

REST Workshop

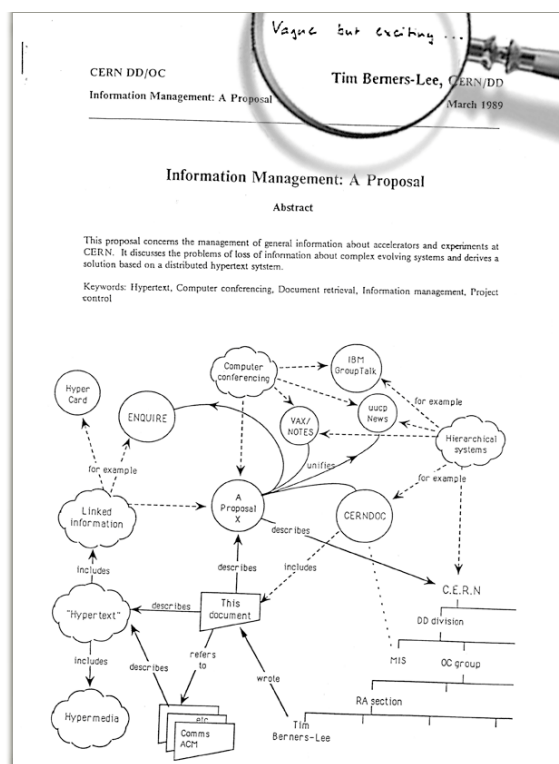
Brian Sletten (@bsletten (<http://twitter.com/bsletten>))
07/16/2013

Speaker's Qualification

- 20 years of experience
- Specialize in benefits and applicability of next-generation technologies
- Author "Resource-Oriented Architectural Patterns" Morgan & Claypool 2013
- One of **Top 100 Semantic Web People** (<http://www.semanticweb.com/semanticweb100/>)

Agenda

- Introduction
- Architecture
- Identity
- Interaction
- Representation



“WEB'S MAJOR GOAL WAS TO
BE A SHARED INFORMATION
SPACE THROUGH WHICH PEOPLE
AND MACHINES COULD
COMMUNICATE.”
TIM BERNERS-LEE

“WHAT WAS NEEDED WAS A WAY FOR PEOPLE
TO STORE AND STRUCTURE THEIR OWN
INFORMATION, WHETHER PERMANENT OR
EPHEMERAL IN NATURE, SUCH THAT IT COULD BE
USABLE BY THEMSELVES AND OTHERS, AND TO BE
ABLE TO REFERENCE AND STRUCTURE THE
INFORMATION STORED BY OTHERS SO THAT IT
WOULD NOT BE NECESSARY FOR EVERYONE TO
KEEP AND MAINTAIN LOCAL COPIES.”
ROY FIELDING

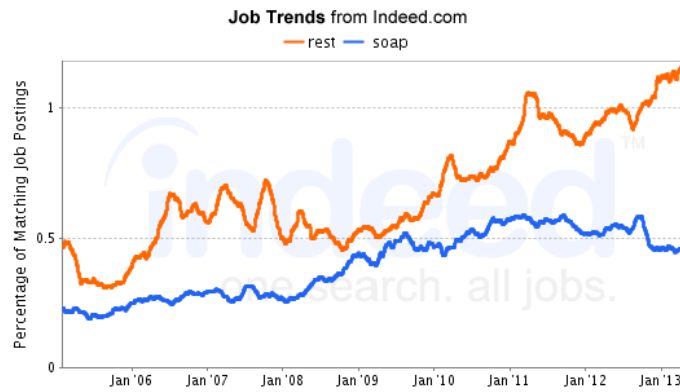
Why the Web works

- Loose coupling
- Localized changes
- Global namespace
- Clicking resolves and requests
- Content negotiation

Does your browser...

- Have to read a website's API documentation?
- Break when a website rearranges its content?
- Break when a website changes its implementation technology?

REST vs SOAP



<http://www.indeed.com/trendgraph/jobgraph.png?q=rest%2C+soap>

SOAP Request

```
POST /quote HTTP/1.1
Host: www.example.org
Content-Type: text/xml; charset=utf-8
Content-Length: nnn

<?xml version="1.0"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://www.w3.org/2001/12/soap-envelope"
  SOAP-ENV:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
  <SOAP-ENV:Body xmlns:m="http://www.example.org/quote">
    <m:GetQuote>
      <m:QuoteName>AAPL</m:QuoteName>
    </m:GetQuote>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

REST Request

```
GET /quote/AAPL HTTP/1.1
Host: www.example.org
```

SOAP Response

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: nnn

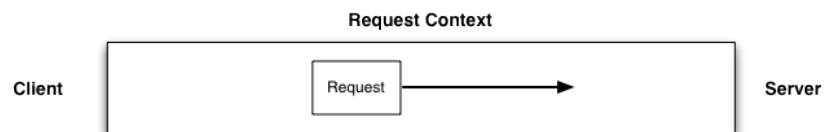
<?xml version="1.0"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://www.w3.org/2001/12/soap-envelope"
  SOAP-ENV:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
  <SOAP-ENV:Body xmlns:m="http://www.example.org/quote">
    <m:GetQuoteResponse>
      <m:Quote>358.16</m:Quote>
    </m:GetQuoteResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

REST Response

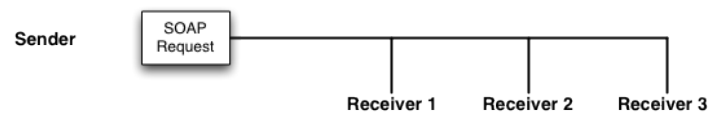
```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: nnn

<?xml version="1.0"?>
<quote name="AAPL">
  <price>358.16</price>
</quote>
```

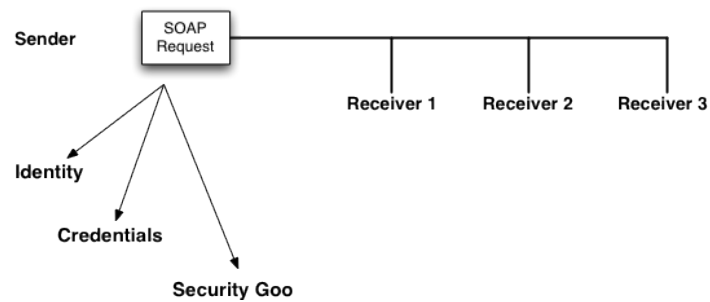
Contextualized Request



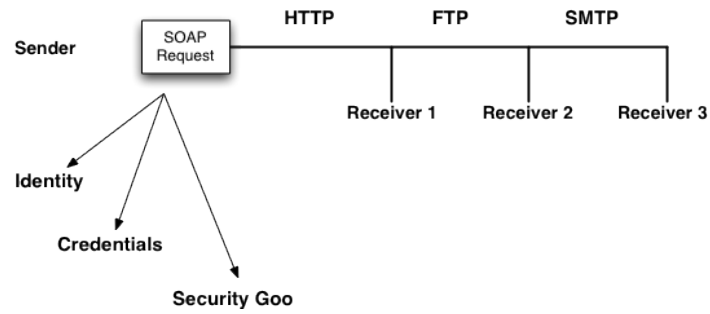
Decontextualized Request



Decontextualized Request



Decontextualized Request



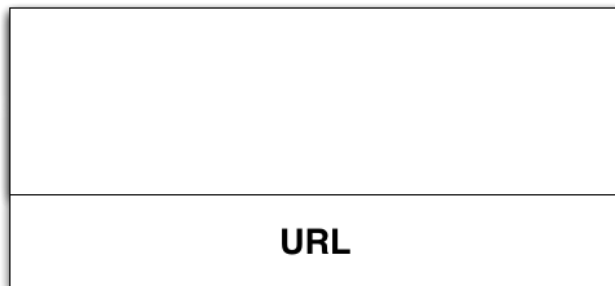
What REST Isn't...

- RPC through URLs
- A direct replacement for SOAP
- XML over HTTP
- Insecure
- A toy
- Whatever you want it to be

RMM Level 0



RMM Level 1



RMM Level 2

HTTP
URL

RMM Level 3

Hyper Media
HTTP
URL

Architecture

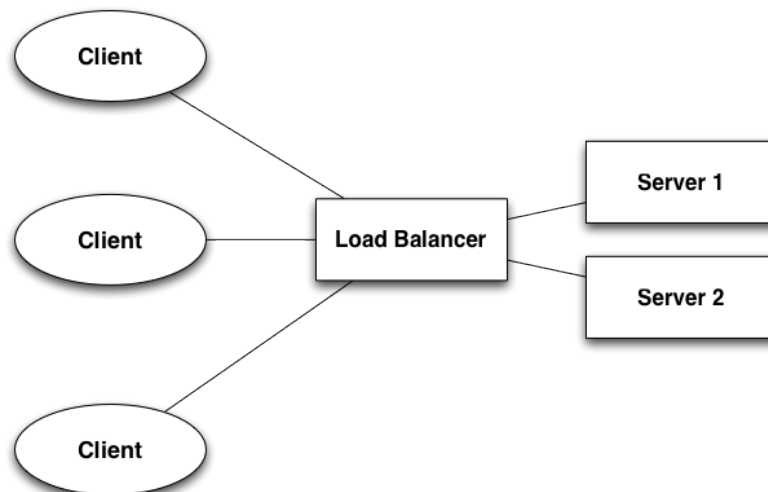
Desired Properties

- Performance
- Scalability
- Generality
- Simplicity
- Modifiability
- Extensibility

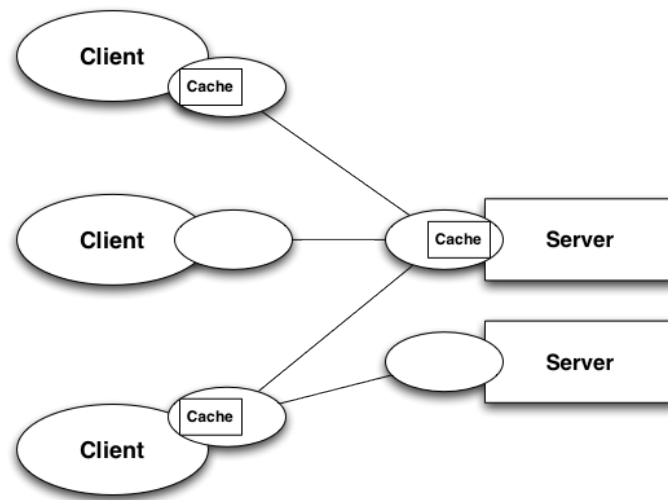
Client Server



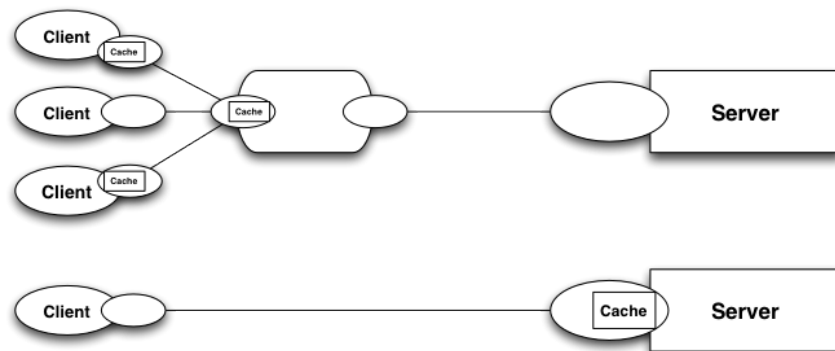
Stateless Client-Server



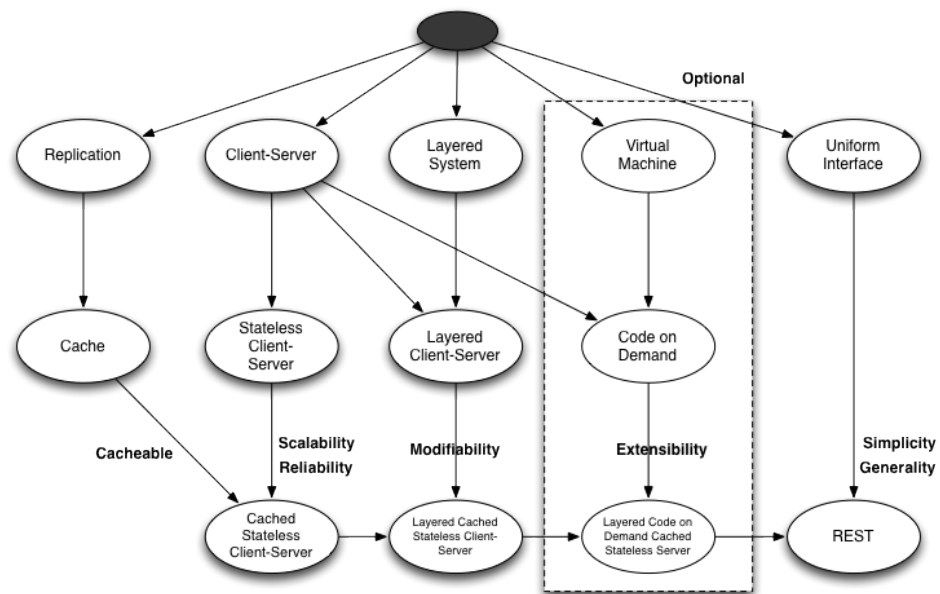
Stateless Client-Server w/ Cache and Uniform Interface



Layered Stateless Client-Server w/ Cache and Uniform Interface



REST on a Slide



"Architectural Styles and the Design of Network-based Software Architectures"
 (<http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm>)

Exercises

Discuss:

- Does REST matter?
- If so, why does REST matter?
- Why is it important to be RESTful?
- Explain the benefit of each of the RMM levels

Identity

“WHAT'S IN A NAME? THAT
WHICH WE CALL A ROSE BY
ANY OTHER NAME WOULD
SMELL AS SWEET.”
SHAKESPEARE, “ROMEO AND JULIET”

“THE FUNCTION OF A NAME
IS TO FACILITATE SHARING.”
ROSS J. ANDERSON

“WHEN SYSTEMS BECOME LARGE, THE SCALE-UP PROBLEMS ARE NOT LINEAR; THERE IS OFTEN A QUALITATIVE CHANGE IN COMPLEXITY, AND SOME THINGS THAT ARE TRIVIAL TO DEAL WITH IN A NETWORK OF ONLY A FEW MACHINES AND PRINCIPALS (SUCH AS NAMING) SUDDENLY BECOME A BIG DEAL.”
ROSS J. ANDERSON

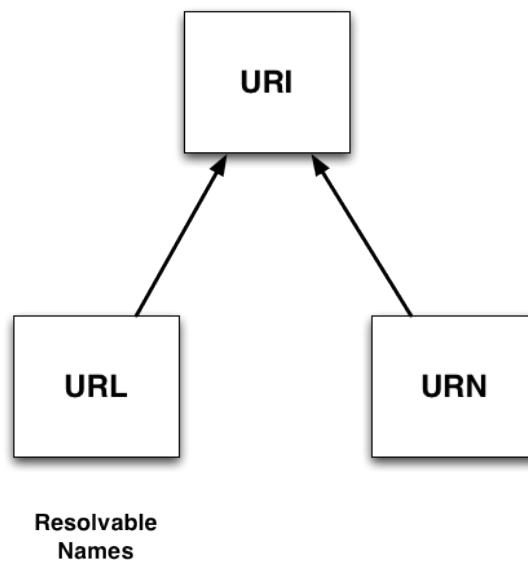
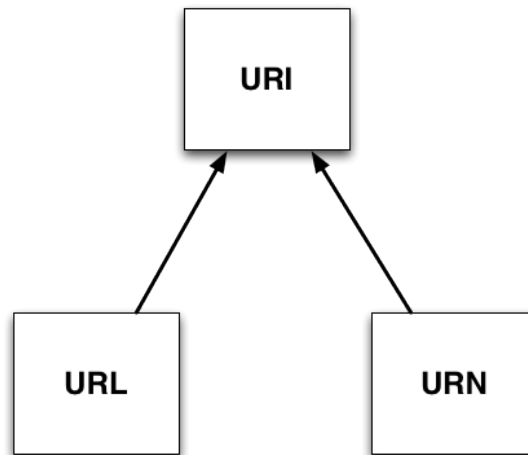
Naming Schemes

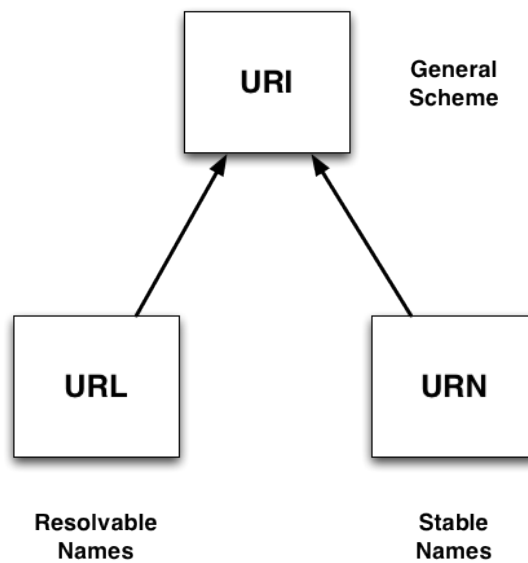
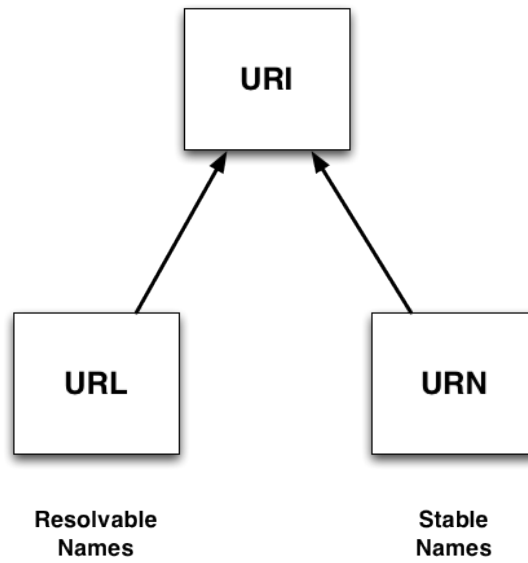
- First name, Last Name
- Database keys
- Social Security Numbers
- ISBN
- DOI
- Telephone numbers

Naming Scheme Properties

- Identity
- Disambiguation
- Scope
- Resolvability

URIs





`http://bosatsu.net/foaf/brian.rdf`

`ftp://ftp.funet.fi/pub/standards/RFC/rfc959.txt`

`urn:isbn:0913966568`

Uses for URLs

Documents

`http://myserver.com/docs/reports/2009/qtr/3`

Data

`http://myserver.com/gene/ade2`

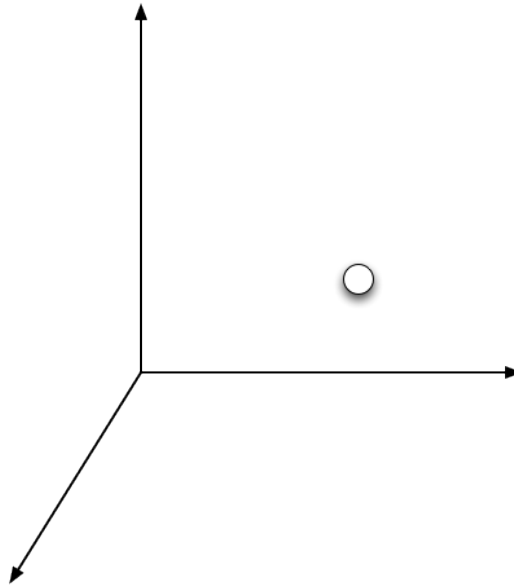
Services

`http://myserver.com/order?type=open`

Concepts

`http://purl.org/net/bsletten`

Information Space

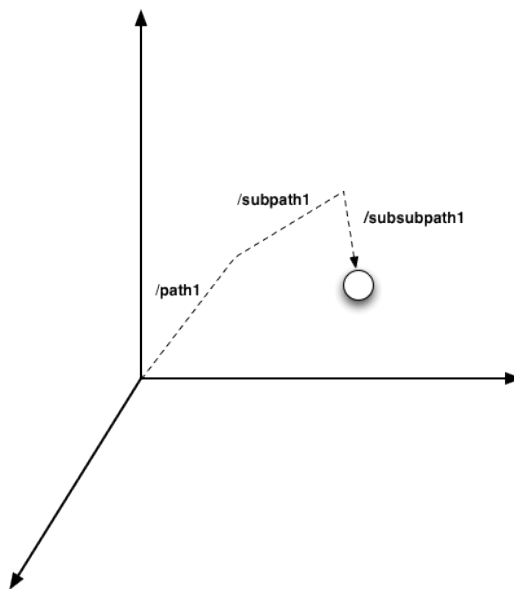


<http://localhost:8080/wink/wiki/bosatsu/uberconf/2013/rest-workshop/>

Page 47 of 229

Bosatsu Uberconf REST Workshop

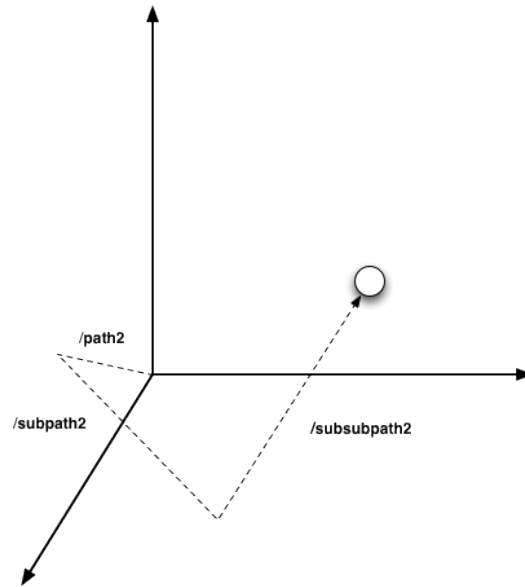
7/16/13 9:00 AM



<http://someserver.com/path1/subpath1/subsubpath1>

<http://localhost:8080/wink/wiki/bosatsu/uberconf/2013/rest-workshop/>

Page 48 of 229



```
http://someserver.com/path2/subpath2/subsubpath2
```

URI Aliases

```
http://bosatsu.net/time/zone/PST
```

```
http://bosatsu.net/time/state/us/hawaii
```

Canonical vs Historical

`http://bosatsu.net/talks/topic/rest`

`http://bosatsu.net/talks/conf/nfjs/2013/rest/minneapolis/1`

“GLOBAL NAMING LEADS TO
GLOBAL NETWORK EFFECTS.”
ARCHITECTURE OF THE WORLD WIDE WEB, VOLUME I

“ANY RESOURCE ANYWHERE
CAN BE GIVEN A URL.”
AXIOMS OF WEB ARCHITECTURE

“ANY RESOURCE OF
SIGNIFICANCE SHOULD BE
GIVEN A URL.”
AXIOMS OF WEB ARCHITECTURE

“IT DOESN'T MATTER TO
WHOM OR WHERE YOU
SPECIFY THAT URI, IT WILL
HAVE THE SAME MEANING.”
AXIOMS OF WEB ARCHITECTURE

“A URI WILL REPEATEDLY
REFER TO 'THE SAME' THING.”
AXIOMS OF WEB ARCHITECTURE

“THE SIGNIFICANCE OF
IDENTITY FOR A GIVEN URL IS
DETERMINED BY THE PERSON
WHO OWNS THE URL WHO
FIRST DETERMINED WHAT IT
POINTS TO.”
AXIOMS OF WEB ARCHITECTURE

“URL SPACE DOES NOT HAVE
TO BE THE ONLY UNIVERSAL
SPACE.”
AXIOMS OF WEB ARCHITECTURE

“WHAT MAKES A COOL URI?
A COOL URI IS ONE WHICH
DOES NOT CHANGE.”
“COOL URIS DON'T CHANGE”

“WHAT SORTS OF URI
CHANGE?
URIS DON'T CHANGE. PEOPLE
CHANGE THEM.”
“COOL URIS DON'T CHANGE”

Things to Avoid

- Author name
- Subject
- Status
- Access
- File name extension
- Software mechanism

A name...

is a name...

is a name...

is an attack vector...

```
http://example.com&gibberish=1234@167772161
```

10.0.0.1

```
http://example.com\coredump.cx
```

In Firefox, <http://coredump.cx>


```
http://example.com;.coredump.cx
```

In IE, <http://coredump.cx>

Safari, Error.

Others, <http://example.com/;.coredump.cx>

Exercises

Design an information space for:

- A county library system
- Fast food restaurant
- A package delivery service
- A museum
- A radio station
- A micro-blogging site

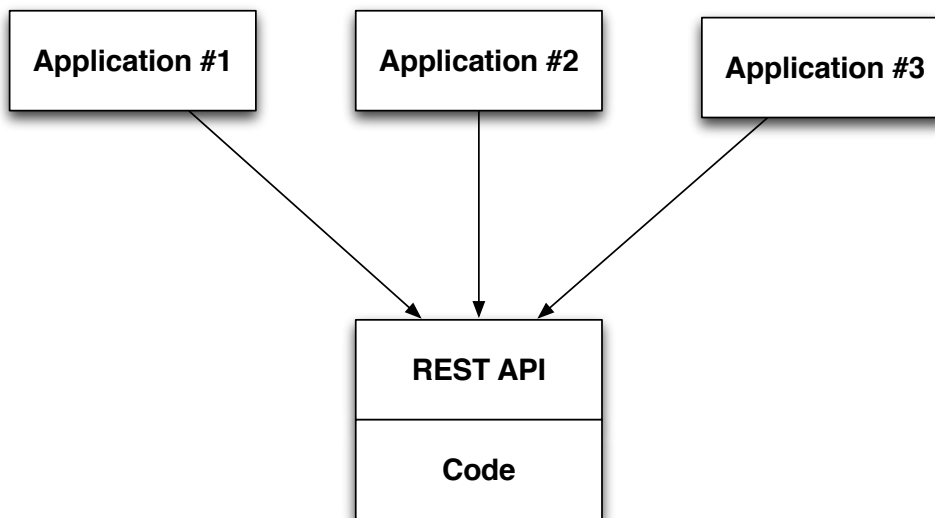
Interaction

<http://localhost:8080/wink/wiki/bosatsu/uberconf/2013/rest-workshop/>

Page 67 of 229

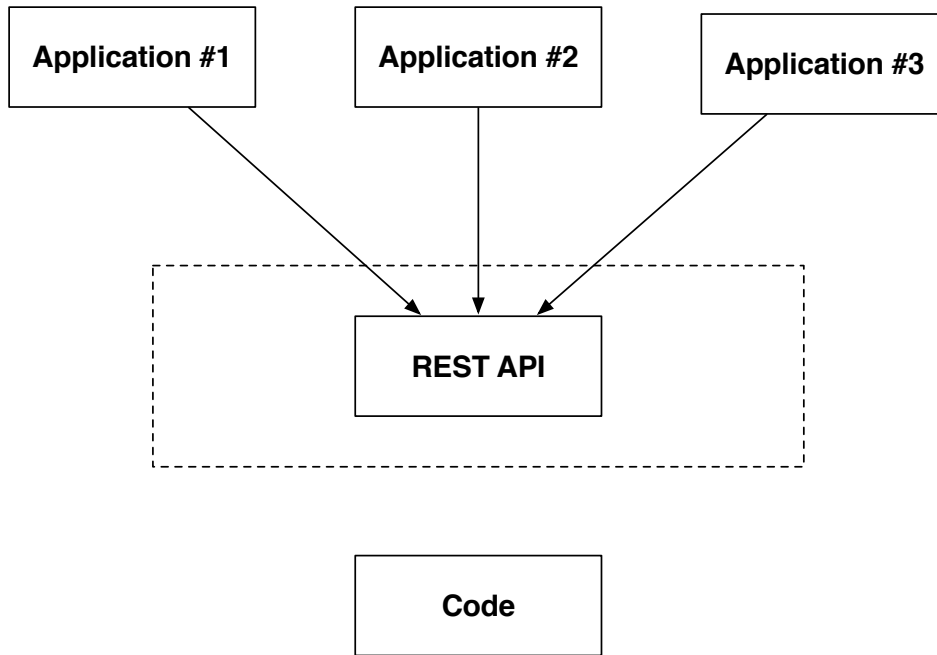
Bosatsu Uberconf REST Workshop

7/16/13 9:00 AM



<http://localhost:8080/wink/wiki/bosatsu/uberconf/2013/rest-workshop/>

Page 68 of 229



GET



POST



PUT



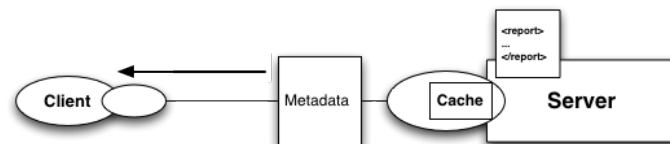
DELETE



<http://localhost:8080/wink/wiki/bosatsu/uberconf/2013/rest-workshop/>

Page 73 of 229

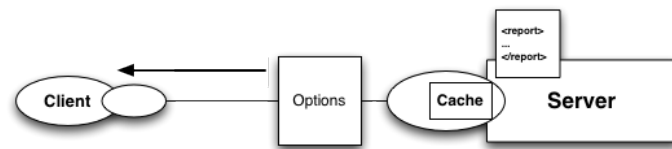
HEAD



<http://localhost:8080/wink/wiki/bosatsu/uberconf/2013/rest-workshop/>

Page 74 of 229

OPTIONS



PATCH



```
OPTIONS /example/buddies.xml HTTP/1.1
Host: www.example.com
```

```
HTTP/1.1 200 OK
Allow: GET, PUT, POST, OPTIONS, HEAD, DELETE, PATCH
Accept-Patch: application/example, text/example
```

```
PATCH /example/buddies.xml HTTP/1.1
Host: www.example.com
Content-Type: application/example
If-Match: "e0023aa4e"
Content-Length: 100
```

```
[patch changes]
```

```
HTTP/1.1 204 No Content
Content-Location: /example/buddies.xml
ETag: "e0023aa4f"
```

Idempotency

GET	✓
POST	✗
PUT	✓
PATCH	~✓
DELETE	✓

Response Codes: 2XX

Code	Meaning
200	Ok
201	Created
202	Accepted
204	Success

Response Codes: 3XX

Code	Meaning
301	Moved Permanently
302	Found
303	See Other
304	Not Modified

Response Codes: 4XX

Code	Meaning
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
405	Method Not Allowed
409	Conflict
411	Length Required
413	Entity Too Long
415	Unsupported Media Type

Response Codes: 5XX

Code	Meaning
500	Internal Error
503	Service Unavailable

“THE WEB IS AN INFORMATION SPACE. WHEN
YOU EXPLORE IT, YOU DON'T END UP BUYING
STUFF, AGREEING TO ANYTHING, OR – IN THIS
CASE, LOSING YOUR DOMAIN NAME...”
TIM BERNERS-LEE

Hall of Flame

```
From: <csupport@registerapi.com>
Date: Fri Apr 11, 2003 19:31:28 US/Eastern
To: timbl@w3.org
Subject: Confirm Domain Transfer

A Transfer Request was submitted for the following domains.
Click on the following link to confirm the domain transfer request for these domains.

https://secure.registerapi.com/order/trx/confirm.php?id=ces0[...]cbI

Your Transfer Request Code IS: ces[...]cbI
https://secure.registerapi.com/order/trx/confirm.php

Domains:
WWW.ORG
If you did not request the transfer of these domains then DO NOT click on the above links.
By not clicking you are preventing a domain registrar transfer from taking place.

Thank you, The Automated Domain Transfer System
```

CSRF

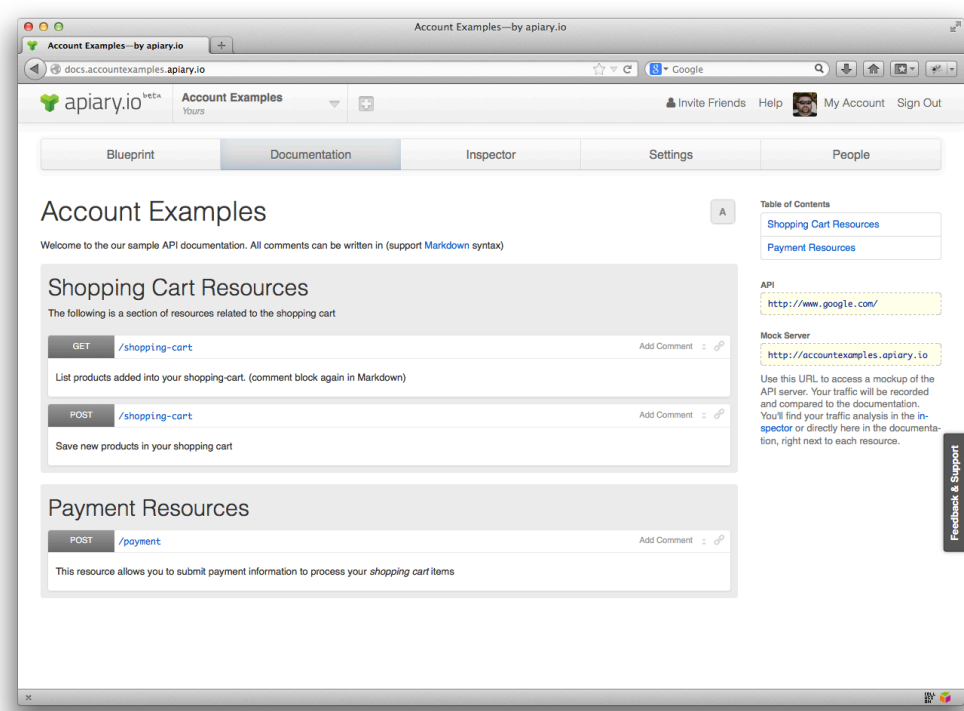
```

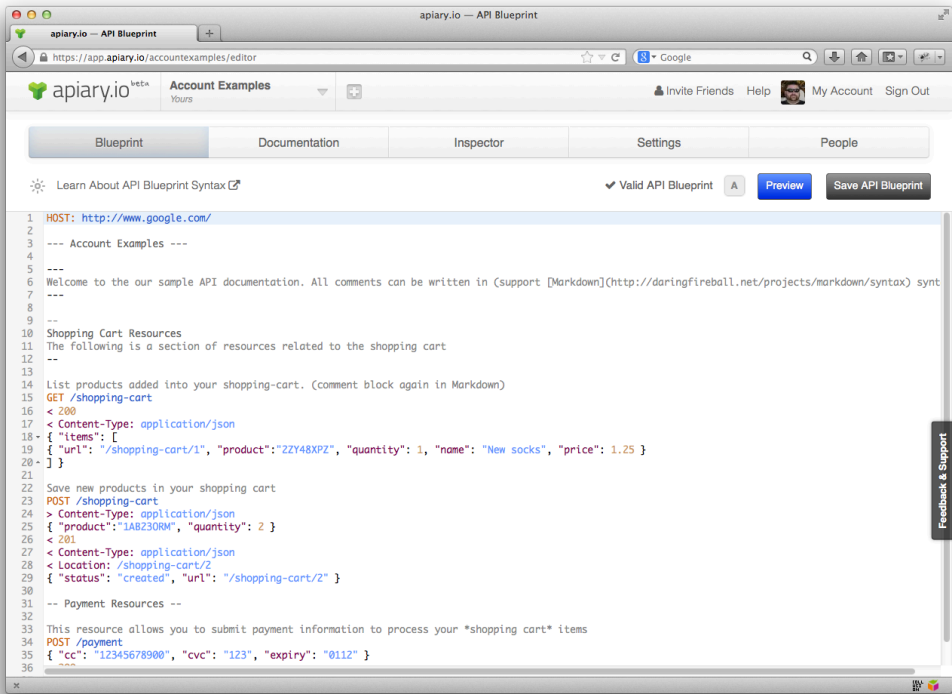
```

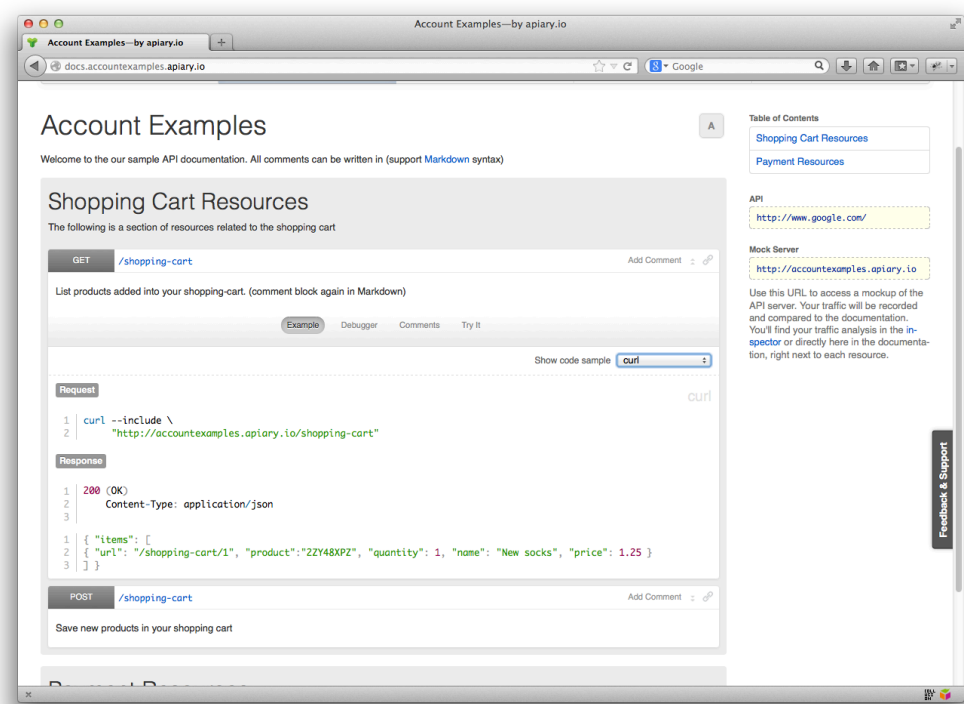
http://example.com/admin?deleteUser=brian

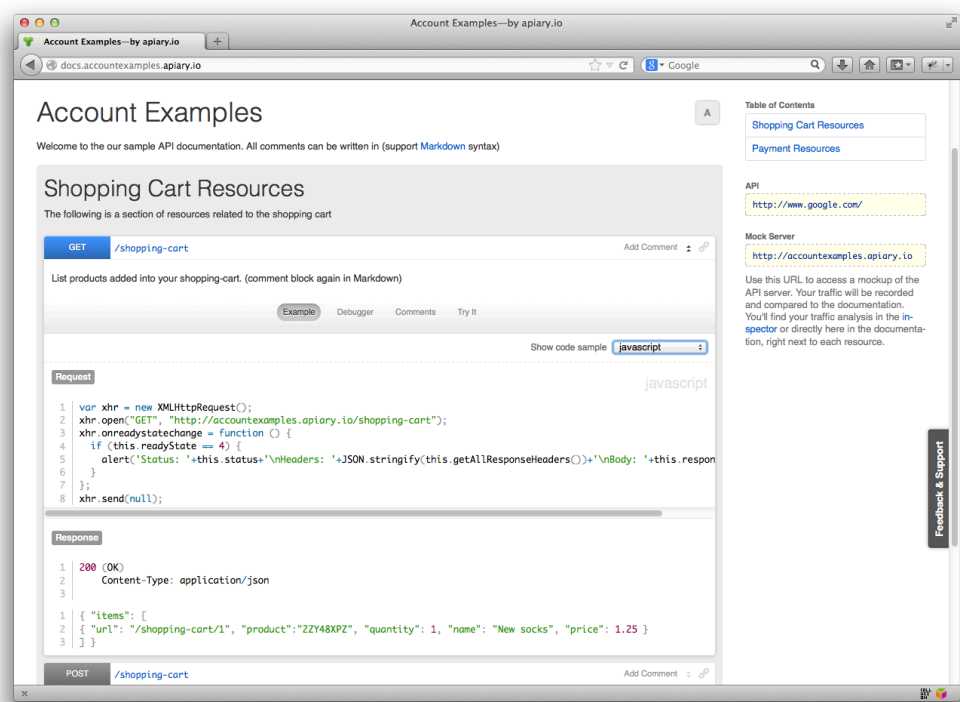
```
<security-constraint>
  <web-resource-collection>
    <url-pattern>/admin/*</url-pattern>
    <http-method>GET</http-method>
    <http-method>POST</http-method>
  </web-resource-collection>
  <auth-constraint>
    <role-name>admin</role-name>
  </auth-constraint>
</security-constraint>
```

apiary.io



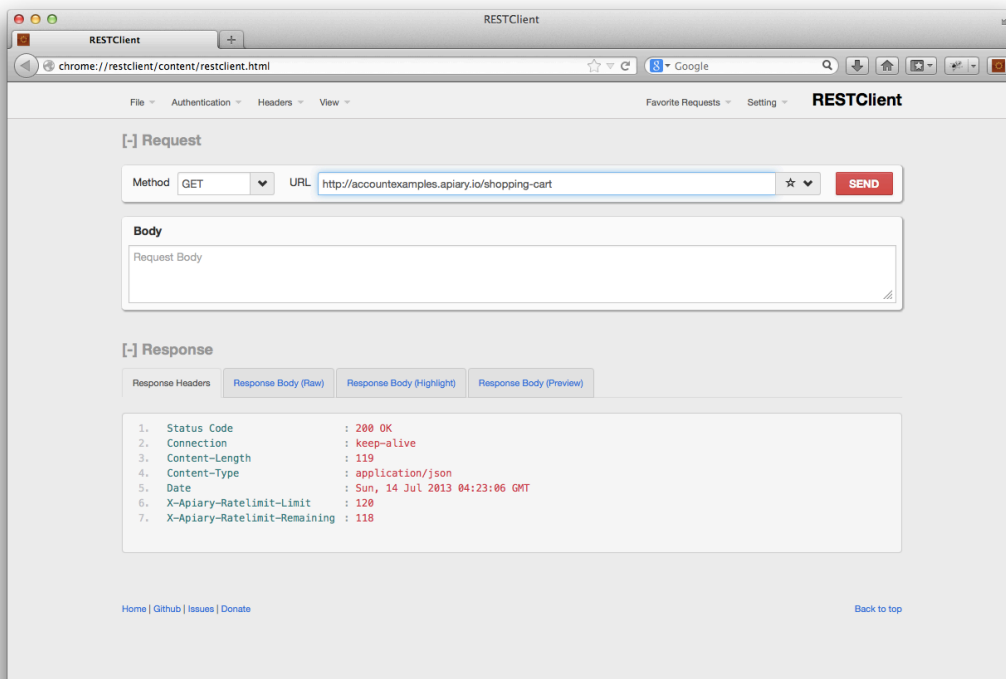


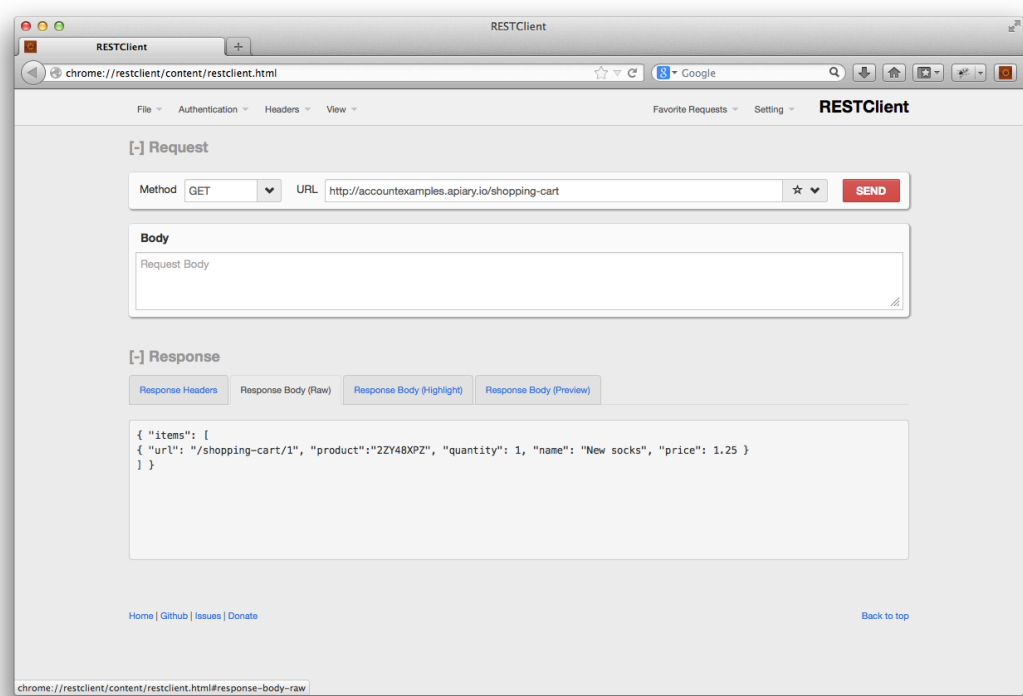


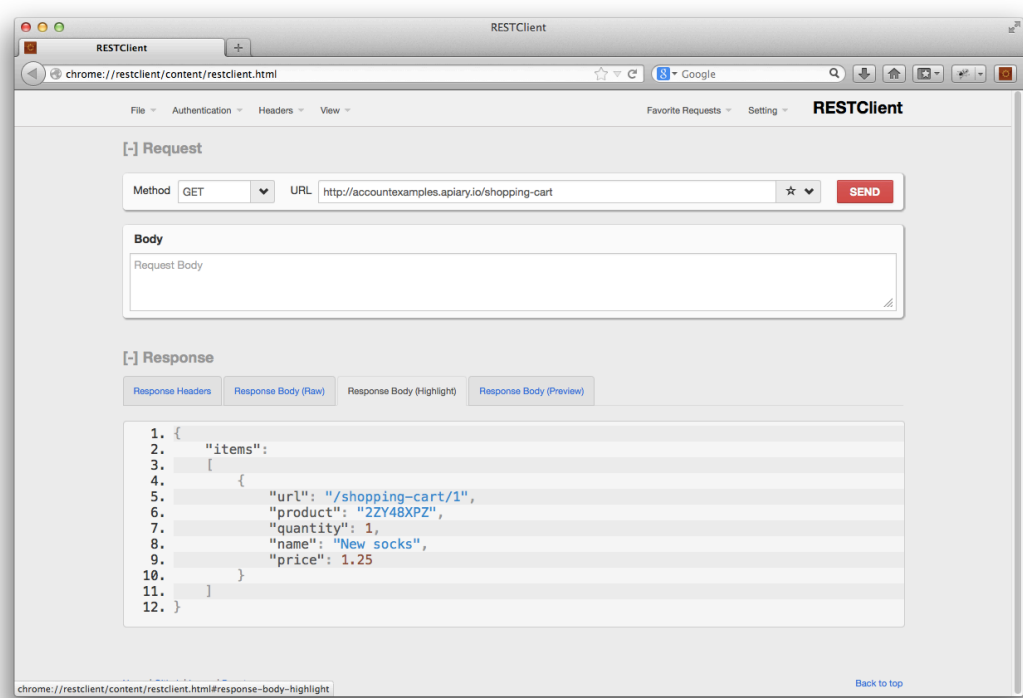


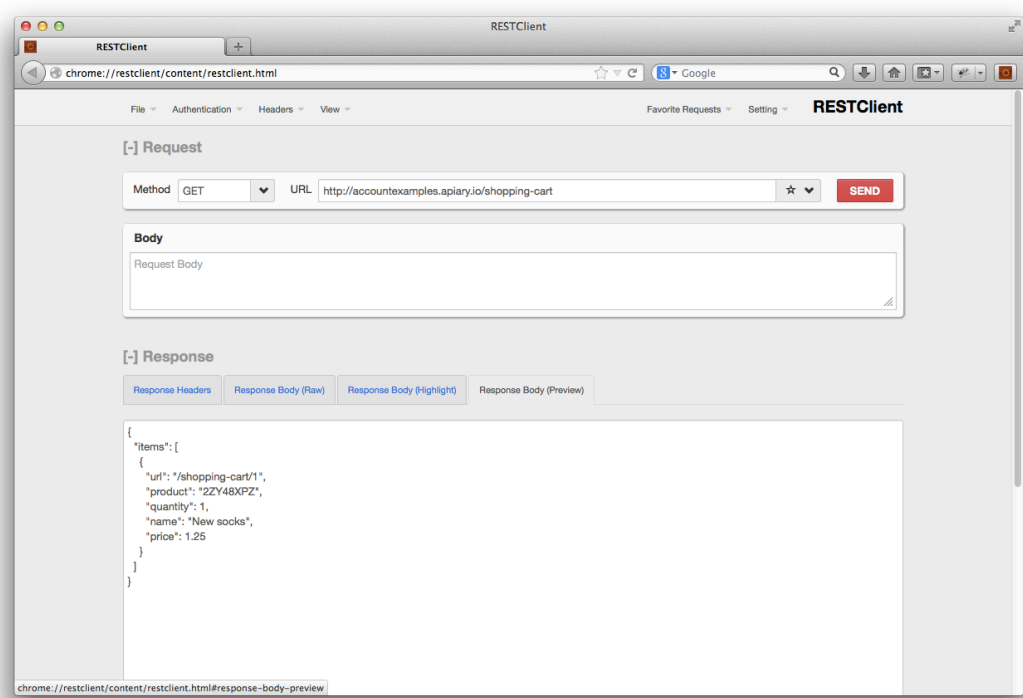
Tools

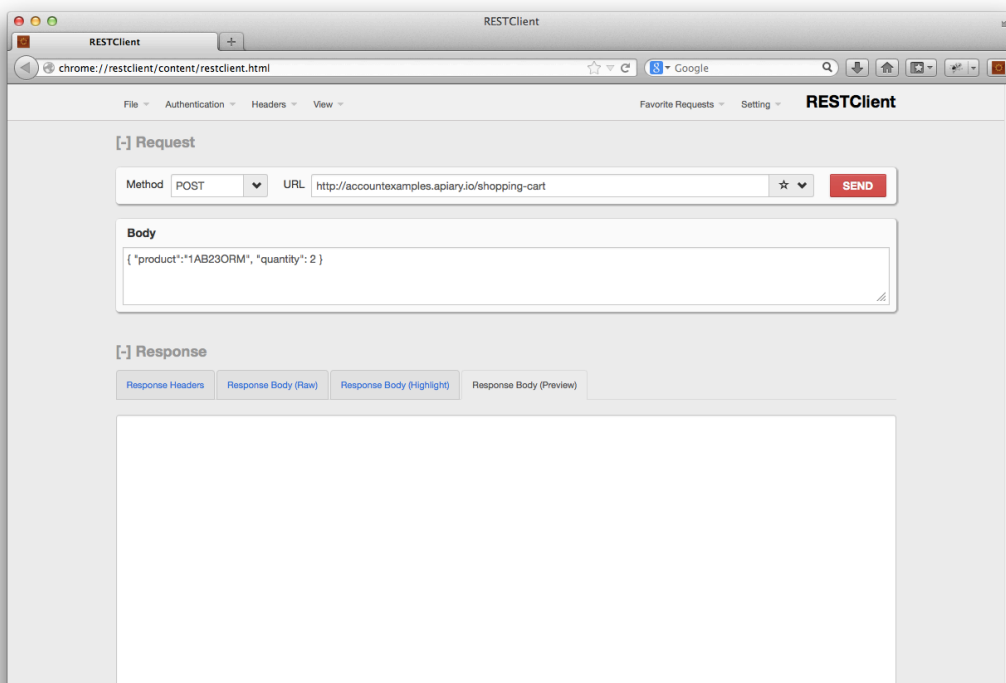
Firefox REST Client

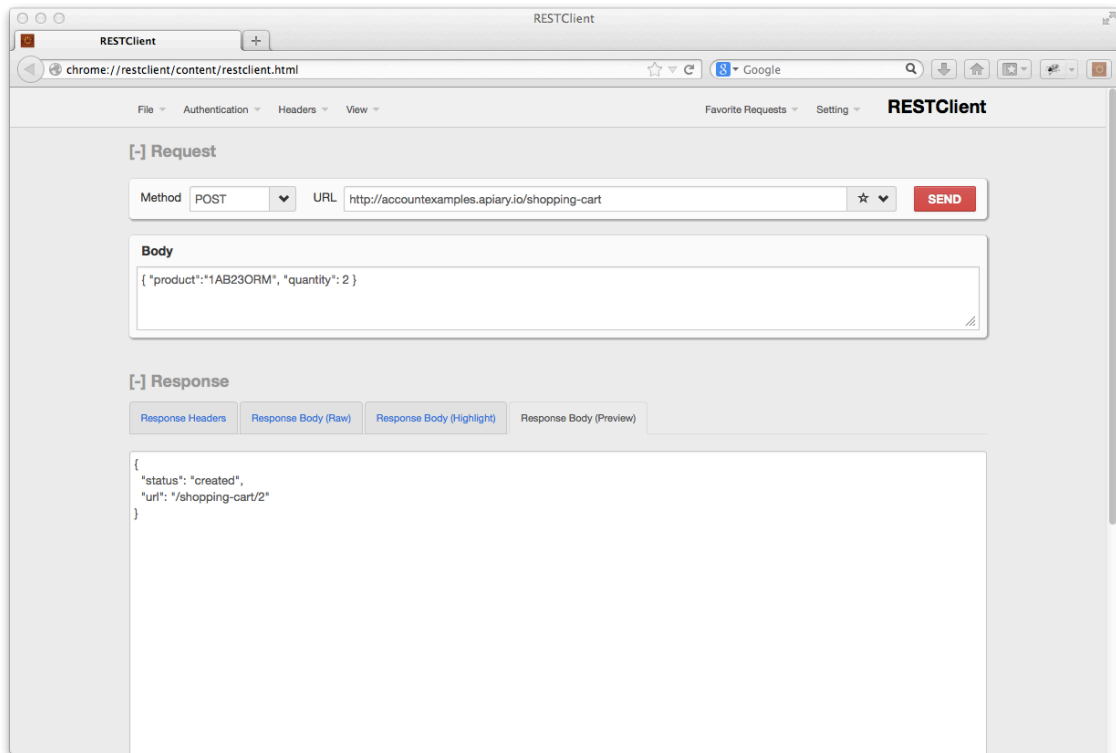












Curl

```
brian@characterzero ~-> curl --include \
    http://accountexamples.apiary.io/shopping-cart
HTTP/1.1 200 OK
Content-Type: application/json
Date: Sun, 14 Jul 2013 05:23:49 GMT
X-APIary-Ratelimit-Limit: 120
X-APIary-Ratelimit-Remaining: 119
Content-Length: 119
Connection: keep-alive

{ "items": [
  { "url": "/shopping-cart/1", "product": "2ZY48XPZ",
    "quantity": 1, "name": "New socks",
    "price": 1.25 }
]}
brian@characterzero ~->
```

```
brian@characterzero ~-> curl --include
--header "Content-Type: application/json" \
--request POST \
--data-binary "{ \"product\": \"1AB230RM\", \"quantity\": 2 }"
http://accountexamples.apiary.io/shopping-cart

HTTP/1.1 201 Created
Content-Type: application/json
Date: Sun, 14 Jul 2013 05:11:03 GMT
Location: http://accountexamples.apiary.io/shopping-cart/2
X-Apiary-Ratelimit-Limit: 120
X-Apiary-Ratelimit-Remaining: 119
Content-Length: 50
Connection: keep-alive

{ "status": "created", "url": "/shopping-cart/2" }
brian@characterzero ~->
```

HTTP Builder

RESTClient

```
import groovyx.net.http.RESTClient
import groovy.util.slurpersupport.GPathResult
import static groovyx.net.http.ContentType.URLENC

twitter = new RESTClient( 'https://twitter.com/statuses/' )
// twitter auth omitted

try { // expect an exception from a 404 response:
    twitter.head path : 'public_timeline'
    assert false, 'Expected exception'
}
// The exception is used for flow control but has access
// to the response as well:
catch( ex ) { assert ex.response.status == 404 }

assert twitter.head( path : 'public_timeline.json' ).status == 200
```

```
def resp = twitter.get( path : 'friends_timeline.json' )
assert resp.status == 200
assert resp.contentType == JSON.toString()
assert ( resp.data instanceof net.sf.json.JSON )
assert resp.data.status.size() > 0
```

```
def msg = "I'm using HTTPBuilder's RestClient on ${new Date()}"

resp = twitter.post( path : 'update.xml',
                    body : [ status:msg, source:'httpbuilder' ],
                    requestContentType : URLENC )

assert resp.status == 200
assert ( resp.data instanceof GPathResult ) // parsed using XmlSlurper
assert resp.data.text == msg
assert resp.data.user.screen_name == userName
def postID = resp.data.id.toInteger()
```

```
resp = twitter.delete( path : "destroy/${postID}.json" )
assert resp.status == 200
assert resp.data.id == postID
println "Test tweet ID ${resp.data.id} was deleted."
```



```
client = new RestClient(url)

client.parser."application/vnd.collections+json" = { resp ->
  def r = new InputStreamReader(resp.entity.content,
    ParserRegistry.getCharset(resp))
  def slurper = new JsonSlurper()
  slurper.parse(r)
}

client.parser."application/vnd.collections+xml" = { resp ->
  def r = new InputStreamReader(resp.entity.content,
    ParserRegistry.getCharset(resp))
  def slurper = new XmlSlurper()
  slurper.parse(r)
}
```

```
resp = client.get(['path' : '/api'])
def rep = resp.data

assert rep.collection.version == '1.0'
assert rep.collection.href == '/api'
assert rep.collection.links.size() == 3

["account", "order", "product"].each { t ->
  assert rep.collection.links.find {
    it['rel'] == t && it['href'] == "/$t"
  }
}
```

- Set up an apiary.io account
- Using curl, Firefox REST Client or HTTPBuilder RESTClient, investigate the default resources

Exercises

Evaluate the REST APIs from:

- Flickr
- Twitter
- Linked In
- NetFlix
- Atom Publishing Protocol

Exercises

For the domain you chose earlier, identify:

- Roles
- A mapping of roles to resources
- A mapping of verbs to resource per role
- Canonical vs historical views of resources
- Candidates for bookmarking

Error-Handling

Client Errors

- 400 Bad Request
- 403 Forbidden
- 404 Not found
- ...
- 500 Internal Server Error
- 503 Service Unavailable

Application-Level Error Message

- Standardize an error message response
- Non-200 response codes

```
HTTP/1.1 400 Bad Request
Content-Type: application/xml;charset=UTF-8
Link: <http://server/error/someexplanation>

<error>
  <message>You have made an invalid request...
</message>
</error>
```

Representation Design

Simple Commerce Engine

- User accounts
- Items
- Orders
- Workflow

Guidelines

- Don't reflect implementation details
- Focus on what makes sense for the domain
- Consider the future in designs
- Don't constrain too early

```
http://someserver/account/12345
```

```
<?xml version="1.0" encoding="utf-8"?>
<account>
  <type>3</type>
  <firstname>Bob</firstname>
  <lastname>Jones</lastname>
  <address>
    <first>12345 Main St.</first>
    <second>Apt. #234</second>
    <city>Minneapolis</city>
    <state>MN</state>
    <zipcode>55413</zipcode>
  </address>
  <email>bob@example.com</email>
</account>
```

URLs as IDs

`http://someserver/account/12345`

`http://someserver/account/new`

`http://someserver/account/best`

`http://someserver/account/id/12345`

Don't put sensitive information in URLs!

Identifying the Resource

```
<account id="12345">
  <link rel="self"
        href="http://someserver/account/id/12345"/>
  ...
</account>
```

Keep it Meaningful

```
<type>3</type>
```

```
<type>priority</type>
```

Clustering Related Information

```
<firstname>Bob</firstname>  
<lastname>Jones</lastname>
```

```
<name>  
  <firstname>Bob</firstname>  
  <lastname>Jones</lastname>  
</name>
```


Plan Ahead

```
<address>
  <first>12345 Main St.</first>
  <second>Apt. #234</second>
  <city>Minneapolis</city>
  <state>MN</state>
  <zipcode>55413</zipcode>
</address>
```

Plan Ahead

```
<addresses>
  <address name="home" default="true">
    <first>12345 Main St.</first>
    <second>Apt. #234</second>
    <city>Minneapolis</city>
    <state>MN</state>
    <zipcode>55413</zipcode>
  </address>
</addresses>
```

Functional Decomposition

```
<email>bob@example.com</email>
```

```
<contact>
  <email primary="true">bob@example.com</email>
  <phone sms="true">555-1212</phone>
  <twitter>bjones9999</twitter>
</contact>
```

Cleaned Up Resource

```
<?xml version="1.0" encoding="utf-8"?>
<account id="12345">
  <link rel="self"
    href="http://someserver/account/id/12345"/>
  <type>priority</type>
  <name>
    <firstname>Bob</firstname>
    <lastname>Jones</lastname>
  </name>
  <addresses>
    <address name="home" default="true">
      <first>12345 Main St.</first>
      <second>Apt. #234</second>
      <city>Minneapolis</city>
      <state>MN</state>
      <zipcode>55413</zipcode>
    </address>
  </addresses>
  <contact>
    <email>bob@example.com</email>
  </contact>
</account>
```

```
http://someserver/item/status/new
```

```
http://someserver/item/status/sale
```

```
http://someserver/item/id/1111
```

```
http://someserver/item/id/1111
```

```
<?xml version="1.0" encoding="utf-8"?>
<item id="1111">
  <link rel="self" href="http://someserver/item/id/1111"/>
  <name>Deadwood Season 4</name>
  <description>Deadwood's most popular season.</description>
  <price>$32.00</price>
  <image>http://someserver/content/image/1111</image>
  <sku>1111</sku>
  <upc>1234123412431234123</upc>
</item>
```

Internationalization

```
<description>
  Deadwood's most popular season.
</description>
```

```
<description lang="en_US">
  Deadwood's most popular season.
</description>
```

Internationalization

```
<price>$32.00</price>
```

Internationalization

```
<price currency="USD">32.00</price>
```

Multiple IDs

```
<sku>1111</sku>
```

```
<upc>1234123412431234123</upc>
```

Multiple IDs

```
<sku>1111</sku>
```

```
<upc>1234123412431234123</upc>
```

```
http://someserver/item/id/sku/1111
```

```
http://someserver/item/id/upc/1234123412431234123
```

```
http://someserver/order/status/new
```

```
http://someserver/order/status/open
```

```
http://someserver/order/id/101-545
```

```
http://someserver/order/cust/12345
```

`http://someserver/order/id/101-545`

```
<?xml version="1.0" encoding="utf-8"?>
<order id="101-545">
  <link rel="self" href="http://someserver/order/id/101-545"/>
  <items>
    <item id="1111">
      <price currency="USD">32.00</price>
    </item>
    <item id="2222">
      <price currency="USD">10.00</price>
    </item>
  </items>
  <addresses>
    <address shipping="true" billing="true">
      <first>12345 Main St.</first>
      <second>Apt. #234</second>
      <city>Minneapolis</city>
      <state>MN</state>
      <zipcode>55413</zipcode>
    </address>
  </addresses>
</order>
```

`application/vnd.acme.account+xml`

`application/vnd.acme.item+xml`

`application/vnd.acme.order+xml`

```
<?xml version="1.0" encoding="utf-8"?>
<order id="101-545">
  <link rel="self" href="http://someserver/order/id/101-545"/>
  <items>
    <item id="1111">
      <price currency="USD">32.00</price>
    </item>
    <item id="2222">
      <price currency="USD">10.00</price>
    </item>
  </items>
</order>
```

```
<?xml version="1.0" encoding="utf-8"?>
<order id="101-545">
  <link rel="self" href="http://someserver/order/id/101-545"/>
  <link rel="submit" href="http://someserver/order/id/101-545/submit"/>
  <items>
    <item id="1111">
      <price currency="USD">32.00</price>
    </item>
    <item id="2222">
      <price currency="USD">10.00</price>
    </item>
  </items>
  <addresses>
    <address id="1" shipping="true" billing="true">
      ...
    </address>
  </addresses>
  <tender>
    <credit-card id="1" number="XXXX1344">
    </credit-card>
  </tender>
</order>
```



```
<?xml version="1.0" encoding="utf-8"?>
<order id="101-545">
  <link rel="self" href="http://someserver/order/id/101-545"/>
  <link rel="submit" href="http://someserver/order/id/101-545/submit"/>
  <items>
    <link rel="self" href="http://someserver/order/id/101-545/item"/>
    <item id="1111">
      <price currency="USD">32.00</price>
    </item>
    <item id="2222">
      <price currency="USD">10.00</price>
    </item>
  </items>
  <addresses>
    <link rel="self" href="http://someserver/order/id/101-545/address"/>
    <address id="1" shipping="true" billing="true">
      ...
    </address>
  </addresses>
  <tender>
    <link rel="self" href="http://someserver/order/id/101-545/tender"/>
    <credit-card id="1" number="XXXX1344">
      </credit-card>
    </tender>
  </order>
```

```
<?xml version="1.0" encoding="utf-8"?>
<order id="101-545">
  <link rel="self" href="http://someserver/order/id/101-545"/>
  <link rel="submit" href="http://someserver/order/id/101-545/submit"/>
  <items>
    <item id="1111">
      <link rel="self" href="http://someserver/order/id/101-545/item/1"/>
      <price currency="USD">32.00</price>
    </item>
    <item id="2222">
      <link rel="self" href="http://someserver/order/id/101-545/item/2"/>
      <price currency="USD">10.00</price>
    </item>
  </items>
  <addresses>
    <address id="1" shipping="true" billing="true">
      <link rel="self" href="http://someserver/order/id/101-545/address/1"/>
      ...
    </address>
  </addresses>
  <tender>
    <credit-card id="1" number="XXXX1344">
      <link rel="self" href="http://someserver/order/id/101-545/tender/1"/>
      </credit-card>
    </tender>
  </order>
```

Content Negotiation

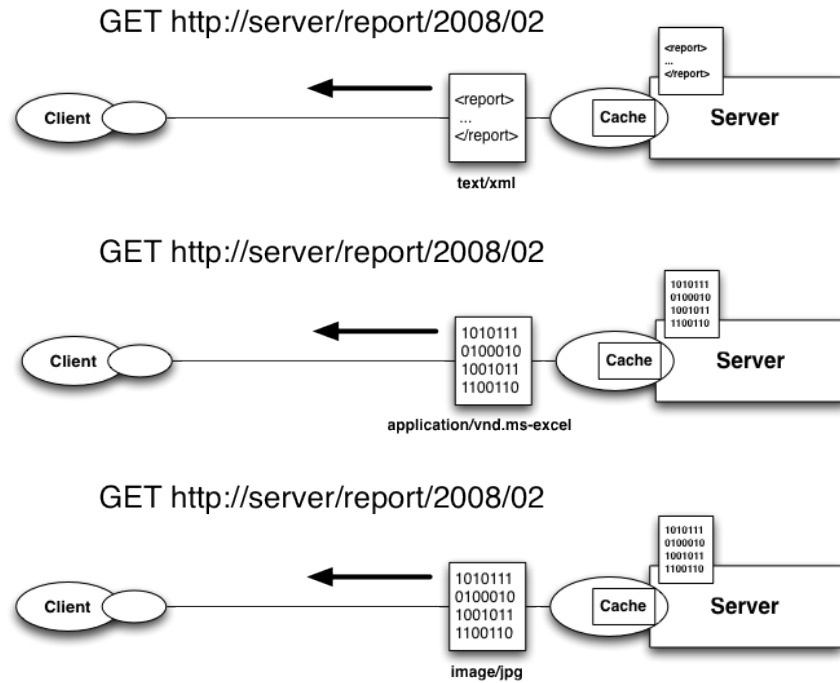
Logically-named Resources

```
http://server/report/2010/01/sales
```

```
http://server/report/2010/01/sales.html
```

```
http://server/report/2010/01/sales.xml
```

```
http://server/report/2010/01/sales.xls
```



Accept

```
Accept: application/xml; q=1.0, application/json; q=0.6, */*; q=0.0
Accept-Language: en;q=1.0, jp;q=0.5
Accept-Encoding: gzip
```

Vary Response Header

```
Vary: Accept-Language  
Vary: Accept
```

Naming Rules

- Don't define resources w/ trailing slashes
- Prefer hyphens to underscores in resource names
- Use lowercase letters
- Don't require filename extensions
- Define subdomains for your APIs

Response Header Rules

- Use Content-Type
- Use Content-Length
- Use Last-Modified
- Use Location for Created Resources

Conditional Requests

- Use ETags
- Support Conditional Requests

Caching

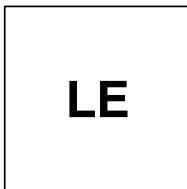
- Encourage Caching
- Use Cache-Control, Expires and Date to Enable Caching
- Use Cache-Control, Expires and Pragma to Disable Caching

Hypermedia

HFactors

Mike Amundsen (@mamund)

Embedded Links (LE)



Dereference specified URI and display within current window.

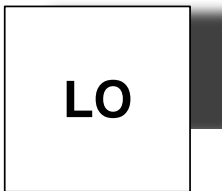
Embedded Links (LE)

```

```

```
<x:include href="http://bosatsu.net/snippets/bio.xml"/>
```

Outbound Links (LO)

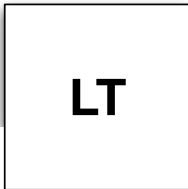


Dereference specified URI and display within new window.

Outbound Links (LO)

```
See you at <a href="http://uberconf.com">Überconf</a>.
```

Templated Links (LT)



Accumulate input parameters into URI definition as part of read operation.

Templated Links (LT) #1

```
<form method="get" action="http://bosatsu.net/submit">  
  <input type="text" name="course" value="REST"/>  
  <input type="submit"/>  
</form>
```

```
http://bosatsu.net/submit?course=REST
```

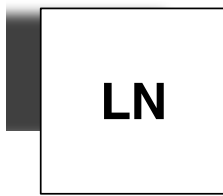
Templated Links (LT) #2

URI Templates (RFC 6570)

```
http://example.com/~{username}/  
http://example.com/dictionary/{term:1}/{term}  
http://example.com/search?q,{lang}
```

```
http://example.com/~fred/  
http://example.com/~mark/  
http://example.com/dictionary/c/cat  
http://example.com/dictionary/d/dog  
http://example.com/search?q=cat&lang=en  
http://example.com/search?q=chien&lang=fr
```

Non-Idempotent Links (LN)



Submit a non-idempotent request.

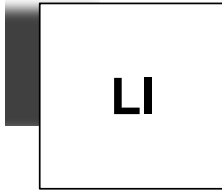
Non-Idempotent Links (LN)

```
<form method="post" action="http://bosatsu.net/submit/">
  <textarea name="feedback">Hey Buddy, nice jacket</textarea>
  <input type="submit"/>
</form>
```

```
POST /submit/ HTTP/1.1
Host: bosatsu.net
Content-Type: application/x-www-form-urlencoded
Length:34

feedback=Hey+Buddy%2C+nice+jacket
```

Idempotent Links (LI)



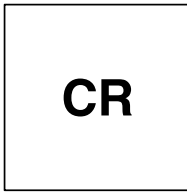
Submit an idempotent request.

Idempotent Links (LI)

```
<link rel="edit" href="http://bosatsu.net/feed/edit/1"/>
```

```
<script type="text/javascript">
function delete(url) {
  var xhr = new XMLHttpRequest();
  xhr.open("DELETE", url);
}
</script>
```

Read Controls (CR)

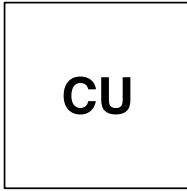


Specify read control context.

Read Controls (CR)

```
<x:include href="http://bosatsu.net/snippets/bio.xml"
  accept-language="en, en-gb; q=0.9, jp;q=0.3"/>
```

Update Controls (CU)



Specify update control context.

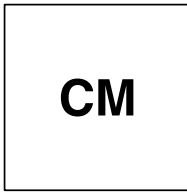
Update Controls (CU)

```
<form method="post" action="http://bosatsu.net/submit/"
  enctype="text/plain">
  <textarea name="feedback">Hey Buddy, nice jacket</textarea>
  <input type="submit"/>
</form>
```

```
POST /submit/ HTTP/1.1
Host: bosatsu.net
Content-Type: text/plain
Length:34

Hey+Buddy%2C+nice+jacket
```

Method Controls (CM)



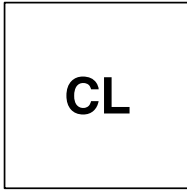
Specify the protocol method for the request.

Method Controls (CM)

```
<form method="post" action="/feedback">  
  <input name="keywords" type="text" value="foo,bar,baz"/>  
  <input type="submit"/>  
</form>
```

```
<form method="get" action="/feedback">  
  <input name="keywords" type="text" value="foo,bar,baz"/>  
  <input type="submit"/>  
</form>
```

Link Controls (CL)



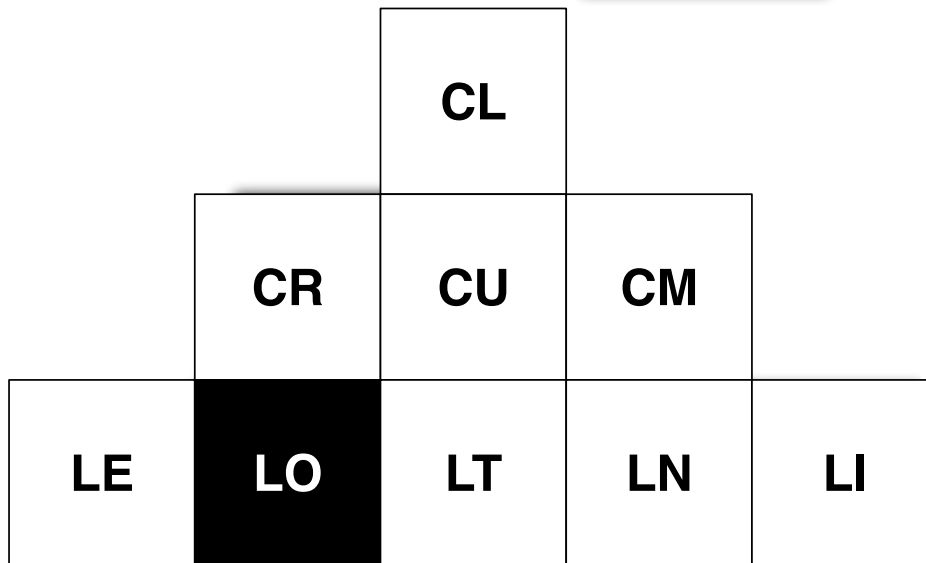
Specify link control context.

Link Controls (CL)

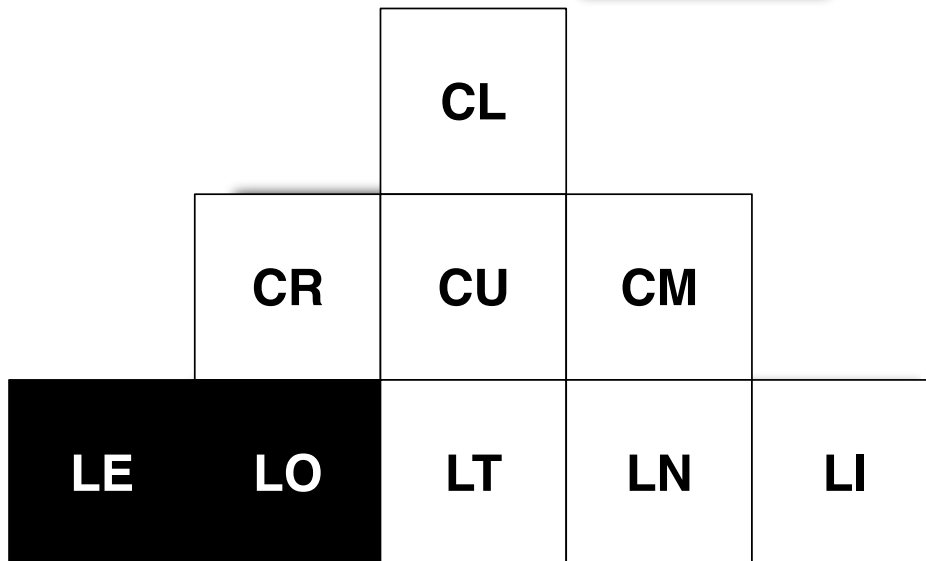
```
<entry>
  <title>The Bully</title>
  <link rel="edit" href="http://bosatsu.net/feed/1"/>
  <id>tag:bosatsu.net,2013:3.2397</id>
  <updated>2013-03-01T12:29:29Z</updated>
  <published>2013-02-13T08:29:29-04:00</published>
  <author>
    <name>Brian Sletten</name>
  </author>
  <content>Hey Buddy, nice car.</content>
</entry>
```

```
<link rel="stylesheet" href="http://bosatsu.net/css/default.css"/>
```

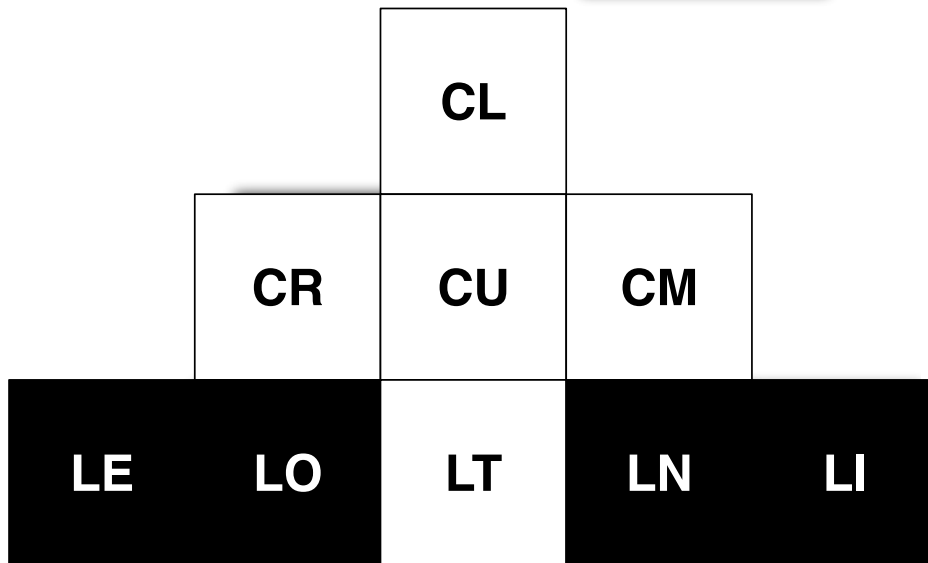

text/uri-list



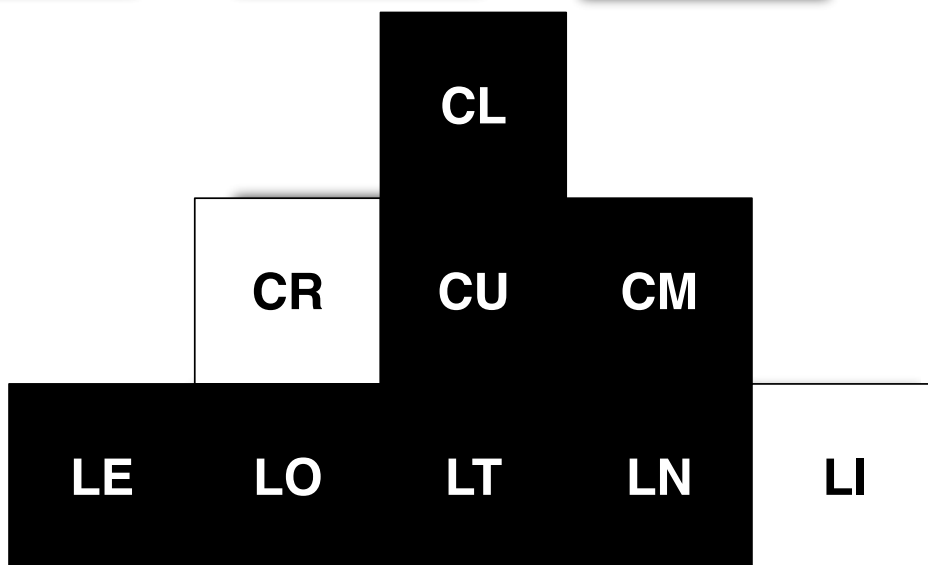
application/svg+xml



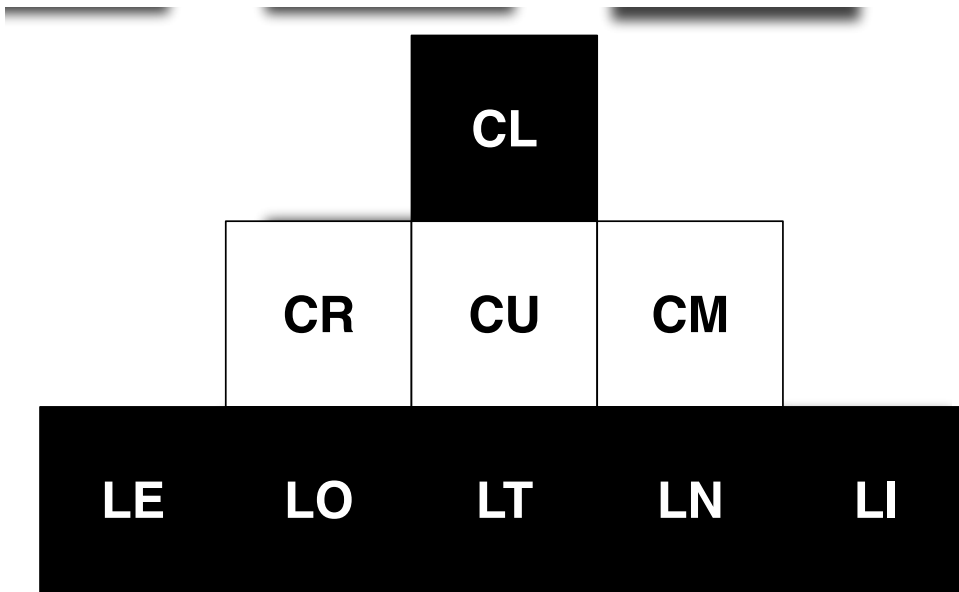
application/atom+xml



text/html



application/vnd.collections+json



```
{ "collection" :  
  {  
    "version" : "1.0",  
    "href" : "http://example.org/friends/"  
  }  
}
```

```
{ "collection" :  
  {  
    "version" : "1.0",  
    "href" : "http://example.org/friends/",  
    "links" : [],  
    "items" : [],  
    "queries" : [],  
    "template" : []  
  }  
}
```

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

```
{  
  "collection": {  
    "version": "1.0",  
    "href": "/api",  
    "links": [  
      {  
        "rel": "account",  
        "href": "/account"  
      },  
      {  
        "rel": "order",  
        "href": "/order"  
      },  
      {  
        "rel": "product",  
        "href": "/product"  
      }  
    ]  
  }  
}
```

`http://example.org/account`

`http://example.org/order`

`http://example.org/product`

`http://example.org/account`

```
{ "collection" :  
  {  
    "version" : "1.0",  
    "href" : "/account",  
    "links" : [  
      {  
        "rel": "next",  
        "href": "/account;page=2"  
      }  
    ],  
    "items" : [],  
    "queries" : [],  
    "template" : []  
  }  
}
```

```
http://example.org/account;page=2
```

```
{ "collection" :  
  {  
    "version" : "1.0",  
    "href" : "/account;page=2",  
    "links" : [  
      {  
        "rel": "prev",  
        "href": "/account;page=1"  
      },  
      {  
        "rel": "next",  
        "href": "/account;page=3"  
      }  
    ],  
    "items" : [],  
    "queries" : [],  
    "template" : []  
  }  
}
```

```
http://example.org/account
```

```
{ "collection" :  
  {  
    ...  
    "items" : [],  
    ...  
  }  
}
```

http://example.org/account

```
...
  "items": [
    {
      "href": "/account",
      "data": [
        {
          "name": "type",
          "value": "account"
        },
        {
          "name": "total",
          "value": 37
        },
        {
          "name": "page",
          "value": 1
        },
        {
          "name": "ipp",
          "value": 10
        }
      ]
    }
  ],
  ...
```

http://example.org/account

```
...
{
  "href": "/account/id/9468",
  "data": [
    {
      "name": "username",
      "value": "bob"
    },
    {
      "name": "id",
      "value": "9468"
    }
  ],
  "links": [
    {
      "name": "open",
      "value": "/order/account/id/9468;status=open"
    },
    {
      "name": "recent",
      "value": "/order/account/id/9468;status=recent"
    }
  ]
},
...
```

http://example.org/account

```
{ "collection" :
{
  ...
  "queries" : [
    {
      "encoding": "uri-template",
      "rel": "search",
      "href": "/account{;status,page,ipp}",
      "prompt": "Search",
      "data": [
        {
          "name": "status",
          "value": [
            ]
          },
        {
          "name": "page",
          "value": [
            ]
          },
        {
          "name": "ipp",
          "value": [
            ]
          }
        ]
      }
    }
  ],
  ...
}
```

http://localhost:8080/wink/wiki/bosatsu/uberconf/2013/rest-workshop#

Page 207 of 229

http://example.org/account;status=open;page=2

```
{ "collection" :
{
  ...
  "queries" : [
    {
      "encoding": "uri-template",
      "rel": "search",
      "href": "/account{;status,page,ipp}",
      "prompt": "Search",
      "data": [
        {
          "name": "status",
          "value": [
            "open"
          ]
        },
        {
          "name": "page",
          "value": [
            "2"
          ]
        },
        {
          "name": "ipp",
          "value": [
            ]
          }
        ]
      }
    }
  ],
  ...
}
```

http://localhost:8080/wink/wiki/bosatsu/uberconf/2013/rest-workshop#

Page 208 of 229

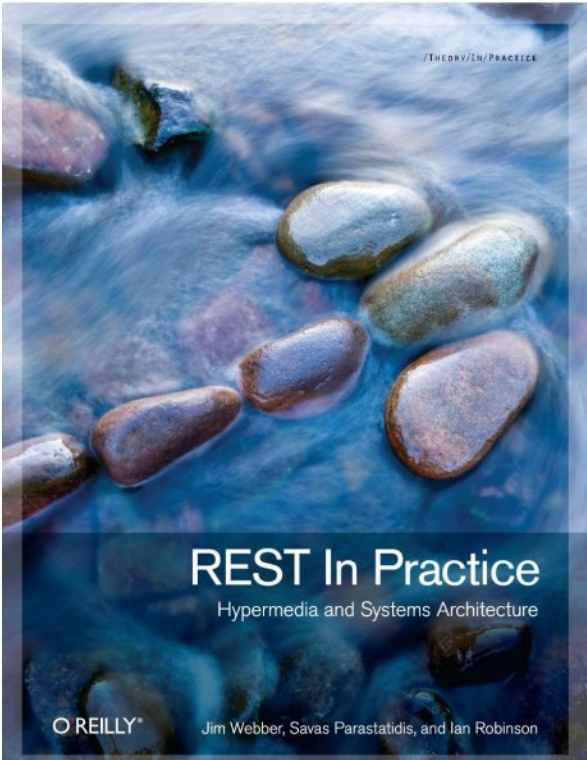
Exercises

- Investigate HAL or SIREN
- Design three representations from our previous exercises
- Discuss your choice of MIME type (custom vs reusable)
- Summarize which HFactors you used
- Try to make links discoverable using RFC 6570
- Describe your error-handling strategy
- Describe your versioning strategy

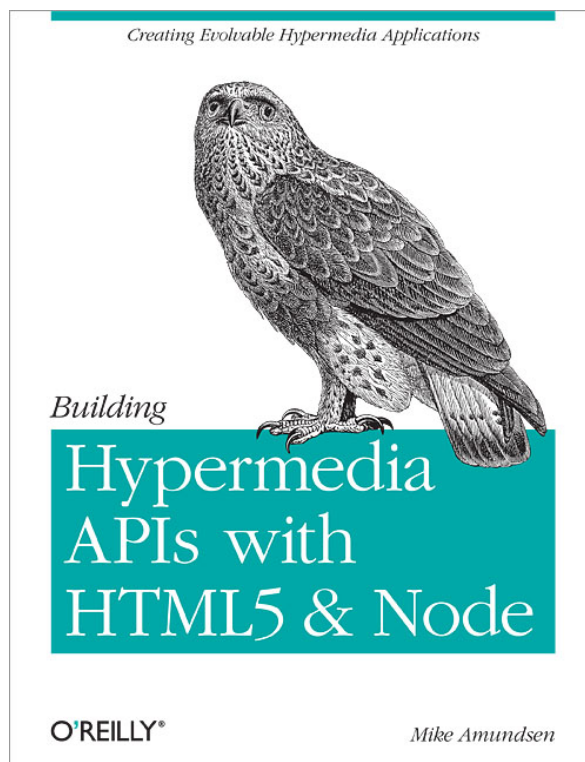
Books

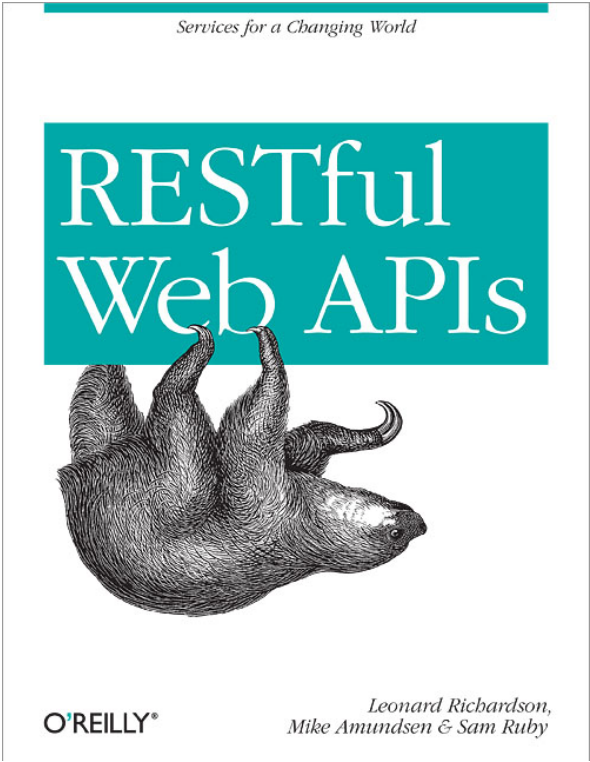














Questions?



 **@bsletten** (<http://twitter.com/bsletten>)  **brian@bosatsu.net** (<mailto:brian@bosatsu.net>)