pyhydllp Documentation

Release 1.0.0

Mike Kittridge

Jun 19, 2020

SECTIONS

pyhydllp is a wrapper package that contains many Python functions for extracting data from Hydstra using the hydllp API. Detailed documentation about hydllp and relevant parameters can be found here: http://kisters.com.au/doco/hydllp.htm. You must have a Hydstra installation and license to run the functions.

The GitHub repository is found here

CHAPTER

ONE

INSTALLATION

Install via pip:

```
pip install pyhydllp
```

Or conda:

```
conda install -c mullenkamp pyhydllp
```

1.1 Requirements

At a minimum to access the base functions, pyhydllp requires a 32bit Python installation and Pandas.

To access the MSSQL functionality, the pdsql package is required:

```
conda install -c mullenkamp pdsql
```

CHAPTER

TWO

HOW TO USE PYHYDLLP

This section will describe how to use the pyhydllp package. The Hyd class and functions depend heavily on the Pandas package. Nearly all outputs are either as Pandas Series or DataFrames.

2.1 Initialising

The package and general usage is via the main Hyd class with the parameters shown below.

```
from pyhydllp import hyd

ini_path = r'\\fileservices02\ManagedShares\Data\Hydstra\prod\hyd'
dll_path = r'\\fileservices02\ManagedShares\Data\Hydstra\prod\hyd\sys\run'
username = ''
password = ''
hydllp_filename = 'hydllp.dll'
hyaccess_filename = 'Hyaccess.ini'
hyconfig_filename = 'HYCONFIG.INI'

hyd1 = hyd(ini_path, dll_path, hydllp_filename=hydllp_filename,
           hyaccess_filename=hyaccess_filename, hyconfig_filename=hyconfig_filename,
           username=username, password=password)
```

Then all of the functions can be accessed via the newly initiated hyd1 object. The following example won't work outside of ECan:

```
sites = [70105, 69607]
datasource = 'A'
varfrom = 100 # the 100 code is water level
varto = 140 # the 140 code is flow
qual_codes = [30, 20, 10 ,11, 21, 18] # It's best to specify as hydllp can
                                      # return bad values for a qual_code 255
from_mod_date = '2018-01-01'
to_mod_date = '2018-03-26'

sites_var = hyd1.get_variable_list(sites)

print(sites_var)

ch1 = hyd1.ts_data_changes(varto=[varfrom], sites=sites, from_mod_date=from_mod_date,
                           to_mod_date=to_mod_date)
print(ch1)
```

(continues on next page)

(continued from previous page)

```
tsdata = hyd1.get_ts_data(sites=sites, start=from_mod_date, end=to_mod_date,
                          varfrom=varfrom, varto=varto, datasource=datasource,
                          qual_codes=qual_codes)

print(tsdata)
```

CHAPTER

THREE

PACKAGE REFERENCES

3.1 Base class

class pyhydllp.**hyd**(*ini_path*, *dll_path*, *hydllp_filename='hydllp.dll'*, *hyaccess_filename='Hyaccess.ini'*, *hyconfig_filename='HYCONFIG.INI'*, *username=''*, *password=''*)

Class to initiate the Hydstra connection and access the extraction functions.

Parameters

- **ini_path** (*str or None*) – Path to the Hyaccess.ini file.
- **dll_path** (*str or None*) – Path to the hydllp.dll file.
- **hydllp_filename** (*str*) – The hydllp file name.
- **hyaccess_filename** (*str*) – The hyaccess file name.
- **hyconfig_filename** (*str*) – The hyconfig file name.
- **username** (*str*) – The login username for Hydstra. Leave a blank str to have Hydstra use the local user machine username.
- **password** (*str*) – Same as username, but for password.

Returns

Return type hyd object

3.2 Methods

hyd.**get_variable_list**(*self*, *sites*, *data_source='A'*)

Function to get the variables list for a list of sites.

Parameters

- **sites** (*list*) – A list of site names.
- **data_source** (*str*) – The data source.

Returns

Return type DataFrame

hyd.**get_ts_blockinfo**(*self*, *sites*, *datasources=['A']*, *variables=['100', '10', '110', '140', '130', '143', '450']*, *start='1900-01-01'*, *end='2100-01-01'*, *from_mod_date='1900-01-01'*, *to_mod_date='2100-01-01'*)

Wrapper function to extract info about when data has changed between modification dates.

Parameters

- **sites** (*list*) – Site numbers.
- **datasource** (*list of str*) – Hydstra datasource code (usually ['A']).
- **variables** (*list of int or float*) – The hydstra conversion data variable (140.00 is flow).
- **start** (*str*) – The start time in the format of '2001-01-01'.
- **end** (*str*) – Same formatting as start.
- **from_mod_date** (*str*) – The starting date of the modification.
- **to_mod_date** (*str*) – The ending date of the modification.

Returns With site, data_source, varto, from_mod_date, and to_mod_date.

Return type DataFrame

hyd.**ts_data_changes**(*self*, *varto*, *sites*, *data_source='A'*, *from_mod_date=None*, *to_mod_date=None*)

Function to determine the time series data indexed by sites and variables that have changed between the from_mod_date and to_mod_date. For non-flow rating sites/variables!!!

Parameters

- **varto** (*list of str*) – The hydstra conversion data variable (140.00 is flow).
- **sites** (*list of str*) – List of sites to be returned.
- **data_source** (*str*) – Hydstra datasource code (usually 'A').
- **from_mod_date** (*str*) – The starting date when the data has been modified.
- **to_mod_date** (*str*) – The ending date when the data has been modified.

Returns With site, varfrom, varto, from_date, and to_date

Return type DataFrame

hyd.**get_ts_data**(*self*, *sites*, *start=0*, *end=0*, *datasource='A'*, *data_type='mean'*, *varfrom=100*, *varto=140*, *qual_codes=None*, *interval='day'*, *multiplier=1*, *report_time=None*, *sites_chunk=20*, *print_sites=False*, *export_path=None*)

Wrapper function over hydllp to read in data from Hydstra's database. Must be run in a 32bit python. If either start_time or end_time is not 0, then they both need a date.

Parameters

- **sites** (*list, array, one column csv file, or dataframe*) – Site numbers.
- **start** (*str or int of 0*) – The start time in the format of either '2001-01-01' or 0 (for all data).
- **end** (*str or int of 0*) – Same formatting as start.
- **datasource** (*str*) – Hydstra datasource code (usually 'A').
- **data_type** (*str*) – mean, maxmin, max, min, start, end, first, last, tot, point, partialtot, or cum.
- **varfrom** (*int or float*) – The hydstra source data variable (100.00 is water level).
- **varto** (*int or float*) – The hydstra conversion data variable (140.00 is flow).

- **interval** (*str*) – The frequency of the output data (year, month, day, hour, minute, second, period). If data_type is 'point', then interval cannot be 'period' (use anything else, it doesn't matter).
- **multiplier** (*int*) – interval frequency.
- **qual_codes** (*list of int or None*) – The quality codes in Hydstra for filtering the data.
- **sites_chunk** (*int*) – Number of sites to request to hydllp at one time. Processing too many sites at a time may exceed the RAM.
- **report_time** (*start or end*) – Specifying the report_time as "end" will cause the time output with aggregated values for mean, total, and partial total data types to be the end of the period instead of the start.
- **print_sites** (*bool*) – print site names as they are extracted.

Returns In long format with site and time as a MultiIndex.

Return type DataFrame

3.3 API Pages

—

CHAPTER

FOUR

LICENSE AND TERMS OF USAGE

This package is licensed under the terms of the Apache License Version 2.0 and can be found on the GitHub project page.

INDEX

G

H

T