Silvia Chiacchiera, Martin T. Horsch,
Michael A. Seaton, Ilian Todorov
**STFC Daresbury Laboratory**
UK Research and Innovation

# Towards an open simulation platform: Building a modelling/simulation Ontology from MODA

**IntOP2018**

$6^{th}$-$7^{th}$ November 2018

**IWM, Freiburg**

VIMMP
VIRTUAL MATERIALS MARKETPLACE

Science & Technology Facilities Council | UK Research and Innovation

VIMMP
VIRTUAL MATERIALS MARKETPLACE

# STFC/ URKI (Science and Technology Facilities Council, UK Research and Innovation)
# Scientific Computing Department, Daresbury Laboratory

- Software development / Research, with academia and industry.
- Lead of WP1 (Metadata, Interoperability design and standards) in VIMMP
- Development of general purpose simulation software packages, both at atomistic (DL_POLY, IT) and mesoscale (DL_MESO, MS) levels
- A variety of approaches:
  - Molecular Dynamics (MD), Dissipative Particle Dynamics (DPD), Lattice Boltzmann Equation (LBE)
  - In our Computational Chemistry group, also: hybrid approach of Quantum Mechanics - Molecular Mechanics (QM-MM), Monte Carlo (MC), Coarse Graining (CG), unified notation for various Force Fields
- A variety of application-related problems in solid state and soft matter: radiation damage (nuclear waste storage), physics of surfactants (notably, micellisation), polarisable mesoscale solvents (oil-water interfaces)

# The context of this talk

Within VIMMP (see talk by Welchy L. Cavalcanti and A. Hashibon) we are developing various ontologies for different purposes.

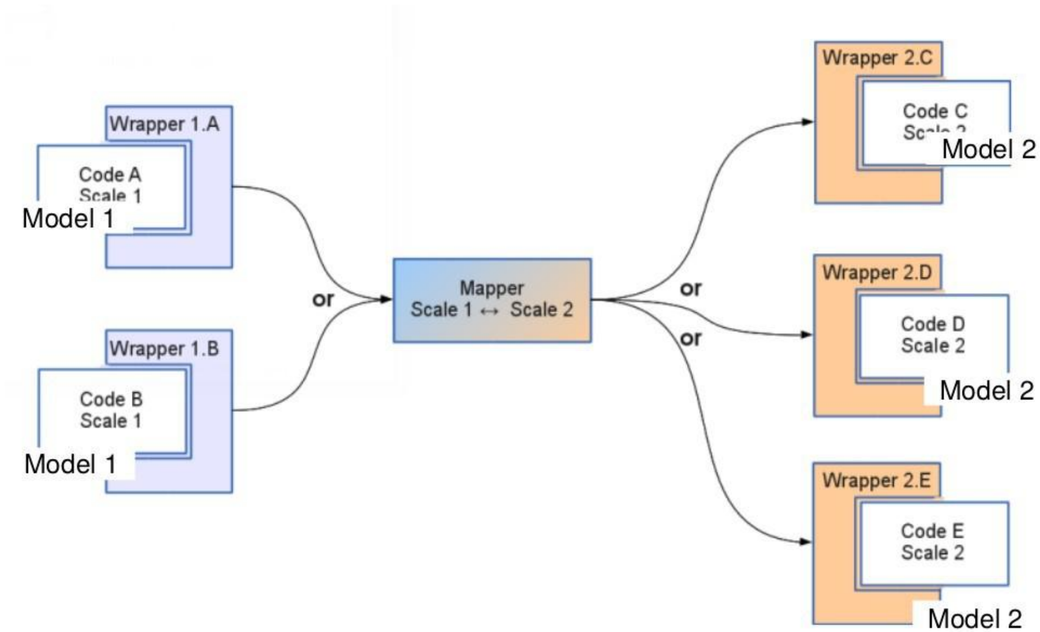
Below the upper level ontology (see talk by Martin Horsch and Georg Schmitz), there will be specific branches: some will deal with software, simulations and workflows.

To be used where and why?

- On the MP user interface: e.g, to browse software capabilities, parameter database coverage, request services
- Within the OSP:
  Interchangeability/interoperability of codes (at the same and at different scales) between them and with databases for model parameters
- General scientific benefit: documentation and reproducibility of results

MP: Marketplace, OSP: Open Simulation Platform

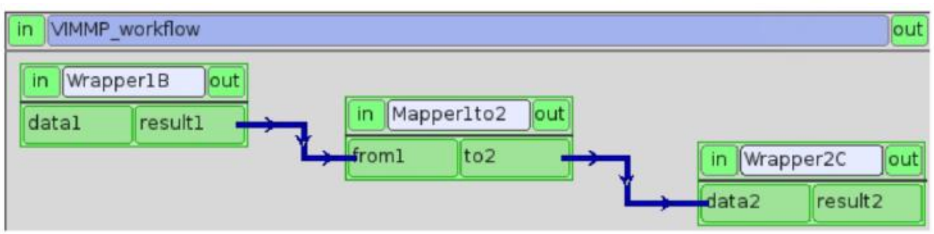
# The Open Simulation Platform concept




Codes interact thanks to wrappers (from code-specific to code-independent terms) and mappers (from one scale/model to the other).

Opennes: metadata are public, any code can be integrated

SALOME is an open-source platform developed by (among others) our partner EDF - Électricité de France
https://www.salome-platform.org/



# General ideas on common metadata and data formats

**Positive points:**

- In general: **value** in the long term, foster collaboration, facilitate **reproducibility**, **validation**, hybrid/adaptive schemes, **transparency** in methodology
- For developers: **wider audience**
- For users: **wider range of tools**, **unified set-ups**, ability to **compare and choose** between different software

**Difficulties:**

- Attaining agreement, **standards being actually adopted** and used. Codes are “**living**” entities, maintenance/funding, technology lifetime.
- For developers: **need to adapt** existing codes
- For users: may have commercial bindings, need of extra tools to work on marked-up and compressed data. **Higher barrier in writing own tools**.

# General ideas on common metadata and data formats

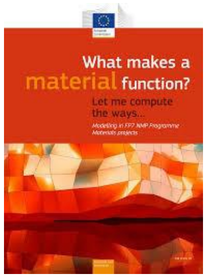
## Different needs across areas:

- The **traditional need** for interoperability and annotation/storage of data **varies a lot** across knowledge areas. (Unavoidable in big experimental facilities, not so much for computational research in various domains, including chemistry.)
- **Machine learning** and **data-driven** research techniques are changing this.

## Strategy:

- We proceed with small steps: start with simple cases
- **Openness**, contact with the different communities and developers
- Balance between logical structure and **actual implementations**
- Accessible **documentation** and **extensibility**
- Success is **driven by need,** commercial need may be key

# Related efforts




**ROMM**

*Reviews of Materials Modelling, What makes a material function? Let me compute the ways...*, Anne F. de Baas (ed), 6th version, 2017.

→ Classification of models and methods according to the used *entities* (electronic, atomistic, mesoscopic, continuum)

**EMMO**

*European Materials Modelling Ontology*, E. Ghedini et al, in development.

→ High-level philosophical framework for the fundamental concepts. Currently mostly focused on the material representation.

# Related efforts (2)

**NOMAD**

*Novel Materials Discovery*, M. Scheffler *et al*, 2015-2018,
https://nomad-coe.eu/

Aim: Repository of input and output (I/O) files of all important computational materials science codes

Model coverage: Molecular Dynamics (mostly ab initio)

Includes: Metadata schema (in JSON), I/O parsers (in Python)

Metadata repository: https://gitlab.mpcdf.mpg.de/nomad-lab/nomad-meta-info

At Nov 2018: 464 public metadata
(In 2017: ~200 public, ~2000 code-specific → 10% is code-independent)

# Related efforts (3)

**SimPhoNY**

*Simulation framework for multi-scale phenomena in micro- and nanosystems*, A. Hashibon *et al*, 2014-2017.

Aim: Facilitate the flow of information across sub-domains

Model coverage: across the four ROMM model types

Includes: CUBA/CUDS common vocabulary (in YAML) and HDF5-based file format (H5CUDS), code wrappers (in Python)

CUDS/CUBA: Common Universal/Unified Data Structure and Basic Attributes

Metadata repository: https://github.com/simphony/simphony-common/tree/master/ontology

# The MODA (Materials Modelling Data)

| Heading<br>Overview |
|---|
| (1) User Case<br>**(2) Model**<br>(3) Solver<br>(4) Processing |

MODA [1] is a **template** designed to describe (*a posteriori*) simulations in a unified framework. On the left, a sketch for the simplest case, an isolated simulation.

Heading: title and author. Overview: workflow rationale

(1) Formulates the physical problem (which of course can be dealt with different models, e.g., at different scales)

(2) Specifies a given model to describe the user case

(3) Specifies a numerical solver (method and tool)

(4) Describes the processing (can be on-the-fly or post) of the raw data (= the variables the equations are solved for, i.e., the minimal outcome)

[1] CEN Workshop Agreement (CWA) 17284, Materials modelling - Terminology, classification and metadata, April 2018.

CEN: European Committee for Standardization

# The MODA on the Market Place

| Heading Overview |
| --- |
| (1) User Case<br>(2) Model<br>(3) Solver<br>(4) Processing |

There are **multiple uses** of this template within the MP:

- Section (1) alone can be used by an end-user of the market place to formulate his problem, and request for translation (into modelling) and solution.

  [Note: Involves material, material properties, perspective, required features. Can be formulated at different levels of detail.]

- Sections (2),(3) and (4) are needed to set-up and run simulations. The information about code accessibility (licensing etc), which MODA contains in the Simulation Overview, would be needed too.

The Translator/Modeller: given (1), proposes (2),(3),(4)

# Building on the MODA

**Heading Overview**

**(1) User Case**
**(2) Model**
**(3) Solver**
**(4) Processing**

We are building, starting from the MODA, a set of **structured keywords** to be able to **exhaustively describe the input and output of particle-based simulations** and to **set up a run**, all in a (as much as possible) code-independent way.

Therefore, we focus on (2), (3) and (4).

At the moment, we have a model branch and a solver branch. Processing is partially covered there.

When categorizing, we keep in mind the **practical use** of the information at different stages.

**Bottom-up** approach: a similar work has to be done for other **sub-domains,** identified as classes of similar methods.

# Particle-Based Simulation: Metadata for DPD and MD

**Modelling and simulation data and metadata for Dissipative Particle Dynamics**

- **(1) User Case**
- **(2) Model**
- **(3) Solver**
- **(4) Processing**

Model:

- Type and name
- Model entity
- Physics equation
- *Materials relations*
- Boundary conditions
- Simulated input

Materials relations:

- Bonding force
- Non-bonding conservative force
- Electrostatic force
- Random force
- Drag force
- External forces
- Population of unbonded species

**DPD simulation code from DL_MESO**

**https://www.scd.stfc.ac.uk/Pages/DL_MESO.aspx**

# Particle-Based Simulation: Metadata for DPD and MD

**Modelling and simulation data and metadata for Dissipative Particle Dynamics**

- **(1) User Case**
- **(2) Model**
- **(3) Solver**
- **(4) Processing**

**(2) Model:**

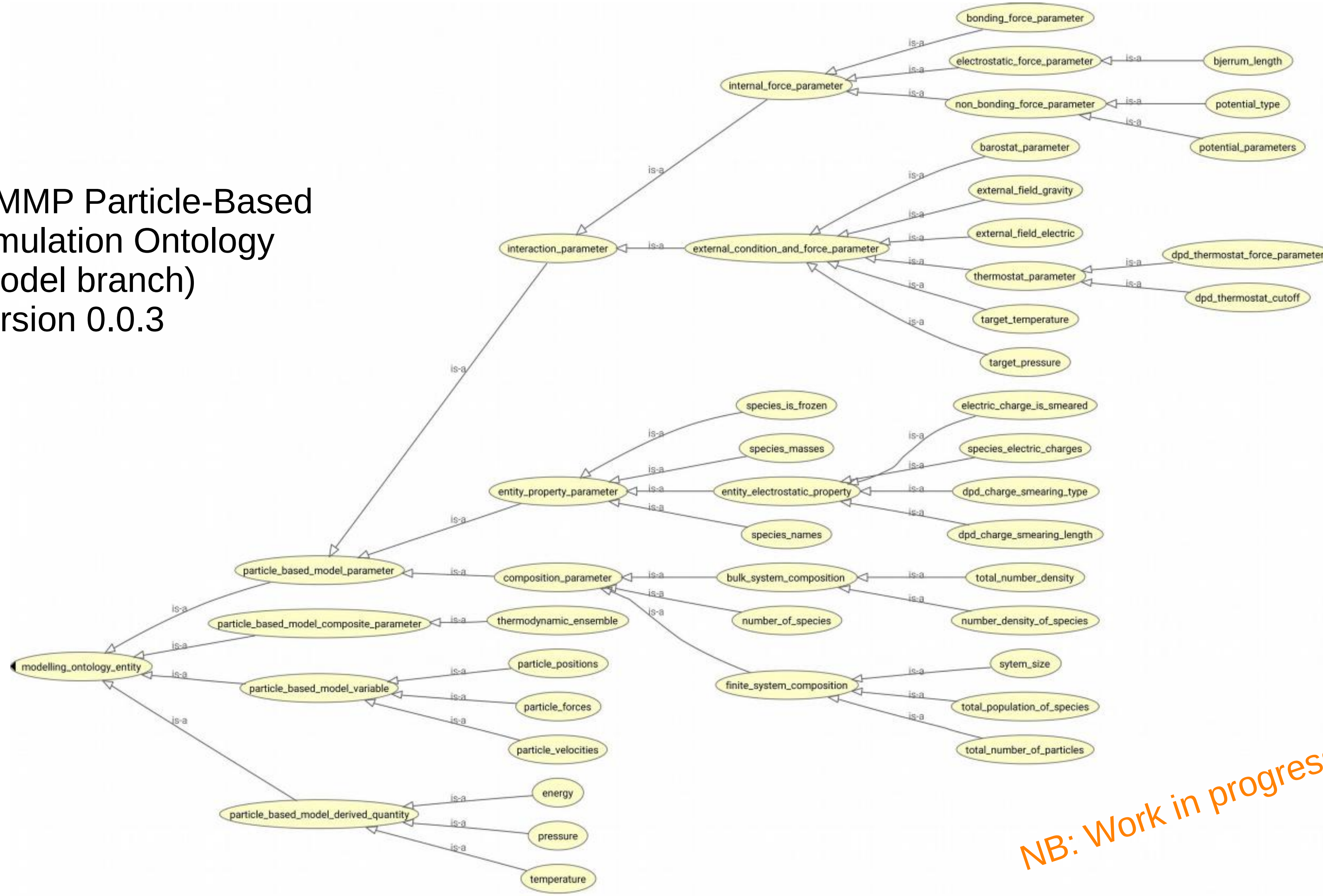
- Type and name
- Model entity
- Physics equation
- *Materials relations*
  - Bonding force
  - Non-bonding conservative force
  - Electrostatic force
  - Random force
  - Drag force
  - External forces
  - Population of unbonded species
- Boundary conditions
- Simulated input

**(3) Solver:**

- Numerical solver
- *Computational representation*
  - time_discretization, ensemble_and_integrator, random_number_generator, …
  - geometric_constraints_algorithm, simulation_box_size, …
- Software tool
- Boundary conditions
- Time step
- Solver parameters

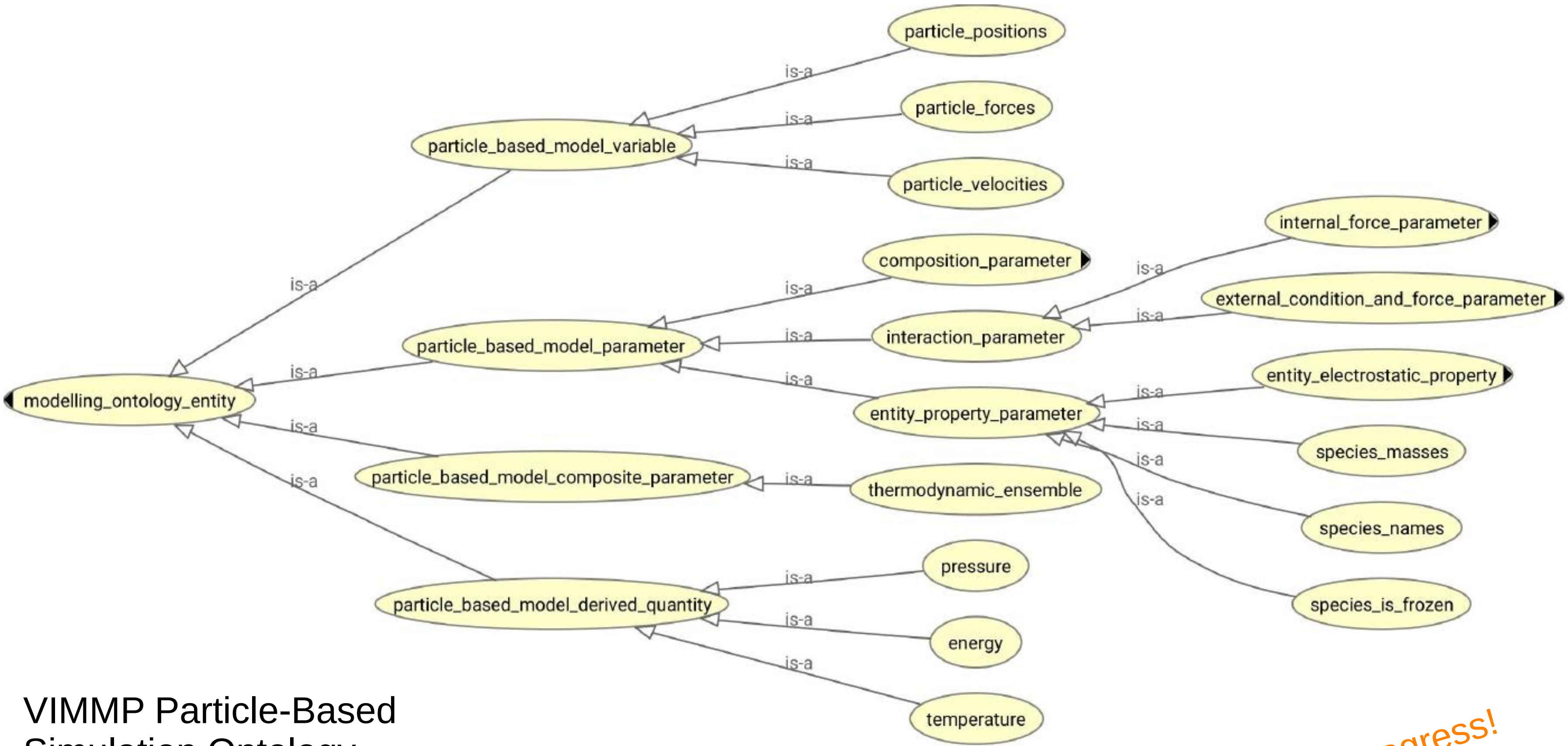
# Particle-Based Simulation Ontology: Model Branch

VIMMP Particle-Based Simulation Ontology (Model branch) Version 0.0.3




NB: Work in progress!

# Particle-Based Simulation Ontology: Model Branch




VIMMP Particle-Based Simulation Ontology (Model branch, zoom) Version 0.0.3

NB: Work in progress!

# Particle-Based Simulation Ontology: Model Branch

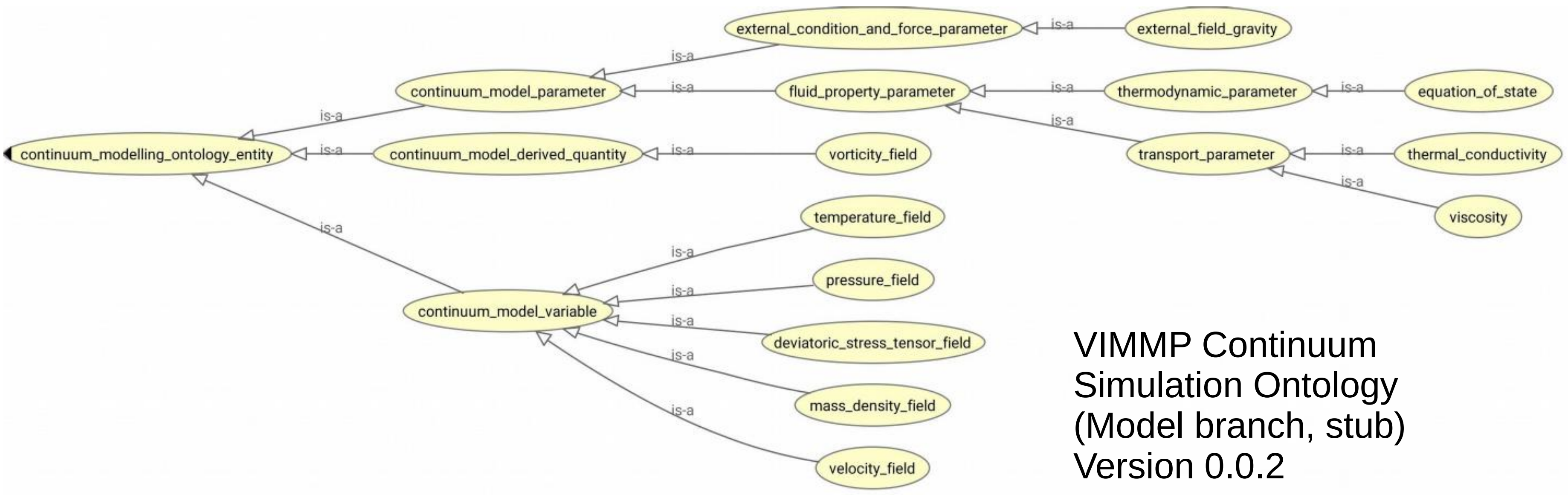
We propose a splitting of the metadata into:

- **Variables** (i.e., quantities appearing in the solved equations)
- **Derived quantities** (i.e., “processed” ones)
- **Parameters** (i.e., constants):
    - Entity property
    - Composition property
    - Interactions
        - Internal force
        - External condition and force

CAVEAT:
Further categorizing is useful to organize, but often ambiguous →Compromise

For example, the separation between Physics Equation and Material Relation is not always straightforward. The internal/external separation is not obvious either (see DPD thermostat, which affects the fluid viscosity).

# Continuum Simulation Ontology: Model Branch (stub)



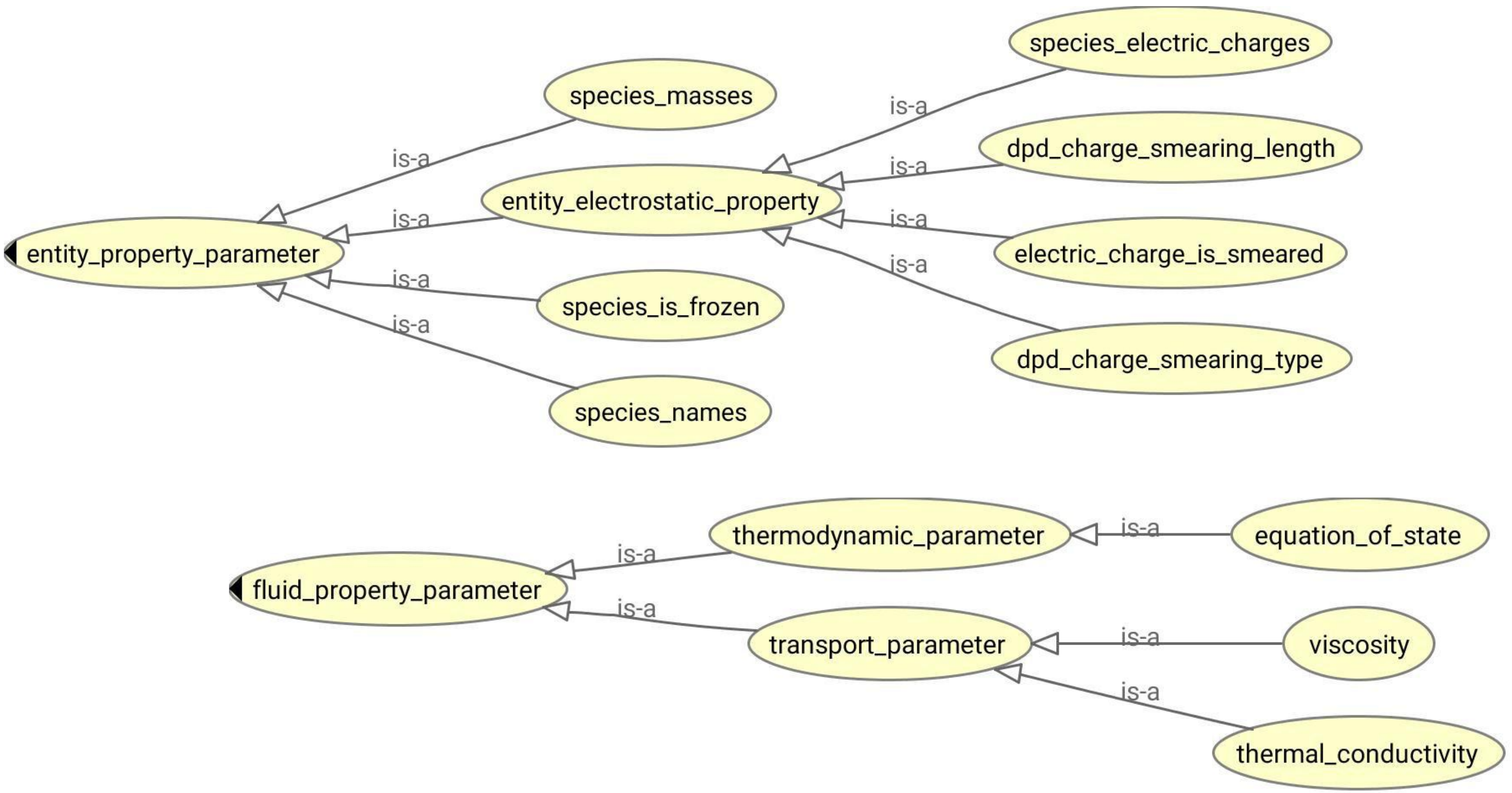

VIMMP Continuum Simulation Ontology (Model branch, stub) Version 0.0.2

NB: Work in progress!

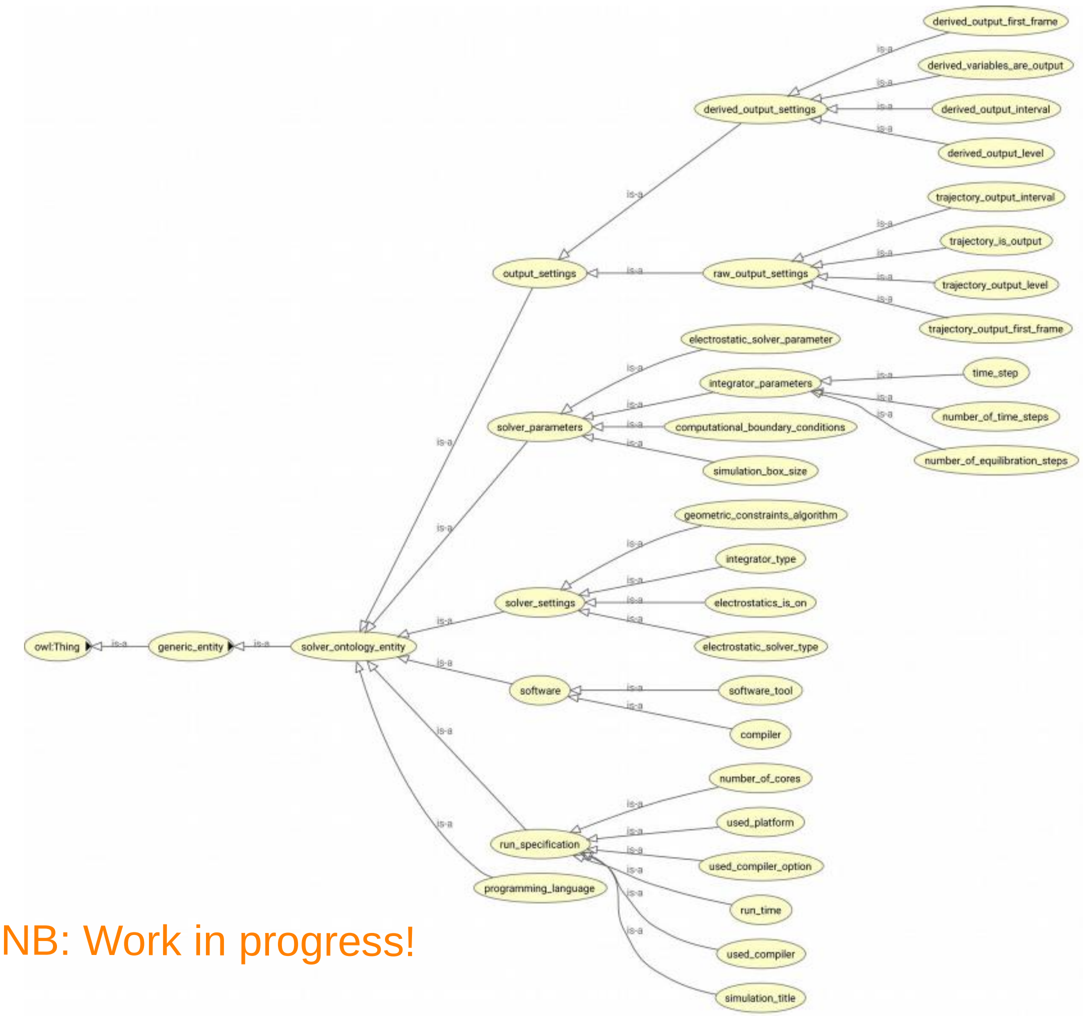
Simple case of a single fluid, 3D Navier-Stokes equation

# Comparison of Model Branches

For example, the properties for particles and those for fluids are:



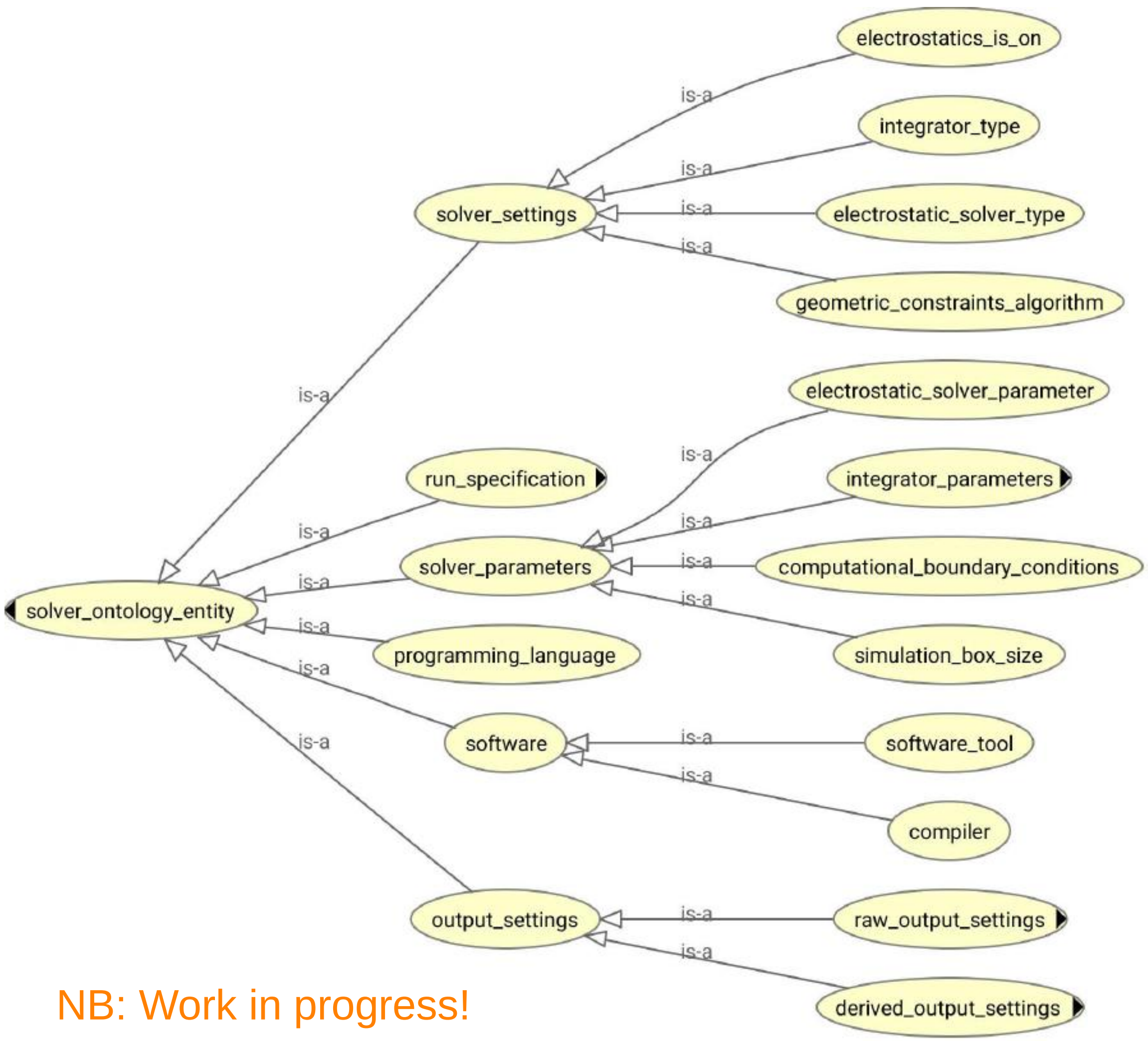
# Particle-Based Simulation Ontology: Solver Branch




NB: Work in progress!

VIMMP Particle-Based Simulation Ontology (Solver branch) Version 0.0.3

# Particle-Based Simulation Ontology: Solver Branch




VIMMP Particle-Based Simulation Ontology (Solver branch, zoom) Version 0.0.3

NB: Work in progress!

# Particle-Based Simulation Ontology: Solver Branch

Proposed splitting (in progress) of the solver metadata into four groups:

- **Info and access conditions**
  - Language
  - Version, repository, commit
  - Licensing (open, free, owner, ...)
  - Required libraries
  - Usable compilers, compiling options and platforms

- **Run specifications**
  - Used compiler, options, and platform, number of cores, run time, ...

- **Solver settings and parameters**

  Time step, sampling schedule, electrostatic solver type and its parameters, implementation of the geometric constraints, computational boundary conditions, …

- **Ouput settings**
  - Raw output settings, derived output settings

# Choice of units. Mapping for mesoscopic models

Most of the quantities described so far carry physical units: a consistent system of units needs to be defined and used.

We suggest to use **dimensional factorization**, which for simple mechanical systems reads: $[X]=L^a M^b T^c$, where L, M and T stay for length, mass and time.

For the units we can use the **International System** (SI) **or a customized one**, for example, defined from reference quantities $L_0$, $M_0$, $E_0$, where $E_0$ is a reference energy.

For a velocity, the units are $LT^{-1}$, and we could use m/s, or $L_0/T_0$, where $T_0=(L_0^2 M_0/E_0)^{1/2}$.

For mesoscopic models (as DPD), even if the reference units are given, the **mapping number** and the **matched properties** should be specified too.

NB: There are **available ontologies** for units such as the QUDT-Catalog, http://www.qudt.org/release2/qudt-catalog.html

# Conclusion and Outlook

- Work is in progress to identify the required metadata and develop a **simulation ontology**
- We will further develop the **sub-domain branches,** add constraints and relations between keywords, keeping in mind the experience of related projects (e.g., NOMAD, and SimPhoNY CUBA/CUDS)
- Allow for compatibility with relevant databases (e.g., as MolMod http://molmod.boltzmann-zuse.de/ for MD model parameters)
- Whenever possible, **connect with the EMMO** concepts (e.g., temperature)
- Develop code **wrappers** using these metadata

# Thank you for your attention!


## Acknowledgements

- The Consortium Partners, especially:

  Fraunhofer IFAM, Bremen (Germany)
  ICPF, Prague (Czech Republic)
  Politecnico di Torino (Italy)
  OSTHUS (Germany)
  EDF (France)

- Stanislav Pařez, ICPF

- Jony Castagna, STFC

*This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 760907.*

*This document and all information contained herein is the sole property of the VIMMP Consortium. It may contain information subject to intellectual property rights. No intellectual property rights are granted by the delivery of this document or the disclosure of its content.*

*Reproduction or circulation of this document to any third party is prohibited without the consent of the author(s).*

*The statements made herein do not necessarily have the consent or agreement of the VIMMP consortium and represent the opinion and findings of the author(s).*

Science & Technology Facilities Council | UK Research and Innovation

VIMMP VIRTUAL MATERIALS MARKETPLACE