



YUI 3

<http://developer.yahoo.com/yui3>

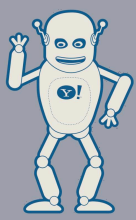
<http://github.com/yui/yui3>

<http://github.com/davglass/openhackday>

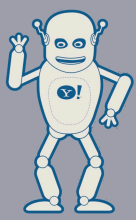
<http://blog.davglass.com>

@davglass

YAHOO!
DEVELOPER NETWORK



```
YUI.use('lighter', function(Y) {  
    Y.all('.faster li').each(function(v) {  
        v.set('innerHTML', 'easier');  
    });  
});
```



● **Lighter**

- Don't load what you don't need
- Don't write code more than once, use it again

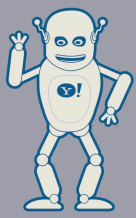
● **Easier**

- Consistent API
Base, Selector, Widget, IO/Get/DataSource
- Convenience
each, bind, nodelist, queue, chainability, general sugar

● **Faster**

- Opportunity to re-factor core performance pain points



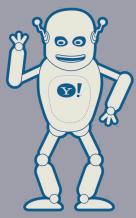


Why the change?

What our users asked for:

- Selectors
- Chaining
- Consistency
- Sandboxing
- Better Dependency Handling
- Versioning

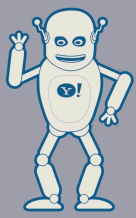




Protected Code

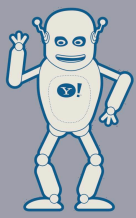
```
<script src="/3.4/build/yui/yui-min.js">
<script>
  YUI().use("overlay", function(Y) {
    Y.on("click", function() {
      new Y.Overlay({...}).render();
    }, "#button");
  });
</script>
```





Protected Code

```
<script src="/3.2/build/overlay/overlay-min.js">
<script>
  YUI().use("overlay", function(Y) {
    Y.on("click", function() {
      new Y.Overlay({...}).render();
    }, "#button2");
  });
</script>
```

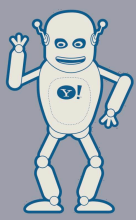


Self Populating

```
<script src="yui-min.js">
<script>
    YUI().use("anim", function(Y) {

    });
</script>
```





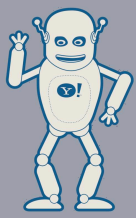
Self Populating

```
<script src="yui-min.js">
<script src="oop-min.js">
<script src="event-min.js">
<script src="attribute-min.js">
<script src="base-min.js">
<script src="dom-min.js">
<script src="node-min.js">
<script src="anim-min.js">

<script>
    YUI().use("anim", function(Y) {

    });
</script>
```



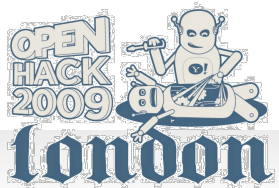


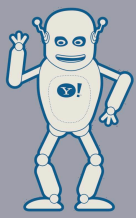
Self Populating

```
<script src="yui-min.js">
<script src="http://yui.yahooapis.com/combo?oop-min.js&event-min.js..">

<script>
    YUI().use("anim", function(Y) {

        });
</script>
```



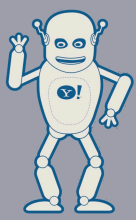


Event

The Enhanced Custom Event System Is An Integral Part of YUI 3's Goal To Be Lighter, Allowing For Highly Decoupled Code

- Event Facades
- Built-in on and after moments
- Default Behavior (preventDefault)
- Event Bubbling (stopPropagation)
- Detach Handles



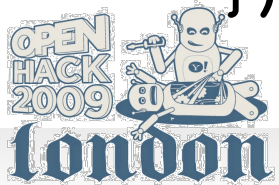


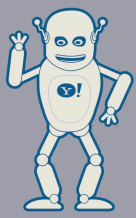
Event

```
// Dom Event
YAHOO.util.Event.on("click", function(e) {
    YAHOO.util.Event.preventDefault(e);
});

// Dom Event
YAHOO.util.Event.on("mousemove", function(e) {
    YAHOO.util.Event.getPageX(e);
});

// Custom Event
slider.on("valueChange", function(e) {
    if (someVal < 200) {
        return false;
    }
});
```

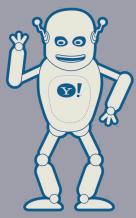




Event Facade

```
// Dom Event
linkNode.on("click", function(e) {
  if (!e.target.hasClass("selector")) {
    e.preventDefault();
  }
  var x = e.pageX;
});

// Custom Event
slider.on("valueChange", function(e) {
  if (e.newVal < 200) {
    e.preventDefault();
  }
});
```

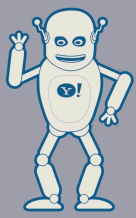


Event Bubbling

```
var dd = new Y.DD.Drag({ node: '#drag' });

//Listen at a higher level
Y.DD.DDM.on('drag:drag', function(e) {
    if (tooFar) {
        e.preventDefault();
    }
});

//Set opacity on all drag objects
Y.DD.DDM.on('drag:start', function(e) {
    var dds = Y.all('.yui-dd-draggable').
        setStyle('opacity', '.5');
});
```



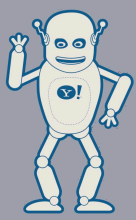
Event

```
// Publisher
show : function() {
    this.fire("show");
},

_defShowFn : function(e) {
    node.removeClass("hidden");
    ...
}

// Subscriber
overlay.on("show", function(e) {
    if (!taskSelected) { e.preventDefault(); }
})

overlay.after("show", function(e) {...});
```



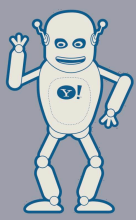
Node

A single convenient location for working with HTML Elements

- Selector Based
- Supports
- Normalizes
- Enhances

- Extendable
- Constraining





Node

Working with Node:

```
Y.get('#foo').addClass('selected').  
  set('innerHTML', 'Here');
```

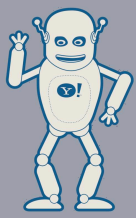
```
var foo = Y.get('#foo');
```

```
foo.on('click', function() { ... });
```

```
foo.addClass('selected');
```

```
foo.getXY();
```

```
foo.setStyle('backgroundColor', 'red');
```

Node

Supports the HTMLElement API

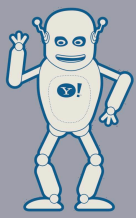
```
node.appendChild(aNode)
```

```
node.cloneNode(aNode)
```

```
node.scrollIntoView()
```

```
node.get("parentNode")
```

```
node.set("innerHTML", "Foo")
```



Node

Normalizes the HTMLElement API

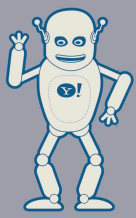
```
node.getAttribute("href")
```

```
node.contains(aNode)
```

```
node.getText()
```

```
node.getStyle("paddingTop")
```

```
node.previous()
```



Node

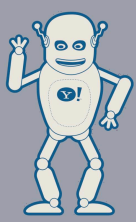
Enhances The HTMLElement API

```
node.addClass("selectable")
```

```
node.toggleClass("enabled")
```

```
node.getXY()
```

```
node.get("region")
```

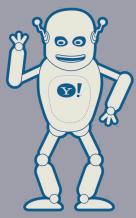


Node

Extendable

Plugins can provide app specific features...

```
node.plugin(IOPlugin);  
node.io.getContent("http://foo/bar");  
  
node.plugin(DragPlugin);  
node.dd.set("handle", ".title");
```

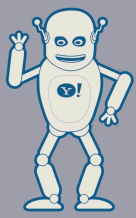


Node

Constrainable

Node is the single point for DOM access, throughout YUI 3

Makes YUI 3 ideal as a trusted source in "constrained" environments such as Caja and Ad-Safe

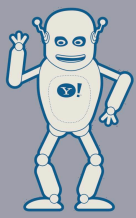


NodeList

Used to Batch Node operations

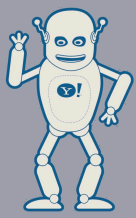
```
var items = Y.Node.all(".actions li");
items.addClass("disabled");
items.set("title", "Item Disabled");

items.each(function(node) {
    node.addClass("disabled");
    node.set("title", "Item Disabled");
});
```



Core Language Addons

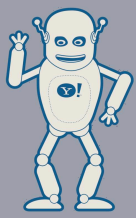
- Array Extras
- isString, isNumber ...
- Bind
- Each
- Later
- OOP
 - Augment, Extend, Aggregate, Merge, Clone
- AOP
 - Before/After For Methods



A Common Foundation

Y.Attribute

- Configurable Attributes
 - readOnly, writeOnce
 - validators, getters and setters
- Attribute Value Change Events
 - on/after
- Complex Attribute Support
 - `set("strings.label_enabled", "Enabled");`

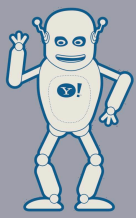


A Common Foundation

Y.Base

- The Class From Which YUI 3 Classes Will Be Derived
- Combines Custom Event And Attribute Support
- Manages the "init" and "destroy" Lifecycle Moments
- Provides Plugin/Extension Management

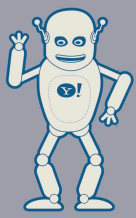




A Common Foundation

Y.Widget

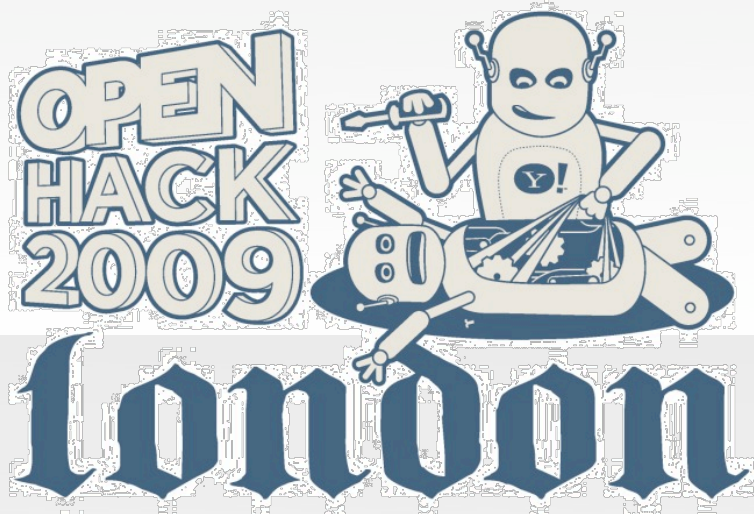
- The Base Implementation For All Widgets
- Introduces Common Attributes, Methods
 - `boundingBox`, `contentBox`
 - `width`, `height`
 - `visible`, `disabled`, `hasFocus`
 - `strings`
 - `show()`, `hide()`, `focus()`, `blur()`, `enable()`, `disable()`
- Manages The "render" Lifecycle Moment
- Establishes A Common Pattern For Widget Development



Code...

Finally, let's see some code..





YUI 3

<http://developer.yahoo.com/yui3>

<http://github.com/yui/yui3>

<http://github.com/davglass/openhackday>

<http://blog.davglass.com>

@davglass

YAHOO!
DEVELOPER NETWORK