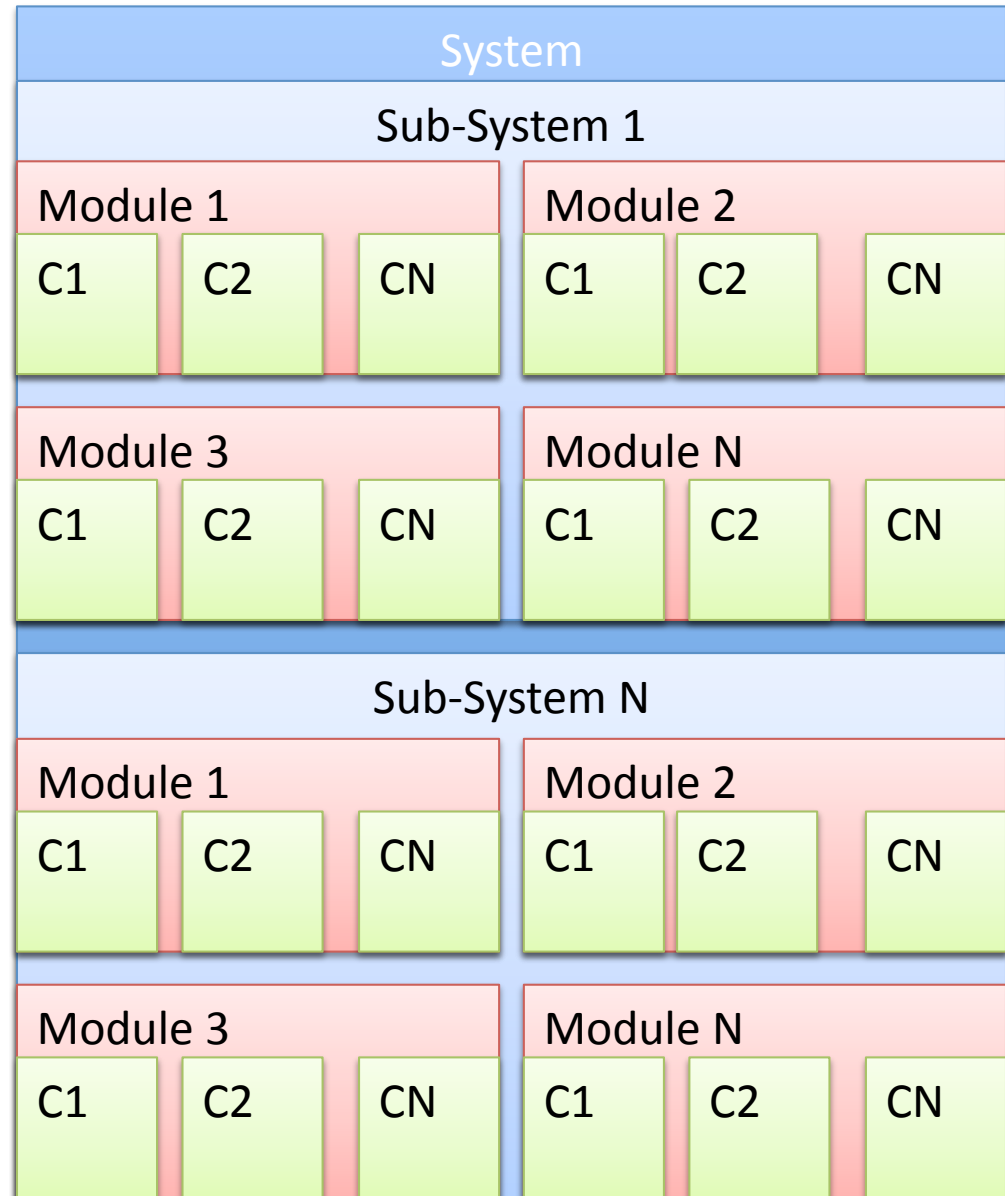


Automated Testing with Sikuli

Emil Börjesson

Testing

- Acceptance Tests
- System testing
- Module testing
- Unit testing
- Free testing
- And so on...

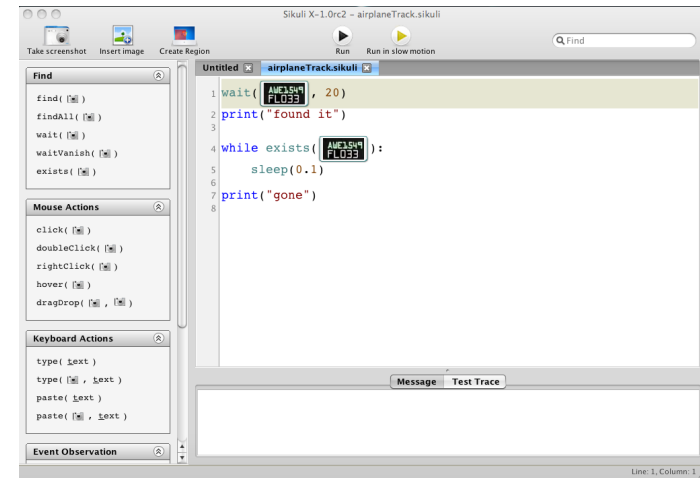


Automated testing

- Unit testing (JUnit)
 - Low level
 - Intrusive
 - Language dependent
- GUI object testing (Selenium, Abbot, etc)
 - Object level
 - Intrusive


Sikuli

- GUI level (image recognition)
- Easy to learn, easy to use
- Open source
 - Currently at RC 2.
 - Java and Python source!
- Automated System testing
 - Regression testing
- State of art!



Sikuli

- Download Sikuli: <http://sikuli.org/download.shtml>

PROJECT SIKULI 

[home](#) [download](#) [demos](#) [documentation](#) [contact](#) [blog](#)

Mac OS X

- [Sikuli IDE for Mac OS X 10.5 Leopard](#)
- [Sikuli IDE for Mac OS X 10.6 Snow Leopard](#)

Windows

- [Sikuli IDE for Windows](#)

Linux

- [Sikuli IDE for Linux 32bit & 64bit](#)
- [Sikuli for Debian GNU/Linux](#)

Others

- [Command Line Tool](#)
- [Source Code](#)
- [Old versions](#)

News

- Sikuli X 1.0 rc2 "Maltese" released on 2011/02/10. [Release notes](#)
- [Sikuli X is released!](#)


Mac OS X

Just unzip the file and drag Sikuli-IDE.app into /Applications.

Leopard (OS X 10.5)

Latest version: 0.10.2
(Sorry, we won't provide binaries for Leopard anymore.)

- [Sikuli-IDE-osx-10.5-0.10.2.zip](#) (for Intel Mac)




Mac


Snow Leopard (OS X 10.6)

Latest version: X 1.0 rc2

- [Sikuli-X-1.0rc2-osx-10.6.dmg](#) (for Intel Mac 32bit & 64bit)



Mac

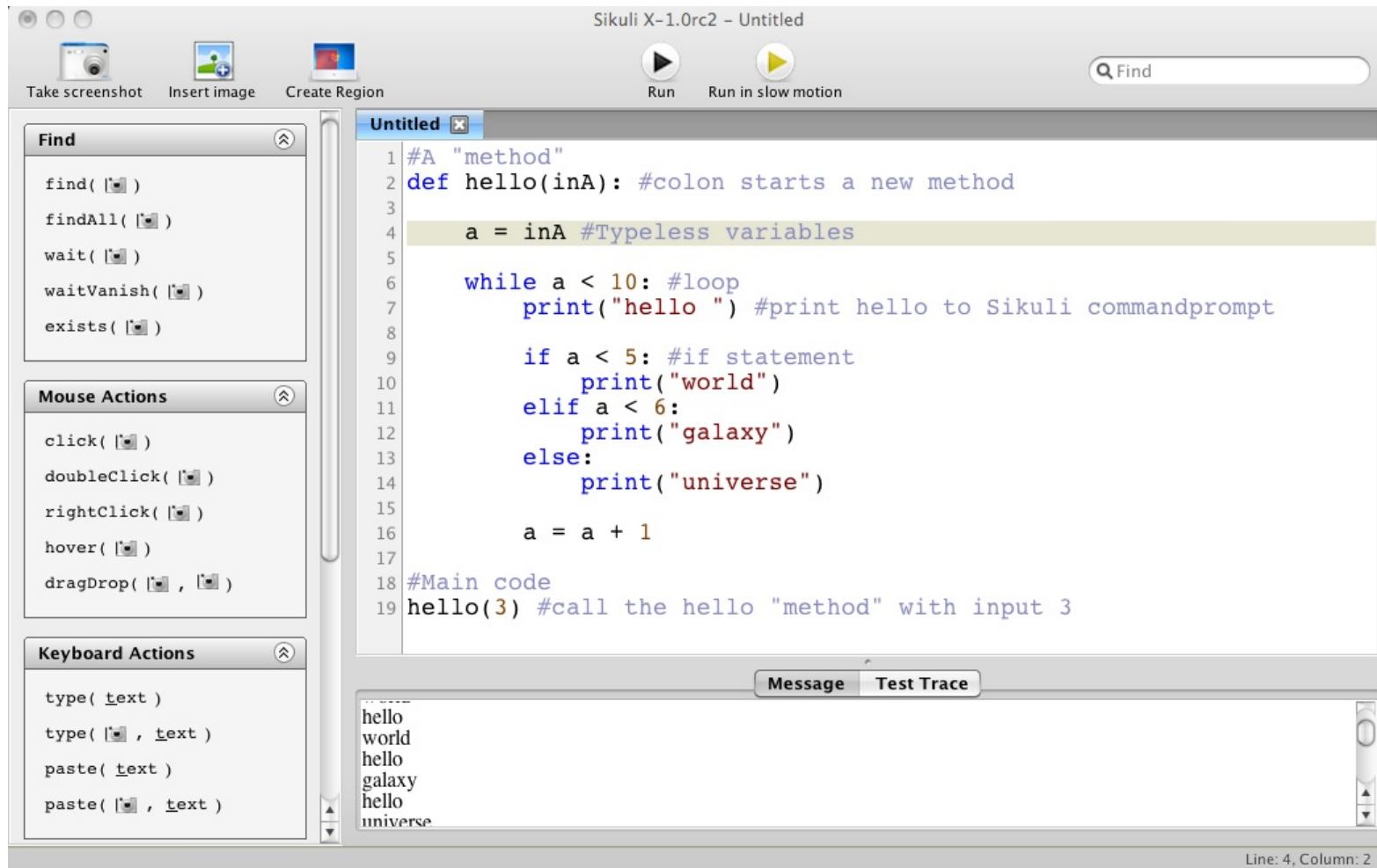


Universal

Install Sikuli

- Windows:
 - Install the .exe file
- Mac:
 - Open the dmg file, move icon to Applications
- Requires that you have Java 6!
 - <http://www.java.com/en/download/index.jsp>

Sikuli IDE (basic code constructs)



Sikuli IDE (Images)

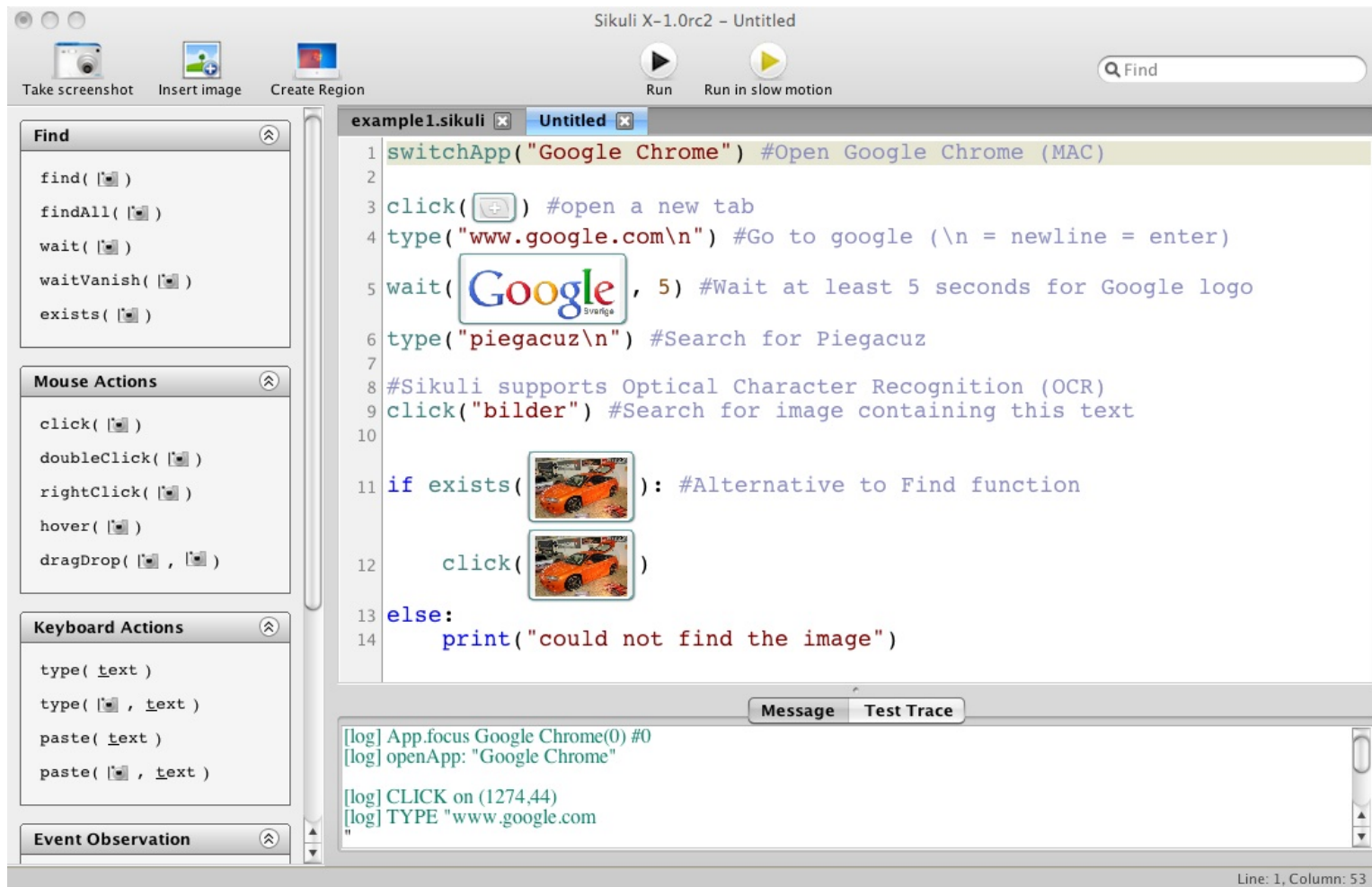
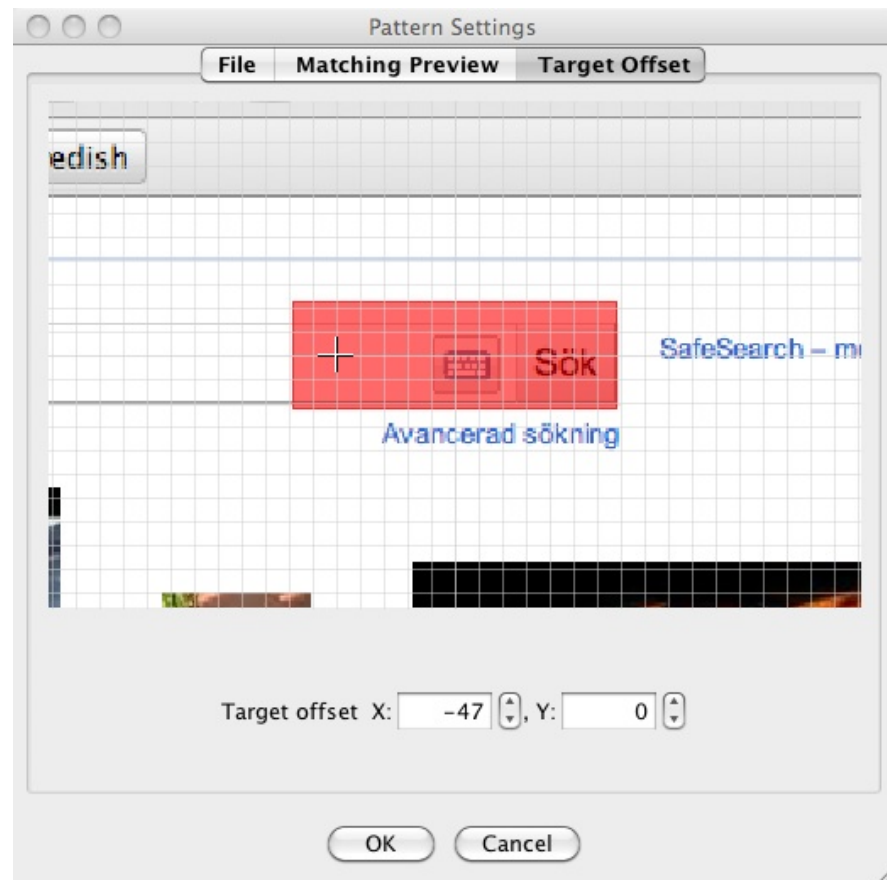
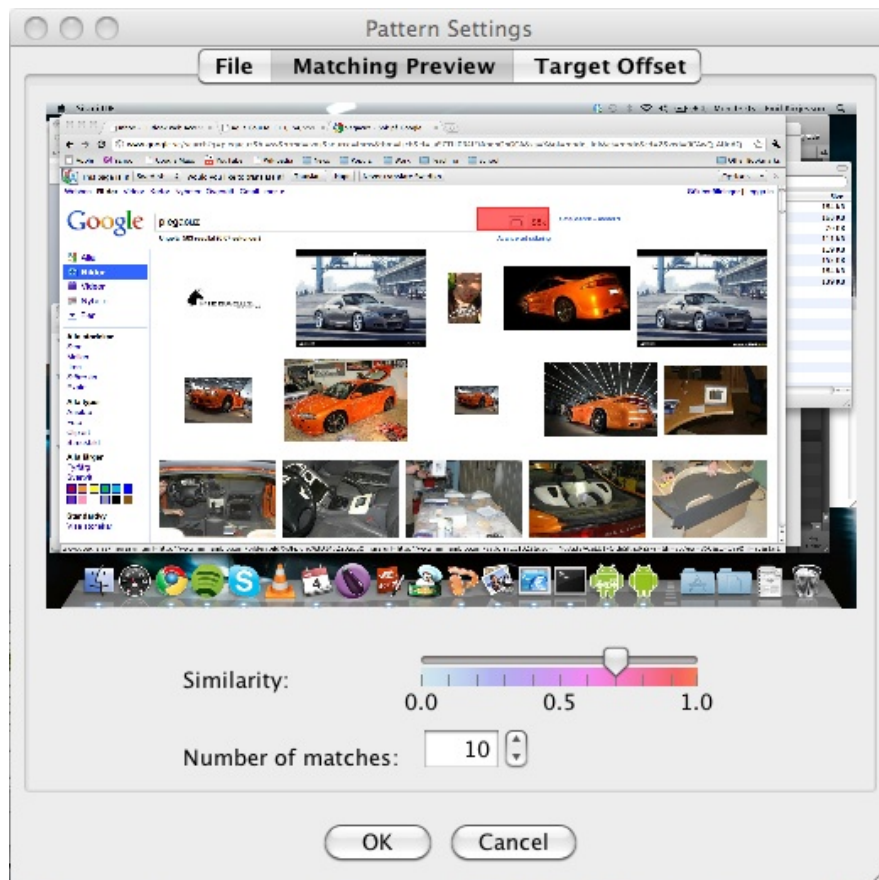
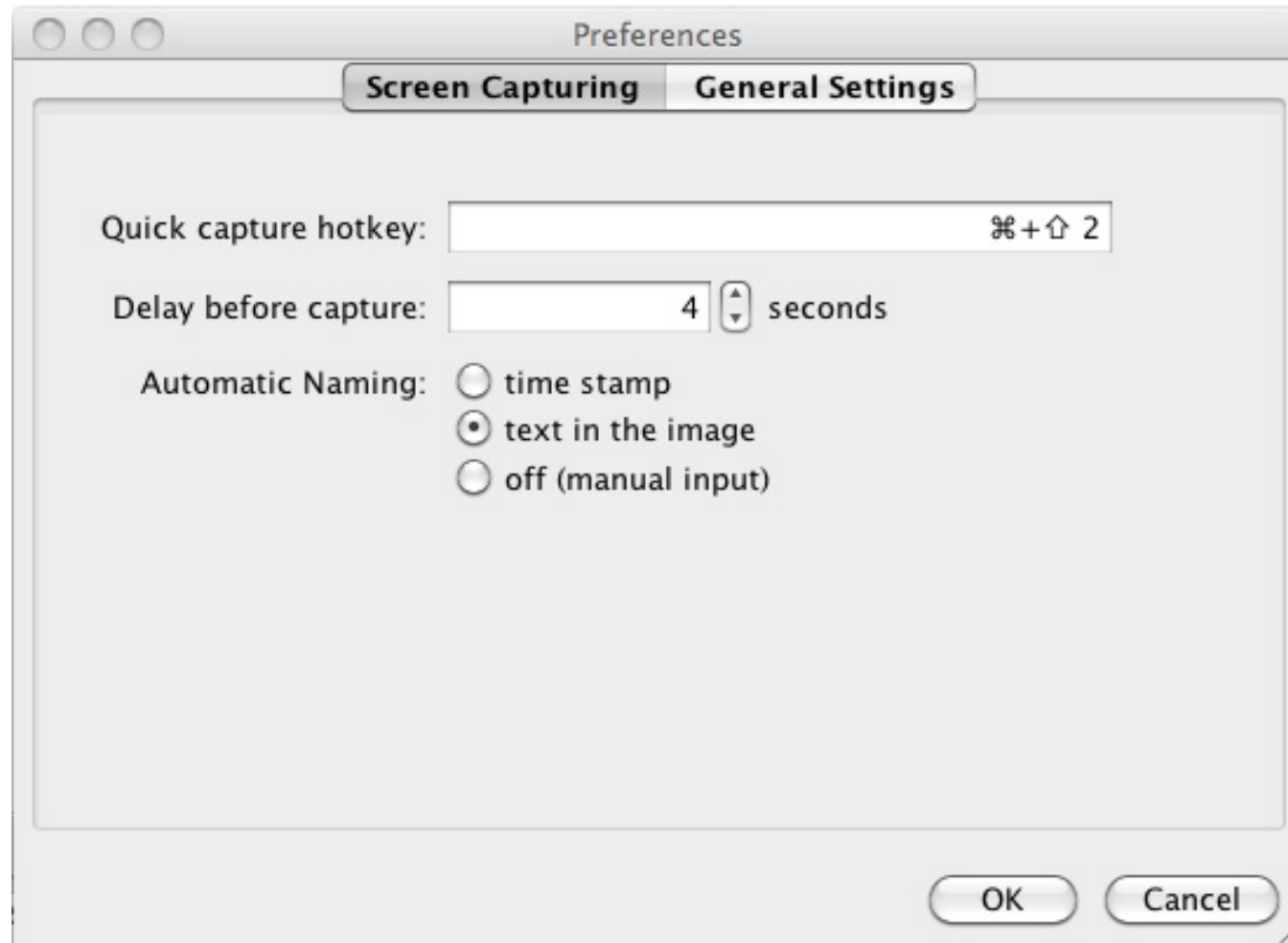


Image settings

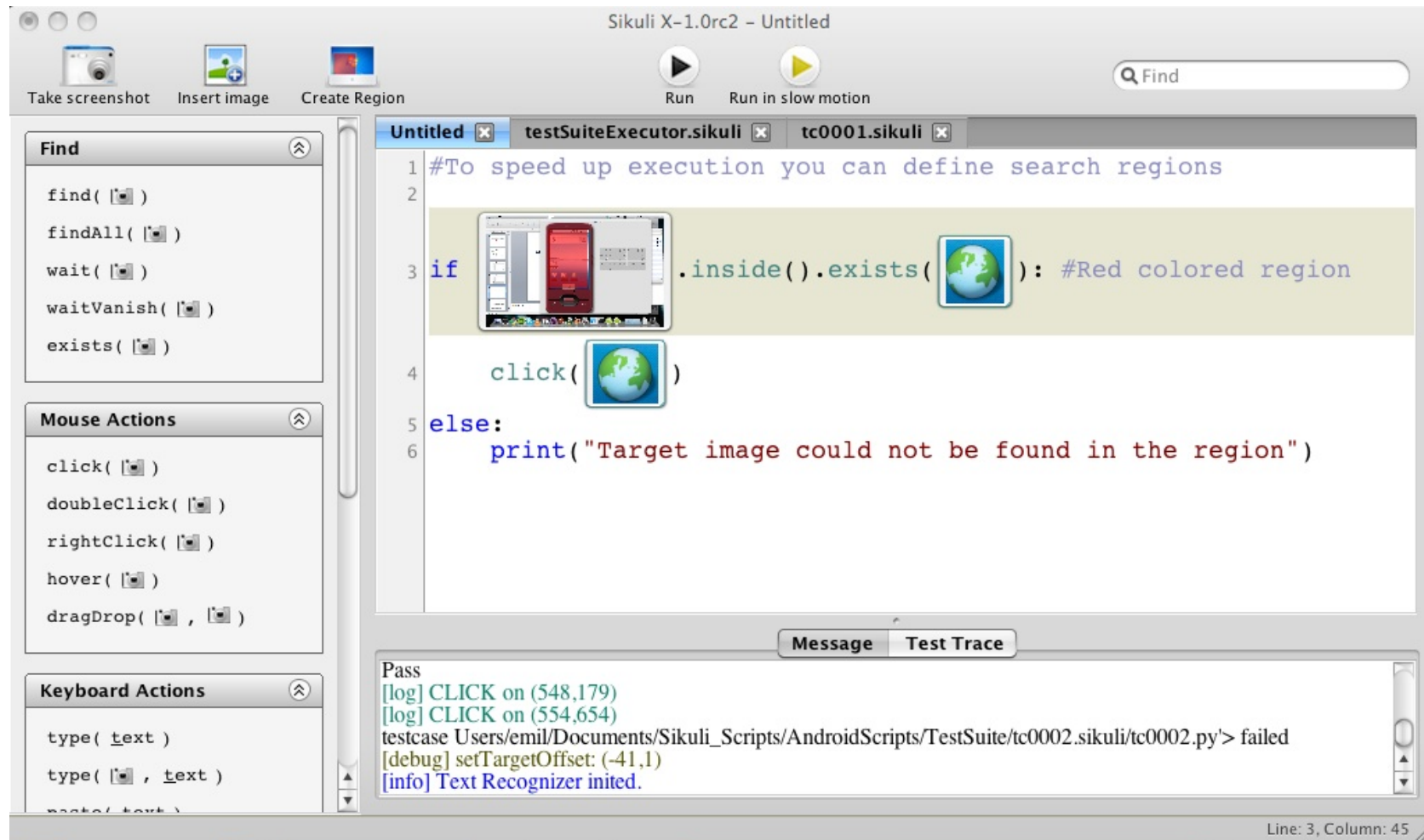
- Click on an image in the script to modify its settings



Preferences



Sikuli IDE (Regions)



DEMO

- YES! It's movie time! ;)

Sikuli test-suites

- Sikuli is state-of-the-art
- Sikuli does not have support for test-suites! ☹
 - SO! We made our own! :P
- Downloadable from the course webpage!
 - Includes:
 - testSuiteExecutor.sikuli – Main script. DON'T TOUCH!
 - outPutWriter.sikuli – Output. Touch A LOT!
 - tonA.sikuli – Helper script (Only works on Android 1.5 HGVA 320x480)
 - tc0001.sikuli – Example test case that writes Hello World in browser
 - tc0002.sikuli – Template test case

Sikuli test-suites

- Use the test-executor script
- Import test cases into the test-executor
- Test cases must have the following definitions:
 - setUp()
 - tearDown()
 - runTest()
- Link to download:

Test-Suite executor Script

```
7 #The test suite: (.sikuli files) and OutPutWriter
8 #Add further test cases here!
9 import outPutWriter
10 import tc0001
11 import tc0002
12 tcList = (tc0001,tc0002) #<--Don't forget to update this!
13
14 #Create a new outPutfile
15 outPutWriter.openEditor()
16
17 #Test suite executor loop:
18
19 for tc in tcList:
20     try:
21         outPutWriter.writeTC(str(tc))
22         tc.setUp()
23         tc.runTest()
24         tc.tearDown()
25     except:
26         outPutWriter.writeResult("-----\n")
27         outPutWriter.writeResult("Test case " + str(tc) + " Failed\n")
28         outPutWriter.writeResult("Sikuli exception:\n")
29         outPutWriter.writeResult(str(sys.exc_info())+"\n")
30         outPutWriter.writeResult("-----\n")
31
32 outPutWriter.closeAndSaveOutPutFile(outPutWriter.getGroupName())
33
```

Test case

```
1 from sikuli.Sikuli import *
2
3 #####
4 #
5 #Description:
6 #   Empty test case template
7 #
8 #####
9
10 #Required imports
11 import outPutWriter
12
13 #Set up the initial state of the test case
14 def setUp():
15     outPutWriter.writeResult("setup")
16
17 #Reset the simulator
18 def tearDown():
19     outPutWriter.writeResult("tearDown")
20
21 #Test definition
22 def runTest():
23     outPutWriter.writeResult("Pass")
```


Imports

- In file imported to:

```
1 #Get path to folder where test-cases are stored
2 import os
3 # get the directory containing your running .sikuli
4 myPath = os.path.dirname(getBundlePath())
5 if not myPath in sys.path: sys.path.append(myPath)
6
```



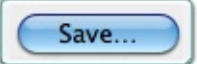
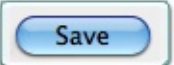
- In imported file (Must be first row!)

```
1 from sikuli.Sikuli import *
2
3 import outPutWriter
```

Writing output

- We have prepared a script-file that you can download from the course webpage.
(outPutWriter.sikuli)
 - IT DOES NOT WORK! :P
 - Edit definitions:
 - getOutWriter()
 - getAndroidSim()
 - closeAndSaveOutPutFile(group)
- See comments in code for help!

outPutWriter.sikuli (Changes)

```
4
5 def groupName():
6     return "Group_admin" #CHANGE
7
8 def getOutWriter():
9     return "textEdit" #CHANGE
10
11 def getAndroidSim():
12     return  #MAC solution!
13
14 def closeAndSaveOutPutFile(group):
15     #Path where to save the output-file
16     savePath = ""
17     filename = "TSR_"+str(strftime("%Y%m%d")) + "_" + group
18     switchApp(getOutWriter())
19     paste("TEST SUITE EXECUTION COMPLETE")
20     click()
21     click()
22     paste(savePath + filename)
23     wait(2)
24     click()
25
```

Issues you WILL encounter!

- TC bug corrections are not recognized by the Test-executor script.
 - Solution: Restart Sikuli
- You cannot copy paste code from a script to another! Images must be recaptured.
 - Solution: Import or recapture ☹
- There is no collection of all standard Sikuli script functions.
 - Solution: Google! “Sikuli +<what you seek>”

More issues you WILL encounter!

- Sikuli is extremely resolution sensitive!
 - Solution: Make sure you all have the same screen resolution!
- Timing is often a problem
 - Solution: add wait functions when needed.
 - Ex: `wait(3)` or `wait("img3",5)`
- The image recognition is sometimes too lenient.
 - Solution: set image recognition properties to higher than 70%.

Questions?