Ali Momenzadeh kholejani – Pawel Staszynski - INM713, Course Project Part 1

TASK OWL.A

We created a Pizza class with the NamedPizza under it. CaliforniaPizza and HawaiianPizza are two of the sample pizza we have choose and then added their corresponding toppings below the PizzaTopping class. One important decision was to make some of the toppings in the pizzas existential (like CheddarCheese and Muchroom in CaliforniaPizza), while some are not (like ChikenBreast in CaliforniaPizza). The reason for this is that in our dataset, we compared different CaliforniaPizza in different venues and came across this fact that some ingredients are necessary parts of the CaliforniaPizza while some others are not. For modeling the full address of venues, we created a location class and added the postcode as data property (hasPostCode) to that class. We used the same approach for modeling the Price and then added hasValue and isCurrency as data properties to the Price class. Finally, different types of places are defined as subclasses of Venue. We also created a new pizza concept, PizzaDaAli, that is served in PizzeriaDaAli venue. This venue serves alcohol too, as defined by sellsAlcohol object property.

TASK OWL.B

We have chosen the Music domain for this part. The following axioms are written in Manchester syntax.

Subtask OWL.B.1 An axiom with an atomic subsumption.

Class: Jazz

SubClassOf: MusicGenre

The class Jazz is a subclass of the class MusicGenre. -> Jazz is a specific type of music genre.

****

Subtask OWL.B.2 An axiom with an universal restriction.

Class: Instrument

SubClassOf: hasSound some Sound

Every instance of the class Instrument has some property “hasSound” with at least one instance of the class Sound. -> Every instrument produces sound.

****

Subtask OWL.B.3 An axiom with an existential restriction.

Class: Musician

SubClassOf: playsInstrument some Instrument

Every instance of the class Musician plays at least one instrument. -> Being a musician entails playing an instrument.

****

Subtask OWL.B.4 An axiom with a union.

Class: MetalOrRock

EquivalentTo: Metal or Rock

MetalOrRock is the union of the classes Metal and Rock. -> An instance of MetalOrRock can be either a metal music piece or rock.

****

Subtask OWL.B.5 An axiom with an intersection.

Class: JazzMusician

EquivalentTo: Musician and Jazz

JazzMusician is the intersection of the classes Musician and Jazz. -> An instance of JazzMusician is both a musician and associated with jazz music.

****

Subtask OWL.B.6 An equivalence axioms with a cardinality restriction.

Class: SoloMusician

EquivalentTo: Musician

and (playsInstrument exactly 1 Instrument)

SoloMusician is equivalent to the class Musician and has the restriction that instance of SoloMusician must play exactly one instrument. -> A solo musician only plays one instrument.

****

Subtask OWL.B.7 An axiom stating two concepts cannot have a common instance.

DisjointClasses: RockBand, JazzBand

RockBand and JazzBand are disjoint classes that indicate they cannot have any common instances. -> A musical group cannot simultaneously be both a rock band and a jazz band.

****

Subtask OWL.B.8 A property chain axiom.

ObjectProperty: hasBandMember

PropertyChain: hasMember o partOfBand

The property hasBandMember is composed of two other properties: hasMember and partOfBand. This indicate that if A hasMember B and B is partOfBand C, then A hasBandMember C, implying a relationship between individuals in a band.

****

Subtask OWL.B.9 A role assertion axiom and a valid inverse.

hasBandMember: John, LedZeppelin

inverse(hasMember): LedZeppelin, John

John is a member of the band Led Zeppelin using the property hasBandMember. The valid inverse axiom states that Led Zeppelin has John as a member using the inverse property hasMember. This establishes a bidirectional relationship between John and Led Zeppelin as a band member.

****

Subtask OWL.B.10 A class assertion axiom where the class is complex.

Individual: John

Types: Musician and (playsInstrument some Guitar)

John is an individual who is both a musician and plays an instrument that is of type Guitar. This complex class assertion provides specific details about John's attributes and associations.

****

Subtask OWL.B.11 (optional) A combination of axioms that makes the ontology to be outside OWL 2.

ObjectProperty: playsSameInstrumentAs

PropertyChain: playsSameInstrumentAs o hasSuccessor

We can define a property chain based on the relationship between musician and their primary instrument. Then we try to assert that if musician A plays the same instrument as musician B, musician A is a successor of musician B. This will introduce a property chain on a functional property, which is not allowed in OWL 2 DL. "playsSameInstrumentAs" represents the relationship where two musicians play the same instrument, and "hasSuccessor" represents the successor relationship between musicians. However, "hasSuccessor" is assumed to be a functional property (each musician has at most one successor), and using it within a property chain violates the OWL 2 DL restriction. Therefore, this combination of axioms results in an ontology outside the OWL 2 DL specification.