

# Climate indices with CDO

---

Climate indices of daily temperature and precipitation extremes  
October 2015

Uwe Schulzweida – *MPI for Meteorology*

Ralf Quast – *Brockmann Consult*

---

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Climate indices reference manual</b>	<b>4</b>
2.0.1	ECACDD - Consecutive dry days index per time period . . . . .	6
2.0.2	ECACFD - Consecutive frost days index per time period . . . . .	6
2.0.3	ECACSU - Consecutive summer days index per time period . . . . .	7
2.0.4	ECACWD - Consecutive wet days index per time period . . . . .	7
2.0.5	ECACWDI - Cold wave duration index w.r.t. mean of reference period . . . . .	8
2.0.6	ECACWFI - Cold-spell days index w.r.t. 10th percentile of reference period . . . . .	8
2.0.7	ECAETR - Intra-period extreme temperature range . . . . .	10
2.0.8	ECAFD - Frost days index per time period . . . . .	10
2.0.9	ECAGSL - Thermal Growing season length index . . . . .	11
2.0.10	ECAHD - Heating degree days per time period . . . . .	12
2.0.11	ECAHWDI - Heat wave duration index w.r.t. mean of reference period . . . . .	12
2.0.12	ECAHWFI - Warm spell days index w.r.t. 90th percentile of reference period . . . . .	13
2.0.13	ECAID - Ice days index per time period . . . . .	13
2.0.14	ECAR75P - Moderate wet days w.r.t. 75th percentile of reference period . . . . .	14
2.0.15	ECAR75PTOT - Precipitation percent due to R75p days . . . . .	14
2.0.16	ECAR90P - Wet days w.r.t. 90th percentile of reference period . . . . .	15
2.0.17	ECAR90PTOT - Precipitation percent due to R90p days . . . . .	15
2.0.18	ECAR95P - Very wet days w.r.t. 95th percentile of reference period . . . . .	16
2.0.19	ECAR95PTOT - Precipitation percent due to R95p days . . . . .	16
2.0.20	ECAR99P - Extremely wet days w.r.t. 99th percentile of reference period . . . . .	17
2.0.21	ECAR99PTOT - Precipitation percent due to R99p days . . . . .	17
2.0.22	ECAPD - Precipitation days index per time period . . . . .	18
2.0.23	ECARR1 - Wet days index per time period . . . . .	19
2.0.24	ECARX1DAY - Highest one day precipitation amount per time period . . . . .	19
2.0.25	ECARX5DAY - Highest five-day precipitation amount per time period . . . . .	21
2.0.26	ECASDII - Simple daily intensity index per time period . . . . .	21
2.0.27	ECASU - Summer days index per time period . . . . .	22
2.0.28	ECATG10P - Cold days percent w.r.t. 10th percentile of reference period . . . . .	23
2.0.29	ECATG90P - Warm days percent w.r.t. 90th percentile of reference period . . . . .	23
2.0.30	ECATN10P - Cold nights percent w.r.t. 10th percentile of reference period . . . . .	24
2.0.31	ECATN90P - Warm nights percent w.r.t. 90th percentile of reference period . . . . .	24
2.0.32	ECATR - Tropical nights index per time period . . . . .	25
2.0.33	ECATX10P - Very cold days percent w.r.t. 10th percentile of reference period . . . . .	25
2.0.34	ECATX90P - Very warm days percent w.r.t. 90th percentile of reference period . . . . .	26
	<b>Operator index</b>	<b>28</b>

# 1 Introduction

The Climate Data Operators (**CDO**) software is a collection of operators for standard processing of climate and forecast model data.

This document describes additional **CDO** operators to compute climate indices of daily temperature and precipitation extreme. The definition of these climate indices are from the European Climate Assessment (ECA) project.

The climate indices were implemented in **CDO** by Ralf Quast (Brockmann Consult) on behalf of the Service Gruppe Anpassung (SGA) in 2006. SGA was part of the Model and Data Group (M&D) at the MPI for Meteorology. In 2010, the Model and Data Group became the Data Management department at DKRZ (Deutsches Klimarechenzentrum) and the SGA was disintegrated. For this reason there is no further user support available for these **CDO** operators.

## 2 Climate indices reference manual

This section gives a description of all **CDO** operators to compute the climate indices of daily temperature and precipitation extreme. Related operators are grouped to modules. For easier description all single input files are named `ifile` or `ifile1`, `ifile2`, etc., and an arbitrary number of input files are named `ifiles`. All output files are named `ofile` or `ofile1`, `ofile2`, etc. Further the following notion is introduced:

- $i(t)$      Timestep  $t$  of `ifile`
- $i(t, x)$    Element number  $x$  of the field at timestep  $t$  of `ifile`
- $o(t)$      Timestep  $t$  of `ofile`
- $o(t, x)$    Element number  $x$  of the field at timestep  $t$  of `ofile`

Here is a short overview of all operators in this section:

<a href="#">eca_cdd</a>	Consecutive dry days index per time period
<a href="#">eca_cfd</a>	Consecutive frost days index per time period
<a href="#">eca_csu</a>	Consecutive summer days index per time period
<a href="#">eca_cwd</a>	Consecutive wet days index per time period
<a href="#">eca_cwdi</a>	Cold wave duration index w.r.t. mean of reference period
<a href="#">eca_cwfi</a>	Cold-spell days index w.r.t. 10th percentile of reference period
<a href="#">eca_etr</a>	Intra-period extreme temperature range
<a href="#">eca_fd</a>	Frost days index per time period
<a href="#">eca_gsl</a>	Growing season length index
<a href="#">eca_hd</a>	Heating degree days per time period
<a href="#">eca_hwdi</a>	Heat wave duration index w.r.t. mean of reference period
<a href="#">eca_hwfi</a>	Warm spell days index w.r.t. 90th percentile of reference period
<a href="#">eca_id</a>	Ice days index per time period
<a href="#">eca_r75p</a>	Moderate wet days w.r.t. 75th percentile of reference period
<a href="#">eca_r75ptot</a>	Precipitation percent due to R75p days
<a href="#">eca_r90p</a>	Wet days w.r.t. 90th percentile of reference period
<a href="#">eca_r90ptot</a>	Precipitation percent due to R90p days
<a href="#">eca_r95p</a>	Very wet days w.r.t. 95th percentile of reference period
<a href="#">eca_r95ptot</a>	Precipitation percent due to R95p days

<b>eca_r99p</b>	Extremely wet days w.r.t. 99th percentile of reference period
<b>eca_r99ptot</b>	Precipitation percent due to R99p days
<b>eca_pd</b>	Precipitation days index per time period
<b>eca_r10mm</b>	Heavy precipitation days index per time period
<b>eca_r20mm</b>	Very heavy precipitation days index per time period
<b>eca_rr1</b>	Wet days index per time period
<b>eca_rx1day</b>	Highest one day precipitation amount per time period
<b>eca_rx5day</b>	Highest five-day precipitation amount per time period
<b>eca_sdi</b>	Simple daily intensity index per time period
<b>eca_su</b>	Summer days index per time period
<b>eca_tg10p</b>	Cold days percent w.r.t. 10th percentile of reference period
<b>eca_tg90p</b>	Warm days percent w.r.t. 90th percentile of reference period
<b>eca_tn10p</b>	Cold nights percent w.r.t. 10th percentile of reference period
<b>eca_tn90p</b>	Warm nights percent w.r.t. 90th percentile of reference period
<b>eca_tr</b>	Tropical nights index per time period
<b>eca_tx10p</b>	Very cold days percent w.r.t. 10th percentile of reference period
<b>eca_tx90p</b>	Very warm days percent w.r.t. 90th percentile of reference period

## 2.0.1 ECACDD - Consecutive dry days index per time period

### Synopsis

```
eca_cdd[,R] ifile ofile
```

### Description

Let `ifile` be a time series of the daily precipitation amount `RR`, then the largest number of consecutive days where `RR` is less than `R` is counted. `R` is an optional parameter with default `R = 1 mm`. A further output variable is the number of dry periods of more than 5 days. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile`. The following variables are created:

- `consecutive_dry_days_index_per_time_period`
- `number_of_cdd_periods_with_more_than_5days_per_time_period`

### Parameter

`R`    `FLOAT`    Precipitation threshold (unit: mm; default: `R = 1 mm`)

### Example

To get the largest number of consecutive dry days of a time series of daily precipitation amounts use:

```
cdo eca_cdd rrfile ofile
```

## 2.0.2 ECACFD - Consecutive frost days index per time period

### Synopsis

```
eca_cfd ifile ofile
```

### Description

Let `ifile` be a time series of the daily minimum temperature `TN`, then the largest number of consecutive days where `TN < 0 °C` is counted. Note that `TN` have to be given in units of Kelvin. A further output variable is the number of frost periods of more than 5 days. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile`. The following variables are created:

- `consecutive_frost_days_index_per_time_period`
- `number_of_cfd_periods_with_more_than_5days_per_time_period`

### Example

To get the largest number of consecutive frost days of a time series of daily minimum temperatures use:

```
cdo eca_cfd tnfile ofile
```

### 2.0.3 ECACSU - Consecutive summer days index per time period

#### Synopsis

```
eca_csu[,T] ifile ofile
```

#### Description

Let `ifile` be a time series of the daily maximum temperature TX, then the largest number of consecutive days where  $TX > T$  is counted. The number T is an optional parameter with default  $T = 25^{\circ}\text{C}$ . Note that TN have to be given in units of Kelvin, whereas  $T$  have to be given in degrees Celsius. A further output variable is the number of summer periods of more than 5 days. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile`. The following variables are created:

- consecutive\_summer\_days\_index\_per\_time\_period
- number\_of\_csu\_periods\_with\_more\_than\_5days\_per\_time\_period

#### Parameter

`T`    FLOAT    Temperature threshold (unit:  $^{\circ}\text{C}$ ; default:  $T = 25^{\circ}\text{C}$ )

#### Example

To get the largest number of consecutive summer days of a time series of daily maximum temperatures use:

```
cdo eca_csu txfile ofile
```

### 2.0.4 ECACWD - Consecutive wet days index per time period

#### Synopsis

```
eca_cwd[,R] ifile ofile
```

#### Description

Let `ifile` be a time series of the daily precipitation amount RR, then the largest number of consecutive days where RR is at least  $R$  is counted.  $R$  is an optional parameter with default  $R = 1$  mm. A further output variable is the number of wet periods of more than 5 days. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile`. The following variables are created:

- consecutive\_wet\_days\_index\_per\_time\_period
- number\_of\_cwd\_periods\_with\_more\_than\_5days\_per\_time\_period

#### Parameter

`R`    FLOAT    Precipitation threshold (unit: mm; default:  $R = 1$  mm)

#### Example

To get the largest number of consecutive wet days of a time series of daily precipitation amounts use:

```
cdo eca_cwd rrfile ofile
```

## 2.0.5 ECACWDI - Cold wave duration index w.r.t. mean of reference period

### Synopsis

```
eca_cwdi[,nday[,T]] ifile1 ifile2 ofile
```

### Description

Let `ifile1` be a time series of the daily minimum temperature `TN`, and let `ifile2` be the mean `TNnorm` of daily minimum temperatures for any period used as reference. Then counted is the number of days where, in intervals of at least `nday` consecutive days,  $TN < TNnorm - T$ . The numbers `nday` and `T` are optional parameters with default `nday = 6` and `T = 5°C`. A further output variable is the number of cold waves longer than or equal to `nday` days. `TNnorm` is calculated as the mean of minimum temperatures of a five day window centred on each calendar day of a given climate reference period. Note that both `TN` and `TNnorm` have to be given in the same units. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile1`. The following variables are created:

- `cold_wave_duration_index_wrt_mean_of_reference_period`
- `cold_waves_per_time_period`

### Parameter

<code>nday</code>	INTEGER	Number of consecutive days (default: <code>nday = 6</code> )
<code>T</code>	FLOAT	Temperature offset (unit: °C; default: <code>T = 5°C</code> )

### Example

To compute the cold wave duration index of a time series of daily minimum temperatures use:

```
cdo eca_cwdi tnfile tnnormfile ofile
```

## 2.0.6 ECACWFI - Cold-spell days index w.r.t. 10th percentile of reference period

### Synopsis

```
eca_cwfi[,nday] ifile1 ifile2 ofile
```

### Description

Let `ifile1` be a time series of the daily mean temperature `TG`, and `ifile2` be the 10th percentile `TGn10` of daily mean temperatures for any period used as reference. Then counted is the number of days where, in intervals of at least `nday` consecutive days,  $TG < TGn10$ . The number `nday` is an optional parameter with default `nday = 6`. A further output variable is the number of cold-spell periods longer than or equal to `nday` days. `TGn10` is calculated as the 10th percentile of daily mean temperatures of a five day window centred on each calendar day of a given climate reference period. Note that both `TG` and `TGn10` have to be given in the same units. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile1`. The following variables are created:

- `cold_spell_days_index_wrt_10th_percentile_of_reference_period`
- `cold_spell_periods_per_time_period`

## Parameter

*nday*    INTEGER    Number of consecutive days (default: *nday* = 6)

## Example

To compute the number of cold-spell days of a time series of daily mean temperatures use:

```
cdo eca_cwfi tgfile tgn10file ofile
```

## 2.0.7 ECAETR - Intra-period extreme temperature range

### Synopsis

```
eca_etr ifile1 ifile2 ofile
```

### Description

Let `ifile1` and `ifile2` be time series of the maximum and minimum temperature TX and TN, respectively. Then the extreme temperature range is the difference of the maximum of TX and the minimum of TN. Note that TX and TN have to be given in the same units. The date information of a timestep in `ofile` is the date of the last contributing timesteps in `ifile1` and `ifile2`. The following variables are created:

- `intra_period_extreme_temperature_range`

### Example

To get the intra-period extreme temperature range for two time series of maximum and minimum temperatures use:

```
cdo eca_etr txfile tnfile ofile
```

## 2.0.8 ECAFD - Frost days index per time period

### Synopsis

```
eca_fd ifile ofile
```

### Description

Let `ifile` be a time series of the daily minimum temperature TN, then the number of days where  $TN < 0$  °C is counted. Note that TN have to be given in units of Kelvin. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile`. The following variables are created:

- `frost_days_index_per_time_period`

### Example

To get the number of frost days of a time series of daily minimum temperatures use:

```
cdo eca_fd tnfile ofile
```

## 2.0.9 ECAGSL - Thermal Growing season length index

### Synopsis

```
eca_gsl[,nday[,T[,fland]]] ifile1 ifile2 ofile
```

### Description

Let `ifile1` be a time series of the daily mean temperature TG, and `ifile2` be a land-water mask. Within a period of 12 months, the thermal growing season length is officially defined as the number of days between:

- first occurrence of at least `nday` consecutive days with  $TG > T$
- first occurrence of at least `nday` consecutive days with  $TG < T$  within the last 6 months

On northern hemispere, this period corresponds with the regular year, whereas on southern hemispere, it starts at July 1st. Please note, that this definition may lead to weird results concerning values  $TG = T$ : In the first half of the period, these days do not contribute to the `gsl`, but they do within the second half. Moreover this definition could lead to discontinuous values in equatorial regions.

The numbers `nday` and `T` are optional parameter with default `nday = 6` and `T = 5°C`. The number `fland` is an optional parameter with default value `fland = 0.5` and denotes the fraction of a grid point that have to be covered by land in order to be included in the calculation. A further output variable is the start day of year of the growing season. Note that TG have to be given in units of Kelvin, whereas `T` have to be given in degrees Celsius.

The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile`. The following variables are created:

- `thermal_growing_season_length`
- `day_of_year_of_growing_season_start`

### Parameter

<code>nday</code>	INTEGER	Number of consecutive days (default: <code>nday = 6</code> )
<code>T</code>	FLOAT	Temperature threshold (unit: °C; default: <code>T = 5°C</code> )
<code>fland</code>	FLOAT	Land fraction threshold (default: <code>fland = 0.5</code> )

### Example

To get the growing season length of a time series of daily mean temperatures use:

```
cdo eca_gsl tgfile maskfile ofile
```

## 2.0.10 ECAHD - Heating degree days per time period

### Synopsis

```
eca_hd[,T1[,T2]] ifile ofile
```

### Description

Let `ifile` be a time series of the daily mean temperature `TG`, then the heating degree days are defined as the sum of  $T1 - TG$ , where only values  $TG < T2$  are considered. If  $T1$  and  $T2$  are omitted, a temperature of  $17^{\circ}\text{C}$  is used for both parameters. If only  $T1$  is given,  $T2$  is set to  $T1$ . Note that `TG` have to be given in units of kelvin, whereas  $T1$  and  $T2$  have to be given in degrees Celsius. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile`. The following variables are created:

- `heating_degree_days_per_time_period`

### Parameter

<code>T1</code>	FLOAT	Temperature limit (unit: $^{\circ}\text{C}$ ; default: $T1 = 17^{\circ}\text{C}$ )
<code>T2</code>	FLOAT	Temperature limit (unit: $^{\circ}\text{C}$ ; default: $T2 = T1$ )

### Example

To compute the heating degree days of a time series of daily mean temperatures use:

```
cdo eca_hd tgfile ofile
```

## 2.0.11 ECAHWDI - Heat wave duration index w.r.t. mean of reference period

### Synopsis

```
eca_hwdi[,nday[,T]] ifile1 ifile2 ofile
```

### Description

Let `ifile1` be a time series of the daily maximum temperature `TX`, and let `ifile2` be the mean `TXnorm` of daily maximum temperatures for any period used as reference. Then counted is the number of days where, in intervals of at least `nday` consecutive days,  $TX > TXnorm + T$ . The numbers `nday` and `T` are optional parameters with default `nday = 6` and `T = 5^{\circ}\text{C}`. A further output variable is the number of heat waves longer than or equal to `nday` days. `TXnorm` is calculated as the mean of maximum temperatures of a five day window centred on each calendar day of a given climate reference period. Note that both `TX` and `TXnorm` have to be given in the same units. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile1`. The following variables are created:

- `heat_wave_duration_index_wrt_mean_of_reference_period`
- `heat_waves_per_time_period`

### Parameter

<code>nday</code>	INTEGER	Number of consecutive days (default: <code>nday = 6</code> )
<code>T</code>	FLOAT	Temperature offset (unit: $^{\circ}\text{C}$ ; default: <code>T = 5^{\circ}\text{C}</code> )

## 2.0.12 ECAHWFI - Warm spell days index w.r.t. 90th percentile of reference period

### Synopsis

```
eca_hwfi[,nday] ifile1 ifile2 ofile
```

### Description

Let *ifile1* be a time series of the daily mean temperature TG, and *ifile2* be the 90th percentile TG<sub>90</sub> of daily mean temperatures for any period used as reference. Then counted is the number of days where, in intervals of at least *nday* consecutive days,  $TG > TG_{90}$ . The number *nday* is an optional parameter with default *nday* = 6. A further output variable is the number of warm-spell periods longer than or equal to *nday* days. TG<sub>90</sub> is calculated as the 90th percentile of daily mean temperatures of a five day window centred on each calendar day of a given climate reference period. Note that both TG and TG<sub>90</sub> have to be given in the same units. The date information of a timestep in *ofile* is the date of the last contributing timestep in *ifile1*. The following variables are created:

- warm\_spell\_days\_index\_wrt\_90th\_percentile\_of\_reference\_period
- warm\_spell\_periods\_per\_time\_period

### Parameter

*nday*    INTEGER    Number of consecutive days (default: *nday* = 6)

### Example

To compute the number of warm-spell days of a time series of daily mean temperatures use:

```
cdo eca_hwfi tgfile tgn90file ofile
```

## 2.0.13 ECAID - Ice days index per time period

### Synopsis

```
eca_id ifile ofile
```

### Description

Let *ifile* be a time series of the daily maximum temperature TX, then the number of days where  $TX < 0$  °C is counted. Note that TX have to be given in units of Kelvin. The date information of a timestep in *ofile* is the date of the last contributing timestep in *ifile*. The following variables are created:

- ice\_days\_index\_per\_time\_period

### Example

To get the number of ice days of a time series of daily maximum temperatures use:

```
cdo eca_id txfile ofile
```

## 2.0.14 ECAR75P - Moderate wet days w.r.t. 75th percentile of reference period

### Synopsis

```
eca_r75p ifile1 ifile2 ofile
```

### Description

Let `ifile1` be a time series RR of the daily precipitation amount at wet days (precipitation  $\geq 1$  mm) and `ifile2` be the 75th percentile RRn75 of the daily precipitation amount at wet days for any period used as reference. Then the percentage of wet days with  $RR > RRn75$  is calculated. RRn75 is calculated as the 75th percentile of all wet days of a given climate reference period. Usually `ifile2` is generated by the operator `ydaypct1,75`. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile1`. The following variables are created:

- moderate\_wet\_days\_wrt\_75th\_percentile\_of\_reference\_period

### Example

To compute the percentage of wet days with daily precipitation amount greater than the 75th percentile of the daily precipitation amount at wet days for a given reference period use:

```
cdo eca_r75p rrfile rrn75file ofile
```

## 2.0.15 ECAR75PTOT - Precipitation percent due to R75p days

### Synopsis

```
eca_r75ptot ifile1 ifile2 ofile
```

### Description

Let `ifile1` be a time series RR of the daily precipitation amount at wet days (precipitation  $\geq 1$  mm) and `ifile2` be the 75th percentile RRn75 of the daily precipitation amount at wet days for any period used as reference. Then the ratio of the precipitation sum at wet days with  $RR > RRn75$  to the total precipitation sum is calculated. RRn75 is calculated as the 75th percentile of all wet days of a given climate reference period. Usually `ifile2` is generated by the operator `ydaypct1,75`. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile1`. The following variables are created:

- precipitation\_percent\_due\_to\_R75p\_days

## 2.0.16 ECAR90P - Wet days w.r.t. 90th percentile of reference period

### Synopsis

```
eca_r90p ifile1 ifile2 ofile
```

### Description

Let `ifile1` be a time series RR of the daily precipitation amount at wet days (precipitation  $\geq 1$  mm) and `ifile2` be the 90th percentile RRn90 of the daily precipitation amount at wet days for any period used as reference. Then the percentage of wet days with  $RR > RRn90$  is calculated. RRn90 is calculated as the 90th percentile of all wet days of a given climate reference period. Usually `ifile2` is generated by the operator `ydaypct1,90`. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile1`. The following variables are created:

- `wet_days_wrt_90th_percentile_of_reference_period`

### Example

To compute the percentage of wet days where the daily precipitation amount is greater than the 90th percentile of the daily precipitation amount at wet days for a given reference period use:

```
cdo eca_r90p rrfile rrn90file ofile
```

## 2.0.17 ECAR90PTOT - Precipitation percent due to R90p days

### Synopsis

```
eca_r90ptot ifile1 ifile2 ofile
```

### Description

Let `ifile1` be a time series RR of the daily precipitation amount at wet days (precipitation  $\geq 1$  mm) and `ifile2` be the 90th percentile RRn90 of the daily precipitation amount at wet days for any period used as reference. Then the ratio of the precipitation sum at wet days with  $RR > RRn90$  to the total precipitation sum is calculated. RRn90 is calculated as the 90th percentile of all wet days of a given climate reference period. Usually `ifile2` is generated by the operator `ydaypct1,90`. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile1`. The following variables are created:

- `precipitation_percent_due_to_R90p_days`

## 2.0.18 ECAR95P - Very wet days w.r.t. 95th percentile of reference period

### Synopsis

```
eca_r95p ifile1 ifile2 ofile
```

### Description

Let `ifile1` be a time series RR of the daily precipitation amount at wet days (precipitation  $\geq 1$  mm) and `ifile2` be the 95th percentile RRn95 of the daily precipitation amount at wet days for any period used as reference. Then the percentage of wet days with  $RR > RRn95$  is calculated. RRn95 is calculated as the 95th percentile of all wet days of a given climate reference period. Usually `ifile2` is generated by the operator `ydaypct1,95`. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile1`. The following variables are created:

- `very_wet_days_wrt_95th_percentile_of_reference_period`

### Example

To compute the percentage of wet days where the daily precipitation amount is greater than the 95th percentile of the daily precipitation amount at wet days for a given reference period use:

```
cdo eca_r95p rrfile rrn95file ofile
```

## 2.0.19 ECAR95PTOT - Precipitation percent due to R95p days

### Synopsis

```
eca_r95ptot ifile1 ifile2 ofile
```

### Description

Let `ifile1` be a time series RR of the daily precipitation amount at wet days (precipitation  $\geq 1$  mm) and `ifile2` be the 95th percentile RRn95 of the daily precipitation amount at wet days for any period used as reference. Then the ratio of the precipitation sum at wet days with  $RR > RRn95$  to the total precipitation sum is calculated. RRn95 is calculated as the 95th percentile of all wet days of a given climate reference period. Usually `ifile2` is generated by the operator `ydaypct1,95`. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile1`. The following variables are created:

- `precipitation_percent_due_to_R95p_days`

## 2.0.20 ECAR99P - Extremely wet days w.r.t. 99th percentile of reference period

### Synopsis

```
eca_r99p ifile1 ifile2 ofile
```

### Description

Let `ifile1` be a time series RR of the daily precipitation amount at wet days (precipitation  $\geq 1$  mm) and `ifile2` be the 99th percentile `RRn99` of the daily precipitation amount at wet days for any period used as reference. Then the percentage of wet days with  $RR > RRn99$  is calculated. `RRn99` is calculated as the 99th percentile of all wet days of a given climate reference period. Usually `ifile2` is generated by the operator `ydaypct1,99`. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile1`. The following variables are created:

- `extremely_wet_days_wrt_99th_percentile_of_reference_period`

### Example

To compute the percentage of wet days where the daily precipitation amount is greater than the 99th percentile of the daily precipitation amount at wet days for a given reference period use:

```
cdo eca_r99p rrfile rrn99file ofile
```

## 2.0.21 ECAR99PTOT - Precipitation percent due to R99p days

### Synopsis

```
eca_r99ptot ifile1 ifile2 ofile
```

### Description

Let `ifile1` be a time series RR of the daily precipitation amount at wet days (precipitation  $\geq 1$  mm) and `ifile2` be the 99th percentile `RRn99` of the daily precipitation amount at wet days for any period used as reference. Then the ratio of the precipitation sum at wet days with  $RR > RRn99$  to the total precipitation sum is calculated. `RRn99` is calculated as the 99th percentile of all wet days of a given climate reference period. Usually `ifile2` is generated by the operator `ydaypct1,99`. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile1`. The following variables are created:

- `precipitation_percent_due_to_R99p_days`

## 2.0.22 ECAPD - Precipitation days index per time period

### Synopsis

```
eca_pd,x ifile ofile
eca_r10mm ifile ofile
eca_r20mm ifile ofile
```

### Description

Let `ifile` be a time series of the daily precipitation amount `RR` in [mm] (or alternatively in [kg m<sup>-2</sup>]), then the number of days where `RR` is at least `x` mm is counted. `eca_r10mm` and `eca_r20mm` are specific ECA operators with a daily precipitation amount of 10 and 20 mm respectively. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile`. The following variables are created:

- `precipitation_days_index_per_time_period`

### Operators

<code>eca_pd</code>	Precipitation days index per time period Generic ECA operator with daily precipitation sum exceeding <code>x</code> mm.
<code>eca_r10mm</code>	Heavy precipitation days index per time period Specific ECA operator with daily precipitation sum exceeding 10 mm.
<code>eca_r20mm</code>	Very heavy precipitation days index per time period Specific ECA operator with daily precipitation sum exceeding 20 mm.

### Parameter

<code>x</code>	FLOAT	Daily precipitation amount threshold in [mm]
----------------	-------	--

### Note

Precipitation rates in [mm/s] have to be converted to precipitation amounts (multiply with 86400 s). Apart from metadata information the result of `eca_pd,1` and `eca_rr1` is the same.

### Example

To get the number of days with precipitation greater than 25 mm for a time series of daily precipitation amounts use:

```
cdo eca_pd,25 ifile ofile
```

## 2.0.23 ECARR1 - Wet days index per time period

### Synopsis

```
eca_rr1[R] ifile ofile
```

### Description

Let *ifile* be a time series of the daily precipitation amount RR in [mm] (or alternatively in [kg m<sup>-2</sup>]), then the number of days where RR is at least *R* is counted. *R* is an optional parameter with default *R* = 1 mm. The date information of a timestep in *ofile* is the date of the last contributing timestep in *ifile*. The following variables are created:

- wet\_days\_index\_per\_time\_period

### Parameter

*R*    FLOAT        Precipitation threshold (unit: mm; default: *R* = 1 mm)

### Example

To get the number of wet days of a time series of daily precipitation amounts use:

```
cdo eca_rr1 rrfile ofile
```

## 2.0.24 ECARX1DAY - Highest one day precipitation amount per time period

### Synopsis

```
eca_rx1day[mode] ifile ofile
```

### Description

Let *ifile* be a time series of the daily precipitation amount RR, then the maximum of RR is written to *ofile*. If the optional parameter *mode* is set to 'm' the maximum daily precipitation amounts are determined for each month. The date information of a timestep in *ofile* is the date of the last contributing timestep in *ifile*. The following variables are created:

- highest\_one\_day\_precipitation\_amount\_per\_time\_period

### Parameter

*mode*    STRING        Operation mode (optional). If *mode* = 'm' then maximum daily precipitation amounts are determined for each month

### Example

To get the maximum of a time series of daily precipitation amounts use:

```
cdo eca_rx1day rrfile ofile
```

If you are interested in the maximum daily precipitation for each month, use:

```
cdo eca_rx1day,m rrfilename outfile
```

Apart from metadata information, both operations yield the same as:

```
cdo timmax rrfilename outfile  
cdo monmax rrfilename outfile
```

## 2.0.25 ECARX5DAY - Highest five-day precipitation amount per time period

### Synopsis

```
eca_rx5day[,x] ifile ofile
```

### Description

Let `ifile` be a time series of 5-day precipitation totals `RR`, then the maximum of `RR` is written to `ofile`. A further output variable is the number of 5 day period with precipitation totals greater than  $x$  mm, where  $x$  is an optional parameter with default  $x = 50$  mm. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile`. The following variables are created:

- `highest_five_day_precipitation_amount_per_time_period`
- `number_of_5day_heavy_precipitation_periods_per_time_period`

### Parameter

`x`    `FLOAT`        Precipitation threshold (unit: mm; default:  $x = 50$  mm)

### Example

To get the maximum of a time series of 5-day precipitation totals use:

```
cdo eca_rx5day rrfile ofile
```

Apart from metadata information, the above operation yields the same as:

```
cdo timmax rrfile ofile
```

## 2.0.26 ECASDII - Simple daily intensity index per time period

### Synopsis

```
eca_sdii[,R] ifile ofile
```

### Description

Let `ifile` be a time series of the daily precipitation amount `RR`, then the mean precipitation amount at wet days ( $RR > R$ ) is written to `ofile`.  $R$  is an optional parameter with default  $R = 1$  mm. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile`. The following variables are created:

- `simple_daily_intensity_index_per_time_period`

### Parameter

`R`    `FLOAT`        Precipitation threshold (unit: mm; default:  $R = 1$  mm)

### Example

To get the daily intensity index of a time series of daily precipitation amounts use:

```
cdo eca_sdii rrfile ofile
```

## 2.0.27 ECASU - Summer days index per time period

### Synopsis

```
eca_su[,T] ifile ofile
```

### Description

Let `ifile` be a time series of the daily maximum temperature TX, then the number of days where  $TX > T$  is counted. The number  $T$  is an optional parameter with default  $T = 25^{\circ}\text{C}$ . Note that TX have to be given in units of Kelvin, whereas  $T$  have to be given in degrees Celsius. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile`. The following variables are created:

- `summer_days_index_per_time_period`

### Parameter

$T$     FLOAT        Temperature threshold (unit:  $^{\circ}\text{C}$ ; default:  $T = 25^{\circ}\text{C}$ )

### Example

To get the number of summer days of a time series of daily maximum temperatures use:

```
cdo eca_su txfile ofile
```

## 2.0.28 ECATG10P - Cold days percent w.r.t. 10th percentile of reference period

### Synopsis

```
eca_tg10p ifile1 ifile2 ofile
```

### Description

Let `ifile1` be a time series of the daily mean temperature TG, and `ifile2` be the 10th percentile TGn10 of daily mean temperatures for any period used as reference. Then the percentage of time where  $TG < TGn10$  is calculated. TGn10 is calculated as the 10th percentile of daily mean temperatures of a five day window centred on each calendar day of a given climate reference period. Note that both TG and TGn10 have to be given in the same units. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile1`. The following variables are created:

- `cold_days_percent_wrt_10th_percentile_of_reference_period`

### Example

To compute the percentage of timesteps with a daily mean temperature smaller than the 10th percentile of the daily mean temperatures for a given reference period use:

```
cdo eca_tg10p tgfile tgn10file ofile
```

## 2.0.29 ECATG90P - Warm days percent w.r.t. 90th percentile of reference period

### Synopsis

```
eca_tg90p ifile1 ifile2 ofile
```

### Description

Let `ifile1` be a time series of the daily mean temperature TG, and `ifile2` be the 90th percentile TGn90 of daily mean temperatures for any period used as reference. Then the percentage of time where  $TG > TGn90$  is calculated. TGn90 is calculated as the 90th percentile of daily mean temperatures of a five day window centred on each calendar day of a given climate reference period. Note that both TG and TGn90 have to be given in the same units. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile1`. The following variables are created:

- `warm_days_percent_wrt_90th_percentile_of_reference_period`

### Example

To compute the percentage of timesteps with a daily mean temperature greater than the 90th percentile of the daily mean temperatures for a given reference period use:

```
cdo eca_tg90p tgfile tgn90file ofile
```

## 2.0.30 ECATN10P - Cold nights percent w.r.t. 10th percentile of reference period

### Synopsis

```
eca_tn10p ifile1 ifile2 ofile
```

### Description

Let `ifile1` be a time series of the daily minimum temperature TN, and `ifile2` be the 10th percentile T<sub>Nn10</sub> of daily minimum temperatures for any period used as reference. Then the percentage of time where  $TN < T_{Nn10}$  is calculated. T<sub>Nn10</sub> is calculated as the 10th percentile of daily minimum temperatures of a five day window centred on each calendar day of a given climate reference period. Note that both TN and T<sub>Nn10</sub> have to be given in the same units. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile1`. The following variables are created:

- `cold_nights_percent_wrt_10th_percentile_of_reference_period`

### Example

To compute the percentage of timesteps with a daily minimum temperature smaller than the 10th percentile of the daily minimum temperatures for a given reference period use:

```
cdo eca_tn10p tnfile tnn10file ofile
```

## 2.0.31 ECATN90P - Warm nights percent w.r.t. 90th percentile of reference period

### Synopsis

```
eca_tn90p ifile1 ifile2 ofile
```

### Description

Let `ifile1` be a time series of the daily minimum temperature TN, and `ifile2` be the 90th percentile T<sub>Nn90</sub> of daily minimum temperatures for any period used as reference. Then the percentage of time where  $TN > T_{Nn90}$  is calculated. T<sub>Nn90</sub> is calculated as the 90th percentile of daily minimum temperatures of a five day window centred on each calendar day of a given climate reference period. Note that both TN and T<sub>Nn90</sub> have to be given in the same units. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile1`. The following variables are created:

- `warm_nights_percent_wrt_90th_percentile_of_reference_period`

### Example

To compute the percentage of timesteps with a daily minimum temperature greater than the 90th percentile of the daily minimum temperatures for a given reference period use:

```
cdo eca_tn90p tnfile tnn90file ofile
```

## 2.0.32 ECATR - Tropical nights index per time period

### Synopsis

```
eca_tr[,T] ifile ofile
```

### Description

Let `ifile` be a time series of the daily minimum temperature TN, then the number of days where  $TN > T$  is counted. The number  $T$  is an optional parameter with default  $T = 20^{\circ}\text{C}$ . Note that TN have to be given in units of Kelvin, whereas  $T$  have to be given in degrees Celsius. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile`. The following variables are created:

- `tropical_nights_index_per_time_period`

### Parameter

$T$     FLOAT        Temperature threshold (unit:  $^{\circ}\text{C}$ ; default:  $T = 20^{\circ}\text{C}$ )

### Example

To get the number of tropical nights of a time series of daily minimum temperatures use:

```
cdo eca_tr tnfile ofile
```

## 2.0.33 ECATX10P - Very cold days percent w.r.t. 10th percentile of reference period

### Synopsis

```
eca_tx10p ifile1 ifile2 ofile
```

### Description

Let `ifile1` be a time series of the daily maximum temperature TX, and `ifile2` be the 10th percentile TXn10 of daily maximum temperatures for any period used as reference. Then the percentage of time where  $TX < TXn10$  is calculated. TXn10 is calculated as the 10th percentile of daily maximum temperatures of a five day window centred on each calendar day of a given climate reference period. Note that both TX and TXn10 have to be given in the same units. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile1`. The following variables are created:

- `very_cold_days_percent_wrt_10th_percentile_of_reference_period`

### Example

To compute the percentage of timesteps with a daily maximum temperature smaller than the 10th percentile of the daily maximum temperatures for a given reference period use:

```
cdo eca_tx10p txfile txn10file ofile
```

## 2.0.34 ECATX90P - Very warm days percent w.r.t. 90th percentile of reference period

### Synopsis

```
eca_tx90p ifile1 ifile2 ofile
```

### Description

Let `ifile1` be a time series of the daily maximum temperature TX, and `ifile2` be the 90th percentile TXn90 of daily maximum temperatures for any period used as reference. Then the percentage of time where  $TX > TXn90$  is calculated. TXn90 is calculated as the 90th percentile of daily maximum temperatures of a five day window centred on each calendar day of a given climate reference period. Note that both TX and TXn90 have to be given in the same units. The date information of a timestep in `ofile` is the date of the last contributing timestep in `ifile1`. The following variables are created:

- `very_warm_days_percent_wrt_90th_percentile_of_reference_period`

### Example

To compute the percentage of timesteps with a daily maximum temperature greater than the 90th percentile of the daily maximum temperatures for a given reference period use:

```
cdo eca_tx90p txfile txn90file ofile
```

# Bibliography

[CDI]

[Climate Data Interface](#), from the [Max Planck Institute for Meteorologie](#)

[CDO]

[Climate Data Operators](#), from the [Max Planck Institute for Meteorologie](#)

[ECA]

[ECA indices of extremes](#), from the [Koninklijk Nederlands Meteorologisch Instituut, KNMI](#)

# Operator index

## E

eca_cdd .....	6
eca_cfd .....	6
eca_csu .....	7
eca_cwd .....	7
eca_cwdi .....	8
eca_cwfi .....	8
eca_etr .....	10
eca_fd .....	10
eca_gsl .....	11
eca_hd .....	12
eca_hwdi .....	12
eca_hwfi .....	13
eca_id .....	13
eca_pd .....	18
eca_r10mm .....	18
eca_r20mm .....	18
eca_r75p .....	14
eca_r75ptot .....	14
eca_r90p .....	15
eca_r90ptot .....	15
eca_r95p .....	16
eca_r95ptot .....	16
eca_r99p .....	17
eca_r99ptot .....	17
eca_rr1 .....	19
eca_rx1day .....	19
eca_rx5day .....	21
eca_sdii .....	21
eca_su .....	22
eca_tg10p .....	23
eca_tg90p .....	23
eca_tn10p .....	24
eca_tn90p .....	24
eca_tr .....	25
eca_tx10p .....	25
eca_tx90p .....	26