SULO – a simplified upper-level ontology

Michel Dumontier, Remzi Çelebi, Komal Gilani, Isabelle de Zegher, Katerina Serafimova, Catalina Martínez Costa, and Stefan Schulz

FOUST

Catania, Italy

September 9, 2025

Maastricht University

Upper Level Ontologies

Ontologies offer a *formalization* of a s*hared conceptualization* of a *domain*. They provide machine interpretable descriptions of entities, their attributes, and their relations.

Upper Level Ontologies (ULOs), aka *foundation ontologies*, offer an overarching axiomatic framework for domain ontologies (DOs) so as to *constrain* the conceptualization and formalisation of a domain.

Many ULOs have been proposed: BFO, GFO, DOLCE, UFO, SUMO, and some such as SIO and BioTop contain and extend ULOs into domain specific classes and relations.

Domain ontologies and schemas

However, many domain ontologies (e.g. SNOMED) and data schemas (e.g. SPHN) are driven by immediate, pragmatic needs and ULOs are not typically used to guide their development.

The lack of adherence to an ULO leads to inefficient representations:

- Inability to extend the domain ontology or schema by reusing domain and/or application-specific relations
- leading to a proliferation of semantically ungrounded relations
- leading to non-interoperable schemas around each target class

ULOs remain difficult to correctly apply

- Most ULOs feature nuanced philosophical considerations and familiarity with logic.
- ULOs adopt unfamiliar or technical labels: continuant, endurant, perdurant, specifically dependent continuant, which avoids meaning overload, but are difficult to grasp and correctly apply by non experts
- ULOs have distinct, missing, underrepresented, or overconstrained areas. BFO and DOLCE focus on particulars, only BFO has time indexed relations (only in their common logic), immaterial bearers are not possible in BFO, BFO/GFO/DOLCE do not offer any data relations.

Even experts in a particular ULO don't consistently use it in the same way

ELSEVIER

Journal of Web Semantics

Volume 57, August 2019, 100469

Measuring expert performance at manually classifying domain entities under upper ontology classes

Robert Stevens [a], Phillip Lord [c], James Malone [b], Nicolas Matentzoglu [a]

8 BFO experts asked to classify 46 commonly known entities from the domain of travel with BFO entities.

- 0.52 inter-rater agreement
- degree of classification consistency is correlated with the frequency the respective BFO classes are used in practice

Projects that inform the effort

AIDAVA - AI-assisted curation of personal health knowledge graphs
-> adopted **SPHN** for RDF-based health data representation

REALM - Regulatory sandbox for medical software devices
-> uses **OMOP** data representations

iCare4CVD - cardiovascular prediction models
-> harmonizes independent cohort data

all use **SNOMED** as a standardized medical domain ontology

SULO

We propose the **Simplified Upper Level Ontology (SULO)** take thats a minimalistic approach to guide the alignment, formalisation, and reusability of upper level and domain ontologies.

SULO attempts to **balance formal rigour with simplicity and practical usability**.

Design Criteria

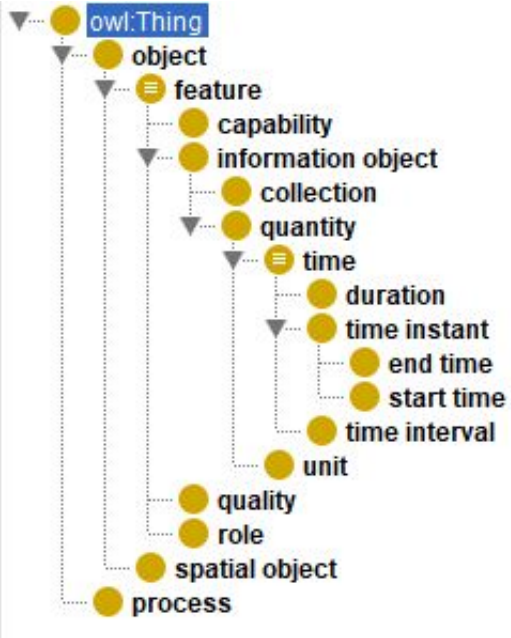
- **Minimalism**: SULO proposes a small taxonomy of disjoint classes and a minimal set of constrained relations to ensure broad applicability across domains.
- **Compatibility**: SULO maintains compatibility with core components of well-known ULOs while remaining accessible to domain experts.
- **Accessibility**: SULO aims to be accessible to users with no or little training in formal ontology through friendly labeling, a simple taxonomy, and fits in a single diagram.
- **Composability**: SULO provides the building blocks to construct complex, machine-readable class expressions.
- **Interoperability**: SULO fosters interoperability by providing a common semantic foundation, including two ontology design patterns, that help domain experts adhere to explicit an implicit semantics.
- **Data validation**: SULO constrains real-world knowledge graphs through automated reasoning and schema validation.

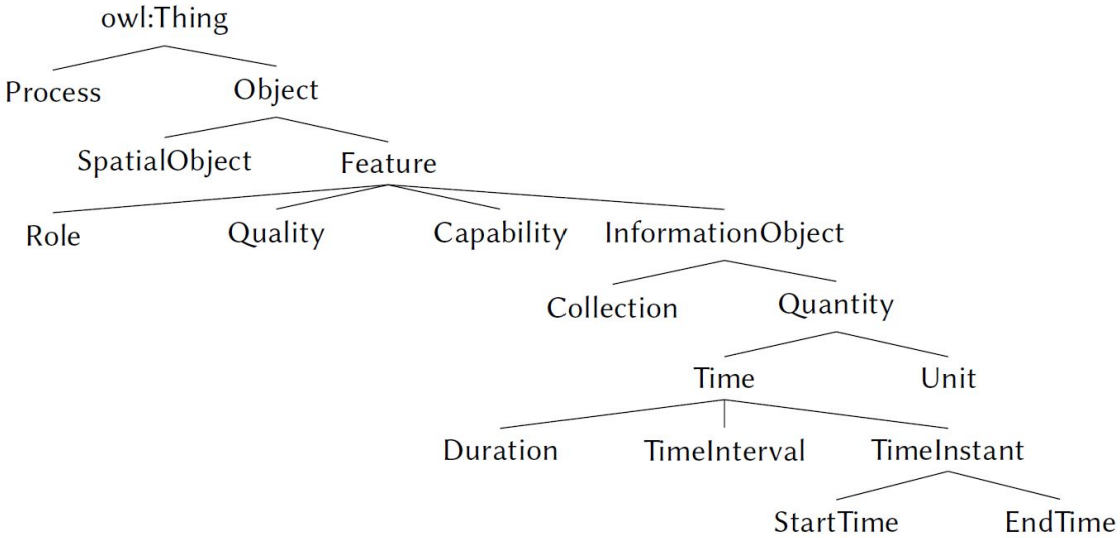
Methods

NEON methodology:

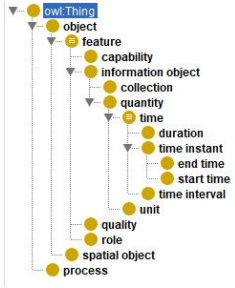
1. domain analysis
2. gathering requirements
3. development of modular and pattern-based designs
4. alignment with standards and existing ontologies
5. iterative development and validation
6. integration and maintenance

17 classes in SULO

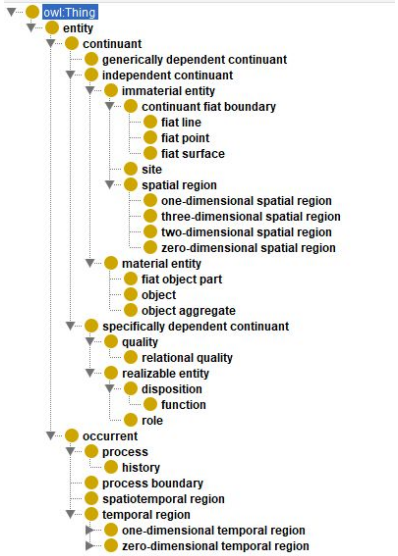




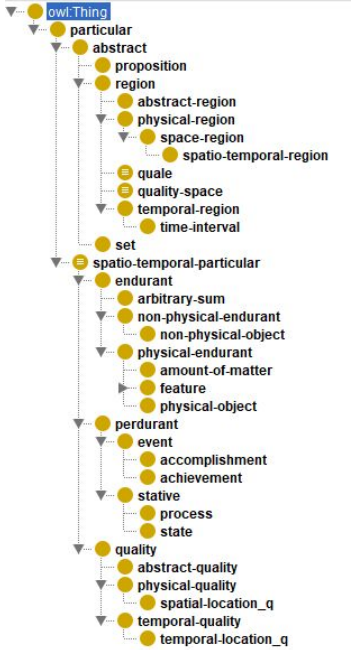
SULO



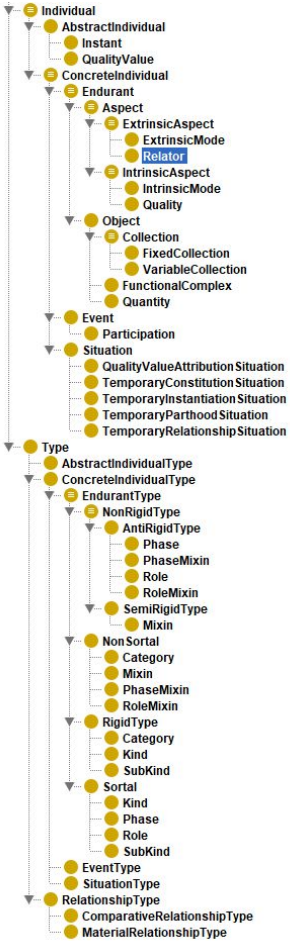
BFO



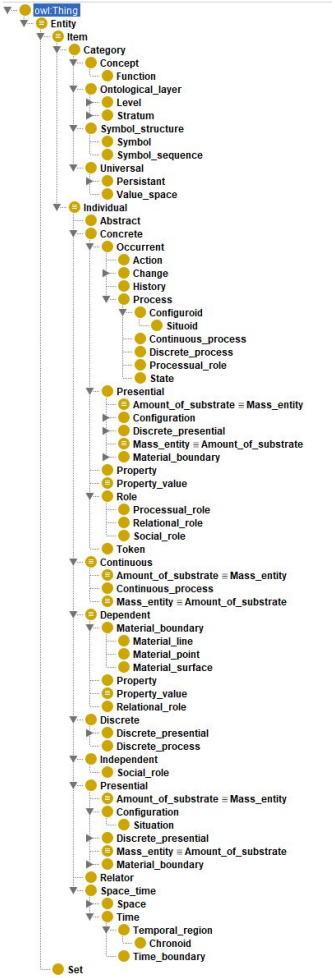
DOLCE



gUFO



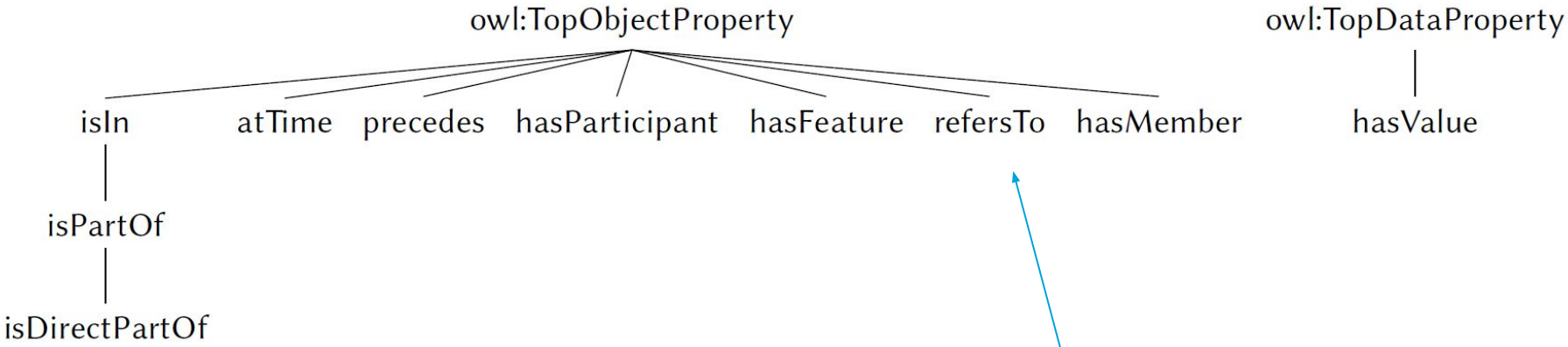
GFO



Maastricht University

SULO Properties

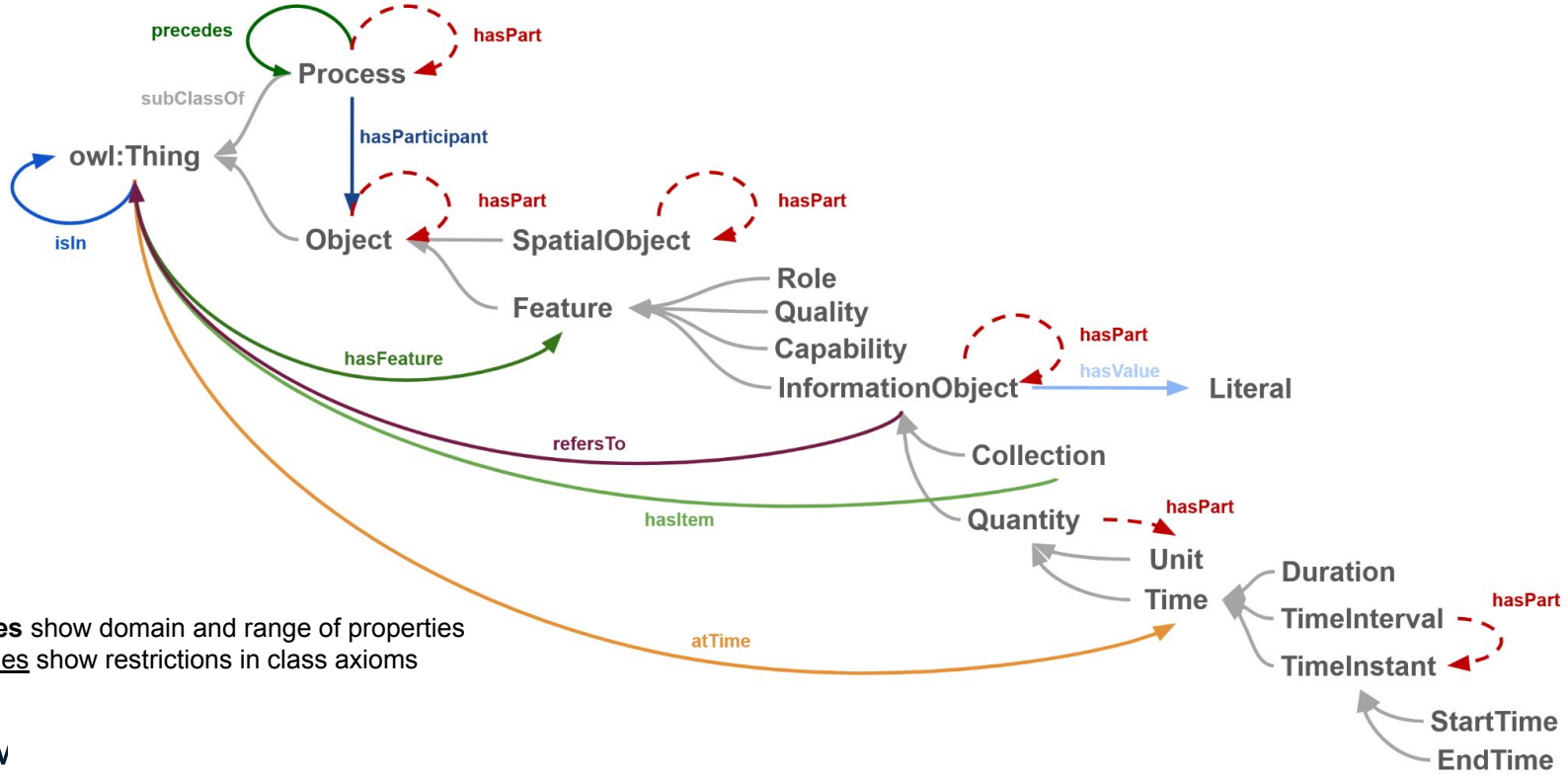
18 object properties (9 direct + 9 inverses) and 1 data property.



Stefan presented in POWERS workshop as it pertains to realizables (and non existing entities)


SULO
owl:topObjectProperty
at time
has feature
has item
has participant
is feature of
is item in
is located in
is part of
is direct part of
is location of
has part
has direct part
is participant in
is preceded by
is referred in
is time of
precedes
refers to
BFO
owl:topObjectProperty
concretizes at some time
continuant part of at some time
environs
exists at
first instant of
generically depends on at some time
generically depends on at all times
has continuant part at some time
has continuant part at all times
has member part at all times
has proper continuant part at all times
has member part at some time
has member part at all times
has proper continuant part at some time
has proper continuant part at all times
has first instant
has history
has last instant
has material basis at some time
has material basis at all times
has occurrent part
has proper occurrent part
has temporal part
has proper temporal part
has participant at some time
has participant at all times
has realization
history of
is carrier of at some time
is carrier of at all times
is concretized by at some time
is concretized by at all times
last instant of
located in at some time
located in at all times
location of at some time
location of at all times
material basis of at some time
material basis of at all times
occupies spatial region at some time
occupies spatial region at all times
occupies spatiotemporal region
occupies temporal region
occurrent part of
proper occurrent part of
temporal part of
proper temporal part of
occurs in
participates in at some time
participates in at all times
preceded by
precedes
realizes
spatially projects onto at some time
spatially projects onto at all times
specifically depended on by
bearer of
specifically depends on
inheres in
temporally projects onto
GFO
owl:topObjectProperty
abstract_has_part
has_categorial_part
has_part
has_proper_part
has_constituent_part
abstract_part_of
categorial_part_of
category_in_layer
sequence_constituent_of
part_of
proper_part_of
constituent_part_of
caused_by
causes
agent_in
depends_on
boundary_of
spatial_boundary_of
time_boundary_of
exists_at
frames
function_determinant_of
functional_item_of
goal_of
requirement_of
function_of
has_agent
has_boundary
has_spatial_boundary
has_time_boundary
has_left_time_boundary
has_right_time_boundary
has_category
has_function
has_function_determinant
has_functional_item
has_goal
has_requirement
has_member
has_participant
has_sequence_constituent
has_value
instance_of
instantiated_by
has_token
layer_of
level_of
stratum_of
left_boundary_of
member_of
necessary_for
occupied_by
occupies
framed_by
on_layer
on_level
on_stratum
participates_in
agent_in
plays_role
projection_of
projects_to
realized_by
realizes
right_boundary_of
role_of
token_of
value_of
gUFO
owl:topObjectProperty
broughtAbout
categorizes
partitions
concernsConstitutedEndurant
concernsNonRigidType
concernsQualityType
concernsReifiedQualityValue
concernsRelatedEndurant
concernsRelationshipType
concernsTemporaryWhole
constitutes
contributedToTrigger
externallyDependsOn
hasAssociatedQualityValueType
hasBeginPoint
hasEndPoint
hasReifiedQualityValue
historicallyDependsOn
inheresIn
isDerivedFrom
isProperPartOf
isAspectProperPartOf
isEventProperPartOf
isObjectProperPartOf
isCollectionMemberOf
isComponentOf
isSubCollectionOf
isSubQuantityOf
isSituationProperPartOf
manifestedIn
mediates
participatedIn
standsIn
standsInQualifiedAttribution
standsInQualifiedConstitution
standsInQualifiedInstantiation
standsInQualifiedParthood
standsInQualifiedRelationship
wasCreatedIn
wasTerminatedIn
DOLCE
owl:topObjectProperty
immediate-relation
generic-constituent
generic-dependent
identity-c
identity-n
inherent-in
t-inherent-in
part
atomic-part
proper-part
boundary
temporary-proper-part
temporary-atomic-part
temporary-part
mereologically-coincides
temporary-proper-part
temporary-atomic-part
participant
constant-participant
life-of
total-constant-participant
temporary-participant
total-temporary-participant
q-location
has-quale
r-location
specific-constant-constituent
specific-constant-dependent
host-of
weak-connection
immediate-relation-i
generic-constituent-of
generically-dependent-on
has-quality
has-t-quality
part-of
atomic-part-of
proper-part-of
boundary-of
temporary-proper-part-of
temporary-atomic-part-of
temporary-part-of
temporary-proper-part-of
temporary-atomic-part-of
participant-in
constant-participant-in
life
total-constant-participant-in
temporary-participant-in
total-temporary-participant-in
q-location-of
quale-of
r-location-of
specific-constant-constituent-of
specifically-constantly-dependent-on
host
mediated-relation
generic-location
exact-location
abstract-location
physical-location
spatio-temporally-present-at
overlaps
partly-compresent
temporary-part
mereologically-coincides
temporary-proper-part
temporary-atomic-part
temporary-part-of
temporary-proper-part-of
temporary-atomic-part-of
q-present-at
sibling-part
strong-connection
mediated-relation-i
generic-location-of
exact-location-of
abstract-location-of
physical-location-of

SULO in a postcard



solid lines show domain and range of properties
dotted lines show restrictions in class axioms

Ontology Design Patterns

Ontology Design Patterns (ODPs) can help to structure data graphs that foster semantic interoperability across domain-specific knowledge representations

We propose to key ODPs: **SOLID** and **PRO**

SOLID focuses describing *literal* containing relations
PRO focuses on describing *role-based* relations

SOLID ODP

The SOLID pattern uses SULO's single *functional* data property, **hasValue**, to assign a literal value to an InformationObject.

Example: Instead of using arbitrary relations such as **hasTemperature** or **hasTemperatureInCelcius**, the design pattern reuses SULO's **hasValue**, **hasFeature**, **refersTo,** and **hasPart** properties in conjunction with two externally defined classes, namely **Temperature** from PATO and **Celcius** from the Unit Ontology

Hence, developing an ontology of InformationObjects is encouraged, rather than a proliferation of data properties.

```
@prefix sulo:   <http://w3id.org/sulo/> .
@prefix uo: <http://purl.obolibrary.org/obo/> .
@prefix pato: <http://purl.obolibrary.org/obo/>
@prefix : <http://example.org/> .

:alice a sulo:SpatialObject, :Person;
    sulo:hasFeature :alice_temperature_measurement_1 .

:alice_temperature_measurement_1 a sulo:Quantity;
    sulo:hasValue "37.8"^^xsd:double ;
    sulo:refersTo [ a pato:PATO_0000146] ; # the quality of temperature
    sulo:hasPart [ a uo:UO_0000027 ] . # the celcius unit
```

PRO ODP

The Process-Role-Object (PRO) ODP provides a way to represent *the manner in which* **objects** participate in **processes** through specified **roles**.

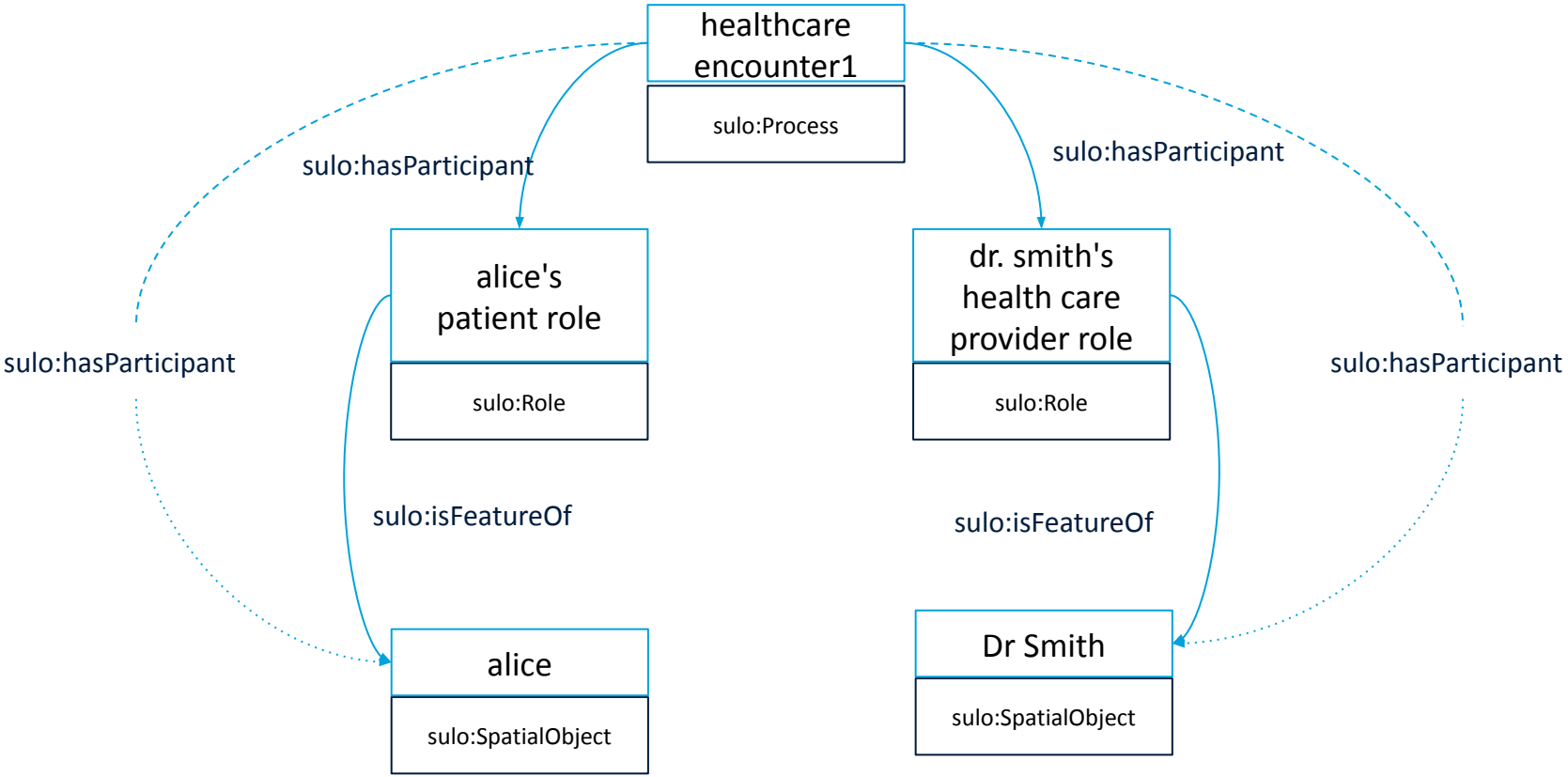
```
EquivalentTo:
  Process and
      hasParticipant some (
          Role and isFeatureOF some Object
      )
```

Applies to spatial objects and information objects!

```
hasParticipant o isFeatureOf -> hasParticipant
```

Role chain to infer participation of role holding object

The pattern makes explicit the semantics for role-aligned object relations such as **hasPatient**, **hasCareProvider**. *Hence, this pattern reduces the impulse to create and proliferate such relations.*



hasParticipant o isFeatureOf -> hasParticipant

SNOMED CT is a formalised domain ontology

SNOMED CT is a formalised international clinical terminology used in coding health records.

SNOMED CT comprises 27 top level categories, 120+ relations.

Some classes in SNOMED have ambiguous meanings, and there is interest in refactoring.

Most SNOMED classes and relations can be mapped to SULO classes and relations, but some SNOMED relations are compound and map to a subgraph

Table 1
Mappings between SNOMED CT Top Concepts and Relations and corresponding SULO Classes and Properties

SNOMED CT Top Concept	SULO Class	SNOMED CT Relation	SULO Object Property
Clinical finding (finding)	sulo:Process	Occurrence (attribute)	sulo:atTime
Procedure (procedure)		Finding site (attribute)	sulo:isIn
Action (qualifier value)		Procedure site - Direct (attribute)	
Organism (organism)	sulo:SpatialObject	Using device (attribute)	sulo:hasParticipant
Specimen (specimen)		Causative agent (attribute)	
Substance (substance)		Due to (attribute)	
Pharmaceutical / biologic product (product)		Procedure device (attribute)	
Environment or geographical location (environment/location)		Laterality (attribute)	sulo:hasFeature
Physical object (physical object)		Associated morphology (attribute)	
Body structure (body structure)		Has basic dose form (attribute)	
Observable entity (observable entity)	sulo:InformationObject	Access (attribute)	
Record artifact (record artifact)		Component (attribute)	
Situation with explicit context (situation)		Has active ingredient (attribute)	sulo:hasPart
Staging and scales (staging scale)		Specimen substance (attribute)	
Physical force (physical force)	sulo:Feature	Method (attribute)	
Time (property) (qualifier value)	sulo:Time	Role Group (attribute)	
Unit of measure (qualifier value)	sulo:Unit		

SNOMED axioms can be expressed with domain-independent SULO relations

```
Class: 'Fracture of femur (disorder)'

  EquivalentTo:
    'Disease (disorder)'
    and 'Role group (attribute)' some (
        'Associated morphology (attribute)' some 'Fracture (morphologic abnormality)'
        and
        'Finding site (attribute)' some 'Bone structure of femur (body structure)'
    )
```

Figure 5: SNOMED class definition for Fracture of Femur (disorder)

```
Class: 'Fracture of femur (disorder)'

    EquivalentTo:
    'Disease (disorder)'
        and sulo:hasPart some (
            (sulo:Process and sulo:isIn some
            'Bone structure of femur (body structure)'
                and sulo:hasFeature some 'Fracture (morphologic abnormality)')
        )
```

Figure 6: SULO representation of SNOMED Fracture of femur

SULO-based SPHN schema contains a more robust semantic

```
Class: 'Administrative Case'

    subClassOf:
    sulo:Process
        and sulo:hasParticipant some (
            'Subject of Care Role'
                and sulo:isFeatureOf some (
                    Person
                    and sulo:hasFeature some (
                        SubjectPseudoIdentifier
                        and sulo:hasValue value xsd:string # subjectpseudoidentifier
                    )
                )
            )
        and sulo:hasPart some (
            PreAdmission
            and sulo:isIn some SpatialObject # sourceLocation
        )
        and sulo:hasPart some (
            Admission
            and sulo:atTime some (
                TimeInstant
                and hasValue value xsd:dateTime # admission datetime
        )
        and sulo:hasPart some CareHandling
        and sulo:hasPart some (
            Discharge
            and sulo:atTime some (
                TimeInstant
                and hasValue value xsd:dateTime # discharge datetime
        )
        and sulo:hasPart some (
            PostDischarge
            and sulo:isIn some SpatialObject # targetLocation
        )
        and sulo:isReferredToIn some InformationObject # sourceSystem
```

SPHN uses *hasX (*object and data) properties, which are brittle when the schema is refactored. A SULO-based formalisation ensures predicate semantics are put into classes, whose instances are connected with domain-independent relations.

Table 2
Comparison of SPHN Administrative Case Schema (v.2023.2 vs v.2025.1)

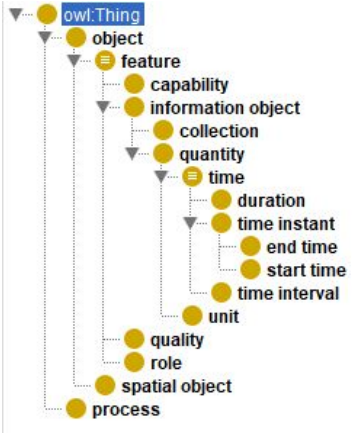
2023.2	2025.1	Property	Cardinality	Range
☒	☒	hasIdentifier	1..1	xsd:string
☒	☒	hasSubjectPseudoIdentifier	1..1	SubjectPseudoIdentifier
☒	☒	hasCareHandling	0..1	CareHandling
-	☒	hasAdmission	1..1	Admission
☒	-	hasAdmissionDatetime	1..1	xsd:dateTime
☒	-	hasAdmissionLocation	0..1	Location
-	☒	hasDischarge	0..1	Discharge
☒	-	hasDischargeDatetime	0..1	xsd:dateTime
☒	-	hasDischargeLocation	0..1	Location
☒	-	hasDataProviderInstitute	1..1	DataProviderInstitute
-	☒	hasSourceSystem	1..*	SourceSystem

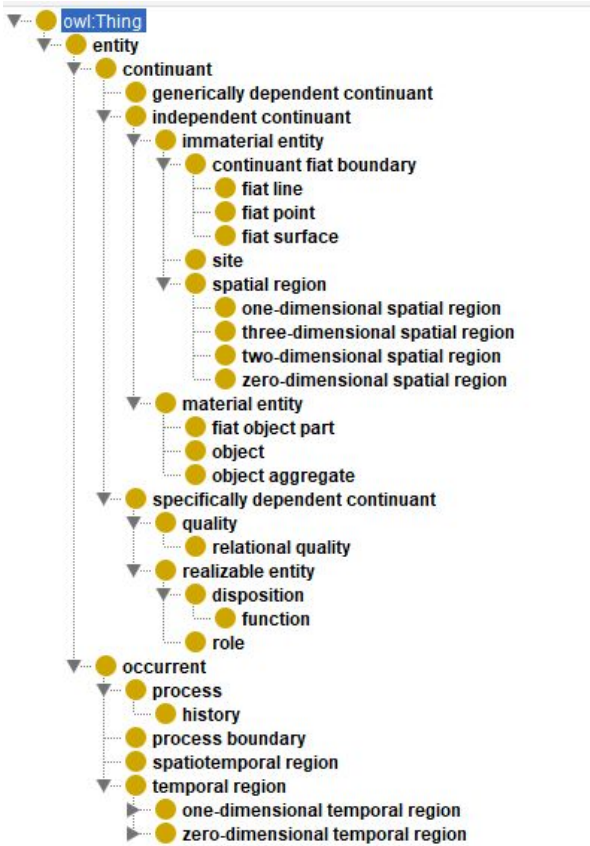
Table 3
Schema definition for SPHN (v.2025.1) Admission and Discharge

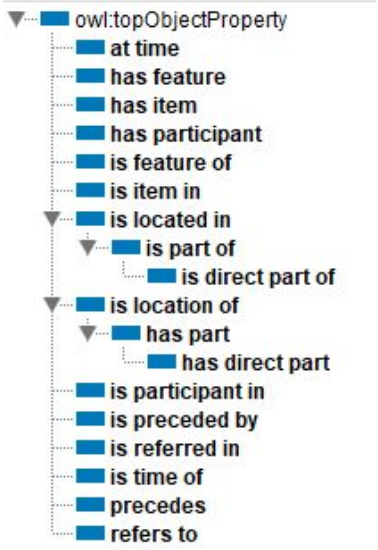
2023.2	2025.1	Class	Property	Cardinality	Range
–	☒	Admission	hasOriginLocation	0 .. 1	Location
–	☒		hasDatetime	1 .. 1	xsd:dateTime
–	☒	Discharge	hasTargetLocation	0 .. 1	Location
–	☒		hasDatetime	1 .. 1	xsd:dateTime

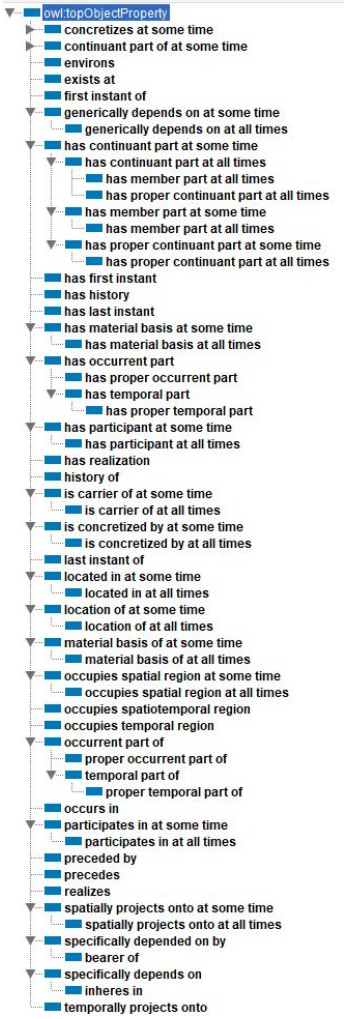
SULO to BFO

- SULO more compact than BFO in both classes and properties
- They share a top level binary division of Process (Occurrent) and Object (Continuant)
- SULO's Object corresponds to GDC + SDC + IC
- SULO includes InformationEntities including Quantity, Collection and Time, while BFO doesn't
- SULO has data property, BFO doesn't.
- BFO has spatial / temporal regions and boundaries, SULO doesn't.






owl:topObjectProperty
at time
has feature
has item
has participant
is feature of
is item in
is located in
is part of
is direct part of
is location of
has part
has direct part
is participant in
is preceded by
is referred in
is time of
precedes
refers to


owl:topObjectProperty
concretizes at some time
continuant part of at some time
environs
exists at
first instant of
generically depends on at some time
generically depends on at all times
has continuant part at some time
has continuant part at all times
has member part at all times
has proper continuant part at all times
has member part at some time
has member part at all times
has proper continuant part at some time
has proper continuant part at all times
has first instant
has history
has last instant
has material basis at some time
has material basis at all times
has occurrent part
has proper occurrent part
has temporal part
has proper temporal part
has participant at some time
has participant at all times
has realization
history of
is carrier of at some time
is carrier of at all times
is concretized by at some time
is concretized by at all times
last instant of
located in at some time
located in at all times
location of at some time
location of at all times
material basis of at some time
material basis of at all times
occupies spatial region at some time
occupies spatial region at all times
occupies spatiotemporal region
occupies temporal region
occurrent part of
proper occurrent part of
temporal part of
proper temporal part of
occurs in
participates in at some time
participates in at all times
preceded by
precedes
realizes
spatially projects onto at some time
spatially projects onto at all times
specifically depended on by
bearer of
specifically depends on
inheres in
temporally projects onto

have the design goals been met?

- **Minimalism**: A small taxonomy of 17 classes and 19 relations: smallest of ULOs, and compatibility with 2 biomedical ontology/schema.
- **Compatibility**: Conceptual and logical compatibility with core components of BFO with key simplifications (no regions, consolidation of SDC & GDCs into 1 class)
- **Accessibility**: Simpler labeling, simpler taxonomy, and fits in a single diagram - user study being planned.
- **Composability**: Building blocks to construct complex, machine-readable class expressions.
- **Interoperability**: Interoperability through proposed to SULO axioms + SOLID and PRO design patterns to reduce arbitrary relations and constrain the data graph.
- **Data validation**: automated reasoning and (planned) schema validation.

limitations

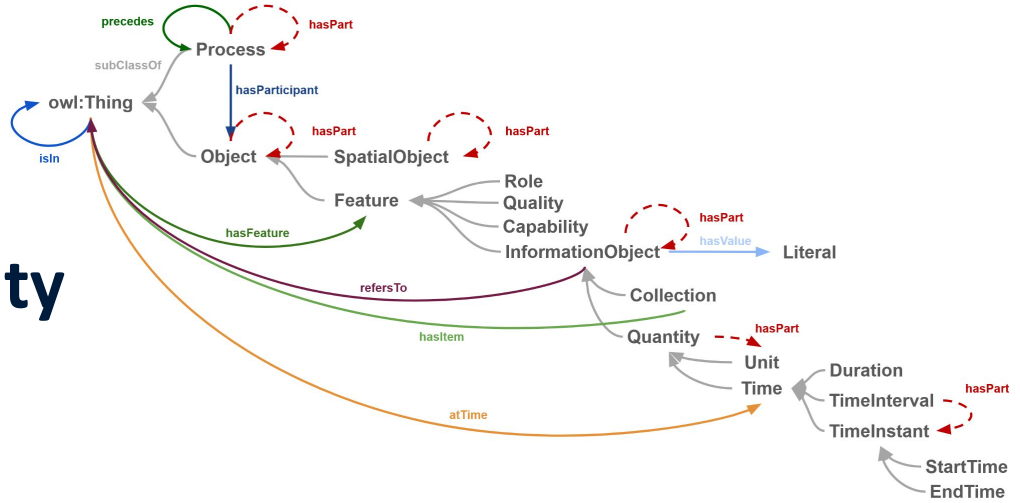
- focused on classes and relations relevant to biomedicine
  - we have not made a full inventory of ULO classes, relations, and their application.
- does not put forward a theory of change, space, and time, and other theories of ontology

Future Work

- Revision/extension/refinement of SULO towards a robust theory across application spaces.
- Tooling to enable users to start an ontology project using SULO.
- Tooling to facilitate SULO as a hub for interoperoperability of different ontologies and schemas.
- Studies on the useability of SULO and other ULOs.

Summary

SULO takes a minimalistic approach to an ULO formulation that balances formal rigour with simplicity and practical usability.



This work supported by Horizon Europe Framework Program under Grant Agreement Nos. 101057062 (AIDAVA), 101112022 (iCare4CVD), 101095435 (REALM), 101057603 (RES-Q+), 101080875 (CMC), and the Spanish Research Agency Grant RYC2020-030190-I.