

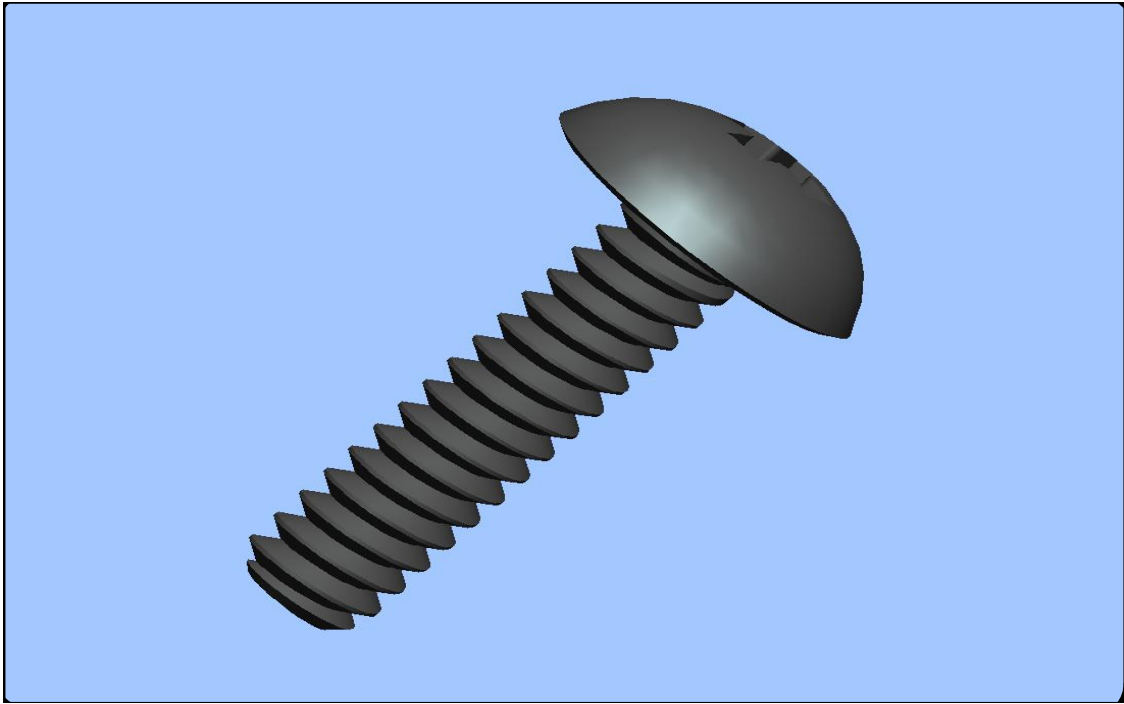
# Software Tools

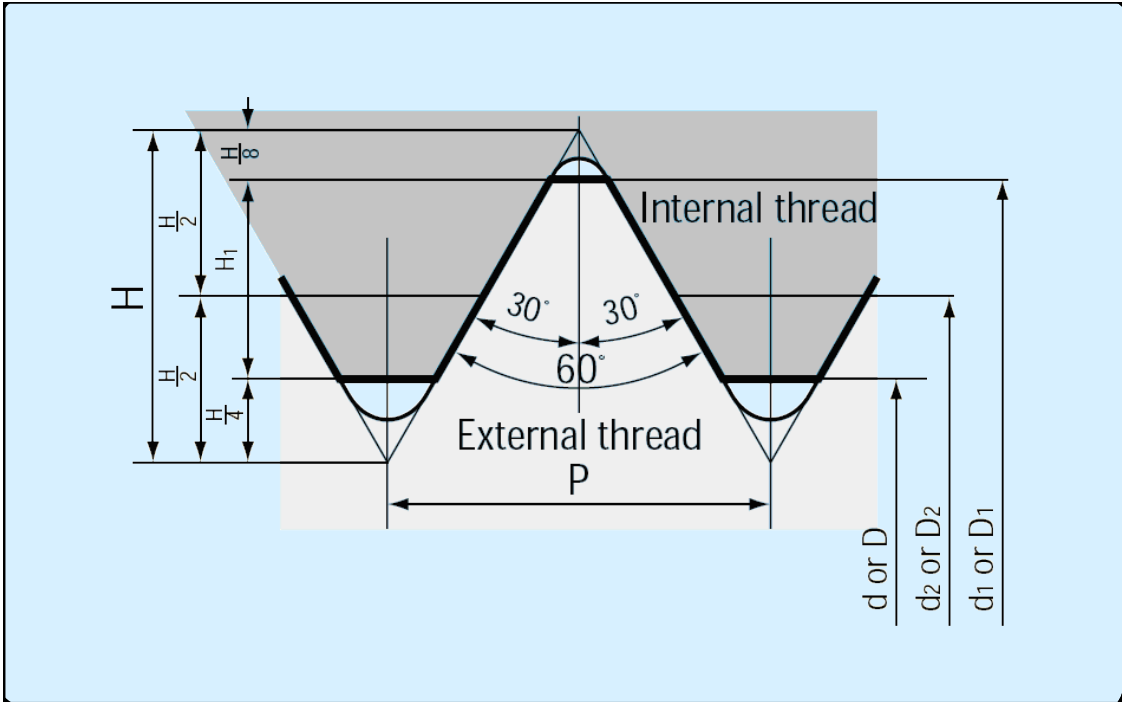
**Ben Bederson**  
CMSC434 – University of Maryland

These slides adapted from [Scott Klemmer](#)  
and [Nelson Padua-Perez](#)

## Questions?

- HCIL [Brown Bag Lunch](#)
  - Login: hcil/hcil
  - Thursdays, 12:30–1:30,
  - HCIL Lab (2015 Hornbake, South Wing)
- HCIL [Annual Symposium](#)
  - May 25–26, 2011
- HOF/S Presentation





**Standards ↔ Innovation**

## What Toolkits Are (a.k.a. Libraries, Frameworks)

Menu  
Scrollbar  
Textbox  
Button  
Canvas  
...

Storage  
Network  
File formats  
Graphics  
...

Compilers  
Run-times  
Communication  
...

## Toolkit Trade-offs

### Positive:

- Code re-use / fewer bugs
- Standardization
- Speed => Rapid prototyping
- Accessibility
- Internationalization
- Design tools => non-programmers
- Lower maintenance costs

### Negative:

- Harder to do non-standard things
- Can discourage change
- NUIs often unsupported
- Generality => higher resource usage

# Predictability

for users

# Path of Least Resistance

for developers

**Model-View-  
Controller**

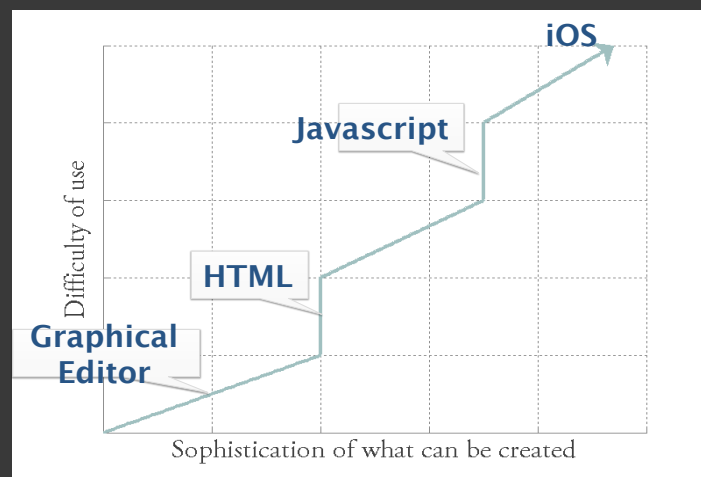
**Model-View-  
Controller**

**Model-View-  
Controller**

**Model-View-  
Controller**

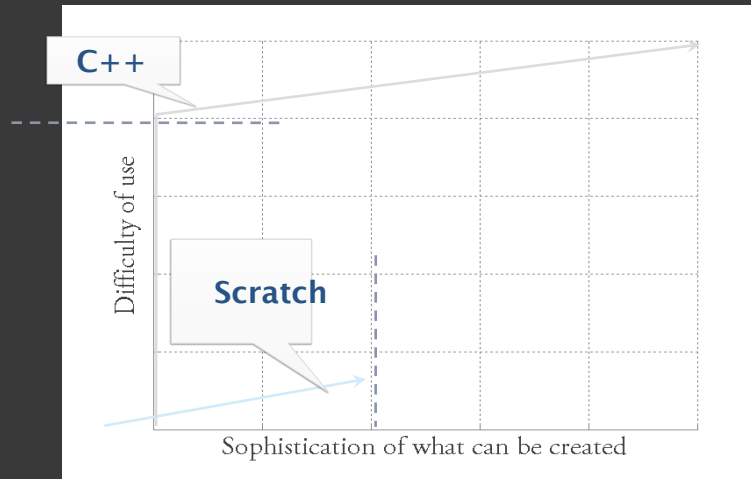
# Model- (ViewController)

## Plotting the Learning Curve





# Threshold, Ceiling, and Walls



# Web Programming

For detailed slides on client stuff,  
see [CMSC122](#)

Client	Communications	Server
HTML	REST	Python
CSS	JSON	SQLite
JavaScript	(AJAX)	Feeds
JQuery		Services

# HTML

- Structured document
- Tags and attributes

## Authoring tools:

- Microsoft Expression Web
- Adobe DreamWeaver
- NVU (free)
- Your favorite text editor
- ...

```
<html>
  <body>
    <h1>Header</h1>
    Some text
    
    <a href="foo.html">Foo</a>
  </body>
</html>
```

# HTML Standards

- Variations across browsers and versions tricky  
=> Toolkits can help

## HTML5 (emerging standard)

- Media
- Canvas
- ...

## CSS3 (emerging standard)

- Prettiness (i.e., drop shadows)
- Multi-column layout
- Hardware accelerated animation
- ...

# CSS – Cascading Style Sheets

- Separates content from visual presentation
- [CSS Zen Garden](#)

Applies **visual styles** to **selections**

Syntax: selector declaration

```
h1 {color: green}
p {
  font-size: 14px;
  color: red;
}
```

## Applying CSS

Inline (to be avoided):

```
<p style="color: red;"> ... </p>
```

Internal (to be minimized):

```
<head>
  <style type="text/css">
    h1 {color: blue}
    p {font-size: 10px}
  </style>
</head>
```

External (encouraged):

```
<head>
  <link rel="stylesheet" href="ExternalFile.css" type="text/css" />
</head>
```

Example ...

# CSS Selectors

- Types (tags)

```
p {color: red}
```

- Pseudo-classes

```
a:link {color: red}  
a:active {color: blue}  
a:visited {color: purple}
```

- Classes

```
HTML                                     CSS  
<div class="nav_item">Top</div>        .nav_item {color: red}
```

- IDs

```
HTML                                     CSS  
<span id="home_nav">Home</span>        #home_nav {color: blue}
```

# Some CSS Properties

- Visibility

```
<head>  
<style type="text/css">  
  #thing1 {visibility:visibile}  
  #thing2 {visibility:hidden}  
</style>  
</head>
```

- Box Model



# CSS Layout

- `display: inline vs. block`
  - `<p>`, `<span>` are inline
  - `<div>` is block
- `position:`
  - static (default) – follows flow
  - relative – modify it's position from where it would have been
  - absolute – relative to first non-static parent
  - fixed – relative to browser window
- `float: (left or right)`
  - Force an item to an edge

# CSS3 (Chrome & Safari for now)

- Hardware accelerated animation

```
-webkit-transition-duration: 0.5s;  
-webkit-transition-timing-function: ease-out;  
-webkit-transition-property: left, top;
```

# JavaScript

- OO interpreter built into browser
- Syntax similar to Java
- Runs in sandbox (no file I/O)
  
- User events
- Can access / modify HTML DOM
- Can talk to server (*it came from*) asynchronously
- Can control browser
  
- => AJAX: Asynchronous Javascript and XML
  - (obsolete with JSON...)

# Applying JavaScript

Internal (to be minimized):

```
<body>
  <script type="text/javascript">
    // YOUR JAVASCRIPT CODE HERE
  </script>
</body>
```

External (encouraged):

```
<head>
  <script type="text/javascript" src="code.js"></script>
</head>
```

Examples ...

# JQuery Library

Provides CSS-like selectors for DOM  
Simpler DOM manipulation  
UI Widget library

[www.jquery.com](http://www.jquery.com)

Example ...

# Python

- Server language
- New syntax
- Many libraries to support
  - files
  - networking
  - string manipulation
  - database connections
  - simple web server for testing
  - ...

[www.python.org](http://www.python.org)

# JSON

- Universal data format for communicating
  - Hierarchical key-value pairs
- Similar to XML but simpler
- Native Javascript

```
{  
  title: "My title",  
  date: "2/17/2001"  
}
```

# REST – Representational State Transfer

- HTTP standard for APIs
  - “RESTful Web Services”
- GET format: uses url parameters
  - Typically used to retrieve information
  - i.e., `http://domain.org/script.py?q="Ben"&order="last_name"`
- POST format: uses HTTP headers
  - Typically used to create new records

EXAMPLE ...



## SQLite

- Like MySQL, but ...
  - Runs off file system – no server
  - Needs CGI permission, but no more
  - Database in simple file that can be copied
  - Simplified management
  - Not as fast, no transactions, etc.

## Putting it all together

- CS Directory Example

## Debugging Tools

- Firefox:
  - Firebug plugin
- Safari:
  - Preferences->Advanced: Show Develop Menu
- Chrome:
  - Built-in

## Publishing Your website

- Terpconnect for files only
- Local for development
- Commercial service for final project
  - Or [Google App Engine](#) free, but must use their DB