Instrumental Interaction in Multisurface Environments

Michel Beaudouin-Lafon
Université Paris-Sud & Institut Universitaire de France

LMU München - 5 February 2014
“The best way to predict the future is to invent it”
(attributed to Alan Kay)

NLS/Augment
1967

Xerox Alto
1973
What happened to the future promised by Ubicomp?

• “The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it.” - Mark Weiser
What happened to the future promised by Augmented Reality?

• “From the isolation of our workstations we try to interact with our surrounding environment, but the two worlds have little in common. How can we escape from the computer screen and bring these two worlds together?” - Wellner, Mackay & Gold
What happened to the future promised by Tangible Interfaces?

• “The term Graspable UI refers to both the ability to physically grasp an object (i.e., placing a hand on an object) as well as conceptual grasping (i.e., to take hold of intellectually or to comprehend)” - George Fitzmaurice
The missing link

• Between the interaction paradigm, illustrated by some prototypes, and the principled design of effective interfaces based on the paradigm.
We need **new Interaction Models**
and associated tools to reinvent user interfaces based on these paradigms
What is an Interaction Model?

• A set of rules and guidelines to help create consistent interactive systems according to a certain style
  
  • **Descriptive**: define the scope of the design space
  
  • **Prescriptive**: provide criteria to compare designs
  
  • **Generative**: support creativity and inspire new designs
Examples of Interaction Models

• CLI: dialogue - language - syntax

• GUI: direct manipulation - desktop metaphor

• Augmented Reality / Mixed Reality / Virtual Reality / Tangible interface / Ubiquitous Computing / Natural User Interfaces / Reality-Based Interfaces / ...
Interaction Model

- Conceptual model
- Interaction techniques
- Hardware
- UI toolkit
- Application
Interaction Model

- Conceptual model
- Interaction techniques
- Application
- UI toolkit
- Hardware
Tools and Instruments

L'encyclopédie - Diderot & d'Alembert, 1751-1772
The power of tools

• Gibson’s Ecological Theory:
  • Affordances = possibilities for action in the environment relative to the capabilities of the subject
  • Tools redefine the affordances of the environment because they change the capabilities of the subject
Gibson on tools

• “When in use, a tool is a sort of extension of the hand, almost an attachment to it or a part of the user's own body, and thus is no longer a part of the environment of the user. […]

This capacity to attach something to the body […] suggests that the absolute duality of "objective" and "subjective" is false.”

(Gibson, emphasis by the author)

• Affordances of objects that afford manipulation (i.e., tools): a stick affords trace-making in the sand.
The power of tools

• The user of a tool internalizes the tool as an extension of one’s body

• A stick extends the reach of the arm
The power of tools

• Holding a pen raises awareness for the affordance for writability
Instrumental Interaction

- Mediated interaction: user - instrument - object of interest
- An instrument **reifies** a command
- Use the same instrument with different objects (polymorphism)
Instrumental Interaction: descriptive

- Covers many interaction styles:
  - Traditional GUI
  - Novel techniques
  - Tangible interaction
Instrumental Interaction: prescriptive

- Provides metrics to compare instruments, for example:
  - Degree of indirection
  - Degree of integration

2 => 1
2 => 3

Wednesday, February 5, 14
Instrumental Interaction: generative

- 3 design principles:
  - **Reification**: extends the notion of what constitutes an object
  - **Polymorphism**: extends the power of instruments w.r.t. objects
  - **Reuse**: provides a way of capturing and reusing interaction patterns
Proof-of-concept: CPN2000

• Bi-manual interaction, Marking menus, Toolglasses

• Combines power and simplicity

• 40,000+ downloads

Lassen & Beaudouin-Lafon, UIST '00

Wednesday, February 5, 14
Reification

• Turns concepts into objects
• In particular, turns commands into instruments
• Interaction instrument
  • Example: scrolling a document => scrollbar
  • Reification of a command into an interface widget
Example: aligning objects

- Align command: align now and forget it

vs.

- Align instrument: align and keep aligned
Example: Graspables

Fitzmaurice, 1995
Polymorphism

- Extends commands to multiple object types
  - Common examples: Cut, paste, delete, move
- Instruments can be applied to many different objects
- Groups take advantage of polymorphism:
  Applying a command to a group applies it to each object
Example: Slap Widgets

Weiss, Wagner, Jansen & Borchers, 2009
Reuse

• Captures interaction patterns for later reuse

• Output reuse
  • Reuse previously created objects
  • Example: duplicate, copy/paste

• Input reuse
  • Reuse previous commands
  • Example: redo, history, macros
Example: Media Blocks

Ullmer, Glas & Ishii, 1998

• Limited form of output reuse: a block can change content

• Limited form of input reuse: replacing a block or changing its location
Combining the principles

- **Layers**: Reify modes
  - Control visual complexity

- **Styles**: Reify collections of attributes
  - Support polymorphism, encourage reuse

- **Groups**: Reify selection
  - Support polymorphism
Example: DataTiles

Rekimoto, Ullmer & Oba, 2001

• Some tiles represent content, others are instruments: reification

• Spatial combinations specify chains of computation: polymorphism of the tiles

• Changing a tile in the chain reuses the chain: reuse
Ubiquitous Instrumental Interaction

• Detaching instruments from the objects of interest ... and from applications

• Instruments spanning multiple interaction surfaces

• **Multisurface interaction**
Multisurface Interaction in the WILD room

Video supplement
IEEE Computer, April 2012

© in situ 2012
Exploring instruments for Multisurface Interaction
Participatory Design

• Create new ways to interact with complex data

• Transport objects with the “shovel”
Reminiscent of Rekimoto’s pick and drop

Rekimoto, 1997
Participatory Design

- Create new ways to interact with complex data
- Use a tablet as a magic lens
CHI 2013
200 sessions
400 papers
16 parallel sessions
over 4 days
no conflicts

Wednesday, February 5, 14
Reminiscent of Fitzmaurice’s Chameleon

Fitzmaurice, CACM’93
Participatory Design

• Create new ways to interact with complex data

• Use a prop to control online objects
Software: Substance Grise

- Display 64 3D brain scans with VISA/Anatomist
- Organize them on the table
- Control their orientation in real time through a prop
Reminiscent of Hinckley’s neurosurgical props

Hinckley et al, CHI 94
A meta-model for instrumental interaction
Information substrates

• Data does not exist in a vacuum

• Substrates provide context for interpreting data and constraints for presenting and interacting with it

• Examples: table, page-based layout, graph, musical score
Substrates

• Not just a view (in the sense of MVC)

• Representation and physical organization of data

• Affordances for certain operations: layout and spatial organization, data flow (a graph linked to a table), ...

• Can embed instruments, e.g. magnetic guidelines. Similar to scaffolding when creating a building
Paper substrates: Video Mosaic

- Storyboard elements printed on paper
- Laying out time in space to organize a sequence of clips
- Paper buttons

Mackay & Pagani, 1994
Paper substrates for music composition

Garcia, Tsandilas, Agon & Mackay, 2012
Tangible substrates: URP

Underkoffler & Ishii, 1999

• Tangible representation of buildings

• Real-time display of simulation data (wind, lighting, ...)

• Tangible tools to control the simulation
Tangible substrates: TinkerLamp

- Tangible representation of a warehouse to train students
- Also uses interactive paper to control the simulation
- Used for teaching at a vocational school in Switzerland
Interaction protocols

• Describe which instruments can operate on which objects

• Support exploration and appropriation (including breaking things)

• Explicit compatibility: object advertises its capabilities

• Implicit compatibility: instrument discovers objects’ properties
Example: color pickers
Example: color pickers

- Select a color:
  - From an external object
  - From a color space
  - From a color palette

- Explicit compatibility: SetColor/GetColor methods
- Implicit compatibility: a property of the object is a color
Benefits of instruments

• Decouple data/information from the tools used to view/edit it

• Provide a natural way to support user customization / appropriation

• Foster a different business model for software, based on components and interoperability
Next steps

• Refine the conceptual model
  • Information substrates
    + interaction protocols
    + instruments
  • Explore the use of instruments with objects they were not designed for

• Build a robust and scalable software infrastructure

• Test in various settings
Thank you!

Michel Beaudouin-Lafon • http://www.lri.fr/~mbl • mbl@lri.fr