

QGIS Application - Feature request #11733

add "add Data" functionality

2014-11-25 05:19 AM - Tom Kralidis

Status: Open	
Priority: Normal	
Assignee:	
Category: MetaSearch Catalogue Client	
Pull Request or Patch supplied:	Resolution:
Easy fix?: No	Copied to github as #: 19968
Description	
Allow the user to download data from the record (like a Shapefile, GeoTIFF, etc.) and load directly into the map canvas. See https://github.com/geopython/MetaSearch/wiki/Discovery-workflow for more info.	
tl; dr; waiting on CSW with records with links to zipped Shapefile or GeoTIFF for implementation.	
Full history at https://github.com/geopython/MetaSearch/issues/25	

History

#1 - 2015-08-21 04:56 AM - Tom Kralidis

- Project changed from 218 to QGIS Application

#2 - 2015-08-21 04:56 AM - Tom Kralidis

- Category set to MetaSearch Catalogue Client

#3 - 2015-12-22 06:20 AM - jorge luis infante

Hi

I'd like contribute with this functionality.

My idea is to implement the reading of the "missing" protocols (we have a lot in <https://github.com/geopython/pycsw/wiki/Geonode-notes>) in the library "owslib" or use the library "ftplib" in the plug, as shown in <https://docs.python.org/2/library/ftplib.html>?

I would also collaborate on access to PostgreSQL / PostGIS.

mainly because, in practice, access through wfs is extremely slow, for large layers (say, all parcels of a city like Rosario, Argentina).

Also, I do not know if there is, I would like to work on a mechanism checkin / checkout layers with concurrent access, and conflict resolution. Obviously, that's for later, but first I want to achieve, at least, to recover a layer efficiently, and have the possibility (which is why I thought ftp), of not having to expose the RDBMS network users.

#4 - 2015-12-22 06:37 AM - Tom Kralidis

Note that current efforts around link types can be found in <https://github.com/OSGeo/Cat-Interop/blob/master/LinkPropertyLookupTable.csv>

I would go about this as follows:

- add "Download Data" button
- look for startswith('WWW:DOWNLOAD') to activate the "Download Data" button
- at this point clicking the button should take you to a direct link
- of the data. Maybe a generic approach is to open the URL in a web
- browser if the URL is a directory listing (test with something like

```
if os.path.splitext(urlparse.urlparse('http://example.org/foo.xml').path)[1]:
```

download direct file

else:

open in browser

#5 - 2017-05-01 12:47 AM - Giovanni Manghi

- *Easy fix? set to No*