Multi-Device Interaction

Pierre Dragicevic
(Pierre.Dragicevic@emn.fr)

Jean-Daniel Fekete
(Jean-Daniel.Fekete@emn.fr)

Ecole des Mines of Nantes
Generations of User Interfaces

Graphical User Interfaces

- Bitmapped graphics
- Graph. Interaction objects

Multimedia Interfaces

- Mixture of static
- and dynamic media

Future I. (multimodal,
multimedia, virtual)

As Input:
- Sound and picture

As Commands:
- Voice
- Gesture
- Eye-movement
- Biosignals
- Manipulation of physical objects

Simulated environments
- Augmented reality
- Multimedia VR

Adapted from Jürgen Ziegler
Input Device Integration

• **There is no easy way to integrate**
  - New devices
  - New interaction styles

• **...Into existing**
  - Applications
  - Toolkits
  - Systems
Why Is Easy Device Integration So Important?

• **User’s abilities**
  • Common skills: *writing,*...
  • Special skills: *cartoonist, puppet animator,*...

• **User’s constraints**
  • Disabilities: *hearing, motor,*...

• **Working environment opportunities**
  • Input rich: *multimedia, multimodal,*...

• **Working environment constraints**
  • Input limited: *noisy environment, small footprint, one-hand control,*...
What We Currently Need

• A good description of the application’s demands

• A good description of what an input device is able to provide

• An efficient tool that helps to connect them together
Our Work

- **Building an “input management toolkit” for multi-device interaction**
  - Designing interaction techniques that rely on this toolkit
  - Integrating it in several systems to show its effectiveness
Choosing Interaction Techniques

- Position \((x,y)\)
- Text \((\text{string})\)
- Stroke \((x,y)^n\)
- Motion control
- Scanning control
- Screen Keyboard
- Speech recognition
- Handwritten recognition

Input Devices

Text Editor

Insertion Point

Annotation
Choosing Interaction Techniques

Position \((x,y)\)

Text \((\text{string})\)

Stroke \((x,y)^n\)

Input Devices

Motion control

Scanning control

Text Editor

Insertion Point

Annotation

Handwritten recognition
Matching Pre- and Post-Conditions

- **Xjoystick**
  - [0, 255] → 127

- **Xdigitizer**
  - Persistent

- **Xmouse**
  - Persistent

- **Xpointer**
  - [0, 639] persistent

- **Button1**
  - [0, 1] → 0

- **Button2**
  - [0, 1] → 0

- **Pressure**
  - [0, 255] → 0

- **Button**
  - [0, 1] → 0

- **Equations**
  - \( k \) = \( \frac{-1}{1} \)
Building Blocks for Input Constructions

Input Device

Virtual Device / Operator

Interaction Task

Output Slot

Name
Value
Post-conditions
(Domain Measure Precision Zero value ...)

Input Slot

Name
Value
Pre-conditions
Building Blocks are Reactive Programs

- Event-driven
- Deterministic
- Executable
- Verifiable (lots of support tools)
Esterel / SugarCubes Implementation

- **Esterel**
  - Has the right semantics for describing concurrent reactive processes
  - Can be used to verify temporal and reachability properties
  - Can be compiled and executed

- **SugarCubes**
  - Provides a set of Java Classes for implementing threadless concurrent reactive systems
  - Has a semantics close to Esterel
  - Allows dynamic description of reactive modules
  - Is Interpreted
Performed Work

• Study of a device model based on physical and transducer levels, with data/properties processing operators [Dragicevic 98]

• Study of reactive scripting of input behavior by using Esterel to implement interactors [Fekete, Richard, Dragicevic 98]

Ongoing Work

• Creation of input constructions with the SugarCubes toolkit
Future Work

- **Write a first prototype of the “Input Management Toolkit”** (January ’99)
  - Find a good device/interaction task description
  - Define & implement connection mechanisms
  - Provide a set of processing operators and customizable interaction techniques
- **Integrate it in several systems**
  - Multimodal CAD system
  - Musical score editor
  - Larger size applications