Interactive Techniques

Topics

Events
Picking and tracking
Widgets
Event propagation
Model-view-controller
Measure and Trigger

Polling
- Periodically read the state of the devices
- Mouse microcontroller sends messages @100Hz
  - State of the buttons
  - Relative motion of the mouse

Events
- Keyboard sends “make” and “break” messages

Event Queue

Operating system places events in an queue
- Time-stamp
- Queues are first-in first-out
- May store input device state at time of event

Callback manager
- Event priority
- Event coalescing

Demo of input.py
Demo of spin.c
Dispatching Events

Associate events with objects

Two methods
- Mouse over object
- Keyboard focus

Picking

Two issues
- Priority
- Geometry

Hit testing
- Bounding box
- Hit method
- Pick support in OpenGL
- Object tags
Tracking

Tracking: Convert in/out to enter/leave

*Flash button and menubutton demo*
HTML Form Elements

Propagating Events

Level 0 DOM

```javascript
widget.onclick = clickHandler;
widget.onchange = changeHandler;
widget.onfocus = focusHandler;
widget.onblur = blurHandler;
widget.onmousemove = moveHandler;
...
```

See p. 354 Definitive Guide to Javascript

Propagating Events

Level 2 DOM

```javascript
widget.addEventListener("click",handler,capture);
widget.removeEventListener("click",handler,capture);
function handler(e) {
  if (!e) var e = window.event;
  var targ;
  if (e.target) targ = e.target;
  else if (e.srcElement) targ = e.srcElement;
}
```

More than one handler per object

Event object contains information about the event

Phases (w3c):
- Capture phase: from root to leaf element (netscape)
- Target phase: leaf element processes event
- Bubble phase: from leaf element to root (microsoft)
Ivan Sutherland’s Sketchpad

Model-View-Controller Design Pattern

m = new Model();
m.addViewListener(v1);
m.addViewListener(v2);
...

v1 = new ViewA(model);
v2 = new ViewA(model);
v3 = new ViewB(model);
...
Model-View-Controller Design Pattern

\[
m\text{onUpdate} = \text{function}() \\
\{
    \text{for } v \text{ in } m\text{.views} \\
    \quad v\text{.onUpdate}();
\}
\]
\[
c\text{setName} = \text{function}(s) \{ \\
    c\text{.m\text{.setName}}(s); \\
    c\text{.m\text{.onUpdate}}();
\}
\]
\[
v\text{onChange}() = \text{function}() \{ \\
    v\text{.c\text{.setName}}(v\text{.value});
\}\]