CSE 440: Introduction to HCI
User Interface Design, Prototyping, and Evaluation

Lecture 10: Interface Implementation

James Fogarty
Eunice Jun
David Wang
Elisabeth Chin
Ravi Karkar

Tuesday / Thursday
10:30 to 11:50
Fogarty Adventures in Bad Visuals

Needed to present for UW Innovation Award

Needed a storyboard, but am visually inept

First experience with oDesk (now called Upwork)
Initial Specification

Unsure What to Do

Jane talks to her sister (could be changed to brother if it’s easier to get a distinct character here)

They had similar issue, for them it turned out to be a need to control stress and get more exercise

People talking about what’s causing Jane’s issue is a recurring thing, look ahead to see that

Stressed person icon is recurring, look ahead to see that
Guidance on Desired Style

Desired Style

What can I buy that is healthy and easy to prepare?

But not this black on white color, see color and presentation slide.
Version 1

Unsure What To Do
Version 2

Unsure What To Do
Version 3

Unsure What To Do
Unsure What to Do

- Consults Brother
- Had Similar Symptoms
- Shares his Triggers
- Stress & Exercise
Two Storyboards

Before Our Advances
After Our Advances

Three Iterations
Less Than Three Hours Time
Approximately $300
Before

Meet Jane

- Severe Symptoms
- Missing Work
- Needs Help
Before

Unsure What to Do

- Consults Brother
- Had Similar Symptoms
- Shares his Triggers
- Stress & Exercise
Tracking Mood and Physical Activity

- Buys a Fitness Band
- Tracks Mood
- Tracks Physical Activity
Before

Making Sense of the Data

Lots of Data
Mood Over Time
Activity Over Time
But No Understanding
Before

Maybe Her Doctor Can Help

- Did not Track Symptoms
- Did not Track Food
- Elimination Diet
- Difficult to Follow
- Lengthy Process
- Possibly Inconclusive
After

Revisiting Jane

Most Common Triggers
- Food
- Stress

Suitable Sensors / Apps
After

Appropriate Capture

- Tracks for a Baseline
- Automated Reminders
- Low-Burden Tracking
- Timely Symptom Input
After

Jane’s Personal Hypotheses

Possible Triggers
- Lactose
- Caffeine
- Stress
- Confounding Effect
Self-Experimentation

After

Self-Experimentation
Lactose
Caffeine
Jane Has Her Answer
After

Engaging Clinician with Data

Data is Actionable

Personalized Interventions
Fogarty Adventures in Bad Visuals

Needed to present for UW Innovation Award

Needed a storyboard, but am visually inept

First experience with oDesk (now called Upwork)

Presentation matters

In the real world, you can spend money on this
Project Status

Looking Forward

2g: Design Review (1x2) Due Tonight
Critique on how this is based in research, why this design, details of this design

2h: Getting the Right Design Report Due Monday

2i: Presentations next Thursday / Friday
Full progression from design research, themes, tasks, design exploration, design

Other Assignments

Reading 3 Posted, Due Tomorrow
Tools and Interfaces

Why Interface Tools?
Case Study of Model-View-Controller
Case Study of Animation
Sapir-Whorf Hypothesis
Thoughtfulness in Tools
Case Study in Self-Tracking
Sequential Programs

Program takes control, prompts for input

Person waits on the program

Program says when it is ready for more input, which the person then provides
Sequential Programs

while true {
    print "Prompt for Input"
    input = read_line_of_text()
    output = do_work()
    print output
}

Person is literally modeled as a file
Event-Driven Programming

A program waits for a person to provide input

All communication done via events
“mouse down”, “item drag”, “key up”

All events go to a queue
Ensures events handled in order
Hides specifics from applications

How many of these queues? How can you tell?
Basic Interactive Software Loop

do {
    e = read_event();
    dispatch_event(e);
    if (damage_exists())
        update_display();
} while (e.type != WM_QUIT);

All interactive software has this somewhere
Basic Interactive Software Loop

Have you ever written this loop?
Basic Interactive Software Loop

Have you ever written this loop?

Contrast with:

“One of the most complex aspects of Xlib programming is designing the event loop, which must take into account all of the possible events that can occur in a window.”

Understanding Tools

We use tools because they

- Identify common or important practices
- Package those practices in a framework
- Make it easy to follow those practices
- Make it easier to focus on our application

What are the benefits of this?
Understanding Tools

We use tools because they
- Identify common or important practices
- Package those practices in a framework
- Make it easy to follow those practices
- Make it easier to focus on our application

What are the benefits of this?
- Being faster allows more iterative design
- Implementation is generally better in the tool
- Consistency across applications using same tool
Understanding Tools

Why is designing tools difficult?

Need to understand the core practices and problems
Those are often evolving with technology and design

Example: Responsiveness in event-driven interface
Event-driven interaction is asynchronous
How to maintain responsiveness in the interface while executing some large computation?
Understanding Tools

Why is designing tools difficult?

- Need to understand the core practices and problems
- Those are often evolving with technology and design

Example: Responsiveness in event-driven interface

Cursor:
- WaitCursor vs. CWaitCursor vs. In Framework

Progress Bar:
- Data Races vs. Idle vs. Loop vs. Worker Objects
Fundamental Tools Terminology

Threshold vs. Ceiling

Threshold: How hard to get started
Ceiling: How much can be achieved
These depend on what is being implemented

Path of Least Resistance

Tools influence what interfaces are created

Moving Targets

Changing needs make tools incomplete or obsolete

Myers et al, 2000
http://dx.doi.org/10.1145/344949.344959
Tools and Interfaces

Why Interface Tools?
Case Study of Model-View-Controller
Case Study of Animation
Sapir-Whorf Hypothesis
Thoughtfulness in Tools
Case Study in Self-Tracking
Model-View-Controller

How to organize the code of an interface?

This is a surprisingly complicated question, with unstated assumptions requiring significant background to understand and resolve.
Seeheim Model

Results from 1985 workshop on user interface management systems, driven by goals of portability and modifiability, based in separating the interface from application functionality

Buxton, 1983
http://dx.doi.org/10.1145/988584.988586
Seeheim Model

Lexical - Presentation

External presentation of interface
Generates the display, receive input

Syntactic - Dialog Control

Parsing of tokens into syntax
Maintain state

Semantic - Application Interface Model

Defines interaction between interface and rest of software

e.g., “add” vs. “append” vs. “^a”
e.g., how to make a “menu” or “button”
e.g., three-state model, interface modes

e.g., drag-and-drop target highlighting
Seeheim Model
Seeheim Model

Diagram of the Seeheim Model:

- **Lexical**
  - Presentation
- **Syntactic**
  - Dialogue Control
- **Semantic**
  - Application Interface Model

User flow:
- USER → Presentation → Dialogue Control → Application Interface Model → APPLICATION

Note: The diagram includes a red-circled section labeled "Huh?"
Seeheim Model

Rapid Semantic Feedback

In practice, all of the code goes in here
Model-View-Controller

Introduced by Smalltalk developers at PARC
Partitions application to be scalable, maintainable
View / Controller Relationship

In theory:

Pattern of behavior in response to input events (i.e., concerns of the controller) are independent of visual geometry (i.e., concerns of the view)

Controller contacts view to interpret what input events mean in context of a view (e.g., selection)
View / Controller Relationship

In practice:

View and controller often tightly intertwined, almost always occur in matched pairs

Many architectures combine into a single class
Model-View-Controller

MVC separates concerns and scales better than global variables or putting everything together

Separation eases maintenance

  Can add new fields to model, new views can leverage, old views will still work

  Can replace model without changing views

Separation of “business logic” can require care

  May help to think of model as the client model
Model-View-Collection on the Web

Core ideas manifest differently according to needs

For example, backbone.js implements client views of models, with REST API calls to web server

Web tools often implement views as templates
Model View View-Model

Design to support data-binding by minimizing functionality in view

Also allows greater separation of expertise
Tools and Interfaces

Why Interface Tools?
Case Study of Model-View-Controller
Case Study of Animation
Sapir-Whorf Hypothesis
Thoughtfulness in Tools
Case Study in Self-Tracking
Luxor Jr.
Animation Case Study

Principles of Traditional Animation Applied to 3D Computer Animation

Lasseter, 1987

http://dx.doi.org/10.1145/37402.37407
Squash and Stretch
Squash and Stretch
Squash and Stretch

FIGURE 4a. In slow action, an object's position overlaps from frame to frame which gives the action a smooth appearance to the eye.

FIGURE 4b. Strobing occurs in a faster action when the object's positions do not overlap and the eye perceives separate images.

FIGURE 4c. Stretching the object so that its positions overlap again will relieve the strobing effect.
Timing

Just two drawings of a head, the first showing it leaning toward the right shoulder and the second with it over on the left and its chin slightly raised, can be made to communicate a multitude of ideas, depending entirely on the Timing used. Each inbetween drawing added between these two "extremes" gives a new meaning to the action.

NO inbetweens.......... The Character has been hit by a tremendous force. His head is nearly snapped off.

ONE inbetweens.......... The Character has been hit by a brick, rolling pin, frying pan.

TWO inbetweens........ The Character has a nervous tic, a muscle spasm, an uncontrollable twitch.

THREE inbetweens..... The Character is dodging a brick, rolling pin, frying pan.
Timing

FOUR inbetweens.......... The Character is giving a crisp order, "Get going!" "Move it!"

FIVE inbetweens.......... The Character is more friendly, "Over here." "Come on-hurry!"

SIX inbetweens.......... The Character sees a good looking girl, or the sports car he has always wanted.

SEVEN inbetweens.......... The Character tries to get a better look at something.
Timing

EIGHT inbetweens.......... The Character searches for the peanut butter on the kitchen shelf.

NINE inbetweens.......... The Character appraises, considering thoughtfully.

TEN inbetweens.......... The Character stretches a sore muscle.
Anticipation
FIGURE 6. Andre's scratch was staged to the side (in "silhouette") for clarity and because that is where his itch was.
Staging

FIGURES 7-8. In Luxo Jr., all action was staged to the side for clarity.
Follow Through, Overlap, Secondary
Objects with mass must accelerate and decelerate. Interesting frames are typically at ends, tweaks perception to emphasize these poses.
Luxor Jr.
Animation Case Study

Animation: From Cartoons to the User Interface

Chang and Ungar, 1993

http://dx.doi.org/10.1145/168642.168647
Frames Three Principles

Solidity
Desktop objects should appear to be solid objects

Exaggeration
Exaggerate physical actions to enhance perception

Reinforcement
Use effects to drive home feeling of reality
Solidity: Motion Blur
Solidity: Arrival and Departure
Solidity: Arrival and Departure
Exaggeration: Anticipation

Figure 7. Objects anticipate major actions with a quick contrary motion that draws the user eye to the object in preparation for the main motion to come.
Reinforcement: Slow In Slow Out

Figure 8. Objects ease out of their beginning poses and ease into their final poses. Although these motions are slower than that during the main portion of the movement, they are still quite fast.
Reinforcement: Arcs

Figure 9. When objects travel under their own power (non-interactively), they move in arcs rather than straight lines.
Reinforcement: Follow Through

Figure 10. When objects come to a stop after moving on their own, they exhibit follow through in the form of wiggling back and forth quickly. This is just suggested by the "wiggle lines" in the figure—in actuality, the object moves back and forth, with motion blur.
Animation Support in a User Interface Toolkit: Flexible, Robust, and Reusable Abstractions

Scott E. Hudson
John F. Stasko

ABSTRACT

Animation can be a very effective mechanism to convey information in visualizations and user interface settings. However, integrating animated presentations into user interfaces has typically been a difficult task since, to date, there has been little or no explicit support for animation in window systems or user interface toolkits. This paper describes how the ArtKit user interface toolkit has been extended with new animation support abstractions designed to overcome this problem. These abstractions provide a powerful but conventional base for building a range of animations, supporting techniques such as simple motion-blur, "squash and stretch", use of aliasing to reduce flicker, and "slow-in / slow-out" transitions. Because these abstractions are provided by the toolkit they are reusable and may be freely mixed with more conventional user interface techniques. In addition, the ArtKit implementation of these abstractions is robust in the face of systems such as the X Window System and Unix which can be ill-suited with respect to timing considerations.

Keywords: object-oriented user interface toolkits, window systems, animation techniques, dynamic interfaces, motion blur, real-time scheduling.

1 INTRODUCTION

Human perceptual capabilities provide a substantial ability to form and understand models of the world from moving images. As a result, in a well-designed display, information can be much more easily comprehended in a moving scene than in a single static image or even a sequence of static images. For example, the "craze" display described in [Rob92] provides a clear illustration that the use of continuous motion can allow much more information to be presented and understood more easily.

However, even though the potential benefits of animation in user interfaces have been recognized for some time ([Blau88] for example, surveys a number of uses for animation in the interface and cites their benefits and [Stas93] reviews principles for using animation in interfaces and describes a number of systems that make extensive use of animation in its interface), explicit support for animation is rarely, if ever, found in user interface support environments. The work described in this paper is designed to overcome this problem by showing how flexible, robust, and reusable support for animation can be incorporated into a full scale object-oriented user interface toolkit. Specifically, this paper describes how the extension mechanisms of ArtKit — the Advanced Reusable Toolkit (supporting interfaces in C++) [Hend90] — have been employed to smoothly integrate animation support with other user interface capabilities.

The animation abstractions provided by the ArtKit system are designed to be powerful and flexible — providing basic support that can be built on to generate a wide range of sophisticated techniques such as: simple motion-blur, "squash and stretch", use of aliasing...
Events and Animation

Figure 5. Animation Event Translation and Dispatch
Not Just an Implementation

Provides tool abstractions for implementing previously presented styles of animation

Overcomes a fundamental clash of approaches

Event loop receives input, processes, repaints

Animations expect careful control of frames, but the event loop has variable timing
Events and Animation

Figure 5. Animation Event Translation and Dispatch
Transition Object

Figure 3. Parts of a Transition Object
Pacing Function

Figure 4. Two Example Pacing Functions
Computing a Frame

Figure 8. Translation from Time to Space
Animation Case Study

Based on increased understanding of how animation should be done in the interface, increasingly mature tools develop

Now built into major commercial toolkits (e.g., Microsoft’s WPF, JavaFX, jQuery)

Once mature, begins to be used as a building block in even more complex behaviors
The Kinetic Typography Engine: An Extensible System for Animating Expressive Text

Lee et al, 2002
http://dx.doi.org/10.1145/571985.571997
Kinetic Typography Engine

Kinetic Typography

Johnny Lee, Jodi Forlizzi, Scott Hudson
Carnegie Mellon University
Human-Computer Interaction Institute
2002
Kinetic Typography Engine

Goals of Kinetic Type

- Emotional content
- Creation of characters
- Direction of attention

Animation Composition

Figure 6. Waveform addition by chaining

Figure 7. Waveform scaling by functional composition with amplitude
Tools and Interfaces

Why Interface Tools?
Case Study of Model-View-Controller
Case Study of Animation
Sapir-Whorf Hypothesis
Thoughtfulness in Tools
Case Study in Self-Tracking
Sapir-Whorf Hypothesis

Roughly, some thoughts in one language cannot be stated or understood in another language

Language is not simply a way of voicing ideas, but is the very thing which shapes those ideas

Our tools define the language of interaction

Beyond the simple matter of code
Frame how we think about possibilities

You must be aware of this when choosing tools, designing applications, and creating new tools
Animation Case Study

Phosphor: Explaining Transitions in the User Interface Using Afterglow Effects

Baudisch et al, 2006

http://dx.doi.org/10.1145/1166253.1166280
Phosphor

Animation can help people follow interface transitions

But the right speed is crucial
  Too fast increases error rate
  Too slow increases task time

The right speed depends on familiarity, distraction, etc.
  It cannot be determined
Phosphor

Phosphor shows the outcome immediately, then explains change in retrospect using a diagrammatic depiction.
Phosphor

phosphor
Challenging Assumptions of Tools

Phosphor breaks from the assumptions that have evolved into current transition tools.
Tools and Interfaces

Tools embody expertise and assumptions

Tools evolve based on emerging understanding of how to address categories of problems

Be conscious of your tool decisions

- Try to think about designs before tying to a tool
- Choose good and appropriate tools
- Understand what you are getting in a tool
- Push yourself to think outside the tool
Prefab

Prefab uses pixel analysis to modify existing applications from the outside, using only pixels.

Prefab is informed by how toolkits work, but not linked to any particular toolkit implementation.

Allows trying and fielding new ideas that are not supported by existing applications or toolkits.
Prefab
Mobile Phones as Pagers

Our notion of technology design for journals / ESM / EMA has been anchored by papers journals and pager-based reminders

Unlock Journaling for Self-Report


Zhang, Pina, Fogarty. Examining Unlock Journaling with Diaries and Reminders ... CHI 2016.
Unlock Journaling for Self-Report

Unlock Journaling for Self-Report

Pleasure and Accomplishment (e.g., self-monitoring depressive symptoms)

Lejuez, Hopko, Acierno, Daughters, Pagoto. ... Behavioral Activation Treatment for Depression ... Behav Modif 2011.
Unlock Journaling for Self-Report

Russell’s Affect Grid

Unlock Journaling vs. Notifications

Unlock journaling is:

- rated less intrusive
  (1.77 vs. 2.22 on a 5-point scale)
- yields greater frequency
  (15.0 vs. 9.8 per 12-hour day)
- comparable timeliness
  (8.6 vs. 9.3 minutes)

Instead of reminders to journal, unlock journaling makes the opportunity visible, easy, and optional.

It should not have taken 10 years to get here.
Mobile Food Journals

Origins in daily recall

Self-monitoring of food can support many goals

Weight Loss
Diabetes Management
Trigger Identification

High burdens detract from potential benefit, data is often wrong

Mobile Food Journals

Mobile devices provide real-time feedback

Search for each food in a large database, often breaking into components

Typically provide calorie-based feedback

High burdens detract from potential benefit, data is often wrong

Perceptions of Healthy Eating

“What does healthy eating look like to you?”

Food types:
“vegetables”
“fruits”
“protein”

Food qualities:
“low processed”
“organic”
“fresh”

Diet qualities:
“balanced”
“variety”
“portion”

Cordeiro, Bales, Cherry, Fogarty. Rethinking the Mobile Food Journal … CHI 2015.
Difficulty as a Negative Nudge

“I just avoided eating things that were hard to log” – SP132

“Prepackaged meals were the easiest because of bar codes but those aren’t healthy” – SP123

“I could make life easier by eating the same things regularly” – SP97

“It discourages you from eating out or at a friend’s, even if it is healthy” – SP42

Cordeiro, Epstein, Thomaz, Bales, Jagannathan, Abowd, Fogarty. Barriers and Negative Nudges ... CHI 2015.
Cordeiro, Bales, Cherry, Fogarty. Rethinking the Mobile Food Journal ... CHI 2015.
Deploying a Photo-Based Journal

Mobile capture and review

Web review and annotation

Cordeiro, Bales, Cherry, Fogarty. Rethinking the Mobile Food Journal … CHI 2015.
Leveling the Difficulty of Journaling

With prior techniques:
60% report not journaling because it was too difficult
65% report not journaling because they did not know

With photo-based capture:
22% report not journaling because it was too difficult
None report not journaling due to food knowledge

“For some meals, it’s just really easy to take a picture … than sit there and type in every ingredient” – FP20
Journaling without Judgment

With prior journals, participants report choosing not to journal because they would exceed a calorie budget or because a food was unhealthy.

- 13% of survey participants
- 45% of field participants

Photos enable mindfulness while avoiding judgment.

“[it was] easier because there were no calorie counts, no judgments, but still makes you aware” – FP14

“Do I really want to eat this? I’m capturing this” – FP17
Triggers and Trends

“I eat too much pizza” – FP10

“I’m surprised at how many times I’m seeing things that I consider an exception to my diet!” – FP4

“I don’t branch out as much as I thought I did, even when I go somewhere new, I kind of get what I always get somewhere else” – FP10
Food Journals as Daily Recall

“it should be noted that much of the use of food journaling is in a more clinical setting with the purpose being sharing and evaluating the journal with nutritionists and care providers …

it’s not relevant if photos are more or less easily understood by the user if a nutritionist is the eventual consumer of the data”

– Actual Anonymous Grumpy R3