CDSupdate: A meta-interface for ERA5 download request, management and storage

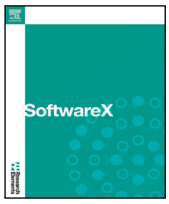
Andreia N S Hisi, Yoann Robin, Davide Faranda, Mathieu Vrac

SoftwareX 28 (2024) 101965

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx



Original software publication

CDSupdate: A meta-interface for ERA5 download request, management and storage

Andreia N.S. Hisi [a,*], Yoann Robin [a], Davide Faranda [a,b,c], Mathieu Vrac [a]

[a] Laboratoire des Sciences du Climat et de l'Environnement, UMR 8212 CEA-CNRS-UVSQ, Université Paris-Saclay/IPSL, Sorbonne Université, Gif-sur-Yvette 91191, France
[b] London Mathematical Laboratory, 8 Margravine Gardens London, W6 8RH London, United Kingdom
[c] Laboratoire de Météorologie Dynamique, École Normale Supérieure, Université PSL, Sorbonne Université, École Polytechnique, IP Paris, CNRS/IPSL, Paris 75005, France

ARTICLE INFO

Keywords:
Tailored data download and storage
Python
Data science

ABSTRACT

CDSupdate is a Python package that automates the process of retrieving, processing, and managing climate data from the Copernicus Climate Change Service (C3S) Climate Data Store (CDS). The tool generates daily climate data summaries, performs calculations to create custom variables such as *relative humidity* and *heat index* which serve as risk assessments, and organizes the data into a user-friendly format. By simplifying data retrieval and performing on-the-fly calculations, it saves users valuable time and effort, enabling more focus on data analysis and interpretation.

Code metadata

Current code version	1.4.0
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-24-00112
Permanent link to Reproducible Capsule	
Legal Code License	GNU-GPL3
Code versioning system used	git
Software code languages, tools, and services used	Python, Copernicus Climate Change Services (C3S) Climate Data Store (CDS)
Compilation requirements, operating environments & dependencies	numpy, pandas, xarray, netCDF4, cftime, cdsapi, windows-curses
If available Link to developer documentation/manual	https://github.com/yrobink/CDSupdate
Support email for questions	yoann.robin.k@gmail.com

Motivation

Understanding climate change and studying the intricacies of weather conditions are fundamental aspects of contemporary scientific research. In the pursuit of unraveling the complexities of our climate system, access to accurate and up-to-date climate data is of paramount importance. Efforts to automate the retrieval of climate data have seen significant advancements in recent years. Prominent among these are the NOAA (The National Oceanic and Atmospheric Administration) [1], NASA (The National Aeronautics and Space Administration) [2], ESA (The European Space Agency) [3], and Copernicus products.

This paper introduces a meta-interface that uses the ECMWF (European Centre for Medium-Range Weather Forecasts) Reanalysis v5 (ERA5) datasets [4] and simplifies access to Copernicus Climate Change Service (C3S) Climate Data Store (CDS) [5] to provide researchers with real-time access to comprehensive weather data while offering flexibility for data management. Our meta-interface, named "CDSupdate" [6], enables researchers to access the most recent ECMWF climate data, ensuring that their analyses and studies are based on the latest information available. This approach enhances the accuracy and relevance of climate research, empowering scientists to diagnose climate change features promptly and study climate variables in an interactive manner.

We focus on outputs produced by the Copernicus program, an initiative of the European Union and the European Space Agency (ESA), aiming to provide accurate, up-to-date, and easily accessible climate information to support policymakers, researchers, and the public. As

* Corresponding author.
E-mail address: andreia.hisi@lsce.ipsl.fr (Andreia N.S. Hisi).

https://doi.org/10.1016/j.softx.2024.101965
Received 20 February 2024; Received in revised form 17 September 2024; Accepted 30 October 2024

part of this program, the C3S CDS acts as a central hub for a wide range of climate-related data, including the ERA5 dataset. By integrating the ERA5 data into the C3S CDS, researchers gain access to a comprehensive repository that enables near real-time retrieval and management of climate information dating back to 1940. This integration not only benefits the scientific community but also facilitates the timely dissemination of climate data to the public.

Existing tools like "era5cli" have made significant strides in simplifying the retrieval of ERA5 climate data. However, unlike "era5cli", we propose here to restore the data to CF (Climate and Forecast) format, aligning variable names and units with CMIP (Coupled Model Intercomparison Project) standards. This alignment facilitates the work of modelers using both sources of information simultaneously, as in impact studies. Additionally, we propose to automatically calculate certain variables not defined in ERA5, further enhancing the utility of the dataset for comprehensive climate analysis.

Another challenge is the need for efficient data processing in high-performance computing environments. While existing tools often provide web-based interfaces, CDSupdate offers a command line interface, which can be used particularly on computing clusters, allowing for streamlined and automated workflows.

Furthermore, our meta-interface distinguishes itself by offering a flexible and user-centric approach to data management. Researchers can define custom parameters, specify temporal and spatial resolutions, and extract subsets of data tailored to their specific research objectives. This flexibility can assist in detailed analyses, helping researchers to better explore the vast climate data more effectively.

The structure of this document is as follows: Section 1 introduces the CDSupdate software, its architecture and functionalities. In Section 2 we present practical examples with an emphasis on its simplicity of use. Section 3 describes the impact and applicability of the software. Finally, we present the overall conclusions in Section 4.

1. Software description

Currently, CDSupdate exclusively supports ERA5 datasets, which represent a significant portion of the C3S CDS's extensive offerings.

CDSupdate has been designed in Python, mainly to:

- **(i)** use the C3S CDS;
- **(ii)** take into account large, robust and open-source Python libraries and community;
- **(iii)** ensure suitability for easy deployment on different systems, enabling widespread use and portability; and
- **(iv)** give access to early-stage developers and users (researchers, students and the general public).

The only requirements for the package are to have the C3S CDS key and the C3S CDS client installed, including the necessary Python modules: *numpy*, *xarray*, *scipy*, *netCDF4*, *cftime* and *cdsapi* (and *curses* in particular for Microsoft Windows users).

The CDSupdate is easy to use and runs through a command line, which can be used in particular on clusters, making it highly suitable for high-performance computing environments. Unlike the CDSToolbox provided by the C3S CDS, which offers a web-based interface for data access and processing, CDSupdate provides a scriptable solution. This allows users to integrate data retrieval and processing into automated workflows and to run batch jobs efficiently on computing clusters. We will illustrate its basic features through simple tasks in Section 2.

The architecture of CDSupdate is designed to ensure seamless and efficient management of climate data, leveraging the capabilities of the C3S CDS. This architecture is modular and data-oriented, allowing for flexibility and scalability in processing and analyzing climate data. Each module within this architecture plays a critical role in the overall functionality of the package, addressing specific aspects of data handling, from retrieval to processing and to storage. The following sections detail the key modules of CDSupdate, highlighting their individual contributions to the system's robustness and efficiency. The modular structure is visually represented in the flowchart Fig. 1, which delineates the interactions and responsibilities of each component within the system. As these two classes, CVarsParams and CDSUParams, are independent, they are shown at the same level in the flow. Even though in the code one is initialized before the other, their execution could be reversed without impacting the functionality. In practice, CVarsParams is initialized before CDSUParams, but this order does not affect the overall process.

In the following sub-sessions, we will provide a simplified guide to command shortcuts: a comprehensive overview that includes a description of the components of the architecture.

1.1. Command line arguments

This tool accepts special shortcuts for various tasks, detailed as follows:

- `--log` Activates logging. Optionally, the level of detail for the logs (defaulting to warnings, but adjustable to errors) and the file location for saving these logs can be specified.
- `--period` Specifies the time frame for data retrieval or update, using the ISO 8601 format YYYY-MM-DD (year-month-day, for example, 2023-01-01). Entering a single date downloads data for that specific day. Adding a slash after the date (e.g., 2023-01-01/) indicates the range from the specified date to today, ensuring all data up to the current available date is included. For the ERA5 dataset, all dates and times are in UTC, meaning that the specified time frame is interpreted in Coordinated Universal Time.
- `--cvar` Specifies which variables to download or update, listed explicitly.
- `--area` Identifies the target area, either as a specific grid or through a keyword (described Table 2).
- `--keep-hourly` Option to retain data updated on an hourly basis.
- `--output-dir` Sets the directory for storing output files.
- `--tmp` Designates a temporary directory for data download and initial processing.
- `--help` Displays a guide on using these command shortcuts.

1.2. Short internal description

At CDSupdate startup, before reading user arguments, the `CVarsParams` class is initialized as an interface to files describing ERA5 variables. The "ERA5-name.csv" file contains the list of supported variables, providing the information described in Table 1. The descriptions given in the ERA5 documentation are provided in files with the format "ERA5-var-description.txt". Finally, the list of pre-built regions is initialized from the "areas.csv" file, see the list in the Table 2. If these files are modified (by adding a new region or variable), CDSupdate will automatically take them into account.

Subsequently, the user's inputs are processed by the `CDSUParams` class, which handles the following tasks:

- Verifies the validity of user inputs, triggering exceptions for any discrepancies;
- Constructs the list of variables to be downloaded and those to be calculated, such as determining relative humidity from temperature and dew-point temperature data;
- Generates the list of requests to be submitted to C3S CDS for data retrieval.

The following sequence of functions then enables ERA5 data to be retrieved and stored:

1. Data are first downloaded in its original format to a temporary directory;

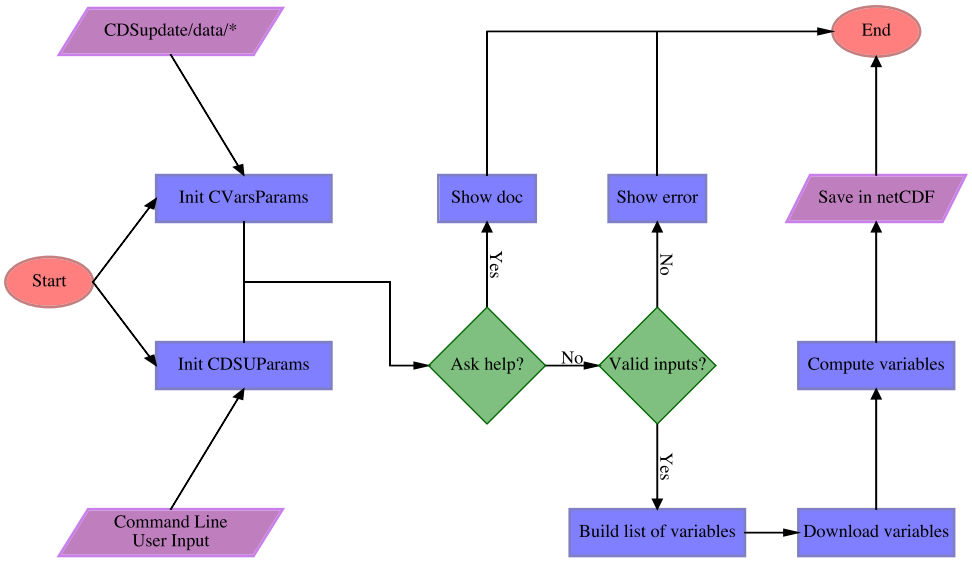


Fig. 1. Architecture of the CDSupdate Package. This flowchart illustrates the sequence of operations in the CDSupdate package.

Table 1
Description of the file "ERA5-name.csv".

Name	Description	Example: tas	Example: hurs
level	Surface or pressure Level	single	single
height	height	2	2
dep	Required variables		tas;dptas
AMIP	AMIP name	`tas`	`hurs`
C3S CDS	C3S CDS name	`2m_temperature`	
ERA5	ERA5 name	`t2m`	
standard_name	AMIP standard name	`air_temperature`	`relative_humidity`
long_name	AMIP long name	Mean Near-Surface Air Temperature	Near-Surface Relative Humidity
units	Unit	K	%
comment	Any comment added in attribute		

Table 2
CDSupdate predefined regions'.

Region	Longitude start	Longitude end	Latitude start	Latitude end
World	−180	180	−90	90
Europe	−25	40	34	72
NorthAtlantic	−80	50	5	72
NorthAmerica	−150	−60	30	80

2. They are then transformed to AMIP format (variable names and unit change if necessary), longitudes are change to the $[-180, 180]$ interval;
3. Additional variables are constructed;
4. Ultimately, the processed data are saved in the netCDF format, following the CF convention [7], and merged with pre-existing data if necessary.

In particular, while the CF-1.11 netCDF convention [8] does not include the "`creation_date`" attribute (but "`date_create`"), the CMIP (Coupled Model Intercomparison Project) data convention does use it. Therefore, we retain "`creation_date`" as an attribute in our data and have updated its format to align with ISO 8601:2004 (YYYY-MM-DDThh:mm) for better compatibility with ACDD-1.3 standards. This approach ensures that our data remains compatible with both CF and CMIP conventions, facilitating its use across a broader range of applications where model data needs to be compared with historical data.

Moreover, in the context of handling pressure level variables, these are simplified by removing the "level" coordinate. For example, the original geopotential height file "zg", which includes data for several pressure levels ranging from 1000 hPa to 1 hPa, is split into multiple files, each representing a fixed pressure level. These resulting files are named accordingly, such as zg1000, zg500, and zg1. For each of these files, a variable named "height" is included, indicating the height corresponding to the specific pressure level (as authorized by the CF convention, see Sec. 4.3.1 of [8]).

Currently, the following additional variables are supported:

- Daily minimum and maximum of any hourly variable, simply by adding the suffix "min" or "max", typical variables being daily minimum and maximum temperature;
- Wind speed, determined by the norm of the u and v components, represented respectively by `uas` and `vas`. The `uas` variable measures the wind speed along the east–west axis, while `vas` represents the v-component for the north-south axis. Together, these components are used to calculate the `sfcWind`, which is the overall wind speed;
- Relative and specific humidity, where it is computed from temperature and dewpoint temperature as the ratio between the vapor pressure and the saturated vapor pressure, using the *Tetens* equation. Specific humidity is computed with vapor pressure and surface pressure. These equations can be found in Table 4.2 of [9].

- Heat Index, using the NOAA equation [10].

Adding a variable managed by ERA5 to CDSupdate is straightforward: simply modify the CDSupdate/data/ERA-name.csv file. However, incorporating additional variables requires changes to the source code itself.

1.3. Data storage

Here we will outline the systematic approach employed for organizing and preserving data in the netCDF format and explain the rationale behind our structured storage methodology.

The storage structure follows a specific hierarchical tree format:
`output_dir/ERA5/'area'/'freq'/'var'/`
Each file is named following the pattern
`ERA5_'var'_'freq'_'area'_YYYY0101-YYYY1231.nc`.
This structure is designed for easy organization, separating data by area (as specified in `'area'`), frequency (`'freq'`, with options like `'day'` or `'hr'`), and variable type (`'var'`). Additionally, to manage file sizes, data is segmented and stored on an annual basis.

2. Illustrative examples

To illustrate the flexibility of CDSupdate we performed three distinct requests:

1. for a *period* which is literally the download request for a well-defined time period. It can be used for fast assignments (short period — assuming there is a waiting time for each request) or for long period (e.g. all available time, 1940-today);
2. for an *update*, defined here as the procedure for updating (or creating) a file containing the latest available meteorological data; and
3. for a specific *region*, this feature enables precise data downloads for targeted areas.

In any case the entire procedure requires one or a sequence of download requests and an optional output customization, such as the change of units, frequency and conventions.

2.1. Example 1: Retrieving data for a specific period

We propose, as our first example, to analyze the wind speed (`sfcWind`) during the Lothar and Martin extra-tropical storm [11], which struck France between the 26th and 28th of December, 1999, causing significant damage. For a comprehensive analysis of these storms, we focus on downloading wind vector components over the North Atlantic. Since the direct `sfcWind` variable is not available in ERA5, we deduce it from these `uas` and `vas` variables, providing a direct view of the wind patterns during the storm. The command line used is the following:

```
cdsupdate --log info test.log
          --period 1999-12-16/1999-12-28
          --cvar sfcWind
          --area NorthAtlantic
          --keep-hourly
          --output-dir <odir>
```

In this command, `--log info test.log` captures detailed logs of the operation in the file `test.log`, the argument `--period 2019-11-09/2022-01-17` specifies the date range, `--cvar sfcWind` selects the `sfcWind` (resulting in the recovery of `uas` and `vas` variables), `--area NorthAtlantic` defines the geographical scope, and `--odir <odir>` determines the output directory. From the output of CDSupdate, we compute the hourly maxima over the period and represent in the Figs. 2a/c. This example highlights the package's precision in handling temporally and geographically specific data queries.

2.2. Example 2: Updating datasets with latest data

To update an existing climate data repository with the latest available information without specifying a time frame, the command is structured as follows:

```
cdsupdate --log info test.log
          --period 2023-12-01/
          --cvar sfcWind
          --area NorthAtlantic
          --keep-hourly
          --output-dir <odir>
```

Using `--period` now enables the tool to automatically retrieve the most recent data, showcasing its capability for ongoing data updates. Here, executing this command on 15/01/2024 updated the `uas`, `vas` and `sfcWind` variables from 01/12/2023 to 09/01/2024 (latest available date on 15/01/2024). It simplifies the process of keeping climate datasets current, demonstrating the tool's utility in ongoing research projects where up-to-date data is important.

2.3. Example 3: Custom data retrieval for a specific region

To download data on the maximum Heat Index for a specific area in India during April 2023, the command is:

```
cdsupdate --log test.log
          --period 2023-04-01/2023-04-30
          --cvar HImax
          --area India,67,98,6,36
          --output-dir <odir>
```

Through this command, CDSupdate is directed to download the dew-point temperature and to calculate the relative humidity and the *Heat Index* (a measure that combines air temperature and relative humidity to determine heat-related risk assessments) [10] for the entire India region Fig. 2d/f, within the defined period, storing it in the desired directory. This example showcases the package's ability to cater to highly specific and localized data needs, which is essential for targeted climate studies.

These examples underscore the adaptability of CDSupdate in various data retrieval scenarios, catering to the specific needs of climate research. The package simplifies the process of accessing and managing diverse sets of climate data.

3. Impact

The CDSupdate python package, released in the PyPi repository, offers notable contributions to climate data analysis tools. Since its prototype development in December 2021, it has been instrumental in studies like the retrospective analysis of weather extremes in 2021 [12] and the investigation of jet stream variability [13]. These applications highlight the tool's capacity for handling complex climate data sets.

CDSupdate has evolved beyond a simple data retrieval system to a comprehensive tool for data analysis. It supports functionalities such as custom index creation and direct data utilization, which simplifies the process for end-users. They can now engage with these indices for their analyses without the need for separate calculations.

The software has gained attention from research laboratories and is becoming increasingly recognized for accessing ERA5 data in routine research. Its user-friendly interface, combined with robust data processing capabilities, establishes CDSupdate as a valuable tool in climate research and decision-making processes. It serves as a gateway to understanding and efficiently managing large climate data sets.

By streamlining data handling and analysis, CDSupdate facilitates more focused research in climate studies. It empowers researchers to delve deeper into climate patterns and contribute to informed discussions on climate-related policy and public awareness. The tool's ongoing development promises to further enhance its role in climate research and its influence on both scientific and public engagement with climate change.

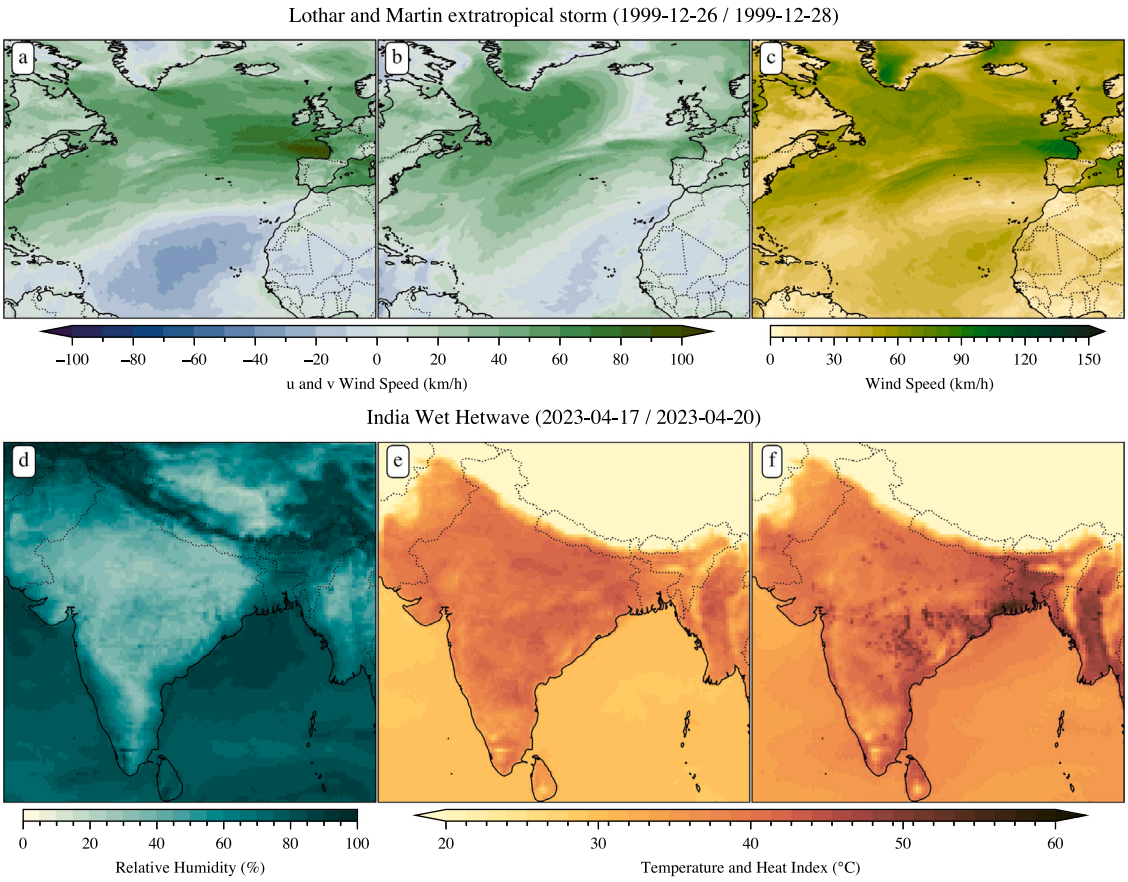


Fig. 2. a, b, c: Lothar and Martin extratropical storm (December 1999, 26 to 28). **a.** Maximal hourly Wind Speed in the u direction. **b.** Maximal hourly Wind Speed in the v direction. **c.** Maximal hourly Wind Speed. **d, e, f**: India Extreme Wet Heatwave (April 2023, 17 to 20). **d.** Maximal hourly Relative Humidity. **e.** Maximal hourly Temperatures. **f.** Maximal Heat Index.

4. Conclusions

CDSupdate provides a modular and adaptable foundation for efficient climate data retrieval, processing, and storage. Its architecture is crafted around procedural programming and data validation for robust and dependable operation.

The tool's parameter-driven design promotes flexibility and customizability, ensuring it can adapt to diverse use cases and research objectives. This feature is coupled with a strategic integration of the Climate Data Store API, enhancing the tool's ability to provide accurate and comprehensive climate data.

By encapsulating intricate data processing procedures into an accessible command-line interface, the tool democratizes access to climate data, empowering researchers, data scientists, and climate enthusiasts alike. As the demand for climate data continues to grow and tools like the CDSupdate have the potential to play an essential role in fueling meaningful analyses and risk assessments. This software represents an important step toward streamlining climate research and fostering a deeper understanding of our planet's climate patterns.

Adding extra variables, like *Wind Speed*, *Relative Humidity* and *Heat Index*, extends the functionality of the CDSupdate by providing additional derived datasets that can serve as meaningful indices or metrics. These new variables are created based on calculations using the original climate data, and they offer users a way to directly access specific information that might otherwise require extra computation or analysis.

Such features transform CDSupdate from a simple data retrieval system into a more comprehensive data analysis tool, enriching its utility and adaptability across a wider range of use-cases and user needs. This enhances the tool's ability to facilitate climate research and data-driven decision-making processes, making it a versatile asset in the field of climate studies.

CRediT authorship contribution statement

Andreia N.S. Hisi: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Conceptualization. **Yoann Robin:** Writing – review & editing, Visualization, Validation, Software. **Davide Faranda:** Writing – review & editing, Supervision, Conceptualization. **Mathieu Vrac:** Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors wish to extend their gratitude to the "Internal and Forced Climate Variability" pole at Institut Pierre Simon Laplace (IPSL) and Sorbonne Université, France for their financial support of this work.

Data availability

No data was used for the research described in the article.

References

[1] National Oceanic and Atmospheric Administration Accessed:29.01.2024.

[2] The National Aeronautics and Space Administration Accessed:29.01.2024.

[3] The European Space Agency Accessed:29.01.2024.

[4] Hersbach H, Bell B, Berrisford P, Hirahara S, Horányi A, Muñoz-Sabater J, et al. Thépaut J-N the ERA5 global reanalysis. Q J R Meteorol Soc 2020;146(730):1999–2049. http://dx.doi.org/10.1002/qj.3803.

[5] ECMWF Climate Data Store API accessed:30.05.2023.

[6] Robin Y, Hisi A. Meta-Interface for Climate Data Store https://pypi.org/project/CDSupdate/, http://dx.doi.org/10.5281/zenodo.10548135.

[7] Eaton B, Gregory J, et al. NetCDF Climate and Forecast (CF) Metadata Conventions, CF-1.11, 2023, Accessed:26.01.2024.

[8] CF-1.11 netCDF convention, Appendix A Accessed:05.07.2024.

[9] Stull Ronald B. Practical Meteorology: An Algebra-based Survey of Atmospheric Science University of British Columbia Vancouver, Canada ISBN: 978-0-88865-283-6.

[10] The Heat Index Equation Accessed:26.01.2024.

[11] Ulbrich U, Fink AH, Klawa M, Pinto JG. Three extreme storms over europe in 1999. Weather 2001;56(3):70–80. http://dx.doi.org/10.1002/j.1477-8696.2001.tb06540.x, https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/j.1477-8696.2001.tb06540.x, https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/j.1477-8696.2001.tb06540.x.

[12] Faranda D, Bourdin S, Ginesta M, Krouma M, Noyelle R, Pons F, et al. A climate-change attribution retrospective of some impactful weather extremes of 2021. Weather Climate Dyn 2022;3(4):1311–40. http://dx.doi.org/10.5194/wcd-3-1311-2022.

[13] Noyelle R, Guette V, Viennet A, Colnet B, Faranda D, Hisi ANS, et al. Decrease of the spatial variability and local dimension of the euro-atlantic eddy-driven jet stream with global warming. Clim Dyn 2023. http://dx.doi.org/10.1007/s00382-023-07022-z.