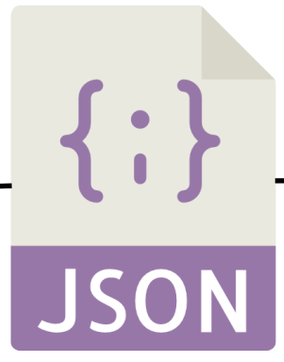
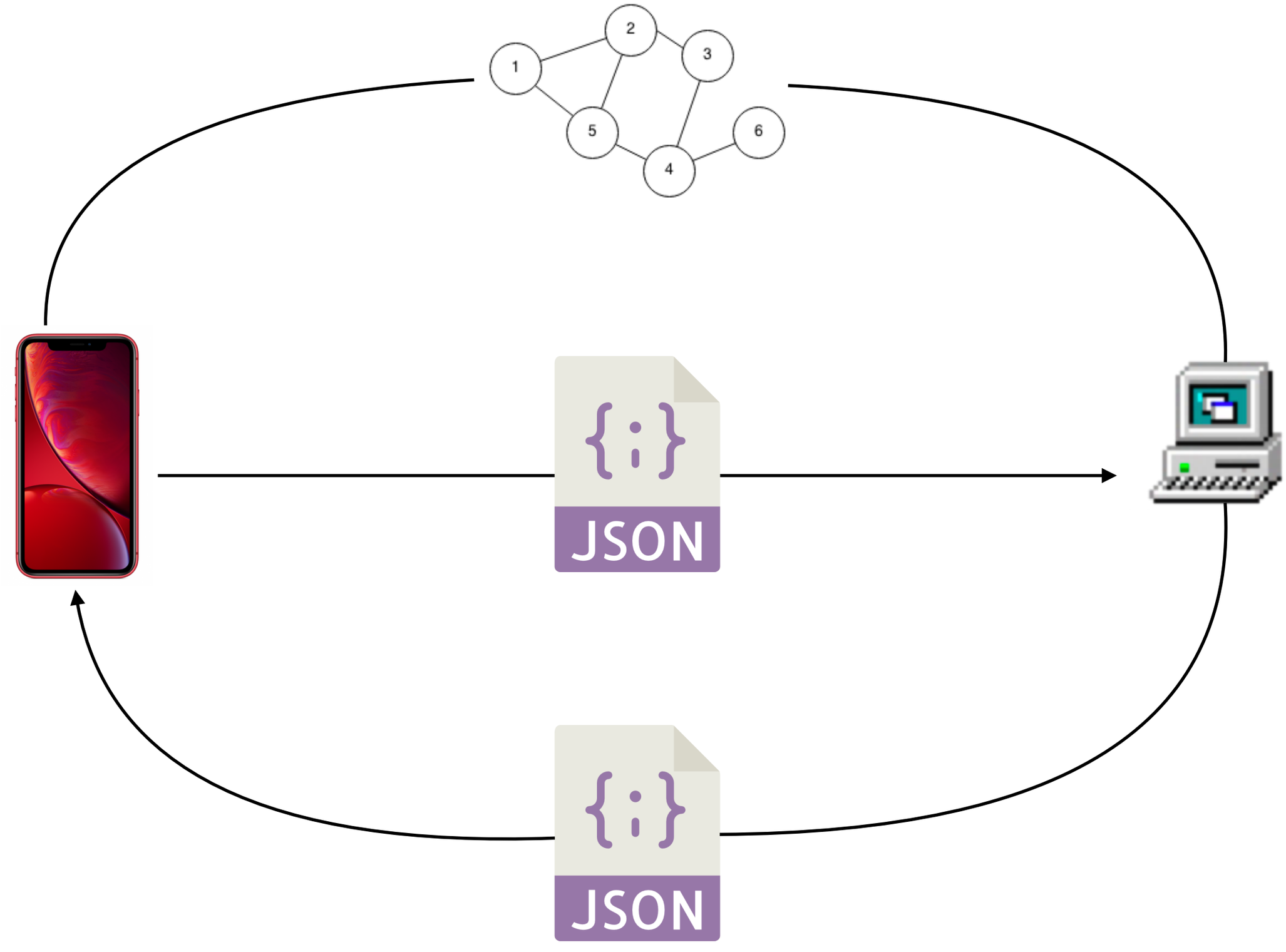
GraphQL

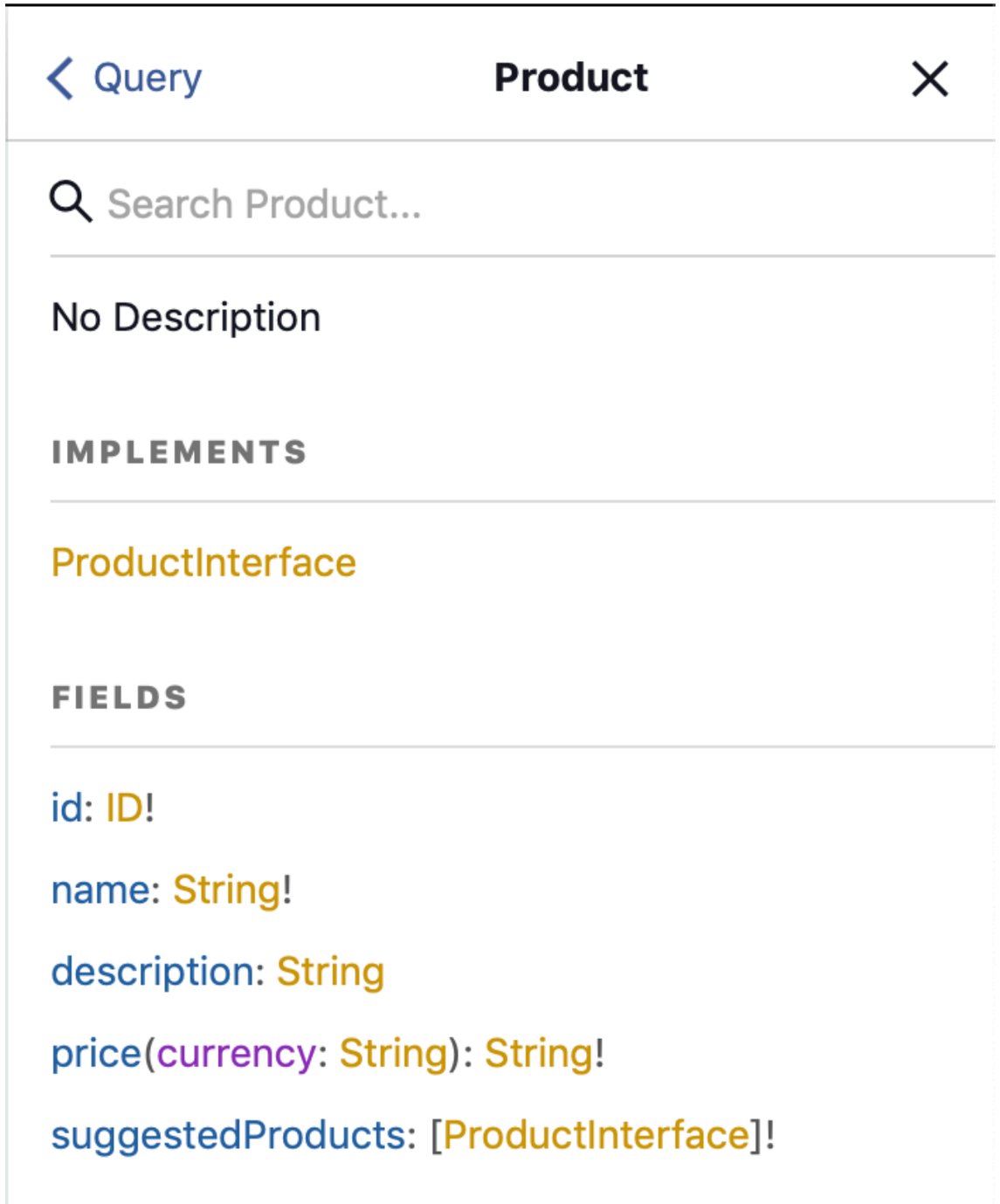
Intro

https://domain/posts/{id}




1
2
3
4
5
6
JSON
JSON

Schema


Query
Product
Search Product...
No Description
IMPLEMENTS
ProductInterface
FIELDS
id: ID!
name: String!
description: String
price(currency: String): String!
suggestedProducts: [ProductInterface]!

Schema


```
"kind": "OBJECT",
"name": "Product",
"description": "",
"fields": [
  {
    "name": "id",
    "description": "",
    "args": [],
    "type": {
      "kind": "NON_NULL",
      "name": null,
      "ofType": {
        "kind": "SCALAR",
        "name": "ID",
        "ofType": null
      }
    },
    "isDeprecated": false,
    "deprecationReason": null
  }
```

Schema


```
"name": "suggestedProducts",
"description": "",
"args": [],
"type": {
  "kind": "NON_NULL",
  "name": null,
  "ofType": {
    "kind": "LIST",
    "name": null,
    "ofType": {
      "kind": "INTERFACE",
      "name": "ProductInterface",
      "ofType": null
    }
  }
},
"isDeprecated": false,
"deprecationReason": null
```

Schema


```
"kind": "OBJECT",
"name": "Query",
"description": "",
"fields": [
  {
    "name": "product",
    "description": "",
    "args": [
      {
        "name": "id",
        "description": "",
        "type": {
          "kind": "NON_NULL",
          "name": null,
          "ofType": {
            "kind": "SCALAR",
            "name": "ID",
            "ofType": null
          }
        },
        "defaultValue": null
      }
    ],
  }
]
```

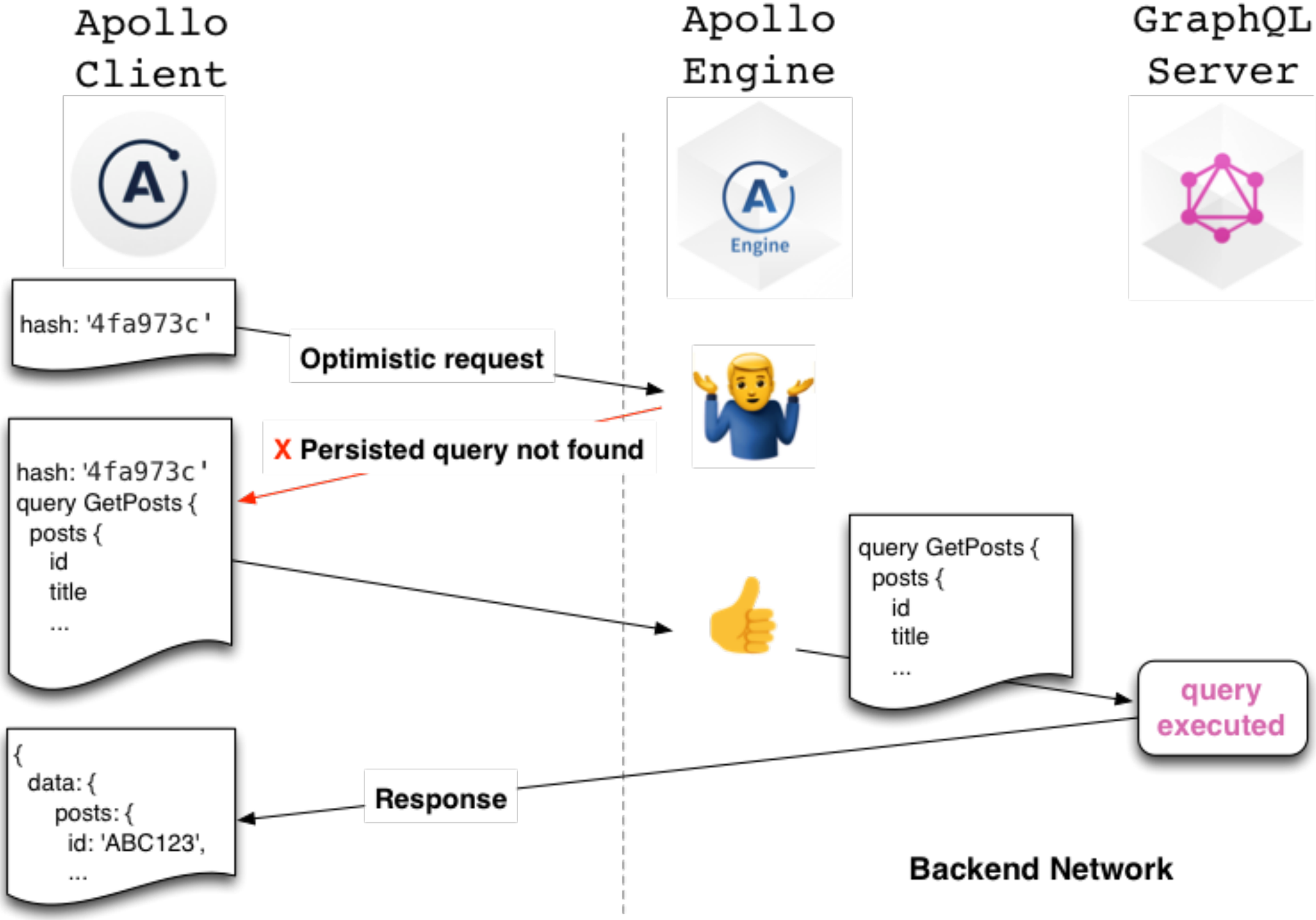
Request

```
query GetDetailFragmentProduct($id: ID!) {
    product(id: $id) {
        ... ProductFragment
        suggestedProducts {
            ... ProductFragment
            ... on RelatedProduct {
                commonTags
            }
            ... on PromotionalProduct {
                discount
            }
        }
    }
    user {
        discount
        age
    }
}

fragment ProductDetails on ProductInterface {
    id
    name
    description
    price(currency: "USD")
}
```

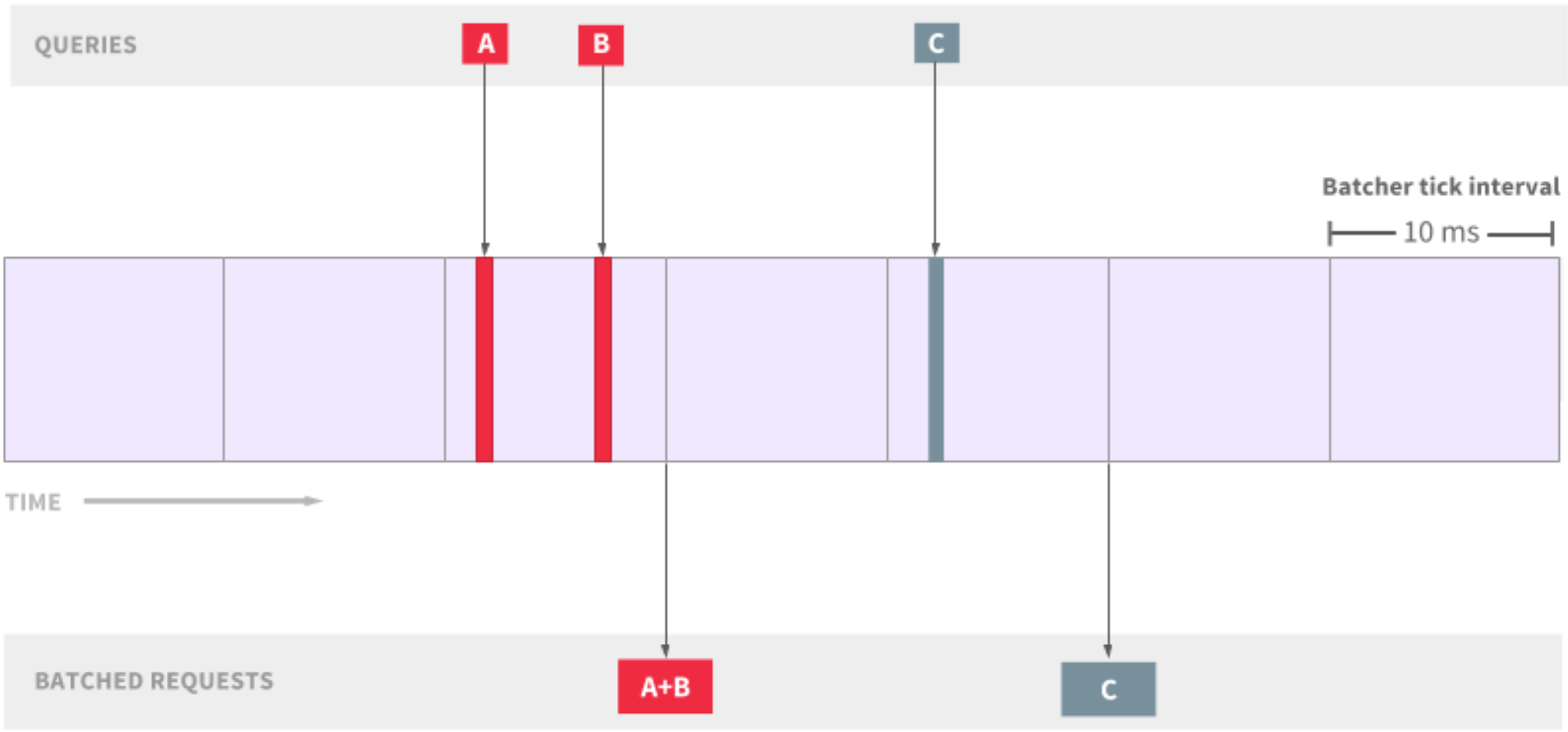
API.swift

```swift
public final class GetFragmentProductQuery: GraphQLQuery {
  public let operationDefinition =
    "query GetFragmentProduct($id: ID!) {\n  product(id: $id) {\n    __typename\n    ...ProductFragment\n
suggestedProducts {\n      __typename\n      ...ProductFragment\n    }\n  }\n}"

  public var queryDocument: String { return operationDefinition.appending(ProductFragment.fragmentDefinition) }

  public var id: GraphQLID

  public init(id: GraphQLID) {
    self.id = id
  }

  public var variables: GraphQLMap? {
    return ["id": id]
  }

  public struct Data: GraphQLSelectionSet {
    public static let possibleTypes = ["Query"]

    public static let selections: [GraphQLSelection] = [
      GraphQLField("product", arguments: ["id": GraphQLVariable("id")], type: .object(Product.selections)),
    ]

    public private(set) var resultMap: ResultMap

    public init(unsafeResultMap: ResultMap) {
      self.resultMap = unsafeResultMap
    }

    public init(product: Product? = nil) {
      self.init(unsafeResultMap: ["__typename": "Query", "product": product.flatMap { (value: Product) -> ResultMap in
value.resultMap }])
    }

    public var product: Product? {
      get {
        return (resultMap["product"] as? ResultMap).flatMap { Product(unsafeResultMap: $0) }
      }
      set {
        resultMap.updateValue(newValue?.resultMap, forKey: "product")
      }
    }
```

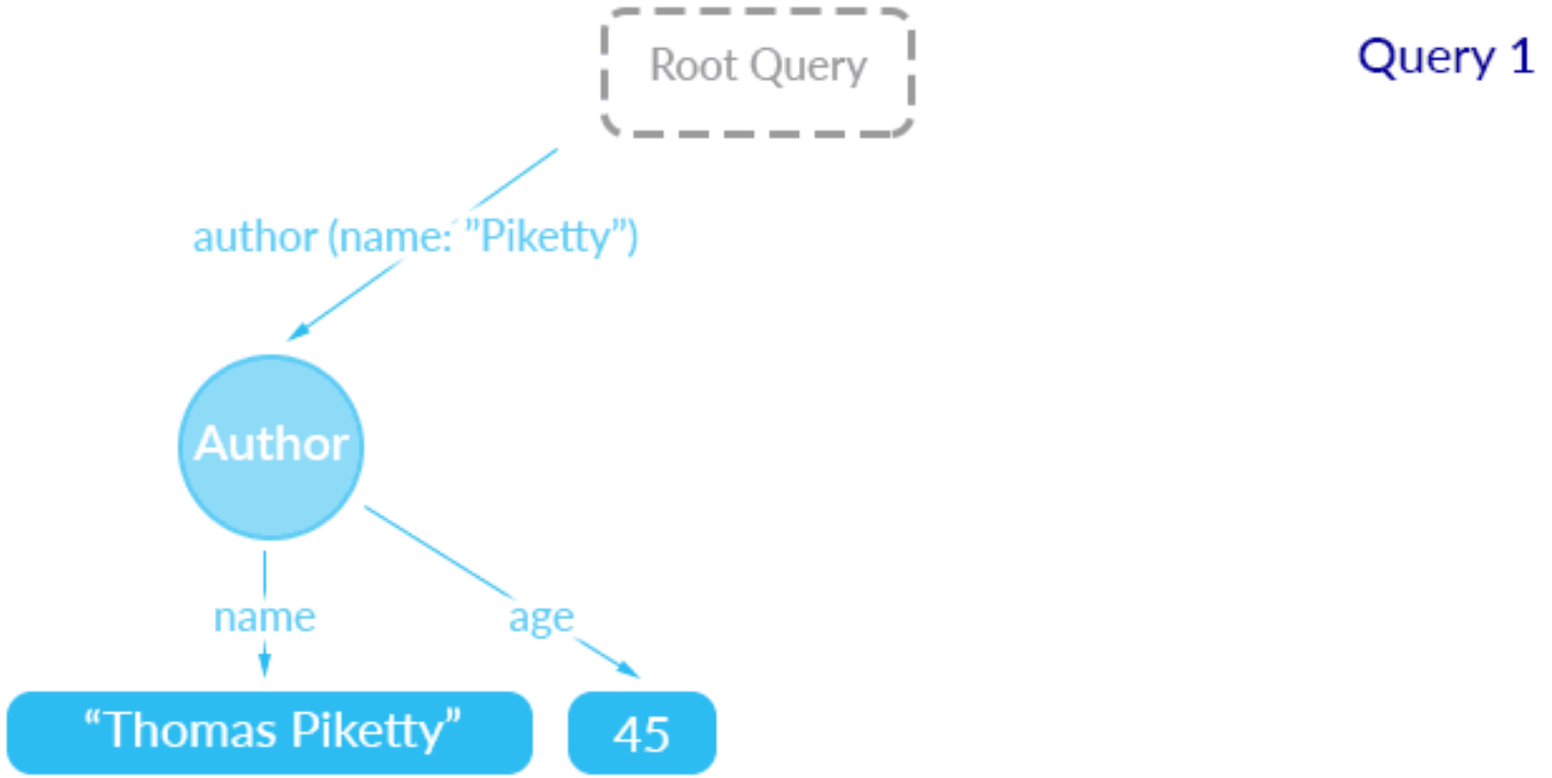
Request body hashing



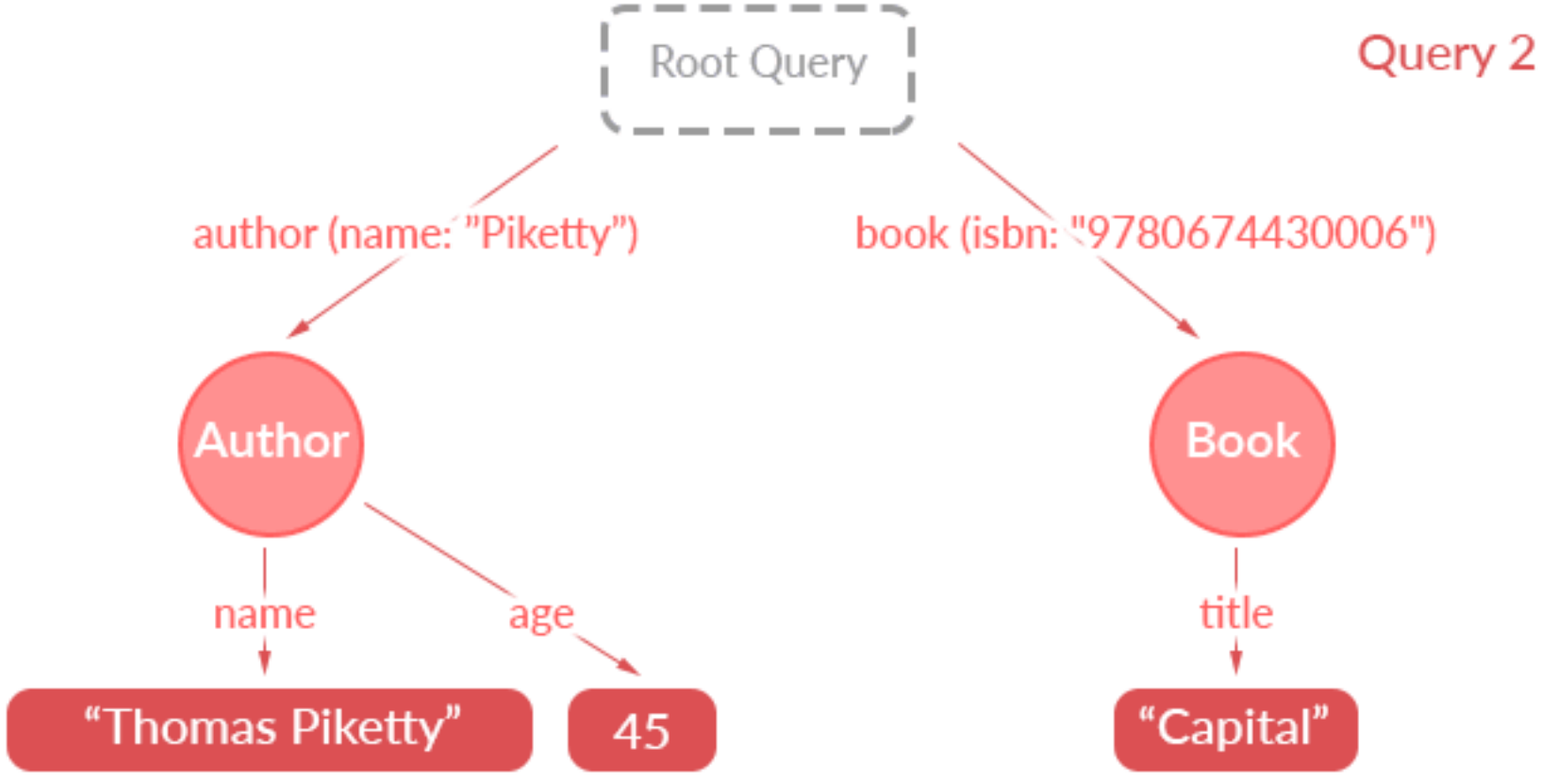
Batching


QUERIES
A
B
C
Batcher tick interval
10 ms
TIME
BATCHED REQUESTS
A+B
C

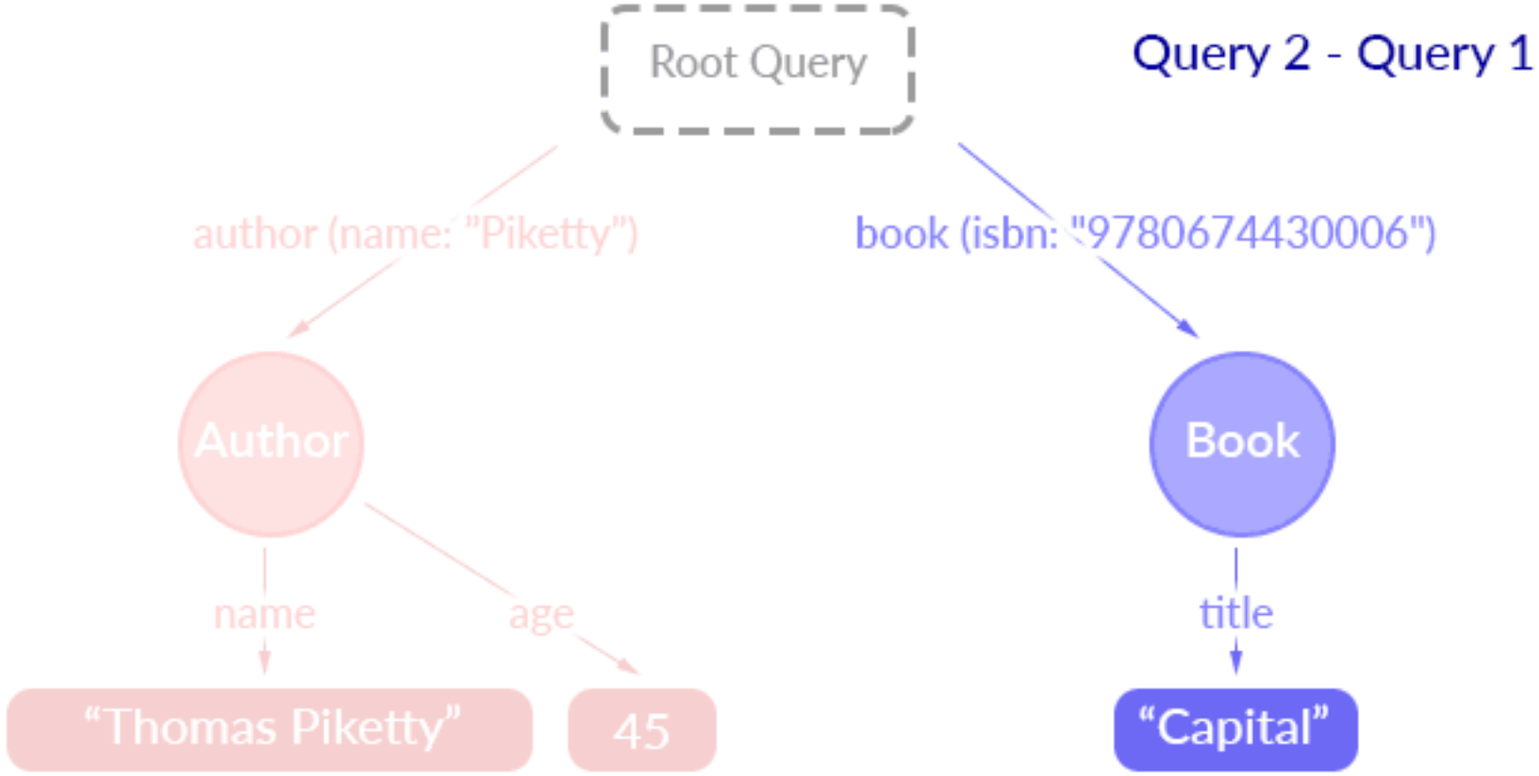
Cache


Root Query
Query 1
author (name: "Piketty")
Author
name
age
"Thomas Piketty"
45

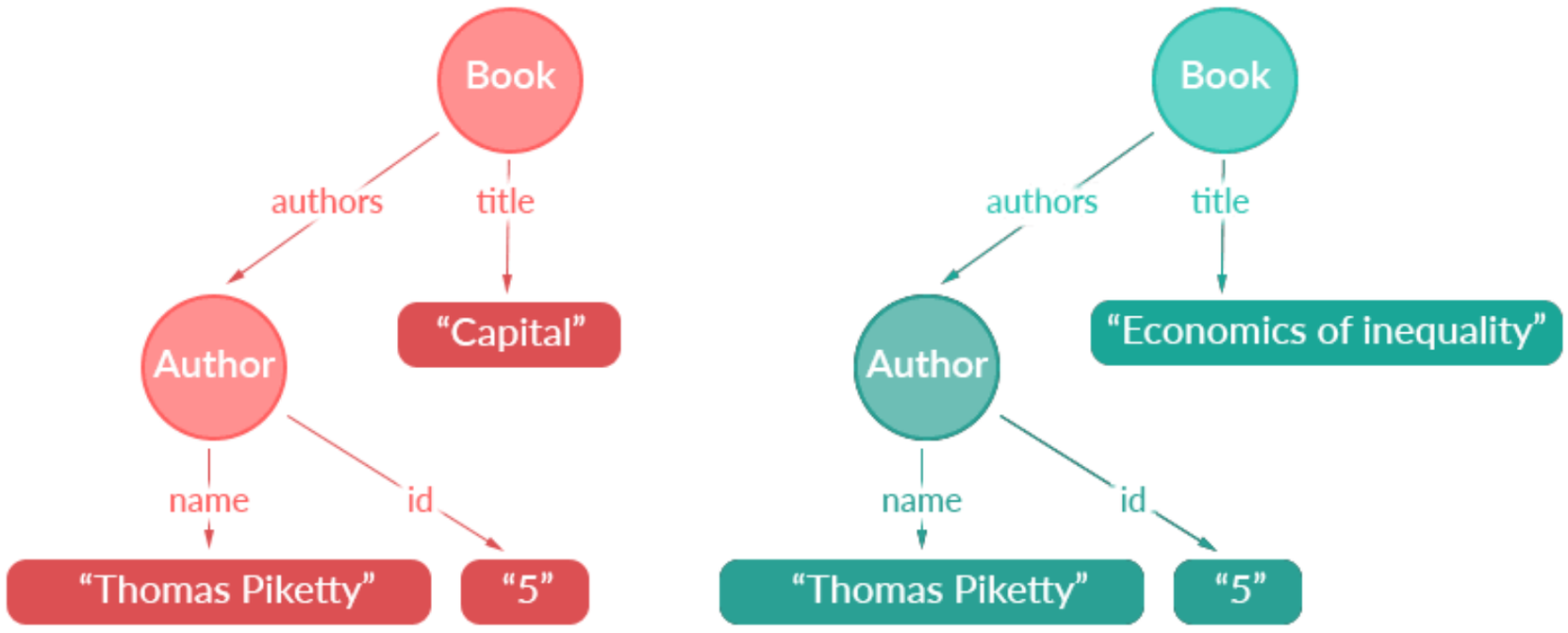
Cache


Root Query
Query 2
author (name: "Piketty")
book (isbn: "9780674430006")
Author
Book
name
age
title
"Thomas Piketty"
45
"Capital"

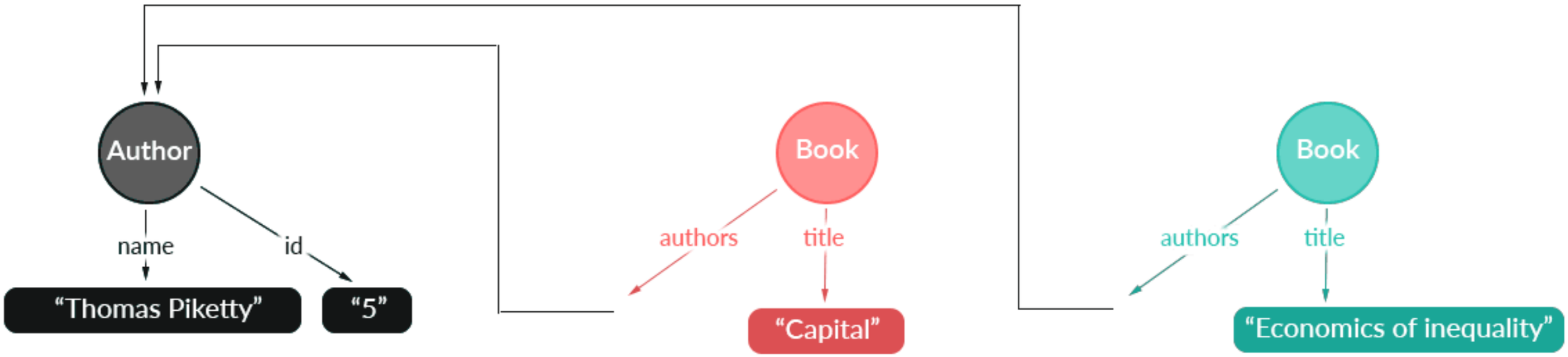
Cache


Root Query
Query 2 - Query 1
author (name: "Piketty")
book (isbn: "9780674430006")
Author
Book
name
age
title
"Thomas Piketty"
45
"Capital"

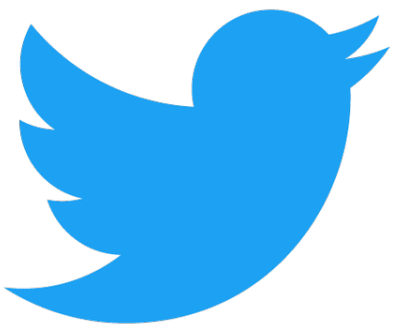
Cache


Book
authors
title
"Capital"
Author
name
id
"Thomas Piketty"
"5"
Book
authors
title
"Economics of inequality"
Author
name
id
"Thomas Piketty"
"5"

Cache


Author
name
id
"Thomas Piketty"
"5"
Book
authors
title
"Capital"
Book
authors
title
"Economics of inequality"


Product Hunt




PayPal


yelp

Fin