Blocks Languages for Creating Tangible Artifacts

Lyn Turbak & The TinkerBlocks Team
Computer Science Department
Wellesley College

New England Programming Language Seminar
June 1, 2012
TinkerBlocks Team

Erin Davis
Smaranda Sandu
Olivia Kotsopoulos
Karishma Chadha
Emily Erdman
Johanna Okerlund
Marie Vasek

NEPLS June 1, 2012
Overview

- The big picture:
  - What are blocks programming languages?
  - Who/what are they good for?
  - Why should you care about them?

- What we’ve done:
  - TurtleBlocks: Logo turtles
  - PictureBlocks: Henderson’s picture language
  - Better handling of names in blocks languages
  - Enable creating tangible artifacts with these environments
What Are Blocks Languages?
Who/What Are They Good For?

- Novices learning programming
- Occasional programmers
- Understanding programming language features
  - Statements vs. expressions
  - Procedures and invocations
  - Naming
  - Typing (Marie Vasek’s talk)
- Democratizing programming in interesting domains
  - Animations/games (Scratch)
  - Smartphone apps (App Inventor)
  - Multi-agent simulations (StarLogo TNG)
  - Robotics (PicoBlocks)
  - Microprocessor programs (ModKit)
  - Tangible artifacts (TurtleBlocks & Picture Blocks)
- Personal programs
Why Care: Popularity

**Scratch:** 5 million downloads, 2.6 million projects shared, 330,000 sharing users

**MIT App Inventor:** 195,000 users and growing

**StarLogo TNG:** 120,000 downloads

**Blockly:** Being developed as user PL for Google products

**Also:** BYOB/Snap, Panther, WebLogo, TaleBlazer, TurtleArt, PicoBlocks, ModKit, WaterBear, ...
Blockly (Neil Fraser @ Google)

http://neil.fraser.name/software/blockly/demos/code/
Why Care: Visualizing Features

Example: choose block (if expression) in TurtleBlocks
Why Care: Confusing Features

Example: Procedure Parameters in StarLogo TNG
Variable Scope in TurtleBlocks
Rapid prototyping @ Wellesley

Laser cutter

3D Printer

Desktop vinyl cutter

NEPLS June 1, 2012
TurtleBlocks

TurtleBlocks program

turtle drawing

cardstock

acrylic

drawing boundary

NEPLS June 1, 2012
Constructive Area Geometry

$J$ $K$ $\text{union}(J,K)$ $\text{difference}(J,K)$
TurtleBlocks Artifacts
Peter Henderson’s Picture Language

- Described in his paper “Functional Geometry” (1982)
- Popularized in Abelson and Sussman’s *Structure and Interpretation of Computer Programs*
- Used in Wellesley College Introduction to Programming course
Picture Combinators

- \texttt{bw}
- \texttt{clockwise90(bw)}
- \texttt{flipHorizontally(bw)}
- \texttt{flipVertically(bw)}
- \texttt{above(bw,rp)}
- \texttt{beside(bw,rp)}
- \texttt{overlay(leaves,kite)}
- \texttt{overlay(kite,leaves)}
PictureBlocks: Cutting

PictureBlocks program

resulting picture

acrylic artifact

picture boundary

NEPLS June 1, 2012
PictureBlocks: Sketching & Engraving

user sketch → PictureBlocks program → resulting picture

print from engraving → wood engraving
PictureBlocks: Engraving + Cutting
Next Steps

- This summer: improving blocks programming in App Inventor
  - fix broken StarLogo-like procedure parameters
  - add local variables
  - converting between blocks and text languages
  - copying blocks between programs.

- Porting TurtleBlocks & PictureBlocks to a web-based blocks environment.

- Environments for 3D artifacts and popups

- Support for debugging and visualization of dynamic program execution in blocks languages.

- Blocks language generators
Questions?
PictureBlocks: fourSame Procedure
...options
PictureBlocks: Knitting Program
PictureBlocks Designs