

Generierung und Design einer Client-Bibliothek für einen RESTful Web Service am Beispiel der Spreadshirt-API

Bachelorverteidigung

Andreas Linz

HTWK - Fakultät für Informatik, Mathematik & Naturwissenschaften

15. Oktober 2013

Inhaltsverzeichnis

Einleitung

- Anforderungen
- Spreadshirt
- Spreadshirt-API

Hauptteil

- Web Services

- Dokumentbeschreibungssprachen
- Codegenerierung
- Datenmodelle & Codegenerator
- Client-Bibliothek

Zusammenfassung

- Ausblick
- Diskussion

Aufgabe

Was?

Client-Bibliothek aus abstrakter Beschreibung eines RESTful Web Service erzeugen.

Warum?

- ▶ Vereinheitlichung bestehender Implementierungen
- ▶ Nutzung der API für externe Entwickler erleichtern
- ▶ Authentifizierung kapseln

Anforderungen

- ▶ Austauschbarkeit der Zielsprache
- ▶ einfache Bedienbarkeit der Bibliothek
- ▶ gute Lesbarkeit des erzeugten Codes
- ▶ größtmögliche Typsicherheit des erzeugten Codes
- ▶ hohe Testabdeckung
- ▶ vollständige Generierung der Methoden aus der API-Beschreibung

Spreadshirt

- ▶ führendes Unternehmen für *personalisierte Bekleidung*
- ▶ *Social-Commerce*
- ▶ Standorte in Europa & Nordamerika, HQ in Leipzig
- ▶ ≈ 450 Mitarbeiter, 50 in der IT
- ▶ $4 * 10^5$ Spreadshirt-Shops mit $33 * 10^6$ Produkten

- ▶ Online-Plattform um Kleidungsstücke, Tassen, Sticker und mehr selbst zu:
 - ▶ gestalten
 - ▶ kaufen
 - ▶ eigene Designs als Motiv oder Produkt zum Verkauf anbieten

Spreadshirt-API

- ▶ API erlaubt Entwicklern die Nutzung eines großen Teils der Funktionen der Online-Plattform in eigenen Applikationen
- ▶ u.a. Produkt Erstellung, Design Upload & Warenkorbverwaltung
- ▶ Erstellen eigener Shops und kundenspezifischer Anwendungen

RESTful Web Service

REST

- ▶ *Representational State Transfer* (Gegenständlicher Zustandstransfer)
- ▶ Softwarearchitekturstil für Webanwendungen
- ▶ Anwendungen bestehen aus *Ressourcen* mit eindeutigem Bezeichner (Abbildung 1)
- ▶ Zustand einer Ressource ist eine *Repräsentation*
- ▶ Aktionen mit einer REST-API über den Austausch von Repräsentationen


The diagram shows the URL `http://api.spreadshirt.net/api/v1/baskets/84/item/42`. A bracket under the first part (`http://api.spreadshirt.net/api/v1/`) is labeled *Basis-URL*. Another bracket under the second part (`baskets/84/item/42`) is labeled *Ressource*. Above `baskets/84` is a bracket labeled *Warenkorb*, and above `item/42` is a bracket labeled *Artikel*.

Abbildung: Beispiel-URI, um den Artikel 42 aus dem Warenkorb 84 anzusprechen

RESTful Web Service ist eine Webanwendung die den REST Prinzipien entspricht

Dokumentbeschreibungssprachen

WADL

- ▶ maschinenlesbare Beschreibung einer HTTP-basierten Webanwendung
- ▶ XML Syntax
- ▶ Baumgrammatik

XSD

- ▶ Dokumentbeschreibungssprache zur Definition von Datentypen
- ▶ XML-Syntax, XSD ist selber ein gültiges XML-Dokument

Codegenerierung

Vorteile

- ▶ Produktivitätssteigerung
- ▶ hohe Konsistenz des Generats
- ▶ zentrale Stelle für Änderungen (Eingabemodell)

Generatorformen

Klassifikation nach Generierungsmenge

- ▶ teilweise
 - ▶ Inline-Code Expander
 - ▶ Mixed-Code Generator
 - ▶ Partial-Class Generator
- ▶ vollständig (Tier-Generator)
- ▶ mehrfach (n-Tier Generator)

Datenmodelle

Eingabe des Generators

- ▶ Applikationsmodell:
 - ▶ REST-Modell
 - ▶ Schema-Modell
- ▶ Sprachenmodell
 - ▶ kapselt Zielsprache
 - ▶ enthält Semantik
 - ▶ Syntax in Ausgabemodul (LanguageVisitor, siehe Abbildung 2)

Codegenerator

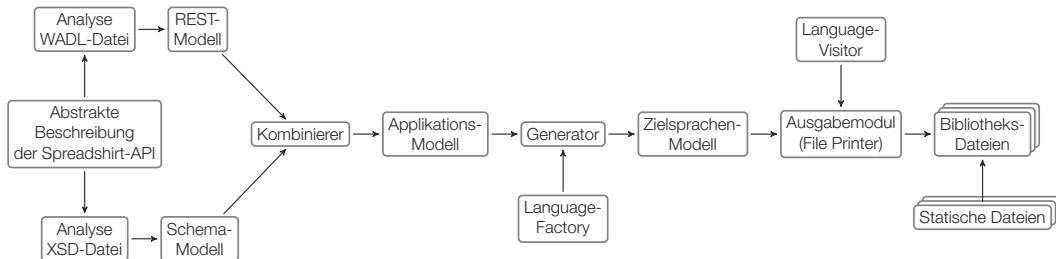


Abbildung: Sequenzdiagramm des Generators für die Spreadshirt-API

Datenklassen

- ▶ zielsprachenabhängige Repräsentation der Typen aus der XML-Schema Beschreibung
- ▶ Variablen bilden die Attribute und Elemente aus dem Schematyp ab
- ▶ Getter- und Setter-Methoden für alle Variablen
- ▶ Methoden zur Serialisierung und Deserialisierung in, bzw. aus, XML oder JSON

Ressourcenklassen

- ▶ zielsprachenabhängige Abbildung der Ressourcen- beschreibungen aus WADL-Datei
- ▶ Methoden der Klassen entsprechen den Methoden der abgebildeten Ressource

statische Klassen

- ▶ wurden manuell erstellt
- ▶ enthalten gemeinsam genutzten Code ohne variable Bestandteile
- ▶ Bibliothek enthält zwei dieser Klassen:
 - ▶ Kommunikation über HTTP-Methoden mit der API
 - ▶ Kapselung der Authentifizierung

Zusammenfassung

- ▶ Überführung der Beschreibung in Eingabedatenmodelle des Generators
- ▶ Entwicklung des Datenmodells
- ▶ Generierung der Bibliothek

Ausblick

- ▶ Parameterobjekte
- ▶ Fluent-Interface
- ▶ Java-Bibliothek (Sprachenmodell)
- ▶ Erzeugung von Dokumentation und Testdaten

Diskussion

- ▶ XSD, WADL
- ▶ RESTful Web Service
- ▶ Datenmodelle für Web Service Beschreibung und Programmiersprache
- ▶ (tier) Stufen-Generator
- ▶ PHP Client-Bibliothek