

RADOLAN-Unterstützung
für Programmierer

Ein grundlegender Java-basierender RADOLAN/RADVOR-Reader - Dokumentation -

Deutscher Wetterdienst
- Abteilung Hydrometeorologie -

Internet:
<http://www.dwd.de/RADOLAN>

Allgemeine Mailadresse:
radolan@dwd.de



1 Überblick

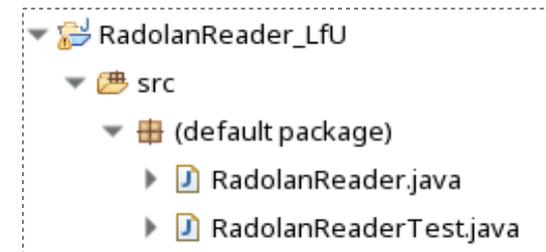
2 Anwendung der Reader-Klasse

3 Anwendung der Test-Klasse

- Der **RadolanReader** liest RADOLAN- und RADVOR-Produkte
- Version: 2.0
- bekannte Nutzer:
 - LfU Augsburg

Bestandteile:

- mit dem eigentlichen **Reader** enthalten ist
 - eine Test-Klasse **RadolanReaderTest**
 - (Start-Shellscript: **start.sh**)
 - diese kleine **Dokumentation**
-
- getestet mit Java 1.8



1 Überblick

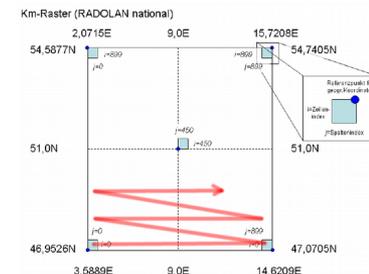
Änderungen: Version 2.0

```
23 *
24 * Anpassungen (Historie mit der Testklasse 'RadolanReaderTest' abgleichen)
25 * -----
26 * 2019-03-07: Lesen des RADVOR-RE-Formats eingebaut (Anfrage LfU)
27 *           da größere Umbauarbeiten an den Methoden und Attributen -
28 *           mit Wegfall der Klasse 'Pixel' großer Versionssprung
29 *           -> Version 2.0
30 * 2017-03-28: RW-Tests und RQ-Kompatibilitätstest (LfU möchte damit auch RADV
31 *           -> Version 1.0
32 * ----- */
33 public class RadolanReader {
34 »
35     //final String VERSION = "1.0";
36     public static final String VERSION = "2.0";    // 2019/03
37     ....
38     ..
```

Versionssprung auf 2.0 aufgrund tiefgreifender Änderungen.

Die Anpassungen gegenüber der Version vor dem Jahr 2019 sind:

- Die Readerklasse ist nun sehr viel einfacher designed:
 - Wegfall der Pixel-Klasse
 - Wegfall der internen Header-Klasse
- Ausgabe von Metadaten (aus dem Header) direkt nach dem Lesen des Produkts
- automatische Drehung des gelesenen Feldes mit Norden oben. (Die Werte am Anfang eines RADOLAN/RADVOR-Produkts beginnen von der südwestlichen Ecke aus).
- Zugriff neben dem Integer-Feld auch auf ein Float-Feld (reale Größen)



Anwendung der Reader-Klasse

- Dieses Tutorial beschreibt die Verwendung der RADOLAN-Reader-Klasse aus einer Java-Umgebung heraus.

```
// Init. RadolanReader:
rr = new RadolanReader("<path_to_RW>");    // file open + read header

// Reader konfigurieren:
//rr.raw_values(true);    // Werte als Roh-Werte nehmen

// Feld lesen:
int[][] int_field = rr.readAll();    // Integer-Feld. Datei wird
dabei geschlossen.
// oder später:
// rr.int_field();

// int-field wird automatisch gedreht, sodass die Werte
// im Norden am Anfang des Arrays stehen.

// Zugriff auf Float-Feld (Kommawerte):
float[][] float_field = rr.float_field()

// Bei Bedarf:

// Metadaten zum Produkt ausgeben:
rr.print_info();
```

Dem Konstruktor wird der Pfad zu einer RADOLAN/RADVOR-Datei übergeben. Hierbei wird bereits automatisch der Text-Header vor dem Binärteil gelesen, der Informationen zum Produkt enthält.

Durch Aufruf der Methode

- `rr.info()`;
- werden einige Informationen aus dem Header ausgegeben.

Lesemethode auswählen: ganzes Feld oder Gebiet daraus lesen.

- `rr.readAll()`
- `rr.readArea()`

Die Dateiverbindung wird dabei geschlossen.

Nach dem Lesen wird das Feld automatisch gedreht, sodass Norden oben ist, so wie man sich die Orientierung vorstellt. Die Datei beginnt nämlich mit den Werten in der südwestlichen Ecke.

Nach Anwendung einer Lesemethode stehen die gelesenen Werte als Integer-Feld zur Verfügung: direkt

- `int[][] int_field = rr.readAll()`
- oder später:
- `rr.int_field()`

Benötigt man das Feld als Kommazahlen (= reale Größen), kann auf das Feld als Float-Feld zugegriffen werden:

- `rr.float_field()`

