Engineering
for a better
world.

DAIRYNET:
ICAR CODE GENERATION

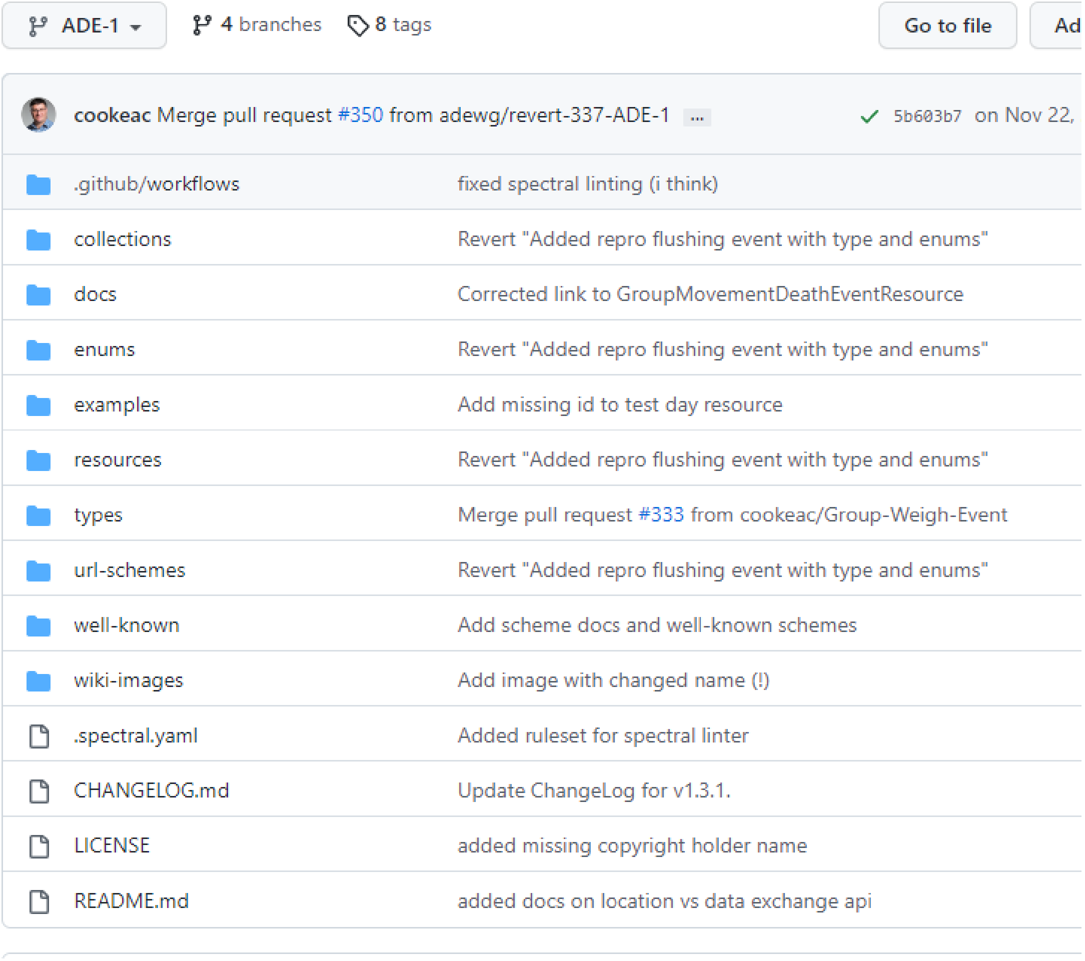
Andreas Schultz, GEA Farm Technologies GmbH, Germany
01/25/2023

Icar@DairyNet

- Why are we generating sources
- For what languages we are generating
- Which tools we use to generate

Base of generation:

ICAR-ADE github repository



cookeac Merge pull request #350 from adewg/revert-337-ADE-1 … ✓ 5b603b7 on Nov 22,

.github/workflows	fixed spectral linting (i think)
collections	Revert "Added repro flushing event with type and enums"
docs	Corrected link to GroupMovementDeathEventResource
enums	Revert "Added repro flushing event with type and enums"
examples	Add missing id to test day resource
resources	Revert "Added repro flushing event with type and enums"
types	Merge pull request #333 from cookeac/Group-Weigh-Event
url-schemes	Revert "Added repro flushing event with type and enums"
well-known	Add scheme docs and well-known schemes
wiki-images	Add image with changed name (!)
.spectral.yaml	Added ruleset for spectral linter
CHANGELOG.md	Update ChangeLog for v1.3.1.
LICENSE	added missing copyright holder name
README.md	added docs on location vs data exchange api

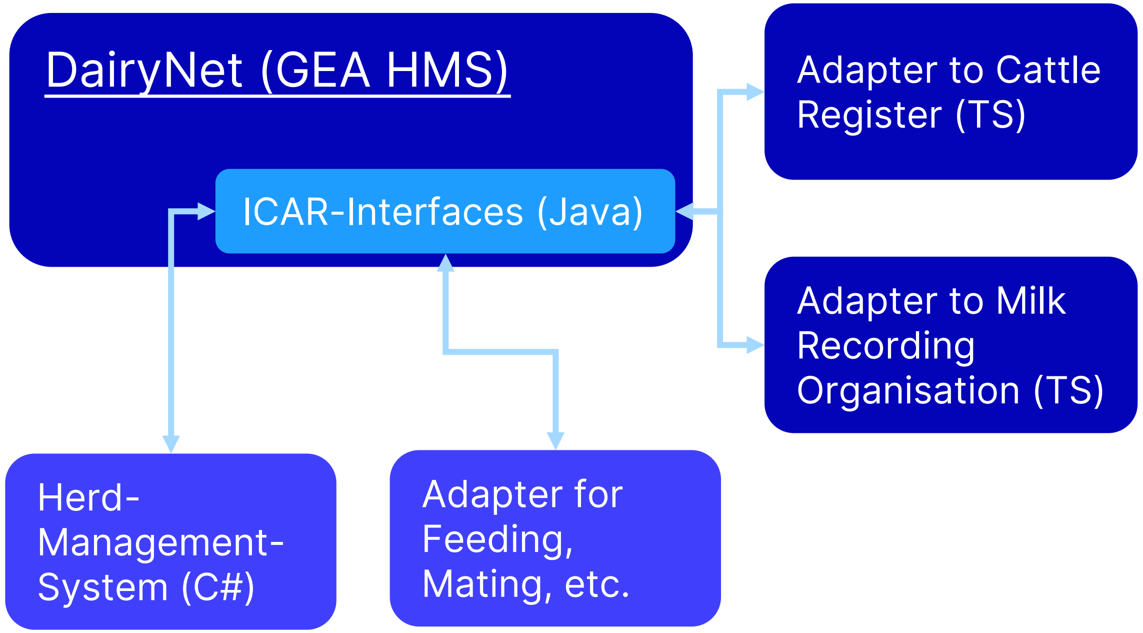
Contact: andreas.schultz@gea.com

Icar@DairyNet

- DairyNet (GEAs Herd Management Software) has to interact with:
  - Milk-Recording-Organisation (MRO):
    - Submit MilkingVisits (incl. Samples) to MRO
    - Request TestDayResults from MRO
  - Cattle-Register (CR):
    - Submit and request animals and animal-events (reproduction) to/from CR
  - Other 3rd. Party Herd Management Software, etc.

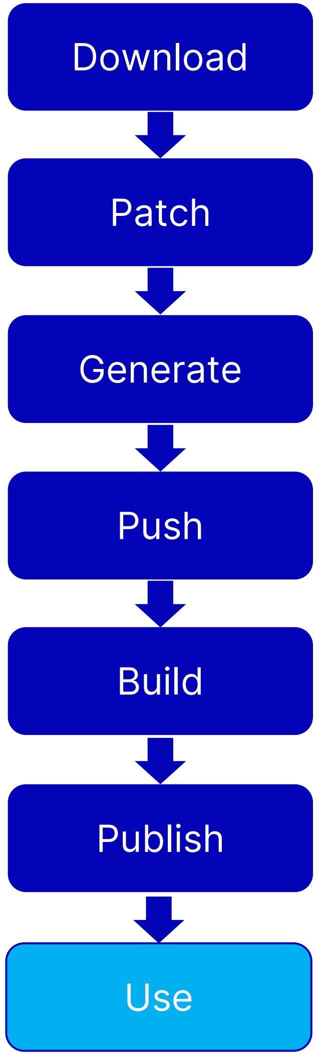
- Programming Languages:
  - Java, Spring-Boot
  - TypeScript (TS)

- Generation Tools:
  - Maven
  - OpenApiGenerator (openapitools.org, release 6.2.1)
  - TypeScript-Generator (habarta.cz, release 2.32.889)


DairyNet (GEA HMS)
ICAR-Interfaces (Java)
Adapter to Cattle Register (TS)
Adapter to Milk Recording Organisation (TS)
Herd-Management-System (C#)
Adapter for Feeding, Mating, etc.

Contact: andreas.schultz@gea.com

Icar@DairyNet: Steps of generation process



Download JSON from ICAR-repository (OpenApi-Specification) – no clone, no fork

Patch JSON to e.g. add custom extensions (2. managementTag) or to adjust filter

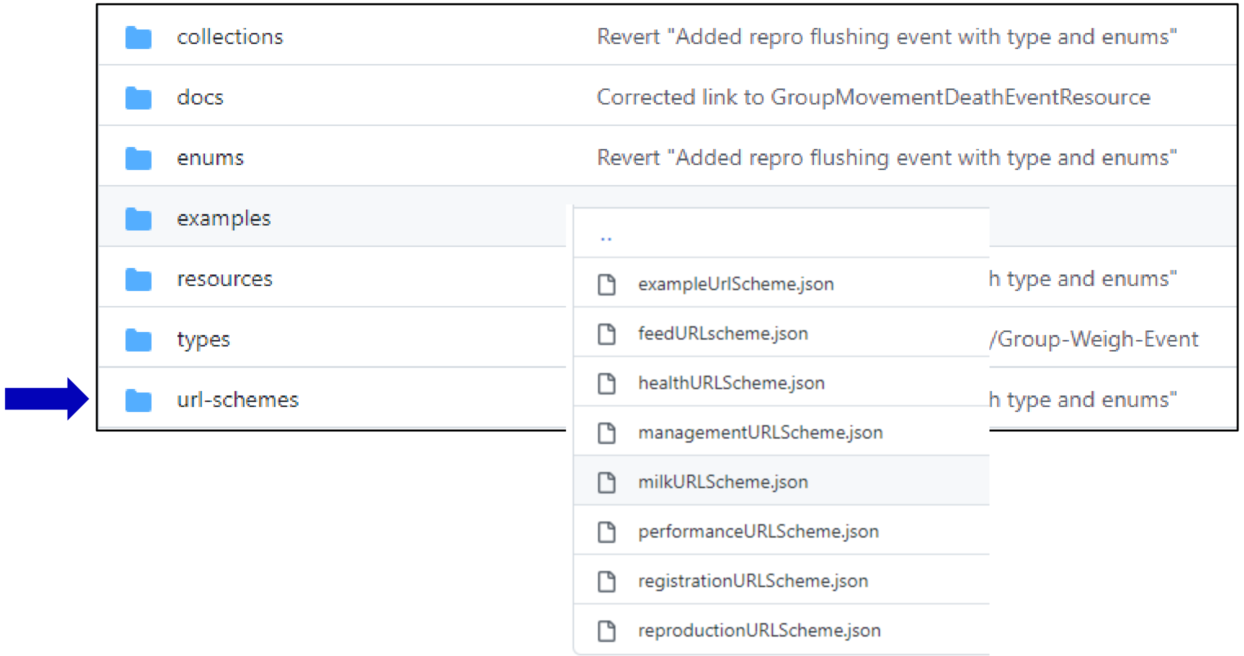
Generate sources from a provided JSON-scheme from download - Java (Spring Boot), TypeScript

Push sources to own (GEA) repository

Build sources at local or build server

Publish artifacts to local repository

Use

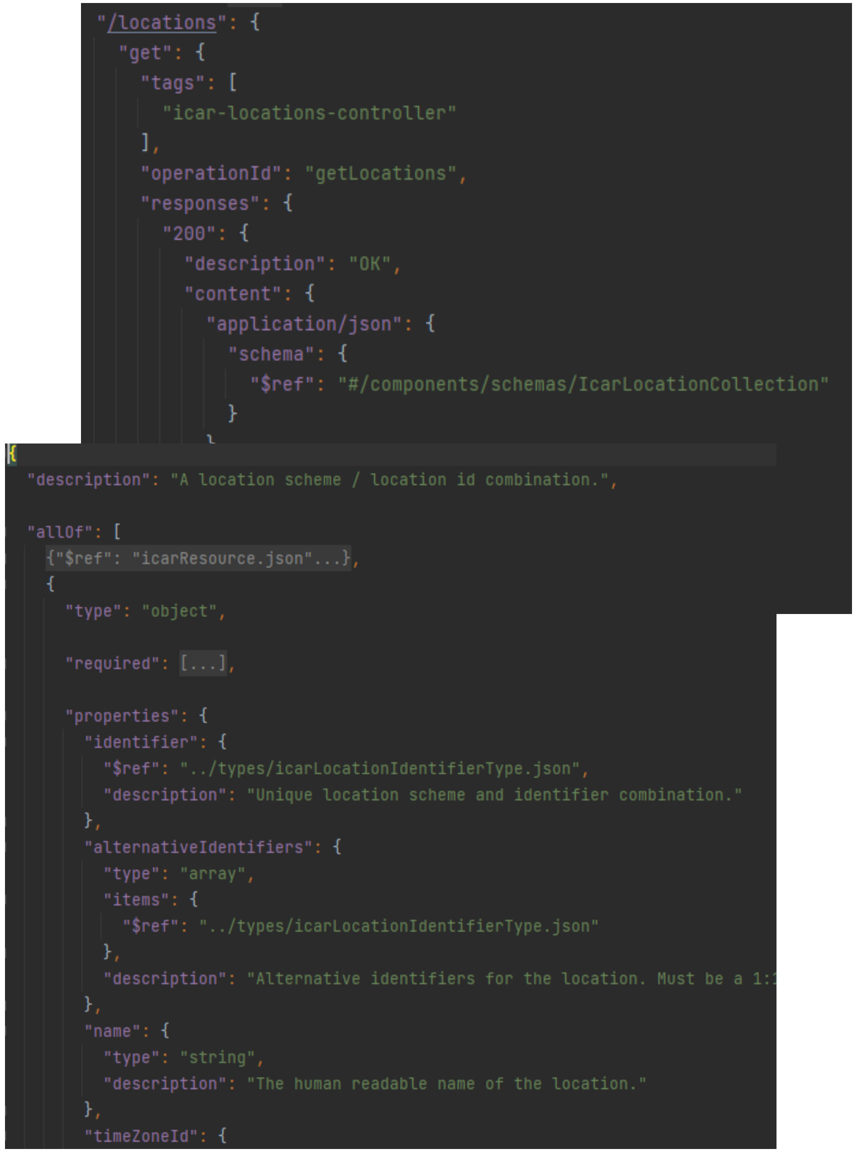


collections	Revert "Added repro flushing event with type and enums"
docs	Corrected link to GroupMovementDeathEventResource
enums	Revert "Added repro flushing event with type and enums"
examples	
resources	h type and enums"
types	/Group-Weigh-Event
url-schemes	h type and enums"

..
- exampleUrlScheme.json
- feedURLscheme.json
- healthURLScheme.json
- managementURLScheme.json
- milkURLScheme.json
- performanceURLScheme.json
- registrationURLScheme.json
- reproductionURLScheme.json

Contact: andreas.schultz@gea.com

Icar@DairyNet: From scheme to data transfer objects

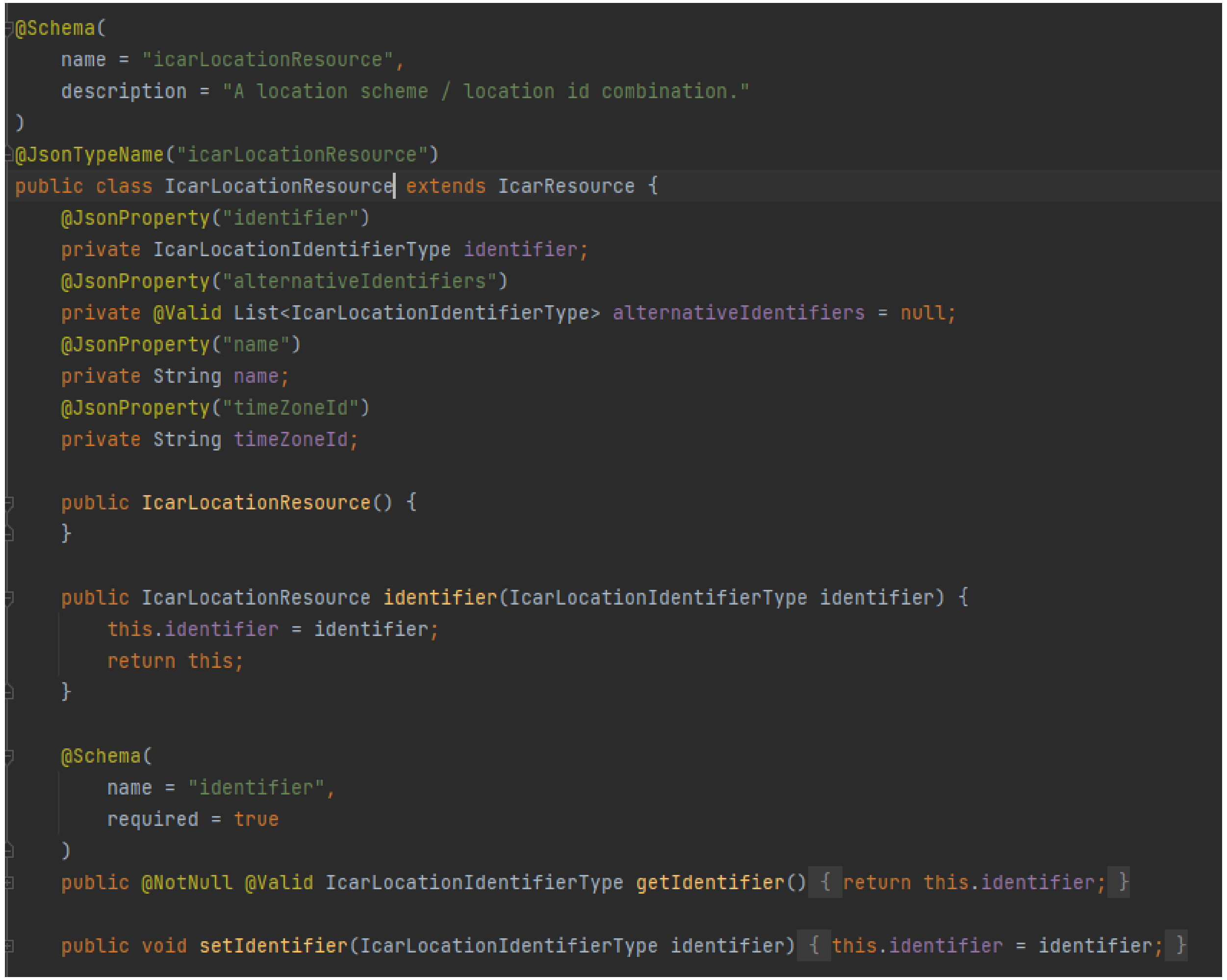


```
"/locations": {
  "get": {
    "tags": [
      "icar-locations-controller"
    ],
    "operationId": "getLocations",
    "responses": {
      "200": {
        "description": "OK",
        "content": {
          "application/json": {
            "schema": {
              "$ref": "#/components/schemas/IcarLocationCollection"
            }
          }
```

```
{
  "description": "A location scheme / location id combination.",

  "allOf": [
    {"$ref": "icarResource.json"...},
    {
      "type": "object",

      "required": [...],

      "properties": {
        "identifier": {
          "$ref": "../types/icarLocationIdentifierType.json",
          "description": "Unique location scheme and identifier combination."
        },
        "alternativeIdentifiers": {
          "type": "array",
          "items": {
            "$ref": "../types/icarLocationIdentifierType.json"
          },
          "description": "Alternative identifiers for the location. Must be a 1:
        },
        "name": {
          "type": "string",
          "description": "The human readable name of the location."
        },
        "timeZoneId": {
```

JSON

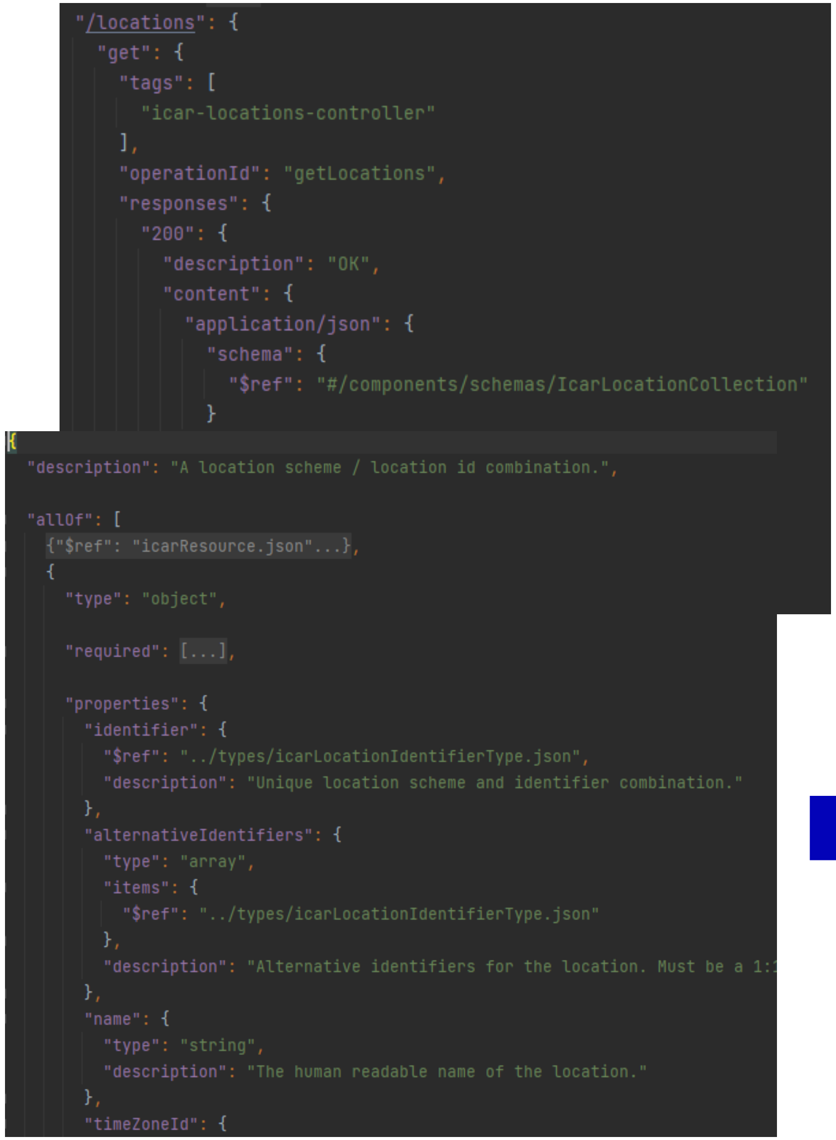
Java



```java
@Schema(
    name = "icarLocationResource",
    description = "A location scheme / location id combination."
)
@JsonTypeName("icarLocationResource")
public class IcarLocationResource extends IcarResource {
    @JsonProperty("identifier")
    private IcarLocationIdentifierType identifier;
    @JsonProperty("alternativeIdentifiers")
    private @Valid List<IcarLocationIdentifierType> alternativeIdentifiers = null;
    @JsonProperty("name")
    private String name;
    @JsonProperty("timeZoneId")
    private String timeZoneId;

    public IcarLocationResource() {
    }

    public IcarLocationResource identifier(IcarLocationIdentifierType identifier) {
        this.identifier = identifier;
        return this;
    }

    @Schema(
        name = "identifier",
        required = true
    )
    public @NotNull @Valid IcarLocationIdentifierType getIdentifier() { return this.identifier; }

    public void setIdentifier(IcarLocationIdentifierType identifier) { this.identifier = identifier; }
```

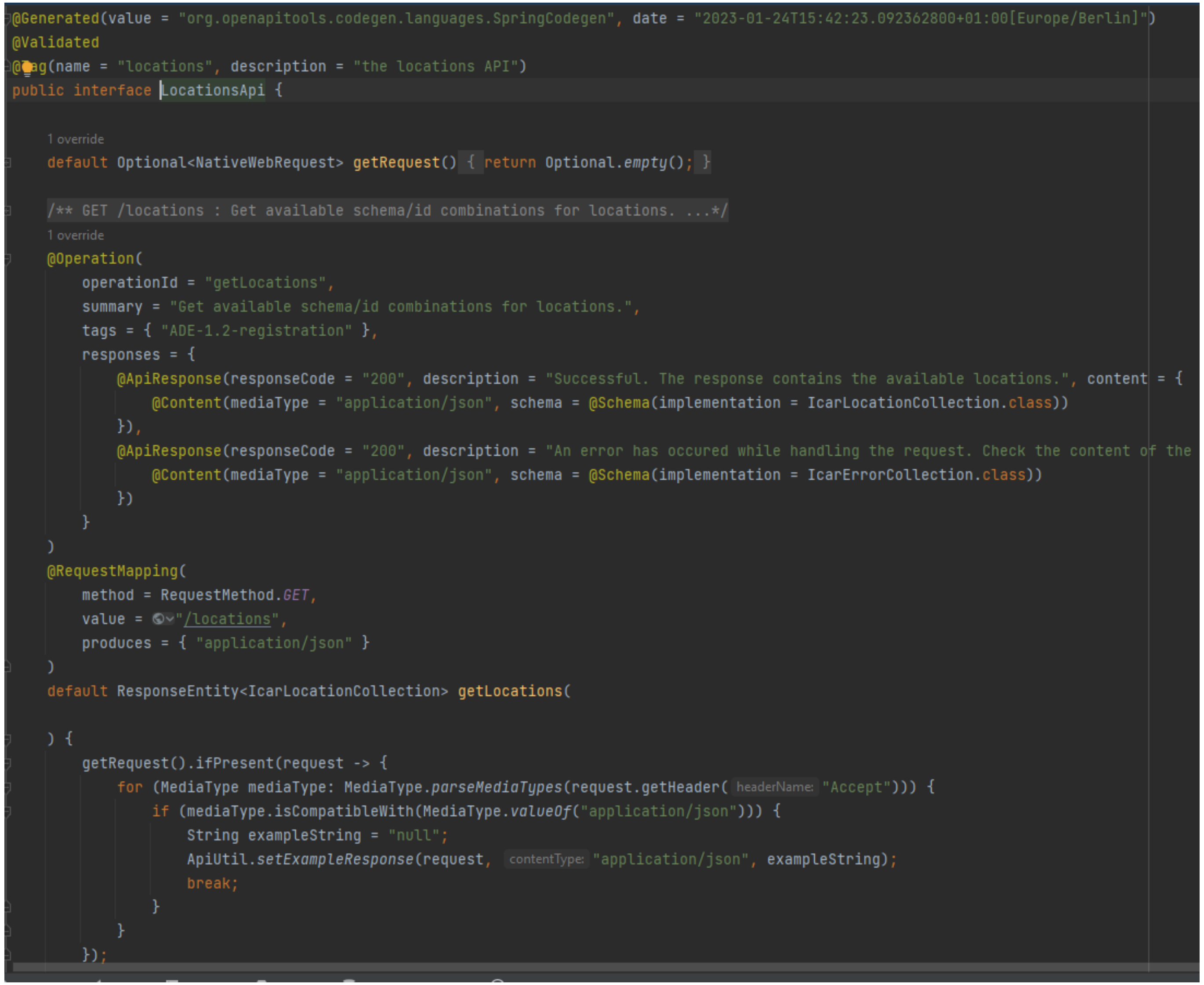
Contact: andreas.schultz@gea.com

Icar@DairyNet: From scheme to ReST-interfaces



```
"/locations": {
  "get": {
    "tags": [
      "icar-locations-controller"
    ],
    "operationId": "getLocations",
    "responses": {
      "200": {
        "description": "OK",
        "content": {
          "application/json": {
            "schema": {
              "$ref": "#/components/schemas/IcarLocationCollection"
            }
```

```
{
  "description": "A location scheme / location id combination.",

  "allOf": [
    {"$ref": "icarResource.json"...},
    {
      "type": "object",

      "required": [...],

      "properties": {
        "identifier": {
          "$ref": "../types/icarLocationIdentifierType.json",
          "description": "Unique location scheme and identifier combination."
        },
        "alternativeIdentifiers": {
          "type": "array",
          "items": {
            "$ref": "../types/icarLocationIdentifierType.json"
          },
          "description": "Alternative identifiers for the location. Must be a 1:
        },
        "name": {
          "type": "string",
          "description": "The human readable name of the location."
        },
        "timeZoneId": {
```

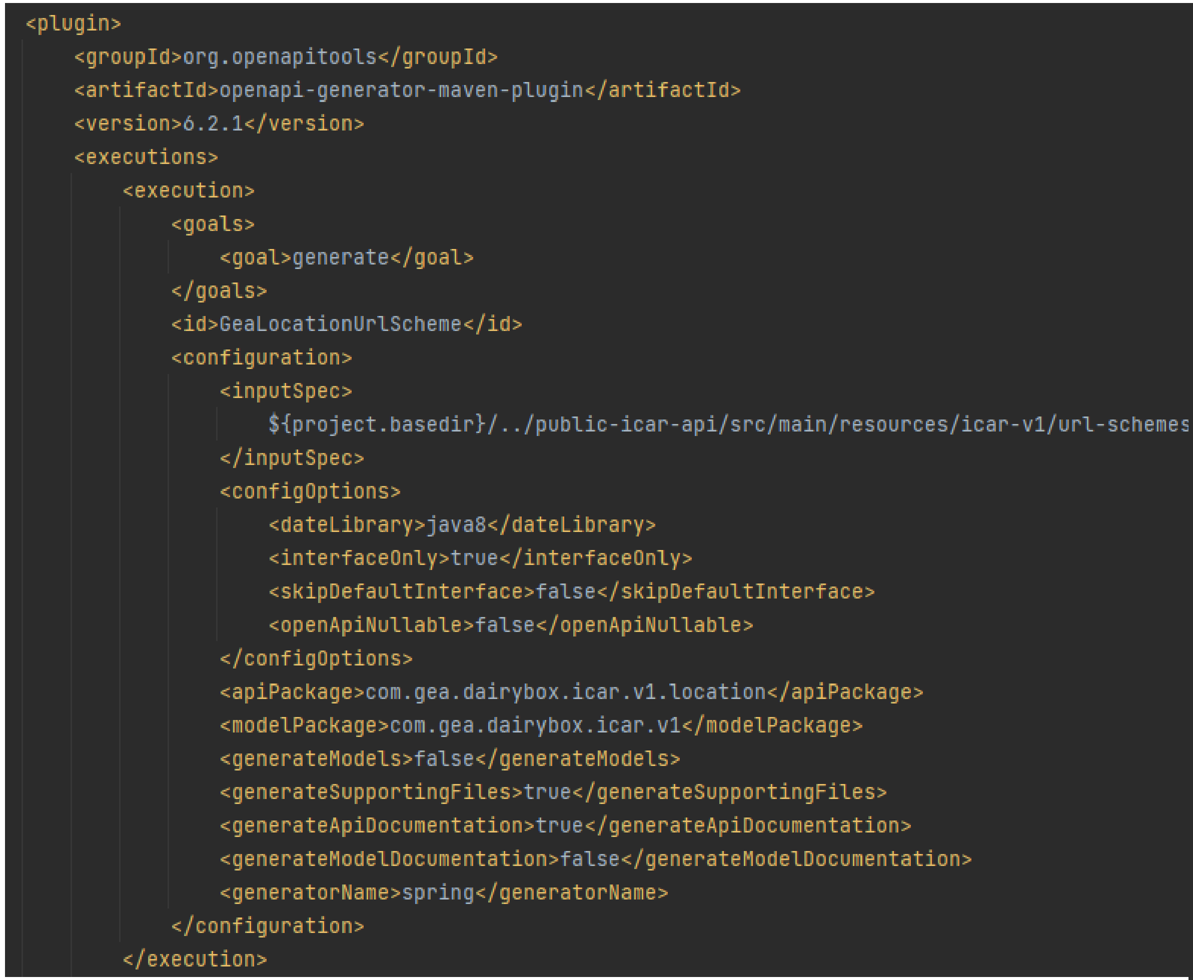


```
@Generated(value = "org.openapitools.codegen.languages.SpringCodegen", date = "2023-01-24T15:42:23.092362800+01:00[Europe/Berlin]")
@Validated
@Tag(name = "locations", description = "the locations API")
public interface LocationsApi {

    1 override
    default Optional<NativeWebRequest> getRequest() { return Optional.empty(); }

    /** GET /locations : Get available schema/id combinations for locations. ...*/
    1 override
    @Operation(
        operationId = "getLocations",
        summary = "Get available schema/id combinations for locations.",
        tags = { "ADE-1.2-registration" },
        responses = {
            @ApiResponse(responseCode = "200", description = "Successful. The response contains the available locations.", content = {
                @Content(mediaType = "application/json", schema = @Schema(implementation = IcarLocationCollection.class))
            }),
            @ApiResponse(responseCode = "200", description = "An error has occured while handling the request. Check the content of the
                @Content(mediaType = "application/json", schema = @Schema(implementation = IcarErrorCollection.class))
            })
        }
    )
    @RequestMapping(
        method = RequestMethod.GET,
        value = "/locations",
        produces = { "application/json" }
    )
    default ResponseEntity<IcarLocationCollection> getLocations(

    ) {
        getRequest().ifPresent(request -> {
            for (MediaType mediaType: MediaType.parseMediaTypes(request.getHeader( headerName: "Accept"))) {
                if (mediaType.isCompatibleWith(MediaType.valueOf("application/json"))) {
                    String exampleString = "null";
                    ApiUtil.setExampleResponse(request,  contentType: "application/json", exampleString);
                    break;
                }
            }
        });
```

Contact: andreas.schultz@gea.com

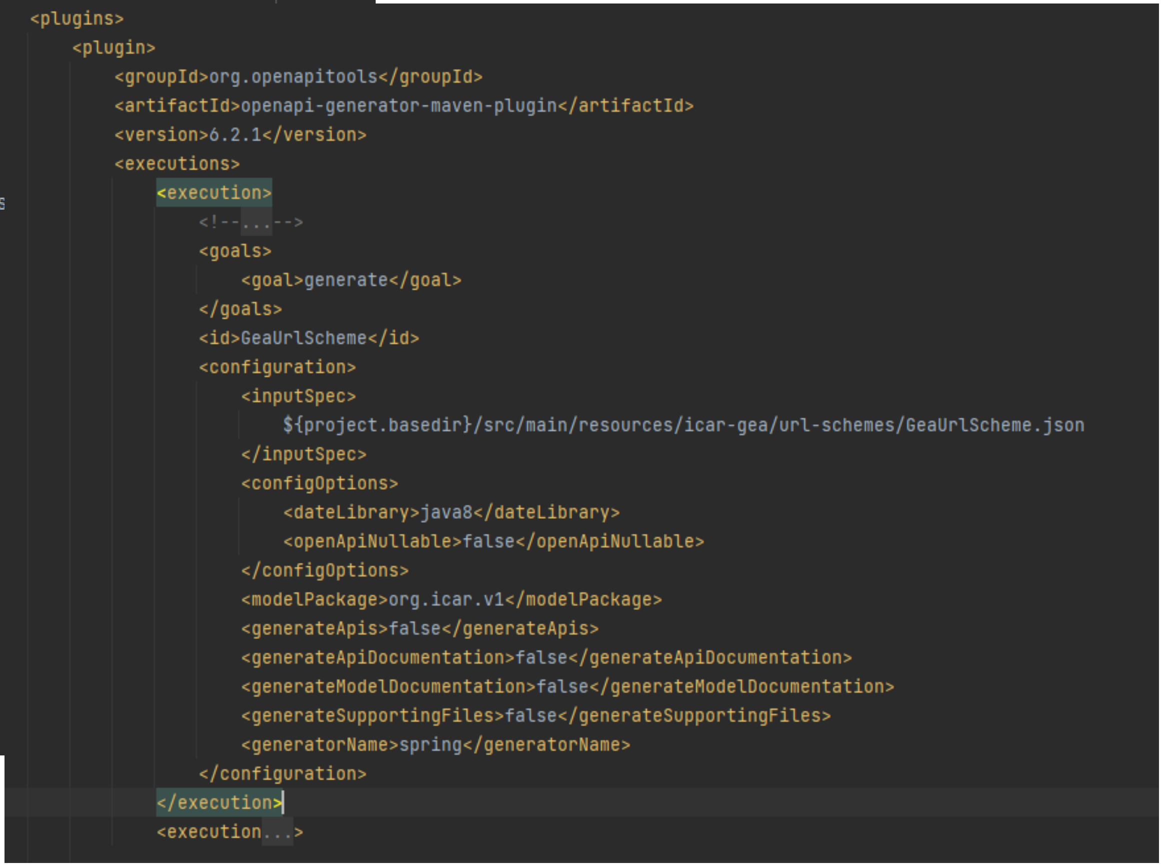
Icar@DairyNet: Technical details Maven – Java

Java (Interfaces)



```
<plugin>
    <groupId>org.openapitools</groupId>
    <artifactId>openapi-generator-maven-plugin</artifactId>
    <version>6.2.1</version>
    <executions>
        <execution>
            <goals>
                <goal>generate</goal>
            </goals>
            <id>GeaLocationUrlScheme</id>
            <configuration>
                <inputSpec>
                    ${project.basedir}/../public-icar-api/src/main/resources/icar-v1/url-schemes
                </inputSpec>
                <configOptions>
                    <dateLibrary>java8</dateLibrary>
                    <interfaceOnly>true</interfaceOnly>
                    <skipDefaultInterface>false</skipDefaultInterface>
                    <openApiNullable>false</openApiNullable>
                </configOptions>
                <apiPackage>com.gea.dairybox.icar.v1.location</apiPackage>
                <modelPackage>com.gea.dairybox.icar.v1</modelPackage>
                <generateModels>false</generateModels>
                <generateSupportingFiles>true</generateSupportingFiles>
                <generateApiDocumentation>true</generateApiDocumentation>
                <generateModelDocumentation>false</generateModelDocumentation>
                <generatorName>spring</generatorName>
            </configuration>
        </execution>
```

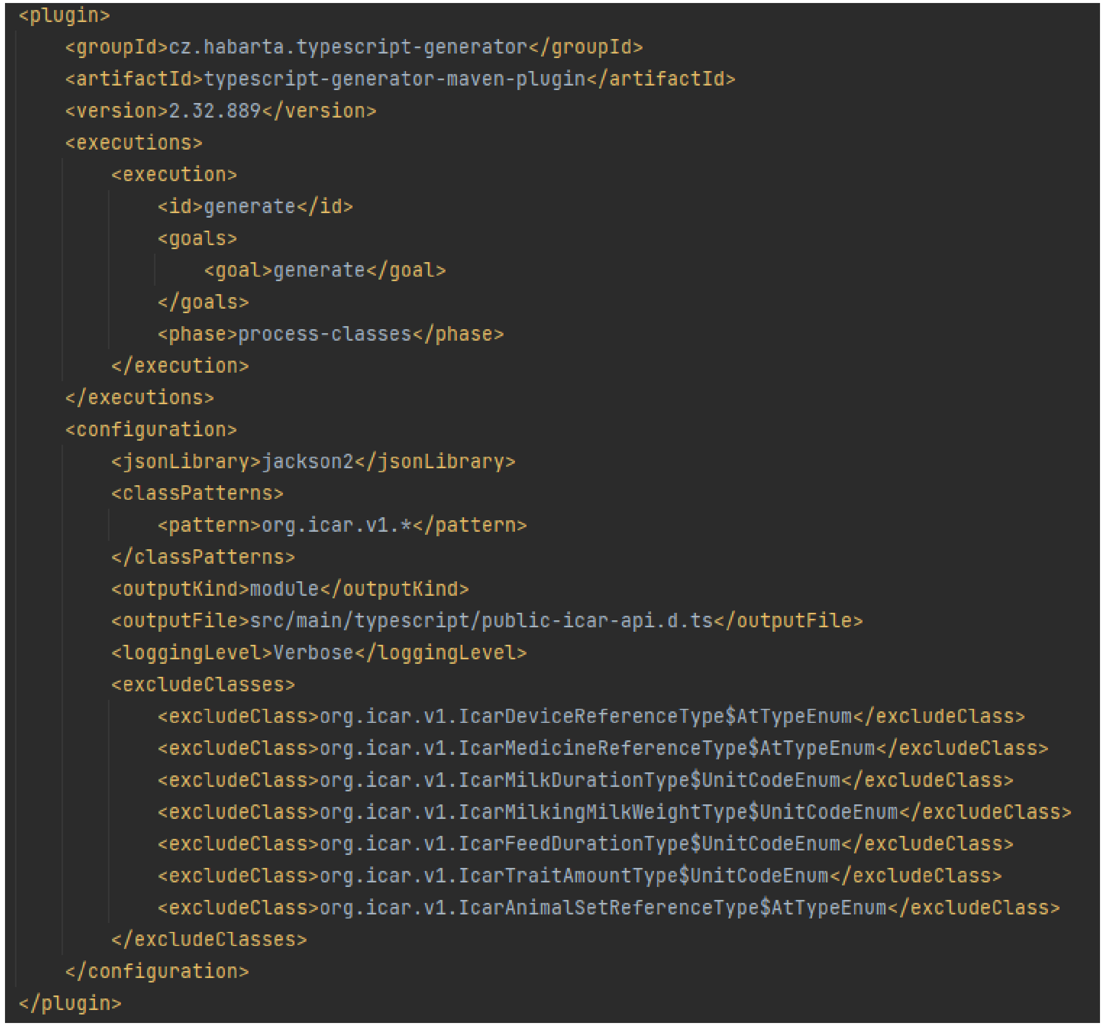
Java (DTOs)



```
<plugins>
    <plugin>
        <groupId>org.openapitools</groupId>
        <artifactId>openapi-generator-maven-plugin</artifactId>
        <version>6.2.1</version>
        <executions>
            <execution>
                <!--...-->
                <goals>
                    <goal>generate</goal>
                </goals>
                <id>GeaUrlScheme</id>
                <configuration>
                    <inputSpec>
                        ${project.basedir}/src/main/resources/icar-gea/url-schemes/GeaUrlScheme.json
                    </inputSpec>
                    <configOptions>
                        <dateLibrary>java8</dateLibrary>
                        <openApiNullable>false</openApiNullable>
                    </configOptions>
                    <modelPackage>org.icar.v1</modelPackage>
                    <generateApis>false</generateApis>
                    <generateApiDocumentation>false</generateApiDocumentation>
                    <generateModelDocumentation>false</generateModelDocumentation>
                    <generateSupportingFiles>false</generateSupportingFiles>
                    <generatorName>spring</generatorName>
                </configuration>
            </execution>
            <execution...>
```

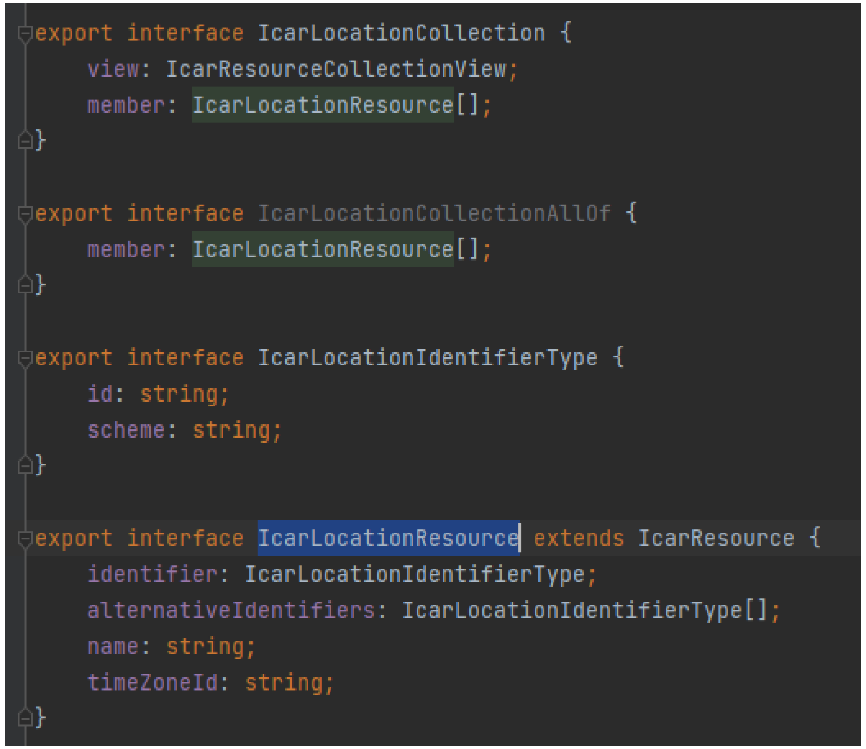
Contact: andreas.schultz@gea.com

Icar@DairyNet: Technical details Maven – TypeScript



```xml
<plugin>
    <groupId>cz.habarta.typescript-generator</groupId>
    <artifactId>typescript-generator-maven-plugin</artifactId>
    <version>2.32.889</version>
    <executions>
        <execution>
            <id>generate</id>
            <goals>
                <goal>generate</goal>
            </goals>
            <phase>process-classes</phase>
        </execution>
    </executions>
    <configuration>
        <jsonLibrary>jackson2</jsonLibrary>
        <classPatterns>
            <pattern>org.icar.v1.*</pattern>
        </classPatterns>
        <outputKind>module</outputKind>
        <outputFile>src/main/typescript/public-icar-api.d.ts</outputFile>
        <loggingLevel>Verbose</loggingLevel>
        <excludeClasses>
            <excludeClass>org.icar.v1.IcarDeviceReferenceType$AtTypeEnum</excludeClass>
            <excludeClass>org.icar.v1.IcarMedicineReferenceType$AtTypeEnum</excludeClass>
            <excludeClass>org.icar.v1.IcarMilkDurationType$UnitCodeEnum</excludeClass>
            <excludeClass>org.icar.v1.IcarMilkingMilkWeightType$UnitCodeEnum</excludeClass>
            <excludeClass>org.icar.v1.IcarFeedDurationType$UnitCodeEnum</excludeClass>
            <excludeClass>org.icar.v1.IcarTraitAmountType$UnitCodeEnum</excludeClass>
            <excludeClass>org.icar.v1.IcarAnimalSetReferenceType$AtTypeEnum</excludeClass>
        </excludeClasses>
    </configuration>
</plugin>
```

TypeScript



```typescript
export interface IcarLocationCollection {
    view: IcarResourceCollectionView;
    member: IcarLocationResource[];
}

export interface IcarLocationCollectionAllOf {
    member: IcarLocationResource[];
}

export interface IcarLocationIdentifierType {
    id: string;
    scheme: string;
}

export interface IcarLocationResource extends IcarResource {
    identifier: IcarLocationIdentifierType;
    alternativeIdentifiers: IcarLocationIdentifierType[];
    name: string;
    timeZoneId: string;
}
```

Contact: andreas.schultz@gea.com

Icar@DairyNet

- Contact
  - DairyNet: martin.schnare@gea.com, Product-Manager
  - Icar@DairyNet: andreas.schultz@gea.com, Senior Software-Engineer

GEA

Engineering
for a better
world.

GEA.com