



Europe's leading AEM Developer Conference
28-30 SEPTEMBER 2020



GraphQL in Apache Sling... but isn't it the opposite of REST?

Bertrand Delacrétaz - Principal Scientist, Adobe

Images are from stock.adobe.com unless otherwise specified
slides revision: 2020-09-17



adaptTo()

What's GraphQL ?

GraphQL Endpoint

Method

[Edit HTTP Headers](#)

GraphiQL



Prettify

History

[Docs](#)

```
1 {
2   navigation {
3     search
4     sections {
5       path
6       name
7     }
8   }
9   article(withText: "Devan") {
10    path
11    title
12    section
13    seeAlso {
14      title
15      path
16    }
17  }
18 }
19
```

Query with the same "shape" as the output JSON.
Also: **Mutations** and **Subscriptions**

QUERY VARIABLES

```
{
  "data": {
    "navigation": {
      "search": "/content/search",
      "sections": [
        {
          "path": "/content/articles/music",
          "name": "Music"
        },
        {
          "path": "/content/articles/adventure",
          "name": "Adventure"
        }
      ]
    },
    "article": [
      {
        "path": "/content/articles/music/devan-thiel-on-the-transmitter-of-haptic-et-recusandae-aka-jbod",
        "title": "Music - Devan Thiel on the transmitter of haptic 'et recusandae' (aka JBOD)",
        "section": "Music",
        "seeAlso": [
          {
            "title": "Adventure - Alicia Yundt on the feed of digital 'culpa ipsa' (aka USB)",
            "path": "/content/articles/adventure/alicia-yundt-on-the-feed-of-digital-culpa-ipsa-aka-usb"
          },
          {
            "title": "Adventure - Delta Steuber on the application of bluetooth 'illum odit' (aka RSS)",
            "path": "/content/articles/adventure/delta-steuber-on-the-application-of-bluetooth-illum-odit-aka-rss"
          }
        ]
      }
    ]
  }
}
```

POST to server returns JSON output with **just** what you need.



GraphQL Schema

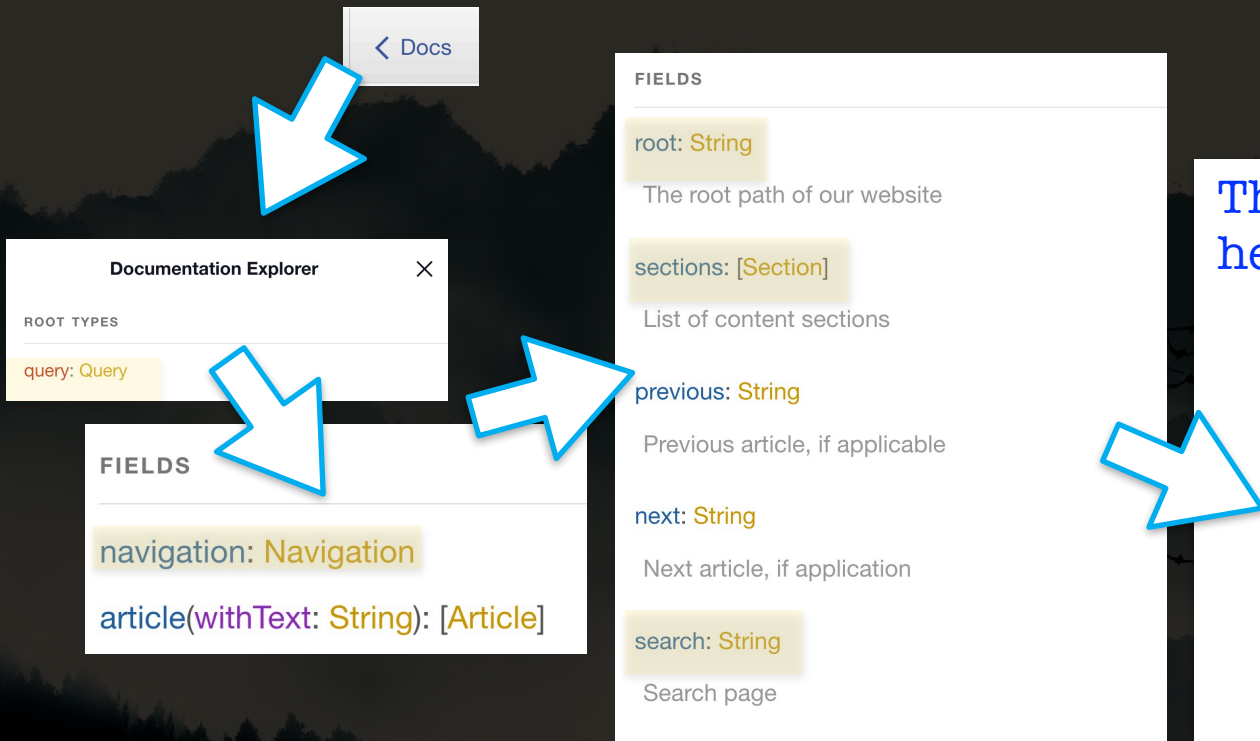
```
type Query {  
  # The current content section  
  section: Section @fetcher(name:"website/currentResource")  
  navigation: Navigation @fetcher(name:"website/navigation")  
}  
  
type Navigation {  
  # The root path of our website  
  root: String  
  
  # List of content sections  
  sections: [Section]  
  
  # Previous/next articles  
  previous: String  
  next: String  
  
  # Search page  
  search: String  
}  
  
...
```

In our case, the schema is generated from an internal Sling request to

<resource>.GQLschema

So it can be specific to a given **sling:resourceType**

GraphQL Schema Introspection



The schema information helps us build this query:

```
{
  navigation {
    search
    sections {
      path
      name
    }
  }
}
```



adaptTo()

Sling GraphQL Demo

Sling-samples GraphQL website demo

[Travel](#) [Music](#) [Adventure](#) [News](#) [Culture](#)
[Search](#)

[Travel](#) [Music](#) [Adventure](#) [News](#) [Culture](#) [Business](#)
[Previous article](#) [Next article](#) [Search](#)

Music

- [Music - Christine MacGyver](#)
- [Music - Conor Considine on](#)
- [Music - Wellington Russel or](#)
- [Music - Eloy Hahn on the sy](#)
- [Music - Roxane O'Connell or](#)
- [Music - Chadd Beier on the](#)
- [Music - Caitlyn Schumm on](#)
- [Music - Anya Macejkovic on](#)
- [Music - Jennie Conroy on th](#)
- [Music - Naomi Beahan on th](#)
- [Music - Clarabelle O'Reilly o](#)
- [Music - Stephen Beahan on](#)
- [Music - Phyllis Fahey on the](#)
- [Music - Phoebe Grimes on tl](#)

Server-side (GraphQL + hbs)

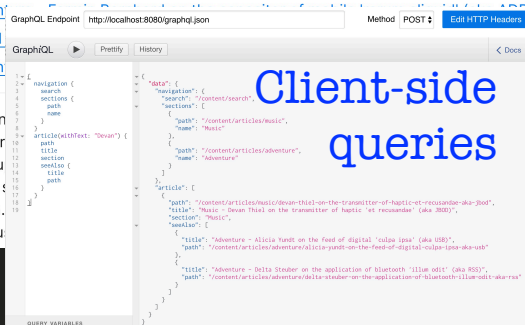
Music - Christine MacGyver on the capacitor of primary 'libero dolore' (aka RSS)

Tags: [bandwidth](#) [monitor](#) [driver](#) [pixel](#) - Find articles [with all these tags](#)

See Also

- [Travel - Solon Davis on the card of primary 'reiciendis omnis' \(aka SQL\)](#)
- [Adventure - Delta Steader on the application of bluetooth 'libre odi' \(aka RSS\)](#)
- [Travel](#)
- [Adventure](#)

As Christin
tenetur rer
atque volu
Commodi
quaerat in.
Accusamus



Client-side queries

Search in articles

Client-side

(GraphQL + hbs)

[Travel](#) [Music](#) [Adventure](#) [Culture](#) ← Click to select/deselect content sections

Found 3 articles containing "kunze" in the selected content sections

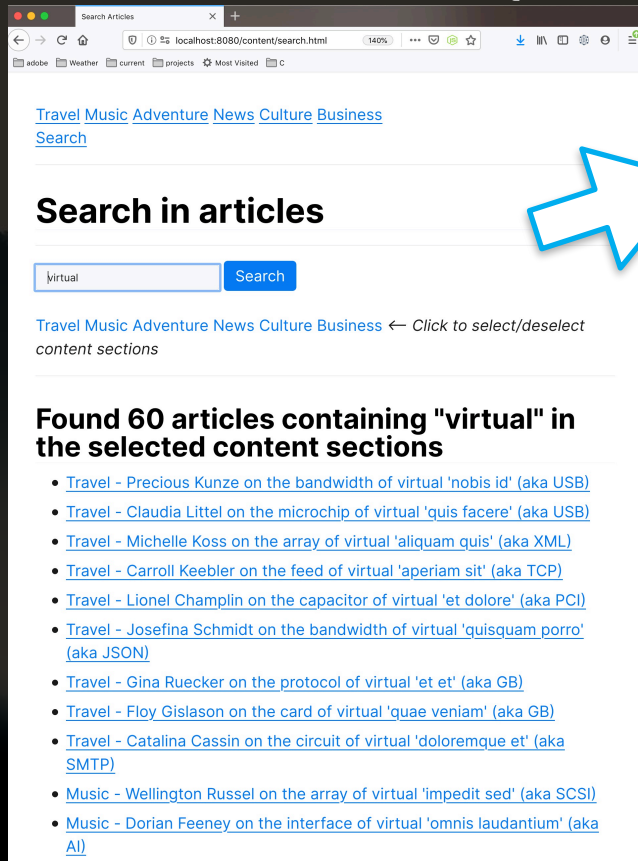
- [Travel - Precious Kunze on the bandwidth of virtual 'nobis id' \(aka USB\)](#)
- [Music - Edward Kunze on the alarm of open-source 'perspiciatis nihil' \(aka PNG\)](#)
- [Culture - Dahlia Kunze on the bus of virtual 'voluptas necessitatibus' \(aka SQL\)](#)

Both server-side and client-side rendering use **GraphQL** and **Handlebars**.

GraphiQL and similar clients supported out-of-the-box.
Clean **JSON** rendering available via GraphQL.

[sling-samples/
org.apache.sling.graphql.samples.website](https://sling-samples.apache.org/apache.sling.graphql.samples.website)

Client-side query + rendering



Travel Music Adventure News Culture Business Search

Search in articles

virtual Search

Travel Music Adventure News Culture Business ← Click to select/deselect content sections

Found 60 articles containing "virtual" in the selected content sections

- Travel - Precious Kunze on the bandwidth of virtual 'nobis id' (aka USB)
- Travel - Claudia Littell on the microchip of virtual 'quis facere' (aka USB)
- Travel - Michelle Koss on the array of virtual 'aliquam quis' (aka XML)
- Travel - Carroll Keebler on the feed of virtual 'aperiam sit' (aka TCP)
- Travel - Lionel Champlin on the capacitor of virtual 'et dolore' (aka PCI)
- Travel - Josefina Schmidt on the bandwidth of virtual 'quisquam porro' (aka JSON)
- Travel - Gina Ruecker on the protocol of virtual 'et et' (aka GB)
- Travel - Floy Gislasen on the card of virtual 'quae veniam' (aka GB)
- Travel - Catalina Cassin on the circuit of virtual 'doloremque et' (aka SMTP)
- Music - Wellington Russel on the array of virtual 'impedit sed' (aka SCSI)
- Music - Dorian Feeney on the interface of virtual 'omnis laudantium' (aka AI)

```
const query = `{
  navigation {
    search
    sections {
      path
      name
    }
  }
  article(withText: "${searchText}") {
    path
    title
    section
  }
}`;
```

POST GraphQL query

```
<div id="resultsTemplate">
  {{#if data.result.article}}
    <h2>
      Found {{data.result.article.length}} articles
      containing "{{data.info.searchText}}"
      in the selected content sections
    </h2>
    <ul>
      {{#each data.result.article}}
        <li class="articleLink">
          <a href="{{this.path}}.html">{{this.title}}</a>
        </li>
      {{/each}}
    </ul>
  {{else}}
    <div class="message">No articles found.</div>
  {{/if}}
</div>
```

**JSON results
-> Handlebars**

Client Sling Server

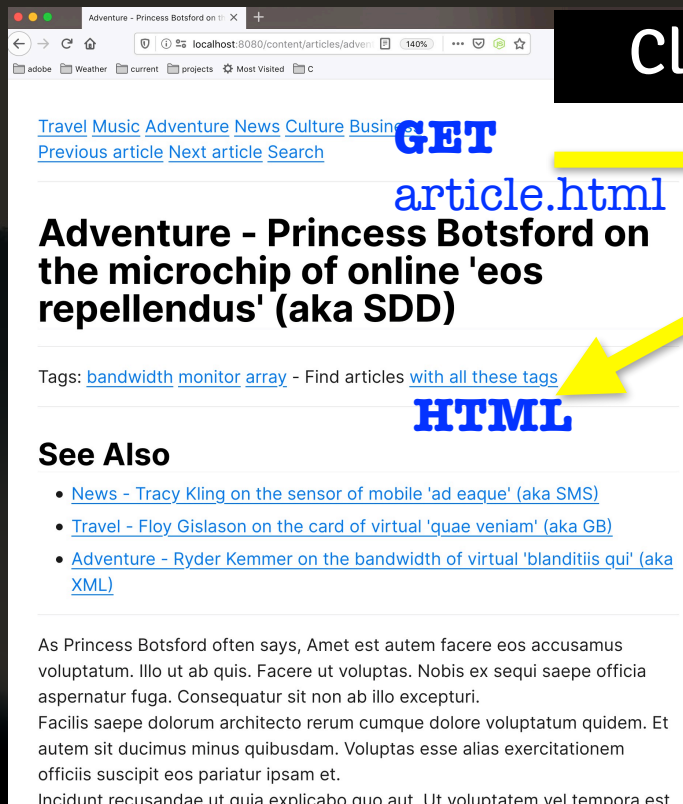
GraphQL
Servlet

Content
Repository



Server-side query + rendering

The **Handlebars** engine renders the **.json** output of the **GraphQL** query



Client Sling Server

Sling Scripts Resolver

article.hbs script

Handlebars Script Engine

Internal Request
<path>.json

json.GQL script

GraphQL Script Engine

Internal Request
<path>.GQLSchema

Content Repository





adaptTo()

Sling GraphQL Core

Sling GraphQL Core 0.0.4

package

o.a.s.graphql.core.api

SchemaProvider

SlingDataFetcher<T>

SlingDataFetcherEnvironment

SlingScalarConverter<T,X>

API Script Engine Servlet

Apache Sling Web Console Script Engines

Available Script Engines

=====

Apache Sling GraphQL Core 0.0.4

- Language : GraphQL, Sling:GraphQL:0.1
- Extensions : gql
- MIME Types : text/x-graphql
- Names : GraphQL, graphql

[javax.servlet.Servlet]	
component.id	301
component.name	org.apache.sling.graphql.core.GraphQLServlet
service.bundleid	170
Service Description	Sling GraphQL Servlet
service.factoryPid	org.apache.sling.graphql.core.GraphQLServlet
Service PID	org.apache.sling.graphql.core.GraphQLServlet~default
service.scope	bundle
Service Vendor	The Apache Software Foundation
sling.servlet.extensions	json
sling.servlet.methods	GET POST
sling.servlet.resourceTypes	samples/servlet
Using Bundles	org.apache.sling.servlets.resolver (168)

Internally uses

<https://github.com/graphql-java/graphql-java>



Default SchemaProvider

```
$ curl http://localhost:8080/content/articles/culture.GQLSchema
```

```
type Query {  
  section: Section @fetcher(name:"website/currentResource")  
  navigation: Navigation @fetcher(name:"website/navigation")  
}
```

```
type Navigation {  
  root: String  
  sections: [Section]  
  previous: String  
  next: String  
  search: String  
}
```

...

Full power of the **Sling rendering pipeline** for resource-specific schemas.

The **@fetcher** directive points to **SlingDataFetcher** services.



adapTto()



SlingDataFetcher

```
@Component(service = SlingDataFetcher.class, property = {"name=website/navigation"})
public class NavigationDataFetcher implements SlingDataFetcher<Object> {

    @Override
    public Object get(SlingDataFetcherEnvironment env) throws Exception {
        final Map<String, Object> result = new HashMap<>();
        result.put("root", Constants.ARTICLES_ROOT);
        result.put("sections", getSections(env.getCurrentResource()));
        result.put("search", Constants.SEARCH_PAGE_PATH);
        maybeAddPrevNext(result, FetcherUtil.getSourceResource(env));
        return result;
    }
}
```

Server-side script:
selected by fetcher name,
returns a Map

```
var result = {
    boolValue: true,
    resourcePath: "From the test script: " + resource.path,
    testingArgument: environment.getArgument("testing"),
    anotherValue: 450 + 1
};
```

Java:
SlingDataFetcher OSGi
service, returns a
Map or POJO



Under the hood: performance?

Client | Sling Server

```
GraphQL Endpoint http://localhost:8080/graphql.json

GraphQL [Prettify] [History]

{
  navigation {
    search
    sections {
      path
      name
    }
  }
  articleOutText: "Devan" {
    path
    title
    section
    sectionLine
    title
    path
  }
}

QUERY VARIABLES
```

Sling GraphQL Core
Request Handling
Schema Acquisition

Schema Generation
(Sling Request Processing)

graphql-java
parsing + delegation
www.graphql-java.com

SlingDataFetcher Services
Data Acquisition + Processing
Caching?

Performance
Happens Here ->

Content
Repository

Other
Data Sources





adaptTo()

Caching?

HTTP-friendly caching?

Client | Sling Server

POST query text

201 Created -> /p/bc6f

future requests
can start here:

GET /p/bc6f

200 OK -> query results
+ Cache-Control headers

GraphQL
Servlet

Queries
Store

Stored
Query
Text

Default POST-based
interaction is generally
not cacheable.





adaptTo()

GraphQL and REST, BFFs ?

GraphQL and REST (+ RPC?)

Quoting Phil Sturgeon, builder of API design tools, <https://phil.tech/2017/graphql-vs-rest-overview/>

Why Not Use Both?

The biggest oddity I notice in the “GraphQL vs REST” conversation, is the falsehood that *you must pick one*.

In a world of SoA, you are likely to have multiple services, which expose multiple APIs. In the [RPC vs REST](#) article I point out that some services might be REST and some might be RPC, and you can absolutely throw some GraphQL in with your REST.

We do need **queries**, but also **caching**, **hypertext**, **scalability**.

GraphQL is certainly better than a badly designed so-called REST API...and it's a great **query** API.

GraphQL

From Wikipedia, the free encyclopedia

GraphQL is an open-source data [query](#) and [manipulation](#) language for [APIs](#), and a runtime for fulfilling queries with existing data.^[2] GraphQL was

Representational state transfer

From Wikipedia, the free encyclopedia

Representational state transfer (REST) is a [software architectural](#) style that defines a set of constraints to be used for creating [Web services](#). Web

RESTful Web services allow the requesting systems to access and manipulate textual representations of [Web resources](#) by using a uniform and predefined set of [stateless](#) operations. Other kinds of Web services, such as [SOAP](#) Web



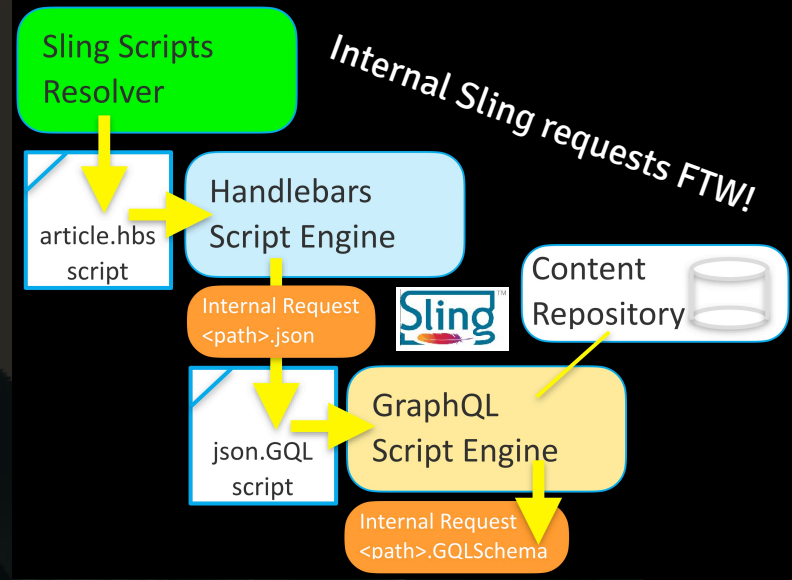
adaptTo()

CODA

CODA

GraphQL is a very nice **query language** that also looks **useful on the server-side**.

It's not better or worse than REST: **different animals** that can **play well together!**



The Sling GraphQL Core is V0.0.4:
patches welcome!



I'm @bdelacretaz - thank you!

Code at <https://github.com/apache/sling-org-apache-sling-graphql-core/>

Fun text by <https://github.com/apache/sling-whiteboard/tree/master/fake-content-generator>

