

Adopting a GraphQL Mindset When Migrating From REST



Michael Watson
Developer Advocate - Apollo Federation

Twitter: [@threebrewmates](https://twitter.com/threebrewmates)
GitHub: [michael-watson](https://github.com/michael-watson)

Housekeeping



Q&A: Be sure to ask your questions in the Q&A box, we will have some time at the end to answer questions.



Post-event survey: Let us know what improvements you are most interested in seeing in the post-event survey.



Recording: The talk will be recorded and sent out in the next few days.



Apollo's next event: Don't forget to register for our next tech talk ***Establishing Your Schema Governance Best Practices*** on August 12th at 10am PT. We will drop the link in the chat.

Talk topics



Why GraphQL?



Should I use Apollo Federation?



Don't make your graph “REST-y”

Questions



Demo: Using Apollo Server to expose your REST capabilities

Questions

Why GraphQL?

- Everybody is doing it
- Better digital experiences
- One graph for the capabilities of your business

glassdoor

Zillow



Adobe

OpenTable

PayPal



NETFLIX

RE/MAX

SONY

nerdwallet

okcupid



DOORDASH

PELOTON

Atlassian

flexport.

Dior

airbnb

The
New York
Times

Square



expedia
group

STITCH FIX

intuit

wayfair

Should I use Apollo Federation?

No*

**Assuming you haven't run GraphQL in production yet*

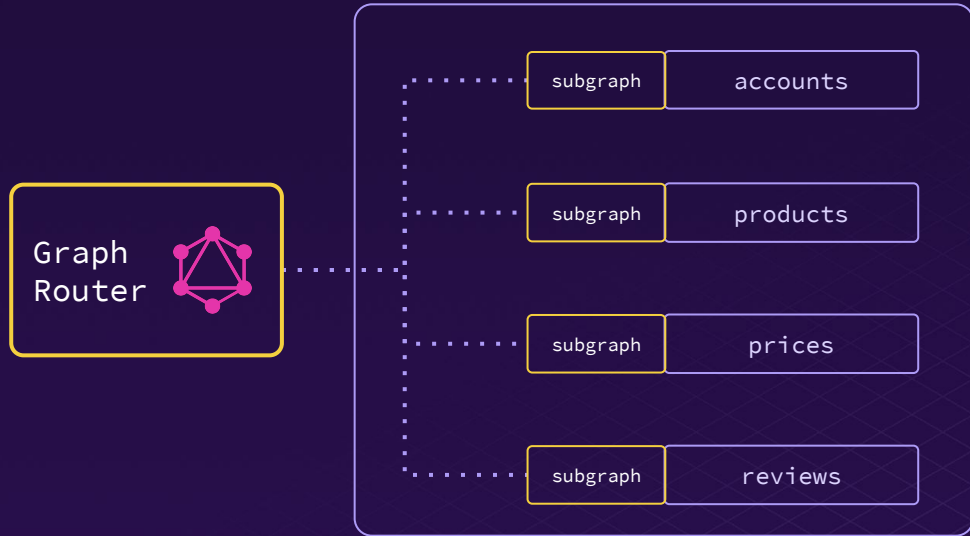
But why?

- You're going to have to learn how to get GraphQL into your production environment
 - This can be a template for Apollo Federation in the future, but adding Apollo Federation on top of this learning curve will be more challenging
- Apollo Federation benefits really kick in when you have 2 or more servers
 - Should you really try and host multiple GraphQL servers in production when we haven't run one in production successfully yet?
 - Get 1 GraphQL server working, then Apollo Federation is an easy next step

But why?*

**I have seen companies successfully use Apollo Federation as their first GraphQL deployment in production, but they understood the additional learning curve and planned resources accordingly*

Should I use Apollo Federation? TL;DR



- You don't need to decide right away (it can be federated in the future!)
- Simplify your first project and start with 1 domain/boundary of the REST API
- A future phase can build a second subgraph and Federation can then be used

Remember, you need 2 GraphQL servers to Federate together. Start out with building just your first GraphQL implementation and have Apollo Federation as a future option

Don't make your graph "REST-y"

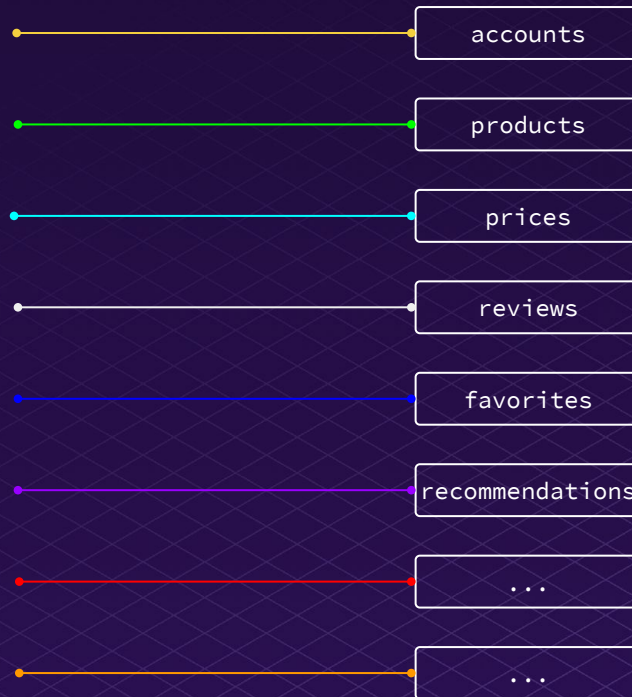
Apps



Distribution

?

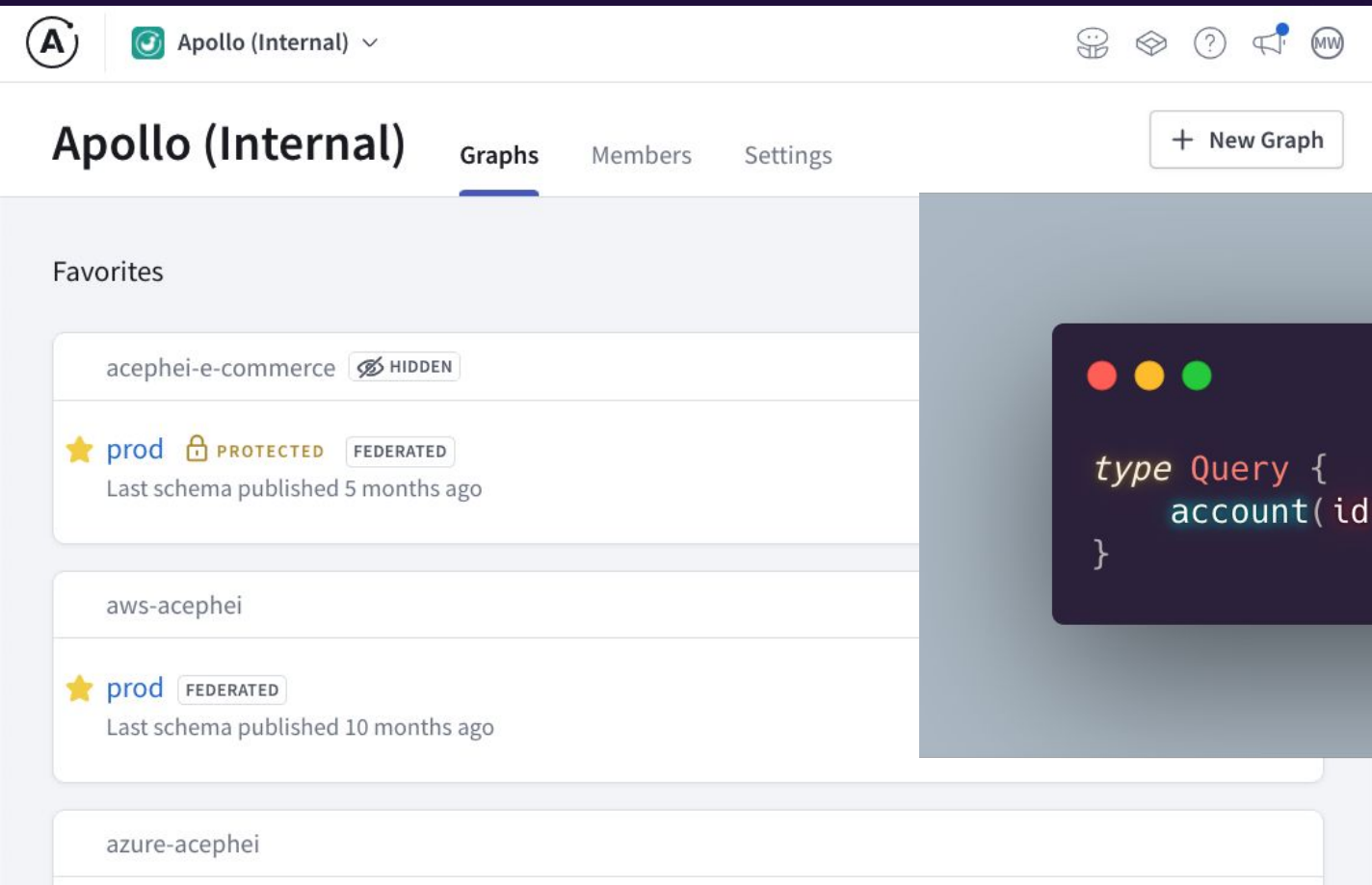
Services



Hint: avoid CRUD

Capabilities

Let's take Accounts as an example



The screenshot shows the Apollo (Internal) dashboard. At the top, there's a header with the Apollo logo, a dropdown menu for 'Apollo (Internal)', and several utility icons. Below the header, there's a navigation bar with 'Apollo (Internal)', 'Graphs', 'Members', and 'Settings'. A '+ New Graph' button is on the right. The main content area is titled 'Favorites' and lists three GraphQL graphs:

- acephei-e-commerce** (HIDDEN)
- prod** (PROTECTED, FEDERATED) - Last schema published 5 months ago
- aws-acephei**
- prod** (FEDERATED) - Last schema published 10 months ago
- azure-acephei**



```
type Query {  
  account(id:ID!): Account  
}
```

Let's take Accounts as an example



Get started with Apollo Studio

Apollo Studio is the single source of truth for your graph. Register your schema, design and explore your graph, and optimize its performance with built-in observability.



Sign up using GitHub


Sign up using email






Enterprise customers, [sign in with Enterprise SSO](#).

Already have an account? [Sign in](#).

Let's take Accounts as an example

A

 Apollo (Internal) ▾



Apollo (Internal)

GraphsMembersSettings

+ New Graph

Organization


Organization Name

Apollo (Internal)

Update name

Organization Logo

Your current organization logo.



Upload image

Organization ID

Update ID

Let's take Accounts as an example



Apollo (Internal) ▾

Apollo (Internal)


Graphs

Members

Delete this organization

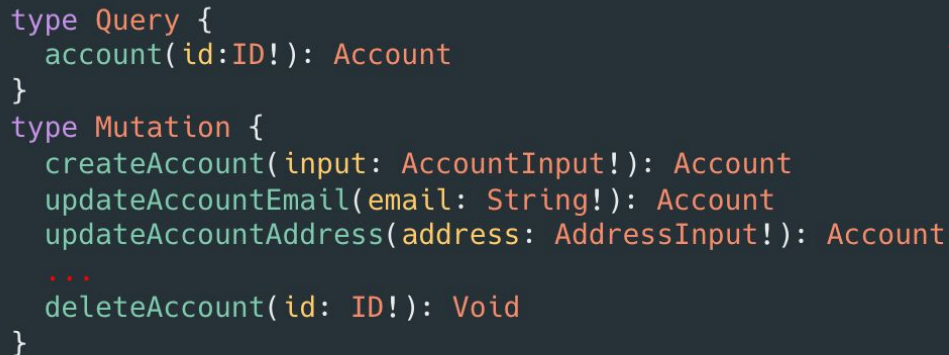
Deleting your organization will delete its graphs and remove its user memberships.
This cannot be undone.

Delete this organization



```
type Query {  
  account(id:ID!): Account  
}  
type Mutation {  
  createAccount(input: AccountInput!): Account  
  updateAccount(account: AccountInput!): Account  
  deleteAccount(id: ID!): Void  
}
```


Better, but still “REST-y”



```
type Query {  
  account(id:ID!): Account  
}  
type Mutation {  
  createAccount(input: AccountInput!): Account  
  updateAccountEmail(email: String!): Account  
  updateAccountAddress(address: AddressInput!): Account  
  ...  
  deleteAccount(id: ID!): Void  
}
```

Make it about the consumer!



dev

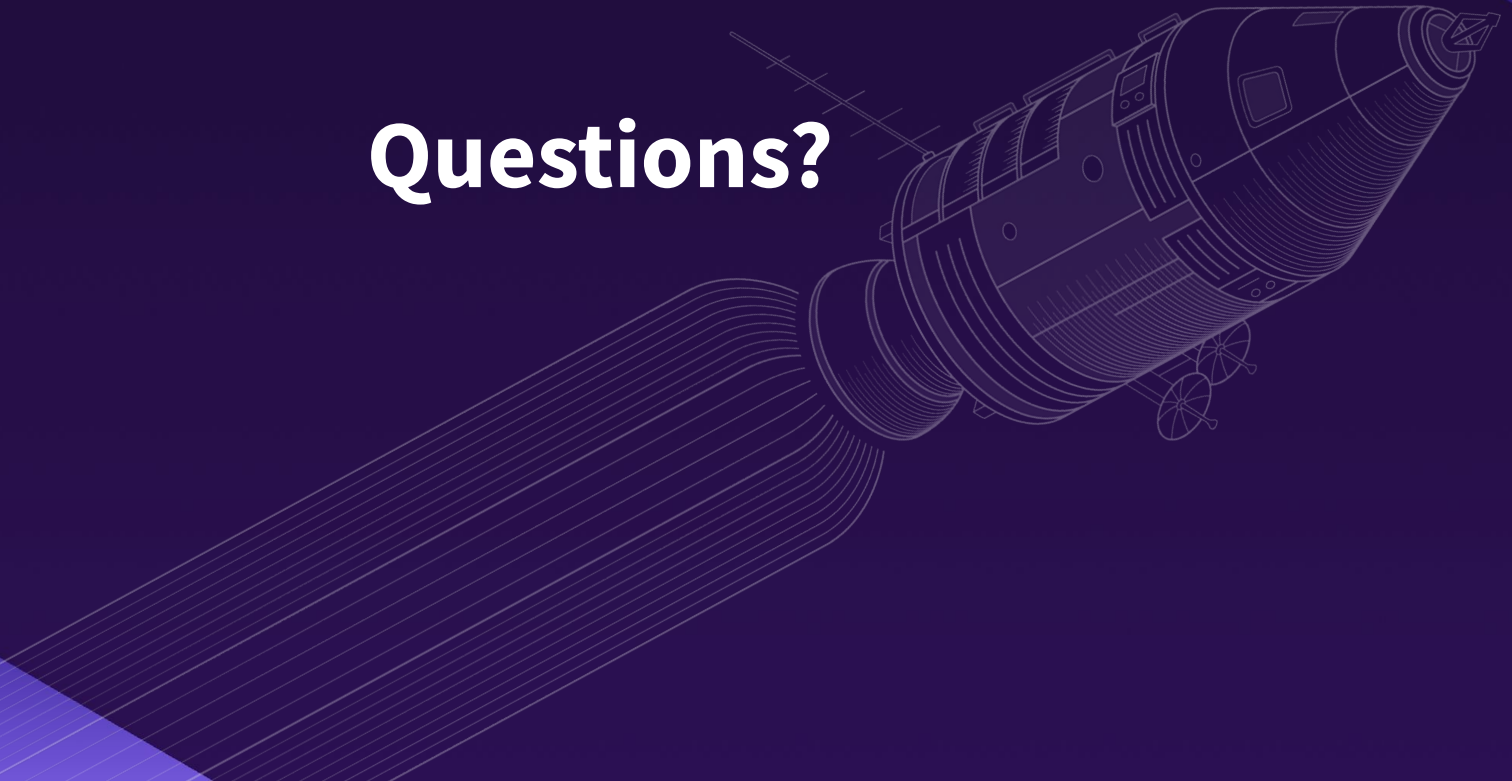
Example of what we do at Apollo in our Graph...



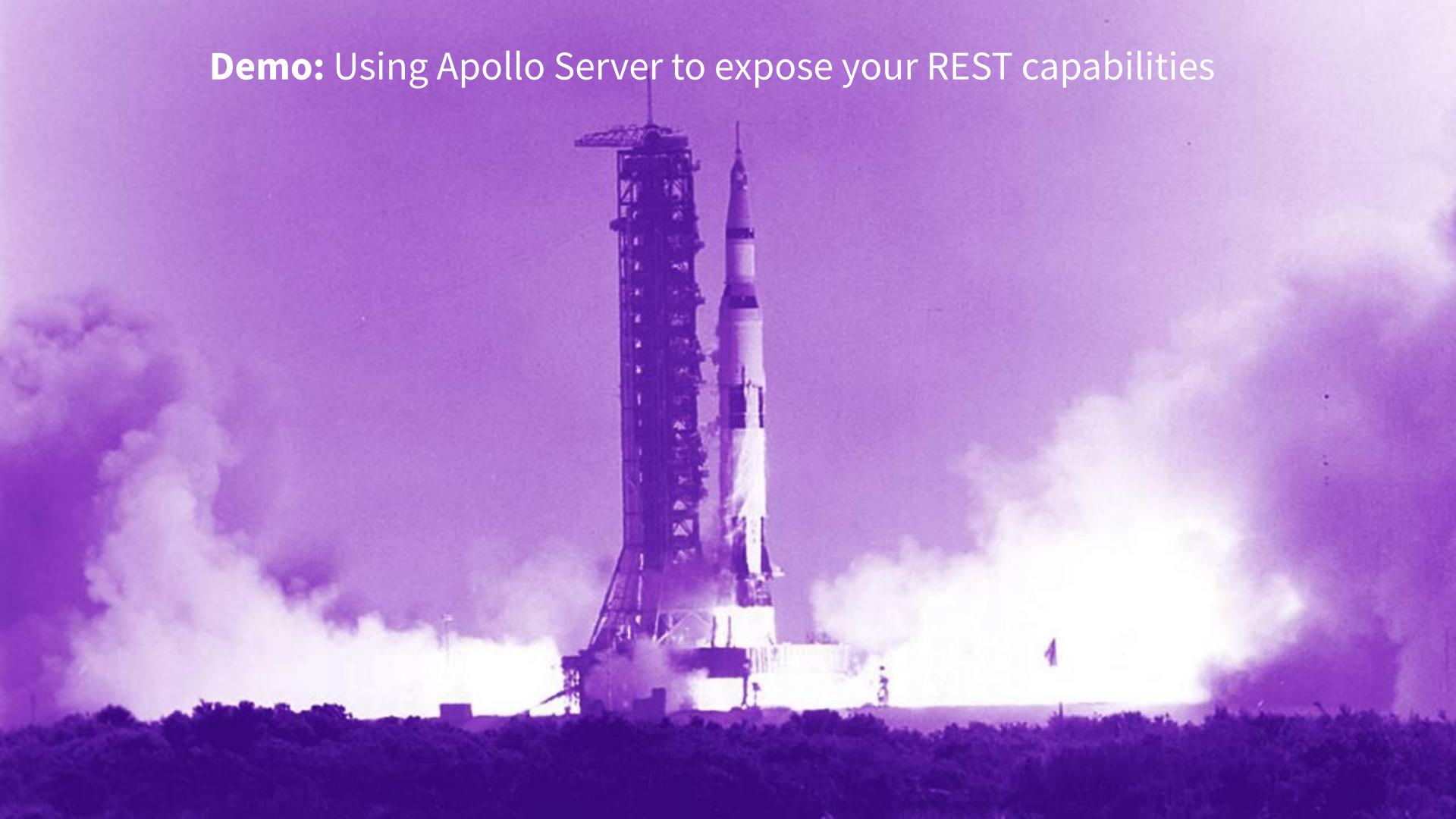
```
type Query {  
  account: Account #ID is inferred from bearer token  
}  
type Mutation {  
  account: Account #ID is inferred from bearer token  
}  
  
type AccountMutation {  
  updateEmail(email: String!): Account  
  addAddress(address: AddressInput!): Account  
  addAddressAsDefault(address: AddressInput!): Account  
  setDefaultShippingAddress(addressID: ID!): Account  
}
```



Questions?



Demo: Using Apollo Server to expose your REST capabilities



Getting Started

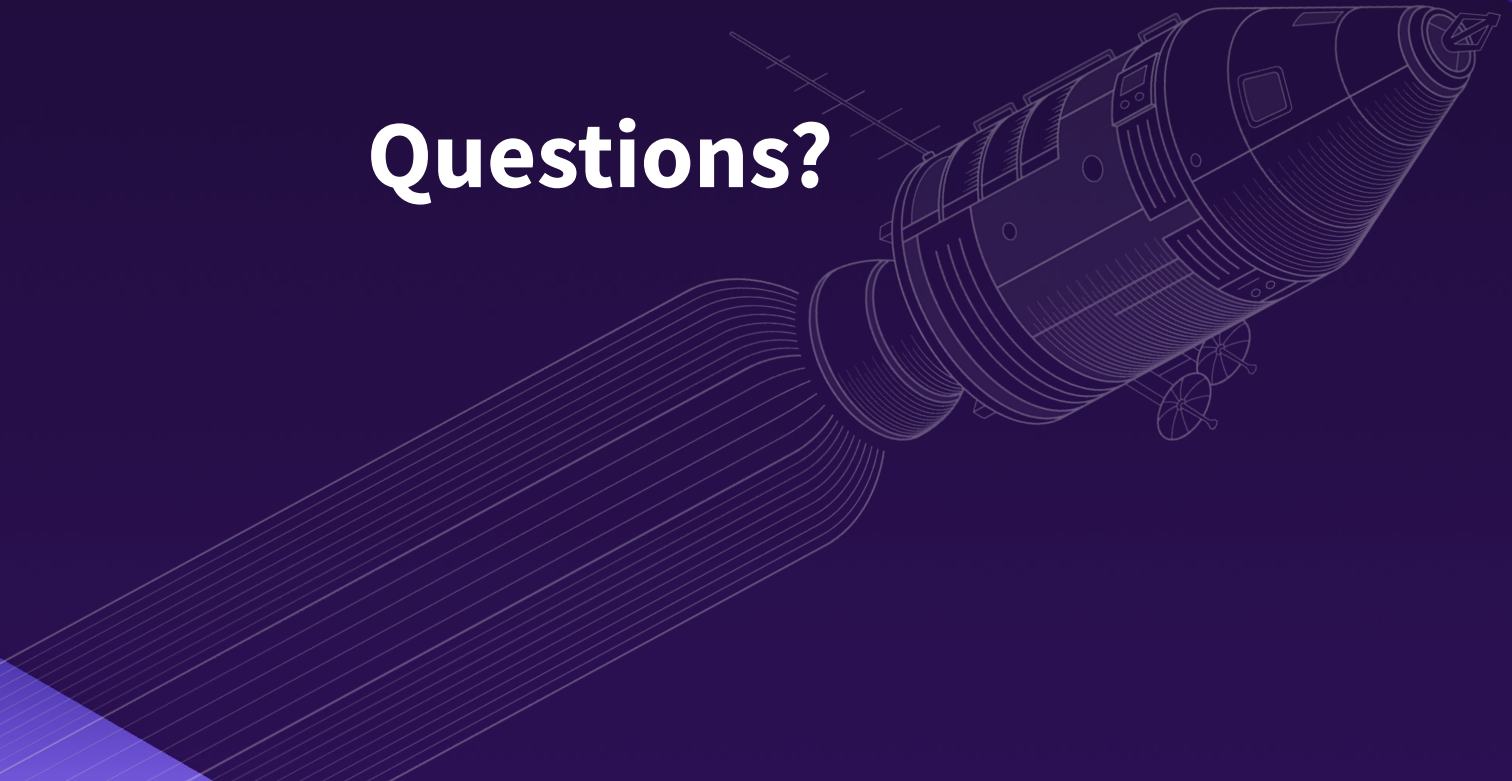
- [Docs: Getting started with Apollo Server](https://www.apollographql.com/docs/apollo-server/getting-started/)
<https://www.apollographql.com/docs/apollo-server/getting-started/>
- [Docs: Manage your graph with the Apollo Studio](https://www.apollographql.com/docs/studio/)
<https://www.apollographql.com/docs/studio/>

If you are going to use Apollo Federation...

- [Docs: Subgraphs in Apollo Federation](https://www.apollographql.com/docs/federation/subgraphs/) (the example uses Apollo Server)
<https://www.apollographql.com/docs/federation/subgraphs/>



Questions?



Thank You!



Michael Watson
Principal Solutions Architect

Twitter: [@threebrewmates](https://twitter.com/threebrewmates)

GitHub: [michael-watson](https://github.com/michael-watson)

