



# The Yahoo! User Interface Library (YUI)

Nate Koechley  
[nate@koechley.com](mailto:nate@koechley.com)  
<http://nate.koechley.com/blog>

Refresh 06  
Orlando, Florida  
2006.11.17

The image features a black background with several flowing, wavy lines in shades of purple and white. These lines create a sense of movement and depth, resembling liquid or smoke. The word "YUI" is centered in a bold, white, sans-serif font.

# YUI

<http://www.flickr.com/photos/cdm/50688378/in/set-1002108/>



# The Yahoo! Developer Network

---

- Utility and Data Web Services
- Design Patterns Library
- Browser Support Guidelines
- Yahoo! User Interface Library (YUI)



# The Yahoo! Developer Network Ecosystem: Utility and Data Web Services

---

## Services

[Browser Based Authentication \(BBAuth\)](#), [Answers](#),  
[Local](#), [Maps](#), [Photos](#), [Search](#), [Shopping](#), [Travel](#),  
[Utilities](#), (Mail coming soon)

## Hackable Sites

[del.icio.us](#), [Flickr](#), [Upcoming.org](#), [Webjay](#)

## Burnable Feeds

[Finance](#), [HotJobs](#), [RSS Feeds](#), [Traffic](#), [Weather](#)

## SDKs

[Messenger](#), [Music](#), [Search Developer Kit](#), [Widgets](#)

## Developer Centers

[JavaScript](#), [Flash](#), [.NET](#), [PHP](#), [Python](#), [Ruby](#)



# The Yahoo! Developer Network Ecosystem: Design Patterns

## YAHOO! DEVELOPER NETWORK

### Design Pattern Library

[Yahoo! Developer Network](#) > Design Pattern Library

#### USER NEEDS TO

##### ▼ NAVIGATE

[Breadcrumbs](#)

► [Faceted Navigation](#)

[Fly-out Menus](#)

[Horizontal Bar](#)

[Hub and Spoke](#)

[Left Navigation](#)

▼ [Tabs](#)

[Module Tabs](#)

[Navigation Tabs](#)

##### ▼ EXPLORE DATA

[Auto Complete](#)

[Calendar Picker](#)

▼ [Pagination](#)

[Item Pagination](#)

[Search Pagination](#)

► [ORGANIZE DATA](#)

▼ [GIVE FEEDBACK](#)

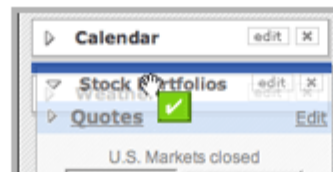
### Welcome

Welcome to the Yahoo! Design Pattern Library. You're looking at the second of many periodic releases, many of which include pointers to the [Yahoo! User Interface Library](#) code. We're thrilled to be sharing patterns and code with the design and development community, hope it's useful, and look forward to your [feedback](#).

### What's a Pattern?

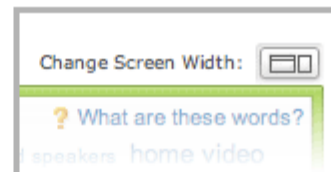
A pattern describes an optimal solution to a common problem within a specific context. [more...](#)

Recent Patterns [see all...](#)



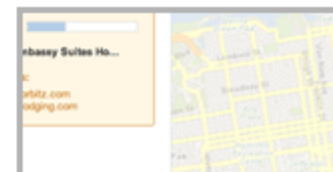
#### [Animate Transition](#)

Designer needs to communicate that an object is changing its spatial relationship within the page.



#### [Collapse Transition](#)

The designer needs to communicate that an object is no longer of primary importance, yet wants to keep it available in a smaller



#### [Dim Transition](#)

Designer needs to communicate that an object is of secondary importance.



# The Yahoo! Developer Network Ecosystem: Design Patterns

## YAHOO! DEVELOPER NETWORK

### Design Pattern Library

[Yahoo! Developer Network](#) > [Design Pattern Library](#) > Auto Complete

#### Auto Complete

##### Problem Summary

The user needs to enter an item into a text box which could be ambiguous or hard to remember and therefore has the potential to be mis-typed.

##### EXAMPLE:

The screenshot shows a 'Compose' window in Yahoo! Mail Beta. The 'To:' field contains 'yu' and a dropdown menu is open, showing two suggestions: 'Yusef Jones <yusef@somewhere.com>' and 'Yusef Smith <yusefs@someplace.org>'. The 'Cc:' field contains 'Yusef Smith <yusefs@someplace.org>' and the 'Subject:' field contains 'yui blogger <yuiblogger@yahoo.com>'. The window also shows a 'Send' button, 'Save Draft', 'Attach', 'Check Spelling', and 'Cancel' buttons. The text area contains 'Hello.' and a 'Play' button. The font is set to Verdana, size 10.

*Auto completion of contacts in Yahoo! Mail Beta*

##### QUICK JUMP

- [Solution](#)
- [Rationale](#)
- [Accessibility](#)

##### RELATED PATTERNS

- [Button Sequence](#)
- [Chooser](#)
- [Input Guidance](#)

##### AS USED ON YAHOO!

- [Yahoo! Mail Beta](#)
- [Yahoo! Maps Beta](#)
- [Yahoo! My Web 2](#)

##### BLOG

- [Blog Article](#)

##### Y! UI CODE EXAMPLES

- [Auto Complete Component](#)



# The Yahoo! Developer Network Ecosystem: Design Patterns



## Use When

- The suggestions can be pulled from a manageable set of data.
- The input item can be entered in multiple ways.
- The input item can be matched with a specific data item in the system.
- Speed and accuracy of entry is an important goal.
- The total number of items would be too large or inconvenient for displaying in a standard drop down box.

## Solution

### Layout

- Use a standard text box for input.
- Label the text box to match the user's expectation of what field will be searched against.

### Interaction

- As the user types, display a list of suggested items that most closely match what the user has typed. Continue to narrow or broaden the list of suggested items based on the user's input.
- Display the suggested items list in a drop down box directly underneath the text box. The suggested items list may be based on the complete set of data or more narrowly based on other criteria such as each item's frequency of use.
- When available, show multiple fields of information for each suggested item. In the

## Y! UI CODE EXAMPLES

- [Auto Complete Component](#)
- [Y! Search \(JSON\)](#)
- [Y! Search \(XML\)](#)
- [In-Memory \(array\)](#)
- [In-Memory \(function\)](#)
- [Flickr \(XML\)](#)
- [Flat data](#)
- [Playground](#)

[Show with revisions](#)



This work is licensed under a [Creative Commons Attribution 2.5 License](#).



# The Yahoo! Developer Network Ecosystem: Browser Support Guidelines

## YAHOO! DEVELOPER NETWORK

### Yahoo! UI Library: Graded Browser Support

#### Yahoo! UI Library

[Home](#)  
[YUIBlog](#)  
[YUI Discussion Forum](#)  
[YUI on Sourceforge](#)  
[API Documentation](#)  
[Examples & Implementations](#)  
[YUI Theater](#)

#### YUI Articles

##### Graded Browser Support

[Table of A-Grade Browsers](#)

#### YUI Components

[Animation](#)  
[AutoComplete](#)  
[Calendar](#)  
[Connection Manager](#)  
[Container](#)  
[Dom](#)  
[Drag & Drop](#)  
[Event](#)  
[Logger](#)  
[Menu](#)  
[Slider](#)  
[TabView](#)  
[TreeView](#)

## Graded Browser Support

By Nate Koechley  
Senior Front-End Engineer  
Yahoo! Inc.  
February 13, 2006

In the first 10 years of professional web development, back in the early '90s, browser support was binary: Do you — or don't you — support a given browser? When the answer was "No", user access to the site was often actively prevented. In the years following IE5's release in 1998, professional web designers and developers have become accustomed to asking at the outset of any new undertaking, "Do I have to support Netscape 4.x browsers for this project?"

By contrast, in modern web development we must support *all* browsers. Choosing to exclude a segment of users is inappropriate, and, with a "Graded Browser Support" strategy, unnecessary.

Graded Browser Support offers two fundamental ideas:

- A broader and more reasonable definition of "support."
- The notion of "grades" of support.

### What Does "Support" Mean?

Support does not mean that everybody gets the same thing. Expecting two users using different browser software to have an identical experience fails to embrace or acknowledge the heterogeneous essence of the Web. In fact, requiring the same experience for all users creates a barrier to participation. Availability and accessibility of content should be our key priority.





# The Yahoo! Developer Network Ecosystem: Browser Support Guidelines

## YAHOO! DEVELOPER NETWORK

### Yahoo! UI Library: Table of A-Grade Browsers

#### Yahoo! UI Library

Home  
YUIBlog  
YUI Discussion Forum  
YUI on Sourceforge  
API Documentation  
Examples & Implementations  
YUI Theater

#### YUI Articles

Graded Browser Support

**Table of A-Grade Browsers**

#### YUI Components

Animation  
AutoComplete  
Calendar  
Connection Manager  
Container  
Dom  
Drag & Drop  
Event  
Logger  
Menu  
Slider  
TabView  
TreeView

## A-Grade Browser Support Chart

This chart lists all browsers that receive A-Grade support as defined by [Graded Browser Support](#).

	Win 98	Win 2000	Win XP	Mac 10.3.x	Mac 10.4
IE 7.0			A-grade		
IE 6.0	A-grade	A-grade	A-grade		
Firefox 2.0.*	A-grade	A-grade	A-grade	A-grade	A-grade
Firefox 1.5.*	A-grade	A-grade	A-grade	A-grade	A-grade
Opera 9.*	A-grade	A-grade	A-grade	A-grade	A-grade
Safari 2.0*					A-grade

## Changes

### Primary Changes

1. REITERATION of A-Grade support, IE 7, Win
2. TERMINATION of A-Grade support, IE 5.5, Win
3. INITIATION of A-Grade support, Firefox 2.x, Win/Mac
4. TERMINATION of A-Grade support Firefox 1.0.x Win/Mac

## Yahoo! UI Library (YUI)

### Yahoo! UI Library

#### Home

[YUIBlog](#)  
[YUI Discussion Forum](#)  
[YUI on Sourceforge](#)  
[API Documentation](#)  
[Examples & Implementations](#)  
[YUI Theater](#)

#### YUI Articles

[Graded Browser Support](#)  
[Table of A-Grade Browsers](#)

#### YUI Components

[Animation](#)  
[AutoComplete](#)  
[Calendar](#)  
[Connection Manager](#)  
[Container](#)  
[Dom](#)  
[Drag & Drop](#)  
[Event](#)  
[Logger](#)  
[Menu](#)  
[Slider](#)  
[TabView](#)  
[TreeView](#)  
[YAHOO Global Object](#)  
[Reset CSS](#)

## Yahoo! User Interface Library

The Yahoo! User Interface (YUI) Library is a set of utilities and controls, written in JavaScript, for building richly interactive web applications using techniques such as DOM scripting, DHTML and AJAX. The YUI Library also includes several core CSS resources. All components in the YUI Library have been released as open source under a [BSD license](#) and are free for all uses. [Download](#) the entire library, its documentation and examples from Sourceforge.

The library's developers blog [at the YUI Blog](#) and the YUI Library community exchanges ideas at [YDN-JavaScript on Yahoo! Groups](#).

#### On This Page:

[Getting Started](#)  
[Licensing](#)  
[Download](#)  
[Components](#)  
[Design Patterns](#)  
[Articles](#)  
[Cheat Sheets](#)  
[Support & Community](#)

#### Searchable API Documentation **NEW!**

API docs are available for every YUI component. You can search the API

#### YUI Library Utilities:

[The YAHOO Global Object](#)  
[Event Utility](#)  
[DOM Collection](#)  
[Connection Manager](#)  
[Drag and Drop Utility](#)  
[Animation Utility](#)

#### YUI Library CSS

**Tools:**  
[CSS Grids](#)  
[CSS Fonts](#)  
[CSS Reset](#)

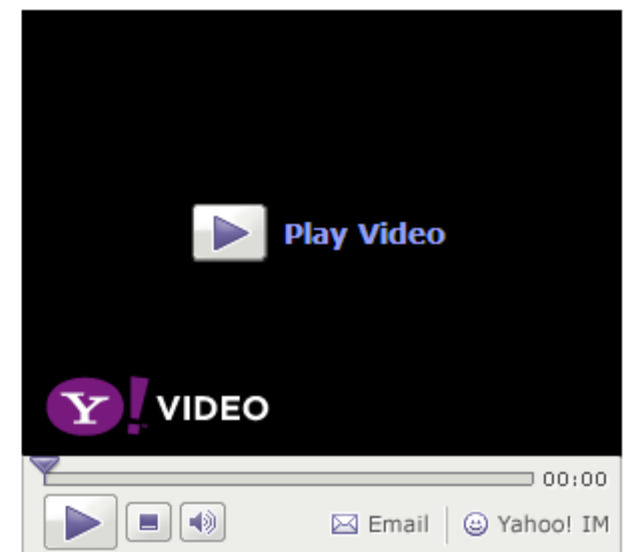
#### YUI Library Controls:

[AutoComplete](#)  
[Calendar](#)  
[Container \(including \[Module\]\(#\), \[Overlay\]\(#\), \[Panel\]\(#\), \[Tooltip\]\(#\), \[Dialog\]\(#\), \[SimpleDialog\]\(#\)\)](#)  
[Logger](#)  
[Menu](#)  
[Slider](#)  
[TabView \*\*NEW!\*\*](#)  
[TreeView](#)

#### YUI Theater:

[\[more videos\]](#)

### Matt Sweeney, "Web 2.0: Getting It Right the Second Time"



YUI developer Matt Sweeney delivers a talk on frontend architecture for Web 2.0 at Yahoo!'s [Open Hack Day](#). (Visit [Yahoo! Video](#) for the full-format version.)

#### Yahoo! User Interface Blog **RSS**

[Graded Browser Support: Q4 Update](#)  
 Nate Koechley 11/15/2006 1:42 AM



# changing seasons



# on the web



People expect **less** online.



But we **are** online...

**... and tied to the browser.**



An aerial photograph of a lush green soccer field. White lines mark the boundaries of the field, including the center circle, the halfway line, and the goal areas. The grass is vibrant green and appears well-maintained.

**So we must  
level  
the playing field.**



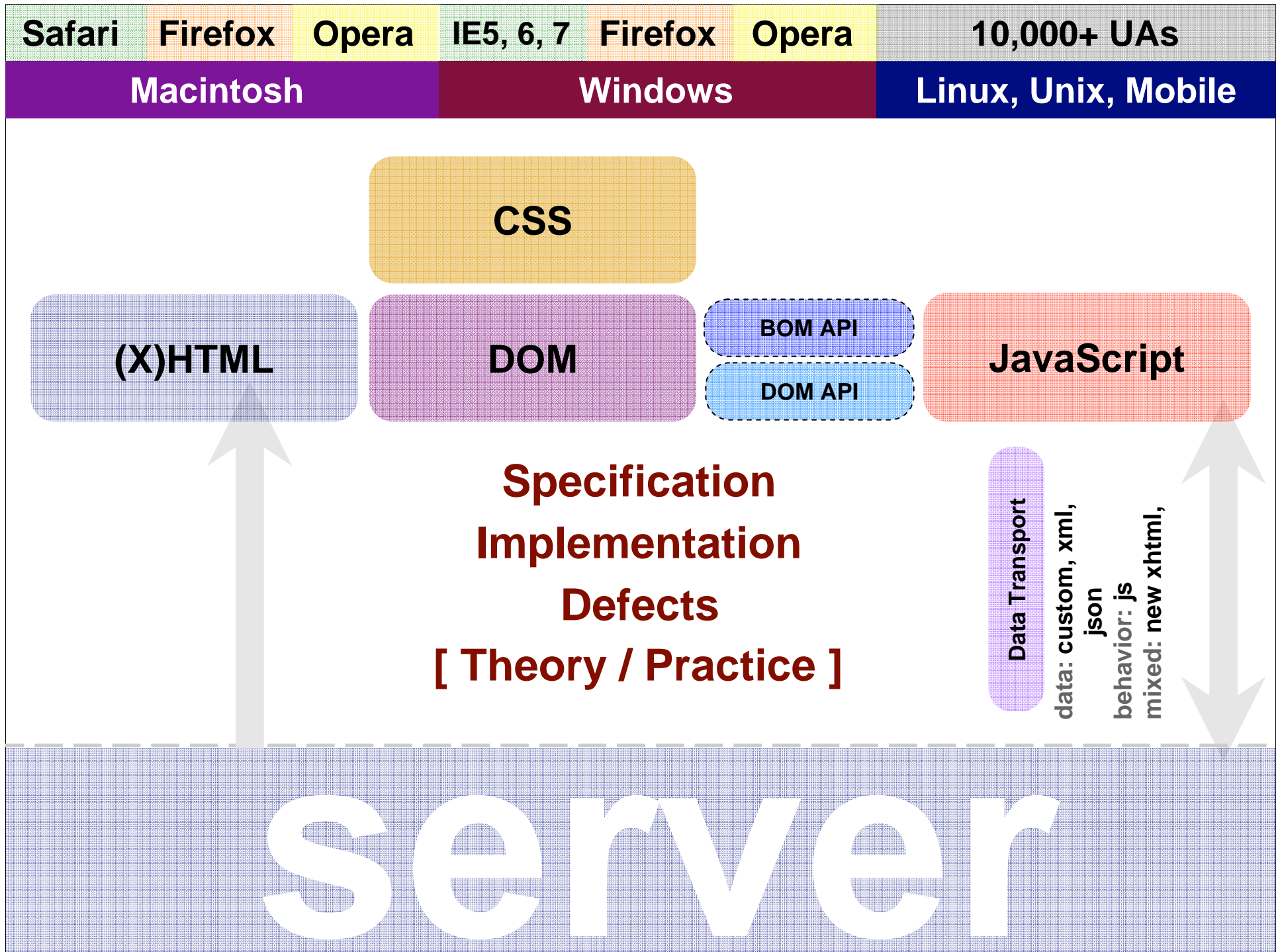




it takes 2 things



We must  
**improve our technology**



Safari

Firefox

Opera

IE5, 6, 7

Firefox

Opera

10,000+ UAs

Macintosh

Windows

Linux, Unix, Mobile

knowledge areas: 7

dimensions: x 4

platforms: x 3

browsers per platform: x 4

rendering modes: x 2

ser=672



(we need a robust platform!)



and  
**raise expectations**



(so users engage and  
explore with confidence!)



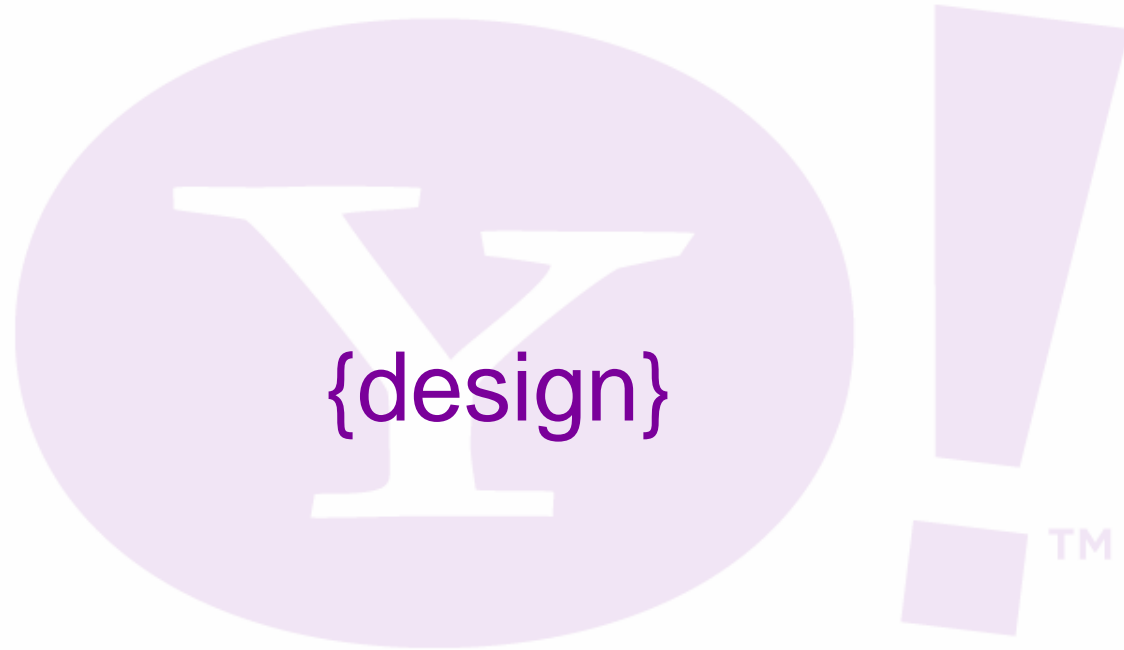


# DESIGN

# DEVELOPMENT

<http://flickr.com/photos/nizboz/81087641/>

from **Implementation  
Models**



to **Mental Models**

from **Heterogeneous  
Environments**



to **Compelling and  
Consistent APIs**





# Nuts and Bolts

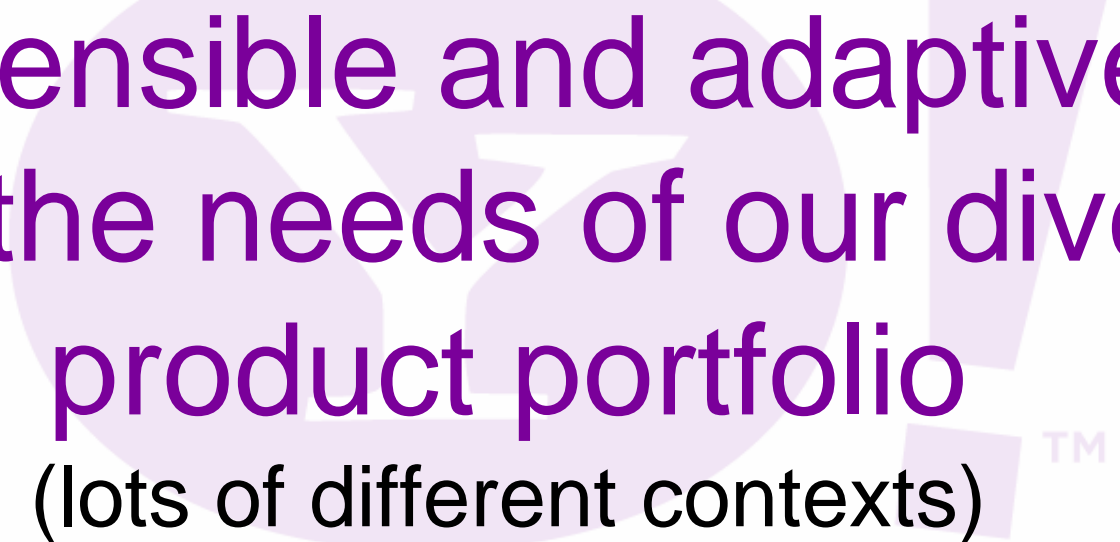
<http://flickr.com/photos/snood/129758197/>



# Six commitments to the platform™

1] Meet properties where they  
are today. Facilitate incremental  
enhancement  
(“Transitional Internet Applications”)

2] Provide a broad, inclusive platform for Web 2.0's "Rich Internet Application" development  
(a la carte, not framework)



3] Extensible and adaptive, to  
meet the needs of our diverse  
product portfolio  
(lots of different contexts)

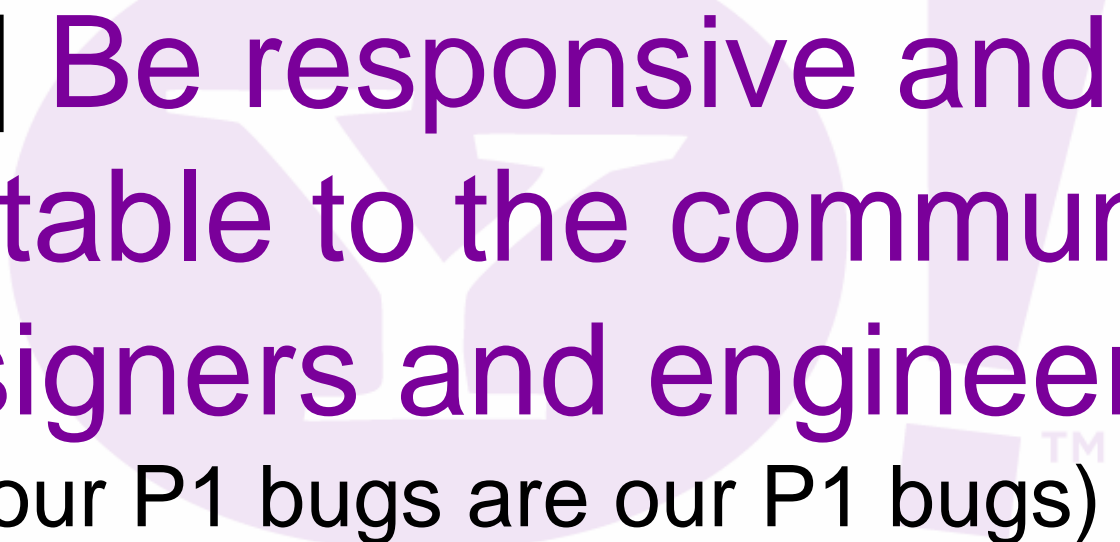


4] Support all A-grade browsers  
– an evolving and still  
demanding challenge  
(Graded Browser Support)

<http://developer.yahoo.com/yui/articles/gbs/gbs.html>



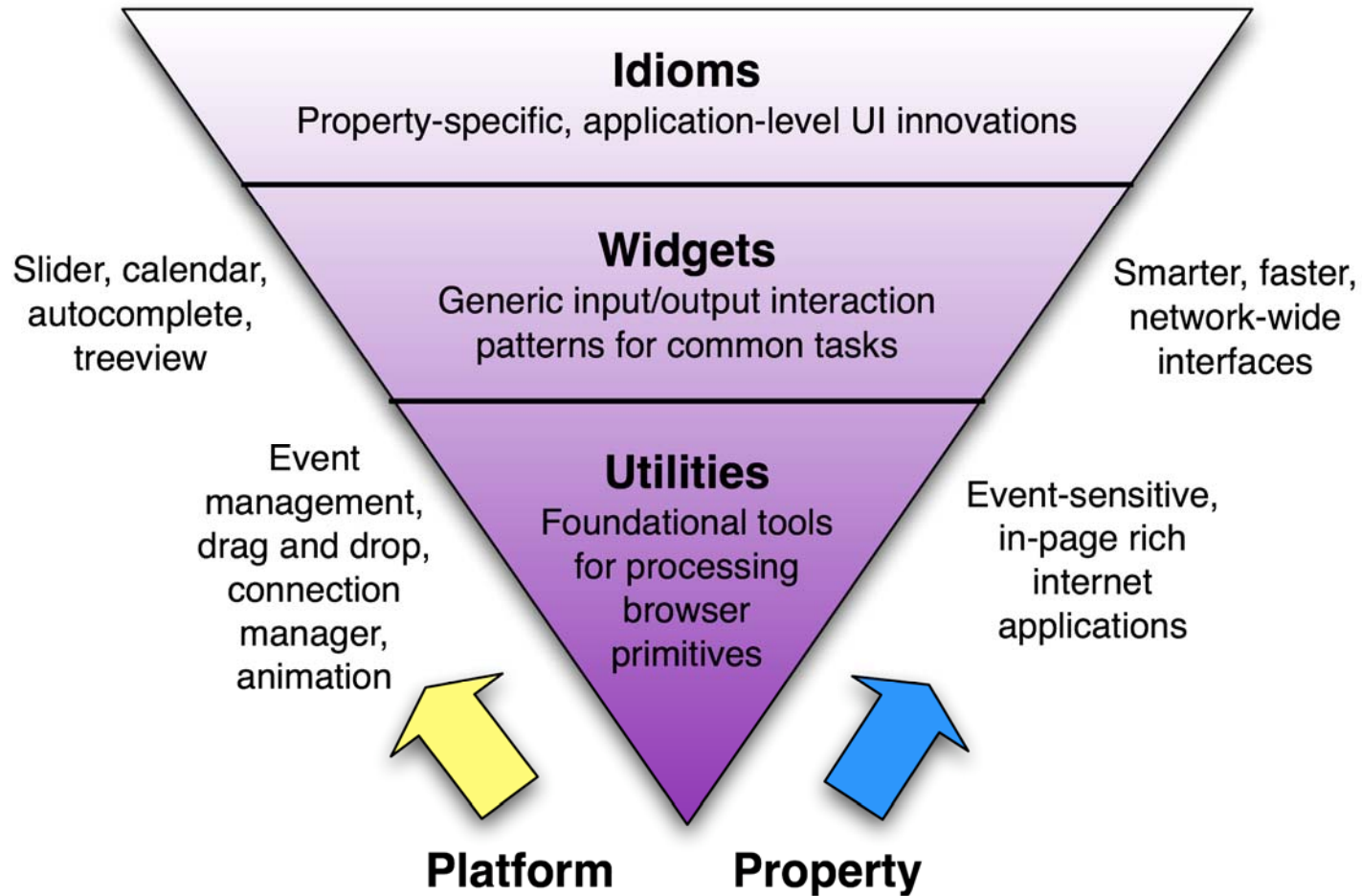
# 5] Support our scale and scope (industrial grade)



6] Be responsive and  
accountable to the community of  
designers and engineers  
(your P1 bugs are our P1 bugs)



# Learn from Desktop History



Adapted from Alan Cooper's "About Face 2.0" Book

# The Yahoo! User Interface (YUI) Library

TreeView

AutoComplete

Slider

Calendar Control

TabView

Menu Control

Logger Control

DHTML Windowing

Animation

Drag & Drop

Connection  
Manager (Ajax)

Event Utility

Dom Collection

Loader

CSS Reset, Fonts, Grids



# YUI DOM Collection

---

- isAncestor(parent, potentialChild)
- inDocument(el)
- getDocumentHeight, getDocumentWidth
- getViewportHeight, getViewportWidth
- getElementsBy
  - `YAHOO.util.Dom.getElementsBy(function(el) { return (/^http:\/\/www\.yahoo\.com\/.test(el.getAttribute('href'))); }, 'a', 'content');`
- Class management utilities

- Event
  - Flexible batch assignment
  - onAvailable and onContentLoaded
  - Scope correction and assignment
  - Automatic cleanup
  - Custom Events



# YUI Event's Custom Events

---

- `var myEvent = new YAHOO.util.CustomEvent( 'myEvent' );`
- `myEvent.subscribe(function() {  
    alert('event fired'); } );`
- `myEvent.fire();`





# YUI Animation

---

```
var anim = new YAHOO.util.Anim(el, {  
    width: {to: 400},  
    height: {by: 400, unit: 'em' },  
    opacity: {from: 0, to: 1}  
    1);  
anim.animate();
```

- Includes support for Bezier math control

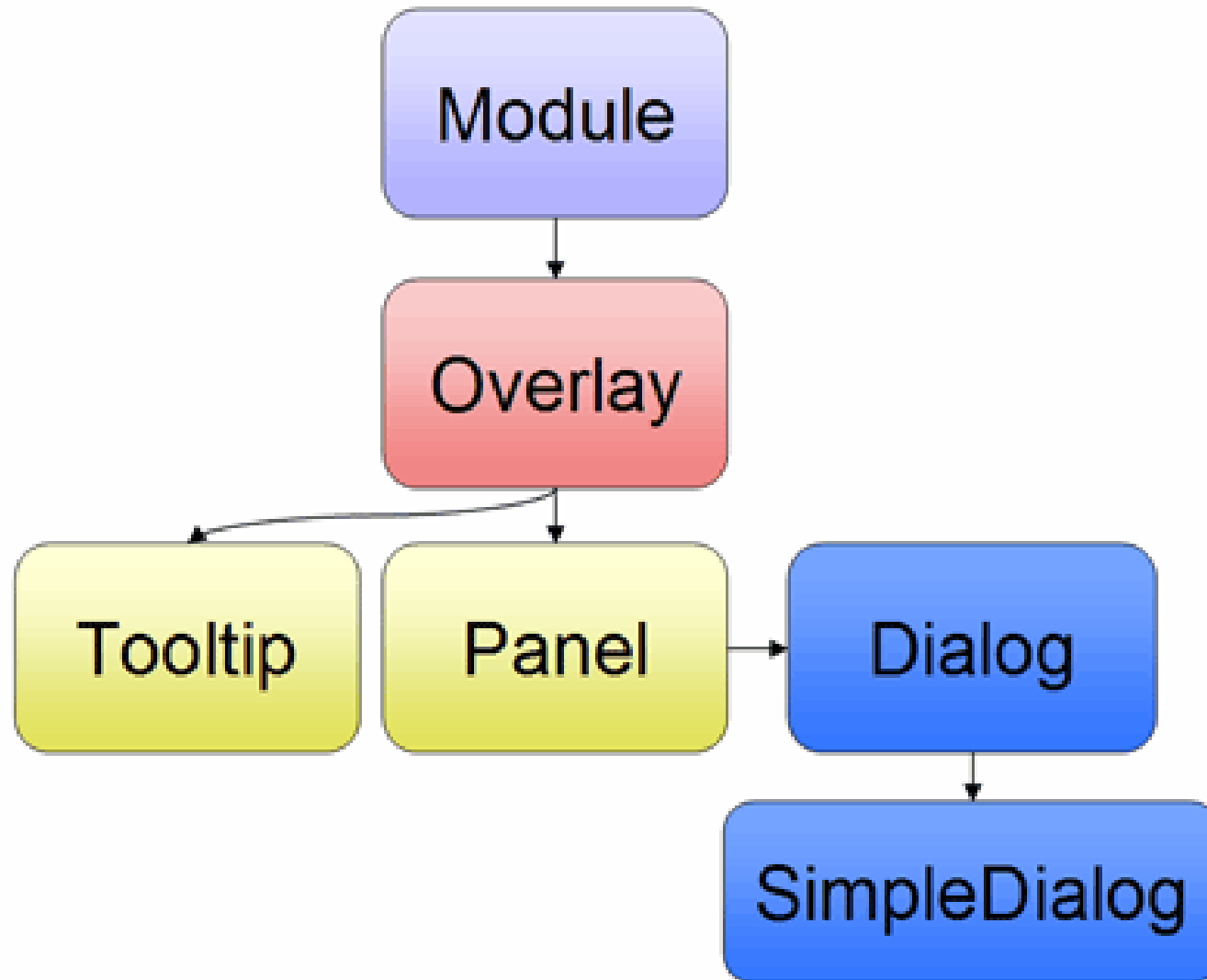


- **File uploading** in addition to XHR
  - Easy implementation of file uploading across the A-Grade browsers
  - Use the upload member of the callback object (instead of "success" or "failure")
- XHR callback with extra arguments



# YUI Container Family

---





## Drag and Drop

---

- It's easy, sorta, but easy to get wrong too
- Point, Region, Multiple Handles, !Handles
- Exposes 15 *interesting moments*

# Drag and Drop Interaction Storyb

Property:

ID:

	Page Load	Mouse Hover	Mouse Down	Drag Initiated	Drag Leaves Original Location	Drag Re-Enters Original Location	Drag Enters Valid Target	Drag Exits Valid Target	Drag Enters Specific Invalid Target	D Inv
Page	drag invitation									
Cursor	normal	dragability grabbable area	selected						drop will be invalid	
Tool Tip		dragability grabbable area								
Drag Object	normal	dragability grabbable area	selected					invalid or drag has exited to target.	drop will be invalid	drop is invalid, what he proceeds and the
Drag Object's Parent Container	normal	dragability grabbable area	selected	dragging	dragging	dragging				draggin
Drop Target	normal		drop invitation	drop invitation; possibly starts in neutral state (not valid/invalid) while drag is positioned over drag object's original location	drop will lose its neutral state, if one has been assigned, becoming explicitly valid or invalid	drop will be invalid or neutral; a distinct treatment for original location may be needed so that visual messaging doesn't signal that the drop's origin is not valid	drop will be valid	any highlighting of target should be removed or an invited off to reflect that the drag's focus has left this target.	drop will be invalid	any time this is sp (e.g. at why the should animate
	What does the page contain to indicate drag and drop?	What happens when the mouse hovers over the draggable object?	What happens when the mouse is pressed on the draggable object but dragging has not initiated?	What happens when drag starts?	What happens when the drag leaves its original location?	What happens when the drag object, having been dragged away from its original location, is dragged back to that location? (E.g., a neutral state can resume, or the original location can be treated as a valid or invalid drop target.)	What happens when I drag over a valid drop target?	What happens when the drag leaves a valid target? (E.g., that target is highlighting turns off or animates off.)	What happens when I drag over an invalid drop target?	What happens when I drag over an invalid drop target?

What happens when the mouse is pressed on the draggable object but dragging has not initiated?



# YUI Logger and Firebug

---

- `alert("stop the insanity");`
- `YAHOO.log("There IS a better way");`
- Debug with Logger or Firebug directly.



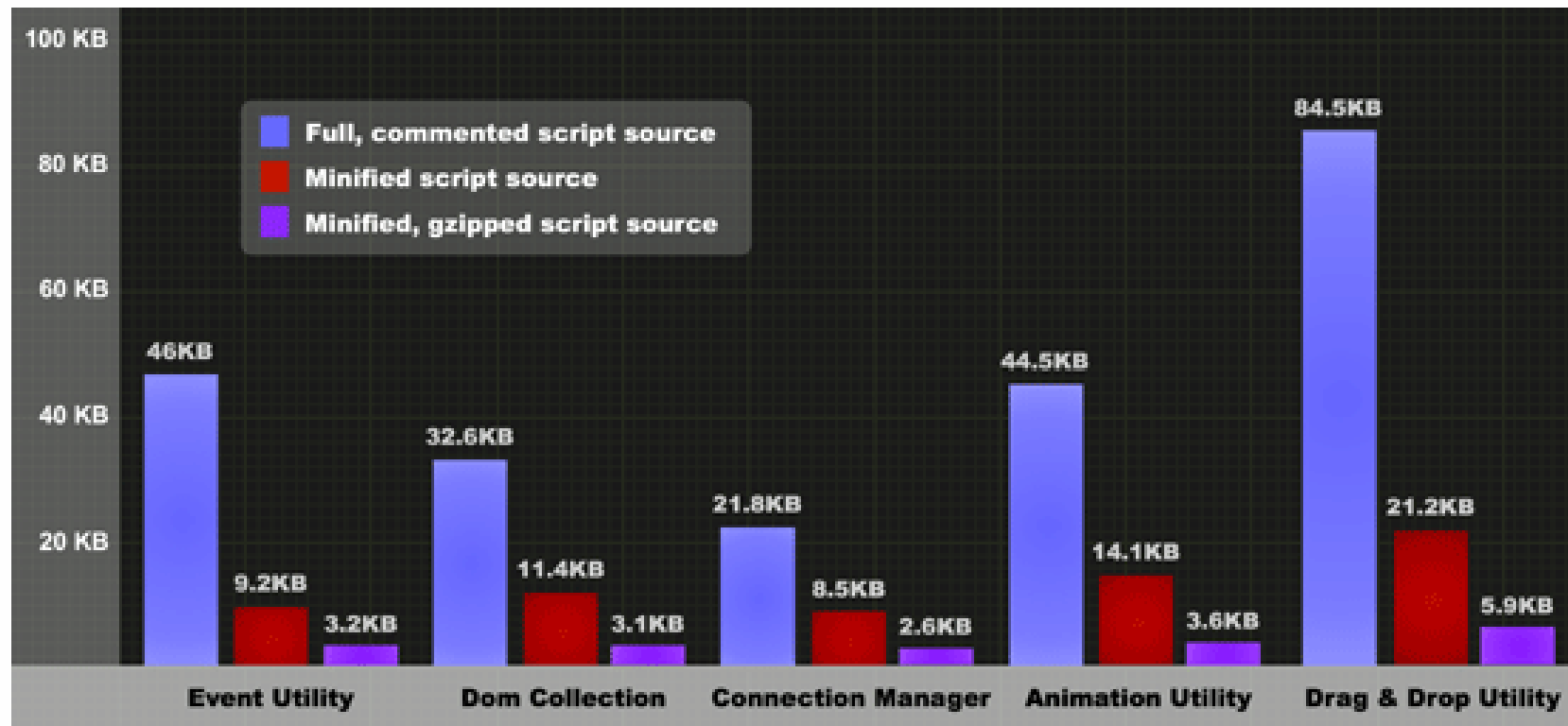
# CSS Grids

---

- Page Widths
- Template Presets
- Nesting Grids
- Together, 200 layouts for 2kb.



# About YUI Page Weight

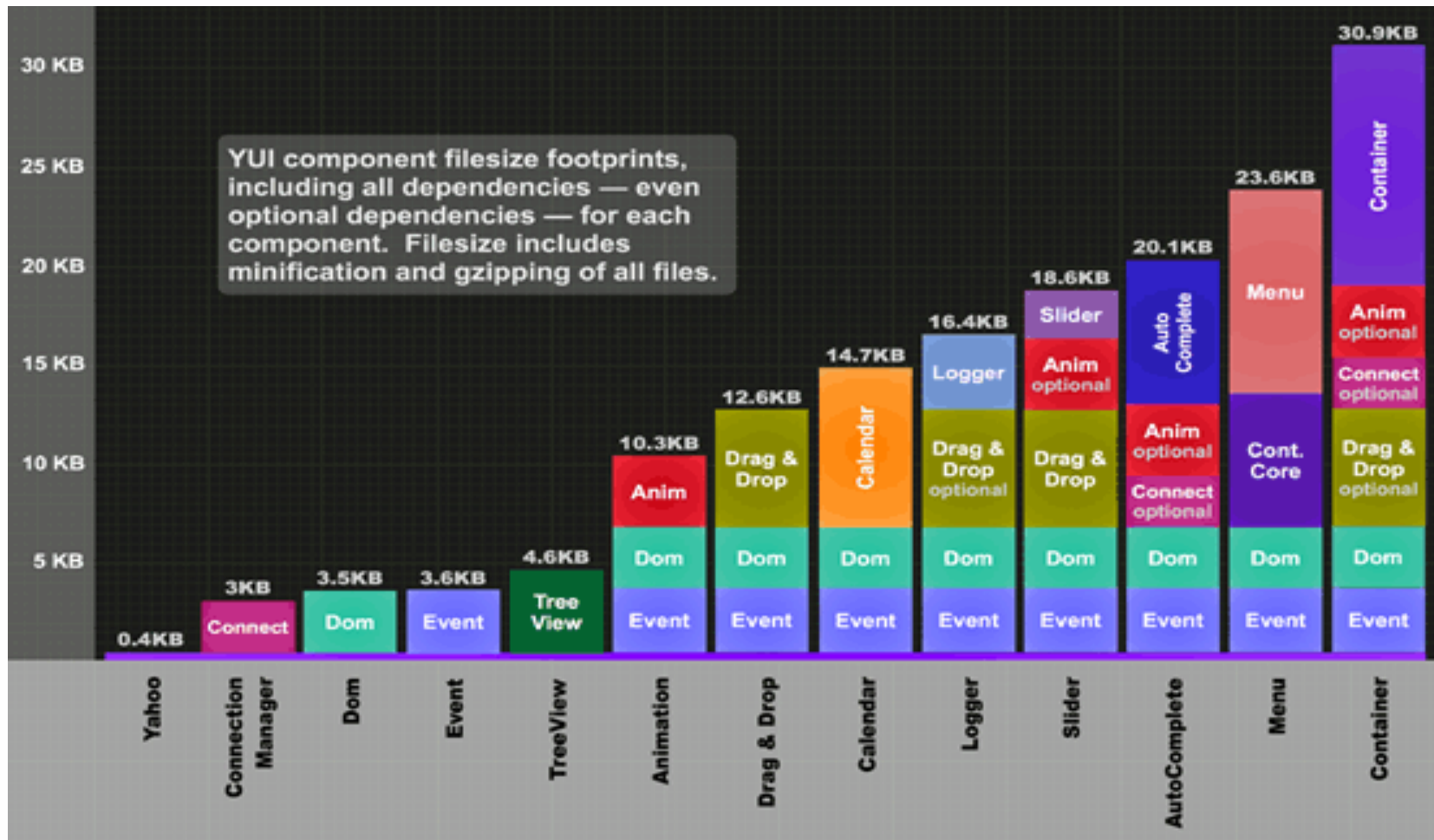


<http://yuiblog.com/blog/2006/10/16/pageweight-yui0114/>



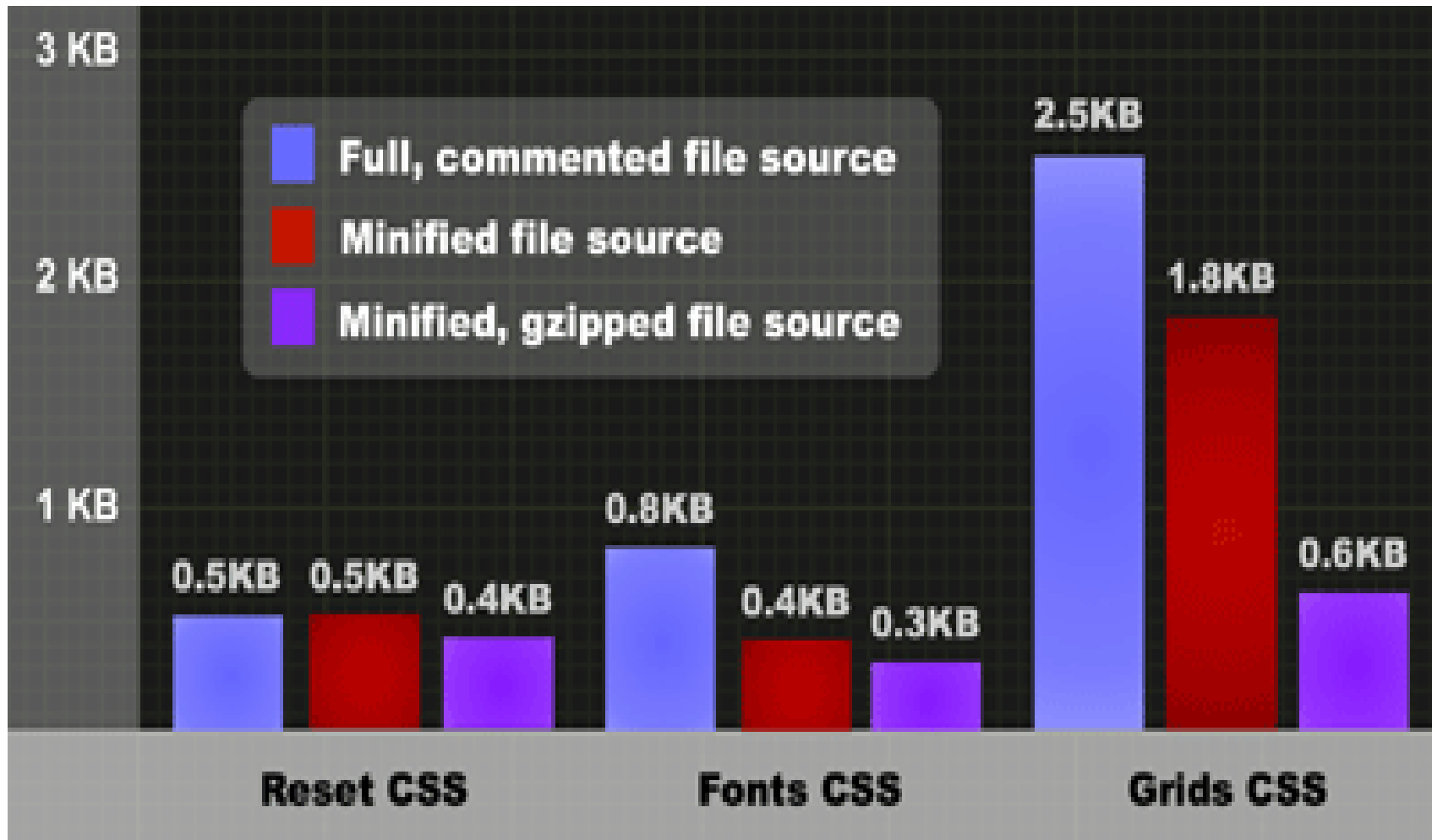


# À la carte Dependencies





# Lightweight CSS Foundation



# Why Open Source?





# What about the timeline?

---

- 2005, January
  - Initial development
  - One-person team
- 2005, July
  - Initial internal release
- 2006, February 13<sup>th</sup>
  - Open-source release
  - #1 on Delicious, Digg, Techmeme
  - 36,000 downloads
- 2006, September 18<sup>th</sup>
  - 3000<sup>th</sup> member on [ydn-javascript](#)
- 2006, November 7<sup>th</sup>
  - Linux Journal Editor's Choice Award: [Best Software Library](#)
  - 300k downloads
- 2006, November 13<sup>th</sup>
  - Current release
  - 9 person team



**What about the  
namespace?**

(because I really like \$)



# Global Variables are Evil

---


- Ideally, only a single global per app|lib|widget
  - Speed unaffected; self documentation; reliable
  - Shorten locally if you want.
- 
- `var $ = YAHOO.util.Dom.get;`
- 
- <http://yuiblog.com/blog/2006/06/01/global-domination/>



# What's in the distribution?

---

- 100s of files
  - All examples
  - All documentation
  - All development builds
  - All production builds
  - All minimized builds



**Yahoo! is powered by  
the exact same bits we  
offer to you.**



# What is the license?

The most-open there is: BSD



**Who's Using It?**



# External Implementations

---

- Wall Street Journal
- Technorati
- IndyCar.com
- SugarCRM
- SmugMug
- PayPal
- eBay
- Yuriz
- PowerReviews.com
- Madonet
- You?



**Documentation?**



# Landing Page Guides

## Maximum Results Displayed

When query results return, an HTML unordered list element (<ul>) of up to ten items (<li> elements) are rendered into the container that you specify. If fewer than ten results are returned, the container's <ul> element will only have that number of <li> elements. Keep in mind that the height, width, and other style characteristics of the container are determined by CSS. You may use the following code to change the maximum number of <li> elements that may populate the <ul> element:

[view plain](#) | [print](#) | ?

```
1 // Display up to 20 results in the container
2 myAutoComp.maxResultsDisplayed = 20;
```

## Minimum Query Length

By default, as soon as a user starts typing characters into the input element, the AutoComplete control starts batching characters for a query to the DataSource. You may increase how many characters the user must type before triggering this batching, which can help reduce load on a server, especially if the first few characters of the input string will not produce meaningful query results. A value of 0 will enable null or empty string queries, which is particularly useful in conjunction with the Always Show Container feature. A negative number value will effectively turn off the widget.

[view plain](#) | [print](#) | ?

```
1 // Require user to type at least 3 characters before triggering a query
2 myAutoComp.minQueryLength = 3;
```

## Query Delay



# Complete API Documentation

## Modules

[animation](#)  
[autocomplete](#)  
[calendar](#)  
[connection](#)  
[container](#)  
[dom](#)  
[dragdrop](#)  
[event](#)  
[logger](#)  
[menu](#)  
[slider](#)  
[tabview](#)  
[treeview](#)  
[yahoo](#)

## Classes

[YAHOO.widget.AutoComplete](#)  
[YAHOO.widget.DataSource](#)  
[YAHOO.widget.DS\\_JsArray](#)  
[YAHOO.widget.DS\\_JsFunction](#)  
[YAHOO.widget.DS\\_XHR](#)

## Files

[AutoComplete.js](#)  
[DataSource.js](#)

## Class YAHOO.widget.AutoComplete

The AutoComplete class provides the customizable functionality of a plug-and-play DHTML auto completion widget. Some key features:

- Navigate with up/down arrow keys and/or mouse to pick a selection
- The drop down container can "roll down" or "fly out" via configurable animation
- UI look-and-feel customizable through CSS, including container attributes, borders, position, fonts, etc

## Constructor

**YAHOO.widget.AutoComplete** ( `elInput` , `elContainer` , `oDataSource` , `oConfigs` )

### Parameters:

`elInput` <HTMLElement> DOM element reference of an input field.  
`elInput` <String> String ID of an input field.  
`elContainer` <HTMLElement> DOM element reference of an existing DIV.  
`elContainer` <String> String ID of an existing DIV.  
`oDataSource` <Object> Instance of YAHOO.widget.DataSource for query/results.  
`oConfigs` <Object> (optional) Object literal of configuration params.

## Properties

`_aListItems` - private Array

Array of <li> elements references that contain query results within the results container.

`_bContainerOpen` - private Boolean

Whether or not the results container is currently open.



# Examples and Tutorials

## YAHOO! USER INTERFACE LIBRARY

### YUI Container: PhotoBox: Subclassing Panel

#### Subclassing Panel to Create PhotoBox

Container classes can be subclassed to create all kinds of rich custom controls. The PhotoBox, which we will build in this tutorial, is an example of how Panel can be subclassed and styled to create a basic popup photo viewer with back and forward navigational controls.

The first step to subclassing the Panel is writing the constructor for the new subclass (PhotoBox, in this case) and specifying its inheritance from the Panel class using `YAHOO.extend`:

[view plain](#) | [print](#) | ?

```
1 // BEGIN PHOTOBX SUBCLASS //
2 YAHOO.widget.PhotoBox = function(e1, userConfig) {
3     if (arguments.length > 0) {
4         YAHOO.widget.PhotoBox.superclass.constructor.call(this, e1, userConfig);
5     }
6 }
7
8 // Inherit from YAHOO.widget.Panel
9 YAHOO.extend(YAHOO.widget.PhotoBox, YAHOO.widget.Panel);
```

☒ [CO](#) Linked to my nickname as [natekoechley](#) Tags:

Next, we will define a few constants for use by the PhotoBox class: "CSS\_PHOTOBX", which defines the CSS class to apply to the Panel, and "NAV\_FOOTER\_HTML", the HTML that will be used for the footer navigation.

[view plain](#) | [print](#) | ?

```
1 // Define the CSS class for the PhotoBox
2 YAHOO.widget.PhotoBox.CSS_PHOTOBX = "photobox";
```

#### YUI Container: Tutorials

[Module: Quickstart](#)

[Overlay: Quickstart](#)

[Tooltip: Quickstart](#)

[Tooltip: One Tooltip, Many Elements](#)

[Panel: Quickstart](#)

[Panel: Advanced Skinning using CSS](#)

[Panel: Creating a 'Loading' Popup](#)

#### PhotoBox: Subclassing Panel

##### Subclassing Panel to Create PhotoBox

##### Functional Example

[ResizePanel: Creating a Resizable Panel](#)

[Dialog Quickstart](#)

[SimpleDialog Quickstart](#)

[Using ContainerEffect](#)

[Using OverlayManager](#)

[Using KeyListener](#)







# Team Blog

```
YAHOO.util.Motion = function(el, attributes, duration, method) {  
    if (el) {  
        this.initMotion(el, attributes, duration, method);  
    }  
};  
  
YAHOO.util.Motion.prototype = new YAHOO.util.Anim();  
YAHOO.util.Motion.prototype.initMotion = function(el, attributes, duration, method) {  
    YAHOO.util.Anim.call(this, el, attributes, duration, method);  
};
```

## YAHOO! USER INTERFACE BLOG

News and Articles about Designing and Developing with Yahoo! Libraries.



Blog

About

### Video: Douglas Crockford, "An Inconvenient API: The Theory of the Dom"

October 20, 2006 at 7:42 am by Eric Miraglia | In [Development](#) |

**Update 20 October 2006:** This video was originally posted with no link to the slide deck that Douglas uses in the talk. That deck is in PowerPoint format and [can be downloaded here](#).

Douglas Crockford is Yahoo!'s leading JavaScript Architect. He has written extensively on JavaScript and has been among the protagonists of the JavaScript developer community for more than a decade. Douglas is the discoverer of the JSON data format and a frequent contributor to YUIBlog.

At Yahoo!, Douglas has been teaching a series of classes on the JavaScript programming language. These classes bear his unique style and perspective and benefit from the depth of

#### SYNDICATE

All Entries:

[+ MY YAHOO!](#) [RSS](#)

All Comments:

[+ MY YAHOO!](#) [RSS](#)

#### RECENT POSTS

[Graded Browser Support: Q4 Update](#)

[YUI version 0.12.0: TabView Control, Improved API Documentation, and More](#)

[JavaScript, We Hardly new Ya](#)

[Design Pattern Conversation: What's the Best Way to Communicate Patterns? Part Five.](#)



## Is there a community?

---


- <http://yuiblog.com>
- <http://groups.yahoo.com/group/ydn-javascript>
- <http://www.jackslocum.com/yui/> YUI.ext
- <http://blog.davglass.com/> YUI.addons
- SourceForge



# Jack Slocum

## Jack Slocum's Blog

Using the Yahoo! UI library - Beyond the Examples



Alloys-Precision stamping  
machining and assembly in low to  
medium volumes  
[www.Hydroblanking.com](http://www.Hydroblanking.com)

Feedback - Ads by Google

Navigation

When I first started looking at the Yahoo! UI library, one of the things I noticed was a lack of real world examples. Sure, the examples make little boxes fly and fade, make them drag and drop and other cool effects but how do you take that and put it into your website or application? This blog is going to be dedicated to demonstrating how to do that.

While writing this blog a small library of reusable components based on YUI has been started. The library was namespaced YAHOO.ext (short for Yahoo! UI extensions), and already has a few very useful classes that make day to day development with YUI much easier. All of the classes are well-documented and there are some examples of using them found in the posts on this blog. The code is of course free and has the same unrestrictive (BSD) license as Yahoo! UI.

Jack

### "Build your own" Download Dialog and RC1 Release


November 11, 2006 by Jack Slocum | 33 Comments »

This post is to notify everyone the .33 RC1 build is up. This release is primarily bug fixes. There's also a new download dialog with a "Build your own" feature that I think demonstrates how easy it is to rapidly build something with YUI + yui-ext. The download dialog consists of a modal BasicDialog with a BorderLayout embedded. It uses a YAHOO.ext.UpdateManager with a JSON renderer to build the list of options. It's ... [more »](#)

#### Download YAHOO.ext

- Downloads
- Online Documentation
- Report Bugs
- Help Forum

#### Help support YAHOO.ext



Need a custom component? Need help implementing YUI or YAHOO.ext? I can help. [Send me an email](#) for details.

Ads by Google

#### CodeThatGrid JavaScript

Web grid: preview, edit, post data right in the browser window  
[www.codethat.com](http://www.codethat.com)

#### Zimbra Web 2.0 Email

AJAX email server, calendar contacts, outlook



# How is it different from GWT?

We believe in JavaScript.™



Thanks!

[nate.koechley.com/talks/2006/11/refresh06/YUI/](http://nate.koechley.com/talks/2006/11/refresh06/YUI/)

---



- Nate Koechley:
  - [nate@koechley.com](mailto:nate@koechley.com) | [natek@yahoo-inc.com](mailto:natek@yahoo-inc.com)
  - <http://nate.koechley.com/blog>
- Yahoo! Developer Network & Y! UI Blog:
  - <http://developer.yahoo.com>
  - <http://developer.yahoo.com/yui>
  - <http://developer.yahoo.com/ypatterns>
  - <http://www.yuiblog.com>
  - <http://groups.yahoo.com/group/ydn-javascript>



# We're Hiring!

---

Josie Aguada: [jaguada@yahoo-inc.com](mailto:jaguada@yahoo-inc.com)

Usual suspects:

JavaScript, PHP, CSS, HTML, ActionScript...