

# affy: Import Methods (HowTo)

Laurent

October 26, 2023

## Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>How-to</b>	<b>1</b>
2.1	Cel objects . . . . .	1
2.2	AffyBatch objects . . . . .	2

## 1 Introduction

This document describes briefly how to write import methods for the `affy` package. As one might know, the Affymetrix data are originally stored in files of type `.CEL` and `.CDF`. The package extracts and store the information contained in R data structures using file parsers. This document outlines how to get the data from other sources than the current<sup>1</sup> file formats.

As usual, loading the package in your R session is required.

```
R> library(affy) ##load the affy package
```

**note: this document only describes the process for `.CEL` files**

Knowing the slots of `Cel` and `AffyBatch` objects will probably help you to achieve your goals. You may want to refer to the respective help pages. Try `help(Cel)`, `help(AffyBatch)`.

## 2 How-to

### 2.1 Cel objects

The functions `getNrowForCEL` and `getNcolForCEL` are assumed to return the number of rows and the number of columns in the `.CEL` file respectively

---

<sup>1</sup>today's date is early 2003

You will also need to have access to the  $X$  and  $Y$  position for the probes in the CEL file. The functions `getPosXForCEL` and `getPosYForCEL` are assumed to return the  $X$  and  $Y$  positions respectively. The corresponding probe intensities are assumed to be returned by the function `getIntensitiesForCEL`.

If you stored **all** the  $X$  and  $Y$  values that were in the .CEL, the functions `getNrowForCEL` and `getNcolForCEL` can be written:

```
> getNrowForCEL <- function() max(getPosXForCEL())
> getNcolForCEL <- function() max(getPosYForCEL())
```

You will also need the name for the corresponding .CDF (although you will probably not need the .CDF file itself, the cdf packages available for download will probably be enough).

```
import.celfile <- function(celfile, ...) {

  cel.nrow <- getNrowForCEL(celfile)
  cel.ncol <- getNcolForCEL(celfile)
  x <- matrix(NA, nr=cel.nrow, nc=cel.ncol)

  cel.intensities <- getIntensitiesForCEL(celfile)

  cel.posx <- getPosXForCEL(celfile) # +1 if indexing starts at 0 (like in .CEL)
  cel.posy <- getPosYForCEL(celfile) # idem

  x[cbind(cel.posx, cel.posy)] <- cel.intensities

  mycdfName <- whatcdf("aCELfile.CEL")

  myCel <- new("Cel", exprs=x, cdfName=mycdfName)

  return(myCel)
}
```

The function `import.celfile` can now replace the function `read.celfile` in the `affy` package

## 2.2 AffyBatch objects

(scratch) the use of `...` should make you able to override the function `read.celfile` by a hack like:

```
read.celfile <- import.celfile
```

The function `read.affybatch` should now function using your `import.celfile`