BACKEND

NodeJS & API-Technologien

AGENDA

NodeJS

API-Technologien

- gRPC
- REST
- GraphQL
- Vergleich

Softwaredemo

NODEJS – ALLGEMEIN

https://tutorials-raspberrypi.de/raspberry-pi-nodejs-webserver-installieren-gpios-steuern/

Serverseitiges Ausführen von JavaScript

Google Chrome‘s JavaScript-Laufzeitumgebung V8

Skalierbare Netzwerkanwendungen

NPM

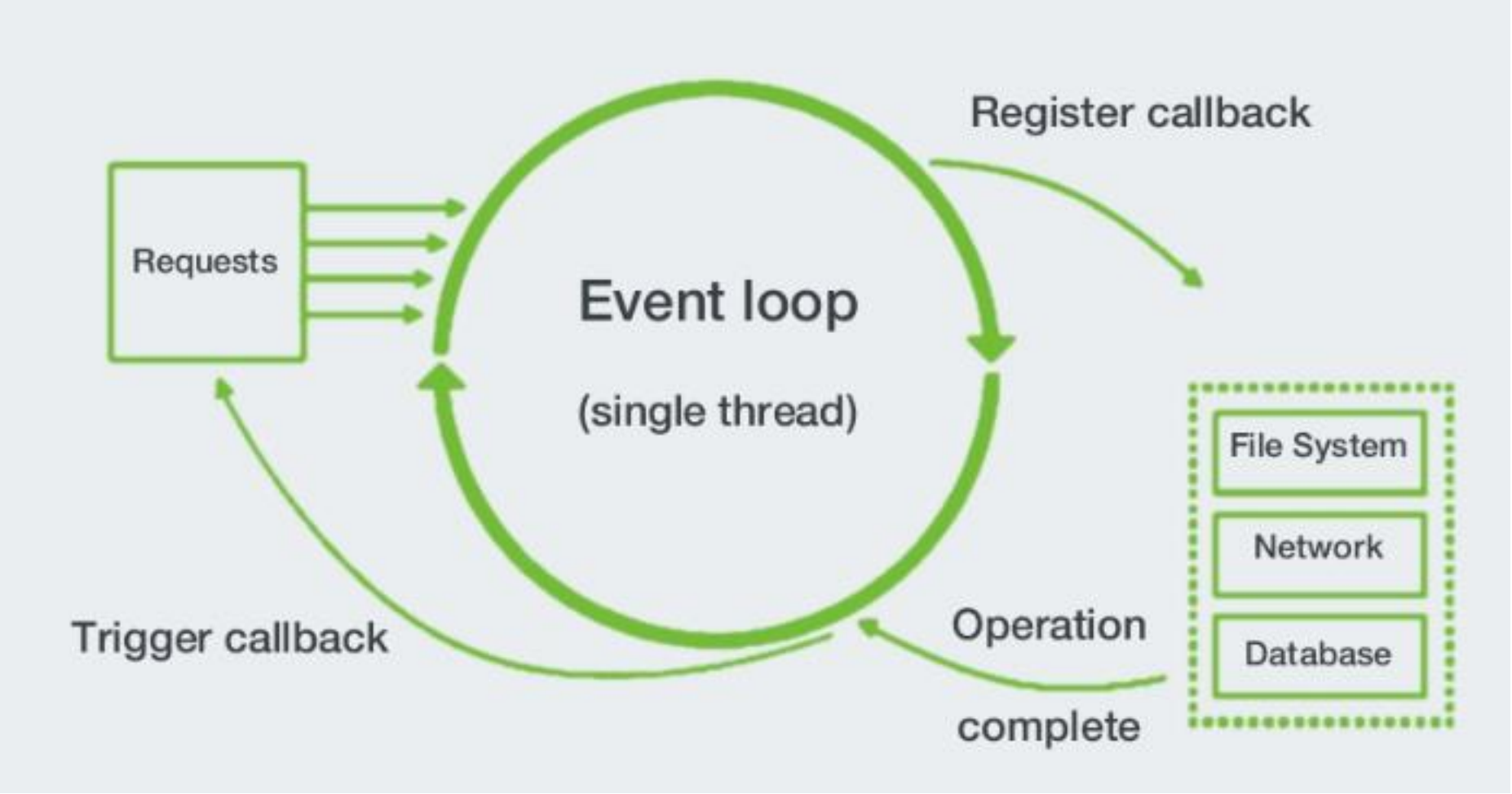
NODEJS - NPM

Weltgrößte Software-Register

Module installieren und veröffentlichen

Dependency-Management

NODEJS – EVENTLOOP



https://www.slideshare.net/elachechebedis/nodejs-essentials

NODEJS – JAVASCRIPT END-TO-END

V8 ermöglicht performante serverseitige Skriptsprache

Bekannte Skriptsprache für Web-Entwickler

Kein Kontextwechsel

Code-Sharing

NODEJS – NACHTEILE

Experimentelles Multi-Threading

Kein Multi-Threading in LTS Version

API-TECHNOLOGIEN

gRPC

REST

GraphQL

GRPC - ALLGEMEIN

https://grpc.io/

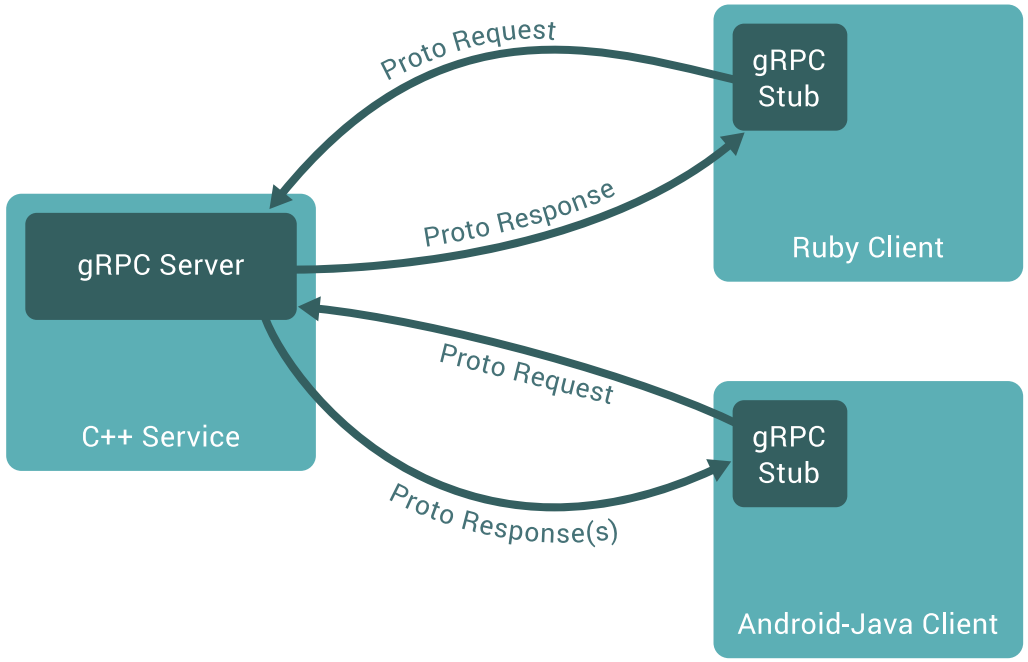
Modernes RPC Framework

Server-Methodenaufrufe vom Client als wäre es ein lokales Objekt

HTTP/2

Protocol Buffers

GRPC – METHODENAUFRUF



https://grpc.io/docs/guides/

GRPC - ALLGEMEIN

https://grpc.io/

Modernes RPC Framework

Server-Methodenaufrufe vom Client als wäre es ein lokales Objekt

HTTP/2

Protocol Buffers

GRPC – HTTP/2

ServerPush

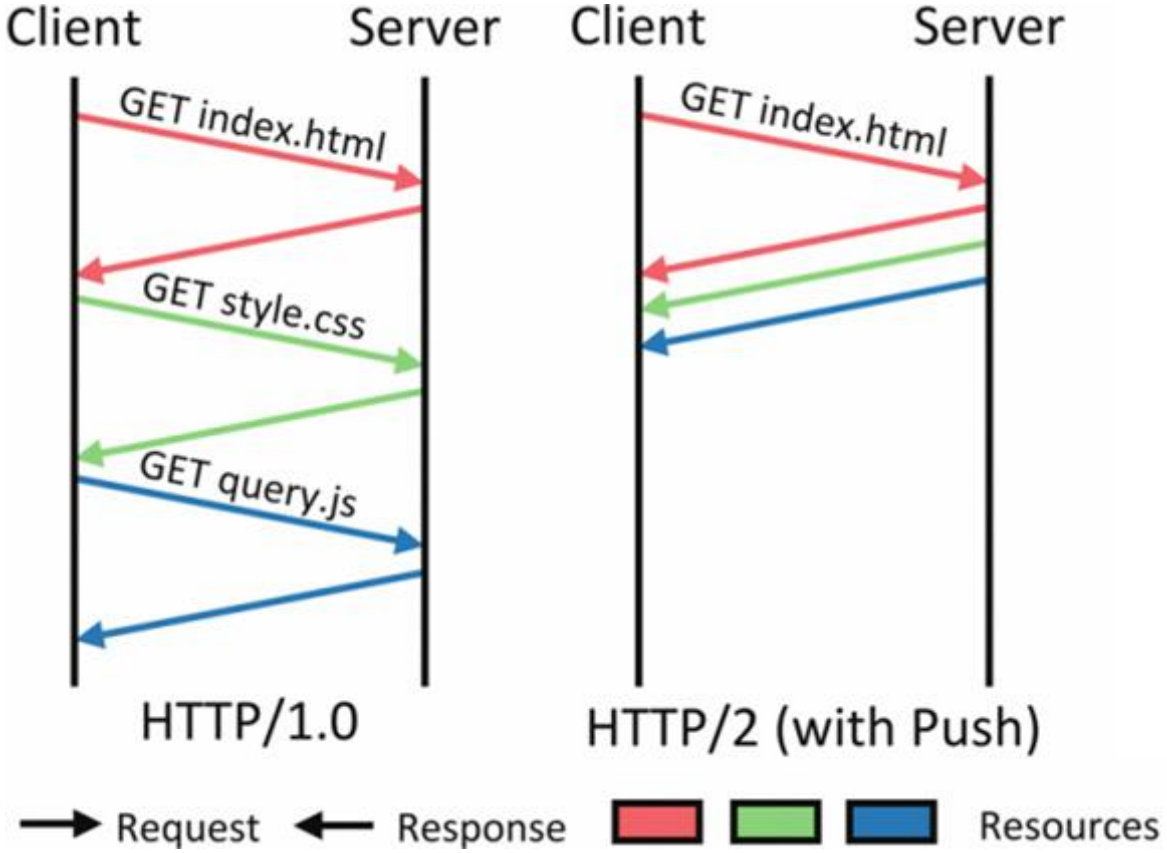
Kommunikation auf einem Kanal

Stream Priorisierung

Kompression des Headers

→ Höhere Geschwindigkeiten und weniger Overhead

GRPC - SERVERPUSH



Geändert von https://ieeexplore.ieee.org/document/8264830/

GRPC – HTTP/2

ServerPush

Kommunikation auf einem Kanal

Stream Priorisierung

Kompression des Headers

→ Höhere Geschwindigkeiten und weniger Overhead

GRPC – PROTOCOL BUFFERS

Serialisierung strukturierter Daten

Ähnlich zu JSON/XML

Generiert Klassen mit Getter-/Setter-Methoden

Umwandlung in Binärformat zur Übertragung

→ Kleinere und schnellere Datenübertragung

```
message Person {
  string name = 1;
  int32 id = 2;
  bool is_admin = 3;
}
```

GRPC – VORTEILE

Performance

Geringe Datengröße

GRPC – NACHTEILE

Browserinkompatibilität

- Aktuelle gRPC-Bibliotheken durch technische Limitierungen beschränkt
- Proxydienst

REST - ALLGEMEIN

Bewährte API-Technologie

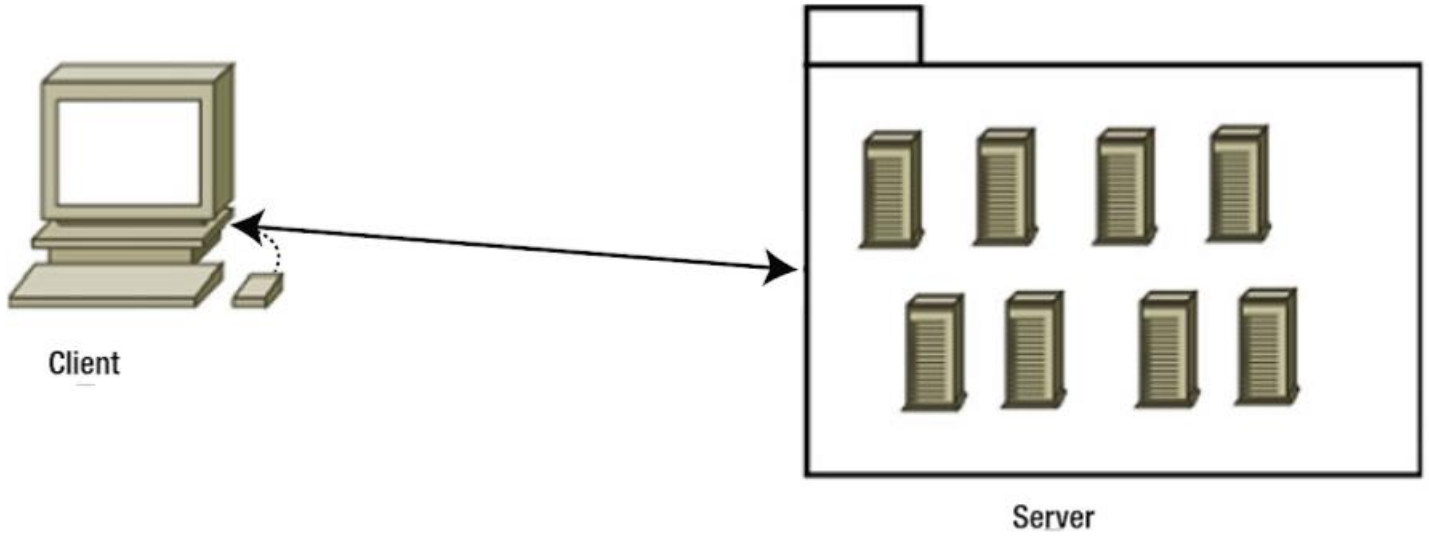
Seit 2000 von Roy Fielding

Kommunikation über HTTP-Anfragen

- GET
- POST
- PUT
- DELETE

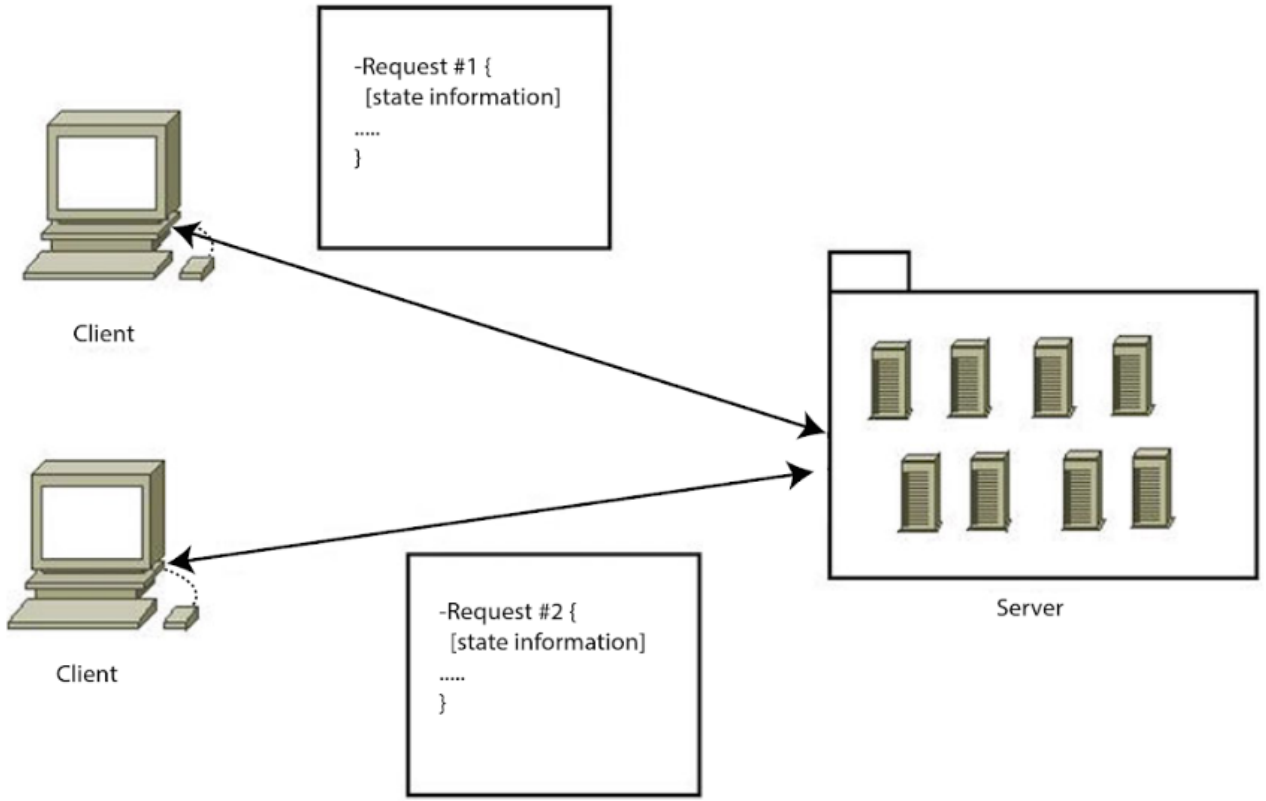
Sechs Prinzipien

REST – CLIENT – SERVER 1/6



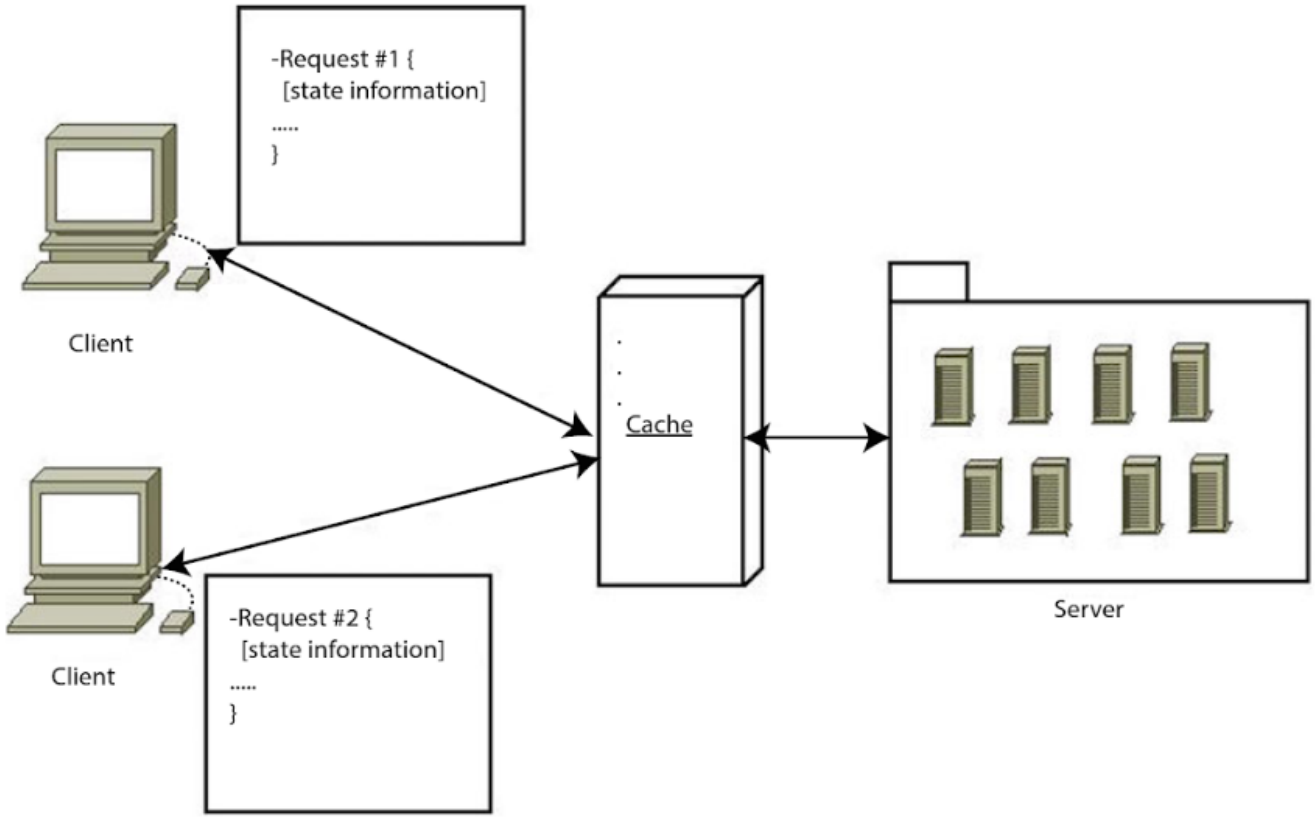
Doglio, Fernando: Pro REST API Development with Node.js

REST – STATELESS 2/6



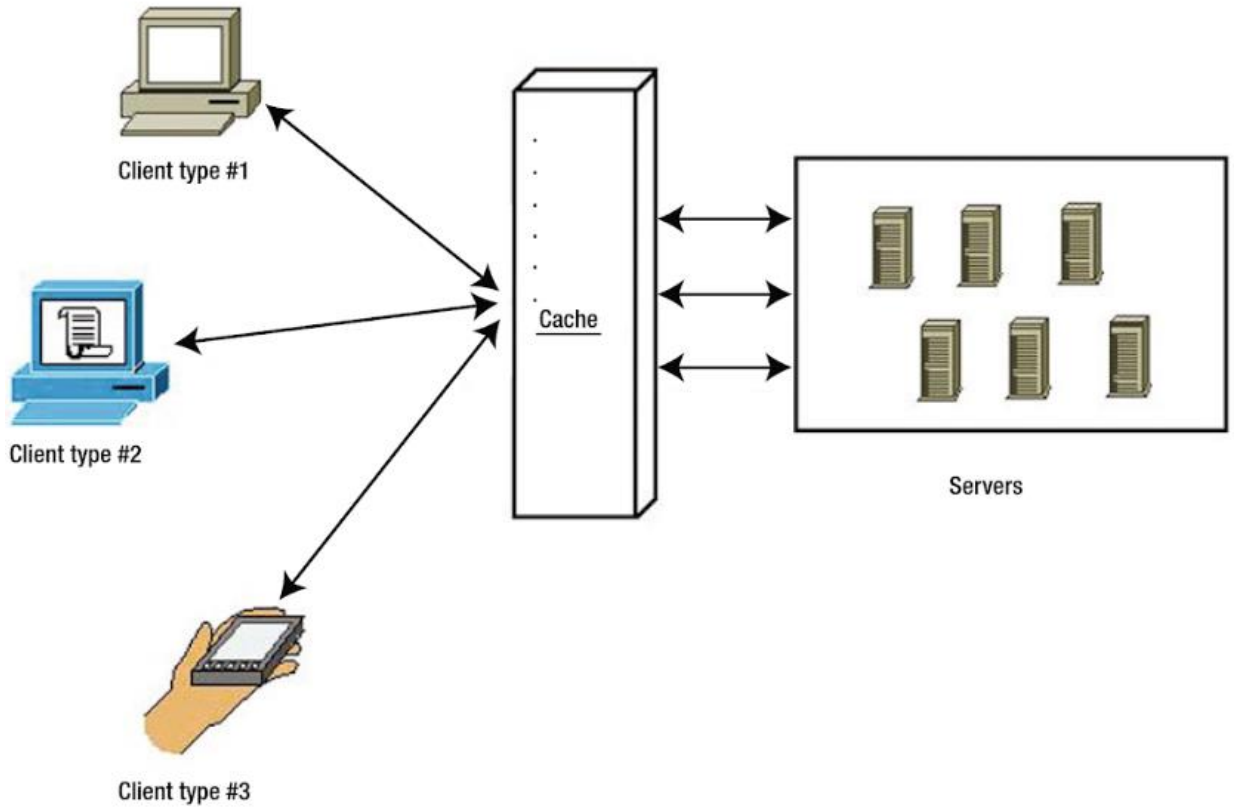
Doglio, Fernando: Pro REST API Development with Node.js

REST – CACHE 3/6



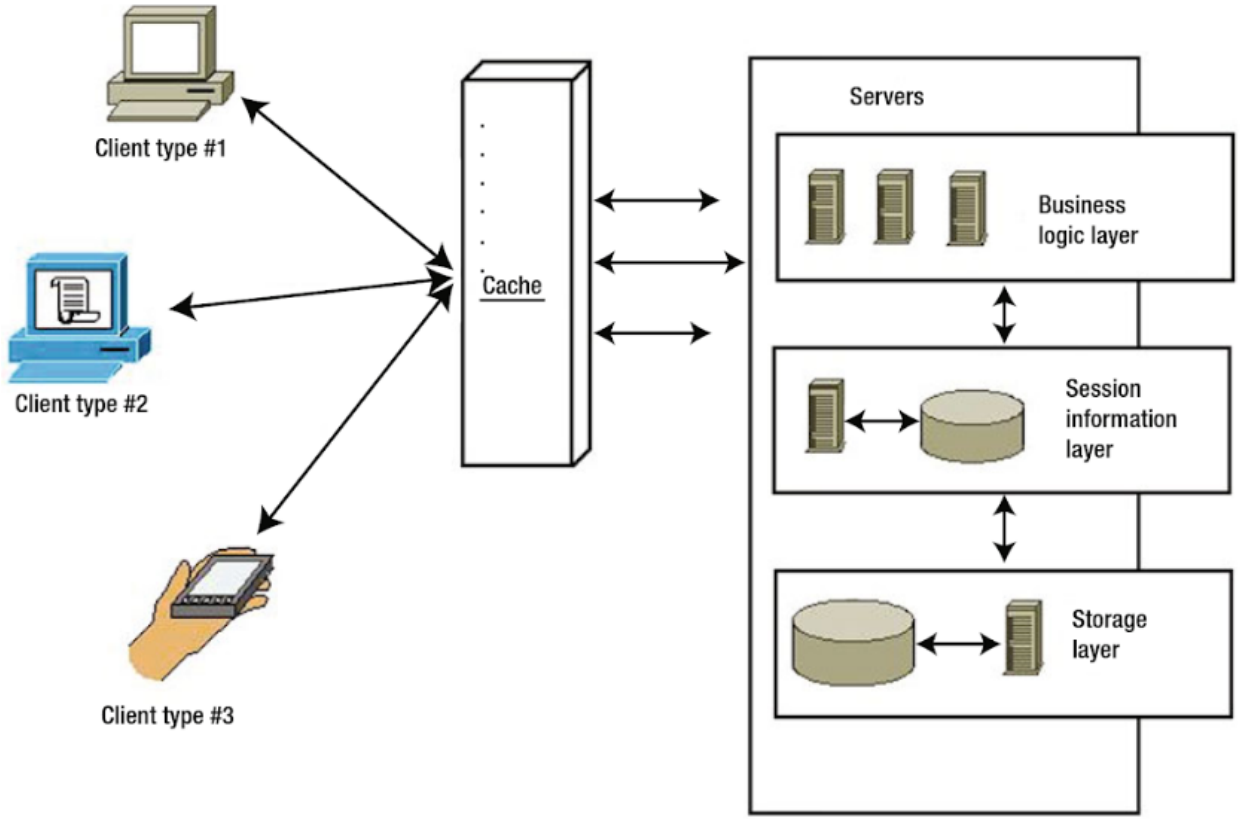
Doglio, Fernando: Pro REST API Development with Node.js

REST – UNIFORM-INTERFACE 4/6



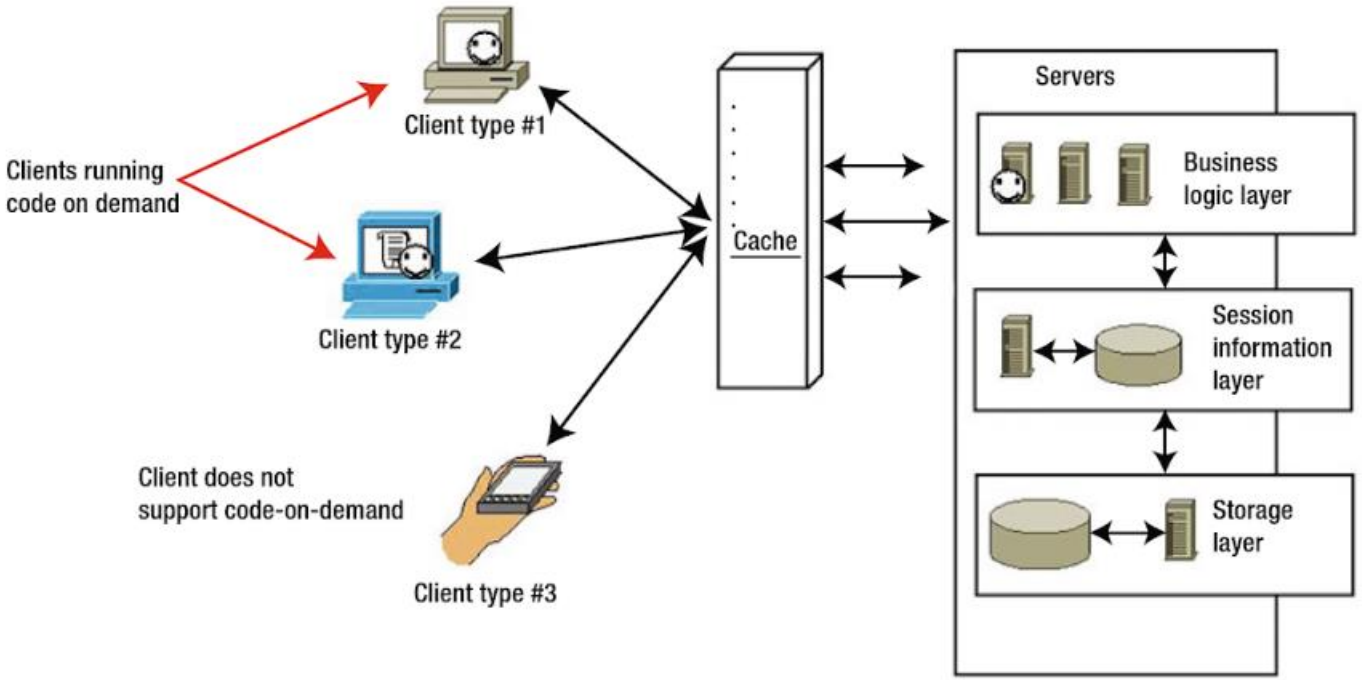
Doglio, Fernando: Pro REST API Development with Node.js

REST – LAYERED SYSTEM 5/6



Doglio, Fernando: Pro REST API Development with Node.js

REST – CODE-ON-DEMAND 6/6



Doglio, Fernando: Pro REST API Development with Node.js

REST – VORTEILE

Ausgereifte Lösung

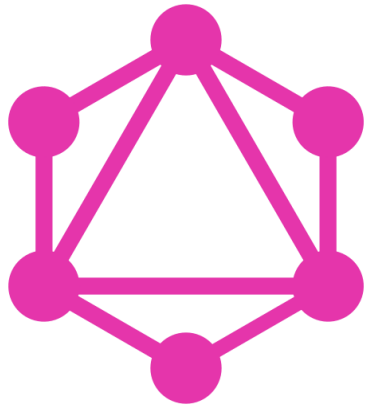
Leitfaden durch HATEOAS

REST – NACHTEILE

Fehlerhafte REST-Implementierungen

Overhead durch die Prinzipien

GRAPHQL – ALLGEMEIN



https://commons.wikimedia.org/wiki/File:GraphQL_Logo.svg

Von Facebook entwickelt

Query-Anfragen

Viele Client-Arten

GraphiQL

```
query getCard($cardID: String!) {
  card(id: $cardID) {
    id
    name
    text
  }
}
```

GRAPHQL – VORTEILE

Optimale Datenrückgabe durch Queries

Kein Over- / Underfetching

Viele Client-Arten auf einmal bedient

GRAPHQL – NACHTEILE

Standardmäßig kein Caching

- Nur ermöglicht durch Client-Bibliotheken

Kein Datei-Upload

- Benötigt zusätzliche Bibliothek oder REST-API

VERGLEICH

gRPC	REST	GraphQL
+ Schnelle Datenübertragung	+ Bewährte Technologie	+ Effiziente Rückgabedaten
+ Weniger Anfragen durch HTTP/2	+ Discoverability durch HATEOAS	+ Weniger Anfragen durch Query-Dynamik
- Browsersupport	- Overhead durch Prinzipien	+ Dynamisch für viele verschiedene Clientarten
	- Große Payloads oder viele spezielle Methoden	- Kein Fileupload
	- Viele Aufrufe	- Kein Caching

Services mit hohem Datendurchsatz

Komplexe Methoden und Abläufe

Komplexe Datenstrukturen oder viele Client-Arten

NODEJS & API-TECHNOLOGIEN

Fragen?