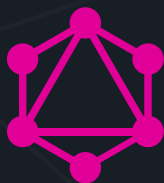


[学习](#) [代码](#) [社区](#) [博客](#) [规范](#)



GraphQL

描述你的数据

```
type Project {  
  name: String  
  tagline: String  
  contributors: [User]  
}
```

请求你所要的数据

```
{  
  project(name: "GraphQL") {  
    tagline  
  }  
}
```

得到可预测的结果

```
{  
  "project": {  
    "tagline": "A query language for APIs"  
  }  
}
```

[马上开始](#)

[了解更多](#)

一种用于 API 的查询语言

GraphQL 既是一种用于 API 的查询语言也是一个满足你数据查询的运行时。GraphQL 对你的 API 中的数据提供了一套易于理解的完整描述，使得客户端能够准确地获得它需要的数据，而且没有任何冗余，也让 API 更容易地随着时间推移而演进，还能用于构建强大的开发者工具。

请求你所要的数据 不多不少

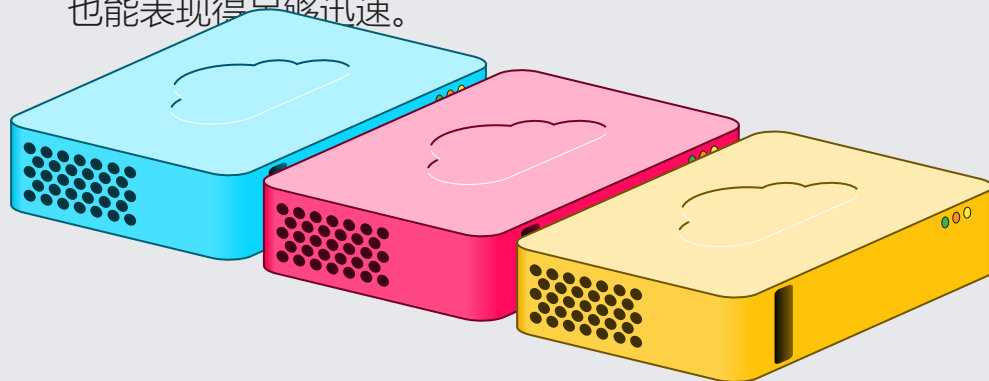
向你的 API 发出一个 GraphQL 请求就能准确获得你想要的数据库，不多不少。GraphQL 查询总是返回可预测的结果。使用 GraphQL 的应用可以工作得又快又稳，因为控制数据的是应用，而不是服务器。

```
{
  hero {
    name
    height
    mass
  }
}
```

```
{
  "hero": {
    "name": "Luke Skywalker",
    "height": 1.72,
    "mass": 77
  }
}
```

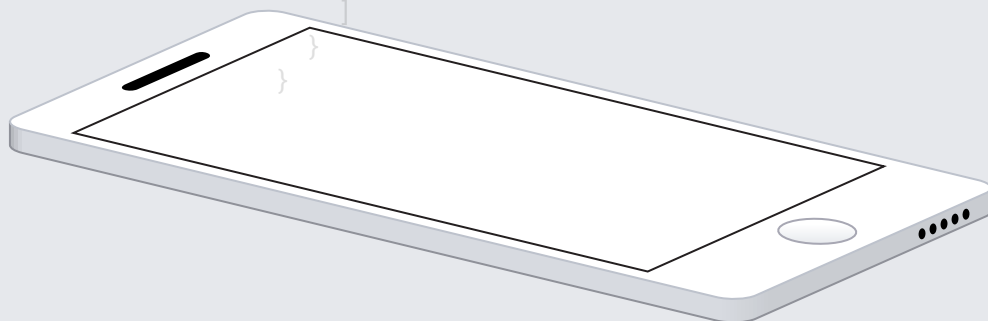
获取多个资源 只用一个请求

GraphQL 查询不仅能够获得资源的属性，还能沿着资源间引用进一步查询。典型的 REST API 请求多个资源时得载入多个 URL，而 GraphQL 可以通过一次请求就获取你应用所需的所有数据。这样一来，即使是比较慢的移动网络连接下，使用 GraphQL 的应用也能表现得足够迅速。



```
{  
  "hero": {  
    "name": "Luke Skywalker",  
    "friends": [  
      { "name": "Obi-Wan Kenobi" },  
      { "name": "R2-D2" },  
      { "name": "Han Solo" },  
      { "name": "Leia Organa" }  
    ]  
  }  
}
```

🔍 搜索文档...



描述所有的可能 类型系统

GraphQL API 基于类型和字段的方式进行组织，而非入口端点。你可以通过一个单一入口端点得到你所有的数据能力。GraphQL 使用类型来保证应用只请求可能的数据，还提供了清晰的辅助性错误信息。应用可以使用类型，而避免编写手动解析代码。

```
{
  hero {
    name
    friends {
      name
      homeWorld {
        name
        climate
      }
      species {
        name
        lifespan
        origin {
          name
        }
      }
    }
  }
}
```

```
type Query {
  hero: Character
}

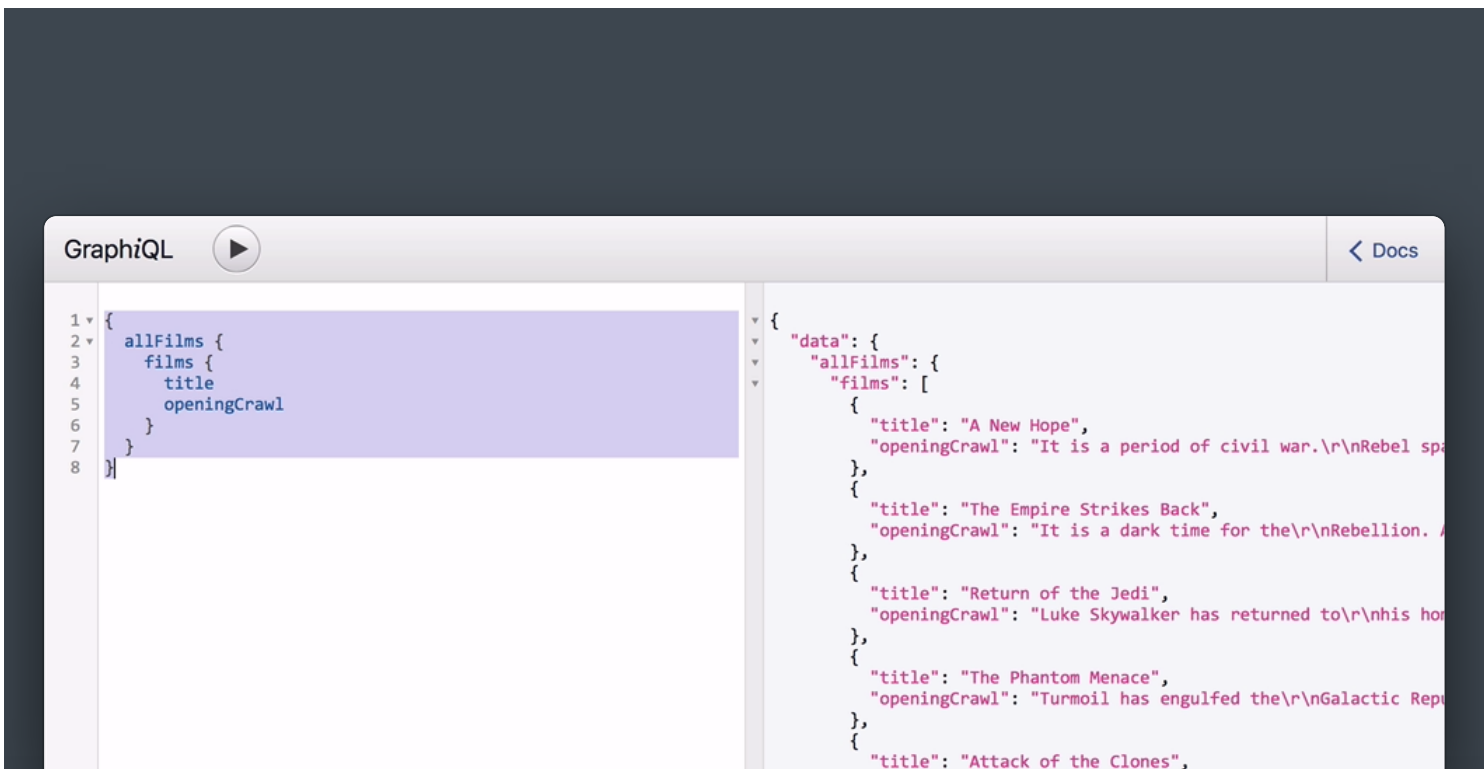
type Character {
  name: String
  friends: [Character]
  homeWorld: Planet
  species: Species
}

type Planet {
  name: String
  climate: String
}

type Species {
  name: String
  lifespan: Int
  origin: Planet
}
```

快步前进 强大的开发者工具

不用离开编辑器就能准确知道你可以从 API 中请求的数据，发送查询之前就能高亮潜在问题，高级代码智能提示。利用 API 的类型系统，GraphQL 让你可以更简单地构建如同 **GraphiQL** 的强大工具。



API 演进 无需划分版本

给你的 GraphQL API 添加字段和类型而无需影响现有查询。老旧的字段可以废弃，从工具中隐藏。通过使用单一演进版本，GraphQL API 使得应用始终能够使用新的特性，并鼓励使用更加简洁、更好维护的服务端代码。

```
type Film {
  title: String
  episode: Int
  releaseDate: String
}
```

```
type Film {
  title: String
  episode: Int
  releaseDate: String
+ openingCrawl: String
}
```

使用你现有的 数据和代码

GraphQL 让你的整个应用共享一套 API，而不用被限制于特定存储引擎。GraphQL 引擎已经有多种语言实现，通过 GraphQL API 能够更好利用你的现有数据和代码。你只需要为类型系统的字段编写函数，GraphQL 就能通过优化并发的方式来调用它们。

```
type Character {  
  name: String  
  homeWorld: Planet  
  friends: [Character]  
}
```

谁在使用 GraphQL?

Facebook 的移动应用从 2012 年就开始使用 GraphQL。GraphQL 规范于 2015 年开源，现已经在多种环境下可用，并被各种体量的团队所使用。



intuit.



更多 GraphQL 使用者



学习

入门

查询语言

类型系统

执行

最佳实践

代码

服务端

客户端

工具

社区

即将到来的活动

Stack Overflow

Facebook Group

Twitter

更多

GraphQL 团队博客

阅读规范

GitHub

编辑本页



Facebook
Open Source

Copyright ©2018 Facebook Inc. The contents of this page are licensed BSD-3-Clause.