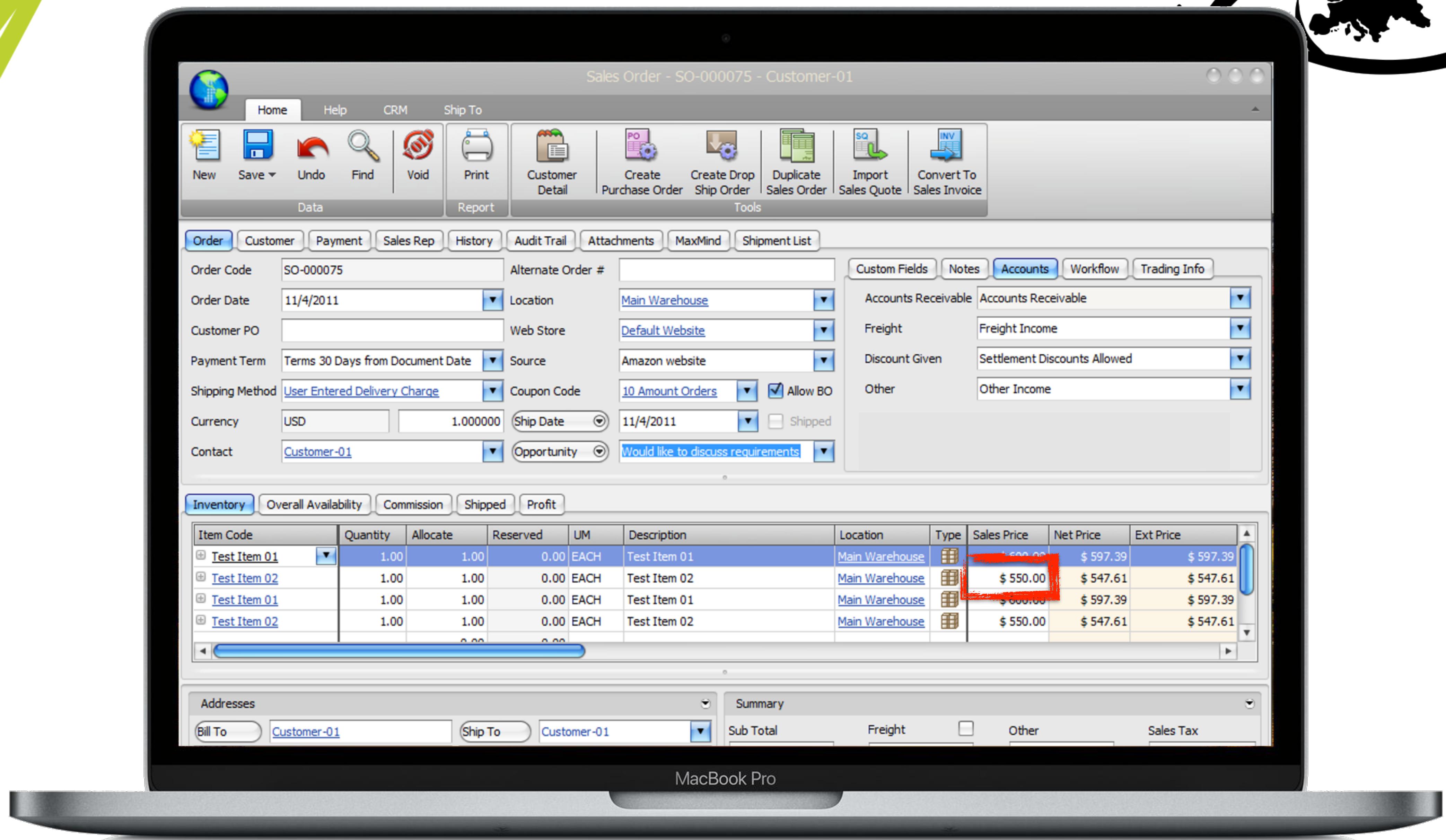


 @roesslerj

 roessler@retest.de

 <https://www.facebook.com/retest.de>

 <http://retest.rocks>

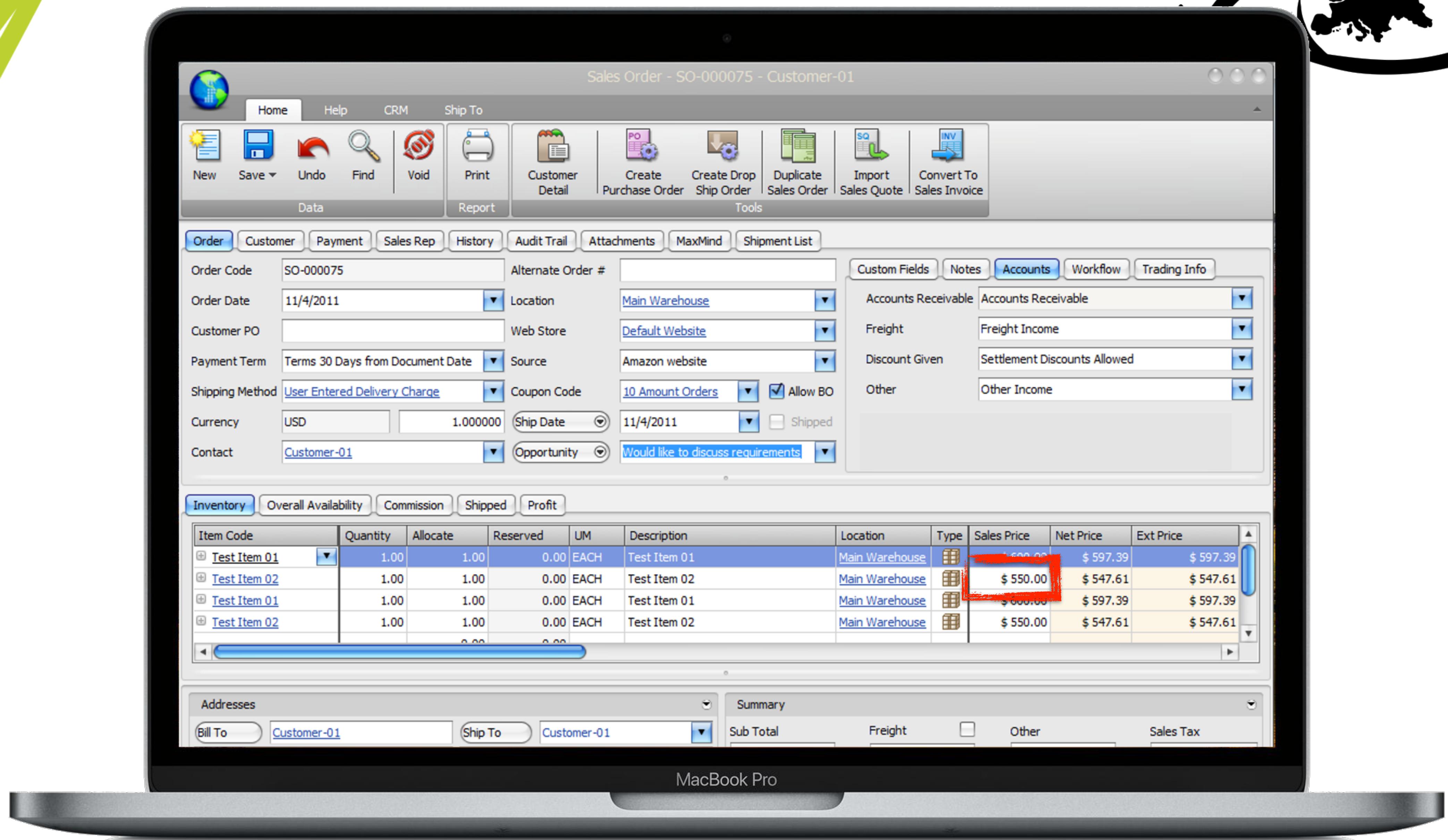




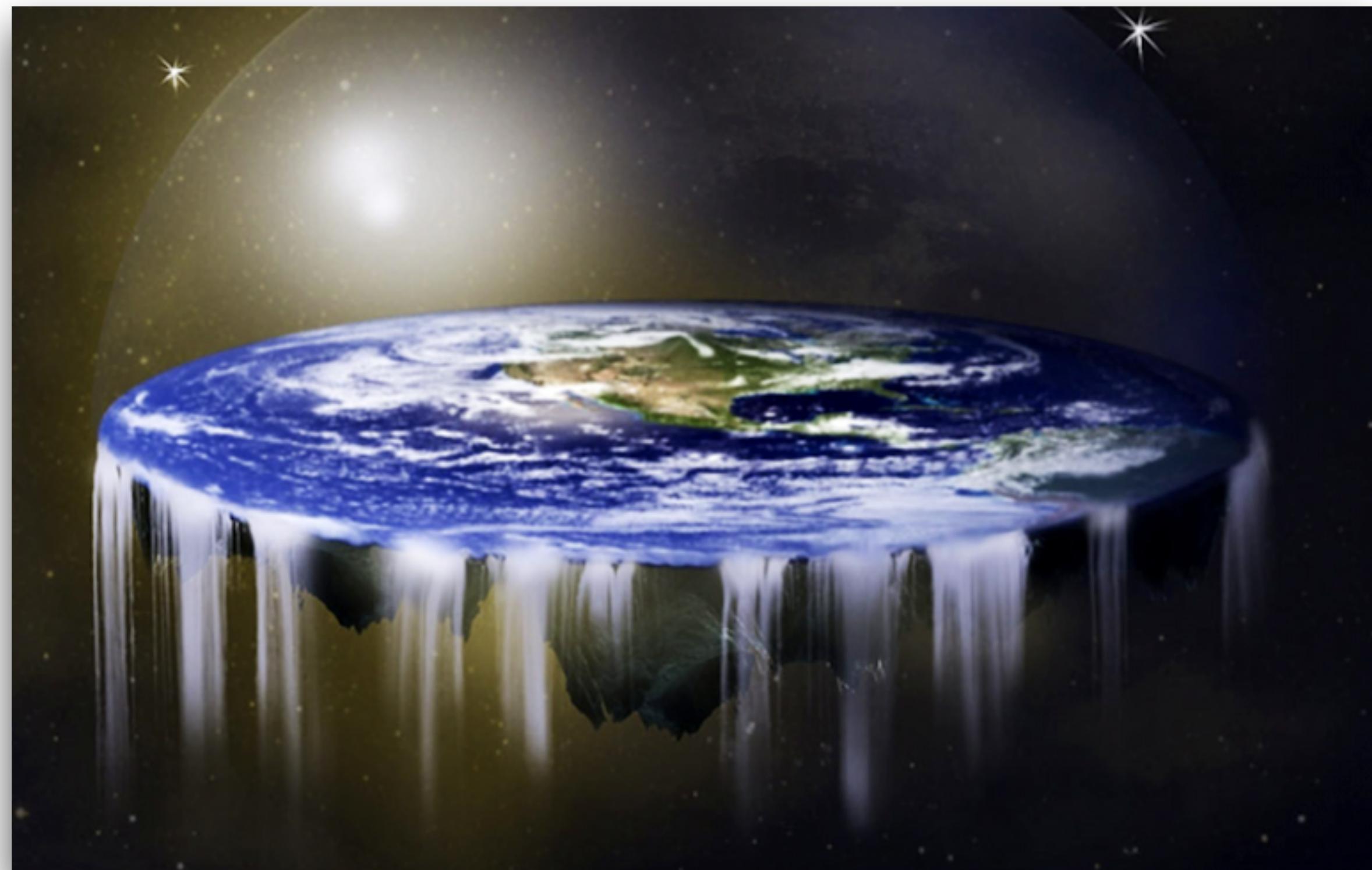
**ORACLE
PROBLEM**



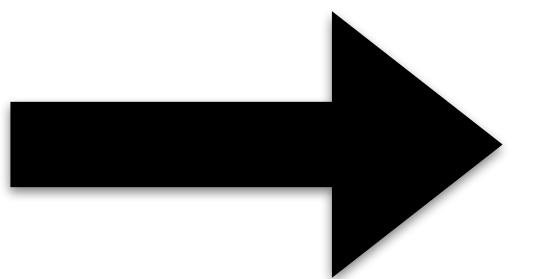
```
roessler — -bash — 64x7
Last login: Wed Mar 13 14:54:50 on ttys002
bonum:~ roessler$ Does God exist?|
```



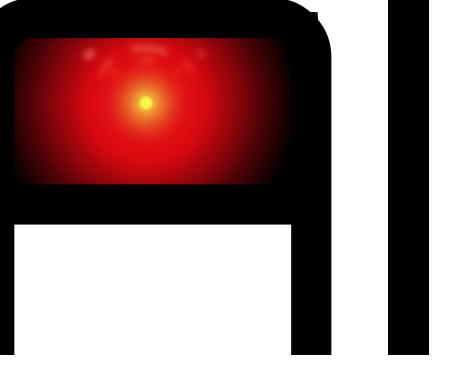
ORACLE PROBLEM



ORACLE PROPHECIES

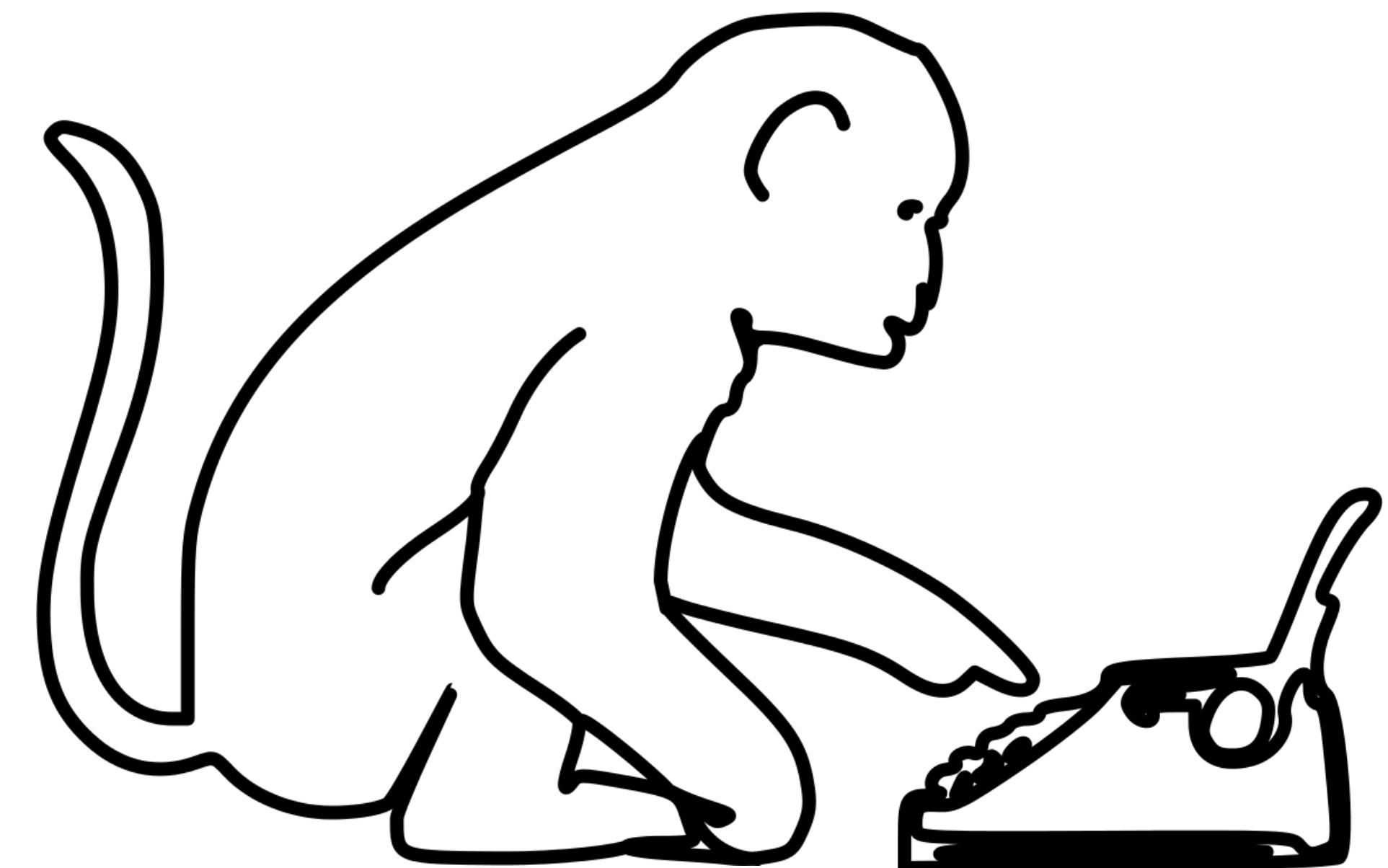




How do you bring  into testing? test automation?

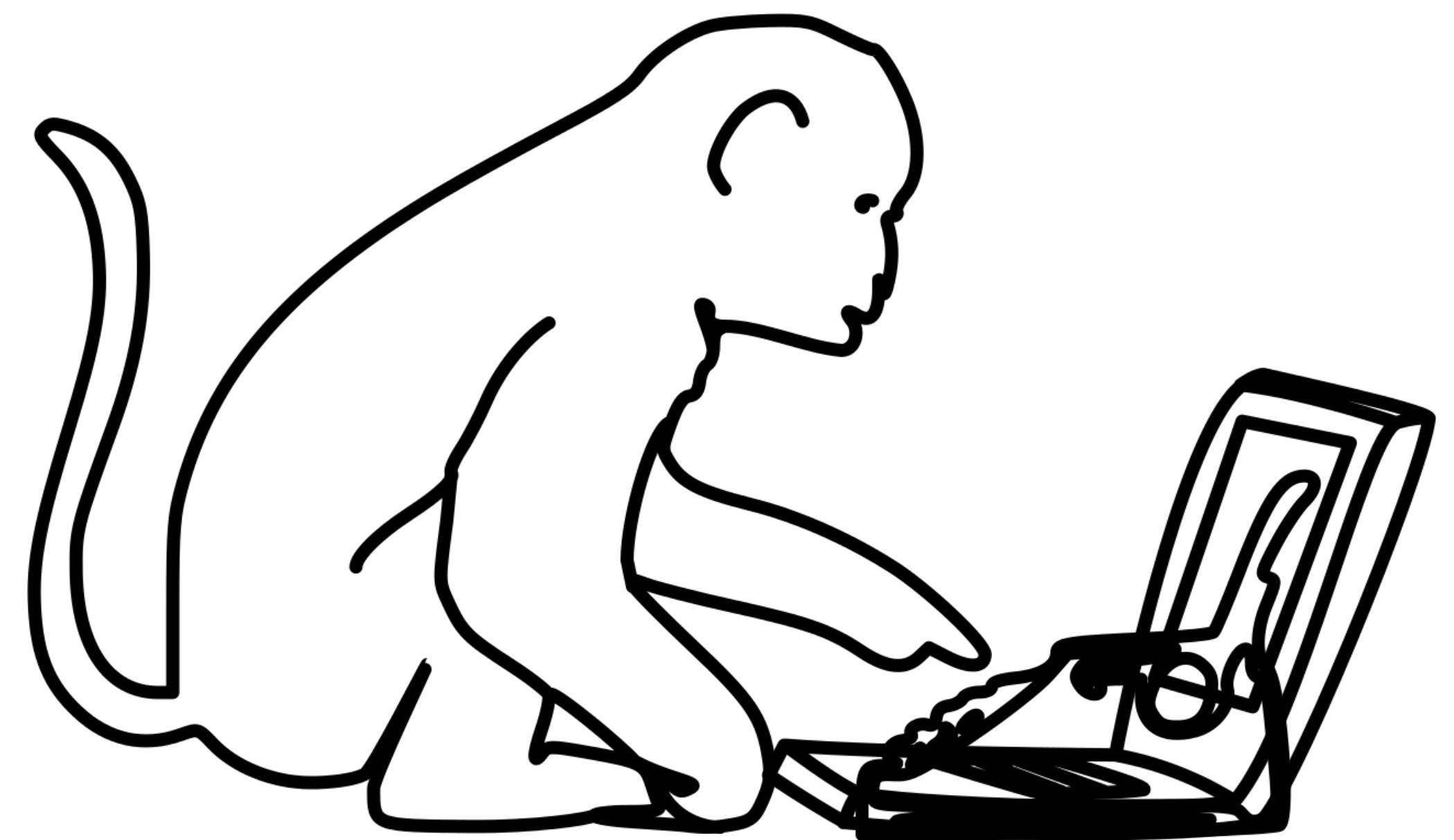


Infinite-Monkey-Theorem:
A monkey hitting keys at random on a typewriter
for a long enough time
will type the complete works of William Shakespeare.





We replace the typewriter with a computer...





```
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. }
```



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

Seitereffekte mehr ins Projekt kommen.

Robustheitstesten von Corner Cases

bekommen Sie gratis dazu.

Mit ReTest behalten Sie die Kontrolle!





- gremlins.js
- NetFlix Chaos Monkey
- UI/Application Exerciser Monkey







```
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.}
```



Intelligent Monkey

DEMO



Start ReTest News Kontakt

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
Seitereffekte mehr ins Projekt kommen.
Robustheitstesten von Corner Cases
bekommen Sie gratis dazu.
Mit ReTest behalten Sie die Kontrolle!





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

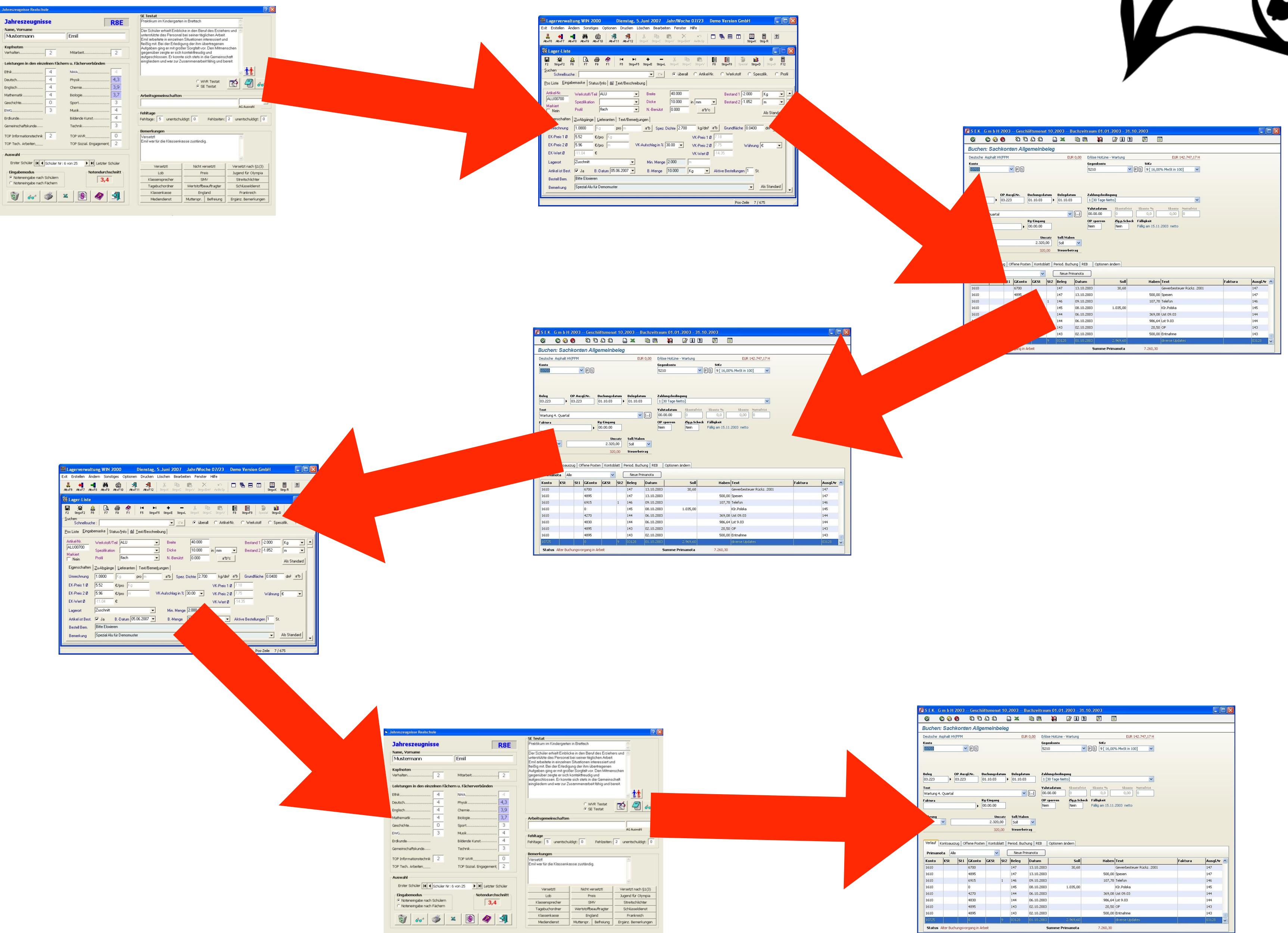
Calculated Values (Expanded U with 95.45% Confidence)

%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

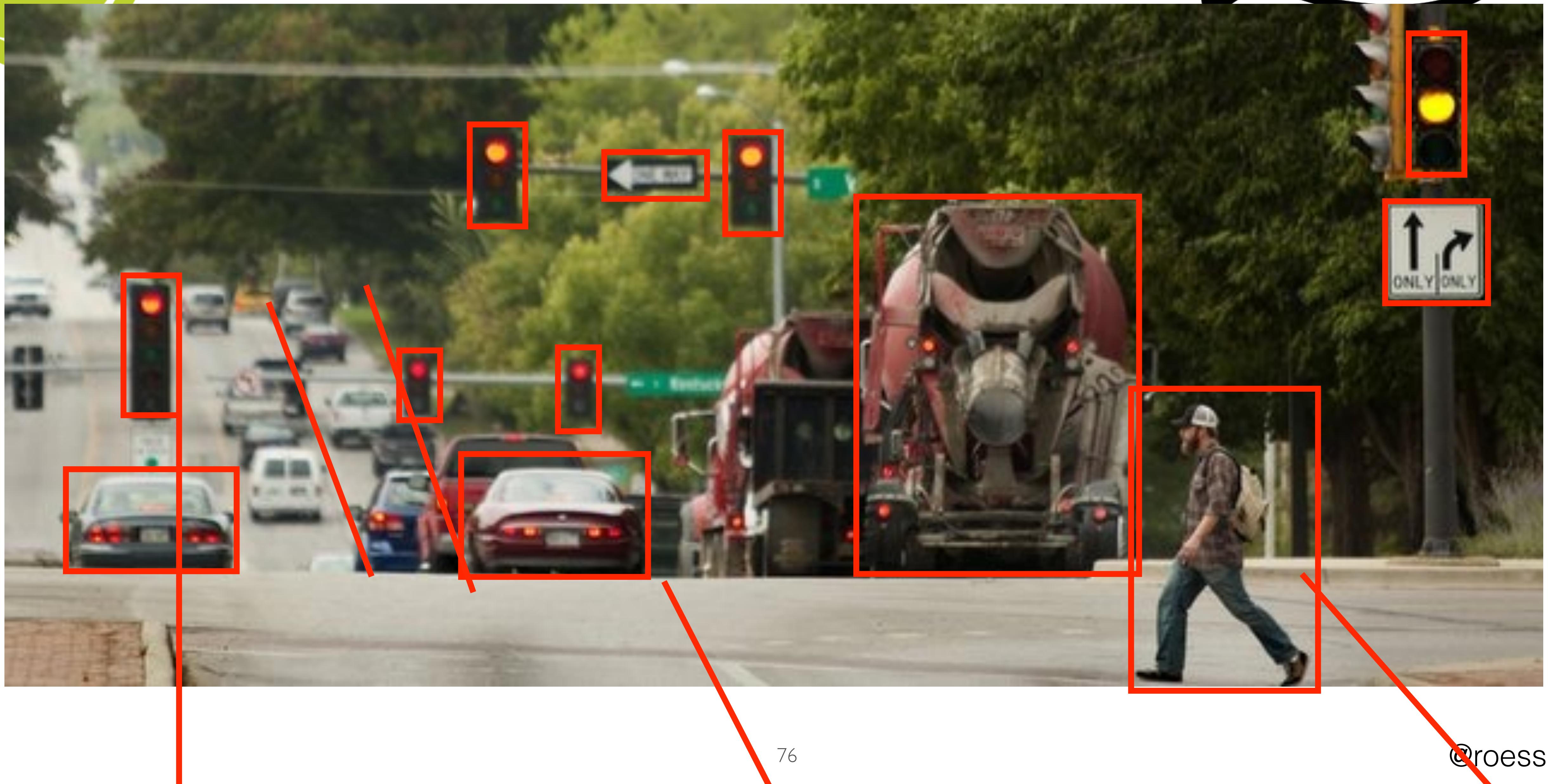
Calculate

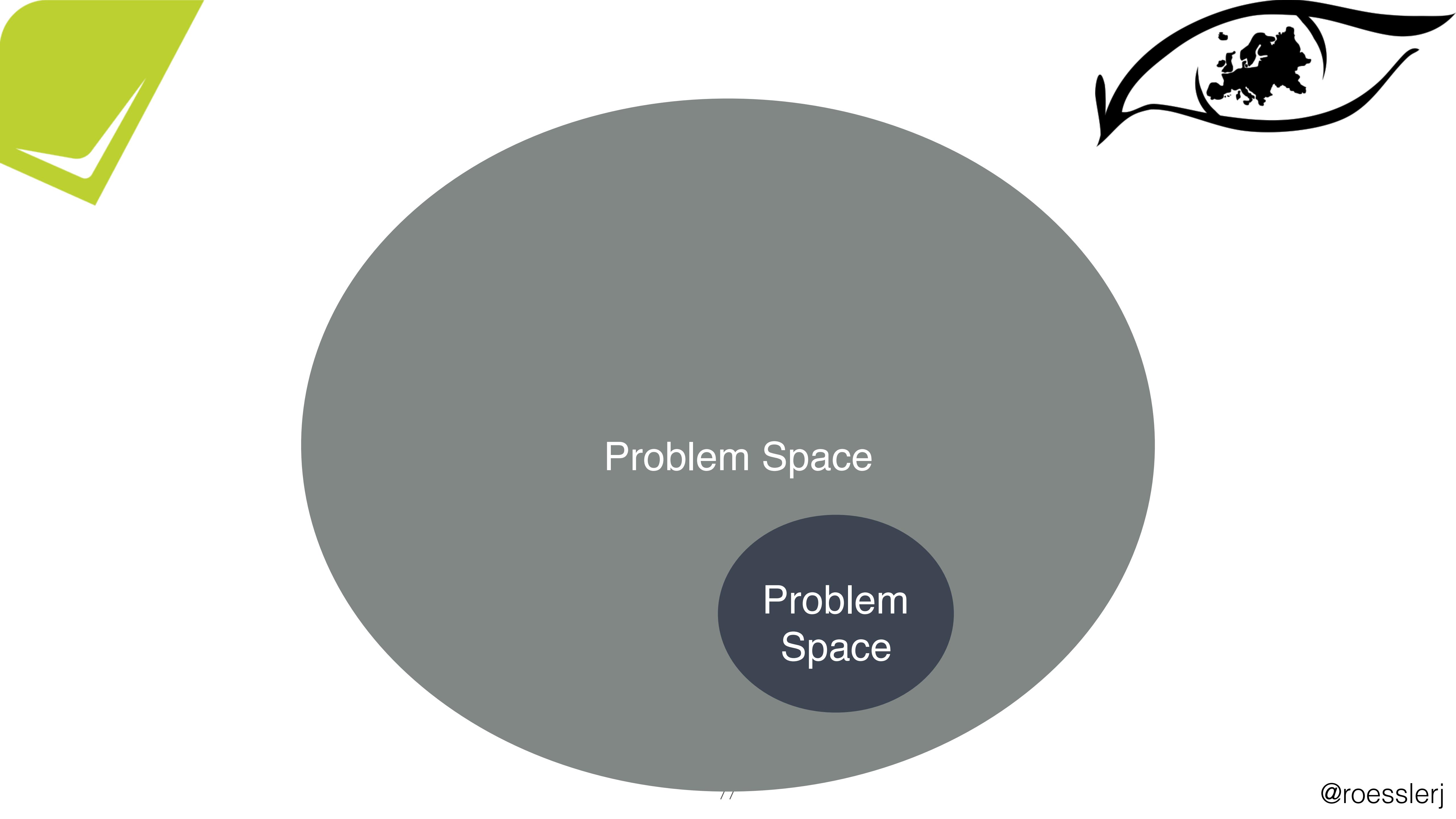
Units

Temperature	°C
Pressure	psia
Vapor Pressure	Pa
Density and Abs Humidity	g/m^3
Enthalpy	J/g



@roesslerj



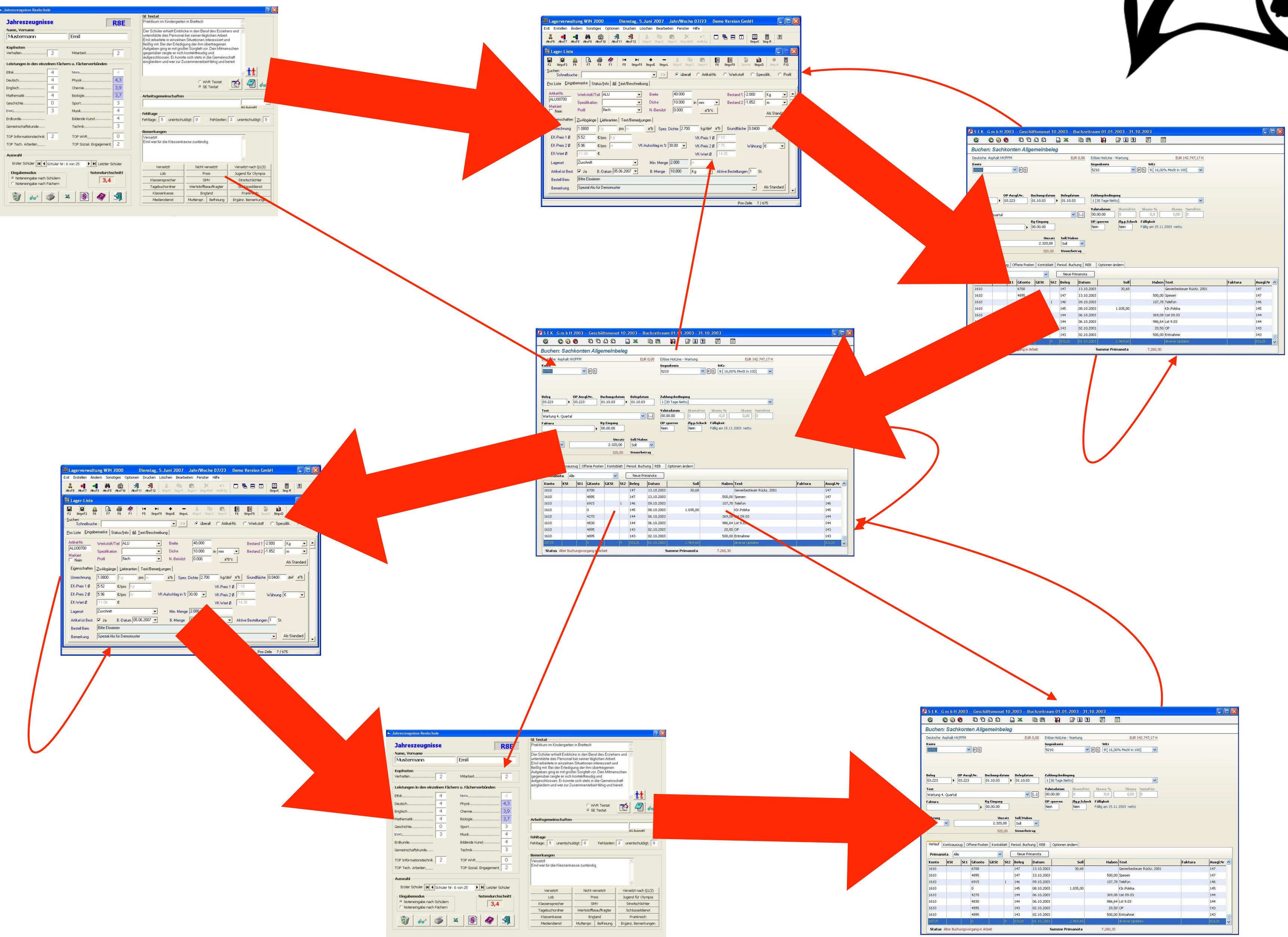


Problem Space

Problem
Space







@roesslerj



__ / __ / __

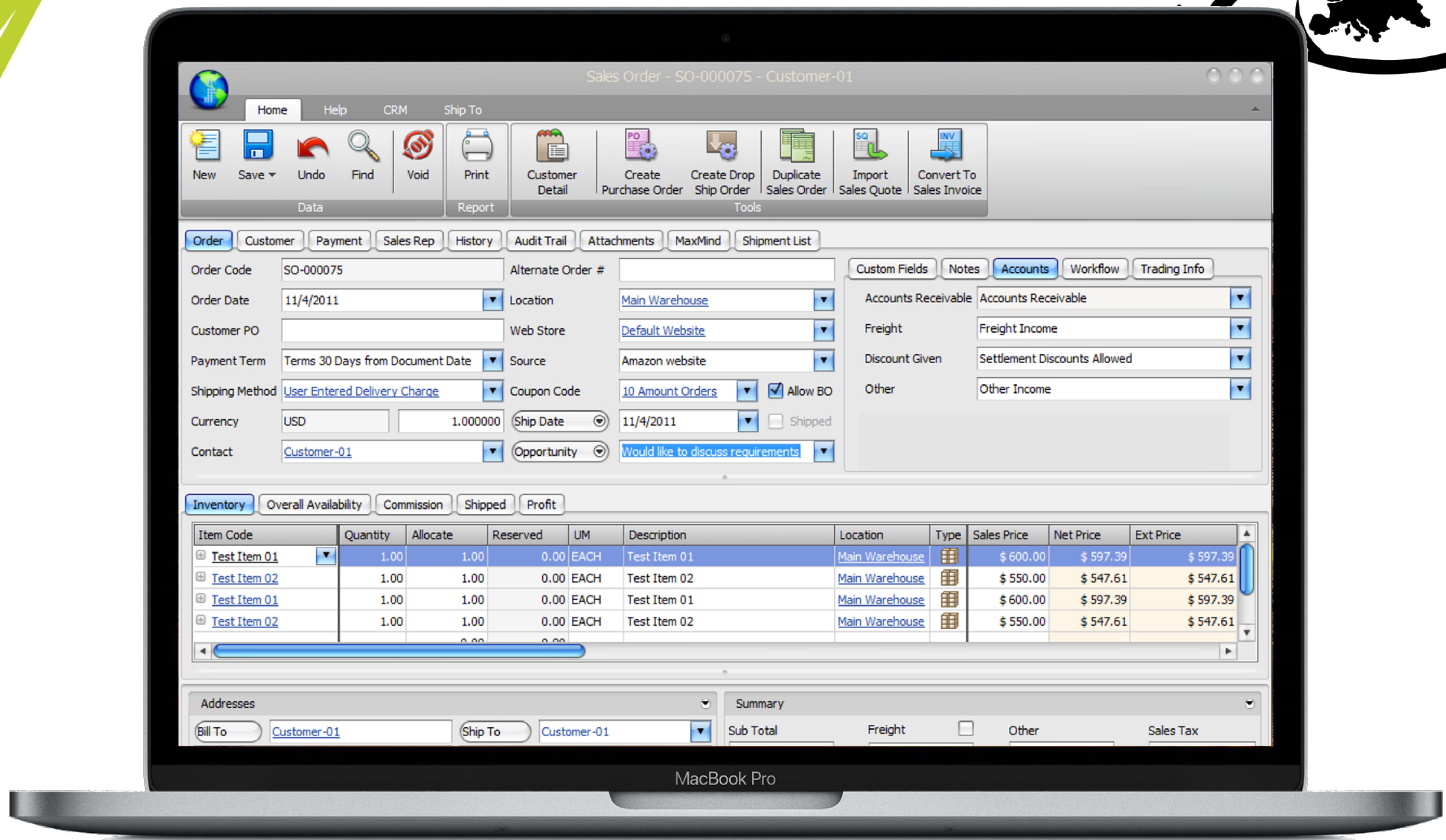


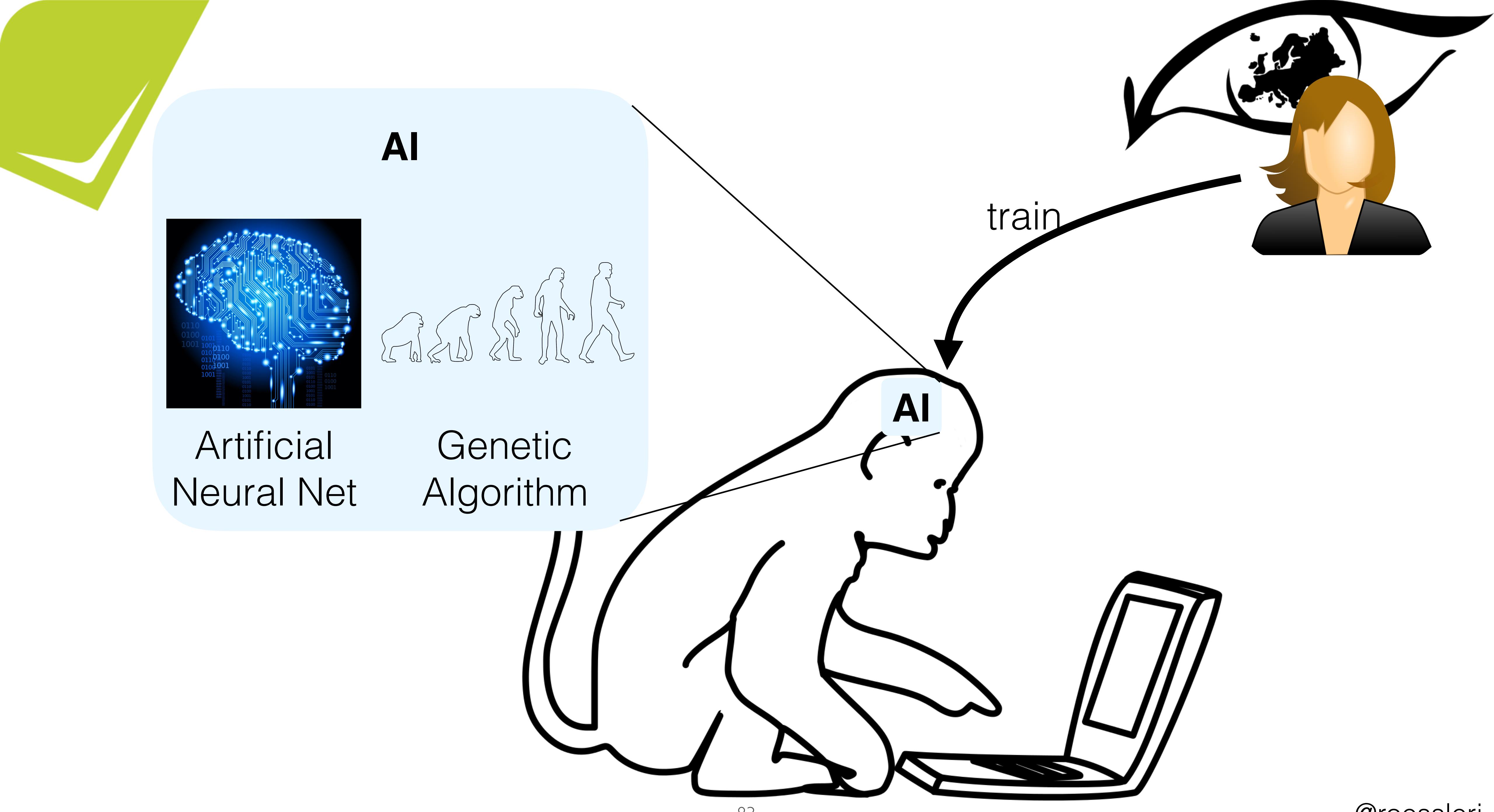
Geburtsdatum

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

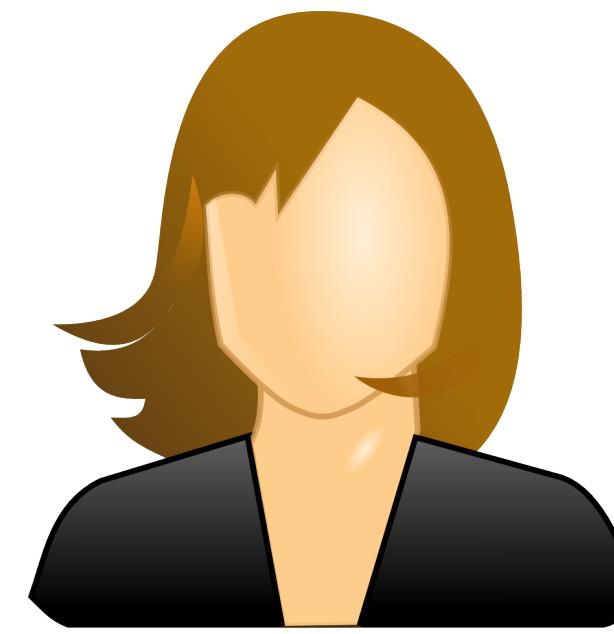






sapienz

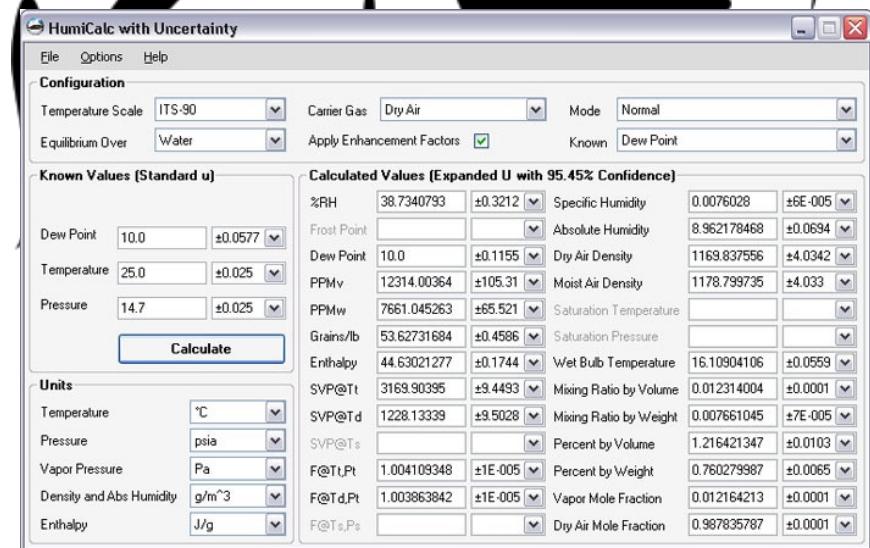
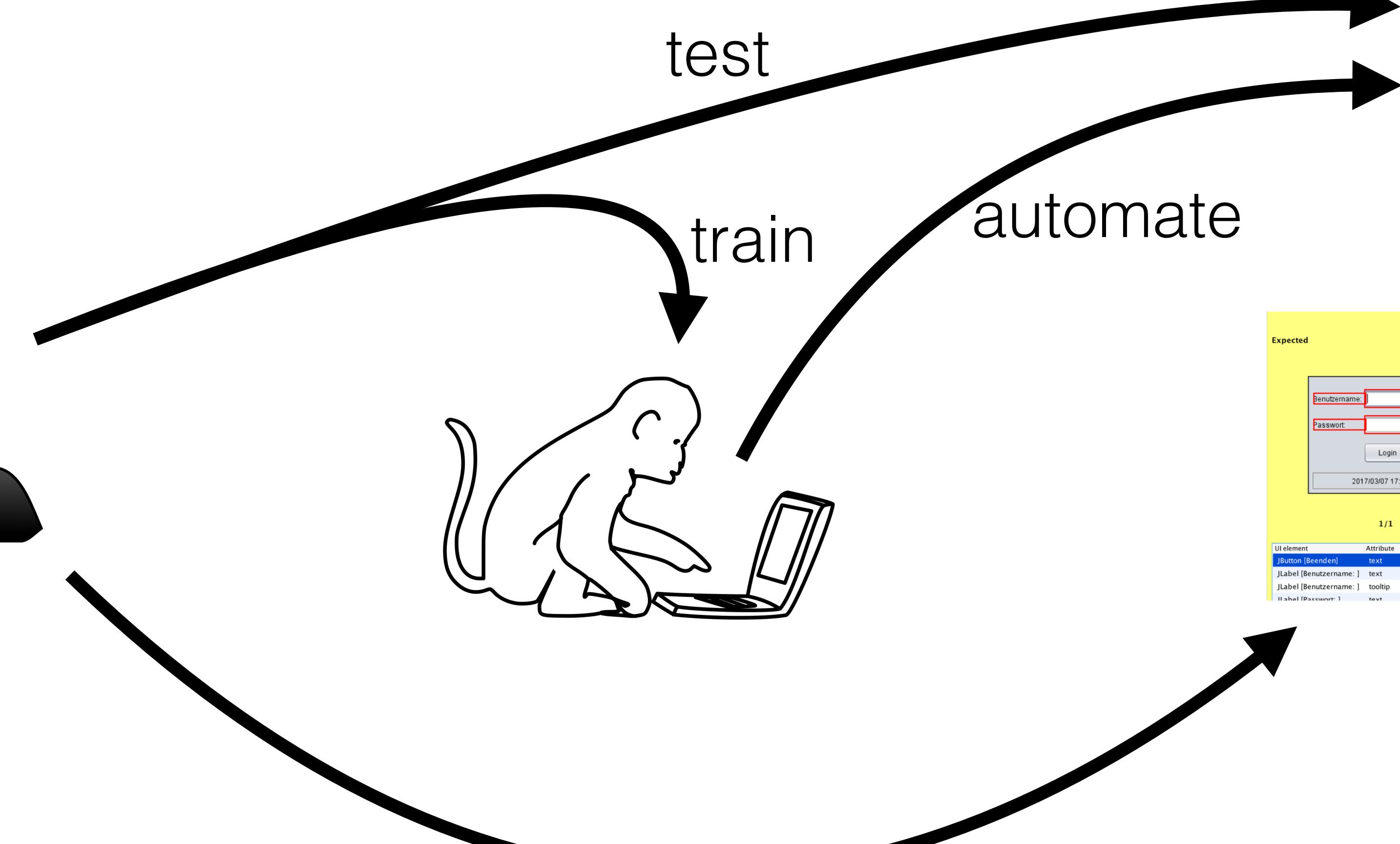
<https://code.fb.com/developer-tools/sapienz-intelligent-automated-software-testing-at-scale/>



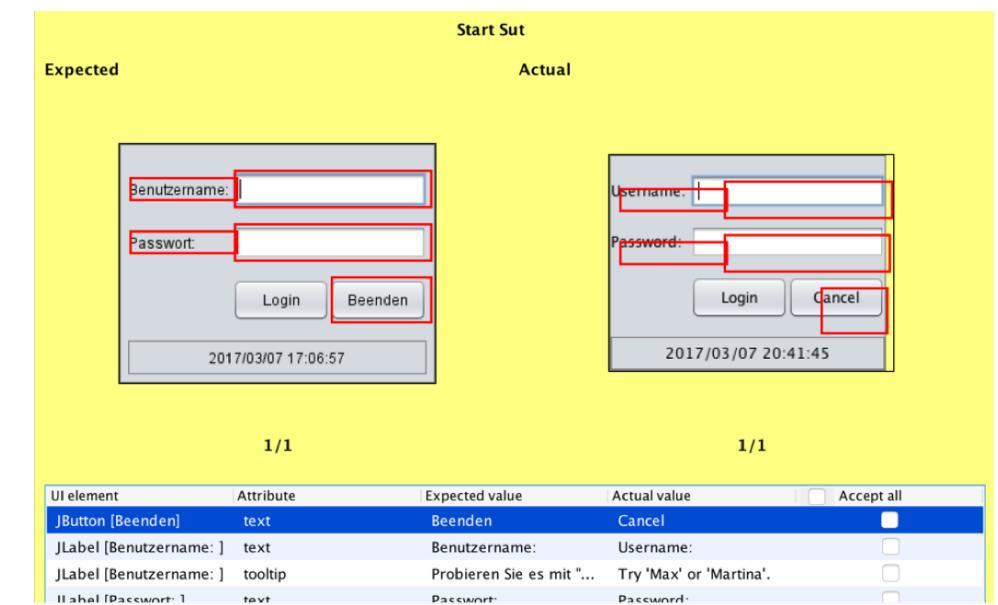
user



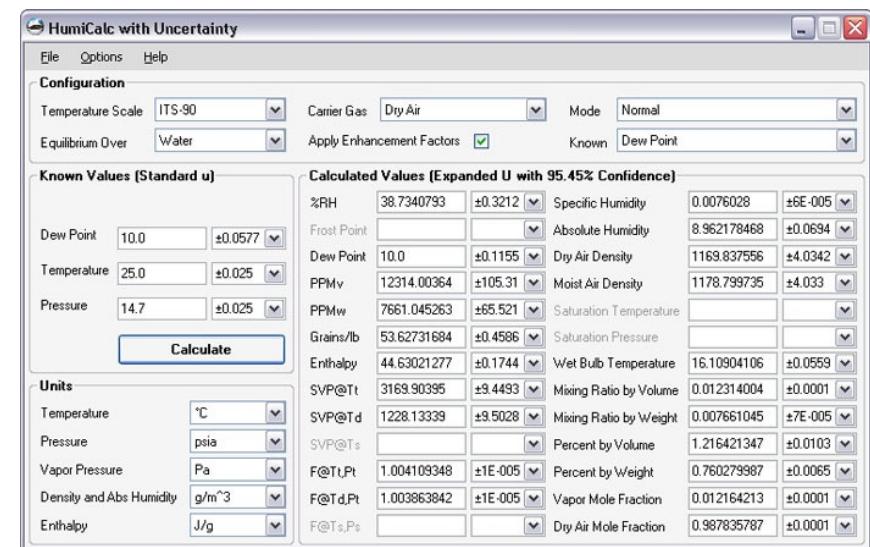
approve



1.0

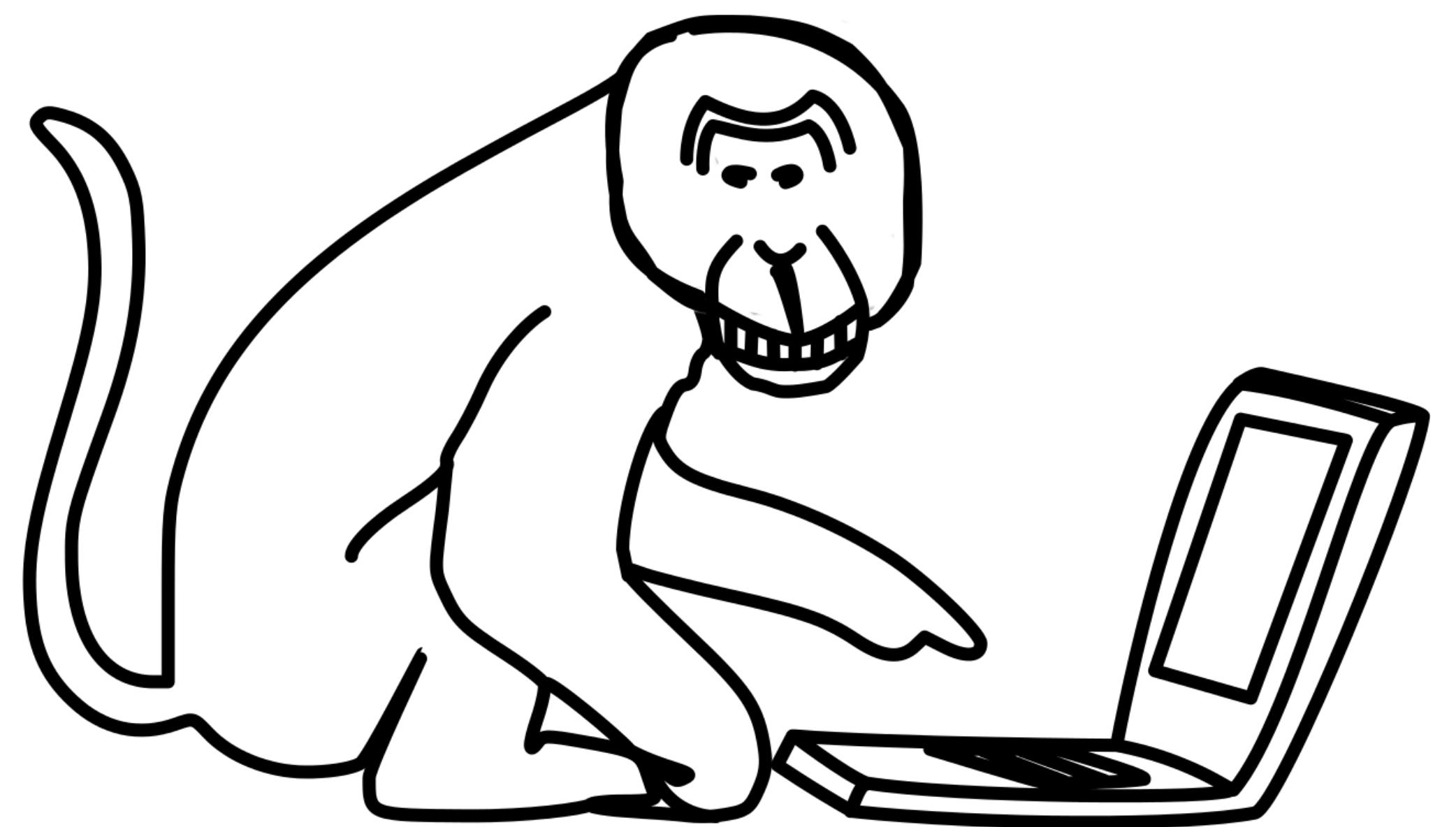


diff



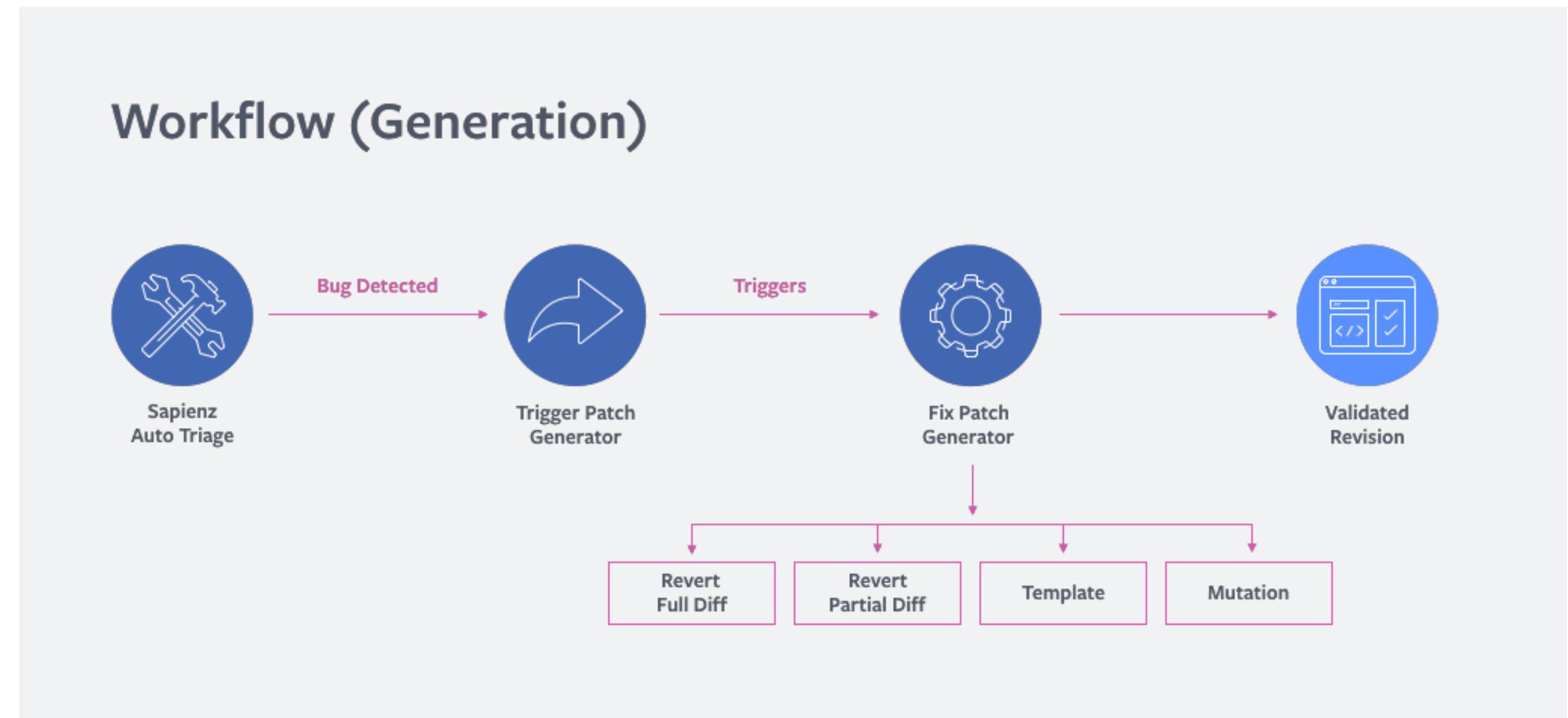
1.1

@roesslerj

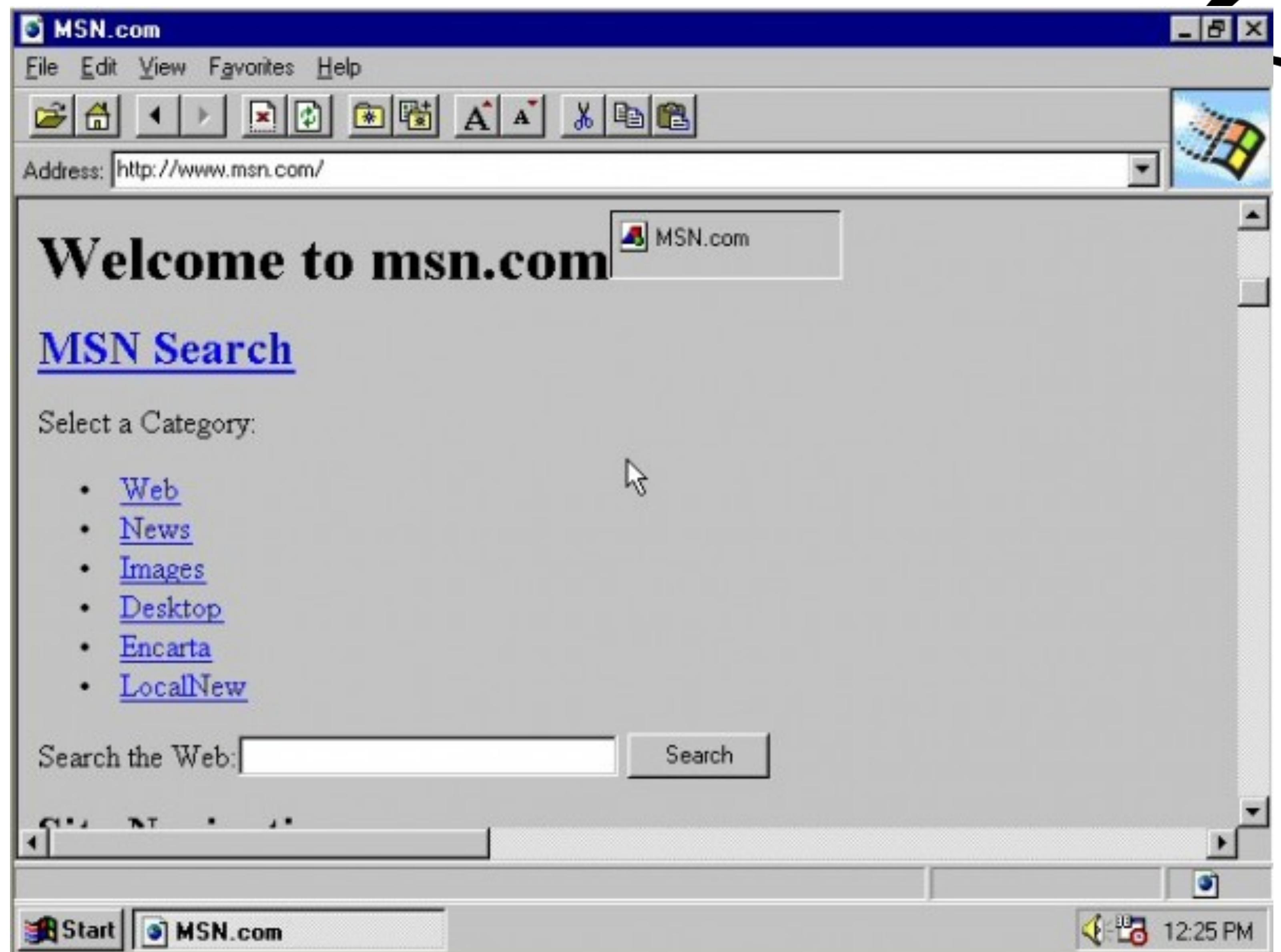




Automatically Fixing Bugs



<https://code.fb.com/developer-tools/finding-and-fixing-software-bugs-automatically-with-sapfix-and-sapienz/>



Summary



<https://github.com/retest/recheck-web>

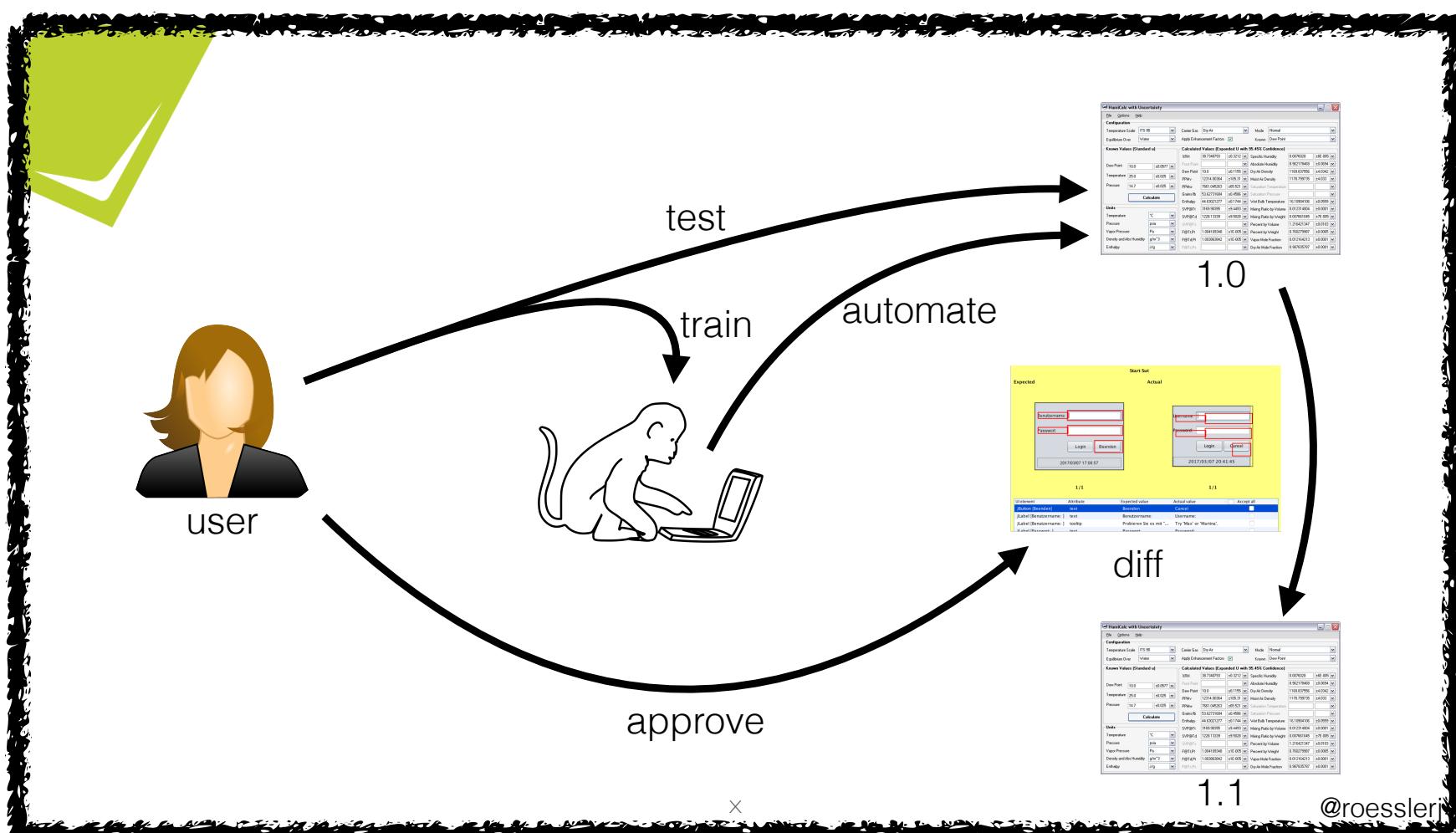
The screenshot shows the GitHub README page for the recheck project. It features the recheck logo, build status (passing), Maven Central version (0.6.2), license (AGPL), and PRs welcome. A note states: "recheck is a Golden Master-based test framework on top of Selenium that comes with powerful features. Check everything at once and create unbreakable tests." Below this is a screenshot of an IDE showing Java code for a Selenium test:

```
17 public class SeleniumLargeTest {  
18     private Vertx vertx;  
19     private WebDriver driver;  
20     private Recheck re;  
21  
22 }
```

At the bottom right is the handle @roesslerj.

Difference Testing

little maintenance
no wasteful testing
yet more complete



Monkey-Testing:

cheaper
faster
multiplies manual tests



roessler@retest.de



<https://www.facebook.com/retest.de>



<http://retest.rocks>