User Interfaces for Live Programming

Jun Kato
https://junkato.jp
Researcher, AIST

LIVE 2017 Keynote, 10/24/2017
Jun Kato  
@junkato  
https://junkato.jp

Research Topic

**Computer Science** (Human-Computer Interaction, Programming Language)

- **Phybots**  
  ACM DIS'12

- **DejaVu**  
  ACM UIST'12

- **Picode**  
  ACM CHI'13

- **It’s Alive!**  
  Live programming ex Navigate to code without
  ACM PLDI'13

- **VisionSketch**  
  GI'14

- **TextAlive**  
  ACM CHI'15

- **f3.js**  
  ACM DIS'17

- **Created Tools and Environments for Creativity/Productivity Support**

- **Application Domains:** Prototyping, Physical Computing, Computer Vision, Robots, Internet of Things, Animation Authoring, ...

- **Founded **SIGPX** (SIG on Programming Experience)** [https://sigpx.org/en](https://sigpx.org/en)
A group of researchers/engineers/teachers in Japan, studying ...

Programming Experience in the intersection of HCI/PL/SE

Meet & Discuss

Collect

Publish

PX Special Issue in IPSJ Journal (Nov 2017)
Emerging Research on Programming Experience: From Language Design to Industrial Applications
Today, I’m going to talk about …

- What is Live Programming?
- UIs for Live Programming with end-users
- UIs for Live Programming of this material world
- UIs for Live Programming with time travel
- Live Programming as User Interface research

<table>
<thead>
<tr>
<th>It is about …</th>
<th>It is not about …</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Showcase of user interfaces for programming</td>
<td>• No ( \lambda ) or greek symbols in slides</td>
</tr>
<tr>
<td>• Not only my work but also others’ notable work</td>
<td>• Not a consensus in the field (it's ongoing!)</td>
</tr>
<tr>
<td>• Discussion on live programming system design</td>
<td>• No peer review involved (my personal view)</td>
</tr>
</tbody>
</table>
Today, I’m going to talk about ...

• What is Live Programming?
• UIs for Live Programming with end-users
• UIs for Live Programming of this material world
• UIs for Live Programming with time travel
• Live Programming as User Interface research
What is Live Programming?

- Programming experience
- Continuous feedback
- Concrete information
- Early examples in VPL and OOP
- Attracting much attention these days

Not new but hot!
Text-based Programming

Dartmouth BASIC [1964]

- Text-based editor
- Text-based debugger
- Text-based ...

Visual Studio Code [as of today]

IDEs haven’t changed much

User Interfaces for Live Programming
With some exceptions ...

**Interface Builder**
- A tool for NeXT UI development
- Later integrated into Xcode
- Many IDEs have similar built-in tools

[since 1986]
Visual Programming

- Mostly dealing with symbolic representations of programs
- Often considered as tools for novices and good for education
- Dataflow languages: early examples of live programming

 Scratch
 MIT

 ConMan
 Haberli
 [SIGGRAPH 1988]
Character-based UIs or Graphical UIs?

Modern Live Programming as a hybrid approach
User Interfaces with Text and Graphics

TouchDevelop
[PLDI 2014]
https://touchdevelop.com
User Interfaces with Text and Graphics

User Interfaces for Live Programming

f3.js
[DIS 2017]
http://f3js.org
Character-based UIs and Graphical UIs

- It's like text and figures in research papers
- Text is good at abstraction
- Graphics are good at presenting concrete information

Integrated Graphical Representations
[2014, dissertation] [2016]

They complement each other
In Live Programming systems, we ...

• first have **vague ideas**
• then explore the ideas with **concrete examples**
• gradually start turning the ideas into **programs**

Live Programming requires decent UIs for exploring the problem space
UIs for Live Programming should ...

- avoid sudden changes in the program behavior
- keep the program and its output relevant
- allow continuously exploring the problem space

Appropriate user interface design differs from application to application
When designing live programming systems ...

Don’t be afraid to be domain-specific

• Good UI is always specifically designed for the target domain
• It might be like replaying the history of end-user computing in the domain of programming
• We might need PX workbench (cf. language workbench)

Cf. Programming eXperience Toolkit (PXT)
https://github.com/microsoft/pxt
Today, I’m going to talk about ...

• What is Live Programming?
• **Uls for Live Programming with end-users**
• Uls for Live Programming of this material world
• Uls for Live Programming with time travel
• Live Programming as User Interface research
UIs for Live Programming

Good mixture of text-based and graphical user interfaces
Uls for Live Programming

Why not expose GUI to users so that they can edit programs?
Mode Switch between “Use” and “Build”

What if we add another layer for users?

Promoting universal usability with multi-layer interface design

Ben Shneiderman [2002]
User Interfaces for Live Programming
User Interfaces for Live Programming

Co-hosting UIs for programmers and users

Literate Programming in Jupyter (Ipython) Notebook

Inline Photos in Picode [CHI 2013]
Merging UIs for programmers and users (direct manipulation on program output)

Sketch-n-Sketch, Hempel et al.  [UIST 2016 etc.]

Para, Jacobs et al.  [CHI 2017]
When designing live programming systems ... 

How about making the ladder of expertise?

- From live programming for programmers
- To live programming for the community of people
Today, I’m going to talk about ...

- What is Live Programming?
- UIs for Live Programming with end-users
- **UIs for Live Programming of this material world**
- UIs for Live Programming with time travel
- Live Programming as User Interface research
What is “live” and what not?

- System response time:
  - Computation
  - Network
  - Touch display response

- Reflex time:
  - Visual 0.25s
  - Audio 0.17s
  - Touch 0.15s
  - and more ...
    - 3D printers and laser cutters
    - Shape changing devices and robots
    - Taste/smell interfaces

How Live are Live Programming Systems?
Rein et al. [PX 2016]

Various kinds of “latencies”
Printing physical computing devices

Slow “framerates” prevent live feedback

User Interfaces for Live Programming
Slow display

Daniel Saakes et al. [SIGGRAPH Etech 2010]

Slow “framerates” can be useful, too

User Interfaces for Live Programming
Provide the source code of a microcontroller or tiny computer in JavaScript. Node.js-based computers are supported. Require f3.js package and use its API to design the device enclosure.

```javascript
var WebSocket = require('ws');
var serverAddr = 'ws://192.168.10.100:8080/entry';
var ws = new WebSocket(serverAddr);
var upmBuzzer = require('jsupm_buzzer');

var numBuzzers = 5; // Number of buzzers [1, 5]
var inputs = [3, 5, 6, 9];
var buzzers = [];
var volume = 0.1;
var r = 80;

var f3js = require('f3js');
var c = f3js.createContainer();
c.x = 50;
c.y = 50;

var c1 = 43,
c2 = 36,
x = 70,
y = 140,
width = 70;
var a = c.createPath();
var ps = a.extrude(60);
ps[0].y = 50;
ps[0].x = 270;
f3js.addPath(ps);

var cc = f3js.createContainer();
var nn = cc.createPath();
```

f3.js: IoT apps with enclosures from single code base

MakeCode for BBC micro:bit, Microsoft Research [2017]
http://makecode.microbit.org
if (pose.eq(1)) {
    showText("Got the command!");
    nxt.setPose(0);
}
When designing live programming systems ...

Deceiving users’ perception is a good thing

• As long as the lie is reasonable
• The actual “framerate” can be very slow
• Emulating, or sometimes even pretending, is needed to provide the continuous feedback
• Make full use of five senses in programming environments
Today, I’m going to talk about ...

• What is Live Programming?
• UIs for Live Programming with end-users
• UIs for Live Programming of this material world
• **UIs for Live Programming with time travel**
• Live Programming as User Interface research
“Live” is not always about “now”

• UIs for exploring and modifying the past
• UIs for predicting and choosing the future
• Absolute vs semantic timeline
float sumDistance = 0;
foreach (Joint joint in skeletonData.Joints) {
    sumDistance += joint.Position.Z;
}
float userDistance = sumDistance / 20;

// If the user is close enough to the camera, show the virtual clothes
bool userIsNear = userDistance < 2.5;
ShowStage(userIsNear);
float sumDistance = 0;
foreach (Joint joint in skeletonData.Joints) {
  sumDistance += joint.Position.Z;
}
float userDistance = sumDistance / 20;

// If the user is close enough to the camera, show the virtual
bool userIsNear = userDistance < 3;
ShowStage(userIsNear);
VisionSketch [GI 2012]
“Live” is not always about “now”

• UIs for exploring and modifying the past
• UIs for predicting and choosing the future
• Absolute vs semantic timeline
Light Table – a new IDE concept, Chris Granger [2012]
Many-Worlds Browsing for Control of Multibody Dynamics
Twigg et al. [SIGGRAPH 2007]
User Interfaces for Live Programming

Rktcr, McCann (TCHOW) [2013]
“Live” is not always about “now”

• UIs for exploring and modifying the past
• UIs for predicting and choosing the future
• **Absolute vs semantic timeline**
Absolute time vs semantic time

DejaVu [UIST 2012]

TextAlive [CHI 2015]
When designing live programming systems ...  

Try providing good sense of time

- Enable time travel to find critical timings in the history
- Allow editing the code and program input to explore futures

Replay & Refresh
Superspeed & slowmo

“Many worlds”
Stroboscopic visualization

Timeline
for absolute/semantic time

User Interfaces for Live Programming
Today, I’m going to talk about ... 

• What is Live Programming?
• UIs for Live Programming with end-users
• UIs for Live Programming of this material world
• UIs for Live Programming with time travel
• Live Programming as User Interface research
Live Programming research as User Interface research

• Don’t be afraid to be domain-specific
• How about making the ladder of expertise?
• Deceiving users’ perception is a good thing
• Try providing good sense of time

It’s **not only** about language design, a single user, a single UI, but about **designing the whole experience**
User Interfaces for Live Programming

Jun Kato

https://junkato.jp

Researcher, AIST

LIVE 2017 Keynote, 10/24/2017
References