



Building Extensible Widgets

Scott González



Scott González

jQuery UI development lead

<http://nemikor.com>

@scott_gonzalez



Building Extensible Widgets



Plugins

Building Extensible ~~Widgets~~



~~Building Extensible Widgets~~ ~~Plugins~~

Writing Extensible Code



What is extensible code
and why do we need it?



API Design: Autocomplete



API Design: Autocomplete

Data source (URL)



API Design: Autocomplete

Data source (URL)

Min chars

Max items



API Design: Autocomplete

Data source (URL)

Min chars

Max items

Delay

Caching



API Design: Autocomplete

Data source (URL/local)

Min chars Case sensitive

Max items Begins/contains

Delay

Caching



API Design: Autocomplete

Data source (URL/local)

Min chars Case sensitive

Max items Begins/contains

Delay Max height/scroll

Caching



API Design: Autocomplete

Data source (URL/local)

Min chars Case sensitive

Max items Begins/contains

Delay Max height/scroll

Caching Highlighting

Custom formatting



API Design: Autocomplete

Data source (URL/local)

Min chars Case sensitive

Max items Begins/contains

Delay Max height/scroll

Caching Highlighting

Extra filters Custom formatting



API Design: Autocomplete

Data source (URL/local)		Multiple
Min chars	Case sensitive	Separator
Max items	Begins/contains	
Delay	Max height/scroll	
Caching	Highlighting	
Extra filters	Custom formatting	



Is this extensible code?

Data source (URL/local)	Multiple	
Min chars	Case sensitive	Separator
Max items	Begins/contains	
Delay	Max height/scroll	
Caching	Highlighting	
Extra filters	Custom formatting	



jQuery Autocomplete Plugin

data, autoFill, cacheLength, delay,
extraParams, formatItem, formatMatch,
formatResult, highlight, matchCase,
matchContains, matchSubset, max,
minChars, multiple, multipleSeparator,
mustMatch, scroll, scrollHeight,
selectFirst, width, result()



Is this easy to work with?

data, autoFill, cacheLength, delay,
extraParams, formatItem, formatMatch,
formatResult, highlight, matchCase,
matchContains, matchSubset, max,
minChars, multiple, multipleSeparator,
mustMatch, scroll, scrollHeight,
selectFirst, width, result()



Creating Clean APIs



Make your primary use case
as clean and simple as possible



Stay focused



Don't try to solve
every possible problem



Adding extension points



Events



```
$( "#loading" )  
  .bind( "ajaxStart", function() {  
    $( this ).show();  
  })  
  .bind( "ajaxStop", function() {  
    $( this ).hide();  
  });
```



```
$( "#autocomplete" )  
  .bind( "autocompleteopen" ) {  
    $( this ).autocomplete( "widget" )  
      .hide()  
      .slideDown();  
  });
```



Cancelable events



```
$( window )  
  .bind( "beforeunload", function() {  
    if ( contentModified ) {  
      return false;  
    }  
  });
```



```
$( "#autocomplete" )  
  .bind( "autocomplete",  
  function( event ) {  
    if ( !/^d/.test( this.value ) ) {  
      event.preventDefault();  
    }  
  }  
);
```



Callbacks



```
$.ajax({  
  url: "/foo/bar",  
  beforeSend: function( xhr ) {  
    xhr.setRequestHeader(  
      "X-Custom", "jQuery" );  
  }  
});
```



```
$( "#autocomplete" ).autocomplete({  
  search: function( event, ui ) {  
    if ( !/^d/.test( this.value ) ) {  
      event.preventDefault();  
    }  
  }  
});
```




Maps



```
$.attrFn.awfulEvent = true;
```

```
$( "<div>", {  
  html: "extensible code",  
  awfulEvent: function( event ) {  
    event.preventDefault();  
  }  
});
```



```
$.fx.speeds.xslow = 10000;
```

```
$( "#msg" ).fadeOut( "xslow" );
```



```
$.fx.step.color = function( fx ) {  
    // handle color animation  
};
```

```
$( "#elem" ).animate({  
    color: "#f00"  
});
```



Other examples



```
$( "#elem" ).offset({  
    top: 100,  
    left: 100,  
    using: function( pos ) {  
        $( this ).animate( pos );  
    }  
});
```



```
$.validator.addMethod( "age",  
    function( value ) {  
        return value > 17 && value < 120;  
    }, "Please enter a valid age." );
```



Keep you API simple



```
<div id="foo">foo</div>  
<div id="bar">bar</div>  
<div id="baz">baz</div>
```

```
var foo = $( "#foo" )[0];  
var baz = $( "#baz" )[0];  
foo.insertBefore( baz );
```



(clean + clear) > (cool + compressed)
// more importantly
fun > fast

- Aaron Quint



Prototype and Iterate



jQuery UI Autocomplete

Options:

delay, minLength, position, source

Events:

search, open, focus, select, close, change



<http://spkr8.com/t/4665>

<http://nemikor.com>

Coupon code: AUTHD

40% off print, 50% off ebook

