REST API Design, Development & Management

raj@acloudfan.com

@acloudfan

http://ACloudFan.com

Discount Coupon Links to UDEMY courses:

https://www.udemy.com/hyperledger/?couponCode=DKHLF1099

https://www.udemy.com/ethereum-dapp/?couponCode=DKETH1099

https://www.udemy.com/rest-api/?couponCode=DKRST1099

mentoring, seeking Blockchain part time work, project guidance, advice … …
http://www.bcmentors.com

This deck is part of a online course on Summary of a course that covers the A to Z of RESTful API. More information.

REST API Design, Development & Management

Design Practices

http://www.acloudfan.com

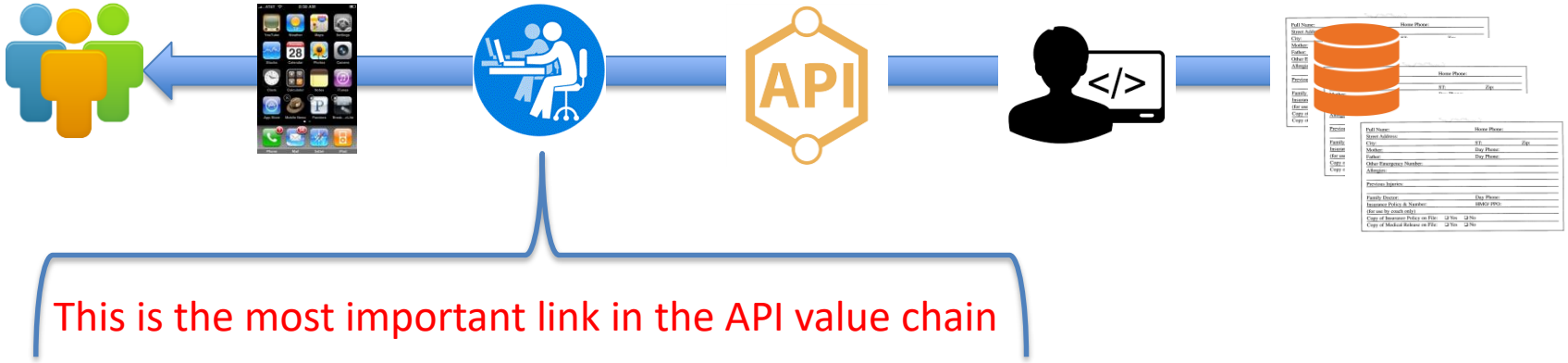
raj@acloudfan.com

Slides/Discussion from a course that covers the A to Z of RESTful API. More information.

Updated: Dec 27th, 2016

API Value Chain

raj@acloudfan.com
http://www.acloudfan.com



Design your API with the needs of App Developers in mind

raj@acloudfan.com
http://www.acloudfan.com

Resource identity, names, actions

http://www.acloudfan.com

raj@acloudfan.com

Summary of a course that covers the A to Z of RESTful API. More information.

UDEMY Link With Coupon Discount

Resource Identity

raj@acloudfan.com
http://www.acloudfan.com

API endpoint or URL for the resource

Walmart Open API

https://api.wlmartlabs.com/v1/items

Resource

Base URL

Version

Resource Names

raj@acloudfan.com
http://www.acloudfan.com

/items /items/{id}

/people /people/{id}

/programmes /programmes/{id}

Use **NOUNS** to identify the resources

Use of Plural is Suggested

Actions

raj@acloudfan.com
http://www.acloudfan.com

API Operation ≠ CRUD

UBER

/estimates/price

/friendships/lookup

Walmart Open API

/search?query

OK to use action as part of resource hierarchy

Or Create an action with subquery

Summary

raj@acloudfan.com
http://www.acloudfan.com

1. Create a simple base URL ; subdomain ; separate domain

2. Resources names should be nouns; Use of plurals suggested

3. Actions can be verbs may be standalone

   http://api.acme.com/search

4. URI Pattern for associations; avoid nesting over 3 levels

   http://api.acme.com/vacations/{id}/reviews/{review-id}

raj@acloudfan.com
http://www.acloudfan.com

CRUD Operations

http://www.acloudfan.com

raj@acloudfan.com

Part of a course that covers the A to Z of RESTful API. More information.

UDEMY Link With Coupon Discount

POST Request Handling

Creating a resource

raj@acloudfan.com
http://www.acloudfan.com

POST — Creates a resource

/vacations — /vacations/{id}/reviews

Success	Code = 201	• May return a link (id) to new resource in Location header *Location: http://api.acme.com/vacations/121/reviews/2* • May return the new object

Failure	4xx	Bad Request	e.g., 400 missing required field
	5xx	Issue in processing	e.g., 503 database unreachable

GET Request Handling

Reading a resource

raj@acloudfan.com
http://www.acloudfan.com

GET — Reads/Retrieves a resource collection or specific resource

/vacations /vacations/{id}

Success	Code = 200 OK	• Send back the response in requested format *GET http://api.acme.com/vacations/121*
Failure	4xx Bad Request	e.g., 404 Resource not found
	5xx Issue in processing	e.g., 500 Internal Server Error

PUT/PATCH Request Handling

Updating a resource

PUT

Updates all attributes of existing resource – effectively replace
Can also CREATE if client provide ID (not suggested)

PATCH

Modifies parts of an existing resource.
PATCH http://api.acme.com/vacations/121?validTill=3/1/2017

Success	Code = 200 Code = 204 No Content Code = 201 Created	• Resource in body of response – OPTIONAL • No need to send the Link – OPTIONAL

Failure	4xx	Bad Request	e.g., 404 Resource Not Found
	5xx	Issue in processing	e.g., 503 database unreachable

DELETE Request Handling

Deleting a resource

raj@acloudfan.com
http://www.acloudfan.com

DELETE — Deletes the resource

/vacations — /vacations/{id}

Success	Code = 200 Code = 204	• May return deleted resource in the response body • No returned content = 204

Failure	4xx	Bad Request	e.g., 404 Resource Not Found
	5xx	Issue in processing	e.g., 503 database unreachable

raj@acloudfan.com
http://www.acloudfan.com

API Response

http://www.acloudfan.com

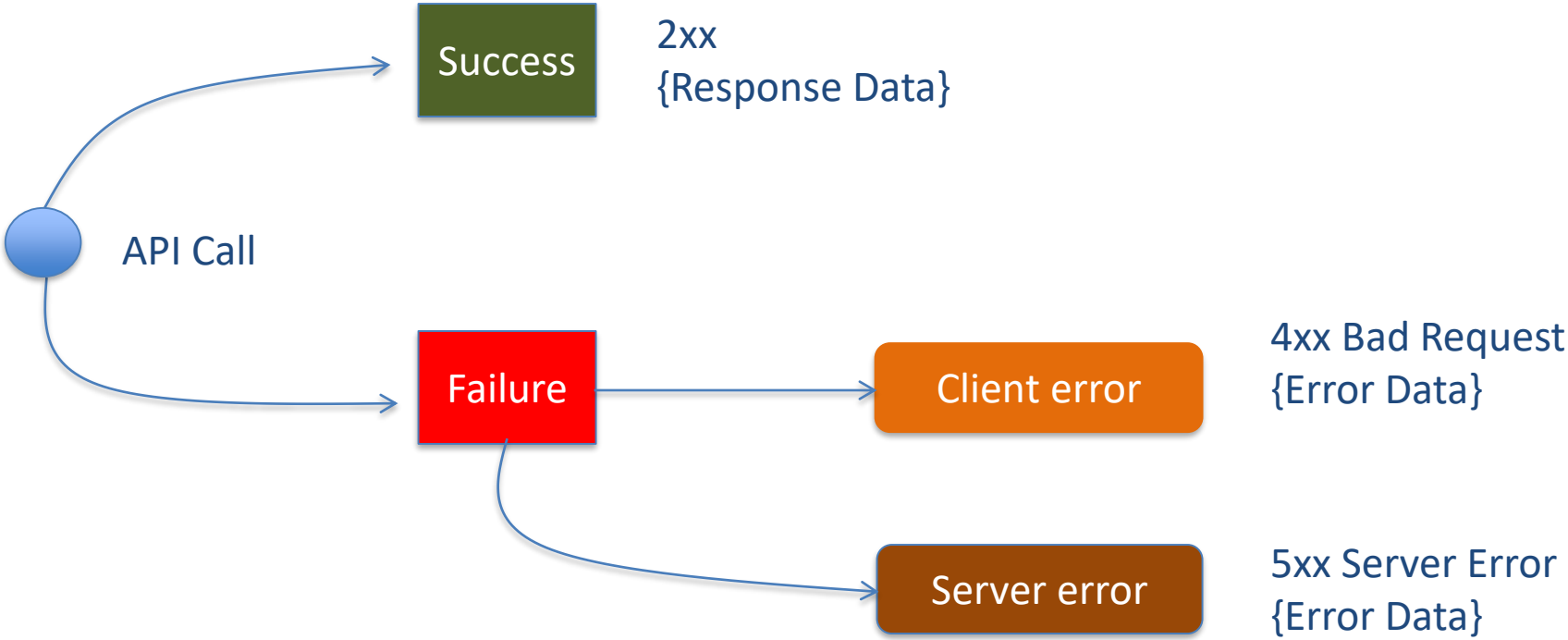
raj@acloudfan.com

Part of a course that covers the A to Z of RESTful API. More information.

UDEMY Link With Coupon Discount

API Response

raj@acloudfan.com
http://www.acloudfan.com



HTTP Status Codes

raj@acloudfan.com
http://www.acloudfan.com

- Over 70 status code that are hard to remember
- Decide on the code that you will use for your API
- Common ones

200 OK

201 Created

400 Bad Request

404 Not Found

401 Unauthorized

403 Forbidden

415 Unsupported media

500 Server Error

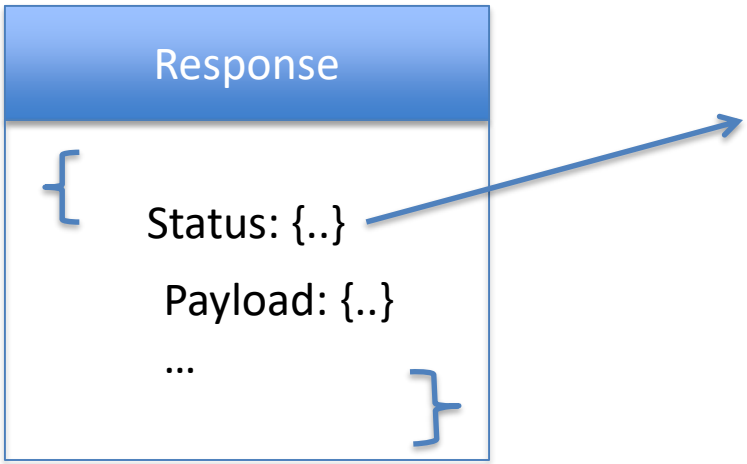
Restrict the number of HTTP codes to a maximum of 8

MUST use 200/400/500

An Alternate Approach to Error Response

raj@acloudfan.com
http://www.acloudfan.com

- Always sends back HTTP Status = 200 OK




```
// Meant for the developer
text:message,
timestamp:new Date(),
// POST, GET ....
method:httpMethod,
// Endpoint information
endpoint:endpointInformation,
// An array of all errors
errors : errorList,
// OPTIONAL -
// Use only during development
payload: receivedPayload
```

```
{
    code:7002,
    text:"Required field vacation 'number of nights' is missing",
    hints:["Please check that user has provided a number (between 1 & 31)"],
    info:"http://developer.acme.com/error#RequiredFields"
},
```

```
{
    code:7001,
    text:"Required field vacation 'name' is missing",
    hints:["Please check that user has provided the non null value for 'name'"],
    info:"http://developer.acme.com/error#RequiredFields"
}
```

facebook

raj@acloudfan.com
http://www.acloudfan.com

Handling Change

http://www.acloudfan.com

raj@acloudfan.com

Summary/Part of a course that covers the A to Z of RESTful API. More information.

UDEMY Link With Coupon Discount

API Changes

raj@acloudfan.com
http://www.acloudfan.com

- Adding a new operation or resource

  GET /hotels

- Adding optional parameters or properties

  GET /vacations?limit=3

Non Breaking

- Change the HTTP verb or methods

  PUT /vacations POST /vacations

- Delete an operation

  GET /vacationsByDestination

Breaking

Handling Change

raj@acloudfan.com
http://www.acloudfan.com

- Avoid changes : is the change really adding value?
  1. Eliminate or minimize impact on app developers
  2. Provide planning opportunity to the app developers
  3. Support backward compatibility (if possible)
  4. Provide support to app developers with the changes
  5. Minimize change frequency e.g., once per 6 months

Version your API right from day#1

raj@acloudfan.com
http://www.acloudfan.com

API Versioning

http://www.acloudfan.com

raj@acloudfan.com

Summary/Part of a course that covers the A to Z of RESTful API. More information.

UDEMY Link With Coupon Discount

Version Information

raj@acloudfan.com
http://www.acloudfan.com

HTTP Header

x-myapi-version: 1.2

Query parameter

/posts?version=1.2

facebook

URL

/v1/products

Multiple Version Support
Key points

raj@acloudfan.com
http://www.acloudfan.com

1. Support at least 1 previous version for a period of time
    - E.g., 3 months
2. Mark the previous version as deprecated
    - For new apps developers can access only the latest version
3. Publish a roll out plan in advance
4. Manage changelog that clearly shows the reason for new version

raj@acloudfan.com
http://www.acloudfan.com

Caching

http://www.acloudfan.com

raj@acloudfan.com

Summary/Part of a course that covers the A to Z of RESTful API. More information.

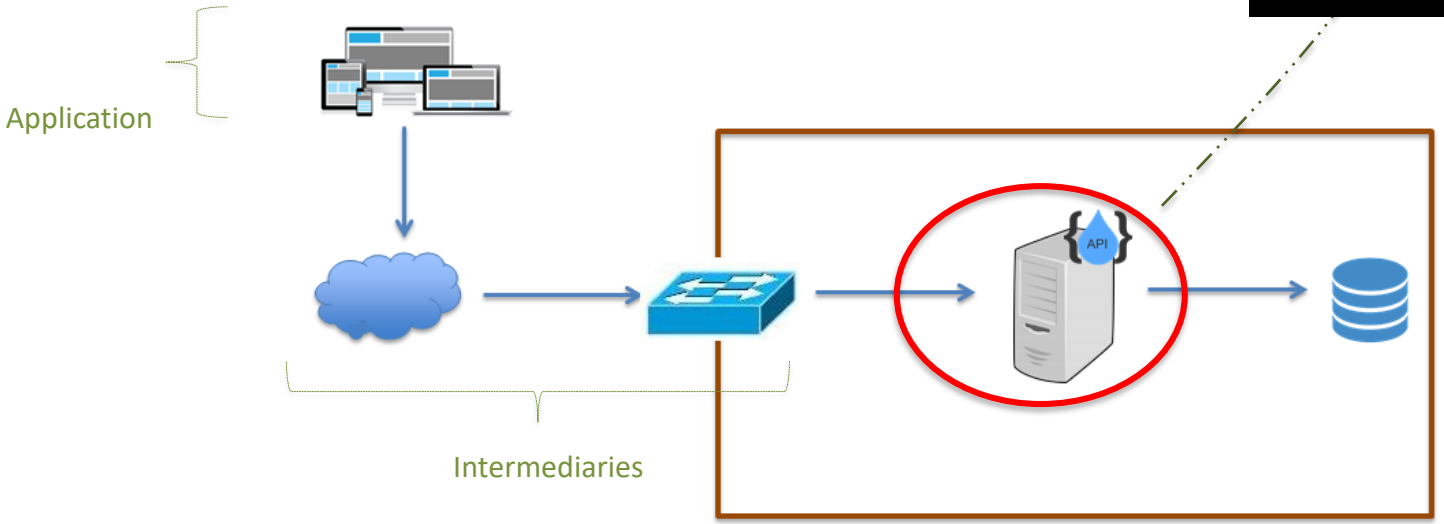
UDEMY Link With Coupon Discount

Design Decisions

raj@acloudfan.com
http://www.acloudfan.com

1. Which component should control the caching?
2. What to cache? Who can cache?
3. For how long is the cached data valid?

HTTP Cache Control Directives



Summary

raj@acloudfan.com
http://www.acloudfan.com

1. Benefits
   - Enhances performance
   - Leads to higher scalability
2. Data to cache depends on
   - Speed of change
   - Time sensitivity
   - Security
3. Design decisions
   - Who can cache?
   - For how long?

raj@acloudfan.com
http://www.acloudfan.com

HTTP Cache-Control

http://www.acloudfan.com

raj@acloudfan.com

Summary/Part of a course that covers the A to Z of RESTful API. More information.

UDEMY Link With Coupon Discount

Cache-Control

raj@acloudfan.com
http://www.acloudfan.com

Cache-Control : " directive-1, directive-2, …"

Cache-Control : "private, max-age=60"

Response HTTP Header
Cache-Control: "……."

- Who can cache the response?
- For how long?
- Under what conditions?

Request HTTP Header
Cache-Control: "……."

- Override the caching behavior
- Protect sensitive data from caching

Practices

raj@acloudfan.com
http://www.acloudfan.com

1. Take advantage of caching especially for high volume API

2. Consider *no-store* and *private* for sensitive data

3. Provide the validation tag (**ETag**) especially for large responses

4. Carefully decide on the optimal *max-age*

raj@acloudfan.com
http://www.acloudfan.com

Partial Response

http://www.acloudfan.com

raj@acloudfan.com

Summary/Part of a course that covers the A to Z of RESTful API. More information.

UDEMY Link With Coupon Discount

Benefits

raj@acloudfan.com
http://www.acloudfan.com

- Better performance & optimized resource usage
  - CPU, Memory, Bandwidth
- API consumer controls the granularity
- Common API version for all consumers
  - E.g., to support multiple devices, use cases – form factors

Specification

raj@acloudfan.com
http://www.acloudfan.com

/people:(id, first-name, last-name)

/friends?fields=id,name,picture

/pins?fields=id,link,creator(first_name)

raj@acloudfan.com
http://www.acloudfan.com

Pagination

http://www.acloudfan.com

raj@acloudfan.com

Summary/Part of a course that covers the A to Z of RESTful API. More information.

UDEMY Link With Coupon Discount

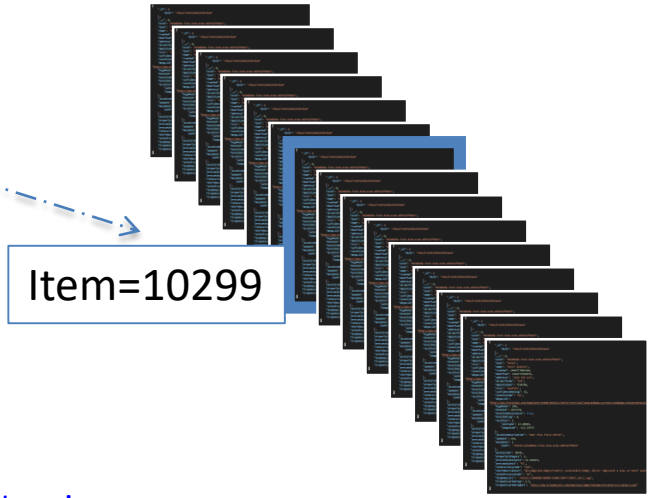
1. Cursors for Pagination

raj@acloudfan.com
http://www.acloudfan.com

- **Cursor** = “Control structure that enables traversal of records”
- Cursor based pagination considered most efficient

Cursor=“Random string”

Points to a specific item



facebook

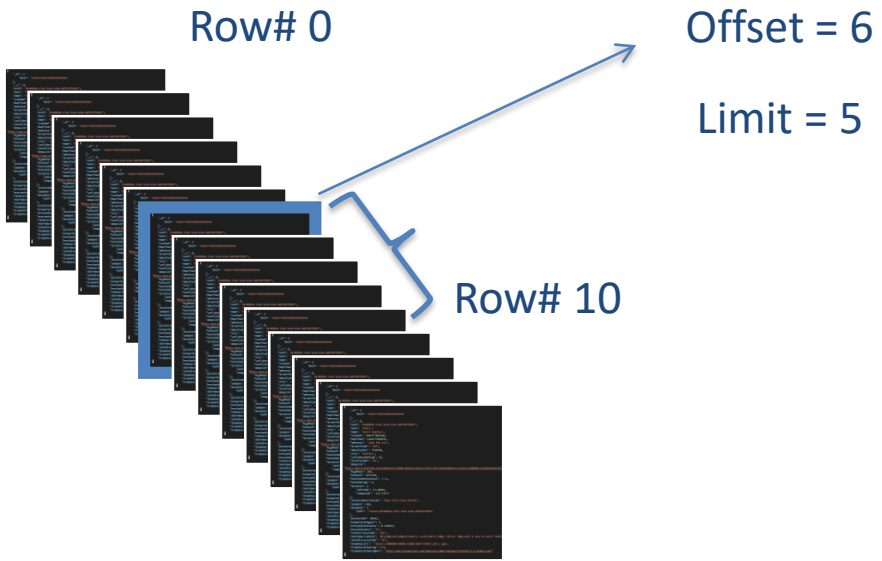
https://developers.facebook.com/docs/graph-api/using-graph-api/#paging

2. Offset based pagination

raj@acloudfan.com
http://www.acloudfan.com

- Commonly used approach

GET /../hotels?offset=6&limit=5



3. HTTP Link Header

raj@acloudfan.com
http://www.acloudfan.com

https://api.github.com/search/code?q=addClass

- Use of Link header *a.k.a. web linking*

  https://tools.ietf.org/html/rfc5988

Link to Next page

HTTP Header

```
Link: <https://api.github.com/user/repos?page=3&per_page=100>; rel="next",

<https://api.github.com/user/repos?page=50&per_page=100>; rel="last"
```

Link to Last page

Practices

raj@acloudfan.com
http://www.acloudfan.com

- Consider pagination for avoiding large data sets

  Decide on the default page size. May be different for API(s)

- Support for multiple pagination for different resources