

John Doe,

A function's **parameters** and **return value** live entirely inside one process:

- They are not externally addressable.
- They carry no transport semantics (methods, status codes, headers).
- They cease to exist when the stack frame unrolls.

Hence they are **not endpoints**.

HTTP endpoints *are* endpoints because:

- Each resource is bound to a stable URI accessible over the network.
- The contract is defined by the HTTP spec—verbs, status codes, headers, caching, content negotiation.
- Calls cross process, trust, and failure boundaries; client and server evolve independently.

A **GraphQL schema** is *not* a set of endpoints:

- It is a typed contract (object types, fields, queries, mutations, subscriptions).
- It is executed via a *single* network endpoint (commonly POST /graphql).
- Individual fields are resolved by the execution engine, not directly over the wire.

GraphQL summary

GraphQL is a strongly typed query language and runtime (Facebook 2012, OSS 2015). Clients send a hierarchical query to one HTTP or WebSocket endpoint; the server validates against the schema, runs field resolvers (often in parallel), and returns precisely the requested data. It supports introspection, real time subscriptions, and schema federation without versioned URLs.

Regards,

Bo Shang