

Advanced jQuery Events

Joe Eames

@josepheames

www.pluralsight.com



Outline

- `on()` method
- Custom Events
- Deferred Overview
- Deferred Example
- Promises
- Parameters and Late Callbacks
- `when()` method
- Summary

on()

- Replaces bind, delegate, live
- off() to remove

on() Basic Usage

```
$(“selector”).on(“click”, myHandlerFunction);
```

```
$(“selector”).off(“click”);
```

OR

```
$(“selector”).off(“click”, myHandlerFunction);
```

on()

- Replaces bind, delegate, live
- off() to remove
- Supports namespacing of events

on() With Namespaces

```
$(“selector”).on(“click.myNamespace”, handler);  
$(“selector”).on(“click.myOtherNamespace”, handler);  
...  
$(“selector”).off(“click.myNamespace”);
```

on()

- Replaces bind, delegate, live
- off() to remove
- Supports namespacing of events
- Multiple event registration shortcuts

on() Multiple Event Registration

```
$(“selector”).on(“click hover”, handler);
```

```
$(“selector”).on({  
    “click”: clickHandler,  
    “hover”: hoverHandler  
});
```


on()

- Replaces bind, delegate, live
- Use off() to remove
- Supports namespacing of events
- Has multiple event registration shortcuts
- Supports stopping bubbling & propagation
- Can use shortcut for returning false

Shortcut to return false

```
$(“selector”).on(“click”, false);
```

Is equivalent to

```
$(“selector”).on(“click”, function() { return false; });
```

on()

- Replaces bind, delegate, live
- Use off() to remove
- Supports namespacing of events
- Has multiple event registration shortcuts
- Supports stopping bubbling & propagation
- Can use shortcut for returning false
- Replaces live() for future element events

Replacing Live

Usage of Live:

```
$(“tr.dataRow”).live(“click”, myHandler);
```

Replacing with On:

```
$(“#parent”).on(“click”, “tr.dataRow”, myHandler);  
OR  
$(document).on(“click”, “tr.dataRow”, myHandler);
```

Custom Events

```
$("#myDiv").on("nonexistentEvent", handler);
```

```
$("#myDiv").trigger("nonexistentEvent");
```

Custom Events on JavaScript Objects

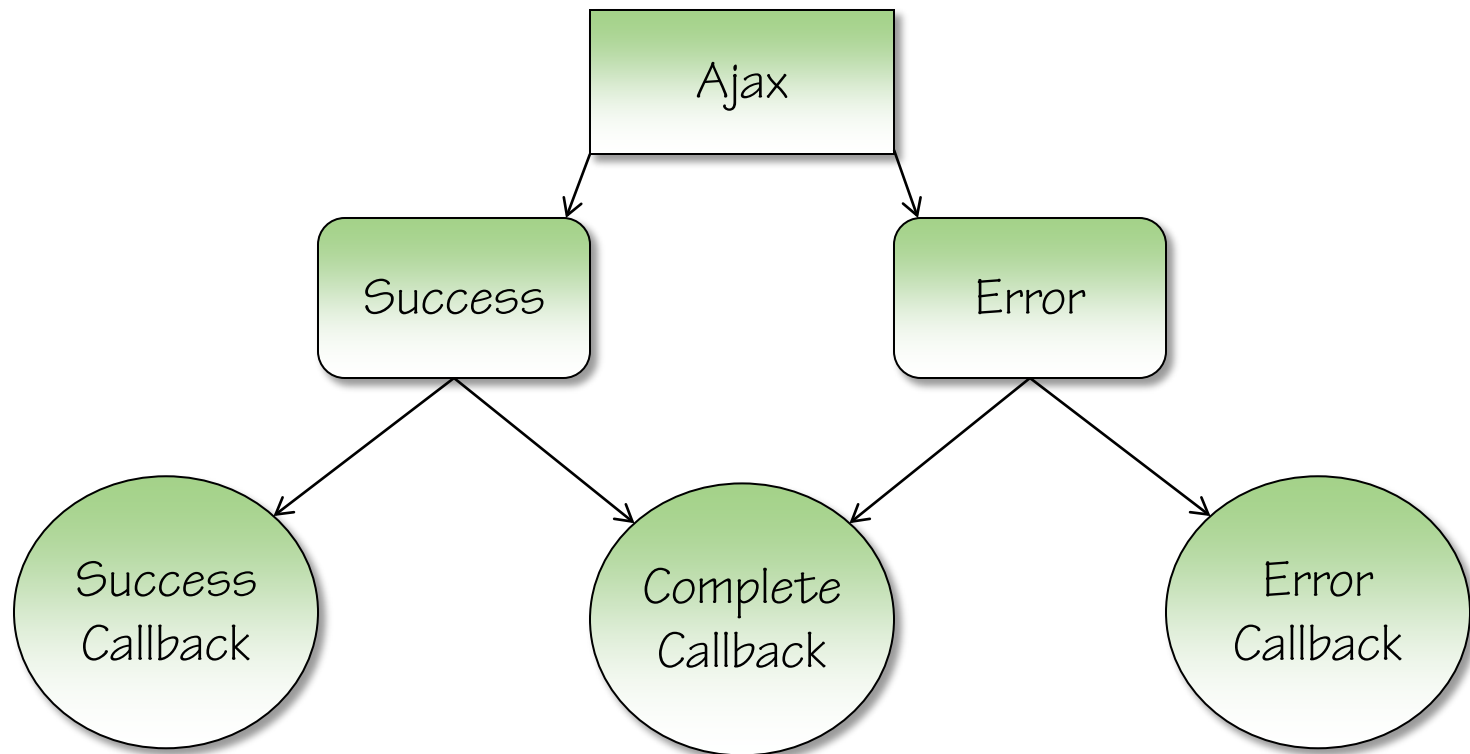
```
var trafficLight = {...}  
var car = {...}
```

```
car.lightTurnedGreen = function() {...}  
$(trafficLight).on("green", car.lightTurnedGreen);  
...  
$(trafficLight).trigger("green");
```

Deferred Objects

- Based on CommonJS promise proposal
- Creates a utility object
- Used mainly for asynchronous operations

Ajax



Ajax Creation

```
$.ajax("/some/url", {  
    success: successHandler,  
    error: errorHandler,  
    complete: completeHandler  
});
```

Ajax Creation

```
function successHandler1() {...}  
function successHandler2() {...}  
function both() {  
    successHandler1();  
    successHandler2();  
}  
  
$.ajax("/some/url", {  
    success: both,  
    error: errorHandler,  
    complete: completeHandler  
});
```

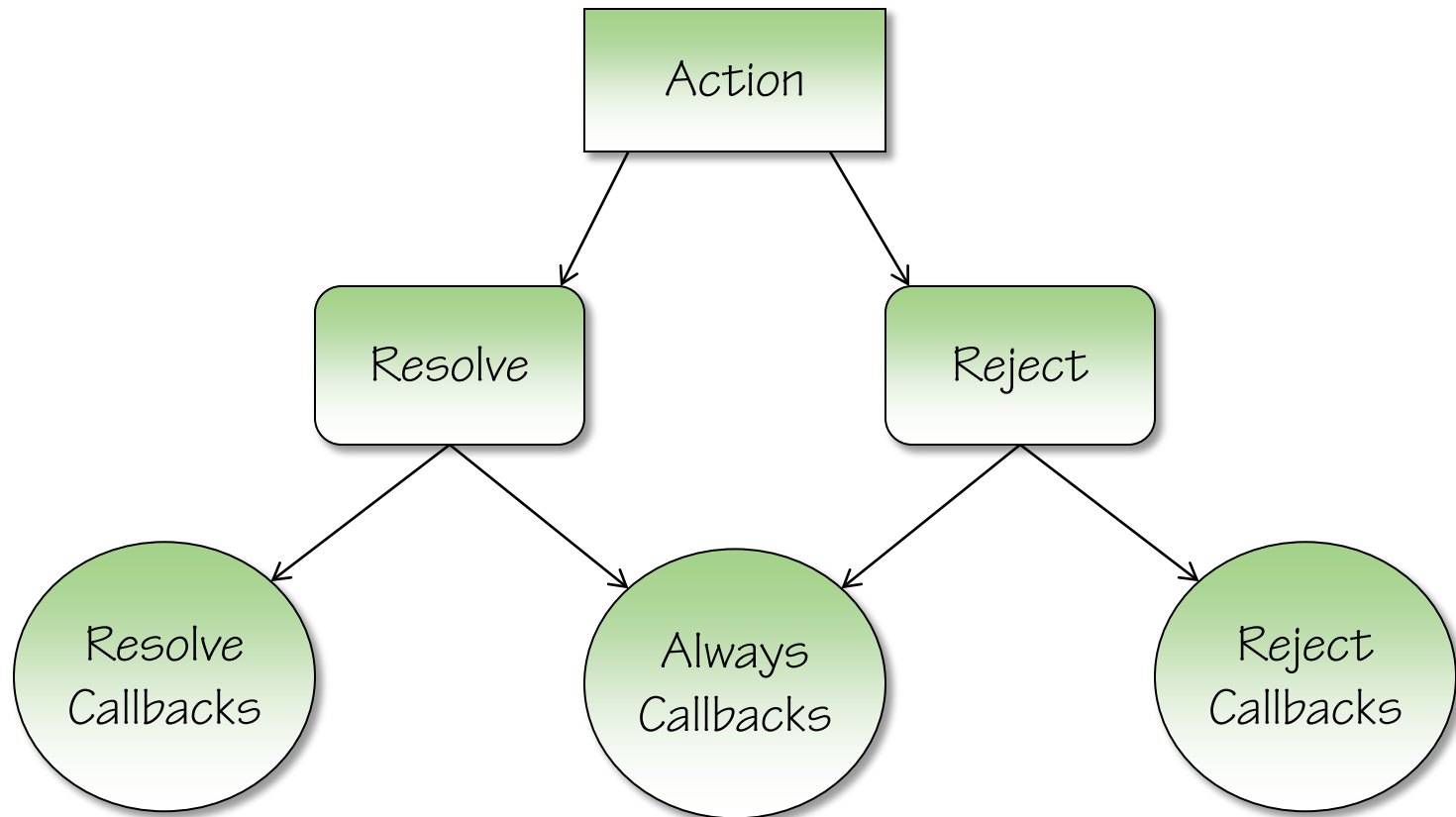
Ajax Creation

```
function successHandler1() {...}
function successHandler2() {...}

var mySuccessHandler;
if(someCondition)
    mySuccessHandler = successHandler1;
else
    mySuccessHandler = successHandler2;

$.ajax("/some/url", {
    success: mySuccessHandler,
    error: errorHandler,
    complete: completeHandler
});
```

Deferred



Deferred

Purpose

- An object for holding & calling a queue of success/failure/complete callbacks

Advantages over typical callback registration

- Allows multiple callback registration
- Allows registration of callbacks at any time
 - During Creation
 - During Execution
 - After Completion

Deferred Example

- Ajax
- SetTimeout
- Animations

\$.when

- Combines 1 or more deferred objects
- Returns a promise, not a deferred object
- Only resolves if all subordinate deferred objects resolve
- Rejects instantly if any subordinate deferred objects reject

Summary

- Events are core to jQuery
- Deferred objects and promises separate callbacks from operations
- Understanding events and how to handle asynchronous code in JavaScript is important