UNSW
SYDNEY

Technical Aspects of Semantic Modelling

Sponsored by:

CAPSICUM
Business Architects

RDF

- RDF is lowest level of expressivity in Semantic Web
- RDF expresses information about resources in a machine readable format
- We have seen that RDF represents information in the form of statements:
- <Subject> <Predicate> <Object>
- Questions
  - How to uniquely represent nodes ? (subject/object)
  - How to uniquely represent connections ? (predicates)

NAMING OF RESOURCES (NODES)

When is a node same node ?

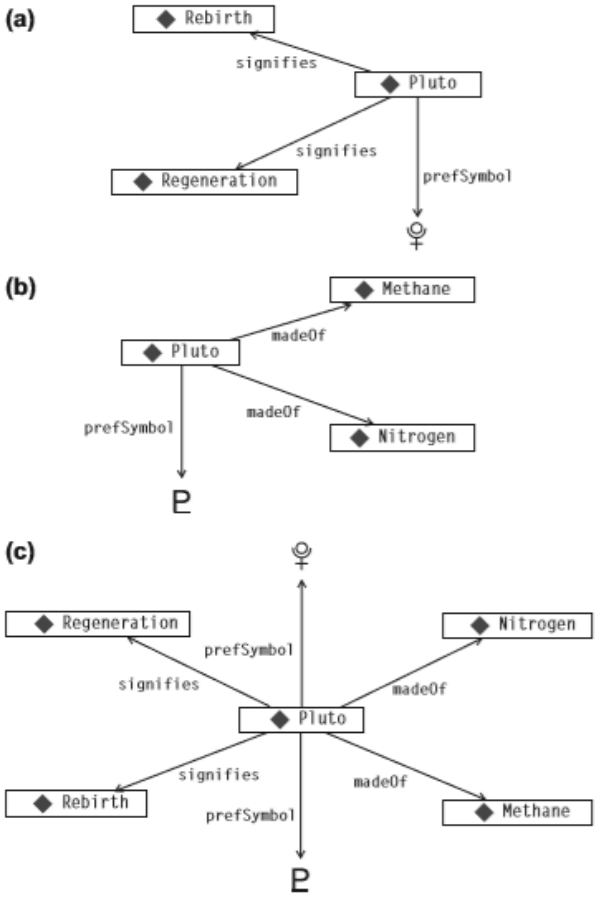


FIGURE 2.3

Layers of modeled information about Pluto.

Naming of nodes

- Each node has a unique identity
- Can use URI
  - http://www.WorkingOntologist.org/Examples/Chapter3/Shakespeare#Shakespeare
  - “A Uniform Resource Identifier (URI) is a compact sequence of characters that identifies an abstract or physical resource.” (RFC 3986)*
- Can use a qname with 2 parts
  - Namespace: lit
  - Identifier: Shakespeare
  - Qname written as lit:Shakespeare
- There must be a namespace declaration somewhere that associates “lit” to a URI e.g. http://www.WorkingOntologist.org/Examples/Chapter3/Shakespeare#

SERIALISATION

Storing semantic data

- Semantic triples need serialisation
  - Process of converting models into files
  - Facilitates storage and exchange
- Three notations available
  - N-triples
  - XML
  - Turtle

N-Triples

- N-Triples notation:

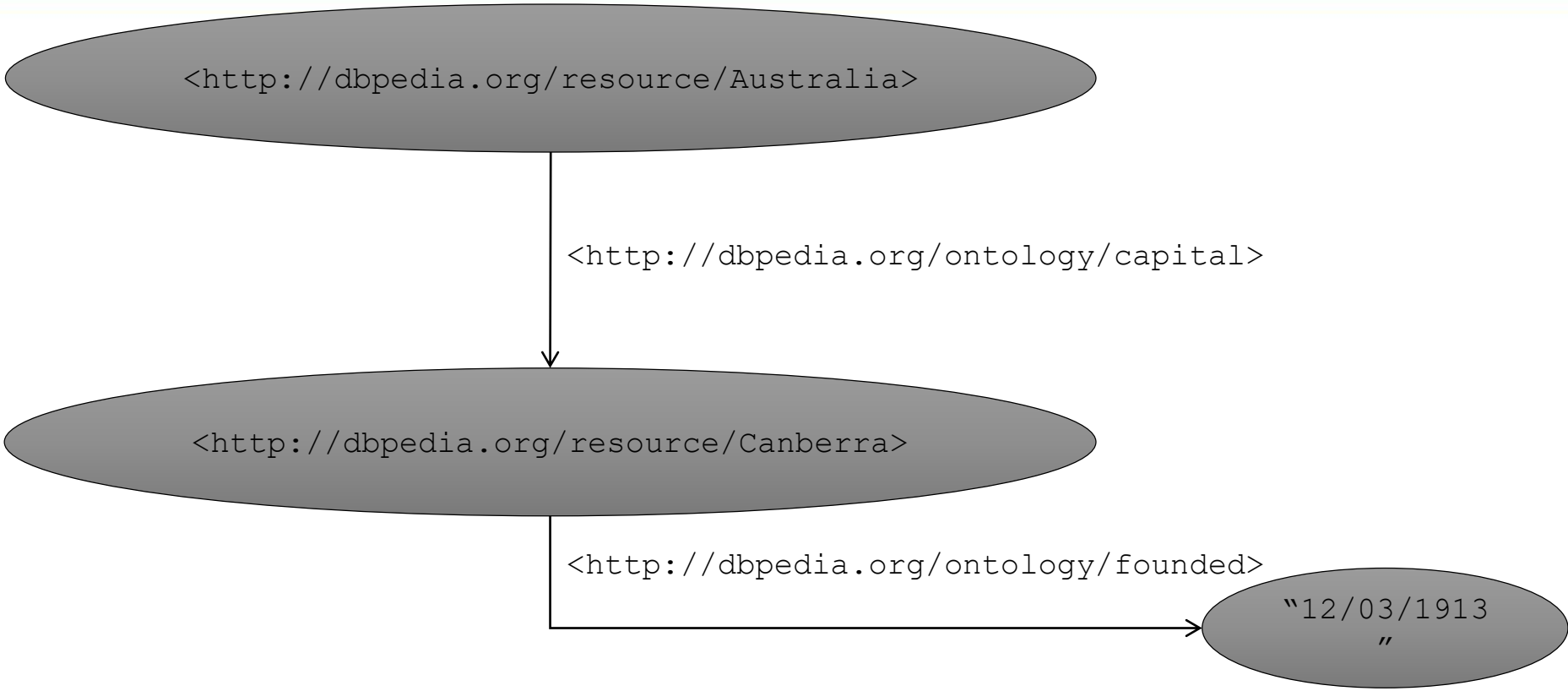
  Example – Canberra is the capital of Australia

```
<http://dbpedia.org/resource/Australia>
<http://dbpedia.org/ontology/capital>
<http://dbpedia.org/resource/Canberra> .
```

  Example – Canberra was founded on 12th March 1913.

```
<http://dbpedia.org/resource/Canberra>
<http://dbpedia.org/ontology/founded> “12/03/1913” .
```

RDF triples graph representation



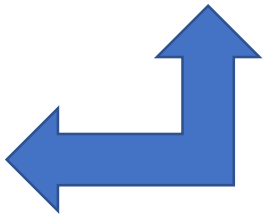

```
<http://dbpedia.org/resource/Australia>
<http://dbpedia.org/ontology/capital>
<http://dbpedia.org/resource/Canberra> .

<http://dbpedia.org/resource/Canberra>
<http://dbpedia.org/ontology/founded> "12/03/1913" .
```

XML

- RDF triples can be represented using XML notation
- Useful for embedding with web pages
- Example:

mfg:Product1	mfg:Product_ID	1
mfg:Product1	mfg:Product_ModelNo	ZX-3
mfg:Product1	mfg:Product_Division	Manufacturing supp
mfg:Product1	mfg:Product_Product_Line	Paper machine
mfg:Product1	mfg:Product_Manufacture_Location	Sacramento
mfg:Product1	mfg:Product_SKU	FB3524
mfg:Product1	mfg:Product_Available	23

```
<rdf:RDF
xmlns:mfg="http://www.WorkingOntologist.com/Examples/Chapter3/Manufacturing#"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntaxns#">
<mfg:Product
rdf:about="http://www.WorkingOntologist.com/Examples/Chapter3/Manufacturing#Product1">
        <mfg:Available>23</mfg:Available>
        <mfg:Division>Manufacturing support</mfg:Division>
        <mfg:ProductLine>Paper machine</mfg:ProductLine>
        <mfg:SKU>FB3524</mfg:SKU>
        <mfg:ModelNo>ZX-3</mfg:ModelNo>
        <mfg:ManufactureLocation>Sacramento</mfg:ManufactureLocation>
</mfg:Product>
</rdf:RDF>
```



Turtle notation

```
@prefix dbo: <http://dbpedia.org/ontology/visited> .
@base <http://dbpedia.org/resource/> .

<Australia> dbo:capital <Canberra> .
<Canberra> dbo:founded "12/03/1913"^^xsd:date .
```

OR

```
@prefix dbo: <http://dbpedia.org/ontology/visited> .
@prefix dbr: <http://dbpedia.org/resource/> .

dbr:Australia dbo:capital dbr:Canberra .
dbr:Canberra dbo:founded "12/03/1913"^^xsd:date .
```

Turtle notation

```
@prefix dbo:
<http://dbpedia.org/ontology/visited> .
@base <http://dbpedia.org/resource/> .

<Australia> dbo:capital <Canberra> ;
            dbo:currency <Australian_dollar> .
```

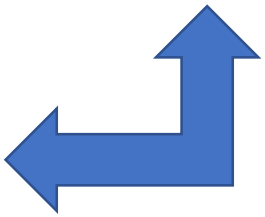
Semicolon is used to indicate same subject for the subsequent triples

Comma is used to indicate same subject and property for the subsequent triples

```
@prefix dbo:
<http://dbpedia.org/ontology/visited> .
@base <http://dbpedia.org/resource/> .

<Australia> dbo:demonym "Australian"@en ,
                        "Aussie"@en .
```

Another Turtle example

mfg:Product1	mfg:Product_ID	1
mfg:Product1	mfg:Product_ModelNo	ZX-3
mfg:Product1	mfg:Product_Division	Manufacturing support
mfg:Product1	mfg:Product_Product_Line	Paper machine
mfg:Product1	mfg:Product_Manufacture_Location	Sacramento
mfg:Product1	mfg:Product_SKU	FB3524
mfg:Product1	mfg:Product_Available	23

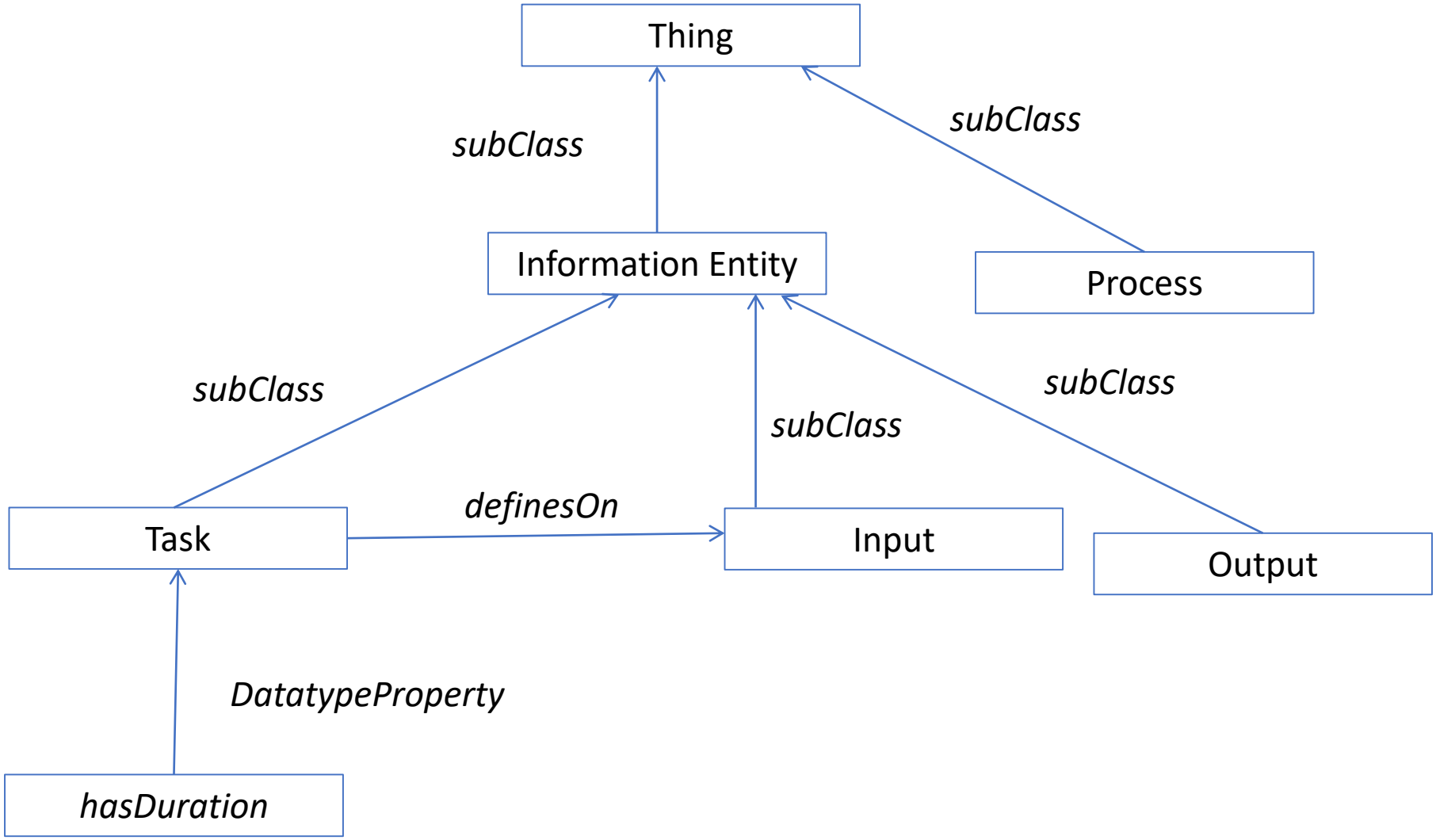


```
@prefix mfg:<http://www.WorkingOntologist.com/Examples/Chapter3/Manufacturing#>
@prefix rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns#

mfg:Product1 rdf:type mfg:Product;
        mfg:Product_Division “Manufacturing support”;
        mfg:Product_ID “1”;
        mfg:Product_Manufacture_Location “Sacramento”;
        mfg:Product_ModelNo “ZX-3”;
        mfg:Product_Product_Line “Paper Machine”;
        mfg:Product_SKU “FB3524”;
        mfg:Product_Available “23” .
```

Turtle example

```
# Classes
comp9322:InformationEntity rdf:type owl:Class .
comp9322:Input rdf:type owl:Class ;
        rdfs:subClassOf comp9322:InformationEntity .
comp9322:Output rdf:type owl:Class ;
        rdfs:subClassOf comp9322:InformationEntity .
comp9322:Process rdf:type owl:Class .
comp9322:Task rdf:type owl:Class ;
        rdfs:subClassOf comp9322:InformationEntity .
# Object Properties
comp9322:definesOn rdf:type owl:ObjectProperty ;
        rdfs:domain comp9322:Task ;
        rdfs:range comp9322:Input .
# Data properties
comp9322:hasDuration rdf:type owl:DatatypeProperty ;
        rdfs:domain comp9322:Task .
```

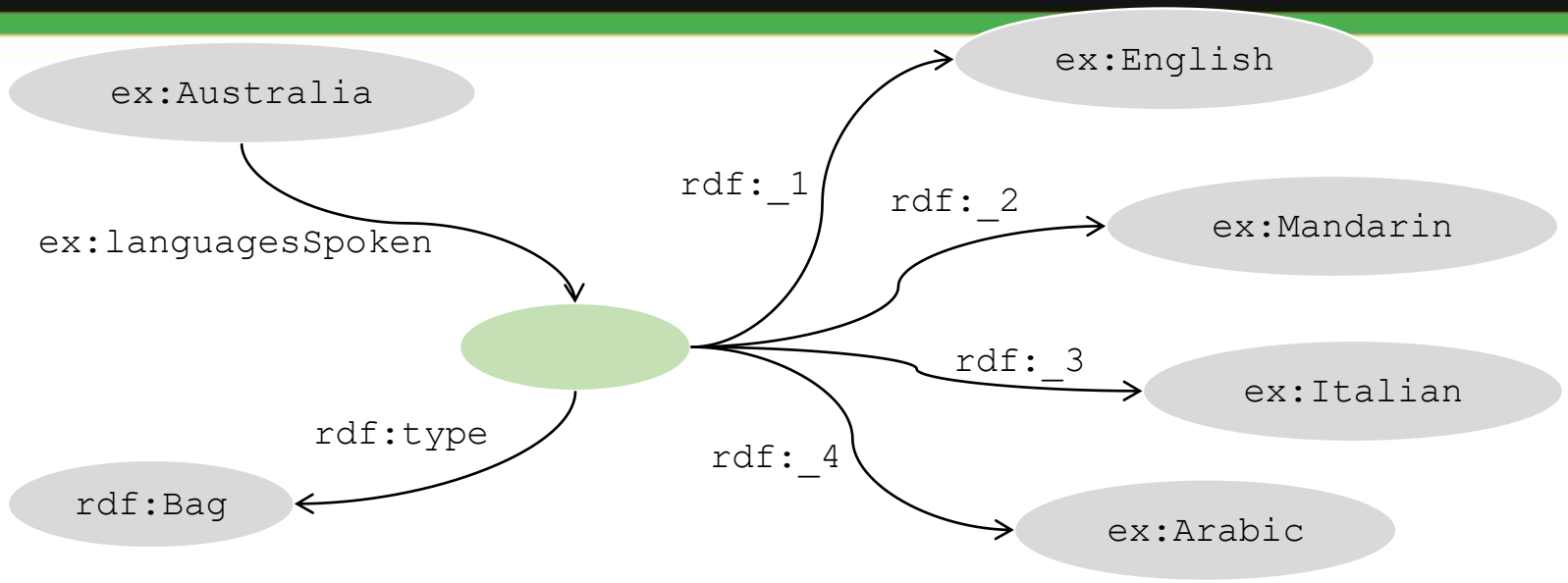
DRAW GRAPH THAT CORRESPOND TO THIS ONTOLOGY


Thing
subClass
subClass
Information Entity
Process
subClass
subClass
subClass
definesOn
Task
Input
Output
DatatypeProperty
hasDuration

Groupings

- There are two kinds of groupings:
  - Container
  - Collection
- Container
  - Is an open list (that is, new entries can be added)
  - In a container, the value of a property is a group of things (resources or literals)
  - Example: list of languages spoken in Australia
  - Types: Bag (unordered set), Seq (ordered set), Alt (alternatives)

Container

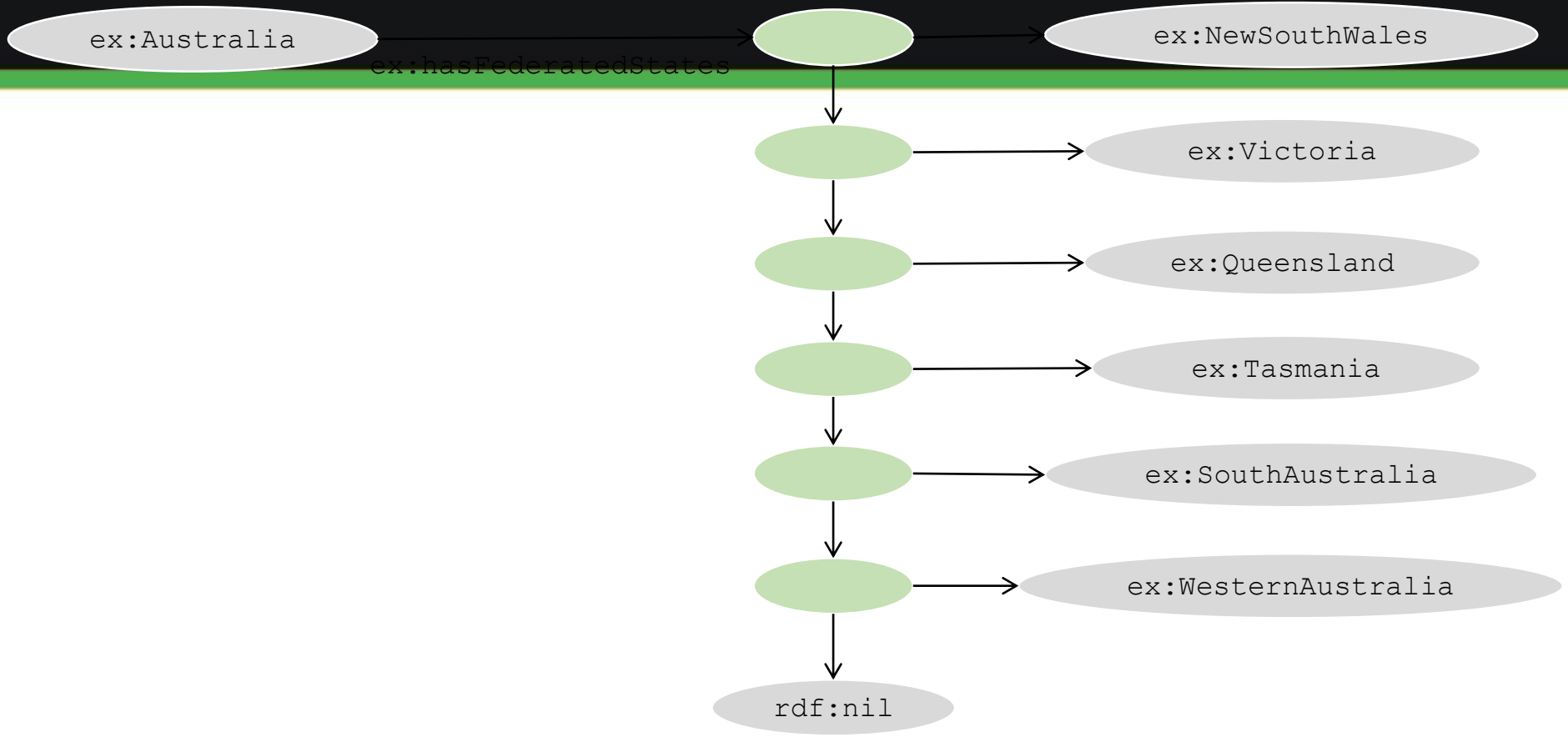



```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix ex: <http://example.org/test#> .

ex:Australia ex:languagesSpoken [
     a rdf:Bag ;
     rdf:_1 ex:English ;
     rdf:_2 ex:Mandarin ;
     rdf:_3 ex:Italian ;
     rdf:_4 ex:Arabic
] .
```

Collection

- Collection
  - Is a closed list (no new entries can be added)
  - Collection is thus a group of specified members only
  - Example: list of Australian states

Example of Collection




```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix ex: <http://example.org/test#> .

ex:Australia ex:hasFederatedStates (
     ex:NewSouthWales ex:Victoria ex:Queensland ex:Tasmania
     ex:SouthAustralia ex:WesternAustralia
) .
```

Semantic Web Architecture

- Components of a Semantic Web application
  - RDF Parser/Serialiser: reads or creates RDF files
  - RDF Store: stores RDF information
  - RDF Engine: queries information in RDF store
  - Converters/Scrappers: loads information into RDF store from Web pages and other data sources
  - Application: provides additional functions over RDF store

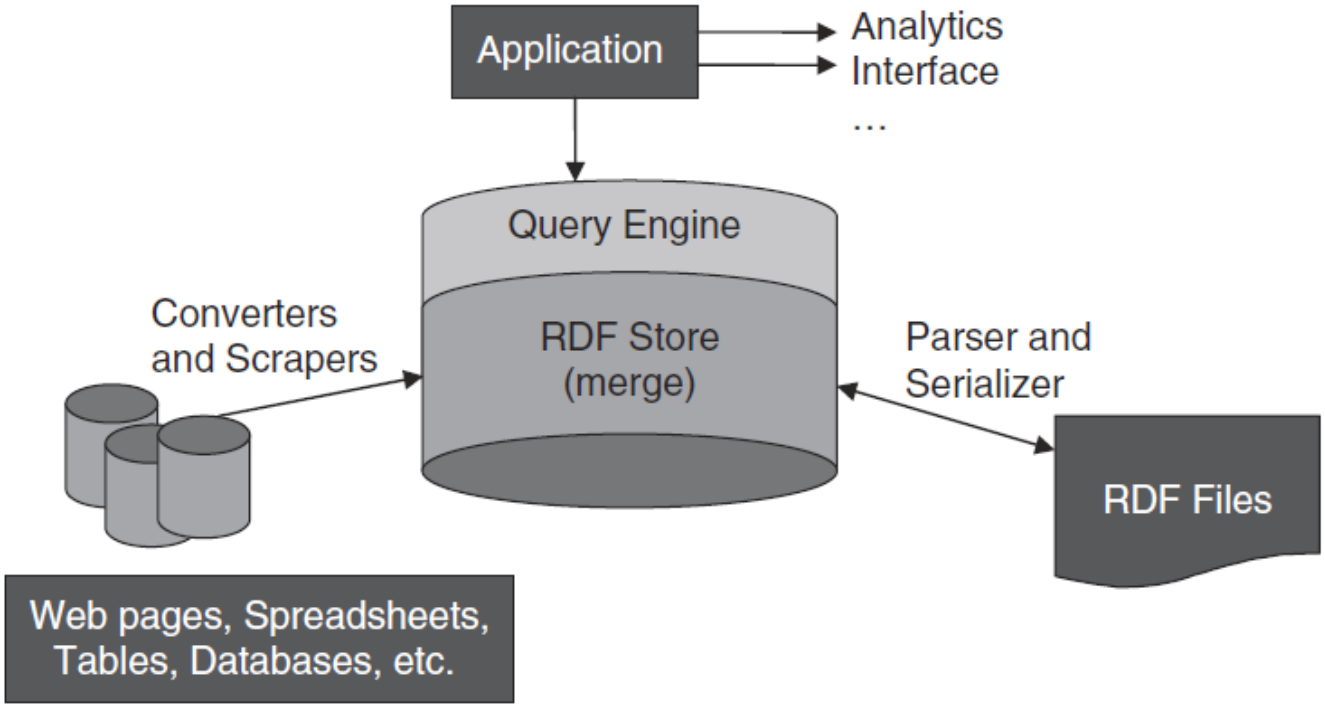


FIGURE 4.2

Application architecture for an RDF application.

References

- Dean Allemang and James Hendler, Semantic Web for the Working Ontologist, 2nd Edition, Morgan Kaufmann, 2011.