

Beyond Myro

Douglas Blank

Bryn Mawr College

Institute for Personal Robots in Education

IPRE Winter Workshop

Atlanta, GA

January 7-9, 2011



BRYN MAWR

Microsoft
Research



Beyond Myro

- Pyjama (Python, Ruby, Scheme, Sympl, Dinah)
 - Doug Blank, Bryn Mawr College
- Myro in C++
 - John Hoare, University of Tennessee, Knoxville
- Myro in Java
 - Douglas Harms, DePauw University
- Myro in Chez Scheme, Indiana University
- Myro in Spanish, Universidad Nacional De La Plata, Argentina

Myro 2 from IPRE

- Open Source Prototype begun in 2006
- Written in CPython, version 2.4
- Cross-platform
- Relies on a number of additional open source projects: pygame, idle, text-to-speech, tkinter, graphics, Python Image Library, chat, serial, and more
- Large number of dependencies

Myro 2 Issues

- Dependencies!
- Python only
- Many components out of our control
- Developing context libraries for additional languages is expensive

Python Shell

File Edit Shell Debug Options Windows Help

```
Python 2.4.4 (#71, Oct 18 2006, 08:34:43) [MSC v.1400 64-bit AMD64]
Type "copyright", "credits" or "license()" for more
```

```
*****
Personal firewall software may warn about the connection
makes to its subprocess using this computer's network
interface. This connection is not visible on the system
interface and no data is sent to or received from the
*****
```

```
IDLE 1.1.4
```

```
>>> from myro import *
```

```
Myro, (c) 2006 Institute for Personal Robots in Education
[See http://www.roboteducation.org/ for more information]
Version 1.0.0, Revision 1.85, ready!
```

```
>>> initialize("com5")
```

```
Waking robot from sleep...
```

```
Hello, I'm Scribby!
```

```
>>> |
```

Frameworks and Contexts

- BlueJ, Greenfoot – Java only
- Alice – tied to 3D context
- 3D context – tied to Alice
- DrScheme – Scheme only
- IDLE – tied to Python, not designed by educators
- MediaComp – Java or Python

Frameworks and Contexts

- Use BlueJ and 3D worlds
- Use Alice's drag-and-drop interface with Robots
- Use Scheme with MediaComp
- Start with drag-and-drop and move to Python, while keeping the context the same

Goal

You should be able to pick your language,
pick your context, and
work in a educator-designed IDE.

Goal

You should be able to pick your language,
pick your context, and
work in a educator-designed IDE.

Every day.

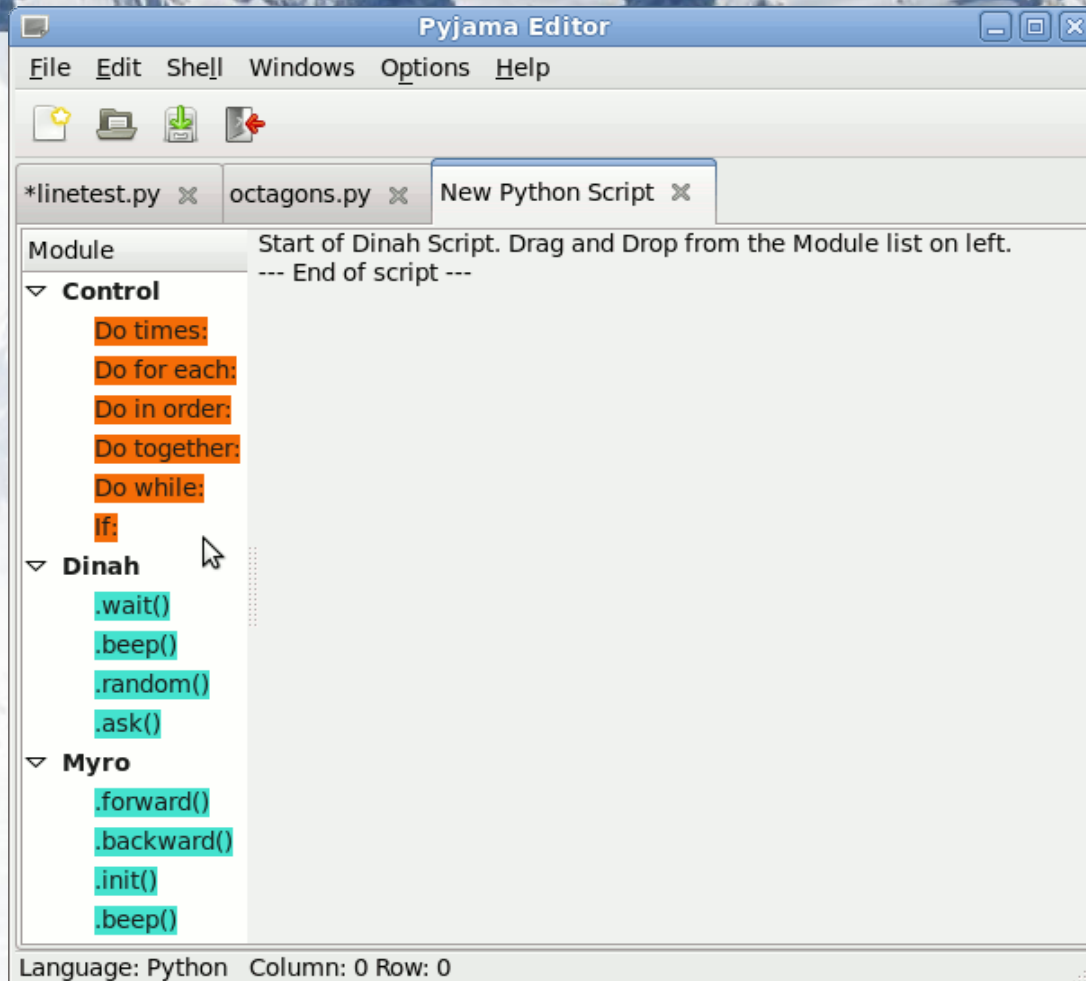
Why Yet Another Framework?

- By Educators, for Educators
- How much time do students spend on task?
- Because languages change
- Because contexts change
- Because contexts shouldn't be language specific
- But the core framework need not change

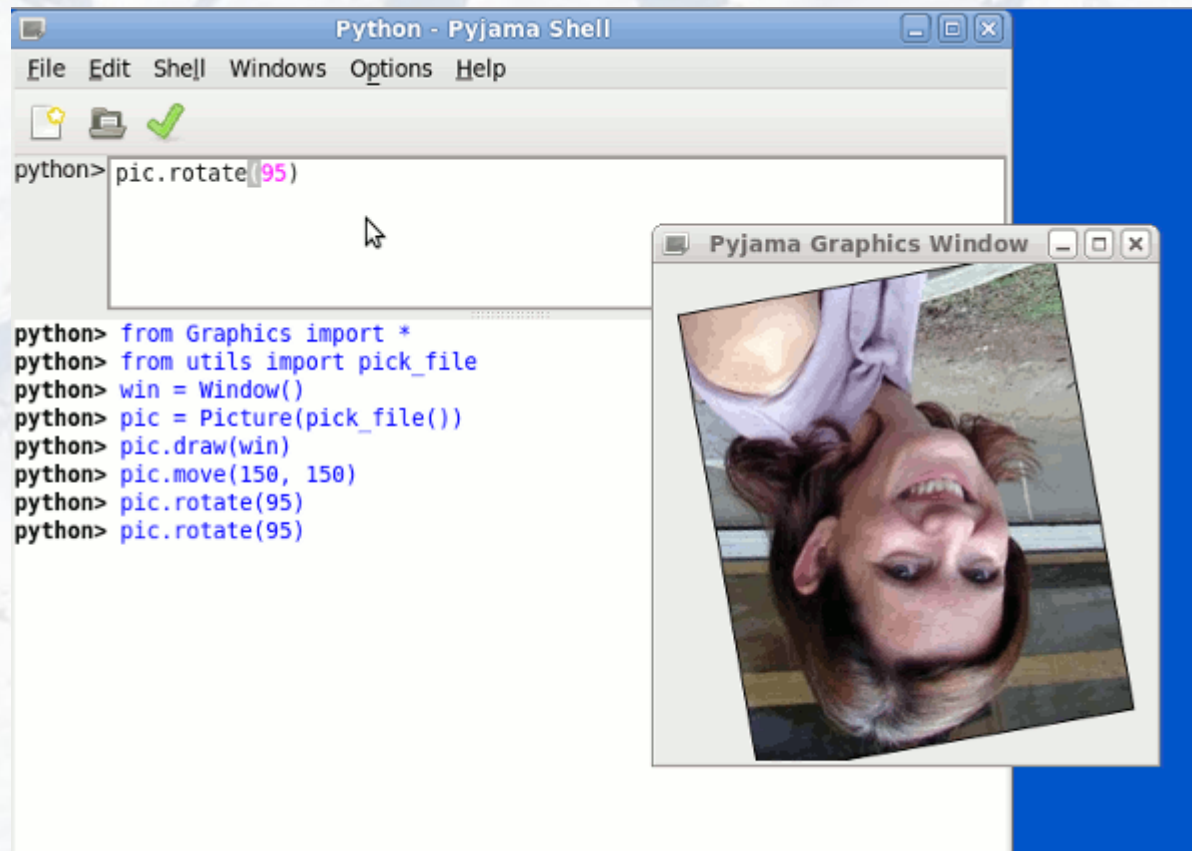
Introducing the Pyjama Project

- Integrated interpreter and editor
- Language independent
- Context Libraries work with all languages
- Designed for education and to fulfill educator's goals
- Classroom and student friendly
- Switch languages instantly
- For “scripting” languages, does not cover C++ nor Java

Dinah



Python, Ruby, SympI, and Scheme



The image shows a screenshot of a Pyjama Shell window titled "Python - Pyjama Shell". The window has a menu bar with "File", "Edit", "Shell", "Windows", "Options", and "Help". Below the menu bar are icons for a folder, a printer, and a green checkmark. The main area of the shell shows a Python prompt "python>" followed by the command "pic.rotate(95)".

Below the shell window, there is a "Pyjama Graphics Window" titled "Pyjama Graphics Window". This window displays a photograph of a woman's face, which has been rotated 95 degrees clockwise. The woman is smiling and wearing a purple top.

```
python> pic.rotate(95)

python> from Graphics import *
python> from utils import pick_file
python> win = Window()
python> pic = Picture(pick_file())
python> pic.draw(win)
python> pic.move(150, 150)
python> pic.rotate(95)
python> pic.rotate(95)
```

www.roboteducation.org

Robotics@GT
& Intelligent Machines

BRYN MAWR

Microsoft
Research

