

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

CS148 Lecture 8

Pat Hanrahan, Winter 2007

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

CS148 Lecture 8

Pat Hanrahan, Winter 2007

Dispatching Events

Associate events with objects

Two methods

- Mouse over object
- Keyboard focus

CS148 Lecture 8

Pat Hanrahan, Winter 2007

Picking

Two issues

- Priority
- Geometry

Hit testing

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

CS148 Lecture 8

Pat Hanrahan, Winter 2007

Tracking

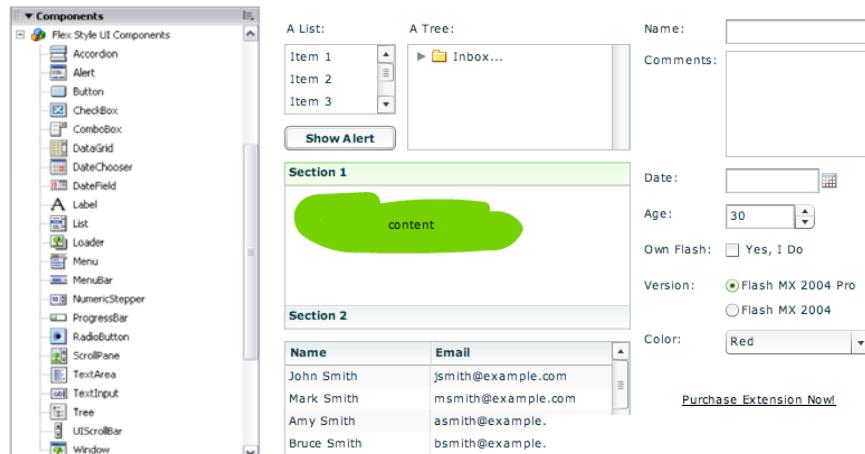
Tracking: Convert in/out to enter/leave

Flash button and menubutton demo

CS148 Lecture 8

Pat Hanrahan, Winter 2007

Flash Components



CS148 Lecture 8

Pat Hanrahan, Winter 2007

HTML Form Elements

The screenshot shows a Mozilla Firefox window displaying a form with several input fields and a legend at the bottom.

- Username:** [1]David (Text input)
- Password:** [2]*** (Password input)
- Filename:** [4]everything.html (File input with Browse... button)
- My Computer Peripherals:** [5] DVD Writer
[5] Printer
[5] Card Reader (List of checkboxes)
- My Web Browser:** [6] Firefox
[6] Internet Explorer
[6] Other (List of radio buttons)
- My Hobbies:** [7] Hacking JavaScript
Surfing the Web
Drinking Coffee
Annoying my Friends (List of checkboxes)
- My Favorite Color:** [8] Blue (Select menu with a dropdown arrow)
- Input Events:** [3]
Focus: username ()
Click: username ()
Change: username (Devid)
Blur: username (Devid)
Focus: password ()
Select: password ()
Change: password (david)
Blur: password (david)
Focus: textarea (...)
Change: textarea (...)
Blur: textarea (...)
Click: file ()
Change: file (everything.html)
Focus: extras (printer)
Click: extras (printer)
Change: extras (printer)
Blur: extras (burner)
Focus: extras (burner)
Click: extras (burner)
Change: extras (burner) (List of event handlers)
- Buttons:**
 - [9] Clear
 - [10] Submit
 - [11] Reset
- Form Elements Legend:**
 - [1] Text
 - [2] Password
 - [3] Textarea
 - [4] FileUpload
 - [5] Checkbox
 - [6] Radio
 - [7] Select (list)
 - [8] Select (menu)
 - [9] Button
 - [10] Submit
 - [11] Reset

<http://examples.oreilly.com/jscript2/17.1.html>

CS148 Lecture 8

Pat Hanrahan, Winter 2007

HTML Form Elements

Button	<code><input type="button"></code> or <code><button type="button"></code>	"button"	A push button; <code>onclick</code> .
Checkbox	<code><input type="checkbox"></code>	"checkbox"	A toggle button without radio-button behavior; <code>onclick</code> .
File	<code><input type="file"></code>	"file"	An input field for entering the name of a file to upload to the web server; <code>onchange</code> .
Hidden	<code><input type="hidden"></code>	"hidden"	Data submitted with the form but not visible to the user; no event handlers.
Option	<code><option></code>	none	A single item within a Select object; event handlers are on the Select object, not on individual Option objects.
Password	<code><input type="password"></code>	"password"	An input field for password entry—typed characters are not visible; <code>onchange</code> .
Radio	<code><input type="radio"></code>	"radio"	A toggle button with radio-button behavior—only one selected at a time; <code>onclick</code> .
Reset	<code><input type="reset"></code> or <code><button type="reset"></code>	"reset"	A push button that resets a form; <code>onclick</code> .
Select	<code><select></code>	"select-one"	A list or drop-down menu from which one item may be selected; <code>onchange</code> . (See also Option object.)
Select	<code><select multiple></code>	"select-multiple"	A list from which multiple items may be selected; <code>onchange</code> . (See also Option object.)
Submit	<code><input type="submit"></code> or <code><button type="submit"></code>	"submit"	A push button that submits a form; <code>onclick</code> .
Text	<code><input type="text"></code>	"text"	A single-line text entry field; <code>onchange</code> .
Textarea	<code><textarea></code>	"textarea"	A multiline text entry field; <code>onchange</code> .

CS148 Lecture 8

Pat Hanrahan, Winter 2007

Propagating Events

Level 0 DOM

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

See p. 354 Definitive Guide to Javascript

CS148 Lecture 8

Pat Hanrahan, Winter 2007

Propagating Events

Level 2 DOM

```
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)

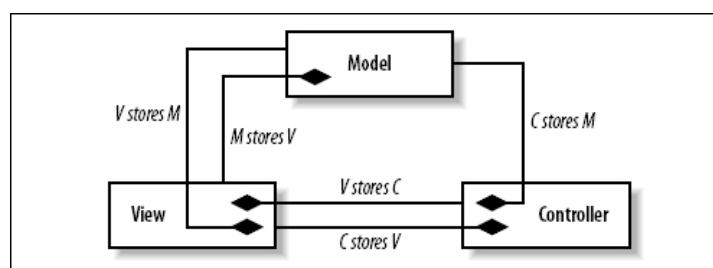
CS148 Lecture 8

Pat Hanrahan, Winter 2007

Ivan Sutherland's Sketchpad

Model-View-Controller Design Pattern

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

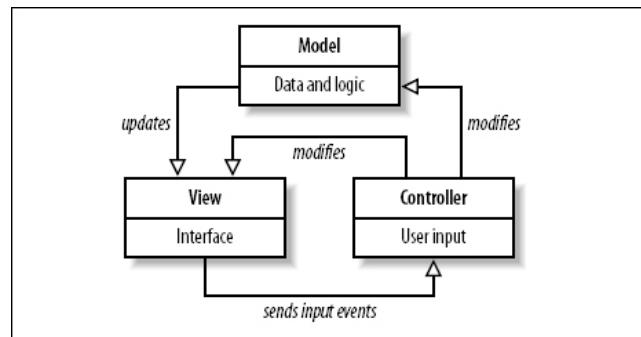


```
v1 = new ViewA(model);
v2 = new ViewA(model);
v3 = new ViewB(model);
...
c = new Controller(model);
```

Model-View-Controller Design Pattern

```
m.onUpdate = function()
{
    for v in m.views
        v.onUpdate();
}
```

```
c.setName = function(s) {
    c.m.setName(s);
    c.m.onUpdate();
}
```



```
v.onChange() = function() {
    v.c.setName(v.value);
}
```

CS148 Lecture 8

Pat Hanrahan, Winter 2007