Walkthrough and exercise for agile metamodelling in AOAME

1 Walkthrough

- Show the 335 cloud services in the ontology by firing the query:

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX bpaas: <http://ikm-group.ch/archimeo/bpaas#>

SELECT ?CS
WHERE {
  ?C rdf:type bpaas:CloudService.
  ?C rdfs:label ?CS .
}

- Extend Service Task with "Send Invoice" AND add Semantic Mapping: hasAPQCCategory and select Transmit billing data to customer.

- You can prove that the relation has been entered by firing the following query:

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX bpaas: <http://ikm-group.ch/archimeo/bpaas#>
prefix mod: <http://fhnw.ch/modelingEnvironment/ModelOntology#>
prefix lo: <http://fhnw.ch/modelingEnvironment/LanguageOntology#>
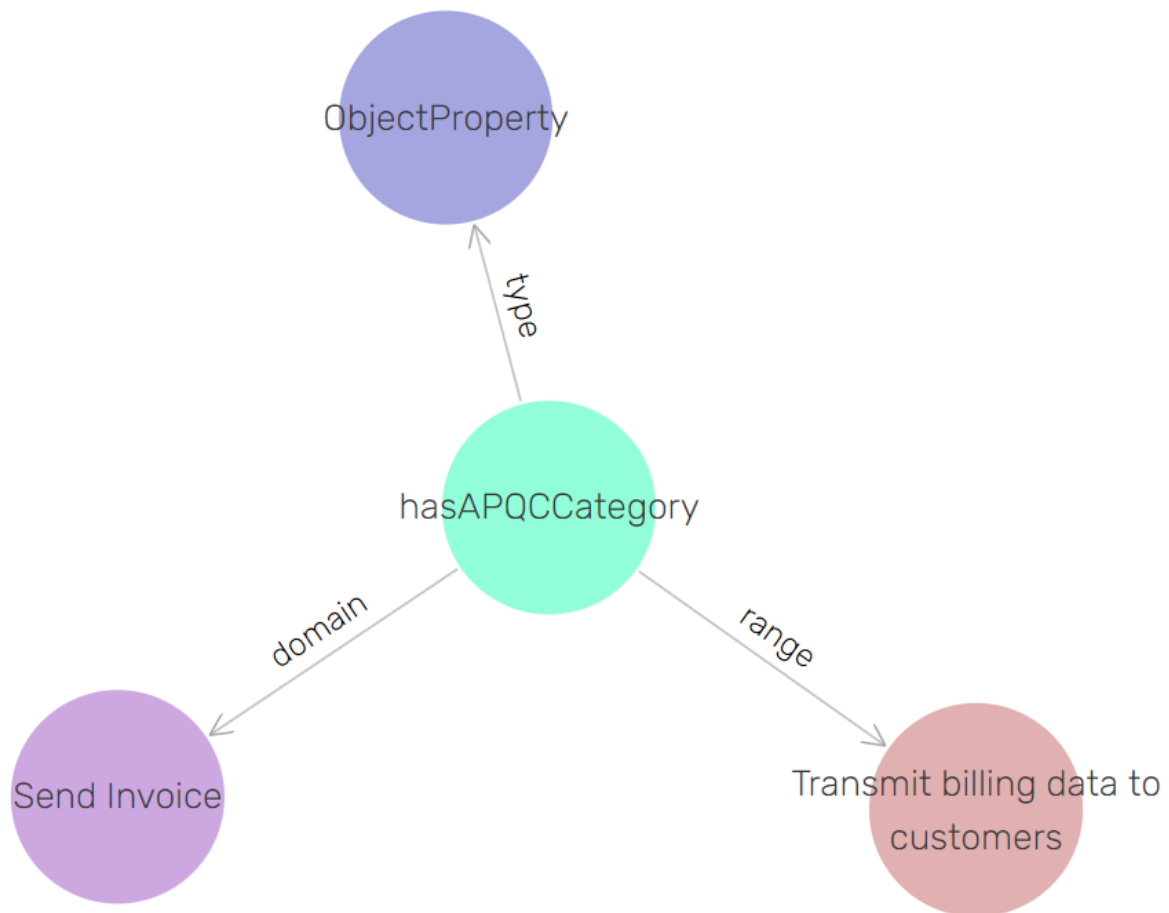
SELECT ?labelSubject ?labelObject
WHERE {
 lo:hasAPQCCategory rdfs:range ?object .
 lo:hasAPQCCategory rdfs:domain ?subject .
 ?object rdfs:label ?labelObject .
 ?subject rdfs:label ?labelSubject .
}

labelSubject		labelObject
1	"Send Invoice"	"Transmit billing data to customers"

- Select all the CloudServices that point to the same APQC category

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX bpaas: <http://ikm-group.ch/archimeo/bpaas#>
prefix mod: <http://fhnw.ch/modelingEnvironment/ModelOntology#>
prefix lo: <http://fhnw.ch/modelingEnvironment/LanguageOntology#>

SELECT ?CS ?label
WHERE {
  ?CS rdf:type bpaas:CloudService .
  ?CS bpaas:cloudServiceHasAPQC ?csAPQC .
  ?CS rdfs:label ?label .
  mod:SendInvoice_5eb7a18c-1686-49ee-8455-02ea15f6a024 rdf:type ?modellingElement .
  ?predicate rdfs:domain ?modellingElement .
 ?predicate rdfs:range ?APQC .
  FILTER (?csAPQC = ?APQC).
}

Show results:

cs		label
1	<http://ikm-group.ch/archiMEO/bdata#Ninja_email>	"Ninja_email"
2	<http://ikm-group.ch/archiMEO/bdata#InvoiceNinja>	"InvoiceNinja"
3	<http://ikm-group.ch/archiMEO/bdata#NextCloud>	"NextCloud"
4	<http://ikm-group.ch/archiMEO/bdata#YMENS_email>	"YMENS Email"
5	<http://ikm-group.ch/archiMEO/bdata#Drive>	"Drive"
6	<http://ikm-group.ch/archiMEO/bdata#Gmail>	"Gmail"
7	<http://ikm-group.ch/archiMEO/bdata#Dropbox>	"Dropbox"
8	<http://ikm-group.ch/archiMEO/bdata#MailJet>	"MailJet"

Below, one can see that the new relation *hasAPCQCategory* allows the navigation to the Cloud-Service.
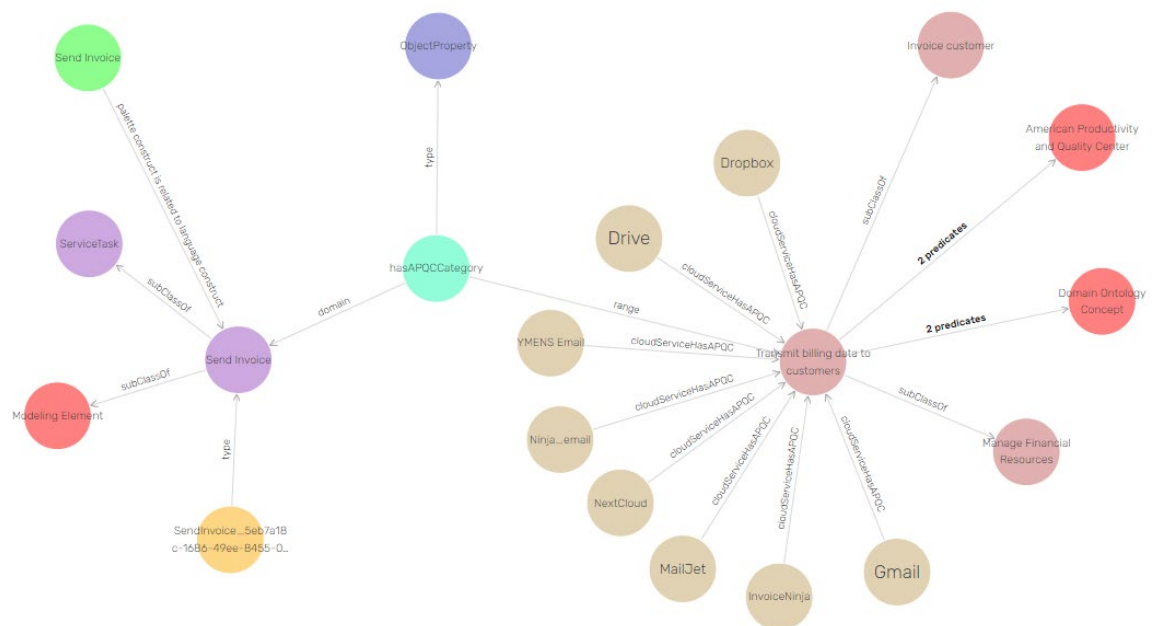
- Let's now add the monthly availability of the cloud service to the extended element by adding the attribute to the extended element: *Downtime in min: Decimal.*
- Cancel the new model element Send Invoice and re-instantiate it
- Then in the newly instantiated model element we add 60 min by right clicking in the Send Invoice element from the canvas -> Model Element Attributes.
    - 60 min is a requirement, therefore the wanted cloud service should have up to 60 monthly minutes downtime.

-
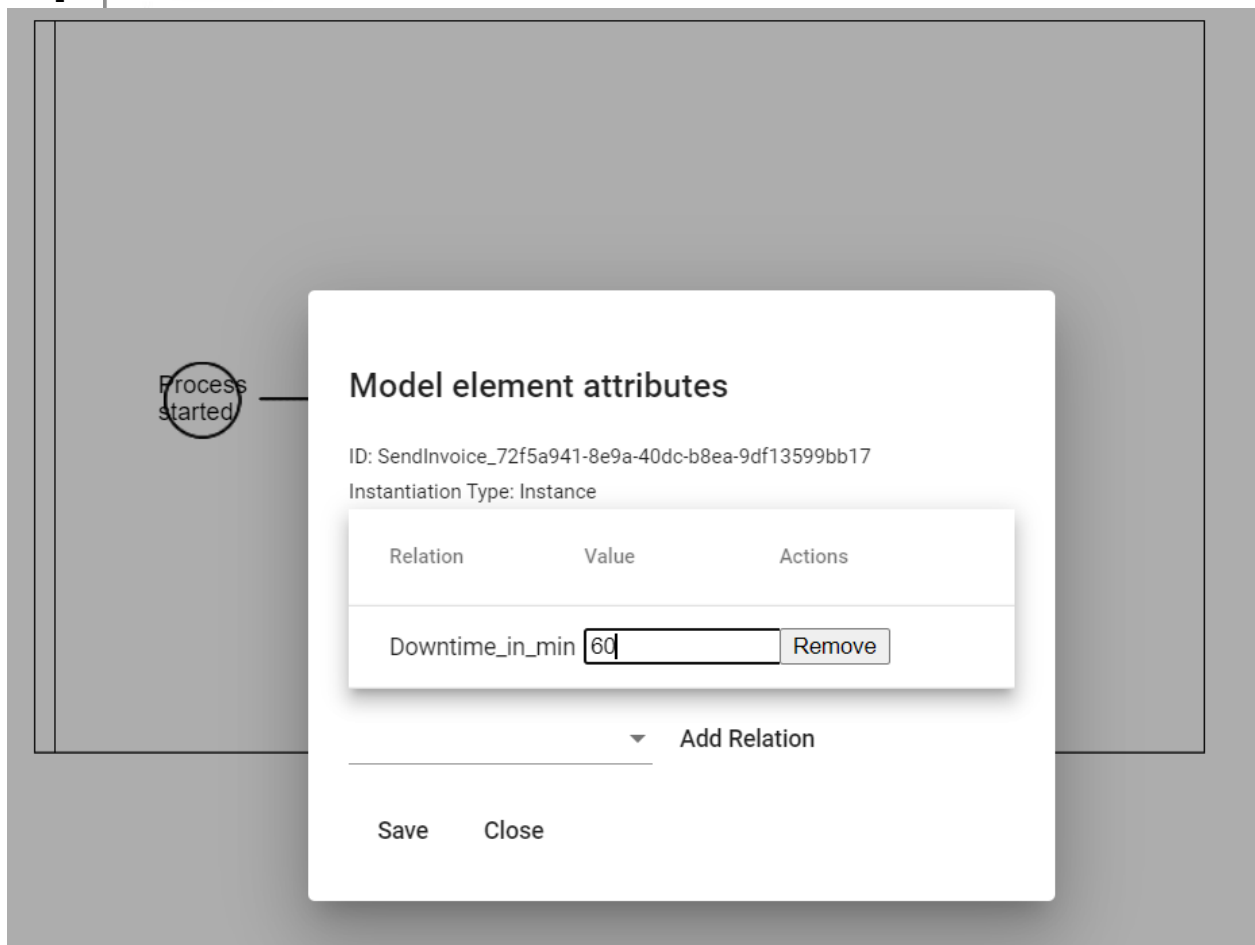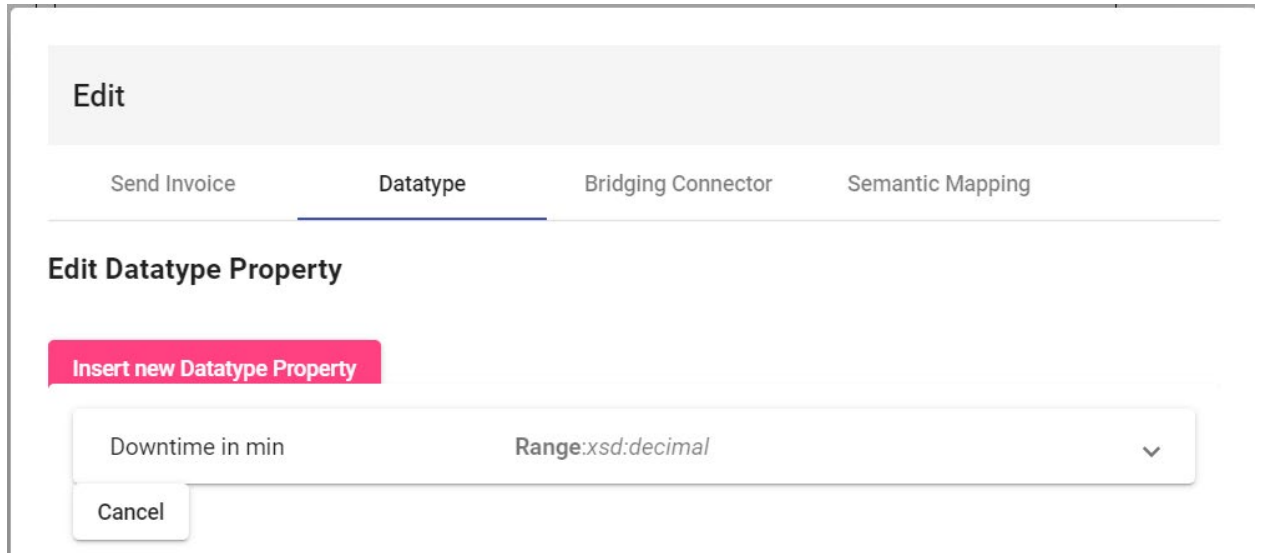
- To look at the downtime that was entered through the model, we can fire the following query:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX bpaas: <http://ikm-group.ch/archimeo/bpaas#>
prefix mod: <http://fhnw.ch/modelingEnvironment/ModelOntology#>
prefix lo: <http://fhnw.ch/modelingEnvironment/LanguageOntology#>

SELECT ?downtime
WHERE {

 mod:SendInvoice_72f5a941-8e9a-40dc-b8ea-9df13599bb17 lo:Downtime_in_min  ?downtime

}
```

2 Exercise:

Create the query that retrieve the Cloud Services for the specified business process. Note that we are looking for Cloud Services that have a downtime less than 60 min.