Developing Apps for Android and Other Platforms with Kivy and Python

Andreas Schreiber <andreas.schreiber@dlr.de>
Outline

• Introduction
• Python
• Kivy
• Demos
• Limitations
• Credits
Me

Scientist, Head of department

Deutsches Zentrum für Luft- und Raumfahrt
German Aerospace Center

Founder, CEO

Enthusiastic about Python
DLR
German Aerospace Center

- Research Institution
- Space Agency
- Project Management Agency
Locations and employees

7400 employees across 32 institutes and facilities at 16 sites.


~1400 employees develop software
Python

Knowledge for Tomorrow
Python

- General-purpose, high-level programming language
- Object-oriented, aspect-oriented, functional
- Dynamic type system
- Easy-to-learn with clear and expressive syntax

```python
def faculty(x):
    if x > 1:
        return x * faculty(x - 1)
    else:
        return 1
```
Python on Mobile Devices

Early Mobile Development with Python

• PyS60 for Symbian
• Python CE for Windows Mobile

Current Mobile Development with Python

• Scripting Layer for Android (SL4A)
• Python for Android (Py4A)
• PySide / Qt for Android
• WinRT / IronPython for Windows 8
• Kivy…
Kivy
Kivy

- Platform-independent Python-Framework

- Available for
  - Android
  - iOS
  - Meego
  - Windows
  - Linux
  - OSX
  - (Raspberry Pi)

- Development in Python on all platforms
  - Not emulated!

kivy.org
Kivy Basics

- Framework for Natural User Interfaces (NUI)
  - Touchscreens / Multi-Touch

- GPU accelerated graphics
  - Based on OpenGL ES 2.0

- Suitable for prototypes as well as products
  - Porting to new platforms is easy
Kivy Software

- Open Source (LGPL), 7 Core developer

- Source code: https://github.com/kivy

- Documentation: http://kivy.org/docs

- Kivy on Google Play:
Kivy says Hello!

```python
from kivy.app import App
from kivy.uix.button import Button

class HelloApp(App):
    def build(self):
        return Button(text='Hello Berlin')

HelloApp().run()
```
Hello Berlin
Development with Kivy

• Python for widgets, input, program logic

• Language **KV** for layout und graphics

• Cython for low-level access to graphic routines
“Hello Berlin” with KV

from kivy.app import App

class HelloApp(App):
    pass

HelloApp().run()
Example: Pong

```python
import kivy
from kivy.app import App
from kivy.uix.widget import Widget

class PongGame(Widget):
    pass

class PongApp(App):
    def build(self):
        return PongGame()

if __name__ == '__main__':
    PongApp().run()
```
#:kivy 1.6.0

<PongGame>:
    canvas:
        Rectangle:
            pos: self.center_x - 5, 0
            size: 10, self.height

    Label:
        font_size: 70
        center_x: root.width / 4
        top: root.top - 50
        text: "0"

    Label:
        font_size: 70
        center_x: root.width * 3 / 4
        top: root.top - 50
        text: "0"
Pong

Full example: [http://kivy.org/docs/tutorials/pong.html](http://kivy.org/docs/tutorials/pong.html)
Accessing Java Classes from Python

- Smartphones have many APIs
  - Camera, Compass, Contacts, Location, ...

- Access from Python via PyJNIus
  - https://github.com/kivy/pyjnius
  - Implemented with JNI and Java reflection

Example

```python
from jnius import autoclass

Hardware = autoclass('org.renpy.android.Hardware')
print 'DPI is', Hardware.getDPI()
```
Packaging

• Creating packages for Windows, OSX, Android und iOS:
  http://kivy.org/docs/guide/packaging.html
Build Tools

Tool chain
• Python-for-android
• Cross compiler for ARM
• Android SDK & NDK
• Python and some Python packages

Buildozer
• Hides the complexity: Downloads, compiles, packages Kivy source code
• https://github.com/kivy/buildozer

% buildozer android debug deploy run
Kivy Showcase
Kivy Pictures
Small Dragon Luki
Speech therapy game for kids
Small Dragon Luki
MQTT Client
Steering Plant Growth

- Webcam takes picture of plants
- Computer detects plant
- Computer generates an image for lighting
- Light source (e.g., a projector) illuminates the plant using the generated image
Other Examples...
iOS-App Deflectouch

https://itunes.apple.com/de/app/deflectouch/id505729681
iOS/Android-App ProcessCraft

http://showgen.com
Limitations
Missing, but Planned (or In Progress)

**User Interface Designer**
- Design tool for Kivy Language KV
- Planned for GSoC

**Abstraction of mobile APIs**
- Platform-independent Python wrapper for platform APIs (Android, iOS, Linux/Mac/Windows)
- Project **Plyer** will start as GSoC project maybe

**Porting to Raspberry Pi**
- Useful for small/cheap standalone systems
-Founded via Crowdsourcing ([bountysource.com](http://bountysource.com))
Credits

Thanks to the Kivy developers

• Mathieu Virbel (@mathieuvirbel)
• Thomas Hansen (@hansent)
• Gabriel Pettier (@tshirtman)
• and many others
Questions?

Summary

- Kivy allows platform-independent development of apps for Android, iOS, Meego, Windows, OSX and Linux
- Suitable for multi-touch and graphics applications, such as kiosk systems, exhibits, games, ...

Andreas Schreiber
Twitter: @onyame
http://www.dlr.de/sc