

# Package ‘affyio’

August 10, 2022

**Version** 1.67.0

**Title** Tools for parsing Affymetrix data files

**Author** Ben Bolstad <bmb@bmbolstad.com>

**Maintainer** Ben Bolstad <bmb@bmbolstad.com>

**Depends** R (>= 2.6.0)

**Imports** zlibbioc, methods

**Description** Routines for parsing Affymetrix data files based upon file format information. Primary focus is on accessing the CEL and CDF file formats.

**License** LGPL (>= 2)

**URL** <https://github.com/bmbolstad/affyio>

**biocViews** Microarray, DataImport, Infrastructure

**LazyLoad** yes

**git\_url** <https://git.bioconductor.org/packages/affyio>

**git\_branch** master

**git\_last\_commit** bbf6af3

**git\_last\_commit\_date** 2022-04-26

**Date/Publication** 2022-08-10

## R topics documented:

check.cdf.type . . . . .	2
get.celfile.dates . . . . .	2
read.cdffile.list . . . . .	3
read.celfile . . . . .	4
read.celfile.header . . . . .	4
read.celfile.probeintensity.matrices . . . . .	5

<b>Index</b>	<b>7</b>
--------------	----------

---

check.cdf.type	<i>CDF file format function</i>
----------------	---------------------------------

---

**Description**

This function returns a text string giving the file format for the supplied filename

**Usage**

```
check.cdf.type(filename)
```

**Arguments**

filename	fullpath to a cdf file
----------	------------------------

**Value**

Returns a string which is currently one of:

text	the cdf file is of the text format
xda	the cdf file is of the binary format used in GCOS
unknown	the parser can not handle this format or does not recognize this file as a CDF file

**Author(s)**

B. M. Bolstad <bmb@bmbolstad.com>

---

get.celfile.dates	<i>Extract Dates from CEL files</i>
-------------------	-------------------------------------

---

**Description**

This function reads the header information for a series of CEL files then extracts and returns the dates.

**Usage**

```
get.celfile.dates(filenamees, ...)
```

**Arguments**

filenamees	a vector of characters with the CEL filenames. May be fully pathed.
...	further arguments passed on to <a href="#">read.celfile.header</a> .

**Details**

The function uses [read.celfile.header](#) to read in the header of each file. The ScanDate component is then parsed to extract the date. Note that an assumption is made about the format. Namely, that dates are in the Y-m-d or m/d/y format.

**Value**

A vector of class `Date` with one date for each celfile.

**Author(s)**

Rafael A. Irizarry <rafa@jimmy.harvard.edu>

**See Also**

See Also as [read.celfile.header](#).

---

read.cdffile.list      *Read CDF file into an R list*

---

**Description**

This function reads the entire contents of a cdf file into an R list structure

**Usage**

```
read.cdffile.list(filename, cdf.path = getwd())
```

**Arguments**

filename	name of CDF file
cdf.path	path to cdf file

**Details**

Note that this function can be very memory intensive with large CDF files.

**Value**

returns a list structure. The exact contents may vary depending on the file format of the cdf file (see [check.cdf.type](#))

**Author(s)**

B. M. Bolstad <bmb@bmbolstad.com>

read.celfile      *Read a CEL file into an R list*

---

### **Description**

This function reads the entire contents of a CEL file into an R list structure

### **Usage**

```
read.celfile(filename,intensity.means.only=FALSE)
```

### **Arguments**

filename      name of CEL file  
intensity.means.only      If TRUE then read on only the MEAN section in INTENSITY

### **Details**

The list has four main items. HEADER, INTENSITY, MASKS, OUTLIERS. Note that INTENSITY is a list of three vectors MEAN, STDEV, NPIXELS. HEADER is also a list. Both of MASKS and OUTLIERS are matrices.

### **Value**

returns a list structure. The exact contents may vary depending on the file format of the CEL file

### **Author(s)**

B. M. Bolstad <bmb@bmbolstad.com>

---

read.celfile.header      *Read header information from cel file*

---

### **Description**

This function reads some of the header information (appears before probe intensity data) from the supplied cel file.

### **Usage**

```
read.celfile.header(filename,info=c("basic","full"),verbose=FALSE)
```

**Arguments**

filename	name of CEL file. May be fully pathed
info	A string. <code>basic</code> returns the dimensions of the chip and the name of the CDF file used when the CEL file was produced. <code>full</code> returns more information in greater detail.
verbose	a <a href="#">logical</a> . When true the parsing routine prints more information, typically useful for debugging.

**Value**

A list data structure.

**Author(s)**

B. M. Bolstad <bmb@bmbolstad.com>

---

read.celfile.probeintensity.matrices

*Read PM or MM from CEL file into matrices*

---

**Description**

This function reads PM, MM or both types of intensities into matrices. These matrices have all the probes for a probeset in adjacent rows

**Usage**

```
read.celfile.probeintensity.matrices(filenamees, cdfInfo, rm.mask=FALSE, rm.outliers=FALSE, rm.extra=F
```

**Arguments**

filenamees	a character vector of filenames
cdfInfo	a list with items giving PM and MM locations for desired probesets. In same structure as returned by <a href="#">make.cdf.package</a>
rm.mask	a <a href="#">logical</a> . Return these probes as NA if there are in the [MASK] section of the CEL file
rm.outliers	a <a href="#">logical</a> . Return these probes as NA if there are in the [OUTLIERS] section of the CEL file.
rm.extra	a <a href="#">logical</a> . Return these probes as NA if there are in the [OUTLIERS] section of the CEL file.
verbose	a <a href="#">logical</a> . When true the parsing routine prints more information, typically useful for debugging.
which	a string specifying which probe type to return

**Value**

returns a [list](#) of [matrix](#) items. One matrix contains PM probe intensities, with probes in rows and arrays in columns

**Author(s)**

B. M. Bolstad <bmb@bmbolstad.com>

# Index

## \* IO

- check.cdf.type, [2](#)
- get.celfile.dates, [2](#)
- read.cdffile.list, [3](#)
- read.celfile, [4](#)
- read.celfile.header, [4](#)
- read.celfile.probeintensity.matrices,  
[5](#)

check.cdf.type, [2, 3](#)

Date, [3](#)

get.celfile.dates, [2](#)

list, [6](#)

logical, [5](#)

make.cdf.package, [5](#)

matrix, [6](#)

read.cdffile.list, [3](#)

read.celfile, [4](#)

read.celfile.header, [2, 3, 4](#)

read.celfile.probeintensity.matrices,  
[5](#)