



Grokking REST
Ben Ramsey • Memphis PHP User Group • 18 Nov 2010

Moontoast™
Your Social Commerce Platform



A tropical beach scene with palm trees and turquoise water. The image is split horizontally. The top half shows a close-up of palm fronds against a clear blue sky. The bottom half shows a wide view of a beach with turquoise water and a line of white buoys in the distance. A semi-transparent blue horizontal band is overlaid across the middle of the image, containing the text.

What is REST?



I'm a **URL** fetishist

#webdevgangsign







Representational State Transfer



SEPARATED

Client-server





Stateless



CACHE
COUNTY

Cache



Uniform Interface

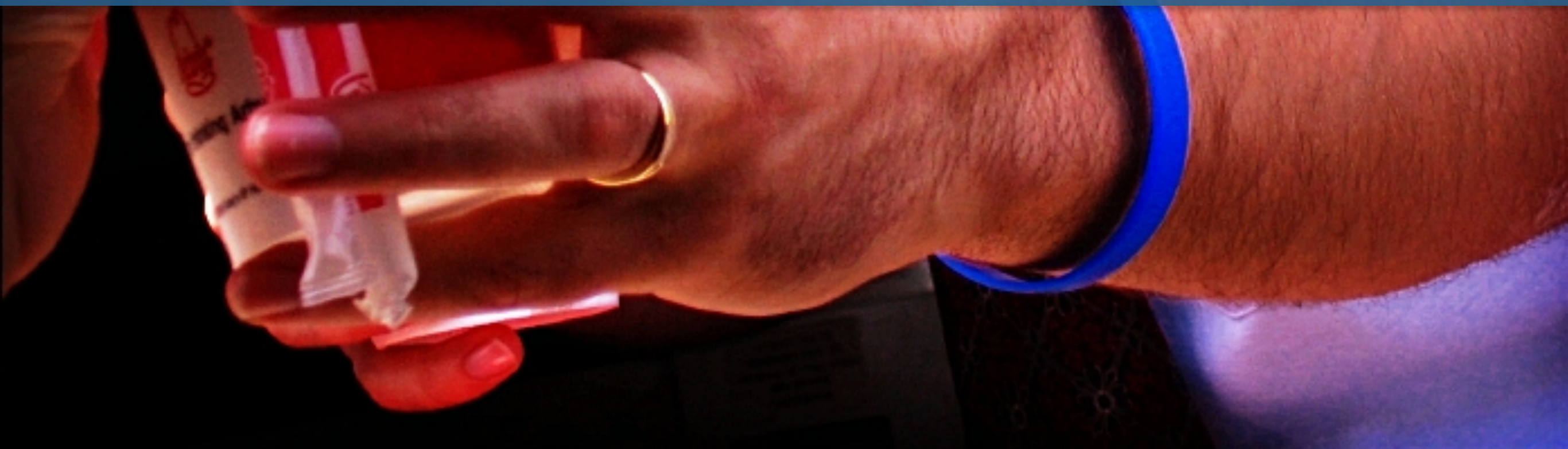


Layered System





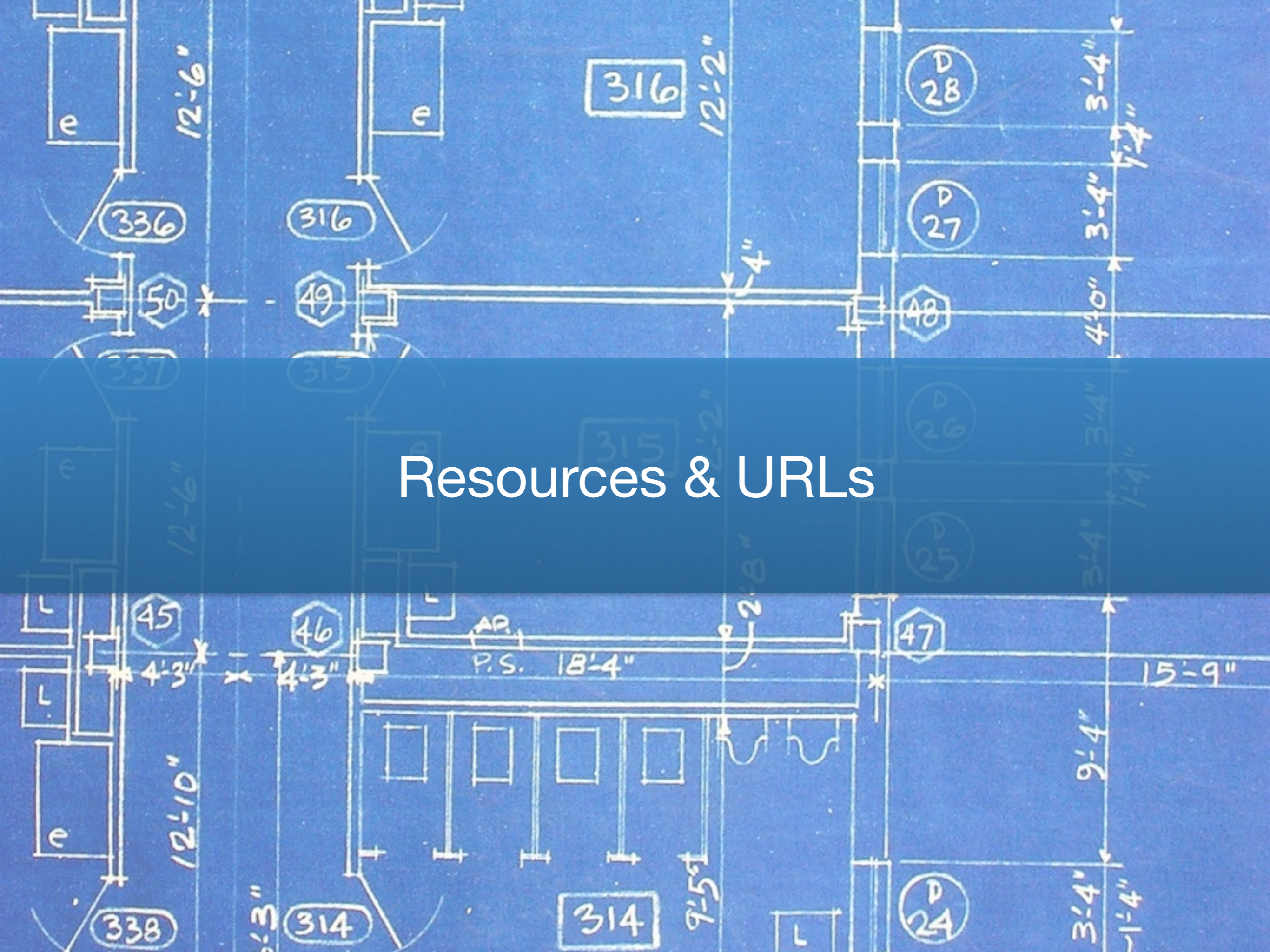
Code-on-demand



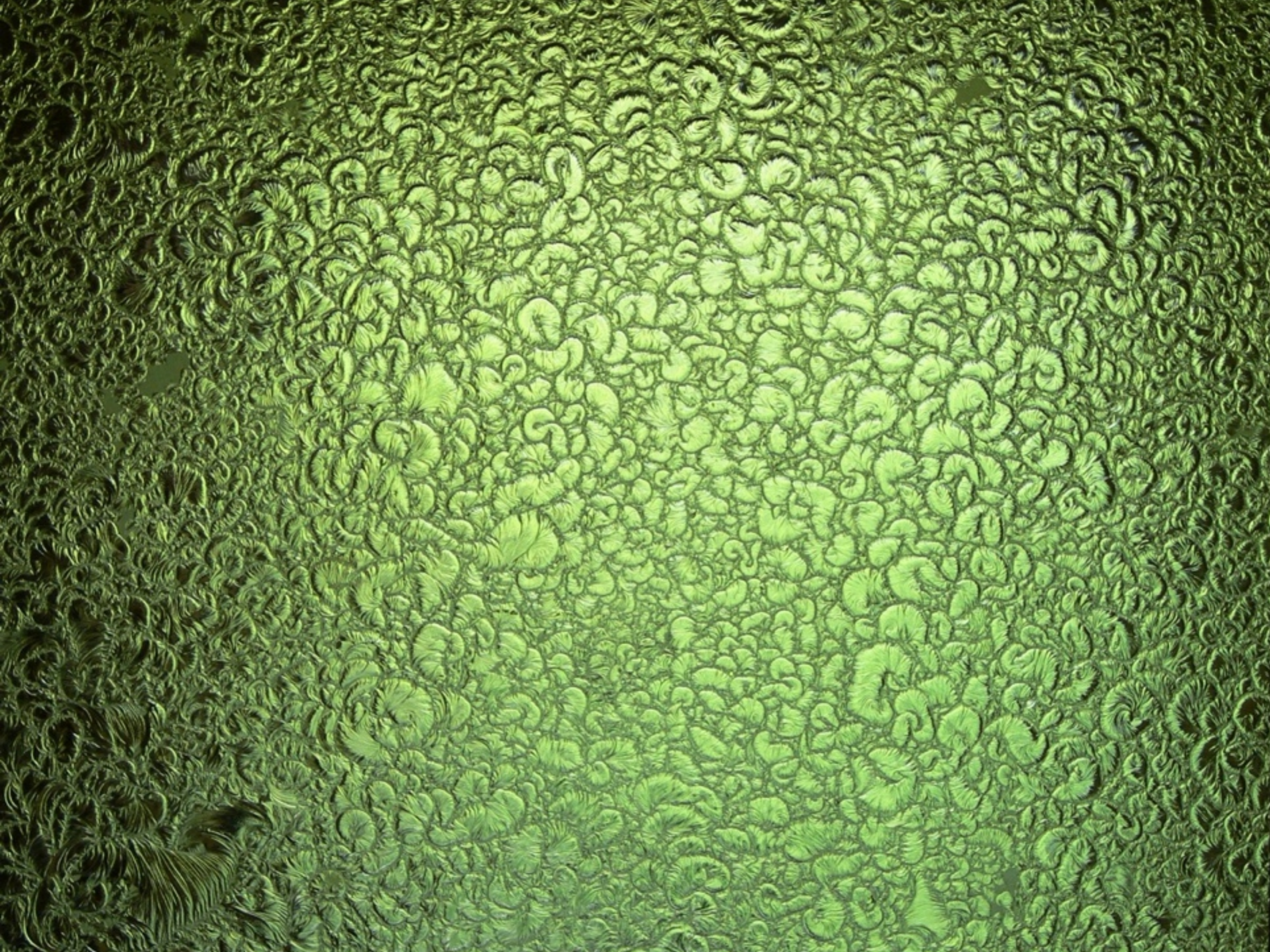
A sunset over a body of water with a sailboat in the foreground. The sky is filled with soft, orange and yellow clouds, and the sun is low on the horizon, casting a warm glow over the water. A single sailboat is visible in the lower right quadrant of the image. The text 'RESTful Concepts' is overlaid in white on a blue horizontal band across the middle of the image.

RESTful Concepts

Resources & URLs



URLs do not matter



Using URLs as interfaces
creates tight coupling
between the client & server

<http://api.example.org/user/getUser?u=ramsey>

Using URL templates creates
tight coupling between the
client & server

`http://api.example.org/user/{username}/photos`

Your hypermedia format
should expose the URLs that
represent your resources


```
<?xml version="1.0" encoding="utf-8"?>
<entry xmlns="http://www.w3.org/2005/Atom"
       xml:base="http://atom.example.org/">
  <title>ramsey</title>
  <author>
    <name>ramsey</name>
  </author>
  <link rel="self" href="user/ramsey"/>
  <link rel="edit" type="application/atom+xml;type=entry"
        href="user/ramsey"/>
  <link rel="related" href="user/ramsey/content"/>
  <id>tag:example.org,2008:user/ramsey</id>
  <updated>2009-09-21T13:45:00Z</updated>
  <published>2008-05-23T16:23:34Z</published>
  <content type="xhtml">
    <div class="vcard" xmlns="http://www.w3.org/1999/xhtml">
      <a class="fn">Ben Ramsey</a>
      <span class="tel">123-456-7890</span>
    </div>
  </content>
</entry>
```




Hypermedia

application/json


```
{  
  "id": "756315701",  
  "name": "Ben Ramsey",  
  "first_name": "Ben",  
  "last_name": "Ramsey",  
  "link": "http://www.facebook.com/ramseyben",  
  "gender": "male",  
  "locale": "en_US"  
}
```


Hypermedia?

No.

application/facebook+json;
type=user

application/xml


```
<?xml version="1.0" encoding="utf-8"?>
<user>
  <id>756315701</id>
  <name>Ben Ramsey</name>
  <first_name>Ben</first_name>
  <last_name>Ramsey</last_name>
  <link>http://www.facebook.com/ramseyben</link>
  <gender>male</gender>
  <locale>en_US</locale>
</user>
```


Hypermedia?

No.

application/facebook+xml;
type=user

application/xhtml+xml

Hypermedia?

Yes!

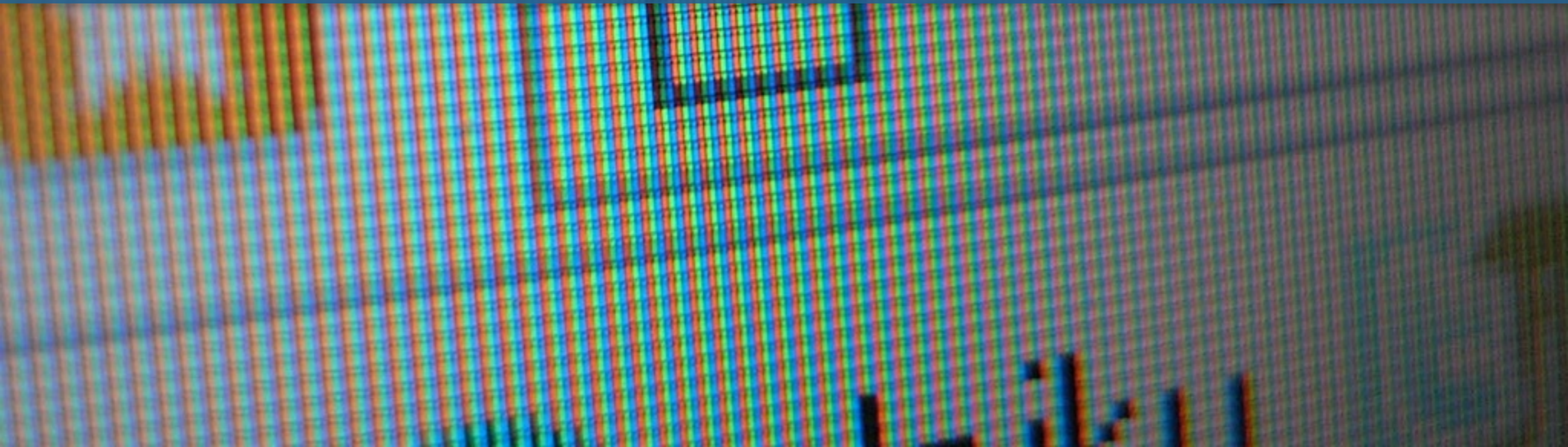
application/atom+xml

Hypermedia?

Yes!



HATEOAS



- ▶ How does a client know what to do with resources?
- ▶ How do you go to the “next” operation?
- ▶ What are the URLs for creating subordinate resources?
- ▶ Where is the contract for the service?

Hypermedia as the Engine of Application State (HATEOAS)

- ▶ Use links to allow clients to discover locations and operations
- ▶ Link relations are used to express the possible options
- ▶ Clients do not need to know URLs, so they can change
- ▶ The entire application workflow is abstracted, thus changeable
- ▶ The hypermedia type itself can be versioned if necessary
- ▶ No breaking of clients if the implementation is updated!



Why is REST so great?



REST scales

It's extensible

It's evolvable

Use this style for versioning

Accept: `application/vnd.myervice+xml;version=2`

Not this style

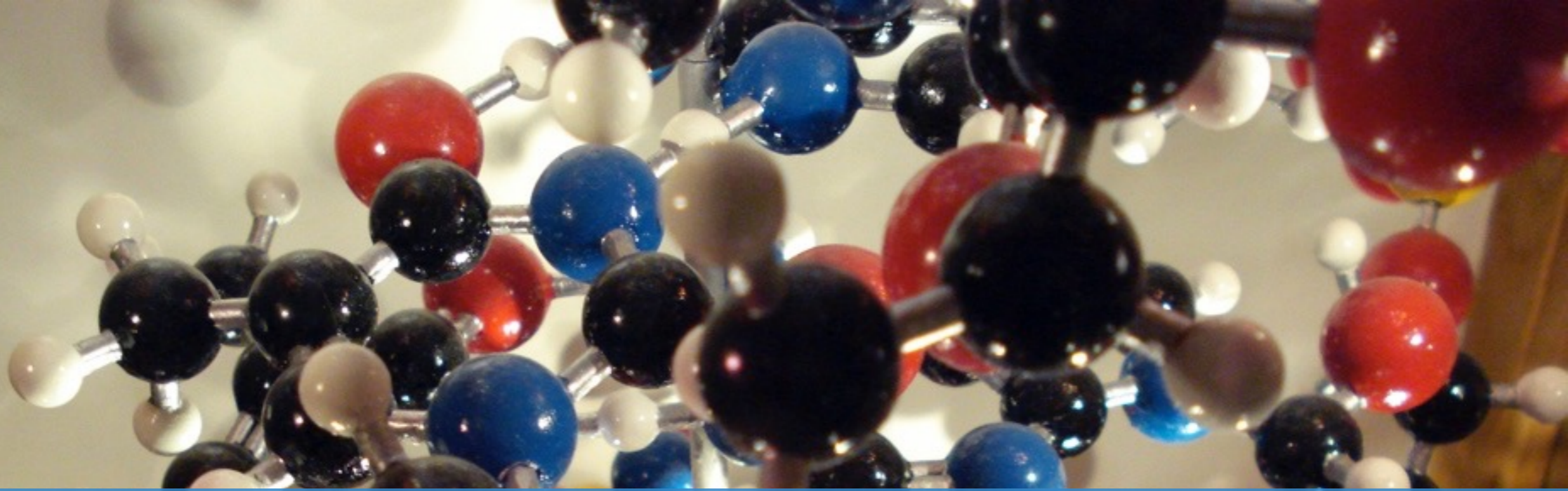
`http://api.example.org/v2/foo`

Clients can specify preferences

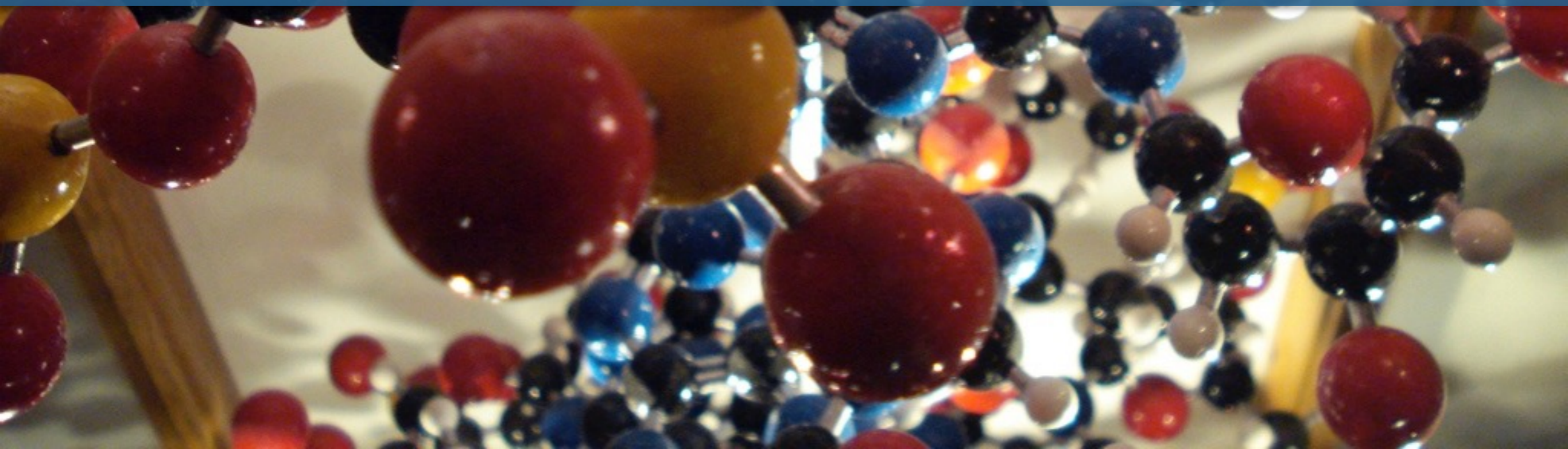
Accept:

```
application/vnd.mytype+xml;version=1;q=0.5,  
application/vnd.myervice+xml;version=2
```


It's seamless



Atom



Entry Document

`application/atom+xml;type=entry`

Feed/Collection Document

`application/atom+xml`

Category Document

`application/atomcat+xml`

Service Document

`application/atomsvc+xml`

```
GET / HTTP/1.1
```

```
Host: atom.example.org
```

```
Accept: application/atomsvc+xml, application/atomcat+xml, application/atom+xml
```

```
HTTP/1.x 200 OK
```

```
Date: Mon, 21 Sep 2009 16:33:45 GMT
```

```
Content-Type: application/atomsvc+xml
```

```
<?xml version="1.0" encoding="utf-8"?>
<service xmlns="http://www.w3.org/2007/app"
  xmlns:atom="http://www.w3.org/2005/Atom"
  xml:base="http://atom.example.org/">
  <workspace>
    <atom:title>Our Content Store</atom:title>
    <collection href="user">
      <atom:title>Users</atom:title>
      <accept>application/atom+xml;type=entry</accept>
      <categories href="cat/user"/>
    </collection>
    <collection href="content">
      <atom:title>Content</atom:title>
      <accept>application/atom+xml;type=entry</accept>
      <categories href="cat/content"/>
    </collection>
  </workspace>
</service>
```



```
GET /user HTTP/1.1
Host: atom.example.org
Accept: application/atom+xml
```

```
HTTP/1.x 200 OK
Date: Mon, 21 Sep 2009 16:34:26 GMT
Content-Type: application/atom+xml

<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom"
      xmlns:app="http://www.w3.org/2007/app"
      xml:base="http://atom.example.org/">
  <title>Users</title>
  <updated>2009-09-21T05:21:19Z</updated>
  <id>tag:example.org,2009-09:user</id>
  <app:collection href="user">
    <title>Users</title>
    <app:accept>application/atom+xml;type=entry</app:accept>
    <app:categories href="cat/user"/>
  </app:collection>
  <link rel="first" href="user"/>
  <link rel="last" href="user?p=23"/>
  <link rel="next" href="user?p=2"/>
  <entry>
    ...
  </entry>
</feed>
```

```
GET /cat/user HTTP/1.1
Host: atom.example.org
Accept: application/atomcat+xml
```

```
HTTP/1.x 200 OK
Date: Mon, 22 Sep 2009 09:39:26 GMT
Content-Type: application/atomcat+xml
```

```
<?xml version="1.0"?>
<app:categories
  xmlns:app="http://www.w3.org/2007/app"
  xmlns:atom="http://www.w3.org/2005/Atom"
  fixed="yes"
  scheme="http://atom.example.com/cat/user">
  <atom:category term="manager"/>
  <atom:category term="team"/>
  <atom:category term="user"/>
</app:categories>
```



```
POST /user HTTP/1.1
Host: atom.example.org
Content-Type: application/atom+xml;type=entry
```

```
<?xml version="1.0" encoding="utf-8"?>
<entry xmlns="http://www.w3.org/2005/Atom"
      xml:base="http://atom.example.org/">
  <title>ramsey</title>
  <author>
    <name>ramsey</name>
  </author>
  <id>tag:example.org,2008:user/ramsey</id>
  <published>2008-05-23T16:23:34Z</published>
  <content type="xhtml">
    <div class="vcard" xmlns="http://www.w3.org/1999/xhtml">
      <a class="fn">Ben Ramsey</a>
      <span class="tel">123-456-7890</span>
    </div>
  </content>
</entry>
```

```
HTTP/1.x 201 Created
```

```
Date: Mon, 22 Sep 2009 09:39:26 GMT
```

```
Location: http://atom.example.org/user/ramsey
```

```
Content-Type: text/html
```

```
<html>
```

```
<body>
```

```
  <p>Your new resource was created
```

```
    <a href="http://atom.example.org/user/ramsey">here</a>.</p>
```

```
</body>
```

```
</html>
```

The image features a sunset or sunrise scene. The sky is a gradient of colors, from a pale grey at the top to a deep orange and red near the horizon. A solid blue horizontal band runs across the middle of the image. In the center of this blue band, the text "Bringing it all together..." is written in a white, sans-serif font. Below the blue band, the sky is a deep orange and red, with a bright yellow sun partially visible on the right side. The horizon is dark, and a small, bright orange light is visible in the center of the dark area.

Bringing it all together...

Questions?

- ▶ I blog at benramsey.com.
- ▶ I tweet at [@ramsey](https://twitter.com/ramsey).
- ▶ Please rate this presentation at joind.in/2373.
- ▶ Find out more about REST by reading [*Architectural Styles and the Design of Network-based Software Architectures*](#) and “[How I Explained REST to my Wife](#).”
- ▶ My company is Moontoast. Check us out at moontoast.com. Follow us on [Twitter](#) and like us on [Facebook](#).

Grokking REST

Copyright © 2010 [Ben Ramsey](#). Some rights reserved.

Presented on November 18, 2010 at Memphis PHP User Group, Phoenix Communications, Nashville, TN.

This work is licensed under a [Creative Commons Attribution-Noncommercial-No Derivative Works 3.0 United States License](#).

For uses not covered under this license, please contact the author.



Photo Credits

- ▶ [Restful Summer](#), by Clear Inner Vision
- ▶ [Roatan Beach - Perfect Day](#), by Janusz Leszczynski
- ▶ [Tom Coates](#), by Patrick Lauke
- ▶ [Web Developer Gang Sign](#), by Josh Lewis
- ▶ [A Bar of Ivory Soap](#), by iirraa
- ▶ [Albert Einstein](#), Public Domain, from Wikimedia Commons
- ▶ [Separated](#), by Hansol Lee
- ▶ [Stateless](#) by Koesmanto Bong
- ▶ [Cache County, Utah](#) by J. Stephen Conn
- ▶ [used to interface](#), by Tom Hensel
- ▶ [Sedimentary Layers](#) by Mark Thurman
- ▶ [jeremy's tie](#) by Gitchel
- ▶ [Sand Banks Sunset](#), by Clear Inner Vision
- ▶ [Where I Teach](#), by Todd Ehlers
- ▶ [Frosted Glass door](#), Goderham Building, by Rick Innis
- ▶ [Drip Drops and the Spider Web](#), by Mike Bitzenhofer
- ▶ [#110 Hypertext Transfer Protocol](#), by maako
- ▶ [Before we had CAD, we had Lead!](#), by Wendell

Photo Credits

- ▶ [Molecule display](#), by Christian Guthier
- ▶ [Conclusion](#), by Mark Cummins