AESTHETIC RECORD

Why GraphQL Over RESTful Design

GraphQL (Graph Query Language) is the succeeding methodology at the forefront of contemporary greenfield application designs. The current standard, REST (Representational State Transfer), while certainly an improvement over the preceding standard of SOAP (Simple Object Access Protocol), fails to address the modern technological issues of overfetching, underfetching and the n+1 problem. GraphQL approaches provide rapid iteration during the user experience, and robust server-side analytics.

Overfetching: Defined as downloading superfluous data due to the fixed data structure nature of REST interactions.

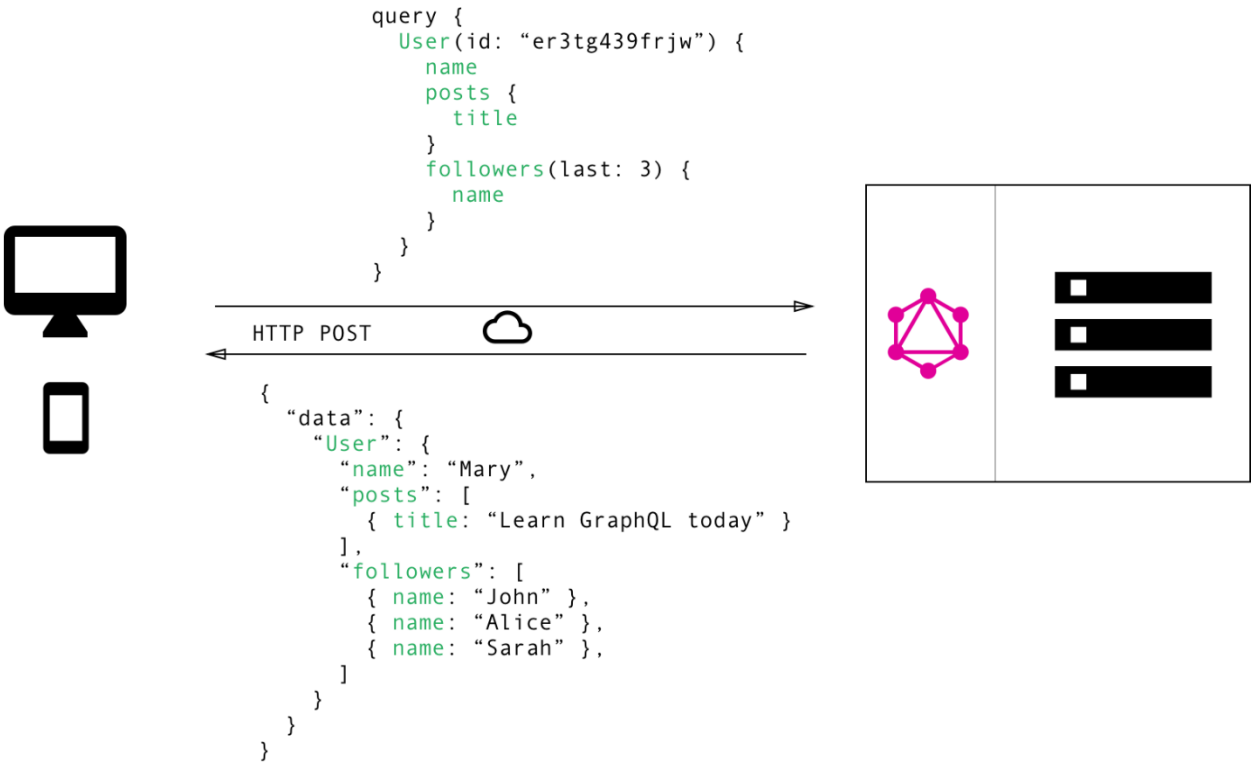
Underfetching and the N+1 Problem: API endpoint designs for applications in REST formats often only provide partial pieces of all the required information, prompting succeeding fetch requests to be necessary to obtain all required information from the data store sources.

Underfetching and the N+1 Problem Example

In the succeeding visual example, you may observe how in traditional RESTful designs multiple fetch queries are required to return the desired data set of a User's followers.

As the visual reflects, the attempt to obtain a single set of information results in 3 separate HTTP GET method requests to 3 separate API endpoints to furnish the desired information. This is not poor design; it is simply the nature of REST based applications.

As observed in the succeeding visual, GraphQL remedies the fetch issue with a dynamic request formation beforehand and then passing only a single request.



Summary

GraphQL by its nature will resolve most performance issues by intelligent, proper application design premised on healthy, efficient scalability. Hosting costs will be reduced, and application response times will be improved. GraphQL innately provides type and schema systems that can be merged seamlessly with either SQL or NoSQL data sources; while there is a learning curve to GraphQL, as this technology is relatively newer and less mature, it is currently the most performant approach in the space. GraphQL benchmarks outperform REST and SOAP driven applications.