**OWL-Specific Rules** 

# OWL - Part 1/2 Class Axioms, Class Assertions, Property Axioms, Property Types, Entailment Rules, Sheet 5 **CLASS AXIOMS** General • owl:Class • owl:Thing • owl:NamedIndividual owl:Nothing Analgous to a "select inverse" function Complement **Equivalence** owl:**equivalentClass** owl:complementOf The complement of a class contains all individuals that are **not** in the class. Equivalent classes contain the same individuals, and have the same definition. owl:equivalentClass ex:a rdf:type ex:A. ex:b rdf:type ex:B.

ex:B.

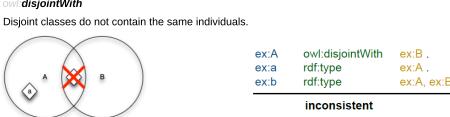
ex:A

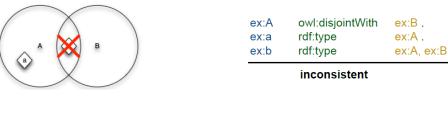
rdf:type

rdf:type

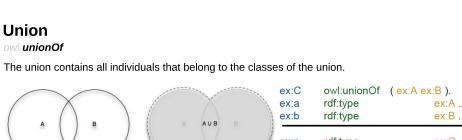
ex:a

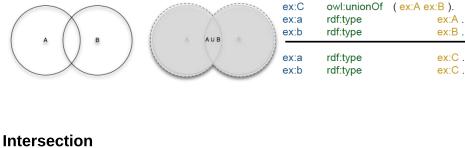


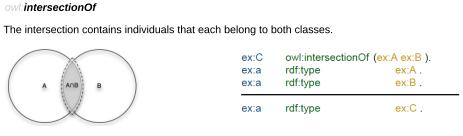


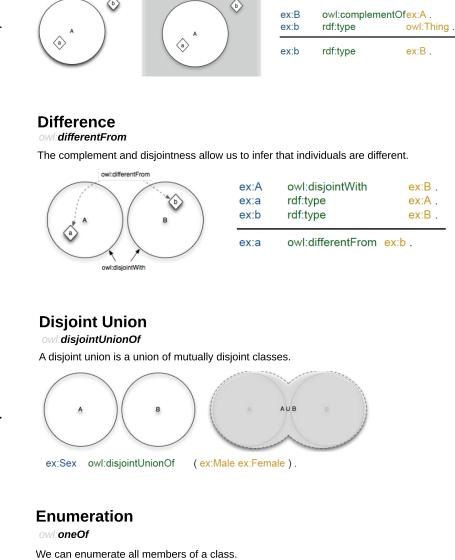


# ex:A, ex:B.









ex:A

ex:a1

ex:a 1

owl:oneOf

rdf:type

rdf:type

(ex:a1 ex:a2 ex:a3).

owl:Thing.

ex:A.

ex:MorningStar, ex:EveningStar .

ex:MorningStar owl:sameAs

ex:EveningStar owl:sameAs

owl:differentFrom

owl:differentFrom





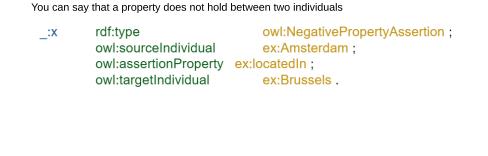
owl:sameAs

ex:Venus

Negation

built with special construct

**Inverse Properties** 

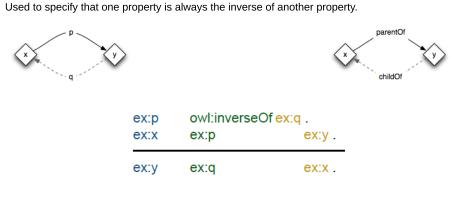


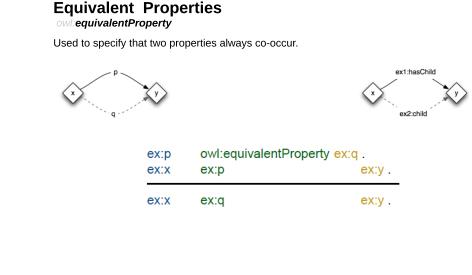
# ex:Venus, ex:EveningStar; ex:AlphaCentauri . ex:Venus, ex:MorningStar;

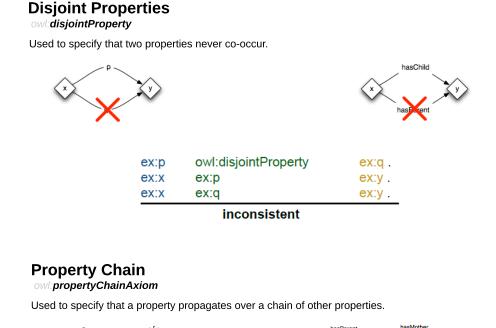
ex:AlphaCentauri .

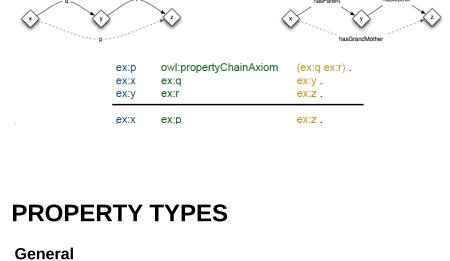
# owl:InverseOf

**PROPERTY AXIOMS** 









non-literals as

range

## owl:DatatypeProperty Have only literals owl:subPropertyOf as range owl:topDatatypeProperty

ex:x

ex:y ex:x

**Functional** 

owl:FunctionalProperty

Read as "Unique Rangeness"

ex:p

ex:p

ex:p

ex:y.

ex:z.

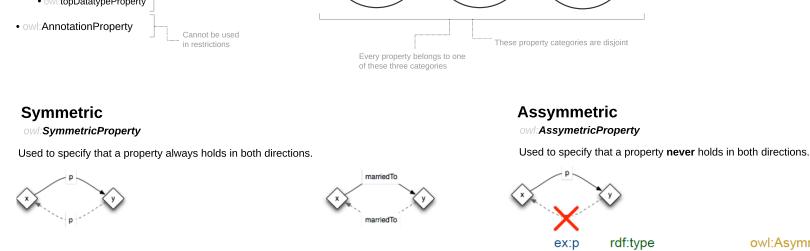
ex:z

Used to specify that a property has only one value for any particular instance.

owl:ObjectProperty

rdfs:subPropertyOf

owl:topObjectProperty



Object

Property

Datatype

Property

Annotation

Property

ex:p

ex:p

**Inverse Functional** 

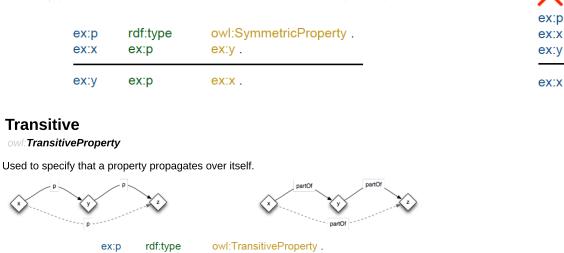
owl:InverseFunctionalProperty

ex:p

ex:x

ex:z

Read as "Unique Domainness"





Used to specify that a value for the property uniquely identifies an instance.

ex:y.

ex:y.

rdf:type

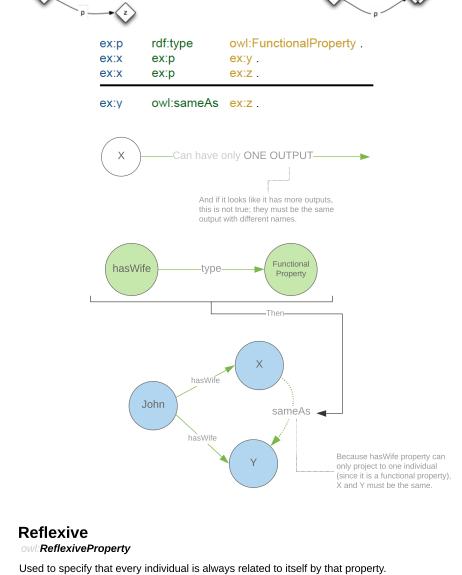
ex:p

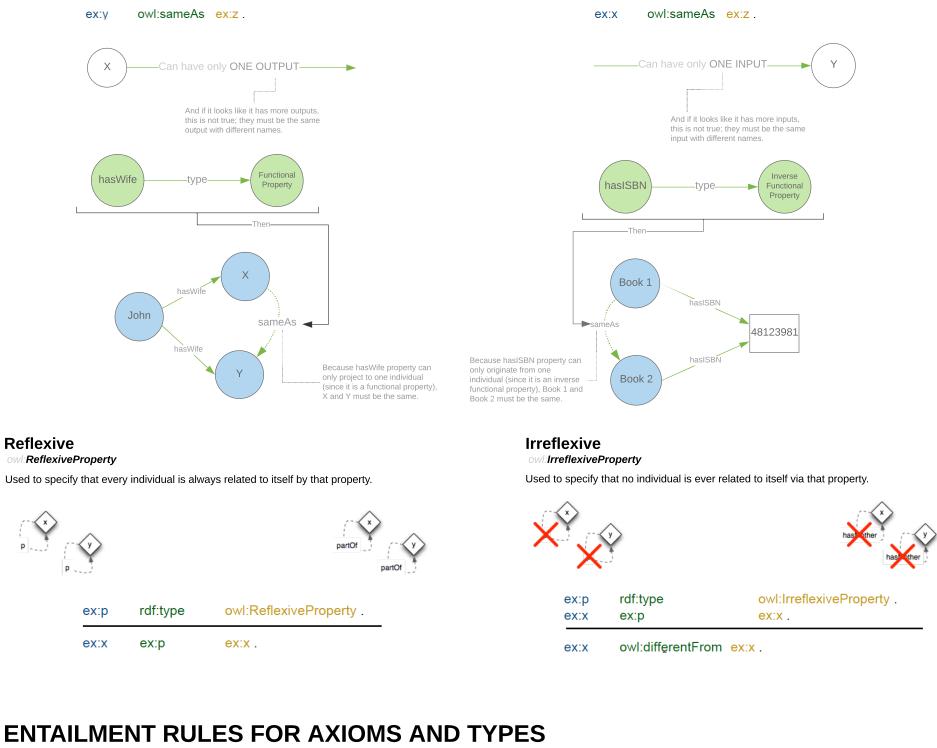
ex:p

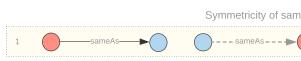
owl:AsymmetricProperty

owl:InverseFunctionalProperty.

ex:x.







rdf:type

ex:p

ex:x

owl:ReflexiveProperty.

ex:x.



rdfs:subPropertyOf q q rdfs:subPropertyOf p

owl:sameAs

owl:sameAs

derivable from axioms or known facts. "

7. p owl:equivalentProperty

p

x

p

W

rdf:type

p

p

rdf:type

10. r owl:propertyChainAxiom

owl:FunctionalProperty

owl: InverseFunctionalProperty

(p q)

**OWL-SPECIFIC RULES Open World Assumption** "Nothing is assumed to be true or false unless it is explicit knowledge or

behaves conservatively when classifying. Unless *sufficient* evidence required for classification is not specified (e.g., by using 'equivalentClass'), OWL would not automatically classify objects based on *necessity* criteria (i.e., which could have been established by using 'subClassOf'). For instance, meeting all the  $\emph{necessary}$  criteria for being a car by having four

OWL does not assume that the ontology contains all possible knowledge, and

wheels and the right size would not lead OWL to categorize a thing as a car (e.g., it may be a car model, but not an actual car), unless it is specifically stated that these two criteria is *sufficient* evidence for being categorized as a car.

# **Punning** OWL does not allow an entity to be used as class instances and objects at the same time (while RDF/S does). To make things a bit easier, thouh, OWL automatically infers an URI's type (i.e., whether it is a class instance or object) from the position of a URI in an axiom.

Sameness of domain due to "inverse functionality"

Novel relationship due to property chain

■ InverseFunctionalProperty

e.g., grandmotherOf

