FLANDERS MAKE

DRIVING INNOVATION IN MANUFACTURING

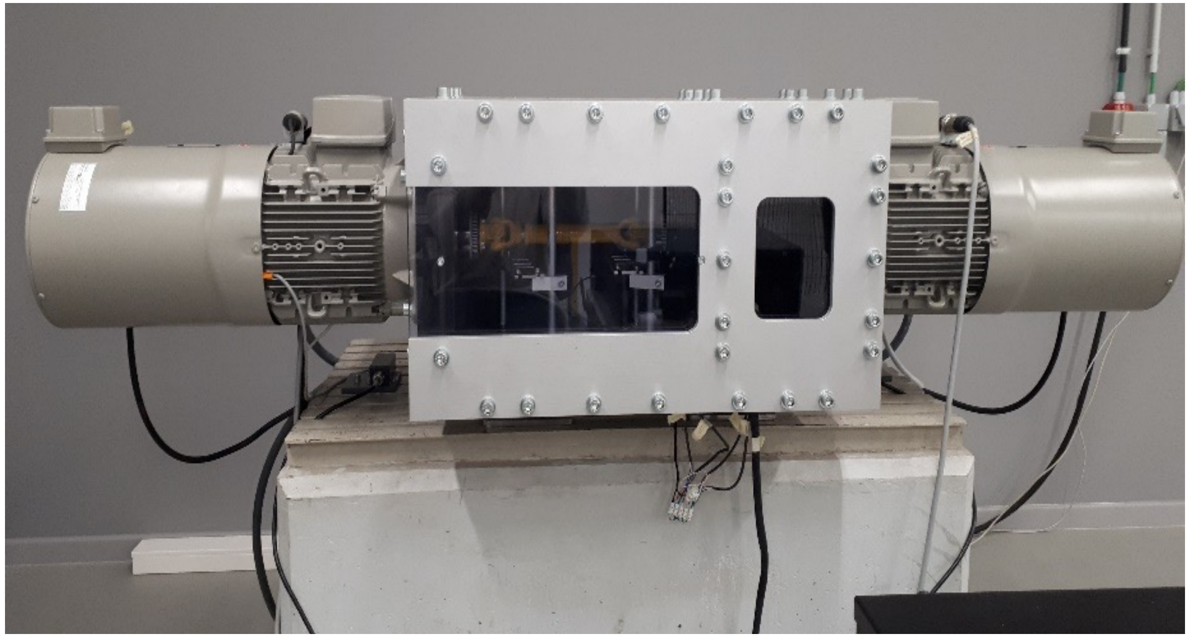
DTDesign Project

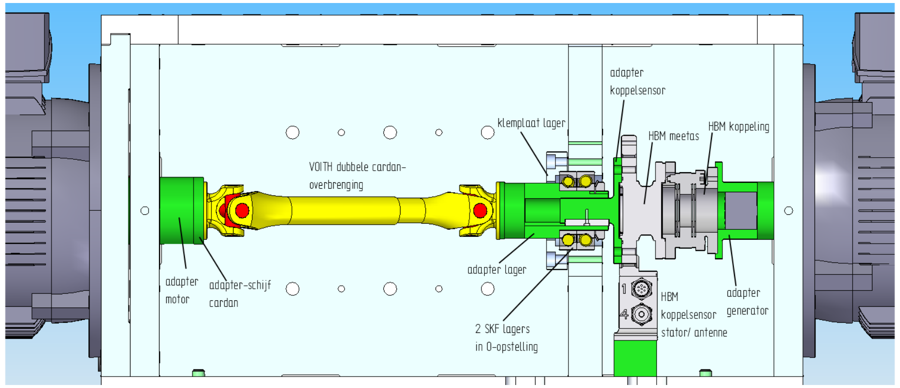
Supporting SE experiments

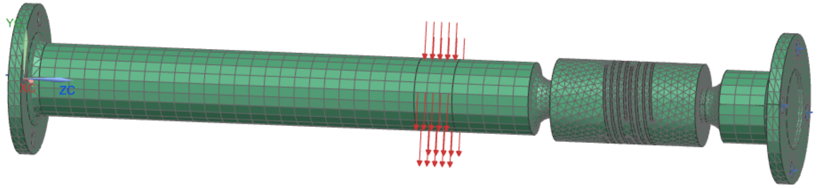
Lucas Lima, Arkadiusz Rys, Joeri Exelmans

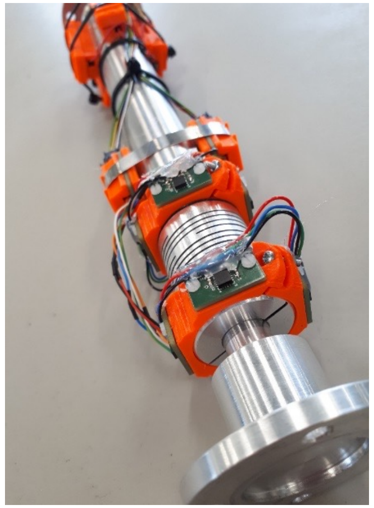
09 October 2023

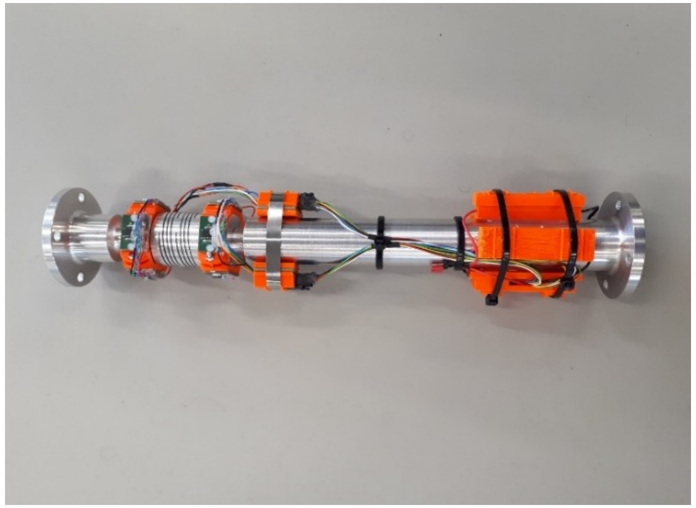
Drivetrain - System

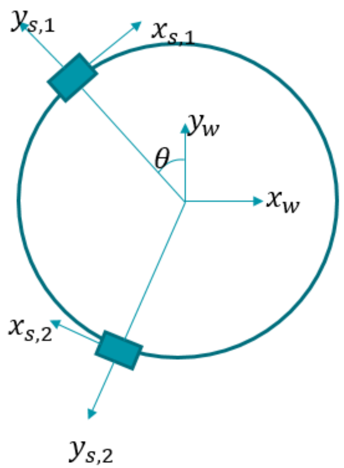




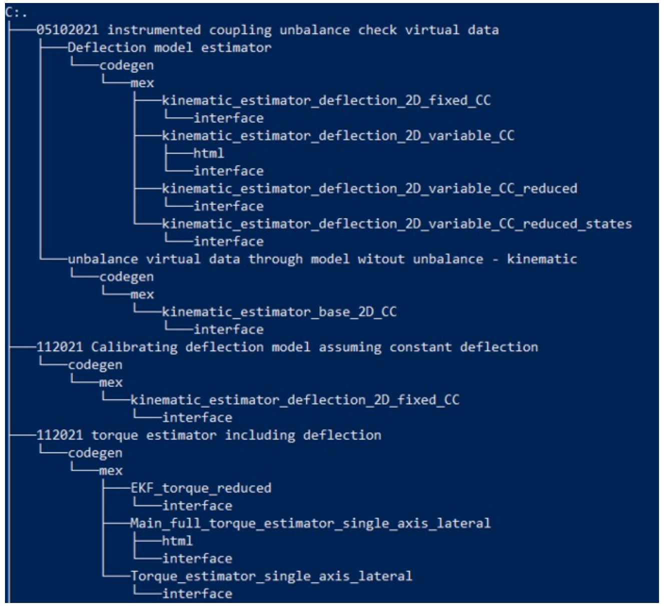






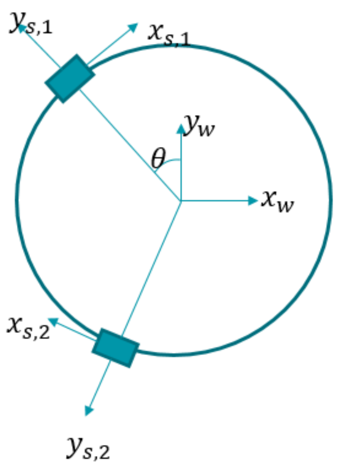


- How to find the model of experiment X?
- What are the steps to perform experiment X?
- Which version did give this result?
- What do I need to execute this next step?
- What should I provide by the end of this step?

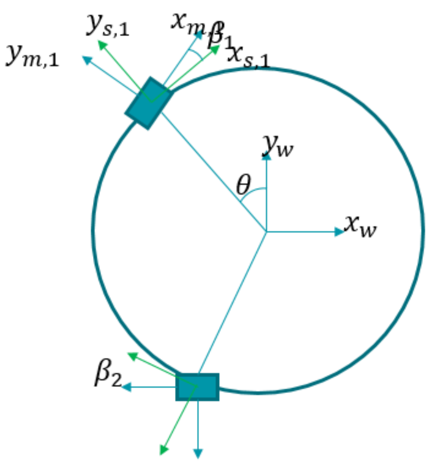


```
C:.
├───05102021 instrumented coupling unbalance check virtual data
│   ├───Deflection model estimator
│   │   └───codegen
│   │       └───mex
│   │           ├───kinematic_estimator_deflection_2D_fixed_CC
│   │           │   └───interface
│   │           ├───kinematic_estimator_deflection_2D_variable_CC
│   │           │   ├───html
│   │           │   └───interface
│   │           ├───kinematic_estimator_deflection_2D_variable_CC_reduced
│   │           │   └───interface
│   │           └───kinematic_estimator_deflection_2D_variable_CC_reduced_states
│   │               └───interface
│   └───unbalance virtual data through model witout unbalance - kinematic
│       └───codegen
│           └───mex
│               └───kinematic_estimator_base_2D_CC
│                   └───interface
├───112021 Calibrating deflection model assuming constant deflection
│   └───codegen
│       └───mex
│           └───kinematic_estimator_deflection_2D_fixed_CC
│               └───interface
├───112021 torque estimator including deflection
│   └───codegen
│       └───mex
│           ├───EKF_torque_reduced
│           │   └───interface
│           ├───Main_full_torque_estimator_single_axis_lateral
│           │   ├───html
│           │   └───interface
│           └───Torque_estimator_single_axis_lateral
│               └───interface
```

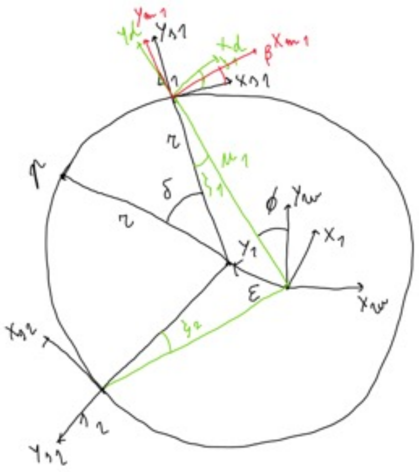
Model 1



Model 2



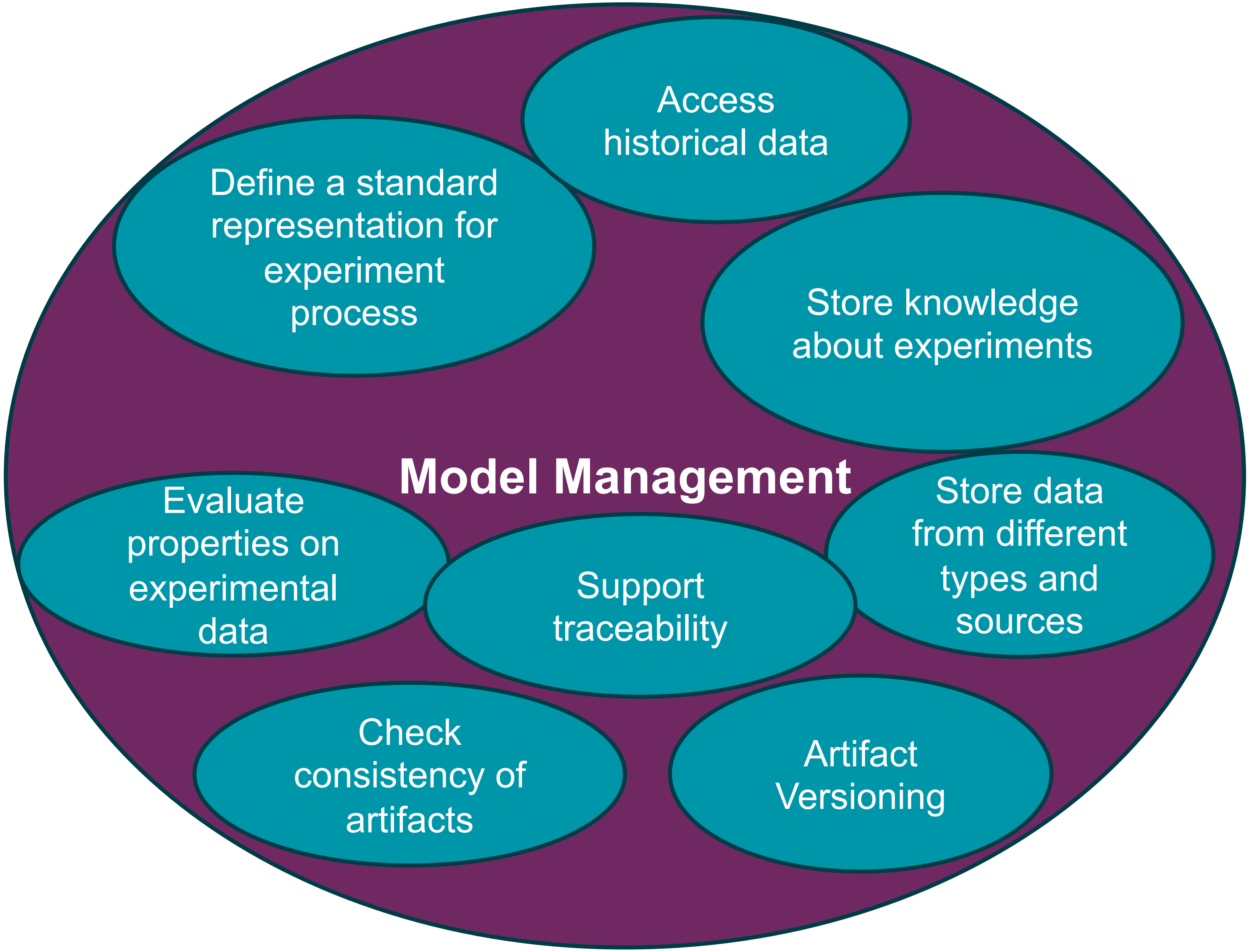
Model 3



Goals - What do we want to accomplish?

- Define a standard representation for experiment process
- Store knowledge about experiments
- Artifact Versioning
- Support traceability
- Access historical data
- Store data from different types and sources
- Evaluate properties on experimental data
- Check consistency of artifacts

Goals - What do we want to accomplish?

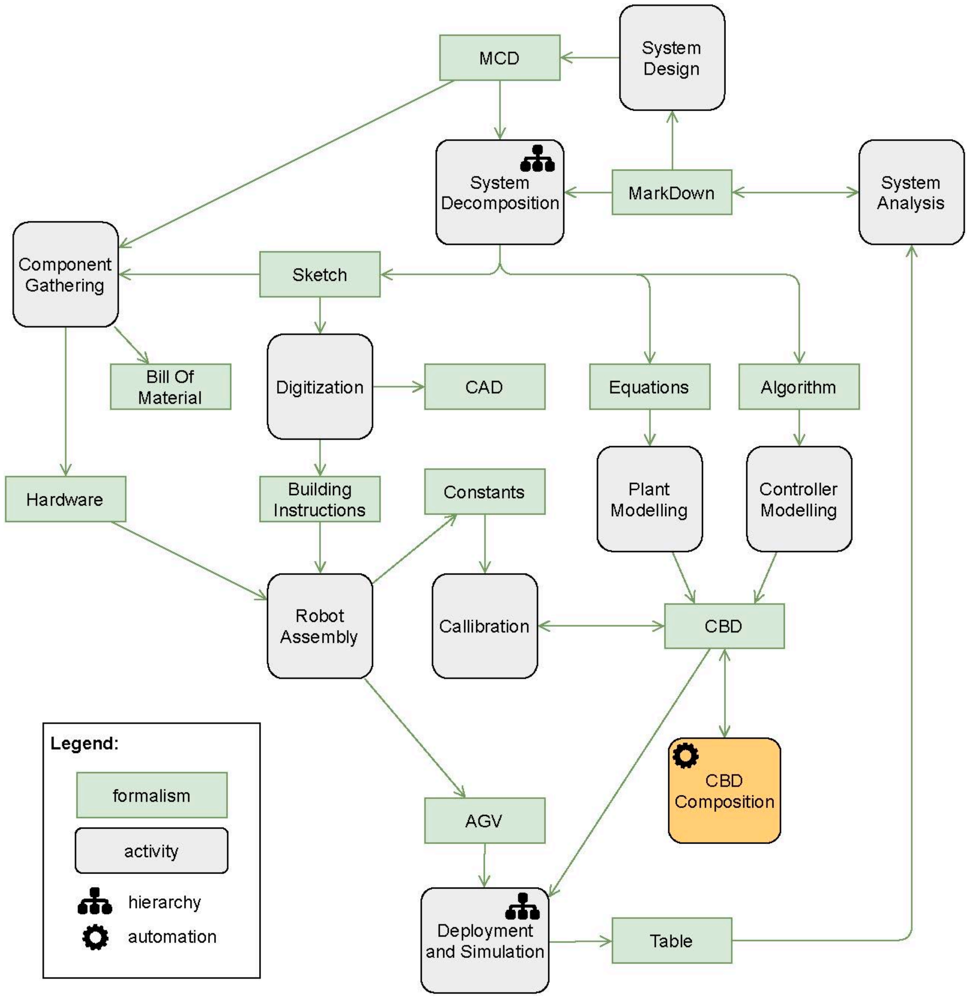


How do we accomplish it?

Meta Models	(MM)
Formalism Transformation Graph	(FTG)
Process Model	(PM)
Process Trace	(PT)
Storage, Services, Real-World Artifacts	(SSRWA)

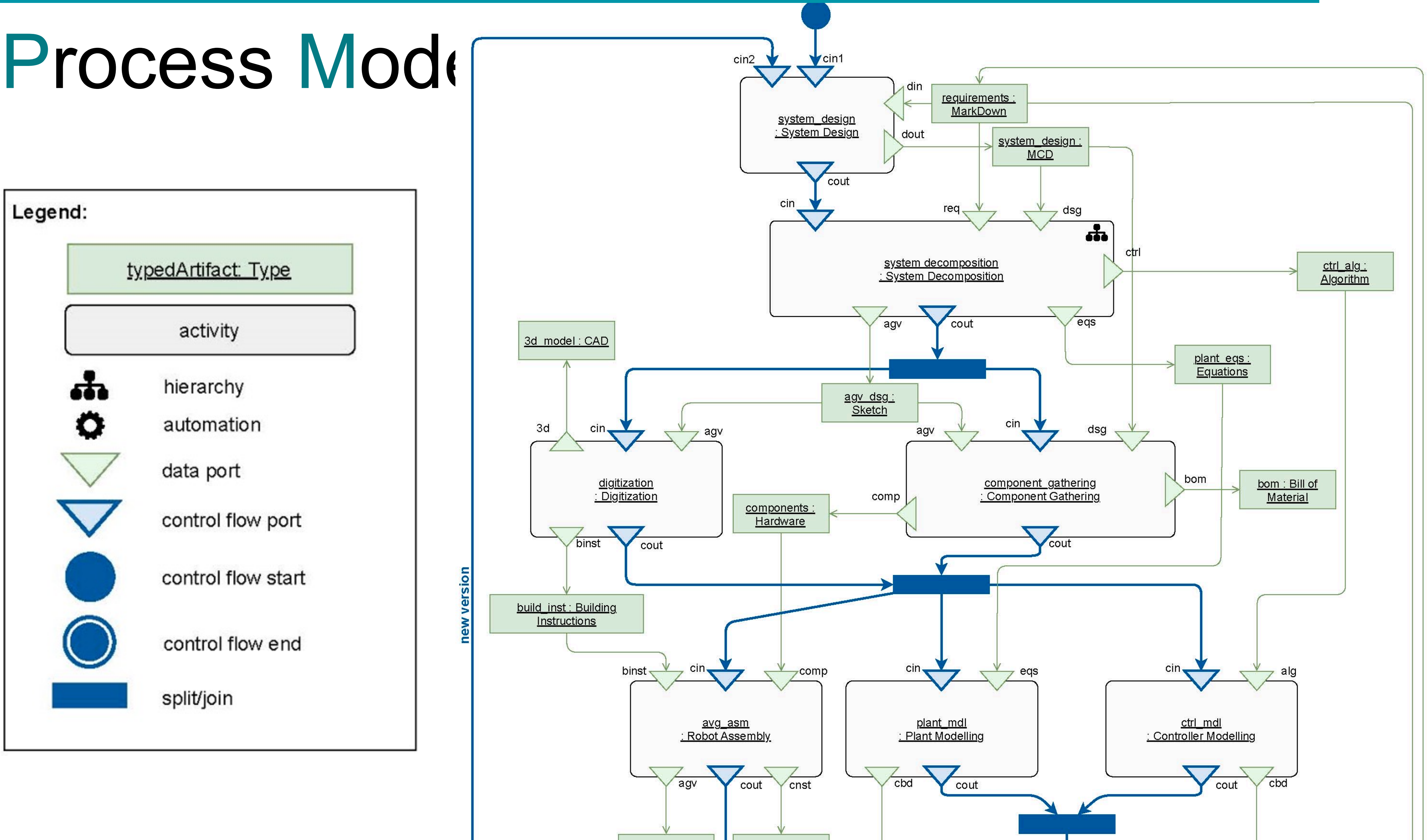
MM + FTG + PM + PT + SSRWA => FTG+PM++

Formalism Transformation Graph



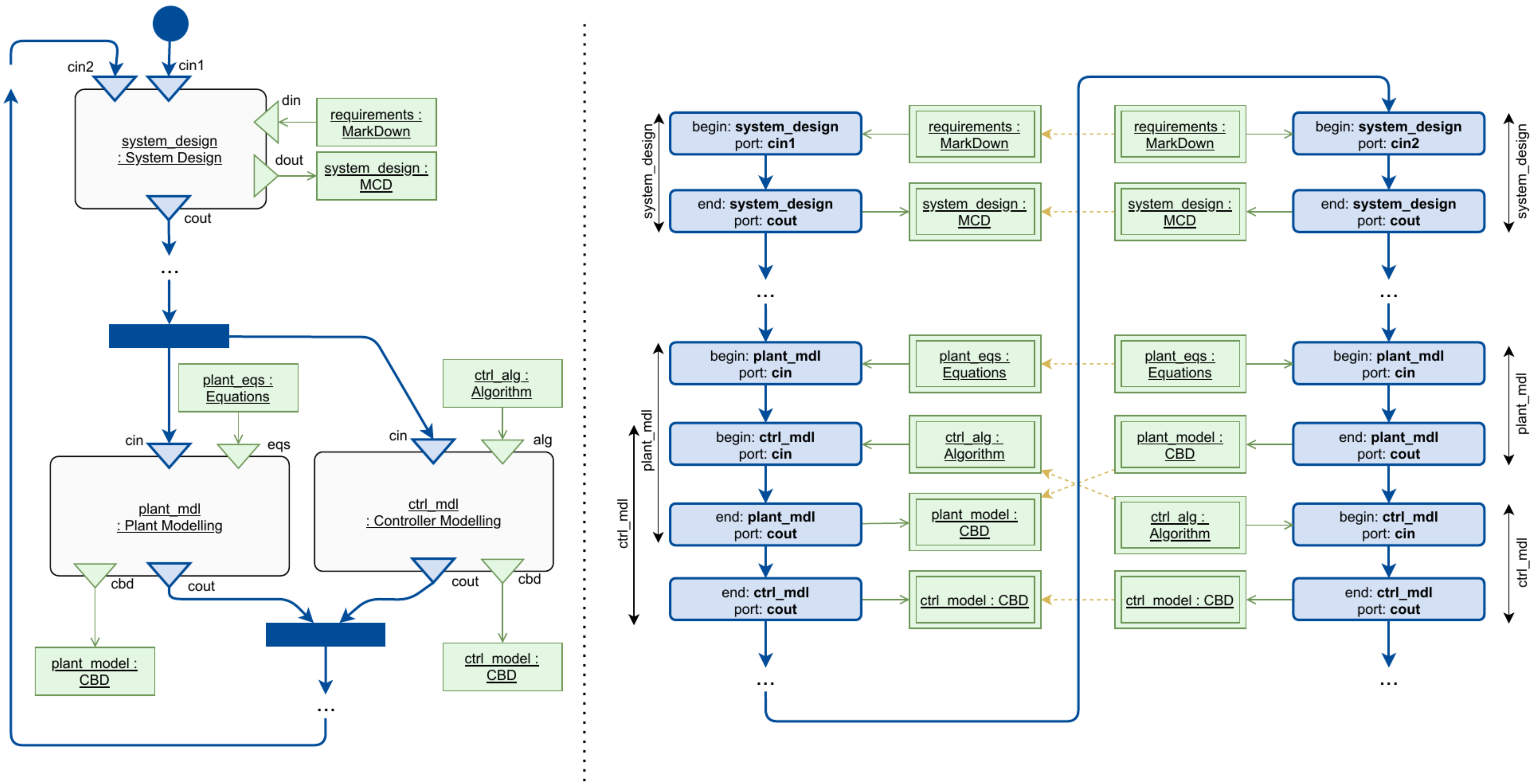
How do we accomplish it?

Process Mode



How do we accomplish it?

Process Trace



A trace allows us to answer some questions

- How many design/requirement iterations are present?
- What changed in the requirements?
- Which files are changed during this project?
- Where did specific data get generated?
- What is the influence of requirement change X on Y?

Types of traceability

- Traceability linking experiment and system
- Traceability across artifact versions
- Traceability based on properties of interest
- Traceability between artifacts on different levels of detail
- Fine-grained traceability between artifact elements
- Traceability between instances and types

First Decision...

First Decision...

Modelverse

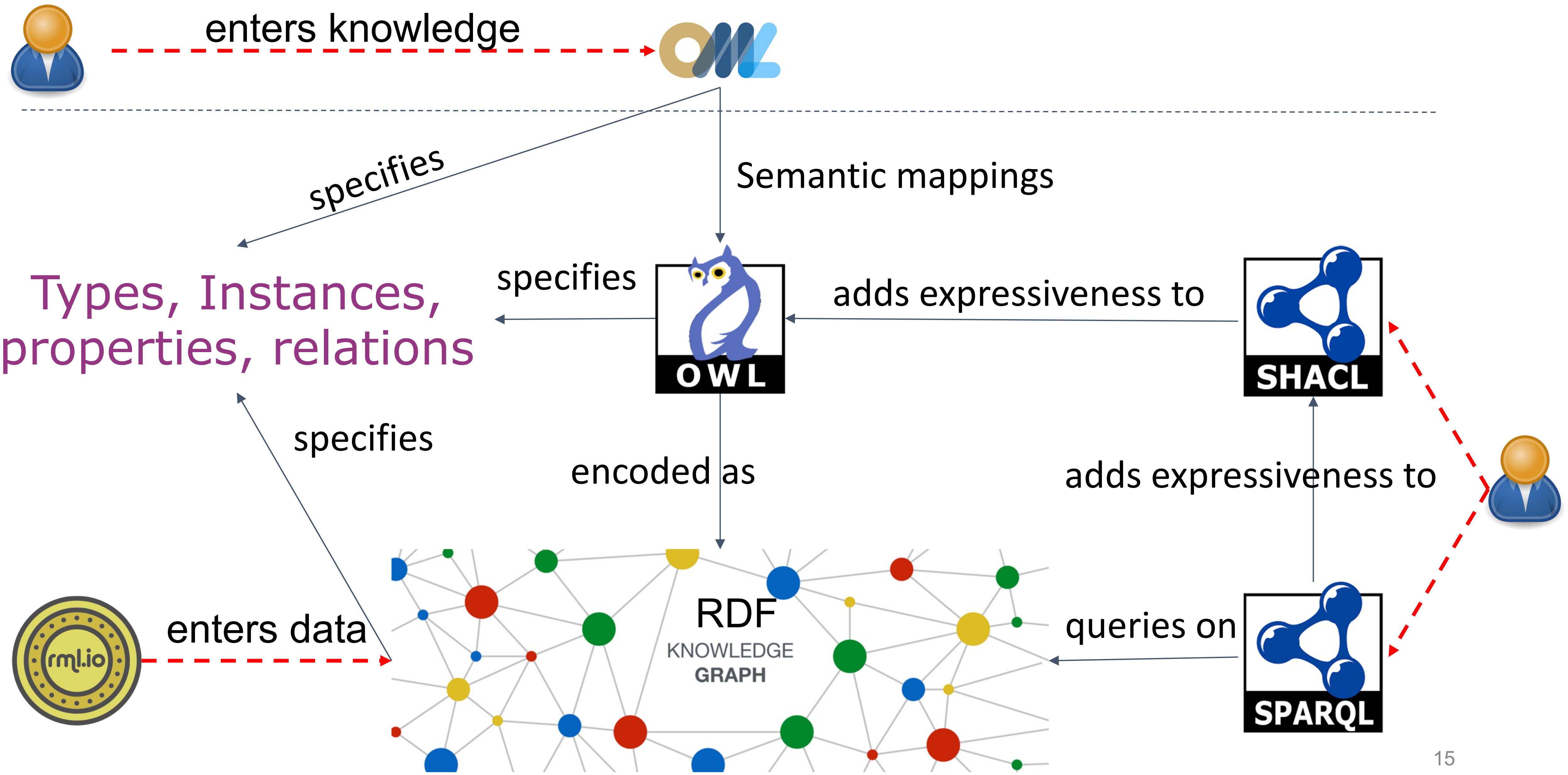
- Born in Antwerp
- Environment for Multi-paradigm Modelling
- Everything is a model
- Do not commit to a specific materialization
  - Memory, cloud, RDB, RDF
- API
- No query language

First Decision...

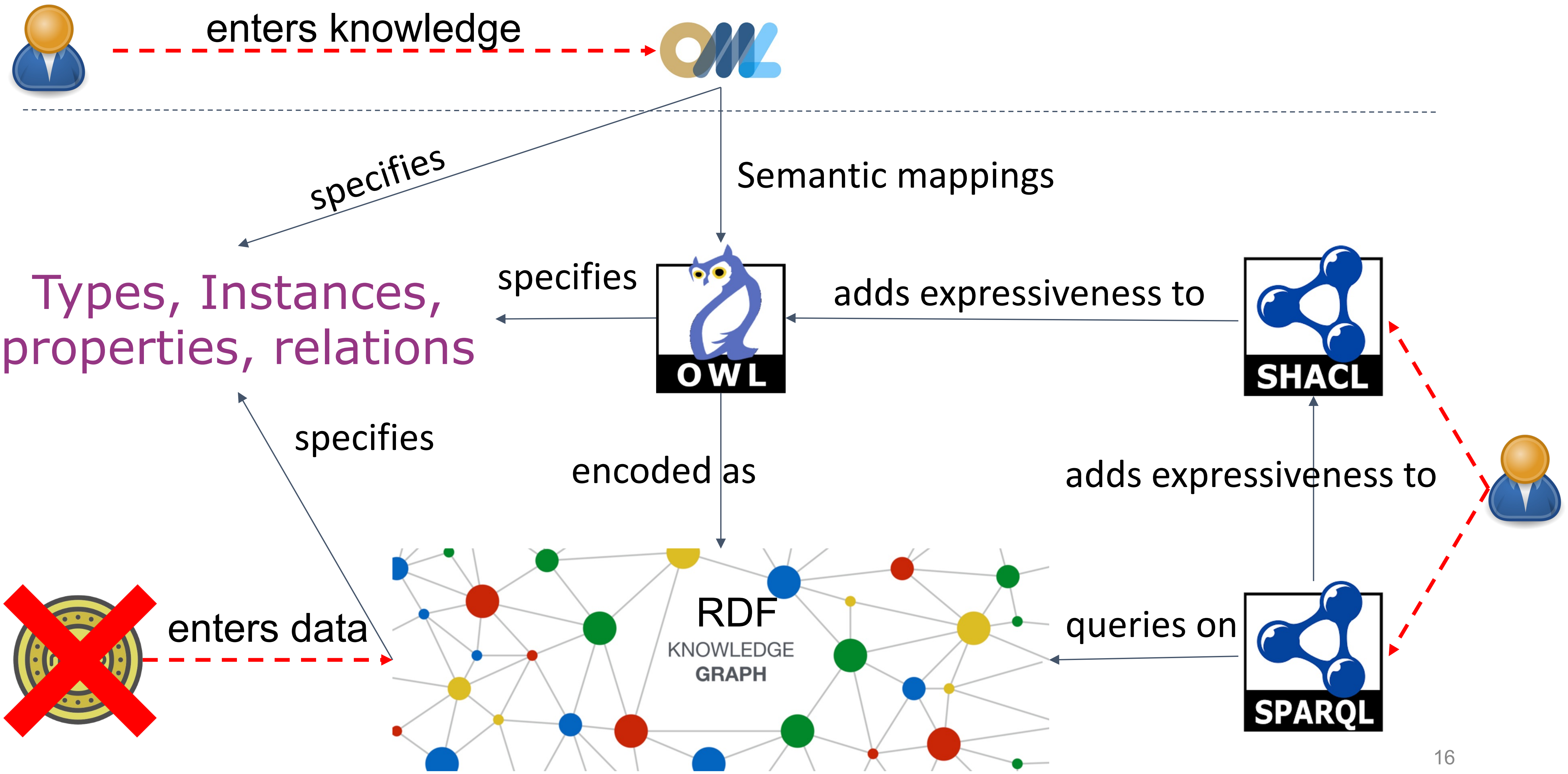
- Investigation on current technologies
  - OML, OWL/RDF, RML, SPARQL, SHACL, Knowledge Graphs
- Potential to be one Modelverse implementation

Ontological stack

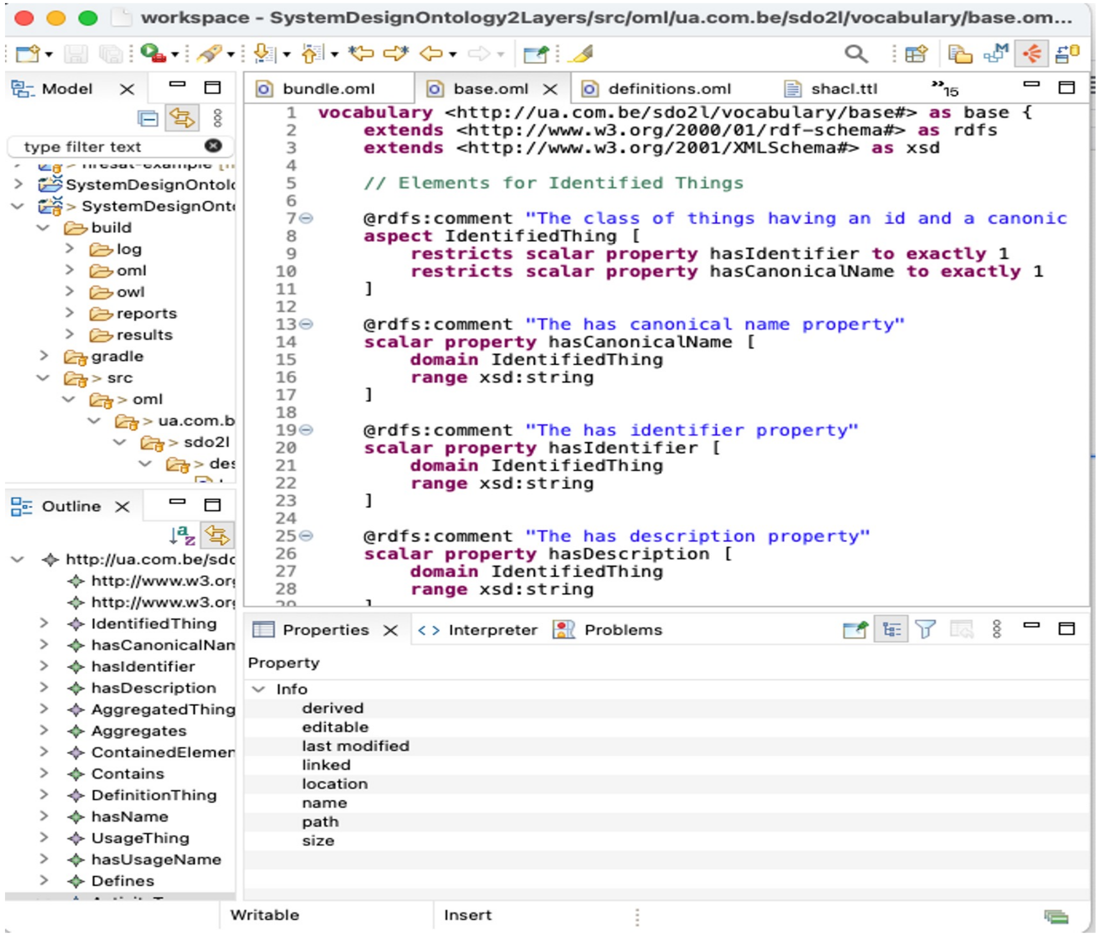
Ontology stack evaluation Study



Ontology stack evaluation Study

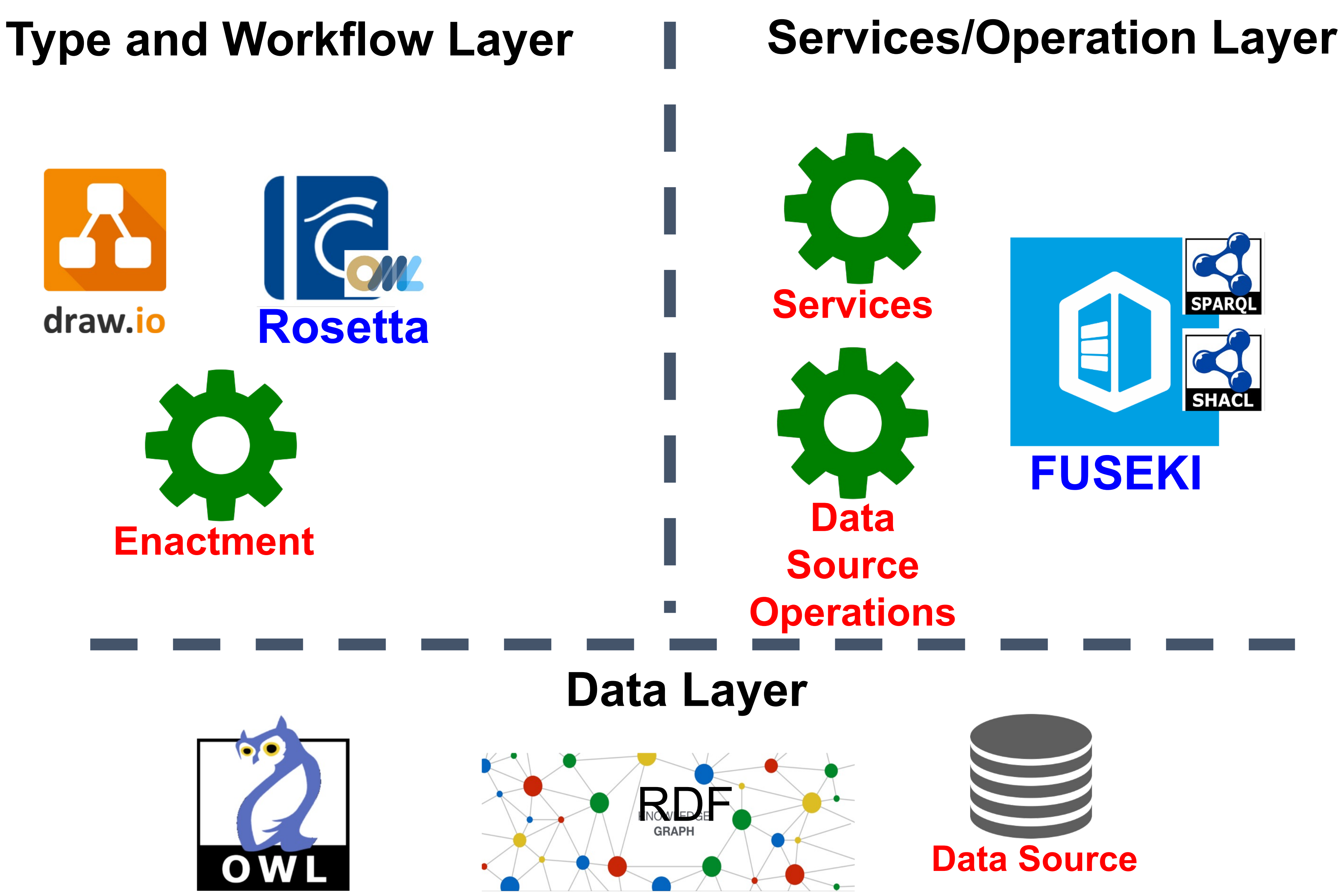


Rosetta: Packaged Eclipse IDE

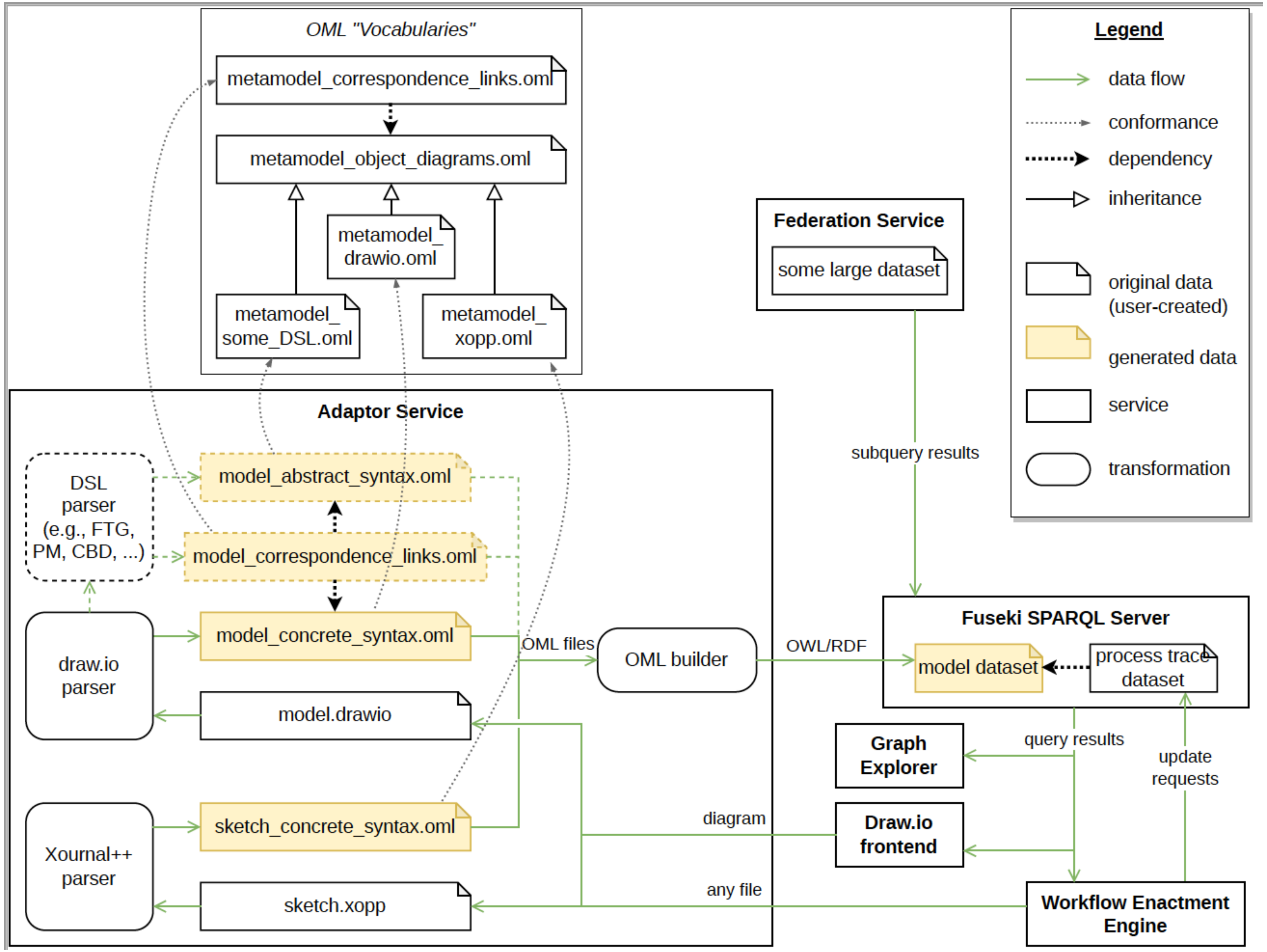


- Ontology for Metamodeling and Consistency Checking
- Vocabularies (T-boxes)
- Descriptions (A-boxes)
- Constraints
  - Max/Min relationships
  - Symmetric/Asymmetric
  - Reflexivity
  - Ruling mechanism

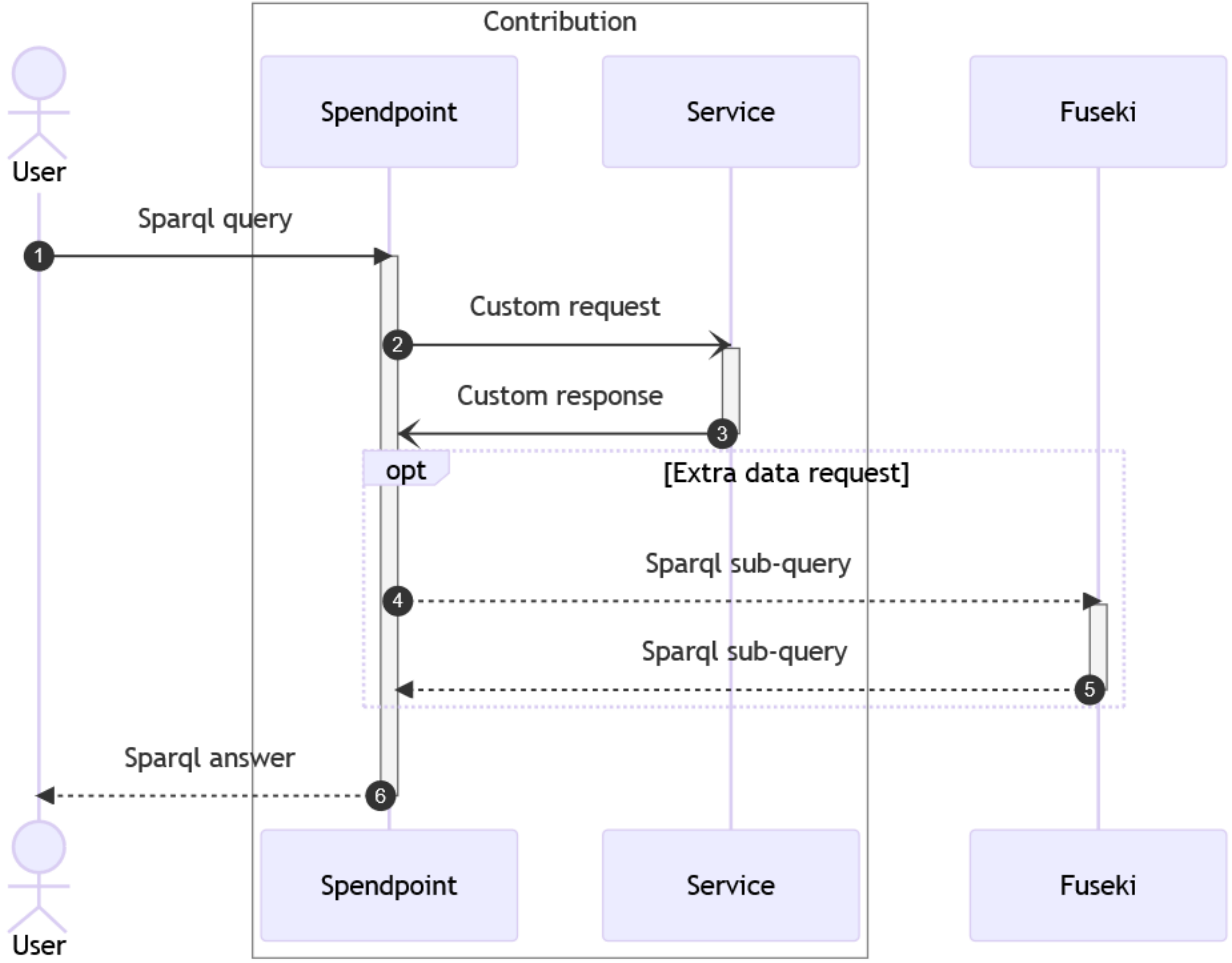
High-level Architecture



Technical Architecture

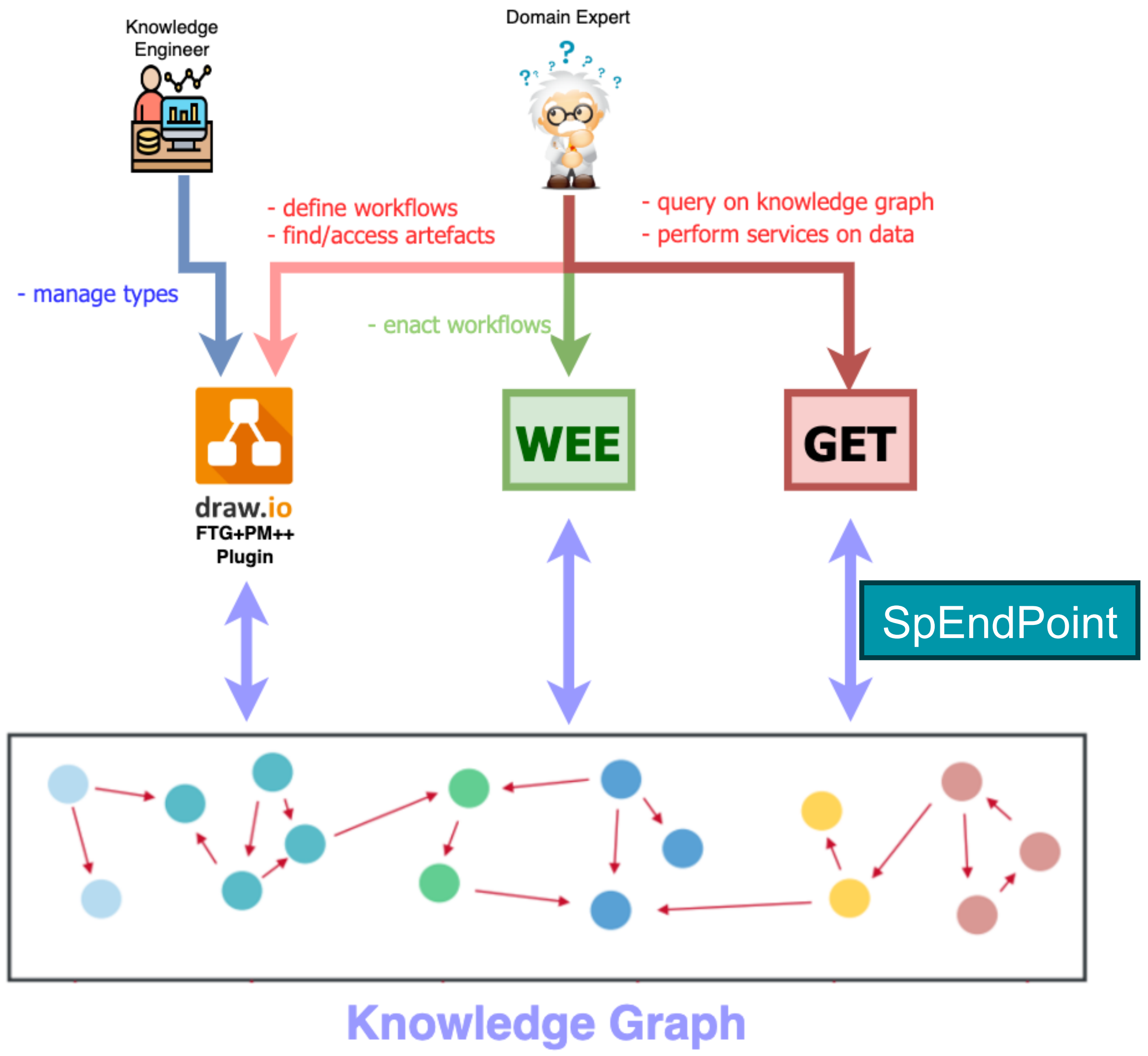


Federation



Contribution
User
Spendpoint
Service
Fuseki
Sparql query
Custom request
Custom response
opt
[Extra data request]
Sparql sub-query
Sparql sub-query
Sparql answer
1
2
3
4
5
6
Spendpoint
Service
Fuseki
User

Goals – How do we accomplish it?



DEMO

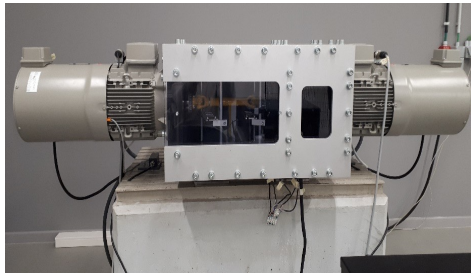
Evaluation

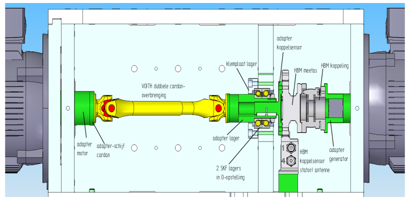
Stakeholder value	Traditional approach	FTG-PM - Ontological approach
Evaluation (details below)	Q=3.5	Q=8
2.2.1. Scenario 1: Experiment Modeling cases • The cases in this section are related to the experimental modeling framework developed for the DTDesign project.		
2.2.1.1. Experiment artifact management	Q=3 › Click here to expand...	Q=8 › Click here to expand...
2.2.1.2. Model versioning	Q=3 › Click here to expand...	Q=8 › Click here to expand...
2.2.1.3. Experiment workflow modeling	This aspect was not included before the given solution, but is valuable to add.	Q=8 › Click here to expand...
2.2.1.4. Experiment reasoning	Q=4 › Click here to expand...	Q=7 › Click here to expand...
2.2.1.5. Experiment repeatability/replicability	Q=4 › Click here to expand...	Q=9 › Click here to expand...

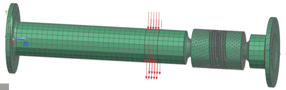
Future works

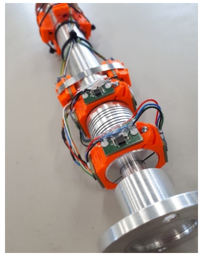
- Apply this framework for different cases:
  - Experiment validity workflow (notch filter case)
  - Early validation reasoning (Volvo case)
  - Workflow support and automation (Octiva case)
- Integration with the Modelverse framework (linguistic type)
- Evolve virtualisation/federation
- Support hierarchical workflows
- Formalization of the workflow language (Brest)
  - Debugger, Model checker, Interpreter

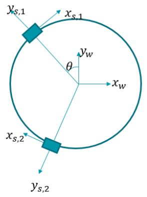
Conclusion

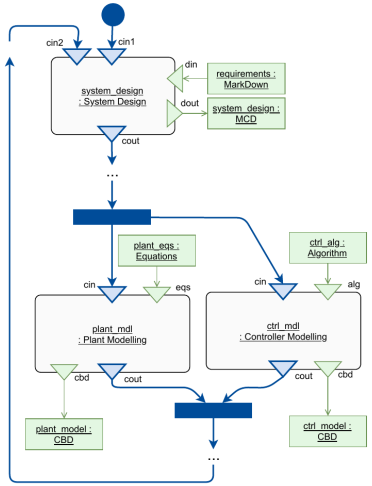


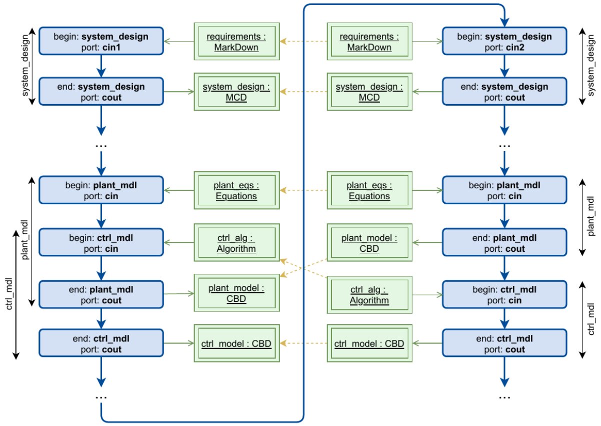


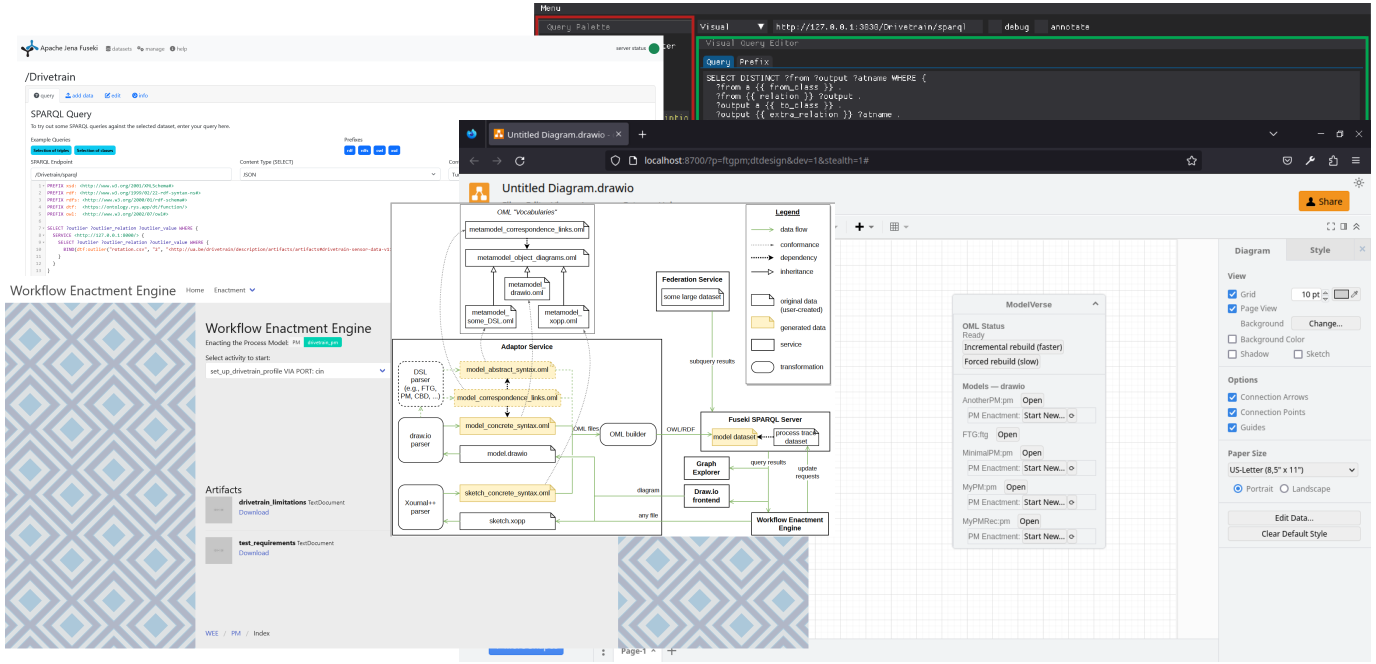












Stakeholder value	Traditional approach	FTG-PM - Ontological approach
Evaluation (details below)	Q=3.5	Q=8
2.2.1. Scenario 1: Experiment Modeling cases		
• The cases in this section are related to the experimental modeling framework developed for the DTDesign project.		
2.2.1.1. EXPERIMENT ARTIFACT MANAGEMENT	Q=3 › Click here to expand...	Q=8 › Click here to expand...
2.2.1.2. MODEL VERSIONING	Q=3 › Click here to expand...	Q=8 › Click here to expand...
2.2.1.3. EXPERIMENT WORKFLOW MODELING	This aspect was not included before the given solution, but is valuable to add.	Q=8 › Click here to expand...
2.2.1.4. EXPERIMENT REASONING	Q=4 › Click here to expand...	Q=7 › Click here to expand...
2.2.1.5. EXPERIMENT REPEATABILITY/REPLICABILITY	Q=4 › Click here to expand...	Q=9 › Click here to expand...

FLANDERS MAKE

DRIVING INNOVATION IN MANUFACTURING

DTDesign Project

Supporting SE experiments

Lucas Lima, Arkadiusz Rys, Joeri Exelmans

09 October 2023