# Package 'WaterBalanceR'

October 14, 2026

**Type** Package

**Title** Calculate High Resolution Water Balance of Starch Potatoes

**Version** 0.1.22

**Description** Calculates the water balance of starch potatoes from Normalized Distance Vegetation Index (NDVI) images, German Weather Service (DWD) reference evapotranspiration, German Weather Service RADOLAN precipitation data and irrigation information. For more details see Piernicke et al. (2025) <doi:10.3390/rs17183227>.

**License** AGPL-3

**Encoding** UTF-8

**Depends** R (>= 4.1.0)

**Imports** raster, sf, janitor, zoo, lubridate, stars, ggplot2, tidyr, rdwd, utils, terra, gridExtra, httr, stringr, rvest, sp, geosphere, RCurl, grDevices, tidyselect, magrittr, jsonlite, dplyr, readxl, scales, methods, RSelenium, openeo, rlang

**URL** https://doi.org/10.5281/zenodo.15046338, https://github.com/thomasp-gfz/WaterBalanceR

**RoxygenNote** 7.3.3

**VignetteBuilder** knitr

**Suggests** knitr, rmarkdown

**NeedsCompilation** no

**Author** Thomas Piernicke [aut, cre] (ORCID: <https://orcid.org/0009-0006-9760-5162>)

**Maintainer** Thomas Piernicke <thomasp@gfz.de>

**Repository** CRAN

**Date/Publication** 2026-10-14 10:20:22 UTC

## Contents

calcWB . . . . . 2
calcWBplots . . . . . 4

DownloadET0fromArable . . . . . 4
DownloadET0fromDWD . . . . . 5
DownloadRadolanFromDWD . . . . . 6
DownloadRaindancer . . . . . 7
DownloadRaindancerCombineCharts . . . . . 8
DownloadSentinel2 . . . . . 8

**Index** **10**

---

`calcWB` *Calculate Water Balance from UAV or PlanateScope NDVI Data*

---

## Description

Calculate Water Balance using DJI Phantom 4 Multispectral or PlanetScope NDVI data. Reference Evapotranspiration can be used either from German Weather Service (DWD) or Arable Mark 2 ground stations from your site. Precipitation is gathered from either the German Weather Service (DWD) product "RADOLAN" or FURUNO WR 2120, if available.

## Usage

```
calcWB(
  mypath,
  shape_site = NA,
  target_res = 5,
  last_NDVI_0 = NA,
  ET_ref = NA,
  ET_ref_dl = "DWD",
  output_year = NA,
  precip_source = precip_source,
  path_WR_precip = NA,
  irrig_sf = NA,
  irrigation_efficiency = 1,
  save_shape = TRUE,
  save_geotiff = TRUE,
  save_RDATA = TRUE,
  arable_user = NA,
  arable_pass = NA
)
```

## Arguments

| | |
|---|---|
| `mypath` | Path to your project main folder (string). The main folder needs to contain the subfolders "NDVI_Files" containing your NDVI-files for your AOI. |
| `shape_site` | shapefile of AOI (string) |
| `target_res` | Resolution of product (integer). Default is 5 m, but can be turned down to at least 3 m. |

`last_NDVI_0` Number of day with day (DOI, integer) with NDVI = 0, i.e. last day before germination.

`ET_ref` Either csv-file with reference ET for every day of vegetation period or recent date (read.csv(paste(mypath,"/ET0_Arable_2021.csv",sep=""),sep=",")) or leave at NA. When using the list, the first column needs to be ascending numerized (integer) from one on with empty header. The second column contains the reference ET value for the certain DOY (float) with header "V1". The third column needs to be the date (format "YYYY-MM-DD", e.g. "2021-05-01). When left NA (default), the reference ET is automatically downloaded from either German Weather Servcice (DWD, default) or Arable, if you have an account. This decision needs to be made in the next step.

`ET_ref_dl` If you do not have any reference ET data, leave "ET_ref" as "NA" and choose here between "DWD" to download from German Weather Service ("DWD") or "Arable" to download from your Arable account ("string"). If you choose to download from your Arable Account, you need to put in your Arable login data.

`output_year` Number of year, you are processing (format: "YYYY", e.g. 2021, integer).

`precip_source` Choose either "RADOLAN" (default, string) or "FURUNO" (string) depending on the source you would like to use.

`path_WR_precip` Choose the path to your precipitation data (string). This should be a folder containing shapefiles with precipitation data for every day during the vegetation period you are interested in. If you leave it an NA (default), precipitation data is downloaded from German Weather Service (DWD).

`irrig_sf` Path to shapefile containing the irrigation data (string), e.g. st_read(paste(mypath,"/Shapefile/Buffer_36m
The shapefile needs to contain the following coloumns: Drck_mn (water pressure, float), Dtm_Uh_ (Date and time, string, format: "YY-MM-DD hh:mm:ss"), timedif (time difference between steps in hours, float), dst_gps (spatial distance between in m the logs of sprinkler, float), gschwn_ (speed of sprinkler in m/s, float), Brg_GPS (irrigation amount, mm, float), Dstnz_k (cumulated spatial distance between logs in m, float), DOY (day of year, integer), geometry (geometric geometry). You can also generate this shapefile by 1st using the function "DownloadRaindancer" to download all of your irrigation data that was logged by raindancer. Take note, that irrigation data can only be downloaded from the last 12 days. So you should downoad regularly. In the 2nd step you can use the function "DownloadRaindancerCombineCharts" to combine the downloaded charts and process them to the needed shapefile. The resulting shapefile is being updated witht every iteration of download.

`irrigation_efficiency` Choose irrigation efficiency, float between 0 and 1 (default). Here, irrigation efficiency is meant to be as the fraction of water that was infiltrated in the soil from the amount that was applied.

`save_shape` Save results as shapefile? (TRUE or FALSE, default: TRUE)

`save_geotiff` Save results as geotiff? (TRUE or FALSE, default: TRUE)

`save_RDATA` Save results as RDATA? (TRUE or FALSE, default: TRUE)

`arable_user` Your user name for your Arable account (string). Only necessary, if you chose "ET_ref_dl" with "Arable". Else: leave at NA.

| | |
|---|---|
| arable_pass | Your password for your Arable account (string). Only necessary, if you chose "ET_ref_dl" with "Arable". Else: leave at NA. |

## Value

Shapefiles, Geotiffs and/or RDATA-files with maps showing the water balance

---

`calcWBplots` *Create .png files for results at a first glance*

---

## Description

Create .png files from .RDATA file created by calcWB() for every DOY within given timespan (earliest till latest NDVI-capture) showing NDVI, crop coefficient, crop evapotranspiration, precipitation, irrigation and water balance. Besides, .RDATA file is created, that contains mean values for selected samples.

## Usage

```
calcWBplots(source_path = NA, plant_doy = NA, buffer20 = NA, shape_site = NA)
```

## Arguments

| | |
|---|---|
| source_path | Path to .RDATA file (string) resulting from calcWB() function. |
| plant_doy | DOY (integer), when planted. |
| buffer20 | Path to buffer (string) containing shapefile with buffers of interest within study site. Read with sf::read_st(). |
| shape_site | Path to shapefile containing your AOI (string). |

## Value

.png files for every DOY within given timespan (earliest till latest NDVI-capture) showing NDVI, crop coefficient, crop evapotranspiration, precipitation, irrigation and water balance

---

`DownloadET0fromArable` *Downloads reference evapotranspiration (ET0) from an Arable account and processes the data for a given period. Only devices located within (or inside a 500 m buffer around) the provided shapefile are considered.*

---

## Description

Downloads reference evapotranspiration (ET0) from an Arable account and processes the data for a given period. Only devices located within (or inside a 500 m buffer around) the provided shapefile are considered.

### Usage

```
DownloadET0fromArable(
  user_name = NA,
  pwd = NA,
  start_date = NA,
  end_date = NA,
  shape_site = NA
)
```

### Arguments

| | |
|---|---|
| user_name | string: "user_name" |
| pwd | DOY string: "password" |
| start_date | Start date of download (string: "YYYY-MM-DD") |
| end_date | End date of download (string: "YYYY-MM-DD") |
| shape_site | Path to shapefile containing your AOI (string). |

### Value

chart containing reference evapotranspiration for every DOY during given timespan

---

## DownloadET0fromDWD *Downloads daily reference evapotranspiration (ET0) grids from the German Weather Service (DWD) open data portal for a given year, extracts the values for a specified AOI (shapefile), and saves the results as a .csv file with daily ET0 values.*

---

### Description

Downloads daily reference evapotranspiration (ET0) grids from the German Weather Service (DWD) open data portal for a given year, extracts the values for a specified AOI (shapefile), and saves the results as a .csv file with daily ET0 values.

### Usage

```
DownloadET0fromDWD(
  target_path = NA,
  test_site_shp = NA,
  target_year = NA,
  timeout = 1000
)
```

### Arguments

| | |
|---|---|
| target_path | Path to download and save csv-file with reference ET for your AOI and timespan of interest |
| test_site_shp | Path to shapefile containing your AOI (string). |
| target_year | year of interest (integer: 2021) |
| timeout | time out span for downloading data (default: 10000, exceed, if your interconnection is slow) |

### Value

chart containing reference evapotranspiration for every DOY during given timespan

---

## DownloadRadolanFromDWD

*Downloads daily precipitation data (RADOLAN) from DWD OpenData for a given AOI and time span. Depending on the date range, it uses either the "recent" or "historical" RADOLAN archives. Data are clipped to the AOI and saved as shapefiles, one per day of year (DOY).*

---

### Description

Downloads daily precipitation data (RADOLAN) from DWD OpenData for a given AOI and time span. Depending on the date range, it uses either the "recent" or "historical" RADOLAN archives. Data are clipped to the AOI and saved as shapefiles, one per day of year (DOY).

### Usage

```
DownloadRadolanFromDWD(
  target_path = NA,
  target_site = NA,
  start_date = NA,
  end_date = NA
)
```

### Arguments

| | |
|---|---|
| target_path | Path to download and save shapefile for every DOY within timespan of interest |
| target_site | Path to shapefile containing your AOI (string). |
| start_date | start date of interest (e.g.: "2021-01-01"). If empty, default is 1st Jan of recent year. |
| end_date | end date of interest (e.g.: "2021-12-31"). If empty, default is yesterday. |

### Value

Shapefiles for every DOY containing precipitation data for your AOI.

DownloadRaindancer *Scrapes data from Raindancer user account for all logged sprinklers. Beware: It only downloads the last 10.000 logs (~12 days). Requirements: Java and Firefox must be installed on your machine.*

## Description

Scrapes data from Raindancer user account for all logged sprinklers. Beware: It only downloads the last 10.000 logs (~12 days). Requirements: Java and Firefox must be installed on your machine.

## Usage

```
DownloadRaindancer(
  sourcepath = NA,
  targetpath = NA,
  port = 4486L,
  client = NA,
  user = NA,
  pass = NA,
  waitfor = 3,
  nozzle_diameter = "25_4",
  target_crs = 32633,
  ff_vis = "no"
)
```

## Arguments

| | |
|---|---|
| sourcepath | Path (string) to Firefox download folder. Look it up in your Firefox browser. |
| targetpath | Path (string) to destination folder for downloaded csv-files from Raindancer. |
| port | You need to open a port to let R and Java scrape the website´s data (default: 4486L). |
| client | Raindancer client number |
| user | Raindancer user account |
| pass | Raindancer password |
| waitfor | time to wait for loading websites. The quicker your computer and internet connection, the less it can be (integer). Default is 3. |
| nozzle_diameter | diameter of nozzle in mm (string, e.g. 17_8 = 17.8 mm). Default is "25_4". |
| target_crs | target crs |
| ff_vis | choose whether Firefox runs visibly ("yes") or hidden ("no", default). If visible, increase the waiting time (e.g., to about 3 seconds). |

## Value

csv file for all irrigation events of all sprinklers, that are logged in Raindancer Account.

DownloadRaindancerCombineCharts *Processes downloaded csv-files from using DownloadRaindancer() to a shapefile. The resulting shapefile is being updated every time, this script is being run.*

## Description

Processes downloaded csv-files from using DownloadRaindancer() to a shapefile. The resulting shapefile is being updated every time, this script is being run.

## Usage

```
DownloadRaindancerCombineCharts(
  sourcepath = NA,
  targetpath = NA,
  start_date = paste(substr(Sys.Date(), 1, 4), "-01-01", sep = ""),
  nozzle_diameter = "25_4",
  target_crs = 32633
)
```

## Arguments

| | |
|---|---|
| sourcepath | Path (string) to Firefox download folder. Look it up in your Firefox browser. |
| targetpath | Path (string) to destination folder for downloaded csv-files from Raindancer. |
| start_date | You need to define a start date (default: 1st Jan of recent year) |
| nozzle_diameter | diameter of nozzle in mm (string, e.g. 17_8 = 17.8 mm). Default is "25_4". |
| target_crs | target crs |

## Value

A shapefile, that contains all irrigation events, that were download. The shapefile is being opdated every time this script is being run, as long as all configuration parameter stay the same.

DownloadSentinel2 *Downloads Sentinel-2 satellite data from Copernicus.*

## Description

Downloads Sentinel-2 satellite data from Copernicus.

## Usage

```
DownloadSentinel2(
  target_path = NA,
  shape_site = NA,
  start_date = NA,
  end_date = NA,
  limit = 200,
  max_cld = 10,
  download_NDVI = TRUE
)
```

## Arguments

| | |
|---|---|
| target_path | Path (string) to destination folder for downloaded csv-files from Raindancer. |
| shape_site | Path (string) to shapefile of AOI |
| start_date | You need to define a start date |
| end_date | You need to define an end date |
| limit | Limiter (int) for number of entrys in resulting table |
| max_cld | Cloud cover (int) as percent of maximum cloud coverage above AOI |
| download_NDVI | If TRUE, each Geotiff contains only the NDVI band NDVI compiled images are downloaded. If FALSE each Geotiff contains all available bands. |

## Value

Geotiffs, containing Sentinel-2 data of AOI and every overpass.

# Index

calcWB, 2
calcWBplots, 4

DownloadET0fromArable, 4
DownloadET0fromDWD, 5
DownloadRadolanFromDWD, 6
DownloadRaindancer, 7
DownloadRaindancerCombineCharts, 8
DownloadSentinel2, 8