

# Use Case 1.1 - Triples Examples

[Planning](#) | [Description](#) | [Requirements](#) | [UI Mockup](#) | **[Triples Examples](#)** | [Queries Examples](#)

## Table of Contents

- [Ontologies](#)
- [Virtual Collections](#)
  - [Virtual Collection's owner](#)
  - [Virtual Collection's metadata](#)
  - [Virtual Collection's list of items](#)
    - [Ordered Collection Examples](#)
      - [Example 1-co: A Virtual Collection as an ordered collection of items using Collection Ontology's List with one item](#)
      - [Example 1-ore: A Virtual Collection as an ordered collection of items using ORE Ontology's Aggregation with one item](#)
      - [Example 2-co: A Virtual Collection as an ordered collection of items using Collection Ontology's List with multiple ordered items](#)
      - [Example 2-ore: A Virtual Collection as an ordered collection of items using ORE Ontology's Aggregation with multiple ordered items](#)
    - [Unordered Collection Examples](#)
      - [Example 3-co: A Virtual Collection as an unordered collection of items using Collection Ontology's Bag with one item](#)
      - [Example 3-ore: A Virtual Collection as an unordered collection of items using ORE Ontology's Aggregation with one item](#)
      - [Example 4-co: A Virtual Collection as an unordered collection of items using Collection Ontology's Bag with multiple unordered items](#)
      - [Example 4-ore: A Virtual Collection as an unordered collection of items using ORE Ontology's Aggregation with multiple unordered items](#)
- [Comments](#)
  - [Comments as Free Form Text Annotation](#)
- [URI Generation](#)
  - [BaseURI](#)
  - [Identifier](#)
- [Access Thoughts](#)
  - [Private vs. Public](#)
  - [Semi-Private with Shared Access](#)
  - [Collaboration](#)

## Ontologies

The following is a list of all ontologies used by the Triple Examples.

| Name              | Prefix  | URL                                                                                                   | Details                                          | Comments                                                                                                                                   |
|-------------------|---------|-------------------------------------------------------------------------------------------------------|--------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|
| RDF               | rdf     | <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#">http://www.w3.org/1999/02/22-rdf-syntax-ns#</a> | <a href="#">specificati<br/>on</a>               |                                                                                                                                            |
| RDF Schema        | rdfs    | <a href="http://www.w3.org/2000/01/rdf-schema#">http://www.w3.org/2000/01/rdf-schema#</a>             | <a href="#">specificati<br/>on</a>               |                                                                                                                                            |
| Dublin Core       | dc      | <a href="http://purl.org/dc/elements/1.1/">http://purl.org/dc/elements/1.1/</a>                       | <a href="#">specificati<br/>on</a>               |                                                                                                                                            |
| Dublin Core Terms | dcterms | <a href="http://purl.org/dc/terms">http://purl.org/dc/terms</a>                                       | <a href="#">specificati<br/>on</a>               | Required by OA                                                                                                                             |
| Dublin Core Types | dctype  | <a href="http://purl.org/dc/dcmitype">http://purl.org/dc/dcmitype</a>                                 | <a href="#">specificati<br/>on</a>               | Required by OA                                                                                                                             |
| Content           | cnt     | <a href="http://www.w3.org/2011/content#">http://www.w3.org/2011/content#</a>                         | <a href="#">specificati<br/>on</a>               | Required by OA                                                                                                                             |
| ORE               | ore     | <a href="http://www.openarchives.org/ore/terms/">http://www.openarchives.org/ore/terms/</a>           | <a href="#">specificati<br/>on</a>               | Recommended as an alternative to the Collections ontology by Rob. Represents both ordered and unordered items using the Aggregation class. |
| IANA              | iana    | <a href="http://www.iana.org/assignments/relation/">http://www.iana.org/assignments/relation/</a>     | <a href="#">specificati<br/>on</a>               | ORE uses this ontology for first, last, next, and prev predicates.                                                                         |
| Collections       | co      | <a href="http://purl.org/co">http://purl.org/co</a>                                                   | <a href="#">specificati<br/>on,<br/>examples</a> | Can represent unordered items using the Bag class and ordered items using the List class.                                                  |

|                    |      |                                                                   |                                |                 |
|--------------------|------|-------------------------------------------------------------------|--------------------------------|-----------------|
| Open Annotation    | oa   | <a href="http://www.w3.org/ns/oa">http://www.w3.org/ns/oa</a>     | <a href="#">specificati on</a> |                 |
| Friend of a Friend | foaf | <a href="http://xmlns.com/foaf/0.1">http://xmlns.com/foaf/0.1</a> | <a href="#">specificati on</a> | Also used by OA |

## Virtual Collections

### Virtual Collection's owner

Note: The owner is represented as dc:creator in the [Virtual Collection's metadata](#).

#### n-triples

```
<http://vivo.cornell.edu/individual/individual24416> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
<http://xmlns.com/foaf/0.1/Person> .
<http://vivo.cornell.edu/individual/individual24416> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
<http://xmlns.com/foaf/0.1/Agent> .
```

#### Turtle

```
@prefix foaf:      <http://xmlns.com/foaf/0.1> .

<http://vivo.cornell.edu/individual/individual24416> a foaf:Person, foaf:Agent .
```

#### NOTE:

- ORE uses foaf:Agent as the dc:creator.

#### QUESTIONS:

- What URI do we use for a user?
  - ANSWER: Each university will determine what to use as the URI for users.
    - Cornell: Will use VIVO URI's for faculty and staff. Doesn't work for students.
      - Ex. VIVO URI - <http://vivo.cornell.edu/individual/rjv23> – rjv23 is a netid
      - Ex. VIVO URI - <http://vivo.cornell.edu/individual/individual24416> – 24416 is not a netid
      - Ex. VIVO URI - <http://vivo.cornell.edu/individual/EleanorRayle> - EleanorRayle is not a netid
      - For the purposes of our experiment, if the user doesn't exist in VIVO, then they can't create a Virtual Collection.
    - Harvard: ?
    - Stanford: ?
- Do we want to keep any other foaf properties beyond setting the type to foaf person?
  - ANSWER: None extra needed in our triplestore. We can revisit this if it becomes clear by the implementation that we need to keep additional information.

### Virtual Collection's metadata

#### n-triples

```
<http://localhost:3000/individual/vc155> <http://purl.org/dc/terms/title> "My Virtual Collection" .
<http://localhost:3000/individual/vc155> <http://purl.org/dc/terms/description> "These are resources I am
gathering together for personal use." .
<http://localhost:3000/individual/vc155> <http://purl.org/dc/elements/1.1/creator> <http://vivo.cornell.edu
/individual/individual24416> .
```

## Turtle

```
@prefix dc:      <http://purl.org/dc/elements/1.1/> .
@prefix dcterms: <http://purl.org/dc/terms> .

<http://localhost:3000/individual/vc155>
  dcterms:title      "My Virtual Collection" ;
  dcterms:description "These are resources I am gathering together for personal use." ;
  dc:creator          <http://vivo.cornell.edu/individual/individual24416> .
```

## QUESTIONS:

- Should dc:creator be used for ownership role?
  - ANSWER: NO - per Paolo
  - REASON: Muddy concept. Did the user create the resource or add the resource to the collection?
  - ALTERNATIVE: [Under exploration](#). Paolo recommends looking at...
    - [PROV ontology](#) - provenance
    - [PAV ontology](#) - provenance, authoring, and versioning
- What properties should be used to express the virtual collection metadata? Possibilities and answer as Selected Property...

| Concept     | Potential Properties                              | Selected Property   | Comments                            |
|-------------|---------------------------------------------------|---------------------|-------------------------------------|
| Title       | rdfs:label, dc:title, dcterms:title               | dcterms:title       | Based on ORE ontology specification |
| Description | rdfs:comment, dc:description, dcterms:description | dcterms:description | Based on ORE ontology specification |
| Owner       | dc:creator, dcterms:creator                       | dc:creator          | Based on ORE ontology specification |

## Virtual Collection's list of items

Two ontologies have been identified as potential candidates for representing a list of items. The pros and cons are listed below for each ontology, followed by examples using each ontology.

|   | Collections ontology                                                                                                                                                                                                                                                                                                                                                                        |   | ORE ontology                                                                                                              |
|---|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|---------------------------------------------------------------------------------------------------------------------------|
| – | unordered and ordered collections are represented by two separate class constructs, Bag and List respectively                                                                                                                                                                                                                                                                               | + | unordered and ordered collections are represented by the same class construct, Aggregation                                |
| + | item property names are easy to understand (e.g., index, itemContent, nextItem)                                                                                                                                                                                                                                                                                                             | – | item property names are cryptic (e.g., proxyFor, proxyIn, next)                                                           |
| – | order is determined by an item property pointing to the next item, i.e., nextItem.                                                                                                                                                                                                                                                                                                          | – | order is determined by an item property pointing to the next item, i.e., next                                             |
| – | URI for the item is not associated with the collection class (e.g., List, Bag).<br><br>URI can be associated directly to the collection class through <code>co:element</code> . <code>co:element</code> is defined through<br><br>a chain: <code>has</code> sub-property chains [ <code>has item o has item content</code> ]. That means that the inference layer can resolve both (Paolo). | + | URI for the item is associated with the collection class (i.e., Aggregation) via of the <code>aggregates</code> property. |
| + | URI for the item is associated with the item class (i.e., Item) via the <code>itemContent</code> property.                                                                                                                                                                                                                                                                                  | + | URI for the item is associated with the item class (i.e., Proxy) via the <code>proxyFor</code> property.                  |

## Ordered Collection Examples

Example 1-co: A Virtual Collection as an ordered collection of items using Collection Ontology's List with one item

### n-triples

```
<http://localhost:3000/individual/vc155> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/co#List> .
<http://localhost:3000/individual/vc155> <http://purl.org/co#size> "1"^^xsd:nonNegativeInteger .
<http://localhost:3000/individual/vc155> <http://purl.org/co#firstItem> <http://localhost:3000/individual/vci162> .
<http://localhost:3000/individual/vc155> <http://purl.org/co#item> <http://localhost:3000/individual/vci162> .
<http://localhost:3000/individual/vc155> <http://purl.org/co#lastItem> <http://localhost:3000/individual/vci162> .

<http://localhost:3000/individual/vci162> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/co#ListItem> .
<http://localhost:3000/individual/vci162> <http://purl.org/co#index> "1"^^xsd:positiveInteger .
<http://localhost:3000/individual/vci162> <http://purl.org/co#itemContent> <http://da-rdf.library.cornell.edu/individual/b3652730> .
```

### Turtle using Collection ontology's List class

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

<http://localhost:3000/individual/vc155> a co:List ;
  co:size      "1"^^xsd:nonNegativeInteger ;
  co:firstItem <http://localhost:3000/individual/vci162> ;
  co:item      <http://localhost:3000/individual/vci162> ;
  co:lastItem  <http://localhost:3000/individual/vci162> .

<http://localhost:3000/individual/vci162> a co:ListItem
  co:index      "1"^^xsd:positiveInteger ;
  co:itemContent <http://da-rdf.library.cornell.edu/individual/b3652730> .
```

Example 1-ore: A Virtual Collection as an ordered collection of items using ORE Ontology's Aggregation with one item

### n-triples

```
<http://localhost:3000/individual/vc155> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.openarchives.org/ore/terms/Aggregation> .
<http://localhost:3000/individual/vc155> <http://www.openarchives.org/ore/terms/aggregates> <http://da-rdf.library.cornell.edu/individual/b3652730> .
```

### Turtle using ORE ontology's Aggregation class

```
@prefix ore:      <http://www.openarchives.org/ore/terms/> .

<http://localhost:3000/individual/vc155> a ore:Aggregation ;
  ore:aggregates <http://da-rdf.library.cornell.edu/individual/b3652730> .
```

Example 2-co: A Virtual Collection as an ordered collection of items using Collection Ontology's List with multiple ordered items

## n-triples

```
<http://localhost:3000/individual/vci155> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/co#List> .
<http://localhost:3000/individual/vci155> <http://purl.org/co#size> "4"^^xsd:nonNegativeInteger .
<http://localhost:3000/individual/vci155> <http://purl.org/co#firstItem> <http://localhost:3000/individual/vci162> .
<http://localhost:3000/individual/vci155> <http://purl.org/co#item> <http://localhost:3000/individual/vci162> .
<http://localhost:3000/individual/vci155> <http://purl.org/co#item> <http://localhost:3000/individual/vci163> .
<http://localhost:3000/individual/vci155> <http://purl.org/co#item> <http://localhost:3000/individual/vci164> .
<http://localhost:3000/individual/vci155> <http://purl.org/co#item> <http://localhost:3000/individual/vci165> .
<http://localhost:3000/individual/vci155> <http://purl.org/co#lastItem> <http://localhost:3000/individual/vci165> .

<http://localhost:3000/individual/vci162> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/co#ListItem> .
<http://localhost:3000/individual/vci162> <http://purl.org/co#index> "1"^^xsd:positiveInteger .
<http://localhost:3000/individual/vci162> <http://purl.org/co#itemContent> <http://da-rdf.library.cornell.edu/individual/b3652730> .
<http://localhost:3000/individual/vci162> <http://purl.org/co#nextItem> <http://localhost:3000/individual/vci163> .

<http://localhost:3000/individual/vci163> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/co#ListItem> .
<http://localhost:3000/individual/vci163> <http://purl.org/co#index> "2"^^xsd:positiveInteger .
<http://localhost:3000/individual/vci163> <http://purl.org/co#itemContent> <http://da-rdf.library.cornell.edu/individual/b3652234> .
<http://localhost:3000/individual/vci163> <http://purl.org/co#nextItem> <http://localhost:3000/individual/vci164> .

<http://localhost:3000/individual/vci164> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/co#ListItem> .
<http://localhost:3000/individual/vci164> <http://purl.org/co#index> "3"^^xsd:positiveInteger .
<http://localhost:3000/individual/vci164> <http://purl.org/co#itemContent> <http://da-rdf.library.cornell.edu/individual/b3652543> .
<http://localhost:3000/individual/vci164> <http://purl.org/co#nextItem> <http://localhost:3000/individual/vci165> .

<http://localhost:3000/individual/vci165> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/co#ListItem> .
<http://localhost:3000/individual/vci165> <http://purl.org/co#index> "4"^^xsd:positiveInteger .
<http://localhost:3000/individual/vci165> <http://purl.org/co#itemContent> <http://da-rdf.library.cornell.edu/individual/b3652884> .
```

### Turtle using Collection ontology's List class

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

<http://localhost:3000/individual/vci155> a co:List ;
  co:size      "4"^^xsd:nonNegativeInteger ;
  co:firstItem <http://localhost:3000/individual/vci162> ;
  co:item      <http://localhost:3000/individual/vci162> ;
  co:item      <http://localhost:3000/individual/vci163> ;
  co:item      <http://localhost:3000/individual/vci164> ;
  co:item      <http://localhost:3000/individual/vci165> ;
  co:lastItem  <http://localhost:3000/individual/vci165> .

<http://localhost:3000/individual/vci162> a co:ListItem ;
  co:index     "1"^^xsd:positiveInteger ;
  co:itemContent <http://da-rdf.library.cornell.edu/individual/b3652730> ;
  co:nextItem  <http://localhost:3000/individual/vci163> .

<http://localhost:3000/individual/vci163> a co:ListItem ;
  co:index     "2"^^xsd:positiveInteger ;
  co:itemContent <http://da-rdf.library.cornell.edu/individual/b3652234> ;
  co:nextItem  <http://localhost:3000/individual/vci164> .

<http://localhost:3000/individual/vci164> a co:ListItem ;
  co:index     "3"^^xsd:positiveInteger ;
  co:itemContent <http://da-rdf.library.cornell.edu/individual/b3652543> ;
  co:nextItem  <http://localhost:3000/individual/vci165> .

<http://localhost:3000/individual/vci165> a co:ListItem ;
  co:index     "4"^^xsd:positiveInteger ;
  co:itemContent <http://da-rdf.library.cornell.edu/individual/b3652884> .
```

Example 2-ore: A Virtual Collection as an ordered collection of items using ORE Ontology's Aggregation with multiple ordered items

## n-triples

```
<http://localhost:3000/individual/vc155> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.openarchives.org/ore/terms/Aggregation> .
<http://localhost:3000/individual/vc155> <http://www.openarchives.org/ore/terms/aggregates> <http://da-rdf.library.cornell.edu/individual/b3652730> .
<http://localhost:3000/individual/vc155> <http://www.openarchives.org/ore/terms/aggregates> <http://da-rdf.library.cornell.edu/individual/b3652234> .
<http://localhost:3000/individual/vc155> <http://www.openarchives.org/ore/terms/aggregates> <http://da-rdf.library.cornell.edu/individual/b3652543> .

<http://localhost:3000/individual/vci162> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.openarchives.org/ore/terms/Proxy> .
<http://localhost:3000/individual/vci162> <http://www.openarchives.org/ore/terms/proxyFor> <http://da-rdf.library.cornell.edu/individual/b3652730> .
<http://localhost:3000/individual/vci162> <http://www.openarchives.org/ore/terms/proxyIn> <http://localhost:3000/individual/vc155> .
<http://localhost:3000/individual/vci162> <http://www.iana.org/assignments/relation/next> <http://localhost:3000/individual/vci163> .

<http://localhost:3000/individual/vci163> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.openarchives.org/ore/terms/Proxy> .
<http://localhost:3000/individual/vci163> <http://www.openarchives.org/ore/terms/proxyFor> <http://da-rdf.library.cornell.edu/individual/b3652234> .
<http://localhost:3000/individual/vci163> <http://www.openarchives.org/ore/terms/proxyIn> <http://localhost:3000/individual/vc155> .
<http://localhost:3000/individual/vci163> <http://www.iana.org/assignments/relation/next> <http://localhost:3000/individual/vci164> .

<http://localhost:3000/individual/vci164> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.openarchives.org/ore/terms/Proxy> .
<http://localhost:3000/individual/vci164> <http://www.openarchives.org/ore/terms/proxyFor> <http://da-rdf.library.cornell.edu/individual/b3652543> .
<http://localhost:3000/individual/vci164> <http://www.openarchives.org/ore/terms/proxyIn> <http://localhost:3000/individual/vc155> .
```

## Turtle using ORE ontology's Aggregation class

```
@prefix ore:    <http://www.openarchives.org/ore/terms/> .
@prefix iana:   <http://www.iana.org/assignments/relation/> .

<http://localhost:3000/individual/vc155> a ore:Aggregation ;
  ore:aggregates <http://da-rdf.library.cornell.edu/individual/b3652730> ;
  ore:aggregates <http://da-rdf.library.cornell.edu/individual/b3652234> ;
  ore:aggregates <http://da-rdf.library.cornell.edu/individual/b3652543> .

<http://localhost:3000/individual/vci162> a ore:Proxy ;
  ore:proxyFor <http://da-rdf.library.cornell.edu/individual/b3652730> ;
  ore:proxyIn <http://localhost:3000/individual/vc155> ;
  iana:next <http://localhost:3000/individual/vci163> .

<http://localhost:3000/individual/vci163> a ore:Proxy ;
  ore:proxyFor <http://da-rdf.library.cornell.edu/individual/b3652234> ;
  ore:proxyIn <http://localhost:3000/individual/vc155> ;
  iana:next <http://localhost:3000/individual/vci164> .

<http://localhost:3000/individual/vci164> a ore:Proxy ;
  ore:proxyFor <http://da-rdf.library.cornell.edu/individual/b3652543> ;
  ore:proxyIn <http://localhost:3000/individual/vc155> .
```

Example 3-co: A Virtual Collection as an unordered collection of items using Collection Ontology's Bag with one item

#### n-triples

```
<http://localhost:3000/individual/vc255> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/co#Bag> .
<http://localhost:3000/individual/vc255> <http://purl.org/co#size> "1"^^xsd:nonNegativeInteger .
<http://localhost:3000/individual/vc255> <http://purl.org/co#item> <http://localhost:3000/individual/vci162> .

<http://localhost:3000/individual/vci262> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/co#Item> .
<http://localhost:3000/individual/vci262> <http://purl.org/co#itemContent> <http://da-rdf.library.cornell.edu/individual/b3652730> .
```

#### Turtle using Collection ontology's Bag class

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

<http://localhost:3000/individual/vc255> a co:Bag ;
  co:size      "1"^^xsd:nonNegativeInteger ;
  co:item      <http://localhost:3000/individual/vci262> .

<http://localhost:3000/individual/vci262> a co:Item ;
  co:itemContent <http://da-rdf.library.cornell.edu/individual/b3652730> .
```

Example 3-ore: A Virtual Collection as an unordered collection of items using ORE Ontology's Aggregation with one item

#### n-triples

```
<http://localhost:3000/individual/vc255> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.openarchives.org/ore/terms/Aggregation>
<http://localhost:3000/individual/vc255> <http://www.openarchives.org/ore/terms/aggregates> <http://da-rdf.library.cornell.edu/individual/b3652730>
```

#### Turtle using ORE ontology's Aggregation class

```
@prefix ore:      <http://www.openarchives.org/ore/terms/> .

<http://localhost:3000/individual/vc255> a ore:Aggregation ;
  ore:aggregates <http://da-rdf.library.cornell.edu/individual/b3652730> .
```

Example 4-co: A Virtual Collection as an unordered collection of items using Collection Ontology's Bag with multiple unordered items

## n-triples

```
<http://localhost:3000/individual/vc255> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/co#Bag> .
<http://localhost:3000/individual/vc255> <http://purl.org/co#size> "4"^^xsd:nonNegativeInteger .
<http://localhost:3000/individual/vc255> <http://purl.org/co#item> <http://localhost:3000/individual/vci262> .
<http://localhost:3000/individual/vc255> <http://purl.org/co#item> <http://localhost:3000/individual/vci263> .
<http://localhost:3000/individual/vc255> <http://purl.org/co#item> <http://localhost:3000/individual/vci264> .
<http://localhost:3000/individual/vc255> <http://purl.org/co#item> <http://localhost:3000/individual/vci265> .

<http://localhost:3000/individual/vci262> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/co#Item> .
<http://localhost:3000/individual/vci262> <http://purl.org/co#itemContent> <http://da-rdf.library.cornell.edu/individual/b3652730> .

<http://localhost:3000/individual/vci263> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/co#Item> .
<http://localhost:3000/individual/vci263> <http://purl.org/co#itemContent> <http://da-rdf.library.cornell.edu/individual/b3652234> .

<http://localhost:3000/individual/vci264> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/co#Item> .
<http://localhost:3000/individual/vci264> <http://purl.org/co#itemContent> <http://da-rdf.library.cornell.edu/individual/b3652543> .

<http://localhost:3000/individual/vci265> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://purl.org/co#Item> .
<http://localhost:3000/individual/vci265> <http://purl.org/co#itemContent> <http://da-rdf.library.cornell.edu/individual/b3652884> .
```

## Turtle using Collection ontology's Bag class

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

<http://localhost:3000/individual/vc255> a co:Bag ;
  co:size      "4"^^xsd:nonNegativeInteger ;
  co:item      <http://localhost:3000/individual/vci262> ;
  co:item      <http://localhost:3000/individual/vci263> ;
  co:item      <http://localhost:3000/individual/vci264> ;
  co:item      <http://localhost:3000/individual/vci265> .

<http://localhost:3000/individual/vci262> a co:ListItem ;
  co:itemContent <http://da-rdf.library.cornell.edu/individual/b3652730> .

<http://localhost:3000/individual/vci263> a co:ListItem ;
  co:itemContent <http://da-rdf.library.cornell.edu/individual/b3652234> .

<http://localhost:3000/individual/vci264> a co:ListItem ;
  co:itemContent <http://da-rdf.library.cornell.edu/individual/b3652543> .

<http://localhost:3000/individual/vci265> a co:ListItem ;
  co:itemContent <http://da-rdf.library.cornell.edu/individual/b3652884> .
```

(Paolo) This can be simplified to (see <https://code.google.com/p/collections-ontology/wiki/SetExample1>) :

```
<http://localhost:3000/individual/vc255> a co:Bag ;
  co:element <http://da-rdf.library.cornell.edu/individual/b3652730> ;
  co:element <http://da-rdf.library.cornell.edu/individual/b3652234> ;
  co:element <http://da-rdf.library.cornell.edu/individual/b3652543> ;
  co:element <http://da-rdf.library.cornell.edu/individual/b3652884> .
```

(Lynette) My understanding of the difference between Set and Bag is that Set cannot have repeating items, but Bag can.

- Can set have repeating items?

- Do we need to have repeating items?
  - Use Case: A professor creates a reading list which includes a work early in the semester and then repeats it later in the semester.

(Paolo) I apologize that should have been a `co:Set` to be clearer. For Bags you need to have an instance of `Item` to support for repeated elements. Once you flatten down to `co:element` the repeated items become one. The idea of the `co:element` (defined as chain) in general is to provide a shortcut to the elements. In that case we are interested in finding the existence (one of them), the repetitions are irrelevant at that level.

(Rebecca) I see on the description of the Collections Ontology that "the Collections Ontology (CO) defines unordered collections (Set and Bag) and ordered collections (or List)." It seems awkward to select an ontology in which the type of the collection has to change if ordering is added to an originally unordered collection, or vice versa. That seems to me an advantage of the ORE ontology.

(Paolo) that is a fair point but the description is not accurate, the approach has been described in the paper, here it is:

The relationships between the mathematical entities and those defined by Collections Ontology - and detailed in the following sections of the paper - can be defined as follow:

`co:Set Set`

`co:Bag Bag`

`co:Set co:Bag =`

`co:List = co:Bag Sequence`

There are precise pragmatic reasons motivating this design choice. First of all, we chose not to model the mathematical identity function in CO for a specific reason: to allow one to use CO even when modelling scenarios that describe "collections in terms of the constructive boundaries of those plural entities that form themselves a whole". Therefore, it is possible to consider two sets of people, composed exactly by the same people, as two different research groups without contradictions. .... Second, from an implementation standpoint, the data structures managing `co:Set` and `co:Bag` are very different.

In other words, if you define a Bag and you add ordering you get a List. See axioms here: <http://www.essepuntato.it/lode/owlapi/http://purl.org/co/#d4e499>

Set is all another story though. That is correct.

(Rob) The repeating items case makes ORE very strange, as it means creating two proxies for the same resource such that it can be in the chain twice.

```
<x> a ore:Aggregation ;
  ore:aggregates <a> ; // should be repeated at the end
  ore:aggregates <b> ;
  ore:aggregates <c> .
```

```
<p1> a ore:Proxy ;
  ore:proxyFor <a> ;
  ore:proxyIn <x> ;
  iana:next <p2> .
```

```
<p2> a ore:Proxy ;
  ore:proxyFor <b> ;
  ore:proxyIn <x> ;
  iana:next <p3> .
```

```
<p3> a ore:Proxy ;
  ore:proxyFor <c> ;
  ore:proxyIn <x> ;
  iana:next <p4> .
```

```
<p4> a ore:Proxy ;
  ore:proxyFor <a> ;
  ore:proxyIn <x> .
```

(Rebecca) I don't actually find this so strange. If proxies are the means to specify relationships, such as sequencing, among the aggregated resources, then if an aggregated resource has more than one relationship to other resources, as in occurring at multiple positions in a sequence, it seems natural that it would have a proxy for each.

(Paolo) I do find it strange but not 'that strange'. In CO you can infer the top part so the `co:element` will have only three objects (with no repetition). But the actual declarative structure will list all of them.

Example 4-ore: A Virtual Collection as an unordered collection of items using ORE Ontology's Aggregation with multiple unordered items

## n-triples

```
<http://localhost:3000/individual/vc255> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://www.openarchives.org/ore/terms/Aggregation> .
<http://localhost:3000/individual/vc255> <http://www.openarchives.org/ore/terms/aggregates> <http://da-rdf.library.cornell.edu/individual/b3652730> .
<http://localhost:3000/individual/vc255> <http://www.openarchives.org/ore/terms/aggregates> <http://da-rdf.library.cornell.edu/individual/b3652234> .
<http://localhost:3000/individual/vc255> <http://www.openarchives.org/ore/terms/aggregates> <http://da-rdf.library.cornell.edu/individual/b3652543> .
```

## Turtle using ORE ontology's Aggregation class

```
@prefix ore:    <http://www.openarchives.org/ore/terms/> .
@prefix iana:   <http://www.iana.org/assignments/relation/> .

<http://localhost:3000/individual/vc255> a ore:Aggregation ;
  ore:aggregates <http://da-rdf.library.cornell.edu/individual/b3652730> ;
  ore:aggregates <http://da-rdf.library.cornell.edu/individual/b3652234> ;
  ore:aggregates <http://da-rdf.library.cornell.edu/individual/b3652543> .
```

- Should Virtual Collection's type be a subclass of co:List instead of a co:List?
  - Discussion:
    - If so, what is the type? Perhaps: Subclass of co:List Ontology to be defined by ontology group.
    - Is it LD4L specific? Consensus: YES to be defined by ontology group.
    - How is LD4L defining new types? Naming convention, namespace, etc.? Final definitions by ontology group.
      - For now, use namespace=LD4L in Ruby with link <<http://ld4l.org/ontology/>> – to be defined by ontology group
      - Class name=ld4l:VirtualCollection
      - type URI=<<http://ld4l.org/ontology/VirtualCollection>>
      - instance URI=<<http://localhost:3000/individual/vc123>> –
      - what ever I want for now
      - suggest use of individual in URI meaning an instance of a class (used by VIVO too)
      - id starts with alpha because syntax something:vc123 – cannot have number directly after :
  - ANSWER:
    - NO - from Rob in Ontology group
    - Reason - When only adding a few extra properties from other existing ontologies, you do not make a new subclass to formally specify that those properties are expected in the subclass. Only create a new subclass if new properties not existing in another ontology are required.

## Comments

### Comments as Free Form Text Annotation

## n-triples

```
<http://localhost:3000/individual/vci165/comment/changeme:93> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
<http://www.w3.org/ns/oa#Annotation> .
<http://localhost:3000/individual/vci165/comment/changeme:93> <http://www.w3.org/ns/oa#hasTarget>
<http://localhost:3000/individual/vci162> .
<http://localhost:3000/individual/vci165/comment/changeme:93> <http://www.w3.org/ns/oa#hasBody>
<http://localhost:3000/individual/ab205> .
<http://localhost:3000/individual/vci165/comment/changeme:93> <http://www.w3.org/ns/oa#annotatedBy>
<http://vivo.cornell.edu/individual/individual24416> .
<http://localhost:3000/individual/vci165/comment/changeme:93> <http://www.w3.org/ns/oa#annotatedAt> "2014-07-
21T12:00:00Z"^^xsd:dateTime .
<http://localhost:3000/individual/vci165/comment/changeme:93> <http://www.w3.org/ns/oa#motivatedBy> <http://www.
w3.org/ns/oa#commenting> .

<http://localhost:3000/individual/vci165/comment/changeme:93/body> <http://www.w3.org/1999/02/22-rdf-syntax-
ns#type> <http://purl.org/dc/dcmitype/Text> .
<http://localhost:3000/individual/vci165/comment/changeme:93/body> <http://www.w3.org/1999/02/22-rdf-syntax-
ns#type> <http://www.w3.org/2011/content#ContentAsText> .
<http://localhost:3000/individual/vci165/comment/changeme:93/body> <http://www.w3.org/2011/content#chars> "This
is my favorite book." .
<http://localhost:3000/individual/vci165/comment/changeme:93/body> <http://purl.org/dc/terms/format> "text
/plain" .
```

## Turtle

```
@prefix rdf:      <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix dcterms:  <http://purl.org/dc/terms> .
@prefix dctypes:  <http://purl.org/dc/dcmitype> .
@prefix cnt:      <http://www.w3.org/2011/content#> .
@prefix oa:       <http://www.w3.org/ns/oa> .

<http://localhost:3000/individual/cm93> a oa:Annotation ;
  oa:hasTarget <http://localhost:3000/individual/vci162> ;
  oa:hasBody <http://localhost:3000/individual/ab205> ;
  oa:annotatedBy <http://vivo.cornell.edu/individual/individual24416> ;
  oa:annotatedAt "2014-07-21T12:00:00Z"^^xsd:dateTime
  oa:motivatedBy oa:commenting .

<http://localhost:3000/individual/ab205> a dctypes:Text, cnt:ContentAsText ;
  cnt:chars "This is my favorite book." ;
  dcterms:format "text/plain" .
```

## URI Generation

### BaseURI

| Environment | Base URI                          |
|-------------|-----------------------------------|
| development | http://localhost:3000/individual/ |
| testing     | TBA                               |
| production  | TBA                               |

### Identifier

\_BASE\_URI\_ + \_CONCEPT\_ABBR\_CODE\_ + \_GENERATED\_NUMBER\_

| CONCEPT                 | ABBR_CODE | Example URI                             | Comments |
|-------------------------|-----------|-----------------------------------------|----------|
| Virtual Collection      | vc        | http://localhost:3000/individual/vc155  |          |
| Virtual Collection Item | vci       | http://localhost:3000/individual/vci262 |          |
| Comment                 | cm        | http://localhost:3000/individual/cm394  |          |
| Tag                     | tg        | http://localhost:3000/individual/tg328  |          |
| Annotation Body         | ab        | http://localhost:3000/individual/ab325  |          |

#### QUESTIONS:

- Is it best practice to use the format specified for identifier?
  - The alternative could be to use the URL for the webpage with a .nt extension if a webpage representation exists AND use the above format if a webpage does not exist.
  - OR force the webpage URL to match this format whenever possible AND continue to use the .nt extension to show the triples.
  - We can punt the final decision on this question until implementation is further along.

## Access Thoughts

#### QUESTIONS:

- What method is used to protect private Virtual Collections?
  - Option 1: maintain two triple stores: 1) private, 2) public
  - Option 2: use [named graphs](#) within a single triplestore
- What triples, if any, are stored for the following concepts related to access control?
  - Roles (e.g. site-admin, library-admin, user)
    - only a small set of roles will exist
  - Groups (e.g. shared\_read\_group, shared\_write\_group)
    - potential for large number of groups to be created by users
  - Privileges
    - definitions of what users with specific roles can do
- What of these are application specific and what are needed in triples?

### Private vs. Public

- Public Virtual Collection
  - Is discoverable through public search.
  - Any user, logged in or not, can view a public virtual collection.
- Private Virtual Collection
  - Is NOT discoverable through public search.
  - User must be logged in and be the owner/creator of the virtual collection to view/edit a private virtual collection.

### Semi-Private with Shared Access

- Allow shared read-only access. Virtual Collection is not discoverable through public search.
  - Option 1: Provide a URL that the owner can share with other users. Anyone using the URL will be able to view the Virtual Collection.
  - Option 2: Owner identifies other users who are allowed to view the Virtual Collection

### Collaboration

- Allow share write access
  - Option 1: Owner identifies other users who are allowed to edit the Virtual Collection.

NOTE: This doesn't really come into play until Use Case 1.2, but I want to think about the other access issues with this in mind