

How Much REST Do You Need?

PHPUGMS 2016

Tobias Schlitt / @tobySen
2016-07-19



About Me

Tobias Schlitt



<https://qafoo.com>

- ▶ Legacy code?
- ▶ Too many bugs?
- ▶ Performance issues?
- ▶ Architecture challenges?

Get in touch:
toby@qafoo.com

REST?




There is no REST

Choose your degree of RESTfulness

Constraints by Example

- ▶ Time & money
- ▶ Project environment
- ▶ Consumers
- ▶ API weight
- ▶ Architecture requirements
 - ▶ Scalability
 - ▶ Simplicity
 - ▶ Backwards compatibility
 - ▶ ...

Example Scenarios

- ▶  Project Internal
 - ▶ Image processing
 - ▶ Mobile
 - ▶ Central account service
- ▶  SAAS API
 - ▶ Your Flickr clone
 - ▶ Tideways (<http://tideways.io>)
- ▶  Enterprise Environment
 - ▶ Global logistics
 - ▶ Financial institution

⚡ Disclaimer!

The examples used in this talk will **never** fit your individual project exactly. Do not apply the opinionated ratings provided by the speaker to your project directly. Instead, please weight your own trade-offs. Thanks.

Outline

Basics

HATEOAS

Versioning

Caching

Authentication

Conclusion

Resources

- ▶ Entities or collections
- ▶ Tree structure
- ▶ URI as **unique** identifier
- ▶ Examples:
 - ▶ `http://plan.qafoo.com/users`
 - ▶ `/users/toby`
 - ▶ `//jobs/23`
 - ▶ `../`

Evaluation: How much REST? ⚡

☰ Project Internal ++

☁ SAAS API ++

🚀 Enterprise Environment ++

HTTP Methods

- ▶ GET
- ▶ HEAD
- ▶ OPTIONS
- ▶ TRACE
- ▶ POST
- ▶ PUT
- ▶ DELETE
- ▶ ...

Evaluation: How much REST? ⚡

☰ Project Internal + (GET/POST)

☁ SAAS API ++

🦋 Enterprise Environment ++

Outline

Basics

HATEOAS

Versioning

Caching

Authentication

Conclusion

Media Types

- ▶ Assign Semantic
- ▶ Drive Application State

`application/psr.com.qafoo.plan-job+xml; charset=UTF-8`

HATEOAS

GET /job/23

```
1 <?xml version="1.0"?>
2 <job xmlns="urn:psr.com.qafoo.plan-job"
3   xmlns:atom="http://www.w3.org/2005/Atom">
4   <!-- ... -->
5   <atom:link rel="urn:psr.com.qafoo.plan-job-assignments"
6     type="application/psr.com.qafoo.plan-job-assignment-list+xml"
7     href="/jobs/23/assignments" />
8 </job>
```

...

POST /job/23/assignments

```
1 <?xml version="1.0"?>
2 <assignment xmlns="urn:psr.com.qafoo.plan-job-assignment"
3   xmlns:atom="http://www.w3.org/2005/Atom">
4   <atom:link rel="urn:psr.com.qafoo.plan-assignee"
5     type="application/psr.com.qafoo.plan-user+xml"
6     href="/users/benjamin" />
7   <days>2</days>
8 </assignment>
```

Evaluation: How much REST? ⚡

☰ Project Internal +/-

☁ SAAS API +

🚀 Enterprise Environment ++

Outline

Basics

HATEOAS

Versioning

Caching

Authentication

Conclusion

Versioning with Media Types

- ▶ `application/....plan-job+xml`
- ▶ `application/....plan-job-v2+xml`
- ▶ `application/....plan-job+xml; version="2"`

Content Negotiation

Accept:

- ▶ application/....plan-job+xml
- ▶ application/....plan-job+json,
application/....plan-job+xml; q=0.5
- ▶ application/....plan-job+xml; version="2",
application/....plan-job+xml; q=0.5,
application/*; q=0.2,
/; q=0.1

Evaluation: How much REST? ⚡

☰ Project Internal -

☁ SAAS API +

🚀 Enterprise Environment +

Outline

Basics

HATEOAS

Versioning

Caching

Authentication

Conclusion

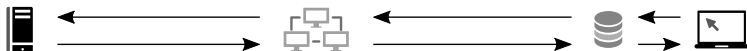
There is no Real-Time

- ▶ Given 10 req/sec
- ▶ 5 seconds caching
- ▶ saves 49 calculations

Caching in HTTP

- ▶ Shared
- ▶ Private

- ▶ Expiration
- ▶ Validation



“There are only two hard things in Computer Science: cache invalidation and naming things.”
– Phil Karlton

Cachability

- ▶ “Sensible defaults”
 - ▶ Method (GET/HEAD)
 - ▶ Request headers
 - ▶ Response status
- ▶ Origin server
 - ▶ Cache-Control:
 - ▶ `public / private`
 - ▶ `no-cache / no-store`
- ▶ Client
 - ▶ Cache-Control:
 - ▶ `no-cache / no-store`

Expiry

- ▶ Origin server
 - ▶ Date:
 - ▶ Expires:
 - ▶ Cache-Control:
 - ▶ max-age=<seconds>
 - ▶ s-maxage=<seconds>
- ▶ User agent
 - ▶ Cache-Control:
 - ▶ max-age
 - ▶ min-fresh
 - ▶ max-stale
 - ▶ Attention: Expiry heuristics




Validation

- ▶ Origin server
 - ▶ ETag:
 - ▶ Last-Modified:
- ▶ Client
 - ▶ If-None-Match:
 - ▶ If-Modified-Since:
- ▶ Optimistic locking!

Caching ...

- ▶ Even more complicated ...
 - ▶ forced re-validation
 - ▶ range caching
 - ▶ Vary
 - ▶ stale handling
 - ▶ PURGE method
- ▶ Be aware of eventual consistency

Evaluation: How much REST? ⚡

 Project Internal	-
 SAAS API	++
 Enterprise Environment	+

Outline

Basics

HATEOAS

Versioning

Caching

Authentication

Conclusion

Statelessness

- ▶ REST = stateless
- ▶ All information must be in request

HTTP Workflow

- ▶ → POST /jobs without auth
- ▶ ← 401 Unauthorized with WWW-Authenticate
- ▶ → POST /jobs with auth for *anonymous*
- ▶ ← 401 Unauthorized with WWW-Authenticate
- ▶ → POST /jobs with auth for *toby*
- ▶ ← 201 Created

Basic / Digest Auth

- ▶ HTTP default auth methods
- ▶ Basic
 - ▶ → Some request
 - ▶ ← WWW-Authenticate: Basic realm="My API"
 - ▶ → Authorization: Basic dG9ieTpxYWZvbW==
- ▶ Digest
 - ▶ Hashing with server provided nonce
 - ▶ Slightly more secure (but not enough!)
- ▶ Use HTTPS with Basic and Digest!

API Key

- ▶ Authenticate an application
- ▶ Not a user
 - ▶ Shared secret exchange
 - ▶ Cryptographic signing
 - ▶ `hash_hmac()`
- ▶ Custom WWW-Authenticate / Authorization format
- ▶ Use HTTPS with API-key!

API Key: JWS/JWT

- ▶ Attempt for standardization
- ▶ JSON Web Signature <http://qa.fo/jws>
- ▶ JSON Web Token <http://qa.fo/jwt>

OAuth2

- ▶ Authenticate users 3rd party apps
 - ▶ e.g. Twitter / Facebook / ...
- ▶ Allows fine-grained permission system
 - ▶ Read personal information
 - ▶ Read friend list
 - ▶ Post as user
 - ▶ ...
- ▶ <http://oauth.net/2/>
- ▶ OAuth2 requires HTTPS!

Evaluation: How much REST? ⚡

 Project Internal	++	API key / Digest
 SAAS API	++	OAuth / API Key
 Enterprise Environment	++	API Key

Outline

Basics

HATEOAS

Versioning

Caching

Authentication

Conclusion

There is no REST

- ▶ There is no (pure) REST
- ▶ You must make **trade-offs**
<http://qa.fo/trade-offs>
- ▶ Decide wisely
 - ▶ REST architecture attributes
 - ▶ Project scope
 - ▶ Consumers
 - ▶ ...
- ▶ Questions?

<https://talks.qafoo.com>



THANK YOU

Rent a quality expert
qafoo.com