

# Direct access to mass spectrometry data in R

Christian Panse and Tobias Kockmann

European Bioconductor Meeting, de Duve Institute, UCLouvain, Campus de Woluwé Brussels, Belgium, Dez 2019

<https://fgcz.ch/>

# Recently on the R for Mass Spectrometry community -mtime -365

- rawDiag: access Orbitrap data on the fast lane (European Bioconductor Meeting 2018)

Journal of  
**proteome**  
research

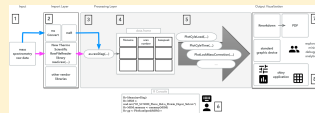
Cite This: J. Proteome Res. 2018, 17, 2008–2014

Technical Note  
pubs.acs.org/jpr

## rawDiag: An R Package Supporting Rational LC–MS Method Optimization for Bottom-up Proteomics

Christian Trachsel, Christian Panse,\* Tobias Kockmann, Witold E. Wolski, Jonas Grossmann, and Ralph Schlapbach

Functional Genomics Center Zurich, Swiss Federal Institute of Technology Zurich, University of Zurich, Winterthurerstr. 190, CH-8057 Zurich, Switzerland

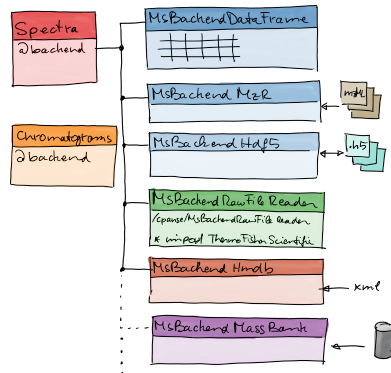


**ABSTRACT:** Optimizing methods for liquid chromatography coupled to mass spectrometry (LC–MS) is a nontrivial task. Here we present rawDiag, a software tool supporting rational method optimization by providing MS operator-tailored diagnostic plots of scan-level metadata. rawDiag is implemented as an R package and can be executed on the R command line or through a graphical user interface (GUI) for less experienced users. The code runs platform-independent and can process 100 raw files in <3 min on current consumer hardware, as we show in our benchmark. As a demonstration of the functionality of our package we include a real-world example taken from our daily core facility business.

**KEYWORDS:** mass spectrometry, R package, multipplatform, method optimization, reproducible research, quality control, visualization

# Recently on the R for Mass Spectrometry community -mtime -365

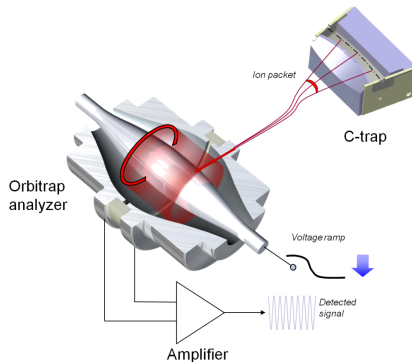
- ▶ rawDiag: access Orbitrap data on the fast lane (European Bioconductor Meeting 2018)
- ▶ rformassspectrometry.org



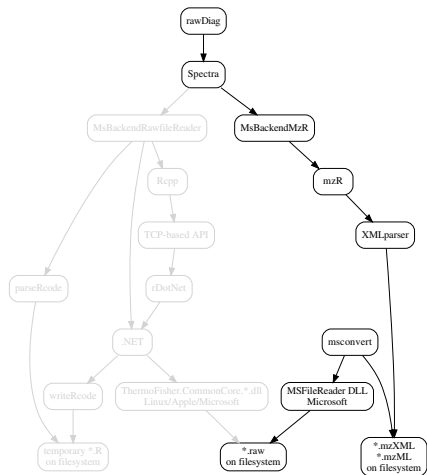
source: [https://github.com/jorainer/swemsa\\_2019](https://github.com/jorainer/swemsa_2019)

# Recently on the R for Mass Spectrometry community -mtime -365

- ▶ rawDiag: access Orbitrap data on the fast lane (European Bioconductor Meeting 2018)
- ▶ rformassspectrometry.org
- ▶ Why? - 80% of today's mass spectrometer in proteomics are Orbitraps (Thermo Fisher Scientific).

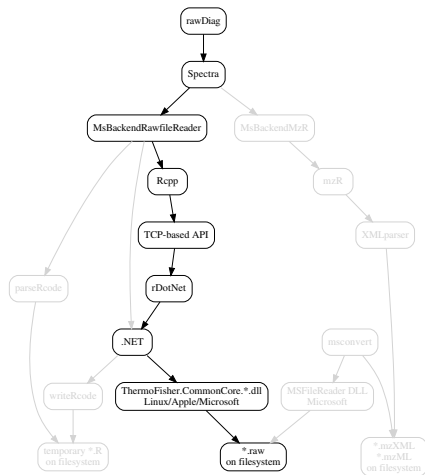


source: <https://en.wikipedia.org/wiki/Orbitrap>



- 😊 HUPO PSI
- 😞 conversion triggered by an external tool
- 😞 ReAdW.exe / msconvert work on Microsoft only
- 😞 mzXML file is not always mzXML - different converter; different options
- 😞 mzXML does not contain all the data provided by the instrument. (see poster)

## MsBackendRawfileReader



## Usage - Spectra::backendInitialize

```
R> library(MsBackendRawFileReader)
R> be <- backendInitialize( MsBackendRawFileReader(), files = rawfile)
```

## Usage - Spectra::backendInitialize

```
R> library(MsBackendRawFileReader)
R> be <- backendInitialize( MsBackendRawFileReader(), files = rawfile)
R> (s <- Spectra(be))
```

```
MSn data (Spectra) with 8742 spectra in a MsBackendRawFileReader backend:
filename:          /Users/cp/Library/Caches/ExperimentHub/24ab678291f6_3238.1
creation date:      7/16/2019 5:56:24 PM
first scan:         1
last scan:          8742
model:              Orbitrap Fusion Lumos
name:               Orbitrap Fusion Lumos
SerialNumber:       FSN20583
```

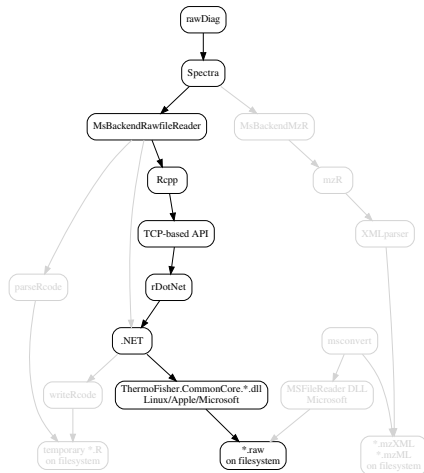
Processing:



# Usage - R package rawDiag – now running on Spectra objects.

Fast and colorful diagnostic plot functions – designed by, and for, mass spectrometer experts

```
R> class(s)
[1] "Spectra"
attr(,"package")
[1] "Spectra"
```



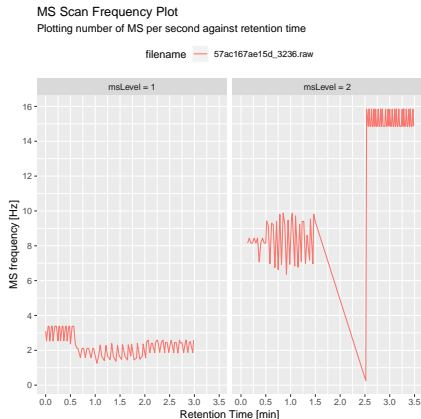
# Usage - R package rawDiag – now running on Spectra objects.

Fast and colorful diagnostic plot functions – designed by, and for, mass spectrometer experts

```
R> class(s)

[1] "Spectra"
attr(,"package")
[1] "Spectra"

R> library(rawDiag)
R> Plot(s, FUN='ScanFrequency')
```



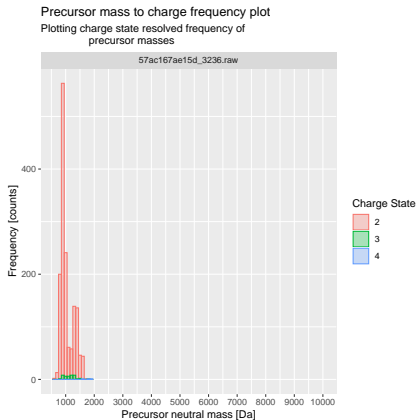
# Usage - R package rawDiag – now running on Spectra objects.

Fast and colorful diagnostic plot functions – designed by, and for, mass spectrometer experts

```
R> class(s)

[1] "Spectra"
attr(,"package")
[1] "Spectra"

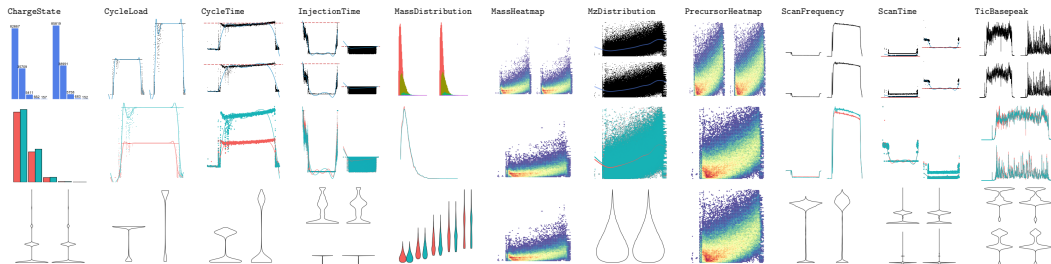
R> library(rawDiag)
R> Plot(s, FUN='ScanFrequency')
R> Plot(s, FUN='MassDistribution')
...
```



# Usage - R package rawDiag – now running on Spectra objects.

Fast and colorful diagnostic plot functions – designed by, and for, mass spectrometer experts

## cheatsheet of plot methods:



PMID: 29978702 DOI: 10.1021/acs.jproteome.8b00173

## C# listings – accessing .NET vendor libraries

<https://github.com/cpanse/MsBackendRawFileReader/blob/Bioc3.10/src/MsBackendRawFileReader.cs>

```
12 using System;
13 using System.Collections.Generic;
14 using System.Diagnostics;
15 using System.IO;
16 using System.Runtime.ExceptionServices;
17 using System.Collections;
18 using System.Linq;
19 using ThermoFisher.CommonCore.Data;
20 using ThermoFisher.CommonCore.Data.Business;
21 using ThermoFisher.CommonCore.Data.FilterEnums;
22 using ThermoFisher.CommonCore.Data.Interfaces;
23 using ThermoFisher.CommonCore.MassPrecisionEstimator;
24 using ThermoFisher.CommonCore.RawFileReader;
```

## C# listings – accessing .NET vendor libraries

<https://github.com/cpanse/MsBackendRawFileReader/blob/Bioc3.10/src/MsBackendRawFileReader.cs>

```
27 namespace MsBackendRawFileReader  
28 {
```

## C# listings – accessing .NET vendor libraries

<https://github.com/cpanse/MsBackendRawFileReader/blob/Bioc3.10/src/MsBackendRawFileReader.cs>

```
27 namespace MsBackendRawFileReader  
28 {
```

```
46     public class Rawfile  
47     {
```

## C# listings – accessing .NET vendor libraries

<https://github.com/cpanse/MsBackendRawFileReader/blob/Bioc3.10/src/MsBackendRawFileReader.cs>

```
27 namespace MsBackendRawFileReader
28 {
46     public class Rawfile
47     {
53         // ThermoRawFileParser accessing the RAW file
54         private IRawDataPlus rawFile;
55
56         public Rawfile(string rawfile)
57         {
```



## C# listings – accessing .NET vendor libraries

<https://github.com/cpanse/MsBackendRawFileReader/blob/Bioc3.10/src/MsBackendRawFileReader.cs>

```
27 namespace MsBackendRawFileReader
28 {
    46     public class Rawfile
    47     {
        53         // ThermoRawFileParser accessing the RAW file
        54         private IRawDataPlus rawFile;
        55
        56         public Rawfile(string rawfile)
        57         {
            620             public double[] GetSpectrumMasses(int scanNumber)
            621             {
                622                 var scanStatistics = rawFile.GetScanStatsForScanNumber(scanNumber);
                623                 var centroidStream = rawFile.GetCentroidStream(scanNumber, false);
                624
                625                 if (scanStatistics.IsCentroidScan && centroidStream.Length > 0)
                626                 {
                    627                     return centroidStream.Masses.ToArray();
                    628                 }
                629                 else
                630                 {
                    631                     var segmentedScan = rawFile.GetSegmentedScanFromScanNumber(scanNumber, scanStatistics);
                    632                     return segmentedScan.Positions.ToArray();
                    633                 }
                634             }
        }
```

# Calling the C# methods from R – through using rDotNet

...running somewhere in the MsBackend

```
R> ## https://CRAN.R-project.org/package=rDotNet
```

```
R> ## create an object and call a method
```

```
R> (x <- .cnew ("Rawfile", rawfile))
```

```
<dotnet obj: 448, class: MsBackendRawFileReader.Rawfile, value: "MsBackendRawFileReader.Rawfile"
```

# Calling the C# methods from R – through using rDotNet

...running somewhere in the MsBackend

```
R> ## https://CRAN.R-project.org/package=rDotNet
```

```
R> ## create an object and call a method
```

```
R> (x <- .cnew ("Rawfile", rawfile))
```

```
<dotnet obj: 448, class: MsBackendRawFileReader.Rawfile, value: "MsBackendRawFileReader.Rawfile"
```

```
R> ## getting a vector of mass values
```

```
R> head(mZ <- x$GetSpectrumMasses(4034))
```

```
[1] 110.0709 111.0744 114.2004 120.0805 129.1020 138.0659
```

# Calling the C# methods from R – through using rDotNet

...running somewhere in the MsBackend

```
R> ## https://CRAN.R-project.org/package=rDotNet
```

```
R> ## create an object and call a method
```

```
R> (x <- .cnew ("Rawfile", rawfile))
```

```
<dotnet obj: 448, class: MsBackendRawFileReader.Rawfile, value: "MsBackendRawFileReader.Rawfile"
```

```
R> ## getting a vector of mass values
```

```
R> head(mZ <- x$GetSpectrumMasses(4034))
```

```
[1] 110.0709 111.0744 114.2004 120.0805 129.1020 138.0659
```

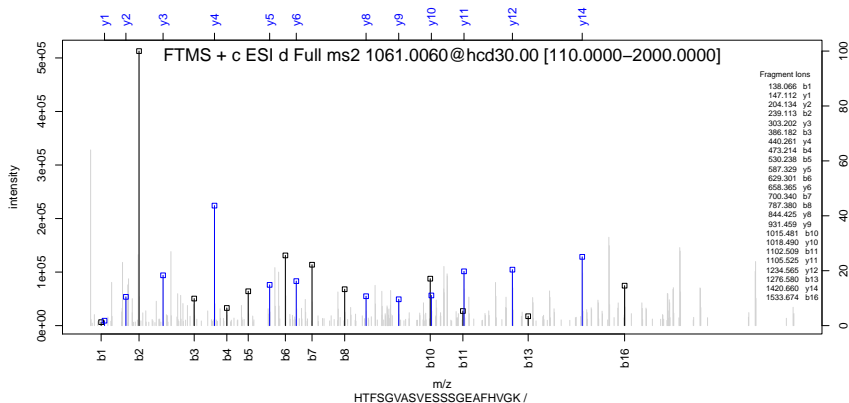
```
R> ## getting a vector of intensities
```

```
R> head(intensity <- x$GetSpectrumIntensities(4034))
```

```
[1] 328132.188 11543.382 4682.475 20583.205 7574.656
```

```
[6] 6460.671
```

```
R> # https://CRAN.R-project.org/package=protViz
R> pp <- protViz::peakplot("HTFSGVASVESSSGEAFHVGK",
+   list(mZ = mZ, intensity = intensity),
+   FUN=function(b,y){cbind(b,y)}, itol=0.01, col='lightgrey')
R> legend("top", "", title=x$GetScanFilter(4034), cex=1.5, bty = "n")
```



## Extracted Ion Chromatogram (XIC)

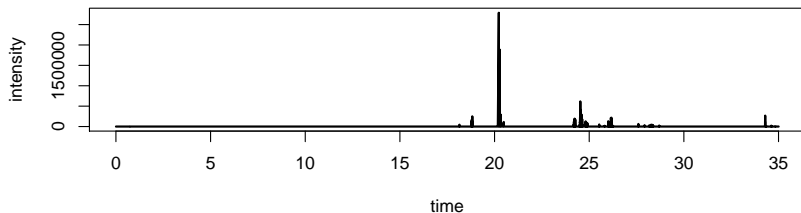
```
R> (mZ <- (protViz::parentIonMass("HTFSGVASVESSSGEAFHVGK") + 1.008) / 2)  
[1] 1060.506
```

## Extracted Ion Chromatogram (XIC)

```
R> (mZ <- (protViz::parentIonMass("HTFSGVASVESSSGEAFHVGK") + 1.008) / 2)
[1] 1060.506
R> X <- x$GetXIC(mZ, tol=5, filter="ms")
```

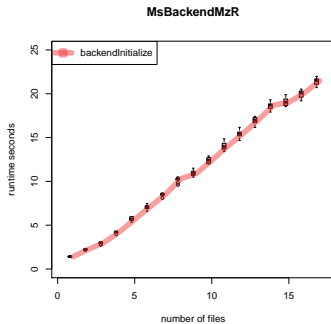
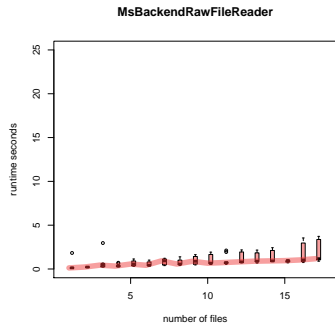
## Extracted Ion Chromatogram (XIC)

```
R> (mZ <- (protViz::parentIonMass("HTFSGVASVESSSGEAFHVGK") + 1.008) / 2)
[1] 1060.506
R> X <- x$GetXIC(mZ, tol=5, filter="ms")
R> idx <- seq(1, length(X), by=2)
R> plot(X[idx], X[-idx],
+       lwd=2, type='l', xlab='time', ylab='intensity')
```



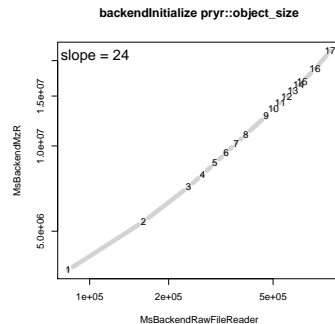
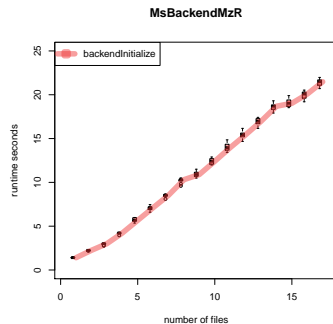
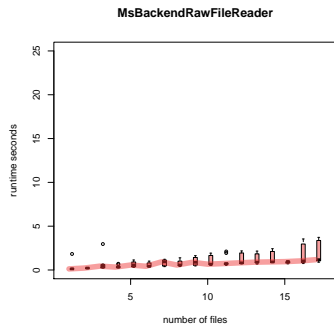


## calling backendInitialize



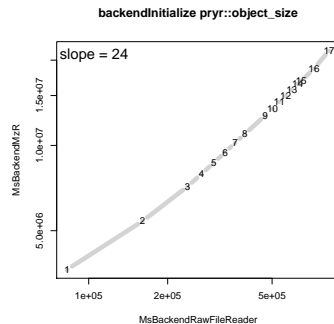
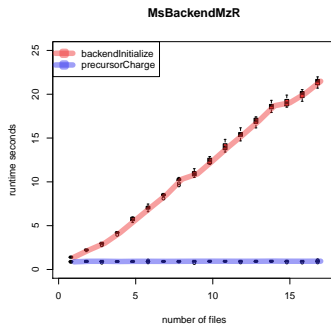
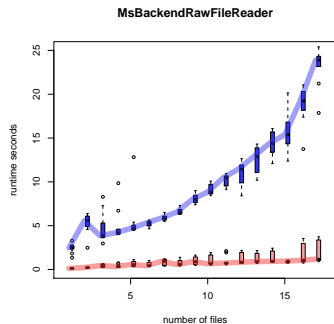
single core; 17 mzXML files size 5.4GB; 17 raw files size 2.5GB

## backendInitialize and estimate of the memory



single core; 17 mzXML files size 5.4GB; 17 raw files size 2.5GB

...calling Spectra::precursorCharge



single core; 17 mzXML files size 5.4GB; 17 raw files size 2.5GB

- ▶ synchronize with community development (Spectra and Chromatograms).

- ▶ synchronize with community development (Spectra and Chromatograms).

## when submitting the MsBackendRawFileReader to Bioconductor

- ▶ How do we handle policy with vendor libraries?
- ▶ How do we handle the rDotNet webservice in the build system?

```
/usr/bin/mono \  
--nollvm /home/cp/R/x86_64-pc-linux-gnu-library/3.6/rDotNet/server/bin/Debug/CLRServer.exe \  
-url svc://localhost:56789/ \  
-dll /home/cp/R/x86_64-pc-linux-gnu-library/3.6/MsBackendRawFileReader/exec/MsBackendRawFileReader.dll
```



## Bookmarks

- ▶ <http://planetorbitrap.com/rawfilereader#.WjkqIUtJmL4>
- ▶ <https://CRAN.R-project.org/package=rDotNet>
- ▶ <https://github.com/rformassspectrometry/>
- ▶ <https://github.com/cpanse/MsBackendRawFileReader>
- ▶ <https://bioconductor.org/packages/tartare/>

# Session information

- ▶ R version 3.6.1 (2019-07-05), x86\_64-apple-darwin15.6.0
- ▶ Locale: en\_US.UTF-8/en\_US.UTF-8/en\_US.UTF-8/C/en\_US.UTF-8/en\_US.UTF-8
- ▶ Running under: macOS Catalina 10.15.1
- ▶ Matrix products: default
- ▶ BLAS: /Library/Frameworks/R.framework/Versions/3.6/Resources/lib/libRblas.0.dylib
- ▶ LAPACK: /Library/Frameworks/R.framework/Versions/3.6/Resources/lib/libRlapack.dylib
- ▶ Base packages: base, datasets, graphics, grDevices, methods, parallel, stats, stats4, utils
- ▶ Other packages: BiocGenerics 0.32.0, BiocParallel 1.19.0, IRanges 2.19.10, MsBackendRawFileReader 0.0.1, ProtGenerics 1.17.4, protViz 0.5.1, rDotNet 0.9.1, S4Vectors 0.24.0, Spectra 0.3.0
- ▶ Loaded via a namespace (and not attached): codetools 0.2-16, compiler 3.6.1, magrittr 1.5, MsCoreUtils 0.0.2, R6 2.4.1, Rcpp 1.0.3, rlang 0.4.2, testthat 2.1.1, tools 3.6.1