

What are REST resources?

Asked 8 years, 8 months ago Active 3 years, 2 months ago Viewed 52k times



53



22



What are REST resources and how do they relate to resource names and resource representations?

I read a few articles on the subject, but they were too abstract and they left me more confused than I was before.

Is the following URL a resource? If it is, what is the name of that resource and what is its representation?

<http://api.example.com/users.json?length=2&offset=5>

The GET response of the URL should look something like:

```
[
  {
    id: 6,
    name: "John"
  },
  {
    id: 7,
    name: "Jane"
  }
]
```

[rest](#) [api](#) [terminology](#)

Share Follow

edited Jan 19 '17 at 12:01



[Ema.jar](#)

2,084 1 31 39

asked May 29 '12 at 12:30



[Emanuil Rusev](#)

31.3k 48 124 191

it seems to be json representation of users... – [Esailija](#) May 29 '12 at 12:34

Does that mean that the resource name is `users` ? – [Emanuil Rusev](#) May 29 '12 at 12:38

@Esailija, is the representation the resource itself? And if you have different representations for the same data, are these different resources or the same? – [Haralan Dobrev](#) May 29 '12 at 12:43

@HaralanDobrev how can representation of a resource be the resource itself? – [Esailija](#) May 29 '12 at 12:46

@Esailija, Could we assume a resource is just a representation of some data, accessed at a specific URI. Something like an abstraction and a connection between the URIs and the physical location of the data. – [Haralan Dobrev](#) May 29 '12 at 12:58

11 Answers

Active	Oldest	Votes
--------	--------	-------

43

The reason why articles on REST resources are abstract is because the concept of a REST resource *is* abstract. It's basically "whatever thing is accessed by the URL you supply". So, in your example, the resource would be the list of two users starting at offset 5 in some bigger list. Note that, how the resource is implemented is a detail you don't care about unless you are the one writing the implementation.



Is the following URL a resource?

The URL is not a resource, it is a label that identifies the resource, it is, if you like, the name of the resource.

The JSON is a *representation* of the resource.

Share Follow

answered May 29 '12 at 12:46



[JeremyP](#)

79.5k

15

113

156

6 Basically the URL is the endpoint where the resource that you want to use is located – [Yago Riveiro](#) May 29 '12 at 13:20

How should I refer to "users"? Is it correct to say that "users" is the data type of the resource? – [Emanuil Rusev](#) May 29 '12 at 14:38

Strictly speaking, `users` is just part of the URL. It need not necessarily "mean" anything. Of course, in reality, you know it is part of the data model of the application. – [JeremyP](#) May 29 '12 at 15:10

What's a Resource?

29

A resource is anything that's important enough to be referenced as a thing in itself. If your users might "want to create a hypertext link to it, make or refute assertions about it, retrieve or cache a representation of it, include all or part of it by reference into another representation, annotate it, or perform other operations on it", then you should make it a resource.



Usually, a resource is something that can be stored on a computer and represented as a stream of bits: a document, a row in a database, or the result of running an algorithm. A resource may be a physical object like an apple, or an abstract concept like courage, but (as we'll see later) the representations of such resources are bound to be disappointing. Here are some possible resources:

- Version 1.0.3 of the software release
- The latest version of the software release

- The first weblog entry for October 24, 2006
- A road map of Little Rock, Arkansas
- Some information about jellyfish
- A directory of resources pertaining to jellyfish
- The next prime number after 1024
- The next five prime numbers after 1024
- The sales numbers for Q42004
- The relationship between two acquaintances, Alice and Bob
- A list of the open bugs in the bug database

The text is from the O'Reilly book "[RESTful Web Services](#)".

Share Follow

edited Jun 4 '12 at 18:32

answered Jun 4 '12 at 15:23



[Emanuil Rusev](#)

31.3k 48 124 191



The URL is never a resource or its name or its representation.

16



URL just tells where the resource is located and You can invoke GET,POST,PUT,DELETE etc on this URL to invoke the resource.



Data responded back are the resources while the form of the data is its representation.

Let's say Your URL with given GET parameters can output a JSON resource - this is the JSON representation of this resource. While with other flag in the GET it could respond with the same data in XML - that will be another representation of the very same resource.

EDIT: Due to the comments to the OP and to my answer I'm adding another explanations.

Also the **resource name** is considered to be the 'script name', e.g. in this case it is `users.json` while this resource name is self describing the resource representation itself - when calling this resource we expect the resource is in JSON, while when calling e.g. `users.xml` we would expect the data in XML.

1. When I change the `offset` parameter in GET the response contains different data set - is it a new resource or its representation?
2. When I define which columns are returned in response in GET, is it a different resource or different representation, or?

1. Well, here the problem and answer are clear - we still call the same URL, the server responses with the data in the same form (still it is JSON), data still contains information about users - just the information itself has changed due to the new offset parameter. So it is obvious that **it is still the same resource with the same representation and the same resource name as before.**
2. Second problem could be a little confusing. Though we are calling the same resource, though the resource contains the same data (just with only predefined column set) and though the data is in the same representation it could seem to us as a different resource. But due to the points in the paragraph above it is nor the different resource or different representation. Though the data set contains less information the requesting side (filtering this data set) should be considering this and behave accordingly. So again: **it is the same resource with the same resource name and the same resource representation.**

Share Follow

edited May 30 '12 at 11:12

answered May 29 '12 at 12:48



shadyyx

15k 5 47 91

-
- 1 If you have a GET parameter fields which allows you to limit the properties listed for a given dataset. Like: [api.example.com/users/123?fields\[\]=id&fields\[\]=email](http://api.example.com/users/123?fields[]=id&fields[]=email) Then: **Is this a different resource? Is this a different representation? Or is this something else?** – Haralan Dobrev May 29 '12 at 12:57
 - 1 Representation is the same as well as the resource is the same but has provided different data set - but still it contains the users data, still in the same data format. So **NO** to Your first two questions and **it's the same resource with the same representation** to Your third question. – shadyyx May 29 '12 at 13:03
-



13

What are REST resources and how do they relate to resource names and resource representations?



REST doesn't mean a great deal more then you use HTTP verbs (GET, POST, PUT, DELETE, etc) properly.



Is the following URL a resource?

All URLs are strings that tell computers where a resource can be located. (Hence the name: Uniform *Resource* Locator).

Share Follow

edited May 29 '12 at 12:43

answered May 29 '12 at 12:37



Quentin

777k 96 1050 1185

-
- 1 +1. a consise but complete explanation of what REST really is :) – verisimilitude May 29 '12 at 12:40
-



REST

13

This architectural style was defined in the [chapter 5](#) of Roy T. Fielding's dissertation.



REST is about resource state manipulation through their representations on the top of stateless communication between client and server. It's a protocol independent architectural style but, in practice, it's commonly implemented on the top of the HTTP protocol.



Resources

The *resource* itself is an abstraction and, in the words of the author, a resource can be *any information that can be named*. The domain entities of an application (e.g. a person, a user, a invoice, a collection of invoices, etc) can be resources. See the following quote from Fielding's dissertation:

[5.2.1.1 Resources and Resource Identifiers](#)

The key abstraction of information in REST is a *resource*. Any information that can be named can be a resource: a document or image, a temporal service (e.g. "today's weather in Los Angeles"), a collection of other resources, a non-virtual object (e.g. a person), and so on. In other words, any concept that might be the target of an author's hypertext reference must fit within the definition of a resource. A resource is a conceptual mapping to a set of entities, not the entity that corresponds to the mapping at any particular point in time.

More precisely, a resource R is a temporally varying membership function $M_R(t)$, which for time t maps to a set of entities, or values, which are equivalent. The values in the set may be *resource representations* and/or *resource identifiers*. [...]

Resource representations

A JSON document is *resource representation* that allows you to represent the state of a resource. A server can provide different representations for the same resource. For example, using XML and JSON documents. A client can use *content negotiation* to request different representations of the same resource.

Quoting Fielding's dissertation:

[5.2.1.2 Representations](#)

REST components perform actions on a resource by using a representation to capture the current or intended state of that resource and transferring that representation between components. A representation is a sequence of bytes, plus representation metadata to describe those bytes. Other commonly used but less precise names for a representation include: document, file, and HTTP message entity, instance, or variant.

A representation consists of data, metadata describing the data, and, on occasion, metadata to describe the metadata (usually for the purpose of verifying message integrity). Metadata is in the form of name-value pairs, where the name corresponds to a standard that defines the value's structure and semantics. Response messages may include both representation metadata and resource metadata: information about the resource that is not specific to the supplied representation. [...]

Over HTTP, request and response headers can be used to exchange metadata about the representation.

Resource identifiers

A URL a *resource identifier* that identifies/locates a resource in the server.

This [answer](#) may also be insightful.

Share Follow

edited Nov 22 '17 at 14:31

answered Nov 22 '17 at 13:48



[cassiomolin](#)

96.5k 22 203 265

A resource is:

7

- a noun
- that is unique
- and can be represented as data
- and has at least one URI

I go into more detail on my blog post, [What, Exactly, Is a RESTful Resource?](#)

Share Follow

answered Jun 15 '14 at 19:07



[techiferous](#)

259 3 5

GET/POST/PUT/Patch/DELETE is **noun?**. – [muneeb_ahmed](#) Apr 25 '18 at 5:34

@muneeb_ahmed, **they are verbs**, not noun. According to **Cambridge dictionary**, noun means: a word that refers to a person, place, thing, event, substance, or quality . Examples of nouns: **Person, Table, Action, Artifact, Owner**, etc. – [Thiago Navarro](#) Sep 19 '20 at 19:40 ✎

Conceptually you can think about a resource as everything which is accessible on the web using an URL. If you stick to this rule `http://api.example.com/users.json?length=2&offset=5` can be considered a resource

Share Follow

edited May 29 '12 at 12:53

answered May 29 '12 at 12:38

[Massimiliano Peluso](#)**24.5k** 6 51 64

How should I refer to "users"? Is it correct to say that "users" is the data type of the resource? –

[Emanuil Rusev](#) May 29 '12 at 14:42 ✎

conceptually it is not correct. An image for example can be a resource

"http://MyApplication/User/Image.jpg" what would be the datatype? I hope it makes sense –

[Massimiliano Peluso](#) May 29 '12 at 14:48 ✎

How then should I refer to "users"? Doesn't it make sense to accept that Image.jpg is of type "image"? –

[Emanuil Rusev](#) May 29 '12 at 14:57

it is just part of the URL that's it. You can consider it just the name of the resource if it helps as for image.jpg – [Massimiliano Peluso](#) May 29 '12 at 15:04 ✎

- 1 I need a way to refer to it. I want to differentiate /users , which retrieves multiple items, from /users/123 , which retrieves one. – [Emanuil Rusev](#) May 29 '12 at 15:07

Representational State Transfer (REST) is a style of software architecture for distributed systems such as the World Wide Web. REST-style architectures consist of clients and servers. Clients initiate requests to servers; servers process requests and return appropriate responses.

Requests and responses are built around the transfer of representations of resources. Resources are a set of addressable objects, basically files and documents, linked using URLs. As correctly pointed out above by Quentin, REST architecture simply implies that you'd use the HTTP verbs GET/POST/PUT/DELETE...

Share Follow

answered May 29 '12 at 12:49

[verisimilitude](#)**4,589** 2 23 32

You've only provided what appear to be relative parameters rather than "ID" which is (or should be) concrete. Remember, get operations should be idempotent (i.e. repeatable with the same outcome).

Share Follow

answered May 29 '12 at 12:37

[Jeff Watkins](#)**6,213** 12 17

What is REST?

REST is an architecture Style which stands for Representational(RE) State(S) transfer(T).

What is REST Resource ?

Rest Resource is data on which we want to perform operation(s). So this data can be present in database as record(s) of table(s) or in any other form. This record has unique identifier with which it can be identified like id for Employee.

Now when this data is requested by unique url like <http://www.example.com/employees/123>, so ultimately data or record which is present in database will be converted to JSON/XML/Plain text format by Rest Service and will be sent to Consumer.

So basically, what is happening here is REPRESENTATIONAL STATE TRANSFER, in a way that state of the data present in database is transferred to another format which can be JSON/XML or plain text.

So in this case 1 employee represents 1 resource which can be accessed by unique url like <http://www.example.com/employees/123>

In case we want to get list of all resources(employees), we will do:
<http://www.example.com/employees>

Hope this will help.

Share Follow

answered Aug 20 '17 at 13:42



[javaguy](#)

653 2 10 25



-2



REST stands for REpresentational State Transfer. It's a method of transferring variable information from one location to another. A common method of doing this is using JSON - a way of formatting your variables so that they can be transferred without loss of information.

PHP, for example, has built in JSON support. Passing a PHP array to `json_encode($array)` will output a string in the format you posted (which by the way, is indeed a REST resource, because it gives you variables and information).

In PHP, the array you posted would turn out as:

```
Array (
    [0]=>Array (
        ['id']=>6;
        ['name']=>'John';
    )
    [1]=>Array (
        ['id']=>7;
        ['name']=>'Jane';
    )
)
```

Share Follow

answered May 29 '12 at 12:35



Death

1,981

11

14

4 He asked what are "Resources" in Rest API. – [RachitSharma](#) Mar 12 '15 at 6:49



Highly active question. Earn 10 reputation in order to answer this question. The reputation requirement helps protect this question from spam and non-answer activity.