



College of Information Studies

University of Maryland Hornbake Library Building College Park, MD 20742-4345

Mobile Apps

INFM 603

Week 10

Agenda

- Questions
- Mobile Apps
- HCI
- Project discussions

Native Apps

- Live on device
- Access to all device features
 - GPS, accelerometer, compass, list of contacts
- Written for specific platform
 - Android, Blackberry, iOS,...
- Can be acquired from an app store
 - Google Play or Apple app store

Mobile Apps

- Not real apps, just websites with look and feel of a native app
- Written to work on specific browser
- Coded in HTML5, Javascript, CSS
- Not in official appstores

Hybrid Apps

- Part native, part web
- The browser is in the app
- Coded in HTML5, CSS and JavaScript
- Wrapper specific to each type of phone
- Available in app store
 - Cheap way of having a presence on the stores

Eye Health Supplements

Eye Health Supplements

Eye Health Supplements

Search

Search

Search

63 Items

63 Items

63 Items

Sort by: Top Sellers

Sort by: Top Sellers

Sort by: Top Sellers

Page : 1

Page : 1

Page : 1



Ocuvite Adult 50+ Lutein & Omega 3 Formula Eye Vitamin & Mineral...

★★★★★ (7)

50 ea

\$16.99

Buy 1, Get 1 50% OFF



Ocuvite Adult 50+ Lutein & Omega 3 Formula Eye Vitamin & Mineral...

★★★★★ (7)

50 ea

\$16.99

Buy 1, Get 1 50% OFF



Ocuvite Adult 50+ Lutein & Omega 3 Formula Eye Vitamin & Mineral...

★★★★★ (7)

50 ea

\$16.99

Buy 1, Get 1 50% OFF



Nature Made Lutein 20 mg Dietary Supplement Liquid Softgels

★★★★★ (4)

30 ea

\$16.99



Nature Made Lutein 20 mg Dietary Supplement Liquid Softgels

★★★★★ (4)

30 ea

\$16.99

Buy 1, Get 1 50% OFF

Find in Store Add to Cart

Find in Store Add to Cart

Find in Store Add to Cart

Comparison

	Web	Hybrid	Native
Dev cost	Reasonable	Reasonable	Expensive
Dev time	Short	Short	Long
App portability	High	High	None
Performance	Fast	Native speed if needed	Very Fast
Native functionality	No	All	All
App store distribution	No	Yes	Yes
Extensible	No	Yes	Yes

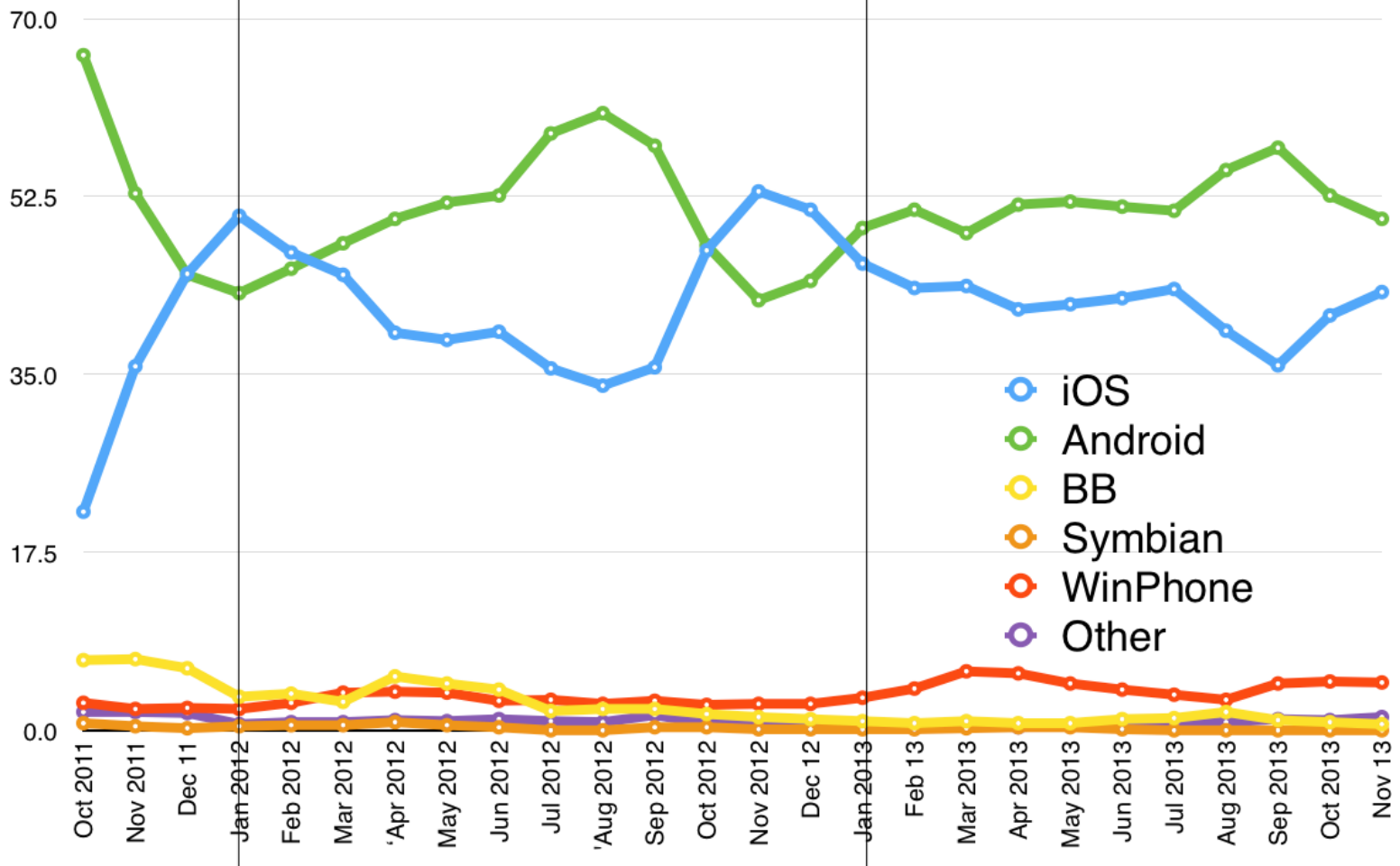
From HTML to the Marketplace

- Generate the code for your app
- Test it on different browsers and platforms
 - Add phone features depending on platform
- “Package” the code to make it an app
 - If Web app, add wrapper around code
 - If native app, simply compile
- Put it in the marketplace

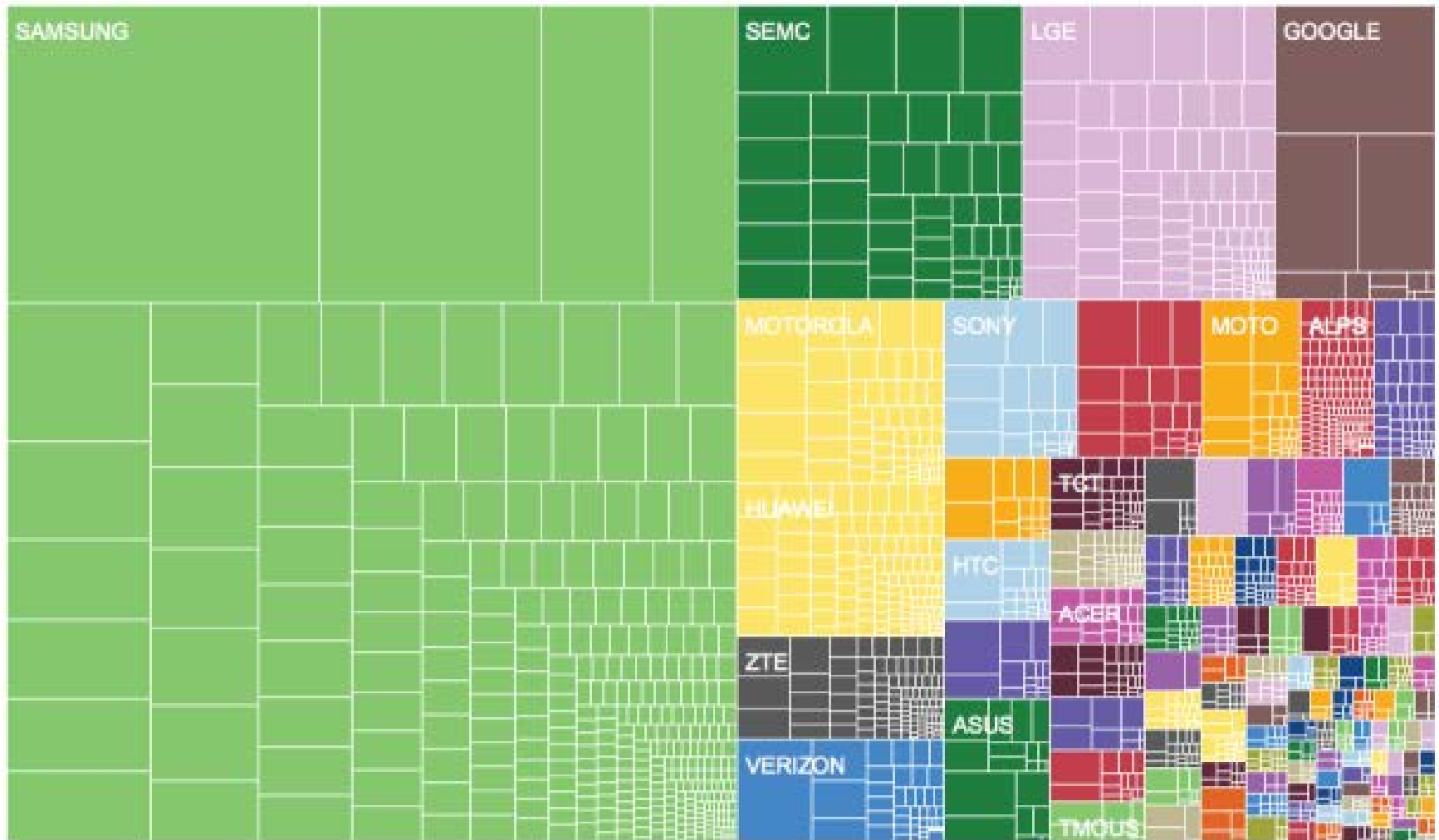
<https://play.google.com/store>

Market Fragmentation: Operating Systems

US smartphone sales market share, Oct 2011-Nov 2013



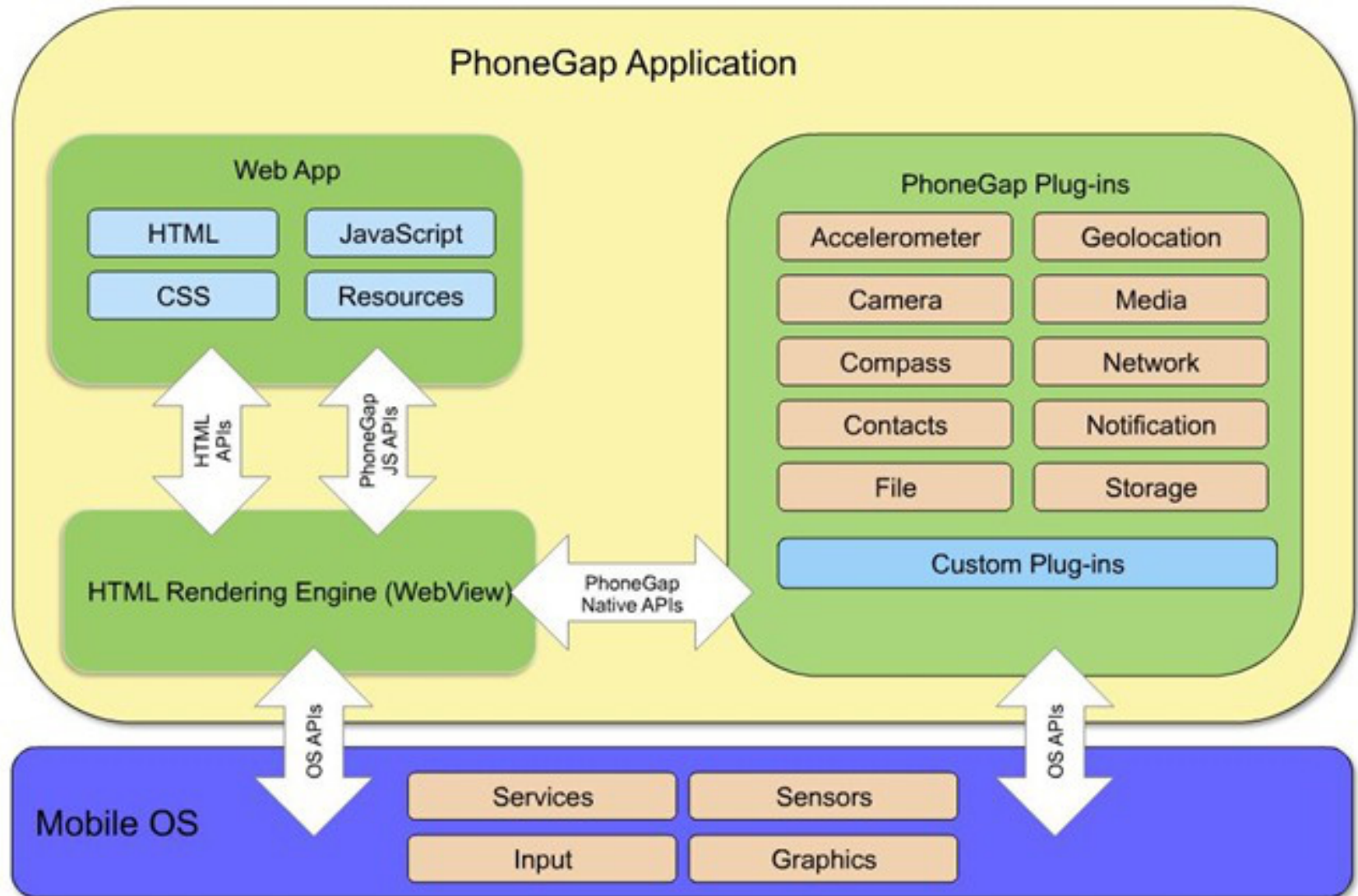
Market Fragmentation: Devices

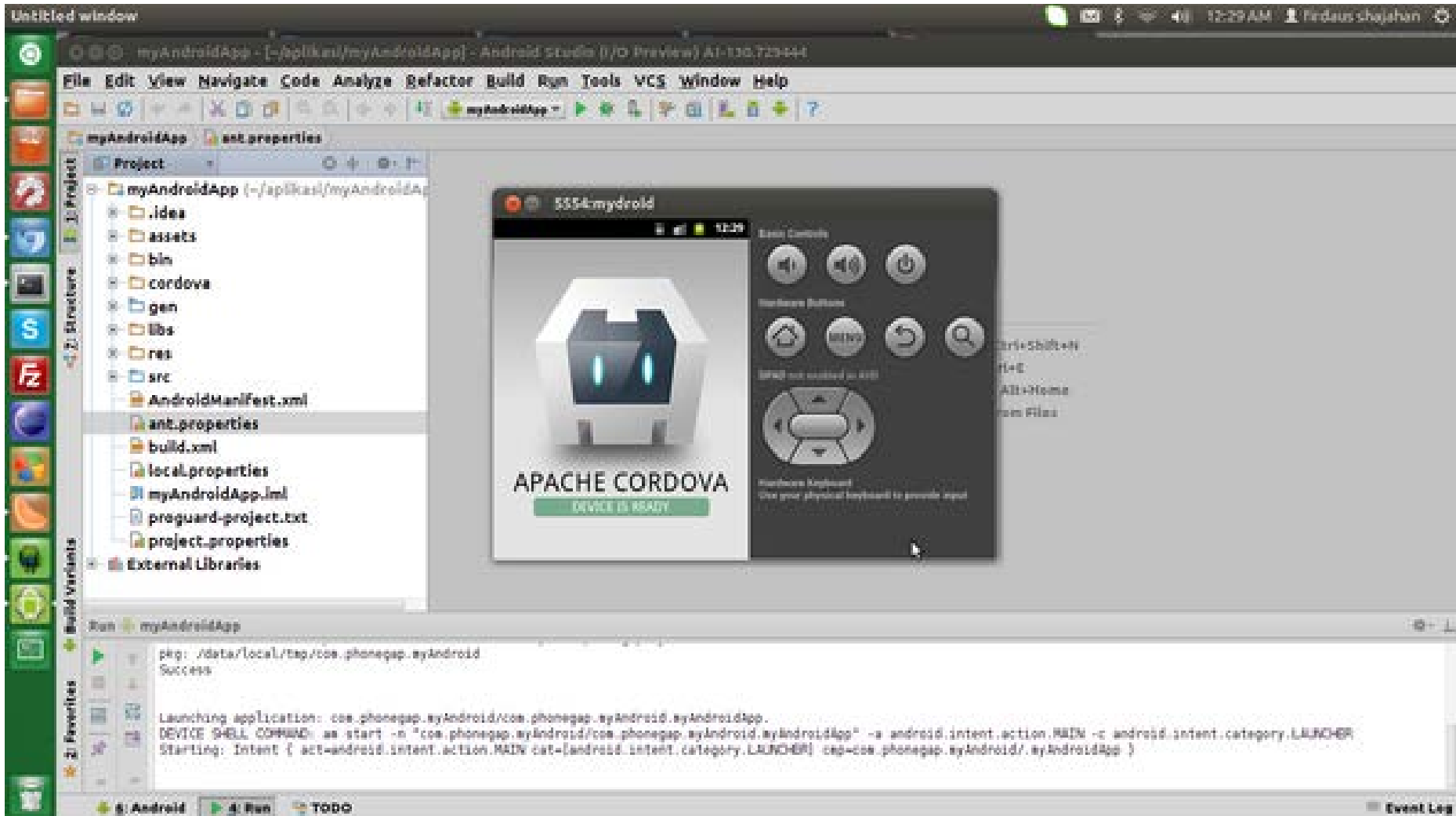


Emulators

- Allow you to test how your code would look like across different types of platforms
 - Sencha or Phonegap
- Compile hybrid applications to make them “native” and ready for market place

PhoneGap





Hybrid App Development

- PhoneGap
 - SDK for each operating system (android, iPhone, ...)
 - Program in HTML, Javascript,...
 - Package as a native app
 - Test in cell phone
- Ripple (<http://emulate.phonegap.com>)
 - Emulates PhoneGap on Chrome

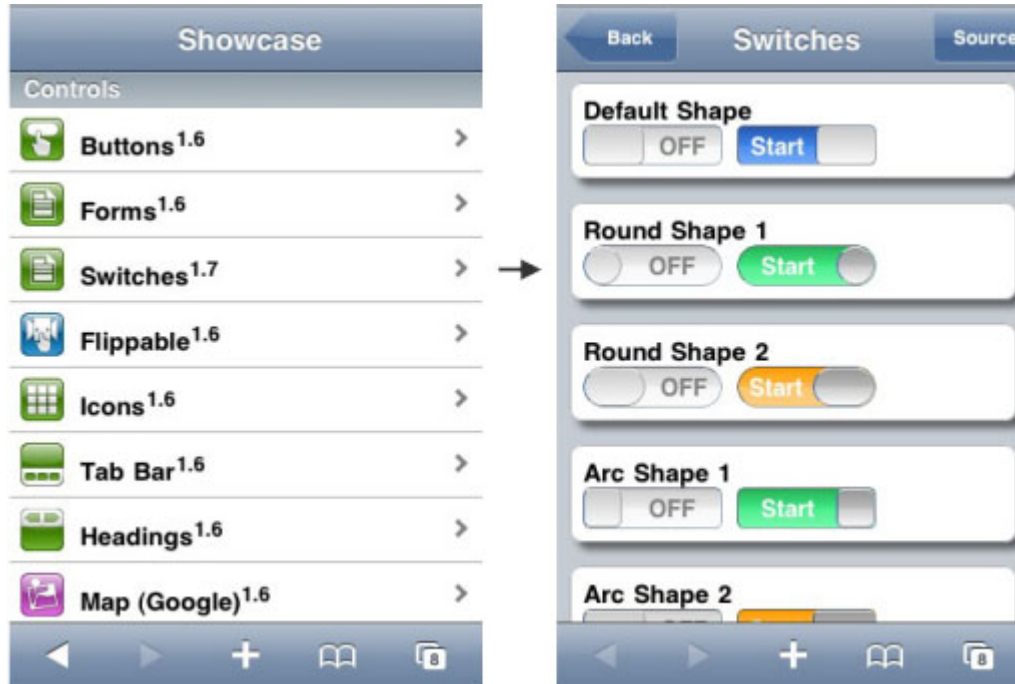
Mobile User Interface

- Smaller form factors
- Touch interfaces
- Acceleration sensing
- Orientation awareness
- Simulations of physical behavior

User Interfaces for Mobiles

Best Practices

- Responsive Layouts



Geolocation

- Location-based services acquire information about where you are
- Think about potential privacy issues

<http://www.google.com/intl/en/privacy/lsf.html>

Geolocation Method

- navigator.geolocation

```
if (navigator.geolocation)  
    navigator.geolocation.getCurrentPosition(showPosition);
```

```
function showPosition(position){  
    x.innerHTML=  
}
```

Geolocation Method

```
function showPosition(position){  
    x.innerHTML  
}
```

The final coordinates are:

`position.coords.latitude`
`position.coords.longitude`

Hands On

- Write a program that prints the geolocation of a user when a button is clicked
 - Test it on your browser
 - Test it on ripple and change geolocation values

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

readlocation.html

```
<!DOCTYPE html>
<html> <body>
<p id="demo">Click the button to get your coordinates:</p>
<button onclick="getLocation()">Try It</button>
<script type="text/javascript" src="phonegap.js"></script>
<script>
var x=document.getElementById("demo");
function getLocation() {
  if (navigator.geolocation) {
    navigator.geolocation.getCurrentPosition(showPosition);
  } else{
    x.innerHTML="Geolocation is not supported by this browser.";
  }
}
function showPosition(position) {
  x.innerHTML="Latitude: " + position.coords.latitude + "<br>Longitude: " +
    position.coords.longitude;
}
</script>
</body> </html>
```

dist2saopaulo.html

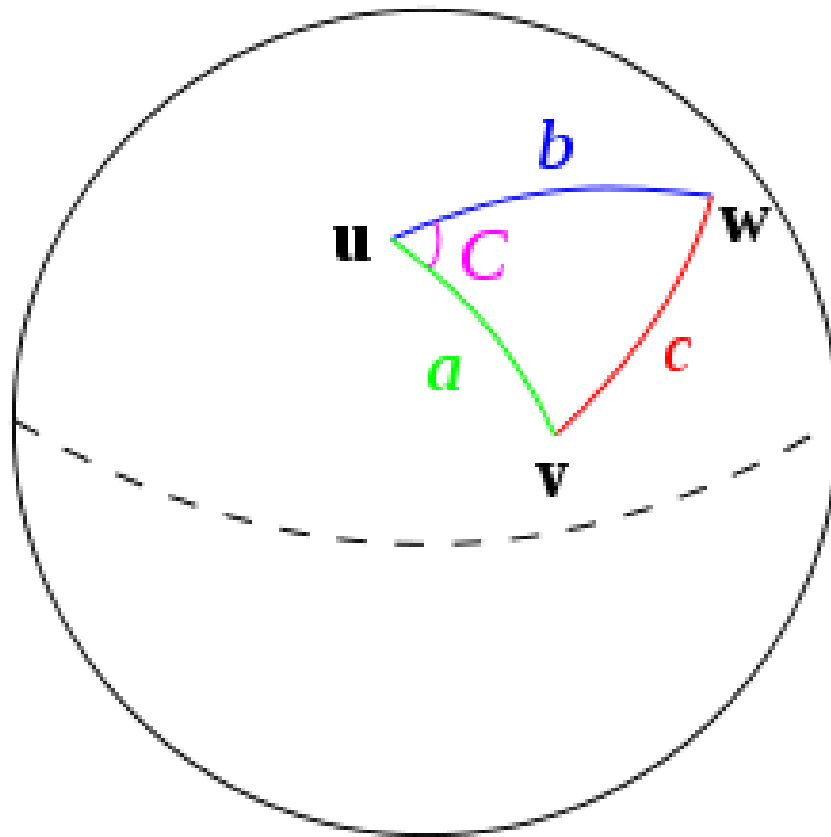
```
<!DOCTYPE html>
<html> <body>
<p id="demo">Click the button to get distance between you and Sao Paolo, Brazil:</p>
<button onclick="getLocation()">Try It</button>
<script type="text/javascript" src="phonegap.js"></script>
<script>
var x=document.getElementById("demo");
function getLocation() {
  if (navigator.geolocation) {
    navigator.geolocation.getCurrentPosition(showPosition);
  } else{
    x.innerHTML=
      "Geolocation is not supported by this browser.";
  }
}

function showPosition(position) {
  var sp1=-23.55;
  var sp2=-46.6333;
  var dist = distance(position.coords.latitude,sp1,position.coords.longitude,sp2);
  x.innerHTML="Latitude: " + position.coords.latitude +
    "<br>Longitude: " + position.coords.longitude + "<br>" + " Distance between the two "+ dist + " km";
}
</script>
</body> </html>
```

```
function distance(lat1,lat2,lon1,lon2) {
  var R = 6371;
  var dLat = toRad(lat2-lat1);
  var dLon = toRad(lon2-lon1);
  var lat1 = toRad(lat1);
  var lat2 = toRad(lat2);
  var a = Math.sin(dLat/2) * Math.sin(dLat/2) +
    Math.sin(dLon/2) * Math.sin(dLon/2) *
    Math.cos(lat1) * Math.cos(lat2);
  var c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1-a));
  var d = R * c;
  return d;
}

function toRad(Value) {
  return Value * Math.PI / 180;
}
```

Haversine Distance



**DC:+38.8951
-77.03**

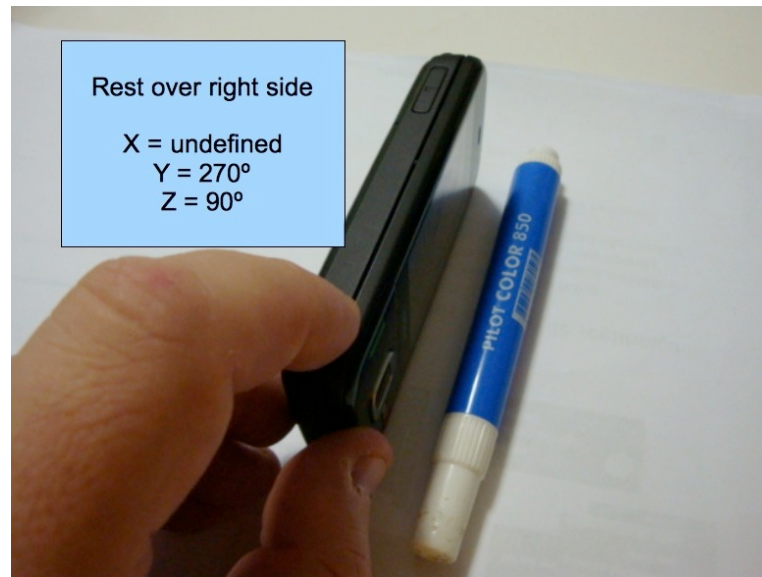
$$d = 2r \arcsin \left(\sqrt{\text{haversion}(\phi_2 - \phi_1) + \cos(\phi_1) \cos(\phi_2) \text{haversion}(\lambda_2 - \lambda_1)} \right)$$

plotlocation.html

```
<!DOCTYPE html>
<html> <body>
<p id="demo">Click the button to get your position:</p>
<button onclick="getLocation()">Try It</button>
<div id="mapholder"></div>
<script>
var x=document.getElementById("demo");
function getLocation() {
  if (navigator.geolocation) {
    navigator.geolocation.getCurrentPosition(showPosition,showError);
  } else{
    x.innerHTML="Geolocation is not supported by this browser.";
  }
}
function showPosition(position) {
  var latlon=position.coords.latitude+","+position.coords.longitude;
  var img_url=http://maps.googleapis.com/maps/api/staticmap?center="+latlon+"&zoom=14&size=400x300&sensor=false";
  document.getElementById("mapholder").innerHTML="<img src='"+img_url+"'>";
}
</script> </body> </html>
```

```
function showError(error) {
  switch(error.code) {
    case error.PERMISSION_DENIED:
      x.innerHTML="User denied the request for Geolocation."
      break;
    case error.POSITION_UNAVAILABLE:
      x.innerHTML="Location information is unavailable."
      break;
    case error.TIMEOUT:
      x.innerHTML="The request to get user location timed out."
      break;
    case error.UNKNOWN_ERROR:
      x.innerHTML="An unknown error occurred."
      break;
  }
}
```

Accelerometer



readaccelerometer.html

```
<!DOCTYPE html>
<html> <head>
  <title>Acceleration Example</title>
  <script type="text/javascript" charset="utf-8" src="phonegap-1.2.0.js"></script>
  <script type="text/javascript" charset="utf-8">
    // Wait for PhoneGap to load
    document.addEventListener("deviceready", onDeviceReady, false);
    // PhoneGap is ready
    function onDeviceReady() {
      navigator.accelerometer.getCurrentAcceleration(onSuccess, onError);
    }
    // onSuccess: Get a snapshot of the current acceleration
    //
    function onSuccess(acceleration) {
      alert('Acceleration X: ' + acceleration.x + '\n' +
        'Acceleration Y: ' + acceleration.y + '\n' +
        'Acceleration Z: ' + acceleration.z + '\n' +
        'Timestamp: ' + acceleration.timestamp + '\n');
    }
    // onError: Failed to get the acceleration
    //
    function onError() {
      alert('onError!');
    }
  </script>
</head> <body>
  <h1>Example</h1>
  <p>getCurrentAcceleration</p>
</body> </html>
```

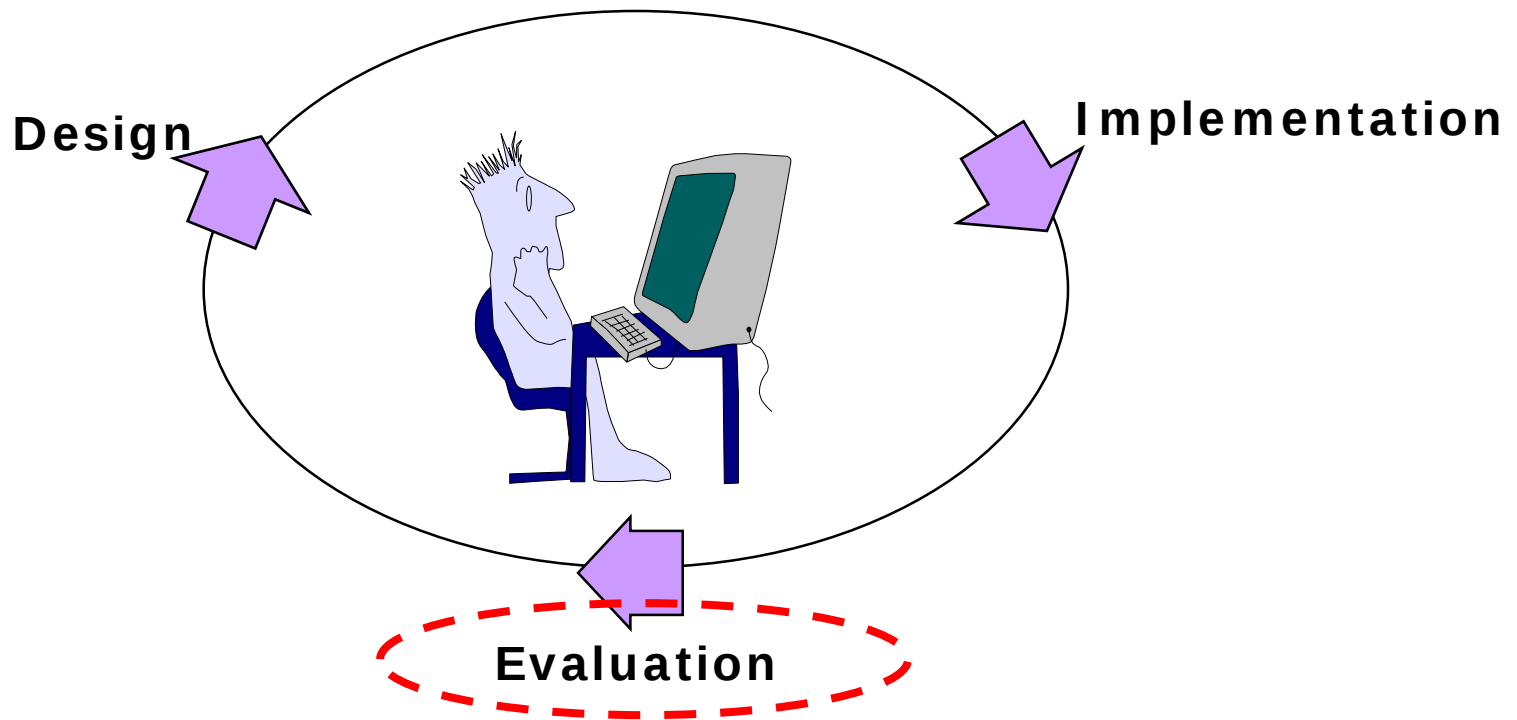
only works with cordova

faceup.html

```
<!DOCTYPE html> <!DOCTYPE html>
<html> <head>
<title>Acceleration Example</title>
<script type="text/javascript" charset="utf-8"
  src="phonegap-1.2.0.js"></script>
<script type="text/javascript" charset="utf-8">
  var watchID = null;
  document.addEventListener("deviceready", onDeviceReady, false);
  function onDeviceReady() {
    startWatch();
  }
  function onSuccess(acceleration) {
    var element = document.getElementById('accelerometer');
    if ((0.9<=acceleration.y) && (acceleration.y<= 1.1) && (0<=acceleration.z) && (acceleration.z <= 0.1)&&
      (0<=acceleration.x) && (acceleration.z <= 0.1) ) {
      element.innerHTML = 'Acceleration X: ' + acceleration.x + '<br />' + 'Acceleration Y: ' + acceleration.y + '<br />' +
        'Acceleration Z: ' + acceleration.z + '<br />' + 'Timestamp: ' + acceleration.timestamp + '<br />' +
        "<img src=upright.JPG>";
    } else {
      element.innerHTML = 'Acceleration X: ' + acceleration.x + '<br />' + 'Acceleration Y: ' + acceleration.y + '<br />' +
        'Acceleration Z: ' + acceleration.z + '<br />' + 'Timestamp: ' + acceleration.timestamp + '<br />' + "not upright";
    }
  }
</script> </head> <body> <div id="accelerometer">Waiting for accelerometer...</div> </body> </html>
```

```
function startWatch() {
  var options = { frequency: 100 };
  watchID =
    navigator.accelerometer.watchAcceleration(
      onSuccess, onError, options);
}
function stopWatch() {
  if (watchID) {
    navigator.accelerometer.clearWatch(watchID);
    watchID = null;
  }
}
function onError() {
  alert('onError!');
}
```

Human-Computer Interaction



of interactive computing systems for human use

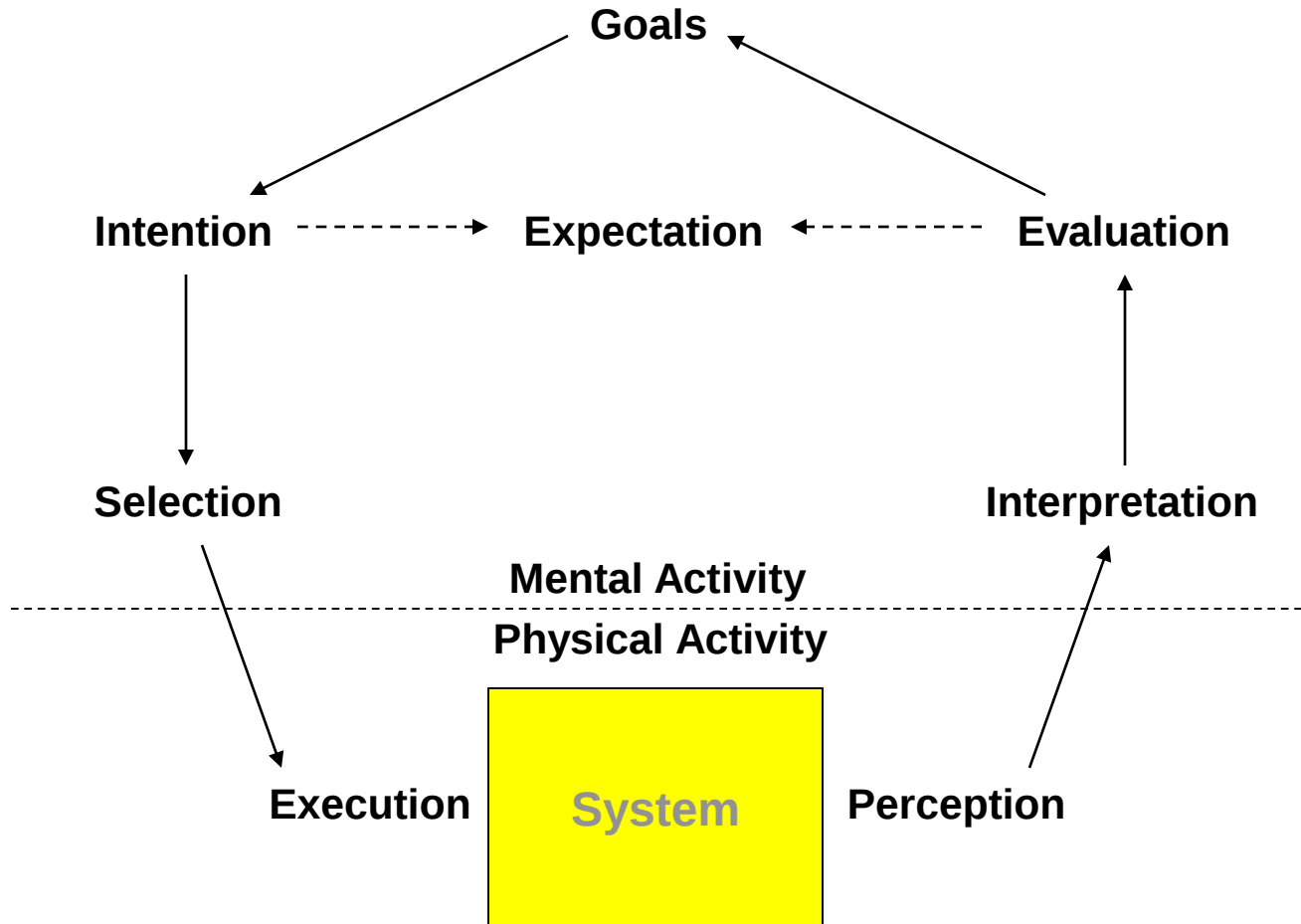
Synergy

- Humans do what they are good at
- Computers do what they are good at
- Strengths of one cover weakness of the other

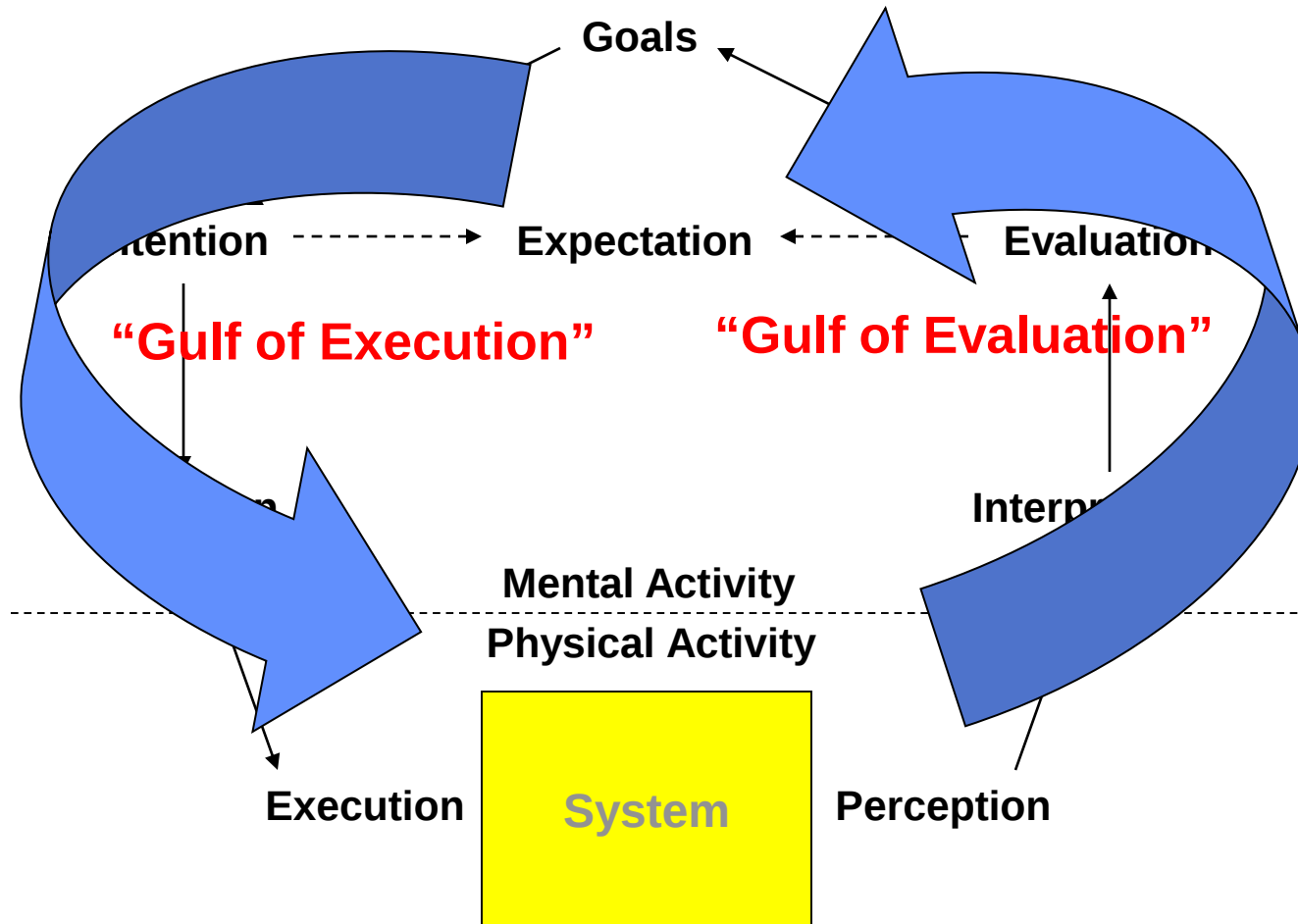
Interaction

- Forming an intention
 - Internal mental characterization of a goal
- Selection of an action
 - Review possible actions, select most appropriate
- Execution of the action
 - Carry out appropriate actions with the system
- Evaluation of the outcome
 - Compare results with expectations

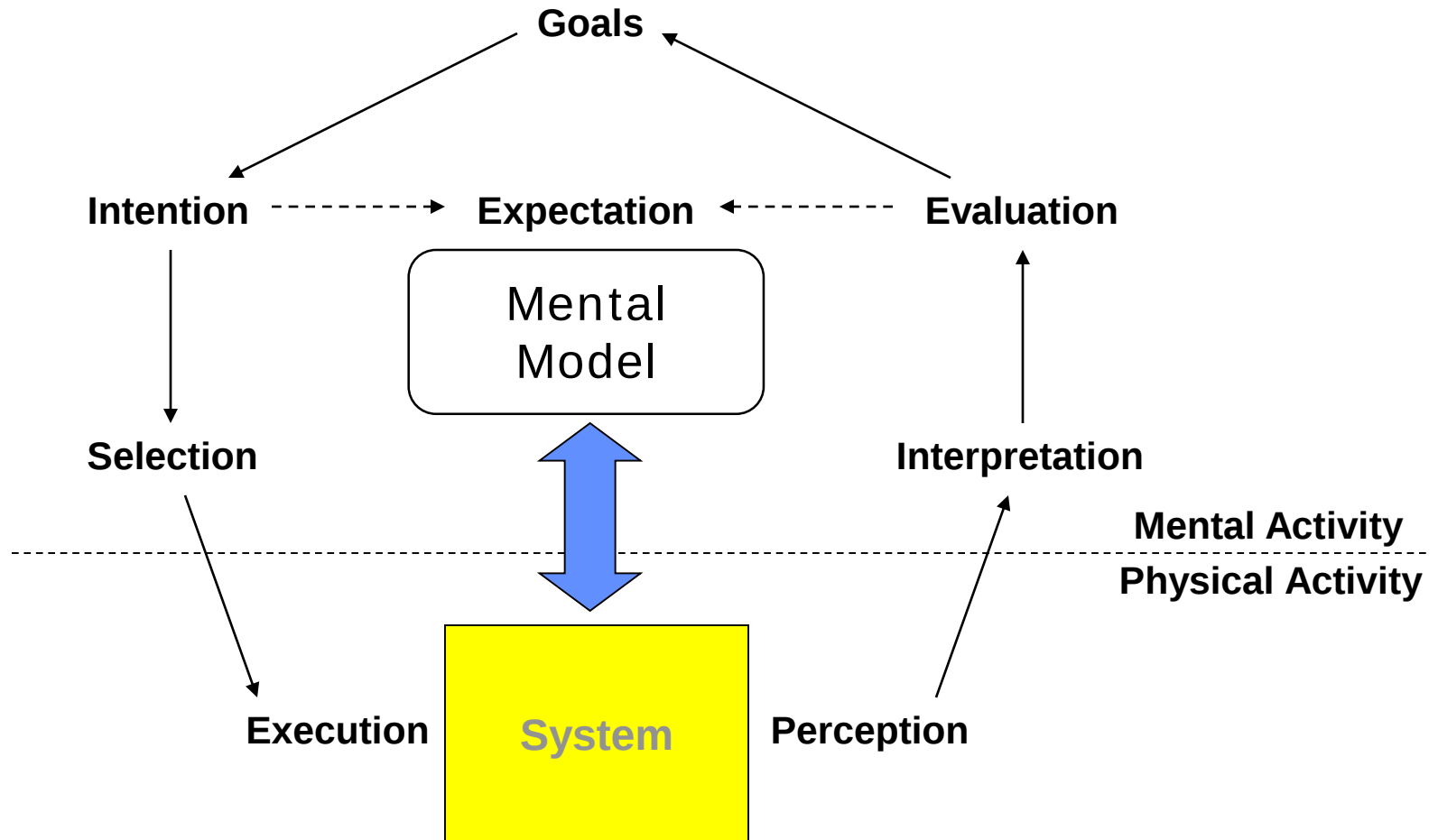
Stages of Interaction



Challenges of HCI

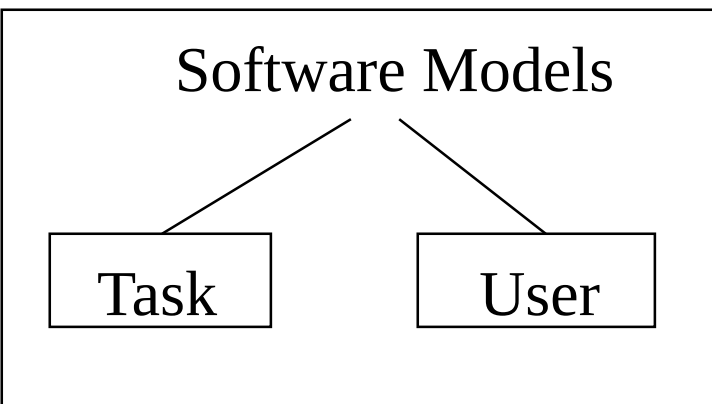
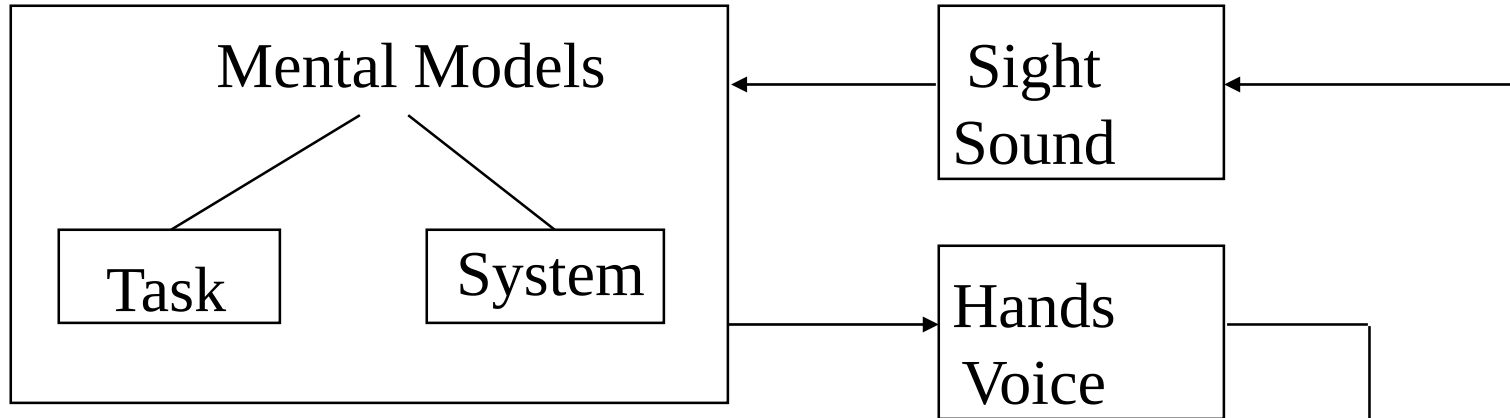


What is good design?



Modeling Interaction

Human



Keyboard
Mouse

Display
Speaker

Computer

Mental Models

- How the user thinks the machine works
 - What actions can be taken?
 - What results are expected from an action?
 - How should system output be interpreted?
- Mental models exist at many levels
 - Hardware, operating system, and network
 - Application programs
 - Information resources

Evaluation Approaches

- Formative vs. summative
- Extrinsic vs. intrinsic
- Quantitative vs. qualitative
 - Deductive vs. inductive
- User study vs. simulation

Evaluation Examples

- Direct observation
 - Evaluator observes users interacting with system
 - in lab: user asked to complete pre-determined tasks
 - in field: user goes through normal duties
 - Validity depends on how contrived the situation is
- Think-aloud
 - Users speak their thoughts while doing the task
 - May alter the way users do the task
- Controlled user studies
 - Users interact with system variants
 - Correlate performance with system characteristics
 - Control for confounding variables

Evaluation Measures

- Time to learn
- Speed of performance
- Error rate
- Retention over time
- Subjective satisfaction