Phybots: A Toolkit for Making Robotic Things

Jun Kato, Daisuke Sakamoto, Takeo Igarashi
The University of Tokyo, JST

14th June, DIS2012
Motivation

Adding mobility to our daily objects
Alarm Clock
Speaker
Gulf of HCI and Robotics
Toolkits for Physical UIs

- Software abstraction of actuators and sensors

[Phidgets ‘99]  [Arduino ‘05]

- Support for iterative development process

[d.tools ‘06]
Gulf of HCI and Robotics

Toolkit for: **Physical UIs**

Target users: HCI researchers & Interaction designers

Focus: Prototyping

Software API: Low-level&Static

Hardware: Small&Cheap

Phidget Kit, $200
Toolkits for Robot Programming

• Middleware for distributed environment
  ROS.org

• Collection of algorithms
  Carmen
  OpenCV

• Education and entertainment

robot.translate(distance);
robot.rotate(angle);
## Gulf of HCI and Robotics

<table>
<thead>
<tr>
<th>Toolkit for:</th>
<th>Physical UIs</th>
<th>Robots</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Target users:</strong></td>
<td>HCI researchers &amp; Interaction designers</td>
<td>Robotics people</td>
</tr>
<tr>
<td><strong>Focus:</strong></td>
<td>Prototyping</td>
<td>Reliability</td>
</tr>
<tr>
<td><strong>Software API:</strong></td>
<td>Low-level&amp;Static</td>
<td>High-level&amp;Extensible</td>
</tr>
<tr>
<td><strong>Hardware:</strong></td>
<td>Small&amp;Cheap</td>
<td>Medium-Large&amp;Expensive</td>
</tr>
</tbody>
</table>

- **Phidget Kit, $200**
- **K-Junior, $938**
## Gulf of HCI and Robotics

<table>
<thead>
<tr>
<th>Toolkit for:</th>
<th>Robotic Things</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Target users:</strong></td>
<td>HCI researchers &amp; Interaction designers</td>
</tr>
<tr>
<td><strong>Focus:</strong></td>
<td>Prototyping</td>
</tr>
<tr>
<td><strong>Software API:</strong></td>
<td>High-level&amp;Extensible</td>
</tr>
<tr>
<td><strong>Hardware:</strong></td>
<td>Small&amp;Cheap</td>
</tr>
</tbody>
</table>
Goal of Phybots

• Add mobility to physical objects
  – In a cheap and easy way
  – Through high-level and extendible API
  – With support for the whole prototyping process
Prototyping with Phybots

Hardware construction

Programming with built-in API
(Extend API when needed)

Test with runtime debug tool
Localization and Locomotion API

• Hardware setup: easy and cheap
  – Camera
  – PC or Mac
  – Robotic things

From our user study:
Miniature drive recorder
Localization and Locomotion API

• Navigation by global coordinates
  – Move
  – Push
  – TracePath

Task move = new Move(
  mouseX, mouseY);
moves.assign(robot);
moves.start();

From our user study:
Beach flags with obstacles
Localization and Locomotion API

• Vector field navigation
  – Easy design of new behavior
    ex) Follow another robot
  – Combination of existing fields
    ex) Move + collision avoidance
Extensible Software Architecture

- **Robot, Resource and Task** abstraction
  - *Robot* passes one or more *Resources* to *Task*.
    ex) *MindstormsNXT* passes *DifferentialWheels* to *Move*.
Extensible Software Architecture

- *Workflow* for higher-level task management

ex) Moving mug:

1. Move to the position just under the electric kettle
2. Wait for the kettle to boil the water
3. Wait for the kettle to pour the hot water
4. Move to the position in front of the user
Runtime Debug Tool

• Entity Monitor
• Service Monitor
• Workflow Monitor
Runtime Debug Tool
Entity Monitor
Runtime Debug Tool
Service Monitor
Runtime Debug Tool

Workflow Monitor
User Studies

1. Alpha version deployment
   – To graduate students

2. Current version deployment
   – To HCI students
   – To robotics students
Course Work for Grad Students

• 11 groups formed of 15 graduate students
• Provided:
  – **Software:** Alpha version of Phybots that only provides localization and locomotion API
  – **Hardware:** Robot kit
    • Camera
    • Robot
    • Visual markers
Results & Lessons Learned

- Lessons learned:
  - Mere mobility was not enough.
  - Most apps only used a single API call.
  - Parameter configuration was painful.
Results & Lessons Learned

- Lessons learned:
  - **Extendibility is important.**
  - Most apps only used a single API call.
  - Parameter configuration was painful.
Results & Lessons Learned

- Lessons learned:
  - **Extendibility is important.**
  - **Higher-level task management is desired.**
  - Parameter configuration was painful.
Results & Lessons Learned

• Lessons learned:
  – Extendibility is important.
  – Higher-level task management is desired.
  – Support for testing phase is needed.
Workshop for HCI Students

• 3 undergrad and 4 grad students from HCI labs
• Provided:
  – **Software**: Current version of Phybots
  – **Hardware**: Robot kit
    • Same as the previous study except for the robot.

Results

1. Miniature drive recorder
2. Beach flags with obstacles
3. MatereARdrone
4. Cameraman robot
5. Alarm clock
6. Hawk view
7. Serving robot
Lessons Learned

• Feedback for robotics:
  – Effectiveness of GUI for controlling robots (while typical Human-Robot Interaction focuses on direct interaction between people and robots.)

• Inner class madness:
  – Chains of event listeners resulted in spaghetti code.
  – Workflow API helped cleaning the code.
Workshop for Robotics Students

• 2 graduate students from a robotics lab
• Provided:
  – Same as the previous study
• Asked:
  – Comparison between Phybots and a famous robotics toolkit (ROS)
Lessons Learned

• Too much abstraction is not good for prototyping
  – Boilerplate
    • Many lines of code for configuration
  – No direct access to the world model
    • Many lines of code to get a specific class instance
  – Too many things to learn before coding
    • Many modules with tiny purposes
Future Work for the Toolkit

• Tighter integration with toolkits for robotics
  – Phybots can be implemented as a module for ROS.

• Kinect camera instead of a normal webcam
  – Phybots can benefit from its depth/skeleton info.

• Support for local tasks
  – Two-dimensional coordinates was sufficient for global locomotion but for local tasks such as picking up an object.
Phybots

• A toolkit for prototyping “robotic things”
  – Localization and Locomotion API
  – Extensible Architecture
  – Runtime Debug Tool

• Open-source software available at http://phybots.com