Loeng 8

Spring Boot

Spring Data JDBC

Rest Repositories

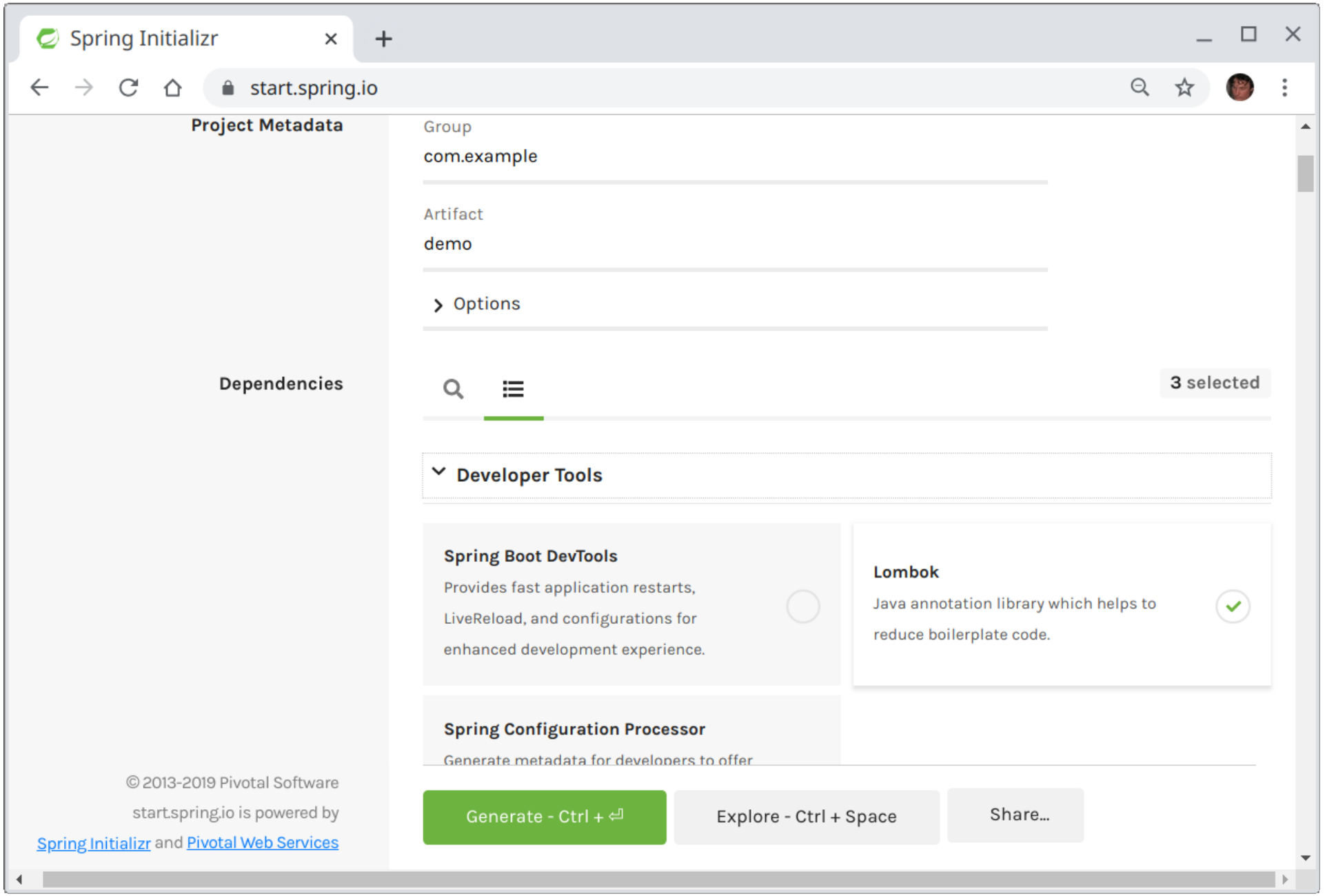
Rest

Swagger

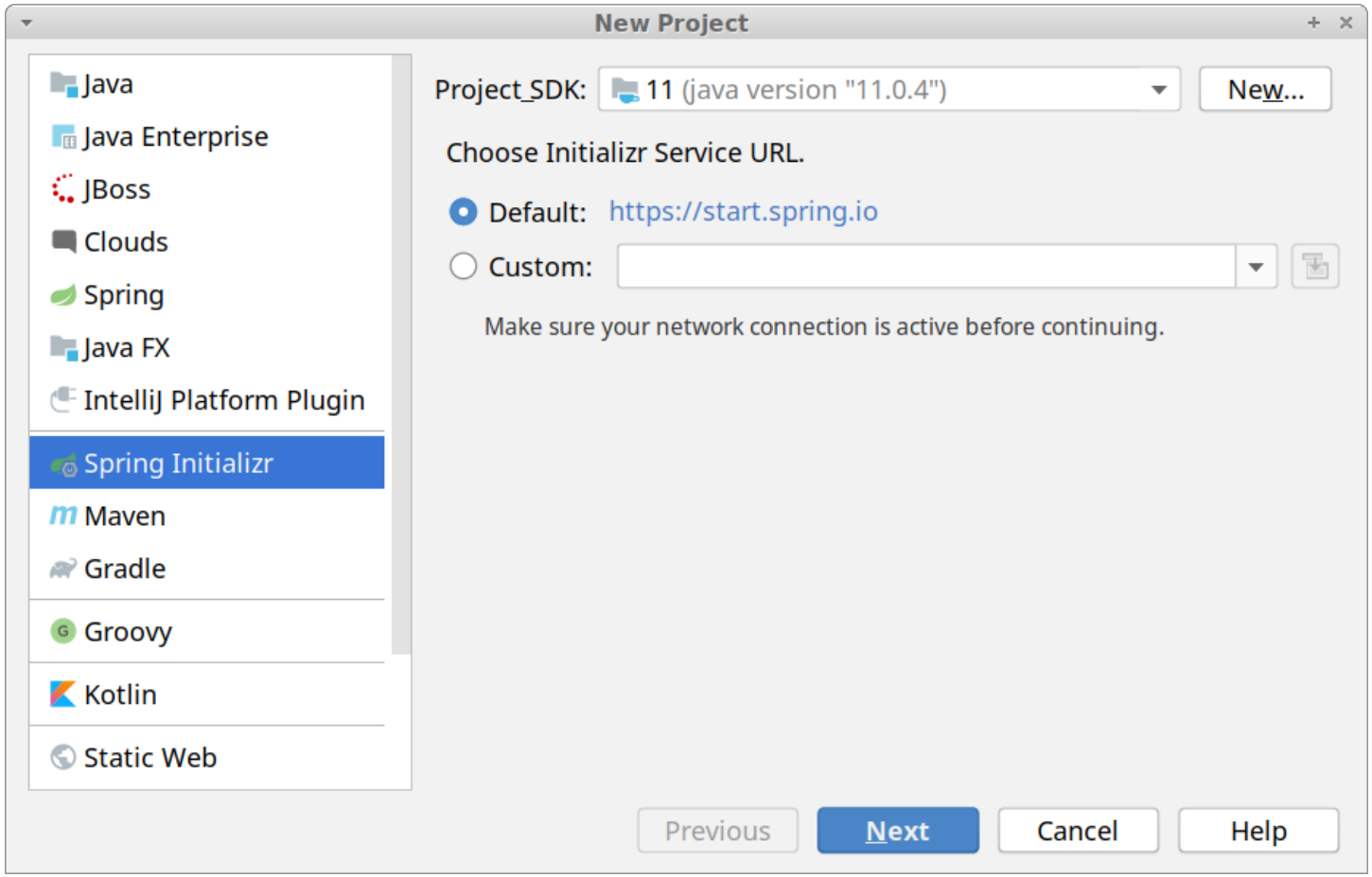
Spring Boot

- Seob kokku paljud Java raamistikud ja muud tööriistad
- Convention over Configuration

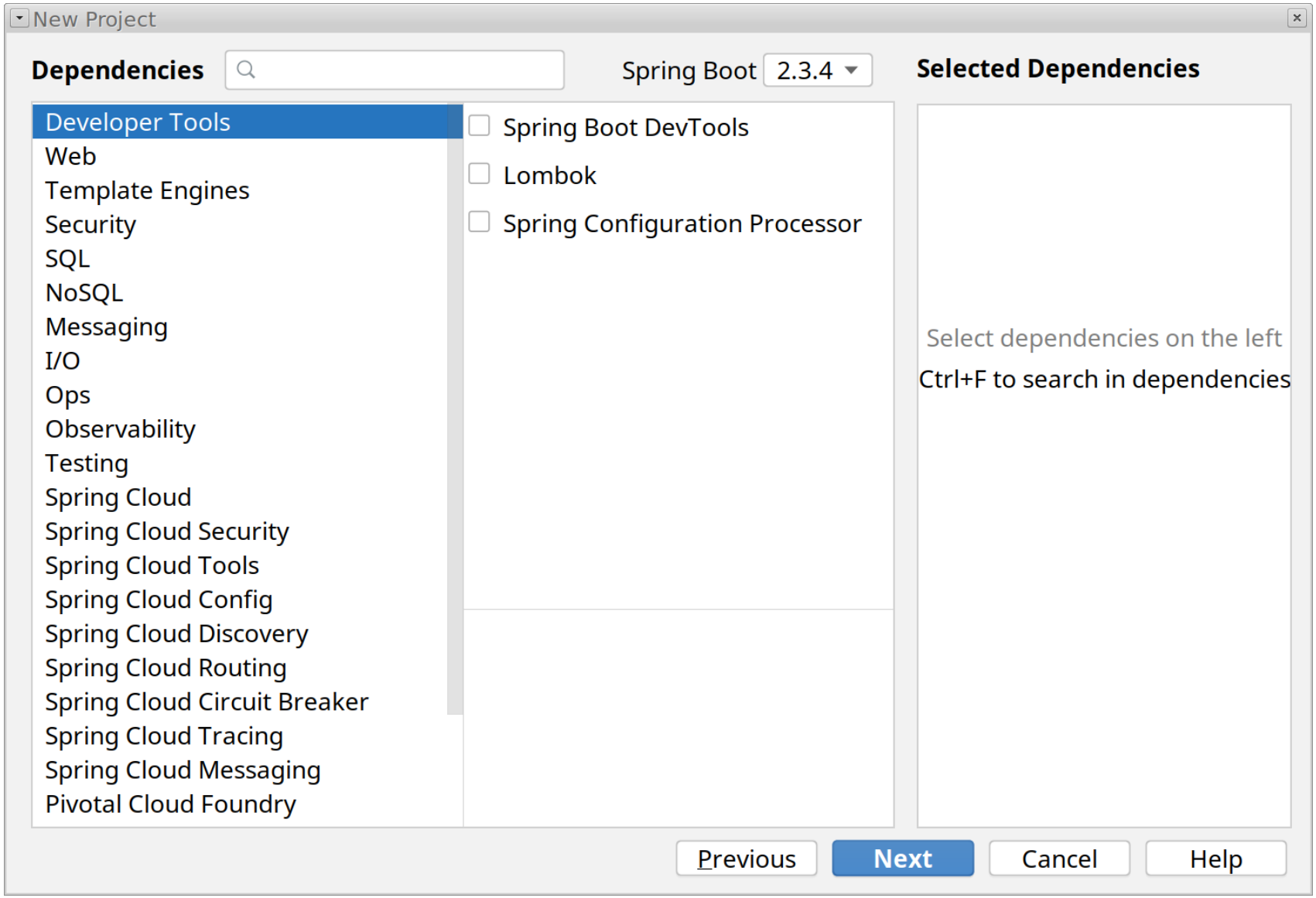
Spring Boot


Spring Initializr
start.spring.io
Project Metadata
Group
com.example
Artifact
demo
Options
Dependencies
3 selected
Developer Tools
Spring Boot DevTools
Provides fast application restarts, LiveReload, and configurations for enhanced development experience.
Lombok
Java annotation library which helps to reduce boilerplate code.
Spring Configuration Processor
Generate metadata for developers to offer
© 2013-2019 Pivotal Software
start.spring.io is powered by
Spring Initializr and Pivotal Web Services
Generate - Ctrl + ⏎
Explore - Ctrl + Space
Share...

Spring Boot


New Project
Java
Java Enterprise
JBoss
Clouds
Spring
Java FX
IntelliJ Platform Plugin
Spring Initializr
Maven
Gradle
Groovy
Kotlin
Static Web
Project SDK:
11 (java version "11.0.4")
New...
Choose Initializr Service URL.
Default: https://start.spring.io
Custom:
Make sure your network connection is active before continuing.
Previous
Next
Cancel
Help

Spring Boot


New Project
Dependencies
Spring Boot 2.3.4
Selected Dependencies
Developer Tools
Web
Template Engines
Security
SQL
NoSQL
Messaging
I/O
Ops
Observability
Testing
Spring Cloud
Spring Cloud Security
Spring Cloud Tools
Spring Cloud Config
Spring Cloud Discovery
Spring Cloud Routing
Spring Cloud Circuit Breaker
Spring Cloud Tracing
Spring Cloud Messaging
Pivotal Cloud Foundry
Spring Boot DevTools
Lombok
Spring Configuration Processor
Select dependencies on the left
Ctrl+F to search in dependencies
Previous
Next
Cancel
Help

Spring Boot

- Genereerimine vs automaatne konfiguratsioon

Spring Boot („Hello, world!”)

Spring Boot (konfigureerimine)

- application.properties
  (nt. spring.datasource.url)
- Vaikimisi kataloogid (staatilised ressursid,
  Sql skriptid, jne.)
- Spring-i konfiguratsiooni failid (nt. Spring Security)
- build.gradle
- ...

Kasutamine projektis

build.gradle

```
task appRun(type: JavaExec) {
    main = 'mypackage.MyMainClass'

    classpath = sourceSets.test.runtimeClasspath
}
```

Spring Data (demo)

Miks mitte Spring Boot?

- Kõike ei saa automaatselt
- Suur võit on vaid algaja jaoks
- Vähem koodi aga ka vähem kontrolli
- Suurem paki maht (palju ülearuseid teeke)

REST

Representational State Transfer

1) Arhitektuuriline lahendus
2) Hulk tavasid

REST

```
PUT /api/customers/1 HTTP/1.1
Host: localhost:8080
Content-Type: application/json

{ "firstName": "Mari", "lastName": "Kask", "code": "A123" }
```

REST arhitektuur

- Roy Fielding (doktoritöö aastast 2000, peatükid 5 ja 6)
- HTTP/1.0 -> HTTP/1.1
- Peateema on skaleeritavus

Roy Fielding (eesmärk)

- REST has been used to limit the scope of standardized HTTP extensions to those that fit within the architectural model, rather than allowing the applications that misuse HTTP to equally influence the standard.

REST

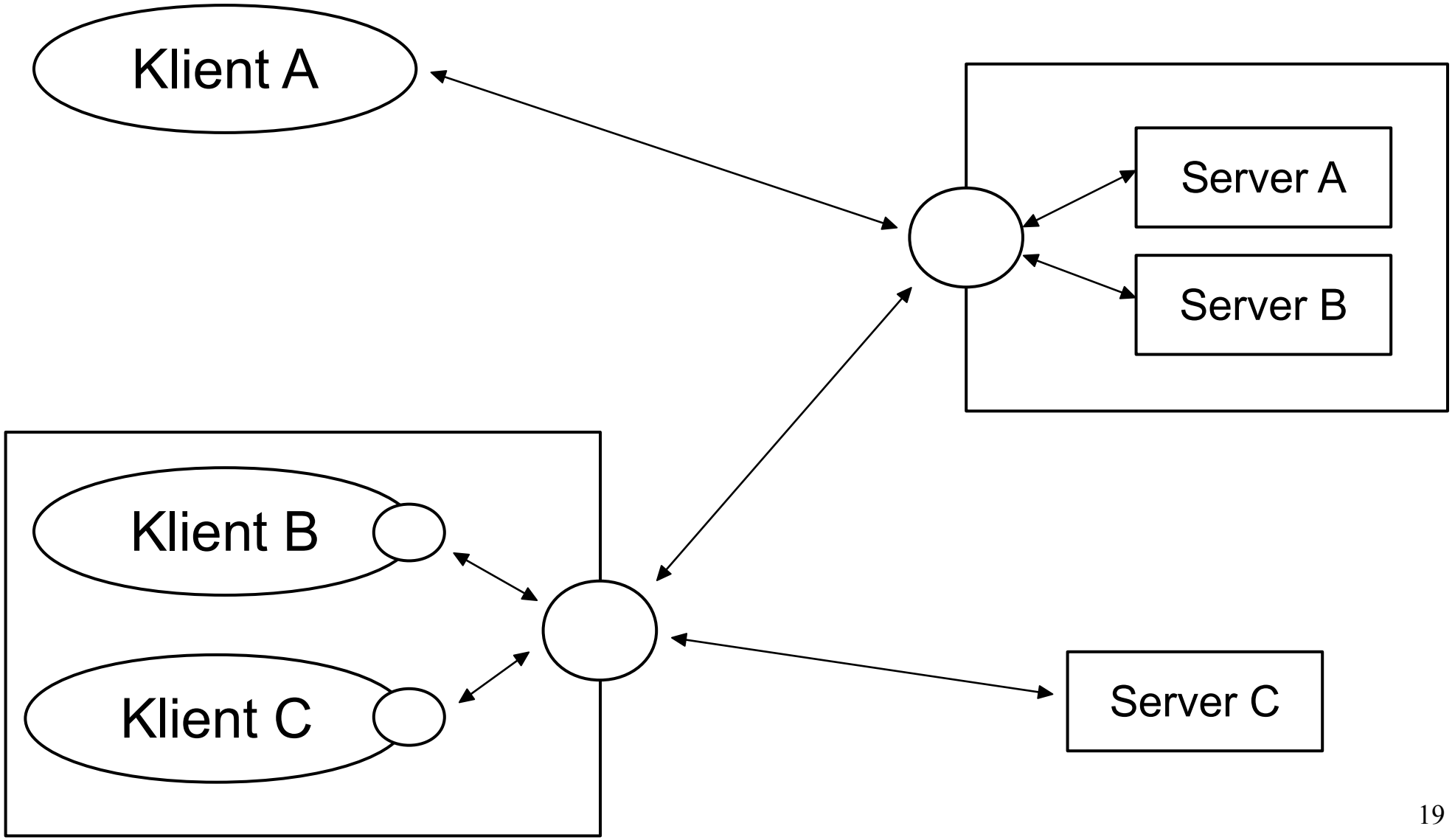
- Representational state transfer (REST) is a software architectural style that defines a set of constraints to be used for creating Web services

https://en.wikipedia.org/wiki/Representational_state_transfer

Piirangud

- Statelessness
- Cacheability
- Layered system
- Uniform interface
  - Resource identification in requests
  - Resource manipulation through representations
  - Self-descriptive messages
  - Hypermedia as the engine of application state (HATEOAS)

Cacheability, Layered system



Resource identification in requests

- /single-endpoint-service

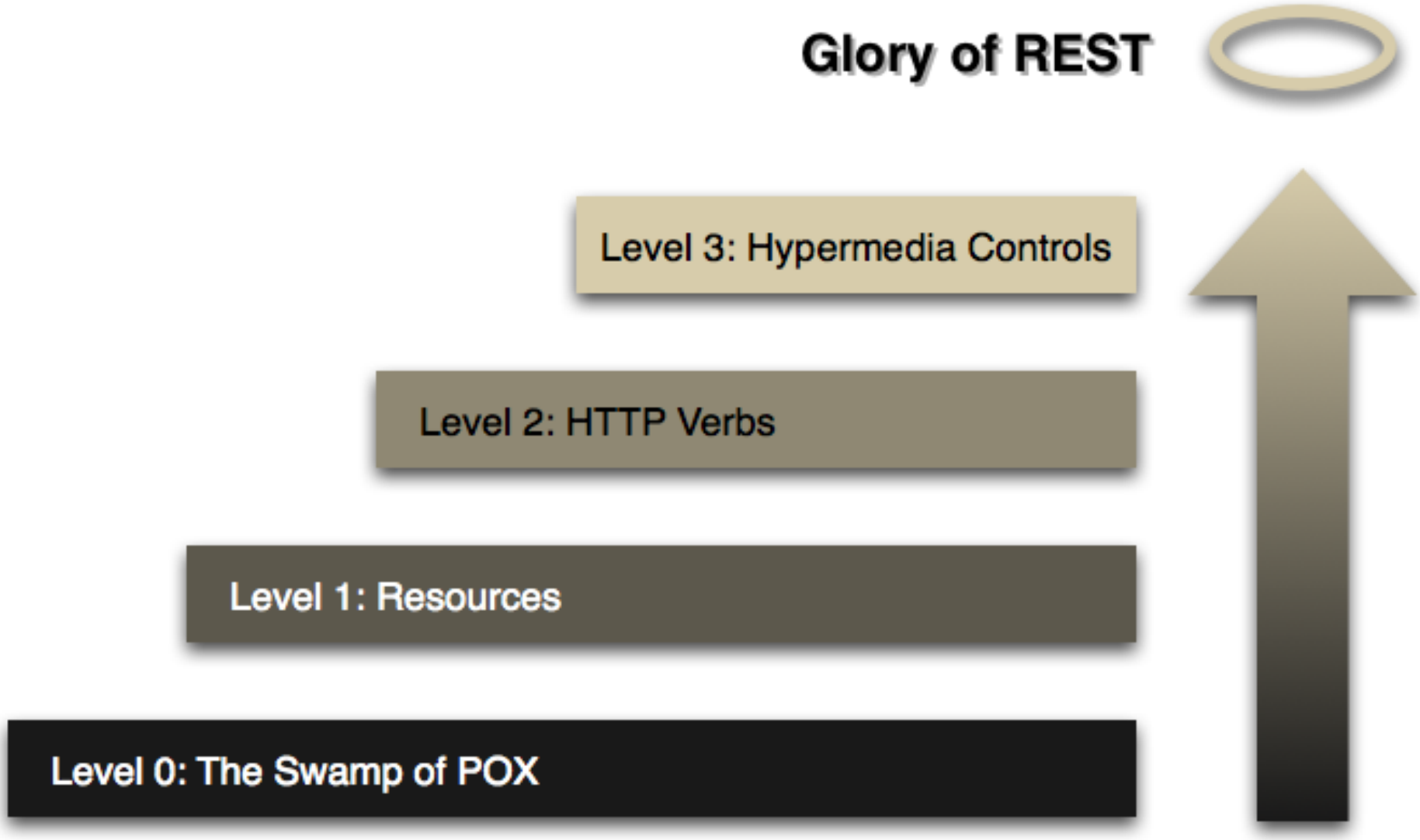
  vs

- /api/customers/1
- /api/customers?id=1
- /api/customers/search?startDate=...&endDate
- /api/customers/search?endDate=...&startDate

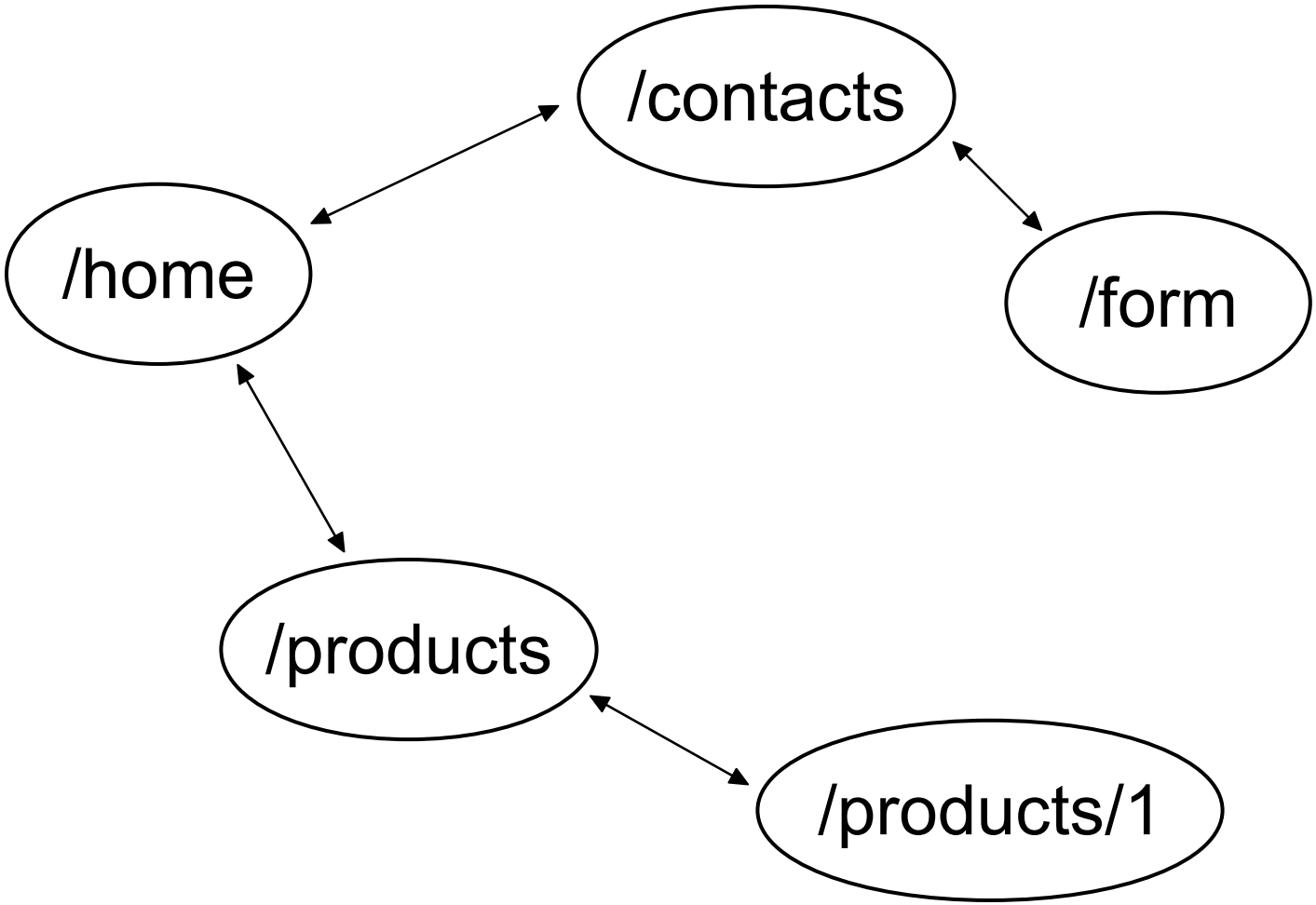
Self-descriptive messages

- Server ei hoia infot eelnevate sõnumite (päringute) kohta
- Iga sõnum (päring) sisaldab kogu infot päringu täitmiseks

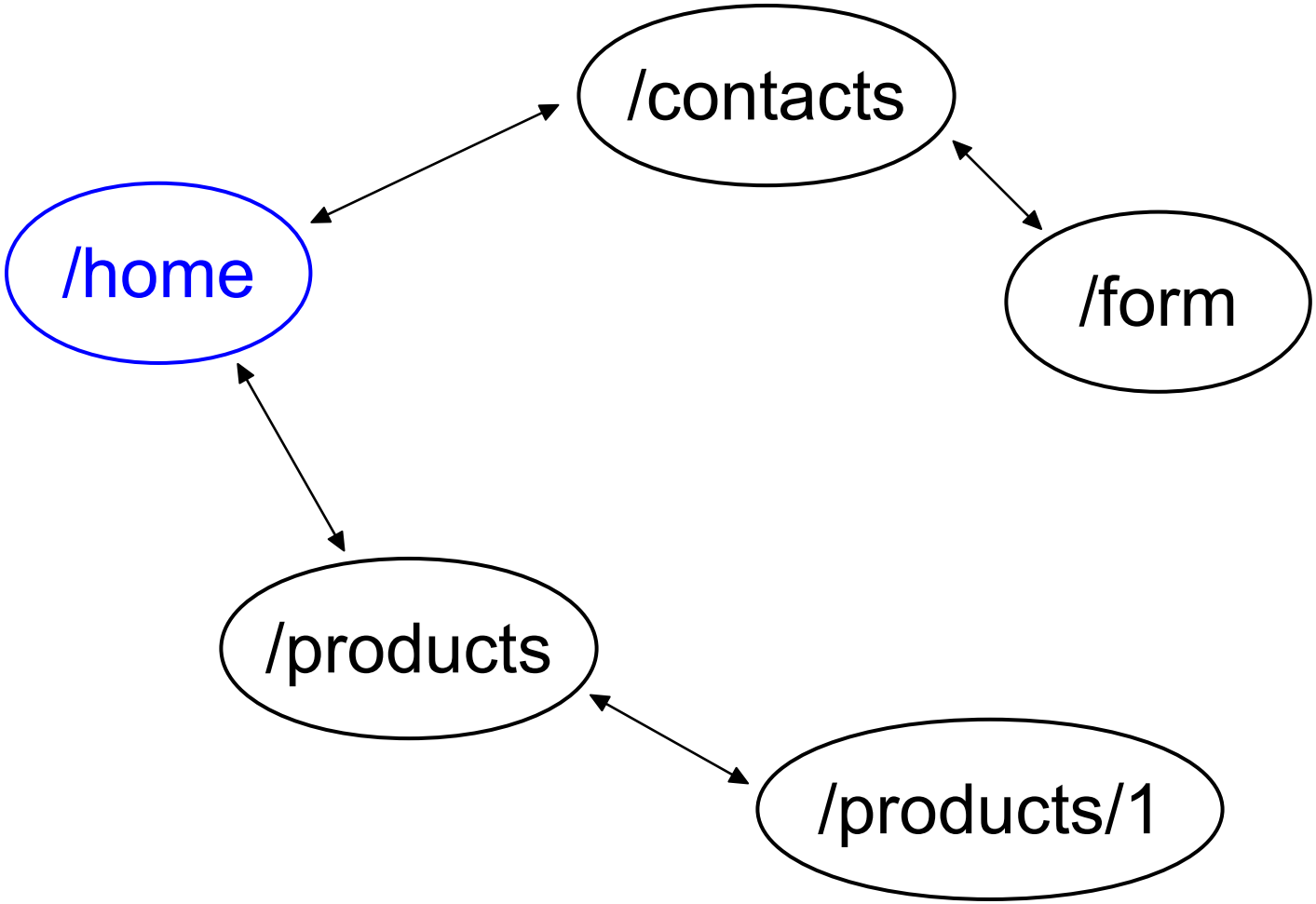
Richardson Maturity Model



Hypermedia (termin)



Hypermedia as the engine of application state



Rest Repositories (Hypermedia)

- http://localhost:8080/profile

Rest Repositories (demo)

Swagger (OpenAPI 3.0)

- Swagger is in essence an Interface Description Language for describing RESTful APIs expressed using JSON.

https://en.wikipedia.org/wiki/Swagger_(software)

Swagger (demo)

Webjars

- https://www.webjars.org/

```
implementation group: 'org.webjars',
               name: 'bootstrap',
               version: '4.5.3'
```

Tavad API loomisel

Verbid ja tagastuskoodid

- GET, POST, PUT, DELETE, PATCH
- 200, 201, 301, 400, 401, 403, 404, 500, ...

PUT

Enne

```
{
    "firstName": "Jill",
    "lastName": "Smith",
    ...
}
```

PUT

```
{
    "lastName": "Jones"
}
```

Pärast

```
{
    "lastName": "Jones"
}
```

PATCH

Enne

```
{
    "firstName": "Jill",
    "lastName": "Smith",
    ...
}
```

PATCH

```
{
    "lastName": "Jones"
}
```

Pärast

```
{
    "firstName": "Jill",
    "lastName": "Jones",
    ...
}
```

Json Patch

```
{
    "firstName": "Jill",
    "lastName": "Smith",
    ...
}
```

```
{ "op": "replace", "path": "/lastName", "value": "Jones" }
```

```
{
    "firstName": "Jill",
    "lastName": "Jones",
    ...
}
```

API loomisega alustamine

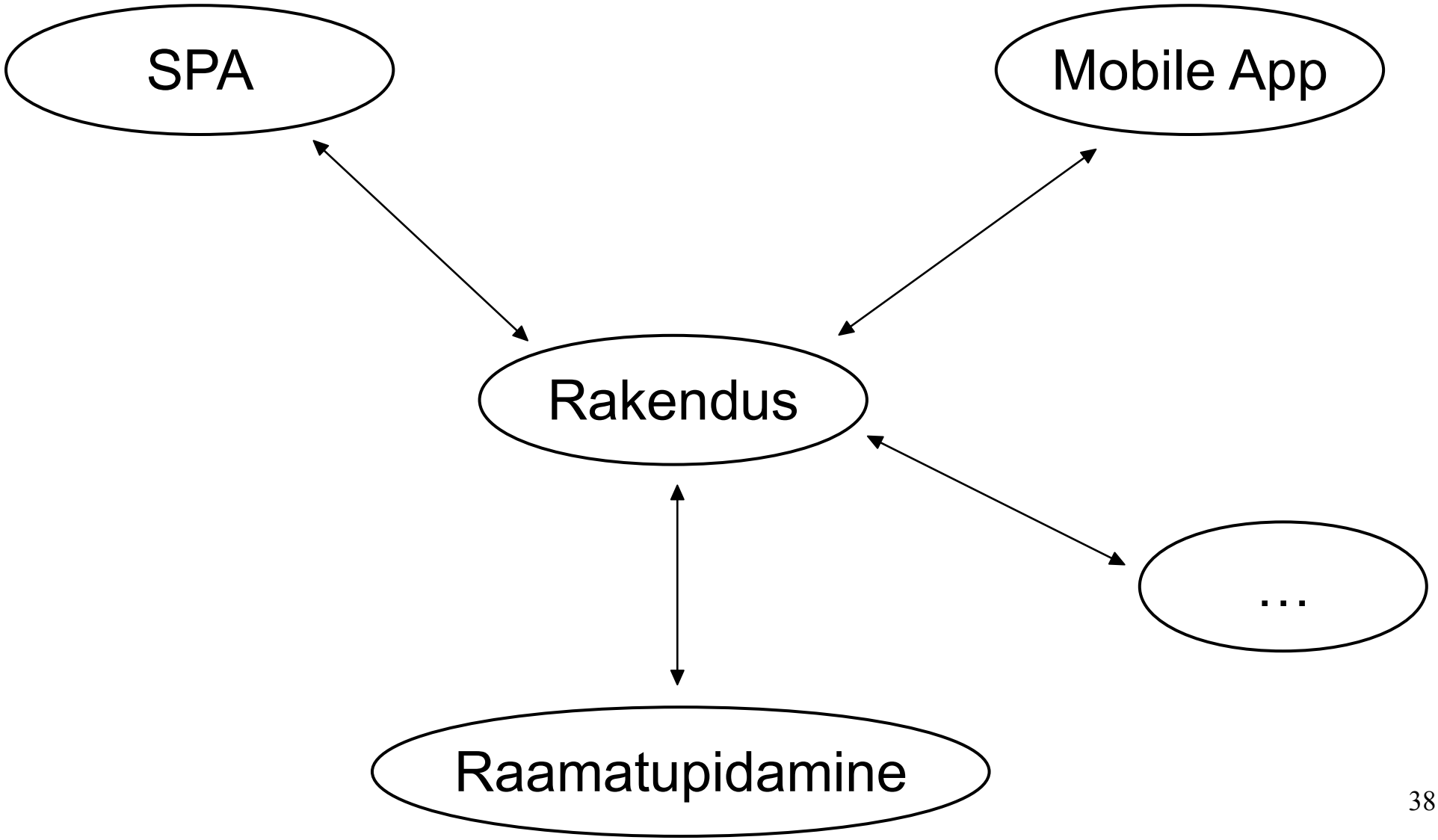
- CRUD
- Mitte CRUD (nt. töö tehtuks märkimine)

API loomisega alustamine

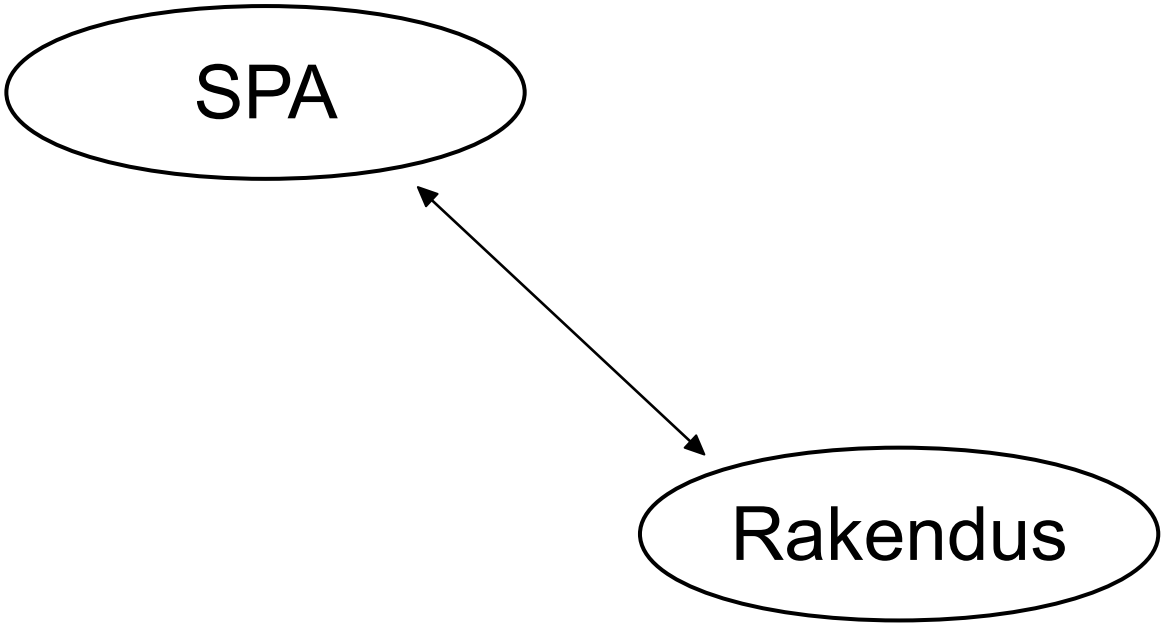
- Nt. https://developers.google.com/tasks/v1/reference/

Google, Facebook, ... vs Ettevõte x

API kavandamine



API kavandamine



Üks klient ja üks server

- Üldine nõuab rohkem tööd
- Üldisesse ei sobi konkreetse rakenduse spetsiifilised asjad
- Üldine ei sobi kellelegi väga hästi

Representational State Transfer

Resource

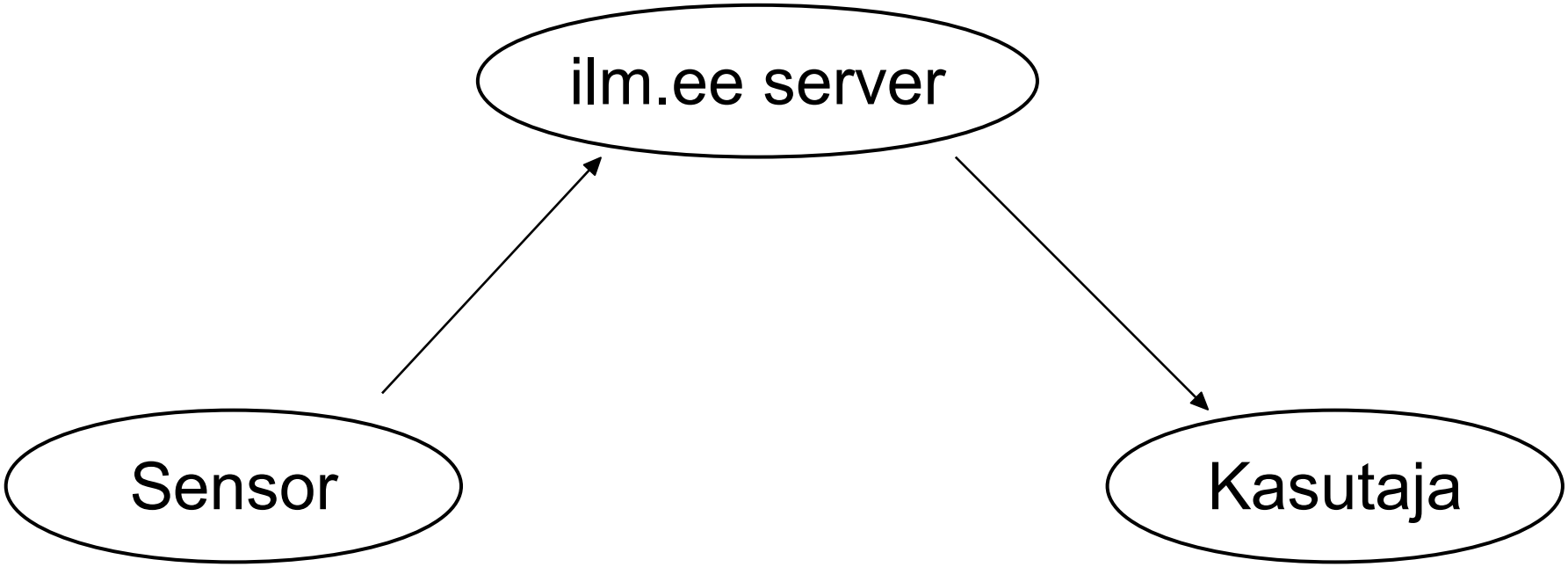
- contacts.html, map.jpg, report.docx, …
- Klient, tellimus, ilm, ...

Resource and representations


```
{
  "main": {
    "temp": 3.4,
    "humidity": 94,
    "pressure": 1016
  },
  "wind": {
    "speed": 7,
    "gust": 10,
    "deg": 5
  },
  "rain": {
    "1h": 0,
    "24h": 4.064,
    "today": 0
  },
}
```


Sooja on 3 kraadi, õhuniiskus on 94%, puhub põhjakaaretuul kiirusega 7-10 m/s.

Representational State Transfer



Resource manipulation through representations

```
GET /image.jpg HTTP/1.1
Host: localhost:8080
```

vs

```
POST /api/orders HTTP/1.1
Host: localhost:8080
Content-Type: application/json

{ "orderNumber": "A123" }
```

Spring Data

- Alustamine koodist
- Alustamine baasi skeemist

Alustamine baasi skeemist

- Hoiab rakendust koos
- Baasi skeem ja GUI mõjutavad rakenduse arhitektuuri

Alustamine koodist

- Sobib agiilse arendusega
- Samas peab baasile aeg-ajalt mõtlema
- Hiljem tuleb baasi skeem siiski fikseerida