Testing Python applications.
A case study

John Alexis Guerra Gómez
jguerrag[en]cs.umd.edu

CMSC 737 Fall 2009
CS - UMD
Outline

- IRIS
- Testing
- Future work
IRIS In a nutshell

- IRIS is a system that allows blind students to see colors and shapes with their hands
IRIS software

- Python
- WxPython
- SQLObject
- PySQLite3
- SVN
IRIS demo
IRIS Abstraction

- We will not use all the functionalities of IRIS, instead we are going to use it just to codify images.
Hudson Integration

- SVN
- Compiling (Error checking)
- Test Oracle
- Code coverage
- Test cases
SVN

- Connect Hudson to our SVN server
- Nothing to different here
“Compiling”

- Python is an interpreted language
- However it offers a “byte-compilation” to speed up modules loading
  - *.pyc files
- Every time a modified module is imported *.pyc file is generated
  - We use this to check for syntax errors
- python -c “import iris2”
  - We use a small hack to avoid program full execution
Test oracle

- We developed a dummy IRIS device to capture the output of the program.
- Basically a simple threaded socket server dumping everything it receives in a log file.
  - Small demo
Code coverage

- coverage module.
  - http://nedbatchelder.com/code/coverage/
- Installation
  - easy_install coverage
- Running it
  - coverage -x iris2.py image.png
  - coverage -b -d htmlconv
Test cases

- We modified the IRIS code to include just a subset of its functionality, removing the GUI and using mainly the codification process.
- This program receives an image to process and some operations using the command line, then sends the codified image to the default IRIS device.
- We use some of our test images as test cases, and the test oracle dumps the codified images as outputs.
Future work

- Static analysis
  - pylint http://www.logilab.org/857
- Calling more IRIS functions from the command line to increase code coverage
- GUITAR type GUI testing
Questions?
Thanks!

More information at:

http://www.duto.org