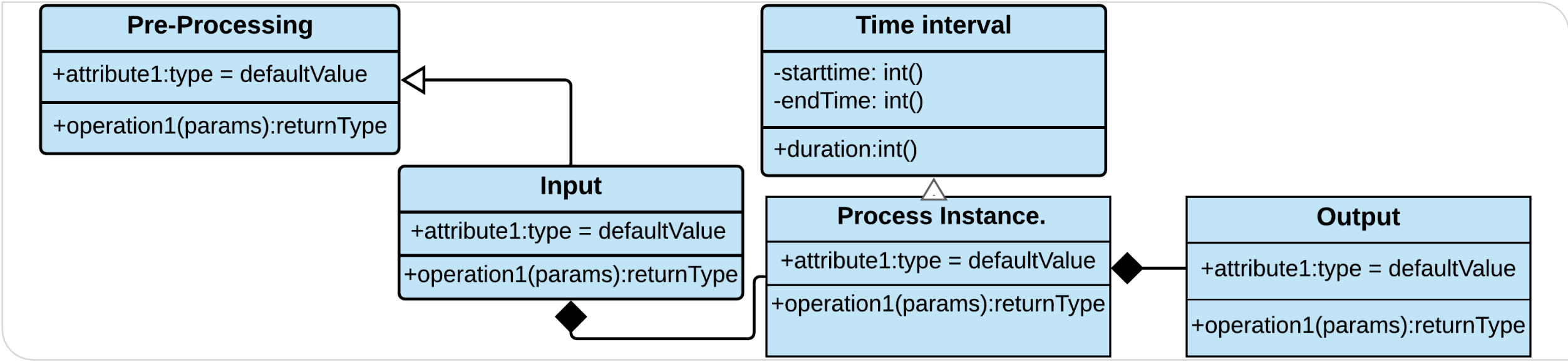
Karlsruher Institut für Technologie

Ontologies for MUSICODE

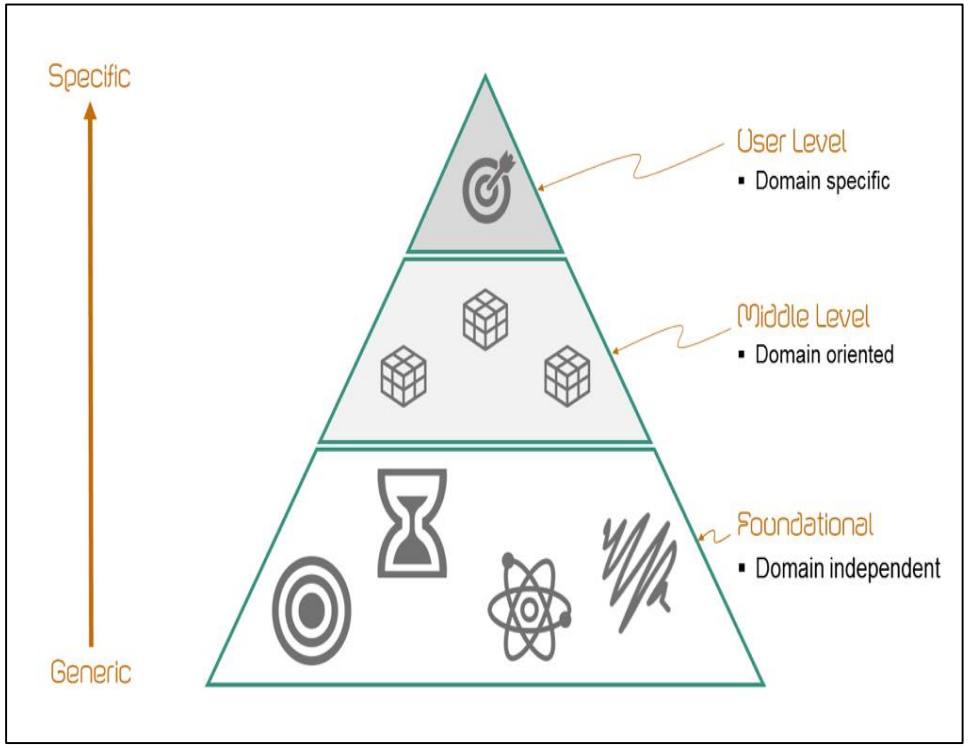
Hafiz Noman

(13-05-2022)



Introduction

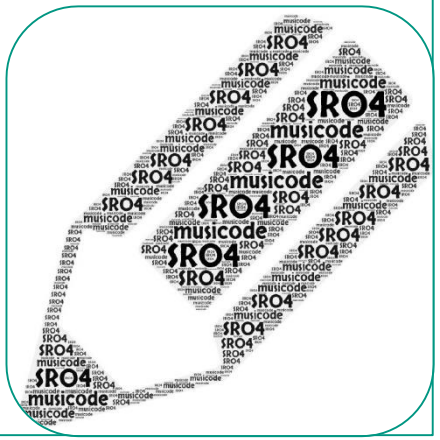
- Ontologies
  - An ontology is a formal description defining the organization of knowledge, intended as a set of concepts within a domain together with the relationships that hold between them. [1]
- Modelling approaches
  - Top down
    - Domain independent
  - Bottom-up
    - Start with describing the most specific application related to a domain
  - Middle-out
    - Model the most important classes/domain first



1. https://emmc.info/emmo-info/
2. https://tishchungoora.medium.com/ontology-mapping-infographic-621272b4d445

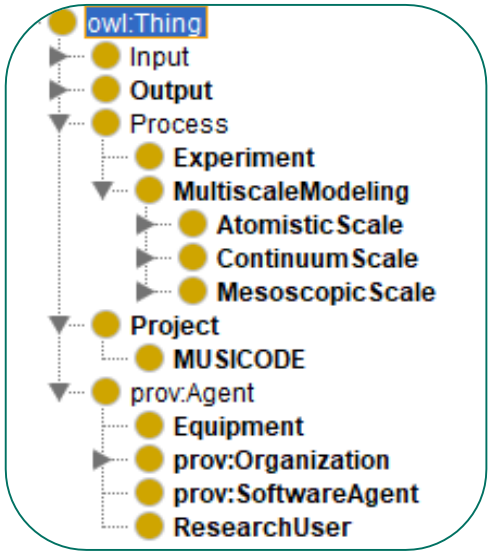
Simulation requirement ontology

- SRO
  - An ontology for describing the generation & organization of simulation process data within musicode project.
  - The SRO aims at reporting all the simulation activities, data generation (outputs), data items requirement (inputs) and organisation of simulation processes according to the length scale. i.e., micro, meso, and macro and the names of partners responsible for them.
- SRO reuses some elements from the following ontologies
  - PROV-O
  - MDMC-NEP-top-level-ontology



Important classes of SRO

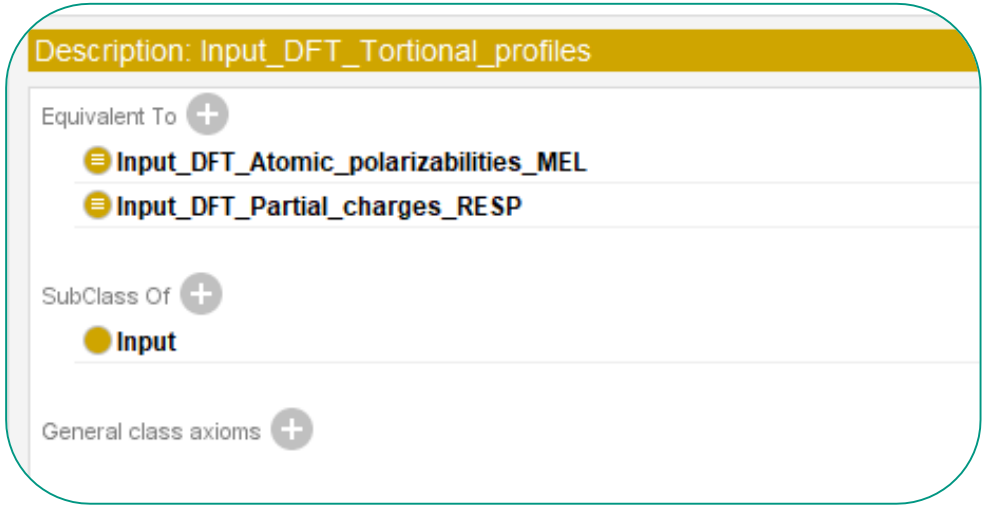
- The figure below shows the class hierarchy of the ontology
  - Input
    - The Input of any process without which a process cannot proceed
  - Process
    - Process, i.e., a physical entity with a temporal evolution that 'has a meaning for the ontologist'
  - Output
    - Data items/output provides by a process after completion
  - Project
    - An enterprise (potentially individual/typically collaborative), planned to achieve a particular aim
  - Agent
    - An agent is something that bears some form of responsibility for an activity taking place, for the existence of an entity, or another agent's activity.



Concept of equivalent classes axiom

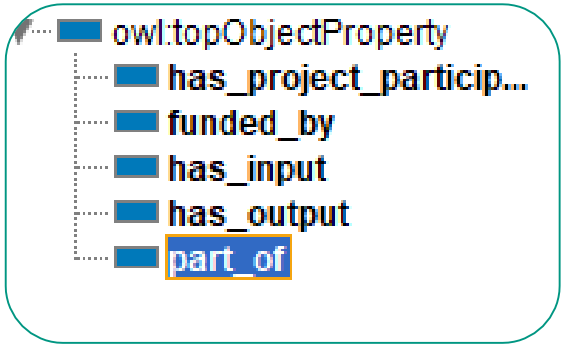
- Equivalent classes
  - It states that two classes in an ontology are semantically equivalent to each other. This axiom allows one to use a class as a synonym for each other class without affecting the meaning of the ontology.

- Following diagram displays an example of equivalent clases in the ontology.
  - Input of DFT Tortional profiles is same as of input of DFT atomic polarizabilities MEL and DFT Partial charges RESP



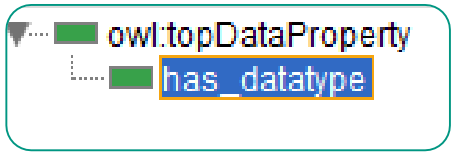
Object properties

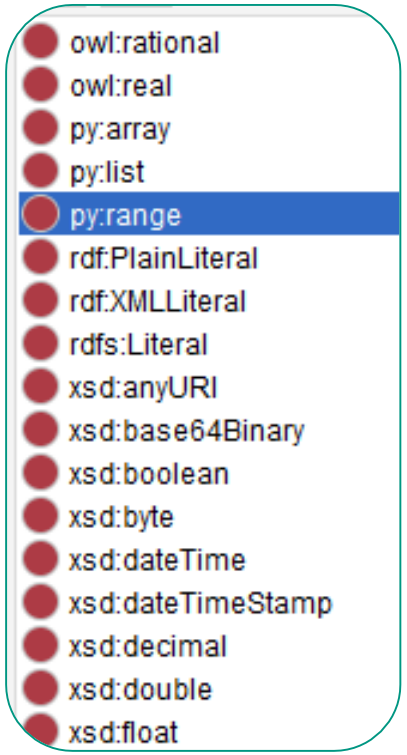
- The property that connects two classes/concepts.
- Following are some of the most important object properties present in the Ontology
  - has_project_participant
    - Is used to connect a process with its respective partner
  - funded_by
    - Is used to connect project (musicode) with EU (funding body)
  - has_input
    - Connects input and a process class
  - has_output
    - Connects a process class with its outcome
  - Part_of
    - Is used to connect a process class with its project class



Data properties

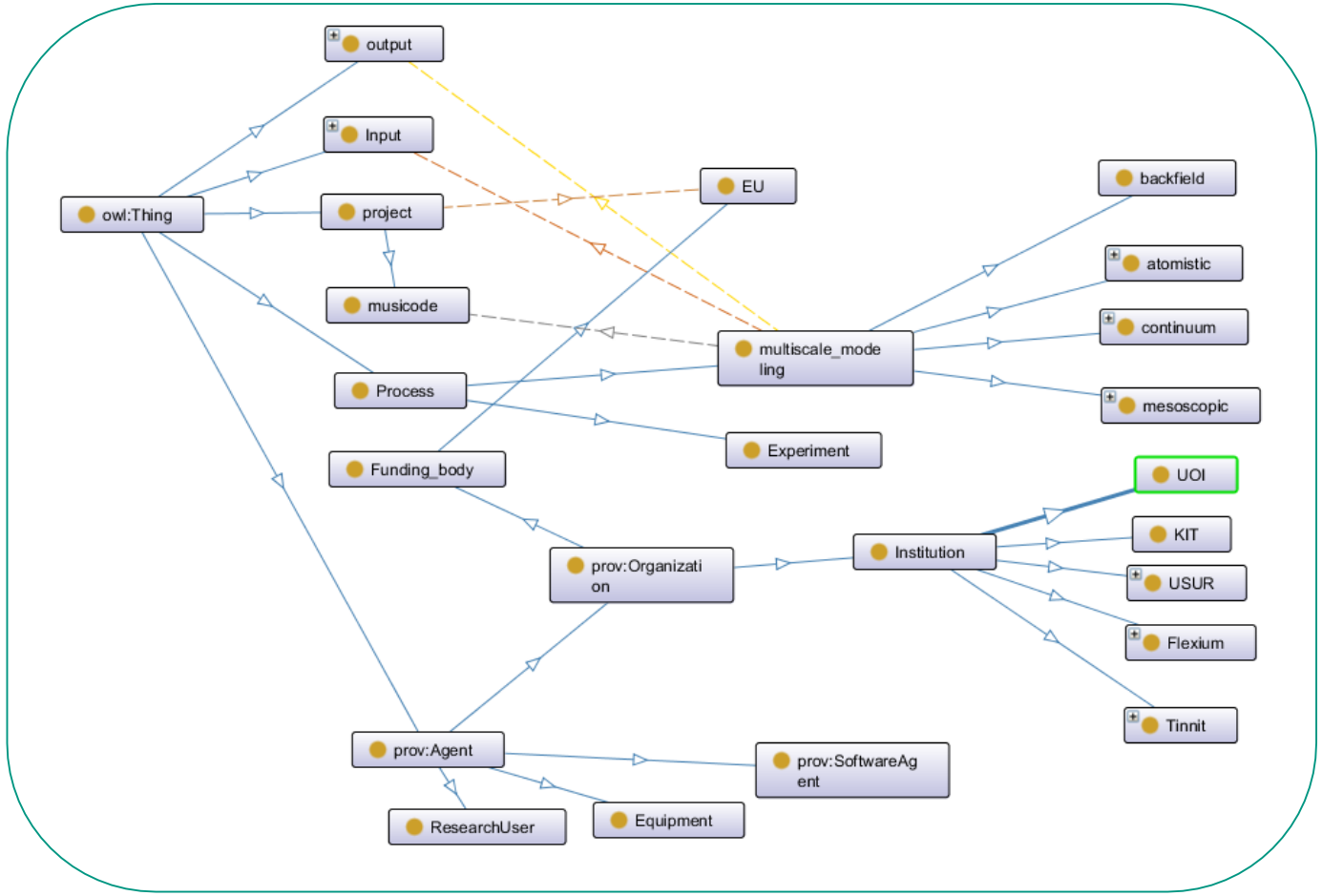
- Data propeties are used to connect individuals / instances of a class to a specific data type.
  - has_data_type





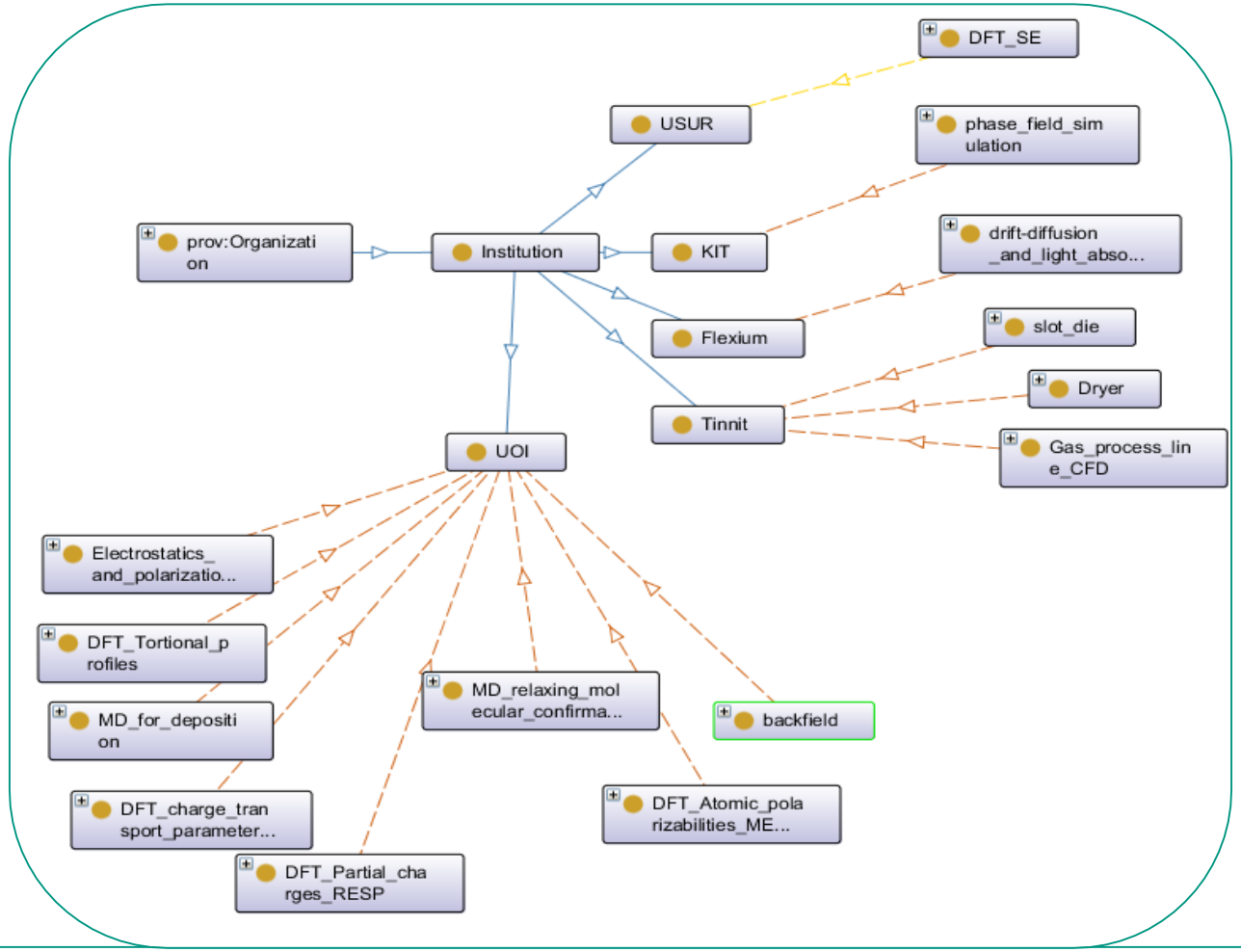
SRO (reduced view)

- General overview



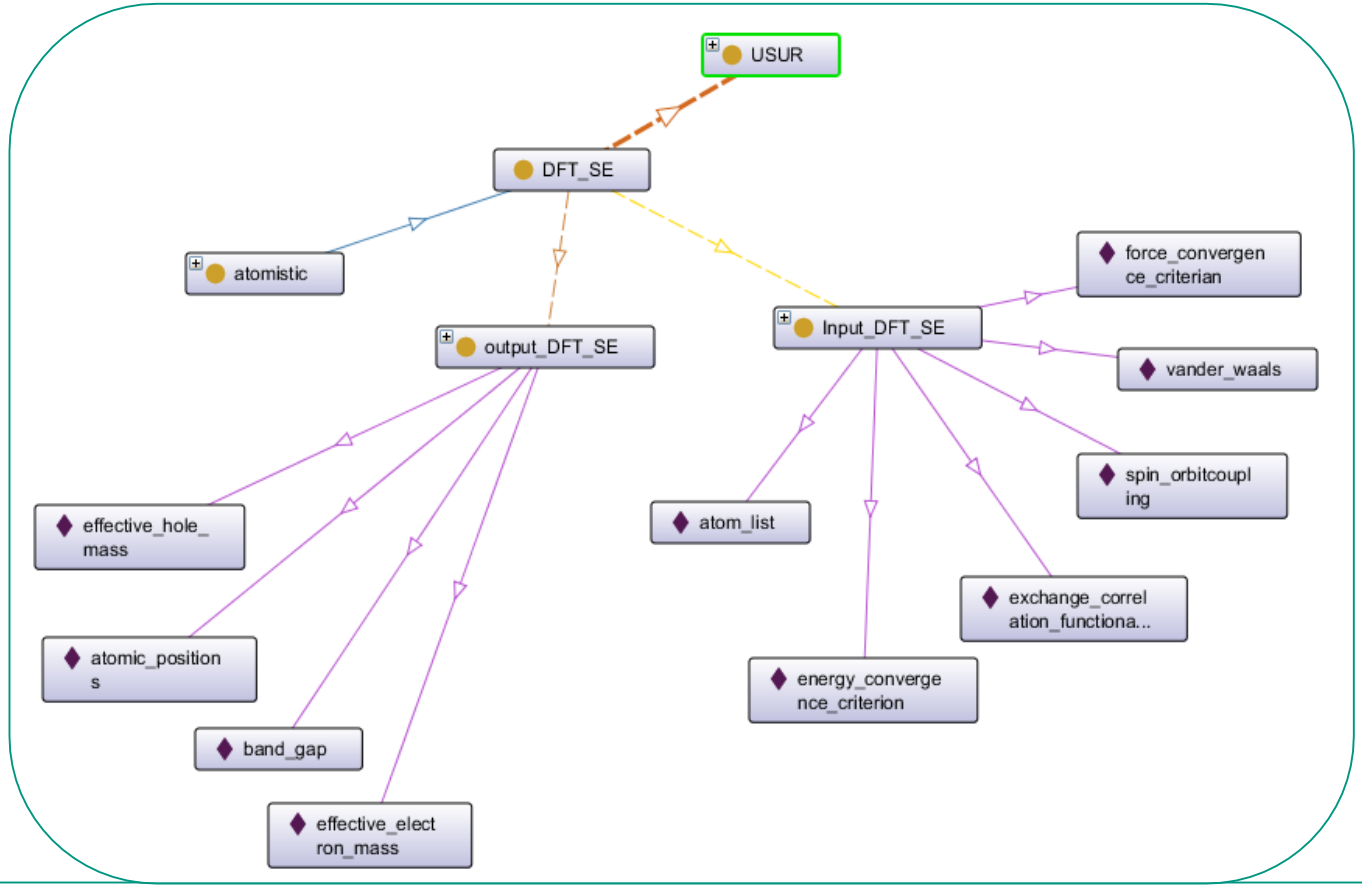
Participant view

- Participant view



Atomistic process view

- DFT for structural and electronic properties

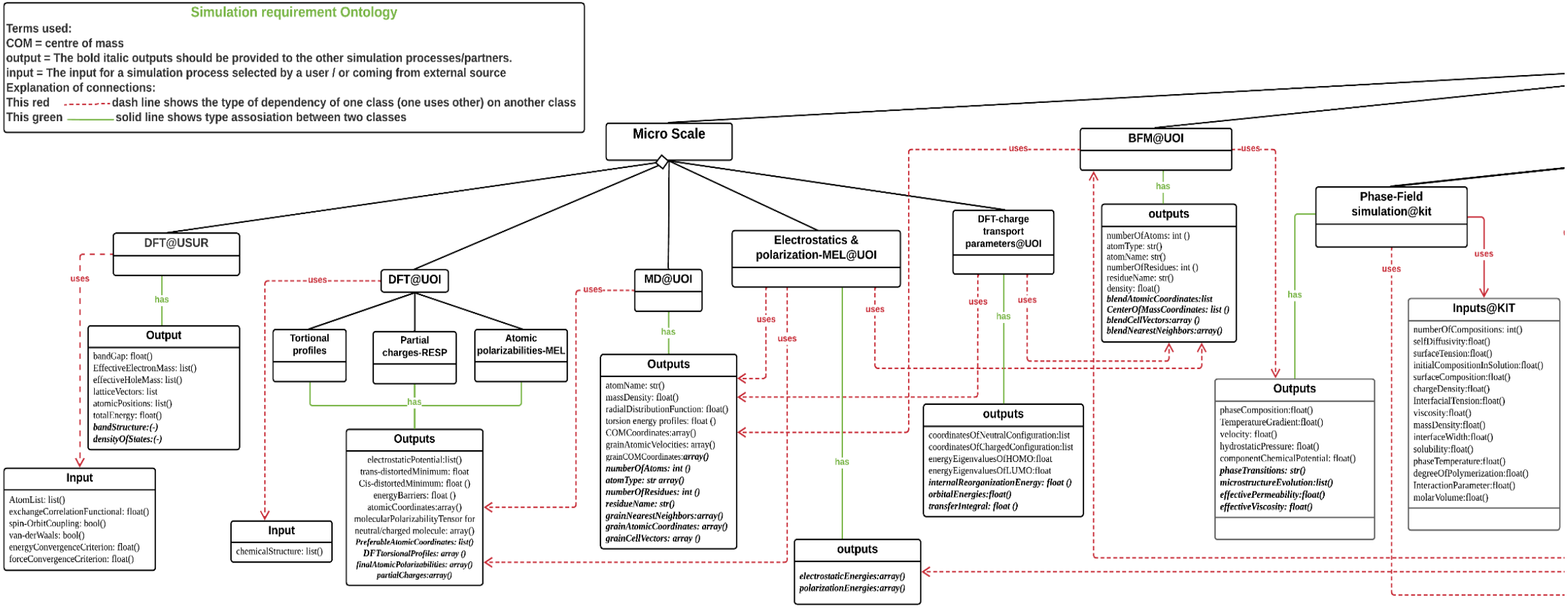


Description of SR-UML Ontology

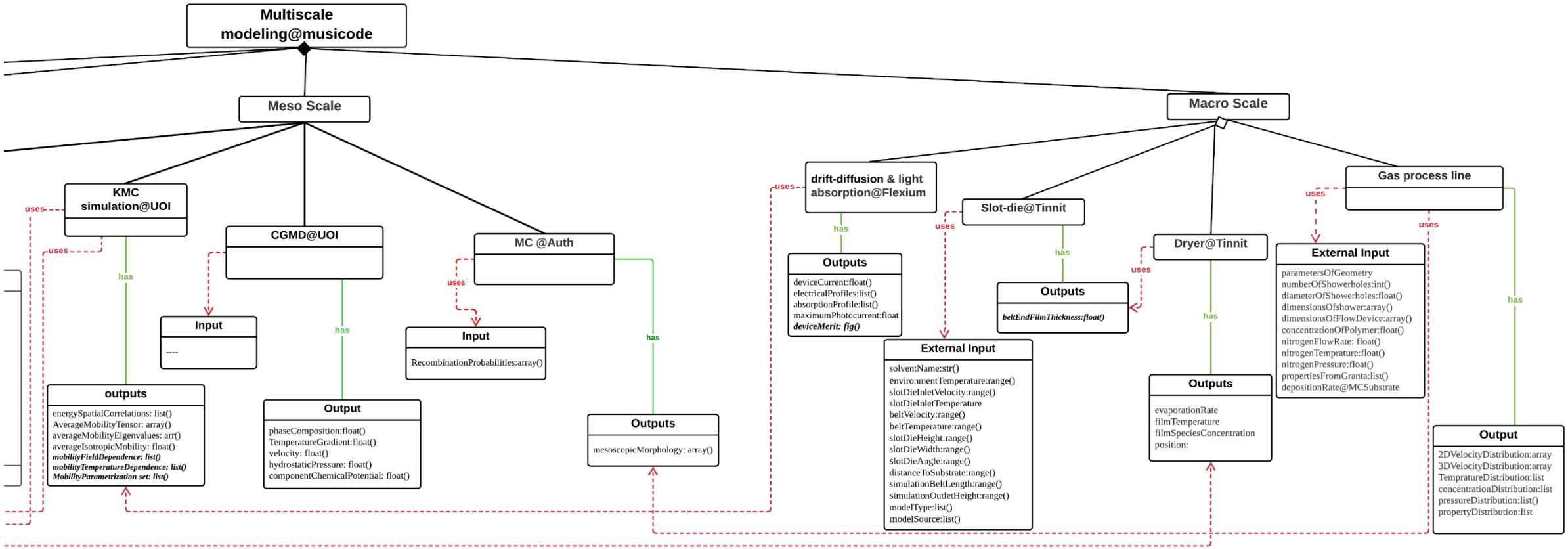
- Simulation requirement Ontology (in UML)
  - Inputs are classified as „external“ and „from a partner“ and Outputs are classified as „for a partner “ or „for the database“.
  - Two types of links have been used extensively.
  - The simulation process that uses the outputs of another process (as input) is joined by the „uses“ link and red dashed line.
  - The output of a simulation process is shown by the „has“ link and green solid line.
  - The name of the process is written at the top of each block before the „@“ sign.
  - After the „@“ sign the name of the partner who is performing the simulation is mentioned.
  - The outputs that should be transferred to the other methods are mentioned in bold italic letters.
  - The outputs that should be saved in the database are mentioned in normal letters.

The simulation requirment Ontology (I)

-



The simulation requirment Ontology (II)



The Ontology viewing link

- https://lucid.app/lucidchart/da44b35f-2534-4717-a51f-499df8ff08a4/edit?invitationId=inv_110ea034-f1bb-4ce7-b7a9-28b55c7992f6

- Click on the link
- Set up a free account
- View the Ontology
- If someone wants to edit the Ontology, he/she can request for it.

Karlsruher Institut für Technologie

Ontologies for MUSICODE

Presented by H.Noman
(13-05-2022)