Demonstrator 19: Materials Databases Integration using the Materials Design Ontology

Huanyu Li, Rickard Armiento, Olaf Hartig, Mina Abd Nikooie Pour, Ying Li, Patrick Lambrix

Linköping University
Sweden
Demonstrator 19: Materials Databases Integration

Agenda

- Background and Motivation
- Materials Design Ontology (MDO)
- Ontology-driven Data Access and Data Integration
- Summary and Ongoing Work
Background and Motivation
Many software programs can achieve materials calculations

A lot of databases provide materials calculation data via Web APIs (e.g., REST APIs)
- Materials databases are heterogenous in nature
- APIs follow different data schemas

A data-driven workflow of materials design will search these databases with desired combinations of properties
Searching ‘GaN’ in Materials Project, OQMD, NOMAD
• Different number of fields returned, different terminology for the same concept
• Some semantics could be added

OPTIMADE (Open Databases Integration for Materials Design)
• To design a common API
• To make materials databases interoperable

Ontologies can help in,
• accessing/integrating data with semantics-aware
• standardizing terminologies
• making data FAIR

...
Motivation

- A semantics-aware and integrated way for querying data among heterogeneous data sources
  - Ontology-driven approaches are needed
  - Ontologies for representing materials design domain knowledge are needed

- There is a lack of methods accessing and integrating data over multiple heterogeneous data sources where data is shared via different ways
  - e.g., tabular data/SQL queries, JSON-formatted data/API requests
  - Classical ontology-based data access and integration (OBDA/OBDI) methods focus on relational data
Materials Design Ontology (MDO)
NeOn methodology in ontology engineering
  • Requirements analysis (Use Cases, Competency Questions, Additional Restrictions)
  • Reusing concepts from existing ontologies
    o PROV-O (PROVenance Ontology), QUDT (Quantities, Units, Dimensions, and Type Ontology)

Modular Design
  • Core Module, Structure Module, Calculation Module, Provenance Module

Discussions with a domain expert

https://github.com/LiUSemWeb/Materials-Design-Ontology
Ontology-driven Data Access and Data Integration
GraphQL-based framework for data access and integration

- GraphQL Server Generation Process, arrows (i) and (ii)
  - Ontology-based GraphQL server generation (OBG-gen)
- GraphQL Query Answering Process, arrows (1)–(4)
- https://github.com/LiUSemWeb/OBG-gen
GraphQL

- **What is GraphQL?**
  - GraphQL is a conceptual framework for building Web APIs
  - GraphQL can work with existing APIs of a system
  - A GraphQL server contains [GraphQL schema](#) and [GraphQL resolver](#)
  - Clients use the GraphQL query language to make requests to a GraphQL server
• Get all the calculations where ID in a given list, and reduced chemical formula contains the chlorine (Cl) or oxygen (O) elements

```graphql
1 query My_First_Query{
2   CalculationList(filter: CalculationFilter): [Calculation]
3   _and: [
4     _or: [
5         ID: { _in: ["6332", "9088", "21331", "mp-561628", "mp-614919"] },
6         { hasOutputStructure: { hasComposition: { ReducedFormula: { _like: "%Cl%" } } } },
7         { hasOutputStructure: { hasComposition: { ReducedFormula: { _like: "%O%" } } } }
8     ]
9   ]
10 }
11 }
```

Summary and Ongoing work
Summary and Ongoing Work

- **Materials Design Ontology**
  - MDO is capable to represent basic domain knowledge
  - MDO can be used for mapping different materials databases’ schemas
  - MDO can be used for semantically enabling materials database search

- **A GraphQL-based framework for data access and integration**
  - with an application in the materials design domain

- **Investigating the compatibility of MDO and top-level ontologies (e.g., EMMO)**
  - aligning MDO-core with EMMO

- **Extending MDO with new concepts and relationships**
  - A new MDO-property module