

## Infrastructure - Task #3596

### Fix Merritt Repository resource map typing of triple objects

2013-02-19 01:34 - Chris Jones

Status:	New	Start date:	2013-02-19
Priority:	Normal	Due date:	
Assignee:	John Kunze	% Done:	0%
Category:	mn.Merritt	Estimated time:	0.00 hour
Target version:		Story Points:	
Milestone:	None		
Product Version:	*		

#### Description

When trying to parse resource maps generated by the CDL Merritt repository, I'm unable to get an Aggregation from a given ResourceMap. It fails with:

Parsed ark:/13030/m50000sp/1/mrt-dataone-map.rdf.  
Exception in thread "main" java.lang.ClassCastException: com.hp.hpl.jena.rdf.model.impl.LiteralImpl cannot be cast to com.hp.hpl.jena.rdf.model.Resource  
at org.dspace.foresite.jena.ResourceMapJena.getAggregation(ResourceMapJena.java:329)  
at org.dataone.tests.OREParserTest.main(OREParserTest.java:60)

After comparing the Merritt resource maps with KNB resource maps, I'm seeing a difference in the way the objects of some triples are being typed. For instance:

From KNB: [https://knb.ecoinformatics.org/knb/d1/mn/v1/object/resourceMap\\_6000141086\\_2.3.2](https://knb.ecoinformatics.org/knb/d1/mn/v1/object/resourceMap_6000141086_2.3.2):

From Merritt: <https://merritt.cdlib.org:8084/knb/d1/mn/v1/object/ark:/13030/m50000sp/1/mrt-dataone-map.rdf>

[ore:describes](http://store.cdlib.org:35121/content/1001/ark%3A%2F13030%2Fm50000sp/1//ore:describes)

The describes statement in these two maps differ in that the KNB map types the object as an rdf:resource, whereas the Merritt statement does not. When the aggregation is being built, the ResourceMapJena class iterates through these triples and creates Resource instances of the objects. In the case of Merritt, it creates a LiteralImpl instance that cannot be cast to a Resource.

When I changed the Merritt resource map document to use the explicit typing, it produced an Aggregation instance just fine.

This likely means that each of the ORE objects in the Merritt repositories need to be MN.update()'d to use explicit typing. There are other examples of this typing difference for other triples as well, such as:

doi:10.5063/AA/6000141086\_2.7.1/[dcterms:identifier](#)

vs

[dcterms:identifier](#)ark:/13030/m50000sp/1/mrt-dataone-map.rdf/[dcterms:identifier](#)

Here's the test code I was using:

```
package org.dataone.tests;
import java.io.InputStream;

import org.dataone.client.D1Client;
import org.dataone.client.MNode;
import org.dataone.service.exceptions.BaseException;
import org.dataone.service.types.v1.Identifier;
import org.dspace.foresite.Aggregation;
import org.dspace.foresite.OREException;
import org.dspace.foresite.OREParser;
import org.dspace.foresite.OREParserException;
```

```

import org.dspace.foresite.OREParserFactory;
import org.dspace.foresite.ResourceMap;

public class OREParserTest {

    public static void main(String[] args) {
        OREParser parser = OREParserFactory.getInstance("RDF/XML");
        MNode mn = D1Client.getMN("https://merritt.cdlib.org:8084/knb/d1/mn");
        String pidStr = "ark:/13030/m50000sp/1/mrt-dataone-map.rdf";
        InputStream rdfStream = null;

        Identifier pid = new Identifier();
        pid.setValue(pidStr);
        try {
            rdfStream = mn.get(pid);
            try {
                ResourceMap resourceMap = parser.parse(rdfStream);
                System.out.println("Parsed " + pidStr);
                Aggregation aggregation = resourceMap.getAggregation();
                System.out.println("Got aggregation from " + pidStr);

            } catch (OREParserException e) {
                e.printStackTrace();

            } catch (OREException e) {
                e.printStackTrace();

            }
        } catch (BaseException e) {
            e.printStackTrace();
        }
    }
}

```

## History

---

#1 - 2013-02-19 02:27 - Matthew Jones

From KNB: [https://knb.ecoinformatics.org/knb/d1/mn/v1/object/resourceMap\\_6000141086\\_2.3.2](https://knb.ecoinformatics.org/knb/d1/mn/v1/object/resourceMap_6000141086_2.3.2):

From Merritt: <https://merritt.cdlib.org:8084/knb/d1/mn/v1/object/ark:/13030/m50000sp/1/mrt-dataone-map.rdf>

[ore:describes](http://store.cdlib.org:35121/content/1001/ark%3A%2F13030%2Fm50000sp/1//ore:describes)<http://store.cdlib.org:35121/content/1001/ark%3A%2F13030%2Fm50000sp/1//ore:describes>

Looking at these examples, the Merritt [ore:describes](#) is describing the string literal, rather than the resource that the string literal points to. Its a critical distinction. The two triples are different as night and day in terms of what they say semantically. The Merritt example should really be:

assuming that the url literal points at the aggregation that this ORE describes. ORE is explicit about the semantics of ore:describes, which is defined quite completely here:

<http://www.openarchives.org/ore/1.0/datamodel.html#ReM-to-aggr>