Sprint 3 Feedback - A Woods

- Test Environment
  - Composition of Resources
  - Initialization
- Test 0 - Export/Import excluding binaries
  - Test 0 Conclusions
- Test 1 - Export/Import with binaries
  - Test 1 Conclusions
- Test 2 - Export/Import single resource
  - Test 2 Conclusions
- Test 3 - Export/Import single resource - with user-provided membership relationship, with internal binaries
  - Test 3 Conclusions
- Test 4 - Export/Import single resource - with user-provided membership relationship, with external binaries
- Test 5 - Export/Import - WebAC resources
- Test 6 - Export/Import - Direct Containers
- Test 7 - Export/Import - Indirect Containers

Test Environment

Ubuntu 16.10

Composition of Resources

1. No structural hierarchy
2. 2 collections
3. 3 books in each collection
4. 2 managed images in each book of first collection
5. 2 external content in the other collection of books
6. relationships:
   a. col <hasMember> book
   b. book <isPartOf> col
   c. book <hasMember> image
7. Some custom types and properties on resources

Initialization

```
HOST=localhost
PORT=8686
CONTEXT=f4
```

1. Start Fedora 4.7.1-RC-2 one-click at port=8686 and context=f4

   ```
   java -Dfcrepo.dynamic.jms.port=61717 -Dfcrepo.dynamic.stomp.port=61713 -jar fcrepo-webapp/target/fcrepo-webapp-4.7.1-SNAPSHOT-jetty-console.jar --port 8686 --headless --contextPath /f4
   ```

2. Start Fedora 4.7.1-RC-2 one-click at port=8080 and context=fcrepo

   ```
   java -jar fcrepo-webapp/target/fcrepo-webapp-4.7.1-SNAPSHOT-jetty-console.jar --port 8080 --headless --contextPath /fcrepo
   ```

3. Load resources per ingest.sh
   a. Example RDF of resources
      i. collection0.rdf
      ii. book0.rdf
      iii. bookA.rdf
4. Download 0.1.0 import/export jar

Test 0 - Export/Import excluding binaries

1. Export from /f4 without binaries

```
java -jar ./bin/fcrepo-import-export-0.1.0.jar -a -d data -m export -r http://localhost:8686/f4/rest/
```

   a. ✅ Audit log looks correct
   b. ✅ Correct resources were exported to data-dir
2. Import to /fcrepo

```
java -jar ./bin/fcrepo-import-export-0.1.0.jar -a -d data -m import -s http://localhost:8686/f4/rest/ -r
http://localhost:8080/fcrepo/rest/
```

   a. Output:

```
$ INFO 10:11:50.360 (Importer) Running importer...
```

   b. ⚠️ No resources loaded into /fcrepo
   c. ⚠️ Looking for user notes on how to configure logging... none found
3. Import to /fcrepo with logging

```
java -Dfcrepo.log.importexport=TRACE -jar ./bin/fcrepo-import-export-0.1.0.jar -a -d data -m import -s
http://localhost:8686/f4/rest/ -r http://localhost:8080/fcrepo/rest/
```

   a. Output:

```
$ DEBUG 10:14:46.806 (Importer) No container exists in the metadata directory data/fcrepo/rest.ttl
for the requested resource http://localhost:8080/fcrepo/rest, importing all contained resources
instead.
```

   b. ⚠️ Question, is it possible to export from :8686/f4 to :8080/fcrepo ?
      i. Resolution: Moved dir data/f4 to data/fcrepo, re-ran:

```
java -Dfcrepo.log.importexport=TRACE -jar ./bin/fcrepo-import-export-0.1.0.jar -a -d data -
m import -s http://localhost:8686/f4/rest/ -r http://localhost:8080/fcrepo/rest/
```

      ii. ⚠️ Audit log indicates collections were imported, but received 400 on books
      iii. ⚠️ Only book placeholder resources exist

Test 0 Conclusions

1. ❌ Round-tripping from one context to another not fully supported
2. ❌ Re-importing resources with inter-relationships not supported

Test 1 - Export/Import with binaries

1. Export from /f4 with binaries

```
java -jar ./bin/fcrepo-import-export-0.1.0.jar -b -a -d data -m export -r http://localhost:8686/f4/rest/
```

   a. ✅ Audit log looks correct
   b. ✅ Correct resources were exported to data-dir, including internal and external binaries
2. Import to /fcrepo
   a. Per discovery in Test 0, moved dir data/f4 to data/fcrepo, before running:

```
java -jar ./bin/fcrepo-import-export-0.1.0.jar -b -a -d data -m import -s http://localhost:8686/f4
/rest/ -r http://localhost:8080/fcrepo/rest/
```

   b. ✅ Audit log looks correct

c. ✅ Visual comparison of HTML view of repositories looks correct
d. Note: namespaces must be defined in an external CND

Test 1 Conclusions

1. ✅ Round-tripping with binaries works as expected, noting that the data-dir must be renamed if importing to a different context

Test 2 - Export/Import single resource

1. Export from /f4 - single resource

```
java -jar ./bin/fcrepo-import-export-0.1.0.jar -b -a -d data -m export -r http://localhost:8686/f4/rest
/collection0
```

   a. ✅ Audit log looks correct
   b. ✅ Correct resource was exported to data-dir
2. Import to /fcrepo
   a. Per discovery in Test 0, moved dir data/f4 to data/fcrepo, before running:

```
java -jar ./bin/fcrepo-import-export-0.1.0.jar -b -a -d data -m import -s http://localhost:8686/f4
/rest/collection0 -r http://localhost:8080/fcrepo/rest/collection0
```

   b. 👍👎 Audit log has confusing output:

```
$ 2017-01-08 10:59:39,544 import /import-export-testing/data/fcrepo/rest/collection0.ttl to
http://localhost:8080/fcrepo/rest/collection0/../collection0
```

      i. Note: /rest/collection0/../collection0
   c. ⚠️ Relationships point to resources with their original (/f4) URIs

Test 2 Conclusions

1. 👍👎 Audit logging is correct but needs polish
2. ❌ Relationships to non-imported resources are not supported

Test 3 - Export/Import single resource - with user-provided membership relationship, with internal binaries

1. Export from /f4 - single resource

```
java -jar ./bin/fcrepo-import-export-0.1.0.jar -b -a -p http://pcdm.org/models#hasMember -d data -m
export -r http://localhost:8686/f4/rest/collection0
```

   a. ✅ Audit log looks correct
   b. ✅ Correct resources were exported to data-dir, including internal binaries
2. Import to /fcrepo
   a. Per discovery in Test 0, moved dir data/f4 to data/fcrepo, before running:

```
java -jar ./bin/fcrepo-import-export-0.1.0.jar -b -a -d data -m import -s http://localhost:8686/f4
/rest/collection0 -r http://localhost:8080/fcrepo/rest/collection0
```

   b. ❓ Only targeted resource was imported
3. Import to /fcrepo, indicating membership predicate

```
java -jar ./bin/fcrepo-import-export-0.1.0.jar -b -a -p http://pcdm.org/models#hasMember -d data -m
import -s http://localhost:8686/f4/rest/collection0 -r http://localhost:8080/fcrepo/rest/collection0
```

   a. ❓Only targeted resource was imported
4. Import to /fcrepo, not specifying single resource

```
java -jar ./bin/fcrepo-import-export-0.1.0.jar -b -a -d data -m import -s http://localhost:8686/f4/rest/
-r http://localhost:8080/fcrepo/rest/
```

   a. ✅ Audit log looks correct
   b. ✅Correct resources were imported from data-dir, including internal binaries, relationships correct - pointing to internal resources

Test 3 Conclusions

1. ✅ User-provided membership produces expected export
2. ❌ Re-imports must either be for a single resource or for the full repository... not clear how to re-import a tree of resources based on user-provided membership predicates

Test 4 - Export/Import single resource - with user-provided membership relationship, with external binaries

1. Export from /f4 - single resource

```
java -jar ./bin/fcrepo-import-export-0.1.0.jar -b -a -p http://pcdm.org/models#hasMember -d data -m
export -r http://localhost:8686/f4/rest/collection1
```

   a. ✅ Audit log looks correct
   b. ✅Correct resources were exported to data-dir, including external binaries
2. Import to /fcrepo, not specifying resource... per lesson from Test 3

```
java -jar ./bin/fcrepo-import-export-0.1.0.jar -b -a -d data -m import -s http://localhost:8686/f4/rest/
-r http://localhost:8080/fcrepo/rest/
```

   a. ✅ Audit log looks correct
   b. ✅Correct resources were imported from data-dir, including external binaries, relationships correct - pointing to external resources

Test 5 - Export/Import - WebAC resources

Test 6 - Export/Import - Direct Containers

Test 7 - Export/Import - Indirect Containers