



# REST API

mit Node.js und ExpressJS

# REST – Representation State Transfer

- Programmierstil insbesondere für Webservices
- Maschine-zu-Maschine-Kommunikation
- Soll Zustandsübergang durch Transfer von Daten verbildlichen
- Meist über http&https

# REST Prinzipien

- Client-Server
- Zustandslos (die Nachricht)
- Nutzung von Caching
- Einheitliche Schnittstelle
- Mehrschichtige Systeme

# Befehle

- GET: Fordert Ressource vom Server an
- POST: Erstellt neue Ressource
- PUT: Aktualisiert eine Ressource
- DELETE: Entfernt eine Ressource

-> CRUD

# Node.js

- Asynchrone JavaScript runtime
- Event-basiert
- Schläft, solange keine Arbeit ansteht

# ExpressJS

- http Bibliothek für Node.js
- Requesthandler für http-Abfragen



Praxis