Simple LNI Client

This page describes a minimal DSpace LNI client. It was designed to be called from an application that produces DSpace SIPs and ingests them through the LNI, but it can also function as a standalone command-line application.

UNSUPPORTED: This is *unsupported* code. Use it at your own risk. It is simple enough that reading the code and examples should be enough to answer any questions you may have, but if not, there may not be any other help available. It is provided freely to the DSpace community but there are no support resources available.

What it Does

First, see the Lightweight Network Interface for a description of the LNI. This is an *LNI client*, the part that runs on your local computer and talks to a DSpace repository.

This is a very *minimal* client: it stands completely alone without any of the DSpace codebase. It is useful as example code and could be extended.

This client does *not* give access to all of the operations offered by the LNI. It only exposes public methods for the GET and PUT operations, which transfer DSpace Items as packages. GET of Bitstreams is not covered, although the LNI server allows it. It implements a couple of other operations (GET of *lookup* URLs and the *PROPFIND* verb) internally but they are not available as part of its API.

Example Code

The best way to illustrate the toolkit is through an example. This code fragment ingests an Item (in the form of a DSpace METS SIP) into a target collection named by its Handle:

```
import edu.mit.libraries.facade.app.LNIclient;

String collection = "some-handle";
LNIclient lni = new LNIclient("http://mydspace:8080/lni/dav/", "eperson@my.edu", "password");
OutputStream os = lni.startPut(collection, "METS", null);

// your application creates a SIP and writes it to the stream
your-code-to-write-METS-package-to-stream(os);

os.close();

// now your application records or displays the new Item's Handle.
String newItemhandle = lni.finishPut();
```

For more examples, see the `main()` method in the source, since it implements the command-line LNI client.

See the source code, which has Javadoc-style comments, for more complete documentation. You can also use `javadoc` to create web pages of API documentation.

Downloading and Running LNIclient

LNIclient.java

To use this class in your application, you will also need the following Java libraries on your classpath:

1. Apache Jakarta Commons HTTPClient version 3
2. Apache Commons Codec (required by HTTPClient)
3. Apache Commons Logging (required by HTTPClient)

4. Apache Commons CLI

NOTE: Some of these libraries are also required by DSpace 1.4 and 1.5, so they should be available in the `lib` subdirectory of your DSpace runtime installation. HTTPClient is the execption.

Future Work

As mentioned, this is *completely unsupported* code.
You are encouraged to use it, improve upon it, and use the Wiki to document and share your improvements.

Some useful extensions that could be added:

- Implement GET of a Bitstream
- Add methods and command-line options to do PROPFIND and access results.

If you find this documentation lacking, please add the missing answers to it.