BlocK-DaQ – Getting Started

# Use DSD Connectors to Connect to Data Sources

Currently, there are implementations for

* CSV files
* Cassandra DBs
* DSD files
* MySQL DBs
* Oracle DBs (still need to be verified and comprehensively tested)

To each of those data sources, it is possible to create a connection using a DSConnector (only accessing the schema, e.g., sufficient for DSD files), or DSInstanceConnector (for accessing schema + instances of a data source).

ConnectorMySQL connNW = ConnectorMySQL.getInstance("jdbc:mysql://localhost:port",

"dbname", "user", "pw");

DSInstanceConnector connPersons = new ConnectorCSV("path/persons.csv", ",",

"\n", "Persons");

In order to transform the schema description of the local data source to virtual DSD elements in the Java runtime environment, it is necessary to load the schema information.

Datasource dsNW = connNW.loadSchema();

# Creation of a Reference Data Profile

To create and annotate a reference data profile to a DSD element, a corresponding method .annotateProfile(RecordSet) is provided, which needs to be called for each DSD element that should be annotated. In the following example, each attribute of each concept in the Northwind (NW) is annotated with a data profile that uses the first 5,000 records of the respective concept (table).

for (Concept c : dsNW.getConceptsAndAssociations()) {

RecordSet rs = conn.getPartialRecordSet(c, 0, 5000);

for (Attribute a : c.getSortedAttributes()) {

a.annotateProfile(rs);

}

}

# Persist Knowledge Graph to GraphDB

The entire knowledge graph (consisting of DSD elements and their annotated reference data profiles) can be persisted to GraphDB for continuous monitoring and retrieval after the Java runtime environment has been shut down.

for (Concept c : dsNW.getConceptsAndAssociations()) {

RecordSet rs = conn.getPartialRecordSet(c, 0, 5000);

for (Attribute a : c.getSortedAttributes()) {

a.annotateProfile(rs);

}

}

# Continuously Monitor the Data Quality