



# Plugins In Your Web App

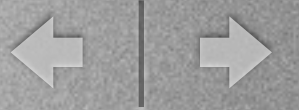
---

a how-to

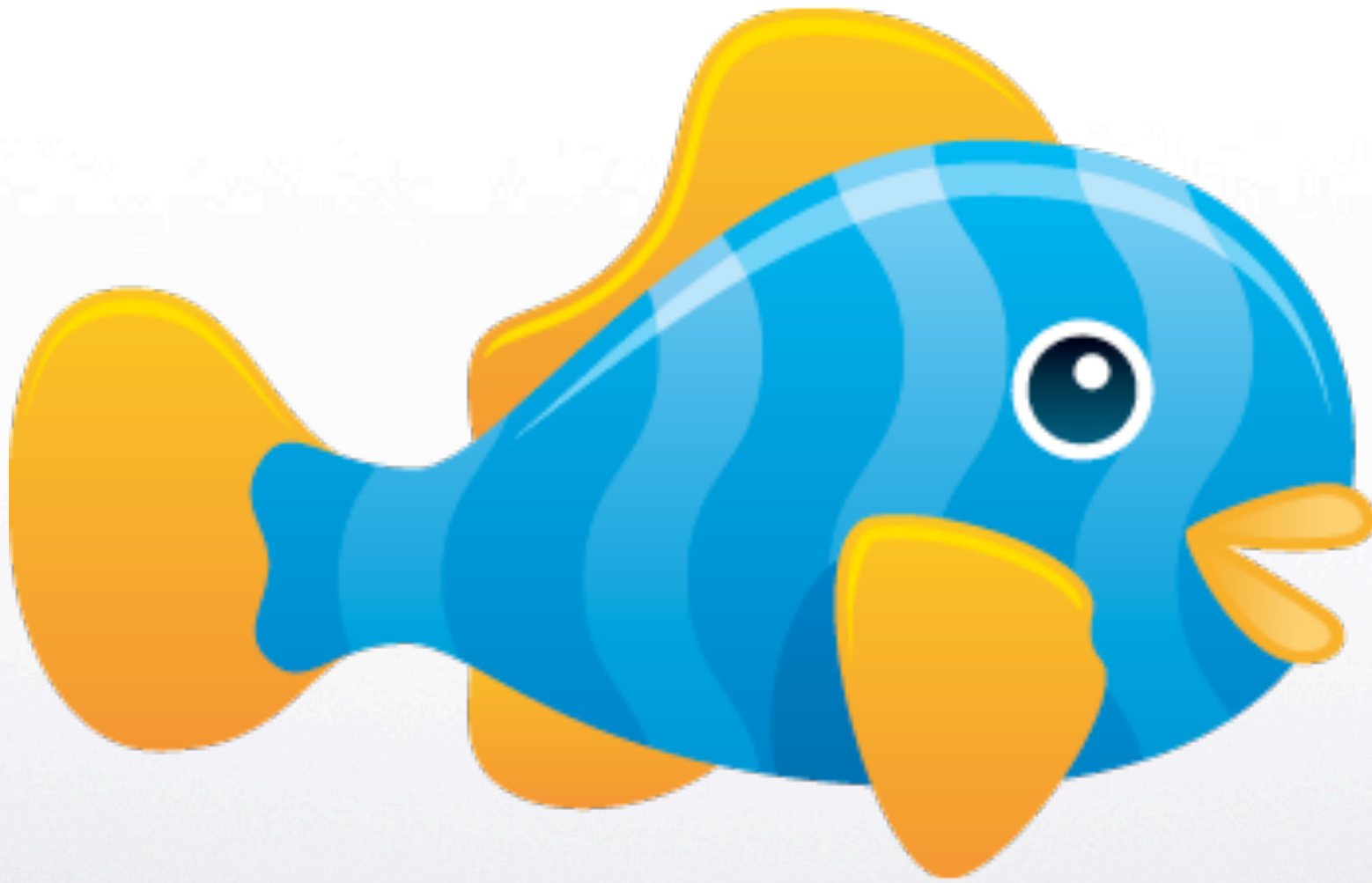


```
var context = {  
  "name": "Gary Haran",  
  "experience": "writing JS since 1996",  
  "oldestProdCode": "2000 IFRAME hack",  
  "currentWork": "http://talkerapp.com",  
  "frameworks": [  
    'jQuery', 'Prototype', 'xapi', 'dynapi'  
  ]  
}
```





# Quick Talker Demo



Not entirely self-serving. It'll give everyone a common ground on which we can discuss real life scenarios where plugins and custom events are used.



# My Purpose

1. Explain the **benefits of plugins** in a web app
2. Bring to light the power of **custom events**
3. Ensure you can add plugins to **your** web app
4. Encourage you to **try Talker** **\*\*Self-serving\*\***





# Benefits of Plugins



- 1) Empower a movement
  - FF 1.0 happened in 2004
  - IE6 was King
  - Popups were everywhere
  - Did anyone here drop Netscape in favor of Internet Explorer?
  - What made you go back to Netscape in its Firefox incarnation?
  - What browser are you using today?
- 2) Really good ideas will rise to the top
- 3) Some people will build stuff you hadn't thought of
- 4) We're not hiding implementation behind a proprietary curtain like a Word file but giving reasons to people to stick with us because we have more value.
- 5) Explain the user's list and how that event doesn't necessarily display in the chat room



# Benefits of Plugins

1. **Empower** a movement
2. Benefit from the dynamics of **meritocracy**
3. Find **new uses** for your application
4. Natural and non-threatening **tie-in**
5. Allows to **simplify** your code

- 1) Empower a movement
  - FF 1.0 happened in 2004
  - IE6 was King
  - Popups were everywhere
  - Did anyone here drop Netscape in favor of Internet Explorer?
  - What made you go back to Netscape in its Firefox incarnation?
  - What browser are you using today?
- 2) Really good ideas will rise to the top
- 3) Some people will build stuff you hadn't thought of
- 4) We're not hiding implementation behind a proprietary curtain like a Word file but giving reasons to people to stick with us because we have more value.
- 5) Explain the user's list and how that event doesn't necessarily display in the chat room





# Native Events

1. Execution chain
2. `return false/event.preventDefault()`

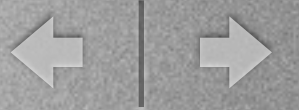


# Custom Events

1. Execution chain
2. onYourOwnEvent
3. When to use custom events?

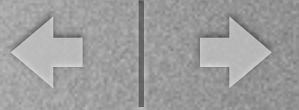
- 1) follow 2 Custom Events folder
- 2) Cmd+Shift+F “onMessageSend” // note that the call can be made in numerous ways.
- 3) When you want to hide working details





# An Event System

- Stores handlers based on an event type
- Executes handlers associated with an event



# Talker Event System

1. Building our own event broadcaster
2. Simple bind
3. Simple trigger
4. Allowing a form of preventDefault

1) show Talker#broadcaster.js  
2) show rooms#show.html.erb  
3) show plugin\_helper.rb





# Creating a Sample Plugin

1. onMessageSend
2. onMessageReceived
3. Talker.getCurrentName()
4. Talker.insertMessage()

```
plugin.onMessageSend = function(talkerEvent){  
  talkerEvent.content = talkerEvent.content.replace(/hello world/gi, 'OHAI WURLD!!!!1!!!one!!!!');  
  Talker.getMessageBox().val("");  
}
```

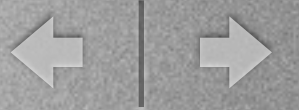
```
plugin.onMessageReceived = function(talkerEvent){  
  var name = Talker.getCurrentUser().name;  
  talkerEvent.content = talkerEvent.content.replace(name, '***' + name + '***');  
}
```

```
plugin.onMessageReceived = function(talkerEvent){  
  var name = Talker.getCurrentUser().name;  
  var re = new RegExp(name, "gi");  
  if (talkerEvent.content.match(re)){  
    talkerEvent.content = talkerEvent.content.replace(re, '<b>' + name + '</b>');  
    Talker.insertMessage(talkerEvent);  
    return false;  
  }  
}
```

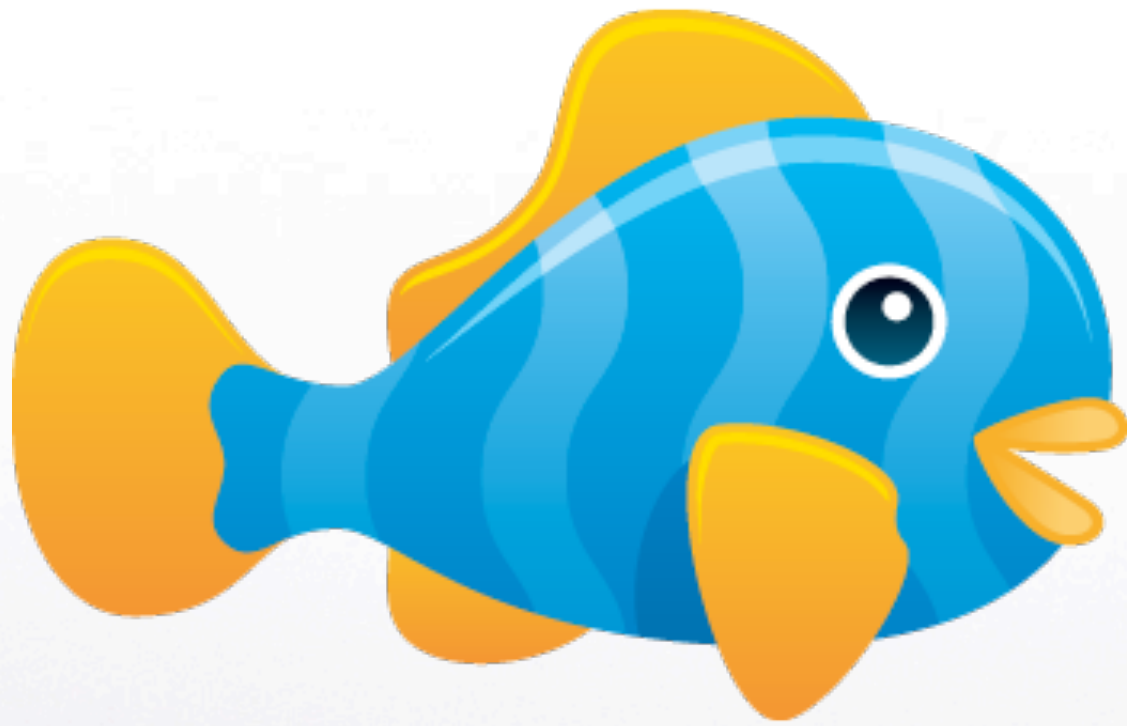


# Questions?





# Self Serving Messages



50%

OFF



# Follow Me

Twitter: @xutopia

GitHub: garyharan