
IHEWAcollect Documentation

Release unknown

Quan Pan

Apr 05, 2023

CONTENTS

1	Installation	3
1.1	Install from pip	3
1.2	Install from source code	3
1.3	Uninstall IHEWAc collect	3
1.4	Optional data	3
2	Docker	5
3	Use	7
4	Products	9
5	Code of Conduct	11
6	Contents	13
6.1	Tutorial	13
6.1.1	Local environment	13
6.1.2	Virtual environment	13
6.2	License	13
6.3	Authors	23
6.4	Changelog	23
6.4.1	Version 0.0.1	23
6.5	Contributing	23
6.5.1	Code of Conduct	23
6.5.2	Rights	23
6.5.3	Issue Conventions	24
6.5.4	Dataset Objects	24
6.5.5	Git Conventions	24
6.5.6	Code Conventions	24
6.5.7	Development Environment	25
6.5.7.1	Initial Setup	25
6.5.7.2	Installing GDAL	25
6.5.7.3	Python build requirements	25
6.5.7.4	Installing IHEWAc collect	26
6.5.7.5	Uninstalling IHEWAc collect	26
6.5.7.6	Running the tests	26
6.6	Development	26
6.7	Products	29
6.8	IHEWAc collect	30
6.8.1	IHEWAc collect package	30
6.8.1.1	Subpackages	30

6.8.1.2	Submodules	41
6.8.1.3	IHEWAcollection.download module	41
6.8.1.4	Module contents	42
7	Indices and tables	43
	Python Module Index	45
	Index	47

This is the documentation of **IHEWAc collect**.

IHEWAc collect is a collection of tools to download remote sensing data. This project is fully developed by Water Accounting team at IHE-Delft.

GDAL library is required, see [How to install GDAL](#).

Warning: Set gdal executable files to system “PATH”.

Must contain **accounts.yml** file, for security reason, we don’t suggest you to share **accounts.yml**, see [accounts.yml-example](#).

accounts.yml-encrypted is safe to be shared, it can be generated by `generate_encrypt()` function.

accounts.yml-credential contains your password and key, to encrypt **accounts.yml-encrypted**.

INSTALLATION

1.1 Install from pip

IHEWAc collect Python package is hosted on [Python Package Index \(PyPI\)](#).

Install from pip.

```
$ pip install IHEWAc collect
```

1.2 Install from source code

Download source code from [Github](#).

```
$ git clone https://github.com/wateraccounting/IHEWAc collect.git  
$ cd IHEWAc collect
```

Install from source code.

```
$ python setup.py install
```

1.3 Uninstall IHEWAc collect

```
$ pip uninstall IHEWAc collect
```

1.4 Optional data

Test data can be download from [IHE FTP](#).

Save the test data to “IHEWAc collect/tests/data/Products” folder.

DOCKER

Set “System -> Base Memory” to max capacity in Virtual Machine.

Set “Shared Folders” in Virtual Machine.

- Folder Path: D:\IHEWAcollect
- Folder Name: d/IHEWAcollect
- Auto-mount: Yes
- Make Permanent: Yes

Download source code from [Github](#).

```
$ git clone https://github.com/wateraccounting/IHEWAcollect.git  
$ cd IHEWAcollect
```

Restart Docker Virtual Machine image.

```
$ docker-machine restart
```

Pull the ihewacollect image.

```
$ docker pull wateraccounting/ihewacollect
```

Or build from source code.

```
$ docker build -t wateraccounting/ihewacollect .
```

Check images.

```
$ docker images
```

Check ip address.

```
$ docker-machine env  
export DOCKER_HOST="tcp://192.168.99.100:2376"
```

Run the image with Jupyter Notebook. To access Jupyter <http://192.168.99.100:8888/>

```
$ docker run -it --name ihewacollect -p 8888:8888 -v /d/IHEWAcollect:/notebooks  
↳ wateraccounting/ihewacollect
```

Check running image (container), in the new cmd window.

```
$ docker container list
```

Access to running image, in the new cmd window.

```
$ docker exec -it ihewacollect bash  
$ cd /notebooks/  
$ python3 setup.py install  
$ python3 ./examples/ex_CFSR_GRIB.py
```

Clean running cache.

```
$ docker system prune -f && docker volume prune -f && docker container prune -f
```

Remove image.

```
$ docker rmi wateraccounting/ihewacollect
```

Examples can be found at [examples](#).

```
import os
import IHEWAcollect

path = os.getcwd()
file_accounts = os.path.join(path, 'accounts.yml')
file_credential = os.path.join(path, 'accounts.yml-credential')
file_encrypted = os.path.join(path, 'accounts.yml-encrypted')

if os.path.exists(file_accounts) or os.path.exists(file_encrypted):
    product = 'CFSR'
    version = 'v1'
    parameter = 'radiation'
    resolution = 'daily'
    variable = 'dlwsfc'
    bbox = {
        'w': -19.0,
        'n': 38.0,
        'e': 55.0,
        's': -35.0
    }
    period = {
        's': '2008-01-01',
        'e': '2008-01-31'
    }
    nodata = -9999

    IHEWAcollect.Download(workspace=path,
                          product=product,
                          version=version,
                          parameter=parameter,
                          resolution=resolution,
                          variable=variable,
                          bbox=bbox,
                          period=period,
                          nodata=nodata,
                          is_status=False)
```


PRODUCTS

Details can be found at [Products](#).

Product	Link
ASCAT	https://www.copernicus.eu
CFSR	https://www.noaa.gov
CHIRPS	https://geog.ucsb.edu
CSR	https://podaac.jpl.nasa.gov
DEM	http://earlywarning.usgs.gov
ECMWF	https://www.ecmwf.int
FEWS	https://earlywarning.usgs.gov/fews
GLDAS	https://ldas.gsfc.nasa.gov/gldas
GLEAM	http://www.gleam.eu
GPM	https://pmm.gsfc.nasa.gov/GPM
HiHydroSoil	No longer provided. Download current version from: www.futurewater.eu/projects/hihydrosoil/
JRC	https://global-surface-water.appspot.com
MCD12Q1	https://lpdaac.usgs.gov/products/mcd12q1v006
MCD43A3	https://lpdaac.usgs.gov/products/mcd43a3v006
MOD09GQ	https://lpdaac.usgs.gov/products/mod09gqv006
MOD10A2	https://nsidc.org/data/MOD10A2/versions/6
MOD11A2	https://lpdaac.usgs.gov/products/mod11a2v006
MOD13Q1	https://lpdaac.usgs.gov/products/mod13q1v006
MOD15A2H	https://lpdaac.usgs.gov/products/mod15a2hv006
MOD16A2	https://lpdaac.usgs.gov/products/mod16a2v006
MOD17A2H	https://lpdaac.usgs.gov/products/mod17a2hv006
MOD17A3H	https://lpdaac.usgs.gov/products/mod17a3hv006
MSWEP	https://princetonclimate.com
PROBAV	http://proba-v.vgt.vito.be/en
RFE	https://data.noaa.gov/dataset
SoilGrids	https://www.isric.org
TRMM	https://pmm.nasa.gov/trmm
TWC	https://www.wateraccounting.org

CODE OF CONDUCT

- Be friendly and patient
- Be welcoming
- Be considerate
- Be respectful
- Be careful in the words that you choose
- When we disagree, try to understand why

CONTENTS

6.1 Tutorial

We prepare solutions for different environment to use **IHEWAc collect**,

- Local environment
- Virtual environment

6.1.1 Local environment

6.1.2 Virtual environment

6.2 License

GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <<https://fsf.org/>> Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps:

(1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents.

States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices”

to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The “source code” for a work means the preferred form of the work for making modifications to it. “Object code” means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work’s System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under

the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
- b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
- c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
- d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms

of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

- a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
- b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
- c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.
- d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.
- e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the

Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

“Additional permissions” are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
- c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
- e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
- f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you

must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party’s predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor’s “contributor version”.

A contributor’s “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor’s essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient’s use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others’ Freedom.

If conditions are imposed on you (whether by court order, agreement or

otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have

permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the

Program specifies that a certain numbered version of the GNU General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy’s public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different

permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided

above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest

possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest

to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the “copyright” line and a pointer to where the full notice is found.

<one line to give the program’s name and a brief idea of what it does.> Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<https://www.gnu.org/licenses/>>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short

notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author> This program comes with ABSOLUTELY NO WARRANTY; for details type `show w`. This is free software, and you are welcome to redistribute it under certain conditions; type `show c` for details.

The hypothetical commands `show w` and `show c` should show the appropriate parts of the General Public License. Of course, your program’s commands might be different; for a GUI interface, you would use an “about box”.

You should also get your employer (if you work as a programmer) or school,

if any, to sign a “copyright disclaimer” for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <<https://www.gnu.org/licenses/>>.

The GNU General Public License does not permit incorporating your program

into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <<https://www.gnu.org/licenses/why-not-lgpl.html>>.

6.3 Authors

- Marloes Mul <m.mul@un-ihe.org>
- Claire Michailovsky <c.michailovsky@un-ihe.org>
- Elga Salvadore <e.salvadore@un-ihe.org>
- Solomon Seyoum <s.seyoum@un-ihe.org>
- Bert Coerver <b.coerver@un-ihe.org>
- Bich Tran <b.tran@un-ihe.org>
- Abebe Chukalla <a.chukalla@un-ihe.org>
- Tim Hessels <t.hessels@unesco-ihe.org>
- Gonzalo Espinoza <g.espinoza@unesco-ihe.org>
- Quan Pan <quanpan302@hotmail.com>

6.4 Changelog

6.4.1 Version 0.0.1

- Template of CHANGELOG.RST!
- Feature A added
- FIX: nasty bug #1729 fixed

6.5 Contributing

Welcome to the IHE WaterAccounting Collect Tool (IHEWAc collect) project. Here's how we work.

6.5.1 Code of Conduct

First of all: the IHEWAc collect project has a code of conduct. Please read the CODE_OF_CONDUCT.txt file, it's important to all of us.

6.5.2 Rights

The GNU General Public License v3.0 license (see LICENSE.rst) applies to all contributions.

6.5.3 Issue Conventions

The IHEWAc collect issue tracker is for actionable issues.

Questions about installation, distribution, and usage should be taken to the project's [#issue](https://wateraccounting.slack.com). Opened issues which fall into one of these three categories may be perfunctorily closed.

Questions about development of IHEWAc collect, brainstorming, requests for comment, and not-yet-actionable proposals are welcome in the project's [#idea](https://wateraccounting.slack.com).

IHEWAc collect is a relatively new project and highly active. We have bugs, both known and unknown.

Please search existing issues, open and closed, before creating a new one.

IHEWAc collect employs Geospatial modules, so bug reports very often hinge on the following details:

- Operating system type and version (Windows? Ubuntu 12.04? 14.04?)
- The version and source of GDAL (UbuntuGIS? Homebrew?)

6.5.4 Dataset Objects

Our term for the kind of object that allows read and write access to raster data is *dataset object*. A dataset object might be an instance of *DatasetReader* or *DatasetWriter*. The canonical way to create a dataset object is by using the *IHEWAc collect.GIS.OpenAsArray()* function.

This is analogous to Python's use of [file object](#).

6.5.5 Git Conventions

We use a variant of centralized workflow described in the [Git Book](#). We have no 1.0 release for IHEWAc collect yet and we are tagging and releasing from the master branch. Our post-1.0 workflow is to be decided.

Work on features in a new branch of the mapbox/IHEWAc collect repo or in a branch on a fork. Create a [GitHub pull request](#) when the changes are ready for review. We recommend creating a pull request as early as possible to give other developers a heads up and to provide an opportunity for valuable early feedback.

6.5.6 Code Conventions

IHEWAc collect supports Python 2 and Python 3 in the same code base, which is aided by an internal compatibility module named `compat.py`. It functions similarly to the more widely known `six` but we only use a small portion of the features so it eliminates a dependency.

We strongly prefer code adhering to [PEP8](#).

Tests are mandatory for new features. We use [pytest](#).

We aspire to 100% coverage for Python modules [Coveralls](#).

6.5.7 Development Environment

Developing IHEWAc collect requires Python 2.7 or any final release after and including 3.4. We prefer developing with the most recent version of Python but recognize this is not possible for all contributors. See the Windows install instructions in the [readme](#) for more information about building on Windows.

6.5.7.1 Initial Setup

First, clone IHEWAc collect's git repo:

```
$ git clone https://github.com/wateraccounting/ihewacollect.git
```

Development should occur within a [virtual environment](#) to better isolate development work from custom environments.

In some cases installing a library with an accompanying executable inside a virtual environment causes the shell to initially look outside the environment for the executable. If this occurs try deactivating and reactivating the environment.

6.5.7.2 Installing GDAL

The GDAL library and its headers are required to build IHEWAc collect. We do not have currently have guidance for any platforms other than Linux and OS X.

On Linux, GDAL and its headers should be available through your distro's package manager. For Ubuntu the commands are:

```
$ sudo add-apt-repository ppa:ubuntugis/ppa
$ sudo apt-get update
$ sudo apt-get install gdal-bin libgdal-dev
```

On OS X, Homebrew is a reliable way to get GDAL.

```
$ brew install gdal
```

6.5.7.3 Python build requirements

Provision a virtualenv with IHEWAc collect's build requirements. IHEWAc collect's `setup.py` script will not run unless Cython and Numpy are installed, so do this first from the IHEWAc collect repo directory.

Linux users may need to install some additional Numpy dependencies:

```
$ sudo apt-get install libatlas-dev libatlas-base-dev gfortran
```

then:

```
$ pip install -U pip
$ pip install -r requirements-dev.txt
```

6.5.7.4 Installing IHEWAccollect

Installing IHEWAccollect in editable mode while developing is very convenient but only affects the Python files.

```
$ python setup.py install
```

6.5.7.5 Uninstalling IHEWAccollect

```
$ pip uninstall IHEWAccollect
```

6.5.7.6 Running the tests

IHEWAccollect's tests live in `python setup.py test` and generally match the main package layout.

To run the entire suite and the code coverage report:

```
$ python setup.py test
```

6.6 Development

Download source code from [Github](https://github.com/wateraccounting/IHEWAccollect).

```
$ git clone https://github.com/wateraccounting/IHEWAccollect.git
$ cd IHEWAccollect
```

In the PyCharm IDE, change “Project Structure -> Source Folders” to “src”

From the root of the project

```
$ python setup.py --version
```

Format scripts by PEP8

```
$ autopep8 --in-place --aggressive src/IHEWAccollect/base/base.py
```

Flake8, pre-commit

```
$ pre-commit install

$ pre-commit run --all-files
[INFO] Initializing environment for git://github.com/pre-commit/pre-commit-hooks.
[INFO] Initializing environment for https://github.com/pre-commit/mirrors-isort.
[INFO] Installing environment for git://github.com/pre-commit/pre-commit-hooks.
[INFO] Once installed this environment will be reused.
[INFO] This may take a few minutes...
[INFO] Installing environment for https://github.com/pre-commit/mirrors-isort.
[INFO] Once installed this environment will be reused.
[INFO] This may take a few minutes...
```

Unit test

```
$ python setup.py test
```

Read the Docs

```
$ python setup.py doctest
```

```
$ python setup.py docs
```

Upload to PyPI

1. In IDE, **commit** the changes “**v0.0.1**”
2. In IDE, **Version Control -> Log**, select this commit
3. In IDE, add version tag, select **VCS -> Git -> tag**
4. In IDE, **Tag window -> Tag Name**, type “**v0.0.1**”
5. In cmd, build package, type `python setup.py sdist bdist_wheel`
6. In cmd, validate build, type `twine check dist/IHEWAc collect-0.0.1*`
7. In cmd, upload build, type `twine upload dist/IHEWAc collect-0.0.1*`
8. In IDE, **push** the commit, with Tag label: “*HEAD*”, “*master*”, “*v0.0.1*”
9. In Github, select **Release** to “create a new release” or “Draft a new release”
10. In Github, **Tag version**, type “**v0.0.1**”
11. In Github, @ **Target**, select this commit
12. In Github, **Publish release**

6.7 Products

Table 1: Product Detail

id	variable	product	account	protocol	version	freq.	parameters	resolution	lat_s	lat_n	time_s	time_e
1	Albedo	MCD43A3	NASA	HTTPS	v6	daily	Albedo AlbedoBSA	500m	Global	Global	24/02/2000	contin- ues
2	Digital Elevation Map	HydroSHEDS	None	HTTPS	v1	n/a	DEM	3s 15s 30s			n/a	n/a
3	Equivalent Water Height	GRACE-FO- CSR/GFZ/DAAC	DAAC	HTTPS	v3.1	daily	EWB	1°	-90	90	01/04/2000	06/2017
4	Equivalent Water Height	GRACE-FO- DAAC CSR/GFZ/JPL	DAAC	HTTPS	v3.2	daily	EWB	1°	-90	90	01/06/2018	contin- ues
5	Evapo- transpira- tion	GLDAS	NASA	HTTPS	v2.1	3 hourly monthly	ETa	0.25°	-60	90	01/01/2000	contin- ues
6	Evapo- transpira- tion	GLEAM	GLEAM	SFTP	v3.3a v3.3b	daily monthly	ETa	0.25°	-90	90	01/01/1980	12/2018
7	Evapo- transpira- tion	MOD16A2	NASA	HTTPS	v6	8-day	ETA	500m	Global	Global	01/01/2000	contin- ues
8	Evapo- transpira- tion	SSE-Bop	None	HTTPS	v4	monthly	ETa	1km	-60	80	01/01/2000	contin- ues
9	Flow Di- rec- tion Map	HydroSHEDS	None	HTTPS	v1	n/a	DIR	3s 15s 30s			n/a	n/a
10	Gross Pri- mary Pro- duc- tion	MOD17A2	NASA	HTTPS	v6	8-day	GPP	500m	Global	Global	18/02/2000	contin- ues
11	Global Cover	MCD12Q1	NASA	HTTPS	v6	yearly	LC	500m	Global	Global	01/01/2001	12/2018
12	Land Sur- face Temperature	MOD11A2	NASA	HTTPS	v6	8-day	LST-day LST	1km	Global	Global	18/02/2000	contin-

6.8 IHEWAc collect

6.8.1 IHEWAc collect package

6.8.1.1 Subpackages

IHEWAc collect.base package

Submodules

IHEWAc collect.base.base module

Base

Examples:

```
from IHEWAc collect.base import Base

base = Base(product='CFSR', is_print=True)
```

```
class IHEWAc collect.base.base.Base(product, is_print)
```

Bases: `object`

This Base class

Load base.yml file.

Parameters

- **product** (*str*) – Product name of data products.
- **is_print** (*bool*) – Is to print status message.

```
classmethod check_conf(key, is_print) → dict
```

Check configuration information

This is the function to get user's configuration data.

Don't synchronize the details to github.

- File to read: `collect.yml`.

Parameters

key (*str*) – Key name.

Returns

Configuration data.

Return type

`dict`

Example

```
>>> from IHEWAc collect.base.base import Base
>>> conf = Base.check_conf('data', is_print=False)
>>> conf['messages'][0]
{'msg': 'No error', 'level': 0}
```

check_input(*vname*, *rtype*, *vdata*) → *bool*

get_conf(*key*) → *dict*

Get configuration information

This is the function to get project's configuration data.

Parameters

key (*str*) – Key name.

Returns

Configuration data.

Return type

dict

Example

```
>>> import os
>>> from IHEWAc collect.base.base import Base
>>> base = Base(is_print=False)
>>> file = base.get_conf('file')
>>> print(file)
base.yml
```

get_status() → *str*

Get status

This is the function to get project status.

Returns

Status message.

Return type

str

status = 'Global status.'

IHEWAc collect.base.exception module

Exception

<https://julien.danjou.info/python-exceptions-guide/>

exception IHEWAc collect.base.exception.IHEClassInitError(*mod*, *msg=None*)

Bases: *Exception*

IHEClassInitError Class

Parameters

- **mod** (*str*) – Module name.
- **msg** (*bool*) – Extra message.

exception IHEWAc collect.base.exception.IHEFileError(*file*, *msg=None*)

Bases: *Exception*

IHEFileError Class

Parameters

- **file** (*str*) – File name.
- **msg** (*bool*) – Extra message.

exception IHEWAc collect.base.exception.IHEKeyError(*key, val, msg=None*)

Bases: [Exception](#)

IHEKeyError Class

Parameters

- **key** (*str*) – Key name.
- **val** (*list*) – Key name list.
- **msg** (*bool*) – Extra message.

exception IHEWAc collect.base.exception.IHEPassError(*vname, msg=None*)

Bases: [Exception](#)

IHEPassError Class

Parameters

- **vname** (*str*) – Account name.
- **msg** (*bool*) – Extra message.

exception IHEWAc collect.base.exception.IHEStringError(*vname, msg=None*)

Bases: [Exception](#)

IHEStringError Class

Parameters

- **vname** (*str*) – Variable name.
- **msg** (*bool*) – Extra message.

exception IHEWAc collect.base.exception.IHETypeError(*vname, rtype, vdata, msg=None*)

Bases: [Exception](#)

IHETypeError Class

Parameters

- **vname** (*str*) – Variable name.
- **rtype** (*str*) – Required type.
- **vdata** (*float*) – Variable value.
- **msg** (*bool*) – Extra message.

IHEWAc collect.base.user module

User

Before use this module, set account information in the `accounts.yml` file.

Examples:

```
from IHEWAc collect.base.user import User

user = User(workspace=path, product='CFSR', is_print=True)
```

Note:

1. Create `accounts.yml` under root folder of the project, based on the `config-example.yml`.
2. Run `User.credential.encrypt_cfg(path, file, password)` to generate `accounts.yml-encrypted` file.
3. Save key to `credential.yml`.

class IHEWAc collect.base.user.User(workspace, product, is_print, **kwargs)

Bases: `Base`

This User class

Description

Parameters

- **workspace** (`str`) – Directory to `accounts.yml`.
- **product** (`str`) – Product name of data products.
- **is_print** (`bool`) – Is to print status message.
- **kwargs** (`dict`) – Other arguments.

generate_encrypt()

Generate encrypted files

get_user(key)

Get user information

This is the function to get user's configuration data.

Don't synchronize the details to github.

- File to read: `accounts.yml-credential` contains key: `accounts.yml-encrypted`.
- File to read: `accounts.yml-encrypted` generated from: `accounts.yml`.

Parameters

key (`str`) – Key name.

Returns

User data.

Return type

`dict`

Example

```
>>> import os
>>> from IHEWAc collect.base.user import User
>>> user = User(os.getcwd(), 'FTP_WA_GUESS', is_print=False)
>>> account = user.get_user('account')
>>> account['FTP_WA_GUESS']
{'username': 'wateraccountingguest', 'password': 'W@t3r@ccounting', ..
↪ .
>>> accounts = user.get_user('accounts')
Traceback (most recent call last):
```

(continues on next page)

(continued from previous page)

```
...
KeyError:
```

```
set_status(fun="", prt=False, ext="")
```

Set status

Parameters

- **fun** (*str*) – Function name.
- **prt** (*bool*) – Is to print on screen?
- **ext** (*str*) – Extra message.

```
status = 'Global status.'
```

Module contents

IHEWAc collect base modules

IHEWAc collect.scripts package**Submodules****IHEWAc collect.scripts.credential module****IHEWAc collect.scripts.main module**

Main command group for WaterAccounting Collect's CLI.

IHEWAc collect.scripts.skeleton module

This is a skeleton file that can serve as a starting point for a Python console script. To run this script uncomment the following lines in the [options.entry_points] section in setup.cfg:

```
console_scripts =
    fibonacci = IHEWAc collect.skeleton:run
```

Then run *python setup.py install* which will install the command *fibonacci* inside your current environment. Besides console scripts, the header (i.e. until `_logger...`) of this file can also be used as template for Python modules.

Note: This skeleton file can be safely removed if not needed!

IHEWAc collect.scripts.skeleton.fib(*n*)

Fibonacci example function

Parameters**n** (*int*) – integer**Returns**

n-th Fibonacci number

Return type`int``IHEWAc collect.scripts.skeleton.main(args)`

Main entry point allowing external calls

Parameters**args** (`[str]`) – command line parameter list`IHEWAc collect.scripts.skeleton.parse_args(args)`

Parse command line parameters

Parameters**args** (`[str]`) – command line parameters as list of strings**Returns**

command line parameters namespace

Return type`argparse.Namespace``IHEWAc collect.scripts.skeleton.run()`

Entry point for console_scripts

`IHEWAc collect.scripts.skeleton.setup_logging(loglevel)`

Setup basic logging

Parameters**loglevel** (`int`) – minimum loglevel for emitting messages**Module contents**

IHEWAc collect script modules

IHEWAc collect.templates package**Subpackages****IHEWAc collect.templates.EU package****Submodules****IHEWAc collect.templates.EU.Copernicus module****IHEWAc collect.templates.EU.ECMWF module****IHEWAc collect.templates.EU.GLEAM module****IHEWAc collect.templates.EU.JRC module****Module contents**

IHEWAc collect.templates.FAO package

Submodules

IHEWAc collect.templates.FAO.WaPOR module

IHEWAc collect.templates.FAO.WaporAPI module

Authors: Bich Tran

IHE Delft 2019

Contact: b.tran@un-ihe.org

Module contents

IHEWAc collect.templates.HydroSHEDS package

Submodules

IHEWAc collect.templates.HydroSHEDS.DEM module

Module contents

IHEWAc collect.templates.IHE package

Submodules

IHEWAc collect.templates.IHE.ALEXI module

IHEWAc collect.templates.IHE.CMRSET module

IHEWAc collect.templates.IHE.ETmonitor module

IHEWAc collect.templates.IHE.ETmonitor_create module

IHEWAc collect.templates.IHE.HiHydroSoil module

IHEWAc collect.templates.IHE.SEBS module

IHEWAc collect.templates.IHE.TWC module

Module contents

IHEWAc collect.templates.ISRIC package

Submodules

IHEWAc collect.templates.ISRIC.SoilGrids module

Module contents

IHEWAc collect.templates.MSWEP package

Module contents

IHEWAc collect.templates.NASA package

Submodules

IHEWAc collect.templates.NASA.CSR module

IHEWAc collect.templates.NASA.GFZ module

IHEWAc collect.templates.NASA.GLDAS module

IHEWAc collect.templates.NASA.GPM module

IHEWAc collect.templates.NASA.JPL module

IHEWAc collect.templates.NASA.PODAAC module

IHEWAc collect.templates.NASA.TRMM module

Module contents

IHEWAc collect.templates.NOAA package

Submodules

IHEWAc collect.templates.NOAA.CFSR module

IHEWAc collect.templates.NOAA.RFE module

Module contents

IHEWAc collect.templates.USGS package

Submodules

IHEWAc collect.templates.USGS.CHIRPS module

IHEWAc collect.templates.USGS.FEWS module

[IHEWAc collect.templates.USGS.MODIS_products module](#)

[IHEWAc collect.templates.USGS.SSEBop module](#)

[Module contents](#)

[IHEWAc collect.templates.VITO package](#)

[Submodules](#)

[IHEWAc collect.templates.VITO.ASCAT module](#)

[IHEWAc collect.templates.VITO.PROBAV module](#)

[Module contents](#)

[Submodules](#)

[IHEWAc collect.templates.collect module](#)

[IHEWAc collect.templates.download_tiles_test module](#)

[IHEWAc collect.templates.dtime module](#)

Dtime

Examples:

```
from IHEWAc collect.templates.dtime import Dtime

dtime = Dtime(workspace=path, is_print=True)
```

```
class IHEWAc collect.templates.dtime.Dtime(_Dtime__status, _Dtime__conf, **kwargs)
```

Bases: `object`

```
conf = {'data': {}, 'dtime': {'i': 0, 'r': []}, 'file': {'i': '', 'o': ''},
'path': '', 'time': {'e': datetime.datetime(2023, 4, 5, 12, 55, 53, 880130),
's': datetime.datetime(2023, 4, 5, 12, 55, 53, 880121)}}
```

```
get_time_range(dtime_s, dtime_e, arg_resolution)
```

```
set_status(fun="", prt=False, ext="")
```

Set status

Parameters

- **fun** (*str*) – Function name.
- **prt** (*bool*) – Is to print on screen?
- **ext** (*str*) – Extra message.

```
status = 'Global status.'
```

IHEWAc collect.templates.gis module

IHEWAc collect.templates.util module

utils

Utilities for IHEWAc collect template modules.

Examples:

```
from IHEWAc collect.templates.util import Extract, Plot, Waitbar, Log
```

```
class IHEWAc collect.templates.util.Extract(file, folder, is_print)
```

Bases: `object`

Extract class

File pre-process, extract.

Parameters

- **file** (*str*) – File.
- **folder** (*str*) – Folder.
- **is_print** (*bool*) – Is to print status message.

gz()

Extract gz file

This function extract gz file.

Parameters

- **file** (*str*) – Name of the file that must be unzipped.
- **outfile** (*str*) – Directory where the unzipped data must be stored.

tar()

Extract tar file

This function extract tar file.

Parameters

- **file** (*str*) – Name of the file that must be unzipped.
- **outfile** (*str*) – Directory where the unzipped data must be stored.

zip()

Extract zip file

This function extract zip file.

Parameters

- **file** (*str*) – Name of the file that must be unzipped.
- **outfile** (*str*) – Directory where the unzipped data must be stored.

```
class IHEWAc collect.templates.util.Log(config, **kwargs)
```

Bases: `object`

Log class

Write message to log file.

Parameters

config (*dict*) – Is to print status message.

data = {}

write(*time*, *msg*="")

class IHEWAcollect.templates.util.**Plot**(*data*, *file*, *folder*, *is_print*, *is_save*, *is_show*)

Bases: *object*

Plot class

File post-process, save or show.

Parameters

- **data** –
- **file** –
- **folder** –
- **is_print** (*bool*) – Is to print status message.
- **is_save** (*bool*) – Is to save files in the folder.
- **is_show** (*bool*) – Is to show files in the folder.

class IHEWAcollect.templates.util.**Waitbar**(*is_print*)

Bases: *object*

Waitbar class

Waitbar on the cmd window.

Parameters

is_print (*bool*) – Is to print status message.

static wait_bar(*i*, *total*, *prefix*="", *suffix*="", *decimals*=1, *length*=100, *fill*="")

Wait Bar Console

This function will print a waitbar in the console

Parameters

- **i** (*int*) – Iteration number.
- **total** (*int*) – Total iterations.
- **prefix** (*str*) – Prefix name of bar.
- **suffix** (*str*) – Suffix name of bar.
- **decimals** (*int*) – Decimal of the wait bar.
- **length** (*int*) – Width of the wait bar.
- **fill** (*str*) – Bar fill.

Module contents

IHEWAc collect template modules

6.8.1.2 Submodules

6.8.1.3 IHEWAc collect.download module

Download

Before use this module, create `accounts.yml` file. And edit account information in the file.

```
class IHEWAc collect.download.Download(workspace="", product="", version="", parameter="", resolution="",
                                       variable="",
                                       acct_path='/home/docs/checkouts/readthedocs.org/user_builds/ihewacollect/checkouts',
                                       bbox={}, period={}, nodata=- 9999, is_status=True,
                                       is_save_temp=False, is_save_remote=False, is_save_list=False,
                                       **kwargs)
```

Bases: *User*

Download class

After initialise the class, data downloading will automatically start.

Parameters

- **workspace** (*str*) – Directory to `accounts.yml`.
- **product** (*str*) – Product name.
- **version** (*str*) – Version name.
- **parameter** (*str*) – Parameter name.
- **resolution** (*str*) – Resolution name.
- **variable** (*str*) – Variable name.
- **bbox** (*dict*) – Spatial range, {'w':, 's':, 'e':, 'n':}.
- **period** (*dict*) – Time range, {'s':, 'e':}.
- **nodata** (*int*) – -9999.
- **is_status** (*bool*) – Is to print status message.
- **kwargs** (*dict*) – Other arguments.

get_products() → *dict*

Get details of all products

Returns

Products data.

Return type

dict

```
status = 'Global status.'
```

6.8.1.4 Module contents

IHEWAcollect: IHE Water Accounting Collect Tools

```
class IHEWAcollect.Download(workspace="", product="", version="", parameter="", resolution="", variable="",
                           acct_path='/home/docs/checkouts/readthedocs.org/user_builds/ihewacollect/checkouts/stable/docs',
                           bbox={}, period={}, nodata=- 9999, is_status=True, is_save_temp=False,
                           is_save_remote=False, is_save_list=False, **kwargs)
```

Bases: *User*

Download class

After initialise the class, data downloading will automatically start.

Parameters

- **workspace** (*str*) – Directory to accounts.yml.
- **product** (*str*) – Product name.
- **version** (*str*) – Version name.
- **parameter** (*str*) – Parameter name.
- **resolution** (*str*) – Resolution name.
- **variable** (*str*) – Variable name.
- **bbox** (*dict*) – Spatial range, {'w':, 's':, 'e':, 'n':}.
- **period** (*dict*) – Time range, {'s':, 'e':}.
- **nodata** (*int*) – -9999.
- **is_status** (*bool*) – Is to print status message.
- **kwargs** (*dict*) – Other arguments.

get_products() → *dict*

Get details of all products

Returns

Products data.

Return type

dict

status = 'Global status.'

INDICES AND TABLES

- `genindex`
- `modindex`
- `search`

PYTHON MODULE INDEX

i

- IHEWAc collect, 42
- IHEWAc collect.base, 34
- IHEWAc collect.base.base, 30
- IHEWAc collect.base.exception, 31
- IHEWAc collect.base.user, 32
- IHEWAc collect.download, 41
- IHEWAc collect.scripts, 35
- IHEWAc collect.scripts.credential, 34
- IHEWAc collect.scripts.main, 34
- IHEWAc collect.scripts.skeleton, 34
- IHEWAc collect.templates, 41
- IHEWAc collect.templates.dtime, 38
- IHEWAc collect.templates.EU, 35
- IHEWAc collect.templates.FAO.WaporAPI, 36
- IHEWAc collect.templates.HydroSHEDS, 36
- IHEWAc collect.templates.IHE, 36
- IHEWAc collect.templates.ISRIC, 37
- IHEWAc collect.templates.MSWEP, 37
- IHEWAc collect.templates.NASA, 37
- IHEWAc collect.templates.NOAA, 37
- IHEWAc collect.templates.USGS, 38
- IHEWAc collect.templates.util, 39
- IHEWAc collect.templates.VITO, 38

B

Base (class in *IHEWAc collect.base.base*), 30

C

check_conf() (*IHEWAc collect.base.base.Base* class method), 30

check_input() (*IHEWAc collect.base.base.Base* method), 30

conf (*IHEWAc collect.templates.dtime.Dtime* attribute), 38

D

data (*IHEWAc collect.templates.util.Log* attribute), 40

Download (class in *IHEWAc collect*), 42

Download (class in *IHEWAc collect.download*), 41

Dtime (class in *IHEWAc collect.templates.dtime*), 38

E

Extract (class in *IHEWAc collect.templates.util*), 39

F

fib() (in module *IHEWAc collect.scripts.skeleton*), 34

G

generate_encrypt() (*IHEWAc collect.base.user.User* method), 33

get_conf() (*IHEWAc collect.base.base.Base* method), 31

get_products() (*IHEWAc collect.Download* method), 42

get_products() (*IHEWAc collect.download.Download* method), 41

get_status() (*IHEWAc collect.base.base.Base* method), 31

get_time_range() (*IHEWAc collect.templates.dtime.Dtime* method), 38

get_user() (*IHEWAc collect.base.user.User* method), 33

gz() (*IHEWAc collect.templates.util.Extract* method), 39

I

IHEClassInitError, 31

IHEFileError, 31

IHEKeyError, 32

IHEPassError, 32

IHEStringError, 32

IHETypeError, 32

IHEWAc collect

module, 42

IHEWAc collect.base

module, 34

IHEWAc collect.base.base

module, 30

IHEWAc collect.base.exception

module, 31

IHEWAc collect.base.user

module, 32

IHEWAc collect.download

module, 41

IHEWAc collect.scripts

module, 35

IHEWAc collect.scripts.credential

module, 34

IHEWAc collect.scripts.main

module, 34

IHEWAc collect.scripts.skeleton

module, 34

IHEWAc collect.templates

module, 41

IHEWAc collect.templates.dtime

module, 38

IHEWAc collect.templates.EU

module, 35

IHEWAc collect.templates.FAO.WaporAPI

module, 36

IHEWAc collect.templates.HydroSHEDS

module, 36

IHEWAc collect.templates.IHE

module, 36

IHEWAc collect.templates.ISRIC

module, 37

IHEWAc collect.templates.MSWEP

module, 37

IHEWAc collect.templates.NASA

module, 37

IHEWAc collect.templates.NOAA

module, 37

`IHEWAccollect.templates.USGS`
module, 38
`IHEWAccollect.templates.util`
module, 39
`IHEWAccollect.templates.VITO`
module, 38

L

`Log` (class in `IHEWAccollect.templates.util`), 39

M

`main()` (in module `IHEWAccollect.scripts.skeleton`), 35
module
 `IHEWAccollect`, 42
 `IHEWAccollect.base`, 34
 `IHEWAccollect.base.base`, 30
 `IHEWAccollect.base.exception`, 31
 `IHEWAccollect.base.user`, 32
 `IHEWAccollect.download`, 41
 `IHEWAccollect.scripts`, 35
 `IHEWAccollect.scripts.credential`, 34
 `IHEWAccollect.scripts.main`, 34
 `IHEWAccollect.scripts.skeleton`, 34
 `IHEWAccollect.templates`, 41
 `IHEWAccollect.templates.dtime`, 38
 `IHEWAccollect.templates.EU`, 35
 `IHEWAccollect.templates.FAO.WaporAPI`, 36
 `IHEWAccollect.templates.HydroSHEDS`, 36
 `IHEWAccollect.templates.IHE`, 36
 `IHEWAccollect.templates.ISRIC`, 37
 `IHEWAccollect.templates.MSWEP`, 37
 `IHEWAccollect.templates.NASA`, 37
 `IHEWAccollect.templates.NOAA`, 37
 `IHEWAccollect.templates.USGS`, 38
 `IHEWAccollect.templates.util`, 39
 `IHEWAccollect.templates.VITO`, 38

P

`parse_args()` (in module `IHEWAccollect.scripts.skeleton`), 35
`Plot` (class in `IHEWAccollect.templates.util`), 40

R

`run()` (in module `IHEWAccollect.scripts.skeleton`), 35

S

`set_status()` (`IHEWAccollect.base.user.User` method), 34
`set_status()` (`IHEWAccollect.templates.dtime.Dtime` method), 38
`setup_logging()` (in module `IHEWAccollect.scripts.skeleton`), 35
`status` (`IHEWAccollect.base.base.Base` attribute), 31

`status` (`IHEWAccollect.base.user.User` attribute), 34
`status` (`IHEWAccollect.Download` attribute), 42
`status` (`IHEWAccollect.download.Download` attribute), 41
`status` (`IHEWAccollect.templates.dtime.Dtime` attribute), 38

T

`tar()` (`IHEWAccollect.templates.util.Extract` method), 39

U

`User` (class in `IHEWAccollect.base.user`), 33

W

`wait_bar()` (`IHEWAccollect.templates.util.Waitbar` static method), 40
`Waitbar` (class in `IHEWAccollect.templates.util`), 40
`write()` (`IHEWAccollect.templates.util.Log` method), 40

Z

`zip()` (`IHEWAccollect.templates.util.Extract` method), 39