Learning Client Hypermedia from the Ground Up

Mike Amundsen
@mamund
@RWCBook

RESTful Web Clients

Morning - From Zero to Links

- What is Hypermedia and Why Use It?
- The Plain JSON Client
- BREAK
- Plain JSON and Change
- The JSON Link Client
- BREAK
- The Link Client and Change
- Daily Wrap-Up and Look Ahead

Afternoon - From Templates to Profiles

- The JSON Template Client
- The JSON Template Client and Change
- BREAK
- The JSON Profile Client
- The JSON Profile Client and Change
- BREAK
- Course Summary

RESTful
Web Clients
ENABLING REUSE THROUGH HYPERMEDIA

Resources

- The online repo will host all the materials
  - https://github.com/apiacademy/2017-07-safari-hypermedia-workshop
  - Slides
  - Code
  - Notes
- This repo will stay "alive" after the course is over
  - Feel free to use the Issues section to discuss details
  - Submit PRs to improve the content

General Remarks

- We'll have a short Q&A review after each section
- Feel free to post questions throughout the day.
- Be sure to mute any microphones unless you are speaking
- We'll have breaks every hour
- If needed, you can step away from the terminal any time

O'REILLY

RESTful
Web Clients

ENABLING REUSE THROUGH HYPERMEDIA

Let's get started!

Afternoon:
From Templates to Profiles

Mike Amundsen
@mamund
@RWCBook

The JSON Template Client

Mike Amundsen
@mamund
@RWCBook

Lecture:
Describing Actions

Describing ACTIONS

- ACTIONS: more than navigation (LO), transclusion (LE)
- Another word for ACTIONS is Templates (LT, LI, LN)
- In HTML, a Template is a FORM
- Template Clients understand ACTION descriptions

Describing ACTIONS

- What is needed for describing ACTIONS for HTTP?
  - URL
  - Method (CM)
  - Format (CR)
- Also INPUT properties
  - Name/Id (givenName)
  - Prompt ("First Name")
  - Constraints (required, input pattern, etc.)
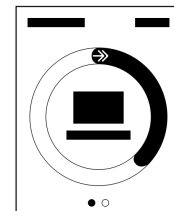
Describing ACTIONS

```json
{
  "template" : {
    "method" : "POST", "id" : "create",
    "href" : "http://api.example.org/todo/",
    "args" : [
      {"name" : "title", "prompt" : "Title", "value" :
""},
      {"name" : "tags", "prompt" : "Tags", "value" : ""}
    ]
  }
}
```

Exercise:
Including Templates

Live Screen Session...

RESTful
Web Clients

Sidebar:
The Siren Media Type

Siren Media Type

**Structured Interface for Representing Entities (Siren)**
Kevin Swiber, 2012

"The biggest differentiator of Siren is Actions. … Siren also has a concept of 'class.' The class attribute may contain multiple descriptors of the current representation. I've avoided calling these 'type descriptors.' They act more like 'mixin descriptors.'"

https://github.com/kevinswiber/siren

Siren Media Type
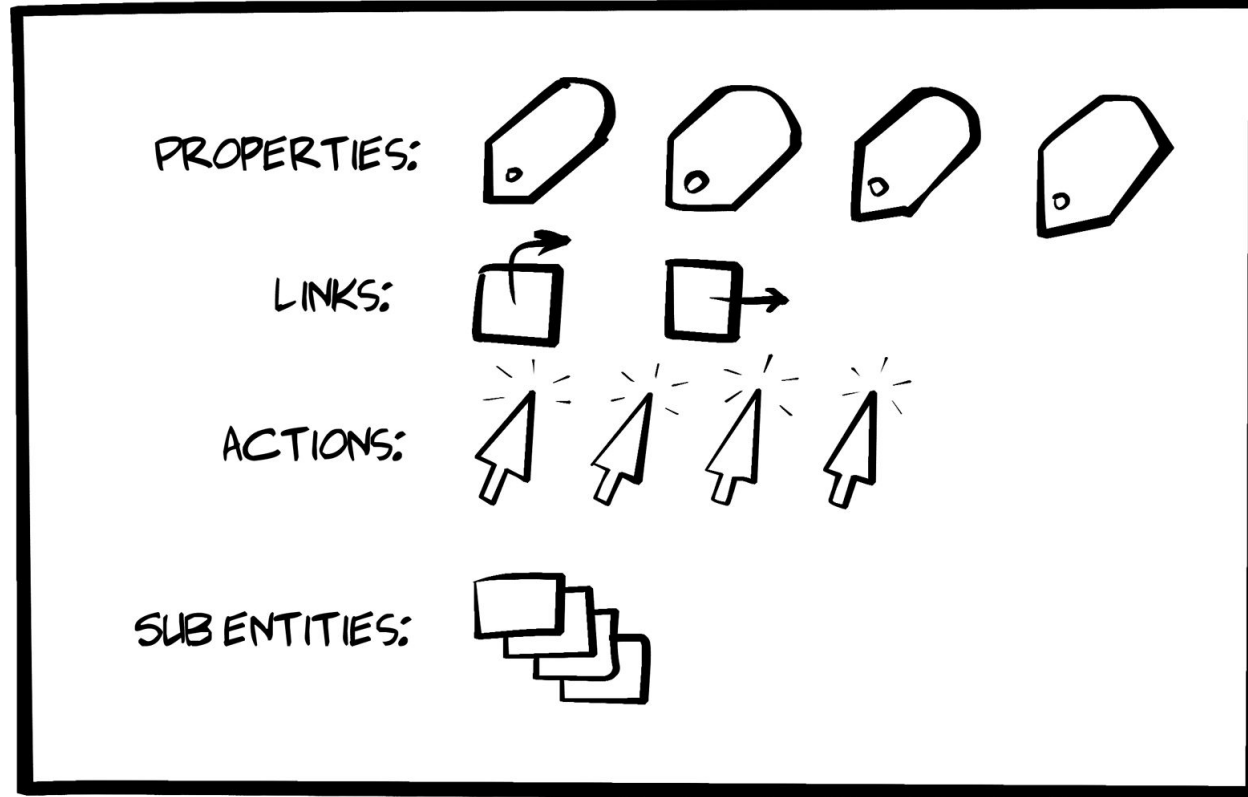
```json
{
  "class": ["order"],
  "properties": {
    "orderNumber": 42,
    "itemCount": 3,
    "status": "pending"
  },
  "entities": [
    {
      "class": ["items", "collection"],
      "rel": ["http://x.io/rels/order-items"],
      "href": "http://api.x.io/orders/42/items"
    },
    {
      "class": ["info", "customer"],
      "rel": ["http://x.io/rels/customer"],
      "properties": {
        "customerId": "pj123",
        "name": "Peter Joseph"
      },
      "links": [
        { "rel": ["self"], "href": "http://api.x.io/customers/pj123"}
      ]
    }
  ],
```

```
      "links": [
        { "rel": ["self"], "href": "http://api.x.io/customers/pj123"}
      ]
    }
  ],
  "actions": [
    {
      "name": "add-item",
      "title": "Add Item",
      "method": "POST",
      "href": "http://api.x.io/orders/42/items",
      "type": "application/x-www-form-urlencoded",
      "fields": [
        {"name": "orderNumber", "type": "hidden", "value": "42"},
        {"name": "productCode", "type": "text"},
        {"name": "quantity", "type": "number"}
      ]
    }
  ],
  "links": [
    {"rel": ["self" ], "href": "http://api.x.io/orders/42"},
    {"rel": ["previous" ], "href": "http://api.x.io/orders/41"},
    {"rel": ["next" ], "href": "http://api.x.io/orders/43"}
  ]
}
```

Quiz:
Is this a Template?
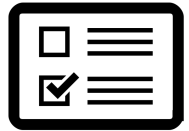
Is this a template?

- `http://api.example.org/filter`

Is this a template?

- `http://api.example.org/filter`

- `http://api.example.org/filter?search={text}`

Is this a template?

- `http://api.example.org/filter`

- `http://api.example.org/filter?search={text}`

- ```
  {
    href:"...", method="patch",
    format="application/json-patch",
    args: [
      {"name":"email","prompt":"Email","value":""},
      {"name":"phone-sms","prompt":"SMS","value",""}
    ]
  }
  ```

RESTful
Web Clients

ENABLING REUSE THROUGH HYPERMEDIA

The JSON Template Client And Change

Mike Amundsen
@mamund
@RWCBook

RESTful
Web Clients
ENABLING REUSE THROUGH HYPERMEDIA

Lecture:
Adding the Tag Field

Adding Tag Field Support

- The **tag** field SHOULD appear for every ToDo record
- It is an OPTIONAL field
- It contains a comma-delimited list of values
- It can be updated when adding new ToDo records
- It can be updated when editing existing ToDo records
- A FilterByTag operation SHOULD be supported

Adding Tag Field Support

- Views Affected
  - List
  - Item
  - Add
  - Update
- Also, a new Filter ACTION

Adding Tag Field Support

- What are the OAA Challenges here?
    - Object (*new field to display*)
    - Address (*none*)
    - Actions
        - Modified Add
        - Modified Update
        - New Filter

Can you predict the results?

Exercise:
Including Tag Field Support
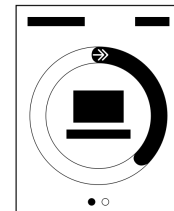
Live Screen Session…

Group Discussion:
Why is Siren Media Type Different?
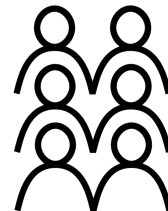
Live Screen Session...

BREAK

RESTful
Web Clients

O'REILLY

ENABLING REUSE THROUGH HYPERMEDIA

The JSON Profile Client

Mike Amundsen
@mamund
@RWCBook

RESTful
Web Clients

Lecture:
Describing Object Profiles

Describing Objects

- Properties of a field:
  - Name
  - Prompt
  - Render Flag
  - Display Mask
  - Order of Appearance

Profiles for Object Description (POD)

Profile for Object Description (POD)

Mike Amundsen 2016

"This specification is designed to describe the rules for displaying domain objects and their properties in a human-centric user interface."

http://rwcbook.github.io/pod-spec/

Profiles for Object Description (POD)

A Valid POD Document

```json
{
  "user" : {
    "id" : {"prompt" : "ID", "render" : "none"},
    "nick" : {"prompt" : "Nickname", "render" : "text"},
    "email" : {"prompt" : "Email", "render" : "text"},
    "name" : {"prompt" : "Full Name", "render" : "text"},
    "password" : {"prompt" : "Password", "render" : "text"},
    "dateCreated" : {"prompt" : "Date Created", "render" : "none"},
    "dateUpdated" : {"prompt" : "Date Updated", "render" : "none"}
  }
}
```

Exercise:
Adding POD Support
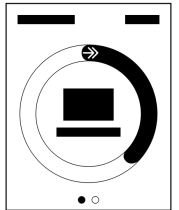
Live Screen Session...

Group Discussion:
Can We Improve the POD Spec?
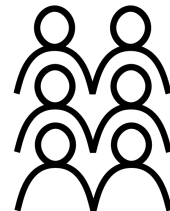
Live Screen Session...

JSON Profile Client And Change

Mike Amundsen
@mamund
@RWCBook

Lecture:
Supporting ToDo Descriptions
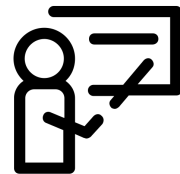
Adding Description Field Support

- The **description** field SHOULD appear for every record
- It is an OPTIONAL field
- It can be updated when adding new ToDo records
- It can be updated when editing existing ToDo records
- A FilterByDescription operation SHOULD be supported

Adding Tag Field Support

- Views Affected
    - List
    - Item
    - Add
    - Update
- Also, a new Filter ACTION

Adding Tag Field Support

- What are the OAA Challenges here?
  - Object (*new field to display*)
  - Address (*none*)
  - Actions
    - Modified Add
    - Modified Update
    - New Filter

Can you predict the results?

Exercise:
Including Description Support

Live Screen Session...

Sidebar:
The Collection+JSON Media Type

Collection+JSON Media Type (Cj)

Mike Amundsen, 2011

"[Cj] is similar to the Atom Syndication Format (RFC4287) and the Atom Publishing Protocol (RFC5023). However, Collection+JSON defines both the format and the protocol semantics in a single media type. [Cj] also includes support for Query Templates and expanded write support through the use of a Write Template."

http://amundsen.com/media-types/collection

Collection+JSON Media Type (Cj)

```json
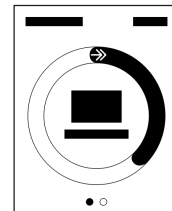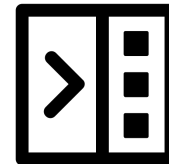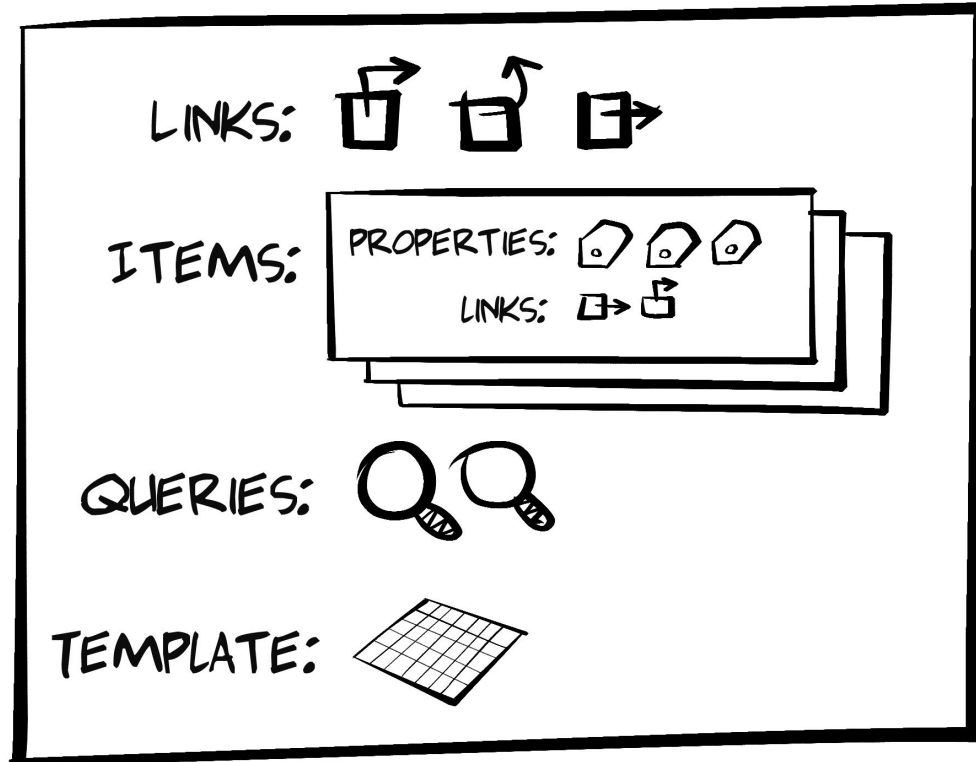{
  "collection": {
    "version": "1.0",
    "href": "http://rwcbook12.herokuapp.com", ❺
    "title": "TPS - Task Processing System",
    "links": [ ❶
      {
        "href": "http://rwcbook12.herokuapp.com/",
        "rel": "collection",
        "prompt": "All task"
      }
    ],
    "items": [ ❷
      {
        "rel": "item",
        "href": "http://rwcbook12.herokuapp.com/1sv697h2yij",
        "data": [
          {"name": "id", "value": "1sv697h2yij", "prompt": "id"},
          {"name": "title", "value": "Marina", "prompt": "title"},
          {"name": "completed", "value": "false", "prompt": "completed"}
        ]
      },
      {
        "rel": "item",
        "href": "http://rwcbook12.herokuapp.com/25ogsjhqtk7",
        "data": [
          {"name": "id", "value": "25ogsjhqtk7", "prompt": "id"},
          {"name": "title", "value": "new stuff", "prompt": "title"},
          {"name": "completed", "value": "true", "prompt": "completed"}
        ]
      }
    ],
```

```
      ]
    },
    {
      "rel": "item",
      "href": "http://rwcbook12.herokuapp.com/25ogsjhqtk7",
      "data": [
        {"name": "id", "value": "25ogsjhqtk7", "prompt": "id"},
        {"name": "title", "value": "new stuff", "prompt": "title"},
        {"name": "completed", "value": "true", "prompt": "completed"}
      ]
    }
  ],
  "queries": [ ❸
    {
      "rel": "search",
      "href": "http://rwcbook12.herokuapp.com/",
      "prompt": "Search tasks",
      "data": [
        {"name": "title", "value": "", "prompt": "Title"}
      ]
    }
  ],
  "template": { ❹
    "prompt": "Add task",
    "rel": "create-form",
    "data": [
      {"name": "title", "value": "", "prompt": "Title"},
      {"name": "completed", "value": "false", "prompt": "Complete"}
    ]
  }
 }
}
```

Group Discussion:
Why use Collection+JSON Media Type?

RESTful Web Clients

ENABLING REUSE THROUGH HYPERMEDIA
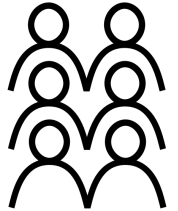
Created by Magicon
from Noun Project

Live Screen Session…

BREAK
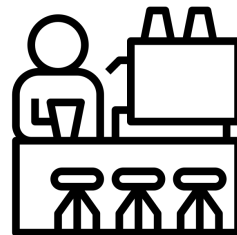
Course Summary

Mike Amundsen
@mamund
@RWCBook

RESTful
Web Clients

H-Factors

H-Factors

"The H Factor of a media-type is a measurement of the level of hypermedia support and sophistication of a media-type."

"H Factor values can be used to compare and contrast media types in order to aid in selecting the proper media-type(s) for your implementation."

- Mike Amundsen (2010)
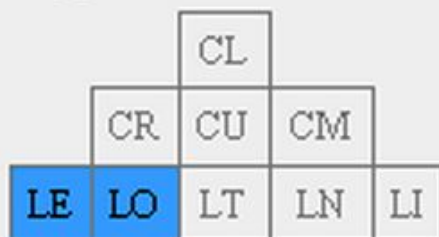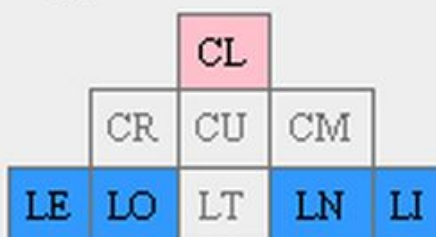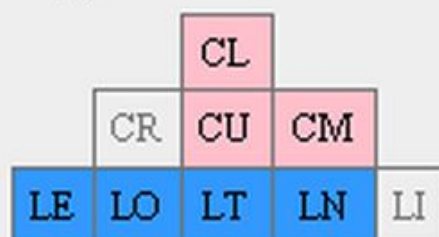
H-Factors

The OAA Challenge

The OAA Challenge

- All client apps need to deal with three key elements
  - Objects (Oxx)
  - Addresses (xAx)
  - Actions (xxA)
- Typically this is handled in client **code.**
- When any details of the OAA change, the client code MUST change, too.

The OAA Challenge

- Adding a field is a change to the Object ("O" in "OAA")
- Changing the URLs is a change to the Addresses
- Adding new filter or write operations is a change to the Actions

The OAA Challenge

- Adding a field is a change to the Object ("O" in "OAA")
- Changing the URLs is a change to the Addresses
- Adding new filter or write operations is a change to the Actions

How do we solve the OAA Challenge?

The Meta-Story
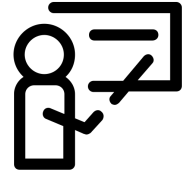
The Meta-Story

- It's the meta-data, silly!
- OBJECT meta-data
- ADDRESS meta-data
- ACTION meta-data

The Meta-Story

- It's the meta-data, silly!
- OBJECT meta-data
- ADDRESS meta-data
- ACTION meta-data

You solve the OAA Challenge with meta-data.

Quiz:
Pin the Client on the Media Type

Pin the Client on the Media Type

- I need a client that does mostly search/filter actions. Which media type should I use?

Pin the Client on the Media Type

- I need a client that does mostly search/filter actions. Which media type should I use?

- I need a client that supports HTTP PUT actions. Which media type should I use?

RESTful
Web Clients

ENABLING REUSE THROUGH HYPERMEDIA

Pin the Client on the Media Type

- I need a client that does mostly search/filter actions. Which media type should I use?

- I need a client that supports HTTP PUT actions. Which media type should I use?

- I need a client that supports HTTP PATCH actions. Which media type should I use?

Review

Morning Review : From Zero to Links

- Hypermedia Basics
- Classic CRUD Pattern
- The OAA Challenge
- The HAL Media Type

RESTful
Web Clients

ENABLING REUSE THROUGH HYPERMEDIA

Afternoon Review : From Templates to Profiles

- Describing Actions
- The Siren Media Type
- Display Profiles
- The Collection+JSON Media Type

O'REILLY

RESTful
Web Clients
ENABLING REUSE THROUGH HYPERMEDIA

Learning Hypermedia from the Ground Up

Mike Amundsen
@mamund
@RWCBook