ROMAIN: Reference Ontology for Industrial Maintenance

Hedi KARRAY (mkarray@enit.fr)

Associate Professor INP-ENIT

Second EC workshop on Data Documentation

ROMAIN: Towards a BFO compliant Reference Ontology for Industrial Maintenance.
MH. Karray, F. Ameri, M. Hodkiewicz, T. Louge. (2019). Applied Ontology. Vol14 (2). DOI:10.3233/AO-190208

What is the application domain of your ontology?

- ***ROMAIN*** is a reference ontology for the ***industrial maintenance*** management domain.

What is the intended purpose of the ontology?

- ROMAIN can be used for maintenance data interoperability, integration and analysing from management perspectives.

- Maintenance management can be viewed through two different lenses:
  - The first is the view of maintenance practitioners of the work management process
    - Maintenance is a process in which work on many assets is identified, planned, scheduled, executed and analysed.
  - The second view considers the requirements of a specific asset,
    - How it should be maintained over its life, its reliability and the risk that its failure presents to the organization.

- ROMAIN deals with the first view.

How do you represent the world?

- While using BFO as an upper level ontology, ROMAIN represents the world as a continuum
  - Dispositions are variable along a continuum from weaker to stronger.
  - BFO views spacetime, as a whole, existing in its entirety in its four (three plus time) dimensions. Processes, in this space- time, have a duration, a beginning, and an end. One can think of each process as a temporally extended continuum, a spacetime *worm*, stretched out in and through the single unified container that is the entirety of spacetime.

NB : this view of spacetime worms is distinct from popular four-dimensionalist views according to which objects (such as molecules or people or planets) would themselves be extended in time and would have temporal parts.

What are the concepts, with definitions, in the upper level of your ontology?

- BFO classes.
- Reused Common Core Ontologies classes
- Reused Information Artifact Ontology classes
- All ROMAIN classes come with definitions, respecting the Aristolian definition principle.

What are the industrial use cases of the ontology

Asset Functional location	Utilized hours	Cumulative utilized hours	Work order trigger	Work order Description	Work order code	Actual costs	Planned costs
EQ1_engine				Starts			
EQ1_engine	210	210	Breakdown	REPAIR ENGINE SHUTTING DOWN	PM03	1,014	-
EQ1_engine	1085	1295	Breakdown	REPAIR LOW ENG OIL LEVEL ERROR	PM03	-	507
EQ1_engine	3080	4375	FromInspection	Replace Eng Oil Level Switch	PM02	768	605
EQ1_engine	1540	5915	Minor	REPAIR LACK OF POWER	PM03	357	357
EQ1_engine	1768	7683	Fixed Interval Restoration	OEM Model 1 LOADER 8000HR ENGINE MIDLIFE	PM01	44,235	49,184
EQ1_engine	0	7683	FromInspection	Replace LH Eng Fuel Rail Clamp	PM02	159	390
EQ1_engine	578	8260	Breakdown	Engine derating.	PM03	312	-
EQ1_engine	3850	12110	Breakdown	REPAIR ENG OIL LEAK	PM03	795	1,102
EQ1_engine	508	12618	FromInspection	REPAIR BROKEN LH ENG COVER MOUNT BRACKET	PM02	323	323
EQ1_engine	35	12653	FromInspection	REPLACE FAULTY ENGINE SPEED TIMINGSENSOR	PM02	1,279	584
EQ1_engine	175	12828	FromInspection	INSPECT/REPAIR ENGINE STARTER MOTOR	PM02	785	785
EQ1_engine	88	12915	FromInspection	WELD UP CRACKED ENGINE PANELS	PM02	521	-
EQ1_engine	1138	14053	FromInspection	INVESTIGATE ENG OIL LEAK L/HFRONT	PM02	2	388
EQ1_engine	735	14788	Fixed Interval Replacement	OEM Model 1 LOADER ENGINE C/O	PM01	225,849	37,277
EQ1_engine	0	14788	FromInspection	REPAIR OIL LEAKS AT FRONT OF ENGINE	PM02	645	4,629
EQ1_engine	368	15155	Fixed Interval Restoration	OEM Model 1 LOADER 8000HR ENGINE MIDLIFE	PM01	-	53,618
EQ1_engine	0	15155	FromInspection	REPLACE CRACKED ENGINE COOLING FAN	PM02	6,547	5,972
EQ1_engine	2625	17780	FromInspection	CMT engine Fuel dilution	PM02	6,426	3,729
EQ1_engine	1295	19075	FromInspection	REPLACE ENGINE SPEED SENSOR	PM02	312	137
EQ1_engine	2083	21158	FromInspection	RESEAL LEAKING A/C DRIVE ON ENGINE	PM02	1,134	1,134
EQ1_engine	1838	22995		Ends			

The ontology is validated in a real maintenance strategy effectiveness use case: *Determine how effective the selected maintenance strategy is in practice*.

ROMAIN is used to integrate data coming from a CMMS used by an operator of *Heavy Mobile Equipment (HME), [*one of the most expensive maintainable items is the engine.*]*

We have integrate all of tasks executed on the engines of four identical HME assets in their first 18,000 operating hours.

SPARQL query:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX romain: <http://www.semanticweb.org/tlouge/ontologies/2018/11/ROMAIN#>
SELECT (count(?failure) as ?failureCount)
        WHERE {?failure rdf:type romain:failureEvent .}
```

ROMAIN allows to answer the following questions:

Such questions need a manual work of human experts for few days to link data !

How many breakdown work orders were executed?

failureCount
"4"^^<http://www.w3.org/2001/XMLSchema#integer>

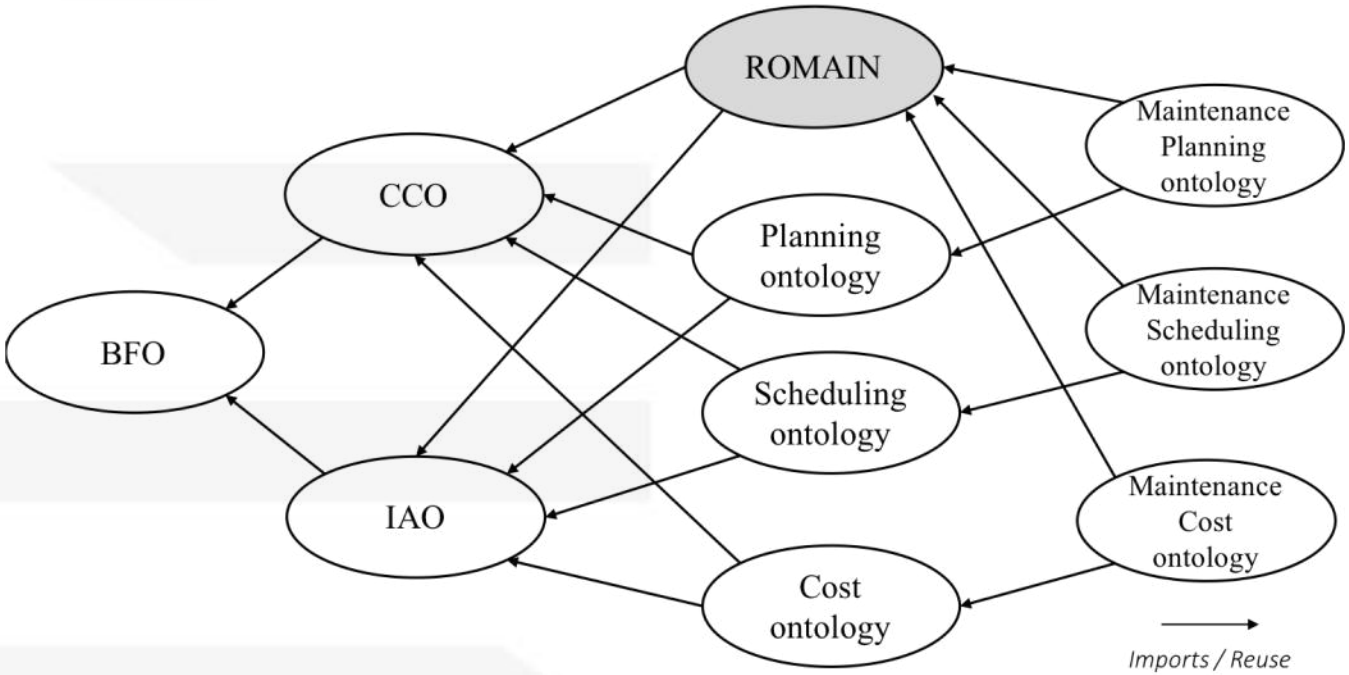
SPARQL query:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX romain: <http://www.semanticweb.org/tlouge/ontologies/2018/11/ROMAIN#>
SELECT distinct ?worder ?event ?val
        WHERE {?event rdf:type romain:event_tirggered_by_Fixed_Interval_Preventative_Maintenance_Task_execution .
        ?worder rdf:type romain:Work_order .
        ?worder romain:isAbout ?event .
        ?worder romain:ROM:_cumulative_hours ?val. FILTER(?val<1600).
}
```

worder	event	val
WO16	Event16	"15155"^^<http://www.w3.org/2001/XMLSchema#integer>
WO5	Event5	"7683"^^<http://www.w3.org/2001/XMLSchema#integer>
WO14	Event14	"14788"^^<http://www.w3.org/2001/XMLSchema#integer>

How many replace orders were executed before the expected replacement interval <x> of operating hours?

What overlaps do you see with other ontologies?



ROMAIN provides a unifying framework that can be used in conjunction with other BFO compliant sub-domain ontologies, such as planning and scheduling ontologies.

What are the (main) relations in your ontology?

- ROMAIN uses relations imported from the Relations Ontology (RO)
  - RO is a collection of relations, in compliance with BFO, intended primarily for standardization across ontologies in OBO Foundry.
  - Some examples of relations:

RO: bearer of	a relation between an independent continuant (the bearer) and a specifically dependent continuant (the dependent), in which the dependent specifically depends on the bearer for its existence.
RO: has role	a relation between an independent continuant (the bearer) and a role, in which the role specifically depends on the bearer for its existence.
RO: has function	a relation between an independent continuant (the bearer) and a function, in which the function specifically depends on the bearer for its existence.
RO: has part	a core relation that holds between a whole and its part.
RO: is about	A primitive (i.e. undefined) relationship between an information content entity and some entity.
RO: prescribes	For all classes T1 and T2, if T1 prescribes T2, then there is some instance of T1, t1, that serves as a rule or guide to some instance of T2, t2 (if T2 is a type of BFO:occurrent) or that serves as a model for some instance of T2, t2 (if T2 is a type of BFO:continuant)..
ROM: has subject	a relation between an occurrent and a dependent continuant in which the dependent continuant is the reason to the appearance of the occurent.
RO: affects	If p is a process and c is a continuant, then p affects c if and only if p influences c in some manner, most often by producing a change in c.
RO: describes	For all classes T1 and T2, if T1 describes T2 then there is some instance of T1, t1 that presents the characteristics by which some instance of T2, t2 can be recognized or visualized.
RO: has output	a relation between a processual entity and a Continuant such that the presence of the Continuant at the end of the processual entity is a necessary condition for the completion of the processual entity.
RO: participates in	a relation between a continuant and a process, in which the continuant is somehow involved in the process.
RO: proceeds	a relation between two occurents. Occurent x precedes occurent y if and only if the time point at which x ends is before or equivalent to the time point at which y starts.

What is the knowledge your specific ontology represents?

- We use a hybrid approach, based on a top-down alignment to an open source top-level ontology, the Basic Formal Ontology (BFO) adopting the realism view, and a bottom up focus on classes that are grounded in maintenance practice.

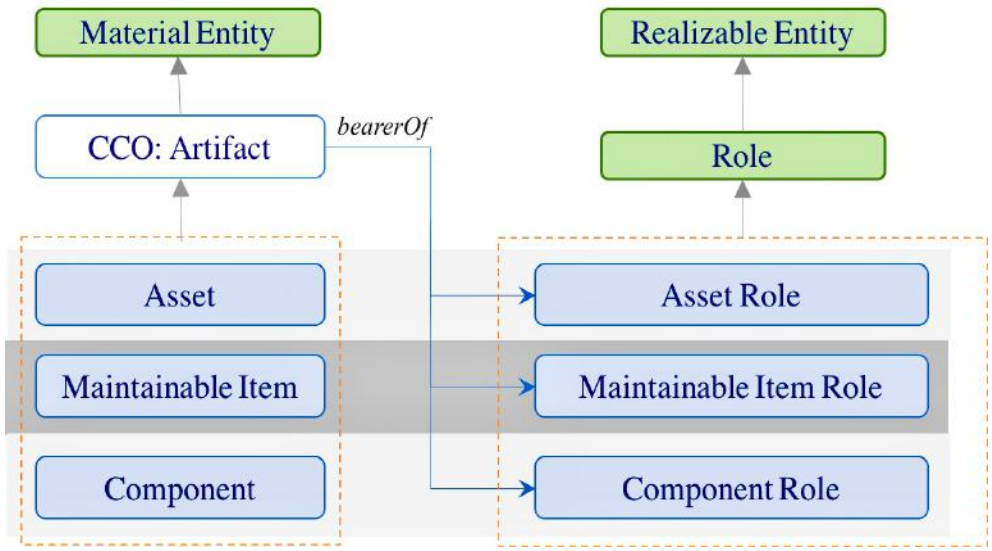
- We constrain the scope of the ontology to the classes that are unique to the maintenance management practice, such as maintenance strategy, degradation, and work order management, rather than modelling the entire domain of maintenance.

How does your ontology represent the relations between different granularity views on the same object?

- parthood relations
  - ***has part*** - a core relation that holds between a whole and its part.

- A ***Component*** *is-an* ***Artifact*** that bears a ***Component Role***. ***A Component Role*** *is-a* ***Role*** that is borne by an ***Artifact*** when it become a *part-of* another ***Artifact***, to provide a particular ***Function***.

How does your taxonomy and/or ontology capture materials?



- Material entities ***(ME)*** are continuants that include some portion of matter as part. ***[in BFO, ME is not the same as material from chemistry point of view.]***
- Every ME is localized and move in space.
- Every entity which has a ME as part is a ME.

- ROMAIN deals with machines (equipment, item, asset).
  - An equipment is a ***ME***. It can be seen as a whole.
  - It has parts different components ***(ME).***
  - All these ***ME*** include portion of matter (quarks and leptons, including electrons, etc.)
- An entity may change its role at some stages in its life
  - The same artifact can has the ***product role*** in its ***manufacturing phase*** and has the ***asset role*** when it is used ***in production***.

How does your ontology represent processes?

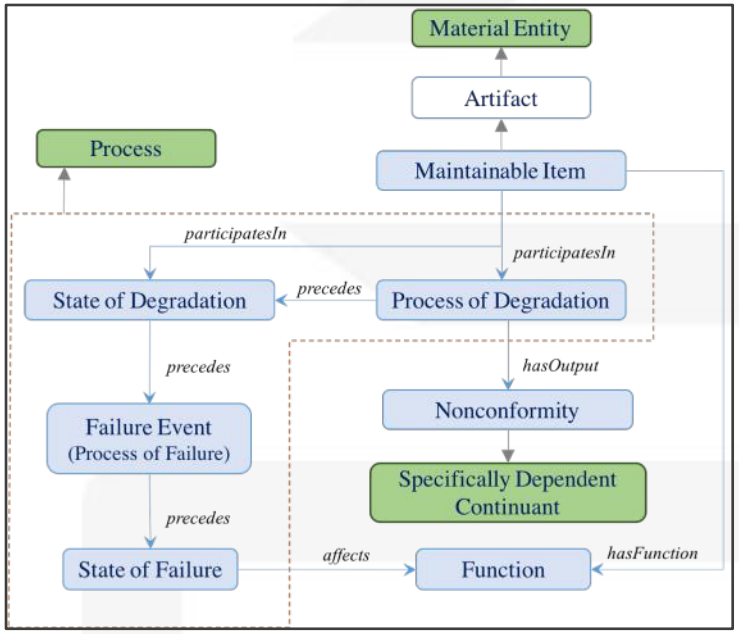


Fig 1.

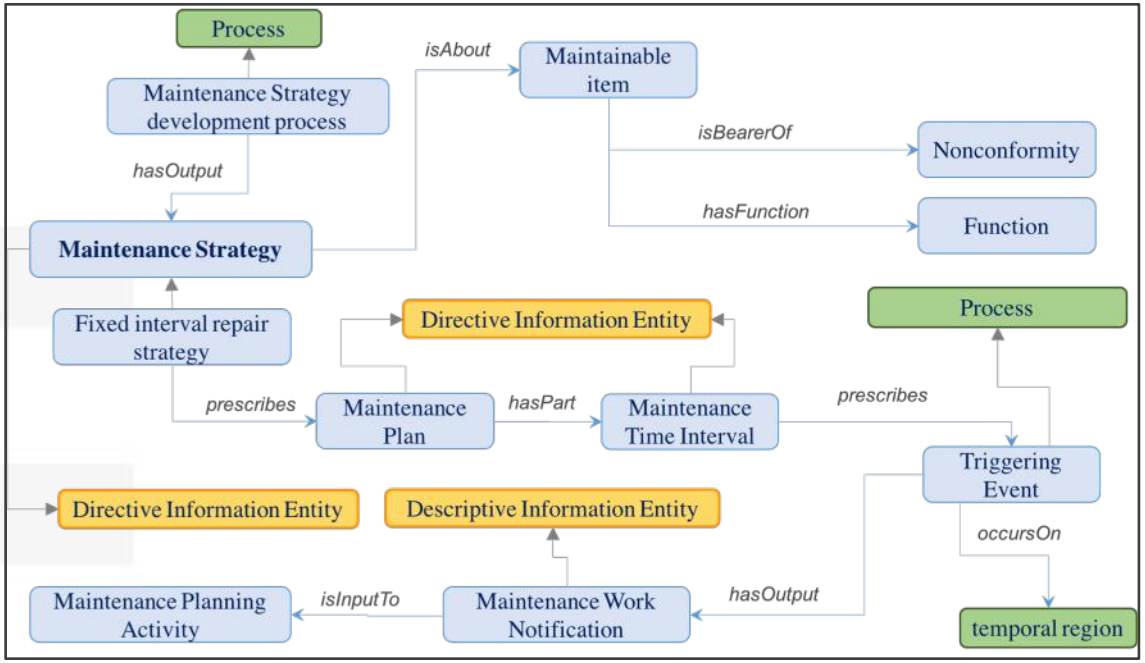
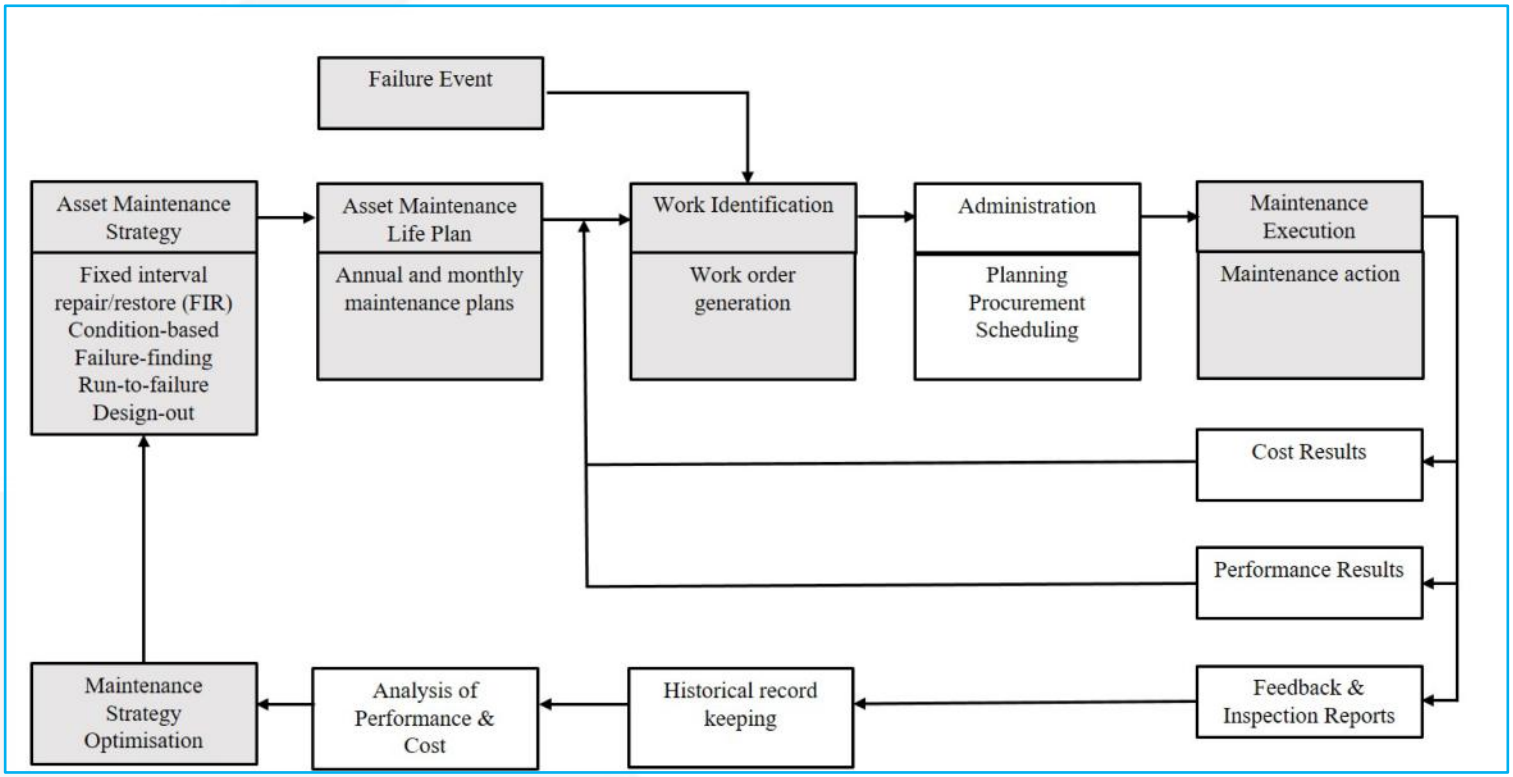
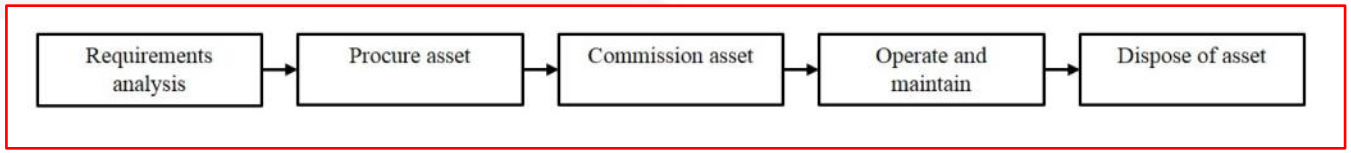


Fig 2.

A ***process*** *is an* ***occurrent*** entity that exists in ***time*** by occurring or happening, has ***temporal parts***, and always depends on at least one ***material entity***. It can be partitioned into temporal parts in different ways and at different levels of granularity.

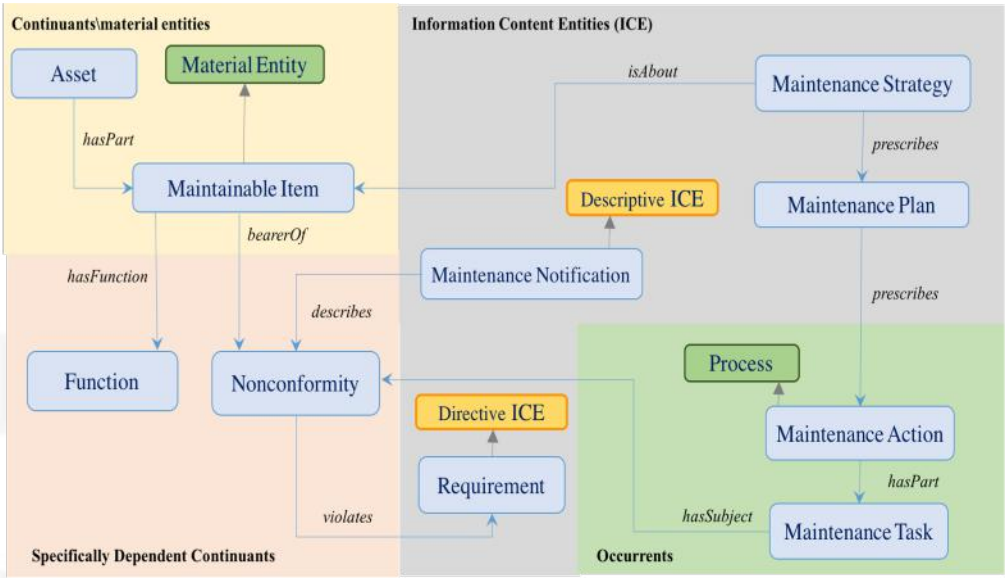
How does your ontology capture manufacturing?





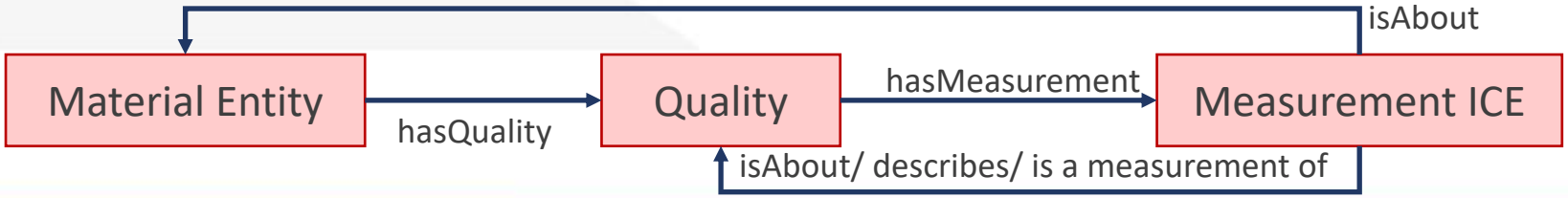
How does your ontology address the circle connection between physical properties, materials models and measurement?

- Models are considered as Descriptive ICE
- Nonconformity is non-fulfilment of a requirement or an expectation:
  - Nonconformity is formally defined as a specifically dependant continuant that inheres in a material entity and deviates from a design specification (***models***).
  - It can be a loss of quality, function, or capability (including the change of physical proprieties) in an item that results from a degradation.
  - A non conformity can be observed by a ***measurement***



Properties of **material entities** in BFO are presented through ***specifically dependent continuant***: it includes **quality**, **disposition**, **function**, **role**. (e.g. Mass, weight, temperature, etc.)

Specifically dependent continuants are described through ***information content entities*** such as ***measurement ICE*** *that* ***has unit*** *and* that consists of a symbol that is a measure of the extent, dimensions, quantity, or quality



What is the representation language and implementation (logics)?

- OWL-DL