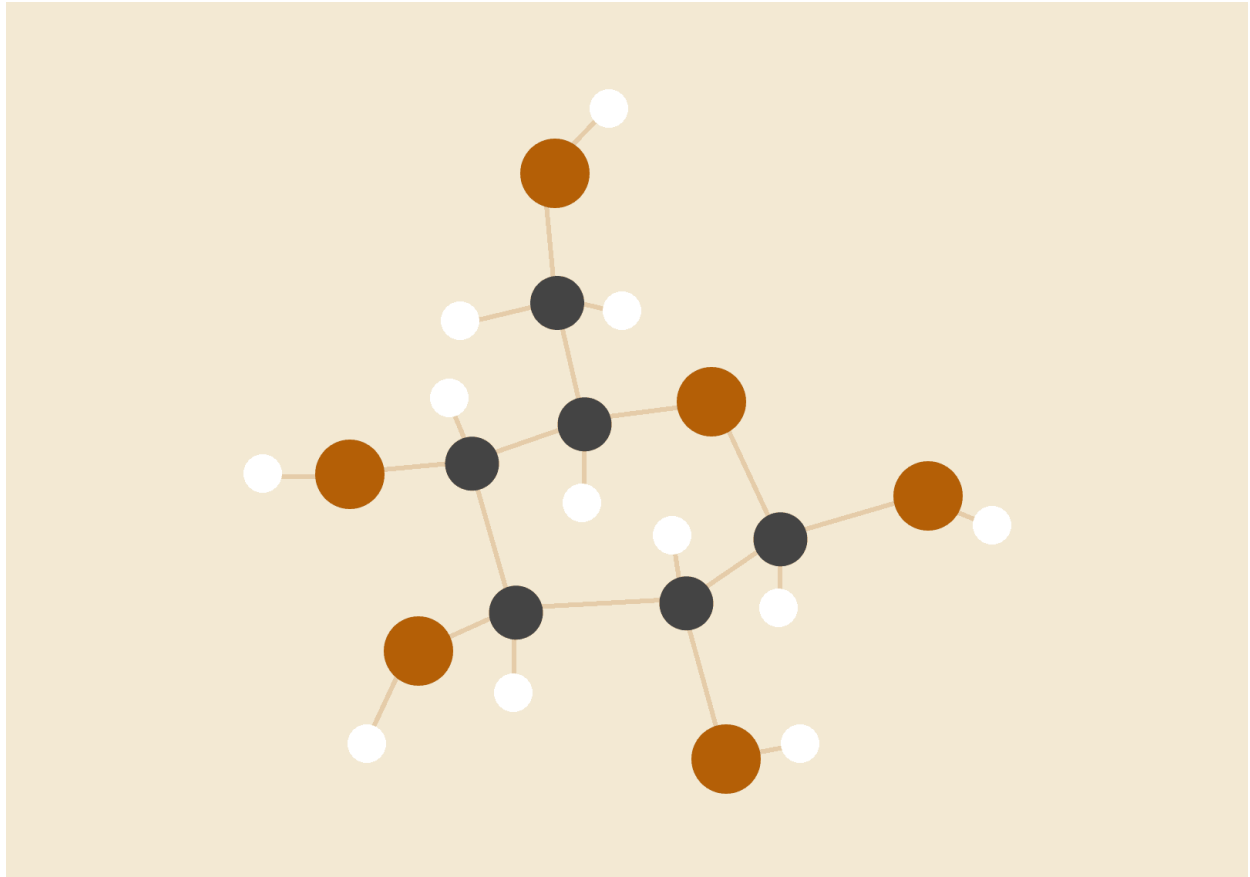


Domestic Cat Ontology Report



Participants of our Group

- ❖ 200734G - N.W.G.Wikumjith
- ❖ 200638P - N.H.D.Tharinda
- ❖ 200440C - Pahasara G.G.V.
- ❖ 200665U - Tissera K.A.U.

INTRODUCTION

According to the definition [1], "an ontology is an explicit specification of a conceptualisation, " which simply means categorising things in a specific domain by their properties.

'Domestic Cats' is the topic of the ontology developed in this report. Therefore the domain of this ontology is the representation of domestic cats. The scope of the ontology is limited to domestic cats in a pet shop, including all domesticated breeds of cats that are available for purchase at a cat store, as well as their physical and behavioural traits. Primarily focused on properties that most pet cat buyers demand.

Protégé is used to develop the ontology, which is freeware.

STEPS OF DEVELOPMENT

[2] is the primary reference in developing this ontology. It describes seven steps when creating an ontology. Those seven steps are as follows for the ontology developed in this report.

1) Determine the Domain and Scope of the Ontology

According to [2], the first step in developing an ontology is to determine the domain and scope of the ontology. In this case, the domain is domestic cats, and a pet shop setting is considered to reduce the domain's scope. The scope includes all domesticated breeds of cats available for purchase at the store and their physical and behavioural traits.

We decided on this domain and the scope with the help of answering the following questions mentioned in [2].

1. What is the domain that the ontology will cover?
 - The ontology covers the domain of domestic cats.
2. For what are we going to use the ontology?
 - The ontology is used to choose the appropriate cat for a customer buying a cat from a pet shop.
3. For what types of questions the information in the ontology should provide answers(competency questions)?
 - What breeds are highly playful?
 - Who are the cats tolerant of other pets in the house?
 - How much grooming is necessary for a cat?
 - What cat breeds like to be outdoors?
 - What cat breeds have a long tail?
 - Who are the cats with hairless coats?
 - What are the breeds that require less grooming?
 - What is the breed of this cat?
4. Who will use and maintain the ontology?
 - The pet shops will use the ontology, and each pet shop will maintain the ontology.

2) *Consider Reusing Existing Ontologies*

For the development of this ontology, we did not reuse any existing ontologies as we could not come across any ontology with the same domain and scope. However, we did refer to several online resources and books that provided information about domestic cats, including [3-8]. These resources provided valuable background information and helped to ensure that our ontology was accurate and comprehensive.

3) *Identifying Important Terms*

After determining the domain and scope mentioned above and deciding whether to use existing ontologies, the next step is identifying the important terms used in the determining domain and scope. For that, we created a list of terms a customer would ask when visiting a pet shop to buy an animal as a pet.

For example, if a customer visits a pet shop to buy a cat as a pet, some potential questions would be,

- What colours are there in this Persian breed?
- Do they have long hair only, or do they have short hair?
- What kind of health issues will they face?
- What is the eye colour of this breed?
- Are there any breeds with odd-coloured eyes?
- Who are the most affectionate cats?

Based on these questions and assuming all sorts of potential questions customers would ask, a list of terms was suggested, and some of those terms are listed below.

Breed	Price	Health Issues
Coat Colour	Weight	Playfulness
Eye Colour	Gender	Age
Coat Colour Pattern	Tail Shape	Body Shape
Genetic Diseases	Aggressiveness	Affection

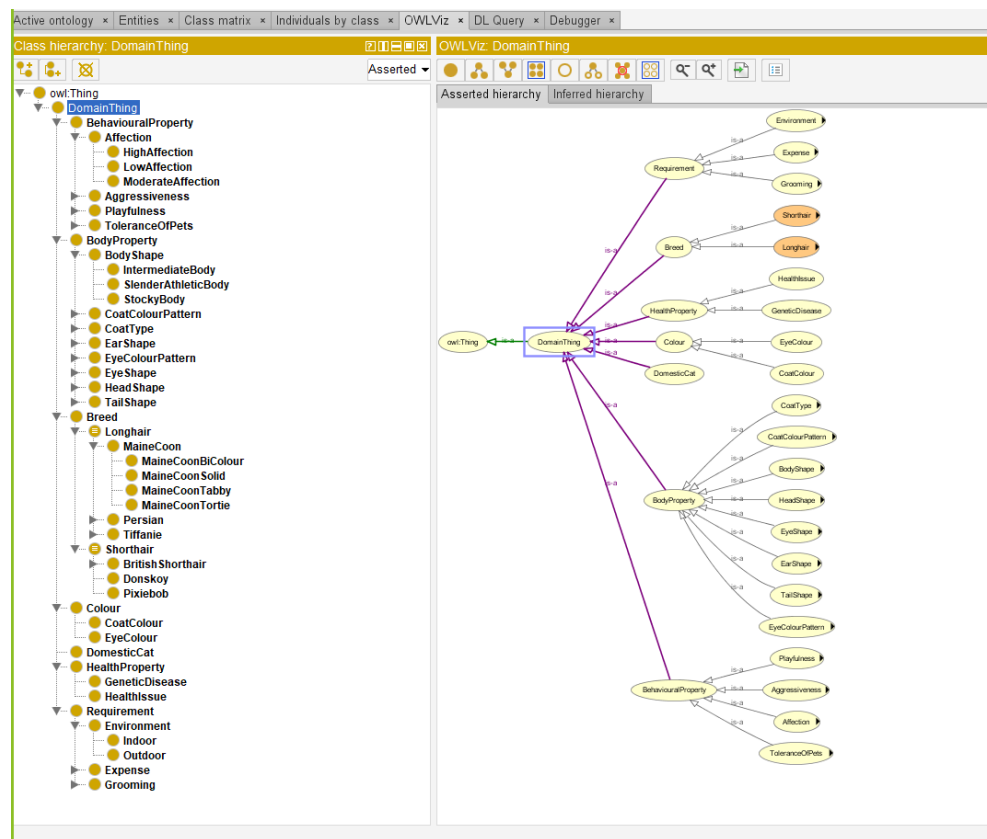
4) *Defining Classes and Developing Class Hierarchy*

When developing the class hierarchy, we mostly used the Top Down method. We started building the hierarchy by first deciding the main subclasses and then going down into defining and deciding on subclasses level by level manner.

The class hierarchy we came up with for this ontology is as follows:

- DomainThing is the top-level class, representing all things within the ontology domain.
- Domestic Cat is a subclass of DomainThing, representing all domestic cats within the domain.
- The breed is a subclass of DomainThing, representing the different breeds of domestic cats. There are subclasses for Short Hair and Longhair breeds, which are further divided into subclasses for specific breeds (e.g. American Shorthair, Persian).
- BehaviouralProperty is a subclass of DomainThing, representing the behavioural characteristics of domestic cats. There are subclasses for Playfulness, Aggressiveness, Affection, and Tolerance of Pets, each of which is further divided into subclasses for Low, Moderate, and High levels of the trait.

- BodyProperty is a subclass of DomainThing, representing the physical characteristics of domestic cats. There are subclasses for Coat Type, Coat Colour Pattern, Body Shape, Tail Shape, Head Shape, Ear Shape, Eye Shape, and Eye Colour Pattern, each of which is further divided into subclasses for specific characteristics (e.g. Short Coated, Solid Colour Pattern, Slender Athletic Body, Pointed Tips Ears).
- The Requirement is a subclass of DomainThing, representing the requirements when caring for a domestic cat. There are subclasses for Environment, Grooming, and Expense, each of which is further divided into subclasses as Indoor and Outdoor for Environment and the levels of Grooming and Expense needed as Low, Moderate and High.
- HealthProperty is a subclass of DomainThing, representing the health-related characteristics of domestic cats. There are subclasses for Genetic Disease and Health Issues.



When deciding the classes and subclasses, an issue arose whether to make the properties such as aggressiveness, playfulness, and grooming requirement an object property(slot) with slot values {"High Aggressiveness", "Moderate Aggressiveness", "Low Aggressiveness"}. However, we decided against that possibility and implemented separate classes for those. For example, the property of Aggressiveness is a subclass of BehavioralProperty as the HighAggressiveness, ModerateAggressiveness, and LowAggressiveness are the subclasses of Aggressiveness. By doing this, we can easily categorise and refer to the aggressiveness of a subclass of a Breed or an instance of DomesticCat.

5) *Defining the Properties of Classes(Slots)*

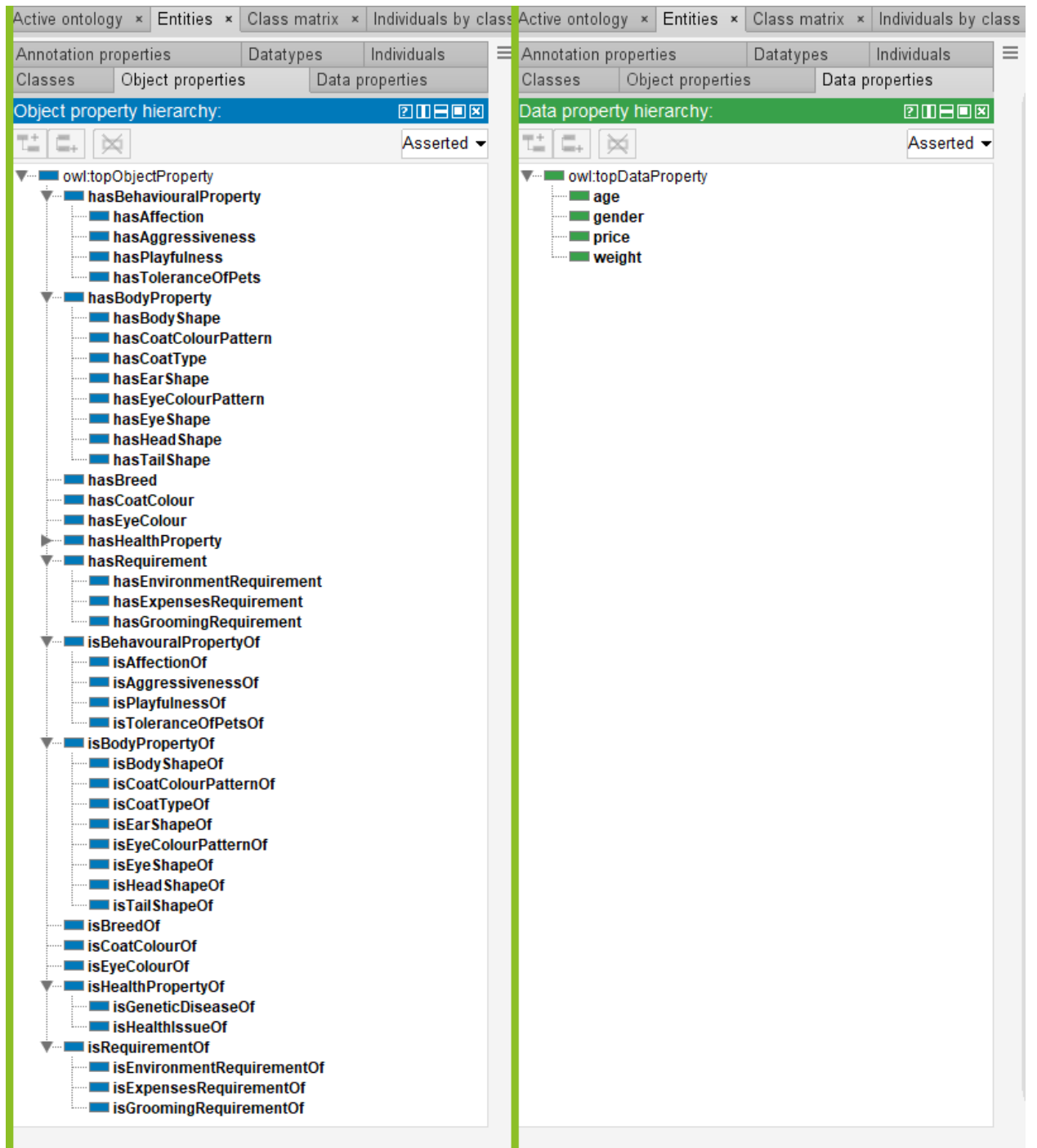
As mentioned in [2], the class hierarchy alone cannot answer the competency questions mentioned in the first development step. For example, in the competency question: “What breeds are highly playful?” Property playfulness has to be defined by the breed class and its subclasses. Consider another competency question: “Who are the highly playful cats?”. Since we defined playfulness as a breed property, a problem arises: How can an instance of a DomesticCat class be associated with Playfulness? To resolve that, we define the property to which breed the cat belongs. Then through the breed, the playfulness of a specific cat can be known.

Through extensive research on domestic cats using various references[3-6], it is decided that the properties except the CoatColour can be attributed to a breed as specific. For example, the property CoatType, TailType, Aggressiveness, and GroomingRequired is specific to the Persian breed. However, the Persian breed has subbreeds with different CoatColourPatterns, EyeColourPattern, and EyeColours. Therefore the properties, CoatColourPatterns and EyeColourPattern and EyeColours are defined by the subclasses(subbreeds) of the Persian class(breed). In contrast, the properties common to all subbreeds of Persian are defined by the Persian class.

Slots are two types in Protégé. They are object properties and data properties. Object properties can be associated with classes and instances, whereas data properties can only be associated with instances of classes. The above-discussed properties are all object properties. For data properties, which are used to differentiate instances, age, gender, price and weight properties are defined by an instance of a DomesticCat class. An instance of DomesticCats also defines the CoatColour object property value because a cat's coat colour cannot be categorised by any means.

Some of the object properties(slots) we defined are given below.

hasBreed	hasEyeColourPattern	hasHealthIssues
hasCoatColour	hasGroomingRequiremnt	hasPlayfulness
hasEyeColour	hasGender	hasToleranceOfPets
hasCoatColourPattern	hasTailShape	hasBodyShape
hasGeneticDiseases	hasAggressiveness	hasAffection



6) Defining the Facets of Slots

Some facets are slot cardinality, slot value type, allowed values for the slot, domain and range. Let us look at how facets are defined for the slots in the ontology. The value type of `hasCoatColour`, age, price, weight, and gender slots is defined as an instance of `CoatColour` class, `Enumeration`, `real value`, `real value`, and `enumeration`, respectively. The allowed values for the age slot are {"Kitten", "Adult", "Junior", "Senior"}. The allowed values are defined for other relevant slots, such as gender. The domain for the slots such as `hasBodyShape`, `hasTailShape`, and `hasAggressiveness` is the `Breed` because a breed has those properties as a common trait. As for the range of slots, relevant ranges such as `Agressiveness` class for `hasAgressiveness`, `Playfulness` for `hasPlayfulness` and `BodyType` for `hasBodyType` is defined. For cardinality, a `Breed` has only one Affection characteristic, either `HighAffection`, `ModerateAffection` or `LowAfferction`. A `DomesticCat` only has one `Breed`. However, the `CoatColour` of a `DomesticCat` can have multiple colours. An example of slot cardinality is provided below.

The screenshot displays the Protégé ontology editor interface. On the left, the 'Class hierarchy' panel shows a tree structure starting from 'owl:Thing'. The hierarchy includes 'DomainThing', 'BehaviouralProperty', 'BodyProperty', 'Breed', 'Longhair', 'MaineCoon', 'MaineCoonBiColour' (highlighted), 'MaineCoonSolid', 'MaineCoonTabby', 'MaineCoonTortie', 'Persian', 'Tiffanie', 'Shorthair', 'Colour', 'DomesticCat', 'HealthProperty', and 'Requirement'.

The right panel shows the 'MaineCoonBiColour' class details. The 'Description' tab is active, showing the class name 'MaineCoonBiColour'. Below the description, there are sections for 'Equivalent To', 'SubClass Of', and 'General class axioms'.

The 'SubClass Of' section lists several axioms for the 'MaineCoonBiColour' class:

- `hasCoatColourPattern` **only** BiColour
- `hasCoatColourPattern` **only** PartiColourPattern
- `MaineCoon`

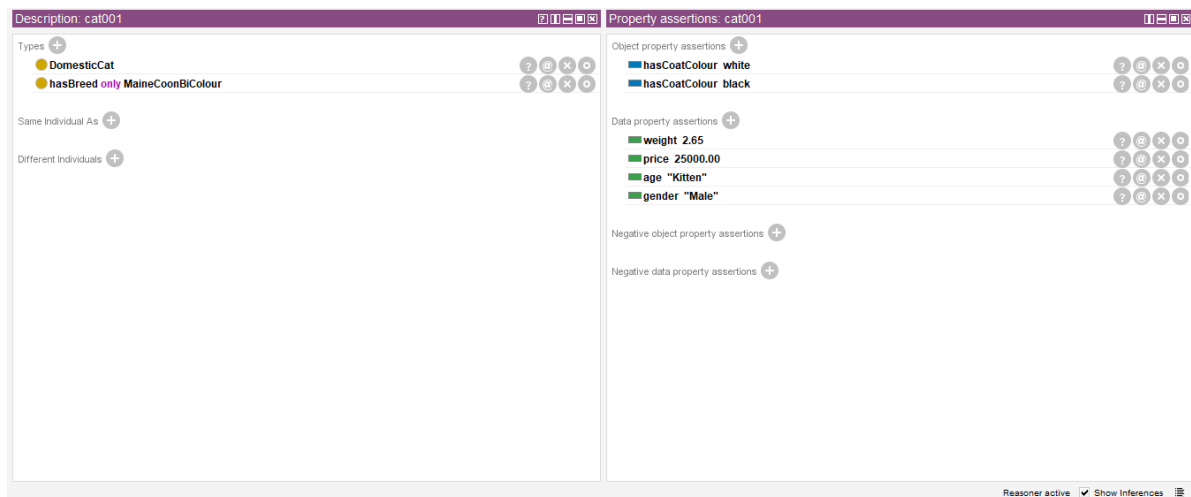
The 'General class axioms' section lists various axioms for the class, including:

- `hasEarShape` **exactly 1** EarShape
- `hasHeadShape` **exactly 1** HeadShape
- `hasEyeShape` **exactly 1** EyeShape
- `hasAffection` **exactly 1** Affection
- `hasExpensesRequirement` **exactly 1** Expense
- `hasEnvironmentRequirement` **exactly 1** Environment
- `hasEyeColourPattern` **exactly 1** EyeColourPattern
- `hasPlayfulness` **exactly 1** Playfulness
- `hasTailShape` **exactly 1** TailShape
- `hasBodyShape` **exactly 1** BodyShape
- `hasCoatColourPattern` **exactly 1** CoatColourPattern
- `hasCoatType` **exactly 1** CoatType
- `hasGroomingRequirement` **exactly 1** Grooming
- `hasAggressiveness` **exactly 1** Aggressiveness
- `hasToleranceOfPets` **exactly 1** ToleranceOfPets
- `hasEarShape` **only** RoundedTipsEars
- `hasAffection` **only** HighAffection
- `hasGroomingRequirement` **only** HighGrooming
- `hasAggressiveness` **only** LowAggressiveness
- `hasEyeColour` **value** copper
- `hasEyeColourPattern` **some** EvenColouredEyes
- `hasCoatColourPattern` **only** (PartiColourPattern **or** SolidColourPattern **or** TabbyPattern **or** TortieColourPattern)

7) *Creating the Instances*

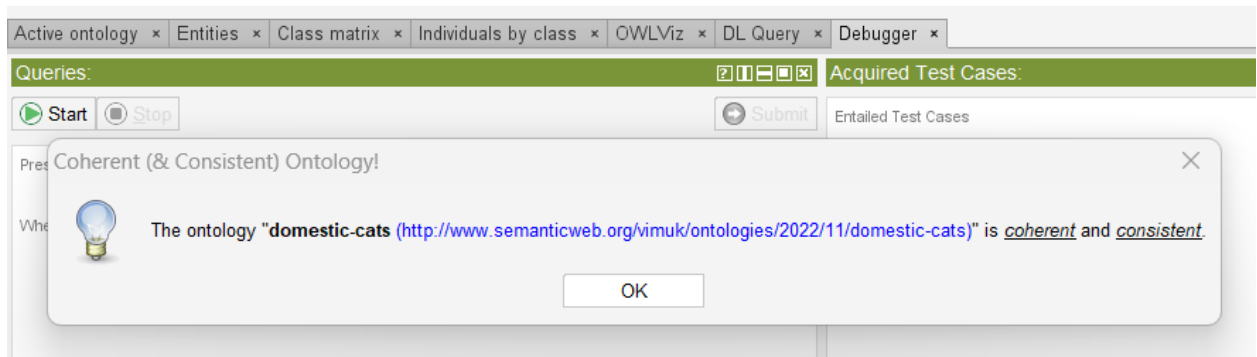
The only classes that can have instances are Colour, GeneticDisease, HealthIssue and DomesticCat; we implemented several instances for demonstration purposes. Some instances created for the Colour class are black, blue, brown, copper and green. Since CoatColour class and the EyeColour class are Colour's subclasses, the Colour's instances are assigned for CoatColour and EyeColour as suitable. For instance, the green instance can only be an instance of EyeColour, whereas the golden instance can be an instance of both CoatColour and EyeColour. Like the previous example, there are no common instances of GeneticDisease and HealthIssue. Some of the instances for GeneticIssue are feline_lower_urinary_tract_disease, polycystic_kidney_disease, hypertrophic_cardiomyopathy, and manx_syndrome. eye_problem and respiratory_problem are instances of HealthIssue. All of the above-discussed examples of instances do not have slot values.

However, the instances of DomesticCat have slot values. One such instance and its slot value declarations are shown below.



8) *Analysing and Debugging the Ontology*

After the ontology has been developed, it is important to analyse and debug it to ensure that it is accurate and complete. For this, we checked all sorts of possible axioms. For example, Persian is a Longhair represented correctly in the ontology. Then Longhair is a breed which implies that the Persian is also a correct breed. For another example, Persian is a subclass of LowAggressiveness, which is true. In the Protégé desktop version, there is also the option of using a Reasoner while developing the ontology to check whether the ontology is logically correct. We used HermiT reasoner while developing the ontology, corrected the logical errors once they appeared, and continued the development. Therefore the domestic cat ontology is free of logical errors that hermiT reasoner checks. However, there is no guarantee that ontology has all the logical connections we are meant to make. To check for errors and debug further, we used the debugger in the Protégé. We got the result that our ontology is coherent and consistent.



We also reviewed the definitions of classes and properties, created additional instances to test the ontology, and made necessary changes to improve the accuracy and completeness of the ontology.

OTHER CONSIDERATIONS

1) *Naming of Classes, Slots and Instances*

For the naming of the classes, camelcase with the first letter is also capitalised style is used. For example, CoatColour, BodyShape, GroomingRequired. For naming slots, the camel case style is used. For example, hasAggressiveness, hasGroomingRequirement, and hasBodyType. For naming instances, simple letter words connected by underscores style is used. For example progressive_retinal_atrophy, scottish_fold_disease, inflammatory_bowel_disease. Also, for naming consistency, singular nouns and verbs are used as class and subclass names, slot names and instance names.

2) *Inverse Slots*

There are inverse slots for most of the ontology for easy backward referencing. For example, Persian hasAffection Highaffection. Then for the inverse, HighAffection isAffectionOf Persian. By implementing inverse slots, answering question pairs such as “what breeds has high affection?” and “what is the affection of the Persian breed?” will become easy.

3) *Disjoint Classes*

There are disjoint classes in the domestic cat ontology. Subclasses such as Affection, Playfulness, GroomingRequired, and BodyType are disjoint because a specific breed cannot have both the HighAggressiveness and LowAggressiveness for the Aggressiveness. Another example is that a specific cat breed cannot have both EvenColouredEyes and OddColouredEyes. However, the Breeds are not disjoint because many breeds have the LowAggressiveness for aggressiveness.

ASSUMPTIONS

In domestic cat ontology, characteristics such as playfulness, aggressiveness, affection and tolerance are considered characteristics of a certain **cat breed** other than taking these characteristics for **single instances of cats**. It was an assumption made to make the characteristics' categorisation simple and useful. For example, there is the property of high affection given to the Maine Coon cat breed, and it is considered that all Maine Coon cats have high affection. However, there may be exceptions when considering some cats in that breed.

CONCLUSION

In this report, we have summarised the steps, considerations, and assumptions followed during the development of domestic cat ontology. The scope chosen for the domain of domestic cats is the setting of a pet shop. The main requirement of the ontology is to filter the cats and domestic cat breeds according to the customers' requirements. To our knowledge, we developed the ontology to satisfy and cover the chosen scope.

The class hierarchy, object and data property hierarchy, and individuals implemented in the domestic cat ontology can answer most of the customer's questions before buying a cat as a pet. There are opportunities for further extending and improving the ontology by introducing a more sophisticated class hierarchy that uses explanatory subclasses for property classes. For example, instead of the level of affection as high, moderate and low, specify what the activities of showing affection are, such as using the lap as a pillow, following the owner, sleeping with the owner and such.

This ontology could support various applications, such as providing information to cat store customers or helping to classify and organise cats within a cat store setting which was the purpose of developing this ontology.

The developed domestic cat ontology can be accessed [here](#).

REFERENCES

- [1] "What is an Ontology?," *What is an Ontology?* <http://www-ksl.stanford.edu/kst/what-is-an-ontology.html> (accessed Dec. 27, 2022).
- [2] N. F. Noy and D. L. McGuinness, "Ontology development 101: A guide to creating your first ontology," ed: Stanford knowledge systems laboratory technical report KSL-01-05 and ..., 2001.
- [3] K. Bryan, *The Complete Cat Breed Book: Choose the Perfect Cat for You*. DK Publishing, 2021.
- [4] D. Kindersley, "The Cat Encyclopedia: The Definitive Visual Guide," ed: Dorling Kindersley Publishing, 2014.
- [5] "Life stages | Cat Care for Life," *Life stages | Cat Care for Life*. <https://www.catcare4life.org/cat-owners/lifestages/#:~:text=The%20average%20lifespan%20for%20a,or%2030%20years%20of%20age>. (accessed Dec. 27, 2022).
- [6] Omlet, "Cat Coat Colours And Patterns | Choosing The Right Cat For You | Cats," *Cat Coat Colours And Patterns | Choosing The Right Cat For You | Cats*. https://www.omlet.co.uk/guide/cats/choosing_the_right_cat_for_you/cat_coat_colours_and_patterns/ (accessed Dec. 27, 2022).
- [7] "<https://www.manhattan-puppies.com/>." <https://www.manhattan-puppies.com/> (accessed Dec. 27, 2022).
- [8] "Cats for Adoption | Petfinder," *Cats for Adoption | Petfinder*. https://www.petfinder.com/search/cats-for-adoption/?sort%5B0%5D=recently_added (accessed Dec. 27, 2022).