OWL
Functional and Manchester Syntax

Syntactic forms for OWL

- RDF/XML:
  - standard XML syntax for RDF called RDF/XML
  - most commonly used syntax
- Other forms
  - Functional Syntax
  - Manchester Syntax
  - OWL/XML

OWL Functional Syntax

- Follows the structural specification and allows OWL 2 ontologies to be written as text
- More compact and human-readable than RDF/XML

```
#Classes
Declaration(Class(:Person))
Declaration(Class(:Lecturer))
SubClassOf(:Lecturer :Person)
Declaration(Class(:Course))

#Properties
Declaration(ObjectProperty(:teach))
ObjectPropertyDomain(:teach :Lecturer)
ObjectPropertyRange(:teach :Course)
```

OWL Functional Syntax (2)

- ClassAssertion and PropertyAssertion

```
#ClassAssertion and PropertyAssertion

#Individual
Declaration(NamedIndividual(:CO1003))
ClassAssertion(:Course  :CO1003)
ClassAssertion(:Student :S1)

#PropertyAssertation
DataPropertyAssertion
 (:course_name uol:CO1003 "Program Design"^^xsd:string)
ObjectPropertyAssertion(:has_registered_student :CO1003 :S1)
```

OWL Functional Syntax (3)

- ClassAssertion and PropertyAssertion

```
# a module can have at most 100 registered students

SubClassOf(:Course owl:Thing)
SubClassOf(:Course
      ObjectMaxCardinality(100 :has_registered_student))

# an interesting module is a module taught by good lecturer

EquivalentClasses(:InterestingCourse
ObjectIntersectionOf(:Course
    ObjectSomeValuesFrom(: taught_by:GoodLecturer)))


DifferentIndividuals(:CO1001 :CO1003 :CO1005 :CO1007
:CO1012 :CO1016))
```

OWL Manchester Syntax

- Follows the structural specification and allows OWL 2 ontologies to be written in a compact form

```
Class: :Course

    SubClassOf:
        owl:Thing,
        :has_registered_student max 100 owl:Thing

Class: :Lecturer

    SubClassOf:
        :Person

Class: :Student

    SubClassOf:
        :Person,
```

OWL Manchester Syntax (2)

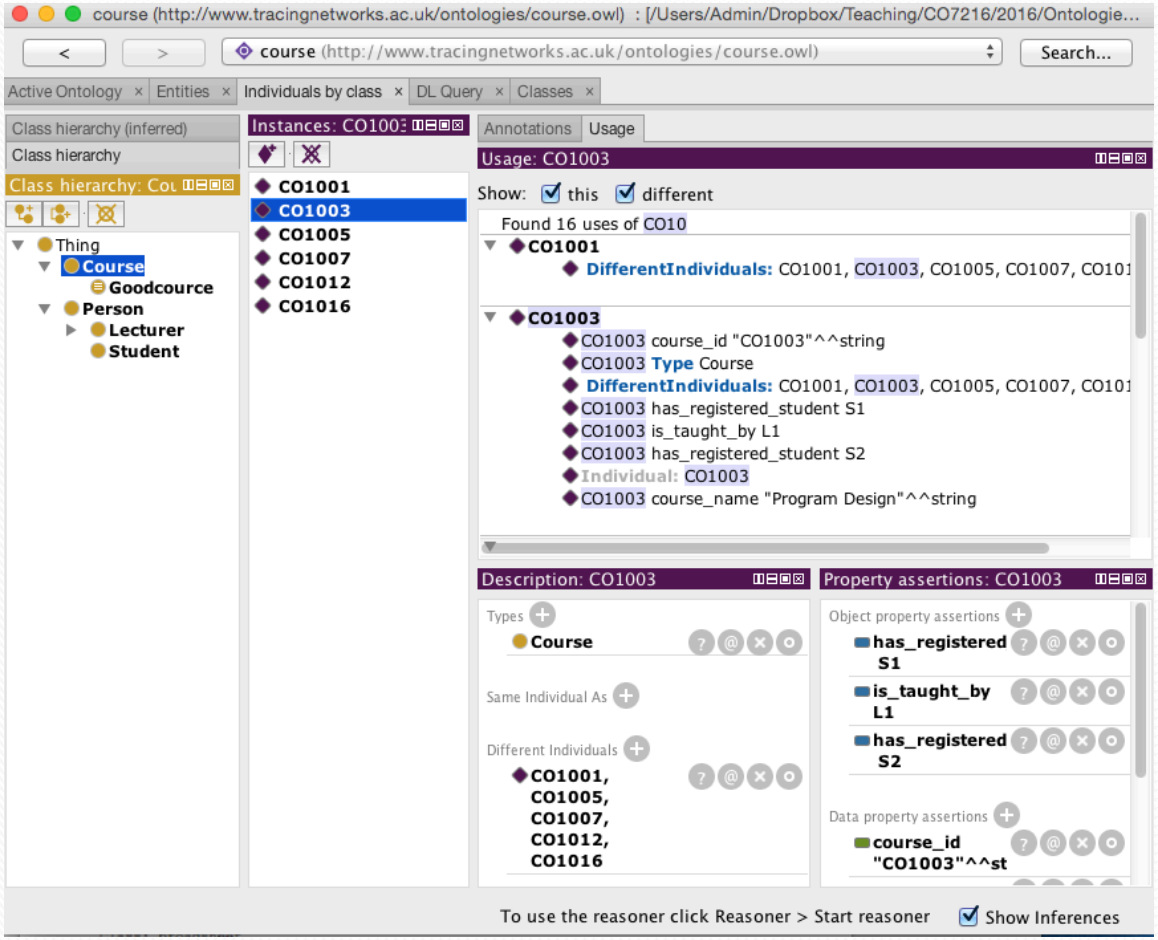
- Follows the structural specification and allows OWL 2 ontologies to be written in a compact form

```
Individual: :CO1003

    Types:
        :Course

    Facts:
     :has_registered_student  :S1,
     :is_taught_by  uol:L1,
     :course_id  "CO1003"^^xsd:string,
     :course_name  "Program Design"
      ^^xsd:string

Class: :InterestingCourse

    EquivalentTo:
        :Course
         and (:is_taught_by some :Goodcource)
```

```
DifferentIndividuals:

:CO1001,:CO1003,:CO1005,:CO1007,
:CO1012,:CO1016
```

OWL Manchester Syntax (3)

- Protégé 4/5 editor's layout resembles OWL OWL Manchester Syntax



OWL Manchester Syntax (4)

- Protégé 4/5 editor's layout resembles OWL OWL Manchester Syntax

OWL Constructor	DL Syntax	Manchester OWL S.	Example
intersectionOf	$C \sqcap D$	C **AND** D	Human **AND** Male
unionOf	$C \sqcup D$	C **OR** D	Man **OR** Woman
complementOf	$\neg C$	**NOT** C	**NOT** Male
oneOf	$\{a\} \sqcup \{b\}...$	{a b ...}	{England Italy Spain}
someValuesFrom	$\exists$ R C	R **SOME** C	hasColleague **SOME** Professor
allValuesFrom	$\forall$ R C	R **ONLY** C	hasColleague **ONLY** Professor
minCardinality	$\geq$ N R	R **MIN** 3	hasColleague **MIN** 3
maxCardinality	$\leq$ N R	R **MAX** 3	hasColleague **MAX** 3
cardinality	$=$ N R	R **EXACTLY** 3	hasColleague **EXACTLY** 3
hasValue	$\exists$ R {a}	R **VALUE** a	hasColleague **VALUE** Matthew

The Manchester OWL Syntax OWL Class Constructors

OWL Manchester Syntax (5)

- OWL Manchester Syntax Operators:
  - SOME
  - ALLVALUE
  - MIN
  - MAX
  - EXACTLY
  - AND (=THAT)
  - OR

OWL Manchester Syntax – Class Description

Syntax:

Class: :classID

SubClassOf:
ClassExpression
…
EquivalentTo:
ClassExpression
…
DisjointWith:
ClassExpression
…

ClassExpression is

A class expression that is constructed using the class constructors

An Example:

Class: :VegetarianPizza

SubClassOf:
owl:Thing

EquivalentTo:

Pizza **AND**
NOT (:hasTopping **some** :FishTopping)
NOT (:hasTopping **some** :MeatTopping)

DisjointWith:
:nonVegetarianPizza

OWL Manchester Syntax – Property

Syntax:

DatatypeProperty/
ObjectProperty: PropertyID

Domain:
Class Expression
…
Range:
Class Expression
….
InverseOf:
Class Expression

Characteristics:
…..

An Example:

ObjectProperty: is_taught_by

Domain:
:Course

Range:
:Lecturer

InverseOf:
:teaches

Characteristics:
Functional

OWL Manchester Syntax – Instance

Syntax:

Individual: :IndividualID

Types:
Class Expression
….

Facts:
Property Assertation
…..

An Example:

Individual: :S1

Types:
:Student

Facts:
:enrol_in :CO1001,
:enrol_in :CO1003,
:enrol_in :CO1012,
:age "20"^^xsd:int,
:name "Mark"^^xsd:string,
:student_id "S0001"^^xsd:string