RESTful **A**PI **M**odeling **L**anguage

How to **design** your API with RAML

Ola Deibitsch| ola.deibitsch@callistaenterprise.se

CALLISTA Enterprise

Problem

SOAP got WSDL, but how do we **describe** a **REST API**?

REST/?

Simple
Lightweight
Structured

SOAP/WSDL

Complex
Constraints (XML 1.0 & UPA)
Structured

CALLISTA Enterprise

How do Users build API:s Today?

SOAP got WSDL, but how do we **describe** a **REST API**?

Swagger…

Apiary
Blueprint…

WADL…

Mashery I/O Docs…

Google Discovery
Docs…

JAX-RS
annotations…*

CALLISTA
Enterprise

RAML

RAML by the RAML Workgroup

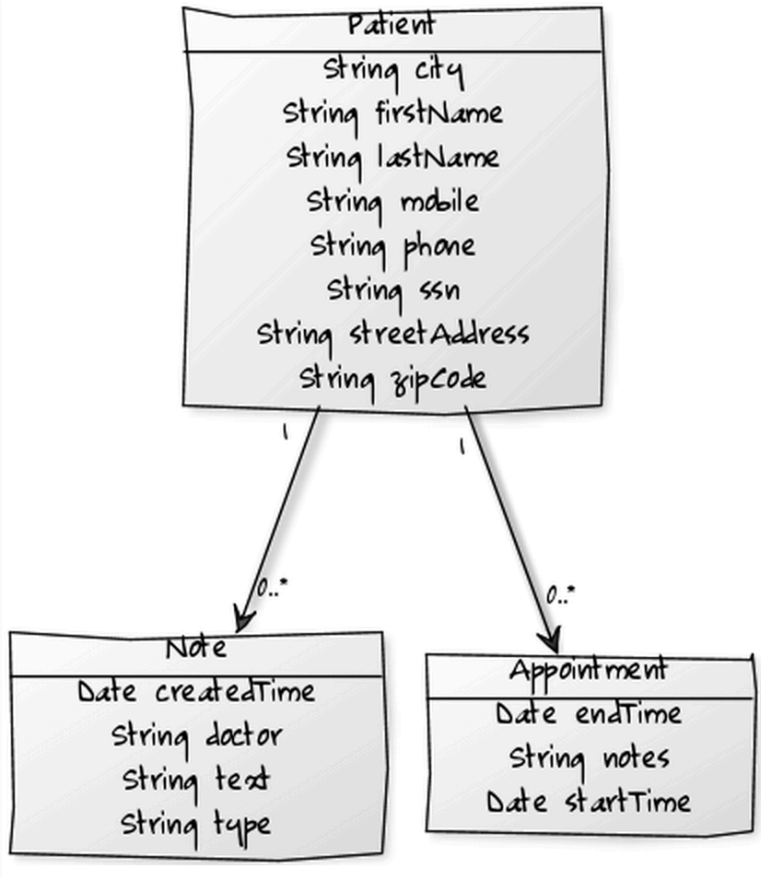
"*RESTful API Modeling Language (RAML) is a* ***simple*** *and* ***succinct*** *way of describing practically-RESTful APIs. It encourages* ***reuse****, enables discovery and* ***pattern-sharing****, and aims for merit-based emergence of* ***best practices****. The goal is to help our current API ecosystem by solving immediate problems and then encourage ever-better API patterns. RAML is built on broadly-used standards such as YAML and JSON and is a non-proprietary,* ***vendor-neutral*** *open spec.*", ***www.raml.org***

CALLISTA Enterprise

DEMO: Callista Care API

Requirements Callista Care API:

- Resources:
  - /patients
  - /patients/{patientId}
  - /patients/{patientId}/notes
  - /patients/{patientId}/appointments
  - …
- Verb
  - Get/Post/Put/Delete…
- Metadata
  - Authorization Header
  - Filtering
- Representations:
  - JSON, XML, ?.



CALLISTA Enterprise

RAML Tooling

RAML Introduction

- **API Designer**
  - *RAML Editor*
  - *RAML Console*
- API Notebook
- **APIkit**
- SoapUI RAML Plugin
- JAX-RS Codegen
- …

See more at www.raml.org

CALLISTA Enterprise

DEMO: Callista Care API (cont.)

1. **Design** using the **API Designer**.
2. **Implement** using **Mule Studio** and **APIkit plugin**.
3. **Test** using **APIkit's** embedded **RAML Console** or directly in **API Designer**

CALLISTA Enterprise

RAML Tooling: RAML -> JAX-RS

JAX-RS Codegen

API Designer

callistacare_v1.raml

```
#%RAML 0.8
title: Callista Care API
version: v1
/patients:
  displayName: Patients
  description: A collection of pa
  get:
    description: Get a collectio
    responses:
      200:
        description: OK
        body:
          application/json:
            example: '{}'
          application/xml:

  post:
    description: Create a patien
    body:
      application/json:
    responses:
```

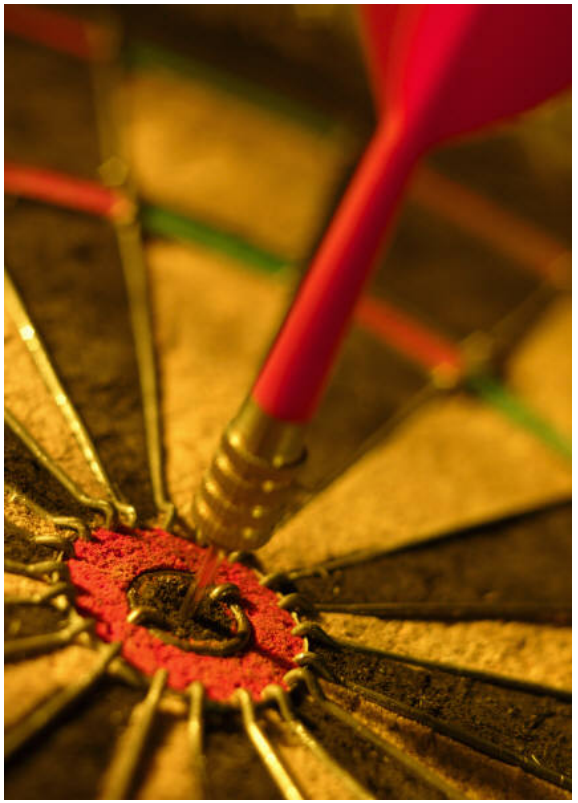
Patients.java

```
@Path("patients")
public interface Patients {


    /**
     * Get a collection of patients.
     *
     */
    @GET
    @Produces({
        "application/xml",
        "application/json"
    })
    Patients.GetPatientsResponse getPatients();

    /**
     * Create a patient.
     *
     * @param entity
     *
     */
    @POST
    @Consumes("application/json")
    @Produces({
        "application/json"
    })
    Patients.PostPatientsResponse postPatients(Reader entity);

    /**
     *
     * @param patientId
     *     Social Security Number e.g. 191212121212
```

Summary

- ✓ *”Simple”, ”Succinct”, ”Reuse”, ”Pattern-Sharing”, ”Vendor-Neutral”…*
- ✓ **Consumption & Producer Friendly**
- ✓ **API Documentation**

➔ **TOO EASY NOT TO TRY!**



CALLISTA Enterprise

REST

Fundamentals

- Verbs: HTTP methods
- Metadata: HTTP headers
- Nouns: HTTP resources, desribed by URI paths.
- Responses: HTTP status codes
- Representations: media type in the body

CALLISTA Enterprise

API Explosion?