

## GraphQL — a data language and new paradigm for APIs

Conférence au GenevaJUG le 29.10.2019, à l'invitation de Maxime.



### Abstract

GraphQL is a schema description and data query language, as well as a specification and a runtime, which are redefining API design and client-server interaction. GraphQL provides an alternative to REST-based architectures, with the purpose of increasing developer productivity and minimizing amounts of data transferred.

Software developers generally want to work with reusable abstractions instead of single-purpose methods. With GraphQL, we define each piece of data once and define how it relates to other data in our system. When a consumer application fetches data from multiple sources, it no longer needs to worry about the complex business logic associated with data join operations.

GraphQL is not tied to any specific backend and leverages your existing code and data, allowing it to integrate heterogeneous systems and expose their services thru a unified API.

Also, GraphQL's expressive schema language has the potential to define an *enterprise-wide data schema* and embrace its evolving nature — incidently, establishing a *common language* between developers, product owners and business analysts, a treasure and huge productivity boost.

Originally developed internally by Facebook, its specification and a reference implementation were open-sourced in 2015. In a matter of a few years, GraphQL was widely adopted by the industry, as it concisely solves many problems at once. Its development and standardization is now hosted under the Linux Foundation. Airbnb, Apollo, Coursera, Elementl, Facebook, GitHub, Hasura, Neo4j, Prisma, Shopify and Twitter are working together to form this new, vendor-neutral foundation that will provide unified governance and stewardship for GraphQL.

This talk will provide a primer on GraphQL, sample service architecture, share best practices and a reflexion on early experiences, its risks & opportunities — namely how to reach its full potential, which requires to adopt early on a *schema-driven* design approach of the business domain, instead of a *data-first* approach, a delicate distinction of the design process we'll discuss.

### Bio

Olivier Lange is the founder and managing-director of *Le Petit Atelier de Génie logiciel*, a software consultancy helping teams to accelerate their development processes and evolve legacy systems.

We provide advisory, business analysis and do craft *ad hoc* software — we build full-fledged apps, mostly business intelligence or data lineage monitoring dashboards, fill functional gaps and move complex datasets between systems.

Over the course of 18 years, we realized 130 assignments, ranging from 3 days to 24 months, for small and medium businesses, private and public institutions in Switzerland, France and Luxembourg— such as Lombard Odier TBI & Asset Management, Radio Television Suisse-romande (RTS), Office fédéral des routes (OFROU), HEI, Pro Helvetia, University of Geneva.

Next to our assignments, we run weekly and yearly *code retreats*, to step aside and take time, to research, learn and practice advanced subjects and pioneering technologies. We're currently focusing on reactive data flows, graph databases and visualizations of complex linked-data structures and multi-layered graphs.

Join us at our *Gōng-fu I/O* weekly hack sessions — you'll find other like-minded « data-nauts! »

[Directions](#)[Gōng-fu I/O](#)[GraphQL](#)