PROJECT DIGITALIZATION

Working Group 4 – Building an ESG Knowledge Graph

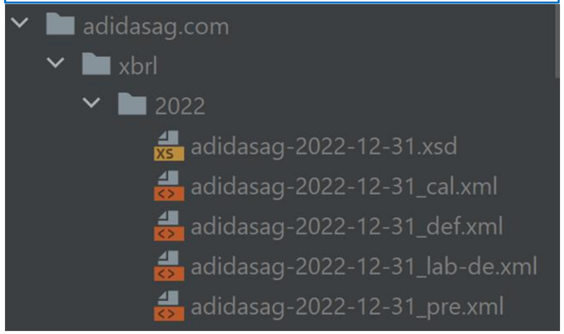
CONTENT

1. Extract data from XBRL
2. Ontologies
3. Knowledge Graph Construction
4. Add data to Knowledge Graph
    a. External data from Wikidata and DBpedia
    b. Text embeddings
5. Knowledge Graph Bot
6. Knowledge Graph Queries

1. EXTRACT DATA FROM XBRL

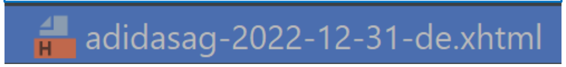
ESRS: XBRL Delivery Requirement

ESEF Taxonomy Schema



```
adidasag.com
  xbrl
    2022
      adidasag-2022-12-31.xsd
      adidasag-2022-12-31_cal.xml
      adidasag-2022-12-31_def.xml
      adidasag-2022-12-31_lab-de.xml
      adidasag-2022-12-31_pre.xml
```

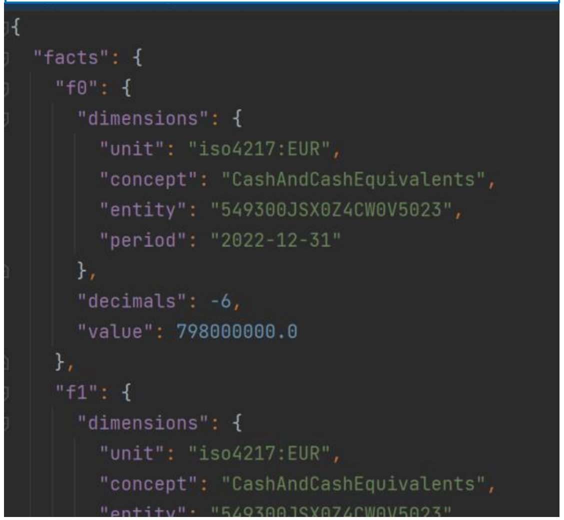
Instance Document



adidasag-2022-12-31-de.xhtml

1. extract and calculate:

JSON

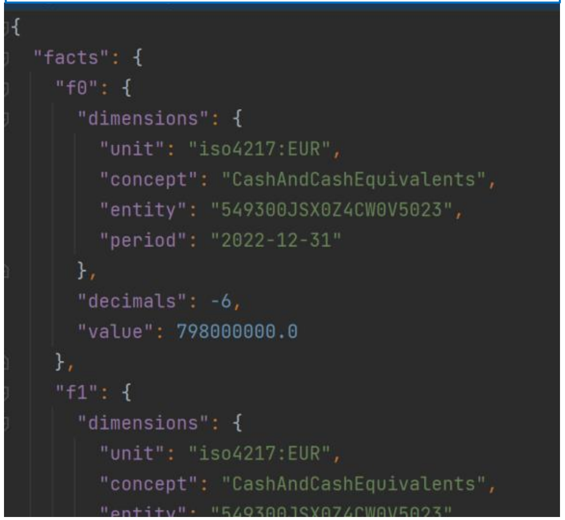


```
{
  "facts": {
    "f0": {
      "dimensions": {
        "unit": "iso4217:EUR",
        "concept": "CashAndCashEquivalents",
        "entity": "549300JSX0Z4CW0V5023",
        "period": "2022-12-31"
      },
      "decimals": -6,
      "value": 798000000.0
    },
    "f1": {
      "dimensions": {
        "unit": "iso4217:EUR",
        "concept": "CashAndCashEquivalents",
        "entity": "549300JSX0Z4CW0V5023"
```

Extract data from XBRL into standardized JSON-Files

1. EXTRACT DATA FROM XBRL

JSON



```
{
  "facts": {
    "f0": {
      "dimensions": {
        "unit": "iso4217:EUR",
        "concept": "CashAndCashEquivalents",
        "entity": "549300JSX0Z4CW0V5023",
        "period": "2022-12-31"
      },
      "decimals": -6,
      "value": 798000000.0
    },
    "f1": {
      "dimensions": {
        "unit": "iso4217:EUR",
        "concept": "CashAndCashEquivalents",
        "entity": "549300JSX0Z4CW0V5023"
```

2. populate

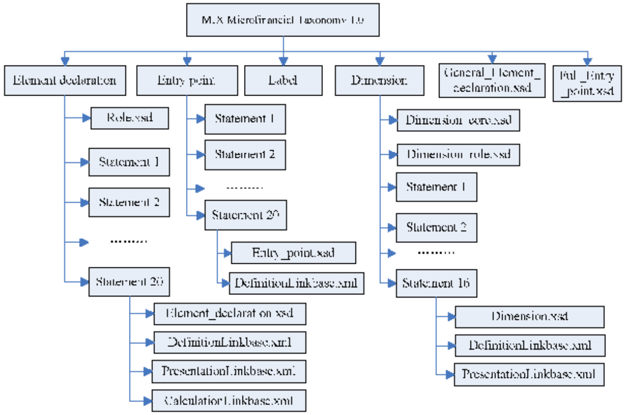
Knowledge Graph

neo4j

No ESG data available yet – we used sample data to populate the Knowledge Graph

2. ONTOLOGIES

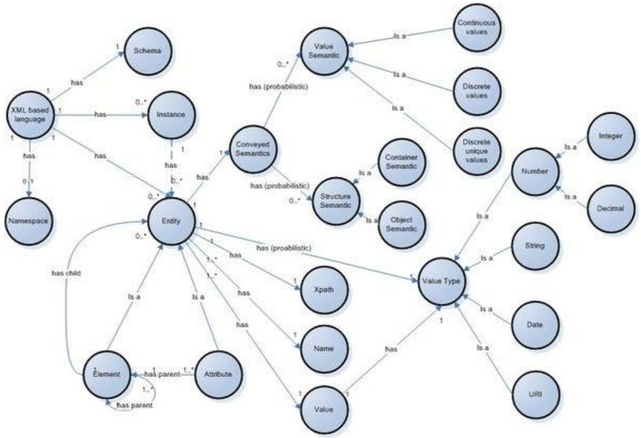
ESRS Taxonomy Schema



- Hierarchy
- Concrete

derive

Ontology / Knowledge Graph

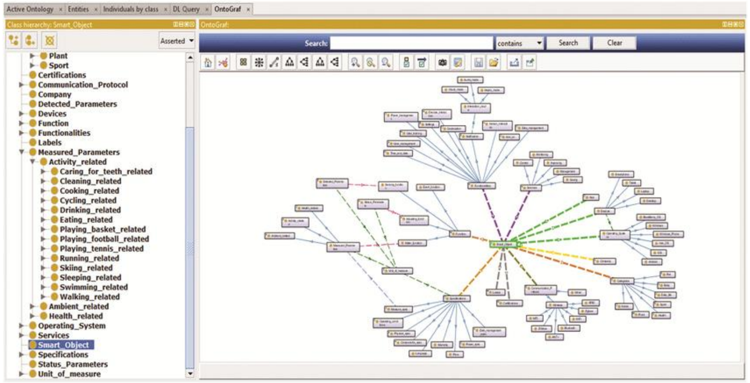


- Semantic Meaning
- Relationships

Derive Ontology based on ESRS Taxonomy

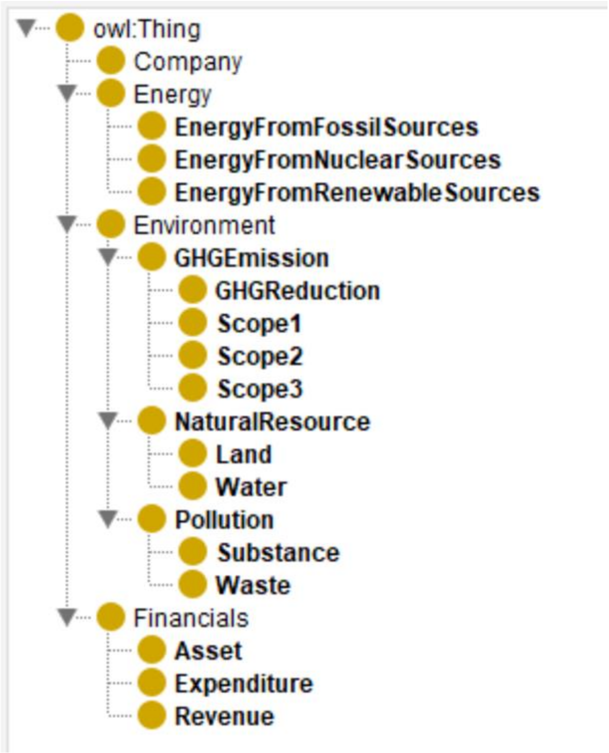
2. ONTOLOGIES

protégé



```
    Plant
    Sport
  Certifications
  Communication_Protocol
  Company
  Detected_Parameters
  Devices
  Function
  Functionalities
  Labels
  Measured_Parameters
    Activity_related
      Caring_for_teeth_related
      Cleaning_related
      Cooking_related
      Cycling_related
      Drinking_related
      Eating_related
      Playing_basket_related
      Playing_football_related
      Playing_tennis_related
      Running_related
      Skiing_related
      Sleeping_related
      Swimming_related
      Walking_related
    Ambient_related
    Health_related
  Operating_System
  Services
  Smart_Object
  Specifications
  Status_Parameters
  Unit_of_measure
```

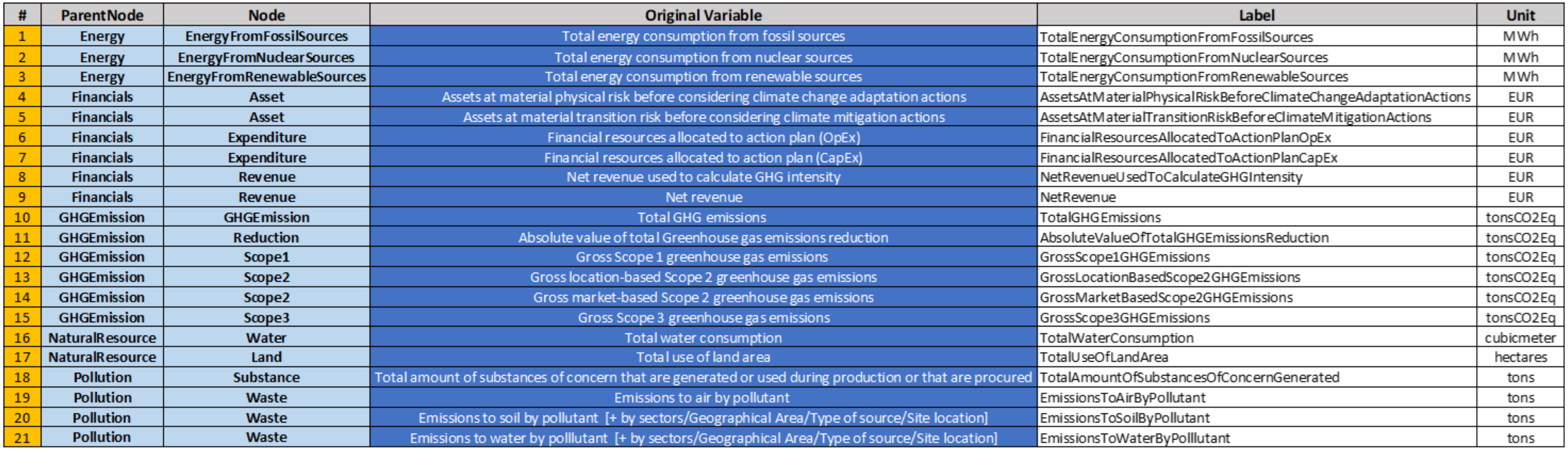
create



```
owl:Thing
  Company
  Energy
    EnergyFromFossilSources
    EnergyFromNuclearSources
    EnergyFromRenewableSources
  Environment
    GHGEmission
      GHGReduction
      Scope1
      Scope2
      Scope3
    NaturalResource
      Land
      Water
    Pollution
      Substance
      Waste
  Financials
    Asset
    Expenditure
    Revenue
```

Created ontology for 21 exemplary data points with 16 Node labels

2. ONTOLOGIES



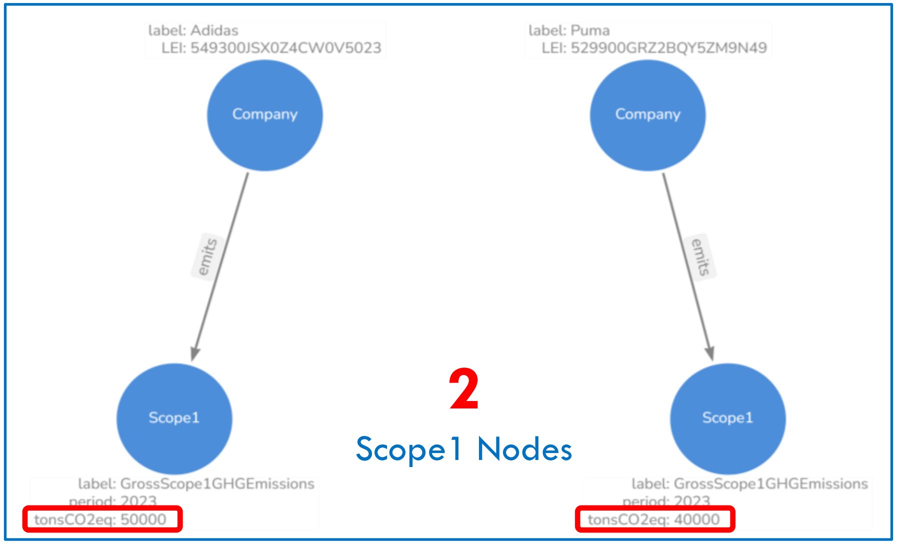
#	ParentNode	Node	Original Variable	Label	Unit
1	Energy	EnergyFromFossilSources	Total energy consumption from fossil sources	TotalEnergyConsumptionFromFossilSources	MWh
2	Energy	EnergyFromNuclearSources	Total energy consumption from nuclear sources	TotalEnergyConsumptionFromNuclearSources	MWh
3	Energy	EnergyFromRenewableSources	Total energy consumption from renewable sources	TotalEnergyConsumptionFromRenewableSources	MWh
4	Financials	Asset	Assets at material physical risk before considering climate change adaptation actions	AssetsAtMaterialPhysicalRiskBeforeClimateChangeAdaptationActions	EUR
5	Financials	Asset	Assets at material transition risk before considering climate mitigation actions	AssetsAtMaterialTransitionRiskBeforeClimateMitigationActions	EUR
6	Financials	Expenditure	Financial resources allocated to action plan (OpEx)	FinancialResourcesAllocatedToActionPlanOpEx	EUR
7	Financials	Expenditure	Financial resources allocated to action plan (CapEx)	FinancialResourcesAllocatedToActionPlanCapEx	EUR
8	Financials	Revenue	Net revenue used to calculate GHG intensity	NetRevenueUsedToCalculateGHGIntensity	EUR
9	Financials	Revenue	Net revenue	NetRevenue	EUR
10	GHGEmission	GHGEmission	Total GHG emissions	TotalGHGEmissions	tonsCO2Eq
11	GHGEmission	Reduction	Absolute value of total Greenhouse gas emissions reduction	AbsoluteValueOfTotalGHGEmissionsReduction	tonsCO2Eq
12	GHGEmission	Scope1	Gross Scope 1 greenhouse gas emissions	GrossScope1GHGEmissions	tonsCO2Eq
13	GHGEmission	Scope2	Gross location-based Scope 2 greenhouse gas emissions	GrossLocationBasedScope2GHGEmissions	tonsCO2Eq
14	GHGEmission	Scope2	Gross market-based Scope 2 greenhouse gas emissions	GrossMarketBasedScope2GHGEmissions	tonsCO2Eq
15	GHGEmission	Scope3	Gross Scope 3 greenhouse gas emissions	GrossScope3GHGEmissions	tonsCO2Eq
16	NaturalResource	Water	Total water consumption	TotalWaterConsumption	cubicmeter
17	NaturalResource	Land	Total use of land area	TotalUseOfLandArea	hectares
18	Pollution	Substance	Total amount of substances of concern that are generated or used during production or that are procured	TotalAmountOfSubstancesOfConcernGenerated	tons
19	Pollution	Waste	Emissions to air by pollutant	EmissionsToAirByPollutant	tons
20	Pollution	Waste	Emissions to soil by pollutant [+ by sectors/Geographical Area/Type of source/Site location]	EmissionsToSoilByPollutant	tons
21	Pollution	Waste	Emissions to water by polllutant [+ by sectors/Geographical Area/Type of source/Site location]	EmissionsToWaterByPolllutant	tons

16 NODE LABELS:
SUBORDINATE POSSIBLY ALL DATAPOINTS

DATA FROM ALL ESRS-AREAS:
ESRS_E1: **Climate Change**
ESRS_E2: **Pollution**
ESRS_E3: **Water and Marine Resources**
ESRS_E4: **Biodiversity and Ecosystems**
ESRS_E5: **Resource Use and Circular Economy**

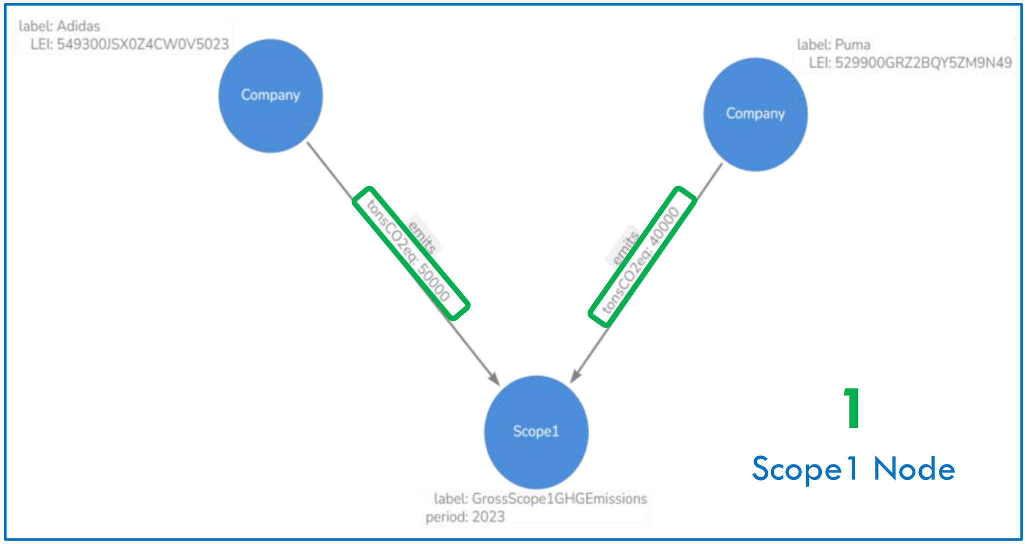
16 Node labels capable to subordinate ALL (>21) data points. Data from all E-Areas (E1-E5).

2. ONTOLOGIES



2

Scope1 Nodes



1

Scope1 Node

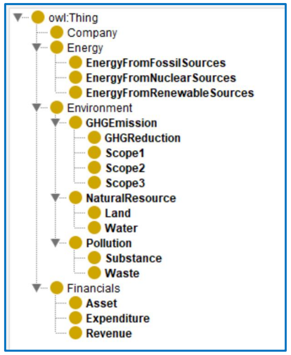
✗

✓

Data point values stored in RELATIONSHIP, not in NODE. Saves Nodes and memory.

3. KNOWLEDGE GRAPH CONSTRUCTION

Ontology



- owl:Thing
  - Company
  - Energy
    - EnergyFromFossilSources
    - EnergyFromNuclearSources
    - EnergyFromRenewableSources
  - Environment
    - GHGEmission
      - GHGReduction
      - Scope1
      - Scope2
      - Scope3
    - NaturalResource
      - Land
      - Water
    - Pollution
      - Substance
      - Waste
  - Financials
    - Asset
    - Expenditure
    - Revenue

+

unique_node_keys

```
{"Company": ["LEI"],
 "Waste": ["period", "label"],
 "Substance": ["period", "label"],
 "EnergyFromNuclearSources": ["period", "label"],
 "EnergyFromFossilSources": ["period", "label"],
 "EnergyFromRenewableSources": ["period", "label"],
 "GHGEmission": ["period", "label"],
 "Scope1": ["period", "label"],
 "Scope2": ["period", "label"],
 "Scope3": ["period", "label"],
 "GHGReduction": ["period", "label"],
 "Land": ["period", "label"],
 "Water": ["period", "label"],
 "Asset": ["period", "label"],
 "Expenditure": ["period", "label"],
 "Revenue": ["period", "label"],
 }
```

+

node_val_props

```
{"Waste": "tons",
 "Substance": "tons",
 "EnergyFromNuclearSources": "MWh",
 "EnergyFromFossilSources": "MWh",
 "EnergyFromRenewableSources": "MWh",
 "GHGEmission": "tonsCO2Eq",
 "Scope1": "tonsCO2Eq",
 "Scope2": "tonsCO2Eq",
 "Scope3": "tonsCO2Eq",
 "GHGReduction": "tonsCO2Eq",
 "Land": "hectares",
 "Water": "cubicmetres",
 "Asset": "EUR",
 "Revenue": "EUR",
 "Expenditure": "EUR"
}
```

1

node_query_templates

```
WITH $node_data AS node_data
MERGE (n:Asset { period: node_data['Asset']['period'],
        label: node_data['Asset']['label'] } )
RETURN count(*) as total
```

relation_query_templates

```
WITH $rel_data AS rel_data
MATCH (source:Company { LEI: rel_data["Company_disposes_Waste"]["source"]["Company"]["LEI"] } )
MATCH (target:Waste { period: rel_data["Company_disposes_Waste"]["target"]["Waste"]["period"],
        label: rel_data["Company_disposes_Waste"]["target"]["Waste"]["label"] } )
MERGE (source)-[r:disposes { tons: rel_data["Company_disposes_Waste"]["target"]["Waste"]["tons"] }]->(target)
RETURN count(*) as total
```

constraint_query_templates

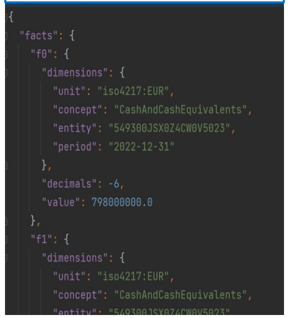
```
CREATE CONSTRAINT constraint_for_node_Waste IF NOT EXISTS
FOR (node:Waste) REQUIRE (node.period, node.label) IS UNIQUE;
```

namespace_query_templates

```
CALL n10s.nsprefixes.add('nso','http://www.semanticweb.org/rainergo/ontologies/2023/10/Ontology4#');
CALL n10s.mapping.add('http://www.semanticweb.org/rainergo/ontologies/2023/10/Ontology4#Scope2', 'Scope2');
CALL n10s.mapping.add('http://www.semanticweb.org/rainergo/ontologies/2023/10/Ontology4#hectares', 'hectares');
CALL n10s.mapping.add('http://www.semanticweb.org/rainergo/ontologies/2023/10/Ontology4#Scope3', 'Scope3');
CALL n10s.mapping.add('http://www.semanticweb.org/rainergo/ontologies/2023/10/Ontology4#Company', 'Company');
CALL n10s.mapping.add('http://www.semanticweb.org/rainergo/ontologies/2023/10/Ontology4#Waste', 'Waste');
CALL n10s.mapping.add('http://www.semanticweb.org/rainergo/ontologies/2023/10/Ontology4#spends', 'spends');
CALL n10s.mapping.add('http://www.semanticweb.org/rainergo/ontologies/2023/10/Ontology4#label', 'label');
```

2

JSON-Data



```
{
  "facts": {
    "f0": {
      "dimensions": {
        "unit": "iso4217:EUR",
        "concept": "CashAndCashEquivalents",
        "entity": "549300JSX0Z4CW0V5023",
        "period": "2022-12-31"
      },
      "decimals": -6,
      "value": 798000000.0
    },
    "f1": {
      "dimensions": {
        "unit": "iso4217:EUR",
        "concept": "CashAndCashEquivalents",
        "entity": "549300JSX0Z4CW0V5023"
```

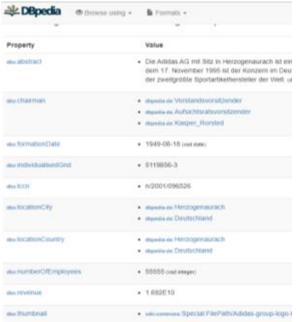
3

neo4j

Ontology and inputs create query-templates that are used to import the JSON-data into NEO4J

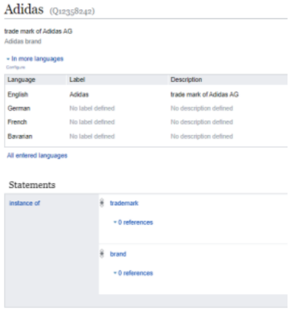
4. ADD DATA TO KNOWLEDGE GRAPH

DBpedia



Property	Value
dbo:abstract	• Die Adidas AG mit Sitz in Herzogenaurach ist ein dem 17. November 1995 ist der Konzern im Deu der zweitgrößte Sportartikelhersteller der Welt. u
dbo:chairman	• dbpedia-de:Vorstandsvorsitzender • dbpedia-de:Aufsichtsratsvorsitzender • dbpedia-de:Kasper_Rorsted
dbo:formationDate	• 1949-08-18 (xsd:date)
dbo:individualisedGnd	• 5119856-3
dbo:lccn	• n/2001/096526
dbo:locationCity	• dbpedia-de:Herzogenaurach • dbpedia-de:Deutschland
dbo:locationCountry	• dbpedia-de:Herzogenaurach • dbpedia-de:Deutschland
dbo:numberOfEmployees	• 55555 (xsd:integer)
dbo:revenue	• 1.692E10
dbo:thumbnail	• wiki-commons:Special:FilePath/Adidas-group-logo-

WIKIDATA



Adidas (Q12358242)

trade mark of Adidas AG

Adidas brand

▾ In more languages

Configure

Language	Label	Description
English	Adidas	trade mark of Adidas AG
German	No label defined	No description defined
French	No label defined	No description defined
Bavarian	No label defined	No description defined

All entered languages

Statements

instance of	trademark ▾ 0 references brand ▾ 0 references

DBpedia:
- abstract
("Company description")

Wikidata:
- industries
- country
- ISIN

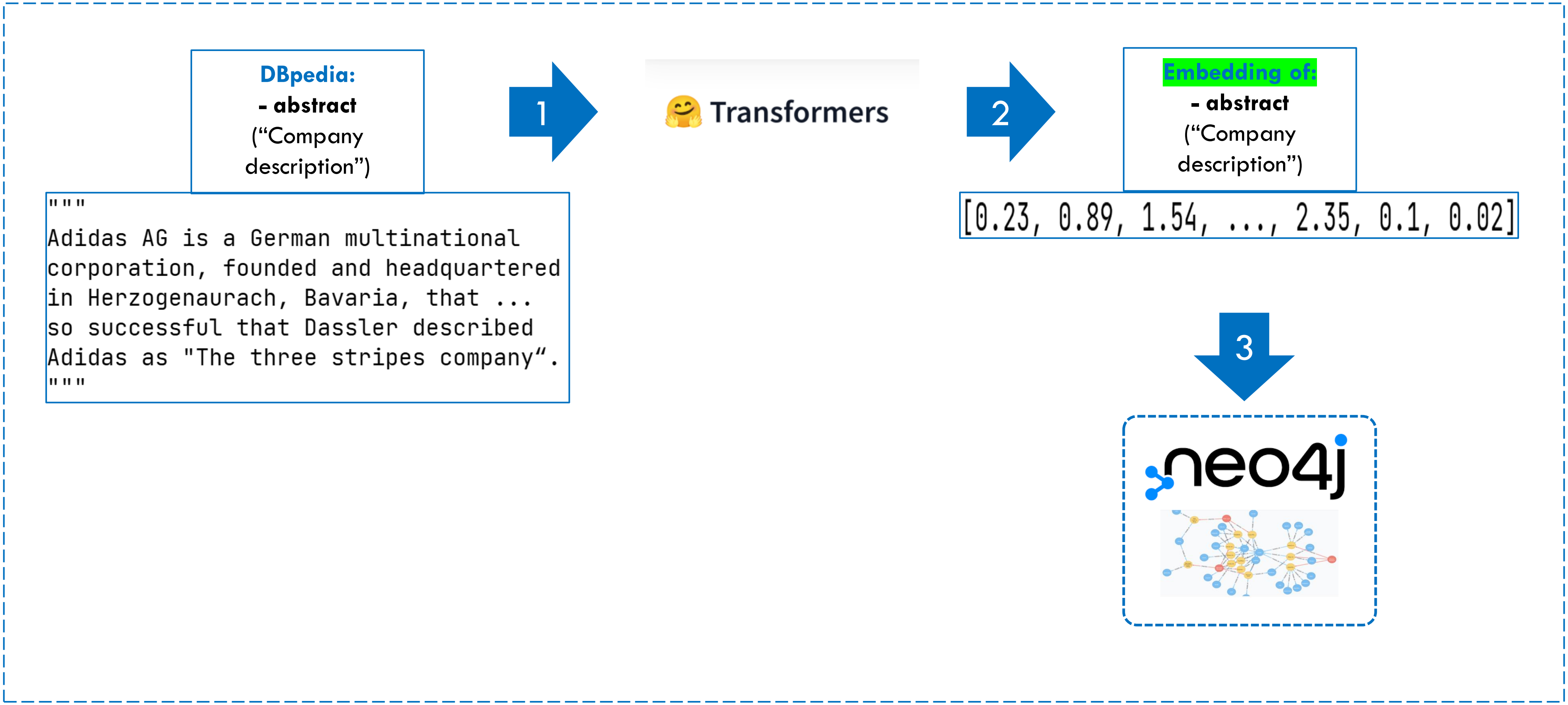
SPARQL

requests

neo4j

External data from Wikidata and DBpedia loaded into NEO4J via SPARQL requests over internet

4. ADD DATA TO KNOWLEDGE GRAPH



DBpedia:
- abstract
("Company
description")

```
"""
Adidas AG is a German multinational
corporation, founded and headquartered
in Herzogenaurach, Bavaria, that ...
so successful that Dassler described
Adidas as "The three stripes company“.
"""
```

Embedding of:
- abstract
("Company
description")

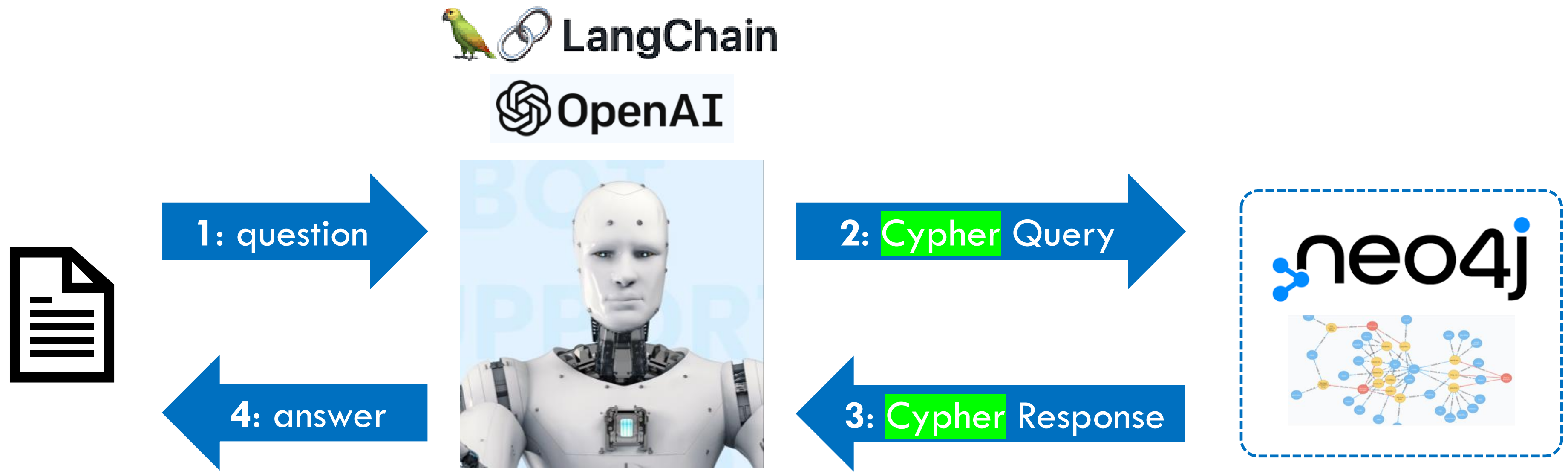
```
[0.23, 0.89, 1.54, ..., 2.35, 0.1, 0.02]
```

Add embedding of text to Knowledge Graph Node ("Company") as property to later do similarity search

LIVE DEMO 1:

Create Knowledge Graph with data in NEO4J

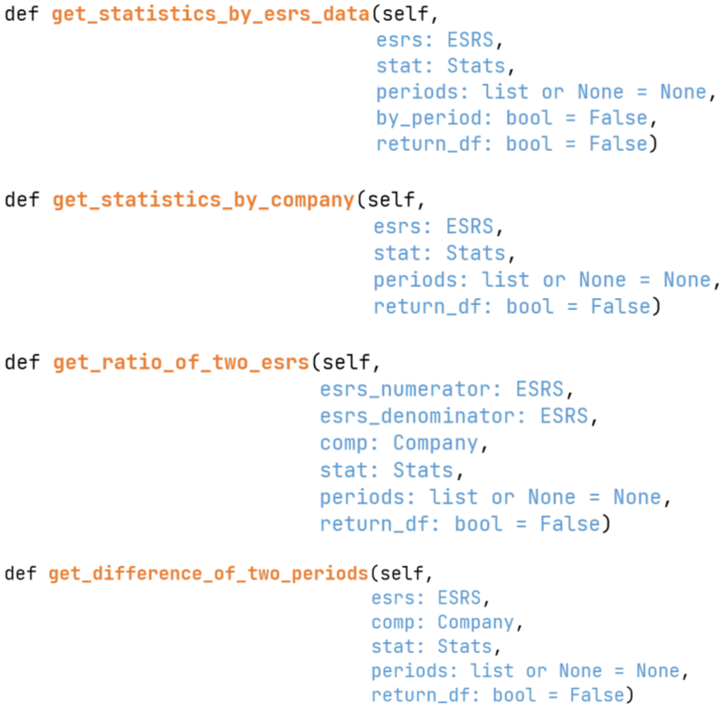
5. KNOWLEDGE GRAPH BOT



Send text question to Knowledge Graph Bot to receive text answer

6. KNOWLEDGE GRAPH QUERIES

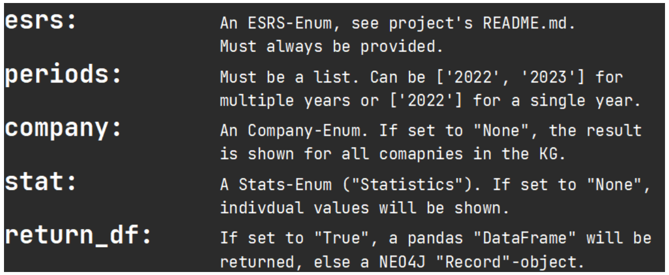
6 Parameterizable Functions:



```
def get_statistics_by_esrs_data(self,
                                esrs: ESRS,
                                stat: Stats,
                                periods: list or None = None,
                                by_period: bool = False,
                                return_df: bool = False)

def get_statistics_by_company(self,
                              esrs: ESRS,
                              stat: Stats,
                              periods: list or None = None,
                              return_df: bool = False)

def get_ratio_of_two_esrs(self,
                          esrs_numerator: ESRS,
                          esrs_denominator: ESRS,
                          comp: Company,
                          stat: Stats,
                          periods: list or None = None,
                          return_df: bool = False)

def get_difference_of_two_periods(self,
                                  esrs: ESRS,
                                  comp: Company,
                                  stat: Stats,
                                  periods: list or None = None,
                                  return_df: bool = False)
```

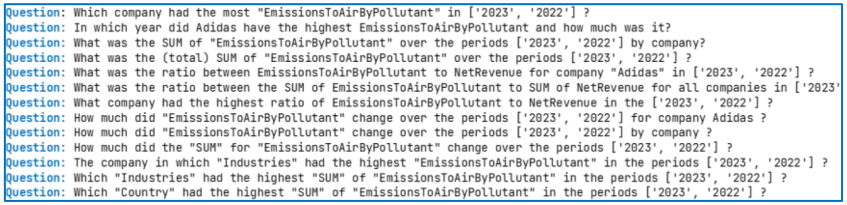
1

Parameters:



```
esrs:         An ESRS-Enum, see project's README.md.
              Must always be provided.
periods:      Must be a list. Can be ['2022', '2023'] for
              multiple years or ['2022'] for a single year.
company:      An Company-Enum. If set to "None", the result
              is shown for all comapnies in the KG.
stat:         A Stats-Enum ("Statistics"). If set to "None",
              indivdual values will be shown.
return_df:    If set to "True", a pandas "DataFrame" will be
              returned, else a NEO4J "Record"-object.
```

2



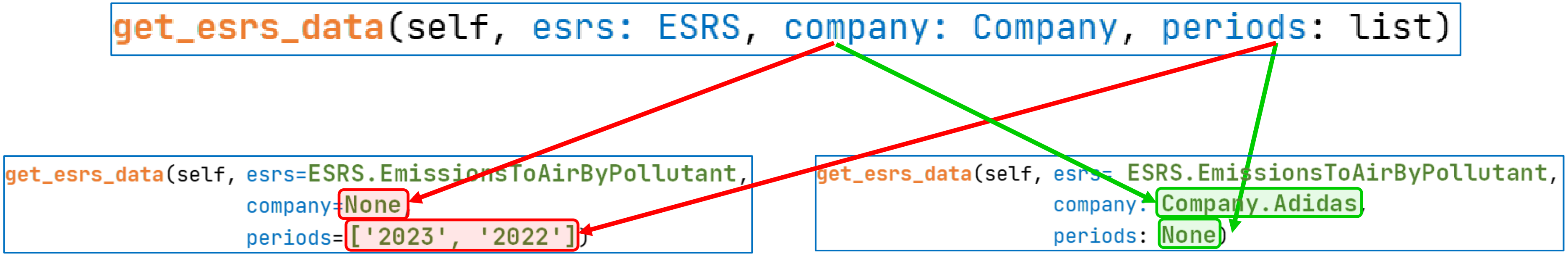
```
Question: Which company had the most "EmissionsToAirByPollutant" in ['2023', '2022'] ?
Question: In which year did Adidas have the highest EmissionsToAirByPollutant and how much was it?
Question: What was the SUM of "EmissionsToAirByPollutant" over the periods ['2023', '2022'] by company?
Question: What was the (total) SUM of "EmissionsToAirByPollutant" over the periods ['2023', '2022'] ?
Question: What was the ratio between EmissionsToAirByPollutant to NetRevenue for company "Adidas" in ['2023', '2022'] ?
Question: What was the ratio between the SUM of EmissionsToAirByPollutant to SUM of NetRevenue for all companies in ['2023'
Question: What company had the highest ratio of EmissionsToAirByPollutant to NetRevenue in the ['2023', '2022'] ?
Question: How much did "EmissionsToAirByPollutant" change over the periods ['2023', '2022'] for company Adidas ?
Question: How much did "EmissionsToAirByPollutant" change over the periods ['2023', '2022'] by company ?
Question: How much did the "SUM" for "EmissionsToAirByPollutant" change over the periods ['2023', '2022'] ?
Question: The company in which "Industries" had the highest "EmissionsToAirByPollutant" in the periods ['2023', '2022'] ?
Question: Which "Industries" had the highest "SUM" of "EmissionsToAirByPollutant" in the periods ['2023', '2022'] ?
Question: Which "Country" had the highest "SUM" of "EmissionsToAirByPollutant" in the periods ['2023', '2022'] ?
```

Parameterized functions allow for a wide range of different questions.

6. KNOWLEDGE GRAPH QUERIES

Example:

Parameterizable Function:



```
get_esrs_data(self, esrs: ESRS, company: Company, periods: list)
```

```
get_esrs_data(self, esrs=ESRS.EmissionsToAirByPollutant,
                    company=None
                    periods=['2023', '2022'])
```

```
get_esrs_data(self, esrs= ESRS.EmissionsToAirByPollutant,
                    company: Company.Adidas,
                    periods: None)
```

QUESTION:

"Which company had the most EmissionsToAirByPollutant in ['2023', '2022'] ?"

QUESTION:

"In which year did Adidas have the highest EmissionsToAirByPollutant and how much was it ?"

Change in parameters will change question, answer and answer components.

LIVE DEMO 2:

Query Knowledge Graph

Thank you
for your attention !