Working Session Agenda

IOF Production Planning and Scheduling Working Group

OAGi

IOF

Working Session Agenda

Thursday, 2/8, Breakout Room 2

- 16:00-16:30 – Raytheon use case(s) - William Mandrick
- 16:30-17:00 – Ontology of future entities and digital artifacts – Dusan Sormaz
- 17:00-17:20 – Theory of planning – Arkopaul Sarkar
- 17:20-17:30 - Production Planning and Scheduling WG Roadmap – Dusan Sormaz
- 17:30-18:00 - Discussion

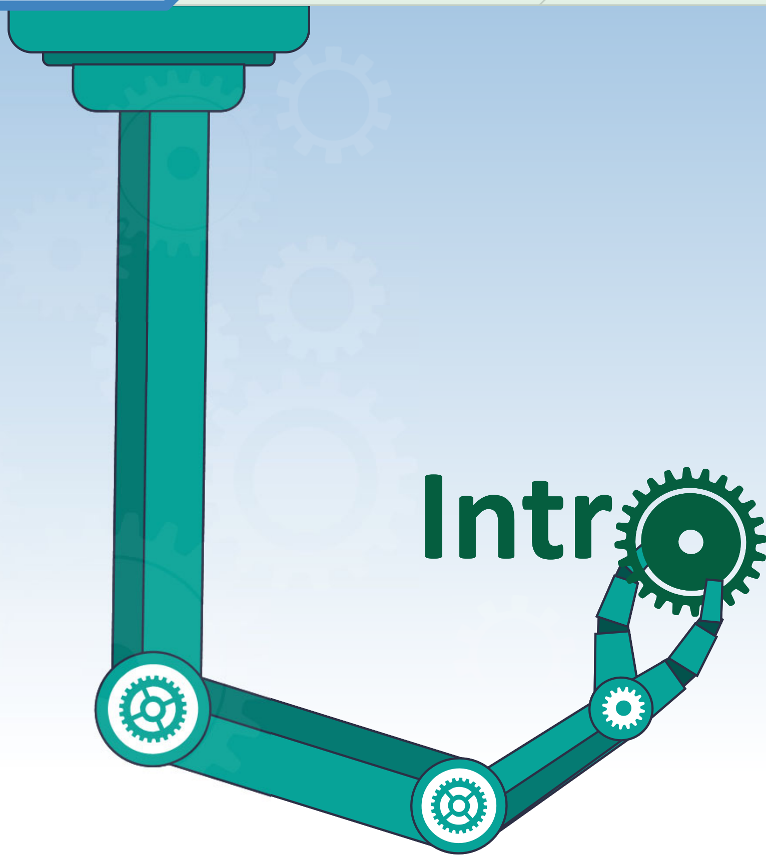
Ontology of Future and Digital Artifacts

IOF Production Planning and Scheduling Working Group

Prepared by Dusan Sormaz
and Saruda Seeharit

OAGi

IOF



Introduction

Motivation and Background (Modal Logic)

- Modal logic is a kind of logic used to represent statements about necessity and possibility. It plays a major role in philosophy and related fields as a tool for understanding concepts such as knowledge, obligation, and causation. [Wikipedia]
- A modal is an expression (like 'necessarily' or 'possibly') that is used to qualify the truth of a judgement. Modal logic is, strictly speaking, the study of the deductive behavior of the expressions 'it is necessary that' and 'it is possible that'. [Stanford Encyclopedia of Philosophy]
- Temporal logic (a kind of modal logic) may be interesting, because design (digital artifact) precedes real artifact
- If I have real engine it is necessary that its design existed (or exists)
- If I have engine design it is possible that real engine exists

Few Observations

- Modal relations talk about necessity and possibility, they do not show how one object make another possible
- Counterpart relation talks about similarities, but in possible worlds, so designed engine and real engine are similar, one is in real world, another in design world (possible)
- Counterpart theory provides modeling tools to define ontologies of literature, film, music and such, in all of these we have objects, natural and artifacts, they have their properties, and are connected by relations. It is not possible to describe all those relations in terms of ICE
- During design phase engineers talk about possible worlds (instances) which may exists in the future
- Engineering product development cycle may be seen as going through different worlds: Reqs, Design, Planning, Scheduling, Manufacturing, Use, Recycle

Requirements

- **General Requirements**
  - ❑Scalability ex. as-designed and as-manufactured
  - ❑Number of constructs (classes, relations, and individuals) required
  - ❑Information retrieval efficiency
  - ❑IOF/BFO (Basic Formal Ontology) compliance
  - ❑Digital twin modeling for a product/process/system

- **Use Case Requirements**
  - ❑Structures and components
  - ❑Process Settings
  - ❑Participants
  - ❑Required attributes (equipment capabilities)
  - ❑Changes in designs from each version
  - ❑Process plan (the implementation of optional steps)

Requirements for Representation

- Evaluation of design
- Performance estimation
- Simulation of ~~behavior~~ interactions (process)
- Extensibility of the design into new generations
- Communications
- Users of information, designers, managers
- Marketing
- Interoperability of designer tools
- Product life cycle: design for reuse, recycle
- Verification and validation of designs

Requirements for Representation

- Evaluation of design
- Performance estimation
- Simulation of ~~behavior~~ interactions (process)
- Extensibility of the design into new generations
- Communications
- Users of information, designers, managers
- Marketing
- Interoperability of designer tools
- Product life cycle: design for reuse, recycle
- Verification and validation of designs

APPROACHES

Approaches for Future/Digital Artifacts

ICE
Information Content Entity

Use ICE to represent all information (knowledge, decisions) about the future artifacts

R/S
Representation and Specification

Use R/S approach (given in a paper by Sarkar and Sormaz), specialize ICE to have Representation and Specification as subclasses

MRO
Modal Relation Ontology

Use MRO approach to represent future artifacts, based on replica of relations (from BFO or any) into Modal relation space

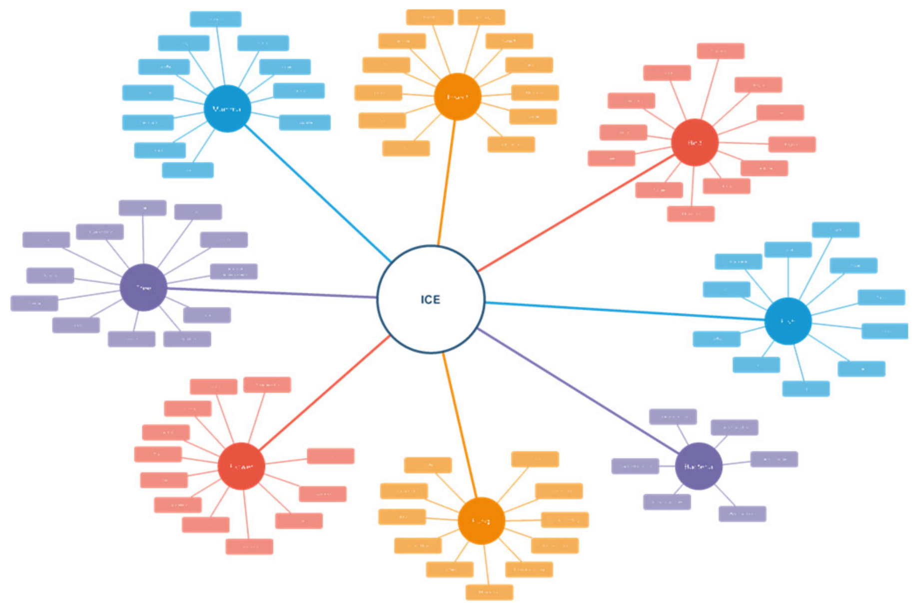
CR
Information Content Entity

Use CR approach, which is motivated by MRO approach but connects relations and provides for new relations between designed/planned entities and actual entities

Sormaz, D. , Kulvatunyou, B. , Drobnjakovic, M. , Seeharit, S. and Sarkar, A. (2023), Comparison of Ontological Representations of Relations between Digital and Physical Artifacts In Manufacturing Domain, Proceedings of the ASME 2023 IDETC Conference, Boston, MA, US,

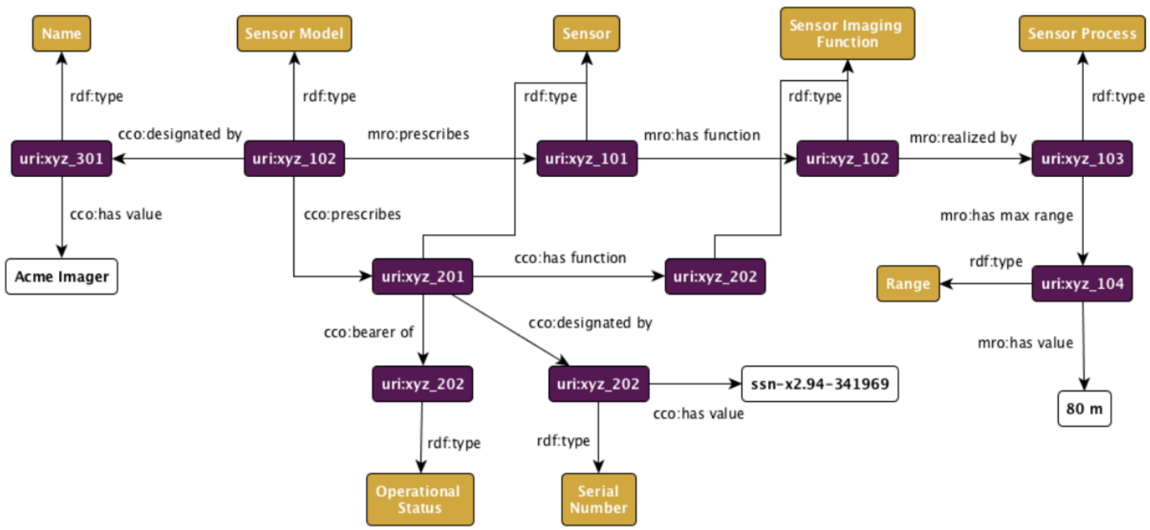
Information Content Entity

- Relies on strict decoupling between physical and digital entities.
- Digital artifacts use specific subclasses of IOF Information Content Entity (ICE) class.
- Individual ICEs reference existing physical entities and potential future ones via class axioms.
- Interrelations of attributes, structures, and precedence for future physical entities captured by:
  - Relations of corresponding ICEs.
  - Relations based on related universal axioms.



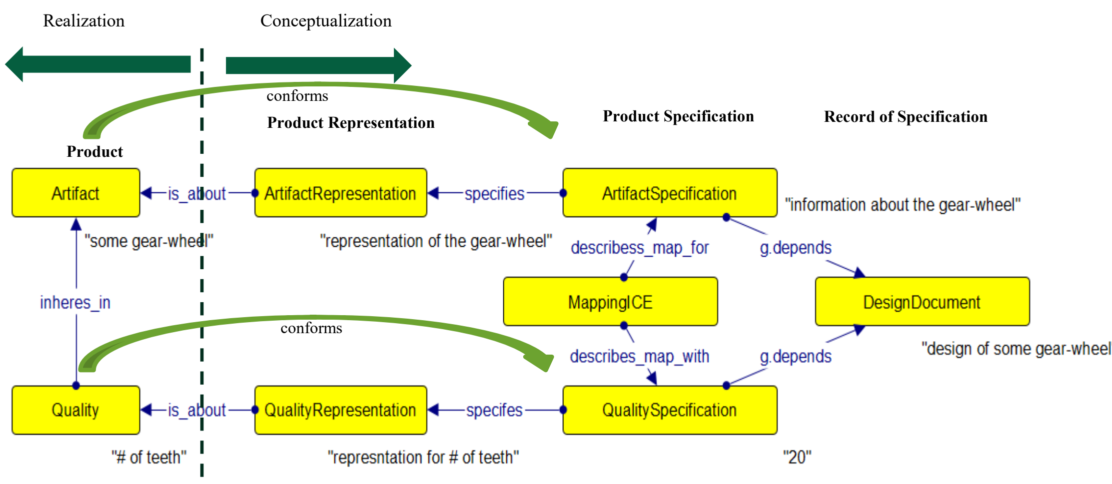
Modal Relation Ontology

- Conceived within CCO effort.
- MRO represents non-existent or potential future states.
- Future states like planned actions or artifact functionalities.
- Benefits in comparing actual instances with plans/specifications for missions, sensors, and assets in military context.
- This approach uses two namespaces for relations: CCO and MRO.
- CCO relations for actual entities (material artifacts and processes).
- MRO relations for planned or future entities (material artifacts and processes).



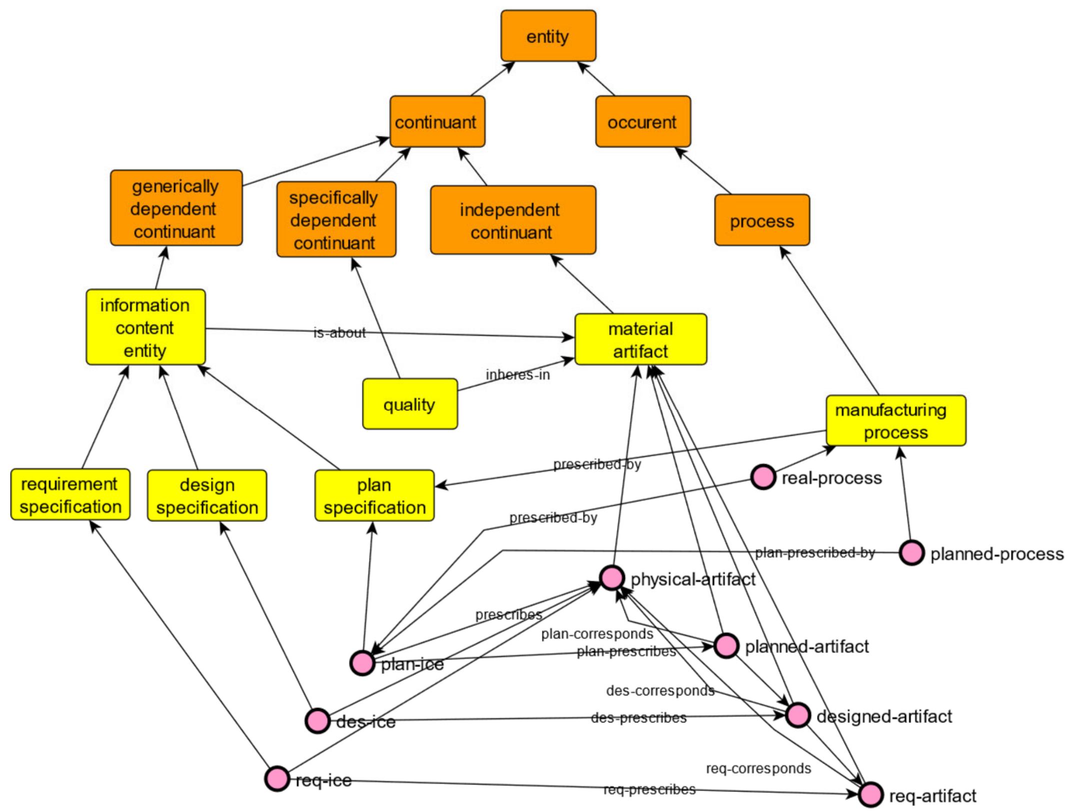
Representation and Specification

- Based on BFO's stance that only physical
- (real) entities can be represented.
- Independent Continuant or Occurrent
- branches for representing physical entities.
- Engineering design and process planning
- outcomes represented through various
- Information Content Entities (ICEs).
- ICE developed as part of CCO, now in IOF Core specification.
- Product design as a list of specifications guiding physical product and manufacturing processes.
- Two concepts used: representation and specification
- MapICE assigns specific roles to the two specifications.



Counterpart Relation

- Based on product life-cycle phases: requirement, design, planning, manufacturing, usage, end-of-life.
- Engineering work focused at start, needs to predict usage and recycling.
- Modern tech creates digital artifacts across stages before physical production.
- Concept of 'product possibilities' and modal relations used in CR approach.
- Represents projected phases and final physical artifacts.
- Engineering tasks done before manufacturing and use.



Requirements for Representation

- During design (planning) we should present and preserve all semantic relations for artifacts and planned processes
  - Has component part
  - Has quality
  - Has participants
  - Duration
  - Subprocesses
- We should be able to reason if real artifact satisfies design
  - Real artifact should have all components as designed
  - Real artifact should have all qualities as designed with satisfaction of values and their constraints (min, max, range)
  - Real artifact can not have extra components or qualities
  - This applies on each component recursively

Analysis

Case Study

An engineering task:

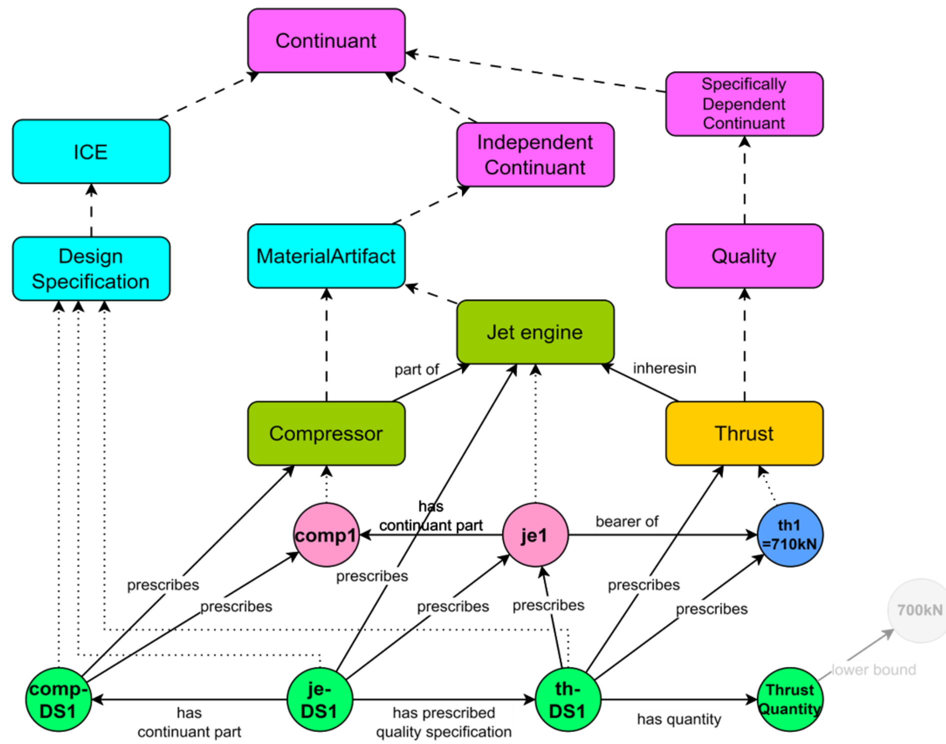
"There is a need to design and produce a jet engine that will have a compressor as its part, and it will be able to produce a minimal thrust of 700 kN".

This simple example provides sufficient elements to compare the approaches.

ICE Approach
(Information Content Entity)

Observations:

- the digital world is completely constrained to ICE
- new properties are not needed since all properties point to universals
- some plans can only be used to analyze the future physical process
- number of properties will increase as the number of relations between physical world entities increase
- introduces new property for linking and comparing various specifications of products and processes
- requires introduction of new classes that account for the specifics of each product or process created
- reasoning cannot fully be applied to expressions pointing to a universal

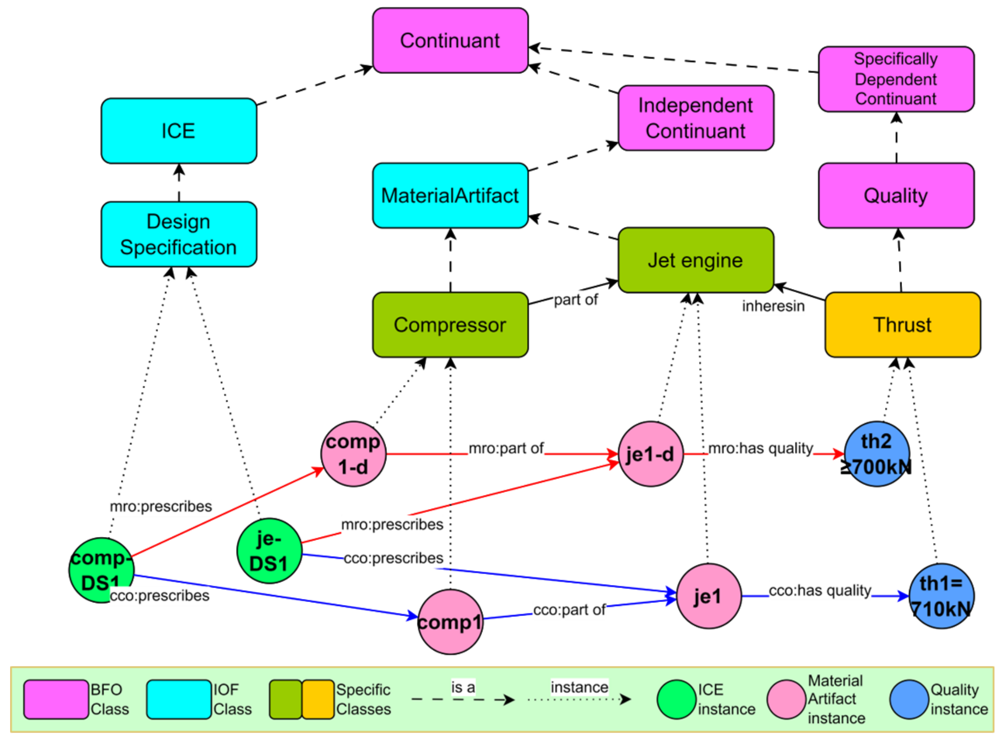


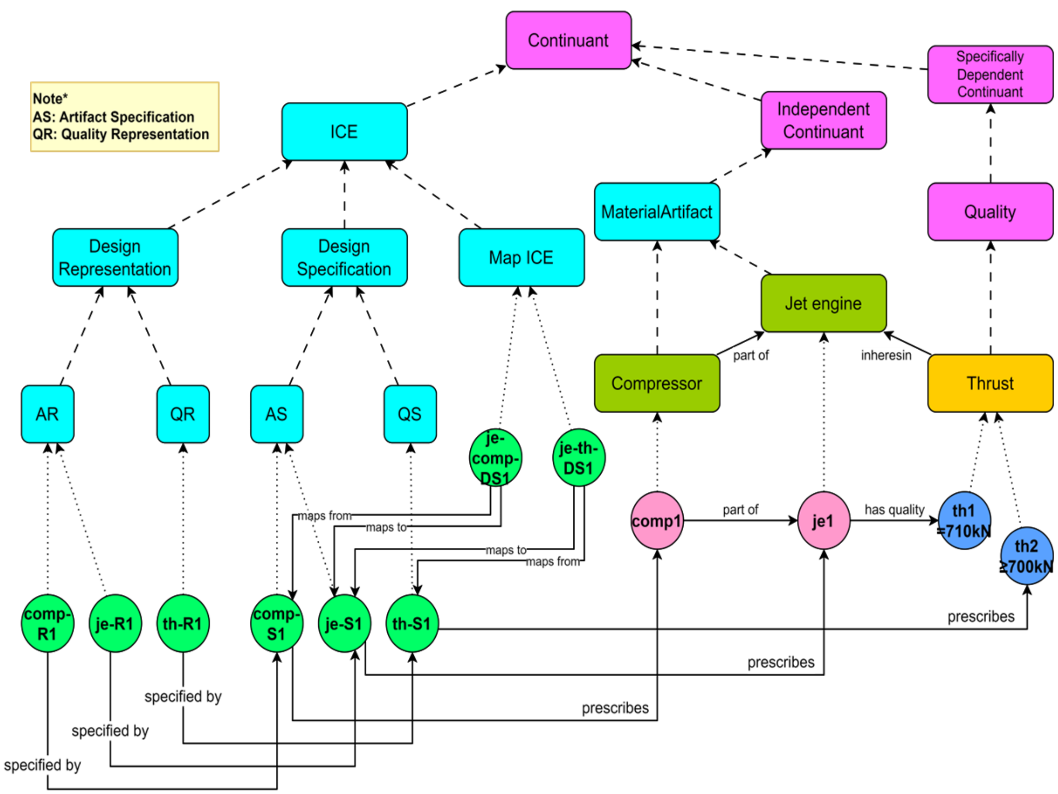
MRO Approach

(Modal Relation Ontology)

Observations:

- no classes for these different, new, or additional information models, renders the current ontologies deficient in their semantic definitions
- maintenance or extensions requires significant revision
- changes and updates are not linked automatically requiring humans to help link them



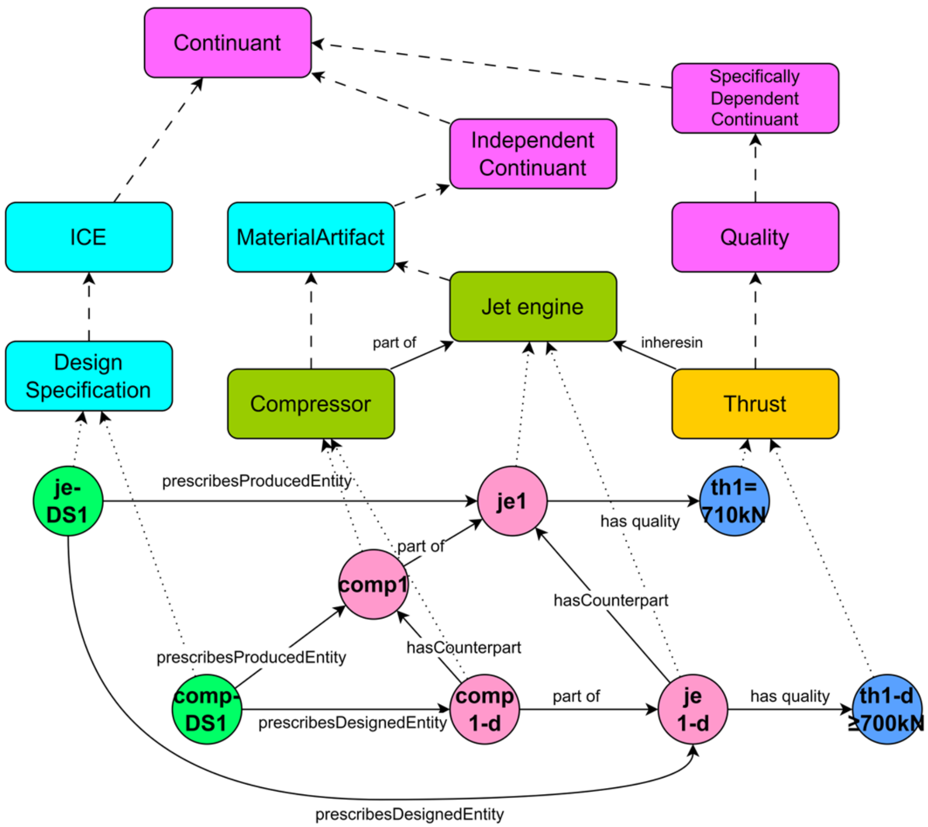


R/S Approach

(Representation and Specification)

Observations:

- includes considerations of the design process, design intent, and mental representation of design
- uses MapICE entity to relate several versions of the same design
- introduces many classes, relations, and instances
- introduces many more classes, relations, and instances
- omits the semantics of various relations between the physical artifacts and their qualities
- uses software as a replacement for relations between different classes or instances

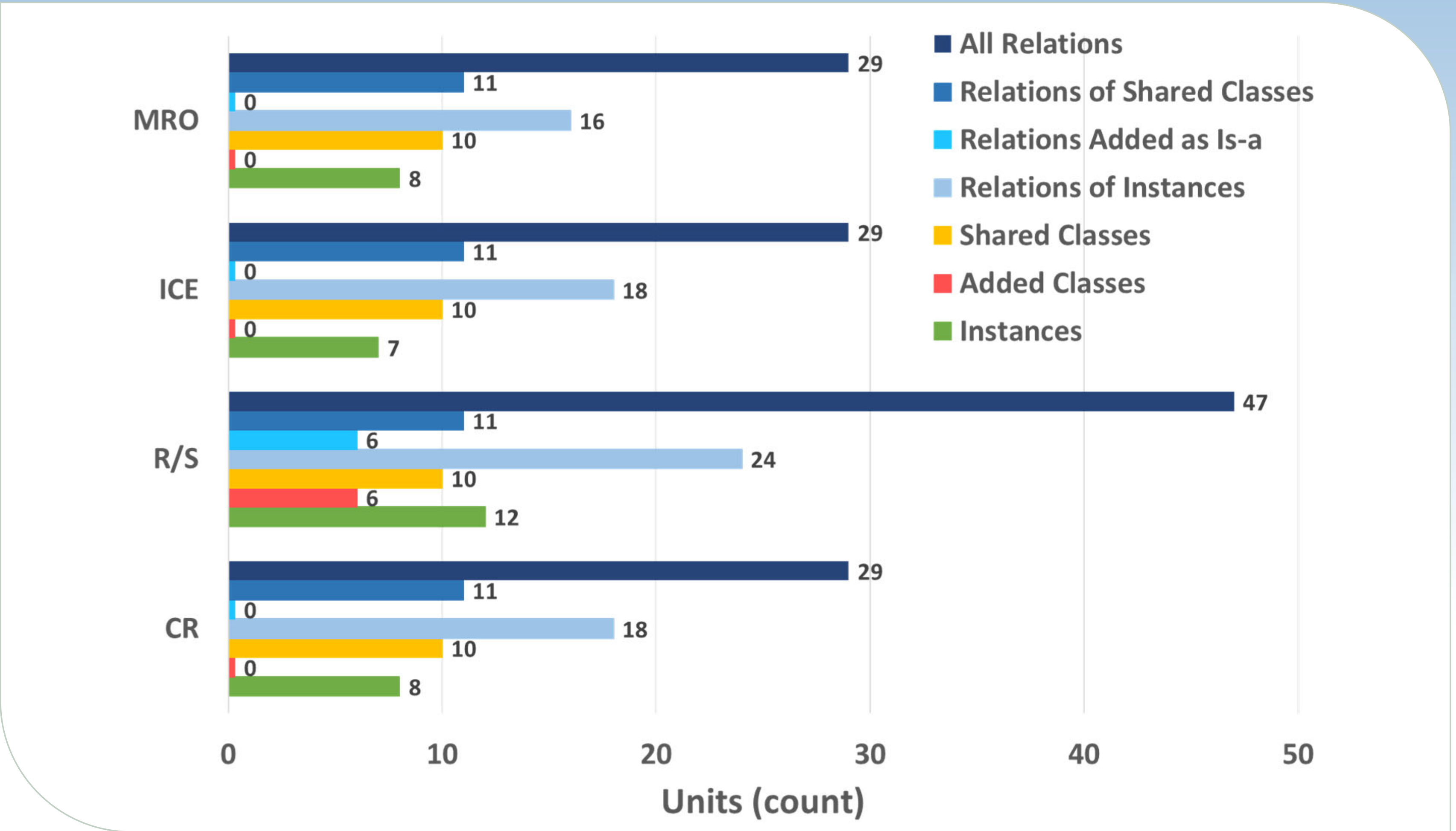


CR Approach

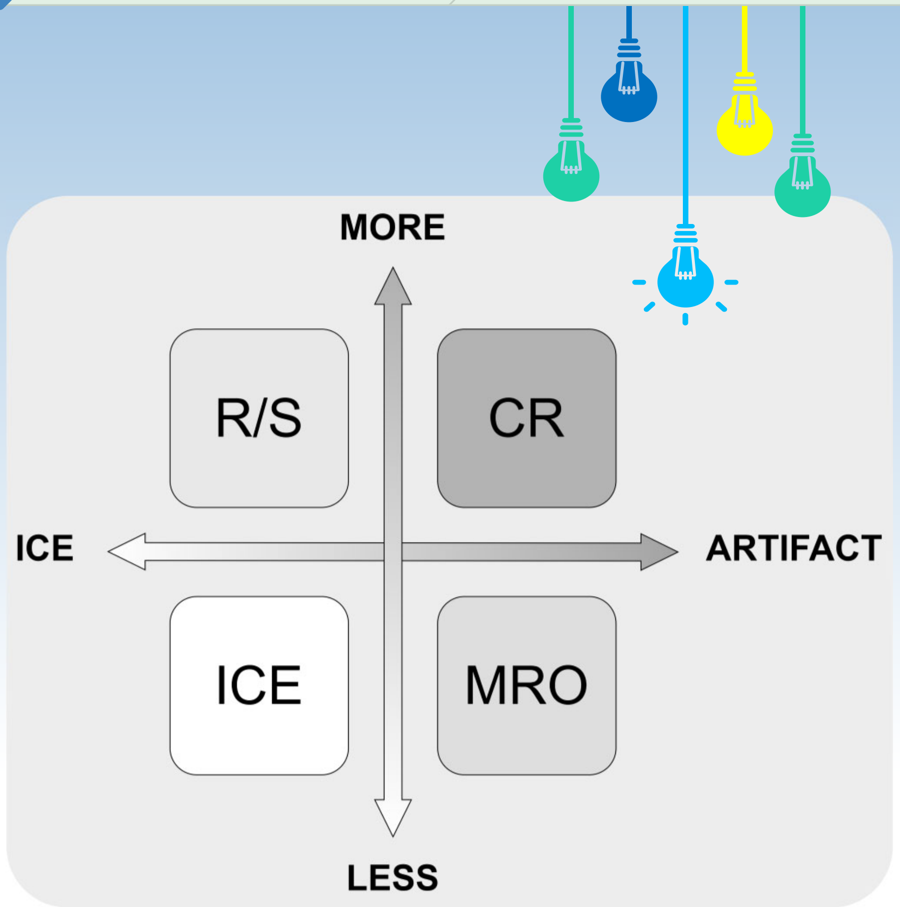
(Counterpart Relation)

Observations:

- uses concise natural languages used by engineers and designers
- requires far fewer classes and entities than the previous two approaches (ICE and Spec/Rep)
- enables proper and simple semantic reasoning about digital artifacts
- connects digital and physical artifacts, by connecting them to the single ICE that prescribes both
- able to compare attributes of digital, designed or planned artifacts with their physical counterparts
- requires a new ontology to differentiate between current and future or virtual physical artifacts


All Relations
Relations of Shared Classes
Relations Added as Is-a
Relations of Instances
Shared Classes
Added Classes
Instances
MRO
29
11
0
16
10
0
8
ICE
29
11
0
18
10
0
7
R/S
47
11
6
24
10
6
12
CR
29
11
0
18
10
0
8
0
10
20
30
40
50
Units (count)

- Key differences among the approaches are summarized below using criteria:
  - Digital artifact class
  - Development level
- ICE and Rep/Spec approaches are distinct, using ICEs for representation, while the other two consider them physical artifacts.
- MRO and Counterpart approaches seem more efficient for industrial representation and reasoning.
- Rep/Spec and Counterpart approaches offer detailed treatment of various digital artifact types.
- Rep/Spec and Counterpart stand out in development level criterion.
- Additional testing and analysis are needed to determine the most suitable approach.



Problems with ICE and Related Approaches

- We can have only ICEs during the design or planning processes
- We lose semantics
  - Artifacts have several relations: hasContinuantPart, hasQuality, Particiaptes in, …
  - ICEs have only one relation between each other, hasContinuantPart (excluding isAbout and its subproperties)
- We need to define new ICE classes for all classes of artifacts/processes
  - To perform reasoning about future artifacts (that do not exist) we add the parallel hierarchy into ICEs to capture the knowledge about artifacts, their qualities, parts, processes, etc.
- Limited reasoning and representation on individual level
  - Reason is that artifacts do not exist
  - We can reason and query only on class level

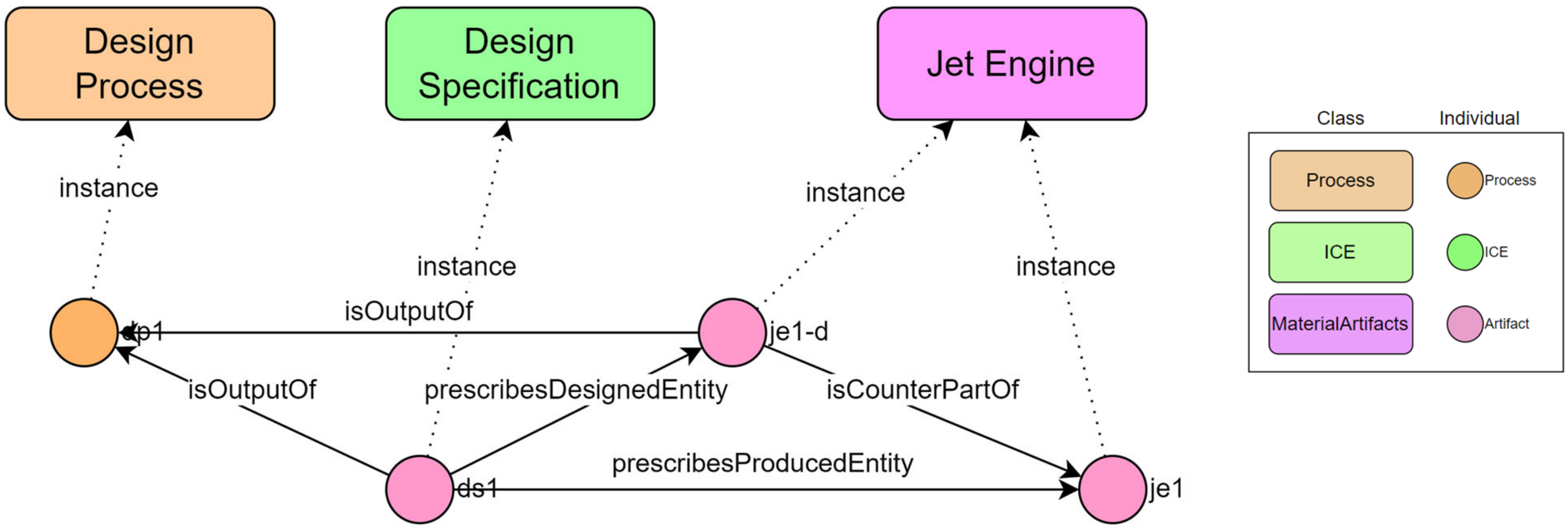
Problems with MRO Approach

- Looks like a fast (easy, dirty) way to be able to run SPARQL queries on a knowledge graph
  - Lot additional work is needed to run intended SPARQL queries

- There is not connection between cco and mro relations
  - Implications is that we can not use mro relations to get the same benefits as in cco relations
  - Reasoner can not jump over from mro space to cco space

SOLUTION

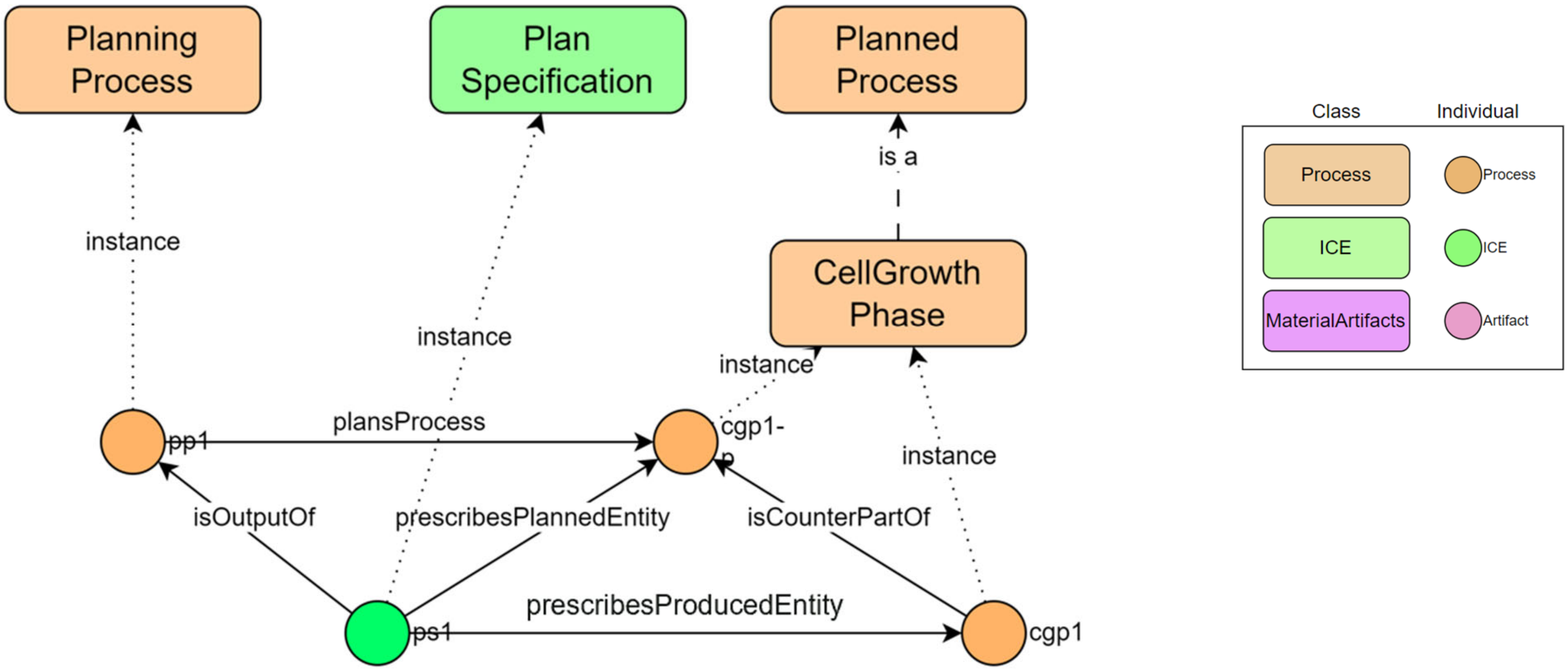
Design Artifact and Actual Artifact



There is timeline here:

Design artifact starts its existence before actual artifact

Future Process and Actual Process



Ontology of Future and Digital Artifacts

- Use subclasses and instances of Artifacts to represent design
- Use subclasses and instances of Planned Process to represent future planned processes
- Design specification prescribes both designed and actual artifacts
- Plan specification prescribes both planned and actual processes
- We subclass prescribes relation to distinguish between different prescribed entities
- Subclasses:
  - prescribesProducedEntity
  - prescribesDesignedEntity
  - prescribesPlannedEntity
  - prescribesRequiredEntity

Relations Between Future and Actual Entity

- Introduce new relation isCounterpartOf/hasCounterPart that will connect future (designed or planned) and actual entities

- IsCounterPartOf is relation between an actual entity and another entity that is prescribed by the same Design specification or Plan specification
- hasCounterPart is inverse

- These two relations serve purpose of controlling the quality of the artifact and the monitoring the procedure of the actual process

- Do we need to specialize them? May be if necessary, their semantic does not change, just domain and/or range become reduced

Definitions of Subclasses of prescribes

- precribesActualEntity – relation from an ICE to an entity such that ICE prescribes entity which exists in its physical or natural environment (world)
- prescribesDesignedEntity – relation from an ICE to an entity such that ICE prescribes entity which does not exist, but is result (output) of a design process
- prescribesPlannedEntity – relation from an ICE to an entity such that ICE prescribes entity which does not exist, but is result (output) of a planning process

Design and Planning Process

- Design process is a planned process
- Product Design Process is a planned process with an objective to design new continuant
- Process Design Process is a planned process with an objective to plan a new process, sometimes called planning process

Jet Engine Use Case

Jet Engine Design and Requirement Verification

An engineering task:

"There is a need to design and produce a jet engine that will have a compressor as its part, and it will be able to produce a minimal thrust of 700 kN".

This simple example provides sufficient elements to compare the approaches.

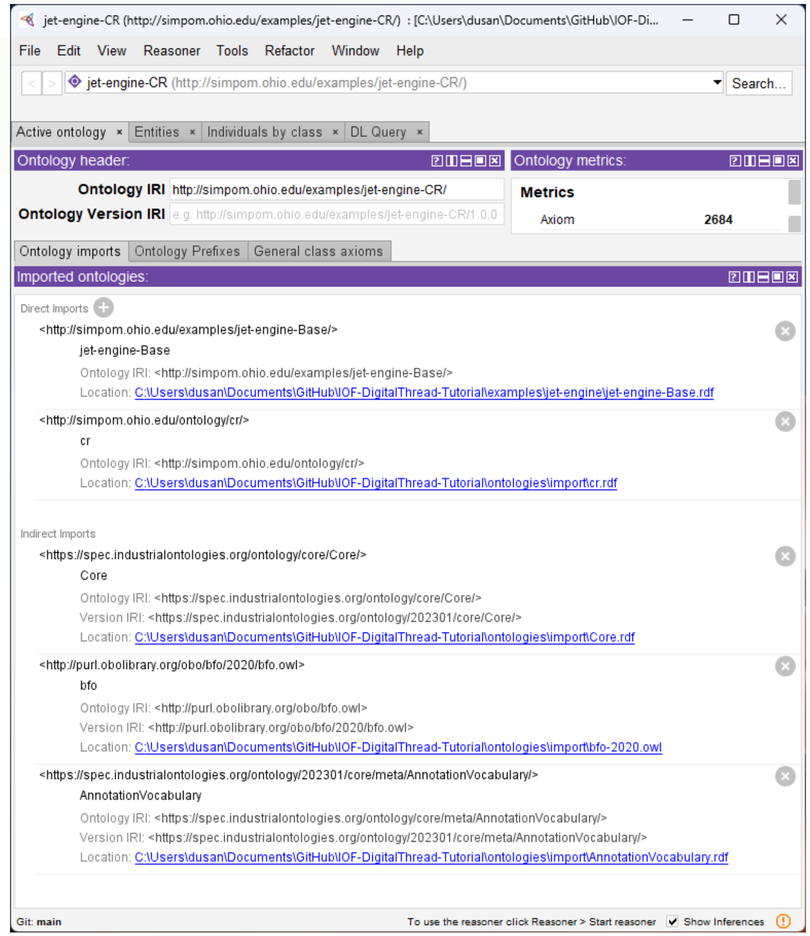
Jet Engine Competency Questions

a. What is the thrust of a real engine?
b. Does a real engine have at least the same thrust as its design?
c. Does a ball bearing of a fan assembly in a real engine meets the load requirement?
d. Does a turbine diameter of real engine have value as specified in the design?
e.
f.
g. Does a real engine have all the designed components and subcomponents and nothing extra?
h. Does a real engine have any extra components or subcomponents not in the design?
i. Which engines are built as designed?
j. Which (real) engines failed design specification?
k. Which revisions of the design meet a revision of its requirements

Does a real engine have at least the same thrust as its design?

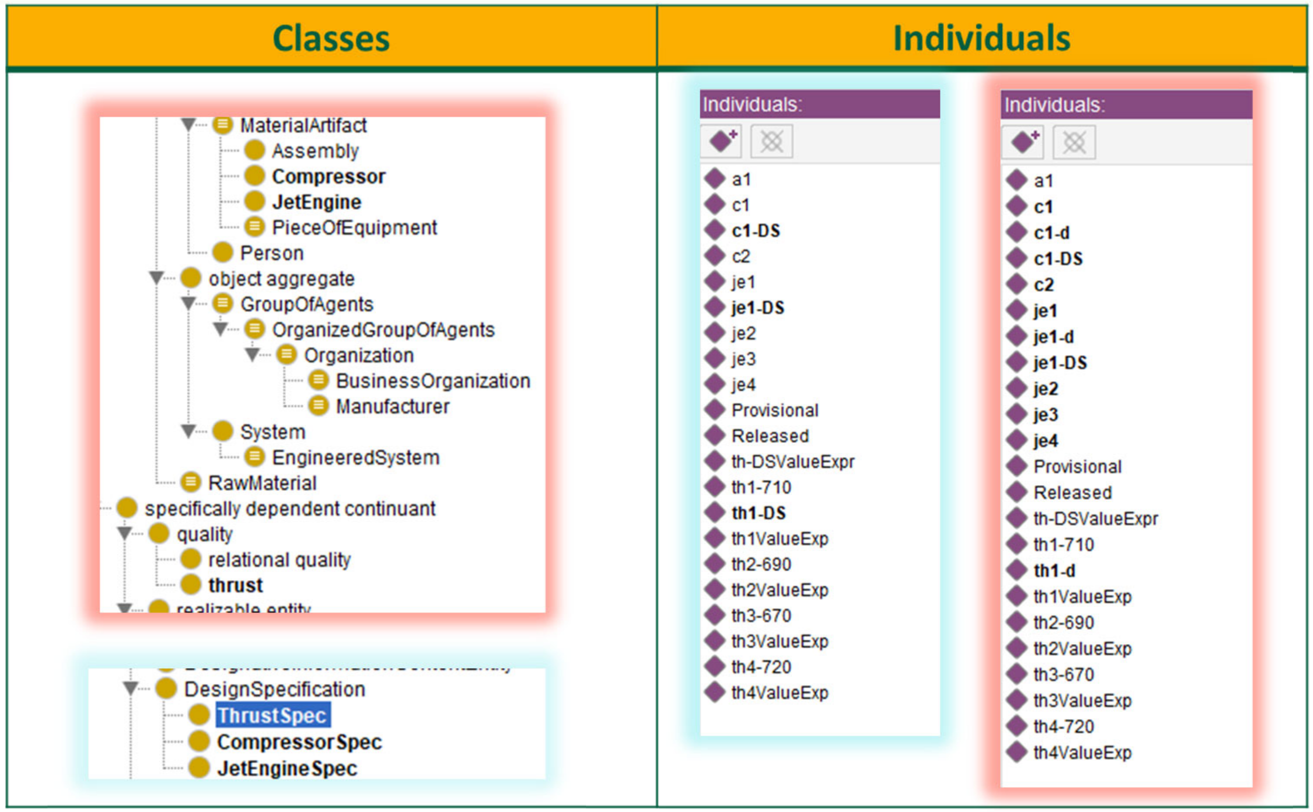
Import hierarchy

- jet-engine-cr.rdf imports
  - jet-engine-base.rdf
  - cr.rdf
- jet-engine-base imports
  - IOF Core
- IOF Core imports
  - bfo
  - AnnotationVocabulary



Results in Comparing ICE and CR approaches

- We ran about 10 SPARQL queries both in ICE approach and CR approach
- ICE approach has different property paths in design artifacts from actual artifacts to capture data
- ICE approach requires introduction of several subclasses of DesignSpecification



SPARQL Query b – CR approach

Does a real engine have at least the same thrust as designed? (minimum of 700 kN)

```
SELECT ?engine ?thrust ?realthrustValue ?spec ?engdesign ?thrustdesign ?thrustDesignValueExpr ?isSatisfactory
  WHERE {
    ?engine rdf:type jeb:JetEngine.                                      # find engine and its thrust
    ?spec cr:prescribesActualEntity ?engine.
    ?engine core:hasQuality ?thrust.

    ?thrust core:hasValueExpressionAtAllTimes ?thrustValueExpr.          # find the thrust value
    ?thrustValueExpr core:hasSimpleExpressionValue ?realthrustValue.

    ?spec cr:prescribesDesignedEntity ?engdesign.                        # find the engine design from associated spec
    ?engdesign core:hasQuality ?thrustdesign.

    ?thrustdesign core:hasValueExpressionAtAllTimes ?thrustDesignValueExpr.   # find design thrust value
    ?thrustDesignValueExpr jeb:hasLowerBoundValue ?designThrustValue.

    BIND((?realthrustValue >= ?designThrustValue) as ?isSatisfactory)   }     # compare them
```

OAGi

IOF

SPARQL Query b – ICE approach

Does a real engine have at least the same thrust as designed? (minimum of 700 kN)

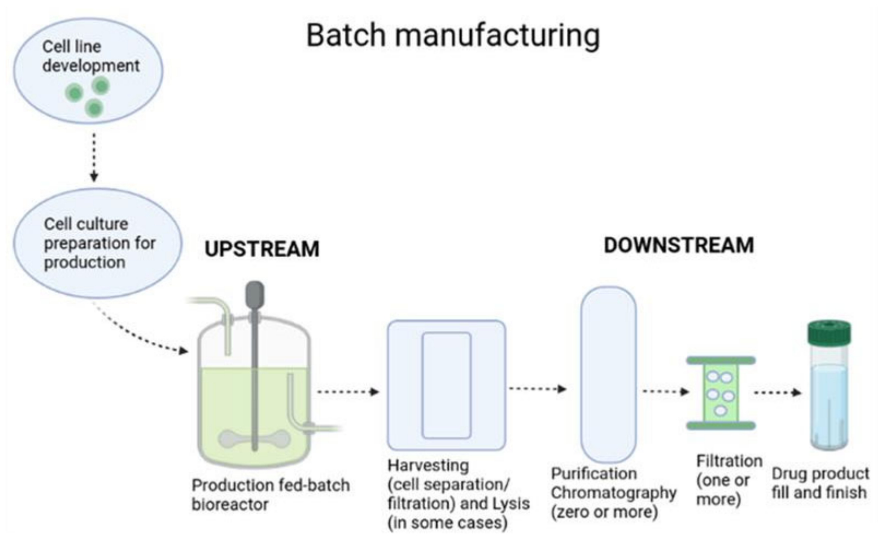
```
SELECT DISTINCT ?jetEngineDesignSpec ?jetEngine ?designedThrustValue ?realThrustValue ?isSatisfactory
    WHERE {

    ?jetEngine rdf:type jet:JetEngine.                                   #retrieve all individuals that are jet engines
    ?jetEngine core:hasQuality ?thrust.

    ?thrust core:hasValueExpressionAtAllTimes ?thrustValueExpr.          #retrieve thrust
    ?thrustValueExpr core:hasSimpleExpressionValue ?realThrustValue.

    ?jetEngineDesignSpec rdf:type jet-ice:JetEngineSpec.                 #retrieve all designs for the jet engine
    ?jetEngineDesignSpec core:prescribes ?jetEngine.
    ?jetEngineDesignSpec core:prescribes ?thrustSpec.

    ?thrustSpec bfo:BFO_0000110 ?thrustDesignValueExpr.                  #retrieve thrust specification value
    ?thrustDesignValueExpr jet-ice:hasLowerBoundValue ?designedThrustValue.

    BIND((?realthrustValue >= ?designedthrustValue) as ?isSatisfactory) }   #compare them
```

Biomanufacturing Process Use Case

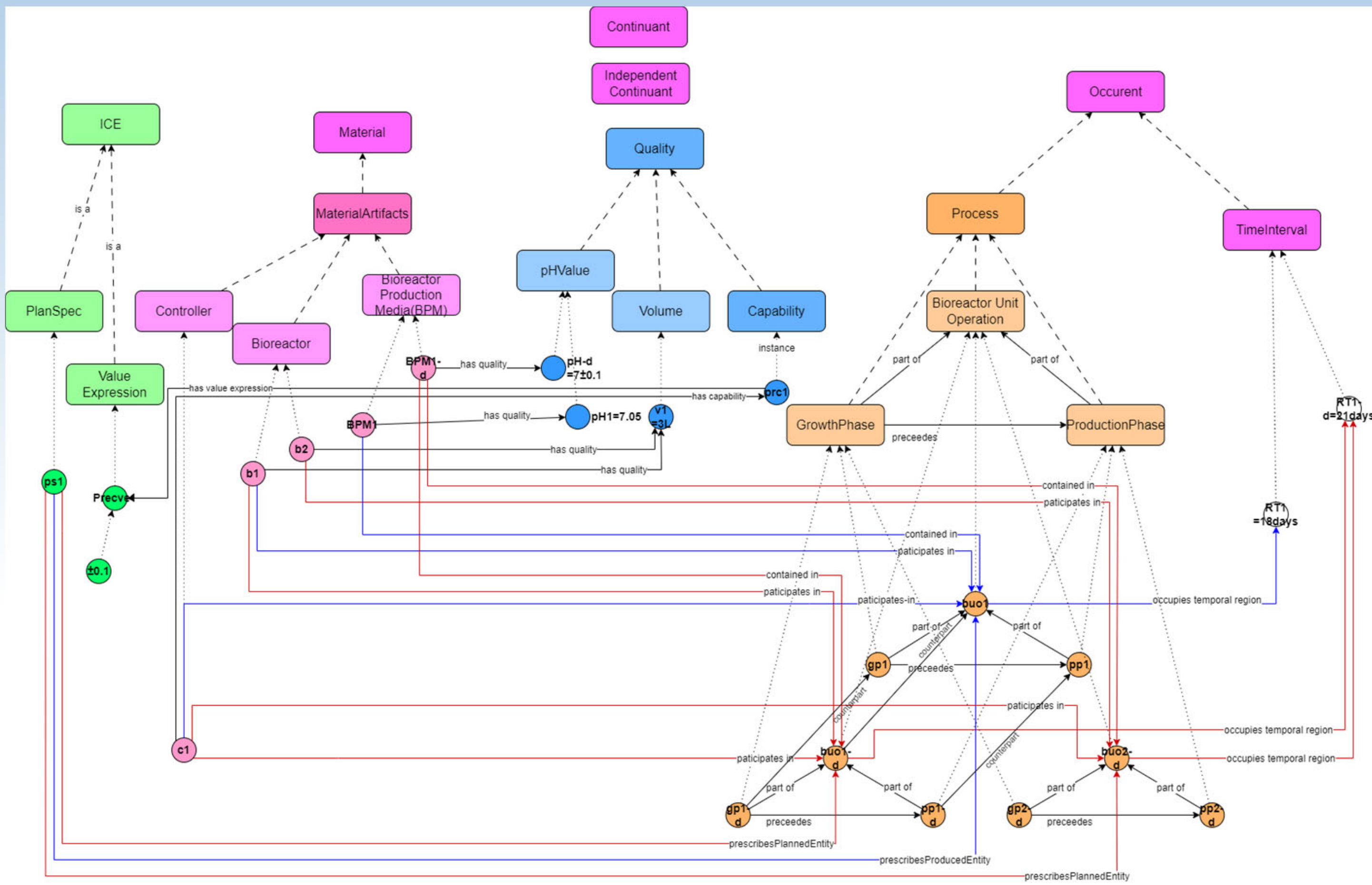
Bio Process

The fed-batch production bioreactor unit operation

- The fed-batch production bioreactor unit operation consists of two phases: growth phase and production phase. The growth phase precedes the production phase. In both phases the pH MUST be kept at 7 ± 0.1. To achieve this a pH controller is required as a process participant which has the capability to control pH up to a precision of 0.1. The process specification prescribes to use either bioreactor instance1 or instance2. In the run instance 1 was used. Both bioreactors are identical w.r.t. volume which is 3L. The unit operation duration is 21 days. However, due to some in-process complications the run only lasted 18days (we do not have to capture the in-process complications) just the duration difference.



Biomanufacturing Process – CR Approach



OAGi

IOF

Bio Process Competency Questions

A. Which equipment are used in a (real) process?

B. What is the difference of duration between actual and planned process?

E. What is the concentration of a drug substance (or unwanted substance) after a unit operation O1 compared to

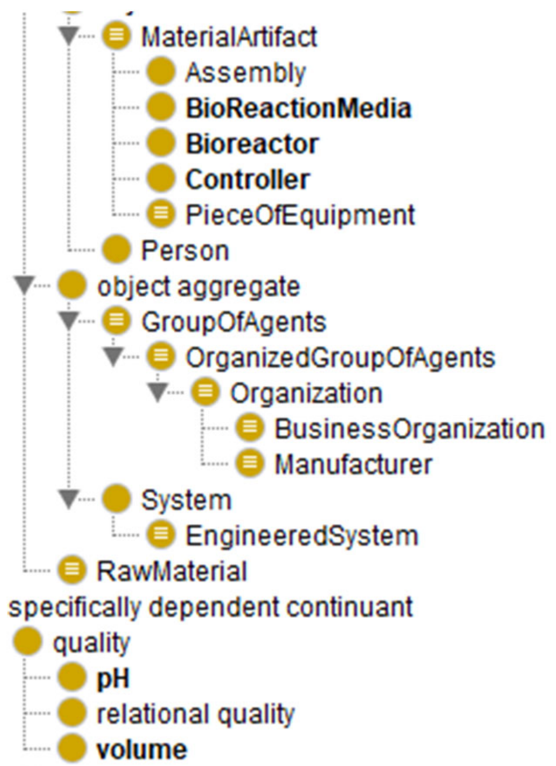
within the control strategy? Is the difference within the plan?

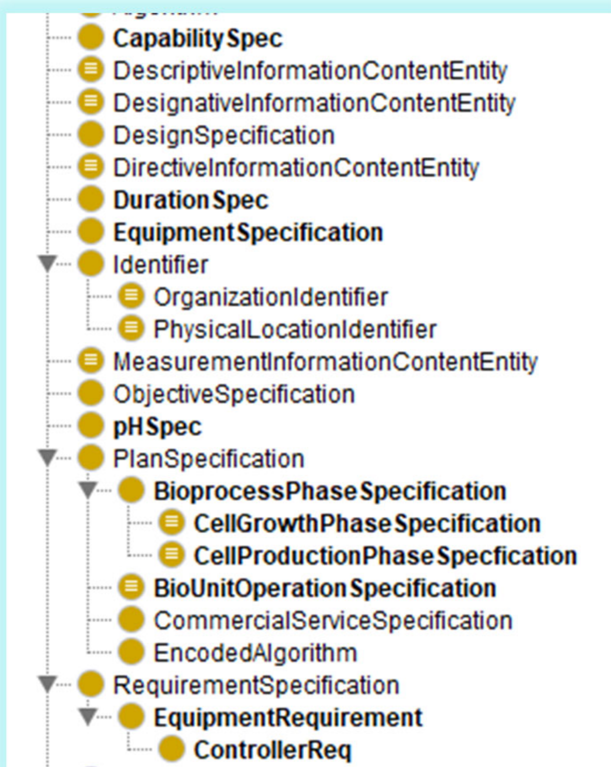
G. Was the pH within limits for the entire process?

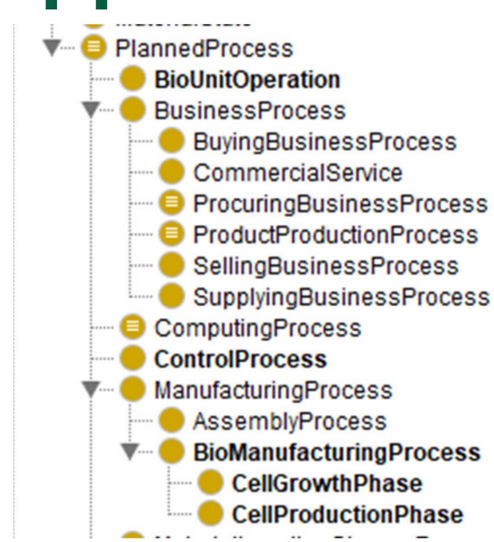
Which equipment are used in a (real) process?

What is the difference of duration between actual and planned process?

Results in Comparing ICE and CR Approaches







Query 3 - ICE

What is the difference of duration between actual and planned process?

```
SELECT DISTINCT ?ProcessPlan ?ActualProcess ?PlannedDurationInDays  ?ActualDurationInDays ?valueDifference
WHERE {
#retrive the needed specifications and the connected processes
      ?ProcessPlan core:prescribes ?ActualProcess.
      {{?ActualProcess rdf:type bpb:BioUnitOperation} UNION
      {?MainProcess bfo:BFO_0000118 ?ActualProcess.                      # has occurrent part
      ?MainProcess rdf:type bpb:BioUnitOperation}}.

#retrieve the planned duration
      ?ProcessPlan bfo:BFO_0000110/bfo:BFO_0000110 ?durationValueExpr.   # has continuant part
      ?durationValueExpr core:hasSimpleExpressionValue ?PlannedDurationInDays.

#retrieve durations of the actual processes
      ?ActualProcess  bfo:BFO_0000199 ?ActualDura.                       # occupies temporal region
      ?ActualDura core:hasValueExpressionAtAllTimes/core:hasSimpleExpressionValue ?ActualDurationInDays.

#retrieve the difference of duration value (- is under, + is over, and 0 same as planned)
      BIND((?ActualDurationInDays - ?PlannedDurationInDays) as ?valueDifference)}
```

Query 3 - CR

What is the difference of duration between actual and planned process?

```
SELECT ?PlannedBUO ?ActualProcess ?PlannedDurationInDays ?ActualDurationInDays ?valueDifference
WHERE {
#retrieve all actual process and their own planned process
    ?PlannedBUO rdf:type biopb:BioUnitOperation.
    ?PlannedBUO cr:hasCounterpart ?ActualProcess.

#retrieve durations the planned processes
    ?PlannedBUO  bfo:BFO_0000199 ?PlannedDura.          # occupies temporal region
    ?PlannedDura core:hasValueExpressionAtAllTimes ?PlannedDurationInstance.
    ?PlannedDurationInstance core:hasSimpleExpressionValue ?PlannedDurationInDays.

    ?ActualProcess  bfo:BFO_0000199 ?ActualDura.        #retrieve durations of the actual processes
    ?ActualDura core:hasValueExpressionAtAllTimes ?ActualDurationInstance.
    ?ActualDurationInstance core:hasSimpleExpressionValue ?ActualDurationInDays.

#retrieve the difference of duration value (- is under, + is over, and 0 same as planned)
    BIND((?ActualDurationInDays - ?PlannedDurationInDays) as ?valueDifference)}
```

Conclusions

- CR approach is superior
- It does not require subclasses of DesignSpecification and PlanSpecification to cover and distinguish various kinds of artifacts and planned processes
- It has uniform reasoning about future and actual artifacts and processes
- It is suitable to represent digital artifacts and digital twins
- It allows easier approach for representing artifact and system performance

- More work
  - Extend the approach to requirements
  - Formalize simulation models of various kinds
  - Formalize relations in FOL
  - Utilize this approach for representation of future and digital artifacts and planned processes

Roadmap

Roadmap for IOF PPS WG Jan-June 2024

Knsnxm%uqfssnsl%yjwrx

X~shmwtsnj%inxhwjyj%gfyhm%
fsi%uwthjxx%rfszkfhyzwnsl%
yjwrx%NXF>:.

\j%sjji%rtwj%ufwynhnufynts%
ns%NTK%UUX%\L

Xhmjizqnsl%yjwrx

Inlnyfq%fwynkfhyx%

Fuuq~%rtwj%zxj%hfxjx

Wjqjfxj%ymj%knwxy%{jwxnts%tk%
uqfssnsl%tsytqtl~%g~%Ozsj%851%7579

Questions

Discussions