

# jQuery-free JavaScript

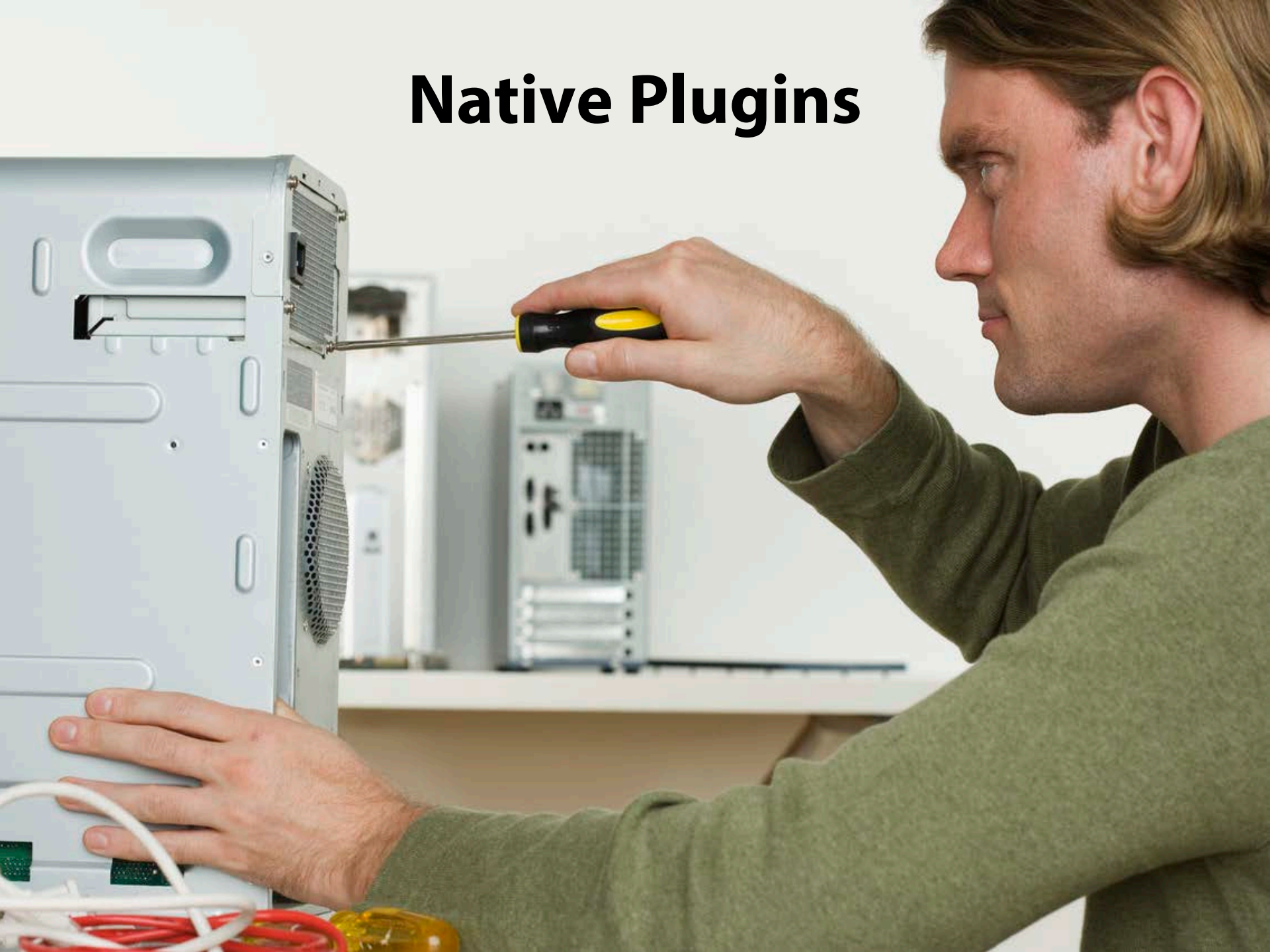
Native Plugins

Elijah Manor  
<http://elijahmanor.com>  
[@elijahmanor](#)



**pluralsight**   
hardcore developer training

# Native Plugins



# **jQuery Plugins**

- **One of the reasons why the library became so popular**
- **Plugins are easy to create**
- **There is a huge number of plugins in the ecosystem**
- **Plugins range from simple to very complex**
- **However, many of them don't need jQuery...**

# jQuery Plugin as a Namespace

jQuery Loaded

```
$.logger = function(message) {  
    console.log(message);  
};
```

```
$.logger("hello world");
```

# jQuery Plugin as a Namespace

Without jQuery Loaded

```
(function() {  
    var $ = window.jQuery || window.etm || (window.etm = {});  
    $.logger = function(message) {  
        console.log(message);  
    };  
    $.logger("hello world");  
})();  
    etm.logger("hello world");
```

```
(function($, undefined) {  
    $.logger = function(message) {  
        console.log(message);  
    };  
})( window.jQuery || window.etm || (window.etm = {}) );
```

# Simple jQuery Plugin

```
$.fn.tooltip = function(options) { /* ... */
    init(options);
    return this.each(function() { wire(this); });
    function init(options) { /* ... */ }
    function wire(element) {
        $(element).on("mouseenter", function() { /* ... */
            $tooltip.css({
                top: position.top - $tooltip.outerHeight(),
                left: position.left
            }).text($this.attr("title"));
        }).on("mouseleave", function() {
            $tooltip.css({ left: "-99999px" });
        });
    }
};
```

# Convert: Extend

## jQuery

```
options = $.extend({}, $.fn.tooltip.defaults, options);
```

## Native

```
options = options || {};  
tooltip.element.style.backgroundColor =  
    options.backgroundColor || tooltip.defaults.backgroundColor;  
tooltip.element.style.color =  
    options.color || tooltip.defaults.color;
```

## Underscore or Lo-Dash

```
options = _.extend({}, $.fn.tooltip.defaults, options);
```

# Convert: Create

## jQuery

```
$tooltip = $("#tooltip");  
if (!$tooltip.length) {  
    $tooltip = $("<div id='tooltip'>Default  
Text</div>").appendTo("body");  
    $tooltip.css(options);  
}
```

## Native

```
tooltip.element = document.getElementById("tooltip");  
if (!tooltip.element) {  
    tooltip.element = document.createElement("div");  
    tooltip.element.id = "tooltip";  
    tooltip.element.innerText = "Default Text";  
    document.body.appendChild(tooltip.element);  
}
```



# Convert: Event

## jQuery

```
$(element)  
  .on("mouseenter", function() { /* ... */ })  
  .on("mouseleave", function() { /* ... */ });
```

## Native

```
element.addEventListener("mouseenter",  
  function() { /* ... */ });  
element.addEventListener("mouseleave",  
  function() { /* ... */ });
```

# Convert: Position

## jQuery

```
var $this = $(this), position = $(this).position();  
$tooltip  
    .text($this.attr("title"))  
    .css({  
        top: position.top - $tooltip.outerHeight(),  
        left: position.left  
    });
```

## Native

```
tooltip.element.innerText = element.title;  
tooltip.element.style.top = element.offsetTop -  
    tooltip.element.offsetHeight + "px";  
tooltip.element.style.left = element.offsetLeft + "px";
```

# Simple Library

```
(function(tooltip, undefined) {  
  tooltip.element = null;  
  tooltip.init = function(options) { /* ... */ };  
  tooltip.wire = function(element) { /* ... */ };  
  tooltip.defaults = { /* ... */ };  
})(window.tooltip = window.tooltip || {}));
```

```
var nodes = document.querySelectorAll("a");  
tooltip.init({ color: "blue", backgroundColor: "thistle" });  
[].forEach.call(nodes, function(element) {  
  tooltip.wire(element);  
});
```

# jQuery Wrapper

```
(function($, tooltip) {  
    $.fn.tooltip = function(options) {  
        tooltip.init(options);  
        return this.each(function() {  
            tooltip.wire(this);  
        });  
    };  
})(jQuery, tooltip);  
  
$("a").tooltip();
```

# jQuery Plugin Adapter



# DEMO

- **jQuery Namespace**
- **jQuery Plugin**
- **jQuery-free Plugin**
  - **Conversion Process**
    - **Extend**
    - **Create**
    - **Event**
    - **Position**
- **jQuery Plugin Wrapper**

# Summary

- **Some jQuery Plugins Shouldn't be jQuery Plugins**
- **Use another namespace if jQuery isn't loaded**
- **If possible, remove jQuery as a dependency from your plugin and release it as a JavaScript library**
- **Then make a jQuery Plugin Adapter around your**