Looking Glass: Supporting Learning using Peer Programs

Caitlin Kelleher
ckelleher@cse.wustl.edu

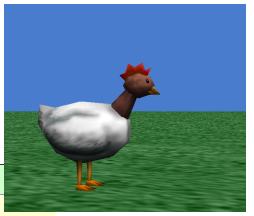
Lack of diversity can create unintentional problems.

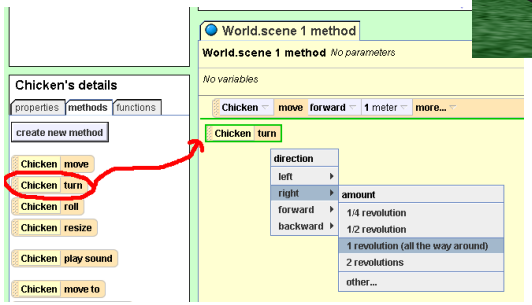
How do we get a broader group of people into computer science?

One piece:
remove the frustration.

Alice 2

Primary goal: Create a drag and drop code editor that prevents syntax errors.



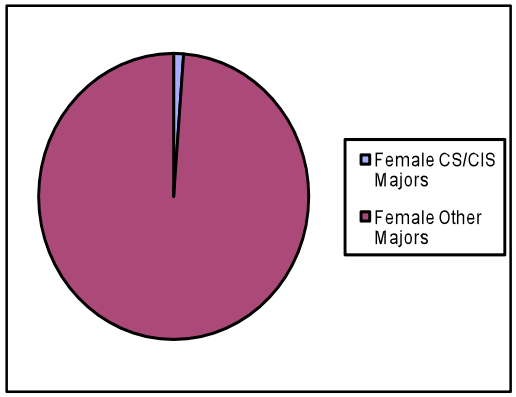


Alice 2 helps retain CS majors

Declared CS majors at Ithaca College and St. Joseph's University	CS1 Grade	Take CS2?
No Alice Class Prior to CS1	C	47%
Alice Class Prior to CS1	B	88%

But these students already wanted to major in computer science.

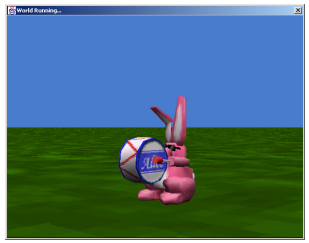
And, there aren't too many of them.



Middle School is potentially a high-leverage window

- Many girls turn away from math and science disciplines, including computer science, during middle school.
- Once they decide to leave, it is difficult to get them back.

For Middle Schoolers Easy Isn't Enough



"OK, so I can make the bunny move around. **But why would I want to?"**

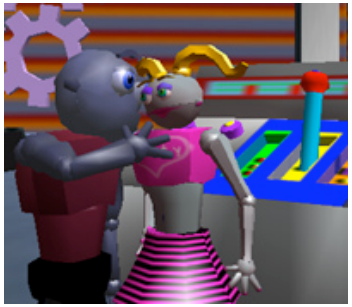
-12 yr old girl

No matter how easy something is, people still need a reason to want to do it.

Why should middle school kids program?

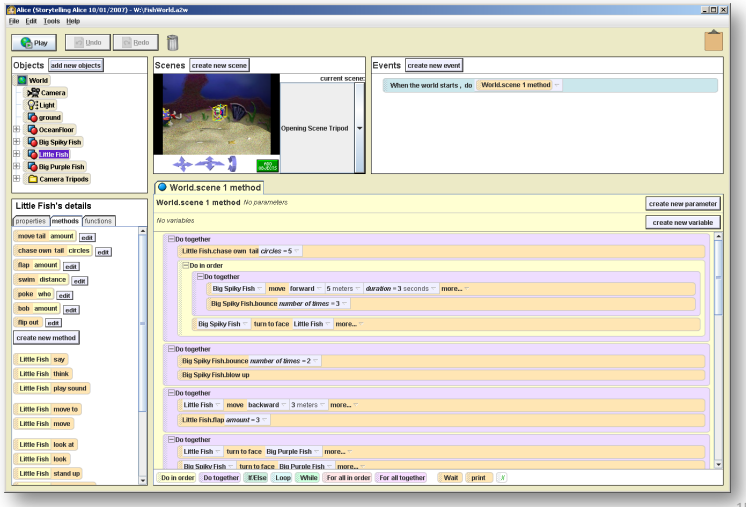
Another piece:
Embed programming in
a motivating context.

Storytelling Alice



Approach: Present programming as a *means* to the *end* of storytelling.

Storytelling Alice Demo



Enabling Storytelling

1. Add high-level animations that enable social interaction
2. Create a story-based tutorial.
3. Provide a gallery of characters and animations that inspire stories.

Generic Alice :
All objects are created equal.

Generic Alice:
All objects are created equal.

Objects can:

- move
- turn, roll
- resize
- play sound

Generic Alice:
All objects are created equal.

Objects can:

- move (translate)
- turn, roll (rotate)
- resize (scale)
- play sound

But….
It’s all about the people.

Storytelling Alice:
Focus on humanoid characters

People need to:

- communicate
- loco mote
- change posture
- attend to
- interact with

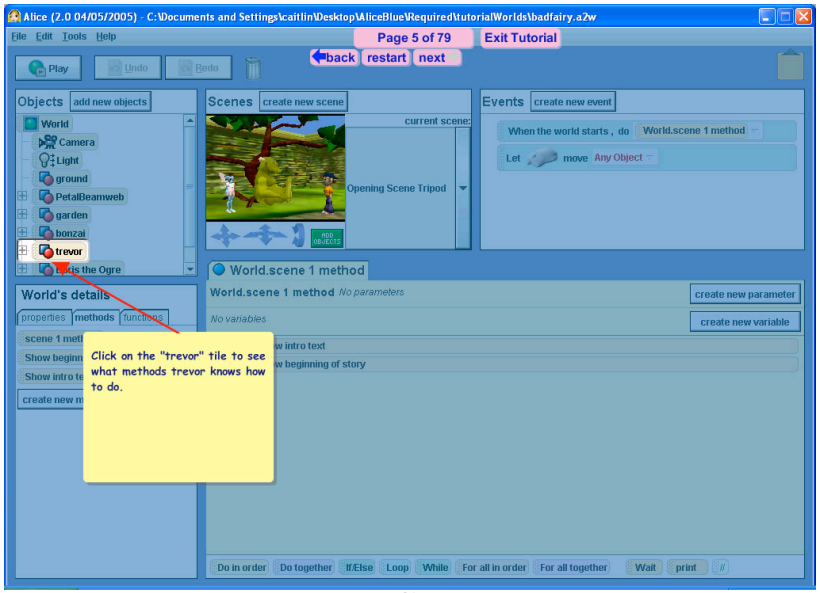
Provide custom animations that require explanation in the story

Harold T. Wireton.crazy go nuts

Animations can be incredibly powerful in helping kids to come up with a story idea.

Generic Alice:

Storytelling Alice:



Does storytelling help?

Isolate Storytelling Focus

Generic Alice
(aka "Alice Green")

vs.

Storytelling Alice
(aka "Alice Blue")

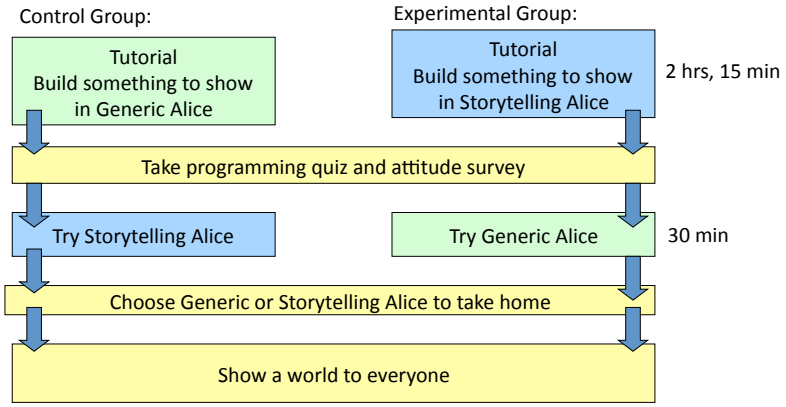
Keep the mechanical supports for programming constant.

Get Representative Subjects

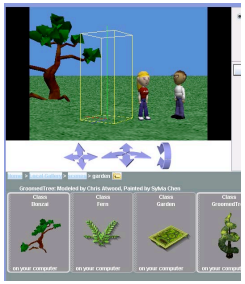
88 Girl Scouts from within 1.5 hours of Pittsburgh

Troops used participation as a fundraiser.

Evaluation Workshop Structure



Three Activities in Alice

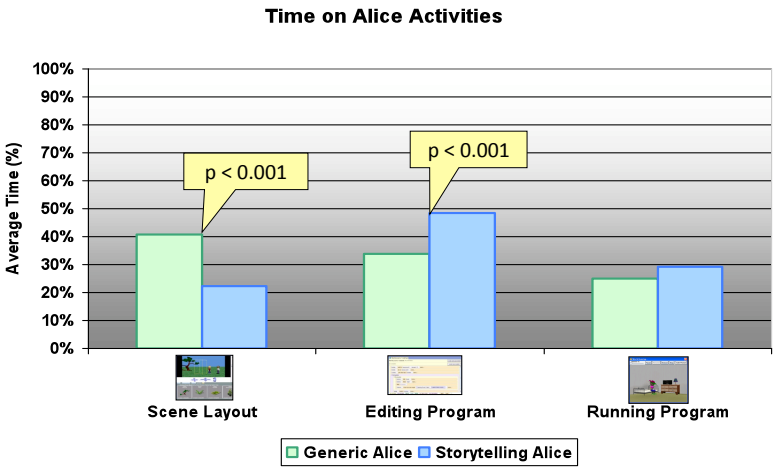


1: Scene Layout

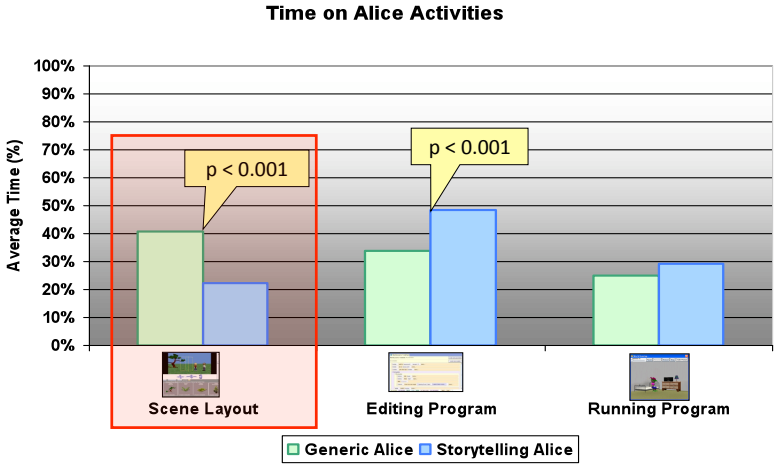
2: Editing Programs

3: Running Programs

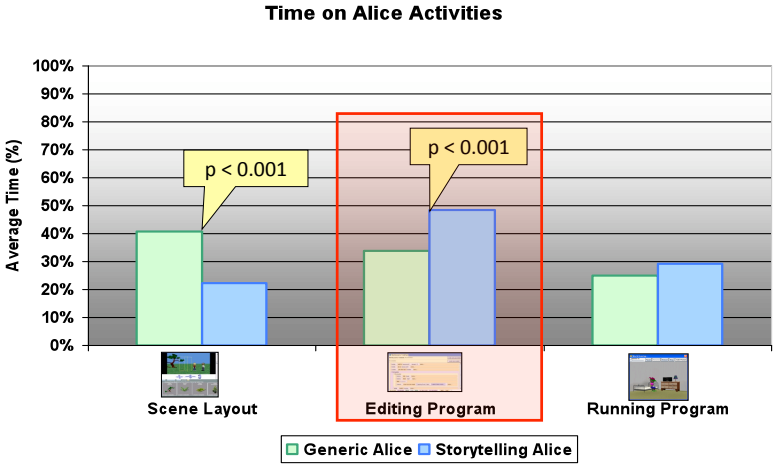
Storytelling Alice users spend 42% more time programming



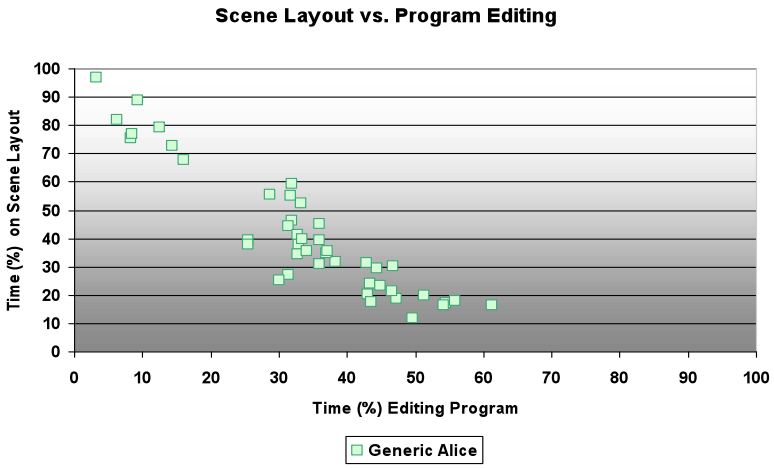
Storytelling Alice users spend 42% more time programming



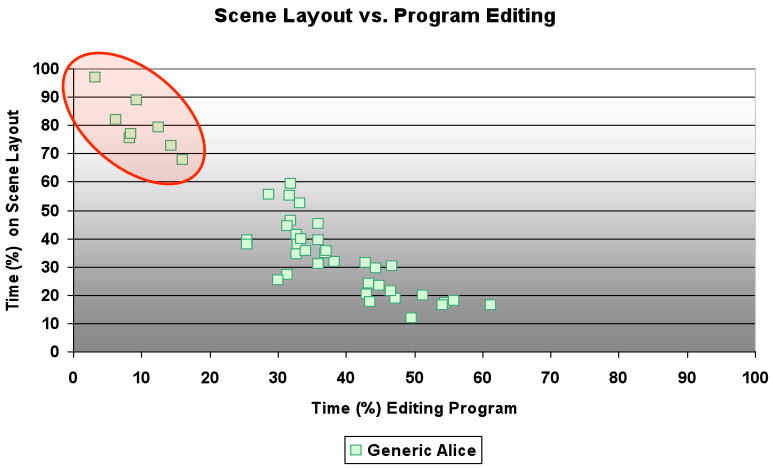
Storytelling Alice users spend 42% more time programming



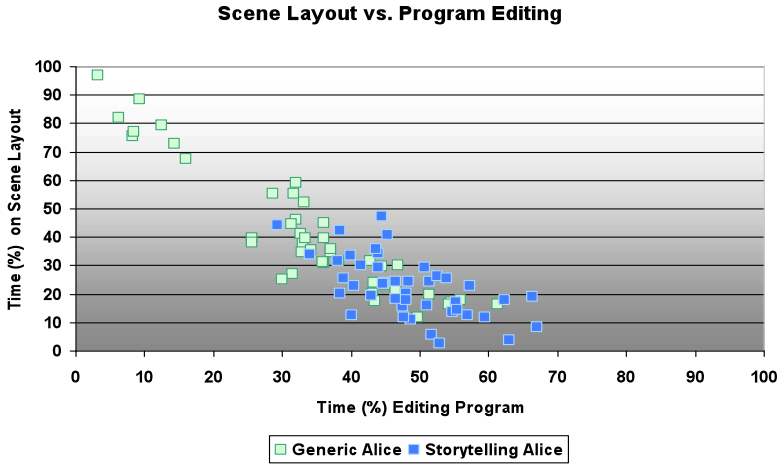
Storytelling Alice motivates reluctant programmers



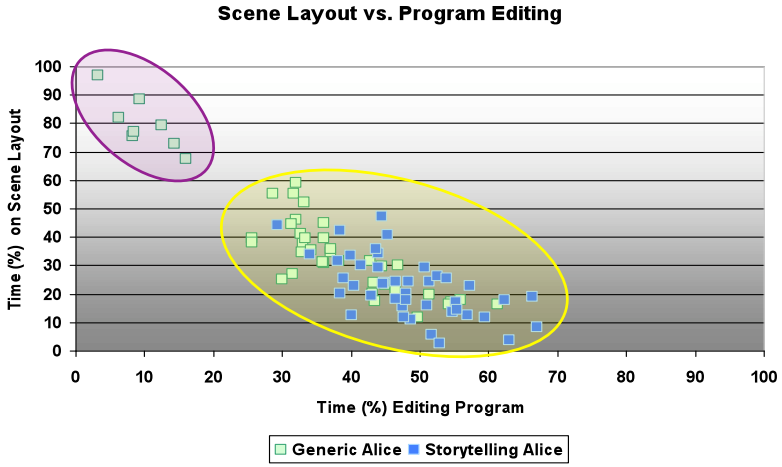
Storytelling Alice motivates reluctant programmers



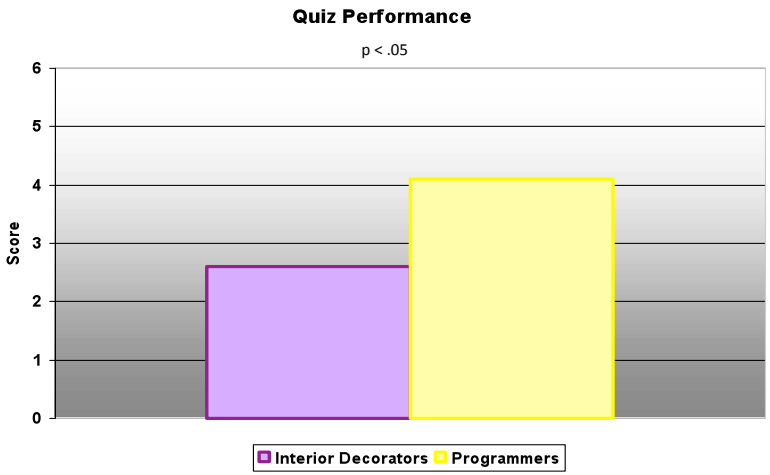
Storytelling Alice motivates reluctant programmers



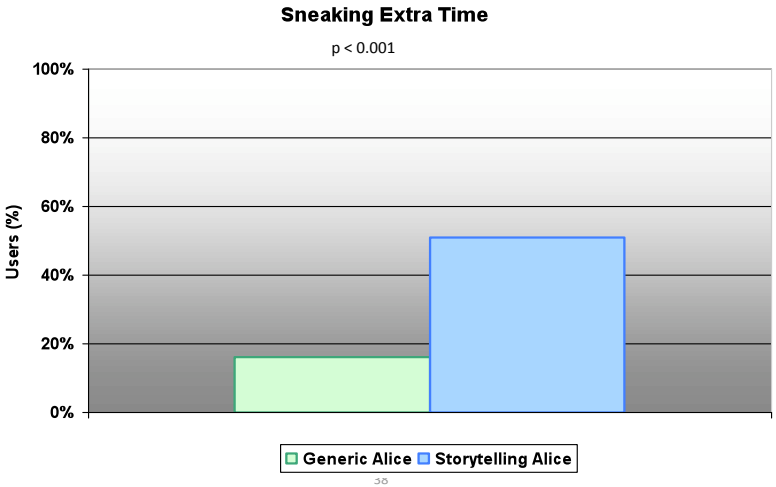
Time on task is a strong predictor of learning



Time on task is a strong predictor of learning



Users of Storytelling Alice are more likely to sneak extra time to continue programming.



Increased engagement isn't enough either.

An All-Too Common Scenario

1. Ashley downloads Storytelling Alice.
2. She builds a small story and gets excited.
3. She begins planning a larger scale project.
4. Early on, she gets stuck.

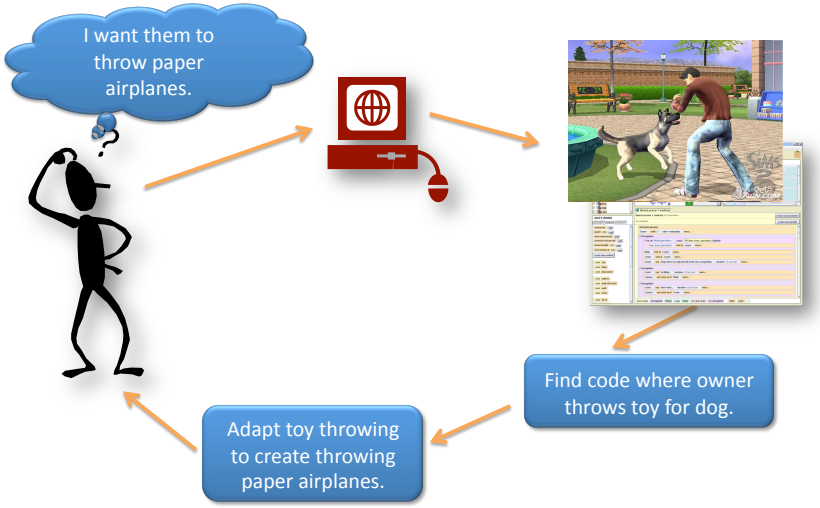
Looking for help.

- Parents/Friends
  - Few have experience with computer programming.
- Teachers
  - CS is rarely taught at the middle school level.
- Internet
  - Some tutorials, but finding one related to a specific question is difficult.

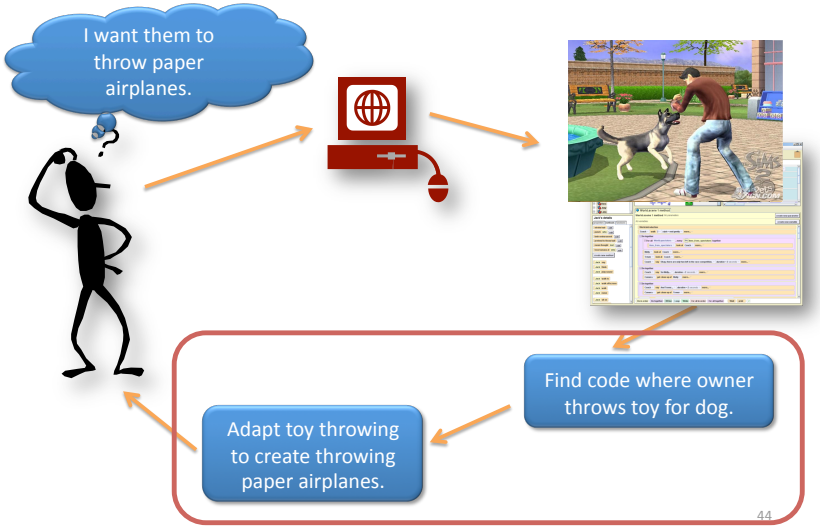
Looking Glass

- Focus on enabling kids to teach themselves in pursuit of their own goals.
- Keep the storytelling and social interaction motivation.
  - Using models learned from a motion library to make it easier to generate appealing procedural animations for humans.

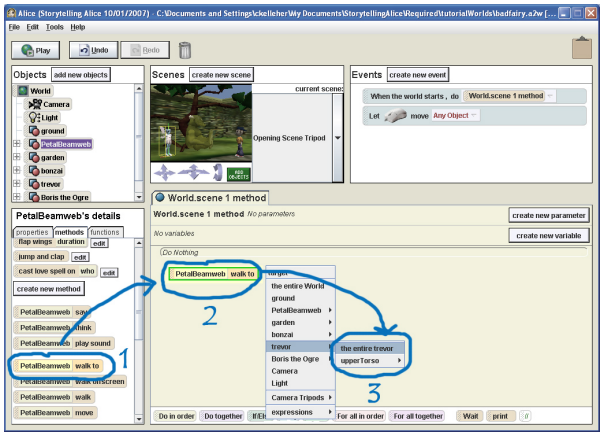
A Looking Glass Scenario



A Looking Glass Scenario

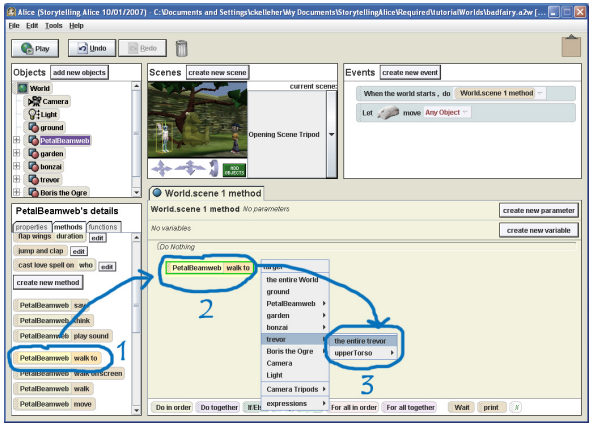


Pre-req: Record high level UI actions



1. Click and drag “Petal Beamweb walk to”
2. Drop it at the top of scene 1 method.
3. Choose “trevor” and “the entire trevor” from the menus.

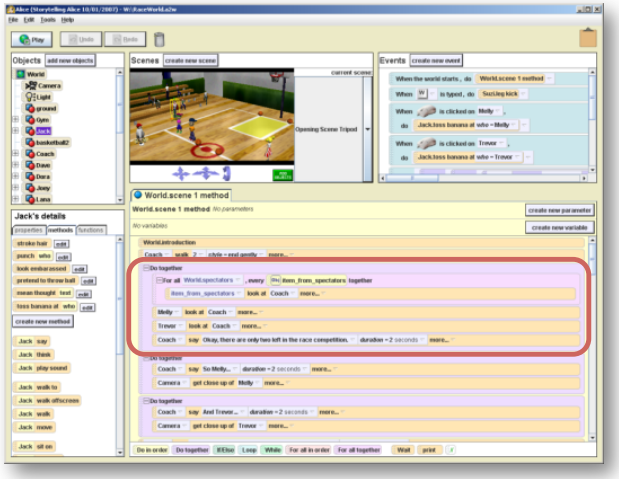
Pre-req: Record high level UI actions



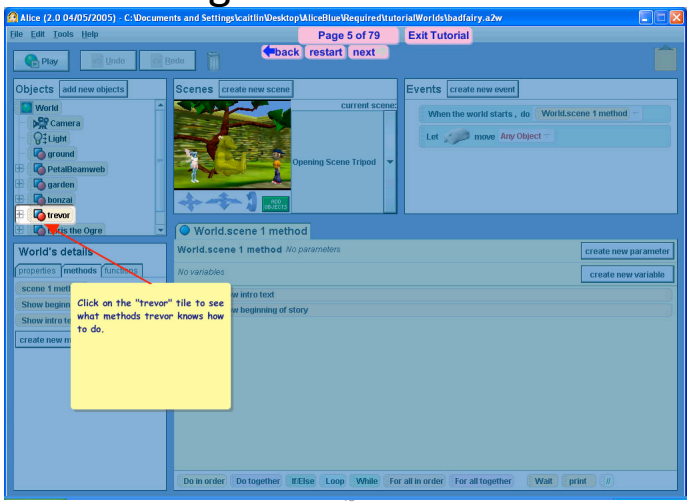
1. Click and drag “Petal Beamweb walk to”
2. Drop it at the top of scene 1 method.
3. Choose “trevor” and “the entire trevor” from the menus.

Every program knows the UI steps necessary to create itself.

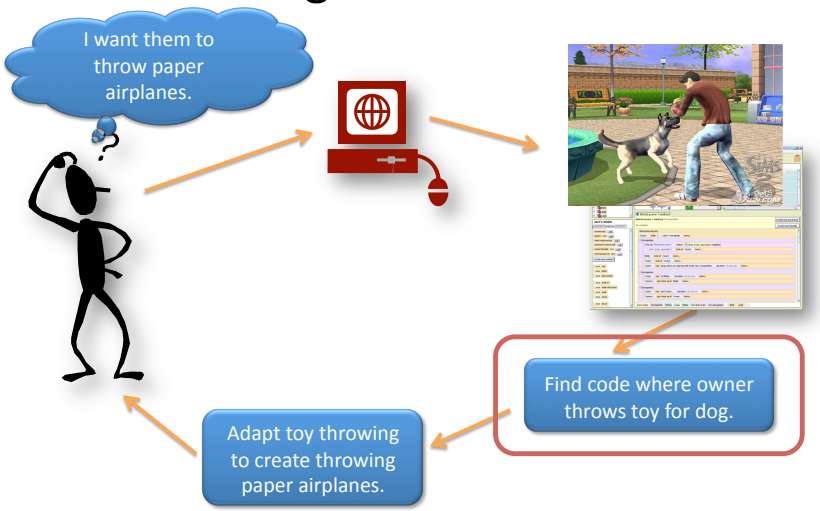
Provide UI tools to help users find code they want to use in other’s programs



Use history for selected code to generate tutorial



A Looking Glass Scenario



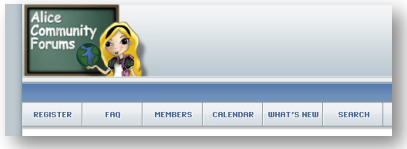
How do novice programmers approach finding code in unfamiliar programs?

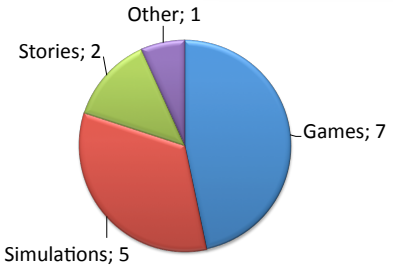
(with Paul Gross)

First, what are the properties of programs they are likely to find?

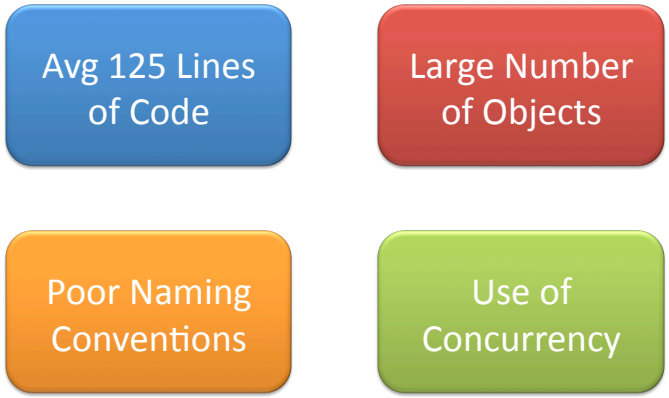
Finding Properties of Web Examples

- Randomly selected 15 programs from Alice.org forums

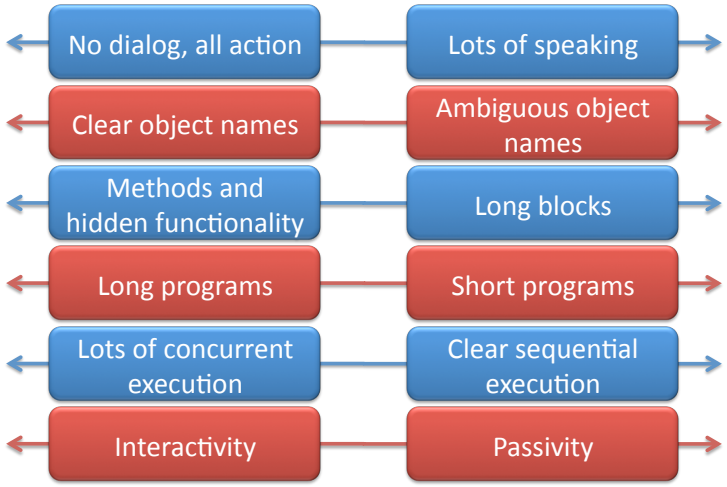




Some Observed Properties of the 15 Sample Programs



Design Dimensions Considered in Writing Programs for Search

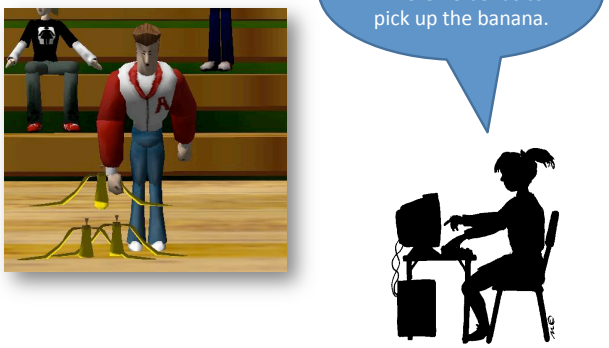


Programs

Fish World	No text, all actions, clear object names
Woods World	Three concurrent 'parts' (methods) timed to appear sequential
Magic Trees	Long program, three concurrent blocks in one method, hidden method functionality
Race World	Interactive program, lists, events, randomness

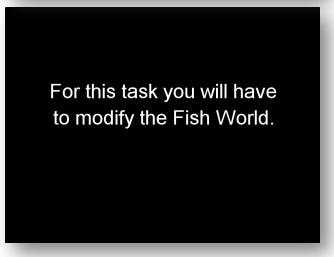
How will users use example programs they find?

Expected Use



Tasks Mirror Expected Use

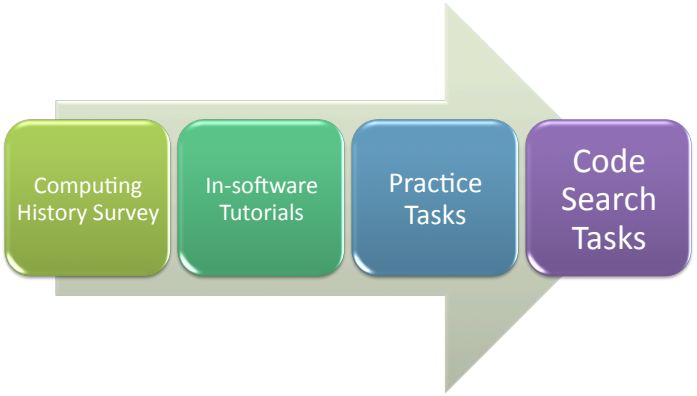
- Bounding Tasks
  - Denote begin, end of highlighted functionality
- Modification Tasks
  - Modify highlighted functionality
- 5 tasks for each of the 4 programs.



Participants

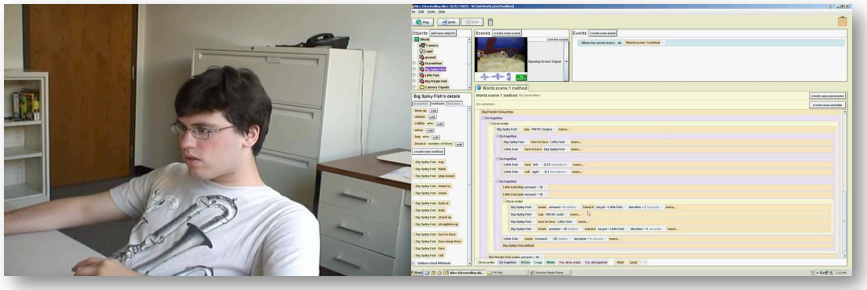
- 14 adults from the Washington University community
  - Mostly staff (summer break?)
- 12 had no prior programming experience of any kind
- 1 had limited experience with Fortran 20+ years earlier
- 1 had some experience with Matlab 6+ years earlier.

Code Search Study Process



Code search tasks were randomly chosen.
Participants completed as many as they could in the remaining time.

Task Data Collection



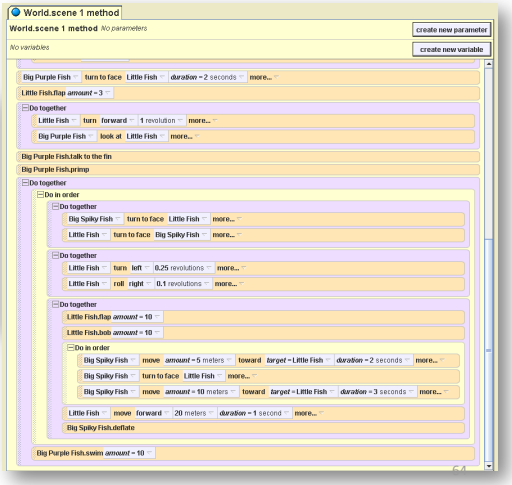
- Video/audio recording of users, screen
  - Talk-aloud protocol

This is hard for novices.

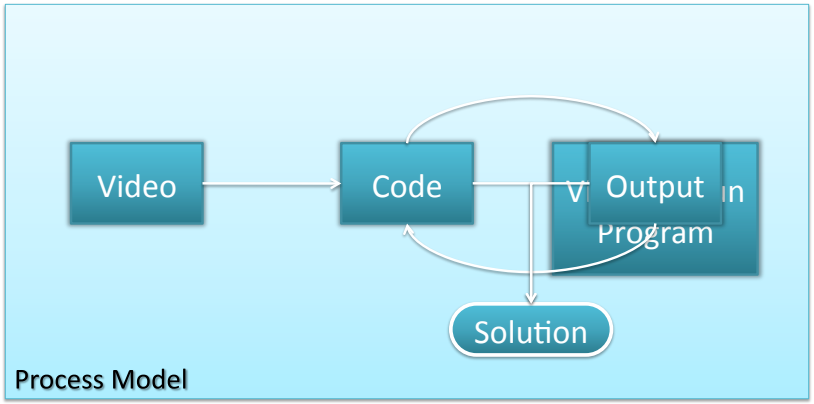
- 41% correct answers
  - 33% correct bounding
  - 72% correct modification
- Bounding task time range: 01:02 – 26:20
- Modification task time range: 01:07 – 27:29

How do novices approach searching for code?

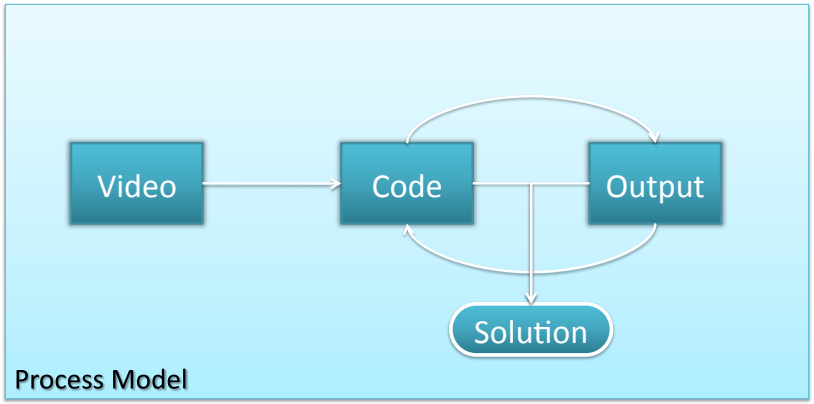
What Users Do



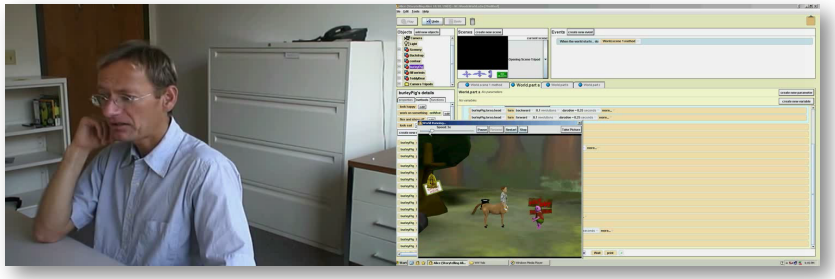
Task Process Model



Task Process Model



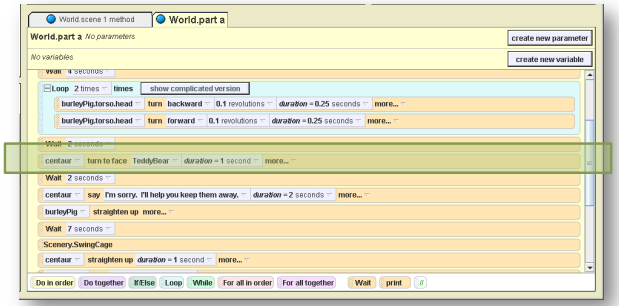
What Users Focus On: Landmarks



Landmark

- A specific feature in output or code

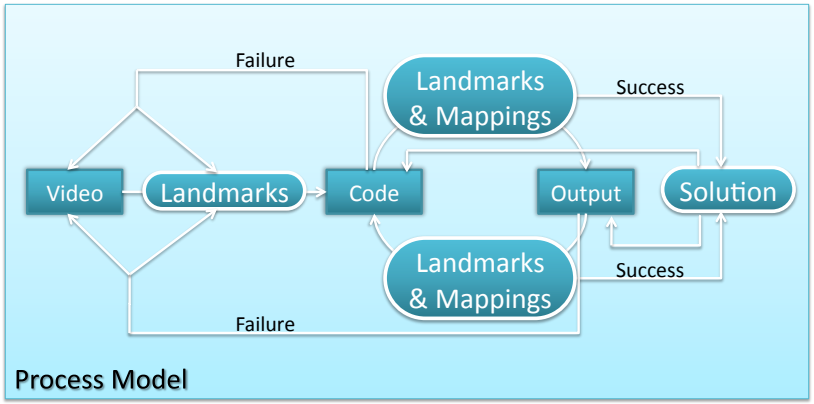
What Users Try to Find: Mappings



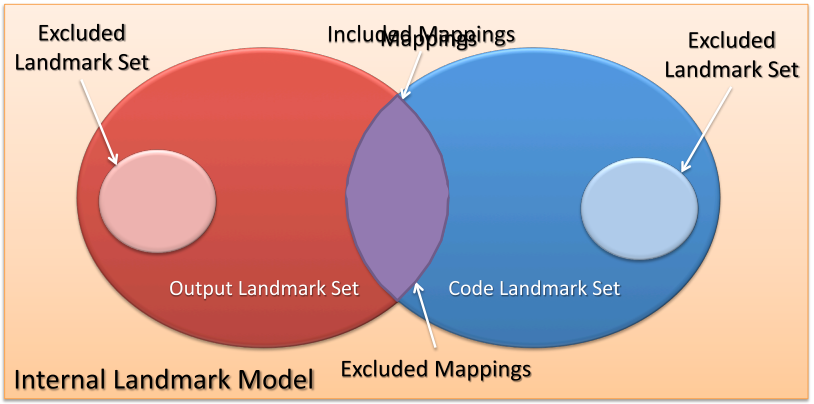
Mapping

- A direct connection between a line of code and output function

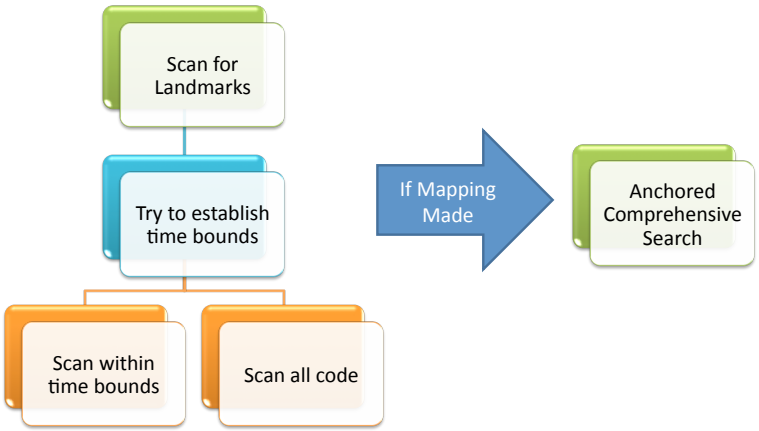
Task Process Model



Internal Landmark Model



Searching for identified landmarks



Handling Search Failure

- Go back to the output, look for some new landmarks and try again.
- As the frustration really sets in….
  - Look at the API for the character of interest and hope that helps identify a new landmark
  - Start randomly clicking around the interface and hope.

Specific Challenges

- Building mappings is often extremely inefficient.
- Users overlook locations that may contain their target code.
- Novice users do not understand programming constructs.
  - Can lead to errors in temporal reasoning
- Users will include/exclude code based on method naming
  - Likely that they will be searching a program with poorly named methods

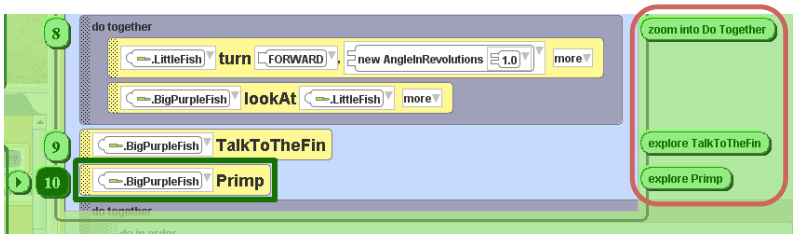
A new tool for helping novices find target code

(Demo)

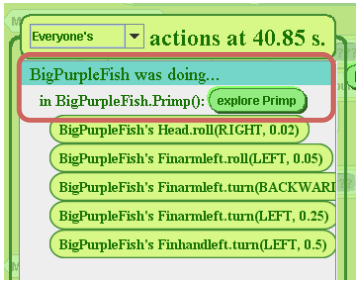
Making Connections Between Code and Output

Making Connections Between Code and Output

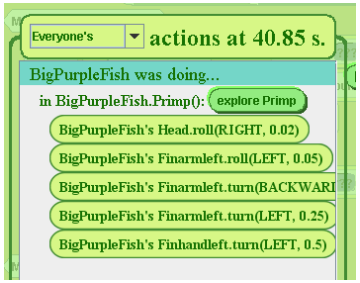
Helping users know where to look

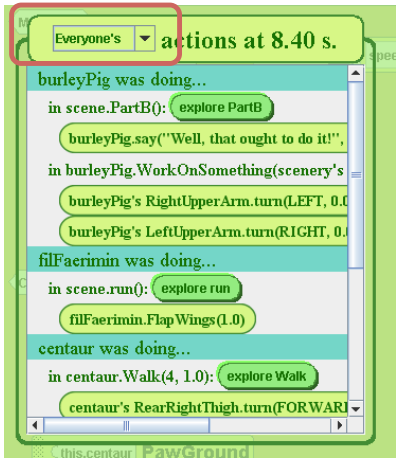


Helping users know where to look

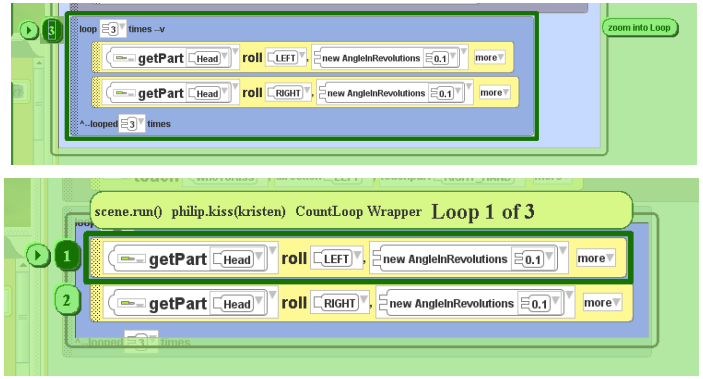


Helping users know where to look

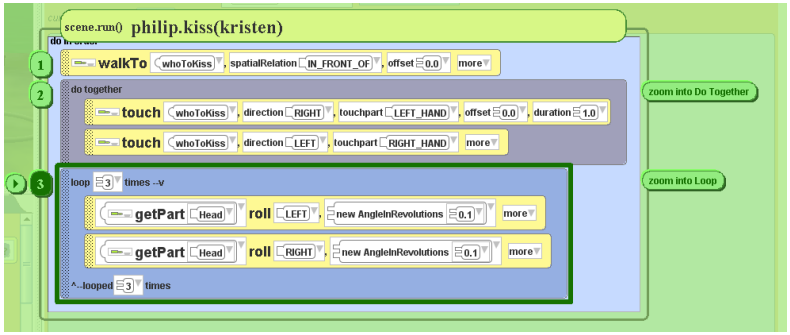




Focusing attention can help users build an understanding of constructs



Focusing attention can help users build an understanding of constructs



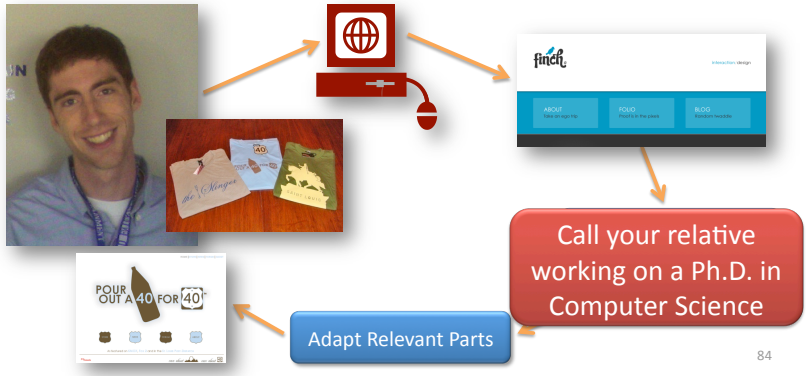
Next Steps

- Formal evaluation of the code finding tool
- We're seeing some natural independent learning of programming constructs
  - Can we use deconstruction as a way to teach basic concepts?
- Connecting in the ability to generate the tutorials based on the UI action history.

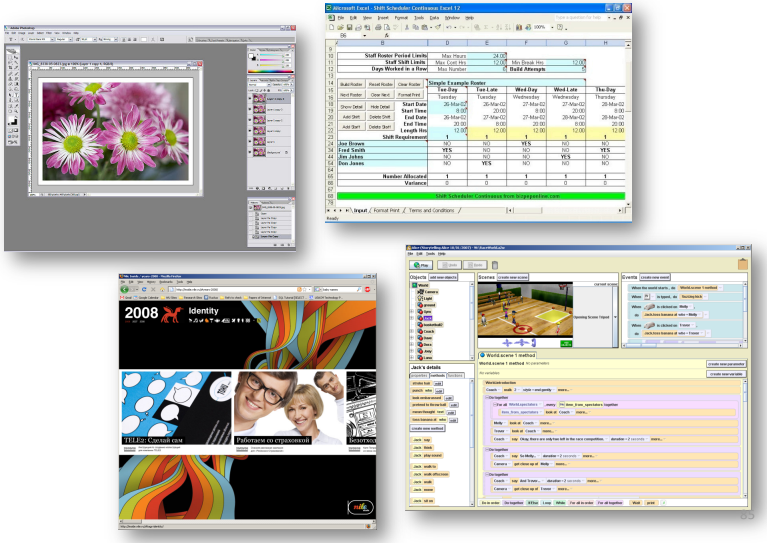
Other applications for these kinds of tools.

Personal Anecdote (web)

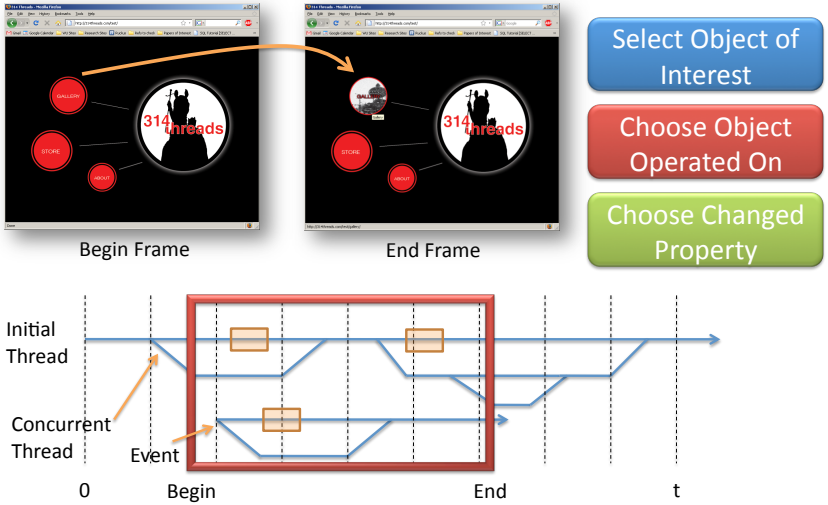
- Finding functionality in unfamiliar code
- Adapting functionality from unfamiliar code



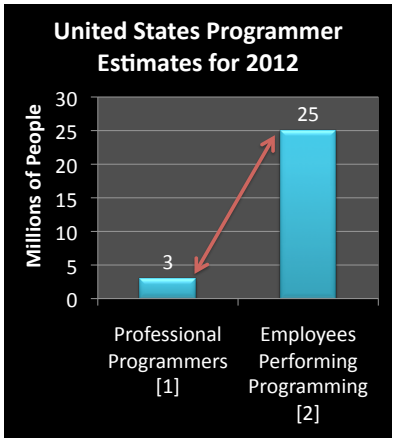
End-Users In and Beyond the Workplace



Idea: Timeline Search with Filtering



The End-User Programming Gap



- Workers benefitting from programming
- Lack formal training
- Heavy use of examples

[1] Hecker, D. Occupational Employment Projections to 2012. Monthly Labor Review, 127, 2, 2004, 80-105.

[2] C. Scaffidi, M. Shaw, and B. Myers. Estimating the Numbers of End Users and End User Programmers. VL/HCC 2005, Dallas, TX, September 2005, pp. 207-214.

Questions?

Download Storytelling Alice from www.alice.org