Toward a Modular Ontology for Robotic Orchestration

By: Michael McCain, Chris Davis Jaldi, Shreyas Casturi, and Susan Shrestha

Overview

- Use Case
- CQ’s
- Datasets
- Key Notions
- Modules and Axioms
- Schema
- Future work

Use Case

- We want to orchestrate robotic agents for task planning and execution
- Many task-planning ontologies, but very few for robotic orchestration
- Issue: Many of these task-planning ontologies are difficult to integrate
- Goal: Engineer a modular, re-usable robotic orchestration ontology

Competency Questions

- What set of agents are required to complete primary goal y?
- What are the classifications/categories that of a set of objects?
- What objects are required for goal g?
- What objects are required to complete task x?
- What are the dependent tasks for task z?
- Which tasks share common dependent tasks across different goals?

Integrated Datasets

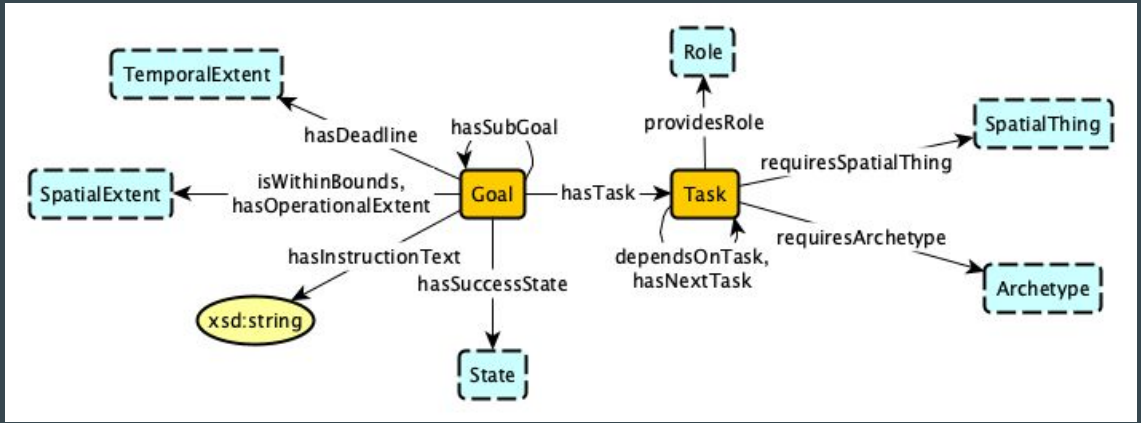
- RH20T-P: Primitive-Level Robotic Dataset Towards Composable Generalization Agents
- Droid Robot Manipulation Dataset
- BridgeData V2 Robot Learning at Scale
- NIST Manufacturing Objects and Assemblies Dataset
- Action Learning From Realistic Environments and Directives
- Princeton Model Net

Key Notions

- Archetype
- Capability
- Category
- Goal-Task
- Role
- Spatial-Thing

Goal-Task Pattern

- **Source Pattern** : Trajectory, SpatiotemporalExtent, Description-Situation
- **Source Data** : DROID, RH20T-P
- **Description:** Specifications to represent desired outcomes and anchors to atomic unit of work with hierarchies, decomposition and sequence semantics



Goal-Task Axioms

- Axiom 1
  - Goal SubClassOf hasDeadline some TemporalExtent
  - If a Goal exists, it must have at least one hasDeadline, and is of type TemporalExtent
- Axiom 2
  - Goal SubClassOf hasDeadline max 1 TemporalExtent
  - For every Goal, there exists no or exactly 1 hasDeadline, and is of type TemporalExtent
- Axiom 3
  - Thing SubClassOf isWithinBounds only SpatialExtent
  - For any thing, if there exists isWithinBounds, the global range is SpatialExtent.

Goal-Task Axioms

- Axiom 4
  - Goal SubClassOf isWithinBounds only SpatialExtent
  - For every Goal there may exist isWithinBounds, and is of type SpatialExtent.
- Axiom 5
  - Thing SubClassOf hasOperationalExtent only SpatialExtent
  - For any thing, if there exists hasOperationalExtent, the global range is SpatialExtent.
- Axiom 6
  - Goal SubClassOf hasInstructionalText max 1 xsd:String
  - For every Goal, there exists no or exactly 1 hasInstructionalText, and is of type xsd:String

Goal-Task Axioms

- Axiom 7
  - hasSubGoal some Goal SubClassOf Goal
  - The scoped domain of hasSubGoal, scoped by Goal, is of Goal.
- Axiom 8
  - Goal SubClassOf hasSubGoal only Goal
  - The scoped range of hasSubGoal, scoped by Goal, is of Goal
- Axiom 9
  - Thing SubClassOf hasSuccessState only State
  - For any thing, if there exists hasSuccessState, the global range is State.

Goal-Task Axioms

- Axiom 10
  - hasTask some Task SubClassOf Goal
  - The scoped domain of hasSubGoal, scoped by Task, is of Goal.
- Axiom 11
  - Goal SubClassOf hasTask only Task
  - The scoped range of hasSubGoal, scoped by Goal, is of Task
- Axiom 12
  - Goal SubClassOf hasTask max 1 Task
  - For every Goal, there exists no or exactly 1 hasTask, and is of type Task

Goal-Task Axioms

- Axiom 13
  - dependsOnTask some Task SubClassOf Task
  - The scoped domain of dependsOnTask, scoped by Task, is of Task.
- Axiom 14
  - Task SubClassOf dependsOnTask only Task
  - The scoped range of dependsOnTask, scoped by Task, is of Task
- Axiom 15
  - Task SubClassOf dependsOnTask only Task
  - For every Task there may exist dependsOnTask, and is of type Task.

Goal-Task Axioms

- Axiom 16
  - hasNextTask some Task SubClassOf Task
  - The scoped domain of hasNextTask, scoped by Task, is of Task.
- Axiom 17
  - Task SubClassOf hasNextTask only Task
  - The scoped range of hasNextTask, scoped by Task, is of Task
- Axiom 18
  - Task SubClassOf hasNextTask only Task
  - For every Task there may exist hasNextTask, and is of type Task.

Goal-Task Axioms

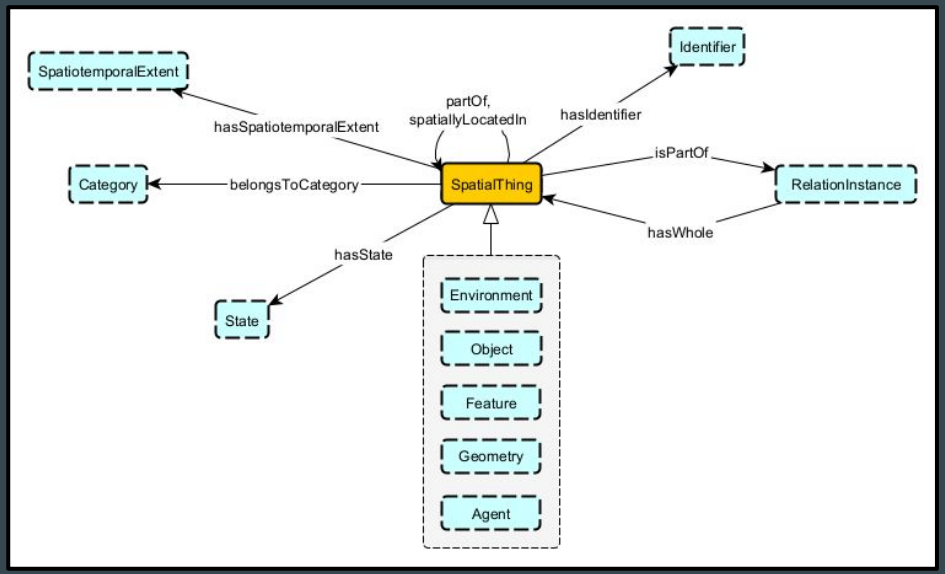
- Axiom 19
  - Task SubClassOf providesRole only Role
  - The scoped range of providesRole, scoped by Task, is of Role
- Axiom 20
  - Role SubClassOf inverse providesRole some Task
  - For every Task, there has to be an inverse providesRole-filler and is of type Role
- Axiom 21
  - Task SubClassOf requiresSpatialThing some SpatialThing
  - If a Task exists, it must have at least one requiresSpatialThing, and is of type SpatialThing

Goal-Task Axioms

- Axiom 22
  - Task SubClassOf requiresArchetype only Archetype
  - The scoped range of requiresArchetype, scoped by Task, is of Archetype
- Axiom 23
  - Task SubClassOf requiresArchetype some Archetype
  - If a Task exists, it must have at least one requiresArchetype, and is of type Archetype
- Axiom 24
  - Task SubClassOf requiresArchetype max 1 Archetype
  - For every Task, there exists no or exactly 1 requiresArchetype, and is of type Archetype

Spatial Thing Pattern

- **Source Pattern** : Winston's Part Whole, Spatial-Object, Identifier
- **Source Data** : All
- **Description:** Specification to represent physical things that exists in time and space.



Spatial Thing Axioms

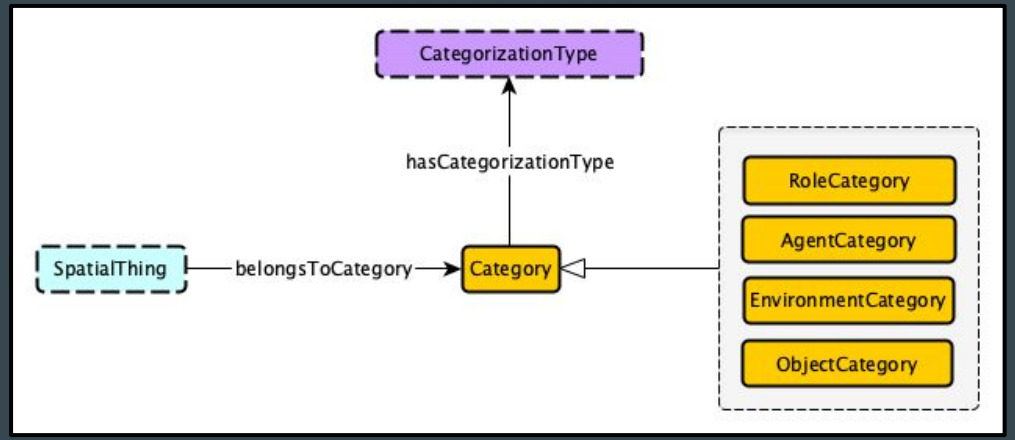
- Axiom 1
  - Thing SubClassOf hasIdentifier only Identifier
  - For any thing, if there exists hasIdentifier, the global range is Identifier.
- Axiom 2
  - Thing SubClassOf hasIdentifier max 1 Identifier
  - For any thing, there exists no or exactly 1, and is of type Identifier
- Axiom 3
  - Thing SubClassOf hasSpatioTemporalExtent only SpatioTemporalExtent
  - For any thing, if there exists hasSpatioTemporalExtent, the global range is SpatioTemporalExtent.

Spatial Thing Axioms

- Axiom 4
  - SpatialThing SubClassOf hasSpatioTemporalExtent some SpatioTemporalExtent
  - If a SpatialThing exists, it must have at least one hasSpatioTemporalExtent, and is of type SpatioTemporalExtent
- Axiom 5
  - Thing SubClassOf hasState only State
  - For any thing, if there exists hasState, the global range is State.
- Axiom 6
  - Thing SubClassOf belongsToCategory only Category
  - For any thing, if there exists belongsToCategory, the global range is Category.

Category Pattern

- **Source Pattern** : Explicit Typing
- **Source Data** : BridgeData V2, Princeton ModelNet
- **Description:** A specification to differentiate SpatialThing. Each Category subclass has an explicit CategorizationType. This type is provided by a controlled vocabulary.



Category Axioms

- **Triple: RoleCategory SubClassOf Category**
- Axiom 1
  - RoleCategory SubClassOf Category (**SubClassOf**)
  - Every RoleCategory is a Category.
- Axiom 2
  - RoleCategory DisjointWith AgentCategory (**DisjointWith**)
  - No RoleCategory can belong to the AgentCategory class
- Axiom 3
  - RoleCategory DisjointWith EnvironmentCategory (**DisjointWith**)
  - No RoleCategory can belong to the EnvironmentCategory class
- Axiom 4
  - RoleCategory DisjointWith ObjectCategory (**DisjointWith)**
  - No RoleCategory can belong to the ObjectCategory class

Category Axioms

- **Triple: AgentCategory SubClassOf Category**
- Axiom 5
  - AgentCategory SubClassOf Category (**SubClassOf**)
  - Every AgentCategory is a Category.
- Axiom 6
  - AgentCategory DisjointWith EnvironmentCategory (**DisjointWith)**
  - No AgentCategory can belong to the EnvironmentCategory class.
- Axiom 7
  - AgentCategory DisjointWith ObjectCategory (**DisjointWith)**
  - No AgentCategory can belong to the ObjectCategory class.
- Axiom 8
  - AgentCategory DisjointWith RoleCategory (**DisjointWith)**
  - No AgentCategory can belong to the RoleCategory class.

Category Axioms

- **Triple: EnvironmentCategory SubClassOf Category**
- Axiom 9
  - EnvironmentCategory SubClassOf Category **(SubClassOf)**
  - Every EnvironmentCategory is a Category.
- Axiom 10
  - EnvironmentCategory DisjointWith AgentCategory **(DisjointWith)**
  - No EnvironmentCategory can belong to the AgentCategory class.
- Axiom 11
  - EnvironmentCategory DisjointWith ObjectCategory **(DisjointWith)**
  - No EnvironmentCategory can belong to the ObjectCategory class.
- Axiom 12
  - EnvironmentCategory DisjointWith RoleCategory **(DisjointWith)**
  - No EnvironmentCategory can belong to the RoleCategory class.

Category Axioms

- **Triple: ObjectCategory SubClassOf Category**
- Axiom 13
  - ObjectCategory SubClassOf Category (**SubClassOf)**
  - Every ObjectCategory is a Category.
- Axiom 14
  - ObjectCategory DisjointWith AgentCategory **(DisjointWith)**
  - No ObjectCategory can belong to the AgentCategory class.
- Axiom 15
  - ObjectCategory DisjointWith EnvironmentCategory **(DisjointWith)**
  - No ObjectCategory can belong to the EnvironentCategory class.
- Axiom 16
  - ObjectCategory DisjointWith RoleCategory **(DisjointWith)**
  - No ObjectCategory can belong to the RoleCategory class.

Category Axioms

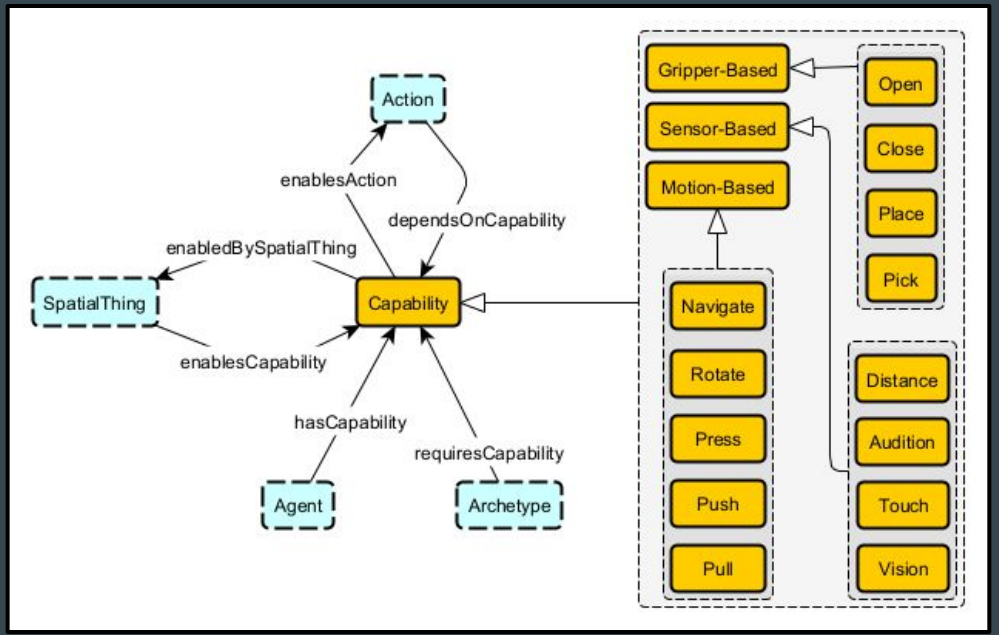
- **Triple: SpatialThing belongsToCategory Category**
- Axiom 17
  - SpatialThing SubClassOf belongsToCategory some Category (**Existential**)
  - Every SpatialThing belongs to at least one Category.

Category Axioms

- **Triple: Category hasCategorizationType CategorizationType**
- Axiom 18
  - hasCategorizationType CategorizationType SubClassOf Category (**Scoped Domain**)
  - Because the tail entity of a “hasCategorizationType” triple is CategorizationType, the head entity must be a Category.
- Axiom 19
  - Category SubClassOf hasCategorizationType only CategorizationType (**Scoped Range**)
  - Because the head entity of a “hasCategorizationType” triple is Category, the tail entity must be a CategorizationType.
- Axiom 20
  - Category SubClassOf hasCategorizationType only CategorizationType (**Existential**)
  - For each Category instance, there exists a CategorizationType instance that is related to the Category instance by the “hasCategorizationType” relationship.

Capability Pattern

- **Source Pattern** : None
- **Source Data** : RH20T-P, DROID
- **Description** : Specifications to differentiate and define what actions an agent can perform.



Note: Additional subclasses to be added in the future.

Capability Axioms

- Axiom 1
  - enablesAction some Action SubClassOf Capability
  - The scoped domain of enablesAction, scoped by Action, is of Capability.
- Axiom 2
  - Action SubClassOf inverse enablesAction some Capability
  - Every action belongs to some (min1) Capability.
- Axiom 3
  - enabledBySpatialThing some SpatialThing SubClassOf Capability
  - The scoped domain of enabledBySpatialThing, scoped by SpatialThing, is of Capability.

Capability Axioms

- Axiom 4
  - Capability SubClassOf enabledBySpatialThing SpatialThing
  - Every Capability belongs to some (min1) SpatialThing.
- Axiom 5
  - Capability SubClassOf enabledBySpatialThing max 1 SpatialThing
  - Every Capability is enabled by at most 1 SpatialThing.
- Axiom 6
  - Gripper-Based SubClassOf Capability
  - Every Gripper-Based Capability is an Capability.

Capability Axioms

- Axiom 7
  - Open/Close SubClassOf Gripper-Based
  - Every Open/Close Gripper-Based Capability is a Gripper-Based Capability.
- Axiom 8
  - Place/Pick SubClassOf Gripper-Based
  - Every Place/Pick Gripper-Based Capability is a Gripper-Based Capability.
- Axiom 9
  - Place/Pick DisjointWith Open/Close
  - Place and Pick Gripper-Based Capabilities are exclusive from Open and Close GBC.

Capability Axioms

- Axiom 10
  - Sensor-Based SubClassOf Capability
  - Every Sensor-Based Capability is an Capability.
- Axiom 11
  - Distance SubClassOf Sensor-Based
  - Every Distance Sensor-Based Capability is a Sensor-Based Capability.
- Axiom 12
  - Audition SubClassOf Sensor-Based
  - Every Audition Sensor-Based Capability is a Sensor-Based Capability.

Capability Axioms

- Axiom 13
  - Touch SubClassOf Sensor-Based
  - Every Touch Sensor-Based Capability is a Sensor-Based Capability.
- Axiom 14
  - Vision SubClassOf Sensor-Based
  - Every Vision Sensor-Based Capability is a Sensor-Based Capability.
- Axiom 15 - 18
  - “Sensor-Based Subclass” DisjointWith “Sensor-Based Subclass”
  - All Sensor-Based Subclasses are exclusive from each other.

Capability Axioms

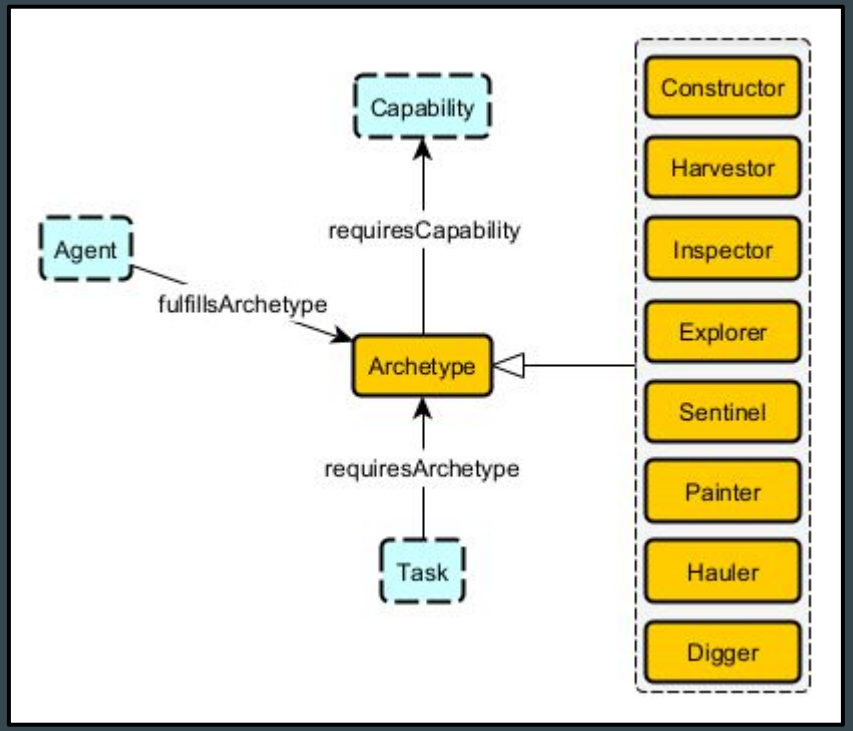
- Axiom 19
  - Motion-Based SubClassOf Capability
  - Every Motion-Based Capability is an Capability.
- Axiom 20
  - Move SubClassOf Motion-Based
  - Every Move Motion-Based Capability is a Motion-Based Capability.
- Axiom 21
  - Rotate SubClassOf Motion-Based
  - Every Rotate Motion-Based Capability is a Motion-Based Capability.

Capability Axioms

- Axiom 22
  - Press SubClassOf Motion-Based
  - Every Press Motion-Based Capability is a Motion-Based Capability.
- Axiom 23
  - Push/Pull SubClassOf Motion-Based
  - Every Push/Pull Motion-Based Capability is a Motion-Based Capability.
- Axiom 24 - 27
  - “Motion-Based Subclass” DisjointWith “Motion-Based Subclass”
  - All Motion-Based Subclasses are exclusive from each other.

Archetype Pattern

- **Source Pattern** : None
- **Source Data** : RH20T-P
- **Description:** Representing the thematic capabilities of a robotic agent (E.g., Explorer, Hauler, Designer, Painter, etc.).



Note: Additional subclasses to be added in the future.

Archetype Axioms

- Axiom 1
  - Constructor SubClassOf Archetype
  - Every Constructor is an Archetype.
- Axiom 2
  - Harvester SubClassOf Archetype
  - Every Harvester is an Archetype.
- Axiom 3
  - Inspector SubClassOf Archetype
  - Every Inspector is an Archetype.

Archetype Axioms

- Axiom 4
  - Explorer SubClassOf Archetype
  - Every Explorer is an Archetype.
- Axiom 5
  - Sentinel SubClassOf Archetype
  - Every Sentinel is an Archetype.
- Axiom 6
  - Painter SubClassOf Archetype
  - Every Painter is an Archetype.

Archetype Axioms

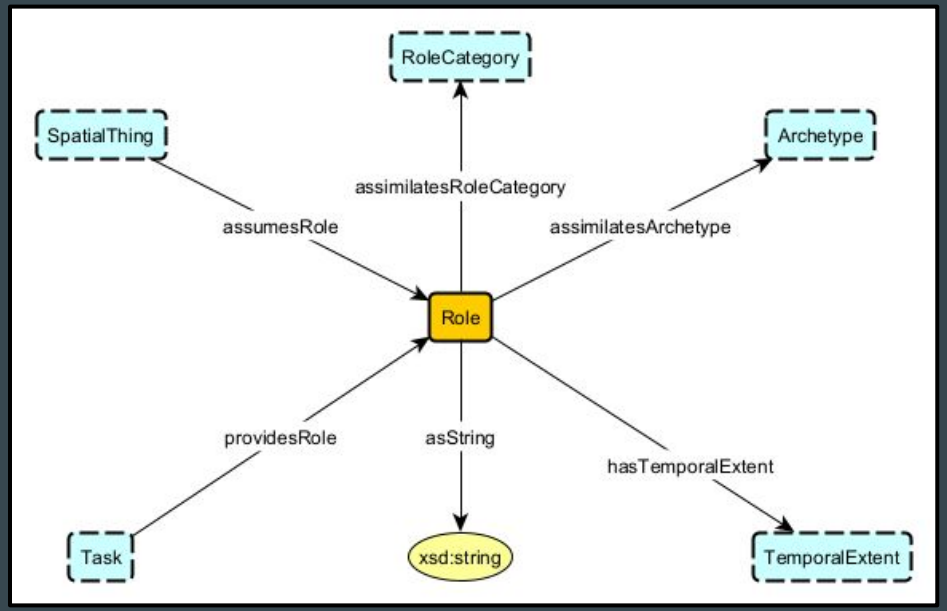
- Axiom 7
  - Hauler SubClassOf Archetype
  - Every Hauler is an Archetype.
- Axiom 8
  - Digger SubClassOf Archetype
  - Every Digger is an Archetype.
- Axiom 9
  - Archetype SubClassOf requiresCapability only Capability
  - The scoped range of requiresCapability, scoped by Archetype, is Capability.

Archetype Axioms

- Axiom 10
  - Archetype SubClassOf requiresCapability some (min 1) Capability
  - Every Archetype has at a minimum 1 Capability.
- Axiom 11 - 18
  - “Archetype SubClass” DisjointWith “Archetype SubClass.”
  - All Archetype SubClasses are all mutually exclusive Archetypes.

Role Pattern

- **Source Pattern** : agent-role-pattern, role-dependent-name, ONTOPRET
- **Source Data** : None
- **Description** : To represent the immediate characteristics and participation of a SpatialThing in a specific context



Note: Role pattern and Axioms are a work in progress!

Role Axioms

- Axiom 1
  - Role SubClassOf asString max 1 xsd:string
  - A Role can have at most 1 string.
- Axiom 2
  - Role SubClassOf hasTemporalExtent some TemporalExtent
  - Every Role has at a minimum 1 TemporalExtent..
- Axiom 3
  - Role SubClassOf hasTemporalExtent max 1 TemporalExtent
  - Every Role has at a most 1 TemporalExtent..

Pattern

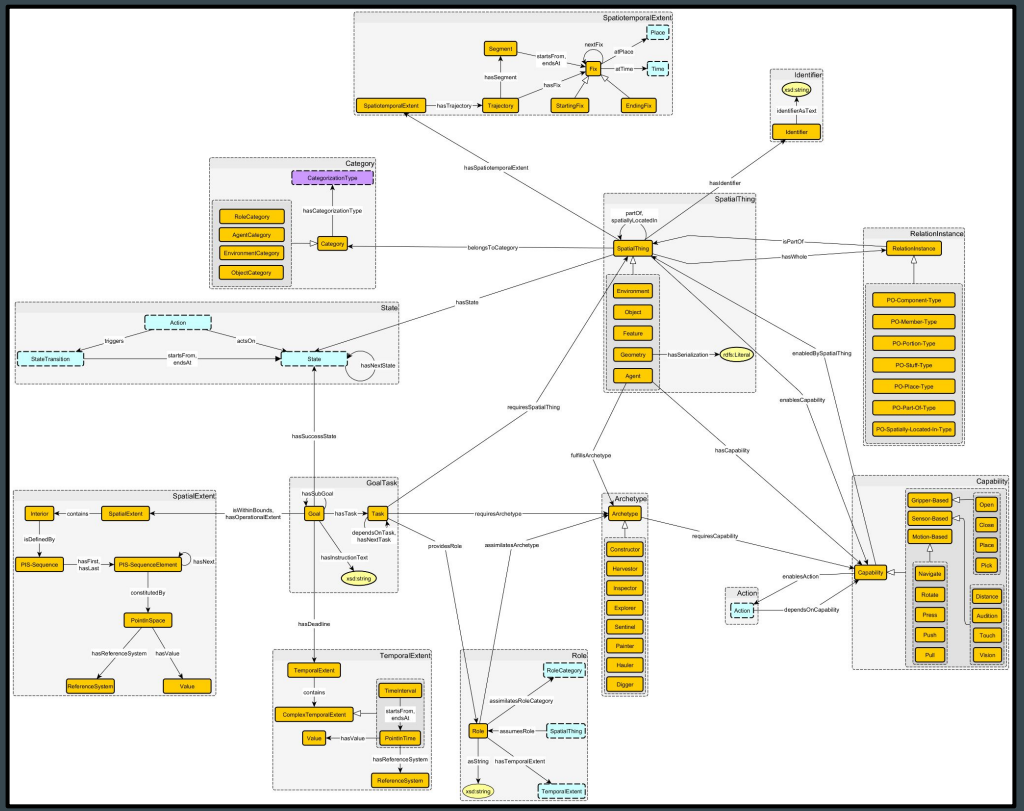
-

Axioms

Relevant Axioms

- Axiom 1
- Axiom 2

Schema


SpatiotemporalExtent
Segment
Place
Fix
Time
startsFrom, endsAt
nextFix
atPlace
atTime
hasSegment
hasFix
SpatiotemporalExtent
hasTrajectory
Trajectory
StartingFix
EndingFix
Identifier
xsd:string
identifierAsText
Identifier
hasSpatiotemporalExtent
hasIdentifier
Category
CategorizationType
RoleCategory
AgentCategory
EnvironmentCategory
ObjectCategory
hasCategorizationType
Category
belongsToCategory
SpatialThing
partOf, spatiallyLocatedIn
SpatialThing
isPartOf
hasWhole
RelationInstance
RelationInstance
PO-Component-Type
PO-Member-Type
PO-Portion-Type
PO-Stuff-Type
PO-Place-Type
PO-Part-Of-Type
PO-Spatially-Located-in-Type
Environment
Object
Feature
Geometry
Agent
hasSerialization
wkt, Literal
enabledBySpatialThing
State
Action
triggers
actsOn
StateTransition
startsFrom
endsAt
State
hasNextState
hasState
requiresSpatialThing
enablesCapability
hasSuccessState
fulfillsArchetype
hasCapability
GoalTask
hasSubGoal
Goal
hasTask
Task
dependsOnTask
hasNextTask
hasInstructionText
xsd:string
requiresArchetype
providesRole
assimilatesArchetype
Archetype
Archetype
Constructor
Harvester
Inspector
Explorer
Sentinel
Painter
Hauler
Digger
requiresCapability
Capability
Gripper-Based
Sensor-Based
Motion-Based
Open
Close
Place
Pick
Navigate
Rotate
Press
Push
Pull
Distance
Audition
Touch
Vision
enablesAction
dependsOnCapability
Action
Action
SpatialExtent
Interior
contains
SpatialExtent
isWithinBounds, hasOperationalExtent
isDefinedBy
PIS-Sequence
hasFirst
hasLast
PIS-SequenceElement
hasNext
constitutedBy
PointInSpace
hasReferenceSystem
hasValue
ReferenceSystem
Value
hasDeadline
TemporalExtent
TemporalExtent
contains
ComplexTemporalExtent
TimeInterval
startsFrom, endsAt
Value
hasValue
PointInTime
hasReferenceSystem
ReferenceSystem
Role
RoleCategory
assimilatesRoleCategory
Role
assumesRole
SpatialThing
asString
hasTemporalExtent
xsd:string
TemporalExtent

Future Work

- Refine Role and Capability
- Axioms for the remainder of the modules
- Materialization
- Validation across CQ with SPARQL queries

References

[1] A. Olivares-Alarcos, S. Foix, S. Borgo, N. G. Alenyà, Ocra – an ontology for collaborative robotics and adaptation, Computers in Industry 138 (2022) 103627–103627. doi:10.1016/j.compind.2022.103627.

[2] E. Aguado, V. Gomez, M. Hernando, C. Rossi, R. Sanz, A survey of ontology-enabled processes for dependable robot autonomy, Frontiers in Robotics and AI 11 (2024). doi:10.3389/frobt.2024.1377897.

[3] H. Du, S. Thudumu, R. Vasa, K. Mouzakis, A survey on context-aware multi-agent systems: Techniques, challenges and future directions, 2024. URL: https://arxiv.org/abs/2402.01968v2.

[4] A. Ricci, M. Piunti, M. Viroli, Environment programming in multi-agent systems: an artifact-based perspective, Autonomous Agents and Multi-Agent Systems 23 (2010) 158–192. doi:10.1007/s10458-010-9140-7.

[5] H. Li, P. Goncalves, V. Ragavan, A. Olivares-Alarcos, E. M. Barreto, D. Beβler, J. Bermejo, S. Borgo, A. Chibani, J. Carbonera, M. Diab, S. Fiorini, A. Gyrard, M. Habib, A. Khamis, K. Moulouel, H. Nakawala, B. Nguyen, C. Nowak, J. Olszewska, E. Pignaton, E. Prestes, J. Quintas, S. Redfield, R. Sanz, C. Schlenoff, E. Tosello, Ieee standard for autonomous robotics (aur) ontology (2021). URL: https://ieeexplore.ieee.org/document/9774339?utm_source=chatgpt.com. doi:10.1109/ieeestd. 2022.9774339.

References

[6] X. Sun, Y. Zhang, J. M. Chen, Rtpo: A domain knowledge base for robot task planning 8 (2019) 1105–1105. doi:10.3390/electronics8101105.

[7] C. Z. Sprenger, J. Antonio, N. U. Baier, Orpp—an ontology for skill-based robotic process planning in agile manufacturing, Electronics 13 (2024) 3666–3666. URL: https://www.mdpi.com/2079-9292/13/18/3666. doi:10.3390/electronics13183666.

[8] L. Kinder, T. Käfer, Static and adaptive planning with wot td by generating python objects as intermediary representations using large language models, Lecture notes in computer science (2025) 256–273. doi:10.1007/978-3-031-78955-7_21.

[9] W. W. W. C. (W3C), Owl 2 web ontology language primer (second edition), URL: https://www.w3.org/TR/2012/REC-owl2-primer-20121211/.

References

[10] C. Shimizu, Q. Hirt, P. Hitzler, Modl: A modular ontology design library, 2019. URL: https://arxiv.org/abs/1904.05405.

[11] C. Shimizu, K. Hammar, P. Hitzler, Modular ontology modeling, Semantic Web (2022) 1–31. doi:https://doi.org/10.3233/sw-222886.

[12] A. Hogan, E. Blomqvist, M. Cochez, C. D'amato, G. D. Melo, C. Gutierrez, S. Kirrane, J. E. L. Gayo, R. Navigli, S. Neumaier, A.-C. N. Ngomo, A. Polleres, S. M. Rashid, A. Rula, L. Schmelzeisen, J. Sequeda, S. Staab, A. Zimmermann, Knowledge graphs, ACM Computing Surveys 54 (2022) 1–37. doi:10.1145/3447772.

[13] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang and J. Xiao, 3D ShapeNets: A Deep Representation for Volumetric Shapes, CVPR 2015. URL: https://openaccess.thecvf.com/content_cvpr_2015/papers/Wu_3D_ShapeNets_A_2015_CVPR_paper.pdf