



Hochschule  
Albstadt-Sigmaringen

Albstadt-Sigmaringen University

# RESTfull APIs



Dipl. Ing. Sven Eppler (FH)  
sodge IT GmbH

A white speech bubble with a dark blue background. The bubble has a rectangular body and a triangular tail pointing downwards and to the left. The text "The web beyond HTML" is centered within the rectangular body in a bold, dark blue font.

**The web beyond HTML**

# RESTfull APIs

- Representational State Transfer (REST)
- Programmierkonzept zur Strukturierung von APIs
- 1994 von Roy Fielding entworfen
- Baut auf Konzepten von HTTP auf
- Verwendung bei
  - WebServices (z.B. Wetterdaten, Cryptobörsen)
  - Machine2Machine Kommunikation (z.B. IoT, Microservices)

# RESTfull APIs

- REST liefert typischerweise kein HTML
- Stattdessen strukturierte Daten
  - JSON, XML, Markdown, etc.
- Sofern notwendig, werden die Daten dann vom Client grafisch aufgearbeitet
  - Z.B. in HTML-Strukturen übersetzt
  - Hohe Wiederverwendbarkeit:  
Die selben Daten werden je nach Client anders angezeigt
- REST APIs sind stateless und lassen sich daher gut skalieren

# Semantik von HTTP Methoden in REST

- **GET**
  - Abrufen einer Resource (Seiteneffektfrei)
- **POST/PUT**
  - Anlegen einer neue Subresource
- **PATCH**
  - Einen Teil einer Resource ändern
- **DELETE**
  - Löschen einer Resource
- **HEAD**
  - Anfragen von Metadaten zu einer Resource
- **OPTIONS**
  - Anfragen welche Methoden für eine Resource erlaubt sind

# RESTfull URL Design

- RESTfull APIs benutzen URL Strukturen um die Angebotenen Ressourcen zu strukturieren
  - URL Namensräume
- GET Parameter werden vermieden, da für die selbe URL anhand der HTTP Methode unterschiedliche Aktionen ausgeführt werden können
- Für spätere Erweiterbarkeit werden REST URLs empfiehlt sich eine Versionierung

# RESTfull URL Design

- Beispiele

- GET /v1/users/
  - Listet alle User
- GET /v1/users/12/
  - Ruft User mit ID 12 ab
- DELETE /v1/users/12/
  - Löscht User mit ID 12
- POST /v1/users/
  - Fügt einen neuen User hinzu
- GET /v1/users/12/avatar/
  - Ruft Avatar für User mit ID 12 ab
- POST /v1/users/12/avatar/
  - Lädt Avatar für User mit ID 12 hoch

# Swagger / OpenAPI Specification

- OpenAPI Specification ist ein offener Standard zum beschreiben von RESTfull APIs
- Swagger ist eine Software zum Dokumentieren und Testen von RESTfull APIs
- Auf Basis einer `swagger.json` nach OpenAPI Spezifikation lassen sich z.B.
  - API-Clients generieren
  - UnitTests generieren
- Beispiel: <https://petstore.swagger.io/>



# RESTfull in Express

- Express bietet alle notwendigen Komponenten für RESTfull WebAPIs
- Beispiel:
  - `$GitHub-Repo/Beispiele/Node.js/GuestbookAPI`