TRASYS

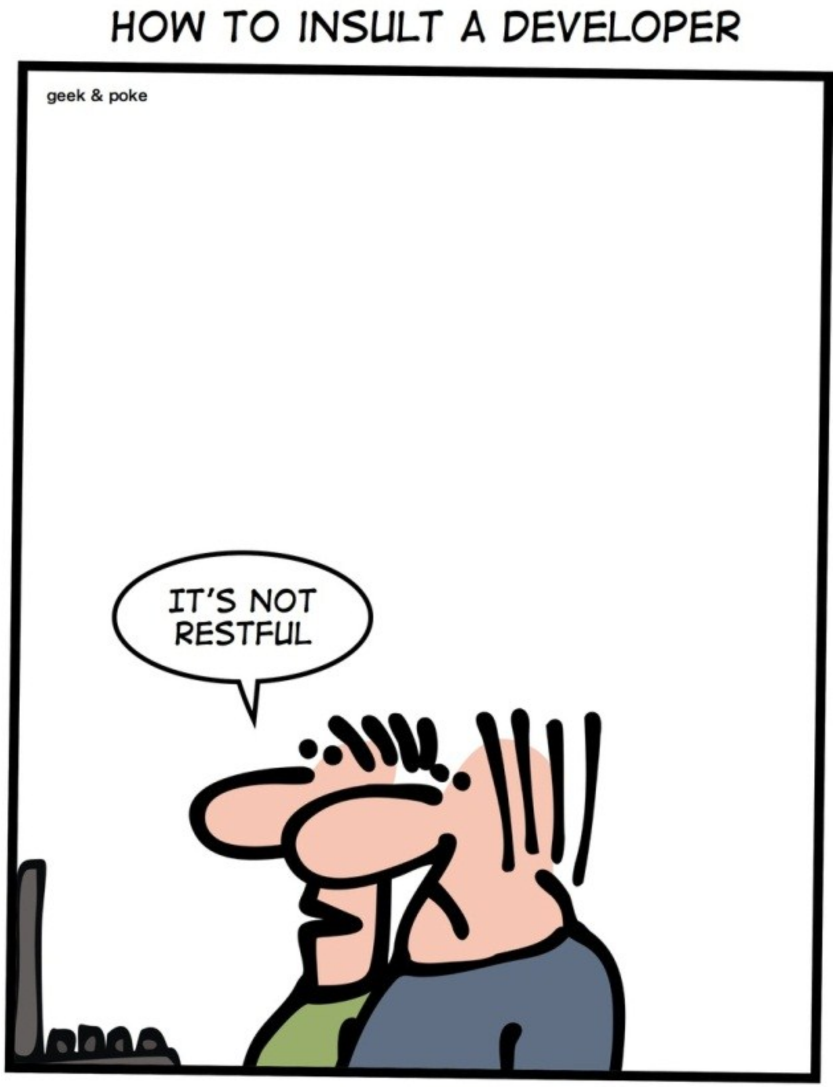
WE GET IT DONE

# Error Handling in REST

Kostis Kapelonis

Athens Greece, November 2013

# Classic comic (original version)



# A little history

# A little history

1991

HTTP 0.9 documented

# A little history

1996

HTTP 1.0 released

Production Ready

Defined in RFC 1945

# A little history

## 1996

HTTP 1.0 team includes Roy Fielding (and Tim Berners Lee)

# Roy Tomas Fielding

Roy was part of the HTTP 1.0 Team

# A little history

## 1999

HTTP 1.1 released

Not really groundbreaking

Defined in RFC 2616

# Roy Tomas Fielding

Roy was part of the HTTP 1.1 Team

# A little history

## 2000

A PHD dissertation is released. (Probably the most hyped over the last decade)

# A little history

2000

“Architectural Styles and the Design of Network-based Software Architectures”

# A little history

2000

The REST Architectural style (Representational state transfer)

# Roy Tomas Fielding

Roy wrote this PHD!

# Roy Tomas Fielding

This guy is important!

He has written the “book”

# A little history

## Conclusion A

In 1996 – 1999 all HTTP apps were NOT restful (by definition)

# A little history

## 1999-2000

Assumption: HTTP 1.1 was affected by REST PHD

# A REST application

- Client – Server
- Stateless
- Cacheable
- Layered system
- Code on demand (What?)
- Uniform interface

# REST versus HTTP

## Quiz 1

How REST applies to HTTP?

# Rest versus HTTP

## Answer

Last chapter (6/6)

6.3 “REST Applied to HTTP”

# Rest versus HTTP

## Conclusion B

REST is independent of HTTP (although it can be applied to it)

# Rest versus HTTP

## Quote

“A REST API should not be dependent on any single communication protocol”

# Roy Tomas Fielding

Roy said it in his blog!

http://roy.gbiv.com/untangled/2008/rest-apis-must-be-hypertext-driven

# Rest versus HTTP

## Conclusion C (A+B)

REST is independent from HTTP. HTTP is independent from REST. (but they can be used together)

# Standards everywhere

- HTTP 1.0 (RFC 2616)
- SSH (RFC 4251)
- FTP (RFC 959)
- HTML (REC-html401-19991224)
- CSS (REC-CSS2-20080411)
- IMAP (RFC 3051)

# Conclusion D

There is NO official Specification for REST

# What is REST

## Conclusion?

Maybe the HTTP 1.1 is the spec?

# What is REST

## Quiz 2

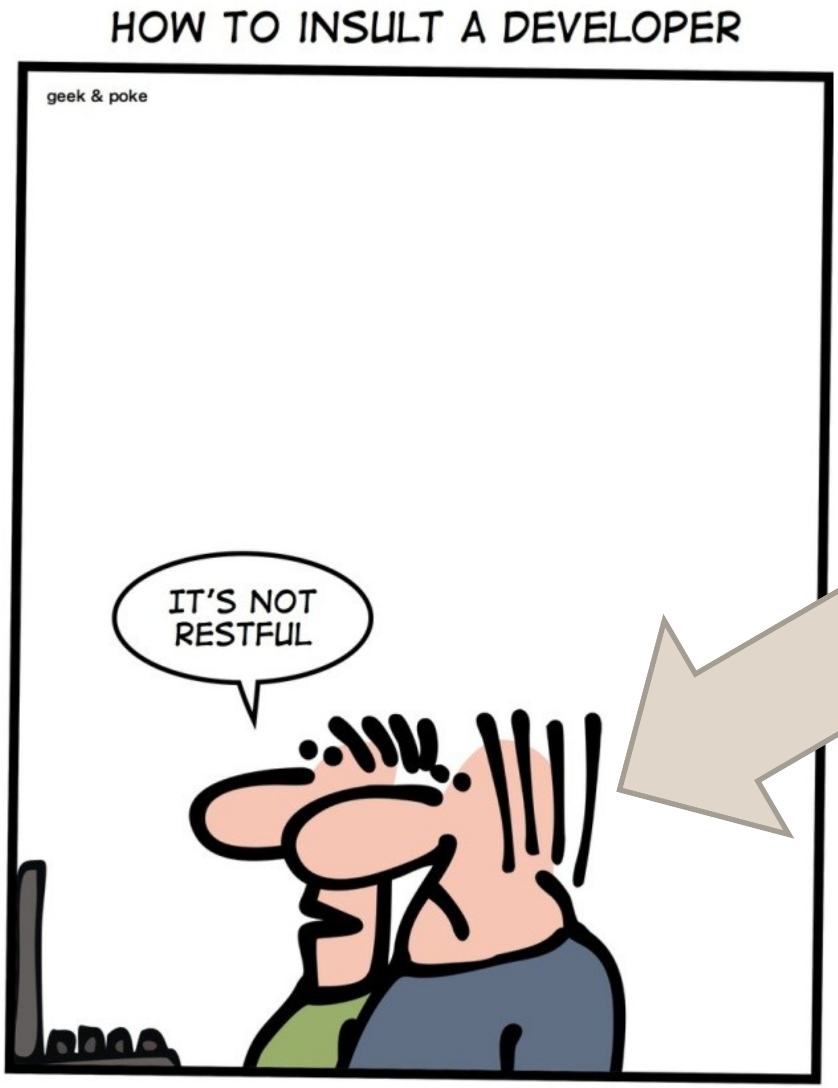
How many times REST in mentioned in the HTTP 1.1 Spec?

# What is REST

Answer

0 times

# Classic comic (My version)



Show me the Spec that says what is and what is NOT RESTFUL.

# The big companies

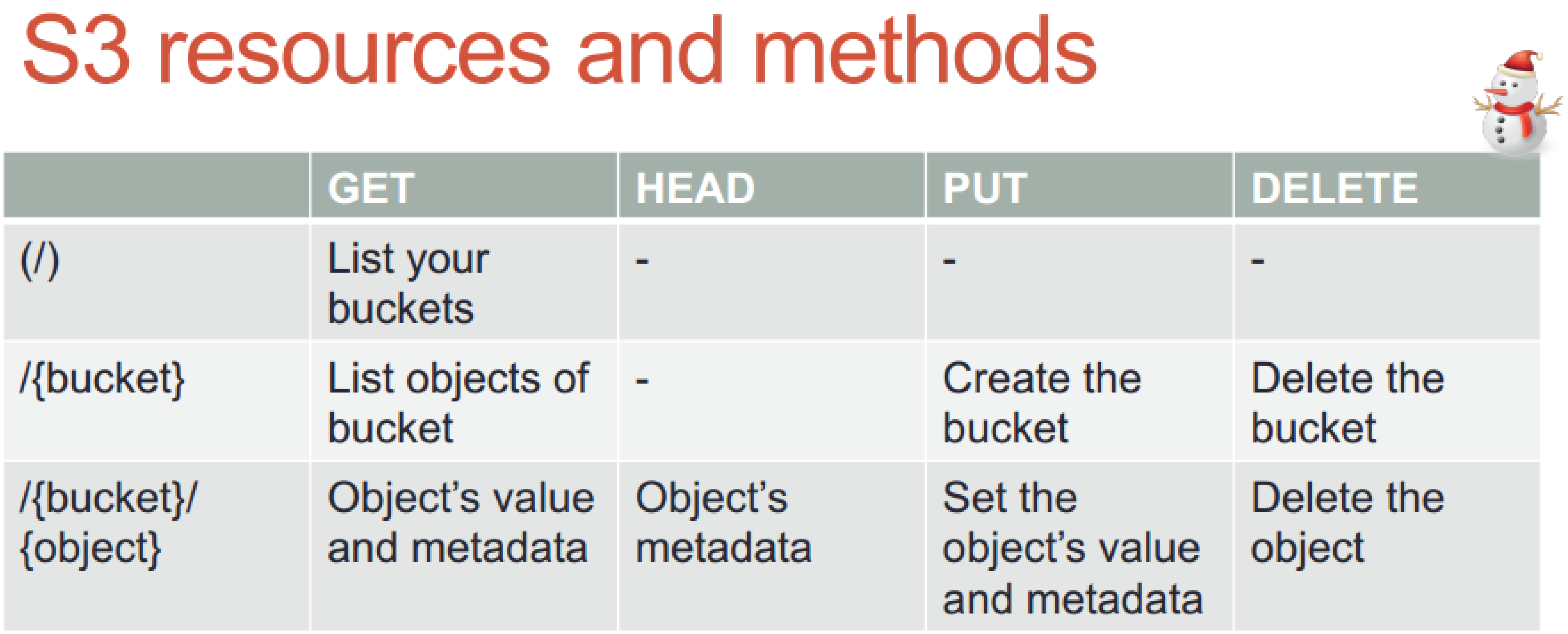
# An example

## Dropbox API

Save/load/sync files on the cloud

Seems natural fit for REST

# The HTTP methods



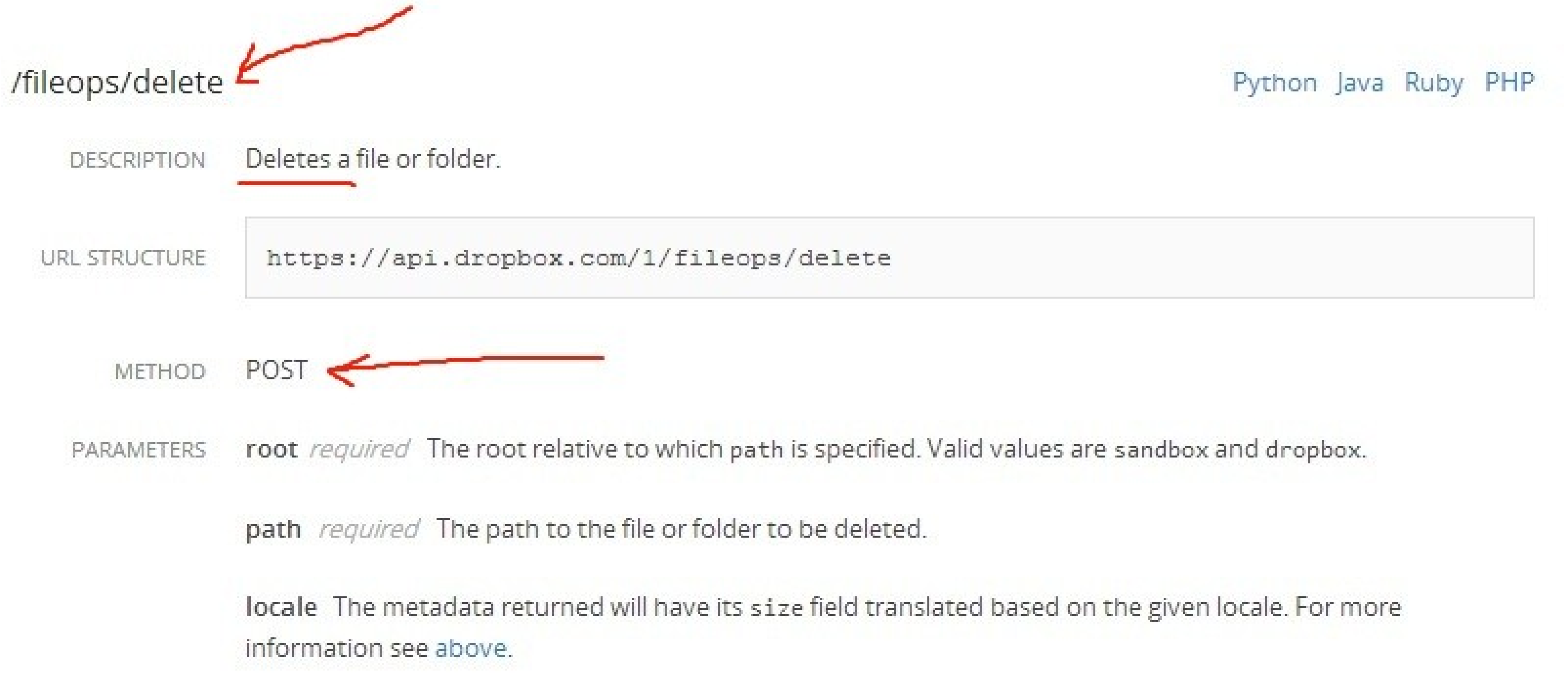
## S3 resources and methods

| | GET | HEAD | PUT | DELETE |
|---|---|---|---|---|
| (/) | List your buckets | - | - | - |
| /{bucket} | List objects of bucket | - | Create the bucket | Delete the bucket |
| /{bucket}/{object} | Object's value and metadata | Object's metadata | Set the object's value and metadata | Delete the object |

I expected something similar

# The DropBox API

REST??



/fileops/delete

Python Java Ruby PHP

DESCRIPTION Deletes a file or folder.

URL STRUCTURE

```
https://api.dropbox.com/1/fileops/delete
```

METHOD POST

PARAMETERS **root** *required* The root relative to which `path` is specified. Valid values are `sandbox` and `dropbox`.

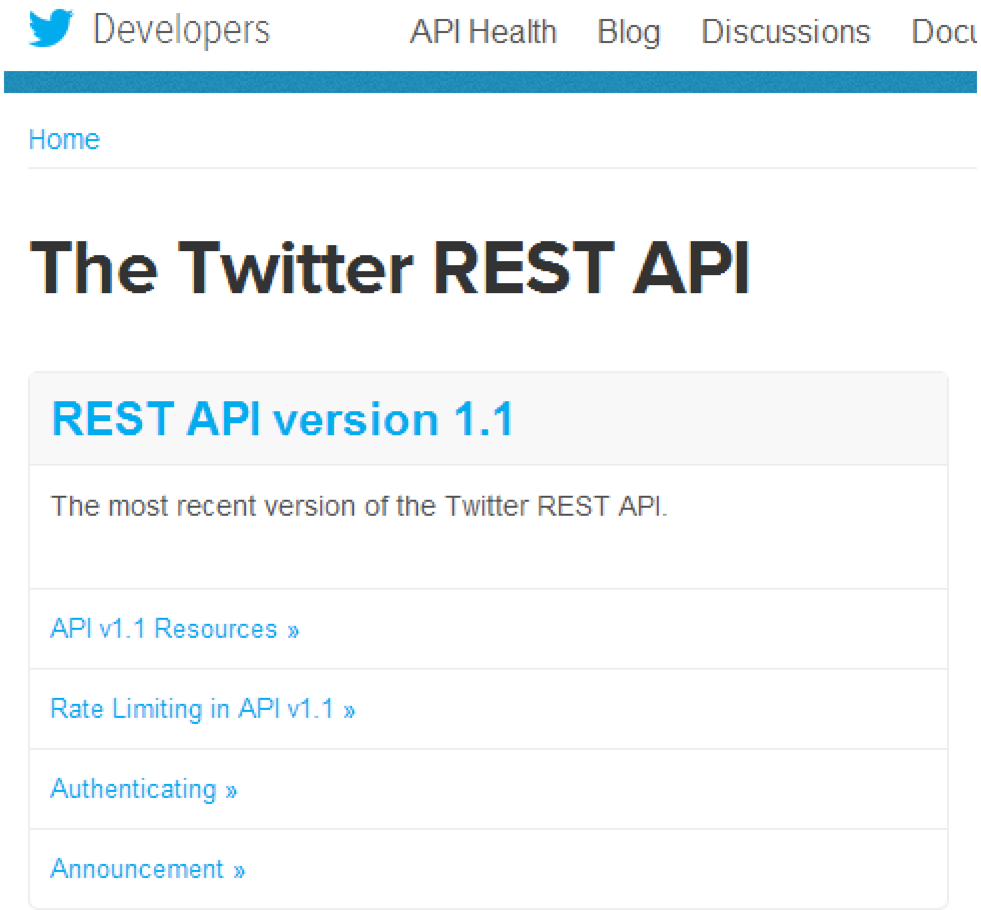
**path** *required* The path to the file or folder to be deleted.

**locale** The metadata returned will have its `size` field translated based on the given locale. For more information see above.

# Ok maybe I lied

Dropbox says that this is their API, not that they actually follow REST

# Another example



Home

## The Twitter REST API

### REST API version 1.1

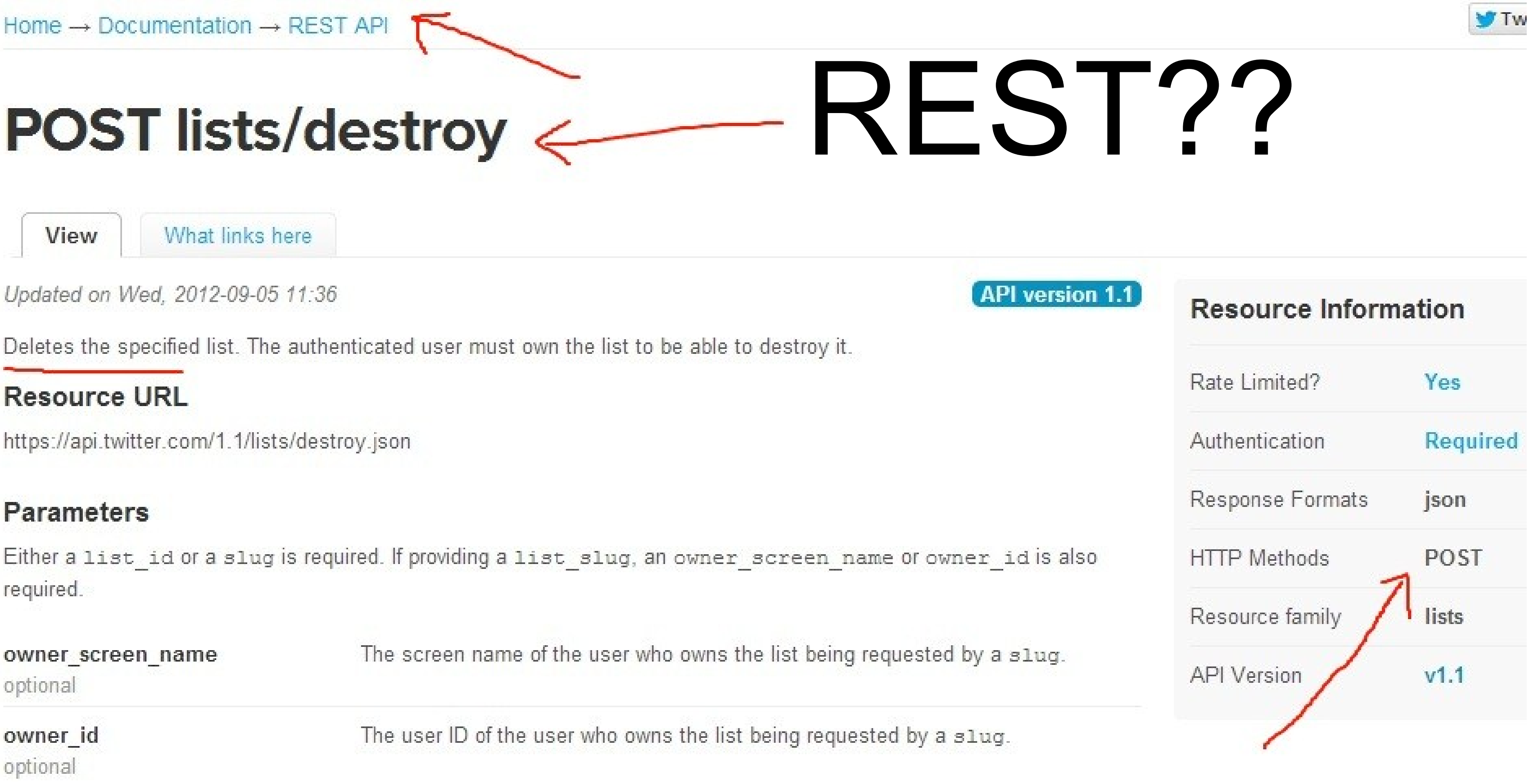
The most recent version of the Twitter REST API.

- API v1.1 Resources »
- Rate Limiting in API v1.1 »
- Authenticating »
- Announcement »

The Twitter REST API

# The Twitter “REST” API

REST??



Home → Documentation → REST API

Tw

## POST lists/destroy

View | What links here

*Updated on Wed, 2012-09-05 11:36*

API version 1.1

Deletes the specified list. The authenticated user must own the list to be able to destroy it.

### Resource URL

https://api.twitter.com/1.1/lists/destroy.json

### Parameters

Either a `list_id` or a `slug` is required. If providing a `list_slug`, an `owner_screen_name` or `owner_id` is also required.

| | |
|---|---|
| **owner_screen_name**<br>optional | The screen name of the user who owns the list being requested by a `slug`. |
| **owner_id**<br>optional | The user ID of the user who owns the list being requested by a `slug`. |

### Resource Information

| | |
|---|---|
| Rate Limited? | Yes |
| Authentication | Required |
| Response Formats | json |
| HTTP Methods | POST |
| Resource family | lists |
| API Version | v1.1 |

# Quote

“I am getting frustrated by the number of people calling any HTTP-based interface a REST API.”

# Roy Tomas Fielding

Roy said it in his blog!

http://roy.gbiv.com/untangled/2008/rest-apis-must-be-hypertext-driven

# Error codes in REST

# Description of part 2

# Error handling in REST applications

## Description of part 2

# Error handling in ~~REST~~ applications

HTTP

# Description of part 2

Error handling in ~~REST~~ HTTP1.1 applications

# REST error codes

## Approach 1

Use HTTP status codes

# HTTP errors

- 200 OK
- 400 bad request
- 401 Unauthorised
- 404 Not found
- 406 Not acceptable
- 413 Request entity too large
- 500 Internal server error

# REST error codes

## Error 407

Proxy authentication required

# REST error codes

Error 415

Unsupported media type

# REST error codes

## Error 500

Internal Server error

# REST error codes

## Error 503

### Service Unavailable

The server is currently unable to handle the
request due to a temporary overloading or maintenance
of the server. The implication is that this is a temporary
condition which will be alleviated
after some delay.

# REST error codes

WTF?

I am a Java developer.
Why should I bother with these error codes?

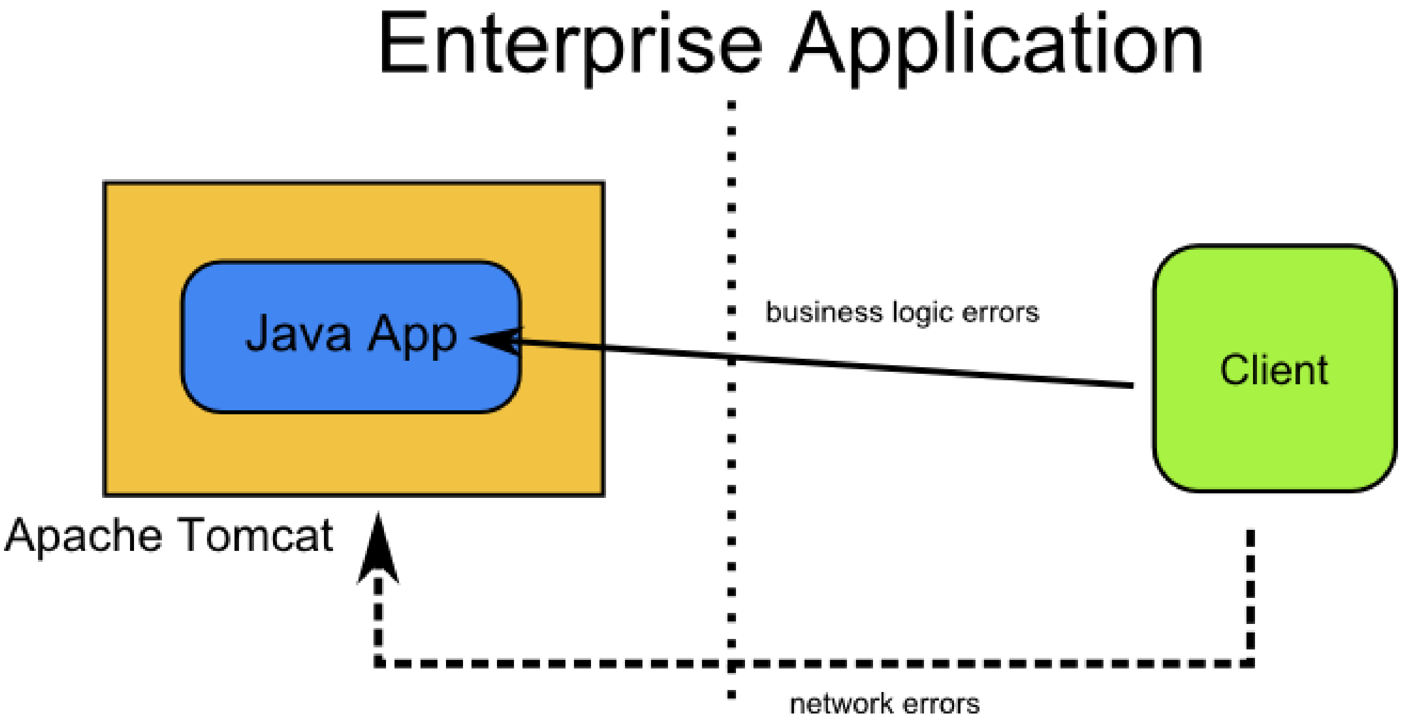
Maybe a system administrator must see those errors

# The HTTP error codes

- 1xx informational
- 2xx success
- 3xx Redirection
- 4xx Client Error
- 5xx Server error

Where is Application error?

# Another view of error handling



# Different view and severity of errors

- Server is overloaded
- User has not supplied a field in a form
- Proxy authentication is required
- Temporary redirect
- Attempt to draw money from empty account
- Request timeout
- HTTP version not supported
- User has not access to this resource

# Another view of error handling

## Another suggestion

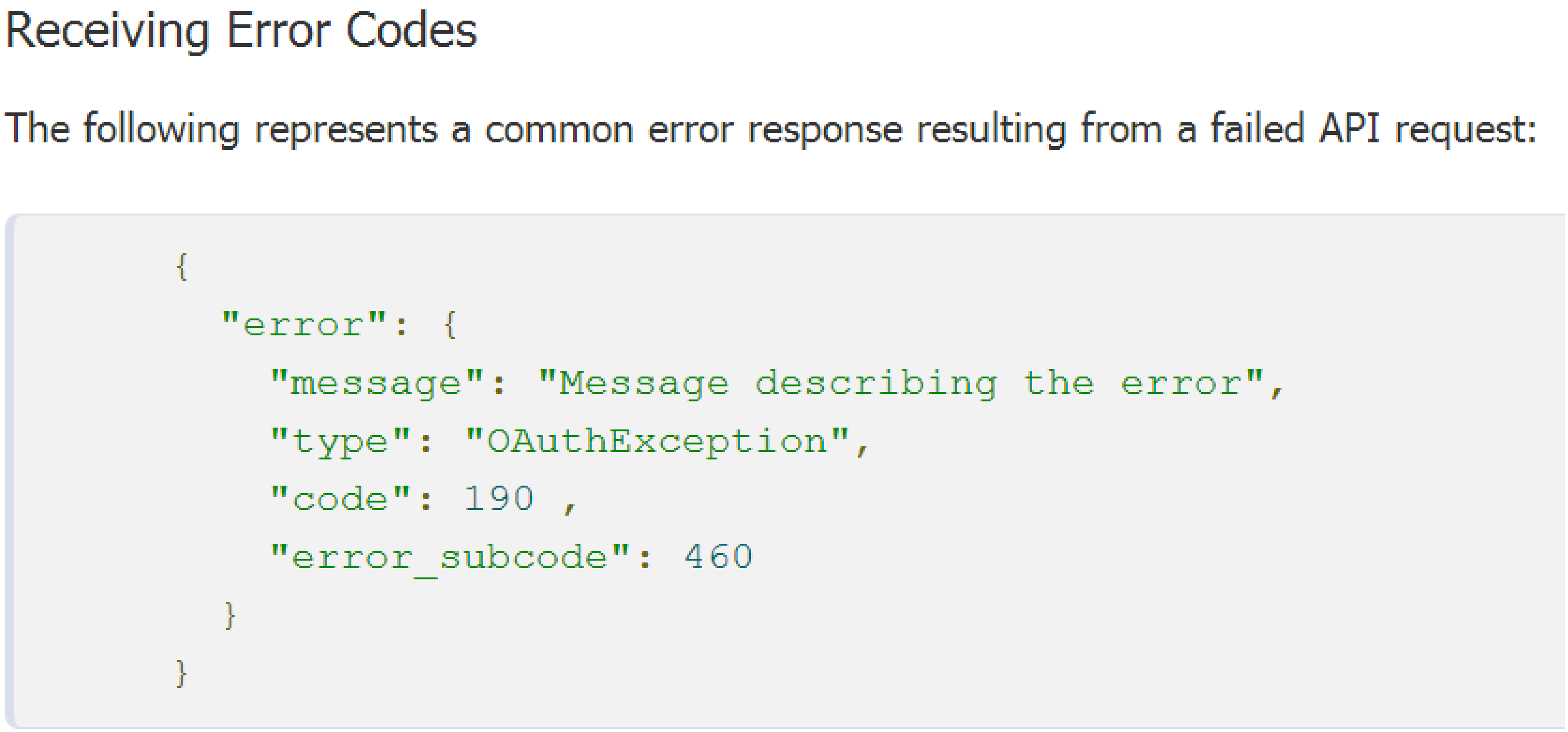
Decouple HTTP error codes for your application

# My suggestion

## Use 200 OK 99%

This means that the client request has reached your Java application (and network is ok)

# Include the error code inside the response



## Receiving Error Codes

The following represents a common error response resulting from a failed API request:

```
{
  "error": {
    "message": "Message describing the error",
    "type": "OAuthException",
    "code": 190 ,
    "error_subcode": 460
  }
}
```

# REST error codes

## Benefits

## Major benefits

- Easy to understand network failures vs development errors (Very important)
- Unlimited codes
- Unlimited ways to return errors (simple string or big object)
- Can clearly define retry-time out and back off policy
- Can use across Http or SOAP or Indian smoke signals

# Facebook does its

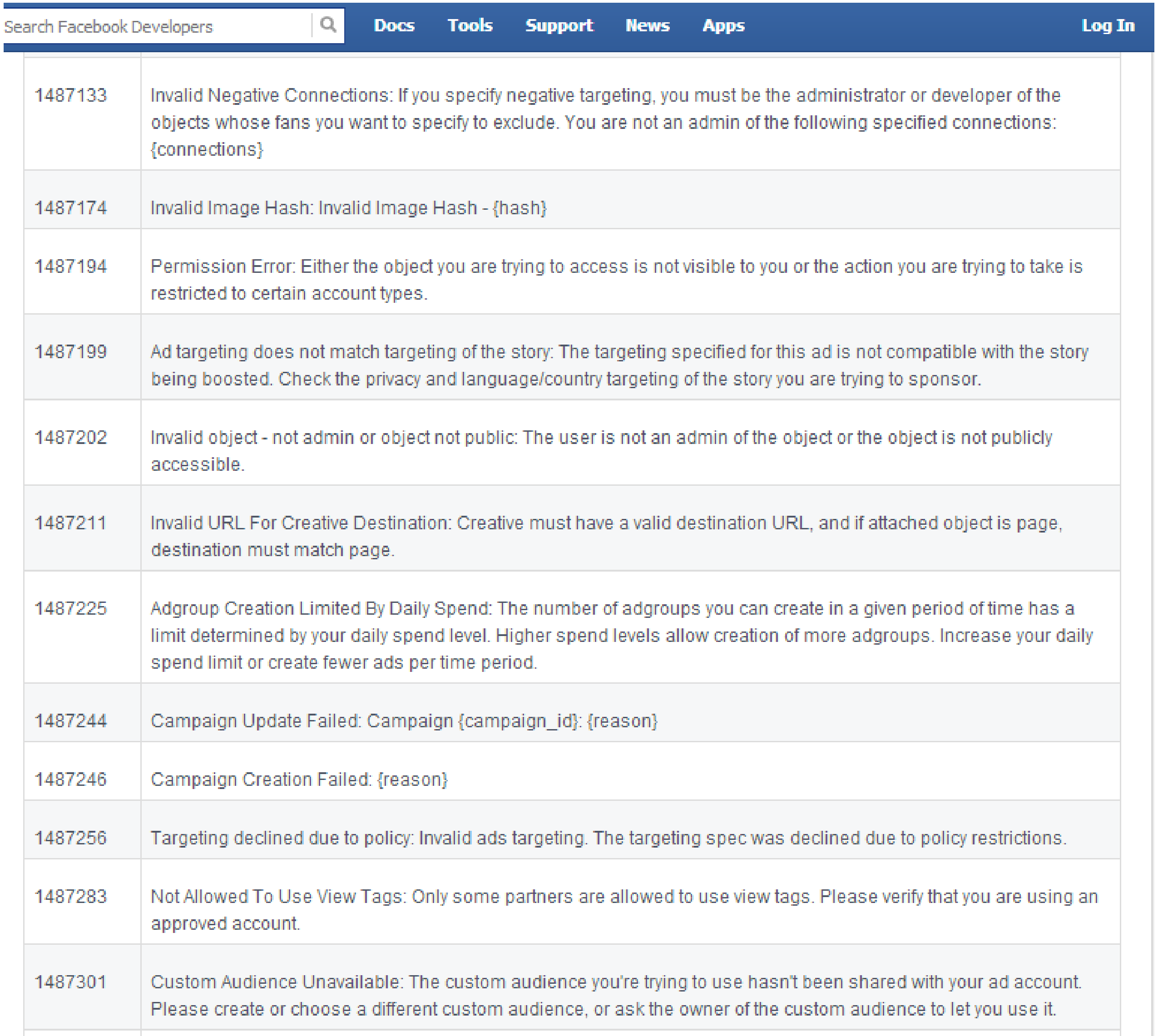
## Facebook API

The “REST API” is obsolete. (December 2011)

New API is “Graph API”

https://developers.facebook.com/blog/post/616/

# Facebook ads error codes



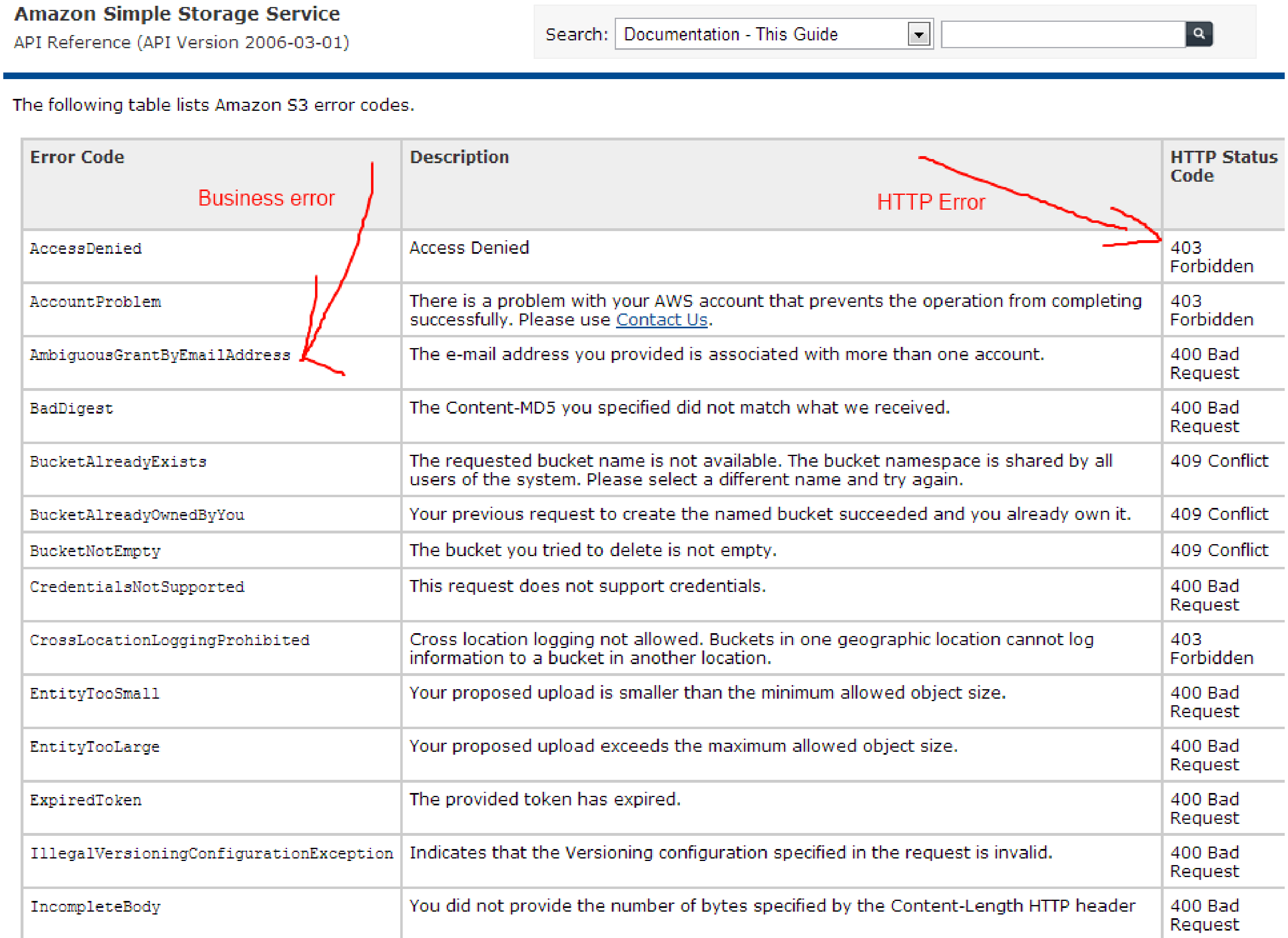
| | |
|---|---|
| 1487133 | Invalid Negative Connections: If you specify negative targeting, you must be the administrator or developer of the objects whose fans you want to specify to exclude. You are not an admin of the following specified connections: {connections} |
| 1487174 | Invalid Image Hash: Invalid Image Hash - {hash} |
| 1487194 | Permission Error: Either the object you are trying to access is not visible to you or the action you are trying to take is restricted to certain account types. |
| 1487199 | Ad targeting does not match targeting of the story: The targeting specified for this ad is not compatible with the story being boosted. Check the privacy and language/country targeting of the story you are trying to sponsor. |
| 1487202 | Invalid object - not admin or object not public: The user is not an admin of the object or the object is not publicly accessible. |
| 1487211 | Invalid URL For Creative Destination: Creative must have a valid destination URL, and if attached object is page, destination must match page. |
| 1487225 | Adgroup Creation Limited By Daily Spend: The number of adgroups you can create in a given period of time has a limit determined by your daily spend level. Higher spend levels allow creation of more adgroups. Increase your daily spend limit or create fewer ads per time period. |
| 1487244 | Campaign Update Failed: Campaign {campaign_id}: {reason} |
| 1487246 | Campaign Creation Failed: {reason} |
| 1487256 | Targeting declined due to policy: Invalid ads targeting. The targeting spec was declined due to policy restrictions. |
| 1487283 | Not Allowed To Use View Tags: Only some partners are allowed to use view tags. Please verify that you are using an approved account. |
| 1487301 | Custom Audience Unavailable: The custom audience you're trying to use hasn't been shared with your ad account. Please create or choose a different custom audience, or ask the owner of the custom audience to let you use it. |

# Amazon does it in another way

amazon.com API

Amazon has a hybrid approach (both types of error codes)

# Hybrid Approach by Amazon



**Amazon Simple Storage Service**
API Reference (API Version 2006-03-01)

The following table lists Amazon S3 error codes.

Business error

HTTP Error

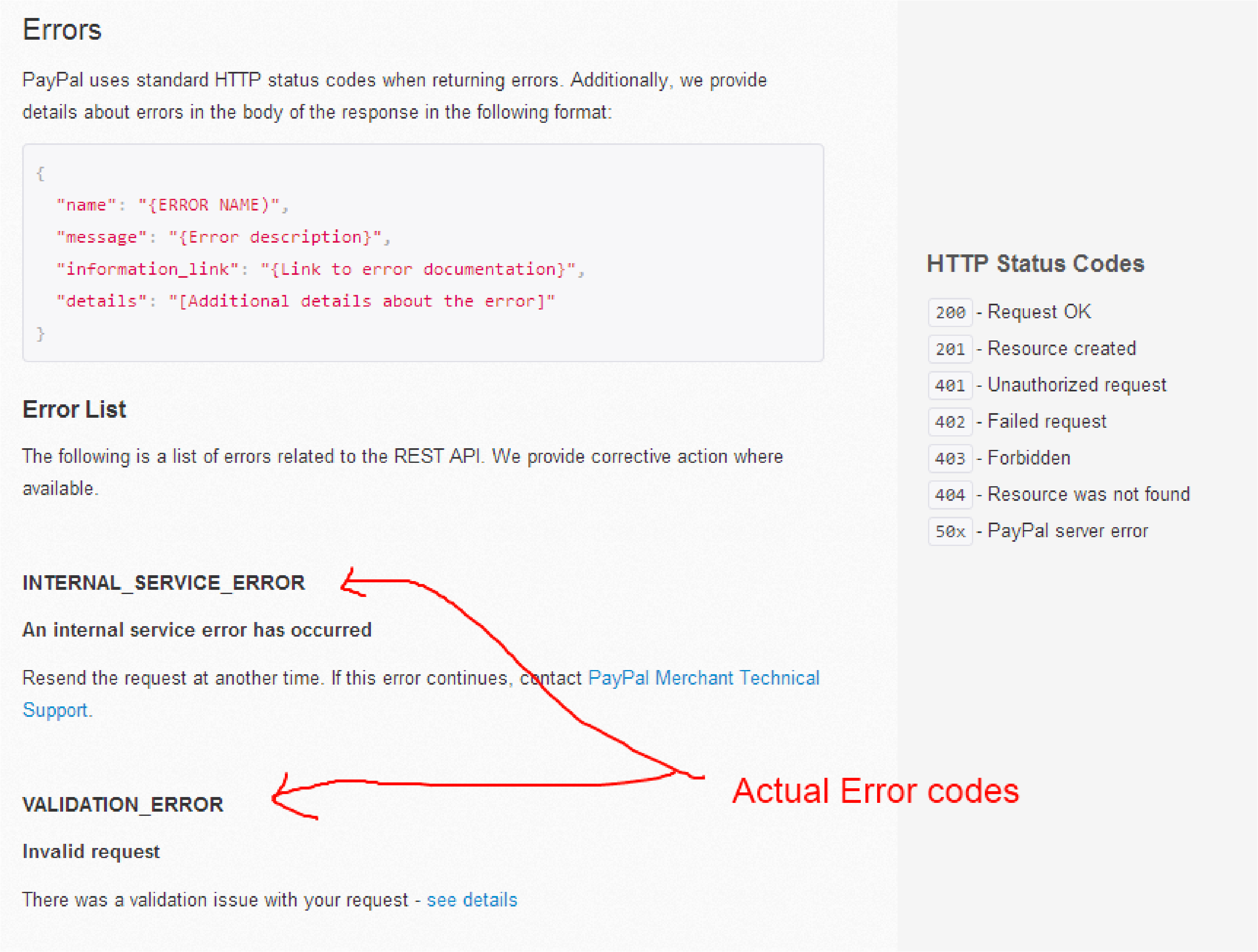
| Error Code | Description | HTTP Status Code |
|---|---|---|
| `AccessDenied` | Access Denied | 403 Forbidden |
| `AccountProblem` | There is a problem with your AWS account that prevents the operation from completing successfully. Please use Contact Us. | 403 Forbidden |
| `AmbiguousGrantByEmailAddress` | The e-mail address you provided is associated with more than one account. | 400 Bad Request |
| `BadDigest` | The Content-MD5 you specified did not match what we received. | 400 Bad Request |
| `BucketAlreadyExists` | The requested bucket name is not available. The bucket namespace is shared by all users of the system. Please select a different name and try again. | 409 Conflict |
| `BucketAlreadyOwnedByYou` | Your previous request to create the named bucket succeeded and you already own it. | 409 Conflict |
| `BucketNotEmpty` | The bucket you tried to delete is not empty. | 409 Conflict |
| `CredentialsNotSupported` | This request does not support credentials. | 400 Bad Request |
| `CrossLocationLoggingProhibited` | Cross location logging not allowed. Buckets in one geographic location cannot log information to a bucket in another location. | 403 Forbidden |
| `EntityTooSmall` | Your proposed upload is smaller than the minimum allowed object size. | 400 Bad Request |
| `EntityTooLarge` | Your proposed upload exceeds the maximum allowed object size. | 400 Bad Request |
| `ExpiredToken` | The provided token has expired. | 400 Bad Request |
| `IllegalVersioningConfigurationException` | Indicates that the Versioning configuration specified in the request is invalid. | 400 Bad Request |
| `IncompleteBody` | You did not provide the number of bytes specified by the Content-Length HTTP header | 400 Bad Request |

# Paypal does it in another way

PayPal® API

Paypal has also a hybrid approach (both types of error codes)

# Hybrid Approach by Paypal



## Errors

PayPal uses standard HTTP status codes when returning errors. Additionally, we provide details about errors in the body of the response in the following format:

```
{
  "name": "{ERROR NAME)",
  "message": "{Error description}",
  "information_link": "{Link to error documentation}",
  "details": "[Additional details about the error]"
}
```

### Error List

The following is a list of errors related to the REST API. We provide corrective action where available.

**INTERNAL_SERVICE_ERROR**

**An internal service error has occurred**

Resend the request at another time. If this error continues, contact PayPal Merchant Technical Support.

**VALIDATION_ERROR**

**Invalid request**

There was a validation issue with your request - see details

### HTTP Status Codes

| | |
|---|---|
| 200 | - Request OK |
| 201 | - Resource created |
| 401 | - Unauthorized request |
| 402 | - Failed request |
| 403 | - Forbidden |
| 404 | - Resource was not found |
| 50x | - PayPal server error |

Actual Error codes

# Final thoughts

## Conclusions

# Conclusion

REST is independent from HTTP. HTTP is independent from REST. (but they can be used together)

# Conclusion

There is no official Spec for REST (or even a reference implementation)

# Conclusions

## Conclusion

REST may or may not be for you

# Conclusions

## Conclusion

Even the major companies are in disagreement

# Conclusions

## Major Conclusion

Use what approach makes your life easy instead of following the trend/hype

# Thank you