

Facilitating radar precipitation data processing, assessment and analysis: a GIS-compatible python approach

Jennifer Kreklow

ABSTRACT

A review of existing tools for radar data processing revealed a lack of open source software for automated processing, assessment and analysis of weather radar composites. The ArcGIS-compatible Python package *radproc* attempts to reduce this gap. *Radproc* provides an automated raw data processing workflow for nationwide, freely available German weather radar climatology (RADKLIM) and operational (RADOLAN) composite products. Raw data are converted into a uniform HDF5 file structure used by *radproc*'s analysis and data quality assessment functions. This enables transferability of the developed analysis and export functionality to other gridded or point-scale precipitation data. Thus, *radproc* can be extended by additional import routines to support any other German or non-German precipitation dataset. Analysis methods include temporal aggregations, detection of heavy rainfall and an automated processing of rain gauge point data into the same HDF5 format for comparison to gridded radar data. A set of functions for data exchange with ArcGIS allows for visualisation and further geospatial analysis. The application on a 17-year time series of hourly RADKLIM data showed that *radproc* greatly facilitates radar data processing and analysis by avoiding manual programming work and helps to lower the barrier for non-specialists to work with these novel radar climatology datasets.

Key words | heavy rainfall, open source software, radar climatology, RADKLIM, *radproc*, weather radar

Jennifer Kreklow

Leibniz Universität Hannover, Institute of Physical Geography and Landscape Ecology, Schneiderberg 50, 30167 Hannover, Germany
E-mail: kreklow@phygeo.uni-hannover.de

ABBREVIATIONS

API	Application Programming Interface	DWD	Deutscher Wetterdienst (German Weather Service)
BUFR	Binary Universal Form for the Representation of meteorological data	EUMETNET	European Meteorological Services Network
CDC	Climate Data Center	GIS	Geographic Information System
DWA	Deutsche Vereinigung für Wasserwirtschaft, Abwasser und Abfall e.V.	GRIB2	General Regularly-distributed Information in Binary form version 2
		GUI	Graphical User Interface
		HDF5	Hierarchical Data Format version 5
		IDE	Integrated Development Environment
		IDL	Interactive Data Language

This is an Open Access article distributed under the terms of the Creative Commons Attribution Licence (CC BY 4.0), which permits copying, adaptation and redistribution, provided the original work is properly cited (<http://creativecommons.org/licenses/by/4.0/>).

doi: 10.2166/hydro.2019.048

NEXRAD	Next-Generation Radar
ODIM	OPERA Data Information Model
OPERA	Operational Program on the Exchange of Weather Radar Information
OSS	Open Source Software
RADKLIM	Radarklimatologie ('Radar Climatology')
RADOLAN	Radar-Online-Aneichung ('Radar Online Adjustment')
WSR-88D	Weather Surveillance Radar – 1988 Doppler

INTRODUCTION

Rainfall and especially heavy and extreme rainfall events are a major trigger for floods and flash floods (Gaume *et al.* 2009; Bouilloud *et al.* 2010; Alfieri *et al.* 2011; Wright *et al.* 2017), soil erosion (Wischmeier & Smith 1978; Panagos *et al.* 2015, 2017; Steinhoff-Knopp & Burkhard 2018), mud flows (Hänsel *et al.* 2018) and landslides (Guzzetti *et al.* 2007; Segoni *et al.* 2014) causing costly damage or even casualties. As the frequency and intensity of heavy rainfall events are likely to increase (IPCC 2013; Quirnbach *et al.* 2013; Panagos *et al.* 2017; Thorndahl *et al.* 2017) and seasonal and spatial distribution of rainfall is shifting due to climate change (Zolina *et al.* 2008; Panagos *et al.* 2017), there are growing needs for adaption and risk prevention measures (Alfieri *et al.* 2012; Winterrath *et al.* 2017).

The high spatial and temporal variability of rainfall (Ramos *et al.* 2005; Fischer *et al.* 2016) dictates that high resolution precipitation data are needed (Thorndahl *et al.* 2017; Winterrath *et al.* 2017). Weather radar observations can help to satisfy this demand and particularly improve severe weather detection and quantification of precipitation during storm events (Krajewski & Smith 2002; Heistermann *et al.* 2013; Wright *et al.* 2013). Moreover, technical developments in hardware and software engineering as well as an increasing availability of data, some of which are available even free of charge, allows a wider audience to apply radar products.

Working with radar data, however, presents a string of challenges which make many potential users still reluctant to take advantage of these data. Weather radar is an indirect measurement method suffering from numerous potential

error sources and uncertainties in terms of precipitation quantification (Krajewski & Smith 2002; Gjertsen *et al.* 2003; Raghavan 2003; Meischner 2004; Sene 2010; Seo *et al.* 2011). Consequently, the use of radar-based precipitation estimates necessitates additional effort for data quality assessment and probably further corrections. Yet, many national weather services recently put much effort into re-analyses of radar data time series applying state-of-the-art bias correction and adjustment algorithms. Keupp *et al.* (2017) give an overview of current reanalysis activities in Europe aimed at the establishment of radar climatologies. These projects, such as the radar climatology RADKLIM (Winterrath *et al.* 2018a, 2018b) provided by the German Weather Service (Deutscher Wetterdienst, (DWD)) will open up new climatological application fields for radar data (Keupp *et al.* 2017; Winterrath *et al.* 2017) which include the characterisation of the spatial variability of long-term rainfall patterns, seasonal variations in rainfall, durations of dry periods and the study of rainfall extremes and their impacts (Overeem *et al.* 2009; Smith *et al.* 2012; Wright *et al.* 2014). Most likely, the derivation of these radar climatology datasets will also lead to an overall enhancement of data quality and a reduced necessity for individual bias corrections.

Beyond the outlined uncertainties regarding data quality, several technical barriers exist that can prevent potential users from working with radar data. These include different file formats for exchange and storage, provision in proprietary binary file formats, a scarcity of easy-to-use and free-of-charge processing software, spatial visualisation and clipping tools, missing compatibility or interfaces to Geographic Information Systems (GIS) and the vast amount of data (Heistermann *et al.* 2013, 2015; Fischer *et al.* 2016). As a consequence, the processing of radar data not only requires considerable expertise in data handling and programming, but it also takes much time to develop user-customised workflows, which discourages many potential users.

Despite initiatives such as the OPERA (Operational Program on the Exchange of Weather Radar Information) weather radar information model (Michelson *et al.* 2014), which has been widely adopted for international data exchange within Europe (Heistermann *et al.* 2015), many national weather authorities still provide weather radar

composites in their own custom formats. This also holds true for the German weather radar data products RADKLIM and its counterpart for operational applications called RADOLAN ('Radar Online Adjustment'). Consequently, software adaption and interfacing to support the processing of different data formats is still necessary on a national scale, and available tools for automated, GIS-compatible radar data processing, e.g., for American (Zhang & Srinivasan 2010) or Norwegian (Abdella & Alfredsen 2010) data, cannot be applied for the German weather radar products.

This paper presents the open source library *radproc* (Kreklow 2018) written in Python (van Rossum 2001–2019) as a possible solution to the bottlenecks in current weather radar data processing and assessment in Germany and beyond. *Radproc* intends to lower the barrier to radar data usage by automating radar and rain gauge data processing and providing an interface for data exchange with GIS. First, an overview of the data basis and existing tools for the processing of German weather radar composites is provided in order to illustrate the motivation and need for developing *radproc*. Moreover, a short outlook on software for the processing of radar data in other countries is given. Next, the development goals, the implementation of the technical framework and its potential for transferability to other precipitation data are presented. Afterwards, *radproc*'s functional scope is demonstrated based on a typical workflow including raw data processing, temporal aggregation, heavy rainfall detection and data export to ArcGIS. Finally, limitations are discussed, a perspective of future developments and improvements is given and conclusions are drawn.

MOTIVATION FOR THE DEVELOPMENT OF RADPROC

Data basis

The DWD operates a network of 17 ground-based C-band Doppler radar stations and, in 2005, launched the operational application of the RADOLAN programme to provide near real-time nationwide quantitative precipitation estimations on a 1 km² raster in temporal resolutions of 5 and 60 minutes (Winterrath *et al.* 2017). The hourly radar

rainfall composites are adjusted to precipitation measurements from a network of approximately 1,300 rain gauges (Bartels *et al.* 2004; Keupp *et al.* 2017).

In 2017, the DWD concluded its 'radar climatology' project, in which all available weather radar data have been re-analysed back to the year 2001 applying state-of-the-art bias correction and adjustment algorithms (Winterrath *et al.* 2017). The resulting dataset called RADKLIM initially offers a largely homogeneous, spatially and temporally highly resolved radar-based precipitation time series of 17 years for Germany. The final datasets of RADKLIM are called RW (60 minute resolution) and YW (5 minute resolution). The DWD intends to update the dataset annually to further extend the time series. Due to a law change in July 2017 (Deutscher Bundestag 2017), radar data are subject to an open access policy, which is why both RADKLIM products are provided free of charge in the DWD Climate Data Centre (CDC). This makes it a very interesting and promising dataset for various applications, for instance in hydrology, meteorology and geography.

Review of available software tools for RADOLAN and RADKLIM radar data processing

In recent years, a variety of processing tools supporting German weather radar data have been developed that contain different functions and target different user groups. The following review is structured according to the software distribution model since the availability, costs and customisability of a tool are factors strongly influencing a user's choice of software.

Open source software

According to the open source definition, a software is regarded as open source software (OSS), if its source code is made available and its license grants the rights to use and modify the software to anyone and for any purpose, including non-exclusive commercial exploitation and redistribution of derivative works of the software itself (St Laurent 2008).

Heistermann *et al.* (2015) give a detailed overview of five international, active OSS for radar data processing and analysis. Their review shows that these tools can help a great deal in coping with the import and management of

different file formats and can promote further research on data quality improvements through continuous community-based development. But all of these tools are mainly developed for and used by specialists since they require a significant technical understanding of data formats and radar data processing techniques as well as programming skills for application and the development and automation of data processing workflows. Thus, they are not primarily targeted at users outside the weather radar community, such as engineering offices, authorities or researchers and users from other water-related fields of application. Most of these are rather interested in the application, analysis and visualisation of quantitative precipitation estimations provided by the national meteorological authorities.

The Python library *wradlib* (Heistermann *et al.* 2013) is the only one of the reviewed tools, which supports Windows operating systems and the binary format of RADOLAN and, since the version 1.2 is available, also RADKLIM products. Moreover, it has an active and growing user community and a website with extensive documentation. Besides many functions dedicated to the typical tasks of weather radar raw data processing outlined above, *wradlib* provides some visualisation tools and a function for reading in a single binary RADOLAN or RADKLIM file into a *NumPy* array. *NumPy* (Oliphant 2006) is a widely used package for scientific computing with Python and provides an array object for efficient numerical computations. Nevertheless, the processing workflows beyond single file import as well as the data structure for storage and further analysis have to be developed and programmed by the user. For users with little programming skill, this is a difficult, time-consuming and most likely discouraging task. Furthermore, many users working with spatial data use GIS. *Wradlib* provides functions to export radar data as GeoTIFFs or ESRI ASCII files as well as a series of functions for georeferencing and reprojection, but it does not have any direct interface to any GIS nor does it support clipping the nationwide composites. However, the latter is an important feature to reduce the amount of data and to limit analyses to a desired study area. In addition, GIS users might encounter serious difficulties installing *wradlib* due to separate and possibly incompatible installations of GDAL (Geospatial Data Abstraction Library; <http://www.gdal.org>), which is indispensable for georeferencing.

In Germany, several other OSS have been developed in the last years that are supposed to lower the barrier for working with RADOLAN and providing processing functions, most of them being rather small projects for data conversion, visualisation or for solving some very specific tasks.

- The *radolan* Go library (<https://gitlab.cs.fau.de/since/radolan>) supports the parsing and visualisation of several RADOLAN products but does not provide any analysis functions.
- The *Java Radolan parser* (<http://www.bitplan.com/index.php/Radolan>) is a Java port and extension of the *radolan* Go library for interactive and animated visualisation of different RADOLAN products. Aggregation functions and RADKLIM support are in preparation.
- *Rdwd* (<https://github.com/brry/rdwd>) is an actively maintained R package (R Core Team 2018) to select, download and read DWD climate data into R. It does not support RADOLAN and RADKLIM data yet, but an extension is in preparation.
- The *C++ RADOLAN library* (<https://github.com/meteo-ubonn/radolan>) offers several functions for the import of RADOLAN files and conversion to NetCDF and Shapefile format, but it does not seem to be an actively maintained project anymore.
- The *RADAR* and *ArViRadDB* toolkit (<https://www.hs-rm.de>) is a collection of freely available compiled routines supporting the hourly RADOLAN RW product and targeted at some very specific needs in hydrological engineering as well as the detection of heavy rainfall intervals. *RADAR* yields GIS-compatible ESRI ASCII files as outputs, but it does not provide a real GIS integration that would allow data clipping.
- *IDL RaBiD* (ftp://ftp-cdc.dwd.de/pub/CDC/grids_germany/hourly/radolan/idlrabid/) is a software for RADOLAN visualisation last updated in 2011. It is freely available in the DWD Climate Data Center, but it requires an IDL (Interactive Data Language) license or Virtual Machine to run.

Of all reviewed OSS projects supporting RADOLAN or RADKLIM, *wradlib* offers by far the widest range of functions and the highest quality and quantity of documentation. Moreover, none of the presented tools except

wradlib supports RADKLIM data up to now and most of them do not support the RADOLAN composite product RY in 5 minute resolution.

None of the OSS provides any Graphical User Interfaces (GUI). They are either run from Integrated Development Environments (IDE) or executed in command line windows. Consequently, besides *wradlib*, which is primarily targeted at advanced users from the weather radar community, there is a scarcity of OSS for RADOLAN data processing and a total absence of OSS for the automated processing and analysis of RADKLIM and the temporally highly resolved RADOLAN and RADKLIM products.

Commercial software

To the best of the author's knowledge, there are currently six commercial software products available that support the processing of German radar data, each of them with different target groups and functionalities.

- The ArcGIS extension *NVIS* (<http://www.itwh.de>) is intended for RADOLAN visualisation, precipitation nowcasting, analysis of heavy rainfall events and calibration of sewer system models. *NVIS* offers functions for operational or short-term analysis of hourly RADOLAN RW data and seamless GIS integration, but is neither targeted at long-term climatological analysis nor does it support any radar composites with 5 minute temporal resolution.
- The *HydroNET-SCOUT* toolkit (<http://hydronet-scout.de/>) supports the import and visualisation of several different radar data formats, radar raw data correction, precipitation nowcasting and warnings as well as the export of temporally or spatially aggregated precipitation data.
- *AquaZIS* (<http://www.aquaplan.de>) is a software that supports the analysis and management of a variety of meteorological data. It is primarily targeted at time series and heavy rainfall analysis for water management tasks and provides the possibility to export results into Shapefiles.
- The *KISTERS Meteo* and *HydroMaster* toolkit (<https://water.kisters.de>) is a collection of software products for water management tasks providing a wide range of analysis, nowcasting and visualisation tools for

many meteorological datasets including all presented RADOLAN and RADKLIM products.

- *Delft-FEWS* (<https://publicwiki.deltares.nl/display/FEWSDOC>) is a software for management of time series data and forecasting processes. According to its documentation, it supports several raw and intermediate RADOLAN products, but none of the final composite products like RW and RY.
- *NinJo* (<http://www.ninjo-workstation.com/editions.0.html>) is a basic software for visualisation and processing of meteorological data for operational weather observation and forecasting.
- *MDMS_Expert* (<https://de.dwa.de/de/messdatenmanagement-expert.html>) is a software for import, selection and display of meteorological data provided by DWA (Deutsche Vereinigung für Wasserwirtschaft, Abwasser und Abfall e.V.), which also supports statistical heavy rainfall and correlation analyses.

All commercial products have many different functions accessible via GUI and the engineering offices selling them provide support for installation and application. Nevertheless, specialised commercial software products like these are hardly affordable for many users. Water management authorities undoubtedly prefer these products for operational purposes due to their easy-to-use, reliable and tailor-made functionality. However, for smaller companies with little financial resources, companies striving to develop new technologies and methods and especially in research, the use of OSS could make the radar data more accessible and facilitate the development of new methods. Moreover, as most of the commercial software products seem to be completely based on GUIs, it is not obvious to what extent they allow for customisation and individual analysis of the data.

All reviewed commercial tools support the processing of some or all RADOLAN products, but up to now, none of them explicitly indicate the support of RADKLIM data on their websites or in their release notes.

Many of the presented open source and commercial tools aim to support either data visualisation, retrospective analysis of single rainfall events, operational rainfall nowcasting or raw radar station data processing including the application of correction, merging and gauge adjustment

algorithms, whereas the number of tools supporting long-term climatological analysis of temporally highly resolved radar data is still rather small. Furthermore, the review revealed that there is a considerable gap in regard to the number of provided functions and ease of usability between open source and commercial software products.

Outlook on software tools for other radar data formats

Without claiming to be exhaustive, this section gives an overview of software tools for processing other radar data and of recent developments that should lead to a standardisation of radar data formats. The great variety of different radar data formats that exists internationally impedes data exchange and has led to the development of many software tools, which are only applicable in specific regions or individual countries. However, there are increasing efforts to foster data exchange and cooperation through a standardisation of data formats and the creation of regional radar composites. In Europe, this is coordinated by the European Meteorological Services Network's (EUMETNET) Operational Program on the Exchange of Weather Radar Information (OPERA) which developed the OPERA Data Information Model (ODIM) in order to facilitate data exchange and create a Pan-European radar composite. The OPERA community also provides several software packages to exchange data between other radar data formats and ODIM, which is implemented in HDF5 as well as in BUFR (Binary Universal Form for the Representation of meteorological data) format (Michelson *et al.* 2014; Saltikoff *et al.* 2018, <http://eumetnet.eu/activities/observations-programme/current-activities/opera/>). Moreover, the OSS *BALTRAD* (Henja *et al.* 2010) for radar data exchange and processing, which was developed and is used by several countries in the Baltic Sea region, is based on the ODIM formats. *Wradlib* also provides support for the import of ODIM for HDF5. For Norwegian radar data provided in HDF5, a GIS toolset for automated processing and evaluation has been developed (Abdella & Alfredsen 2010).

For the processing, visualisation and analysis of the American WSR-88D (Weather Surveillance Radar – 1988 Doppler) data, also referred to as NEXRAD (Next-Generation Radar), there are different available software tools. *HEC-MetVUE*, which was developed by the Hydrologic

Engineering Center (McWilliams 2017; Benson *et al.* 2018), and the *Weather and Climate Toolkit* (<https://www.ncdc.noaa.gov/wct/>) developed by the National Oceanic and Atmospheric Administration (NOAA), are software tools provided by national agencies. Furthermore, with *LROSE*, *TITAN*, *Py-ART* and *RSL*, a series of open source tools is available. Along with *BALTRAD*, these tools have been discussed in detail by Heistermann *et al.* (2015). Moreover, a GIS-based software to automatically create a NEXRAD precipitation database has been developed by Xie *et al.* (2005), the GIS software *NEXRAD-VC* allows for validation and calibration of NEXRAD data (Zhang & Srinivasan 2010) and *Hydro-NEXRAD* is a prototype software to provide hydrologists with radar-rainfall maps (Seo *et al.* 2011).

In Southeast Asia, radar data exchange is encouraged in order to create a regional radar composite for disaster risk reduction. This is supposed to be achieved by using the same data format, GRIB2 (General Regularly-distributed Information in Binary form) (Kakihara 2018). This can be used by *SATAID* (<https://www.wis-jma.go.jp/cms/sataid/app.html>), which is a software for daily weather analysis and forecasting widely used by meteorological service providers in Southeast Asia. It has been developed by the Japan Meteorological Agency's Meteorological Satellite Center. In South Korea, the web-based module *WERM-S* has been developed for rainfall erosivity index calculations from radar data provided in ASCII format (Risal *et al.* 2018).

The South African Weather Service uses the commercial *HydroNet* software for its rainfall monitoring and decision support based on radar data (<http://www.weathersa.co.za/product-and-services/2-uncategorised/443-hydronet>) and the application of this software is also endorsed in Australia (<https://www.hydronet.com.au/>). Moreover, NEXRAD, OPERA HDF5, GRIB2 and some other weather radar data formats are also supported by ICMLive (<https://www.innoaqua.de/de/software/article/icmlive-638.html>) for real-time operational hydraulic modelling and early warning applications.

DEVELOPMENT GOALS, IMPLEMENTATION AND TRANSFERABILITY OF RADPROC

Radproc has been developed with the intention to reduce the existing gap outlined in the software review and to

lower the entrance barrier for the usage of RADOLAN and RADKLIM data for GIS users with little or no programming skills. It is an open source tool that provides an automated data processing workflow based on flexible data structures and designed with high extensibility regarding additional functions and interfaces to other input data formats and GIS. Moreover, the hardware requirements and programming skills to use the tool are kept as low as possible, but individual analyses, modifications and the implementation of new precipitation datasets by advanced users are possible and welcome. The development of such a tool requires a balanced trade-off between necessary hardware, calculation speed and required programming skills for application.

As the programming language, Python was chosen as it is an open source language with a high and still increasing popularity for various applications, especially for Data Science, resulting in a large quantity of robust additional packages and a large and active community. Python is implemented in many GIS (e.g., ArcGIS and QGIS) and is

an easy to learn programming language. Moreover, the DataFrame introduced by the *pandas* package (Mckinney 2010) is a very flexible data structure well suited for time series data. Besides some implemented visualisation methods and full compatibility with several plotting libraries such as *matplotlib*, *seaborn* and *bokeh*, *pandas* also offers a direct interface to store DataFrames in HDF5 files (The HDF Group 2018), which allows for a structured and compressible storage of large datasets.

In order to reduce the data amount by clipping to a study area and to allow for geospatial analyses and sophisticated visualisations, a high compatibility to ArcGIS was sought as this software programme is one of the most mature and most widely used GIS in academia, hydrological engineering and public institutions. Since ArcGIS is a commercial software, which contradicts the open source approach, all ArcGIS-based functions were encapsulated in a separate module (see Figure 1). This way, a partial use of *radproc* without clipping and GIS export functions, is

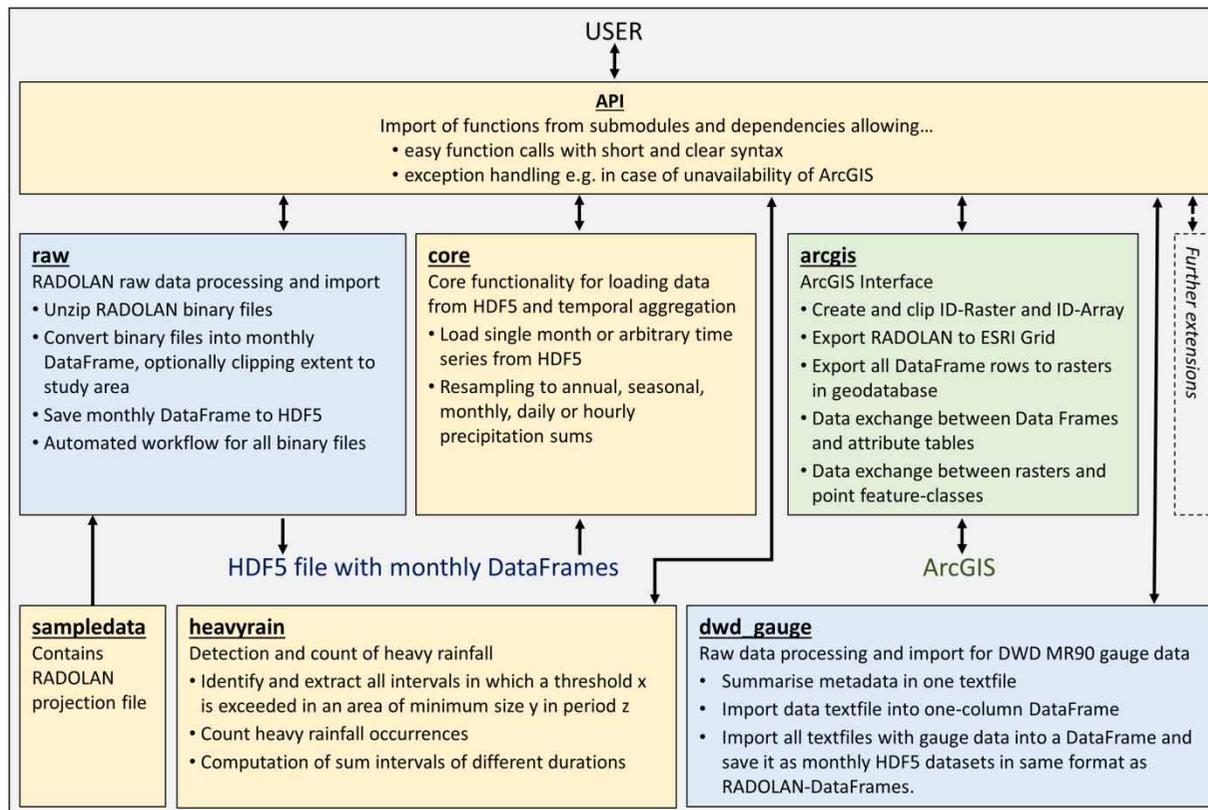


Figure 1 | Module structure and interactions of *radproc*.

Table 1 | *Radproc* development goals and decisions for implementation

Development goals	Implementation
> Open source	✓ Python as programming language
> Maximum compatibility with GIS to reduce data amount by clipping to study area and allow for custom visualisation and geoprocessing	
> Availability of tools for statistical analysis and visualisation to allow for custom analyses beyond <i>radproc</i> 's functions	✓ DataFrames from the <i>pandas</i> Python package as primary data structure
> Availability of a flexible, widely used and well-documented data format with support for statistical analyses and time series which is equally suitable for radar and rain gauge data	
> Structured, compressible data storage format enabling fast data access	✓ Data storage in HDF5 with one group per year and therein monthly <i>pandas</i> DataFrames as datasets
> Storage of processed data in a uniform data format on which all analysis and export functions are built in order to enable extensibility and interfacing to other precipitation data formats	
> Widely used, stable and well-documented GIS as a basis for all spatial analysis and visualisation tasks	✓ Choice of ArcGIS as the most mature and widely used GIS

still possible without ArcGIS and Python-based open source alternatives such as QGIS that can be implemented in the future.

Table 1 shows an overview of the defined development goals and the design decisions and the derived software choices.

In the current version 0.1.4, *radproc* consists of five modules for data processing and analysis as well as an API and a sample-data module (see Figure 1).

- The *raw* module comprises all functions for the automated processing and import of RADOLAN and RADKLIM raw data into HDF5. This includes extracting the binary data from compressed monthly or daily data archives, importing data into monthly DataFrames and saving these into *radproc*'s uniform HDF5 file structure.
- The *dwd_gauge* module offers automated processing and import of rain gauge data with 1 minute resolution provided by the DWD into the same HDF5 file structure as the radar data.
- The *core* module offers a variety of functions to load data from HDF5 and to resample them to annual, seasonal, monthly, daily or hourly precipitation sums. All functions of this module build solely upon the created HDF5 files and are thus independent from the original raw data formats.
- The *heavyrain* module contains functions for the calculation of duration sums as well as for the identification and counting of heavy rainfall events exceeding arbitrary thresholds. As it loads all data from HDF5 files via the *core* module, it is independent of the raw precipitation data formats.
- The *arcgis* module comprises all functions based on the ArcGIS *arcpy* package, e.g., functions for clipping data to a study area and for data exchange between DataFrames and raster datasets or attribute tables.
- The *API* module serves for more convenient function calls and takes care of exception handling, e.g., in case ArcGIS is not available.
- The *sampledata* module contains data for facilitating the use of *radproc* such as the projection file for the stereographic projection defined by DWD for RADOLAN products.

Radproc's fundamental concept constitutes a conversion of all input data into a standardised HDF5 file containing a uniform structure with one group per year and therein monthly Dataframes as datasets. Thus, one HDF5 file contains the entire time series of a precipitation dataset for a defined study area split into monthly portions. The splitting is necessary in order to keep the required working memory to a manageable amount and to enable the processing of temporally highly resolved data for large study areas

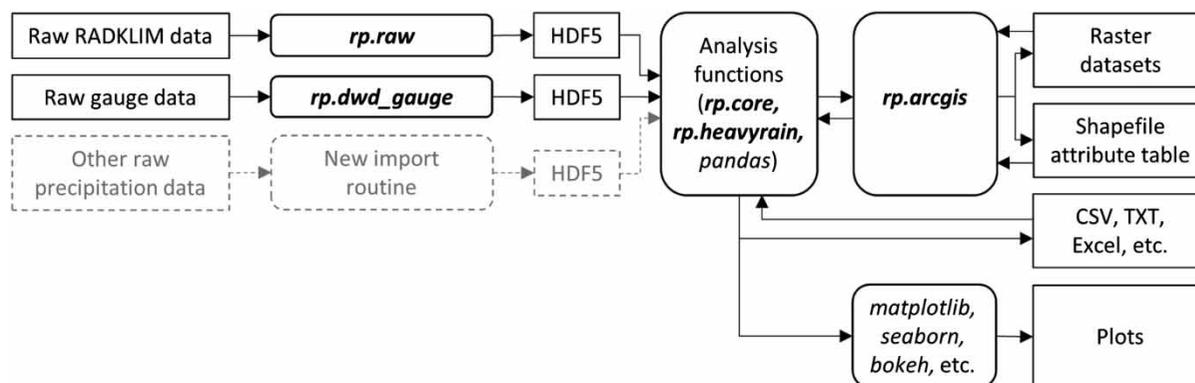


Figure 2 | Extensible data processing workflow with multiple precipitation datasets. Radproc is abbreviated with *rp* and its modules are highlighted in bold.

on average workstation computers. Within these monthly DataFrames, each column corresponds to a spatial location, which can be either a grid cell or a point identifiable by means of a unique ID, whereas each row corresponds to a timestamp. This way, the selection of a DataFrame column yields a time series for a specific cell or point and the selection of a row yields the spatially distributed precipitation at a given time. After the creation of an HDF5 file for a precipitation dataset, the original input data are not required and accessed anymore. From this point onwards, all data, no matter whether gridded or point-scale data, are loaded from HDF5 and analysed with the same functions. Consequently, *radproc*'s entire analysis functionality is independent from the input data formats, which allows for a high extensibility in terms of new input data and analysis methods. As soon as new functions for the automated import of other precipitation datasets into the standardised HDF5 format have been developed, *radproc*'s whole functionality is available for this dataset. Such import functions can either be stand-alone scripts developed by individual users or functions added to a new or existing *radproc* module. In return, if a new analysis method is added, it can be applied on all datasets imported into HDF5. Moreover, since there is one separate HDF5 file for each precipitation dataset and study area, datasets are still independent of each other (see Figure 2). These are fundamental features since *radproc* is intended, among other applications, for data quality assessment, which necessarily involves the intercomparison of different datasets.

The only difference in data processing between gridded and point-scale data can be the export of results into GIS,

because data can either be exported to raster datasets or to new fields in attribute tables. As all environment settings during raster export (e.g., location, spatial reference) are derived from a so-called ID raster required as input parameter (see section on raw data processing below), this function is – as well as all analysis functions – neither limited to Germany nor to the RADOLAN, RADKLIM and gauge datasets currently implemented in *radproc*. Thus, *radproc*'s data processing workflow is transferable to any other precipitation time series dataset, provided that the required individual import routine converts the dataset into monthly *pandas* DataFrames, stores them into the described uniform HDF5 format and creates an ID raster for it in order to clip and export the data.

A TYPICAL DATA PROCESSING WORKFLOW USING RADPROC

In the following, a typical basic radar data processing and analysis workflow including raw data processing, temporal aggregation, heavy rainfall detection and data exchange with ArcGIS using *radproc* and the 17-year time series of the hourly RADKLIM RW product is illustrated and an overview of the most important functions is given. Whereas the RADKLIM and DWD gauge raw data processing is specific for Germany, the analyses and GIS exports shown in the other subsections are equally applicable for any other precipitation dataset imported into *radproc*'s standardised HDF5 file format, introduced above.

RADKLIM raw data processing and clipping

The raw RADOLAN and RADKLIM data are usually provided as gzip compressed monthly tar archives containing one uncompressed binary file per 5- (YW, RY) or 60-minute (RW) time interval. Every binary file starts with a metadata header and then contains 900×900 (RADOLAN) or $1,100 \times 900$ (RADKLIM) gridded precipitation values as integers in 1/10 mm for the whole of Germany, whereby every value describes the spatially averaged precipitation sum per time interval and 1 km grid cell.

As the RADKLIM data formats were adopted from RADOLAN, the data processing is very similar for both products and both will be referred to as RADOLAN throughout this section.

All raw data archives need to be unzipped for data import using the function `unzip_RW_binaries()` for hourly data or `unzip_YW_binaries()` for 5-minute data from *radproc*'s raw module. Both functions automatically generate a folder structure of yearly and monthly directories for the available time series, and gzip compress all unzipped binary files. The latter is a relatively slow process because of the large number of files but it is necessary to save hard drive space.

Subsequently, the new folder with all unpacked, compressed binary files can be passed to the overarching function `create_idraster_and_process_radolan_data()` which automates the entire process of data import, conversion to DataFrames and saving to HDF5. Internally, this function calls a series of helper and wrapper functions dividing the task into separate parts. The underlying binary file import into a two-dimensional *NumPy* array and a metadata dictionary is based on a slightly modified version of *wradlib*'s `read_RADOLAN_composite()` function. Consecutively, all binary data are imported and the row order is reversed for each array. The latter is necessary in order to avoid the data grid to be upside down because the binary data block starts in the lower left grid corner whereas ESRI grids are created starting in the upper left corner. Next, the reversed arrays are reshaped to one-dimensional arrays and these are inserted into monthly DataFrames by another function. The RADOLAN pixels are numbered and converted to DataFrame columns whereas every DataFrame row is labelled with the corresponding timestamp from the

RADOLAN metadata. These monthly DataFrames are saved as datasets in the specified HDF5 file.

Optionally, if ArcGIS is available, a polygon GIS shapefile or feature class containing the outline of a study area can be passed to the processing function. In that case, *radproc*'s `arcgis` module is accessed to create a so-called ID raster for the national RADOLAN grid in stereographic projection which allows for spatial localisation of the numbered RADOLAN pixels. Each ID value of this raster corresponds to a DataFrame column since these are labelled with the ID numbers. The tool automatically detects the input radar data product and applies the corresponding grid size and location. The ID raster is then clipped to the extent of the given shapefile to obtain the IDs located within the study area. Finally, the clipped ID raster is converted into a one-dimensional *NumPy* array called ID array, and NoData values are removed (see Figure 3). The resulting ID array is used to select the RADOLAN pixels within the study area upon DataFrame creation.

The generated HDF5 file with monthly datasets, which is compressed by default to save hard drive space, can be directly and quickly accessed by *pandas* and is the basis for all other *radproc* functions. The entire workflow of raw data processing is illustrated in Figure 4.

Temporal aggregation

Besides the use of precipitation sums for climatological or hydrological analysis or as model inputs, the aggregation of longer time periods should always be one of the first steps in a workflow using weather radar data in order to assess data quality in a given study area. Many systematic measurement and correction errors which cause bias such as spokes, clutter pixels or areas of missing data, are visible, e.g., in a map showing the mean annual precipitation sum.

From any HDF5 file having the structure described above, single monthly DataFrames can be loaded with *radproc*'s `load_month()` function or longer periods can be loaded with `load_months_from_hdf5()` for further analysis, plotting or data exports.

Furthermore, the *core* module offers several functions for automated temporal aggregation to hours, days, months, years or hydrological seasons. These functions access the

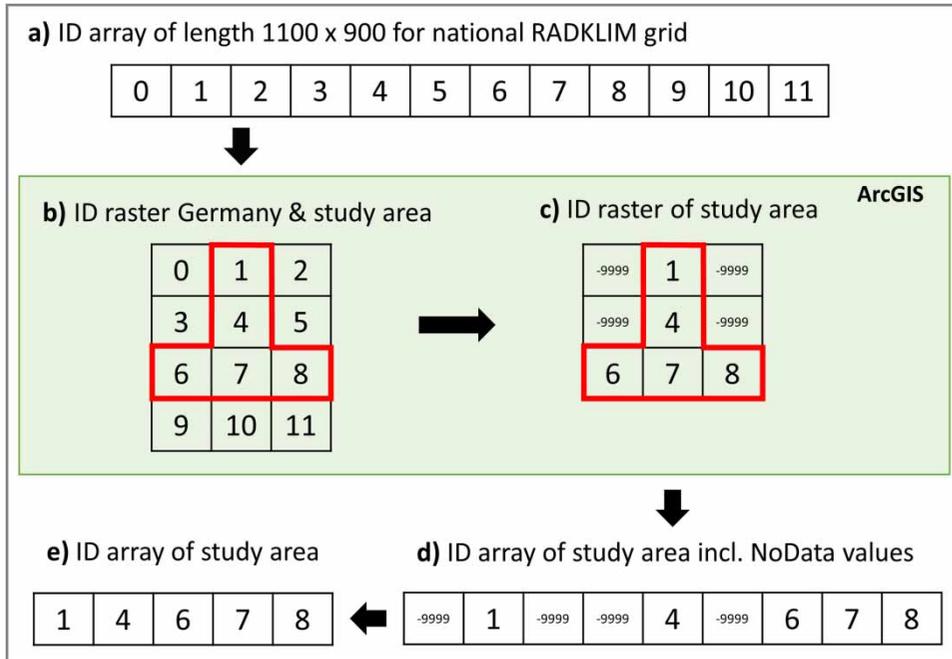


Figure 3 | Methodology for data clipping using an ID raster.

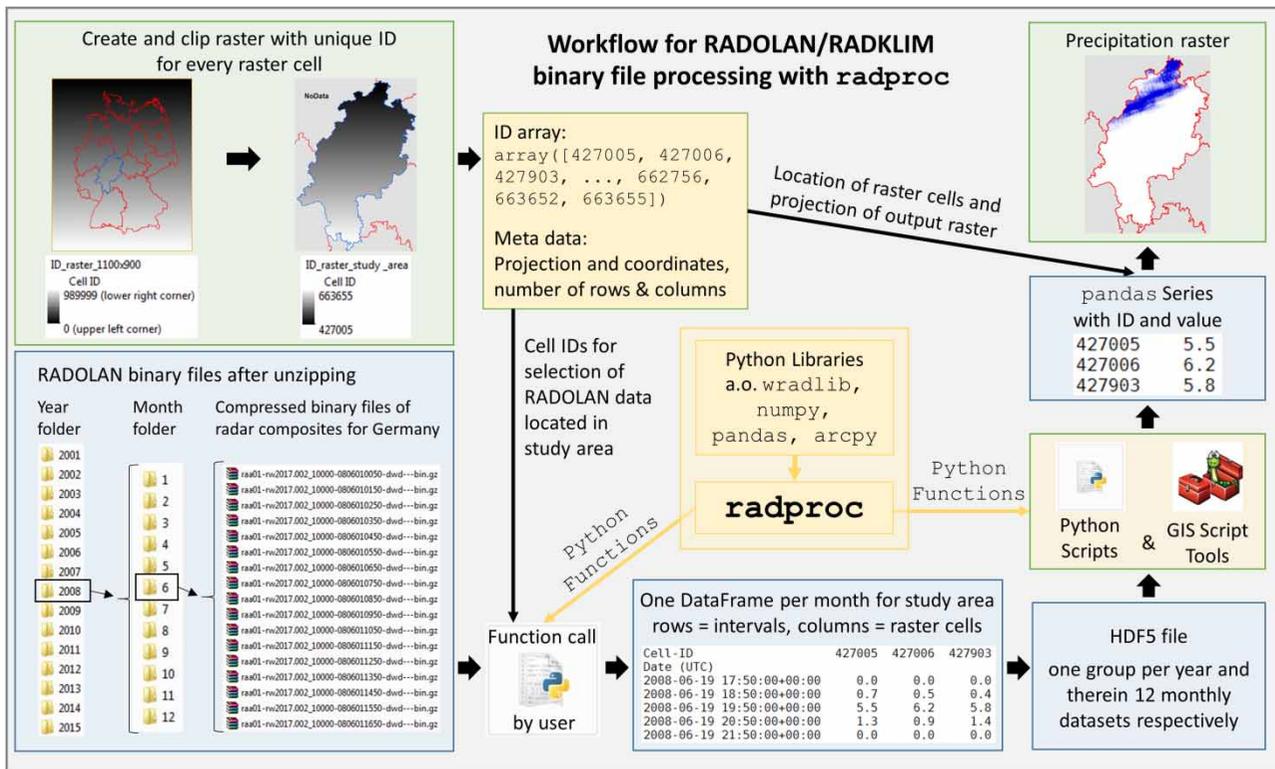


Figure 4 | Workflow for RADOLAN and RADKLIM data processing with radproc.

HDF5 file via the load functions and iteratively load and resample all data within the specified time period. For example, a call of the function `hdf5_to_years()` with the parameters `year_start` and `year_end` set to 2012 and 2017, respectively, returns a DataFrame with six rows, each of them containing the annual precipitation sum per pixel. A subsequent call of this DataFrames' `mean()` method yields – depending on the specified axis – either the spatially or temporally averaged annual precipitation.

Figure 5 shows the function call described above and an excerpt of the created output DataFrame located in the Harz Mountains, a low mountain range in the transition area between Northern and Central Germany, in a Jupyter Notebook (<https://jupyter.org/>).

Internally, `hdf5_to_years()` is only a wrapper function that calls `load_years_and_resample()`, which is actually used by all of *radproc*'s resampling functions. It iterates over all months within all years of the specified time period, whereby the DataFrame for each month is loaded and resampled individually in order to reduce the required memory. The DataFrames are either resampled to the respective target frequency or, if the latter is equal to or lower than 'month', they are resampled to a single-row DataFrame with the monthly precipitation sum. The first resampled month DataFrame of the first year is initialised as the future output DataFrame and afterwards, one after the other, all resampled month DataFrames are appended. After the loops, the output DataFrame is finally resampled to the target frequency.

Data exchange with ArcGIS

Radproc's *arcgis* module provides a set of functions for data exchange between ArcGIS and Python as well as some geospatial analysis functions, e.g., for extended zonal statistics and data extraction from raster cells to points.

For the export of radar data from DataFrames to single raster datasets, the function `export_to_raster()` can be used, whereas the function `export_dfrows_to_gdb()` handles the export of entire DataFrames into new File Geodatabases. The latter function exports every DataFrame row to one raster dataset, whereby it automatically derives the file names from the DataFrame index. Additionally, a list of statistical parameters can be passed to the function to calculate some statistical characteristics from the input DataFrame and export these, too. For example, a *statistics* list with the entries 'mean' and 'max' yields two additional exported raster datasets, each of them containing the mean and maximum value per cell, respectively. Figure 6 shows the function call and its results for exporting the DataFrame with the annual precipitation sums generated in the 'temporal aggregation' subsection.

Moreover, feature-class attribute tables can be directly imported into *pandas* DataFrames with `attribute_table_to_df()` and, in return, a list of DataFrame columns can be joined to an attribute table using `join_df_columns_to_attribute_table()`. Besides data exchange with other geodata, this provides a seamless integration of point feature-classes, which is the typical geodata format for rain gauge

```
RW = r"C:\HDF5\RADKLIM_RW_Germany.h5"
selection = [410886,410887,410888,411786,411787,411788,412686,412687,412688]
df_years = rp.hdf5_to_years(HDFFile=RW, year_start=2012, year_end=2017)
df_years.loc[:,selection]
```

Cell-ID	410886	410887	410888	411786	411787	411788	412686	412687	412688
Date (UTC)									
2012-12-31 00:00:00+00:00	839.000000	684.700012	751.500000	749.000000	639.500000	688.899963	816.899963	787.799988	687.900024
2013-12-31 00:00:00+00:00	913.299988	940.900024	979.900024	946.700012	954.299988	928.299988	895.099976	940.900024	903.500000
2014-12-31 00:00:00+00:00	1255.300049	1318.099976	1284.800049	1236.400024	1300.000000	1277.799927	1196.099976	1197.200073	1216.599976
2015-12-31 00:00:00+00:00	1096.000000	1156.400024	1134.500000	1108.699951	1202.300049	1181.400024	1133.800049	1130.500000	1160.099976
2016-12-31 00:00:00+00:00	861.700012	887.799988	883.500000	845.400024	909.700012	897.799988	855.400024	841.000000	854.799988
2017-12-31 00:00:00+00:00	1459.599976	1542.400024	1534.599976	1499.000000	1603.500000	1570.599976	1521.199951	1505.400024	1512.400024

Figure 5 | Aggregating RADKLIM RW data to annual precipitation sums.

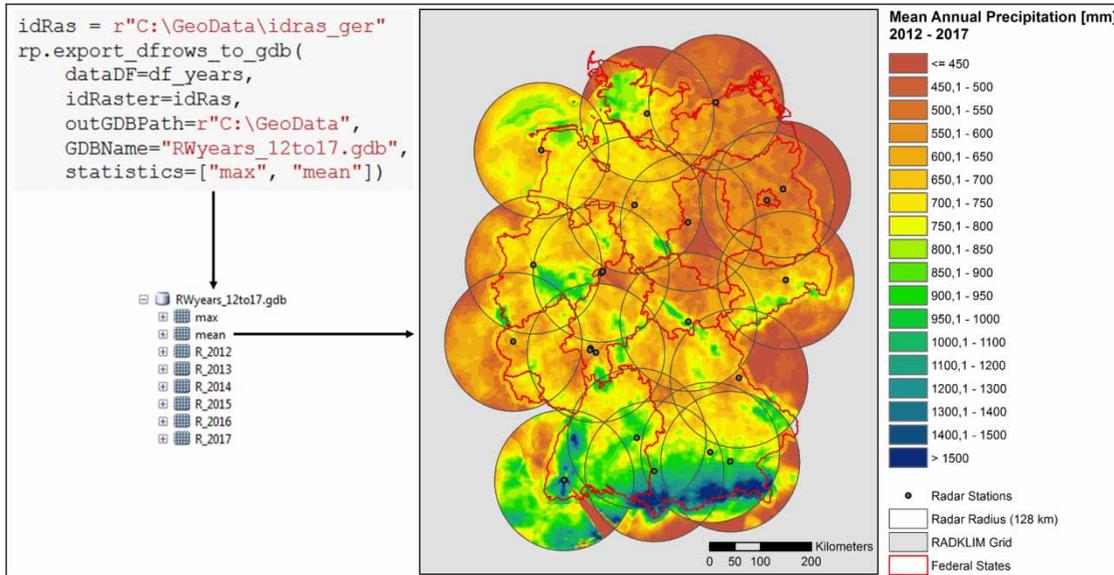


Figure 6 | Exporting the annual precipitation sums to raster datasets.

measurements, into the data analysis workflow. This is an important feature for comparison of gauge and radar datasets. To complete this data exchange circle, the function *rastervalues_to_points()* receives a list of raster datasets and a point feature-class and, by location, extracts all corresponding raster values to fields in the attribute table.

Detection and count of heavy rainfall

One of the primary reasons for developing RADKLIM was to provide a highly resolved nationwide dataset for the analysis of recent changes in rainfall-related extreme weather events (Winterrath *et al.* 2017). As a starting point for heavy rainfall analysis, *radproc* currently offers three functions providing an overview of the heavy rainfall behaviour and frequency in a given study area.

The function *find_heavy_rainfalls()* checks a time period for the exceedance of a given rainfall intensity threshold and returns a DataFrame with all intervals meeting the given criteria. This way, the exact time and location of heavy rainfall intervals can be identified and the selected intervals can subsequently be exported for visualisation.

Using the same iterative approach as the resampling functions, *find_heavy_rainfalls()* accesses a given HDF5 file via the load functions in the *core* module and checks the time series between the parameters *year_start* and

year_end for rainfall intervals exceeding specific thresholds. Here, the parameter *thresholdValue* defines the rainfall intensity threshold in mm per time unit (given by input data) to be checked for exceedance independently for each raster cell. Additionally, the parameter *minArea* specifies the number of raster cells in which the threshold must be exceeded for the interval to be selected, whereby these cells do not need to be adjacent. This parameter can be used to consider the surface area of rainfall cells, but also to take potentially known cells biased by clutter into account. Finally, the time period to be checked can be described in more detail by setting the *season* parameter to periods such as year, summer, winter or any single month or range of months.

As an example, Figure 7 shows a function call, which checks whether a precipitation amount of 100 mm/h (as the input dataset RW has an hourly resolution) has been exceeded in at least one cell anywhere in the nationwide 1,100 × 900 grid in any month of May in the period 2001 to 2017. If this holds true, the respective interval is contained in the output DataFrame. The last two lines of code select all columns (cells) containing any value greater than 100 in order to reduce the number of displayed columns. Moreover, this cell selection gives an idea, in how many cells such high rainfall amounts occurred.

As a result, this short analysis of the RADKLIM RW dataset reveals that, throughout the entire dataset, a

precipitation amount of 100 mm has occurred in nine hourly intervals between 2001 and 2017 in the month of May with a total number of 97 cells exceeding this threshold at least once.

Taking the same parameters into account, the function `count_heavy_rainfall_intervals()` also checks a time period for exceedances meeting the given criteria, but returns a single-row DataFrame with a count of exceedances per cell instead of the intervals themselves. This count gives a good overview of the heavy rainfall frequency and its spatial distribution in the study area.

Finally, the third function `duration_sum()` computes the rolling precipitation sum from data in 5 minute resolution for a defined duration D and saves the resulting DataFrames to a new HDF5 file. The calculation considers transitions between subsequent months and yields monthly DataFrames in 5 minute resolution, whose intervals contain the respective precipitation sum of the last D minutes, that is, the last $D/5$ intervals. Due to the standardised format, the resulting HDF5 file can be used as input for

`find_heavy_rainfalls()` to further detect and analyse extreme rainfall events which may have been separated and thus attenuated by the artificial interval boundaries in data with a lower temporal resolution such as RW. Nevertheless, when analysing the results, it has to be taken into account that subsequent intervals are not statistically independent because a single original 5 minute interval influences several intervals in the duration dataset. As duration sums are a commonly used method in hydrologic engineering, further analysis methods building upon them might be implemented to `radproc` in future.

DWD MR90 rain gauge data processing

In order to facilitate data comparison and, thus, data quality assessment, `radproc`'s `dwd_gauge` module provides functions for automated rain gauge data processing. Currently, only 1-minute gauge data in the DWD MR90 format are supported, but further functions to support other input

```
RW = r"C:\HDF5\RADKLIM_RW_Germany.h5"
hr = rp.find_heavy_rainfalls(HDFFile=RW, year_start=2001, year_end=2017,
                           thresholdValue=100, minArea=1, season="May")
cellsExceeding100mm = hr[hr>100].dropna(axis='columns', how='all').columns
hr.loc[:, cellsExceeding100mm]
```

Cell-ID	285040	286844	467863	467864	470547	470548	471446	471447	471448	471449	...
Date (UTC)											
2007-05-26 19:50:00+00:00	0.000000	0.000000	0.000000	0.000000	0.0	0.0	0.000000	0.000000	0.000000	0.0	...
2008-05-31 14:50:00+00:00	0.000000	0.000000	0.000000	0.000000	0.0	0.0	0.000000	0.000000	0.000000	0.0	...
2008-05-31 18:50:00+00:00	0.000000	0.000000	115.099998	104.400002	6.8	4.7	6.400000	6.200000	7.500000	6.1	...
2009-05-26 14:50:00+00:00	0.000000	0.000000	0.000000	0.000000	0.6	0.3	0.700000	0.600000	0.600000	0.5	...
2009-05-26 15:50:00+00:00	0.000000	0.000000	0.300000	0.400000	0.6	0.6	0.400000	0.500000	0.600000	0.7	...
2011-05-19 19:50:00+00:00	0.000000	0.000000	0.000000	0.000000	0.0	0.0	0.000000	0.000000	0.000000	0.0	...
2016-05-23 18:50:00+00:00	0.000000	0.000000	1.100000	0.600000	110.5	102.5	102.400002	110.400002	105.400002	104.0	...
2016-05-27 19:50:00+00:00	0.000000	0.000000	0.000000	0.000000	0.0	0.0	0.000000	0.000000	0.000000	0.0	...
2017-05-29 23:50:00+00:00	109.099998	106.800003	0.000000	0.000000	0.0	0.0	0.000000	0.000000	0.000000	0.0	...

9 rows × 97 columns

Figure 7 | Detecting rainfall intervals exceeding 100 mm/h.

formats, especially the freely available data from DWD Climate Data Centre, are currently under development.

A MR90 rain gauge dataset comprises one data file and one metadata file. These two files per gauge station need to be saved in separate directories. To support the creation of a point feature-class from the metadata, the function `summarize_metadata_files()` summarises the information on station ID, station name, geographic coordinates and height above sea level from the metadata files into one single text file. A single data file can be imported into a one-column DataFrame with `stationfile_to_df()`.

Finally, the function `dwd_gauges_to_hdf5()` offers an automated iterative processing and import of all data files in a directory. The gauge data are converted into the same DataFrame format as the radar data. To make the data formats completely match, the time zone of the gauge data is converted to UTC and the data are resampled to the same 5-minute intervals as the 5-minute RADKLIM product YW. The final DataFrame contains one column per rain gauge. Finally, it is divided into monthly DataFrames, which are saved to the standardised HDF5 file format. As described above, *radproc*'s analysis and resampling functions work for all datasets converted this way. Consequently, the function calls for resampling and heavy rainfall detection shown in Figures 5 and 7 are exactly the same for the gauge data except for a different input HDF5 file path. However, instead of exporting the rows of the output DataFrame to rasters as shown in Figure 6, the rows can be exported to new fields of a feature class attribute table using `join_df_columns_to_attribute_table()`.

FUTURE DEVELOPMENTS, LIMITATIONS AND CONCLUSIONS

In this paper, the Python library *radproc* providing a GIS-compatible platform for automated radar data processing and analysis was introduced.

The software review revealed that there is a considerable gap concerning functionality and ease of usability between open source and commercial software products for weather radar data processing and analysis in Germany, and the outlook on software for other radar datasets indicated a similar situation in other parts of the world. Moreover,

only a small number of tools – none of which are OSS – support long-term climatological analysis of RADOLAN and RADKLIM data, yet.

The development of the RADKLIM dataset and other radar climatologies has opened up new application fields which, along with the vast amount of data, require new, innovative processing frameworks and analysis methods. These are probably most likely to develop in community-based open source research software projects, which could be demonstrated by the development of the *wradlib* library (Heistermann *et al.* 2013). Moreover, OSS helps users to build upon each other's work and to increase the reproducibility of research results.

The development of *radproc* is an attempt to reduce the gap identified in the review of existing tools by providing a highly extensible OSS that facilitates and largely automates data processing and converts data into flexible, widely used data formats. *Radproc* is the first OSS to provide support for long-term 5-minute RADOLAN and RADKLIM data processing and analysis. The developed modules and the innovative processing workflow with a focus on the unification of different data formats are a solid foundation to turn the project into a community-based platform for radar data processing, analysis and conversion in future. In order to enable a wide usage and support collaboration for further development, *radproc* is distributed under the permissive MIT license, complemented with an additional provision, which requires the source code of modified versions of the software to be made freely available in a public repository (<http://www.pgweb.uni-hannover.de/licensing.html>). The software is being used by several working groups in the fields of geography, hydrology and natural hazards risk assessment throughout Germany, primarily for detection and reanalysis of past heavy rainfall events and for the provision of model inputs. Along with the feature requests and feedback the author received, this shows, that there is considerable demand for such a project. Due to the same programming language and the usage of *NumPy* arrays as common fundamental data structure, *radproc* is also compatible with *wradlib* and allows the application of, e.g., *wradlib*'s georeferencing and visualisation functions. Consequently, both libraries can be combined for individual radar data analysis workflows and complement each other.

Nevertheless, the chosen implementation of *radproc* still has some technical limitations. The most important one is that all data processing operations are performed in working memory. Unfortunately, in HDF5 files, the space reserved for column header information in flexible ‘table’ datasets is limited to approximately 2,000 columns, which is by far exceeded by the number of cells in most study areas. Consequently, the monthly DataFrames need to be stored as ‘fixed’ tabular datasets, which do not support flexible operations such as searching and selecting subsets of the data. Instead, the DataFrames need to be loaded into working memory entirely. For the hourly RW products, this is not a major issue, but for 5-minute radar data, the size of the study area that can be processed is scaled with the available memory. Furthermore, the *pandas* HDF5 API does not yet provide options for flexible metadata storage, which spills over to *radproc*.

Like all other reviewed OSS, *radproc* does not provide any GUI, yet. Hence, its application still requires a certain readiness by the user to learn some very basic Python syntax. But with the increasing number of online courses and *radproc*'s extensive documentation including an installation guide, a full library reference and tutorials directly generated from Jupyter Notebooks, this is feasible without much effort even for users without any prior programming skills.

However, due to the tight integration of ArcGIS, it could be an option to develop a *radproc* GIS toolbox to facilitate application. So far, this has been tested, but the connection between the separate Python installations of ArcGIS and a scientific distribution like Anaconda (<https://www.anaconda.com>), which is necessary to access all of *radproc*'s dependencies and the ArcGIS *arcpy* module, is rather difficult to establish. Enabling an Anaconda IDE such as Spyder or the Jupyter Notebook to import *arcpy* is easy and quickly done, but enabling the import of any additional site-packages into ArcGIS, which is necessary to execute GIS tools accessing *radproc*, is much more complicated and sparsely documented. This might become easier through the planned porting to Python 3 and the implementation of ArcGIS Pro and will be pursued within this context.

Another option for future developments and a repeated request is the addition of a module to support QGIS as an alternative to ArcGIS in order to turn the entire workflow into an open source project. Due to *radproc*'s extensible

modular structure, such a QGIS module or any other modules to support further radar or gauge data formats could be added, but neither of these is specifically planned, yet, except for the additional DWD gauge data import routines described in the previous section. Currently, an additional module for the calculation of rainfall erosivity, the R factor of the Universal Soil Loss Equation (Wischmeier & Smith 1978), is being developed and will be added in future.

Radproc constitutes a powerful open source tool for automated weather radar data processing and analysis and has considerable potential for further development and improvement. It contributes to facilitating radar data processing, allowing non-specialised users to cope with the vast amount of binary data and put the novel RADKLIM dataset to use. Thus, *radproc* can help to enable radar data usage for all applications that benefit from high resolution precipitation data, e.g., in research, hydrological engineering, disaster control, erosion and flood protection and environmental planning.

ACKNOWLEDGEMENTS

The author would like to thank the Hessian Agency for Nature Conservation, Environment and Geology (HLNUG) for providing partial funding for this research within the project ‘KLIMPRAX – Starkregen’, working package 1.4, and the German Weather Service for providing the RADKLIM and MR90 rain gauge data. The contributions from Gerald Kuhnt, Benjamin Burkhard, Ina Sieber and Tobias Kreklow are highly appreciated. Special thanks are extended to the *radproc* user community members Jan Lunge, Denise Harders, Senta Meinecke, Bastian Steinhoff-Knopp and Detlef Deumlich for their feedback, contributions and bug reports that helped improve *radproc*. Finally, the author would like to thank the two anonymous reviewers for their proficient and constructive comments which helped complement and improve the manuscript.

REFERENCES

- Abdella, Y. & Alfredsen, K. 2010 [A GIS toolset for automated processing and analysis of radar precipitation data.](#)

- Computers & Geosciences* **36** (4), 422–429. <https://doi.org/10.1016/j.cageo.2009.08.008>.
- Alfieri, L., Velasco, D. & Thielen, J. 2011 **Flash flood detection through a multi-stage probabilistic warning system for heavy precipitation events**. *Advances in Geosciences* **29**, 69–75. <https://doi.org/10.5194/adgeo-29-69-2011>.
- Alfieri, L., Salamon, P., Pappenberger, F., Wetterhall, F. & Thielen, J. 2012 **Operational early warning systems for water-related hazards in Europe**. *Environmental Science & Policy* **21**, 35–49. <https://doi.org/10.1016/j.envsci.2012.01.008>.
- Bartels, H., Weigl, E., Reich, T., Lang, W., Wagner, A., Kohler, O. & Gerlach, N. MeteoSolutions GmbH 2004 *Projekt RADOLAN – Routineverfahren zur Online-Aneicherung der Radarniederschlagsdaten mit Hilfe von Automatischen Bodenniederschlagsstationen (Ombrometer) (Project RADOLAN – Operational Procedures for an Online-Adjustment of Radar Precipitation Data Using Automated Rain Gauges)*. Zusammenfassender Abschlussbericht für die Projektlaufzeit von 1997 bis 2004, Deutscher Wetterdienst, Offenbach am Main, Germany.
- Benson, S., Hunter, J. & McWilliams, C. D. 2018 **HEC-MetVue – Tool for Real-Time Forecasting; Analyzing and Manipulating Gridded Datasets; Development of Design Storms for Probable Maximum Precipitation Estimation**. In *American Meteorological Society 98th Annual Meeting*, 8 January 2018, Austin, TX, USA. <https://ams.confex.com/ams/98Annual/webprogram/Paper336164.html> (accessed 19 February 2019).
- Bouilloud, L., Delrieu, G., Boudevillain, B. & Kirstetter, P.-E. 2010 **Radar rainfall estimation in the context of post-event analysis of flash-flood events**. *Journal of Hydrology* **394** (1–2), 17–27. <https://doi.org/10.1016/j.jhydrol.2010.02.035>.
- Deutscher Bundestag 2017 *Erstes Gesetz zur Änderung des Gesetzes über den Deutschen Wetterdienst: DWD-Gesetz (DWD Law), 2017 Teil I*.
- Fischer, F., Hauck, J., Brandhuber, R., Weigl, E., Maier, H. & Auerswald, K. 2016 **Spatio-temporal variability of erosivity estimated from highly resolved and adjusted radar rain data (RADOLAN)**. *Agricultural and Forest Meteorology* **223**, 72–80. <https://doi.org/10.1016/j.agrformet.2016.03.024>.
- Gaume, E., Bain, V., Bernardara, P., Newinger, O., Barbuc, M., Bateman, A., Blaškovičová, L., Blöschl, G., Borga, M., Dumitrescu, A., Daliakopoulos, I., Garcia, J., Irimescu, A., Kohnova, S., Koutroulis, A., Marchi, L., Matreata, S., Medina, V., Preciso, E., Sempere-Torres, D., Stancalie, G., Szolgay, J., Tsanis, I., Velasco, D. & Viglione, A. 2009 **A compilation of data on European flash floods**. *Journal of Hydrology* **367** (1–2), 70–78. <https://doi.org/10.1016/j.jhydrol.2008.12.028>.
- Gjertsen, U., Salek, M. & Michelson, D. B. 2003 **Gauge-adjustment of radar-based precipitation estimates – a review**. *Cartographic Perspectives* **45**, 1. <https://doi.org/10.14714/CP45.494>.
- Guzzetti, F., Peruccacci, S., Rossi, M. & Stark, C. P. 2007 **Rainfall thresholds for the initiation of landslides in central and southern Europe**. *Meteorology and Atmospheric Physics* **98** (3–4), 239–267. <https://doi.org/10.1007/s00703-007-0262-7>.
- Hänsel, P., Kaiser, A., Buchholz, A., Böttcher, F., Langel, S., Schmidt, J. & Schindewolf, M. 2018 **Mud flow reconstruction by means of physical erosion modeling, high-resolution radar-based precipitation data, and UAV monitoring**. *Geosciences* **8** (11), 427. <https://doi.org/10.3390/geosciences8110427>.
- Heistermann, M., Jacobi, S. & Pfaff, T. 2013 **Technical Note: an open source library for processing weather radar data (*wradlib*)**. *Hydrology and Earth System Sciences: HESS* **17** (2), 863–871. <https://doi.org/10.5194/hess-17-863-2013>.
- Heistermann, M., Collis, S., Dixon, M. J., Giangrande, S., Helmus, J. J., Kelley, B., Koistinen, J., Michelson, D. B., Peura, M., Pfaff, T. & Wolff, D. B. 2015 **The emergence of open-source software for the weather radar community**. *Bulletin of the American Meteorological Society* **96** (1), 117–128. <https://doi.org/10.1175/BAMS-D-13-00240.1>.
- Henja, A., Szewczykowski, M., Ernes, S. & Michelson, D. B. 2010 **The BALTRAD technical platform**. In: *ERAD 2010 - The Sixth European Conference on Radar in Meteorology and Hydrology*. http://erad2010.com/pdf/oral/wednesday/dataex/02_ERAD2010_0232.pdf (accessed 19 February 2019).
- IPCC 2013 *Climate Change 2013: The Physical Science Basis*. Contribution of Working Group I to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change, Cambridge University Press, Cambridge, UK and New York, NY. http://www.climatechange2013.org/images/report/WG1AR5_ALL_FINAL.pdf (accessed 16 April 2018).
- Kakihara, K. 2018 **Radar Network in Southeast Asia**. In *Japan Meteorological Agency, WMO/ASEAN Training Workshop on Weather Radar Data Quality and Standardization*, 5 February 2018, Bangkok, Thailand. https://www.jma.go.jp/jma/en/photogallery/WMO-ASEAN_Radar_Workshop_Feb2018/1-2_Keynote_WIGOS-Project (accessed 19 February 2019).
- Keupp, L., Winterrath, T. & Hollmann, R. 2017 **Use of Weather Radar Data for Climate Data Records in WMO Regions IV and VI**. https://www.wmo.int/pages/prog/wcp/ccl/opace/opace2/documents/TT-URSDCM_Use_Remote_Sensing_DataClimateMonitoringRAIV-VI.pdf
- Krajewski, W. F. & Smith, J. A. 2002 **Radar hydrology: rainfall estimation**. *Advances in Water Resources* **25**, 1387–1394.
- Kreklow, J. 2018 **Radproc – A Gis-Compatible Python-Package For Automated Radolan Composite Processing And Analysis**. <https://doi.org/10.5281/zenodo.1316376>. Zenodo.
- Mckinney, W. 2010 **Data Structures for Statistical Computing in Python**. In: *Proceedings of the 9th Python in Science Conference*, Austin, TX, USA, pp. 51–56. <https://conference.scipy.org/proceedings/scipy2010/pdfs/mckinney.pdf> (accessed 15 November 2018).
- McWilliams, C. D. 2017 **HEC-MetVue Implementation**, 19 January 2017. https://acwi.gov/hydrology/minutes/MetVue_Presentation-19Jan2017.pdf (accessed 19 February 2019).

- Meischner, P. 2004 *Weather Radar: Principles and Advanced Applications*, 1st edn. Springer, Berlin, Heidelberg, Germany.
- Michelson, D. B., Lewandowski, R., Szewczykowski, M. & Beekhuis, H. 2014 *EUMETNET OPERA Weather Radar Information Model for Implementation with the HDF5 File Format: Version 2.2*. EUMETNET, Brussels. http://eumetnet.eu/wp-content/uploads/2017/01/OPERA_hdf_description_2014.pdf (accessed 16 November 2018).
- Oliphant, T. E. 2006 *A Guide to NumPy*. Trelgol Publishing, USA.
- Overeem, A., Holleman, I. & Buishand, A. 2009 *Derivation of a 10-year radar-based climatology of rainfall*. *Journal of Applied Meteorology and Climatology* **48** (7), 1448–1463. <https://doi.org/10.1175/2009JAMC1954.1>.
- Panagos, P., Ballabio, C., Borrelli, P., Meusburger, K., Klik, A., Rousseva, S., Tadic, M. P., Michaelides, S., Hrabalíková, M., Olsen, P., Aalto, J., Lakatos, M., Rymaszewicz, A., Dumitrescu, A., Begueria, S. & Alewell, C. 2015 *Rainfall erosivity in Europe*. *Science of the Total Environment* **511**, 801–814. <https://doi.org/10.1016/j.scitotenv.2015.01.008>.
- Panagos, P., Ballabio, C., Meusburger, K., Spinoni, J., Alewell, C. & Borrelli, P. 2017 *Towards estimates of future rainfall erosivity in Europe based on REDES and WorldClim datasets*. *Journal of Hydrology* **548**, 251–262. <https://doi.org/10.1016/j.jhydrol.2017.03.006>.
- Quirnbach, M., Einfalt, T., Langstädtler, G., Janßen, C., Reinhardt, C. & Mehlig, B. 2013 *Extremwertstatistische Untersuchung von Starkniederschlägen in NRW (ExUS) (A study of extreme value statistics of heavy rainfalls in NRW)*. *Korrespondenz Abwasser und Abfall* **60** (7), 591–599.
- Raghavan, S. 2003 *Radar Meteorology*, 1st edn. Springer Science + Business, Dordrecht, Germany.
- Ramos, M. H., Creutin, J.-D. & Leblois, E. 2005 *Visualization of storm severity*. *Journal of Hydrology* **315** (1–4), 295–307. <https://doi.org/10.1016/j.jhydrol.2005.04.007>.
- R Core Team 2018 *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. <https://www.R-project.org> (accessed 20 November 2018).
- Risal, A., Lim, K. J., Bhattarai, R., Yang, J. E., Noh, H., Pathak, R. & Kim, J. 2018 *Development of web-based WERM-S module for estimating spatially distributed rainfall erosivity index (EI30) using RADAR rainfall data*. *Catena* **161**, 37–49. <https://doi.org/10.1016/j.catena.2017.10.015>.
- Saltikoff, E., Haase, G., Leijnse, H., Novák, P. & Delobbe, L. 2018 *OPERA – past, present and future*. In: *ERAD 2018 – 10th European Conference on Radar in Meteorology and Hydrology*, 2 July 2018, Utrecht, The Netherlands. http://projects.knmi.nl/erad2018/ERAD2018_extended_abstract_025.pdf (accessed 18 February 2019).
- Segoni, S., Rossi, G., Rosi, A. & Catani, F. 2014 *Landslides triggered by rainfall: a semi-automated procedure to define consistent intensity–duration thresholds*. *Computers & Geosciences* **63**, 123–131. <https://doi.org/10.1016/j.cageo.2013.10.009>.
- Sene, K. 2010 *Hydrometeorology*. Springer, Dordrecht, The Netherlands.
- Seo, B.-C., Krajewski, W. F., Kruger, A., Domaszczynski, P., Smith, J. A. & Steiner, M. 2011 *Radar-rainfall estimation algorithms of Hydro-NEXRAD*. *Journal of Hydroinformatics* **13** (2), 277–291. <https://doi.org/10.2166/hydro.2010.003>.
- Smith, J. A., Baeck, M. L., Villarini, G., Welty, C., Miller, A. J. & Krajewski, W. F. 2012 *Analyses of a long-term, high-resolution radar rainfall data set for the Baltimore metropolitan region*. *Water Resources Research* **48** (4), 616. <https://doi.org/10.1029/2011WR010641>.
- Steinhoff-Knopp, B. & Burkhard, B. 2018 *Soil erosion by water in Northern Germany: long-term monitoring results from Lower Saxony*. *Catena* **165**, 299–309. <https://doi.org/10.1016/j.catena.2018.02.017>.
- St Laurent, A. M. 2008 *Understanding Open Source and Free Software Licensing*. O'Reilly Media, Sebastopol, CA, USA.
- The HDF Group 2018 *Hierarchical Data Format*. <https://portal.hdfgroup.org> (accessed 18 December 2018).
- Thorndahl, S., Einfalt, T., Willems, P., Nielsen, J. E., Veldhuis, M.-C. t., Arnbjerg-Nielsen, K., Rasmussen, M. R. & Molnar, P. 2017 *Weather radar rainfall data in urban hydrology*. *Hydrology and Earth System Sciences: HESS* **21** (3), 1359–1380. <https://doi.org/10.5194/hess-21-1359-2017>.
- van Rossum, G. 2001–2019 *Python*. Python Software Foundation, Beaverton, USA. <https://www.python.org/> (accessed 27 February 2019).
- Winterrath, T., Brendel, C., Hafer, M., Junghänel, T., Klameth, A., Walawender, E., Weigl, E. & Becker, A. 2017 *Erstellung Einer Radargestützten Niederschlagsklimatologie (Creation of A Radar-Based Precipitation Climatology)*. Berichte des Deutschen Wetterdienstes 251, Deutscher Wetterdienst, Offenbach am Main, Germany. https://www.dwd.de/DE/leistungen/pfb_verlag_berichte/pdf_einzelbaende/251_pdf.pdf?__blob=publicationFile&v=2 (accessed 10 October 2018).
- Winterrath, T., Brendel, C., Hafer, M., Junghänel, T., Klameth, A., Lengfeld, K., Walawender, E., Weigl, E. & Becker, A. 2018a *Radar Climatology (RADKLIM) Version 2017.002 (RW); Gridded Precipitation Data for Germany*. https://doi.org/10.5676/DWD/RADKLIM_RW_V2017.002.
- Winterrath, T., Brendel, C., Hafer, M., Junghänel, T., Klameth, A., Lengfeld, K., Walawender, E., Weigl, E. & Becker, A. 2018b *Radar Climatology (RADKLIM) Version 2017.002 (YW); Gridded Precipitation Data for Germany*. https://doi.org/10.5676/DWD/RADKLIM_YW_V2017.002.
- Wischmeier, W. H. & Smith, D. D. 1978 *Predicting Rainfall Erosion Losses. A guide to conservation planning*. U.S. Department of Agriculture, Agriculture Handbook No. 537, Washington, USA.
- Wright, D. B., Smith, J. A., Villarini, G. & Baeck, M. L. 2013 *Estimating the frequency of extreme rainfall using weather radar and stochastic storm transposition*. *Journal of Hydrology* **488**, 150–165. <https://doi.org/10.1016/j.jhydrol.2013.03.003>.
- Wright, D. B., Smith, J. A., Villarini, G. & Baeck, M. L. 2014 *Long-term high-resolution radar rainfall fields for urban hydrology*. *Journal of the American Water Resources Association* **50** (3), 713–734. <https://doi.org/10.1111/jawr.12139>.

- Wright, D. B., Mantilla, R. & Peters-Lidard, C. D. 2017 [A remote sensing-based tool for assessing rainfall-driven hazards](#). *Environmental Modelling & Software* **90**, 34–54. <https://doi.org/10.1016/j.envsoft.2016.12.006>.
- Xie, H., Zhou, X., Vivoni, E. R., Hendrickx, J. M. H. & Small, E. E. 2005 [GIS-based NEXRAD Stage III precipitation database: automated approaches for data processing and visualization](#). *Computers & Geosciences* **31** (1), 65–76. <https://doi.org/10.1016/j.cageo.2004.09.009>.
- Zhang, X. & Srinivasan, R. 2010 [GIS-based spatial precipitation estimation using next generation radar and raingauge data](#). *Environmental Modelling & Software* **25** (12), 1781–1788. <https://doi.org/10.1016/j.envsoft.2010.05.012>.
- Zolina, O., Simmer, C., Kapala, A., Bachner, S., Gulev, S. & Maechel, H. 2008 [Seasonally dependent changes of precipitation extremes over Germany since 1950 from a very dense observational network](#). *Journal of Geophysical Research* **113** (D6), D05109. <https://doi.org/10.1029/2007JD008393>.

First received 1 March 2019; accepted in revised form 23 April 2019. Available online 15 May 2019