Direct Manipulation

Arnav Gupta

April 19, 2024

Contents

1 Instrumental Interaction

Direct manipulation is when a virtual representation of an object is manipulated in a similar way to a real world object. Goal is to make the interaction feel like manipulating a real object instead of working through and intermediary.

Requires a representation of task objects (something the user can manipulate). Can be the object of interest or an interface widget.

Characteristics of Direct Manipulation:

1. continuous representation of task objects and actions
2. task objects manipulated by physical actions, not complex syntax
3. fast, incremental, and reversible actions with effects on task objects immediately apparent
4. layered, self-revealing approach to learning

Benefit of direct manipulation is the feel as if interacting wih the task object rather than the interface, so focus is on the task, not the technology.

Feeling of direct involvement with world of task objects rather than communication with an intermediary.

An **interaction model** is a set of principles, rules, and properties that guide the design of an interface. Describes how to combine interaction techniques in a meaningful and consistent way and defines the look and feel of the interaction from the user's perspective. Properties of the interaction model can be used to evaluate specific interaction designs.

Instrumental interaction is a model of interaction based on natural use of tools/instruments to manipulate objects in the physical world.

Interfaces have:

- **interaction instruments**: necessary mediators between user and domain objects
- **domain objects**: things of interest, data, and associated attributes, which are manipulated using an interaction instrument

Instrument activation is how an instrument is triggered for use. GUI instruments activated spatially (movement cost) and temporally (time cost). UI layout and design is concerned with the tradeoff of these costs.

Reification: turning concepts into something concrete

An instrument is the reificiation of a command.

Meta-instrument: an instrument that acts on another instrument, which becomes an object of interest

Object reficiation involves turning attributes of a primary object into other objects of interest.

3 properties of instruments:

- degree of **indirection**: spatial/temporal offset between instrument and action on object
- degree of **integration**: match between input device to instrument degrees of freedom
- degree of **compatibility**: similarity of action on device/instrument to action on object

Degree of indirection is the 2D measure of distance from instrument to object:

- spatial dimension can be near (drag to translate, handle to resize), medium (scrollbar near page), or far (dialog box)
- temporal dimension can be short (direct drag response), medium (activate tool in toolbar, start direct manipulation), or long (using dialog, full drag-and-drop operation)

Degree of integration is the ratio of the degrees of freedom (DOF) of the instrument over the DOF captured by the input device, which captures the suitability of the device.

Degree of compatibility is the similarity of physical actions on instrument and response of the object (more similar is better since it's more natural or intuitive).

A direct manipulation interface allows a user to directly act on a set of objects in the interface. (low indirection, high integration, high compatibility)

Ideal level of direct manipulation is when instruments are visually indistinguishable from object they control:

- actions on instrument/object entities are analogous to actions on similar object in the real world
- action on instrument/object entities preserve the conceptual linkage between instrument and object

2 Dragging

Mousedown on shape starts drag (find offset from position to the shape frame-of-reference), mousemove to drag, and mouseup to end drag.

3 Drag-and-Drop

Most UI tooklits have built-in support for drag-and-drop.

A drag-and-drop interaction is:

1. press button (mousedown) with cursor on source object/data
2. drag source object onto target object
3. release button (mouseup) to drop object/data on target

HTML drag-and-drop API enables drag-and-drop interactions in browser applications:

- make element draggable with `draggable` attribute
- manage a drag operation by handling drag-and-drop events: dragstart, drag, dragenter, dragleave, dragover, drop, dragend
- define what info is manipulated

Direct manipulation feedback is important:

- visual indication of elements that can be dragged
- feedback showing possible drop targets