# Hypermedia Hacking Workshop

Mike Amundsen
API Academy at CA Technologies
@mamund

# Plain JSON and Change

# What happens when you add a field?

- First release went greaty but…
- "We need to add a new field"
- Now what?

# Adding a new field on the server

- You can change the server models independently
  - Data Model (storage.js)
  - Object Model (map.js)
  - Representation Model (transitions.js)
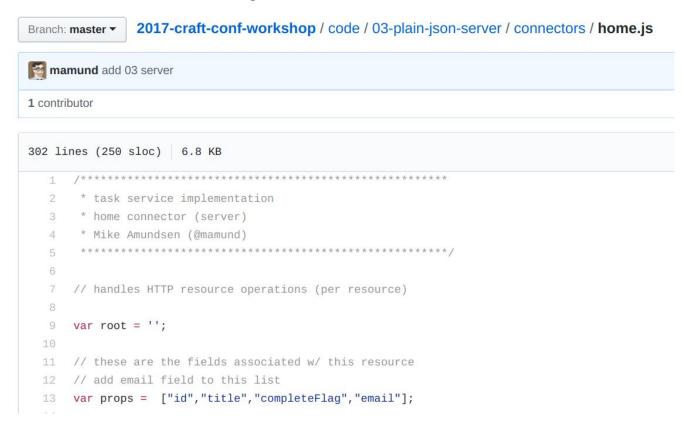- You can release that intro production but…
- **The client ignores it!**

# Adding a new field means building a new client

- Update client object model
- Update client representation model
- Update client transition model
- **We usually call this versioning**

# Exercise:
## Add Email Support

# Update server home.js

**2017-craft-conf-workshop** / code / 03-plain-json-server / connectors / **home.js**

**mamund** add 03 server

1 contributor

302 lines (250 sloc) | 6.8 KB

```
 1   /*********************************************************
 2    * task service implementation
 3    * home connector (server)
 4    * Mike Amundsen (@mamund)
 5    *********************************************************/
 6
 7   // handles HTTP resource operations (per resource)
 8
 9   var root = '';
10
11   // these are the fields associated w/ this resource
12   // add email field to this list
13   var props = ["id","title","completeFlag","email"];
```

# Update map.js

**mamund** add 03 server

**1** contributor

77 lines (63 sloc) │ 1.32 KB

```
 1  /*********************************************************
 2   * task service implementation
 3   * mapping document (server)
 4   * Mike Amundsen (@mamund)
 5   *********************************************************/
 6
 7  // hold mapping rules for interface v. stored data
 8
 9  // key = internal name
10  // value = interface name
11  function storeMap() {
12    var todo, rtn;
13    rtn = {};
14
15    todo = {};
16    todo.id = "id";
17    todo.completeFlag = "completed";
18    todo.title = "title";
19    // add email field here
20    todo.email = "email";
21    rtn.todo = todo;
```

# Update transitions.js

mamund add 03 server

1 contributor

```
107    trans.push({
108      name : "addLink",
109      type : "safe",
110      action : "read",
111      kind : "todo",
112      target : "list",
113      prompt : "Add ToDo"
114    });
115    // add email field here
116    trans.push({
117      name : "addForm",
118      type : "unsafe",
119      action : "append",
120      kind : "todo",
121      target : "list",
122      prompt : "Add ToDo",
123      inputs : [
124        {name : "title", prompt : "Title"},
125        {name : "completed", prompt : "Complete", value : "false"},
126        {name : "email", prompt : "Email"}
127      ]
128    });
```

# Update json-client.js

```
31    // the only fields to process
32    // add email to this list
33    g.fields = ["id","title","email"];
34
35    // all URLs & action details
36    // add email field to "add" and "edit"
37    g.actions = {
38      collection: {href:"/", prompt:"All Tasks"},
39      item:        {href:"/{id}", prompt:"Item"},
40      add:         {href:"/", prompt:"Add Task", method:"POST",
41                      args:{
42                        title: {value:"", prompt:"Title", required:true},
43                        email: {value:"", prompt:"Email", required:false}
44                      }
45                    },
46      edit:        {href:"/{id}", prompt:"Edit", method:"PUT",
47                      args:{
48                        id: {value:"{id}", prompt:"Id", readOnly:true},
49                        title: {value:"{title}", prompt:"Title", required:true},
50                        email: {value:"{email}", prompt:"Email", required:false}
51                      }
52                    }
53    };
```

# The OAA Challenge

# The OAA Challenge

- All clients deal with three key elements:
  - Objects
  - Addresses
  - Actions
- This is usually done in code
- When any of the OAA change, the code MUST change
- Adding email support is a change to the "O" in "OAA"
- What are examples of Address and Action changes?
- How can we support OAA changes w/o changing client code?