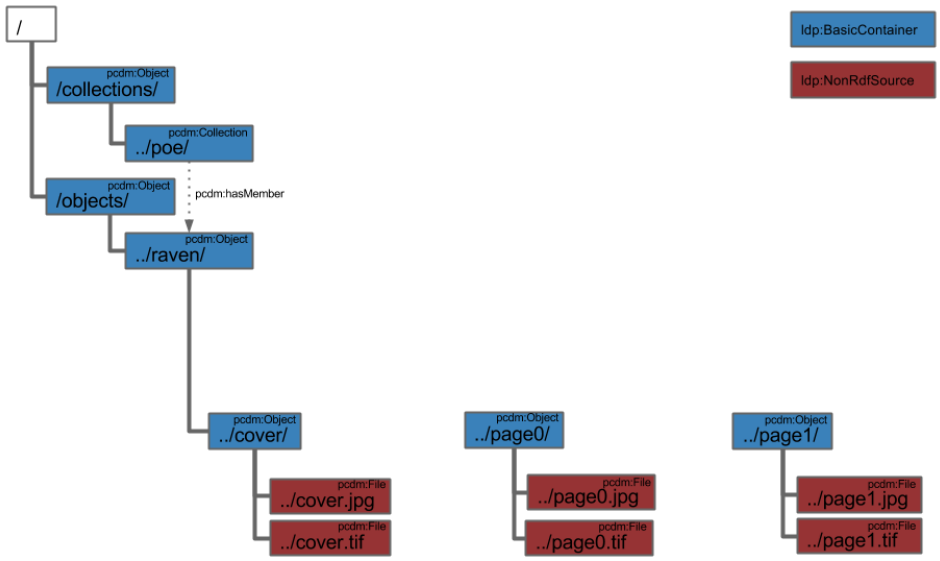
LDP-PCDM-F4 In Action - Collection

Collections In Action

2: Final State - Collection

Continuing with the previous example of modeling and creating a book with LDP, PCDM and F4, here we will detail an approach for adding that book, "raven/" to a new collection, "poe/".

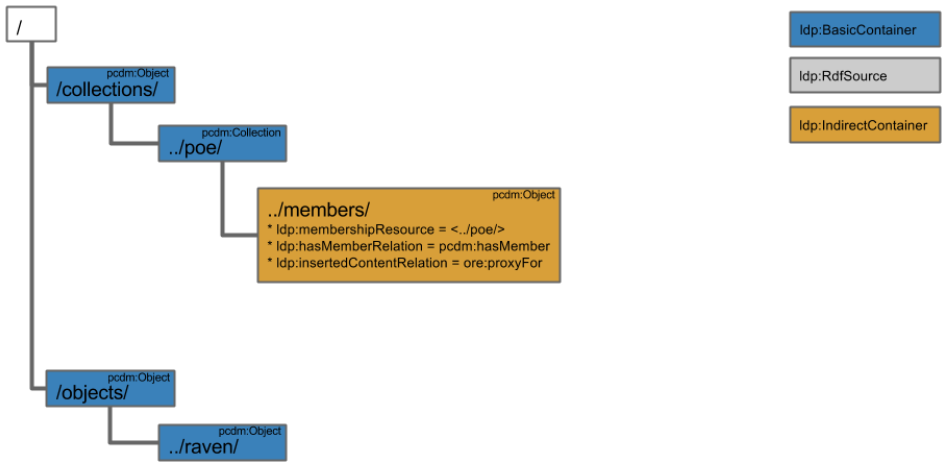


The objective in this section is to leverage LDP interaction models to not only create the appropriate pcdm:hasMember relationship between the collection "poe/" and the book "raven/", but to put the LDP structure in place for a simplified addition of new items to the "poe/" collection.

Collection - Create IndirectContainer

Collection - Create IndirectContainer

Here we will begin to walk through the mechanics of creating the structures that will facilitate creation of the collection and its single member, in this case.



First, create the top-level "collections/" pcdm:Object, which is also an ldp:BasicContainer.

```
curl -i -XPUT -H"Content-Type: text/turtle" --data-binary @pcdm-object.ttl localhost:8080/fcrepo/rest
/collections/
```

Where "pcdm-object.ttl" follows:

pcdm-object.ttl

```
@prefix pcdm: <http://pcdm.org/models#>

<> a pcdm:Object .
```

Second, create the nested "poe/" pcdm:Collection, which is also another ldp:BasicContainer.

```
curl -i -XPUT -H"Content-Type: text/turtle" --data-binary @pcdm-collection.ttl localhost:8080/fcrepo/rest
/collections/poe/
```

Where "pcdm-collection.ttl" follows:

```
@prefix pcdm: <http://pcdm.org/models#>

<> a pcdm:Collection .
```

Lastly, create an ldp:IndirectContainer, "members/" that will facilitate the establishment of relationships between "poe/" and the collection members.

```
curl -i -XPUT -H"Content-Type: text/turtle" --data-binary @ldp-indirect.ttl localhost:8080/fcrepo/rest
/collections/poe/members/
```

Where "ldp-indirect.ttl" follows:

ldp-indirect.ttl

```
@prefix ldp: <http://www.w3.org/ns/ldp#>
@prefix pcdm: <http://pcdm.org/models#>
@prefix ore: <http://www.openarchives.org/ore/terms/>

<> a ldp:IndirectContainer, pcdm:Object ;
  ldp:membershipResource </fcrepo/rest/collections/poe/> ;
  ldp:hasMemberRelation pcdm:hasMember ;
  ldp:insertedContentRelation ore:proxyFor .
```

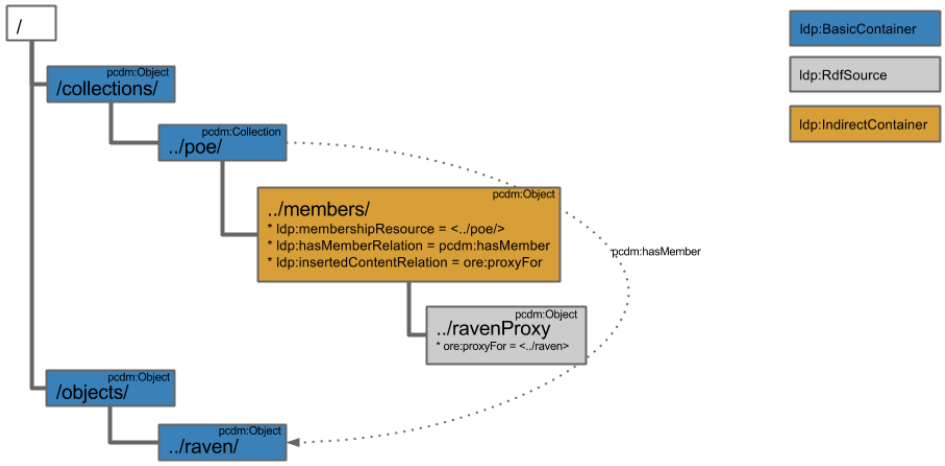
Similar to the previously described ldp:DirectContainer, an ldp:IndirectContainer is an LDP construct that also activates the creation of certain RDF triples when a new resource is added as a child of this container.
Just like with a DirectContainer, when a new resource is added inside of the "members/" IndirectContainer, a new triple on the ldp:membershipResource ("poe/") will be created with the predicate defined by the ldp:hasMemberRelation property ("pcdm:hasMember").
However, the difference from a DirectContainer is that the object of the created triple is not the newly added child, but instead the resource defined by the ldp:insertedContentRelation property (ore:proxyFor, in this case) found on the newly added child of this container.

We will see this in action next!

Collection - Create Raven Proxy

Collection - Create Raven Proxy

Create a new pcdm:Object, "ravenProxy/", that is an ldp:RdfSource within the "members/" IndirectContainer.




```
curl -i -XPUT -H"Content-Type: text/turtle" --data-binary @pcdm-raven-proxy.ttl localhost:8080/fcrepo/rest
/collections/poe/members/ravenProxy
```

Where "pcdm-object.ttl" follows:

pcdm-raven-proxy.ttl

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

<> a pcdm:Object ;
  ore:proxyFor </fcrepo/rest/objects/raven/> .
```

As mentioned in the previous step, the addition of "ravenProxy/" automatically creates the following new triple on "poe/".

```
<http://localhost:8080/fcrepo/rest/collections/poe/> pcdm:hasMember <http://localhost:8080/fcrepo/rest/objects
/raven/>
```

The ldp:IndirectContainer defines the creation of this triple as follows:

- the subject of the triple comes from the "ldp:membershipResource" defined on "members/"
- the predicate of the triple comes from the "ldp:hasMemberRelation" defined on "members/", and
- the object of the triple is the resource defined by the ldp:insertedContentRelation property (ore:proxyFor) found on the newly added child resource, "ravenProxy".

Collection - Conclusion

Using LDP in conjunction with PCDM terms, we have created a collection, "poe/", with its single member, "raven/".